



HAL
open science

Pseudo-healthy image reconstruction with deep generative models for the detection of dementia-related anomalies

Ravi Hassanaly

► **To cite this version:**

Ravi Hassanaly. Pseudo-healthy image reconstruction with deep generative models for the detection of dementia-related anomalies. Medical Imaging. Sorbonne Université, 2024. English. NNT : 2024SORUS118 . tel-04681117

HAL Id: tel-04681117

<https://theses.hal.science/tel-04681117v1>

Submitted on 29 Aug 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

SORBONNE UNIVERSITÉ

DOCTORAL THESIS

Pseudo-healthy image reconstruction with
deep generative models for the detection of
dementia-related anomalies

Author:

Ravi HASSANALY

Supervisors:

Ninon BURGOS

Olivier COLLIOT

Referees:

Carole LARTIZIEN

Shadi ALBARQOUNI

Jean-François MANGIN

*A thesis submitted in fulfilment of the requirements
for the degree of PhD in Computer Science*

in the

ARAMIS Lab

Sorbonne Université, Institut du Cerveau - Paris Brain Institute (ICM), CNRS,
Inria, Inserm, AP-HP Hôpital de la Pitié Salpêtrière

April 30, 2024



Abstract

Neuroimaging has become an essential tool in the study of markers of Alzheimer’s disease. However, analyzing complex multimodal brain images remains a major challenge for clinicians. To overcome this difficulty, deep learning methods have emerged as a promising solution for the automatic and robust analysis of neuroimaging data.

In this thesis, we explore the use of deep generative models for the detection of anomalies associated with dementia in ^{18}F -fluorodeoxyglucose positron emission tomography (FDG PET) data. Our method is based on the principle of pseudo-healthy reconstruction, where we train a generative model to reconstruct healthy images from pathological data. This approach has the advantage of not requiring annotated data, which are time-consuming and costly to acquire, as well as being generalizable to different types of anomalies. We chose to implement a variational autoencoder (VAE), a simple model, but that proved its worth in the field of deep learning. However, assessing the performance of our generative models without labeled data or ground truth anomaly maps leads to an incomplete evaluation.

To solve this issue, we have introduced an evaluation framework based on the simulation of hypometabolism on FDG PET images. Thus, by creating pairs of healthy and diseased images, we are able to assess the model’s ability to reconstruct pseudo-healthy images. In addition, this methodology has enabled us to define new metrics for assessing the quality of reconstructions obtained from generative models. The evaluation framework allowed us to carry out a comparative study on twenty VAE variants in the context of FDG PET pseudo-healthy reconstruction. The proposed benchmark enabled us to identify the best-performing models for detecting dementia-related anomalies.

Finally, several significant contributions have been made to open-source software. A PET image processing pipeline has been integrated into the Clinica software. In addition, this thesis gave rise to numerous contributions to the development of the ClinicaDL software, including its improvement, the addition of new functionalities, software maintenance and participation in project management.

Résumé

La neuroimagerie est devenue un outil essentiel dans l'étude des marqueurs de la maladie d'Alzheimer. Cependant, l'analyse de ces images complexes provenant de différentes modalités d'imagerie cérébrale reste un défi majeur pour les cliniciens. Pour surmonter cette difficulté, les méthodes de deep learning ont émergé comme une solution prometteuse pour l'analyse automatique et robuste des données de neuroimagerie.

Dans cette thèse, nous explorons l'utilisation de modèles génératifs profonds pour la détection d'anomalies associées à la démence dans les données de tomographie par émission de positons au ^{18}F -fluorodésoxyglucose (TEP au FDG). Notre méthode repose sur le principe de la reconstruction pseudo-saine, où nous entraînons un modèle génératif à reconstruire des images saines à partir de données pathologiques. Cette approche présente l'avantage de ne pas nécessiter de données annotées, qui sont longues et coûteuses à acquérir, ainsi que d'être généralisable à différents types d'anomalies. Nous avons choisi d'implémenter un auto-encodeur variationnel (VAE), un modèle simple mais qui a fait ses preuves dans le domaine du deep learning. Cependant, analyser la performance de nos modèles génératifs sans disposer de données labellisées ou de cartes d'anomalies mène à une évaluation incomplète.

Pour résoudre ce problème, nous avons mis en place un cadre d'évaluation basé sur la simulation d'hypométabolisme dans les images de TEP au FDG. Ainsi, en créant des paires d'images saines et pathologiques, nous sommes en mesure d'évaluer la capacité du modèle à reconstruire des images pseudo-saines. De plus, cette méthodologie nous a permis de définir de nouvelles métriques pour évaluer la qualité des reconstructions générées par les modèles génératifs. Le cadre d'évaluation a rendu possible une étude comparative sur une vingtaine de variantes du VAE dans le contexte de la reconstruction pseudo-saine de TEP au FDG. Cela nous a permis d'identifier les modèles les plus performants pour la détection des anomalies liées à la démence.

Enfin, plusieurs contributions significatives ont été apportées à des logiciels open-source. Un pipeline de traitement d'images TEP a été intégré au logiciel Clinica. De plus, cette thèse a donné lieu à de nombreux apports au logiciel ClinicaDL, avec notamment l'amélioration de sa structure, l'ajout de nouvelles fonctionnalités, la maintenance du logiciel, ou encore la participation à la gestion du projet.

Remerciements

La rédaction de ce manuscrit, aboutissement de 3 ans et demi de travail, n’aurait pu se faire sans le soutien et l’aide de nombreuses personnes que j’aimerais remercier ici.

Je remercie tout d’abord le jury de ma thèse, notamment Carole Lartizien et Shadi Albaqouni, tous les deux relecteurs, pour avoir lu ma thèse ainsi que pour les échanges que nous avons eus après ma présentation. Je remercie également Jean-François Mangin, le président du jury, dont les questions m’ont permis de faire germer de nouvelles idées pour la poursuite de mes travaux.

J’exprime également ma plus grande gratitude à Ninon Burgos, mon encadrante et directrice de thèse, d’abord pour m’avoir accordé sa confiance il y a quatre ans pour un stage de recherche, puis avec qui nous avons construit ce projet de thèse et une vraie relation de confiance. Son encadrement, d’un point de vue scientifique comme humain, a été essentiel au bon déroulement de cette thèse. J’ai pu grâce à elle apprendre le métier de chercheur et obtenir ce diplôme, mais aussi m’épanouir dans mon quotidien au travail. Je remercie également mon co-directeur de thèse, Olivier Colliot, pour ses conseils scientifiques éclairés et son soutien pendant ces quatre années, mais aussi pour le recul qu’il m’a apporté sur mon projet de thèse et plus généralement sur le monde de la recherche et la science.

Au délai de l’encadrement, j’ai eu la chance de faire cette thèse dans une équipe formidable. À ce titre, j’aimerais commencer par remercier les « anciens », Simona Bottani, Juliana Gonzalez, Alexandre Routier, Paul Vernet, Raphael Couronne, qui nous ont accueilli et intégré dans l’équipe, à un moment où la vie sociale a été bousculée par la pandémie liée à la covid.

Je remercie également les collègues avec qui j’ai pu travailler au quotidien dans l’équipe « détection d’anomalies », Maëlys Solal, et Hugues Roy. Ça a été un vrai plaisir de collaborer avec vous sur ce projet de recherche, et je suis très content de continuer avec vous, on fait une bonne équipe !

Je souhaite aussi remercier l’équipe Clinica et ClinicaDL. Participer à ce projet de développement logiciel m’a permis de compléter mon apprentissage sur le côté informatique du projet. Plus particulièrement, j’aimerais remercier Elina Thibeau—Sutre qui m’a montré que le code est une composante importante d’un projet de recherche en IA, et qui m’a guidé dans la voie de la reproductibilité. J’aimerais également remercier Mauricio Diaz, Omar El Rifai et Ghislain Vaillant, de m’avoir expliqué les subtilités de git et m’avoir donné les bonnes pratiques du développement logiciel en recherche. J’aimerais aussi remercier Camille Brianseau pour tout le travail que nous avons pu faire ensemble et qui a donné un réel second souffle à la vie du projet ClinicaDL.

Je souhaite remercier tous les autres collègues de l’équipe ARAMIS: Tristan, Vito, Charley, Rémi, Pierre-Emmanuel, Juliette, Némó, Sophie, Lisa, Guanghui, Octave, avec qui j’ai partagé de très bons moments, que ce soit pour les échanges scientifiques, les pauses café, et surtout les moments en dehors de labo, comme la semaine au ski, les voyages vélos, les sessions de sport ou les soirées aux bars ! Je remercie particulièrement Matthieu et Arya, mes deux acolytes, avec qui nous avons créé de vraies moments de complicité.

Pour finir, je remercie infiniment ma mère, mon père, et ma petite soeur Sapna. Votre soutien inconditionnel, vos encouragements constants et votre amour indéfectible m’ont permis de devenir la personne que je suis aujourd’hui.

Scientific production

First author journal papers

1. **Ravi Hassanaly**, Camille Brianceau, Maëlys Solal, Olivier Colliot, Ninon Burgos. “Evaluation of pseudo-healthy image reconstruction for anomaly detection with deep generative models: Application to brain FDG PET”. *Journal of Machine Learning for Biomedical Imaging*, 2024, Special Issue for Generative Models, 2, pp.611. [⟨10.59275/j.melba.2024-b87a⟩](https://doi.org/10.59275/j.melba.2024-b87a).

Submitted first author journal papers

1. **Ravi Hassanaly**, Maëlys Solal, Olivier Colliot, Ninon Burgos. “Comparative study of variational autoencoder based methods for unsupervised anomaly detection on brain FDG PET”. Submitted to *Medical Image Analysis*.

Journal papers as co-author

1. Elina Thibeau-Sutre, Mauricio Diaz, **Ravi Hassanaly**, Alexandre M Routier, Didier Dormont, Olivier Colliot, Ninon Burgos. “ClinicaDL: an open-source deep learning software for reproducible neuroimaging processing”. *Computer Methods and Programs in Biomedicine*, 2022, 220, pp.106818. [⟨10.1016/j.cmpb.2022.106818⟩](https://doi.org/10.1016/j.cmpb.2022.106818).
2. Alexandre Routier, Ninon Burgos, Mauricio Diaz, Michael Bacci, Simona Bottani, Omar El-Rifai, Sabrina Fontanella, Pietro Gori, Jérémy Guillon, Alexis Guyot, **Ravi Hassanaly**, Thomas Jacquemont, Pascal Lu, Arnaud Marcoux, Tristan Moreau, Jorge Samper-González, Marc Teichmann, Elina Thibeau-Sutre, Ghislain Vaillant, Junhao Wen, Adam Wild, Marie-Odile Habert, Stanley Durrleman, Olivier Colliot. “Clinica: an open-source software platform for reproducible clinical neuroscience studies”. *Frontiers in Neuroinformatics*, 2021, 15, pp.689675. [⟨10.3389/fninf.2021.689675⟩](https://doi.org/10.3389/fninf.2021.689675).

Book chapters

1. Baptiste Couvy-Duchesne, Simona Bottani, Etienne Camenen, Fang Fang, Mulusew Fikere, Juliana Gonzalez-Astudillo, Joshua Harvey, **Ravi Hassanaly**, Irfahan Kassam, Penelope Lind, Qianwei Liu, Yi Lu, Marta Nabais, Thibault Rolland, Julia Sidorenko, Lachlan Strike, Margie Wright. “Main existing datasets for open data research on humans”. In *Machine Learning for Brain Disorders*, edited by Olivier Colliot, Neuromethods, Springer, 2023. [⟨10.1007/978-1-0716-3195-9_24⟩](https://doi.org/10.1007/978-1-0716-3195-9_24).

Conference papers

1. **Ravi Hassanaly**, Simona Bottani, Benoît Sauty, Olivier Colliot, Ninon Burgos. “Simulation-based evaluation framework for deep learning unsupervised anomaly detection on brain FDG PET”. *SPIE Medical Imaging: Image Processing*, Feb 2023, San Diego, United States. [⟨10.1117/12.2653893⟩](https://doi.org/10.1117/12.2653893) – Runner-up best poster award
2. **Ravi Hassanaly**, Camille Brianceau, Olivier Colliot, Ninon Burgos. “Unsupervised anomaly detection in 3D brain FDG PET: A benchmark of 17 VAE-based approaches”. *Deep Generative Models workshop at the 26th International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI 2023)*, Oct 2023, Vancouver, Canada.
3. **Ravi Hassanaly**, Camille Brianceau, Mauricio Diaz, Sophie Loizillon, Elina Thibeau-Sutre, et al. “Recent advances in the open-source ClinicaDL software for reproducible neuroimaging with deep learning”. *SPIE Medical Imaging: Image Processing*, Feb 2024, San Diego, United States.
4. Maëlys Solal, **Ravi Hassanaly**, Ninon Burgos. “Leveraging healthy population variability in deep learning unsupervised anomaly detection in brain FDG PET”. *SPIE Medical Imaging: Image Processing*, Feb 2024, San Diego (California), United States.

Conference abstracts

1. Elina Thibeau-Sutre, Mauricio Díaz, **Ravi Hassanaly**, Alexandre Routier, Didier Dormont, Olivier Colliot, Ninon Burgos. “ClinicaDL: an open-source deep learning software for reproducible neuroimaging processing”. *MRI Together 2021 - A global workshop on Open Science and Reproducible MR Research*, Dec 2021, Online, France.
2. Omar El-Rifai, Mauricio Diaz Melo, **Ravi Hassanaly**, Matthieu Joulot, Alexandre M Routier, Elina Thibeau-Sutre, Ghislain Vaillant, Stanley Durrleman, Ninon Burgos, Olivier Colliot. “Clinica: an open-source software platform for reproducible clinical neuroscience studies”. *MRI Together 2021 - A global workshop on Open Science and Reproducible MR Research*, Dec 2021, Online, France.
3. Elina Thibeau-Sutre, Mauricio Díaz, **Ravi Hassanaly**, Olivier Colliot, Ninon Burgos. “A glimpse of ClinicaDL, an open-source software for reproducible deep learning in

- neuroimaging”, *MIDL 2022 – Medical Imaging with Deep Learning*, Jun 2022, Zurich, Switzerland.
4. Elina Thibeau-Sutre, Mauricio Díaz, **Ravi Hassanaly**, Olivier Colliot, Ninon Burgos. “ClinicaDL: an open-source deep learning software for reproducible neuroimaging processing”. *OHBM 2022 - Annual meeting of the Organization for Human Brain Mapping*, Jun 2022, Glasgow, United Kingdom.
 5. Omar El Rifai, Mauricio Diaz Melo, **Ravi Hassanaly**, Matthieu Joulot, Alexandre M Routier, Elina Thibeau-Sutre, Ghislain Vaillant, Stanley Durrleman, Ninon Burgos, Olivier Colliot. “Advances in the Clinica software platform for clinical neuroimaging studies”. *OHBM 2022 - Annual meeting of the Organization for Human Brain Mapping*, Jun 2022, Glasgow, United Kingdom.
 6. **Ravi Hassanaly**, Simona Bottani, Benoit Sauty, Olivier Colliot, Ninon Burgos. “Simulation d’anomalies pour l’évaluation de la synthèse d’images pseudo-saines de TEP au FDG cérébrales”. *IABM 2023 - Colloque Français d’Intelligence Artificielle en Imagerie Biomédicale*, Mar 2023, Paris, France.
 7. **Ravi Hassanaly**, Camille Brianceau, Olivier Colliot, Ninon Burgos. “Détection non supervisée d’anomalies cérébrales dans des images 3D de TEP au FDG : un benchmark de 17 modèles basés sur les VAEs”. *IABM 2024 - Colloque Français d’Intelligence Artificielle en Imagerie Biomédicale*, Mar 2024, Grenoble, France.

Talks and Posters

1. Elina Thibeau-Sutre, Mauricio Diaz, **Ravi Hassanaly**, Alexandre Routier, Didier Dormont, Olivier Colliot, Ninon Burgos. “ClinicaDL: an open-source deep learning software for reproducible neuroimaging processing”. Poster presented at the *Prairie Days*, Sep 2021, Paris, France.
2. Elina Thibeau-Sutre, Mauricio Diaz, **Ravi Hassanaly**, Alexandre Routier, Didier Dormont, Olivier Colliot, Ninon Burgos. “ClinicaDL: an open-source deep learning software for reproducible neuroimaging processing”. Talk and Poster at the *3IA doctoral workshop*, Nov 2021, Toulouse, France.
3. **Ravi Hassanaly**, Olivier Colliot, Ninon Burgos. “Pseudo-healthy FDG-PET image synthesis for brain anomaly detection, application on Alzheimer’s disease”. Poster presented at the *CURE-ND workshop for Early Career Researchers*, May 2022, London, United-Kingdom.
4. **Ravi Hassanaly**, Olivier Colliot, Ninon Burgos. “Pseudo-healthy brain FDG-PET synthesis for unsupervised anomaly detection”, Talk at the *3IA doctoral workshop*, Nov 2022, Grenoble, France.

5. **Ravi Hassanaly**, Olivier Colliot, Ninon Burgos. “Pseudo-healthy brain FDG-PET synthesis for unsupervised anomaly detection”. Talk at the *ICM Ajités’ internal workshop*, Nov 2022, France
6. **Ravi Hassanaly**, Simona Bottani, Benoît Sauty, Olivier Colliot, Ninon Burgos. “Simulation-based evaluation framework for deep learning unsupervised anomaly detection on brain FDG PET”. Poster presented at the *CURE-ND workshop for Early Career Researchers*, Mar 2023, Louvain, Belgium.
7. **Ravi Hassanaly**, Camille Brianceau, Olivier Colliot, Ninon Burgos. “Pseudo healthy synthesis using deep learning method for unsupervised anomaly detection on brain 3D FDG PET”. Poster presented at the *ICM Ajités poster contest*, Jun 2023, Paris, France. - Creativity Price.
8. **Ravi Hassanaly**, Olivier Colliot, Ninon Burgos. “Pseudo-healthy brain FDG-PET synthesis for unsupervised anomaly detection”. Talk at the *ICM Ajités’ internal workshop*, Nov 2023, France.

Open-source contributions

1. Camille Brianceau, Nathan Cassereau, Mauricio Diaz, Nicolas Gensollen, **Ravi Hassanaly**, Sophie Loizillon, Alexandre M Routier, Elina Thibeau-Sutre, Ghislain Vaillant, Olivier Colliot, Ninon Burgos. “ClinicaDL”, <https://github.com/aramis-lab/clinicadl>
2. Alexandre Routier, Ninon Burgos, Mauricio Diaz, Michael Bacci, Simona Bottani, Omar El-Rifai, Sabrina Fontanella, Nicolas Gensollen, Pietro Gori, Jérémy Guillon, Alexis Guyot, Matthieu Joulot, **Ravi Hassanaly**, Thomas Jacquemont, Pascal Lu, Arnaud Marcoux, Tristan Moreau, Jorge Samper-González, Marc Teichmann, Elina Thibeau-Sutre, Ghislain Vaillant, Junhao Wen, Adam Wild, Marie-Odile Habert, Stanley Durrleman, Olivier Colliot., “Clinica”, <https://github.com/aramis-lab/clinica>

Contents

Abstract	iii
Résumé	v
Remerciements	vii
Scientific production	ix
Contents	xiii
List of Figures	xix
List of Tables	xxiii
List of Abbreviations	xxv
Introduction	1
Dementia and Alzheimer’s disease	1
Neuroimaging data for neurodegenerative disorders	3
Deep learning for computer-aided diagnosis	6
Contributions	7
Outline of the manuscript	9
1 Anomaly detection in brain FDG PET	11
1.1 Unsupervised anomaly detection in medical imaging	11
1.1.1 Supervised vs unsupervised approaches	11
1.1.2 State-of-the-art on anomaly detection in medical imaging	12
1.2 Materials	14
1.2.1 Positron emission tomography	14
1.2.2 Data preprocessing using Clinica	15
1.2.3 Data selection	17
1.2.4 Data quality control	19
1.2.5 Data preparation using ClinicaDL	20
2 Variational autoencoder for pseudo-healthy reconstruction	23
2.1 Variational autoencoder	23
2.2 Pseudo-healthy reconstruction on a toy dataset	26
2.2.1 Shepp-Logan dataset	26
2.2.2 2D convolutional VAE	27

2.2.3	Results	27
2.2.4	Latent space analysis	29
2.3	Pseudo-healthy reconstruction on FDG PET images	31
2.3.1	Experimental setting	31
	Materials	31
	3D convolutional VAE	31
	Model training	32
	Model evaluation	32
2.3.2	Results	33
2.3.3	Discussion and limitations	35
3	Evaluation and validation of unsupervised anomaly detection methods in neuroimaging	37
3.1	Evaluation of UAD approaches in the literature	37
3.2	Pseudo-healthy image reconstruction evaluation procedure	38
3.2.1	Evaluation metrics for image reconstruction	39
3.2.2	Simulation-based evaluation framework	40
3.2.3	Measuring the healthiness of reconstructed images	42
3.2.4	Anomaly detection and localization	44
3.3	Results	44
3.3.1	Evaluation of the model using the simulation framework	45
	Results on simulated AD-like FDG PET images	45
	Results when simulating various types of dementia	46
	Measuring healthiness of a pseudo-healthy reconstruction	47
	Anomaly detection applied to simulated data	48
3.3.2	Results on AD patients from the ADNI dataset	50
3.4	Comparison between VAE and Unet	50
3.5	Discussion	52
3.6	Conclusion	54
4	Study on the VAE latent space	57
4.1	Latent space visualization	57
4.2	Learning the data distribution	58
4.2.1	Intra- vs inter-subject distance	58
4.2.2	Linear mixed effect models applied to latent representations	60
	Linear mixed effect model	60
	Results	61
4.3	Discussion	61
4.3.1	Conclusion	63
5	Benchmark of VAE-based approaches	65
5.1	Extensions to the variational autoencoder framework	66
5.2	Selection method and evaluation of the models	67
5.3	Materials	68

5.4	Model selection	68
5.4.1	Selection of the encoder-decoder architecture	69
5.4.2	Selection of the models' hyper-parameters	71
5.4.3	Selection of the best trained models	77
5.5	Results obtained for the best models on the test sets	77
5.5.1	Quantitative evaluation of the pseudo-healthy reconstructions from images of control subjects	77
5.5.2	Quantitative evaluation of the pseudo-healthy reconstructions from images with simulated dementia	79
5.5.3	Qualitative evaluation of the pseudo-healthy reconstructions	79
5.5.4	Quantitative evaluation with the healthiness metric	82
5.5.5	Qualitative analysis of the pseudo-healthy reconstructions obtained from real AD patients	84
5.6	Discussion	84
5.6.1	Model selection	85
5.6.2	Model evaluation	86
5.6.3	Limitations and perspectives	89
5.6.4	Reproducibility	90
5.7	Conclusion	91
6	Reproducible neuroimaging processing with deep learning with Clinica and ClinicaDL open-source software packages	93
6.1	Clinica	96
6.1.1	Data structures	96
	Brain Imaging Data Structure (BIDS)	96
	ClinicaA Processed Structure (CAPS)	96
6.1.2	Main functionalities	97
6.1.3	Integration of the <code>pet-linear</code> pipeline in Clinica	97
6.2	ClinicaDL	99
6.2.1	Main functionalities	99
	Preprocessing images	99
	Generation of toy datasets	99
	Preparing metadata	100
	Random search	101
	Training networks	101
	Performance evaluation	102
	Interpretation	102
6.2.2	Model Analysis and Processing Structure (MAPS)	102
6.2.3	Main features of ClinicaDL	103
	Easy use of neuroimaging	103
	Reproducibility of deep learning studies	103
	Avoid common methodological biases in your neuroimaging studies	104
6.2.4	Development practices	105

Distribution and Installation	106
Continuous Integration and Deployment	106
Documentation	107
6.2.5 Recent advances	107
6.2.6 Personal contribution	109
6.2.7 Discussion and future development	110
6.3 Other contributions	111
Conclusion and Perspectives	113
Conclusion	113
Perspectives	114
A PubMed database queries	117
B Anomaly detection in Shepp-Logan phantoms	119
C Examples of reconstructions obtained for healthy subjects and simulated hypometabolic images	123
D Unsupervised anomaly detection in 3D brain FDG PET: A benchmark of 17 VAE-based approaches	125
D.1 Introduction	125
D.2 Methods	126
D.2.1 Variational autoencoder framework for pseudo-healthy image reconstruction	126
D.2.2 Extensions to the variational autoencoder framework	127
D.2.3 Evaluation of the models	128
D.2.4 Materials	129
D.2.5 Experimental setting	129
D.3 Results	130
D.3.1 Pseudo-healthy reconstruction from images of control subjects	130
D.3.2 Pseudo-healthy reconstruction from images simulating dementia	130
D.4 Conclusion	132
E Description of the VAE variants and of their hyper-parameter selection procedure	135
E.1 Adversarial Autoencoder	135
E.2 β -TC VAE	136
E.3 β -VAE	137
E.4 Disentangled β -VAE	137
E.5 Factor VAE	138
E.6 Hamiltonian VAE	139
E.7 Info VAE MMD	140
E.8 IWAE	140
E.9 MS-SSIM VAE	141

E.10 Regularized auto-encoder	142
E.11 Hyperspherical VAE	143
E.12 VAEGAN	143
E.13 VAE with inverse auto-regressive flows	144
E.14 VAE with linear normalizing flows	145
E.15 VAE with VampPrior	146
E.16 Vector-quantized VAE	146
E.17 Wasserstein auto-encoder	147
F Details of the encoder-decoder architecture selection procedure	149
G Benchmark reconstructions	153
H Example of MAPS	157
I Data access	159
ADNI	159
Bibliography	161

List of Figures

1	Different types of dementia and their rate	1
2	T1-weighted MRI of a cognitively normal subject and of a patient with Alzheimer’s disease	4
3	FDG PET of a healthy control subject and of a patient with Alzheimer’s disease	5
4	Number of articles presenting machine learning and deep learning approaches for the computer-aided diagnosis of Alzheimer’s disease published over the years	6
1.1	Example of PET images	15
1.2	Masks of the cerebellum-pons region	17
1.3	FDG PET images in its native space and registered to the MNI space . . .	18
1.4	Illustration of <code>pet-linear</code> quality control	19
2.1	Shepp-Logan phantom generated by ClinicaDL	26
2.2	Reconstruction of Shepp-Logan phantom on two different normal images . .	28
2.3	Reconstruction of Shepp-Logan phantom on two different pathological images	28
2.4	VAE pseudo-healthy reconstruction on images with different kinds of anomalies	30
2.5	Visualization of the latent space after a PCA	31
2.6	Architecture of the 3D convolutional VAE.	32
2.7	Examples of reconstructions obtained from real images of CN subjects . . .	34
2.8	Examples of reconstruction obtained from real images of patients	35
3.1	Hypometabolism simulation pipeline	41
3.2	Evaluation framework using simulated images	42
3.3	Evolution of the MSE with increasing degrees of hypometabolism simulating AD-like anomalies	46
3.4	Example of results obtained from a real image of a CN subject and an image simulating AD hypometabolism	46
3.5	Evolution of the healthiness metric when increasing the percentage of AD-like simulated hypometabolism	48
3.6	Evolution of the healthiness metric for different dementias simulated at 30%	49
3.7	Distribution of the mean FDG PET uptake in different regions of the brain: comparison between the CN subjects from the test set, their AD-like hypometabolic simulation, and their pseudo-healthy reconstruction.	50
3.8	Distribution of the mean FDG PET uptake in different regions of the brain: comparison between the original image from AD patients, their pseudo-healthy reconstruction and the CN population.	50

3.9	Example of results obtained with the Unet from a real image of a CN subject and an image simulating AD hypometabolism	51
3.10	Comparison of the distribution of the healthiness metric between the Unet and VAE for both CN and simulated AD subjects	52
3.11	Schema summarizing proposed evaluation framework	55
4.1	Latent space representation of the different data sets	59
4.2	Intra- vs inter-subject distance box plot	60
4.3	Evolution of the MSE and the SSIM compared with the Minkowski distance in the latent space	61
5.1	Diagram summarizing the benchmark steps	69
5.2	Encoder-decoder modular architecture	70
5.3	Diagram of the selected VAE architecture	71
5.4	Examples of reconstructions obtained with the different VAE variants from the image of a CN subject, and from the same subject with AD simulated at different intensity degrees	81
5.5	Ridgeline plot showing the distribution of the healthiness metric for images from Test AD 30	83
5.6	Evolution of the healthiness metric depending on the severity of the anomalies simulated for the different VAE variants	84
5.7	Distribution of the healthiness metric depending on the dementia simulated at 30% hypometabolism: PCA, bvFTD, lvPPA, svPPA and nfPPA	85
5.8	Joint density plot of the healthiness metric and SSIM	86
5.9	Example of reconstructions obtained from the different VAE variants on an AD patient	87
6.1	Overview of the Clinica and ClinicaDL workflows	95
6.2	List of the pipelines currently available in Clinica with their dependencies and outputs	98
6.3	ClinicaDL main functionalities	100
6.4	Schema of the <code>prepare-data</code> pipeline	104
6.5	Illustration of the scenarios that can lead to data leakage.	105
6.6	New features of the ClinicaDL software platform.	108
6.7	First UML diagram of ClinicaDL	110
B.1	VAE pseudo-healthy reconstruction on images with different missing components	119
B.2	VAE pseudo-healthy reconstruction on images with anomalies on the bottom ROI	120
B.3	VAE pseudo-healthy reconstruction on images with different intensities	121
B.4	VAE pseudo-healthy reconstruction on image with a 15° rotation.	122
C.1	Reconstructions obtained from a real image of CN subjects and the image simulating 30% AD hypometabolism based on the same CN subject	124

D.1	Example of FDG PET image of a CN subject, the corresponding image simulating AD and their pseudo-healthy reconstructions and difference maps	130
D.2	Bar plot of the evolution of the MSE when computed within the mask characteristic of AD between the image simulated with different degrees of hypometabolism and its reconstruction	132
G.1	Examples of reconstructions (coronal slices) obtained with the different VAE variants from the image of a CN subject and from the same subject with AD simulated at different intensity degrees	154
G.2	Examples of reconstructions (sagittal slices) obtained with the different VAE variants from the image of a CN subject and from the same subject with AD simulated at different intensity degrees	155
G.3	Examples of reconstructions obtained with the different VAE variants from the same subject with different dementia subtypes simulated at 30% intensity degree	156

List of Tables

1.1	Summary of participant demographics at baseline for the different training/validation splits and test sets considered	21
2.1	Distribution of the toy data in the train, validation and test sets	27
2.2	Reconstruction performance of our VAE on normal and pathological images	29
2.3	Reconstruction metrics obtained on the test set for images from healthy subjects over the 6 folds.	33
2.4	SSIM obtained on the validation set for all the splits.	33
2.5	Comparison of reconstruction metrics computed on test set with healthy subjects and test set with patients with AD.	34
3.1	Regions associated with different dementia and the masks used for hypometabolism simulation	43
3.2	Reconstruction metrics between the original healthy images from CN subjects and the pseudo-healthy reconstruction	47
3.3	Structural similarity between the pseudo-healthy reconstruction and the reconstruction from the healthy image for the different dementias simulated .	47
3.4	Comparison of the reconstruction results obtained between the Unet and VAE	51
4.1	Result of linear mixed effect models	62
5.1	Hyper-parameters included in our encoder-decoder VAE architecture random search	71
5.2	Summary of the layer parameters in the final VAE architecture	72
5.3	Summary of the hyper-parameters optimized and selected thanks to the random search for each VAE variant. The hyper-parameters are detailed in E. .	74
5.3	Summary of the hyper-parameters optimized and selected thanks to the random search for each VAE variant. The hyper-parameters are detailed in E. .	75
5.4	Reconstruction metrics obtained for the best configuration of each VAE variant on the validation sets	76
5.5	SSIM obtained on each validation set of the 6-fold cross-validation for the different trained models (mean \pm std over the images from the validation set). The best split of each VAE variant is highlighted in bold.	78
5.6	Reconstruction metrics obtained for images from Test CN	79
5.7	Reconstruction metrics obtained between pseudo-healthy reconstructions obtained from the simulated images of Test AD 30 and the ground truth images	80

D.1	Reconstruction metrics computed between the pseudo-healthy reconstructions and the original healthy PET image of CN subjects	131
E.1	Results of the random search on the hyper-parameters of the Adv. AE . . .	136
E.2	Results of the random search on the hyper-parameters of the β -TC VAE . .	137
E.3	Results of the random search on the hyper-parameters of the β -VAE	137
E.4	Results of the random search on the hyper-parameters of the Dis. β -VAE . .	138
E.5	Results of the random search on the hyper-parameters of the FactorVAE . .	139
E.6	Results of the random search on the hyper-parameters of the HVAE	139
E.7	Results of the random search on the hyper-parameters of the InfoVAE . . .	140
E.8	Results of the random search on the hyper-parameters of the IWAE	141
E.9	Results of the random search on the hyper-parameters of the MS-SSIM VAE	141
E.10	Results of the random search on the hyper-parameters of the RAE- ℓ^2	142
E.11	Results of the random search on the hyper-parameters of the RAE-GP . . .	143
E.12	Results of the random search on the hyper-parameters of the SVAE	143
E.13	Results of the random search on the hyper-parameters of the VAEGAN . .	144
E.14	Results of the random search on the hyper-parameters of the VAE-IAF . . .	145
E.15	Results of the random search on the hyper-parameters of the VAE LinNF .	145
E.16	Results of the random search on the hyper-parameters of the VAMP	146
E.17	Results of the random search on the hyper-parameters of the VQVAE . . .	147
E.18	Results of the random search on the hyper-parameters of the WAE	148
F.1	Results of the random search on the VAE architecture: ranking according to the SSIM of the 10 best configurations	151
F.2	Results of the random search on the VAE architecture: ranking according to the MSE of the 10 best configurations	152
H.1	Example of the Model Analysis and Processing Structure (MAPS) obtained when training a classification network on whole images	157

List of Abbreviations

AD	Alzheimer’s Disease
ADNI	Alzheimer’s Disease Neuroimaging Initiative
Adv. AE	Adversarial AutoEncoder
AE	AutoEncoder
β-TC VAE	β -Variational AutoEncoder with Total Correlation
BIDS	Brain Imaging Data Structure
bvFTD	behavioral variant of Fronto-Temporal Dementia
CAPS	ClinicA Processed Structure
CDR	Clinical Dementia Rating
CN	Cognitively Normal
DDPM	Denoising Diffusion Probabilistic Model
Dis. β-VAE	Disentangled β -Variational AutoEncoder
FDA	Food and Drug Administration
FDG [PET]	FluoroDeoxyGlucose [Positron Emission Tomography]
GAN	Generative Adversarial Network
GPU	Graphics Processing Unit
HVAE	Hamiltonian Variational AutoEncoder
IWAE	Importance Weighted AutoEncoder
KL [divergence]	Kullback-Leibler [divergence]
lvPPA	logopenic variant Primary Progressive Aphasia
MAE	Mean Absolute Error
MAPS	Model Analysis and Processing Structure
MMSE	Mini Mental State Examination
MRI	Magnetic Resonance Imaging
MSE	Mean Squared Error
MS-SSIM	Multi-Scale Structural SIMilarity
nfvPPA	non-fluent variant Primary Progressive Aphasia
NIFTI	Neuroimaging Informatics Technology Initiative
PCA	Posterior Cortical Atrophy
PCA	Principal Component Analysis
PET	Positron Emission Tomography
PSNR	Peak Signal-to-Noise Ratio
QC	Quality Control
RAE-GP	Regularized AutoEncoder with Gradient Penalty
RAE-ℓ^2	Regularized AutoEncoder with ℓ^2 penalty
ROI	Region Of Interest
SSIM	Structural SIMilarity
SUVr	Standardized Uptake Value Ratio
SVAE	Hyperspherical Variational AutoEncoder
SVM	Support Vector Machines
svPPA	semantic variant Primary Progressive Aphasia
T1w [MRI]	T1-weighted [Magnetic Resonance Imaging]
UAD	Unsupervised Anomaly Detection
VAE	Variational AutoEncoder

VAEGAN	Variational AutoEncoder with Generative Adversarial Network
VAE-IAF	Variational AutoEncoder with Inverse Auto-Regressive flows
VAE LinNF	Variational AutoEncoder with Linear Normalizing Flows
VAMP	VARIational Mixture of Posteriors
VQVAE	Vector-Quantized Variational AutoEncoder
WAE	Wasserstein AutoEncoder

Introduction

Dementia and Alzheimer's disease

Dementia is a broad term used to describe a variety of symptoms associated with a decline in cognitive functions that hamper daily life. It is defined in the fifth edition of the Diagnostic and Statistical Manual of Mental Disorders (American Psychiatric Association, 2013) as a neurocognitive disorder with:

- a significant cognitive decline in one or more cognitive domain: complex attention, executive function, learning and memory, language, perceptual motor skill, social cognition;
- an interference with everyday activities;
- symptoms not due to other medical disorders such as depression or schizophrenia.

According to the study of Nichols et al., 2022, as dementia is more likely to develop with age, and given the global population's aging trend, the global burden of the disease will increase significantly in the coming decades.

The most common cause of dementia is Alzheimer's disease, with over 60% of the cases. Other causes include vascular dementia, Lewy body dementia, frontotemporal dementia, and more rarely Parkinson's disease, posterior cortical atrophy, variants of primary progressive aphasia, etc. (see Figure 1).

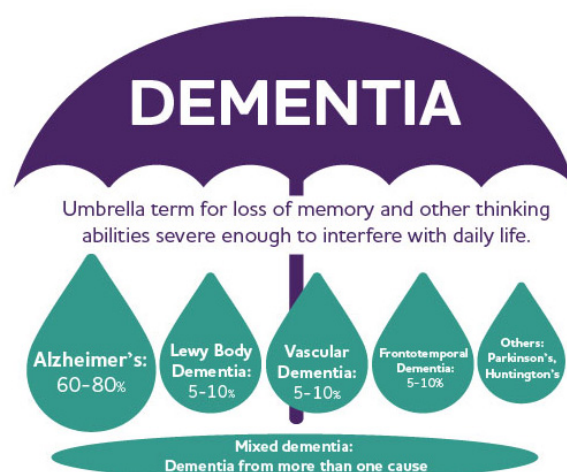


FIGURE 1: Different types of dementia and their rate. *Alzheimer's Association, "What Is Dementia?"*, www.alz.org/alzheimers-dementia/what-is-dementia

Alzheimer's disease is a chronic neurodegenerative disease that affects cognitive function, in particular memory, thinking and behavior. The disease appears with age and touches

mostly the elderly population. The first symptoms of Alzheimer’s disease are short-term memory loss, which gradually worsens over time. As the disease progresses, patients may encounter additional symptoms such as deterioration in language abilities, disorientation, confusion, and changes in mood and behavior (McKhann et al., 1984).

The diagnosis of Alzheimer’s disease relies on clinical assessment, where clinicians employ a set of tests to evaluate the cognitive abilities of the patient (Khan, 2016; Epelbaum et al., 2023). A commonly used cognitive test is the mini mental state examination (MMSE), during which the clinician evaluates various aspects of cognitive function with a quiz of 30 questions (Folstein et al., 1975). These questions cover six categories: orientation in time and space, memory, learning and transcription, attention and calculation, language and object identification, and praxis. Typically, a score above 27 out of 30 indicates normal cognitive function, while scores below this threshold indicate varying degrees of cognitive impairment, with lower scores corresponding to more severe dementia. Another score, the Clinical Dementia Rating (CDR), evaluates six distinct domains: memory, orientation, judgment and problem-solving, engagement in community affairs, management of home and hobbies, and personal care (Hughes et al., 1982). CDR ranges from 0 to 3: a score of 0 means that the patient is cognitively normal, whereas a score superior to 0 indicates cognitive impairment, from low (0.5) to severe (3). However, these scores gauge the severity of the patient’s symptoms, meaning that the diagnosis is made once the disease has already manifested.

Even though the causes of Alzheimer’s disease are not totally understood, the amyloid cascade hypothesis, introduced by Hardy et al., 1992, is a prominent theory in the field of Alzheimer’s disease research. It suggests that the disease develops through the following steps:

- The event triggering the apparition of the disease is the accumulation of β -amyloid peptide in the brain under the form of plaques.
- One of the consequences of this is the excessive phosphorylation of tau proteins in the brain, that will form tangles inside neurons, altering their functioning.
- This will cause the neurodegeneration, and possibly the death of the neurons.
- Ultimately, this will lead to the cognitive decline that is characteristic of Alzheimer’s disease.

While trying to understand the causes of Alzheimer’s disease, this hypothesis also provides several biomarkers that could facilitate early diagnosis and potentially help to predict the evolution of the disease. Jack et al., 2016 proposed the A/T/N classification system to characterize the pathological changes associated with Alzheimer’s disease. It categorizes these changes into three major components: amyloid (A), tau (T), and neurodegeneration (N), which are also the components of the amyloid cascade hypothesis. This classification has later been expanded by Hampel et al., 2021.

There is currently no cure for Alzheimer’s disease. However, very recently, several phase 3 trials of β -amyloid depleting therapies have demonstrated their effectiveness to slow down the progression of cognitive decline (Van Dyck et al., 2023; Sevigny et al., 2016). This has

led to approval of several treatments by the FDA either through the accelerated approval¹ or traditional pathway². In each of these cases, it is essential to detect the disease as early as possible, and if possible before the appearance of the first symptoms, to ensure effectiveness of the medication. Furthermore, given that Alzheimer’s disease progresses gradually over several years, conducting clinical trials for such treatments becomes exceedingly costly. This underscores the importance of meticulous patient selection, particularly targeting those exhibiting early signs of Alzheimer’s disease. In this context, medical imaging plays a key role to observe the physiological changes that appear in the brain several years before the symptoms, such as neurodegeneration, cortical atrophy, β -amyloid aggregation or accumulation of tau protein (Hardy et al., 1992; Jack et al., 2016; Hampel et al., 2021).

Neuroimaging data for neurodegenerative disorders

Medical imaging is a process which consists in creating a 2D or 3D image of the interior of the body. There are multiple acquisition techniques, including x-ray, computed tomography, magnetic resonance imaging or positron emission tomography. These modalities use the different physical properties of the body, allowing the creation of its visual representation (Smith et al., 2010). Medical images give various indications to clinicians, such as changes in shape (enlargement or atrophy of specific structures), shifts in tissues’ intensity, and the emergence of abnormal characteristics like tumors or lesions, all of which may suggest the presence of a disease.

In the context of brain disorders, medical imaging allows the observation of several biomarkers related to dementia. Thus, medical imaging plays a crucial role for the detection, diagnosis and monitoring of neurodegenerative diseases (Burgos, 2023).

One of the most used modalities in clinical applications but also in research is magnetic resonance imaging (MRI), and more specifically structural MRI that provides high resolution visualization of the brain anatomy, while remaining non-invasive. MRI exploits the magnetic properties of hydrogen nuclei present in body tissues to generate contrast. Depending on the electromagnetic radiation emitted and the image post-processing step, it is possible to obtain very different images, also called sequences, with different contrasts between tissues, allowing the observation of diverse structures and features. T1-weighted (T1w), T2-weighted and FLAIR are the most commonly used sequences for brain structural MRI, but many others exist. Moreover, several data acquisition techniques and image reconstruction algorithms can be used to improve the quality of the image (improve resolution, improve contrast, filter artifacts, correct bias field, etc.).

In the case of dementia, brain structural MRI is used for computer-aided diagnosis, since various features can be derived from T1w images such as whole brain volume, density of specific tissues like gray matter, or local cortical thickness and surface area, which are indicative of atrophy, a key marker of neurodegenerative diseases.

¹<https://www.fda.gov/news-events/press-announcements/fda-grants-accelerated-approval-alzheimers-drug>

²<https://www.fda.gov/news-events/press-announcements/fda-converts-novel-alzheimers-disease-treatment-traditional-approval>

For instance, we can observe the structure of a brain thanks to a T1w MRI of a cognitively normal subject in Figure 2. For comparison, in the T1w MRI of the patient with Alzheimer’s disease, we note neuronal loss as evidenced by gray matter atrophy and the increased space occupied by cerebrospinal fluid (appearing dark).

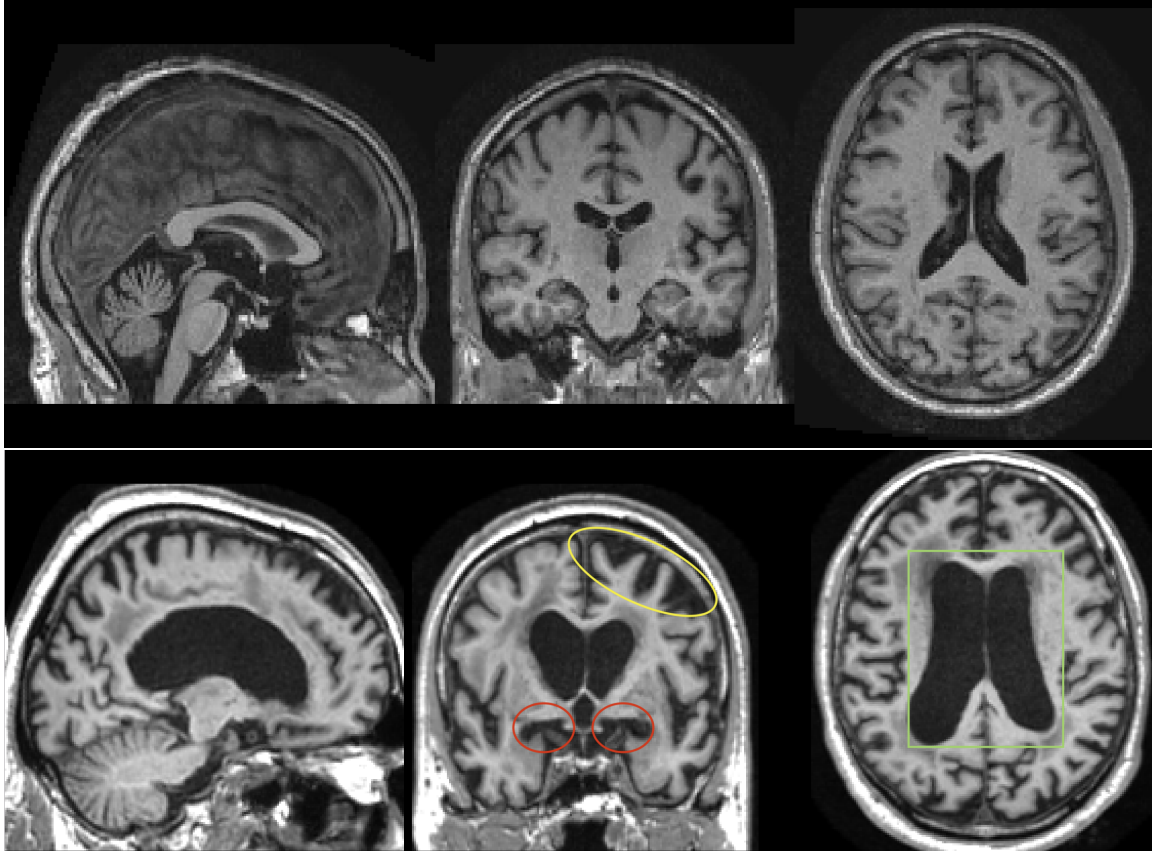


FIGURE 2: T1-weighted MRI of a cognitively normal subject (top) and of a patient with Alzheimer’s disease (bottom). We highlighted significant atrophy in the parietal lobe in yellow, important neuronal loss in the hippocampus in red, and expansion of the ventricles in green.

Another relevant imaging modality for Alzheimer’s disease and dementia is positron emission tomography (PET) as it allows observing the physiological changes that appear in the brain several years before the symptoms, such as neurodegeneration, β -amyloid aggregation or accumulation of tau protein (Herholz, 1995; Herholz et al., 2007; Quigley et al., 2011). PET is a modality using nuclear properties of radioactive substances that are injected intravenously to the patient. It results in a 3D image that highlights the concentration of the radioactive tracer that has been administered. Since it is an invasive method, and the equipment and operation is quite expensive, this modality is more used for research purposes than in clinical routine for the computer-aided diagnosis of dementia.

There are three types of tracers commonly used for the diagnosis of Alzheimer’s disease (Nordberg et al., 2010) that allows observing the three biomarkers associated with Alzheimer’s disease: the concentration of β -amyloid, of the tau protein, and the metabolism of the brain that can indicate neurodegeneration. In this PhD work, we will focus on the use of ^{18}F -fluorodeoxyglucose (FDG), which is a glucose analog that concentrates in areas that consume a lot of it, such as the brain, and will thus highlight its metabolism. In the case of

neurodegenerative diseases, FDG PET is used to localize brain areas with altered glucose metabolism, also called hypometabolism. It is a tracer that allows observing the earliest signs of Alzheimer’s disease, and is commonly used in clinical practice. Indeed, like other fluorine-18 tracers, it has a relatively long half-life (approximately 110 minutes), eliminating the need to synthesize it onsite.

As an example, we display in Figure 3 FDG PET images of a cognitively normal subject and of a patient with Alzheimer’s disease. The tracer is expected to concentrate in the gray matter, which is the brain region containing neurons that have a high metabolic activity and significant glucose consumption. We observe on the scan of the cognitively normal subject that the tracer intensity is relatively homogeneous within the cortical ribbon, demonstrating the healthy functioning of the brain. However, on the scan of the patient with Alzheimer’s disease, we observe some regions with a lower intensity in the gray matter, indicating hypometabolism related to the disease.

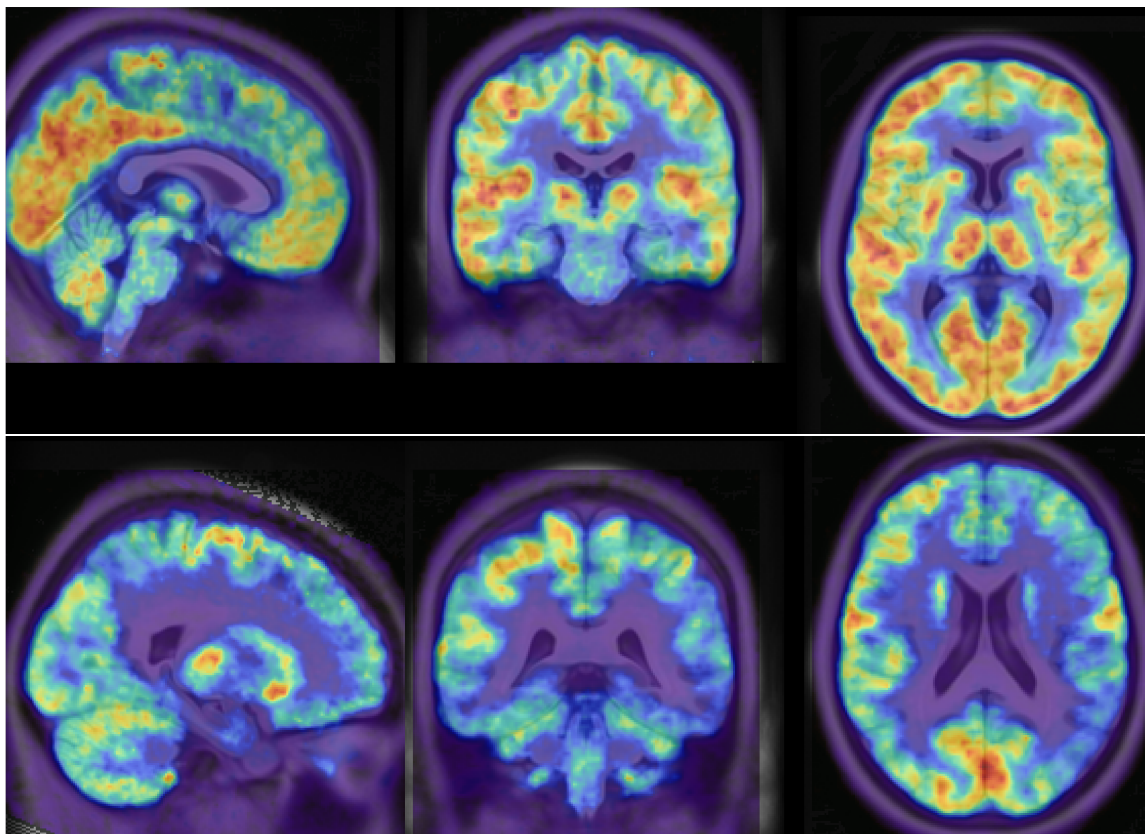


FIGURE 3: FDG PET of a healthy control subject (top) and of a patient with Alzheimer’s disease (bottom)

We saw in the examples above that some biomarkers that are observable through PET images can be used to detect physiological changes in the brain that are related to dementia, such as amyloid accumulation or metabolic dysfunction. However, in clinical practice, PET scans are mostly analyzed visually by a nuclear physician and reading interpretation highly depends on the physician expertise (Perani et al., 2014). Therefore, we would like to automate the analysis of PET images and provide to clinicians a reliable and robust computer-aided diagnosis tool for detecting dementia-related anomalies.

Deep learning for computer-aided diagnosis

During the last decade, breakthroughs in deep learning and computer vision combined with the increasing quality and quantity of medical data available have offered many new possibilities in medical image processing and analysis (Litjens et al., 2017; Esteva et al., 2017; Zhou et al., 2021). Deep learning algorithms are now capable of accomplishing tasks that require a high level of expertise, leading to the development of tools for computer-aided diagnosis (Litjens et al., 2017; Burgos et al., 2021a; Suganyadevi et al., 2022). These technologies are meant to assist healthcare professionals in the diagnostic process by analyzing medical data, such as images or clinical data, and providing additional information that can be used to make a more accurate or early diagnosis.

Neuroimaging does not escape this trend as diagnostic support appears to be helpful for many brain disorders such as dementia, brain tumor and stroke (Venkatraghavan et al., 2023). When reducing the scope to dementia, computer-aided diagnosis consists in using machine learning and deep learning algorithms to analyze neuroimages obtained from modalities like MRI or PET. These systems can automatically detect subtle anomalies, quantify certain features, and highlight areas of concern in order to support clinicians in diagnosing dementia. Computer-aided diagnosis systems do not aim to replace the expertise of healthcare professionals, but rather serve as a supplementary tool to automatize repetitive, long and arduous tasks, and by reducing interpretation errors, especially when there is a need to analyze a large volume of data.

We show in Figure 4 the increasing number of articles published on the computer-aided diagnosis of Alzheimer’s disease since 2005, proving the growing interest of the scientific community and the high potential of these methods. We also notice that in 2023, the number of articles about deep learning approaches surpasses the number of articles about machine learning approaches, showing that deep learning is becoming a new standard for analyzing medical images.

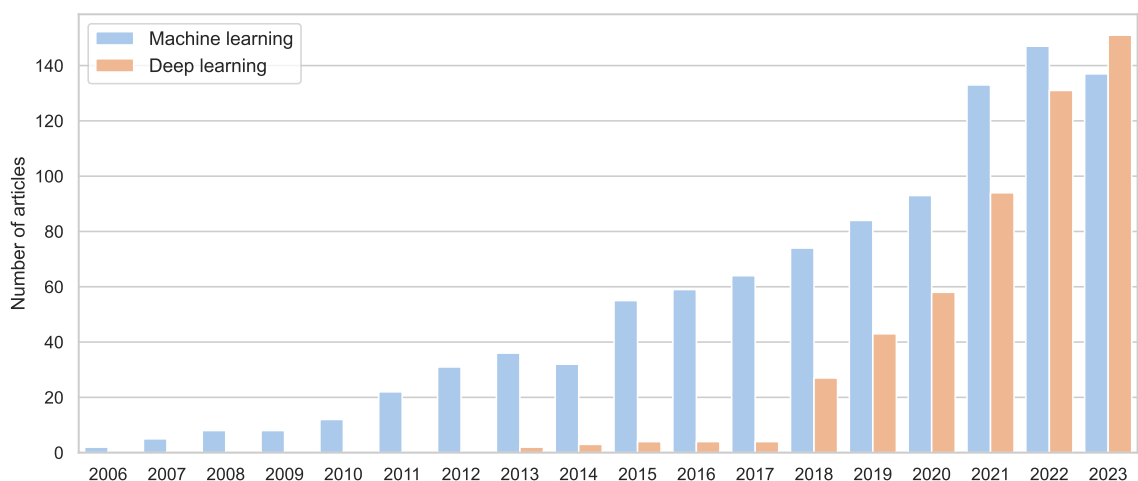


FIGURE 4: Number of articles presenting machine learning and deep learning approaches for the computer-aided diagnosis of Alzheimer’s disease published over the years, according to PubMed (query available in Appendix A).

A first strategy to exploit neuroimaging data involves using them to train a classification

algorithm to distinguish between subjects with dementia from those considered healthy. This approach either leverages patterns and features extracted from neuroimages to develop a machine learning algorithm, or directly uses images to train deep learning classifiers (Samper-González et al., 2018; Pellegrini et al., 2018; Wen et al., 2020; Burgos et al., 2020; Burgos et al., 2021a; Ebrahimighahnavieh et al., 2020).

An alternative approach involves training an algorithm to identify patterns or abnormalities within medical data. Anomaly detection refers to the detection of abnormal patterns or deviations from what can be normally observed. In the context of dementia, it includes anomalies in neuroimaging scans, such as changes in brain structure (visible in structural MRI) or function (visible in functional MRI and FDG PET) that may be indicative of dementia-related pathology. Machine learning, and more recently deep learning, is increasingly being used for anomaly detection in dementia research (Choi et al., 2019; De Carli et al., 2019; Baydargil et al., 2021; Shi et al., 2023; Hinge et al., 2022; Hassanaly et al., 2024b; Hassanaly et al., 2024c; Solal et al., 2024a). These algorithms can learn to detect subtle abnormalities in neuroimaging data that may not be apparent to the human eye, thus potentially enabling earlier and more accurate diagnosis of dementia and related conditions.

To detect subtle anomalies in brain FDG PET, an approach has been proposed by Burgos et al. (Burgos et al., 2015; Burgos et al., 2017; Burgos et al., 2021b). The proposed method consists in generating a pseudo-healthy PET image specific to the subject and using this model to create a subject specific abnormality map. To this end, subjects that are the most similar to the patient under investigation in terms of demographic characteristics and morphology are selected in a control dataset; and by combining it with the patient’s MRI, a model of the PET image is created. The abnormality map is then generated by comparing both PET images (the real one and the synthesized one). To create the PET image model, images selected in the control dataset are first transformed using a registration algorithm to be in the same space as the subject under investigation. Indeed, these images are not originally aligned due to acquisition differences (rotation, translation). Then the model is created using a fusion algorithm. The model is composed of the standard deviation and mean value of the selected subjects.

Although this method for early diagnosis of dementia on FDG PET gave good results, many new deep learning algorithms developed in recent years have the potential to enhance them, and improve anomaly detection in neuroimaging. For instance, deep generative models have shown promising results for anomaly detection in medical imaging (Schlegl et al., 2017; Baur et al., 2021a; Chen et al., 2022; Zhang et al., 2023; Lagogiannis et al., 2023). We would like to use deep generative models for pseudo-healthy reconstruction in order to detect anomalies in brain FDG PET, with the goal of assisting clinicians diagnosing diseases causing dementia.

Contributions

The objective of this PhD was to develop an unsupervised anomaly detection approach based on deep generative models applied to brain FDG PET. More specifically, we focused on using variational autoencoders (VAEs) for reconstructing pseudo-healthy images for the detection

of dementia-related anomalies, without the need for labeled data. During the thesis, three main categories of contributions emerged: methodological advancements in deep learning, the application of these techniques to the field of neuroimaging, and the development of software tools to support the implementation and deployment of these methods in clinical research.

A preliminary work was to develop a pipeline for the preprocessing of PET images. This pipeline had to be in the Brain Imaging Data Structure (BIDS) framework, as the images are stored following this convention (Gorgolewski et al., 2017). The pipeline performs an affine registration to a standard space and the intensity normalization of PET images.

Once the FDG PET data were pre-processed, we trained VAEs to reconstruct pseudo-healthy images with images of cognitively normal subjects. However, if we do not need manually labeled data for this task, we also do not have labeled data for the evaluation of our trained models. One solution would be to refer to a clinician. However, in order to provide a tool for robust and automatic assessment of the performance of the models, we built a framework for the evaluation of pseudo-healthy reconstruction approaches in the absence of ground truth. This framework consists in simulating anomalies in images of healthy subjects to generate pairs of pathology-free and pathological (e.g., mimicking dementia-like lesions) images. We complemented the framework by defining new healthiness and anomaly metrics. The healthiness metric measures whether the reconstructed image is of healthy appearance to evaluate the model capacity to reconstruct pseudo-healthy images, whereas the anomaly metric measures whether the input image contains anomalies using both the pseudo-healthy reconstruction and the input image. This resulted in a multitude of experiments to extensively evaluate a 3D VAE trained on full resolution PET using the framework, including an analysis of the VAE latent space. A preliminary version of this work has been published as a conference proceeding (Hassanally et al., 2023a), before being extended to a journal version published in the Special Issue for Generative Models of Machine Learning for Biomedical Imaging (Hassanally et al., 2024b).

We then proposed a benchmark of 20 VAE-based models focused on the pseudo-healthy reconstruction of 3D FDG PET images in the context of dementia. We compared many VAE-based models that have not been applied to medical image analysis yet: in contrast to computer vision works, where datasets typically contain several tens of thousands of images, it has been interesting to examine the performance of such models when trained on a relatively small dataset, comprising only a few hundred images, which is typical in medical imaging. The models were evaluated and compared thanks to the evaluation framework. A preliminary version of this work has been published as a conference proceeding (Hassanally et al., 2023b), before being extended to a journal version that has been submitted to Medical Image Analysis (Hassanally et al., 2024c).

Finally, a significant contribution of this thesis is the participation to the development of the open-source software packages Clinica and ClinicaDL. Clinica (Routier et al., 2021) is an open-source software for reproducible processing of neuroimaging datasets and multi-modal neuroscience studies. I added to Clinica the PET images processing pipeline, named `pet-linear`; I also updated the BIDS converter for the ADNI database. ClinicaDL

(Thibeau-Sutre et al., 2022b) is an open-source software that aims at enhancing the reproducibility and rigor of research on deep learning in neuroimaging. My contributions to ClinicaDL are numerous: from a refactoring of the whole software engine and structure, to the micromanagement of the project, passing by the addition of new features and the maintenance of the software. This work has been used extensively by other researchers of the laboratory and led to several publications in journals (Routier et al., 2021; Thibeau-Sutre et al., 2022b) and conferences (Thibeau-Sutre et al., 2022a; Hassanaly et al., 2024a).

Outline of the manuscript

The manuscript is organized as follows:

- In **Chapter 1**, after presenting the state-of-the-art of unsupervised anomaly detection applied to medical imaging and neuroimaging, we focus on our application by describing the imaging modality, the dataset and more generally the materials we use in our work.
- In **Chapter 2** we demonstrate that the VAE is well suited for pseudo-healthy reconstruction through a theoretical analysis, and implement this generative model first for a toy dataset and then for real brain FDG PET data, before raising some limitations due to the lack of evaluation materials and tools.
- In **Chapter 3** we therefore introduce an evaluation method based on the simulation of anomalies related to dementia on 3D FDG PET in order to measure the performance of generative deep learning models for unsupervised anomaly detection when no ground truth data is available.
- In **Chapter 4** we use both the VAE regularized latent space and the introduced evaluation method to push the evaluation further and try to explain and interpret the results of our model.
- In **Chapter 5** we compare about 20 different VAE variants in the context of UAD applied to dementia, and provide a method to select the best architectures and parameters when benchmarking models.
- In **Chapter 6** we present the numerous software contributions that have been made during this thesis, especially to the Clinica (Routier et al., 2021) and ClinicaDL (Thibeau-Sutre et al., 2022b) open source software packages.
- Finally, in the **Conclusion and Perspectives** chapter, we sum up our contributions, discuss the results and outline potential future research directions.

Chapter 1

Anomaly detection in brain FDG PET

1.1 Unsupervised anomaly detection in medical imaging

The synergy between innovations in imaging technologies, the growing volume of medical data, and sophisticated machine learning algorithms have given rise to algorithms capable of performing complex tasks such as anomaly detection for computer aided diagnosis (Fernando et al., 2021).

1.1.1 Supervised vs unsupervised approaches

A strategy for anomaly detection with deep learning consists in using a supervised algorithm that learns from annotated data. This has the advantage of having remarkable performance on the specific task learned, which can be classification between normal and abnormal images (Esteva et al., 2017; Wen et al., 2020) or anomaly segmentation (Zhou et al., 2018; Isensee et al., 2021). However, this strategy has several drawbacks: the first one is that it requires a large amount of annotated data that are time-consuming and costly to acquire. The second one is that the model's results will be affected by potential annotation errors. The last disadvantage is that the models will be specific to the data, diseases and anomalies they have been trained on. This might be an issue especially for rare diseases for which few data samples are available.

Another strategy, called unsupervised anomaly detection (UAD), consists in using self-supervised, weakly supervised or unsupervised learning for anomaly detection (Chen et al., 2022; Zhang et al., 2023). The underlying idea of these methods is to learn the distribution of healthy data. One can then use it to detect out-of-distribution samples, and thus identify abnormal cases. Another way is to use generative models to reconstruct pseudo-healthy images from the healthy data distribution: since the model is trained to reconstruct only normal data, we assume that the reconstruction of abnormal images will be imperfect, and by comparing the input real image to the reconstruction, we should be able to detect anomalies. The first advantage of this strategy is that it does not require voxel-level annotation. Another benefit is that it should be able to detect any type of anomaly, potentially linked to different diseases. Deep generative models such as variational autoencoders (VAEs) (Kingma et al., 2014), generative adversarial networks (GANs) (Goodfellow et al., 2014) and more recently denoising diffusion probabilistic models (DDPMs) (Ho et al., 2020)

have shown great results for image generation tasks and unsupervised anomaly detection in medical imaging (Esmaceli et al., 2023), including neuroimaging (Wang et al., 2023; Gong et al., 2023).

1.1.2 State-of-the-art on anomaly detection in medical imaging

Unsupervised methods having many benefits, their use for anomaly and outlier detection in medical imaging has increased over the last years, especially methods based on image synthesis.

A family of methods do not rely on the image reconstruction, but rather consist of out-of-distribution detection algorithms. For instance, Alaverdyan et al., 2020 used a siamese autoencoder on 2D patches to detect epilepsy lesions on T1w and FLAIR MRI. The model learns a latent representation of each voxel of MRI from healthy controls, using the patch surrounding the voxel. The siamese autoencoder consist of two autoencoders sharing the same weights and latent space, that are regularized by enforcing two similar patches x_1 and x_2 to have close latent space representations z_1 and z_2 . Then, a one-class support vector machine (SVM) is trained to classify abnormal voxels' latent representations as outliers in the latent space, allowing to detect a neighborhood of abnormal voxels that correspond to brain lesions. This method has further been applied to the detection of white matter hyper-intensities (Pinon et al., 2023b), by training the one-class SVM at the patient level instead of training it on the training set. It has further been applied to Parkinson's disease and improved by using a generative mixture model to have a proximity measure in order to detect out-of-distribution latent vectors (Pinon et al., 2023a).

Concerning pseudo-healthy reconstruction, the underlying idea is to create new healthy looking images from existing data using a generative model such a VAE (Kingma et al., 2014), GAN (Goodfellow et al., 2014) or DDPM (Ho et al., 2020). By training only with images from healthy subjects, the model learns the distribution of normal or healthy data. We then expect that the reconstruction of an image with this model will look like a healthy version of the original image, whether the image is that of a healthy subject or a patient with a disease, thus the name pseudo-healthy reconstruction. The reconstructed pseudo-healthy image is finally compared to the real one to detect anomalies and possibly compute an anomaly score.

Pseudo-healthy reconstruction has been used in numerous fields of medical imaging (Fernando et al., 2022), for instance to detect various lung anomalies such a pneumonia on chest x-ray (Nakao et al., 2021; Kim et al., 2023), retinal anomalies on optical coherence tomography (Schlegl et al., 2017; Schlegl et al., 2019; Zhou et al., 2023), breast cancer on mammogram (Park et al., 2023), skin cancer on dermatoscopic images (Lu et al., 2018), tumors detection on PET, computed tomography and PET-CT (Astaraki et al., 2023), or malignant tissues on colonoscopy (Tian et al., 2021). In the following, we will reduce the scope to methods applied to neuroimaging.

Several methods based on autoencoders have been developed, starting by Zimmerer et al., 2018 who used a context-encoding VAE for the detection of brain glioma and multiple sclerosis lesions on anatomical MRI. The localization of the anomalies was improved in a subsequent work using a pixel-wise KL distance (Zimmerer et al., 2019). Another work by

Chen et al., 2018b introduced constrained adversarial autoencoders. Marimont et al., 2021 used both prior-based anomaly score and reconstruction-based anomaly score with a vector quantized VAE (VQVAE). Pinaya et al., 2022b also used a VQVAE together with an autoregressive transformer in the latent space to better learn the probability density function of healthy data. To show the efficiency of autoencoders, Baur et al., 2021b implemented an autoencoder with a spatial latent space and skip-connections and compared the result to a UNet trained for supervised anomaly segmentation. Bercea et al., 2023c tried to generalize UAD to non hyper-intense anomalies in order to detect various pathological features using a reverse autoencoder. Lüth et al., 2023 reused the general principle of pseudo-healthy reconstruction with autoencoders but in their case, the encoder is improved thanks to contrastive learning in order to use high level features of the image to learn a better latent representation.

Based on the fundamental work of Schlegl et al., 2017 who introduced AnoGAN and its improved version, the f-AnoGAN (Schlegl et al., 2019), several frameworks using GANs have been developed. For instance, the VAEGAN (Baur et al., 2019), the ANT-GAN (Sun et al., 2020), or the cycleGAN (Xia et al., 2019; Xia et al., 2020). More recently, Shi et al., 2023 introduced GANCMLE, a GAN-based approach combined with an autoencoder and constrained by multiple losses for the early detection of brain atrophy. Another novel and interesting approach has been proposed by Siddiquee et al., 2023, who train a GAN-based model with both healthy and abnormal images to have a fully unsupervised method. Finally, Bercea et al., 2023d combined both a latent generative model and high quality reconstruction networks based on in-painting GAN to detect stroke lesions on T1w MRI.

Baur et al., 2021a summed up and compared many of the VAE and GAN approaches that had been used for unsupervised brain tumor and multiple sclerosis lesion segmentation on MRI data.

More recently, following the success of diffusion models for image generation, DDPMs have also been used for anomaly detection tasks in medical imaging. Wyatt et al., 2022 introduced AnoDDPM, a DDPM based anomaly detection method on T1w MRI with partial diffusion strategy, in order to reduce the computational cost for both training and inference. They also explore the use of Simplex noise, claiming that Gaussian noise doesn't allow to detect anomalies of different scale. This approach has been ranked among the best in a recent review (Bercea et al., 2023a). Wolleb et al., 2022 used denoising diffusion implicit models combined with classifier guidance applied to lung X-ray and brain MRI. Pinaya et al., 2022a combined a VQVAE with diffusion model in the latent space to identify abnormal areas in the latent space, in order to further localize anomalies in the image. He showed that DDPM methods outperform his previous work, in which he used transformers in the latent space (Pinaya et al., 2022b). This work has subsequently been improved by Graham et al., 2023 who introduced the latent diffusion model. It consists of a diffusion model that is trained on the 3D latent space of a VQVAE, in order to improve the reconstruction quality. Finally, (Bercea et al., 2023b) uses an iterative process combined with in-painting approach to refine the anomaly mask obtained with DDPM.

Most methods for UAD have been applied to brain structural MRI, often targeting sharp and visible anomalies such as tumors or multiple sclerosis lesions. Only a few studies

have focused on other modalities such as computed tomography or PET, probably because fewer data is available. Choi et al., 2019 implemented a simple VAE for anomaly detection on 2D slices extracted from brain FDG PET. Baydargil et al., 2021 used a GAN with an autoencoder architecture for the generator (with a parallel model for the encoder) to detect anomalies on FDG PET in the context of Alzheimer’s disease. Another interesting approach is the use of multi-modal VAE in order to combine features of different modalities Kumar et al., 2023; Lawry Aguila et al., 2023.

In most cases, the proposed methods work with 2D images that are extracted from 3D volumes. But recently, numerous articles working directly with 3D images or trying to reconstruct 3D volumes have been published. Pinaya et al., 2022b validated their model on both 2D and 3D images. Chatterjee et al., 2022 proposed a compact version of the context encoding VAE of Zimmerer et al., 2018 that is trained on 2D slices that are stacked to obtain a 3D volume. Han et al., 2021 presented a similar strategy, which consists of using three successive slices to reconstruct the following three slices to take into account the 3D structure of the image. Luo et al., 2023 directly trained a 3D encoder to detect brain abnormalities on T2-weighted volumes. Bengs et al., 2021 compared 3D and 2D VAEs for anomaly detection on brain MRI. Bengs et al., 2022 trained a VAE on 3D T1-weighted MRI by additionally considering the age information. Simarro Viana et al., 2020 proposed a 3D extension of the 2D f-AnoGAN and refined the training steps to detect traumatic brain injuries. Finally, DDPMs do not scale well to 3D images, as they require much more memory (Wyatt et al., 2022; Graham et al., 2023).

In this work, we aim to apply UAD methods to identify metabolic changes associated with Alzheimer’s disease and other dementias (Chételat et al., 2020) that are visible in brain ^{18}F -fluorodeoxyglucose (FDG) positron emission tomography (PET) images. ^{18}F -FDG PET images are 3D images that highlight the concentration of administered FDG, a tracer used to localize hypometabolism in the case of neurodegeneration (Herholz, 1995). This application is particularly interesting as deep learning methods for UAD have rarely been applied for the diagnosis of dementia (Choi et al., 2019; Baydargil et al., 2021; Hinge et al., 2022), whereas this approach could enable early diagnosis since changes visible in neuroimaging can occur years before the onset of initial symptoms (Jack et al., 2016). The metabolic abnormalities can be difficult to detect as they are diffuse and sometimes subtle (limited difference in intensity between normal tissues and areas with hypometabolism) (Burgos et al., 2021b), contrary to glioblastoma or white matter hyper-intensities usually studied on structural MRI (Baur et al., 2021a; Xia et al., 2020; Chen et al., 2018b; Zimmerer et al., 2019).

1.2 Materials

1.2.1 Positron emission tomography

Positron emission tomography is a modality using nuclear properties of radioactive materials that are injected in the patient intravenously. This method is invasive, but the quantity of radioactive isotope is small enough to be harmless for the patient. As soon as the radioactive isotope disintegrates, it emits a positron. When the emitted positron encounters an electron,

they will combine to form a positronium. This positronium will quickly annihilate into two photons (corresponding to gamma rays) that will propagate in opposite directions. The sensor around the patient will catch these photons, and when it detects two photons in a really short interval, it is possible to deduce the origin of the emission (on the line between the two points) (Sharp et al., 2005).

Positron emission tomography is used to visualize a biological process of an organ by attaching a radioactive isotope to a specific molecule we want to trace, forming a radiotracer. Finding the emission location allows finding the concentration of the molecule we are tracing, and if this molecule (for example glucose) intervenes in the functioning of an organ, we can deduce in which part of it the operation is done. For example, this can be used to find which parts of the brain use glucose and which do not (which implies disorder).

There are three types of tracers commonly used for the diagnosis of Alzheimer’s disease (Nordberg et al., 2010). The first category is that of amyloid tracers that allow visualizing the aggregation of β -amyloid in the brain. Several molecules exist such as Pittsburgh compound B, ^{18}F -florbetapir, ^{18}F -flutemetamol, or ^{18}F -florbetaben (Landau et al., 2014). The second category corresponds to tracers used for tau protein imaging, such as ^{18}F -florbetapir, which are useful for any tau pathology (Leuzy et al., 2019). Examples are given in Figure 1.1.

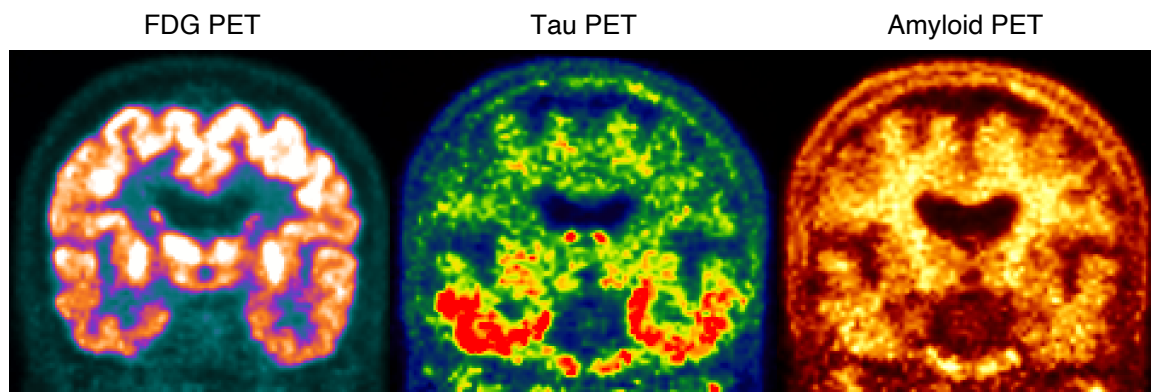


FIGURE 1.1: Example of PET images. Left: ^{18}F -FDG PET displaying brain glucose metabolism. Middle: ^{18}F -florbetapir PET displaying the presence of tau neurofibrillary tangles. Right: ^{18}F -florbetapir PET displaying the presence of amyloid plaques. All the images correspond to the same Alzheimer’s disease patient from the ADNI database.

In this PhD work, we will focus on the use of the last category of tracers: FDG.

1.2.2 Data preprocessing using Clinica

During this PhD project, we used FDG PET scans that have been acquired using different protocols, scanners and imaging centers. Therefore, a preprocessing pipeline is essential for preparing PET images for deep learning training, aiming to reduce biases due to different image provenance and acquisition. The `pet-linear` pipeline¹ have been developed for this purpose, and added it to the Clinica open-source software (Routier et al., 2021).

The `pet-linear` pipeline performs a spatial registration to the MNI ICBM 2009c Non-linear Symmetric template (Fonov et al., 2009; Fonov et al., 2011), followed by an intensity

¹https://aramislab.paris.inria.fr/clinica/docs/public/dev/Pipelines/PET_Linear/

normalization of PET images. A prerequisite of the pipeline is to register the MRI scan associated to the PET scan in the MNI space using `t1-linear` pipeline from Clinica². Then, several steps of processing are applied to the PET images.

Step 1: Register the PET image to the associated T1w image The first step of the pipeline is a rigid transformation of the PET scan to the associated T1w MRI in its native space. Only the transformation from the PET native space to the T1w MRI space is saved (and not the registered PET image). The reason is that it is easier to find a rigid transformation in the patient anatomical space from the PET native space to the T1w MRI space than directly computing a linear transformation from the PET space to the MNI space.

Step 2: Compose transformation to register PET image to MNI The `t1-linear` pipeline that have to be run before using the `pet-linear` pipeline to save the affine registration between the T1w MRI and the MNI template (Fonov et al., 2009; Fonov et al., 2011). We then compose this transformation and the one from PET to MRI computed during the first step to get the transformation from the PET native space to the MNI space. This resulting transformation is applied to the PET image using the SyN algorithm (Avants et al., 2008) from the ANTs software package (Avants et al., 2014), and the registered PET image is saved.

Step 3: Perform intensity normalization using a reference region defined in the MNI space Then, the registered PET image intensity is normalized using the mean intensity in reference regions, resulting in a standardized uptake value ratio (SUVR) map (Nugent et al., 2020). This is necessary because the image intensity depends on the quantity of tracer injected to the subject during the acquisition but also the physiology and the morphology of the subject. The pipeline uses a binary mask of the reference region to compute the mean SUVR, then the whole image intensity is divided by this value for normalization. The reference region used to compute the SUVR can be chosen by the users depending on the tracer used for the PET and the disease studied: the region is selected depending on where the tracer concentration is expected to be unaffected by the disease under study and remain relatively constant.

In our case, as we work with FDG PET on Alzheimer’s disease, we created two masks for pons and cerebellum-pons regions. We used the Pick Atlas³ because it is the only open-source atlas with those regions defined in the MNI space. However, as we can see in Figure 1.2, the cerebellum-pons region defined in the atlas (in blue) is overflowing on other tissues, so we had to refine the masks. To make it more specific, we removed voxels overlapping with tissues that cannot be in the cerebellum or the pons. To this end, we used SPM software⁴ that provides tissue probability maps in the MNI space. We binarized these probability maps to make tissue masks, and merged the binary masks of the cerebrospinal fluid, the skull, the “others” (skin...) and the background to have a mask of all tissues

²https://aramislab.paris.inria.fr/clinica/docs/public/dev/Pipelines/T1_Linear/

³https://www.nitrc.org/projects/wfu_pickatlas/

⁴<https://www.fil.ion.ucl.ac.uk/spm>

outside the brain. We then removed voxels of the regions defined in the Pick Atlas that were overlapping with the mask we obtained (in green). Finally, we eroded the cerebellum and pons regions to be sure that it will not overflow on any images registered in the MNI space (in red).

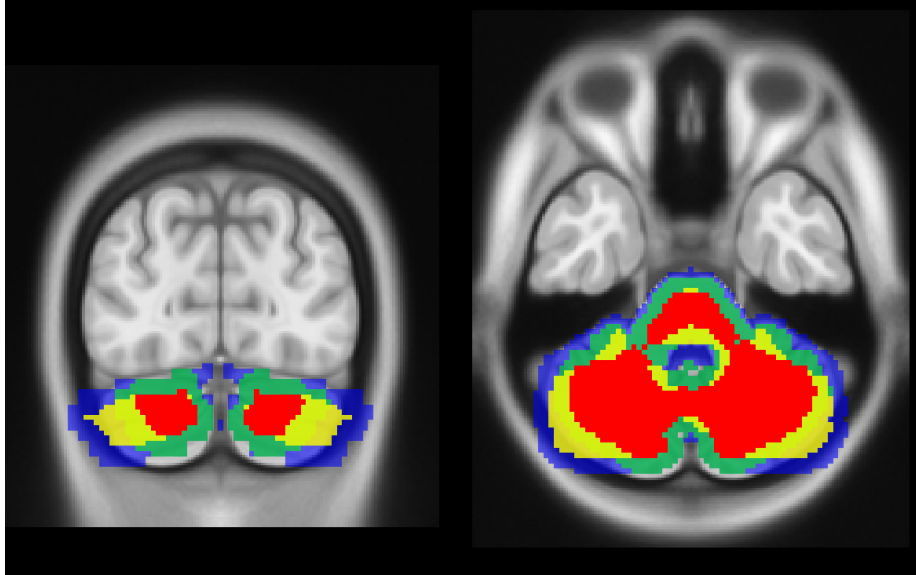


FIGURE 1.2: Different masks of the cerebellum-pons region. We can see in blue the region as defined in the Pick atlas that is clearly overflowing. In yellow there is the old mask used in Clinica, which is the original mask cropped. However, this mask still overflows. In green, we can see the mask after removing the overlapping parts with SPM extra brain regions. And in red the final mask we obtained after eroding the green one.

As intensity normalization is a very important step in the pipeline, even if an accurate mask of the reference regions was defined, the first and last deciles from the voxel intensity distribution are removed to compute the SUVR to be sure to filter outliers from voxels from neighboring regions. What is more, an additional non-linear registration of the PET native space to the MNI space is computed, only to refine the calculation of the SUVR. Indeed, linear registration is simple and does not assure that cerebellum-pons region is perfectly registered to the MNI space. So this extra step is performed to have the best estimation possible of the SUVR. However, this non-linear transformation is just a side step, it is not saved or used for other purposes.

Step 4: Cropping A last optional step can be performed: the cropping of the image to center the image and remove the extra background.

An illustration of the resulting registered, normalized and cropped FDG PET image is displayed in Figure 1.3. The pipeline can be applied to other tracers than FDG.

1.2.3 Data selection

FDG PET scans used in this study were obtained from the ADNI database (Mueller et al., 2005; Jagust et al., 2010; Jagust et al., 2015). The ADNI was launched in 2003 as a public-private partnership, led by Principal Investigator Michael W. Weiner, MD. The primary goal of ADNI has been to test whether serial MRI, PET, other biological markers, and

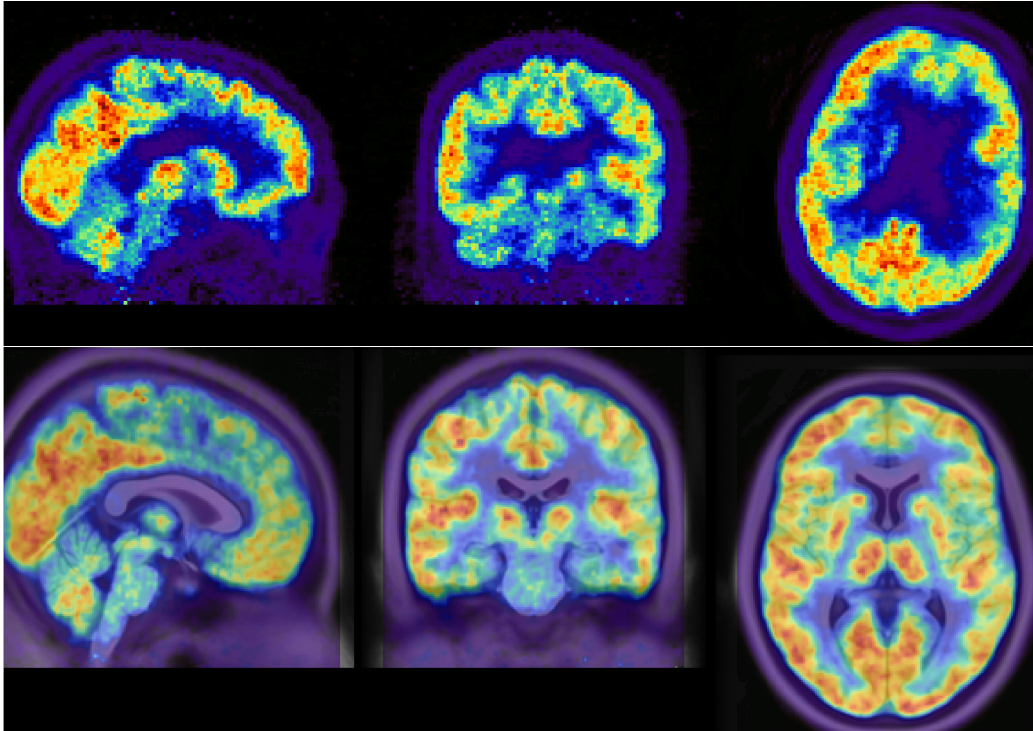


FIGURE 1.3: FDG PET images in its native space (top), and the same image registered to the MNI space and normalized in intensity (bottom, overlaid on the MNI template)

clinical and neuropsychological assessment can be combined to measure the progression of mild cognitive impairment and early AD.

We selected FDG PET images co-registered, averaged and uniformized to a resolution of 8 mm full width at half maximum to reduce the variability due to the use of different scanners. We only used PET images for which a T1-weighted (T1w) MR image was available at the same session for preprocessing purposes. The images were then processed using Clinica’s (Routier et al., 2021) `pet-linear` pipeline: they were registered using a rigid transformation to the corresponding T1w MRI of the same session, and then affinely registered to the MNI ICBM 2009c Nonlinear Symmetric template (Fonov et al., 2009; Fonov et al., 2011) using the transformation computed with the `t1-linear` pipeline. They were then normalized in intensity using the average PET uptake in a region comprising cerebellum and pons, and cropped. In the end, the dimension of the PET scan is $169 \times 208 \times 179$ with 1 mm isotropic voxels.

In the ADNI database, there is a total of 3511 FDG PET scans from 1600 participants. This includes 554 cognitively normal (CN) subjects (1010 images) that we selected since UAD models are trained only on images from healthy subjects. We know that physiological changes can appear several years before the first clinical symptoms, so to ensure that images really correspond to a healthy brain, we kept only scans from subjects that are CN for at least three years after the session considered. We discarded 78 subjects (129 images) for whom diagnosis progresses to AD, 72 subjects (72 images) for whom there is a unique session (which is not enough to assess the reliability of the CN label) and 21 subject (49 images) for whom there are multiple conversions or regressions. We finally keep 383 stable CN subjects (760 images).

There are also 560 AD patients (791 images). We removed 2 patients (2 images) with unstable AD diagnosis, 3 patients (3 images) of regressive AD, 189 patients (189 images) for whom there is a unique session, and 4 subjects that were already in the training set. In the end, we keep the 362 baseline sessions of the remaining AD patients for testing purposes and discarded all the other images.

1.2.4 Data quality control

To filter out potential PET images not correctly registered to the MNI template, we performed quality control. We first controlled the quality of the registration between the T1w MRI and the MNI template, as it is an intermediate step when registering the PET image to the MNI space. The approach relies on a pre-trained neural network called DARQ (Fonov et al., 2022) that learned to classify images that are adequately registered to the MNI template.

We then assessed the quality of the alignment of the PET image itself with the MNI template. Here the approach relies on a metric that measures the overlap between the output of the `pet-linear` pipeline, i.e. the PET image supposedly aligned with the MNI template, and a mask corresponding to the outside of the brain obtained from the MNI template. If the overlap is large, we assume that the PET image is not well-registered, as illustrated in Figure 1.4.

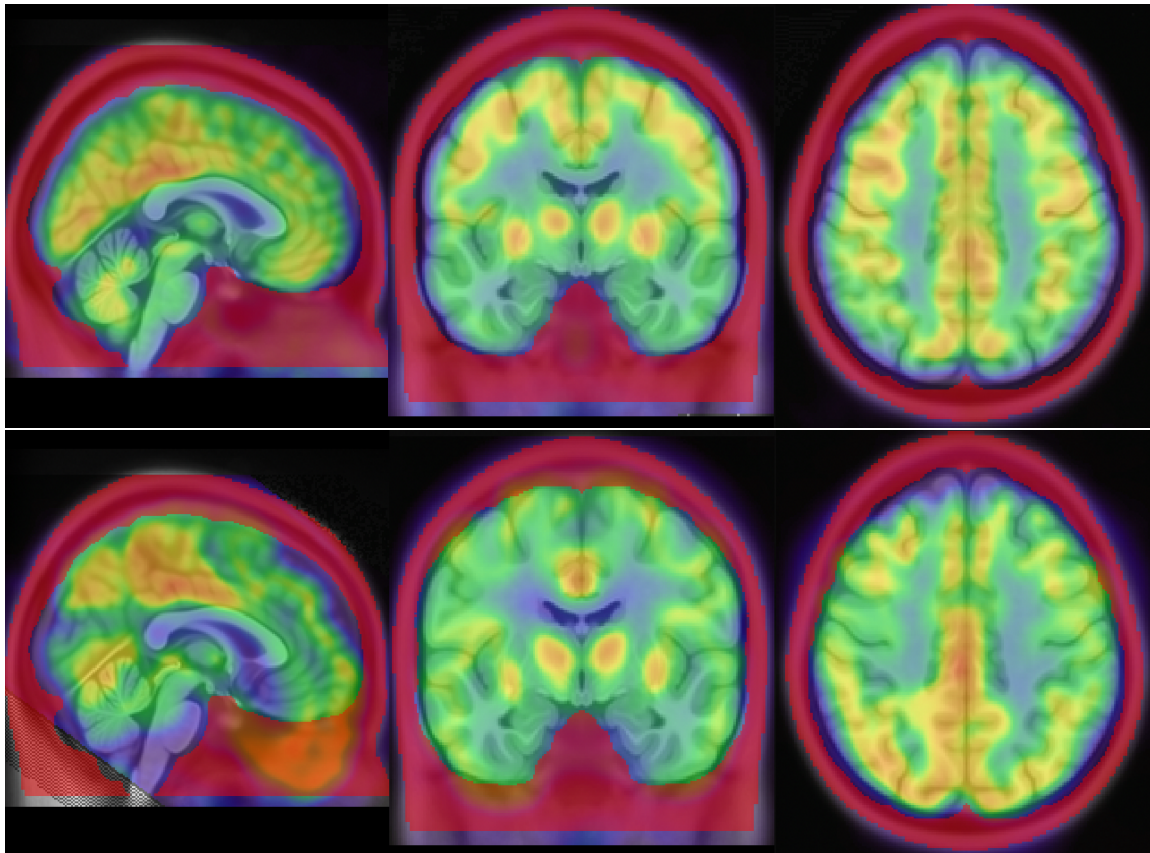


FIGURE 1.4: Example of an FDG that is well registered and passed quality control (top), and an FDG PET that is badly registered and did not pass quality control (bottom).

Both quality control pipelines are available in the ClinicaDL open-source software⁵ (Thibeau-Sutre et al., 2022b).

After running both quality control pipelines, we discarded a total of 30 images: 18 images from CN subjects and 9 images from AD patients after `t1-linear` quality control, and 3 images from CN subjects after `pet-linear` quality control.

Finally, our dataset is composed of 739 images from 378 CN subjects and 353 images at baseline from 353 AD patients.

1.2.5 Data preparation using ClinicaDL

For our deep learning experiments, we split our dataset of 378 CN subjects into training and test sets at the subject’s level to avoid any form of data leakage (Wen et al., 2020), stratifying by sex and age to reduce biases. Only baseline sessions were kept in the test set to avoid biased results. The test set, comprising 60 CN subjects (60 images), is used to assess whether the healthy images are reconstructed as healthy. We denote it as “Test CN”. We then performed a six-fold cross validation on the training data to estimate the variance due to data splitting Bouthillier et al., 2021. 53 subjects (53 images) belong to the validation sets to monitor the training and 265 subjects (between 510 and 538 images depending on the fold) are used to train our models. The details of the dataset splitting and folds statistics are summarized in Table 1.1.

⁵<https://clinica dl.readthedocs.io/en/latest/Preprocessing/QualityCheck/>

TABLE 1.1: Summary of participant demographics at baseline for the different training/validation splits and test sets considered. Note that split s corresponds to using fold s from the 6-fold cross-validation as validation set and the other folds as training set.

	Set	# subjects (%F)	# images	Avg. age \pm std [min; max]	Avg. MMSE \pm std [min; max]
split 0	train	265 (52.8%)	536	74.9 \pm 6.1 [55.8; 95.0]	29.07 \pm 1.08 [25; 30]
	validation	53 (41.5%)	53	72.8 \pm 5.5 [62.3; 85.3]	28.75 \pm 1.51 [24; 30]
split 1	train	265 (53.5%)	533	74.7 \pm 6.1 [55.8; 95.0]	29.01 \pm 1.16 [24; 30]
	validation	53 (37.7%)	53	73.4 \pm 5.6 [59.9; 88.9]	29.08 \pm 1.25 [25; 30]
split 2	train	265 (49.8%)	537	74.6 \pm 5.9 [55.8; 95.0]	29.01 \pm 1.19 [24; 30]
	validation	53 (56.6%)	53	74.4 \pm 6.6 [63.8; 93.6]	29.06 \pm 1.06 [26; 30]
split 3	train	265 (48.6%)	538	74.7 \pm 6.0 [55.8; 95.0]	28.99 \pm 1.20 [24; 30]
	validation	53 (62.2%)	53	73.9 \pm 6.2 [61.2; 86.2]	29.15 \pm 0.98 [26; 30]
split 4	train	265 (49.1%)	511	75.0 \pm 5.9 [59.7; 95.0]	29.02 \pm 1.19 [24; 30]
	validation	53 (60.4%)	53	72.4 \pm 6.5 [55.8; 84.7]	29.00 \pm 1.08 [25; 30]
split 5	train	265 (51.6%)	510	74.6 \pm 5.9 [55.8; 93.6]	29.01 \pm 1.19 [24; 30]
	validation	53 (47.2%)	53	73.1 \pm 6.6 [59.7; 92.8]	29.08 \pm 1.08 [26; 30]
Test	CN test	60 (63.3%)	60	73.5 \pm 6.6 [59.8; 85.8]	29.27 \pm 1.19 [24; 30]
	AD test	353 (41.1%)	353	75.3 \pm 7.6 [55.1; 90.3]	24.45 \pm 2.64 [19; 30]

Chapter 2

Variational autoencoder for pseudo-healthy reconstruction

In this chapter, we begin by providing an in-depth description of the variational autoencoder framework, proving its suitability as a generative model for unsupervised anomaly detection. Following this, we show the results of preliminary experiments conducted on a toy dataset, before presenting subsequent analyses on real medical images, and more precisely on brain 3D FDG PET.

2.1 Variational autoencoder

A VAE is a deep learning model (Kingma et al., 2014) combining two parameterized models: the encoder or recognition model, and the decoder or generative model.

Let's consider an observed variable \mathbf{x} randomly sampled from an unknown process with an unknown probability distribution $p(\mathbf{x})$. We try to approximate this process with a model $p_\theta(\mathbf{x})$, with parameters θ such that $\mathbf{x} \sim p_\theta(\mathbf{x})$.

Our goal is to approximate the true distribution of the data $p(\mathbf{x})$ with $p_\theta(\mathbf{x})$, by learning a set of parameters θ , such that for any observed \mathbf{x}

$$p_\theta(\mathbf{x}) \approx p(\mathbf{x}) .$$

To do so, we use a deep latent variable model that will allow us to map the complex unknown real distribution $p(\mathbf{x})$ to a latent distribution that can be simple. Let \mathbf{z} be a random vector jointly-distributed with \mathbf{x} , we have

$$p_\theta(\mathbf{x}) = \int_z p_\theta(\mathbf{x}, \mathbf{z}) dz \quad (2.1)$$

where $p_\theta(\mathbf{x}, \mathbf{z})$ represents the joint distribution under p_θ of the observable data \mathbf{x} and its latent representation or encoding \mathbf{z} .

If we apply the chain rule to Equation 2.1, we obtain

$$p_\theta(\mathbf{x}) = \int_z p_\theta(\mathbf{z}) p_\theta(\mathbf{x} | \mathbf{z}) dz \quad (2.2)$$

where:

- $p_\theta(\mathbf{z})$ is the latent space's prior distribution, usually specified by the user. It is often approximated by a Gaussian normal centered distribution $p(\mathbf{z}) = \mathcal{N}(\mathbf{z}; 0, I)$.
- $p_\theta(\mathbf{x} | \mathbf{z})$ is the generative model (also called decoder by analogy with the auto-encoder) that needs to be computed.

However, $p_\theta(\mathbf{x})$ is impossible to compute because it is intractable, we thus cannot optimize the generative model and find θ . Indeed, the posterior $p_\theta(\mathbf{z} | \mathbf{x})$ is intractable, which leads to an intractable joint distribution $p_\theta(\mathbf{x}, \mathbf{z})$. To make it feasible and solve this issue, it is necessary to introduce a parametric inference model $q_\Phi(\mathbf{z} | \mathbf{x})$ to approximate the posterior distribution

$$q_\Phi(\mathbf{z} | \mathbf{x}) \approx p_\theta(\mathbf{z} | \mathbf{x})$$

with Φ parameterizing q . The model $q_\Phi(\mathbf{z} | \mathbf{x})$ is called the recognition model or encoder (by analogy with the auto-encoder).

Finally, we have two models $q_\Phi(\mathbf{z} | \mathbf{x})$ and $p_\theta(\mathbf{x} | \mathbf{z})$ (the encoder and the decoder) with two sets of parameters Φ and θ (the weights of the models) to optimize.

In order to train the model and find optimal values for Φ and θ , we need an optimization criterion. We will use the observations of \mathbf{x} to maximize the likelihood of our model. The idea is to jointly optimize the parameters θ to improve the generated data quality, which means minimizing the reconstruction error, and the parameters Φ such that $q_\Phi(\mathbf{z} | \mathbf{x})$ is the closest to the posterior $p_\theta(\mathbf{z} | \mathbf{x})$. Our criterion is then the addition of a reconstruction error (for instance, we can use the mean squared error or the binary cross entropy) and a distance between the distributions $q_\Phi(\mathbf{z} | \mathbf{x})$ and $p_\theta(\mathbf{z} | \mathbf{x})$. We will here use the Kullback-Leibler (KL) divergence

$$D_{\text{KL}}(q_\Phi(\mathbf{z} | \mathbf{x}) || p_\theta(\mathbf{z} | \mathbf{x})) = \int q_\Phi(\mathbf{z} | \mathbf{x}) \log \frac{q_\Phi(\mathbf{z} | \mathbf{x})}{p_\theta(\mathbf{z} | \mathbf{x})} dx . \quad (2.3)$$

After replacing $p_\theta(\mathbf{z} | \mathbf{x})$ by $\frac{p_\theta(\mathbf{x}, \mathbf{z})}{p_\theta(\mathbf{x})}$ we quickly arrive to

$$\begin{aligned} D_{\text{KL}}(q_\Phi(\mathbf{z} | \mathbf{x}) || p_\theta(\mathbf{z} | \mathbf{x})) &= \log(p_\theta(\mathbf{x})) + \int q_\Phi(\mathbf{z} | \mathbf{x}) \log \frac{q_\Phi(\mathbf{z} | \mathbf{x})}{p_\theta(\mathbf{z}, \mathbf{x})} dx \\ &= \log(p_\theta(\mathbf{x})) - \mathbb{E}_{q_\Phi(\mathbf{z} | \mathbf{x})} \left[\frac{p_\theta(\mathbf{x}, \mathbf{z})}{q_\Phi(\mathbf{z} | \mathbf{x})} \right] , \end{aligned} \quad (2.4)$$

which gives us

$$\log(p_\theta(\mathbf{x})) = \mathbb{E}_{q_\Phi(\mathbf{z} | \mathbf{x})} \left[\log \frac{p_\theta(\mathbf{x}, \mathbf{z})}{q_\Phi(\mathbf{z} | \mathbf{x})} \right] + D_{\text{KL}}(q_\Phi(\mathbf{z} | \mathbf{x}) || p_\theta(\mathbf{z} | \mathbf{x})) . \quad (2.5)$$

Besides, by definition, the KL divergence between $q_\Phi(\mathbf{z} | \mathbf{x})$ and $p_\theta(\mathbf{z} | \mathbf{x})$ is positive or null if both distributions are equal. We can deduce from Equation 2.5 that

$$\log(p_\theta(\mathbf{x})) \leq \mathbb{E}_{q_\Phi(\mathbf{z} | \mathbf{x})} [\log(p_\theta(\mathbf{x}, \mathbf{z})) - \log(q_\Phi(\mathbf{z} | \mathbf{x}))] , \quad (2.6)$$

which means that $\mathbb{E}_{q_{\Phi}(\mathbf{z}|\mathbf{x})} [\log(p_{\theta}(\mathbf{x}, \mathbf{z})) - \log(q_{\Phi}(\mathbf{z} | \mathbf{x}))]$ is the evidence lower bound (ELBO) of the log-likelihood of the function $p_{\theta}(\mathbf{x})$ (Kingma et al., 2014) and define our loss function $\mathcal{L}_{\theta, \Phi}(x)$

$$\begin{aligned} \mathcal{L}_{\theta, \Phi}(x) &= \mathbb{E}_{q_{\Phi}(\mathbf{z}|\mathbf{x})} [\log(p_{\theta}(\mathbf{x}, \mathbf{z})) - \log(q_{\Phi}(\mathbf{z} | \mathbf{x}))] \\ &= \log(p_{\theta}(\mathbf{x})) - D_{\text{KL}}(q_{\Phi}(\mathbf{z} | \mathbf{x}) \| p_{\theta}(\mathbf{z} | \mathbf{x})) . \end{aligned} \quad (2.7)$$

A great advantage of the ELBO is that it allows optimizing both Φ and θ using stochastic gradient descent (SGD).

After a few calculations, the loss of our model can be expressed by the following equation:

$$\mathcal{L}_{\theta, \Phi}(x) = \mathbb{E}_{q_{\Phi}(\mathbf{z}|\mathbf{x})} [\log(p_{\theta}(\mathbf{x} | \mathbf{z}))] - D_{\text{KL}}(q_{\Phi}(\mathbf{z} | \mathbf{x}) \| p_{\theta}(\mathbf{z})) . \quad (2.8)$$

In practice, when training a VAE, we approximate all the distributions by Gaussian distributions

$$p_{\theta}(\mathbf{z}) = \mathcal{N}(\mathbf{z}; 0, I) , \quad (2.9)$$

$$q_{\Phi}(\mathbf{z} | \mathbf{x}) = \mathcal{N}(\mu(\mathbf{x}), \sigma(\mathbf{x})^2) \quad (2.10)$$

with $\mu(\mathbf{x})$ and $\sigma(\mathbf{x})$ being respectively the mean and the standard deviation of the probability distribution of a true sample \mathbf{x} in the latent space. That is to say, the latent representation \mathbf{z} will be sampled from this distribution. This gives us the following formula:

$$\begin{aligned} D_{\text{KL}}(q_{\Phi}(\mathbf{z} | \mathbf{x}) \| p_{\theta}(\mathbf{z})) &= D_{\text{KL}}(\mathcal{N}(\mu(\mathbf{x}), \sigma(\mathbf{x})^2) \| \mathcal{N}(\mathbf{z}; 0, I)) \\ &= -\frac{1}{2} [1 + \log(\sigma(\mathbf{x})^2) - \sigma(\mathbf{x})^2 - \mu(\mathbf{x})^2] , \end{aligned} \quad (2.11)$$

which gives us our loss function

$$\mathcal{L}_{\theta, \Phi}(x) = \mathbb{E}_{q_{\Phi}(\mathbf{z}|\mathbf{x})} [\log(p_{\theta}(\mathbf{x} | \mathbf{z}))] - \frac{1}{2} [\sigma(\mathbf{x})^2 + \mu(\mathbf{x})^2 - \log(\sigma(\mathbf{x})^2) - 1] \quad (2.12)$$

with $\mathbb{E}_{q_{\Phi}(\mathbf{z}|\mathbf{x})} [\log(p_{\theta}(\mathbf{x} | \mathbf{z}))]$ the reconstruction loss.

Finally, if we approximate $p_{\theta}(\mathbf{x} | \mathbf{z})$ with a normal distribution such that $p_{\theta}(\mathbf{x} | \mathbf{z}) = \mathcal{N}(f(\mathbf{z}), cI)$, we can find the mean squared error in our reconstruction loss. This gives us the final loss that we will use

$$\mathcal{L}_{\theta, \Phi}(x) = MSE(\mathbf{x}, f(\mathbf{z})) - \frac{1}{2} [\sigma(\mathbf{x})^2 + \mu(\mathbf{x})^2 - \log(\sigma(\mathbf{x})^2) - 1] . \quad (2.13)$$

However, since \mathbf{z} is sampled from a random stochastic operation that is not differentiable, we cannot perform the gradient back propagation. To tackle this issue, the VAE framework introduces a reparameterization: instead of sampling a random vector \mathbf{z} from $\mu(\mathbf{x})$ and $\sigma(\mathbf{x})$, we use a new variable ϵ , such that $\mathbf{z} = \mu(\mathbf{x}) + \sigma(\mathbf{x}) \cdot \epsilon$ with $\epsilon \sim \mathcal{N}(0, I)$. Therefore, all the operations in the forward process become deterministic, and we can perform SGD.

The VAE is particularly suited for pseudo-healthy reconstruction. Let's consider D a set of medical images collected following a similar protocol. D contains healthy and pathological

images and is the union of two complementary subsets D_h and D_p . For instance, D_h could be a set of healthy FDG PET images $\mathbf{x} \in D$ whose distribution is $p(\mathbf{x})$. The goal of pseudo-healthy image reconstruction is to reconstruct an FDG PET image of healthy appearance given an input $\mathbf{x} \in D$. During the training process, an approximation of the posterior distribution $q_\phi(\mathbf{z} | \mathbf{x})$ is learned for $x \in D_h$ as the model is trained using only healthy subjects. In other words, the healthy image true distribution $p(\mathbf{x})$ is approximated with the learned parametric distribution $p_\theta(\mathbf{x})$ such that $p_\theta(\mathbf{x}) \approx p(\mathbf{x})$. During reconstruction, this approximate posterior is used to estimate the latent variable \mathbf{z} for $\mathbf{x} \in D$ (it can be from D_h or D_p), i.e., the images (of healthy subjects or patients) are projected into that “healthy images” learned subspace. Then, the decoder can generate healthy images from \mathbf{z} .

2.2 Pseudo-healthy reconstruction on a toy dataset

We run a preliminary set of experiments on a toy dataset to study a simple case of anomaly detection in a controlled setting. We generated a dataset of 2D synthetic Shepp-Logan phantoms (Shepp et al., 1974). Shepp-Logan phantoms are 2D images composed of ten ellipses that schematize human brains (Figure 2.1). It was first used in the seventies (Shepp et al., 1974) to develop and test human brain image reconstruction algorithms. It is now widely used as a brain model in computational neuroimaging.

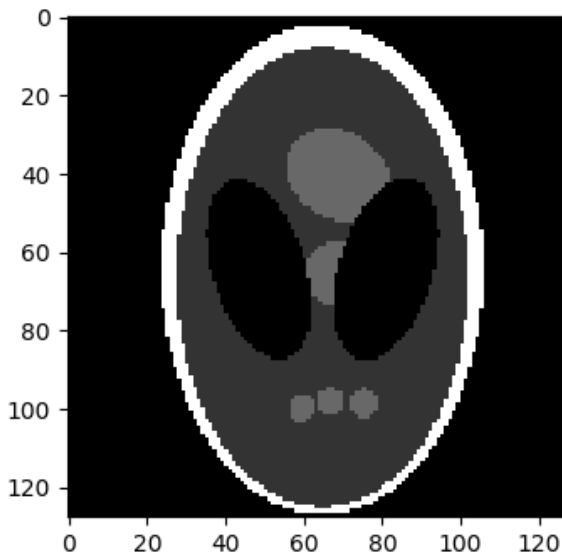


FIGURE 2.1: Shepp-Logan phantom generated by ClinicaDL. We can distinguish different regions of interest: the two ventricles in black, the first region of interest at the top and the second region of interest constituted of three small circles at the bottom

2.2.1 Shepp-Logan dataset

The dataset was generated using ClinicaDL¹ (Thibeau-Sutre et al., 2022b). The different parameters, such as the size of the ellipses and intensity of the different regions, are random to have different images. The algorithm can synthesize Shepp-Logan phantoms considered as cognitively normal (CN), or images considered as pathological in which some regions

¹<https://clinica dl.readthedocs.io/en/latest/Preprocessing/Generate/>

of interest appear smaller. We generated a total of 2000 images of size 128×128 : 1000 normal images and 1000 pathological images. Only normal images are used for training and validation, and all pathological images belong to the test set.

TABLE 2.1: Distribution of the data in the train, validation and test sets according to the labels of the generated images

Set	Composition
Training	900 normal images
Validation	50 normal images
Test normal	50 normal images
Test pathological	1000 pathological images

2.2.2 2D convolutional VAE

Our model architecture is a simple convolutional VAE. It is composed of a symmetric encoder-decoder architecture, with four 2D convolutional layers in the encoder and four 2D transpose convolutional layers in the decoder. The latent space is 2D with a size of 8×8 .

We used the pixel-wise binary cross entropy (BCE) loss as reconstruction criterion, even though it might not be the best choice (since the value of the pixel are in the range $[0, 1]$). A reconstruction loss such as the L1 loss or MSE would be more appropriate. We accidentally used the BCE because we prototyped our model on the MNIST dataset whose images are composed of binary pixels (0 or 1). Since the results were satisfying, we did not retrain the model and kept the BCE for this preliminary experiment.

Our total loss \mathcal{L} is the sum of the BCE and the KL divergence

$$\mathcal{L}(x, \hat{x}) = BCE(x, \hat{x}) - \frac{1}{2} [\sigma(\mathbf{x})^2 + \mu(\mathbf{x})^2 - \log(\sigma(\mathbf{x})^2) - 1] . \quad (2.14)$$

We trained our model over 50 epochs, with a batch size of 2, a learning rate of 10^{-4} , and we used the Adam optimizer. In total, the training lasted 11 min on a GPU.

2.2.3 Results

We first evaluated our model on the normal images to see if the reconstruction of images similar to the training samples was correct. As we can see in Figure 2.2, the reconstruction image is very similar to the real image. The main difference is the blurriness of the two regions of interest located at the top and the bottom of the phantom. Otherwise, the intensity of the different regions are well reconstructed as we can see on the difference map and the shapes are accurate enough.

We then reconstructed pathological images from test pathological to see if the model can correct the anomaly. On the example displayed in Figure 2.3, we can see that the abnormal zone, which is the region at the top of the image, is not well reconstructed. Indeed, we simulate the pathology by making this region smaller. On the reconstructed phantom, this

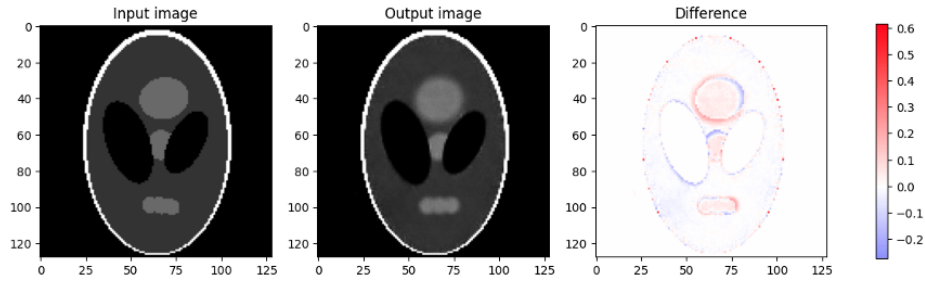


FIGURE 2.2: Reconstruction of Shepp-Logan phantom on two different normal images. We can see that the reconstruction is accurate: the ventricles and the border of the images are well reconstructed. We can also observe that the shapes are less sharp, especially for the regions of interest.

region of interest is bigger than on the input, as highlighted in the difference map. This is the expected behavior, since the model only learned to reconstruct normal images. In that case, the model was not able to reconstruct the anomaly as is, and reconstructed the abnormal region larger than on the input.

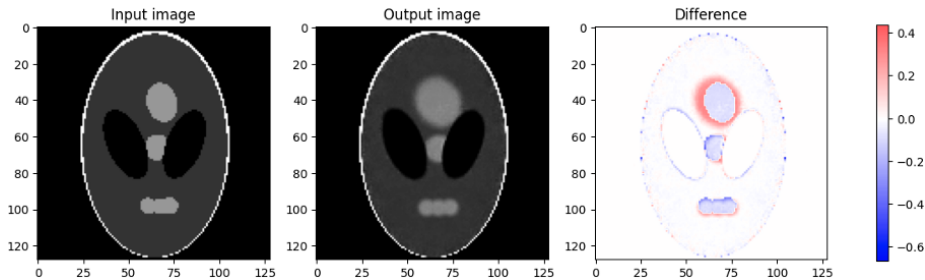


FIGURE 2.3: Reconstruction of Shepp-Logan phantom on two different pathological images. We can observe that the atrophied region of interest is generated larger than on the real image.

We notice that, in both cases, the reference region at the bottom of the phantom (the three circles) are in general not reconstructed very well: the model seems to reconstruct them always well aligned and with three circles of the same size. We also generated images with anomalies in this region by generating smaller circles, or having two or four circles instead of three; and in every case the model reconstructed the same pattern for this region of interest, that is three aligned circles with the same size. On one hand, it is a positive result since it means that the anomalies are not reconstructed by the model. On the other side, this also means that the difference between the subjects are not well learned by the VAE.

Even if the results seem visually acceptable, we used quantitative reconstruction metrics (SSIM, MSE and PSNR, more details in Section 3.2.1) to evaluate the performance of the model on the whole test sets. Results are reported in Table 2.2. The MSE is just above 10^{-3} for both normal and pathological images, and the SSIM is above 0.95 in both cases, indicating a good reconstruction ability of the VAE. Moreover, the reconstruction scores are slightly better for normal images than for pathological images. This is a positive point because it is expected that the model reconstructs the normal images better than the abnormal ones (since it should not reconstruct the anomalies well).

TABLE 2.2: Reconstruction performance of our VAE on normal and pathological images. The reconstruction is slightly better on normal images than on pathological images.

Test set	SSIM \uparrow	MSE ($\times 10^{-3}$) \downarrow	PSNR \uparrow
Normal (50 images)	0.958	1.09	29.68
Pathological (1000 images)	0.954	1.28	29.01

To test the robustness of the model, we artificially created different kind of anomalies. We generated new images for testing purposes by changing the shape of some structures in the Shepp-Logan phantom, their number, their intensity, and tried different distortions on the image. This allows us to evaluate how the model reacts to different kinds of anomaly, and not only pathological images that resemble normal images. In Figure 2.4, we display some of the results of this experiment. Other results are available in Appendix B. If the anomaly is not too severe, such as the elastic deformation on the third image, or local, such as the addition of a squared artifact on the first image or the change of an ellipse by a triangle on the second image, then the model can reconstruct a normal version of the image. However, if the deformation is too important, such as in the last row, then the model cannot reconstruct a normal image, probably because the input image is too far from the normal image distribution.

An idea to understand why the model behaves like this is to look at the latent representation of these images.

2.2.4 Latent space analysis

One of the main advantages of using VAEs for pseudo-healthy reconstruction is the regularized latent space that provides a great insight of the model’s behavior. To this end, we display in Figure 2.5 the distribution of the latent vectors of both test sets, after reducing the latent space dimension from 64 to 2 using a principal component analysis (PCA). Although the first two components of the PCA explain only 20% of the variance, we can still make some interesting observations. First, all the normal images define a unique region in the graph (in orange). The pathological images, in blue, are also located in the same region, possibly explaining why they are reconstructed as normal. Indeed, the decoder part of the VAE learned to reconstruct normal images from latent samples distributed in this part of the latent space. Finally, we plotted in green the latent representation of the abnormal images that we created to test the robustness of the method. We observe that most of them are also distributed in the same region, explaining why the anomalies are reconstructed as normal by the VAE. We nevertheless notice that there are few points out of the distribution. For instance, one of them, highlighted in red, corresponds to the latent representation of the bottom image of Figure 2.4. Since this image is very different from the normal image distribution, the encoder is not able to deduce a latent representation that corresponds to the training data distribution. Consequently, the reconstruction from this latent vector by the decoder is different from a normal Shepp-Logan phantom.

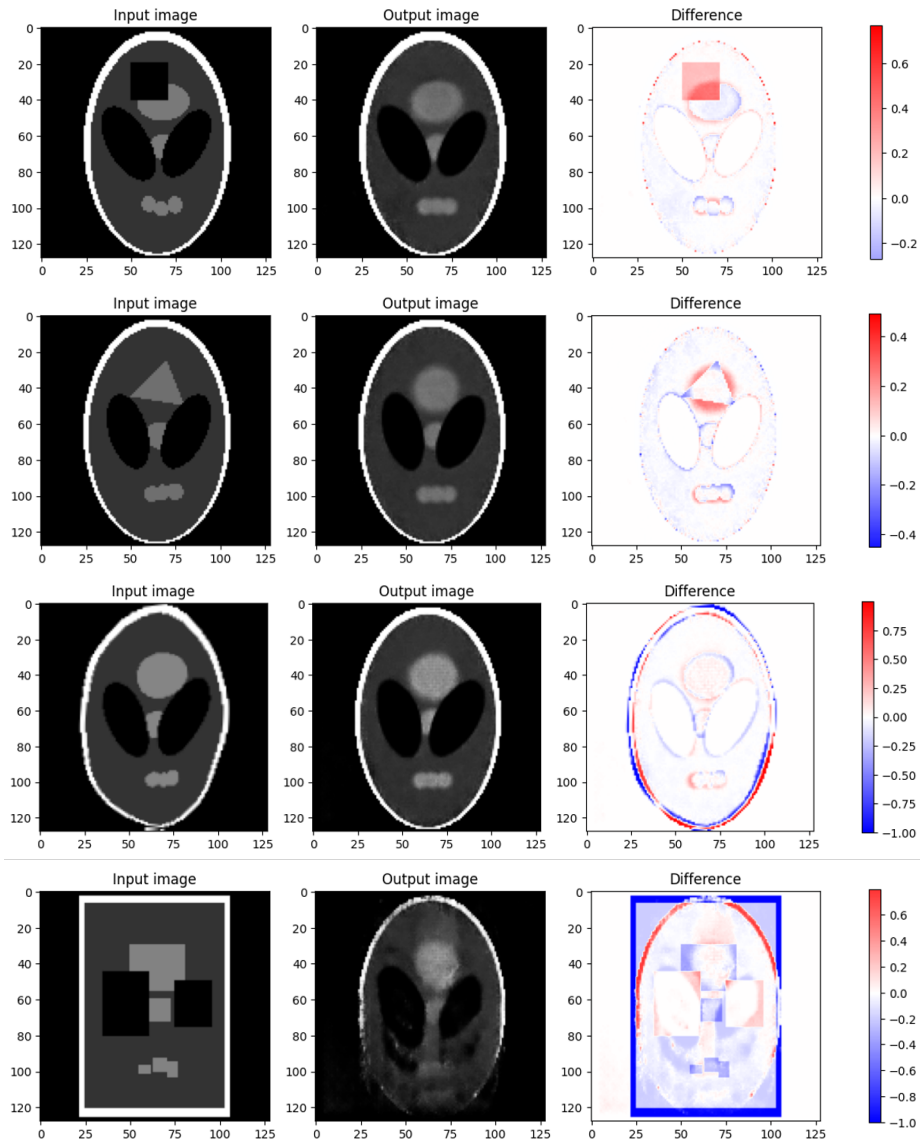


FIGURE 2.4: VAE pseudo-healthy reconstruction on images with different kinds of anomalies: presence of a black square hiding a part of the image (top row), changing the shape of a region of interest from an ellipse to a triangle (second row), applying an elastic deformation on the image (third row), and replacing all the ellipses by rectangles (last row).

From this preliminary experiment on Shepp-Logan data, we conclude that the VAE is a promising generative model for pseudo-healthy reconstruction. It can learn the training data distribution, and when trained only with normal data, it allows detecting anomalies from inaccurate reconstructions of pathological images. Moreover, the VAE is simple to implement, easy to train, and efficient both in terms of computation and resources used. Finally, by analyzing its regularized latent space, we can understand the behavior of the model and explain the results we obtained. We will now apply the VAE framework to real medical images with the aim to detect dementia-related anomalies.

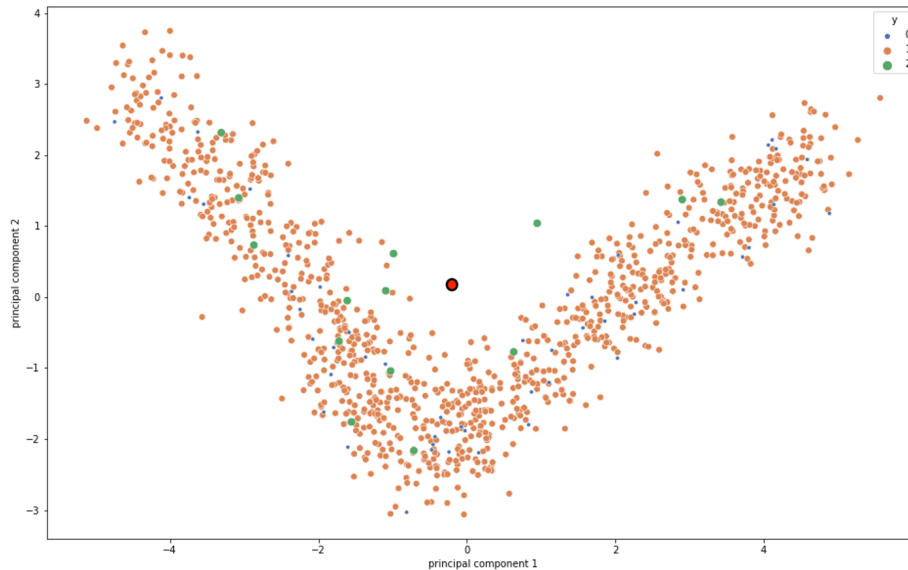


FIGURE 2.5: Visualization of the latent space after a PCA. We observe that almost all the images are in the same zone. The image that is out of the distribution is not correctly reconstructed.

2.3 Pseudo-healthy reconstruction on FDG PET images

In this section, we present the results we obtained with a VAE for the pseudo-healthy reconstruction of brain 3D FDG PET images.

2.3.1 Experimental setting

Materials

As explained in details in Section 1.2, we use FDG PET images from the ADNI dataset (Mueller et al., 2005; Jagust et al., 2010; Jagust et al., 2015). The images are preprocessed using the `pet-linear` pipeline from the Clinica open source software (Routier et al., 2021). They are then carefully selected, for a final set of 739 images from 378 CN subjects and 353 images at baseline from 353 AD patients. We split our CN subjects into train/validation and test sets at the subject level, and perform a 6-fold cross validation on the train/validation set using the ClinicaDL open source software (Thibeau-Sutre et al., 2022b). All the AD patients belong to the test set.

3D convolutional VAE

We opt for a 3D convolutional VAE as VAEs have already shown their efficacy for UAD in medical imaging (Baur et al., 2021a; Chen et al., 2022): they are easy to train, easily scalable, are able to handle small datasets, and with good interpretation capacity thanks to their regularized latent space. Moreover, the VAE framework allows us to learn the training data distribution, which is an import point specifically for our study, as we will see in Chapter 4. We implement a 3D VAE to fully exploit the 3D context of high resolution PET images.

The VAE’s encoder is composed of five convolutional blocks that are the succession of a 3D convolutional layer and a batch normalization with a ReLU activation. Then the vector is flattened and passes through a dense layer to output the latent space of size 256 in one dimension. Our decoder is almost symmetrical as it transforms a single vector sampled from the latent space in a 3D image with a dense layer followed by four deconvolutional blocks that are composed of an upsampling layer, a 3D convolutional layer and a batch normalization with a leaky ReLU activation. The output block is composed of an upsampling layer and a 3D convolutional layer with a sigmoid activation. Encoder convolutional layers have a kernel size of (4, 4, 4), a stride of (2, 2, 2) and a padding of (1, 1, 1) while decoder convolutional layers have a kernel size of (3, 3, 3), a stride of (1, 1, 1) and a padding of (1, 1, 1). A detailed schema of the VAE we use can be found in Figure 2.6.

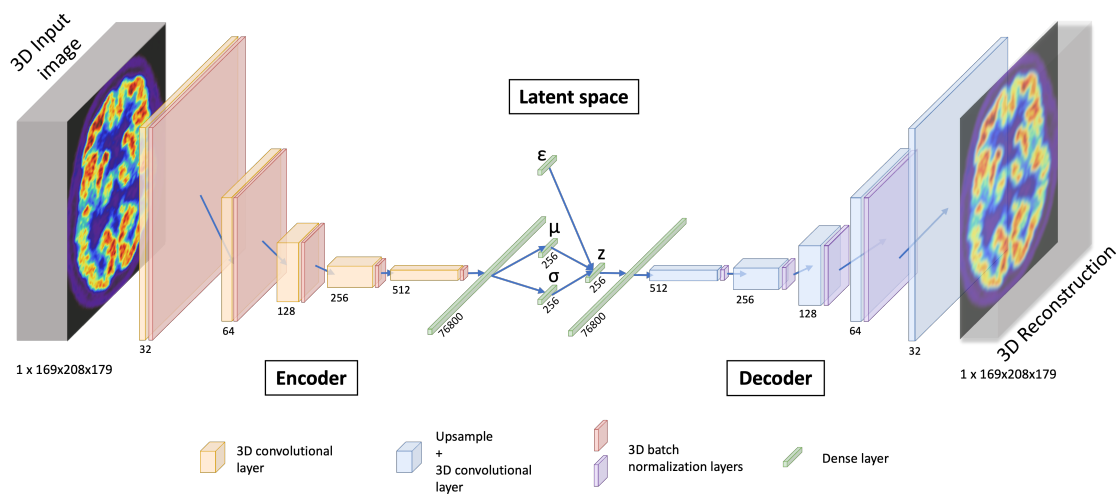


FIGURE 2.6: Architecture of the 3D convolutional VAE.

Model training

Our 3D convolutional VAE implementation relies on the open-source python library Pythae (Chadebec et al., 2022). The model was trained on 200 epochs, with a learning rate of 10^{-5} using the ClinicaDL (Thibeau-Sutre et al., 2022b) software that aims to facilitate the use of neuroimages with deep learning and improve reproducibility of the experiments.

We trained the VAE on Jean Zay high performance computer cluster with Nvidia Tesla V100 GPUs that have 32GB of memory, which allowed us to use a batch size of 8. It took approximately 10 hours to train each fold over the 200 epochs.

Model evaluation

In order to accurately detect anomalies, we first need to assess if the model is able to accurately reconstruct 3D brain FDG PET. Therefore, we use pairwise reconstruction metrics between the input and reconstructed images (Nečasová et al., 2022). The reconstruction metrics we use are the mean-squared error, the peak signal-to-noise ratio, the structural similarity (Wang et al., 2004), and the multi-scale structural similarity (Wang et al., 2003). For more information, please refer to Chapter 3, Section 3.2.1.

2.3.2 Results

The variational autoencoder is trained on six folds in order to evaluate the variance due to data splitting (Bouthillier et al., 2021).

The results obtained on the test set are summarized in Table 2.3. We first observe that the MSE is almost identical on the six folds, with a mean-squared reconstruction error around 1.82×10^{-3} . The performance measured with PSNR are also very similar, which is coherent as the PSNR is a function of the MSE. We also report that the SSIM varies from 0.874 on average to 0.879 between the folds 4 and 2.

TABLE 2.3: Reconstruction metrics obtained on the test set for images from healthy subjects over the 6 folds.

Dataset	Fold	MSE ($\times 10^{-3}$) \downarrow	PSNR \uparrow	SSIM \uparrow	MS-SSIM \uparrow
CN test set	0	1.834 ± 0.654	27.527 ± 1.080	0.875 ± 0.027	0.944 ± 0.015
	1	1.815 ± 0.649	27.572 ± 1.074	0.878 ± 0.026	0.944 ± 0.014
	2	1.841 ± 0.746	27.532 ± 1.113	0.879 ± 0.025	0.944 ± 0.015
	3	1.826 ± 0.665	27.547 ± 1.079	0.876 ± 0.027	0.943 ± 0.014
	4	1.858 ± 0.619	27.456 ± 1.031	0.874 ± 0.026	0.944 ± 0.014
	5	1.836 ± 0.727	27.546 ± 1.134	0.876 ± 0.028	0.943 ± 0.015

For the following experiments, as it is difficult to interpret the results across several folds for a reconstruction task, we select a fold according to the SSIM, which is a perceptual metric substantially different from the loss being minimized. We select the fold 1, which presents an average SSIM on the validation set similar to that of the other folds, but has the highest minimum SSIM (see Table 2.4).

TABLE 2.4: SSIM obtained on the validation set for all the splits.

	Split 0	Split 1	Split 2	Split 3	Split 4	Split 5
mean	0.861	0.865	0.876	0.871	0.868	0.867
std	0.035	0.024	0.030	0.037	0.026	0.034
min	0.698	0.801	0.741	0.66	0.795	0.736
25%	0.852	0.852	0.870	0.863	0.854	0.858
50%	0.870	0.872	0.884	0.881	0.870	0.873
75%	0.882	0.882	0.893	0.891	0.887	0.890
max	0.912	0.905	0.923	0.907	0.908	0.907

The reconstruction metrics of the model trained on the split 1 appear consistent with the reconstructions of images of CN subject that are displayed in Figure 2.7. The reconstructions are acceptable for 3D full resolution images: the shape of the brain and the main structures, such as the ventricles, are well captured. However, smaller anatomical features such as cortical folds are not well reconstructed.

Once we validated that the reconstruction quality of the VAE is acceptable, we test it on images of AD patients to verify that we can reconstruct pseudo-healthy images that allow us to detect hypometabolism. Some examples of reconstructions from AD patients

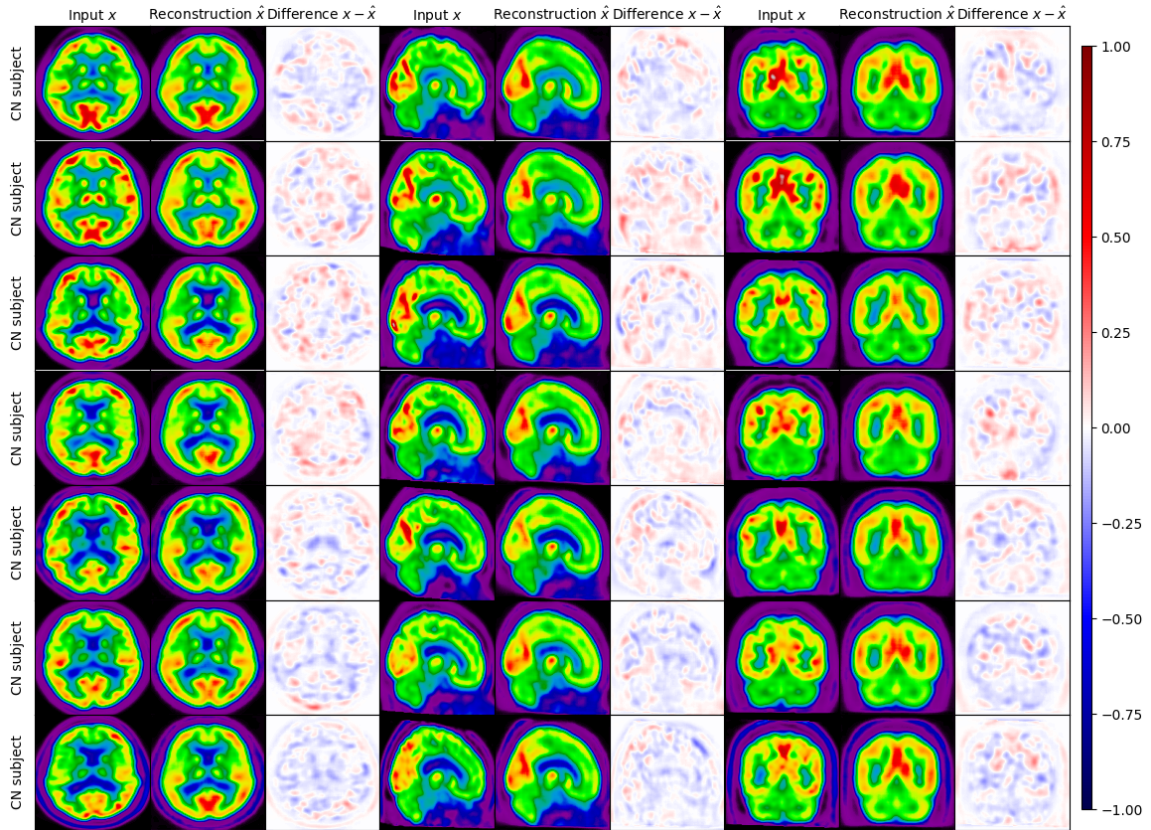


FIGURE 2.7: Examples of reconstructions obtained from real images of CN subjects (even rows). For each plane, the first image is the input, the second one the model’s reconstruction and the third one the difference (input - reconstruction).

are displayed in Figure 2.8. We first notice that the intensity of the reconstructed image is overall higher than the input image, as the difference maps are mostly blue. Moreover, the hypometabolism are reconstructed as pseudo-healthy (or at least with a higher intensity). It is particularly visible on the second, fourth and fifth row, where the difference maps show big differences in the parietal lobe.

To be consistent with the literature, we also compared the reconstruction metrics computed on images of AD patients with the metrics computed on images of CN subjects. We report the results in Table 2.5. We can see that the reconstruction performance is slightly better on the images from CN subjects, which is a positive point because, as the model should correct anomalies on images from AD patients, the reconstruction error is expected to be higher. However, the difference is not large enough to separate healthy controls from AD patients.

TABLE 2.5: Comparison of reconstruction metrics computed on test set with healthy subjects and test set with patients with AD.

Simulated dementia	MSE ($\times 10^{-3}$) \downarrow	PSNR \uparrow	SSIM \uparrow	MS-SSIM \uparrow
CN	1.815 ± 0.649	27.572 ± 1.074	0.878 ± 0.026	0.944 ± 0.014
AD	2.554 ± 1.391	26.272 ± 1.560	0.853 ± 0.045	0.928 ± 0.025

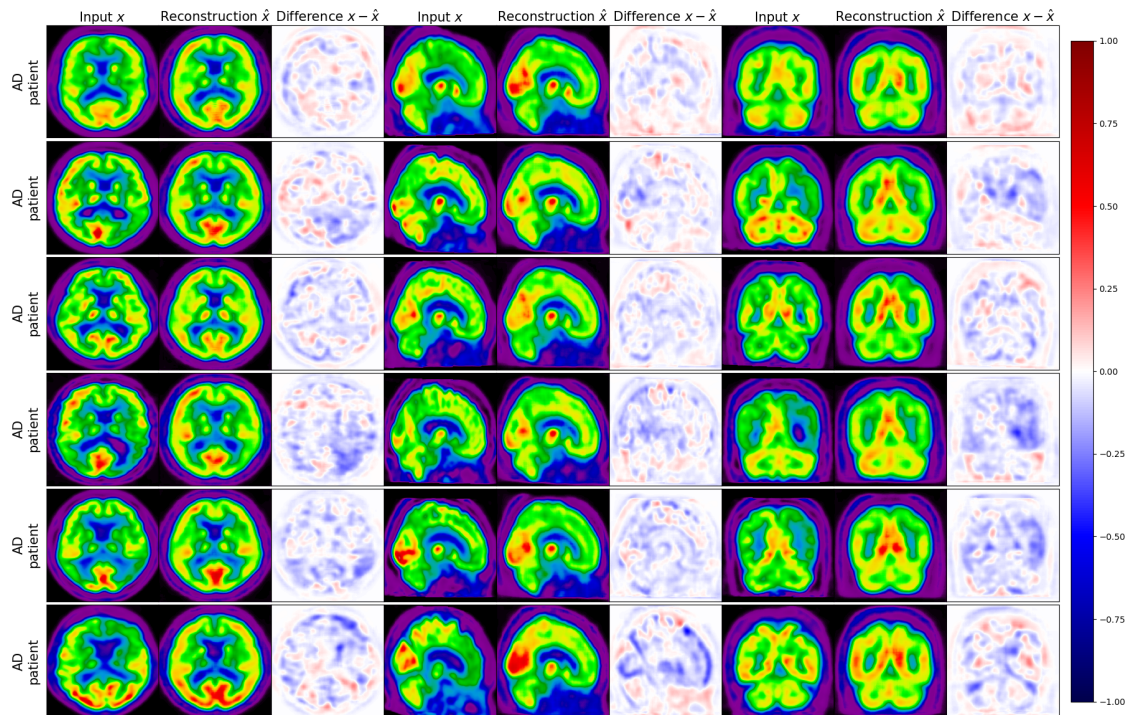


FIGURE 2.8: Examples of reconstruction obtained from real images of patients. For each plane, the first image is the input, the second one the model’s reconstruction and the third one the difference (input - reconstruction).

2.3.3 Discussion and limitations

Using images from healthy subjects and common reconstruction metrics, we confirmed that the VAE was able to reconstruct subject-specific 3D FDG PET. However, the quality of the reconstruction can be improved, as we observe that the VAE does not generate the sharp details of the image very well. The reconstruction is quite blurry, which is expected as the model is rather simple and generating high resolution 3D images is a difficult task. The results are still satisfactory for FDG PET images at this resolution since they are smooth by nature, but we can imagine that the VAE will be limited to other modalities that have a lot of structural details, such as anatomical MRI. Even though the reconstruction quality may be sufficient to detect anomalies, it would be difficult to qualify the reconstruction as being pseudo-healthy at this stage of the evaluation (Baur et al., 2021a). Moreover, using only reconstruction error to differentiate between healthy subjects and patients is not robust enough. The major weakness is that little intense anomalies might be difficult to detect for several reasons: first the model is able to reconstruct a meaningful image even though the image is abnormal, the reconstruction error due to abnormal regions will be drowned in the reconstruction noise due to model imperfection, and the reconstruction metrics being computed on the whole image may not highlight significant difference (Meissen et al., 2021).

In most applications of pseudo-healthy synthesis for UAD, the performance of the model on real diseased images is measured with a similarity metric using a ground truth anomaly mask. This has the advantage of giving a quantitative measure for anomaly detection, however this does not really measure if the reconstructed image is pseudo-healthy. Evaluating models on anomaly detection only may lead to incomplete and biased evaluation,

as the model might not be able to detect every kind of anomalies. In our case, the final validation step would be to ask a clinician to evaluate the healthiness of the reconstructed images. However, having a clinician manually rate images is very time-consuming and can be expensive.

If annotated data are not needed for training, testing the model without labels results in approximate and incomplete evaluation. This highlights the need of having ground truth masks for lesions we want to detect, or the target pseudo-healthy reconstruction for images with anomalies. We will see in the next chapter, how to use simulated data in order to overcome this limitation.

Chapter 3

Evaluation and validation of unsupervised anomaly detection methods in neuroimaging

This chapter is a part of an article published in the Special Issue for Generative Models of Machine Learning for Biomedical Imaging.

- **Title:** Evaluation of pseudo-healthy image reconstruction for anomaly detection with deep generative models: Application to brain FDG PET
- **Authors:** Ravi Hassanaly, Camille Brianceau, Maëlys Solal, Olivier Colliot and Ninon Burgos
- **DOI:** [10.59275/j.melba.2024-b87a](https://doi.org/10.59275/j.melba.2024-b87a)

In this chapter, we introduce a framework for the evaluation of pseudo-healthy reconstruction approaches in the absence of ground truth. This framework consists in simulating anomalies on images of healthy subjects to generate pairs of pathology-free and pathological (e.g., mimicking dementia-like lesions) images. We complement the framework by defining new healthiness and anomaly metrics. The healthiness metric measures whether the reconstructed image is of healthy appearance to evaluate the model capacity to reconstruct pseudo-healthy images, whereas the anomaly metric measures whether the input image contains anomalies using both the pseudo-healthy reconstruction and the input image. A preliminary version of this work was published as a conference paper (Hassanaly et al., 2023a).

3.1 Evaluation of UAD approaches in the literature

In the literature about unsupervised anomaly detection in neuroimaging, many studies have used the BraTS (glioma) (Menze et al., 2014), ISLES (multiple sclerosis lesions) (Maier et al., 2017) or ATLAS (stroke lesions) (Liew et al., 2017) datasets, which directly provide ground truth anomaly masks (Zimmerer et al., 2018; Zimmerer et al., 2019; Chen et al., 2018b; Baur et al., 2021a; Bercea et al., 2023c; Bercea et al., 2023d; Lüth et al., 2023;

Wagnier-Dauchelle et al., 2023; Pinaya et al., 2022b; Chatterjee et al., 2022; Luo et al., 2023; Bengs et al., 2022; Xia et al., 2019; Xia et al., 2020; Sun et al., 2020). Other studies have used in-house data that may include ground truth anomaly masks (Baur et al., 2019; Baur et al., 2021b; Siddiquee et al., 2023; Alaverdyan et al., 2020; Luo et al., 2023; Han et al., 2021). In that case, the evaluation of the model is straightforward: one only has to compute a metric such as the dice score between the predicted anomaly and the ground truth, as we would do for the evaluation of supervised anomaly segmentation. Some works have gone further by introducing new original metrics: Xia et al., 2020 defined a "healthiness" metric, using a segmentation network to estimate the size of a potential lesion in the pseudo-healthy reconstruction; and an "identity" metric, based on a multi-scale structural similarity index on non-pathological tissues. In most of the other cases, when the ground truth anomaly mask is not available, the evaluation consists in applying a classifier to the reconstructed images that was trained to distinguish pathological and healthy images, or using the reconstruction error itself from which an anomaly score is derived. One way to improve the evaluation is to use synthetic data by corrupting real healthy data with sprites (Bercea et al., 2023c; Pinaya et al., 2022b).

Strategies developed for pseudo-healthy reconstruction and, more generally, unsupervised anomaly detection, often lack rigorous evaluation. Furthermore, the majority of studies utilize 2D images, with very few focusing on PET images. This is why, in this chapter, we introduce an evaluation framework particularly adapted for experiments where ground truth data is unavailable. Subsequently, we apply this framework to conduct a rigorous evaluation of a 3D model for pseudo-healthy reconstruction. We apply this model to the detection of anomalies associated with dementia, a task that has received limited exploration and presents significant challenges.

3.2 Pseudo-healthy image reconstruction evaluation procedure

Rigorous and in-depth evaluation of machine learning models and of their training procedure is crucial, especially in the medical field as overestimated or biased results may lead to dramatic consequences (Varoquaux et al., 2022). As far as we know, there is no guidelines nor standard procedure for the evaluation of pseudo-healthy reconstruction for UAD, especially when a ground truth of the anomalies that should be detected is not available. We propose here such procedure.

In this context, we can identify two objectives: i) preserve the identity of the subject in the reconstructed image, ii) reconstruct an image of healthy appearance (Xia et al., 2020). We have to evaluate the performance of the model for both objectives. For the first one, we can measure the similarity between images of healthy subjects and their reconstruction. With regard to the second objective, we can either evaluate whether the images reconstructed are looking healthy (pseudo-healthy reconstruction task), or measure if the anomalies detected by this method correspond to the real anomalies present in the image (anomaly detection task). However, depending on the type of disorder studied, we may not have ground truth healthy images, nor ground truth anomaly masks, so we cannot use a metric to quantify how healthy the reconstructed images are nor how well anomalies are

detected. This is why we developed an evaluation framework that consists in simulating an abnormal image \mathbf{x}' from a healthy image \mathbf{x} in order to have a pair with a diseased image and its healthy version. To evaluate the healthiness of a reconstructed image $\widehat{\mathbf{x}'}$ from an abnormal simulated image \mathbf{x}' , we can measure the similarity between the pseudo-healthy reconstruction $\widehat{\mathbf{x}'}$ and the the original healthy image \mathbf{x} .

3.2.1 Evaluation metrics for image reconstruction

The first step to validate a pseudo-healthy reconstruction model is to evaluate the quality of the reconstruction in the case of images of healthy subjects. We use four metrics that are common in the image synthesis literature (Nečasová et al., 2022): the mean squared error (MSE), the peak signal-to-noise ratio (PSNR), the structural similarity index (SSIM) (Wang et al., 2004) and the multi-scale structural similarity (MS-SSIM) (Wang et al., 2003). This also aims to validate the fact that the model can reconstruct images that are as healthy as they originally look.

Mean Absolute Error The MAE is simply the mean of each absolute value of the difference between the true pixel X_i and the generated pixel \widehat{X}_i

$$MAE = \frac{1}{n} \sum_{i=1}^n |X_i - \widehat{X}_i| . \quad (3.1)$$

This metric needs to be minimal and is equal to 0 if both images are identical.

Mean squared error The MSE is the mean of the square of the difference between the true pixel X_i and the reconstructed pixel \widehat{X}_i

$$MSE = \frac{1}{n} \sum_{i=1}^n (X_i - \widehat{X}_i)^2 . \quad (3.2)$$

A low MSE means that the images are close to each other. They are identical if the MSE is 0.

Peak signal-to-noise ratio The PSNR is a function of the MSE and allows for comparing images encoded with different dynamic ranges

$$PSNR = 10 \log_{10} \left(\frac{MAX^2}{MSE} \right) , \quad (3.3)$$

with MAX the maximum possible value of the image. We can see that if the images are similar, the MSE is close to 0, and the PSNR tends toward $+\infty$.

Structural similarity The SSIM is a weighted combination of three comparison measurements between the true image X and the reconstructed image \widehat{X} : the luminance l , the contrast c and the structure s (Wang et al., 2004)

$$l(X, \hat{X}) = \frac{2\mu_X\mu_{\hat{X}} + c_1}{\mu_X^2 + \mu_{\hat{X}}^2 + c_1}, \quad c(X, \hat{X}) = \frac{2\sigma_X\sigma_{\hat{X}} + c_2}{\sigma_X^2 + \sigma_{\hat{X}}^2 + c_2}, \quad s(X, \hat{X}) = \frac{\sigma_X\hat{X} + c_3}{\sigma_X\sigma_{\hat{X}} + c_3},$$

$$SSIM = l(X, \hat{X})^\alpha \cdot c(X, \hat{X})^\beta \cdot s(X, \hat{X})^\gamma,$$

with α , β and γ the weights assigned to each measurement. If we set them all to 1, we obtain the following formula (Wang et al., 2004):

$$SSIM = \frac{(2\mu_X\mu_{\hat{X}} + c_1)(2\sigma_X\sigma_{\hat{X}} + c_2)}{(\mu_X^2 + \mu_{\hat{X}}^2 + c_1)(\sigma_X^2 + \sigma_{\hat{X}}^2 + c_2)}, \quad (3.4)$$

where:

- μ_X and $\mu_{\hat{X}}$ are the means of the true and reconstructed image respectively,
- σ_X and $\sigma_{\hat{X}}$ are the standard deviations of the true and reconstructed image respectively,
- c_1 and c_2 are positive constants to stabilize the division. Typical values are $c_1 = 0.01$ and $c_2 = 0.03$.

The SSIM ranges between 0 and 1, with 1 meaning that the two images are identical.

Multi-scale structural similarity The MS-SSIM is similar to the SSIM: it is a weighted combination of the luminance l , the contrast c and the structure s computed at different scales between the true image X and the reconstructed image \hat{X} (Wang et al., 2003). To do so, we iteratively apply M times a 3D average pooling, which is a low pass filter, to down-sample the images by a factor of two.

$$MS-SSIM = l_M(X, \hat{X})^{\alpha_M} \cdot \prod_{j=1}^M c_j(X, \hat{X})^{\beta_j} \cdot s_j(X, \hat{X})^{\gamma_j}, \quad (3.5)$$

with l_j , c_j and s_j being respectively the luminance, the contrast and the structure between X and \hat{X} at the scale j . We choose $M = 5$, $\alpha_j = \beta_j = \gamma_j$ and $\alpha_j, \beta_j, \gamma_j$ taking the following values (0.0448, 0.2856, 0.3001, 0.2363, 0.1333) for $j \in [1, M]$ based on the values introduced in the original paper from Wang et al., 2003.

However, since the reconstruction metrics are computed on the whole 3D image and not only in the abnormal region, their values do not substantially vary when computed for healthy or abnormal images, as the major part of the image is normal. This is amplified when the anomalies are subtle. Thus, we cannot rely only on whole image reconstruction metrics to differentiate healthy subjects from patients.

3.2.2 Simulation-based evaluation framework

In practice, the healthy version of an image with anomalies is rarely available, it is thus impossible to measure the healthiness of the reconstructed image. In most of the studies on UAD in medical imaging, the datasets provide lesion masks that can be used as ground

truths. In this case, one can simply compute a metric such as the dice score between the anomaly map generated (and usually post-processed to binarize it) and the real lesion mask to evaluate the capacity of the model to detect and localize anomalies. This can be used as a proxy measure of the healthiness of the reconstructed image: if we perfectly detect lesions, it means that the reconstruction is healthy compared to the input image. However, when studying disorders such as dementia, such lesion masks are not available.

Another way to evaluate the healthiness of the reconstructed images would be to consult a neuro-radiologist or nuclear physician. However, manually rating images is time-consuming, especially if the aim is to compare different models, and possible only for small datasets. We are thus looking for a strategy to automatically evaluate models, as a preliminary validation, before soliciting clinicians.

The idea is to simulate abnormal images to evaluate our model. In the literature, anomalies are often under the form of sprites (i.e. non-realistic artifacts added to the images) (Bercea et al., 2023c; Pinaya et al., 2022b). However, this is not satisfactory as we try to detect subtle and less intense anomalies. Realistic anomaly generation has also been explored, mainly to study the progression of diseases such as cancer or dementia (Manzanera et al., 2021). When reducing the scope to neuroimaging, most anomaly generation methods are applied to structural MRI; for instance to simulate the growth of a glioblastoma (Ezhov et al., 2023), or the progression of atrophy in case of dementia (Khanal et al., 2017; Ravi et al., 2022). The proposed approaches often rely on complex modeling or the use of deep learning.

We here propose to simply generate new test sets by simulating hypometabolism on healthy images to have pairs of healthy (considered as ground truth) and abnormal images. For this purpose, we designed a mask corresponding to regions associated with AD (parietal and temporal lobes) (Landau et al., 2012) that were extracted from the third automated anatomical labelling (AAL3) atlas (Rolls et al., 2020). To obtain a realistic simulated image, we smoothed the mask with a Gaussian convolution filter with $\sigma = 5$. We then reduced the intensity of the PET signal within the region defined by the mask by different factors to simulate various degrees of hypometabolism as illustrated in Figure 3.1.

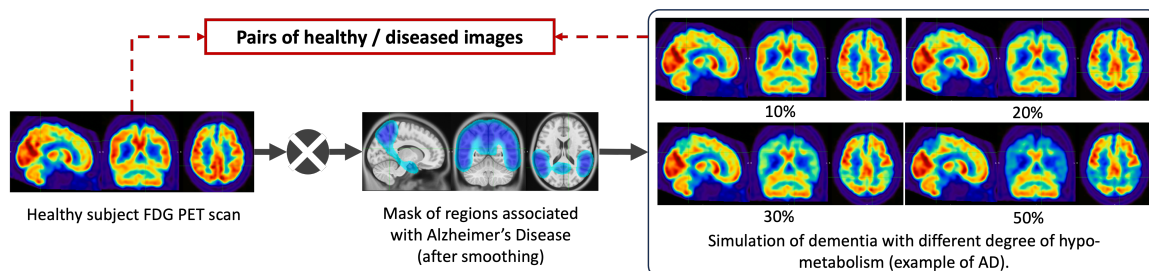


FIGURE 3.1: Hypometabolism simulation pipeline. The intensity of the image from a healthy subject is reduced by a chosen factor in a region associated with a dementia.

Having such pairs of images allows us to compare the pseudo-healthy image reconstructed by the model $\hat{\mathbf{x}}'$ from images presenting anomalies \mathbf{x}' with their corresponding healthy images \mathbf{x} (Figure 3.2), hence better evaluating the model capacity to synthesize pseudo-healthy images.

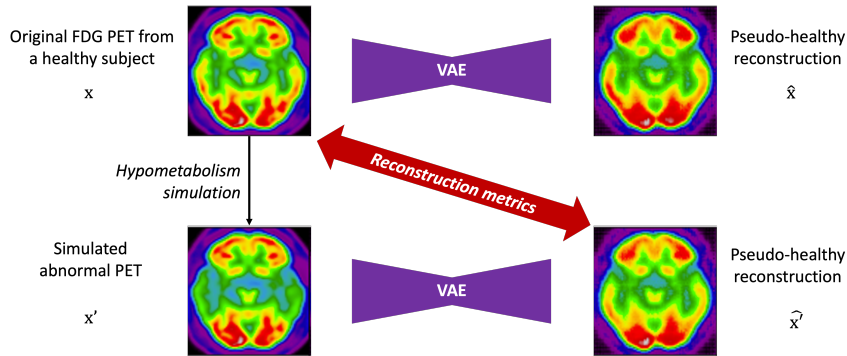


FIGURE 3.2: Evaluation framework using simulated images. We simulate an abnormal PET scan \mathbf{x}' from an image of a healthy subject \mathbf{x} . If the model works perfectly, the reconstruction $\hat{\mathbf{x}}'$ should be identical to the original image \mathbf{x} .

To ensure that the UAD model being evaluated can generalize to dementia other than AD, we also generated masks corresponding to five other dementia subtypes: behavioral variant frontotemporal dementia (bvFTD), logopenic variant primary progressive aphasia (lvPPA), semantic variant PPA (svPPA), non-fluent variant PPA (nfvPPA) and posterior cortical atrophy (PCA) based on the regions defined by Burgos et al., 2021b. All the details about the selected regions are available in Table 3.1 and a pipeline to use the simulation framework has been integrated into the ClinicaDL open-source software¹ (Thibeau-Sutre et al., 2022b).

This framework will help us to extensively evaluate our model on different kinds of anomalies (different shapes, locations and intensities) and allow us to define a new metric to assess the healthiness of reconstructed images.

3.2.3 Measuring the healthiness of reconstructed images

Now that we have pairs of healthy and abnormal images, we want to define a metric that would help evaluate if the model is able to reconstruct images that are looking healthy. We call this metric "healthiness" and denote it as H . We can define H as follows:

$$H = \frac{\mu_M}{\mu_{\bar{M}}} , \quad (3.6)$$

with μ_M the average uptake in the region of the mask M used to simulate the anomaly and $\mu_{\bar{M}}$ the average uptake of voxels in the brain excluding the mask M .

This metric compares the average uptake in the region in which we simulate the disease and the other regions of the brain. For an image from a healthy subject \mathbf{x} , the average uptake in M is similar to the one in \bar{M} , so H will be close to 1. However, for a simulated image \mathbf{x}' , as the intensity is decreased within the mask M , H will be lower than one. We then have to measure if the healthiness of the pseudo-healthy reconstruction $\hat{\mathbf{x}}'$ is similar to the one of the original image \mathbf{x} (close to 1), or at least superior to that of the input \mathbf{x}' . This can also be used to measure healthiness for hyper-intense anomalies: the score of the input image would then be above 1 and, similarly to hypometabolism detection, the reconstruction's score should be around 1 (or at least lower than the input image healthiness).

¹<https://clinicadl.readthedocs.io/en/latest/Preprocessing/Generate>

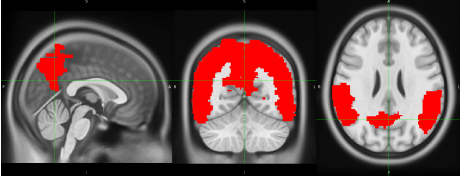
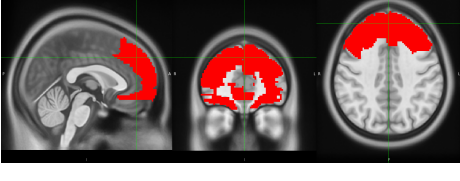

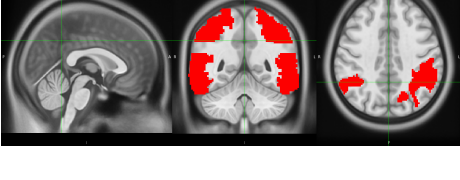
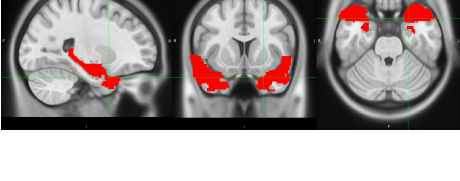
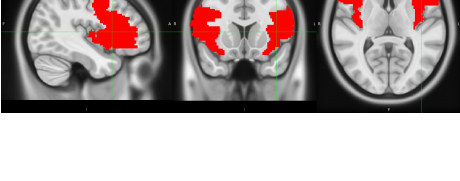
Dementias	Regions Associated	Masks
Alzheimer's disease (AD)	<ul style="list-style-type: none"> • temporal lobe, including the lateral and medial regions and temporal pole • parietal lobe, including the superior and inferior regions 	
Behavioral variant frontotemporal dementia (bvFTD)	<ul style="list-style-type: none"> • orbitofrontal region, comprising the anterior, posterior, medial, and lateral orbital gyri, • dorsolateral prefrontal region, comprising the inferior, middle, and superior frontal gyri • ventromedial prefrontal region, comprising the gyrus rectus, medial frontal cortex, subcallosal area, and superior frontal gyrus medial segment. 	
Logopenic variant primary progressive aphasia (lvPPA)	<ul style="list-style-type: none"> • tempo-parietal region, comprising the inferior parietal lobule, posterior middle and superior temporal gyri. 	
Semantic variant primary progressive aphasia (svPPA)	<ul style="list-style-type: none"> • anterior temporal region, comprising the hippocampus, amygdala and temporal pole. 	
Non-fluent variant primary progressive aphasia (nfvPPA)	<ul style="list-style-type: none"> • frontal region, comprising the inferior frontal gyrus, precentral gyrus and anterior insula. 	
Posterior cortical atrophy (PCA)	<ul style="list-style-type: none"> • occipital region, comprising the inferior, middle and superior occipital gyri. 	

TABLE 3.1: Regions associated with different dementia as defined in Burgos et al., 2021b and the masks used for hypo-metabolism simulation.

This simple metric will allow us to evaluate the performance of the model for reconstructing healthy images, but as it uses the framework described in Section 3.2.2, it cannot then be used on real images since we do not have the a priori information on the location of anomalies. This is why we introduce another method to validate the anomaly detection using the pseudo-healthy reconstruction.

3.2.4 Anomaly detection and localization

The goal of this other method is to localize and assess the severity of anomalies in real images of patients by comparing them to their pseudo-healthy reconstruction. The idea is similar to the healthiness metric, but we cannot use an anomaly mask to compute the metric. Instead, we use a brain atlas to define regions in which we compare the average uptake between the input image \mathbf{x} and the pseudo-healthy reconstruction $\hat{\mathbf{x}}'$. This region-wise anomaly score allows us to assess if an image contains anomalies and if so, to localize them.

This method can be validated using the evaluation framework described in Section 3.2.2 as we know where the anomalies are.

We define the regions that we use starting from that of the second automated anatomical labelling (AAL2) atlas (Rolls et al., 2015). To simplify the analysis, we merged the 120 regions into 23 regions: orbitofrontal, dorsolateral prefrontal (DLPFC), ventromedial prefrontal (VMPFC), motor, opercular, medial temporal, lateral temporal, temporal pole, sensory, medial occipital, lateral occipital, medial parietal, lateral parietal anterior cingulate gyrus, middle cingulate gyrus, posterior cingulate gyrus, midbrain, amygdala, thalamus, insula, hippocampus, cerebellum and cerebellar vermis. We refine these regions using the gray matter mask of the MNI ICBM 2009c Nonlinear Symmetric template (Fonov et al., 2009; Fonov et al., 2011) to keep only the tracer uptake in the gray matter.

Note that both metrics have different objectives: the first one is used to evaluate the model ability to reconstruct pseudo-healthy images, and the second one is a metric for the anomaly detection task.

3.3 Results

In this section, we present the results obtained when applying the proposed validation procedure to the pseudo-healthy reconstruction of 3D FDG PET images with a VAE for the detection of anomalies characteristic of Alzheimer’s disease and other dementias. As a reminder, the validation procedure consists of four steps:

- computing reconstruction metrics for images of healthy subjects to evaluate the quality of the reconstruction (results presented in Chapter 2, Section 2.3.2);
- measuring the healthiness of the reconstructed pseudo-healthy images using the simulation framework by comparing the pseudo-healthy reconstruction $\hat{\mathbf{x}}'$ to the original scan from a healthy subject \mathbf{x} , and also by computing the newly introduced healthiness metric H ;

- detecting regions of the brain containing anomalies using an atlas and validating this approach with the simulation framework;
- detecting anomalies on a real dataset of patients diagnosed with AD.

The results are presented for the model trained on split 1, because it is difficult to interpret the results across several folds (see Section 2.3.2).

3.3.1 Evaluation of the model using the simulation framework

Results on simulated AD-like FDG PET images

To evaluate the impact of the anomaly severity on the ability of the VAE to reconstruct pseudo-healthy images, we simulated different degrees of hypometabolism from 5% to 70% (Figure 3.3). We first remark that the reconstruction is satisfying until around 20% of simulated hypometabolism as the MSE between the reconstruction $\hat{\mathbf{x}}'$ and the ground truth \mathbf{x} is almost constant.

Figure 3.3 also shows that the MSE between the simulated input image \mathbf{x}' and the output $\hat{\mathbf{x}}'$ is higher for more severe anomalies. This confirms that the model cannot reconstruct well highly abnormal areas.

We then compared the reconstruction error that was obtained for the simulated data (i.e., $MSE(\mathbf{x}', \hat{\mathbf{x}}')$, in orange in Figure 3.3) with the error that exists between the real healthy images from the CN test set and the reconstructions obtained from the simulated data (i.e., $MSE(\mathbf{x}, \hat{\mathbf{x}}')$, in blue in Figure 3.3). We remark that $MSE(\mathbf{x}', \hat{\mathbf{x}}')$ does not increase as much as $MSE(\mathbf{x}, \hat{\mathbf{x}}')$ with the hypometabolism severity. This means that the reconstruction $\hat{\mathbf{x}}'$ is more similar to the ground truth \mathbf{x} than the abnormal simulated image \mathbf{x}' . However, the error still increases between the reconstruction $\hat{\mathbf{x}}'$ and the ground truth \mathbf{x} , meaning that a healthy image cannot be totally recovered when the anomalies are too intense. We also observe that for low degree hypometabolism (<20%), both MSEs are similar. This means that the residual error due to the model imperfect reconstruction dissimulates the reconstruction error due to low degree anomalies. To confirm our observations, we computed a t-test assessing whether there was a significant difference in MSE between the reconstruction from the simulated input $\hat{\mathbf{x}}'$ and the ground truth \mathbf{x} using images with various degrees of anomalies. The p-values were corrected for multiple comparisons using the Bonferroni method with 10 comparisons. The difference in MSE becomes significant (p-value<0.005) for anomalies of degree 20% and above. This shows that we can detect hypometabolism around 25% using the residual error, which corresponds to the average difference in metabolism between CN subjects and AD patients in a region of interest relevant to AD in ADNI (Landau et al., 2012), knowing that this dataset includes patients at a very early stage of the disease.

Figure 3.4 displays the real image of a CN subject \mathbf{x} and its pseudo-healthy reconstruction $\hat{\mathbf{x}}$, as well as the simulated AD version \mathbf{x}' (with a hypometabolism degree of 30%) and its reconstruction $\hat{\mathbf{x}}'$ obtained from the same CN subject, together with the residual images. We observe that the input and output images of the CN subject are quite similar, both the shape of the brain and the uptake distribution look alike. The differences are due to the model imperfect reconstruction and correspond to the minimal error that it can achieve.

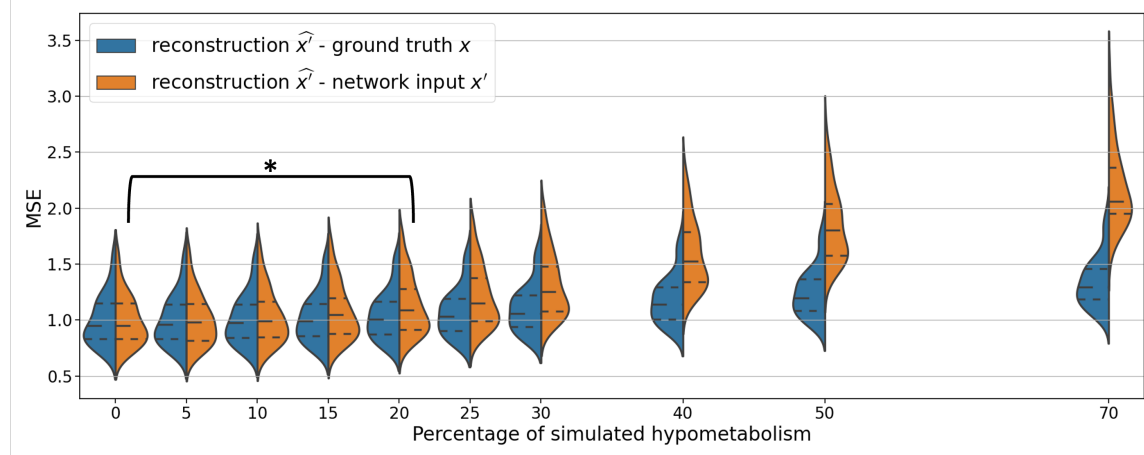


FIGURE 3.3: Evolution of the MSE with increasing degrees of hypometabolism simulating AD-like anomalies. We plot the distribution of the MSE between the pseudo-healthy reconstruction and the original image $MSE(\mathbf{x}, \hat{\mathbf{x}}')$ blue, and the MSE between the pseudo-healthy reconstruction and the simulated data $MSE(\mathbf{x}', \hat{\mathbf{x}}')$ orange. Each MSE is normalized by the average MSE obtained when reconstructing from the original healthy images.

When feeding the simulated hypometabolic image \mathbf{x}' to the model, we observe that the reconstructed image $\hat{\mathbf{x}}'$ looks healthier than the input image. The areas highlighted in blue in the residual map correspond to the regions where hypometabolism was simulated.

Another interesting point is that both images reconstructed from the same CN subject $\hat{\mathbf{x}}$ and $\hat{\mathbf{x}}'$ are almost identical, with an SSIM of 0.987. This shows that the model reconstructs almost the same image for the same subject, whether the input image is healthy or presents anomalies.

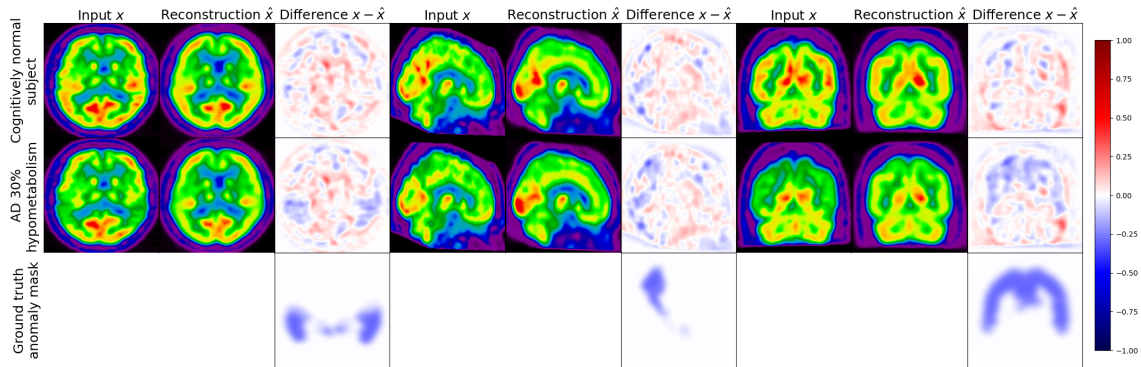


FIGURE 3.4: Example of results obtained from a real image of a CN subject (top row) and an image simulating AD hypometabolism based on the same CN subject (bottom row). For each plane, the first image is the input, the second one the model’s reconstruction and the third one the difference (input - reconstruction). More examples of reconstructions are available in Appendix C.

Results when simulating various types of dementia

In this section, the degree of hypometabolism is set to 30% but the brain region where it is simulated changes to reflect various types of dementia. We report in Table 3.2 the different reconstruction metrics computed between the original images from CN subjects in

the test set \mathbf{x} and the images reconstructed from the hypometabolic scans simulating the different types of dementia $\hat{\mathbf{x}}'$. We observe that the metrics are similar for all the simulated dementias, which means that the model can generalize to anomalies with different locations and shapes, as well as different severity degrees, as we showed previously.

TABLE 3.2: Reconstruction metrics computed between the original healthy PET scans from CN subjects in the test set and the images reconstructed with the 3D VAE from the hypometabolic scans simulating different types of dementia.

Simulated dementia	MSE ($\times 10^{-3}$) \downarrow	PSNR \uparrow	SSIM \uparrow	MS-SSIM \uparrow
AD	2.230 ± 0.655	26.646 ± 0.996	0.848 ± 0.050	0.938 ± 0.015
bvFTD	2.268 ± 0.686	26.584 ± 1.042	0.849 ± 0.051	0.940 ± 0.015
PCA	2.090 ± 0.698	26.962 ± 1.119	0.851 ± 0.051	0.942 ± 0.015
lvPPA	2.073 ± 0.680	26.992 ± 1.104	0.850 ± 0.051	0.941 ± 0.015
nfvPPA	2.093 ± 0.692	26.953 ± 1.107	0.851 ± 0.051	0.941 ± 0.015
svPPA	2.029 ± 0.703	27.101 ± 1.150	0.852 ± 0.051	0.942 ± 0.016

We also computed the metrics between the images reconstructed from the original healthy scans $\hat{\mathbf{x}}$ and the images reconstructed from the simulated hypometabolic scans $\hat{\mathbf{x}}'$ in Table 3.3. Both reconstructions are almost identical, with an SSIM on average superior to 0.99. We can conclude from this experiment that the model is able to reconstruct the healthy version of an image independently of the nature of the dementia that causes the anomaly.

TABLE 3.3: Structural similarity between the pseudo-healthy reconstruction $\hat{\mathbf{x}}'$ and the reconstruction from the healthy image $\hat{\mathbf{x}}$ for the different dementias simulated.

Simulated dementia	SSIM \uparrow
AD	0.9878 ± 0.0014
bvFTD	0.9921 ± 0.0013
PCA	0.9974 ± 0.0003
lvPPA	0.9937 ± 0.0008
nfvPPA	0.9964 ± 0.0005
svPPA	0.9995 ± 0.0002

Measuring healthiness of a pseudo-healthy reconstruction

We computed the proposed healthiness metric for the different simulation experiments: on the test sets simulating AD with various intensity degrees and on the test sets simulating the different dementia subtypes.

We can see in Figure 3.5 that the healthiness score for the original PET scans from CN subjects \mathbf{x} ranges between 0.99 and 1.08, which can define a baseline of what we can consider as healthy with this metric. As expected, we observe that the healthiness of simulated images

\mathbf{x}' is lower than that of the original image \mathbf{x} . At 5% of simulated hypometabolism, the score is still between 0.97 and 1.06, so it can be considered as healthy, which is coherent for very low anomaly severity. From 15% of simulated hypometabolism, we can clearly see that the healthiness score drops and become much lower than that of the healthy images: it is inferior to 1.0 for 15% of simulated hypometabolism, and it is between 0.82 and 0.91 for 30% of simulated hypometabolism. The important point is that the healthiness score of the reconstruction $\hat{\mathbf{x}}'$ is always superior to the one of the simulated image \mathbf{x}' . We can see that it is even really close to the healthiness of the original image \mathbf{x} : for 30% of simulated hypometabolism the healthiness of reconstructed images is between 0.95 and 1.03.

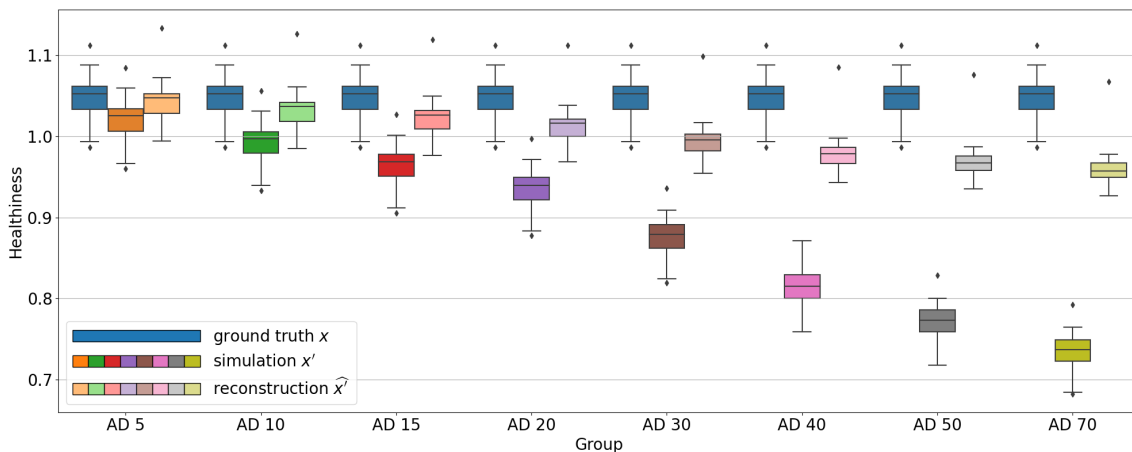


FIGURE 3.5: Evolution of the distribution of the healthiness metric computed for the ground truth healthy images, their corresponding simulated images and their pseudo-healthy reconstructions when increasing the percentage of AD-like simulated hypometabolism.

We observe the same behavior for all the other simulated dementias in Figure 3.6. However, we note that the healthiness of the ground truth, i.e. that obtained for images of CN subjects, varies depending on the dementia simulated because the mask used to compute it differs. For example, in the case of svPPA, the healthiness of the ground truth is lower than that of AD (between 0.67 and 0.92). This can be explained by the fact that the mask used for svPPA is located in the temporal pole, where FDG uptake is lower compared to other regions, even on healthy images, as we can see in Figure 3.7. However, we can still observe that the healthiness is lower on the simulated image compared to the healthy image and almost equal on the reconstruction.

This method relies on the simulation framework and can only be used to evaluate the model performance. However, we would like a method or metric that allows clinicians to know if an image presents anomalies, and possibly localize them.

Anomaly detection applied to simulated data

To detect anomalies in real images of patients, i.e., without having to rely on the simulation framework, we proposed to compute the mean uptake in regions of an atlas and compared the values between input and output. If the value in the reconstructed image $\hat{\mathbf{x}}$ is close to the input image \mathbf{x} , then the region is not likely to be abnormal, otherwise, if the regional

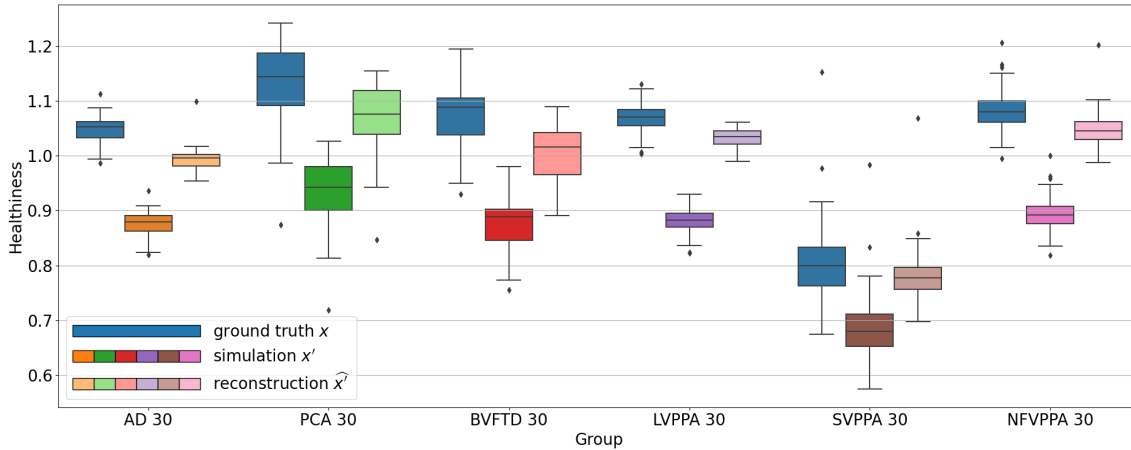


FIGURE 3.6: Evolution of the distribution of the healthiness metric computed for the ground truth healthy images, their corresponding the simulated images and their pseudo-healthy reconstructions for different dementias simulated at 30%.

uptake in the reconstructed image $\hat{\mathbf{x}}$ is significantly different from the one in the original image \mathbf{x} , then there may be an anomaly. We validated this assumption using the simulation framework.

In Figure 3.7, we plotted the average uptake in the regions of the atlas we used and compared these values between the original image \mathbf{x} , the simulated one \mathbf{x}' and the reconstruction $\hat{\mathbf{x}}$ using a Wilcoxon-Mann-Whitney test corrected with Bonferroni for multiple comparisons using the `statannotations` package². First, we remark that the average uptake is not consistent between all the regions of the brain, so we cannot really compute a shift from an average value for the whole brain, but we have to analyze the average uptake for every region. We can then observe that the average uptake is not significantly different between the original image \mathbf{x} and the simulated one \mathbf{x}' , the original image \mathbf{x} and the reconstruction $\hat{\mathbf{x}}$, and the simulation \mathbf{x}' and its reconstruction $\hat{\mathbf{x}}$ for most of the regions, except the hippocampus, the amygdala, the parietal lobe and the temporal lobe. These regions correspond to the regions used to simulate anomalies corresponding to AD. We can see that the average uptake is lower on the simulated image \mathbf{x}' compared to the original image \mathbf{x} . This is expected as the hypometabolism simulation consists in lowering the intensity in those regions. We can also see that the average uptake is significantly higher in these regions compared to the average uptake on the simulated images. Without a priori knowledge on the nature of the anomaly we want to detect, we can see that on this test set, it is likely to be abnormal in these regions. This corroborates with the regions we actually used to simulate hypometabolism.

Now that we extensively used the simulation framework to validate our model on different aspects: pseudo-healthy reconstruction, anomaly detection, generalization to anomalies of different intensities, locations and shapes, we will examine the results on the images of AD patients from the ADNI database.

²<https://statannotations.readthedocs.io/en/latest/index.html>

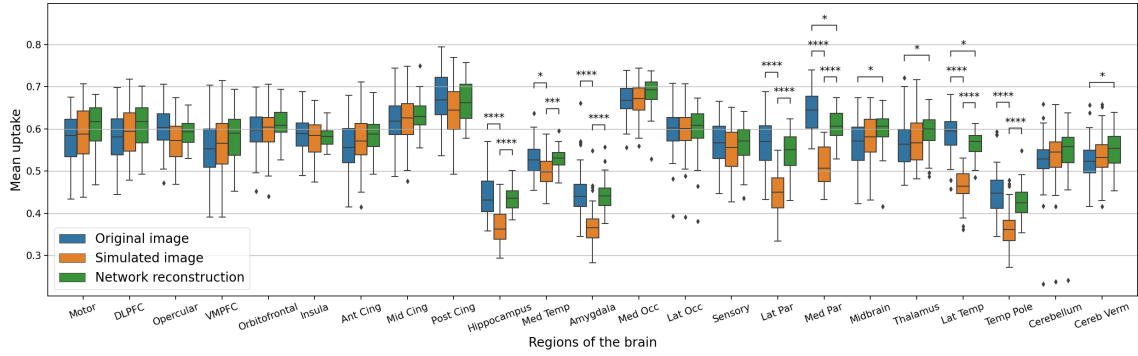


FIGURE 3.7: Distribution of the mean FDG PET uptake in different regions of the brain: comparison between the CN subjects from the test set, their AD-like hypometabolic simulation, and their pseudo-healthy reconstruction.

3.3.2 Results on AD patients from the ADNI dataset

The results of the anomaly detection method applied to real AD patients are reported in Figure 3.8. We can observe that in general, the average uptake is higher in the pseudo-healthy reconstruction. We cannot really detect abnormal areas even though we can see that regions such as the posterior cingulate, hippocampus, parietal lobe, lateral temporal lobe seem to be regions with the largest differences between the AD patients and their pseudo-healthy reconstruction. This global analysis can help us describe the cohort and understand at the population level the shift from the CN population. However, it is not really an image-level anomaly detection tool. For that, we need to observe individually each image. Some examples of reconstructions from AD patients are displayed in Chapter 2, Section 2.3.2, Figure 2.8.

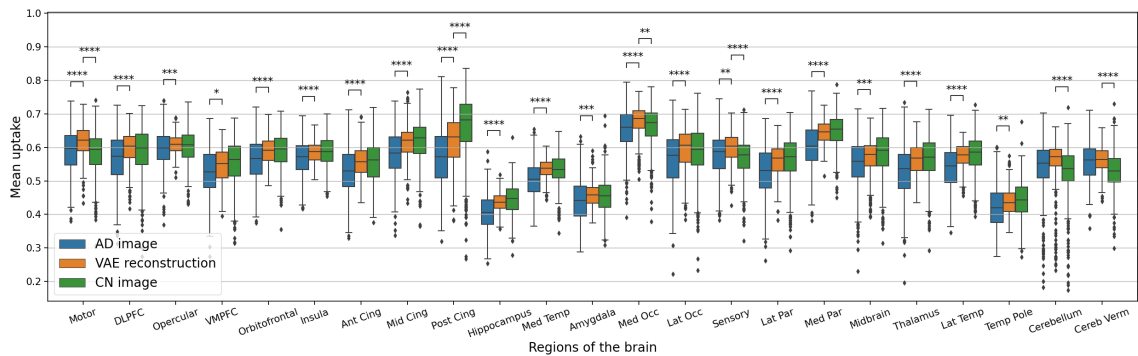


FIGURE 3.8: Distribution of the mean FDG PET uptake in different regions of the brain: comparison between the original image from AD patients, their pseudo-healthy reconstruction and the CN population.

3.4 Comparison between VAE and Unet

To demonstrate the value of the evaluation procedure, we trained an Unet with an architecture very similar to the VAE's one (we added skip connections and removed the probabilistic part of the latent space). The expected result of this experiment is that the model should be able to reconstruct good quality images by learning the identity function.

Indeed, the model learning to reconstruct its inputs, it should probably use higher level skip-connections to minimize the reconstruction error. However, when trying to reconstruct images with anomalies, the model should also reconstruct an image similar to the input, instead of a pseudo-healthy version, as it did not learn any data distribution, but just an identity function.

We can observe in Table 3.4 that the reconstruction of the Unet is almost identical to the input image, with an SSIM of 0.99 on average. Compared to the VAE (Table 2.3), the MSE is almost 100 times lower with the Unet. This means that the Unet is able to reconstruct images of high quality. However, we can see that the reconstructions are also similar to the input when the input is a simulated abnormal image, meaning that the model probably also reconstructs the anomalies.

TABLE 3.4: Comparison of the reconstruction results obtained for Split 1 between the Unet and VAE.

Model	Dataset	MSE ($\times 10^{-3}$) \downarrow	PSNR \uparrow	SSIM \uparrow	MS-SSIM \uparrow
Unet	Test CN	0.024 ± 0.005	46.281 ± 0.878	0.990 ± 0.001	0.999 ± 0.000
	Test AD	0.028 ± 0.020	45.864 ± 1.555	0.990 ± 0.002	0.999 ± 0.000
	AD 30	0.024 ± 0.006	46.271 ± 0.934	0.990 ± 0.001	0.999 ± 0.000
VAE	Test CN	1.815 ± 0.649	27.572 ± 1.074	0.878 ± 0.026	0.944 ± 0.014
	Test AD	2.554 ± 1.391	26.272 ± 1.560	0.853 ± 0.045	0.928 ± 0.025
	AD 30	2.345 ± 0.639	26.403 ± 0.890	0.869 ± 0.027	0.934 ± 0.015

This can be verified by observing directly the images in Figure 3.9. We can see that the reconstruction is identical to the input, and the difference is null. We thus cannot detect anomalies.

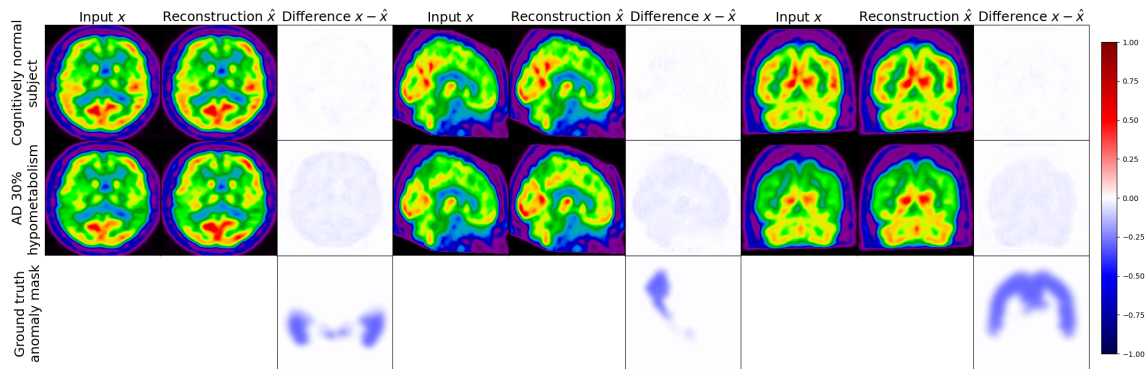


FIGURE 3.9: Example of results obtained with the Unet from a real image of a CN subject (top row) and an image simulating AD hypometabolism based on the same CN subject (bottom row). For each plane, the first image is the input, the second one the model’s reconstruction and the third one the difference (input - reconstruction).

This is confirmed when computing the healthiness metric defined in Section 3.2.3. Even though we can see from the reconstruction metrics that the model is not able to reconstruct pseudo-healthy FDG PET images, this is a limit case. In a more realistic scenario, reconstruction metrics and visual assessment of the images are not enough to estimate if a model is able to perform well. In Figure 3.10a, we can see that the healthiness of the reconstructed

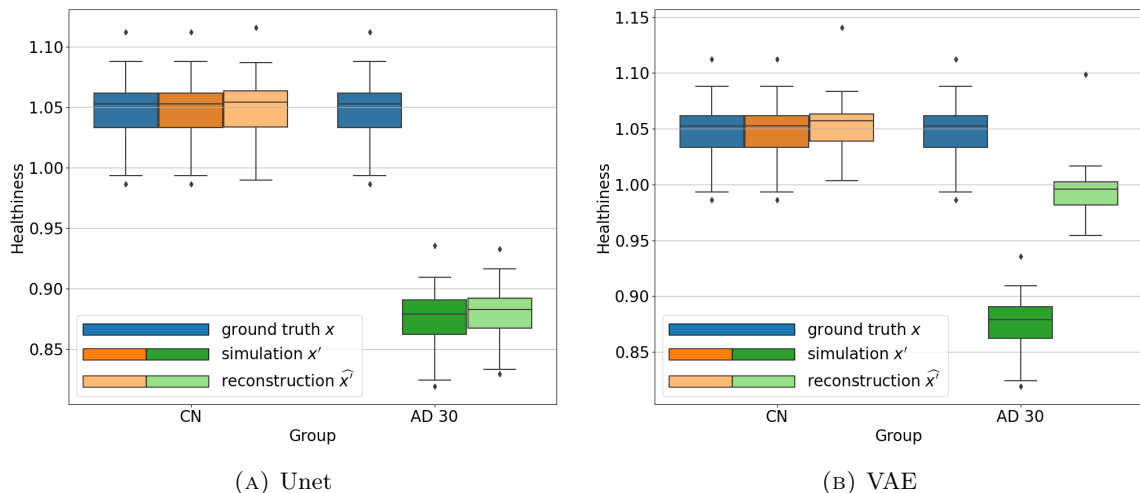


FIGURE 3.10: Comparison of the distribution of the healthiness metric between the Unet and VAE for both CN and simulated AD subjects. For both models, the healthiness is constant (between 0.98 and 1.09) for images from healthy subjects of the test set (in blue), and the reconstruction is healthy as expected (in orange). For simulated images (AD 30), we can see that the healthiness is between 0.83 and 0.91 (dark green). However, the healthiness of the Unet reconstruction (light green) is the same (between 0.84 and 0.92), meaning that the reconstruction cannot be considered as pseudo-healthy. On the other hand, the healthiness of the VAE reconstruction, between 0.96 and 1.02, is higher than for the simulated image given in input. The VAE reconstruction can thus be considered as healthy.

image is equal to the one of the input image, meaning that the model could not reconstruct pseudo-healthy images.

3.5 Discussion

In this chapter, we proposed an in-depth validation procedure for pseudo-healthy synthesis with deep generative models in the context of UAD. This evaluation method relies on a simulation framework and is suited for applications for which ground truths are not available to measure the performance of the model. This procedure helps to extensively test a model on different aspects: the quality of the reconstruction, the healthiness of the reconstructed images, and the possibility to detect anomalies on both simulated and real data. We applied this procedure to evaluate the ability of a 3D VAE to detect anomalies related to Alzheimer’s disease on 3D brain FDG PET data from the ADNI database.

To overcome the absence of ground truth anomaly masks for the evaluation of the model, we introduced a framework to simulate different kinds of dementias from images of healthy subjects. Having such pairs of diseased and healthy images allowed us to measure the reconstruction error between the pseudo-healthy reconstruction and the original image from the healthy subject. We showed that the pseudo-healthy reconstruction is more similar to the original image from the healthy subject than from the hypometabolic simulated image (Figure 3.3). Moreover, in the case of AD, the typical variation of metabolism in relevant regions is around 25% (Landau et al., 2014), which corresponds to the intensity degree of anomalies that we can detect using the reconstruction error (Figure 3.3).

We also showed that the reconstructed images are looking healthy by introducing a new healthiness metric, which we validated thanks to the simulation framework. This analysis showed that the relative average uptake in the region used to simulate hypometabolism compared to the other regions of the brain is higher on the reconstruction, which means that the VAE can reconstruct a pseudo-healthy image (Figure 3.5 and Figure 3.6). Actually, if the simulated hypometabolism is reasonable ($<30\%$), the healthiness of the reconstructed pseudo-healthy images is similar to that obtained for the original image from a healthy subject. We also simulated dementias other than AD and showed that the VAE was indeed able to generalize to anomalies in different parts of the brain. This is an important point as many diseases are rare, so it is impossible to detect them using a traditional supervised machine learning approach due to lack of data.

We do not only rely on the simulated data to estimate the performance of the model, but we also use the image from AD patients to test the model in a more realistic context. Using the anomaly metric, we see that the pseudo-healthy reconstructions of images from real AD patients seem to have an average uptake similar to the healthy population (Figure 3.8). Unfortunately, at this stage, the individual analysis remains only visual and not quantitative since there is no ground truth healthy image for these patients, nor lesion masks.

The main advantage of using the proposed simulation framework is the possibility of quantitatively measuring the performance of the model using metrics (reconstruction and healthiness). This is crucial for further evaluation when one needs to compare models. To illustrate this, we trained a Unet model with a similar architecture to the one of the VAE. This experiment highlights the importance of not only relying on reconstruction metrics and observations, and demonstrates that simulated data can be useful to identify models that are not suited for pseudo-healthy reconstruction.

It also allows testing the model in many different conditions, with various kinds of anomalies, and validates the fact that the model can generalize well. Another benefit is that this may lead to more robust anomaly detection by the clinician, as it may be difficult to be vigilant on the whole image when manually inspecting a 3D scan.

One weakness of using simulated data is that they might not be very realistic. However, we can clearly see that the simulated images are abnormal, which still allows evaluating performance, even though they are not totally realistic. For an even more comprehensive assessment, we may consider simulating a broader variety of anomaly types. Specifically, simulating non-symmetric or non-uniform anomalies could better capture the heterogeneity observed in dementia. Additionally, simulating smaller anomalies, which may challenge the detection using a VAE, would further enrich the evaluation.

Another potential limitation of the proposed evaluation framework is that the evaluation only relies on the difference between input and reconstructed images, i.e., the residual. The use of an anomaly mask that could be compared with the ground truth using a metric such as the dice score could be a great improvement. A simple solution may be thresholding the difference maps, or use Z-scores to attenuate the reconstruction noise and accentuate the anomaly (Solal et al., 2024a). In addition, the only parameter for the simulation of anomalies is the degree of hypometabolism. However, establishing a correlation between this parameter and the progression or severity of the disease, as measured by cognitive

scores (such as the MMSE or CDR), or the time elapsed before the first symptoms, poses a challenge. In other words, interpreting the intensity of the simulated anomalies in relation to the patient’s cognitive status is not straightforward.

The proposed validation procedure is applicable outside the use case presented here. Most of the code that we used is available in ClinicaDL (Thibeau-Sutre et al., 2022b), an open-source software that is developed for reproducibility of deep learning studies in neuroimaging. Pipelines are available to perform the following steps:

- selecting subjects from a neuroimaging dataset,
- rigorously separating data into independent training, validation and testing sets,
- easily training a VAE on neuroimages,
- constructing new test sets by generating simulated data using the proposed method,
- running tests to evaluate models.

Moreover, all the preprocessing pipelines are also available in Clinica (Routier et al., 2021), an open-source software for reproducible processing of neuroimaging datasets and multi-modal neuroscience studies. Clinica has been used to:

- curate and organize the ADNI dataset following a community standard, namely the brain imaging data structure (BIDS) (Gorgolewski et al., 2016),
- perform linear registration and intensity normalization of the FDG PET scans.

Finally, all the code for training and evaluating the model is available on a Github repository: https://github.com/rav1h18/UAD_evaluation_framework; and is tagged on Zenodo under the following DOI: <https://zenodo.org/doi/10.5281/zenodo.10568859>.

3.6 Conclusion

In this chapter, we presented an extensive evaluation procedure of pseudo-healthy reconstruction for unsupervised anomaly detection in the case where ground truths are not available. It consists in different steps that are: the measurement of the reconstruction error on images from healthy subjects, the use of a simulation framework to create pairs of healthy and diseased images; the introduction of a metric to measure the healthiness of images when using the simulation framework; the use of a brain atlas to detect anomalies by comparing the input and the reconstructed images using the simulation framework and the real pathological images from ADNI dataset. The procedure is summarized in Figure 3.11.

This procedure has been applied to a 3D VAE that is suited to detect anomalies due to dementia on brain FDG PET. The VAE has been trained to reconstruct healthy-looking images using images of healthy subjects. We saw that the model can indeed reconstruct subject-specific pseudo-healthy images and can help to detect anomalies. We also validated the model ability to detect anomalies of different intensities, shapes and locations. However, the performance could be increased by improving the quality of the reconstruction.

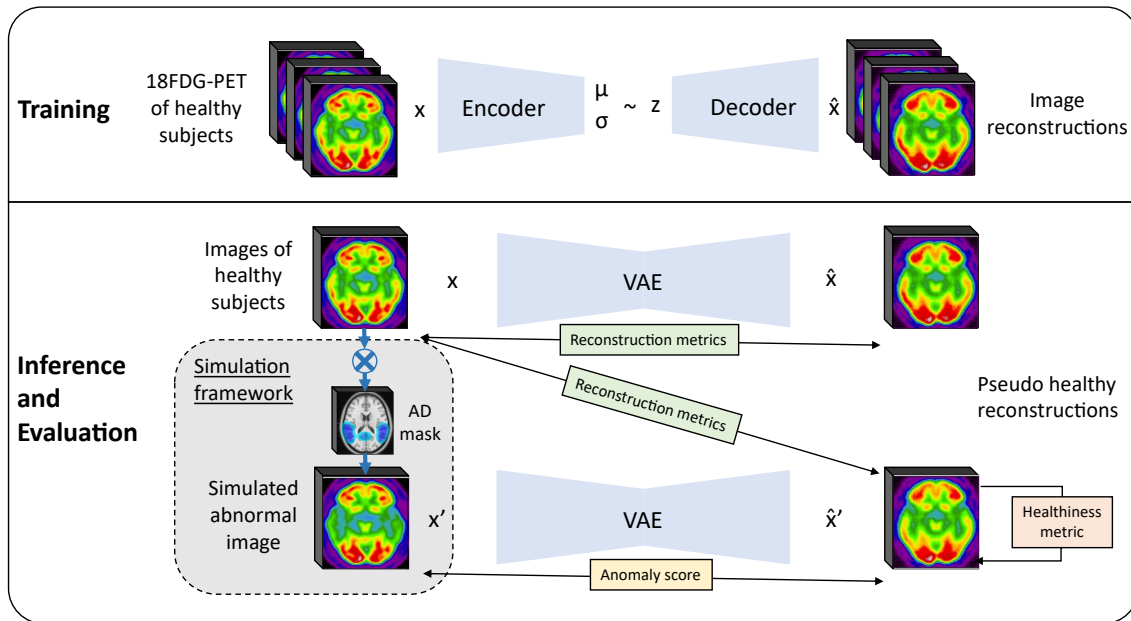


FIGURE 3.11: Schema summarizing proposed evaluation framework. The VAE is trained only with FDG PET of healthy subjects. Then, FDG PET scan of healthy subjects from the test set are used to assess the reconstruction performance of the model, and also to build new test sets with simulated hypometabolic scans. The simulated data are finally used to measure the ability of the VAE to reconstruct pseudo healthy images thanks to the introduced healthiness metric and anomaly score.

In order to benefit from both the evaluation framework and the VAE regularized latent space, we will analyze in the following chapter the latent representation of simulated data. Our aim is to provide a tool for interpreting the outcomes of the generative model.

Chapter 4

Study on the VAE latent space

This chapter is a part of an article published in the Special Issue for Generative Models of Machine Learning for Biomedical Imaging.

- **Title:** Evaluation of pseudo-healthy image reconstruction for anomaly detection with deep generative models: Application to brain FDG PET
- **Authors:** Ravi Hassanaly, Camille Brianceau, Maëlys Solal, Olivier Colliot and Ninon Burgos
- **DOI:** [10.59275/j.melba.2024-b87a](https://doi.org/10.59275/j.melba.2024-b87a)

Now that we extensively tested the VAE under various conditions in Chapter 3 thanks to the simulation framework, we would like to better understand its behavior and interpret the results. One of the main advantages of the VAE over other generative models is its consistent latent space that we use for this purpose.

In this chapter, we will present the results of a set of experiments on the latent space to verify that the VAE can learn the healthy image distribution, given that there are only images of healthy subjects in the training set. Indeed, for instance, we would like to understand why the reconstruction $\hat{\mathbf{x}}'$ obtained from a simulated abnormal image \mathbf{x}' looks similar to the reconstruction $\hat{\mathbf{x}}$ obtained from the original input image \mathbf{x} . Actually, there is no reason that would a priori explain why the reconstruction of an abnormal image would be realistic and correspond to a healthier version of the input image. We could imagine that, like for out-of-distribution detection methods, the model would not reconstruct the input image at all. This is why we will use the latent space representation to study our model and understand what the VAE learns. In the latent space, all the input images are projected into a one dimension vector space of size 256 through the encoder. The advantage of the VAE is that the latent space is consistent, that is to say that, in theory, the latent representation of the images are organized with respect to the image distribution. We will verify whether this is actually the case.

4.1 Latent space visualization

We first visualize the latent space using a principal component analysis (PCA) to reduce the latent dimension from 256 to 2. We fit the PCA on the latent representation of healthy

images from the training set, as we can see in Figure 4.1a. Even if we plot only the first two principal components, this already indicates how the encoder behaves. This will help us to verify whether the learned posterior is the same for healthy and abnormal images, i.e., whether $q_\phi(\mathbf{z} | \mathbf{x}_h) \approx q_\phi(\mathbf{z} | \mathbf{x}_p)$.

We then predict the principal components of the latent representation of images from the CN test set with the same PCA, as well as their hypometabolic version simulating AD. A remarkable point is that the projection is almost the same for images that have been simulated from this test set, as shown with the paired points in Figure 4.1b. This explains why their reconstruction are almost identical, as we noticed in section 3.3.1. Indeed, the decoder will reconstruct two similar images from two similar latent vectors.

We also project latent vectors of images from the AD test set, and we can see that the points are in the same area of the latent space (Figure 4.1a). This validates our hypothesis that images presenting anomalies (real or simulated) are projected into the healthy images' latent distribution that was learned on the training set. We can observe that in practice $q_\phi(\mathbf{z} | \mathbf{x}_h) \approx q_\phi(\mathbf{z} | \mathbf{x}_p)$ and that the latent representation \mathbf{z} is a small sphere in the latent space.

Another interesting point is that the latent space seems to capture the simulated progression of AD. We observe in Figure 4.1d that the principal component vectors of AD simulated images are aligned in the latent space, near the original image latent representation, and ordered by severity.

4.2 Learning the data distribution

We can also verify that if two images are close in the image space, there are close in the latent space and vice versa. We compare the distance between images in both latent- and image-space. More precisely, for each image latent vector z_i of the dataset, we compute the Minkowski distance with the latent vectors z_j of all the other images

$$D_{Minkowski} = \left(\sum_{i=1}^n |z_i - z_j|^p \right)^{\frac{1}{p}}. \quad (4.1)$$

We arbitrarily choose $p = 10$ in all our applications as we wanted p to be high enough to compute distance in a space of dimension 256.

4.2.1 Intra- vs inter-subject distance

When studying dementia, datasets are often longitudinal, which means that several images are available per subject. This allows us to first evaluate an intra-subject distance that is computed between a certain image of a subject and all the other images available for the same subject. We can also compute an inter-subject distance.

We used the Minkowski distance to compute intra-subject and inter-subject distances in the latent space. We first observe that, for a given image of a subject, all the closest images in the latent space are images of the same subject, acquired during other visits. Figure 4.2 displays the box plots of the mean distance between the latent representation of an image

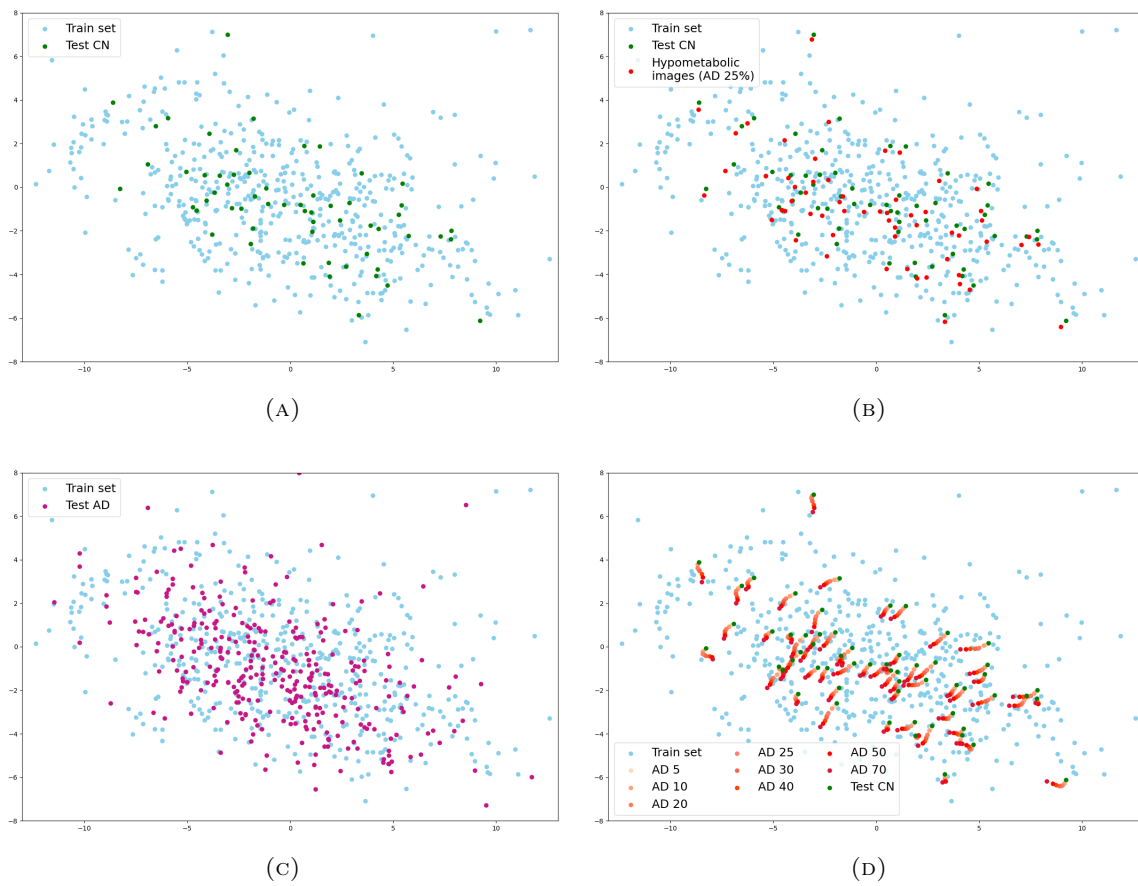


FIGURE 4.1: Latent space representation (first two PCA components). The latent distribution of the train set learned by the VAE (in blue) is compared to the latent distribution inferred on the test set with CN subjects (a), test set with AD patients (c), and image simulating AD-like hypometabolism with progression from 5% to 70% (b, d).

and that of the other images of the same subject, and the mean distance between an image and the five closest latent representations of images that are not from the same subject. The difference between intra-subject and inter-subject distances is statistically significant (p -value $\ll 0.005$ according to a Mann-Whitney U test). This clearly indicates that all the images from a same participant are very close in the latent space compared to the average distance between two images from different participants.

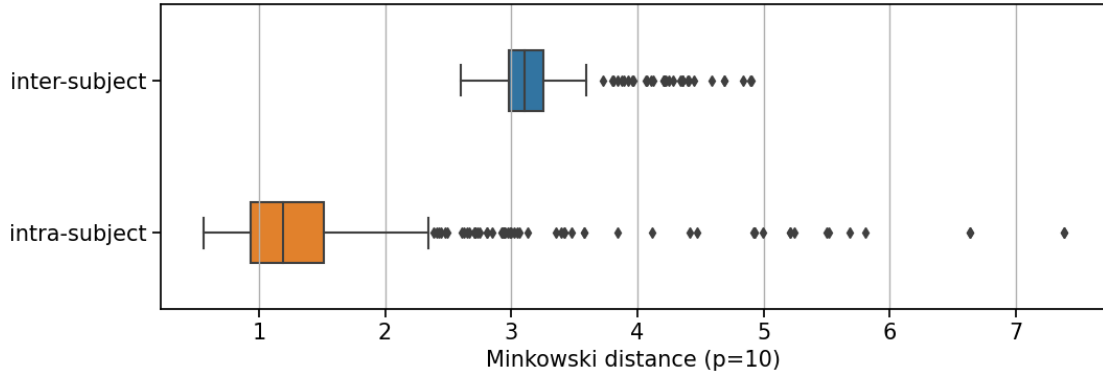


FIGURE 4.2: Box plot showing the distribution of the Minkowski distance computed between the latent representation of images from the same subject (intra-subject) and between the closest latent representation of images from other subjects (inter-subject).

4.2.2 Linear mixed effect models applied to latent representations

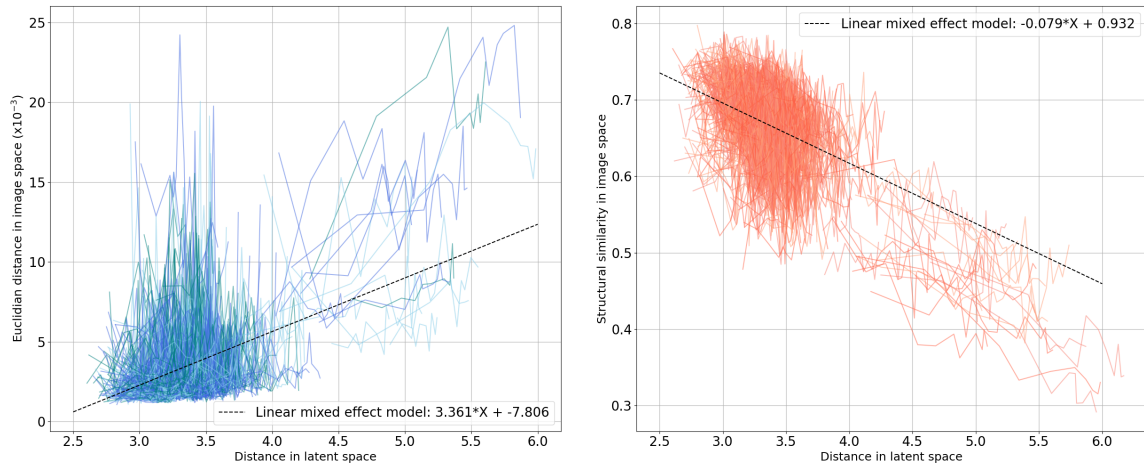
We finally want to check if the model can learn the data distribution, in the sense that, if two images are similar in the image space, are they close in the latent space?

For a certain image of a subject, the closest image from another subject is selected, as well as the tenth, the twentieth, the thirtieth and the fortieth closest images in the latent space (after discarding images from the same subject). We then compute the Euclidean L_2 norm and the SSIM between our image and the collection of five images selected. Computing these distances in both the latent space and the image space allows identifying potential correlation between the two representations. To this end, we fitted a linear mixed effect model (LMM) to estimate the tendency of the evolution of the distance in the latent space with regard to the distance in the image space (or the similarity in the image space).

Linear mixed effect model

LMM is a statistical regression method used to analyze data that are dependent. It is particularly adapted to studies in which several observations are available per subject, such as in longitudinal studies. Here our different observations are the N closest images in the latent space (the tenth, the twentieth, the thirtieth and the fortieth closest images). The equation of the LMM is

$$Y_{ij} = \beta_0 + \beta_1 X_{ij} + \gamma_{0i} + \gamma_{1i} X_{ij} + \epsilon_{ij} ,$$



(A) Distance in latent space (Minkowski) compared to distance in image space (MSE) (B) Distance in latent space (Minkowski) compared to similarity in image space (SSIM)

FIGURE 4.3: Evolution of the MSE and the SSIM compared with the Minkowski distance in the latent space. Each curve represents an image i and comprises ten points. Each point of the curve corresponds to the distance D_{ij} of this image i with the j^{th} closest images in the latent space, j being in $\{1, 6, 11, 16 \dots 41, 46\}$.

where (X_{ij}, Y_{ij}) are respectively the distance in the latent space and the image space of the j^{th} closest subject of subject i , β_0 and β_1 are the population effect parameters, γ_{0i} and γ_{1i} are the individual effect parameters and ϵ_{ij} the residual error. Each subject is modeled by a linear function of intercept $\beta_0 + \gamma_{0i}$ and slope $\beta_1 + \gamma_{1i}$. β_1 is the mean slope of the population and γ_{1i} is the variance of each individual. We can then estimate the tendency of the evolution of the distance in the latent space with regard to the distance in the image space by observing the values of $\beta_1 + \gamma_{1i}$.

Results

To ensure that, if two points are close in the latent space, their corresponding images are close in the image space; and similarly, if they are far in the latent space, their corresponding images are not similar. We plot curves representing the evolution of the MSE and the SSIM with regard to their Minkowski distance in the latent space (Figure 4.3) and fit two linear mixed effects models on both sets of curves to observe the general tendency: one for the MSE in Figure 4.3a and one for the SSIM in Figure 4.3b. We can see that the distance in the latent space increases when the MSE between two images grows and when the SSIM between the two images decreases. In other words, similar images have close latent representations. Detailed results of LLMs are available in Table 4.1.

4.3 Discussion

Choosing a VAE as generative model allowed us to perform analyses in the latent space. In particular, we used it to explain how the VAE behaves and interpret some of the results. We first observed that the latent representation of a same patient is always very close in the latent space, and almost identical between the simulated and the original PET scans (Figure 4.1). More globally, we showed that the VAE encoder is able to map the complex

TABLE 4.1: Result of linear mixed effect models corresponding to Figure 4.3. The top two rows correspond to the model fitted on the MSE against the Minkowski distance in the latent space. The bottom two rows correspond to the model fitted on the SSIM against the Minkowski distance in the latent space. "intercept" correspond to the intercept of the model and "latent" to the slope. "Coef." is the estimation of the value of the intercept or the latent, "Std. Err." is the standard error on this value, z is the z-score of this estimation, $P > |z|$ the p-value associated with this z-score and the last column is the confidence interval of the value.

		Coef.	Std.Err.	z	$P > z $	[0.025 0.975]
MSE ($\times 10^{-3}$)	intercept	-7.806	0.583	-13.397	0.000	-8.948 -6.664
	latent	3.361	0.164	20.459	0.000	3.039 3.683
SSIM	intercept	0.932	0.009	106.935	0.000	0.915 0.949
	latent	-0.079	0.002	-31.688	0.000	-0.084 -0.074

data distribution to a simple multivariate Gaussian distribution of lower dimension. This explains why, for small deviations from the healthy image distribution (anomalies that we simulated), the model is able to reconstruct an image that is plausible (Figure 3.4), that corresponds to the patient under investigation, and seems to be pseudo-healthy. This can be explained by disentangling the functioning of the encoder and the decoder: the encoder catches the image structural information that is specific to the subject, and the decoder, given a latent representation \mathbf{z} , can only reconstruct healthy looking images because it is what it has been trained to do. This is not straightforward, and the model could have other behaviors. Indeed, we identify three different scenarios:

- the model reconstructs the identity, meaning that a healthy image has a healthy reconstruction and an abnormal image is reconstructed with its anomalies,
- the model does not reconstruct abnormal images at all as it has never seen some, which would be a behavior similar to out-of-distribution detection,
- the model reconstructs pseudo-healthy images since it could learn well the healthy image distribution, which seems to be the case.

Combining the latent space analysis with our simulation framework shows that the model has a similar behavior for the different kinds of anomalies (Table 3.2) and that the VAE can generalize well.

Our experiments on the latent space show that the encoder is working as we can expect. To improve the quality of the reconstructed images, a first simple step would be to use a more powerful generator (Duquenne et al., 2022). If given a latent representation \mathbf{z} the model can reconstruct perfectly the image \mathbf{x} , then we could detect anomalies with a high accuracy. We can also imagine that combining a VAE with a diffusion model as done by Pandey et al., 2022 might be a good solution to improve the decoder. It showed great results on 2D images, and future work could consist in comparing and evaluating such approach on a 3D task on all the different aspects we enunciated.

4.3.1 Conclusion

In this chapter, we exploited the latent space properties to understand the VAE behavior and interpret the results. We saw that the model can encode very similar latent representations for different images of a same subject (different sessions, or simulated images), and more generally, that the latent distribution represents well the image distribution. This is the expected behavior for the encoder. That means that if we want to improve the quality of the reconstruction, we would have to use a better decoder.

Now that we extensively tested the vanilla VAE, it would be interesting to try other generative models, and more specifically the numerous VAE variants that have been developed in the computer vision literature to improve the VAE framework (Chadebec et al., 2022), and see if it is possible to find a model that performs better than the baseline VAE.

Chapter 5

Benchmark of VAE-based approaches

This chapter has been submitted to Medical Image Analysis.

- **Title:** Pseudo-healthy image reconstruction with variational autoencoders for anomaly detection: A benchmark on 3D brain FDG PET
- **Authors:** Ravi Hassanaly, Maëlys Solal, Olivier Colliot, Ninon Burgos
- **Contributions:** This was a shared work between Maëlys Solal and me. I have led the study, implemented most of the code and wrote the major part of the manuscript.

Pseudo-healthy reconstruction approaches that have been developed for medical imaging often rely on generative models such as variational autoencoders (VAEs) (Kingma et al., 2014), generative adversarial networks (GANs) (Goodfellow et al., 2014) and more recently diffusion models (Ho et al., 2020). Even though diffusion models have shown remarkable performance for image generation, they do not easily scale to 3D images (Graham et al., 2023), mainly because of memory issues. On the GAN side, after the foundational work of Schlegl et al., 2017, AnoGAN and f-AnoGAN (Schlegl et al., 2019), only a few works have been published. They either use cycle GANs (Xia et al., 2020) or combine GANs with autoencoders (Shi et al., 2023; Bercea et al., 2023d). On the other hand, even though VAEs' image generation quality is lower, they are easy to train, they scale well to high-dimensional data, provide good interpretation capacity thanks to their regularized latent space, and are able to handle small datasets. Many new VAE extensions have shown their efficacy in the computer vision literature (Burda et al., 2016; Burgess et al., 2018; Caterini et al., 2018; Chen et al., 2018a; Davidson et al., 2018; Ghosh et al., 2019; Higgins et al., 2017; Kim et al., 2018; Kingma et al., 2016; Larsen et al., 2016; Makhzani et al., 2015; Rezende et al., 2015; Snell et al., 2017; Tolstikhin et al., 2018; Tomczak et al., 2018; Van Den Oord et al., 2017; Zhao et al., 2019), but only a handful have been applied to medical imaging (Baur et al., 2021a; Baur et al., 2021b; Chen et al., 2018b; Choi et al., 2019; Mostapha et al., 2019; Uzunova et al., 2019; Harkness et al., 2023).

In 2021, Baur et al., 2021a compared VAE-based approaches to the best GANs for unsupervised anomaly segmentation in brain structural magnetic resonance imaging (MRI). It was conducted on models that had already been employed for UAD in the medical imaging context, such as Context VAE (Zimmerer et al., 2019), Constrained AAE (Chen et al.,

2018b), or AnoVAEGAN (Baur et al., 2019). They showed that the vanilla VAE used for density-based restoration (Chen et al., 2020) outperforms other models, including GAN approaches, at the cost of a longer inference time. They compared the performance of their models using segmentation metrics such as the dice similarity coefficient (DSC) and the area under the precision-recall curve (AUPRC) computed between the residual (i.e. the difference between the input and the reconstructed image) and the ground truth anomaly mask provided in the datasets they used. This study focused on the segmentation of glioblastoma and multiple sclerosis lesions, which consist of sharp and intense anomalies that are segmented in 2D slices extracted from MRI volumes.

Following the work of Baur et al., 2021a, we propose a benchmark of 20 VAE-based models focused on the pseudo-healthy reconstruction of 3D FDG PET images for anomaly detection in the context of dementia. We compare many VAE-based models that have not been applied to medical image analysis yet, thanks to the software package of Chadebec et al., 2022¹. In contrast to computer vision works, where datasets typically contain several tens of thousands of images, it will be interesting to examine the performance of such models when trained on a relatively small dataset, comprising only a few hundred images, which is typical in medical imaging. Our contributions are threefold:

1. first, we propose a rigorous method and provide the associated software tool that we used to define the optimal architecture of the vanilla VAE and select the best hyper-parameters of the VAE variants in the context of neuroimaging;
2. then, we put in application the evaluation framework introduced in Chapter 3 to thoroughly assess the ability of 20 VAE models to reconstruct pseudo-healthy images for the detection of dementia-related anomalies in 3D brain FDG PET and compare their performance;
3. finally, we conclude on the best performing models, providing a state-of-the-art on the use of 3D convolutional VAEs in such context.

A preliminary version of this work was published as a conference paper (Hassanally et al., 2023b), and is available in Appendix D. The present chapter is an extension of this previous work with the following improvement: (i) the addition of new VAE-based models; (ii) an extensive search of the best encoder-decoder architecture and hyper-parameters for each model; (iii) the use of full resolution 3D brain FDG PET; (iv) and an extensive evaluation of the different models.

5.1 Extensions to the variational autoencoder framework

Several contributions have been proposed to improve the VAE framework (Chadebec et al., 2022). These contributions can be divided into four categories that correspond to different objectives.

The aim of the first category of approaches is to improve the prior distribution $p(\mathbf{z})$ by using a variational mixture of posteriors as prior (VAMP) (Tomczak et al., 2018), by using a

¹<https://pythae.readthedocs.io/en/latest/index.html>

specific geometry in the latent space such as hyperspherical VAE (SVAE) (Davidson et al., 2018), by learning the prior on a discrete latent space with vector quantized-VAE (VQVAE) (Van Den Oord et al., 2017), or by substituting the prior with a density estimation method using regularization with a gradient penalty (RAE-GP) or an ℓ^2 penalty on the decoder (RAE- ℓ^2) (Ghosh et al., 2019).

Other methods aim to better estimate the lower bound by using importance weighting (IWAE) (Burda et al., 2016), by using linear normalizing flows (VAE LinNF) (Rezende et al., 2015), inverse autoregressive flows (VAE-IAF) (Kingma et al., 2016) or Markov chain Monte Carlo using Hamiltonian importance sampling (HVAE) (Caterini et al., 2018) to better estimate the posterior.

Approaches in the third category encourage disentanglement of the features in the latent space by adding a weight to balance the terms of the loss in Equation 2.12 (β -VAE) (Higgins et al., 2017), subsequently improved with a better reconstruction capacity by progressively increasing the KL-divergence term (Disentangled β -VAE) (Burgess et al., 2018), by decomposing the loss to show a total correlation term (β -TC VAE) (Chen et al., 2018a), or by encouraging the distribution of the latent variable $q(\mathbf{z})$ to be factorial (FactorVAE) (Kim et al., 2018).

Finally, other methods change the distance computed between the distributions by adding the mutual information between \mathbf{x} and \mathbf{z} as regularization (InfoVAE) (Zhao et al., 2019), using another divergence term in the loss such as the maximum mean discrepancy in the Wasserstein autoencoder (WAE) (Tolstikhin et al., 2018) or a discriminator to differentiate a prior's sample from a posterior's sample in the adversarial autoencoder (Adv. AE) (Makhzani et al., 2015), or by changing the reconstruction metric for another similarity metric such as the multi-scale structural similarity (MS-SSIM VAE) (Snell et al., 2017), or for the prediction of a discriminator on the output of the VAE (VAEGAN) (Larsen et al., 2016).

All of these models, described in more detail in Appendix E, perform well in computer vision, as shown by Chadebec et al., 2022 who compared 19 of them on classic computer vision datasets (MNIST, CIFAR10 and CELEBA) on five tasks: image reconstruction, image generation, classification, clustering and interpolation. However, they have not been compared in the context of medical imaging.

5.2 Selection method and evaluation of the models

When evaluating unsupervised anomaly detection approaches, two aspects are usually assessed: their ability to reconstruct images of high quality and their ability to detect anomalies. The first aspect can only be fully assessed when reconstructing images of healthy subjects. Commonly used metrics are the mean-squared error (MSE), the peak signal-to-noise ratio (PSNR) and the structural similarity (SSIM) (Wang et al., 2004). These paired metrics are computed between the input and reconstructed images (Nečasová et al., 2022). See Chapter 3, Section 3.2.1 for more information. To assess the second aspect, since we not have the ground truth anomaly masks, we rely on the evaluation framework presented in Chapter 3, Section 3.2.2. It consists in simulating the effect of a disease on images of

healthy subjects by reducing the PET uptake within areas of the brain associated with different dementias (Burgos et al., 2021b) defined using a mask M . This approach effectively replicates realistic regional hypometabolism and provides pairs of diseased images with the original healthy scan that is used as the target ground truth for the pseudo-healthy reconstruction. We also use the defined healthiness score \mathcal{H} to evaluate whether a model is able to reconstruct images that are looking healthy. This metric is supposed to be around 1 for images of healthy subjects, lower than 1 for simulated images, and expected to be around 1 again for the pseudo-healthy reconstructions.

As we consider that to accurately detect anomalies a model should reconstruct pseudo-healthy images of high quality, we use the pairwise performance measures as a first step of our evaluation to select the best models. We especially rely on the SSIM, rather than the MSE or PSNR, as it is a perceptual metric that appears more informative than a pixel-wise difference, and because it is a different metric than the optimization criterion, which is MSE for all the models except for the MS-SSIM VAE (Snell et al., 2017). In particular, the SSIM is used as the selection criterion when searching for the best hyper-parameters' configurations and selecting the best trained models, and is combined with the MSE when searching for the best encoder-decoder architecture. We use the simulation framework with the healthiness metric in a second step to push further the evaluation of the trained models being compared.

5.3 Materials

As explained in details in Section 1.2, we use FDG PET images from the ADNI dataset (Mueller et al., 2005; Jagust et al., 2010; Jagust et al., 2015). The images are preprocessed using the `pet-linear` pipeline from the Clinica open source software (Routier et al., 2021). They are then carefully selected, for a final set of 739 images from 378 CN subjects and 353 images at baseline from 353 AD patients. We split our CN subjects into train/validation and test sets at subject level, and perform a 6-fold cross validation on the train/validation set using the ClinicaDL open source software (Thibeau-Sutre et al., 2022b). All the AD patients belongs to the test set.

We also use the 60 images from the CN test set to build new test sets by using our simulation method presented in Chapter 3, Section 3.2.2, which results in a total of 14 simulated test sets. These test sets are denoted using the dementia simulated and the hypometabolism intensity. For instance, "Test AD 30" corresponds to images simulating AD with a 30% hypometabolism.

5.4 Model selection

We aim to compare 20 AE and VAE-based models. For the comparison to be meaningful, we must find the best architectures and parameters for each model. We decided to use the same encoder-decoder architecture for all the models as it would have been too long to find an optimal architecture for each model, and as we believe it makes the comparison fairer. The architecture was obtained using a random search on the vanilla VAE. We then

attempted to find optimal hyper-parameters for each model through either a random search or a grid search. Once the best parameters for each model were found, we trained them on all the six splits of the cross-validation, and we selected the best. The best trained models were finally evaluated using the simulation framework and metric presented in Chapter 3. The procedure is summarized in Figure 5.1. The random search and evaluation procedures are implemented in ClinicaDL (Thibeau-Sutre et al., 2022b)² while the VAE-based models are implemented in Pythae (Chadebec et al., 2022)³, which are both open-source software packages.

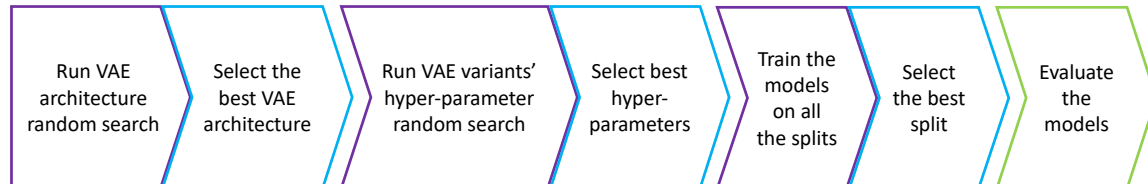


FIGURE 5.1: Diagram summarizing the benchmark steps. We represent steps performed on training sets in purple (random search and training), the selections on the validation sets are represented in blue and the final evaluation on test sets is in green.

All the models were trained for 200 epochs on an HPC with Nvidia Tesla V100 GPUs that have 32 GB of dedicated memory. The choice for the batch size and the learning rate will be discussed further in this section. We used the same environment to train all the models.

5.4.1 Selection of the encoder-decoder architecture

The training parameters and the encoder-decoder architecture were selected with a random search for the vanilla VAE. We trained the models on a random selection of three splits as a trade-off between reducing the variance due to data splitting and the computational time required to train the models. We then selected the models based on the average SSIM and MSE, computed within the full image field of view, on the validation sets.

We defined a modular architecture for the encoder and decoder, which is shown in Figure 5.2. The encoder (shown in Figure 5.2a) is composed of a number B_e of blocks, each containing a number S_e of sub-blocks. Similarly, the decoder is composed of a number B_d of blocks, each containing a number S_d of sub-blocks (Figure 5.2b). For a chosen architecture, the sub-blocks can either all be convolutional or all be residual (see Figure 5.2c). In both cases, the convolution layers are followed by a batch normalization, and we use a swish activation function as suggested in Vahdat et al., 2020.

In the encoder, the number of channels is doubled by the first convolution in each block. At the same time, the size of the image is divided by 2 along each dimension by using a 3D convolution with kernel size (4, 4, 4), stride (2, 2, 2) and padding (1, 1, 1). The following sub-blocks are optional, and their convolution operations have kernel size (3, 3, 3), stride (1, 1, 1) and padding (1, 1, 1). In the decoder, the last sub-block of each block is preceded by an upsampling layer, to be roughly symmetrical with the encoder. Convolution operations in

²<https://clinicadl.readthedocs.io/en/latest/>

³<https://pythae.readthedocs.io/en/latest/>

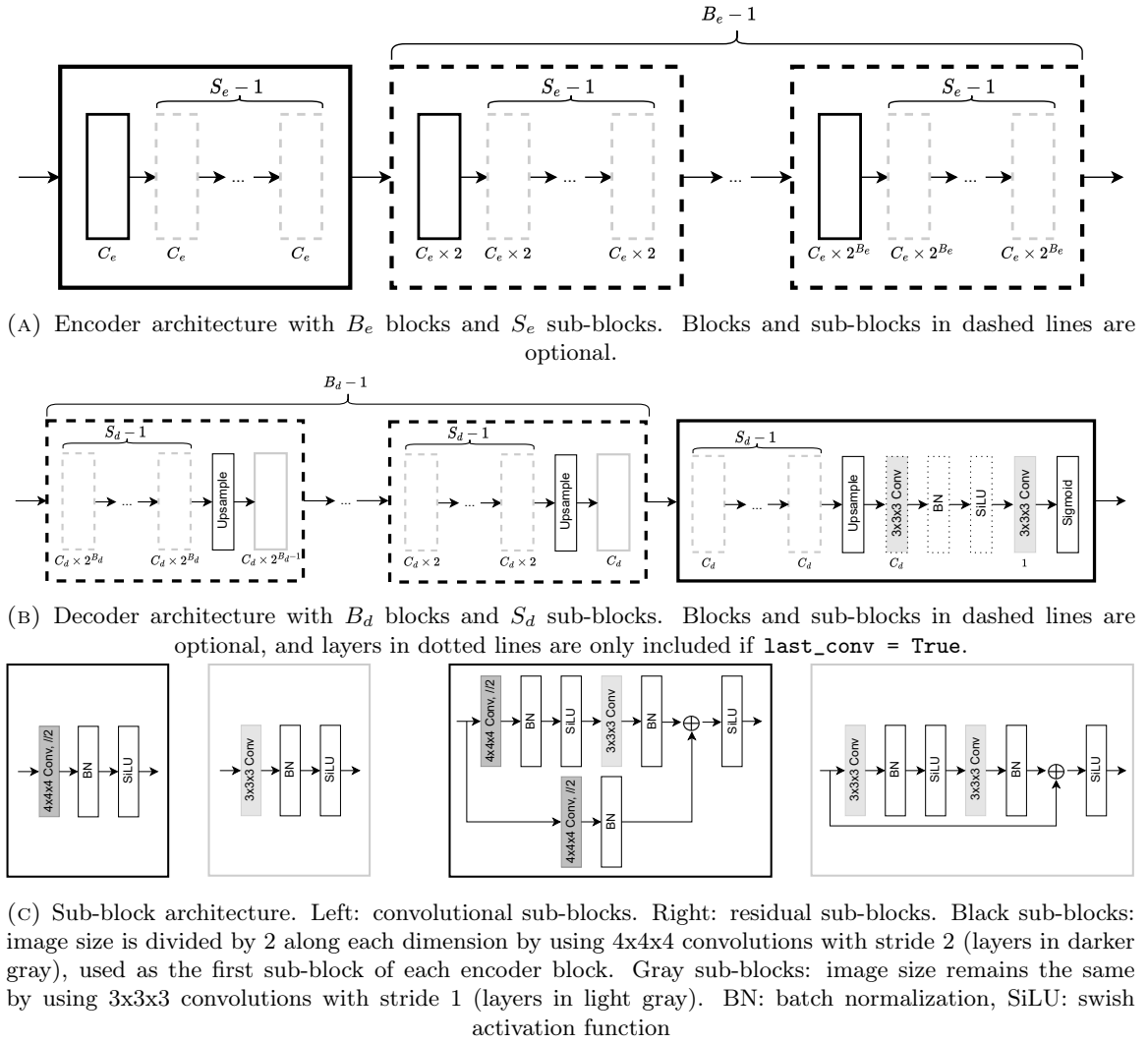


FIGURE 5.2: Encoder-decoder modular architecture. The number of convolution kernels in each sub-block is indicated under the sub-block (e.g. C_e).

the decoder have kernel size $(3, 3, 3)$, stride $(1, 1, 1)$ and padding $(1, 1, 1)$. This architecture was inspired from ResNet models (He et al., 2016) and VGG models (Simonyan et al., 2014).

The parameters of this modular architecture (summarized in Table 5.1) are therefore the following: the latent space size, the number of blocks in the encoder B_e , the number of blocks in the decoder B_d , the number of sub-blocks per encoder block S_e , the number of sub-blocks per decoder blocks S_d , the number of channels for the first encoder block C_e , the number of channels for the last decoder block C_d , and the layer type (convolution or residual). We implement a random search to explore this parameter space, and choose possible values for each parameter based on previous experiments, intermediate results (as we launched the random search in successive batches) and intuition. We decided to set the batch size of the data loader to 8. Even though this is a constraint for configurations that would require more memory, this choice allows flexibility; in scenarios where certain VAE variants require more memory, we can reduce the batch size while maintaining a reasonable number of images per batch (e.g., 6 or 4). Details of all the parameters tested, and their impact are discussed in Appendix F.

TABLE 5.1: Hyper-parameters included in our encoder-decoder VAE architecture random search

Hyper-parameter	Label	Search space	Selected value
Number of encoder blocks	B_e	{4, 5, 6}	5
Number of sub-blocks per encoder block	S_e	{1, 2, 3}	1
Number of channels for the first encoder sub-block	C_e	{16, 32}	16
Number of decoder blocks	B_d	{4, 5, 6}	5
Number of decoder sub-blocks	S_d	{1, 2, 3}	1
Number of channels for the last decoder sub-block	C_d	{16, 32}	16
Latent space size		{256, 512, 1024}	256
Learning rate		{ 10^{-3} , 10^{-4} , 10^{-5} }	10^{-4}
Block type		{conv, res}	conv
Added convolution in last decoder block	last_conv	{True, False}	False

After comparing around 200 configurations, the encoder architecture selected is composed of five blocks, each with one sub-block, each containing a convolutional layer, a batch normalization and a swish activation function. These blocks are followed by a flatten and a fully connected layer. The latent space has size 256. The decoder is symmetrical, it is composed of a fully connected layer followed by five blocks, each with one sub-block, each composed of an upsampling layer, a convolutional layer, a batch normalization and a swish activation. This model has 16 channels after the first encoder block and before the last decoder block. This is shown in Figure 5.3 and detailed in Table 5.2.

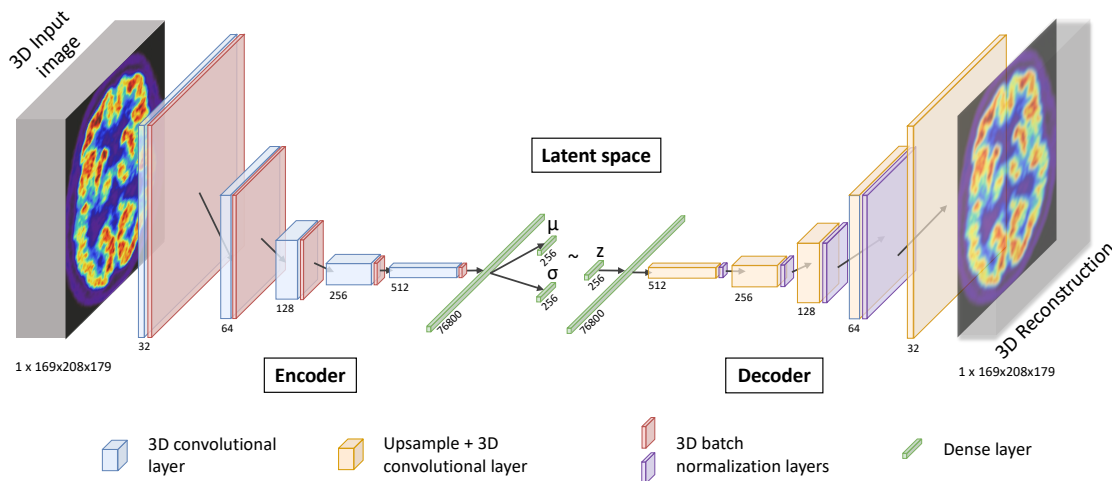


FIGURE 5.3: Diagram of the selected VAE architecture

5.4.2 Selection of the models' hyper-parameters

Once we found an encoder-decoder architecture that gave good performance, we used it for the AE and 18 VAE variants presented in Section 5.1. However, all of these variants, except the SVAE (Davidson et al., 2018), have supplementary hyper-parameters that may have significant impact on the models' performance. We therefore searched for the best configuration of hyper-parameters for each model in the context of 3D brain FDG PET

TABLE 5.2: Summary of the layer parameters in the final VAE architecture

Layer	Input shape	Output shape	Kernel	Stride	Padding	BN	Activation
Encoder							
Conv3D	(1, 169, 208, 179)	(16, 84, 104, 89)	(4, 4, 4)	(2, 2, 2)	(1, 1, 1)	True	SILU
Conv3D	(16, 84, 104, 89)	(32, 42, 52, 44)	(4, 4, 4)	(2, 2, 2)	(1, 1, 1)	True	SILU
Conv3D	(32, 42, 52, 44)	(64, 21, 26, 22)	(4, 4, 4)	(2, 2, 2)	(1, 1, 1)	True	SILU
Conv3D	(64, 21, 26, 22)	(128, 10, 13, 11)	(4, 4, 4)	(2, 2, 2)	(1, 1, 1)	True	SILU
Conv3D	(128, 10, 13, 11)	(256, 5, 6, 5)	(4, 4, 4)	(2, 2, 2)	(1, 1, 1)	True	SILU
Flatten	(256, 5, 6, 5)	(38400)	-	-	-	-	-
FC × 2	(38400)	(256) × 2	-	-	-	False	SILU
Decoder							
FC	(256)	(38400)	-	-	-	False	ReLU
Unflatten	(38400)	(256, 5, 6, 5)	-	-	-	-	-
Upsample	(256, 5, 6, 5)	(256, 10, 13, 11)	-	-	-	-	-
Conv3D	(256, 10, 13, 11)	(128, 10, 13, 11)	(3, 3, 3)	(1, 1, 1)	(1, 1, 1)	True	SILU
Upsample	(128, 10, 13, 11)	(128, 21, 26, 22)	-	-	-	-	-
Conv3D	(128, 21, 26, 22)	(64, 21, 26, 22)	(3, 3, 3)	(1, 1, 1)	(1, 1, 1)	True	SILU
Upsample	(64, 21, 26, 22)	(64, 42, 52, 44)	-	-	-	-	-
Conv3D	(64, 42, 52, 44)	(32, 42, 52, 44)	(3, 3, 3)	(1, 1, 1)	(1, 1, 1)	True	SILU
Upsample	(32, 42, 52, 44)	(32, 84, 104, 89)	-	-	-	-	-
Conv3D	(32, 84, 104, 89)	(16, 84, 104, 89)	(3, 3, 3)	(1, 1, 1)	(1, 1, 1)	True	SILU
Upsample	(16, 84, 104, 89)	(16, 169, 208, 179)	-	-	-	-	-
Conv3D	(16, 169, 208, 179)	(1, 169, 208, 179)	(3, 3, 3)	(1, 1, 1)	(1, 1, 1)	False	Sigmoid

BN: batch normalization, FC: fully connected, SILU: swish activation function

reconstruction by launching either a random search (when we searched for more than one hyper-parameter) or grid search (when there is only one hyper-parameter). Similarly to the architecture search, we train each configuration on three splits. We then selected the best set of hyper-parameters for each VAE-based model, using the best average SSIM on the validation set as criterion.

As there were many models to optimize, we limited the number of random searches to $N \times 10$ with N the number of parameters to search. For instance, when there was only one hyper-parameter to tune, we launched a grid search with maximum 10 different values for that parameter, if there were two parameters, we trained a maximum of 20 models and so on. This may not be the fairest decision as it does not allow exploring the same percentage of the parameter space depending on N (as it scales to the power N and not linearly), but it accounts for the fact that a model with more parameters may be more tedious to tune. Moreover, we carefully chose a range of values to test for each hyper-parameter of each model based on the original implementation papers, our prior knowledge of the models, and the work done by Chadebec et al., 2022. Note that some of the hyper-parameters were excluded from our search when an optimal value was provided in the literature, which allowed reducing the number of configurations trained.

Following the vanilla VAE training, the different configurations were trained by default with a batch size of 8 and a learning rate of 10^{-4} on 200 epochs. When some set of hyper-parameters led to memory errors, we gradually reduced the batch size to 6, 4 or 2, and when they lead to errors in the computation of the loss, we reduced the learning rate to 10^{-5} . In spite of this, combinations leading to errors were removed, further reducing the size of the hyper-parameter space.

The details relating to the different VAE-based models, their hyper-parameters, the random search, the trained configurations and the results are provided in Appendix E. A summary is proposed in Table 5.3.

TABLE 5.3: Summary of the hyper-parameters optimized and selected thanks to the random search for each VAE variant. The hyper-parameters are detailed in E.

Models	Hyper-parameters	Search space	Selected value
Adv. AE (Makhzani et al., 2015)	adv. loss scale	{0.001, 0.01, 0.05, 0.1, 0.25,	0.9
		0.5, 0.75, 0.9, 0.95, 0.99}	
β -TC VAE (Chen et al., 2018a)	β	{0.001, 0.005, 0.01, 0.05, 0.1, 1, 2, 5, 10}	2
	α	{1, 3}	1
	γ	{1, 3}	1
β -VAE (Higgins et al., 2017)	β	{0.001, 0.005, 0.01, 0.05,	10
		0.1, 0.5, 2, 5, 10, 100}	
Dis. β -VAE (Burgess et al., 2018)	β	{0.01, 0.1, 1, 5, 10}	10
	C	{5, 25, 50}	50
	warm-up epochs	{100, 1000}	1000
FactorVAE (Kim et al., 2018)	γ	{2, 5, 10, 15, 20, 30, 40, 50, 100, 200}	40
HVAE (Caterini et al., 2018)	n_{lf}	{1, 2, 10, 15, 20}	10
	ϵ_{lf}	{0.00001, 0.0001, 0.001, 0.01}	0.00001
	β_0	{0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0}	0.8
InfoVAE (Zhao et al., 2019)	kernel choice	{rbf, imq}	rbf
	α	{0.0, 0.2, 0.4, 0.6, 0.8, 1.0}	1
	λ	{0.01, 0.1, 1, 10, 100}	0.1
	kernel bandwidth	{0.01, 0.1, 0.5, 1, 5, 10, 100}	0.1
IWAE (Burda et al., 2016)	n samples	{2, 3, 4, 5, 6, 8, 10, 12, 15, 20}	6
MS-SSIM VAE (Snell et al., 2017)	β	{0.01, 0.1, 1, 10, 100}	-
	window size	{2, 3, 5, 11}	-

TABLE 5.3: Summary of the hyper-parameters optimized and selected thanks to the random search for each VAE variant. The hyper-parameters are detailed in E.

Models	Hyper-parameters	Search space	Selected value
RAE- ℓ^2 (Ghosh et al., 2019)	embedding weight	{0.00001, 0.0001, 0.001, 0.01, 0.1, 1}	0.0001
	reg. weight	{0.00001, 0.0001, 0.001, 0.01, 0.1, 1}	1
RAE-GP (Ghosh et al., 2019)	embedding weight	{0.00001, 0.0001, 0.001, 0.01, 0.1, 1}	0.01
	reg. weight	{0.00001, 0.0001, 0.001, 0.01, 0.1, 1}	0.0001
SVAE (Davidson et al., 2018)	latent space size	{8, 16, 32}	-
VAEGAN (Larsen et al., 2016)	adv. loss scale	{0.3, 0.5, 0.7, 0.9}	0.5
	reconstruction layer	{1, 2, 3, 4}	1
VAE-IAF (Kingma et al., 2016)	n MADE blocks	{2, 4, 6, 8}	4
	n hidden in MADE	{2, 3, 4, 5}	4
	hidden size	{64, 128, 256}	128
VAE LinNF (Rezende et al., 2015)	flows	{10P, 10R, 5P, 5R, 5P5R, 5R5P, 5PR, 5RP, 2PR, 2RP}	10R
VAMP (Tomczak et al., 2018)	number components	{10, 20, 30, 40, 50}	-
	linear scheduling steps	{0, 20, 40}	-
VQVAE (Van Den Oord et al., 2017)	quantization loss factor	{0.25, 0.5, 0.75, 0.9, 1, 1.5, 2, 4}	2
	n embeddings	{128, 256, 512, 1024}	512
WAE (Tolstikhin et al., 2018)	kernel choice	{rbf, imq}	rbf
	reg. weight	{0.01, 0.1, 0.5, 1, 5, 10, 100}	0.1
	kernel bandwidth	{0.01, 0.1, 0.5, 1, 5, 10, 100}	5

We report in Table 5.4 reconstruction metrics for the 18 VAE variants with the best configuration of hyper-parameters that we tested. Out of all the models, only three did not perform well on the validation set (highlighted in gray): the VAMP (Tomczak et al., 2018) with an average SSIM of 0.702, the MS-SSIM VAE (Snell et al., 2017) with an average SSIM of 0.472 and the SVAE with a very low average SSIM of 0.151. We found it quite surprising that the MS-SSIM VAE (Snell et al., 2017) performed so poorly in terms of average SSIM, since it optimizes a perceptual metric related to the SSIM, namely the multi-scale SSIM. These results could potentially be explained by the fact that the MS-SSIM computation in 3D is very costly, meaning that the only kernel size that allowed training in a reasonable amount of time was 2, potentially leading to a poor estimation of the metric, especially since the kernel size suggested in the MS-SSIM original implementation is 11 (Wang et al., 2003). In the end, only three models with three different combinations of parameters were trained successfully, possibly explaining why we did not find a configuration giving acceptable reconstruction. Finally, the SVAE did not train with a high dimensional latent space (hundred and above) due to the computation of the Bessel function in the loss. Since this model does not have any hyper-parameter to tune, we decided to launch a grid search to find the best latent space size (within the set $\{8, 16, 32\}$). The reduction of the latent space size may explain why the reconstruction is not satisfying, as we know that low latent dimensions lead to poorer reconstructions. Moreover, the SVAE seems to be better suited for hyperspherical data distributions, which is not the case in our application. For the following experiments, we decided not to consider the VAMP (Tomczak et al., 2018), MS-SSIM VAE (Snell et al., 2017) and SVAE (Davidson et al., 2018).

TABLE 5.4: Reconstruction metrics obtained for the best configuration of each VAE variant on the validation sets (mean \pm std computed over the three splits randomly selected)

Models	SSIM \uparrow	MSE ($\times 10^{-3}$) \downarrow	PSNR \uparrow
β -TC VAE (Chen et al., 2018a)	0.870 \pm 0.002	1.901 \pm 0.123	27.41 \pm 0.27
β -VAE (Higgins et al., 2017)	0.868 \pm 0.003	1.995 \pm 0.067	27.17 \pm 0.14
Dis. β -VAE (Burgess et al., 2018)	0.874 \pm 0.006	2.004 \pm 0.153	27.18 \pm 0.30
FactorVAE (Kim et al., 2018)	0.876 \pm 0.003	1.895 \pm 0.084	27.47 \pm 0.14
HVAE (Caterini et al., 2018)	0.873 \pm 0.007	1.862 \pm 0.068	27.48 \pm 0.14
InfoVAE (Zhao et al., 2019)	0.877 \pm 0.006	1.813 \pm 0.075	27.63 \pm 0.13
IWAE (Burda et al., 2016)	0.865 \pm 0.007	2.087 \pm 0.146	27.02 \pm 0.24
MS-SSIM VAE (Snell et al., 2017)	0.472 \pm 0.034	70.174 \pm 5.660	11.61 \pm 0.36
RAE-GP (Ghosh et al., 2019)	0.880 \pm 0.006	1.715 \pm 0.105	27.84 \pm 0.26
RAE- ℓ^2 (Ghosh et al., 2019)	0.884 \pm 0.005	1.815 \pm 0.049	27.61 \pm 0.11
SVAE (Davidson et al., 2018)	0.151 \pm 0.001	632.694 \pm 5.106	1.99 \pm 0.03
VAEGAN (Larsen et al., 2016)	0.860 \pm 0.013	2.241 \pm 0.193	26.64 \pm 0.38
VAE-IAF (Kingma et al., 2016)	0.823 \pm 0.005	2.272 \pm 0.057	26.65 \pm 0.08
VAE LinNF (Rezende et al., 2015)	0.871 \pm 0.001	1.855 \pm 0.125	27.54 \pm 0.21
VAMP (Tomczak et al., 2018)	0.702 \pm 0.097	5.581 \pm 0.874	22.73 \pm 0.72
VQVAE (Van Den Oord et al., 2017)	0.881 \pm 0.003	1.805 \pm 0.032	27.62 \pm 0.07
WAE (Tolstikhin et al., 2018)	0.881 \pm 0.005	1.862 \pm 0.075	27.54 \pm 0.08

5.4.3 Selection of the best trained models

Once the best parameters were selected through the random search, all 17 models (AE, VAE and the 15 remaining VAE-based models) were trained on the six splits of the cross-validation. We kept the same training parameters as for the random search: the models were trained on 200 epochs, with a learning rate of 10^{-4} and a batch size of 8. There were a few exceptions: the VAE-IAF (Kingma et al., 2016) was trained with a learning rate of 10^{-5} to avoid errors during training. The RAE-GP (Ghosh et al., 2019) was trained with a batch size of 6, the VAEGAN (Larsen et al., 2016) with a batch size of 4 and the IWAE (Burda et al., 2016) with a batch size of 2 because of the high memory usage of these models.

We then selected the best fold for each model using the average SSIM on the validation sets. The performance of all 17 models on the six splits are presented in Table 5.5, with the best split of each model highlighted in bold. We can notice that the splits 2 and 3 are over-represented among the selected models. This can be explained by the fact that the cross-validation is not stratified, and the distributions of age and sex between training and validation sets for split 2 and 3 are more similar than for the other splits (Table 1.1).

5.5 Results obtained for the best models on the test sets

Once all the models with a correct reconstruction were trained and the best model was selected (optimal set of parameters among those tested and best split), we could evaluate each model using the procedure defined in Section 5.2. Pseudo-healthy images were reconstructed for each of the 15 test sets (the test set with the images of healthy subjects and the 14 test sets with simulated images) in order to measure the performance of the models both qualitatively by visualizing the pseudo-healthy reconstructions, and quantitatively by computing the reconstruction metrics and the healthiness score.

5.5.1 Quantitative evaluation of the pseudo-healthy reconstructions from images of control subjects

We first assessed whether the different models could correctly reconstruct images of healthy subjects from the test set by computing the SSIM, MSE and PSNR between the input and the pseudo-healthy reconstruction. Results are reported in Table 5.6. We observe that the reconstruction metrics of all but two models are in the same order of magnitude, with an SSIM on average between 0.873 (VAE, Kingma et al., 2014) and 0.887 (RAE-GP, Ghosh et al., 2019), an MSE on average between 1.6×10^{-3} for the RAE-GP (Ghosh et al., 2019) and 1.859×10^{-3} for the IWAE (Burda et al., 2016), and a PSNR on average between 26.7 (VAEGAN, Larsen et al., 2016) and 28.1 (RAE-GP, Ghosh et al., 2019). This shows that the RAE-GP (Ghosh et al., 2019) has the best reconstruction capacity. On the other hand, the VAEGAN (Larsen et al., 2016) and the VAE-IAF (Kingma et al., 2016) perform the worst, with respectively an average SSIM of 0.866 and 0.837, and an average MSE of 2.195×10^{-3} and 2.099×10^{-3} , which is even worse than the vanilla VAE and the AE.

TABLE 5.5: SSIM obtained on each validation set of the 6-fold cross-validation for the different trained models (mean \pm std over the images from the validation set). The best split of each VAE variant is highlighted in bold.

Models	Split 0	Split 1	Split 2	Split 3	Split 4	Split 5
AE	0.865 \pm 0.032	0.860 \pm 0.026	0.876 \pm 0.036	0.876 \pm 0.024	0.875 \pm 0.030	0.859 \pm 0.040
Adv. AE (Makizani et al., 2015)	0.866 \pm 0.034	0.852 \pm 0.032	0.868 \pm 0.034	0.881 \pm 0.024	0.871 \pm 0.029	0.862 \pm 0.037
β -TC VAE (Chen et al., 2018a)	0.858 \pm 0.036	0.869 \pm 0.024	0.869 \pm 0.030	0.876 \pm 0.024	0.860 \pm 0.035	0.871 \pm 0.037
β -VAE (Higgins et al., 2017)	0.870 \pm 0.029	0.869 \pm 0.024	0.874 \pm 0.030	0.872 \pm 0.027	0.865 \pm 0.032	0.864 \pm 0.036
Dis. β -VAE (Burgess et al., 2018)	0.870 \pm 0.029	0.868 \pm 0.023	0.879 \pm 0.027	0.868 \pm 0.026	0.865 \pm 0.033	0.865 \pm 0.036
FactorVAE (Kim et al., 2018)	0.863 \pm 0.033	0.849 \pm 0.024	0.874 \pm 0.033	0.872 \pm 0.028	0.873 \pm 0.027	0.861 \pm 0.040
HVAE (Caterini et al., 2018)	0.840 \pm 0.040	0.869 \pm 0.025	0.864 \pm 0.031	0.867 \pm 0.027	0.878 \pm 0.027	0.858 \pm 0.038
InfoVAE (Zhao et al., 2019)	0.870 \pm 0.030	0.871 \pm 0.025	0.874 \pm 0.031	0.873 \pm 0.024	0.876 \pm 0.027	0.870 \pm 0.036
IWAE (Burda et al., 2016)	0.861 \pm 0.036	0.860 \pm 0.031	0.867 \pm 0.033	0.868 \pm 0.028	0.875 \pm 0.026	0.861 \pm 0.046
RAE-GP (Ghosh et al., 2019)	0.878 \pm 0.030	0.872 \pm 0.028	0.884 \pm 0.029	0.884 \pm 0.029	0.880 \pm 0.026	0.873 \pm 0.033
RAE- ℓ^2 (Ghosh et al., 2019)	0.857 \pm 0.037	0.873 \pm 0.025	0.850 \pm 0.040	0.882 \pm 0.023	0.868 \pm 0.031	0.863 \pm 0.038
VAE (Kingma et al., 2014)	0.866 \pm 0.030	0.868 \pm 0.024	0.869 \pm 0.030	0.865 \pm 0.029	0.851 \pm 0.041	0.868 \pm 0.037
VAEGAN (Larsen et al., 2016)	0.804 \pm 0.044	0.863 \pm 0.025	0.846 \pm 0.038	0.866 \pm 0.026	0.855 \pm 0.038	0.858 \pm 0.035
VAE-IAF (Kingma et al., 2016)	0.827 \pm 0.037	0.818 \pm 0.032	0.829 \pm 0.034	0.828 \pm 0.027	0.828 \pm 0.034	0.823 \pm 0.037
VAE LinNF (Rezende et al., 2015)	0.860 \pm 0.033	0.870 \pm 0.024	0.865 \pm 0.035	0.852 \pm 0.032	0.858 \pm 0.039	0.870 \pm 0.036
VQVAE (Van Den Oord et al., 2017)	0.857 \pm 0.036	0.852 \pm 0.027	0.869 \pm 0.031	0.883 \pm 0.025	0.868 \pm 0.031	0.864 \pm 0.037
WAE (Tolstikhin et al., 2018)	0.870 \pm 0.032	0.871 \pm 0.024	0.871 \pm 0.033	0.882 \pm 0.024	0.869 \pm 0.034	0.862 \pm 0.038

TABLE 5.6: Reconstruction metrics obtained for images from Test CN (mean \pm std computed over images from the test set)

Models	SSIM \uparrow	MSE ($\times 10^{-3}$) \downarrow	PSNR \uparrow
AE	0.882 ± 0.026	1.649 ± 0.613	28.00 ± 1.10
Adv. AE (Makhzani et al., 2015)	0.882 ± 0.028	1.707 ± 0.610	27.83 ± 1.06
β -TC VAE (Chen et al., 2018a)	0.878 ± 0.025	1.720 ± 0.565	27.79 ± 1.05
β -VAE (Higgins et al., 2017)	0.876 ± 0.027	1.846 ± 0.638	27.49 ± 1.04
Dis. β -VAE (Burgess et al., 2018)	0.880 ± 0.023	1.841 ± 0.634	27.50 ± 1.06
FactorVAE (Kim et al., 2018)	0.879 ± 0.026	1.651 ± 0.584	27.98 ± 1.06
HVAE (Caterini et al., 2018)	0.882 ± 0.024	1.809 ± 0.635	27.59 ± 1.09
InfoVAE (Zhao et al., 2019)	0.883 ± 0.024	1.704 ± 0.594	27.84 ± 1.04
IWAE (Burda et al., 2016)	0.876 ± 0.026	1.859 ± 0.564	27.44 ± 1.03
RAE-GP (Ghosh et al., 2019)	0.887 ± 0.023	1.605 ± 0.671	28.14 ± 1.13
RAE- ℓ^2 (Ghosh et al., 2019)	0.882 ± 0.024	1.631 ± 0.531	28.02 ± 1.02
VAE (Kingma et al., 2014)	0.873 ± 0.028	1.736 ± 0.566	27.75 ± 1.02
VAEGAN (Larsen et al., 2016)	0.866 ± 0.027	2.195 ± 0.641	26.72 ± 1.04
VAE-IAF (Kingma et al., 2016)	0.837 ± 0.027	2.099 ± 0.720	26.92 ± 1.00
VAE LinNF (Rezende et al., 2015)	0.881 ± 0.023	1.807 ± 0.610	27.58 ± 1.05
VQVAE (Van Den Oord et al., 2017)	0.884 ± 0.026	1.649 ± 0.593	27.99 ± 1.10
WAE (Tolstikhin et al., 2018)	0.883 ± 0.026	1.651 ± 0.618	27.99 ± 1.10

5.5.2 Quantitative evaluation of the pseudo-healthy reconstructions from images with simulated dementia

The first evaluation step with simulated data is to compute reconstruction metrics between the pseudo-healthy reconstructions obtained from these simulated data and the ground truth images used to simulate hypometabolic images, which are the targets. These results are reported in Table 5.7. For all the models, the reconstructions are slightly worse than for images reconstructed from the ground truth itself (Table 5.6) with an average SSIM between 0.854 (VAEGAN, Larsen et al., 2016) and 0.878 (RAE-GP, Ghosh et al., 2019), an average MSE between 1.997×10^{-3} for the RAE- ℓ^2 (Ghosh et al., 2019) and 2.650×10^{-3} for the VAEGAN (Larsen et al., 2016), and an average PSNR between 25.88 (VAEGAN, Larsen et al., 2016) and 27.12 (RAE- ℓ^2 , Ghosh et al., 2019). The only exception is the VAE-IAF (Kingma et al., 2016), for which the SSIM increases from 0.837 on average to 0.842. However, the reconstruction metrics are still quite high, meaning that the reconstructions from simulated hypometabolic images are similar to their target.

5.5.3 Qualitative evaluation of the pseudo-healthy reconstructions

Examples of pseudo-healthy reconstructions obtained from the original image of a control subject and images with simulated dementia are displayed in Figure 5.4. We first observe that all the models are able to reconstruct the input image of a healthy subject. We can recognize the shape of the brain, the areas with high metabolism (gray matter) and the others with a lower metabolism (white matter, ventricles). The VAE-IAF (Kingma et al., 2016) reconstruction has an artifact in the precuneus, which appears as a spherical

TABLE 5.7: Reconstruction metrics obtained between pseudo-healthy reconstructions obtained from the simulated images of Test AD 30 and the ground truth images (mean \pm std computed over images from the test set)

Models	SSIM \uparrow	MSE ($\times 10^{-3}$) \downarrow	PSNR \uparrow
AE	0.876 \pm 0.025	2.067 \pm 0.643	26.99 \pm 1.03
Adv. AE (Makhzani et al., 2015)	0.878 \pm 0.027	2.021 \pm 0.620	27.07 \pm 0.99
β -TC VAE (Chen et al., 2018a)	0.871 \pm 0.024	2.094 \pm 0.605	26.92 \pm 1.02
β -VAE (Higgins et al., 2017)	0.873 \pm 0.026	2.107 \pm 0.678	26.90 \pm 1.00
Dis. β -VAE (Burgess et al., 2018)	0.874 \pm 0.023	2.158 \pm 0.667	26.80 \pm 1.02
FactorVAE (Kim et al., 2018)	0.873 \pm 0.024	2.157 \pm 0.603	26.78 \pm 0.94
HVAE (Caterini et al., 2018)	0.876 \pm 0.023	2.071 \pm 0.646	26.98 \pm 1.04
InfoVAE (Zhao et al., 2019)	0.878 \pm 0.022	2.044 \pm 0.595	27.02 \pm 0.97
IWAE (Burda et al., 2016)	0.864 \pm 0.027	2.265 \pm 0.571	26.55 \pm 0.91
RAE-GP (Ghosh et al., 2019)	0.878 \pm 0.022	2.118 \pm 0.690	26.88 \pm 0.99
RAE- ℓ^2 (Ghosh et al., 2019)	0.877 \pm 0.023	1.997 \pm 0.564	27.12 \pm 0.99
VAE (Kingma et al., 2014)	0.870 \pm 0.027	2.075 \pm 0.589	26.95 \pm 0.95
VAEGAN (Larsen et al., 2016)	0.854 \pm 0.027	2.650 \pm 0.662	25.88 \pm 0.97
VAE-IAF (Kingma et al., 2016)	0.842 \pm 0.025	2.322 \pm 0.735	26.47 \pm 0.97
VAE LinNF (Rezende et al., 2015)	0.876 \pm 0.022	2.179 \pm 0.614	26.74 \pm 0.97
VQVAE (Van Den Oord et al., 2017)	0.878 \pm 0.025	2.089 \pm 0.596	26.92 \pm 0.97
WAE (Tolstikhin et al., 2018)	0.877 \pm 0.025	2.087 \pm 0.650	26.95 \pm 1.04

hypermetabolism. This probably explains why the average SSIM is lower for the VAE-IAF (Kingma et al., 2016) than for other models. We can also see that the VAEGAN (Larsen et al., 2016) tends to reconstruct the image with a higher average intensity, as shown by the fact that the difference map is mostly negative (meaning that the reconstruction’s voxel values are superior to the input’s voxel values).

When reconstructing images with different degrees of simulated AD, we observe that all the models are able to reconstruct images that are visibly healthy by correcting the hypometabolism simulated. On the difference maps, we can recognize the mask used for the simulation as an anomaly, meaning that the model is able to reconstruct pseudo-healthy images. From this qualitative analysis, the models that seem to perform the best in terms of anomaly detection are the VAE-IAF (Kingma et al., 2016) (excluding the fact that it reconstructs an artifact), the β -VAE (Higgins et al., 2017), the disentangled β -VAE (Burgess et al., 2018) and the HVAE (Caterini et al., 2018), at least for images with low (AD 15) and medium (AD 30) severity. It is indeed possible to better distinguish the abnormal area in both hemispheres on the difference maps, and the reconstruction errors do not hide the anomaly.

Additional examples of pseudo-healthy reconstructions obtained for different subjects and different simulated dementias are displayed in Appendix G.

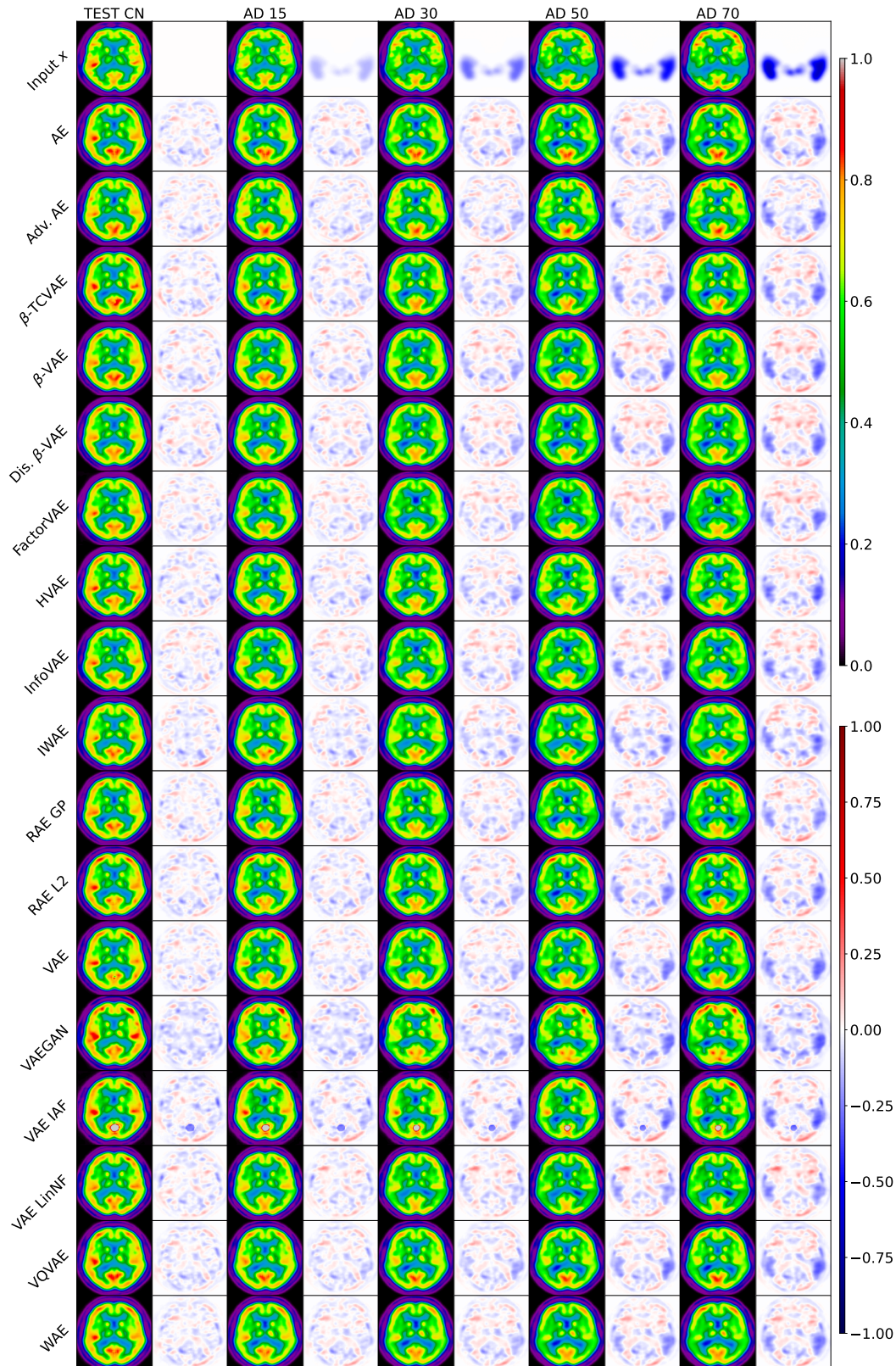


FIGURE 5.4: Examples of reconstructions obtained with the different VAE variants from the original image of a cognitively normal subject (images of the first column, Test CN) and from the same subject with AD simulated at different intensity degrees (AD 15, AD 30, AD 50 and AD 70). The first row shows the input image in odd columns and the mask of the simulated disease in even columns when the input is a simulated image. All the other rows are the pseudo-healthy reconstructions of the models in odd columns and the difference between the pseudo-healthy reconstruction and the input in even columns.

5.5.4 Quantitative evaluation with the healthiness metric

After qualitatively analyzing the pseudo-healthy reconstructions, we computed the healthiness score defined in Section 5.2 for the different simulated test sets.

Figure 5.5 displays the distribution of the healthiness score for the ground truth (i.e., the images of healthy subjects), the images simulating AD with 30% hypometabolism (AD 30) and the reconstructions obtained for the different models from the AD 30 images. As expected, the healthiness of the ground truth is between 1.0 and 1.08, and it drops to between 0.83 and 0.90 when simulating AD with a hypometabolism intensity of 30%. Studying the healthiness of the pseudo-healthy reconstruction for each model, we first observe that all the models are able to reconstruct images that are healthier than the simulated input as the healthiness of the reconstructions (around 1) is superior to the healthiness of the simulated images they were reconstructed from (around 0.87). We can observe that three models seem to perform slightly better than the others, namely the β -VAE (Higgins et al., 2017), the disentangled β -VAE (Burgess et al., 2018) and the VAE-IAF (Kingma et al., 2016) with a healthiness between 0.97 and 1.04 for the first two and 0.96 and 1.03 for the third. On the other hand, the VAEGAN (Larsen et al., 2016) appears to be the model with the worst performance (with a healthiness between 0.93 and 1.0), followed by the FactorVAE (Kim et al., 2018) and the RAE-GP (Ghosh et al., 2019) (which have a healthiness score between 0.94 and 1.01).

These results are consistent with the qualitative analysis, as we observed that the β -VAE (Higgins et al., 2017), disentangled β -VAE (Burgess et al., 2018) and VAE-IAF (Kingma et al., 2016) seemed to better highlight the simulated anomalies, while the VAEGAN’s (Larsen et al., 2016) poor reconstructions tended to hide the anomalies.

We also analyzed the impact of the severity of the simulated disease on the healthiness metric. Figure 5.6 displays the evolution of the healthiness for all the models with increasing severity of simulated AD from 5% to 70%. We notice that all the models reconstruct images that are decreasingly healthy according to this metric when increasing the severity of the simulated disease. As in the previous experiment, the β -VAE (Higgins et al., 2017) and disentangled β -VAE perform the best for high hypometabolism, followed by the VAE-IAF (Kingma et al., 2016). The VAEGAN (Larsen et al., 2016) and RAE-GP (Ghosh et al., 2019) have the worst performance. However, the healthiness of the reconstruction remains above the one of simulated data, which means that all the models can reconstruct pseudo-healthy images.

Figure 5.7 displays for various dementia subtypes (PCA, bvFTD, lvPPA, svPPA and nfvPPA simulated at 30%) the distribution of the healthiness computed for the ground truth, the simulated images and the images reconstructed with all the models. All the models have very similar performance with a healthiness between 0.95 and 1 when simulating PCA, between 0.9 and 1.1 for bvFTD, between 0.96 and 1.6 for lvPPA, between 0.68 and 0.87 for svPPA, and between 1.0 and 1.1 for nfvPPA. As for AD, the VAEGAN’s (Larsen et al., 2016) performance is slightly lower than that of the other models, and the β -VAE (Higgins et al., 2017) and disentangled β -VAE (Burgess et al., 2018) seem to perform slightly better than the average. We notice that the healthiness of the ground truth, derived from the images of CN subjects, depend on the simulated dementia, given the different masks used

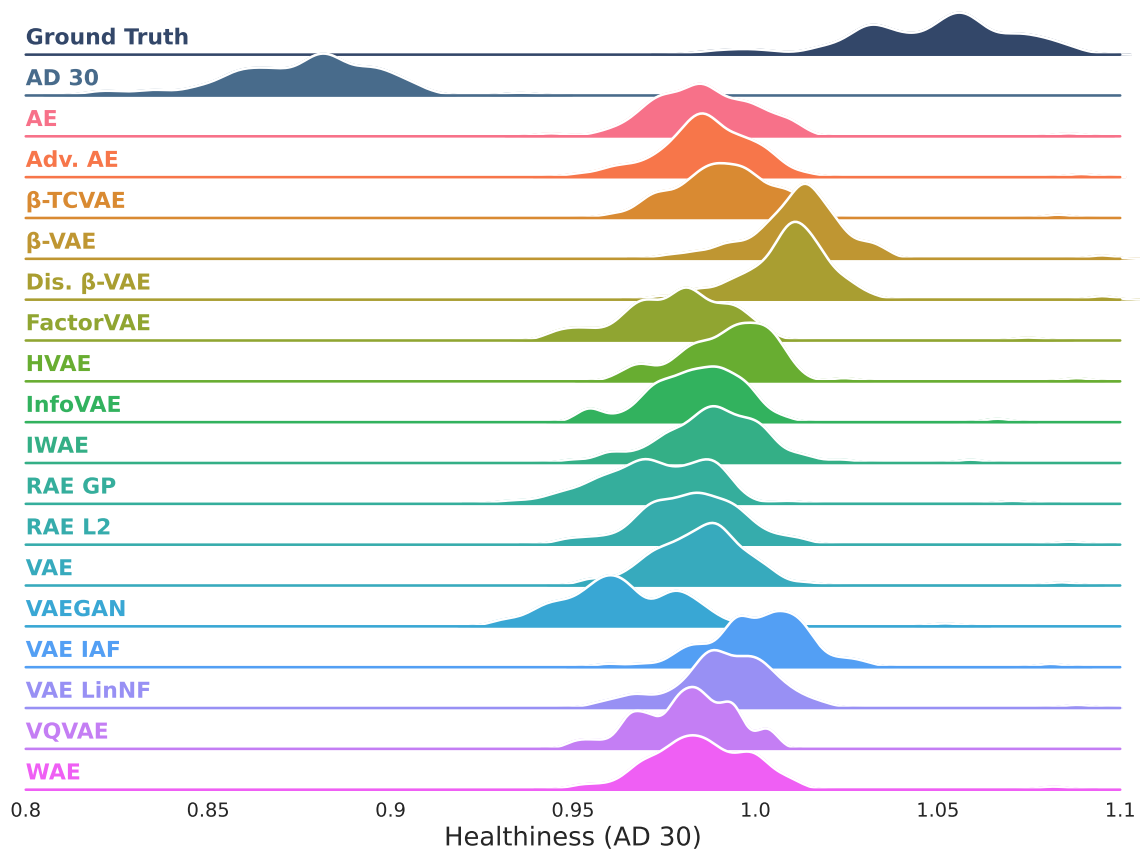


FIGURE 5.5: Ridgeline plot showing the distribution of the healthiness metric for images from Test AD 30. The first row corresponds to the healthiness of the ground truth, the second row to the healthiness of the images simulating AD with 30% hypometabolism used as input, and the remaining rows to the healthiness of the pseudo-healthy reconstructions obtained with the VAE models.

for computation. For example, in the case of svPPA, the ground truth’s healthiness tends to be lower than that of AD (falling between 0.67 and 0.92). This difference comes from the mask’s location in the temporal pole for svPPA, where FDG uptake is naturally lower even in healthy images, in contrast to other regions (Solal et al., 2024a).

To push further the comparison of the models, we jointly analyzed their performance in terms of reconstruction accuracy and healthiness. Figure 5.8 displays a joint density plot of the SSIM and healthiness metric computed for pseudo-healthy reconstructions obtained from images simulating AD at 30% of hypometabolism. We carefully selected four models that we compare to the VAE: the VAEGAN (Larsen et al., 2016) that performs worse than most models, both in terms of reconstruction accuracy and healthiness, the RAE-GP (Ghosh et al., 2019) that has a good reconstruction but a low healthiness performance, the VAE-IAF (Kingma et al., 2016) that has the worst reconstruction accuracy but a good healthiness, and finally the β -VAE (Higgins et al., 2017) that has both good reconstruction and healthiness performance. This analysis confirms that, among the selected models, the β -VAE is the one that performs the best, and the VAEGAN is the one performing the worst.

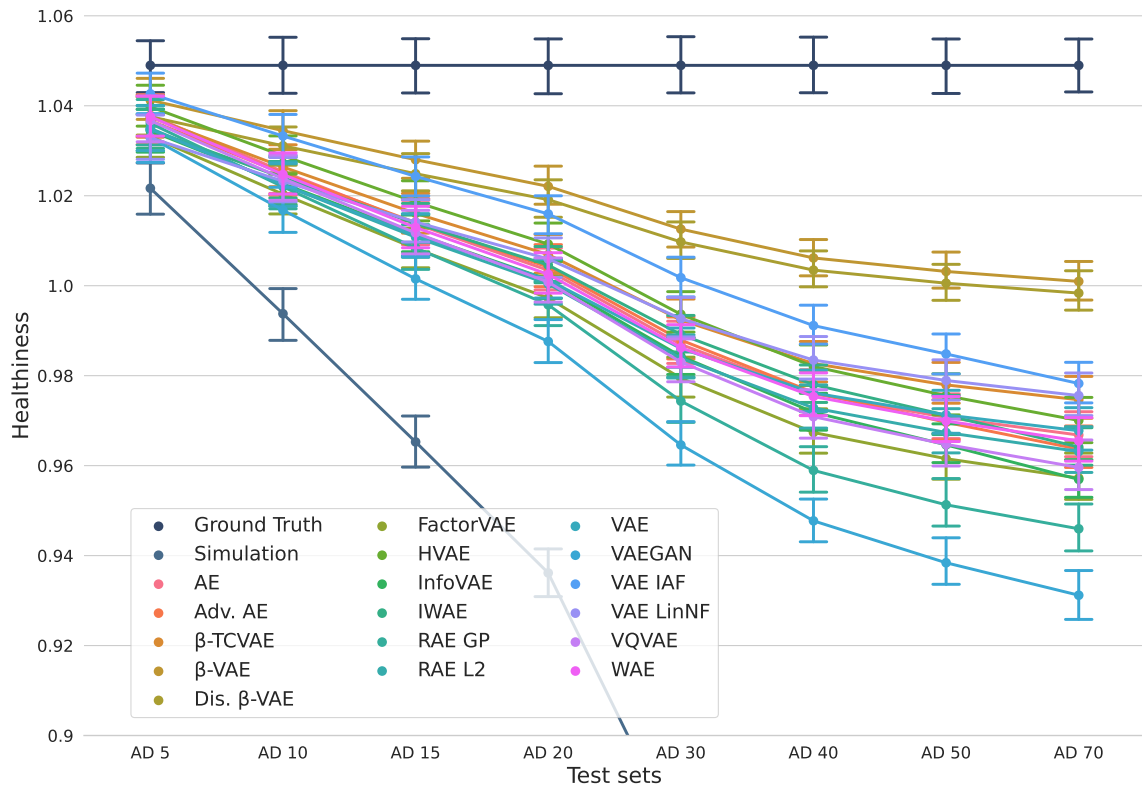


FIGURE 5.6: Healthiness metric depending on the severity of the anomalies simulated. The healthiness of the ground truth, which is constant, is displayed as reference. The healthiness of the images simulating AD rapidly drops with the hypometabolism increasing from 5% to 70%. The other curves correspond to the healthiness of the reconstructions obtained with the VAE models. Each dot represents the mean value of the healthiness, and the error bar represents the standard deviation.

5.5.5 Qualitative analysis of the pseudo-healthy reconstructions obtained from real AD patients

Even though no ground truth is available, it is important to analyze the behavior of the VAE models on data from real patients, here with AD. Figure 5.9 displays examples of pseudo-healthy reconstructions obtained from the image of an AD patient. This patient presents a typical hypometabolism in the parietal and temporal lobes, which is detected by all the models. However, we also observe for all the models what appears as hypermetabolism in the frontal lobe, which is not typical of AD and probably results from reconstruction inaccuracies as this tendency was also visible for the CN subject displayed in Figure 5.4, better seen in Appendix G, Figure G.2.

5.6 Discussion

This benchmark assessed the ability of 20 VAE models to reconstruct pseudo-healthy 3D brain FDG PET images for anomaly detection. We first searched for the best encoder-decoder architecture for the vanilla VAE. We then optimized the hyper-parameters of all the VAE-based models. After discarding the models with low reconstruction performance, we trained the 17 remaining ones on all the splits of the cross-validation to select the best split

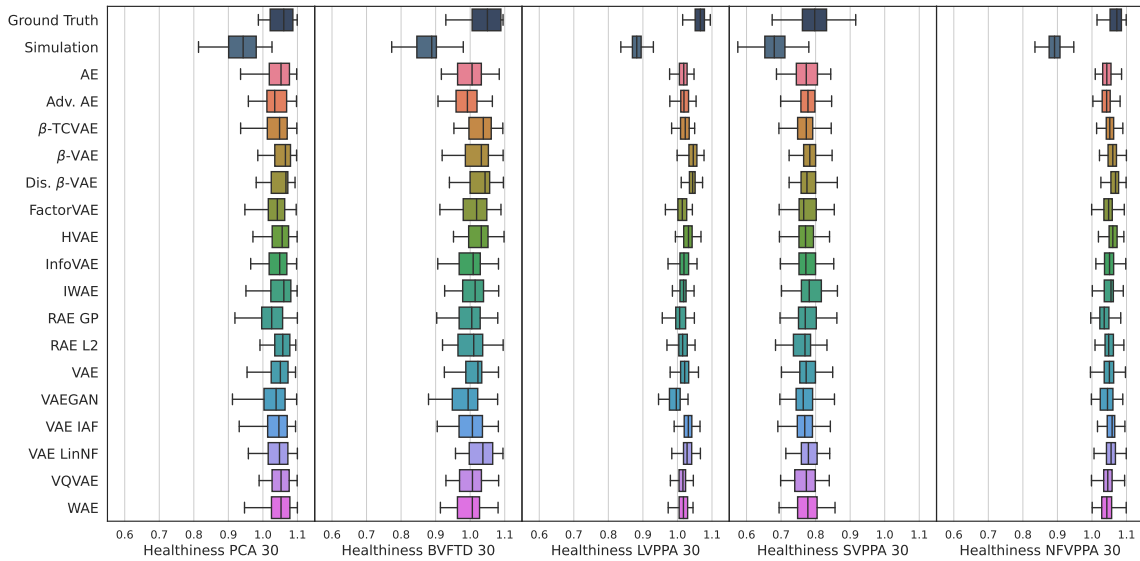


FIGURE 5.7: Distribution of the healthiness metric depending on the dementia simulated at 30% hypometabolism: PCA, bvFTD, lvPPA, svPPA and nfvPPA. Each box plot displays the median, the lower and upper quartiles and the minimum and maximum (excluding outliers) of the healthiness. The first box (top row) shows the healthiness of the ground truth, the second one the healthiness of the simulated images used as input and the remaining ones the healthiness of the pseudo-healthy reconstructions obtained with the VAE models.

for each model. Finally, we compared the trained models using conventional reconstruction metrics, as well as the simulation framework paired with the healthiness metric we previously proposed in Chapter 3.

5.6.1 Model selection

We performed an extensive random search to define the optimal encoder-decoder architecture for the vanilla VAE. 200 models were trained for a total of approximately 5000 GPU hours. The architecture we obtained is similar to what we could implement following examples and guidelines found in the literature with the objective to obtain a small model that allows fitting 3D high resolution images in the GPU memory: the encoder and decoder are symmetric, they are composed of five blocks, each containing only one 3D convolution layer, a batch normalization and a swish activation (Vahdat et al., 2020). This architecture is for instance very similar to the one we tested in Chapter 2, Section 2.3. Having a small encoder and decoder proves especially advantageous in this benchmark for models with heightened memory-requirements, like the VAEGAN (Larsen et al., 2016) (due to its extra discriminator network), the IWAE (Burda et al., 2016) (since it uses several samples from the latent space), and the VAE-IAF (Kingma et al., 2016) (since it has extra layers for the auto-regressive flows in the latent space). This architecture was used for all the VAE-based models. Whilst optimizing the architecture for the vanilla VAE may give this model an advantage, it was not conceivable for us, given our computational resources, to optimize the encoder-decoder architecture separately for all the models.

To optimize the hyper-parameters of each VAE variant, 324 models were trained for a total of approximately 18,000 hours of GPU use. At this stage we removed three models from the study, as they led to poor reconstructions in comparison with the others: the SVAE

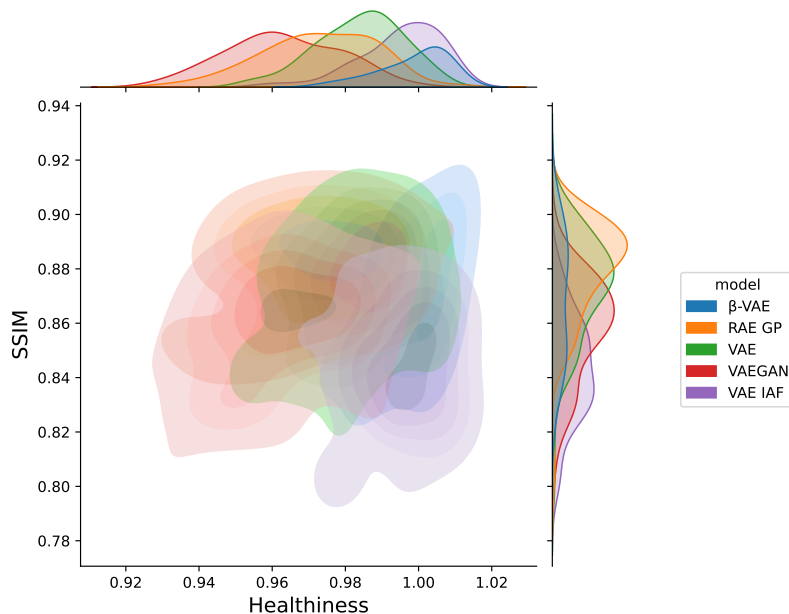


FIGURE 5.8: Joint density plot of the healthiness metric (x-axis) and SSIM (y-axis) computed for pseudo-healthy reconstructions obtained from images from Test AD 30. A good model should appear on the top right part of the graph (high SSIM and healthiness close to 1).

(Davidson et al., 2018), the MS-SSIM VAE (Snell et al., 2017) and the VAMP (Tomczak et al., 2018). For each remaining model, it was possible to find a set of hyper-parameters that led to good reconstruction performance.

After training the models with the selected hyper-parameters on the six splits of the cross-validation, we selected the best split for each of them. We observed that splits 2 and 3 gave the best results for 13 models out of 17 (Table 5.5). This can be explained by the fact that the cross-validation was not stratified, and so the validation set may not be representative of the training population for some of the splits (Table 1.1). This may have biased the selection of the hyper-parameters since some models were not trained on splits 2 and 3 when randomly selecting three folds out of six, which may result in underestimated performance for these configurations. However, it would have been too long to train all the configurations on all the splits; and we appraise that we still found a satisfying combination of parameters with respect to the reconstruction metrics, even though it may not be the best one.

All the selection steps were based on the validation sets, potentially leading to overfitting on these validation sets. Performing a 6-fold cross-validation and randomly selecting the splits reduced this risk.

5.6.2 Model evaluation

To evaluate the different models, we applied the evaluation procedure presented in Chapter 3. This evaluation consists in two main steps: first measuring the reconstruction performance of the models using reconstruction metrics for images of healthy subjects, and then using simulated data in order to evaluate the ability of the models to reconstruct pseudo-healthy images (i.e. whether the reconstructions appear healthy).

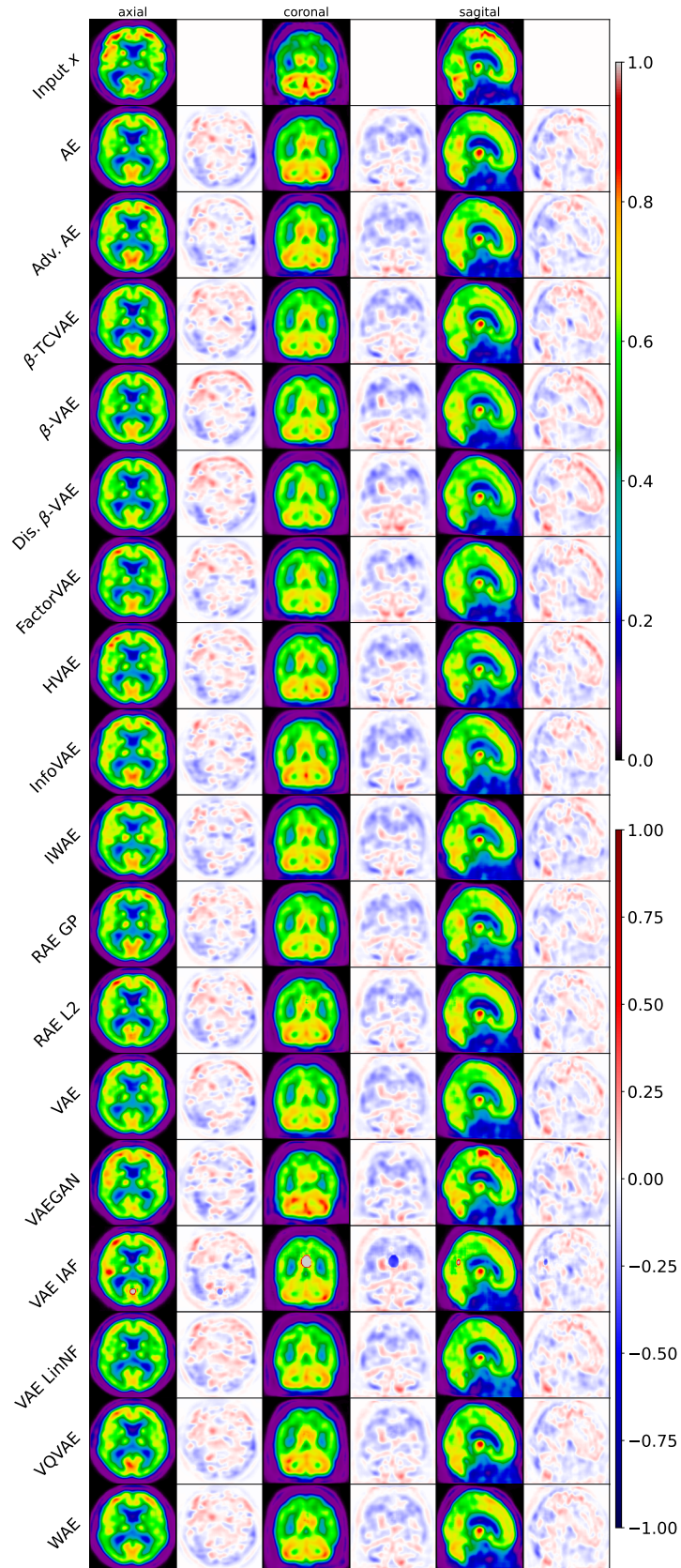


FIGURE 5.9: Example of reconstructions obtained from the different VAE variants on an AD patient (on axial, coronal and sagittal slices). The first row shows the input image in odd columns. The rows below are the pseudo-healthy reconstruction of the models in odd columns, and the difference between the pseudo-healthy reconstruction and the input in even columns.

In terms of reconstruction metrics, for the images of healthy subjects (Table 5.6), all the trained models led to similar performance, with the nine best models having an average SSIM above 0.88, seven models having an average SSIM between 0.86 and 0.88, and only one having an average SSIM below 0.86. All the models were able to reconstruct realistic brain images, as shown in Figure 5.4. Many models performed better than the vanilla VAE according to both the MSE and the SSIM, but not substantially. The reconstruction metrics computed between the reconstructions obtained from simulated images and the ground truth (Table 5.7) show that the reconstructed images are quite similar to their healthy target (i.e., the original images used to simulate hypometabolic images), indicating that the reconstruction capacity of the models is not affected when using images with anomalies as input.

In terms of healthiness, on images simulating AD, it appears that the β -VAE (Higgins et al., 2017), the disentangled β -VAE (Burgess et al., 2018) and the VAE-IAF (Kingma et al., 2016) performed better than the other VAEs, whereas the VAEGAN (Larsen et al., 2016) and the RAE-GP (Ghosh et al., 2019) gave the worst results (Figure 5.5). Interestingly, the healthiness distribution of the ground truth images is multi-modal, and so is logically the distribution of simulated images. This is also the case of the healthiness distributions of the reconstruction for most of the models, especially for the FactorVAE (Kim et al., 2018) and the VQVAE (Van Den Oord et al., 2017) for which we can properly recognize the shape of the distribution. However, the reconstructions of the two best performing models, the β -VAE (Higgins et al., 2017) and the disentangled β -VAE (Burgess et al., 2018), have a uni-modal healthiness distribution, potentially explaining why they are not the best performing models in terms of reconstruction metrics. This may be explained by the fact that, in both cases, we set $\beta = 10 (\gg 1)$, giving more weight to the KL term than the reconstruction term in the loss.

As illustrated in Figure 5.7, most of the models were able to reconstruct images of healthy appearance also for dementia subtypes different from AD. As for AD, the best performing models were the β -VAE (Higgins et al., 2017) and the disentangled β -VAE (Burgess et al., 2018). The VAE LinNF (Rezende et al., 2015) and the HVAE (Caterini et al., 2018) also seem to have a higher healthiness than the other models. On the other hand, the VAEGAN (Larsen et al., 2016), the RAE-GP (Ghosh et al., 2019) and the FactorVAE (Kim et al., 2018) were the models with the lowest performance. We could further see that the healthiness metric was not optimal for all the dementia subtypes. For instance, for PCA and svPPA, the healthiness of simulated images was not substantially different from that of the ground truth. Nevertheless, we observed that the healthiness of the reconstruction was close to that of the ground truth and higher than that of simulated data, which is sufficient to assess the healthiness of reconstructed images.

In general, we observed that all the models were able to reconstruct images with a healthiness substantially above the healthiness of simulated images, regardless of the kind of simulated anomalies, and almost equal to the healthiness of the ground truth, indicating that the reconstructions are indeed healthy looking.

To push further the model comparison, we jointly analyzed reconstruction and healthiness metrics (Figure 5.8). The RAE-GP (Ghosh et al., 2019) was the model with the

best reconstruction, but was ranked among the worst in terms of healthiness. Even though the reconstructions look healthy when compared to the simulated input, it means that the RAE-GP (Ghosh et al., 2019) did not learn the healthy image distribution as well as other models, but rather learned to reconstruct the input as is. On the contrary, the VAE-IAF (Kingma et al., 2016) was the model with the worst reconstruction, but was among the best in terms of healthiness. This can be explained by the presence of a reconstruction artifact that impacts the reconstruction score. The β -VAE (Higgins et al., 2017) was the best model in terms of healthiness and was average in terms of reconstruction, and the VAEGAN (Larsen et al., 2016) under-performed both in terms of reconstruction and healthiness.

A surprising result highlighted by the benchmark is that the simple AE performed well in comparison with more complex models, especially according to the healthiness for simulated data. Although it was expected that this model would be able to reconstruct images of healthy subjects, there was no certainty that the AE would be able to reconstruct healthy looking images from simulated images, especially when simulating severe hypometabolism (50% and more). It would be interesting to assess the performance of this model when given real images from AD patients.

In our previous study (Hassanally et al., 2023b), we compared a subset of the models that we present here on down-sampled 3D brain FDG PET. Another major difference with the present work is that we had trained the models with default hyper-parameters' values. We observe that some of the models that performed poorly in this previous study, such as the VAEGAN (Larsen et al., 2016) and the VAE LinNF (Rezende et al., 2015), performed much better after searching for optimal hyper-parameters, whereas the VAMP (Tomczak et al., 2018) and the MS-SSIM VAE (Snell et al., 2017) still perform poorly even after hyper-parameters tuning. Even though not surprising, this highlights the benefit and need of optimizing each model, even though this step does not guarantee reaching good performance.

5.6.3 Limitations and perspectives

The main limitation of our work is the absence of ground truth masks for the anomalies we aim to detect. However, this benchmark proved the utility of the simulation-based evaluation framework we previously introduced in Chapter 3, which allowed evaluating the pseudo-healthy images reconstructed by the models using pairs of abnormal and healthy images for the same subjects, for different dementia subtypes and severity degrees. The evaluation framework also introduced the healthiness metric that automatically quantifies whether a reconstruction is pseudo-healthy. This framework is a first evaluation step that does not require the involvement of a clinician: it would indeed be impossible to ask a clinician to rate the reconstructions of 20 different models. However, a limitation is that we do not really evaluate how well each model is able to detect anomalies using these pseudo-healthy reconstructions. A solution would be to use the anomaly score proposed in Chapter 3, Section 3.2.4, or abnormality maps using Z-scores (Solal et al., 2024a). A comprehensive evaluation would ultimately require using real images with real anomalies and having the results reviewed by clinicians.

The current evaluation is limited to the quality of the reconstructions and their degree of healthiness, and does not directly assess how well each model learned the healthy distribution. An interesting work would be to compare the latent distributions of the trained models to assess whether the posterior learned by the different models is the same for images from healthy and diseased subjects. This could be done using the simulation framework of Chapter 3 and comparing the latent representations of both the original and simulated images. It would help us to understand the performance difference between the various VAEs, and may give us some ideas to improve them.

The models were compared on a single modality, FDG PET. It would be further interesting to test these models on structural MRI, which have different characteristics, such as sharp structures. This would also allow us to compare the performance of these VAE variants with other approaches in the literature, as many have been developed to detect lesions in structural MRI. Similarly, it would be interesting to include other VAE models that performed well in computer vision such as Hierarchical VAEs (Sønderby et al., 2016; Ranganath et al., 2016; Vahdat et al., 2020; Maaløe et al., 2019), that have already successfully applied to medical imaging (Dorent et al., 2023), or compare VAEs to other generative models such as GANs and diffusion models.

5.6.4 Reproducibility

In order to make this study as reproducible as possible (Colliot et al., 2023; Colliot et al., 2024), we tried to follow the guidelines of the MICCAI reproducibility checklist⁴:

- the publicly available dataset and final cohort we work with is mainly described in Section 1.2 with details of the preprocessing and data selection steps presented in Section 1.2.3. We provide a summary of participant demographics for the train, validation and test splits in Table 1.1;
- the architecture choices for the VAE and the impact of those choices are detailed in Section 5.4.1 and Appendix F;
- the VAE variants are described in Appendix E with the range of hyper-parameters considered for each of them;
- the training protocol and the method to tune and select hyper-parameters are described in Section 5.4.2 and Appendix E;
- we also provided a clear definition of the specific evaluation metrics and statistics used to report results in Chapter 3, Section 3.2.1 and Section 3.2.2.

Moreover, most of the code that we used is available in ClinicaDL (Thibeau-Sutre et al., 2022b), an open-source software that is developed to enable reproducible deep learning studies in neuroimaging. Pipelines are available to perform the following steps:

- selecting subjects from a neuroimaging dataset,
- rigorously separating data into independent training and testing sets,

⁴<https://miccai2021.org/files/downloads/MICCAI2021-Reproducibility-Checklist.pdf>

- rigorously splitting the training set using a cross-validation,
- launching random searches to optimize architecture and hyper-parameters,
- easily training VAE-based models on neuroimages,
- constructing new test sets by generating simulated data using the method described in Chapter 3,
- reconstructing pseudo-healthy images from trained models for the tests sets and computing the reconstruction metrics used in evaluation.

All the VAE-based models are implemented in Pythae (Chadebec et al., 2022), an open-source Python library that aims at unifying the implementation of VAE-based models, and facilitating benchmarks. Moreover, all the preprocessing pipelines are available in Clinica (Routier et al., 2021), an open-source software for reproducible processing of neuroimaging datasets. Clinica has been used to:

- curate and organize the ADNI dataset following a community standard, namely the brain imaging data structure (BIDS) (Gorgolewski et al., 2016),
- perform linear registration and intensity normalization of the FDG PET scans (`pet-linear` pipeline).

Finally, all the code for random searches, model training and evaluation is available in the following repository: https://github.com/ravih18/UAD_VAE_benchmark. This repository includes dependencies and software versions used.

5.7 Conclusion

In summary, we presented in this chapter a benchmark of twenty VAE-based models for the unsupervised detection of dementia related anomalies in 3D brain FDG PET. The aim was to introduce the use of recent VAE variants with medical imaging data of high dimension and compare their performance. We proposed a random search method to find the optimal architecture for the vanilla VAE, as well as a random search method to tune the hyper-parameters of the implemented models.

We observed that 17 of the 20 models had a good reconstruction quality. Using our previously proposed evaluation framework presented in Chapter 3, we showed that the 17 models were able to reconstruct pseudo-healthy images when fed with simulated abnormal images. By simulating AD with varying intensity and dementia other than AD, we also showed that these models were able to generalize to anomalies of different shapes, localizations and intensities. If no model clearly outperformed the others, the β -VAE (Higgins et al., 2017) and disentangled β -VAE (Burgess et al., 2018) slightly outperformed the other models, while remaining easy to tune and not being noticeably computationally costly.

Even if it is recognized that VAEs generate blurry images, all these experiments showed that most of the models were able to reconstruct good quality pseudo-healthy 3D FDG

PET. The VAE variants showed similar performance and did not systematically outperform the vanilla VAE (or even the simple AE).

Finally, we can conclude that most VAEs are well suited for pseudo-healthy reconstruction of brain FDG PET images

Chapter 6

Reproducible neuroimaging processing with deep learning with Clinica and ClinicaDL open-source software packages

This chapter is a compilation of two journal articles and one conference proceeding. The first one has been published in *Frontiers in Neuroinformatics*.

- **Title:** Clinica: an open-source software platform for reproducible clinical neuroscience studies
- **Authors:** Alexandre Routier, Ninon Burgos, Mauricio Díaz, Michael Bacci, Simona Bottani, Omar El-Rifai, Sabrina Fontanella, Pietro Gori, Jérémy Guillon, Alexis Guyot, **Ravi Hassanaly**, Thomas Jacquemont, Pascal Lu, Arnaud Marcoux, Tristan Moreau, Jorge Samper-González, Marc Teichmann, Elina Thibeau-Sutre, Ghislain Vaillant, Junhao Wen, Adam Wild, Marie-Odile Habert, Stanley Durrleman, Olivier Colliot
- **DOI:** [10.3389/fninf.2021.689675](https://doi.org/10.3389/fninf.2021.689675)
- **Contributions:** Integration of `pet-linear` pipeline. Small contributions.

The second one has been published in *Computer Methods and Programs in Biomedicine*

- **Title:** ClinicaDL: an open-source deep learning software for reproducible neuroimaging processing
- **Authors:** Elina Thibeau-Sutre, Mauricio Diaz, **Ravi Hassanaly**, Alexandre Routier, Didier Dormont, Olivier Colliot, Ninon Burgos
- **DOI:** [10.1016/j.cmpb.2022.106818](https://doi.org/10.1016/j.cmpb.2022.106818)
- **Contributions:** Detailed in Section [6.2.6](#)

The third one has been published in the proceedings of SPIE Medical Imaging 2024: Image Processing conference.

- **Title:** Recent advances in the open-source ClinicaDL software for reproducible neuroimaging with deep learning
- **Authors:** **Ravi Hassanaly**, Camille Brianceau, Mauricio Diaz, Sophie Loizillon, Elina Thibeau-Sutre, Nathan Cassereau, Olivier Colliot, Ninon Burgos

In this chapter, we present the software contributions that have been made during this PhD thesis. This includes contributions to two open-source projects: Clinica and ClinicaDL. Clinica and ClinicaDL are two open-source packages that aim to enhance the reproducibility of neuroimaging studies. An overview of the Clinica and ClinicaDL workflow is available in Figure 6.1. Additionally, all the code developed for the different experiments performed during this thesis have been released under the form of GitHub repositories for reproducibility purposes.

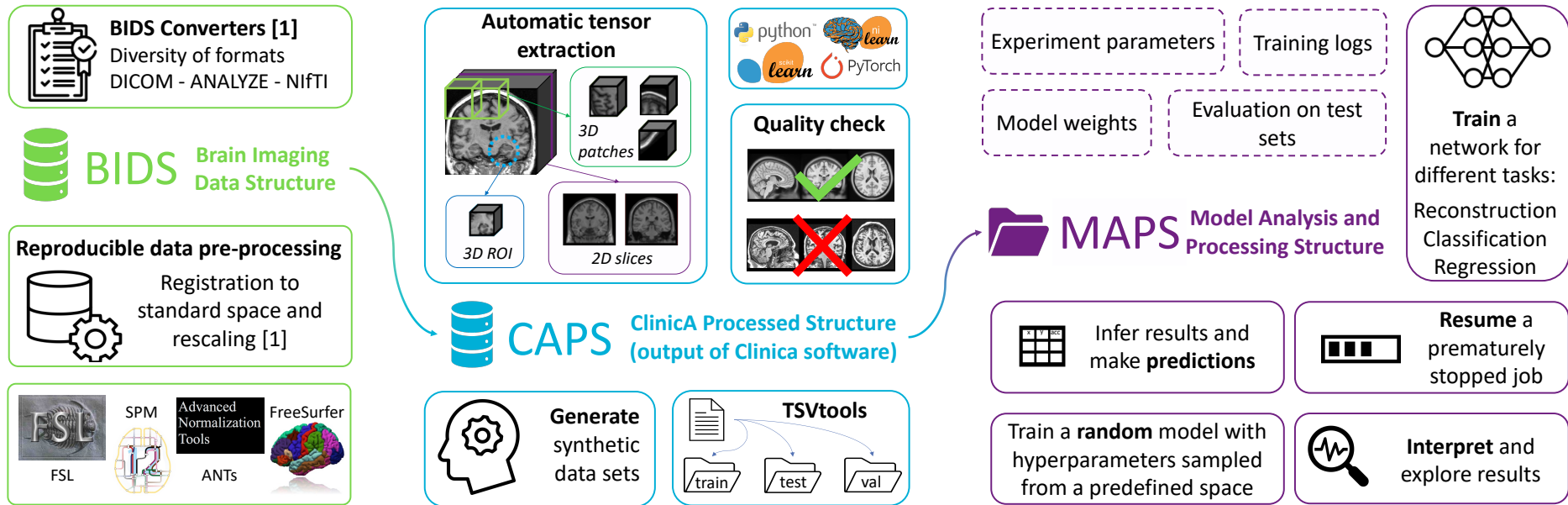


FIGURE 6.1: Overview of the Clinica and ClinicaDL workflows. Data preprocessed by Clinica are stored in a BIDS-like structure called CAPS. ClinicaDL further allows preparing data experiments and creating training, validation and test sets. Various models can then be trained and results are stored in a unified structure called MAPS. Finally, ClinicaDL allows running inference, saving predictions and interpreting the results.

6.1 Clinica

Clinica is an open-source software platform designed to make clinical neuroscience studies easier and more reproducible (Routier et al., 2021). Clinica aims for researchers to (i) spend less time on data management and processing, (ii) perform reproducible evaluations of their methods, and (iii) easily share data and results within their institution and with external collaborators. The core of Clinica is a set of automatic pipelines for processing and analysis of multi-modal neuroimaging data (currently, T1-weighted MRI, diffusion MRI, and PET data), as well as tools for statistics and machine learning. It relies on the brain imaging data structure (BIDS) for the organization of raw neuroimaging datasets and on established tools written by the community to build its pipelines. It also provides converters of public neuroimaging datasets to BIDS.

6.1.1 Data structures

Brain Imaging Data Structure (BIDS)

When dealing with multiple datasets, it is difficult to automate the execution of neuroimaging pipelines, since their organization may vary from each other or even within each individual dataset. If we consider neuroimaging datasets involving many participants, the lack of a clear structure will necessitate a large amount of time to curate these databases and make them easily usable. Besides, large databases are often associated with database management systems, which involve additional technical and financial resources to be maintained.

The brain imaging data structure (BIDS)(Gorgolewski et al., 2016) is a community standard enabling the storage of multiple neuroimaging modalities and behavioral data. The BIDS standard provides a unified structure and makes easier the development and distribution of code that uses neuroimaging datasets. Moreover, the BIDS format is based on a file hierarchy rather than on a database management system, thus avoiding the installation and maintenance of additional software. As a result, BIDS can be easily deployed in any environment. The specification is intentionally based on simple file formats and folder structures to reflect current laboratory practices, which makes it accessible to a wide range of scientists coming from different backgrounds.

For these reasons, Clinica has also adopted this standard, and expects input data that are BIDS-compliant for the execution of pipelines.

Clinica Processed Structure (CAPS)

Clinica has its own specifications for hierarchical storage of processed data, called CAPS¹ (Clinica Processed Structure). The idea is to include in a single folder all the results generated by the different pipelines, and to organize the data following the main patterns of the BIDS specification. CAPS folders are kept separate from the raw data. Indeed, when processing data, it is very common to have the raw dataset located on a separated storage or read-only storage, while ongoing processed data are located on a separate location or on a faster data storage.

¹<https://aramislab.paris.inria.fr/clinica/docs/public/latest/CAPS/Introduction/>

Processed data include image-valued scalar fields (e.g., segmentation labels, tissue maps), meshes, mesh-valued scalar fields (e.g., cortical thickness maps), deformation fields, scalar outputs (e.g., volumes, regional averages), etc.

Of note, there exists an ongoing initiative called BIDS-derivatives that aims to provide a BIDS standard for processed data. However, CAPS specification have been written before the start of the BIDS-derivatives, which explains why Clinica does not use the latter. Moreover, in their current state, several outputs needed by Clinica are not covered or well-adapted.

6.1.2 Main functionalities

Clinica provides tools to curate several publicly available neuroimaging datasets and automatically convert them into the BIDS standardized data structure. For all converters, the user only needs to download the dataset. All subsequent conversion steps are performed automatically (no user intervention is required) and use parallelization for faster processing. For further details, the reader can refer to (Samper-González et al., 2018). Clinica currently provides converters for the following studies: Alzheimer’s Disease Neuroimaging Initiative², the Australian Imaging, Biomarker & Lifestyle Flagship Study of Ageing³, the Open Access Series of Imaging Studies⁴, the frontotemporal lobar degeneration neuroimaging initiative⁵ and the UKbiobank⁶.

Additionally, Clinica provides processing pipelines that involve the combination of different software packages. It currently relies on FreeSurfer (Fischl, 2012), FSL (Jenkinson et al., 2012), SPM (Friston, 2003), Advanced Normalization Tools (ANTs)⁷ (Avants et al., 2014), MRtrix38 (Tournier et al., 2012), and the PET Partial Volume Correction (PETPVC) toolbox⁹ (Thomas et al., 2016). The pipelines are written using Nipype (Gorgolewski et al., 2011). Features extracted with the different pipelines can be used as inputs to statistical analysis, which relies on SPM (Friston, 2003) and SurfStat10 (Worsley et al., 2009), or machine learning analysis, which relies on scikit-learn (Pedregosa et al., 2011). The pipelines are described in Figure 6.2.

6.1.3 Integration of the pet-linear pipeline in Clinica

The `pet-linear` pipeline performs a spatial normalization to the MNI space and intensity normalization of PET images. The first step of the pipeline is an affine registration to the MNI152NLin2009cSym template (Fonov et al., 2009; Fonov et al., 2011) in MNI space with the SyN algorithm (Avants et al., 2008) from the ANTs software package (Avants et al., 2014). Then, the registered image intensity is normalized using the mean intensity in reference regions, resulting in a standardized uptake value ratio (SUVR) map. The normalized imaged is finally cropped to remove the background, before being saved as a NIfTI file in the CAPS. The details of the pipeline are given in Chapter 1, Section 1.2.2.

²<http://adni.loni.usc.edu>

³<https://aibl.csiro.au>

⁴<https://www.oasis-brains.org>

⁵<https://memory.ucsf.edu/research-trials/research/allftd>

⁶<https://www.ukbiobank.ac.uk/>


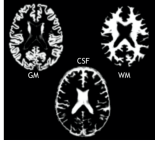
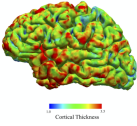
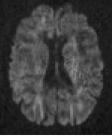
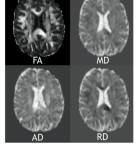
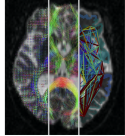
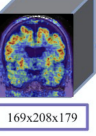
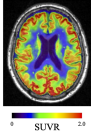
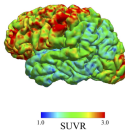
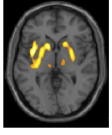
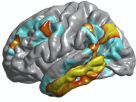
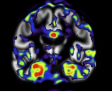

Anatomical MRI	<p>t1-linear Bias field correction, affine registration and cropping Dependencies: ANTs</p> <ul style="list-style-type: none"> • T1 MRI on ICBM 2009c nonlinear symmetric template • Used as input for deeplearning-prepare-data  <p>169x208x179</p>	<p>t1-volume Tissue segmentation (GM, WM, CSF), inter-subject registration using Dartel, spatial normalization to standard space (MNI) Dependencies: SPM, CAT12</p> <ul style="list-style-type: none"> • Voxel-based features (GM, WM, CSF) • Regional features (average GM) using atlases (currently AALZ, AICHA, Hammers, LPBA40, Neuromorphometrics) 	<p>t1-freesurfer t1-freesurfer-longitudinal Cortical surface extraction, segmentation of subcortical structures, cortical thickness estimation, spatial normalization to standard space (FsAverage) Dependencies: FreeSurfer</p> <ul style="list-style-type: none"> • Surface-based features (cortical thickness) • Regional features (average cortical thickness) using atlases (currently Desikan, Destrieux)  <p>Cortical Thickness</p>
Diffusion MRI	<p>dwi-preprocessing Correction of raw DWI data Dependencies: FSL, ANTs, Convert3D</p> <ul style="list-style-type: none"> • EPI correction using phase-difference map fieldmap or T1w ("fieldmap-less") • Prerequisite for dwi-dti or dwi-connectome pipelines 	<p>dwi-dti Extraction of DTI-based measures, normalization to standard space (MNI) Dependencies: FSL, ANTs, MRtrix3</p> <ul style="list-style-type: none"> • Voxel-based features (FA, MD, AD, RD) • Regional features (average FA, MD, AD, RD) using atlases (JHU DTI81, JHUTracts) 	<p>dwi-connectome Tractography & connectome Dependencies: FreeSurfer, FSL, MRtrix</p> <ul style="list-style-type: none"> • Probabilistic tractography • Structural connectome using atlases (currently Desikan, Destrieux) 
PET	<p>pet-linear Affine registration, intensity normalization and cropping Dependencies: ANTs</p> <ul style="list-style-type: none"> • PET on ICBM 2009c nonlinear symmetric template • Used as input for deeplearning-prepare-data  <p>169x208x179</p>	<p>pet-volume Registration to T1 MRI, partial volume correction, spatial normalization to standard space (MNI), intensity normalization Dependencies: SPM, PETPVC, CAT12</p> <ul style="list-style-type: none"> • Voxel-based features (e.g. FDG uptake, amyloid uptake) • Regional features (average FDG, amyloid uptake) using atlases (currently AALZ, AICHA, Hammers, LPBA40, Neuromorphometrics)  <p>SUVR</p>	<p>pet-surface pet-surface-longitudinal Registration to T1 MRI, intensity normalization, partial volume correction, projection to cortical surface, spatial normalization to standard space (FsAverage) Dependencies: FreeSurfer, FSL, SPM, PETPVC</p> <ul style="list-style-type: none"> • Surface-based features (e.g. FDG uptake, amyloid uptake) • Regional features (average cortical thickness) using atlases (currently Desikan, Destrieux)  <p>SUVR</p>
Statistics	<p>statistics-volume Voxel-based mass-univariate analysis with SPM Dependencies: SPM, Matlab</p> <ul style="list-style-type: none"> • Voxel-based features from t1-volume or pet-volume pipelines • Group comparison using GLM 	<p>statistics-surface Surface-based mass-univariate analysis with SurfStat Dependencies: Matlab</p> <ul style="list-style-type: none"> • Surface-based features from t1-freesurfer or pet-surface pipelines • Group comparison or correlations analysis using GLM 	
Machine Learning	<p>machinelearning-prepare-spatial-svm Preparation of T1 MRI and PET data for spatially regularized SVM Dependencies: None</p> <ul style="list-style-type: none"> • Regularization that accounts for the spatial and anatomical structure of neuroimaging data leading to a more regular and anatomically interpretable decision function. • Used as input for machine learning classification 	<p>(No command line interface) Classification based on machine learning Dependencies: None</p> <ul style="list-style-type: none"> • Voxel-based, surface-based or regional features • Classifications (SVM, ℓ_2 logistic regression, random forest) using cross-validations (K-fold, repeated K-fold, repeated hold-out) 	

FIGURE 6.2: List of the pipelines currently available in Clinica with their dependencies and outputs. GM, gray matter; CSF, cerebrospinal fluid; WM, white matter; FA, fractional anisotropy; MD, mean diffusivity; AD, axial diffusivity; RD, radial diffusivity, SVM, Support Vector Machine; ICBM, International Consortium for Brain Mapping.

I added the `pet-linear` pipeline to Clinica. All the steps were originally implemented in Python using Nipype (Gorgolewski et al., 2011), a Python library that provides an interface for most of the neuroimaging software tools. The pipeline has now migrated from Nipype to Pydra for its backbone dataflow engine (Jarecka et al., 2020) in the more recent versions of Clinica Vaillant et al., 2023. A set of instantiation and non-regression tests for the continuous integration process have been developed to ensure the robustness of the pipeline (and the software in general). Finally, the documentation of the pipeline⁷ is available online. The pipeline is available since the release 0.4 of Clinica.

⁷https://aramislab.paris.inria.fr/clinica/docs/public/latest/Pipelines/PET_Linear/

6.2 ClinicaDL

ClinicaDL is an open-source software package entirely written in Python. It uses the PyTorch library as backbone. ClinicaDL extends PyTorch for neuroimaging applications, where the dataset structure plays a key role in the organization of the data and metadata. The software is publicly distributed as an easy-to-install package and is referenced in the Pypi package index⁸. Releases are performed on a periodic basis and the code follows the most standard current practices for software development. The functionalities described in this chapter correspond to version 1.5.1. For more information on the versions of the dependencies, the reader can refer to the `poetry.lock` file⁹.

ClinicaDL has been designed to be used via the command line interface, with separate sub-commands performing the main tasks, as defined in a classical machine learning pipeline: `prepare-data`, `train`, `predict`. Other sub-commands are available in order to allow the user to structure the datasets, create synthetic data, search for hyperparameters and interpret trained networks. These functionalities are also available through the command line (`tsvtools`, `generate`, `random-search`, and `interpret`).

6.2.1 Main functionalities

The main functionalities of ClinicaDL cover all the steps needed for deep learning experiments, from dataset management to the evaluation of results and network interpretation. ClinicaDL's workflow is illustrated in Figure 6.3. In addition to pre-implemented options, the source code aims at being modular and the documentation helps users to easily implement their custom experiments¹⁰. Technical details for each command can be found in the user documentation.

Preprocessing images

ClinicaDL works with preprocessed images obtained using Clinica for different imaging modalities. This software provides, light preprocessing pipelines for anatomical MRI and PET images that output images suited for further analysis with deep learning.

ClinicaDL proposes a simple tool to transform NIfTI images into PyTorch tensors. The objective is to facilitate the training phase by decompressing and save the images beforehand (the NIfTI format usually provides compressed images). The user can choose the shape of these tensors by selecting a mode that is an `image`, a `patch`, a region of interest (`roi`) or a `slice`.

Generation of toy datasets

ClinicaDL facilitates the generation of synthetic data for evaluation and verification purposes. The synthetic data is already organized in the CAPS format (see Section 6.1.1). Four types of data can be created:

⁸<https://pypi.org/project/clinicadl>

⁹<https://github.com/aramis-lab/clinicadl/blob/v1.5.1/poetry.lock>

¹⁰<https://clinicadl.readthedocs.io/en/latest/Contribute/Custom/>

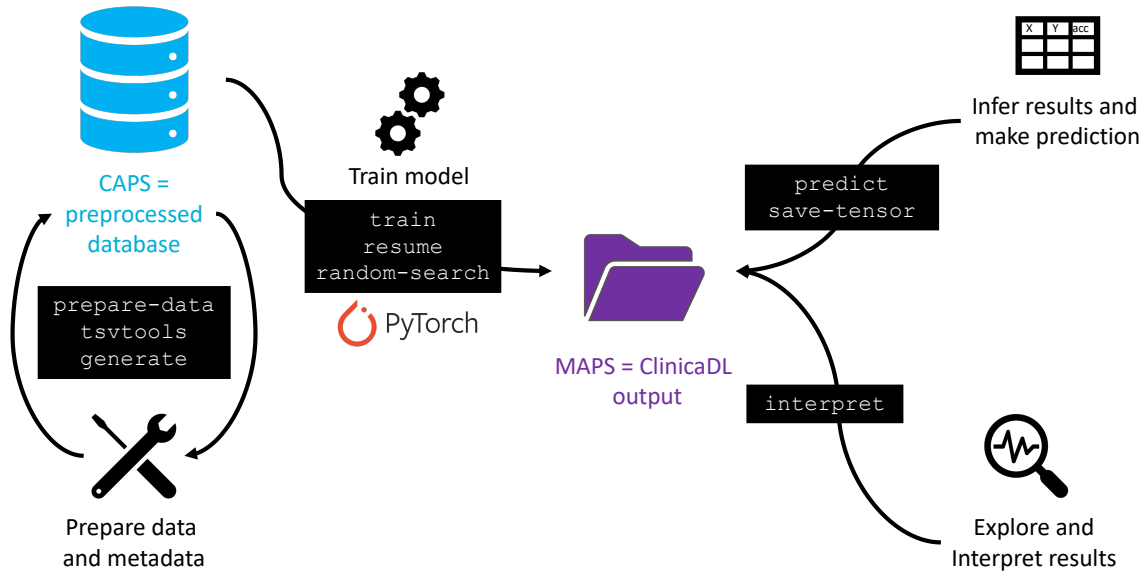


FIGURE 6.3: ClinicaDL main functionalities. `extract`, `tsvtools` and `generate` functionalities read and write in the Clinica Processed Structure (CAPS), which contains neuroimaging data preprocessed by Clinica pipelines. ClinicaDL writes its own output, the Model Analysis and Processing Structure (MAPS), which contains the results of the training phase as well as inference on new data or the results of interpretability methods.

- **Trivial data:** A mask is used to create incomplete images. By default, a mask based on a neuroanatomical atlas is used to create images where only half of the brain is present (half-left or half-right). Other kinds of tampering can be created by supplying a customized mask. The final result is the suppression of the region present in the mask.
- **Random data:** All the images belonging to this type of data are obtained from a single image, adding random white noise. The standard deviation of the noise is a parameter chosen by the user. Resulting images are then randomly distributed between two possible labels.
- **Shepp-Logan data:** 2D images whose appearance is based on the Shepp-Logan phantom Shepp et al., 1974 are generated (see Chapter 2, Section 2.2).
- **Hypo-metabolic data:** 3D FDG PET images with an hypometabolism simulated following the procedure detailed in Chapter 3, Section 3.2.2.

Preparing metadata

To use the train and inference functionalities of the software or to analyze the data, inputs must be organized in the right way. A collection of tools to handle metadata of BIDS-formatted datasets is proposed with ClinicaDL. These tools are intended to provide the correct organization of the data: get the labels used for classification, split the data to define test, validation and train subsets, and analyze the population of interest. This set of commands is available through the command `clinica dl tsvtools`, it includes:

- Extraction of labels specific to a particular diagnosis trajectory (e.g. participants labeled with Alzheimer’s disease diagnosis for all their sessions).
- Splitting the dataset at subject level to produce similar distributions from a specific population, using as parameters sex and age.
- Splitting the dataset at subject level to perform k-fold cross validation.
- Writing reports to summarize the demographics and clinical distributions of a specific label.

Random search

Random search consists in randomly sampling sets of hyperparameters (architecture and other training hyperparameters) to select the best set of hyperparameters as a result. In ClinicaDL, this hyperparameter space is described by a configuration file created by the user. We used this functionality to perform experiments of Chapter 5.

Training networks

The main functionality of ClinicaDL is to train neural networks to learn a task. These tasks can be:

1. **Classification** (of a categorical label, for example the diagnosis),
2. **Regression** (of a continuous label, for example the age),
3. **Image reconstruction.**

Segmentation is currently not handled by ClinicaDL. However, as the software is meant to be extensible, new tasks can be easily added by advanced users.

Some pre-built deep learning architectures for each task are available in ClinicaDL and their list and details can be displayed with the command `clenicadl train list_models`. However, an objective of the library is to allow the users to add and use their custom architectures easily. To this end, users can implement their custom networks by filling an abstract template, which includes specific methods that are used in ClinicaDL. The procedure of such addition is detailed in the documentation¹¹.

The models produced by ClinicaDL correspond to the ones that obtained the best performance on the validation set according to metrics chosen by the user. ClinicaDL saves at the end of each epoch the state of the network and of the optimizer. For each selection metric given in input, it replaces the corresponding current best model by the current state if the performance on the validation set is better than the current best value. To minimize the size of the produced MAPS, the checkpoints are deleted at the end of the training procedure. They are only used to resume a stopped job, thanks to the dedicated command `resume`.

The command line interface of ClinicaDL offers many options, as there is a large number of training parameters. This is why we tend to a parametrization by configuration files only.

¹¹<https://clenicadl.readthedocs.io/en/latest/Contribute/Custom/>

Performance evaluation

ClinicaDL provides specific functions to easily perform inference with models previously trained with the tool. For instance, this is useful to evaluate the model performance on an independent test set. The results are written in the MAPS as pre-formatted reports with the metric values at different levels (e.g. image-level and patch-level) and the output values computed for each input image of the data group.

The metrics computed depend on the task learned by the network. The regression task is associated with the mean squared error and mean absolute error, reconstruction task is associated with the mean squared error, mean absolute error the structural similarity (Wang et al., 2004) and the peak signal-to-noise ration, and the classification task is evaluated thanks to balanced accuracy, accuracy, sensitivity, specificity, positive and negative predictive values. Advanced users can add any new metric by following the procedure described in the advanced user guide.

Interpretation

The most critical issue of deep learning methods is their lack of transparency. This is why some interpretability methods have been developed specifically for the field. These methods allow better understanding which patterns or zones of the images have been linked to the result produced by the network. For instance, the gradient back-propagation method proposed in (Simonyan et al., 2013) is implemented in ClinicaDL.

6.2.2 Model Analysis and Processing Structure (MAPS)

As Clinica, ClinicaDL has its own output data structure, called the Model Analysis and Processing Structure (MAPS). All the functions of ClinicaDL are meant to work on this structure to easily retrieve the parameters of the command line, the weights of the best models, the checkpoints, or the predictions made on the training and validation sets to compute the results at the image level on independent test sets. At the root of the hierarchy, the file `environment.txt` summarizes the environment used for training, and `maps.json` gathers the arguments provided to the command line.

This structure includes a hierarchy of three levels:

1. **Splits** The first level contains one folder per train / validation split. The training procedure of each split can be launched independently.
2. **Selection metrics** During the training procedure of a particular split, one network is selected per selection metric given in input. These networks correspond to the network having the best validation performance according to their metric during the training procedure.
3. **Data groups** Finally, the best networks selected are evaluated on data groups. The characteristics of these data groups (TSV file of participant and session IDs with label values, and path to the CAPS directory) are stored at the first level of the hierarchy in the `groups` folder. This specification ensures the consistency between the evaluations of different networks trained on different splits and selected on different metrics.

An example of the MAPS obtained when training a classification convolutional neural network trained on images is displayed in Appendix H. The MAPS also stores training logs. Two different formats are available: they can be opened with Tensorboard¹² and are also available as TSV files.

6.2.3 Main features of ClinicaDL

In this section, we focus on how ClinicaDL aims to address three recurrent issues of deep learning applied to neuroimaging: the difficulties using neuroimaging datasets, the lack of reproducibility of deep learning studies, and the methodological flaws that can be found in the literature.

Easy use of neuroimaging

One difficulty faced by data scientists is the manipulation of raw neuroimaging datasets, as their organization can be quite difficult to understand. Moreover, raw images coming from different scanners may need some preprocessing to be handled by deep neural networks. These preprocessing steps are easier to perform and manage when data are organized in a standard manner.

A first step to make it easier to use neuroimaging datasets and to make experiments reproducible is to be part of the effort made by the BIDS community to standardize the organization of the datasets. The CAPS is part of this effort: this BIDS-like structure has the added benefit of considering all datasets as longitudinal and always using compressed NIFTI files (Li et al., 2016). ClinicaDL can automatically read in a CAPS and load images that have been converted and processed by Clinica for training and inference, enabling to easily train deep neural networks on the most common neuroimaging datasets. ClinicaDL also allows extracting 2D slices, 3D patches or regions of interest as PyTorch tensors from 3D brain volumes to facilitate training (Figure 6.4).

Reproducibility of deep learning studies

The initial step to achieve reproducibility is through transparency by sharing a usable code. The source code of ClinicaDL is available on GitHub¹³. Moreover, a set of instantiation and non-regression tests are run at each commit to ensure that the code does not break when adding new features (see Section 6.2.4 for more information on tests). This is crucial for the stability of the tool, especially for open-source software, where any person can contribute to the project.

However, sharing code is not enough to be fully transparent. The code and documentation of ClinicaDL are versioned to allow the user to retrieve the exact version needed for method reproducibility. Then, two files at the root of the experiment folder identify the software and dependencies' versions (`environment.txt`) and variables such as threading, GPU usage and random seed (`maps.json`), allowing to re-run experiments with the same environment and same computational parameters.

¹²<https://www.tensorflow.org/tensorboard>

¹³<https://github.com/aramis-lab/ClinicaDL>

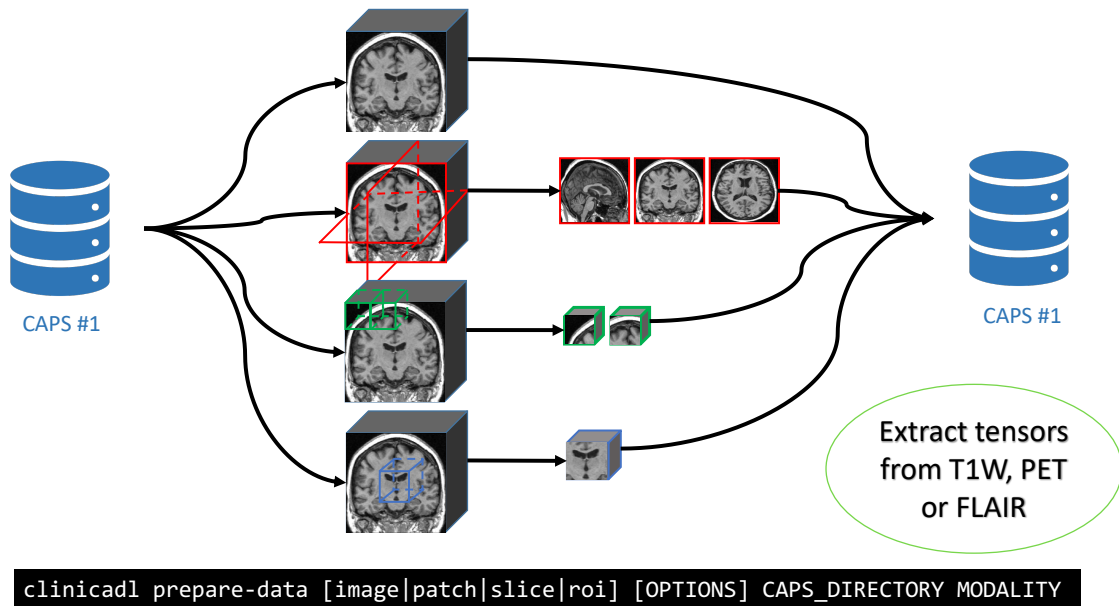


FIGURE 6.4: Schema of the `prepare-data` pipeline. It offers the possibility of extracting 2D slices, 3D patches or regions of interest as PyTorch tensors from 3D NIfTI files and store them in the same CAPS.

Moreover, the function `clinicadl train -config_file` was designed to repeat experiments based on a configuration file. Together with the `maps.json` in which all the hyperparameters of each experiment are saved, it allows to easily re-run experiments with the exact same configuration. However, we remind that it is still the users' responsibility to describe their GPU system.

Documentation is also a crucial point to ensure transparency and code usability by other teams, which then allows result reproducibility. This is why ClinicaDL comes with documentation support, and tutorials¹⁴.

Avoid common methodological biases in your neuroimaging studies

As explained by Kaufman et al., 2012, data leakage is “the introduction of information about the target of a data mining [a.k.a. machine learning] problem that should not be legitimately available to mine from”. They give two main reasons for data leakage:

- leaking features, occurring for example when input data include features that are highly correlated to the target label due to a selection bias or if the target is a cause of the feature,
- leakage in training examples, occurring when data used for training is not legitimate towards data used for performance evaluation (for example, if there is an intersection between training and test data).

Wen et al., 2020 reported that data leakage contaminated nearly half of the studies using a convolutional neural network on T1c MRI for the diagnosis of Alzheimer's disease. They also identified different scenarios of data leakage that may corrupt the model training and bias the results. These scenarios are summarized in Figure 6.5.

¹⁴<https://aramislab.paris.inria.fr/clinicadl/tuto/2023/html/>

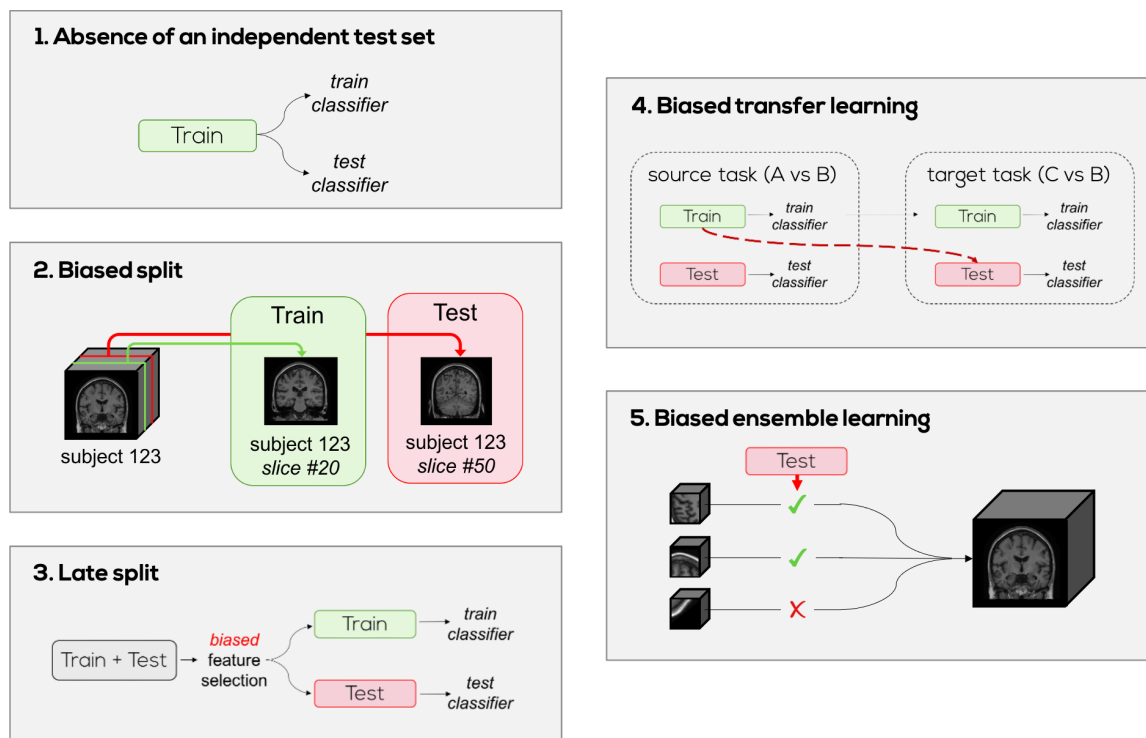


FIGURE 6.5: Illustration of the scenarios that can lead to data leakage.

To limit the risk of data leakage, ClinicaDL includes a set of pipelines and tests to avoid users making the most common methodological mistakes:

1. Data splits are performed at the subject level and cannot be performed on-the-fly, but must be done prior to training networks (to avoid a biased split).
2. Data splits are performed independently for each label. However, if labels B & C are subsets of a parent label A, transfer learning from a task implying A to a task implying B and/or C may result in a biased transfer learning. Therefore, ClinicaDL splits B and C with respect to A split.
3. In the classification case, the image-level prediction is the weighted sum of parts of the image. These weights are computed from the predictions on the training or the validation sets, but no other set (to avoid biased ensemble learning).
4. At the root of the MAPS, the file `train+validation.tsv` comprises all the participant and session IDs seen during the training procedure. If transfer learning is performed, this list of IDs is updated to include the IDs of participants and sessions seen during the training of the source task. ClinicaDL prevents the user from creating a data group having common IDs with this list (to avoid biased data split and transfer learning).

6.2.4 Development practices

ClinicaDL has adopted standard practices for software development and distribution of the software with the aim to facilitate the reproduction of experiments.

Distribution and Installation

The source code is hosted on Github¹⁵. It uses a version control system and the releases are strictly labeled with the version number. In consequence, the source code used in a specific experiment can be easily retrieved. Labeled versions of the code are released as Python packages that are permanently stored in the official Python Package Index. Good practices related to the version control system include atomic committing, clear commit messages and peer-reviewed contributions.

The installation of the released packages is performed with a single command (`pip install clinicadl`). As often, when installing Python packages, users are advised to install it into a virtual environment to avoid requirement conflicts. Instructions for developer installation are also available in the `README` of the repository.

Continuous Integration and Deployment

Each contribution is peer-reviewed by a developer different from the original author. The resulting code is only integrated to the development branch if the post commit actions are executed in a satisfactory way. The ensemble of these actions is described in the Continuous Integration pipeline:

- **Environment and dependencies verification:** The creation of an environment with all the dependencies necessary to install the package is performed in this step.
- **User interface tests:** The command line interface is tested using the Pytest library. This library allows combining several sets of possible commands used in the user interface. These are systematically tested to avoid errors in the main interface of ClinicaDL.
- **Functional tests:** A different kind of tests is executed before the integration of new code. These tests are called functional tests and are designed to check for the proper operation of the different functionalities proposed by the software: e.g. “Train”, “Transfer Learning”, “Interpretation” and “Random Search” tests use a truncated dataset to verify that these functionalities run properly on a GPU machine. Other functionalities such as “Predict” to perform inference, “Generate” to create custom datasets or “TSV Tools” to generate files adapted to the task / dataset are also checked.
- **Documentation build:** New contributions and/or modifications to the code are expected to be accompanied by the respective documentation. For that reason, documentation is built during the continuous integration pipeline. More details are explained in Section 6.2.4.
- **Deployment:** This step is only executed on labeled commits. Indeed, if a commit has a label to reference a version, a Python package is built and uploaded to the Python Package Index and a new version is published.

¹⁵<https://github.com/aramis-lab/ClinicaDL>

Documentation

The documentation of ClinicaDL is available online at <https://clinicadl.readthedocs.io>. It is automatically built after each commit by Read the Docs¹⁶. The documentation is versioned in the same way as the source code. All previous tags are easily accessible online with the version panel in the bottom right corner of any page.

6.2.5 Recent advances

In this section, we present in more detail ClinicaDL’s new features, which have been designed and implemented since the last journal publication (Thibeau-Sutre et al., 2022b) and conference presentation (Thibeau-Sutre et al., 2022a). These new features are summarized in Figure 6.6. Three of them concern the topics described in Section 6.2.3 (easy use of neuroimaging data, reproducibility and validation). We also added features related to usability (making the platform more user-friendly and adding deep learning features) and performance.

Easy use of neuroimaging. We added various functionalities for data augmentation and synthetic data generation. ClinicaDL now supports TorchIO (Pérez-García et al., 2021) data augmentation. Other generation pipelines have been implemented to generate motion artifacts (Loizillon et al., 2023) and hypometabolic data (Chapter 3, Section 3.2.2). These generation pipelines can be used for data augmentation, but also to validate models on synthetic data. Finally, ClinicaDL can now be used with MRI FLAIR sequence that is processed by Clinica.

Reproducibility. We made major improvements to the continuous integration. We added versioning of test data used for continuous integration with data version control (DVC)¹⁷. We also added non-regression tests for some pipelines and unitary tests for some critical functionalities of the software. There is now the possibility to fix the seed to improve the reproducibility of the results. This will, for example, determine the initialization of the model and the data loading sequence. However, despite having control over some GPU seeds, certain hardware-related factors such as architecture, memory configuration, clock speed, and calculation variations may still be beyond control, impacting reproducibility across different GPUs.

Rigorous validation. Another development axis has been to generalize experiment preparation to any neuroimaging dataset. Indeed, ClinicaDL initially resulted from work on the reproducibility of Alzheimer’s disease classification (Wen et al., 2020) and thus some of its features were not generic enough. We have enhanced tools for manipulating TSV files to make them more generic and to handle both cross-sectional and longitudinal studies. Quality control (QC) of both raw and processed data is important to mitigate sources of bias and short-cut learning. QC algorithms that were already included in ClinicaDL have been updated to their latest version (Fonov et al., 2022) and we added a new pipeline to check the

¹⁶<https://readthedocs.org/>

¹⁷<https://dvc.org>

registration of positron emission tomography (PET) images with a template. We further added a new method to interpret pretrained models.

Usability and performance. Making the software user-friendly has always been a goal for the development team by keeping the documentation up to date, writing tutorials or providing options such as the use of configuration files to simplify the command line. To go further, many new features have been added. Moreover, ClinicaDL now supports tracking of experiments via MLflow¹⁸ (an open-source platform) and Weights and Biases¹⁹ (a Python based platform) (Biewald et al., 2020), which are widely used in the machine learning community. The command line could sometimes be tedious to use, especially when changing many parameters of the experiments. This is why we created a TOML generator²⁰, a web application that helps to configure experiments through a graphical user interface. Furthermore, new tutorials²¹ are available online: the aim is to show and explain how to use Clinica and ClinicaDL as well as providing guidelines and spreading good practices to the community

We also added new features to enhance the performance of model training with ClinicaDL. First, we integrated the PyTorch profiler that helps users to track GPU usage. Then, we performed developments to allow people to harness the power of high-performance computing (HPC) clusters (with multiple GPUs) and of state-of-the-art GPUs (Nvidia Tesla V100 and even A100), in particular those including tensor cores. We implemented multi-GPU training through distributed data parallelism. We added the use of automatic mixed precision for optimal use of GPU cards with tensor cores. This is even more crucial in medical imaging, as the size of data can saturate the memory of the GPUs. Thus, it gives the possibility to use larger or more complex models, use full resolution high dimensional images or increase the batch size.

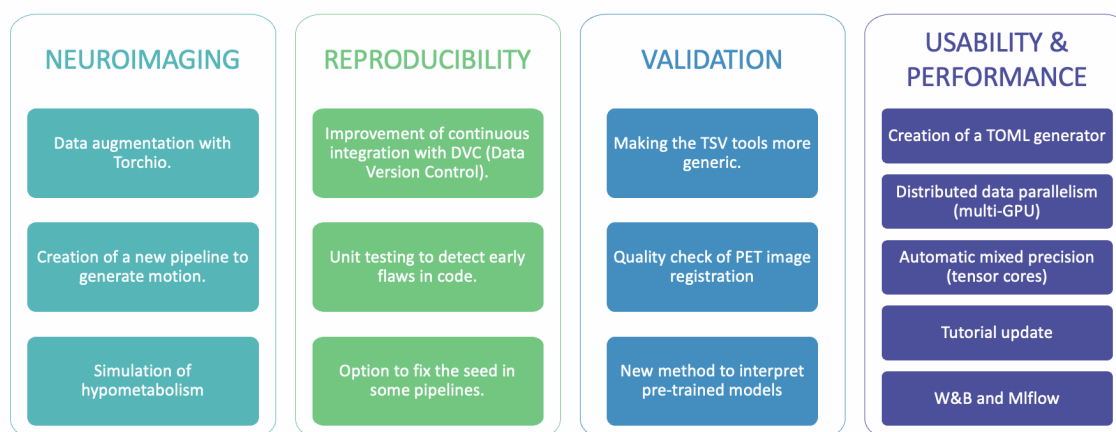


FIGURE 6.6: New features of the ClinicaDL software platform.

¹⁸<https://mlflow.org>

¹⁹<https://wandb.ai/site>

²⁰<https://clenicadl-toml-generator.streamlit.app>

²¹<https://aramislab.paris.inria.fr/clenicadl/tuto/2023/html/index.html>

6.2.6 Personal contribution

Since the beginning of my PhD, I have been involved in the ClinicaDL project and participated in its development. Over the past three years, I have made many contributions, both in terms of software development and project management.

The first contribution was to initiate a full refactoring of the software. The software had historically been developed for the reproducible classification of T1w MRI in the context of AD (Wen et al., 2020). After some years, many new features have been added, that still relied on that original implementation. The structure of the software, and the repository organization, started to be outdated, and understanding the code, maintaining it, and adding new features became more and more difficult. This is why, with the other project members, we decided to completely re-code the backbone engine of the software. Following this initiative, ClinicaDL has been refactored to have an object-oriented code, whereas it was only executing function in its initial version. This considerably increased the flexibility and organization of the code. This first version of the classes that we have designed is displayed in Figure 6.7. This prototype has then been upgraded during the following years. We also defined the MAPS (Section 6.2.2) to manage the outputs of our deep learning experiments. The goal was to design a structure that can store all the results of an experiment, including parameters and environment, in order to favor reproducibility. Finally, we added new standard Python dependencies, such as Click²² to manage the command line interface.

After the software refactoring, many new features have been added to ClinicaDL during the thesis. Especially, all the deep learning tools that have been used to run experiments have been integrated to the software. For instance, it is possible to train VAEs for reconstruction as done in Chapter 2. It is also possible to generate hypometabolic FDG PET and use the evaluation framework introduced in Chapter 3. Finally, all the VAE models tested in Chapter 5 are implemented in Pythae, and are available in a separate package called ClinicaDL-Pythae²³. In addition to that, many existing features have been improved, and new ones have been added. It includes the parallelization of the `prepare-data` pipeline, a new option to save reconstruction tensors and latent tensors, new metrics for evaluation, new deep learning models, and a quality check pipeline for PET images.

As part of this project, a significant effort was dedicated to promoting the software by presenting it during conferences, congresses, and workshops. Additionally, I took the responsibility of training newcomers, including PhD students who would be using ClinicaDL for their research and software engineers involved in its development. Furthermore, I played an active role in providing user support, by addressing inquiries and issues raised by external users on the forums dedicated to the software. Additionally, I provided assistance to users within our laboratory, ensuring that they could utilize the software efficiently and effectively to meet their research needs. Finally, I actively participated in updating the user's documentation.

Last but not least, I have also contributed to the management of the project. This involved creating short and midterm roadmaps, suggesting ideas for new features and functionalities, prioritizing tasks based on the roadmap, and planning the software releases.

²²<https://click.palletsprojects.com/en/8.1.x/>

²³<https://github.com/aramis-lab/clinicadl-pythae>

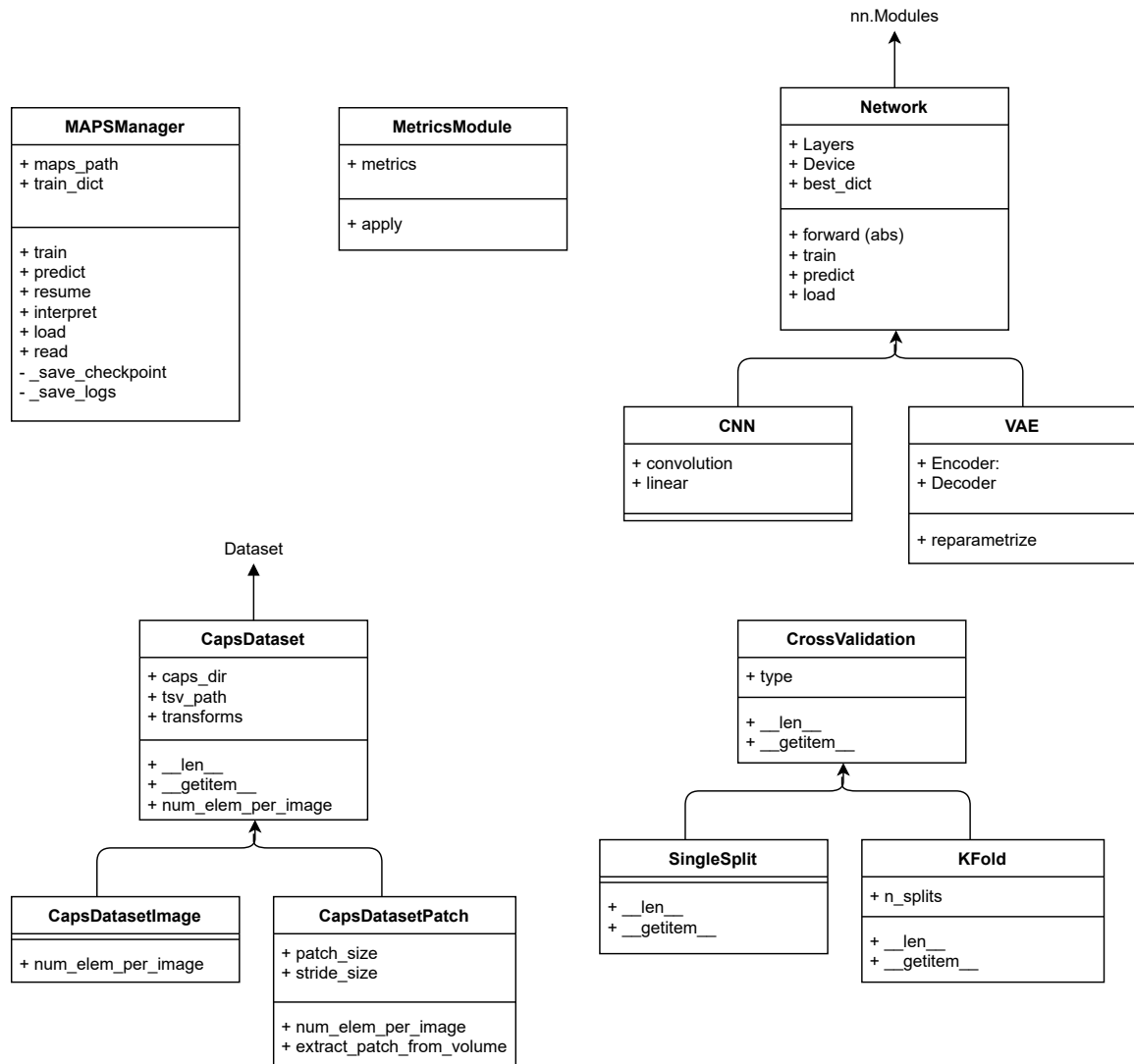


FIGURE 6.7: First UML diagram of ClinicaDL. We distinguish five main blocks: one related to data (CapsDataset), one for the deep learning models that inherited from Pytorch (Network), one for the validation (CrossValidation), one that contain metrics (MetricModule), and finally one that manage the trainer and the MAPS (MapsManager).

Additionally, I was during a long time in charge of reviewing new contributions before integrating them into the main codebase. This review process aimed to maintain code quality, consistency, and stability throughout the development cycle.

6.2.7 Discussion and future development

In this section, we presented the ClinicaDL software platform. It can facilitate and improve the trustworthiness of research in deep learning for neuroimaging.

Good practices are essential in research to provide strong foundations to those whose work is built on the findings of others. Working with neuroimaging data can be complex, making it intricate to reproduce experiments. The same applies to the field of deep learning, the important number of parameters to choose from can make difficult usability, reproducibility or validation. In this way, a versioned and open-source software like ClinicaDL is a first step towards reproducibility. ClinicaDL is built in a way that makes it easy

to get started. Researchers can use it “as is” or as a starting point to further develop tools for their research.

Future direction includes the development of feature such as:

- adding the segmentation task;
- adding new state-of-the-art models;
- giving the possibility to directly read neuroimaging data from BIDS (and not only CAPS);
- improving the trainer in order to enhance the training performances;
- improving the code structure to facilitate new contributions;
- improving tests and continuous integration pipeline;
- providing more tools for validation of training procedures.

The general objectives are oriented toward a more user oriented software by improving the interface, usability and performance, and integrate deep learning tools that have been adopted by the community.

6.3 Other contributions

In our quest of enhancing reproducibility in research, we have made all the code utilized in the different experiments conducted for this thesis publicly available. As a result, each article that has been published is accompanied by an associated GitHub repository, ensuring the reproducibility of our research. Moreover, this initiative facilitates the accessibility of our methods and results for fellow researchers, enabling them to easily utilize and build upon our work.

Conclusion and Perspectives

Conclusion

In this thesis, we explored the use of deep learning to automatically analyze neuroimages and provide a computer-aided-diagnosis tool in the context of dementia. Especially, we focused on brain FDG PET, an imaging modality used for the detection of hypometabolism indicating neurodegeneration, a biomarker used for early diagnosis of AD and other dementia causes. Our strategy was to use generative models, and more precisely, we trained variational autoencoders, to reconstruct pseudo-healthy images. This generative model, when trained with FDG PET images of healthy subjects only, learns their distribution. Thus, when reconstructing an image with unknown diagnosis, we expect the reconstruction to be anomaly free, since the model only learned to reconstruct “healthy images”. Then, by comparing the input image to its reconstruction, areas of the brain that are substantially different are probably abnormal, indicating the presence of anomalies related to the disease. The main advantage of this approach is that it does not require a labeled dataset.

A prerequisite to the VAE training was to preprocess the FDG PET images that we selected for our experiments. To this end, we developed the `pet-linear` pipeline that performs a linear registration to the MNI152NLin2009cSym template and an intensity normalization, resulting in an SUVR map. Moreover, we implemented a quality control pipeline to select images that were correctly preprocessed. We finally carefully selected stable healthy patients from the ADNI database in order to use their FDG PET images to train the VAE.

Once our dataset preprocessed, we trained a first simple 3D convolutional VAE to reconstruct healthy looking 3D brain FDG PET. Although we did not need annotated data for training purposes (we only ensured that the images used in the train/validation set corresponded to CN subjects), it was a challenge to evaluate quantitatively the ability of our model to reconstruct pseudo-healthy images without any ground truth masks of the anomalies we aimed to detect.

Since visual and qualitative evaluation was not an option at such an early stage of the method development, we proposed an evaluation framework to enable a complete assessment of generative models for the pseudo-healthy reconstruction of brain FDG PET. This framework relies on simulating hypometabolism mimicking the effect of the diseases causing dementia on images of healthy subjects from the test set. Using this technique, we obtained pairs of healthy and abnormal images that allowed us to evaluate both the ability of the model to reconstruct pseudo-healthy images, and the capacity to detect anomalies thanks to the pseudo-healthy reconstruction. Additionally, we defined a new healthiness metric and an anomaly score to quantitatively measure the performance of the generative model

for this task. Finally, we exploited the possibility given by the VAE latent properties and simulated images in order to explain the VAE results.

After proving that the VAE is well suited for pseudo-healthy reconstruction of 3D brain FDG PET thanks to its simplicity and efficiency, we benchmarked 20 models: the autoencoder, the vanilla VAE and 18 variants of the VAE, and compared them using the previously introduced evaluation procedure. We first performed a random search in order to find the best architecture on the vanilla VAE. We then used the same architecture for all the 20 models, and searched for the optimal hyperparameters for all the models of this study. Finally, we compared the models both in terms of reconstruction quality and healthiness of the reconstructed images. We concluded from this benchmark that 17 out of 20 models can reconstruct good quality images, and that the 17 are able to reconstruct 3D brain FDG PET looking healthy when fed with abnormal images.

Finally, I have highly contributed to the development of ClinicaDL during this thesis. The purpose of this open-source software is to offer tools that facilitate the use of neuroimaging data in deep learning research, with an emphasis on reproducibility and conducting rigorous experiments.

As a conclusion, during this PhD thesis, we developed, implemented and shared most of the tools needed to conduct research in the field of unsupervised anomaly detection for brain FDG PET. It includes preparation of the data, software to implement deep learning models, and an evaluation framework. Thanks to this, we could provide a state-of-the-art on the use of 3D convolutional VAEs in such context, and opened the way to future developments.

Perspectives

A first avenue of research to continue and improve this work, would be to use a generative model that is able to reconstruct images of better quality. We have already tested many VAEs variants. However, it would be interesting to implement other approaches such as generative adversarial networks and diffusion probabilistic models (DDPMs) that have proven their ability to reconstruct images of better quality than VAEs (Esmaili et al., 2023; Wang et al., 2023; Gong et al., 2023). Especially, DDPMs seem to be the new state-of-the-art for image generation (Dhariwal et al., 2021).

For the moment, we mainly rely on the reconstruction error to detect anomalies. The use of difference maps is quite limited and could be improved in order to obtain more robust and precise anomaly maps. For instance, the use of z-scores have been explored (Solal et al., 2024a; Solal et al., 2024b) to leverage different sources of variance that would affect the model reconstruction.

More generally, rigorous evaluation and validation is a crucial step in deep learning applied to medical imaging (Varoquaux et al., 2022), since wrong estimation of the results can have dramatic impact. We showed in Figure 4 that the number of articles published about deep learning for computer-aided diagnosis is on the rise. However, in practice, translation to clinical applications are really limited. One of the reasons is the lack of trust of clinicians and patients in these tools. This why it is essential to build new evaluation methods in order to improve trust in deep learning algorithm.

Besides, interpretability is a field of deep learning research that has gained a lot of interest in recent years. It is also a great tool to increase reliability of deep learning algorithms, that are often perceived as “black boxes”, in the context of medical imaging.

Another avenue for improvement would be to apply the anomaly detection approach to different data. This includes using a different PET tracer, for instance amyloid tracers, in order to detect signs of AD earlier, or using different modalities, such as anatomical, diffusion or functional MRI, in order to see if the approach can be more robust. Ultimately, the use of multi-modal data, i.e. using different imaging modalities together with clinical data, can be a key to build better performing and robust computer-aided diagnosis tools. Finally, most of the research is realized on research datasets, that are well curated, often acquired following a rigorous protocol. It is rarely the case of clinical data. Because of this, models developed in a research context may not generalize to real clinical applications, slowing down the process of translation to clinic. This highlights the need to extend what we have done from research datasets to clinical datasets.

Last but not least, the development of software tools is indispensable, not only for ensuring research reproducibility but also for facilitating the transition from research to clinical applications. For the moment, most of the software frameworks focus on providing resources to develop and train new models using medical imaging data. However, to the best of our knowledge, only MONAI²⁴ offers the possibility to easily deploy trained models for clinical settings. Indeed, deploying deep learning models can be challenging, particularly in medical imaging applications where stringent security measures and performance considerations are primordial. This is probably a significant drag for the use of deep learning in clinical practice.

In a nutshell, I believe that greater emphasis should be placed on enhancing evaluation methodologies, developing software tools for model deployment, creating robust and interpretable models, and providing access to larger and more diverse datasets, including clinical data. These efforts are essential for facilitating the translation of research methods for computer-aided diagnosis into clinical applications.

²⁴<https://monai.io/deploy.html>

Appendix A

PubMed database queries

Machine learning query

(alzheimer [Title] OR "Cognitive Impairment" [Title] OR "MCI" [Title])

AND ("classif*" [Title] OR "diagnos*" [Title] OR "identif*" [Title] OR "detect*" [Title] OR "recogni*" [Title] OR "prognos*" [Title] OR "predict*" [Title])

AND (mri OR "Magnetic Resonance Imaging" OR neuroimaging OR (brain AND imaging) OR positron OR PET)

AND ("Matrix completion" [Title/Abstract] OR "Support vector machine\$" [Title/Abstract] OR "linear mixed-effect\$" [Title/Abstract] OR "Machine Learning" [Title/Abstract] OR "logistic regression" [Title/Abstract] OR "Random Forest" [Title/Abstract] OR "kernel\$" [Title/Abstract] OR "decision tree\$" [Title/Abstract] OR "least-squares" [Title/Abstract])

NOT ("cnn\$" [Title] OR "Convolutional Network\$" [Title] OR "Convolutional neural Network\$" [Title] OR "Deep Learning" [Title] OR "Neural Network\$" [Title] OR "autoencoder\$" [Title] OR gan [Title] OR adversarial [Title] OR "deep belief network\$" [Title])

Deep learning query

(alzheimer [Title] OR "Cognitive Impairment" [Title] OR "MCI" [Title])

AND ("classif*" [Title] OR "diagnos*" [Title] OR "identif*" [Title] OR "detect*" [Title] OR "recogni*" [Title] OR "prognos*" [Title] OR "predict*" [Title])

AND (mri OR "Magnetic Resonance Imaging" OR neuroimaging OR (brain AND imaging) OR positron OR PET)

AND ("cnn\$" [Title/Abstract] OR "Convolutional Network\$" [Title/Abstract] OR "Convolutional neural Network\$" [Title/Abstract] OR "Deep Learning" [Title/Abstract] OR "Neural Network\$" [Title/Abstract] OR "autoencoder\$" [Title/Abstract] OR gan [Title/Abstract] OR adversarial [Title/Abstract] OR "deep belief network\$" [Title/Abstract])

NOT ("Matrix completion" [Title] OR "Support vector machine" [Title] OR "linear mixed-effect" [Title] OR "Machine Learning" [Title] OR "logistic regression" [Title] OR "Random Forest" [Title] OR "kernel" [Title] OR "decision tree" [Title] OR "decision trees" [Title] OR "least-squares" [Title])

Appendix B

Anomaly detection in Shepp-Logan phantoms

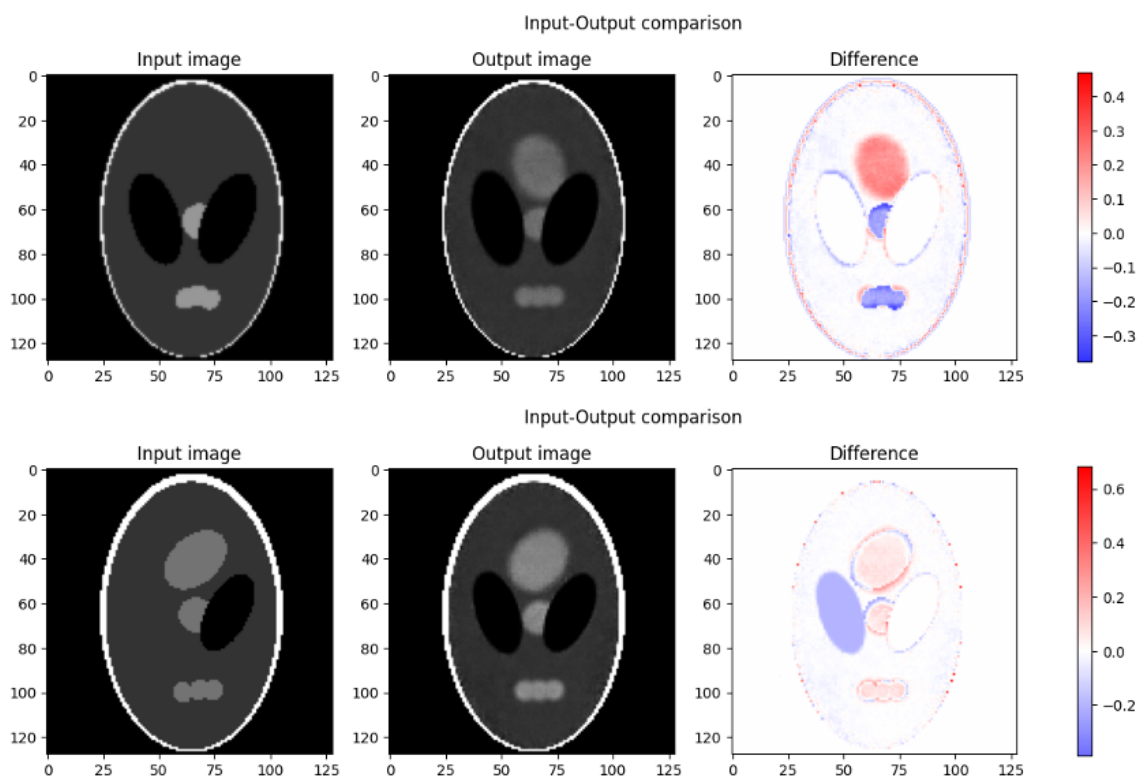


FIGURE B.1: VAE pseudo-healthy reconstruction on images with different missing components: the top ROI is missing (top), the left ventricle is missing (bottom).

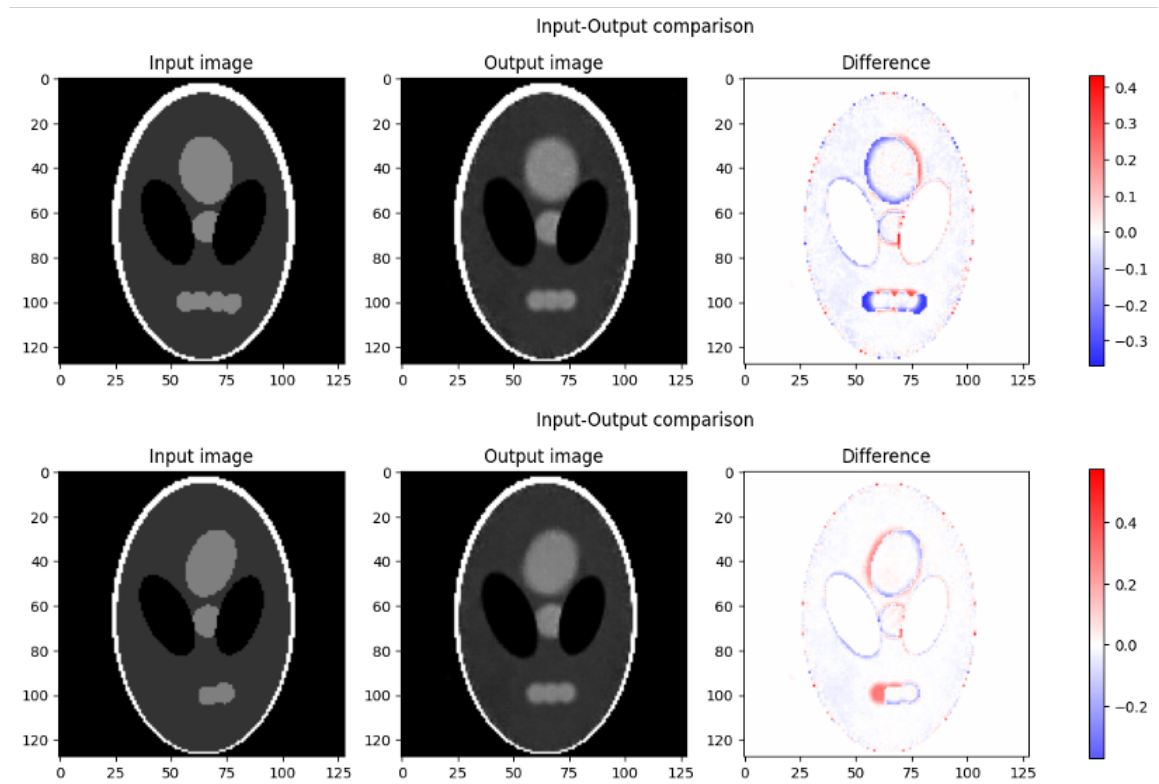


FIGURE B.2: VAE pseudo-healthy reconstruction on images with anomalies on the bottom ROI: the presence of an extra ellipse (top), the absence of an ellipse (bottom).

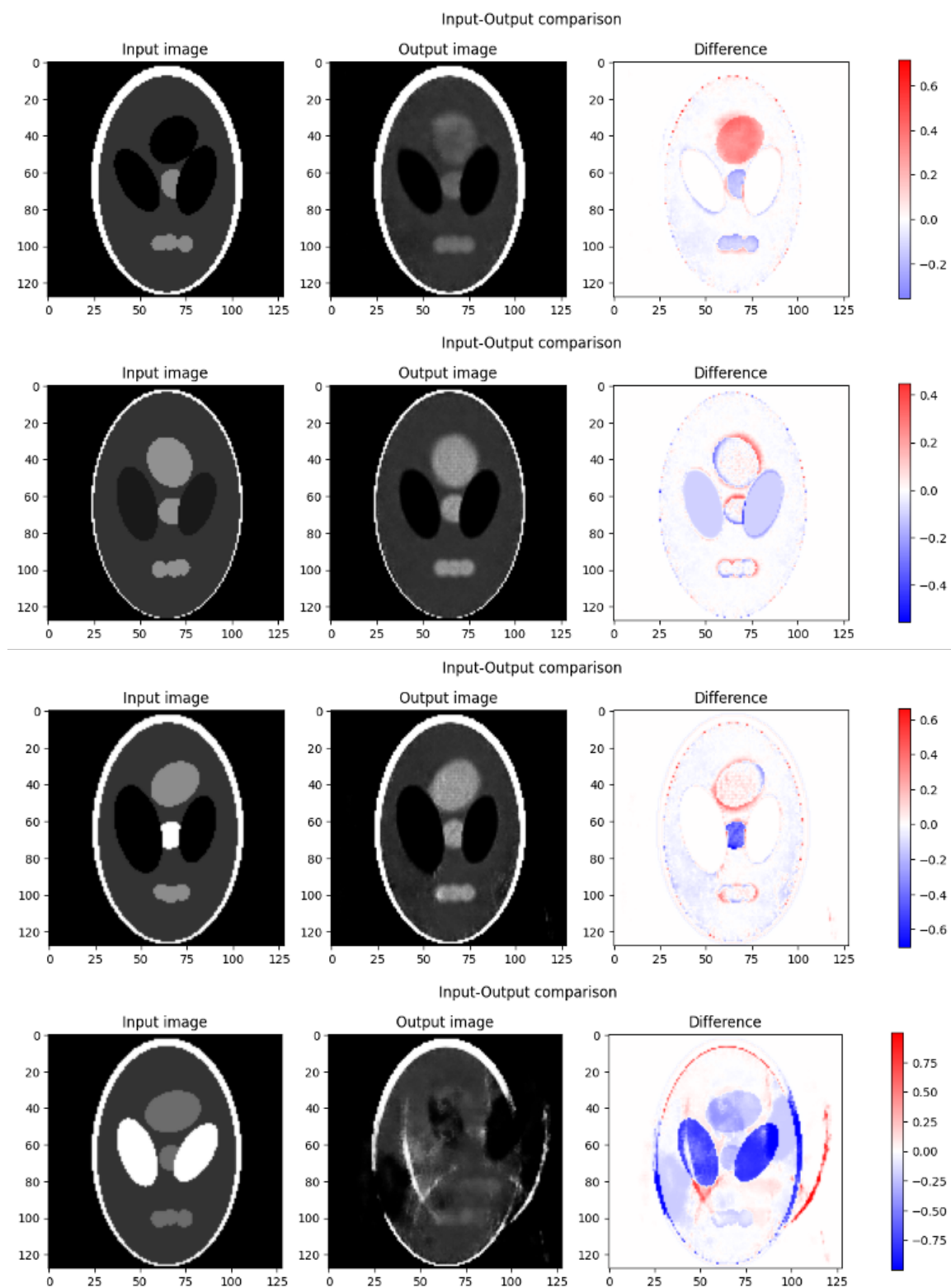


FIGURE B.3: VAE pseudo-healthy reconstruction on images with different intensities: intensity of the top ROI set to 0 (first row), intensity of the ventricles set to 0.2 (second row), intensity of the central ROI set to 1 (third row), and intensity of the ventricles set to 1 (last row).

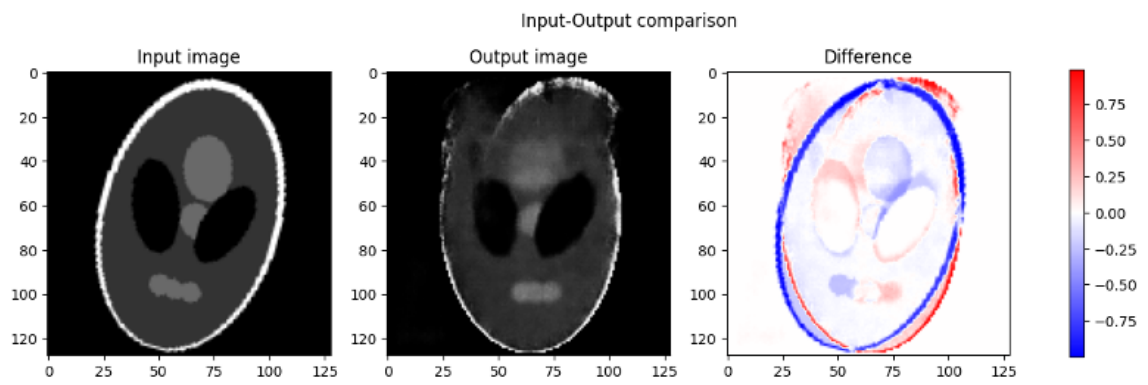


FIGURE B.4: VAE pseudo-healthy reconstruction on image with a 15° rotation.

Appendix C

Examples of reconstructions obtained for healthy subjects and simulated hypometabolic images

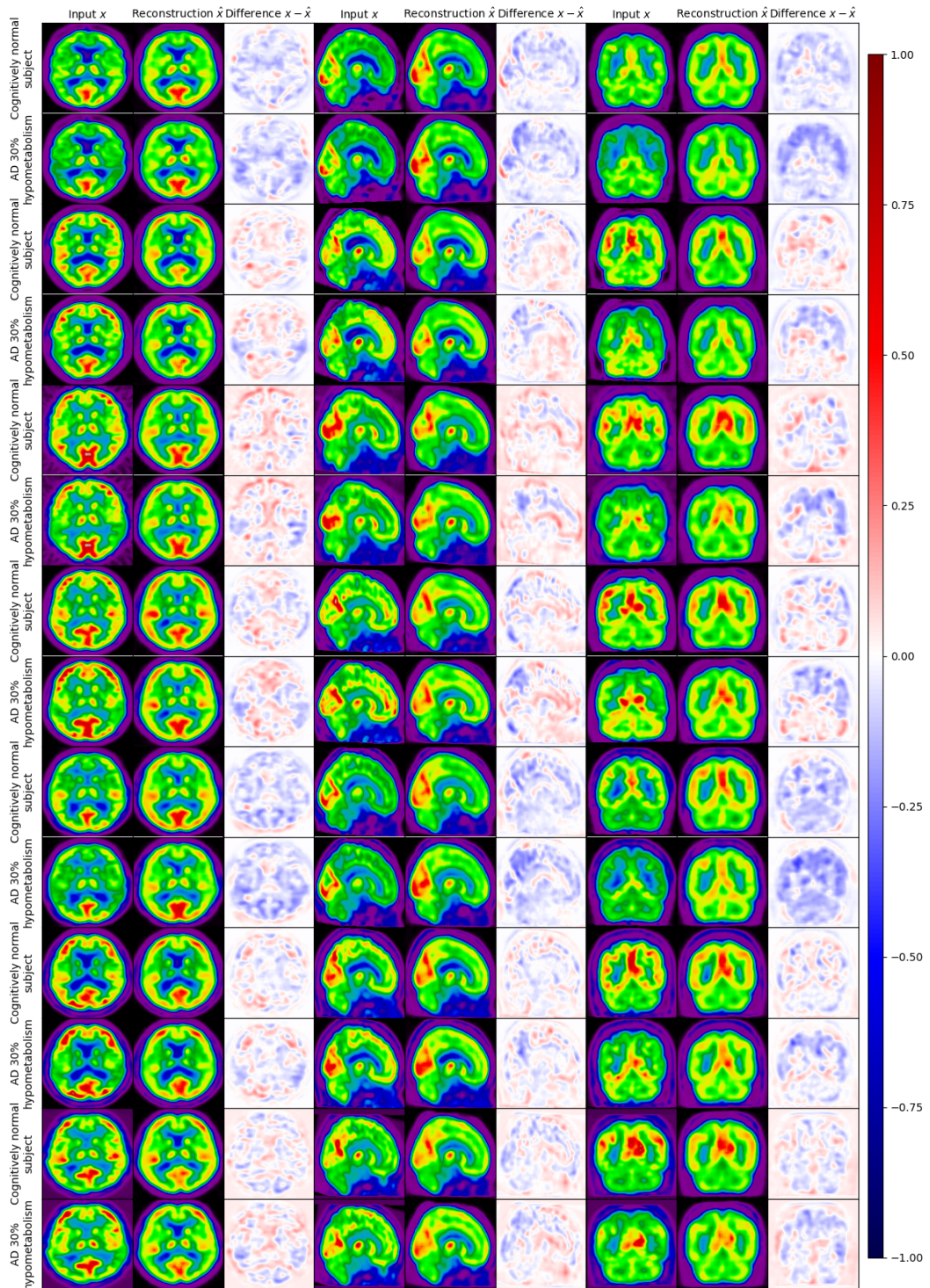


FIGURE C.1: Examples of reconstructions obtained from a real image of CN subjects (even rows) and the image simulating 30% AD hypometabolism based on the same CN subject (odd rows). For each plane, the first image is the input, the second one the model's reconstruction and the third one the difference (input - reconstruction).

Appendix D

Unsupervised anomaly detection in 3D brain FDG PET: A benchmark of 17 VAE-based approaches

This Appendix has been published as a conference proceeding in the Deep Generative Models workshop at the 26th International Conference on Medical Image Computing and Computer Assisted Intervention (DGM@MICCAI 2023, Vancouver, Canada).

- **Title:** Unsupervised anomaly detection in 3D brain FDG PET: A benchmark of 17 VAE-based approaches
 - **Authors:** Ravi Hassanaly, Camille Brianceau, Olivier Colliot, Ninon Burgos.
-

D.1 Introduction

Recent advances in medical image analysis have allowed the emergence of algorithms that can perform complex tasks such as computer-aided diagnosis (Chen et al., 2022; Fernando et al., 2021) with pseudo-healthy reconstruction for unsupervised anomaly detection (UAD). Contrary to supervised approaches, UAD does not require human annotations that are costly and time-consuming, and enables the detection of any type of anomalies, without having seen them before. Most approaches rely on generative models to reconstruct healthy looking images, also called pseudo-healthy images (Baur et al., 2021a; Chen et al., 2022; Fernando et al., 2021). The assumption is that if a model is trained with images from subjects diagnosed as healthy, the reconstruction of images with a pathology should not contain pathology-specific features and look like a healthy image. Comparing the pseudo-healthy reconstruction with the real image then allows the detection of anomalies.

The application context of our work is the detection of metabolic changes visible in brain ^{18}F -fluorodeoxyglucose (FDG) positron emission tomography (PET) caused by Alzheimer’s disease and other dementias (Chételat et al., 2020). These subtle changes appear several years before the first symptoms and can be used for early diagnosis (Jack et al., 2016; Hampel et al., 2021). In neuroimaging, deep learning methods for UAD have not been much

applied for the diagnosis of dementia (Choi et al., 2019). It is a challenging task because the metabolic abnormalities are diffuse and little intense, which makes them difficult to detect (Burgos et al., 2021b).

The different pseudo-healthy reconstruction approaches that have been developed for medical imaging rely on variational autoencoders (VAEs) (Kingma et al., 2014), generative adversarial networks (GANs) (Goodfellow et al., 2014) and more recently diffusion models (Ho et al., 2020). We aim to compare VAE-based models as they have shown their efficacy for UAD in medical imaging (Baur et al., 2021a; Chen et al., 2022), are easy to train, easily scalable, with good interpretation capacity thanks to their regularized latent space, and are able to handle small datasets. Much research to improve the original VAE has been achieved in the computer vision literature (Burda et al., 2016; Chen et al., 2018a; Ghosh et al., 2019; Higgins et al., 2017; Kim et al., 2018; Kingma et al., 2016; Larsen et al., 2016; Makhzani et al., 2015; Rezende et al., 2015; Snell et al., 2017; Tolstikhin et al., 2018; Tomczak et al., 2018; Van Den Oord et al., 2017; Zhao et al., 2019), but only a few have been translated to medical imaging applications (Baur et al., 2021a; Chen et al., 2018b; Choi et al., 2019; Mostapha et al., 2019; Uzunova et al., 2019).

We propose a benchmark of seventeen VAE-based models and show results in the context of pseudo-healthy reconstruction for dementia from 3D FDG PET. As far as we know, the only study that has compared VAEs for neuroimaging data is that of Baur et al., 2021a. However, it was restricted to models that had already been used for medical imaging applications. Many other VAE extensions have thus not been assessed. Also, it was dedicated to the detection of very sharp and intense anomalies, such as brain tumors or multiple sclerosis lesions, which is very different from the identification of subtle anomalies found in PET images of patients with cognitive disorders. Finally, it was performed in 2D. Our work aims to contribute to this effort by evaluating a much wider set of approaches, including many that were never used in medical imaging, relying on the work of Chadebec et al., 2022. This will provide an insight into the performance that such models can achieve in detecting anomalies in 3D data when trained with a relatively small dataset (few hundreds of images) compared to most datasets used in the computer vision literature (several tens of thousands images). The models will be evaluated and compared based on reconstruction quality and on their ability to generate healthy looking images using a previously proposed simulation framework (Hassanaly et al., 2023a).

D.2 Methods

D.2.1 Variational autoencoder framework for pseudo-healthy image reconstruction

Let D be a set of medical images of the same modality acquired following a similar protocol. D can contain healthy and pathological images and can be divided in respectively two complementary subsets D_h and D_p . Let's take as an example a set of FDG PET images $\mathbf{x} \in D_h$ whose distribution is $p(\mathbf{x})$. The goal of pseudo-healthy image reconstruction is to generate an FDG PET image of healthy appearance. The idea is to approximate the healthy image true distribution $p(\mathbf{x})$ with a chosen model $p_\theta(\mathbf{x})$ such that $p_\theta(\mathbf{x}) \approx p(\mathbf{x})$.

Then, during reconstruction, the images (of healthy subjects or patients) are projected into that “healthy images” learned subspace by the generative model.

This can be modeled using the VAE framework (Kingma et al., 2014) by assuming that a latent variable \mathbf{z} is involved in the generation process of \mathbf{x} : $p_\theta(\mathbf{x}) = \int_{\mathbf{z}} p(\mathbf{z})p_\theta(\mathbf{x} | \mathbf{z})d\mathbf{z}$ where $\mathbf{z} \sim p_\theta(\mathbf{z})$ is the prior distribution on the latent space and $p_\theta(\mathbf{x} | \mathbf{z})$ is the generative model (or the decoder) that learns to generate healthy images from \mathbf{z} . To compute the appropriate \mathbf{z} for each data input \mathbf{x} of our dataset, we need the posterior distribution $p_\theta(\mathbf{z} | \mathbf{x})$. Since it is untractable, we approximate it using variational inference by introducing another model $q_\phi(\mathbf{z} | \mathbf{x})$ such that $q_\phi(\mathbf{z} | \mathbf{x}) \approx p_\theta(\mathbf{z} | \mathbf{x})$. $q_\phi(\mathbf{z} | \mathbf{x})$ is the inference model (or encoder). Both the decoder and encoder are parametric models whose parameters are given by a neural network.

The objective is to maximize the likelihood of $p_\theta(\mathbf{x})$, which is equivalent to maximizing the evidence lower bound, which defines our loss function $\mathcal{L}_{\theta,\phi}$ (Kingma et al., 2014)

$$\log(p_\theta(\mathbf{x})) \geq \mathcal{L}_{\theta,\phi}(x) = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} \left[\log(p_\theta(\mathbf{x} | \mathbf{z})) \right] - D_{\text{KL}} \left(q_\phi(\mathbf{z} | \mathbf{x}) \| p_\theta(\mathbf{z}) \right) \quad (\text{D.1})$$

with D_{KL} the Kullback-Leibler divergence.

During the training process, we learn an approximation of the posterior distribution $q_\phi(\mathbf{z} | \mathbf{x})$ for $x \in D_h$ as we train our model using only healthy subjects. When using the model for inference, we use this approximate posterior to estimate the latent variable \mathbf{z} for $\mathbf{x} \in D$ (it can be from D_h or D_p).

D.2.2 Extensions to the variational autoencoder framework

As explained in detail in Chadebec et al., 2022, several contributions have been proposed to improve the VAE framework. They can be divided into four categories that correspond to different objectives:

- improve the prior distribution $p(\mathbf{z})$ by using a variational mixture of posteriors as prior (VAMP) (Tomczak et al., 2018), by learning the prior on a discrete latent space with vector quantized-VAE (VQVAE) (Van Den Oord et al., 2017), or by substituting the prior with a density estimation method using regularization with a gradient penalty (RAE-GP), or an ℓ^2 penalty on the decoder (RAE- ℓ^2) (Ghosh et al., 2019);
- better estimate the lower bound by using importance weighting (IWAE) (Burda et al., 2016), and using a linear normalizing flow (VAE LinNF) (Rezende et al., 2015) or an inverse autoregressive flow (VAE-IAF) (Kingma et al., 2016) to better estimate the posterior;
- encourage disentanglement of the features in the latent space by adding a weight to balance the terms of the loss in Eq. D.1 (β -VAE) (Higgins et al., 2017), decomposing the loss to show a total correlation term (β -TC VAE) (Chen et al., 2018a), or by encouraging the distribution of the latent variable $q(\mathbf{z})$ to be factorial (FactorVAE) (Kim et al., 2018);

- and change the distance computed between the distributions by adding the mutual information between \mathbf{x} and \mathbf{z} as regularization (InfoVAE) (Zhao et al., 2019), using another divergence term in the loss such as the maximum mean discrepancy in the Wasserstein autoencoder (WAE) (Tolstikhin et al., 2018) or a discriminator to differentiate a prior’s sample from a posterior’s sample in the adversarial autoencoder (AAE) (Makhzani et al., 2015), or by changing the reconstruction metric for another similarity metric such as the multi-scale structural similarity (MS-SSIM VAE) (Snell et al., 2017), or for the prediction of a discriminator on the output of the VAE (VAEGAN) (Larsen et al., 2016).

In our benchmark, these models will be compared to the autoencoder (AE) and VAE (Kingma et al., 2014), which makes a total of seventeen models. All of these methods have shown great results in other fields of computer vision, and, since VAE-based models can learn the data distribution on a small dataset, we keep the focus on them and aim to assess their performance in the context of medical imaging.

D.2.3 Evaluation of the models

We can distinguish two main objectives when generating pseudo-healthy images: preserving the subject’s identity in the reconstructed image and ensuring that the reconstruction appears healthy (Xia et al., 2020).

For the subject identity preservation, we evaluate the models on real images from healthy subjects only: the pseudo-healthy reconstruction of an image of a healthy subject should be identical to the input. This is assessed using three commonly used paired reconstruction metrics: the mean-squared error (MSE), the peak signal-to-noise ratio (PSNR) and the structural similarity (SSIM) (Wang et al., 2004).

To evaluate the capability of each model to reconstruct healthy looking images, since we do not have access to ground-truth lesions masks, we use the evaluation framework that has been introduced in Hassanaly et al., 2023a. It consists in simulating the effect of the disease by reducing the intensity of the PET uptake within regions associated with different dementias, thus mimicking regional hypometabolism (Burgos et al., 2021b). After locally reducing the intensity of the image by a certain percentage, a Gaussian smoothing is applied to have a realistic result and diffuse anomalies. That way we can have pairs of diseased images with the original healthy scan that is used as ground-truth for the pseudo-healthy reconstruction as we do not have ground truths for images from real patients in our dataset. We simulate five different dementias on images of healthy subjects: Alzheimer’s disease (AD), behavioral variant frontotemporal dementia (bvFTD), logopenic variant primary progressive aphasia (lvPPA), semantic variant PPA (svPPA) and posterior cortical atrophy (PCA). This allows us to evaluate the capability of the model to generalize to anomalies caused by different dementia subtypes. In addition, we simulate different degrees of AD severity by varying the reduction in intensity from five to seventy percents to study the sensitivity of the UAD approaches on subtle and severe anomalies. We compute the reconstruction error in the whole image, in the region associated with the simulated dementia and in the complementary of this region in the brain.

D.2.4 Materials

FDG PET scans used in this study were obtained from the publicly available ADNI database (Jagust et al., 2010) (<https://adni.loni.usc.edu>). We selected FDG PET images co-registered, averaged and uniformized to a resolution of 8 mm FWHM to reduce the variability due to the use of different scanners. The images were then linearly registered to the standard MNI space, normalized in intensity using the average PET uptake in a region comprising cerebellum and pons, and cropped using the Clinica `pet-linear` pipeline (Routier et al., 2021). We finally down-sampled the images to a voxel size of $80 \times 96 \times 80$ to reduce their dimension and the memory usage.

ADNI includes a total of 733 FDG PET scans of cognitively normal (CN) participants with a stable diagnosis over a three-year window (corresponding to 301 subjects). We discarded 144 images that were not correctly registered according to the quality check algorithms implemented in ClinicaDL (Thibeau-Sutre et al., 2022b).

D.2.5 Experimental setting

We split our dataset of 247 remaining CN subjects at the subject’s level to avoid data leakage (Wen et al., 2020): 50 CN subjects (50 images) compose the test set, 19 subjects (19 images) belong to the validation set and 178 subjects (452 images) are used to train our models. The split is stratified by sex and age to reduce biases. The 50 images of the CN subjects from the test set are also used to simulate the hypometabolic images, mimicking various dementias and AD severity degrees.

For the comparison to be as fair as possible, all the models share the same encoder and decoder architecture. The encoder is composed of three blocks that are the succession of a 3D convolutional layer and a batch normalization with a ReLU activation. Then the tensor is flattened and passes through a dense layer to output a one dimensional latent space. The decoder is almost symmetrical: it is composed of a dense layer followed by three blocks that are composed of a 3D deconvolutional layer and a batch normalization with a leaky ReLU activation. We tested several sizes of latent space (16, 64, 128 and 256), but as we observed similar performance, we report the results for a size of 128, consistent with the choice made in Baur et al., 2021a.

We also use the same training parameters and environment to train all the models. We trained each model on 300 epochs with a learning rate of 10^{-5} and a batch size of 24 on an HPC with Nvidia Tesla V100 GPUs that have 32GB of memory. We are aware that model performance can greatly vary depending on these parameters, but for fair comparison, we decided to choose the best parameters on the VAE and use the same for all models. It takes on average between 1’ and 1’30” to train one epoch with comparable performance for each model on our computer cluster, meaning around 7 h per model for 300 epochs.

VAE-based model implementation relies on Pythae (Chadebec et al., 2022) and neuroimage processing on ClinicaDL (Thibeau-Sutre et al., 2022b), two open source software tools. The code used for this study is available on GitHub and can be used to reproduce the experiments: <https://github.com/ravih18/VAE-models-for-UAD>.

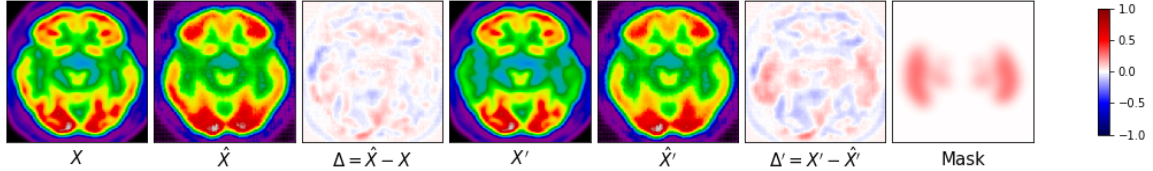


FIGURE D.1: Example of FDG PET image of a CN subject (X) with the corresponding pseudo-healthy reconstruction (\hat{X}) and difference image (Δ), followed by an image simulating AD hypometabolism obtained from X (X') with the corresponding pseudo-healthy reconstruction (\hat{X}') and difference image (Δ'), and the mask used to generate X' (M). The pseudo-healthy reconstructions were obtained from the vanilla VAE model.

D.3 Results

D.3.1 Pseudo-healthy reconstruction from images of control subjects

We first assessed whether the different models could preserve the subject’s identity by computing the MSE, PSNR and SSIM between the input and reconstructed images of the CN subjects. Results are reported in Table D.1. We observe that no model clearly outperforms the others. On the other hand, VAMP (Tomczak et al., 2018), VAE LinNF (Rezende et al., 2015), MS-SSIM VAE (Snell et al., 2017) and VAEGAN (Larsen et al., 2016) perform less well than the others (MSE > 0.05, PSNR < 20 dB, SSIM < 0.5). A possible explanation is that the dataset is too small for these models to learn the data distribution.

The other models obtain a similar performance with, on average, an MSE < 0.04, PSNR > 24 dB and SSIM comprised between 0.69 and 0.75. Not surprisingly, the AE leads to a good performance for this reconstruction task according to the MSE, as it is the optimized metric. The vanilla VAE (Kingma et al., 2014) seems to be one of the best models but does not stand out from the other models. It is probable that some models would benefit from hyper-parameter fine-tuning to perform better, but it is interesting to see that optimal parameters obtained on classic computer vision datasets do generalize to this different application for many models.

D.3.2 Pseudo-healthy reconstruction from images simulating dementia

In the following, we discarded the four models that did not give acceptable reconstructions. We first report, for the five dementia subtypes considered simulated with a hypometabolism of 30%, the MSE and SSIM between the simulated image and their reconstructions within the binary mask where hypometabolism was applied (e.g. between X' and \hat{X}' within the binarized mask M in Fig. D.1). All the models reach a very similar performance with an MSE on average across models of 0.0132 (min MSE of 0.0096 for the RAE GP (Ghosh et al., 2019) and max MSE of 0.0183 for the IWAE (Burda et al., 2016)) and an average SSIM of 0.710 (min SSIM of 0.684 for the IWAE (Burda et al., 2016) and max SSIM of 0.733 for the RAE- ℓ^2 (Ghosh et al., 2019)). This means that the VAE-based models can generalize to various kinds of anomalies located in different parts of the brain, and that none of the tested models can be selected based on this criteria. The average MSE over all the models and all the dementia subtypes (between X' and \hat{X}') is 0.0132 in the pathological masks M against 0.0072 outside the masks, which makes a 58.6% difference between both regions.

TABLE D.1: Reconstruction metrics computed between the pseudo-healthy reconstructions obtained with the various models evaluated and the original healthy PET image of CN subjects from the test set. Light gray highlights the worst performing models.

Model	MSE ↓	PSNR (dB) ↑	SSIM ↑
AE	0.02694 ± 0.00603	25.78 ± 0.84	0.725 ± 0.033
VAE (Kingma et al., 2014)	0.02471 ± 0.00517	26.15 ± 0.79	0.771 ± 0.027
VAMP (Tomczak et al., 2018)	1.09029 ± 0.10416	9.64 ± 0.41	0.057 ± 0.015
RAE-GP (Ghosh et al., 2019)	0.02363 ± 0.00480	26.34 ± 0.79	0.750 ± 0.030
RAE- ℓ^2 (Ghosh et al., 2019)	0.02385 ± 0.00532	26.31 ± 0.83	0.761 ± 0.029
VQVAE (Van Den Oord et al., 2017)	0.02645 ± 0.00608	25.87 ± 0.85	0.731 ± 0.032
IWAE (Burda et al., 2016)	0.03531 ± 0.00711	24.60 ± 0.80	0.692 ± 0.030
VAE LinNF (Rezende et al., 2015)	0.12887 ± 0.02875	18.99 ± 0.89	0.483 ± 0.036
VAE-IAF (Kingma et al., 2016)	0.02900 ± 0.00560	25.45 ± 0.77	0.706 ± 0.032
β -VAE (Higgins et al., 2017)	0.03927 ± 0.00654	24.12 ± 0.71	0.708 ± 0.028
β -TC VAE (Chen et al., 2018a)	0.02819 ± 0.00499	25.55 ± 0.67	0.729 ± 0.031
FactorVAE (Kim et al., 2018)	0.02869 ± 0.00550	25.49 ± 0.74	0.704 ± 0.032
InfoVAE (Zhao et al., 2019)	0.03223 ± 0.00566	24.97 ± 0.69	0.706 ± 0.030
WAE (Tolstikhin et al., 2018)	0.02920 ± 0.00509	25.40 ± 0.66	0.690 ± 0.032
AAE (Makhzani et al., 2015)	0.02919 ± 0.00597	25.43 ± 0.81	0.709 ± 0.032
MS-SSIM VAE (Sneil et al., 2017)	1.22541 ± 0.18918	9.17 ± 0.73	0.167 ± 0.027
VAEGAN (Larsen et al., 2016)	0.86575 ± 0.03080	10.63 ± 0.15	0.073 ± 0.014

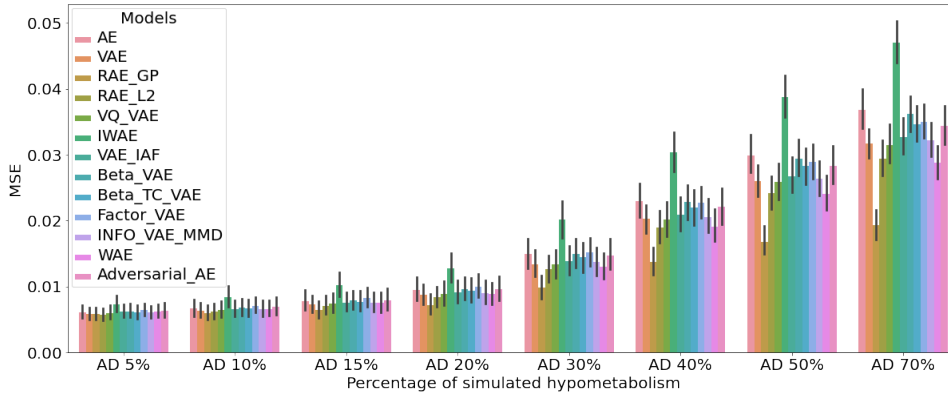


FIGURE D.2: Bar plot of the evolution of the MSE when computed within the mask characteristic of AD between the image simulated with different degrees of hypometabolism and its reconstruction. We observe that most models can scale to large anomalies.

The average SSIM is 0.710 inside masks M against 0.772 outside the masks for a 8.4% difference. This shows that the reconstruction error is much larger in regions that have been used for hypometabolism simulation, as expected. For comparison, the percentage difference is only 10.2% for the MSE and 0.2% for the SSIM when computed between the pseudo-healthy reconstruction \widehat{X}' and the real pathology-free images X . This illustrates that the models are all capable of reconstructing the pathological regions as healthy.

We then report in Fig.D.2 the MSE within the mask simulating AD when generating hypometabolism of various degrees (5% to 70%) for each model. It is interesting to observe that most of the models could be used to detect anomalies of higher intensity, as they have an increasing difference in terms of MSE for hypometabolism of 20% and more. The same trend was observed with the SSIM. The RAE- ℓ^2 (Ghosh et al., 2019) does not scale as well as other models, probably because the regularization is done on the decoder weights, so nothing prevents the encoder from learning a posterior that is less general. We also notice that the IWAE (Burda et al., 2016) has a worse reconstruction on the pathological region compared to other models, and this becomes more pronounced when the severity of the disease is increased. However, this does not mean that IWAE (Burda et al., 2016) better detects pathological areas since the reconstruction is poor in the whole image as well, meaning that IWAE (Burda et al., 2016) cannot perform well when the image is out of the training distribution. Surprisingly, the simple autoencoder gives similar results as other methods.

D.4 Conclusion

The proposed benchmark aimed to introduce the use of recent VAE variants with medical imaging data of high dimension and compare their performance on the detection of dementia-related anomalies on 3D FDG PET brain images. We observed that most models have a comparable reconstruction ability when fed with images of healthy subjects, and that their outputs correspond to healthy looking images when fed with images simulating anomalies. Exceptions are the VAEGAN (Larsen et al., 2016), VAMP (Tomczak et al., 2018), VAE

LinNF (Rezende et al., 2015), MS-SSIM VAE (Snell et al., 2017), RAE- ℓ^2 (Ghosh et al., 2019) and IWAE (Burda et al., 2016). Thanks to the evaluation framework that consists in simulating images with anomalies from pathology-free images, we showed that most models can generalize pseudo-healthy reconstruction to different dementias and different severity degrees. These results are interesting as it means that VAE-based models developed for natural images can generalize well to other tasks (here 3D brain imaging): they are easy to use and do not necessarily require a large training set, which might not be the case for other types of generative models. We also showed that in our scenario (small dataset of complex 3D images) the simplest models (vanilla AE and VAE) lead to results comparable to that of the more complex ones. Nevertheless, the results are for now limited to the detection of simulated anomalies. An evaluation on real images would be necessary to confirm these observations.

The proposed benchmark could be used in future work to assess whether the posterior learned by the different models is the same for images from healthy and diseased subjects using the simulation framework to compare the latent representation of both the original and simulated images, thus explaining the results of the models. It would also be interesting to compare some of the VAE-based models to GANs or diffusion models, and assess whether it would be possible to improve reconstruction quality while learning the distribution of healthy subject images.

Appendix E

Description of the VAE variants and of their hyper-parameter selection procedure

This Appendix describes all the VAE variants. Hyper-parameters were chosen following implementations and recommendations from the original papers and the benchmark previously done by Chadebec et al., 2022. The results of the random searches are reported for each of the models. For each model, we selected the configuration with the best average SSIM on the validation folds.

E.1 Adversarial Autoencoder

The adversarial autoencoder (Makhzani et al., 2015) is a probabilistic autoencoder model that uses the GAN framework to perform variational inference in the latent space. It uses a discriminator network to differentiate a prior’s sample from a posterior’s sample as a form of regularization. Its objective and training are quite similar to that of a VAE

$$\mathcal{L}_{\text{Adv. AE}} = \mathbb{E}_{z \sim q_{\phi}(z|x)} [\log p_{\theta}(x|z)] + \alpha \mathcal{L}_{\text{GAN}} ,$$

where

$$\mathcal{L}_{\text{GAN}} = \mathbb{E}_{\tilde{z} \sim p_z(z)} [\log(1 - D(\tilde{z}))] + \mathbb{E}_{x \sim p_{\theta}} \left[\mathbb{E}_{z \sim q_{\phi}(z|x)} [\log D(z)] \right] .$$

We set the discriminator to be the same as in Chadebec et al., 2022, that is, a multilayer perceptron with a single hidden layer with 256 units and ReLU activation. We performed a grid search of 10 configurations for

$$\alpha \in \{0.001, 0.01, 0.05, 0.1, 0.25, 0.5, 0.75, 0.9, 0.95, 0.99\} .$$

The results are reported in Table E.1.

TABLE E.1: Results of the random search on the hyper-parameters of the Adv. AE: ranking according to the SSIM of the 10 best configurations (mean \pm std over the three folds randomly selected).

adversarial loss scale	SSIM \uparrow	MSE ($\times 10^{-3}$) \downarrow
0.9	0.873 ± 0.005	1.770 ± 0.083
0.01	0.872 ± 0.003	1.771 ± 0.144
0.1	0.869 ± 0.005	1.846 ± 0.115
0.5	0.869 ± 0.015	1.811 ± 0.094
0.75	0.869 ± 0.006	1.784 ± 0.142
0.25	0.866 ± 0.002	1.841 ± 0.155
0.99	0.865 ± 0.014	1.863 ± 0.094
0.05	0.863 ± 0.009	1.779 ± 0.103
0.001	0.863 ± 0.007	1.860 ± 0.075
0.95	0.856 ± 0.001	1.814 ± 0.118

E.2 β -TC VAE

The β -TCVAE, or Total Correlation VAE (Chen et al., 2018a), is an extension of the β -VAE (Higgins et al., 2017), which aims at further isolating sources of disentanglement by rewriting the ELBO in the following way:

$$\mathcal{L}_{\beta\text{-TCVAE}} = \mathbb{E}_{z \sim q_\phi(z|x)} [\log p_\theta(x|z)] - \mathcal{L}_{\text{reg}} ,$$

where

$$\mathcal{L}_{\text{reg}} = \alpha \mathcal{D}_{\text{KL}} [q_\phi(z, x) || q_\phi(z) p_\theta(x)] + \beta \mathcal{D}_{\text{KL}} \left[q_\phi(z) || \prod_j q_\phi(z_j) \right] + \gamma \sum_j \mathcal{D}_{\text{KL}} [q_\phi(z_j) || p_z(z_j)] .$$

The regularization term is therefore the sum of the mutual information between x and z , the total correlation, which models the dependence between dimensions of the latent vector, and the dimension-wise KL divergence, which prevents each dimension of the latent variable from diverging too far from its prior.

Following the authors' suggestion, we set $\alpha = \gamma = 1$ for most of the models and performed a grid search for parameter $\beta \in \{0.001, 0.005, 0.01, 0.05, 0.1, 1, 2, 5, 10\}$. We also tried the configurations $(\beta, \alpha, \gamma) = (1, 1, 3)$ and $(\beta, \alpha, \gamma) = (1, 3, 1)$, which made a total of 12 configurations. The results are reported in Table E.2.

TABLE E.2: Results of the random search on the hyper-parameters of the β -TC VAE: ranking according to the SSIM of the 10 best configurations (mean \pm std over the three folds randomly selected).

β	α	γ	SSIM \uparrow	MSE ($\times 10^{-3}$) \downarrow
2	1	1	0.870 ± 0.002	1.901 ± 0.123
0.05	1	1	0.866 ± 0.002	1.953 ± 0.082
1	1	3	0.866 ± 0.003	1.993 ± 0.077
5	1	1	0.864 ± 0.004	1.923 ± 0.072
0.005	1	1	0.864 ± 0.009	1.871 ± 0.113
0.001	1	1	0.863 ± 0.005	1.903 ± 0.138
1	3	1	0.862 ± 0.010	1.810 ± 0.034
10	1	1	0.862 ± 0.008	1.969 ± 0.095
0.01	1	1	0.860 ± 0.010	1.917 ± 0.096
0.1	1	1	0.855 ± 0.010	1.864 ± 0.100

E.3 β -VAE

The β -VAE (Higgins et al., 2017) was introduced to encourage the disentanglement of features in the latent space by adding a weight β in front of the KL term to adjust the balance between reconstruction and regularization. The objective is:

$$\mathcal{L}_{\beta\text{-VAE}} = \mathbb{E}_{z \sim q_\phi(z|x)} [\log p_\theta(x|z)] - \beta \mathcal{D}_{\text{KL}} [q_\phi(z|x) || p_z(z)] \quad ,$$

where setting $\beta > 1$ leads to stronger disentanglement whereas using a smaller β can favor better reconstruction abilities.

We performed a grid search of 10 configurations for $\beta \in \{0.001, 0.005, 0.01, 0.05, 0.1, 0.5, 2, 5, 10, 100\}$. The results are reported in Table E.3.

TABLE E.3: Results of the random search on the hyper-parameters of the β -VAE: ranking according to the SSIM of the 10 best configurations (mean \pm std over the three folds randomly selected).

β	SSIM \uparrow	MSE ($\times 10^{-3}$) \downarrow
10	0.868 ± 0.003	1.995 ± 0.067
0.005	0.868 ± 0.006	1.785 ± 0.142
0.01	0.867 ± 0.006	1.755 ± 0.122
0.001	0.866 ± 0.005	1.825 ± 0.072
0.05	0.866 ± 0.008	1.859 ± 0.090
2	0.863 ± 0.009	1.9 ± 0.072
0.1	0.863 ± 0.011	1.894 ± 0.103
0.5	0.858 ± 0.011	1.835 ± 0.117
5	0.856 ± 0.011	1.969 ± 0.095
100	0.816 ± 0.008	3.716 ± 0.292

E.4 Disentangled β -VAE

The disentangled β -VAE (Burgess et al., 2018) introduces a way to progressively increase the latent encoding capacity to improve the reconstruction accuracy in comparison with the

β -VAE (Higgins et al., 2017). The objective becomes

$$\mathcal{L}_{\text{disentangled } \beta\text{-VAE}} = \mathcal{L}_{\text{rec}} - \beta |\mathcal{D}_{KL}(q_\phi(z|x)||p(z)) - C| ,$$

with C the value of the KL divergence term we would like to approach.

We performed a random search on the three parameters: $\beta \in \{10^{-2}, 10^{-1}, 1, 5, 10\}$, $C \in \{5, 25, 50\}$ and the number of epochs (warm-up epochs) during which the KL divergence in the ELBO will increase from 0 to C , which can be 100 or 1000. We trained a total of 20 configurations (out of 60 possible combinations), and the results of the random search are given in the Table E.4.

TABLE E.4: Results of the random search on the hyper-parameters of the Dis. β -VAE: ranking according to the SSIM of the 10 best configurations (mean \pm std over the three folds randomly selected).

β	C	warm-up epoch	SSIM \uparrow	MSE ($\times 10^{-3}$) \downarrow
10	50	1000	0.874 ± 0.006	2.004 ± 0.153
0.1	25	1000	0.873 ± 0.002	1.821 ± 0.056
1	50	100	0.873 ± 0.007	1.852 ± 0.092
0.01	5	1000	0.871 ± 0.004	1.755 ± 0.055
0.1	50	100	0.871 ± 0.009	1.869 ± 0.073
0.01	25	1000	0.870 ± 0.003	1.753 ± 0.074
10	5	1000	0.870 ± 0.003	2.053 ± 0.110
0.1	5	1000	0.869 ± 0.014	1.815 ± 0.036
1	5	100	0.869 ± 0.008	1.879 ± 0.064
5	25	1000	0.867 ± 0.002	2.009 ± 0.068

E.5 Factor VAE

Kim et al., 2018 proposed a new metric for disentanglement that encourages the latent representation to be factorial, and independent across each dimension of the latent space. The loss function is the following:

$$\mathcal{L}_{\text{FactorVAE}} = \mathcal{L}_{\text{VAE}} - \gamma \mathcal{D}_{KL}(q_\phi(z)||\bar{q}_\phi(z)) ,$$

with $\bar{q}_\phi(z) := \prod_{j=1}^d q_\phi(z_j)$ for a model with a latent space of dimension d .

We performed a grid search of 10 configurations to find the optimal $\gamma \in \{2, 5, 10, 15, 20, 30, 40, 50, 100, 200\}$. The results are reported in Table E.5.

TABLE E.5: Results of the random search on the hyper-parameters of the FactorVAE: ranking according to the SSIM of the 10 best configurations (mean \pm std over the three folds randomly selected).

γ	SSIM \uparrow	MSE ($\times 10^{-3}$) \downarrow
40	0.876 \pm 0.003	1.895 \pm 0.084
100	0.875 \pm 0.007	1.827 \pm 0.092
15	0.874 \pm 0.004	1.872 \pm 0.048
20	0.869 \pm 0.010	1.875 \pm 0.090
50	0.866 \pm 0.011	1.850 \pm 0.070
200	0.864 \pm 0.008	1.820 \pm 0.032
10	0.864 \pm 0.011	1.859 \pm 0.086
30	0.864 \pm 0.020	1.805 \pm 0.096
5	0.862 \pm 0.019	1.890 \pm 0.075
2	0.852 \pm 0.016	1.901 \pm 0.081

E.6 Hamiltonian VAE

Caterini et al. introduced a new method to obtain a low variance unbiased estimation of the ELBO using Markov chain Monte Carlo with Hamiltonian importance sampling (Neal, 2005) and by proposing a method to select optimal reverse kernels, building the Hamiltonian VAE (Caterini et al., 2018) with the following loss:

$$\mathcal{L}_{HVAE} = \mathbb{E}_{z_0 \sim q_{\theta, \phi}^0(\dots)} \left[\log p_{\theta}(x, z_K) - \frac{1}{2} \rho_K^T \rho_K - \log q_{\theta, \phi}^0(z_0) \right] + \frac{l}{2}$$

where $(z_0, \rho_0) = \mathcal{H}_{\theta, \phi}(z_0, \gamma_0 / \sqrt{\beta_0})$, \mathcal{H} is the Hamiltonian importance sampling (Neal, 2005), β_0 is the inverse temperature and $\gamma_0 \sim \mathcal{N}(\cdot | 0, I)$.

There are three hyper-parameters that we randomly searched for: the number of step in the leapfrog $n_{lf} \in \{1, 2, 10, 15, 20\}$, the leapfrog step size $\epsilon_{lf} \in \{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}\}$ and $\beta_0 \in \{0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0\}$ the tempering factor in the Hamiltonian Monte Carlo Sampler. We trained 20 configurations out of 220 possible combinations. The results are reported in Table E.6. Note that some configurations were really long to train, sometimes exceeding the time limit of the HPC used to train the models.

TABLE E.6: Results of the random search on the hyper-parameters of the HVAE: ranking according to the SSIM of the 10 best configurations (mean \pm std over the three folds randomly selected).

n_{lf}	ϵ_{lf}	β_0	SSIM \uparrow	MSE ($\times 10^{-3}$) \downarrow
10	0.00001	0.8	0.873 \pm 0.007	1.862 \pm 0.068
2	0.00001	0.7	0.870 \pm 0.002	1.905 \pm 0.079
2	0.001	0.2	0.870 \pm 0.004	1.847 \pm 0.082
1	0.001	0.5	0.869 \pm 0.008	1.853 \pm 0.101
15	0.00001	0.2	0.868 \pm 0.009	1.854 \pm 0.075
1	0.001	0.7	0.865 \pm 0.004	1.890 \pm 0.102
2	0.01	1	0.865 \pm 0.005	1.911 \pm 0.066
15	0.001	0.4	0.865 \pm 0.008	1.805 \pm 0.024
15	0.001	0.1	0.864 \pm 0.009	1.908 \pm 0.084
10	0.0001	0.9	0.863 \pm 0.003	1.882 \pm 0.104

E.7 Info VAE MMD

To improve both the generative model and the amortized inference distribution, Zhao et al., 2019 proposed to add the mutual information between z and x in the objective function of the VAE. To be optimized, the loss is rewritten as follows:

$$\mathcal{L}_{InfoVAE} = \mathbb{E}_{p_{\mathcal{D}(x)}} \mathbb{E}_{q_{\phi}(z|x)} [\log p_{\theta}(x|z)] - (1-\alpha) \mathbb{E}_{p_{\mathcal{D}(x)}} \mathcal{D}_{KL}(q_{\phi}(z|x)||p(z)) - (\alpha+\lambda-1) \mathcal{D}(q_{\phi}(z)||p(z))$$

with \mathcal{D} the maximum mean discrepancy (MMD).

We performed a random search of the following parameters: $\alpha \in \{0.0, 0.2, 0.4, 0.6, 0.8, 1.0\}$, $\lambda \in \{0.01, 0.1, 1, 10, 100\}$, the choice of the kernel for the MMD $\in \{\text{rbf}, \text{imq}\}$ (rbf: radial basis function, imq: inverse multi-quadratic) and the kernel bandwidth $\in \{0.01, 0.1, 0.5, 1, 5, 10, 100\}$. We trained 30 configurations out of 420 possible combinations. The results are reported in Table E.7.

TABLE E.7: Results of the random search on the hyper-parameters of the InfoVAE: ranking according to the SSIM of the 10 best configurations (mean \pm std over the three folds randomly selected).

kernel choice	α	λ	kernel bandwidth	SSIM \uparrow	MSE ($\times 10^{-3}$) \downarrow
rbf	1	0.1	0.1	0.877 ± 0.006	1.813 ± 0.075
rbf	0.4	1	0.5	0.875 ± 0.006	1.804 ± 0.052
rbf	1	0.1	0.5	0.874 ± 0.003	1.770 ± 0.077
rbf	0.00001	100	1	0.873 ± 0.002	1.852 ± 0.095
rbf	0.00001	10	0.5	0.873 ± 0.004	1.846 ± 0.094
imq	0.4	100	0.1	0.872 ± 0.008	1.866 ± 0.045
imq	1	10	1	0.872 ± 0.006	1.830 ± 0.068
rbf	0.6	0.01	5	0.871 ± 0.007	1.832 ± 0.088
rbf	1	100	0.01	0.870 ± 0.004	1.768 ± 0.079
imq	0.2	0.01	0.01	0.870 ± 0.005	1.830 ± 0.107

E.8 IWAE

Instead of relying on a single sample for estimating the posterior, the IWAE (Burda et al., 2016) utilizes importance weights during the sampling process in the latent space on multiple samples (Monte Carlo estimator), assigning higher weights to more probable samples. This provides a new ELBO that becomes tighter when the number of samples increases. The loss is the following:

$$\mathcal{L}_{IWAE} = \mathbb{E}_{z_1, \dots, z_k \sim q_{\phi}(z|x)} \left[\log \frac{1}{k} \sum_{i=1}^k \frac{p_{\theta}(x, z_i)}{q_{\phi}(z_i|x)} \right]$$

with $k \in \{2, 3, 4, 5, 6, 8, 10, 12, 15, 20\}$ the number of samples to use in the Monte Carlo estimator.

When k grows, the IWAE becomes very memory greedy and time-consuming during training, especially with 3D images. We had to reduce the batch size to 2, and, in spite

of this, the model would crash because of memory when setting $k > 6$. The results are reported in Table E.7.

TABLE E.8: Results of the random search on the hyper-parameters of the IWAE: ranking according to the SSIM of the 3 best configurations (mean \pm std over the three folds randomly selected).

number of samples	SSIM \uparrow	MSE ($\times 10^{-3}$) \downarrow
6	0.865 ± 0.007	2.087 ± 0.146
3	0.861 ± 0.002	2.048 ± 0.064
4	0.854 ± 0.013	2.178 ± 0.201

E.9 MS-SSIM VAE

Snell et al., 2017 proposed an extension of the VAE, called the expected loss VAE, where the pixel-wise reconstruction loss can be replaced by any deterministic reconstruction loss. For this, the probabilistic decoder p_θ is replaced by a deterministic equivalent f_θ so that the reconstruction \hat{x} of x given $z \sim q_\phi(z|x)$ is given by $\hat{x} = f_\theta(x)$ and the reconstruction loss is given by $\Delta(x, \hat{x})$. The objective becomes

$$\mathcal{L}_{\text{EL-VAE}} = \mathbb{E}_{q_\phi(z|x)} [\Delta(x, \hat{x})] - \beta \mathcal{D}_{\text{KL}} [q_\phi(z|x) || p_z(z)].$$

Following the authors' suggestion, we use the MS-SSIM, or multi-scale structural similarity, as our reconstruction loss.

We performed a random search on β and the window size used in the computations of the MS-SSIM, where β is sampled from $\{0.01, 0.1, 1, 10, 100\}$ and the window size is sampled from $\{2, 3, 5, 11\}$. We trained 10 configurations out of 20 possible combinations. The results are reported in Table E.9.

The training time was too long for configurations with a window size different from 2, explaining why Table E.9 contains only five configuration with a window size of 2.

TABLE E.9: Results of the random search on the hyper-parameters of the MS-SSIM VAE: ranking according to the SSIM of the 5 best configurations (mean \pm std over the three folds randomly selected).

β	window size	SSIM \uparrow	MSE ($\times 10^{-3}$) \downarrow
100	2	0.472 ± 0.034	70.174 ± 5.660
1	2	0.453 ± 0.050	75.443 ± 11.098
1	2	0.448 ± 0.018	74.110 ± 1.158
1	2	0.445 ± 0.050	76.354 ± 8.802
0.01	2	0.393 ± 0.039	84.579 ± 7.183

E.10 Regularized auto-encoder

Ghosh et al., 2019 claimed that the probabilistic sampling in VAE is equivalent to a noise injection to the decoder, acting as a stochastic regularization of the latent space. The authors proposed a new approach that consists in replacing the random noise injection by a deterministic regularization in the decoder. The training objective becomes

$$\mathcal{L}_{RAE} = \|x - \hat{x}\|_2^2 + \beta \cdot \mathcal{L}_Z^{RAE} + \lambda \cdot \mathcal{L}_{REG} ,$$

with \mathcal{L}_{REG} the regularization term for the decoder and $\mathcal{L}_Z^{RAE} = 1/2\|z\|_2^2$ a constraint on the latent space. The authors suggested two different regularization terms for the decoder:

- the first option is a \mathcal{L}_2 norm on the weights of the decoder $\mathcal{L}_{REG} = \|\theta\|_2^2$, giving the RAE- ℓ^2 model;
- another choice is to apply a gradient penalty on the discriminator $\mathcal{L}_{REG} = \|\nabla D_\theta(E_\phi(x))\|_2^2$, giving the RAE-GP model.

We performed a random search on both λ and β , that are both sampled from $\{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}\}$. We trained 20 configurations for both the RAE- ℓ^2 and the RAE-GP out of 36 possible combinations for each model. The results are respectively reported in Tables E.10 and E.11.

TABLE E.10: Results of the random search on the hyper-parameters of the RAE- ℓ^2 : ranking according to the SSIM of the 10 best configurations (mean \pm std over the three folds randomly selected).

embedding weight	reg weight	SSIM \uparrow	MSE ($\times 10^{-3}$) \downarrow
0.0001	1	0.884 \pm 0.005	1.815 \pm 0.049
0.0001	0.001	0.883 \pm 0.002	1.765 \pm 0.070
0.0001	1	0.879 \pm 0.008	1.848 \pm 0.059
0.0001	0.01	0.879 \pm 0.009	1.857 \pm 0.064
0.00001	0.01	0.879 \pm 0.007	1.868 \pm 0.055
0.1	0.001	0.878 \pm 0.007	1.814 \pm 0.052
0.00001	0.01	0.878 \pm 0.006	1.785 \pm 0.076
0.1	0.0001	0.878 \pm 0.007	1.853 \pm 0.077
0.00001	0.1	0.878 \pm 0.007	1.783 \pm 0.107
0.00001	0.01	0.877 \pm 0.005	1.831 \pm 0.121

TABLE E.11: Results of the random search on the hyper-parameters of the RAE-GP: ranking according to the SSIM of the 10 best configurations (mean \pm std over the three folds randomly selected).

embedding weight	reg weight	SSIM \uparrow	MSE ($\times 10^{-3}$) \downarrow
0.01	0.0001	0.880 ± 0.006	1.715 ± 0.105
0.0001	0.0001	0.877 ± 0.008	1.744 ± 0.056
0.1	0.00001	0.877 ± 0.009	1.820 ± 0.093
1	0.001	0.867 ± 0.003	1.756 ± 0.063
0.1	0.01	0.861 ± 0.011	1.828 ± 0.031
0.1	0.1	0.845 ± 0.012	1.750 ± 0.107
0.0001	0.1	0.842 ± 0.010	1.769 ± 0.079
0.1	0.1	0.839 ± 0.008	1.799 ± 0.101
0.00001	1	0.825 ± 0.013	1.906 ± 0.135
0.1	1	0.808 ± 0.004	1.924 ± 0.145

E.11 Hyperspherical VAE

The hyperspherical VAE (Davidson et al., 2018) uses a von Mises-Fisher (vMF) distribution as prior, leading to a hyperspherical latent space. This model has the advantage of not having additional hyper-parameters compared to a standard VAE but only works with a small latent space as large values lead to errors when computing the modified Bessel function involved in the probability density function of the vMF distribution. Therefore, we performed a grid search on three different smaller latent space sizes: 8, 16, 32. The results are reported in Table E.12.

TABLE E.12: Results of the random search on the hyper-parameters of the SVAE: ranking according to the SSIM of the 3 best configurations (mean \pm std over the three folds randomly selected).

latent space size	SSIM \uparrow	MSE ($\times 10^{-3}$) \downarrow
16	0.151 ± 0.001	632.694 ± 5.106
32	0.150 ± 0.002	640.791 ± 4.328
8	0.083 ± 0.028	189.998 ± 68.394

E.12 VAEGAN

In the VAE-GAN (Larsen et al., 2016), a discriminator is trained on the output of a VAE to enhance the VAE’s reconstruction abilities. The idea is to use the learned feature representations from intermediate layers of the GAN discriminator as a basis for the VAE reconstruction objective, assuming that the discriminator can capture high-level structures relevant to the data distribution. Overall, this allows replacing voxel-wise similarity between input and output by feature-wise similarity. For $z \sim p_z(z)$ and $\hat{x} \sim D_\theta(z)$, the objective is given by

$$\mathcal{L}_{\text{VAE-GAN}} = \mathbb{E}_{z \sim q_\phi(z|x)} [\log \mathcal{N}(D_l(x) | D_l(\hat{x}), \mathbf{I})] - \mathcal{D}_{\text{KL}} [q_\phi(z|x) || p_z(z)] - \log \left(\frac{D(x)}{1 - D(D_\theta(z))} \right),$$

where D denotes the discriminator, D_l the hidden representation of the l -th layer of the discriminator, and D_θ the decoder. We also added a hyper-parameter α to the decoder’s loss, such that high values of α encourage better reconstruction with respect to the features learned at the layer l of the discriminator.

We set the discriminator to be a neural network with 4 convolutions and 2 fully connected layers, with batch normalization and ReLU activation. We set the margin to 0.4 and the equilibrium to 0.68 as in the original paper. We performed a random search for $\alpha \in \{0.3, 0.5, 0.7, 0.9\}$ and $l \in \{1, 2, 3, 4\}$. This model is particularly long to train and, due to memory constraints, we reduced the batch size to 4 instead of 8 for these models. We trained 10 different configurations out of 16 possible combinations. The results are reported in Table E.13. We note that there is a strong correlation between the chosen reconstruction layer in the decoder and the quality of the reconstruction.

TABLE E.13: Results of the random search on the hyper-parameters of the VAEGAN: ranking according to the SSIM of the 10 best configurations (mean \pm std over the three folds randomly selected).

adversarial loss scale	reconstruction layer	SSIM \uparrow	MSE ($\times 10^{-3}$) \downarrow
0.5	1	0.860 \pm 0.013	2.241 \pm 0.193
0.1	1	0.851 \pm 0.015	2.671 \pm 0.090
0.1	1	0.849 \pm 0.006	2.547 \pm 0.356
0.3	2	0.799 \pm 0.050	3.705 \pm 0.559
0.3	2	0.780 \pm 0.101	3.968 \pm 0.642
0.9	3	0.714 \pm 0.148	9.727 \pm 4.423
0.7	3	0.692 \pm 0.101	9.832 \pm 1.185
0.8	3	0.572 \pm 0.061	10.336 \pm 5.712
0.9	4	0.560 \pm 0.113	24.463 \pm 6.516

E.13 VAE with inverse auto-regressive flows

The VAE with inverse auto-regressive flows (Kingma et al., 2016) incorporates a series of inverse auto-regressive flows in the encoder, enhancing the flexibility of the learned posterior distribution, and scaling well to high-dimensional latent spaces. We use masked autoencoder for distribution estimation (MADE) (Germain et al., 2015) as normalizing flow, as suggested in Kingma et al., 2016 and implemented in Pythae (Chadebec et al., 2022).

We performed a random search on the following parameters: the number of MADE blocks $\in \{2, 3, 4, 5, 6, 8\}$, the number of hidden layers in the MADE blocks $\in \{2, 3, 4, 5\}$ and the size of the hidden layers $\in \{64, 128, 256\}$. We trained 30 different configurations out of 72 possible combinations. However, we noticed that the performance was really poor when the number of MADE blocks was odd, reducing the possible values for this parameter to even numbers. The results are reported in Table E.14.

TABLE E.14: Results of the random search on the hyper-parameters of the VAE-IAF: ranking according to the SSIM of the 10 best configurations (mean \pm std over the three folds randomly selected).

n MADE blocks	n hidden in MADE	hidden size	SSIM \uparrow	MSE ($\times 10^{-3}$) \downarrow
4	4	128	0.823 ± 0.005	2.272 ± 0.057
4	5	256	0.823 ± 0.008	2.248 ± 0.063
2	5	256	0.823 ± 0.007	2.220 ± 0.071
8	4	128	0.822 ± 0.005	2.282 ± 0.092
6	5	128	0.820 ± 0.004	2.259 ± 0.148
6	5	64	0.820 ± 0.003	2.403 ± 0.107
4	3	128	0.819 ± 0.008	2.260 ± 0.055
4	2	64	0.818 ± 0.015	2.331 ± 0.133
4	5	64	0.817 ± 0.006	2.359 ± 0.114
8	5	64	0.816 ± 0.005	2.546 ± 0.028

E.14 VAE with linear normalizing flows

The VAE with linear normalizing flows (Rezende et al., 2015) enables a better approximation of the posterior distribution $q_\phi(z|x)$ using a series of linear normalizing flows, which are invertible transformations. To get the latent vector z_K , z_0 is sampled from $q_\phi(z|x)$ and passes through K linear normalizing flows f_k such that $z_K = f_K \circ \dots \circ f_2 \circ f_1(z_0)$. These flows enable the model to capture complex distributions in the latent space. The authors suggest using a succession of linear flows, and more precisely planar or radial flows, because it is computationally efficient to compute their Jacobian, as needed to compute the loss

$$\mathcal{L}_{VAElinNF} = \mathcal{L}_{rec} + \log q_\phi(z_0) - \log q(z_K) - \sum_{k=1}^K \log \left| \det \frac{\partial f_k}{\partial z} \right| .$$

We tried 10 different configurations of flows $\in \{10P, 10R, 5P, 5R, 5P5R, 5R5P, 5PR, 5RP, 2PR, 2RP\}$, with P designating a planar flow and R a radial flow. The results are reported in Table E.15. We note that configurations with radial flows clearly outperform configurations with planar flows.

TABLE E.15: Results of the random search on the hyper-parameters of the VAE LinNF: ranking according to the SSIM of the nine best configurations (mean \pm std over the three folds randomly selected). P designate a planar flow, R designate a radial flow.

flows	SSIM \uparrow	MSE ($\times 10^{-3}$) \downarrow
10R	0.871 ± 0.001	1.855 ± 0.125
5R	0.860 ± 0.008	1.897 ± 0.066
2PR	0.827 ± 0.005	2.262 ± 0.135
5PR	0.737 ± 0.058	3.218 ± 0.205
5P5R	0.720 ± 0.095	3.148 ± 0.763
5RP	0.716 ± 0.112	3.829 ± 0.942
5R5P	0.708 ± 0.080	4.652 ± 1.518
5P	0.679 ± 0.098	4.897 ± 1.878
10P	0.570 ± 0.064	8.082 ± 4.095

E.15 VAE with VampPrior

The VAE with a ‘‘Variational Mixture of Posteriors’’ prior, or VampPrior (Tomczak et al., 2018), aims to replace the simple normal prior with a mixture of distributions (e.g. mixture of Gaussians), allowing capturing more complex data distributions. We optimize the following loss:

$$\mathcal{L}_{VAMP} = \mathcal{L}_{rec} - (\log p_\lambda(z) - \log q_\phi(z|x))$$

with $p_\lambda(z) = \frac{1}{K} \sum_{k=1}^K q_\phi(z|u_k)$, K being the number of components, and u_k being the ‘‘pseudo-input’’ learned through back-propagation.

We performed a random search on the number of components $K \in \{10, 20, 30, 40, 50\}$ and the number of linear scheduling steps $\in \{0, 20, 40\}$. The results are reported in Table E.16.

TABLE E.16: Results of the random search on the hyper-parameters of the VAMP: ranking according to the SSIM of the 10 best configurations (mean \pm std over the three folds randomly selected).

number components	linear scheduling steps	SSIM \uparrow	MSE ($\times 10^{-3}$) \downarrow
20	40	0.702 \pm 0.097	5.581 \pm 0.874
10	20	0.686 \pm 0.019	7.231 \pm 5.478
10	20	0.678 \pm 0.025	5.841 \pm 0.902
20	20	0.633 \pm 0.007	3.640 \pm 0.025
20	0	0.631 \pm 0.003	3.569 \pm 0.108
20	20	0.628 \pm 0.002	3.586 \pm 0.120
30	20	0.625 \pm 0.005	3.892 \pm 0.150
30	0	0.622 \pm 0.001	3.965 \pm 0.091
40	40	0.621 \pm 0.005	4.151 \pm 0.226
40	0	0.620 \pm 0.004	4.074 \pm 0.169

E.16 Vector-quantized VAE

Van Den Oord et al., 2017 suggested using discrete (rather than continuous) latent representations and having a learned (rather than static) prior. The latent space is structured as an $\mathbb{R}^{K \times D}$ vector space. We denote $\mathcal{E} = \{e_1, e_2, \dots, e_K\}$ where $e_i \in \mathbb{R}^D$ for $i \in \{1, 2, \dots, K\}$. We say that K is the size of the latent embedding space, and D is the dimension of the embedding vectors.

For an embedding size d , the input x is passed through the encoder to obtain the output $z_e(x) \in \mathbb{R}^{d \times D}$, which is then passed through the discretisation bottleneck to map it to an element of $z_q(x) \in \mathcal{E}^d$ such that $(z_q(x))_j = e_k$ where $k = \arg \min_l \|z_e(x) - e_l\|_2$ for $j \in \{1, 2, \dots, d\}$. As the argmin operation lacks differentiability, learning of the embeddings and regularisation of the latent space is managed by integrating the stopgradient operator sg into the training objective:

$$\mathcal{L}_{\text{VQVAE}}(x) = \log p(x|z_q(x)) + \|\text{sg}[z_e(x)] - e\|_2^2 + \beta \|z_e(x) - \text{sg}[e]\|_2^2 .$$

As suggested by the authors, we replaced the term $\|\text{sg}[z_e(x)] - e\|_2^2$ in the loss by the exponential moving average (EMA) update with a decay of 0.99. We then considered two hyper-parameters in our random search: the size of the latent embedding space $K \in \{128, 256, 512, 1024\}$ and the regularization weight $\beta \in \{0.25, 0.5, 0.75, 0.9, 1, 1.5, 2, 4\}$. The results are reported in Table E.17.

TABLE E.17: Results of the random search on the hyper-parameters of the VQVAE: ranking according to the SSIM of the 10 best configurations (mean \pm std over the three folds randomly selected).

commitment loss factor	quantization loss factor	num embeddings	use EMA	decay	SSIM \uparrow	MSE ($\times 10^{-3}$) \downarrow
0.25	2	512	True	0.99	0.881 ± 0.003	1.805 ± 0.032
0.25	0.25	1024	True	0.99	0.880 ± 0.009	1.866 ± 0.064
0.25	0.5	256	True	0.99	0.879 ± 0.005	1.797 ± 0.037
0.25	0.25	512	True	0.99	0.877 ± 0.007	1.836 ± 0.093
0.25	4	1024	True	0.99	0.876 ± 0.005	1.854 ± 0.065
0.25	0.75	256	True	0.99	0.874 ± 0.011	1.896 ± 0.065
0.25	0.9	512	True	0.99	0.870 ± 0.004	1.927 ± 0.023
0.25	1.5	256	True	0.99	0.870 ± 0.011	1.856 ± 0.100
0.25	4	1024	True	0.99	0.870 ± 0.011	1.854 ± 0.056
0.25	1.5	1024	True	0.99	0.868 ± 0.014	1.827 ± 0.084

E.17 Wasserstein auto-encoder

The Wasserstein auto-encoder (Tolstikhin et al., 2018) introduces the use of a penalized form of the Wasserstein distance to measure the dissimilarity between the model’s generated distribution and the true data distribution. This leads to more stable training, mitigating mode collapse and improving the model’s ability to generate diverse and realistic samples.

$$\mathcal{L}_{WAE} = \mathcal{L}_{rec} + \lambda \mathcal{D}_Z(p_z(z), q_\phi(z)) \text{ ,}$$

with \mathcal{D}_Z an arbitrary divergence. Different divergences \mathcal{D}_Z are suggested by the authors, we here use the maximum mean discrepancy (MMD).

We performed a random search on three parameters: the kernel choice $\in \{\text{rbf}, \text{imq}\}$, the regularization weight $\lambda \in \{0.01, 0.1, 0.5, 1, 5, 10, 100\}$ and the kernel bandwidth $\in \{0.01, 0.1, 0.5, 1, 5, 10, 100\}$. The results are reported in Table E.18.

TABLE E.18: Results of the random search on the hyper-parameters of the WAE: ranking according to the SSIM of the 10 best configurations (mean \pm std over the three folds randomly selected).

kernel choice	regularization weight	kernel bandwidth	SSIM \uparrow	MSE ($\times 10^{-3}$) \downarrow
rbf	0.1	5	0.881 ± 0.005	1.862 ± 0.075
rbf	0.5	0.1	0.880 ± 0.002	1.835 ± 0.096
rbf	0.5	0.5	0.879 ± 0.004	1.798 ± 0.035
rbf	0.01	0.1	0.879 ± 0.006	1.866 ± 0.061
imq	5	1	0.878 ± 0.003	1.838 ± 0.073
rbf	10	5	0.877 ± 0.005	1.838 ± 0.090
imq	1	1	0.876 ± 0.006	1.835 ± 0.097
imq	1	0.01	0.876 ± 0.007	1.882 ± 0.067
imq	5	100	0.874 ± 0.002	1.894 ± 0.070
imq	100	100	0.874 ± 0.016	1.865 ± 0.069

Appendix F

Details of the encoder-decoder architecture selection procedure

In previous works (Hassanaly et al., 2024b; Hassanaly et al., 2023a), we set the latent space size to 128 as a trade-off between performance and resources but observed that a larger latent space would lead to better reconstructions. We therefore decided to try sizes from the set $\{256, 512, 1024\}$. For choosing the number of blocks for the encoder and decoder, B_e and B_d , we initially tried the integer range from 3 to 7. This parameter influences the size of the last feature map before the fully connected layer, and therefore the number of parameters in that layer. We noticed that having an encoder with 3 blocks leads to a very large number of parameters in the fully connected layer (around 750,000 if $C_e = 16$ or double if we double C_e), whereas an encoder with 7 blocks would lead to very small feature maps ($1 \times 1 \times 1$). We therefore reduced the range, and chose values between 4 and 6. We kept the number of sub-blocks in the encoder and decoder S_e and S_d relatively small to restrain the number of parameters of our model whilst still testing deep architectures. We chose the number of channels C_e and C_d based on previous experiments and decided to set it to either 16 or 32. We also added the possibility to add a convolution layer in our last decoder block (shown by dotted lines in Figure 5.2). We also included the learning rate and the optimizer as parameters in our random search. We first performed experiments where the learning rate was chosen from $\{10^{-5}, 10^{-4}, 10^{-3}\}$, but setting it to 10^{-3} led to errors in the computation of the loss, so we pursued our search with only 10^{-5} and 10^{-4} as options. The optimizer could be either Adam or Adamax, following the suggestions from Vahdat et al., 2020. The parameters included in our random search are summarized in Table 5.1.

After attempting to train 200 models, a pattern emerged, and we could select and test an additional architecture following our intuition. These results are summarized in Table F.1 and Table F.2. Certain parameters, such as the learning rate, the latent space size, and the number of channels C_e and C_d were easy to choose as a clear relation with the reconstruction metrics could be observed, allowing us to choose to set the learning rate to 10^{-4} , the latent space size to 256 and $D_e = C_d = 16$. We particularly struggled to train models with residual sub-blocks due to memory constraints, and those that were able to train did not give very good results, so we focused our efforts on models with convolutional sub-blocks. The optimizer did not seem to make a large difference, so we chose Adam. For the remaining parameters, we observed that there was no need for a very deep architecture (or large number of sub-blocks within blocks), whereas a large number of blocks was beneficial in terms of

memory as it induced a smaller number of parameters before the fully connected layer. After analyzing these results and noticing these patterns, we decided to test an extra model which we designed to be symmetrical (for the sake of simplicity) and as light as possible in terms of memory (128 MB instead of 286 MB), as we knew that some of the models that we would train later with this architecture are much more memory greedy. We selected random splits by drawing 3 cards from a deck of 6 cards to train our model, and found that this model performed similarly to the best performing models from our random search (equivalent SSIM and best MSE). We therefore decided to select this architecture as it was simpler (because symmetrical) and smaller in terms of memory and number of parameters.

TABLE F.1: Results of the random search on the VAE architecture: ranking according to the SSIM of the 10 best configurations (mean \pm std over the three folds randomly selected). The selected configuration is highlighted in gray.

Id	block type	C_e	B_e	S_e	latent size	B_d	S_d	C_d	last conv	learning rate	SSIM \uparrow	MSE \downarrow
1	conv	16	5	2	256	5	2	16	True	0.0001	0.866 \pm 0.006	2.158 \pm 0.043
2	conv	16	5	2	256	4	3	32	False	0.0001	0.864 \pm 0.008	2.221 \pm 0.111
3	conv	16	5	3	1024	4	3	16	False	0.0001	0.861 \pm 0.017	2.202 \pm 0.082
4	conv	32	5	1	256	5	1	32	False	0.00001	0.861 \pm 0.004	2.081 \pm 0.021
5	conv	32	5	1	512	5	2	32	False	0.0001	0.857 \pm 0.001	2.153 \pm 0.056
6	conv	16	5	1	256	5	1	16	False	0.0001	0.856 \pm 0.004	1.919 \pm 0.088
7	conv	32	5	3	512	5	1	16	True	0.0001	0.852 \pm 0.002	2.138 \pm 0.101
8	conv	32	5	1	512	4	1	32	True	0.0001	0.851 \pm 0.006	2.303 \pm 0.187
9	conv	16	5	2	1024	4	1	16	True	0.0001	0.850 \pm 0.007	2.572 \pm 0.145
10	conv	16	5	2	1024	5	1	16	True	0.0001	0.840 \pm 0.024	2.014 \pm 0.083

TABLE F.2: Results of the random search on the VAE architecture: ranking according to the MSE of the 10 best configurations (mean \pm std over the three folds randomly selected). The selected configuration is highlighted in gray. The Id corresponds to the rank of the model when sorting them according to the SSIM (Table F.1).

Id	block type	C_e	B_e	S_e	latent size	B_d	S_d	C_d	last conv	learning rate	SSIM \uparrow	MSE \downarrow
6	conv	16	5	1	256	5	1	16	False	0.0001	0.856 \pm 0.004	1.919 \pm 0.088
10	conv	16	5	2	1024	5	1	16	True	0.0001	0.840 \pm 0.024	2.014 \pm 0.083
14	conv	32	5	1	1024	4	3	16	False	0.00001	0.826 \pm 0.013	2.062 \pm 0.065
4	conv	32	5	1	256	5	1	32	False	0.00001	0.861 \pm 0.004	2.081 \pm 0.021
11	conv	16	5	1	256	4	1	32	True	0.0001	0.800 \pm 0.049	2.081 \pm 0.107
13	conv	32	6	1	256	6	2	32	False	0.00001	0.832 \pm 0.009	2.091 \pm 0.113
12	conv	16	5	3	512	4	2	16	True	0.0001	0.818 \pm 0.032	2.111 \pm 0.069
7	conv	32	5	3	512	5	1	16	True	0.0001	0.853 \pm 0.002	2.128 \pm 0.101
5	conv	32	5	1	512	5	2	32	False	0.0001	0.857 \pm 0.001	2.153 \pm 0.056
1	conv	16	5	2	256	5	2	16	True	0.0001	0.866 \pm 0.006	2.158 \pm 0.043

Appendix G

Benchmark reconstructions

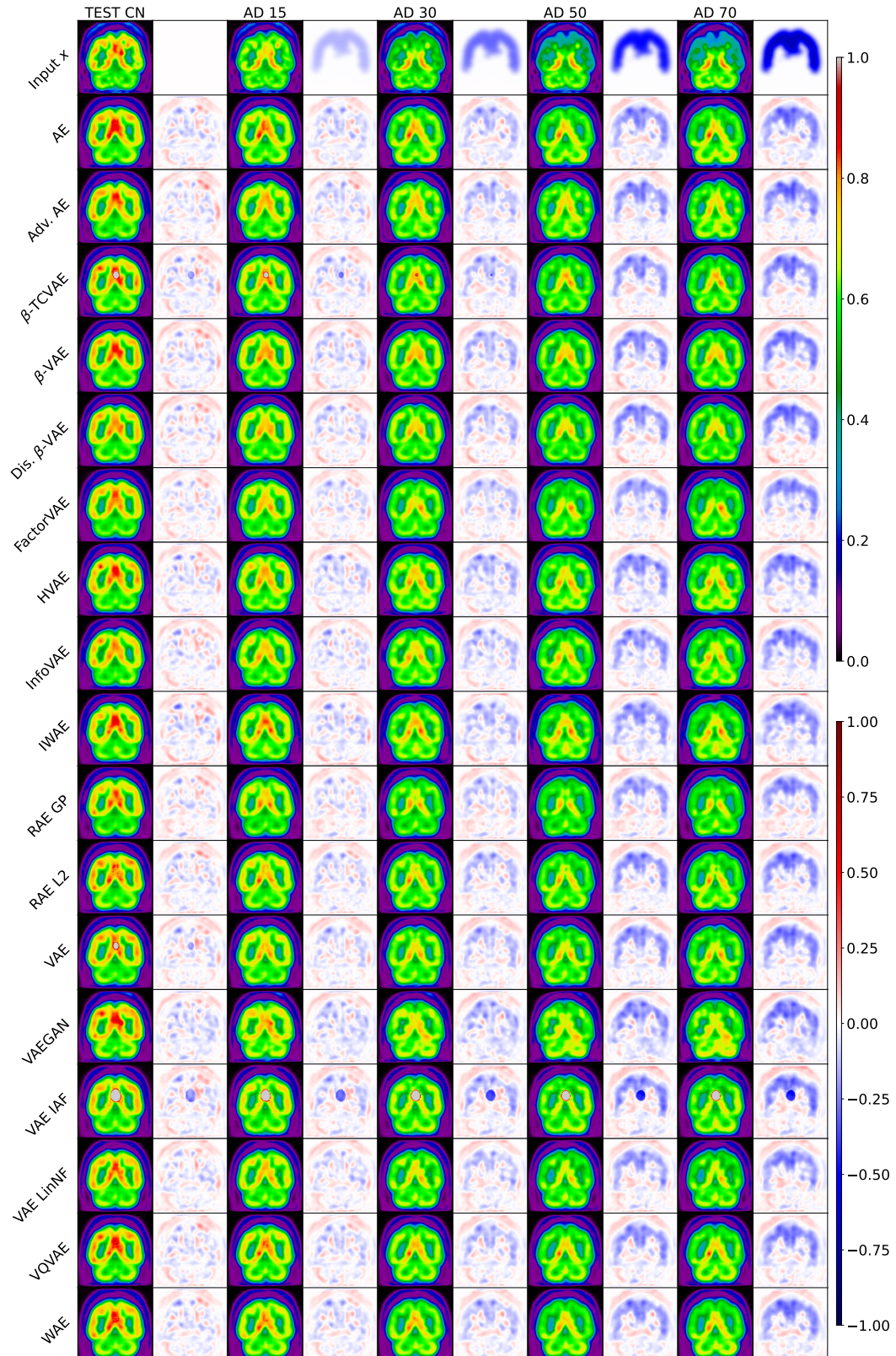


FIGURE G.1: Examples of reconstructions (coronal slices) obtained with the different VAE variants from the original image of a cognitively normal subject (images of the first column, Test CN) and from the same subject with AD simulated at different intensity degrees (AD 15, AD 30, AD 50 and AD 70). The first row shows the input image in odd columns and the mask of the simulated disease in even columns when the input is a simulated image. All the other rows are the pseudo-healthy reconstructions of the models in odd columns and the difference between the pseudo-healthy reconstruction and the input in even columns.

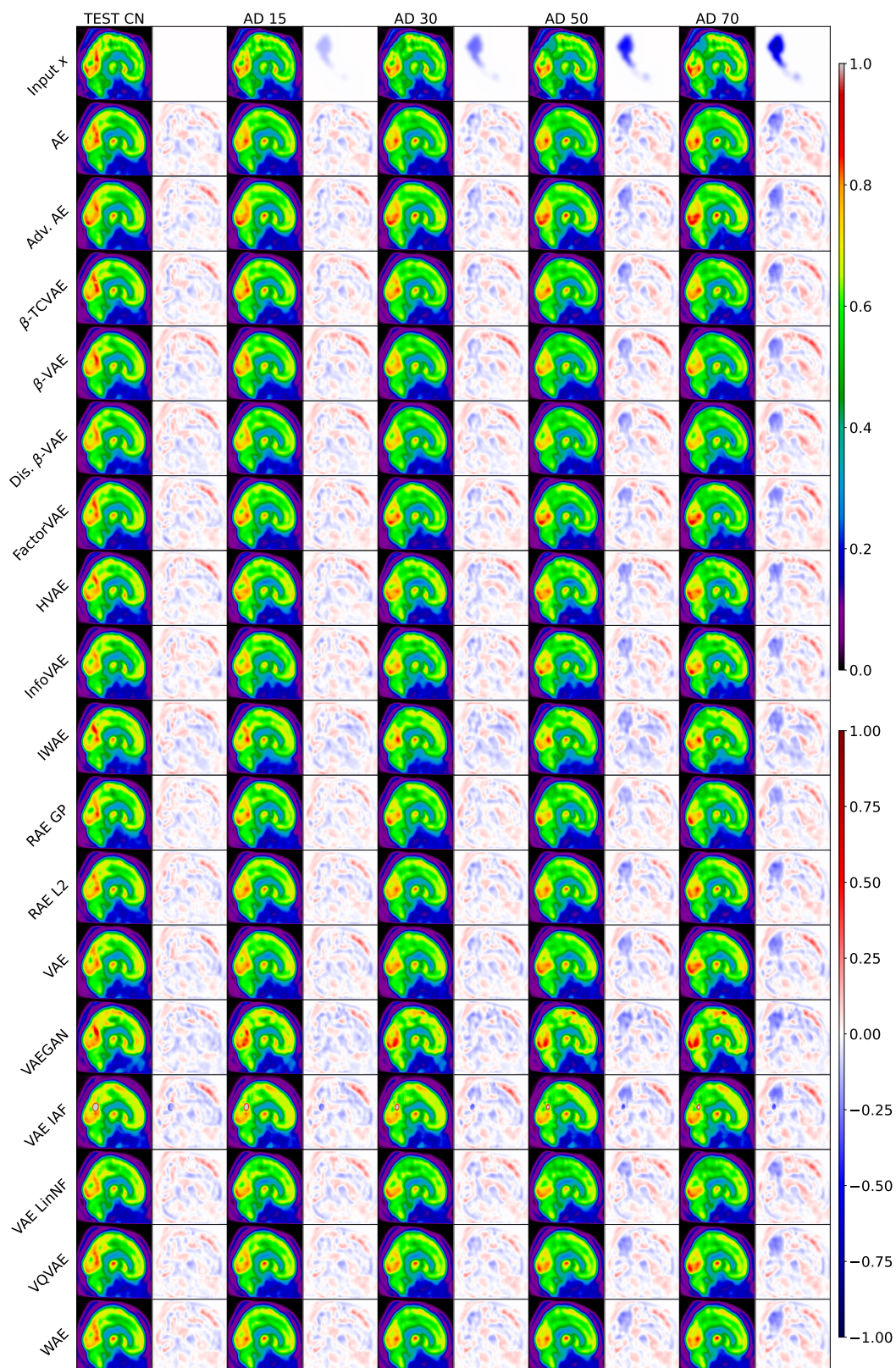


FIGURE G.2: Examples of reconstructions (sagittal slices) obtained with the different VAE variants from the original image of a cognitively normal subject (images of the first column, Test CN) and from the same subject with AD simulated at different intensity degrees (AD 15, AD 30, AD 50 and AD 70). The first row shows the input image in odd columns and the mask of the simulated disease in even columns when the input is a simulated image. All the other rows are the pseudo-healthy reconstructions of the models in odd columns and the difference between the pseudo-healthy reconstruction and the input in even columns.

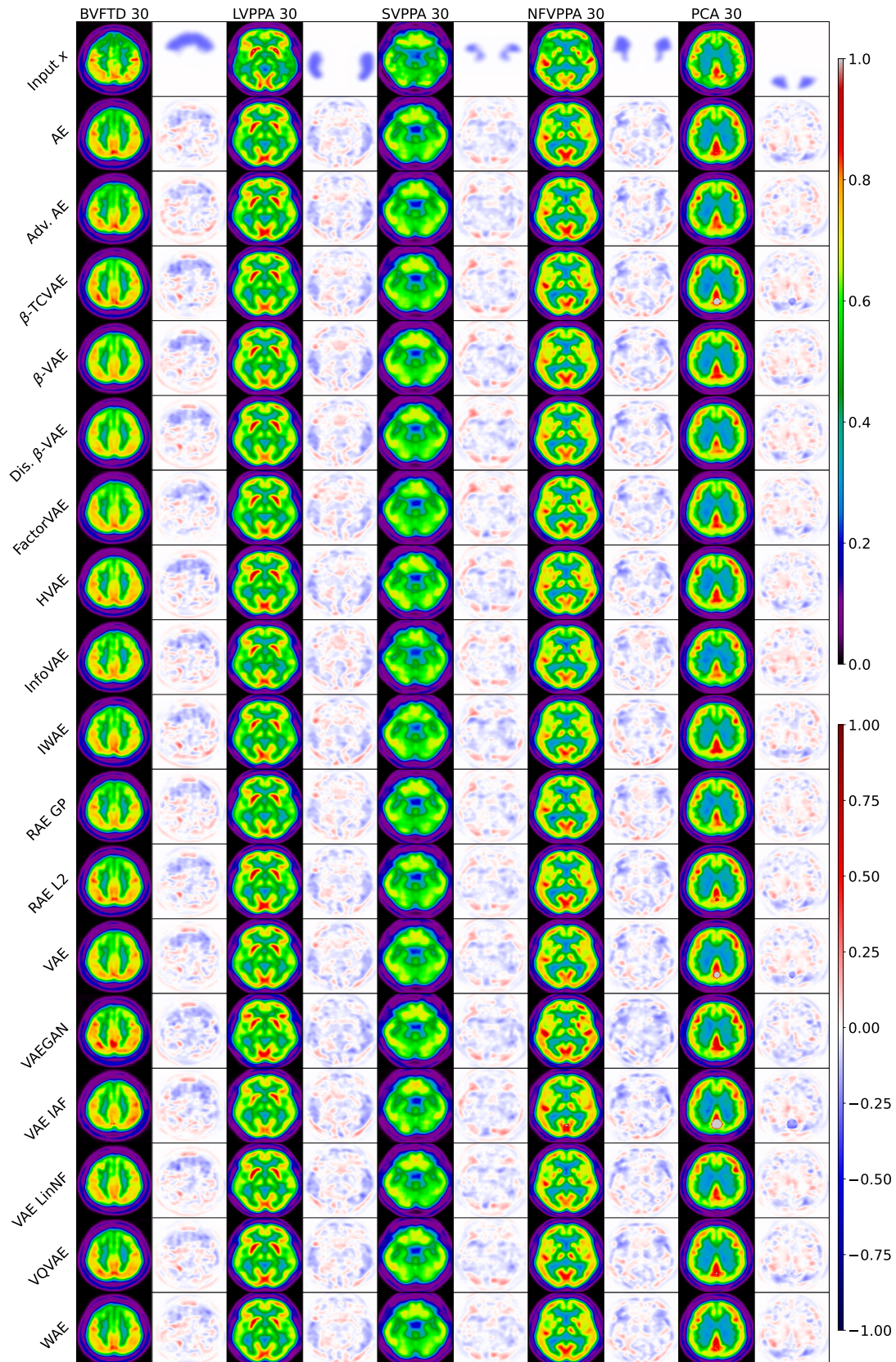


FIGURE G.3: Examples of reconstructions (axial slices) obtained with the different VAE variants from the same subject with different dementia subtypes simulated at 30% intensity degree (bvFTD 30, lvPPA 30, svPPA 30, nfvPPA 30 and PCA 30). The first row shows the input image in odd columns and the mask of the simulated disease in even columns. All the other rows are the pseudo-healthy reconstructions of the models in odd columns and the difference between the pseudo-healthy reconstruction and the input in even columns.

Appendix H

Example of MAPS

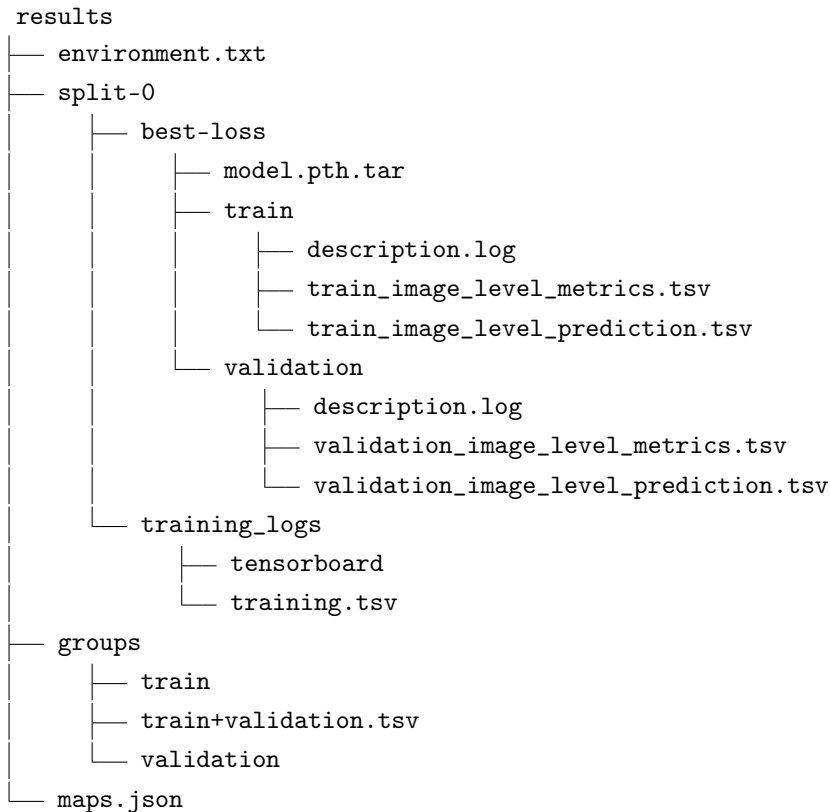


TABLE H.1: Example of the Model Analysis and Processing Structure (MAPS) obtained when training a classification network on whole images. Folders are in bold. Only the first split was trained (folder **split-0**) and one model was selected based on its validation loss (folder **best-loss**).

The only data groups are **train** and **validation**, which are automatically created during training. The characteristics of these groups are defined in **groups**, whereas the folder in **split-0/best-loss** contains the results for each input image (file ***_prediction.tsv**) and a set of metrics (file ***_metrics.tsv**) for each data group.

Finally, training logs are available for each split training in the folder **training_logs**.

These logs are available in two different formats, Tensorboard compatible and TSV.

As the training procedure ended without raising an error, the checkpoints were erased (this allows saving storage space).

Appendix I

Data access

ADNI

Data collection and sharing for this work was funded by the Alzheimer's Disease Neuroimaging Initiative (ADNI) (National Institutes of Health Grant U01 AG024904) and DOD ADNI (Department of Defense award number W81XWH-12-2-0012). ADNI is funded by the National Institute on Aging, the National Institute of Biomedical Imaging and Bioengineering, and through generous contributions from the following: AbbVie, Alzheimer's Association; Alzheimer's Drug Discovery Foundation; Araclon Biotech; BioClinica, Inc.; Biogen; Bristol-Myers Squibb Company; CereSpir, Inc.; Cogstate; Eisai Inc.; Elan Pharmaceuticals, Inc.; Eli Lilly and Company; EuroImmun; F. Hoffmann-La Roche Ltd and its affiliated company Genentech, Inc.; Fujirebio; GE Healthcare; IXICO Ltd.; Janssen Alzheimer Immunotherapy Research & Development, LLC.; Johnson & Johnson Pharmaceutical Research & Development LLC.; Lumosity; Lundbeck; Merck & Co., Inc.; Meso Scale Diagnostics, LLC.; NeuroRx Research; Neurotrack Technologies; Novartis Pharmaceuticals Corporation; Pfizer Inc.; Piramal Imaging; Servier; Takeda Pharmaceutical Company; and Transition Therapeutics. The Canadian Institutes of Health Research is providing funds to support ADNI clinical sites in Canada. Private sector contributions are facilitated by the Foundation for the National Institutes of Health (www.fnih.org). The grantee organization is the Northern California Institute for Research and Education, and the study is coordinated by the Alzheimer's Therapeutic Research Institute at the University of Southern California. ADNI data are disseminated by the Laboratory for Neuro Imaging at the University of Southern California.

Bibliography

- Alaverdyan, Z., J. Jung, R. Bouet, and C. Lartizien (2020). “Regularized siamese neural network for unsupervised outlier detection on brain multiparametric magnetic resonance imaging: application to epilepsy lesion screening”. In: *Medical image analysis* 60, p. 101618.
- American Psychiatric Association (2013). *Diagnostic and statistical manual of mental disorders (5th ed.)n*. Washington, DC.
- Astaraki, M., F. De Benetti, Y. Yeganeh, I. Toma-Dasu, Ö. Smedby, C. Wang, N. Navab, and T. Wendler (2023). “AutoPaint: A Self-Inpainting Method for Unsupervised Anomaly Detection”. In: *arXiv preprint arXiv:2305.12358*.
- Avants, B. B., C. L. Epstein, M. Grossman, and J. C. Gee (2008). “Symmetric diffeomorphic image registration with cross-correlation: Evaluating automated labeling of elderly and neurodegenerative brain”. In: *Medical Image Analysis*. Special Issue on The Third International Workshop on Biomedical Image Registration – WBIR 2006 12.1, pp. 26–41. DOI: [10.1016/j.media.2007.06.004](https://doi.org/10.1016/j.media.2007.06.004).
- Avants, B. B., N. J. Tustison, M. Stauffer, G. Song, B. Wu, and J. C. Gee (2014). “The Insight ToolKit image registration framework”. In: *Frontiers in Neuroinformatics* 8.
- Baur, C., S. Denner, B. Wiestler, N. Navab, and S. Albarqouni (2021a). “Autoencoders for unsupervised anomaly segmentation in brain MR images: A comparative study”. In: *Medical Image Analysis* 69, p. 101952. DOI: [10.1016/j.media.2020.101952](https://doi.org/10.1016/j.media.2020.101952).
- Baur, C., B. Wiestler, S. Albarqouni, and N. Navab (2019). “Deep autoencoding models for unsupervised anomaly segmentation in brain MR images”. In: *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries: 4th International Workshop, BrainLes 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 16, 2018, Revised Selected Papers, Part I 4*. Springer, pp. 161–169.
- Baur, C., B. Wiestler, M. Muehlau, C. Zimmer, N. Navab, and S. Albarqouni (2021b). “Modeling healthy anatomy with artificial intelligence for unsupervised anomaly detection in brain MRI”. In: *Radiology: Artificial Intelligence* 3.3, e190169.
- Baydargil, H. B., J.-S. Park, and D.-Y. Kang (2021). “Anomaly Analysis of Alzheimer’s Disease in PET Images Using an Unsupervised Adversarial Deep Learning Model”. In: *Applied Sciences* 11.5, p. 2187. DOI: [10.3390/app11052187](https://doi.org/10.3390/app11052187).
- Bengs, M., F. Behrendt, J. Krüger, R. Opfer, and A. Schlaefer (2021). “Three-dimensional deep learning with spatial erasing for unsupervised anomaly segmentation in brain MRI”. In: *International journal of computer assisted radiology and surgery* 16, pp. 1413–1423.
- Bengs, M., F. Behrendt, M.-H. Laves, J. Krüger, R. Opfer, and A. Schlaefer (2022). “Unsupervised anomaly detection in 3D brain MRI using deep learning with multi-task brain

- age prediction”. In: *Medical Imaging 2022: Computer-Aided Diagnosis*. Vol. 12033. SPIE, pp. 291–295.
- Bercea, C., B. Wiestler, D. Rueckert, and J. Schnabel (2023a). “Evaluating Normative Learning in Generative AI for Robust Medical Anomaly Detection”. In.
- Bercea, C. I., M. Neumayr, D. Rueckert, and J. A. Schnabel (2023b). “Mask, Stitch, and Re-Sample: Enhancing Robustness and Generalizability in Anomaly Detection through Automatic Diffusion Models”. In: *ICML 3rd Workshop on Interpretable Machine Learning in Healthcare (IMLH)*.
- Bercea, C. I., B. Wiestler, D. Rueckert, and J. A. Schnabel (2023c). “Generalizing Unsupervised Anomaly Detection: Towards Unbiased Pathology Screening”. In: *Medical Imaging with Deep Learning*.
- (2023d). “Reversing the abnormal: Pseudo-healthy generative networks for anomaly detection”. In: *International Conference on Medical Image Computing and Computer Assisted Intervention*. Springer Nature Switzerland, pp. 293–303.
- Biewald, L. et al. (2020). “Experiment tracking with weights and biases”. In: *Software available from wandb.com* 2, p. 233.
- Bouthillier, X., P. Delaunay, M. Bronzi, A. Trofimov, B. Nichyporuk, J. Szeto, N. Mohammadi Sepahvand, E. Raff, K. Madan, V. Voleti, et al. (2021). “Accounting for variance in machine learning benchmarks”. In: *Proceedings of Machine Learning and Systems* 3, pp. 747–769.
- Burda, Y., R. B. Grosse, and R. Salakhutdinov (2016). “Importance Weighted Autoencoders”. In: *ICLR*.
- Burgess, C. P., I. Higgins, A. Pal, L. Matthey, N. Watters, G. Desjardins, and A. Lerchner (2018). “Understanding disentangling in β -VAE”. In: *arXiv preprint arXiv:1804.03599*.
- Burgos, N. (2023). “Neuroimaging in Machine Learning for Brain Disorders”. In: *Machine Learning for Brain Disorders*. Springer, pp. 253–284.
- Burgos, N., S. Bottani, J. Faouzi, E. Thibeau-Sutre, and O. Colliot (2021a). “Deep learning for brain disorders: from data processing to disease treatment”. In: *Briefings in Bioinformatics* 22.2, pp. 1560–1576.
- Burgos, N., M. J. Cardoso, A. F. Mendelson, J. M. Schott, D. Atkinson, S. R. Arridge, B. F. Hutton, and S. Ourselin (2015). “Subject-Specific Models for the Analysis of Pathological FDG PET Data”. In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*. Lecture Notes in Computer Science. Springer, pp. 651–658. DOI: [10.1007/978-3-319-24571-3_78](https://doi.org/10.1007/978-3-319-24571-3_78).
- Burgos, N., M. J. Cardoso, J. Samper-González, M.-O. Habert, S. Durrleman, S. Ourselin, O. Colliot, A. D. N. Initiative, F. L. D. N. Initiative, et al. (2021b). “Anomaly detection for the individual analysis of brain PET images”. In: *Journal of Medical Imaging* 8.2, p. 024003.
- Burgos, N. and O. Colliot (2020). “Machine learning for classification and prediction of brain diseases: recent advances and upcoming challenges”. In: *Current Opinion in Neurology* 33.4, pp. 439–450.
- Burgos, N., J. Samper-González, A. Bertrand, M.-O. Habert, S. Ourselin, S. Durrleman, M. J. Cardoso, and O. Colliot (2017). “Individual Analysis of Molecular Brain Imaging

- Data through Automatic Identification of Abnormality Patterns”. In: *Molecular Imaging, Reconstruction and Analysis of Moving Body Organs, and Stroke Imaging and Treatment*. Lecture Notes in Computer Science. Springer, pp. 13–22. DOI: [10.1007/978-3-319-67564-0_2](https://doi.org/10.1007/978-3-319-67564-0_2).
- Caterini, A. L., A. Doucet, and D. Sejdinovic (2018). “Hamiltonian variational auto-encoder”. In: *Advances in NeurIPS* 31.
- Chadebec, C., L. J. Vincent, and S. Allasonniere (2022). “Pythae: Unifying Generative Autoencoders in Python - A Benchmarking Use Case”. In: *Thirty-sixth Conference on NeurIPS Datasets and Benchmarks Track*.
- Chatterjee, S., A. Sciarra, M. Dünnwald, P. Tummala, S. K. Agrawal, A. Jauhari, A. Kalra, S. Oeltze-Jafra, O. Speck, and A. Nürnberger (2022). “StRegA: Unsupervised anomaly detection in brain MRIs using a compact context-encoding variational autoencoder”. In: *Computers in Biology and Medicine* 149, p. 106093.
- Chen, R. T., X. Li, R. B. Grosse, and D. K. Duvenaud (2018a). “Isolating sources of disentanglement in variational autoencoders”. In: *Advances in NeurIPS* 31.
- Chen, X. and E. Konukoglu (2018b). “Unsupervised Detection of Lesions in Brain MRI using constrained adversarial auto-encoders”. In: *MIDL*.
- (2022). “Unsupervised abnormality detection in medical images with deep generative methods”. In: *Biomedical Image Synthesis and Simulation*. Ed. by [Burgos, N.](#) and [Svoboda, D.](#) Elsevier, pp. 303–324.
- Chen, X., S. You, K. C. Tezcan, and E. Konukoglu (2020). “Unsupervised lesion detection via image restoration with a normative prior”. In: *Medical image analysis* 64, p. 101713.
- Chételat, G., J. Arbizu, H. Barthel, V. Garibotto, I. Law, S. Morbelli, E. van de Giessen, F. Agosta, F. Barkhof, D. J. Brooks, et al. (2020). “Amyloid-PET and 18F-FDG-PET in the diagnostic investigation of Alzheimer’s disease and other dementias”. In: *The Lancet Neurology* 19.11, pp. 951–962.
- Choi, H., S. Ha, H. Kang, H. Lee, D. S. Lee, and Alzheimer’s Disease Neuroimaging Initiative (2019). “Deep learning only by normal brain PET identify unheralded brain anomalies”. In: *EBioMedicine* 43, pp. 447–453. DOI: [10.1016/j.ebiom.2019.04.022](https://doi.org/10.1016/j.ebiom.2019.04.022).
- Colliot, O., E. Thibeau-Sutre, C. Brianceau, and N. Burgos (2024). “Reproducibility in medical image computing: What is it and how is it assessed?”
- Colliot, O., E. Thibeau-Sutre, and N. Burgos (2023). “Reproducibility in Machine Learning for Medical Imaging”. In: *Machine Learning for Brain Disorders*. Ed. by O. Colliot. Neuromethods. New York, NY: Springer US, pp. 631–653. DOI: [10.1007/978-1-0716-3195-9_21](https://doi.org/10.1007/978-1-0716-3195-9_21).
- Davidson, T. R., L. Falorsi, N. De Cao, T. Kipf, and J. M. Tomczak (2018). “Hyperspherical variational auto-encoders”. In: *arXiv:1804.00891*.
- De Carli, F., F. Nobili, M. Pagani, M. Bauckneht, F. Massa, M. Grazzini, C. Jonsson, E. Peira, S. Morbelli, D. Arnaldi, and f. t. A. D. N. Initiative (2019). “Accuracy and Generalization Capability of an Automatic Method for the Detection of Typical Brain Hypometabolism in Prodromal Alzheimer Disease”. In: *European Journal of Nuclear Medicine and Molecular Imaging* 46.2, pp. 334–347. DOI: [10.1007/s00259-018-4197-7](https://doi.org/10.1007/s00259-018-4197-7).

- Dhariwal, P. and A. Nichol (2021). “Diffusion models beat gans on image synthesis”. In: *Advances in neural information processing systems* 34, pp. 8780–8794.
- Dorent, R., N. Haouchine, F. Kogl, S. Joutard, P. Juvekar, E. Torio, A. J. Golby, S. Ourselin, S. Frisken, T. Vercauteren, et al. (2023). “Unified brain MR-ultrasound synthesis using multi-modal hierarchical representations”. In: *International conference on medical image computing and computer-assisted intervention*. Springer, pp. 448–458.
- Duquenne, P.-A., H. Gong, B. Sagot, and H. Schwenk (Dec. 2022). “T-Modules: Translation Modules for Zero-Shot Cross-Modal Machine Translation”. In: *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, pp. 5794–5806.
- Ebrahimighahnavieh, M. A., S. Luo, and R. Chiong (2020). “Deep learning to detect Alzheimer’s disease from neuroimaging: A systematic literature review”. In: *Computer methods and programs in biomedicine* 187, p. 105242.
- Epelbaum, S. and F. Cacciamani (2023). “Clinical Assessment of Brain Disorders”. In: *Machine Learning for Brain Disorders*. Springer, pp. 233–252.
- Esmaeili, M., A. Toosi, A. Roshanpoor, V. Changizi, M. Ghazisaeedi, A. Rahmim, and M. Sabokrou (2023). “Generative Adversarial Networks for Anomaly Detection in Biomedical Imaging: A Study on Seven Medical Image Datasets”. In: *IEEE Access* 11, pp. 17906–17921.
- Esteva, A., B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, H. M. Blau, and S. Thrun (2017). “Dermatologist-level classification of skin cancer with deep neural networks”. In: *nature* 542.7639, pp. 115–118.
- Ezhov, I., K. Scibilia, K. Franitza, F. Steinbauer, S. Shit, L. Zimmer, J. Lipkova, F. Kofler, J. C. Paetzold, L. Canalini, et al. (2023). “Learn-Morph-Infer: a new way of solving the inverse problem for brain tumor modeling”. In: *Medical Image Analysis* 83, p. 102672.
- Fernando, T., H. Gammulle, S. Denman, S. Sridharan, and C. Fookes (2022). “Deep Learning for Medical Anomaly Detection A Survey”. In: *ACM Computing Surveys* 54.7. DOI: [10.1145/3464423](https://doi.org/10.1145/3464423).
- Fernando, T., H. Gammulle, S. Denman, S. Sridharan, and C. Fookes (2021). “Deep Learning for Medical Anomaly Detection – A Survey”. In: *ACM Computing Surveys* 54.7.
- Fischl, B. (2012). “FreeSurfer”. In: *Neuroimage* 62.2, pp. 774–781.
- Folstein, M. F., S. E. Folstein, and P. R. McHugh (1975). ““Mini-mental state”: a practical method for grading the cognitive state of patients for the clinician”. In: *Journal of psychiatric research* 12.3, pp. 189–198.
- Fonov, V., A. C. Evans, K. Botteron, C. R. Almli, R. C. McKinstry, and D. L. Collins (2011). “Unbiased average age-appropriate atlases for pediatric studies”. In: *NeuroImage* 54.1, pp. 313–327. DOI: [10.1016/j.neuroimage.2010.07.033](https://doi.org/10.1016/j.neuroimage.2010.07.033).
- Fonov, V. S., M. Dadar, T. P.-A. R. G. Adni, and D. L. Collins (2022). “DARQ: Deep learning of quality control for stereotaxic registration of human brain MRI to the T1w MNI-ICBM 152 template”. In: *NeuroImage* 257, p. 119266.

- Fonov, V., A. Evans, R. McKinstry, C. Almlil, and D. Collins (2009). “Unbiased nonlinear average age-appropriate brain templates from birth to adulthood”. In: *NeuroImage*. Organization for Human Brain Mapping 2009 Annual Meeting 47, S102. DOI: [10.1016/S1053-8119\(09\)70884-5](https://doi.org/10.1016/S1053-8119(09)70884-5).
- Friston, K. J. (2003). “Statistical parametric mapping”. In: *Neuroscience databases: a practical guide*, pp. 237–250.
- Germain, M., K. Gregor, I. Murray, and H. Larochelle (2015). “Made: Masked autoencoder for distribution estimation”. In: *International conference on machine learning (ICML)*. PMLR, pp. 881–889.
- Ghosh, P., M. S. Sajjadi, A. Vergari, M. Black, and B. Schölkopf (2019). “From variational to deterministic autoencoders”. In: *arXiv:1903.12436*.
- Gong, C., C. Jing, X. Chen, C. M. Pun, G. Huang, A. Saha, M. Nieuwoudt, H.-X. Li, Y. Hu, and S. Wang (2023). “Generative AI for brain image computing and brain network computing: a review”. In: *Frontiers in Neuroscience* 17, p. 1203104.
- Goodfellow, I., J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio (2014). “Generative Adversarial Nets”. In: *Advances in Neural Information Processing Systems*. Vol. 27.
- Gorgolewski, K., C. Burns, C. Madison, D. Clark, Y. Halchenko, M. Waskom, and S. Ghosh (2011). “Nipype: A Flexible, Lightweight and Extensible Neuroimaging Data Processing Framework in Python”. In: *Frontiers in Neuroinformatics* 5. DOI: [10.3389/fninf.2011.00013](https://doi.org/10.3389/fninf.2011.00013).
- Gorgolewski, K. J., F. Alfaro-Almagro, T. Auer, P. Bellec, M. Capotà, M. M. Chakravarty, N. W. Churchill, A. L. Cohen, R. C. Craddock, G. A. Devenyi, A. Eklund, O. Esteban, G. Flandin, S. S. Ghosh, J. S. Guntupalli, M. Jenkinson, A. Keshavan, G. Kiar, F. Liem, P. R. Raamana, D. Raffelt, C. J. Steele, P.-O. Quirion, R. E. Smith, S. C. Strother, G. Varoquaux, Y. Wang, T. Yarkoni, and R. A. Poldrack (2017). “BIDS Apps: Improving Ease of Use, Accessibility, and Reproducibility of Neuroimaging Data Analysis Methods”. In: *PLOS Computational Biology* 13.3, e1005209. DOI: [10.1371/journal.pcbi.1005209](https://doi.org/10.1371/journal.pcbi.1005209).
- Gorgolewski, K. J., T. Auer, V. D. Calhoun, R. C. Craddock, S. Das, E. P. Duff, G. Flandin, S. S. Ghosh, T. Glatard, Y. O. Halchenko, et al. (2016). “The brain imaging data structure, a format for organizing and describing outputs of neuroimaging experiments”. In: *Scientific data* 3.1, pp. 1–9.
- Graham, M. S., W. H. L. Pinaya, P. Wright, P.-D. Tudosiu, Y. H. Mah, J. T. Teo, H. R. Jäger, D. Werring, P. Nachev, S. Ourselin, et al. (2023). “Unsupervised 3D out-of-distribution detection with latent diffusion models”. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*. Springer, pp. 446–456.
- Hampel, H., J. Cummings, K. Blennow, P. Gao, C. R. Jack Jr, and A. Vergallo (2021). “Developing the ATX (N) classification for use across the Alzheimer disease continuum”. In: *Nature Reviews Neurology* 17.9, pp. 580–589.
- Han, C., L. Rundo, K. Murao, T. Noguchi, Y. Shimahara, Z. Á. Milacski, S. Koshino, E. Sala, H. Nakayama, and S. Satoh (2021). “MADGAN: Unsupervised medical anomaly detection GAN using multiple adjacent brain MRI slice reconstruction”. In: *BMC bioinformatics* 22.2, pp. 1–20.

- Hardy, J. A. and G. A. Higgins (1992). “Alzheimer’s disease: the amyloid cascade hypothesis”. In: *Science* 256.5054, pp. 184–185.
- Harkness, R., A. F. Frangi, K. Zucker, and N. Ravikumar (2023). “Learning disentangled representations for explainable chest x-ray classification using Dirichlet VAEs”. In: *Medical Imaging 2023: Image Processing*. Vol. 12464. SPIE, p. 1246411. DOI: [10.1117/12.2654345](https://doi.org/10.1117/12.2654345).
- Hassanally, R., S. Bottani, B. Sauty, O. Colliot, and N. Burgos (2023a). “Simulation based evaluation framework for deep learning unsupervised anomaly detection on brain FDG-PET”. In: *Medical Imaging 2023: Image Processing*. Vol. 12464. SPIE, pp. 511–518.
- Hassanally, R., C. Brianceau, O. Colliot, and N. Burgos (2023b). “Unsupervised anomaly detection in 3D brain FDG PET: A benchmark of 17 VAE-based approaches”. In: *Deep Generative Models workshop at the 26th International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI 2023)*. Vancouver, Canada.
- Hassanally, R., C. Brianceau, M. Diaz, S. Loizillon, E. Thibeau-Sutre, N. Cassereau, O. Colliot, and N. Burgos (2024a). “Recent advances in the open-source ClinicaDL software for reproducible neuroimaging with deep learning”. In: *SPIE Medical Imaging*. San Diego, United States.
- Hassanally, R., C. Brianceau, M. Solal, O. Colliot, and N. Burgos (2024b). “Evaluation of pseudo-healthy image reconstruction for anomaly detection with deep generative models: Application to Brain FDG PET”. In: *Machine Learning for Biomedical Imaging 2 (Special Issue for Generative Models)*, pp. 611–656. DOI: [10.59275/j.melba.2024-b87a](https://doi.org/10.59275/j.melba.2024-b87a).
- Hassanally, R., M. Solal, O. Colliot, and N. Burgos (2024c). “Pseudo-healthy image reconstruction with variational autoencoders for anomaly detection: A benchmark on 3D brain FDG PET”.
- He, K., X. Zhang, S. Ren, and J. Sun (2016). “Deep residual learning for image recognition”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778.
- Herholz, K, S. F. Carter, and M Jones (2007). “Positron emission tomography imaging in dementia”. In: *The British Journal of Radiology* 80 (special_issue_2), S160–S167. DOI: [10.1259/bjrr/97295129](https://doi.org/10.1259/bjrr/97295129).
- Herholz, K. (1995). “FDG PET and differential diagnosis of dementia”. In: *Alzheimer Disease and Associated Disorders* 9.1, pp. 6–16. DOI: [10.1097/00002093-199505000-00004](https://doi.org/10.1097/00002093-199505000-00004).
- Higgins, I., L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner (2017). “beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework”. In: *ICLR*.
- Hinge, C., O. M. Henriksen, U. Lindberg, S. G. Hasselbalch, L. Højgaard, I. Law, F. L. Andersen, and C. N. Ladefoged (2022). “A Zero-Dose Synthetic Baseline for the Personalized Analysis of [18F]FDG-PET: Application in Alzheimer’s Disease”. In: *Frontiers in Neuroscience* 16.
- Ho, J., A. Jain, and P. Abbeel (2020). “Denoising diffusion probabilistic models”. In: *Advances in NeurIPS* 33, pp. 6840–6851.

- Hughes, C. P., L. Berg, W. Danziger, L. A. Coben, and R. L. Martin (1982). “A new clinical scale for the staging of dementia”. In: *The British journal of psychiatry* 140.6, pp. 566–572.
- Isensee, F., P. F. Jaeger, S. A. Kohl, J. Petersen, and K. H. Maier-Hein (2021). “nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation”. In: *Nature methods* 18.2, pp. 203–211.
- Jack, C. R., D. A. Bennett, K. Blennow, M. C. Carrillo, H. H. Feldman, G. B. Frisoni, H. Hampel, W. J. Jagust, K. A. Johnson, D. S. Knopman, R. C. Petersen, P. Scheltens, R. A. Sperling, and B. Dubois (2016). “A/T/N: An Unbiased Descriptive Classification Scheme for Alzheimer Disease Biomarkers”. In: *Neurology* 87.5, pp. 539–547. DOI: [10.1212/WNL.0000000000002923](https://doi.org/10.1212/WNL.0000000000002923).
- Jagust, W. J., D. Bandy, K. Chen, N. L. Foster, S. M. Landau, C. A. Mathis, J. C. Price, E. M. Reiman, D. Skovronsky, and R. A. Koeppe (2010). “The Alzheimer’s Disease Neuroimaging Initiative Positron Emission Tomography Core”. In: *Alzheimer’s & Dementia* 6.3, pp. 221–229. DOI: [10.1016/j.jalz.2010.03.003](https://doi.org/10.1016/j.jalz.2010.03.003).
- Jagust, W. J., S. M. Landau, R. A. Koeppe, E. M. Reiman, K. Chen, C. A. Mathis, J. C. Price, N. L. Foster, and A. Y. Wang (2015). “The Alzheimer’s Disease Neuroimaging Initiative 2 PET Core: 2015”. In: *Alzheimer’s & Dementia* 11.7, pp. 757–771. DOI: [10.1016/j.jalz.2015.05.001](https://doi.org/10.1016/j.jalz.2015.05.001).
- Jarecka, D., M. Goncalves, C. J. Markiewicz, O. Esteban, N. Lo, J. Kaczmarzyk, and S. Ghosh (2020). “Pydra-a flexible and lightweight dataflow engine for scientific analyses”. In: *Proceedings of the 19th python in science conference*. Vol. 132, p. 139.
- Jenkinson, M., C. F. Beckmann, T. E. Behrens, M. W. Woolrich, and S. M. Smith (2012). “Fsl”. In: *Neuroimage* 62.2, pp. 782–790.
- Kaufman, S., S. Rosset, C. Perlich, and O. Stitelman (2012). “Leakage in data mining: Formulation, detection, and avoidance”. In: *ACM Transactions on Knowledge Discovery from Data* 6.4, 15:1–15:21. DOI: [10.1145/2382577.2382579](https://doi.org/10.1145/2382577.2382579).
- Khan, T. (2016). “Chapter 2-Clinical diagnosis of Alzheimer’s disease”. In: *Biomarkers in Alzheimer’s disease*, pp. 27–48.
- Khanal, B., N. Ayache, and X. Pennec (2017). “Simulating longitudinal brain MRIs with known volume changes and realistic variations in image intensity”. In: *Frontiers in neuroscience* 11, p. 132.
- Kim, H. and A. Mnih (2018). “Disentangling by factorising”. In: *ICML*. PMLR, pp. 2649–2658.
- Kim, M., K.-R. Moon, and B.-D. Lee (2023). “Unsupervised anomaly detection for posteroanterior chest X-rays using multiresolution patch-based self-supervised learning”. In: *Scientific Reports* 13.1, p. 3415.
- Kingma, D. P. and M. Welling (2014). *Auto-Encoding Variational Bayes*. arXiv: [1312.6114](https://arxiv.org/abs/1312.6114).
- Kingma, D. P., T. Salimans, R. Jozefowicz, X. Chen, I. Sutskever, and M. Welling (2016). “Improved variational inference with inverse autoregressive flow”. In: *Advances in NeurIPS* 29.

- Kumar, S., P. R. Payne, and A. Sotiras (2023). “Normative modeling using multimodal variational autoencoders to identify abnormal brain volume deviations in Alzheimer’s disease”. In: *Medical Imaging 2023: Computer-Aided Diagnosis*. Vol. 12465. SPIE, p. 1246503.
- Lagogiannis, I., F. Meissen, G. Kaissis, and D. Rueckert (2023). “Unsupervised Pathology Detection: A Deep Dive Into the State of the Art”. In: *arXiv preprint arXiv:2303.00609*.
- Landau, S. M., B. A. Thomas, L. Thurfjell, M. Schmidt, R. Margolin, M. Mintun, M. Pontecorvo, S. L. Baker, W. J. Jagust, and the Alzheimer’s Disease Neuroimaging Initiative (2014). “Amyloid PET imaging in Alzheimer’s disease: a comparison of three radiotracers”. In: *European Journal of Nuclear Medicine and Molecular Imaging* 41.7, pp. 1398–1407. DOI: [10.1007/s00259-014-2753-3](https://doi.org/10.1007/s00259-014-2753-3).
- Landau, S. M., M. A. Mintun, A. D. Joshi, R. A. Koeppe, R. C. Petersen, P. S. Aisen, M. W. Weiner, and W. J. Jagust (2012). “Amyloid Deposition, Hypometabolism, and Longitudinal Cognitive Decline”. In: *Annals of Neurology* 72.4, pp. 578–586. DOI: [10.1002/ana.23650](https://doi.org/10.1002/ana.23650).
- Larsen, A. B. L., S. K. Sønderby, H. Larochelle, and O. Winther (2016). “Autoencoding beyond pixels using a learned similarity metric”. In: *ICML*. PMLR, pp. 1558–1566.
- Lawry Aguila, A., J. Chapman, and A. Altmann (2023). “Multi-modal Variational Autoencoders for Normative Modelling Across Multiple Imaging Modalities”. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, pp. 425–434.
- Leuzy, A., K. Chiotis, L. Lemoine, P.-G. Gillberg, O. Almkvist, E. Rodriguez-Vieitez, and A. Nordberg (2019). “Tau PET imaging in neurodegenerative tauopathies—still a challenge”. In: *Molecular Psychiatry* 24.8, pp. 1112–1134. DOI: [10.1038/s41380-018-0342-8](https://doi.org/10.1038/s41380-018-0342-8).
- Li, X., P. S. Morgan, J. Ashburner, J. Smith, and C. Rorden (2016). “The First Step for Neuroimaging Data Analysis: DICOM to NIfTI Conversion”. In: *Journal of Neuroscience Methods* 264, pp. 47–56. DOI: [10.1016/j.jneumeth.2016.03.001](https://doi.org/10.1016/j.jneumeth.2016.03.001).
- Liew, S.-L., J. M. Anglin, N. W. Banks, M. Sondag, K. L. Ito, H. Kim, J. Chan, J. Ito, C. Jung, S. Lefebvre, et al. (2017). “The anatomical tracings of lesions after stroke (atlas) dataset-release 1.1”. In: *bioRxiv*, p. 179614.
- Litjens, G., T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghahfoorian, J. A. van der Laak, B. van Ginneken, and C. I. Sánchez (2017). “A survey on deep learning in medical image analysis”. In: *Medical Image Analysis* 42, pp. 60–88. DOI: <https://doi.org/10.1016/j.media.2017.07.005>.
- Loizillon, S., S. Bottani, A. Maire, S. Ströer, D. Dormont, O. Colliot, and N. Burgos (2023). “Transfer learning from synthetic to routine clinical data for motion artefact detection in brain T1-weighted MRI”. In: *SPIE Medical Imaging 2023: Image Processing*. DOI: [10.1117/12.2648201](https://doi.org/10.1117/12.2648201).
- Lu, Y. and P. Xu (2018). “Anomaly detection for skin disease images using variational autoencoder”. In: *arXiv preprint arXiv:1807.01349*.
- Luo, G., W. Xie, R. Gao, T. Zheng, L. Chen, and H. Sun (2023). “Unsupervised anomaly detection in brain MRI: Learning abstract distribution from massive healthy brains”. In: *Computers in Biology and Medicine* 154, p. 106610.

- Lüth, C. T., D. Zimmerer, G. Koehler, P. F. Jaeger, F. Isensee, J. Petersen, and K. H. Maier-Hein (2023). “CRADL: Contrastive Representations for Unsupervised Anomaly Detection and Localization”. In: *Bildverarbeitung für die Medizin 2023*.
- Maaløe, L., M. Fraccaro, V. Liévin, and O. Winther (2019). “Biva: A very deep hierarchy of latent variables for generative modeling”. In: *Advances in neural information processing systems* 32.
- Maier, O., B. H. Menze, J. Von der Gablentz, L. Häni, M. P. Heinrich, M. Liebrand, S. Winzeck, A. Basit, P. Bentley, L. Chen, et al. (2017). “ISLES 2015-A public evaluation benchmark for ischemic stroke lesion segmentation from multispectral MRF”. In: *Medical image analysis* 35, pp. 250–269.
- Makhzani, A., J. Shlens, N. Jaitly, I. Goodfellow, and B. Frey (2015). “Adversarial autoencoders”. In: *arXiv:1511.05644*.
- Manzanera, O. E. M., S. Ellis, V. Baltatzis, A. Nair, L. Le Folgoc, S. Desai, B. Glocker, and J. A. Schnabel (2021). “Patient-specific 3D cellular automata nodule growth synthesis in lung cancer without the need of external data”. In: *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)*. IEEE, pp. 5–9.
- Marimont, S. N. and G. Tarroni (2021). “Anomaly detection through latent space restoration using vector quantized variational autoencoders”. In: *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)*. IEEE, pp. 1764–1767.
- McKhann, G., D. Drachman, M. Folstein, R. Katzman, D. Price, and E. M. Stadlan (1984). “Clinical diagnosis of Alzheimer’s disease: Report of the NINCDS-ADRDA Work Group* under the auspices of Department of Health and Human Services Task Force on Alzheimer’s Disease”. In: *Neurology* 34.7, pp. 939–939.
- Meissen, F., B. Wiestler, G. Kaissis, and D. Rueckert (2021). “On the Pitfalls of Using the Residual as Anomaly Score”. In: *Medical Imaging with Deep Learning*.
- Menze, B. H., A. Jakab, S. Bauer, J. Kalpathy-Cramer, K. Farahani, J. Kirby, Y. Burren, N. Porz, J. Slotboom, R. Wiest, et al. (2014). “The multimodal brain tumor image segmentation benchmark (BRATS)”. In: *IEEE transactions on medical imaging* 34.10, pp. 1993–2024.
- Mostapha, M., J. Prieto, V. Murphy, J. Girault, M. Foster, A. Rumble, J. Blocher, W. Lin, J. Elison, J. Gilmore, et al. (2019). “Semi-supervised VAE-GAN for out-of-sample detection applied to MRI quality control”. In: *MICCAI*. Springer, pp. 127–136.
- Mueller, S. G., M. W. Weiner, L. J. Thal, R. C. Petersen, C. Jack, W. Jagust, J. Q. Trojanowski, A. W. Toga, and L. Beckett (2005). “The Alzheimer’s Disease Neuroimaging Initiative”. In: *Neuroimaging Clinics of North America*. Alzheimer’s Disease: 100 Years of Progress 15.4, pp. 869–877. DOI: [10.1016/j.nic.2005.09.008](https://doi.org/10.1016/j.nic.2005.09.008).
- Nakao, T., S. Hanaoka, Y. Nomura, M. Murata, T. Takenaga, S. Miki, T. Watadani, T. Yoshikawa, N. Hayashi, and O. Abe (2021). “Unsupervised deep anomaly detection in chest radiographs”. In: *Journal of Digital Imaging* 34, pp. 418–427.
- Neal, R. M. (2005). “Hamiltonian importance sampling”. In: *talk presented at the Banff International Research Station (BIRS) workshop on Mathematical Issues in Molecular Dynamics*.

- Nečasová, T., N. Burgos, and D. Svoboda (2022). “Validation and Evaluation Metrics for Medical and Biomedical Image Synthesis”. In: *Biomedical Image Synthesis and Simulation*. Ed. by N. Burgos and D. Svoboda. The MICCAI Society Book Series. Academic Press, pp. 573–600.
- Nichols, E., J. D. Steinmetz, S. E. Vollset, K. Fukutaki, J. Chalek, F. Abd-Allah, A. Abdoli, A. Abualhasan, E. Abu-Gharbieh, T. T. Akram, et al. (2022). “Estimation of the global prevalence of dementia in 2019 and forecasted prevalence in 2050: an analysis for the Global Burden of Disease Study 2019”. In: *The Lancet Public Health* 7.2, e105–e125.
- Nordberg, A., J. O. Rinne, A. Kadir, and B. Långström (2010). “The use of PET in Alzheimer disease”. In: *Nature Reviews Neurology* 6.2, pp. 78–87. DOI: [10.1038/nrneuro.2009.217](https://doi.org/10.1038/nrneuro.2009.217).
- Nugent, S., E. Croteau, O. Potvin, C.-A. Castellano, L. Dieumegarde, S. C. Cunnane, and S. Duchesne (2020). “Selection of the optimal intensity normalization region for FDG-PET studies of normal aging and Alzheimer’s disease”. In: *Scientific Reports* 10.1, p. 9261. DOI: [10.1038/s41598-020-65957-3](https://doi.org/10.1038/s41598-020-65957-3).
- Pandey, K., A. Mukherjee, P. Rai, and A. Kumar (2022). “DiffuseVAE: Efficient, Controllable and High-Fidelity Generation from Low-Dimensional Latents”. In: *Transactions on Machine Learning Research*.
- Park, S., K. H. Lee, B. Ko, and N. Kim (2023). “Unsupervised anomaly detection with generative adversarial networks in mammography”. In: *Scientific Reports* 13.1, p. 2925.
- Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al. (2011). “Scikit-learn: Machine learning in Python”. In: *the Journal of machine Learning research* 12, pp. 2825–2830.
- Pellegrini, E., L. Ballerini, M. d. C. V. Hernandez, F. M. Chappell, V. González-Castro, D. Anblagan, S. Danso, S. Muñoz-Maniega, D. Job, C. Pernet, et al. (2018). “Machine learning of neuroimaging for assisted diagnosis of cognitive impairment and dementia: a systematic review”. In: *Alzheimer’s & Dementia: Diagnosis, Assessment & Disease Monitoring* 10, pp. 519–535.
- Perani, D., P. A. Della Rosa, C. Cerami, F. Gallivanone, F. Fallanca, E. G. Vanoli, A. Panzavochi, F. Nobili, S. Pappatà, A. Marcone, et al. (2014). “Validation of an optimized SPM procedure for FDG-PET in dementia diagnosis in a clinical setting”. In: *NeuroImage: Clinical* 6, pp. 445–454.
- Pérez-García, F., R. Sparks, and S. Ourselin (2021). “TorchIO: a Python library for efficient loading, preprocessing, augmentation and patch-based sampling of medical images in deep learning”. In: *Computer Methods and Programs in Biomedicine* 208, p. 106236.
- Pinaya, W. H., M. S. Graham, R. Gray, P. F. Da Costa, P.-D. Tudosiu, P. Wright, Y. H. Mah, A. D. MacKinnon, J. T. Teo, R. Jager, et al. (2022a). “Fast unsupervised brain anomaly detection and segmentation with diffusion models”. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, pp. 705–714.
- Pinaya, W. H., P.-D. Tudosiu, R. Gray, G. Rees, P. Nachev, S. Ourselin, and M. J. Cardoso (2022b). “Unsupervised brain imaging 3D anomaly detection and segmentation with transformers”. In: *Medical Image Analysis* 79, p. 102475.

- Pinon, N., G. Oudoumanessah, R. Trombetta, M. Dojat, F. Forbes, and C. Lartizien (2023a). “Brain subtle anomaly detection based on auto-encoders latent space analysis: application to de novo parkinson patients”. In: *2023 IEEE 20th International Symposium on Biomedical Imaging (ISBI)*. IEEE.
- Pinon, N., R. Trombetta, and C. Lartizien (2023b). “One-Class SVM on siamese neural network latent space for Unsupervised Anomaly Detection on brain MRI White Matter Hyperintensities”. In: *Medical Imaging with Deep Learning*.
- Quigley, H., S. J. Colloby, and J. T. O’Brien (2011). “PET imaging of brain amyloid in dementia: a review”. In: *International Journal of Geriatric Psychiatry* 26.10. DOI: [10.1002/gps.2640](https://doi.org/10.1002/gps.2640).
- Ranganath, R., D. Tran, and D. Blei (2016). “Hierarchical variational models”. In: *International conference on machine learning*. PMLR, pp. 324–333.
- Ravi, D., S. B. Blumberg, S. Ingala, F. Barkhof, D. C. Alexander, N. P. Oxtoby, A. D. N. Initiative, et al. (2022). “Degenerative adversarial neuroimage nets for brain scan simulations: Application in ageing and dementia”. In: *Medical Image Analysis* 75, p. 102257.
- Rezende, D. and S. Mohamed (2015). “Variational inference with normalizing flows”. In: *ICML*. PMLR, pp. 1530–1538.
- Rolls, E. T., C.-C. Huang, C.-P. Lin, J. Feng, and M. Joliot (2020). “Automated anatomical labelling atlas 3”. In: *Neuroimage* 206, p. 116189.
- Rolls, E. T., M. Joliot, and N. Tzourio-Mazoyer (2015). “Implementation of a new parcellation of the orbitofrontal cortex in the automated anatomical labeling atlas”. In: *Neuroimage* 122, pp. 1–5.
- Routier, A., N. Burgos, M. Díaz, M. Bacci, S. Bottani, O. El-Rifai, S. Fontanella, P. Gori, J. Guillon, A. Guyot, R. Hassanaly, T. Jacquemont, P. Lu, A. Marcoux, T. Moreau, J. Samper-González, M. Teichmann, E. Thibeau-Sutre, G. Vaillant, J. Wen, A. Wild, M.-O. Habert, S. Durrleman, and O. Colliot (2021). “Clinica: An Open-Source Software Platform for Reproducible Clinical Neuroscience Studies”. In: *Frontiers in Neuroinformatics* 15, p. 39. DOI: [10.3389/fninf.2021.689675](https://doi.org/10.3389/fninf.2021.689675).
- Samper-González, J., N. Burgos, S. Bottani, S. Fontanella, P. Lu, A. Marcoux, A. Routier, J. Guillon, M. Bacci, J. Wen, A. Bertrand, H. Bertin, M.-O. Habert, S. Durrleman, T. Evgeniou, and O. Colliot (2018). “Reproducible Evaluation of Classification Methods in Alzheimer’s Disease: Framework and Application to MRI and PET Data”. In: *NeuroImage* 183, pp. 504–521. DOI: [10.1016/j.neuroimage.2018.08.042](https://doi.org/10.1016/j.neuroimage.2018.08.042).
- Schlegl, T., P. Seeböck, S. M. Waldstein, G. Langs, and U. Schmidt-Erfurth (2019). “f-AnoGAN: Fast unsupervised anomaly detection with generative adversarial networks”. In: *Medical Image Analysis* 54. DOI: [10.1016/j.media.2019.01.010](https://doi.org/10.1016/j.media.2019.01.010).
- Schlegl, T., P. Seeböck, S. M. Waldstein, U. Schmidt-Erfurth, and G. Langs (2017). “Unsupervised Anomaly Detection with Generative Adversarial Networks to Guide Marker Discovery”. In: *Information Processing in Medical Imaging*. LNCS. Cham. DOI: [10.1007/978-3-319-59050-9_12](https://doi.org/10.1007/978-3-319-59050-9_12).
- Sevigny, J., P. Chiao, T. Bussière, P. H. Weinreb, L. Williams, M. Maier, R. Dunstan, S. Salloway, T. Chen, Y. Ling, et al. (2016). “The antibody aducanumab reduces A β plaques in Alzheimer’s disease”. In: *Nature* 537.7618, pp. 50–56.

- Sharp, P. F. and A. Welch (2005). *Practical Nuclear Medicine*. London: Springer, pp. 35–48.
- Shepp, L. A. and B. F. Logan (1974). “The Fourier Reconstruction of a Head Section”. In: *IEEE Transactions on Nuclear Science* 21.3, pp. 21–43. DOI: [10.1109/TNS.1974.6499235](https://doi.org/10.1109/TNS.1974.6499235).
- Shi, R., C. Sheng, S. Jin, Q. Zhang, S. Zhang, L. Zhang, C. Ding, L. Wang, L. Wang, Y. Han, et al. (2023). “Generative adversarial network constrained multiple loss autoencoder: A deep learning-based individual atrophy detection for Alzheimer’s disease and mild cognitive impairment”. In: *Human brain mapping* 44.3, pp. 1129–1146.
- Siddiquee, M. M. R., J. Shah, T. Wu, C. Chong, T. J. Schwedt, G. Dumkrieger, S. Nikolova, and B. Li (2023). “Brainomaly: Unsupervised Neurologic Disease Detection Utilizing Unannotated T1-weighted Brain MR Images”. In: *arXiv preprint arXiv:2302.09200*.
- Simarro Viana, J., E. de la Rosa, T. Vande Vyvere, D. Robben, D. M. Sima, and C.-T. P. a. Investigators (2020). “Unsupervised 3d brain anomaly detection”. In: *MICCAI Brainlesion Workshop*. Springer, pp. 133–142.
- Simonyan, K., A. Vedaldi, and A. Zisserman (2013). “Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps”. In: arXiv: [1312.6034](https://arxiv.org/abs/1312.6034).
- Simonyan, K. and A. Zisserman (2014). “Very deep convolutional networks for large-scale image recognition”. In: *arXiv preprint arXiv:1409.1556*.
- Smith, N. B. and A. Webb (2010). *Introduction to medical imaging: physics, engineering and clinical applications*. Cambridge university press.
- Snell, J., K. Ridgeway, R. Liao, B. D. Roads, M. C. Mozer, and R. S. Zemel (2017). “Learning to generate images with perceptual similarity metrics”. In: *ICIP*. IEEE, pp. 4277–4281.
- Solal, M., R. Hassanaly, and N. Burgos (2024a). “Leveraging healthy population variability in deep learning unsupervised anomaly detection in brain FDG PET”. In: *SPIE Medical Imaging*. San Diego (California), United States.
- (2024b). “Studying model variability in deep learning unsupervised anomaly detection in brain FDG PET”. In: *Submitted to Medical Imaging with Deep Learning*.
- Sønderby, C. K., T. Raiko, L. Maaløe, S. K. Sønderby, and O. Winther (2016). “Ladder variational autoencoders”. In: *Advances in neural information processing systems* 29.
- Suganyadevi, S, V Seethalakshmi, and K Balasamy (2022). “A review on deep learning in medical image analysis”. In: *International Journal of Multimedia Information Retrieval* 11.1, pp. 19–38.
- Sun, L., J. Wang, Y. Huang, X. Ding, H. Greenspan, and J. Paisley (2020). “An Adversarial Learning Approach to Medical Image Synthesis for Lesion Detection”. In: *IEEE Journal of Biomedical and Health Informatics* 24.8, pp. 2303–2314. DOI: [10.1109/JBHI.2020.2964016](https://doi.org/10.1109/JBHI.2020.2964016).
- Thibeau-Sutre, E., M. Díaz, R. Hassanaly, O. Colliot, and N. Burgos (2022a). “A glimpse of ClinicaDL, an open-source software for reproducible deep learning in neuroimaging”. In: *Medical Imaging with Deep Learning*.
- Thibeau-Sutre, E., M. Díaz, R. Hassanaly, A. Routier, D. Dormont, O. Colliot, and N. Burgos (2022b). “ClinicaDL: An open-source deep learning software for reproducible neuroimaging processing”. In: *Computer Methods and Programs in Biomedicine* 220. DOI: [10.1016/j.cmpb.2022.106818](https://doi.org/10.1016/j.cmpb.2022.106818).

- Thomas, B. A., V. Cuplov, A. Bousse, A. Mendes, K. Thielemans, B. F. Hutton, and K. Erlandsson (2016). “PETPVC: a toolbox for performing partial volume correction techniques in positron emission tomography”. In: *Physics in Medicine & Biology* 61.22, p. 7975.
- Tian, Y., G. Pang, F. Liu, Y. Chen, S. H. Shin, J. W. Verjans, R. Singh, and G. Carneiro (2021). “Constrained contrastive distribution learning for unsupervised anomaly detection and localisation in medical images”. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, pp. 128–140.
- Tolstikhin, I., O. Bousquet, S. Gelly, and B. Schoelkopf (2018). “Wasserstein Auto-Encoders”. In: *ICLR*.
- Tomczak, J. and M. Welling (2018). “VAE with a VampPrior”. In: *International Conference on Artificial Intelligence and Statistics*. PMLR, pp. 1214–1223.
- Tournier, J.-D., F. Calamante, and A. Connelly (2012). “MRtrix: diffusion tractography in crossing fiber regions”. In: *International journal of imaging systems and technology* 22.1, pp. 53–66.
- Uzunova, H., S. Schultz, H. Handels, and J. Ehrhardt (2019). “Unsupervised pathology detection in medical images using conditional variational autoencoders”. In: *IJCARS* 14, pp. 451–461.
- Vahdat, A. and J. Kautz (2020). “NVAE: A deep hierarchical variational autoencoder”. In: *Advances in Neural Information Processing Systems* 33, pp. 19667–19679.
- Vaillant, G., N. Gensollen, M. Joulot, O. El-Rifai, M. Diaz, O. Colliot, and N. Burgos (2023). “From Nipype to Pydra: a Clinica story”. In: *OHBM 2023-Annual meeting of the Organization for Human Brain Mapping*.
- Van Den Oord, A., O. Vinyals, et al. (2017). “Neural discrete representation learning”. In: *Advances in NeurIPS* 30.
- Van Dyck, C. H., C. J. Swanson, P. Aisen, R. J. Bateman, C. Chen, M. Gee, M. Kanekiyo, D. Li, L. Reyderman, S. Cohen, et al. (2023). “Lecanemab in early Alzheimer’s disease”. In: *New England Journal of Medicine* 388.1, pp. 9–21.
- Varoquaux, G. and O. Colliot (2022). “Evaluating machine learning models and their diagnostic value”. In: *Machine Learning for Brain Disorders*. Springer, pp. 601–630.
- Venkatraghavan, V., S. R. v. d. Voort, D. Bos, M. Smits, F. Barkhof, W. J. Niessen, S. Klein, and E. E. Bron (2023). “Computer-aided diagnosis and prediction in brain disorders”. In: *Machine Learning for Brain Disorders*. Springer, pp. 459–490.
- Wang, R., V. Bashyam, Z. Yang, F. Yu, V. Tassopoulou, S. S. Chintapalli, I. Skampardoni, L. P. Sreepada, D. Sahoo, K. Nikita, et al. (2023). “Applications of generative adversarial networks in neuroimaging and clinical neuroscience”. In: *Neuroimage*, p. 119898.
- Wang, Z., A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli (2004). “Image quality assessment: from error visibility to structural similarity”. In: *IEEE Transactions on Image Processing* 13.4, pp. 600–612.
- Wang, Z., E. P. Simoncelli, and A. C. Bovik (2003). “Multiscale structural similarity for image quality assessment”. In: *The Thirty-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*. Vol. 2. Ieee, pp. 1398–1402.

- Wargnier-Dauchelle, V., T. Grenier, F. Durand-Dubief, F. Cotton, and M. Sdika (2023). “A Weakly Supervised Gradient Attribution Constraint for Interpretable Classification and Anomaly Detection”. In: *IEEE Transactions on Medical Imaging*.
- Wen, J., E. Thibeau-Sutre, M. Diaz-Melo, J. Samper-González, A. Routier, S. Bottani, D. Dormont, S. Durrleman, N. Burgos, and O. Colliot (2020). “Convolutional neural networks for classification of Alzheimer’s disease: Overview and reproducible evaluation”. In: *Medical Image Analysis* 63, p. 101694. DOI: [10.1016/j.media.2020.101694](https://doi.org/10.1016/j.media.2020.101694).
- Wolleb, J., F. Bieder, R. Sandkühler, and P. C. Cattin (2022). “Diffusion models for medical anomaly detection”. In: *International Conference on Medical image computing and computer-assisted intervention*. Springer, pp. 35–45.
- Worsley, K. J., J Taylor, F Carbonell, M Chung, E Duerden, B Bernhardt, O Lyttelton, M Boucher, A Evans, et al. (2009). “A Matlab toolbox for the statistical analysis of univariate and multivariate surface and volumetric data using linear mixed effects models and random field theory”. In: *NeuroImage Organisation for Human Brain Mapping 2009 Annual Meeting*. Vol. 47, S102.
- Wyatt, J., A. Leach, S. M. Schmon, and C. G. Willcocks (2022). “Anoddpm: Anomaly detection with denoising diffusion probabilistic models using simplex noise”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 650–656.
- Xia, T., A. Chatsias, and S. A. Tsiftaris (2019). “Adversarial Pseudo Healthy Synthesis Needs Pathology Factorization”. In: *International Conference on Medical Imaging with Deep Learning*. PMLR, pp. 512–526.
- (2020). “Pseudo-Healthy Synthesis with Pathology Disentanglement and Adversarial Learning”. In: *Medical Image Analysis* 64, p. 101719. DOI: [10.1016/j.media.2020.101719](https://doi.org/10.1016/j.media.2020.101719).
- Zhang, C., H. Zheng, and Y. Gu (2023). “Dive into the details of self-supervised learning for medical image analysis”. In: *Medical Image Analysis*, p. 102879.
- Zhao, S., J. Song, and S. Ermon (2019). “Infovae: Balancing learning and inference in variational autoencoders”. In: *Proc AAAI conference on artificial intelligence*. Vol. 33, pp. 5885–5892.
- Zhou, S. K., H. Greenspan, C. Davatzikos, J. S. Duncan, B. Van Ginneken, A. Madabhushi, J. L. Prince, D. Rueckert, and R. M. Summers (2021). “A Review of Deep Learning in Medical Imaging: Imaging Traits, Technology Trends, Case Studies With Progress Highlights, and Future Promises”. In: *Proceedings of the IEEE* 109.5, pp. 820–838. DOI: [10.1109/JPROC.2021.3054390](https://doi.org/10.1109/JPROC.2021.3054390).
- Zhou, X., S. Niu, X. Li, H. Zhao, X. Gao, T. Liu, and J. Dong (2023). “Spatial-contextual variational autoencoder with attention correction for anomaly detection in retinal OCT images”. In: *Computers in Biology and Medicine* 152, p. 106328.
- Zhou, Z., M. M. Rahman Siddiquee, N. Tajbakhsh, and J. Liang (2018). “Unet++: A nested u-net architecture for medical image segmentation”. In: *MICCAI Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support Workshop*. Springer, pp. 3–11.
- Zimmerer, D., F. Isensee, J. Petersen, S. Kohl, and K. Maier-Hein (2019). “Unsupervised Anomaly Localization Using Variational Auto-Encoders”. In: *International Conference*

- on International conference Medical Image Computing and Computer Assisted Intervention*. LNCS. Springer, pp. 289–297. DOI: [10.1007/978-3-030-32251-9_32](https://doi.org/10.1007/978-3-030-32251-9_32).
- Zimmerer, D., S. A. A. Kohl, J. Petersen, F. Isensee, and K. H. Maier-Hein (2018). *Context-encoding Variational Autoencoder for Unsupervised Anomaly Detection*. arXiv: [1812.05941](https://arxiv.org/abs/1812.05941).