Decision support system for enhancing health care services to reduce potentially avoidable hospitalizations

Tu Ngo

THÈSE POUR OBTENIR LE GRADE DE DOCTEUR DE L'UNIVERSITÉ DE MONTPELLIER

En Informatique

École doctorale : Information, Structures, Systèmes

Unité de recherche LIRMM

En partenariat avec DIM, CHU de Montpellier

Conception d'un Systeme Decisionnel pour la Reduction des Hospitalisations Potentiellement Evitables

Présentée par Tu NGO

Le 21 juillet 2020

Sous la direction de Prof. Anne LAURENT
et Prof. Carmen GERVET
et Dr. Grégoire MERCIER

Devant le jury composé de

Sandra BRINGAY, Professeur, Université Paul Valéry	Présidente
Adrien COULET, MCF HDR, Université de Lorraine	Rapporteur
Ricard GAVALDA, Professeur, Universitat Politècnica de Catalunya	Rapporteur
Vera GEORGESCU, Ingénieure de Recherche, CHU Montpellier	Examinatrice
Carmen GERVET, Professeur, Université de Montpellier	Co-encadrant
Anne LAURENT, Professeur, Université de Montpellier	Directrice
Thérèse LIBOUREL, Professeur émérite, Université de Montpellier	Invitée
Grégoire MERCIER, Praticien Hospitalier, CHU Montpellier	Co-encadrant
Catherine QUANTIN, Professeur - Praticien Hospitalier, CHU Dijon	Examinatrice

UNIVERSITÉ DE MONTPELLIER

Résumé

Les hospitalisations potentiellement évitables (HPE) sont les admissions à l'hôpital qui auraient pu être évitées grâce à des traitements rapides et efficaces. Les taux élevés d'hospitalisations potentiellement évitables sont associés à de nombreux facteurs. Ces facteurs comprennent des taux de mortalité élevés, une faible densité de médecins de soins primaires, le manque de continuité des soins et le manque d'accès aux soins primaires, le faible revenu médian ou le faible niveau d'instruction ainsi que les caractéristiques organisationnelles des systèmes comme une mauvaise coordination entre les professionnels de santé. La France compte environ 300 000 hospitalisations potentiellement évitables chaque année. Ces hospitalisations évitables sont associées à un coût de plusieurs centaines de millions d'euros pour l'assurance maladie. En d'autres termes, la réduction des hospitalisations potentiellement évitables améliore non seulement la qualité de vie des patients, mais pourrait également économiser des coûts substantiels grâce au traitement des patients. Par conséquent, les autorités sanitaires sont intéressées par des solutions améliorant les services de santé pour réduire les hospitalisations potentiellement évitables.

Certaines études récentes en France ont suggéré que l'augmentation du nombre d'infirmières dans certaines zones géographiques pourraient entraîner une réduction des taux d'hospitalisations potentiellement évitables dans ces secteurs. Les autorités sanitaires pourraient recommander les zones géographiques d'installation des infirmiers et infirmières sur la base de statistiques descriptives telles que des taux élevés d'hospitalisations potentiellement évitables. Ces approches statistiques descriptives ont des limites car les autres les facteurs associés aux hospitalisations potentiellement évitables mentionnés ci-dessus ont été ignorés. Par conséquent, nous abordons l'apprentissage automatique qui a été largement appliqué dans le but d'améliorer les services des prestataires de santé et donc d'améliorer la santé de la population. En particulier, comme les taux d'hospitalisations potentiellement évitables sont des valeurs numériques, toute méthode de régression pourrait être considérée. Afin de sélectionner la méthode la plus adaptée, nous avons évalué le potentiel ainsi que la qualité des méthodes de régression courantes. Celles-ci comprennent (1) la régression multilinéaire; (2) K plus proches voisins pour la régression; (3) les réseaux de régression; (4) les machines à vecteurs de support pour la régression. Les performances ont été mesurées et validées en considérant l'erreur quadratique moyenne et les méthodes de leave-one-out. Nous avons sélectionné la machine à vecteurs de support pour la régression pour notre travail. De plus, dans cette approche, outre la prise en compte de tous les facteurs potentiels, nous prenons également en compte les contraintes liées aux revenus et à l'égalité d'accès aux soins. En particulier, nous avons étendu les machines à vecteur de support régression à l'information spatiale en ajoutant ces contraintes. Cette approche nous permet de sélectionner non seulement les zones géographiques mais aussi le nombre d'infirmiers et d'infirmières à ajouter dans ces zones pour maximiser la réduction du nombre d'hospitalisations potentiellement évitables. Plus précisément, notre approche est appliquée en Occitanie, en France et les zones géographiques mentionnées ci-dessus sont les espaces de vie transfrontaliers (Bassins de vie - BVs). Cependant, notre approche

peut être considérée à un niveau national ou vers d'autres régions ou pays. De plus, puisque nous visons à construire un système d'aide à la décision, les résultats de nos travaux sont visualisés sur des cartes spatiales.

D'un autre côté, il est clair qu'il y a de forts impacts de températures extrêmement froides et chaudes (ou canicule) à la santé humaine. Cela signifie que la température extrême pourrait être un facteur potentiellement associé à des taux élevés d'hospitalisations potentiellement évitables. Par conséquent, une partie de notre travail consiste à mesurer l'impact des températures extrêmes sur les hospitalisations potentiellement évitables. Nous avons de plus inclus ces données environnementales dans notre approche ci-dessus. En particulier, nous avons utilisé les valeurs de température mesurées toutes les heures par des capteurs dans les stations météorologiques. Cependant, ces les valeurs sont parfois discontinues et nous avons besoin d'une méthode d'imputation pour ces valeurs manquantes. D'autre part, en particulier dans notre travail, lorsque nous définissons les événements de canicule, 0,5 degré La différence en degrés Celsius peut changer les résultats. Par conséquent, la méthode d'imputation doit être fiable. Dans la littérature, il existe de nombreuses approches pour traiter cette étape de traitement. Deux plus populaires sont celles qui exploitent soit la composante spatiale, soit la composante temporelle du données de température. Respectivement, ces approches sont des méthodes d'interpolation spatiale telles que les modèles pondérés en fonction de la distance (IDW) et chronologiques tels que les modèles ARIMA. Pour nous aider à choisir la méthode la plus fiable, nous comparons d'abord les performances des deux approches en imputation de température manquante. De plus, comme chaque approche ci-dessus exploite une dimension différente des données spatio-temporelles, nous proposons une nouvelle approche qui combine les deux dimensions pour améliorer les performances en termes de qualité. Plus précisément, au lieu d'appliquer directement la méthode IDW ou le modèle ARIMA, nous calculons d'abord les valeurs estimées par ces méthodes, puis les utiliser comme variables d'entrée d'un apprentissage automatique supplémentaire. Pour mener à bien notre travail, nous avons collecté les données de température qui sont mesurées toutes les heures en mai 2019 à partir de plus de 600 stations météo implantées en France métropolitaine. Pour évaluer les performances de toutes les approches, nous utilisons l'erreur quadratique moyenne entre la température estimée et la température observée aux stations météorologiques. Nos expériences sont validées avec la méthode du leave-one-out. Les résultats montrent que (1) ARIMA fonctionne généralement mieux que IDW et (2), par rapport aux méthodes IDW et ARIMA, notre approche fonctionne mieux à respectivement 100% et 99,8% (604 sur 605) des stations météorologiques.

De plus, comme mentionné ci-dessus, les taux élevés d'hospitalisations potentiellement évitables sont associés à des caractéristiques organisationnelles des systèmes de santé telles que la coordination entre les les fournisseurs de soins. En d'autres termes, l'amélioration de la coordination entre les professionnels de santé pourrait conduire à la réduction des hospitalisations potentiellement évitables. En outre, dans les cas où les patients changent d'hôpital pour des traitements, il est clair que le traitement serait plus efficace et le risque pour la santé des patients serait éliminé ou réduit si les hôpitaux ultérieurs pouvaient accéder à les dossiers médicaux des patients des hôpitaux précédents. Par conséquent, il apparait opportun d'autoriser les systèmes d'information à partager les dossiers médicaux entre les hôpitaux, sauf à ce que tous les hôpitaux de France soient regroupés en un seul. Or cela serait coûteux alors que certains hôpitaux n'ont jamais partagé aucun dossier patient. En attendant, les flux de patients l'évolution des hôpitaux pour les traitements peuvent être présentés par un graphe non orienté dans lequel les nœuds sont les hôpitaux tandis que les arcs représentent les flux de patients. Par conséquent, nous proposons des approches utilisant des méthodes de regroupement des graphiques pour regrouper ces hôpitaux en

communautés. En particulier, afin de sélectionner la méthode de regroupement des graphes pour notre travail, nous comparons les performances de deux méthodes différentes de regroupement des graphes. Ces méthodes sont le clustering spectral et les méthodes de Louvain. De plus, nous devons considérer plusieurs options de regroupement des hôpitaux dans les communautés. Par exemple, une option est que chaque cluster final doit contenir un hôpital universitaire public (Centre Hospitalier Universitaire - CHU). Ces contraintes sont ajoutées à notre mise en œuvre par personnalisation de la méthode de regroupement de graphes sélectionnée qui est la méthode de Louvain. En conséquence, notre travail consiste à segmenter les hôpitaux de France en 19 communautés dont 17 communautés en France métropolitaine sont visualisés sur une carte spatiale.

En résumé, notre travail présente un outil pour sélectionner le nombre optimal d'infirmières à mettre en place dans les zones géographiques pour la plus forte réduction du nombre d'hospitalisations potentiellement évitables en étendant les méthodes de machines à vecteurs de support pour la régression à l'information spatiale. Nous avons également travaillé sur l'extension de la méthode pour inclure les données de température et nous avons proposé une nouvelle approche qui améliore les performances d'imputation de température manquante. Enfin, pour améliorer la coordination entre les professionnels de santé afin de réduire les hospitalisations potentiellement évitables, La méthode de Louvain a été personnalisée pour proposer des regroupements d'hôpitaux français.

Asbtract

Potentially avoidable hospitalizations (PAHs) are the hospital admissions that could have been prevented with timely and effective treatments. The high rates of potentially avoidable hospitalizations are associated with many factors. These factors include high mortality rates, low density of primary care physicians, lack of continuity of care, and lack of access to primary care, low median income or low education levels as well as organizational features of health systems such as poor coordination between health care providers. On the other side, in France, there are about 300,000 potentially avoidable hospitalizations every year. These preventable hospitalizations are associated with a cost of several hundred million Euros for the Health Insurance. In other words, reducing potentially avoidable hospitalizations not only enhances patients' quality of life but also could save substantial costs due to patient treatments. Therefore, health authorities are highly interested in solutions improving health care services to reduce the potentially avoidable hospitalizations.

Some recent studies in France have suggested that increasing the number of nurses in selected geographic areas could lead to the reduction of the rates of potentially avoidable hospitalizations in those areas. However, health authorities could select the geographic areas for new nurse implementation only based on descriptive statistics such as high rates of potentially avoidable hospitalizations. Clearly, these descriptive-statistics approaches have limitations because the other factors associated with potentially avoidable hospitalizations mentioned above have been ignored. Therefore, we approach machine learning that has been widely applied in the healthcare sector to improve the services of health providers and therefore improve population health. In particular, since the rates of potentially avoidable hospitalizations are numeric values, any regression method could be the option for our approach. In order to select the most suitable method, we have evaluated the potential as well as the quality performance of the common regression methods. These methods include (1) Multilinear regression; (2) K–nearest neighbors for regression; (3) Neural networks for regression; (4) Support vector machine for regression. Based on the performances which were measured and validated by root-mean-square error and leave-one-out methods, we have selected the support vector machine for regression for our work. In addition, in this approach, besides considering all the potential factors, we also take into account the constraints related to the budget and the equality of healthcare access. In particular, we extended the support vector machine for regression to spatial information by adding these constraints. This approach allows us to select not only the geographic areas but also the number of to-be-added nurses in these areas for the biggest reduction in the number of potentially avoidable hospitalizations. Specifically, our approach is applied in the Occitanie region, France and geographic areas mentioned above are the cross-border living areas (fr. Bassins de vie - BVs). However, our approach can be extended at the national level or to other regions or countries. In addition, since we aim at building a user-friendly decision support system, the results of our work are visualized on spatial maps.

On the other side, it is clear that there are strong impacts of extreme cold and hot temperature

(or heatwave) to human health. That means that the extreme temperature could be one potential factor associated with high rates of potentially avoidable hospitalizations. Therefore, a part of our works is to measure the impact of the extreme temperature to potentially avoidable hospitalizations as well as to include this environmental data in our approach above. In particular, we used the temperature values measured hourly by sensors at the weather stations. However, these values are sometimes discontinuous and we need an imputation method for these missing values. On the other hand, particularly in our work, when we define the heatwave events, 0.5 degree Celsius difference can change the results. Therefore, the imputation method must be reliable. In the literature, there are many approaches to deal with this processing step. Two most popular approaches are the ones that exploit either the spatial component or temporal component of the temperature data. Respectively, these approaches are spatial interpolation methods such as Inverse Distance Weighted (IDW) and time-series models such as Autoregressive Integrated Moving Average (ARIMA). To help us select the more reliable method, we first compare the performances of both approaches in missing temperature imputation. In addition, as each approach above only exploits one different dimension of the spatio-temporal data, we propose a novel approach that combines both dimensions to improve the performance in terms of quality. Specifically, instead of applying directly the IDW method or the ARIMA model, we firstly compute the estimated values by these methods and then use them as the input variables of an additional machine learning method. To conduct our work, we collected the temperature data that is measured hourly in May 2019 from more than 600 weather stations implemented across Metropolitan France. To evaluate the performances of all approaches, we use the root-mean-square-error between the estimated temperature and the observed temperature at the weather stations and our experiments are validated with the leave-one-out method. The results show that (1) ARIMA generally performs better than IDW and (2), compared with IDW and ARIMA methods, our approach performs better at 100% and 99.8% (604 over 605) weather stations respectively.

In addition, as mentioned at the beginning, the high rates of potentially avoidable hospitalizations are associated with organizational features of health systems such as coordination between health care providers. In other words, improving the coordination between the health care providers could lead to the reduction of the potentially avoidable hospitalizations. Moreover, in the cases that the patients change hospitals for treatments, it is clear that the treatment would be more efficient and the risk on patients' health would be eliminated or reduced if the later hospitals are able to access the medical records of the patients at the previous hospitals. Therefore, allowing the information technology systems to share medical records among the hospitals is needed. However, it is neither necessary nor practical if all the hospitals in France are grouped as one because it would be costly while some hospitals have never been sharing any patient. In the meantime, the flows of patients changing hospitals for the treatments can be presented by an undirected graph in which the nodes are the hospitals while the edges present the patient flows. Therefore, we propose the approaches of using graph clustering methods to cluster these hospitals into communities. Particularly, in order to select the graph clustering method for our work, we compare the performance of two different graph clustering methods. These methods are spectral clustering and Louvain methods. In addition, we need to consider several options of clustering hospitals into the communities. For example, one option is that each final cluster must contain a public University Hospital (fr. Centre Hospitalier Universitaire - CHU). These constraints are added into our implementation by customizing the selected graph clustering method which is the Louvain method. As the result, our work has partitioned hospitals in France into 19 communities among which 17 communities in metropolitan France are visualized in a spatial map.

In summary, this work presents a tool for selecting the optimal number of nurses to be implemented in geographic areas for the biggest reduction in the number of potentially avoidable hospitalizations by extending the support vector machine for regression to spatial information. We also worked on extending the method to include temperature data and we have proposed a novel approach that improves the performance in missing temperature imputation. Finally, to improve the coordination between the health care providers as a way to reduce the potentially avoidable hospitalizations, Louvain method has been customized for clustering French hospitals into communities.

Acknowledgements

Undertaking this PhD has been a truly life-changing experience for me and it would not have been possible to do without the support and guidance that I received from many people.

Firstly, I would like to say a very big thank you to my supervisors Prof. Anne LAURENT, Prof. Carmen GERVET, and Dr. Grégoire MERCIER for all the supports and encouragements they gave me. Without their guidance and constant feedback this PhD would not have been achievable.

My sincere thanks also goes to Dr. Vera GEORGESCU and Prof. Therese LIBOUREL for their big supports by giving me strong academic advice during my research works.

I would like to thank Prof. Sandra BRINGAY for presiding the jury and I present my especial gratitude to Dr. Adrien COULET and Prof. Ricard GAVALDA and for their valuable feedback on this thesis report.

On the other side, I greatly appreciate the support I received from my colleagues at the DIM department CHU of Montpellier. Eespecially, I would like to say much thanks to Mr. Nicolas MALAFAYE who has helped me exporting the data needed for my study from the hospital database systems.

My special thanks to my dear colleague, Mrs. Jénica PASTOR, as she not only gave me administrative advice but also brings the family-style working environment at the economic evaluation unit where we can have fun together besides the works.

I also would like to thank the DIM department from the CHU of Montpellier as well as the OpenHealth Institute for funding my PhD.

Last but not the least, I would like to thank my family: my wife and to my children for supporting me spiritually throughout writing this thesis and my life in general.

Contents

List of Figures

List of Tables

List of Abbreviations

AHRQ: Agency for Healthcare Research and Quality

AR: AutoRegressive model

ARIMA: AutoRegressive Integrated Moving Average model

ARMA: AutoRegressive and Moving Average model

BV: fr. Bassins de Vie

CHU: fr. Centre Hospitalier Universitaire

CMU-C: fr. Couverture Maladie Universelle Complémentaire

CNN: Convolutional Neural Networks

COPD: Chronic Obstructive Pulmonary Disease

DIM: fr. Département de l'Information Médical

ICD-10: International Classification of Diseases, 10th revision

IDW: Inverse Distance Weighted

INSEE: fr. Institut National de la Statistique et des Études Économiques

IoT: Internet of Things

LSTM: Long Short-Term Memory

MA: Moving Average model

MAE: Mean Absolute Error

MAUP: Modifiable Areal Unit Problem

MLP: Multi-Layer Perceptron

NLP: Natural Language Processing

PAH: Potentially Avoidable Hospitalizations

PLA: Perception Learning Algorithm

PMSI: fr. Programme de Médicalisation des Systèmes d'Information

RBF: Gaussian radial Basic Function

RMSE: Root-Mean-Square Error

SAR: Spatial AutoRegressive

SC: Spectral Clustering

SLX: Spatial Lag X

SVM: Support Vector Machine

SVR: Support Vector machine for Regression

Chapter 1

Introduction

Contents

1.1 Context

1.1.1 Artificial intelligence in health sciences

Artificial intelligence has wide range of applications in this medicine and health care sector. These applications include the ones that allow patients to understand and manage their own health and symptoms. In the other side, artificial intelligence also assists clinician care teams in enhancing the quality and safety of care with the applications in early disease detection and prediction as well as in selection of optimal treatments so that we can enhance and optimize care delivery to patients. For example, artificial intelligence can be applied to detect diseases such as cancer [45] or diabetes [99] as well as to predict hospital readmission for patients with diabetes [38] or with heart failure [32]. Another type of application of artificial intelligence is to reduce medication errors. More specifically, artificial intelligence holds promise for improving medication error detection and reducing costs associated with adverse events [71]. Similarity, artificial intelligence can be applied to identify subgroups of patients for whom, for example, treatment A is more effective than treatment B, and vice versa. The treatment group identification step is of key importance to the development of personalized medicine [41]. The other side of artificial intelligence application in health care sector is to optimize hospital processes such as resource allocation and patient flow. More specifically, by early and accurate prediction of patients outcomes, we can better predict demand and allocate scarce hospital resources such as beds and operating rooms. For example, artificial intelligence is used to forecast hospital discharge volume [53] or is used in emergency department capacity planning [58]. The purpose of these applications is generally to improve quality of care and population health outcomes, while reducing healthcare costs. That is also the purpose of our works that aim at **helping the health authorities with a decision support system to reduce the number of potentially avoidable hospitalizations in France**.

1.1.2 Work context

The research projects that I have been involved are at the Economic Evaluation Unit, University Hospital of Montpellier, France. This unit is in charge of health services research, focusing on efficiency and equity (health care pathways and health access). In particular, its functions are to develop suitable methods for these topics, such as geographic variation analysis, description and classification of health care pathways. Moreover, this unit is also active in the assessment of health technology including: pharmaceuticals, medical device and equipment, and organizational and e-health innovations. The aims of the unit are to provide policy-makers with research information about the effectiveness, costs and organizational impact of innovations. With that goal, the unit designs studies aiming at assessing the cost-effectiveness, cost-utility, budget impact and out-of-

pocket burden of innovations. On the other side, there are health economics experts in about 30 national and international research projects in this unit. One of the projects is a national research project on potentially avoidable hospitalizations. This project was funded by the French Ministry of Health. Specifically, I have been working on this project since March 2017 as a Master 2 intern at first and then as a PhD candidate from December 2017. The specific purpose of the project is to approach machine learning in reducing the number of potentially avoidable hospitalizations in France.

1.1.3 Potentially avoidable hospitalizations (PAH)

As mentioned in the previous section, our work aims at recommending the health authorities related actions to reduce potentially avoidable hospitalizations (PAH). By definition, PAHs (also referred to as admissions for ambulatory care sensitive conditions) are the hospital admissions that could have been prevented [74]. In particular, these hospitalizations are in fact the consequence of the sudden aggravation of a chronic disease (diabetes, heart failure, respiratory failure). These acute episodes could have been prevented with timely and effective treatments and therefore the hospitalizations could have been avoided [12]. Technically, the PAHs can be identified based on the principal and related diagnoses which are associated to the codes of the 10th revision of the international statistical classification of diseases and related health problems (or ICD-10 codes). More specifically, in our works, the datasets of PAHs are extracted from French national hospital discharge database (section 1.3) by following the national guide (in French) [79]. In details, the PAH stays are categorized into 6 following groups which are defined by Agency for Healthcare Research and Quality (AHRQ) which is a U.S. federal agency.

- Asthma in adults (age >= 18)
- Congestive heart failure (age >= 40)
- Chronic obstructive pulmonary disease (COPD) (age >= 18)
- Dehydration in elderly people (age >= 65)
- Diabetes short-term complication (age >= 40)
- Angina without procedure (age >= 40, urgent admission)

In French context, there are about 300,000 cases of PAH per year with the rate of about 6 cases per 1,000 inhabitants. In which, about 50% of the cases are related to congestive heart failure. More specially, the details of PAHs in three years 2013, 2014 and 2015 are presented in the table 1.1.

Table 1.1: Potentially avoidable hospitalizations by AHRQ in France [79]

	2013		2014		2015	
	Nb	Rate	Nb	Rate	Nb	Rate
Asthma in adults	16,629	0.33	16,475	0.32	16,291	0.32
Congestive heart failure	146,851	2.90	149,561	2.94	156,545	3.07
COPD	65,160	1.29	63,625	1.25	67,936	1.33
Dehydration in elderly people	26,049	0.52	24,949	0.49	30,719	0.60
Diabetes short-term complication	5,904	0.12	5,956	0.12	6,234	0.12
Angina without procedure	34,252	0.68	33,976	0.67	31,515	0.62
TOTAL	**294,845**	**5.83**	**294,542**	**5.80**	**309,240**	**6.06**

Nb: Number of potentially avoidable hospitalizations

Rate: Number of potentially avoidable hospitalizations per 1,000 inhabitants

1.2 Objectives

As mentioned in the PAH section (section 1.1.3), every year, in France, there are about 300,000 PAHs. These preventable hospitalizations are associated with a cost of several hundred million Euros for the Health Insurance [14]. That means avoiding these hospital admissions not only could enhance quality of live of the patients but also could decrease substantial costs caused by patient treatments [54, 31]. Therefore, both the national- and regional-level health authorities in France are highly interested in enhancing the health care services in order to reduce the number of PAHs. Moreover, there are previous studies on PAHs and the potential factors that could be associated with high rates of PAHs [54, 33]. Some of the recent studies in France have revealed that the higher (age-and-sex-standardized) rates of PAHs are linked to higher mortality rates, lower density of acute care beds and ambulatory care nurses, lower median income, and lower education levels [54]. More specifically, these studies suggested that by increasing the number of nurses at some geographic areas, the number of PAHs in these areas could be reduced [54].

On the other hand, in France context, the public health decision makers can have influence on the factors related to health care such as the density of physicians, nurses, or the density of hospital beds. However, there are strong constraints in the healthcare system that the health decision makers need to take into account. In particular, the healthcare system must provide quality care while controlling associated costs and ensuring equality of access to the health care services. The latter states that all patient-citizens must be able to benefit from the care they need, regardless of their geographical and socioeconomic situation. Hence, being able to select geographic areas in order to maximize the impact of an intervention is of high importance.

These reasons gave birth to our first project that aims at building a decision support system for the biggest reduction of PAH numbers while integrating the socio-economic constraints such as the limited budget for health care service improvement and the equality of health care access. More specifically, our work is going to recommend not only geographic areas for improving health care service but also the optimized actions at these areas. To achieve that goal, artificial intelligence methods as mentioned in section 1.1.1 show the potential approach to us. In particular, since the target of our project is the rate of PAHs which are numerical values, any regression method could be a solution to our problem. More specifically, after analyzing the potential regression methods, we integrate the constraints into the most suitable regression method in building up the decision support system.

In addition, parts of our work are to collect data that could be the potential determinants of PAHs. There data can be obtained from many sources including the French Ministry of Health, the National Institute for Statistics and Economic Studies, the Regional Health Agency of Occitanie, or French health insurance fund ambulatory care claims database as well as open data. In particular, data of primary care supply and hospital supply, socioeconomic data such as education or income, epidemiological data such as mortality rates are taken into account. In addition, it is clear that temperature, especially temperature extremes, have negative impacts to human health. For example, the extreme heat (or so called heatwave) that occurred in summer 2003 in France caused about 15,000 more deaths than expected in France (an increase of 55%) [28]. Therefore, we want to include the data of temperature in our work. To collect the temperature data, we rely on the temperature values measured by sensors at weather stations. However, for many reasons the values measured at these stations are sometimes discontinuous. In other words, there are missing values for temperatures measured at the weather stations. To select the reliable method in missing temperature imputation, we compare the quality performance of two different methods representative for both the spatial interpolation methods and the time-series models. Then, we search for a novel approach that combines these methods to improve the quality performance.

On the other side, the high rates of potentially avoidable hospitalizations are associated with organizational features of health systems such as coordination between health care providers. That is because patients frequently change hospitals, especially for the management of chronic diseases. There any many reasons for that. For example, patients have changed their addresses, they are not happy with the service of the previous hospital, or they need to seek specialized care in a tertiary hospital. In such cases, it is clear that the treatment would be more efficient and the risk to patients' health could be eliminated or reduced if the later hospitals are able to access the medical records of the patients at the previous hospitals. In other words, the information technology systems that allow sharing medical records among the hospitals are needed. However, it is neither necessary nor practical for all hospitals in France to be grouped as one because it would be costly while some hospitals will never share any patient. Therefore, health authorities are interested in building hospital communities so that medical records can be shared among the

hospitals in those communities. This brought up us another project which aims at splitting French hospital networks into communities for sharing patients' medical records. Particularly, our work is based on the flows of patients changing the hospitals for the treatments. These flows can be presented by a undirected weighted graph in which the nodes present the hospitals while the edges present the size of patient flows. Therefore, to cluster these hospitals into communities, we rely on the graph clustering methods. Particularly, we evaluate different graph clustering methods in order to select the most suitable method for our work. In addition, we need to consider several options of clustering hospitals into the communities. For example, one option is that each final cluster must contain a public University Hospital (fr. Centre Hospitalier Universitaire - CHU). The selected graph clustering method will be customized to include these constraints in clustering French hospitals into communities.

To sum up, our works include three parts:

- Extending the most suitable regression method to spatial information after analyzing the potentials of different regression methods in building the decision support system related to PAHs. More specifically, the system is to recommend health decision makers not only geographic areas for improving health care service but also the optimized actions at these areas for the biggest reduction of PAH numbers. Furthermore, since we aim at building a user-friendly decision support system, the results of our work are visualized on spatial maps.
- Proposing a novel and reliable approach in missing temperature imputation after comparing the quality performance of two different methods representative for both the spatial interpolation methods and the time-series models.
- Customizing the suitable graph clustering method after evaluating two different methods to include constraints in partitioning all public and private French hospitals into communities. The results are also presented on spatial maps.

1.3 Overview of data sources

In our work, the datasets are collected from many sources including the French Ministry of Health, the National Institute for Statistics and Economic Studies, the Regional Health Agency of Occitanie, French Health Insurance Fund ambulatory care claims database as well as open data. However, the main datasets are the patient datasets which are exported from the French National Hospital Discharge Database (fr. Programme de Médicalisation des Systèmes d'Information - PMSI). This PMSI database stores hospitalisation data from all French public and private hospitals. The database contains a record for each acute inpatient stay and represents about 25 million

records per year [13]. The records describe the stays in a standardized data set. In particular, the records include information about the discharge diagnoses (principal, related, associated in ICD-10 codes), the medical procedures with specific coding performed during hospital stay, as well as diagnosis-related groups (fr. Groupe Homogène de Malades) to classify patients in subgroups according to medical procedures and discharge diagnoses. In addition, the lengths of the stays as well as specific aspects of the stays (for instance, a stay in an intensive care unit) are included. In term of security and privacy reasons, no plain patient identity information are available. Instead, pseudonyms are used for record linkage. In addition, a specific geographic codes which are roughly equivalent to postal codes are used instead of patients' details addresses. This database is available upon registration with and payment to a habilitated provider, or through collaboration with a French university hospital health information management department. For example, since our work is at the Economic Evaluation Unit that is actual a part of the department of medical information (fr. Département de l'Information Médicale - DIM) of University Hospital of Montpellier, our project team is able to access the PMSI system.

1.4 Thesis organization

The thesis includes six following chapters:

- Chapter 1: Introduction. In this chapter, the objectives of our works are presented.
- Chapter 2: Related Works. In this chapter, the literature reviews of machine learning in health sciences as well as the previous studies on PAHs are presented. Moreover, methods for health data standardization as well as a brief introduction about spatial analysis are also provided in this chapter.
- Chapter 3: Regression methods for enhancing health care service to reduce PAHs. In this chapters, four regression methods are introduced. The potential applications of these methods to build the decision support system mentioned the objective section are evaluated and compared. After selecting the most suitable method, the constraints of the health system are taken into account while building the system that recommend not only geographic areas for adding nurses but also the number of to-be-added nurses at these areas for the biggest reduction of PAH numbers. The results are visualized on spatial maps.
- Chapter 4: Spatial interpolations and time-series models and the combination to improve temperature missing imputation. In this chapter, two methods representative for spatial interpolations and time-series models are introduced. The performance of these two methods are presented before a novel approach that combines the results of the two methods are proposed to improve temperature missing imputation.

- Chapter 5: Graph clustering approaches for hospital communities. In this chapter, two different graph clustering methods are introduced. The results of the two methods on PMSI dataset are compared and evaluated in order to select the more suitable one for our work. The results of clustering French public and private hospitals into communities are visualized on spatial maps.
- Chapter 6: Conclusion. In this chapter, the summary of the work and discussion are provided.

Chapter 2

Related works

Contents

2.1 Machine learning

Machine learning is an sub-domain of artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed. It relies on underlying hypothesis of creating the model and tries to improve it by fitting more data into the model over time [56, 96]. There are different approaches to getting machines to learn, from using basic decision trees to clustering to layers of artificial neural networks (the latter of which has given way to deep learning), depending on what task we are trying to accomplish and the type and amount of data that you have available. One way to represent machine learning algorithms is to sub-categorize them by how they learn inference from the data (as shown in figure 2.1). The subcategories are unsupervised learning, supervised learning, and reinforcement learning.

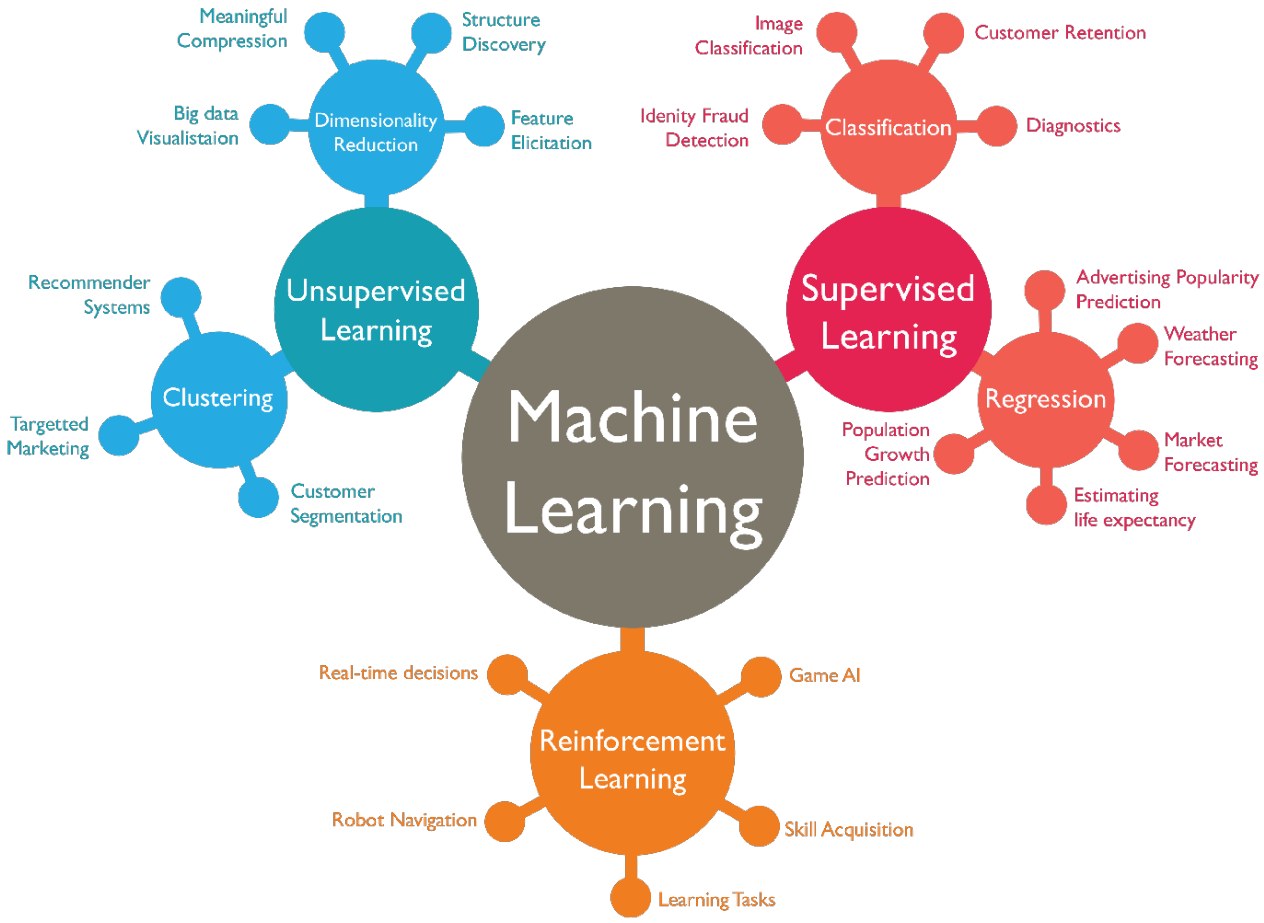


Figure 2.1: Sub-categorize machine learning [21]

Machine learning has wide range of applications in many domains. For example, in automotive, self-driving car could be the best example of machine learning application. In concept, a self-driving car is capable of sensing its environment and navigating without human input. To accomplish this task, each self-driving car is usually equipped with GPS, an navigation system, and a range of sensors including laser rangefinders, radar, and video cameras. Data collected

from the sensors are analyzed to understand the environments from then make right decision. For example, objects filmed by the cameras are recognized by sophisticated image recognition called computer vision using deep learning. Another example of application is spam email detection. There are actually many machine learning techniques that can be applied for this classification work. These techniques such as Naïve Bayesian, Support Vector Machine or K Nearest Neighbor analyses words, the occurrence, and distributions of words and phrases in the content of emails and used then use generated rules to filter the incoming email spams [19]. Like other sectors, health science sector has been also taking the advantage of machine learning for decades. In the next section, we present in more details the application of machine learning in health care sector.

2.2 Machine learning in health sciences

To present machine learning in health sciences, we categorize the users (or stakeholders) of the applications in health care sector into:

- Patients
- Clinician care teams
- Back office teams
- Health authorities

2.2.1 Patients as users

The first application of this category could be virtual agents such as Ada application [2] which play roles like virtual doctors or nurses. Like the application in customer service sector, these virtual agents could help patients monitor health and symptoms at home. For example, the patients can search for medical advice by providing symptoms to the agents. The way agents communicate with the user via speech recognition or natural language processing (NLP) [48].

Another way the patients can monitor their health status is through wearable and smart devices. These devices such as accelerometers, gyroscopes, microphones, cameras, and other sensors generate the raw data of the person who caries the devices. Machine learning algorithms can be trained to recognize patterns from the raw data inputs and then categorize these patterns as indicators of an individual's behavior and health status. These systems can allow patients to understand and manage their own health and symptoms as well as share data with medical providers.

Figure 2.2: Ada application [80]

2.2.2 Clinician care teams as users

Clinician care teams or in other words are those who deliver health care to patients. These type of users include specialists, nurses, physician assistants, pharmacists, and other health care professionals. Applications of machine learning would enhance the quality and safety of care. In particular, machine learning is applied in early disease detection and prediction as well as in selection of optimal treatments so that we can enhance and optimize care delivery to patients.

In **disease detection and prediction**, the main sub category of machine learning is classification. For instance, machine learning classifiers have already demonstrated strong performance in image-based diagnoses. As an example, deep convolutional neural networks (CNNs) (figure 2.3) is used to diagnose skin cancer [26]. Another example, multiple machine learning algorithms including long short-term memory (LSTM), CNN and support vector machine (SVM) for classification are deployed for early detection of diabetes [86]. Prediction such as hospital readmission is also benefited from machine learning classification. For example, multi-layer perceptron (MLP)-based approach is used to predict heart failure patients to be readmitted or death in 30 days after hospital discharge [7]. Although some believe that could replace physicians in diagnostic, but it would be better to use approaches of machine learning as assistance in diagnostic prediction to decrease human errors by physicians.

In **surgery**, machine learning is becoming more important for surgical decision making as it can use diverse sources of information such as patient risk factors, anatomic information, disease history to help physicians and patients make better predictions regarding the consequences of

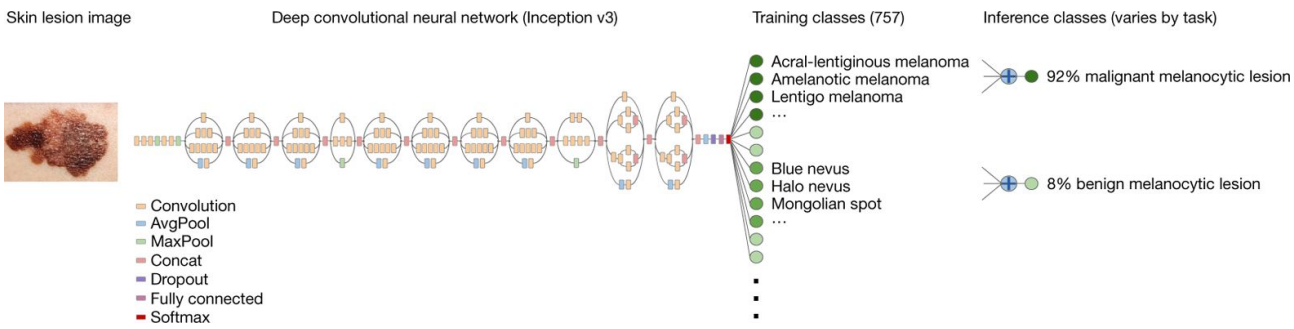


Figure 2.3: A deep convolutional neural networks models to diagnose skin cancer [26]

surgical decisions. For example, a deep learning model is applied to determine seizure control after epilepsy surgery [34]. The other main application of machine learning in surgery is in surgical robots which are able to control the trajectory, depth, and speed of their movements with great precision. With the integration of artificial intelligence, surgical robotics would be able to perceive and understand complicated surroundings, conduct real-time decision making and perform surgical tasks with increased precision, safety, automation, and efficiency [100]

Machine learning can be applied to provide **personalized treatment** to patients. Precision medicine allows clinicians to tailor medical treatment to the individual patients through the identification of common features, including their genetics, environments, and medical histories. For example, in the treatment of cardiovascular disease, there are different available drugs. To provide effective treatments to patients, responses of drugs on the other patients from should be studied. On the other sides, grouping these patients based on their similarity in order to generate reliable predictions of drug response. To sum up, the learning process of personalized medicine is presented in figure 2.4

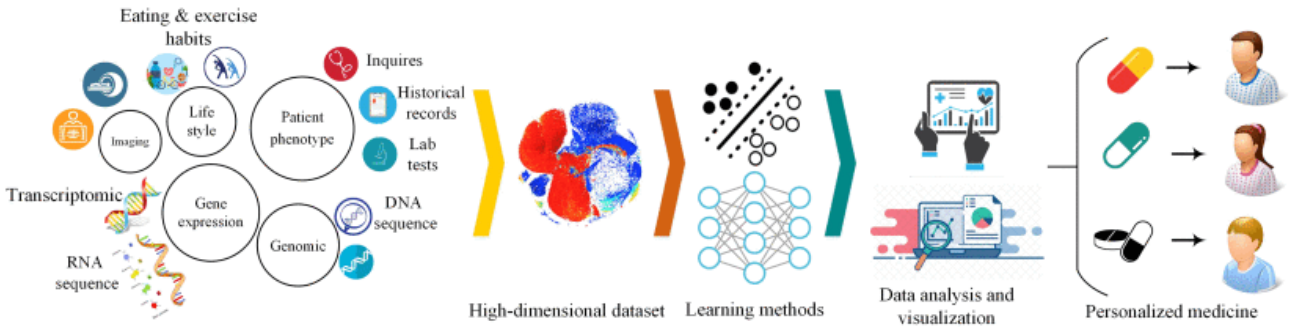


Figure 2.4: Learning process of personalized medicine [98]

2.2.3 Back office teams as users

The back-office works such as management of information system or schedule arrangement are important parts of health care sector. These types of works are also benefited from machine learning. An example is the application of machine learning in management of electrical health records (EHR) which may include a range of data, including demographics, medical history, medication and allergies, immunization status, laboratory test results, radiology images, vital signs, personal statistics like age and weight, and billing information. Clearly, EHR helps increase the quality care not only by helping clinicians identify and stratify chronically ill patients but also by using the data and analytics to prevent hospitalizations among high-risk patients. Therefore, EHR system must be reliable in term of quality and consistent of the data that is not always the case because the inputs of the system come from different sources as well as different users. One example of application of machine learning to improve EHR quality as well as reduce the labor working time is an **automated International Classification of Diseases (ICD) coding**. More specifically, a hierarchical deep learning model with attention mechanism which can automatically assign ICD diagnostic codes given written diagnosis was proposed [76]. Another example of application of machine learning in this sub-category is in **patient scheduling**. In this application, the authors applied classification machine learning algorithms to optimize scheduling after identifying no-shows based on the many sources of the data such as EHR, weather condition as well as driving time [82].

2.2.4 Health authorities as users

Health authorities or managements are the bodies who are responsible for identifying population health needs; planning appropriate programs and services; ensuring programs and services are properly funded and managed; and meeting performance objectives. Machine learning has been approached in building decision support systems that support health authorities in issuing new policies or in making right decisions. For example, machine learning is applied in optimizing hospital processes such as resource allocation and patient flow. More specifically, by early and accurate prediction of patients outcomes, we can better predict demand and allocate scarce hospital resources such as beds and operating rooms. For example, time-series machine learning methods are used to **forecast hospital discharge volume** [53] or recurrent neural network and simulation approach is applied in **emergency department capacity planning** [58]. Another example is that network clustering methods are applied to **detect hospital communities** [20] for more effectively sharing medical records between the hospitals so that providing the effective treatments to the patients (figure 2.6).

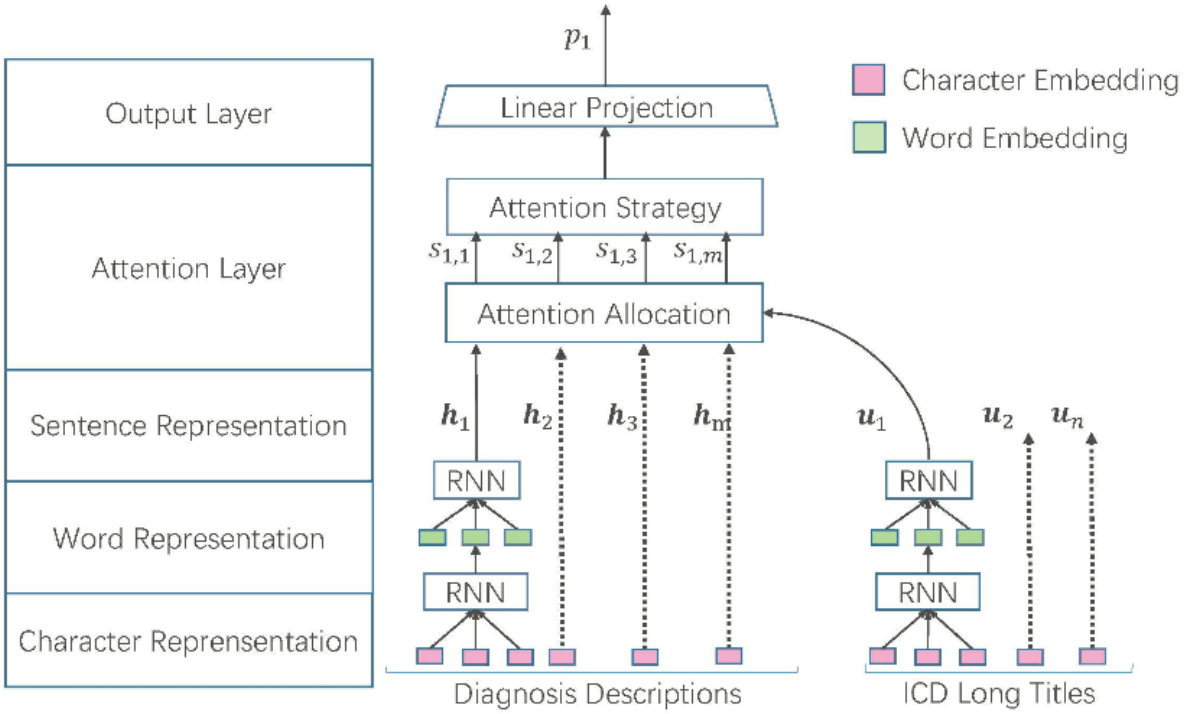


Figure 2.5: Model architecture for automated International Classification of Diseases (ICD) coding[76]

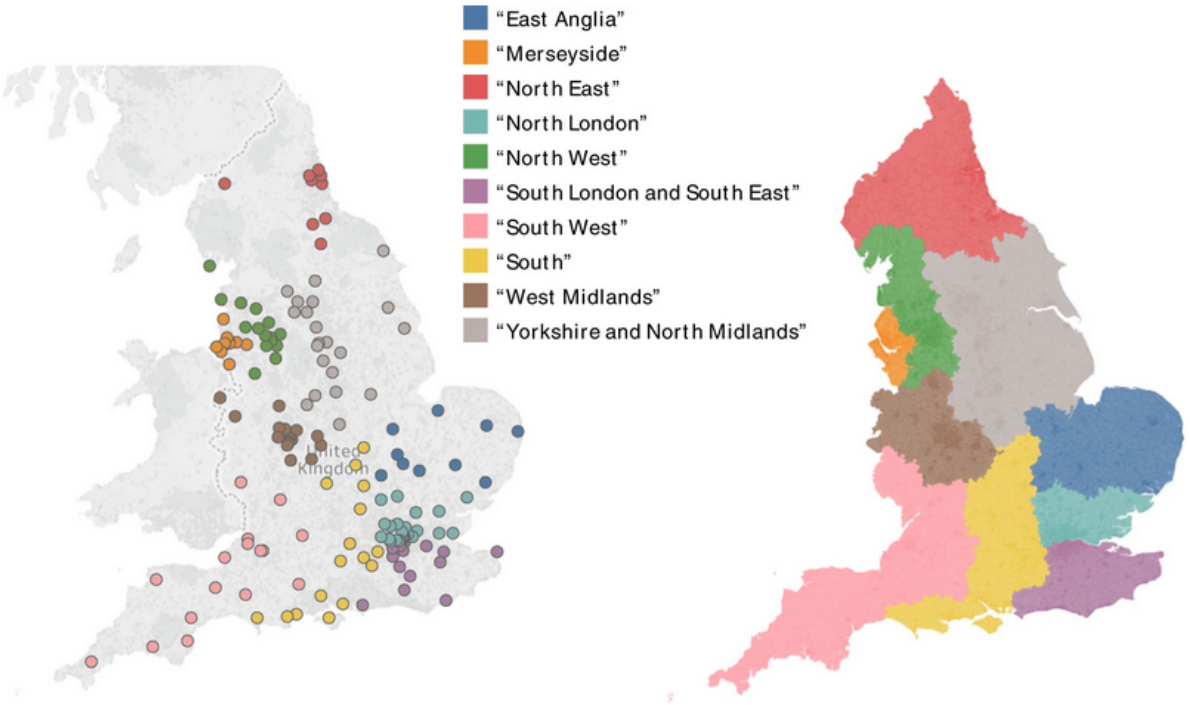


Figure 2.6: Spatial representation of data-sharing communities of hospitals which are presented as round points [20]

2.2.5 Conclusions

In this section, we have presented some current applications of machine learning in health care sector. Although we cannot cover all the applications in this report, it has been proved that machine learning has been applying widely in health care sector. The purpose of these applications are to improve the services of health providers and therefore improve of population health. That is also the purpose of our works which focuses on potentially avoidable hospitalizations. In the next section, we present about the previous studies on potentially avoidable hospitalizations.

2.3 Studies on potentially avoidable hospitalizations

As mentioned in the introduction section, our main works focus on the solutions related to potentially avoidable hospitalizations (PAHs). In particular, like other studies which are not to justify these hospitalizations at the time they are involved, the studies are rather than how to avoid so that reduce the number of these hospitalizations. Before doing our research works, we review the recent studies on PAHs that are presented in this section.

2.3.1 Technical definitions of potentially avoidable hospitalizations

Before starting our review, we need to understand which medical conditions to include as PAHs. Actually, there are many definitions of PAHs that have been proposed. In this report, we present two definitions that are used the studies in French context [79]. The first one is based on Weissman and colleagues [95] method that defines a list of the 12 categories of PAHs (age $\geq$ 20 years) which are associated with the ICD-10 codes as shown in table 2.1.

On the other hand, the AHRQ organization proposed to distinguish the hospital admissions for pathologies sensitive to the first treatment into two lists [79]:

- A list of management of acute pathologies
- A list of management of chronic pathologies

Among the two lists above, the list of management of chronic pathologies shows that more effective management of patients reduces the risk of hospitalization and therefore are chosen to define the medical conditions to include as PAHs [79]. In particular, this list consists of 6 categories can be

Table 2.1: List of the 12 categories of PAHs (age $\geq$ 20 years) by Weissman and colleagues [79, 95]

Pathology	**ICD-10 codes**
Bacterial lung diseases	J13 J14 J15 J16818
Congestive heart failure	I50
Skin / soft tissue infection	J340 K122 L02 L03 L88
Asthma	J45
Hypokalemia	E876
Pathologies to vaccination	A35 A36 A37 A80 B05 B26
Gangrene	I702 I730 R02
Complicated gastroduodenal ulcer	K250 K251 K252 K254 K255 K256 K260 K261 K262 K264 K265 K266 K270 K271 K272 K274 K275 K276 K280 K281 K282 K284 K285 K286
Pyelonephritis and other kidney problems	N10 N11 N12 N136 N158 N159 N172
Acute complications of diabetes	E100 E101 E110 E111 E130 E131 E140 E141
Complicated appendicitis	K352 K353
Hypertension	I10 I11 I12 I13 I15 I674

extracted from the hospital discharge database based on the ICD-10 codes of the principal and related diagnoses (Table 2.2).

In French context, two definitions are highly correlated. In particular, the Pearson correlation values for the datasets exported for the year of 2014 are 0,864 and 0,877 at the levels of departments (Figure 2.7) and geographic PMSI respectively [79].

2.3.2 Studies on potentially avoidable hospitalizations in France

In France, the works related to PAHs are still limited although under development. The first main study carried out at the national level were initiated in the 2000s. This study was included in the 2007 report for the Ministry of Health [70] as the basis of a pilot study on the prevention of hospitalization using PMSI data. This report concluded that high levels of PAHs were positively correlated with age, males, and negatively with the number of medicine, surgery and acute beds, at the density of general practitioners and sector specialists. In addition, the work was about a

Table 2.2: List of chronic pathologies by AHRQ [79]

Pathology	**Inclusion criteria on the principal diagnoses (PD)**	**Exclusion criteria on associated diagnoses (AD)**
Asthma in adults (Age $\geq$ 18)	PD = J45 J46 OR PD = J96.0 if AD = J45	Pregnancy, childbirth and post-childbirth (O00-O99) Heart failure (I09.9 I11.0 I13.0 I13.2 I50) Cystic fibrosis (E84.0-E84.9 Q25.1-Q25.4 Q30 Q31 Q32 Q33 Q34 Q39 Q89.3 P26) Mental disorders (F10-F19 F20 F21 F22 F23 F24 F25 F29 F30 F31 F32 F33 F34 F38 F39 F40-F45 F44 F48 F50-F52 F54 F60 F63 F68 F28 F53 F55 F59 F61 F62 F69) Respiratory diseases (J47 J84.10 J98 J99) COPD (J42 J43 J44 J47 J41.1 J41.8)
Congestive heart failure (Age $\geq$ 40)	PD = I09.9 I11.0 I13.0 I13.2 I50	Pregnancy, childbirth and post-childbirth (O00-O99) COPD (J42 J43 J44 J47 J41.1 J41.8) Ischemic heart disease (I20 I21 I22 I24.0 I24.) Kidney failure (I12 I13.1 N17 N18 N19)
Chronic obstructive pulmonary disease (COPD) (Age $\geq$ 18)	PD = J42 J43 J44 J47 J41.1 J41.8 OR PD = J20 if AD = J42 J43 J44 J47 J41.1 J41.8 OR PD = J40 if AD = J42 J43 J44 J47 J41.1 J41.8 OR PD = J96.0 if AD = J42 J44.9 J47 OR PD = J96.9 if AD = J42 J44.9 J47	Pregnancy, childbirth and post-childbirth (O00-O99) Heart failure (I09.9 I11.0 I13.0 I13.2 I50) Cystic Fibrosis (E84.0-E84.9 Q25.1-Q25.4 Q30 Q31 Q32 Q33 Q34 Q39 Q89.3 P26) Mental disorders (F10-F19 F20 F21 F22 F23 F24 F25 F29 F30 F31 F32 F33 F34 F38 F39 F40-F45 F44 F48 F50-F52 F54 F60 F63 F68 F28 F53 F55 F59 F61 F62 F69)
Dehydration in elderly people (Age $\geq$ 65)	PD = E86 E87.0 E87.1	
Diabetes short-term complication (Age $\geq$ 40)	PD = E10.0 E10.1 E11.0 E11.1 E13.0 E13.1	Pregnancy, childbirth and post-childbirth (O00-O99) Mental disorders (F10-F19 F20 F21 F22 F23 F24 F25 F29 F30 F31 F32 F33 F34 F38 F39 F40-F45 F44 F48 F50-F52 F54 F60 F63 F68 F28 F53 F55 F59 F61 F62 F69)
Angina without procedure, urgent admission) (Age $\geq$ 40)	PD = I20.0 I24.0 I24.8 I20.8 I20.1 I20.9	Pregnancy, childbirth and post-childbirth (O00-O99)

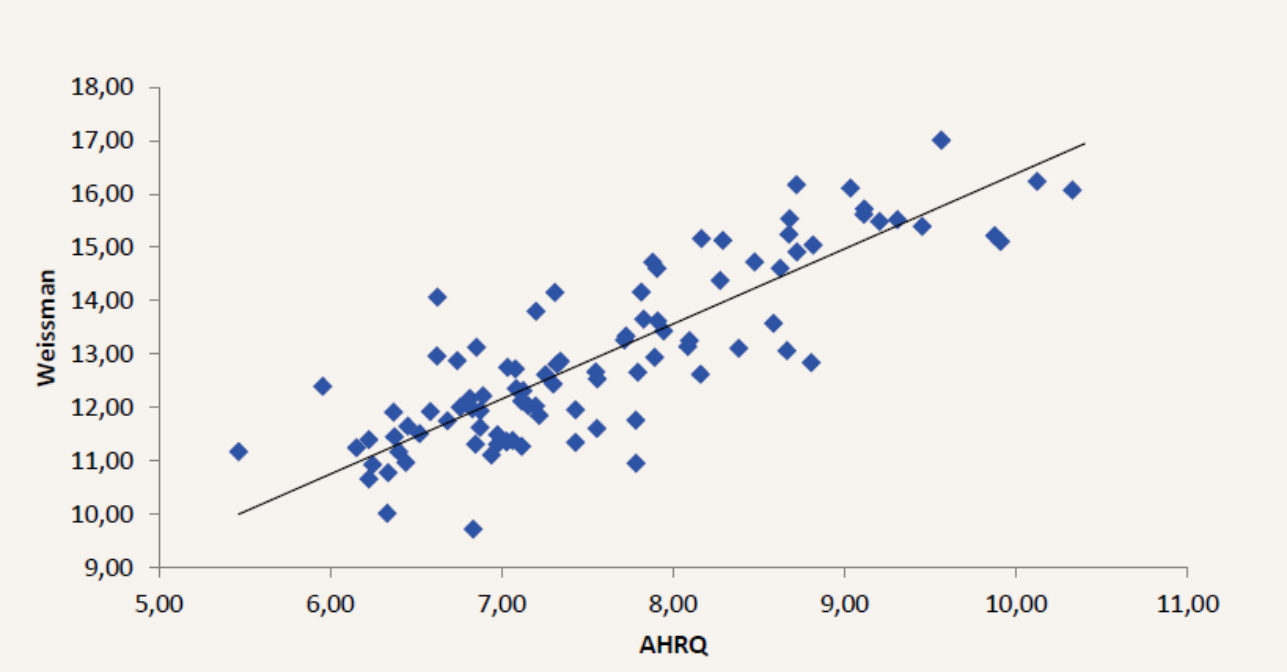


Figure 2.7: Correlation at department level between PAH rates defined by Weissman and AHRQ in France in 2014[79]

methodology for locating PAHs. In particular, they proposed the method that defines a PAH as a hospitalization presenting, in principal diagnosis, a code ICD-10 corresponding to a pathology from the list developed by Weissman and colleagues. (Table 2.1). The main limit of the analysis was the geographic high level (department). This limit did not allow identification significant variations from one territory to another. This work therefore notably underlined the need for an analysis to be carried out with a finer geographic level.

Another nationwide study that was conducted by our colleagues at the economic evaluation unit in 2015 [54]. This study was based on the 2012 PMSI data and the PAHs were identified by a modified Weissman approach. Regarding to geographic level, in the study, department and ZIP code levels were included in a multilevel mixed model. On the other hand, in this study, the data of wide ranges of potential determinants for the variation in PAH rates were included. The data of potential determinants consists of (1) data of health care supply such as density of acute care hospital beds, density of general practitioners, ambulatory care specialist physicians, and ambulatory care nurses; (2) socio-economic data such as median household income, the education level, the proportion of recipients of Couverture Maladie Universelle Complémentaire (CMU-C); (3) as well as epidemiological data such as mortality rate which was used as a proxy for health status. In this study, the multilevel mixed model showed that age-sex standardization rate of PAHs positively associated with the standard mortality ratio but negatively associated with the density of acute care beds and ambulatory care nurses, median pretax income, and education levels. In other words, the PAH rate is higher in areas with high mortality, low income, and low levels of education. The PAH rate is also associated with a shortage of ambulatory care nurses

and a low density of acute care beds.

Our continue study on PAHs was conducted at the time I worked at the economic evaluation unit as an intern for my Master 2 degree from March to August 2017. The result of this work was published in an international conference of Computer Science (SOFSEM 2018) [63]. In this work, we extended a method called gradual patterns that aim at automatically extracting co-variations between variables of data sets in the form of "*the more/the less*" such as "*the more experience, the higher salary*". In particular, the gradual patterns was extended on spatial data to extract co-variations between PAH rates and its potential determinants. With this new approach, we are not only able to find the associations between the increase of PAH rates with its determinants, but also are able to identify how the geographical areas follow or not the tendencies. Particularly, our work is twofold. Firstly, we propose a methodology for extracting gradual patterns at several hierarchical levels. In addition, we introduce a methodology for visualizing this knowledge. For this purpose, we rely on spatial maps for allowing decision makers to easily notice how the areas follow or not the gradual patterns. As an example result, the spatial maps were used to visualize how each geographic PMSI code follows the pattern of smaller nurse density, higher PAH rates (Figure 2.8). Moreover, at higher geographic level (department in this case), we were also able to show aggregation values. For example, figure 2.9 shows the percentages of geographic PMSI codes inside departments following the pattern of smaller nurse density, higher PAH rates or figure 2.10 shows the most influence pattern at each department.

2.3.3 Conclusions

In this section, we have briefly introduced about the previous studies as well as the technical definitions of PAHs used in the studies in French context. In the next section, we introduce about the data standardization methods that are applied in measurement of health status as the way to avoid bias.

2.4 Age-sex standardization in health status measurement

Unlike the standardization (or normalization) like min-max or z-score, in measurement of health status of geographic areas such as mortality or morbidity rates, to avoid bias, the structure of the population should be taken into account. This structure are the age and the sex of the population. For example, in our work, as mentioned in the previous section, we use age-sex standardization of rates for PAHs. There are two methods for calculating standardization rates, namely direct and

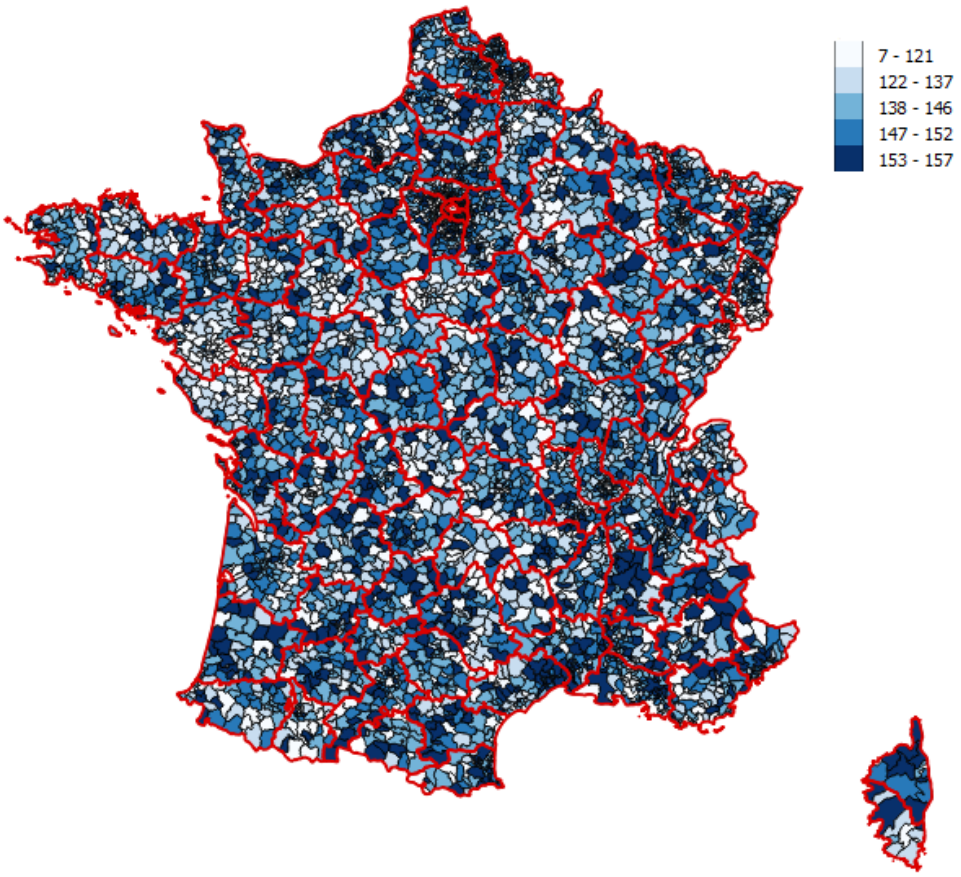


Figure 2.8: Graduated map for item support values of pattern of smaller nurse density, higher PAH rates at geographic PMIS code level [63]

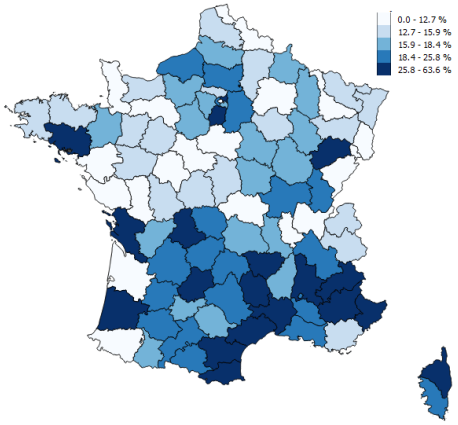


Figure 2.9: Percentage of geographic PMSI codes inside departments following the pattern of smaller nurse density, higher PAH rates [63]

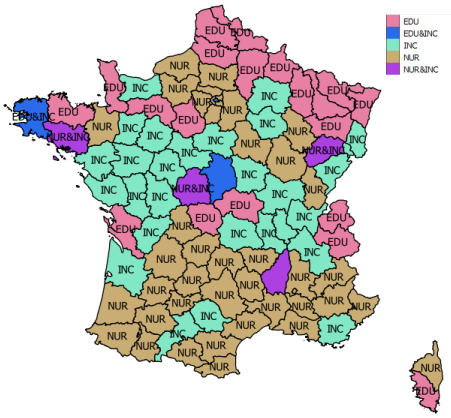


Figure 2.10: Best gradual patterns at each department level [63]. **EDU**: pattern of education and PAH; **INC**: pattern of income and PAH; **NUR**: pattern of nurse and PAH;

indirect standardization. In this section, we introduce how to apply these methods.

Table 2.3: Example reference data for direct age-sex standardization

Age	Sex	Population	Reference rate
0-64	F	2,262,714	0.4021
0-64	M	2,237,209	0.3976
65-74	F	281,155	0.0500
65-74	M	249,725	0.0444
75-79	F	124,524	0.0221
75-79	M	98,292	0.0175
80-84	F	112,044	0.0199
80-84	M	75,156	0.0134
85-89	F	81,769	0.0145
85-89	M	42,861	0.0076
>=90	F	44,775	0.0080
>=90	M	16,563	0.0029
TOTAL		5,626,787	1

2.4.1 Direct age-sex standardization

To demonstrate how direct age-sex standardization is computed, we use the example data of two geographic areas (Areas A and B) in which we have the data of number of PAH (column *No*) as well as the number of inhabitant (column *Pop*) at each specific range of age and sex (Table 2.4). The age and sex of the populations are taken into account in direct age-sex standardization by referring to the structure of the whole regions or country where we conduct the analysis. For example, in our work at Occitanie France region, the age and sex structure data of the region is presented at table 2.3 in which the values at column *Reference rate* (column *Ref. rate* in table 2.4) are computed by:

$$Reference\ rate = \frac{Population}{TOTAL}$$

Also, in table 2.4, at each area, the *raw rate* of PAH per 1,000 inhabitants corresponding to the range of age and sex is computed by:

$$Raw\ rate = \frac{No * 1,000}{Pop}$$

Where **No** and **Pop** are the number of PAHs and the size of population corresponding to the

Table 2.4: Example data for direct age-sex standardization

			Area A				**Area B**			
Age	Sex	Ref. rate	No	Pop	Raw rate	St. rate	No	Pop	Raw rate	St. rate
0-64	F	0.4021	9	14,515	0.62	0.25	11	10,452	1.05	0.42
0-64	M	0.3976	16	14,278	1.12	0.45	12	10,245	1.17	0.47
65-74	F	0.0500	11	1,221	9.01	0.45	2	1,163	1.72	0.09
65-74	M	0.0444	11	1,005	10.95	0.49	10	883	11.33	0.50
75-79	F	0.0221	3	469	6.40	0.14	4	612	6.54	0.14
75-79	M	0.0175	9	391	23.02	0.40	8	391	20.46	0.36
80-84	F	0.0199	9	351	25.64	0.51	15	596	25.17	0.50
80-84	M	0.0134	7	229	30.57	0.41	14	326	42.94	0.57
85-89	F	0.0145	5	235	21.28	0.31	12	539	22.26	0.32
85-89	M	0.0076	7	124	56.45	0.43	5	237	21.10	0.16
>=90	F	0.0080	15	145	103.45	0.82	7	299	23.41	0.19
>=90	M	0.0029	5	44	113.64	0.33	9	105	85.71	0.25
Total		1	107	33,007		4.99	109	25,848		3.98

range of age and sex.

Now, we take into account the population structure of the whole regions (column *Ref. rate*) in computing the standardization rate (column *St. rate*) of PAHs per 1,000 inhabitants corresponding to the range of age and sex.

$$St.\ rate = (Raw\ rate) * (Ref.\ rate)$$

By summarizing the standardization rates of the whole area, we obtain the direct age-sex standardization of that area. In table 2.4, these values are 4.99 and 3.98 for area A and area B respectively. Therefore, if we compare the area A and area B, then the rate at area A is higher than the rate at area B

On the other side, in case we do not take into account the age and sex of the PAH patients, the raw PAH rates per 1,000 inhabitant for area A and B will be:

$$raw\_rate\_A = \frac{107 * 1,000}{33,007} = 3.24$$

$$raw\_rate\_B = \frac{109 * 1,000}{25,848} = 4.22$$

The rates above indicate that if we do not take into account the structure of the populations, the rate of area B which is 4.22 is higher than the rate of area A which is 3.24. This conclusion is the opposite of the previous conclusion when we compared the standardization rates.

In conclusion, the example above demonstrates the way the structure of population data are used in computing the direct age-sex standardization. It also shows that when we measure of health status between geographic areas, it is often bias if we just use the raw rates instead of age-sex standardization.

2.4.2 Indirect age-sex standardization

In the previous, we have presented the use of the direct age-sex standardization in order to avoid bias in measure health status between geographic areas. However, medical information is strictly confidential. On the other words, to avoid people can predict who the patients are, normally, the information of age and sex are not provided on the data exported for small geographic areas. Therefore, sometimes we cannot apply age-sex standardization directly. In those cases, indirect age-sex standardization are used instead. In definition, the indirect age-sex standardization values are computed by the ration between the observed numbers over the expected numbers. To demonstrate how to compute indirect age-sex standardization, given that at the entire region (or country), we know the number of PAHs (column *No PAHs*) corresponding to each range of age and sex (Table 2.5) in which the values at column *Reference rate* (column *Ref. rate* in table 2.6) are computed by:

$$Reference\ rate = \frac{(No\ PAHs) * 1,000}{Population}$$

Back to the example of area A and area B above. We know the structure of the populations of these areas (Column *Pop* in table 2.6), but we do not know the number of PAHs corresponding to each range of age ans sex. Based on the reference rate at the region level, we can compute the expected number of PAHs corresponding to each range of age ans sex at each area:

$$Expeced\ No = \frac{(Ref.\ rate) * Pop}{1,000}$$

By summarizing the expected numbers, we have the expected numbers of PAHs for each geographic area. In the example above, the expected number of PAHs of area A and area B are 113.60 and 145.16 respectively. On the other side, we also know the total number of PAHs (known as the

Table 2.5: Example reference data for indirect age-sex standardization

Age	Sex	No PAHs	Population	Reference rate
0-64	F	1,852	2,052,788	0.90
0-64	M	3,060	2,133,039	1.43
65-74	F	1,666	246,605	6.76
65-74	M	2,915	234,976	12.41
75-79	F	1,452	104,882	13.84
75-79	M	2,020	89,397	22.60
80-84	F	2,347	103,721	22.63
80-84	M	2,454	70,176	34.97
85-89	F	2,765	77,369	35.74
85-89	M	2,106	39,806	52.91
>=90	F	2,630	41,786	62.94
>=90	M	1,245	13,898	89.58
Total		26,512	5,208,443	

Table 2.6: Example data for direct age-sex standardization

			Area A		**Area B**	
Age	Sex	Ref. rate	Pop	Expected No	Pop	Expected No
0-64	F	0.90	14,515	13.10	10,452	9.43
0-64	M	1.43	14,278	20.48	10,245	14.70
65-74	F	6.76	1,221	8.25	1,163	7.86
65-74	M	12.41	1,005	12.47	883	10.95
75-79	F	13.84	469	6.49	612	8.47
75-79	M	22.60	391	8.83	391	8.83
80-84	F	22.63	351	7.94	596	13.49
80-84	M	34.97	229	8.01	326	11.40
85-89	F	35.74	235	8.40	539	19.26
85-89	M	52.91	124	6.56	237	12.54
>=90	F	62.94	145	9.13	299	18.82
>=90	M	89.58	44	3.94	105	9.41
Total				113.60		145.16

observed numbers) in these areas which are 107 in area A and 109 in area B. The indirect age-sex standardization is computed by the ratio between the observed numbers and the expected numbers.

$$indirect\_rate\_A = \frac{observed\ number}{expected\ number} = \frac{107}{113.60} = 0.94$$

$$indirect\_rate\_B = \frac{observed\ number}{expected\ number} = \frac{109}{145.16} = 0.75$$

Comparing with the direct age-sex standardization method above, the indirect age-sex standardization method also returns the result that the rate of area A is higher than the rate of area B.

2.4.3 Conclusions

In this section, we have introduced the direct and undirect age-sex standardization methods that are used in measurement of health status of geographic areas as the way to avoid bias. In the next section, we introduce about basic spatial analysis that helped us explore our spatial datasets.

2.5 Basic spatial analysis

It is often estimated that over 80% of data integrates spatial information [30]. Such spatial information are currently taking more and more importance with the emergence of Internet of Things (IoT) and popular applications integrating spatial information (e.g., Google maps). On the other hand, as the first Law of Geography, "everything is related to everything else, but near things are more related than distant things." [87]. As an example, the temperatures across the globe is visualized in the map below (figure 2.11) in which blue color shows colder temperatures in blue and red color shows warmer temperatures.

It is clear that the temperatures at two close locations are quite the same. Generally, the temperatures in example are called feature values and the close locations are called neighbors. The feature values at a location and at its neighbors can be systematically high-high, low-low, low-high, and high-low or randomly appear at those locations. How these feature values are geographically

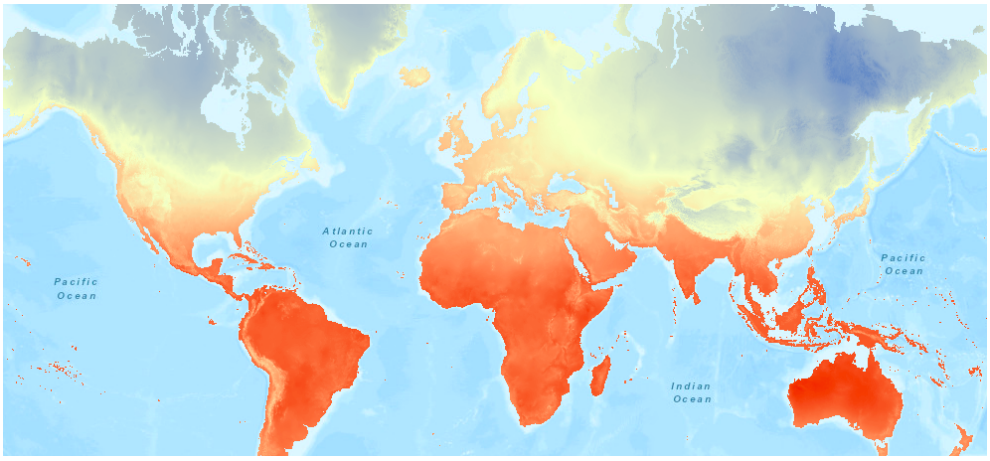


Figure 2.11: Example of first Law of Geography: global temperature [85]

related is interpreted by spatial autocorrelation. As shown in figure 2.12, positive spatial autocorrelation tells that the feature values at a neighborhood tend to be similar (high-high or low-low) while negative spatial autocorrelation indicates that the feature values tend to be different (low-high or high-low). On the other hand, no spatial autocorrelation tends to indicate that feature values are associated with the locations randomly. In this section, we will present about how to measure the spatial autocorrelation as well as a method to detect spatial clusters and spatial outliers. These methods are mainly based on the lectures conducted by Professor Luc Anselin [4]. We also briefly introduce the ways that the spatial data is included in linear regression models.

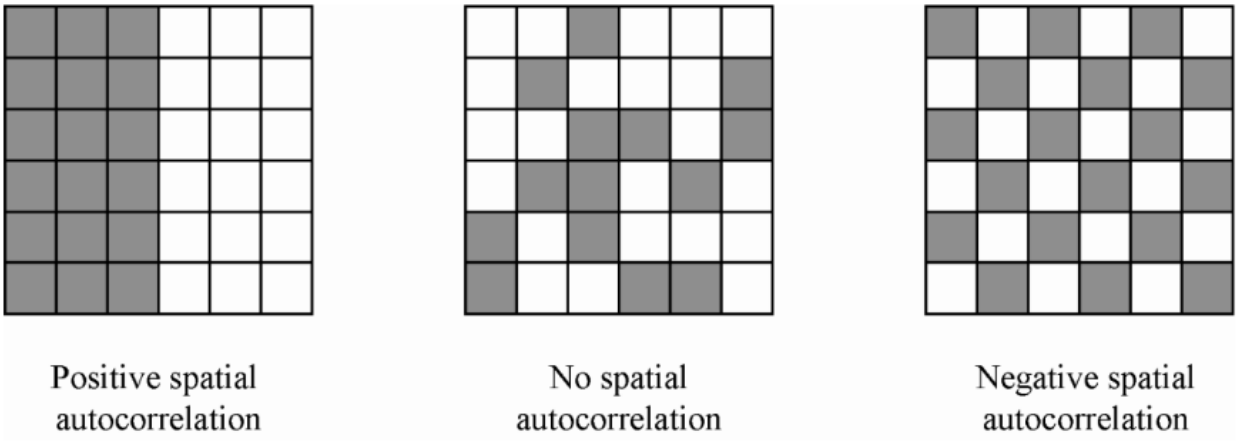


Figure 2.12: Example of spatial autocorrelation [85]

2.5.1 Measure of spatial autocorrelation

In the previous section, we mentioned that we can use spatial autocorrelation to interpret how the feature values are geographically related. In the literature there are two most popular indexes to measure spatial autocorrelation. These indexes are Moran's I and Geary's C. While Moran's I

index measures how the feature value at a location is similar to the feature values of its neighbors, Geary's C index focuses on the dissimilarity between the neighborhood. Mathematically, Moran's I index (denoted by I) is computed with formula 2.1 and Geary's C index (denoted by C) is computed with formula 2.2

$$I = \frac{N}{W} \frac{\sum_i \sum_j w_{ij}(x_i - \bar{x})(x_j - \bar{x})}{\sum_i (x_i - \bar{x})^2} \tag{2.1}$$

$$C = \frac{(N-1) \sum_i \sum_j w_{ij}(x_i - x_j)^2}{2W \sum_i (x_i - \bar{x})^2} \tag{2.2}$$

In both formulas 2.1 and 2.2 above:

- N is the number of spatial units indexed by i and j
- x is the variable of feature such as temperature in the example above.
- $\bar{x}$ is the mean value of x.
- w_{ij} is spatial weight between spatial unit i and spatial unit j.
- W is the sum all w_{ij}

Both formulas above are straight forward as long as we can define the spatial weights matrices (or w_{ij}) between the spatial units.

2.5.2 Spatial weight matrices

In the previous section, we mentioned about using spatial weight between spatial unit i and spatial unit j to reflect the "spatial influence" between unit i and unit j. There are actually several approaches to define this weight matrix.

2.5.2.1 Spatial weight matrices based on boundaries

A simplest way to define spatial weight matrices is based on boundaries. More specifically, if unit i and unit j have common boundaries then w_{ij} (also w_{ji}) has value 1. Mathematically, if we denote the set of boundary points of unit i by $bnd(i)$ then,

$$w_{ij} = \begin{cases} 1 & if \;\; bnd(i) \cap bnd(j) \neq \emptyset \\ 0 & if \;\; bnd(i) \cap bnd(j) = \emptyset \end{cases}$$

Now, if we denote l_{ij} as the number of points in $bnd(i) \cap bnd(j)$, then the formula becomes:

$$w_{ij} = \begin{cases} 1 & if \;\; l_{ij} > 0 \\ 0 & if \;\; l_{ij} = 0 \end{cases}$$

The matrix defined by formula above is called *queen contiguity weight* matrix. In this matrix, $l_{ij} = 1$ also give 1 for w_{ij}. Another approach called *rook contiguity weights* requires $l_{ij} > 1$ instead of $l_{ij} > 0$ for w_{ij} having value of 1. In practice, depending how the spatial units are defined such as using grid networks, the two corresponding matrices might be significantly different as shown in figure 2.13.

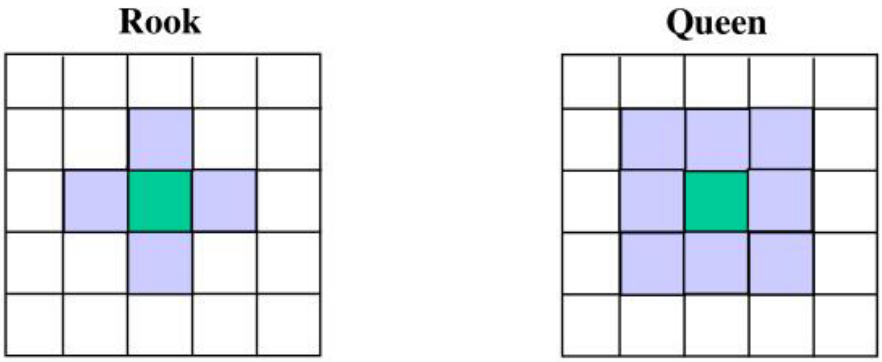


Figure 2.13: Rook contiguity weights vs queen contiguity weights

2.5.2.2 Spatial weight matrices based on distances

Another approach for spatial weight matrices is based on the distances between the spatial units. For example, the w_{ij} has value of 1 if the distance between unit i and unit j is less than 30 km (like figure 2.14). Generallly, if we denote d_{ij} is the distance between unit i and unit j, and τ is the threshold, then we can formula for w_{ij}:

$$w_{ij} = \begin{cases} 1 & if \;\; d_{ij} <= \tau \\ 0 & if \;\; d_{ij} > \tau \end{cases}$$

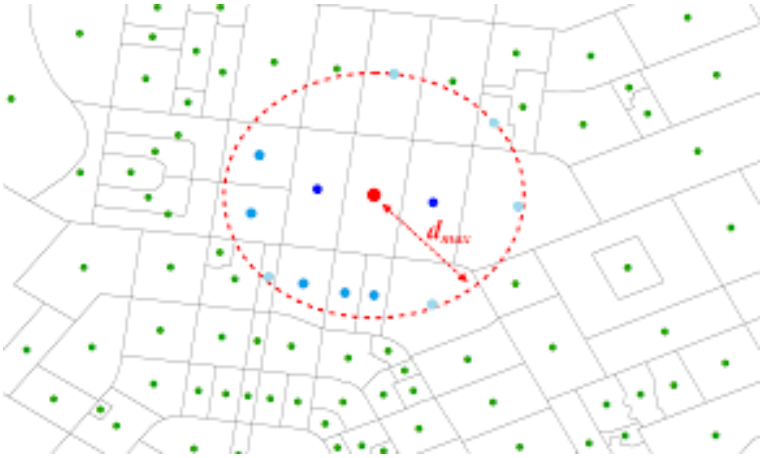


Figure 2.14: Example of spatial weight matrices based on distances

On the other side, in some applications in practice, the size of the spatial units are different. For example, if we use communes as the spatial units, the communes in the big cities are small in term of spatial size while the communes in remote areas are big. Therefore, the approach of using thresholds as we just mentioned could lead to a problem that some spatial units have many neighbors while some others have very few or even no neighbors. A solution to that problem is to apply K-nearest neighbors. More specifically, for each spatial unit i, we select K spatial units whose distances to unit i are smallest. Mathematically, if we denote $N_K(i)$ is the set of these K nearest units of unit i, then:

$$w_{ij} = \begin{cases} 1 & if \;\; j \in N_K(i) \\ 0 & if \;\; j \notin N_K(i) \end{cases}$$

However, what does the distance d_{ij} mean? The first option is the space distance. For example, if the space units are the points, the space distance between two points is the length of the straight line connecting the two points. In the case that the spatial units are the polygons, we can consider to use the space distances between the corresponding centroids. The other option, in some applications, instead of the distance of the straight line connecting point i and point j, the distances can be the lengths of the road to go from point i to point j is used. Similarity, the travel time between i and j might be the one to be used. As a conclusion, depending on the application, we need to define the relevant way to measure the distance to be used in spatial weight matrices.

2.5.2.3 w_{ij} values

In the previous sections, the spatial weight w_{ij} receives the values that are either 1 or 0. Those values do not reflect well the distance between the spatial units. On the other words, spatial

weights w_{ij} should link to the distance values d_{ij}. There are several things we need to consider while including d_{ij} in w_{ij}. The first thing is the unit measuring the distance. For example, kilometre and metre should return the different results. The second thing is how the distance should be presented in w_{ij}. In any presentation of w_{ij}, there is a rule that the smaller distance returns the bigger w_{ij} than the larger distance. As examples, also in practice, there are several ways computing w_{ij} such as the followings:

$$w_{ij} = \frac{1}{d_{ij}^{\alpha}}$$

Or

$$w_{ij} = exp(-\alpha d_{ij})$$

Where α is any positive scalar, but typically $\alpha = 1$ or $\alpha = 2$ [9].

Moreover, to remove dependence on extraneous scale factors, it is necessary to normalize of these spatial weights. For example, row normalized weights can be applied that the new spatial weights u_{ij} is computed from w_{ij} as below:

$$u_{ij} = \frac{w_{ij}}{\sum_k w_{ik}}$$

2.5.3 Global spatial autocorrelation

Once we define the spatial weight matrices, we can compute the Moran's I or Geary's C indexes by following the formula 2.1 or formula 2.2 respectively. To simplify our report, we now focus on Moran's I index which is more popular. It can be mathematically proved that Moran's I index has values ranging from -1 to 1. However, how do these indexes imply? Actually, these indexes do not directly tell whether or not there is a global spatial autocorrelation. However, if there is a global spatial autocorrelation, these indexes can tell it is a positive or negative spatial autocorrelation. In particular, for Moran's I index, the negative values indicate negative spatial autocorrelation while the positive values indicate positive spatial autocorrelation. However, still, is there a global spatial autocorrelation? We can answer this question through statistical hypothesis tests using p-value. More particular, we compute the p-value by generating the samples for Moran's I indexes. A sample Moran's I index is generated by randomly re-placing all the feature values on all the

spatial units. After K times we randomly re-place the all the feature values on all the spatial units, we have a data set of K samples of the Moran's I index.

If the generation of the dataset of K samples of the Moran's I index is considered as step 1 (also called **permutation** step), then step 2 is to standardize this dataset with z-score standardization:

$$z_k = \frac{I_k - \mu}{\sigma}$$

in which, I_k is value of item k, μ is the mean value and σ is the standard deviation of the dataset.

As the final step, we convert the Moran's I index to the corresponding z value, and from that we can compute p-value for this Moran's I index. If the p-value is sufficiently small, for example smaller than 0.05, then there is technically a spatial autocorrelation.

2.5.4 Spatial clusters/outliers detection

One most applications of spatial analysis is to detect clusters or outliers. We often use spatial maps such as choropleth maps to visualize the data of feature values. By looking at these maps, we can somewhat detect spatial clusters or outliers through the colors presenting feature values. For example, on the example map (figure 2.15), it seems that there is a cluster around MS and AL states. However, the feature values are associated with these states systematically or randomly? On the other words, we need technical ways to measure and then detect the spatial clusters or outliers.

One approach is based on statistical significance. More specifically, we measure the statistical significance that a feature value is associated with the corresponding spatial unit as the way to confirm that this association is not by random. One approach is based on local Moran's I index which is formulated in the same way as global Moran's I index:

$$I_i = \frac{(N-1)(x_i - \bar{x}) \sum_{j \neq i} w_{ij}(x_j - \bar{x})}{\sum_{j \neq i}(x_j - \bar{x})^2} \tag{2.3}$$

In which,

- I_i is local Moran's I index corresponds to spatial unit i
- N is the number of spatial units indexed by i and j

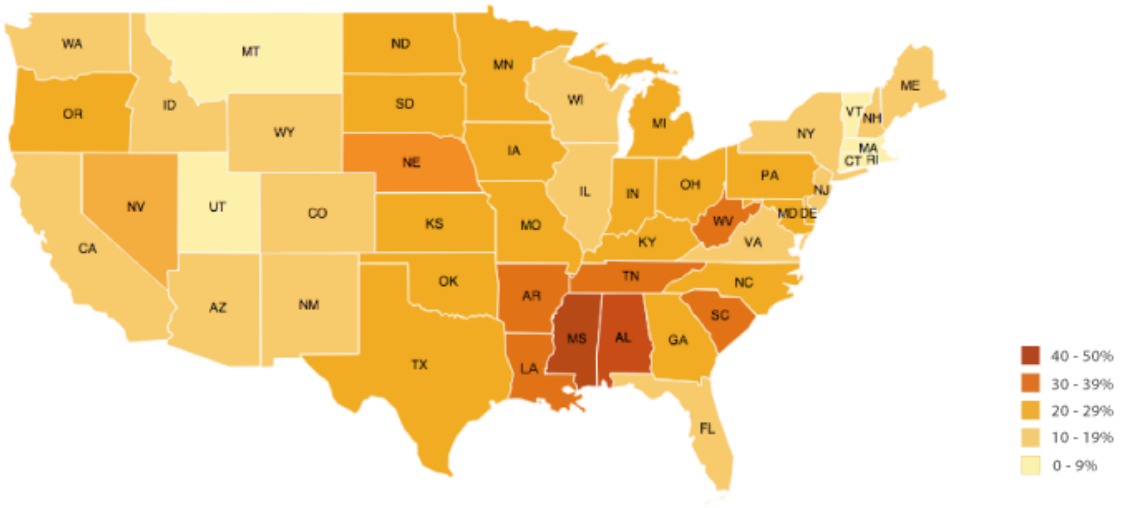


Figure 2.15: Example of choropleth map to visualize feature values [69]

- x is the variable of feature such as temperature in the example above.
- $\bar{x}$ is the mean value of x.
- w_{ij} is spatial weight between spatial unit i and spatial unit j.

Like the way to compute the statistical significance of global spatial autocorrelation above, we run the permutations step to randomly generate a data set of the local Moran's I index for each spatial unit i. However, different to the permutations step in global spatial autocorrelation, to generate a local Moran's I index sample, the spatial unit i is excluded when randomly re-placing the feature values (N-1 values) on the other spatial units (N-1 units as unit i is excluded). After this permutations step, for each each spatial unit i, we have a dataset of local Moran's I index samples.

Once we have the dataset of local Moran's I index samples, we take the same steps as we do while measuring the statistical significance of global spatial autocorrelation. As the results, we can point out the statistical significance of any spatial unit through the p-value. For example, the spatial units whose p-values are smaller than 0.05 technically indicate that they are either clusters (the values are high-high or low-low compared with the neighborhood) or outliers (high-low or low-high compared with the neighborhood). The next question is to find out the clusters are the type of high-high or low-low as well as the outliers are high-low or low-high. The answers lie on the Moran scatter plot.

Moran scatter plot and its application to classify clusters/outliers

Once we can indicate a spatial unit has statistical significance of a cluster or outlier, the the next step is to use Moran scatter plot to identify the spatial unit is:

- A cluster high - higt compared with its neighbors
- A cluster low - low compared with its neighbors
- A outlier high - low compared with its neighbors
- A outlier low - higt compared with its neighbors

Moran scatter plot is a scatter plot that the values of x-axis and y-axis are defined as below:

- x-axis: $z_i = (x_i - \bar{x})$
- y-axis: $\sum_j w_{ij} z_j$

In which, the notations are the same as we have been using.

- x is the variable of feature such as temperature.
- $\bar{x}$ is the mean value of x.
- w_{ij} is spatial weight between spatial unit i and spatial unit j.

By the definition for the x-axis and y-axis above, all spatial units can be presented on the corresponding Moran scatter plot. As demonstrated in figure 2.16, the two relative mean lines corresponding to x-axis and y-axis, which are close to 0, divide the spatial units into four parts corresponding high-high clusters, low-low clusters, high-low outliers, and low-high outliers.

To sum up, in this section, the method of detecting clusters or outliers based on statistical significance has been introduced. In the next section, we introduce the approaches that spatial data are taken into account while measuring the relationship between response variables and explanatory variables. These approaches are spatial regression models.

2.5.5 Spatial regression models

We often use linear regression approaches to modeling the relationship between a scalar response (or dependent variable) and one or more explanatory variables (or independent variables). Suppose that we have data consisting of set of observations $\{y_i, x_{1i}, x_{2i}, ..., x_{Ki}\}$ in which,

- y_i is the value of item i of dependent variable

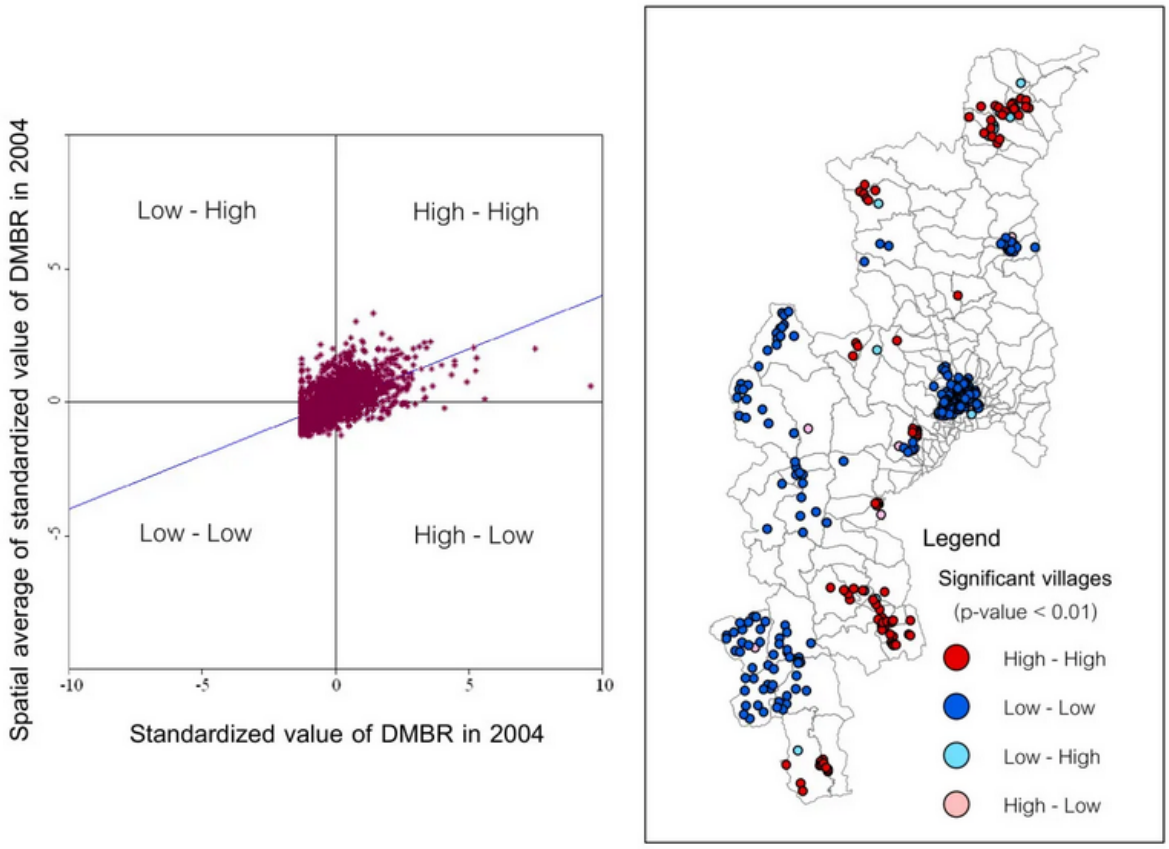


Figure 2.16: Example of Moran scatter plot [18]

- x_{ki} is the value of item i of independent variable k.

In a linear regression model, the response variable, y_i, is a linear function of the independent variables:

$$y_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \cdots + \beta_K x_{Ki} + \varepsilon_i \tag{2.4}$$

In which, ε_i presents the corresponding error.

To shorten the formula above, the vector of $\{1, x_{1i}, x_{2i}, ..., x_{Ki}\}$ is denoted as X_i, the vector of $\{\beta_0, \beta_1, \beta_1, ..., \beta_K\}$ is denoted as β then formula 2.4 becomes

$$y_i = \beta X_i + \varepsilon_i \tag{2.5}$$

Or in general,

$$y = \beta X + \varepsilon \tag{2.6}$$

However, in the formula 2.6, the spatial data are not taken into account. As mentioned in the previous section, once there is a global spatial autocorrelation (section 2.5.3), we should not ignore spatial information. In other words, if the dependent variable y has the global spatial autocorrelation, the spatial information should be included into formula 2.6.

$$y = \rho W y + \beta X + \varepsilon \tag{2.7}$$

In formula 2.7, W is the spatial weight matrix that we mentioned in section 2.5.2. Formula 2.7 is called spatial autoregressive (SAR) model [49].

Similarly, if the independent variables X have the global spatial autocorrelation, the spatial lag of independent variables X or $WX\theta$ can be added in formula 2.6. The new model is called spatial lag X (SLX) model.

$$y = WX\theta + \beta X + \varepsilon \tag{2.8}$$

In addition, there could be global spatial autocorrelation with the errors ε because there could be some explanatory variables that we do not have the corresponding data to be included in the above models. In those cases, it is possible to add the spatial lag for the errors and then so-called spatial error model. Mathematically, 2.6 becomes:

$$y = \beta X + u, \quad u = \lambda W u + \varepsilon \tag{2.9}$$

Moreover, since the spatial lags of dependent variable, independent variables, as well as the errors are independent so that they can be combined into one spatial regression model [16]. In particular, we have:

Spatial Durbin model [49] is a combination of the SAR model and the SLX model:

$$y = \rho W y + WX\theta + \beta X + \varepsilon \tag{2.10}$$

Kelejian-Prucha model or SAC model is a combination of the SAR model and the spatial error model:

$$y = \rho W y + \beta X + u, \quad u = \lambda W u + \varepsilon \tag{2.11}$$

In the cases that all spatial lags are included, we have Manski model:

$$y = \rho W y + WX\theta + \beta X + u, \quad u = \lambda W u + \varepsilon \tag{2.12}$$

In summary, together with the previous section that introduced the method of detecting clusters/outliers, in this section, we have introduced a basic spatial analysis, which are a spatial regression models as the way to measure the relationship between variables. However, there are also some notes that we should pay attentions to. One of them is the modifiable areal unit problem that is presented in the next section.

2.5.6 Modifiable areal unit problem in spatial data analysis

In the section 2.4, we introduced about direct and indirect age-sex standardization as the ways to avoid bias when we work with health data. In this section, we introduce another issue that we should pay attention when we collect geographic data. Specifically, the problems come when we use the data aggregated from a set of geographic area units. To demonstrate the issue, we use an example that there are two teams (team 1 and team 2) conducting geographical analysis of the same region in which there are sick and normal people who are presented by black and white points respectively (figure 2.17). Now, imaging that team 1 divides the region into 4 geographic units (Analysis 1) and team 2 also divides the region into four geographic units but in different way (Analysis 2). As it can be seen at figure 2.17, from the data they collected at the four geographic units, the two teams would have different conclusions. Specifically, team 1 can concludes that the sick rates of all the four geographic units are 50% while with team 2, the sick rates are 100% in two geographic units whereas in the other two units, the rates are 0%. As a conclusion, the way the whole region are divided into geographic units while we collect data have the impacts to the data analysis. As this problem is because of the way the geographic areas are defined, it is called modifiable area unit problem (MAUP).

There are actually two issues related to the MAUP. The example above shows the first issue which is called "zoning effect". This issue is about the different ways we divide the entire region into geographic units, but these units are somewhat equivalent in term of the size of the units. The

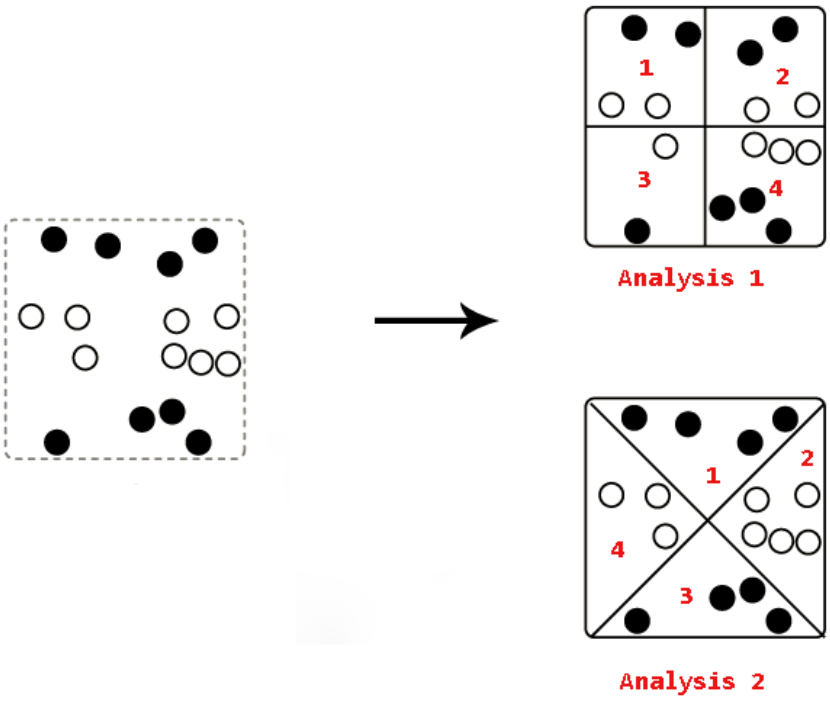


Figure 2.17: Example of modifiable areal unit problem

other issue of MAUP is "scale effect". This second issue is about the impacts of the size of the geographic units in geographic analysis. One example for the "scale effect" issue is that when we conduct the analysis at different levels of administrative borders such as zipcode level and department level, the results from different levels could be different and that is because of "scale effect" issue.

There are several solutions that have been made to deal with MAUP. The first method to counteract this effect is simply make analysis as fine scale as possible. For example, points are measured against other points help to remove or diminish MAUP. The other approach is to check the robustness of results by changing geographic scale levels and compare the results between different levels. If these results are significantly different, then that likely means that the scale needs to be reevaluated or reapplied so that more consistent results are achieved [59]. For example, analysis are conducted at both zipcode and department levels then the results are compared to evaluate the robustness of the analysis. Similarly, Bayesian spatial models and sampling procedures are used to estimate appropriate scales of aggregation by varying the scale and boundaries in which aggregation occurs [11]. This helps to determine where values for autocorrelation among tested variables are most robust or provide the most stable outputs.

In summary, in this section, the MAUP problem in spatial analysis has been introduced. Some solutions to it is also briefly presented. Since we work on spatial data, our approach to deal with

this famous problem is presented in the next chapter.

2.6 Conclusion

In this chapter, we have presented a brief literature review on the applications of machine learning in heath care sector as well as the recent studies on PAHs in France. In addition, we also presented some notes to avoid bias in health care analysis such as direct and in-direct age and sex standardization as well as a brief introduction about basic spatial analysis. In the next chapter, we will present our works of extending one of regression methods for enhancing heath care services to reduce the PAHs.

Chapter 3

Regression methods for enhancing health care service to reduce PAHs

Contents

3.1 Introduction

In section 1.1.3, we briefly introduce about potentially avoidable hospitalizations (PAHs). To remind, PAHs are defined as hospital admissions that could have been prevented [74]. In particular, these hospitalizations are in fact the consequence of the sudden aggravation of a chronic disease (diabetes, heart failure, respiratory failure). These acute episodes could have been prevented with timely and effective treatments and therefore the hospitalizations could have been avoided [12]. As mentioned in section 1.2, every year, in France, there are about 300,000 preventable hospitalizations [79], associated with a cost of several hundred million Euros for the Health Insurance [14]. That means avoiding these hospital admissions not only could enhance quality of live of the patients but also could decrease substantial costs caused by patient treatments [54, 31].

In addition, in section 2.3, we also briefly introduce some previous studies on PAHs and the potential factors that could be associated with high rates of PAHs [63, 54, 33]. Particularly, these recent studies in France have revealed that the higher (age-and-sex-standardized) rates of PAHs are linked to higher mortality rates, lower density of acute care beds and ambulatory care nurses, lower median income, and lower education levels [63, 54]. More specifically, these studies suggested that by increasing the number of nurses at some geographic areas, the number of PAHs in these areas could be reduced [54]. On the other hand, typically in France, the public health decision makers can have influence on the factors related to health care such as the density of physicians, nurses, or the density of hospital beds while socioeconomic determinants such as income and education are not actionable inside the health system sector. Specifically, both the national- and regional-level health authorities are highly interested in enhancing the health care services in order to reduce the number of PAHs.

Moreover, the health system is subject to strong constraints. In particular, they must provide quality care while controlling associated costs and ensuring equality of access to the health care services. The latter states that all patient-citizens must be able to benefit from the care they need, regardless of their geographical and socioeconomic situation. Hence, being able to select geographic areas in order to maximize the impact of an intervention is of high importance. That gives birth to our work which aims at building a decision support system that recommends the optimal actions targeting on the geographic areas while considering the constraints.

In particular, the purpose of our work is to find the geographic areas to increase the nurses for the biggest reduction of PAHs while not only integrating socioeconomic constraints such as the available budgets as well as ensuring the equal access to health care but also considering other potential determinants of PAHs. The geographic areas we mention here are the cross-border living areas (fr. Bassins de vie - BVs) which define the geographic areas in which the inhabitants have access to the most common equipment and services including trade, education, health, etc.

In our approach, for every BV, we compare the predicted rates of PAHs before and after trying to add new nurses. Our idea is that the BVs that return the biggest reduction of these predicted values after trying to increase the number of nurses could be the best ones for the actual nurse implementation. Since the rates of PAHs are the numeric values, so any regression method could be the option for our approach. Therefore, after evaluating all common regression methods, we extended the support vector machine for regression to spatial information so that we can take into account the constraints mentioned above in building the decision support system.

3.1.1 Dataset and pre-processing works

As a way to deal with the modifiable areal unit problem (MAUP), which is introduced in the section 2.5.6, we select "Bassins de vie" or *BVs* as spatial units. *BVs* are the geographic areas that are defined by French National Institute for Statistics and Economic Studies (INSEE). In particular, communes (denoted as *INSEE codes*) are grouped into the same *BVs* if the inhabitants in these communes accessing to the most common equipment and services including trade, education, health, etc. [43].

On the other side, the data of PAHs are extracted from French national hospital discharge database (PMSI, section 1.3) in which for the privacy reason, the patients are geographically coded by *PMSI codes*. These *PMSI codes* are roughly equivalent to French postal codes [6] which mostly (not all) have 1-n relationship with communes or *INSEE codes* [23].

Therefore, the first pre-processing our dataset is to convert the data set from the spatial units of *PMSI codes* to spatial units of *BVs*. This task is based on the relationships between *BVs*, *PMSI codes*, *Postal codes*, and *INSEE codes* which are presented in figure 3.1. In particular, our approach includes two main steps:

1. **Geographically adjust PMSI codes** (denoted as *adjusted PMSI codes*) so that the relationship between the *adjusted PMSI codes* and *INSEE codes* is 1-n
2. **Geographically adjust BVs** (denoted as *adjusted BVs*) so that the relationship between the *adjusted BVs* and the *adjusted PMSI codes* is 1-n

3.1.1.1 Geographically adjust PMSI codes

As shown in figure 3.1, both the *PMSI codes* and the *INSEE codes* have relationships with *Postal codes*, therefore we join the two tabular datasets through *Postal codes*. This joining step creates

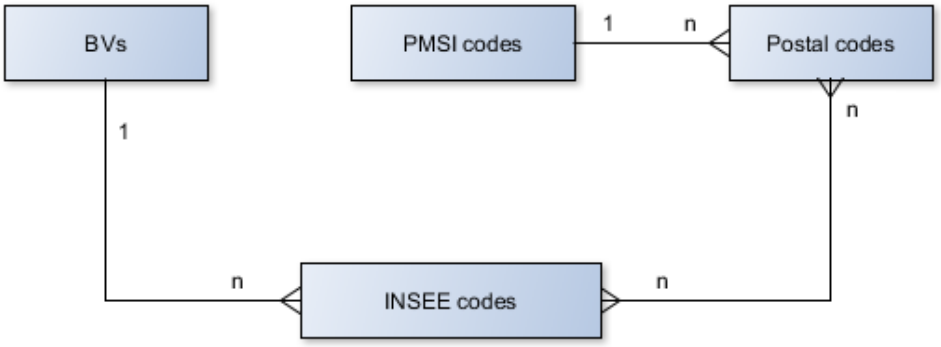


Figure 3.1: Relationships between *BVs*, *PMSI codes*, *Postal codes*, and *INSEE codes*

a new dataset in which the relationships between the *PMSI codes* and the *INSEE codes* are n-n as described in table 3.1.

Table 3.1: Example of n-n relationship between *PMSI codes* and the *INSEE codes*

INSEE code - PMSI code relationship	**Example**
INSEE 1 - PMSI 1	12176 - 12850
INSEE 1 - PMSI 2	12176 - 12000
INSEE 2 - PMSI 2	12090 - 12000
INSEE 2 - PMSI 3	12090 - 12510

However, there are very few *INSEE codes* that are linked to more than one *PMSI codes*. Particularly, in Occitanie region, France, this number is 14 over 4,565 INSEE codes [1]. Therefore, our approach to transform n-n relationship above to 1-n relationship by merging all the *PMSI codes* that are linked to same *INSEE codes* into one new *adjusted PMSI code*. For the example above, the three *PMSI codes* (*12850*, *12000*, and *12510*) are merged to create a new *adjusted PMSI code* which is, for example, *N0001*. Then we have 1-n relationships between *adjusted PMSI code* and *INSEE codes* as shown in table 3.2

Table 3.2: Example of geographically adjusting *PMSI codes*

INSEE code - PMSI code	**INSEE code - adjusted PMSI code**
12176 - 12850	12176 - N0001
12176 - 12000	
12090 - 12000	12090 - N0001
12090 - 12510	

[1] Dataset in 2015

To perform the process of geographically adjusting *PMSI codes*, we just mentioned, the following algorithm 1 is implemented.

Data: $fullDF$ is the PMSI-INSEE dataset
Result: Update $fullDF$ with adjusted PMSI codes
DF = subset of $fullDF$ such that INSEE codes linked to more than 1 PMSI code
$i = 1$ `// Just a variable to control adjusted PMSI codes`
while *not at end of* DF **do**
  $toMerge = \emptyset$
  $INSEE$ = first INSEE code in DF
  $findPMSI2Merge(INSEE)$ `// Both` DF `and` $toMerge$ `will be updated`
  $adjPMSI =$ "$N" + i$ `// adjusted PMSI codes have simple format like N1, N2`
  $i = i + 1$
  for $P \in toMerge$ **do**
    update $fullDF$ the PMSI code from P to $adjPMSI$
  end
end

Algorithm 1: Algorithm to geographically adjust *PMSI codes*

In algorithm 1, we are based on a recursive function *findPMSI2Merge* that finds *PMSI codes* to be merged starting from a "shared" *INSEE code*.

Parameter: $INSEE$ code to find PMSI codes to merge
Data: DF and $toMerge$ are global variables declared in Algorithm 1
Result: Update both DF and $toMerge$
$lstPMSIs$ = list of PMSI codes in DF linking to the $INSEE$ code
DF = subset of DF that the records containing the $INSEE$ code are removed
for $P \in lstPMSIs$ **do**
  Append P to $toMerge$
  $lstINSEEs$ = list of INSEE codes in DF linking to PMSI code P
  for $newINSEE \in lstINSEEs$ **do**
    $findPMSI2Merge(newINSEE)$ `// Recursive call`
  end
end

Algorithm 2: Recursive function *findPMSI2Merge* that finds *PMSI codes* to be merged starting from a "shared" *INSEE code*

3.1.1.2 Geographically adjust BVs

As the same as the step of geographically adjusting *PMSI codes*, we firstly join the dataset of *BVs* and the dataset of *adjusted PMSI codes* through *INSEE codes* with which both *BVs* and *adjusted*

PMSI codes have the relationship 1-n, we will have a new dataset in which the relationship between *BVs* and *adjusted PMSI codes* are n-n. However, with this new dataset of *BVs* and *adjusted PMSI codes*, if we merge all the *BVs* that share the same *adjusted PMSI codes* as we do while geographically adjusting *PMSI codes* above, it tends to merge all the *BVs* together. To deal with this problem, depending on the percentages in term of the population sizes of the shared *adjusted PMSI code* in a *BV*, the *BV* can be (1) adjusted its border to cover all parts of the *adjusted PMSI code*; (2) adjusted to not cover the part of *adjusted PMSI code*; (3) merged with other *BVs* to cover all parts of the *adjusted PMSI code*.

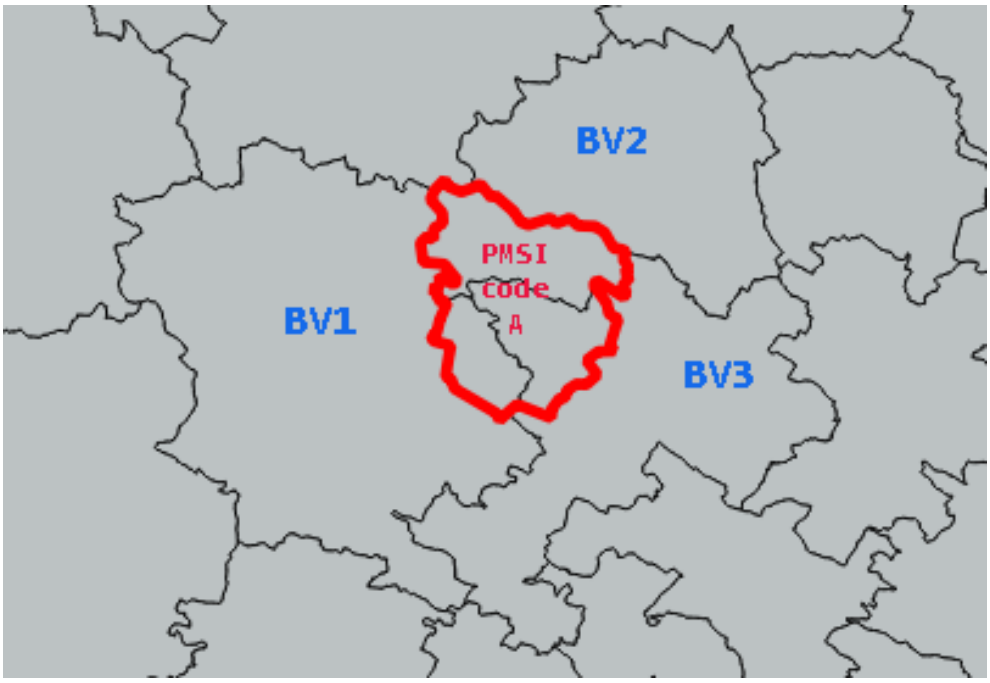


Figure 3.2: Spatial problem of transforming dataset from PMSI codes to BVs

To demonstrate the problem more clearly, an example of the problem is visualized using map (figure 3.2). As the map shows, three *BVs* (1, 2 and 3) share the the *adjusted PMSI code* A that has the border highlighted in red. In other words, the *adjusted PMSI code A* is divided into 3 parts. One part is in *BV1*, another part is in *BV2*, and the other part is in *BV3*. Because both the *BVs* and the *adjusted PMSI codes* have the relationship 1-n with the *INSEE codes* that contain the population information, the new dataset of *BVs* and *adjusted PMSI codes* can have the information of percentages of adjusted PMSI code A in each BV in term of the population sizes.

Now back to how the *BVs* are adjusted as mentioned above, we set two thresholds, the first one is for the percentages above, and second one is the maximum number of *BVs* to be merged at one time. The values of these thresholds in our work are 50% and 2 respectively.

To demonstrate how the process works, we are back to the example dataset (table 3.3). For the *adjusted PMSI code A*, because the percentage of it in *BV1* is 55% ($>$ 50%), the border of *BV1* is

Table 3.3: Example of percentage of adjusted PMSI code in each BV in term of population sizes.

#	**Adjusted PMSI code**	**BV**	**% of Population**
1	adjusted PMSI code A	BV1	55 %
2	adjusted PMSI code A	BV2	35 %
3	adjusted PMSI code A	BV3	10 %
4	adjusted PMSI code B	BV4	45 %
5	adjusted PMSI code B	BV5	30 %
6	adjusted PMSI code B	BV6	25 %

adjusted to cover entirely the *adjusted PMSI code A*. In other words, *BV2* and *BV3* are ignored in this case. Similarly, for the *adjusted PMSI code B*, because the percentage of it in *BV4* is 45% ($<$ 50%), *BV4* and *BV5* are merged into a *new adjusted NBV1* and then the border of this *adjusted NBV1* is also adjusted to cover entirely the adjusted PMSI code B. In this case, *BV6* is ignored.

To perform the process of geographically adjusting *BVs*, we just explained, our algorithm has two steps.

Step 1: Removing (ignoring) the *BVs* in which the population percentage of the shared *adjusted PMSI codes* are small.

Data: $fullDF$ is the adjustedPMSI-BV dataset
Result: Update $fullDF$ with adjusted BV codes
DF = copy of $fullDF$
$lstPMSIs$ = unique *adjusted PMSI codes* in DF
Order descending DF by *PMSI codes* and the percentage `// as shown in table 3.3`
for $P \in lstPMSIs$ **do**
  if *First record of P having the percentage >= 0.5* **then**
    Delete other records of P, but keep the first record of P, from DF
  else
    Delete other records of P, but keep the two first records of P, from DF
  end
end
$fullDF = DF$ `// Update final dataset`

Algorithm 3: Step 1 of geographically adjusting *BVs*

Step 2: From adjustedPMSI-BV dataset obtained from step 1, the *BVs* share same *adjusted PMSI codes* are merged to a *new adjusted BV*.

Data: $fullDF$ is the adjustedPMSI-BV dataset obtained from step 1
Result: Update $fullDF$ with adjusted BV codes
DF = subset of $fullDF$ such that adjusted PMSI codes linked to more than 1 BVs
$i = 1$ `// Just a variable to control adjusted BV codes`
while *not at end of* DF **do**
  $toMerge = \emptyset$
  $PMSI$ = first PMSI code in DF
  $findBV2Merge(PMSI)$ `// Same ideas as` $findPMSI2Merge$ `at Algorithm 2`
  $adjBV =$ "$NBV" + i$ `// adjusted BV codes have simple format like NBV1`
  $i = i + 1$
  for $BV \in toMerge$ **do**
    update $fullDF$ the BV code from BV to $adjBV$
  end
end

Algorithm 4: Algorithm to geographically adjust *BVs*

3.1.1.3 Dataset summary

After the pre-processing mentioned above, we finally have the dataset of 201 *adjusted BVs* containing the aggregated values of PAHs. These PAHs are particularly computed using AHRQ definition (section 2.3.1). These aggregated values are then standardized using the direct age-sex standardization (section 2.4.1). On the other side, the datasets of the potential determinants of PAHs are collected from many sources including the French Ministry of Health, the National Institute for Statistics and Economic Studies, the Regional Health Agency of Occitanie, French Health Insurance Fund ambulatory care claims database as well as open data. In particular, the datasets include:

- The primary care supply and hospital supply data including the densities of general practitioners, nurses, specialists, the densities of acute beds, travel time to the closest emergency department, and acute care hospital and medical group practice

- The socioeconomic data such as the median income, the unemployment rates, the proportion of population having an education level equal or above the baccalaureate, the proportion of population living in isolated rural areas, the proportion of workers in the active population.

- The epidemiological data such as the age-sex standardized rates of all-cause and premature mortality.

3.1.2 Evaluation and validation methods

As mentioned in the introduction, our approach are based on regression methods. To select most suitable method, we evaluate the potentials of the regression methods. In particular, we use both root-mean-square error (RMSE) and mean-absolute error (MAE) values for the performance evaluations [17]

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N} e_i^2}$$

$$MAE = \frac{1}{N}\sum_{i=1}^{N} |e_i|$$

In both formulas above, e_i (i = 1, 2, 3...N) are the errors (differences) between the predicted values from the regression methods and actual (observed) values.

On the other side, to validate the regression methods, we use leave-one-out validation method. Particularly, the predicted value of a BV is computed by using all the BVs except that BV as the training dataset. This approach requires us to repeat the training for any BV. Clearly, this approach does not work for big datasets, but it is not our case.

3.2 Regression methods and our evaluations related to our work

3.2.1 Multilinear regression

3.2.1.1 Introduction to linear regression

Linear regression is a method that predicts dependent variables through independent variables by fitting a linear equation to observed data. It could be said that whenever we need a regression method, the first choice is often multilinear regression because of its simplicity. To demonstrate how the linear regression method works, we consider a simple example with a dataset of heights and weights of some people (table 3.4).

The question is that we need to predict the weight of a person that we know his height is 170

Table 3.4: Example dataset to demonstrate linear regression.

#	**Height (cm)**	**Weight (kg)**
1	147	49
2	150	50
3	153	51
4	155	52
5	158	54
6	160	56
7	163	58
8	165	59
9	170	?

cm. In this case, the variables presents the weights and the heights are called dependent variable (labeled y) and independent variable (labeled x) respectively. In linear regression approach, we search for the straight line that best fits the given dataset (table 3.4).

$$\hat{y} = w_0 + w_1 x \tag{3.1}$$

If we call ε_i is the error or the different between the predicted value $\hat{y}_i$ and the observed value y_i, w_0 and w_1 in formula 3.1 can be found by minimizing:

$$R^2 = \sum_{i=1}^{N} (\varepsilon_i)^2 = \sum_{i=1}^{N} (y_i - \hat{y}_i)^2 = \sum_{i=1}^{N} [y_i - (w_0 + w_1 x_i)]^2 \tag{3.2}$$

The condition for R^2 to be a minimum is that its derivative equals 0 or in formula:

$$\frac{\partial R^2}{\partial w} = 0$$

That means

$$\frac{\partial R^2}{\partial w_0} = -2 \sum_{i=1}^{N} [y_i - (w_0 + w_1 x_i)] = 0 \tag{3.3}$$

and

$$\frac{\partial R^2}{\partial w_1} = -2 \sum_{i=1}^{N} [y_i - (w_0 + w_1 x_i)] x_i = 0 \tag{3.4}$$

With some mathematical works, 3.3 and 3.4 become:

$$w_0 N + w_1 \sum_{i=1}^{N} x_i = \sum_{i=1}^{N} y_i$$

$$w_0 \sum_{i=1}^{N} x_i + w_1 \sum_{i=1}^{N} (x_i)^2 = \sum_{i=1}^{N} x_i y_i$$

or in the format of matrix, we have:

$$\begin{bmatrix} N & \sum_{i=1}^{N} x_i \\ \sum_{i=1}^{N} x_i & \sum_{i=1}^{N} (x_i)^2 \end{bmatrix} \begin{bmatrix} w_0 \\ w_1 \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{N} y_i \\ \sum_{i=1}^{N} x_i y_i \end{bmatrix}$$

So that

$$\begin{bmatrix} w_0 \\ w_1 \end{bmatrix} = \begin{bmatrix} N & \sum_{i=1}^{N} x_i \\ \sum_{i=1}^{N} x_i & \sum_{i=1}^{N} (x_i)^2 \end{bmatrix}^{-1} \begin{bmatrix} \sum_{i=1}^{N} y_i \\ \sum_{i=1}^{N} x_i y_i \end{bmatrix} \tag{3.5}$$

Back to the example dataset above (table 3.4), we have

$N = 8$
$\sum_{i=1}^{N} x_i = 1,251$
$\sum_{i=1}^{N} y_i = 429$
$\sum_{i=1}^{N} (x_i)^2 = 195,901$
$\sum_{i=1}^{N} x_i y_i = 67,247$

Placing these values into formula 3.5, we have:

$$\begin{bmatrix} w_0 \\ w_1 \end{bmatrix} = \begin{bmatrix} 8 & 1,251 \\ 1,251 & 195,901 \end{bmatrix}^{-1} \begin{bmatrix} 429 \\ 67,247 \end{bmatrix} \tag{3.6}$$

Or w_0 = -38.27 and w_1 = 0.59. That means we have:

$$\hat{y} = -38.27 + 0.59x \tag{3.7}$$

This result is visualized by the scatter plot (figure 3.3) in which the formula 3.7 is the blue line.

Now, back to the question that the weight of a person whose height is 170 cm is predicted by replacing 170 for x in formula 3.7:

$$weight = -38.27 + 0.59 * 170 = 62.03\ kg$$

The example above is for the cases of one independent variable. For the cases of multi variables $(x_1, x_2, ..., x_N)$ or so-called multilinear regression, the formula to compute the predicted value $\hat{y}$ is:

$$\hat{y} = w_0 + w_1 x_1 + w_2 x_2 + ... + w_N x_N \tag{3.8}$$

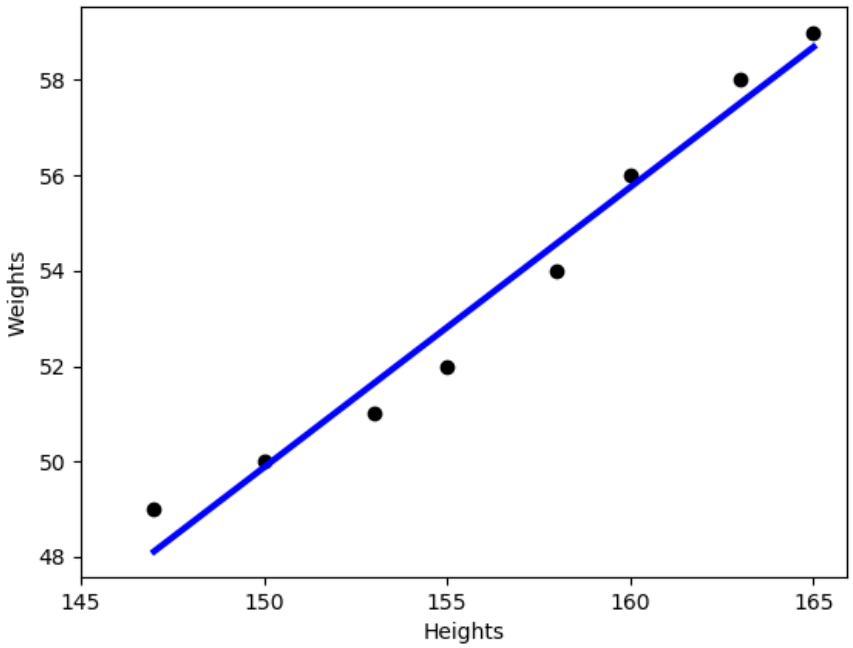


Figure 3.3: Linear regression of the example dataset

3.2.1.2 Evaluation related to our work

In our work, these predicted values are the rates of PAHs, so that if we apply the multilinear regression method, the formula will be:

$$P\hat{A}H = w_0 + w_1x_1 + w_2x_2 + ... + w_Nx_N \tag{3.9}$$

in which x_i is variable of dimension i. For example, x_1 stands for the density of the nurses.

As we have introduced above, in our work, we compare the predicted PAH values before ($P\hat{A}H_b$) and after ($P\hat{A}H_a$) trying to add new nurses for the biggest reduction of these predicted PAH values. The reduction rate at each BV can be mathematically presented by:

$$P\hat{A}H_b - P\hat{A}H_a = (w_0 + w_1x_1 + w_2x_2 + ... + w_Nx_N)_b - (w_0 + w_1x_1 + w_2x_2 + ... + w_Nx_N)_a \tag{3.10}$$

As we only make changes on the density of nurses (represented by x_1), equation (3.10) becomes:

$$P\hat{A}H_b - P\hat{A}H_a = (w_1x_1)_b - (w_1x_1)_a = w_1(x_{1b} - x_{1a}) \tag{3.11}$$

in which the density of nurses or the number of nurses per 10,000 people is computed as:

$$x_{1b} = \frac{Number\ of\ nurses}{Size\ of\ Population} * 10,000 \tag{3.12}$$

When we increase some nurses, for example A nurses, we have:

$$x_{1a} = \frac{(Number\ of\ nurses + A)}{Size\ of\ Population} * 10,000 \tag{3.13}$$

Apply (3.12) and (3.13) into (3.11), we have:

$$P\hat{A}H_b - P\hat{A}H_a = -w_1 * \frac{A}{Size\ of\ Population} * 10,000 \tag{3.14}$$

In addition, $(P\hat{A}H_b - P\hat{A}H_a)$ presents the difference between rates of PAHs per 1,000 inhabitants. Therefore, the expected number of PAHs to be reduced (*ExpectedPAHReduction*) is:

$$ExpectedPAHReduction = (P\hat{A}H_b - P\hat{A}H_a) * \frac{Size\ of\ Population}{1,000} \tag{3.15}$$

Finally, applying (3.14) to (3.15), we have the result:

$$ExpectedPAHReduction = -w_1 * 10 * A \tag{3.16}$$

Since Equation (3.16) will be applied for every BV, it indicates that the expected numbers of PAHs to be reduced are the same for every BV when we increase the same number of the nurses. That is definitely not the answer we are looking for.

On the other side, it should be noted that we do not compute the *ExpectedPAHReduction* as the differences between the actual numbers of PAHs before adding nurses and the predicted numbers of PAHs after adding nurses because by with this approach the BVs to be selected for adding nurses are actually the ones at which the differences (or the errors) between the actual values and the predicted values of PAHs before adding nurses are the biggest. That does not give us the right answer to our problem either.

3.2.2 K-nearest neighbors for regression

3.2.2.1 Introduction to K-nearest neighbors for regression

Another approach for regression method is K-nearest neighbors. The idea is that the similar objects tend to return the similar responses to same events. To demonstrate how the K-nearest neighbors for regression works, we consider another simple example predicting the people's weights,

Table 3.5: Example dataset to demonstrate K-nearest neighbors for regression

ID	**Height (cm)**	**Age (years)**	**Weight (kg)**
1	172	30	55
2	180	34	59
3	177	36	62
4	176	26	60
5	168	23	45
6	171	32	58
7	175	28	?

but this time besides their heights, we also have the information of their ages. The new example dataset is provided as table 3.5 that we need to predict the weight of the person whose ID is 7.

To find the answer, we use a scatter plot to visualize the people by their heights and their ages. These people are labeled by their IDs. As shown in the figure 3.4, person whose ID is 7 (called P7) is nearest to the person whose ID is 4 (called P4) in term of distance so that we believe that the weight of P7 is somewhat equal to the weight of P4. On the other words, we predict the weight of P7 = 60 kg.

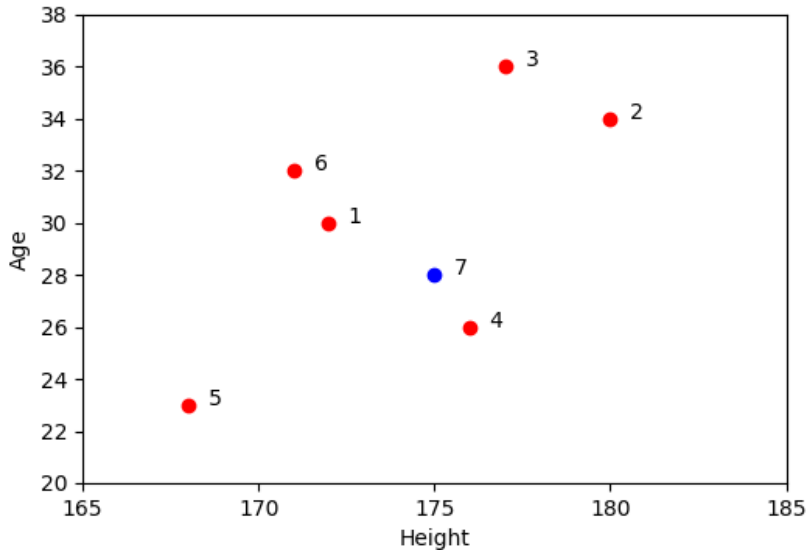


Figure 3.4: K-nearest neighbors regression of the example dataset

The method we use in the example above is called K-nearest neighbors for regression. The parameter K in this example has the value of 1, but it can be any small integer number such as 2, 3. In the cases that K $>$ 1, the predicted values could be the mean values of the K neighbors.

Back to the example above, with K = 2, the two nearest neighbors are P4 and P1. Therefore, the predicted weight of P7 is:

$$Weight\ of\ P7 = \frac{Weight\ of\ P4 + Weight\ of\ P1}{2} = \frac{60 + 55}{2} = 57.5\ kg$$

Generally, we have the formula for the mean value:

$$\hat{y} = \frac{1}{K}\sum_{i=1}^{K} y_i \tag{3.17}$$

Moreover, instead of using the mean value as above, we can also take into accounts the distances. By denoting w_i as the way the distance between each neighbor (P1 and P4) to the target object (P7) are taken into accounts, with i = 1, 2, ... K, then we can have:

$$\hat{y} = \frac{\sum_{i=1}^{K} w_i y_i}{\sum_{i=1}^{K} w_i} \tag{3.18}$$

There are many options computing w_i. However, there is a rule that smaller distances (d_i) return the bigger w_i than the larger distances. One common method to compute w_i is:

$$w_i = \frac{1}{d_i^{\alpha}}$$

In which α is any positive scalar, but typically $\alpha = 1$ or $\alpha = 2$ [9].

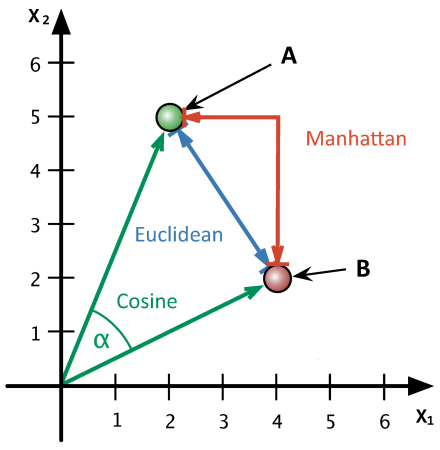
Lastly, we have been mentioning about distances between of objects. However, we have not mentioned how they are calculated. There are actually several methods to calculate distance d between object A and object B. The common methods include euclidean distance ($d_{euclidean}$), Mahatan distance ($d_{mahatan}$). Cosine distance (d_{cosine}) which is computed from cosine similarity (s_{cosine}) is another common method :

$$d_{euclidean} = \sqrt{\sum_{i=1}^{N}(x_{iA} - x_{iB})^2}$$

$$d_{mahatan} = \sum_{i=1}^{N}|x_{iA} - x_{iB}|$$

$$s_{cosine} = cos(\alpha) = \frac{\sum_{i=1}^{N} x_{iA}x_{iB}}{\sqrt{\sum_{i=1}^{N}(x_{iA})^2}\sqrt{\sum_{i=1}^{N}(x_{iB})^2}}$$

$$d_{cosine} = 1 - s_{cosine}$$



In the formulas computing the distances above, N is the number of dimensions/features. For the two point A and B example above, N = 2 and we have the corresponding values:

$$d_{euclidean} = \sqrt{(x_{1A} - x_{1B})^2 (x_{2A} - x_{2B})^2} = \sqrt{(2-4)^2 (5-2)^2} = 6$$

$$d_{mahatan} = |x_{1A} - x_{1B}| + |x_{2A} - x_{2B}| = |2-4| + |5-2| = 5$$

$$s_{cosine} = \frac{x_{1A}x_{1B} + x_{2A}x_{2B}}{\sqrt{x_{1A}^2 + x_{2A}^2}\sqrt{x_{1B}^2 + x_{2B}^2}} = \frac{2*4 + 5*2}{\sqrt{2^2 + 5^2}\sqrt{4^2 + 2^2}} = 0.75$$

$$d_{cosine} = 1 - s_{cosine} = 1 - 0.75 = 0.25$$

Finally, there is one note that we should normalize the dataset before applying k-nearest neighbors so that there is no feature dimension that makes the other feature dimensions useless. For example, we have a dataset that has two feature dimensions, one has the values ranging from 0 to 1, while the other dimention has the values ranging from 1,000 to 2,000. If we apply the k-nearest neighbors method without normalizing the dataset, then the first feature dimension has almost no contribution to the algorithm.

3.2.2.1 Evaluation related to our work

In our work, the feature dimensions of the BVs such as such as the densities of nurses or the levels of education are used to measure the distances between the BVs. To implement, we use language R with *distances* library [72] that has two functions: (1) *distances* function to calculate the distances between data points and (2) *nearest_ neighbor_ search* function to find the K nearest neighbors from a matrix of distances. After evaluating all methods based on the evaluation and validation approaches mentioned in section 3.1.2, we select euclidean distance and K = 5 for our work. In particular, the predicted rates of PAHs ($P\hat{A}H$) of a BV can be computed by the mean (average) values of its 5 nearest BVs.

Back to our project that is to compare the predicted rates of PAHs before and after adding nurses. At first, we compute the predicted rates of PAHs for all BVs before adding nurses. These values are $P\hat{A}H_b$. Then for each BV, we try to add new nurses, if at least one of its neighbors is changed, then we can have the new predicted rate of PAHs for that BV, $P\hat{A}H_a$. Finally, we select the BVs for adding more nurses by the biggest reduction of the expected number of PAHs (*ExpectedPAHReduction*).

$$ExpectedPAHReduction = (P\hat{A}H_b - P\hat{A}H_a) * \frac{Size\ of\ Population}{1,000}$$

At the beginning, this approach looked promising to us, but it actually does not work in our case because of the following limitations:

- When the dimension of the variables (the number of the attributes) is high, then the neighbors will not be able to be changed if we just make small change on one dimension (density of nurses in our case)
- Also regarding to the dimension of the variables, changing the size of dimension means changing the opportunities for the BVs to change the new predicted rates of PAHs. That leads to the unstable results in our work.

3.2.3 Neural networks for regression

3.2.3.1 Introduction to neural networks

Neural networks could be very promising to any problem regardless of classification or regression. To introduce about Neural networks we start with Gradient descent algorithm.

A. Gradient descent algorithm

In machine learning, we often have to solve the optimization problem of cost functions. For example, for the linear regression above, the coefficient (w_1) and intercept (w_0) were found by solving the minimum problem of the total square of the errors (formula 3.2). Moreover, the optimization problem are solved by solving the formulas of the corresponding derivative functions equal 0 (ex. formulas 3.3 and 3.4). However, solving the derivative problems are sometimes difficult or even impossible. Therefore, gradient descent algorithm was introduced as approximate solutions. To demonstrate how the gradient descent algorithm works, we consider a simple example that is to solve the minimum of the function below.

$$f(x) = \frac{1}{2}(x-1)^2 - 2 \tag{3.19}$$

Finding the solution to the optimization of $f(x)$ by solving its derivative equals 0

$$f'(x) = x - 1 = 0 \; or \; x = 1 \tag{3.20}$$

Therefore, the minimum value of $f(x)$ above is $f(1)$ and it is -2. The solution can be presented by graph (figure 3.5).

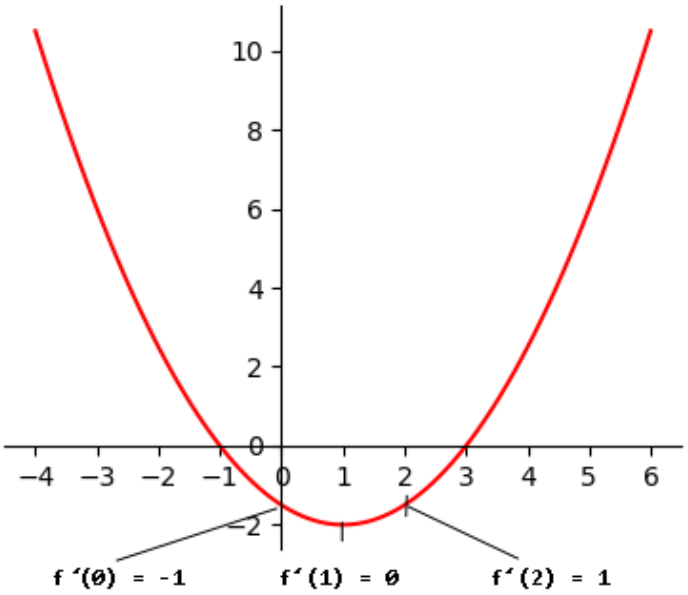


Figure 3.5: Example of gradient descent algorithm

However, imaging that we were not able to solve $f'(x) = 0$, we would apply the gradient descent approach which is to find x^* so that $f'(x^*) \approx 0$. In particular, gradient descent algorithm will run a loop and at every step of the loop, it updates the x so that the new x gets closer and closer to x^*. To formula how the algorithm works, we denote that at a point of time t, x has the value of x_t and the corresponding derivative $f'(x_t)$. Now at the point of time $(t+1)$, the new x has value of x_{t+1} and corresponding $f'(x_{t+1})$. If x_{t+1} can be updated from x_t:

$$x_{t+1} = x_t + \Delta \tag{3.21}$$

In order to $f'(x_{t+1})$ get closer to zero than $f'(x_t)$,

$$\Delta = -\eta f'(x_t) \tag{3.22}$$

In which η is a positive number and it is called learning rate.

To demonstrate how algorithm works, we are back to the example above (figure 3.5). Given that $x_t = 0$ and then $f'(0) = -1$. If we select $\eta = 0.5$, we have

$$x_{t+1} = x_t - \eta f'(x_t) = 0 - 0.5 * (-1) = 0.5$$

Then $f'(x_{t+1}) = f'(0.5) = 0.5 - 1 = -0.5$. The new $f'(x_{t+1})$ is closer to zero than $f'(x_t)$.

On the other side, with $x_t = 2$, then $f'(2) = 1$

$$x_{t+1} = x_t - \eta f'(x_t) = 2 - 0.5 * 1 = 1.5$$

Then $f'(x_{t+1}) = f'(1.5) = 1.5 - 1 = 0.5$. The new $f'(x_{t+1})$ is also closer to zero than $f'(x_t)$.

In summary, given that our approach for the loop is that the loop will stop after K steps, then the gradient descent algorithm for the example above can be implemented as below:

Parameter: x_0, eta, K
Result: x after K steps
$x = x_0$
$i = 0$
while $i < K$ **do**
    $derivative = x - 1$
    $x = x - eta * derivative$
    $i = i + 1$
end
return x

Algorithm 5: Example of gradient descent algorithm after K steps

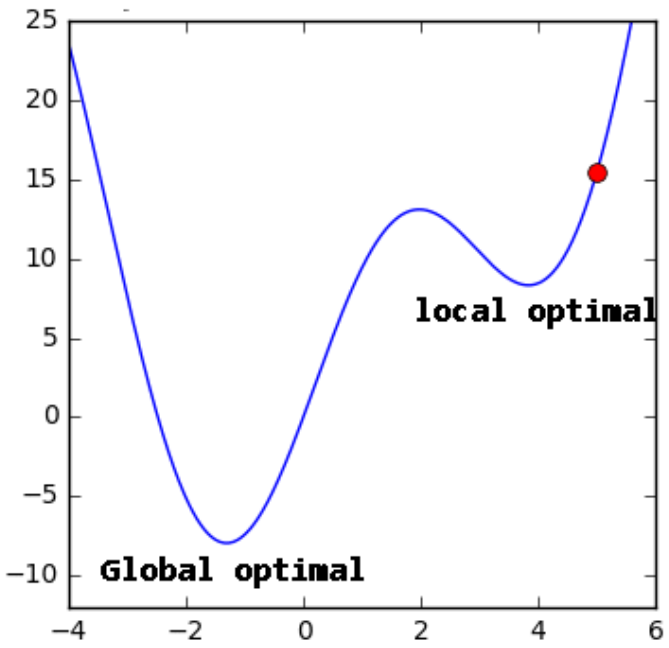


Figure 3.6: Local optimal problem of simple gradient descent algorithm [93]

In the example above, we introduce the most simple gradient descent algorithm which is for only one variable. The same principle is applied for the cases of multi variables. However, in practice,

this simple algorithm has many limitations, for example, the solutions found can be at the local optimal rather than the global optimal (as shown in figure 3.6). To improve the learning speed as well as to deal with the limitations, many optimization algorithms are introduced:

- Momentum [66]
- Nesterov accelerated gradient (NAG) [60]
- Adagrad [25]
- Adadelta [97]
- RMSprop [40]
- Adam [44]
- Nadam [24]

On the other hand, when we work with training data that has, for example, N data points. The gradient descent algorithms are also categorized into three groups. The first group is called batch gradient descent when all the N data points are used when the variables are updated - one step of the loop mentioned. The second group is called stochastic gradient descent when only one data point is used instead of N. The last group is called mini-batch gradient descent when the number of data points is ranging from 2 to N-1. In addition, each time all the N data points are used to train is called one *epoch*. In the cases that we use batch gradient descent, the number of *epoch* equals the number of steps of the loop.

In this section, we have briefly introduced about the Gradient descent algorithm, in the next section, we will introduce about Perception learning algorithm, another foundation component of neural networks.

B. Perception learning algorithm

Together with the gradient descent algorithm, perception learning algorithm (PLA) could be considered as another foundation of neural networks. To demonstrate how the PLA works, we use a simple example of binary classification problem (figure 3.7). In particular, the work is to classify the triangle point, with the question mark, to either the group of the small blue squares or the group of the small red circles.

The approach of PLA is that from the training data which are already known as either blue squares or red circles, we need to find a straight line that separates the blue squares from red

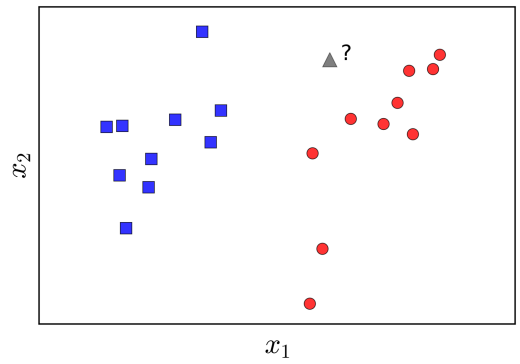


Figure 3.7: An example of classification problem [93]

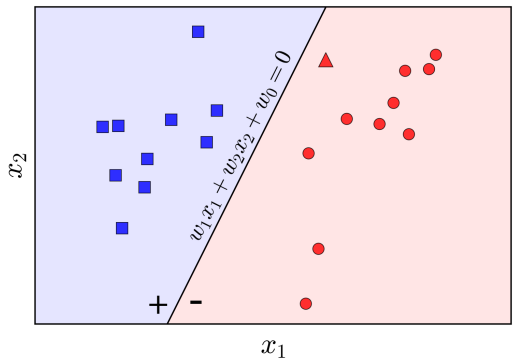


Figure 3.8: A solution to classification problem [93]

circles (figure 3.8). In this example, because the triangle point is in the part of red circles, it is predicted as a red circle.

Mathematically, we denote $f_w(x)$ for the straight line above:

$$f_w(x) = w_1x_1 + w_2x_2 + w_0$$

Or in general,

$$f_w(x) = w^T x$$

in which, $x_0 = 1$ is included in data point x

Moreover, if we assign 1 for the blue squares and -1 for the red circles then the class of a point x, called label(x), will be

$$label(x) = \begin{cases} 1 & if\ f_w(x) \geq 0 \\ -1 & if\ f_w(x) < 0 \end{cases}$$

This equation can be shortened as

$$label(x) = sign(f_w(x))$$

In which, *sign* is the function determines the sign of expressions, given that $sign(0) = 1$.

Now, given that we have a training dataset with N data points $\{x_i, y_i\}$ where $i = 1, 2, .., N$. As the label of a data point, y_i will have a value of either 1 or -1. With any function $f_w(x)$ mentioned above, consider the expression below

$$J_i = -y_i * sign(f_w(x_i))$$

It can be easily proved that J_i receives value of 1 if x_i is assigned to the wrong class and value of -1 if it is assigned to the right class. For example, if x_i belongs to class 1, then $y_i = 1$ or $-y_i = -1$. However, it is assigned to the wrong class or we have $sign(f_w(x_i)) = -1$. Placing all these values to $J_i = -1 * (-1) = 1$

Extending the expression J_i, given that $\mathcal{M}$ is the set of the data points which are assigned to wrong classes, we can compute the total number of data points which are assigned to wrong classes.

$$J_w = \sum_{x_i \in \mathcal{M}} J_i = \sum_{x_i \in \mathcal{M}} (-y_i * sign(f_w(x_i))) = \sum_{x_i \in \mathcal{M}} (-y_i * sign(w^T x_i)) \tag{3.23}$$

Since J_w measures the total number of data points which are assigned to wrong classes, it becomes the cost function of the binary classification problem. On the other words, the job becomes solving the optimization problem. As the function $sign$ makes it impossible to compute the derivative of J_w, we change the cost function by removing $sign$, the new cost function becomes

$$J_w = \sum_{x_i \in \mathcal{M}} (-y_i * w^T x_i) \tag{3.24}$$

To solve this optimization problem or in other words to find values for w so that J_w gets the minimum (or approximate) value, we can apply stochastic gradient descent introduced above. In particular, with a data point (x_i, y_i) that is assigned to wrong class, we have the corresponding cost function

$$J_w(x_i, y_i) = -y_i * w^T x_i$$

and its derivative by w

$$J'_w(x_i, y_i) = -y_i x_i$$

Since we are applying gradient descent, the new w_{t+1} will be updated by

$$w_{t+1} = w_t - \eta J'_w(x_i, y_i) = w_t + \eta y_i x_i$$

In summary, the above algorithm of finding the appropriate w can be summarized as below.

In this section, we have briefly just introduced about the perception learning algorithm that is the basic component of neural networks that is presented in the next section.

C. Feed forward neural networks

The perception learning algorithm above can be visualized as figure 3.9. This could be the simplest neural networks model that have one layer besides the inputs and the output. In addition, this layer has only one neural network unit.

Data: Training dataset with N data points $\{x_i, y_i\}$
Result: The appropriate w
Randomly select w
while *not reach the number of epoch* **do**
  $nWrongClasses = 0$
  for *each* $\{x_i, y_i\} \in dataset$ **do**
    if $y_i <> sign(w^T x_i)$ **then**
      $w = w + \eta y_i x_i$
      $nWrongClasses = nWrongClasses + 1$
    end
  end
  if $nWrongClasses == 0$ **then**
    Finish
  end
end

Algorithm 6: Algorithm to find w

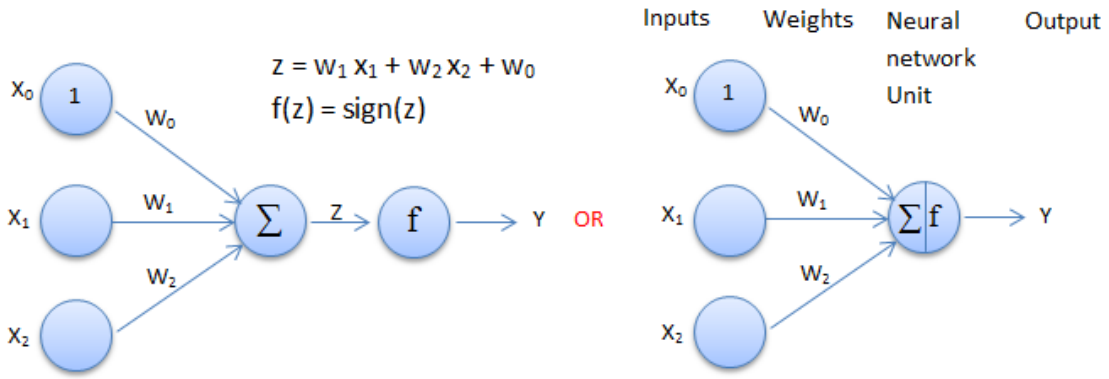


Figure 3.9: A simple neural networks model

The simple neural networks model is only to solve the simple problem. For the complicated cases, we would need more layers (so called hidden layers) and at each layer we need more neural network units. A neural network model where all the units of the previous layer are connected to all the units of the next layer is called fully connected feed forward neural networks model (figure 3.10). Moreover, the way the gradient descent works, the first vector W (W_1 in figure 3.10) is updated from the errors of the outputs, is done by the back-propagation algorithm [50].

On the other side, inside each network unit, there is a function (labeled f) such as the function *sign* in the previous example. This function is called *activation* function. The common *activation* functions include

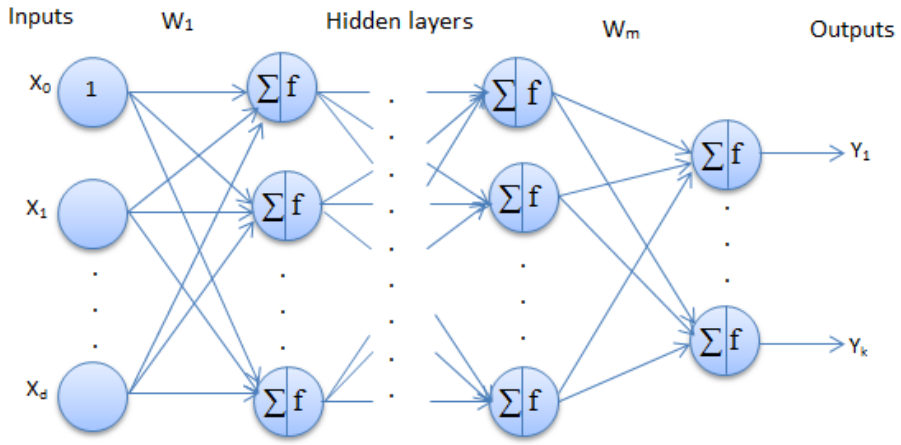


Figure 3.10: A sample fully connected feed forward neural networks model

- Binary step
- Linear
- ReLU
- LeakyReLU
- Sigmoid
- Tanh
- Softmax

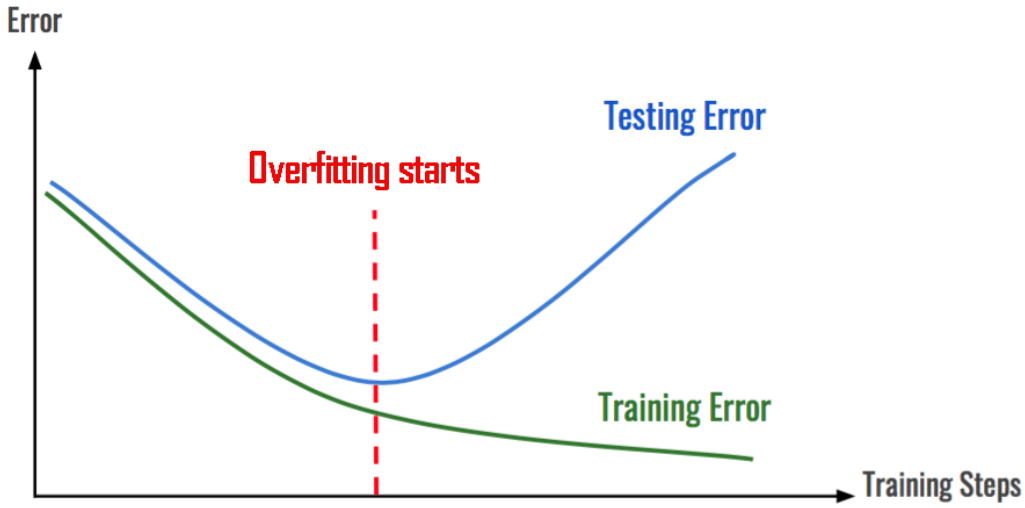


Figure 3.11: A sample of overfitting

In addition, when we apply machine learning in general and neural networks in particular in our project, we often face with a problem called *overfitting*. As it is visualized in figure 3.11, *overfitting* happens when the model responds too closely or exactly to the training data, but fails to fit with other data. There are several common solutions to the overfitting problem.

- Simplifying training model or decrease the complexity of the model. There is no general rule on how much to remove or how large your network should be. But, if your neural network is overfitting, try making it smaller.
- Early stopping. Early stopping rules provide guidance as to how many iterations can be run before the model begins to overfit (figure 3.11).
- Data augmentation or increase size of the training data if it is possible.
- Regularization. Regularization is a technique that adds a penalty term to the loss function. The most common techniques are known as L1 and L2 regularization
- Dropouts. Dropouts modify the network itself by randomly dropping neurons from the neural network during training in each iteration.

3.2.3.2 Evaluation related to our work

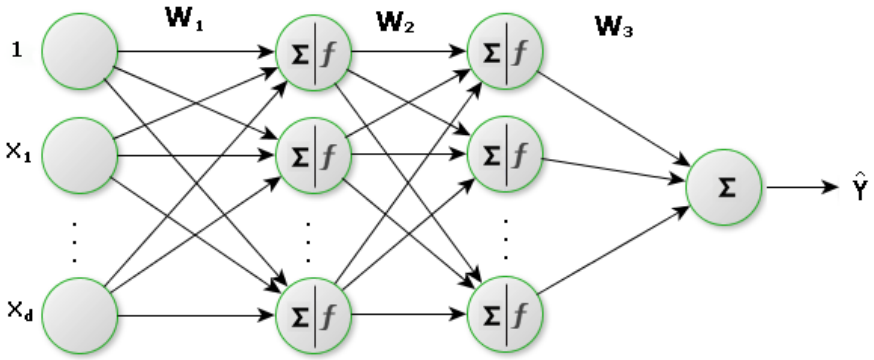


Figure 3.12: Neural network two hidden layers for regression

In the previous section, we have introduced neural networks, but they are for the classification problem. For the regression problems like the one in our work, there are no *activation* functions (no f in the network units) at the corresponding output layer.

Suppose that if we deploy the neural network with only one layer for regression, the predicted values will be in a linear formula:

$$P\hat{A}H = w_0 + w_1x_1 + w_2x_2 + ... + w_Nx_N$$

Compared this formula with the multilinear regression formula, they are the same. As explained in the multilinear regression case above, we cannot use the neural network one layer regression to solve our problem. That means we need at least one more hidden layer for our work (Figure 3.12). Moreover, since the output is regression, the *loss* function (or *cost* function) to be used is

mean square error. In particular, since we use R language with *keras* library [27], our function to build the model has the following template.

```
build_model<-function(var_dim, unit_n, layer_n,act_func,dropout_rate, reg_l2_rate,opt)
{
    model <- keras_model_sequential() %>%
    layer_dense(units = unit_n, activation = act_func, input_shape = c(var_dim)) %>%
    for(i in 1: layer_n)
    {
        layer_dense(units = unit_n, activation = act_func) %>%
        if(dropout_rate > 0)
        {
           layer_dropout(rate = dropout_rate) %>%
        }
    }
    if(reg_l2_rate)
    {
        layer_dense(units = unit_n, kernel_regularizer=regularizer_l2(reg_l2_rate))%>%
    }
    layer_dense(units = 1)

    model %>% compile(optimizer = opt,
        loss = "mean_squared_error")
}
```

In the function above, there are some parameters:

- *var_dim*: number (dimension) of the variables;
- *unit_n*: number of units in the hidden layers.
- *layer_n*: number of the hidden layers.
- *act_func*: The *activation* function such as *relu* to be used at the hidden layers.
- *dropout_rate*: the dropout rate if dropout technique is used to avoid overfitting. It should have the value between 0 and 1.
- *reg_l2_rate*: the rate of regularization l2 if regularization l2 technique is used to avoid overfitting.
- *opt*: The optimization technique such as *adam* to be used

Unfortunately, after trying with different models (or different parameters): more hidden layers, different activation functions at the hidden layers as well as applying different techniques such as L1, L2 regularization or dropout to avoid overfitting, we have failed to get the better results for the predicted rates of PAHs compared with the support vector machine for regression (SVR) method (Table 3.6). Another negative point of neural networks is that they work like "black boxes" on how a certain output is produced and therefore it is very difficult to explain their outputs to others. Hence, we think that the neural networks method is not the right method for our work.

3.2.4 Support vector machine for regression

3.2.4.1 Introduction to support vector machine for regression

In practice, support vector machine (SVM) has been applied widely in classification problem, but it can also be used as a regression method. The method was introduced by Vapnik and his colleagues [90] and has been applied in many fields such as financial forecasting [88]. In this paper, we present the ideas of this method.

A. Linear cases:

Given a dataset $\{(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)\}$, in which $x_i \in R^d$ and $y_i \in R$. At first, the idea of the support vector machine for regression (SVR) is to find the straight predicted line $\hat{y} = wx + b$ that has two conditions:

- The straight predicted line is parallel as possible to the line y = 0;
- All the errors (differences) between the actual values y_i and the predicted values $\hat{y_i}$ are not greater than a given ε (Figure 3.13)

Mathematically, the conditions can be formulated as a convex optimization problem (3.25) below:

$$\begin{aligned} &\text{minimize} && \frac{1}{2}w^2 \\ &\text{subject to} && \begin{cases} y_i - wx_i - b \leq \varepsilon \\ wx_i + b - y_i \leq \varepsilon \end{cases} \end{aligned} \tag{3.25}$$

As it can be imagined, when the given ε is big enough, solving this optimization problem is feasible. However, in most cases, this method does not return good predicted lines. Hence, a "soft margin" loss function was introduced [22] to allow the cases that some errors are bigger than

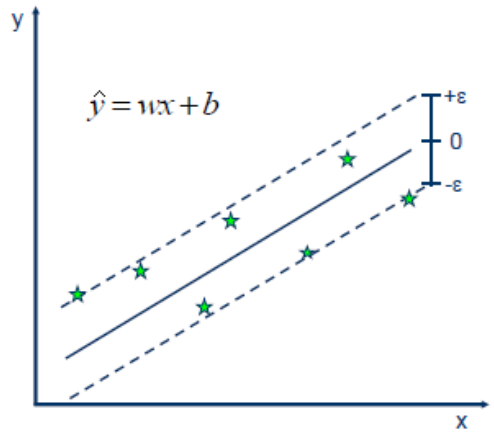


Figure 3.13: SVR - The first idea[73]

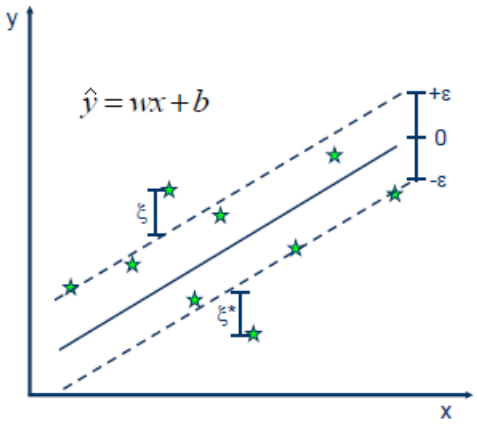


Figure 3.14: SVR with slack variables[73]

the given ε. In particular, in the modified approach, slack variables ξ_i, ξ_i^* are used to present the differences between the errors and the given ε (Figure 3.14). Correspondingly, the original optimization problem (Formula 3.25) above turns to the new one (Formula 3.26) below:

$$\begin{aligned} &\text{minimize} && \frac{1}{2}w^2 + C\sum_{i=1}^{N}(\xi_i + \xi_i^*) \\ &\text{subject to} && \begin{cases} y_i - wx_i - b \leq \varepsilon + \xi_i \\ wx_i + b - y_i \leq \varepsilon + \xi_i^* \\ \xi_i \geq 0 \\ \xi_i^* \geq 0 \end{cases} \end{aligned} \tag{3.26}$$

In which, C is a constant that determines the trade-off between the flatness of the line and amount up to which ε the errors are accepted.

The optimization problem (Formula 3.26) above can be solved by using dual formulation that constructs a Lagrange function from both the objective function and the corresponding constraints. In particular, the Lagrange function of formula 3.26 above is presented as below:

$$\begin{aligned} L = \frac{1}{2}w^2 + C\sum_{i=1}^{N}(\xi_i + \xi_i^*) \\ + \sum_{i=1}^{N}\alpha_i(y_i - wx_i - b - \varepsilon - \xi_i) + \sum_{i=1}^{N}\alpha_i^*(wx_i + b - y_i - \varepsilon - \xi_i^*) \\ - \sum_{i=1}^{N}(\eta_i\xi_i) - \sum_{i=1}^{N}(\eta_i^*\xi_i^*) \end{aligned} \tag{3.27}$$

By solving this mathematics optimization problem (Formula 3.27) [78], we have the result below:

$$w = \sum_{i=1}^{N} (\alpha_i - \alpha_i^*) x_i \tag{3.28}$$

and therefore, we have

$$\hat{y} = f(x) = \sum_{i=1}^{N} (\alpha_i - \alpha_i^*) x_i x + b \tag{3.29}$$

B. Non-linear cases:

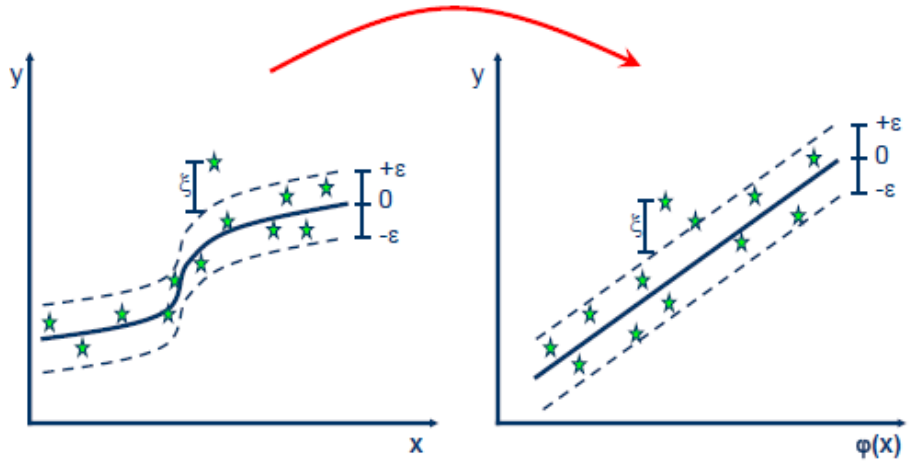


Figure 3.15: SVR for non-linear cases [73]

As explained in the multilinear regression section, the linear case (Formula 3.29) does not work in our case. However, we also can apply SVR for the non-linear cases in which the predicted lines are not straight lines. In particular, for the non-linear problems, the way the method works is to transfer the original independent variables x into a new coordinate system $\varphi(x)$ so that in the new coordinate system the non-linear problems turn to the linear problems (Figure 3.15). Consequently, in the new coordinate system, the formula 3.29 to compute the predicted values $\hat{y}$ becomes formula 3.30 [88, 78, 73]

$$\hat{y} = \sum_{i=1}^{N} (\alpha_i - \alpha_i^*) \varphi(x_i) \varphi(x) + b \tag{3.30}$$

In practice, the number of the new dimensions of $\varphi(x)$ is often very high or even infinite. Hence, computing $\varphi(x)$ from x becomes difficult or even unfeasible. Therefore, a technique called **kernel trick**, $K(x_i, x_j) = \varphi(x_i)\varphi(x_j)$, is applied to directly compute $\varphi(x_i)\varphi(x)$ rather than computing all $\varphi(x)$. Particularly, the following kernel functions are often used:

Polynomial:

$$K(x_i, x_j) = (x_i, x_j)^d$$

Gaussian Radial Basic Function - RBF:

$$K(x_i, x_j) = exp\left(- \frac{(x_i - x_j)^2}{2\sigma^2} \right)$$

3.2.4.2 Evaluation related to our work

Related to our work, to implement SVR, we rely on language R and the *e1071* library [55]. In particular, this library has a corresponding function *svm* that we need to provides the values for the parameters corresponding to the kernel functions, C, σ, and ε. After testing all the kernel functions, we have found that RBF returns the predicted values that are closest to the actual rates of PAHs. In addition, comparing with the results from the other regression methods presented previously, the predicted values by SVR using RBF are closest to the actual rates of PAHs (Table 3.6). More specifically, Table 3.6 presents the performance of the regression methods on our dataset in which we use both root-mean-square error (RMSE) and mean-absolute error (MAE) values for the performance evaluations:

Table 3.6: Performance evaluations of regression methods on our dataset

Method	**RMSE**	**MAE**
SVR using RBF	0.98	0.76
Multi-linear regression	1.04	0.82
K-nearest neighbors	1.03	0.80
Neural networks	1.13	0.87

Based on this result and the analysis for the possible application of the regression methods in our work mentioned above, we have agreed that the SVR method is the best choice for our work.

3.3 Extending support vector machine regression in recommending the optimal actions targeting on the geographic areas

As we mentioned in the introduction, the purpose of our work is to select the cross-border living areas (fr. Bassins de vie - BVs) in Occitanie region, France for adding nurses for the most effective PAHs reduction. In particular, we select these BVs by comparing the predicted rates of PAHs before and after trying to add new nurses in every BV. The BVs to be selected are the ones that return the biggest reduction of these predicted values. Hereafter we present the ideas in details.

3.3.1 Possible constraints

The first thing we need to consider is that there are some constraints on the number of nurses to be added. The first constraint should be the budget that the health authorities can spend for the health service improvement. This constraint indicates that the total number of nurses to be added in the whole region is limited. Another constraint we must consider is to ensure equal access to health care for the inhabitant living in the region. The later constraint can be defined by (1) the maximum number of to-be-added nurses in each BV; and (2) making sure that in the to-be-selected BVs, the densities of the nurses must not be greater than a given threshold. The latter to make sure that we do not add nurses in the BVs whose densities of nurses are already high. To sum up, we have three possible constraints in our work as below:

- The maximum number of nurses in total that can be added into the whole region. We denote this constrain as **maxGlobal**
- The maximum number of nurses that can be added in each BV. We denote this constrain as **maxLocal**
- The maximum density of nurses that can be reached in each BV. We denote this constrain as **maxLocalDensity**

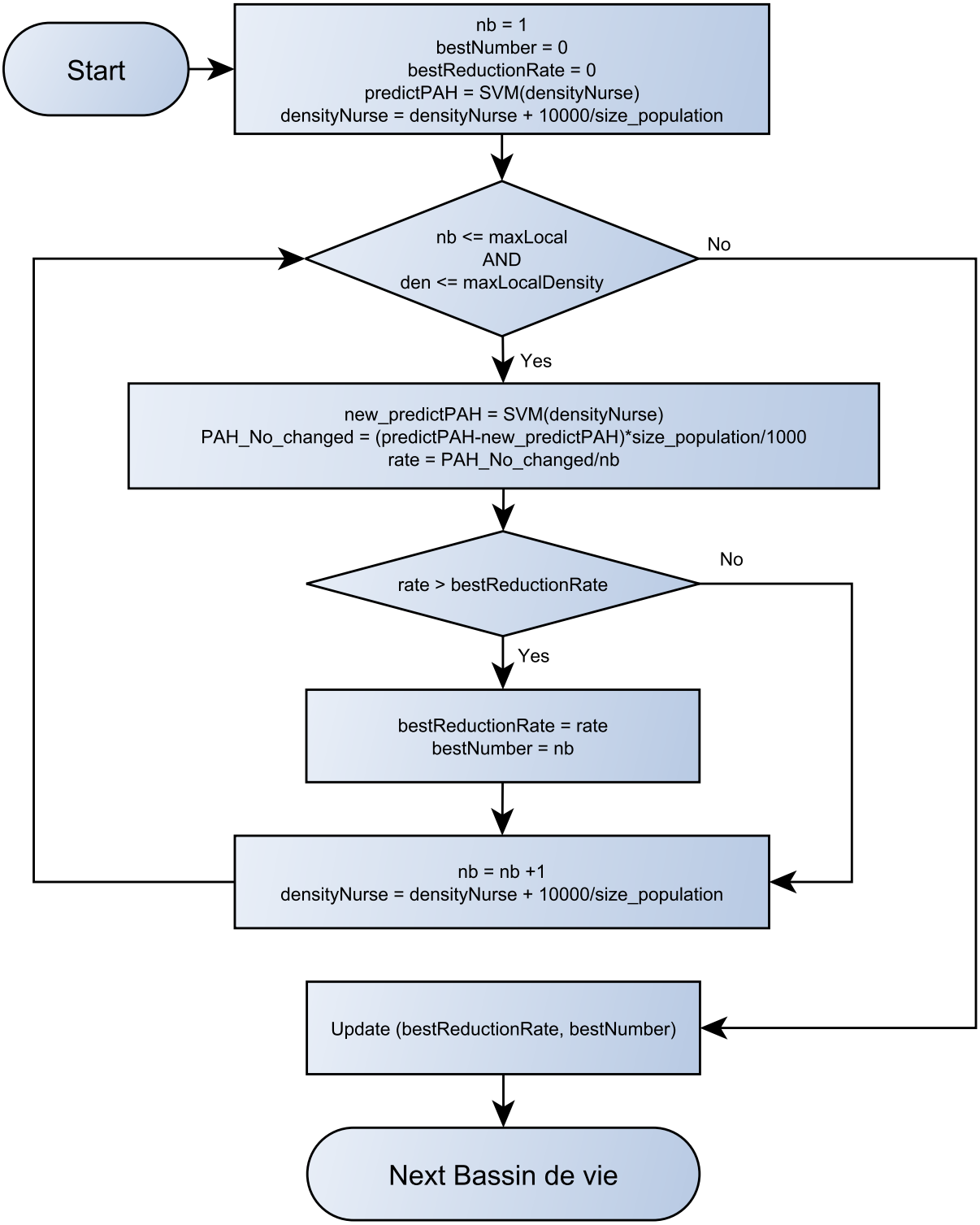


Figure 3.16: Process flow to find the biggest reduction rate of PAH per to-be-added nurse and best number of to-be-added nurses in each BV

3.3.2 Best numbers of to-be-added nurses and the biggest PAH reduction rates

After defining the constraints, the second step is to find the best number of nurses to be added in each BV. In particular, in this step, at each BV, we try to add nurses one by one until we reach either the defined **maxLocal** or the maximum density of nurses **maxLocalDensity**. Each time adding a nurse, we compute the reduction rate of PAHs per added nurse to identify at each BV (1) the biggest reduction rate (denoted **bestReductionRate**); and (2) the best number of to-be-added nurses (denoted **bestNumber**) corresponding to the **bestReductionRate**. The whole process is described in the Figure 3.16.

In the process flow described in Figure 3.16, it should be noted that in our work, the PAHs are the standardized rates per 1,000 people so that we need to compute the number of PAHs to be reduced (variable *PAH_No_changed* in Figure 3.16) after increasing nurses in order to get the reduction rate of PAHs per to-be-added nurse (*rate*). One important thing to note here is the SVM function (*SVM(densityNurse)*) that actually the SVR method we mentioned in the previous section. We firstly train SVR model using the dataset of PAHs and its potential determinants, then we can get the predicted rates of PAHs before and after trying to add nurses to the BVs.

The final result of this step will return the list of all the BVs with their information of **bestReductionRate** and **bestNumber** of to-be-added nurses.

3.3.3 BVs to be selected

After having the values of **bestReductionRate** for all the BVs, the task to find BVs for adding new nurses becomes easy. More specifically, the BVs to be selected are the ones whose **bestReductionRate** are the biggest. However, to avoid the cases that in the BVs to selected, the actual rates of PAHs are already small, we add one more condition to the BVs to be selected that we only select a BV if its actual rate of PAHs is higher than its predicted rate of PAHs ($actualPAH \geq predictPAH$ in Figure 3.17). The process of finding BVs for adding nurses is described in Figure 3.17. In this process, we firstly order the list of the BVs descendingly by their **bestReductionRate** (function *orderBVsByBestDeduectionRate* in Figure 3.17). After that we select the top first BVs until either we reach the maximum number of to-be-added nurses (**maxGlobal**) in the whole region or we reach the last BVs in the list (reach the total number of BVs, $nbBVs$ in Figure 3.17). There is a note in Figure 3.17 that *BV(Attr, index)* function returns the value of the attribute (*Attr*) of the BV associated with its *index*.

The output of this step is a list of the to-be-selected BVs (*selectedBVs* in Figure 3.17) for adding

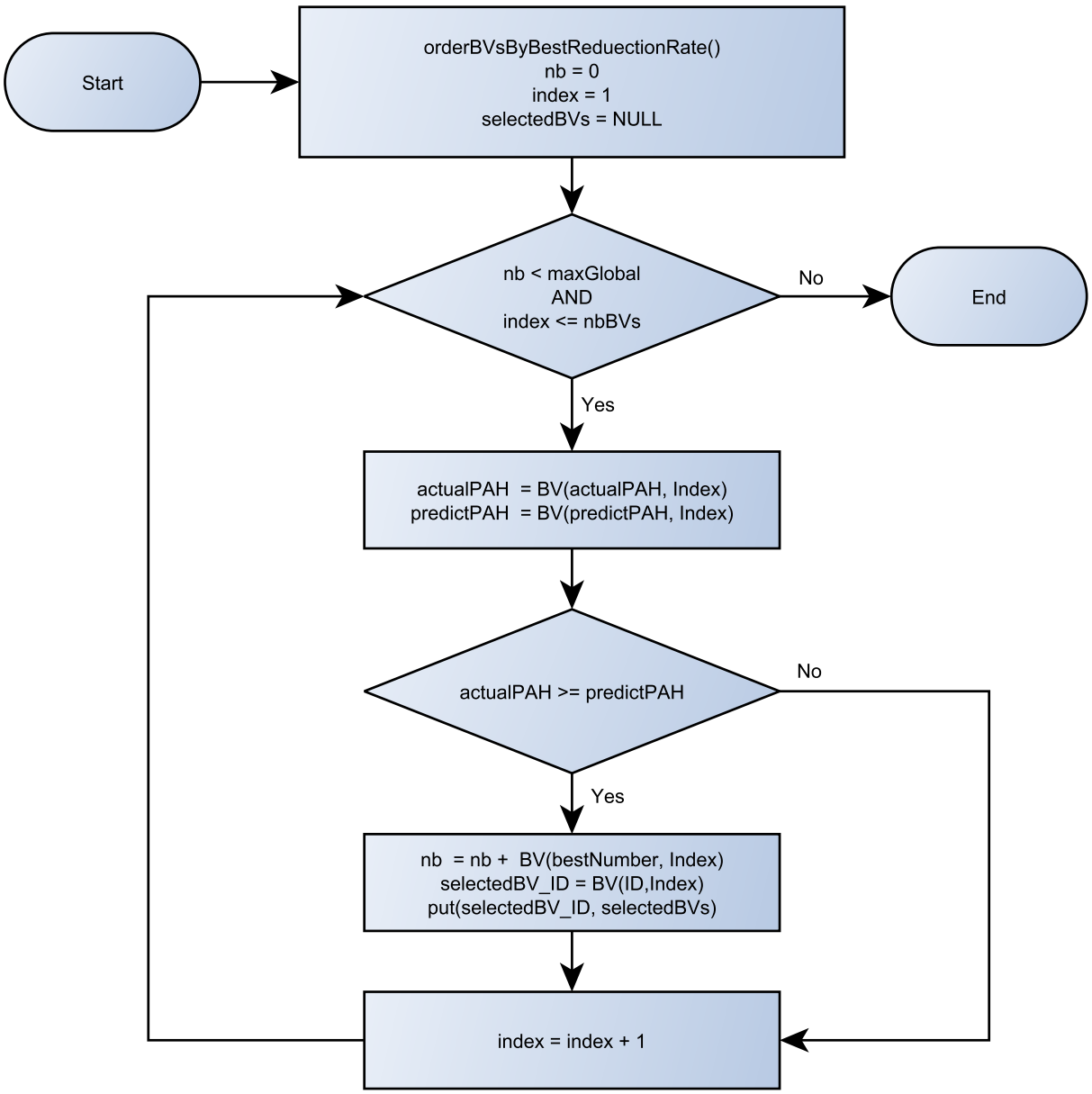


Figure 3.17: Process flow to select the BVs for adding more nurses

more nurses and the best number of to-be-added nurses in each BV. There is a point that this algorithm might return the total number of to-be-added nurses little more than the constraint on the maximum number of can-be-added nurses in the whole region (**maxGlobal**). But this does not cause any problem as we also know how many to-be-added nurses in every BV and the decision makers can decide to either increase budget or adjust the number of to-be-added nurses in the last BV in the selected list.

3.4 Result and discussions

As mentioned in the previous section, the output of the algorithm is the list of BVs where nurses should be added and the number of nurses to be added in order to obtain the highest decrease in the number of PAH. For better visualization for the decision makers, we rely on spatial maps. For example, the map below (Figure 3.18) recommends the BVs to increase nurses (the darker colors indicate stronger recommendation) and the optimal number of nurses to be added (the labels in red) should be added in those BVs for the biggest reduction of PAH according to the corresponding constraints.

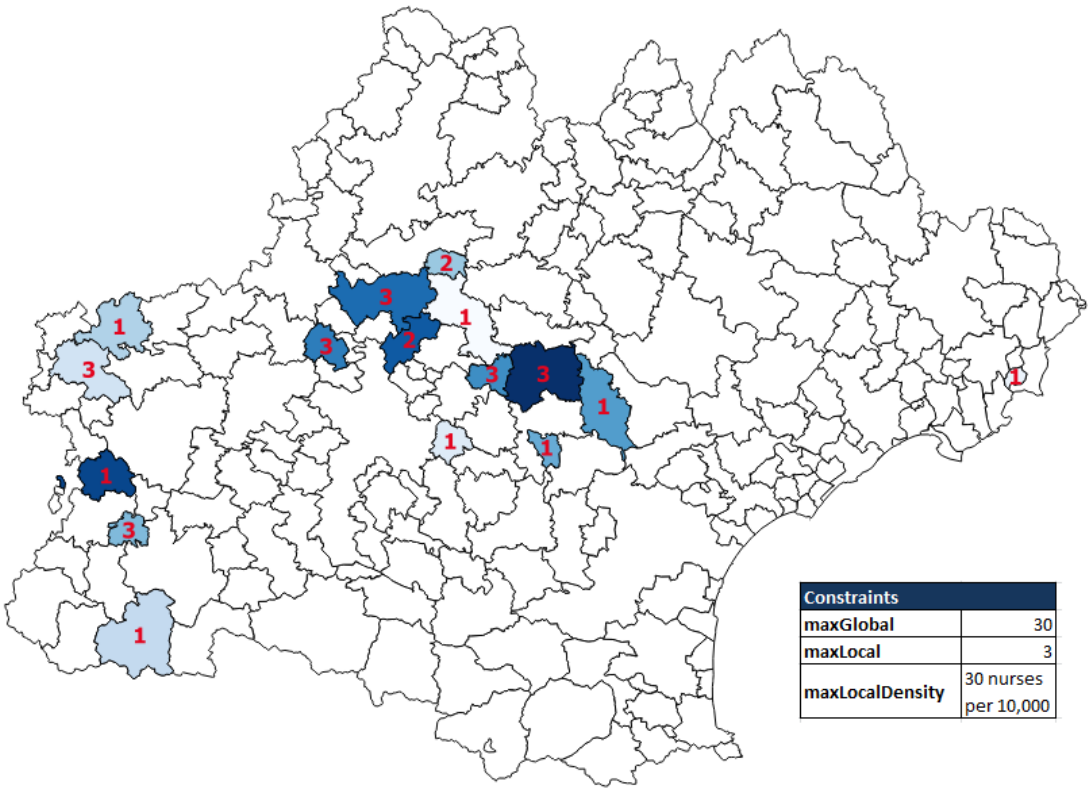


Figure 3.18: BVs to increase nurses and the best number of nurses to add for the biggest reduction of PAH recommended by SVR

Now let us compare our approach with two approaches using simple descriptive statistic methods. The first map (Figure 3.19) indicates top 15 BVs recommended by the actual rates of PAHs with the condition on the densities of nurses. Specifically, the BVs recommended are the ones whose the actual rates of PAHs are the biggest with the condition that the densities of nurses are smaller than 25 nurses per 10,000 inhabitants. Similarity, the other map (Figure 3.20) indicates top 15 BVs recommended by the lowest densities of nurses with the condition that the actual rates of PAHs are higher than 4.5 PAHs per 1,000 inhabitants. As it can be seen through the maps, the BVs selected by approach using SVR are different to the ones selected by the descriptive statistic

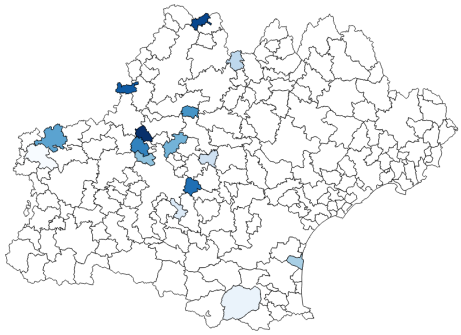


Figure 3.19: BVs to increase nurses recommended by the high rates of PAH

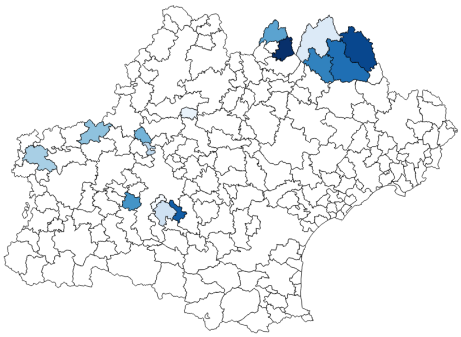


Figure 3.20: BVs to increase nurses recommended by the low densities of nurses

methods.

In addition, as our algorithm also returns the rates of PAH reduction per to-be-added nurse, we can assess the effectiveness of the approach using SVR by comparing it with the two descriptive statistic methods. For example, in the Table 3.7, if we increase 9 nurses (number of nurses - *No Nurses* in Table 3.7), we expect the number of PAHs to be reduced is 6.3 (*No PAHs* in Table 3.7), and therefore the rate of PAH reduction per to-be-added nurse is $6.3/9 = 0.7$ (*Reduction Rate* in Table 3.7)

Table 3.7: PAH reduction per to-be-added nurse by SVR

No Nurses	**No PAHs**	**Reduction Rate**
9	6.3	0.70
15	9.7	0.65
20	12.4	0.62
24	14.4	0.60
30	17.0	0.57

It should be noted that the descriptive statistic methods do not support to compute the rates of PAH reduction per to-be-added nurse. Therefore, for the purpose of comparison the effectiveness of different approaches, we use the reduction numbers of PAHs from the approach using SVR, we can obtain the rates of PAH reduction per to-be-added nurse for the selected BVs as shown in Tables 3.8 and 3.9.

By comparing the results in the Table 3.7 with the results in the other Tables (3.8 and 3.9), we can somehow confirm the effectiveness of the approach using SVR for selecting the BVs to increase nurses.

Table 3.8: PAH reduction per to-be-added nurse recommended by high rates of PAHs

No Nurses	No PAHs	Reduction Rate
9	2.47	0.27
14	5.30	0.38
19	6.09	0.32
24	7.80	0.32
30	9.54	0.32

Table 3.9: PAH reduction per to-be-added nurse recommended by low densities of nurses

No Nurses	No PAHs	Reduction Rate
10	0.42	0.04
16	2.84	0.18
19	3.19	0.17
25	5.29	0.21
30	5.73	0.19

3.5 Conclusions

In this chapter, we have presented our approach of machine learning in improving health care services. In particular, we firstly evaluated the potentials as well as the performances of some common regression methods including multilinear regression, k-nearest neighbors, neural networks, support vector machine for regression (SVR). Secondly, as the most suitable method, SVR has bee extended in our work by integrating the constraints, which are related to the budget (or the maximum number of nurses to be added) and the equality of health care access for the inhabitants in the region regardless of their geographical and socioeconomic situation. Our goal of our work is a decision support system that recommends to the local health authorities for health care service improvement in general and nurse incremental in particular. As the result, our works are not only to select the living areas (fr. Bassins de vie, BVs), but also to recommend the number of to-be-added nurses in each BV for the biggest reduction of the number of potentially avoidable hospitalizations (PAHs).

In addition, our approach is applied to the Occitanie region, but it can be applied to other regions or extended at the national level or even to other countries. Moreover, this approach could be applied to other health care policy issues, such as the reduction of hospital re-admissions or access to innovation. In particular, our approach has led to a start-up project in France.

Although our works are promising, we still have limitations. One of the limitations is that in our opinion, some potential factors of PHAs have not been taken into account. In particular, living conditions related to environment such as pollution and temperature have not been included while it is clear that there are strong impacts of extreme cold and hot temperature (or heatwave) to human health. In other words, extreme temperature could be one potential factor associated with high rates of PAHs. Therefore, our future work is to measure the impact of the extreme temperature to PAHs as well as to include this environmental data in our approach above. However,

while we collected the temperature data measured hourly by sensors at the weather stations, the temperature values are discontinuous (or missing). Therefore we firstly need a reliable missing temperature imputation that is presented in the next chapter.

Chapter 4

Missing temperature imputation: improvement by combining spatial interpolations and time-series models

Contents

4.1 Introduction

In previous chapter, we mentioned that we would like test new determinants of PAHs related to the environment and weather conditions. That is because it is clear that temperature, especially temperature extremes, have negative impacts to human health. For example, the extreme heat (or so called heatwave) that occurred in summer 2003 in France caused about 15,000 more deaths than expected in France (an increase of 55%) [29]. Because of these impacts and also because global warming makes these heatwaves more frequent, there are more and more studies that have been conducted to assess the impacts as well as to search for effective solutions to reduce them. In these researches, the temperature values measured by sensors at weather stations are used as the main data source. However, for many reasons including running out of batteries or losing connections to the stations, the values measured at these stations are sometimes discontinuous. In other words, there are missing values for temperatures measured at the weather stations. On the other hand, the way we treat these missing temperature values can have an impact on the accuracy of the studies. For example, in our work, to measure the impact of extreme hot temperature (or heat waves) to human health in the French context, we need to define the heatwave events at which a difference of 0.5° C can lead to the definition of a heatwave or not [89]. Therefore, a reliable missing data imputation is often needed as one preprocessing step for the temperature data collected at the weather stations.

In the literature, there are many approaches to deal with missing weather temperature data. The approaches can be as simple as ignore the missing values or fill in the missing values with statistical values like the mean, median or mode values or with the values standing just before or after the missing values. Approaches using machine learning methods can also be applied to fill in these missing temperature values. These methods are linear regression [61] or k-nearest neighbors [10] or more complicated approaches like support vector machines [67] or different types of artificial neural networks [1]. However, the temperature data measured at weather stations is spatio-temporal data which has both a spatial component and a temporal component. These components can be exploited in the missing data imputation. More specifically, the spatial component can be exploited in the missing data imputation because there is a correlation between data measured at the same time at nearby stations (so-called spatial autocorrelations). In general, the methods that exploit spatial autocorrelations are called spatial interpolation methods which include Inverse Distance Weighted (IDW), Spline, Kriging, and others. On the other side, the temporal autocorrelation of the spatio-temporal data can be also exploited to fill in the missing values. This temporal autocorrelation exists due to internal structure of the time-series data. This internal structure consists of both seasonality and trend. The seasonality indicates that there is a repetitive, predictable pattern in the value series while the trend tells us the tendency of the value series to increase or decrease over time. The most common method to work with time-series data is the Autoregressive Integrated Moving Average (ARIMA) model. In literature, both the spatial

interpolations and time-series models are used, but separately.

To select the more reliable method in missing temperature imputation, we first compare the quality performance of two different methods representative for both spatial interpolation methods and time-series models. These methods are IDW and ARIMA model respectively. In addition, as both approaches exploit only one different dimension of the spatio-temporal data, we also propose a novel approach that combines these methods to improve the quality performance. The performances of all the methods above are evaluated using the root mean square error (RMSE) between the estimated temperature and the observed temperature at the weather stations. The chapter is organized as follows. Section 2 briefly introduces the IDW method and ARIMA model as well as the dataset and the evaluation method we use for the experiments. Section 3 presents the experimental results the two methods above as well as the method we proposed. Section 4 will be the conclusion.

4.1.1 Dataset

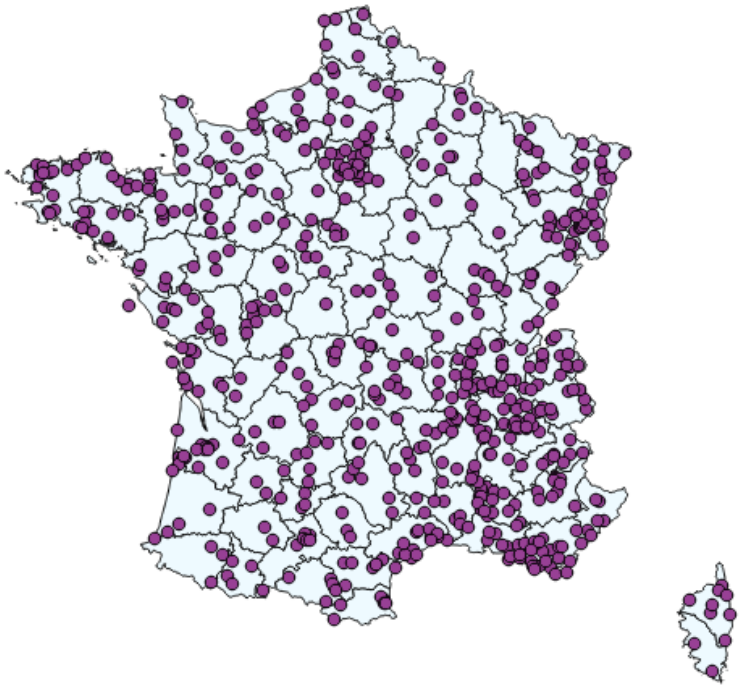


Figure 4.1: Locations of 605 weather stations in Metropolitan France

To conduct the experiment, we collect the temperature data which is measured in Celsius from the website French infoclimat [42], on which, besides the hourly-measured temperature, other weather data such as wind speed or precipitation are also available. More specifically, the temperature data we used were the values hourly-recorded in May 2019 at the weather stations located in Metropolitan France. Moreover, we also conducted data cleaning by following steps:

1. Removing error-recorded values which are higher than 50.

2. Using the absolute z-score of 3 and higher at values measured at the same time (same day, same hour) to mark potential outliers.

3. Manually verifying the potential outliers as error-recorded values by looking at the other values measured at the same station.

In addition, 32 stations on which the number of recorded data over time is small (less than 300 hours over the 744 hours) are also excluded. After the cleaning step, the final number of stations is 605 and their locations are visualized in the map below (Figure 4.1). In addition, with these 605 stations, the percentages of missing values at the stations range from 0% to 51.6% with the median and the mean values are 2.8% and 5.2% respectively.

4.1.2 Evaluation and validation methods

To evaluate the performance of the methods above, we use root mean square error (RMSE) between the estimated temperature and the observed temperature at every weather station of all 605 stations. On the other side, our experiments were validated with the leave-one-out method. More specifically, when we compute the estimated temperature at a time slot at a station (specific hour h, day d, station s), for any approach, we build the training data by removing the observed temperature value at that time slot of that station.

4.2 IDW method and ARIMA model

4.2.1 IDW method

As mentioned in the introduction, the values of temperature measured at nearby stations are correlated to each other. The Inverse Distance Weighted (IDW) method exploits this spatial component of the data. More specifically, the spatial interpolation method estimates an unknown value at a location using the known values at nearby locations. In this method, the distances between the location of the unknown value and the locations of known values are also taken into accounts. In particular, this method is based on the principle that the greater the distance, the less influence the known values have on the estimated value. This principle forms the following

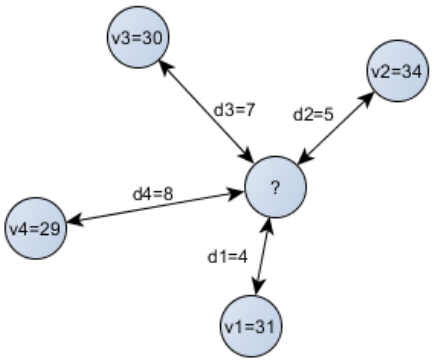
formula to estimate the unknown value, represented by v:

$$v = \frac{\sum_{i=1}^{n}(\frac{v_i}{d_i^p})}{\sum_{i=1}^{n}(\frac{1}{d_i^p})} \tag{4.1}$$

In formula (4.1), v_i are the known values, d_i are the distances from the unknown values to the known value i. Distances are usually taken into accounts with a power p of 1 or 2 for square distances.

For example, we need to estimate the unknown value representative, marked with a question mark, with n=4, denoting the neighboring known values as shown in figure below. These known values are v1=31, v2=34, v3=30, and v4=29. We also know the geographic distances from the unknown location to the known locations which are d1=4, d2=5, d3=7, and d4=8 respectively. If we select p=2 then the unknown value v will be:

$$v = \frac{\sum_{i=1}^{4}(\frac{v_i}{d_i^2})}{\sum_{i=1}^{4}(\frac{1}{d_i^2})} = \frac{\frac{31}{4^2} + \frac{34}{5^2} + \frac{30}{7^2} + \frac{29}{8^2}}{\frac{1}{4^2} + \frac{1}{5^2} + \frac{1}{7^2} + \frac{1}{8^2}}$$



As IDW is simple and straightforward, it and its variants have been applied in a wide range of applications, especially related to environmental data such as estimation of spatial variability of rainfall [46] or in spatial mapping of coastal water quality patterns [83]. Related to our missing temperature imputation work, we fill the missing temperature values of a station by using the known values measured at the same time at the nearby stations.

4.2.2 ARIMA model

The method in the previous section is exploiting the spatial component of the hourly measured temperature data. This section is about a method exploiting the temporal component of the spatio-temporal data. Since the temperature is measured (observed) hourly at the stations, this data is a univariate time series. This specific univariate time series data has a seasonality, since the temperature is lower at night and higher when it gets close to mid-day. There might be also a trend at the beginning and ending of a season. For instance, the temperature gets hotter day

after day at the beginning of summer. One common method to deal with time series data could be Box-Jenkins ARMA model [15]. This ARMA model is a combination of the Autoregressive (AR) model and Moving Average (MA) model.

4.2.2.1 Autoregressive (AR) model

The first part of the Box-Jenkins ARMA model is Autoregressive (AR) model. In this model, the time-series value at t denoted as X_t can be estimated using previous values through a linear regression model:

$$X_t = b + \sum_{i=1}^{p} \Phi_i X_{t-i} \tag{4.2}$$

In which:
- X_i are the time series values
- b is the intercept
- Φ_i are the parameters of the model
- p is called the order of the model

For example, the following AR(1) model, here the number 1 indicates the order of the model, fits with a given training dataset.

$$X_t = 2 + 4X_{t-1} \tag{4.3}$$

Then, to predict the value at $t = 10$, we use the value at $t = 9$. For example, if $X_9 = 5$, then, by placing these numbers to formula 4.3, we have:

$$X_{10} = 2 + 4X_9 = 2 + 4 * 5 = 22$$

4.2.2.2 Moving Average (MA) model

The other part of Box-Jenkins ARMA model is Moving Average (MA) model. In MA model, the time-series value X_t is predicted using the mean value and the previous errors through the formula (4.5) below.

$$X_t = \mu + \sum_{i=1}^{q} \theta_i \xi_{t-i} \tag{4.4}$$

In which:
- X_t is the predicted time-series value
- μ is the mean value of the series

- θ_i are the parameters of the model
- ξ_i are the errors of previous predictions
- q is the order of the model.

For example, from a given dataset, a the following MA(1) model, the number 1 here also indicates the order of the model, is found a the best fit.

$$X_t = 10 + 0.5\xi_{t-i} \tag{4.5}$$

As mentioned, 10 in formula 4.5 is the mean (average) value of the time-series. Also, given that at $t = 9$, the observed value $X_9 = 5$ and the predicted value $\hat{X}_9 = 4$. That means the error at $t = 9$, $\xi_9 = 5 - 4 = 1$. By placing these numbers into formula 4.5, we have the predicted value at $t = 10$

$$X_{10} = 10 + 0.5\xi_9 = 10 + 0.5 * 1 = 10.5$$

4.2.2.3 ARMA model

When combining AR model and MA model, we have ARMA model and the new formula to estimate X_t is:

$$X_t = b + \sum_{i=1}^{p} \Phi_i X_{t-i} + \sum_{i=1}^{q} \theta_i \xi_{t-i} \tag{4.6}$$

4.2.2.4 ARIMA model

Moreover, the ARMA models above are supposed to run on stationary time series or more specifically, the time series should have properties that are the mean, variance and autocorrelation structure do not change over time. That is not always the case. Therefore, to achieve stationary series before applying the ARMA models, it is recommended to transform from the non-stationary series X_t to the new one Z_t by following:

$$Z_t = X_t - X_{t-1} \tag{4.7}$$

This step is also called differencing. For example, given that we have a time-series dataset X = {1, 2, 3, 4, 5, 6}, after a differencing step, we achieve a new time-series dataset Z = {2-1, 3-2, 4-3, 5-4, 6-5} = {1, 1, 1, 1, 1}.

This differencing can be repeated for several times until achieving stationary series. This additional step adds letter I standing for Integrated to ARMA model so that it becomes ARIMA model.

Furthermore, ARIMA models are mathematically written as $ARIMA(p, d, q)$ in which p, and q are the orders of AR models and MA models respectively, and d is the number of times we need to take the differencing step to achieve stationary series.

In application, the ARIMA model can be applied in many fields such as energy [75], real-estate [68], or in health science [51]. In our work, we apply ARIMA model to fill in the missing temperature values at each weather station using the time-series temperature data measured at that station overtime.

4.3 Experimental results and an improvement approach

4.3.1 Experiment implementations

Our experiments are conducted using R language and available libraries to apply IDW method and ARIMA models.

For the IDW method, we use the "idw" function of the "gstat" package [64, 35] using 2 as the power p. In particular, to estimate the temperature at hour h of day d at station s, we apply $gstat :: idw$ function with the training data is the observed data hourly-recorded at hour h of day d at all the other stations except station s. The details of the process is described in the flowchart provided (Figure 4.2). Note that, as mentioned in the dataset section, the observed data we used are recorded in May 2019, so that the number of days (variable d) of the month is 31, while the hour (variable h) is from 0 to 23 and the number of stations (variable s) is 605.

On the other side, to measure the performance of the ARIMA model on the time series data that was hourly-recorded at all 605 weather stations, we use method "na.kalman" function of the "imputeTS" package [57]. Particularly, for any station s, we apply the $imputeTS :: na.kalman$ on the time-series data recorded at that station to estimate the temperatures. More specifically, to estimate the temperature at hour h of day d at station s, we build the training data by removing the observed temperature recorded at hour h of day d from the observed temperatures of the station s before applying the ARIMA model. The details are presented in the flowchart provided in figure 4.3. Note that, In this flowchart, when we apply "na.kalman" function, to simplify the work, we choose "auto.arima" model or in other words we let the package itself select the values for parameter set (p, d, q).

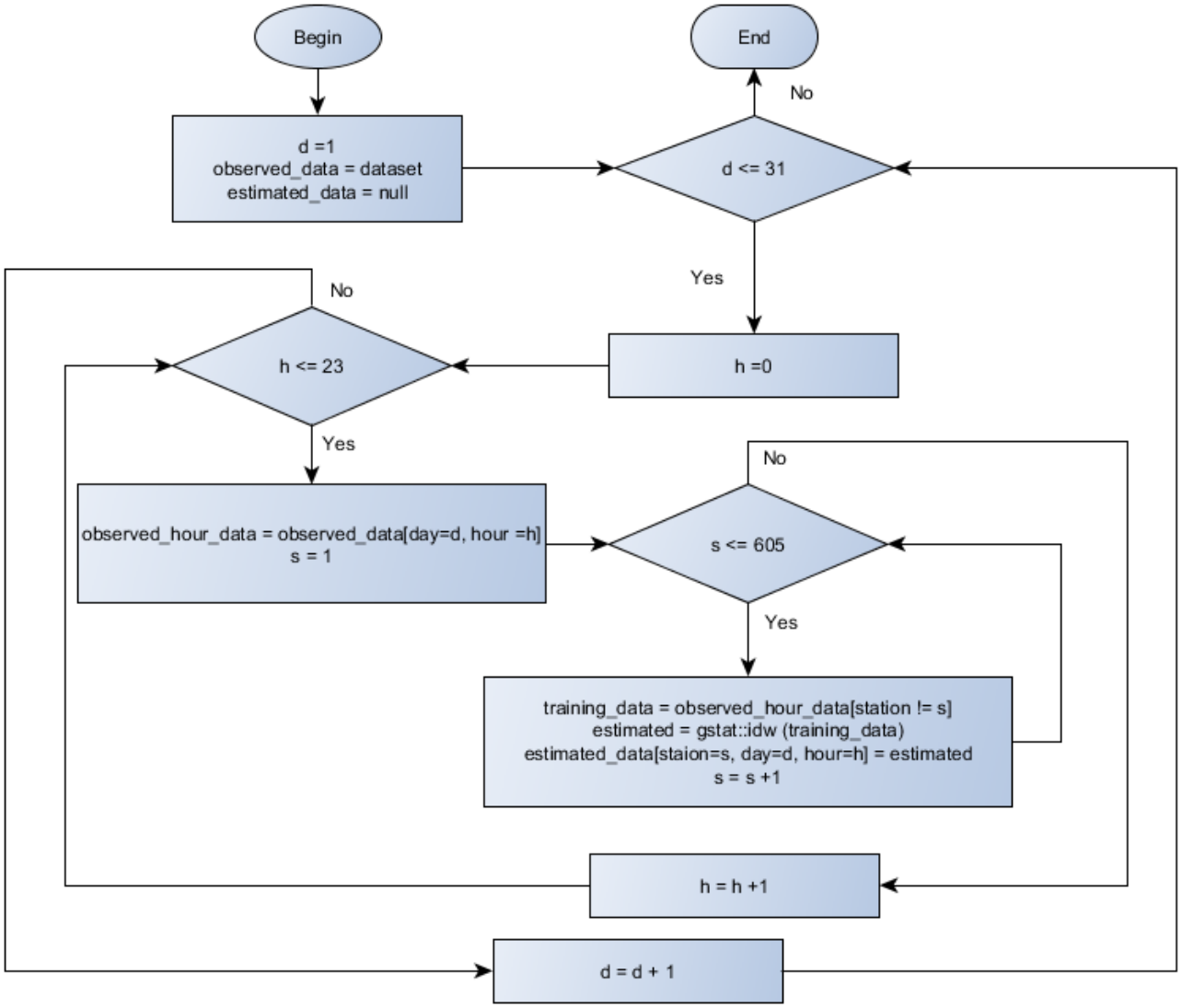


Figure 4.2: Programming flowchart of experiment using IDW. **observed_data**: temperatures hourly-recorded at 605 weather stations in May 2019. **observed_hour_data**: temperatures recorded at hour h of day d at 605 weather stations. **training_data**: temperatures recorded at hour h of day d at all the stations except station s. **estimated**: temperature estimated by *gstat* :: *idw* method using the training_data. **estimated_data**: estimated temperatures of all 605 weather stations in May 2019

4.3.2 Experimental results

For both the approaches above, after having all the estimated values, we compute RMSEs between them and the observed values of all 605 weather stations. Particularly, for the IDW method, the RMSEs have a median value of 1.26 and vary between 0.3 and 11.56 while for the ARIMA model, the corresponding values are 0.75, 0.38, and 1.51 respectively. In addition to these numbers, the boxplots [Figure 4.4] clearly show that the RMSEs of the ARIMA model (labeled ARIMA in

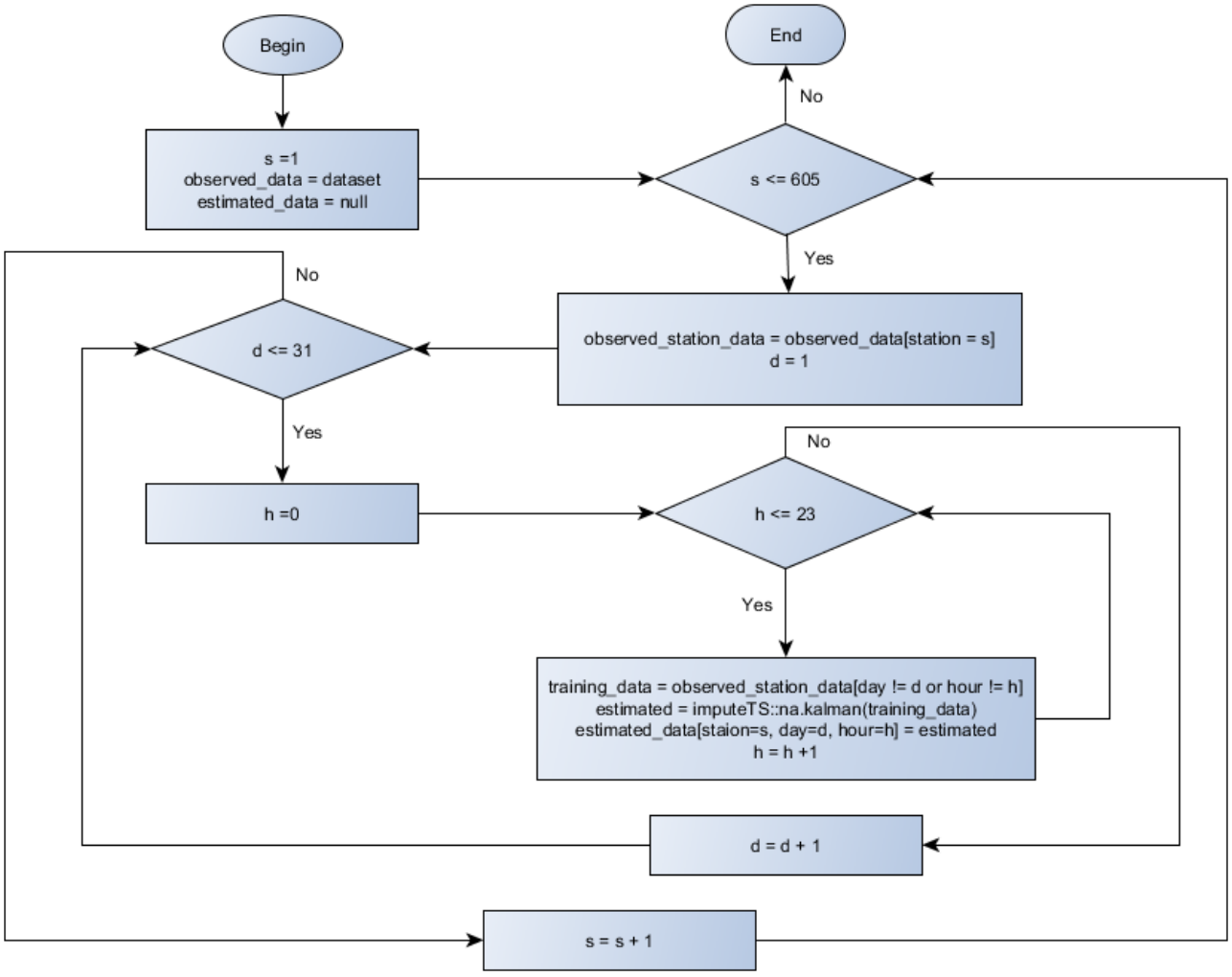


Figure 4.3: Programming flowchart of experiment using ARIMA model. **observed_data**: temperatures hourly-recorded at 605 weather stations in May 2019. **observed_station_data**: temperatures hourly-recorded at station s in May 2019. **training_data**: temperatures recorded at station s in May 2019 except the one recorded at hour h of day d. **estimated**: temperature estimated by *imputeTS* :: *na.kalman* method using the training_data. **estimated_data**: estimated temperatures of all 605 weather stations in May 2019

figure 4.4) are smaller than the RMSEs of the IDW method (labeled IDW in figure 4.4). More particularly, at 92,1% (557 out of 605) stations, the RMSEs of ARIMA model are smaller than the RMSEs of IDW method. Furthermore, with the ARIMA model, 98.3% (595 out of 605) stations have RMSEs smaller than 1 while with the IDW method, the corresponding number is only 30% (181 over 605). In conclusion, we can say that in missing temperature imputation, the approach using the ARIMA model generally performs better than the approach using the IDW method.

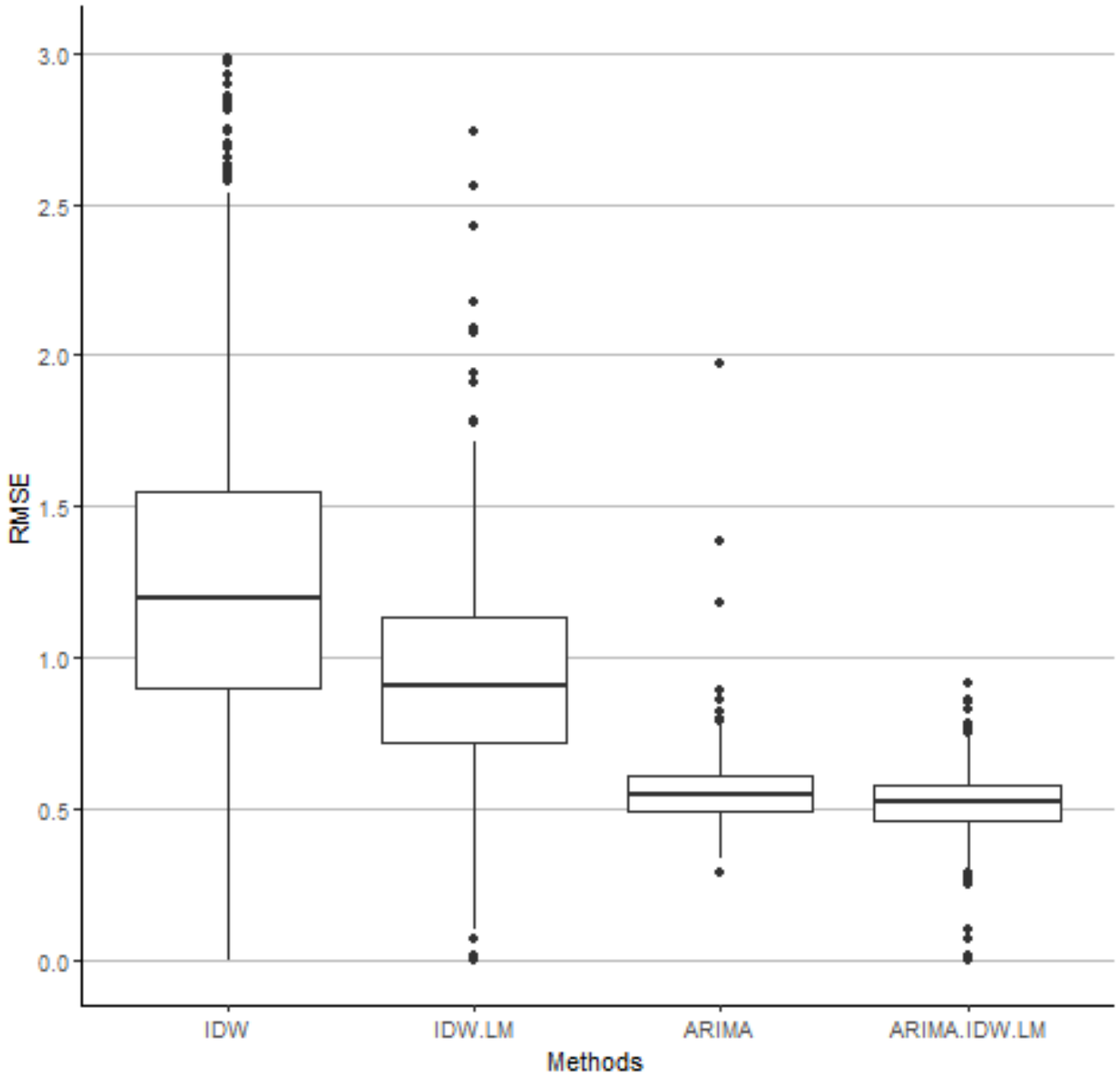


Figure 4.4: Quality performance comparison between different approaches for missing temperature imputation. **RMSE**: Root mean square error between estimated temperatures and observed temperatures at 605 weather stations. **IDW**: Results from IDW method. **IDW.LM**: Results from the linear regression model using the results from IDW method as independent variables. **ARIMA**: Results from ARIMA model. **ARIMA.IDW.LM**: Results from the linear regression model using the results from both IDW method and ARIMA model as independent variables.

4.3.3 Possible improvement approach

As mentioned in the introduction, both approaches have limitations as each of them exploits only one dimension of the spatio-temporal data, which is either the spatial component or temporal component. Therefore, to improve the performance further, we developed an integrated approach that combines the strengths of both approaches to tackle the spatio temporal dimensions together. Specifically, in our approach, we firstly consider the estimated values by the IDW method as an input of an additional machine learning method. As it can be seen at the boxplots of RMSEs using the IDW method (Figure 4.4), we have high values for RMSEs. One reason is that when we apply IDW method directly, the other conditions such as the elevations of stations are not taken into accounts. Moreover, after examining the results, we have found that the estimated values and the observed values are highly correlated. In particular, the pearson correlation values between the estimated values and the observed values at 605 weather stations have the min and the mean values of 0.69 and 0.97 respectively. Therefore, we choose linear regression model as an additional machine learning method to improve the performance in missing temperature imputation. More specifically, to estimate the temperatures, the estimated values resulting from the IDW method are used as the independent variables in this linear model. In addition, the observed times of the day should be added into the linear model because there are patterns between the temperature and the time of the day (temperatures are colder at night and warmer at noon). However, instead of using directly these hourly values (from 0 to 23), we use the absolute values of hours after subtracting the value 12 (thus the new values are all between 0 and 12) as the additional independent variables. Mathematically, at each weather station, the new estimated values is predicted by the formula (4.8).

$$estimate = b_0 + b_1 * X_1 + b_2 * X_2 \tag{4.8}$$

In formula 4.8,
- X_1 are the absolute values of hours after subtracting the value 12
- X_2 are the estimated values resulting from the IDW method

As shown by the boxplots of RMSEs (Figure 4.4), the performance of the new approach (labeled IDW.LM) is much better than the original IDW result (labeled IDW). More specifically, with the new approach, the new RMSEs have a median value of 0.94 and vary between 0.26 and 3.05 compared with the ones of original IDW result which are 1.26, 0.3 and 11.56 respectively.

Continually, we also add the estimated values of the ARIMA model as a new additional independent variables to the linear regression model above. Or in other words, the formula (4.8) becomes formula (4.9) below.

$$estimate = b_0 + b_1 * X_1 + b_2 * X_2 + b_3 * X_3 \tag{4.9}$$

In which,

- X_1 are the absolute values of hours after subtracting the value 12
- X_2 are the estimated values resulting from the IDW method
- X_3 are the estimated values resulting from the ARIMA model

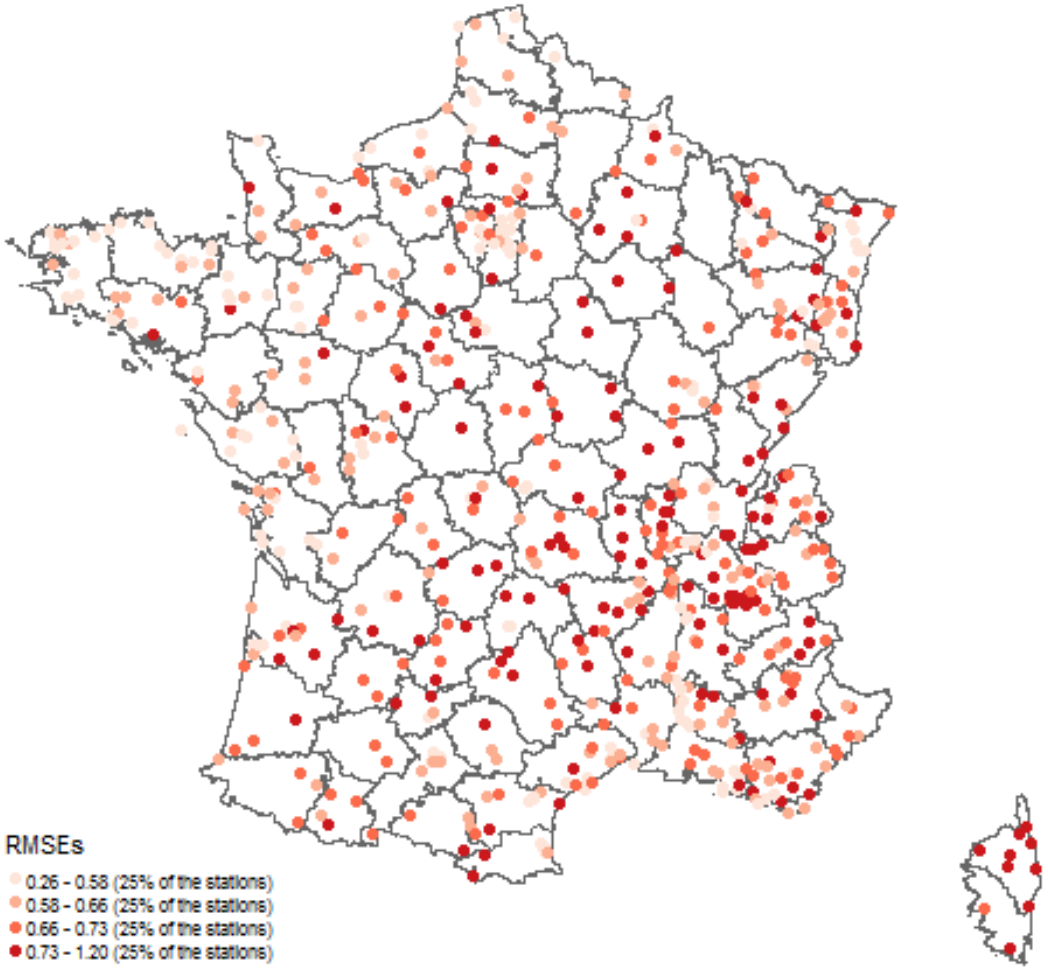


Figure 4.5: The root mean square errors (RMSE) of all the stations applying our approach.

The results of this approach, as shown in figure 4.4 (labeled ARIMA.IDW.LM), have been improved. Particularly, compared with the orginal IDW method, the new approach performs better at all 605 stations. On the other hand, compared with the ARIMA model, the new approach performs better at 604 over 605 stations. For the only one station left, both the new approach and the ARIMA model return the same RMSE which is 1.03. More specifically, with the new approach, the first quartile, the median and the third quartile of the RMSE values are 0.58, 0.66 and 0.73 respectively while the corresponding values resulting from the ARIMA model are 0.68, 0.75 and 0.82 respectively. Furthermore, the map (Figure 4.5) shows the RMSE values of this new approach of all the weather stations. Lastly, to verify whether or not there are patterns of the performance of our approach related to the locations of the weather stations, we use GeoDa tool [5] to measure the global spatial autocorrelation of the RMSE values. With the spatial weight matrix that is built with the 5 nearest neighbors and the weights are the inverse distances with the

power of 2, the tool returns the pseudo p-value of 0.345. This high pseudo p-value statically means that we are failed to reject the randomness of the quality performance on the weather stations in term of their spatial locations.

4.4 Conclusion

In this chapter, we have presented two different approaches using spatial interpolation methods and time series models for missing temperature imputation. In particular, we measured the quality performance of the IDW method and the ARIMA model respectively. To conduct the experiments, we collected the temperature data that was hourly-recorded in May 2019 from more than 600 weather stations in Metropolitan France. The results show that the ARIMA model performs much better in this type of application. We also bring a new idea to improve the performance. Specifically, instead of applying directly the IDW method or the ARIMA model, we firstly compute the estimated values by these methods and then use them as the input variables of an additional machine learning method. With a simple linear regression model, the performance has been improved. Part of future work includes evaluating other machine learning methods for regression such as neural networks for regression, and evaluate their potential added value in missing temperature data imputation. In addition, it is possible to apply our approach to other environmental spatio-temporal missing data such as air pollutants.

Now, we are back to our main work related to PAHs. Although our proposed approach for missing temperature imputation is more reliable, the goal of measuring the impact of the extreme temperature to PAHs has not been obtained because of the limitation of the PAH dataset we have. Particularly, in the PAH dataset, the hospital admission dates of PAH patients are monthly instead of daily. On the other side, the lag of the extreme temperature impacts to human health could be for only several days. Therefore, we keep the work of measuring the impacts of the extreme temperature to PAHs for the future when we can obtain the more detail PAH data.

On the other side, as mentioned in chapter 1, improving the coordination between the health care providers could lead to the reduction of PAHs. Therefore, we approach of grouping hospitals into communities so that the hospitals within the communities could share medical records and therefore provide efficient and high-quality treatments to the patients. This work is presented in the next chaper.

Chapter 5

Graph clustering approaches for hospital communities

Contents

5.1 Introduction

In chapter 3, we have presented an approach of machine learning in building a decision support system for the reduction of PAHs. In particular, we extended support vector machine for regression to select the geographic areas and the number of nurses to be added for the highest reduction in term of number of PAHs. In this chapter, we present another approach of machine learning that could lead to the reduction of PAHs.

Particularly, it is a noticeable fact that patients do not visit the same hospitals every time. There are many reasons for that. For example, patients have changed addresses, they are not happy with the service of the previous hospital, or they need to seek specialized care in a tertiary hospital. In such cases, it is clear that the treatment would be more efficient and the risk to patients' health could be eliminated or reduced if the later hospitals were able to access the medical records of the patients at the previous hospitals. In other words, there is a need to allow information technology systems to share medical records among hospitals. However, it is neither necessary nor practical for all hospitals in France to be grouped as one because it would be costly while some hospitals will never share any patient. Therefore, health authorities are interested in building hospital communities so that medical records can be shared among the hospitals in those communities.

In the meantime, in the French context, public hospitals are already grouped into regional hospital groups (fr. Groupements hospitaliers de territoire - GHT). As these GHTs are proposed by the regional health agencies (Agences régionales de santé - ARS), these GHTs have limitations due to the administrative boundaries. In addition, in these GHT, private hospitals are not included. Therefore, a scientific approach at the national level for all hospitals types is of high interest to hospitals, health authorities as well as health scientific communities.

On the other side, in France, a national hospital discharge database (fr. Programme de Médicalisation des Systèmes d'Information - PMSI) is available[1]. This PMSI database stores discharge data from all French public and private hospitals. In particular, this database contains a record for each acute inpatient stay[2] [13]. In other words, the patients' pathway can be described. For example, a patient P has the pathway such as $h1 \rightarrow h2 \rightarrow h2 \rightarrow h1 \rightarrow h2$ in which h1 and h2 are the hospitals the patient has gone to.

To group hospitals into communities we could use graph clustering methods. Particularly, in our approach, patients' flows between hospitals are represented by an undirected graph in which the nodes represent hospitals and the edges represent the size of patient flows (Figure 5.1). For

[1] Upon registration with and payment to a habilitated provider, or through collaboration with a French university hospital health information management department

[2] There are about 25 million records per year

example, the pathway of patient P above would be plus 3 (2 for h1 - h2 and 1 for h2 - h1) for the edge between h1 and h2 on the undirected graph.

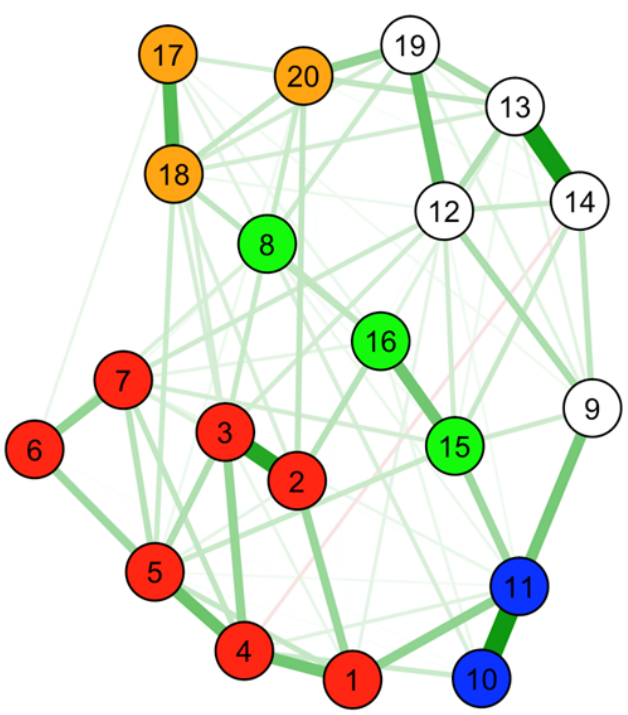


Figure 5.1: Approach using graph clustering approaches for hospital communities

Based on the undirected graph, the goal of our work is to group hospitals into communities, for example, presented in different colors in figure 5.1. To achieve this goal, two different graph clustering methods, spectral clustering and Louvain in particular, are implemented and their performances in terms of quality on our dataset are compared. Particularly, multi criteria are used to evaluate the performances. In addition, in our works, we need to try several options of grouping hospitals into the communities. One option is that each final clusters must contain a public University Hospital (fr. Centre Hospitalier Universitaire - CHU). These constraints are added into our work by customizing the graph clustering method. Our work is presented in this chapter which is organized as follows. The introduction section is to deliver the information about the dataset as well as the evaluation method to be used. Section 2 briefly introduces two graph clustering methods to be applied. Section 3 presents the experimental results together with the discussions.

5.1.1 Dataset

As mentioned in the introduction section, our work is based on the PMSI database system which keeps record of every hospitalization of any patient at both public and private hospitals. This

database system allows us to extract the flows of patients between hospitals. In particular, the patient flows of hospitalizations in three continuous years, 2016 to 2018, are extracted. This dataset contains a total of 1,777 hospitals, either public or private, in France. Among these hospitals, the total number of times patients changed hospitals is 13,094,068. Other descriptive information of the dataset is provided below (Table 5.1).

Table 5.1: Descriptive information of the graph presenting patient flow dataset

Number of nodes	1,777
Number of edges	290,707
Max value of weights	34,248
Min value of weights	1
Mode value of weights	1
Median value of weights	2
Total weight	13,094,068

5.1.2 Evaluation for hospital communities

The modularity value is one criterion that is used to evaluate the graph clustering methods. The modularity has values ranging from -1 to 1 and the higher values indicate the better results in graph clustering. In section 5.2.3, we explain in details how the modularity values are computed. On the other side, since we are grouping the hospitals into the communities for the purpose of effectively sharing medical records, we also use the of percentage that the previous hospitals located outside the communities to evaluate the efficient of the methods. This percentage value indicates the rate the hospitals cannot access to the patients' medical record from previous hospitalization after obtaining the communities. Therefore, the methods return smaller values for this percentage are the better methods. There is also the fact that the number of hospitals in each community has the impact to these two values above. For example, a community structure that has one very big community containing almost all the hospitals while other communities contain only one hospital naturally gives the highest values for the percentage value above. Therefore, the balance in term of number of hospitals in each communities should be taken into account when we conduct the evaluation.

5.2 Graph clustering methods

Graph clustering is also known as graph partitioning or community detection has been studied and applied in many domains including social network [81], chemical informatics [47], computer vision [77], Many graph clustering methods have been proposed, including random walk based methods [39], spectral clustering [92, 37], modularity based methods [91]. In our work, we consider two approaches which are the spectral clustering method and a modularity-based or Louvain method in paricular. In this section, we briefly present the details of these two method.

5.2.1 Graph notation

A graph can be presented as G = (V, E) where V = $\{v_1, v_2, ..., v_n\}$, is a set of nodes or vertices and E is a set of edges which are two-element subsets of V like $\{v_i, v_j\}$, with $v_i, v_j \in$ V. In the case of weighted graph, each edge carries a non-negative weight $w_{ij} > 0$. A matrix W = (w_{ij}) where i, j = 1, ...,n is called *weight matrix*. Furthermore, if the graph is an undirect graph then $w_{ij} = w_{ji}$.

On the other side, a node $v_i \in$ V has a degree d_i which is defined by:

$$d_i = \sum_{j=1}^{n} w_{ij}$$

A diagonal matrix D with the degrees $d_1, ..., d_n$ on the diagonal is *degree matrix*

For example, the undirected weighted graph below (Figure 5.2) has the *weight matrix* W and the *degree matrix* D as follows:

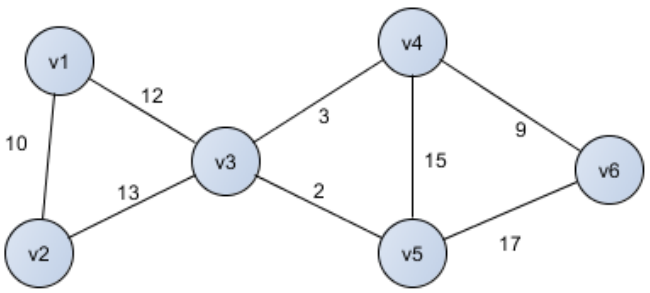


Figure 5.2: An example undirected weighted graph

$$\mathrm{W} = \begin{pmatrix} 0 & 10 & 12 & 0 & 0 & 0 \\ 10 & 0 & 13 & 0 & 0 & 0 \\ 12 & 13 & 0 & 3 & 2 & 0 \\ 0 & 0 & 3 & 0 & 15 & 9 \\ 0 & 0 & 2 & 15 & 0 & 17 \\ 0 & 0 & 0 & 9 & 17 & 0 \end{pmatrix}$$

$$\mathrm{D} = \begin{pmatrix} 22 & 0 & 0 & 0 & 0 & 0 \\ 0 & 23 & 0 & 0 & 0 & 0 \\ 0 & 0 & 30 & 0 & 0 & 0 \\ 0 & 0 & 0 & 27 & 0 & 0 \\ 0 & 0 & 0 & 0 & 34 & 0 \\ 0 & 0 & 0 & 0 & 0 & 26 \end{pmatrix}$$

Moreover, graph clustering is a process of partitioning a graph into sub graphs. Mathematically, if we split the graph G above into K sub graphs whose sets of the nodes are $A_1, ..., A_k$, then we have $A_1 \cup A_2 \cup ...A_k$ = V and $A_i \cap A_j = \emptyset$ with any $i \neq j$ and i, j = 1, ..., K. To measure the qualities of the graph clustering, we define:

- The total weights to be lost by a pair of the sub graphs, denoted as *cut*(A_i, A_j)

$$cut(A_i, A_j) = \sum_{v_i \in A_i, v_j \in A_j} w_{ij}$$

- The size of each sub graph. The size of a graph A can be measured by the number of the nodes denoted as $|A|$ or by the weights of its edges denoted as *vol(A)*.

$$|A| = \text{the number of nodes in A.}$$
$$vol(A) = \sum d_i \text{ where } d_i \text{ is the degree of node } i \text{ in A.}$$

As an example, if we cut the example graph above into two sub graphs presented by $A_1 = \{v_1, v_2\}$ and $A_2 = \{v_3, v_4, v_5, v_6\}$ (Figure 5.2), then we have:

$$|A_1| = 2 \text{ and } |A_2| = 4$$

$$cut(A_1, A_2) = (12 + 13) = 25$$

Furthermore, in the cases that the number of the sub graphs, K = 2 as the example above, A_1, A_2 can be presented as A and $\bar{A}$ where $\bar{A}$ = V $-$ A denotes the complement of A in V. Then *cut*$(A, \bar{A})$ = *cut*$(\bar{A}, A)$ is used to measure total weights of the edges escaping from A.

5.2.2 Spectral clustering

5.2.2.1 Graph cut

Given a weighted graph like the example one above (Figure 5.2), the goal of clustering is to cut the graph into different sub graphs so that the edges between the sub graphs have the low weights and the edges within these sub graphs have high weights. Therefore, we can formalize the graph clustering problem as an optimization problem. More specifically, if we want to cut G into K sub graphs, our work is to minimize the quantity.

$$cut(A_1, A_2, ..., A_k) = \sum_{i=1}^{K} cut(A_i, \bar{A}_i)$$

This problem is therefore called mincut problem which can be solved efficiently [84]. As an example, with K=2, the mincut problem of the example graph above is solved with the following solution (Figure 5.3).

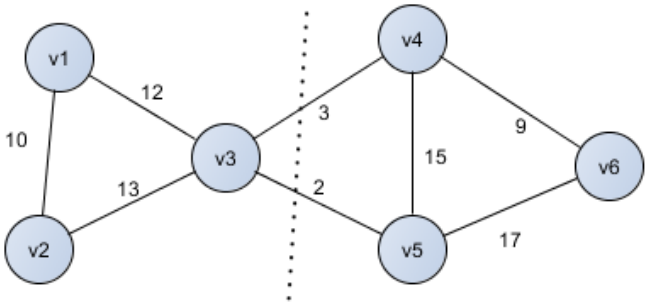


Figure 5.3: An example mincut solution (K = 2)

The mincut solution to the example above (Figure 5.3) is ideal since the minimun value of $cut(A_1, A_2)$, which is (2+3) = 5, splits the graph into two sub graphs that each of them has 3 nodes. However, in many real cases, the mincut solution separates just one node from the rest of the graph. Therefore, we need a solution to keep the number of nodes of each sub graph "reasonably large". The possible approaches are to take into account the size of the sub graphs in the cost function above. Correspondingly to two ways to consider the size of a graph, there are two approaches, RatioCut [36] and normalized cut (Ncut) [77]. In RatioCut, the size of a subset A of a graph is measured by its number of the nodes or $|A|$ above while in Ncut the size is measured by the weights of its edges or vol(A) above.

$$RatioCut(A_1, A_2, ..., A_k) = \sum_{i=1}^{K} \frac{cut(A_i, \bar{A}_i)}{|A_i|}$$

$$Ncut(A_1, A_2, ..., A_k) = \sum_{i=1}^{K} \frac{cut(A_i, \bar{A}_i)}{vol(A_i)}$$

Unfortunately, solving these problems are NP-hard problems [94]. However, spectral clustering is a way to solve relaxed versions of these problems [92, 52].

5.2.2.2 Graph Laplacian matrices and spectral clustering method

As mentioned in the previous section, spectral clustering can be used to solve the relaxed versions of the optimization of the cut-based graph clustering. More specifically, a matrix form can be used to express the optimization measures and the spectrum (eigenvectors) of this matrix can be used to obtain the final clusters (sub graphs) [52]. The matrix are Laplacian matrices L that the unnormalized version is formed by the following equation 5.1.

$$L = D - W \tag{5.1}$$

In equation 5.1, D and W are the *degree matrix* and *weight matrix* mentioned in the section 5.2.1. Computing the unnormalized Laplacian L by the equation 5.1 is the first step of the spectral clustering method to solve the relax version of the mincut problem. The next step is to compute eigenvectors V and eigenvalues λ of that matrix by solving the equation 5.2

$$LV = \lambda V \tag{5.2}$$

Solving equation 5.2 normally returns several eigenvalues λ. If we order these values, then the first eigenvalue will be 0 and the second eigenvalue which is called the Fiedler value. The eigenvector corresponds to Fiedler value is also called Fiedler vector. In the cases that we separate the graph into two sub graphs (K = 2), this Fiedler vector will be used directly. More specifically, since the Fiedler vector will have the positive values and negative values, the original graph can be cut into two sub graphs by putting the nodes corresponding to positive values to one sub graph and the nodes corresponding to negative values to the other sub graph.

To demonstrate the steps, we use the example graph above (Figure 5.2). At first, we compute unnormalized Laplacian matrix

$$\text{L = D - W = } \begin{pmatrix} 22 & 0 & 0 & 0 & 0 & 0 \\ 0 & 23 & 0 & 0 & 0 & 0 \\ 0 & 0 & 30 & 0 & 0 & 0 \\ 0 & 0 & 0 & 27 & 0 & 0 \\ 0 & 0 & 0 & 0 & 34 & 0 \\ 0 & 0 & 0 & 0 & 0 & 26 \end{pmatrix} - \begin{pmatrix} 0 & 10 & 12 & 0 & 0 & 0 \\ 10 & 0 & 13 & 0 & 0 & 0 \\ 12 & 13 & 0 & 3 & 2 & 0 \\ 0 & 0 & 3 & 0 & 15 & 9 \\ 0 & 0 & 2 & 15 & 0 & 17 \\ 0 & 0 & 0 & 9 & 17 & 0 \end{pmatrix}$$

$$\text{L} = \begin{pmatrix} 22 & -10 & -12 & 0 & 0 & 0 \\ -10 & 23 & -13 & 0 & 0 & 0 \\ -12 & -13 & 30 & -3 & -2 & 0 \\ 0 & 0 & -3 & 27 & -15 & -9 \\ 0 & 0 & -2 & -15 & 34 & -17 \\ 0 & 0 & 0 & -9 & -17 & 26 \end{pmatrix}$$

The second step, solving equation $LV = \lambda V$ returns pairs of eigenvalue and eigenvector as follows (Table 5.2).

Table 5.2: Eigenvalues and eigenvectors of the example Laplacian matrix

Eigenvalues	**Eigenvectors**
0.00	(0.408, 0.408, 0.408, 0.408, 0.408, 0.408)
2.96	(0.443, 0.439, 0.337, -0.374, -0.402, -0.443)
32.40	(-0.743, 0.659, 0.095, 0.045, -0.004, -0.051)
34.99	(0.054, 0.191, -0.218, -0.704, 0.032, 0.645)
41.73	(-0.286, -0.412, 0.813, -0.266, 0.021, 0.130)
49.91	(0.015, 0.0189, -0.050, -0.353, 0.819, -0.449)

The Fiedler value and Fiedler vector corresponding to the reults (Table 5.2) are 2.96 and (0.443, 0.439, 0.337, -0.374, -0.402, -0.443) respectively. By putting the nodes corresponding to positive values to one graph A_1 and the nodes corresponding to negative values to the other graphs A_2, A_1 will contains $\{v_1, v_2, v_3\}$ and A_2 will have $\{v_4, v_5, v_6\}$ as its nodes. This is the mincut solution we mentioned above (Figure 5.3).

On the other hand, since the solution above can be applied in the case that number of sub graphs or K = 2, for the general cases including both K = 2 and K > 2, there are several more steps after computing eigenvectors. Particularly, the next step is to build a matrix M that has K columns which are the first K eigenvectors. For the example above, in the case K = 3, the matrix M will be:

$$\mathrm{M} = \begin{pmatrix} 0.408 & 0.443 & -0.743 \\ 0.408 & 0.439 & 0.659 \\ 0.408 & 0.337 & 0.095 \\ 0.408 & -0.374 & 0.045 \\ 0.408 & -0.402 & -0.004 \\ 0.408 & -0.443 & -0.051 \end{pmatrix}$$

In the matrix M above, the orders of the graph nodes are presented by the order of the rows of the matrix. Therefore, the final step is to perform k-means algorithm to cluster the rows of the matrix into K clusters. Returning to the example matrix M (K = 3) above, k-means algorithm returns the solution that A_1 contains $\{v_1\}$, A_2 contains $\{v_2, v_3\}$, and A_3 contains $\{v_4, v_5, v_6\}$ (Figure 5.4)

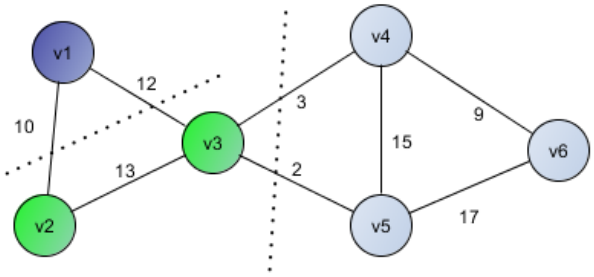


Figure 5.4: An example of using spectral clustering to cluster a graph (K = 3)

To sum up, the spectral clustering can be used to cluster a graph into K sub graphs by following the steps below.

Data: W and D are the weight matrix and degree matrix of graph G
Result: Cluster graph G to K sub graphs

$L = D - W$
Compute eigenvectors V and eigenvalues λ of L
Order eigenvectors V by eigenvalues λ
Building matrix M that has K columns which are the first K eigenvectors V
Perform k-means algorithm on M to K clusters

Algorithm 7: Spectral clustering algorithm

However, algorithm 7 does not take into account the sizes of the sub graphs (or *mincut* solution). Therefore, normalized Laplacian matrices, which can be L_{rw} or L_{sym}, are used to replace the unnormalized Laplacian matrix [77, 37, 62]

$$L_{rw} = D^{-1}L \text{ [77, 37]}$$

$$L_{sym} = D^{-1/2}LD^{-1/2} \text{ [37, 62]}$$

In addition, as the last step of the spectral clustering method is to apply k-means algorithm which uses the distance to cluster the matrix M, it can also be helpful if we normalize matrix M before performing k-means [62].

On the other side, like k-means algorithm, the main issue to consider before applying the spectral clustering method is to estimating the number of clusters. One technique can be applied that we examine the gaps between the Eigenvalues of the Laplacian matrices. For example, the Eigenvalues (0.00, 2.96, 32.40, 34.99, 41.73, 49.91) (Table 5.2) tells us that the number of the clusters should be 2 since there is a big gap between the second Eigenvalue and the third Eigenvalue.

5.2.3 Modularity and Louvain method

Another approach for density-based graph clustering is based on modularity. This modularity was actually designed to measure the strength of division of a graph into clusters (or communities). However, modularity is often used as the objective functions in graph clustering. A popular method of this approach is the Louvain method [91].

5.2.3.1 Modularity

In definition, modularity is the fraction of the edges that fall within the given groups minus the expected fraction if edges were distributed at random. To mathematically formulate the modularity by this definition, we firstly compute the expected number of edges. To do so, each edge of the graph is broken into two halves which are called *stubs* (Figure 5.5)

If the total number of the *stubs* is called l and m is the total weights of the edges then we have:

$$\text{l} = 2\text{m}$$

For the unweighted example above (Figure 5.5), $m = 7$ and $l = 14$.

We now consider two nodes labeled v_i and v_j of a graph. These nodes have the degrees d_i and d_j. The probability of selecting one *stub* from node v_i and node v_j is $\frac{d_i}{2m}$ and $\frac{d_j}{2m}$ respectively.

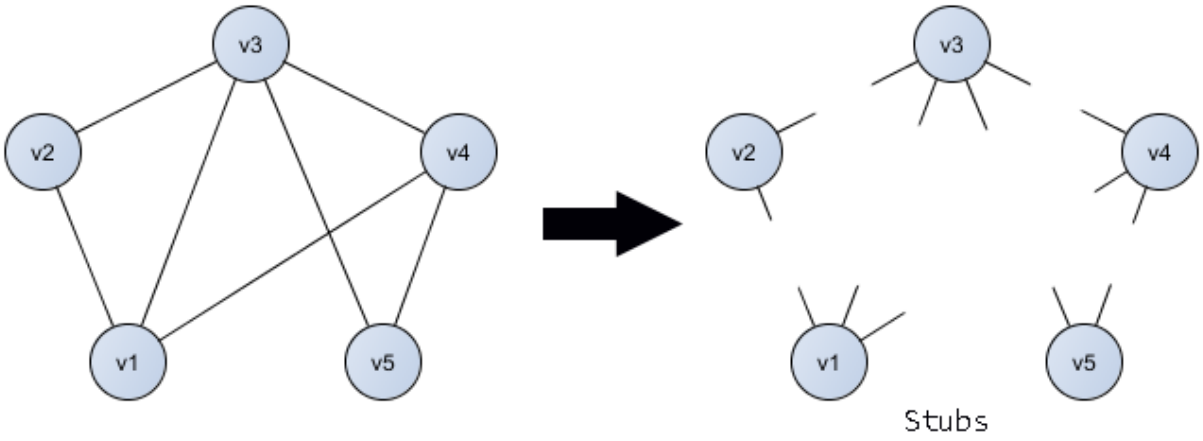


Figure 5.5: Modularity computation: breaking a graph to stubs

Therefore, the probability (or the expected value, called e_{ij}) of having an edge (weight = 1) connecting node v_i with node v_j is:

$$e_{ij} = \frac{d_i}{2m} * \frac{d_j}{2m}$$

Given that a graph is clustered into sub graphs. A sub graph (called A) has the total degrees of all the nodes in it (called d_A):

$$d_A = \sum_{v_i \in A} d_i$$

Then the expected fraction of the edges in the sub graph A (called E_A) will be:

$$E_A = \frac{d_A}{2m} * \frac{d_A}{2m}$$

On the other side, with w_{ij} is the weight of the edge connecting node v_i with node v_j, the fraction of the edge (called r_{ij}) and of all the edges in the sub graph A (called R_A) are:

$$r_{ij} = \frac{w_{ij}}{m}$$

$$R_A = \sum_{v_i, v_j \in A} r_{ij} = \sum_{v_i, v_j \in A} (\frac{w_{ij}}{m})$$

The modularity of the sub graph A (called Q_A) is calculated by the following equation:

$$Q_A = R_A - E_A \tag{5.3}$$

In general, given that the graph is clustered into K sub graphs presented by $A_1, A_2, ..., A_K$, the modularity of this clustering (called Q) is the sum of the modularity of all the sub graphs, Q_i:

$$Q = Q_1 + Q_2 + ... + Q_K \tag{5.4}$$

To illustrate how to compute the modularity value of a graph clustering, we return to the example of 2 clusters which are A_1 contains $\{v_1, v_2, v_3\}$ and A_2 contains $\{v_4, v_5, v_6\}$ (Figure 5.3).

Firstly, we compute the total weight of the whole graph.

$$m = (10 + 12 + 13 + 3 + 2 + 15 + 9 + 17) = 81$$

as well as the degrees of the nodes:

$$d_1 = (10 + 12) = 22$$

$$d_2 = (10 + 13) = 23$$

$$d_3 = (12 + 13 + 3 + 2) = 30$$

$$d_4 = (3 + 15 + 9) = 27$$

$$d_5 = (2 + 15 + 17) = 34$$

$$d_6 = (9 + 17) = 26$$

and the total degrees of all the nodes in each sub graph, A_1, A_2.

$$d_{A1} = (d_1 + d_2 + d_3) = 22 + 23 + 30 = 75$$

$$d_{A2} = (d_4 + d_5 + d_6) = 27 + 34 + 26 = 87$$

Secondly, we compute the modularity Q_1 of the sub graph A_1 by equation 5.3.

$$Q_1 = R_1 - E_1$$

In which,

$$E_1 = \frac{d_{A1}}{2m} * \frac{d_{A1}}{2m} = \frac{75}{2 * 81} * \frac{75}{2 * 81} = 0.214$$

$$R_1 = \sum_{v_i, v_j \in A_1} (\frac{w_{ij}}{m}) = \frac{(10 + 12 + 13)}{81} = 0.432$$

So that, we have:

$$Q_1 = R_1 - E_1 = 0.432 - 0.214 = 0.218$$

Similarity, we compute the modularity Q_2 of the sub graph A_2.

$$E_2 = \frac{d_{A2}}{2m} * \frac{d_{A2}}{2m} = \frac{87}{2 * 81} * \frac{87}{2 * 81} = 0.288$$

$$R_2 = \sum_{v_i, v_j \in A_2} (\frac{w_{ij}}{m}) = \frac{(15 + 9 + 17)}{81} = 0.506$$

$$Q_2 = R_2 - E_2 = 0.506 - 0.288 = 0.218$$

The last step is to compute the modularity of the graph clustering by summing up the modularities of the sub graphs as equation 5.4.

$$Q = Q_1 + Q_2 = 0.218 + 0.218 = 0.436$$

This section has introduced the modularity of graph clustering. This scalar value which ranges from -1 to 1 is used to evaluate the strength of graph clustering. More specifically, the higher modularity the better graph clustering we have [91]. Therefore, the modularity can also be used as the objective function in clustering graphs. One of the most popular method is Louvain method that will be presented in the next section.

5.2.3.2 Louvain method

The Louvain method is a graph clustering that is based on the modularity value. The idea of the approach is that the nodes will be moved around to the their neighbor clusters so that the modularity of the clustering increases. More specifically, the Louvain method has several phases that are presented below.

At the first phase, the method firstly considers each node of the graph as an individual cluster. That means at the beginning, the number of clusters equals the number of nodes of the weighted graph. Each node v_i has number of neighbors v_j that there is an edge $\{v_i, v_j\}$. The Louvain method works by moving every node v_i from its cluster to the clusters of v_j (called neighbor clusters) for maximum gain of modularity. To illustrate how this step works, let call Q_{ib} and Q_{ia} are the modularities of the cluster containing node v_i respectively before and after removing node v_i from it. Similarity, Q_{jb} and Q_{ja} are the modularities of the neighbor cluster containing neighbor node v_j respectively before and after adding node v_i into that neighbor cluster. The gain of modularity (called ΔQ) by moving node v_i from its cluster to the neighbor cluster of node v_j is calculated by the following formula:

$$\Delta Q = (Q_{ia} + Q_{ja}) - (Q_{ib} + Q_{jb})$$

By calculating ΔQ with all the neighbor clusters, node v_i will be placed in the cluster that brings the maximum of ΔQ that must also be a positive number. In the case that all the ΔQ are the negative numbers, node v_i will stay in its cluster. This first phrase terminates when no movement of nodes can help increase the modularity. In the other words, the output of this first phrase is the clusters of the graph that has the modulariry of maximum (Figure 5.6 is an example result after the first phrase)

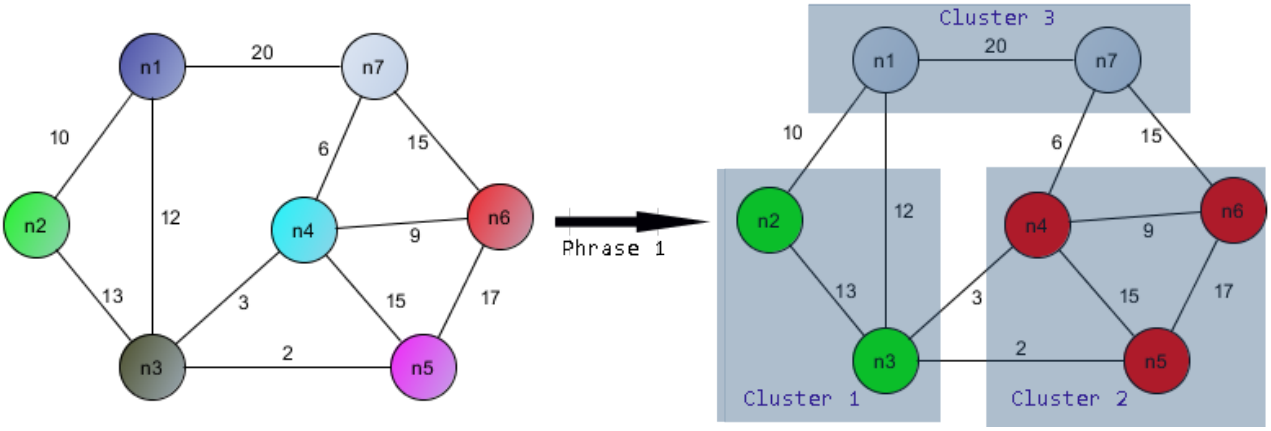


Figure 5.6: Example phrase 1 of Louvain method

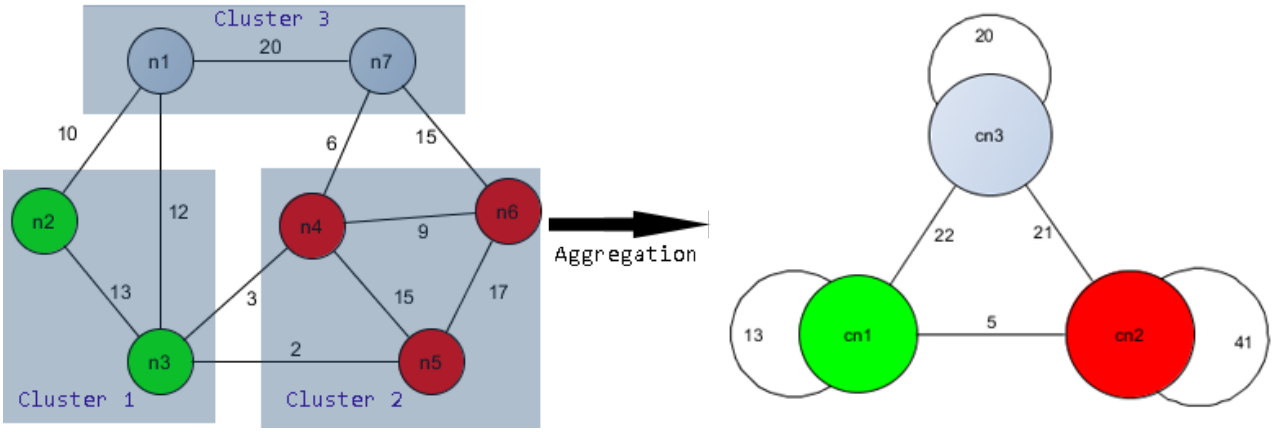


Figure 5.7: Example of aggregation process in Louvain method

The second phrase of the method starts by building a new graph from the result of the first phrase. In particular, in the new graph, each cluster now is considered as a node (called cluster node). The weight of the edge connecting two cluster nodes is the total weights of all the edges connecting two nodes in the two clusters. In addition, the nodes in the same clusters will create a self-loop edge whose weight is the total weights of all the edges connecting two nodes inside that cluster. Returning to the example result of the first phrase, the aggregation process is illustrated by Figure 5.7.

After building the new graph, the steps taken in the first phrase are repeated to cluster the new graph. The question is how many phrases we should take to cluster a graph? The answer is that it depends on the needs. More specifically, if we want more clusters, we can stop after the first phrase. We also can run the algorithm until the new graph cannot be clustered. For example, when we applied on our dataset (section 5.1.1), the algorithm stopped after 3 phrases.

One main issue we need to consider in the case we apply the algorithm on the large graph is the computation time. To improve the computation time, the Louvain method also presented some simple heuristics such as stopping the first phase when the gain of modularity is below a given

threshold or by removing the nodes of degree 1 (leaves) [91]. Another approach for the heuristics is that instead of moving a node to all of its neighbor clusters, we move it to the neighbor clusters of certain number of neighbor nodes after order them by weights of the connecting edges. For example, in our dataset (section 5.1.1), the number of the neighbor nodes we tried is 10 and it returns the same result as the one that no heuristics is applied.

5.2.3.3 Customization of Louvain method

In our work, we need to add several constraints to this method. One constraint is that each final hospital cluster must contain a public University Hospital (fr. Centre Hospitalier Universitaire - CHU). This constraint is taken into account in our implementation by customizing the Louvain method. In particular, these CHUs are considered as "seed" nodes of the graph. Our customization method is that these "seed" nodes will not be moved. Instead, the other nodes will be moved to neighbor clusters which contain the "seed" nodes.

5.3 Hospital community experiments, results, and discussions

As mentioned in the introduction section, the purpose of our works is to effectively split French hospital networks into communities for sharing medical records. In our approach, we are based on the dataset of flows that patients change the hospitals for the treatments (section 5.1.1). This dataset can be presented by an undirected graph on which the nodes are the hospital IDs and the weight of an edge indicates the number of patient exchange flows between the two hospitals. In this section, we firstly delivery our implementation approach for both the spectral clustering method the Louvain method. Secondly, we present the experiment results of thes methods on our dataset. Finally, to visualize the hospital communities, we use spatial maps.

5.3.1 Implementation approaches

In literature, there are already the libraries that implementing both the graph clustering methods introduced in the previous section (section 5.2). For example, *sklearn* library [65] and *python-louvain* library [8] have already implemented *SpectralClustering* and *Louvain* method respectively. However, as we have mentioned in the introduction section, we are going to not only compare the performance of the two methods, but also to customize the method so that we can add the

constraints to meet our needs. On the other words, implementing these methods are needed. Our approaches for the implementation are based on the programming language of Python 3 and the environments of Anaconda 3 and Ubuntu 18. In particular, we programmed a class in Python called *HospitalCluster*. This class has some main *methods* as follows:

- *modularity_ q_ i*: This method returns the modularity Q_i such as Q_1, Q_2 mentioned in the section 5.2.3.
- *modularity_ move_ hospital*: This method is to find the neighbor cluster to place node i for the maximum of modularity.
- *modularity_ optimization*: This method performs the movement of all the nodes until no node movement help improve the modularity.
- *aggregation_ process*: This method is corresponding to the aggregation process to build the new graph from the result of the previous phrase.
- *spectral_ clustering*: This method performs graph clustering based on the method of spectral clustering mentioned in the section 5.2.2.

The details of these methods will be presented in the next sections.

5.3.1.1 Method *modularity_ q_ i*

To compute the modularity Q_i, this method needs two parameters. The first parametter is a kind of dictionary (named *dict_ clusters*) telling the cluster number of each nodes while the second one is number of the cluster i (named c_i) corresponding to Q_i. In addition, while we search for the optimization solution of modularity, the nodes will be moved back and forth many times. Therefore, to reduce the computation time, the modularity of a cluster containing the same nodes should be memorized instead of re-computing. In particular, we use a global variable (called *memorized*) that is a directory with the keys are vectors of node IDs to record the modularity corresponding to a cluster containing these nodes. Moreover, this method also calls the following functions: (1) *extract_ member* to obtain the list of the nodes that are the members of the cluster c_i; (2) *extract_ degree* to return the total degrees of all the nodes that are passes as the parameters; (3) *extract_ weight* for the total weights of edges between two nodes among all the nodes that are passes as the parameters. Besides that the method also use a pre-computed variable named *total_ w_ all* which holds the total weight value of the entire graph. The details of the implementation of this method is presented below.

```
function modularity_q_i(dict_clusters, c_i)
begin
    lst_members = extract_member(dict_clusters, c_i) # member nodes of cluster c_i
    if (memorized[lst_members] exists) then
        q_i =  memorized[lst_members]
    else
        # Total degrees of nodes inside c_i
        total_d_i = extract_degree(dict_clusters, lst_members)
        expected_fraction = (total_d_i*total_d_i)/((2*total_w_all)*(2*total_w_all))
        # Total weights connecting nodes in lst_members
        total_w_i = extract_weight(dict_clusters, lst_members)
        actual_fraction = total_w_i/total_w_all
        q_i = actual_fraction - expected_fraction
        memorized[lst_members] = q_i
    end if
    return q_i
end function
```

5.3.1.2 Method *modularity_ move_ hospital*

To find the neighbor cluster to place node v_i for the highest modularity that can be gained, besides the parameter of v_i, this method also needs *dict_ clusters* (the same definition as the one in method *modularity_ q_ i*) as the second parameter. The method works by comparing for the highest modularity while moving v_i around to its neighbor clusters. As it can be seen in the code below, this method also use some other functions: (1) *which_ cluster* that returns the list of clusters of the corresponding nodes; (2) *extract_ neighbor* that returns the list of neightbors of node v_i. The function *extract_ neighbor* has two optional parameters. The first one (called n) to indicate the number of neighbor nodes to consider to place node v_i to the corresponding neighbor clusters. This parameter n can be used to reduce the computation time because we do not need to try moving v_i to all neighbor clusters. Instead, for example with our dataset, it returns the same result by the first 10 neighbors whose weights of edges with v_i are biggest. On the other side, in our works, we need to try several options such as there is a constraint that each final clusters must contain a public University Hospital (fr. Centre Hospitalier Universitaire - CHU). To consider this constrain, we add to the function *extract_ neighbor* a new parameter that are a list of the nodes (called *seeds*). The function *extract_ neighbor* will return neighbor nodes that are also the *seeds*. Moreover, as it can be also seen in the code below, there are two values this method returns. While the first returned value tells that whether or not the *dict_ clusters* is updated (or in other words, the node v_i either stays in its original cluster or is moved to a new cluster), the second value is the new *dict_ clusters* after the node v_i is moved to a new cluster.

```
function modularity_move_hospital(dict_clusters, v_i, n, seeds)
begin
    c_i = which_cluster(dict_clusters, v_i) # cluster of node v_i
    q_i_b = modularity_q_i(dict_clusters, c_i) # Q_i before removing v_i from c_i
    # find first n neighbor nodes if n > 0
    # if seeds is not empty, a neighbor node is also a seed node
    neighbor_nodes = extract_neighbor(v_i, n, seeds)
    # find neighbor_clusters but exclude c_i
    neighbor_clusters = which_cluster(dict_clusters, neighbor_nodes) - {c_i}
    best_cluster = c_i
    best_delta = 0
    foreach (c_j in neighbor_clusters) do
        # Q_j before moving v_i to c_j
        q_j_b = modularity_q_i(dict_clusters, c_j)
        dict_clusters[v_i] = c_j  # assign new cluster for v_i
        # Q_i after removing v_i from c_i
        q_i_a = modularity_q_i(dict_clusters, c_i)
        # Q_j after moving v_i to c_j
        q_j_a = modularity_q_i(dict_clusters, c_j)
        delta = (q_i_a + q_j_a) - (q_i_b + q_j_b)
        if(delta > best_delta) then
            best_delta = delta
            best_cluster = c_j
        end if
    end foreach
    dict_clusters[v_i] = best_cluster # assign v_i to the best cluster.
    if(best_cluster == c_i) then
        b_update = false # no update occurs
    else
        b_update = true # update occurs
    end if
    return b_update, dict_clusters
end function
```

5.3.1.3 Method *modularity_ optimization*

This is the method that conducts graph clustering by trying to maximize the modularity for each phrase mentioned in section 5.2.3. At the beginning, this method assigns an one-node cluster for every node by *init_ cluster* function. That means the number of the initial clusters equals the number of the nodes. After that, the method works by trying to move every node

v_i (except the *seeds* if any)to its neighbor clusters until no node movement can help improve the modularity. In this method, the optional parameters, n and *seeds* mentioned in section of the method *modularity_ move_ hospital* are included as its parameters. The value this method returns is the dictionary *dict_ clusters* that indicates the cluster of each node.

```
function modularity_optimization(n, seeds)
begin
    b_continue = true
    dict_clusters = init_cluster()
    movable_nodes = all_nodes - seeds # seed nodes are not to move
    while (b_continue == true) do
        b_stop = true
        foreach (v_i in movable_nodes) do
            b_update, dict_clusters = modularity_move_hospital(dict_clusters, v_i, n, seeds)
            if(b_update) then
                b_stop = false
            end if
            if(b_stop) then
                b_continue = false
            end if
        end foreach
    end while
    return dict_clusters
end function
```

5.3.1.4 Method *aggregation_ process*

This method is to build the aggregation graph from the result of the previous phrase. This method also calls the following functions: (1) *create_ graph* to create a blank graph; (2) *get_ unique_ clusters* to get the list of clusters; (3) *add_ node* to add a node to a graph; (4) *extract_ member* to extract the list of member of a cluster; (5) *extract_ weight* to extract total weights of the edges connecting nodes in two separated lists of nodes; (6) *add_ edge* to add edge with its weights to graph.

```
function aggregation_process(dict_clusters)
begin
    agg_graph = create_graph() # Create a blank graph
    lst_clusters = get_unique_clusters(dict_clusters) # Get unique clusters
    foreach (c_i in lst_clusters) do
```

```
        add_node(agg_graph, c_i)  # Add a node to graph
    end for
    foreach (c_i in lst_clusters) do
        c_i_members = extract_member(dict_clusters, c_i)
        foreach (c_j in lst_clusters) do
            c_j_members = extract_member(dict_clusters, c_j)
            # Extract total weights connecting nodes in c_i vs nodes in c_j
            edge_weight = extract_weight(dict_clusters, c_i_members, c_j_members)
            # Add edge with its weights to graph
            add_edge(agg_graph, c_i, c_j, edge_weight)
        end for
    end for
    return agg_graph
end function
```

5.3.1.5 Method *spectral_ clustering*

To implement graph clustering method which is based on matrix operations, we rely on the libraries of *numpy* and *scipy* written in Python. Moreover, we also use K-means method from the library of *sklearn.cluster*. In addition, the way we normalize the matrix of the K first eigenvectors (matrix M in section 5.2.2) by normalizing the row sums to have norm 1 (equation 5.5)

$$u_{ij} = \frac{v_{ij}}{\sum_k (v_{ik}^2)^{1/2}} \tag{5.5}$$

Particularly, the codes of the method *spectral_ clustering* in Python to cut graph presenting by *weight matrix* W into K clusters is presented below. As mentioned in section 5.2.2, in spectral clustering method, the Laplacian can be unnormalized or normalized, we use *Laplacian_ type* as the parameter to indicate what type of Laplacian matrix to be used. In particular, the parameter *Laplacian_ type* has three options. The first option which is also the default option is empty string (“ ”) that indicates that Laplacian matrix is unnormalized. The two other options which are “rw” and “sym” correspond to two normalized Laplacian matrice, L_{rw} and L_{sym} respectively. Moreover, in this method, there is another parameter *is_ norm* that is to indicate whether or not matrix M will be normalized. In particular, the function *normalize_ matrix* conducts the normalization by equation 5.5

```
function spectral_clustering(W, K, Laplacian_type = "", is_norm= False)
begin
    D = numpy.diag(W.sum(axis=1)) # Degree matrix
    L = D - W # unnormalized Laplacian matrix
```

```
    if(Laplacian_type == "rw") then # Lrw matrix
        Drw = scipy.linalg.fractional_matrix_power(D, -1)
        L = numpy.dot(Drw, L)
    end if
    if(Laplacian_type == "sym") then # Lsym matrix
        Dsym = scipy.linalg.fractional_matrix_power(D, -0.5)
        L = reduce(numpy.dot, [Dsym, L, Dsym])
    end if
    vals, vecs = numpy.linalg.eig(L)  # eigenvalues and eigenvectors
    vecs = vecs[:,numpy.argsort(vals)]  # sort eigenvectors based on the eigenvalues
    M = vecs[:,:K] # Matris of K first eigenvectors
    if(is_norm == True):
        M = normalize_matrix(M)
    kmeans = sklearn.cluster.KMeans(n_clusters=K)
    kmeans.fit(M)
    dict_clusters = kmeans.labels_
    return  dict_clusters
end function
```

5.3.2 Results and discussions

5.3.2.1 Method comparison

After having all necessary implementations, we conducted the experiments on our dataset mentioned in the introduction section. To compare the performance, we focus on the quality rather than the effectiveness in term of time computation. In particular, as mentioned in the introduction section, to measure the performance, we use three criteria: (1) modularity value; (2) percentage of the hospitals cannot access to the patients' medical record from previous hospitalization after building the communities; (3) the balance in term of number of the hospitals in each community. In particular, the table 5.3 below contains the values for the three criteria of the methods mentioned below. However, before looking for the details, it should be noted that on the table 5.3, the number of the clusters is 19. This number of 19 is generated by the Louvain method after running three phrases. For the purpose of comparison between the approaches, we use the same number of 19 for all the spectral clustering methods mentioned below.

- Spectral clustering with unnormalized Laplacian matrix L (SC).
- Normalized spectral clustering with normalized Laplacian matrix L_{sym} and the matrix of the first eigenvectors is not normalized (SC_{sym}).

- Normalized spectral clustering with normalized Laplacian matrix L_{rw} (SC_{rw}).
- Normalized spectral clustering with normalized Laplacian matrix L_{rw} and the matrix of the first eigenvectors is normalized by the equation 5.5 ($SC_{rw+norm}$).
- Louvain method.

Table 5.3: Performance of spectral clustering (SC) and Louvain methods

Evaluation criteria	SC	SC_{sym}	SC_{rw}	$SC_{rw+norm}$	Louvain
Modularity value	0.000	0.701	0.804	0.816	0.822
% previous hospitals outside community	0.006	20.44	9.33	10.32	9.84
# hospitals in biggest community	1,758	838	379	269	260
# hospitals in smallest community	1	2	2	22	22

As it can be seen on the table 5.3, the SC method (or *mincut* solution) does not work on our dataset. In particular, it returns a very big community covering almost all hospitals (1,758 over 1,777) while in the other communities, the numbers of hospitals are just 1 or 2. This result is an example of the problem caused by the mincut solution we mentioned in section 5.2.2. The solutions to the mincut problem are to take into account the size of the sub clusters. More specifically, instead of using unnormalized Laplacian matrix, the normalized Laplacian matrices have been used. These matrices are L_{rw} and L_{sym} which are mentioned in section 5.2.2. The methods corresponding to these matrice are SC_{rw} and SC_{sym} respectively. Between SC_{rw} and SC_{sym} methods, as it is shown in the table 5.3, the SC_{rw} method returns better results in all the criteria. As we use Python, we also compare our spectral clustering methods with the available spectral clustering method of the sklearn library (named *SpectralClustering*). The result shows that the default *SpectralClustering* gives the same result as the SC_{rw} method does. Moreover, by normalizing the matrix of the first K eigenvertors (matrix M that the k-means algorithm is applied on, section 5.2.2), the corresponding method labeled $SC_{rw+norm}$ returns the higher value for the modularity as well as more-balance communities. Particularly, the modularity increases from 0.804 to 0.816 while the numbers of hospitals in the biggest community reduces from 379 to 269 and the number of hospitals in the smallest community increases from 2 to 22. The only one criteria that $SC_{rw+norm}$ method is not better than SC_{rw} is the percentage (%) previous hospitals outside community. In other words, $SC_{rw+norm}$ method returns the communities of hospitals that the rate the hospitals cannot access to the patients' medical record from previous hospitalization is higher than SC_{rw} method. These values are 10.32 and 9.33 for $SC_{rw+norm}$ and SC_{rw} respectively. This result can be explained by the numbers of patient flows inside the biggest community by each method. The $SC_{rw+norm}$ method returns the biggest community that has 269 hospitals and the numbers of patient flows inside this community is 2,248,178 (17.17%). On the other hand,

the numbers that SC_{rw} method returns are 379 and 2,892,368 (22.09%) respectively. It is clear that when the biggest community gets bigger then the flows of patients outside the communities (connecting communities) will be smaller. These flows of patients outside has the same meaning as the number of patients that hospitals cannot access to the patients' medical record from previous hospitalization. Therefore, although compared with SC_{rw} method, $SC_{rw+norm}$ method returns the higher rate that the hospitals cannot access to the patients' medical record from previous hospitalization, we can still conclude that $SC_{rw+norm}$ method is better in this case. Finally, to help us select the better method, we compare $SC_{rw+norm}$ method with Louvain method. As it can been seen on the table 5.3, Louvain method returns better results in all the criteria. Therefore, we have selected Louvain method to cluster the hospitals into the communites for sharing patients' medical records.

5.3.2.2 Final result

As mentioned in the introduction, our work aims at clustering the hospitals into the hospital communities for sharing medical records in order for the hospitals to deliver more effective treatments to the patients. After comparing the spectral clustering methods and the Louvain method, we have selected Louvain method for our work. In this section, we present the results of our work. In particular, the table 5.4 shows the summary of the Louvain method running for three phrases.

In addition to the summary in table 5.4, table 5.5 brings in the details inside each community after phrase 3. The communities listed in this table are ordered by the number of hospitals, not by the numbers of the patient flows inside them. Moreover, the locations of these communities can be visualized with a spatial map of metropolitan France (Figure 5.8). More in details, the figure 5.8 only maps the locations of the first 17 communities listed in the table 5.5, the 2 last communities which are in France overseas are not included.

Table 5.4: Results of Louvain method

	Phrase 1	Phrase 2	Phrase 3
Number of communities	103	27	19
Modularity value	0.728	0.815	0.822
% previous hospitals outside community	22.89%	11.42%	9.84%
# hospitals in biggest community	216	260	260
# hospitals in smallest community	2	14	22

Table 5.5: Details of communities by Louvain method

Community	Number of hospitals	Previous provider located within community	Previous provider located outside community	% Within community
1	260	2,186,805	230,950	90.45
2	171	1,171,110	91,188	92.78
3	138	967,312	94,914	91.06
4	133	637,415	84,148	88.34
5	125	529,591	59,327	89.93
6	102	638,005	82,341	88.57
7	95	751,129	81,940	90.16
8	94	892,364	41,113	95.60
9	85	422,831	70,399	85.73
10	83	455,684	82,867	84.61
11	78	580,529	62,943	90.22
12	77	529,977	60,202	89.80
13	66	463,606	43,567	91.41
14	60	426,626	33,373	92.74
15	59	321,093	54,159	85.57
16	48	289,226	59,739	82.88
17	47	313,069	36,590	89.54
18	34	119,008	12,970	90.17
19	22	110,166	5,792	95.01

As the map (Figure 5.8) shows, the two biggest communities (community 1 and 2) in term of both the number of hospitals and the number of patient flows inside are located in Paris and Lyon, which are the biggest cities of France, and their nearby regions. In addition, 19 over 96 departments in metropolitan France are split into at least two different communities. The "split" departments are shown in the map with lighter colors compared to the color presenting the communities. Moreover, 15 over 132 GHTs [3] are split into different communities. This knowledge can be used to advise health authorities that they should not use administrative region borders as constraints when creating hospital communities.

[3]There are no patient flows inside 5 GHTs

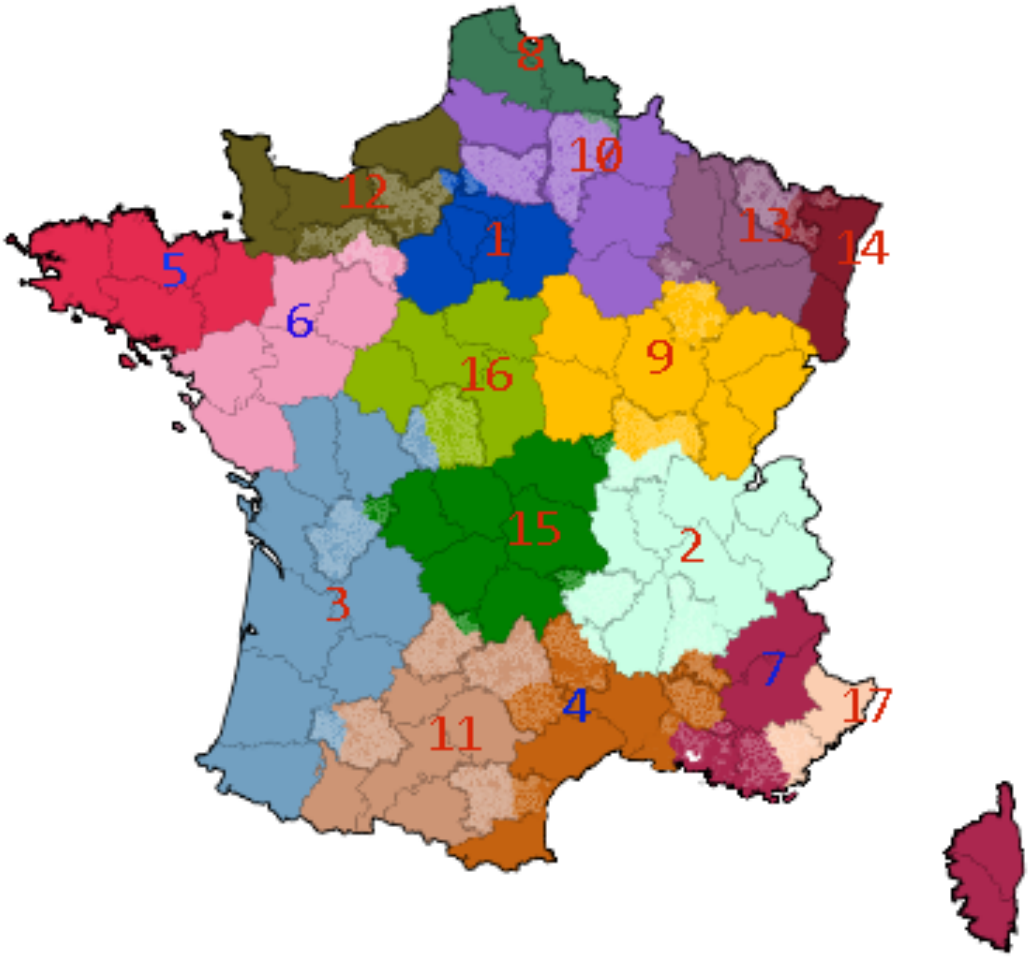


Figure 5.8: Locations of hospital communities in France

5.4 Conclusions

Lacking medical information from the previous hospitalizations about a patient can prevent hospitals from providing effective and high-quality treatments to those patients. Therefore, building hospital communities among which medical records are shared is needed. Since grouping all the hospitals in French hospital networks into one community is costly and impractical, our works aim at effectively clustering the French hospital networks into hospital communities. In particular, based on the dataset of patient flows between hospitals, we approach graph clustering methods to effectively group the hospitals into communities. After comparing the performance with spectral clustering methods, we have selected Louvain method for our work. In addition, since we need to consider several options of clustering hospitals into the communities, we have customized Louvain method so that we can take into account the related constraints while partitioning the hospitals into the communities. As a result, after running three phases of the Louvain method, we obtained 19 hospital communities. Among them, the 17 biggest communities are in metropolitan France. More importantly, some departments in metropolitan France as well as some GHTs are

split into at least two different communities. This knowledge confirms the limitations of building hospital communities based on administrative boundaries. In addition, such methods could be used to effectively design groups of hospitals that should share common electronic medical records.

Chapter 6

Conclusions and future works

Contents

6.1 Conclusions

As mentioned in the introduction section (Chapter 1), reducing the number of potentially avoidable hospitalization (PAHs) not only helps enhance quality of lives of the patients but also decrease substantial costs caused by patient treatments. Therefore, both the national- and regional-level health authorities in France are highly interested in enhancing the health care services in order to reduce the number of PAHs. The previous studies in France suggested that the number of PAHs in some geographic areas could be reduced by increasing the number of nurses at those geographic areas. Moreover, in France context, the public health decision makers can have influence on the densities of nurses at the geographic areas. However, there are also strong constraints that the healthcare system must provide quality care while controlling associated costs and ensuring equality of access to the health care services. In other words, all patient-citizens must be able to benefit from the care they need, regardless of their geographical and socioeconomic situation. These reasons gave birth to our project that aims at building a decision support system for the biggest reduction of PAH numbers while integrating the socio-economic constraints such as the limited budget for health care service improvement and the equality of health care access. More specifically, our work is going to recommend not only the geographic areas for improving health care service but also the optimized actions at those areas. Particularly, the geographic areas we worked on are the cross-border living areas (fr. Bassins de vie - BVs) that are defined as the geographic areas in which the inhabitants have access to the common equipment and services including trade, education, health, etc. In our approach, for every BV, we compare the predicted rates of PAHs before and after trying to add new nurses. Our idea is that the BVs that return the biggest reduction of these predicted values after trying to increase the number of nurses could be the best ones for the actual nurse implementation. Since the rates of PAHs are the numeric values, we have evaluated the potential of all the common regression methods. In particular, we have evaluated the potential as well as the quality performance of the following methods:

- Multilinear regression
- K–nearest neighbors for regression
- Neural networks for regression
- Support vector machine for regression

Based the performance which were measured and validated by root-mean square error and leave-one-out methods, support vector machine for regression (SVR) has been extended to spatial information by integrating the socio-economic constraints. Particularly, as mentioned above, we need to consider some constraints related to the number of nurses to be added. The first constraint should be the budget that the health authorities can spend for the health service improvement. This constraint indicates that the total number of nurses to be added in the whole region is limited. Another constraint we must consider is to ensure equal access to health care for the inhabitant

living in the region. The later constraint can be defined by the maximum number of to-be-added nurses in each BV and the densities of the nurses must not be greater than a given threshold. Taking account these constraints by extending support vector machine for regression method, we have been able to not only identify the BVs but also the number of to-be-added nurses for the biggest reductions in number of PAHs. For example, with the constraints that (1) the total amount of nurses can be added into the entire region is 30, (2) the maximum number of nurses can added into a BV is 3, and (3) the density of nurses must not be more than 25 nurses per 10,000 habitants, we are able to identify 16 BVs and the number of to-be-added nurses at each of 16 BVs for the biggest reduction of PAHs in number which is 17. The results are visualized using spatial maps as a user-friendly decision support system. Moreover, our approach is applied to the Occitanie region France, but it can be applied to other regions or extended at the national level or even to other countries. In addition, this approach could be applied to other health care policy issues, such as the reduction of hospital re-admissions or access to innovation.

On the other side, parts of our work are to collect data that could be the potential determinants of PAHs. Since it is clear that temperature, especially temperature extremes, have negative impacts to human health. For example, the extreme heat (or so called heatwave) that occurred in summer 2003 in France caused about 15,000 more deaths than expected in France (an increase of 55%). Therefore, we would like to conduct the analysis of the impacts of extremes temperature to PAHs as well as to include this environmental data in our decision support system mentioned above. To collect the temperature data, we rely on the temperature values measured by sensors at weather stations. However, for many reasons the values measured at these stations are sometimes discontinuous. In other words, there are missing values for temperatures measured at the weather stations. To select the reliable method in missing temperature imputation, we have compared the quality performance of two different methods representative of both the spatial interpolation methods and the time-series models. These methods are Inverse Distance Weighted (IDW) and Autoregressive Integrated Moving Average (ARIMA) respectively. Moreover, we have proposed a novel approach that combines these methods to improve the quality performance. Our method performs better at 100% and 99.8% the weather stations compared with IDW and ARIMA respectively. The performances of these methods were measured and validated by root-mean square error and leave-one-out methods using the temperature data that are hourly recorded by sensors at more than 600 weather stations implemented across Metropolitan France.

In addition, as mentioned at the introduction section, the high rates of potentially avoidable hospitalizations are associated with organizational features of health systems such as coordination between health care providers. In other words, improving the coordination between the health care providers could lead to the reduction of the potentially avoidable hospitalizations. That is because noticeable fact that patients do not visit the same hospitals every time. There are many reasons for that. For example, patients have changed addresses, they are not happy with the service of the previous hospital, or they need to seek specialized care in a tertiary hospital. In such cases,

it is clear that the treatment would be more efficient and the risk to patients' health could be eliminated or reduced if the later hospitals were able to access the medical records of the patients at the previous hospitals. In other words, there is a need to allow information technology systems to share medical records among hospitals. However, it is neither necessary nor practical for all hospitals in France to be grouped as one because it would be costly while some hospitals will never share any patient. Therefore, health authorities are interested in building hospital communities so that medical records can be shared among the hospitals in those communities. This brought up us another project which aims at dividing French hospital networks into communities for sharing patients' medical records. Particularly, our work is based on the flows of patients changing the hospitals for the treatments. These flows can be presented by a undirected weight graph in which the nodes present the hospitals while the edges present the size of patient flows. Therefore, to cluster these hospitals into communities, we rely on the approaches of the graph clustering. In particular, we have compared two different approaches. The first approach is the spectral clustering method and the second one is Louvain method, which is based on modularity values. To evaluate the performance of these methods, we are based on many criteria that include:

- Modularity value of graph clustering
- Percentage that the previous hospitals located outside the communities
- The balance in term of number of hospitals in each communities

Moreover, in our work, we need to consider several constraints. For example, one constraint is that each final hospital cluster must contain a public University Hospital (fr. Centre Hospitalier Universitaire - CHU). Therefore, besides comparing the performances of the two graph clustering methods, we need tocustomize them so that we can add the constraints to meet our needs. Therefore, we have im-plemented these method ourselves. As a result, the hospital network in France has been clustered 19 hospital communities. Among them, the 17 biggest communities are in metropolitan France. More importantly, some departments in metropolitan France as well as some GHTs are split into at least two different communities. This knowledge confirms the limitations of building hospital communities based on administrative boundaries. In addition, such methods could be used to effectively design groups of hospitals that should share common electronic medical records.

6.2 Future works

6.2.1 Extending the work to national level

As mentioned in chapter 3, our approach is applied to the Occitanie region France. However, it can be applied to other regions or extended at the national level or even to other countries. That could be the parts of our work in the future. More specifically, in the near future, we plan to extend our work at the national level because of the availability as well as the similarity of the related datasets.

6.2.2 Extending the work to include environmental data

As mentioned before, although there are many studies confirming the negative impacts of extreme temperatures to human health, the impacts of extreme temperatures to specific PAHs are still unclear to us because of the limitation of the PAH dataset we have. Particularly, at the time of this report we only have the data of PAHs that does not contain the information of exact dates the patients were admitted to the hospitals. On the other side, the lag of the extreme temperature impacts to human health could be for only several days. Therefore, we have neither been able to measure the impact of the extreme temperature to PAHs nor include the temperature in the decision support system. In the mean time, as introduced in chapter 4, we have already proposed a more reliable methods for temperature missing imputation. This work can be useful in the future when we are able to extract the more detail dataset of PAHs that include the exact dates the patients are admitted by the hospitals.

Furthermore, another type of environmental data, that is air pollution, should be considered. Like the extreme temperature, there is strong evidence to suggest high levels of air pollution negatively affect human health. However, there is no previous study on these effect on specific PAHs. That could be our interesting work in the future. Moreover, our proposal method for reliable temperature missing imputation introduced in chapter 4 could be applied for other spatio-temporal data like air pollutants. In particular, we would like to measure the performance of our proposal method on some air pollutants as parts of our future works.

6.2.3 Considering other constraints in hospital clustering

In chapter 5, we have introduced the approach of graph clustering for partitioning French hospital network into communities for sharing patient medical records. In addition, we have also taken into account the constraint while clustering the hospitals into communities. However, this work still has limitations because the characteristics of the hospitals such as the capacities of hosting patients (number of beds, number of doctors, etc) as well as the speciality in patient treatments like cancer centers have not been taken into account in the current approach. Therefore, in the near future, we would like to extend our work to integrate hospitals' characteristics in hospital clustering.

6.2.4 Prediction on PAH readmission

A hospital readmission is when a patient who is discharged from the hospital, gets re-admitted again within a certain period of time. Hospital readmission rates for certain conditions are now considered an indicator of hospital quality, and also affect the cost of care adversely. For example, American hospitals spent over $41 billion on diabetic patients who got readmitted within 30 days of discharge [3]. Hence, being able to determine factors that lead to higher readmission in such patients, and correspondingly being able to predict which patients will get readmitted can help hospitals save millions of dollars while improving quality of care. Therefore, one of our future work is to answer the following question: What factors are the strongest predictors of hospital readmission in PAHs patients?

Publications

Published

T. Ngo, V. Georgescu, T. Libourel, A. Laurent, and G. Mercier. Spatial Gradual Patterns: Application to the Measurement of Potentially Avoidable Hospitalizations. In: *SOFSEM 2018 Conference Proceedings*, Krems an der Donau, Austria. 2018.

T. Ngo, V. Georgescu, G. Carmen, A. Laurent , T. Libourel, G. Mercier. Extending Support Vector Regression to Constraint Optimization: Application to the Reduction of Potentially Avoidable Hospitalizations. In: *SECML PKDD (SoGood) 2018 Workshops*, Dublin, Ireland. 2018.

T. Ngo, V. Georgescu, G. Carmen, A. Laurent , T. Libourel, G. Mercier. Machine learning application to the reduction of ambulatory care sensitive admissions (ACSA). In: *European Journal of Public Health*, Marseille France. 2019.

Submitted

T. Ngo, V. Georgescu, G. Carmen, A. Laurent , T. Libourel, G. Mercier. Graph clustering for hospital communities. *Atelier IA and Santé 2020*, Angers France, 2020.

Bibliography

[1] K. Abhishek, M.P. Singh, S. Ghosh, and A. Anand. Weather forecasting model using artificial neural network. *Procedia Technology*, 4:311–318, 2012.

[2] Ada. Ada health gmbh. `https://ada.com/`. (Accessed 26 March 2020).

[3] Hines AL, Barrett ML, Jiang HJ, and Steiner CA. Conditions with the largest number of adult hospital readmissions by payer. *statistical brief*, 172, 2011.

[4] L. Anselin. Spatial data science. `https://www.youtube.com/channel/UCzvhOfSmJpRsFRF2Pgrv-Wg`. (Accessed 26 March 2020).

[5] L. Anselin, I. Syabri, and Y. Kho. Geoda: An introduction to spatial data analysis. `https://spatial.uchicago.edu/software`. (Accessed 26 March 2020).

[6] ATIH. Mise à jour 2019 de la liste de correspondance codes postaux codes géographiques pmsi. `https://www.atih.sante.fr/mise-jour-2019-de-la-liste-de-correspondance-codes-postaux-codes-geographiques-PMSI` (Accessed 26 March 2020).

[7] S.E. Awan, M. Bennamoun, F. Sohel, F.M. Sanfilippo, and G. Dwivedi. Machine learning-based prediction of heart failure readmission or death: implications of choosing the right model and the right metrics. *ESC Heart Fail*, 6(2):428–435, 2019.

[8] T. Aynaud. Louvain community detection. `https://github.com/taynaud/python-louvain`. (Accessed 26 March 2020).

[9] O. Babak and C. Deutsch. Statistical approach to inverse distance interpolation. *Stochastic Environmental Research and Risk Assessment*, 23:543–553, 2008.

[10] S. Badhiye, N. Sambhe, and P.N. Chatur. Knn technique for analysis and prediction of temperature and humidity data. *International Journal of Computer Applications*, 61:7–13, 2013.

[11] D.N. Barton, T.H. Bakken, and A.L. Madsen. Using a bayesian belief network to diagnose significant adverse effect of the eu water framework directive on hydropower production in norway. *Journal of Applied Water Engineering and Research*, 4 (1):11–24, 2016.

[12] A.B. Bindman, K. Grumbach, D. Osmond, M. Komaromy, K. Vranizan, N. Lurie, J. Billings, and A. Stewart. Preventable hospitalizations and access to health care. *JAMA*, 274(4):305–11, 1995.

[13] T. Boudemaghe and I. Belhadj. Data resource profile: The french national uniform hospital discharge data set database (pmsi). *International Journal of Epidemiology*, 46(2):392–392d, 2017.

[14] R. Bourret, G. Mercier, J. Mercier, O. Jonquet, J.E. De La Coussaye, P.J. Bousquet, J.M. Robine, and J. Bousquet. Comparison of two methods to report potentially avoidable hospitalizations in france in 2012: a cross-sectional study. *BMC Health Serv Res.*, 15(4), 2015.

[15] G. E. P. Box, G. M. Jenkins, and G. C. Reinsel. *Time series analysis: Forecasting control (3rd edition)*. Prentice Hall, NJ, USA, 1994.

[16] M. L. Burkey. A short course on spatial econometrics and gis. *REGION*, 5(3):R13–R18, 2018.

[17] T. Chai and R. R. Draxler. Root mean square error (rmse) or mean absolute error (mae)? – arguments against avoiding rmse in the literature. *Geosci. Model Dev.*, 7:1247–1250, 2014.

[18] N. Chaikaew, N.K. Tripathi, and M. Souris. Exploring spatial patterns and hotspots of diarrhea in chiang mai, thailand. *Int J Health Geogr*, 8:36, 2009.

[19] V. Christina, S. Karpagavalli, and G. Suganya. Email spam filtering using supervised machine learning techniques. *Int. J. Comput. Sci. Eng.*, 02 (09):3126–3129, 2010.

[20] J. Clarke, L. Warren, S. Arora, M. Barahona, and A. Darzi. Guiding interoperable electronic health records through patient-sharing networks. *pj Digital Med*, 2018.

[21] Isazi Consulting. What is machine learning? `https://www.isaziconsulting.co.za/machinelearning.html`. (Accessed 26 March 2020).

[22] C. Cortes and V. Vapnik. Support vector networks. *M Learning*, 20(3):273–297, 1995.

[23] data.gouv.fr. Correspondance code insee - code postal. `https://www.data.gouv.fr/fr/datasets/correspondance-code-insee-code-postal/`. (Accessed 26 March 2020).

[24] T. Dozat. Incorporating nesterov momentum into adam. *ICLR Workshop*, page 2013–2016, 2016.

[25] J. Duchi, E. Hazan, and Y. Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12:2121–2159, 2011.

[26] A. Esteva, B. Kuprel, R. Novoa, J. Ko, S. Swetter, H. Blau, and S. Thrun. Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 542:115–118, 2017.

[27] D. Falbel, J.J. Allaire, F. Chollet, Google RStudio, Y. Tang, W. Van Der Bijl, M. Studer, and S. Keydana. *R Interface to 'Keras'*, 2019.

[28] A. Fouillet, G. Rey, F. Laurent, G. Pavillon, S. Bellec, C. Guihenneuc-Jouyaux, J. Clavel, E. Jougla, and D. Hémon. Excess mortality related to the august 2003 heat wave in france. *Int Arch Occup Environ Health*, 80(1):16–24, 2006.

[29] A. Fouillet, G. Rey, F. Laurent, G. Pavillon, S. Bellec, C. Guihenneuc-Jouyaux, J. Clavel, E. Jougla, and D. Hémon. Excess mortality related to the august 2003 heat wave in france. *Int Arch Occup Environ Health*, 80(1):16–24, 2006.

[30] C. Franklin and H. Paula. An introduction to geographic information systems: Linking maps to databases. *Database*, 15(2):12–15, 1992.

[31] T. Freund, S. Campbell, S. Geissler, C. Kunz, C. Mahler, F. Peters-Klimm, and J. Szecsenyi. Strategies for reducing potentially avoidable hospitalizations for ambulatory care–sensitive conditions. *Ann Fam Med*, 11(4):363–370, 2013.

[32] J.D. Frizzell, L. Liang, P.J. Schulte, C.W. Yancy, P.A. Heidenreich, A.F. Hernandez, D.L. Bhatt, G.C. Fonarow, and W.K. Laskey. Prediction of 30-day all-cause readmissions in patients hospitalized for heart failure: Comparison of machine learning and other statistical approaches. *AMA Cardiol*, 2(2):204–209, 2017.

[33] J. Gao, E. Moran, Y. Li, and P. Almenoff. Predicting potentially avoidable hospitalizations. *Med Care.*, 52(2):164–71, 2014.

[34] E. Gleichgerrcht, B. Munsell, S. Bhatia, W.A. Vandergrift, C. Rorden, C. McDonald, J. Edwards, R. Kuzniecky, and L. Bonilha. Deep learning applied to whole-brain connectome to determine seizure control after epilepsy surgery. *Epilepsia*, 59(9):1643–1654, 2018.

[35] B. Gräler, E. Pebesma, and G. Heuvelink. Spatio-temporal interpolation using gstat. *The R Journal*, 8(1):204–218, 2016.

[36] L. Hagen and A. Kahng. New spectral methods for ratio cut partitioning and clustering. *IEEE Trans, Computer-Aided Design*, 11(9):1074–1085, 1992.

[37] D. Hamad and P. Biel. Introduction to spectral clustering. In *2008 3rd International Conference on Information and Communication Technologies: From Theory to Applications*, pages 1–6, 2008.

[38] A. Hammoudeh, G. Al-Naymat, I. Ghannam, and N. Obied. Predicting hospital readmission among diabetics using deep learning. *Procedia Computer Science*, 141:484–489, 2018.

[39] D. Harel and Y. Koren. On clustering using random walks. In *Hariharan R., Vinay V., Mukund M. (eds) FST TCS 2001: Foundations of Software Technology and Theoretical*

Computer Science. FSTTCS 2001. Lecture Notes in Computer Science, vol 2245. Springer, Berlin, Heidelberg, 2001.

[40] G. Hinton. Overview of mini-batch gradient descent. `http://www.cs.toronto.edu/~tijmen/csc321/slides/lecture_slides_lec6.pdf`. (Accessed 26 March 2020).

[41] C. Huber, N. Benda, and T. Friede. A comparison of subgroup identification methods in clinical drug development: Simulation study and regulatory considerations. *Pharm. Stat.*, 18(5):600–626, 2019.

[42] Association Infoclimat. Climatologie. `https://www.infoclimat.fr`, 2019. (Accessed 26 March 2020).

[43] Insee. Base des bassins de vie. `https://www.insee.fr/fr/information/2115016`. (Accessed 26 March 2020).

[44] D. P. Kingma and J. L. Ba. Adam: a method for stochastic optimization. *International Conference on Learning Representations*, page 1–13, 2015.

[45] K. Kourou, T. Exarchos, K. Exarchos, M. Karamouzis, and D. Fotiadis. Machine learning applications in cancer prognosis and prediction. *Computational and Structural Biotechnology Journal*, 13:8–17, 2015.

[46] K.C. Lam, R.G. Bryant, and J. Wainright. Application of spatial interpolation method for estimating the spatial variability of rainfall in semiarid new mexico, usa. *Mediterranean Journal of Social Sciences*, 6(4), 2015.

[47] W. W. M. Lam and K.C.C. Chan. A graph mining algorithm for classifying chemical compounds. In *2008 IEEE International Conference on Bioinformatics and Biomedicine*, 2008.

[48] L. Laranjo, A.G. Dunn, H.L. Tong, A.B. Kocaballi, J. Chenand, R. Bashir, D. Surian, B. Gallego, F. Magrabi, A.Y.S. Lau, and E Coiera. Conversational agents in healthcare: A systematic review. *Journal of the American Medical Informatics Association*, 25(9):1248–1258, 2018.

[49] J.P. LeSage. An introduction to spatial econometrics. *Revue d'économie industrielle*, 123 | 3e:19–44, 2008.

[50] J. Li, J. Cheng, J. Shi, and F. Huang. Brief introduction of back propagation (bp) neural network algorithm and its improvement. In *Jin D., Lin S. (eds) Advances in Computer Science and Information Engineering. Advances in Intelligent and Soft Computing*, volume 169. Springer, Berlin, Heidelberg, 2012.

[51] Y. Lin, M. Chen, G. Chen, X. Wu, and T. Lin. Application of an autoregressive integrated moving average model for predicting injury mortality in xiamen, china. *BMJ Open*, 5(12):5:e008491, 2015.

[52] F.D. Malliaros and M. Vazirgiannis. Clustering and community detection in directed networks: A survey. *CoRR*, abs/1308.0971, 2013.

[53] T.H. Jr. McCoy, A.M. Pellegrini, and R.H. Perlis. Assessment of time-series machine learning methods for forecasting hospital discharge volume. *JAMA Netw Open*, 1(7):e184087, 2018.

[54] G. Mercie, V. Georgescu, and J. Bousquet. Geographic variation in potentially avoidable hospitalizations in france. *Health Affairs (Millwood)*, 34(5):836–43, May 2015.

[55] D. Meyer, E. Dimitriadou, K. Hornik, A. Weingessel, and F. Leisch. *e1071: Misc Functions of the Department of Statistics, Probability Theory Group (Formerly: E1071), TU Wien*, 2017. R package version 1.6-8.

[56] T. Mitchell. *Machine Learning*. McGraw Hill, 1997.

[57] S. Moritz and T. Bartz-Beielstein. imputets: Time series missing value imputation in r. *The R Journal*, 9(1):207–218, 2017.

[58] S. Nas and M. Koyuncu. Emergency department capacity planning: A recurrent neural network and simulation approach. *Comput Math Methods Med*, 2019:4359719, 2019.

[59] J. K. Nelson and C. A. Brewer. Evaluating data stability in aggregation structures across spatial scales: revisiting the modifiable areal unit problem. *Cartography and Geographic Information Science*, 44(1):35–50, 2017.

[60] Y. Nesterov. A method for unconstrained convex minimization problem with the rate of convergence o(1/k2). *Doklady ANSSSR (translated as Soviet.Math.Docl.)*, 269:543– 547, 1983.

[61] D. Neumann, B. Rajagopalan, and E. Zagona. Regression model for daily maximum stream temperature. *Journal of Environmental Engineering*, 129 (7):667–674, 2003.

[62] A. Ng, M. Jordan, and Y. Weiss. On spectral clustering: analysis and an algorithm. In *In T. Dietterich, S. Becker, and Z. Ghahramani (Eds.), Advances in Neural Information Processing Systems 14*. MIT Press, 2002.

[63] T. Ngo, V. Georgescu, T. Libourel, A. Laurent, and G. Mercier. Spatial gradual patterns: Application to the measurement of potentially avoidable hospitalizations. *Proc. of the SOFSEM Int. Conf.*, pages 596–608, 2018.

[64] E. J. Pebesma. Multivariable geostatistics in s: the gstat package. *Computers Geosciences*, 30(7):683–691, 2004.

[65] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[66] N. Qian. On the momentum term in gradient descent learning algorithms. *Neural Networks : The Official Journal of the International Neural Network Society*, 12(1):145–151, 1999.

[67] Y. Radhika and M. Shashi. Atmospheric temperature prediction using support vector machines. *International Journal of Computer Theory and Engineering*, 1(1):55–58, 2009.

[68] Y. C. T. Raymond. An application of the arima model to real-estate prices in hong kong. *Journal of Property Finance*, 8(2):152–163, 1997.

[69] S. Ribecca. The data visualisation catalogue. `https://datavizcatalogue.com/methods/choropleth.html`. (Accessed 26 March 2020).

[70] V.G. Rodwin, M.K. Gusmano, D. Weisz, and C. Le Pen. Prévenir l'hospitalisation : une étude pilote à partir des données du pmsi. *Rapport pour le Ministère chargé de la santé*, 2007.

[71] R. Rozenblum, R. Rodriguez-Monguio, L.A. Volk, K.J. Forsythe, S. Myers, M. McGurrin, D.H. Williams, D.W. Bates, G. Schiff, and E. Seoane-Vazquez. Using a machine learning system to identify and prevent medication prescribing errors: A clinical and cost analysis evaluation. *Jt Comm J Qual Patient Saf.*, 46(1):3–10, 2020.

[72] F. Savje. *distances: Tools for Distance Metrics*, 2017. R package version 0.1.2.

[73] S. Sayad. Support vector machine - regression (svr). `http://www.saedsayad.com/support_vector_machine_reg.htm`. (Accessed 26 March 2020).

[74] M. Segal, E. Rollins, K. Hodges, and M. Roozeboom. Medicare-medicaid eligible beneficiaries and potentially avoidable hospitalizations. *Medicare Medicaid Res Rev*, 4(1), 2014.

[75] P. Sen, M. Roy, and P. Pal. Application of arima for forecasting energy consumption and ghg emission: A case study of an indian pig iron manufacturing organization. *Energy*, 116:1031–1038, 2016.

[76] H. Shi, P. Xie, Z. Hu, M. Zhang, and E.P. Xing. Towards automated icd coding using deep learning. *Computation and Language*, 2017.

[77] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

[78] A.J. Smola and B. Schölkopf. A tutorial on support vector regression. *Statistics and Computing*, 14(3):199–222, 2004.

[79] A. Solomiac, A. Townsend, M. Le Bail, I. Hernando, N. Rigollot, and F. Pinelli. Guide méthodologique de calcul de l'indicateur hospitalisations potentiellement évitables (hpe) et présentation des principaux résultats. https://www.scansante.fr/sites/default/files/content/396/vf_-_guide_hpe_2018_03_20.pdf, 2018. (Accessed 26 March 2020).

[80] J. Somauroo. Ada health launches world's first ai-powered health app in swahili. https://www.forbes.com/sites/jamessomauroo/2019/12/02/ada-health-launches-worlds-first-ai-powered-health-app-in-swahili/. (Accessed 26 March 2020).

[81] J.G. Souza, E.M. Silva, P.F. Brito, J.A.F. Costa, A.C. Salgado, and S.R.L. Meira. Using graph clustering for community discovery in web-based social networks. In *Tan Y., Shi Y., Mo H. (eds) Advances in Swarm Intelligence. ICSI 2013. Lecture Notes in Computer Science*. Springer, Berlin, Heidelberg, 2013.

[82] S. Srinivas and A. Ravindran. Optimizing outpatient appointment system using machine learning algorithms and scheduling rules: A prescriptive analytics framework. *Expert Systems With Applications*, 2018.

[83] J. Stachelek and C. Madden. Application of inverse path distance weighting for high-density spatial mapping of coastal water quality patterns. *International Journal of Geographical Information Science*, 29(7), 2015.

[84] M. Stoer and F. Wagner. A simple min-cut algorithm. *J. ACM*, 44(4):585–591, 1997.

[85] Arcgis Storymaps. Spatial autocorrelation and spatial sampling. https://storymaps.arcgis.com/stories/a03364ab651c481e9a434353339744d4. (Accessed 26 March 2020).

[86] G. Swapna, R. Vinayakumar, and K.P. Soman. Diabetes detection using deep learning algorithms. *ICT Express*, 4(4):243–246, 2018.

[87] W. Tobler. A computer movie simulating urban growth in the detroit region. *Economic Geography*, 46(Supplement):234–240, 1970.

[88] T.B. Trafalis and H. Ince. Support vector machine for regression and applications to financial forecasting. *Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks IJCNN 2000*, 6:348–353, 2000.

[89] A. Ung, M. Corso, M. Pascal, K. Laaidi, V. Wagner, P. Beaudeau, and A. Le-Tertre. *Évaluation de la surmortalité pendant les canicules des étés 2006 et 2015 en France métropolitaine*. Saint-Maurice : Santé publique France, 2019. (Accessed 26 March 2020).

[90] V. Vapnik and A. Lerner. Pattern recognition using generalized portrait method. *Automation and Remote Control*, 24:774–780, 1963.

[91] B. D. Vincent, J. L. Guillaume, R. Lambiotte, and E. Lefebvre. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, P10008(10):1–12, 2008.

[92] U. Von Luxburg. A tutorial on spectral clustering. *CoRR*, abs/0711.0189, 2007.

[93] T. Vu. Machine learning co ban. `https://machinelearningcoban.com/`. (Accessed 26 March 2020).

[94] D. Wagner and F. Wagner. Between min cut and graph bisection. In *In Proceedings of the 18th International Symposium on Mathematical Foundations of Computer Science (MFCS)*, page 744–750. London: Springer, 1993.

[95] J.S. Weissman, C. Gatsonis, and A.M. Epstein. Rates of avoidable hospitalization by insurance status in massachusetts and maryland. *JAMA*, 268(17):2388–94, November 1992.

[96] I. Witten, E. Frank, M. Hall, and C. Pal. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, Burlington, MA, USA, 2016.

[97] M. D. Zeiler. Adadelta: An adaptive learning rate method. `http://arxiv.org/abs/1212.5701h`. (Accessed 26 March 2020).

[98] S. Zhang, S. M. H. Bamakan, Q. Qu, and S. Li. Learning for personalized medicine: A comprehensive review from a deep learning perspective. In *IEEE Reviews in Biomedical Engineering*, volume 12, pages 194–208, 2019.

[99] T. Zheng, W. Xie, L. Xu, X. He, Y. Zhang, M. You, G. Yang, and Y. Chen. A machine learning-based framework to identify type 2 diabetes through electronic health records. *International Journal of Medical Informatics*, 97:120–127, January 2017.

[100] X.Y. Zhou, Y. Guo, M. Shen, and G.Z. Yang. Artificial intelligence in surgery. *physics.med-ph*, 2019.