Sparse linear model with quadratic interactions

Florent Bascou

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

En Biostatistiques

École doctorale : Information, Structures, Systèmes

Unité de recherche : Institut Montpelliérain Alexander Grothendieck

Sparse Linear Model with Quadratic Interactions

Présentée par Florent Bascou

Le 09/09/2022

Sous la direction de Joseph Salmon et Sophie Lèbre

Devant le jury composé de

Joseph Salmon	Professeur, Université de Montpellier	Directeur de thèse
Sophie Lèbre	Maître de conférence, Université Montpellier 3	Co-encadrante
Julien Chiquet	Directeur de recherche, INRAE	Rapporteur
Karim Lounici	Professeur, École Polytechnique	Rapporteur
Marine le Morvan	Chargée de recherche, Inria	Examinatrice
Mathurin Massias	Chargé de recherche, Inria	Examinateur
Jean-Michel Marin	Professeur, Université de Montpellier	Président du jury

UNIVERSITÉ DE MONTPELLIER

Remerciements

Mes premiers remerciements vont à ma direction de thèse. Sophie, Joseph, merci de m'avoir aussi bien encadré ces 3 dernières années. Je tiens tout particulièrement à vous remercier pour votre attention constante, notamment pendant les mois de confinement, à ma thèse tout comme à mes projets futurs. Plus personnellement, merci Sophie de m'avoir aidé à comprendre le problème de régulation des gènes et à me dépatouiller avec les données, en plus de tes super idées pour améliorer mes graphiques et déceler leurs interprétations. Joseph, je garderai comme précieux souvenir mes premiers pas en tant qu'enseignant avec toi. Je te remercie également pour cette rigueur que tu m'as inculquée dès le début pour être sur de bon rails, aussi bien en Python qu'en rédaction, mais aussi pour tout ces "à cotés" qui ont fait que la thèse est restée une aventure agréable. À tous les deux, merci pour tout.

Je remercie ensuite Julien Chiquet et Karim Lounici d'avoir accepté de rapporter mon manuscrit, et Marine Le Morvan, Jean-Michel Marin, Mathurin Massias d'avoir accepté de constituer mon jury et d'assister à ma soutenance.

À mes collègues de l'IMAG. Benjamin, je te dédie la taille de ce manuscrit, parfaite illustration de ce précieux conseil que tu m'as donnée en Master 1 : « moins tu écris, moins tu prends le risque d'écrire une erreur ». Je te remercie également pour ce souvenir incroyable d'un vendredi après-midi passé avec Jean-Michel Marin à se battre pour installer Conda sur mon ordinateur. J'en profite pour remercier ce dernier pour tous ces précieux souvenirs que Tiffany et moi avons en commun avec toi et une certaine histoire de tricotage. Je remercie également Elodie, je me sens immensément privilégié d'avoir pu enseigner avec toi, merci pour ton implication, tes nombreux conseils et ton accompagnement dans nos encadrements d'étudiants de Master 1 avec Tiffany. Je dois aussi remercier toutes les personnes que j'ai rencontrées grâce à la popularité de Tiffany: Anne, merci de m'avoir permis de prêcher la bonne parole Mathématique auprès des lycéens de Mende. Monia merci pour ton incroyable gentillesse et ces petits moments chaleureux. Nicolas, je comprends parfaitement que tu sois la personne préférée de Tiffany à l'IMAG et te remercie pour les agréables moments qu'ont toujours été nos discutions. Laurence, merci pour ton attention et tes encouragements, aussi bien pour la thèse que pour le personnel. Enfin, je remercie celles et ceux qui au détour d'un couloir ont toujours eu un sourire et un mot gentil, en particulier Xavier, Ghislain, Ali, Paul, André, Jérémie, Clément et Gwladys, que je remercie aussi pour son implication lors de mes CSI.

Aux (Post)-doctorant.e.s. Je remercie d'abord Thibo, Alan et Tom : vous m'avez intégré dès mon master au groupe des doctorant.e.s, et m'avez laissé de très beaux souvenirs autour d'une bière et d'un billard, merci pour tout ça et votre soutien lors de ma préparation du concours doctoral. Je remercie aussi Morgane, Guillaume, Julien et

Maelli, pour les randonnées, la balade à vélo option crevaison et les sorties ciné. À mes amis sportifs, Aurelio et Emmanuel: Emmanuel, merci pour ta joie de vivre contagieuse et tes précieux conseils pour progresser aux tractions. Aurelio, merci de m'avoir suivi dans mes entrainements, d'avoir fait du Luc-Léger avec moi pour m'aider à m'améliorer et enfin, pour cette délicieuse bouteille d'Amaretto apportée d'Italie. Je remercie ensuite l'ensemble des doctorant.e.s pour les repas/cafés/SemDocs animés : Faustine, Pascal, Juliette, Pablo, Thibault, Thiziri, Victor, Zained, etc. Enfin, je finis par Tanguy : ta venue en stage puis en thèse a été une véritable bouffée d'air frais contre la morosité du distantiel. Merci pour ta bonne humeur, ton aide, tes petites astuces en Python mais surtout, de m'avoir fait découvrir Star Wars : The Clone Wars.

Aux anciens doctorants de Joseph. Mathurin, Quentin et Nidham, merci à vous trois pour ces bons moments passés lors de vos venues à Montpellier. En particulier Mathurin, j'en profite pour te remercier plus personnellement : merci pour tout le temps que tu m'as accordé, de tes précieux conseils pour cleaner mon code, jusqu'à tes remarques sur mon manuscrit. Nul doute que si un jour tu encadres des doctorant.e.s, ces derniers auront une chance incroyable.

Sur les épaules des géants. J'ai maintenant une pensée pour M. Villalongue et M. Palacios. Merci pour votre dévouement extraordinaire, qui a dépassé largement le cadre de la mission qui vous était donnée : merci de tout coeur d'avoir été des modèles pour moi et de m'avoir permis de tenir le coup dans les moments difficiles que j'ai traversés.

À mes amis. Je remercie maintenant Alejandro et Vincent. Alejandro, merci pour ton soutien toute ces années, nos passions partagées pour Star Wars, Marvel et la pizza cholestérol. Vincent, nul doute que j'ai pu être un véritable chemin de croix pour toi lorsque nous étions en L3, pour autant, tu as toujours été d'une patience et d'une gentillesse incroyable, merci pour ces beaux souvenirs et le privilège d'assister à ton ordination.

À ma famille. À toi Papa, merci de l'immense intérêt que tu as porté à mes recherches, à celles de Tiffany et plus généralement à notre vie. Merci aussi d'avoir toujours fait en sorte que les moments passés ensemble soient si agréables et nous ressourcent. Maman, merci de m'avoir appris, très tôt, à travailler pour avoir ce que je veux et de m'avoir soutenu durant toutes ces années. Laureen, malgré nos chamailleries de frère et soeur, je suis heureux qu'on ai toujours fait en sorte de se soutenir l'un l'autre, aussi, je te dis un grand merci et te dédie toutes les fautes d'orthographes en anglais de ce manuscrit. À Nadine, Jean-Luc et à mes grands parents, merci pour la place que vous avez dans ma vie, et de votre soutien aussi bien pour mes études que mes projets d'avenir.

À toi mon Amour. C'est le moment de te remercier et c'est réellement la partie la plus difficile à écrire de cette thèse. Merci, tout d'abord, d'avoir vécu avec moi cette extraordinaire aventure de la « double-thèse », ensuite de ton apport à mon travail: merci pour toutes ces brillantes idées qui me permettent de mettre en valeur mes résultats et m'ont aidé à expliquer mon travail. Merci d'avoir été celle avec qui j'ai pu partager la primeur de mes résultats, d'être ma première lectrice et pour finir ma première spectatrice. Merci de ton soutien sans faille qui me permet aujourd'hui de présenter mes travaux. Connaissant ton amour pour Grey's Anatomy, je te remercie de nous permettre d'être «extraordinaires ensemble, plutôt qu'ordinaires séparément», même si tu es déjà extraordinaire. Je suis heureux de passer cette ligne d'arrivée à tes côtés, pour commencer une nouvelle aventure ensemble.

Gaëtan, c'est à toi que je dédie cette thèse, en cet anniversaire si particulier.

Résumé. Nous présentons un estimateur pour l'ajustement, en grande dimension, d'un modèle linéaire avec interactions quadratiques. Un tel modèle ayant un très grand nombre de variables, son estimation soulève de nombreux défis statistiques et numériques. Ainsi, son estimation a motivé de nombreux travaux ces deux dernières décennies, et reste un enjeu dans de nombreuses applications. Statistiquement, un des enjeux est de pouvoir faire de la sélection de variables, pour faciliter l'interprétabilité du modèle. De plus, les variables d'interactions ajoutées pouvant être fortement corrélées, une régularisation adaptée doit permettre de les prendre en compte. On propose alors d'adapter l'estimateur ElasticNet, pour prendre en compte les potentielles corrélations via la pénalité ℓ_2 et obtenir un modèle parcimonieux via la pénalité ℓ_1.

Aussi, une approche communément utilisée dans la littérature, pour favoriser les effets principaux tout en réduisant le nombre d'interactions à considérer, est l'hypothèse d'hérédité. Cette hypothèse n'autorise à inclure une interaction que si et seulement si les effets principaux associés sont sélectionnés dans le modèle. Ainsi, elle mène à des modèles parcimonieux, plus faciles à interpréter, tout en réduisant le nombre d'interactions à visiter et le coût computationnel. Cependant, elle ne permet pas d'explorer les variables d'interactions dont les effets principaux ne sont pas sélectionnés, alors que ces variables peuvent être pertinentes à considérer. Aussi, on propose de s'affranchir de cette hypothèse structurelle d'hérédité, et de pénaliser davantage les interactions que les effets simples, pour favoriser ces dernières et l'interprétabilité.

Aussi, on sait que les estimateurs pénalisés tels que l'Elastic Net biaisent les coefficients en les réduisant agressivement vers zéro. Une conséquence est la sélection de variables supplémentaires pour compenser la perte d'amplitude des coefficients pénalisés, affectant la calibration des hyperparamètres lors de la validation croisée. Une solution simple est alors de sélectionner les variables par l'Elastic Net, puis d'estimer ces coefficients par l'estimateur des moindres carrés, pour chaque hyperparamètre. Cependant, si les variables sont fortement corrélées, l'étape des moindres carrés peut échouer. Aussi, on choisit d'adapter une méthode de débiaisage permettant d'obtenir simultanément les coefficients de l'Elastic Net et leur version débiaisée.

Un premier enjeu de ce travail est de développer un algorithme qui ne requiert pas de stocker la matrice des interactions, qui peut dépasser la capacité mémoire d'un ordinateur. Pour ce faire, on adapte un algorithme de descente par coordonnées, permettant de construire les colonnes de cette matrice à la volée sans les stocker, mais ajoute des calculs supplémentaires à chaque mise-à-jour d'un coefficient d'interactions, augmentant les temps de calculs. Aussi, sachant que notre estimateur est parcimonieux, ces calculs peuvent être d'autant plus inutiles que beaucoup de coefficients d'interactions sont nuls, et donc inutilement mis à jour. Un second enjeu est de proposer un algorithme qui reste efficace, malgré le grand nombre d'interactions à considérer et ce surcoût de calculs. Par conséquent, afin d'exploiter la parcimonie de l'estimateur et de réduire le nombre de coefficients d'interactions à mettre à jour, on adapte un algorithme d'ensembles actifs. Enfin, on adapte l'accélération d'Anderson, qui permet d'accélérer les algorithmes de descente par coordonnées pour les problèmes type LASSO.

Finalement, les performances de notre estimateur sont illustrées aussi bien sur données simulées que sur données réelles, et comparées avec des méthodes de l'état de l'art.

Mots-clés. Modèle linéaire, Interactions quadratiques, Elastic Net, parcimonie, algorithme d'ensembles actifs, optimisation convexe non-lisse

Abstract. We present an estimator for the high-dimensional fitting of a linear model with quadratic interactions. As such a model has a very large number of features, its estimation raises many statistical and computational challenges. Thus, its estimation has motivated a lot of work over the last two decades, and remains a challenge in many applications. From a statistical point of view, one of the challenges is to be able to select the features, to facilitate the interpretability of the model. Moreover, since the added interaction features can be highly correlated, an adapted regularization must be able to take them into account. We then propose to adapt the Elastic Net estimator, to take into account the potential correlations thanks to the ℓ_2 penalty, and to obtain a parsimonious model using the ℓ_1 penalty.

Moreover, a common approach used in the literature, to favor main effects while reducing the number of interactions to be considered, is the heredity assumption. This assumption allows the inclusion of an interaction only if and when the associated main effects are selected in the model. Thus, it leads to parsimonious models, easier to interpret, while reducing the number of interactions to be visited and the computational cost. However, it does not allow the exploration of interaction variables whose main effects are not selected, although these variables may be relevant to consider. We therefore propose to emancipate ourselves from this structural heredity assumption, and to penalize interactions more than main effects, in order to favor the latter and interpretability.

It is also known that penalized estimators such as Elastic Net bias the coefficients by aggressively shrinking them towards zero. A consequence is the selection of additional features to compensate for the loss of amplitude of the penalized coefficients, which affects the calibration of the hyperparameters during cross-validation. A simple solution is then to select the features by the Elastic Net, then to estimate these coefficients by the Least Squares estimator, for each hyperparameter. However, if the features are highly correlated, the Least Squares step may fail. Therefore, we choose to adapt a debiasing method allowing to obtain simultaneously the Elastic Net coefficients and their debiased version.

A first challenge of this work is to develop an algorithm that does not require to store the interaction matrix, which could exceed the memory capacity of a computer. To do this, we adapt a coordinate descent algorithm, allowing to build the columns of this matrix *on-the-fly*. Although this step avoids storage, it adds extra computations to each step of the algorithm, thus increasing its computation time. Moreover, knowing that our estimator is parsimonious, these computations may be all the more useless as many interaction coefficients are zero, and thus unnecessarily updated. A second issue is then to propose an algorithm that remains computationally efficient, despite the large number of interactions to consider and this computational overhead. Therefore, to exploit the parsimony of the estimator and to reduce the number of interaction coefficients to be updated, we adapt an active set algorithm. Second, we adapt the Anderson acceleration, which allows us to speed up the coordinate descent algorithms for solving LASSO type problems.

Finally, the performance of our estimator is illustrated on simulated and real data, and compared with state-of-the-art methods.

Keywords. Linear Model, Quadratic Interactions, Elastic Net, Sparsity, Active Sets Algorithm, non-smooth convex optimization

Contents

List of Figures

Résumé en Français

Dans cette thèse, nous présentons un estimateur pour l'ajustement, en grande dimension, d'un modèle linéaire avec interactions quadratiques.

Motivations. Pouvoir prendre en compte les effets cocktails entre des variables est un enjeu dans de nombreuses applications, puisqu'ils peuvent donner une meilleure compréhension des phénomènes étudiés. En génomique, par exemple, de nombreux travaux visent à étudier ces effets cocktails entre gènes [Ritchie et al., 2001, Marchini et al., 2005, Park and Hastie, 2008, D'Angelo et al., 2009, Wu et al., 2010, Wang et al., 2014, Wang and Chen, 2018, Vandel et al., 2019, Zrimec et al., 2021], ou entre gènes et environnement [Liu et al., 2013, Figueiredo et al., 2014, Laville et al., 2020, Zhou et al., 2021, Zemlianskaia et al., 2022].

Modèle linéaire avec interactions quadratiques, section 1.1. Aussi, estimer un modèle linéaire avec interactions (Equation (1.1)) est devenu un défi majeur au cours des deux dernières décennies. Cependant, même limité aux interactions quadratiques, un tel modèle considère un très grand nombre de variables additionnelles. Ainsi, son estimation soulève de nombreux défis statistiques et numériques, et ne doit pas se faire au prix de son interprétabilité.

Dans cet objectif, lorsque la matrice des interactions peut être construite et stockée en mémoire, les estimateurs connus pour leur parcimonie tels que le LASSO [Tibshirani, 1996, Chen et al., 1998] ou l'Elastic Net [Zou and Hastie, 2005] peuvent encore être utilisés pour estimer de tels modèles. Cependant, lorsque le nombre de variables augmente, ces approches standards peuvent devenir inutilisables pour deux raisons. La première raison est numérique, puisque la taille de la matrice des interactions peut rapidement dépasser la capacité de mémoire de l'ordinateur (Figure 1.1a). La deuxième raison est statistique, car les variables principales deviennent rapidement minoritaires et sont noyées parmi les variables d'interactions (Figure 1.1b). Ainsi, traiter les effets principaux et les effets d'interaction de la même façon peut donner des modèles ayant detecté seulement quelques effets principaux, parmi de nombreuses interactions sélectionnées, les rendant plus difficiles à interpréter.

Approches avec hypothèses d'hérédité, section 1.2. Pour résoudre les problèmes de régression quadratiques et effectuer la sélection des variables, une approche commune consiste à considérer des hypothèses de structure hiérarchique entre les effets principaux et les interactions. Elles se dérivent principalement en deux versions : l'hérédité forte (Figure 1.2a) et l'hérédité faible (Figure 1.2b). La première n'autorise à inclure une interaction que si et seulement si les effets principaux associés sont sélectionnés dans le modèle, alors que la deuxième n'exige la présence que d'un seul effet principal. De nombreux moyens ont été proposés pour imposer ces structures, notamment des approches basées sur l'optimisation (section 1.2.1), ou des procédures pas à pas (section 1.2.2).

Les approches basées sur un problème d'optimisation [Yuan et al., 2009, Radchenko and James, 2010, Bien et al., 2013, Lim and Hastie, 2015, Haris et al., 2016, Hazimeh and Mazumder, 2020] imposent la structure d'hérédité en ajoutant des contraintes ou des pénalités supplémentaires à un estimateur imposant déjà la parcimonie, tels que le LASSO. Les procédures pas à pas [Park and Hastie, 2008, Hao and Zhang, 2014, Hao et al., 2018] peuvent brièvement se résumer comme suit. Une première étape permet de sélectionner les effets principaux actifs tandis qu'une deuxième étape sélectionne les interactions actives entre les effets principaux sélectionnés à la première étape. Les méthodes basées sur l'optimisation bénéficient de meilleures propriétés statistiques, puisqu'elles considèrent les effets principaux et d'interactions ensemble, alors que les procédures pas à pas considèrent un nombre d'interactions réduit, ce qui est numériquement avantageux.

Ainsi, les hypothèses d'hérédités mènent à des modèles parcimonieux, plus faciles à interpréter, tout en réduisant le nombre d'interactions à visiter et le coût computationnel. Cependant, elles ne permettent pas d'explorer les variables d'interactions dont les effets principaux ne sont pas sélectionnés, alors que ces variables peuvent être pertinentes à considérer. En génomique, par exemple, l'expression des gènes nécessite souvent la présence de protéines coopérantes, c'est-à-dire que la présence d'une seule protéine ne peut pas activer l'expression des gènes [Vandel et al., 2019, Zrimec et al., 2021].

Approches sans hypothèses d'hérédités, section 1.3. En général, aucun a priori n'est connu sur la structure sous-jacente des interactions. Aussi, deux types d'approches ont été développées pour s'affranchir des hypothèses d'hérédité, via des hypothèses supplémentaires sur les données (section 1.3.1), ou via le tri des interactions (section 1.3.2).

Le premier type d'approche [Nakagawa et al., 2015, 2016, Le Morvan and Vert, 2018] repose sur l'hypothèse que les coefficients de la matrice de design sont binaires ou continus dans $[0, 1]$. Grâce à cette hypothèse et en exploitant la structure d'arbre des interactions, ces travaux fournissent différents critères permettant de filtrer les interactions non pertinentes, réduisant le nombre d'interactions à visiter. Bien que ces approches soient numériquement très efficaces, cette hypothèse restreint leur potentiel applicatif.

Le second type d'approche [Fan et al., 2016, Reese et al., 2018] repose sur le tri des interactions selon un certain critère, généralement basé sur la corrélation entre l'interaction et la réponse, pour ne sélectionner que celles ayant un score supérieur à un seuil donné. Si cela est très efficace numériquement, une limite est de ne pas considérer conjointement les effets principaux et les interactions dans un même problème d'optimisation.

Aussi, on décide dans cette thèse de s'affranchir des hypothèses sur la structure d'hérédité et de celles sur les données, avec l'objectif d'estimer l'ensemble des interactions et des variables principales dans un seul problème d'optimisation.

Notre approche, Chapitre 2. Dans cette thèse, nous développons un estimateur pour l'ajustement, en grande dimension, d'un modèle de régression linéaire avec interactions quadratiques (section 2.1). Aussi, pour estimer un tel modèle tout en restant interprétable, on adapte l'estimateur Elastic Net [Zou and Hastie, 2005], pour bénéficier à la fois de la parcimonie grâce à la pénalité ℓ_1 et prendre en compte les potentielles corrélations entre variables grâce à la pénalité ℓ_2 (Equation (2.3)).

Cet estimateur ayant une pénalité composée de 4 termes et de 4 hyper-paramètres à ajuster, on propose une re-paramétrisation de l'Elastic Net avec Interactions (section 2.1.1). Les objectifs sont de réduire le nombre d'hyper-paramètres à ajuster pour réduire le coût computationnel associé, et d'adapter ces pénalités au cas des interactions, notamment en proposant de pénaliser les interactions plus que les effets principaux, comme dans Hao et al. [2018], Hazimeh and Mazumder [2020].

Ensuite, un des enjeux de cette thèse est de développer un algorithme qui ne requiert pas de stocker la matrice des interactions, qui peut dépasser la capacité mémoire d'un ordinateur. Dans cet objectif, on adapte l'algorithme de descente par coordonnées [Tseng, 2001, Friedman et al., 2007], dont le principe consiste à transformer un problème d'optimisation de taille p en p problèmes d'optimisation à 1 dimension (section 2.1.2). Ainsi, il permet de mettre à jour les coefficients d'interactions un par un, permettant de construire des colonnes de la matrice des interactions à la volée sans jamais avoir besoin de la stocker entièrement en mémoire (Algorithme 2). Néanmoins, l'inconvénient de cette approche est que la mise à jour de chaque coefficient d'interactions nécessite des calculs supplémentaires, augmentant le temps de calcul de l'algorithme. Par ailleurs, on sait que la solution de l'Elastic Net avec Interactions est parcimonieuse, *i.e.,* de nombreux coefficients d'interactions sont nuls à la solution et donc leurs mises à jour sont d'autant plus inutiles.

Les premiers résultats statistiques illustrent que pénaliser les interactions plus que les effects principaux améliore significativement les performances de sélection de variables (Figures 2.2 et 2.3) sans déteriorer les performances prédictives (Figure 2.1, section 2.1.3).

Débiaisage d'estimateurs pénalisés, section 2.2. Aussi, il est connu que les estimateurs pénalisés tels que l'Elastic Net biaisent les coefficients en les réduisant agressivement vers zéro (Figure 1.4). Une conséquence est la sélection de variables supplémentaires pour compenser la perte d'amplitude des coefficients pénalisés, affectant la calibration des hyper-paramètres lors de la validation croisée.

Dans un premier temps, on rappelle une solution simple [Efron et al., 2004, Belloni and Chernozhukov, 2013, Lederer, 2013] qui est de sélectionner les variables par l'Elastic Net, puis d'estimer les coefficients associés par l'estimateur des moindres carrés, pour chaque hyper-paramètre (section 1.4). Bien que cette approche soit assez simple, elle souffre de plusieurs inconvénients. Le premier vient du fait que, si les variables actives identifiées par l'Elastic Net sont fortement corrélées, l'étape des moindres carrés peut échouer, car elle est censée être utilisée sur une matrice de plein rang. Néanmoins, l'inconvénient principal est la complexité du pipeline permettant de réaliser une telle méthode, puisqu'il doit être fait sur toute la grille des hyper-paramètres testés par l'Elastic Net, et pour chaque sous-ensemble de la procédure de validation croisée.

Pour ces raisons, on choisit d'adapter une méthode de débiaisage permettant d'obtenir simultanément les coefficients de l'Elastic Net avec Interactions et leur version débiaisée. Pour ce faire, nous adaptons l'estimateur CLEAR (Covariant LEAst-square Reffiting, [Deledalle et al., 2017]), à l'Elastic Net avec Interactions, appelé CLEAR-Enet avec Interactions par la suite (section 2.2.2). Le principal intérêt d'adapter CLEAR à notre contexte est qu'il préserve les propriétés de l'Elastic Net, notamment la parcimonie. L'objectif est alors de déterminer un hyper-paramètre amenant à sélectionner moins de coefficients que l'Elastic Net avec Interactions pour une erreur de prédiction similaire, afin de faciliter l'interprétabilité du modèle. Ensuite, l'estimateur CLEAR nécessitant le calcul de la Jacobienne de l'Elastic Net avec Interactions, on adapte un schéma de différenciation automatique pour calculer efficacement cette dernière (section 2.2.3).

Enfin, les résultats statistiques montrent que l'étape de débiaisage ne diminue jamais les performances de l'Elastic Net avec Interactions (section 2.2.4). Au contraire, pour certains scénarios de standardisation, CLEAR-Enet avec Interactions améliore même significativement les performances de sélection de variables (Figures 2.4 et 2.5). En effet, pour une erreur prédictive similaire, ce dernier diminue le nombre de faux positifs de l'Elastic Net avec Interactions, *i.e.,* réduit le nombre de variables sélectionnées à tort par l'estimateur. Aussi, on observe que le temps de calcul de CLEAR-Enet avec Interactions est environ le double de celui de l'Elastic Net avec Interactions (Figure 2.6). Finalement, ces résultats montrent également que pénaliser les interactions plus que les effets principaux permet de réduire le temps de calcul dans tous les cas.

Algorithmes d'ensembles actifs, Chapitre 3. Ensuite, le second enjeu numérique de cette thèse est d'exploiter la parcimonie de l'Elastic Net avec Interactions, afin de limiter autant que possible le nombre de variables d'interactions à mettre à jour et les calculs supplémentaires associés.

Pour ce faire, des algorithmes d'ensembles actifs [Fan et al., 2008, Kim and Park, 2010, Boisbunon et al., 2014, Johnson and Guestrin, 2015, Massias et al., 2017, 2018, Bertrand et al., 2022] ont été développés. Ces derniers visent à trier puis sélectionner un sous-ensemble de variables, résoudre le sous-problème associé, puis à tester si le problème global est résolu. Si le sous-ensemble de variables sélectionnées contient toutes les variables actives attendues, l'algorithme s'arrête, sinon, la procédure recommence sur un sous-ensemble plus grand de variables (Figure 1.7). Des algorithmes d'ensembles actifs ont déjà été employés dans la littérature sur les interactions [Hazimeh and Mazumder, 2020, Le Morvan and Vert, 2018], cependant ces travaux reposent soit sur l'hypothèse d'hérédité, soit sur une hypothèse sur la matrice de design.

Aussi, pour adapter cette approche au cas des interactions en s'affranchissant de ces hypothèses, on adapte CELER [Massias et al., 2018]. Pour limiter le plus possible les coûts de calcul associés aux interactions (Algorithme 7), nous proposons de différencier les ensembles de travail des effets principaux et des interactions (section 3.2.2), et d'évaluer des heuristiques d'arrêts entre chaque itération (section 3.2.3). En effet, un enjeu est d'éviter le cas critique où toutes les interactions ont été identifiées mais pas tous les effets principaux, entraînant un coût computationnel élevé sur les interactions alors que les effets principaux sont moins coûteux à explorer. Aussi, si CELER vérifie l'optimalité du problème global à chaque itération de l'algorithme, ces calculs sont prohibitifs dans notre cas, puisqu'ils requièrent de visiter toutes les interactions (Figure 3.1b). On propose alors d'évaluer des heuristiques d'arrêts entre chaque itération, afin de ne pas calculer trop tôt et inutilement le critère d'arrêt du problème global.

On adapte l'accélération d'Anderson [Anderson, 1965, Scieur et al., 2016, Zhang et al., 2020, Mai and Johansson, 2020, Bertrand and Massias, 2021], qui permet d'accélérer les algorithmes de descente par coordonnées pour les problèmes type LASSO (section 3.3).

Finalement, on compare les performances numériques de notre approche à scikit-learn et CELER , via Moreau et al. [2022], qui permet de comparer différents solveurs pour un même problème d'optimisation (section 3.4). On note que stocker la matrice des interactions, en grande dimension, augmente le temps de calcul par rapport aux algorithmes qui la construisent à la volée. Aussi, si l'accélération d'Anderson réduit le nombre d'itérations nécessaires pour atteindre la solution optimale, son coût computationnel peut être trop grand par rapport au gain observé. Finalement, sur nos données génomiques (section 1.6.2) c'est l'approche combinant l'accéleration d'Anderson et les ensembles actifs qui est la plus rapide, avec des performances similaires à celles de CELER.

Performances Statistiques, Chapitre 4. Pour finir, on compare les performances statistiques de notre estimateur à celles d'HierNet [Bien et al., 2013] et de RAMP [Hao et al., 2018], deux estimateurs imposant les structures de hiérarchies fortes et faibles, le premier via un problème d'optimisation et le deuxième via une procédure pas à pas. Pour ce faire, on compare 5 différents scenarios d'hérédité (Figures 1.2 et 1.3), aussi bien sur des données semi-simulées (section 4.1), que sur données réelles (section 4.2).

Sur données semi-simulées, les resultats montrent que notre approche fournit les meilleures performances prédictives dans tous les cas (Figures 4.1, 4.4 et 4.7).

Concernant la sélection des variables, les simulations montrent qu'on obtient des performances: meilleures que celles d'HierNet et comparables à celles de RAMP, dans les scénarios de hiérarchie fortes et faibles (Figures 4.2, 4.5 et 4.8). Cependant, on observe des comportements légèrement différents. En effet, RAMP sélectionne un petit nombre de variables actives, menant à sélectionner peu de faux positifs, mais manque certaines variables importantes en contrepartie. À l'inverse, notre estimateur sélectionne plus de variables, privilégiant l'ajout de faux positifs au fait de manquer certaines variables importantes. Enfin, HierNet sélectionne plus de variables et oublie moins de faux positifs que l'Elastic Net avec Interactions, mais il inclut beaucoup plus de faux positifs que notre approche manque de variables importantes.

Sur le plan computationel, la première simulation montre qu'HierNet n'est pas compétitif, tandis que notre approche sans débiaisage apparaît être parmi les plus rapides (comparée aux différentes versions de RAMP), sur l'ensemble des simulations (Figures 4.3, 4.6 et 4.9).

Sur données réelles, on montre que notre méthode obtient le meilleur score de précision (Figure 4.10), et que l'étape de débiaisage réduit drastiquement le nombre de variables actives dans certains scénarios de standardisation (Figure 4.11). Enfin, on note que considérer le maximum entre interactions à la place du produit améliore les perfomances prédictives sur nos données génomiques, et peut avoir plus de sens au regard de l'application étudiée.

Chapter 1

Introduction

Contents

Targeting cocktail effects between features has become a main challenge in many applications over the last twenty years, as it can provide new insights. In genomics, for example, numerous works aim at studying such cocktail effects between genes [Ritchie et al., 2001, Marchini et al., 2005, Park and Hastie, 2008, D'Angelo et al., 2009, Wu et al., 2010, Wang et al., 2014, Wang and Chen, 2018, Vandel et al., 2019, Zrimec et al., 2021] or between genes and environment [Liu et al., 2013, Figueiredo et al., 2014, Laville et al., 2020, Zhou et al., 2021, Zemlianskaia et al., 2022]. However, interactions are not only of interest for genomic applications. More recently, in their studies on food additives, the authors of Chazelas et al. [2020, 2021] explain in the conclusion that cocktail effects, *i.e.,* interactions between the main effects they have selected, should be explored.

Nonetheless, targeting such interactions should not be done at the cost of the model interpretability. For example, the Random Forest method [Breiman, 2001] easily allows exploring interactions, since using a tree with a depth of k means considering the interactions of order k between k features. However, it is admitted that these methods perform poorly as a feature selection method and do not provide an interpretable model.

Due to their interpretability, linear models have been widely used for many learning tasks. Also, developing a sparse linear model estimator which takes account the quadratic interactions between features has become a challenge over the past two decades, since it raises many statistical and optimization challenges. Most works in this area assume either a heredity assumption between main effects and interactions effects, or that the design matrix is binary.

In this thesis, we aim to develop an estimator for a linear model with quadratic interactions, for any design matrix, and which is not based on any structural assumptions between main effects and interactions. Such estimator must provide sparse estimate to be interpretable, while being computationally tractable.

The aim of this chapter is to present the linear model with quadratic interactions and the associated challenges in section 1.1. Then, section 1.2 presents the heredity assumptions and the estimators of the linear model with interactions based on them. Also, section 1.2.1 details the approaches based on optimization problems while section 1.2.2 describe the one based on stage-wise procedures. Then, section 1.3 details the methods that are free from the heredity hypothesis, based on assumptions on the design matrix or based on a sorting criterion. We present our approach in section 1.3.3, based on a penalized estimator without any hypothesis. Since regularized methods suffer from a bias due to the penalties, section 1.4 details the impact of such bias and presents a simple approach to correct it. Moreover, section 1.5 gives the optimization techniques allowing to solve our optimization problem and to accelerate it. Finally, section 1.6 presents the gene regulation problem and the associated statistical problems that challenges us.

1.1 Linear Model with interactions

If the standard linear model does not naturally take into account cocktail effects, it is possible to extend it to target quadratic interactions between the variables. Therefore, we recall the formulation of the linear model with second order interactions. Denoting n the number of samples of a dataset, p the number of main features and $q = \frac{p(p+1)}{2}$ the number of quadratic interactions (including pure quadratic effects, *i.e.,* interaction of one feature with itself), we tackle the following quadratic regression problem:

$$y = \beta_0 \mathbb{1}_n + X\beta + Z\Theta + \varepsilon \ , \tag{1.1}$$

where we denote by $X = [x_1, x_2, \ldots, x_p] \in \mathbb{R}^{n \times p}$ and $\beta \in \mathbb{R}^p$ the main effects design matrix and the associated coefficients vector. Let $\odot$ be a non-additive element-wise operation, we note $z_{jj} = x_{j_1} \odot x_{j_2}$ an interaction between x_{j_1} and x_{j_2} main effects. Then, we note $Z = [z_1, z_2, \ldots, z_q] = [x_1 \odot x_1, x_1 \odot x_2, \ldots, x_{p-1} \odot x_p, x_p \odot x_p] \in \mathbb{R}^{n \times q}$ and $\Theta \in \mathbb{R}^q$ the interactions design matrix and associated coefficients vector, $y \in \mathbb{R}^n$ the response vector, $\mathbb{1}_n \in \mathbb{R}^n$ the vector of size n with only 1's, $\beta_0 \in \mathbb{R}$ the intercept, and ε some Gaussian noise.

Nonetheless, when targeting k-order interactions effects between features, linear model quickly include too many features, and it is painstaking to interpret the relevance of each of them. Indeed, even limited to pairwise effects, considering a linear regression model with second order interactions implies estimating a quadratic number of coefficients, which quickly brings statistical and computational challenges.

Different interactions operations. While the most commonly used interaction in the literature is the element-wise product, any other non-additive operation is possible, such as the element-wise maximum or minimum, for example.

Naive approaches. Of course, for a moderate number p of main features, the interactions design matrix Z can be build and stored in memory. Then, statistical inference can be performed with standard tools developed for linear model as LASSO [Tibshirani, 1996, Chen et al., 1998], Ridge [Tikhonov, 1943, Hoerl and Kennard, 1970] or Elastic Net [Zou and Hastie, 2005]. In such case, it amounts to solving the following optimization problem:

$$\widehat{\beta}_0, \widehat{\beta}, \widehat{\Theta} \in \underset{\beta_0 \in \mathbb{R}, \beta \in \mathbb{R}^p, \Theta \in \mathbb{R}^q}{\arg\min} \frac{1}{2n} \left\| y - \beta_0 \mathbb{1}_n - X\beta - Z\Theta \right\|_2^2 + \mathrm{pen}\left(\beta, \Theta, \alpha_1, \alpha_2\right) \ , \tag{1.2}$$

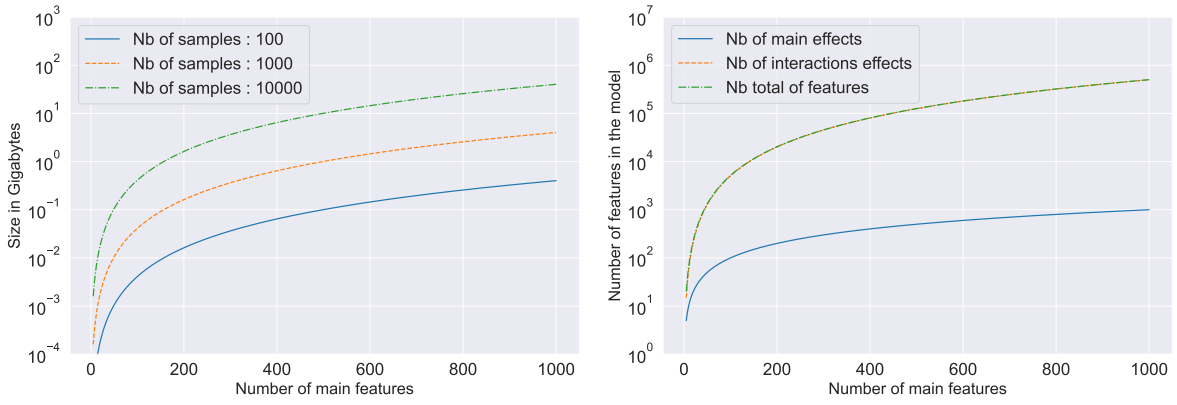
where the penalties are defined respectively for the LASSO, Ridge and Elastic Net in Equations (1.3) to (1.5) as:

$$\text{pen}_{\ell_1}(\beta, \Theta, \alpha_1, \alpha_2) = \alpha_1 \left(\|\beta\|_1 + \|\Theta\|_1 \right) \quad , \tag{1.3}$$

$$\text{pen}_{\ell_2}(\beta, \Theta, \alpha_1, \alpha_2) = \alpha_2 \left(\|\beta\|_2^2 + \|\Theta\|_2^2 \right) \quad , \tag{1.4}$$

$$\text{pen} \quad (\beta, \Theta, \alpha_1, \alpha_2) = \alpha_1 \left(\|\beta\|_1 + \|\Theta\|_1 \right) + \frac{\alpha_2}{2} \left(\|\beta\|_2^2 + \|\Theta\|_2^2 \right) \quad . \tag{1.5}$$

Scaling problem. However, as the number p of main features increases, these naive approaches may became numerically unfeasible, as the size of the interactions matrix Z may quickly exceed the memory capacity of the computer. For example, Figure 1.1a shows that for a dataset that contains $n = 1\,000$ samples and $p = 1\,000$ features, the main and interactions design matrix require more than 1 Gb to be stored. Another striking example is the Leukemia classification dataset [Golub et al., 1999], which is widely used to test optimization algorithms. While this dataset only has $p \approx 7\,000$ features for about $n \approx 70$ samples, the number of interactions is $q \approx 24 \times 10^6$, and the overall design matrix to store is around 14 Gb.



(a) Size of the design matrix

(b) Number of features.

Figure 1.1: Evolution of the size of main and interactions matrix on the left while right-hand side illustrates the evolution of the number of features to be estimated. Interactions design matrix quickly increases and main effects are drowned out by interactions.

Features selection and model interpretability. For all that follows, we call an *active feature* a feature whose estimated coefficient is non-zero, and the set of all active features is called *the support*. Statistically, another problem is that the proportion of interactions to the total number of features tends to one, as represented in Figure 1.1b, hence main effects become a minority. Thus, treating main effects and interaction effects at the same level can lead to models with only a few active main effects, among many active interactions, leading to more difficult to interpret and unstable models.

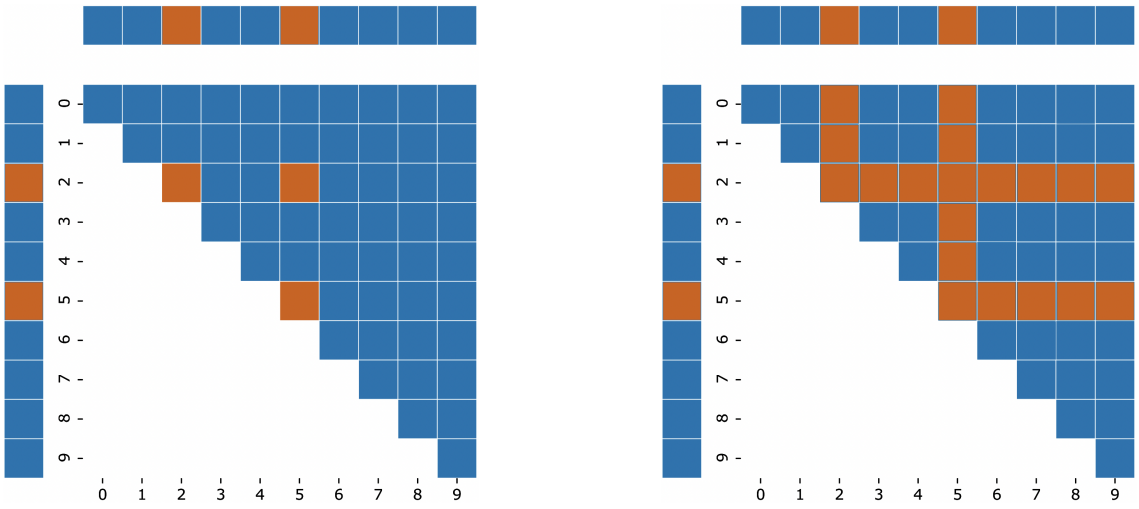
1.2 Structural assumptions for feature selection

Therefore, to tackle quadratic regression problems and perform variable selection, a common approach is to consider hierarchical structure assumptions between main and interactions effects: principle of heredity or marginality [Nelder, 1977, Peixoto, 1987, Hamada and Wu, 1992, Chipman, 1996]. Such heredity assumption assesses that an interaction feature may be selected in the model if both main effects are selected (Equation (1.6)) or if at least one (Equation (1.7)) of the main effects is selected. The former is called strong heredity while the latter is called weak heredity.

$$\Theta_{i,j} \neq 0 \Rightarrow \beta_i \neq 0 \text{ and } \beta_j \neq 0 \ , \tag{1.6}$$

$$\Theta_{i,j} \neq 0 \Rightarrow \beta_i \neq 0 \text{ or } \beta_j \neq 0 \ . \tag{1.7}$$

As illustrated in Figure 1.2, these assumptions drastically reduce the number of interactions to consider and lead to sparse models, which encourage main effects and remain easy to interpret. In particular, in the strong hierarchy setting illustrated by Figure 1.2a, if k main effects are active, only $\frac{k(k+1)}{2}$ second order effects can be active; here in Figure 1.2a there are three possible interactions. While, for weak hierarchy setting, there are $\sum_{i=0}^{k}(p-i)$ possibles actives interactions, here in Figure 1.2b: 19 possible interactions.



(a) Strong Heredity (b) Weak Heredity

Figure 1.2: Graphical illustration of the strong and weak heredity assumptions with ten main effects (scored from 0 to 9), two of which are active in orange, while the blue ones represent null coefficients. The upper-right matrix represents the state of the interaction coefficients (vector Θ), where active coefficients are highlighted in orange, while zero coefficients are in blue. We observe that strong heredity (a) is much more restrictive than weak (b), leading to models with fewer possible interactions.

Many ways have been proposed to enforce these structures, including optimization-based approaches, stage-wise procedures or Bayesian approaches [Chipman, 1996, Thanei et al., 2018].

The former enforces hierarchy by adding supplementary constraints or penalties to the problem, which leads to solve an optimization problem considering main and interaction effects altogether [Yuan et al., 2009, Radchenko and James, 2010, Bien et al., 2013, Lim and Hastie, 2015, Haris et al., 2016, Hazimeh and Mazumder, 2020].

Alternatively, stage-wise procedures can be briefly summarized as a first step aiming to discover active main effects, and a second step aiming to detect active interactions between active main effects only [Park and Hastie, 2008, Hao and Zhang, 2014, Hao et al., 2018]. The main practical advantage of the latter approach is that the number of interactions to consider is drastically reduced, which is numerically advantageous.

However, optimization-based approaches benefit from better statistical properties, since they consider main and interaction effects altogether.

1.2.1 Optimization based approaches

A large amount of works exists in the literature, to enforce strong or weak hierarchy, with a constrained or penalized problem. For instance, Yuan et al. [2009] adapt the non-negative garrote of Breiman [1995] by adding a supplementary constraint to enforce both strong or weak hierarchy, while to estimate linear and non-linear regression model, Radchenko and James [2010] provide a penalized problem enforcing hierarchical constraint.

Alternatively, the authors of GLINTERNET [Lim and Hastie, 2015] developed an estimator to target regression and binary classification problem, using group lasso penalties to enforce strong hierarchy, with each group consisting of two main features and associated interactions.

The work called FAMILY [Haris et al., 2016] tackles regression and logistic regression, to enforce both strong and weak hierarchy, with a penalty using three terms. A specificity of their work is that, given two design matrices X_A and X_B, they only consider interactions between X_A and X_B, but not interactions of X_A or X_B with itself.

While these approaches naturally lead to a convex optimization problem, the authors of HierScale [Hazimeh and Mazumder, 2020] first transpose the strong hierarchy constraint into a *mixed integer program* (MIP) problem. However, this problem being NP-Hard, and difficult to scale, they propose to solve a convex relaxation of it.

We decide to compare our work with previous works [Bien et al., 2013], since to the best of our knowledge, they are the most used in the literature and will be suitable for comparison in the following.

HierNet [Bien et al., 2013]. To enforce a hierarchical structure, in regression setting, HierNet solves the following constrained and regularized convex problem. Unlike our notation (eq. (1.1)), the interaction coefficients are represented by a matrix, denoted by $\Theta \in \mathbb{R}^{p\times p}$, in this problem:

$$\underset{\substack{\beta_0\in\mathbb{R},\beta^{\pm}\in\mathbb{R}^p,\\ \Theta\in\mathbb{R}^{p\times p}}}{\arg\min} \frac{1}{2}\left\| y - \beta_0 \mathbb{1}_n - X(\beta^+ - \beta^-) - Z\frac{\mathrm{Vec}(\Theta)}{2}\right\|_2^2 + \alpha \mathbb{1}_n^\top(\beta^+ - \beta^-) + \frac{\alpha}{2}\|\Theta\|_1 \quad , \tag{1.8}$$

$$\text{s.t. } \Theta = \Theta^\top, \ \forall j \in [\![1,p]\!], \ \begin{cases} \|\Theta_j\|_1 \leqslant \beta_j^+ + \beta_j^- \quad , \\ \beta_j^+ \geqslant 0, \ \beta_j^- \geqslant 0 \quad , \end{cases}$$

where β^+ represents the positive part of the main effects, and β^- the negative one.

Thanks to this formulation, HierNet can enforce both strong and weak heredity. The difference is that when a strong heredity constraint is enforced, the interaction coefficients matrix Θ must be symmetric, whereas the symmetric constraint of Equation (1.8) is removed when weak heredity is enforced. To solve their optimization problem in a weak heredity setting, they develop a generalized gradient descent algorithm. While, to solve the strong heredity constraint, they develop an Alternating Direction Method of Multipliers (ADMM, [Boyd et al., 2011]) algorithm, where the generalized gradient descent of the weak constraint, slightly modified, is used as inner solver. Moreover, an interesting point for comparison with us is that they proposed an Elastic Net version of their work.

1.2.2 Stage-wise procedures

As mentioned above, another way used to enforce heredity is to apply stage-wise procedures. The main advantage of this type of procedure is its computational efficiently, since it considers an interaction feature only if the associated main effects have been selected.

For example, in Park and Hastie [2008], the authors propose to tackle binary classification with interaction through a modified logistic regression, with a supplementary ℓ_2 norm. To obtain a sparse and interpretable estimate, which the Ridge penalty does not provide, they propose to estimate such models using forward selection step, followed by a backward deletion step.

Another example by Hao and Zhang [2014], who proposed IFORT and IFORM, stage-wise approaches based on forward selection method with Least Squares estimators. The first is a two-stage procedure, which first selects the main effects through a forward

procedure, and then, builds the interaction from the main effects selected in the first step. A second stage of forward selection is done on the set of all possible main effects (*i.e.,* not only the ones active at the first stage) and the interactions allowed according to the heredity structure. IFORM is a forward approach which selects the main and the interactions effects altogether. The forward step starts by selecting a main effect, and then selects interactions according to the main effects possibly added earlier.

The main advantage of these procedures is that they explore only the interactions of the quadratic features that respect the structure of heredity, thus allowing the design matrix of this small number of variables to be stored.

Then, we take interest in a latter approach from the same authors, called RAMP, which is a path-stage method, that we describe in the following.

RAMP [Hao et al., 2018]. In their paper, the authors propose RAMP, a path-stage procedure based on LASSO to tackle quadratic regression problem. Their approach is based on a modified LASSO estimator, which includes the interactions features at the k-th alpha on the regularization path from the non-zero main features at the previous alpha.

Let us define the sets $\mathcal{A}_\beta^{k-1}$ and $\mathcal{A}_\Theta^{k-1}$: the index of active main effects and interactions at the $k-1$ step. They define $\mathcal{M}^{k-1}$ the set of main effects which are parents of an active interaction at α_{k-1} associated problem. Immediately, by heredity structure, $\mathcal{A}_\beta^{k-1} \subseteq \mathcal{M}^{k-1}$. They also define $\overline{\mathcal{M}}^{k-1} = [\![1, p]\!] - \mathcal{M}^{k-1}$, the set of indices which are not active and which are not parents of an active interaction.

Finally, they define the set of possible interactions at step k by $\mathcal{I}^k$, according to the heredity structure and the active main effects in $\mathcal{A}_\beta^{k-1}$. Then, they solve the following optimization problem, to find $\mathcal{A}_\beta^{k-1}$ and $\mathcal{A}_\Theta^{k-1}$:

$$\underset{\beta_0,\beta,\Theta_{\mathcal{I}^{k-1}}}{\arg\min} \frac{1}{2n} \left\| y - \beta_0 \mathbb{1}_n - X\beta - Z_{\mathcal{I}^{k-1}} \Theta_{\mathcal{I}^{k-1}} \right\|_2^2 + \alpha_k \left(\left\| \beta_{\overline{\mathcal{M}}^{k-1}} \right\|_1 + \left\| \Theta_{\mathcal{I}^k} \right\|_1 \right) \quad . \tag{1.9}$$

Afterwards, once they obtain the support of the main and the interaction effects, they perform a Least Squares step on both, to finally estimate coefficients of the active features.

As for IFORM, a key advantage of this approach is that it is computationally efficient, since it does not explore all possible interactions but only those that have at least one active main effect (weak heredity). Moreover, in their R implementation, they propose an optional feature to penalize interactions more than main effects, as we have proposed in this work, in Chapter 2.

1.3 Approaches without heredity hypothesis

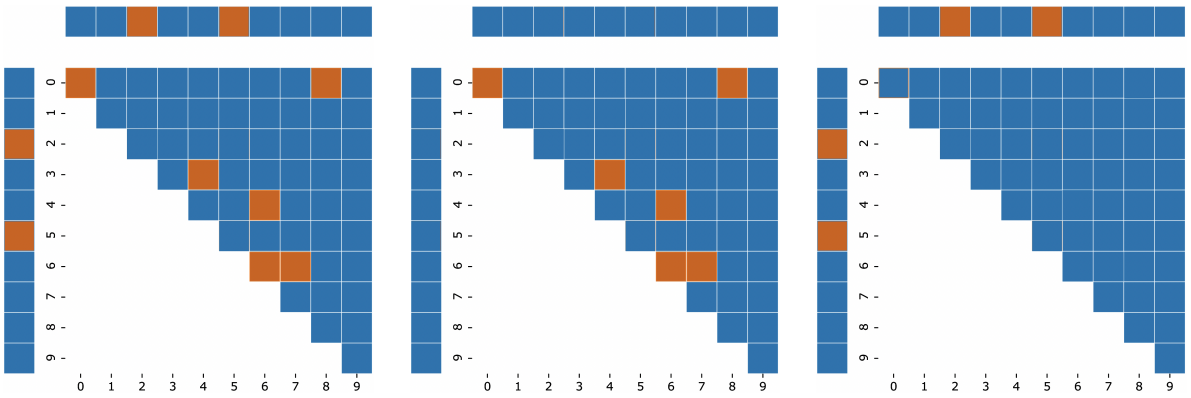
Nevertheless, while heredity structures can lead to sparse (interpretable) models, they do not allow us to easily explore interaction variables whose main effects are not selected, which may nonetheless be relevant. In genomics, for example, gene expression often requires the presence of cooperating proteins (*i.e.,* the presence of a single protein cannot activate gene expression) [Vandel et al., 2019, Zrimec et al., 2021].

In particular, there are three other heredity scenarios in the literature (see for example Tibshirani et al. [2012]), which are used to test the performance of the different quadratic estimators: anti-heredity, interactions only, and main effects only, as depicted in Figure 1.3. For example, anti-heredity settings (1.10) assume that an interaction is non-zero if and only if both associated main effects are zeros, while interaction only (1.11) and main only (1.12) indicate that only interaction are active, with all main effects being zeroed for the former and inversely for the latter.

$$\Theta_{i,j} \neq 0 \Rightarrow \beta_i = \beta_j = 0 \ , \tag{1.10}$$

$$\exists\ (i,j)\ \in [\![1,p]\!],\ \Theta_{i,j} \neq 0 \text{ and } \forall\ i \in [\![1,p]\!],\ \beta_i = 0 \ , \tag{1.11}$$

$$\exists\ i \in [\![1,p]\!],\ \beta_i \neq 0 \text{ and } \forall\ (i,j)\ \in [\![1,p]\!],\ \Theta_{i,j} = 0 \ . \tag{1.12}$$



(a) Anti-heredity (b) Interactions Only (c) Main Effects Only

Figure 1.3: Others heredity settings used in simulation studies.

In general, no prior is known about the underlying interactions structure. In particular, depending on the application, the underlying structure is probably a mixture of the five possible interaction structures.

This observation leads us to attempt to estimate a quadratic regression model without any further heredity assumption, leading to consider a more difficult problem to solve. We first describe approaches which rely on data hypothesis without heredity assumptions, while we secondly describe ranking methods and lastly our approach.

1.3.1 Approaches based on data structure

In the case where $x_{i,j}$ belongs to $\{0,1\}$ or $[0,1]$, to reduce the number of interactions to visit in a linear model with k-order interactions, with $k \geqslant 2$, the authors of Nakagawa et al. [2015, 2016] suggest exploiting the tree structure of interactions. The interactions of order k form a tree of depth k, whose roots are constituted by the main effects, while the leaves of the tree are constituted by the interactions of order k. Moreover, the set of interactions of order $k'+1$ descending from an interaction of order k', is a tree branch. They developed a pruning criterion for the interaction's tree such that an interaction and all its descendants are sure to be non-active variables, thus reducing the number of interactions to visit. This idea also inspired WHInter [Le Morvan and Vert, 2018], an efficient LASSO solver for the linear quadratic model, in the specific case where $x_{i,j}$ belongs to $\{0,1\}$.

1.3.2 Ranking methods

Still in the idea of efficiently discarding interaction features without any structural or data assumptions, another approach aims to rank each feature according to some criteria, and then discarding all features with a score lower than some constant.

For example, the authors of *interaction pursuit* [Fan et al., 2016], abbreviated IP, aim to develop a criterion to rank main and interactions features, without the need to visit each interaction. They use the marginal correlation between pure quadratic effects $x_{j_1}^2$ and the squared response y^2, *i.e.,* they compute p terms of $\mathrm{corr}(x_j^2, y^2)$ instead of computing q terms $\mathrm{corr}(x_{j_1} \odot x_{j_2}, y)$. Then, IP selects all the interactions features $x_{j_1}x_1, \ldots, x_{j_1}x_p$ whose correlation $\mathrm{corr}(x_j^2, y^2)$ is higher than a threshold to be determined.

Another similar approach can be found in [Reese et al., 2018], where the authors develop a method for ultra-high dimensional data called *Joint Cumulant Interaction Screening*, abbreviated JCIS. The JCIS method consists in computing a ranking criterion for each interaction, inspired by the Pearson correlation, to select the features whose rank is higher than a threshold, which remains to be determined. However, these methods do not allow estimating the associated coefficients, but can be used as a pre-processing step, to estimate a subset of coefficients with Least Squares or LASSO type estimator, for example.

The main advantage of these approaches is that they are very efficient and that theoretical properties prove their screening consistency. However, as for stage-wise procedure, the approaches which evaluate main effects and interactions altogether are considered to have better statistical properties.

1.3.3 Our approach

Consequently, in this thesis, we aim to develop an estimator of linear regression model with second-order interactions, which does not rely on any structural or data assumptions and which considers both main and interactions effects in a single optimization problem. With this idea in mind, we aim to develop an Elastic Net with Interactions estimator to tackle the quadratic regression problem, which depends on four hyperparameters: $\alpha = (\alpha_{1,1}, \alpha_{1,2}, \alpha_{2,1}, \alpha_{2,2})$, associated to the following penalty:

$$\text{pen}\left(\beta, \Theta, \alpha\right) = \alpha_{1,1} \left\|\beta\right\|_1 + \alpha_{1,2} \left\|\Theta\right\|_1 + \frac{\alpha_{2,2}}{2} \left\|\beta\right\|_2 + \frac{\alpha_{2,2}}{2} \left\|\Theta\right\|_2 \quad . \tag{1.13}$$

We will sometimes abbreviate Elastic Net by Enet in figures, table and algorithms. By doing so, we will have a flexible penalty which allow us to notably manage the sparsity of main effects and interaction, separately. However, finding four hyperparameters can be time-consuming, thus, in Chapter 2, we parametrize the penalty to easily tune it.

1.4 Debiasing regularization estimator: a naive way

While penalized estimators are efficient to deal with over-parametrized models or overly correlated features, they also bias the estimated coefficients as they shrink large coefficients aggressively toward zero [Hastie et al., 2009, Haris et al., 2016, Salmon, 2017]. Here we detail the consequences of such bias and a naive approach to address it. The first consequence is that the amplitude of the active features is not fully recovered, which is often compensated by including wrongly more features. This may lead to adding too many false positive and downgrade the selection ability of the estimator. In particular, Elastic Net as a trade-off between LASSO and Ridge estimators, suffers from the bias of both penalties. We detail the 1D case for each estimator, which are also illustrated in Figure 1.4:

$$f_{lasso}(x, \alpha_1) = \text{sign}(x) \max(|x| - \alpha_1, 0) \quad , \tag{1.14}$$

$$f_{ridge}(x, \alpha_2) = \frac{x}{1 + \alpha_2} \quad , \tag{1.15}$$

$$f_{enet}(x, \alpha_1, \alpha_2) = \frac{\text{sign}(x) \times \max(|x| - \alpha_1, 0)}{1 + \alpha_2} \quad . \tag{1.16}$$

As shown in Figure 1.4, the LASSO penalty (1.14) shrinks the coefficients aggressively toward zero between $[-\alpha_1, \alpha_1]$, then contracts coefficients from a factor α_1. Alternatively, near the origin, Ridge penalty eq. (1.15) contracts the coefficients less, but more than LASSO for large coefficients. Hence, as a mixture between these two penalties Equation (1.16), Elastic Net is more biased, as observed in Figure 1.4.

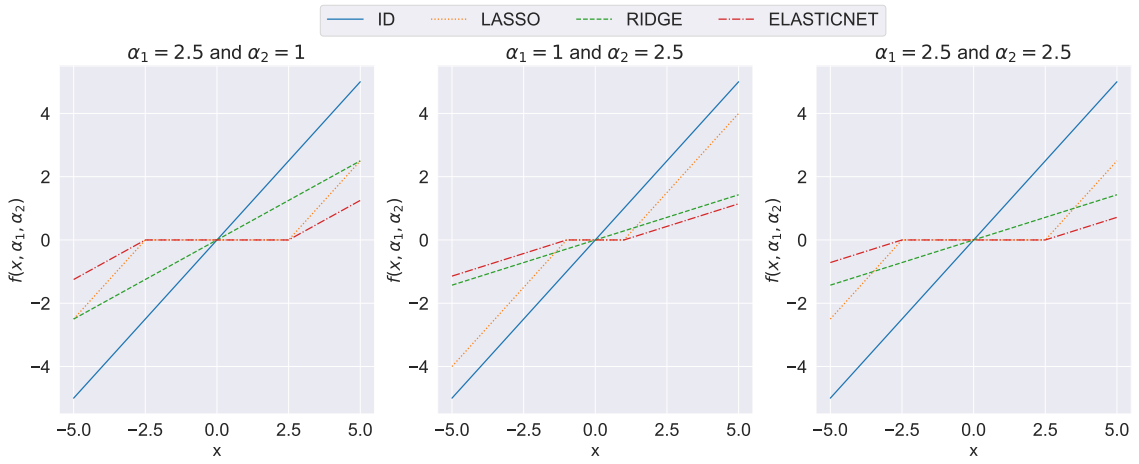


Figure 1.4: LASSO, Ridge and Elastic Net penalties in orthogonal cases, *i.e.*, $X^\top X = \mathrm{Id}_1$. For values of x close to 0, LASSO penalizes more than Ridge, while the opposite trend is observed after. In all cases, Elastic Net penalizes always more.

To avoid deteriorating the feature selection ability of such a method, a solution consists of a two-stage procedure [Efron et al., 2004, Belloni and Chernozhukov, 2013, Lederer, 2013], where the regularization method aims at selecting the set of active features and then, in a second step, a Least Squares step is performed to estimate their coefficients, as detailed in Equation (1.17). This method is known as Naive-LSEnet.

$$\widehat{\beta}_0^{LS}, \widehat{\beta}^{LS}, \widehat{\Theta}^{LS} \in \underset{\substack{\beta_0 \in \mathbb{R}, \beta \in \mathbb{R}^p, \Theta \in \mathbb{R}^q, \\ \mathrm{supp}(\beta) = \mathrm{supp}\left(\widehat{\beta}^{Enet}_{\alpha_1,\alpha_2}\right) \\ \mathrm{supp}(\Theta) = \mathrm{supp}\left(\widehat{\Theta}^{Enet}_{\alpha_1,\alpha_2}\right)}}{\arg\min} \frac{1}{2n} \left\| y - \beta_0 \mathbb{1}_n - X\beta - Z\Theta \right\|_2^2 \quad . \tag{1.17}$$

This approach is quite simple, but suffers from several drawbacks. The first comes from the Least Squares, if the active features identified by Elastic Net are highly correlated, it may fail, as it is supposed to be used on full column rank design matrix. The second reason is that the pipeline to perform such a method is complex, since it must be done not only on the final hyperparameter obtained by Elastic Net, but on the whole grid of hyperparameters tested by Elastic Net, and for each fold of cross-validation procedure. For these reasons, it is preferable to compute a debiased version of the coefficients along with Elastic Net with Interactions, an approach we will detail in Chapter 2.

In all cases, with or without debiasing step, the main challenge to estimate Elastic Net with Interactions is to develop an efficient algorithm to solve the associated minimization problem and to ensure that the optimization process is achieved. In both cases, the challenge is to avoid storing the whole interaction matrix. We adapt a cyclic coordinate gradient descent [Fu, 1998, Tseng, 2001, Friedman et al., 2007, 2010] algorithm, building *on-the-fly* the interactions features, as classical approaches in interactions settings [Lim and Hastie, 2015, Hazimeh and Mazumder, 2020].

1.5 Optimization framework to solve Elastic Net

Here we present an algorithm to solve the Elastic Net problem without interactions.

1.5.1 Proximal coordinate gradient descent algorithm

For nearly two decades, the coordinate gradient descent algorithm has been widely used to solve convex machine learning problems [Tseng, 2001, Friedman et al., 2007]. The basic principle of the coordinate (gradient) descent algorithm is to transform an optimization problem of size p into p 1-dimensional optimization problems. So, instead of solving a problem of p variables, $p-1$ features are frozen and a 1-dimension problem is solved iteratively, as illustrated in Algorithm 1. There $\gamma_1, \ldots, \gamma_p$ are coordinate-wise step-size parameters, usually chosen based on directional Lipschitz constant computations.

Algorithm 1: Proximal Coordinate Gradient Descent for: $\min f(\beta) + \alpha g(\beta)$, with f a convex differentiable function, g a separable function and $\alpha \in \mathbb{R}^+$.

Initialization: $t = 0$, $\boldsymbol{\beta}^{(0)} = 0_p \in \mathbb{R}^p$ and $\gamma \in \mathbb{R}^p$

while *stopping criterion is not respected* **do**

$$\beta_1^{(t+1)} \leftarrow \text{prox}_{\frac{\alpha}{\gamma_1}}\left(\beta_1^{(t)} - \frac{1}{\gamma_1}\frac{\partial f\left(\beta_1 \quad, \beta_2^{(t)} \quad, \beta_3^{(t)} \quad, \cdots, \beta_p^{(t)}\right)}{\partial \beta_1}\right)$$

$$\beta_2^{(t+1)} \leftarrow \text{prox}_{\frac{\alpha}{\gamma_2}}\left(\beta_2^{(t)} - \frac{1}{\gamma_2}\frac{\partial f\left(\beta_1^{(t+1)}, \beta_2 \quad, \beta_3^{(t)} \quad, \cdots, \beta_p^{(t)}\right)}{\partial \beta_2}\right)$$

$\vdots$

$$\beta_p^{(t+1)} \leftarrow \text{prox}_{\frac{\alpha}{\gamma_p}}\left(\beta_p^{(t)} - \frac{1}{\gamma_p}\frac{\partial f\left(\beta_1^{(t+1)}, \beta_2^{(t+1)}, \beta_3^{(t+1)}, \cdots, \beta_p\right)}{\partial \beta_p}\right)$$

Output : $\beta^{(t+1)}$

In particular, proximal cyclic coordinate descent algorithm has become a state-of-the-art algorithm to solve the LASSO [Friedman et al., 2007, 2010]. We detail in Chapter 2 how to get the closed-form expression allowing to solve the one-dimensional problem of Elastic Net using coordinate descent. Here, the focus is to detail the weaknesses of this algorithm applied to the LASSO, and to present some classical solutions that have been applied in the interaction literature.

Although coordinate descent is easy to use, applied to LASSO type problem, it suffers from a major drawback: it optimizes features that will be null at the end of the optimization problem, as illustrated in Figure 1.5. Indeed, LASSO type problem is expected to bring a sparse estimate. Hence, coordinate descent by optimizing all features will unnecessarily update these coefficients, which slows down the algorithm and becomes even more critical in a quadratic setting. Another challenge is to stop the coordinate

descent to avoid unnecessary updates. In the following section, we detail the *duality gap*, one of the standard criteria used for LASSO type problems.

1.5.2 Stopping criterion

Since ℓ_1 penalized methods are not differentiable, multiple stopping criteria have been used in the literature, such as the duality gap [Kim et al., 2007]. Such a criterion relies on the fact that, associated to the minimization problem $\mathcal{P}(\beta)$ of LASSO type estimators, a maximization problem $\mathcal{D}(\nu)$ in ν variable is of interest. We detail the maximization problem and how to compute the associated dual variable $\widehat{\nu}$ of Elastic Net with Interactions in Chapter 3. These maximization and minimization problems are linked by the fact that the difference between their optimal values vanishes, when strong duality holds, which is the case for LASSO. Hence, denoting $p^{\odot}$ and $d^{\odot}$ the optimal values of the minimization and maximization problems in one hand, and $p^{(t)}$ and $d^{(t)}$ the values of the minimization and maximization after the t^{th} iteration, we define the *duality gap* $\mathcal{G}(\beta, \nu)$ as follows:

$$p^{(t)} - p^{\odot} \leqslant p^{(t)} - d^{(t)} = \mathcal{P}(\beta^{(t)}) - \mathcal{D}(\nu^{(t)}) = \mathcal{G}\left(\beta^{(t)}, \nu^{(t)}\right) \quad . \tag{1.18}$$

Since computing the dual gap costs as much time as one pass of coordinate descent on all features, in standard LASSO implementations, the duality gap is evaluated not after each pass, but every five or ten passes of coordinate descent. While this criterion ensures stopping at a certain tolerance, it is also closely linked to the *screening* and *active set* or *working set* strategies, state-of-the-art approaches to reduce the computational burden of coordinate descent for LASSO and Elastic Net. These methods attempt to identify the coefficients which will be zeroed, so that no time is wasted optimizing them. We first detail screening and then *active set* strategies in the following.

1.5.3 Screening rules

Screening strategies aim at discarding as many coefficients as possible that are guaranteed to be null, to reduce the number of visited features. The core idea comes from Karush-Khun-Tucker (KKT) conditions of the LASSO problem, see for example Massias et al. [2018], which state that:

$$\forall j \in [\![1, p]\!], \quad \left|x_j^{\top} \widehat{\nu}\right| < 1 \Rightarrow \widehat{\beta}_j \neq 0 \quad . \tag{1.19}$$

Hence, we obtain the first screening rule: the correlation between the optimal dual variable and the j^{th} feature. Unfortunately, this criterion is impossible to use, because

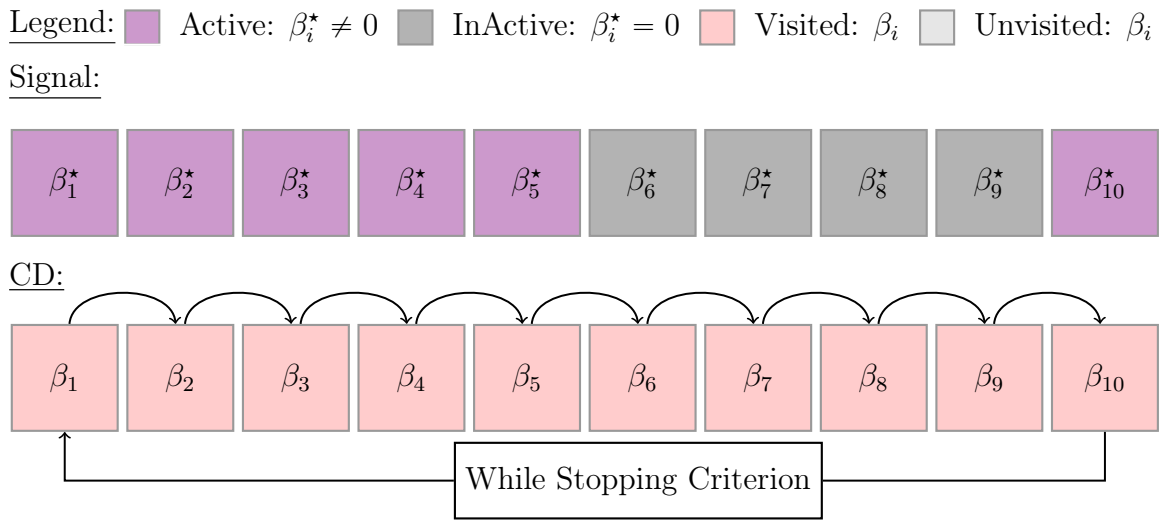


Figure 1.5: Representation of the functioning of a coordinate descent algorithm to solve a LASSO type problem, where features 1 to 5, and 10 have non-zeros coefficients at solution and where coefficients of features 6 to 9 are expected to be null.

it requires the knowledge of the exact dual variable. A solution to address this problem is to construct a set $\mathcal{C}$, called *Safe Region*, such that $\widehat{\nu}$ is contained in $\mathcal{C}$. This allows to establish the *safe screening rules* [El Ghaoui et al., 2012]:

$$\text{If } \widehat{\nu} \in \mathcal{C}, \text{ then: } \max_{\nu \in \mathcal{C}} \left| x_j^\top \nu \right| < 1 \Longrightarrow \left| x_j^\top \widehat{\nu} \right| < 1 \Longrightarrow \widehat{\beta}_j = 0 \ . \tag{1.20}$$

Hence, the literature focuses on the efficient construction of smaller $\mathcal{C}$ regions that always contain $\widehat{\nu}$, since, bigger the region is, fewer screened variables are. In particular, thanks to their simplicity, sphere-based regions have been well studied (see for examples El Ghaoui et al. [2012], Fercoq et al. [2015], Ndiaye et al. [2017]), to improve the center and the radius of the sphere. Also, one thing that helps to build a more efficient region is to take into account the solution of the previous hyperparameter LASSO problem [El Ghaoui et al., 2012]. Still with the idea of benefiting from the optimization process, the dynamic safe sphere rule [Bonnefoy et al., 2014, 2015, Fercoq et al., 2015, Ndiaye et al., 2017] aims to benefit from the coordinate descent algorithm to refine the set of discarded features. In particular, last works [Fercoq et al., 2015, Ndiaye et al., 2017] have developed the *Gap Safe Rule*, which uses duality gap, to increase the set of discarded features along the optimization steps:

$$\left| x_j^\top \nu \right| + \left(\sqrt{\frac{2\mathcal{G}\left(\beta, \nu\right)}{n\alpha^2}} \right) \left\| x_j \right\|_2 < 1 \Longrightarrow \widehat{\beta}_j = 0 \ . \tag{1.21}$$

Hence, even if only a few features are discarded at the beginning of the process, their number increases as the optimization progresses, as illustrated in the Figure 1.6.

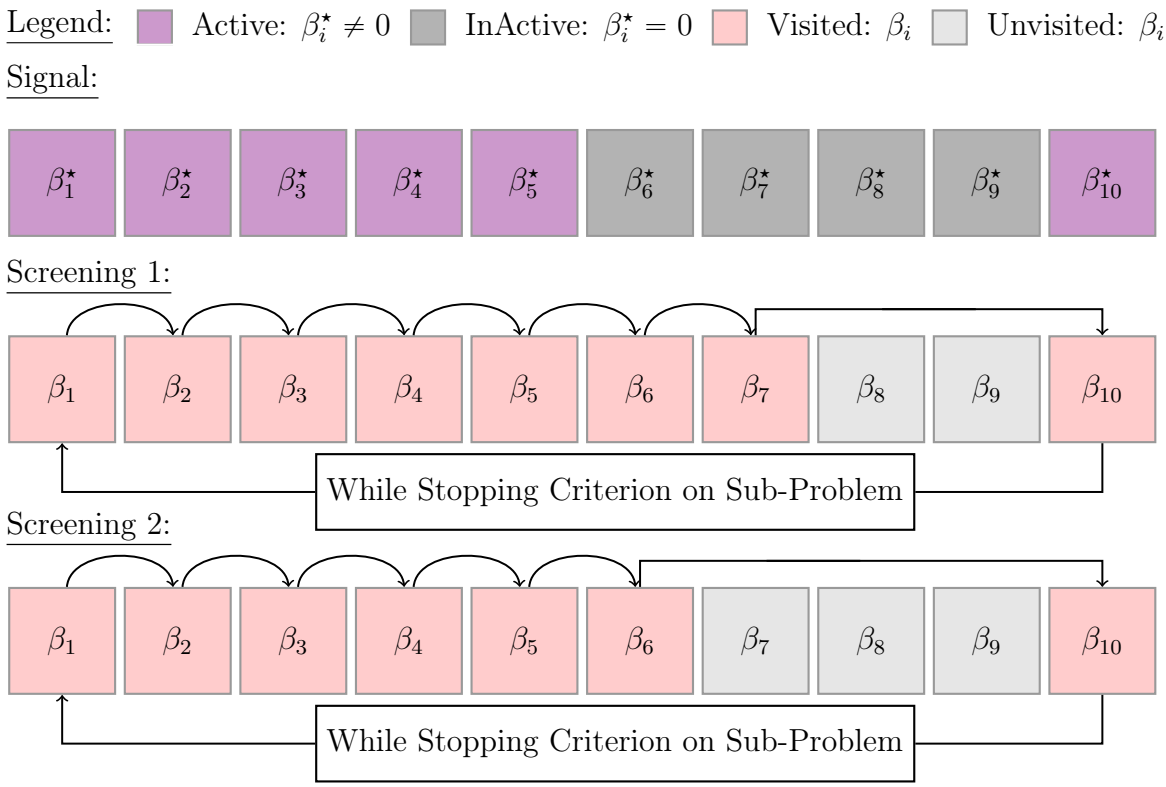


Figure 1.6: Representation of the functioning of a coordinate descent algorithm with screening strategy.

1.5.4 Active set

While safe screening rules aim at discarding features guaranteed to be zero at the optimum and then reducing the number of visited coefficients, the *working set* or *active set* algorithm, abbreviated WS and AS respectively, (see [Fan et al., 2008, Kim and Park, 2010, Boisbunon et al., 2014, Johnson and Guestrin, 2015, Massias et al., 2017, 2018, Bertrand et al., 2022]) aims at selecting a subset of a few features $\mathcal{W}^{(t)}$ and solving the associated subproblem then testing if the global problem is solved by computing the duality gap on the whole dataset. If the global problem is satisfied, *i.e.,* the subset of selected features contains all the expected actives features, the algorithm stops. Otherwise, it means that the first one does not contain one or more active features and therefore it must start over on an increased set, as depicted in Figure 1.7.

Although this procedure seems simple, it raises two main questions: what are the criteria to classify and selecting the features, and how to increase the feature set. In the following, we detail CELER [Massias et al., 2018], a state-of-the-art working set algorithm to solve LASSO type problems.

Figure 1.7: Representation of the functioning of a coordinate descent algorithm with active or working set strategies.

CELER [Massias et al., 2018]. From the safe screening rules [Fercoq et al., 2015, Ndiaye et al., 2017], CELER derives the following criterion d_j (1.22) to rank each possible feature:

$$d_j(\nu) = \frac{1 - |x_j^\top \nu|}{\|x_j\|_2} > \sqrt{\frac{2\mathcal{G}(\beta, \nu)}{n\alpha}} \Rightarrow \widehat{\beta}_j = 0 \ . \tag{1.22}$$

This criterion allows us to prioritize the features to be added, from the smallest d_j to the largest, with the smallest corresponding to the features expected to be the most relevant.

Regarding the way to increase the size of the set, CELER suggests doubling its size. A more aggressive option, called pruning, allows taking twice many features than the number of actives features in the current set. Finally, another key point of CELER it to propose to use an improved dual point, to improve the criterion which ranks features, but mostly to stop the optimization earlier.

We summarize the CELER algorithm in Figure 1.8.

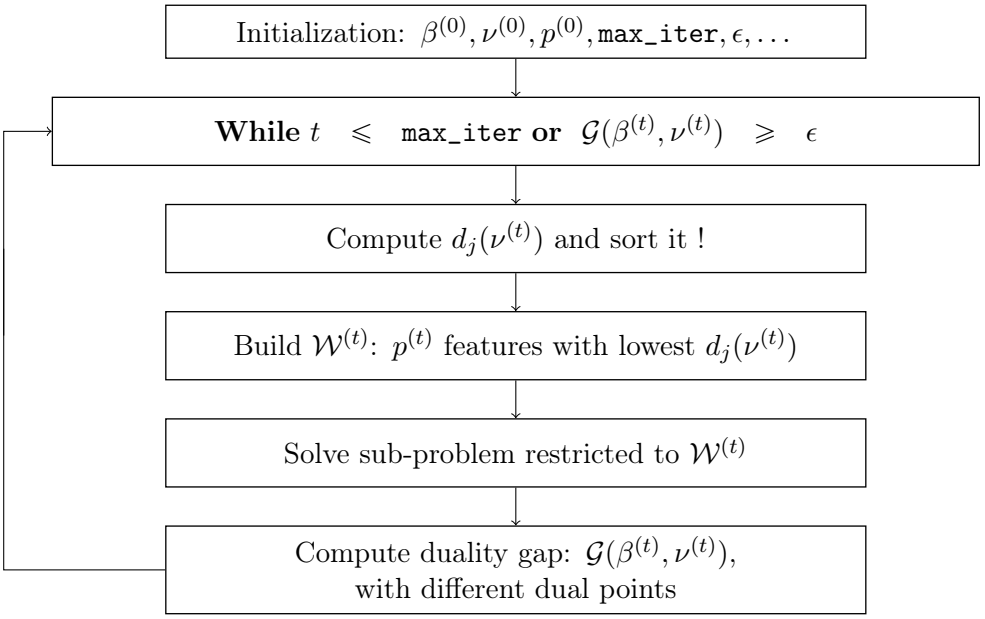


Figure 1.8: CELER algorithm in a schematic way: as long as the duality gap on the whole problem does not reach a tolerance of ϵ, CELER continues. We observe that between each sub-problem, it computes the duality gap and updates the ranking rules.

1.5.5 Screening and active set in interactions literature

Since the quadratic regression problem has even more coefficients, these strategies have been widely applied for the selection of interaction effects. For example, the GLINTERNET [Lim and Hastie, 2015] method relies on strong rules [Tibshirani et al., 2012] to discard main effects and then add interactions respecting strong heredity. However, screening allow us to address the problem of quadratic regression without heredity assumptions. For example, the tree pruning criterion of Nakagawa et al. [2015, 2016] is based on screening rules, while *working set* strategies are used by HierScale in Hazimeh and Mazumder [2020] and in WHInter [Le Morvan and Vert, 2018]. In particular, WHInter exploits the tree structure of interactions to determine which interactions should constitute the working set, without having to sort them all. To this end, WHInter's criterion allows, from a criterion based on a main effect, to indicate if the associated branch contains active interactions that are not yet present in the current working set. Finally, in the case where the criterion would not ensure that all the active interactions in a branch are already present in the current set, WHInter adapts a variant of the *maximum inner product search* problem to quickly sort all the quadratic variables in a branch.

In our turn, we aim to develop an *active set* algorithm to estimate Elastic Net with Interactions, relying on CELER solver, since we have no data or structural hypothesis.

1.6 Gene expression regulation mechanism

While we first aim to develop an estimator to tackle interaction in a wide range of applications, we will use a genomic dataset from Bessière et al. [2018] to illustrate its interest and analyse the performance of our approach.

1.6.1 Gene expression regulation

DNA structure. The genetic information enables the development, functioning, growth and reproduction of an organism. All this information is stored in the genome, a (macro)-molecule called deoxyribonucleic acid, most classically abbreviated DNA.

The DNA was initially discovered by Friedrich Miescher in 1869 but its double helix structure was discovered in 1953 by Watson, Crick, Wilkins, and Franklin. The DNA sequence is constituted by only four base elements called nucleotides: Adenine (A), Cytosine (C), Guanine (G) and Thymine (T). The DNA is a double-stranded chain, where both strands are complementary. The nucleotides form pairs two by two: Adenine with Thymine, and Cytosine with Guanine.

Transcription process. Some subsequences of the DNA sequence (in particular genes coding for proteins, but not only) can be *transcribed*, that is copied into a single-stranded molecule called Ribonucleic Acid, traditionally abbreviated RNA.

A sequence of nucleotides in the DNA sequence that encodes an RNA molecule is called a gene. Some genes, called coding-genes, produce a certain type of RNA, Messenger RNA, shortened mRNA, which is in a second step translated into a protein. Coding genes are the ones that interest us in the following.

A gene is said to be *expressed* when transcriptions of this gene are operated. Genes can be transcribed from several Transcription Start Site, abbreviated TSS. Gene sequences can be divided into 2 types of subsequences: exons and introns. Introns, which represent half of the genome, are the part of coding genes which are deleted from mRNA but play a role in transcription control process. Only the exons are present in the maturated mRNA. Moreover, gene sequences start and finish with Untranslated Transcribed Regions (UTR). The starting gene sequence, corresponding to the opening of the gene, is called 5UTR while the ending gene sequence is called 3UTR.

We take interest in gene regulation which is the whole mechanism orchestrating transition from DNA to RNA or protein.

1.6.2 Dataset Description

We take over the linear model proposed in Bessière et al. [2018], where the aim is to explain the amount of `mRNA` produced for each gene, according to `DNA` statistical summary associated to this gene. To perform both estimation and features selection, they propose to use the LASSO estimator, from centered response and standardized predictors.

Measurements details. We attempt to express the `mRNA` copy number of $n = 16\,294$ human genes from different cancerous cells, in function of `DNA` resume. RNA-Seq data, measuring the copy number associated with each `mRNA`, was extracted from the TCGA Research Network database (https://www.cancer.gov/tcga) for 12 different cancer types, with around 20 patients for each type, leading to a matrix of responses $Y \in \mathbb{R}^{n \times m}$, $m = 241$. To fit a linear model, the counts were log-transformed. The `DNA` resume X is constituted of `DNA` 1-words, *i.e.,* nucleotide frequencies (letters A, C, G and T) and 2-words frequencies, *i.e.,* frequencies of dinucleotide (couples of letters: AA, ..., TT), for 8 regions associated to each gene.

Region descriptions. The frequencies have been measured in 5UTR, 3UTR and in the introns (abbreviated Intr), but equally in five others regions. Three of them were centered on the first TSS to take into account the information contained in the beginning of the gene. The first is Core region, which starts arbitrarily 500 nucleotides before the TSS and ends 500 after. We have also two more distant regions: Distal Upstream and Distal Downstream, respectively abbreviated DU and DD. These three regions are illustrated in Figure 1.9.

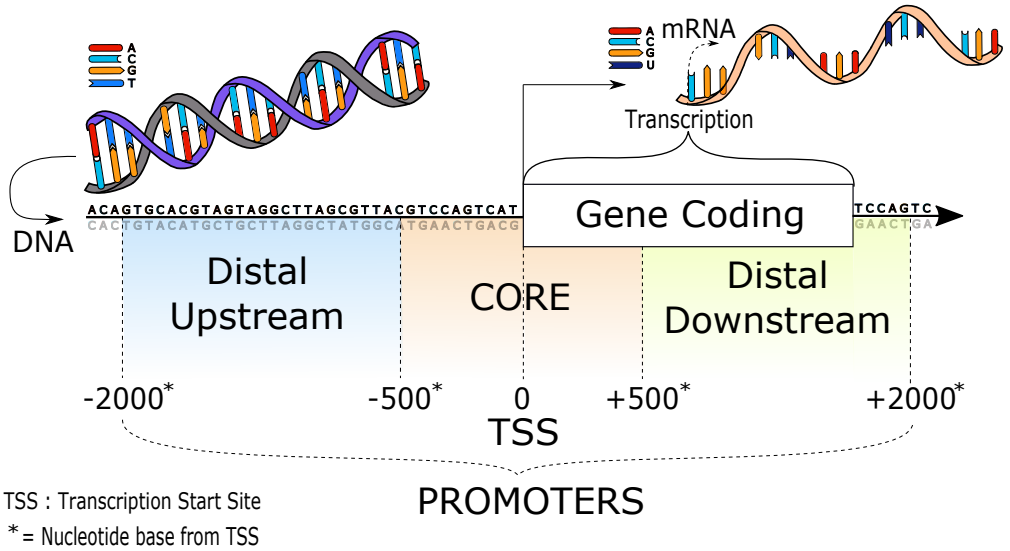


Figure 1.9: Gene-Regulation problem illustration, with three regions: Core, DU, DD.

The DU region starts 2000 nucleotides before the TSS and ends where the Core begins. The DD region starts with the end of Core region and finishes 2000 nucleotides before the TSS. Then, we have the CDS regions, for Coding `DNA` Sequence, which represent the set of the nucleotide which can be present in `mRNA`.Finally, the last region is DFR, for Downstream Flanking Regions, which corresponds to the region which spans on 1000 nucleotides right after the end of the genes.

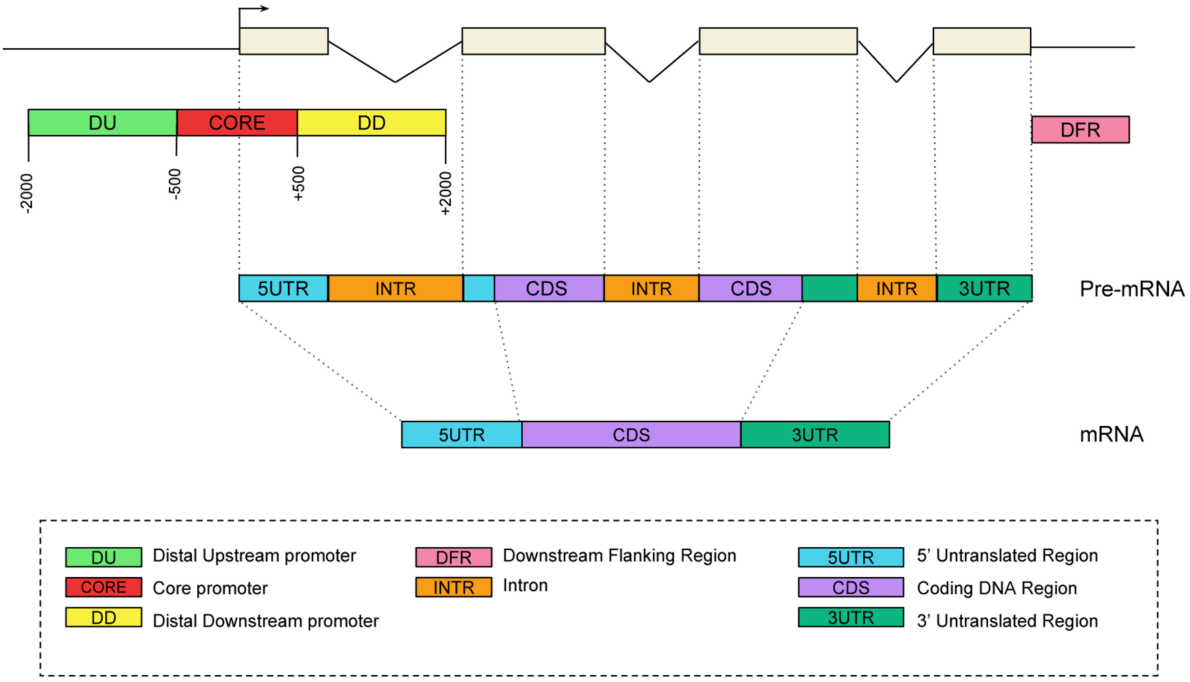


Figure 1.10: Gene-Regulation problem illustration, with the all eight regions from [Bessière et al., 2018] article.

1.6.3 Statistical Challenges

Dataset dimension. Hence, with 4 nucleotides (`A, C, G, T`) and 16 dinucleotides (`AA, AC, AG, AT, CA, ..., TT`) rates, we have 20 features of interest for each of the 8 regions, so a total number of 160 covariates. Adding the interactions, this leads to having 12 720 or 12 880 features, depending on whether we include pure quadratic terms or not, for each of the $n = 16\ 294$ human genes. Furthermore, the interaction design matrix requires 1.67 Gby to be stored in memory.

Interactions. As mentioned in the beginning, element wise product is mainly considered. However, product is not always interpretable and other non-additive operations can be useful to bring new insight in comprehension. Indeed, element wise maximum can be interpreted as a logical `AND` since, a low maximum corresponds to two both low frequencies, while high maximum indicates that at least one of the features has a high frequency, which can bring more interpretability, than product.

First Standardization Scheme (STD 1)

Get X ⟶ Build Z from X ⟶ Standardize X and Z

Second Standardization Scheme (STD 2)

Get X ⟶ Standardize X ⟶ Build Z from standardized X ⟶ Standardize Z

Figure 1.11: Representation of the standardization scheme.

Standardization scheme. While, for only main effects problems, there exists a standard standardization, interaction problem brings two possibilities. The first is to build, on the fly as well as to store it, the interactions matrix Z from X, and then standardize X and Z altogether, as illustrated in the upper part of Figure 1.11 (STD 1). However, another standardization scheme is possible, which starts by standardizing the main design matrix X, then builds Z from this standardized X matrix, and to finish, standardizes Z as depicted in the lower part of Figure 1.11 (STD 2). While these two standardization processes are not proper to this dataset, we will see that they affect both correlation and condition number, the latter representing how much the optimization problem is hard to solve.

While, the dimension indicates that, even with interactions, there are more samples ($n = 16\,294$) than features ($p+q = 160+12\,880 = 13\,040$), the main challenge comes from correlation between main features but also between interactions, what makes identifying relevant features difficult as well as solving the optimization problem.

Correlation. This dataset has highly correlated features which makes variable selection difficult, and high conditioning number, which affect the estimator convergence. One reason of this high level of correlation comes from the construction of the dataset, since the measures record the frequencies of each nucleotide, frequencies which sum to one. Hence, one of the nucleotide rates can be obtained by subtracting the three others from one. The same thing appears with the 16 di-nucleotides rates.

We represent in Figure 1.12 the correlation of nucleotide and di-nucleotides between themselves for each region. While the first row of the figure presents only main effects, which are already highly correlated, the others row present product in row two and three, while maximum is illustrated in the rows four and five. The first observation is that clustermaps of maximum indicates higher correlation than for product. We observe that the second standardization scheme helps to reduce the correlation level for both element-wise product and maximum interactions.

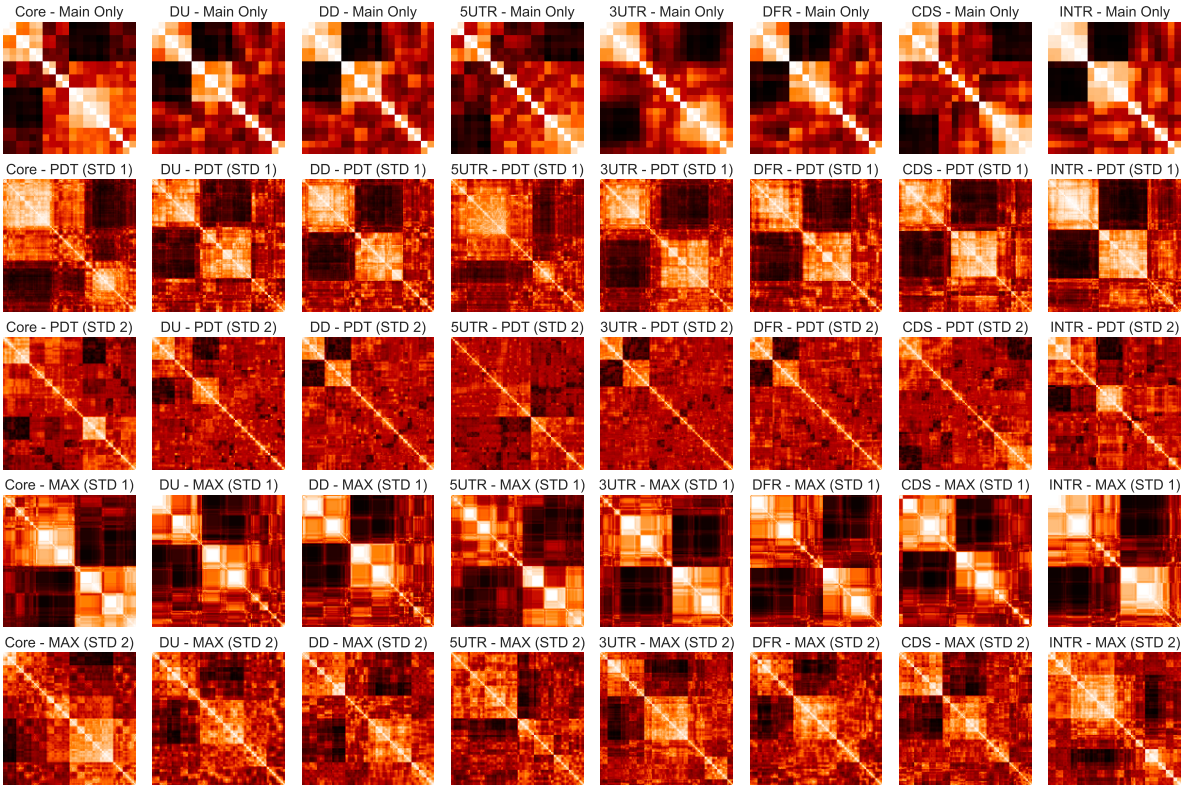


Figure 1.12: Correlations matrices for each region, without interactions for the first line and then for product and maximum. First, we observe that considering the interactions increases the level of correlation between the features, however we observe that the second standardization scheme decreases this level of correlation. Lastly, we observe that maximum increases more the level of correlation than the product.

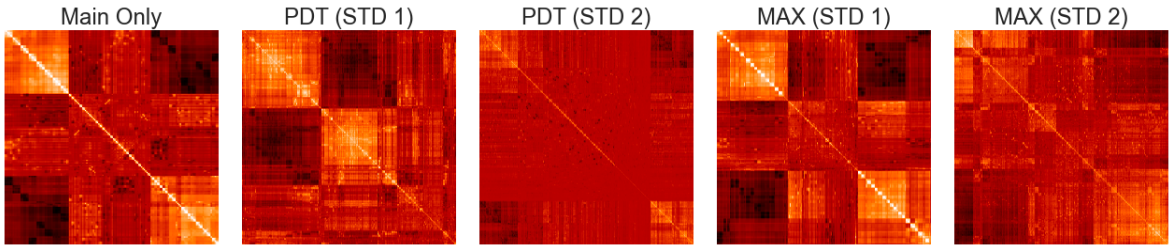


Figure 1.13: Correlation matrices of the complete data set, without interactions on the left, then for product and maximum in function of standardization scheme. As for correlations matrices of each region, we observe that second standardization manages correlation level while maximum tends to increase it.

This effect is also observable on the whole dataset. Again, we observe on the correlation clustermaps between $W = [X, Z]$ that the second standardization scheme helps to reduce correlation level. Hence, this second standardization scenario will be statistically helpful.

Conditioning Number. In this last part, we discuss the matrix conditioning number $\rho(X)$ which is defined as the ratio between the highest and lowest singular values, *i.e.*, $\rho(X) = \frac{\sigma_{\max}(X)}{\sigma_{\min}(X)}$. This number indicates how difficult the numerical problem is to solve. We observe an identical behavior in Figure 1.14 as in Figures 1.12 and 1.13, hence considering interactions, in particular maximum increases the difficulty to solve the associated problem. In addition, the second standardization scheme also reduces the difficulty of the optimization problem.

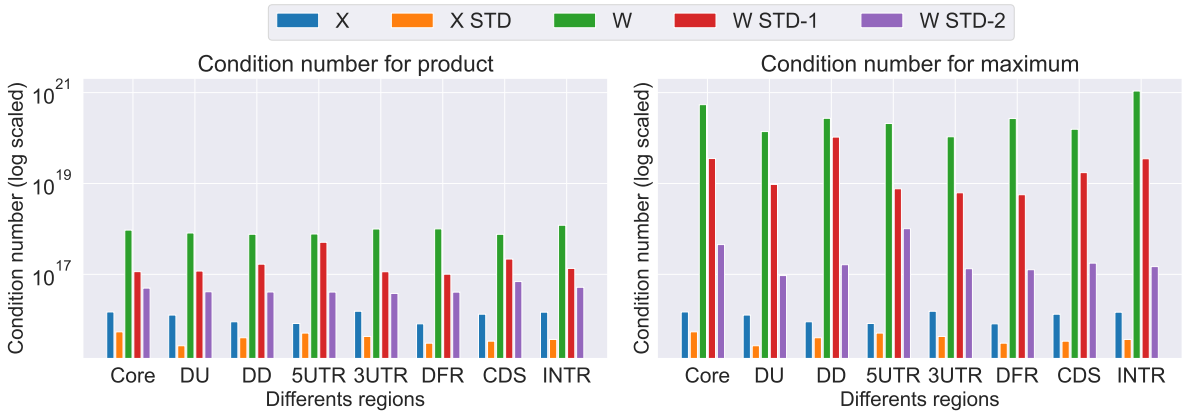


Figure 1.14: Conditioning number for each region, with element-wise product interactions on the left-hand side and element-wise maximum interactions on the right-hand side. As for the level of correlation, we observe that maximum increases the level of correlation, whereas the second standardization scheme (STD-2) succeeds in reducing the conditioning number.

1.7 Thesis organization

The thesis is organized as follows.

In Chapter 2, we first explain how to tackle the interactions problem in section 2.1, by suggesting first to reparameterize Elastic Net problem in section 2.1.1. We then provide an algorithm which allows to compute *on-the-fly* the interactions columns in section 2.1.2, avoiding to store the interactions matrix. We then discuss first results on toy example in section 2.1.3. In a second part, we detail the solution applied to take into account the bias of the Elastic Net penalty in section 2.2. We detail the debiasing method in section 2.2.1 while in section 2.2.2 we provide a differentiation algorithm necessary for its implementation. However, this first algorithm being intractable for computational reasons, we provide a tractable version in section 2.2.3 as well as the first statistical results of our approach with the debiasing step in section 2.2.4.

Then, in Chapter 3, we start by deriving the dual problem associated to our approach in section 3.1, and in particular, we illustrate its computational burden. Afterwards, we

develop in section 3.2 an active set algorithm whose main idea is to avoid as much as possible to explore the set of quadratics interactions. In this purpose, we develop an algorithm which approximates CELER rules in one hand, and develop heuristic stopping criterion, to avoid computing the duality gap. Lastly, in section 3.3 we adapt Anderson Acceleration to accelerate again our algorithm whereas the last part of the chapter illustrates numerical performances in section 3.4.2.

Finally, in Chapter 4, we perform comparisons with RAMP and HierNet estimators. The first part of the chapter provides comparisons on the semi-artificial datasets, allowing to discuss predictive performances, as well as selection ability scores and computational performances, on three different simulations studies. Then, the last part section 4.2 provides statistical results on real datasets, and show that our approach improves predictive performances whereas our debiasing step allows us to reduce number of active features and thus promotes interpretability.

Chapter 2

A debiased Elastic Net with Interactions

Contents

In this chapter, we describe the technical and statistical reasons, motivated by our genomics data, that led us to estimate a linear model with interactions using an adapted version of the Elastic Net estimator.

The first reason is that, to estimate the coefficients of a linear model for high dimensional data while preserving interpretability, ℓ_1 estimators are commonly used for their sparsity, especially in interaction setting.

Nevertheless, we have seen in section 1.6.3 that the genomics datasets that challenge us, have highly correlated features. Then, it is well-known that correlated features lead to decrease the selection performances of the LASSO estimator, which tends to select only one feature from a group of several relevant correlated features. Therefore, adding

a ℓ_2 penalty allows us to handle these correlations and ill-conditioning, which helps to solve the associated optimization problem.

For these reasons, we chose to adapt the Elastic Net [Zou and Hastie, 2005] estimator to fit linear models with interactions. The Elastic Net (2.1) has been introduced as a regularization and feature selection estimator, which benefits from sparsity thanks to the ℓ_1 penalty and can select groups of correlated features, with ℓ_2 penalty.

$$\widehat{\beta} \in \arg\min_{\beta \in \mathbb{R}^p} \frac{\|y - X\beta\|^2}{2n} + \alpha_1 \|\beta\|_1 + \frac{\alpha_2}{2} \|\beta\|_2^2 \quad . \tag{2.1}$$

However, it is well-know that penalized estimators lead to shrink the coefficients towards zero, implying a loss of amplitude for the coefficients from the set of active features. In practice, this amplitude loss leads to include false positive features, which affects the selection ability of the method. In order to recover the amplitude loss, we adapt the CLEAR framework to the Elastic Net case, to debias the Elastic Net estimate along its estimation.

Nonetheless, other estimators than Elastic Net, which do not suffer from penalization bias, would have been interesting to study and do not require a debiasing step. Among the possible estimators, the best subset selection estimator [Beale et al., 1967, Hocking and Leslie, 1967] has recently seen a surge of interest. This estimator is based on the ℓ_0 norm, as written in its penalized form in Equation (2.2), leading to a non-convex problem that is NP-hard to solve.

$$\widehat{\beta} \in \arg\min_{\beta \in \mathbb{R}^p} \frac{\|y - X\beta\|^2}{2n} + \alpha \|\beta\|_0 \quad . \tag{2.2}$$

However, recent progress [Bertsimas et al., 2016] has been done to speed up the optimization of the best subset selection problem, using mixed integer optimization (MIO). While previously, using the best subset estimator was commonly limited to $p = 10$ main features, the new solver allows tackling problem with $p = 1000$ main features. Unfortunately, taking interactions into account induces a total number of features that quickly exceeds 1000 features. Moreover, some extensive studies [Hastie et al., 2020] have shown that, in simulated setting with a low signal-to-noise ratio (SNR), LASSO outperforms the best subset selection estimator while having a faster solver (around 0.01 s for LASSO against more than 1 hour).

Lastly, other estimators, for example Fan and Li [2001], Zhang [2010], have been developed to tackle the linear regression problem while removing nearly all penalty bias. Unfortunately, such estimators provide a non-convex optimization problem and may suffer from local minima, which causes difficulties to stop the algorithm.

This chapter is organized as follows. In a first part, section 2.1 specifies how we tackle

the interactions problem. Section 2.1.1 describes how we parametrize Elastic Net with Interactions, while section 2.1.2 details how to solve the Elastic Net with Interactions problem, thanks to proximal coordinate descent algorithm. Section 2.1.3 presents the first statistical results on a toy example. In a second part, section 2.2 describes how we propose to tackle Elastic Net bias problem, with CLEAR estimator [Deledalle et al., 2017]. Section 2.2.1 details the CLEAR framework, while section 2.2.2 describes how to adapt it to interactions cases, by adapting an automatic differentiation scheme to address the main challenge: computing the Jacobian of our estimator. Then, section 2.2.3 details how we handle the numerical challenges of this automatic differentiation scheme to reduce both its storage and computational burden. Finally, section 2.2.4 presents the first statistical results on a toy example.

2.1 Elastic Net for linear models with interactions

A direct adaptation of Elastic Net to the second order interactions cases leads to the following estimator Elastic Net with Interactions (2.3), which was initiated in our works [Bascou et al., 2020, 2021].

$$(\widehat{\beta}, \widehat{\Theta}) \in \underset{\substack{\beta \in \mathbb{R}^p \\ \Theta \in \mathbb{R}^q}}{\arg\min} \frac{\|y - X\beta - Z\Theta\|^2}{2n} + \mathrm{pen}(\beta, \Theta; \alpha = (\alpha_{1,1}, \alpha_{1,2}, \alpha_{2,1}, \alpha_{2,2})) \ , \tag{2.3}$$

$$\text{with } \ \mathrm{pen}(\beta, \Theta; \alpha) = \alpha_{1,1} \|\beta\|_1 + \alpha_{1,2} \|\Theta\|_1 + \frac{\alpha_{2,1}}{2} \|\beta\|_2^2 + \frac{\alpha_{2,2}}{2} \|\Theta\|_2^2 \ ,$$

This equation requires the estimation of four hyperparameters, implying a search in $\mathbb{R}^4$. The computational cost of estimating these parameters is prohibitive, so we detail in the following the two approaches explored in this work to reduce the number of hyperparameters.

2.1.1 Elastic Net parametrization

A first approach to reduce the computational burden is to set $\alpha_1 = \alpha_{1,1} = \alpha_{1,2}$ and $\alpha_2 = \alpha_{2,1} = \alpha_{2,2}$, which reduces the hyperparameter search to $\mathbb{R}^2$. Moreover, to further reduce the search space, we introduce, as in GLMNET [Friedman et al., 2010] and scikit-learn [Pedregosa et al., 2011], an additional parameter: $\gamma \in \ ]0, 1]$ which controls the trade-off between the penalty ℓ_1 and ℓ_2, with $\gamma = 1$ corresponding to LASSO. Finally, this supplementary parameter is not tested on hundreds of values, but only on a few, such as $\gamma \in \{1, 0.95, 0.9, 0.5\}$, for example. Hence, the following penalty (2.4) reduces the

hyperparameter search to $\mathbb{R}$ times the number of discrete values tested for γ and κ.

$$\text{pen}\left(\beta, \Theta; \alpha\right) = \alpha\gamma\left(\|\beta\|_1 + \|\Theta\|_1\right) + \frac{1}{2}\alpha(1-\gamma)\left(\|\beta\|_2^2 + \|\Theta\|_2^2\right) \quad . \tag{2.4}$$

While this first approach helps to reduce hyperparameters search, it leads to consider main effects and interactions effects at the same level. This can lead to having more active interactions than main effects, which may be painstaking for interpretability.

Hence, we propose adding a supplementary hyperparameter to penalize more the interactions than the main effects. This approach is already considered in [Hao et al., 2018, Hazimeh and Mazumder, 2020]. We denote this parameter by κ, set up by default to $\kappa = 5$, whose importance is discussed in sections 2.1.3.2 and 2.2.4.1. Once γ and κ are fixed, we need a non-increasing grid, as for the classical Elastic Net problem. Usually, such a grid is initialized to $\alpha^0 = \frac{\|y^\top X\|_\infty}{n\times\gamma}$, sometimes referred as $\alpha_{\max}$, such that the solution of Elastic Net is the null vector, *i.e.*, $\widehat{\beta} = 0_p$. So, we propose to build such a grid for main Equation (2.5) and one for interactions Equation (2.6), as follows:

$$\frac{\left\|X^\top y\right\|_\infty}{n \times \gamma} = \alpha_\beta^0 > \alpha_\beta^1 > \cdots > \alpha_\beta^K = \epsilon \times \alpha_\beta^0 \quad , \tag{2.5}$$

$$\frac{\left\|Z^\top y\right\|_\infty}{n \times \gamma} = \alpha_\Theta^0 > \alpha_\Theta^1 > \cdots > \alpha_\Theta^K = \epsilon \times \alpha_\Theta^0 \quad . \tag{2.6}$$

The main interest of this double grid is that it avoids favoring interaction over main effects, in the case where $\|Z^\top y\|_\infty$ is much larger than $\|X^\top y\|_\infty$, while the search space is still $\mathbb{R}$. So on, we parametrize our problem as follows and denoted $\mathcal{P}(\beta, \Theta)$:

$$(\widehat{\beta}, \widehat{\Theta}) \in \underset{\substack{\beta\in\mathbb{R}^p \\ \Theta\in\mathbb{R}^q}}{\arg\min} \frac{\|y - X\beta - Z\Theta\|^2}{2n} + \text{pen}(\beta, \Theta; \alpha = (\alpha_\beta, \alpha_\Theta, \gamma, \kappa)) \quad , \tag{2.7}$$

$$\text{with} \quad \text{pen}(\beta, \Theta; \alpha) = \alpha_\beta\gamma\|\beta\|_1 + \alpha_\Theta\kappa\gamma\|\Theta\|_1 + \frac{\alpha_\beta(1-\gamma)\|\beta\|_2^2 + \alpha_\Theta\kappa(1-\gamma)\|\Theta\|_2^2}{2} \quad ,$$

where $\gamma \in \ ]0, 1]$ control the trade-off between ℓ_1 and ℓ_2 penalties, and κ allows penalizing more the interactions than the main effects.

Finally, we propose to adjust those hyperparameters using cross-validation (with "five folds" as default). For ease of presentation, we will also use the following notation:

$$\alpha_{1,1} = \alpha_\beta\gamma, \ \alpha_{1,2} = \alpha_\Theta\kappa\gamma, \ \alpha_{2,1} = \alpha_\beta(1-\gamma), \ \alpha_{2,2} = \alpha_\Theta\kappa(1-\gamma) \quad . \tag{2.8}$$

In the following, we explain how to solve the optimization problem (2.7), with coordinate descent.

2.1.2 Coordinate descent for Elastic Net with interactions

To solve this optimization problem, we first adapt a cyclic coordinate gradient descent algorithm [Friedman et al., 2007, 2010], to avoid storing the interaction matrix in memory. Similarly to [Lim and Hastie, 2015, Hazimeh and Mazumder, 2020], we propose to evaluate *on-the-fly* the interaction columns, *i.e.,* we build each interaction column when we update the associated coefficients. Then, a key point is to efficiently find which columns j_1 and j_2 of main design matrix X, are required to build the associated interaction columns. We suggest using a matrix, called *Rosetta* and abbreviated $\mathcal{R}$, to easily and efficiently do it.

Rosetta matrix. This matrix, given an interaction index $jj \in [\![1, q]\!]$, returns the two associated main index $j_1,\ j_2 \in [\![1, p]\!]$. Hence, we get a bijection between an interaction index and the associated main doublet. For example, in the case where we consider interactions with pure quadratic effect, *i.e.,* the interaction columns of an index with itself $x_{j_1} \odot x_{j_1}$, this *Rosetta* $\mathcal{R} \in \mathbb{R}^{3\times\frac{p(p+1)}{2}}$ matrix is defined as:

$$\mathcal{R} = \begin{bmatrix} 1 & \cdots & p & p+1 & \cdots & 2p-1 & 2p & \cdots & 3p-2 & \cdots & q \\ 1 & \cdots & 1 & 2 & \cdots & 2 & 3 & \cdots & 3 & \cdots & p \\ 1 & \cdots & p & 2 & \cdots & p & 3 & \cdots & p & \cdots & p \end{bmatrix} . \tag{2.9}$$

This matrix is used as follows: if $\mathcal{R}_{1,jj}$ is the index of an interaction, we obtain the first main effect associated index j_1 with the coordinate $\mathcal{R}_{2,jj}$ while the second index j_2 is given with $\mathcal{R}_{3,jj}$. The matrix is similar when dropping the quadratic effects, with $\mathcal{R} \in \mathbb{R}^{3\times\frac{p(p-1)}{2}}$, the columns corresponding to pure quadratic effects being erased accordingly. It allows iterating efficiently on interaction matrix giving a subset of actives interactions $\mathcal{A}_\Theta$, and to build a coordinate gradient descent algorithm that may not visit all quadratic features. Notably, *Rosetta* allow us to build an interaction between j_1 and j_2 columns even if j_1 or j_2 are not features which are in $\mathcal{A}_\beta$, the set of active first order features. Finally, a key advantage is that this matrix can easily be computed and then stored at the beginning of the cross-validation procedure and then used throughout the pipeline.

Proximal coordinate descent algorithm. Then, the first step to build a proximal gradient descent to solve our Elastic Net with Interactions estimator is to be able to minimize the associated one-dimensional problem. The following proposition gives the solution of each one-dimensional problem.

Proposition 2.1.1. *We write $\widehat{\beta}^{(t)}$ and $\widehat{\Theta}^{(t)}$ for the coefficients computed at the t-th pass over the data by the cyclic proximal coordinate gradient descent algorithm, and*

$r = y - X\widehat{\beta}^{(t)} - Z\widehat{\Theta}^{(t)}$ is the associated residuals (at the t-th pass). The coordinate update rules for the j_0 main effects and j_0' interaction coordinate read:

$$\widehat{\beta}_{j_0}^{(t+1)} = \frac{1}{\|x_{j_0}\|^2 + n\alpha_{2,1}} \mathrm{ST}\left(x_{j_0}^\top \left(r + \widehat{\beta}_{j_0}^{(t)} x_{j_0}\right), n\alpha_{1,1}\right) \quad , \tag{2.10}$$

$$\widehat{\Theta}_{j_0'}^{(t+1)} = \frac{1}{\|z_{j_0'}\|^2 + n\alpha_{2,2}} \mathrm{ST}\left(z_{j_0'}^\top \left(r + \widehat{\Theta}_{j_0'}^{(t)} z_{j_0'}\right), n\alpha_{1,2}\right) \quad , \tag{2.11}$$

and ST *represents the soft-thresholding operator, defined for any $x \in \mathbb{R}$ by:*

$$\mathrm{ST}(x, \alpha) = \max(0, |x| - \alpha)\,\mathrm{sign}(x) = (|x| - \alpha)_+ \,\mathrm{sign}(x) \quad . \tag{2.12}$$

Applying these rules to coordinate descent, with one main effects `for` loop and one interactions `for` loop, leads to Algorithm 2, which describes how to make one pass on the index set of main $\mathcal{A}_\beta$ and quadratic $\mathcal{A}_\Theta$ features.

Algorithm 2: One epoch of cyclic coordinate descent algorithm

Input : $X,\ y,\ \beta^{(t)},\ \Theta^{(t)},\ \alpha = (\alpha_{1,1},\ \alpha_{1,2},\ \alpha_{2,1},\ \alpha_{2,2}),\ \mathcal{A}_\beta = [\![1, p]\!],\ \mathcal{A}_\Theta = [\![1, q]\!]$

Init : $r = y - X\widehat{\beta}^{(t)} - Z\widehat{\Theta}^{(t)}$

for $j \in \mathcal{A}_\beta$ **do** `// set of updated main features`

  $\beta_j^{(t+1)} = \frac{1}{\|x_j\|^2 + n\alpha_{2,1}} \mathrm{ST}(x_j^\top (r + \beta_j^{(t)} x_j), n\alpha_{1,1})$

  if $\beta_j^{(t+1)} \neq \beta_j^{(t)}$ **then**

    $r \mathrel{+}= \left(\beta_j^{(t)} - \beta_j^{(t+1)}\right) x_j$ `// update residuals`

for $jj \in \mathcal{A}_\Theta$ **do** `// set of updated interactions features`

  $j_1, j_2 = \mathcal{R}_{jj}$ `// Rosetta transforms index of Z in double indices`

  $z_{jj} = x_{j_1} \odot x_{j_2}$

  $\Theta_{jj}^{(t+1)} = \frac{1}{\|z_{jj}\|^2 + n\alpha_{2,2}} \mathrm{ST}(z_{jj}^\top (r + \Theta_{jj}^{(t)} z_{jj}), n\alpha_{1,2})$

  if $\Theta_{jj}^{(t+1)} \neq \Theta_{jj}^{(t)}$ **then**

    $r \mathrel{+}= \left(\Theta_{jj}^{(t)} - \Theta_{jj}^{(t+1)}\right) z_{jj}$ `// update residuals`

Output : $\beta^{(t+1)},\ \Theta^{(t+1)}$

This algorithm requires defining a set of main features $\mathcal{A}_\beta$ and a set of interaction features $\mathcal{A}_\Theta$ among which it iterates. It is defined by default to $\mathcal{A}_\beta = [\![1, p]\!]$ for the main effects and to $\mathcal{A}_\Theta = [\![1, q]\!]$ for the interactions, which allows to explore all possible features. Nevertheless, since the Elastic Net solution is expected to be sparse, many coefficients will be null and updated unnecessarily, thus generating an avoidable computational cost. Also, one of the challenges to have an efficient solver is to efficiently identify the set of relevant features, in order to build the most refined sets possible. We will address in Chapter 3 how to build these sets, to use this algorithm as a key inner solver.

2.1.3 Statistical results on toy example

Since it is difficult to replicate the true correlation (structure and level) of a real dataset, we decide to evaluate Elastic Net with Interactions and other methods with semi-artificial datasets, as in [Bühlmann and Mandozzi, 2014]. The semi artificial datasets are generated as follows:

$$y = \beta_0^\star \mathbb{1}_n + X\beta^\star + Z\Theta^\star + \varepsilon \ , \tag{2.13}$$

where X is a genomic dataset detailed in section 1.6.2, while $\beta_0^\star, \beta^\star$ and $\Theta^\star$ are true coefficients, chosen according different heredity structures and ε is a Gaussian noise.

2.1.3.1 Semi-generative datasets process

To evaluate the importance of the parameter κ, on different heredity scenarios and under different levels of sparsity, we choose to select a subset of samples and features, as a minimal example. Since the dataset have naturally many correlations, we decided to simply randomly select c columns and m rows, which constitute the design matrix X. Fo this illustration, it is $p = 30$ main effects, $q = 465$ interactions effects (including pure quadratic terms for simplicity), and with $n = 325$ samples.

Interactions construction. Regarding the interaction matrix Z, it is generated *on-the-fly* using element-wise product between columns of X, according to one of the two standardization process described in section 1.6.3. Hence, in both cases, the two design matrix X and Z are standardized such that each column has zero mean and variance equal to one. Doing this, we expect that no column has higher importance than others, in particular, main and interactions effects have equal importance in the generative process. Consequently, we also choose to set the truth intercept $\beta_0^\star$ to zero.

Heredity generative scenario. The support of the true coefficients, *i.e.,* the active coefficients of $\beta^\star$ and $\Theta^\star$ are chosen according to one of the five hierarchical interaction scenarios, detailed in section 1.2, that we briefly resume. In the Strong and Weak heredity scenarios eqs. (2.14) and (2.15), an interaction coefficient is active for the former if and only if both main coefficients are active, while for the latter, at least one main coefficient must be active. Alternatively, in anti-heredity scenario eq. (2.16), an interaction coefficient is active if and only if none of the associated main coefficients is active. Lastly, interactions only and main effects only eqs. (2.17) and (2.18) neglect main effects for the

first and then interactions for the latter.

$$\Theta_{i,j} \neq 0 \Rightarrow \beta_i \neq 0 \text{ and } \beta_j \neq 0 \ , \tag{2.14}$$

$$\Theta_{i,j} \neq 0 \Rightarrow \beta_i \neq 0 \text{ or } \beta_j \neq 0 \ , \tag{2.15}$$

$$\Theta_{i,j} \neq 0 \Rightarrow \beta_i = \beta_j = 0 \ , \tag{2.16}$$

$$\exists \ (i,j) \ \in [\![1,p]\!], \ \Theta_{i,j} \neq 0 \text{ and } \forall \ i \in [\![1,p]\!], \ \beta_i = 0 \ , \tag{2.17}$$

$$\exists \ i \in [\![1,p]\!], \ \beta_i \neq 0 \text{ and } \forall \ (i,j) \ \in [\![1,p]\!], \ \Theta_{i,j} = 0 \ . \tag{2.18}$$

Regarding the active coefficients of $\beta^\star$ and $\Theta^\star$, we choose them randomly between ± 1. Also, in the strong, weak and anti heredity setting, we choose randomly 10 main effects to be active and then have $[5, 10, 15, 20, 25\}$ possible active interactions, hence $\|\Theta^\star\|_0 \in \{5, 10, 15, 20, 25\}$. For the main only case and interaction only case, the number of possible features also lives in $\{5, 10, 15, 20, 25\}$, such that $\|\beta^\star\|_0 \in \{5, 10, 15, 20, 25\}$ and $\|\Theta^\star\|_0 \in \{5, 10, 15, 20, 25\}$.

Controlling noise. In addition, we control the noise by the signal-to-noise ratio (SNR), defined in [Bühlmann and Mandozzi, 2014] and that we adapt to interaction (2.19), to generate the noise ε from a normal distribution with zero mean and $\sigma \, \mathrm{Id}_{n\times n}$ variance-covariance matrix:

$$\mathrm{SNR} = \sqrt{\frac{\beta^{\star\top} X^\top X \beta^\star + \Theta^{\star\top} Z^\top Z \Theta^\star}{n\sigma^2}} \ . \tag{2.19}$$

For all the following, we set the SNR to 8.

Elastic Net with Interactions optimization parameters. As for the standard LASSO method, Elastic Net with Interactions performances rely on hyperparameters which require to be determined. To find them for CLEAR-Enet with Interactions, we implement cross validation, set by default to "five folds". Moreover, other important parameters are the number of α's in the grid, the depth ϵ of the grid and the tolerance stopping criteria of the different models, since they control both time resolution and statistical performance. For all the following, we use $n_\alpha = 100$ points in the grid of all methods, a depth $\epsilon = 0.001$ and a tolerance of 10^{-4} as stopping criteria, which seems standard regarding the classical implementation of LASSO solvers.

Performance metrics. For the following, we divide the data into train (80%) and test (20%) samples. We also define the following predictive error for the semi-artificial

datasets, since we know the original signal, which we will call MSE by abuse of language:

$$\text{MSE} = \frac{\left\| X_{test}\left(\beta^\star - \widehat{\beta}\right) + Z_{test}\left(\Theta^\star - \widehat{\Theta}\right)\right\|_2^2}{2n_{test}} \ . \tag{2.20}$$

In addition, to compare the feature selection ability of the methods, we measure precision, recall and as a trade-off between both, F_1-score. While these scores are generally used for binary classification problems, *i.e.,* to compare the predicted class and the true class, we used them to compare the true active support and the estimated one. In more detail, we note:

- True Positive (TP) are non-zero coefficients for both estimation and truth;
- True Negative (TN) are zero coefficients for both estimation and truth;
- False Negative (FN) are coefficients whose estimate is zero while the associated truth is non-zero;
- False Positive (FP) are estimated active coefficients, *i.e.,* non-zero, while in truth they are inactive, *i.e.,* null.

Therefore, we are able to compute precision, recall and F_1-score, whose measures closest to one are those associated with the best statistical performance.

- Precision: $\frac{\text{TP}}{\text{TP+FP}} \in [0, 1]$: measures how many false positive features are added by the estimator;
- Recall: $\frac{\text{TP}}{\text{TP+FN}} \in [0, 1]$: measures how many active features are retrieved by the estimator;
- F_1-score: $2 \times \frac{\text{precision} \times \text{recall}}{\text{precision+recall}} \in [0, 1]$: measures a trade-off between precision and recall, while allowing to take into account the high number of true negative coefficients, which are numerous in LASSO settings, expected to be parcimonious.

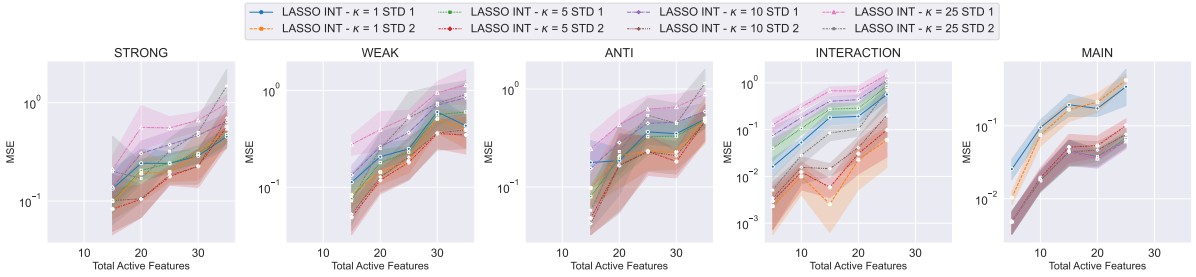
Moreover, the size of the estimated support is also investigated, since it is closely linked to model interpretability. Finally, in order to average the predictive and selection ability measures, we repeat each experiment ten times.

2.1.3.2 Statistical results

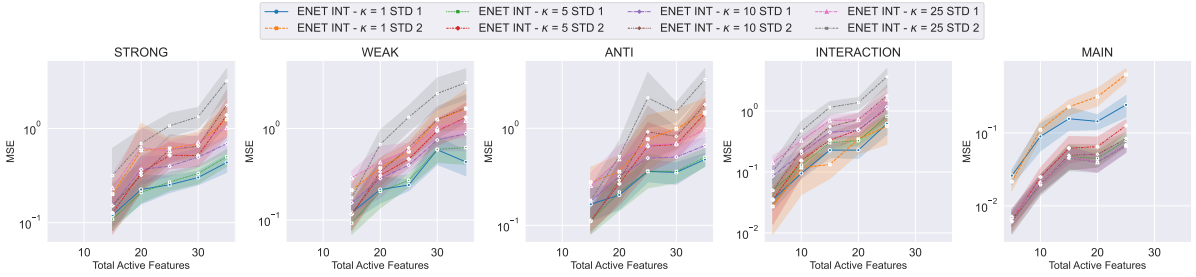
Predictive performance. Figure 2.1 illustrates the predictive performances of LASSO and Elastic Net with Interactions, in function of both standardization of the generative process in one hand, but also of estimation process, in function of sparsity level for the five possible hierarchical settings.

We observe that predictive performances can be improved by not penalizing equally the main features and the interactions ($\kappa = 1$). In particular, Figures 2.1a and 2.1b show that penalizing more the interactions improves the MSE when the generative process uses only the main effects, even if, in pure interaction setting, penalizing more the interactions increases the MSE.

For other scenarios, the MSE result depends on the standardization and the γ ratio level. For example, penalizing 5 more and penalizing equally, perform closely, while penalizing more *i.e.,* 10 or 25 leads to downgrade predictive performance. Lastly, the Elastic Net estimator performs better than LASSO in the first standardization setting, while inversely, LASSO performs better in the second standardization process, which we recall is the standardization setting with the least correlation, which explains the success of the LASSO.



(a) MSE: LASSO with Interactions case



(b) MSE: Elastic Net with Interactions case

Figure 2.1: Predictive performances of LASSO and Elastic Net with Interactions in function of different levels of interaction penalty (κ), in two different standardization settings. Note that the predictive performances can be improved by not penalizing main and interaction features equally ($\kappa = 1$).

Features selection performance. Figures 2.2 and 2.3 detail the precision, recall, F_1-score and the number of active features, *i.e.,* the support, in LASSO case for the former and Elastic Net for the latter.

In Figures 2.2 and 2.3, precision increases with κ, for both standardization, LASSO and Elastic Net with Interactions, hence penalizing interactions more than main effects helps to reduce the number of false positive.

In particular, for both LASSO and Elastic Net with Interactions estimators, precision favors the first standardization scheme. Regarding the recall score, the second standardization scheme helps the estimator to find all relevant features, since it does not matter: κ penalization level is close to one, in all hierarchical settings. Regarding the first standardization scheme, except in interactions settings, penalizing equally leads to poor results, in particular in LASSO case. Nevertheless, penalizing too much the interactions deteriorates the performances, since $\kappa = 5$ leads to the best recall result. Lastly, regarding F_1-score, penalizing equally leads to the poorly result, while for the penalized cases, the results are close (except for LASSO with $\kappa = 25$ in first standardization). Furthermore, it appears that support decreases with penalty on the one hand, while on the other hand, the second standardization scheme leads to bigger support.

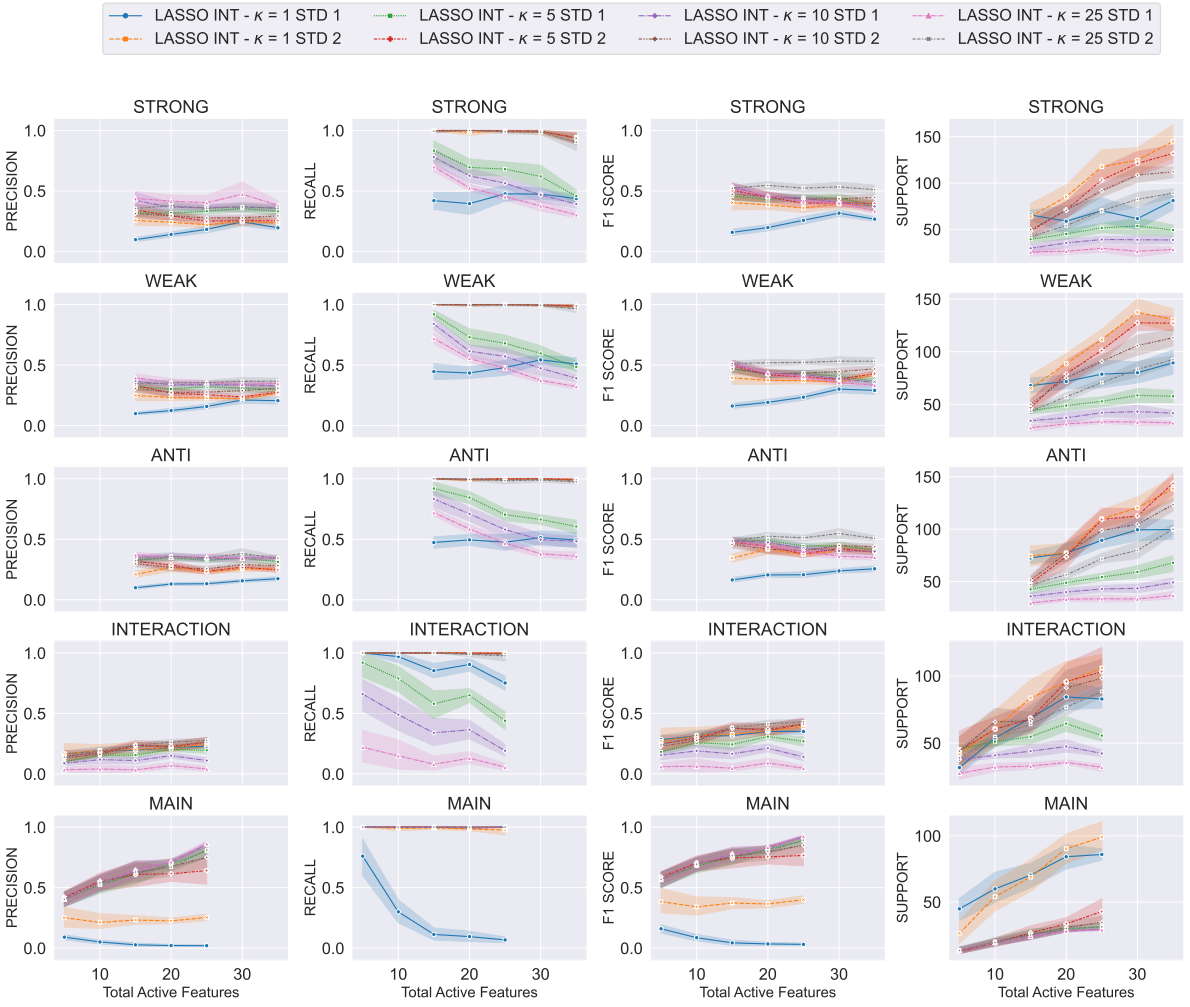


Figure 2.2: Features selection performances of LASSO with Interactions in function of different levels of interaction penalty (κ), in two different standardization settings. Note that predictive performances can be improved by not penalizing main and interaction features equally ($\kappa = 1$), in particular $\kappa = 5$ seems a good compromise.

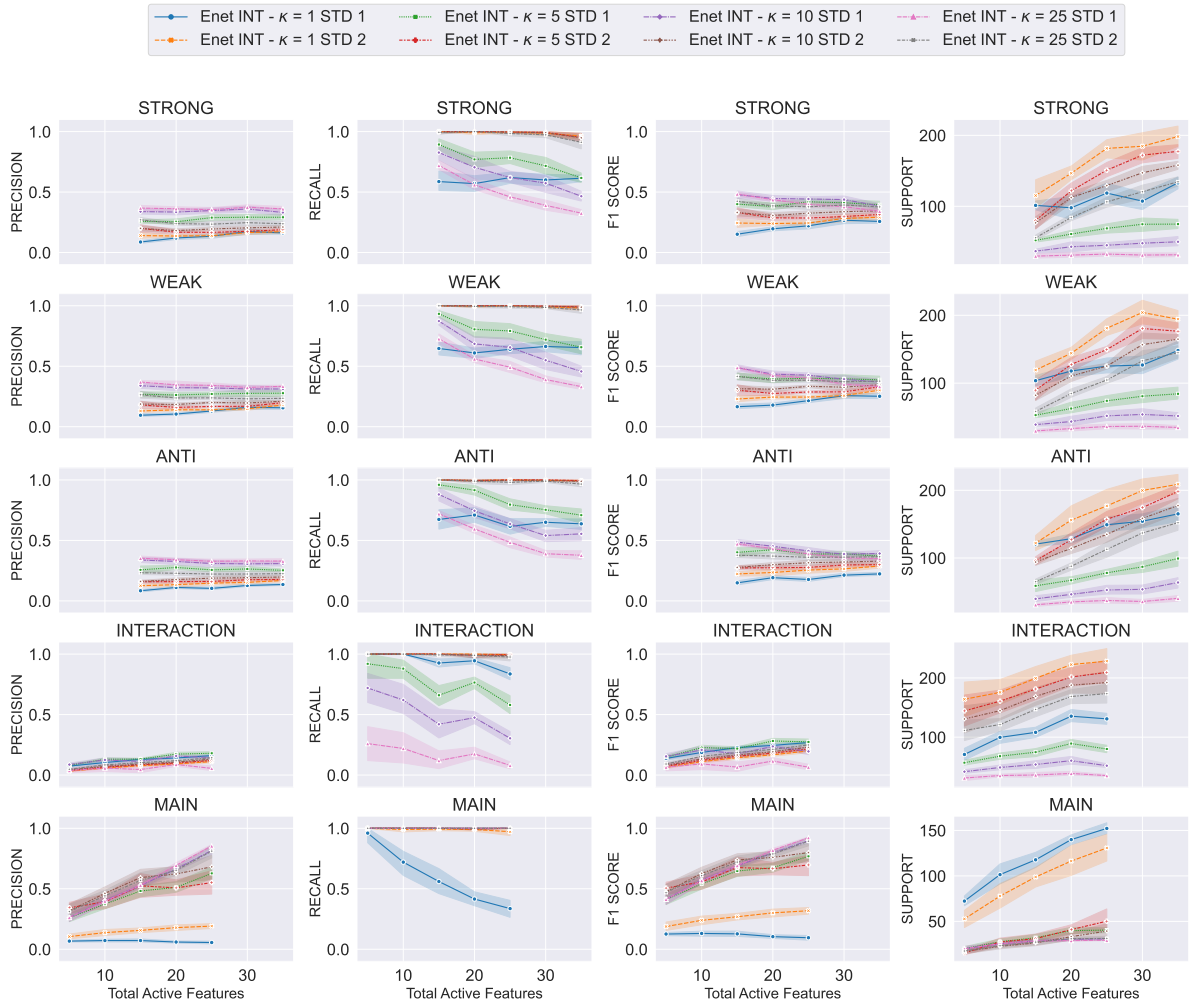


Figure 2.3: Feature selection performances of Elastic Net with Interactions as a function of different levels of interaction penalty (κ), in two different standardization settings. Note that predictive performances can be improved by not penalizing main and interaction features equally ($\kappa = 1$), in particular $\kappa = 5$ seems a good compromise.

Finally, LASSO with Interactions outperforms Elastic Net with Interactions compared to F_1-score, in all interaction settings, except when the generative signal is only composed of interactions.

In any case, penalized estimators such as LASSO or Elastic Net tend to produce biased coefficients, which may affect features selection by two points. Firstly, due to the penalization, the real coefficients tend to be shrinked toward zero, which may hurt predictive tasks. Secondly, to compensate the loss of magnitude of the fitted coefficients, LASSO and Elastic Net tend to add other features to balance. So on, unbiased estimate can be useful, especially when feature selection is the final aim of a statistical study.

2.2 CLEAREnet: a debiased Elastic Net

Since Naive-LSEnet approach (section 1.4) has many drawbacks, we instantiate Covariant LEAst-square Refitting (CLEAR) [Deledalle et al., 2017]. Thus, while estimating the coefficients with Elastic Net with Interactions, CLEAR allows to obtain a debiased version. A key advantage is that it can fit well with our coordinate descent algorithm. In addition, CLEAR preserves the theoretical properties of the original estimator, the sparsity in our case. Moreover, in most cases, CLEAR finds the same solution as a post-refitting with Least Squares, here Naive-LSEnet, while being more robust. Indeed, since its estimation is done jointly with the initial estimator, it is less sensitive to the support identification error than the methods identifying the support after the estimation. Statistically, we have shown in [Bascou et al., 2021] that debiasing Elastic Net with Interactions with CLEAR effectively leads to fewer features being selected for a similar prediction error. We named CLEARLASSO with Interactions the resulting method, when $\gamma = 1$, and CLEAR-Enet with Interactions otherwise.

Section 2.2.1 describes the CLEAR framework, while in section 2.2.2 we adapt an automatic differentiation scheme to compute the Jacobian of Elastic Net with Interactions. Then, we provide computational tricks for this scheme in section 2.2.3, and lastly, section 2.2.4 provides statistical results and illustrates the numerical cost of the debiasing step.

2.2.1 CLEAR framework

CLEAR is based on a first order correction of the coefficients [Tukey, 1977, Osher et al., 2005], which for main effects estimator is defined as follows:

Definition 2.2.1. *The CLEAR estimator of an almost everywhere (a.e.) differentiable estimator* $\mathbb{R}^n \ni y \mapsto \widehat{\beta}(y) \in \mathbb{R}^p$ *is, for all* $y \in \mathbb{R}^n$*:*

$$\widetilde{\beta}(y) := \widehat{\beta}(y) + \rho J(y - X\widehat{\beta}(y)) \text{ with } \rho := \begin{cases} \dfrac{\langle XJ\delta | \delta \rangle}{\|XJ\delta\|^2} & , \text{ if } XJ\delta \neq 0, \\ 1 & , \text{ otherwise,} \end{cases} \tag{2.21}$$

where $\delta = y - X\widehat{\beta}(y)$ *et* $J = J_{\widehat{\beta}}(y) = \dfrac{\partial \widehat{\beta}(y)}{\partial y} \in \mathbb{R}^{p \times n}$ *is the Jacobian matrix of* $\widehat{\beta}$ *at* y*.*

We observe that CLEAR's definition mainly involves some matrix products in addition to the Jacobian of the original estimator, Elastic Net in our case.

CLEAR background The CLEAR estimator follows a first debiasing work [Deledalle et al., 2015]. In this paper, the debiased estimator called *invariant refitting*, proposes

a linearly invariant refitting of the initial estimator, allowing to keep some properties of the initial estimator. For example, in the LASSO case, looking for the best linearly invariant estimator of the LASSO estimator, is equivalent to orthogonally projecting y onto the subset of β coefficients sharing the same support as the LASSO solution. Thus, searching for a linearly invariant estimator in the LASSO case preserves sparsity and is equivalent to performing the Naive-LSLASSO procedure. However, this approach fails to recover certain first-order properties of the initial estimator, called covariant, such as the regularity of the solution, which is desirable in some contexts.

The CLEAR estimator is motivated to capture these first-order properties of the initial estimator by imposing to preserve, at least locally, the Jacobian structure of the estimator. Thus, intuitively, a covariant refitting estimator $\widetilde{\beta}(y)$ of an initial estimator $\widehat{\beta}(y)$ should be a solution of the following constrained problem (definition 5 from [Deledalle et al., 2017], with $z = y$):

$$\widetilde{\beta} \in \underset{\beta \in \mathcal{H}}{\arg\min} \|X\beta(y) - y\|_2^2 \quad , \tag{2.22}$$

where $\mathcal{H}$ is the set of maps $\beta : \mathbb{R}^n \to \mathbb{R}^p$ satisfying for all $y \in \mathbb{R}^n$:

1. Affine map: $\beta(y) = Ay + b$ for some $A \in \mathbb{R}^{n \times p}$, $b \in \mathbb{R}^p$,
2. Covariant preserving: $J_\beta(y) = \rho J_{\widehat{\beta}}(y)$ for some $\rho \in \mathbb{R}$,
3. Coherent map: $\beta\left(X\widehat{\beta}(y)\right) = \widehat{\beta}(y)$.

The first constraint is to restrict the debiased estimator $\widetilde{\beta}(y)$ to a class of estimators that are easy to compute. The second constraint enforces to maintain locally the Jacobian between the initial $\widehat{\beta}(y)$ estimator and the debiased estimator $\widetilde{\beta}(y)$, to preserve the first-order properties of the original estimator. Then, the third constraint means that the application of refitting to a prediction made with the initial estimator, must not change the latter.

Lastly, it was proved [Deledalle et al., 2017, Theorem 6] that an estimator respecting this constrained problem is necessarily the CLEAR estimator, with formula given in definition 2.2.1. Hence, the CLEAR estimator preserves the sparsity but also the first order properties of the original estimator. In the following, we start by deriving a simple one-dimensional example to show how CLEAR works.

One-dimensional example: Let $(x, y) \in \mathbb{R}^2$, we want to solve:

$$\underset{\beta \in \mathbb{R}}{\arg\min}\, F(\beta, y) := \underset{\beta \in \mathbb{R}}{\arg\min} \left(\frac{1}{2}(\beta - y)^2 + \alpha_{1,1}|\beta| + \frac{\alpha_{2,1}}{2}\beta^2 \right) \quad . \tag{2.23}$$

We get the following (partial) sub-gradient for F:

$$\partial_\beta \left(F(\beta, y)\right) = (\beta - y) + \alpha_{1,1}\partial(|\beta|) + \alpha_{2,1}\beta = (1+\alpha_{2,1})\beta - y + \alpha_{1,1}\partial(|\beta|) \ , \tag{2.24}$$

So, with the following convention $(\cdot)_+ = \max(\cdot, 0)$, Fermat rule gives:

$$0 \in \partial_\beta \left(F\left(\hat{\beta}, y\right)\right) \iff \hat{\beta} = \frac{\operatorname{sign}(y)(|y| - \alpha_{1,1})_+}{(1+\alpha_{2,1})} \ . \tag{2.25}$$

Finally, a simple estimator of β is:

$$\widehat{\beta}(y) = \frac{\operatorname{sign}(y)(|y| - \alpha_{1,1})_+}{(1+\alpha_{2,1})} = \begin{cases} \frac{y + \operatorname{sign}(y)\alpha_{1,1}}{(1+\alpha_{2,1})} & , \text{ if } |y| > \alpha_{1,1}, \\ 0 & , \text{ if } |y| \leqslant \alpha_{1,1}. \end{cases} \tag{2.26}$$

We can see that y is shrunk in two ways: by the subtraction and the division associated respectively to the ℓ_1 and ℓ_2 regularization. We obtain the following Jacobian J and δ:

$$J = \begin{cases} \frac{1}{1+\alpha_{2,1}}, & \text{if } |y| > \alpha_{1,1} \\ 0, & \text{if } |y| \leqslant \alpha_{1,1} \end{cases} \ ; \ \delta = \begin{cases} \frac{\alpha_{2,1} y - \alpha_{1,1}\operatorname{sign}(\widehat{\beta}(y))}{(1+\alpha_{2,1})} & \text{if } |y| > \alpha_{1,1} \ , \\ y & \text{if } |y| \leqslant \alpha_{1,1} \ . \end{cases} \tag{2.27}$$

Thus,

$$XJ\delta = \begin{cases} \frac{\alpha_{2,1} y - \alpha_{1,1}\operatorname{sign}(\widehat{\beta}(y))}{(1+\alpha_{2,1})^2} & \text{if } |y| > \alpha_{1,1} \ , \\ 0 & \text{otherwise} \ . \end{cases} \tag{2.28}$$

Hence, if $|y| > \alpha_{1,1}$ we get:

$$\|XJ\delta\|^2 = \frac{\left(\alpha_{2,1} y - \alpha_{1,1}\operatorname{sign}(\widehat{\beta}(y))\right)^2}{(1+\alpha_{2,1})^4} \text{ and } \langle XJ\delta | \delta \rangle = \frac{\left(\alpha_{2,1} y - \alpha_{1,1}\operatorname{sign}(\widehat{\beta}(y))\right)^2}{(1+\alpha_{2,1})^3} \ . \tag{2.29}$$

Then,

$$\rho = \begin{cases} (1+\alpha_{2,1}) & \text{if } |y| > \alpha_{1,1} \ , \\ 1 & \text{if } |y| \leqslant \alpha_{1,1} \ , \end{cases} \quad \text{so} \quad \rho J = \begin{cases} 1 & \text{if } |y| > \alpha_{1,1} \ , \\ 0 & \text{if } |y| \leqslant \alpha_{1,1} \ . \end{cases} \tag{2.30}$$

Finally, the CLEAR estimator associated to this one-dimensional Elastic Net is:

$$\widetilde{\beta}(y) = \widehat{\beta}(y) + \rho J(y - X\widehat{\beta}(y)) = \begin{cases} y & \text{if } |y| > \alpha_{1,1} \ , \\ 0 & \text{if } |y| \leqslant \alpha_{1,1} \ . \end{cases} \tag{2.31}$$

Hence, one recovers the hard-thresholding in this context.

2.2.2 Adapting CLEAR to Elastic Net with Interactions

Then, from CLEAR Definition 2.2.1, we immediately get CLEAR for interactions.

Definition 2.2.2 (CLEAR for Interactions)**.** *The Covariant LEAst-square Refitting associated to an a.e. differentiable estimator* $y \mapsto \left(\widehat{\beta}(y), \widehat{\Theta}(y)\right)$ *is, for almost all* $y \in \mathbb{R}^n$*, given by:*

$$\widetilde{\beta}^{(t+1)}(y) = \widehat{\beta}(y) + \rho J_{\widehat{\beta}(y)} \left(y - X\widehat{\beta}(y) - Z\widehat{\Theta}(y)\right) \quad , \tag{2.32}$$

$$\widetilde{\Theta}^{(t+1)}(y) = \widehat{\Theta}(y) + \rho J_{\widehat{\Theta}(y)} \left(y - X\widehat{\beta}(y) - Z\widehat{\Theta}(y)\right) \quad . \tag{2.33}$$

where:

$$\rho = \begin{cases} \frac{\langle WJ\delta | \delta \rangle}{\|WJ\delta\|}, & \text{if}, \quad WJ\delta \neq 0 \\ 1, & \text{otherwise}, \end{cases} \quad \text{with} \quad W = [X, Z] \in \mathbb{R}^{n \times (p+q)} \quad . \tag{2.34}$$

and $\delta = y - X\widehat{\beta}(y) - Z\widehat{\Theta}(y)$ *and* $J = \left(J_{\widehat{\beta}(y)}, J_{\widehat{\Theta}(y)}\right)$ *is the Jacobian matrix of* $\left(\widehat{\beta}(y), \widehat{\Theta}(y)\right)$ *at the point* y*.*

The main difficulty in our case is the computation of the two Jacobians of $\left(\widehat{\beta}(y), \widehat{\Theta}(y)\right)$ at the point y, *i.e.*, get $J_{\widehat{\beta}(y)}$ and $J_{\widehat{\Theta}(y)}$ To compute it, we adapt an automatic differentiation scheme, proposed by Deledalle et al. [2014], which leads to the following iterative coefficients updates:

Proposition 2.2.1. *Let us suppose that the coefficients* $\widehat{\beta}(y)$ *and* $\widehat{\Theta}(y)$ *are iteratively updated with the following scheme, where we define residuals as* $r = y - X\widehat{\beta}^{(t)} + Z\widehat{\Theta}^{(t)}$*:*

$$\widehat{\beta}_j^{(t+1)} = \frac{\mathrm{ST}\left(x_j^\top \left(r + \widehat{\beta}_j^{(t)} x_j\right), n\alpha_{1,1}\right)}{\|x_j\|^2 + n\alpha_{2,1}} \quad , \tag{2.35}$$

$$\widehat{\Theta}_{jj}^{(t+1)} = \frac{\mathrm{ST}\left(z_{jj}^\top \left(r + \widehat{\Theta}_{jj}^{(t)} z_{jj}\right), n\alpha_{1,2}\right)}{\|z_{jj}\|^2 + n\alpha_{2,2}} \quad . \tag{2.36}$$

Nothing that e_j *and* e_{jj} *are the canonical basis vector of* $\mathbb{R}^p$ *and* $\mathbb{R}^q$ *respectively, we can compute iteratively the Jacobian of* β *and* Θ *applied to the residuals* r*:* $J_{\widehat{\beta}_j^{(t+1)}} r$ *and* $J_{\widehat{\Theta}_{jj}^{(t+1)}} r$*, with the scheme:*

$$J_{\widehat{\beta}_j}^{(t+1)} r = \frac{\left(e_j \|x_j\|^2 - X^\top x_j\right)^\top J_\beta^{(t)} r - \left(x_j^\top Z\right)^\top J_\Theta^{(t)} r + x_j^\top r}{\|x_j\|^2 + n\alpha_{2,1}} \mathbb{1}_{\left\{\left|x_j^\top \left(r + \widehat{\beta}_j^{(t)} x_j\right)\right| \geqslant n\alpha_{1,1}\right\}} \quad , \tag{2.37}$$

$$J_{\widehat{\Theta}_{jj}}^{(t+1)} r = \frac{\left(e_{jj} \|z_{jj}\|^2 - Z^\top z_{jj}\right)^\top J_\Theta^{(t)} r - \left(X^\top z_{jj}\right)^\top J_\beta^{(t)} r + z_{jj}^\top r}{\|z_{jj}\|^2 + n\alpha_{2,2}} \mathbb{1}_{\left\{\left|z_{jj}^\top \left(r + \widehat{\Theta}_{jj}^{(t)} z_{jj}\right)\right| \geqslant n\alpha_{1,2}\right\}} \quad . \tag{2.38}$$

Hence, with $\rho^{(t+1)} = \frac{\left\langle [X,Z][J_{\hat{\beta}}^{(t+1)} r,\ J_{\hat{\Theta}}^{(t+1)} r]^\top, r^{(t+1)} \right\rangle}{\left\| [X,Z][J_{\hat{\beta}}^{(t+1)} r,\ J_{\hat{\Theta}}^{(t+1)} r]^\top \right\|^2}$*, the CLEAR estimate of Elastic Net with Interactions, which we call CLEAR-Enet with Interactions is:*

$$\widetilde{\beta}^{(t+1)} = \widehat{\beta}^{(t+1)} + \rho^{(t+1)} J_{\hat{\beta}}^{(t+1)} r \ , \tag{2.39}$$

$$\widetilde{\Theta}^{(t+1)} = \widehat{\Theta}^{(t+1)} + \rho^{(t+1)} J_{\hat{\Theta}}^{(t+1)} r \ . \tag{2.40}$$

Proof. We note: $t_j = x_j^\top \left(r^k + \widehat{\beta}_j^{(t)} x_j \right)$ and recall that $J_{\widehat{\beta}_j^{(t+1)}} = \dfrac{\partial \widehat{\beta}_j^{(t+1)}}{\partial y}$, so:

$$\begin{aligned}
\frac{\partial \widehat{\beta}_j^{(t+1)}}{\partial y} &= \frac{1}{\|x_j\|^2 + n\alpha_{2,1}} \frac{\partial \, \mathrm{ST}\left(t_j, n\alpha_{1,1}\right)}{\partial y} \\
&= \frac{1}{\|x_j\|^2 + n\alpha_{2,1}} \left(\frac{\partial \, \mathrm{ST}\left(t_j, n\alpha_{1,1}\right)}{\partial \beta} \frac{\partial \widehat{\beta}^{(t)}}{\partial y} + \frac{\partial \, \mathrm{ST}\left(t_j, n\alpha_{1,1}\right)}{\partial \Theta} \frac{\partial \widehat{\Theta}^{(t)}}{\partial y} + \frac{\partial \, \mathrm{ST}\left(t_j, n\alpha_{1,1}\right)}{\partial y} \right) \\
&= \frac{1}{\|x_j\|^2 + n\alpha_{2,1}} \left(\frac{\partial \, \mathrm{ST}\left(t_j, n\alpha_{1,1}\right)}{\partial \beta} J_{\widehat{\beta}^{(t)}} + \frac{\partial \, \mathrm{ST}\left(t_j, n\alpha_{1,1}\right)}{\partial \Theta} J_{\widehat{\Theta}^{(t)}} + \frac{\partial \, \mathrm{ST}\left(t_j, n\alpha_{1,1}\right)}{\partial y} \right) \\
&= \frac{1}{\|x_j\|^2 + n\alpha_{2,1}} \left(\left(e_j \|x_j\|^2 - x_j^\top X \right)^\top J_{\widehat{\beta}^{(t)}} - x_j^\top Z^\top J_{\widehat{\Theta}^{(t)}} + x_j^\top \right) \mathbb{1}_{\{|t_j| \geqslant n\alpha_{1,1}\}} \ .
\end{aligned}$$

Finally, by factorizing on the left by $x_j^\top$ and thus applied residuals r on the right, we get:

$$J_{\hat{\beta}_j}^{(t+1)} r = \frac{\left(e_j \|x_j\|^2 - X^\top x_j \right)^\top J_\beta^{(t)} r - \left(x_j^\top Z \right)^\top J_\Theta^{(t)} r + x_j^\top r}{\|x_j\|^2 + n\alpha_{2,1}} \mathbb{1}_{\left\{ \left| x_j^\top \left(r + \widehat{\beta}_j^{(t)} x_j \right) \right| \geqslant n\alpha_{1,1} \right\}} \ . \tag{2.41}$$

The results for the Jacobian of interactions can be established in a similar way. □

Nevertheless, we observe that computing the Jacobian involves many matrix products, in Equations (2.37) and (2.38): $X^\top x_j, Z^\top x_j, X^\top z_{jj}, Z^\top z_{jj}$. While these matrix products are identical from one coordinate pass to the next, in the context of interaction solvers, we can not compute them once and store the result before running the algorithm.

Indeed, the last product, *i.e.,* $Z^\top z_{jj}$, must be performed for each possible interaction and then leads to save $Z^\top Z \in \mathbb{R}^{q \times q}$, which is in general much larger than $Z \in \mathbb{R}^{n \times q}$. Therefore, we can not store this whole matrix, but on the other hand, computing such products $Z^\top Z \in \mathbb{R}^{q \times q}$ *on-the-fly* each time the coefficient is updated represents a much too large computational burden.

Since storing a too large matrix or calculating *on-the-fly* are not tractable solutions, we detail in the next section the calculation tricks that make this possible.

2.2.3 A tractable version of CLEAR-Enet with Interactions

In this section, we rely on a computational trick from Bertrand et al. [2020] to make Jacobian computations possible. Taking Equation (2.38), the full computation is not $Z^\top z_{jj}$ but $z_{jj}^\top Z J_\Theta^{(t+1)} r$, however the second part $Z J_\Theta^{(t+1)} r$ lies in $\mathbb{R}^n$, which is easy to store. Hence, $dr_\beta = X J_\beta^{(t+1)} r \in \mathbb{R}^n$ and $dr_\Theta = Z J_\Theta^{(t+1)} r \in \mathbb{R}^n$ can be updated efficiently (lines 5 and 13) and allow efficient Jacobian updates rules (lines 4 and 12 of Algorithm 3).

Algorithm 3: One epoch of coordinate descent algorithm with CLEAR

Input : $X,\ y,\ \beta^{(t)},\ \Theta^{(t)},\ J_{\beta_j}^{(t)} r,\ J_{\Theta_j}^{(t)} r,\ \alpha,\ \mathcal{A}_\beta,\ \mathcal{A}_\Theta$

Init : $r = y - X\widehat{\beta}^{(t)} - Z\widehat{\Theta}^{(t)}$, $dr_\beta = X J_{\beta_j}^{(t)} r$, $dr_\Theta = Z J_{\Theta_j}^{(t)} r$

1 **for** $j \in \mathcal{A}_\beta$ **do** // set of updated main features

2 $\quad \beta_j^{(t+1)} = \frac{1}{\|x_j\|^2 + n\alpha_{2,1}} \mathrm{ST}(x_j^\top (r + \beta_j^{(t)} x_j), n\alpha_{1,1})$

3 $\quad$ **if** $\beta_j^{(t+1)} \neq \beta_j^{(t)}$ **then** // debiasing main features

4 $\quad\quad J_{\beta_j}^{(t+1)} r = \dfrac{\left(e_j \|x_j\|^2\right)^\top J_\beta^{(t)} r + x_j^\top (dr_\beta + dr_\Theta) + x_j^\top r}{\|x_j\|^2 + n\alpha_2} \mathbb{1}_{\{|x_j^\top (r + \beta_j^{(t)} x_j)| \geqslant n\alpha_1\}}$

5 $\quad\quad dr_{\beta_j}$ += $\left(J_{\beta_{jj}}^{(t)} r - J_{\beta_{jj}}^{(t+1)} r\right) x_j$ // update computational trick

6 $\quad\quad r$ += $\left(\beta_j^{(t)} - \beta_j^{(t+1)}\right) x_j$ // update residuals

7 **for** $jj \in \mathcal{A}_\Theta$ **do** // set of updated interaction features

8 $\quad j_1, j_2 = \mathcal{R}_{jj}$ // Rosetta transforms index of Z in double indices

9 $\quad z_{jj} = x_{j_1} \odot x_{j_2}$

10 $\quad \Theta_{jj}^{(t+1)} = \frac{1}{\|z_{jj}\|^2 + n\alpha_{2,2}} \mathrm{ST}(z_{jj}^\top (r + \Theta_{jj}^{(t)} z_{jj}), n\alpha_{1,2})$

11 $\quad$ **if** $\Theta_{jj}^{(t+1)} \neq \Theta_{jj}^{(t)}$ **then** // debiasing interactions features

12 $\quad\quad J_{\Theta_{jj}}^{(t+1)} r = \dfrac{(e_{jj} \|z_{jj}\|^2)^\top J_\Theta^{(t)} r + z_{jj}^\top (dr_\beta + dr_\Theta) + z_{jj}^\top r}{\|z_{jj}\|^2 + n\alpha_2} \mathbb{1}_{\{|z_{jj}^\top (r + \Theta_{jj}^{(t)} z_{jj})| \geqslant n\alpha_1\}}$

13 $\quad\quad dr_{\Theta_{jj}}$ += $\left(J_{\Theta_{jj}}^{(t)} r - J_{\Theta_{jj}}^{(t+1)} r\right) z_{jj}$ // update computational trick

14 $\quad\quad r$ += $\left(\Theta_{jj}^{(t)} - \Theta_{jj}^{(t+1)}\right) z_{jj}$ // update residuals

Output : $\beta^{(t+1)},\ \Theta^{(t+1)},\ J_\beta^{(t+1)} r,\ J_\Theta^{(t+1)} r$

For Algorithms 2 and 3, we initialize all the terms $\beta^{(t)}$, $\Theta^{(t)}$ and $J_{\beta_j}^{(t)} r$, $J_{\Theta_j}^{(t)} r$ into null vectors at the first regularization parameters of the path, while for the others, we re-use the solution provided by the previous path regularization parameter. In addition, residuals $r = y - X\widehat{\beta}^{(t)} - Z\widehat{\Theta}^{(t)}$ and computational tricks $dr_\beta = X J_{\beta_j}^{(t)} r$, $dr_\Theta = Z J_{\Theta_j}^{(t)} r$ are not re-computed at each novel pass but updated *on-the-fly*; they are in initialization part of the algorithms only to ease their presentation.

Lastly, when the stopping criterion of Elastic Net with Interactions (defined section 3.1) is verified, the updates of the Jacobian stop. The algorithm ends with the computation of $\rho^{(t+1)}$ (eq. (2.34)), allowing the debiasing of the coefficients.

2.2.4 Statistical results on a toy example

To discuss the effects of debiasing on statistical and computational performance, we reuse the generative process detailed previously in section 2.2.4. The first part deals with the predictive and selection ability of the estimator, while the second part illustrates the computational cost of the debiasing step.

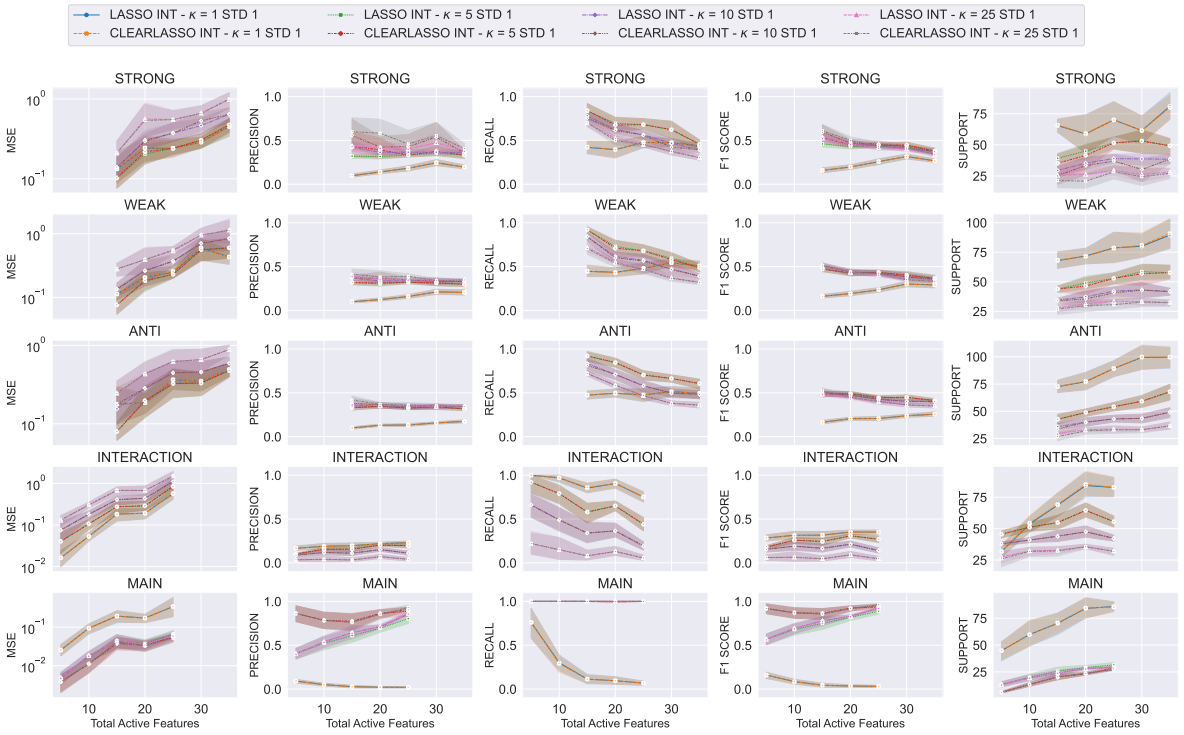
2.2.4.1 Statistical results with a debiasing step

For ease of analysis, we have separated standardization schemes into different figures. Hence, Figures 2.4a and 2.5a show CLEARLASSO and CLEAR-Enet with Interactions performances in the first standardization scheme, while Figures 2.4b and 2.5b show it for the second.

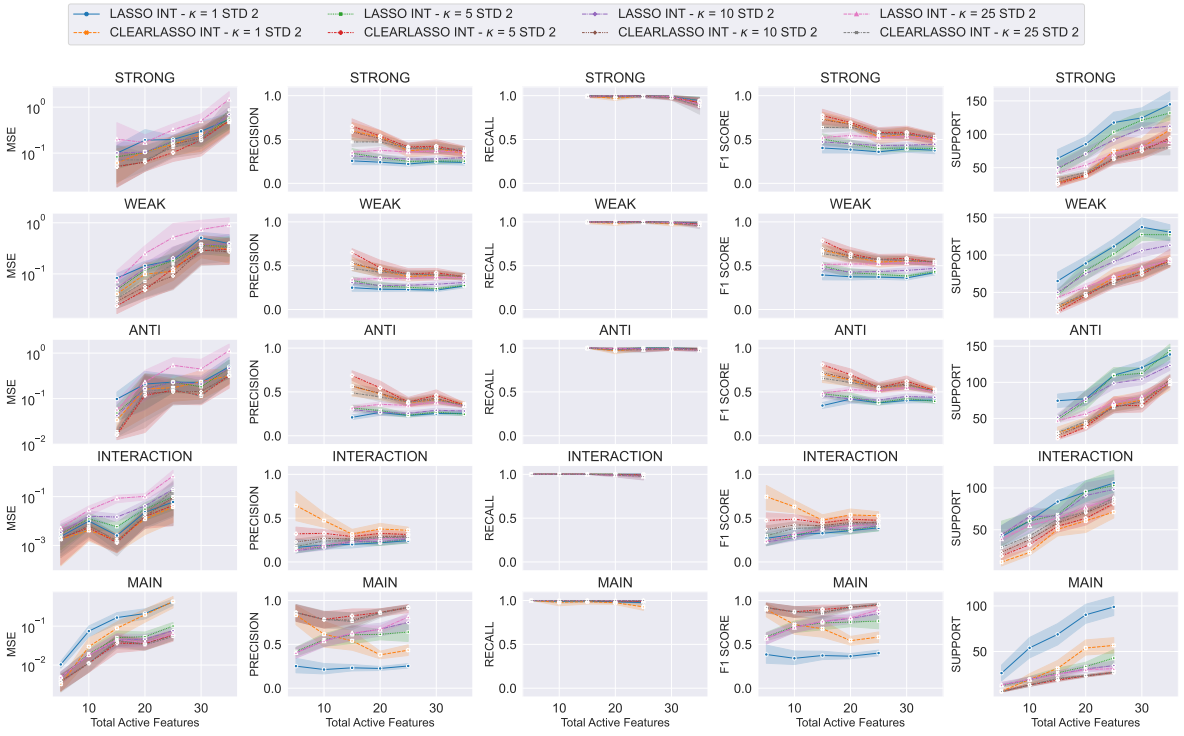
Predictive performances. We first focus on the MSE measure, shown in the first column of each figure. For the first standardization scheme, CLEAR-Enet with Interactions as well as CLEARLASSO with Interactions do not improve the predictive performances compared to their non-debiased version. Nevertheless, the second shows that the debiasing step slightly improves the statistical performance, in particular for the LASSO case.

Features selection performance. As previously, we measure precision, recall, F_1-score and number of active features, and we observe that standardization matters a lot. Indeed, the first standardization scheme shows that for both LASSO and Elastic Net, the debiasing step does not improve but does not degrade these measures either, except for main only settings, where debiasing improves precision without deteriorating the recall. However, for the second standardization scheme, the debiasing step improves the precision in all interaction settings, *i.e.,* it reduces the number of false positives, in particular for the LASSO, without deteriorating the recall. Therefore, this second case greatly improves the F_1-score, and reduces the number of active features, which implies a better interpretability for the costumers.

In conclusion, while performing a debiasing step neither improves nor degrades the statistical performances in the first standardization setting, it shows in the second one a great improvement on selection ability performances. Moreover, we also observe a slight improvement for MSE whereas it reduces the number of active features, as expected. Lastly, for all the following, we will consider CLEARLASSO with Interactions with $\kappa = 5$, as it showed the best results overall all parameters.

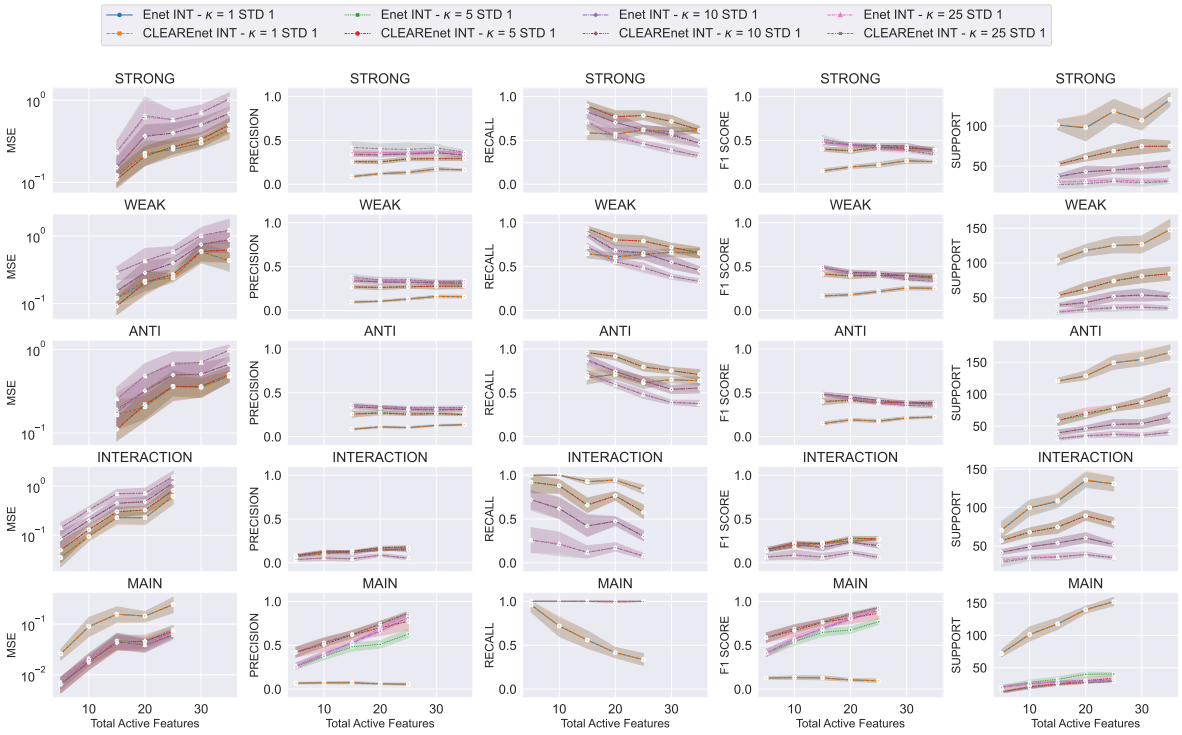


(a) CLEARLASSO with Interactions case with first standardization scheme

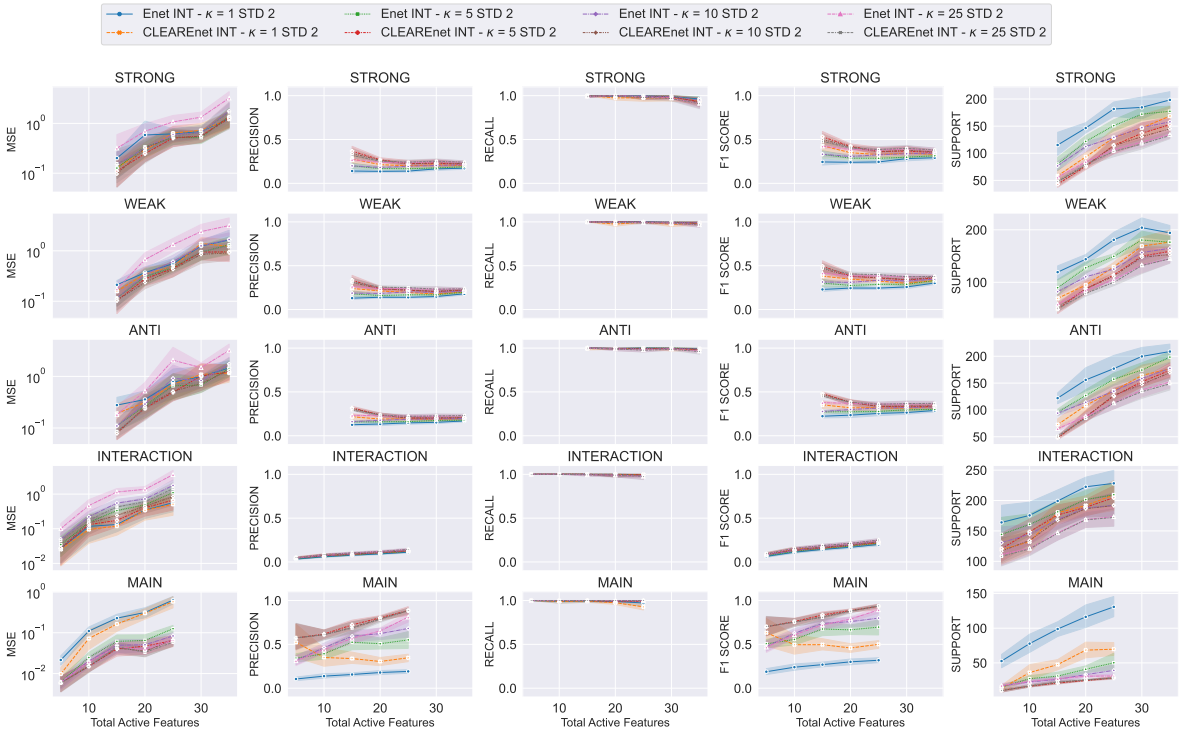


(b) CLEARLASSO with Interactions case with second standardization scheme

Figure 2.4: Performances of CLEARLASSO with Interactions in function of different levels of interaction penalty (κ), in both standardization settings. While first standardization does not imply improvement with debiasing step, second slightly improve MSE, greatly improve selection ability for a reduced number of active features.



(a) CLEAR-Enet with Interactions case with first standardization scheme



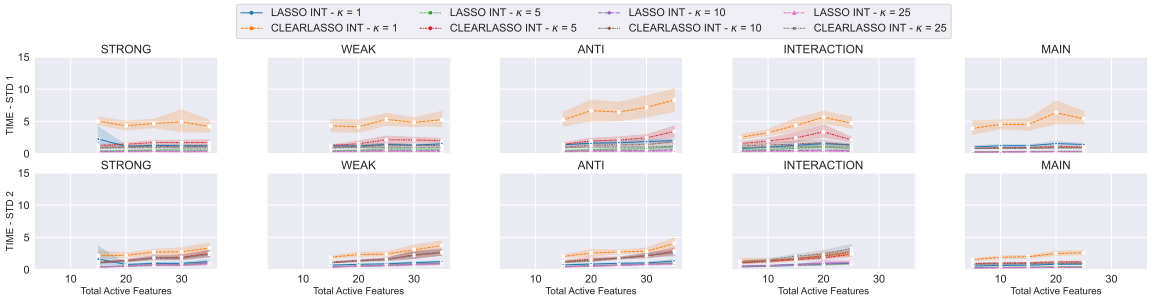
(b) CLEAR-Enet with Interactions case with second standardization scheme

Figure 2.5: Performances of CLEAR-Enet with Interactions in function of different levels of interaction penalty (κ), in both standardization settings. While first standardization does not imply improvement with debiasing step, second slightly improve MSE, greatly improve selection ability for a reduced number of active features, even if the difference are less visible than for CLEARLASSO.

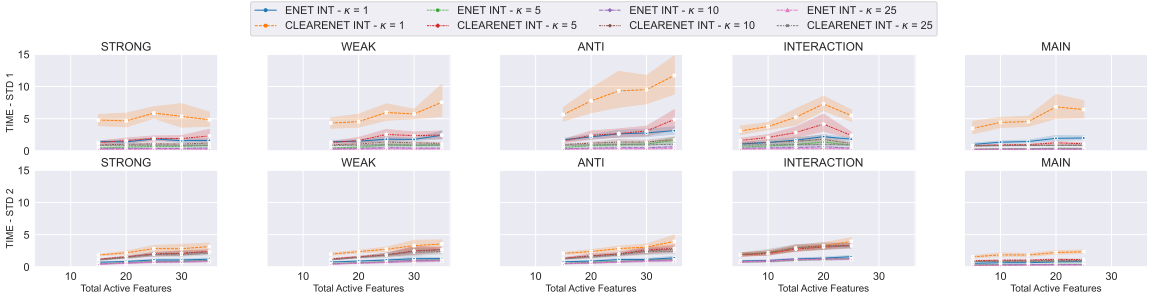
2.2.4.2 Impact of the debiasing step on computation time

Lastly, we detail in Figure 2.6 the impact of the debiasing step on the computation time. The times given are those obtained with the active set algorithm developed in Chapter 3, and not those with Algorithms 2 and 3, since the only purpose of this figure is to illustrate the cost of the debiasing step.

Obviously, the debiasing step increases computing time, but as detailed in Figure 2.6, CLEAR-Enet with Interactions and CLEARLASSO with Interactions have a comparable computing time to the non-debiased version. Indeed, debiasing only increases the computational cost by a factor 2 or 3, however, choice of κ and γ have an impact on computation time. For example, with the active set solver, the LASSO estimator is faster than the Elastic Net and penalizes more the interactions than the main effects. It also reduces the computing time, since fewer interactions have to be visited.



(a) CLEARLASSO with Interactions case for both standardization schemes



(b) CLEAR-Enet with Interactions case for both standardization schemes

Figure 2.6: Computation time (in second) of Elastic Net with Interactions and LASSO with Interactions with and without debiasing step. While debiasing increases by 2 or 3 the computational time, others parameters such as κ and γ also matter.

Finally, these experiments confirm that CLEARLASSO with Interactions, with $\kappa = 5$ is also a good compromise, since it achieves a better computational time than $\kappa = 1$ but close to $\kappa = 10$ and $\kappa = 25$.

2.3 Conclusion

In this chapter, we have presented an Elastic Net with Interactions, adapted from Elastic Net. The first step was to propose a reparametrization of the hyperparameters, to reduce the number of parameters to be adjusted and the associated computation time. One of the main ideas of this reparametrization is to penalize the interactions more than the main effects, to favor the latter in the model. The first results have shown that this penalization allows significantly improving the performance of the feature selection, without degrading the predictive performance, if the level of penalization remains moderate (*i.e.,* $\kappa = 5$).

Then, we developed a debiased version of our estimator, with the objective of selecting fewer coefficients than the Elastic Net with Interactions, for a similar prediction error. The first statistical results showed that the contribution of the debiasing step depends mainly on the chosen standardization. In all cases, the debiasing step does not deteriorate the performances of the Elastic Net with Interactions, and allows to significantly improve them in the second standardization scenario (STD2).

Finally, these experiments were made possible by the algorithms developed in this chapter, adapted from coordinate descent, which allows to build the interactions matrix *on-the-fly* without having to store it entirely in memory. The second numerical issue is then to take advantage of the parsimony of the Elastic Net to limit the number of interaction features to be updated.

The next chapter develops and illustrates the numerical cost of the duality gap in our case, and then proposes an active set algorithm, to take advantage of the parsimony and to reduce the computation time.

Chapter 3

An accelerated algorithm for Elastic Net with Interactions

Contents

In this chapter, we describe an optimization framework allowing to develop a scalable algorithm to estimate Elastic Net with Interactions and CLEAR-Enet with Interactions. The first part section 3.1 focuses on the choice of a stopping criterion, a point that has not been detailed so far. Computing a stopping criterion, *e.g.,* a duality gap, might cost as much as one pass over the full set of coordinates, as it requires visiting all interactions features. Hence, we observe that computing duality gaps must be done parsimoniously, as in Figure 3.1. Furthermore, as CELER [Massias et al., 2018, 2020] relies strongly on creating dual variables, a modification of the structure of this working set algorithm is hence required. Moreover, to avoid computing duality gaps, we develop an active set

algorithm which differentiates main and interactions features (section 3.2.2) and leverages the sparsity induced by the Soft-Thresholding operator to build our active set. Lastly, in section 3.3 we adapt Anderson acceleration, a non-linear acceleration which has recently gotten a surge of interest to accelerate LASSO type solvers.

3.1 Duality gap of Elastic Net with Interactions

Since LASSO and Elastic Net are not differentiable, multiple stopping criteria have been considered in the literature. A classical stopping criterion is the duality gap [Kim et al., 2007, Massias et al., 2018] which is the difference between the value of the original minimization problem (also named primal problem), and a maximization problem, called the dual problem. Morevoer, strong duality holds for LASSO (see Massias et al. [2018]) and hence at the optimum, the difference between the primal and dual problems, *i.e.*, the duality gap, vanishes. We detail below the (dual) maximization problem.

Proposition 3.1.1. *To the Elastic Net with Interactions minimization problem $\mathcal{P}(\beta, \Theta)$ (eq. (2.7)), is associated a maximization problem, called dual problem $\mathcal{D}(\nu)$:*

$$\widehat{\nu}^{(t)} = \underset{\nu \in \Delta_{X,Z}}{\arg\max} \left(\frac{1}{2n} \|y\|_2^2 - \frac{n\alpha^2}{2} \left\| \nu - \frac{y}{\alpha n} \right\|_2^2 - \left(\frac{n\alpha}{c_\alpha^{(t)}} \right)^2 \left(\frac{\alpha_{2,1}}{2} \left\| \beta^{(t)} \right\|_2^2 + \frac{\alpha_{2,2}}{2} \left\| \Theta^{(t)} \right\|_2^2 \right) \right) , \tag{3.1}$$

$$\begin{aligned} \text{with } \alpha =& \frac{1}{4}(\alpha_{1,1} + \alpha_{1,2} + \alpha_{2,1} + \alpha_{2,2}) \ , \\ c_\alpha^{(t)} =& \alpha \max \left(n, \frac{\|X^\top r - n\alpha_{2,1}\beta^{(t)}\|_\infty}{\alpha_{1,1}}, \frac{\|Z^\top r - n\alpha_{2,2}\Theta^{(t)}\|_\infty}{\alpha_{1,2}} \right) \ , \\ \Delta_{X,Z} =& \left\{ \nu \in \mathbb{R}^n : \max \left(\frac{\|X^\top \nu - \frac{n\alpha_{2,1}}{c_\alpha^{(t)}}\beta^{(t)}\|_\infty}{\alpha_{1,1}}, \frac{\|Z^\top \nu - \frac{n\alpha_{2,2}}{c_\alpha^{(t)}}\Theta^{(t)}\|_\infty}{\alpha_{1,2}} \right) \leqslant \frac{1}{\alpha} \right\} \ . \end{aligned}$$

A canonical dual variable $\widehat{\nu}$ is the rescaled residuals [Mairal, 2010], defined as follows:

$$\widehat{\nu}^{(t)} = \frac{r}{c_\alpha^{(t)}} = \frac{y - X\widehat{\beta}^{(t)} - Z\widehat{\Theta}^{(t)}}{c_\alpha^{(t)}} \ . \tag{3.2}$$

Proof. We obtain Elastic Net with Interactions dual problem from the dual lasso formulation (see for example [Massias et al., 2018]). We first add interaction terms in the original problem while in a second step, we use the fact that Elastic Net is a LASSO problem with increased matrix and signal (see [Zou and Hastie, 2005]). We provide the full proof in section 6.2 of appendix (chapter 6). □

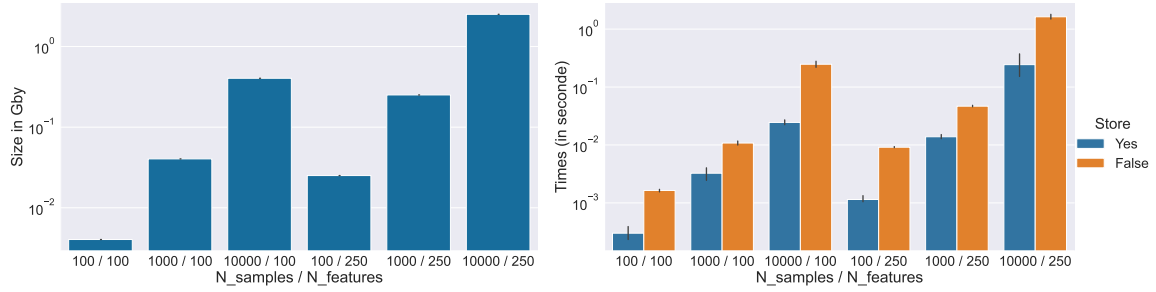
Hence, Elastic Net being a reformulation of a LASSO problem, strong duality holds, and the duality gap provides an upper bound converging to zero as for Elastic Net.

Proposition 3.1.2. *Being $\mathcal{P}(\beta^{(t)}, \Theta^{(t)})$ the objective value of the minimization problem after t epochs and being $\mathcal{D}(\nu^{(t)})$ the value of the maximization problem, with the dual variable $\nu^{(t)}$ computed from rescaled residuals. Then, the duality gap $\mathcal{G}(\beta^{(t)}, \Theta^{(t)})$ is:*

$$\mathcal{P}(\beta^{(t)}, \Theta^{(t)}) - \mathcal{P}(\beta^{\odot}, \Theta^{\odot}) \leqslant \mathcal{P}(\beta^{(t)}, \Theta^{(t)}) - \mathcal{D}(\nu^{(t)}) = \mathcal{G}(\beta^{(t)}, \Theta^{(t)}) \ ,$$

where $\beta^{\odot}$ and $\Theta^{\odot}$ are the optimal values of main and interactions coefficients.

Unfortunately, even if this criterion enjoys good theoretical properties, the rescaling constant $c_{\alpha}^{(t)}$ requires to compute $Z^{\top}\nu^{(t)}$ whose cost is quickly prohibitive, as in Figure 3.1.



(a) Memory Cost for Z (b) Time Cost

Figure 3.1: Cost of computing the dual variable for different number of samples and features. The left part illustrates the memory cost associated to store Z (interactions design matrix); the right part shows the time budget needed to evaluate $Z^{\top}y$ with stored interaction matrix (blue) or without (orange), thanks to `Numba` [Lam et al., 2015]). For a dataset with $n = 10\,000$ observations and $p = 100$, computing *on-the-fly* requires more than 0.1 s and more than 1 s for $p = 250$. For our genomics dataset, the cost is between 0.1 s and 1 s, limiting dual gap evaluations drastically.

Figure 3.1 illustrates the memory cost to store Z on the left part, while the right part indicates in orange the computational time to evaluate *on-the-fly* $Z^{\top}y \in \mathbb{R}^{q}$ and in blue the same quantity with Z stored in memory. As expected, computing *on-the-fly* increases computational burden. For our genomics dataset, the evaluation takes between 0.1 s and 1 s, which limits the use of duality gap as a stopping criterion on the whole dataset. However, we observe that restricted to a small subset of features, for the same number of observations, computing $Z^{\top}y$ is faster. Hence, even if the duality gap cost is prohibitive for the whole problem, its computation is fast enough to test sub-problem convergence for (relatively small) inner problems.

3.2 Active set for coordinate descent

One challenge of building active sets in our context is to reduce as much as possible the number of duality gap evaluations for the full problem, a key difference between CELER implementation and ours. Indeed, between two active sets, CELER computes the duality gap and uses the new dual point to update the feature "ranking" and create the active set. We detail in the following a direct and naive application of CELER to handle interactions and then describe our proposed approach.

Limitations of a naive adaptation of CELER. With a duality gap and associated dual features computation, CELER is able to build precise priority rules. However, updating the dual variable costs one pass over the dataset, and updating priorities costs another one, which in interactions settings is prohibitive. Hence, we restrict our use of the duality gap between two active sets, by using a heuristic stopping criterion as a first criterion before computing a duality gap. Moreover, a drawback to apply CELER directly it does not differentiate main and interaction features, whereas the former has a much lower computational cost than the latter. In particular, although it probably rarely happens, having to find and estimate all the important interaction variables but having to continue because the active set of main effects has missed an important main effect results in unnecessary extra computational cost.

Hence, a first challenge is to intensively use heuristic criterion to avoid computing prohibitive duality gaps on the full problem. For this purpose, we develop two heuristic stopping criteria in section 3.2.3, which if satisfied are complemented by an evaluation of the duality gap on the whole problem. Otherwise the algorithm continues, without evaluating unnecessarily the duality gap on the complete problem. We first derive in section 3.2.1 the CELER ranking rules for Elastic Net with Interactions and discuss the choice to approximate it and to differentiate the way main and interactions effects are handled. We detail in section 3.2.2 how we use the Soft-Thresholding operator to refine active sets. Lastly, we also detail in section 3.2.2 how differentiation affects growth strategies of main and interactions of actives sets.

3.2.1 Ranking rules for Elastic Net with Interactions

The first step is to derive the Elastic Net with Interactions priority rules from the CELER ones (from LASSO) to rank each possible main and interaction features.

Proposition 3.2.1 (Celer for Elastic Net with Interactions)**.** *Let $\widehat{\nu}$ a dual feasible point of the dual problem and c_α the associated constant (to rescale residuals), as in Proposition 3.1.1. We get the following d_j and d_{jj} priority rules, for the main and interactions*

effects respectively:

$$d_j(\widehat{\nu}) = \frac{1 - \left| x_j^\top \widehat{\nu} - \frac{\alpha_{1,2} n}{c_\alpha} \beta_j \right|}{\sqrt{\|x_j\|_2^2 + \alpha_{1,2} n}} \text{ and } d_{jj}(\widehat{\nu}) = \frac{1 - \left| z_{jj}^\top \widehat{\nu} - \frac{\alpha_{2,2} n}{c_\alpha} \Theta_{jj} \right|}{\sqrt{\|z_{jj}\|_2^2 + \alpha_{2,2} n}} \ . \tag{3.3}$$

Proof. We obtain these rules applying to the CELER priorities rules Equation (1.22) adapting the duality gap computation proof for Elastic Net with Interactions (Proposition 3.1.1). Again, we provide the full proof in section 6.3 of appendix (chapter 6). □

These rules allow us to rank the main effects thanks to d_j and the quadratic effects thanks to d_{jj}. A naive adaptation of active set algorithms to the interactions setting will lead to take the $p^{(0)}$ best ranked main effects and $q^{(0)}$ first for the quadratics ones, to have two first *working set* of potential features: $\mathcal{W}_p^{0)}$ for first order and $\mathcal{W}_q^{(0)}$ for second. These two lists are then concatenated to solve the inner problem restricted to the union of these two sets. As explained before, in interactions settings, it can happen that $\mathcal{W}_q^{(t)}$ contains all quadratics relevant features, while $\mathcal{W}_p^{(t)}$ misses some main features. These cases lead to unnecessary growth of $\mathcal{W}_q^{(t)}$ which leads to unnecessary updates and increases computational time since the bottleneck is visiting the interactions, and not main features.

Therefore, we decide to not apply these ranking rules on the main features and consider $\mathcal{W}_p^{(t)} = [\![1, p]\!]$. Hence, we only rank interactions features, as summarized in Algorithm 4. The priority rule must be computed for each interaction, and requires an evaluation of dual variable and the associated rescaled constant, which are costly to compute. Hence, to reduce the computational cost associated to evaluating $d_{jj}^{\alpha_k}$, we approximate it using the dual variable $\widehat{\nu}^{\alpha_{k-1}}$ and $c_{\alpha_{k-1}}$ obtained for a previous set of hyperparameters. However, these two variables are not available when the algorithm is first used, *i.e.,* for the first hyperparameter considered, so we decide to initialize them to $c_{\alpha_0} = \alpha n$ and then $\widehat{\nu}^{\alpha_0} = \frac{y}{c_{\alpha_0}}$.

Lastly, since computing the priority rule list d_{jj} for $jj = \{1, \dots, q\}$ is equivalent to a pass on all interactions, we decide to compute it only once at the initialization of the algorithm and then to freeze it for a fixed α_k, except if it is the first hyperparameter considered. In this particular case, if the evaluation of the duality gap on the complete problem does not stop the algorithm, it is then allowed to recompute once the priority rule.

So, with this strategy, the main features working set $\mathcal{W}_p^{(t)} = [\![1, p]\!]$ contain all features, while interactions features working set $\mathcal{W}_q^{(t)}$ can be highly inaccurate and contain irrelevant features, depending on how close α_{k-1} and α_k are. Consequently, in the following, we describe a second strategy to refine working sets to build our final active set.

Algorithm 4: Compute ranking criterion

Input : $X = [x_1, \dots, x_p]$, $\nu^{(t)}$, $c_\alpha^{(t)}$, $\alpha = (\alpha_{1,1},\ \alpha_{1,2},\ \alpha_{2,1},\ \alpha_{2,2})$

for $jj = 1, \dots, q$ **do**

$j_1, j_2 = \mathcal{R}_{jj}$ `// Rosetta transforms index of Z in double indices`

$z_{jj} = x_{j_1} \odot x_{j_2}$

$d_{jj} = \left(1 - \left|z_{jj}^\top \nu^{(t)} - \frac{\alpha_{2,2} n}{c_\alpha^{(t)}} \widehat{\Theta}_{jj}\right|\right) \Big/ \sqrt{\|(z_{jj})\|_2^2 + \alpha_{2,2} n}$

Output : d_{jj}

3.2.2 Active sets definition and growth strategies

Active set definition. To reduce the amount of interactions visited, and to avoid visiting all main features, we perform a second ranking step, using the Soft-Thresholding ability to zero out coefficients.

In detail, we perform a single pass of Algorithm 2 (detailed in section 2.1.2), over all the main features, *i.e.,* over $\mathcal{W}_p^{(t)}$ and over the interactions working set $\mathcal{W}_q^{(t)}$ built previously. Then, we create the active sets $\mathcal{A}_\beta^{(t)}$ and $\mathcal{A}_\Theta^{(t)}$ that correspond to the current support of main and interaction features, *i.e.,* $\mathcal{A}_\beta^{(t)}$ and $\mathcal{A}_\Theta^{(t)}$ is constituted from the set of non-zeroed main and interaction features. This allows to take into account the fact that the list of d_{jj}'s is approximated by avoiding wasting time optimizing variables in the working set $\mathcal{W}_q^{(t)}$ that are finally found to be zeroed.

Growth strategies. We then discuss strategies controlling the growth of working sets between two iterations. We start by discussing the two growth strategies proposed by CELER. The size of the new working set $\mathcal{W}_q^{(t)}$ is,

- either doubled, *i.e.,* the number of variables considered in $\mathcal{W}_q^{(t)}$ is twice that of the previous working set $\mathcal{W}_q^{(t-1)}$;
- or increased by taking into account the number of active variables in $\mathcal{W}_q^{(t-1)}$, *i.e.,* the number of variables considered in $\mathcal{W}_q^{(t)}$ is twice the support of $\mathcal{W}_q^{(t-1)}$. This second strategy is called pruning.

The first strategy doubles the size of the working set, which is advantageous to detect the relevant variables, especially during the first iterations, but can lead to consider sets too large compared to the true support when most of the variables have already been identified, implying an unnecessary computational overload. Alternatively, the pruning strategy allows refining by growing slower, and even to reduce the size of the working set if the initial size is too large.

Problem with these strategies in the case of interactions. Although these two strategies are efficient, they depend heavily on the priority rule, which is prohibitively expensive to update in the case of interactions, unlike CELER where it is done between each working set. Also, if the sorting rule has wrongly classified some active variables, the first strategy can lead from one iteration to another to consider a very large set without detecting any new active variables. Conversely, with the pruning rule in our case, it is possible that the algorithm stops growing by failing to detect some misclassified variables that should be active.

Our approach. Thus, the adaptation of CELER to the case of interactions requires to define an adapted strategy, which allows to quickly increase the size of the working set, in particular during the first iterations, and then to do it more moderately once most of the active variables have been identified. To do so, we propose to determine the largest rank $q^{(t)}$ of the active variables of the working set and to grow according to this rank $q^{(t)}$ rather than the size of the working set or its support.

This rank $q^{(t)}$ is necessarily lower than the size of the working set and higher than the size of its support. Thus, if this rank $q^{(t)}$ is close to the size of the working set $\mathcal{W}_q^{(t-1)}$, the growth corresponds to the first strategy, which is useful during the first iterations. Conversely, if this rank $q^{(t)}$ is close to the size of the support, the growth corresponds to the pruning strategy, which is useful to control their growth and allow them to shrink if necessary.

Finally, if an iteration has not identified any additional active variable and the stopping heuristics (as follows in section 3.2.3) are satisfied, the complete duality gap is calculated. If the latter completes the desired tolerance, the algorithm stops, otherwise, we propose to perform a pass on each of the possible interactions, to detect those to be added to the working set and update the rank $q^{(t)}$.

Algorithm 5 summarizes active sets definitions and growth strategies.

Algorithm 5: Growing strategy and active set construction

Input : $X,\ y,\ \beta^{(t)},\ \Theta^{(t)},\ \alpha = (\alpha_{1,1},\ \alpha_{1,2},\ \alpha_{2,1},\ \alpha_{2,2}),\ q^{(t)}$

$\mathcal{W}_q^{(t)} = \left\{jj \in [p] : d_{jj} \text{ among } 2q^{(t)} \text{ smallest values of } d_{jj}\right\}$ // Update $\mathcal{W}_q^{(t)}$

Get $\beta^{(t)}, \Theta^{(t)}$ with a solver applied to $\left(\beta^{(t)}, \Theta^{(t)}, [\![1:p]\!], \mathcal{W}_q^{(t)}, \texttt{itr} = 1\right)$ // 1 pass

$\mathcal{A}_\beta^{(t)}, \mathcal{A}_\Theta^{(t)} = \left\{j \in [\![1:p]\!] : \beta_j^{(t)} \neq 0, jj \in \mathcal{W}_q^{(t)} : \Theta_{jj}^{(t)} \neq 0\right\}$ // Update $\mathcal{A}_\beta^{(t)}, \mathcal{A}_\Theta^{(t)}$

Output : $\mathcal{A}_\beta^{(t)}, \mathcal{A}_\Theta^{(t)}$

3.2.3 Avoid computing duality gap

We now detail how to control the optimization progress between two active sets, without having to compute the duality gap on the whole features. With this objective in mind, we derive two naive stopping criteria: the absolute difference between the current and the previous iterate on consecutive actives sets and the difference between the current and previous objective function (eq. (3.4)).

$$\max\left(\left\|\beta^{(t+1)}-\beta^{(t)}\right\|_\infty, \left\|\Theta^{(t+1)}-\Theta^{(t)}\right\|_\infty\right) \leqslant \epsilon \quad \text{and} \quad \left|\text{obj}^{(t)}-\text{obj}^{(t+1)}\right| \leqslant \epsilon \ . \tag{3.4}$$

In more details, after solving the sub problem associated to $\mathcal{A}_\beta^{(t)}$ and $\mathcal{A}_\Theta^{(t)}$, we perform a single pass on $\mathcal{W}_p^{(t+1)}$ and $\mathcal{W}_q^{(t+1)}$, to evaluate these criteria. Hence, if an added feature is different from zero and appears in $\mathcal{W}_q^{(t+1)}$ or if the difference between two consecutive objective values is too important, the algorithm continues; otherwise the (full) duality gap is computed.

3.3 Inner solver with Anderson extrapolation

While working set or active set as well as screening exploit the sparsity of the LASSO solution to improve numerical performances, more generic optimization tools as inertial or non-linear acceleration also permit obtaining additional speed up. The former, also known as Nesterov Acceleration [Nesterov, 1983] has been widely studied and has notably lead to Fast Iterative Soft-Tresholding Algoritm, abbreviated FISTA [Beck and Teboulle, 2009].

We focus on Anderson's acceleration which has recently received a surge of interest in modern machine learning optimization problems. Consider a fixed point iteration problem, as follows:

$$\beta^{(t+1)} = T\beta^{(t)} + b \ , \tag{3.5}$$

where $T \in \mathbb{R}^{p\times p}$ represents an iteration matrix with a spectral radius $\rho(T) < 1$. Anderson acceleration [Anderson, 1965] aims to improve convergence of such problems by extrapolating a new point β_E from a linear combination of previous iterates $\beta^{(t)}$:

$$\beta_E = \sum_{t=0}^{T} c_t \beta^{(t)} \ , \tag{3.6}$$

where $(c_t)_{0\leqslant t\leqslant T}$ is a sequence of weights, which leads to lower objective function.

Anderson acceleration for proximal coordinate descent. Main results are provided to solve quadratic problem with symmetric iteration matrix T [Scieur et al., 2016]. Moreover, it can be shown that gradient descent iterates [Bertrand and Massias, 2021] can be written as a fixed point iteration problem with a such symmetric matrix. Since proximal gradient descent is widely used in machine learning problems, recent works have adapted Anderson acceleration [Zhang et al., 2020, Mai and Johansson, 2020] to it. In addition, proximal coordinate descent algorithm is known to have much better results than proximal gradient descent on numerous machine learning problem, unfortunately, Bertrand and Massias [2021] shown that the iterates of this algorithm can be written as a fixed point iteration problem, but with a non-symmetric matrix. Hence, to ensure convergence of Anderson acceleration, they propose to check that the extrapolated point achieves a lower objective than the current iterate. In particular, regarding proximal coordinate descent algorithm, they prove that asymptotically, the iterates of the algorithm follow noisy linear iterations, which allow applying Anderson acceleration to such problem, and hence allow to perform such acceleration on proximal coordinate descent [Bertrand and Massias, 2021], recently coupled with active set [Bertrand et al., 2022].

Anderson acceleration for Elastic Net with Interactions. Bertrand and Massias [2021] provide two versions, the offline and online. The former uses a linear combination of all the iterates, while the online version only requires the K last iterates. We rely on the latter, since it needs to store less coefficients and limit the memory burden in interactions settings. Moreover, following their recommandation, we choose $K = 5$ and compute the weight sequence from $c = \left(U^\top U\right)^{-1} \mathbb{1}_K / \mathbb{1}_K^\top \left(U^\top U\right)^{-1} \mathbb{1}_K$, where U is the matrix of iterates difference, as Line 5 of Algorithm 6.

Algorithm 6: Proximal Coordinate Descent with Anderson Extrapolation

Input : $X,\ y,\ \beta^{(0)},\ \Theta^{(0)},\ \alpha = (\alpha_{1,1}, \alpha_{2,1}, \alpha_{1,2}, \alpha_{2,2}), \nu^{(0)}, \mathcal{A}_\beta^{(0)},\ \mathcal{A}_\Theta^{(0)},\ q^{(0)}, \mathcal{R}$
param. : $\epsilon = 10^{-6},\ \texttt{itr} = 10^4,\ \mathcal{R},\ K = 5$

1 **for** $t = 1, \ldots, T$ **do**
2 Get $\beta^{(t)},\ \Theta^{(t)}$ with Algorithm 2 `// Make one pass on actives sets`
3 **if** $t = 0 \mod K$ **then** `// Do Anderson acceleration each K epoch`
4 Compute $\text{obj}^{(t)}$ with Equation (2.7)
5 $U = \left([\beta, \Theta]^{(t-K+1)} - [\beta, \Theta]^{(t-K)}, \ldots, [\beta, \Theta]^{(t)} - [\beta, \Theta]^{(t-1)} \right)$
6 $c = \left(U^\top U\right)^{-1} \mathbb{1}_K / \mathbb{1}_K^\top \left(U^\top U\right)^{-1} \mathbb{1}_K$ `// Compute extrapolation weight`
7 $\beta_A, \Theta_A = \sum_{k=1}^K c_k \beta^{(t-K+k)}, \sum_{k=1}^K c_k \Theta^{(t-K+k)}$ `// Compute extrapolation`
8 Compute $\text{obj}_A^{(t)}$ with Equation (2.7) `// Compute objective of extrapolate`
9 **if** $\text{obj}_A^{(t)} \leqslant \text{obj}^{(t)}$ **then** `// Update coefficients only if objective is lower`
10 $\beta^{(t)}, \Theta^{(t)} = \beta_A, \Theta_A$

Output : $\beta^{(t)}, \Theta^{(t)}$

3.4 Summary and benchmark with Benchopt

We first provide a short summary of our active set algorithm with Anderson extrapolation, while the second part presents Benchopt [Moreau et al., 2022] which allows comparing different solvers for a given machine learning problem on different data. Then, the two last parts illustrate on a moderate scale (section 3.4.3) and then on a higher scale (section 3.4.4) comparisons between LASSO solvers. We compare different versions of our own approach but also use two others Python implementation: CELER (active set with dual extrapolation) and scikit-learn (classical proximal coordinate descent algorithm). With regard to our approach, we compare coordinate descent algorithm with and without Anderson acceleration, and with and without active set scheme.

3.4.1 Summary: double active set coordinate descent

To summarize, we build a two-step active set algorithm whose key idea is to avoid visiting all the interaction features. To reduce the computational cost to estimate the main and interaction coefficients, we rely on approximate priority rules and soft-thresholding ability to zeroed coefficients. On the another hand, we intensively use heuristic stopping criteria to avoid computing duality gap of the full problem and thus, reduce stopping criterion cost. Lastly, we use proximal coordinate descent with Anderson acceleration (Algorithm 6) as inner solver of the active set. This leads to Algorithm 7 that we implement in Python using `Numba` [Lam et al., 2015] *just-in-time* compiler to accelerate computing intensive parts.

3.4.2 Benchopt adaptation to quadratic problems

We rely on Benchopt to measure the computational cost of the different solvers. Benchopt CELER and scikit-learn are not initially build to taking account interactions. A simple adjustment for Benchopt is to replace the design matrix of main effects by the main and interactions matrix if it can be stored in memory. Regarding CELER and scikit-learn solvers, we also compute the interaction design matrix and then store it to feed the solvers. However, to perform fair comparisons with our approach, we include the time to build and store the quadratic matrix associated to each solver. Moreover, as the standardization or normalization process can differ from one solver to another, we only consider settings where the intercept term is considered, *i.e.,* we subtract to each column its mean, without rescaling it. In each setting, we will consider three penalties levels $\alpha \in \alpha_{\max} \times \{0.1, 0.01, 0.001\}$, to evaluate performance on different parts of the LASSO path.

Algorithm 7: Double Active-Set coordinate descent

Input : $X,\ y,\ \beta^{(0)},\ \Theta^{(0)},\ \alpha = (\alpha_{1,1},\ \alpha_{1,2},\ \alpha_{2,1},\ \alpha_{2,2}), \nu^{(0)}, \mathcal{A}_\beta^{(0)},\ \mathcal{A}_\Theta^{(0)},\ q^{(0)}$

Init : $\epsilon = 10^{-6},\ \texttt{itr} = 10^4$.

Compute $\text{obj}^{(1)}$ with Equation (2.7) applied to $(\beta^{(0)}, \Theta^{(0)},\ \alpha)$

Get $\beta^{(1)}, \Theta^{(1)}$ with a solver applied to $(\beta^{(0)}, \Theta^{(0)}, \mathcal{A}_\beta^{(0)}, \mathcal{A}_\Theta^{(0)}, \texttt{tol} = \epsilon)$ `// WarmStart`

Get d_{jj} with Algorithm 4 applied to $(X, \Theta^{(1)}, \nu^{(1)}, c_\alpha^{(1)}, \alpha)$ `// Rank interactions`

Get $\mathcal{A}_\beta^{(1)}, \mathcal{A}_\Theta^{(1)}$ with Algorithm 5 applied to $(X,\ y,\ \beta^{(0)},\ \Theta^{(0)}, \alpha, \mathcal{A}_\beta^{(0)},\ \mathcal{A}_\Theta^{(0)},\ q^{(0)})$

for $t = 1, \dots, T$ **do** `// Until stop, solve sub-problem and update` $\mathcal{A}_\beta^{(t)}$ `and` $\mathcal{A}_\Theta^{(t)}$

  Get $\beta^{(t+1)}, \Theta^{(t+1)}$ with a solver applied to $(\beta^{(t)}, \Theta^{(t)}, \mathcal{A}_\beta^{(t)}, \mathcal{A}_\Theta^{(t)}, \texttt{tol} = \epsilon)$

  Update $q^{(t+1)}$, then get $\mathcal{A}_\beta^{(t+1)}, \mathcal{A}_\Theta^{(t+1)}$ from Algorithm 5

  Compute $\text{obj}^{(t+1)}$ with Equation (2.7) applied to $(\beta^{(t)}, \Theta^{(t)},\ \alpha)$

  if $\max(|\beta^{(t+1)} - \beta^{(t)}|, |\Theta^{(t+1)} - \Theta^{(t)}|) \leqslant \epsilon$ ***and*** $(\text{obj}^{(t+1)} - \text{obj}^{(t)}) \leqslant \epsilon$ **then**

    Get $\nu^{(t+1)}$ from Equation (3.2) `// Compute dual variable`

    if $|\mathcal{G}(\beta^{(t+1)}, \Theta^{(t+1)}; \nu^{(t+1)})| \leqslant \epsilon$ **then** `break` `// test dual gap`

    else `// Make one pass on all features and update actives sets`

      Get $\mathcal{A}_\beta^{(t+1)}$,$\mathcal{A}_\Theta^{(t+1)}$ with Algorithm 5 applied to $(X, y, \beta^{(t+1)}$,$\Theta^{(t+1)}, \alpha, q)$

Output : $\beta^{(\cdot)}, \Theta^{(\cdot)}, \nu^{(\cdot)}, \mathcal{A}_\beta^{(\cdot)}, \mathcal{A}_\Theta^{(\cdot)}, q^{(\cdot)}$

Parametrization of algorithms. With respect to scikit-learn and CELER solver, we keep the initial parameters from LASSO Benchopt benchmark. In detail, scikit-learn duality gap stopping criterion is fixed at 0 while number of iteration increase as defined by inner Benchopt code. The CELER solver sets the duality gap tolerance to 10^{-12} and allows one million iterations in the inner solver, while the maximum number of working set iteration increases by one at each Benchopt step. Regarding LASSO with Interactions, we also fix the duality gap tolerance to 10^{-12} in Benchopt but allow only one thousand iterations in the inner solver. However, thanks to our heuristic stopping criterion, if the thousand interactions are not sufficient to solve the subproblem, the algorithm updates the active set if needed and continue the optimization problem, without the costly full duality gap computation. Additionally, we increase the maximum number of active sets iteration five by five between each Benchopt step.

Benchopt Methodology and Limits of the approach. The key idea of Benchopt is to measure the computational cost of a solver for an optimization task. Given a predetermined penalty level, Benchopt measures the computational cost to perform k iterations of a solver, then evaluates primal and dual problem from coefficients returned with such iterations budget and continues for the next iteration budget $k + 1$. Hence,

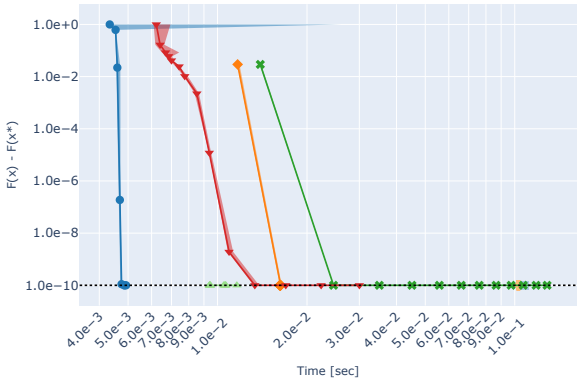
the Benchopt framework allows to easily compare solvers, even when they have different stopping criteria. Nonetheless, it does not yet handle the time required to fit an entire path or the computational performance of a solver which use warm-start from a previous LASSO problem, on which we rely in our active set algorithm.

3.4.3 Moderate scale studies

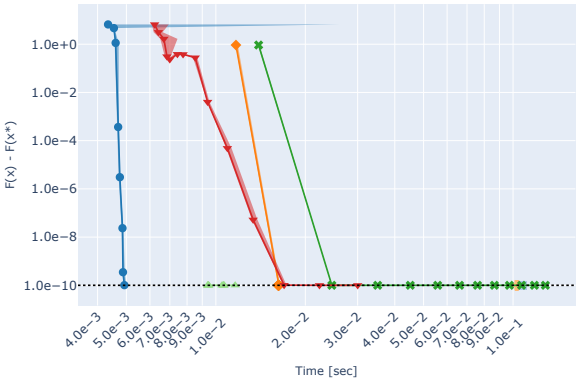
We start by two small problems, to be able to perform benchmark with all solvers. The first one is done with the first hundred columns of the Leukemia dataset [Golub et al., 1999], which is a problem with more features than samples: $n = 72 < p = 100 < q = 5050$. The second benchmark considers the sixty-first column of the genomics dataset, then the problem have fewer features than samples $p = 60 < q = 1830 < n = 16294$.

Leukemia dataset: first 100 features. In this experiment, CELER is the fastest method, in particular on the two first penalties level, while in the last our approach with active set algorithm and which benefit from stored design matrix perform closely. We also remark that duality gap curve of CELER solver stop before others solvers: this is explained by the fact that CELER perform dual extrapolation to build better dual point and hence stop earlier algorithm, as illustrated here. In particular, we observe that LASSO with Interactions without active set achieves immediately the optimal score, which is explained by the fact by this option perform $k \times 10^3$ iteration, on the whole problem. Nevertheless, those two cases are the longest to run, in all settings, even if we observe that Anderson acceleration can accelerate convergence with and without active set schemes. In particular, LASSO with Interactions with active set algorithm appears to be as effective as CELER in the lowest penalties, *i.e.,* ($\alpha = \alpha_{\max} \times 0.001$). Notably, LASSO with Interactions with active set strategy appears to be faster than scikit-learn in the two smallest penalties. Lastly, we observe that evaluating *on-the-fly* the interactions' matrix does not increase too much the running time.

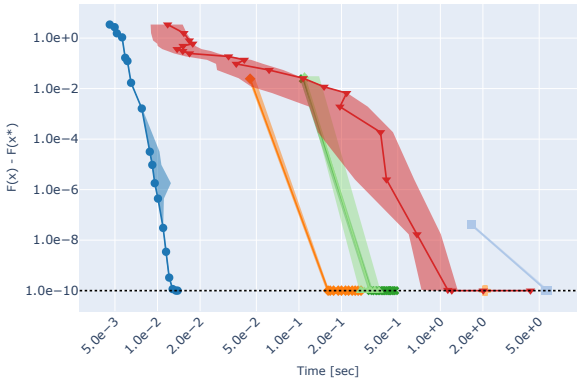
Genomics dataset: first 60 features. For this dataset, methods without active set strategies are the slowest and behave similarly as for the Leukemia dataset. Moreover, the method leveraging *on-the-fly* interaction matrix appears the fastest in the case where $\alpha = \alpha_{\max} \times 0.1$, and works as fast as CELER, compared to scikit-learn and method without active set. Lastly, extrapolation also helps to accelerate convergence in all cases.



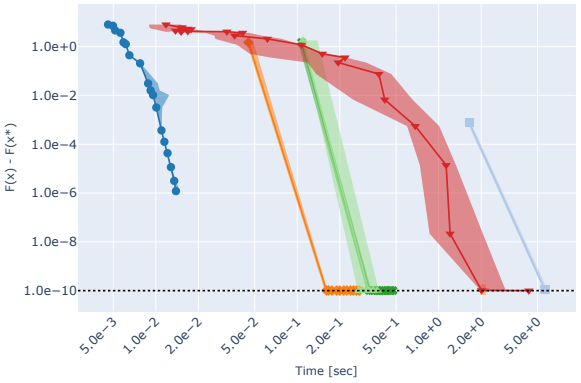
(a) Primal: $\alpha = 0.1 \times \alpha_{\max}$



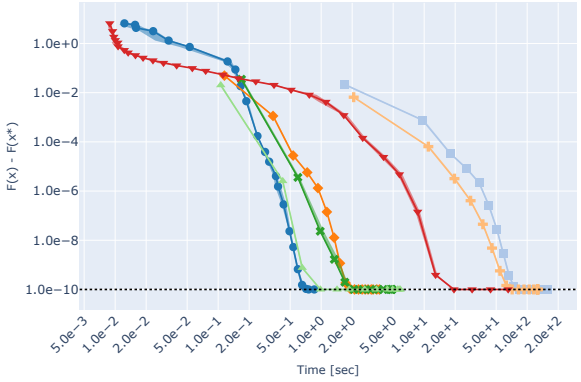
(b) Duality Gap: $\alpha = 0.1 \times \alpha_{\max}$



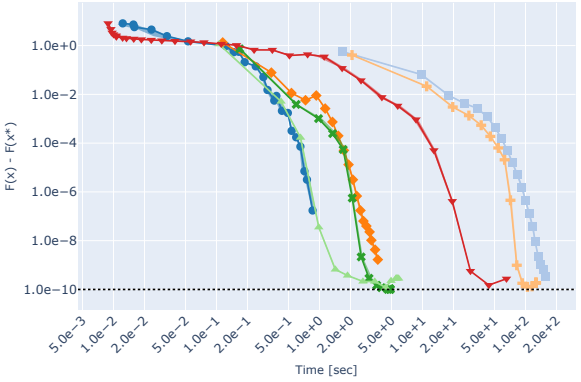
(c) Primal: $\alpha = 0.01 \times \alpha_{\max}$



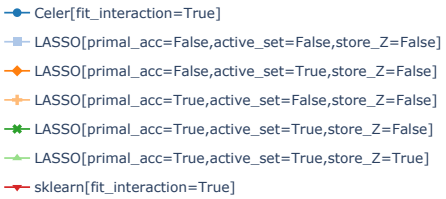
(d) Duality Gap: $\alpha = 0.01 \times \alpha_{\max}$



(e) Primal: $\alpha = 0.001 \times \alpha_{\max}$

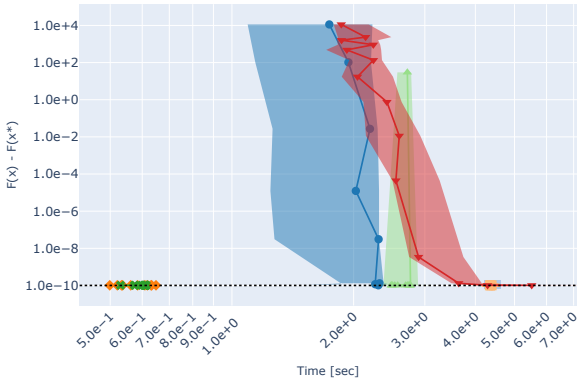


(f) Duality Gap: $\alpha = 0.001 \times \alpha_{\max}$

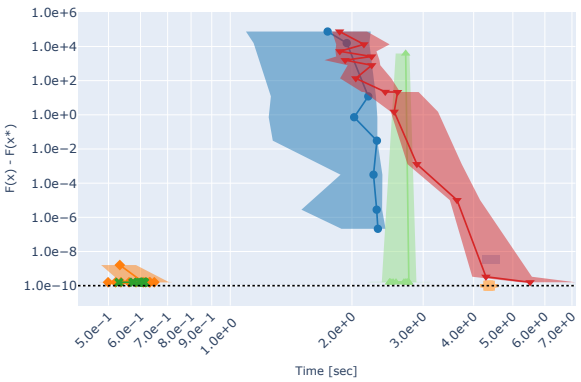


(g) Solvers Legend

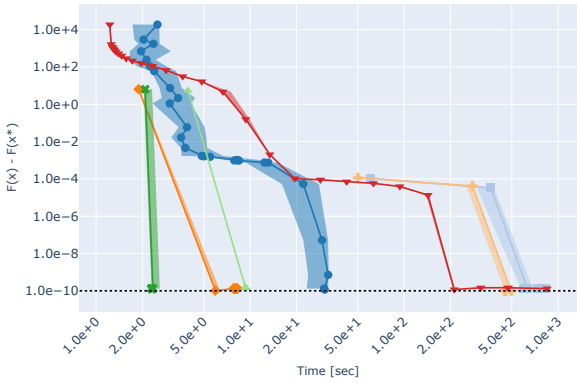
Figure 3.2: Primal convergence on the left and duality gap convergence on the right, with the first hundred columns of Leukemia dataset, leading to $q = 5050$ interactions features, for $n = 72$ samples. CELER is the fastest algorithm while LASSO with Interactions with non-linear acceleration and active set reach similar results for the smallest penalty.



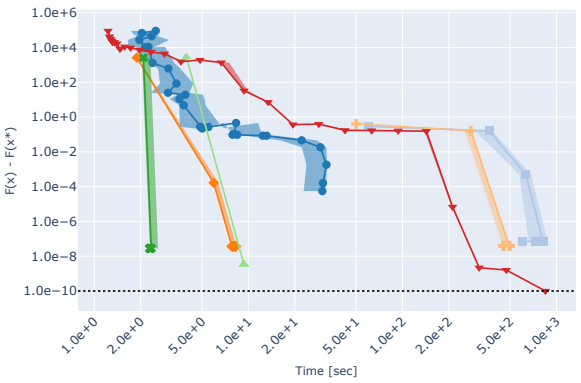
(a) Primal: $\alpha = \alpha_{\max} \times 0.1$



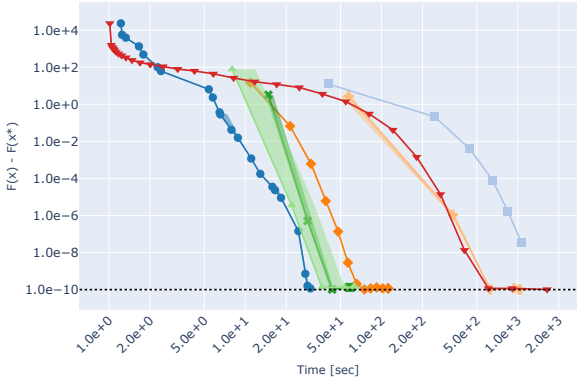
(b) Duality Gap: $\alpha_{\max} \times 0.1$



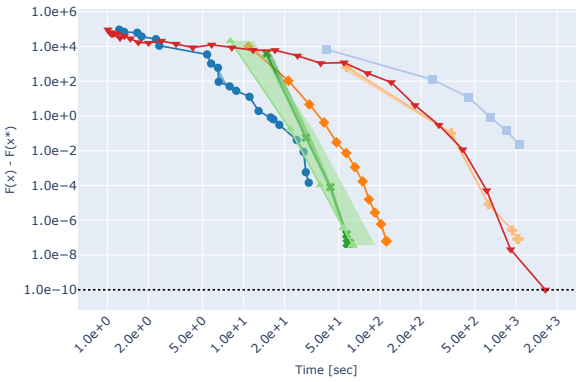
(c) Primal: $\alpha_{\max} \times 0.01$



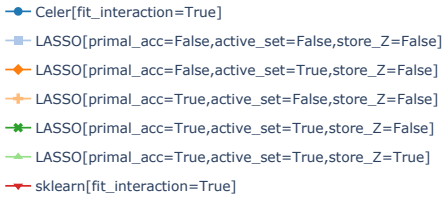
(d) Duality Gap: $\alpha_{\max} \times 0.01$



(e) Primal: $\alpha_{\max} \times 0.001$



(f) Duality Gap: $\alpha_{\max} \times 0.001$



(g) Solvers Legend

Figure 3.3: Primal convergence on the left and duality gap convergence on the right, with the first sixty columns of Genomics dataset, leading to $q = 1830$ interactions, for $n = 16294$ samples. LASSO with Interactions with active set and acceleration are the fastest for the two largest penalties while CELER achieves slightly better in last penalty.

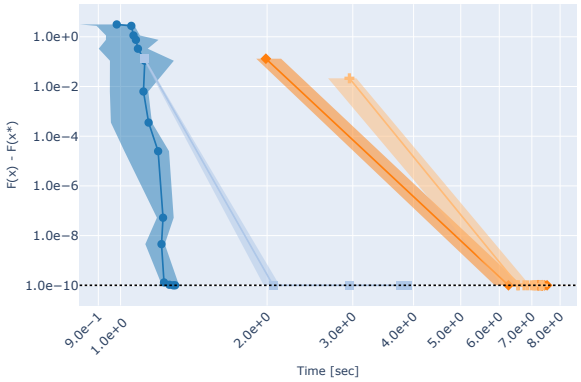
Conclusion of moderate scale studies. From these two benchmarks, we conclude that even with a moderate number of main features, method not using active sets fail to be competitive with the ones leveraging active sets. With regard to Anderson acceleration, while in Leukemia two first α penalties they increase the computational cost, on the last Leukemia penalty while it improves time on Genomics dataset. Lastly, our approach with the optional possibility to store the interaction matrix seems to have similar performance with the version without storage. In particular, it appears that storage helps to accelerate algorithms for Leukemia (Figures 3.2e and 3.2f) while it does not speed up on genomics (Figures 3.3e and 3.3f). For all these reasons, we only keep CELER and LASSO with Interactions with active set, with or without Anderson acceleration, with the possibility of storing the interaction matrix on the setting with non-linear acceleration.

3.4.4 Large scale studies

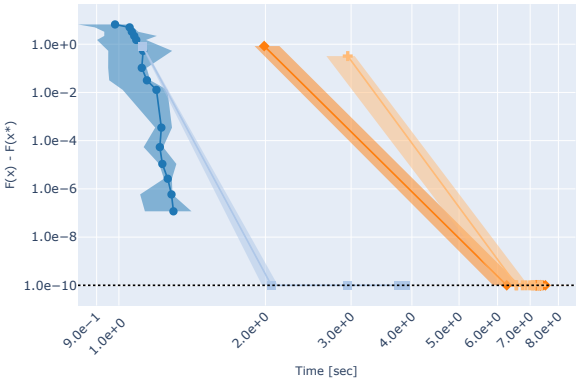
In this last part, comparisons are made on a higher number of features, using first the first thousand columns of Leukemia dataset, leading to almost a half million interactions features, while we secondly perform optimization on the complete genomics dataset, which even if it is a problem with more samples than main and quadratic features ($p = 160 < q = 12880 < n = 16294$), is numerically hard to solve, as we illustrate in Figure 3.5.

Leukemia dataset: first 1000 features. The first thing that we observe is that unlike the first study on Leukemia, our approach which performs Anderson acceleration is the longest for the three penalties, even if we observe that non-linear acceleration permits to reduce the number of active set iterations. Hence, while it reduces the number of iterations again, the associated computational cost of the acceleration is not competitive in this case. Notably, here storing the design matrix leads to the poorest computational performance. Regarding CELER, it obtains the best computational time in the first penalty (Figures 3.4a and 3.4b), a slightly better in the second one (Figures 3.4c and 3.4d) but appears to be longer that our active set without acceleration in the last penalty (Figures 3.4e and 3.4f).

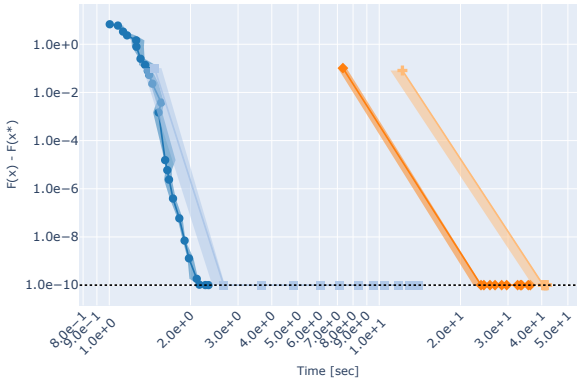
Genomics dataset: whole dataset. For this last benchmark, we observe that non-linear extrapolation helps to achieve the lowest computational time in the two last penalties and notably permits reaching high precision compared to LASSO with Interactions without such acceleration in Figures 3.5e and 3.5f. Moreover, we observe that CELER working set is slower than our approach in the two first penalties, while it is slightly faster than us in the last penalties. Lastly, stored design matrix shows in this benchmark the worst result in all the cases.



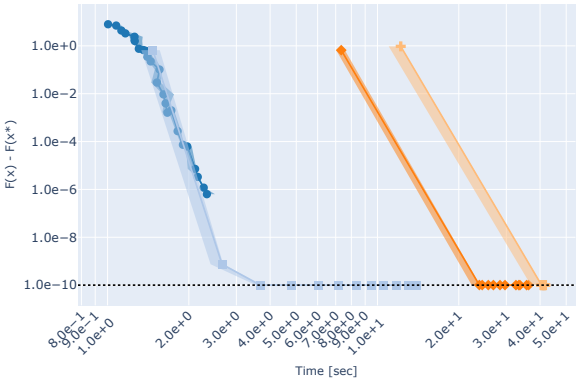
(a) Primal: $\alpha_{\max} \times 0.1$



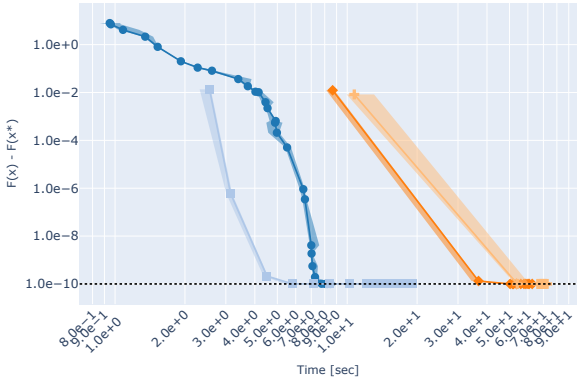
(b) Duality Gap: $\alpha_{\max} \times 0.1$



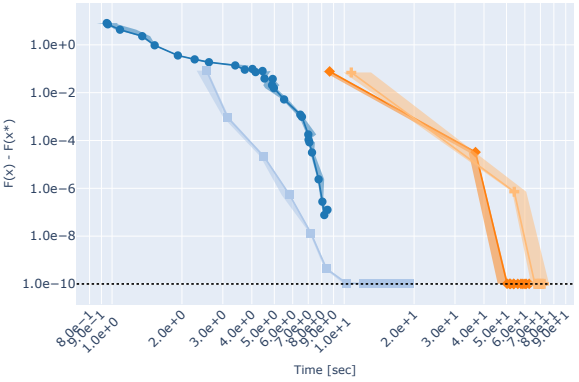
(c) Primal: $\alpha_{\max} \times 0.01$



(d) Duality Gap: $\alpha_{\max} \times 0.01$

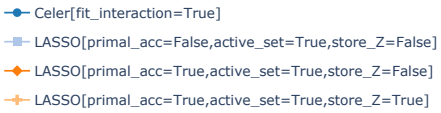


(e) Primal: $\alpha_{\max} \times 0.001$



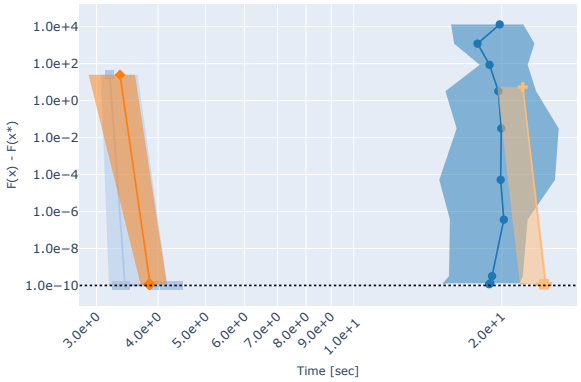
(f) Duality Gap: $\alpha_{\max} \times 0.001$

s

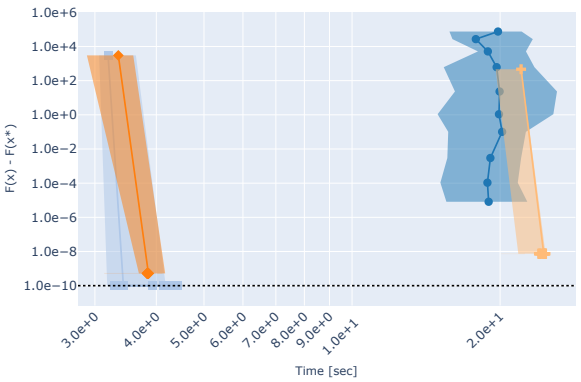


(g) Solvers Legend

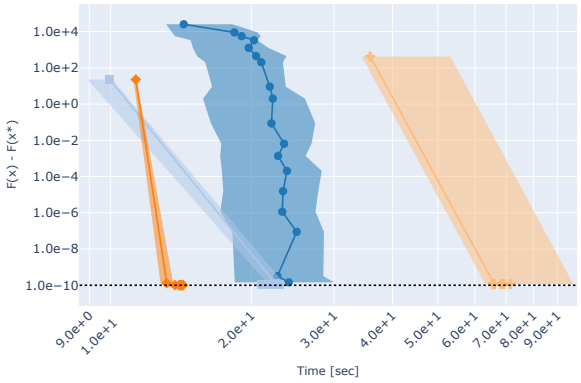
Figure 3.4: Primal convergence on the left and duality gap convergence on the right, with the first thousand columns of Leukemia dataset, leading to $q = 500\,500$ interactions features, for $n = 72$ samples. The first striking thing is that Anderson acceleration costs more computation time than it reduces the number of iterations. Moreover, storing the interactions matrix, in this case slows down solvers. Lastly, CELER appears to be faster on the two first cases, while it takes more time than our active set in the last case.



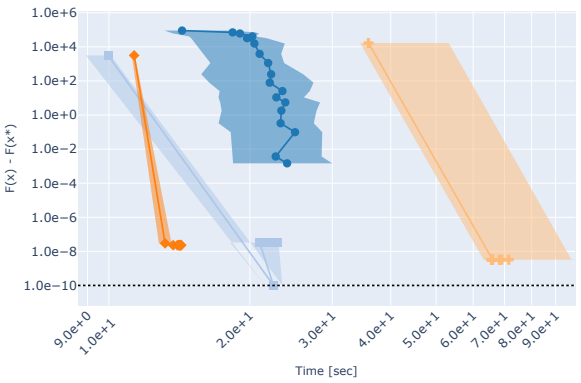
(a) Primal: $\alpha_{\max} \times 0.1$



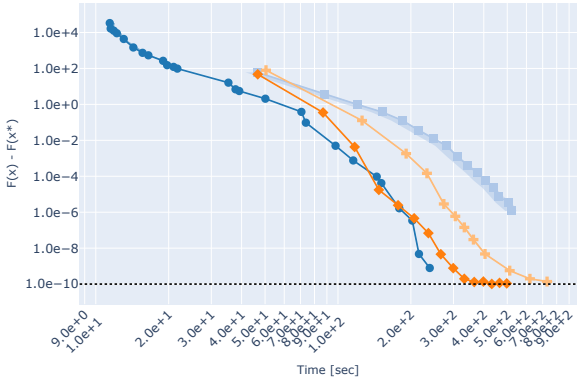
(b) Duality Gap: $\alpha_{\max} \times 0.1$



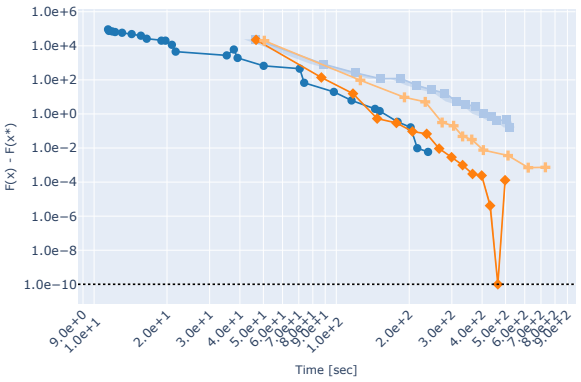
(c) Primal: $\alpha_{\max} \times 0.01$



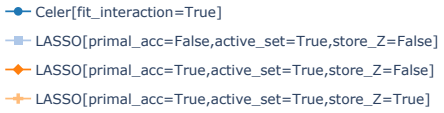
(d) Duality Gap: $\alpha_{\max} \times 0.01$



(e) Primal: $\alpha_{\max} \times 0.001$



(f) Duality Gap: $\alpha_{\max} \times 0.001$



(g) Solvers Legend

Figure 3.5: Primal convergence on the left and duality gap convergence on the right, with the whole dataset, *i.e.*, the one hundred sixty columns of Genomics dataset, leading to $q = 12\ 880$ interactions features, for $n \approx 16\ 294$ samples. Non-linear acceleration improves time in the two last penalties and notably allows performance close to CELER in the last one. Again, storing the interactions matrix slows down active set solvers.

Conclusion of large scale studies. To conclude, we observe that storing the interactions design matrix slows down our solver in all the cases, in addition to needing more memory. We also observe that depending on the dataset, non-linear acceleration can effectively speed up solvers, while depending on the dataset, our approach is faster or slower than CELER. Lastly, we recall that Benchopt can not measure how solvers behave when warm-start *i.e.,* solution of the previous LASSO problem, is available whereas CELER and our active set improve their perfomance thanks to such pipeline.

3.5 Conclusion

In this chapter, we first detailed the dual problem associated with the Elastic Net with Interactions allowing to compute the duality gap, as well as the computational cost of its evaluation. This numerical cost is equivalent to that of a pass over all the variables, since it requires visiting all the interactions. Also, it appears from the first numerical results that the cost of this evaluation is prohibitive, compared to the computational cost of a pass on a subset of variables.

From this observation, we developed an algorithm that does not evaluate the duality gap on the full problem between two iterations, but only heuristic stopping criteria with a negligible computational cost. Also, the developed algorithm allows treating simple effects and interactions on two different scales. Moreover, it uses a flexible growth criterion for the working set of interactions between two iterations, allowing to grow quickly during the first iterations and then more slowly once the majority of the active variables have been identified.

Finally, we also adapted Anderson acceleration to further reduce the computation time of the estimator. Comparisons of the numerical performances of our algorithm with scikit-learn and CELER showed that our algorithm was competitive. In particular, it appeared that storing the interaction matrix, in high dimension, increases the computation time of the working set algorithms compared to those that build it *on-the-fly*. It appeared that although Anderson acceleration reduces the number of working set iterations needed to reach the optimal solution, the associated computational cost may be too large compared to the observed gain. Finally, on the genomic data we are interested in, we observe that the approach combining Anderson acceleration with active set strategy is the fastest of our approach with numerical performances similar to those of CELER.

In the next chapter, we will focus on the statistical performances of our approach, on both simulated and real data, by comparing to state-of-the-art methods that enforce heredity.

Chapter 4

Statisticals Results

Contents

In this chapter, we perform statistical analysis with different state-of-the-art competitors: HierNet, RAMP and CLEAR-Enet with Interactions.

We first evaluate these methods on semi-artificial data, as for CLEAR-Enet with Interactions parameters in Chapter 2.

In a second part, we focus on a real dataset describing genomic features to explain the gene expression regulation. In particular, this real data application is an opportunity to observe the behaviour of two operators defining interaction (product or maximum) and two standardization schemes (STD1 or STD2).

4.1 Semi-Artificial Datasets

We first briefly explain the semi-artificial data generative process and present the results obtained in 3 simulation contexts.

4.1.1 Semi-Generative data process

We follow the semi-generative process detailed in section 2.1.3.1, with the difference that we focus only on the first standardization scenario since RAMP and HierNet do not support estimation with the second standardization scenario (STD2). We build the interactions matrix *on-the-fly* and consider standardized main and interaction effects (STD1 only). The generative heredity scenario process does not change, we continue to explore the five different heredity assumptions but at three different scales, which we will detail in each simulation. Regarding the noise level, we again consider a normal distribution with zero mean and $\sigma \, \mathrm{Id}_{n \times n}$ variance-covariance matrix, where the value of σ is controlled by a signal-to-noise ratio (SNR) fixed at 8. Then we further divide the datasets into training sets (80%) and test sets (20%) and again consider MSE, Recall, Precision, F_1-score but also the number of non-zero coefficients and computational time for each estimator. Finally, in order to average the results, we repeat each simulation setting ten times. We recall hereafter the optimization parameters of CLEAR-Enet with Interactions and detail those used for RAMP and HierNet.

Hyperparameters of estimators. RAMP, HierNet and CLEAR-Enet with Interactions are path methods. Penalty path hyperparameters are set as follows: we consider a grid of $n_\alpha = 100$ points and geometrically distributed between $\alpha_{\max}$ and $\alpha_{\min} = \frac{\alpha_{\max}}{1000}$ *i.e.,* the depth ϵ is set to 0.001. Regarding the estimation of the α hyperparameter, we follow HierNet and use 5-fold cross validation, except for RAMP for which the R package does not provide cross validation. However, the authors of RAMP propose several methods to tune the hyperparameter among which AIC, BIC or Extended BIC [Chen and Chen, 2008]. This last criterion is the one that had the best results in their article, we will also use this criterion. The stopping criterion tolerance of each method is set to 10^{-4}. Regarding CLEAR-Enet with Interactions parameters, based on Chapter 2 results, we consider LASSO case *i.e.,* CLEARLASSO with Interactions, where interaction coefficients are penalized five times more than those of the main effects (*i.e.,* $\kappa = 5$). RAMP offers a similar option in its R implementation, hence we also compare two versions of the method: one that penalizes main and interaction coefficients equally, simply denoted RAMP, and another that penalizes quadratic effects five times more than main effects, denoted RAMP κ . We enforce both strong and weak heredity constraints with RAMP, respectively denoted RAMP-ST and RAMP-WK. For computational reasons, we only consider the weak heredity constraint with HierNet. However, we consider both LASSO and Elastic Net penalties of HierNet, the former is denoted HierNet-ℓ_1 while the latter is simply denoted HierNet. Lastly, we also consider the LASSO and CLEARLASSO versions of our work without interaction features (targeting only the main effects).

4.1.2 Simulation 1: p=30 features and n=325 samples

In our first simulation, we take the same setting as in section 2.1.3, *i.e.,* $p = 30$, $q = 465$ (with pure quadratic terms) and $n = 325$. Also, we recall that in strong, weak and anti-heredity settings, $\|\beta^\star\|_0 = 10$, while $\|\Theta^\star\|_0 \in \{5, 10, 15, 20, 25\}$. In main only setting, we have $\|\beta^\star\|_0 \in \{5, 10, 15, 20, 25\}$ and in interaction only setting $\|\Theta^\star\|_0 \in \{5, 10, 15, 20, 25\}$.

Predictive performances. We first detail the Mean Squared Error (MSE) results shown in Figure 4.1. The first thing we notice is that LASSO with Interactions and its debiased version have similar performance and get the lowest MSE score in all parameters. The second thing that we observe is the poor performance of RAMP-WK in all settings, followed by RAMP-ST in all settings except strong heredity case. While RAMP-WK κ and RAMP-ST κ bring a lower score than RAMP, they do not reach a score as low as HierNet or our approach. In particular, we observe that LASSO version of HierNet outperforms the Elastic Net version, in all sparsity levels and heredity cases. Lastly, when the true coefficients are only main effects, we observe that CLEARLASSO with and without quadratic coefficients behave in the same way. Hence, considering quadratic effects (whereas not needed here) did not deteriorate the results. Our approach is very flexible and succeeded in all settings. We can notice that, without enforcing any heredity assumption, we achieved a better MSE performance even in settings supported by strong and weak heredity.

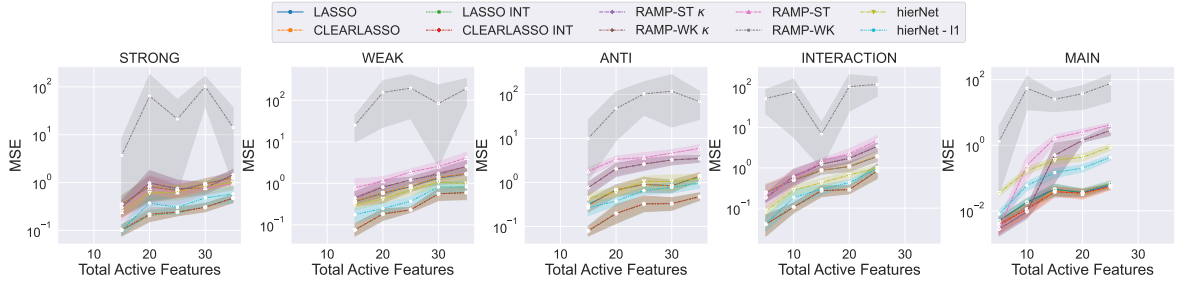


Figure 4.1: Mean Squared Error (MSE) comparisons of CLEARLASSO with Interactions with RAMP and HierNet solution, in the case where $p = 30$, $q = 465$ and $n = 325$. We observe that LASSO with Interactions and CLEARLASSO with Interactions outperform the other methods in all heredity scenarios.

Features selection performances. We now discuss the selection ability performance of the estimators given in Figure 4.2. We observe that both HierNet and RAMP-WK fail as feature selection methods, for the former because of a high number of active features and then a low precision score, while the latter selects a small number of active but irrelevant features, *i.e.,* adds False Positive features except in interaction settings.

Inversely, RAMP-ST succeeds in limiting the number of false positives *i.e.,* reaches a high precision score, notably in strong heredity and main settings, leading to a high F_1-score but as expected, fails to recover relevant features in weak, anti and interaction settings, leading to a lower selection ability in those cases. Moreover, we observe that RAMP-ST κ and RAMP-WK κ which penalize interactions more than main effects, again perform equally while the former enforces strong heredity and the latter weak. These results suggest that penalizing interactions more than main effects is more important than the heredity structure. Both methods bring the highest F_1-score in main and strong heredity settings, and a good F_1-score in weak heredity case. Nonetheless, in interactions and anti heredity settings, even if they fail to add relevant features, they succeed in not adding false positives and finally, reach a good F_1-score. Lastly, our approach obtains the best F_1-score in the main part of weak, anti and interactions heredity settings, while we are close the highest score of RAMP κ in strong and main settings. One thing to notice is that the debiasing step in main settings helps to reduce false positive rate, *i.e.,* achieves a better precision for a similar recall and thus leads to a better F_1-score, even if we get a higher number of active features than RAMP or estimators targeting only main effects.

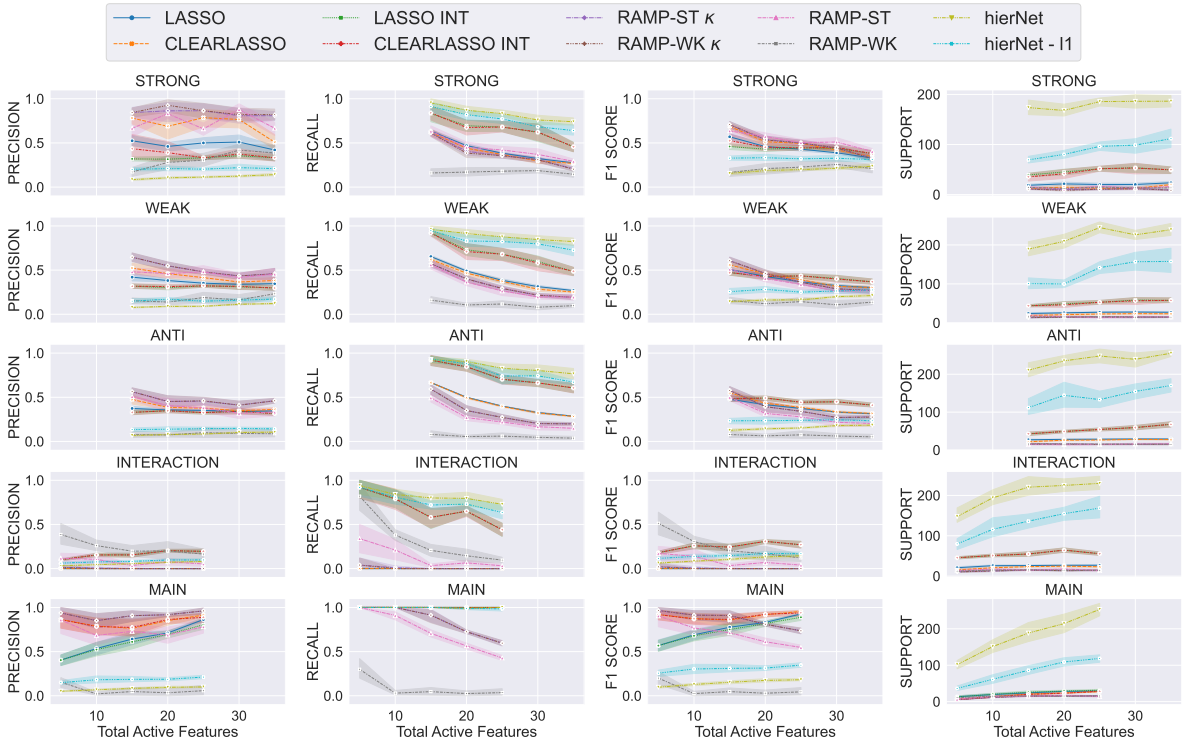


Figure 4.2: Precision, Recall, F_1-score and support comparison between CLEARLASSO with Interactions, HierNet and RAMP, in the case where $p = 30$, $q = 465$ and $n = 325$. A detailed interpretation of the results is given in the paragraph *Features Selection Performances*.

Computational time performances. Computation times measured in seconds are shown in Figure 4.3. We immediately observe that HierNet is not time competitive, partly because the cross-validation step is not parallelized, unlike our approach. Debiasing the LASSO and LASSO with Interactions reasonably increases computational time, and finally, CLEARLASSO with Interactions leads to computational time close to RAMP-ST. The other RAMP methods have similar computational time to LASSO with Interactions while the estimators targeting only main effects are the fasted, as expected. Note that the high computation time of the LASSO estimator in the case of strong heredity is due to the first (and last) compilation by `Numba` of our Python code which is shared between our different estimators, that benefit from this first compilation.

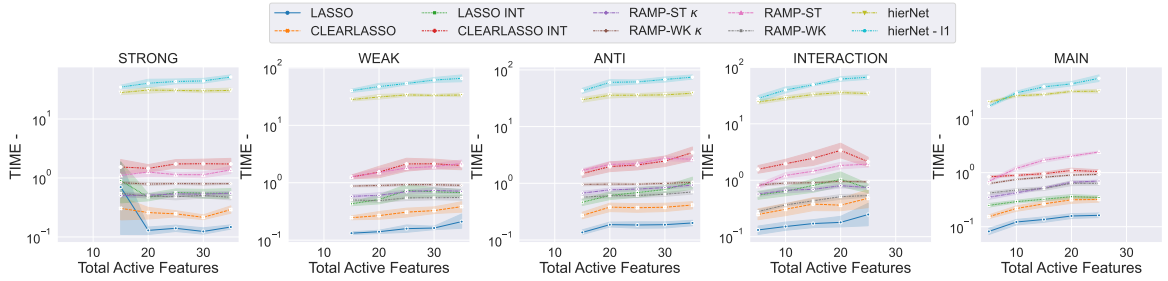


Figure 4.3: Computational time (in second) of CLEARLASSO with Interactions, HierNet and RAMP for hyperparameter selection and final refitting, in the case where $p = 30$, $q = 465$ and $n = 325$. Except for HierNet which is the longest approach, the other estimators have a similar computation time.

Summary of this first simulation. From this first study, it appears in such simulation settings that HierNet achieves good predictive performances, but fails as a feature selection, and its computational cost is too important compared to RAMP or CLEARLASSO with Interactions. Therefore, since the following experiments are on a larger scale, both in terms of number of features and samples, we do not continue to use the HierNet methods. Regarding RAMP method, even if RAMP-WK fails in all settings as predictive or features selection method, RAMP-ST performs better while both RAMP κ versions appear to have good selection ability, in particular in high sparsity level of experiments, for a low computational cost. However, the poor predictive performance of RAMP may be explained in part by the fact that, unlike the EBIC criterion, the cross-validation method optimizes the MSE. Lastly, our approach performs well in all situations and debiasing step significantly improves performances in main effects only.

4.1.3 Simulation 2: p=160 features and n=1629 samples

For this second experiment, we take the following simulation settings: $p = 160$, so $q = 12\ 880$, and $n = 1629$. In strong, weak and anti-heredity settings, $\|\beta^\star\|_0 = 40$, while $\|\Theta^\star\|_0 \in \{20, 40, 60, 80, 100\}$. Regarding main only settings, we have $\|\beta^\star\|_0 \in \{20, 40, 60, 80, 100\}$ and while $\|\Theta^\star\|_0 \in \{20, 40, 60, 80, 100\}$ in interactions only setting.

Predictive performances. We are interested again, in a first step, in the predictive performances Figure 4.4. As before it appears, that RAMP-WK fail since it get in all settings the worst Mean Squared Error. While RAMP-WK κ , RAMP-ST κ and RAMP-ST bring a lower score than RAMP-WK, they do not reach a score as low as our approach. However, CLEARLASSO with Interactions always achieve the lowest MSE, except in the main only settings, where estimators targeting only the main effects get similar results.

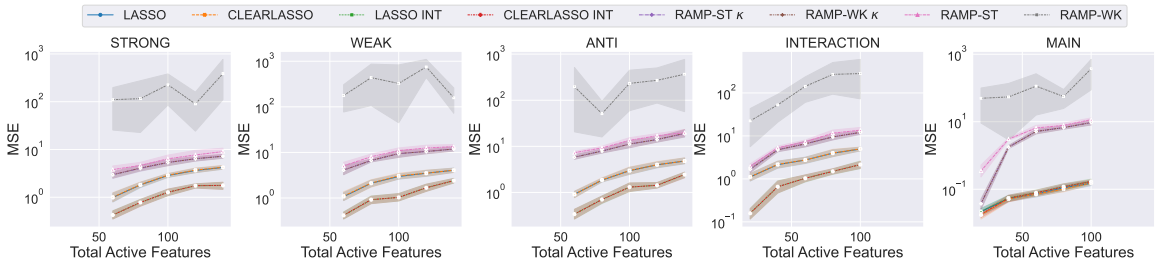


Figure 4.4: Mean Squared Error (MSE) comparisons of CLEARLASSO with Interactions with RAMP in the case where $p = 160$, $q = 12\ 880$ and $n = 1629$. As in first simulation, RAMP-WK fails to predict new outcome, while LASSO with Interactions and CLEARLASSO with Interactions outperform all methods.

Features selection performances. Then, we focus on features selection ability performances Figure 4.5. We first observe that LASSO and CLEARLASSO with Interactions have the largest number of active features, followed by method targeting only main effects and then by the RAMP estimators, nonetheless, all methods achieve close F_1-score in strong, weak and main only settings, except RAMP-WK which fails again. More precisely, RAMP-ST κ and RAMP-WK κ have the same performance and notably obtain the highest precision in all settings, except interactions settings. Unfortunately, they also achieve the lowest recall, notably in anti and interaction settings, and then achieve low F_1-score in anti and interactions cases. Then, regarding RAMP-ST, it behaves as RAMP-ST κ and RAMP-WK κ . With regard to CLEARLASSO with Interactions approach, it achieves in all cases the best recall score, followed by method targeting only main-effects, except in interactions settings where it fails as expected. Finally, RAMP κ and our approach have a similar F_1-score score in strong and weak setting, while

in anti heredity and interactions only setting, LASSO and CLEARLASSO with Interactions perform better. Regarding main only settings, we observe that debiasing step helps to improve precision and thus the F_1-score. Hence, debiasing step achieves a F_1-score close to RAMP κ in the highest sparsity level, while CLEARLASSO with Interactions achieves the best F_1-score when the sparsity level is low. As in the previous example, the RAMP method adds fewer false positive features than our approach, while conversely, LASSO and CLEARLASSO with Interactions miss less relevant features than RAMP.

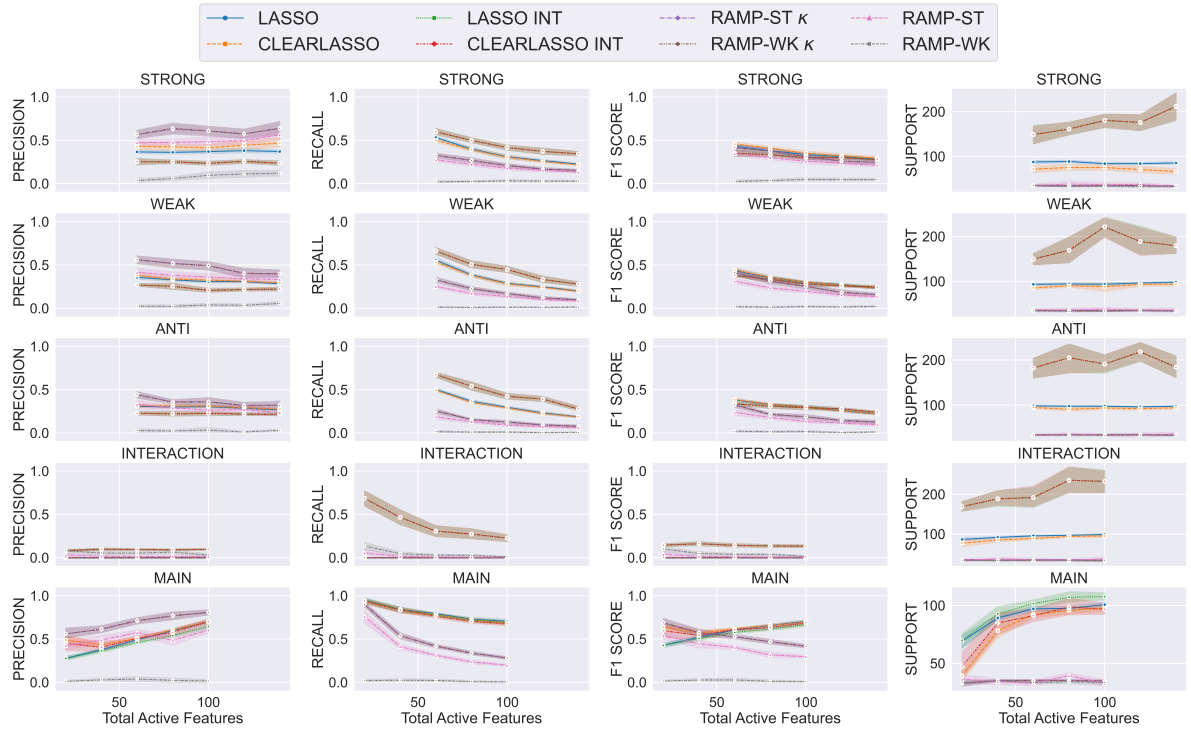


Figure 4.5: Precision, Recall, F_1-score and support comparison between CLEARLASSO with Interactions and RAMP, in the case where $p = 160$, $q = 12\,880$ and $n = 1629$. As a resume, RAMP methods add less false positive features than LASSO with Interactions and CLEARLASSO with Interactions, while the latter miss less relevant features.

Computational performances. Regarding computational performances Figure 4.6, we observe that computational cost are all quite similar. In detail, RAMP-WK is the fastest estimator that targets quadratic regression coefficients while other RAMP estimators take more time than LASSO with Interactions and a similar time that with the debiasing step.

Summary of this second simulation. Since RAMP-ST and RAMP-WK leads to the poorest result, we do not consider it again. Moreover, RAMP and CLEARLASSO with Interactions perform similarly with respect to F_1-score, while regarding precision and recall, RAMP favors the former while LASSO with Interactions the latter.

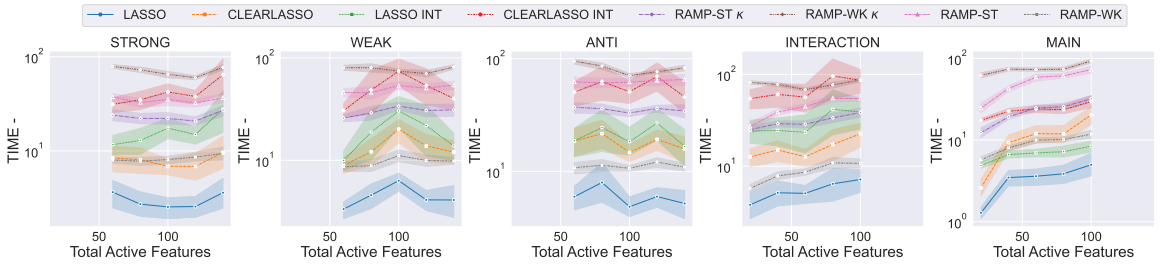


Figure 4.6: Computational time (in second) of CLEARLASSO with Interactions and RAMP for hyperparameter selection and final refitting, in the case where $p = 160$, $q = 12\ 880$ and $n = 1629$.

4.1.4 Simulation 3: p=160 features and n=16294 samples

This last artificial experiment focuses on same scale that our biological application, *i.e.*, $p = 160$, $q = 12\ 880$ and $n = 16294$. As previously, in strong, weak and anti-heredity settings, $\|\beta^\star\|_0 = 40$, while $\|\Theta^\star\|_0 \in \{20, 40, 60, 80, 100\}$. In other case, we have $\|\beta^\star\|_0 \in \{20, 40, 60, 80, 100\}$ in main only settings and $\|\Theta^\star\|_0 \in \{20, 40, 60, 80, 100\}$ in interaction only setting.

Predictive performances. Again, we start by predictive performances Figure 4.7: while the method targeting only main effects fails, we observe that RAMP κ and our approach obtain close MSE when the umber of active features is low, however, when this number increases, LASSO with Interactions achieves a lower MSE than RAMP κ . Lastly, with respect to RAMP-ST κ and RAMP-WK κ , the former fails in anti-heredity settings while the latter fails in pure interactions. This may be explained by the fact that the first selection step of RAMP κ selected highly correlated features and then, the second estimation step done performed by Least Squares fails because of a non-full rank matrix. Nevertheless, in main only settings, all methods perform similarly.

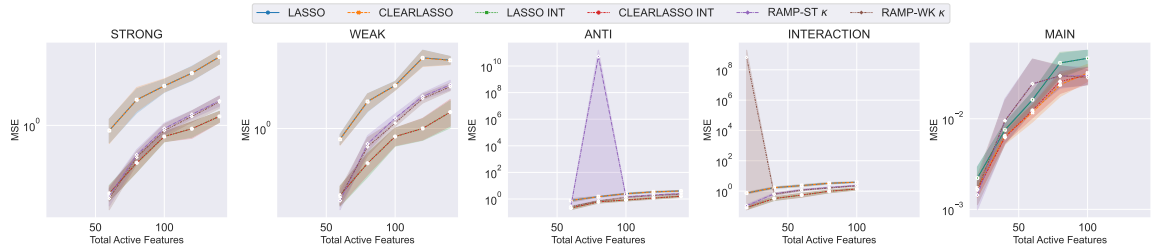


Figure 4.7: Mean Squared Error (MSE) comparisons of CLEARLASSO with Interactions with RAMP in the case where $p = 160$, $q = 12\ 880$ and $n = 16294$. LASSO and CLEARLASSO which do not target interaction perform the worst MSE score, whereas LASSO with Interactions and CLEARLASSO with Interactions outperform others methods in all scenarios with active interactions. More detail in paragraph *Predictive Performances*.

Features selection performances. Figure 4.8 illustrates features selection ability. We observe that all estimators get close F_1-score in strong and weak cases, while LASSO with Interactions does slightly better in anti-heredity and much better in interactions only settings. Regarding main only settings, the non-debiased version achieve the lowest F_1-score, even the standard LASSO without interactions, while both debiased version recover result close to RAMP κ estimator. With regard to precision, in strong, weak and anti heredity settings, all methods achieve close score, while LASSO with Interactions gets better results in interactions only settings. We observe similar precision result than F_1-score, in main only settings. Finally, LASSO and CLEARLASSO with Interactions achieve the best recall performance, which must be related to the large number of active features, followed by RAMP which unlike second experiments, gets higher number of active features than method targeting only main effects.

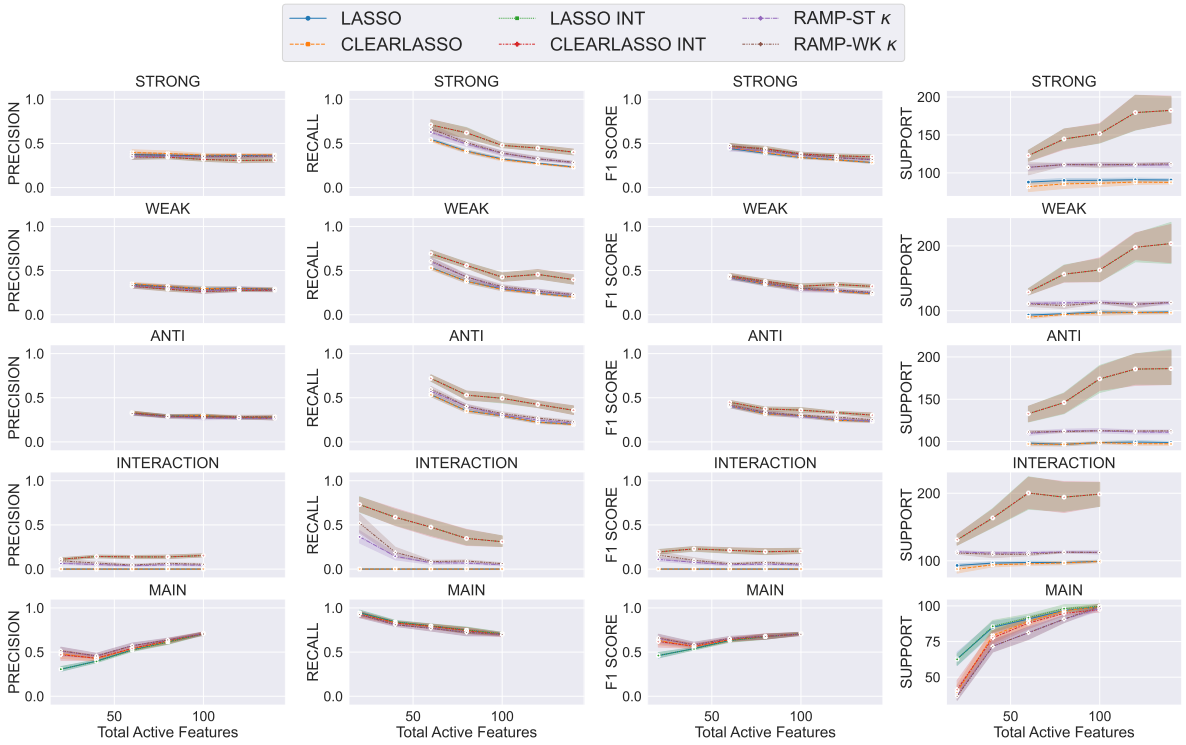


Figure 4.8: Precision, Recall, F_1-score and support comparison between CLEARLASSO with Interactions and RAMP, in the case where $p = 160$, $q = 12\ 880$ and $n = 16294$. In this experiment, all estimators perform closely, with similar precision in strong, weak and anti heredity settings, while we achieve better precision in interaction, but lower than RAMP κ in main effects only.

Finally, RAMP-ST κ and RAMP-WK κ do not fail in this experiment, whereas they have failed with respect to the MSE score Figure 4.7 in some experiments, suggesting that the selection ability works, but the estimation failed due to highly correlated features.

Computational performances. We conclude with the computational performances in Figure 4.9, where we obtain similar results compare to the second experiment: RAMP-WK κ is the estimator which have the heaviest computational cost, followed by CLEAR-LASSO with Interactions and then RAMP-ST κ . Hence, LASSO with Interactions achieves the lowest computational cost among estimators targeting quadratic coefficients.

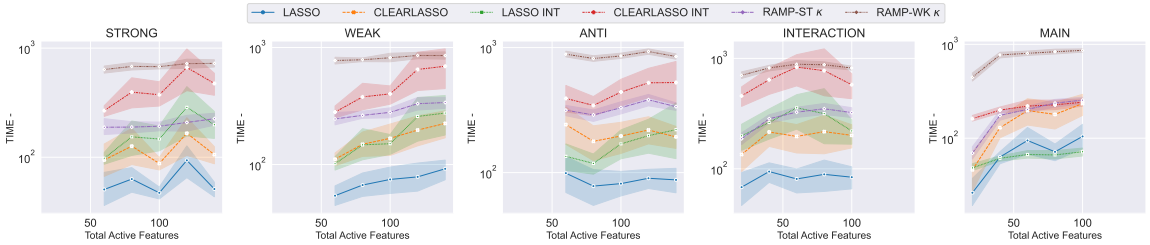


Figure 4.9: Computational time (in second) of CLEARLASSO with Interactions and RAMP for hyperparameter selection and final refitting, in the case where $p = 160$, $q = 12\ 880$ and $n = 16294$. LASSO with Interactions is the fastest method in this experiment among method targeting interactions features.

Summary of this third simulation. Lastly, to summarize, we observe that RAMP and LASSO with Interactions (and its debiased version) achieve similar both predictive and features selection performances, even if, LASSO and CLEARLASSO with Interactions seem to be less sensitive to highly correlated features than RAMP according MSE and succeed to adapt to all the different interactions settings, while RAMP or HierNet cannot. Finally, we also notice that with an equal number of features between the second and the third experiments, the computational time of the latter which have more sample greatly increases. To conclude, for all estimators, we observe that recall and F_1-score are identical between strong, weak, and anti heredity, while precision decreases from strong to weak heredity, and then further decreases from weak to anti-heredity.

4.2 Experiments on real dataset

For real data application, we consider the same genomics dataset from [Bessière et al., 2018] described in introduction (section 1.6.2) and used for semi-artificial data in simulation study. We keep the same optimization parameters as in the previous part and continue to split the dataset in 80% for the train set and 20% for the test set. We consider 10 samples (with size $n = 16294$ genes, $p = 160$ main effects and $q = 12\ 880$ interactions) measured in different cancer types, in order to compare RAMP with CLEARLASSO with Interactions and CLEAR-Enet with Interactions approach. Based on its performance on the semi-artificial data, we decided to keep only the RAMP κ version of the method

for the real dataset study. Considering other interaction operators than element-wise product can improve the prediction as well as interpretation. For example, in the context of this genomic dataset, products between features are not easily interpretable. We therefore consider here the element-wise maximum which can be interpreted as a logical `AND` if both frequencies are low, while a high maximum indicates that at least one of the features has a high frequency and can be interpreted as a logical `OR` (section 1.6.3). Additionally, we compare our solvers with both standardization schemes (STD 1 and STD 2) described in the introduction section 1.6.3 and investigated in Chapter 2. We start by discussing the predictive performance, the number of active features and the computational cost for each of the studied methods, while in a second part, we are interested in the interpretations of these results, in relation to our biological problem.

4.2.1 Statistical performance

In this section, we analyze the studied methods in terms of predictive performance, number of selected features and computation time.

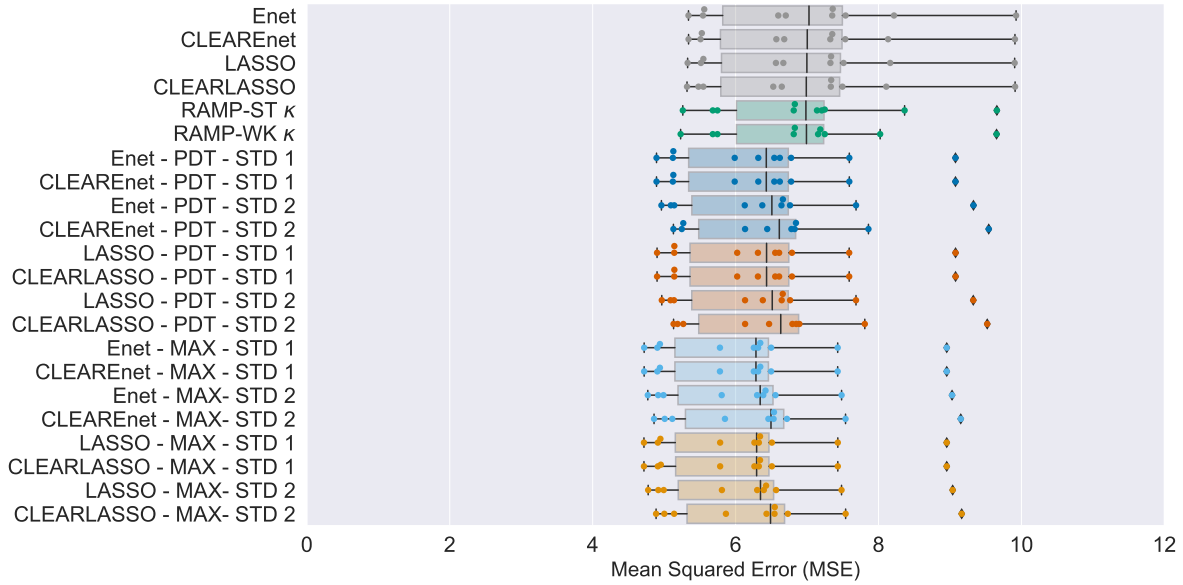


Figure 4.10: Mean Squared Error comparison of RAMP with different versions of CLEAR-Enet with Interactions estimators. While RAMP κ perform as well as main effects methods, CLEAR-Enet with Interactions reduces the MSE, notably with maximum interactions. Regarding standardization, we observe that the first scheme (STD 1) leads to a lower MSE than the second (STD 2), however, interactions choice has more importance.

Predictive performances. Boxplots of the mean square error obtained with each method on the 10 samples are shown in Figure 4.10. Our different approaches with interactions succeed to reduce MSE compared to methods based on main effect only (top 4

boxplots: Elastic Net, CLEAR-Enet, LASSO and CLEARLASSO), or even to RAMP κ method. RAMP-ST κ and RAMP-WK κ again have similar behavior and obtain MSE similar to methods which do not target quadratic effects, but we know that RAMP's hyperparameter selection method does not optimize MSE (unlike cross-validation). Among our approaches exploiting feature interactions, we observe that the first standardization scheme (STD 1) leads to smaller MSE, with or without a debiasing step, while the second standardization scheme (STD 2), slightly decreases the performances, in particular when debiasing step is applied. Lastly, we observe that considering maximum interactions slightly reduces the MSE with respect to the product.

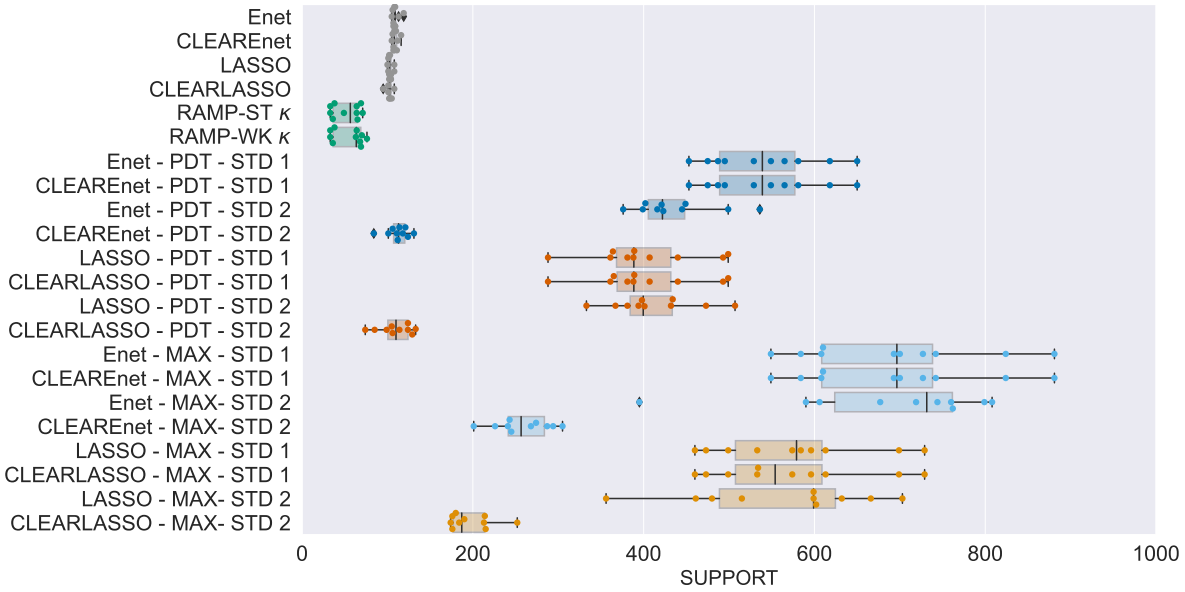


Figure 4.11: Number of active features of RAMP with different versions of CLEAR-Enet with Interactions estimators. We observe that RAMP κ methods obtain a number of active features smaller than methods targeting main features only. It appears that while the choice of a LASSO or Elastic Net penalties affects the number of active features, the choice of interactions or standardization have more influences. Notably, the product interactions provides a more parsimonious estimate, but it is the debiasing step with the second standardization scheme that most reduces the number of active features.

Number of active features. Boxplots of the number of active features are plotted in Figure 4.11. The method including the lowest number of active features is RAMP κ . Indeed, LASSO and Elastic Net, even with debiasing step, select more features than RAMP κ , for a similar MSE. Regarding our approach with interactions, the number of features estimated non-zeros varies greatly depending on the estimators (pure LASSO or an Elastic Net). As expected, LASSO provides fewer active features than Elastic Net does. However, the most noticeable result here is that the number of active features is drastically reduced when using the standardization process (STD 2) (*i.e.,* building the

interactions design matrix Z from standardized main design matrix X) combined with the debiased step, while keeping a small MSE (Figure 4.10). Notably, we observe that, with the element-wise product for modeling interactions, debiasing reduces the number of active features to a number similar to that without interaction.

Computational cost. Finally, regarding computational burden, we observe in Figure 4.12 that RAMP κ , methods which only target main effects and finally LASSO with Interactions and Elastic Net with Interactions for element-wise product have a similar computational cost. However, we also observe that considering the maximum takes longer than the product, which may be partly explained by the higher level of correlation of the element-wise maximum interactions. Also, we see in Figure 4.11 that considering maximum interactions increases the number of active features, hence, more features requires to be estimated. Lastly, it appears that regardless the interactions or the standardization scenario; the debiasing step of CLEARLASSO and CLEAR-Enet with Interactions increases by a factor two or three the computational burden.

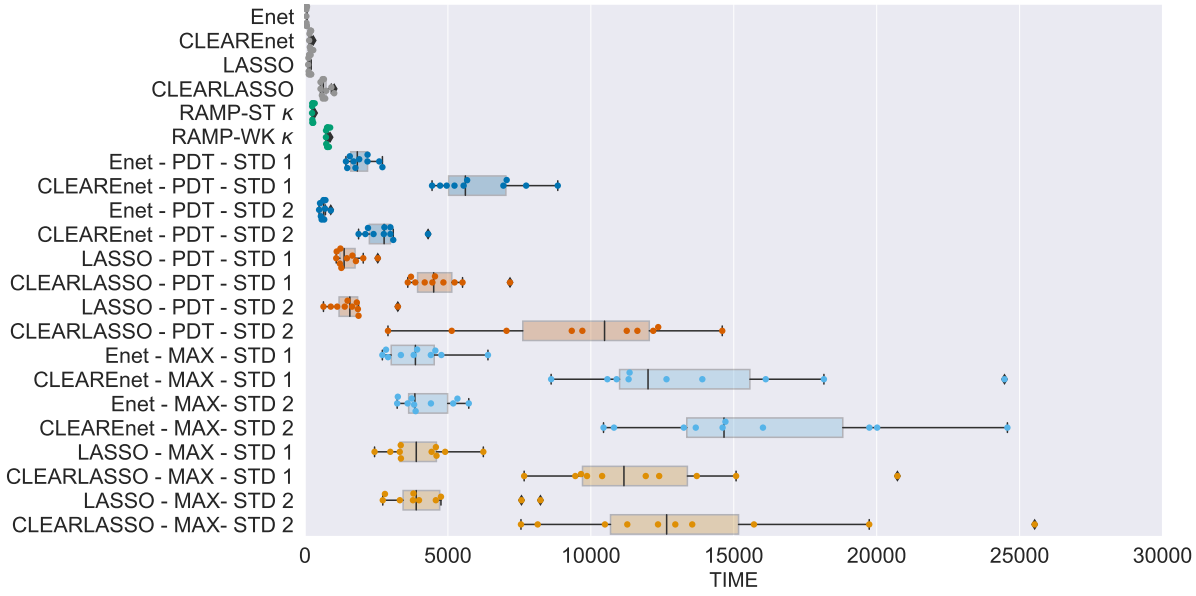


Figure 4.12: Computational cost of RAMP with different versions of CLEAR-Enet with Interactions estimators. Regarding maximum, it greatly increases computational time, but the factor which increase it the most is the debiasing step, since it double or even triples the computational cost.

Summary. Even if CLEARLASSO with Interactions with maximum interactions need a long time to be estimated, it achieves a low MSE and brings highly sparse estimates using an interpretable operator for interactions. Hence CLEARLASSO with Interactions constitutes the best choice in this application.

4.2.2 Features decomposition

In this section, we begin by discussing the distribution of active features between main and interaction effects. We also want to know whether the active interactions satisfy the assumptions of strong or weak heredity, although our approach does not enforce it.

Decomposition of active features. The first thing that we observe in Figure 4.13 is that RAMP selects mainly main effects and only a few active interaction features. Regarding LASSO with Interactions and Elastic Net with Interactions, these estimators select mostly quadratic features, as expected since they have only 160 main effects. Moreover, it appears that they always have less than one hundred main active features, even with the debiasing step. Notably, debiasing step in the second standardization settings mainly discard interactions, and does not seem to erase main effects as much.

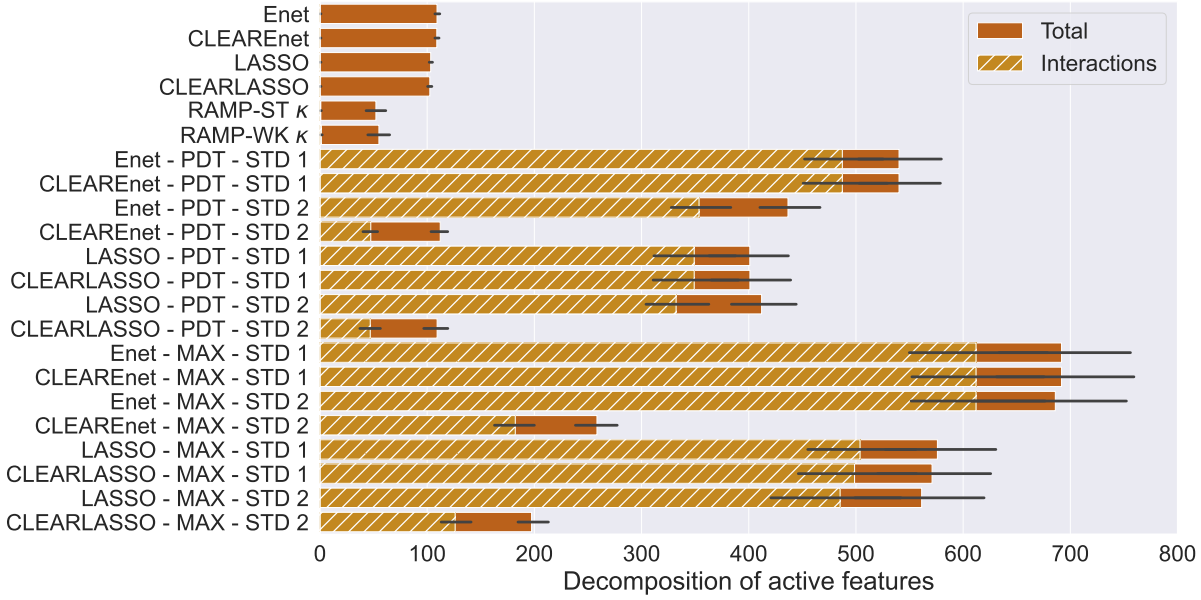


Figure 4.13: Decomposition of active features between main and interactions. We observe that RAMP κ only selects a few interactions, notably compare to LASSO with Interactions and Elastic Net with Interactions. However, debiasing step can discard many active interactions, without delete as many main effects. Thus, in plus to reduce the number of active features, it aims to bring estimate with more main effects, which are more stable.

Hierachial decomposition. Figure 4.14 illustrates whether or not the quadratic active coefficients respect the strong or weak hierarchy assumptions. It appears that in product interaction settings, the half of the interactions appears to satisfy weak hierarchical settings, *i.e.,* the half of the interactions has at least one of their main associated effects which is active. With respect to maximum interactions, they have a one-third of

the active quadratic coefficients that appears to respect strong heredity, whereas two-third seems respects the weak assumptions. Hence, even if we do not enforce strong or weak hierarchy, it appears that most active features respect a hierarchy, regardless of the normalization scheme or the debiasing step.

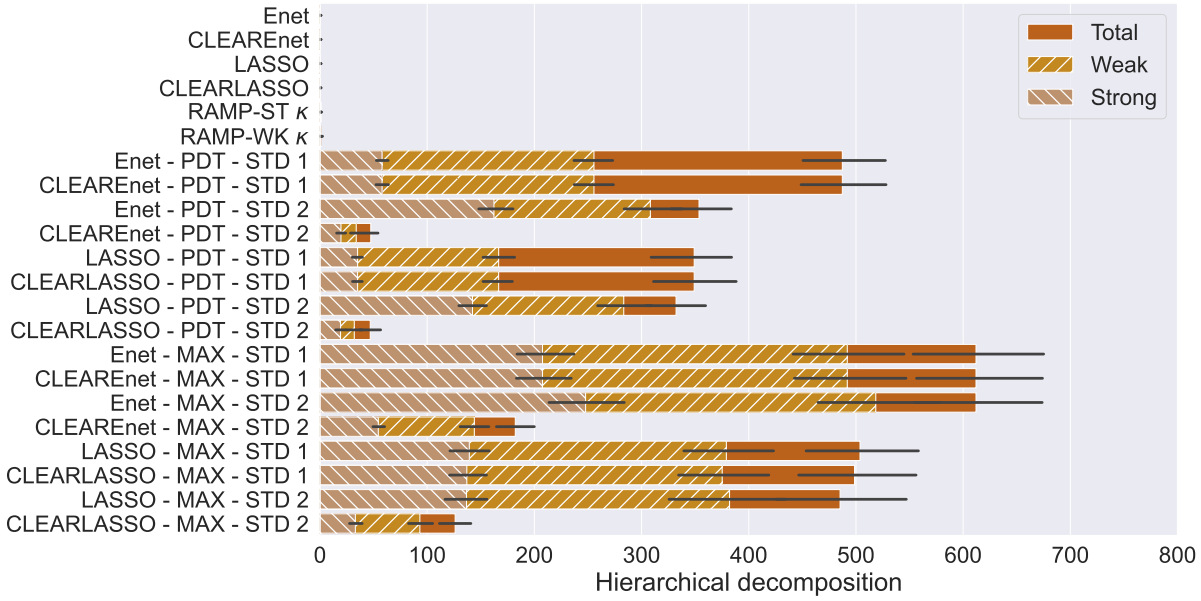


Figure 4.14: Hierarchical Decomposition of active interactions. Main part of the actives interactions satisfies a heredity assumption, regardless of the standardization or choice of interactions.

4.2.3 Biological interpretation

In this last part, we are interested in biological interpretation of active interactions. In the first Figure 4.15, we illustrate the number of interaction which have these both main effects in the same `DNA` regions. Moreover, in the second Figure 4.16 we show the number of active features in each of the eight regions, while in the third Figure 4.17, we illustrate the number of nucleotides and dinucleotides which are non-zero for each estimator. Lastly, Figures 4.18 and 4.19 give the couples of active regions-regions.

Decomposition of interactions features. We observe that the active quadratic features are mainly constituted, for product as well as for maximum interactions, by interactions whose associated main effects remains in different `DNA` regions. Hence, it appears that interactions between regions seems more important than interactions intra-regions.

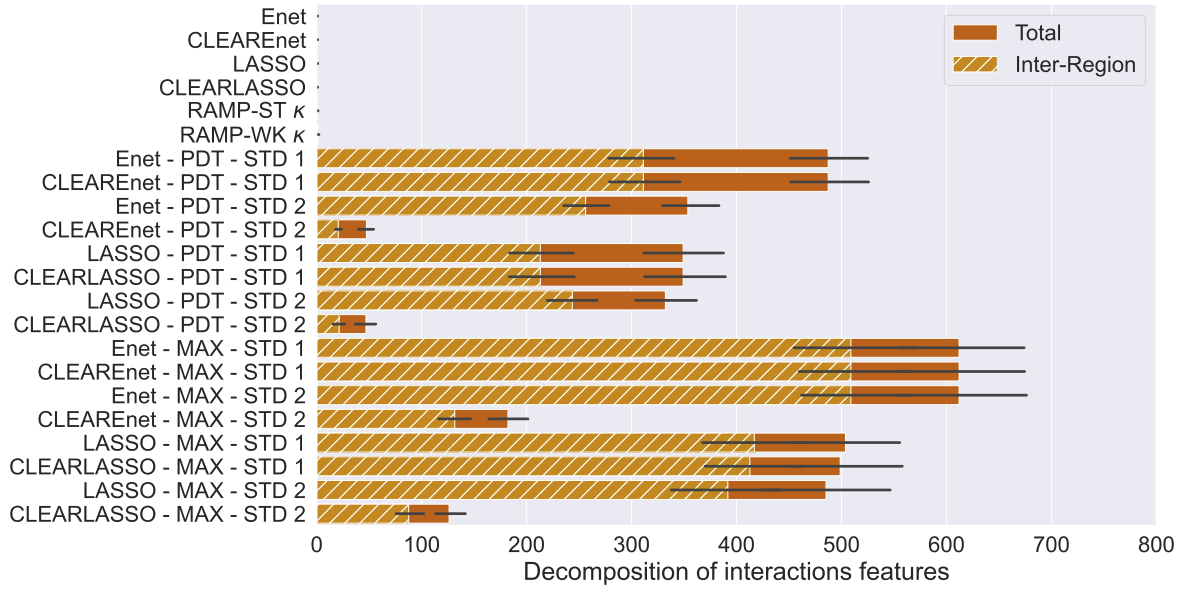


Figure 4.15: Study of number of active quadratic features which have associated main effects in different regions. It appears that the main part of interactions are associated to main effects which are not present in the same DNA regions.

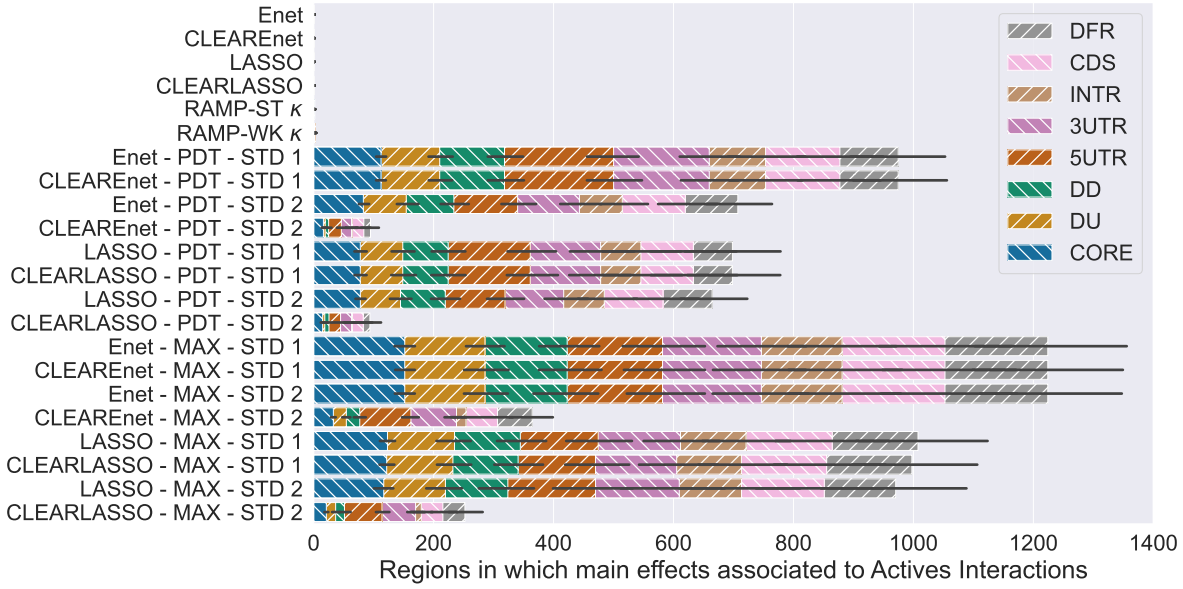


Figure 4.16: Regions where main effects associated to actives interactions are present. All regions seem to be present in a similar way.

Regions which contain main effects associated to actives quadratic coefficients. Regarding Figure 4.16, all the eight DNA regions are represented, in similar proportions. However, the debiasing step of CLEARLASSO with Interactions and CLEAR-

Enet with Interactions with the second standardization scheme in the maximum case, reduces all the regions but 5UTR, 3UTR, CDS and INTR seem less decreased.

Nucleotide and di-nucleotide associated to active quadratic coefficients. We illustrate in fig. 4.17, which are the features associated to active interactions. The four nucleotides are represented in the left of each bar, while the sixteen di-nucleotides are represented on the right after nucleotides.

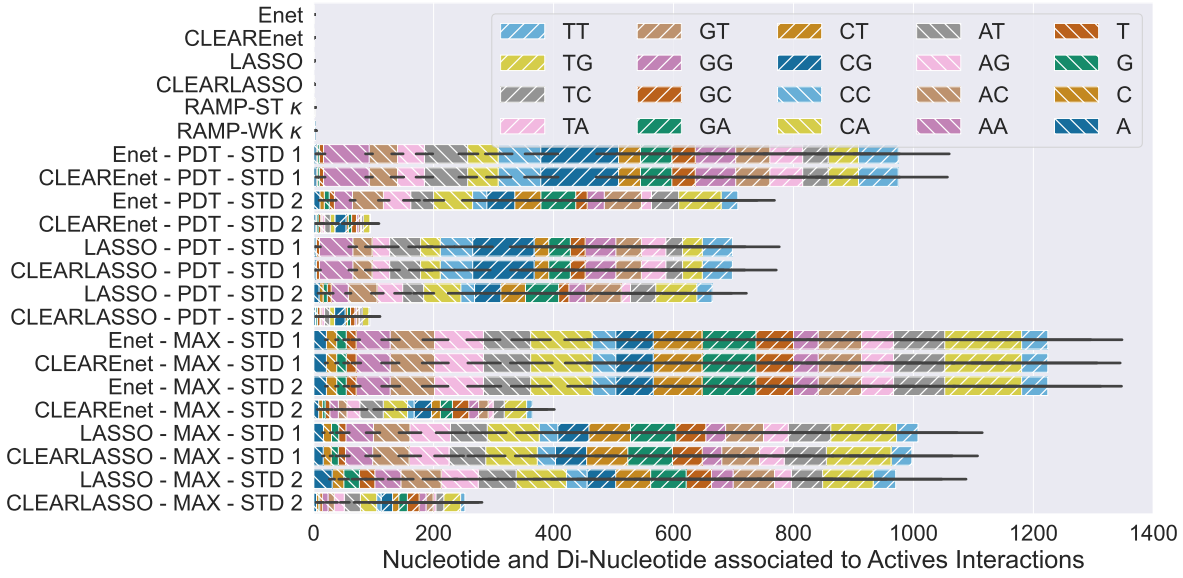
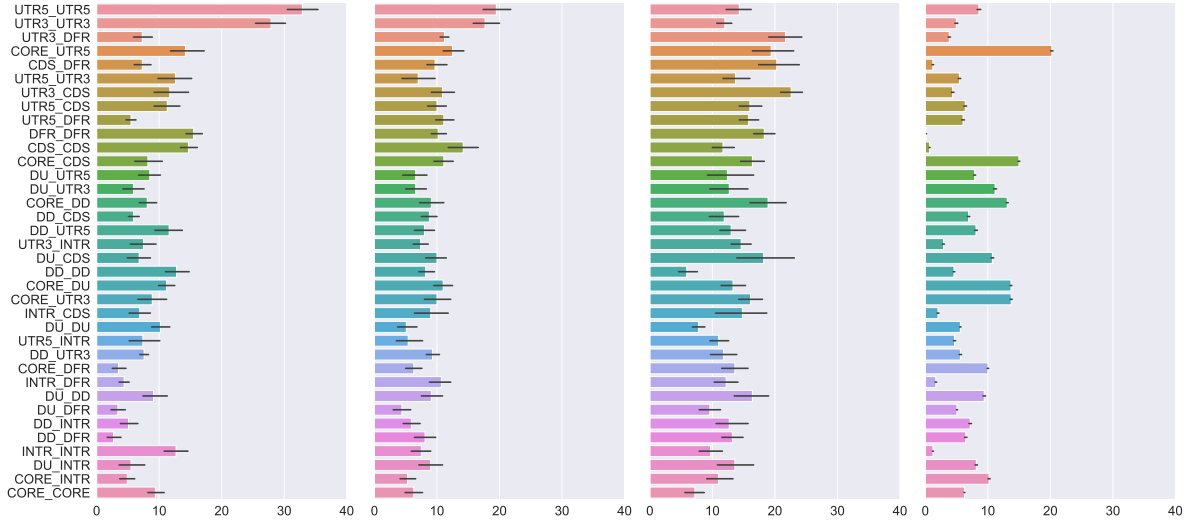


Figure 4.17: Nucleotide and Di-Nucleotide Distribution among the features associated to active interactions. The left part of each bar represented the nucleotides while the right part associated show di-nucleotides. The former are less represented in the decomposition, while the latter seems more represented even if all di-nucleotides do not have equal part.

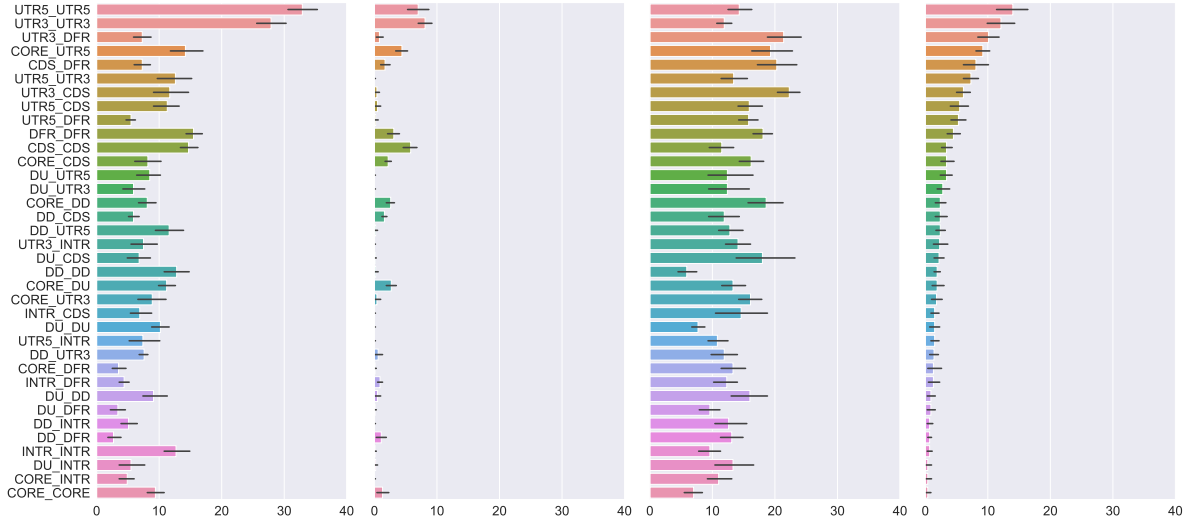
We observe that the four nucleotides are in the minority compared to the total number of features, while all sixteen di-nucleotides are represented. However, we observe that some di-nucleotides are more represented than others, as TG for example.

Region-region pairs of main effects corresponding to the active interactions. Figure 4.18 illustrates the region-region pair for the LASSO with Interactions for maximum and product, whereas Figure 4.19 illustrates it for the CLEARLASSO with Interactions estimator. We decide to rank all the pair in function of CLEARLASSO with Interactions with the element-wise maximum in the second standardization scheme (STD-2) results. These graphs tend to confirm what we have already observed in Figure 4.16, most pairs of regions are considered, but it seems once again that the regions UTR3, UTR5 and CDS are more selected than others.



(a) Product - STD1 (b) Product - STD2 (c) Maximum - STD1(d) Maximum - STD2

Figure 4.18: Region-region pairs of main effects corresponding to the active interactions for LASSO with Interactions. All pairs seem to be present in a similar way, even if the pairs that involve the regions UTR3, UTR5 and CDS seem more represented.



(a) Product - STD1 (b) Product - STD2 (c) Maximum - STD1(d) Maximum - STD2

Figure 4.19: Region-region pairs of main effects corresponding to the active interactions for CLEARLASSO with Interactions. Debiasing reduce the number of active pair region in second standardization, however, the pairs which involve the regions UTR3, UTR5 and CDS seem again more represented.

4.3 Conclusion

In this chapter, we are interested in the statistical performance of our approach on semi-simulated and real datasets.

Semi-simulated data The first part on semi-simulated data, has first allowed to show that our approach, with or without the debiasing step, always obtained the lowest predictive error (Figures 4.1, 4.4 and 4.7).

It also showed that in terms of features selection, it was competitive in the strong and weak heredity simulation scenarios, while in the anti-hierarchical or pure interactions scenarios, it succeeds to obtain better F_1-score than others estimators (Figures 4.2, 4.5 and 4.8). If we are interested in the precision score, we observe that RAMP has a smaller number of active features than us, thus it includes less false positives and obtains a higher precision score than ours. However, this smaller support also leads it to miss relevant features, so we obtain a better recall score. Conversely, HierNet estimates a large number of non-zero features and get a better recall than us, but failing to limit the number of false positives, leading to low a precision score. Also, these global performances in features selection are lower than ours or RAMP's as illustrated by the F_1-score.

Moreover, since we enforce no heredity structure, on the first three scenarios: strong, weak and anti-hierarchical heredity, we obtain relatively constant recall and F_1-score, although the precision scores decrease slightly from strong to weak heredity, then from weak to anti-hierarchical. Also, these results tend to indicate that regardless of the generative scenario, our approach obtains constant results.

Moreover, if we focus on the generative scenario involving only main effects, we observe that the debiasing step of CLEARLASSO with Interactions allows to match the performances of the methods involving only main effects.

Finally, concerning the numerical aspect, our algorithm, without the debiasing step, is the fastest among those that search for interactions (Figures 4.3, 4.6 and 4.9), except in comparison to RAMP-WK which however gets the worst statistical results.

Real dataset. In a second part, we compare only RAMP κ to our approach on real dataset, HierNet being computationally intractable. In particular, we proposed to use our approach with two possible interactions: the product and the maximum, but also with the two possible standardization scenarios. We observe that RAMP selects very few interactions (Figure 4.13) but still manages to obtain a prediction similar to the method searching only the main effects. Moreover, thanks to its debiasing step, RAMP obtains a smaller support than the latter. Regarding our results, we observed that we outperform competitor in predictive performance (Figure 4.10), while managing to limit the number of active features thanks to debiasing in the second standardization scenario (Figure 4.11). Also, although the maximum and the debiasing step favor the interpretability, the numerical cost of the approach is still higher (Figure 4.11).

Furthermore, regarding our approach, we observe in Figure 4.11 that the majority of the selected features are interactions. Moreover, Figure 4.14 illustrates the fact that the

main part of these interactions respect the hypothesis of weak heredity. In addition, we observe in Figure 4.15 that the majority of the selected interactions have the two main effects in different regions, which encourages us to study the interactions between regions, rather than the ones within the regions. Also, it appears in Figures 4.16, 4.18 and 4.19 that the main effects associated with active interactions are present equally in all regions, if we look at the results without debiasing. On the other hand, if we look at the results with the debiasing step, it appears that the regions UTR3, UTR5 and CDS are favored. Finally, concerning the distinction between nucleotides and di-nucleotides, we observe in Figure 4.17 that nucleotides are very little selected in contrast to the di-nucleotides, which all seem to be selected in similar proportions, except for the two di-nucleotides TG and CA which are slightly preferred.

Chapter 5

Conclusions and Perspectives

Conclusions. In this thesis, we first introduced an Elastic Net with Interactions based estimator to tackle quadratic regression problems, with an additional debiasing step to counterweight the bias from both ℓ_1 and ℓ_2 penalties. Experiments on semi-artificial datasets have shown that penalizing the interactions coefficients more than the main effects coefficients improves statistical performance, while the debiasing step provides additional improvement in most settings.

Then, in a second part, we developed a scalable algorithm based on an active set strategy, to benefit from the sparse solution of Elastic Net with Interactions, whose key idea is to avoid as much of as possible visiting the full interactions design matrix Z. In addition, we adapted the Anderson Acceleration to provide additional speedup. Experiments have shown that our algorithm scales to a large number of both features and samples, without ever storing the interaction matrix.

Finally, the last part of the thesis focuses on the comparison of CLEAR-Enet with Interactions with two standard estimators of the quadratic regression problem under hierarchical constraints: HierNet and RAMP. We have illustrated on semi-artificial datasets that CLEAR-Enet with Interactions performs as well as RAMP and better than HierNet in strong and weak scenarios, while in anti-hierarchical and pure interaction scenarios CLEAR-Enet with Interactions performs better than both approaches. Moreover, controlled experiments have shown that when only the main effects are active, CLEAR-Enet with Interactions thanks to the debiasing step, performs as well as the methods without interactions.

Thus, it seems that CLEAR-Enet with Interactions does not add interaction features if the truth does not contain any. Concerning the experiments on real data, the results have shown that taking interactions into account effectively reduces the predictive error. In particular, it illustrates the interest of considering interactions in applications, since element-wise maximum reduce the predictive error compared to the element-wise prod-

uct. Lastly, we observe that the debiasing step effectively reduces the number of active features for a similar predictive error, in some standardization settings.

Perspectives. An immediate perspective would be to consider a MultiTaskElasticNet with Interactions estimator, to take into account the multiple responses for each cancer of the genomics dataset. A first step to consider such an estimator is to adapt the proximal gradient coordinate descent of Elastic Net with Interactions by proximal gradient block coordinate descent, and then adapt MultiTaskLASSO [Obozinski et al., 2010] the duality gap and working set strategies from MultiTaskLASSO CELER works [Massias et al., 2020].

A second perspective with greater potential for application would be to adapt our work to the sparse logistic regression case [Koh et al., 2007]. However, [Hsieh et al., 2014] proves that unlike the LASSO case, first-order algorithms such as proximal gradient coordinate descent are slower than second-order algorithms such as Proximal Newton algorithm for solving sparse logistic problems. Hence, Massias et al. [2020] have provided WorkingSet algorithm, whose inner solver is based on the latter. A first step to adapt these estimators to the interactions' case would require to adapt their algorithm.

In both cases, the main challenge is again to provide an algorithm that avoids as much as possible visiting the quadratic coefficients and the associated design matrix. Lastly, since the debiasing step greatly improves the selection ability of Elastic Net with Interactions, a second step will be to adapt the CLEAR update rules in this context.

Chapter 6

Appendix

Contents

In this chapter, we provide the proofs of propositions 3.1.1 and 3.2.1 from chapter 3. We first provide in section 6.1 the proof of the equivalence between LASSO with Interactions and Elastic Net with Interactions. Then, thanks to this equivalence, we provide the duality gap proof in section 6.2 and the CELER for Elastic Net with Interactions proof in section 6.3.

6.1 Equivalence between LASSO and Elastic Net

From [Zou and Hastie, 2005, Lemma 1], we know that Elastic Net can be written as a LASSO with augmented data. The following proposition shows the equivalence between LASSO and Elastic Net in the interaction settings.

Proposition 6.1.1. *Let us recall the Elastic Net with Interactions estimator:*

$$\left(\widehat{\beta}, \widehat{\Theta}\right) \in \underset{\substack{\beta \in \mathbb{R}^p \\ \Theta \in \mathbb{R}^q}}{\arg\min} \frac{\|y - X\beta - Z\Theta\|_2^2}{2n} + \alpha_{1,1} \|\beta\|_1 + \alpha_{1,2} \|\Theta\|_1 + \frac{\alpha_{2,1} \|\beta\|_2^2 + \alpha_{2,2} \|\Theta\|_2^2}{2} \quad , \quad (6.1)$$

Given a dataset (y, X, Z), we define the following augmented artificial dataset (y^*, X^*, Z^*):

$$y^* = \begin{pmatrix} y \\ 0_p \\ 0_q \end{pmatrix} \quad \text{and} \quad X^* = \begin{pmatrix} X \\ \sqrt{\alpha_{2,1} n}\, \mathrm{Id}_{p \times p} \\ 0_{q \times p} \end{pmatrix} \quad \text{and} \quad Z^* = \begin{pmatrix} Z \\ 0_{p \times q} \\ \sqrt{\alpha_{2,2} n}\, \mathrm{Id}_{q \times q} \end{pmatrix} . \quad (6.2)$$

The Elastic Net estimator can be rewritten as a LASSO with augmented dataset:

$$\left(\widehat{\beta},\widehat{\Theta}\right) \in \underset{\substack{\beta\in\mathbb{R}^p\\ \Theta\in\mathbb{R}^q}}{\arg\min}\left(\frac{1}{2n}\left\|y^*-X^*\beta-Z^*\Theta\right\|_2^2+\alpha_{1,1}\left\|\beta\right\|_1+\alpha_{1,2}\left\|\Theta\right\|_1\right) \quad . \tag{6.3}$$

Proof. We define $\mathrm{pen}_{\ell_1}(\beta,\Theta;\alpha)=\alpha_{1,1}\left\|\beta\right\|_1+\alpha_{1,2}\left\|\Theta\right\|_1$, to shorten the equations. The proof is simple linear algebra:

$$\begin{aligned}
&\left(\widehat{\beta},\widehat{\Theta}\right) \in \underset{\substack{\beta\in\mathbb{R}^p\\ \Theta\in\mathbb{R}^q}}{\arg\min}\frac{1}{2n}\left\|y-X\beta-Z\Theta\right\|_2^2+\frac{\alpha_{2,1}\left\|\beta\right\|_2^2+\alpha_{2,2}\left\|\Theta\right\|_2^2}{2}+\mathrm{pen}_{\ell_1}(\beta,\Theta;\alpha) \quad ,\\
\iff&\left(\widehat{\beta},\widehat{\Theta}\right) \in \underset{\substack{\beta\in\mathbb{R}^p\\ \Theta\in\mathbb{R}^q}}{\arg\min}\frac{1}{2n}\left\|y-X\beta-Z\Theta\right\|_2^2+\frac{\left\|\sqrt{\alpha_{2,1}}\beta\right\|_2^2+\left\|\sqrt{\alpha_{2,2}}\Theta\right\|_2^2}{2}+\mathrm{pen}_{\ell_1}(\beta,\Theta;\alpha) \quad ,\\
\iff&\left(\widehat{\beta},\widehat{\Theta}\right) \in \underset{\substack{\beta\in\mathbb{R}^p\\ \Theta\in\mathbb{R}^q}}{\arg\min}\frac{1}{2n}\left(\sum_{i=1}^{n}(y_i-x_{i,:}\beta-z_{i,:}\Theta)^2+\sum_{i=1}^{p}\left(\sqrt{n\alpha_{2,1}}\beta_i\right)^2+\sum_{i=1}^{q}\left(\sqrt{n\alpha_{2,2}}\Theta_i\right)^2\right)\\
&\qquad\qquad+\mathrm{pen}_{\ell_1}(\beta,\Theta;\alpha) \quad ,\\
\iff&\left(\widehat{\beta},\widehat{\Theta}\right) \in \underset{\substack{\beta\in\mathbb{R}^p\\ \Theta\in\mathbb{R}^q}}{\arg\min}\frac{1}{2n}\sum_{i=1}^{n+p+q}(y_i^*-x_{i,:}^*\beta-z_{i,:}^*\Theta)^2+\mathrm{pen}_{\ell_1}(\beta,\Theta;\alpha) \quad ,\\
\iff&\left(\widehat{\beta},\widehat{\Theta}\right) \in \underset{\substack{\beta\in\mathbb{R}^p\\ \Theta\in\mathbb{R}^q}}{\arg\min}\frac{1}{2n}\left\|y^*-X^*\beta-Z^*\Theta\right\|_2^2+\mathrm{pen}_{\ell_1}(\beta,\Theta;\alpha) \quad .
\end{aligned}$$

□

6.2 Duality Gap for Elastic Net proof

Proposition 6.2.1. *To the Elastic Net with Interactions minimization problem $\mathcal{P}(\beta,\Theta)$ (eq. (2.7)), is associated a maximization problem, called dual problem $\mathcal{D}(\nu)$:*

$$\widehat{\nu}=\underset{\nu\in\Delta_{X,Z}}{\arg\max}\left(\frac{1}{2n}\left\|y\right\|_2^2-\frac{n\alpha^2}{2}\left\|\nu-\frac{y}{\alpha n}\right\|_2^2-\left(\frac{n\alpha}{c_\alpha}\right)^2\left(\frac{\alpha_{2,1}}{2}\left\|\beta\right\|_2^2+\frac{\alpha_{2,2}}{2}\left\|\Theta\right\|_2^2\right)\right) \quad , \tag{6.4}$$

$$\begin{aligned}
\textit{with } \alpha =&\frac{1}{4}(\alpha_{1,1}+\alpha_{1,2}+\alpha_{2,1}+\alpha_{2,2}) \textit{ and } r=y-X\beta-Z\Theta \quad ,\\
c_\alpha =&\alpha\max\left(n,\frac{\left\|X^\top r-n\alpha_{2,1}\beta\right\|_\infty}{\alpha_{1,1}},\frac{\left\|Z^\top r-n\alpha_{2,2}\Theta\right\|_\infty}{\alpha_{1,2}}\right) \quad ,\\
\Delta_{X,Z}=&\left\{\nu\in\mathbb{R}^n:\max\left(\frac{\left\|X^\top\nu-\frac{n\alpha_{2,1}}{c_\alpha}\beta\right\|_\infty}{\alpha_{1,1}},\frac{\left\|Z^\top\nu-\frac{n\alpha_{2,2}}{c_\alpha}\Theta\right\|_\infty}{\alpha_{1,2}}\right)\leqslant\frac{1}{\alpha}\right\} \quad .
\end{aligned}$$

A canonical dual variable $\widehat{\nu}$ is the rescaled residuals [Mairal, 2010], defined as follows:

$$\widehat{\nu} = \frac{r}{c_\alpha} = \frac{y - X\widehat{\beta} - Z\widehat{\Theta}}{c_\alpha} \quad . \tag{6.5}$$

We use the result from the proposition 6.1.1 to adapt the duality gap from LASSO to the Elastic Net with Interactions estimator.

Proof. Starting from the maximization problem of LASSO [Kim et al., 2007], we provide the maximization problem associated to Elastic Net with Interactions. From the LASSO dual problem, we get that LASSO with Interactions dual problem is the following:

$$\max_{\nu \in \Delta_{X^*,Z^*}} \left(\frac{\|y^*\|_2^2}{2n} - \frac{n\alpha^2}{2} \left\| \nu^* - \frac{y^*}{\alpha n} \right\|_2^2 \right) \tag{6.6}$$

with: $\Delta_{X^*,Z^*} := \left\{ \nu \in \mathbb{R}^{n+p+q}, \left\{ \left\| \frac{X^{*\top}\nu^*}{\alpha_{1,1}} \right\|_\infty \leqslant \frac{1}{\alpha} \right\} \cap \left\{ \left\| \frac{Z^{*\top}\nu^*}{\alpha_{1,2}} \right\|_\infty \leqslant \frac{1}{\alpha} \right\} \right\}$.
Immediately, we get that: $\|y^*\|_2^2 = \|y\|_2^2$, while the residuals are defined as follows:

$$y^* - X^*\beta - Z^*\Theta = \begin{pmatrix} y \\ 0_p \\ 0_q \end{pmatrix} - \begin{pmatrix} X \\ \sqrt{\alpha_{2,1}n}\,\mathrm{Id}_{p\times p} \\ 0_{q\times p} \end{pmatrix} \beta - \begin{pmatrix} Z \\ 0_{p\times q} \\ \sqrt{\alpha_{2,2}n}\,\mathrm{Id}_{q\times q} \end{pmatrix} \Theta \tag{6.7}$$

$$= \begin{pmatrix} y - X\beta - Z\Theta \\ -\sqrt{\alpha_{2,1}n}\beta \\ -\sqrt{\alpha_{2,1}n}\Theta \end{pmatrix} \tag{6.8}$$

Also, the dual variable ν, correspond to the residuals rescaled by the maximum c_α:

$$c_\alpha = \alpha \max \left(n, \frac{\left\| X^{*\top}(y^* - X^*\beta - Z^*\Theta) \right\|_\infty}{\alpha_{1,1}}, \frac{\left\| Z^{*\top}(y^* - X^*\beta - Z^*\Theta) \right\|_\infty}{\alpha_{1,2}} \right) \tag{6.9}$$

Let us adapt this constant, denoting $r = y - X\beta - Z\Theta$:

$$X^{*\top}(y^* - X^*\beta - Z^*\Theta) = \begin{pmatrix} X^\top & \sqrt{\alpha_{2,1}n}\,\mathrm{Id}_{p\times p} & 0_{q\times p} \end{pmatrix} \begin{pmatrix} r \\ -\sqrt{\alpha_{2,1}n}\beta \\ -\sqrt{\alpha_{2,2}n}\Theta \end{pmatrix} \tag{6.10}$$

$$= X^\top r - n\alpha_{2,1}\beta \tag{6.11}$$

In the same way, it comes that: $Z^{*\top}(y^* - X^*\beta - Z^*\Theta) = Z^\top r - n\alpha_{2,2}\Theta$.

So, the maximum c_α is:

$$c_\alpha = \alpha \max\left(n, \frac{\left\|X^\top r - n\alpha_{2,1}\beta\right\|_\infty}{\alpha_{1,1}}, \frac{\left\|Z^\top r - n\alpha_{2,2}\Theta\right\|_\infty}{\alpha_{1,2}}\right) \tag{6.12}$$

Hence, we get the dual variable of Elastic Net with Interactions:

$$\nu^* = \frac{y^* - X^*\beta - Z^*\Theta}{c_\alpha} = \begin{pmatrix} \frac{y - X\beta - Z\Theta}{c_\alpha} \\ -\frac{\sqrt{n\alpha_{2,1}}\beta}{c_\alpha} \\ -\frac{\sqrt{n\alpha_{2,2}}\Theta}{c_\alpha} \end{pmatrix} = \begin{pmatrix} \nu \\ -\frac{\sqrt{n\alpha_{2,1}}\beta}{c_\alpha} \\ -\frac{\sqrt{n\alpha_{2,2}}\Theta}{c_\alpha} \end{pmatrix} \tag{6.13}$$

Then, to compute the second norm of eq. (6.6), we need:

$$\nu^* - \frac{y^*}{\alpha n} = \begin{pmatrix} \nu \\ -\frac{\sqrt{n\alpha_{2,1}}\beta}{c_\alpha} \\ -\frac{\sqrt{n\alpha_{2,2}}\Theta}{c_\alpha} \end{pmatrix} - \begin{pmatrix} \frac{y}{\alpha n} \\ 0_p \\ 0_q \end{pmatrix} = \begin{pmatrix} \nu - \frac{y}{\alpha n} \\ -\frac{\sqrt{n\alpha_{2,1}}\beta}{c_\alpha} \\ -\frac{\sqrt{n\alpha_{2,2}}\Theta}{c_\alpha} \end{pmatrix} \tag{6.14}$$

Hence, we get:

$$\left\|\nu^* - \frac{y^*}{\alpha n}\right\|_2^2 = \sum_{i=1}^{n+p+q}\left(\nu_i^* - \frac{y_i^*}{n\alpha}\right)^2 \tag{6.15}$$

$$= \sum_{i=1}^{n}\left(\nu_i - \frac{y_i}{n\alpha}\right)^2 + \sum_{i=1}^{p}\left(-\sqrt{\alpha_{2,1}n}\frac{\beta_i}{c_\alpha}\right)^2 + \sum_{i=1}^{q}\left(-\sqrt{\alpha_{2,2}n}\frac{\Theta_i}{c_\alpha}\right)^2 \tag{6.16}$$

$$= \left\|\nu - \frac{y}{\alpha n}\right\|_2^2 + \frac{n\alpha_{2,1}}{c_\alpha^2}\left\|\beta\right\|_2^2 + \frac{n\alpha_{2,2}}{c_\alpha^2}\left\|\Theta\right\|_2^2 \tag{6.17}$$

In order to compute the set of constraints, we need to compute $X^{*\top}\nu^*$ and $Z^{*\top}\nu^*$:

$$X^{*\top}\nu^* = \begin{pmatrix} X^\top & \sqrt{\alpha_{2,1}n}\,\mathrm{Id}_{p\times p} & 0_{p\times q} \end{pmatrix} \begin{pmatrix} \nu \\ -\frac{\sqrt{n\alpha_{2,1}}\beta}{c_\alpha} \\ -\frac{\sqrt{n\alpha_{2,2}}\Theta}{c_\alpha} \end{pmatrix} = X^\top\nu - n\alpha_{2,1}\frac{\beta}{c_\alpha} \tag{6.18}$$

Then, we also obtain: $Z^{*\top}\nu^* = Z^\top\nu - n\alpha_{2,2}\dfrac{\Theta}{c_\alpha}$.
Hence, the set of the feasible dual variable $\Delta_{X,Z}$ is:

$$\Delta_{X,Z} := \left\{\nu \in \mathbb{R}^p, \left\{\left\|\frac{X^\top\nu - n\alpha_{2,1}\frac{\beta}{c_\alpha}}{\alpha_{1,1}}\right\|_\infty \leqslant \frac{1}{\alpha}\right\} \cap \left\{\left\|\frac{Z^\top\nu - n\alpha_{2,2}\frac{\Theta}{c_\alpha}}{\alpha_{1,2}}\right\|_\infty \leqslant \frac{1}{\alpha}\right\}\right\} \tag{6.19}$$

Finally, we get the following duality gap:

$$\iff \underset{\nu^*\in\Delta_{X^*,Z^*}}{\arg\max}\left(\frac{1}{2n}\|y^*\|_2^2 - \frac{n\alpha^2}{2}\left\|\nu^* - \frac{y^*}{\alpha n}\right\|_2^2\right) \tag{6.20}$$

$$\iff \underset{\nu\in\Delta_{X,Z}}{\arg\max}\left(\frac{\|y\|_2^2}{2n} - \frac{n\alpha^2}{2}\left\|\nu - \frac{y}{\alpha n}\right\|_2^2 - \left(\frac{n\alpha}{c_\alpha}\right)^2\left(\frac{\alpha_{2,1}}{2}\|\beta\|_2^2 + \frac{\alpha_{2,2}}{2}\|\Theta\|_2^2\right)\right) \tag{6.21}$$

□

6.3 CELER for Elastic Net proof

As for the proof of proposition 3.1.1, we use the result from the proposition 6.1.1 to adapt CELER LASSO ranking rules to Elastic Net with Interactions case.

Proposition 6.3.1 (Celer for Elastic Net with Interactions)**.** *Let $\widehat{\nu}$ a dual feasible point of the dual problem and c_α the associated constant (to rescale residuals), as in Proposition 3.1.1. We get the following d_j and d_{jj} priority rules, for the main and interaction effects respectively:*

$$d_j\left(\widehat{\nu}\right) = \frac{1 - \left|x_j^\top\widehat{\nu} - \dfrac{\alpha_{1,2}n}{c_\alpha}\beta_j\right|}{\sqrt{\|x_j\|_2^2 + \alpha_{1,2}n}} \text{ and } d_{jj}\left(\widehat{\nu}\right) = \frac{1 - \left|z_{jj}^\top\widehat{\nu} - \dfrac{\alpha_{2,2}n}{c_\alpha}\Theta_{jj}\right|}{\sqrt{\|z_{jj}\|_2^2 + \alpha_{2,2}n}} \ . \tag{6.22}$$

Proof. The CELER ranking rules[Massias et al., 2018], for main effects and by extension for the interaction effects, are defined as follows:

$$d_j(\nu^*) = \frac{1 - |x_j^{*\top}\nu^*|}{\left\|x_j^*\right\|_2} \quad \text{and } d_{jj}(\nu^*) = \frac{1 - |z_{jj}^{*\top}\nu^*|}{\left\|z_{jj}^*\right\|_2} \tag{6.23}$$

where x_j^* (resp. z_{jj}^*) is the j^{th} (resp. jj^{th}) column of the augmented design matrix X^* (resp. Z^*) and ν^* the dual variable associated to the LASSO augmented problem.

From the proof of proposition 3.1.1 (or equivalently proposition 6.2.1), we get:

$$\nu^* = \frac{y^* - X^*\beta - Z^*\Theta}{c_\alpha} = \begin{pmatrix} \frac{y - X\beta - Z\Theta}{c_\alpha} \\ -\frac{\sqrt{n\alpha_{2,1}}\beta}{c_\alpha} \\ -\frac{\sqrt{n\alpha_{2,2}}\Theta}{c_\alpha} \end{pmatrix} = \begin{pmatrix} \nu \\ -\frac{\sqrt{n\alpha_{2,1}}\beta}{c_\alpha} \\ -\frac{\sqrt{n\alpha_{2,2}}\Theta}{c_\alpha} \end{pmatrix} \tag{6.24}$$

So, we immediately get $\left|x_j^{*\top}\nu^*\right| = \left|x_j^\top \nu - \frac{n\alpha_{2,1}}{c_\alpha}\beta_j\right|$ and $\left|z_{jj}^{*\top}\nu^*\right| = \left|z_{jj}^\top \nu - \frac{n\alpha_{2,2}}{c_\alpha}\Theta_{jj}\right|$.
Moreover,

$$\left\|x_j^*\right\|_2 = \sqrt{\sum_{i=1}^{n+p+q}(x_{i,j}^*)^2} = \sqrt{\sum_{i=1}^{n}(x_{i,j})^2 + (\sqrt{n\alpha_{2,1}})^2 + 0} = \sqrt{\|x_j\|_2^2 + n\alpha_{2,1}} \tag{6.25}$$

Also, $\left\|z_j^*\right\|_2 = \sqrt{\|z_j\|_2^2 + n\alpha_{2,2}}$.
Finally, we get the following ranking rules:

$$d_j(\nu^*) = \frac{1 - \left|x_j^\top \nu - \frac{\alpha_{1,2}n}{c_\alpha}\beta_j\right|}{\sqrt{\|x_j\|_2^2 + \alpha_{1,2}n}} \text{ and } d_{jj}(\nu^*) = \frac{1 - \left|z_j^\top \nu - \frac{\alpha_{2,2}n}{c_\alpha}\Theta_{jj}\right|}{\sqrt{\|z_j\|_2^2 + \alpha_{2,2}n}} \tag{6.26}$$

□

Bibliography

Donald G Anderson. Iterative procedures for nonlinear integral equations. *Journal of the ACM (JACM)*, 12(4):547–560, 1965. 5, 64

F. Bascou, S. Lèbre, and J. Salmon. Debiasing the elastic net for models with interactions. In *Journées de Statistique*, 2020. 35

F. Bascou, S. Lèbre, and J. Salmon. Elasticnet avec gestion des interactions et débiaisage. In *EGC*, 2021. 35, 45

EML Beale, MG Kendall, and DW Mann. The discarding of variables in multivariate analysis. *Biometrika*, 54(3-4):357–366, 1967. 34

A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. Imaging Sci.*, 2(1):183–202, 2009. 64

A. Belloni and V. Chernozhukov. Least squares after model selection in high-dimensional sparse models. *Bernoulli*, 19(2):521–547, 2013. 4, 18

Q. Bertrand, Q. Klopfenstein, M. Blondel, S. Vaiter, A. Gramfort, and J. Salmon. Implicit differentiation of lasso-type models for hyperparameter optimization. In *ICML*, 2020. 50

Q. Bertrand, Q. Klopfenstein, P.-A. Bannier, G. Gidel, and M. Massias. Beyond l1: Faster and better sparse models with skglm, 2022. URL https://arxiv.org/abs/2204.07826. 5, 22, 65

Quentin Bertrand and Mathurin Massias. Anderson acceleration of coordinate descent. In *AISTATS*, pages 1288–1296. PMLR, 2021. 5, 65

Dimitris Bertsimas, Angela King, and Rahul Mazumder. Best subset selection via a modern optimization lens. *The annals of statistics*, 44(2):813–852, 2016. 34

Chloé Bessière, May Taha, Florent Petitprez, Jimmy Vandel, Jean-Michel Marin, Laurent Bréhélin, Sophie Lèbre, and Charles-Henri Lecellier. Probing instructions for

expression regulation in gene nucleotide compositions. *PLOS Computational Biology*, 14(1):1–28, 01 2018. 25, 26, 27, 84

J. Bien, J. Taylor, and R. Tibshirani. A lasso for hierarchical interactions. *Ann. Statist.*, 41(3):1111–1141, 2013. 2, 6, 12, 13

A. Boisbunon, R. Flamary, and A. Rakotomamonjy. Active set strategy for high-dimensional non-convex sparse optimization problems. In *ICASSP*, pages 1517–1521, 2014. 5, 22

A. Bonnefoy, V. Emiya, L. Ralaivola, and R. Gribonval. A dynamic screening principle for the lasso. In *EUSIPCO*, pages 6–10, 2014. 21

A. Bonnefoy, V. Emiya, L. Ralaivola, and R. Gribonval. Dynamic screening: accelerating first-order algorithms for the Lasso and Group-Lasso. *IEEE Trans. Signal Process.*, 63(19):20, 2015. 21

Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, Jonathan Eckstein, et al. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine learning*, 3(1):1–122, 2011. 13

L. Breiman. Random Forests. *Mach. Learn.*, 45(1):5–32, 2001. 8

Leo Breiman. Better subset regression using the nonnegative garrote. *Technometrics*, 37 (4):373–384, 1995. 12

P. Bühlmann and J. Mandozzi. High-dimensional variable screening and bisais in subsequent inference, with an empirical comparison. *Computational Statistics*, 29(3): 407–430, 2014. 39, 40

Eloi Chazelas, Mélanie Deschasaux, Bernard Srour, Emmanuelle Kesse-Guyot, Chantal Julia, Benjamin Alles, Nathalie Druesne-Pecollo, Pilar Galan, Serge Hercberg, Paule Latino-Martel, et al. Food additives: distribution and co-occurrence in 126,000 food products of the french market. *Scientific reports*, 10(1):1–15, 2020. 8

Eloi Chazelas, Nathalie Druesne-Pecollo, Younes Esseddik, Fabien Szabo de Edelenyi, Cédric Agaesse, Alexandre De Sa, Rebecca Lutchia, Pauline Rebouillat, Bernard Srour, Charlotte Debras, et al. Exposure to food additive mixtures in 106,000 french adults from the nutrinet-santé cohort. *Scientific Reports*, 11(1):1–21, 2021. 8

Jiahua Chen and Zehua Chen. Extended bayesian information criteria for model selection with large model spaces. *Biometrika*, 95(3):759–771, 2008. 76

S. S. Chen, D. L. Donoho, and M. A. Saunders. Atomic decomposition by basis pursuit. *SIAM J. Sci. Comput.*, 20(1):33–61, 1998. 1, 9

Hugh Chipman. Bayesian variable selection with related predictors. *Canadian Journal of Statistics*, 24(1):17–36, 1996. 11, 12

Gina M D'Angelo, D Chandrasekhra Rao, and C Charles Gu. Combining least absolute shrinkage and selection operator (lasso) and principal-components analysis for detection of gene-gene interactions in genome-wide association studies. In *BMC proceedings*, volume 3, pages 1–5. BioMed Central, 2009. 1, 8

C.-A. Deledalle, N. Papadakis, J. Salmon, and S. Vaiter. CLEAR: Covariant LEAst-square Re-fitting with applications to image restoration. *SIAM J. Imaging Sci.*, 10(1): 243–284, 2017. 4, 35, 45, 46

C.-A. Deledalle, N. Papadakis, and J. Salmon. On debiasing restoration algorithms: applications to total-variation and nonlocal-means. In *SSVM*, pages 129–141, 2015. 45

Charles-Alban Deledalle, Samuel Vaiter, Jalal Fadili, and Gabriel Peyré. Stein unbiased gradient estimator of the risk (sugar) for multiple parameter selection. *SIAM Journal on Imaging Sciences*, 7(4):2448–2487, 2014. 48

B. Efron, T. J. Hastie, I. M. Johnstone, and R. Tibshirani. Least angle regression. *Ann. Statist.*, 32(2):407–499, 2004. With discussion, and a rejoinder by the authors. 4, 18

L. El Ghaoui, V. Viallon, and T. Rabbani. Safe feature elimination in sparse supervised learning. *J. Pacific Optim.*, 8(4):667–698, 2012. 21

Jianqing Fan and Runze Li. Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American statistical Association*, 96(456):1348–1360, 2001. 34

R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. Liblinear: A library for large linear classification. *J. Mach. Learn. Res.*, 9:1871–1874, 2008. 5, 22

Yingying Fan, Yinfei Kong, Daoji Li, and Jinchi Lv. Interaction pursuit with feature screening and selection. *arXiv preprint arXiv:1605.08933*, 2016. 3, 16

O. Fercoq, A. Gramfort, and J. Salmon. Mind the duality gap: safer rules for the lasso. In *ICML*, pages 333–342, 2015. 21, 23

Jane C Figueiredo, Li Hsu, Carolyn M Hutter, Yi Lin, Peter T Campbell, John A Baron, Sonja I Berndt, Shuo Jiao, Graham Casey, Barbara Fortini, et al. Genome-wide diet-gene interaction analyses for risk of colorectal cancer. *PLoS genetics*, 10(4):e1004228, 2014. 1, 8

J. Friedman, T. J. Hastie, H. Höfling, and R. Tibshirani. Pathwise coordinate optimization. *Ann. Appl. Stat.*, 1(2):302–332, 2007. 3, 18, 19, 37

J. Friedman, T. J. Hastie, and R. Tibshirani. Regularization paths for generalized linear models via coordinate descent. *J. Stat. Softw.*, 33(1):1–22, 2010. 18, 19, 35, 37

W. J. Fu. Penalized regressions: the bridge versus the lasso. *JCGS*, 7(3):397–416, 1998. 18

Todd R Golub, Donna K Slonim, Pablo Tamayo, Christine Huard, Michelle Gaasenbeek, Jill P Mesirov, Hilary Coller, Mignon L Loh, James R Downing, Mark A Caligiuri, et al. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *science*, 286(5439):531–537, 1999. 10, 68

Michael Hamada and CF Jeff Wu. Analysis of designed experiments with complex aliasing. *Journal of quality technology*, 24(3):130–137, 1992. 11

Ning Hao and Hao Helen Zhang. Interaction screening for ultrahigh-dimensional data. *J. Amer. Statist. Assoc.*, 109(507):1285–1301, 2014. 2, 12, 13

Ning Hao, Yang Feng, and Hao Helen Zhang. Model selection for high-dimensional quadratic regression via regularization. *J. Amer. Statist. Assoc.*, 113(522):615–625, 2018. 2, 3, 6, 12, 14, 36

Asad Haris, Daniela Witten, and Noah Simon. Convex modeling of interactions with strong heredity. *Journal of Computational and Graphical Statistics*, 25(4):981–1004, Oct 2016. ISSN 1537-2715. doi: 10.1080/10618600.2015.1067217. 2, 12, 17

T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning : Data Mining, Inference, and Prediction. Second Edition.* Springer Series in Statistics. Springer, 2009. 17

Trevor Hastie, Robert Tibshirani, and Ryan Tibshirani. Best subset, forward stepwise or lasso? analysis and recommendations based on extensive comparisons. *Statistical Science*, 35(4):579–592, 2020. 34

Hussein Hazimeh and Rahul Mazumder. Learning hierarchical interactions at scale: A convex optimization approach. In *AISTATS*, pages 1833–1843. PMLR, 2020. 2, 3, 5, 12, 18, 24, 36, 37

Ronald R Hocking and RN Leslie. Selection of the best subset in regression analysis. *Technometrics*, 9(4):531–540, 1967. 34

A. E. Hoerl and R. W. Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970. 9

Cho-Jui Hsieh, Mátyás A Sustik, Inderjit S Dhillon, Pradeep Ravikumar, et al. Quic: quadratic approximation for sparse inverse covariance estimation. *J. Mach. Learn. Res.*, 15(1):2911–2947, 2014. 96

T. B. Johnson and C. Guestrin. Blitz: A principled meta-algorithm for scaling sparse optimization. In *ICML*, pages 1171–1179, 2015. 5, 22

J. Kim and H. Park. Fast active-set-type algorithms for l1-regularized linear regression. In *AISTATS*, pages 397–404, 2010. 5, 22

S.-J. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinevsky. An interior-point method for large-scale ℓ_1-regularized least squares. *IEEE J. Sel. Topics Signal Process.*, 1(4): 606–617, 2007. 20, 58, 99

K. Koh, S.-J. Kim, and S. Boyd. An interior-point method for large-scale l1-regularized logistic regression. *J. Mach. Learn. Res.*, 8(8):1519–1555, 2007. 96

Siu Kwan Lam, Antoine Pitrou, and Stanley Seibert. Numba: A llvm-based python jit compiler. In *Proceedings of the Second Workshop on the LLVM Compiler Infrastructure in HPC*, pages 1–6, 2015. 59, 66

Vincent Laville, Timothy Majarian, Paul S De Vries, Amy R Bentley, Mary F Feitosa, Yun J Sung, DC Rao, Alisa Manning, and Hugues Aschard. Deriving stratified effects from joint models investigating gene-environment interactions. *BMC bioinformatics*, 21(1):1–11, 2020. 1, 8

Marine Le Morvan and Jean-Philippe Vert. Whinter: A working set algorithm for high-dimensional sparse second order interaction models. In *ICML*, pages 3632–3641, 2018. 2, 5, 16, 24

J. Lederer. Trust, but verify: benefits and pitfalls of least-squares refitting in high dimensions. *ArXiv e-prints*, 2013. 4, 18

Michael Lim and Trevor Hastie. Learning interactions via hierarchical group-lasso regularization. *Journal of Computational and Graphical Statistics*, 24(3):627–654, 2015. 2, 12, 18, 24, 37

Jin Liu, Jian Huang, Yawei Zhang, Qing Lan, Nathaniel Rothman, Tongzhang Zheng, and Shuangge Ma. Identification of gene–environment interactions in cancer studies using penalization. *Genomics*, 102(4):189–194, 2013. 1, 8

Vien Mai and Mikael Johansson. Anderson acceleration of proximal gradient methods. In *ICML*, pages 6620–6629. PMLR, 2020. 5, 65

J. Mairal. *Sparse coding for machine learning, image processing and computer vision*. PhD thesis, École normale supérieure de Cachan, 2010. 58, 99

Jonathan Marchini, Peter Donnelly, and Lon R Cardon. Genome-wide strategies for detecting multiple loci that influence complex diseases. *Nature genetics*, 37(4):413–417, 2005. 1, 8

M. Massias, A. Gramfort, and J. Salmon. From safe screening rules to working sets for faster lasso-type solvers. In *NIPS-OPT*, 2017. 5, 22

M. Massias, A. Gramfort, and J. Salmon. Celer: a Fast Solver for the Lasso with Dual Extrapolation. In *ICML*, 2018. 5, 20, 22, 23, 57, 58, 101

M. Massias, S. Vaiter, A. Gramfort, and J. Salmon. Dual extrapolation for sparse generalized linear models. *J. Mach. Learn. Res.*, 21:1–33, 2020. 57, 96

Thomas Moreau, Mathurin Massias, Alexandre Gramfort, Pierre Ablin, Pierre-Antoine Bannier, Benjamin Charlier, Mathieu Dagréou, Tom Dupré la Tour, Ghislain Durif, Cassio F. Dantas, Quentin Klopfenstein, Johan Larsson, En Lai, Tanguy Lefort, Benoit Malézieux, Badr Moufad, Binh T. Nguyen, Alain Rakotomamonjy, Zaccharie Ramzi, Joseph Salmon, and Samuel Vaiter. Benchopt: Reproducible, efficient and collaborative optimization benchmarks, 2022. URL https://arxiv.org/abs/2206.13424. 5, 66

K. Nakagawa, S. Suzumura, M. Karasuyama, K. Tsuda, and I. Takeuchi. Safe feature pruning for sparse high-order interaction models. *ArXiv e-prints*, 2015. 2, 16, 24

K. Nakagawa, S. Suzumura, M. Karasuyama, K. Tsuda, and I. Takeuchi. Safe pattern pruning: An efficient approach for predictive pattern mining. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1785–1794. ACM, 2016. 2, 16, 24

E. Ndiaye, O. Fercoq, A. Gramfort, and J. Salmon. Gap safe screening rules for sparsity enforcing penalties. *J. Mach. Learn. Res.*, 18(128):1–33, 2017. 21, 23

J.A. Nelder. A reformulation of linear models. *Journal of the Royal Statistical Society: Series A (General)*, 140(1):48–63, 1977. 11

Y. Nesterov. A method for solving a convex programming problem with rate of convergence $O(1/k^2)$. *Soviet Math. Doklady*, 269(3):543–547, 1983. 64

G. Obozinski, B. Taskar, and M. I. Jordan. Joint covariate selection and joint subspace selection for multiple classification problems. *Statistics and Computing*, 20(2):231–252, 2010. 96

S. Osher, M. Burger, D. Goldfarb, J. Xu, and W. Yin. An iterative regularization method for total variation-based image restoration. *Multiscale Model. Simul.*, 4(2): 460–489, 2005. 45

Mee Young Park and Trevor Hastie. Penalized logistic regression for detecting gene interactions. *Biostatistics*, 9(1):30–50, 2008. 1, 2, 8, 12, 13

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. 35

Julio L Peixoto. Hierarchical variable selection in polynomial regression models. *The American Statistician*, 41(4):311–313, 1987. 11

P. Radchenko and G. M. James. Variable selection using adaptive nonlinear interaction structures in high dimensions. *J. Amer. Statist. Assoc.*, 105(492):1541–1553, 2010. 2, 12

Randall Reese, Xiaotian Dai, and Guifang Fu. Strong sure screening of ultra-high dimensional data with interaction effects. *arXiv preprint arXiv:1801.07785*, 2018. 3, 16

Marylyn D Ritchie, Lance W Hahn, Nady Roodi, L Renee Bailey, William D Dupont, Fritz F Parl, and Jason H Moore. Multifactor-dimensionality reduction reveals high-order interactions among estrogen-metabolism genes in sporadic breast cancer. *The American Journal of Human Genetics*, 69(1):138–147, 2001. 1, 8

J. Salmon. *On high dimensional regression: computational and statistical perspectives*. Habilitation à diriger des recherches, ENS Paris-Saclay, 2017. 17

Damien Scieur, Alexandre d'Aspremont, and Francis Bach. Regularized nonlinear acceleration. *Advances In Neural Information Processing Systems*, 29, 2016. 5, 65

Gian-Andrea Thanei, Nicolai Meinshausen, and Rajen D Shah. The xyz algorithm for fast interaction search in high-dimensional data. *Journal of Machine Learning Research*, 19:37, 2018. 12

R. Tibshirani. Regression shrinkage and selection via the lasso. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, 58(1):267–288, 1996. 1, 9

R. Tibshirani, J. Bien, J. Friedman, T. J. Hastie, N. Simon, J. Taylor, and R. J. Tibshirani. Strong rules for discarding predictors in lasso-type problems. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, 74(2):245–266, 2012. 15, 24

A. N. Tikhonov. On the stability of inverse problems. *Dokl. Akad. Nauk SSSR*, 39: 176–179, 1943. 9

P. Tseng. Convergence of a block coordinate descent method for nondifferentiable minimization. *J. Optim. Theory Appl.*, 109(3):475–494, 2001. 3, 18, 19

J. W. Tukey. *Exploratory data analysis*. Addison-Wesley Publishing Company, 1977. 45

J. Vandel, O. Cassan, S. Lèbre, C. H. Lecellier, and L. Bréhélin. Probing transcription factor combinatorics in different promoter classes and in enhancers. *BMC Genomics*, 20(1), 2019. 1, 2, 8, 15

Jie-Huei Wang and Yi-Hau Chen. Overlapping group screening for detection of gene-gene interactions: application to gene expression profiles with survival trait. *BMC bioinformatics*, 19(1):1–14, 2018. 1, 8

Lu Wang, Jincheng Shen, and Peter F Thall. A modified adaptive lasso for identifying interactions in the cox model with the heredity constraint. *Statistics & probability letters*, 93:126–133, 2014. 1, 8

Jing Wu, Bernie Devlin, Steven Ringquist, Massimo Trucco, and Kathryn Roeder. Screen and clean: a tool for identifying interactions in genome-wide association studies. *Genetic Epidemiology: The Official Publication of the International Genetic Epidemiology Society*, 34(3):275–285, 2010. 1, 8

Ming Yuan, V Roshan Joseph, and Hui Zou. Structured variable selection and estimation. *Ann. Appl. Stat.*, pages 1738–1757, 2009. 2, 12

Natalia Zemlianskaia, W James Gauderman, and Juan Pablo Lewinger. A scalable hierarchical lasso for gene–environment interactions. *Journal of Computational and Graphical Statistics*, pages 1–13, 2022. 1, 8

Cun-Hui Zhang. Nearly unbiased variable selection under minimax concave penalty. *The Annals of statistics*, 38(2):894–942, 2010. 34

Junzi Zhang, Brendan O'Donoghue, and Stephen Boyd. Globally convergent type-i anderson acceleration for nonsmooth fixed-point iterations. *SIAM Journal on Optimization*, 30(4):3170–3197, 2020. 5, 65

Fei Zhou, Jie Ren, Xi Lu, Shuangge Ma, and Cen Wu. Gene–environment interaction: A variable selection perspective. *Epistasis*, pages 191–223, 2021. 1, 8

H. Zou and T. J. Hastie. Regularization and variable selection via the elastic net. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, 67(2):301–320, 2005. 1, 3, 9, 34, 58, 97

Jan Zrimec, Filip Buric, Mariia Kokina, Victor Garcia, and Aleksej Zelezniak. Learning the regulatory code of gene expression. *Frontiers in Molecular Biosciences*, 8:673363, 2021. 1, 2, 8, 15