Literature Review

Sahir Rai Bhatnagar

July 21, 2018

It is easy to write more than you truly need, so try to keep it as limited as possible. (1) The problems of high dimension regressions such as overfitting and redundancy of variables. (2) then penalization generally as a solution to (1). You might I suppose mention very briefly the alternative solutions like apriori dimension reduction or forward selection, but I would spend as little time as possible on other methods (acknowledging their existence merely and saying your thesis focuses on penalization). (3) then L1 methods including lasso & group lasso (do you discuss elastic net? if so then you might need to include this too). Here I think you should be quite detailed in terms of the theory, how the penalization parameters are chosen, convergence, etc. (4) Then something about other structured L1 penalizations that have been proposed. There are quite a few examples of penalties built for specific applications, and maybe you could find a few such examples and cite them. Not comprehensively. Then you can point to the sail chapter as a new structured penalty. (5) A brief intro to linear mixed models followed by why naive penalization violates the normality of residuals to motivate your ggmix chapter. I'd stop there

1 High-dimensional regression methods

In this thesis, we consider the prediction of an outcome variable y observed on n individuals from p variables, where p is much larger than n. Challenges in this high-dimensional context include not only building a good predictor which will perform well in an independent dataset, but also being able to interpret the factors that contribute to the predictions. This latter issue can be very challenging in ultra-high dimensional predictor sets. For example, multiple different sets of covariates may provide equivalent measures of goodness of fit [1], and therefore how does one decide which are important? With the advent of high-throughput technologies in genomics and brain imaging studies, computational approaches to variable selection have become increasingly important. Broadly speaking, there are three main approaches to analyze high-dimensional data: 1) univariate regression followed by a multiple testing correction 2) multivariable penalized regression and 3) dimension reduction followed by a multivariable regression. We briefly introduce each of these analytic strategies below.

1.1 Univariate regression

Genome-wide association studies (GWAS) have become the standard method for analyzing genetic datasets. A GWAS consists of a series of univariate regressions followed by a multiple testing correction. This approach is simple and easy to implement, and has successfully identified thousands of genetic variants associated with complex diseases (https://www.genome.gov/gwastudies/). Despite these impressive findings, the discovered markers have only been able to explain a small proportion of the phenotypic variance known as the missing heritability problem [2]. One plausible explanation is that there are many causal variants that each explain a small amount of variation with small effect sizes [3]. GWAS are likely to miss these true associations due to the stringent significance thresholds required to reduce the number of false positives [2]. Most statistical methods for performing multiple testing adjustments assume weak dependence among the variables being tested [4]. Dependence among multiple tests can lead to incorrect Type 1 error rates [5] and highly variable significance measures [4]. Even in the presence of weakly dependent variables, adjusting for multiple tests in whole genome studies can result in low power. Furthermore, the univariate regression approach does not allow for modeling the joint effect of many variants which may be biologically more plausible [6]. In the next section, we introduce multivariable penalized regression approaches which have been proposed to address some of these limitations.

1.2 Multivariable penalized regression

For n observations and p covariates, consider the multiple linear regression model $\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$, where $\mathbf{Y}$ is a length n phenotype vector, $\mathbf{X}$ is an $n \times p$ design matrix, $\boldsymbol{\beta}$ is a p length coefficient vector and $\boldsymbol{\varepsilon}$ is an n length error vector. The least squares estimate is given by $\widehat{\boldsymbol{\beta}} = \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{Y}$. In high-dimensional data, the problem is that $\mathbf{X}^T\mathbf{X}$ is singular because the number of covariates greatly exceeds the number of subjects. For example DNA microarrays measure the expression of approximately 20,000 genes. However, due to funding constraints, the sample size is often less than a few hundred. A common solution to this problem is through penalized regression, i.e., apply a constraint on the values of $\boldsymbol{\beta}$. The problem can be formulated as finding the vector $\boldsymbol{\beta}$ that minimizes the penalized sum of squares:

$$\underbrace{\sum_{i=1}^{n}\left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j\right)^2}_{\text{goodness of fit}} + \underbrace{\sum_{j=1}^{p} p(\beta_j;\lambda,\gamma)}_{\text{penalty}} \tag{1}$$

The first term of (1) is the squared loss of the data and can be generalized to any loss function while the second term is a penalty which depends on non-negative tuning parameters λ and γ that control the amount of shrinkage to be applied to $\boldsymbol{\beta}$ and the degree of concavity of the penalty function, respectively. Several penalty terms have been developed in the literature. Ridge regression places a bound on the square of the coefficients (ℓ_2 penalty) [11] which has the effect of shrinking the magnitude of the coefficients. This however does not produce parsimonious models as none of the coefficients can be shrunk to exactly 0. The Lasso [12] overcomes this problem by placing a bound on the sum of the absolute values of the coefficients (ℓ_1 penalty) which sets some of them to 0, thereby simultaneously performing model selection. The Lasso, along with other forms of penalization (e.g. SCAD [13], Fused Lasso [14], Adaptive Lasso [15], Relaxed Lasso [16], MCP [17]) have proven successful in many practical problems. Despite these encouraging results, such methods have low sensitivity in the presence of high empirical correlations between covariates because only one variable tends to be selected from the group of correlated or nearly linearly dependent variables [18]. As a consequence, there is rarely consistency on which variable is chosen from one dataset to another (e.g. in cross-validation folds). This behavior is not well suited to genomic data in which large sets of predictors are highly correlated (e.g. a regulatory module) and are also associated with the response. The elastic net was proposed to benefit from the strengths of ridge regression's treatment of correlated variables and lasso's sparsity [19]. By placing both an ℓ_1 and ℓ_2 penalty on $\boldsymbol{\beta}$, the elastic net achieves model parsimony while yielding similar regression coefficients for correlated variables. These methods however do not take advantage of the grouping structure of the data. For example, cortical thickness measurements from magnetic resonance imaging (MRI) scans are often grouped into cortical regions of the Automated Anatomical Labelling (AAL) atlas [20]. Genes involved in the same cellular process (e.g. KEGG pathway [21]) can also be placed into biologically meaningful groups. When regularizing with the ℓ_1 penalty, each variable is selected individually regardless of its position in the design matrix. Existing structures between the variables (e.g. spatial, networks, pathways) are ignored even though in many real-life applications the estimation can benefit from this prior knowledge in terms of both prediction accuracy and interpretability [22]. The group lasso [23] (and generalizations thereof) overcomes this problem by producing a structured sparsity [22], i.e., given a predetermined grouping of non-overlapping variables, all members of the group are either zero or non-zero. The main drawback when applying these methods to genomic data is that these groups may not be known *a priori*. Known pathways may not be relevant to the response of interest and the study of inferring gene networks is still in its infancy.

1.3 Dimension reduction together with regression

Due to the unknown grouping problem, several authors have suggested a two-step procedure where they first cluster or group variables in the design matrix and then subsequently proceed to model fitting where the feature space is some summary measure of each group. This idea dates back to 1957 when Kendall [24] first proposed using principal components in regression. Hierarchical clustering based on the correlation of the design matrix has also been used to create groups of genes in microarray studies and for each level of hierarchy, the cluster average was used as the new set of potential predictors in forward-backward selection [25] or the lasso [26]. Bühlmann *et al.* [18] proposed a bottom-up agglomerative clustering algorithm based on canonical correlations and used the group lasso on the derived clusters. There are some advantages to these methods over the ones previously mentioned in Sections 1.1 and 1.2. First, the results can be more interpretable than the traditional lasso (and related methods) because the non-zero components of the prediction model represent sets of genes as opposed to individual ones. Second, by using genes which cluster well, we bias the inputs towards correlated sets of genes which are more likely to have similar function. Third, taking a summary measure of the resulting clusters can reduce the variance in prediction (overfitting) due to the compressed dimension of the feature space. Lastly, from a practical point of view this approach is flexible and easy to implement because efficient algorithms exist for both clustering [7] and model fitting [8, 9]. A limitation of these approaches is that the clustering is done in an unsupervised manner, i.e., the clusters do not use the response information. This has the effect of assigning similar coefficient values to correlated features. Witten *et* *al.* [27] recently proposed a method which encourages features that share an association with the response to take on similar coefficient values. This is useful in situations where only a fraction of the features in a cluster are associated with the response.

2 then L1 methods including lasso & group lasso (do you discuss elastic net? if so then you might need to include this too)

Here I think you should be quite detailed in terms of the theory, how the penalization parameters are chosen, convergence, etc.

3 Block Coordinate Descent Algorithm

We use a general purpose block coordinate descent algorithm (CGD) [31] to solve (**??**). At each iteration, the algorithm approximates the negative log-likelihood $f(\cdot)$ in $Q_\lambda(\cdot)$ by a strictly convex quadratic function and then applies block coordinate decent to generate a decent direction followed by an inexact line search along this direction [31]. For continuously differentiable $f(\cdot)$ and convex and block-separable $P(\cdot)$ (i.e. $P(\boldsymbol{\beta}) = \sum_i P_i(\beta_i)$), [31] show that the solution generated by the CGD method is a stationary point of $Q_\lambda(\cdot)$ if the coordinates are updated in a Gauss-Seidel manner i.e. $Q_\lambda(\cdot)$ is minimized with respect to one parameter while holding all others fixed. The CGD algorithm can thus be run in parallel and therefore suited for large p settings. It has been successfully applied in fixed effects models (e.g. [32], [8]) and [33] for mixed models with an ℓ_1 penalty. Following Tseng and Yun [31], the CGD algorithm is given by Algorithm 1.

The Armijo rule is defined as follows [31]:

Algorithm 1: Coordinate Gradient Descent Algorithm to solve (**??**)

```
Set the iteration counter $k \leftarrow 0$ and choose initial values for the parameter vector $\boldsymbol{\Theta}^{(0)}$;
repeat
    Approximate the Hessian $\nabla^2 f(\boldsymbol{\Theta}^{(k)})$ by a symmetric matix $H^{(k)}$:
        $H^{(k)} = \text{diag}\left[\min\left\{\max\left\{\left[\nabla^2 f(\boldsymbol{\Theta}^{(k)})\right]_{jj}, c_{min}\right\} c_{max}\right\}\right]_{j=1,\ldots,p+1}$    (2)
    for $j = 1, \ldots, p+1$ do
        Solve the descent direction $d^{(k)} := d_{H^{(k)}}(\Theta_j^{(k)})$ ;
        if $\Theta_j^{(k)} \in \{\beta_1, \ldots, \beta_p\}$ then
            $d_{H^{(k)}}(\Theta_j^{(k)}) \leftarrow \arg\min_d \left\{\nabla f(\Theta_j^{(k)})d + \frac{1}{2}d^2 H_{jj}^{(k)} + \lambda P(\Theta_j^{(k)} + d)\right\}$    (3)
        end
        if $\Theta_j^{(k)} \in \{\eta\}$ then
            $d_{H^{(k)}}(\Theta_j^{(k)}) \leftarrow -\nabla f(\Theta_j^{(k)}) / H_{jj}^{(k)}$    (4)
        end
        Choose a stepsize;
            $\alpha_j^{(k)} \leftarrow$ line search given by the Armijo rule
        Update;
            $\widehat{\Theta}_j^{(k+1)} \leftarrow \widehat{\Theta}_j^{(k)} + \alpha_j^{(k)} d^{(k)}$
    end
    Update;
        $\widehat{\eta}^{(k+1)} \leftarrow \arg\min_\eta \frac{1}{2}\sum_{i=1}^{N_T} \log(1 + \eta(\Lambda_i - 1)) + \frac{1}{2\sigma^{2\ (k)}} \sum_{i=1}^{N_T} \frac{\left(\widetilde{Y}_i - \sum_{j=0}^{p} \widetilde{X}_{ij+1}\beta_j^{(k+1)}\right)^2}{1 + \eta(\Lambda_i - 1)}$    (5)
    Update;
        $\widehat{\sigma^2}^{\ (k+1)} \leftarrow \frac{1}{N_T} \sum_{i=1}^{N_T} \frac{\left(\widetilde{Y}_i - \sum_{j=0}^{p} \widetilde{X}_{ij+1}\beta_j^{(k+1)}\right)^2}{1 + \eta^{(k+1)}(\Lambda_i - 1)}$    (6)
    $k \leftarrow k + 1$
until convergence criterion is satisfied;
```

Choose $\alpha_{init}^{(k)} > 0$ and let $\alpha^{(k)}$ be the largest element of $\left\{\alpha_{init}^{k}\delta^{r}\right\}_{r=0,1,2,\ldots}$ satisfying

$$Q_\lambda(\Theta_j^{(k)} + \alpha^{(k)} d^{(k)}) \leq Q_\lambda(\Theta_j^{(k)}) + \alpha^{(k)} \varrho \Delta^{(k)} \tag{7}$$

where $0 < \delta < 1$, $0 < \varrho < 1$, $0 \leq \gamma < 1$ and

$$\Delta^{(k)} := \nabla f(\Theta_j^{(k)}) d^{(k)} + \gamma (d^{(k)})^2 H_{jj}^{(k)} + \lambda P(\Theta_j^{(k)} + d^{(k)}) - \lambda P(\Theta^{(k)}) \tag{8}$$

Common choices for the constants are $\delta = 0.1$, $\varrho = 0.001$, $\gamma = 0$, $\alpha_{init}^{(k)} = 1$ for all k [33].

Below we detail the specifics of Algorithm 1 for the ℓ_1 penalty.

3.1 ℓ_1 penalty

The objective function is given by

$$Q_\lambda(\boldsymbol{\Theta}) = f(\boldsymbol{\Theta}) + \lambda |\boldsymbol{\beta}| \tag{9}$$

3.1.1 Descent Direction

For simplicity, we remove the iteration counter (k) from the derivation below.

For $\Theta_j^{(k)} \in \{\beta_1, \ldots, \beta_p\}$, let

$$d_H(\Theta_j) = \arg\min_d G(d) \tag{10}$$

where

$$G(d) = \nabla f(\Theta_j) d + \frac{1}{2} d^2 H_{jj} + \lambda |\Theta_j + d|$$

Since $G(d)$ is not differentiable at $-\Theta_j$, we calculate the subdifferential $\partial G(d)$ and search for d with $0 \in \partial G(d)$:

$$\partial G(d) = \nabla f(\Theta_j) + d H_{jj} + \lambda u \tag{11}$$

where

$$u = \begin{cases} 1 & \text{if} \quad d > -\Theta_j \\ -1 & \text{if} \quad d < -\Theta_j \\ [-1, 1] & \text{if} \quad d = \Theta_j \end{cases} \tag{12}$$

We consider each of the three cases in (11) below

1. $d > -\Theta_j$

   $$\partial G(d) = \nabla f(\Theta_j) + dH_{jj} + \lambda = 0$$
   $$d = \frac{-(\nabla f(\Theta_j) + \lambda)}{H_{jj}}$$

   Since $\lambda > 0$ and $H_{jj} > 0$, we have

   $$\frac{-(\nabla f(\Theta_j) - \lambda)}{H_{jj}} > \frac{-(\nabla f(\Theta_j) + \lambda)}{H_{jj}} = d \overset{\text{def}}{>} -\Theta_j$$

   The solution can be written compactly as

   $$d = \text{mid}\left\{\frac{-(\nabla f(\Theta_j) - \lambda)}{H_{jj}}, -\Theta_j, \frac{-(\nabla f(\Theta_j) + \lambda)}{H_{jj}}\right\}$$

   where mid $\{a, b, c\}$ denotes the median (mid-point) of a, b, c [31].

2. $d < -\Theta_j$

   $$\partial G(d) = \nabla f(\Theta_j) + dH_{jj} - \lambda = 0$$
   $$d = \frac{-(\nabla f(\Theta_j) - \lambda)}{H_{jj}}$$

   Since $\lambda > 0$ and $H_{jj} > 0$, we have

   $$\frac{-(\nabla f(\Theta_j) + \lambda)}{H_{jj}} < \frac{-(\nabla f(\Theta_j) - \lambda)}{H_{jj}} = d \overset{\text{def}}{<} -\Theta_j$$

Again, the solution can be written compactly as

$$d = \text{mid}\left\{\frac{-(\nabla f(\Theta_j) - \lambda)}{H_{jj}}, -\Theta_j, \frac{-(\nabla f(\Theta_j) + \lambda)}{H_{jj}}\right\}$$

3. $d_j = -\Theta_j$

   There exists $u \in [-1, 1]$ such that

$$\partial G(d) = \nabla f(\Theta_j) + dH_{jj} + \lambda u = 0$$
$$d = \frac{-(\nabla f(\Theta_j) + \lambda u)}{H_{jj}}$$

   For $-1 \leq u \leq 1$, $\lambda > 0$ and $H_{jj} > 0$ we have

$$\frac{-(\nabla f(\Theta_j) + \lambda)}{H_{jj}} \leq d \overset{\text{def}}{=} -\Theta_j \leq \frac{-(\nabla f(\Theta_j) - \lambda)}{H_{jj}}$$

   The solution can again be written compactly as

$$d = \text{mid}\left\{\frac{-(\nabla f(\Theta_j) - \lambda)}{H_{jj}}, -\Theta_j, \frac{-(\nabla f(\Theta_j) + \lambda)}{H_{jj}}\right\}$$

We see all three cases lead to the same solution for (10). Therefore the descent direction for $\Theta_j^{(k)} \in \{\beta_1, \ldots, \beta_p\}$ for the ℓ_1 penalty is given by

$$d = \text{mid}\left\{\frac{-(\nabla f(\beta_j) - \lambda)}{H_{jj}}, -\beta_j, \frac{-(\nabla f(\beta_j) + \lambda)}{H_{jj}}\right\} \tag{13}$$

3.1.2 Solution for the β parameter

If the Hessian $\nabla^2 f(\boldsymbol{\Theta}^{(k)}) > 0$ then $H^{(k)}$ defined in (2) is equal to $\nabla^2 f(\boldsymbol{\Theta}^{(k)})$. Using $\alpha_{init} = 1$, the largest element of $\left\{\alpha_{init}^{(k)} \delta^r\right\}_{r=0,1,2,\ldots}$ satisfying the Armijo Rule inequality is reached for

$\alpha^{(k)} = \alpha_{init}^{(k)}\delta^0 = 1$. The Armijo rule update for the $\boldsymbol{\beta}$ parameter is then given by

$$\beta_j^{(k+1)} \leftarrow \beta_j^{(k)} + d^{(k)}, \qquad j = 1, \ldots, p \tag{14}$$

Substituting the descent direction given by (13) into (14) we get

$$\beta_j^{(k+1)} = \text{mid}\left\{\beta_j^{(k)} + \frac{-(\nabla f(\beta_j^{(k)}) - \lambda)}{H_{jj}}, 0, \beta_j^{(k)} + \frac{-(\nabla f(\beta_j^{(k)}) + \lambda)}{H_{jj}}\right\} \tag{15}$$

We can further simplify this expression. Let

$$w_i := \frac{1}{\sigma^2\left(1 + \eta(\Lambda_i - 1)\right)} \tag{16}$$

.

Re-write the part depending on $\boldsymbol{\beta}$ of the negative log-likelihood in (**??**) as

$$g(\boldsymbol{\beta}^{(k)}) = \frac{1}{2}\sum_{i=1}^{N_T} w_i \left(\widetilde{Y}_i - \sum_{\ell \neq j} \widetilde{X}_{i\ell}\beta_\ell^{(k)} - \widetilde{X}_{ij}\beta_j^{(k)}\right)^2 \tag{17}$$

The gradient and Hessian are given by

$$\nabla f(\beta_j^{(k)}) := \frac{\partial}{\partial \beta_j^{(k)}} g(\boldsymbol{\beta}^{(k)}) = -\sum_{i=1}^{N_T} w_i \widetilde{X}_{ij}\left(\widetilde{Y}_i - \sum_{\ell \neq j} \widetilde{X}_{i\ell}\beta_\ell^{(k)} - \widetilde{X}_{ij}\beta_j^{(k)}\right) \tag{18}$$

$$H_{jj} := \frac{\partial^2}{\partial {\beta_j^{(k)}}^2} g(\boldsymbol{\beta}^{(k)}) = \sum_{i=1}^{N_T} w_i \widetilde{X}_{ij}^2 \tag{19}$$

Substituting (18) and (19) into $\beta_j^{(k)} + \frac{-(\nabla f(\beta_j^{(k)})-\lambda)}{H_{jj}}$

$$\begin{aligned}
&\beta_j^{(k)} + \frac{\sum_{i=1}^{N_T} w_i \widetilde{X}_{ij} \left(\widetilde{Y}_i - \sum_{\ell \neq j} \widetilde{X}_{i\ell} \beta_\ell^{(k)} - \widetilde{X}_{ij} \beta_j^{(k)} \right) + \lambda}{\sum_{i=1}^{N_T} w_i \widetilde{X}_{ij}^2} \\
&= \beta_j^{(k)} + \frac{\sum_{i=1}^{N_T} w_i \widetilde{X}_{ij} \left(\widetilde{Y}_i - \sum_{\ell \neq j} \widetilde{X}_{i\ell} \beta_\ell^{(k)} \right) + \lambda}{\sum_{i=1}^{N_T} w_i \widetilde{X}_{ij}^2} - \frac{\sum_{i=1}^{N_T} w_i \widetilde{X}_{ij}^2 \beta_j^{(k)}}{\sum_{i=1}^{N_T} w_i \widetilde{X}_{ij}^2} \\
&= \frac{\sum_{i=1}^{N_T} w_i \widetilde{X}_{ij} \left(\widetilde{Y}_i - \sum_{\ell \neq j} \widetilde{X}_{i\ell} \beta_\ell^{(k)} \right) + \lambda}{\sum_{i=1}^{N_T} w_i \widetilde{X}_{ij}^2}
\end{aligned} \tag{20}$$

Similarly, substituting (18) and (19) in $\beta_j^{(k)} + \frac{-(\nabla f(\beta_j^{(k)})+\lambda)}{H_{jj}}$ we get

$$\frac{\sum_{i=1}^{N_T} w_i \widetilde{X}_{ij} \left(\widetilde{Y}_i - \sum_{\ell \neq j} \widetilde{X}_{i\ell} \beta_\ell^{(k)} \right) - \lambda}{\sum_{i=1}^{N_T} w_i \widetilde{X}_{ij}^2} \tag{21}$$

Finally, substituting (20) and (21) into (15) we get

$$\begin{aligned}
\beta_j^{(k+1)} &= \operatorname{mid} \left\{ \frac{\sum_{i=1}^{N_T} w_i \widetilde{X}_{ij} \left(\widetilde{Y}_i - \sum_{\ell \neq j} \widetilde{X}_{i\ell} \beta_\ell^{(k)} \right) - \lambda}{\sum_{i=1}^{N_T} w_i \widetilde{X}_{ij}^2}, 0, \frac{\sum_{i=1}^{N_T} w_i \widetilde{X}_{ij} \left(\widetilde{Y}_i - \sum_{\ell \neq j} \widetilde{X}_{i\ell} \beta_\ell^{(k)} \right) + \lambda}{\sum_{i=1}^{N_T} w_i \widetilde{X}_{ij}^2} \right\} \\
&= \frac{\mathcal{S}_\lambda \left(\sum_{i=1}^{N_T} w_i \widetilde{X}_{ij} \left(\widetilde{Y}_i - \sum_{\ell \neq j} \widetilde{X}_{i\ell} \beta_\ell^{(k)} \right) \right)}{\sum_{i=1}^{N_T} w_i \widetilde{X}_{ij}^2}
\end{aligned} \tag{22}$$

Where $\mathcal{S}_\lambda(x)$ is the soft-thresholding operator

$$\mathcal{S}_\lambda(x) = \operatorname{sign}(x)(|x| - \lambda)_+$$

sign(x) is the signum function

$$\text{sign}(x) = \begin{cases} -1 & x < 0 \\ 0 & x = 0 \\ 1 & x > 0 \end{cases}$$

and $(x)_+ = \max(x, 0)$.

We note that the parameter update for β_j given by (22) takes the same form as the weighted updates of the `glmnet` algorithm [8] (Section 2.4, equation (10)) with $\alpha = 1$.

4 Group Lasso with Low-rank Similarity Matrix

This section focuses on the part of the log-likelihood (**??**) that depends on $\boldsymbol{\beta}$.

This description follows mainly Yang and Zou (2015). We add the weight matrix.

4.1 Model

Only the third term of the log-likelihood (**??**) depends on $\boldsymbol{\beta}$:

$$\frac{1}{2}\left\{(\mathbf{Y}-\mathbf{X}\boldsymbol{\beta})^T\left[\frac{1}{\sigma^2(1-\eta)}\left(\mathbf{I}_{N_T}-\mathbf{U}_1\left(\frac{1-\eta}{\eta}\boldsymbol{\Sigma}_1^{-1}+\mathbf{I}_k\right)^{-1}\mathbf{U}_1^T\right)\right](\mathbf{Y}-\mathbf{X}\boldsymbol{\beta})\right\} \tag{23}$$

Equation (23) can be written more generally as

$$L(\boldsymbol{\beta} \mid \mathbf{D}) = \frac{1}{2}\left[\mathbf{Y}-\widehat{\mathbf{Y}}\right]^\top \mathbf{W}\left[\mathbf{Y}-\widehat{\mathbf{Y}}\right]$$

where $\widehat{\mathbf{Y}} = \sum_{j=1}^{p} \beta_j X_j$, $\mathbf{D}$ is the working data $\{\mathbf{Y}, \mathbf{X}\}$, and $\mathbf{W}$ is an $N_T \times N_T$ weight matrix given by

$$\mathbf{W} = \frac{1}{\sigma^2(1-\eta)} \left(\mathbf{I}_{N_T} - \mathbf{U}_1 \left(\frac{1-\eta}{\eta} \mathbf{\Sigma}_1^{-1} + \mathbf{I}_k \right)^{-1} \mathbf{U}_1^T \right) \tag{24}$$

Assume that we the predictors in the design matrix $\mathbf{X} \in \mathbb{R}^{N_T \times p}$ belong to K groups and that the group membership is already defined such that $(1, 2, \ldots, p) = \bigcup_{k=1}^{K} I_k$ and the cardinality of index set I_k is p_k, $I_k \bigcap I_{k'} = \emptyset$ for $k \neq k', 1 \leq k, k' \leq K$. Thus group k contains p_k predictors, which are x_j's for $j \in I_k$, and $1 \leq k \leq K$. If an intercept is included, then $I_1 = \{1\}$. Given the group partition, we use $\boldsymbol{\beta}_{(k)}$ to denote the segment of $\boldsymbol{\beta}$ corresponding to group k. This notation is used for any p-dimensional vector. We consider the group lasso penalized estimator

$$\min_{\boldsymbol{\beta}} L(\boldsymbol{\beta} \mid \mathbf{D}) + \lambda \sum_{k=1}^{K} w_k \|\boldsymbol{\beta}_{(k)}\|_2, \tag{25}$$

The loss function L satisfies the quadratic majorization (QM) condition, since there exists a $p \times p$ matrix $\mathbf{H} = \mathbf{X}^\top \mathbf{W} \mathbf{X}$, and $\nabla L(\boldsymbol{\beta}|\mathbf{D}) = -\left(Y - \hat{Y}\right)^\top \mathbf{W}\mathbf{X}$, which may only depend on the data $\mathbf{D}$, such that for all $\boldsymbol{\beta}, \boldsymbol{\beta}^*$,

$$L(\boldsymbol{\beta} \mid \mathbf{D}) \leq L(\boldsymbol{\beta}^* \mid \mathbf{D}) + (\boldsymbol{\beta} - \boldsymbol{\beta}^*)^\top \nabla L(\boldsymbol{\beta}^*|\mathbf{D}) + \frac{1}{2}(\boldsymbol{\beta} - \boldsymbol{\beta}^*)^\top \mathbf{H} (\boldsymbol{\beta} - \boldsymbol{\beta}^*). \tag{26}$$

4.2 Algorithm

Noticing that the penalty term $\sum_{k=1}^{K} w_k ||\boldsymbol{\beta}_{(k)}||_2$ is separable with respect to the indices of the features $k = 1, \ldots, K$, we can derive the *groupwise-majorization-descent* (GMD) algorithm for computing the solution of (25) when the loss function satisfies the QM condition. Let $\widetilde{\boldsymbol{\beta}}$ denote the current solution of $\boldsymbol{\beta}$. Without loss of generality, let us derive the GMD update of $\widetilde{\boldsymbol{\beta}}_{(k)}$, the coefficients of group k. Define $\mathbf{H}_k$ as the sub-matrix of $\mathbf{H}$ corresponding to group

k. For example, if group 2 is $\{2,4\}$ then $\mathbf{H}_{(2)}$ is a 2×2 matrix with

$$\mathbf{H}_{(2)} = \begin{bmatrix} h_{2,2} & h_{2,4} \\ h_{4,2} & h_{4,4} \end{bmatrix},$$

where $h_{i,j}$ is the i,jth entry of the $\mathbf{H}$ matrix. Write $\boldsymbol{\beta}$ such that $\boldsymbol{\beta}_{(k')} = \widetilde{\boldsymbol{\beta}}_{(k')}$ for $k' \neq k$. Given $\boldsymbol{\beta}_{(k')} = \widetilde{\boldsymbol{\beta}}_{(k')}$ for $k' \neq k$, the optimal $\boldsymbol{\beta}_{(k)}$ is defined as

$$\arg\min_{\boldsymbol{\beta}^{(k)}} L(\boldsymbol{\beta} \mid \mathbf{D}) + \lambda w_k \|\boldsymbol{\beta}_{(k)}\|_2. \tag{27}$$

Unfortunately, there is no closed form solution to (27) for a general loss function with general design matrix. We overcome the computational obstacle by taking advantage of the QM condition. From (26) we have

$$L(\boldsymbol{\beta} \mid \mathbf{D}) \leq L(\widetilde{\boldsymbol{\beta}} \mid \mathbf{D}) + (\boldsymbol{\beta} - \widetilde{\boldsymbol{\beta}})^\top \nabla L(\widetilde{\boldsymbol{\beta}}|\mathbf{D}) + \frac{1}{2}(\boldsymbol{\beta} - \widetilde{\boldsymbol{\beta}})^\top \mathbf{H}(\boldsymbol{\beta} - \widetilde{\boldsymbol{\beta}}).$$

Write $U(\widetilde{\boldsymbol{\beta}}) = -\nabla L(\widetilde{\boldsymbol{\beta}}|\mathbf{D})$. Using

$$\boldsymbol{\beta} - \widetilde{\boldsymbol{\beta}} = (\underbrace{0,\ldots,0}_{k-1}, \boldsymbol{\beta}_{(k)} - \widetilde{\boldsymbol{\beta}}_{(k)}, \underbrace{0,\ldots,0}_{K-k}),$$

we can write

$$L(\boldsymbol{\beta} \mid \mathbf{D}) \leq L(\widetilde{\boldsymbol{\beta}} \mid \mathbf{D}) - (\boldsymbol{\beta}_{(k)} - \widetilde{\boldsymbol{\beta}}_{(k)})^\top U_{(k)} + \frac{1}{2}(\boldsymbol{\beta}_{(k)} - \widetilde{\boldsymbol{\beta}}_{(k)})^\top \mathbf{H}_{(k)}(\boldsymbol{\beta}_{(k)} - \widetilde{\boldsymbol{\beta}}_{(k)}). \tag{28}$$

where

$$U_{(k)} = \frac{\partial}{\partial \boldsymbol{\beta}_{(k)}} L_Q(\boldsymbol{\beta} \mid \mathbf{D}) = -\left(Y - \hat{Y}\right)^\top \mathbf{W}\mathbf{X}_{(k)}, \tag{29}$$

$$\mathbf{H}_{(k)} = \frac{\partial^2}{\partial \boldsymbol{\beta}_{(k)} \partial \boldsymbol{\beta}_{(k)}^\top} L_Q(\boldsymbol{\beta} \mid \mathbf{D}) = \mathbf{X}_{(k)}^\top \mathbf{W}\mathbf{X}_{(k)}. \tag{30}$$

Let η_k be the largest eigenvalue of $\mathbf{H}_{(k)}$. We set $\gamma_k = (1+\varepsilon^*)\eta_k$, where $\varepsilon^* = 10^{-6}$. Then we can further relax the upper bound in (28) as

$$L(\boldsymbol{\beta} \mid \mathbf{D}) \leq L(\widetilde{\boldsymbol{\beta}} \mid \mathbf{D}) - (\boldsymbol{\beta}^{(k)} - \widetilde{\boldsymbol{\beta}}^{(k)})^\top U_{(k)} + \frac{1}{2}\gamma_k(\boldsymbol{\beta}^{(k)} - \widetilde{\boldsymbol{\beta}}^{(k)})^\top(\boldsymbol{\beta}^{(k)} - \widetilde{\boldsymbol{\beta}}^{(k)}). \tag{31}$$

It is important to note that the inequality strictly holds unless for $\boldsymbol{\beta}^{(k)} = \widetilde{\boldsymbol{\beta}}^{(k)}$. Instead of minimizing (27) we solve

$$\arg\min_{\boldsymbol{\beta}^{(k)}} L(\widetilde{\boldsymbol{\beta}} \mid \mathbf{D}) - (\boldsymbol{\beta}^{(k)} - \widetilde{\boldsymbol{\beta}}^{(k)})^\top U_{(k)} + \frac{1}{2}\gamma_k(\boldsymbol{\beta}^{(k)} - \widetilde{\boldsymbol{\beta}}^{(k)})^\top(\boldsymbol{\beta}^{(k)} - \widetilde{\boldsymbol{\beta}}^{(k)}) + \lambda w_k \|\boldsymbol{\beta}^{(k)}\|_2. \tag{32}$$

Denote by $\widetilde{\boldsymbol{\beta}}^{(k)}(\text{new})$ the solution to (32). It is straightforward to see that $\widetilde{\boldsymbol{\beta}}^{(k)}(\text{new})$ has a simple closed-from expression

$$\widetilde{\boldsymbol{\beta}}^{(k)}(\text{new}) = \frac{1}{\gamma_k}\left(U^{(k)} + \gamma_k \widetilde{\boldsymbol{\beta}}^{(k)}\right)\left(1 - \frac{\lambda w_k}{\|U^{(k)} + \gamma_k \widetilde{\boldsymbol{\beta}}^{(k)}\|_2}\right)_+. \tag{33}$$

Algorithm 2 summarizes the details of GMD.

Algorithm 2: The GMD algorithm for general group-lasso learning.

```
1. For $k = 1, \ldots, K$, compute $\gamma_k$, the largest eigenvalue of $\mathbf{H}^{(k)}$.
2. Initialize $\widetilde{\boldsymbol{\beta}}$.
3. Repeat the following cyclic groupwise updates until convergence:
    — for $k = 1, \ldots, K$, do step (3.1)–(3.3)
        3.1 Compute $U(\widetilde{\boldsymbol{\beta}}) = -\nabla L(\widetilde{\boldsymbol{\beta}}|\mathbf{D})$.
        3.2 Compute $\widetilde{\boldsymbol{\beta}}^{(k)}(\text{new}) = \frac{1}{\gamma_k}\left(U^{(k)} + \gamma_k \widetilde{\boldsymbol{\beta}}^{(k)}\right)\left(1 - \frac{\lambda w_k}{\|U^{(k)} + \gamma_k \widetilde{\boldsymbol{\beta}}^{(k)}\|_2}\right)_+$.
        3.3 Set $\widetilde{\boldsymbol{\beta}}^{(k)} = \widetilde{\boldsymbol{\beta}}^{(k)}(\text{new})$.
```

4.3 Convergence

We can prove the strict descent property of GMD by using the MM principle [34, 35, 36]. Define

$$Q(\boldsymbol{\beta} \mid \mathbf{D}) = L(\widetilde{\boldsymbol{\beta}} \mid \mathbf{D}) - (\boldsymbol{\beta}^{(k)} - \widetilde{\boldsymbol{\beta}}^{(k)})^\top U^{(k)} + \frac{1}{2}\gamma_k(\boldsymbol{\beta}^{(k)} - \widetilde{\boldsymbol{\beta}}^{(k)})^\top(\boldsymbol{\beta}^{(k)} - \widetilde{\boldsymbol{\beta}}^{(k)}) + \lambda w_k \|\boldsymbol{\beta}^{(k)}\|_2. \quad (34)$$

Obviously, $Q(\boldsymbol{\beta} \mid \mathbf{D}) = L(\boldsymbol{\beta} \mid \mathbf{D}) + \lambda w_k \|\boldsymbol{\beta}^{(k)}\|_2$ when $\boldsymbol{\beta}^{(k)} = \widetilde{\boldsymbol{\beta}}^{(k)}$ and (**??**) shows that $Q(\boldsymbol{\beta} \mid \mathbf{D}) > L(\boldsymbol{\beta} \mid \mathbf{D}) + \lambda w_k \|\boldsymbol{\beta}^{(k)}\|_2$ when $\boldsymbol{\beta}^{(k)} \neq \widetilde{\boldsymbol{\beta}}^{(k)}$. After updating $\widetilde{\boldsymbol{\beta}}^{(k)}$ using (**??**), we have

$$\begin{aligned} L(\widetilde{\boldsymbol{\beta}}^{(k)}(\text{new}) \mid \mathbf{D}) + \lambda w_k \|\widetilde{\boldsymbol{\beta}}^{(k)}(\text{new})\|_2 &\leq Q(\widetilde{\boldsymbol{\beta}}^{(k)}(\text{new}) \mid \mathbf{D}) \\ &\leq Q(\widetilde{\boldsymbol{\beta}} \mid \mathbf{D}) \\ &= L(\widetilde{\boldsymbol{\beta}} \mid \mathbf{D}) + \lambda w_k \|\widetilde{\boldsymbol{\beta}}^{(k)}\|_2. \end{aligned}$$

Moreover, if $\widetilde{\boldsymbol{\beta}}^{(k)}(\text{new}) \neq \widetilde{\boldsymbol{\beta}}^{(k)}$, then the first inequality becomes

$$L(\widetilde{\boldsymbol{\beta}}^{(k)}(\text{new}) \mid \mathbf{D}) + \lambda w_k \|\widetilde{\boldsymbol{\beta}}^{(k)}(\text{new})\|_2 \; < \; Q(\widetilde{\boldsymbol{\beta}}^{(k)}(\text{new}) \mid \mathbf{D}).$$

Therefore, the objective function is strictly decreased after updating all groups in a cycle, unless the solution does not change after each groupwise update. If this is the case, we can show that the solution must satisfy the KKT conditions, which means that the algorithm converges and finds the right answer. To see this, if $\widetilde{\boldsymbol{\beta}}^{(k)}(\text{new}) = \widetilde{\boldsymbol{\beta}}^{(k)}$ for all k, then by the update formula (33) we have that for all k

$$\widetilde{\boldsymbol{\beta}}^{(k)} = \frac{1}{\gamma_k}\left(U^{(k)} + \gamma_k \widetilde{\boldsymbol{\beta}}^{(k)}\right)\left(1 - \frac{\lambda w_k}{\|U^{(k)} + \gamma_k \widetilde{\boldsymbol{\beta}}^{(k)}\|_2}\right) \qquad \text{if } \|U^{(k)} + \gamma_k \widetilde{\boldsymbol{\beta}}^{(k)}\|_2 > \lambda w_k, \tag{35}$$

$$\widetilde{\boldsymbol{\beta}}^{(k)} = \mathbf{0} \qquad \text{if } \|U^{(k)} + \gamma_k \widetilde{\boldsymbol{\beta}}^{(k)}\|_2 \leq \lambda w_k. \tag{36}$$

By straightforward algebra we obtain the KKT conditions:

$$\begin{aligned} -U^{(k)} + \lambda w_k \cdot \frac{\widetilde{\boldsymbol{\beta}}^{(k)}}{\|\widetilde{\boldsymbol{\beta}}^{(k)}\|_2} &= \mathbf{0} \qquad \text{if } \widetilde{\boldsymbol{\beta}}^{(k)} \neq \mathbf{0}, \\ \left\|U^{(k)}\right\|_2 &\leq \lambda w_k \qquad \text{if } \widetilde{\boldsymbol{\beta}}^{(k)} = \mathbf{0}, \end{aligned}$$

where $k = 1, 2, \ldots, K$. Therefore, if the objective function stays unchanged after a cycle, the algorithm necessarily converges to the right answer.

5 Then something about other structured L1 penalizations that have been proposed

There are quite a few examples of penalties built for specific applications, and maybe you could find a few such examples and cite them. Not comprehensively. Then you can point to the sail chapter as a new structured penalty.

Type	Method	Software
Linear	CAP [37]	✗
	SHIM [38]	✗
	hiernet [30]	hierNet(x, y)
	GRESH [39]	✗
	FAMILY [40]	FAMILY(x, z, y)
	glinternet [41]	glinternet(x, y)
	RAMP [42]	RAMP(x, y)
	LassoBacktracking [43]	LassoBT(x, y)
Non-linear	VANISH [44]	✗
	sail	sail(x, y, e)

5.1 Current methods overview and their limitations

We consider a regression model for an outcome variable $Y = (y_1, \ldots, y_n)$ which follows an exponential family. Let $E = (e_1, \ldots, e_n)$ be the binary environment vector and $\boldsymbol{x} = (X_1, \ldots, X_p)$ be the matrix of high-dimensional data. Consider the regression model with main effects and their interactions with E:

$$g(\boldsymbol{\mu}) = \beta_0 + \underbrace{\beta_1 X_1 + \cdots + \beta_p X_p + \beta_E E}_{\text{main effects}} + \underbrace{\alpha_{1E}(X_1 E) + \cdots + \alpha_{pE}(X_p E)}_{\text{interactions}} \tag{37}$$

where $g(\cdot)$ is a known link function and $\boldsymbol{\mu} = \mathsf{E}\left[Y|\boldsymbol{x}, E, \boldsymbol{\beta}, \boldsymbol{\alpha}\right]$. Our goal is to estimate the parameters $\boldsymbol{\beta} = (\beta_1, \beta_2, \ldots, \beta_p, \beta_E) \in \mathbb{R}^{p+1}$ and $\boldsymbol{\alpha} = (\alpha_{1E}, \ldots, \alpha_{pE}) \in \mathbb{R}^p$ and to improve prediction of Y. In fact, in the light of our goals to improve prediction and interpretability, we also consider the related model

$$g(\boldsymbol{\mu}) = \beta_0^* + \sum_{k=1}^{q} \beta_k^* \widetilde{X}_k + \beta_E^* E + \sum_{k=1}^{q} \alpha_k^* E \widetilde{X}_k \tag{38}$$

where $\widetilde{X}_k, k = 1, \ldots, q$ are linear combinations of X designed to reduce the dimension, such that $q << p$, and the superscript asterisk on the parameters is just to emphasize that these are different from those in (37). In what follows, we omit the asterisk on the parameters for clarity.

5.2 Phase 3: Variable Selection

We are interested in imposing the strong heredity principle [45]:

$$\hat{\alpha}_{jE} \neq 0 \qquad \Rightarrow \qquad \hat{\beta}_j \neq 0 \qquad \text{and} \qquad \hat{\beta}_E \neq 0 \tag{39}$$

In words, the interaction term will only have a non-zero estimate if its corresponding main effects are estimated to be non-zero. One benefit brought by hierarchy is that the number of measured variables can be reduced, referred to as practical sparsity [30, 39]. For example, a model involving $X_1, E, X_1 \cdot E$ is more parsimonious than a model involving $X_1, E, X_2 \cdot E$, because in the first model a researcher would only have to measure two variables compared to three in the second model. In order to address these issues, we propose to extend the model of Choi *et al.* [38] to simultaneously perform variable selection, estimation and impose the strong heredity principle in the context of high dimensional interactions with the environment (HD×E). To do so, we follow Choi and reparametrize the coefficients for the interaction terms as $\alpha_{jE} = \gamma_{jE}\beta_j\beta_E$. Plugging this into (37):

$$g(\boldsymbol{\mu}) = \beta_0 + \beta_1\widetilde{X}_1 + \cdots + \beta_q\widetilde{X}_q + \beta_E E + \gamma_{1E}\beta_1\beta_E(\widetilde{X}_1 E) + \cdots + \gamma_{qE}\beta_q\beta_E(\widetilde{X}_q E) \tag{40}$$

where $\widetilde{\boldsymbol{x}} = (\widetilde{X}_1, \ldots, \widetilde{X}_q)$ are the cluster representatives derived in phase 2 and $q < p$. This reparametrization directly enforces the strong heredity principle (Eq. (39)), i.e., if either main effect estimates are 0, then $\hat{\alpha}_{jE}$ will be zero and a non-zero interaction coefficient implies non-zero $\hat{\beta}_j$ and $\hat{\beta}_E$. To perform variable selection in this new parametrization, we follow Choi *et al.* [38] and penalize $\boldsymbol{\gamma} = (\gamma_{1E}, \ldots, \gamma_{pE})$ instead of penalizing $\boldsymbol{\alpha}$ as in (**??**), leading to the following penalized least squares criterion:

$$\arg\min_{\beta_0, \boldsymbol{\beta}, \boldsymbol{\gamma}} \frac{1}{2} \|Y - g(\boldsymbol{\mu})\|^2 + \lambda_\beta \left(w_1\beta_1 + \cdots + w_q\beta_q + w_E\beta_E \right) + \lambda_\gamma \left(w_{1E}\gamma_{1E} + \cdots + w_{qE}\gamma_{qE} \right) \quad (41)$$

where $g(\boldsymbol{\mu})$ is from (40), λ_β and λ_γ are tuning parameters and $\mathbf{w} = (w_1, \ldots, w_q, w_{1E}, \ldots, w_{qE})$ are prespecified adaptive weights. The λ_β tuning parameter controls the amount of shrinkage applied to the main effects, while λ_γ controls the interaction estimates and allows for the possibility of excluding the interaction term from the model even if the corresponding main effects are non-zero. The adaptive weights serve as a way of allowing parameters to be penalized differently. Furthermore, adaptive weighting [15] has been shown to construct oracle procedures [13], i.e., asymptotically, it performs as well as if the true model were given in advance. The oracle property is achieved when the weights are a function of any root-n consistent estimator of the true parameters e.g. maximum likelihood (MLE) or ridge regression estimates. It can be shown that the procedure in (41) asymptotically possesses the oracle property [38], even when the number of parameters tends to ∞ as the sample size increases, if the weights are chosen such that

$$w_j = \left| \frac{1}{\hat{\beta}_j} \right|, \quad w_{jE} = \left| \frac{\hat{\beta}_j \hat{\beta}_E}{\hat{\alpha}_{jE}} \right| \qquad \text{for } j = 1, \ldots, q \quad (42)$$

where $\hat{\beta}_j$ and $\hat{\alpha}_j$ are the MLEs, *using the transformed variables*, from (37) or the ridge regression estimates when $q > n$. The rationale behind the data-dependent $\hat{\boldsymbol{w}}$ is that as the sample size grows, the weights for the truly zero predictors go to ∞ (which translates to a large penalty), whereas the weights for the truly non-zero predictors converge to a finite constant [15].

There have been several more recent proposals for modeling interactions with the strong heredity constraint in the variable selection via penalization literature including Composite Absolute Penalties (CAP) [37], Variable selection using Adaptive Nonlinear Interaction Structures in High dimensions (VANISH) [44], Strong Hierarchical Lasso (hierNet) [30], Group-Lasso Interaction Network (glinternet) [41], Group Regularized Estimation under Structural Hierarchy (GRESH) [39] and a Framework for Modeling Interactions with a Convex Penalty (FAMILY) [46]. While each method has their own merit, including that they are all convex optimization problems, they all contain complex penalty functions which are hard to interpret and lead to computationally expensive fitting algorithms. On the other hand, the objective function in (41) can be solved using an iterative approach (by first fixing $\boldsymbol{\beta}$ and then $\boldsymbol{\alpha}$) which simplifies to a LASSO type problem; one that has been extensively studied, is well understood and can be solved efficiently using existing software (e.g. `glmnet` [8]). A limitation of this approach is that the optimization problem is non-convex, arising from the reparametrization of $\boldsymbol{\alpha}$ as a product of optimization variables $(\boldsymbol{\beta}, \boldsymbol{\gamma})$, and hence convergence to the global minimum is not guaranteed [38]. We argue that since there is only one E, and that $\widetilde{X}$ is much smaller in dimension, finding a solution is much more likely.

To our knowledge, strong hierarchies have never previously been used in HD interaction analysis in genomics or brain imaging studies. Furthermore, the specific choices of weights proposed here, i.e., based on the transformed variables from phase 2, have not been previously used. Choi *et al.* [38] estimated their weights simultaneously, but this would not be feasible in HD data. Finally, the adaptation to interactions with one key E variable is specific to our situation and this leads to computational efficiencies. These three points constitute novel aspects of this thesis. I have a working implementation of this, and am in the process of conducting simulation studies.

6 (5) A brief intro to linear mixed models followed by why naive penalization violates the normality of residuals to motivate your ggmix chapter.

7 you will need a section on things like gradient descent and other algorithms for finding solutions efficiently

References

[1] Jianqing Fan, Fang Han, and Han Liu. Challenges of big data analysis. *National science review*, 1(2):293–314, 2014. 2

[2] Teri A Manolio, Francis S Collins, Nancy J Cox, David B Goldstein, Lucia A Hindorff, David J Hunter, Mark I McCarthy, Erin M Ramos, Lon R Cardon, Aravinda Chakravarti, et al. Finding the missing heritability of complex diseases. *Nature*, 461 (7265):747–753, 2009. 2, 3

[3] Jian Yang, Beben Benyamin, Brian P McEvoy, Scott Gordon, Anjali K Henders, Dale R Nyholt, Pamela A Madden, Andrew C Heath, Nicholas G Martin, Grant W Montgomery, et al. Common snps explain a large proportion of the heritability for human height. *Nature genetics*, 42(7):565, 2010. 2

[4] Jeffrey T Leek and John D Storey. A general framework for multiple testing dependence. *Proceedings of the National Academy of Sciences*, 105(48):18718–18723, 2008. 3

[5] Xinyi Lin, Seunggeun Lee, David C Christiani, and Xihong Lin. Test for interactions between a genetic marker set and environment in generalized linear models. *Biostatistics*, page kxt006, 2013. 3

[6] Eric E Schadt. Molecular networks as sensors and drivers of common human diseases. *Nature*, 461(7261):218–223, 2009. 3

[7] Daniel Müllner. fastcluster: Fast hierarchical, agglomerative clustering routines for R and Python. *Journal of Statistical Software*, 53(9):1–18, 2013. URL http://www.jstatsoft.org/v53/i09/. 5

[8] Jerome Friedman, Trevor Hastie, and Rob Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of statistical software*, 33(1):1, 2010. 5, 6, 13, 22

[9] Yi Yang and Hui Zou. gglasso: Group lasso penalized learning using a unified bmd algorithm. 2014. URL http://CRAN.R-project.org/package=gglasso. R package version 1.3. 5

[10] Max Kuhn. Caret package. *Journal of Statistical Software*, 28(5), 2008.

[11] Arthur E Hoerl and Robert W Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970. 4

[12] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996. 4

[13] 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. 4, 21

[14] Robert Tibshirani, Michael Saunders, Saharon Rosset, Ji Zhu, and Keith Knight. Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(1):91–108, 2005. 4

[15] Hui Zou. The adaptive lasso and its oracle properties. *Journal of the American statistical association*, 101(476):1418–1429, 2006. 4, 21

[16] Nicolai Meinshausen. Relaxed lasso. *Computational Statistics & Data Analysis*, 52(1): 374–393, 2007. 4

[17] Cun-Hui Zhang. Nearly unbiased variable selection under minimax concave penalty. *The Annals of Statistics*, pages 894–942, 2010. 4

[18] Peter Bühlmann, Philipp Rütimann, Sara van de Geer, and Cun-Hui Zhang. Correlated variables in regression: clustering and sparse estimation. *Journal of Statistical Planning and Inference*, 143(11):1835–1858, 2013. 4, 5

[19] Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2):301–320, 2005. 4

[20] Nathalie Tzourio-Mazoyer, Brigitte Landeau, Dimitri Papathanassiou, Fabrice Crivello, Olivier Etard, Nicolas Delcroix, Bernard Mazoyer, and Marc Joliot. Automated anatomical labeling of activations in spm using a macroscopic anatomical parcellation of the mni mri single-subject brain. *Neuroimage*, 15(1):273–289, 2002. 4

[21] Minoru Kanehisa, Michihiro Araki, Susumu Goto, Masahiro Hattori, Mika Hirakawa, Masumi Itoh, Toshiaki Katayama, Shuichi Kawashima, Shujiro Okuda, Toshiaki Tokimatsu, et al. Kegg for linking genomes to life and the environment. *Nucleic acids research*, 36(suppl 1):D480–D484, 2008. 4

[22] Francis Bach, Rodolphe Jenatton, Julien Mairal, Guillaume Obozinski, et al. Structured sparsity through convex optimization. *Statistical Science*, 27(4):450–468, 2012. 4

[23] Ming Yuan and Yi Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67, 2006. 4

[24] Maurice Kendall. *A Course in Multivariate analysis*. London: Griffin, 1957. 5

[25] Trevor Hastie, Robert Tibshirani, David Botstein, and Patrick Brown. Supervised harvesting of expression trees. *Genome Biology*, 2(1):1–0003, 2001. 5

[26] Mee Young Park, Trevor Hastie, and Robert Tibshirani. Averaged gene expressions for regression. *Biostatistics*, 8(2):212–227, 2007. 5

[27] Daniela M Witten, Ali Shojaie, and Fan Zhang. The cluster elastic net for high-dimensional regression with unknown variable grouping. *Technometrics*, 56(1):112–122, 2014. 6

[28] Marian J Bakermans-Kranenburg and Marinus H Van IJzendoorn. The hidden efficacy of interventions: Gene× environment experiments from a differential susceptibility perspective. *Annual review of psychology*, 66:381–409, 2015.

[29] David R Cox. Interaction. *International Statistical Review/Revue Internationale de Statistique*, pages 1–24, 1984.

[30] Jacob Bien, Jonathan Taylor, Robert Tibshirani, et al. A lasso for hierarchical interactions. *The Annals of Statistics*, 41(3):1111–1141, 2013. 19, 20, 22

[31] Paul Tseng and Sangwoon Yun. A coordinate gradient descent method for nonsmooth separable minimization. *Mathematical Programming*, 117(1):387–423, 2009. 6, 9

[32] Lukas Meier, Sara Van De Geer, and Peter Bühlmann. The group lasso for logistic regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(1):53–71, 2008. 6

[33] Jürg Schelldorfer, Peter Bühlmann, GEER DE, and SARA VAN. Estimation for high-dimensional linear mixed-effects models using l1-penalization. *Scandinavian Journal of Statistics*, 38(2):197–214, 2011. 6, 8

[34] K. Lange, D. Hunter, and I. Yang. Optimization transfer using sur- rogate objective functions (with discussion). *Journal of Computational and Graphical Statistics*, 9:1–20, 2000. 17

[35] D.R. Hunter and K. Lange. A tutorial on MM algorithms. *The American Statistician*, 58(1):30–37, 2004. ISSN 0003-1305. 17

[36] T. Wu and K. Lange. The MM alternative to EM. *Statistical Science*, 4:492–505, 2010. 17

[37] Peng Zhao, Guilherme Rocha, and Bin Yu. The composite absolute penalties family for

grouped and hierarchical variable selection. *The Annals of Statistics*, pages 3468–3497, 2009. 19, 22

[38] Nam Hee Choi, William Li, and Ji Zhu. Variable selection with the strong heredity constraint and its oracle property. *Journal of the American Statistical Association*, 105 (489):354–364, 2010. 19, 20, 21, 22

[39] Yiyuan She and He Jiang. Group regularized estimation under structural hierarchy. *arXiv preprint arXiv:1411.4691*, 2014. 19, 20, 22

[40] Asad Haris, Daniela Witten, and Noah Simon. Convex modeling of interactions with strong heredity. *Journal of Computational and Graphical Statistics*, 25(4):981–1004, 2016. 19

[41] Michael Lim and Trevor Hastie. Learning interactions via hierarchical group-lasso regularization. *Journal of Computational and Graphical Statistics*, 24(3):627–654, 2015. 19, 22

[42] Ning Hao, Yang Feng, and Hao Helen Zhang. Model selection for high-dimensional quadratic regression via regularization. *Journal of the American Statistical Association*, pages 1–11, 2018. 19

[43] Rajen D Shah. Modelling interactions in high-dimensional data with backtracking. *Journal of Machine Learning Research*, 17(207):1–31, 2016. 19

[44] Peter Radchenko and Gareth M James. Variable selection using adaptive nonlinear interaction structures in high dimensions. *Journal of the American Statistical Association*, 105(492):1541–1553, 2010. 19, 22

[45] Hugh Chipman. Bayesian variable selection with related predictors. *Canadian Journal of Statistics*, 24(1):17–36, 1996. 20

[46] Asad Haris, Daniela Witten, and Noah Simon. Convex modeling of interactions with strong heredity. *arXiv preprint arXiv:1410.3517*, 2014. 22