Guiding adaptive shrinkage by co-data to improve regression-based prediction and feature selection

Mark A. van de Wiel[1], Wessel N. van Wieringen[1,2]

[1]*Dept of Epidemiology and Data Science, Amsterdam Public Health research institute, Amsterdam University Medical Centers, Amsterdam, The Netherlands;* [2]*Dept of Mathematics, VU University, Amsterdam, The Netherlands*

Abstract

The high dimensional nature of genomics data complicates feature selection, in particular in low sample size studies - not uncommon in clinical prediction settings. It is widely recognized that complementary data on the features, 'co-data', may improve results. Examples are prior feature groups or p-values from a related study. Such co-data are ubiquitous in genomics settings due to the availability of public repositories. Yet, the uptake of learning methods that structurally use such co-data is limited. We review guided adaptive shrinkage methods: a class of regression-based learners that use co-data to adapt the shrinkage parameters, crucial for the performance of those learners. We discuss technical aspects, but also the applicability in terms of types of co-data that can be handled. This class of methods is contrasted with several others. In particular, group-adaptive shrinkage is compared with the better-known sparse group-lasso by evaluating feature selection. Finally, we demonstrate the versatility of the guided shrinkage methodology by showing how to 'do-it-yourself': we integrate implementations of a co-data learner and the spike-and-slab prior for the purpose of improving feature selection in genetics studies.

1 Introduction

Genetics and genomics data are usually of a high-dimensional nature: the number of measured features vastly exceeds the number of samples. Two plagues of such high-dimensional data are low signal-to-noise ratio and multicollinearity. All prediction and feature selection models are affected by those plagues. Here, we focus on regression-based models, which employ regularization by introducing shrinkage either through a penalty or an informative prior. Let us first elaborate on these two plagues and their implications before discussing potential cures.

First, a low signal-to-noise ratio implies an abundance of irrelevant features. As, by default, shrinkage parameters are shared by all features, such an abundance may lead to overshrinkage for relevant features. On its turn, this may harm the predictive accuracy of the resulting model. The second one is multicollinearity, which doubles in omics as many genomic features are highly correlated due to shared biological properties. In a predictive model features are competing with one another. Selecting one feature will likely de-select another one if the two are strongly correlated. Then, small fluctuations in the data set drive the feature selection, rendering it instable.

Three cures for the two plagues come immediately to mind. First, the low signal-to-noise ration is countered by enforcing sparsity, e.g. by a lasso penalty or a horseshoe prior, to better accommodate a strong contrast between relevant and non-relevant features. This certainly helps when there indeed exists such a strong contrast, but may be less appropriate in many genomics settings in which many small effects may pile up (Boyle et al., 2017). That is, in the latter case one may wish to accommodate the grey scale between relevant and non-relevant. Second, the instable feature selection due to multicollinearity may be countered by stability selection (Meinshausen and Bühlmann, 2010). In a nutshell, one generates many random copies of the data set, e.g. by bootstrapping, and then composes the ultimate set of selected features from those which are selected in a large proportion of those copies. The third solution is to bring in external knowledge. The promises of this solution are clear: external information allows for better modelling of the shrinkage and can break the strong competition between features. This solution may be combined with the former ones.

We argue that the last solution is still under-used in practice, which is why we focus on it in this review. One reason for the under-use is pragmatism: it takes time to compile external data and requires thinking on what to include. Moreover, leveraging external knowledge can be achieved in many different ways, rendering it difficult to have an overview and pick an algorithm for one's needs. Our aim is two-fold: on one hand convince potential users that leveraging external knowledge may be well worth the effort and on the other hand provide guidance on which algorithms to use for which settings.

Our focus lies on methods that allow guided adaptive shrinkage. That is, the shrinkage is modeled as a function of the external information on the features. We refer to the latter as 'co-data' (Neuenschwander et al., 2010; Van de Wiel et al., 2016), 'complementary data'. Alternatively, the term 'features of features' has been coined (Tay et al., 2023). The group-adaptive lasso (Zeng et al., 2021; van Nee et al., 2023b) is a special case of guided adaptive shrinkage. As the (sparse) group-lasso is a better-known method that applies to the same setting, i.e. one co-data source defining feature groups, we start by contrasting the penalty functions of these two methods. Then, we shift the focus to a general formulation of guided adaptive shrinkage, and review several methods. In particular, we consider what types of co-data can be handled, what types of response are accommodated, and what strategy is used to estimate hyperparameters.

We contrast the guided adaptive shrinkage approach to related ones. These either share the adaptive nature of the former - such as the adaptive lasso - or the use of co-data, such as structured regularization methods (including the sparse group-lasso), and regression-based transfer learning. As the contrast between group-adaptive lasso and group-lasso is particularly relevant, we use simulations to compare the two for a varying number of feature groups in terms of feature selection performance. Throughout, we focus on evaluating feature selection, as the potential benefit of using co-data for the purpose of prediction has already been demonstrated by many of the discussed works.

Finally, we show the versatility of the approach by discussing a 'do-it-yourself' solution for one's favorite model. We use it to illustrate the benefit of co-data for feature selection in a spike-and-slab model, which is a popular Bayesian model for selecting features in large scale genomics studies (Carbonetto and Stephens, 2012).

2 Group-adaptive lasso versus sparse group lasso

Before defining the guided adaptive shrinkage methodology in a general framework, we briefly discuss a canonical setting: co-data that consists of one grouping of the features. Examples are different data modalities - all used within one regression - with possibly very different dimensions (e.g. gene expression, mutations, clinical variables, imaging-derived features) or a grouping based on genomic location, such as the chromosomes. Both the group-adaptive lasso (Zeng et al., 2021; van Nee et al., 2023b) and sparse group lasso (Simon et al., 2013) can accommodate such groups, denoted by $G_g, g = 1, \ldots, G$. The essential difference between the two is adaptivity. Sparse group-lasso extends the lasso by augmenting the L_1 penalty function on the regression coefficients of the p features, $(\beta_j)_{j=1}^p$, with a group-penalty on the G groups of features, whereas group-adaptive lasso employs different penalties across groups of features. The penalty functions P for group-lasso and its adaptive counterpart are:

$$P_{\lambda,\lambda'}(\boldsymbol{\beta}) = \lambda \sum_{j=1}^{p} |\beta_j| + \lambda' \sum_{g=1}^{G} ||\boldsymbol{\beta}_g||_2 \qquad \text{(sparse group-lasso)} \qquad (1)$$

$$P_{\boldsymbol{\lambda}}(\boldsymbol{\beta}) = \sum_{g=1}^{G} \lambda_g \sum_{j \in G_g} |\beta_j| \qquad \text{(group-adaptive lasso),} \qquad (2)$$

with norm $||\boldsymbol{\beta}_g||_2 = (\sum_{k \in G_g} \beta_k^2)^{1/2}$. Penalized regression can also be cast in an equivalent constraint optimization setting, where the constraints are one-to-one linked to the penalties. Figure 1 illustrates these constraints for a toy example with two groups of two features. Clearly, the constraint adapts to the strength of a group of features in the group-adaptive setting, whereas the constraint is essentially the same for both groups in the sparse group-lasso setting. Those constraints are very important in high-dimensional settings, because features compete to be selected. Hence, depending on the situation, the two methods may perform very differently, as we will illustrate further on.

3 Guided adaptive shrinkage

Here, we introduce and illustrate the general framework before reviewing a variety of guided adaptive shrinkage methods.

3.1 General framework

Let $\mathbf{y} = (y_i)_{i=1}^n$ be the response of interest, $X = (x_{ij})_{i=1,j=1}^{n,p}$ the data matrix with x_{ij} : the value of feature j for sample i, $\boldsymbol{\beta} = (\beta_j)_{j=1}^p$ the regression coefficients, and $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_k)$ denote the nuisance parameters (such as noise variance σ^2). We assume that the regression model linking X to $\mathbf{y}$ by $\boldsymbol{\beta}$ defines a likelihood function $\mathcal{L}$. Other fit functions could be used as well. Then, the guided adaptive shrinkage framework is summarized by:

$$\begin{aligned} &\mathcal{L}(\boldsymbol{\beta}, \boldsymbol{\theta}; X, \mathbf{y}) && \text{(Likelihood)} \\ &P_{\boldsymbol{\lambda}}(\boldsymbol{\beta}) && \text{(Shrinkage)} \\ &\lambda_j = f_{\boldsymbol{\alpha}}(Z_{.j}) && \text{(Guided adaptation),} \end{aligned}$$

with penalty vector $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_p)$, and co-data matrix $Z = (z_{cj})_{c,j=1}^{C,p}$. Z comprises of rows $Z_{c.}$ that corresponds to co-data source c and of columns $Z_{.j}$ corresponding to feature j. $P_{\boldsymbol{\lambda}}(\boldsymbol{\beta})$ is either a penalty function in a classical setting, or a prior in a Bayesian setting. In fact, when

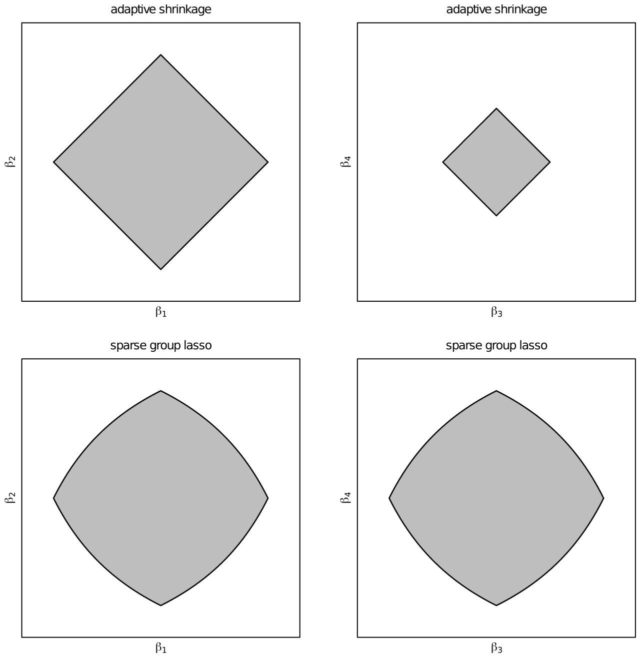

Figure 1: Parameter constraints for two groups of two features. Top-row: group-adaptive lasso; bottom-row: the sparse group lasso

the penalty function is formulated as a log-prior, maximization of the penalized likelihood renders the Bayesian posterior mode estimate. This equivalence facilitates switching between the two paradigms. Besides the choice of paradigm and type of shrinkage, methods differ in terms of how they estimate $\boldsymbol{\lambda}$ and how they incorporate co-data matrix or vector Z. Here, co-data function f, parameterized by a lower dimensional parameter $\boldsymbol{\alpha}$, connects Z to the penalties $\boldsymbol{\lambda}$. We emphasize the relative low dimension of $\boldsymbol{\alpha}$ (w.r.t. p): this facilitates stable estimation of the high-dimensional penalty vector $\boldsymbol{\lambda}$. Figure 2 illustrates guided adaptive shrinkage.

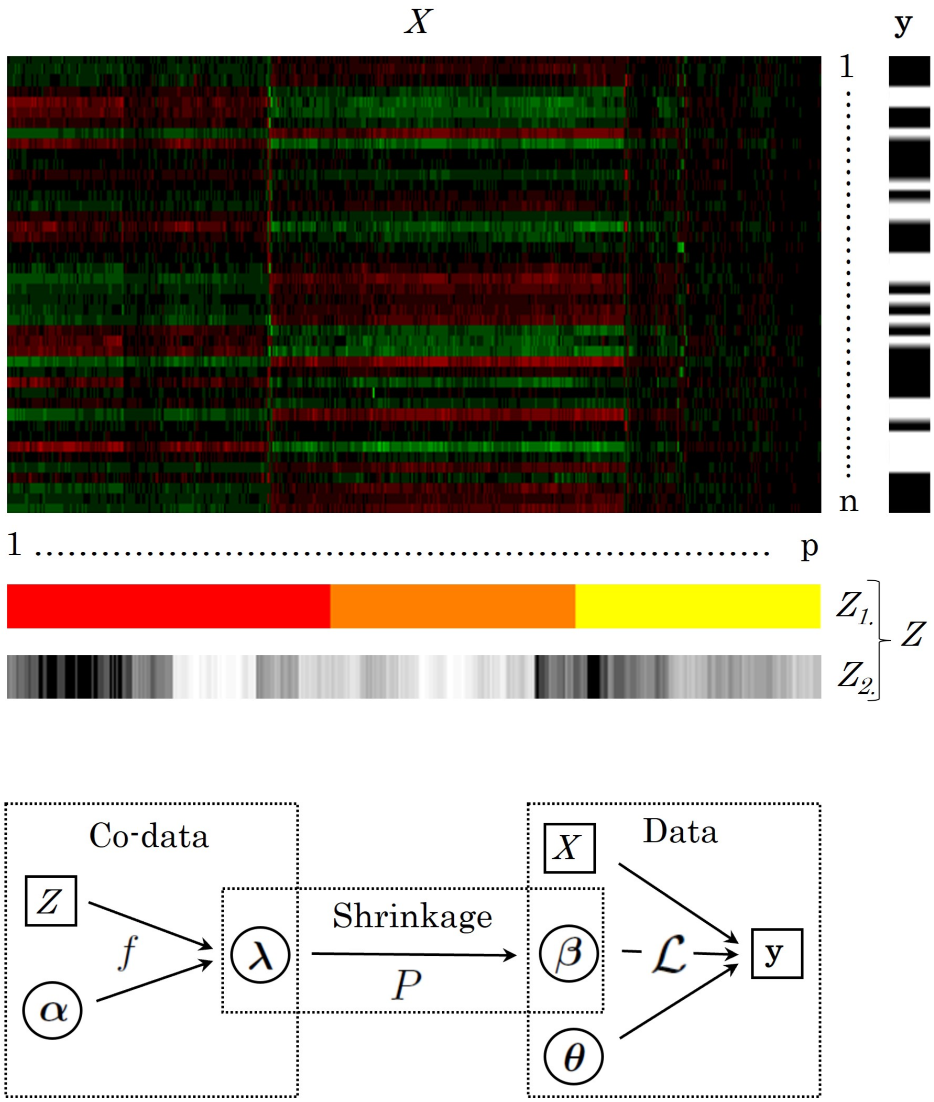

Figure 2: Guided adaptive shrinkage using co-data $Z = \left(\begin{smallmatrix} Z_{1.} \\ Z_{2.} \end{smallmatrix}\right)$. $\boldsymbol{\alpha}$: hyper-parameters; f: adaptation function; $\boldsymbol{\lambda}$: penalty vector; P: shrinkage via penalty or prior; $\boldsymbol{\beta}$: regression parameters; X: data; $\boldsymbol{\theta}$: nuisance parameters; $\mathcal{L}$: likelihood; $\mathbf{y}$: response

3.2 Methods

Guided adaptive shrinkage methods can be classified in multiple ways. The likelihood $\mathcal{L}$, largely determined by the type of response $\mathbf{y}$ (e.g. continuous/binary/survival), and the type of penalty/prior $P_{\boldsymbol{\lambda}}$ are obvious classifiers. Moreover, the type of co-data Z allowed by co-data function $f_{\boldsymbol{\alpha}}$ differentiates the applicability of methods in two ways: can multiple co-data sources be accommodated, and what types are allowed: grouped, continuous or mixed? Finally, the methods crucially differ in how hyperparameters are estimated. Here, we distinguish 1) Cross-validation: hyperparameters are tuned by cross-validation; 2) Empirical Bayes: hyperparameters are tuned by empirical Bayes techniques; 3) Full Bayes: hyperparameters are endowed with an hierarchical prior; 4) Joint estimation: Hyperparameters and regression parameters are estimated jointly. Table 1 classifies co-data guided shrinkage methods according to these criteria. Below we provide more specific details on each of these methods.

Method	Reference*	Software	Likelihood$^+$	Shrinkage	Co-data	Hyp.$^\$$
Multi-pen. PLS	Tai (2007)	-	Bin	Lasso	Group Uni	CV
Weighted lasso	Bergensen (2011)	-	Gauss, Bin, Cox	Lasso	Cont Uni	CV
Group-regul. ridge	vd Wiel (2016)	`GRridge`	Gauss Bin, Cox	Ridge	Group Multi	EB
Integrated pen. factors lasso	Boulesteix (2017)	`ipflasso`	Gauss Bin, Cox	Lasso	Group Uni	CV
Group-adapt. pen. regr.	Velten (2019)	`graper`	Gauss Bin	Lasso, Ridge Spike & Slab	Group Uni	FB
Group-regul. ridge	Ignatiadis (2020)	`SigmaRidge` (Julia)	Gauss	Ridge	Group Uni	EB+ CV
Co-data adapt. ridge	Van Nee (2021)	`ecpc`	Gauss Bin, Cox	Ridge	Mixed Multi	EB
Incorp. prior info pen. regr.	Zeng (2021)	`xtune`	Gauss Bin, Mult	Elastic Net	Mixed Multi	EB
Hierarchical ridge	Kawaguchi (2022)	`xrnet`	Gauss GLM	Ridge	Mixed Multi	CV
Group-adapt. elastic net	Van Nee (2023b)	`squeezy`	Gauss Bin, Cox	Elastic Net	Group Uni	EB
Feature-weight. elastic net	Tay (2023)	`fwelnet`	Gauss Bin	Elastic net	Mixed Multi	Joined
Co-data adapt. Horseshoe regr.	Busatto (2023)	`infHS`	Gauss Probit	Horseshoe	Mixed Multi	FB

* : First author only

$^+$: Gauss: Gaussian, continuous; Bin: Binomial, binary; Cox: Proportional hazards, survival

$^\$$: Hyperparameter estimation. CV: Cross-validation; EB: Empricial Bayes; FB: Full Bayes

Table 1: Co-data adaptive shrinkage methods

`Multi-penalty PLS` (Tai and Pan, 2007) is a pioneering method on the adaptation of penalties based on grouped co-data. It models co-data function $f_{\boldsymbol{\alpha}}$ simply by $\boldsymbol{\alpha} = (\lambda_1, \ldots, \lambda_G)$. The authors cast their method in a partial least squares setting, but they show the correspondence to ordinary regression. The method uses soft-thresholding with adaptive thresholds, which is strongly related to group-adaptive lasso regression. It accommodates only one grouped co-data source. It introduces an heuristic to reduce computing time for tuning $\boldsymbol{\lambda}$ by cross-validation, using a weighting function.

`Weighted lasso` (Bergersen et al., 2011) models the co-data function $f_{\boldsymbol{\alpha}} = f_{\lambda,q}$, which depends on à priori defined similarity weights between X and Z or between $\mathbf{y}$ and Z. The weighting function is somewhat arbitrary and limited in flexibility. It relies on two tuning parameters: the lasso penalty λ and q, which tunes the importance of the weights. Two examples of similarity weights are: correlation between an mRNA feature (X) and its DNA counterpart (Z), or regression coefficients that relate $\mathbf{y}$ to Z. This method is simple, easy to extend and implement. Moreover, it is relatively fast, because only two hyper-parameters are tuned. It accommodates only one co-data source, though, that needs to be continuous.

`GRridge` and `ecpc` (Van de Wiel et al., 2016; van Nee et al., 2021) are both based on ridge regression, but the latter extends on the former by allowing non-grouped co-data using a regression parametrization, $f_{\boldsymbol{\alpha}} = Z_{\cdot j}\boldsymbol{\alpha}$). In addition, `ecpc` implements hyperparameter shrinkage, which is useful when a co-data source consists of many feature groups. The methods share the methodology for hyperparameter estimation by moment-based empirical Bayes. The estimation procedure is modular, which provides computational efficiency and flexibility in terms of implementation, but does not propagate uncertainty as full Bayesian procedures do. The setting is not sparse, although posterior variable selection is implemented, and shown to be competitive to lasso-based methods.

`ipflasso` (Boulesteix et al., 2017) is a group-adaptive method, so $f_{\boldsymbol{\alpha}}$ is simply parameterized by $\boldsymbol{\alpha} = (\lambda_1, \ldots, \lambda_G)$. The method is simple and easy to extend. As it tunes $\boldsymbol{\lambda}$ by full-blown multi-grid cross-validation it may be slow when the number of feature groups increases. It accommodates only one grouped co-data source. The methods is extended by Zhao and Zucknick (2020) to allow for hierarchical groups.

`graper` (Velten and Huber, 2019) is the first fully Bayesian co-data method that is flexible in terms of penalization and incorporates, next to lasso and ridge priors, the spike-and-slab. It is less flexible in terms of co-data as it only allows groups. Computational scalability is achieved by developing a variational Bayes approximation. As a fully Bayesian method it provides uncertainty quantification, although this was not evaluated by the authors and may be compromised by the use of variational Bayes.

`SigmaRidge` (Ignatiadis and Lolas, 2020) is a hybrid method that deals with group-adaptive shrinkage in the ridge setting. It combines cross-validation and empirical Bayes for hyperparameter tuning. In the Bayesian formulation, the error noise in the linear model, σ, is also a hyperparameter. Here, it is treated as a global parameter tuned by CV, whereas $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_G)$ is determined by moment-based empirical Bayes. As the latter is analytical, it is embedded in the cross-validation of σ. This leads to truly joint estimation of all hyperparameters. Asymptotic optimality results are provided and computational efficiency is achieved by approximate, analytical leave-one-out-CV. The scope of the method is limited, though, as it only handles linear regression with grouped co-data.

`xtune` (Zeng et al., 2021) supports the use of generic, mixed-type co-data by modeling $f_{\boldsymbol{\alpha}} = Z_{\cdot j}\boldsymbol{\alpha}$. It provides efficient estimation of hyperparameters $\boldsymbol{\alpha}$ by a Gaussian approximation of the elastic net prior. Moreover, it uses a majorization technique to speed up optimizing of the marginal likelihood to tune $\boldsymbol{\alpha}$ by empirical Bayes. Hence, it is computationally efficient and versatile in its use. Results are presented for linear response only, but the software supports binary and multi-class response as well.

`xrnet` (Kawaguchi et al., 2022) differs from the other ones in the way it shrinks $\boldsymbol{\beta}$: to a co-data moderated *mean* $Z_{\cdot j}\boldsymbol{\alpha}$ instead of a variance. It shrinks $\boldsymbol{\alpha}$ to zero within the same objective function using a hierarchical ridge penalty. It is computationally efficient, as once the two hyperparameters are fixed, the objective function is convex. It does not provide a solution for feature selection.

`squeezy` (van Nee et al., 2023b) is similar in spirit to `xtune`, as it also approximates the marginal likelihood by a Gaussian one. It compliments this approximation, however, with a proof based on the multivariate central limit theorem. It is computationally very efficient as it makes use of many shortcuts available in the Gaussian setting. The main implementation supports grouped co-data only, but it can directly use the output of ridge-based `ecpc` (discussed above) to handle mixed co-data and hyperparameter shrinkage.

`fwelnet` (Tay et al., 2023) models, conditionally on a global penalty parameter λ, feature-specific penalties by $f_{\boldsymbol{\alpha}} = \lambda w_j(\boldsymbol{\alpha})$, with weights $w_j(\boldsymbol{\alpha}) = \left(\exp(Z_{\cdot j}\boldsymbol{\alpha})/\sum_{j=1}^{p}\exp(Z_{\cdot j}\boldsymbol{\alpha})\right)^{-1}$. The approach uniquely optimizes hyperparameters $\boldsymbol{\alpha}$ and regression parameters $\boldsymbol{\beta}$ jointly. The optimization algorithm alternates between updating $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$, using gradient search for $\boldsymbol{\alpha}$ and an elastic net solver for $\boldsymbol{\beta}$. A potential identification problem is circumvented by normalizing the weights. Hence, it is a non-hierarchical formulation, which adapts weights instead of shrinkage.

`infHS` (Busatto and van de Wiel, 2023) is a fully Bayesian method that uses the popular horseshoe prior to encode sparsity into the predictive model. It uses a regression parametrization ($Z_{\cdot j}\boldsymbol{\alpha}$) to allow mixed co-data types. It modifies the prior mean of the local regularization parameters, thereby particularly facilitating high-dimensional settings with many small signals and a few outlying large ones. It is suitable for feature selection, and the variational Bayes approximation of the posteriors provides computational scalability of the method. Binary outcome is accommodated by a probit formulation.

4 Related methods

Below we discuss some related regression-based methods that either share the use of external knowledge or the concept of adaptation of shrinkage with the discussed guided adaptive shrinkage methodology.

4.1 Group penalties, structured regularization

Guided adaptive shrinkage relates to structured regularization as both frameworks allow to incorporate external information in the regularization of regression models. Well-known examples of the latter are the group-lasso and the hierarchical lasso, but many more methods are available; see Vinga (2021) for a extensive review, and Zhu et al. (2019) for a Bayesian perspective. As both the sparse group-lasso (and variations thereof) and the group-adaptive lasso can be applied to co-data groups, we restrict ourselves to a comparison between those two types of methods with penalty functions (1) and (2). Here, co-data matrix Z consists of only one row vector $Z_{1\cdot}$ containing categorical entries that correspond to the feature groups.

The sparse group-lasso focuses on selecting groups and features, whereas the group-adaptive lasso selects only features while adapting to different group strengths. Hence, the former may be more suitable in settings with many groups, of which a large part is not relevant, while the latter is more flexible in settings with few groups. As an illustration, we briefly study this claim in a simulation setting.

In a linear regression setting, $\mathbf{y} = X\boldsymbol{\beta} + \boldsymbol{\epsilon}$, we simulate $n = 200$ samples for $p = 2,000$ features divided over $G = 3, 6, 9, 15, 24, 39, 60, 99$ equally-sized groups. Of these groups, 1/3rd contains non-zero coefficients. To allow some variation between the non-zero groups, the proportion of non-zero coefficients in those groups is sampled from a Beta(2,6) distribution, averaging to 1/4 non-zero's per group. Non-zero β's are generated from a scaled t_3 distribution, such that the total explained variation of the features equals that of the Gaussian noise, $\epsilon_i \sim N(0, \sigma^2 = 1)$. Features x_{ij} are independently sampled from a standard Gaussian.

On the simulated training data, the group-adaptive lasso and the sparse group-lasso were fitted using the R packages `squeezy` and `SGL`, respectively, using the known feature groups as input and using defaults for other parameters. We focus on feature selection by evaluating the F1-score, the harmonic mean of precision and recall. As such scores are somewhat incomparable for models of different size, we opt to fix the number of selected non-zero features, p_{sel}. Both methods allow this as both produce a regularization path that may be used for this purpose. For each group-size G, simulations were repeated 25 times. Figure 3 shows the results for $p_{\text{sel}} = 25, 50$.

The simulations clearly support the claim: for a small to intermediate number of groups, group-adaptive lasso outperforms sparse group-lasso, whereas the latter becomes superior when many groups are used. Within one setting, group-adaptive lasso is generally somewhat more variable in performance across repeats, possibly due to the higher number of hyperparameters that need to be estimated.

The Supplementary Material shows an extra simulation scenario that is more 'group-sparse': $p = 10,000, G = 60, 99$ and 5 groups with non-zero coefficients. These results support our conclusion above: for a large number of groups ($G = 99$) sparse group-lasso is somewhat superior to group-adaptive lasso, but the latter is competitive for an intermediate number of groups ($G = 60$), even in this fairly group-sparse scenario. Finally, the Supplementary Material also provides a solution to shrink hyperparameters in the group-adaptive lasso setting. This is shown to be particularly useful when the number of groups is large and when these groups are not informative (the 'null-setting').

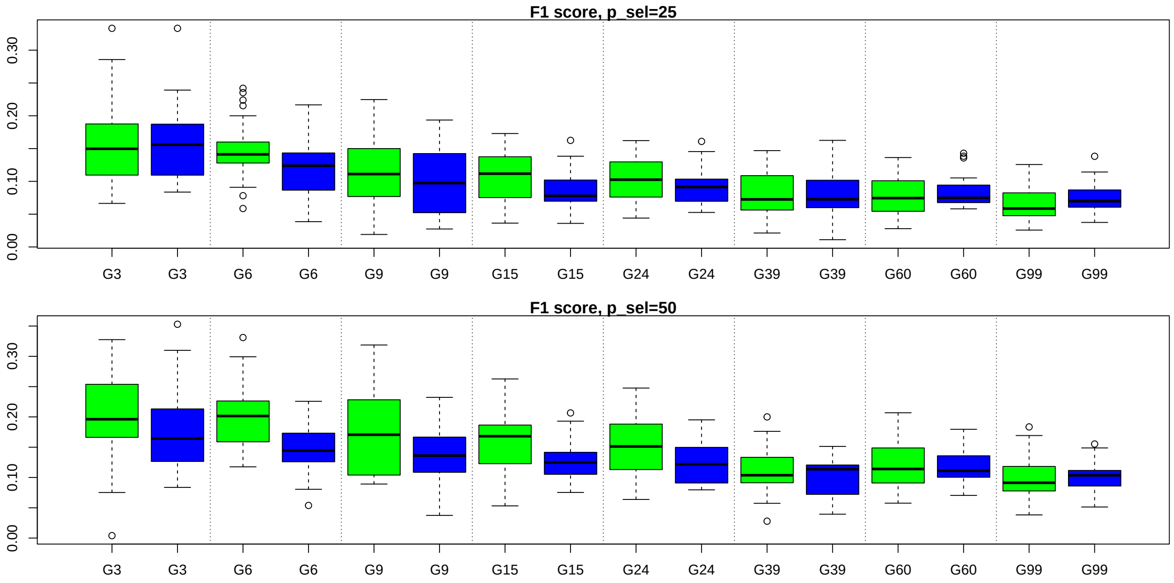

Figure 3: F1 score for variable selection ($p_{\text{sel}} = 25, 50$) for group-adaptive lasso (green) and sparse group-lasso (blue) for $G = 3, \ldots, 99$ feature groups (25 simulations per G)

4.2 Adaptive lasso and variations

A second class of methods related to guided adaptive shrinkage, consists of the adaptive lasso (Zou, 2006), and variations thereof. There is a fundamental difference between the two though. The former uses external data to guide the adaptation, whereas the latter bases the adaptation purely on the data itself: it uses the parameter estimates of an unpenalized (or L_2-penalized) model as inverse penalty weights. The adaptive lasso has proven its use in low-to-medium dimensional settings, in which it can counter the overpenalization of non-zero parameters by the lasso. In high-dimensional settings, however, double use of the same data may come at the cost of overfitting. The extend of the latter likely depends on the (unknown) underlying sparsity. Belhechmi et al. (2020) present a hybrid adaptive/group-adaptive algorithm which may (partly) overcome the overfitting issue. Basically, it groups the parameters estimates of an initial fit according to the prior groups, and uses these to define weights in the adaptive step.

4.3 Regression-based transfer learning

A third class of alternative methods fits under the umbrella of regression-based transfer learning. Transfer learning is a large subfield of machine learning that aims to transfer knowledge from one domain to another. Without the aim to be complete we discuss three methods in the regression framework that apply to high-dimensional data; we refer to those papers for further reading. These three methods all transfer previously found regression coefficients *directly* to the regression coefficients at hand. Some of the discussed guided adaptive shrinkage methods also accommodate such continuous co-data (external regression coefficients), but these methods transfer this information *indirectly*: to the shrinkage instead of to the coefficients.

First, Boonstra et al. (2013) discuss targeted ridge shrinkage: $\boldsymbol{\beta}$ is shrunken to a non-zero mean $\boldsymbol{\beta}_0$, which are regression coefficients obtained from a similar prediction problem. Hence, somewhat similar in spirit as in (Kawaguchi et al., 2022), but the latter can use more general co-data to modify the target. Second, the prior lasso (Jiang et al., 2016) transfers predictions from a prior model into the objective function: it augments the likelihood with a weighted likelihood part that uses the predictions from the prior model as the outcome. This renders one estimate $\hat{\boldsymbol{\beta}}$ that accommodates both the primary data and, to a lesser extent, the prior data, the impact of which can be tuned. Third, `transreg` (Rauschenberger et al., 2023) estimates regression coefficients as a function of the prior coefficients available from co-data. It uses either a specific parameteric form, or a non-parametric one that respects monotony to ensure stability. It bags the prior-informed predictors, possibly from multiple co-data sources, with a prior-agnostic one to create a meta-learner, and optimizes the weights by efficient cross-validation.

4.4 Other related methods

Finally, we discuss a few other methods that are related to guided adaptive shrinkage. First, priority-lasso (Klau et al., 2018) also handles multiple groups of features (e.g. data modalities), but it prioritizes some groups over others, a strategy also explored by Aben et al. (2016). The idea is that a given data modality may have some practical advantages over other ones: e.g. (stable) DNA markers may be preferred over less stable mRNA markers. The Integrated Elastic Net (Culos et al., 2020) is used in the context of immune response prediction and is similar in spirit as fwelnet, although it optimizes the hyperparameters *separately* from $\boldsymbol{\beta}$ using cross-validation. It focuses on tensored binary co-data that codes for cell types, stimulations and type of response. In Yang et al. (2023), external information is also used to improve feature selection, but only after initial feature selection by lasso. It uses the prior information to determine a set of features that is relatively stable and relatively well in line with that information. Finally, (Aldahmani and Zoubeidi, 2020) present a graphical-group ridge. It uses a graphical model to determine network modules. Its nodes represent feature groups in a ridge regression with group-specific penalties.

5 Do-it-yourself for your favorite model

The guided adaptive shrinkage papers reviewed in Table 1 focus on the most popular penalties/priors, in particular variations of the elastic net. Many other penalties and priors have been proposed, which triggers the question whether these can easily be modified to allow for adaptive shrinkage. For many methods, this is indeed the case. For MCMC-based methods, one generic solution is extensively discussed in Van de Wiel et al. (2018), building upon an algorithm in Casella and George (2001). They show that the hyperparameters which link the co-data to the priors can be estimated by alternating MCMC with likelihood-based optimization. The method is conceptually straightforward, but can be very time-consuming, as it requires multiple MCMC runs. Below, we focus on a 'do-it-yourself' solution that is computationally much more efficient.

Our solution applies when the penalty corresponds to a prior with finite variance and a non-sparse data matrix X. Gene expression data usually satisfies the latter condition, whereas data on very rare mutations may not. van Nee et al. (2023b) prove that under these conditions a Gaussian approximation of the prior is asymptotically appropriate to estimate the hyperparameters $\boldsymbol{\alpha}$. This corresponds to ridge regression for which very efficient algorithms are available to determine the hyperparameters by maximizing the marginal likelihood. This method has been implemented for the elastic net in the R-packages xtune(Zeng et al., 2021) and squeezy(van Nee et al., 2023b). It can, however, also be used to estimate other penalties/priors using the following algorithm:

1. Determine co-data sources Z and define Gaussian (= ridge) variances $v_j^R = \lambda_j^{-1}$, with $f(\lambda_j) = Z_{\cdot j}\boldsymbol{\alpha}$
2. Estimate $\boldsymbol{\alpha}$, and hence v_j^R, in the Gaussian setting using xtune or squeezy
3. Equate the theoretical variance v_j of the desired prior to $\hat{v}_j^R$ to obtain feature-specific prior parameters
4. Estimated the high-dimensional model using those feature-specific prior parameters

Next, we give an example for the spike-and-slab prior. This prior is a versatile, natural and powerful prior for high-dimensional settings, in particular useful for Bayesian feature selection (Carbonetto and Stephens, 2012; Newcombe et al., 2014; Velten and Huber, 2019).

We illustrate the 'do-it-yourself' principle in this setting: the co-data guided spike-and-slab prior. For this purpose, we focus on the simplest formulation of the spike-and-slab prior:

$$\beta_j \sim (1-q)\delta_0 + qN(0,\tau^2), \tag{3}$$

with an appropriate prior on the global parameter τ^2. Carbonetto and Stephens (2012) derive a computationally very efficient variational Bayes algorithm to approximate posteriors of $\boldsymbol{\beta}$, and use its implementation, `varbvs`, for feature selection in very high-dimensional genetic studies. Now suppose one wishes to incorporate co-data sources $Z_{1.} = (z_{1j})_{j=1}^p$ and $Z_{2.} = (z_{2j})_{j=1}^p$ to modify the prior inclusion probability, q. E.g. $Z_{1.}$ presents a prior grouping of SNPs into two groups, while $Z_{2.}$ represents log-p-values for those SNPs in a previous study. Then, we may modify the prior to:

$$\beta_j \sim (1-q_j)\delta_0 + q_jN(0,\tau^2), \tag{4}$$

where the feature specific inclusion probability depends on the co-data for feature j: $Z_{.j} = (z_{1j}, z_{2j})$. This prior is available in `varbvs`, but requires specification of $(q_j)_{j=1}^p$. We now explain how to estimate this quantity with the existing software tools following the algorithm above.

First, we model the ridge penalties $\lambda_j = f_{\boldsymbol{\alpha}}(Z_{.j}) = \alpha_1 z_{1j} + \alpha_2 z_{2j}$. Second, we estimate (α_1, α_2) by using the R-package `xtune`. The reciprocal penalties render ridge-based variances $\hat{v}_j^R$. Third, we compute the theoretical prior variances v_j from (4). For that, denote the latent indicator $I_{\{\beta_j=0\}}$ by I_j. Then, we have for the prior variance of β_j:

$$v_j = V(\beta_j) = E_{I_j}[V[\beta_j|I_j]] + V_{I_j}[E[\beta_j|I_j]] = (1-q_j)*0 + q_j\tau^2 + 0 = q_j\tau^2.$$

Equating v_j to $\hat{v}_j^R$ renders relative estimates of q_j, as we have $q_j = Cv_j$, with C an unknown constant. Our benchmark is prior model (3), fitted by `varbvs` (Carbonetto and Stephens, 2012), which requires to specify q. We set $q = 0.01$, implying that we expect a fairly sparse signal with a prior 99% probability for β_j to equal 0. Then, to ensure a meaningful comparison with the benchmark model, we set $\bar{q} = p^{-1}\sum_{j=1}^p q_j = 0.01$, which determines C, resulting in absolute estimates of q_j. Fourth, these estimates are then used to define the feature-specific priors (4)and the spike-and-slab model is fit using the `varbvs` package.

As a proof of concept, we show the benefit of moderating the prior inclusion probabilities by co-data in a high-dimensional SNP setting (minor allele frequencies (MAF), simulated as in (Carbonetto and Stephens, 2012)). For samples $i = 1, \ldots, 500$ and features $j = 1, \ldots, p = 10000$, generate:

$$\begin{aligned} M_j &= 0.05 + 0.45U_j, U_j \sim^{\text{iid}} U[0,1] && \text{(MAFs)} \\ x_{ij} &\sim^{\text{iid}} \text{Bin}(2, M_j) && \text{(Allele counts)} \\ (\beta_j)_{j=1}^{150} &\sim^{\text{iid}} N(0, \tau^2 = 0.25); (\beta_j)_{j=151}^p = 0 && \text{(Coefficients)} \\ Y_i &= \sum_{j=1}^{10000} \beta_j x_{ij} && \text{(Response)} \end{aligned}$$

Furthermore, the co-data is generated as follows. The grouping $Z_{1.}$ contains two groups: a small group of size 500 features of which 100 features correspond to those with non-zero β_j's, and 400 correspond to those with β_j's equalling 0; and a large group of the remaining 9500 features, 50 of which correspond to non-zero β_j's. Therefore, the first group is enriched in signal. The second co-data source $Z_{2.}$ consists of log (external) p-values. The first 150 p-values are generated from a balanced mixture of two beta distributions: $\mathcal{B}(0.1, 10)$ and $\mathcal{B}(1, 5)$, and the remaining ones where generated from a uniform distribution. Hence, this co-data source should also be informative, as the non-zero β_j's tend to correspond to relatively small external p-values.

For this simulated data set the co-data is indeed very informative. The estimated hyperparameters $(\hat{\alpha}_1, \hat{\alpha}_2)$ render estimated prior inclusion probabilities $(\hat{q})_{j=1}^{150}$ with median: 0.136 and quartiles: (0.0096, 0.306) for relevant features. These are considerably higher than the summaries for the irrelevant ones, $(\hat{q})_{j=151}^{p}$, with median: 0.0031 and quartiles: (0.0028, 0.0078). Figure 4 shows the results for one simulated data set (as results are qualitatively very similar among multiple data sets). Not surprisingly, the improvement in feature selection performance with respect to the benchmark (no co-data) model is noticeable when either or both co-data sources are used.

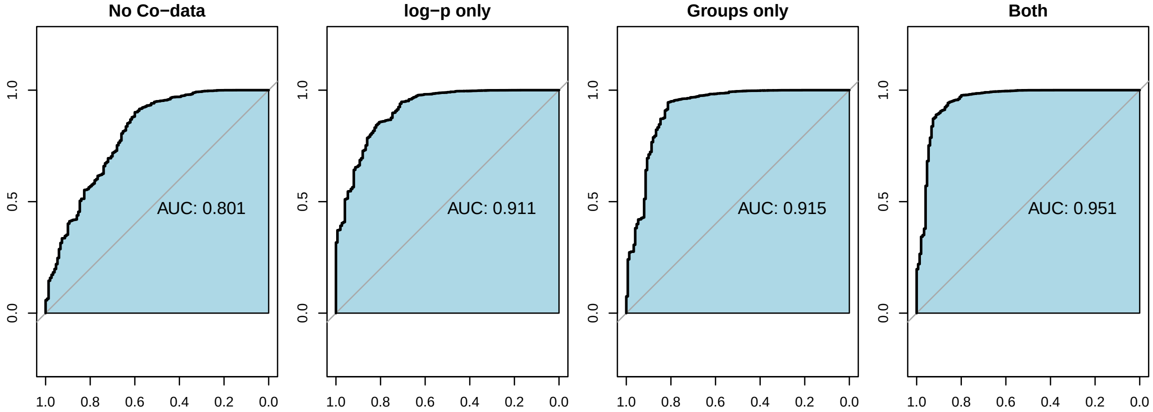

Figure 4: ROC curves for feature selection using spike-and-slab models. X-axis: specificity, y-axis: sensitivity.

6 Software

R-scripts to reproduce the results in this manuscript are available via: `https://github.com/markvdwiel/CodataReview/`.

7 Discussion

We reviewed methods that implement guided adaptive shrinkage, including group-adaptive methods. The latter framework was contrasted with group-regularized methods such as the sparse group-lasso. In genomics settings, multiple sources of co-data of mixed types, such as external p-values and known gene signatures, are often available. Fortunately, this setting is nowadays conveniently accommodated by several methods via a regression-type parametrization that links those co-data to the hyperparameters. We emphasize that the combination of *multiple* co-data can be a very powerful tool to improve feature selection, as was illustrated for the simulated SNP data. Therefore, tools that accommodate automatic retrieval of co-data are a very welcome addition to the methodology. For this purpose, Perscheid (2021) developed `Comprior`, which provides tools for extraction of gene and/or pathway scores or lists from several well-known genomic data bases. Moreover, it integrates with `xtune` to use such co-data for lasso-based feature selection. In addition, Wang et al. (2023) developed a module for automatic retrieval of gene-centered co-data from scientific articles. In short, they use a convolutional neural net based text analysis to learn a gene score for its relation to the disease of interest. This score may then be used as co-data for a given study.

An important criticism on guided adaptive shrinkage methods is that they may be prone to overfitting when the number of hyperparameters (size of $\boldsymbol{\alpha}$) is large. Full Bayesian methods like `graper` (Velten and Huber, 2019) and `infHS` (Busatto and van de Wiel, 2023) may counter this by using a (weakly) informative prior, whereas `ecpc` (van Nee et al., 2023a) allows to regularize the empirical Bayes moment equations to stabilize the estimation of $\boldsymbol{\alpha}$. For the group-adaptive lasso setting we provide a potential solution in the Supplementary material, based on targeted hyperparameter shrinkage.

We primarily focused on evaluating accuracy of feature selection, as most of the reviewed guided adaptive shrinkage methods contain extensive results on the potential of these methods for improving prediction. Moreover, it has been demonstrated that also the stability of the selected feature set improves with the use of co-data (van Nee et al., 2023a).

Finally, we note that the use of prior information is fundamental to science: 'science builds on science'. The plethora of guided shrinkage methods reviewed here provides researchers the means to structurally do so in high-dimensional -omics settings.

References

Aben, N. et al. (2016). TANDEM: a two-stage approach to maximize interpretability of drug response models based on multiple molecular data types. *Bioinformatics*, **32**, i413–i420.

Aldahmani, S. and Zoubeidi, T. (2020). Graphical group ridge. *Journal of Statistical Computation and Simulation*, **90**, 3422–3432.

Belhechmi, S. et al. (2020). Accounting for grouped predictor variables or pathways in high-dimensional penalized cox regression models. *BMC bioinformatics*, **21**, 1–20.

Bergersen, L.C. et al. (2011). Weighted lasso with data integration. *Stat. Appl. Genet. Mol. Biol.*, **10**, 1–29.

Boonstra, P.S. et al. (2013). Incorporating auxiliary information for improved prediction in high-dimensional datasets: an ensemble of shrinkage approaches. *Biostatistics*, **14**, 259–272.

Boulesteix, A.L. et al. (2017). IPF-LASSO: Integrative-penalized regression with penalty factors for prediction based on multi-omics data. *Comp. Math. Meth. Med.*, **2017**.

Boyle, E.A. et al. (2017). An expanded view of complex traits: From polygenic to omnigenic. *Cell*, **169**, 1177–1186.

Busatto, C. and van de Wiel, M.A. (2023). Informative co-data learning for high-dimensional horseshoe regression. *arXiv preprint arXiv:2303.05898*.

Carbonetto, P. and Stephens, M. (2012). Scalable variational inference for Bayesian variable selection in regression, and its accuracy in genetic association studies. *Bayesian Analysis*, **7**, 73–108.

Casella, G. and George, E.I. (2001). Empirical Bayes Gibbs sampling. *Biostatistics*, **2**, 485–500.

Culos, A. et al. (2020). Integration of mechanistic immunological knowledge into a machine learning pipeline improves predictions. *Nature machine intelligence*, **2**, 619–628.

Ignatiadis, N. and Lolas, P. (2020). σ-ridge: group regularized ridge regression via empirical bayes noise level cross-validation. *arXiv preprint arXiv:2010.15817*.

Jiang, Y. et al. (2016). Variable selection with prior information for generalized linear models via the prior lasso method. *Journal of the American Statistical Association*, **111**, 355–376.

Kawaguchi, E.S. et al. (2022). Hierarchical ridge regression for incorporating prior information in genomic studies. *Journal of data science: JDS*, **20**, 34.

Klau, S. et al. (2018). Priority-Lasso: a simple hierarchical approach to the prediction of clinical outcome using multi-omics data. *BMC bioinformatics*, **19**, 322.

Meinshausen, N. and Bühlmann, P. (2010). Stability selection. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, **72**, 417–473.

Münch, M.M. et al. (2021). Adaptive group-regularized logistic elastic net regression. *Biostatistics*, **22**, 723–737.

Neuenschwander, B. et al. (2010). Summarizing historical information on controls in clinical trials. *Clin Trials*, **7**, 5–18.

Newcombe, P.J. et al. (2014). Weibull regression with Bayesian variable selection to identify prognostic tumour markers of breast cancer survival. *Stat. Meth. Med. Res.*, **26**, 1–23.

Perscheid, C. (2021). Comprior: facilitating the implementation and automated benchmarking of prior knowledge-based feature selection approaches on gene expression data sets. *BMC bioinformatics*, **22**, 1–15.

Rauschenberger, A. et al. (2023). Penalized regression with multiple sources of prior effects. *Bioinformatics*, **39**, btad680.

Simon, N. et al. (2013). A sparse-group lasso. *J Comput Graph Stat*, **22**, 231–245.

Tai, F. and Pan, W. (2007). Incorporating prior knowledge of predictors into penalized classifiers with multiple penalty terms. *Bioinformatics*, **23**, 1775–1782.

Tay, J.K. et al. (2023). Feature-weighted elastic net: using "features of features" for better prediction. *Statistica Sinica*, **33**, 259.

Van de Wiel, M.A. et al. (2016). Better prediction by use of co-data: adaptive group-regularized ridge regression. *Statist Med*, **35**, 368–381.

Van de Wiel, M.A. et al. (2018). Learning from a lot: Empirical Bayes in high-dimensional prediction settings. *Scand J Statist*, pages 1–24.

van Nee, M.M. et al. (2021). Flexible co-data learning for high-dimensional prediction. *Statist Med*, **40**, 5910–5925.

van Nee, M.M. et al. (2023a). ecpc: an r-package for generic co-data models for high-dimensional prediction. *BMC bioinformatics*, **24**, 172.

van Nee, M.M. et al. (2023b). Fast marginal likelihood estimation of penalties for group-adaptive elastic net. *Journal of Computational and Graphical Statistics*, **32**, 950–960.

Velten, B. and Huber, W. (2019). Adaptive penalization in high-dimensional regression and classification with external covariates using variational bayes. *Biostatistics*. Kxz034.

Vinga, S. (2021). Structured sparsity regularization for analyzing high-dimensional omics data. *Briefings in Bioinformatics*, **22**, 77–87.

Wang, F. et al. (2023). Prior information-assisted integrative analysis of multiple datasets. *Bioinformatics*, **39**, btad452.

Yang, S. et al. (2023). TSPLASSO: A two-stage prior lasso algorithm for gene selection using omics data. *IEEE Journal of Biomedical and Health Informatics*.

Zeng, C. et al. (2021). Incorporating prior knowledge into regularized regression. *Bioinformatics*, **37**, 514–521.

Zhao, Z. and Zucknick, M. (2020). Structured penalized regression for drug sensitivity prediction. *Journal of the Royal Statistical Society Series C: Applied Statistics*, **69**, 525–545.

Zhu, L. et al. (2019). Bayesian indicator variable selection to incorporate hierarchical overlapping group structure in multi-omics applications. *The Annals of Applied Statistics*, **13**, 2611–2636.

Zou, H. (2006). The adaptive lasso and its oracle properties. *J. Amer. Statist. Assoc.*, **101**, 1418–1429.

8 Supplementary Material

9 Additional figures group-lasso vs group-adaptive lasso

Simulation settings. Figure 5 corresponds to a group-sparse setting: $p = 10,000, n = 200, G = 60, 99$; 5 groups contains signal.

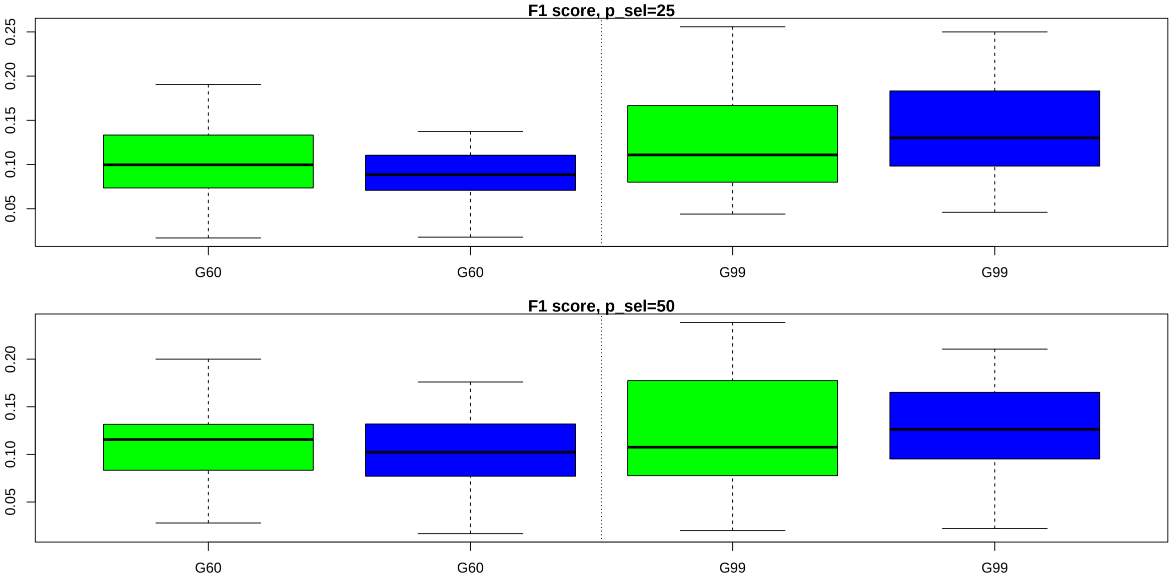

Figure 5: F1 score for variable selection ($p_{\text{sel}} = 25, 50$) for group-adaptive lasso (green) and sparse group-lasso (blue) for $G = 60, 99$ feature groups (25 simulations per G); group-sparse setting.

9.1 Targeted hyperparamer shrinkage for the group-adaptive lasso

Below, we discuss an extension of the group-adaptive lasso (Zeng et al., 2021; van Nee et al., 2023b) that target the shrinkage of the hyperparameters, the group-specific lasso penalties $(\lambda_g)_{g=1}^G$, by maximizing a penalized marginal likelihood based on a weakly informative prior. Previously, we showed that the marginal likelihood with lasso priors can be approximated by use of central Gaussian priors which second moments matched to those of the double-exponential (=lasso) priors (van Nee et al., 2023b). Therefore, for stabilizing the group-adaptive lasso it suffices to perform the hyperparameter shrinkage on the level of the group-level *ridge* penalties, which we denote by $(\lambda_g^R)_{g=1}^G$. For efficient optimization `squeezy` (van Nee et al., 2023b) determines the log-marginal likelihood and its gradient, which are both additive in terms of the hyperparameters, with respect to $\lambda_{\text{log}}^R = \lambda_{\text{log},g}^R = \log \lambda_g^R$. We augment the log-marginal likelihood by a penalty, which is simply the sum of the log-priors of $\log \lambda_g^R$. For the latter, we use a Laplace prior with location $\log \lambda_{\text{common}}^R$ as target and scale 1. We opt for fixing the latter as this renders a very efficient algorithm that does not require further tuning, while still covering a wide range of potential values of λ_g^R. The Laplace prior is used to accommodate group-sparse settings. The target is obtained by applying `squeezy` for the $G = 1$ setting, which is very fast.

As the Laplace prior has no gradient at its location parameter, we approximate the absolute value in this prior by $\sqrt{(x^2 + c)}$ using a small value of c. We experienced that this optimization is computationally very competitive to that of `squeezy`, which was reported to outperform other group-adaptive lasso methods, such as `gren` (Münch et al., 2021) and `ipf-lasso` (Boulesteix et al., 2017). In fact, we noticed that when the number of hyperparameters is large, the extra penalty helps to identify the optimum faster.

We implemented the approach by adapting `squeezy`. As over-fitting is particularly a concern when the grouping is not relevant at all, we first illustrate the results on this setting. To mimic such a setting we repeated the simulations as lined out in the Main Document for $G = 6, 15, 39, 60$, but with non-zero features randomly assigned to all groups. Then, one would want the performance of a group-adaptive method to be close to that of its non-group-adaptive counterpart, here the ordinary lasso. Figure 6 shows the results for feature selection. We observe that the targeted hyperparameter shrinkage improves results substantially in this 'null setting', although results are still somewhat inferior those of the lasso, as the latter accommodates this setting best. In addition, Figure 7 shows that the targeted shrinkage of hyperparameters renders only minor loss of feature selection performance as compared to group-adaptive lasso without hyperparameter shrinkage when the groups are informative in the aforementioned simulation settings.

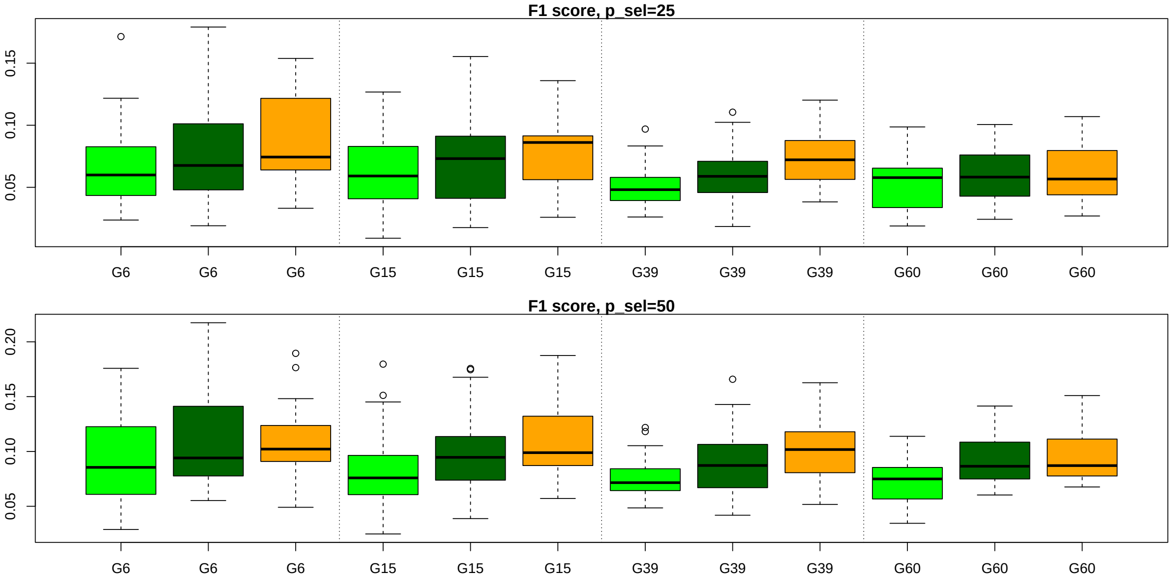

Figure 6: F1 score for variable selection ($p_{\text{sel}} = 25, 50$) for group-adaptive lasso (green), targeted group-adaptive lasso (dark-green) and lasso (orange) for $G = 6, 15, 39, 60$ *non-informative* feature groups (25 simulations per G)

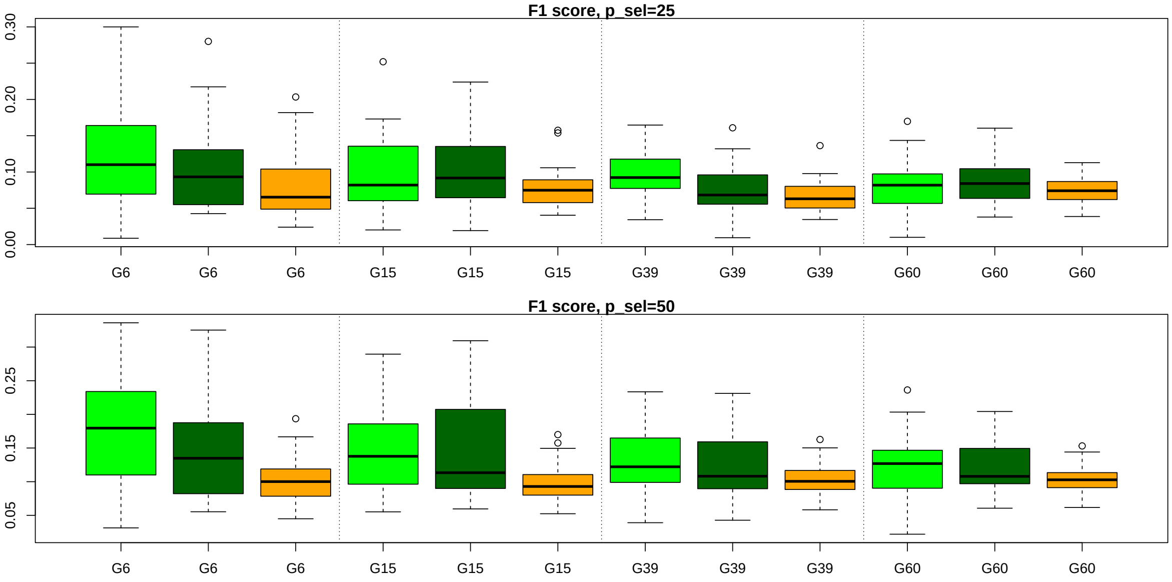

Figure 7: F1 score for variable selection ($p_{\text{sel}} = 25, 50$) for group-adaptive lasso (green), targeted group-adaptive lasso (dark-green) and lasso (orange) for $G = 6, 15, 39, 60$ informative feature groups (25 simulations per G)