Statistica Sinica Preprint No: SS-2016-0482R1	
Title	Supervised learning via the "hubNet" procedure
Manuscript ID	SS-2016.0482
URL	http://www.stat.sinica.edu.tw/statistica/
DOI	10.5705/ss.202016.0482
Complete List of Authors	Leying Guan Zhou Fan and Robert Tibshirani
Corresponding Author	Leying Guan
E-mail	lguan@stanford.edu
Notice: Accepted version subject to English editing.	

Statistica Sinica

Supervised learning via the "hubNet" procedure

Leying Guan[1], Zhou Fan[1], Robert Tibshirani[1,2]

Departments of Statistics[1] and Biomedical Data Sciences[2], Stanford University

Abstract: We propose a new method for supervised learning. The *hubNet* procedure fits a hub-based graphical model to the predictors, to estimate the amount of "connection" that each predictor has with other predictors. This yields a set of predictor weights that are then used in a regularized regression such as the lasso or elastic net. The resulting procedure is easy to implement, can often yield higher or competitive prediction accuracy with fewer features than the lasso, and can give insight into the underlying structure of the predictors.

HubNet can be generalized seamlessly to supervised problems such as regularized logistic regression (and other GLMs), Cox's proportional hazards model, and nonlinear procedures such as random forests and boosting. We prove recovery results under a specialized model and illustrate the method on real and simulated data.

HubNet; Adaptive Lasso; Graphical Model; Unsupervised Weights

1. Introduction

We consider the usual linear regression model: given n realizations of p predictors $\mathbf{X} = \{x_{ij}\}$ for $i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots, p$, the response

$Y = (y_1, \ldots, y_n)$ is modeled as

$$y_i = \beta_0 + \sum_j x_{ij}\beta_j + \epsilon_i \tag{1.1}$$

with $\epsilon \sim (0, \sigma^2)$. The ordinary least squares (OLS) estimates of β_j are obtained by minimizing the residual sum of squares. There has been much work on regularized estimators that offer an advantage over the OLS estimates, both in terms of accuracy of prediction on future data and interpretation of the fitted model. One major focus has been on the *lasso* (Tibshirani, 1996), which minimizes

$$J(\beta_0, \beta) = \frac{1}{2}\|Y - \beta_0 - \mathbf{X}\beta\|_2^2 + \lambda\|\beta\|_1 \tag{1.2}$$

where $\beta = (\beta_1, \ldots, \beta_p)$, and the tuning parameter $\lambda \geq 0$ controls the sparsity of the final model. This parameter is often selected by cross-validation. The objective function $J(\beta_0, \beta)$ is convex, which means that the solutions can be found efficiently even for very large n and p, in contrast to combinatorial methods like best subset selection. A body of mathematical work shows that under certain conditions, the lasso often will provide good recovery of the underlying true model and will produce predictions that are mean-square consistent (Knight and Fu, 2000; Meinshausen and Bühlmann, 2006; Zhao and Yu, 2006; Bunea et al., 2007; Zhang and Huang, 2008; Meinshausen and Yu, 2009; Bickel et al., 2009; Wainwright, 2009). The *elastic*

net of Zou and Hastie (2005) generalizes the lasso by adding an ℓ_2 penalty,

$$\frac{1}{2}\|Y - \beta_0 - \mathbf{X}\beta\|_2^2 + \lambda(\alpha\|\beta\|_1 + (1-\alpha)\|\beta\|_2^2), \tag{1.3}$$

where $\alpha \in [0, 1]$ is a second tuning parameter. This approach sometimes yields lower prediction error than the lasso, especially in settings with highly correlated predictors.

Zou (2006) introduced the *adaptive lasso*, which minimizes

$$\frac{1}{2}\|Y - \beta_0 - \mathbf{X}\beta\|_2^2 + \lambda \sum_j w_j|\beta_j| \tag{1.4}$$

for feature weights w_j. The feature weights can be chosen in various ways: For example, when $n > p$, we can first compute the OLS estimates $\hat{\beta}_j$ and then set $w_j = 1/|\hat{\beta}_j|$. For $p > n$, we can set w_j by first computing univariate regression coefficients (Huang et al., 2008). Other similar "two-step" procedures include variants of the non-negative garrote (Breiman, 1995; Yuan and Lin, 2007) and the adaptive elastic net (Zou and Zhang, 2009). One less-than-ideal property of these methods of feature weighting is that there is to no underlying generative model leading to the weights. Perhaps as a result, it is difficult to simulate datasets that show substantial gains relative to the usual lasso.

In this paper, we provide a new perspective by choosing weights in the adaptive lasso in an unsupervised manner. All of the above two-step proce-

dures select weights by computing an initial estimate $\hat{\beta}$ using the response Y. We instead propose to use the partial correlations of the features in $\mathbf{X}$ to select good weights.

Our proposal is based on an underlying conceptual model in which there is a core subset S of "hub" features that explains both the other features and Y. For example, each member of S might be the RNA or protein expression of a "driver" gene in a pathway which simultaneously influences other gene expressions and the phenotype under study. Our method, called *hubNet*, fits an (unsupervised) graphical model to the features in a way that tries to discover these "hubs". These features are then given higher weight in the adaptive lasso. The hubNet procedure can sometimes yield lower prediction error and better support recovery than the lasso, and the discovered hubs can provide insight on the underlying structure of the data.

The idea of first identifying structure in $\mathbf{X}$ before performing regression is similar to principal components regression (PCR), and the hub features identified by hubNet may be thought of as analogous to the principal directions in PCR. An important difference is that hubNet assigns weights to the original features, rather than combining them into new principal directions. This preserves the interpretability of the features, and also allows the method to be more robust to the possibility that some of the structure in

$\mathbf{X}$ may be unrelated to Y. Furthermore, performing PCR may be problematic if p is large, unless sparsity assumptions are imposed on the principal component loadings using sparse PCA methods (e.g., Zou et al. (2006); d'Aspremont et al. (2007)). Sparse PCA assumes a sparse covariance matrix for the p features, whereas our model assumes row-wise sparsity for the inverse covariance. The latter may be more suitable for certain applications.

This paper is organized as follows. In Section 2, we introduce our underlying model and the hubNet procedure. Section 3 examines applications to real datasets. Some theoretical results on the recovery of the underlying model are given in Section 4, while further topics, such as extensions to random forests, are discussed in Section 5.

1.1. Illustrative example: Olive oil data

The data for this example, from Forina et al. (1983), consists of measurements of 8 fatty acid concentrations for 572 olive oils, with each olive oil classified into one of two geographic regions. The goal is to determine the geographic region based on these 8 predictors. We randomly divided the data into training and test sets of equal size. Results from hubNet and lasso-regularized logistic regression are given in Figure 1. HubNet yields a more parsimonious model than the lasso, with perhaps lower error. More details are given in the caption. (Extension of hubNet to logistic regression

is straightforward and discussed in Section 2.3.)

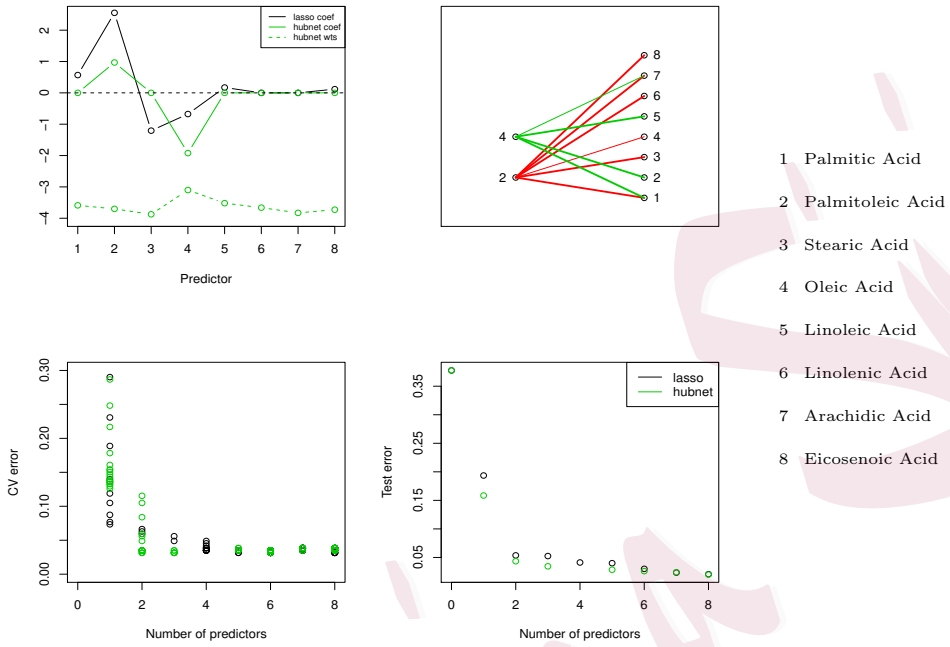

Figure 1: *Results from hubNet and lasso-regularized logistic regression. HubNet focuses on just two predictors—2 and 4, which have apparent connections to the other six. In the process, it yields a more parsimonious model than the lasso, with perhaps a lower CV and test error.*

2. The hubNet procedure

Let $Y = (y_1, \ldots, y_n)$ and let $\mathbf{X} = \{x_{ij}\}$ be the $n \times p$ matrix of features. Define the core set S to be a subset of $\{1, 2, \ldots p\}$, with corresponding

feature matrix $\mathbf{X}_S$. Our proposal is based on the following model:

$$Y = \beta_0 + \mathbf{X}_S\beta + \epsilon \tag{2.5}$$

$$X_j = \mathbf{X}_S\Gamma_j + \epsilon_j, \quad j \notin S \tag{2.6}$$

where each Γ_j is an $s \times 1$ coefficient vector. This model postulates that the outcome Y is a function of an (unknown) core set of predictors S, and that the predictors not in S are also a function of this same core set.

If this model holds, even approximately, then we can examine the partial correlations among the features to determine the features more likely to belong to this core set S, and hence do a better job of predicting Y. Following this logic, our proposal for estimating β in (2.5) consists of three steps:

The hubNet procedure

1. Fit a model of the form $\mathbf{X} \approx \mathbf{XB}$ with $\mathbf{B}_{ii} = 0$ using the "edge-out" procedure detailed in Section 2.1 below. Note that Γ_j in the generating model (2.6) correspond to coefficients of $\mathbf{B}$ in rows S and columns S^C.

2. Let $s_j = \|\hat{\mathbf{B}}_{j,\cdot}\|_2$ $(j = 1, \ldots, p)$ and construct feature weights $w_j = 1/s_j$.

3. Fit the adaptive lasso using feature weights w_j (e.g., using w_j as "penalty factors" in the `glmnet` R package.) [If $s_j = 0$, then $w_j = \infty$ and X_j is not used.]

The hubNet procedure has a number of attractive features:

(a) The construction of weights is completely unsupervised, separating it from the fitting of the response model in step 3. Thus for example, cross-validation can be applied in step 3, and we can use cross-validation to choose between hubNet and lasso for a given problem. In addition, tools for post-selection inference for the lasso can be directly applied.

(b) The supervised fitting in step 3 is simply a lasso (or elastic net) with feature weights, and hence fast off-the-shelf solvers can be used.

(c) Examination of the estimated hub structure for the chosen predictors can shed light on the structure of the final model.

(d) The procedure can be directly applied to generalized regression settings, such as generalized linear models and the proportional hazards model for survival data, using an appropriate method in step 3.

The challenging task of the hubNet procedure is to perform step 1 in a way that identifies the hub features. Applying the graphical lasso for this step, or performing an individual lasso regression to predict each feature using the others, can produce a sparse estimate of $\mathbf{B}$ corresponding to an edge-sparse feature graph. However, we would like a procedure that further encourages the appearance of hub nodes, i.e., features having many non-zero partial correlations with other features. These hub nodes then represent our estimate of the core set S. Tan et al.

(2014) propose a method called *hglasso* for learning graphical models with hubs. Their procedure uses an ADMM algorithm having computational complexity $O(p^3)$ per iteration, which is too slow for problems with $p = 1000$ or greater. We instead use a generalization of the (unpublished) "edge-out" method of Friedman et al. (2010), which has complexity $O(\min(np^2, sp^2))$ per iteration. Simulations comparing this edge-out method, hglasso, and individual lasso regressions for estimating $\mathbf{B}$ are given in the supplementary material.

2.1. The edge-out procedure

To estimate $\mathbf{B}$ in step 1 of the hubNet procedure, we use the edge-out estimator

$$\hat{\mathbf{B}}_{eo} = \underset{\mathbf{B}\in\mathbb{R}^{p\times p}:\, \mathbf{B}_{ii}=0\,\forall i}{\arg\min} \frac{1}{2}\|\mathbf{X} - \mathbf{X}\mathbf{B}\|_F^2 + \theta \sum_{i=1}^{p} \Big(\gamma\|\mathbf{B}_{i,\cdot}||_1 + (1-\gamma)\sqrt{p-1}\|\mathbf{B}_{i,\cdot}\|_2\Big) \quad (2.7)$$

Here, $\theta, \gamma > 0$ are tuning parameters, $\|\cdot\|_F$ denotes the Frobenius norm, and $\mathbf{B}_{i,\cdot}$ denotes the ith row of $\mathbf{B}$.

By constraining the diagonal entries of $\mathbf{B}$ to 0, the edge-out estimator simultaneously regresses each feature onto the remaining features of $\mathbf{X}$. These regressions are coupled by the ℓ_2 penalties $\|\mathbf{B}_{i,\cdot}\|_2$, which are group-lasso penalties that encourage zeroing-out of entire rows of $\mathbf{B}$. It is this coupling that leads to the appearance of hub nodes in the resulting estimate. The additional ℓ_1 penalties $\|\mathbf{B}_{i,\cdot}\|_1$ encourage additional sparsity in the non-zero rows of $\mathbf{B}$; we include this primarily for purposes of interpretability, to identify which features

are influenced by the hubs. (The original hubNet proposal of Friedman et al. (2010) used only the ℓ_2 penalty, i.e., $\gamma = 0$.)

The estimate $\hat{\mathbf{B}}_{eo}$ is not symmetric. We expect the "hub" features in the core set S to correspond to the rows of $\mathbf{B}$ having many non-zero entries, and hence the row sums should give higher weight to these features. Our procedure for minimizing this objective is outlined in the supplementary material.

2.2. Choosing tuning parameters for edge-out

We have two proposals for setting the tuning parameter θ in the edge-out method. The first is K-fold cross-validation, applied to the objective function $\frac{1}{2}||\mathbf{X} - \mathbf{XB}||_F^2$. The second uses a form of generalized cross validation

$$\mathrm{GCV}(\hat{\mathbf{X}}) = \frac{||\mathbf{X} - \hat{\mathbf{X}}||_2^2}{np - \mathrm{df}(\hat{\mathbf{X}})}.$$

If there is only an ℓ_1 penalty, we use for $\mathrm{df}(\hat{\mathbf{X}})$ the number of non-zero entries $|\hat{\mathbf{B}}|_0$. If there is also an ℓ_2 penalty, we propose the following adjustment based on our updating formula:

$$\mathrm{df}(\hat{\mathbf{X}}) = \sum_{i=1}^{p} \frac{\|\hat{\mathbf{B}}_{i,\cdot}||_2}{\|\hat{\mathbf{B}}_{i,\cdot}\|_2 + \theta(1-\gamma)\sqrt{p-1}} \|\hat{\mathbf{B}}_{i,\cdot}\|_0.$$

Note that this is not an exact formula for degrees of freedom, but rather a rough estimate.

2.3. Extension to generalized regression models

The hubNet procedure can be extended in a straightforward manner to the class of generalized linear models and other settings such as Cox's proportional hazards model. If the outcome Y depends on a parameter vector η, we assume that a core set of predictors S determines both η and the other predictors:

$$\begin{aligned} \eta &= \beta_0 + \mathbf{X}_S\beta \\ X_j &= \mathbf{X}_S\Gamma_j + \epsilon_j, \;\; j \notin S \end{aligned} \tag{2.8}$$

As in the linear case, we fit a model $\mathbf{X} = \mathbf{XB}$ using the edge-out procedure, and use the absolute row sums of $\hat{\mathbf{B}}$ as predictor weights in an ℓ_1-regularized (generalized) regression of Y on X.

For logistic regression, an alternative strategy would assume that a model of the form $X_j = \mathbf{X}_S\Gamma_j^k + \epsilon_j^k$ for $j \notin S$ holds within each class $k = 1, 2$. We may then estimate a hub model from the *pooled within class* covariance matrix of X, and use the absolute row sums as predictor weights.

2.4. Simulated data example.

Figure 2 shows hubNet applied to a simulated data example. Here $n = 60$, $p = 40$, and the first 3 predictors are the core set, explaining both Y and predictors 4 through 12. The estimated coefficients and various error rates of hubNet over 20 realizations are shown, in comparison to the elastic net, adaptive lasso, and lasso. We see that hubNet does a much better job at recovering the

true coefficients, which in turn leads to substantially lower prediction error. In Figure 3 we have generated data from an adversarial setting where the first 3 predictors are hub predictors, but the signal is a function of predictors 4 to 6. As expected, the hubNet procedure does poorly; however, its CV error is also high, so this poor behavior would be detectable in practice. Detailed comparisons between hubNet and other methods are given in the supplementary material – we found that hubNet produces better results not only when the generative model is true but also in several other settings with correlated predictors.

3. Application to real datasets

We compare hubNet with the lasso, elastic net, and/or principal components regression (PCR) on several real datasets. We tested ordinary PCR as well as sparse PCR using 10, 50, and 100 non-zero loadings. Results are shown for 100 non-zero loadings, corresponding to the lowest obtained test errors with cross-validated tuning parameter λ. Results for the other settings of PCR are reported in the supplementary materials.

Lipidomic breast cancer data: This data, from the lab of RT's collaborator Livia Schiavinato Eberlin at UT Austin, consists of 806 features measured on 15,359 pixels in tissue images from 24 breast cancer patients. The pixels are divided into two classes, normal and cancer, and we fit a regularized logistic regression model using each procedure. Cross-validation classification errors are

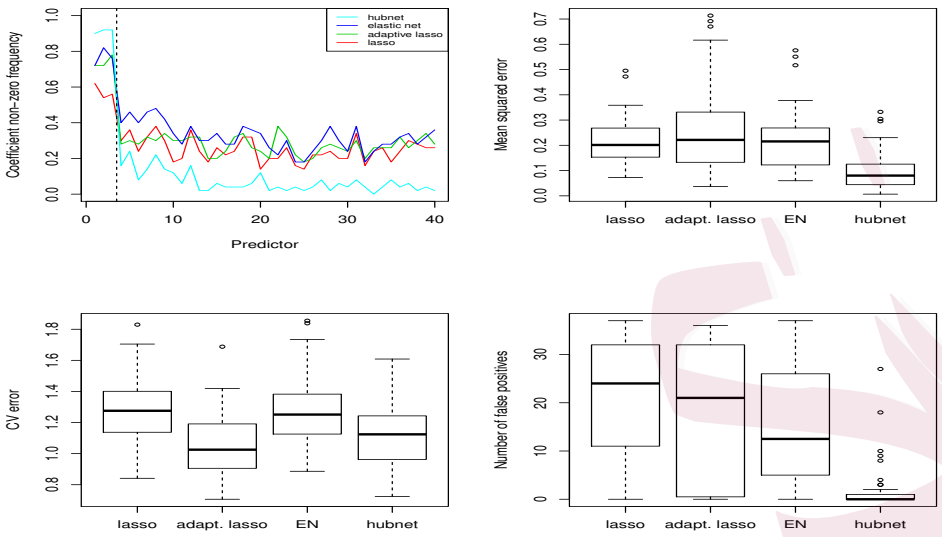

Figure 2: *Estimates from 20 simulations from a favorable underlying hub model;* $n = 60, p = 40$, *and the first 3 predictors are hub predictors that contain the signal and also influence predictors 4 through 12. The top left panel shows the fraction of simulations for which the estimated coefficient was non-zero. The top right panel displays the mean-squared test error with the tuning parameter chosen by cross-validation for each method. The bottom left panel shows the minimum CV error for each realization: note that the adaptive lasso CV error is not a valid estimate of error since the weights are estimated in a supervised manner. The bottom right panel shows the number of false positive predictors, in the smallest model where in the procedure has "screened", i.e. contains all of the true predictors.*

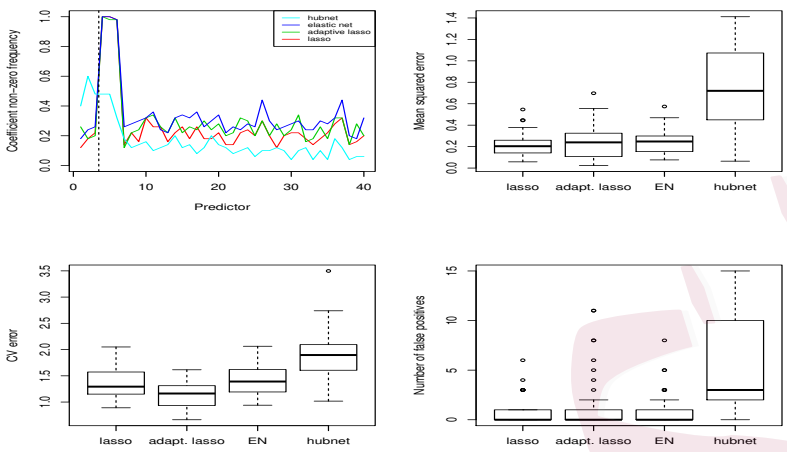

Figure 3: *Estimates from 20 simulations from an adverserial underlying hub model;* $n = 60, p = 40$*, first 3 predictors are hub predictors, but the signal is a function of predictors 4 to 6. See previous figure caption for details of panels.*

shown in Figure 4 as λ varies. Table 1 reports results for λ selected using 5-fold cross-validation.

B cell lymphoma gene expression data: This data from Rosenwald et al. (2002) consists of survival times (observed or right-censored) and 7399 gene expression features for 240 patients with diffuse large B-cell lymphoma (DLBCL). We divided the data with survival time $Y > 0$ into 156 training and 79 test samples, and trained a regularized proportional hazards model using each procedure. The p-value of the log-likelihood ratio (LR) statistic of this trained model evaluated on the test set is shown in the left subplot of Figure 5 as λ varies. Table 1 reports results for λ selected using 20-fold cross-validation.

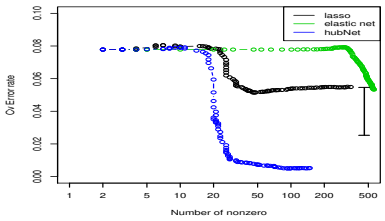

Figure 4: *Cross-validation classification error rates for breast cancer data. (The error bar represents one standard deviation of cross-validation error.)*

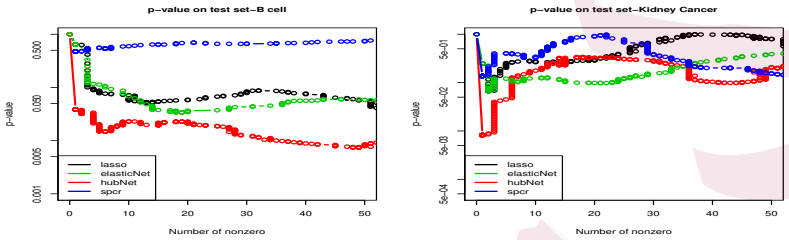

Figure 5: *Results for B-cell lymphoma (left) and kidney cancer (right): p-values of LR statistics*

Kidney cancer gene expression data: This data from Zhao et al. (2005) consists of survival times and 14,814 gene expression features for 177 patients with conventional renal cell carcinoma. We divided the data into 88 training samples and 89 test samples and trained a regularized proportional hazards model using each procedure. For computational reasons, hubNet was fit using the 7999 features with largest absolute row sum in the pairwise correlation matrix; lasso and elastic net were fit using all features. Test set LR p-values are shown in the right subplot of Figure 5 as λ varies, and Table 1 reports results for λ selected using 8-fold cross validation.

Table 1: *Comparisons among lasso, elasticNet and hubNet on three real data sets.*

		cvm(se)	Num. features	test error	common features (lasso)
Breast Cancer Data	lasso	5.15%(3.86%)	46	–	–
$p = 806$	elasticNet	5.85%(3.97%)	303	–	46
$n_{\text{train}} = 15359$	hubNet	3.52%(2.92%)	92	–	26
		cvm(se)	Num. features	test p-value	common features (lasso)
Kidney Cancer Data	lasso	9.90(0.59)	20	0.29	–
$p = 14814$	elasticNet	9.92(0.56)	24	0.11	4
$n_{\text{train}} = 88, n_{\text{test}} = 89$	hubNet	9.98(0.40)	1	0.008	0
	SPCR(100 non-zeros)	10.0(0.40)	1	0.137	–
		cvm(se)	Num. features	test p-value	common features (lasso)
DLBCL-patient Data	lasso	10.9(0.39)	29	0.076	–
$p = 7399$	elasticNet	10.9(0.39)	37	0.052	28
$n_{\text{train}} = 156, n_{\text{test}} = 79$	hubNet	10.9(0.36)	21	0.020	1
	SPCR(100 non-zeros)	11.07(0.26)	1	0.473	–

Table 1 summarizes the cross-validation errors, test errors, number of selected features, and number of such features in common with those selected by lasso.

4. Theory

In this section we study the recovery of the core set S assuming that our generating model (2.5, 2.6) holds. We first establish conditions under which the unsupervised edge-out procedure alone can recover S, and then discuss recovery of S by the second adaptive lasso step even if the edge-out procedure does not yield perfect recovery.

We assume the asymptotic regime $n, p \to \infty$ where $s \ll \min(n, p)$, as well

as a fully random design where the rows of $\mathbf{X}$ are independent and distributed as $N(0, \mathbf{\Sigma})$, normalized so that $\mathbf{\Sigma}_{jj} = 1$ for all $j = 1, \ldots, p$. Without loss of generality, we suppose S contains the first s predictors. By (2.6), if $X := (X_S, X_{S^C}) \sim N(0, \mathbf{\Sigma})$, then

$$X_S \sim N(0, \mathbf{\Sigma}_{SS}),$$
$$X_j | X_S \overset{ind}{\sim} N(X_S^T \Gamma_j, \sigma_j^2), \quad j \in S^C \tag{4.9}$$

where $\sigma_j^2 = \mathrm{Var}(\epsilon_j) \in (0, 1)$. Specifically, $\mathbf{\Gamma} := (\Gamma_{s+1}, \ldots, \Gamma_p)$ is given by $\mathbf{\Sigma}_{SS}^{-1} \mathbf{\Sigma}_{SS^C}$. We assume that this model holds in all of the results that follow.

4.1. Recovery of the core set using the edge-out procedure

We analyze recovery of S by the edge-out procedure applied with only the group-lasso penalty term in (2.7), corresponding to the setting $\gamma = 0$. For any matrix $\mathbf{M}$, denote by $\mathbf{M}_{i,\cdot}$ and $\mathbf{M}_{\cdot,j}$ the ith row and jth column of $\mathbf{M}$. We use the following operator norms which measure the maximum ℓ_1 and ℓ_2 norm of any row of $\mathbf{M}$:

$$\|\mathbf{M}\|_\infty := \sup_{\|x\|_\infty = 1} \|\mathbf{M}x\|_\infty = \max_i \|\mathbf{M}_{i,\cdot}\|_1, \quad \|\mathbf{M}\|_{\infty,2} := \sup_{\|x\|_2 = 1} \|\mathbf{M}x\|_\infty = \max_i \|\mathbf{M}_{i,\cdot}\|_2.$$

We define also the usual spectral norm, given by the largest singular value of $\mathbf{M}$: $\|\mathbf{M}\|_2 := \sup_{\|x\|_2 = 1} \|\mathbf{M}x\|_2 = \sigma_{\max}(\mathbf{M})$.

We show that in the asymptotic regime $n, p \to \infty$, the edge-out procedure can recover the true core set S for a suitable choice of the tuning parameter θ when the following conditions hold:

Assumption 4.1. *Let $\lambda_{\min}(\mathbf{\Sigma}_{SS})$ be the smallest eigenvalue of $\mathbf{\Sigma}_{SS}$. For a fixed constant $C_{\min} > 0$, we have $\lambda_{\min}(\mathbf{\Sigma}_{SS}) \geq C_{\min}$.*

Assumption 4.2. *Define $\mathbf{D} := \mathrm{diag}(1/\|\Gamma_{s+1}\|_2, \ldots, 1/\|\Gamma_p\|_2)$. For a fixed constant $\delta \in (0, 1]$, we have $\|\mathbf{\Gamma}^T \mathbf{D} \mathbf{\Gamma}\|_{\infty,2} \leq 1 - \delta$.*

Assumption 4.3. *(Number of hub nodes). The size s of the core set satisfies the constraint $s \ll \min(\sqrt{n}, n/\log p)$.*

Assumption 4.4. *(Hub strength). The minimum hub strength $\Gamma_{\min} = \min_i \|\mathbf{\Gamma}_{i,\cdot}\|_2$ satisfies $\Gamma_{\min} \gg \max(\|\mathbf{\Gamma}^T\|_\infty, 1)\|\mathbf{\Sigma}_{SS}^{-1}\|_\infty \max(1, \sqrt{p/n}, \sqrt{p \log p}/n)$.*

Under these assumptions, we can ensure perfect recovery of the core set S by the edge-out method:

Theorem 4.5. *Let $\hat{\mathbf{B}} := \hat{\mathbf{B}}_{eo}$ be the edge-out estimate in (2.7) applied with $\gamma = 0$, and denote $\hat{S} = \{i : \|\hat{\mathbf{B}}_{i,\cdot}\|_2 > 0\}$. Suppose Assumptions 4.1, 4.2, 4.3, and 4.4 hold. Defining $\theta_n = \theta\sqrt{p-1}/n$, if the tuning parameter θ is chosen so that*

$$\frac{\Gamma_{\min}}{\max(\|\mathbf{\Gamma}^T\|_\infty, 1)\|\mathbf{\Sigma}_{SS}^{-1}\|_\infty} \gg \theta_n \gg \max\left(1, \sqrt{\frac{p}{n}}, \frac{\sqrt{p \log p}}{n}\right), \tag{4.10}$$

then $P[\hat{S} = S] \to 1$.

Assumption 4.1 ensures that the hub features are not too correlated. Assumptions 4.3 and 4.4 restrict the maximal size of the core set and minimal "strength" of the hub features, as measured by the minimum ℓ_2 row norm of $\mathbf{\Gamma}$.

Let us remark that our normalization implies an additional implicit constraint on s, namely $p \geq \sum_{j\in S^C} \mathrm{Var}(X_j) = \sum_{j\in S^C} \Gamma_j^T \mathbf{\Sigma}_{SS}\Gamma_j + \sigma_j^2 \geq \|\mathbf{\Gamma}\|_F^2 C_{\min} \geq sC_{\min}\Gamma_{\min}^2$, so by Assumption 4.4

$$s \ll \frac{\min(n, p, n^2/\log p)}{\max(\|\mathbf{\Gamma}^T\|_\infty, 1)^2 \|\mathbf{\Sigma}_{SS}^{-1}\|_\infty^2}.$$

In the worst case, we have the upper bounds $\|\mathbf{\Sigma}_{SS}^{-1}\|_\infty \leq \sqrt{s}\|\mathbf{\Sigma}_{SS}^{-1}\|_2 \leq \sqrt{s}/C_{\min}$ and $\|\mathbf{\Gamma}^T\|_\infty \leq \sqrt{s}\|\mathbf{\Gamma}^T\|_{\infty,2} \leq \sqrt{s/C_{\min}}$, where the latter bound follows from our normalization condition

$$\|\mathbf{\Gamma}^T\|_{\infty,2}^2 C_{\min} \leq \max_{j\in S^C} \Gamma_j^T \mathbf{\Sigma}_{SS}\Gamma_j \leq \mathrm{Var}(X_j) \leq 1. \tag{4.11}$$

Assuming $\log p \ll \sqrt{n}$, recovery can occur in this worst case when $s \ll \min(n^{1/3}, p^{1/3})$. In the best case where an "irrepresentable condition" $\|\mathbf{\Gamma}^T\|_\infty \leq 1$ holds (see below) and $\mathbf{\Sigma}_{SS} = \mathbf{Id}$, then we have $\max(\|\mathbf{\Gamma}^T\|_\infty, 1)\|\mathbf{\Sigma}_{SS}^{-1}\|_\infty = 1$, and recovery can occur for $s \ll \min(\sqrt{n}, p)$.

Assumption 4.2 is analogous to but much weaker than the "irrepresentable condition" of Zhao and Yu (2006) (see also Wainwright (2009)) that is required for perfect support recovery by the standard lasso procedure. In our random design setting, the irrepresentable condition corresponds to

$$\|\mathbf{\Gamma}^T\|_\infty \leq 1 - \delta \tag{4.12}$$

for some $\delta \in (0, 1]$. When (4.12) holds, Assumption 4.2 is implied by $\|\mathbf{\Gamma}^T\mathbf{D}\mathbf{\Gamma}\|_{\infty,2} \leq \|\mathbf{\Gamma}^T\|_\infty \|\mathbf{D}\mathbf{\Gamma}\|_{\infty,2} = \|\mathbf{\Gamma}^T\|_\infty$. The following example illustrates that Assumption

4.2 is weaker than (4.12):

Example 4.6. *Suppose the entries of* $\mathbf{\Gamma}$ *are i.i.d. and equal to* $(1-2\delta)/\sqrt{s}$ *or* $-(1-2\delta)/\sqrt{s}$ *each with probability 1/2. Then* $\|\mathbf{\Gamma}^T\mathbf{D}\mathbf{\Gamma}\|_{\infty,2} \le \|\mathbf{\Gamma}^T\|_{\infty,2}\|\mathbf{D}\|_2\|\mathbf{\Gamma}\|_2 = \sqrt{s/(p-s)}\|\mathbf{\Gamma}\|_2$. *If* $p \to \infty$ *with* $s \ll p$, *the maximal singular value of* $\mathbf{\Gamma}$ *satisfies, for any fixed* $\varepsilon > 0$ *with probability approaching 1,* $\|\mathbf{\Gamma}\|_2 \le (1+\varepsilon)\sqrt{p}\cdot(1-2\delta)/\sqrt{s}$. *(See e.g. Theorem 5.39 of Vershynin (2012).) Hence for large p,* $\mathbf{\Gamma}$ *satisfies Assumption 4.2 with high probability. However,* $\|\mathbf{\Gamma}^T\|_\infty = (1-2\delta)\sqrt{s} \gg 1$.

This example shows that Assumption 4.2 can hold even in the worst-case setting where $\|\mathbf{\Gamma}^T\|_\infty \asymp \sqrt{s}$, as long as the non-hub features are not influenced by the hub features "in the same way".

4.2. Recovery of the core set using adaptive lasso

We now consider the linear model (2.5) where $\epsilon = (\epsilon_1, \ldots, \epsilon_p)$ is independent of $\mathbf{X}$ with $\epsilon_i \overset{iid}{\sim} N(0, \sigma^2)$. We study recovery of S by the adaptive lasso step of the hubNet procedure in two cases: (a) the edge-out estimate yields exact recovery of S , and (b) it yields a superset of S.

Let $w_1, \ldots, w_p \in (0, \infty]$ be any feature weights derived from $\mathbf{X}$. (Setting $w_i = \infty$ corresponds to $\|(\hat{\mathbf{B}}_{eo})_{i,\cdot}\|_2 = 0$, i.e. a hard constraint that requires $\beta_i = 0$.) Define

$$\rho := w_{\max}(S)/w_{\min}(S^C), \quad w_{\min}(S^c) := \min_{i \in S^c} w_i, \quad w_{\max}(S) := \max_{i \in S} w_i,$$

with the convention $\infty/\infty = \infty$. We consider the following conditions as $n, p \to$

∞:

Assumption 4.7. *There exists $\eta \in (0, 1]$ such that with probability approaching 1,*

$$\rho\sqrt{\frac{s}{C_{\min}}}\left(1+\sqrt{\frac{12\log p}{n}}\right) \leq 1-\eta.$$

Assumption 4.8. *The minimum predictor strength $\beta_{\min} = \min_{i\in S}|\beta_i^*|$ satisfies*

$$\beta_{\min} \gg \sigma\sqrt{\frac{s\log p}{n}\left(1+\frac{\log p}{n}\right)}.$$

Then, under our model (2.5) and (2.6), the following result holds for the adaptive lasso:

Theorem 4.9. *Let $n, p \to \infty$ such that $s \ll n$ and Assumption 4.1 holds. Furthermore, let $w_1, \ldots, w_p \in (0, \infty]$ be weights (depending on $\mathbf{X}$) such that Assumption 4.7 holds. Denote by $\hat{\beta}_0, \hat{\beta}$ the estimator minimizing the adaptive lasso objective (1.4), and let $\hat{S} = \{i : \hat{\beta}_i \neq 0\}$.*

(a) Denoting $\lambda_n = \lambda/n$, if the tuning parameter λ of the adaptive lasso is chosen such that

$$\lambda_n \gg \frac{1}{w_{\min}(S^C)}\sigma\sqrt{\frac{\log p}{n}\left(1+\frac{\log p}{n}\right)}$$

with probability approaching 1, then $P[\hat{S} \subseteq S] \to 1$.

(b) If, in addition, Assumption 4.8 holds and $\lambda_n \ll \beta_{\min}/(w_{\max}(S)\sqrt{s})$ with probability approaching 1, then $P(\hat{S} = S) \to 1$.

This result holds for any procedure that selects $w_1, \ldots, w_p$ using $\mathbf{X}$. Assumption 4.8 is comparable to the beta-min condition in Theorem 3 of Wainwright (2009) for the standard lasso procedure, if $\sqrt{s}$ is replaced by $\|\mathbf{\Sigma}_{SS}^{-1/2}\|_\infty^2$. In the context of hubNet, Assumption 4.7 should be interpreted as a weakening of the conditions required for selection consistency of S by the edge-out procedure alone: If the edge-out procedure successfully recovers S, then $w_{\min}(S^c) = \infty$ and $w_{\max}(S) < \infty$, so Assumption 4.7 holds. More generally, Assumption 4.7 holds when there is a separation in size between the rows of $\hat{\mathbf{B}}_{eo}$ belonging to S and to S^C, even if the rows belonging to S^C are not identically 0.

Proofs for Theorems 4.5 and 4.9 are given in the supplementary material. The proof of Theorem 4.9 is a simple application of the Sign Recovery Lemma in Zhou et al. (2009) for the adaptive lasso procedure. A more refined statement of Theorem 4.9 in terms of the quantities $\|\mathbf{\Gamma}^T\|_\infty$ and $\|\mathbf{\Sigma}_{SS}^{-1}\|_\infty$, similar to that of Theorem 4.5, is possible, although we have stated the above version for simplicity and interpretability.

5. Further topics

5.1. Adaptive, non-linear models

We can extend our basic model (2.6) to allow the dependence of Y on the core set of predictors to be of a more general form:

$$Y = f(\mathbf{X}_S) + \epsilon \tag{5.13}$$

$$X_j = \mathbf{X}_S\Gamma_j + \epsilon_j, j \notin S \tag{5.14}$$

Here $f(\cdot)$ is a general, non-linear function. For this model, we can estimate hub weights s_j as before and then apply a more flexible prediction procedure such as random forests or gradient boosting using the s_j as feature weights. With random forests, the candidate predictors for splitting are chosen at random. Hence it is natural to implement feature weighting by using the weights to determine the probabilities in this sampling. For example, the `ranger` package in R provides this option.

We tried this idea in the example of Figure 2, with additional interactions $.5x_1x_2$ and $-2x_2x_3$ added to the mean of Y, so that there were interactions for the random forest to find. We used sampling probabilities proportional to s_j^2. In Figure 6 we show the ratio of the mean squared error of the hubNet/RF over that for the vanilla random forest, as the error standard deviation σ is varied. We see that the hub weights can decrease the mean squared error by as much as 15%.

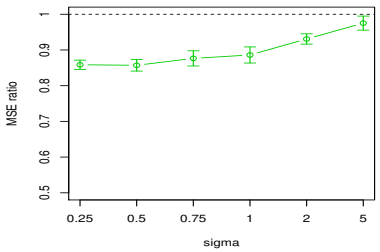

Figure 6: *MSE ratio of the hub-weighted random forest to the standard random forest, for varying error standard deviation*

5.2. Random forests: a drug discovery application

We consider classification data collected by the NCI, described in Feng et al. (2003) and analyzed further in Chipman et al. (2010). It consists of $p = 266$ molecular characteristics of $n = 29,374$ compounds, of which 542 were classified as active ($Y = 1$). These predictors represent topological aspects of molecular structure. We randomly created training and test sets of equal size, and for computational reasons we downsampled the class 0 cases to a set of size 2000 out of the 14,687 class 0s in the training set. We applied both random forests and hubNet/RF, using the `ranger` package in R. The results in Figure 7 show that the hubNet weighting can reduce the number of features by a factor of about 10 (down to 28) with barely any loss in accuracy, and these 28 features would not be detectable from standard RF importance scores (right panel).

6. Discussion

We have proposed a new procedure, hubNet, that is applicable to many supervised learning problems. The procedure estimates "hub weights" from the matrix of predictor values and then uses these weights in a supervised learning method such as the lasso or random forest.

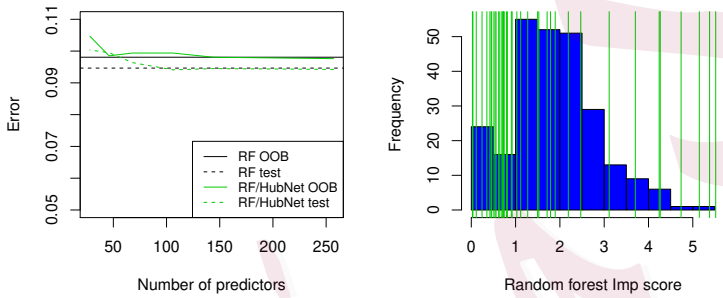

Figure 7: *Results for drug discovery dataset. Left panel show out-of-bag error and test error for vanilla random forest (horizontal lines), and the same for hubNet/RF as a function of the number of features having non-zero hub weights (by varying θ in the edge-out model). We see that the error increases very little, even as the number of number of features is reduced to about one-tenth (28) of the total number. These 28 features are indicated by the green lines in the right panel, superimposed on the RF impurity importance scores for all features.*

HubNet provides a way of utilizing structural information in the predictors, and it can yield more accurate prediction and support recovery in certain situ-

ations known to be hard if we neglect such knowledge. Since the estimation of weights is done in an unsupervised manner, both standard cross-validation and recently developed post-selection inference tools can be applied in the weighted fitting step. We observe in practice that this new procedure can sometimes yield lower prediction error than the unweighted approach, or give similar prediction error using fewer features. Moreover, the estimation of the hub structure can also be useful for interpretation.

Further work is needed in making the edge-out algorithm for hub estimation more efficient, so that it can be applied to very large datasets. An R language for hubNet will soon be available on the public CRAN repository.

Supplementary Material. This material contains: (*i*) the optimization algorithm for the edge-out model; (*ii*) proofs for Theorems 4.5 and 4.9; (*iii*) simulation comparisons between hubNet and other methods; (*iv*) comparisons between the edge-out model and hglasso.

References

Bickel, P. J., Ritov, Y., and Tsybakov, A. B. (2009). Simultaneous analysis of Lasso and Dantzig selector. *The Annals of Statistics*, 37(4):1705–1732.

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

Bunea, F., Tsybakov, A., and Wegkamp, M. (2007). Sparsity oracle inequalities for the Lasso.

Electronic Journal of Statistics, 1:169–194.

Chipman, H. A., George, E. I., and McCulloch, R. E. (2010). BART: Bayesian additive regression trees. *Ann. Appl. Stat.*, 4(1):266–298.

d'Aspremont, A., El Ghaoui, L., Jordan, M. I., and Lanckriet, G. R. (2007). A direct formulation for sparse pca using semidefinite programming. *SIAM review*, 49(3):434–448.

Feng, J., Lurati, L., Ouyang, H., Robinson, T., Wang, Y., Yuan, S., and Young, S. (2003). Predictive toxicology: Benchmarking molecular descriptors and statistical methods. *Journal of Chemical Information and Computer Sciences*, 43:1463–1470.

Forina, M., Armanino, C., Lanteri, S., and Tiscornia, E. (1983). Classification of olive oils from their fatty acid composition. *Food Research and Data Analysis*, pages 189–214.

Friedman, J., Hastie, T., and Tibshirani, R. (2010). Applications of the Lasso and grouped Lasso to the estimation of sparse graphical models. Technical report, Stanford University, Statistics Department.

Huang, J., Ma, S., and Zhang, C.-H. (2008). Adaptive Lasso for sparse high-dimensional regression models. *Statistica Sinica*, 18(4):1603–1618.

Knight, K. and Fu, W. (2000). Asymptotics for lasso-type estimators. *Annals of Statistics*, 28(5):1356–1378.

Meinshausen, N. and Bühlmann, P. (2006). High-dimensional graphs and variable selection with the Lasso. *The Annals of Statistics*, 34(3):1436–1462.

REFERENCES

Meinshausen, N. and Yu, B. (2009). Lasso-type recovery of sparse representations for high-dimensional data. *The Annals of Statistics*, 37(1):246–270.

Rosenwald, A., Wright, G., Chan, W. C., Connors, J. M., Campo, E., Fisher, R. I., Gascoyne, R. D., Muller-Hermelink, H. K., Smeland, E. B., and Staudt, L. M. (2002). The use of molecular profiling to predict survival after chemotherapy for diffuse large b-cell lymphoma. *The New England Journal of Medicine*, 346:1937–1947.

Tan, K. M., London, P., Mohan, K., Lee, S.-I., Fazel, M., and Witten, D. M. (2014). Learning graphical models with hubs. *Journal of Machine Learning Research*, 15(1):3297–3331.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B*, 58:267–288.

Vershynin, R. (2012). Introduction to the non-asymptotic analysis of random matrices. In Eldar, Y. C. and Kutyniok, G., editors, *Compressed Sensing*, pages 210–268. Cambridge University Press.

Wainwright, M. J. (2009). Sharp thresholds for high-dimensional and noisy sparsity recovery using-constrained quadratic programming (Lasso). *IEEE transactions on information theory*, 55(5):2183–2202.

Yuan, M. and Lin, Y. (2007). On the non-negative garrotte estimator. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 69(2):143–161.

Zhang, C.-H. and Huang, J. (2008). The sparsity and bias of the Lasso selection in high-dimensional linear regression. *The Annals of Statistics*, 36(4):1567–1594.

REFERENCES

Zhao, H., Tibshirani, R., and Brooks, J. (2005). Gene expression profiling predicts survival in conventional renal cell carcinoma. *PLOS Medicine*, pages 511–533.

Zhao, P. and Yu, B. (2006). On model selection consistency of Lasso. *Journal of Machine Learning Research*, 7(Nov):2541–2563.

Zhou, S., van de Geer, S., and Bühlmann, P. (2009). Adaptive Lasso for high dimensional regression and Gaussian graphical modeling. *arXiv preprint arXiv:0903.2515*.

Zou, H. (2006). The adaptive lasso and its oracle properties. *Journal of the American Statistical Association*, 101:1418–1429.

Zou, H. and Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society Series B*, 67(2):301–320.

Zou, H., Hastie, T., and Tibshirani, R. (2006). Sparse principal component analysis. *Journal of computational and graphical statistics*, 15(2):265–286.

Zou, H. and Zhang, H. H. (2009). On the adaptive elastic-net with a diverging number of parameters. *The Annals of Statistics*, 37(4):1733.

Stanford Univ, Depts. of Statistics

lguan@stanford.edu

Stanford Univ, Depts. of Statistics

zhoufan@stanford.edu

Stanford Univ, Depts. of Biomedical Data Sciences, and Statistics

tibs@stanford.edu

Acknowledgements Zhou Fan was supported by a Hertz Foundation Fellowship and an NDSEG Fellowship (DoD AFOSR 32 CFR 168a). Robert Tibshirani was supported by NIH grant 5R01 EB001988-16 and NSF grant DMS1208164.