EigenNet: A Bayesian hybrid of generative and conditional models for sparse learning

Yuan Qi
Computer Science and Statistics Depts.
Purdue University
West Lafayette, IN 47907, USA

Feng Yan
Computer Science Dept.
Purdue University
West Lafayette, IN 47907, USA

Abstract

For many real-world applications, we often need to select *correlated* variables—such as genetic variations and imaging features associated with Alzheimer's disease—in a high dimensional space. The correlation between variables presents a challenge to classical variable selection methods. To address this challenge, the elastic net has been developed and successfully applied to many applications. Despite its great success, the elastic net does not exploit the correlation information embedded in the data to select correlated variables. To overcome this limitation, we present a novel hybrid model, EigenNet, that uses the eigenstructures of data to guide variable selection. Specifically, it integrates a sparse conditional classification model with a generative model capturing variable correlations in a principled Bayesian framework. We develop an efficient active-set algorithm to estimate the model via evidence maximization. Experimental results on synthetic data and imaging genetics data demonstrate the superior predictive performance of the EigenNet over the lasso, the elastic net, and the automatic relevance determination.

1 Introduction

In this paper we consider the problem of selecting correlated variables in a high dimensional space. Among many variable selection methods, the lasso and the elastic net are two popular choices (Tibshirani, 1994; Zou and Hastie, 2005). The lasso uses a l_1 regularizer on model parameters. This regularizer shrinks the parameters towards zero, removing irreverent variables and yielding a sparse model (Tibshirani, 1994). However, the l_1 penalty may lead to over-sparisification: given many correlated variables, the lasso often only select a few of them. This not only degenerates its prediction accuracy but also affects the interpretability of the estimated model. For example, based on high-throughput biological data such as gene expression and RNA-seq data, it is highly desirable to select multiple correlated genes associated with a phenotype since it may reveal underlying biological pathways. Due to its over-sparsification, the lasso may not be suitable for this task. To address this issue, the elastic net has been developed to encourage a grouping effect, where strongly correlated variables tend to be in or out of the model together (Zou and Hastie, 2005). However, the grouping effect is just the result of its composite l_1 and l_2 regularizer; the elastic net does not explicitly incorporate correlation information among variables in its model.

In this paper, we propose a new sparse Bayesian hybrid model to utilize the eigen-information extracted from data for the selection of correlated variables. Specifically, it integrates a sparse *conditional* classification model with a *generative* model in a principle Bayesian framework (Lasserre et al., 2006): the conditional model achieves sparsity via automatic relevance determination (ARD) (MacKay, 1991), an empirical Bayesian approach for model sparisification; and the generative model is a latent variable model in which the observations are the eigenvectors of the unlabeled data, capturing correlations between variables. By integrating these two models together, the hybrid

model enables identification of groups of correlated variables guided by the eigenstructures. At the same time, the model passes the information from its conditional part to its generative part, selecting informative eigenvectors for the classification task. Furthermore, using the Bayesian hybrid model, we can automate the estimation of model hyperparameters.

From the regularization perspective, the new hybrid model naturally generalizes the elastic net using a composite regularizer adaptive to the data eigenstructures. It contains a sparsity regularizer and a directional regularizer that encourages selecting variables associated with eigenvectors chosen by the model. When the variables are independent of each other, the eigenvectors are parallel to the axes and this composite regularizer reduces to the combination of the ARD and a l_2 regularizer (similar to the composite regularizer of the elastic net). But when some of the input variables are strongly correlated, the regularizer will encourage the classifier aligned with eigenvectors selected by the model. On one hand, our model is like the elastic net to retain 'all the big fish'. On the other hand, our model is different from the elastic net by the guidance from the eigen-information. Hence the name EigenNet.

Experiments on synthetic data are presented in Section 5. Our results demonstrate that the EigenNet significantly outperforms the lasso, and the elastic net in terms of prediction accuracy. We applied this new approach to two tasks in imaging genetics: i) predicting cognitive function of healthy subjects and AD patients based on brain imaging markers, and ii) classifying the healthy and AD subjects based on single-nucleotide polymorphism (SNP) data. Compared to the lasso, the elastic net and the ARD, our approach achieves improved prediction accuracy.

2 Background: lasso and elastic net

We denote n independent and identically distributed samples as $\mathcal{D} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$, where $\mathbf{x}_i$ is a p dimensional input features (i.e., explanatory variables) and y_i is a scalar label (i.e., response). Also, we denote $[\mathbf{x}_1, \ldots, \mathbf{x}_n]$ by $\mathbf{X}$ and $(\mathbf{y}_1, \ldots, \mathbf{y}_n)$ by $\mathbf{y}$. Although our presentation focuses on the binary classification problem ($y_i \in \{-1, 1\}$), our approach can be readily applied to other problems such as regression and survival analysis by choosing appropriate likelihood functions.

For classification, we use a probit model as the data likelihood:

$$p(\mathbf{y}|\mathbf{X}, \mathbf{w}) = \prod_{i=1}^{n} \sigma(y_i \mathbf{w}^{\mathrm{T}} \mathbf{x}_i) \tag{1}$$

where $\sigma(z)$ is the Gaussian cumulative distribution function and $\mathbf{w}$ denotes the classifier.

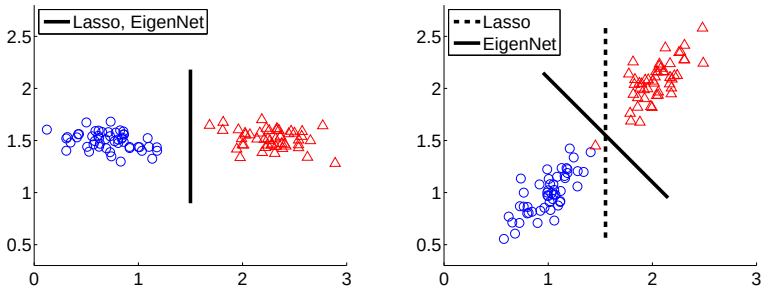
To identify relevant variables for high dimensional problems, the lasso (Tibshirani, 1994) uses a l_1 penalty, effectively shrinking $\mathbf{w}$ and b towards zero and pruning irrelevant variables. In a probabilistic framework this penalty corresponds to a Laplace prior distribution:

$$p(\mathbf{w}) = \prod_{j} \lambda \exp(-\lambda |w_j|) \tag{2}$$

where λ is a hyperparameter that controls the sparsity of the estimated model. The larger the hyperparameter λ, the sparser the model.

As described in Section 1, the lasso may over-penalize relevant variables and hurt its predictive performance, especially when there are strongly correlated variables. To address this issue, the elastic net (Zou and Hastie, 2005) combines l_1 and l_2 regularizers to avoid the over-penalization. The combined regularizer corresponds to the following prior distribution, $p(\mathbf{w}) \propto \prod_j \exp(-\lambda_1 |w_j| - \lambda_2 w_j^2)$, where λ_1 and λ_2 are hyperparameters. While it is well known that the elastic net tends to select strongly correlated variables together, it does not uses correlation information embedded in the unlabeled data. The selection of correlated variables is merely the result of a less aggressive regularizer for sparisty.

Besides the elastic net, there are many variants (and extensions) to the lasso, such as the bridge (Frank and Friedman, 1993) and smoothly clipped absolute deviation (Fan and Li, 2001). These variants modify the l_1 penalty to improve variable selection, but do not explicitly use the correlation information embedded in data.



(a) Independent variables (b) Correlated variables

Figure 1: Toy examples. (a) When the variables x^1 and x^2 are independent of each other, both the lasso and the EigenNet select only x^1. (b) When the variables x^1 and x^2 are correlated, the lasso selects only one variable. By contrast, guided by the major eigenvector of the data, the EigenNet selects both variables.

3 EigenNet: eigenstructure-guided variable selection

In this section, we propose to use the covariance structure in data to guide the sparse estimation of model parameters. First, let us consider the following toy examples.

3.1 Toy examples

Figure 1(a) shows samples from two classes. Clearly the variables x^1 and x^2 are not correlated. The lasso or the elastic net can successfully select the relevant variable x^1 to classify the data. For the samples in Figure 1(b), the variables x^1 and x^2 are strongly correlated. Despite the strong correlation, the lasso would select only x^1 and ignore x^2. The elastic net may select both x^1 and x^2 if the regularization weight λ_1 is small and λ_2 is big, so that the elastic net behaves like l_2 regularized classifier. The elastic net, however, does not explore the fact that x^1 and x^2 are correlated.

Since the eigenstructure of the data covariance matrix captures correlation information between variables, we propose to not only regularize the classifier to be sparse, but also encourage it to be aligned with certain eigenvector(s) that are helpful for the classification task. Note that although classical Fisher linear discriminant also uses the data covariance matrix to learn the classifier, it generally does not provide a sparse solution, thus not suitable for the task of selecting correlated variables and removing irrelevant ones.

For the data in Figure 1(a), since the two eigenvectors are parallel to the horizontal and vertical axes, the EigenNet essentially reduces to the elastic net and selects x^1. For the data in Figure 1(b), the principle eigenvector can guide the EigenNet to select both x^1 and x^2. The minor eigenvector is, however, not useful for the classification task (in general, we need to select which eigenvectors are relevant to classification). We use a Bayesian framework to materialize the above ideas as described in the following section.

3.2 Bayesian hybrid of conditional and generative models

The EigenNet is a hybrid of conditional and generative models. The conditional component allows us to learn the classifier via "discriminative" training; the generative component captures the correlations between variables; and these two models are glued together via a joint prior distribution, so that the correlation information is used to guide the estimation of the classifier and the classification task is used to choose or scale relevant eigenvectors. Our approach is based on the general Bayesian framework proposed by Lasserre et al. (2006)), which allows one to combine conditional and generative models in an elegant principled way.

Specifically, for the conditional model we have the same likelihood as (1), $p(\mathbf{y}|\mathbf{X}, \mathbf{w}) = \prod_i \sigma(y_i \mathbf{w}^{\mathrm{T}} \mathbf{x}_i)$. For the classifier $\mathbf{w}$, we use a Gaussian prior: $p(\mathbf{w}) = \prod_{j=1}^{p} \mathcal{N}(w_j|0, \beta_j^{-1})$. We will describe later how to efficiently learn the precision parameter β_j from the data to obtain a sparse classifier.

To encourage the classifier aligned with certain eigenvectors, we introduce $\tilde{\mathbf{w}}$—a latent vector (tightly) linked to the classifier $\mathbf{w}$—in the generative model:

$$p(\mathbf{V}|\mathbf{s}, \tilde{\mathbf{w}}) \propto \prod_{j=1}^{m} \mathcal{N}(\mathbf{v}_j | s_j \tilde{\mathbf{w}}, (\lambda_v \eta_j)^{-1} \mathbf{I}) \quad (3)$$

where $\mathbf{v}_j$ and η_j are the j-th eigenvector and eigenvalue of the data covariance matrix, λ_v is a hyperparameter, $\mathbf{s} = [s_1, \ldots, s_m]$ are scaling factors for the parameter $\tilde{\mathbf{w}}$. To combat overfitting, we assign a Gamma prior $\text{Gam}(\lambda_v | c_0, d_0)$ over λ_v. Note that this generative model encourages $\tilde{\mathbf{w}}$ to align with the major eigenvectors with bigger eigenvalues. However, eigenvectors are noisy and not all of them relevant to the classification task—we need to select relevant eigenvectors (i.e. the relevant sub-eigenspace) and remove irrelevant ones.

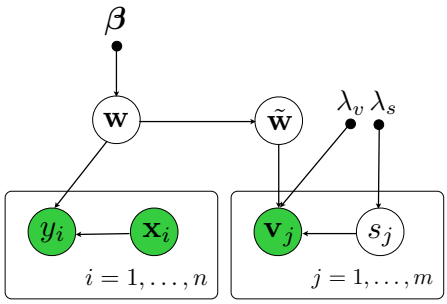


Figure 2: The graphical model of the EigenNet.

To enable the selection of the relevant eigenvectors, we assign a Laplace prior on s_j:

$$p(\mathbf{s}) \propto \prod_{j=1}^{m} \lambda_s \exp(-\lambda_s |s_j|) \quad (4)$$

where λ_s is a hyperparameter.

Finally, to link the conditional and generative models together, we use a prior for $\tilde{\mathbf{w}}$ conditional on $\mathbf{w}$:

$$p(\tilde{\mathbf{w}}|\mathbf{w}) \propto \mathcal{N}(\tilde{\mathbf{w}}|\mathbf{w}, r\mathbf{I}) \quad (5)$$

Note that the variance parameter r controls how similar $\mathbf{w}$ and $\tilde{\mathbf{w}}$ are in our joint model. For simplicity, we set $r = 0$ here so that $p(\tilde{\mathbf{w}}|\mathbf{w}) = \delta(\tilde{\mathbf{w}} - \mathbf{w})$ where $\delta(a) = 1$ if $a = 1$ and $\delta(a) = 0$ otherwise. The graphical model representation of the EigenNet is given in Figure 2.

3.3 Model estimation

In this section we present how to estimate the model based on an empirical Bayesian approach. Specifically, we will use expectation propagation (EP) (Minka, 2001) to estimate the posterior of the classifier $\mathbf{w}$ (and $\tilde{\mathbf{w}}$) and optimize the marginal likelihood of the joint model over the scaling variables $\mathbf{s}$ and the precision parameters $\boldsymbol{\beta}$.

First, given the hyperparameter λ_v and the latent variable $\mathbf{s}$, the posterior distribution of $\mathbf{w}$ is

$$p(\mathbf{w}|\mathbf{y}, \mathbf{X},) \propto \mathcal{N}(\mathbf{w}|\mathbf{0}, \text{diag}(\boldsymbol{\beta})^{-1}) \Big(\prod_i \sigma(y_i \mathbf{w}^{\mathrm{T}} \mathbf{x}_i) \Big) \prod_j \mathcal{N}(\mathbf{v}_j | s_j \mathbf{w}, (\lambda_v \eta_j)^{-1} \mathbf{I}) \quad (6)$$

$$\propto \mathcal{N}(\mathbf{w}|\mathbf{m}_p, \mathbf{V}_p) \prod_i \sigma(y_i \mathbf{w}^{\mathrm{T}} \mathbf{x}_i) \quad (7)$$

where $\mathbf{V}_p = (\text{diag}(\boldsymbol{\beta} + \lambda_v \sum_j \eta_j s_j^2 \mathbf{I}))^{-1}$ and $\mathbf{m}_p = \lambda_v \sum_j \eta_j s_j \mathbf{v}_j$. Then we initialize the EP updates by $p(\mathbf{w}) = \mathcal{N}(\mathbf{w}|\mathbf{m}_p, \mathbf{V}_p)$ and then iteratively approximate each likelihood factor $\sigma(y_i \mathbf{w}^{\mathrm{T}} \mathbf{x}_i)$ by a factor with the Gaussian form: $\mathcal{N}(t_i | \mathbf{x}_i^{\mathrm{T}} \mathbf{w}, h_i^{-1})$. In other words, EP maps the nonlinear non-Gaussian factor to the Gaussian factor with the virtual observation t_i and the noise variance h_i^{-1}. After the convergence of EP, we obtain both the mean $\mathbf{m}_w$ and the covariance $\mathbf{V}_w$.

Given the approximate posterior $q(\mathbf{w})$, we maximize the variational lower bound over λ_v:

$$L(\lambda_v) = \mathbf{E}_{q_w}[\sum_j \ln \mathcal{N}(\mathbf{v}_j | s_j \mathbf{w}, (\lambda_v \eta_j)^{-1} \mathbf{I}) + \ln \text{Gam}(\lambda_v | c_0, d_0)] \quad (8)$$

$$= \frac{pm}{2} \ln \lambda_v - \frac{F}{2} \lambda_v + (c_0 - 1) \ln \lambda_v - d_0 \lambda_v + \text{contant}$$

Algorithm 1 The empirical Bayesian estimation algorithm

1. Initialize the model to contain a small fraction of features and initialize the parameters: $\mathbf{s} = \mathbf{0}$, $\lambda_v = 1$, $\mathbf{t} = \mathbf{0}$ $\mathbf{h} = \infty$.
2. Run EP to obtain the initial mean and the covariance $\mathbf{m}_w$ and $\mathbf{V}_w$.
3. Loop until convergence or reaching the maximum number of iterations
  4. Loop over the j-th active set
    a. Update $\boldsymbol{\beta}$ via (12) and (13).
    b. If $u_j^2 < r_j$, remove the features in the j-th active set from the model
    c. Update the posterior mean $\mathbf{m}_w$ and the covariance $\mathbf{V}_w$ based on EP.
    d. Optimize the precision parameter λ_v via (9).
    e. Optimize the scaling factors $\mathbf{s}$ via (11).

where $F = \sum_j \eta_j - 2(\sum_j \mathbf{v}_j \eta_j s_j)^{\mathrm{T}} \mathbf{m}_w + \sum_j \eta_j s_j^2((\mathbf{m}_w)_i^2 + (\mathbf{V}_w)_{i,i})$. As a result, we have

$$\lambda_v = \frac{c_0 - 1 + pm/2}{d_0 + F/2}. \tag{9}$$

Similarly, we maximize the variational lower bound over $\mathbf{s}$:

$$L(\mathbf{s}) = \sum_j \left(\mathbf{E}_{q_w}[\ln \mathcal{N}(\mathbf{v}_j | s_j \mathbf{w}, (\lambda_v \eta_j)^{-1} \mathbf{I})] - \lambda_s |s_j| \right) + \text{contant}. \tag{10}$$

Consequently we have for each j,

$$\text{if } |\mathbf{v}_j^{\mathrm{T}} \mathbf{m}_w| < \frac{\lambda_s}{\eta_j \lambda_v},\ s_j = \text{Sign}(\mathbf{v}_j^{\mathrm{T}} \mathbf{m}_w) \frac{|\mathbf{v}_j^{\mathrm{T}} \mathbf{m}_w| - \lambda_s/(\eta_j \lambda_v)}{(\mathbf{m}_w)_i^2 + (\mathbf{V}_w)_{i,i}};\ \text{otherwise}, s_j = 0. \tag{11}$$

To estimate $\boldsymbol{\beta}$, we develop an active-set method to iteratively maximize the model marginal likelihood over elements of $\boldsymbol{\beta}$. In particular, we use a strategy similar to Tipping and Faul (2003)'s approach: given the approximation factors $\mathcal{N}(\mathbf{t}|\mathbf{X}^{\mathrm{T}}\mathbf{w}, \mathrm{diag}(\mathbf{h})^{-1})$, the distribution over eigenvectors $\mathcal{N}(\mathbf{v}_j | s_j \mathbf{w}, (\lambda_v \eta_j)^{-1} \mathbf{I})$, and the prior distribution $\mathcal{N}(\mathbf{w}|\mathbf{0}, \mathrm{diag}(\boldsymbol{\beta})^{-1})$, we can compute and decompose the log marginal likelihood $L(\boldsymbol{\beta}) = \log p(\mathbf{y}|\mathbf{X}, \mathbf{s}, \lambda_v)$ into two parts: $L(\beta_j)$ and $L(\boldsymbol{\beta}_{\setminus j})$ where j and $\setminus j$ index the elements of $\boldsymbol{\beta}$ in the active set and the rest elements, respectively. Note that because the effective prior over $\mathbf{w}$ becomes $\mathcal{N}(\mathbf{w}|\mathbf{m}_p, \mathbf{V}_p)$ as in (7) — instead of the zero mean prior $\mathcal{N}(\mathbf{w}|\mathbf{0}, \mathrm{diag}(\boldsymbol{\beta})^{-1})$— we cannot apply the algorithm proposed by Tipping and Faul (2003). Instead, we decompose $L(\boldsymbol{\beta})$ into $L(\beta_j)$ and $L(\boldsymbol{\beta}_{\setminus j})$ as follows.

First let us define

$$U_j = \mathbf{t}^{\mathrm{T}} \mathrm{diag}(\mathbf{h}) \mathbf{x}^j + \lambda_v \sum_{k=1}^m \eta_k s_k v_k^j - \mathbf{b}^{\mathrm{T}} \mathbf{m}_w,\ R_j = (\mathbf{x}^j)^{\mathrm{T}} \mathrm{diag}(\mathbf{h}) \mathbf{x}^j + \lambda_v \sum_{k=1}^m \eta_k s_k^2 - \mathbf{b}^{\mathrm{T}} \mathbf{V}_w \mathbf{b}$$

$$u_j = \frac{\beta_j U_j}{\beta_j - R_j} \qquad\qquad r_j = \frac{\beta_j R_j}{\beta_j - R_j} \tag{12}$$

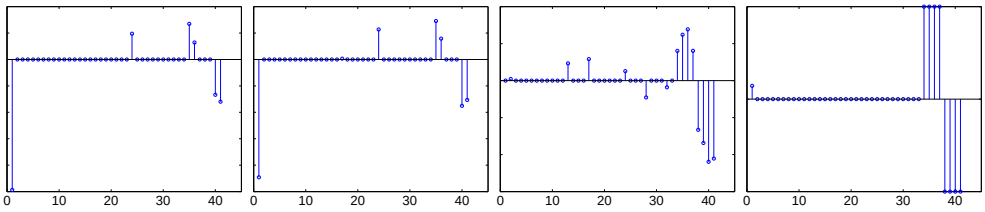
where $\mathbf{b} = (\mathbf{x}^j)^{\mathrm{T}} \mathrm{diag}(\mathbf{h}) \mathbf{X}^a + \lambda_v \mathbf{e}_j^a \sum_{k=1}^m \eta_k s_k^2$, $\mathbf{x}^j$ is the j-th column of the data matrix $\mathbf{X}$, v_k^j is the j-th element of the vector $\mathbf{v}_k$, $\mathbf{X}^a$ are the columns of $\mathbf{X}$ associated with currently selected features (indexed by a), and $\mathbf{e}_j^a$ are the a-th elements of the j-th row of the identity matrix.

Then we have $L(\boldsymbol{\beta}) = L(\boldsymbol{\beta}_{\setminus j}) + \frac{1}{2}(\ln \beta_j - \ln(\beta_j + u_j) + \frac{r_j^2}{\beta_j + u_j})$. where $L(\boldsymbol{\beta}_{\setminus j})$ does not depend on β_j. Therefore, we can directly optimize over β_j without updating $\boldsymbol{\beta}_{\setminus j}$.

Setting the gradient of $L(\boldsymbol{\beta})$ over β_j, we easily obtain the following optimality condition: if $u_j^2 \geq r_j$,

$$\beta_j = \frac{r_j^2}{u_j^2 - r_j}; \tag{13}$$

if $u_j^2 < r_j$, $\beta_j = \infty$. In the latter case we remove the j-th feature if it is currently in the model.



(a) Lasso (b) Elastic net (c) EigenNet (d) True

Figure 3: Visualization of the lasso, the elastic net, the EigenNet and the true classifier weights. We used 80 training samples with 40 features. The test error rates of the lasso, the elastic net, and the EigenNet on 2000 test samples are 0.297, 0.245, and 0.137, respectively.

The above active-set updates are very efficient, because during each iteration we only deal with a reduced model defined on the currently selected features. This approach significantly reduces the computational cost of EP from $O(np^2)$ to $O(nl^2)$ where l is the biggest model size during the active-set iterations. The empirical Bayesian estimation algorithm of EigenNet is summarized in Algorithm 1.

4 Related work

The EigenNet is related to the classical eigenface approaches (Turk and Pentland, 1991; Sirovich and Kirby, 1987). The eigenface approach learns a model in the subspace spanned by the major eigenvectors of the data covariance matrix. The EigenNet also uses the eigensubspace to guide the model estimation. However, unlike the eigenface approach, the EigenNet adaptively selects eigenvectors and learns a sparse classifier.

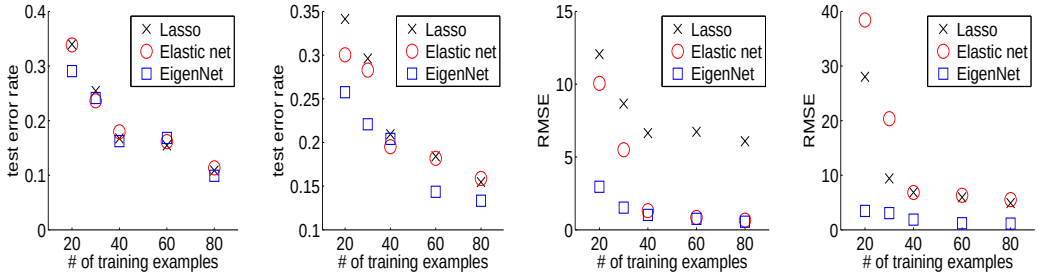
There are Bayesian versions of the lasso and the elastic net. Bayesian lasso (Park et al., 2008) puts a hyper-prior on the regularization coefficient and use a Gibbs sampler to jointly sample both regression weights and the regularization coefficient. Using a similar treatment to Bayesian lasso, Bayesian elastic net (Li and Lin, 2010) samples the two regularization coefficients simultaneously, potentially avoiding the "double shrinkage" problem described in the original elastic net paper (Zou and Hastie, 2005). As the EigenNet, these methods are grounded in a Bayesian framework, sharing the benefits of obtaining posterior distributions for handling estimation uncertainty. However, Bayesian lasso and Bayesian elastic net are presented to handle regression problems (though certainly they can be generalized for classification problems) and do not use the eigen-information embedded in data. The EigenNet, by contrast, selects the eigen-subspace and uses it to guide classification.

Group lasso (Jacob et al., 2009) enforces sparsity on the groups of predictors—an entire group of correlated predictors may be retained or pruned off. However, applying the idea of group lasso to the EigenNet faces several difficulties: First, this approach won't give (approximately) sparse classifiers unless we truncate eigenvectors. If we use truncation, we need to decide what threshold we should use to truncate each eigenvector—again it's a difficult task. Second, it will be hard to tune all regularization coefficients associated with all major eigenvectors–cross validation would not suffice. By contrast, our classifier is sparse because of the ARD effect. More importantly, the latent variables s_j in our model are automatically estimated from data, deciding how important each eigenvector is for the classification task in a principled Bayesian framework.

5 Experimental results

We evaluated the new sparse Bayesian model, the EigenNet, on both synthetic and real data and compared it with three representative variable selection methods, the lasso, the elastic net, and an ARD approach (Qi et al., 2004). For the lasso and the elastic net, we used the Glmnet software package that uses cyclical coordinate descent in a pathwise fashion[1]. Like the EigenNet, the ARD approach also uses EP to approximate the model marginal likelihood. For the lasso and the elastic net, we used cross-validation to tune the hyperparameters; for the EigenNet, we estimated λ_v from data and tuned λ_s by cross-validation.

[1] http://www-stat.stanford.edu/ tibs/glmnet-matlab/



(a) independent features (b) correlated features (c) independent features (d) correlated features

Figure 4: Predictive performance on synthetic datasets. (a) and (b): classification; (c) and (d): regression The results were averaged over 10 runs. For the data with independent features, the EigenNet outperforms the alternative methods when the number of training samples is small; for data with correlated features, the EigenNet outperforms the alternative methods consistently.

5.1 Visualization of estimated classifiers

First, we tested these methods on synthetic data that contain correlated features. We sampled 40 dimensional data points, each of which contains two groups of correlated variables. The correlation coefficient between variables in each group is 0.81 and there are 4 variables in each group. We set the values of the classifier weights in one group as 5 and in the other group as -5. We also generated the bias term randomly from a standard Gaussian distribution. We set the number of training points to 80. Figure 3 shows the estimated classifiers and the true classifier we used to produce the data labels. Unlike the lasso and the elastic net, the EigenNet clearly identifies two groups of correlated variables, very close to the ground truth. As a result, on 2000 test points, the EigenNet achieves the lowest prediction error rate, 0.137, while the test error rates of the lasso and the elastic net are 0.297 and 0.245, respectively.

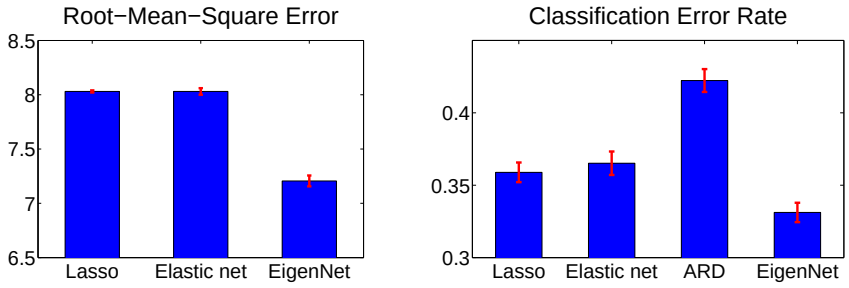
5.2 Experiments on synthetic data

Now we systematically compared these methods for classification and regression on synthetic datasets containing correlated features and containing independent features (Although this presentation so far has been focused on classification, we can easily implement the EigenNet for regression; since we can compute the marginal likelihood exactly, the EP approximation is not needed for regression.) To generate data with correlated variables we used a similar procedure as in the visualization example: we sampled 40 dimensional data points, each of which contains two groups of correlated variables. The correlation coefficient between variables in each group is 0.81 and there are 4 variables in each group. However, unlike for the previous example where the classifier weights are the same for the correlated variables, now we set the weights within the same group to have the same sign, but with different random values. We varied the number of training points, ranging from 10 to 80, and tested all these methods. For the datasets with independent features, we followed the same procedure except that the features are independently sampled. We ran the experiments 10 times. Figure 4 shows the results averaged over 10 runs. We did not report the standard errors since they are very small.

For the datasets with independent features, the EigenNet outperforms the alternative methods when the number of training examples is small (probably because in this case the eigenspace has a smaller dimension than than that of the classifier, effectively controlling the model flexibility); with more training examples, it is not unsurprising to see all these methods perform quite similarly. For the data with correlated features, although the results of the elastic net appear to overlaps with those of the lasso in the figure, the elastic net often outperforms the lasso with a small margin; also, the EigenNet consistently outperforms the lasso and the elastic net signficantly. The improved predictive performance of the EigenNet reflects the benefit of using the valuable correlation information to help the model estimation.

5.3 Application to imaging genetics

Imaging genetics is an emerging research area where imaging markers and genetic variations (e.g., SNPs) are used to study neurodegenerative diseases, in particular, Alzheimer's disease (AD). We



(a) *Regression of ADAS-Cog score* (b) *Classification of healthy & AD subjects*

Figure 5: Imaging genetics applications: (a) prediction of the ADAS-Cog score based on 14 imaging features and (b) AD classification based on 2000 SNPs. The error bars represent the standard errors.

applied the EigenNet to two critical problems in imaging genetics and compared its performance with that of alternative sparse learning methods.

First, we considered a regression problem where the predictors are imaging features, which were generated by Holland et al. (2009) for ADNI and include volume measured in 14 brain regions of interest (ROI)—including the whole brain, ventricles, hippocampus, *etc*. We used these imaging features to predict the ADAS-Cog score, which is widely used to assess cognitive function of AD patients. It is hypothesized that the brain ROI volumes are associated with the ADAS-Cog score. But this association has not been rigorously studied by statistical learning methods. After removing missing entries, we obtained the data of 726 subjects, including healthy people, people with mild cognitive impairment (MCI), and AD patients. Then we applied the lasso, the elastic net, and the EigenNet to this prediction task. We randomly selected 508 training samples and 218 test samples for 50 times. The results are shown in Figure 5.(a).

Second, we used SNP data to classify a subject into the healthy group or AD patients. We chose the top 2000 SNPs that are associated with AD based on a simple statistical test. There are 374 subjects in total (roughly the same size for each class). We compared the EigenNet with the lasso and the elastic net as well as the the ARD approach—since it corresponds to EigenNet's conditional component. We randomly split the dataset into 262 training and 112 test samples 10 times. The results are summarized in Figure 5.(b). As shown in the Figure, for both the regression and classification problems, the EigenNet outperforms the alternative methods significantly.

6 Conclusions

In this paper, we have presented a novel sparse Bayesian hybrid model to select correlated variables for regression and classification. It integrates the sparse conditional ARD model with a latent variable model for eigenvectors.

For this hybrid model, we could explore other latent variable models, such as sparse projection methods (Guan and Dy, 2009; Archambeau and Bach, 2009); these models can better deal with noise in the unlabeled data and improve the selection of interdependent features (i.e., predictors). Furthermore, if we have certain prior knowledge about the interdependence between features, such as linkage disequilibrium between SNPs, we could easily incorporate them into our model. Thus, our model provides an elegant framework for integrating complex data generation processes and domain knowledge in sparse learning.

7 Acknowledgments

The authors thank the anonymous reviewers and T. S. Jaakkola for constructive suggestions. This work was supported by NSF IIS-0916443, NSF CAREER award IIS-1054903, and the Center for Science of Information (CSoI), an NSF Science and Technology Center, under grant agreement CCF-0939370.

References

Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B*, 58:267–288, 1994.

Hui Zou and Trevor Hastie. Regularization and variable selection via the Elastic Net. *Journal of the Royal Statistical Society B*, 67:301–320, 2005.

Julia A. Lasserre, Christopher M. Bishop, and Thomas P. Minka. Principled hybrids of generative and discriminative models. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pages 87–94, 2006.

David J.C. MacKay. Bayesian interpolation. *Neural Computation*, 4:415–447, 1991.

Ildiko E. Frank and Jerome H. Friedman. A statistical view of some chemometrics regression tools. *Technometrics*, 35(2):109–135, 1993.

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

Thomas P. Minka. Expectation propagation for approximate Bayesian inference. In *Proceedings of the 17th Conference in Uncertainty in Artificial Intelligence*, pages 362–369, 2001.

Michael E. Tipping and Anita C. Faul. Fast marginal likelihood maximisation for sparse Bayesian models. In *Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics*, 2003.

Matthew Turk and Alex Pentland. Eigenfaces for recognition. *J. Cognitive Neuroscience*, 3:71–86, 1991.

L. Sirovich and M. Kirby. Low-dimensional procedure for the characterization of human faces. *J. Opt. Soc. Am. A*, 4(3):519–524, 1987.

Park, Trevor, Casella, and George. The Bayesian Lasso. *Journal of the American Statistical Association*, 103(482):681–686, 2008.

Qing Li and Nan Lin. The Bayesian Elastic Net. *Bayesian Analysis*, 5(1):151–170, 2010.

Laurent Jacob, Guillaume Obozinski, and Jean-Philippe Vert. Group lasso with overlap and graph lasso. In *Proceedings of the 26th Annual International Conference on Machine Learning*, 2009.

Yuan Qi, Thomas P. Minka, Rosalind W. Picard, and Zoubin Ghahraman. Predictive automatic relevance determination by expectation propagation. In *Proceedings of Twenty-first International Conference on Machine Learning*, pages 671–678, 2004.

Dominic Holland, James B Brewer, Donald J Hagler, Christine Fenema-Notestine, and Anders M Dale. Subregional neuroanatomical change as a biomarker for alzheimer's disease. *Proceedings of the National Academy of Sciences*, 106(49):20954–20959, 2009.

Yue Guan and Jennifer Dy. Sparse probabilistic principal component analysis. *JMLR W&CP: AISTATS*, 5, 2009.

Cédric Archambeau and Francis Bach. Sparse probabilistic projections. In *Advances in Neural Information Processing Systems 21*. 2009.