ISSN 1746-7659, England, UK
Journal of Information and Computing Science
Vol. 6, No. 4, 2011, *pp*. 295-302

A Novel Sparse Learning Method: Compressible Bayesian Elastic Net Model

Ke-Yang Cheng[1, 2], Qi-rong Mao[2],Xiao-yang Tan[1],Yong-zhao Zhan[2]

[1] School of Information Science & Technology, Nanjing University of aeronautics & astronautics, Nanjing, Jiangsu, China,210016

[2] School of Computer Science & Telecommunications Engineering, Jiangsu University, Zhenjiang, Jiangsu, China,212013

(*Received February 10, 2011, accepted May 22, 2011*)

Abstract. In this paper, we study the combination of compression and Bayesian elastic net. By including a compression operation into the ℓ1 and ℓ2 regularization, the assumption on model sparsity is relaxed to compressibility: model coefficients are compressed before being penalized, and sparsity is achieved in a compressed domain rather than the original space. We focus on the design of compression operations, by which we can encode various compressibility assumptions and inductive biases. We show that use of a compression operation provides an opportunity to leverage auxiliary information from various sources. The compressible Bayesian elastic net has another two major advantages. Firstly, as a Bayesian method, the distributional results on the estimates are straightforward, making the statistical inference easier. Secondly, it chooses the two penalty parameters simultaneously, avoiding the “double shrinkage problem” in the elastic net method. We conduct extensive experiments on braincomputer interfacing, handwritten character recognition and text classification. Empirical results show clear improvements in prediction performance by including compression in Bayesian elastic net. We also analyze the learned model coefficients under appropriate compressibility assumptions, which further demonstrate the advantages of learning compressible models instead of sparse models.

Keywords: Sparse Learning, compression operation, Bayesian elastic net

1. Introduction

Regularization was initially proposed to solve ill-posed problems (Tikhonov & Arsenin, 1977)[1]. In statistical learning, regularization is widely used to control model complexity and prevent overfitting (Hastie et al., 2001)[2]. Regularization seeks a trade-off between fitting the observations and reducing the model complexity, which is justified by the minimum description length (MDL) principle in information theory (Rissanen, 1978)[3] and the bias-variance dilemma in statistics (Sullivan, 1986)[4]. Since the introduction of lasso (Tibshirani, 1996)[5], ℓ1-regularization has become very popular for learning in high-dimensional spaces. A fundamental assumption of ℓ1-regularization is the sparsity of model parameters, i.e., a large fraction of coefficients are zeros. While demonstrating promising performance for many problems, the lasso estimator does have some shortcomings.

Zou and Hastie (2005) [6] emphasized three inherent drawbacks of the lasso estimator. Firstly, due to the nature of the convex optimization problem, the lasso method cannot select more predictors than the sample size. But in practice there are often studies that involve much more predictors than the sample size, e.g. microarray data analysis (Guyon et al. 2002)[7]. Secondly, when there is some group structure among the predictors, the lasso estimator usually selects only one predictor from a group while ignoring others. Thirdly, when the predictors are highly correlated, the lasso estimator performs unsatisfactorily. Zou and Hastie (2005) proposed the elastic net (en) estimator to achieve improved performance in these cases. The en estimator can also be viewed as a penalized least squares method where the penalty term is a convex combination of the lasso penalty and the ridge penalty.

Another shortcoming of Lasso is that the sparsity assumption on model coefficients might be too restrictive and not necessarily appropriate in many application domains. Indeed, many signals in the real

[1] Corresponding author. *E-mail address*: kycheng@ujs.edu.cn

DOI https://doi.org/2024-JICS-22671 | Generated on 2025-04-09 07:03:09

world (e.g., images, audio, videos, time series) are found to be compressible (i.e., sparse in certain compressed domain) but not directly sparse in the observed space. Naturally, the assumption of sparsity can be relaxed to compressibility. Inspired by the recent development of compressive sampling (or compressed sensing) (Candes, 2006[8]; Donoho, 2006[9]), we study learning compressible models: a compression on model coefficients can be included in the ℓ1 and ℓ2 penalty, and model is assumed to be sparse after compression.

The rest of this paper is organized as follows. In section 2 we will briefly introduce naïve elastic net. In Section 3 we discuss the definition, computation issues and potential benefits of learning compressible Bayesian elastic net model. In this Section, we propose some classes of model compressibility assumptions and model hierarchy distributions. In Sections 4, we empirically study some real-world problems using compressibility as a more appropriate inductive bias than sparsity. Experimental results also demonstrate the advantages of compressible Bayesian elastic net (cben) than compressible Bayesian lasso (cbl), elastic net (en) and lasso. Section 5 concludes and mentions some discussions.

2. Naive elastic net

Suppose that the data set has n observations with p predictors. Let $y=(y_1,...,y_n)^T$ be the response and $X=(x_1,...,x_p)$ be the model matrix, where $x_j=(x_{1j},...,x_{nj})^T, j=1,...,p$, are the predictors. After a location and scale transformation, we can assume that the response is centred and the predictors are standardized.

$$\sum_{i=1}^{n} y_i = 0 \ \sum_{i=1}^{n} x_{ij} = 0 \text{ and } \sum_{i=1}^{n} x^2{}_{ij} = 1 \text{.for j=1,2,…,p.}$$

For any fixed non-negative λ1 and λ2, we define the naive elastic net criterion

$$L(\lambda_1,\lambda_2,\beta)=|y-X\beta|^2+\lambda_2|\beta|_2^2+\lambda_1|\beta|_1 \quad (1)$$

where

$$|\beta|_2^2=\sum_{j=1}^{p}\beta_j^2,$$

$$|\beta|_1=\sum_{j=1}^{p}|\beta_j|.$$

The naive elastic net estimator $\hat{\beta}$ is the minimizer of equation (1):

$$\hat{\beta}=\arg\min_{\beta}\{L(\lambda_1,\lambda_1,\beta)\}.$$

This procedure can be viewed as a penalized least squares method. Let α=λ₁+λ₂; then solving $\hat{\beta}$ in equation (1) is equivalent to the optimization problem

$$\hat{\beta}=\arg\min_{\beta}|y-X\beta|^2, \qquad \text{subject to } (1-\alpha)|\beta|_1+\alpha|\beta|_2^2\le t \text{ for some t.}$$

We call the function $(1-\alpha)|\beta|_1+\alpha|\beta|_2^2$ the elastic net penalty, which is a convex combination of the lasso and ridge penalty. When α=1, the naive elastic net becomes simple ridge regression. In this paper, we consider only α<1. For all α∈[0, 1), the elastic net penalty function is singular (without first derivative) at 0 and it is strictly convex for all α>0, thus having the characteristics of both the lasso and ridge regression. Note that the lasso penalty (α=0) is convex but not strictly convex. These arguments can be seen clearly from Fig. 1.

DOI https://doi.org/2024-JICS-22671 | Generated on 2025-04-09 07:03:09

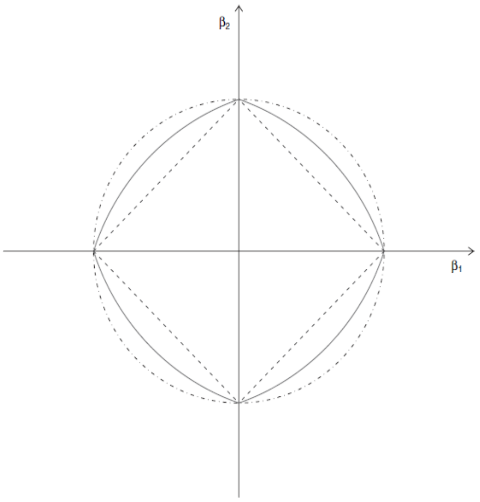


Fig 1. The α-ball with α> 0 (solid line, contour of the elastic net penalty), the square (ℓ1-ball), which is the α-ball with α= 0 (dashed line, contour of the lasso penalty), and the disc (ℓ2-ball), which is the α-ball with α→∞(dotted line, shape of the ridge penalty). we see that singularities at the vertices and the edges are strictly convex; the strength of convexity varies with α.

3. Compressible Bayesian Elastic Net Model

3.1. Learning Compressible Model

Assuming the model to be sparse and shrinking model coefficients to exactly zero might not be the most appropriate inductive bias in many problems. For example, real-world signals (such as audio, images, videos and time series) are usually compressible but not directly sparse in the observation domain. Interestingly, compressive sampling (Candes, 2006)[8] or compressed sensing (Donoho, 2006) [9]was recently developed in signal acquisition. The target signal is assumed to be compressible and sparse after being compressed. We consider the problem of learning compressible models as follows:

$$\min L(y,1\alpha + X\beta) + \lambda_1 \| W(\beta) \|_1 + \lambda_2 \| W(\beta) \|_2^2 \tag{2}$$

The loss function L depends on the prediction model, e.g., sum of squares loss for linear regression, log-likelihood loss for logistic regression, hinge loss for SVMs, and so forth. The compression operation W() encodes our assumption on compressibility: model coefficients are compressed by W() before being penalized, and thus tend to follow the compression pattern (i.e., sparse in the compressed domain) rather than simply shrink to zero.

For simplicity, we restrict our attention to linear compression. Given that the compression operation is a linear and invertible transform, learning compressible models is represented by eq. (2).

The p×p matrix W denotes the linear and invertible compression transform, where p is the dimensionality of model coefficients. The optimization of eq. (2) can be achieved by applying the inverse compression transform (i.e., the decompression operation) to the feature space and solving the elastic net model in the transformed space. First, transform the training examples by

$$\widetilde{X} = XW^{-1} \tag{3}$$

Second, solve the following standard elastic net model:

$$\min L(y,1\alpha + \widetilde{X}\widetilde{\beta}) + \lambda_1 \| W\widetilde{\beta} \|_1 + \lambda_2 \| W\widetilde{\beta} \|_2^2 \tag{4}$$

Finally, the solution for eq. (2) is obtained by:

$$\beta = W^{-1}\widetilde{\beta} \tag{5}$$

$$\alpha = \alpha \tag{6}$$

This equivalence is derived from $X\beta = XW^{-1}\widetilde{\beta} = \widetilde{X}\widetilde{\beta}$,$\| W\beta \|_1 = \| WW^{-1}\widetilde{\beta} \|_1 = \| \widetilde{\beta} \|_1$ and

$$\| W\beta \|_2^2 = \| WW^{-1}\widetilde{\beta} \|_2^2 = \| \widetilde{\beta} \|_2^2 .$$

Why do we want to learn compressible models, which are not necessarily sparse in the original space? Compressible models are useful in several aspects. The first is model fitting and prediction accuracy. The

DOI https://doi.org/2024-JICS-22671 | Generated on 2025-04-09 07:03:09

inductive bias of model compressibility might be more appropriate than model sparsity, especially if an informative compression operation is specified based on additional information from domain knowledge, unlabeled data or related problems. The second reason, as claimed for standard sparse models, is interpretability. Model coefficients that are sparse in a compressed domain can still be insightful in the original space in many problems, as later shown by our empirical studies on brain-computer interface (in Section 5) and handwritten digit recognition (in Section 4). The third reason is that, when the compression operation is known in advance, compressible models are very efficient for storage and transmission in the compressed domain. This advantage has been widely recognized in compressive sensing (Candes, 2006[8]; Donoho, 2006[9]) for general signals and thus also valid when signals are model coefficients.

3.2. Compression Operation

In this section, we discuss how to get the compression operation matrix W. The local smoothness assumption on models is related to compression operation. Smoothness characterizes the properties of derivatives of a function. For example, a constant (or piecewise constant) function has zero first-order derivatives at all (or most) locations, and a quadratic function has zero third-order derivatives at all locations. Here we will show that use of a compression transform is very flexible and can represent various smoothness assumptions on model coefficients.

Suppose we have a natural order over model coefficients $\{\beta_j\}_{j=1}^{p}$, e.g., in temporal domains where each dimension corresponds to a time point, or spectral domains where each dimension corresponds to a frequency. *Order*-1 *smoothness* assumes the coefficients "do not change very often" along the natural order. Such an assumption characterizes the first-order derivatives. It has been studied in fused lasso (Tibshirani et al., 2005[5]) where absolute values of the difference of successive coefficients, i.e., $\sum_{j=2}^{p} |\beta_j - \beta_{j-1}|$, are penalized. This idea was also explored in total variation minimization for noise removal and image enhancement (Rudin et al., 1992[10]). As a motivating example, we show that the fused lasso penalty can be approximated by a linear and invertible compression in the ℓ1 penalty.

The p × p matrix W for model compression based on order-1 smoothness can be defined as:

$$W = S_p^1 = \begin{bmatrix} 1/p & 1/p & \dots & \dots & \dots & 1/p \\ 1 & -1 & 0 & \dots & \dots & 0 \\ 0 & 1 & -1 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & & & 1 & -1 \end{bmatrix} \tag{7}$$

Model coefficients in the compressed domain $W\beta = [\overline{\beta}, \beta_1 - \beta_2, ..., \beta_{p-1} - \beta_p]$ tend to be sparse due to ℓ1 regularization, which achieves the order-1 smoothness. The averaging operation in the first row of W makes the transform invertible. Note that if the first row of W is multiplied by a small constant (e.g., 0.001), $\|W\beta\|_1$ approximates the fused lasso penalty. In our study, we will use the compression in eq. (7) without scaling the averaging operation. Also, we keep the compression operation invertible to make the optimization efficient, as discussed in eq. (3) - eq. (5).

Smoothness of higher orders is also common. For example, a piecewise linear function has piecewise constant first-order derivatives, indicating zero second-order derivatives at most locations. This is defined as *order*-2 *smoothness*. In this case, the p × p compression transform W can be:

$$W = S_p^2 = \begin{bmatrix} 1 & 0^T \\ 0 & S_{p-1}^1 \end{bmatrix} S_p^1 \tag{8}$$

where 0 is a (p−1)×1 column vector. By this definition, model coefficients in the compressed domain are $W\beta = [\overline{\beta}, \overline{\Delta\beta}, \Delta\beta_{1,2} - \Delta\beta_{2,3}, ..., \Delta\beta_{p-2,p-1} - \Delta\beta_{p-1,p}]$, where $\Delta\beta_{i,i+1} = \beta_i - \beta_{i+1}$. In this sense, sparsity of the compressed model coefficients corresponds to order-2 smoothness assumption in the original space. Also, S_p^2 is invertible since both S_{p-1}^1 and S_p^1 are invertible. Finally, model compression for *higher-order smoothness* can be defined recursively.

Sometimes features under consideration do not follow a universal order, but can be divided into groups, where each group of features has an order or at least some groups of features are ordered. The compression operation can be defined as a block matrix to handle the use of different groups of features. For example, suppose features can be divided into three groups. We assume p_1 model coefficients on the first group of features satisfy order-1 smoothness, p_2 coefficients on the second group of features satisfy order-2 smoothness, and we have no knowledge about the third group of p_3 features. In this case, model compression is defined as:

$$W = \begin{bmatrix} S_{p_1}^1 & 0 & 0 \\ 0 & S_{p_2}^2 & 0 \\ 0 & 0 & I_{p_3} \end{bmatrix} \tag{9}$$

3.3. Model hierarchy and prior distributions of Compressible Bayesian Elastic Net Model

Consider the linear model $E(y \mid X, \beta) = X\beta$; where we assume that the response variables follow the normal distribution conditionally, i.e. $y \mid X, \beta \sim N(X\beta, \sigma^2 I_n)$ We assume that all analysis hereafter is conditional on X. Zou and Hastie (2005)[6] pointed out that, under these assumptions, solving the cben problem is equivalent to finding the marginal posterior mode of β|y when the prior distribution of β is given by:

$$\pi(\beta) \propto \exp\{-\lambda_1 \|\beta\|_1 - \lambda_2 \|\beta\|_2^2\} \tag{10}$$

a compromise between Gaussian and Laplacian priors. Specifically, the conditional posterior distribution has the probability density function (pdf)

$$\begin{aligned} f(\beta \mid \sigma^2, y) &\propto \exp\{-\frac{1}{2\sigma^2}(y - X\beta)^T (y - X\beta) - \lambda_1 \|\beta\|_1 - \lambda_2 \|\beta\|_2^2\} \\ &= \exp\{-\frac{1}{2\sigma^2}(y - X\beta)^T (y - X\beta) + (2\sigma^2 \lambda_1) \|\beta\|_1 + (2\sigma^2 \lambda_2) \|\beta\|_2^2\} \end{aligned} \tag{11}$$

However, under (10), neither the conditional posterior mode of $\beta \mid \sigma^2, y$ nor the marginal posterior mode of $\beta \mid y$ would be equivalent to the cben estimator $\hat{\beta}_{EN}$ unless the analysis is conditional on σ^2 or σ^2 is given a point-mass prior. Instead we propose a prior for β, which is conditional on σ^2, as

$$\pi(\beta \mid \sigma^2) \propto \exp\{-\frac{1}{2\sigma^2}(\lambda_1 \|\beta\|_1 + \lambda_2 \|\beta\|_2^2\}\} \tag{12}$$

A noninformative prior is then assigned for σ^2 ,i.e. $\pi(\sigma^2) \propto 1/\sigma^2$ In this setup, the marginal posterior distribution for $\beta \mid y$ has the pdf

$$f(\beta \mid y) = \int_0^\infty \frac{C(\lambda_1, \lambda_1, \sigma^2)}{(2\pi\sigma^2)^{n/2}} \exp\{-\frac{\|y - X\beta\|_2^2 + \lambda_1 \|\beta\|_1 + \lambda_2 \|\beta\|_2^2}{2\sigma^2}\} \pi(\sigma^2) d\sigma^2 \tag{13}$$

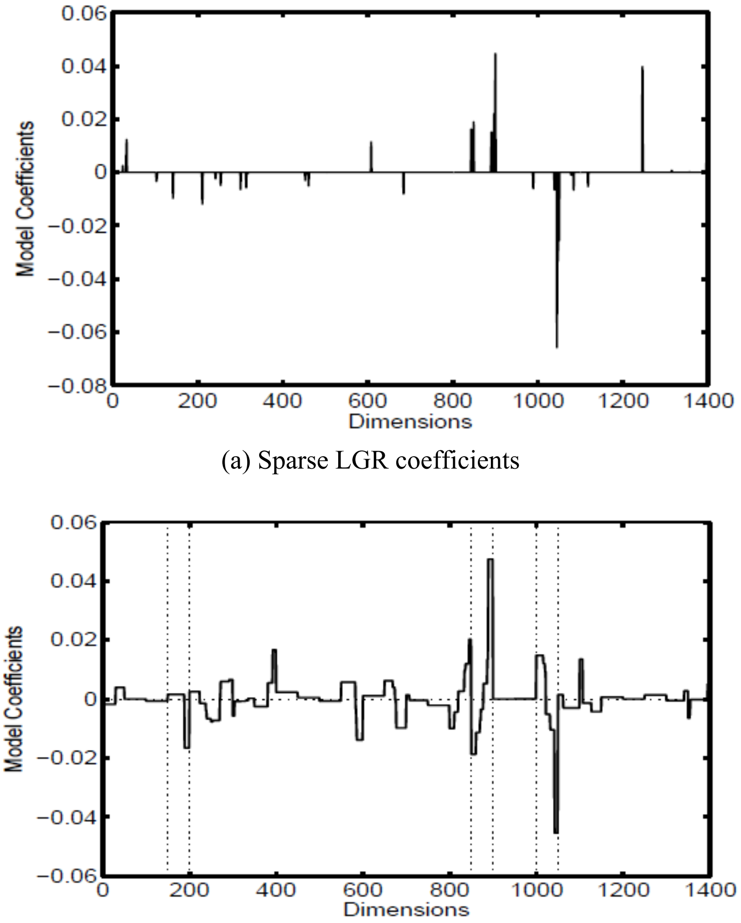
where $C(\lambda_1, \lambda_2, \ \sigma^2)$ is the normalizing constant later mentioned in Lemma 1. From (13), we can see that the cben estimator $\hat{\beta}$ maximizes the integrand for each σ^2, and thus equals the marginal posterior mode of $\beta \mid y$. This conditional prior specification is also used by Park and Casella (2008)[11] for the sake of accelerating the convergence of the Gibbs sampler in that paper. Based on the discussion above, we have the following hierarchial model.

$$\begin{aligned} & y \mid \beta, \sigma^2 \sim N(X\beta, \sigma^2 I_n), \\ & \beta \mid \sigma^2 \sim \exp\{-\frac{1}{2\sigma^2}(\lambda_1 \|\beta\|_1 + \lambda_2 \|\beta\|_2^2)\} \\ & \sigma^2 \sim \frac{1}{\sigma^2} \end{aligned} \tag{14}$$

DOI https://doi.org/2024-JICS-22671 | Generated on 2025-04-09 07:03:09

4. Experimental Results and Analysis

We use data set IV, self-paced tapping, of BCI Competition 2003 (Blankertz et al., 2004[12]), which is a binary classification task. The task contains a training set of 316000 examples and a testing set of 100000 examples. Each example has 1400 features, corresponding to 28 channels and 50 measurements from each channel. The number of features is much larger than the number of training examples, indicating the importance of regularization. Each example is measured when a healthy subject, sitting in a chair with fingers in the standard typing position, tries to press the keys using either the left hand or right hand. The objective is to classify an Electroencephalography (EEG) signal to either a left-hand movement or right-hand movement.



(a) Sparse LGR coefficients

(b) Compressible LGR coefficients

Fig. 2: Model coefficients of sparse and compressible (i.e., piecewise smooth) logistic regression on brain-computer interfacing (EEG signal classification)

We plot the model coefficients learned by a sparse logistic regression and a compressible logistic regression in Figure 1. From the plot we have several interesting observations. 1) Sparse logistic regression learns sparse coefficients, and compressible logistic regression (LGR) leads to (piecewise) smooth coefficients. These two different patterns represent the inductive biases we incorporate into the learning process (via different regularization penalties). 2) Although in the compressible logistic regression we mainly penalize the difference of successive coefficients, most learned coefficients are actually close to zero. The proposed regularization (piecewise local smoothness) effectively controls the model complexity not only in terms of smoothness but also in terms of the norm of coefficients. 3) In the compressible logistic regression, there still exist a few large coefficient jumps over successive dimensions (within the same channel): we plot in Fig. 2b the boundaries (vertical dashed lines) of three selected channels that contain large coefficient jumps. These jumps correspond to large coefficients in the compressed domain (recall that the compressed domain defined by our compression operation is composed of the difference of successive coefficients within the same channel in the original space). The existence of a few large coefficients in the compressed domain is consistent with the notation of compressibility: most information of the original signal is concentrated on a few components after being compressed.

We also carry out Monte Carlo simulations and use the median of the prediction mean-squared errors (MMSE) to compare the performance of cben, cbl, en and lasso in prediction accuracy and variable selection. For cben, instead of using the penalty parameters (λ_1; λ_2), it is equivalent to use (s,λ_2), where $s = \|\beta\|_1 / \|\beta_{OLS}\|_1$ with β_{OLS} being the ordinary least squares (ols) estimate (Zou and Hastie 2005[6]). Similarly, s instead of λ_1 is used in lasso.

The data are simulated from the following linear model,

$$y = X\beta + \varepsilon, \varepsilon \sim N(0, \sigma^2 I) \tag{15}$$

Each simulated sample was partitioned into a training set, a validation set and a testing set. The validation set is used to select s for lasso and (S, λ_2) for en. After the penalty parameter is selected, we combine the training set and the validation set together to estimate β. For Bayesian methods, we directly combine the training and validation sets together for estimation.

For the first two simulation studies, we simulate 50 data sets, each of which has 20 observations for the training set, 20 for the validation set, and 200 for the testing set. In Simulation 1, we set β = $(3,1.5,0,0,2,0,0,0)^T$ and σ^2=9. The design matrix X is generated from the multivariate normal distribution with mean 0, variance 1 and pairwise correlations between x_i and x_j equal to $0.5^{|i-j|}$ for all i and j. In Simulation 2, we set β = $(0.85,0.85,0.85,0.85,0.85,0.85,0.85,0.85)^T$, and leave other setups the same as in Simulation 1. In Simulation 3, we first generate Z_1, Z_2 and Z_3 independently from $N(0,1)$. Then let $x_i = Z_1 + \varepsilon_i$, $i = 1,\ldots, 5$, $x_i = Z_2 + \varepsilon_i$, $i = 6,\ldots,10$, $x_i = Z_3 + \varepsilon_i$, $i = 11,\ldots,15$ and $x_i \sim N(0, 1)$, $i = 16,\ldots,30$, where $\varepsilon_i \sim N(0,0.01)$, $i = 1,\ldots,15$. We perform 50 simulations, in each of which we have a training set of size 100, a validation set of size 100 and a testing set of size 400. The parameters are set as σ^2 = 225 and

$$\beta = (\underbrace{3,\ldots,3}_{5},\underbrace{3,\ldots,3}_{5},\underbrace{3,\ldots,3}_{5},\underbrace{0,\ldots,0}_{15})^T .$$

In Simulation 4, we set the sizes of the training set and the validation set both to 200, while leaving other setups the same as in Simulation 3. In Simulation 5, we set the sizes of the training set and the validation set both to 20, and the true parameter value to

$$\beta = (\underbrace{3,\ldots,3}_{10},\underbrace{0,\ldots,0}_{10},\underbrace{3,\ldots,3}_{10}),$$

while leaving other setups the same as in Simulation 3.

Table 1: Comparison of the four methods (ben, bl, en, and lasso) on prediction accuracy

Method	Simulation1	Simulation2	Simulation3	Simulation4	Simulation5
	MMSE(SE)	MMSE(SE)	MMSE(SE)	MMSE(SE)	MMSE(SE)
cben	11.99(0.28)	11.39(0.29)	232.3(3.83)	215.1(1.95)	335.3(4.17)
en	11.29(0.53)	11.10(0.29)	281.4(6.80)	243.0(4.24)	376.9(9.81)
cbl	10.45(0.24)	10.55(0.21)	227.1(4.20)	211.4(2.36)	352.6(13.1)
lasso	10.99(0.20)	11.41(0.33)	280.1(8.00)	243.0(4.34)	384.9(9.61)

Table 1 shows that cbl has a better prediction accuracy than other methods in most simulation studies. Secondly, when the true model structure is relatively simple (Simulations 1 and 2), the four methods perform comparably in prediction accuracy. But when the true model has a complex structure (Simulations 3, 4 and 5), cben and cbl outperform en and lasso significantly. By comparing cben and cbl in Simulation 3 and en and lasso in Simulation 4 (the bolded numbers in Table 1), we can see that even with only half as many data, the Bayesian methods perform better than the non-Bayesian methods in prediction accuracy. One possible reason is that complicated models would result in highly variational estimators, and the Bayesian methods use prior information to integrate across uncertainty to reduce the variance, which leads to a smaller mean squared error. In this sense, the Bayesian methods furnish better results with less data when the true model is complicated. Furthermore, with the sample size doubled from Simulation 3 to Simulation 4, the MMSE of the Bayesian methods decreases about 15 while that of the non-Bayesian methods decreases about 40.

5. Conclusion and discussion

By including a compression operation into ℓ1 and ℓ2 regularized learning, model coefficients are compressed before being penalized and sparsity is achieved in a compressed domain. This relaxes the assumption on model sparsity to compressibility, and provides an opportunity encode more appropriate inductive biases. Empirical results show significant improvements in prediction performance by including compression in the ℓ1 and ℓ2 penalty. We analyze the learned model coefficients under different compressibility assumptions, which further demonstrate the advantages of learning compressible models instead of sparse models. We also propose a Bayesian formulation of the en problem and the posterior inference can be obtained efficiently using Gibbs sampling. Real data examples and simulation studies show that cben behaves comparably to en in prediction accuracy en does slightly better than cben for simple models, but cben performs significantly better than en for more complicated models. Simulation studies suggest that cben outperforms en in variable selection when coupled with the scaled neighborhood criterion with a proper probability threshold, and cben gives better prediction accuracy than cbl for small samples from less sparse models.

6. Acknowledgments

This research is supported by the national science foundation of China (NFSC) No.61003183, natural science foundation of Jiangsu province No.BK2009199 and innovative base (laboratory) open foundation of Nanjing University of aeronautics & astronautics.

7. References

[1] Tikhonov, A. N., & Arsenin, V. Y. *Solutions of ill-posed problems*. Winston and Sons. 1977.

[2] Hastie, T., Tibshirani, R., & Friedman, J. *The Elements of Statistical Learning: Data Mining, Inference and Prediction*. Springer. 2001.

[3] Rissanen, J. Modeling by shortest data description. *Automatica*. 1978, **14**: 465–471.

[4] Sullivan, F. O. A statistical perspective on ill-posed inverse problems. *Statist. Sci.* 1986, pp. 502–518.

[5] Tibshirani, R. Regression Shrinkage and Selection via the Lasso. *Journal of the Royal Statistical Soceity, Series B*. 1996, **58**(1): 267–288.

[6] Zou, H., & Hastie, T. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society B*. 2005, **67**: 301–320.

[7] Guyon, I., Weston, J., Barnhill, S., and Vapnik, V. Gene Selection for Cancer Classification Using Support Vector Machines. *Mach. Learn.* 2002, **46**: 389-422.

[8] Candes, E. J. Compressive Sampling. *Proceedings of International Congress of Mathematicians*. 2006.

[9] Donoho, D. L. Compressed Sensing. *IEEE Trans. Information Theory*. 2006, **52**(4): 1289–1306.

[10] Rudin, L., Osher, S., & Fatemi, E. Nonlinear total variation based noise removal algorithms. *Physica D*. 1992, ***60***: 259–268.

[11] Park, T. and Casella, G. The Bayesian Lasso. *J. Amer. Statist. Assoc.* 2008, **103**: 681-686.

[12] Blankertz, B., et al. The BCI Competition 2003: Progress and Perspectives in Detection and Discrimination of EEG Single Trails. *IEEE Trans. Biomedical Engineering*. 2004, **51**(6): 1044–1051.

DOI https://doi.org/2024-JICS-22671 | Generated on 2025-04-09 07:03:09