A Simple Bayesian Algorithm for Feature Ranking in High Dimensional Regression Problems

Enes Makalic and Daniel F. Schmidt

Centre for MEGA Epidemiology, The University of Melbourne,
Carlton, VIC 3053, Australia
{emakalic,dschmidt}@unimelb.edu.au

Abstract. Variable selection or feature ranking is a problem of fundamental importance in modern scientific research where data sets comprising hundreds of thousands of potential predictor features and only a few hundred samples are not uncommon. This paper introduces a novel Bayesian algorithm for feature ranking (BFR) which does not require any user specified parameters. The BFR algorithm is very general and can be applied to both parametric regression and classification problems. An empirical comparison of BFR against random forests and marginal covariate screening demonstrates promising performance in both real and artificial experiments.

1 Introduction

Variable selection or feature ranking is a problem of fundamental importance in modern scientific research where data sets comprising hundreds of thousands of potential features and only a few hundred samples are not uncommon. In this setting, popular methods for importance ranking of features include the non-negative garotte [1], the least angle shrinkage and selection operator (LASSO) [2] and variants [3–5] as well as algorithms based on independence screening [6, 7]. The availability of computationally efficient learning algorithms for LASSO-type methods [8, 9] has made this approach particularly common in the literature. In addition, the LASSO and its variants fit all the covariates simultaneously, taking into account the correlation between the covariates, in contrast to marginal methods that examine each covariate in isolation.

An important issue with the application of LASSO-type methods for variable selection is how to specify the regularization or shrinkage parameter which determines the actual ranking of variables [10]. This is a highly challenging problem where a model selection method such as cross validation (CV) can lead to inconsistent results [11]. The problem may be circumvented by framing the LASSO in a Bayesian setting [12, 13] where the regularization parameter is automatically determined by posterior sampling. However, Bayesian LASSO-type algorithms cannot *fully* exclude any particular variable and thus do not provide an automatic importance ranking for the candidate features.

D. Wang and M. Reynolds (Eds.): AI 2011, LNAI 7106, pp. 223–230, 2011.

This paper presents a novel Bayesian algorithm, henceforth referred to as BFR, for variable selection in any parametric model where samples from the posterior distribution of the parameters are available. The new algorithm (see Section 2) computes an importance ranking of all observed features as well as credible intervals for these feature rankings. The credible intervals can then be used to remove features from further analysis that contribute little to explaining the data. The algorithm is very general, requires no user specified parameters and is applicable to both parametric regression and classification problems. The BFR algorithm is compared against random forests [14] and independence screening by generalized correlation [7] in Section 3. Empirical tests using artificially generated data as well as real data demonstrate excellent performance of the BFR algorithm.

2 Bayesian Feature Ranking (BFR) Algorithm

Given a data set comprising n samples

$$\mathcal{D} = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_n, y_n)\}, \tag{1}$$

where $\mathbf{x}_i \in \mathbb{R}^p$ and $y_i \in \mathbb{R}$, the task is to select which of the p covariates, if any, are relevant to explaining the target $\mathbf{y} = (y_1, \ldots, y_n)'$. The target variable is assumed to be either real (regression task) or m-ary (classification task, $m \geq 2$), and to belong to the generalized linear family of statistical models with coefficients $\boldsymbol{\theta} \in \Theta$. Arrange the covariates into an $(n \times p)$ matrix $\mathbf{X} = (\mathbf{x}'_1, \ldots, \mathbf{x}'_n)'$. Without any loss of generality, we assume that the covariates are standardised to have zero mean and unit length, that is,

$$\sum_{i=1}^{n} X_{ij} = 0, \quad \sum_{i=1}^{n} X_{ij}^2 = 1. \tag{2}$$

Furthermore, we assume that there exists $B > 0$ samples $\{\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots, \boldsymbol{\theta}_B\}$ from the posterior distribution $p(\boldsymbol{\theta}|\mathbf{X}, \mathbf{y})$ of the coefficients given the data.

The BFR algorithm proceeds by ranking the p covariates based on the absolute magnitude of the parameters in each posterior sample. That is, given a posterior sample $\boldsymbol{\theta}_i$, the parameters are ranked in descending order of $|\theta_{ij}|$ for $j = 1, 2, \ldots, p$. This process requires that the covariates are standardised as in (2) so that the absolute magnitude of some parameter θ_{ij} is an indication of the amount of variance explained by the corresponding column of the design matrix $(X_{kj}, k = 1, \ldots, n)$. The motivation for this comes from the fact that in a linear regression model, the amount of variance explained by covariate j with associated parameter θ_j is

$$\theta_j^2 \left(\sum_{k=1}^{n} X_{kj}^2 \right).$$

Due to the fact that we have standardised the covariates to have unit length, the amount of variance explained reduces simply to θ_j^2. This implies that ranking

covariates in decreasing order of absolute magnitude of their associated coefficients is equivalent to ranking them in descending order of variance explained. The ranking process is repeated in turn for each of the posterior samples, resulting in B possible rankings of the p covariates. The final ranking of the covariates is determined from the complete set of rankings based on the empirical 75th percentile of each of the B possible rankings. Furthermore, the set of rankings can also be used to compute Bayesian credible intervals for the inclusion of each covariate. The BFR procedure is formally described in Algorithm 1.

Algorithm 1. BFR algorithm for feature ranking

Input: standardised feature matrix $\mathbf{X} \in \mathbb{R}^{n \times p}$, standardised target vector $\mathbf{y} \in \mathbb{R}^n$
Output: feature ranking $\tilde{\mathbf{r}} = (\tilde{r}_1, \tilde{r}_2, \ldots, \tilde{r}_p)$ $(1 \leq \mathbf{r}_i \leq p)$, credible intervals

1: Obtain B samples $\{\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots, \boldsymbol{\theta}_B\}$ from the posterior distribution, $\boldsymbol{\theta}|\mathbf{X}, \mathbf{y}$

$$\boldsymbol{\theta}_i \sim \boldsymbol{\theta}|\mathbf{X}, \mathbf{y} \tag{3}$$

2: b $\leftarrow \lfloor B/10 \rfloor$ {number of burnin samples}
3: t $\leftarrow 5$ {tempering step}
4: Initialise ranking matrix $\mathbf{R} = (\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_p) = \mathbf{0}_{p \times B}$, $\mathbf{r}_i \in \mathbb{R}^p$
5: **for** $i = b$ to B step t **do**
6: Sort $\boldsymbol{\theta}_i = (\theta_{i1}, \theta_{i2}, \ldots, \theta_{ip})'$ by absolute magnitude $|\boldsymbol{\theta}_{ij}|$ in descending order
1. Denote the sorted parameter vector

$$\boldsymbol{\theta}_i^* = (\theta_{i1}^*, \theta_{i2}^*, \ldots, \theta_{ip}^*)'$$

7: Compute ranking $\mathbf{r}_i = (r_{i1}, r_{i2}, \ldots, r_{ip})'$ from $\boldsymbol{\theta}_i^*$ for all p features
1. The rank of feature j is r_{ij}, where $1 \leq r_{ij} \leq p$
2. Absolute value of $|\theta_{ij}^*|$ determines rank of feature j
3. If $r_{ij'} = 1$ then $|\theta_{ij'}^*| \geq |\theta_{ij}^*|$, $\forall j \neq j'$

8: **end for**
9: For each feature, compute the 25th and 75th rank percentiles using rank matrix $\mathbf{R}$
10: Compute the final feature ranking, $\tilde{\mathbf{r}}$, using $\mathbf{R}$
1. Sort the 75th percentiles for each feature in ascending order
2. Final rank of feature j is $\tilde{r}_j$, where $1 \leq \tilde{r}_j \leq p$
3. If $\tilde{r}_{j'} = 1$ then 75th percentile for feature j is smaller than all $j \neq j'$

11: Compute 95% CI for features from 2.5th and 97.5th percentiles of $\mathbf{R}$

The algorithm begins by choosing which of the posterior samples will be used for ranking of the p covariates. The first ten percent of the initial posterior samples are discarded as *burnin* and then every $t = 5$th sample is used for the ranking method (Lines 1–3). The covariates are ranked based on the absolute magnitude of the parameters for each accepted sample, resulting in a set of possible rankings for the p covariates (Lines 5–8). The final ranking of the covariates, based on the 75-th percentile, and the 95% credible interval are computed in Line 10 and Line 11, respectively. The algorithm does not specify the type of

sampler that should be used to generate the B posterior samples since, in theory, any reasonable sampling approach should result in a sensible covariate ranking procedure. This is furthed discussed in Section 3.

3 Discussion and Results

The BFR algorithm is now empirically compared against two popular feature selection methods: (i) random forests (RF) with default parameters [14], and (ii) independence screening by generalized correlation (HM) [7]. As this is a preliminary investigation of BFR, the empirical comparison will concentrate on the problem of covariate selection in the linear regression model. A Bayesian ridge regression sampler was chosen for the implementation of the BFR algorithm as it is: (i) similar to the commonly used method of least squares, and (ii) applicable when the number of covariates is greater than the sample size. The hierarchy depicting the Bayesian ridge regression [13] is

$$\begin{aligned} \mathbf{y}|\mathbf{X}, \boldsymbol{\beta}, \sigma^2 &\sim \mathrm{N}_n(\mathbf{X}\boldsymbol{\beta}, \sigma^2\mathbf{I}_n), \\ \boldsymbol{\beta} &\sim \mathrm{N}_p(\mathbf{0}_p, \sigma^2/\lambda^2\mathbf{I}_p), \\ \sigma^2 &\sim \sigma^{-2}d\sigma^2, \\ \lambda &\sim \mathrm{Gamma}(1, 0.01), \end{aligned}$$

where $\boldsymbol{\beta} \in \Theta \subset \mathbb{R}^p$ are the regression parameters, σ^2 is a normally distributed noise variable and λ is the ridge regularization parameter.

The three ranking methods will be compared on simulated data, where the true covariate set is known in advance, as well as two real data sets. The complete simulation code was written on the MATLAB numerical computing platform and is available for download from `www.emakalic.org/blog`.

3.1 Simulated Data

The BFR, RF and HM ranking methods are now compared on three linear regression functions borrowed from the simulation setup in [6]. For each of the three functions, we generated 100 data sets, with each data set comprising $n = 50$ samples and $p = 100$ covariates. All the generated data sets were standardised such that each covariate had a mean of zero and unit length. Noise was added to the target variables such that the signal-to-noise ratio (SNR) was in the set $\{1, 8\}$. The functions used for testing are detailed below.

(**Function** I). The generating regression coefficients were

$$\boldsymbol{\beta}^* = (1.24, -1.34, -1.35, -1.80, -1.58, -1.60, \mathbf{0}'_{p-6})',$$

where $\mathbf{0}_k$ is a k-dimensional zero vector. All predictors $\mathbf{x}_i$ $(i = 1, 2, \ldots, p)$ were generated from the standard Gaussian distribution, $\mathbf{x}_i \sim \mathrm{N}_n(0, 1)$.

(**Function** II). The generating regression coefficients were

$$\boldsymbol{\beta}^* = (4, 4, 4, -6\sqrt{2}, \mathbf{0}'_{p-4})'.$$

The predictors were marginally distributed as per a standard Gaussian distribution, $\mathbf{x}_i \sim \mathrm{N}_n(0, 1)$; the correlation between predictors was $\mathrm{corr}(X_i, X_4) = 1/\sqrt{2}$ for all $i \neq 4$; $\mathrm{corr}(X_i, X_j) = 1/2$ if i and j were distinct elements in $\{1, 2, \ldots, p\} \backslash \{4\}$.

(**Function** III). The generating regression coefficients were

$$\boldsymbol{\beta}^* = (4, 4, 4, -6\sqrt{2}, 4/3, \mathbf{0}'_{p-5})'.$$

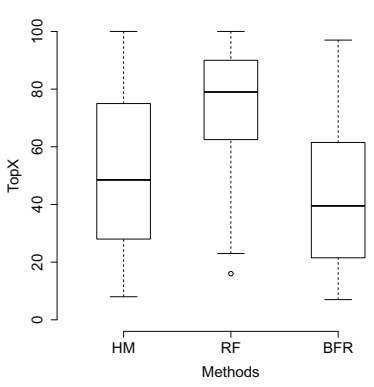
The predictors were marginally distributed as per a standard Gaussian distribution, $\mathbf{x}_i \sim \mathrm{N}_n(0, 1)$; the correlation between predictors was $\mathrm{corr}(X_i, X_5) = 0$ for all $i \neq 5$, $\mathrm{corr}(X_i, X_4) = 1/\sqrt{2}$ for all $i \notin \{4, 5\}$ and $\mathrm{corr}(X_i, X_4) = 1/2$ if both i and j were distinct elements in $\{1, 2, \ldots, p\} \backslash \{4, 5\}$.

Function I consists of independently generated covariates, while functions II and III contain varying levels of correlation. Feature selection is therefore expected to be somewhat more difficult for Functions II and III in contrast to function I. The ranking methods were compared on the `TopX` metric: the rank below which all the true features are included. For example, for Function I, a `TopX` of 15 indicates that the true six features are included among the first 15 selected covariates; the minimum possible `TopX` values for the three examples are six, four and five respectively. Box-and-whisker plots of the `TopX` metric for each method on the three test functions are depicted in Figure 1.

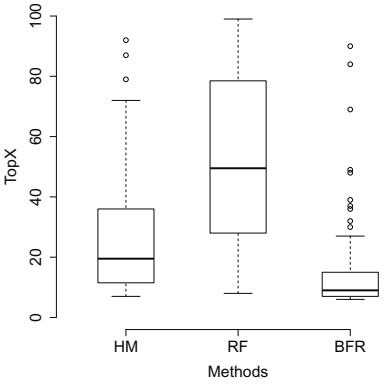
For all the tests functions, the BFR algorithm exhibited the smallest value of the median `TopX` metric of all the ranking methods considered. This was especially evident when the signal-to-noise ratio was larger, indicating that BFR is able to adapt well to varying levels of noise. Unsurprisingly, all three ranking methods performed better on function I, especially when SNR=8, in contrast to functions II and III. The HM algorithm performed better than random forests and slightly worse than the BFR method on all three test functions considered. As the amount of noise was decreased (SNR $\gg 8$), the performance of the three methods became indistinguishable.

3.2 Real Data

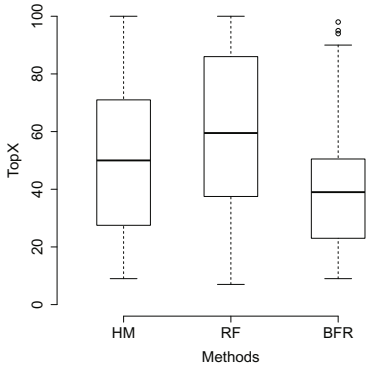
The performance of the three methods was also examined on two real data sets: (i) the diabetes data set ($n = 442, p = 10$) downloaded from Trevor Hastie's homepage and analysed in [8], and (ii) the communities and crime data set ($n = 319, p = 123$) obtained from the UCI machine learning repository. Each data set was standardised similarly to the simulation data in Section 3.1. As the second data set contained a number of missing attributes, rows where one or more variables had missing entries were removed before analysis. The HM, RF and BH ranking of the $p = 10$ features for the diabetes data set is shown in Table 1.



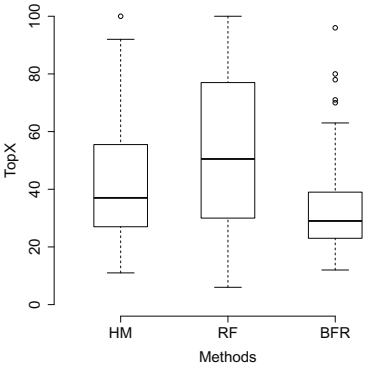
(a) Function I, SNR=1



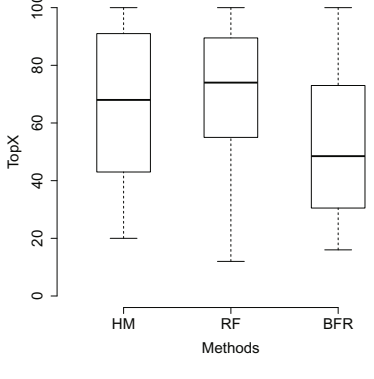
(b) Function I, SNR=8



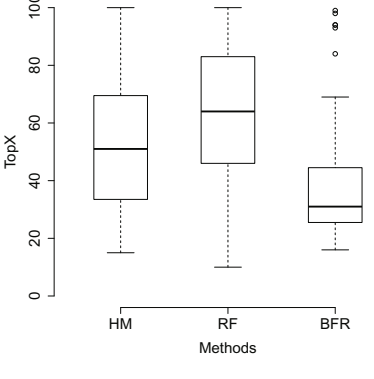
(c) Function II, SNR=1



(d) Function II, SNR=8



(e) Function III, SNR=1



(f) Function III, SNR=8

Fig. 1. Comparison of feature ranking methods on three test functions using the `TopX` metric

Table 1. HM, RF and BFR ranking of the ten features in the diabetes data set

Method	Feature rank									
	age	sex	bmi	map	tc	ldl	hdl	tch	ltg	glu
HM	3	9	4	7	8	10	5	1	6	2
RF	9	3	4	8	7	10	5	2	6	1
BFR	3	9	2	4	7	8	5	6	10	1

The top seven covariates selected by HM and RF were identical though with a slightly different ordering. All three ranking methods selected the `bmi` and `ltg` variables as the two most important features in terms of explanatory power. The BFR ranking is mostly similar to both HM and RF with one significant exception; BFR ranked the `sex` covariate much higher than the other ranking algorithms. Similarly, the BFR procedure ranked `glu` much lower in contrast to both HM and RF.

The performance of HM, RF and BFR was also examined on the communities and crime data set. Here, a five-fold cross validation procedure was used to estimate the generalisation error for each of the three methods over 100 test iterations. The mean squared prediction error for the BFR algorithm is shown in Figure 2. We notice that the generalisation error decreases sharply as the first few features are added to the model. The generalisation error begins to increase after approximately 30 features are included and smoothly rises until all $p = 123$

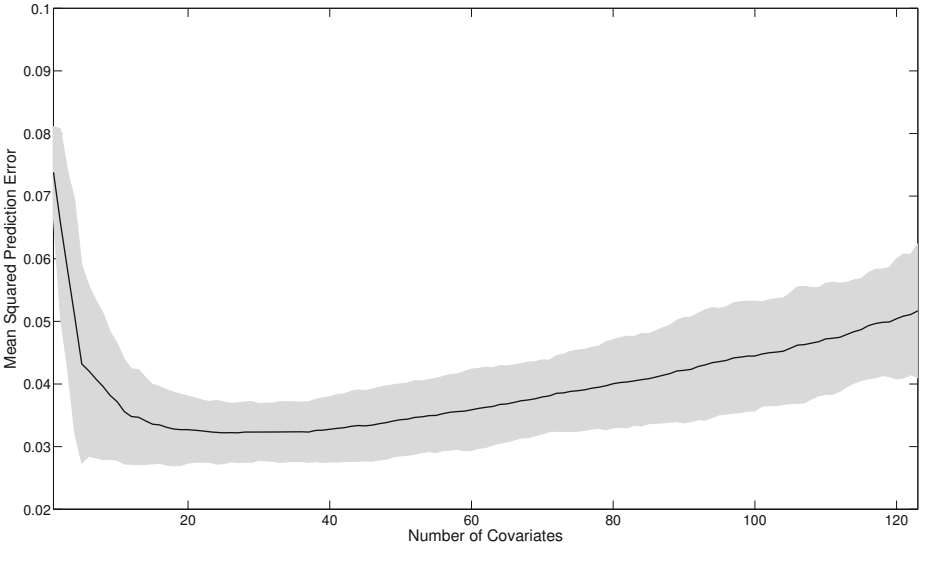


Fig. 2. Bayesian feature ranking for the communities and crime data set; standard errors represented by the shaded area

features are in the model. Importantly, the generalisation error does not drop after the first 30 features were included which indicates that BFR has included all the important features in the first 30 covariates. For this data set, both HM and RF algorithms were virtually indistinguishable from BFR and hence omitted from the plot for reasons of clarity.

4 Conclusion

This paper has presented a new Bayesian algorithm for feature ranking based on sampling from the posterior distribution of the parameters given the data. The new algorithm was applied to the linear regression model using both simulated and real data sets. BFR resulted in reasonable feature ranking in all empirical simulations, often outperforming random forests and feature ranking by generalised correlation. The excellent performance of BFR suggests that the idea is worthy of further exploration. Future work includes empirical examination of the sensitivity of BFR to the choice of Bayesian hierarchy, as well as application of BFR to feature ranking in classification problems.

References

1. Breiman, L.: Better subset regression using the nonnegative garrote. Technometrics 37, 373–384 (1995)
2. Tibshirani, R.: Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society (Series B) 58(1), 267–288 (1996)
3. Zou, H., Hastie, T.: Regularization and variable selection via the elastic net. Journal of the Royal Statistical Society (Series B) 67(2), 301–320 (2005)
4. Zou, H.: The adaptive lasso and its oracle properties. Journal of the American Statistical Association 101(476), 1418–1429 (2006)
5. James, G.M., Radchenko, P.: A generalized Dantzig selector with shrinkage tuning. Biometrika 96(2), 323–337 (2009)
6. Fan, J., Samworth, R., Wu, Y.: Ultrahigh dimensional feature selection: Beyond the linear model. Journal of Machine Learning Research 10, 2013–2038 (2009)
7. Hall, P., Miller, H.: Using generalized correlation to effect variable selection in very high dimensional problems. Journal of Computational and Graphical Statistics 18(3), 533–550 (2009)
8. Efron, B., Hastie, T., Johnstone, I., Tibshirani, R.: Least angle regression. The Annals of Statistics 32(2), 407–451 (2004)
9. Friedman, J., Hastie, T., Höfling, H., Tibshirani, R.: Pathwise coordinate optimization. The Annals of Applied Statistics 1(2), 302–332 (2007)
10. Zou, H., Hastie, T., Tibshirani, R.: On the "degrees of freedom" of the lasso. The Annals of Statistics 35(5), 2173–2192 (2007)
11. Leng, C., Lin, Y., Wahba, G.: A note on the lasso and related procedures in model selection. Statistica Sinica 16(4), 1273–1284 (2006)
12. Park, T., Casella, G.: The Bayesian lasso. Journal of the American Statistical Association 103(482), 681–686 (2008)
13. Kyung, M., Gill, J., Ghosh, M., Casella, G.: Penalized regression, standard errors, and Bayesian lassos. Bayesian Analysis 5(2), 369–412 (2010)
14. Breiman, L.: Random forests. Machine Learning 45(1), 5–32 (2001)