Group Project 2 & 3

Xinlei Chen, Guojing Wu, Yujing Yao

Introduction

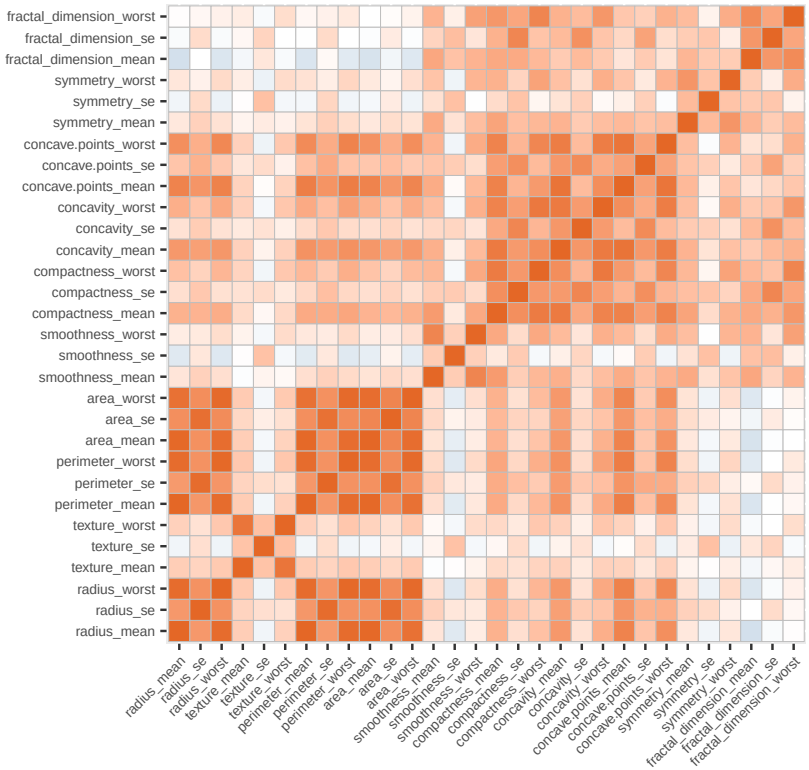
- **Group project 2:** Optimization algorithms on a breast cancer diagnosis dataset
- **Group project 3:** Bootstrapping on developing classification model

Breast Cancer Data

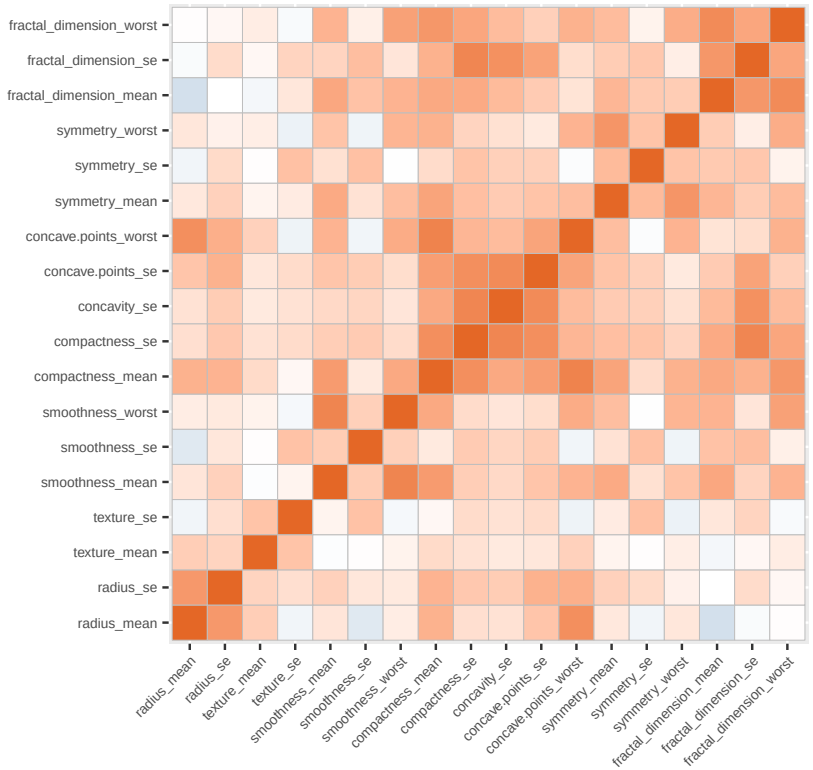
- **Amin:** Build a predictive model based on logistic regression to faciliate cancer diagnosis, and we compared methods including Newton Raphson, Gradient Decent with general logistic regression and Pathwise Coordinate Descent with regularized logistic regression
- **Variable Selection:** Reduce multicollinearity based on both correlation coefficient and eigenvalue of correlation matrix

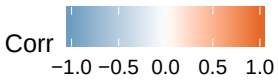
Multicollinearity plot of the dataset

A



B





Logistic Model with Newton-Raphson

Logistic Regression:

y: the vector of n response random variable

X: the $n \times p$ design matrix (X_i denote the ith row)

β: the $p \times 1$ coefficient

- The logistic regression model:

$$\log(\frac{\eta}{1-\eta}) = X\beta$$

- The likelihood function:

$$L(\beta; X, y) = \prod_{i=1}^{n} \{(\frac{\exp(X_i\beta)}{1+\exp(X_i\beta)})^{y_i}(\frac{1}{1+\exp(X_i\beta)})^{1-y_i}\}$$

- The log likelihood:

$$l(\beta) = \sum_{i=1}^{n} \{y_i(X_i\beta) - \log(1 + \exp(X_i\beta))\}$$

- The gradient:

$$\nabla l(\beta) = X^T(y - p)$$

  where $p = \frac{\exp(X\beta)}{1+\exp(X\beta)}$

- The Hessian:

$$\nabla^2 l(\beta) = -X^T W X$$

  where $W = diag(p_i(1 - p_i)), i = 1, \cdots, n$. The Hessian is negative definite.

Newton-Raphson

Update coefficients

$$\beta_{i+1} = \beta_i - [\nabla^2 l(\beta_i)]^{-1} \nabla l(\beta_i)$$

Step-halving

$$\beta_{i+1}(\gamma) = \beta_i - \gamma [\nabla^2 l(\beta_i)]^{-1} \nabla l(\beta_i)$$

- Set $\gamma = 1$
- If $f(\theta_{i+1}(1)) \geq f(\theta_i)$, then set $\theta_{i+1} = \theta_{i+1}(1)$
- If $f(\theta_{i+1}(1)) \leq f(\theta_i)$, search for a value $\gamma \in (0, 1)$ for which $f(\theta_{i+1}(\gamma)) \geq f(\theta_i)$, set $\theta_{i+1} = \theta_{i+1}(\gamma)$

Newton-Raphson: gradient decent

For Newton's method with a large p, the computational burden in calculating the inverse of the Hessian Matrix $[\nabla^2 f(\beta_i)]^{-1}$ increases quickly with p. One can update

$$\beta_{i+1} = \beta_i + H_i \nabla f(\beta_i)$$

where $H_i = (X^T X)^{-1}$ for every i. This is easy to compute, but could be slow in convergence.

The steps are:

- get the objective (loglik,grad,Hess) function
- use the principle of newton raphson to update the estimate, if the step size too large, step-halving step
- stop searching until the convergences of the estimates.

Logistic-LASSO Model with Pathwise Coordinate Descent

- Applied coordinate-wise descent with weighted update:

$$\tilde{\beta}_j^{lasso}(\lambda) \leftarrow \frac{S(\sum_{i=1}^{n} \omega_i x_{i,j}(y_i - \tilde{y}_i^{(-j)}), \lambda)}{\sum_{i=1}^{n} \omega_i x_{i,j}^2}$$

  where $\tilde{y}_i^{(-j)} = \sum_{k \neq j} x_{i,k} \tilde{\beta}_k$ and $S(\hat{\beta}, \lambda) = sign(\hat{\beta})(|\hat{\beta}| - \lambda)_+$

- In the context of logistic regression, we are aiming to maximize the penalized log likelihood:

$$\max_{\beta \in \mathbb{R}^{p+1}} \frac{1}{n} \sum_{i=1}^{n} \{y_i(X_i\beta) - \log(1 + \exp(X_i\beta))\} - \lambda \sum_{j=0}^{p} |\beta_j|$$

  for some $\lambda \geq 0$

Estimation Path and Cross validation for LASSO

Figure 2: A path of solutions with a sequence of descendi

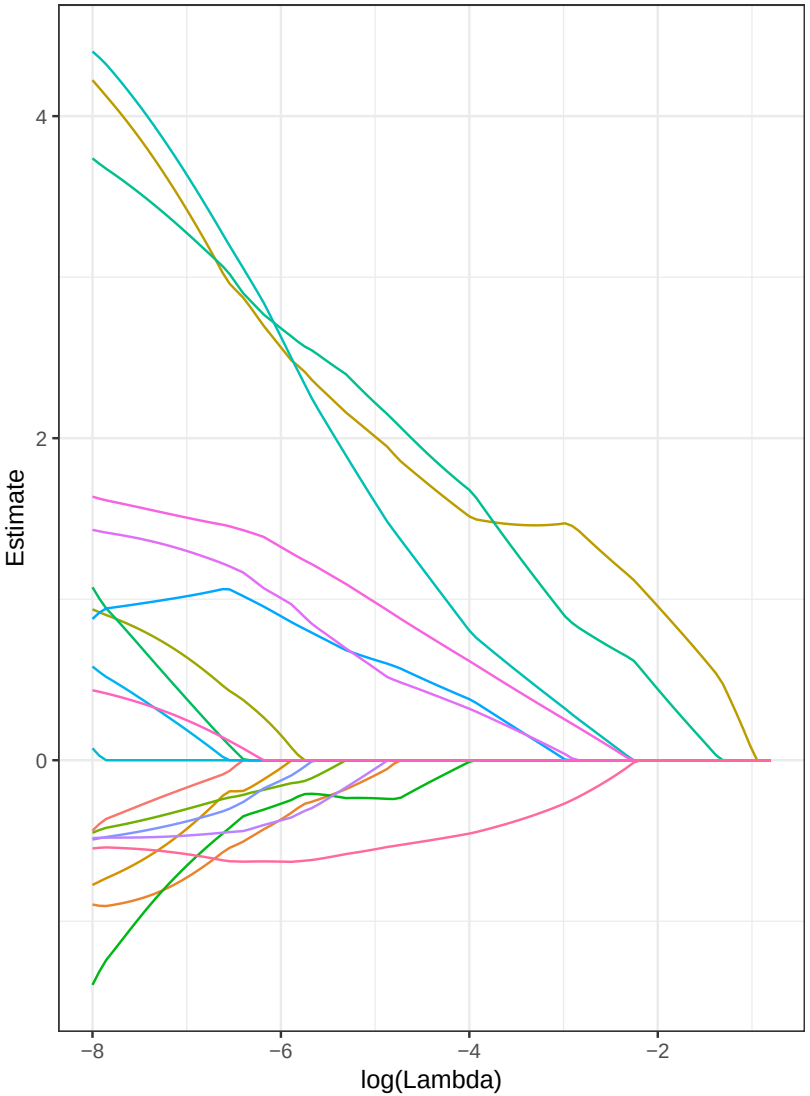
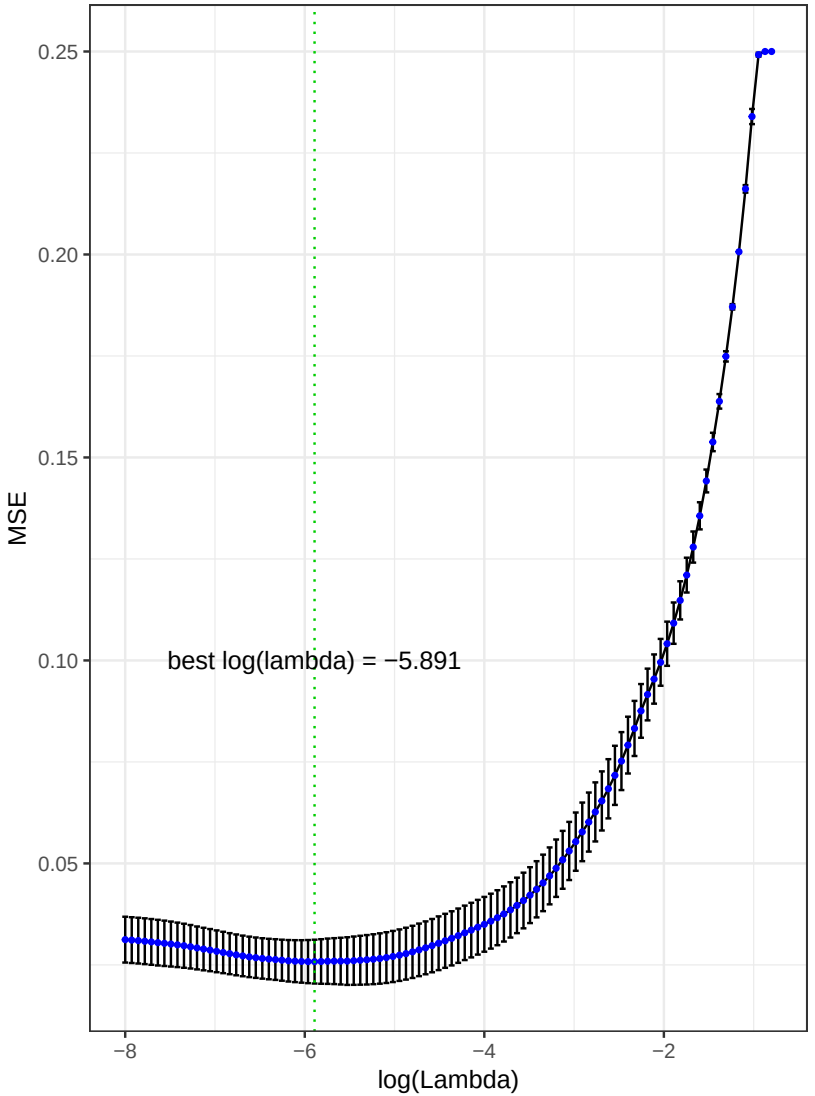


Figure 3: Lasso regression by 5 fold cross validation



Model Comparison

Table 1: The comparison of performance for estimation algorithms and models

	GLM package	Newton Raphson	Gradient Decent	Logistic Lasso	Lasso package
iteration times	NA	12	1001	100	NA
MSE	0.02	0.02	0.02	0.02	0.02

[a] Dataset: Breast Cancer Diagnosis

Model Comparison

Table 2: The comparison of performance for estimation algorithms and models

	GLM package	Newton Raphson	Gradient Decent	Logistic Lasso	Lasso package
radius_mean	4.43	4.43	3.18	2.63	2.71
texture_mean	1.89	1.89	1.34	1.29	1.37
smoothness_mean	0.78	0.78	0.47	0.00	0.00
compactness_mean	-1.14	-1.14	-0.59	0.00	0.00
symmetry_mean	-0.63	-0.63	-0.44	-0.10	-0.14
fractal_dimension_mean	-0.66	-0.66	-0.72	-0.14	-0.21
radius_se	5.13	5.13	3.28	2.50	2.58
texture_se	0.59	0.59	0.46	0.00	0.00
smoothness_se	1.10	1.10	0.77	0.00	0.00
compactness_se	-0.80	-0.80	-0.68	-0.33	-0.38
concavity_se	1.24	1.24	0.88	0.08	0.19
concave.points_se	-1.11	-1.11	-0.80	0.00	0.00
symmetry_se	-0.53	-0.53	-0.39	-0.36	-0.42
fractal_dimension_se	-2.73	-2.73	-1.55	-0.25	-0.31
smoothness_worst	0.31	0.31	0.31	0.86	0.92
concave.points_worst	5.13	5.13	3.65	2.48	2.62
symmetry_worst	1.60	1.60	1.28	0.97	1.06
fractal_dimension_worst	2.19	2.19	1.41	0.00	0.00
intercept	-0.62	-0.62	-0.71	-0.63	-0.77

Conclusion

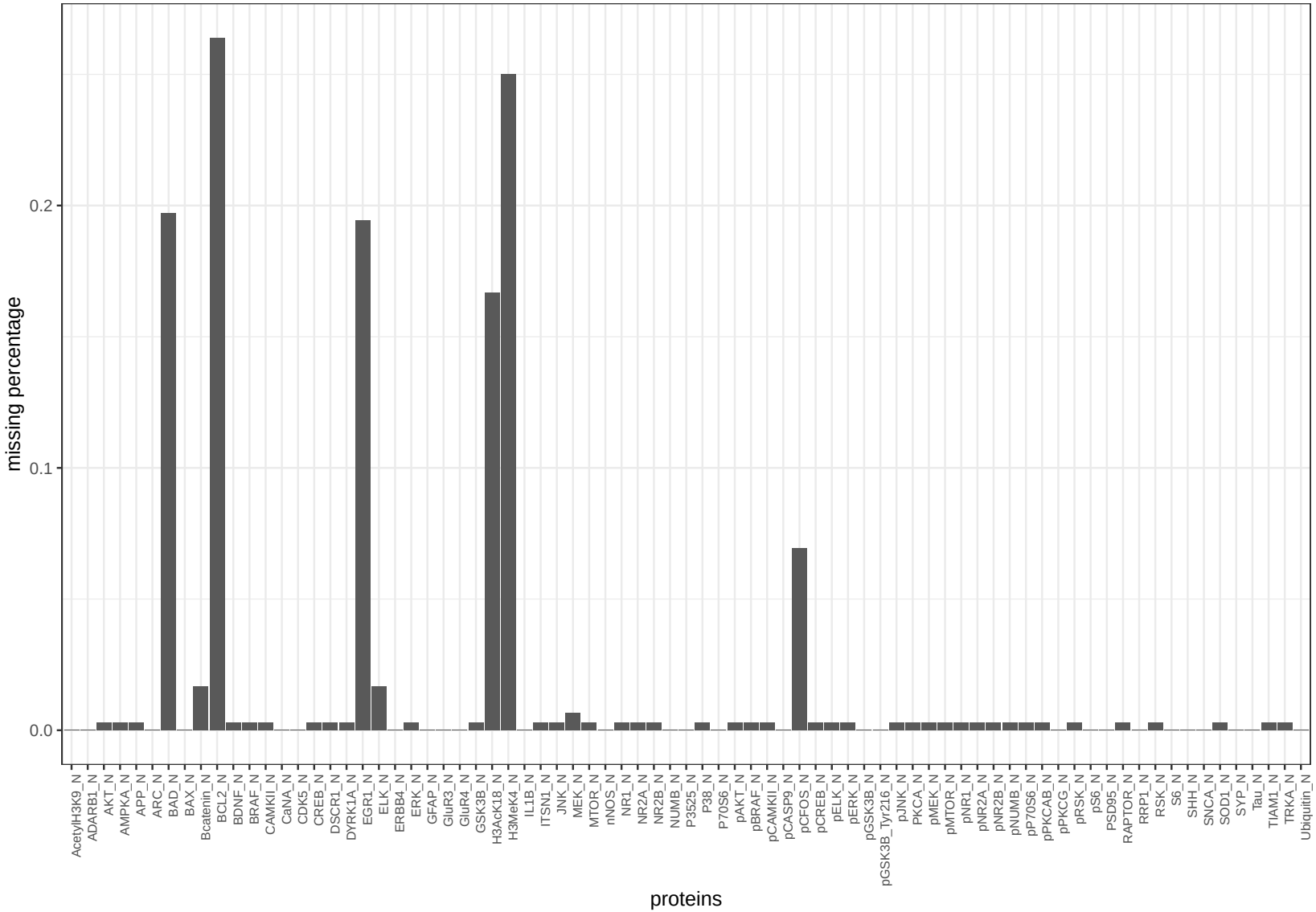
- The results of our methods are compared to the the same parameter estimation as R's built-in packages
- Newton-Raphson has the convincing estimation and it converged quickly
- Gradient decent method showed similar estimation as Newton-Raphson method but it was less efficient
- For Pathwise Coordinate descent with LASSO logistic, according to the result of 5 fold cross validation and estimation result, the λ with the lowest MSE and it shrunk six parameters to zero, which is comparable to the result by R's built-in packages.

Down Syndrome Data

- **Aim:** Build a predictive model based on logistic regression to faciliate down syndrome diagnosis, and compared methods including and Pathwise Coordinate Descent with regularized logistic regression and smoothed bootstrap estimation.
- **Variable Selection:** Delete variables with high missing rate ($\geq 20\%$)

Also due to the intrinsic correlation between individual proteins (Fig. 2), it's impossible to apply normal regression methods to this dataset because of sigularity propblem. Instead, we choose regularized methods, LASSO, to be more specific.

Figure 1: proteins' missing percentage



Smoothed Bootstrap Estimation (SBE) and Inference

- First we need to prepare a couple of candidate models
- for each bootstrap in bootstrap with B times, select the best model and get estimates for the coefficient denoted as $t(y^*)$
- smooth $\hat{\mu} = t(y)$ by averaging over the bootstrap replications, defining

$$\tilde{\mu} = s(y) = \frac{1}{B}\sum_{i=1}^{B} t(y^*)$$

And in addition to the percentile confidence interval, the nonparametric delta-method estimate of standard deviation for $s(y)$ in the nonideal case is:

$$\tilde{sd}_B = [\sum_{i=1}^{n} \hat{cov}_j^2]^{1/2}$$

where

$$\hat{cov}_j = \sum_{i=1}^{B} (Y_{ij}^* - Y_{.j}^*)(t_i^* - t_.^*)/B$$

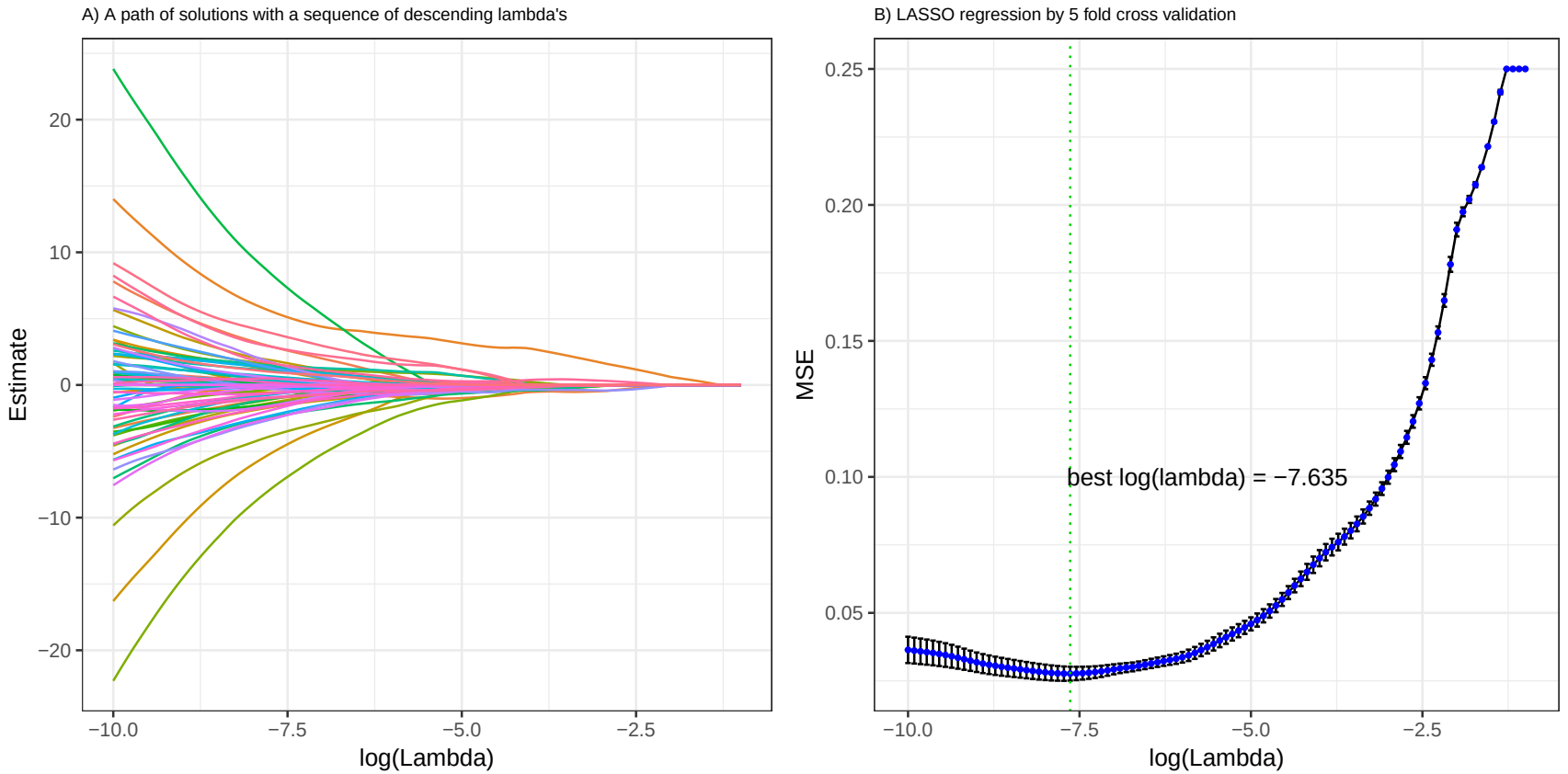
with $Y_{.j}^* = \sum_{i=1}^{B} Y_{ij}^*/B$ and $t_.^* = \sum_{i=1}^{B} t_i^*/B = s(y)$.

Result

Pathwise coordinate descent logistic-LASSO

Fig. 3A shows us that as the λ increases, all the variable estimates of parameters shrink accordingly since we penalize all the parameters. When $\lambda = 0$, the result is the same as least square method and when λ is too large, all the estimates of parameters shrink to 0. Fig. 3B shows us the cross validation result for choosing the best λ.

Figure 3: results of PCD-LASSO



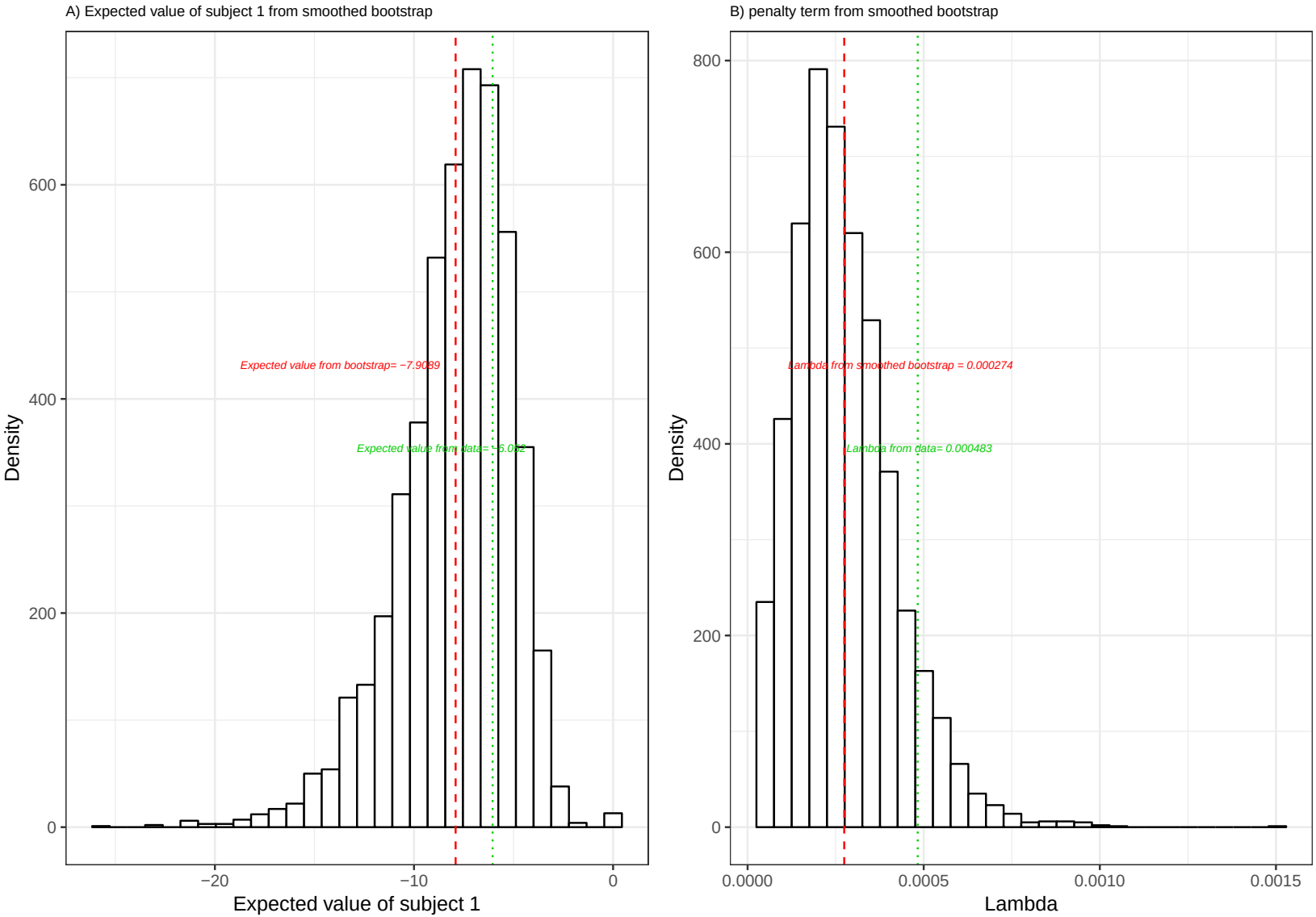
Model selection based on SBE for Logistic Lasso

algorithm:

- bootstrap data from the original dataset
- do cross validation and select the best λ_i^* for each repetition
- calculate average $\lambda^* = \frac{1}{B}\sum_{i=1}^{B}\lambda_i^*$

We can see the discrepancy between results of PCD-LASSO and SBE for Logistic LASSO both in prediction (Fig 4A) and finding the best λ (Fig 4B), the results of PCD-LASSO is deviated from the center of empirical distribution.

Figure 4: Lambda selection based on SBE for Logistic LASSO



Cross validation for model prediction comparison

We used 10 fold cross-validation to compare two different models, one is with λ selected from data, the other is selected from the SBE. Table 1 shows us that while the Cross Validation MSE are similar between these two methods, SBE provides a more accurate classification result.

Significant random variable selection from SBE

Table 2 & 3 provide the full results of SBE for logistic LASSO. Our identification criterions here are:

1. the chosen probability greater than 96%
2. SBE confidence interval excludes zero.

Based on that, we got 27 proteins that meets these two criterions (Table 1).

Conclusion

In this study, we first used cross validation to find the best λ for PCD-LASSO prediction. Then we used bootstrap to plot empirical distribution of the best λ for LASSO and the prediction of subject 1, which proved that classical statistical theory does ignore model selection in assessing estimation accuracy. The cross validation result of comparing PCD-LASSO best λ and SBE best λ also showed that SBE method provide a better result. Then based on the chosen probability and SBE CI, we identified a subset of proteins that are significantly associated with the Down syndrome.