KSBi-BIML 2023

Bioinformatics & Machine Learning(BIML)
Workshop for Life Scientists, Data Scientists,
and Bioinformatians

생물정보학&머신러닝 워크샵(온라인)

Shrinkage Methods and Tree Ensembles for High-dimensional Sparse Data

황규백 _ 숭실대학교

SBi 한국생명정보학회
Korean Society for Bioinformatics

본 강의 자료는 한국생명정보학회가 주관하는 BIML 2023 워크샵 온라인 수업을 목적으로 제작된 것으로 해당 목적 이외의 다른 용도로 사용할 수 없음을 분명하게 알립니다.

이를 다른 사람과 공유하거나 복제, 배포, 전송할 수 없으며 만약 이러한 사항을 위반할 경우 발생하는 **모든 법적 책임은 전적으로 불법 행위자 본인에게 있음을 경고**합니다.

KSBi-BIML 2023

Bioinformatics & Machine Learning (BIML)
Workshop for Life Scientists, Data Scientists, and Bioinformatians

안녕하십니까?

한국생명정보학회가 개최하는 동계 교육 워크샵인 BIML-2023에 여러분을 초대합니다. 생명정보학 분야의 연구자들에게 최신 동향의 데이터 분석기술을 이론과 실습을 겸비해 전달하고자 도입한 전문 교육 프로그램인 BIML 워크샵은 2015년에 시작하여 올해로 9차를 맞이하게 되었습니다. 지난 2년간은 심각한 코로나 대유행으로 인해 아쉽게도 모든 강의가 온라인으로 진행되어 현장 강의에서만 가능한 강의자와 수강생 사이에 다양한 소통의 기회가 없음에 대한 아쉬움이 있었습니다. 다행히도 최근 사회적 거리두기 완화로 현장 강의가 가능해져 올해는 현장 강의를 재개함으로써 온라인과 현장 강의의 장점을 모두 갖춘 프로그램을 구성할 수 있게 되었습니다.

BIML 워크샵은 전통적으로 크게 인공지능과 생명정보분석 두 개의 분야로 구성되었습니다. 올해 AI 분야에서는 최근 생명정보 분석에서도 응용이 확대되고 있는 다양한 심층학습(Deep learning) 기법들에 대한 현장 강의가 진행될 예정이며, 관련하여 심층학습을 이용한 단백질구조예측, 유전체 분석, 신약개발에 대한 이론과 실습 강의가 함께 제공될 예정입니다. 또한 싱글셀오믹스 분석과 메타유전체분석 현장 강의는 많은 연구자의 연구 수월성 확보에 큰 도움을 줄 것으로 기대하고 있습니다. 이외에 다양한 생명정보학 분야에 대하여 30개 이상의 온라인 강좌가 개설되어 제공되며 온라인 강의의 한계를 극복하기 위해서 실시간 Q&A 세션 또한 마련했습니다. 특히 BIML은 각 분야 국내 최고 전문가들의 강의로 구성되어 해당 분야의 기초부터 최신 연구 동향까지 포함하는 수준 높은 내용의 강의가 될 것입니다.

이번 BIML-2023을 준비하기까지 너무나 많은 수고를 해주신 BIML-2023 운영위원회의 남진우, 우현구, 백대현, 정성원, 정인경, 장혜식, 박종은 교수님과 KOBIC 이병욱 박사님께 커다란 감사를 드립니다. 마지막으로 부족한 시간에도 불구하고 강의 부탁을 흔쾌히 허락하시고 훌륭한 현장 강의와 온라인 강의를 준비하시는데 노고를 아끼지 않으신 모든 연사분께 깊은 감사를 드립니다.

2023년 2월

한국생명정보학회장 이 인 석

강의개요

Shrinkage Methods and Tree Ensembles for High-dimensional Sparse Data

생물정보학에서 다루는 많은 데이터들은 변수의 개수는 많지만 표본 크기는 "상대적으로 작은" 고차원 희박 데이터(high-dimensional sparse data)이다. 예를 들어 마이크로어레이나 RNA 시퀀싱으로 얻어지는 유전자 발현 데이터는 수천 ~ 수만 개의 유전자에 대한 발현 정보를 가지고 있지만 표본의 크기는 대부분 수백 ~ 수만에 지나지 않는다.

본 강의에서는 고차원 희박 데이터가 기계학습에 어떠한 악영향을 미치는지를 직관적으로 설명하고, 이러한 데이터를 분석하는 데 널리 사용되는 shrinkage 방법과 tree ensemble에 대해 설명한다. 선형회귀 및 로지스틱 회귀 기반의 shrinkage 방법이 어떠한 전략으로 고차원 희박 데이터 문제를 해결하는지 설명하고, 그 구체적인 활용 방법에 대해 강의한다. 또한, 고차원 희박 데이터를 다룰 수 있는 비선형 방법인 결정트리(decision tree) 기반의 tree ensemble도 상세히 다룬다.

강의는 다음의 내용을 포함한다:

- Bias-Variance Trade-Off
- 고차원 희박 데이터의 문제점
- Shrinkage 방법 (Ridge, Lasso, Elastic Net)
- Tree Ensemble (Bagging, Random Forest, Boosting)

* 참고강의교재:

An Introduction to Statistical Learning: with Applications in R (Springer, 2013)

* 교육생준비물:

노트북 (동영상 강의 시청용)

* 강의 난이도: 초급

* 강의: 황규백 교수 (숭실대학교 컴퓨터학부)

Curriculum Vitae

Speaker Name: Kyu-Baek Hwang, Ph.D.

▶ Personal Info

Name	Kyu-Baek Hwang
Title	Professor
Affiliation	Soongsil University

▶ Contact Information

Address	369 Sangdo-ro, Dongjak-gu, Soongsil University, Seoul 06978
Email	kbhwang@ssu.ac.kr
Phone Number	02-820-0925

Research Interest

Machine learning and bioinformatics

Educational Experience

1997	B.S.E. in Computer Engineering, Seoul National University, Korea
1999	M.S.E. in Computer Engineering, Seoul National University, Korea
2005	Ph.D. in Computer Science and Engineering, Seoul National University, Korea

Professional Experience

2004	Short-term Visiting Scholar, Children's Hospital Boston, USA
2012	Visiting Research Associate, Boston Children's Hospital, USA
2006-	Professor, Soongsil University, Korea

Selected Publications (5 maximum)

1. Hwang, K.-B.+, Lee, I.-H.+, Li, H., Won, D.-G., Hernandez-Ferrer, C., Negron, J.A., and Kong, S.W., Comparative analysis of whole-genome sequencing pipelines to minimize false negative findings, Scientific Reports, vol. 9, p. 3219, 2019.
2. Li, H.+, Park, J.+, Kim, H., Hwang, K.-B.*, and Paek, E.*, Systematic comparison of false-discovery-rate-controlling strategies for proteogenomic search using spike-in experiments, Journal of Proteome Research, vol. 16, no. 6, pp. 2231-2239, 2017.
3. Li, H., Joh, Y.S., Kim, H., Paek, E., Lee, S.-W., and Hwang, K.-B., Evaluating the effect of database inflation in proteogenomic search on sensitive and reliable peptide identification, BMC Genomics, vol., 17, no. Suppl 13, p. 3327, 2016.
4. Seok, H.-S., Song, T., Kong, S.W., and Hwang, K.-B., An efficient search algorithm for finding genomic-range overlaps based on the maximum range length, IEEE/ACM Transactions on Computational Biology and Bioinformatics, vol. 12, no. 4, pp. 778-784, 2015.

KSBi-BIML

Shrinkage Methods and Tree Ensembles for High-dimensional Sparse Data

황규백 (숭실대학교)

들어가면서

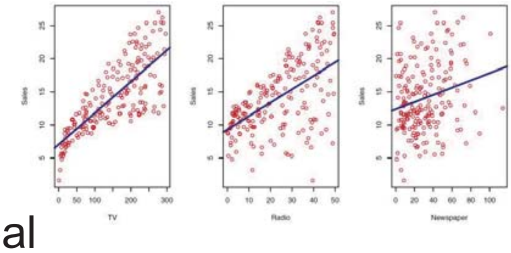
- 강의 내용
  - 편향-분산 딜레마
  - 선형회귀와 고차원 희박 데이터
  - Shrinkage 방법
  - Tree Ensemble 방법

- 참고 교재
  - An Introduction to Statistical Learning: with Applications in R (Springer, 2013)

Bias-Variance Trade-Off

A Machine Learning Example: Advertising Problem

- How to improve sales of a particular product
  - By controlling the advertising expenditure
- Data
  - Sales of the product in 200 different markets
  - Advertising budgets for the product in each of those markets for three different media: TV, radio, and newspaper



- Goal
  - Develop an accurate model for predicting sales given the three media budgets

A Statistical Learning Setting

- Input variables
  - TV budget (X_1), radio budget (X_2), and newspaper budget (X_3)
  - Different names
    - Predictors, independent variables, features, and variables
- Output variable
  - sales (Y)
  - Different names
    - Response, dependent variable, and target variable
- Our assumption
  - $Y = f(\mathbf{X}) + \varepsilon$
  - f: a function
  - ε: an error term

Statistical (Machine) Learning

- We try to estimate "f" from a given (training) data set
- Machine learning is about a set of approaches to estimating the f
- Diverse disciplines are related to machine learning
  - Computer science
  - Electronic engineering
  - Statistics

Types of Machine Learning

- Supervised learning
  - A target variable (Y) is given
  - Regression vs classification
  - Disease diagnosis based on a lab test
- Unsupervised learning
  - There is no target variable
  - Exploratory data analysis; feature extraction
  - Clustering of genes based on their expression patterns
- Reinforcement learning
  - Instead of a target variable, *reward* is given to an agent
  - AlphaGo
  - Robot navigation (mapping and localization)

Types of Supervised Learning

- Quantitative target-variables
  - Numerical values
  - Age, height, income, sales
  - Regression
    - Advertising problem
- Qualitative target-variables
  - Categorical values
  - Gender, cancer diagnosis
  - Classification

Why Estimate f?

- Prediction
  - If we estimated f, we can use it for predicting the value of Y (output variable) for a specific x
- Inference
  - We are interested in understanding the way that Y is affected as X_1, …, X_p change
- Possible questions addressed
  - Which predictors are associated with the response?
  - What is the relationship between the response and each predictor?
    - Increasing the predictor will increase or decrease the response
  - Can the relationship between Y and each predictor be adequately summarized using a linear equation, or is the relationship more complicated?

Performance of a (Learned) Regression Model: Mean Squared Error (MSE)

- Average difference between the true observed-response (y_i) and the predicted one ($\hat{f}(x_i)$)
  - If we have a training data (**X** and **y**)

$$\mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} \quad \mathbf{X} = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{pmatrix} \qquad MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{f}(x_i))^2$$

  - A.k.a. Training MSE
- However, we are more interested in MSE for future observations
  - Stock market prediction
  - Diabetes risk prediction

Test MSE

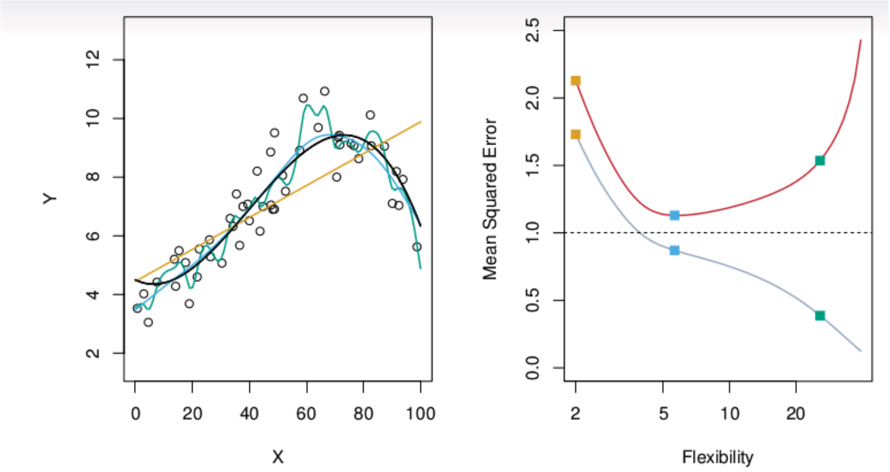
- We could think about MSE over test observations (x_0, y_0)

  $$Ave(y_0 - \hat{f}(x_0))^2$$

  – Minimization of test MSE is required!!!

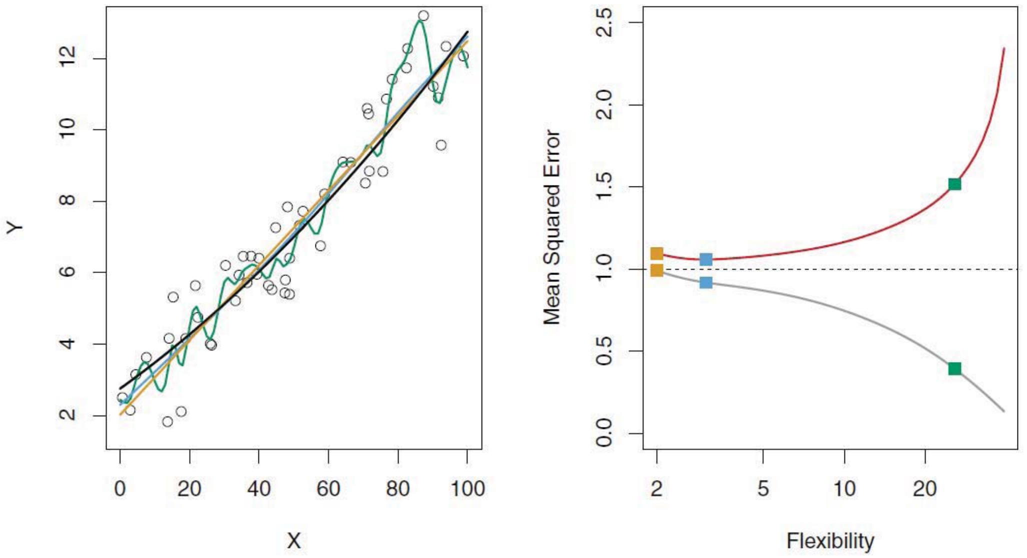
- How can we minimize test MSE
  – If we have a set of test observations, the problem is simple
    - Test observations are not used for training
  – What if we do not have test observations?
    - Can we use training MSE instead of test MSE for assessing models?

Training MSE vs Test MSE



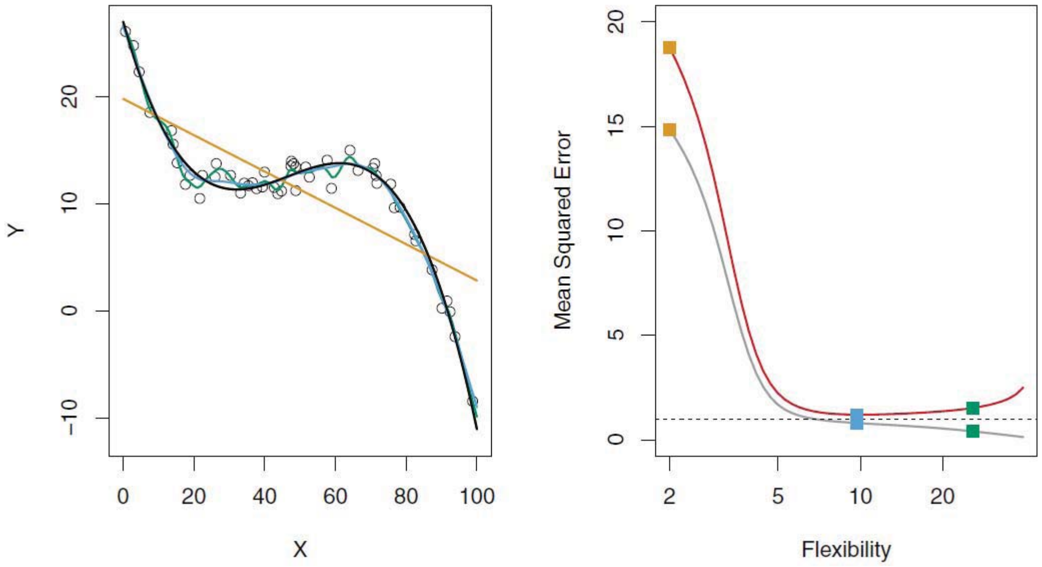
- Simulation experiments
  – Black curve: truth
  – Circles: training data (sampled from the black curve)
  – Orange, blue, and green curves: learned results with differing complexity levels (different machine learning models)
  – Overfitting phenomenon

A Smoother True Function



- How does the linear line work?

A More Flexible Truth Function



- Linear line now?

How Come Such a Phenomenon Occurs

- Mathematical proof is possible
- We are concerned with

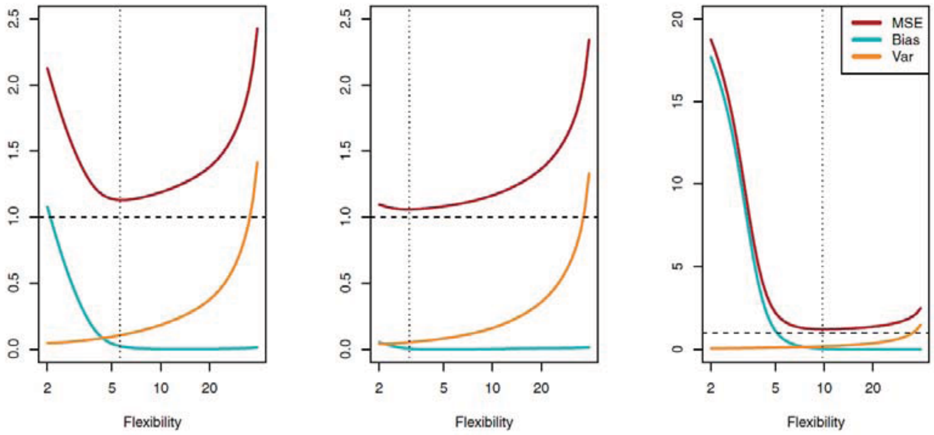
$$Ave(y_0 - \hat{f}(x_0))^2$$

- It can be decomposed as

$$E(y_0 - \hat{f}(x_0))^2 = Var\left(\hat{f}(x_0)\right) + \left[Bias(\hat{f}(x_0))\right]^2 + Var(\varepsilon)$$

  - Expectation over training observations (= training data)
    - Variance
      - The amount by which f (learned result) changes according to the given training data set
    - Bias
      - The error introduced by modeling the given problem using a machine learning model

Observations vs Theory

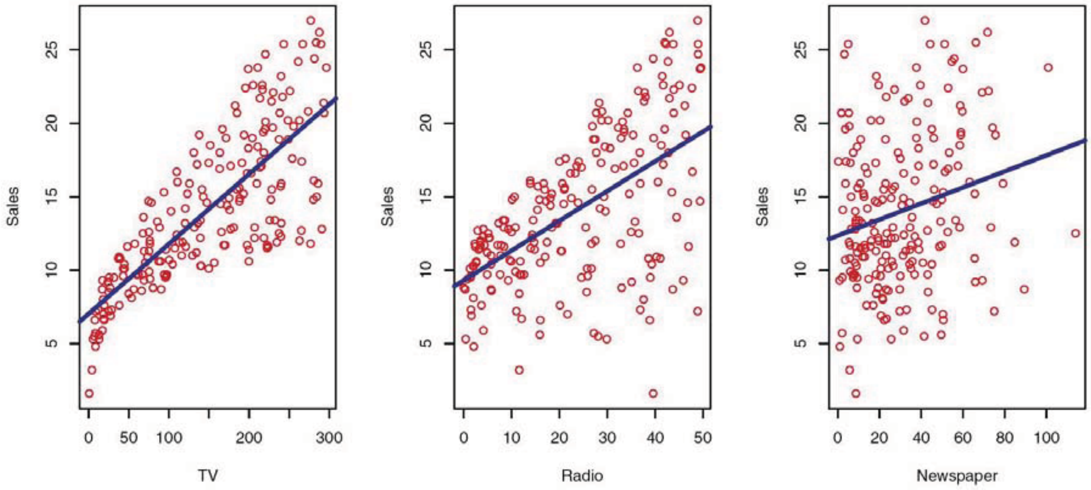


- The bias-variance trade-off
- Training errors decrease as the model complexity increases
- Test errors show a u-shaped curve
  - We must choose an appropriate level of model complexity to obtain a good test error

Linear Regression & High-Dimensional Sparse Data

Regression for the Advertising Data Set

- We have a data set (Advertising)
  - Sales (Y), TV (X_1), radio (X_2), and newspaper (X_3) (from 200 cities)
  - $Y = f(X_1, X_2, X_3) + \varepsilon$



SBi 한국생명정보학회 Korean Society for Bioinformatics

Multiple Linear Regression

- Regression formula

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p + \epsilon$$

$$\texttt{sales} = \beta_0 + \beta_1 \times \texttt{TV} + \beta_2 \times \texttt{radio} + \beta_3 \times \texttt{newspaper} + \epsilon$$

- Meaning of β_j
  - Average effect of X_j on Y when all other predictor values are fixed

Estimation of the Coefficients in Multiple Linear Regression

- We estimate $\beta_0, \beta_1, \ldots, \beta_p$ as the values that minimize the sum of squared residuals

$$\begin{aligned}\text{RSS} &= \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \\ &= \sum_{i=1}^{n}(y_i - \hat{\beta}_0 - \hat{\beta}_1 x_{i1} - \hat{\beta}_2 x_{i2} - \cdots - \hat{\beta}_p x_{ip})^2\end{aligned}$$

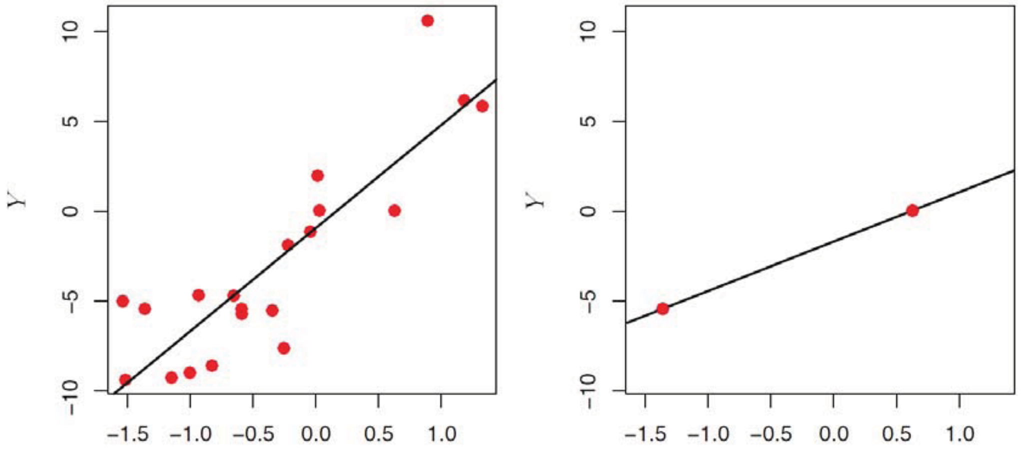
- Least squares method
- Measures for model fit in multiple linear regression
  - $RSE = \sqrt{\frac{RSS}{n-p-1}}$ (residual standard error)
  - $R^2 = \frac{TSS-RSS}{TSS}$ (the fraction of variance explained)
    - $TSS = \sum_{i=1}^{n}(y_i - \bar{y})^2$

High-Dimensional Sparse Data

- Low dimensional data
  - Predicting blood pressure based on age, gender, and body mass index
  - Data from thousands of people can be obtained
  - p << n
- High-dimensional sparse data
  - Blood pressure prediction using millions of single nucleotide polymorphisms (SNPs)
  - Data from thousands of people can be obtained
  - p > n
- Classical approaches such as the least squares is not appropriate for the high-dimensional cases

Least Squares Regression in a Low-Dimensional Setting

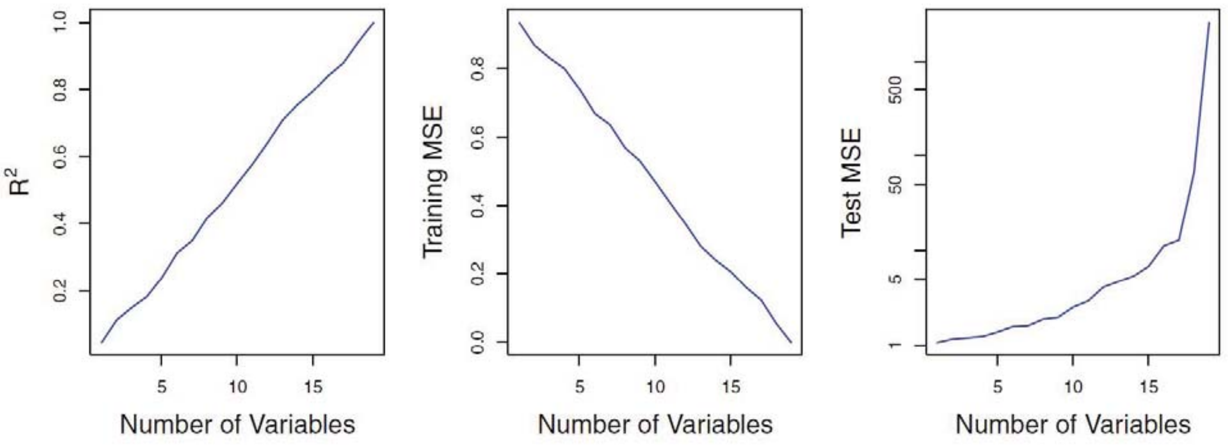
- p = 1; n = 20 vs n = 2



  - When n < p or $n \approx p$, the least squares is too flexible to prevent the overfitting

Impact of the Number of Predictors

- n = 20; p = 1 to 20
  - All the predictors were unrelated with the response



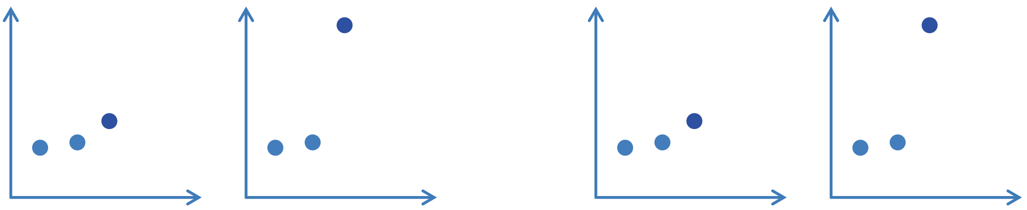
Shrinkage Methods

Linear Models for High-Dimensional Sparse Data

- Even linear models with the least squares are too flexible for some cases
  - If n > p: high variability → overfitting
  - If n < p: infinite variability → infinite models can fit the data
- Alternative fitting procedures than the (ordinary) least squares are required

Idea of the Shrinkage Method

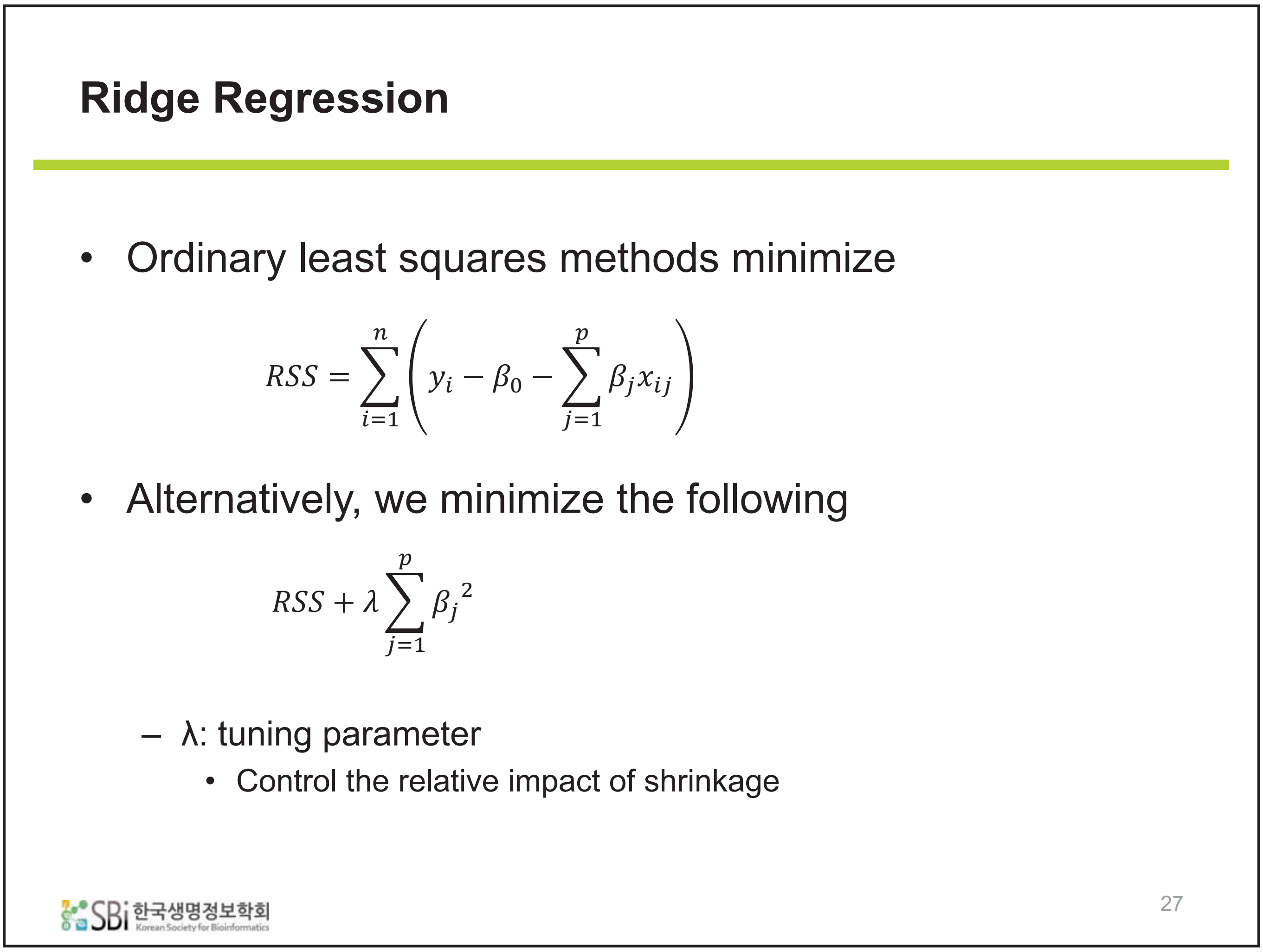
- Constrain or regularize the coefficient estimates
  - Shrink the coefficient estimates towards zero
- Shrinking the coefficient estimates could reduce their variance
  - Bias-variance trade-off



No shrinking

Shrinking

- Ridge
- Lasso



Ridge Regression

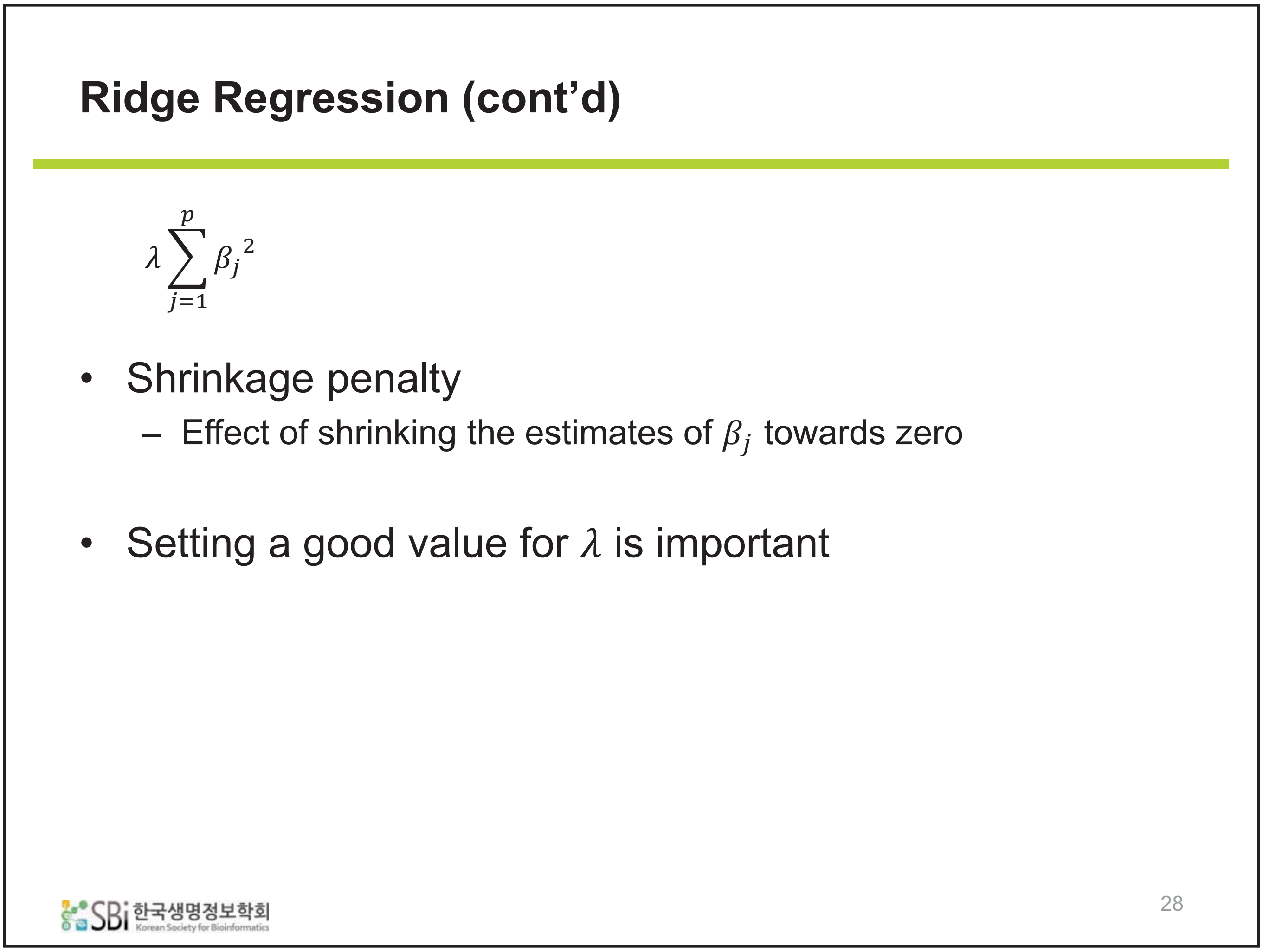
- Ordinary least squares methods minimize

$$RSS = \sum_{i=1}^{n}\left(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij}\right)$$

- Alternatively, we minimize the following

$$RSS + \lambda\sum_{j=1}^{p}{\beta_j}^2$$

  - λ: tuning parameter
    - Control the relative impact of shrinkage



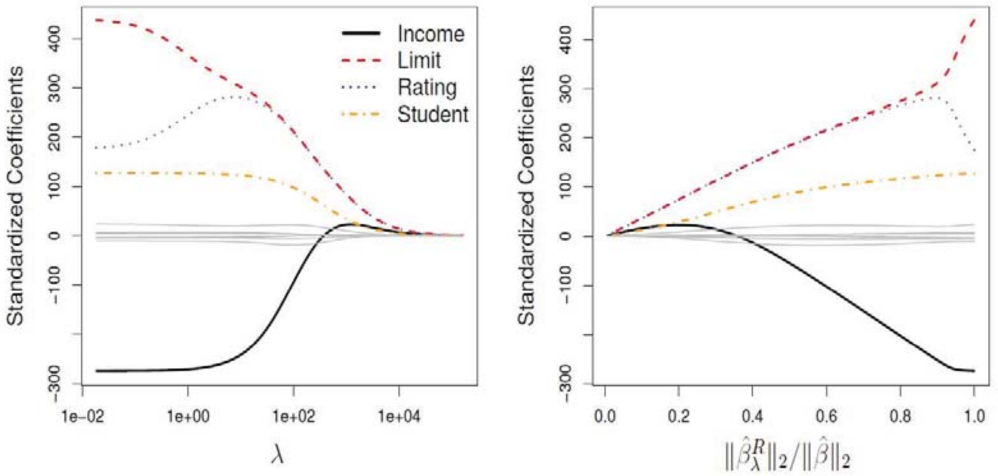
Ridge Regression (cont’d)

$$\lambda\sum_{j=1}^{p}{\beta_j}^2$$

- Shrinkage penalty
  - Effect of shrinking the estimates of β_j towards zero

- Setting a good value for λ is important

Effect of Ridge on Regression Coefficients

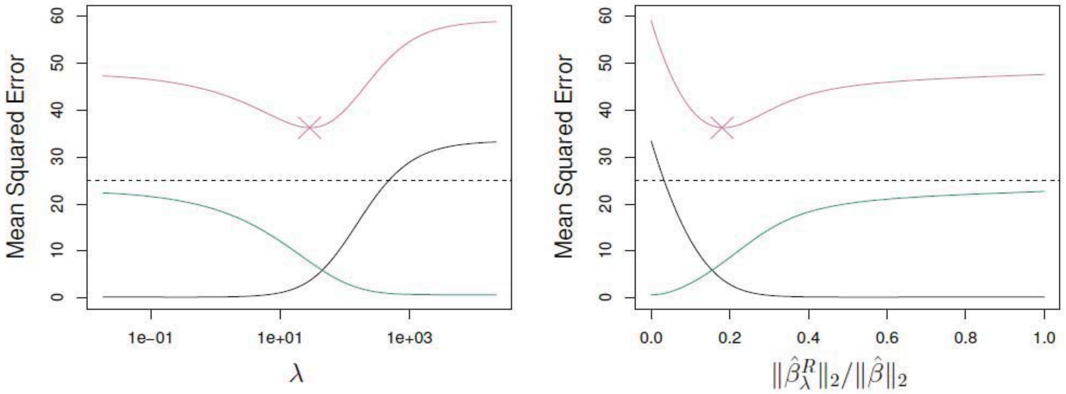


- Predict balance using ten predictors including income, limit, rating, and student
- Left-hand plot: λ as x-axis value
- Right-hand plot: $\frac{\|\hat{\beta}_\lambda{}^R\|_2}{\|\hat{\beta}\|_2}$ as x-axis value
  - l_2 norm

Effect of Ridge on Regression Coefficients (cont'd)

- Scale equivariant
  - Ordinary least square estimates
    - $X_j\hat{\beta}_j$ is invariant regardless of the scale of X_j
  - Ridge regression
    - Standardizing the predictors is needed (y-axis of the previous plot)
    - $\tilde{x}_{ij} = \frac{x_{ij}}{\sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_{ij}-\bar{x}_j)^2}}$

Bias-Variance Trade-Off in Ridge Regression



- Simulated data set (p = 45, n = 50)
  - Very sparse
- Squared bias, variance, and test MSE
- The variance decreases substantially without substantial increase in bias till $\lambda = \sim 10$ (left-hand plot)

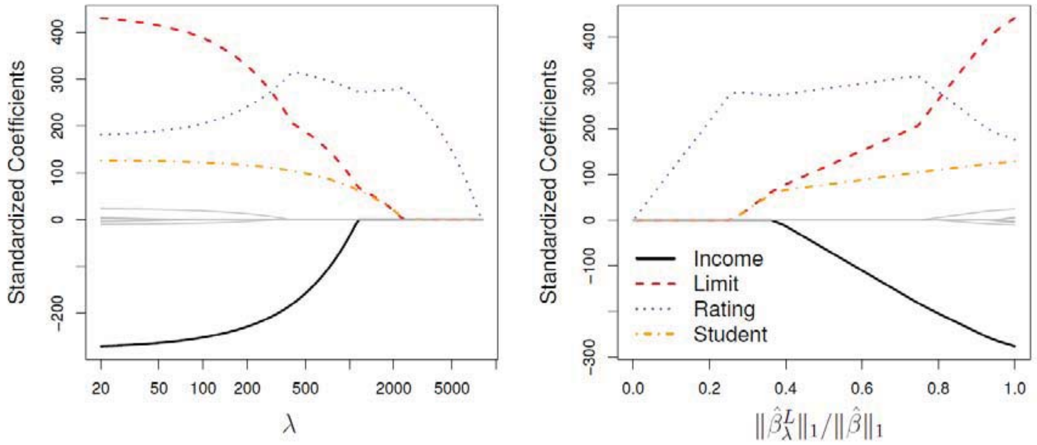
Advantages and Disadvantages of Ridge Regression

- Advantages
  - Ridge regression works very well in situations where the least squares method results in high variance
    - In many bioinformatics data sets, e.g., microarray analysis
  - Other benefits of ridge regression
    - Less computation is needed compared with other methods, e.g., best subset selection
- Disadvantages
  - All predictors are used unless $\lambda = \infty$
    - Can be problematic when interpreting the regression result (especially when p is large)
    - The subset selection approach could do this
    - Shrinkage methods for this?

Lasso

- Least Absolute Shrinkage and Selection Operator
- Objective function for lasso
  - $RSS + \lambda \sum_{j=1}^{p} |\beta_j|$
  - l_1 penalty
- In lasso, coefficient estimates for some predictors are exactly zero if λ is sufficiently large

Effect of Lasso on Regression Coefficients

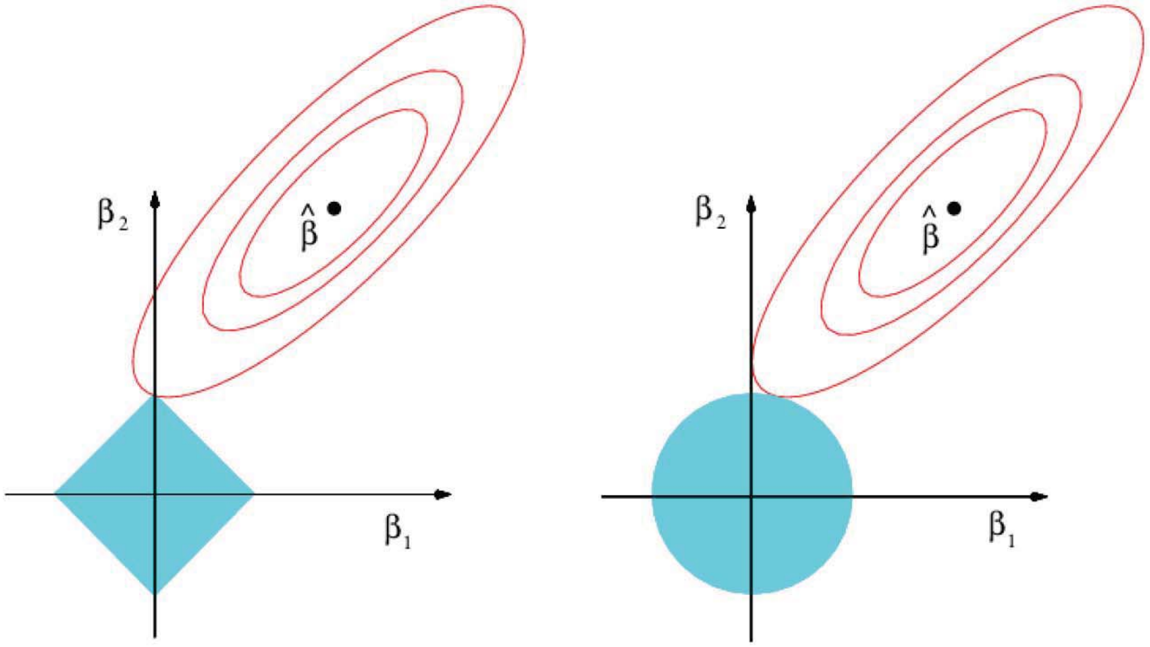


- When λ is very large (i.e., > 5,000), only one predictor (rating) is included.
- As λ decreases, student and limit are added
- Effect of predictor subset selection

Shrinkage Viewed as Constrained Optimization

- Ridge
  - $\min_{\beta}\left\{\sum_{i=1}^{n}\left(y_i-\beta_0-\sum_{j=1}^{p}\beta_j x_{ij}\right)^2\right\}$ subject to $\sum_{j=1}^{p}\beta_j{}^2\leq s$
- Lasso
  - $\min_{\beta}\left\{\sum_{i=1}^{n}\left(y_i-\beta_0-\sum_{j=1}^{p}\beta_j x_{ij}\right)^2\right\}$ subject to $\sum_{j=1}^{p}\left|\beta_j\right|\leq s$

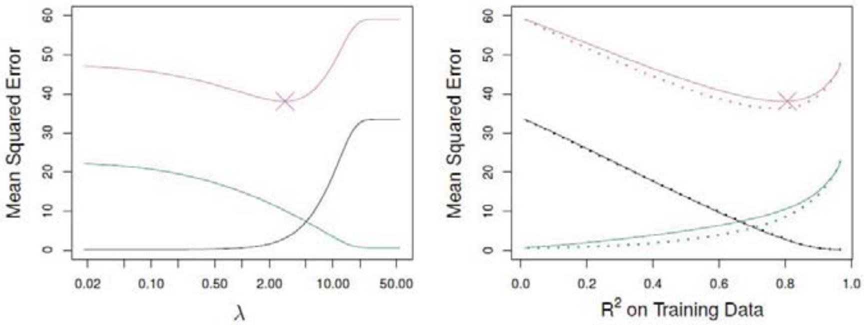
Comparison between Ridge and Lasso



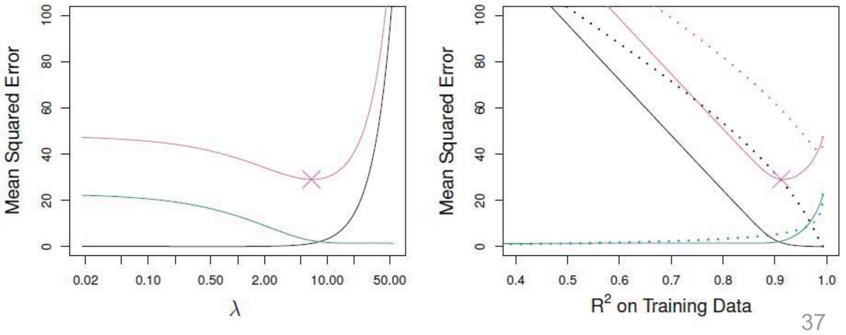
- Contours of the error and constraint functions for ridge (right) and lasso (left)

Results of Lasso on Simulated Data Sets (Compared with Ridge)

- 45 predictors

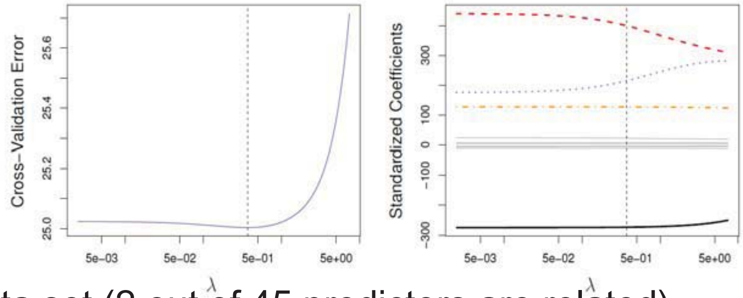


- Only 2 predictors out of the 45 were used for data generation

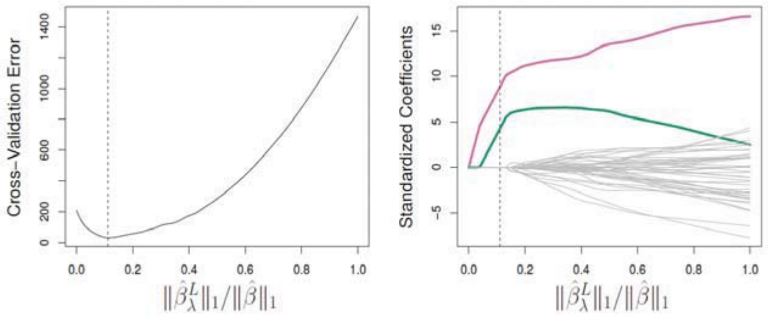


How to Determine the Value of λ for the Shrinkage Methods

- Cross-validation can be applied
  - The Credit data

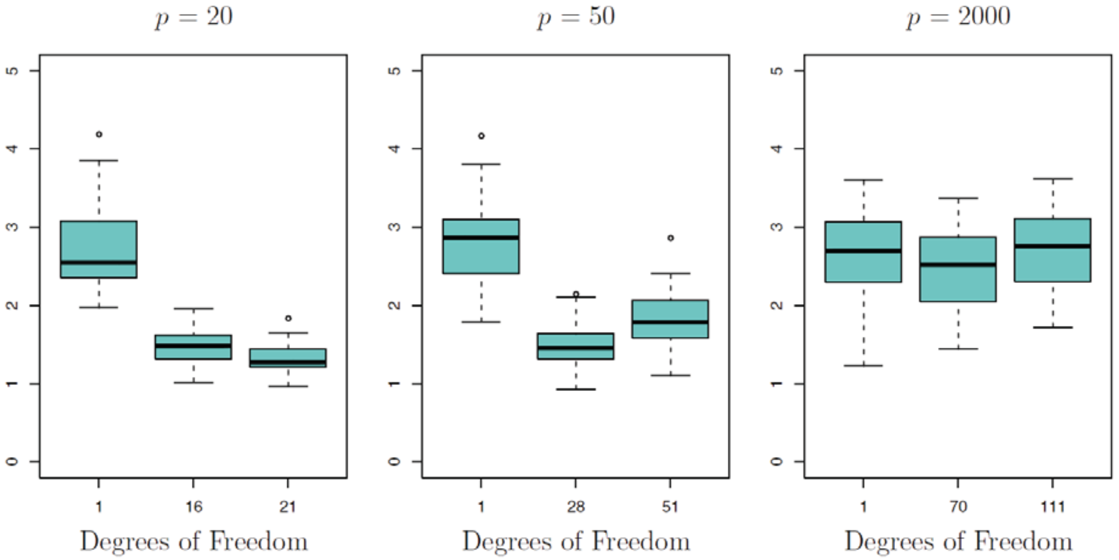


  - The simulated data set (2 out of 45 predictors are related)



Lasso on a High-Dimensional Data Set

- n = 100; p = 20, 50, 2000
  - Only 20 predictors were related with the response



  - Degrees of freedom: # of non-zero coefficients

Curse of Dimensionality

- Adding additional signal features will improve the fitted model
- Adding noise features will lead to a deterioration in the fitted model
- Thus, new technologies (or hypotheses) that allow for the collection of measurements for thousands/millions of features are a double-edged sword
  - Even if they are signal features, the variance incurred in fitting their coefficients may outweigh the reduction in bias

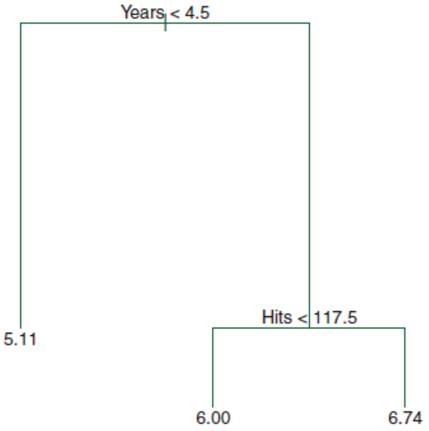
Tree Ensembles

Tree-Based Methods

- Decision tree methods
  - Stratifying or segmenting the predictor space into a set of simple regions
  - Use the mean or the mode of the training examples in the region
  - The splitting rules can be summarized as a tree
- A simple and useful method
  - Especially for interpretation
  - However, not competitive with the best supervised learning method in terms of prediction accuracy
  - Some techniques such as bagging, random forests, and boosting can be used for addressing the prediction accuracy problem

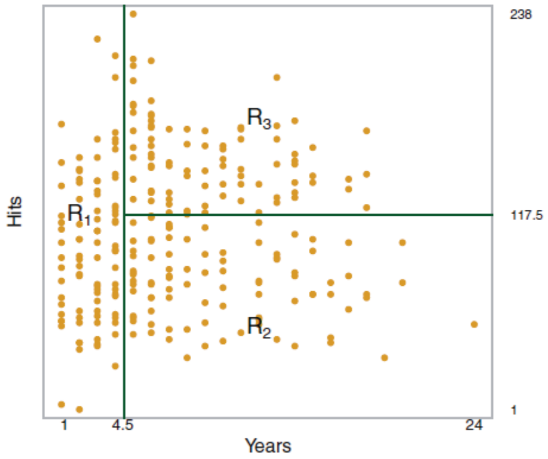
An Example Regression Tree

- Predict baseball players' salaries using regression trees
  - Response: Salary (in natural logarithm)
  - Predictors: Years and Hits
- A regression tree learned from the Hitters data set
  - An upside-down tree
  - Each internal node: a splitting rule
  - Each terminal (leaf) node: a region containing a set of examples
    - The number denotes the mean Salary value of the examples included

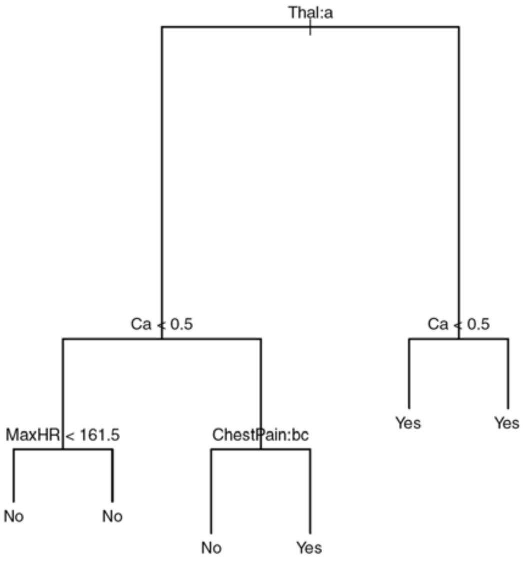


The Regions for the Hitters Data

- Three regions
  - R_1 = {X | Years < 4.5}
  - R_2 = {X | Years ≥ 4.5 and Hits < 117.5}
  - R_3 = {X | Years ≥ 4.5 and Hits ≥ 117.5}
- Interpretation of the tree
  - Years is the most important factor
  - If a player is less experienced, Hits does not play an important role
  - Otherwise, Hits matters



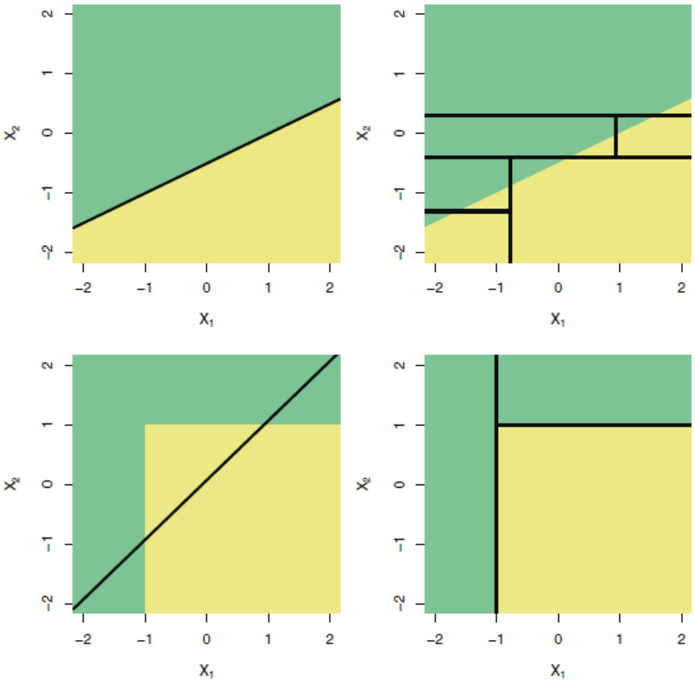
An Example Classification Tree



- Heart data set
  - A binary outcome for 303 patients having chest pain
  - Have heart disease or not

Trees vs Linear Models

- Depends on the problem at hand



Performance Improvement of Tree-Based Methods

- High variance in decision trees
  - If we randomly divide a data set into two and learn a decision tree from each of them, then the results would be quite different
  - Methods with low variance such as linear regression tends to have low variance (if n is much larger than p)
- Bootstrap aggregation (i.e., bagging) could reduce this problem

Averaging for Reducing Variances

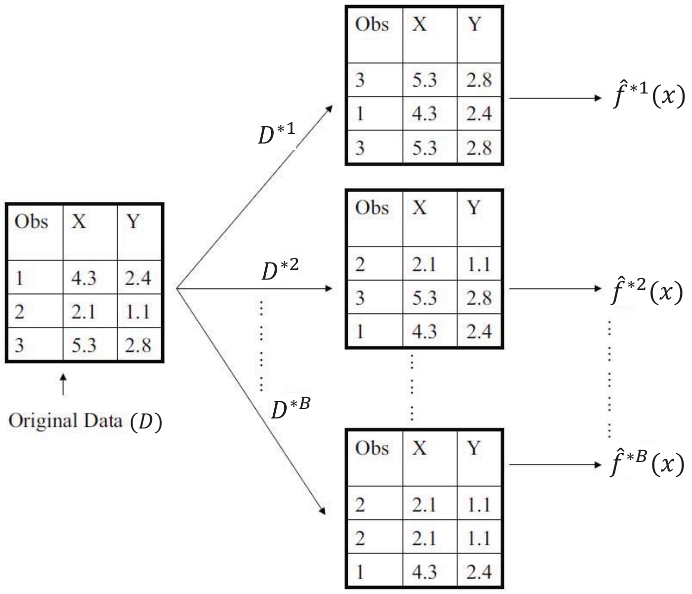
- Given a set of independent observations Z_1, Z_2, …, Z_n with a common variance σ^2
  - The variance of the mean $\bar{Z}$ is $\frac{\sigma^2}{n}$
- In a similar way, we could take B training data sets, build a model from each of them, and average the resulting B predictions
  - $\hat{f}^1(x), \hat{f}^2(x), \ldots, \hat{f}^B(x)$
  - $\hat{f}_{avg}(x) = \frac{1}{B}\sum_{b=1}^{B} \hat{f}^b(x)$
- Of course, the above procedure is not practical because we usually do not have multiple training data sets

Bagging

- We can use bootstrap for taking averages from a single data set
- Generate B bootstrapped training data sets (with replacement)
- Train a method using each of the bootstrapped training sets
- Average the predictions
  - $\hat{f}_{bag}(x) = \frac{1}{B}\sum_{b=1}^{B} \hat{f}^{*b}(x)$

Bagging (cont'd)

- A graphical representation of the bootstrap approach



Obs	X	Y
1	4.3	2.4
2	2.1	1.1
3	5.3	2.8

Original Data (D)

D^{*1}

Obs	X	Y
3	5.3	2.8
1	4.3	2.4
3	5.3	2.8

$\hat{f}^{*1}(x)$

D^{*2}

Obs	X	Y
2	2.1	1.1
3	5.3	2.8
1	4.3	2.4

$\hat{f}^{*2}(x)$

D^{*B}

Obs	X	Y
2	2.1	1.1
2	2.1	1.1
1	4.3	2.4

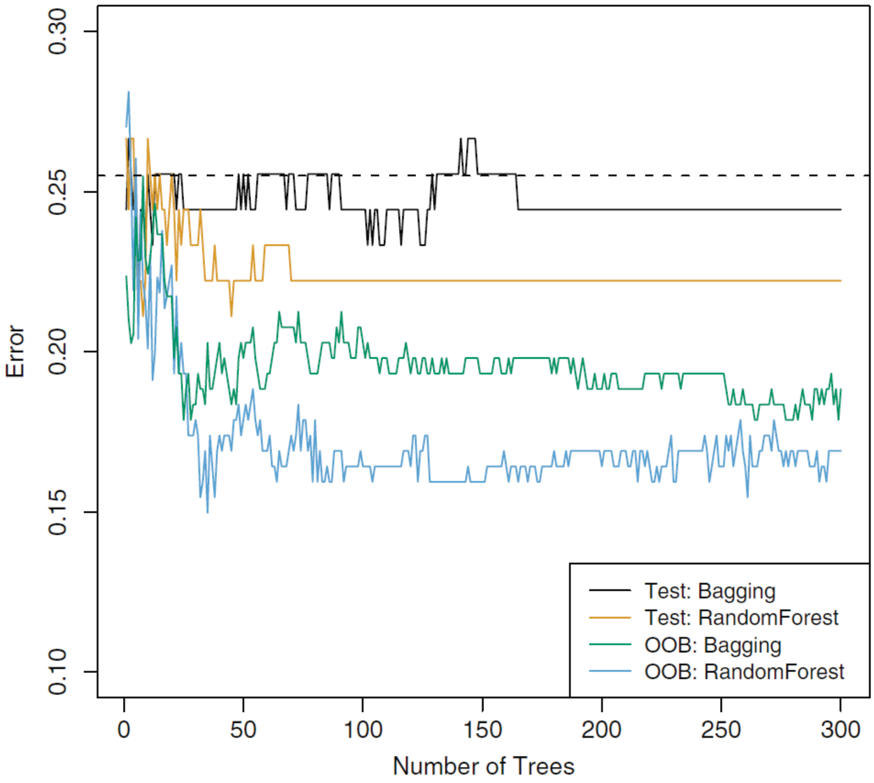
$\hat{f}^{*B}(x)$

Bagging (cont'd)

- Trees in bagging are grown deep and not pruned
  - Thus, each tree has low bias but high variance
  - Averaging these trees reduces the variance
- Bagging has been demonstrated to give impressive improvements by combining hundreds or thousands of individual trees
- Bagging on the Heart data set
  - Bagging with more than 100 trees could improve test accuracy
  - Test error was estimated using a validation set approach

SBi 한국생명정보학회 Korean Society for Bioinformatics

Performance of Bagging on the Heart Data Set

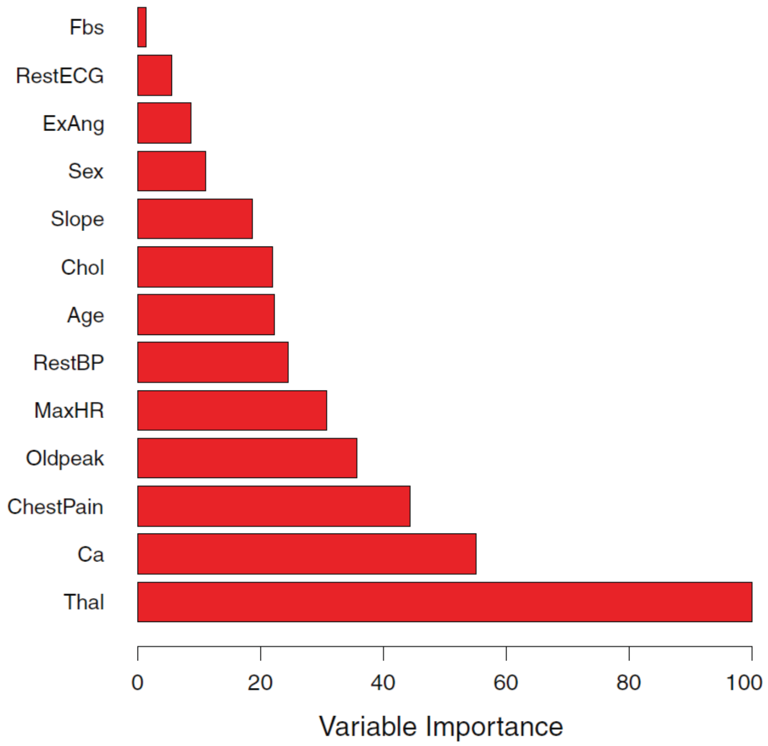


SBi 한국생명정

Variable Importance Measure

- Bagged trees are hard to interpret
  - Bagging improves the prediction accuracy at the expense of interpretability
- Instead, we can aggregate the importance of each predictor in each tree
  - A large value denotes a high importance

Importance of Variables in the Heart Data Set



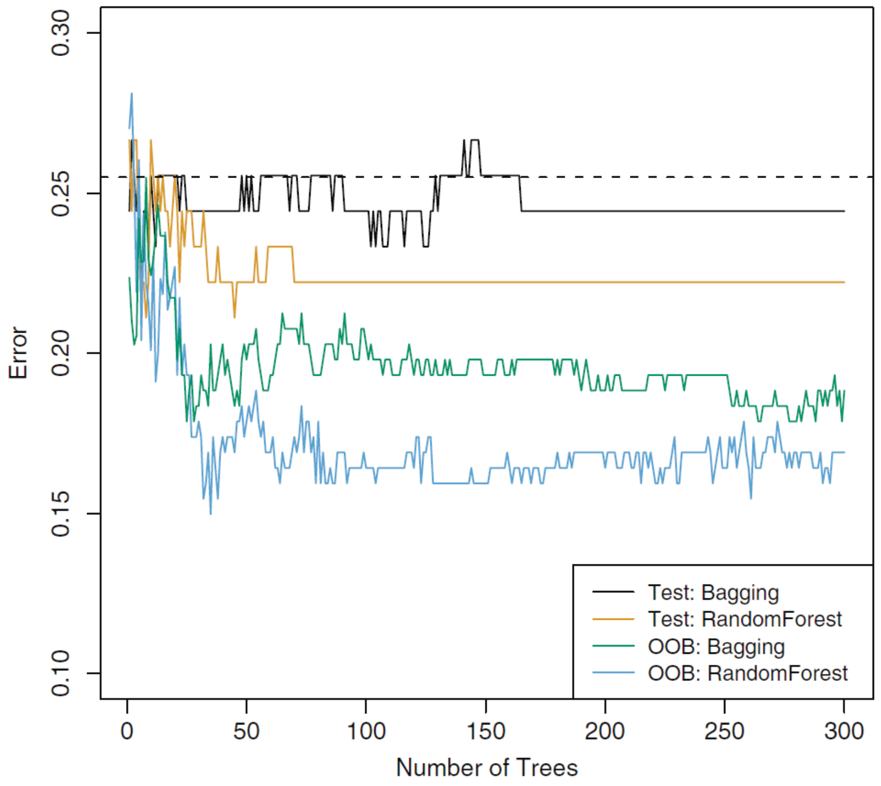
Correlation between Trees

- If there is one very strong predictor in a data set, that predictor will be always included in the bagged decision trees
  - Moreover, most of the trees will use that predictor on top of the splits
  - Thus, all the bagged trees will look quite similar to one another, resulting in a high correlation among them
- Averaging high correlated variables usually does not lead to a large reduction of variance
  - Test error of bagging would be large
- Thus, it is important to *"decorrelate"* the bagged trees

Random Forests

- Idea for decorrelating the trees
  - At each iteration of tree building, a random sample of m predictors are considered instead of all p predictors
  - This, we hope that the set of strong predictors would not be chosen in some cases
  - Usually $m = \sqrt{p}$ is used for classification ($p/3$ for regression)
- By decorrelating the trees, the reduction of variance would be substantial
- Random forests applied to the Heart data set

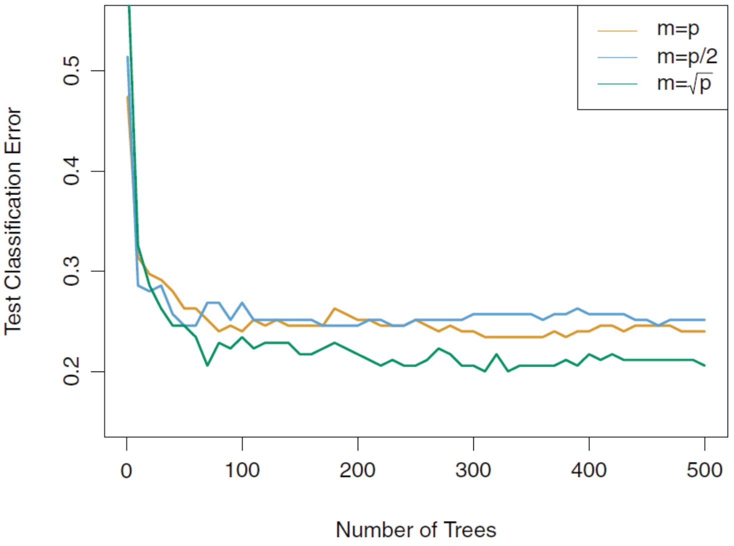
Performance of Random Forests on the Heart Data Set



Random Forests for a Gene Expression Data Set

- A gene expression data set
  - 4,718 genes
  - 349 patients
  - 15 class labels: normal and 14 different types of cancer
- 500 genes with high variance were selected
  - 349 x 500 data matrix (*very sparse!!*)

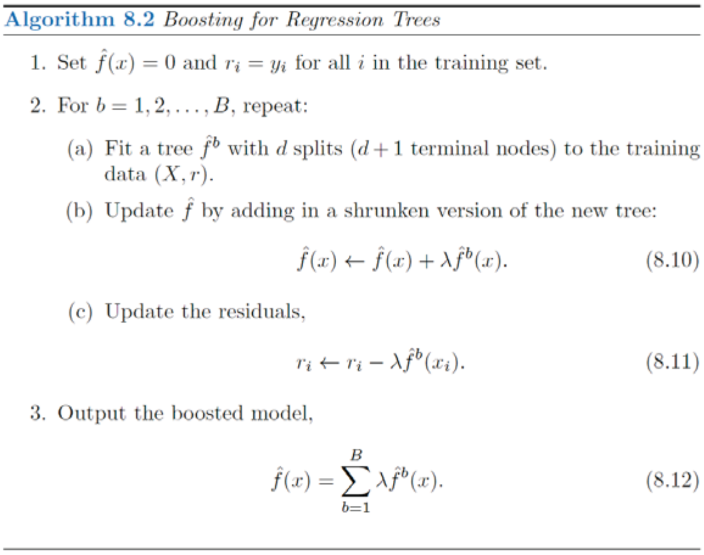
Performance of Random Forests on the Gene Expression Data Set



- A validation set approach was used
- Test error rate of a single tree: 0.457
- Random forests performed well

Boosting

- Another method for prediction performance improvement
- Trees are grown sequentially
  - Each tree is grown using information from previously grown trees
  - Each tree is fit on a modified version of the original data set



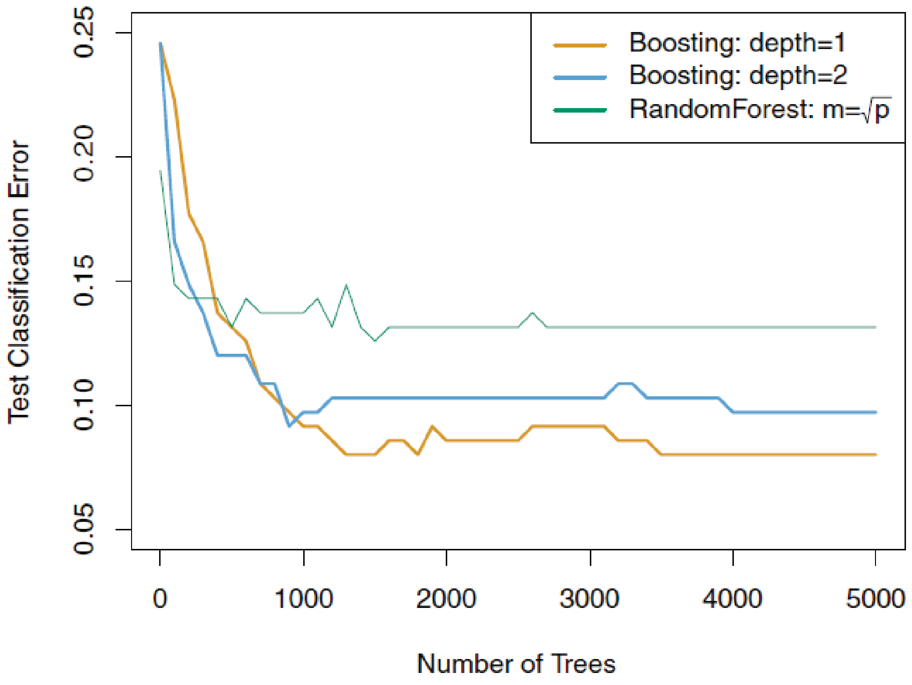
Algorithm 8.2 *Boosting for Regression Trees*

```
1. Set $\hat{f}(x) = 0$ and $r_i = y_i$ for all $i$ in the training set.
2. For $b = 1, 2, \ldots, B$, repeat:
    (a) Fit a tree $\hat{f}^b$ with $d$ splits ($d+1$ terminal nodes) to the training data $(X, r)$.
    (b) Update $\hat{f}$ by adding in a shrunken version of the new tree:
            $\hat{f}(x) \leftarrow \hat{f}(x) + \lambda \hat{f}^b(x).$        (8.10)
    (c) Update the residuals,
            $r_i \leftarrow r_i - \lambda \hat{f}^b(x_i).$        (8.11)
3. Output the boosted model,
            $\hat{f}(x) = \sum_{b=1}^{B} \lambda \hat{f}^b(x).$        (8.12)
```

Parameters for Boosting

- Number of trees B
  - A large B values could result in overfitting
  - CV is used to select B
- Shrinkage parameter λ
  - A small positive number such as 0.01 and 0.001
- Number d of splits in each tree
  - Controls the complexity of each tree
  - Often d = 1 works well in practice (a.k.a. decision stumps)

Comparison between Boosting and Random Forests (the Gene Expression Data Set: Cancer vs Normal)



마치면서

- 학습 오류와 테스트 오류는 불일치할 수 있다
  - 과대적합
- 테스트 오류는 편향과 분산으로 구성된다
  - 모델의 복잡도가 크고 주어진 데이터의 크기가 작은 경우 분산이 커질 수 있다
- 고차원 희박 데이터의 경우 복잡도가 낮은 선형 모델도 분산이 클 수 있다
  - Shrinkage 방법은 이러한 문제를 완화할 수 있다
- Tree 기반 방법은 결과의 해석이 용이한 장점이 있지만 예측 성능은 다른 기계학습 방법에 비해 떨어진다
  - 성능을 향상시키는 방법으로 tree ensemble이 주로 적용된다
  - 고차원 희박 데이터에도 잘 적용될 수 있다

Acknowledgement

- Some of the figures in this presentation are taken from "An Introduction to Statistical Learning, with applications in R" (Springer, 2013) with permission from the authors: G. James, D. Witten, T. Hastie and, R. Tibshirani.