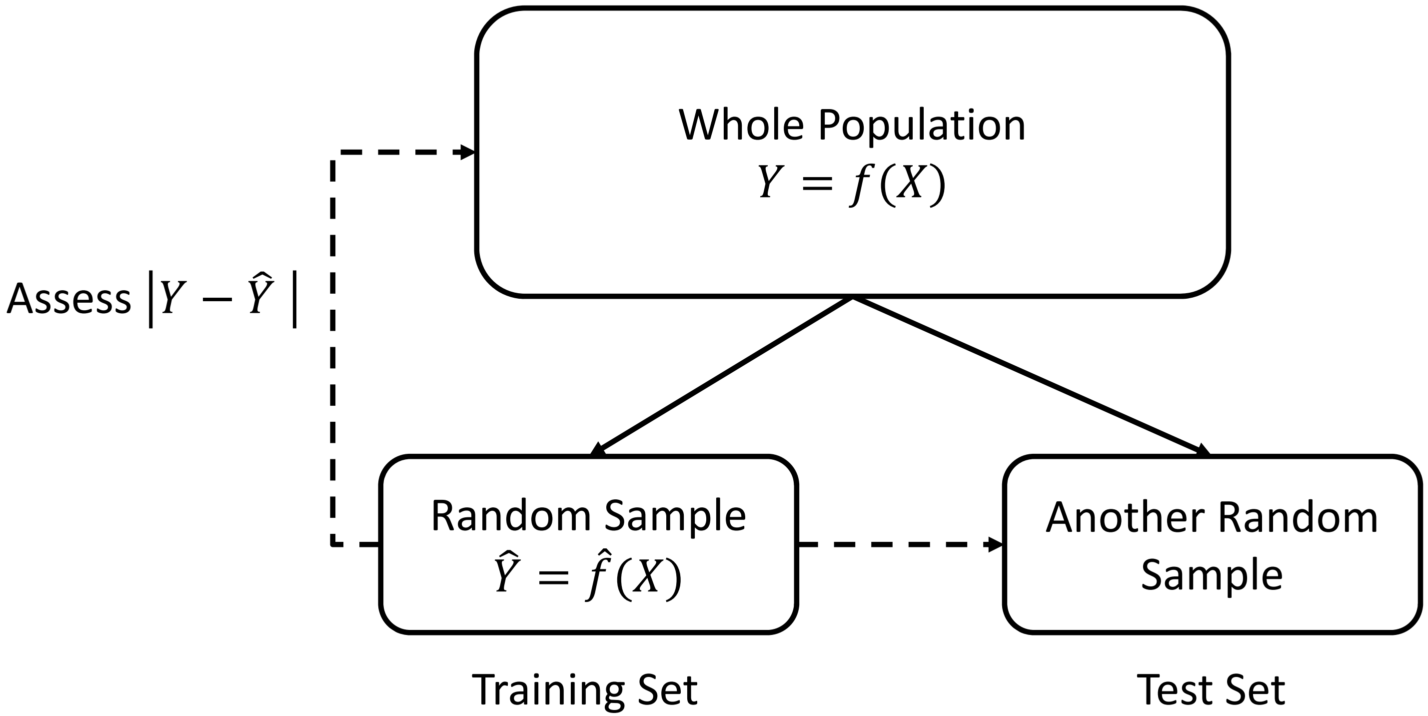
AI 양성 과정
- Model Selection -

Junhee Seok, Ph.D.
Associate Professor
School of Electrical Engineering
Korea University, Seoul, Korea

Cross-Validation

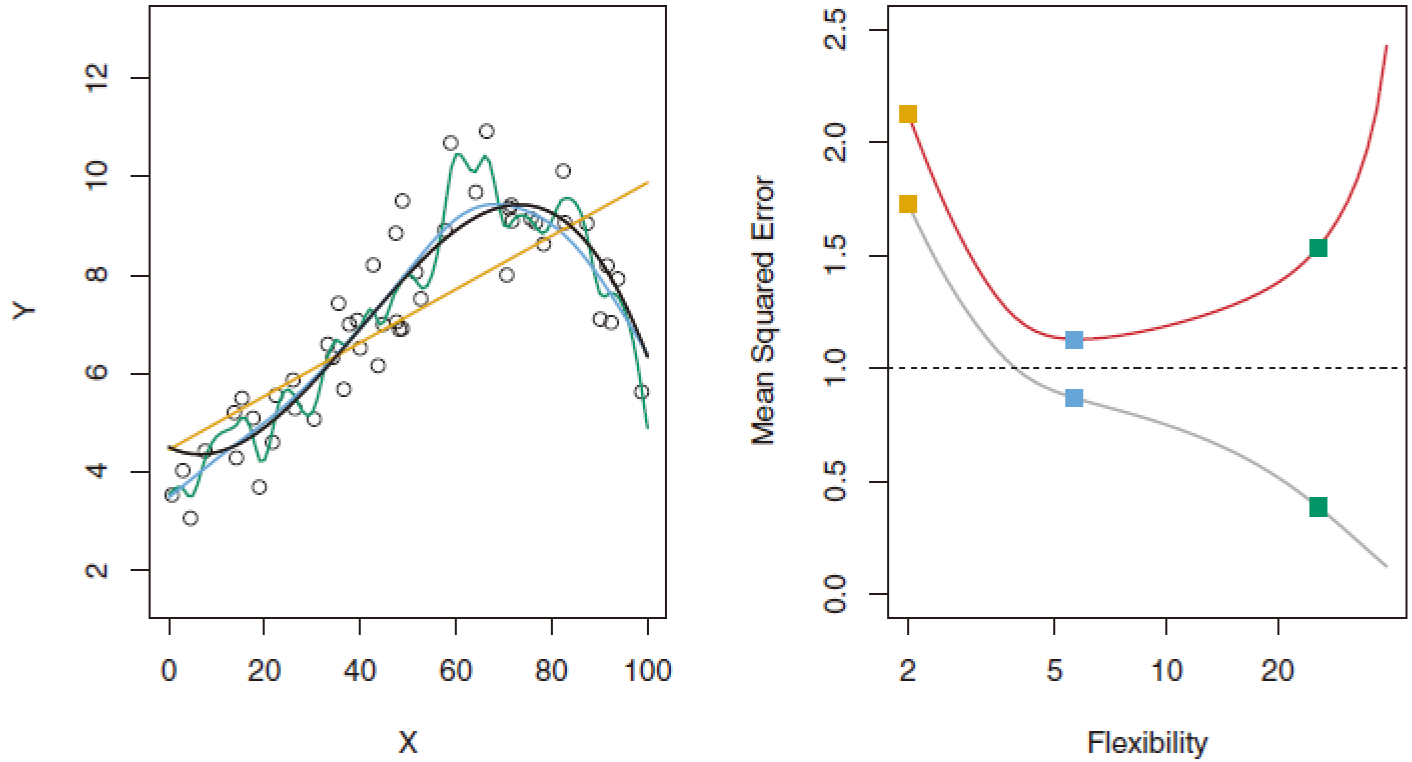
The Goal of Statistical Learning

- Estimating true $f(X)$ from random samples.



The Goal of Statistical Learning

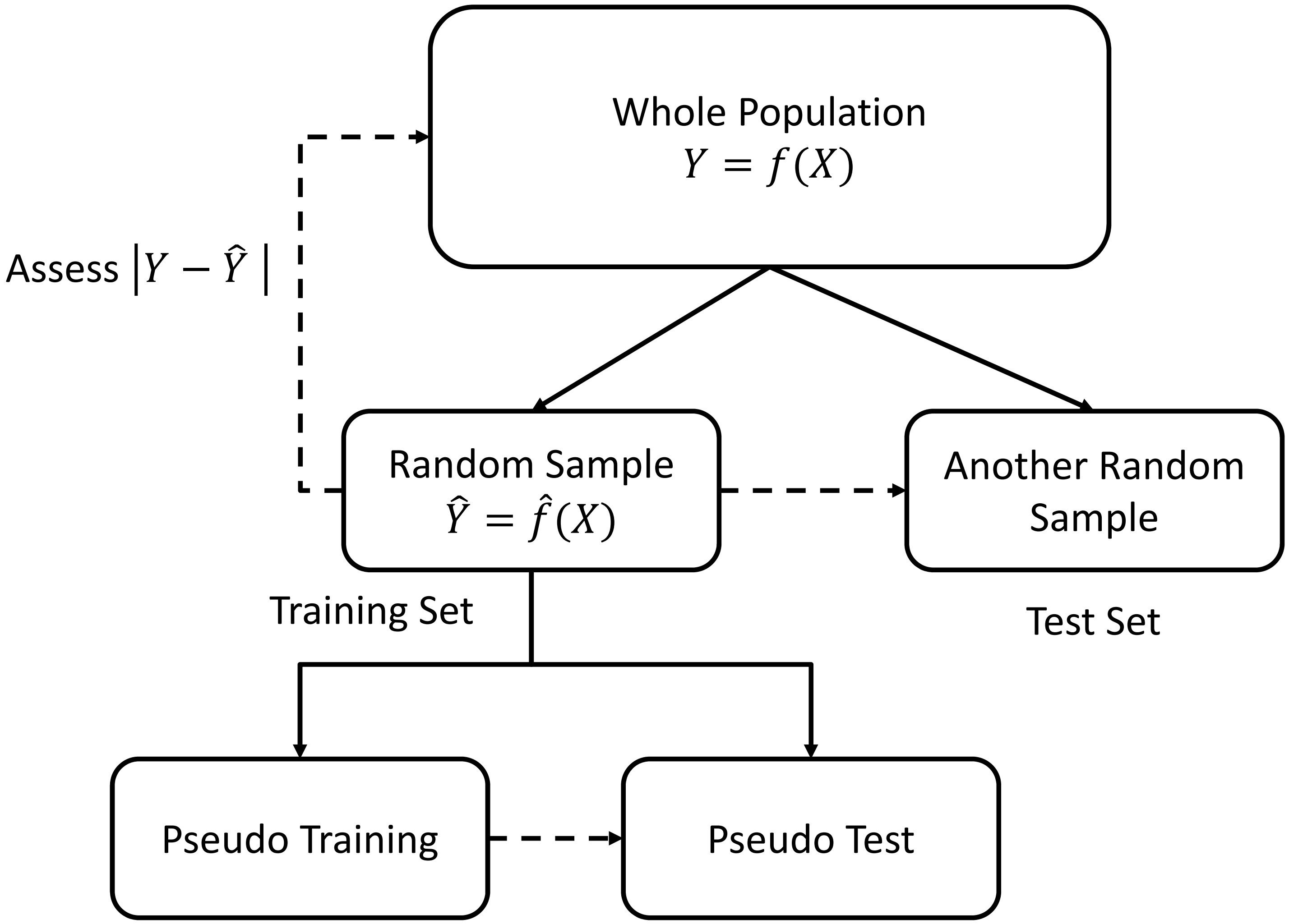
- Training errors and test errors are different.



- We know nothing about the test set when we estimate *f*(*X*).
- **How to estimate the test errors without looking at the test set?**

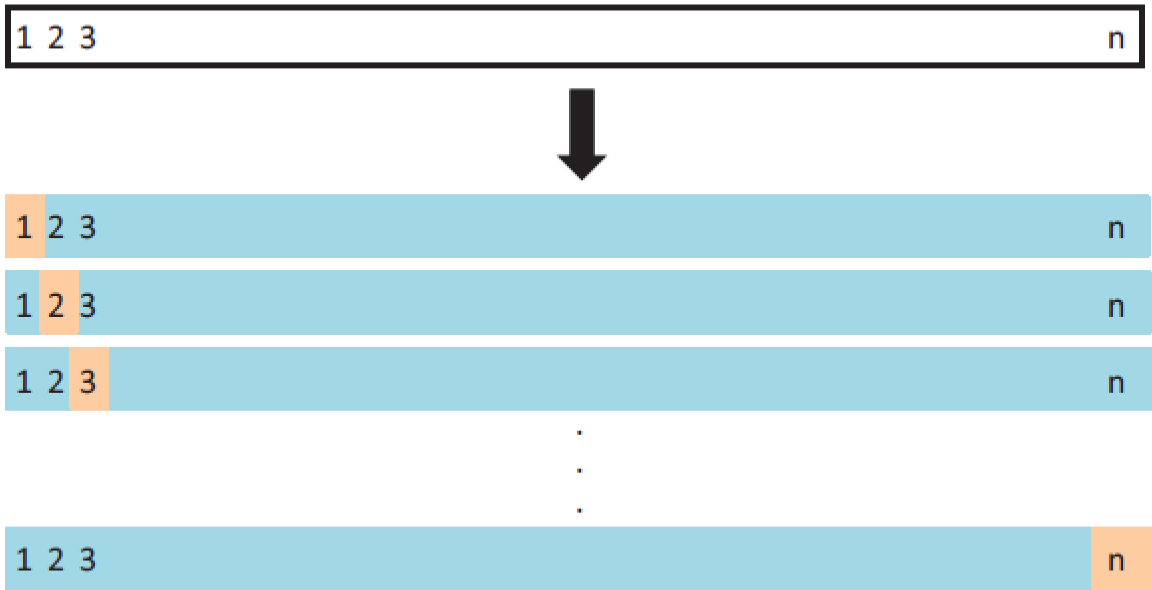
Cross-Validation

- Simulating training-test sets within the real training set.



Leave-One-Out Cross Validation (LOOCV)

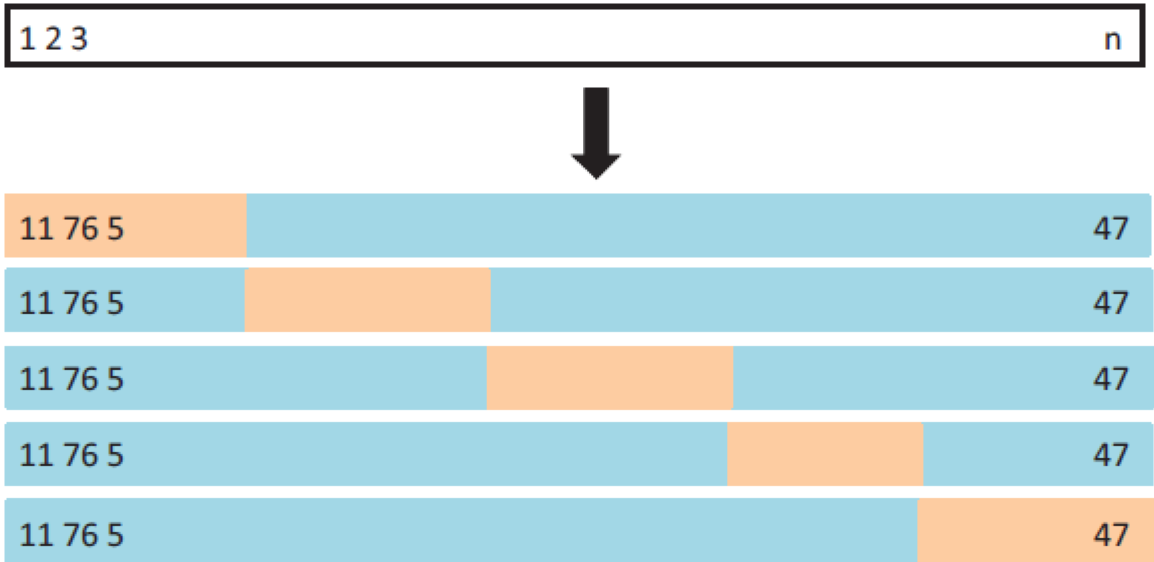
- Consider one sample (x_j, y_j) as a test set, and the rest (x_i, y_i) $i \neq j$ as a training set. Repeat it for all samples.



$$\mathrm{CV}_{(n)} = \frac{1}{n}\sum_{i=1}^{n} \mathrm{MSE}_i = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

k-Fold Cross Validation

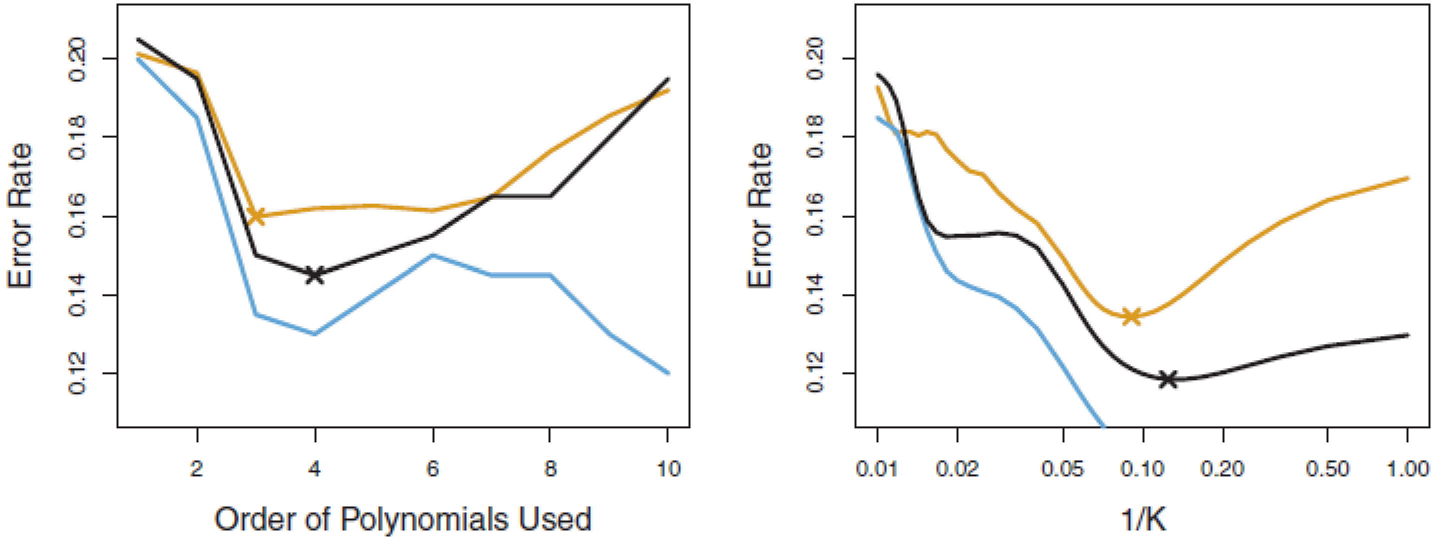
- Randomly divide the whole training data into *k* bins. Consider one bin is a test set and the rest bins are a training set. Repeat it for all *k* bins.
  - LOOCV is *n*-fold CV.



$$\mathrm{CV}_{(k)} = \frac{1}{k}\sum_{i=1}^{k}\mathrm{MSE}_i.$$

Examples

- Classification



  – Orange: true test error; Blue: training error; Black: CV error.

- CV error reflects the pattern of the true test errors well
  – **Useful for model assessment.**

LOO vs. *k*-Fold Cross Validation

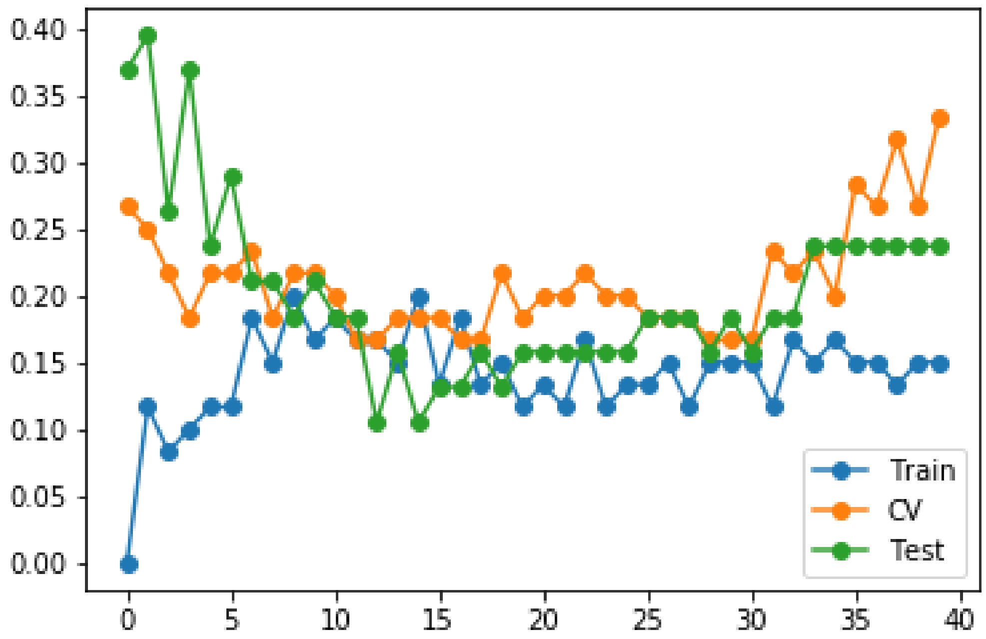
- LOOCV
  - Almost unbiased estimation for the true test errors because it uses n–1 samples for training: less bias.
  - The n fitted models are similar to each other and highly stick to the training data: high variance.
  - Computationally intensive: n model fittings.

- k-Fold CV (extremely k=2)
  - Underestimated the true test errors because it uses n/2 samples: high bias.
  - The k fitted models can be different and less stick to the original training set: low variance.
  - Computationally less intensive: k model fitting.

Practices

- Cross-validation
  - sklearn.model_selection.LeaveOneOut
  - sklearn.model_selection.Kfold
  - sklearn.model_selection.cross_val_score

- Practice
  - Read ‘data07_iris.cvs’ and plot train, cv, and test errors using KNN method by changing K from 1 to 40



Feature Selection

Best Subset Selection

- Selecting *k* best predictors among *p* predictors.
  - "Best" often means the lowest MSE.

- Algorithm

Algorithm 6.1 *Best subset selection*

---

1. Let $\mathcal{M}_0$ denote the *null model*, which contains no predictors. This model simply predicts the sample mean for each observation.

2. For $k = 1, 2, \ldots p$:

   (a) Fit all $\binom{p}{k}$ models that contain exactly k predictors.

   (b) Pick the best among these $\binom{p}{k}$ models, and call it $\mathcal{M}_k$. Here *best* is defined as having the smallest RSS, or equivalently largest R^2.

3. Select a single best model from among $\mathcal{M}_0, \ldots, \mathcal{M}_p$ using cross-validated prediction error, C_p (AIC), BIC, or adjusted R^2.

  - We will see later about AIC, BIC and adjusted R^2.

Stepwise Selection

- **Forward stepwise selection**: starting from a null model, and adding the best variables one-by-one.
  - In total, $1+p(p+1)/2$ models are fitted.

Algorithm 6.2 *Forward stepwise selection*

1. Let $\mathcal{M}_0$ denote the *null* model, which contains no predictors.
2. For $k = 0, \ldots, p-1$:
   (a) Consider all $p-k$ models that augment the predictors in $\mathcal{M}_k$ with one additional predictor.
   (b) Choose the *best* among these $p-k$ models, and call it $\mathcal{M}_{k+1}$. Here *best* is defined as having smallest RSS or highest R^2.
3. Select a single best model from among $\mathcal{M}_0, \ldots, \mathcal{M}_p$ using cross-validated prediction error, C_p (AIC), BIC, or adjusted R^2.

- Example

# Variables	Best subset	Forward stepwise
One	`rating`	`rating`
Two	`rating`, `income`	`rating`, `income`
Three	`rating`, `income`, `student`	`rating`, `income`, `student`
Four	`cards`, `income`, `student`, `limit`	`rating`, `income`, `student`, `limit`

Stepwise Selection

- **Backward stepwise selection**: starting from a full model, and removing the worst variables one-by-one.
  - In total, $1+p(p+1)/2$ models are fitted.

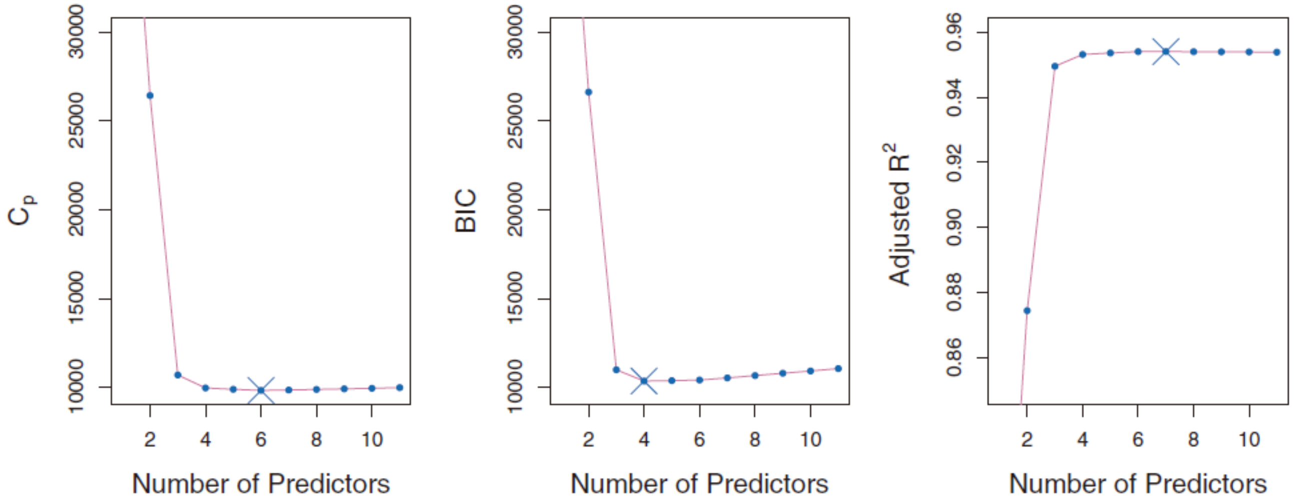
Algorithm 6.3 *Backward stepwise selection*

1. Let $\mathcal{M}_p$ denote the *full* model, which contains all p predictors.
2. For $k = p, p-1, \ldots, 1$:
   (a) Consider all k models that contain all but one of the predictors in $\mathcal{M}_k$, for a total of $k-1$ predictors.
   (b) Choose the *best* among these k models, and call it $\mathcal{M}_{k-1}$. Here *best* is defined as having smallest RSS or highest R^2.
3. Select a single best model from among $\mathcal{M}_0, \ldots, \mathcal{M}_p$ using cross-validated prediction error, C_p (AIC), BIC, or adjusted R^2.

- Forward stepwise selection if $p > n$.
- Backward stepwise selection if $n > p$.

Model Selection Criteria

- More predictors always decrease the training error.
- Adjusting the training error by accounting the number of predictors.



$$C_p = \frac{1}{n}\left(\text{RSS} + 2d\hat{\sigma}^2\right),$$

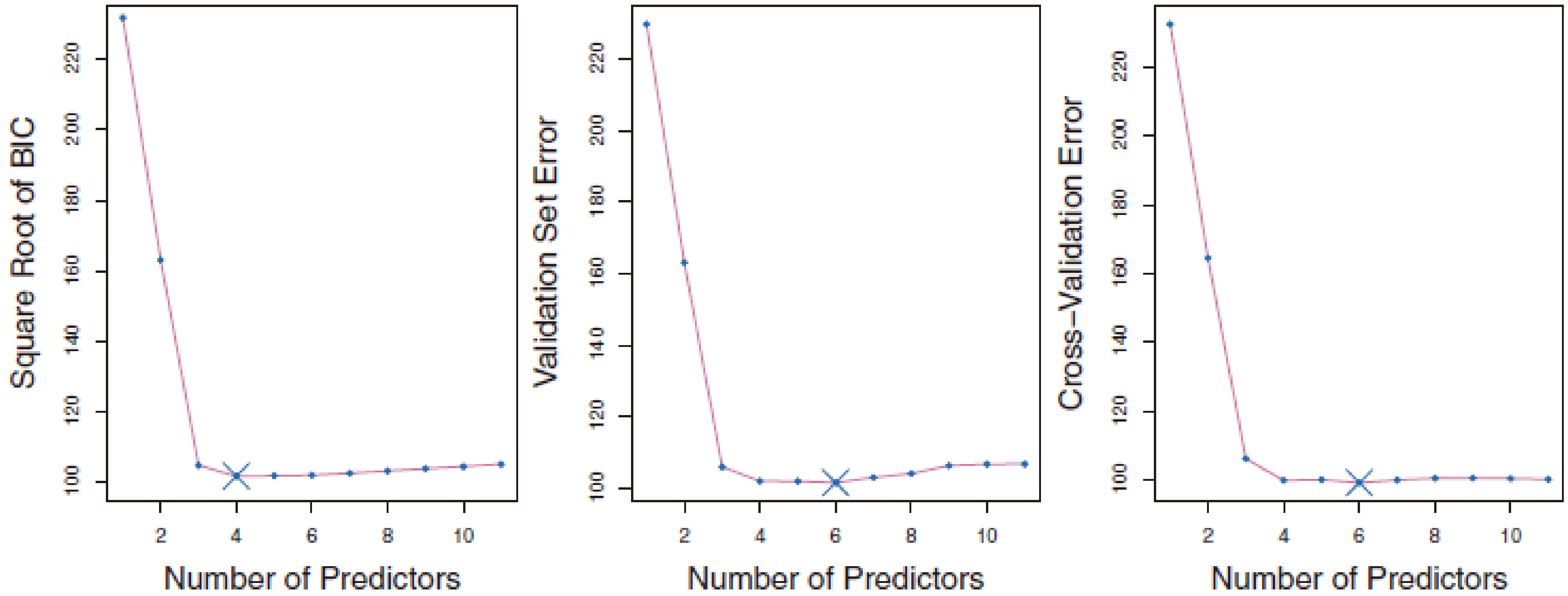
$$\text{AIC} = \frac{1}{n\hat{\sigma}^2}\left(\text{RSS} + 2d\hat{\sigma}^2\right),$$

$$\text{BIC} = \frac{1}{n}\left(\text{RSS} + \log(n)d\hat{\sigma}^2\right).$$

$$\text{Adjusted } R^2 = 1 - \frac{\text{RSS}/(n-d-1)}{\text{TSS}/(n-1)}.$$

Model Selection Criteria

- The adjustment methods simply assess the test errors based on many assumptions.
- Benefiting from high computation power, **cross-validation** assess the test error with less assumptions.



Regularization

Subset Selection vs. Shrinkage Methods

- In a linear regression model,

$$Y \approx \beta_0 + \beta_1 X_1 + \cdots \beta_p X_p$$

- Subset selection makes some selected β's zero.

- Shrinkage reducing all β's towards zeros.

- **Ridge regression**
- **Lasso**

Ridge Regression

- A usual least squares fitting finds β's that minimize

$$\text{RSS} = \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 .$$

- **Ridge regression** finds β's that minimizes

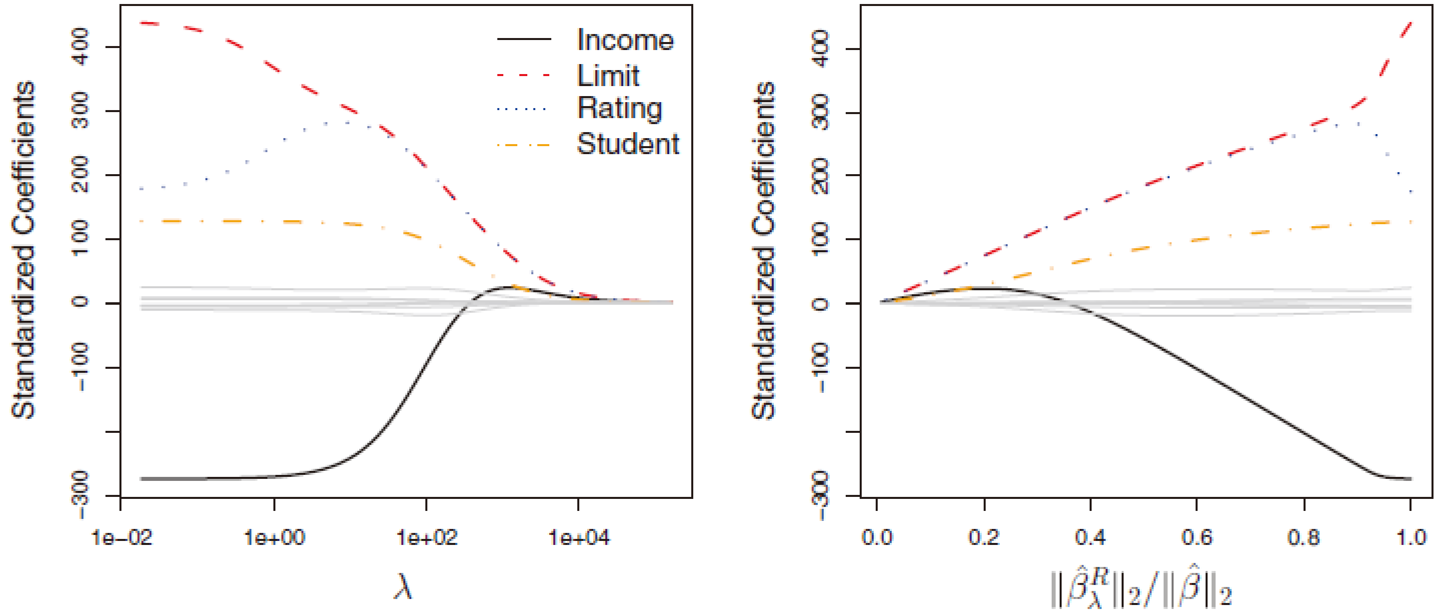
Shrinkage penalty

$$\sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 = \text{RSS} + \lambda \sum_{j=1}^{p} \beta_j^2,$$

- λ is a <u>tuning parameter</u> that determines the amount of shrinkage.
  - Determined separately, often from cross-validation.

Ridge Regression

- Example



L2 norm of β's
with regularization

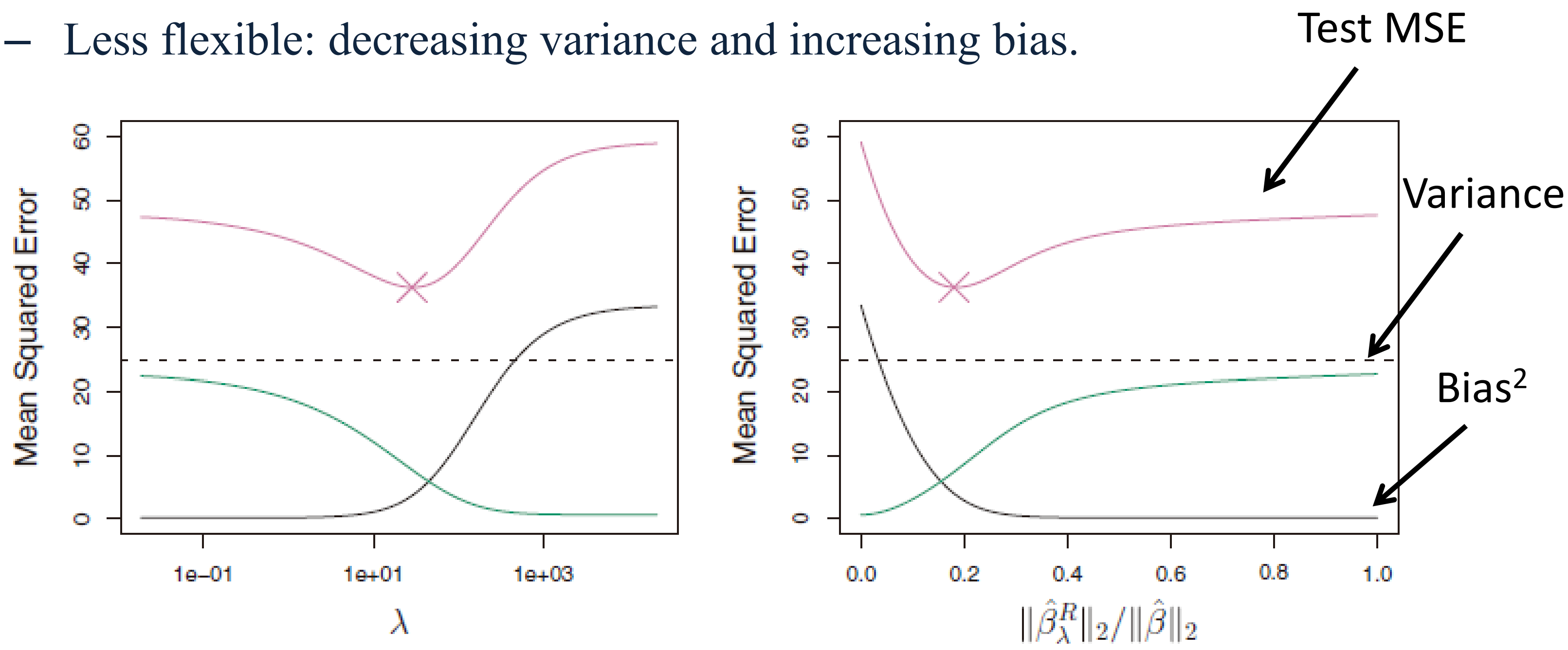
L2 norm of β's
without regularization

- Recommended to be standardized because shrinkage is affected by the sizes of predictors

$$\tilde{x}_{ij} = \frac{x_{ij}}{\sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_{ij} - \overline{x}_j)^2}},$$

Ridge Regression

- Bias-variance tradeoff
  - Ridge regression regulates the variability of coefficients.
  - Less flexible: decreasing variance and increasing bias.



- Computationally easy
  - It has an analytic solution: $\hat{\boldsymbol{\beta}} = \left(\mathbf{X}^T\mathbf{X} + \boldsymbol{\lambda}^2\mathbf{I}\right)^{-1}\mathbf{X}^T\mathbf{y}$.
  - It can be solved by one common matrix inversion for any λ.

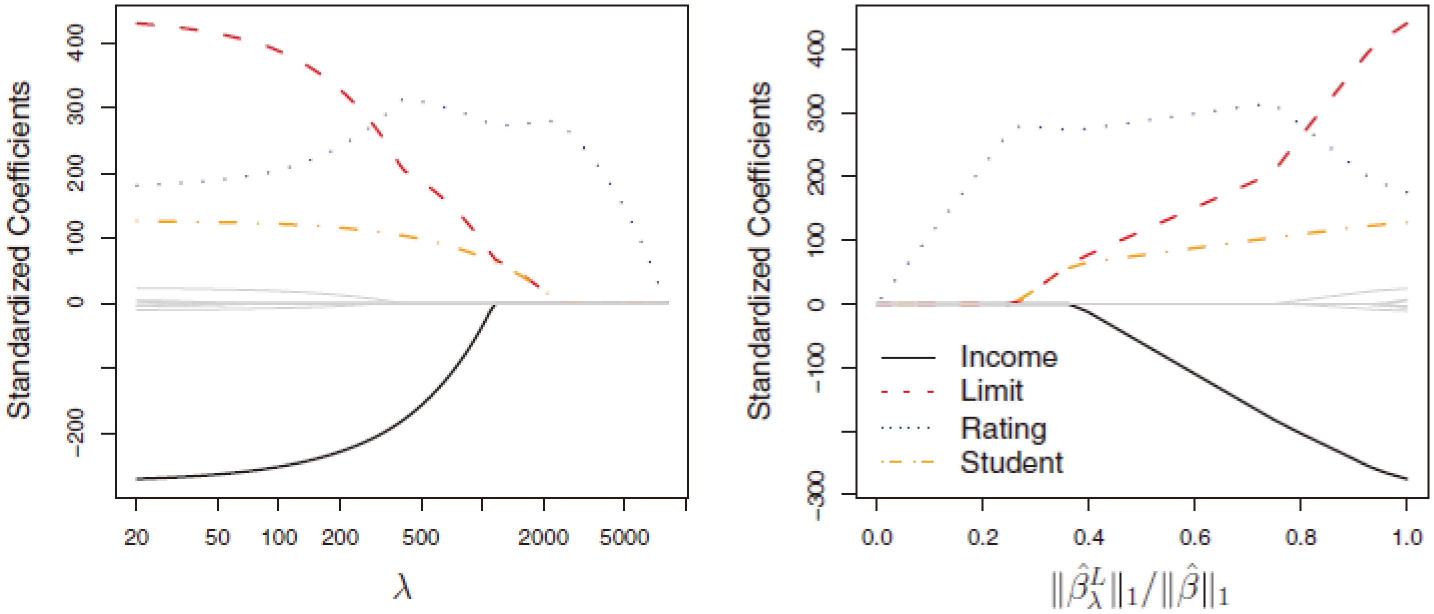
Lasso

- **Lasso** finds β's that minimizes

$$\sum_{i=1}^{n}\left(y_i-\beta_0-\sum_{j=1}^{p}\beta_j x_{ij}\right)^2+\lambda\sum_{j=1}^{p}|\beta_j| \;=\mathrm{RSS}+\lambda\sum_{j=1}^{p}|\beta_j|.$$

  – Lasso selects variables because it makes β's zero (ridge regression doesn't).

- Example



Ridge Regression vs. Lasso

- They are all the same family of convex optimization problem.

- Ridge regression

$$\underset{\beta}{\text{minimize}} \left\{ \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \right\} \quad \text{subject to} \quad \sum_{j=1}^{p} \beta_j^2 \leq s,$$

- Lasso

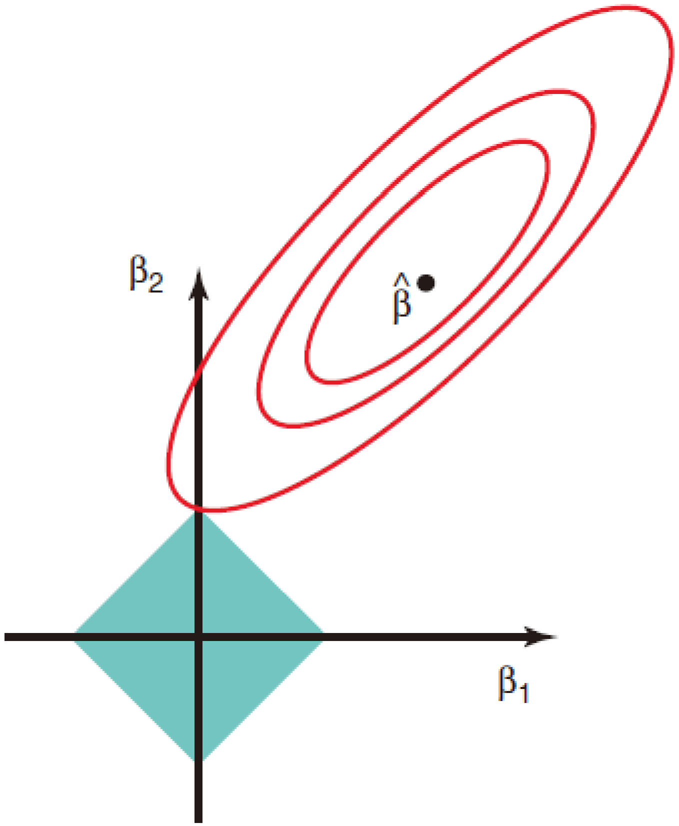
$$\underset{\beta}{\text{minimize}} \left\{ \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \right\} \quad \text{subject to} \quad \sum_{j=1}^{p} |\beta_j| \leq s$$

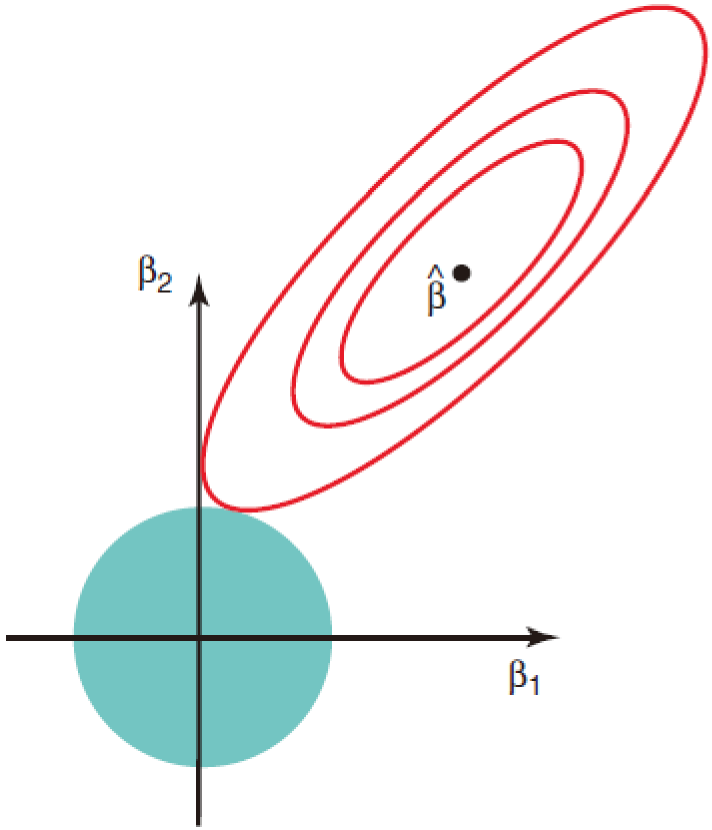
- Subset selection

$$\underset{\beta}{\text{minimize}} \left\{ \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \right\} \quad \text{subject to} \quad \sum_{j=1}^{p} I(\beta_j \neq 0) \leq s.$$

Ridge Regression vs. Lasso

- Lasso selects variables while ridge regression doesn't.
  - The RSS contour often hits the corner of the Lasso regularization area.





Elastic Net

- Ridge

$$\beta^* = \text{argmin}\left(RSS + \lambda \sum \beta_i^2 \right)$$

- Lasso

$$\beta^* = \text{argmin}\left(RSS + \lambda \sum |\beta_i| \right)$$

- Elastic Net

$$\beta^* = \text{argmin}\left(RSS + \lambda_1 \sum |\beta_i| + \lambda_2 \sum \beta_i^2 \right)$$

Practices

- Cross-validation of linear model
  - sklearn.linear_model.Ridge
  - sklearn.linear_model.Lasso

- Practice 1
  - By applying 5-fold cross-validation for lasso and elastic net, find the best model. What is you test score?

- Practice
  - Read ‘data02_college.csv’, calculate the acceptance rate from the data, and predict the acceptance rate using elastic net.
  - What is your score on the test set?

Appendix

References

- Probability and Stochastic Processes: A Friendly Introduction to Electrical and Computer Engineers (3rd edition), Yates and Goodman, Wiley
- Probability, Statistics, and Random Processes for Electrical Engineering (3rd edition), Leon-Garcia, Pearson International Edition.
- An Introduction to Statistical Learning with Applications in R, James, Witten, Hastie, Tibshirani, Springer
- Pattern Recognition and Machine Learning, Bishop, Springer

About the Lecturer

- **Junhee Seok, PhD**
  - Assistant Professor, Electrical Engineering, Korea University
  - Director of Mirae Asset AI Fintech Research Center
  - Education
    - BS, Electrical Engineering, KAIST, 2001
    - PhD, Electrical Engineering, Stanford University, 2011
  - Professional Experiences
    - Postdoctoral Fellow, Statistics, Stanford University
    - Assistant Professor, HBMI, Northwestern University
  - Research Area
    - Big data analytics, Machine Learning, AI
    - Biomedicine, Finance, Climate, IoT, Materials, and etc.