Data Analysis 2021 Spring





**Lecture 09:**

**Linear Model Selection and Regularization**

May 5 & May 10, 2021

**Taesoo Kwon**

[tskwon@seoultech.ac.kr](mailto:tskwon@seoultech.ac.kr)

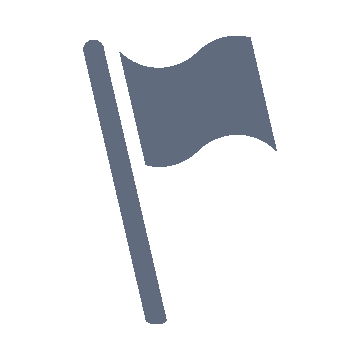
©copyright CIEL 2015 https://ciel.seoultech.ac.kr/

# Course Schedule (Tentative)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Week** | **Topics** | **Note** | **Date (W)** | **Date (M)** |
| 1 | Orientation, Statistical Learning (Ch2) | Online | 03/03 | 03/08 |
| 2 | Statistical Learning (Ch2), Python Programming | Online | 03/10 | 03/15 |
| 3 | Probability & Statistics | Online | 03/17 | 03/22 |
| 4 | Probability & Statistics | Online | 03/24 | 03/29 |
| 5 | Linear Regression (Ch3) | Online | 03/31 | 04/05 |
| 6 | Linear Regression (Ch3) | Online | 04/07 | 04/12 |
| 7 | Classification (Ch4) | Online | 04/14 | 04/19 |
| 8 | **Midterm exam** | **Class hours (W1-W7)** | **04/21** | **04/26** |
| 9 | Resampling Methods (Ch5) | Online | 04/28 | 05/03 |
| **10** | Linear Model Selection and Regularization (Ch6) | Online | 05/05 | 05/10 |
| 11 | Moving Beyond Linearity (Ch7) | Online | 05/12 | 05/17 |
| 12 | Tree-Based Methods (Ch8) | Online | 05/19 | 05/24 |
| 13 | Support Vector Machines (Ch9) | Online | 05/26 | 05/31 |
| 14 | Unsupervised Learning (Ch10) | Online | 06/02 | 06/07 |
| 15 | **Final exam** | **7pm or Class hours (W9-W14)** | **06/09or14** | **06/09or14** |

©copyright CIEL 2015 **2**





* Linear model selection and regularization

**OUTLINES**

* + Subset selection
  + Shrinkage methods
  + Dimension reduction methods
* Python lab
* Summary & Next class

©copyright CIEL 2015



**Linear Model Selection and Regularization: Ch 6**



* Linear model selection and regularization
* Python lab
* Summary & Next class

©copyright CIEL 2015 https://ciel.seoultech.ac.kr/

# Linear Model Selection and Regularization

* Recall the linear model



* In the lectures that follow, we consider some approaches for extending the linear model framework
  + In the lectures covering Chapter 7 of the text, we generalize the linear model in order to accommodate nonlinear, but still additive, relationships
* In the lectures covering Chapter 8 we consider even more general nonlinear models.

©copyright CIEL 2015 **5**

# In Praise of Linear models

* Despite its simplicity, the linear model has distinct advantages in terms of its interpretability and often shows good predictive performance
* Hence we discuss in this lecture some ways in which the simple linear model can be improved, by replacing ordinary least squares fitting with some alternative fitting procedures

©copyright CIEL 2015 **6**

# Why Consider Alternatives to Least Squares?

* Prediction Accuracy
  + Especially when 𝑝𝑝 > 𝑛𝑛, to control the variance
* Model Interpretability
  + By removing irrelevant features that is, by setting the corresponding coefficient estimates to zero, we can obtain a model that is more easily interpreted
  + We will present some approaches for automatically performing feature selection

©copyright CIEL 2015 **7**

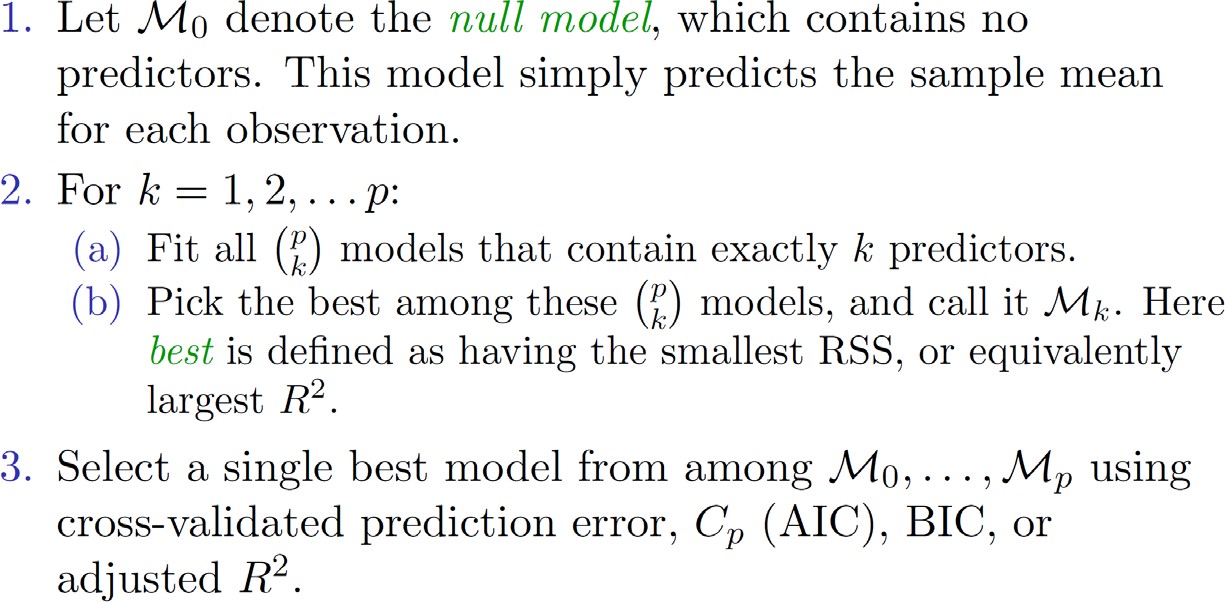
# Three Classes of Methods

* Subset Selection
  + We identify a subset of the 𝑝𝑝 predictors that we believe to be related to the response
  + We then fit a model using least squares on the reduced set of variables.
* Shrinkage
  + We fit a model involving all 𝑝𝑝 predictors, but the estimated coefficients are shrunken towards zero relative to the least squares estimates
  + This shrinkage (also known as regularization) has the effect of reducing variance and can also perform variable selection
* Dimension Reduction
  + We project the 𝑝𝑝 predictors into a 𝑀𝑀-dimensional subspace, where 𝑀𝑀 < 𝑝𝑝
  + This is achieved by computing 𝑀𝑀 different linear combinations, or projections, of the variables
  + Then these 𝑀𝑀 projections are used as predictors to fit a linear regression model by least squares

©copyright CIEL 2015 **8**

# Subset Selection

* Best subset and stepwise model selection procedures
* Best subset selection



©copyright CIEL 2015 **9**

# Example: Credit Data Set



* For each possible model containing a subset of the ten predictors in the Credit data set, the RSS and

𝑅𝑅2 are displayed

* The red frontier tracks the best model for a given number of predictors, according to RSS and 𝑅𝑅2
* Though the data set contains only ten predictors, the 𝑥𝑥-axis ranges from 1 to 11, since one of the variables is categorical and takes on three values, leading to the creation of two dummy variables

©copyright CIEL 2015 **10**

# Extensions to Other Models

* Although we have presented best subset selection here for least squares regression, the same ideas apply to other types of models, such as logistic regression
* The deviance, negative two times the maximized log-likelihood, plays the role of RSS for a broader class of models.

©copyright CIEL 2015 **11**

# Stepwise Selection

* For computational reasons, best subset selection cannot be applied with very large 𝑝𝑝.
* Best subset selection may also suffer from statistical problems when 𝑝𝑝 is large
  + Larger the search space, the higher the chance of finding models that look good on the training data, even though they might not have any predictive power on future data.
* Thus an enormous search space can lead to overfitting and high variance of the coefficient estimates.
* For both of these reasons, stepwise methods, which explore a far more restricted set of models, are attractive alternatives to best subset selection.

©copyright CIEL 2015 **12**

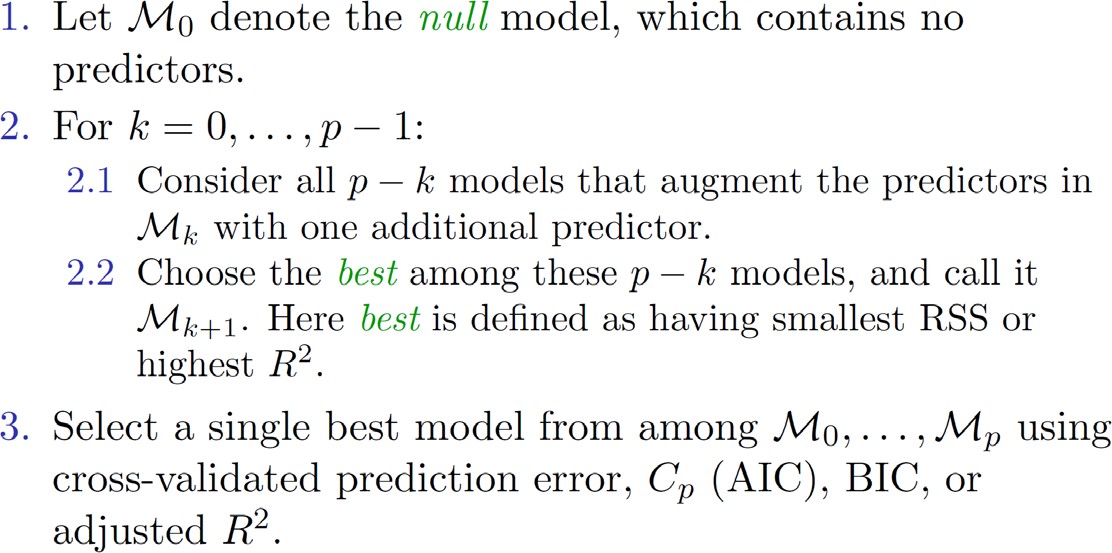
# Forward Stepwise Selection

* Forward stepwise selection begins with a model containing no predictors, and then adds predictors to the model, one-at-a-time, until all of the predictors are in the model
* In particular, at each step the variable that gives the greatest additional improvement to the fit is added to the model

©copyright CIEL 2015 **13**

# Forward Stepwise Selection [cont.]

* Forward Stepwise Selection



©copyright CIEL 2015 **14**

# More on Forward Stepwise Selection

* Computational advantage over best subset selection is clear
  + Only 1 + ∑𝑝𝑝−1(𝑝𝑝 − 𝑘𝑘) = 1 + 𝑝𝑝(𝑝𝑝 + 1)/2 models

𝑘𝑘=0

* It is not guaranteed to find the best possible model out of all 2𝑝𝑝 models containing subsets of the 𝑝𝑝 predictors
  + Why not?

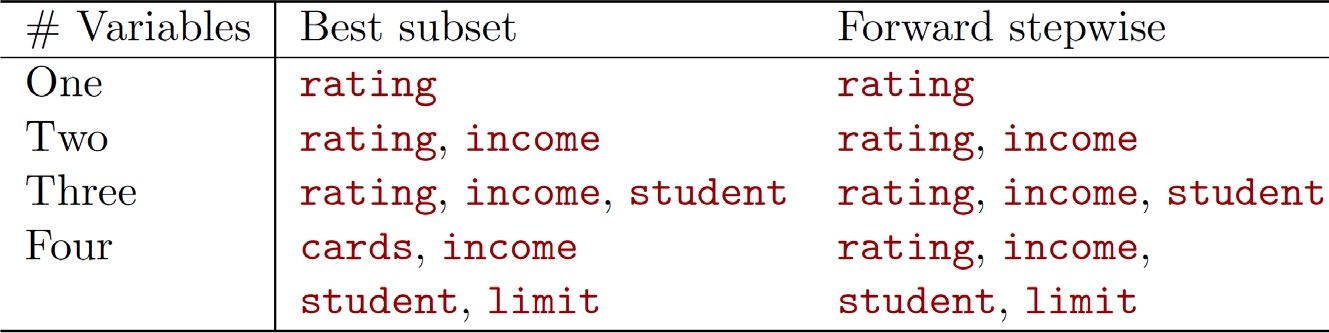
©copyright CIEL 2015 **15**

# Forward Stepwise Selection: Credit Data Example

* The first four selected models for best subset selection and forward stepwise selection on the

Credit data set

* The first three models are identical but the fourth models differ



©copyright CIEL 2015 **16**

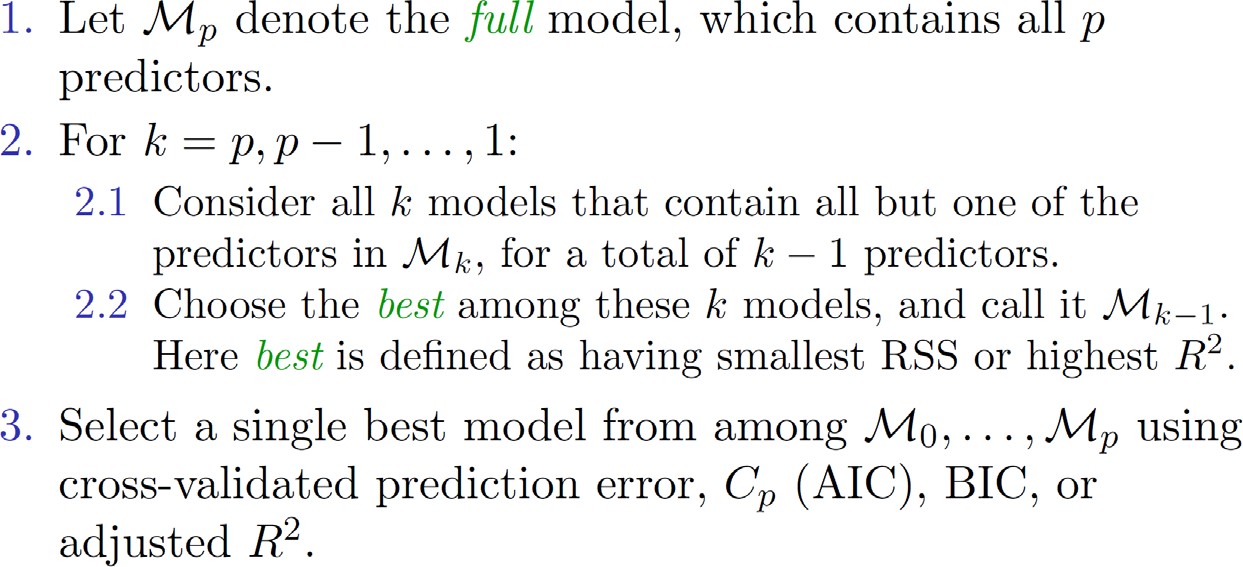
# Backward Stepwise Selection

* Like forward stepwise selection, backward stepwise selection provides an efficient alternative to best subset selection
* However, unlike forward stepwise selection, it begins with the full least squares model containing all 𝑝𝑝 predictors, and then iteratively removes the least useful predictor, one-at-a-time

©copyright CIEL 2015 **17**

# Backward Stepwise Selection [cont.]

* Backward Stepwise Selection



©copyright CIEL 2015 **18**

# More on Backward Stepwise Selection

* Like forward stepwise selection, the backward selection approach searches through only 1 +

𝑝𝑝(𝑝𝑝 + 1)/2 models, and so can be applied in settings where 𝑝𝑝 is too large to apply best subset selection

* Like forward stepwise selection, backward stepwise selection is not guaranteed to yield the best model containing a subset of the 𝑝𝑝 predictors
* Backward selection requires that the number of samples 𝑛𝑛 is larger than the number of variables

𝑝𝑝 (so that the full model can be fit)

* + In contrast, forward stepwise can be used even when 𝑛𝑛 < 𝑝𝑝, and so is the only viable subset method when 𝑝𝑝 is very large.

©copyright CIEL 2015 **19**

# Choosing the Optimal Model

* The model containing all of the predictors will always have the smallest RSS and the largest 𝑅𝑅2, since these quantities are related to the training error
* We wish to choose a model with low test error, not a model with low training error. Recall that training error is usually a poor estimate of test error
* Therefore, RSS and 𝑅𝑅2 are not suitable for selecting the best model among a collection of models with different numbers of predictors

©copyright CIEL 2015 **20**

# Estimating Test Error: Two Approaches

* We can indirectly estimate test error by making an adjustment to the training error to account for the bias due to overfitting.
* We can directly estimate the test error, using either a validation set approach or a cross- validation approach, as discussed in previous lectures
* We illustrate both approaches next

©copyright CIEL 2015 **21**

𝑪𝑪𝒑𝒑**, AIC, BIC, and Adjusted** 𝑹𝑹𝟐𝟐

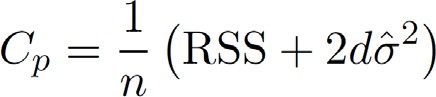
* These techniques adjust the training error for the model size, and can be used to select among a set of models with different numbers of variables
* The next figure displays 𝐶𝐶𝑝𝑝, BIC, and adjusted 𝑅𝑅2 for the best model of each size produced by best subset selection on the Credit data set

©copyright CIEL 2015 **22**



# Now for Some Details

* 𝐶𝐶𝑝𝑝: for a fitted least square model

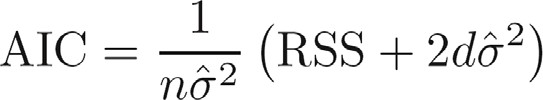


* + 𝑑𝑑 is the total # of parameters used and each response measurement

𝜎𝜎�2 is an estimate of the variance of the error 𝜖𝜖 associated with



* AIC (Akaike information criterion) criterion is defined for a large class of models fit by maximum likelihood:



* + 𝐿𝐿 is the maximized value of the likelihood function for the estimated model
* In the case of the linear model with Gaussian errors, maximum likelihood and least squares are the same thing, and 𝐶𝐶𝑝𝑝 and AIC are equivalent

©copyright CIEL 2015 **23**

# Details on BIC

* Like 𝐶𝐶𝑝𝑝, the BIC will tend to take on a small value for a model with a low test error, and so generally we select the model that has the lowest BIC value.
* Notice that BIC replaces the 2𝑑𝑑𝜎𝜎�2 used by 𝐶𝐶𝑝𝑝 with a 2 log(𝑛𝑛) 𝑑𝑑𝜎𝜎�2 term, where 𝑛𝑛 is the number of observations

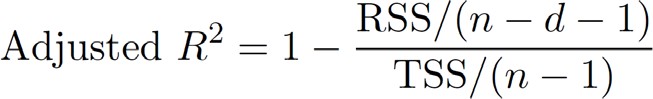


* Since log 𝑛𝑛 > 2 for any 𝑛𝑛 > 7, the BIC statistic generally places a heavier penalty on models with many variables, and hence results in the selection of smaller models than 𝐶𝐶𝑝𝑝

©copyright CIEL 2015 **24**

**Adjusted** 𝑹𝑹𝟐𝟐

* For a least squares model with 𝑑𝑑 variables, the adjusted 𝑅𝑅2 statistic is calculated as
  + TSS is the total sum of squares



* Unlike 𝐶𝐶𝑝𝑝, AIC, and BIC, for which a small value indicates a model with a low of adjusted 𝑅𝑅2 indicates a model with a small test error

test error, a large value

* Maximizing the adjusted 𝑅𝑅2 is equivalent to minimizing RSS

𝑛𝑛−𝑑𝑑−1

* While RSS always decreases as the number of variables in the model increases, RSS

may increase or

decrease, due to the presence of 𝑑𝑑 in the denominator

𝑛𝑛−𝑑𝑑−1



* Unlike the 𝑅𝑅2 statistic, the adjusted 𝑅𝑅2 statistic pays a price for the inclusion of unnecessary variables in the model

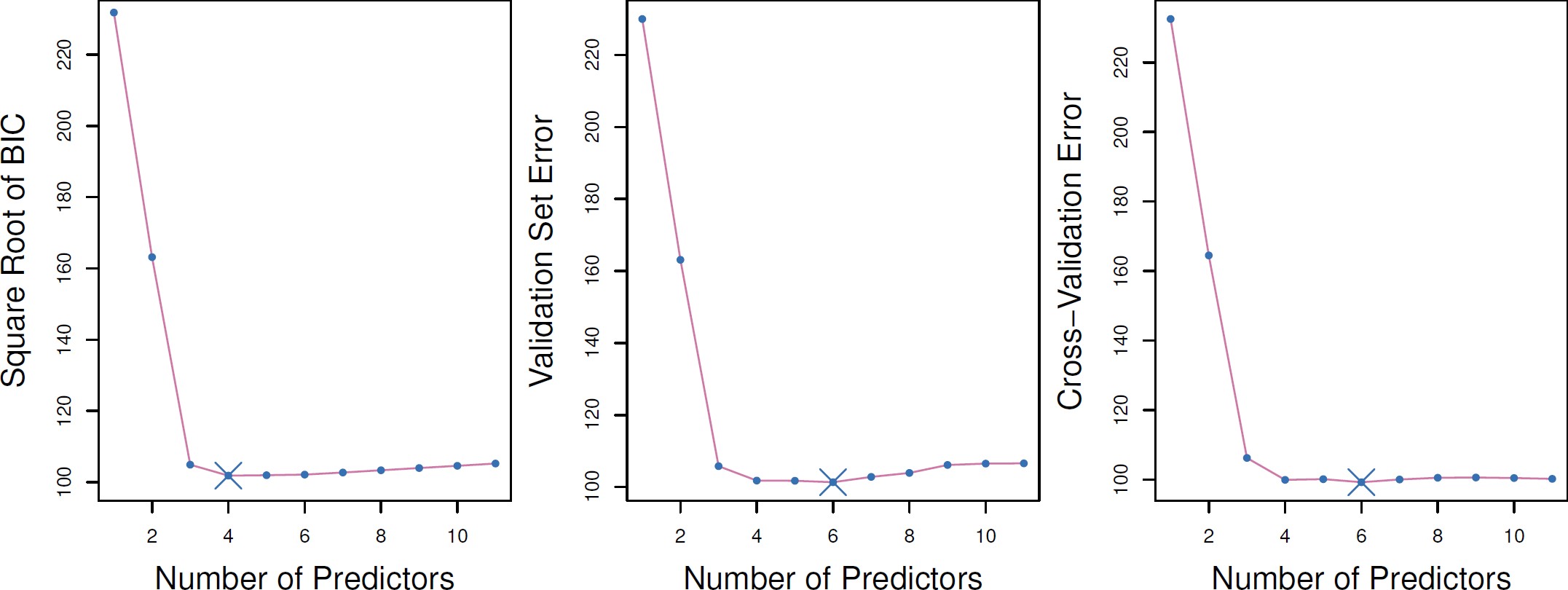
©copyright CIEL 2015 **25**

# Validation and Cross-Validation

* Each of the procedures returns a sequence of models  indexed by model size 𝑘𝑘 = 0,1,2, ⋯.
  + Our job here is to select 𝑘𝑘�.
  + Once selected, we will return model
* We compute the validation set error or the cross-validation error for each model  under consideration, and then select the 𝑘𝑘 for which the resulting estimated test error is smallest
* This procedure has an advantage relative to AIC, BIC, 𝐶𝐶𝑝𝑝, and adjusted 𝑅𝑅2, in that it provides a direct estimate of the test error, and doesn't require an estimate of the error variance 𝜎𝜎2
* It can also be used in a wider range of model selection tasks, even in cases where it is hard to pinpoint the model degrees of freedom (e.g. the number of predictors in the model) or hard to estimate the error variance 𝜎𝜎2.

©copyright CIEL 2015 **26**

# Validation and Cross-Validation: Credit Data Example



©copyright CIEL 2015 **27**

# Details of Previous Figure

* The validation errors were calculated by randomly selecting three-quarters of the observations as the training set, and the remainder as the validation set
* The cross-validation errors were computed using 𝑘𝑘 = 10 folds
* In this case, the validation and cross-validation methods both result in a six-variable model
* However, all three approaches suggest that the four-, five-, and six-variable models are roughly equivalent in terms of their test errors.
* In this setting, we can select a model using the one-standard-error rule
  + We first calculate the standard error of the estimated test MSE for each model size, and then select the smallest model for which the estimated test error is within one standard error of the lowest point on the curve

©copyright CIEL 2015 **28**

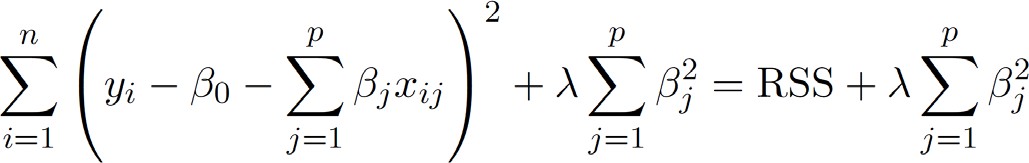
# Shrinkage Methods

* Ridge regression and Lasso
* The subset selection methods use least squares to fit a linear model that contains a subset of the predictors
* As an alternative, we can fit a model containing all 𝑝𝑝 predictors using a technique that constrains or regularizes the coefficient estimates, or equivalently, that shrinks the coefficient estimates towards zero.
* It may not be immediately obvious why such a constraint should improve the fit, but it turns out that shrinking the coefficient estimates can significantly reduce their variance

©copyright CIEL 2015 **29**

# Ridge Regression

* Recall that the least squares fitting procedure estimates 𝛽𝛽0, 𝛽𝛽1, ⋯ , 𝛽𝛽𝑝𝑝 using the values that minimize
* In contrast, the ridge regression coefficient estimates 𝛽𝛽̂𝑅𝑅 are the values that minimize



* + 𝜆𝜆 ≥ 0 is a tuning parameter, to be determined separately

©copyright CIEL 2015 **30**

# Ridge Regression [cont.]

* As with least squares, ridge regression seeks coefficient estimates that fit the data well, by making the RSS small
* However, the second term, 𝜆𝜆 ∑𝑗𝑗 𝛽𝛽2, called a shrinkage penalty, is small when 𝛽𝛽0, 𝛽𝛽1, ⋯ , 𝛽𝛽𝑝𝑝 are

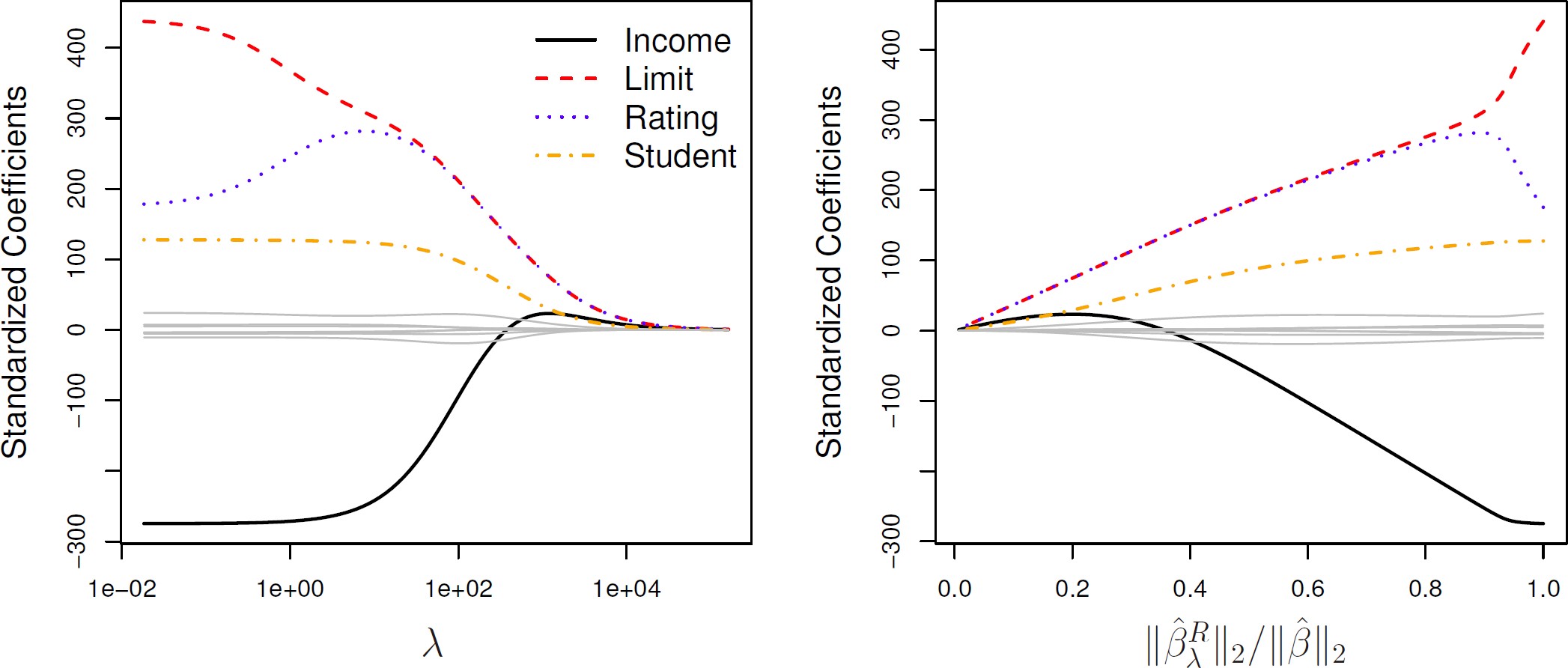
𝑗𝑗

close to zero, and so it has the effect of shrinking the estimates of 𝛽𝛽𝑗𝑗 towards zero

* The tuning parameter 𝜆𝜆 serves to control the relative impact of these two terms on the regression coefficient estimates.
* Selecting a good value for 𝜆𝜆 is critical
  + Cross-validation is used for this.

©copyright CIEL 2015 **31**

# Ridge Regression: Credit Data Example



©copyright CIEL 2015 **32**

# Details of Previous Figure

* In the left-hand panel, each curve corresponds to the ridge regression coeefficient estimate for one of the ten variables, plotted as a function of 𝜆𝜆
* The right-hand panel displays the same ridge coefficient estimates as the left-hand panel, but

instead of displaying 𝜆𝜆 on the x-axis, we now display least squares coefficient estimates

̂𝑅𝑅

𝜆𝜆

𝛽𝛽

� 𝛽𝛽̂

2

, where

2

𝛽𝛽̂

denotes the vector of

* The notation

∑𝑝𝑝

𝑗𝑗=1

𝛽𝛽2

𝑗𝑗

𝛽𝛽 2 =

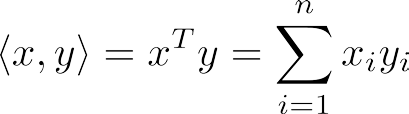
𝛽𝛽

2 denotes the 𝑙𝑙2 norm (pronounced “ell 2”) of a vector, and is defined as



©copyright CIEL 2015 **33**

# [FYI] Inner Product and Norm of Vectors

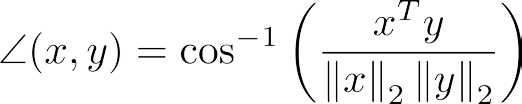
* Standard inner product on
* Euclidean norm, or *l*2-norm



* Cauchy-Schwartz inequality



* Angle between nonzero vectors 



* + and are orthogonal if 

©copyright CIEL 2015 **34**

# [FYI] Inner Product and Norm of Vectors

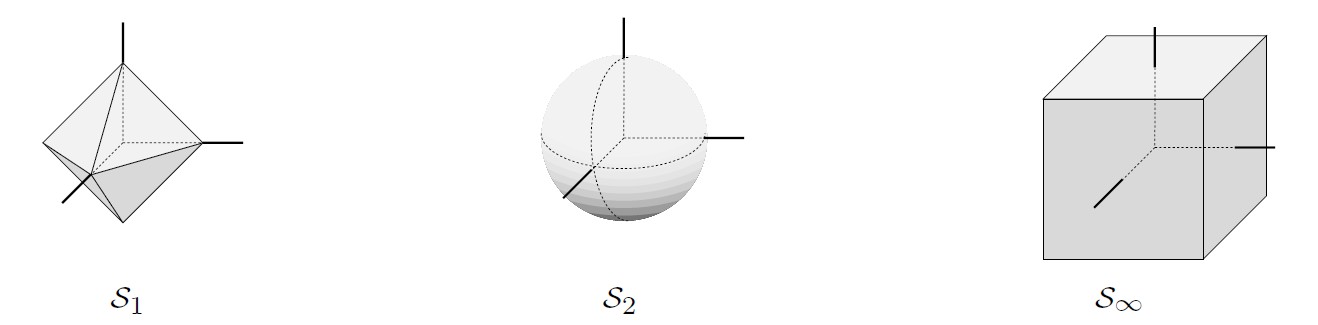
* 𝑙𝑙𝑝𝑝 norms
  + For the 𝑙𝑙𝑝𝑝 norm of is defined as 
* Unit *p*-spheres  for
  + Unit 1-, 2-, -spheres in  are an octahedron, a ball, and cube, respectively

  fits inside  which in turn fits inside 

* + This means that  for all 

o In general, this is true in

**35**

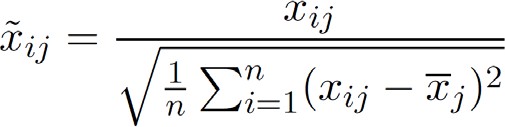


©copyright CIEL 2015

# Ridge Regression: Scaling of Predictors

* The standard least squares coefficient estimates are scale equivariant
  + Multiplying 𝑋𝑋𝑗𝑗 by a constant 𝑐𝑐 simply leads to a scaling of the least squares coefficient estimates by a factor of 1/𝑐𝑐.
  + In other words, regardless of how the 𝑗𝑗th predictor is scaled, 𝑋𝑋 𝛽𝛽̂ will remain the same

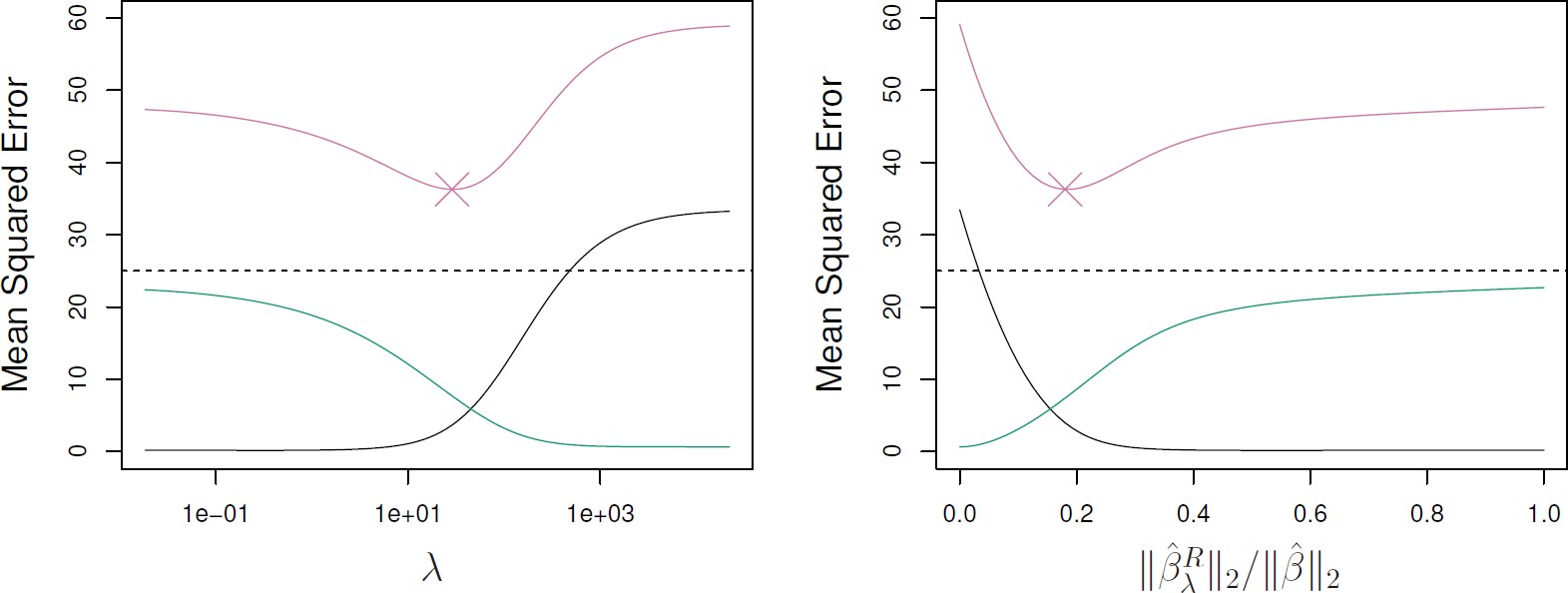
𝑗𝑗 𝑗𝑗

* In contrast, the ridge regression coefficient estimates can change substantially when multiplying a given predictor by a constant, due to the sum of squared coefficients term in the penalty part of the ridge regression objective function
* Therefore, it is best to apply ridge regression after standardizing the predictors, using the formula

©copyright CIEL 2015 **36**

# Why Does Ridge Regression Improve over Least Squares?

* The Bias-Variance tradeoff



* + Simulated data with 𝑛𝑛 = 50 observations, 𝑝𝑝 = 45 predictors, all having nonzero coefficients
  + Squared bias (black), variance (green), and test mean squared error (purple) for the ridge regression

predictions on a simulated data set, as a function of 𝜆𝜆 and

̂𝑅𝑅

𝜆𝜆

𝛽𝛽

� 𝛽𝛽̂

2 2

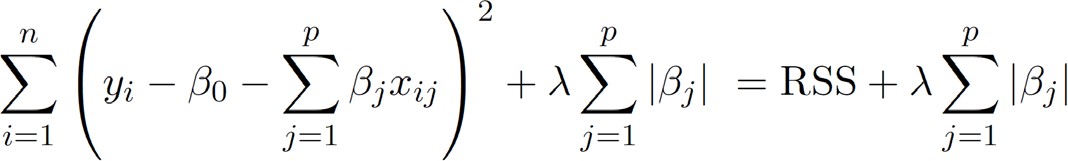
* + The horizontal dashed lines indicate the minimum possible MSE
  + The purple crosses indicate the ridge regression models for which the MSE is smallest

©copyright CIEL 2015 **37**

# Lasso

* Least absolute shrinkage and selection operator (Lasso)
* Ridge regression does have one obvious disadvantage
  + Unlike subset selection, which will generally select models that involve just a subset of the variables, ridge regression will include all 𝑝𝑝 predictors in the final model
* The Lasso is a relatively recent alternative to ridge regression that overcomes this disadvantage.
  + The lasso coefficients, 𝛽𝛽̂𝐿𝐿, minimize the quantity

𝜆𝜆



* In statistical parlance, the lasso uses and 𝑙𝑙1 (pronounced “ell 1”) penalty instead of an 𝑙𝑙2 penalty
  + The 𝑙𝑙1 norm of a coefficient vector 𝛽𝛽 is given by 𝛽𝛽 1 = ∑𝑝𝑝 𝛽𝛽𝑗𝑗

𝑗𝑗=1

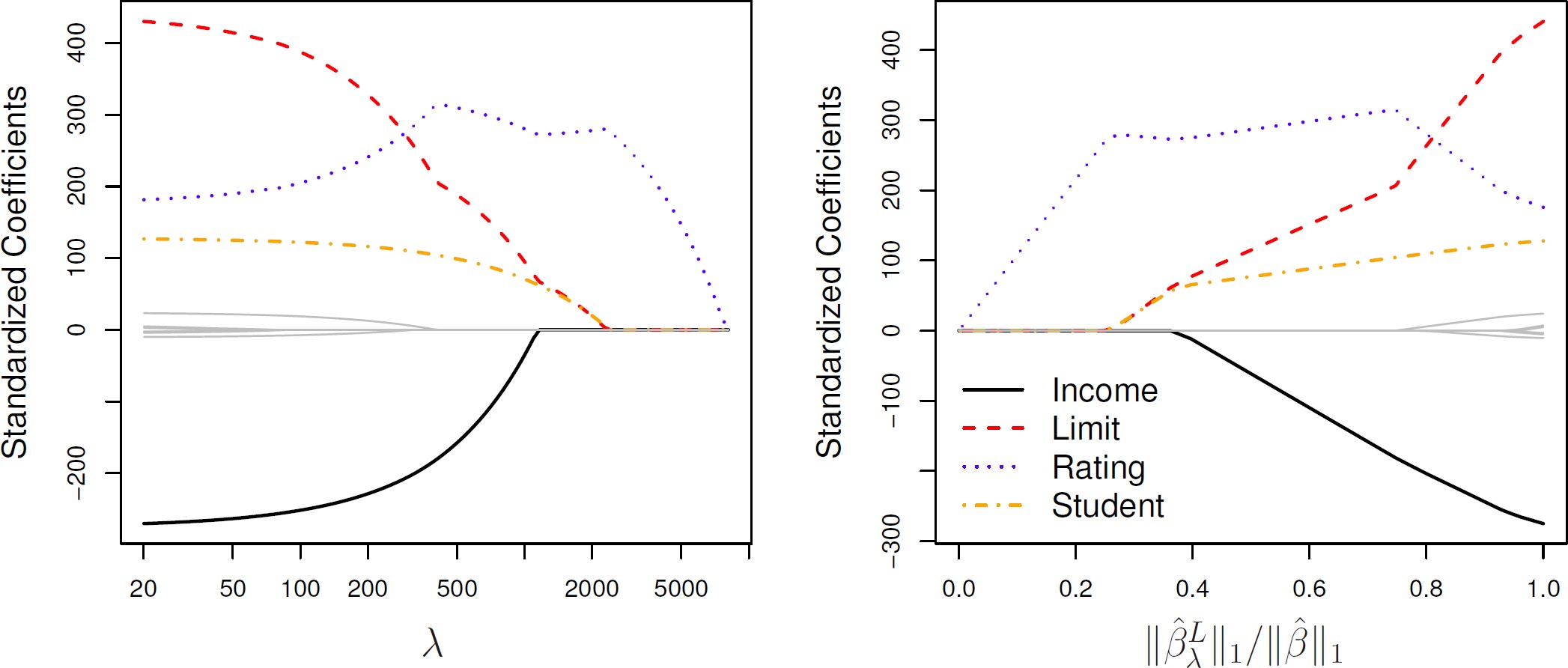
©copyright CIEL 2015 **38**

# Lasso [cont.]

* As with ridge regression, the lasso shrinks the coefficient estimates towards zero
* However, in the case of the lasso, the 𝑙𝑙1 penalty has the effect of forcing some of the coefficient estimates to be exactly equal to zero when the tuning parameter 𝜆𝜆 is sufficiently large
* Hence, much like best subset selection, the lasso performs variable selection
* We say that the lasso yields sparse models. that is, models that involve only a subset of the variables
* As in ridge regression, selecting a good value of 𝜆𝜆 for the lasso is critical
  + Cross-validation is again the method of choice

©copyright CIEL 2015 **39**

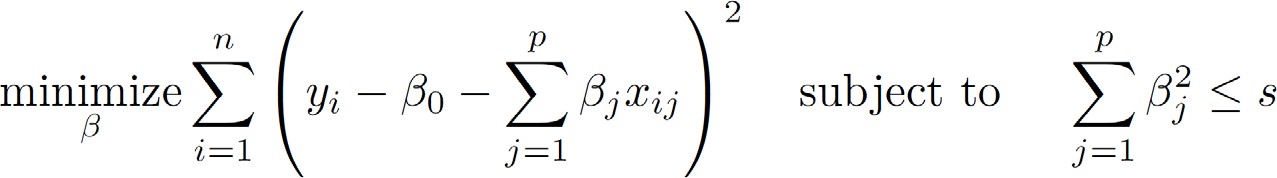
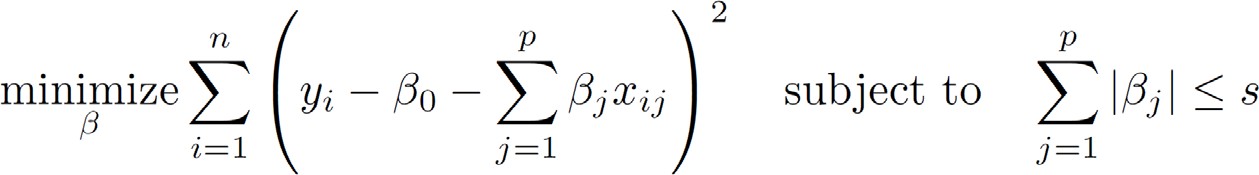
# Lasso: Credit Data Example



©copyright CIEL 2015 **40**

# Variable Selection Property of Lasso

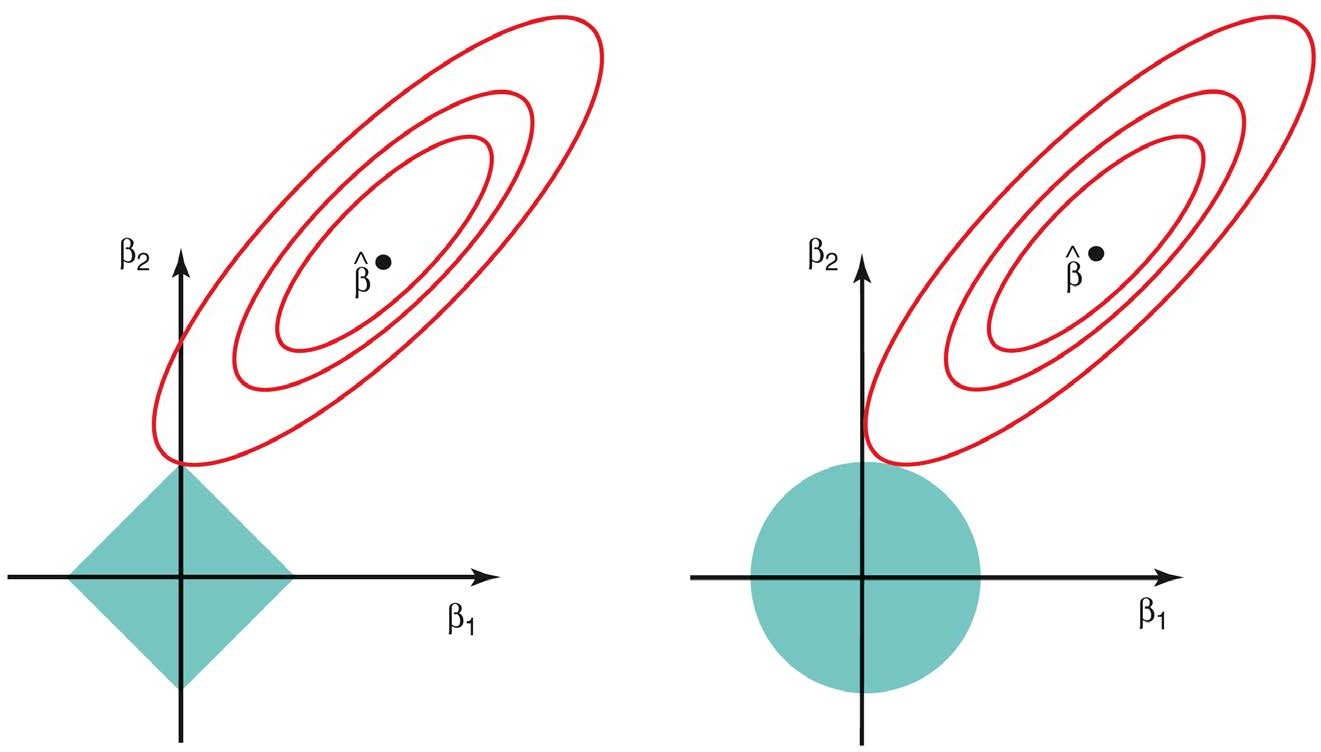
* Why is it that the lasso, unlike ridge regression, results in coefficient estimates that are exactly equal to zero?
* One can show that the lasso and ridge regression coefficient estimates solve the problems



©copyright CIEL 2015 **41**

# Lasso Picture

* Contours of error and constraint functions
  + Lasso  Ridge regression



©copyright CIEL 2015 **42**

# Comparing Lasso and Ridge Regression



* Left: Plots of squared bias (black), variance (green), and test MSE (purple) for the lasso on simulated data set
* Right: Comparison of squared bias, variance and test MSE between lasso (solid) and ridge (dashed)
* Both are plotted against their 𝑅𝑅2 on the training data, as a common form of indexing
* The crosses in both plots indicate the lasso model for which the MSE is smallest.

©copyright CIEL 2015 **43**

# Comparing Lasso and Ridge Regression [cont.]

* Left: Plots of squared bias (black), variance (green), and test MSE (purple) for the lasso.
  + The simulated data is similar to that in the last slide, except that now only two predictors are related to the response.
* Right: Comparison of squared bias, variance and test MSE between lasso (solid) and ridge (dashed).
* Both are plotted against their 𝑅𝑅2 on the training data, as a common form of indexing
* The crosses in both plots indicate the lasso model for which the MSE is smallest

©copyright CIEL 2015 **44**

# Comparing Lasso and Ridge Regression [cont.]

* These two examples illustrate that neither ridge regression nor the lasso will universally dominate the other
* In general, one might expect the lasso to perform better when the response is a function of only a relatively small number of predictors
* However, the number of predictors that is related to the response is never known a priori for real data sets.
* A technique such as cross-validation can be used in order to determine which approach is better on a particular data set

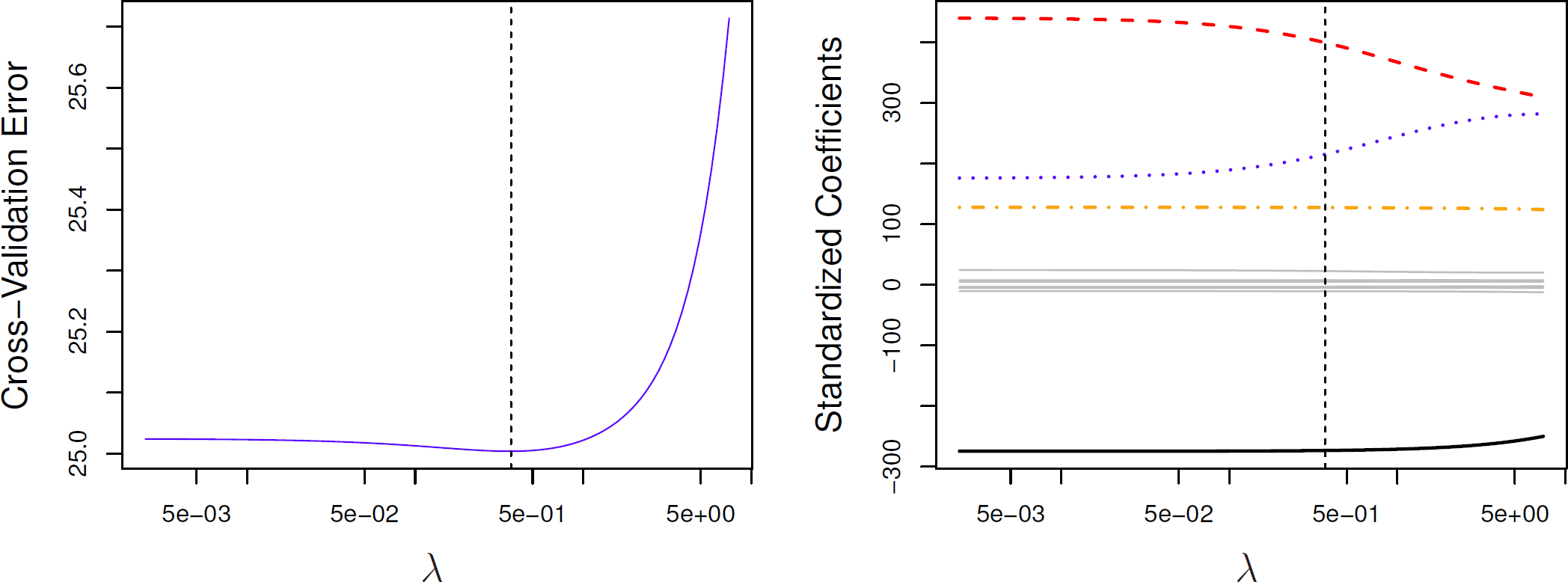
©copyright CIEL 2015 **45**

# Selecting Tuning Parameter for Ridge Regression and Lasso

* As for subset selection, for ridge regression and lasso we require a method to determine which of the models under consideration is best
* That is, we require a method selecting a value for the tuning parameter 𝜆𝜆 or equivalently, the value of the constraint 𝑠𝑠
* Cross-validation provides a simple way to tackle this problem
  + We choose a grid of 𝜆𝜆 values, and compute the cross-validation error rate for each value of 𝜆𝜆
* We then select the tuning parameter value for which the cross-validation error is smallest
* Finally, the model is re-fit using all of the available observations and the selected value of the tuning parameter

©copyright CIEL 2015 **46**

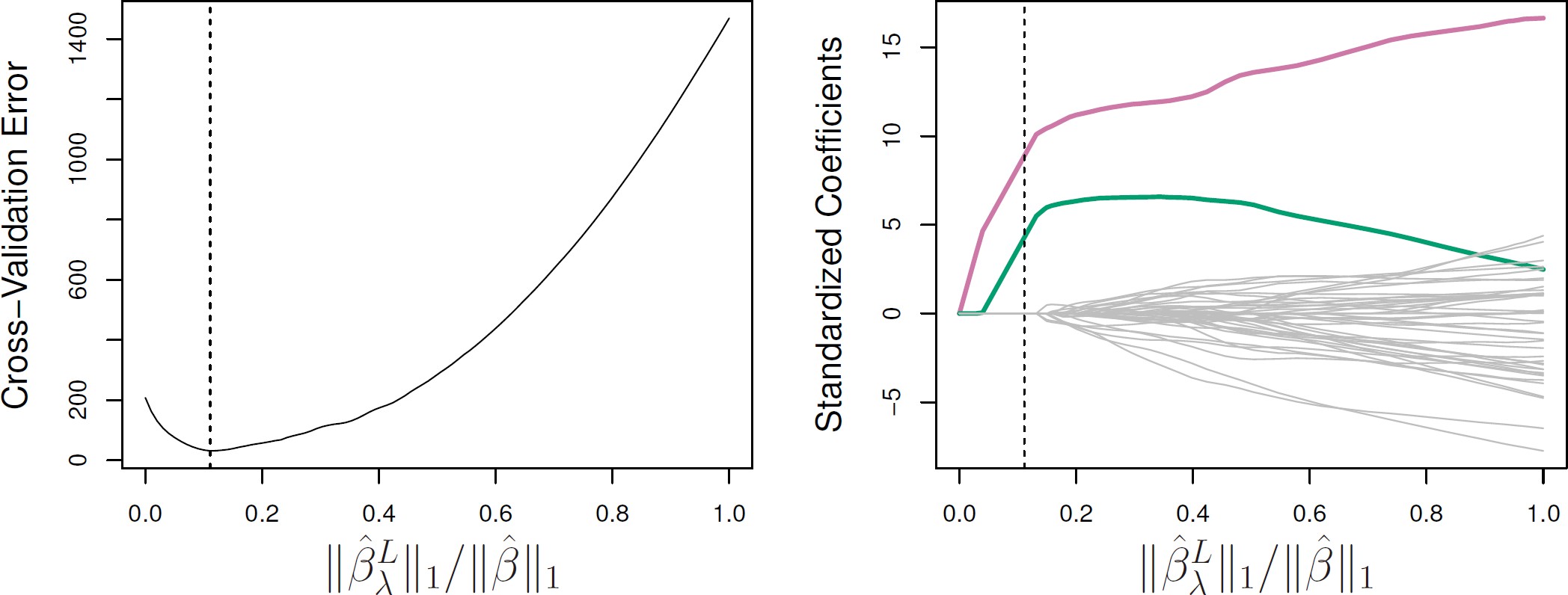
# Selecting Tuning Parameter: Credit Data Example



* Left: Cross-validation errors that result from applying ridge regression to the Credit data set with various values of 𝜆𝜆
* Right: The coefficient estimates as a function of 𝜆𝜆
* The vertical dashed lines indicates the value of 𝜆𝜆 selected by cross-validation

©copyright CIEL 2015 **47**

# Selecting Tuning Parameter: Simulated Data Example



* Left: Ten-fold cross-validation MSE for the lasso, applied to the sparse simulated data set
* Right: The corresponding lasso coefficient estimates are displayed.
* The vertical dashed lines indicate the lasso fit for which the cross-validation error is smallest.

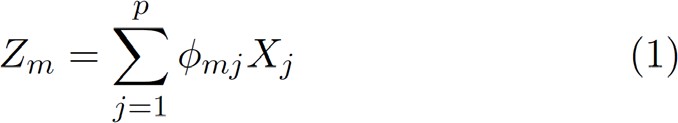
©copyright CIEL 2015 **48**

# Dimension Reduction Methods

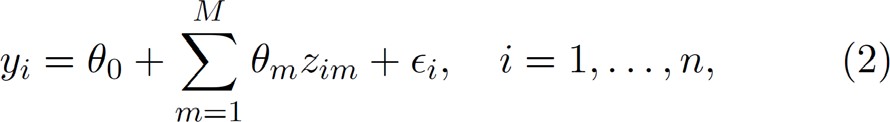
* The methods that we have discussed so far in this chapter have involved fitting linear regression models, via least squares or a shrunken approach, using the original predictors, 𝑋𝑋1, 𝑋𝑋2, ⋯ , 𝑋𝑋𝑝𝑝
* We now explore a class of approaches that transform the predictors and then fit a least squares model using the transformed variables
  + We will refer to these techniques as dimension reduction methods

©copyright CIEL 2015 **49**

# Dimension Reduction Methods [cont.]

* Let 𝑍𝑍1, 𝑍𝑍2, ⋯ , 𝑍𝑍𝑀𝑀 represent 𝑀𝑀 < 𝑝𝑝 linear combinations of our original 𝑝𝑝 predictors. That is,

for some constants 𝜙𝜙𝑚𝑚1, 𝜙𝜙𝑚𝑚2, ⋯ , 𝜙𝜙𝑚𝑚𝑝𝑝

* We can then fit the linear regression model,

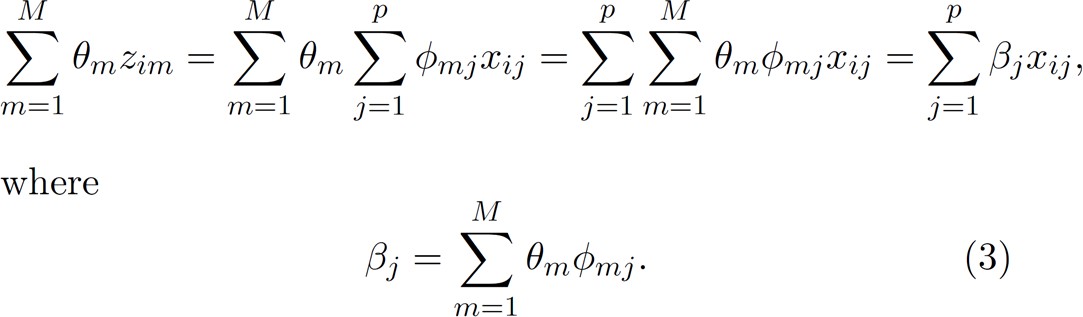
using ordinary least squares

* Note that in model (2), the regression coefficients are given by 𝜃𝜃1, 𝜃𝜃2, ⋯ , 𝜃𝜃𝑀𝑀
* If the constants 𝜙𝜙𝑚𝑚1, 𝜙𝜙𝑚𝑚2, ⋯ , 𝜙𝜙𝑚𝑚𝑝𝑝 are chosen wisely, then such dimension reduction approaches can often outperform OLS regression

©copyright CIEL 2015 **50**

# Dimension Reduction Methods [cont.]

* Notice that from definition (1),



* Hence model (2) can be thought of as a special case of the original linear regression model
* Dimension reduction serves to constrain the estimated 𝛽𝛽𝑗𝑗 coefficients, since now they must take the form (3)
* Can win in the bias-variance tradeoff

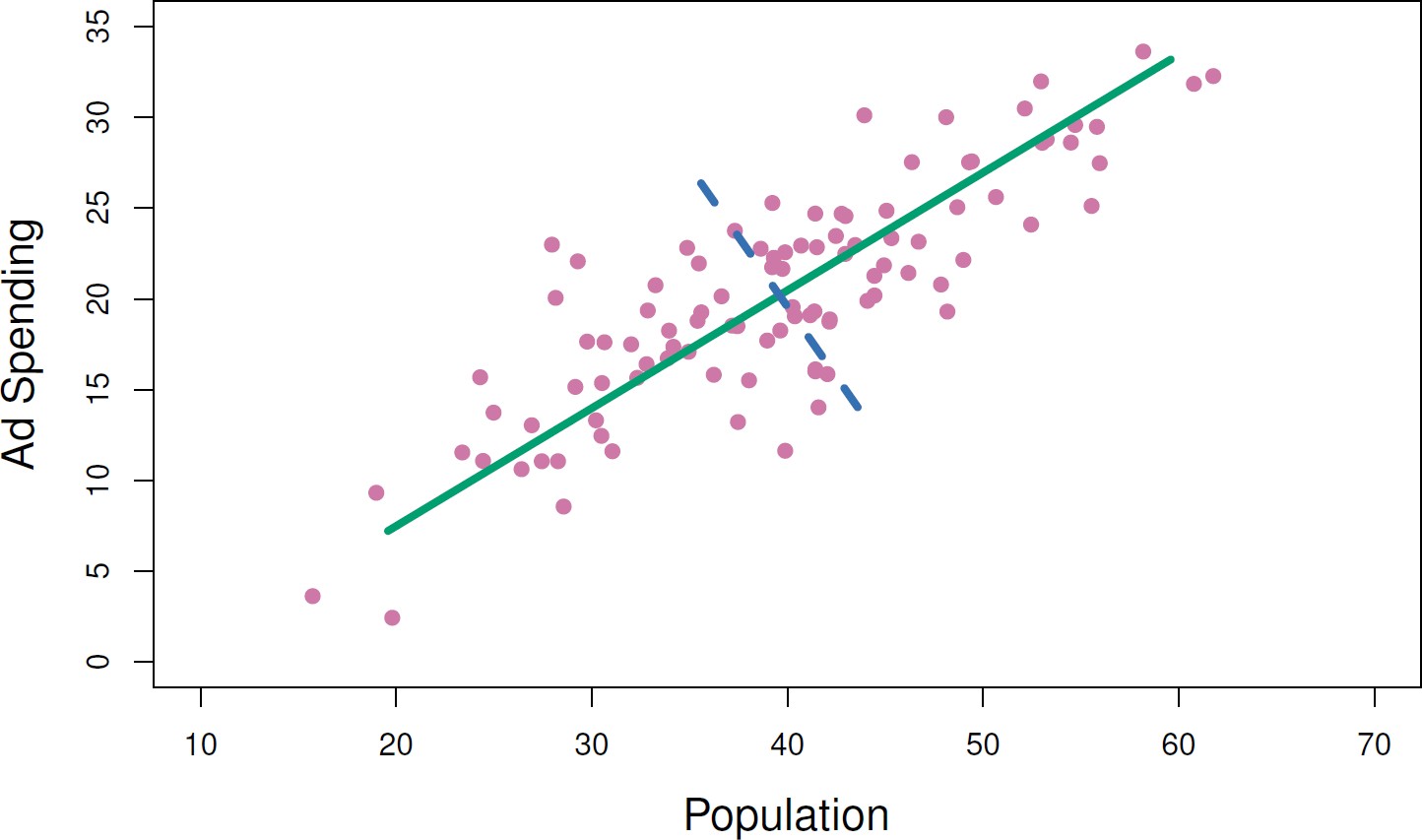
©copyright CIEL 2015 **51**

# Principal Components Regression

* Here we apply principal components analysis (PCA) (discussed in Chapter 10 of the text) to define the linear combinations of the predictors, for use in our regression
* The first principal component is that (normalized) linear combination of the variables with the largest variance
* The second principal component has largest variance, subject to being uncorrelated with the first
* And so on
* Hence with many correlated original variables, we replace them with a small set of principal components that capture their joint variation

©copyright CIEL 2015 **52**

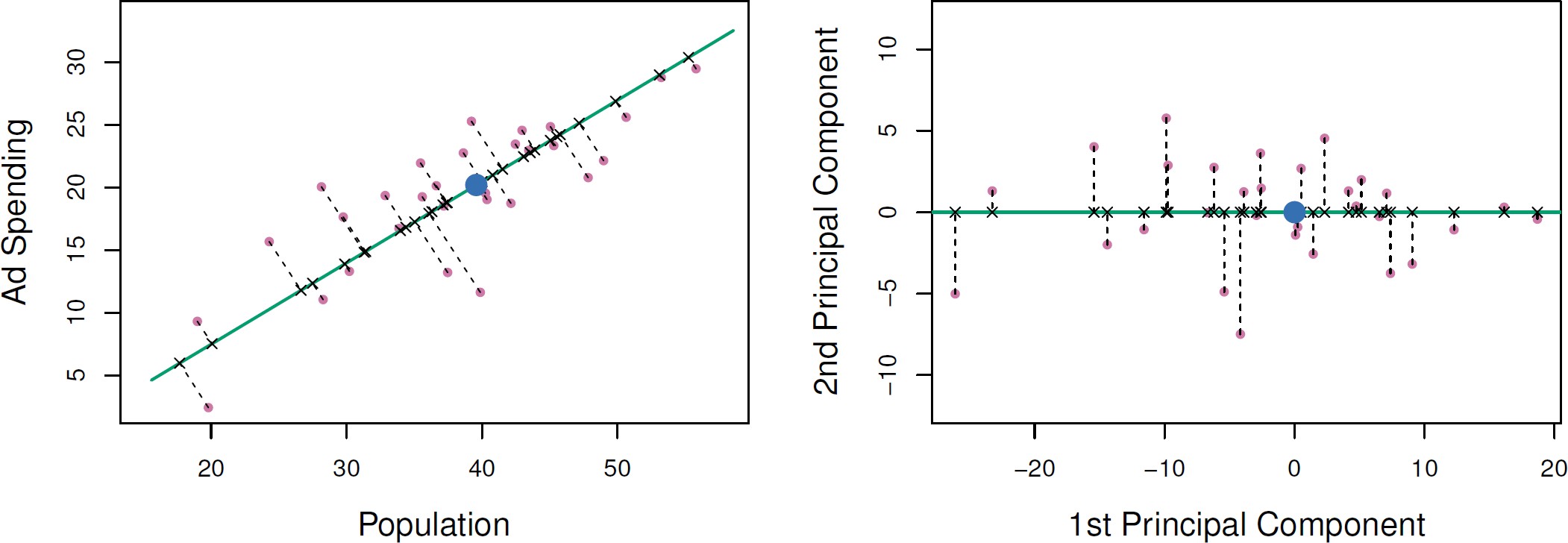
# Pictures of PCA



* The population size (pop) and ad spending (ad) for 100 different cities are shown as purple circles.
* The green solid line indicates the first principal component, and the blue dashed line indicates the second principal component

©copyright CIEL 2015 **53**

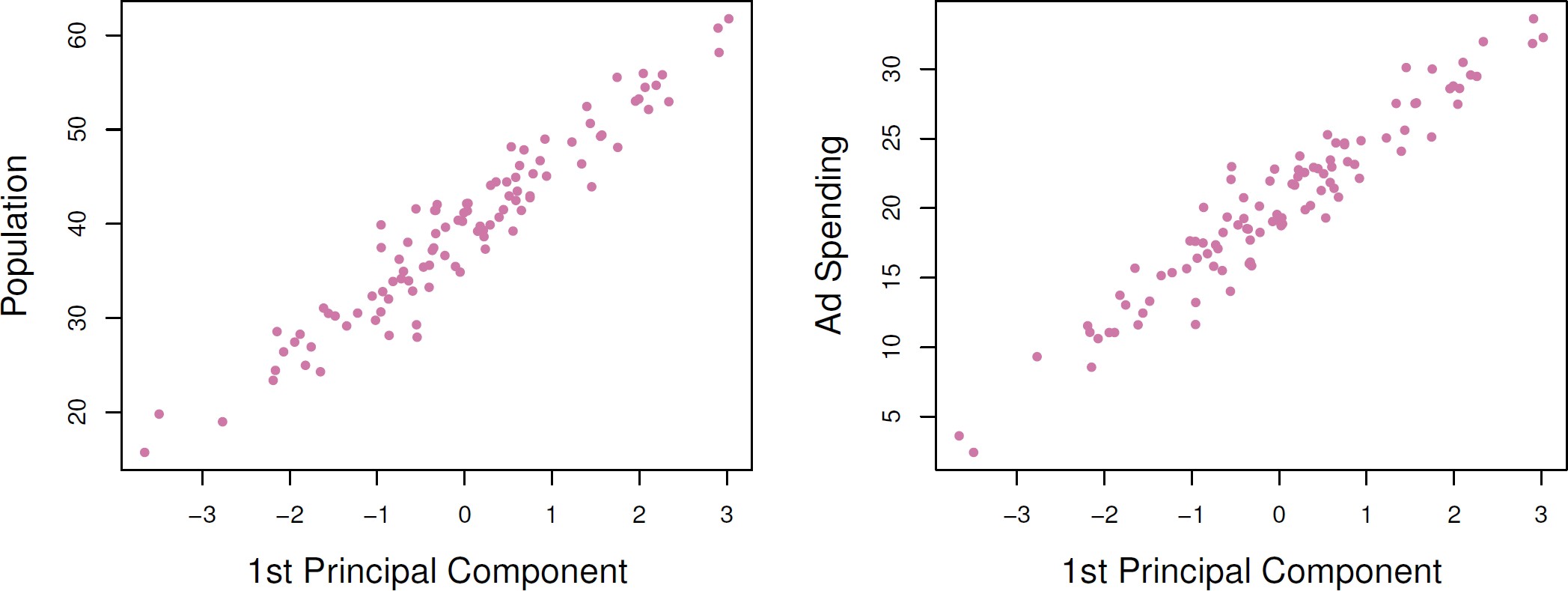
# Pictures of PCA [cont.]



* A subset of the advertising data
* Left: The first principal component, chosen to minimize the sum of the squared perpendicular distances to each point, is shown in green.
  + These distances are represented using the black dashed line segments.
* Right: The left-hand panel has been rotated so that the first principal component lies on the 𝑥𝑥-axis

©copyright CIEL 2015 **54**

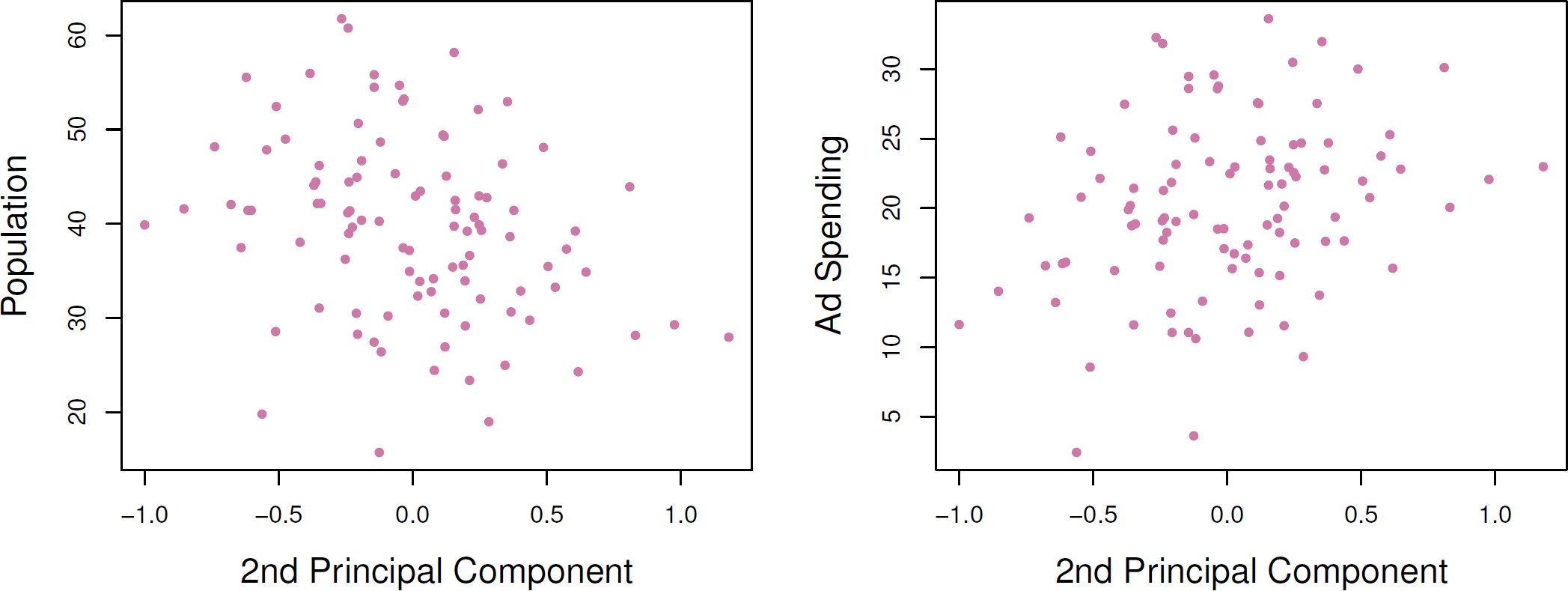
# Pictures of PCA [cont.]



* Plots of the first principal component scores 𝑧𝑧𝑖𝑖1 versus pop and ad
* The relationships are strong

©copyright CIEL 2015 **55**

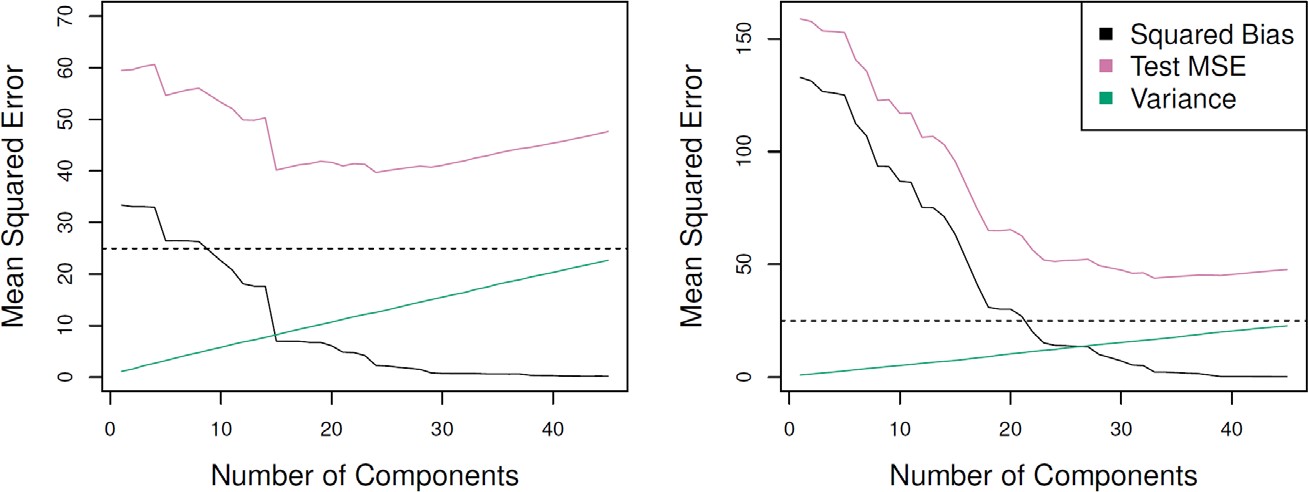
# Pictures of PCA [cont.]



* Plots of the second principal component scores 𝑧𝑧𝑖𝑖2 versus pop and ad
* The relationships are weak

©copyright CIEL 2015 **56**

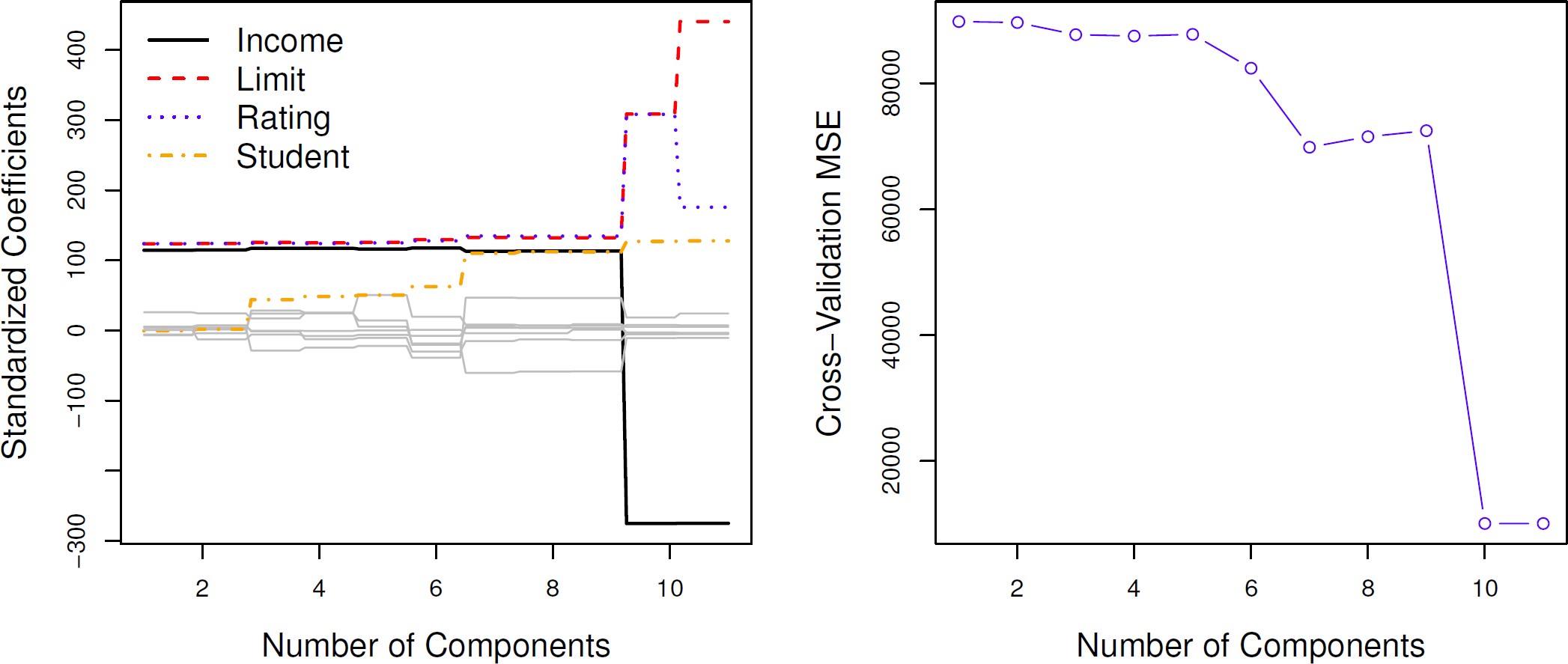
# Application to Principal Components Regression



* PCR was applied to two simulated data sets
* The black, green, and purple lines correspond to squared bias, variance, and test mean squared error, respectively
* Left: Simulated data from slide 37
* Right: Simulated data from slide 44

©copyright CIEL 2015 **57**

# Choosing the number of directions 𝑴𝑴



* Left: PCR standardized coefficient estimates on the Credit data set for different values of 𝑀𝑀
* Right: The 10-fold cross validation MSE obtained using PCR, as a function of 𝑀𝑀

©copyright CIEL 2015 **58**

# Partial Least Squares

* PCR identifies linear combinations, or directions, that best represent the predictors 𝑋𝑋1, 𝑋𝑋2, ⋯ , 𝑋𝑋𝑝𝑝
* These directions are identified in an unsupervised way, since the response 𝑌𝑌 is not used to help determine the principal component directions
* That is, the response does not supervise the identification of the principal components
* Consequently, PCR suffers from a potentially serious drawback
  + There is no guarantee that the directions that best explain the predictors will also be the best directions to use for predicting the response

©copyright CIEL 2015 **59**

# Partial Least Squares [cont.]

* Like PCR, PLS is a dimension reduction method, which first identifies a new set of features

𝑍𝑍1, 𝑍𝑍2, ⋯ , 𝑍𝑍𝑝𝑝 that are linear combinations of the original features, and then fits a linear model via OLS using these 𝑀𝑀 new features

* But unlike PCR, PLS identifies these new features in a supervised way, that is, it makes use of the response 𝑌𝑌 in order to identify new features that not only approximate the old features well, but also that are related to the response
* Roughly speaking, the PLS approach attempts to find directions that help explain both the response and the predictors

©copyright CIEL 2015 **60**

# Partial Least Squares [cont.]

* After standardizing the 𝑝𝑝 predictors, PLS computes the first direction 𝑍𝑍1 by setting each 𝜙𝜙1𝑗𝑗 in

(1) equal to the coefficient from the simple linear regression of 𝑌𝑌 onto 𝑋𝑋𝑗𝑗

* One can show that this coefficient is proportional to the correlation between 𝑌𝑌 onto 𝑋𝑋𝑗𝑗
* Hence, in computing 𝑍𝑍 = ∑𝑝𝑝 𝜙𝜙1𝑗𝑗𝑋𝑋𝑗𝑗, PLS places the highest weight on the variables that are most strongly related to the response

𝑗𝑗=1

* Subsequent directions are found by taking residuals and then repeating the above prescription

©copyright CIEL 2015 **61**



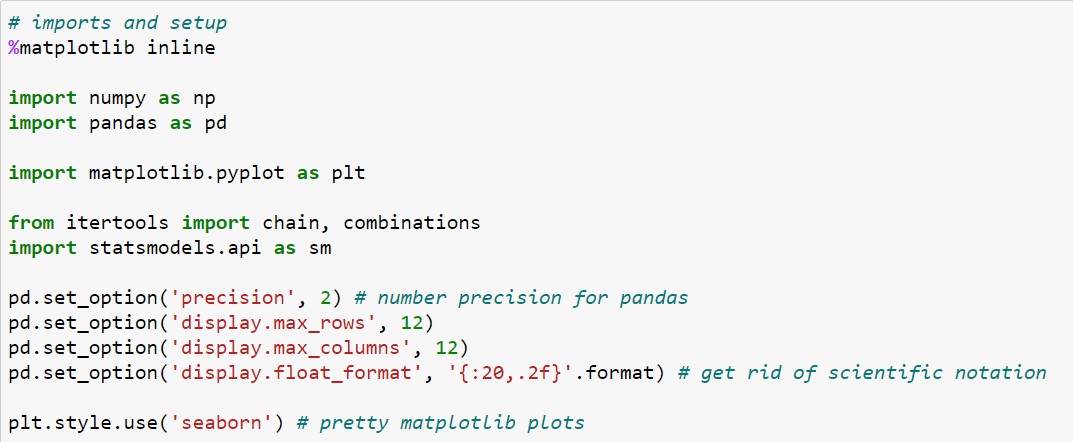
**Python Lab**

* Linear model selection and regularization
* Python lab
* Summary & Next class

©copyright CIEL 2015 https://ciel.seoultech.ac.kr/

# 6.5.1 Best Subset Selection

* + Using Python Libraries
    - Import the libraries that are often used for data analysis



Functions creating iterators for efficient looping

©copyright CIEL 2015 **63**

# 6.5.1 Best Subset Selection

* + Load data: Salary data set
    - Predict a baseball player’s Salary on basis of various statistics associated with performance in previous year

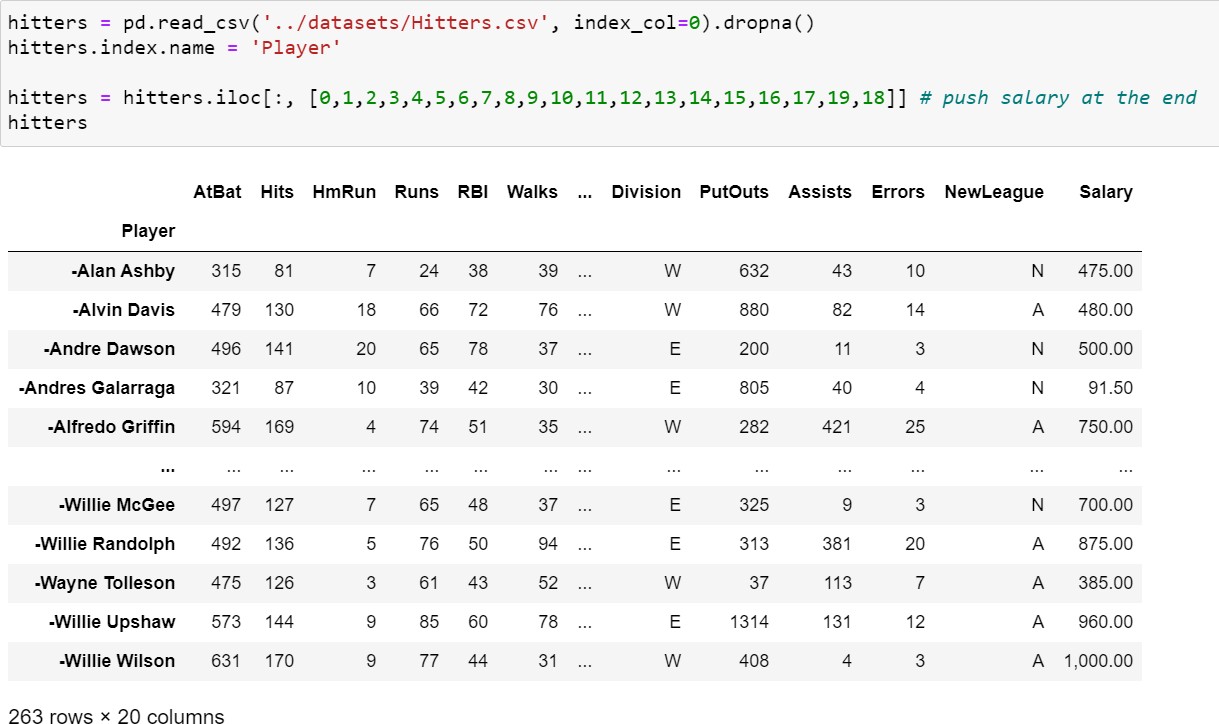


Convert categorical variable into dummy/indicator variables

©copyright CIEL 2015 **64**

# Best Subset Selection

* + - * pandas.get\_dummies : convert categorical variable into dummy/indicator variables



Categorical variable: ‘N’ or ‘A’



Encoded as 0 or 1

©copyright CIEL 2015 **65**

# 6.5.1 Best Subset Selection

## Implement best subset selection

Make an iterator that returns elements from the first iterable until it is exhausted, then proceeds to the next iterable, until all of the iterables are exhausted



itertools.combinations(iterable, r) Return 𝑟𝑟 length subsequences of elements from the input iterable



Ordinary least squares

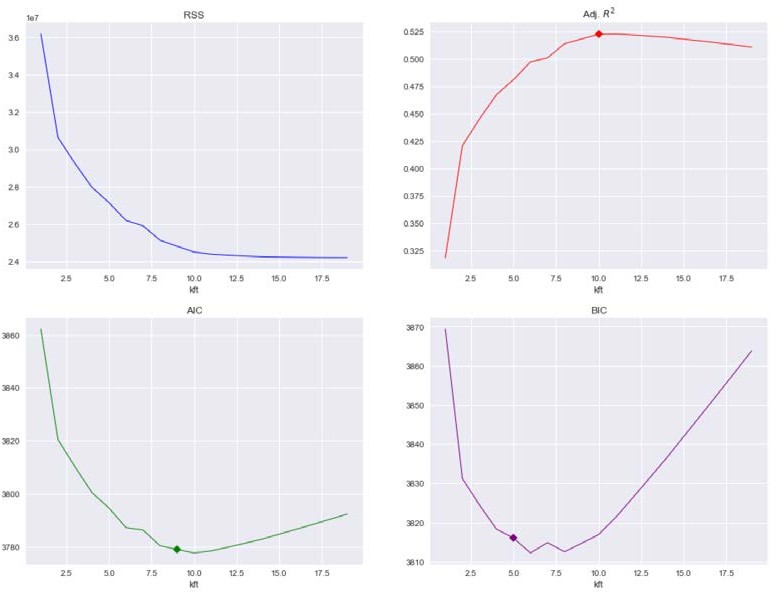
𝛽𝛽0 + 𝛽𝛽1𝑥𝑥

→ 𝛽𝛽0 × 1 + 𝛽𝛽1𝑥𝑥

©copyright CIEL 2015 **66**

# 6.5.1 Best Subset Selection

## Plot RSS, AIC, BIC, and Adjusted 𝑅𝑅2





Read an arbitrary object stored in Python pickle format (pickle module implements binary protocols for serializing and de-serializing a Python object structure)

©copyright CIEL 2015 **67**

# 6.5.2 Forward and Backward Stepwise Selection

## Implement forward and backward stepwise selection



Output: regression model, RSS

Linear regression

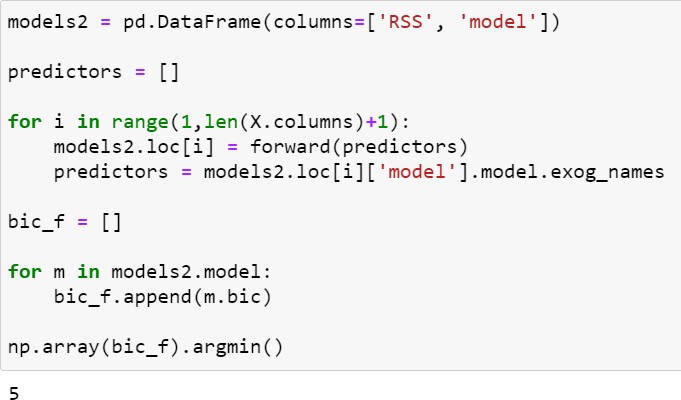
best model: minimizing RSS

best model: minimizing RSS

©copyright CIEL 2015 **68**

# 6.5.2 Forward and Backward Stepwise Selection

## Run forward stepwise selection

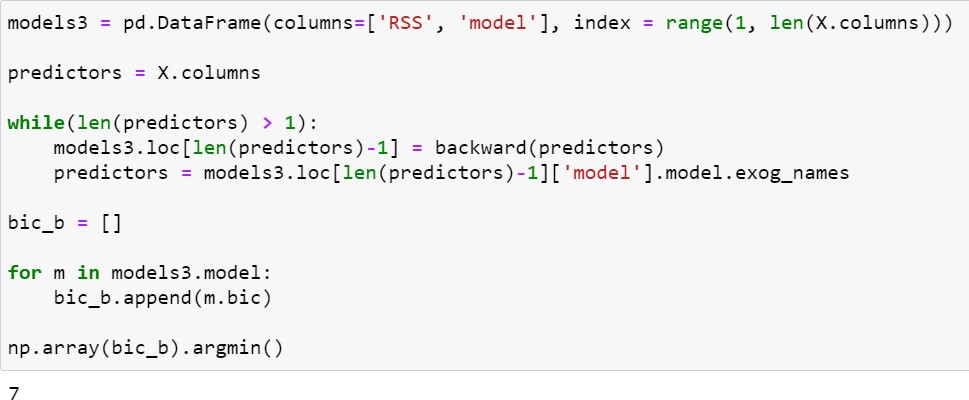


Independent variable

©copyright CIEL 2015 **69**

# 6.5.2 Forward and Backward Stepwise Selection

## Run backward stepwise selection



©copyright CIEL 2015 **70**

**[P] 6.5.3 Choosing Among Models Using the Validation Set Approach and Cross-Validation**

* + Forward stepwise selection using validation set approach

Sum of squared test errors



Based on test set

©copyright CIEL 2015 **71**

**[P] 6.5.3 Choosing Among Models Using the Validation Set Approach and Cross-Validation**

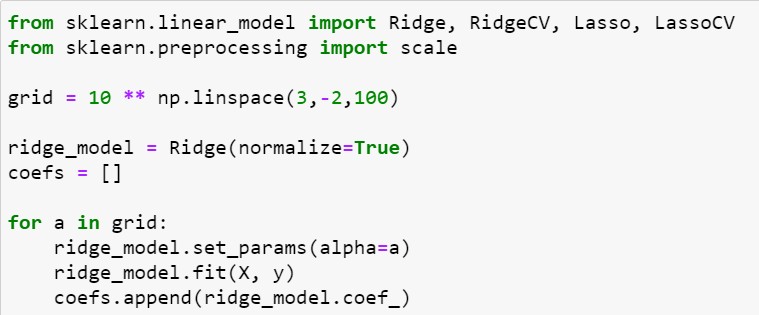
* + Forward stepwise selection using validation set approach



©copyright CIEL 2015 **72**

# 6.6.1 Ridge Regression

## Apply ridge regression



Linear least squares with l2 regularization.

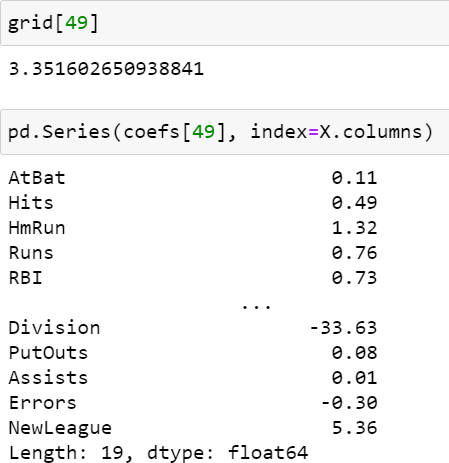
This parameter is ignored when fit\_intercept is set to False. If True, the regressors X will be normalized before regression by subtracting the mean and dividing by the l2-norm. If you wish to standardize, please use StandardScaler before calling fit on an estimator with normalize=False.

In textbook, tuning parameter

©copyright CIEL 2015 **73**

# 6.6.1 Ridge Regression

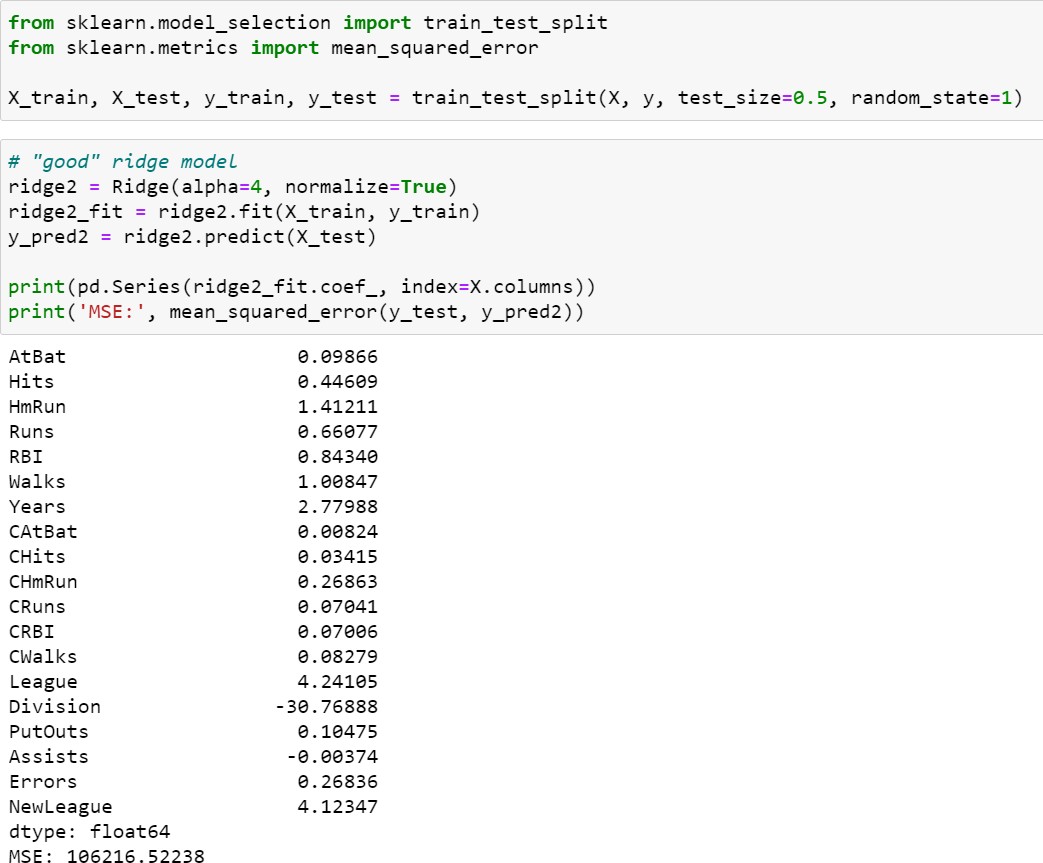
## Results of ridge regression



©copyright CIEL 2015 **74**

# 6.6.1 Ridge Regression

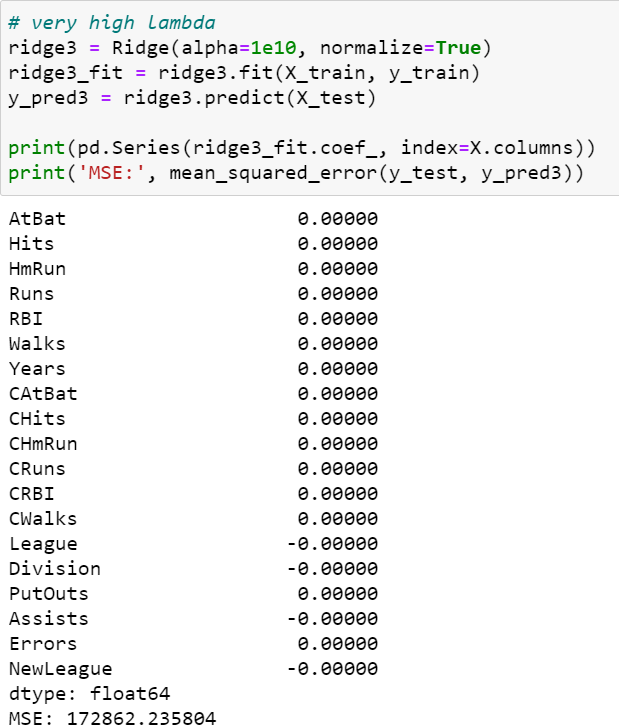
* + Good ridge model using validation set approach



©copyright CIEL 2015 **75**

# 6.6.1 Ridge Regression

## Very high lambda



©copyright CIEL 2015 **76**

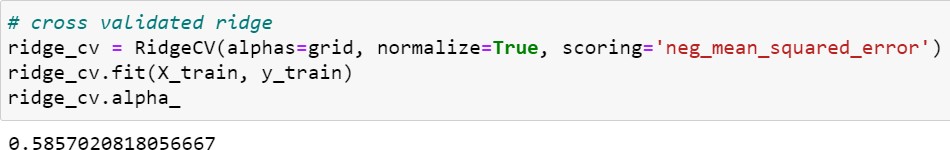
# 6.6.1 Ridge Regression

## Lambda = 0  least square regression

©copyright CIEL 2015 **77**

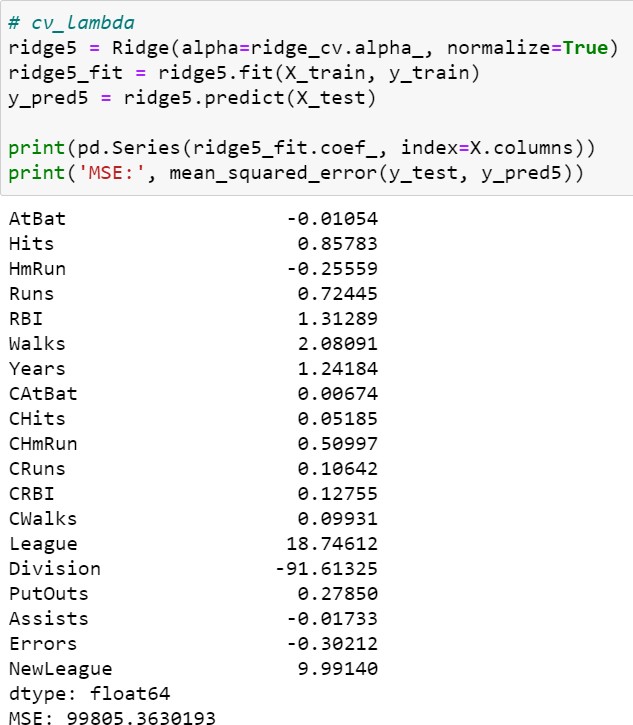
# 6.6.1 Ridge Regression

## Cross-validated ridge



Ridge regression with built-in cross-validation

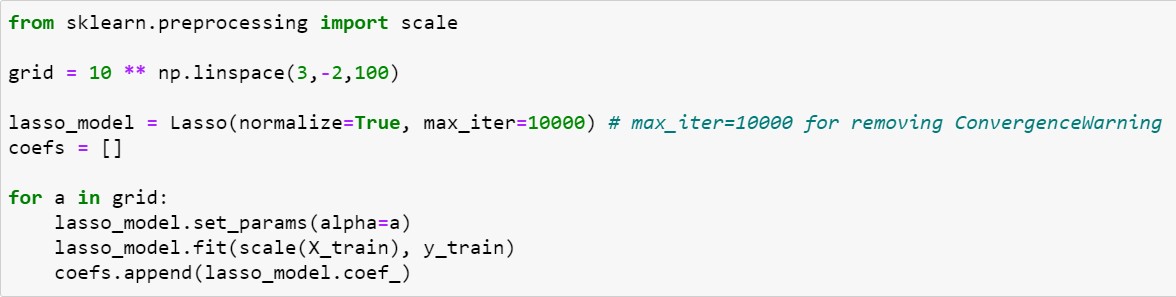
By default, it performs efficient Leave-One-Out Cross-Validation.



©copyright CIEL 2015 **78**

# 6.6.2 The Lasso

## Results of Lasso

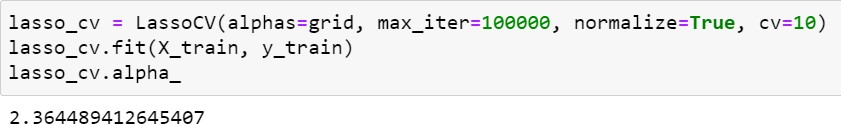


Linear Model trained with L1 prior as regularizer (aka the Lasso)

©copyright CIEL 2015 **79**

# 6.6.2 The Lasso

## Cross-validated Lasso (10-fold)



The best model is selected by cross-validation By default, it performs 5-fold Cross-Validation.

But, herein, 10-fold is applied

©copyright CIEL 2015 **80**

# 6.7.1 Principal Components Regression

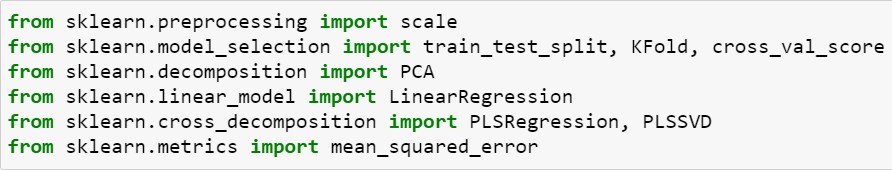
* + Scikitlearn for principle component regression



Fit the model with X and apply the dimensionality reduction on X

Standardize a dataset along any axis

Principal component analysis (PCA).



Partial Least Square SVD.

This transformer simply performs a SVD on the crosscovariance matrix X’Y.

Linear dimensionality reduction using Singular Value Decomposition of the data to project it to a lower dimensional space. The input data is centered but not scaled for each feature before applying the SVD

PLS regression

PLSRegression is also known as PLS2 or PLS1, depending on the number of targets



©copyright CIEL 2015 **81**

# 6.7.1 Principal Components Regression

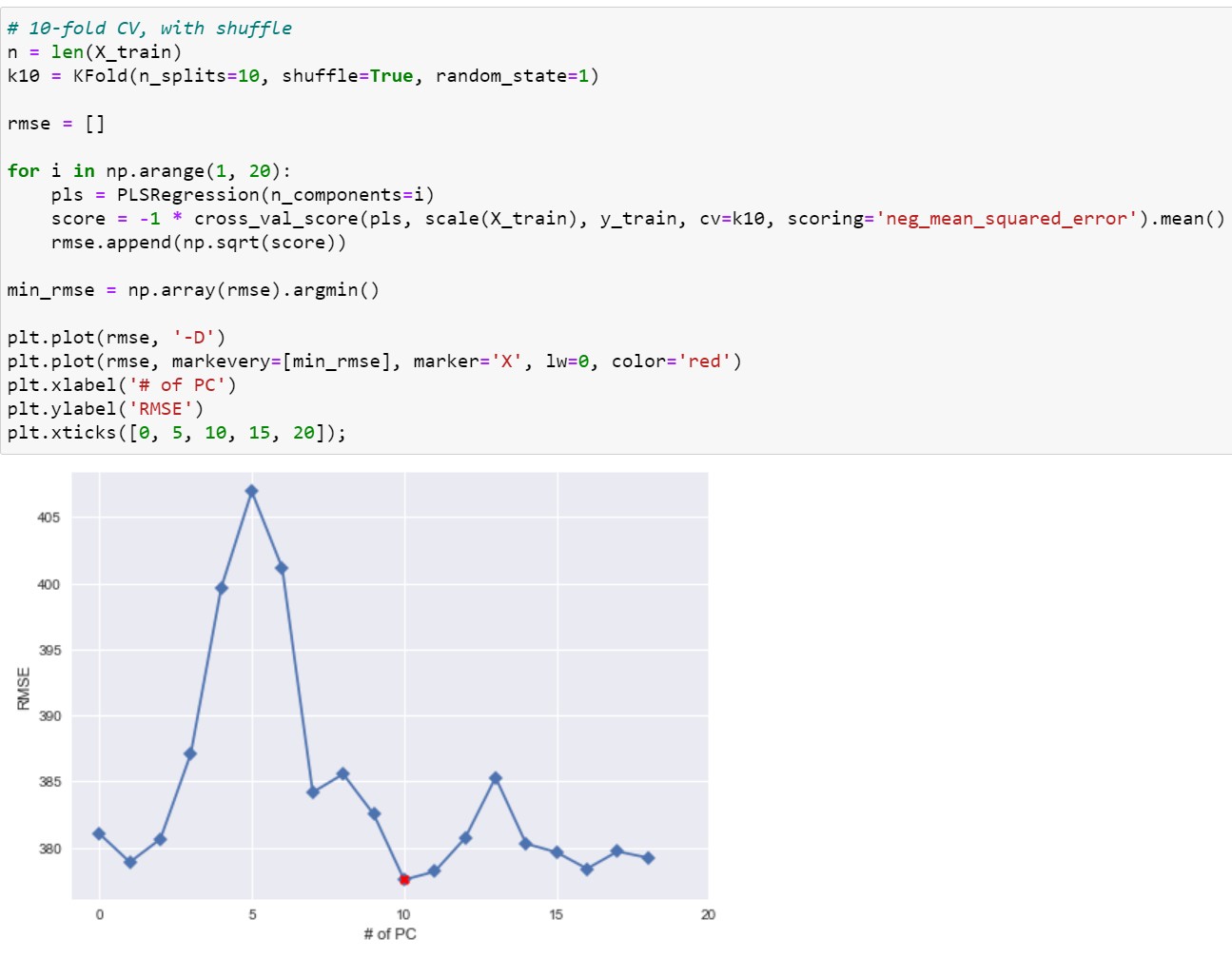
* + Selection of number of principle components based on 10-fold CV

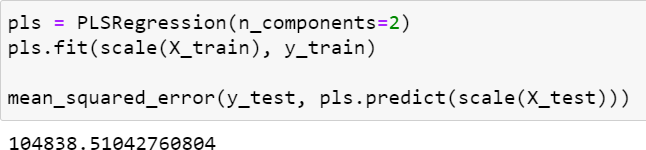
©copyright CIEL 2015

# 6.7.2 Partial Least Squares

* + Partial least squares with 10-fold CV

PLS regression





©copyright CIEL 2015



**Summary & Next Class**

* + - Linear model selection and regularization
    - Python lab
    - Summary & Next class

©copyright CIEL 2015 https://ciel.seoultech.ac.kr/

# Summary

* + Linear model selection and regularization
* Subset selection: best subset selection, forward stepwise selection, backward stepwise selection

o 𝐶𝐶𝑝𝑝, AIC, BIC

* Shrinkage methods: ridge regression, lasso
* Dimension reduction methods: principle component regression, partial least squares
* Model selection methods are an essential tool for data analysis, especially for big datasets involving many predictors
* Research into methods that give sparsity, such as the lasso is an especially hot area
  + Python lab
* Implementing forward/backward stepwise selection
* sklearn.linear\_model.Ridge, sklearn.linear\_model.RidgeCV
* sklearn.linear\_model.Lasso, sklearn.linear\_model.LassoCV
* sklearn.decomposition.PCA, sklearn.cross\_decomposition.PLSRegression

©copyright CIEL 2015 **85**

# Assignments

* + eClass > Assignments
* Upload files (do not compress them)
  + Python practices in today’s lecture
* Upload a single ipynb file
* Referring to the lecture slides marked with [P]
* File name: “StudentID” + “\_AssignmentNo w/ 2 digits” + “\_1.ipynb”, e.g., **20211234\_02\_1.ipynb**
  + Textbook exercise problems for today’s lecture
* Conceptual
  + Solving the given problems, then upload your own solution (only docx/hwp formats, not pdf/jpg/bmp etc.)
  + Only include your answers (not need to describe problems)
  + File name: “StudentID” + “\_AssignmentNo w/ 2 digits” + “\_2.ipynb”, e.g., **20211234\_02\_2.docx**
* Applied
  + Implement your Python code for the given problems, then upload another single ipynb file
  + File name: “StudentID” + “\_AssignmentNo w/ 2 digits” + “\_1.ipynb”, e.g., **20211234\_02\_3.ipynb**
  + If not complying with the above policies, some penalty on assignment scores may be given.

©copyright CIEL 2015 **86**

# Course Schedule (Tentative)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Week** | **Topics** | **Note** | **Date (W)** | **Date (M)** |
| 1 | Orientation, Statistical Learning (Ch2) | Online | 03/03 | 03/08 |
| 2 | Statistical Learning (Ch2), Python Programming | Online | 03/10 | 03/15 |
| 3 | Probability & Statistics | Online | 03/17 | 03/22 |
| 4 | Probability & Statistics | Online | 03/24 | 03/29 |
| 5 | Linear Regression (Ch3) | Online | 03/31 | 04/05 |
| 6 | Linear Regression (Ch3) | Online | 04/07 | 04/12 |
| 7 | Classification (Ch4) | Online | 04/14 | 04/19 |
| 8 | **Midterm exam** | **Class hours (W1-W7)** | **04/21** | **04/26** |
| 9 | Resampling Methods (Ch5) | Online | 04/28 | 05/03 |
| 10 | Linear Model Selection and Regularization (Ch6) | Online | 05/05 | 05/10 |
| **11** | Moving Beyond Linearity (Ch7) | Online | 05/12 | 05/17 |
| 12 | Tree-Based Methods (Ch8) | Online | 05/19 | 05/24 |
| 13 | Support Vector Machines (Ch9) | Online | 05/26 | 05/31 |
| 14 | Unsupervised Learning (Ch10) | Online | 06/02 | 06/07 |
| 15 | **Final exam** | **7pm or Class hours (W9-W14)** | **06/09or14** | **06/09or14** |

©copyright CIEL 2015 **87**