Data Analysis 2021 Spring





**Lecture 07: Classification**

April 14 & April 19, 2021

**Taesoo Kwon**

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

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

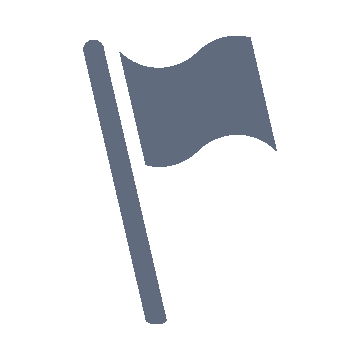
# Course Schedule (Tentative)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **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**



* Classification



**OUTLINES**

* + Logistic regression
  + Linear discriminant analysis (LDA)
  + Quadratic discriminant analysis (QDA)
* Python lab
* Summary & Next class

©copyright CIEL 2015



**Classification**

**: Ch 4**

## Classification

* Python lab
* Summary & Next class

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

# Classification

* Qualitative variables take values in an unordered set *C*, such as:
  + eye color ∈ {brown, blue, green}
  + email ∈ {spam, ham}
* Given a feature vector 𝑋𝑋 and a qualitative response 𝑌𝑌 taking values in the set *C*, the classification task is to build a function 𝐶𝐶(𝑋𝑋) that takes as input the feature vector 𝑋𝑋 and predicts its value for

𝑌𝑌 ; i.e., 𝐶𝐶(𝑋𝑋) ∈ *C*.

* Often we are more interested in estimating the probabilities that 𝑋𝑋 belongs to each category in

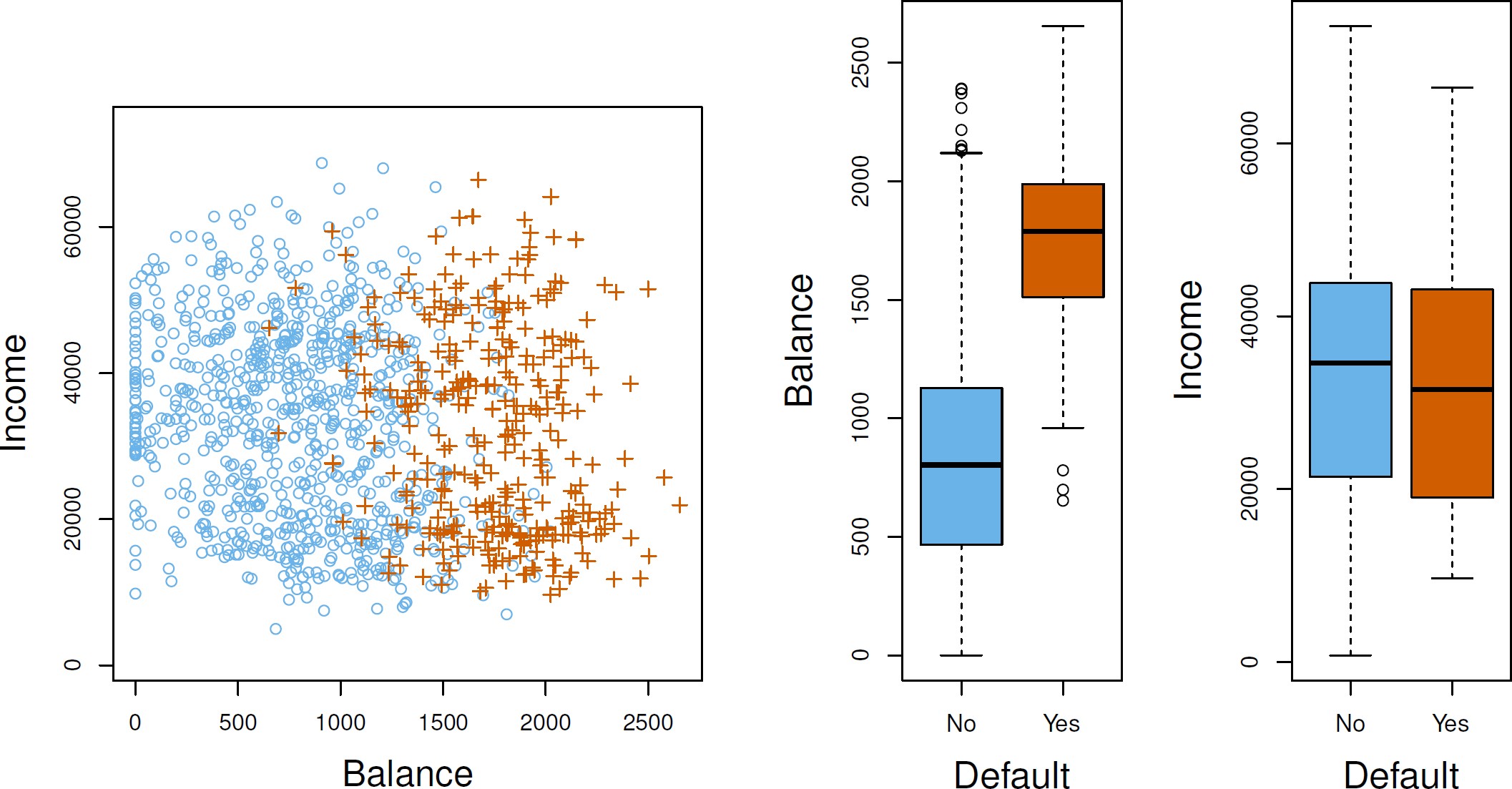
*C*

* + For example, it is more valuable to have an estimate of the probability that an insurance claim is fraudulent, than a classification fraudulent or not.

©copyright CIEL 2015 **5**

# Example: Credit Card Default

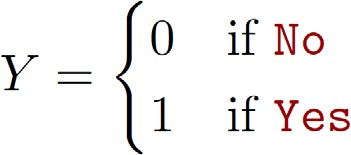
Not defaulted Defaulted



©copyright CIEL 2015 **6**

# Can We Use Linear Regression?

* Suppose for the Default classification task that we code



* Can we simply perform a linear regression of 𝑌𝑌 on 𝑋𝑋 and classify as Yes if

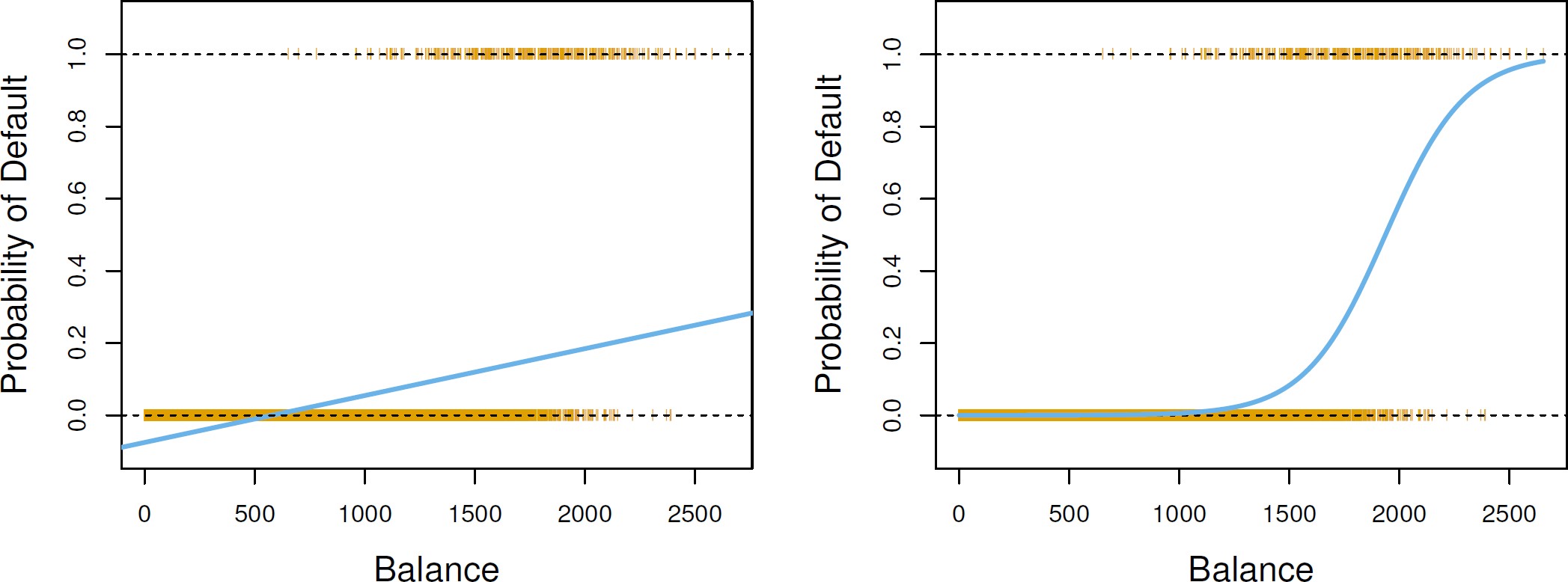
𝑌𝑌�

> 0.5?

* + In this case of a binary outcome, linear regression does a good job as a classier, and is equivalent to linear discriminant analysis which we discuss later
  + Since in the population 𝐸𝐸 𝑌𝑌 𝑋𝑋 = 𝑥𝑥 = Pr(𝑌𝑌 = 1|𝑋𝑋 = 𝑥𝑥), we might think that regression is perfect for this task
  + However, linear regression might produce probabilities less than zero or bigger than one
  + Logistic regression is more appropriate.

©copyright CIEL 2015 **7**

# Linear versus Logistic Regression

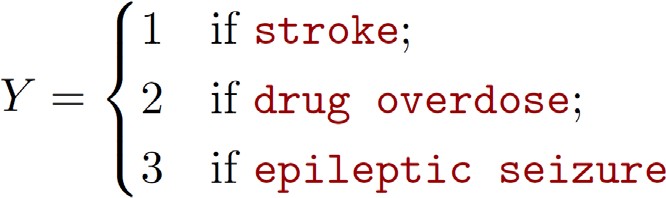


* The orange marks indicate the response 𝑌𝑌, either 0 or 1
* Linear regression does not estimate Pr(𝑌𝑌 = 1|𝑋𝑋) well
* Logistic regression seems well suited to the task.

©copyright CIEL 2015 **8**

# Linear Regression Continued

* Now suppose we have a response variable with three possible values
  + A patient presents at the emergency room, and we must classify them according to their symptoms



* + This coding suggests an ordering, and in fact implies that the difference between stroke and drug overdose is the same as between drug overdose and epileptic seizure
* Linear regression is not appropriate here Multiclass Logistic Regression or Discriminant Analysis are more appropriate.

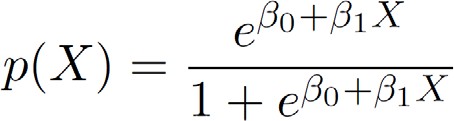
©copyright CIEL 2015 **9**

# Logistic Regression

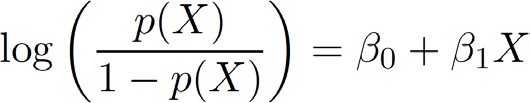
* Let’s write 𝑝𝑝 = Pr(𝑌𝑌 = 1|𝑋𝑋) for short and consider using balance to predict default

𝑋𝑋

* Logistic regression uses the form



* + 𝑒𝑒 = 2.71828 is a mathematical constant [Euler's number]
  + It is easy to see that no matter what values 𝛽𝛽0, 𝛽𝛽1 or 𝑋𝑋 take, 𝑝𝑝(𝑋𝑋) will have values between 0 and 1
* A bit of rearrangement gives

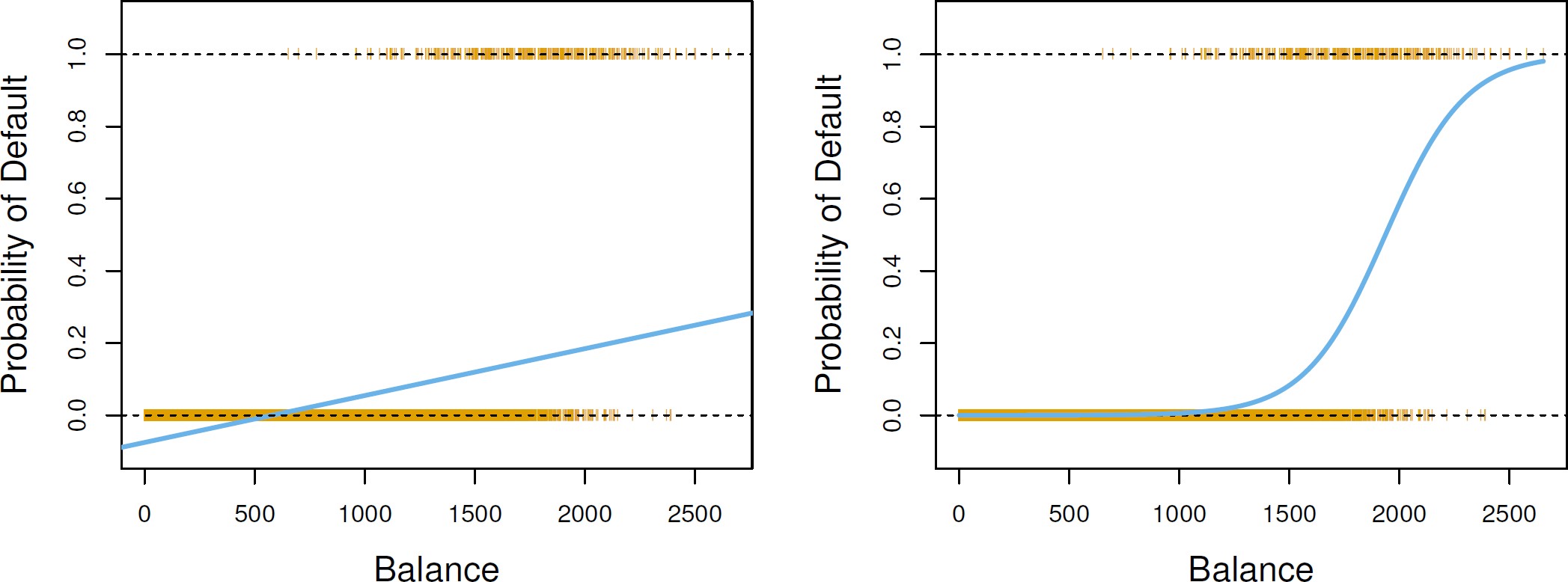


* + This monotone transformation is called the log odds or logit transformation of p(X)

o by log we mean natural log: ln

©copyright CIEL 2015 **10**

# Linear versus Logistic Regression

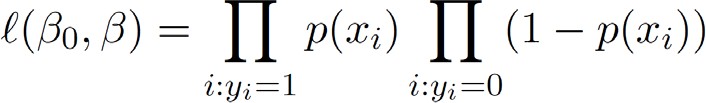


* Logistic regression ensures that our estimate for 𝑝𝑝(𝑋𝑋) lies between 0 and 1

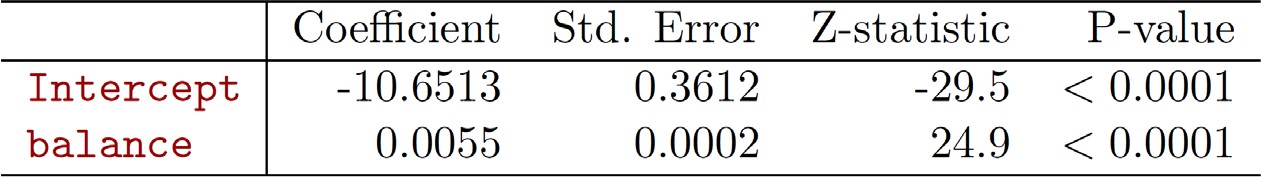
©copyright CIEL 2015 **11**

# Maximum Likelihood

* We use maximum likelihood to estimate the parameters



* + This likelihood gives the probability of the observed zeros and ones in the data
  + We pick 𝛽𝛽0 and 𝛽𝛽1 to maximize the likelihood of the observed data
* Most statistical packages can fit linear logistic regression models by maximum likelihood



Statistic for random variable for 𝛽𝛽�

𝑆𝑆𝑆𝑆

©copyright CIEL 2015 **12**

# [Review] Point Estimates: Maximum Likelihood Estimators

* Estimator of 𝜃𝜃
  + Any statistic used to estimate the value of an unknown parameter 𝜃𝜃
  + E.g., usual estimator of the mean of a normal population: sample mean

𝑋𝑋�

= ∑𝑖𝑖 𝑋𝑋𝑖𝑖 /𝑛𝑛

* Estimate
  + Observed value of the estimator
  + E.g., a sample of size 3 yields the data 𝑋𝑋1 = 2, 𝑋𝑋2 = 3, 𝑋𝑋3 = 4, then estimate of population mean is 3
* Maximum likelihood estimate 𝜃𝜃̂, maximum likelihood estimator

𝑥𝑥1, ⋯ , 𝑥𝑥𝑛𝑛 𝜃𝜃

* + 𝑓𝑓

: joint pmf (discrete RV) or pdf (continuous RV) of the random variables

* + - Representing the likelihood that the values 𝑥𝑥1, 𝑥𝑥2, ⋯ , 𝑥𝑥𝑛𝑛 will be observed when 𝜃𝜃 is the true value of parameter
  + 𝜃𝜃̂: value of 𝜃𝜃 maximizing 𝑓𝑓 where 𝑥𝑥1, ⋯ , 𝑥𝑥𝑛𝑛 are the observed values

𝑥𝑥1, ⋯ , 𝑥𝑥𝑛𝑛 𝜃𝜃

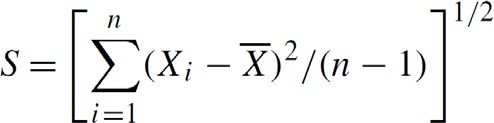
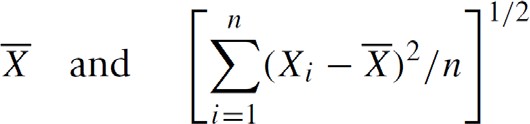
* + Sometimes, log 𝑓𝑓 (log-likelihood function) is useful

𝑥𝑥1, ⋯ , 𝑥𝑥𝑛𝑛 𝜃𝜃

©copyright CIEL 2015 **13**

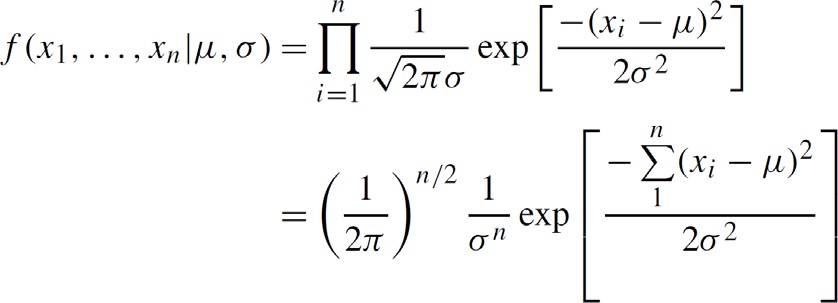
# [Review] Point Estimates: Maximum Likelihood Estimators [cont.]

* Example: maximum likelihood estimator of a normal population
  + normal random variables each with unknown mean 𝜇𝜇 and unknown standard deviation 𝜎𝜎2
  + Solution: cf.

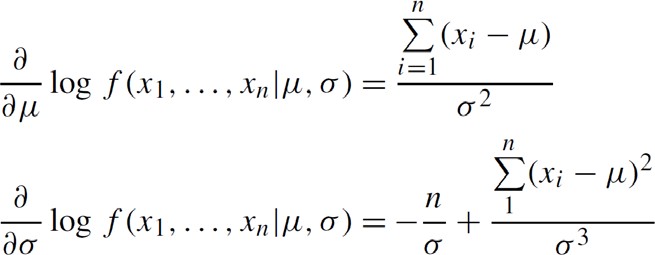


* + Proof

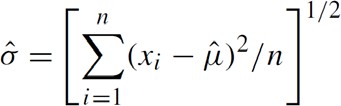
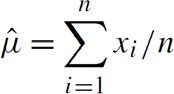
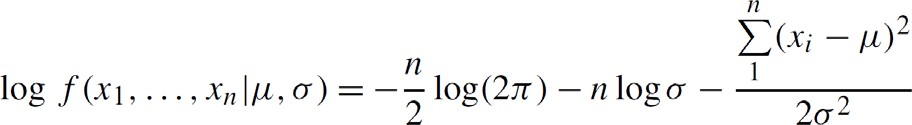
= 0



Log-concave



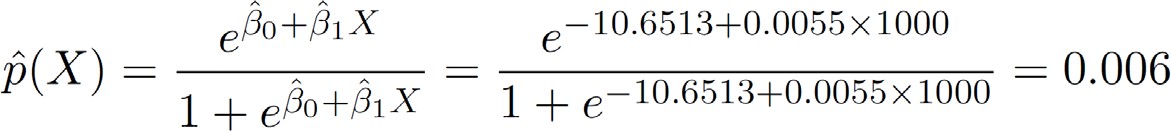
= 0



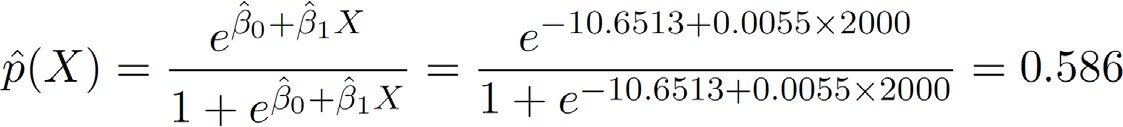
©copyright CIEL 2015 **14**

# Making Predictions

* What is our estimated probability of default for someone with a balance of $1000?



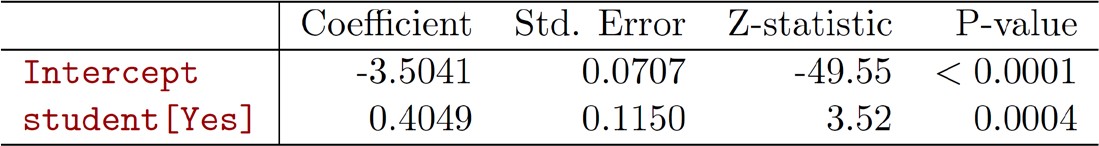
* + With a balance of $2000

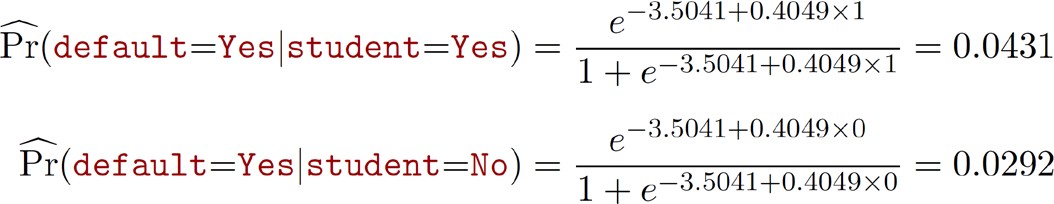


©copyright CIEL 2015 **15**

# Making Predictions [cont.]

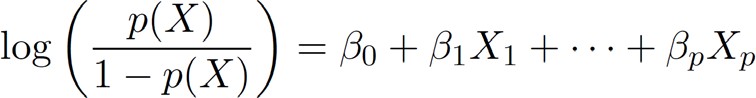
* Lets do it again, using student as the predictor

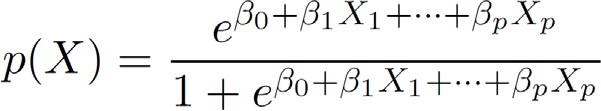


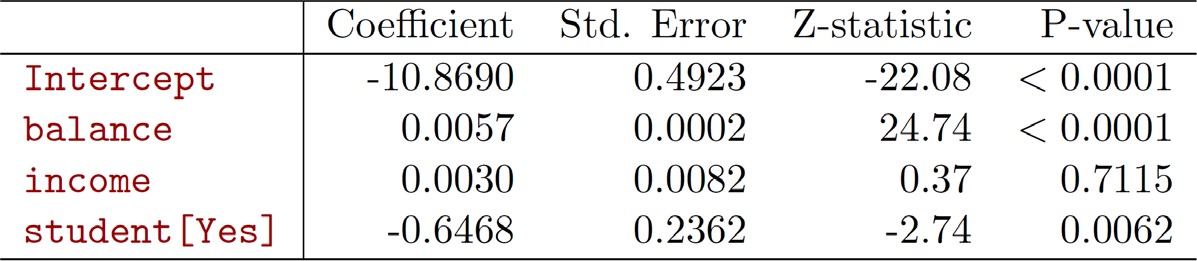


©copyright CIEL 2015 **16**

# Logistic Regression with Several Variables



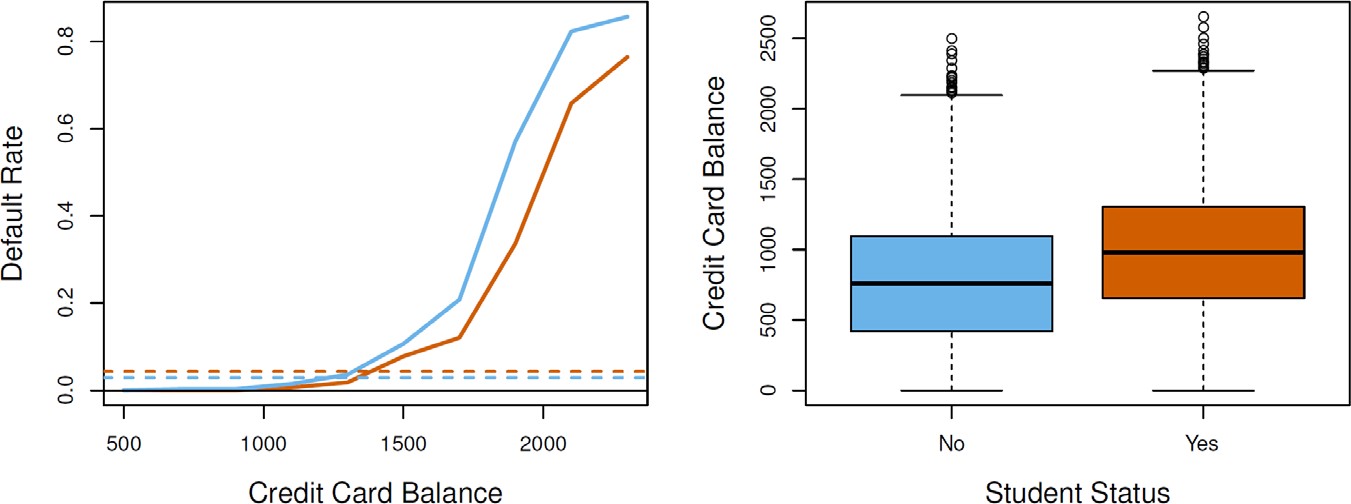




* Why is coefficient for student negative, while it was positive before?

©copyright CIEL 2015 **17**

# Confounding

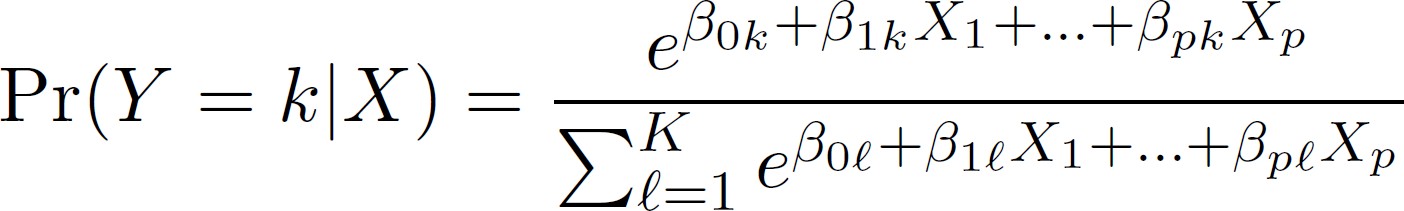


* Students tend to have higher balances than non-students, so their marginal default rate is higher than for non-students
* But for each level of balance, students default less than non-students.
* Multiple logistic regression can tease this out

©copyright CIEL 2015 **18**

# Logistic Regression with More than Two Classes

* So far we have discussed logistic regression with two classes
* It is easily generalized to more than two classes



* Here there is a linear function for each class
* Not to be used all that often
  + Discriminant analysis is more popular for multiple-class classification

©copyright CIEL 2015 **19**

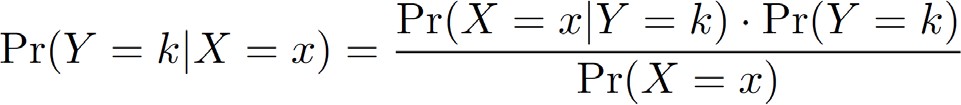
# Discriminant Analysis

* Here the approach is to model the distribution of 𝑋𝑋 in each of the classes separately, and then use Bayes theorem to flip things around and obtain Pr(𝑌𝑌|𝑋𝑋)
* When we use normal (Gaussian) distributions for each class, this leads to linear or quadratic discriminant analysis
* However, this approach is quite general, and other distributions can be used as well
* We will focus on normal distributions

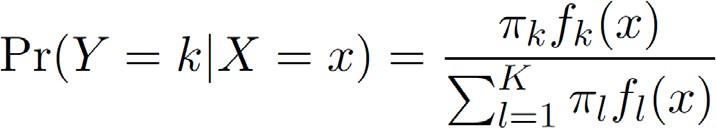
©copyright CIEL 2015 **20**

# Bayes Theorem for Classification

* Thomas Bayes was a famous mathematician whose name represents a big subfield of statistical and probabilistic modeling
* Here we focus on a simple result, known as Bayes theorem:



* One writes this slightly differently for discriminant analysis:



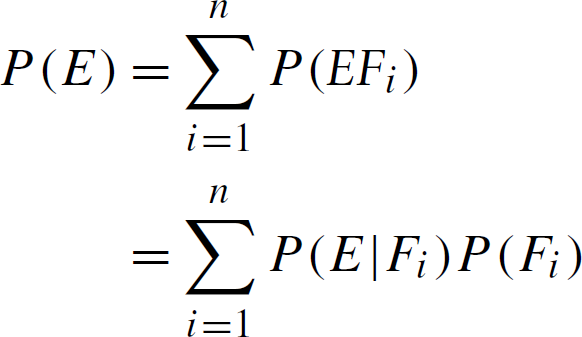
* + 𝑓𝑓𝑘𝑘 = Pr(𝑋𝑋 = 𝑥𝑥|𝑌𝑌 = 𝑦𝑦) is the density for 𝑋𝑋 in class 𝑘𝑘

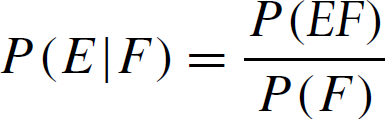
𝑥𝑥

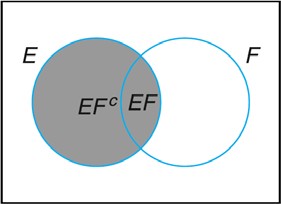
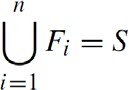
* + - Here we will use normal densities for these, separately in each class
  + 𝜋𝜋𝑘𝑘 = Pr(𝑌𝑌 = 𝑘𝑘)is the marginal or prior probability for class 𝑘𝑘

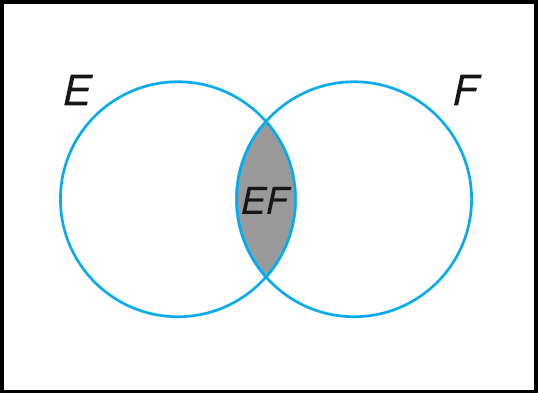
©copyright CIEL 2015 **21**

# [Review] Conditional Probability and Bayes’ formula

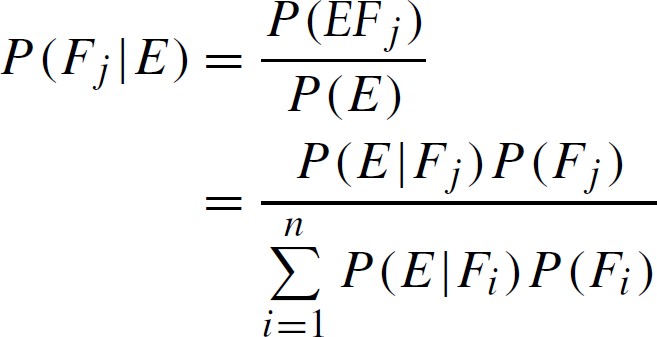
* Conditional Probability
  + Probability of *E* given that *F* has occurred



* Bayes’ formula
  + From conditioning when ,



©copyright CIEL 2015 **22**

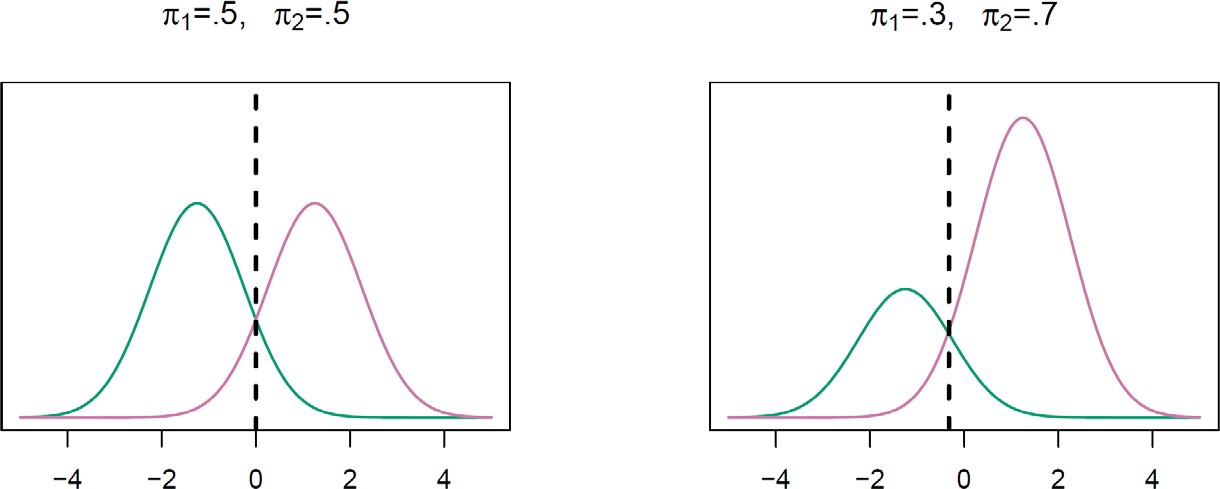


Prior

Posterior

Likelihood

# Classify to Highest Density



* We classify a new point according to which density is highest
* When the priors are different, we take them into account as well, and compare 𝜋𝜋𝑘𝑘 𝑓𝑓𝑘𝑘 (𝑥𝑥)
  + On the right, we favor the pink class  the decision boundary has shifted to the left

©copyright CIEL 2015 **23**

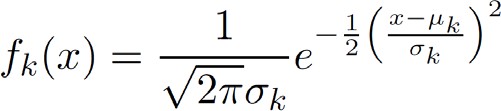
# Why Discriminant Analysis?

* When the classes are well-separated, the parameter estimates for the logistic regression model are surprisingly unstable. Linear discriminant analysis does not suffer from this problem.
* If 𝑛𝑛 is small and the distribution of the predictors 𝑋𝑋 is approximately normal in each of the classes, the linear discriminant model is again more stable than the logistic regression model
* Linear discriminant analysis is popular when we have more than two response classes, because it also provides low-dimensional views of the data.

©copyright CIEL 2015 **24**

**Linear Discriminant Analysis when** 𝒑𝒑 = 𝟏𝟏

* The Gaussian density has the form

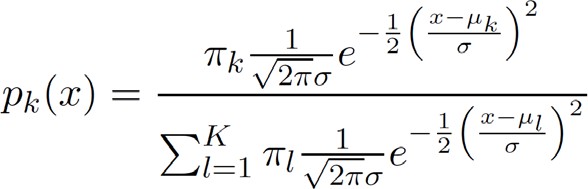


* + Here 𝜇𝜇𝑘𝑘 is the mean, and 𝜎𝜎2 the variance (in class 𝑘𝑘)
    - We will assume that all the 𝜎𝜎𝑘𝑘 = 𝜎𝜎 are the same

𝑘𝑘

* Plugging this into Bayes formula, we get a rather complex expression for 𝑝𝑝𝑘𝑘

𝑥𝑥

𝑘𝑘):

* + Happily, there are simplifications and cancellations

= Pr(𝑌𝑌 = 𝑘𝑘|𝑋𝑋 =

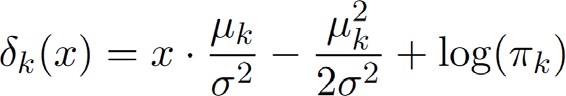


©copyright CIEL 2015 **25**

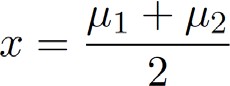
# Discriminant Functions

* To classify at the value 𝑋𝑋 = 𝑥𝑥, we need to see which of the 𝑝𝑝𝑘𝑘 (𝑥𝑥) is largest
  + Taking logs, and discarding terms that do not depend on 𝑘𝑘, we see that this is equivalent to assigning 𝑥𝑥

to the class with the largest discriminant score:

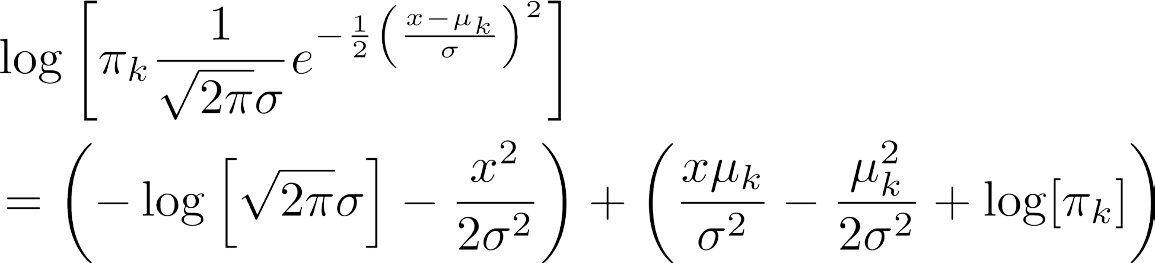


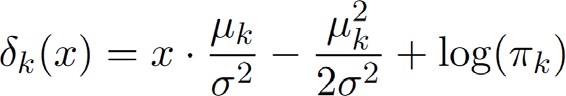
* + Note that 𝛿𝛿𝑘𝑘 is a linear function of 𝑥𝑥
* If there are 𝐾𝐾 = 2 classes and 𝜋𝜋1 = 𝜋𝜋2 = 0.5, then one can see that the decision boundary is at



©copyright CIEL 2015 **26**

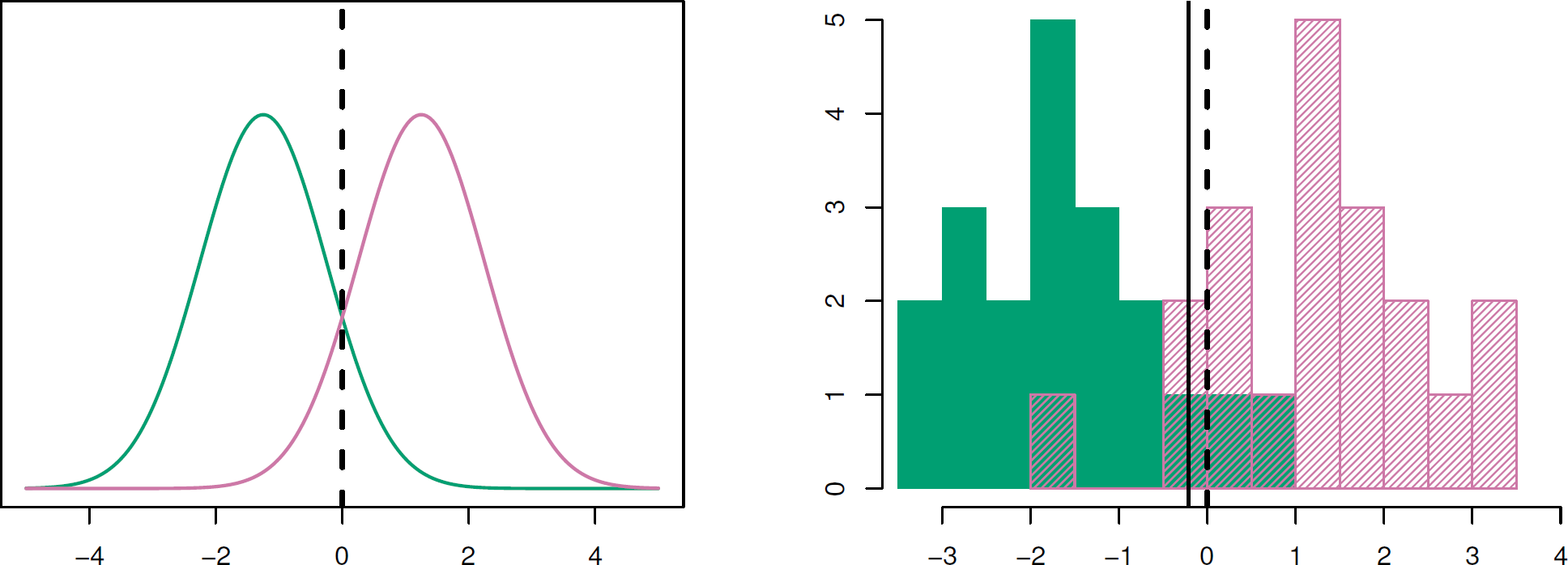
# [FYI] Logarithm of Normal PDF





©copyright CIEL 2015 **27**

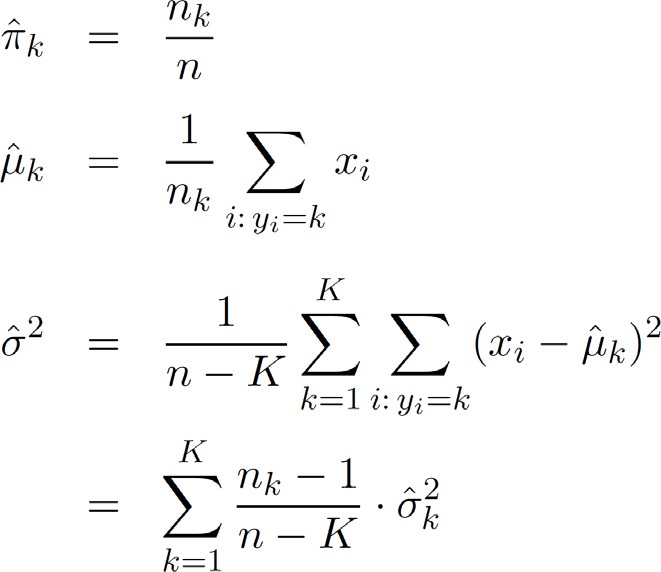
# Discriminant Functions [cont.]



* Example with 𝜇𝜇1 = −1.5, 𝜇𝜇2 = 1.5, 𝜋𝜋1 = 𝜋𝜋2= 0.5, and 𝜎𝜎2 = 1
* Typically we don't know these parameters
  + We just have the training data
  + In that case we simply estimate the parameters and plug them into the rule

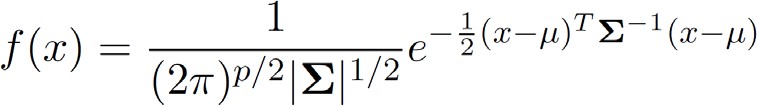
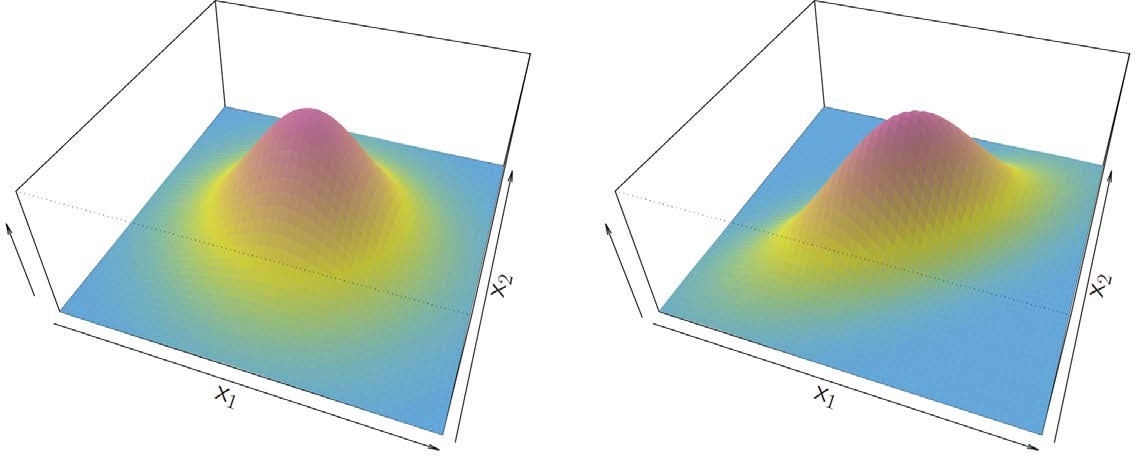
©copyright CIEL 2015 **28**

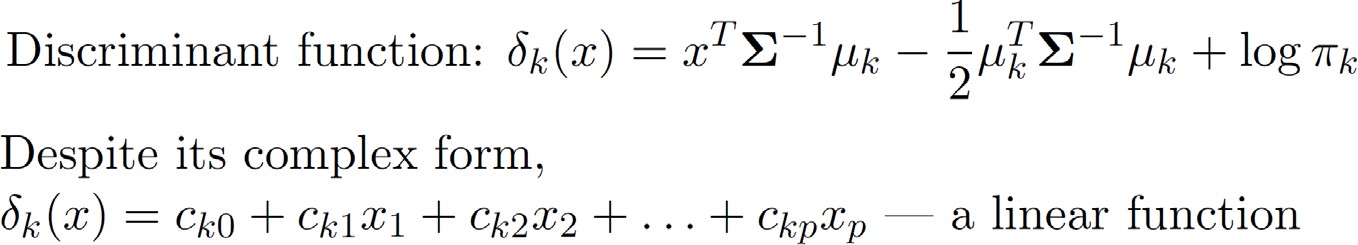
# Estimating Parameters



where  is the usual formula for the estimated variance in the 𝑘𝑘th class

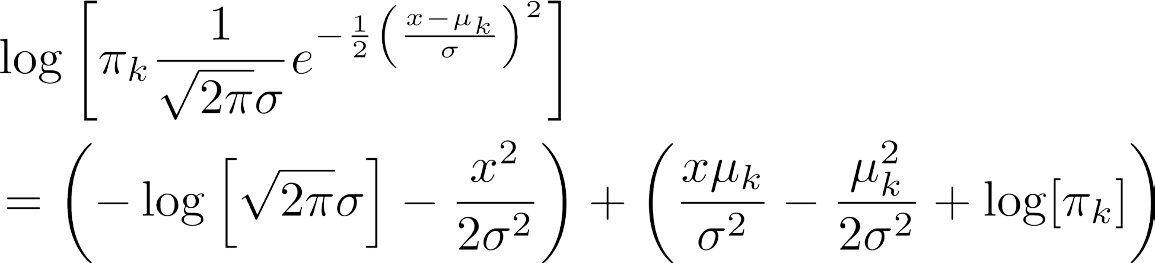
©copyright CIEL 2015 **29**

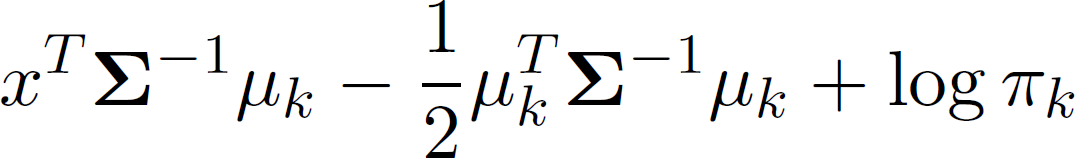
**Linear Discriminant Analysis when** 𝒑𝒑 > 𝟏𝟏



©copyright CIEL 2015 **30**

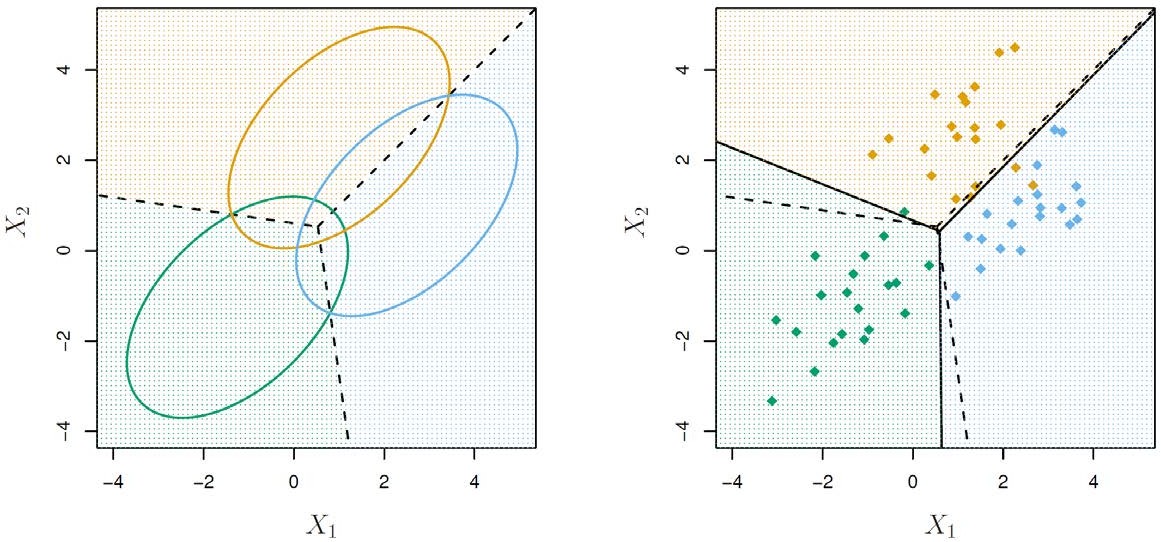
# [FYI] Logarithm of Normal PDF





©copyright CIEL 2015 **31**

**Illustration:** 𝒑𝒑 = 𝟐𝟐 **and** 𝑲𝑲 = 𝟑𝟑 **Classes**

* Here 𝜋𝜋1 = 𝜋𝜋2 = 𝜋𝜋3 =

1⁄3

* The dashed lines are known as the Bayes decision boundaries
* Were they known, they would yield the fewest misclassification errors, among all possible classifiers.

©copyright CIEL 2015 **32**

**From** 𝜹𝜹𝒌𝒌(𝒙𝒙) **to Probabilities**

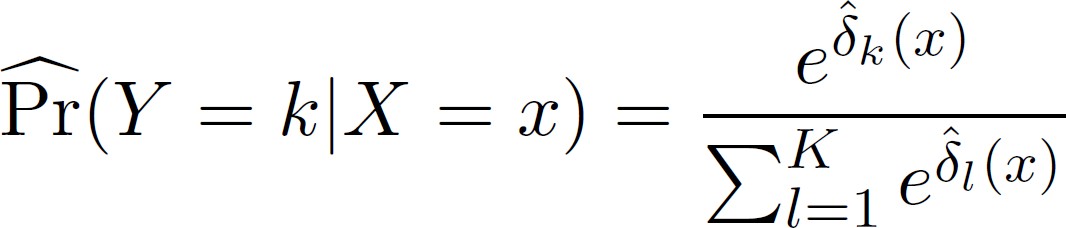
* Once we have estimates

̂

𝑘𝑘

𝛿𝛿

(𝑥𝑥), we can turn these into estimates for class probabilities:

Softmax function

* So classifying to the largest is largest

𝑥𝑥

̂

𝑘𝑘

𝛿𝛿

amounts to classifying to the class for which

P�r

𝑌𝑌 = 𝑘𝑘 𝑋𝑋 = 𝑥𝑥

* When 𝐾𝐾 = 2, we classify to class 2 if

𝑌𝑌 = 2 𝑋𝑋 = 𝑥𝑥

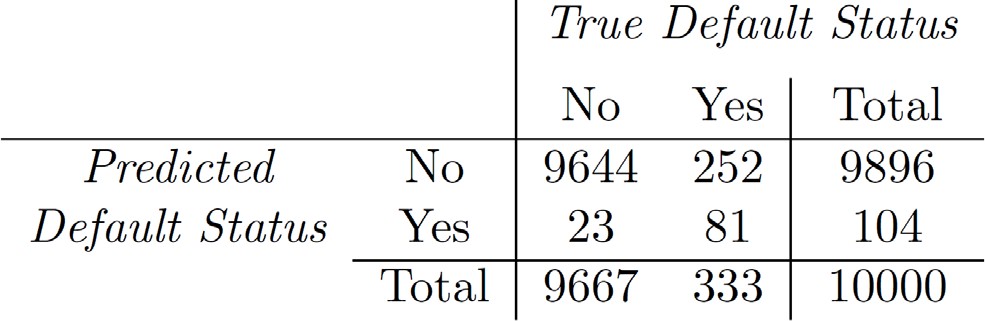
P�r

≥ 0.5, else to class 1



©copyright CIEL 2015 **33**

# LDA on Credit Data



* (23 + 252)/10000 errors  a 2.75% misclassification rate
* Some caveats
  + This is training error, and we may be overfitting
  + Not a big concern here since 𝑛𝑛 = 10000 and 𝑝𝑝 = 2
  + If we classified to the prior (always to class No in this) 333/10000 errors, or only 3.33%
  + Of the true No‘s, we make 23/9667 = 0.2% errors
  + Of true Yes ‘s, we make 252/333 = 75.7% errors

©copyright CIEL 2015 **34**

# Types of Errors

* False positive rate: The fraction of negative examples that are classified as positive
  + 0.2% in example
* False negative rate: The fraction of positive examples that are classified as negative
  + 75.7% in example
* We produced this table by classifying to class Yes if



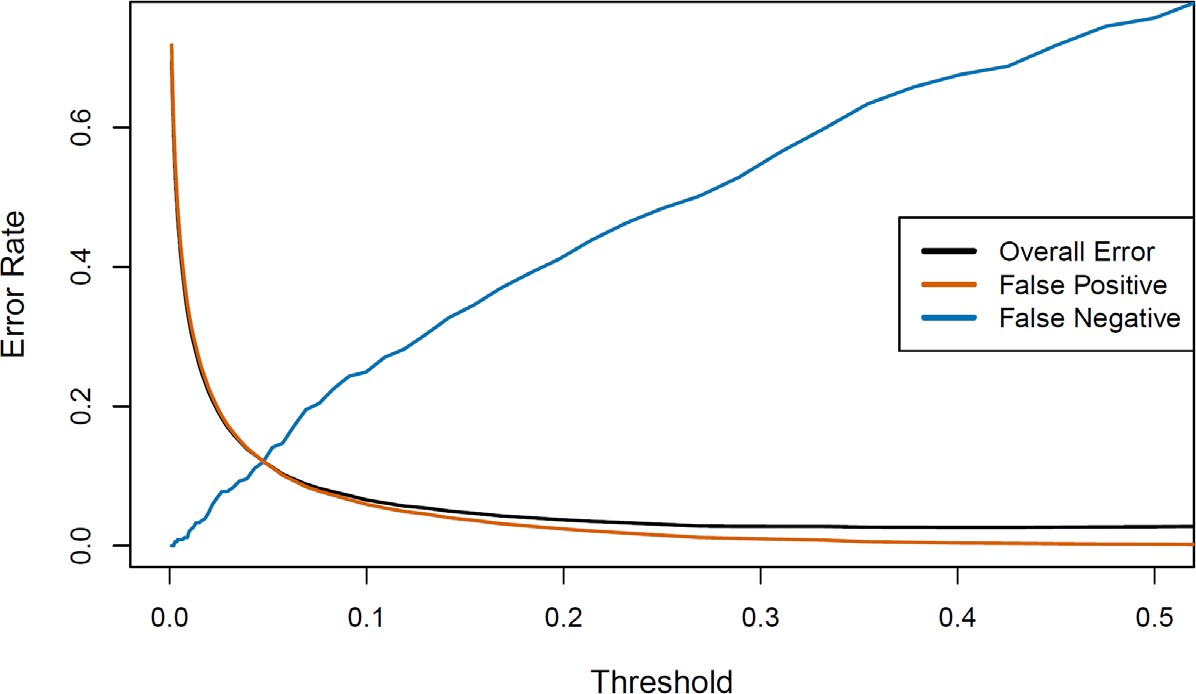
* We can change the two error rates by changing the threshold from 0.5 to some other value in [0, 1]:



* + and vary threshold

©copyright CIEL 2015 **35**

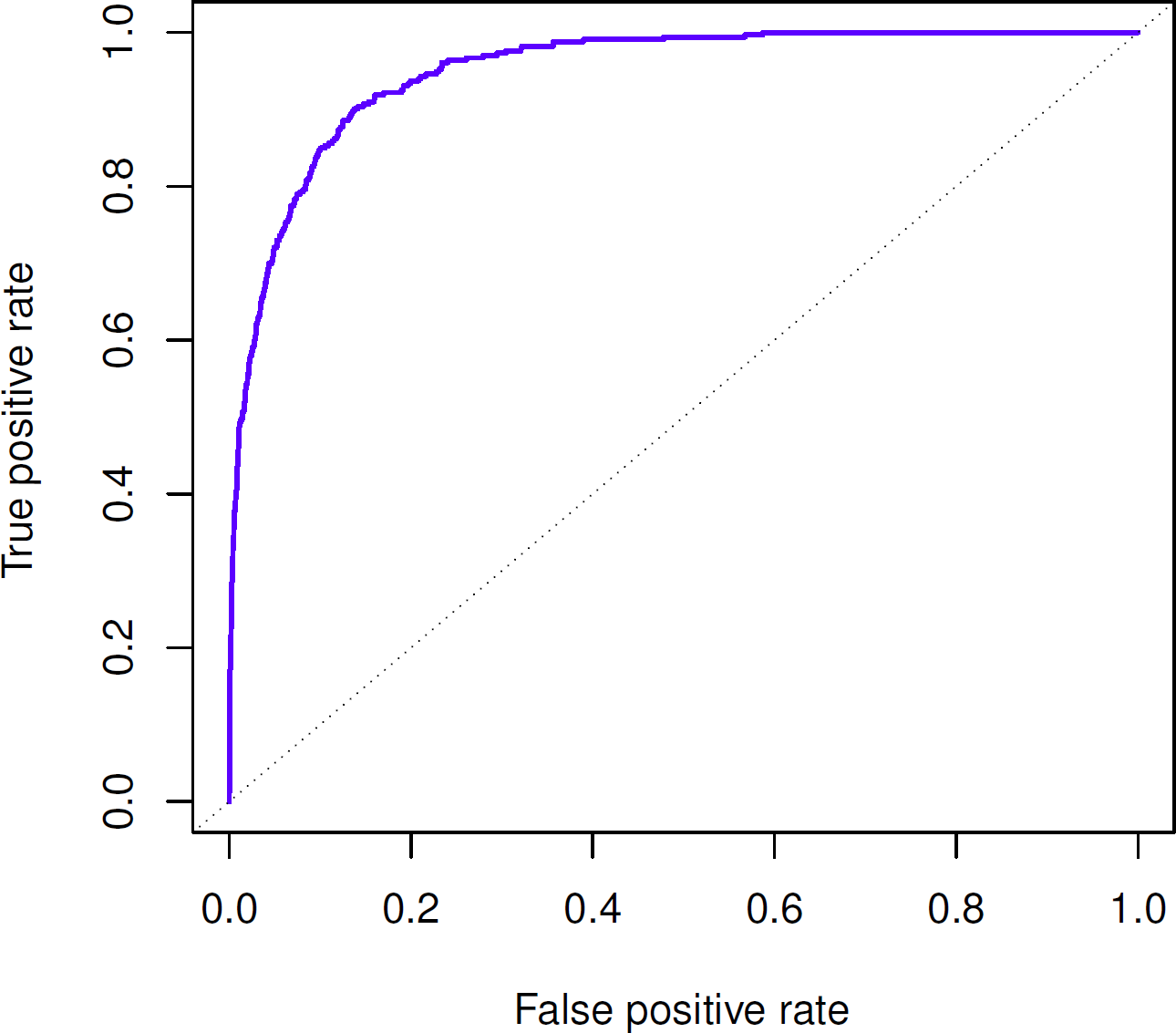
# Varying Threshold



* In order to reduce the false negative rate, we may want to reduce the threshold to 0.1 or less

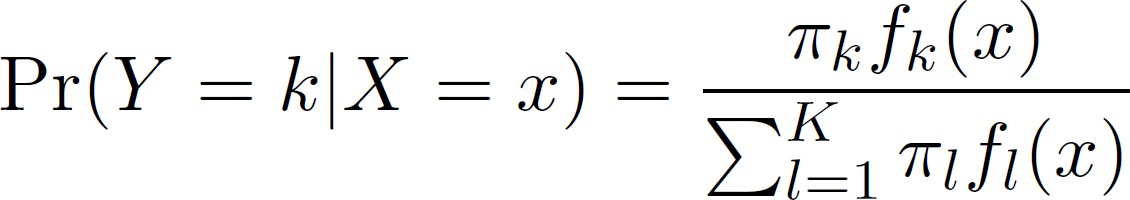
©copyright CIEL 2015 **36**

# ROC Curve

* The ROC plot displays both simultaneously
* Sometimes we use the AUC (area under the curve) to summarize the overall performance
* Higher AUC is good

©copyright CIEL 2015 **37**

# Other Forms of Discriminant Analysis



* When 𝑓𝑓𝑘𝑘 (𝑥𝑥) are Gaussian densities, with the same covariance matrix **Σ** in each class, this leads to linear discriminant analysis
* By altering the forms for 𝑓𝑓𝑘𝑘 (𝑥𝑥), we get different classifiers
  + With Gaussians but different **Σ**𝑘𝑘 in each class, we get quadratic discriminant analysis
  + With 𝑓𝑓𝑘𝑘 = ∏𝑝𝑝 𝑓𝑓𝑗𝑗𝑘𝑘(𝑥𝑥𝑗𝑗) (conditional independence model) in each class we get naive Bayes

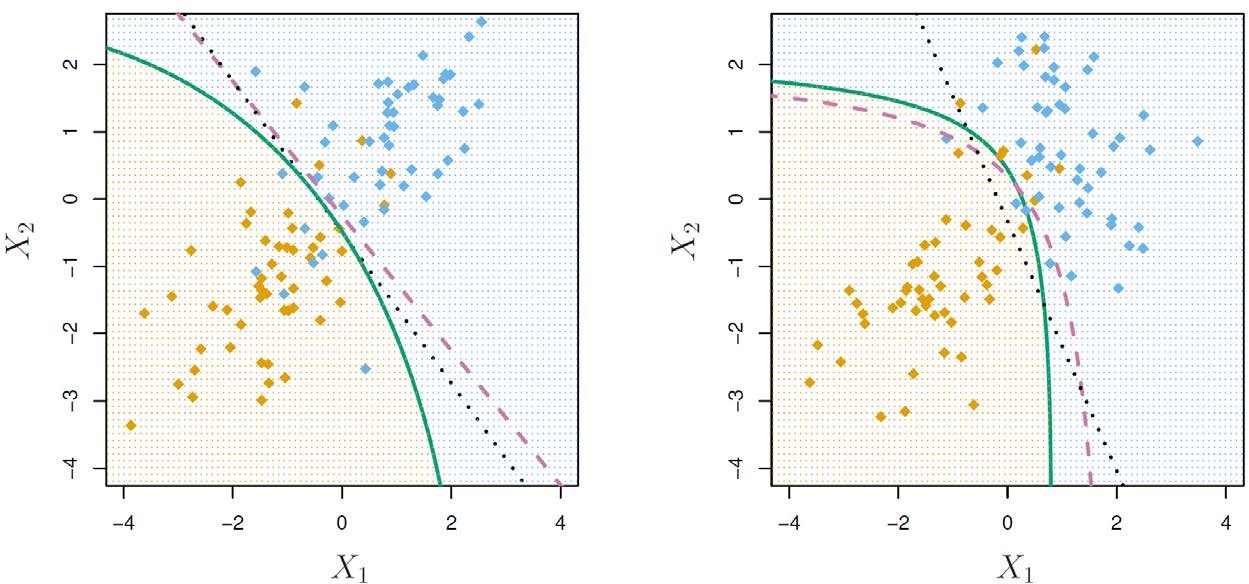
𝑥𝑥

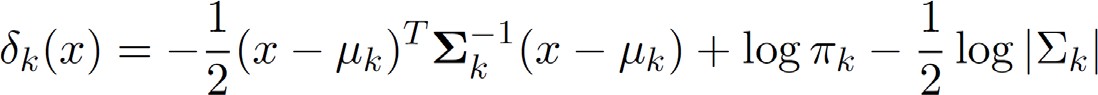
𝑗𝑗=1

* + - For Gaussian this means the **Σ**𝑘𝑘 ‘s are diagonal
  + Many other forms, by proposing specific density models for 𝑓𝑓𝑘𝑘(𝑥𝑥), including nonparametric approaches

©copyright CIEL 2015 **38**

# Quadratic Discriminant Analysis

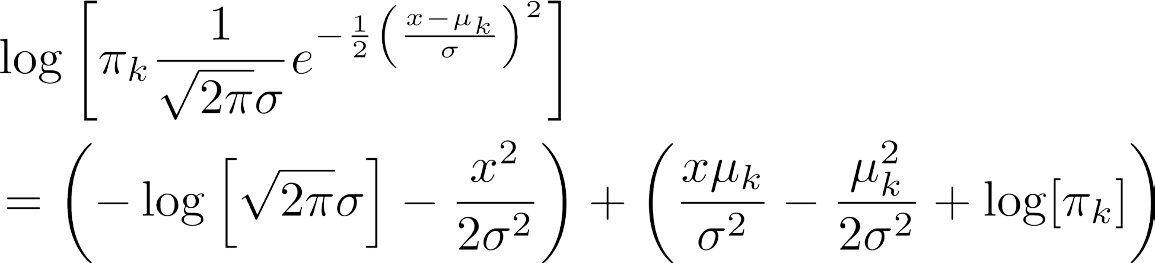


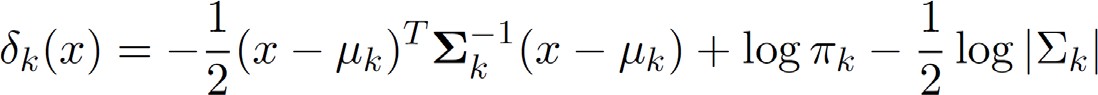


* Because the **Σ**𝑘𝑘‘s are different, the quadratic terms matter

©copyright CIEL 2015 **39**

# [FYI] Logarithm of Normal PDF

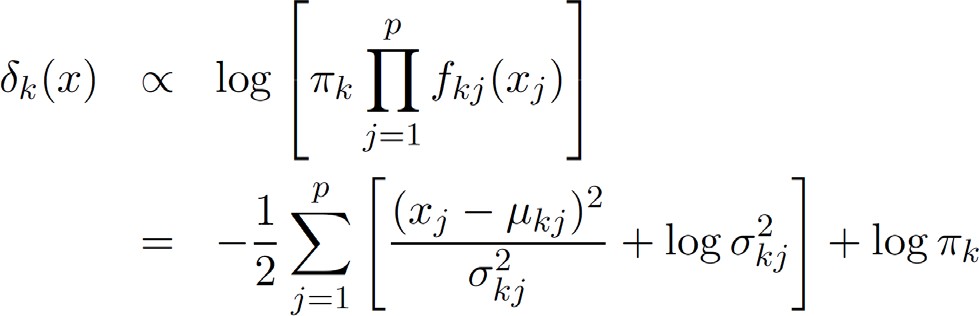




©copyright CIEL 2015 **40**

# Naive Bayes

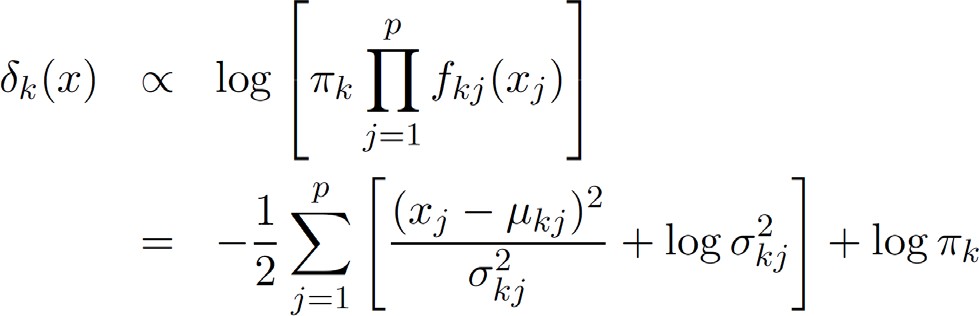
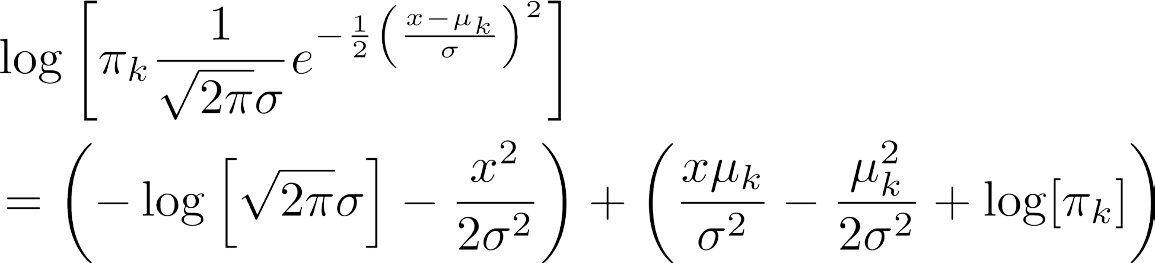
* Assumes features are independent in each class
* Useful when 𝑝𝑝 is large, and so multivariate methods like QDA and even LDA break down
  + Gaussian naive Bayes assumes each **Σ**𝑘𝑘 is diagonal:



* + Can use for mixed feature vectors (qualitative and quantitative). If 𝑋𝑋𝑗𝑗 is qualitative, replace 𝑓𝑓𝑘𝑘𝑗𝑗(𝑥𝑥𝑗𝑗) with probability mass function (histogram) over discrete categories
* Despite strong assumptions, naive Bayes often produces good classfication results

©copyright CIEL 2015 **41**

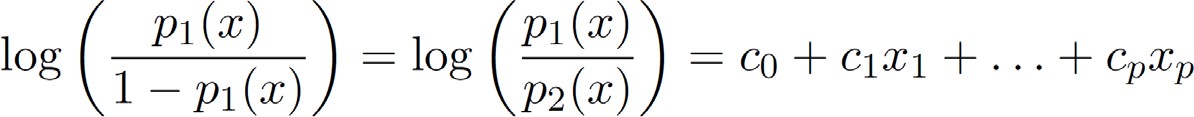
# [FYI] Logarithm of Normal PDF



©copyright CIEL 2015 **42**

# Logistic Regression versus LDA

* For a two-class problem, one can show that for LDA



* + So it has the same form as logistic regression
* The difference is in how the parameters are estimated
  + Logistic regression uses the conditional likelihood based on Pr(𝑌𝑌|𝑋𝑋) (known as discriminative learning)
  + LDA uses the full likelihood based on Pr(𝑋𝑋, 𝑌𝑌) (known as generative learning)
  + Despite these differences, in practice the results are often very similar
* Logistic regression can also t quadratic boundaries like QDA, by explicitly including quadratic terms in the model

©copyright CIEL 2015 **43**



**Python Programming**

* Multiple linear regression
* Other considerations in regression model

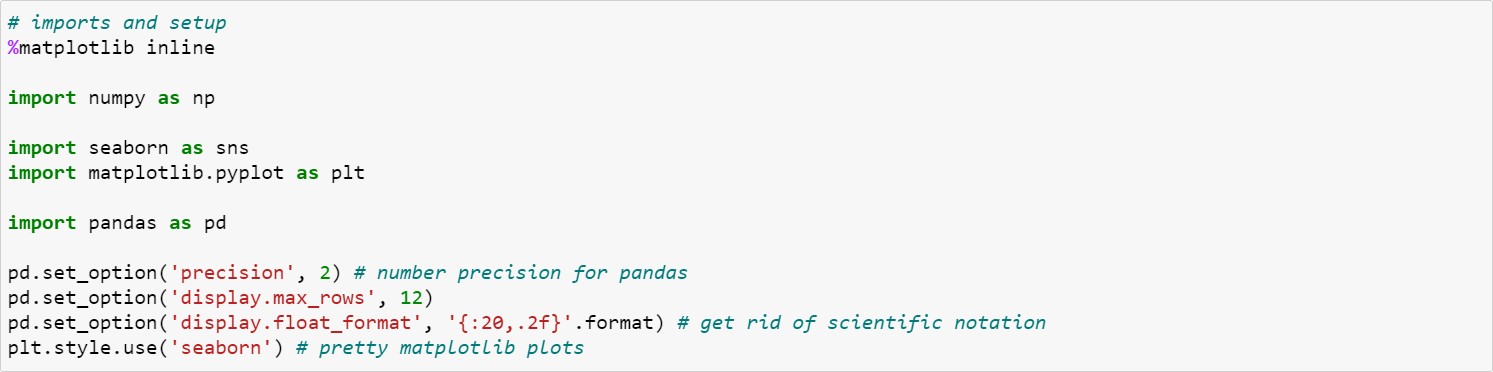
## Python lab

* Summary & Next class

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

# 4.6.1 The Stock Market Data

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

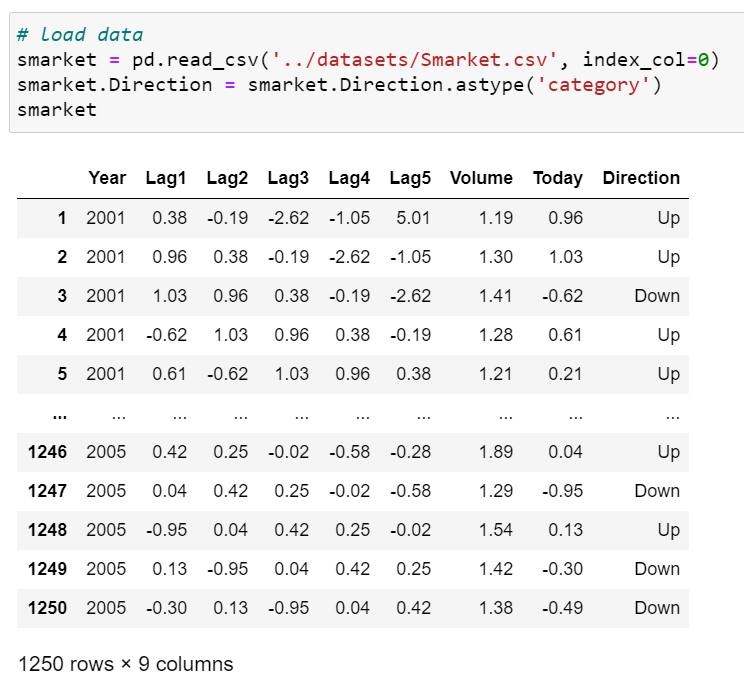


©copyright CIEL 2015 **45**

# 4.6.1 The Stock Market Data

* + Load data: Percentage returns for S&P 500 stock index over 1,250 days: 2001~2005

Percentage returns for each of five previous trading days



Number of shares traded on previous day, in billion

Percentage return on date in question

Up or Down on this date



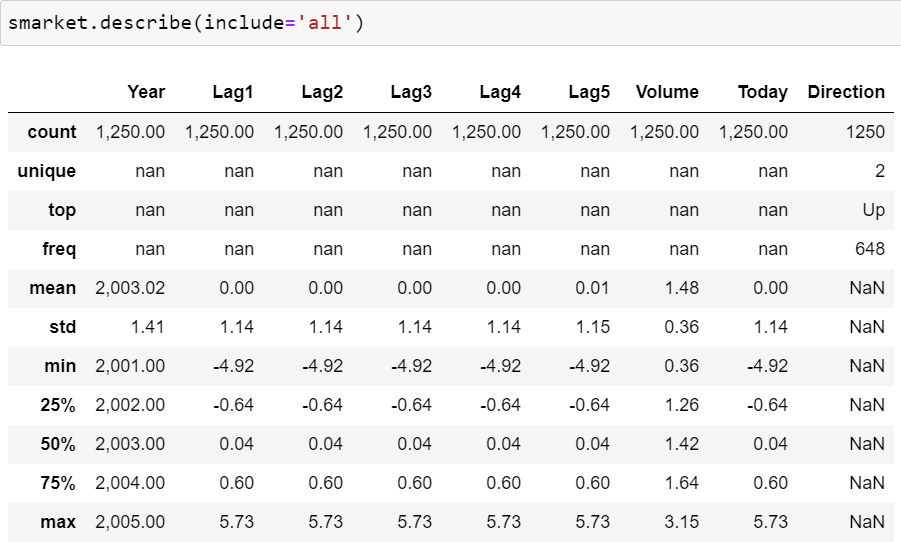
Converting an array of Python string object to categorical



©copyright CIEL 2015 **46**

# 4.6.1 The Stock Market Data

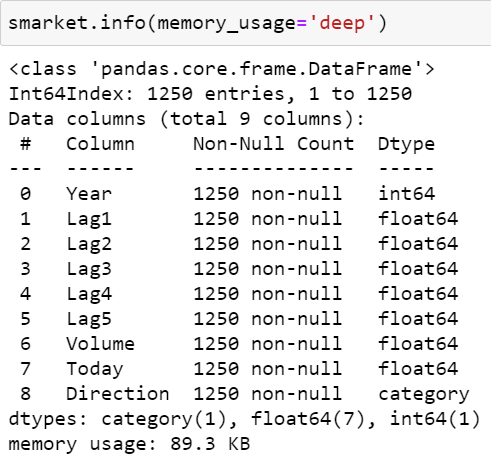
* + Describe: Producing multiple summary statistics in one shot
    - Include=‘all’ : including a union of attributes of each type.



©copyright CIEL 2015 **47**

# 4.6.1 The Stock Market Data

* + Printing information about a DataFrame
    - Memory\_usage=‘deep’ : how much memory is used in bytes by each column



©copyright CIEL 2015 **48**

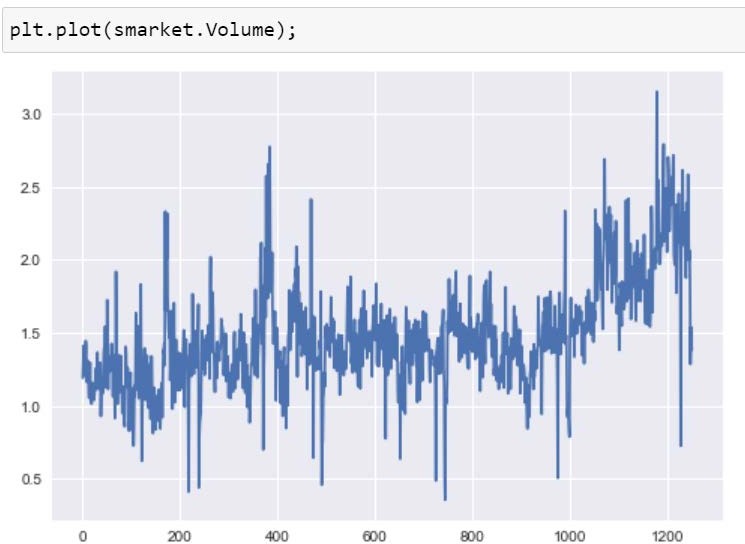
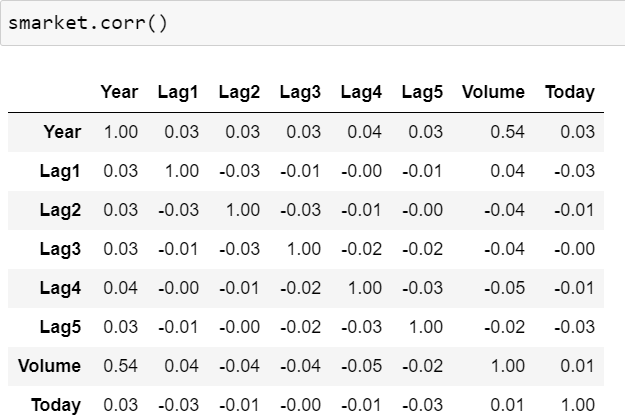
# 4.6.1 The Stock Market Data

* + Plot pairwise relationships in a dataset
    - sns.pairplot(smarket, hue=‘Direction’);
    - hue='Direction’ : Variable in data to map plot aspects to different colors

©copyright CIEL 2015 **49**

# 4.6.1 The Stock Market Data

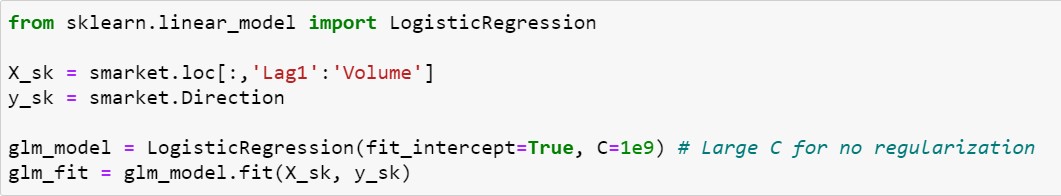
* + Correlation matrix  Plot



©copyright CIEL 2015 **50**

# 4.6.2 Logistic Regression

* + Using scikit-learn: LogisticRegression



Selecting a subset of the rows and columns

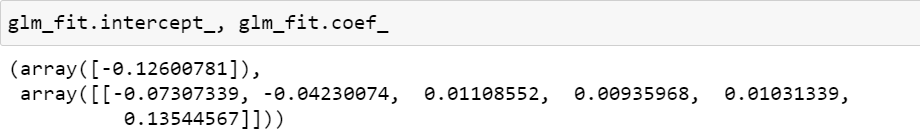
Inverse of regularization strength

Name: generalized linear model (GLM)

Fit the model according to the given training data.

Specifies if a constant (a.k.a. bias or intercept) should be added to the decision function.





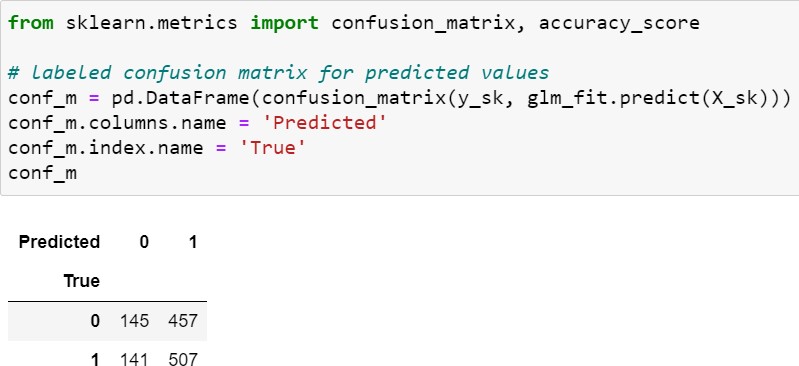
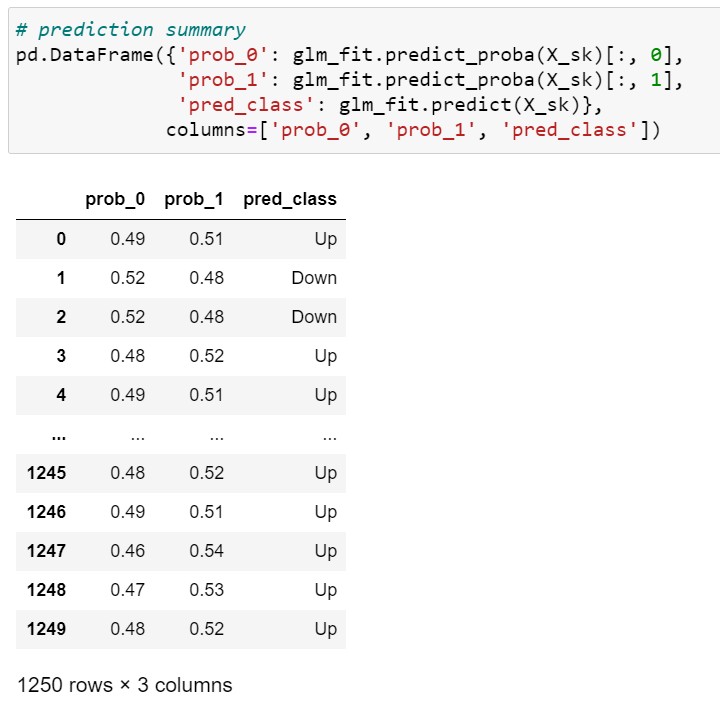
**51**

©copyright CIEL 2015

# 4.6.2 Logistic Regression

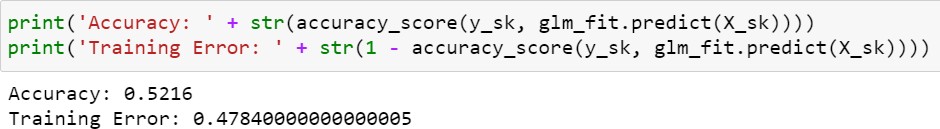
* + Predict probability that the market will go up  Confusion matrix

Probability estimates



Compute confusion matrix to evaluate the accuracy of a classification.

Predict class labels for samples in X\_sk

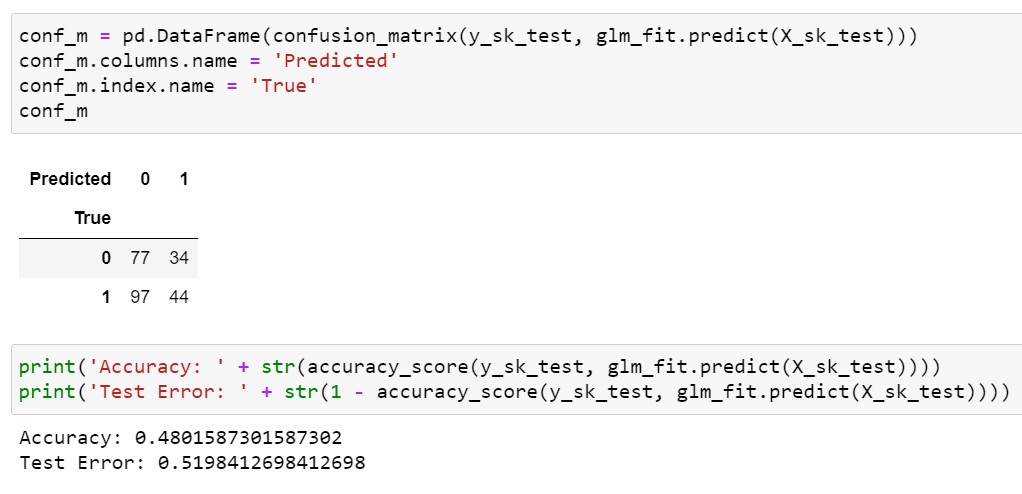
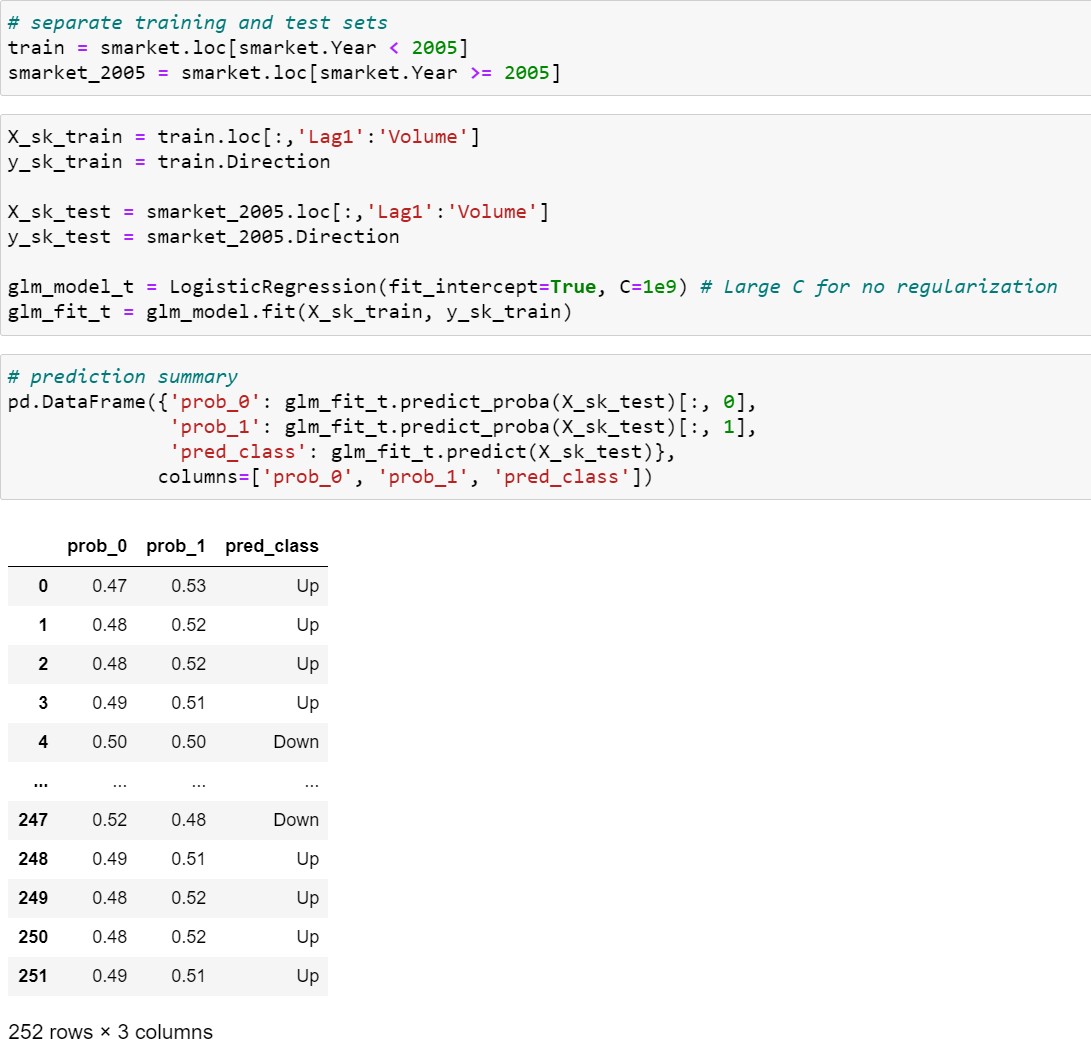


Proportion of class labels predicted correctly

©copyright CIEL 2015 **52**

# 4.6.2 Logistic Regression

* + Trained & Tested



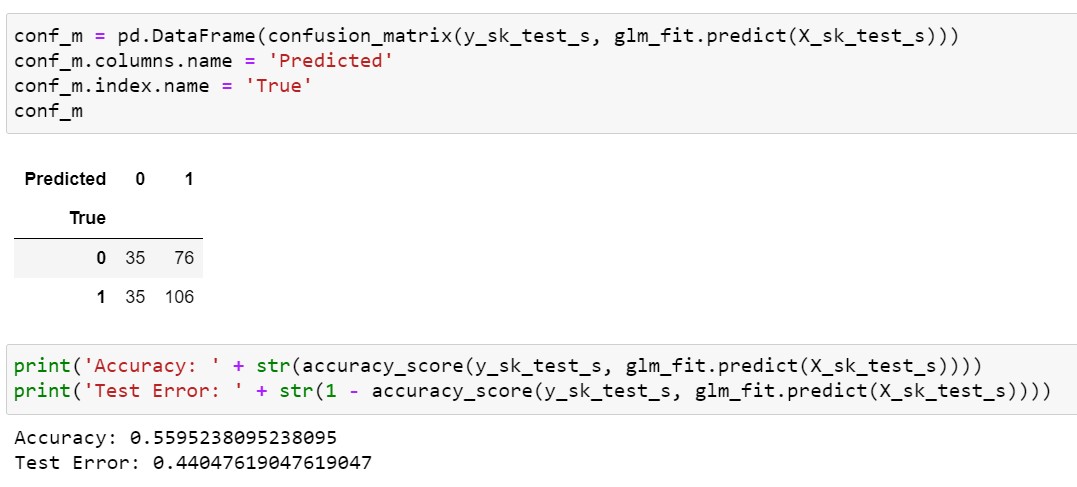
Test data set

©copyright CIEL 2015 **53**

# 4.6.2 Logistic Regression

* + Logistic regression using just Lag1 and Lag2

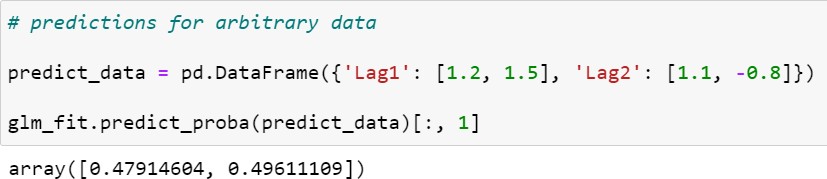
Just Direction ~ Lag1+Lag2



©copyright CIEL 2015 **54**

# 4.6.2 Logistic Regression

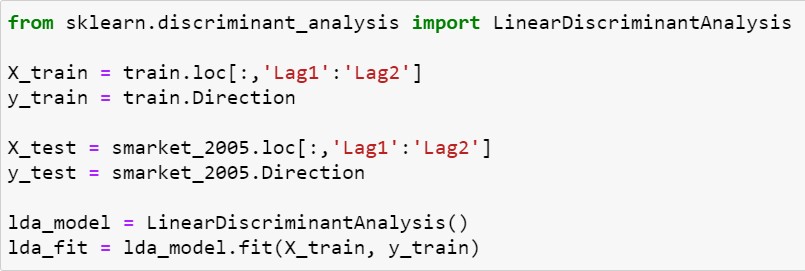
* + Predictions for arbitrary data



©copyright CIEL 2015 **55**

# 4.6.3 Linear Discriminant Analysis

* + Using scikit-learn: LinearDiscriminantAnalysis



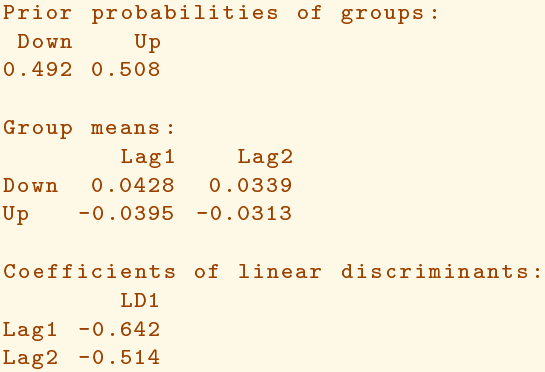
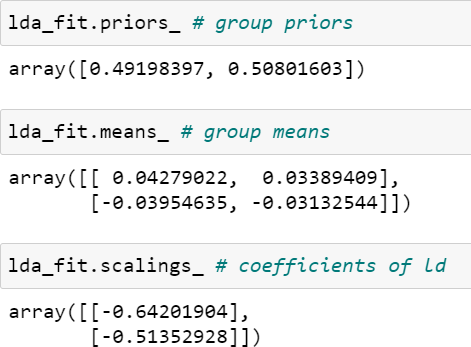
Just two predictors: Direction ~ Lag1+Lag2

LDA

©copyright CIEL 2015 **56**

# 4.6.3 Linear Discriminant Analysis

* + LDA training

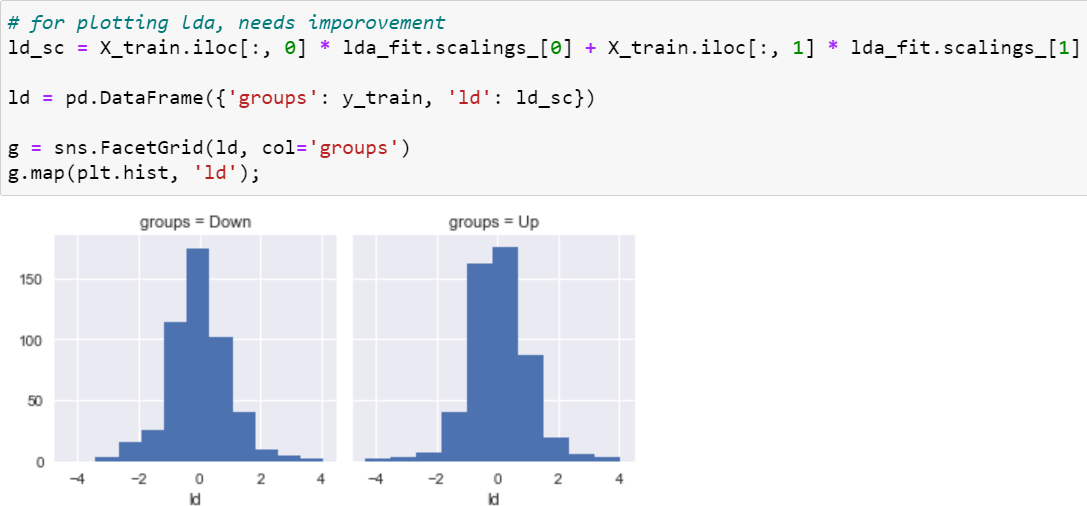


©copyright CIEL 2015 **57**

# 4.6.3 Linear Discriminant Analysis

* + Plotting LDA

Multi-plot grid for plotting conditional relationships

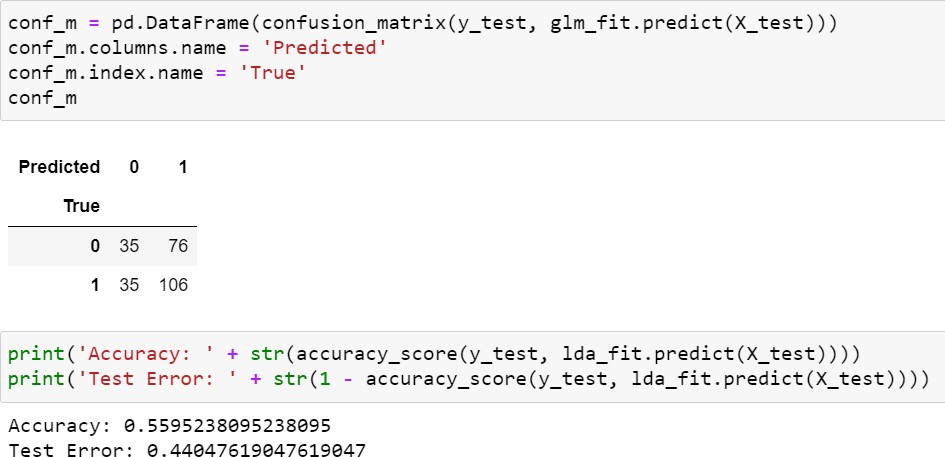
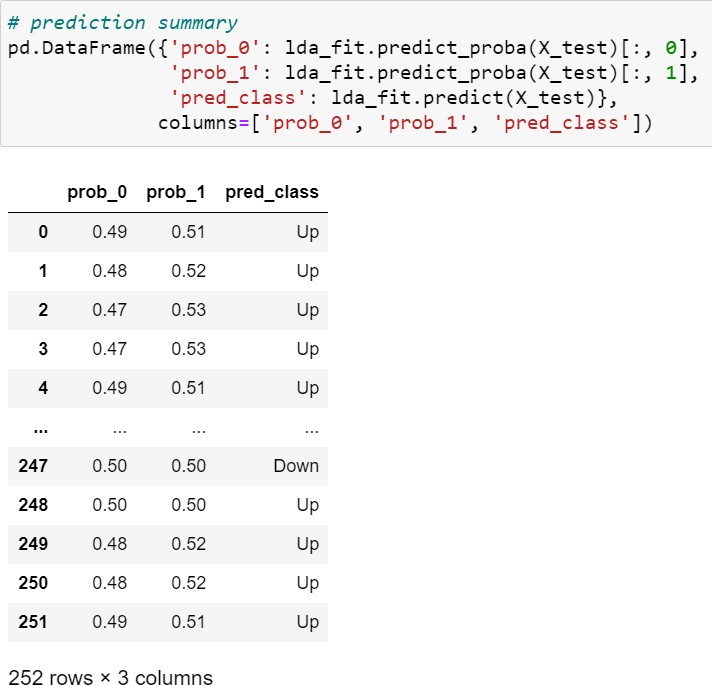


Apply a plotting function to each facet’s subset of the data

©copyright CIEL 2015 **58**

# 4.6.3 Linear Discriminant Analysis

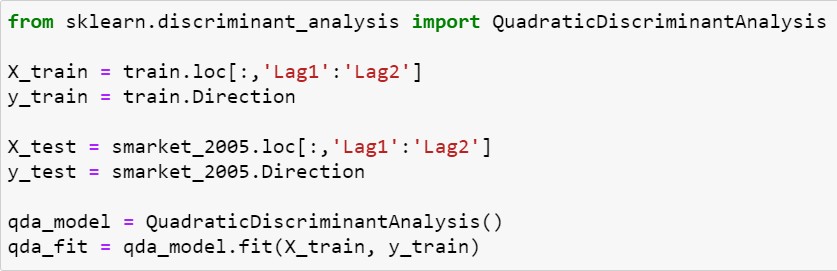
* + LDA training results



©copyright CIEL 2015 **59**

# 4.6.4 Quadratic Discriminant Analysis

* + Using scikit-learn: LinearDiscriminantAnalysis



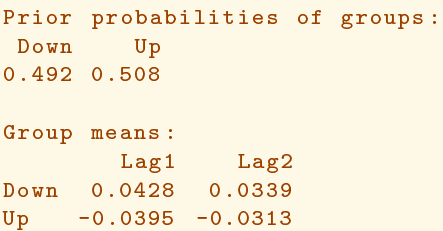
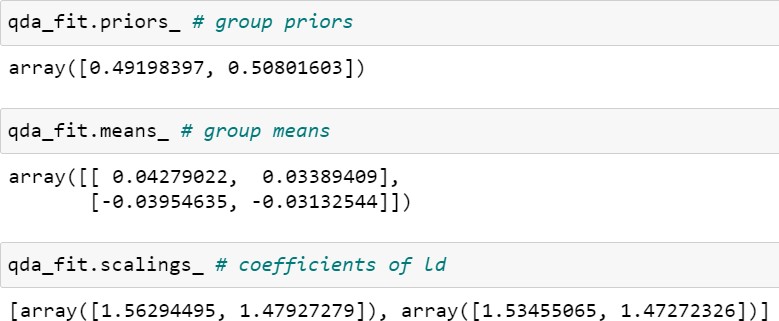
Just two predictors: Direction ~ Lag1+Lag2

QDA

©copyright CIEL 2015 **60**

# 4.6.4 Quadratic Discriminant Analysis

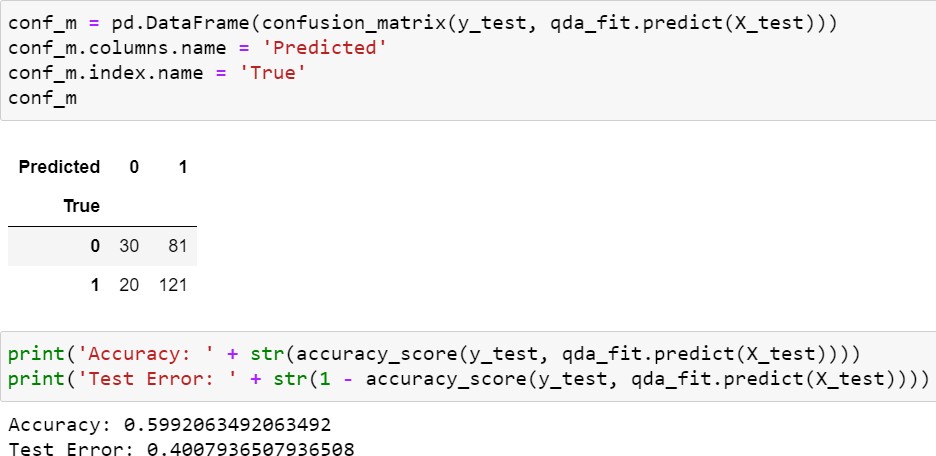
* + QDA training



©copyright CIEL 2015 **61**

# 4.6.4 Quadratic Discriminant Analysis

* + QDA training results



©copyright CIEL 2015 **62**

# 4.6.5 K-Nearest Neighbors

* + Using scikit-learn: KNeighborsClassifier



Just two predictors: Direction ~ Lag1+Lag2

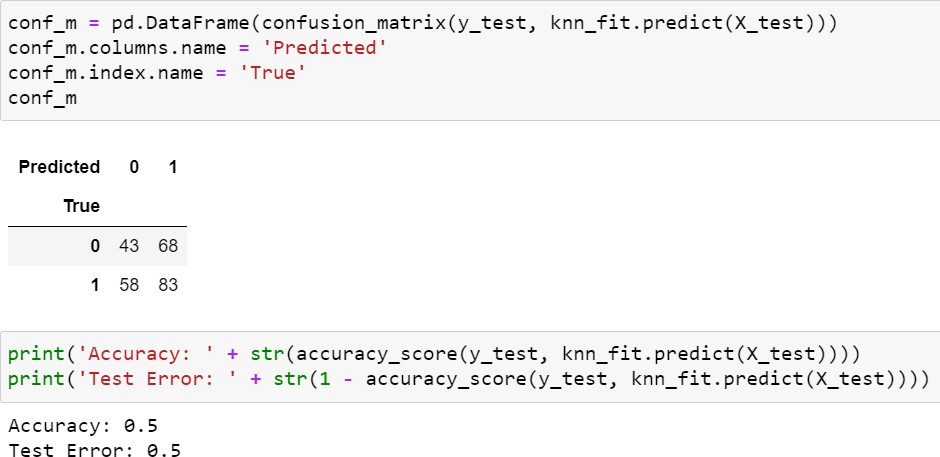
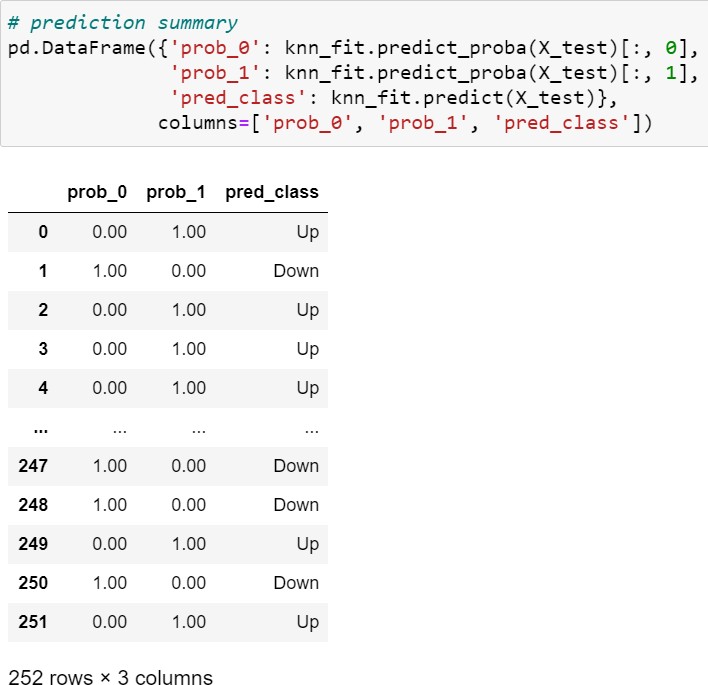
KNN

k=1

©copyright CIEL 2015 **63**

# 4.6.5 K-Nearest Neighbors

* + KNN training results

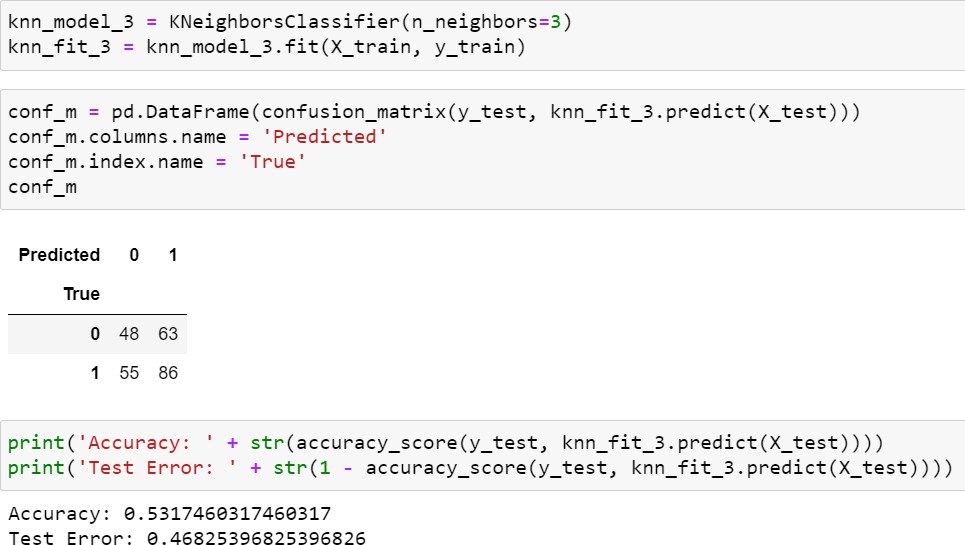


©copyright CIEL 2015 **64**

# 4.6.5 K-Nearest Neighbors

* + KNN: k=3

Improved slightly compared with k=1 Increasing K no further improvements



©copyright CIEL 2015 **65**



**Summary & Next Class**

* + - Multiple linear regression
    - Other considerations in regression model
    - Python lab

## Summary & Next class

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

# Summary

* + Classification
* Logistic regression is very popular for classification, especially when 𝐾𝐾 = 2
* LDA is useful when 𝑛𝑛 is small, or the classes are well separated, and Gaussian assumptions are reasonable. Also when 𝐾𝐾 > 2
* Naive Bayes is useful when 𝑝𝑝 is very large
* See Section 4.5 for some comparisons of logistic regression, LDA and KNN
  + Python lab
* scikit-learn: LogisticRegression
* scikit-learn: LinearDiscriminantAnalysis
* scikit-learn: QuadraticDiscriminantAnalysis
* scikit-learn: KNeighborsClassifier

©copyright CIEL 2015 **67**

# Assignments

* + eClass > Assignments
* Upload 2 or 3 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 **68**

# 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 **69**