# IST 5535: Machine Learning Algorithms and Applications

Langtao Chen, Spring 2021

## 4. Classification

# Reading

- Book Chapter 4

# Learning Objectives

- Understand logistic regression, linear discriminant analysis, and quadratic discriminant analysis.
- Understand performance measures including sensitivity, specificity, false positive rate, false negative rate, and AUC.
- Understand the impact of prediction threshold on performance measures.
- Be able to compare logistic regression, linear discriminant analysis, quadratic discriminant analysis, and KNN.
- Be able to use R to conduct logistic regression, linear discriminant analysis, and quadratic discriminant analysis.

# Agenda

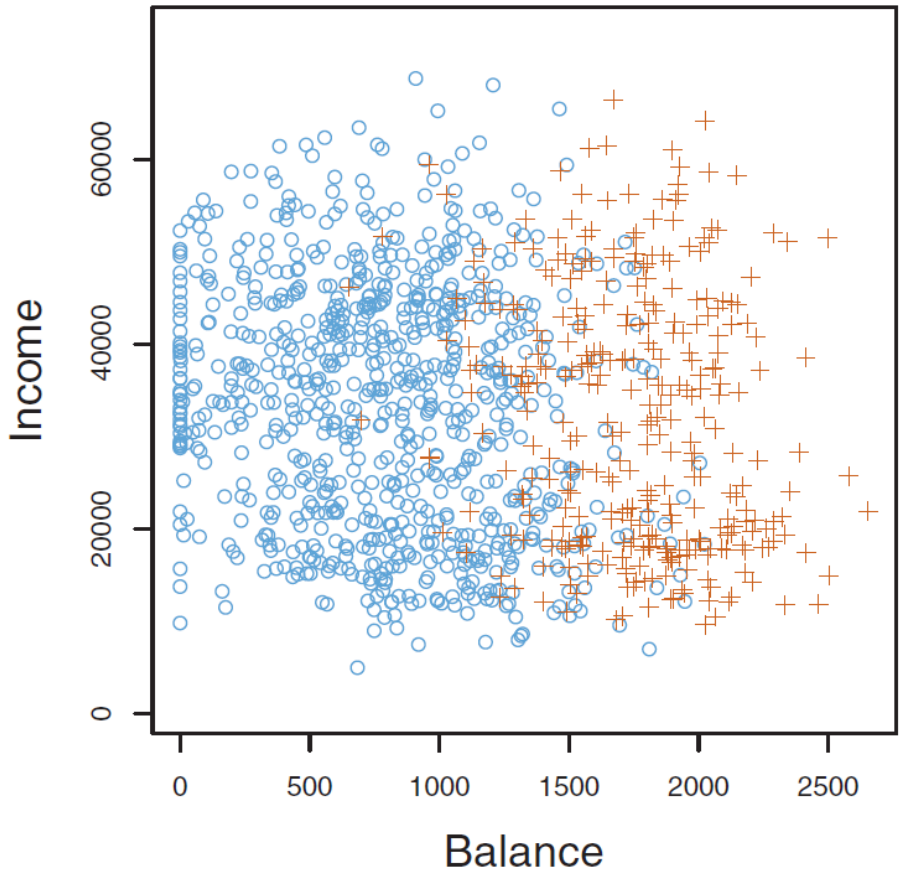
- Logistic Regression
- Linear Discriminant Analysis (LDA)
- More Performance Measures
- Quadratic Discriminant Analysis (QDA)
- A Comparison of Classification Methods

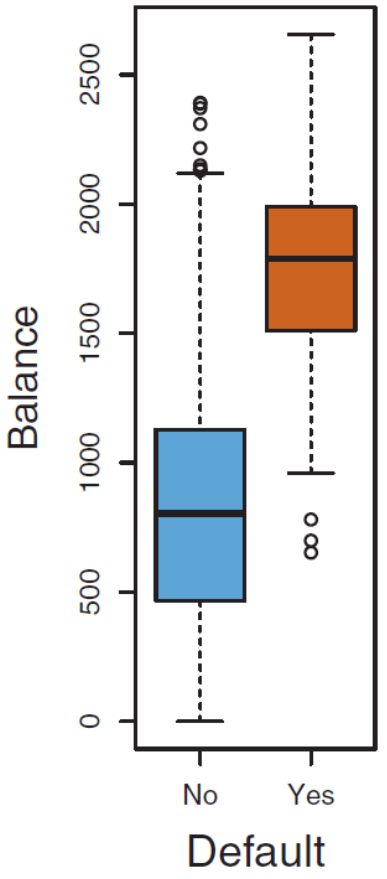
# Classification

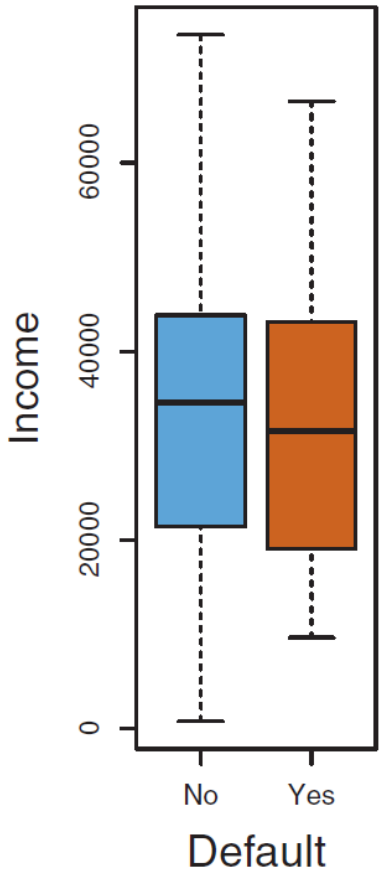
- In many cases, the response variable is qualitative or categorical:
  - Will the account holder pay off or default on the loan? $\text{default} \in \{\text{yes}, \text{no}\}$
  - Is the email spam or ham? $\text{email} \in \{\text{spam}, \text{ham}\}$
  - Is this bank transaction true or fraudulent? $\text{transaction} \in \{\text{true}, \text{fraudulent}\}$
  - ……
- Usually, we are interested in estimating the probabilities of Y belonging to each class.
- In this section, we'll discuss three classifiers:
  - Logistic regression
  - Linear discriminant analysis
  - Quadratic discriminant analysis

# Default Credit Card Payment

- Predict whether a customer will default on his or her credit card payment, on the basis of annual income and monthly credit card balance.



# Can We Use Linear Regression?

- For a qualitative response variable with more than 2 levels, there is no way to code this qualitative variable as a continuous variable.

- We may consider coding the response as follows:

$$y = \begin{cases} 1 & \text{if stroke;} \\ 2 & \text{if drug overdose;} \\ 3 & \text{if epileptic seizure.} \end{cases}$$

Ordinal

Nominal

# Can We Use Linear Regression?

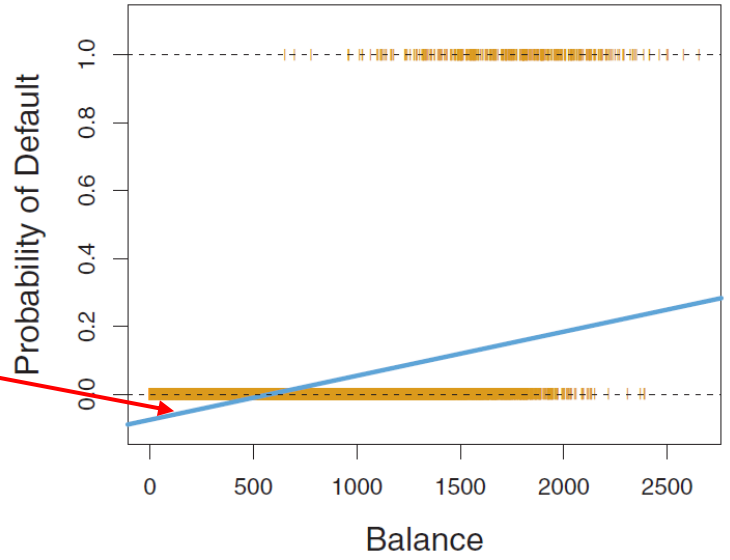
- For a qualitative response variable with 2 levels, linear regression could be used:

$$y = \begin{cases} 1, \text{if Yes;} \\ 0, \text{if No.} \end{cases}$$

=> Linear Probability Model

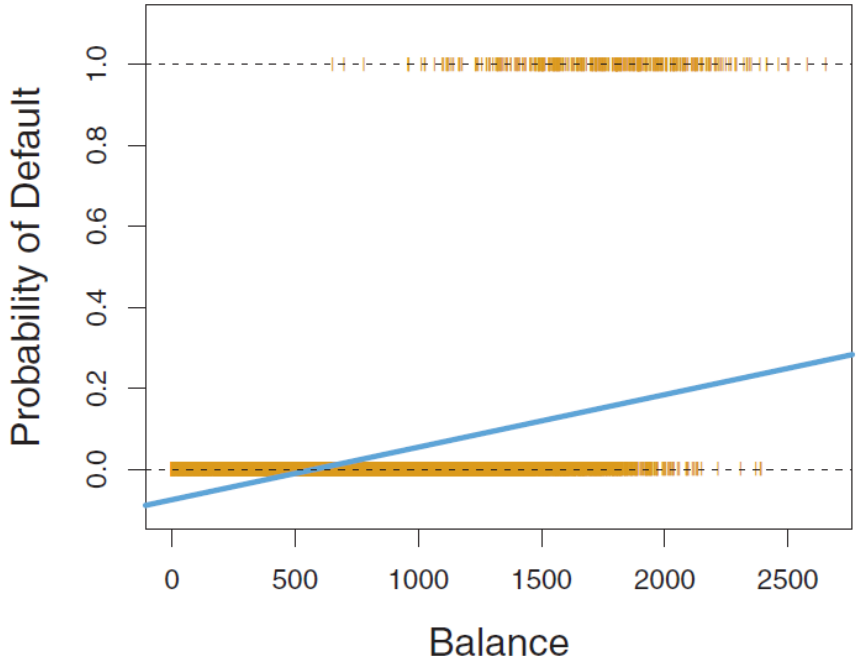
- We simply perform a linear regression of y on X and classify as Yes if $\hat{y} > 0.5$
- However, the predicted values can be outside the [0, 1] interval, making them hard to interpret.

When balance < 500, Pr(Default) is negative.



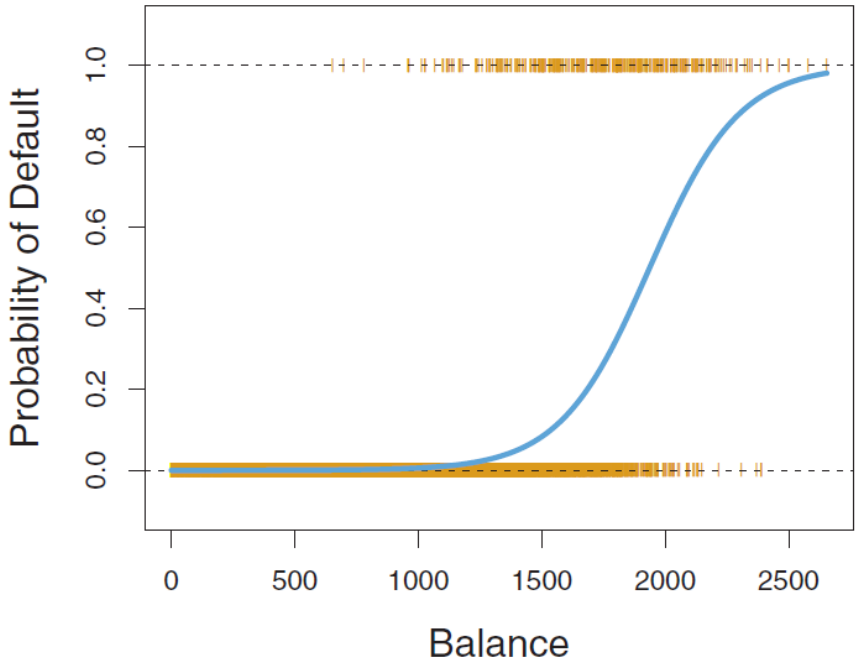
# Can We Use Linear Regression?

- In such case of binary response, logistic regression is preferred than linear regression




Linear Regression

$\text{Default} = \beta_0 + \beta_0 * \text{Balance}$




Logistic Regression

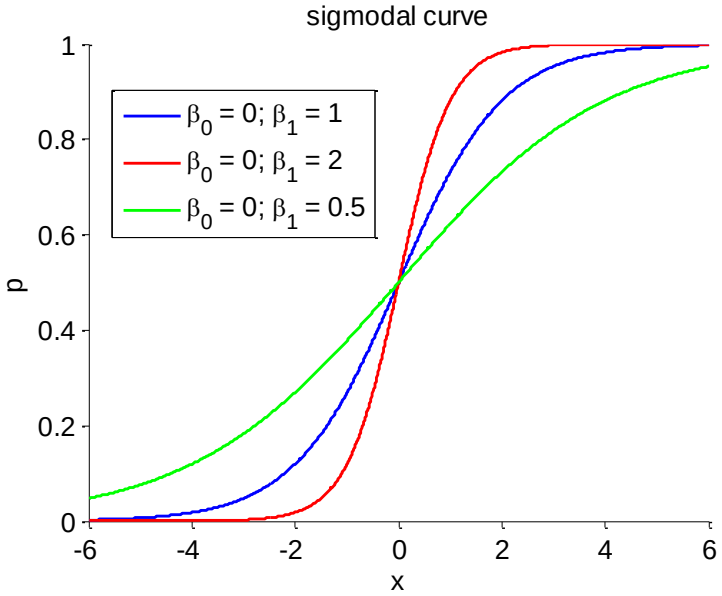
$Logit\ (\text{Default} = \text{Yes}) = \beta_0 + \beta_0 * \text{Balance}$

# Fitting a Probability

- Logistic regression model maps a linear combination $\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots\ldots + \beta_n x_n$ from $(-\infty, +\infty)$ to [0,1] by using a probability function

$$p(y|X) = \frac{\exp(X\beta)}{1+\exp(X\beta)} = \frac{exp(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots\ldots + \beta_n x_n)}{1+exp(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots\ldots + \beta_n x_n)}$$

- We can fit the distribution of y with a Logistic Curve




- The intercept basically just 'scale' the input variable
- Large regression coefficient => risk factor strongly influences the probability

# Transform Logistic to Linear Model

$$P(y|X) = \frac{e^{(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots\ldots + \beta_n x_n)}}{1 + e^{(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots\ldots + \beta_n x_n)}}$$

- Step 1: Specify a probability as odds
  - $odds = \frac{P(y|X)}{1 - P(y|X)} = e^{(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots\ldots + \beta_n x_n)}$
- Step 2: Calculate the logit function
  - $\text{Logit} = \ln(odds) = \ln\left(\frac{P(y|X)}{1 - P(y|X)}\right)$

$$= \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots\ldots + \beta_n x_n$$

# Implement Logistic Regression in R

- Use glm() function to fit generalized linear models;
- Need to specify the family as the binomial distribution, or else it would be a linear regression model;
- Then call summary() function to report the logistic regression results;
- An alternative approach is to use the stargazer package to report the result.

```
model <- glm(default ~ balance + income + student,
             family=binomial(link='logit'),data = Default)
summary(model)
```

# Interpreting Logistic Regression Result

```
Logistic Regression
=========================================
                      Dependent variable:
                     ---------------------
                           default
-----------------------------------------
balance                     0.0057***
                            (0.0002)

income                      0.000003
                            (0.00001)

studentYes                  -0.6468**
                            (0.2363)

Constant                    -10.8690***
                            (0.4923)

-----------------------------------------
Observations                  10,000
Log Likelihood               -785.7724
Akaike Inf. Crit.          1,579.5450
=========================================
Note:    *p<0.05; **p<0.01; ***p<0.001
```

- Balance has a positive and significant effect on default (p-value < 0.001). A unit increase in balance increases the log odds by 0.0057 after controlling for other factors.
- Income does not have a statistically significant on default.
- Being a student has a negative and significant effect on default (p-value < 0.01), keeping all other factors constant. Being student reduces the log odds by 0.6468 after controlling for other factors.

# Confounding

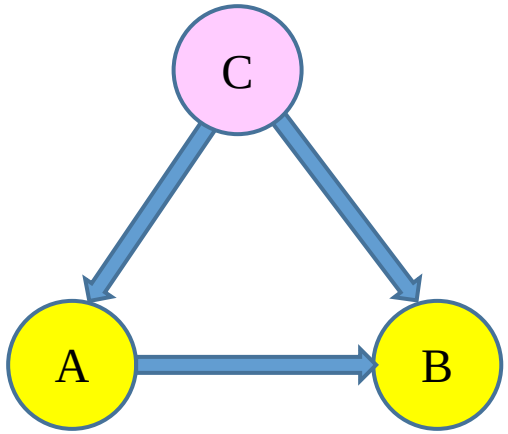
- Confounding due to high correlation between student and balance. Balance is a confounder or confounding variable in this case.

```
Logistic Regression
================================================
                        Dependent variable:
                    ----------------------------
                              default
                          (1)             (2)
------------------------------------------------
balance                                0.0057***
                                        (0.0002)

income                                  0.000003
                                       (0.00001)

studentYes                0.4049***     -0.6468**
                          (0.1150)       (0.2363)

Constant                 -3.5041***    -10.8690***
                          (0.0707)       (0.4923)

------------------------------------------------
Observations               10,000         10,000
Log Likelihood           -1,454.3410     -785.7724
Akaike Inf. Crit.        2,912.6830     1,579.5450
================================================
Note:                *p<0.05; **p<0.01; ***p<0.001
```

- If only student is included, the effect is positive and significant;
- If all predictors are included, the effect of student is negative and significant.

# Confounding

- As ice cream sales increase, the rate of drowning deaths increases sharply. Therefore, ice cream consumption causes drowning.




Third Factor C Causes both A and B

C is called a confounder or confounding variable

# Confounding

- Confounders can distort the relationship between the predictor and the response.
- Dealing with confounding:
  - Experimental design: random assignment, within subject design
  - Observational design: statistical control

# Agenda

- Logistic Regression
- Linear Discriminant Analysis (LDA)
- More Performance Measures
- Quadratic Discriminant Analysis (QDA)
- A Comparison of Classification Methods

# When we have more than 2 response classes

- The regular logistic regression model can only deal with a binary response;
- Logistic regression can be extended to handle response variables with more than 2 classes;
- In practice, we often use linear discriminant analysis (LDA) for multi-class classification.

# Why Not Logistic Regression?

- When the classes are well-separated, the parameter estimates for the logistic regression model are surprisingly unstable. LDA does not suffer from this problem.

- LDA is more stable than logistic regression when:
  - $n$ is small, and
  - the distribution of the predictors $X$ is approximately normal in each of the classes.

- LDA is popular when we have more than two response classes.

# Using Bayes Theorem for Classification

- The famous Bayes theorem:

$$Pr(Y = k|X = x) = \frac{\Pr(X = x|Y = k) \cdot \Pr(Y = k)}{\Pr(X = x)}$$

- Re-write it as:

$$Pr(Y = k|X = x) = \frac{\pi_k f_k(x)}{\sum_{l=1}^{K} \pi_l f_l(x)}$$

  - $f_k(x) = \Pr(X = x|Y = k)$: *density* for X in class k
  - $\pi_k = \Pr(Y = k)$: marginal or *prior* probability for class k

# Bayes Classifier

- Bayes Classifier is the gold standard, but unattainable since the density is unknown.

$$Pr(Y = k|X = x) = \frac{\pi_k f_k(x)}{\sum_{l=1}^{K} \pi_l f_l(x)}$$

- Recap: Does a Bayes classifier lead to perfect prediction (zero error rate)?

- However, if we can find a way to estimate the density, then we can develop a classifier that approximates the Bayes classifier.

# Linear Discriminant Analysis when p =1

- Assume normal/Gaussian density

$$f_k(x) = \frac{1}{\sqrt{2\pi}\sigma_k} e^{-\frac{1}{2}\left(\frac{x-u_k}{\sigma_k}\right)^2}$$

- Further assume variances are the same, i.e., $\sigma_k = \sigma$
- Then, we get

$$p_k(x) = Pr(Y = k|X = x) = \frac{\pi_k \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\left(\frac{x-u_k}{\sigma}\right)^2}}{\sum_{l=1}^{K} \pi_l \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\left(\frac{x-u_l}{\sigma}\right)^2}}$$

# Discriminant Functions

- Classification is to find the class k when X=x for which $p_k(x) = Pr(Y = k|X = x)$ is the largest.

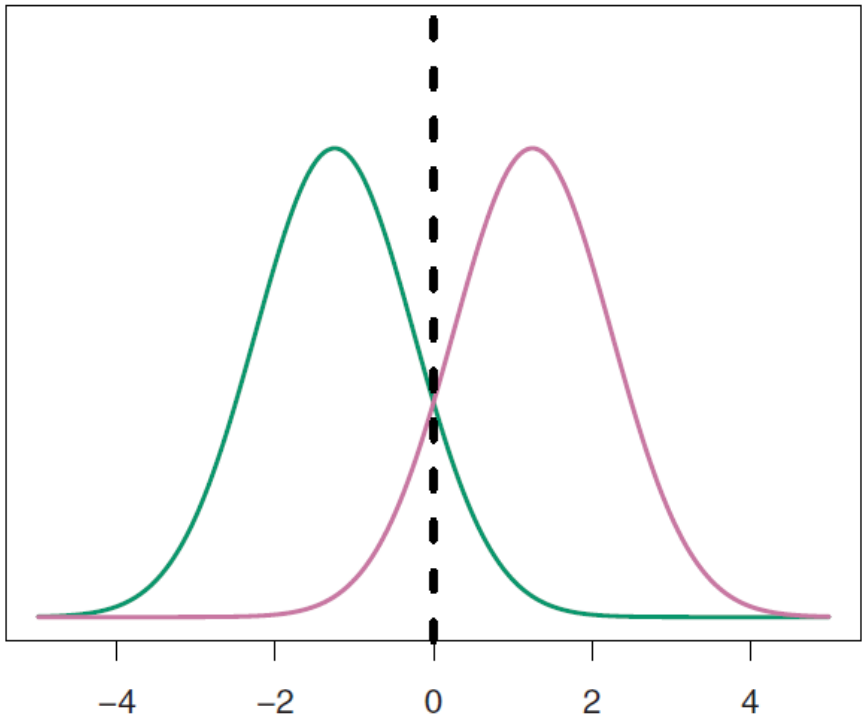
- After log-transform the previous formula and discard terms not depending on k, this is equivalent to assigning k to the class with the largest discriminant score:

$$\delta_k(x) = x \cdot \frac{u_k}{\sigma^2} - \frac{u_k^2}{2\sigma^2} + \log(\pi_k)$$

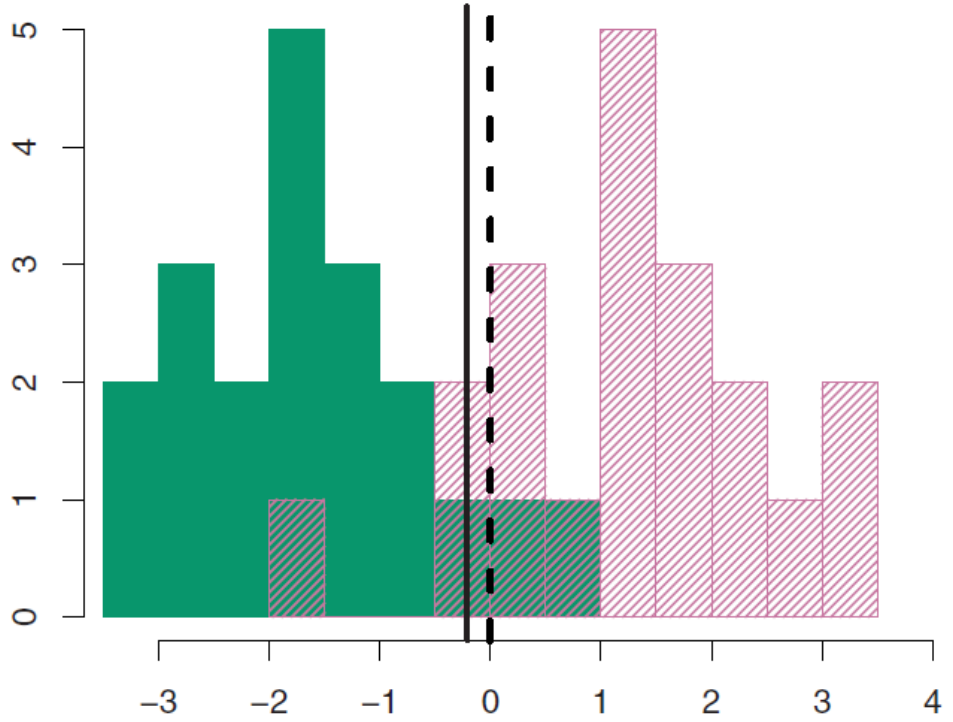
The word "linear" in LDA stems from the fact that the discriminant function is a linear function of x.

# LDA Example

- Two classes K=2, $u_1$=-1.25, $u_2$=1.25, $\pi_1=\pi_2$=0.5, $\sigma^2$=1




Dashed vertical line: Bayes decision boundary




Solid vertical line: LDA decision boundary estimated from training data

# Estimating the Parameters

- In practice, we don't know the parameters in normal distribution.
- LDA approximates the Bayes classifier by simply estimating the parameters and plugging them into the discriminant function.

$$\hat{\mu}_k = \frac{1}{n_k} \sum_{i:y_i=k} x_i$$

$$\hat{\sigma}^2 = \frac{1}{n-K} \sum_{k=1}^{K} \sum_{i:y_i=k} (x_i - \hat{\mu}_k)^2$$

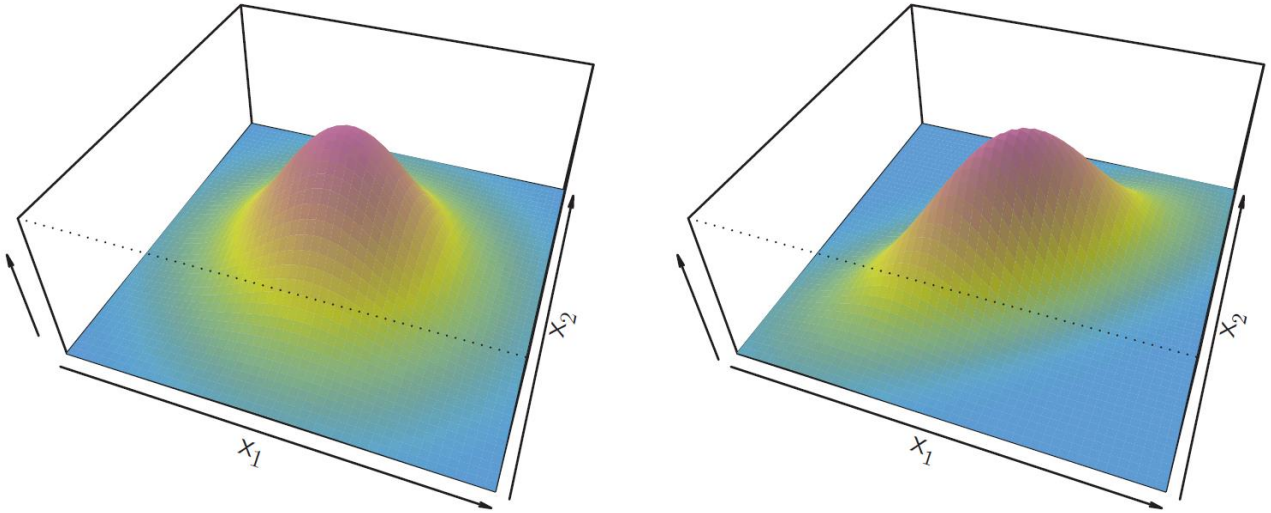
$$\hat{\pi}_k = n_k/n$$

$$\hat{\delta}_k(x) = x \cdot \frac{\hat{\mu}_k}{\hat{\sigma}^2} - \frac{\hat{\mu}_k^2}{2\hat{\sigma}^2} + \log(\hat{\pi}_k)$$

# Linear Discriminant Analysis when p >1

- When X contains multiple predictors, the similar approach is applied by using a multivariate density function.

Examples: Two multivariate Gaussian density functions




$X_1$ and $X_2$ are uncorrelated

Corr($X_1$, $X_2$) = 0.7

$$\delta_k(x) = x^T \mathbf{\Sigma}^{-1} \mu_k - \frac{1}{2} \mu_k^T \mathbf{\Sigma}^{-1} \mu_k + \log \pi_k$$

# A General Process of LDA

- Extract discriminant functions
  - Number of LD = min(number of predictors, number of classes -1)
  - $LD_m = \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p$
- Use linear discriminants to classify response

# Summary of LDA Procedure

- Assume that observations from each class are drawn from a Gaussian distribution;
- Estimate parameters (means, variance/covariance) in the Gaussian distribution density from the data;
- Plug parameter estimates into Bayes theorem to calculate $p_k(x)$;
- Assign the class that has the largest probability.

# LDA on the Default Data

- A confusion matrix comparing LDA predictions to true statuses

| | | True default status | | |
|---|---|---|---|---|
| | | No | Yes | Total |
| Predicted default status | No | 9,645 | 254 | 9,899 |
| | Yes | 22 | 79 | 101 |
| | Total | 9,667 | 333 | 10,000 |

- The overall prediction accuracy seems good

$$Accuracy = \frac{9{,}645 + 79}{10{,}000} = 97.24\%$$

- A null classifier (always classifying the response as the majority class) yields to 96.7% accuracy!

- However, if our purpose is trying to identify high-risk customer, this model performs not that well:
  - This model only detect 79 out of 333 true default customers;
  - 254 customers who default are incorrectly predicted by the model as no default customers.

# Agenda

- Logistic Regression
- Linear Discriminant Analysis (LDA)
- More Performance Measures
- Quadratic Discriminant Analysis (QDA)
- A Comparison of Classification Methods

# Sensitivity and Specificity

- *Sensitivity* (*true positive rate*, *recall*, or *hit rate*): the percentage of true positive observations that are correctly identified.

$$Sensitivity = \frac{79}{333} = 23.7\%$$

- *Specificity* (or *true negative rate*): the percentage of true negative observations that are correctly identified.

$$Specificity = \frac{9645}{9667} = 99.8\%$$

| | | True default status | | |
|---|---|---|---|---|
| | | No | Yes | Total |
| Predicted default status | No | 9,645 | 254 | 9,899 |
| | Yes | 22 | 79 | 101 |
| | Total | 9,667 | 333 | 10,000 |

Why does LDA have such a low sensitivity?

# False Positive Rate and False Negative Rate

- False positive rate: The fraction of negative observations that are classified as positive.

$$False\ positive\ rate = \frac{22}{9667} = 0.2\%$$

- False negative rate: The fraction of positive observations that are classified as negative.

$$False\ negative\ rate = \frac{254}{333} = 76.3\%$$

| | | True default status | | |
|---|---|---|---|---|
| | | No | Yes | Total |
| Predicted default status | No | 9,645 | 254 | 9,899 |
| | Yes | 22 | 79 | 101 |
| | Total | 9,667 | 333 | 10,000 |

$False\ positive\ rate = 1 - specificity$

$False\ negative\ rate = 1 - sensitivity$

# The Problem of Imbalanced Dataset

- A dataset is unbalanced when it has uneven class distribution.
  - In a customer loan dataset, only 5 out of 100 customers have bad credit.
  - Suppose we need to train a classifier to classify whether a customer has good credit.
- The accuracy paradox
  - A "dumb" algorithm (null classifier) is to always predict the majority class for new data;
  - Such an algorithm does not learn the underlying patterns from the data, but it "really" performs very good: accuracy = 95%.

| Pred. Class \ True Class | Yes | No |
|---|---|---|
| Yes | 95 | 5 |
| No | 0 | 0 |

$$Accuracy = \frac{95+0}{95+5+0+0} = 95\% \qquad Sensitivity = \frac{95}{95+0} = 100\%$$

However, $Specificity = \frac{0}{0+5} = 0\%$

This algorithm fails to detect risky customers.

**The problem: machine learning algorithms tends to bias towards the majority class.**

# Deal with Imbalanced Dataset

- Plot the confusion matrix, understand problems in your analysis.
- Use the right performance metrics
  - Not rely on accuracy, choose other metrics such as balanced accuracy, recall, specificity, AUC, precision, f1, etc.

$$Balanced\ Accuracy = \frac{Sensitivity + Specificity}{2}$$

- Resample the training dataset
  - Over-sample the minority class
  - Under-sample the majority class
- Use different threshold for prediction
- Customize the cost function to assign larger penalty to the misclassified minority class

# Use 0.2 as Threshold for Prediction

- $Pr(default = yes | X = x) > threshold$

Threshold = 0.5

| | | True default status | | |
|---|---|---|---|---|
| | | No | Yes | Total |
| Predicted default status | No | 9,645 | 254 | 9,899 |
| | Yes | 22 | 79 | 104 |
| | Total | 9,667 | 333 | 10,000 |

$$False\ positive\ rate = \frac{22}{9667} = 0.2\%$$

$$False\ negative\ rate = \frac{254}{333} = 76.3\%$$

Threshold = 0.2

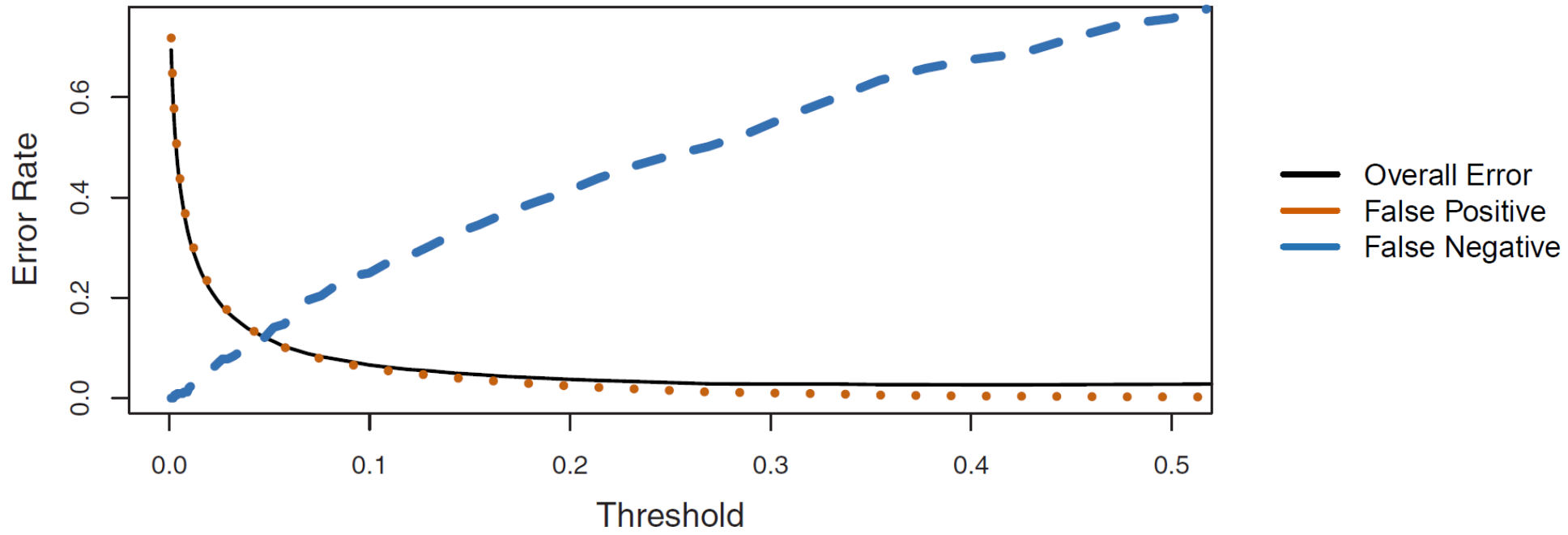
| | | True default status | | |
|---|---|---|---|---|
| | | No | Yes | Total |
| Predicted default status | No | 9,435 | 140 | 9,570 |
| | Yes | 232 | 193 | 430 |
| | Total | 9,667 | 333 | 10,000 |

$$False\ positive\ rate = \frac{232}{9667} = 2.4\%$$ ↗

$$False\ negative\ rate = \frac{140}{333} = 42.0\%$$ ↘

# Error Rates as a Function of Threshold

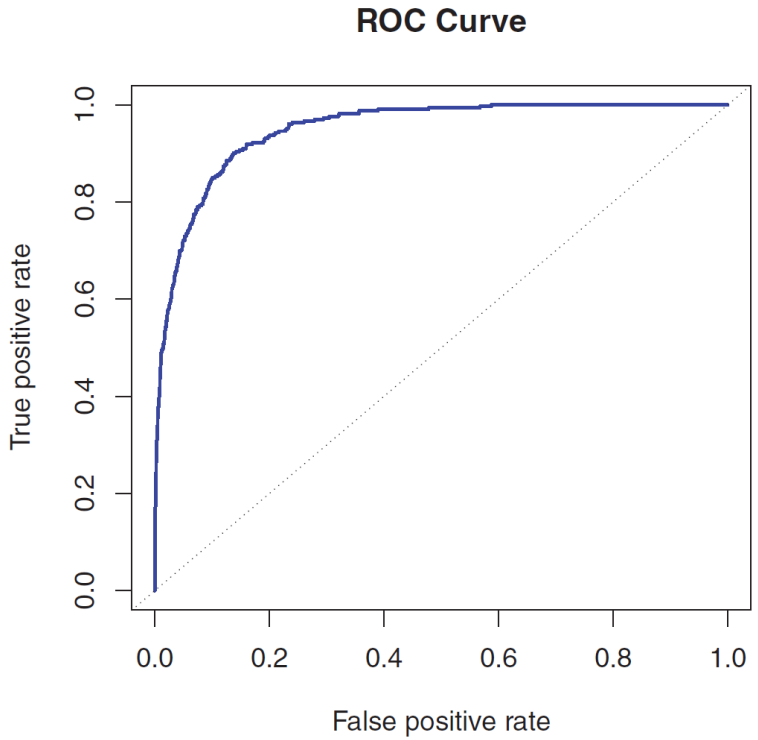
- We can change the two error rates using different thresholds in [0, 1].
- The best threshold should be based on domain knowledge.




In order to further reduce the false negative rate (or increase sensitivity), we may want to reduce the threshold to 0.1 or less.

# ROC (receiver operating characteristics) Curve

- ROC plot displays both false positive rate and true positive rate simultaneously with varying thresholds.
- We can use the *AUC* (area under the curve) to summarize the overall performance.
- Good classifier has large area under curve (AUC).



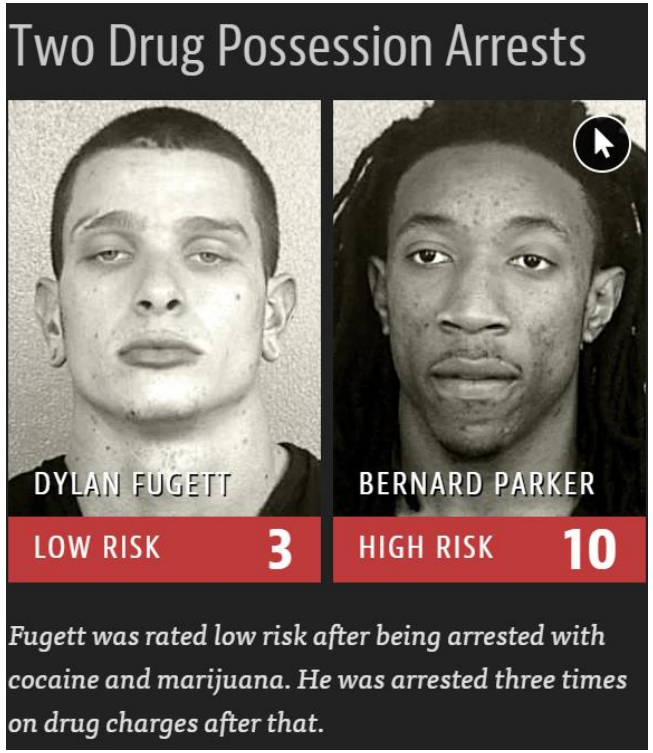

General guide

.90-1 = excellent (A)
.80-.90 = good (B)
.70-.80 = fair (C)
.60-.70 = poor (D)
.50-.60 = fail (F)

# More Fundamental Bias in Machine Learning

- Garbage in, garbage out
  - Biased data => biased machine learning models
- *Pro Publica* found machine learning algorithms falsely flagged black defendants as future criminals, wrongly labeling them at almost twice the rate of white defendants.

DYLAN FUGETT
Prior Offense:
1 attempted burglary
Subsequent Offenses:
3 drug possessions




BERNARD PARKER
Prior Offense:
1 resisting arrest without violence
Subsequent Offenses:
None



Source: https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing

# Further Reading

- Chouldechova, A., and Roth, A. 2020. "A Snapshot of the Frontiers of Fairness in Machine Learning," *Communications of the ACM* (63:5), pp. 82–89.

# Agenda

- Logistic Regression
- Linear Discriminant Analysis (LDA)
- More Performance Measures
- Quadratic Discriminant Analysis (QDA)
- A Comparison of Classification Methods

# Quadratic Discriminant Analysis (QDA)

- Issues of LDA:
  - LDA assumes the same variance/covariance for each class;
  - LDA may perform poorly due to this strong assumption.

- QDA assumes each class has its own covariance matrix. Then the discriminant function would be:

$$\begin{aligned}\delta_k(x) &= -\frac{1}{2}(x-\mu_k)^T\mathbf{\Sigma}_k^{-1}(x-\mu_k) - \frac{1}{2}\log|\mathbf{\Sigma}_k| + \log\pi_k \\ &= -\frac{1}{2}x^T\mathbf{\Sigma}_k^{-1}x + x^T\mathbf{\Sigma}_k^{-1}\mu_k - \frac{1}{2}\mu_k^T\mathbf{\Sigma}_k^{-1}\mu_k - \frac{1}{2}\log|\mathbf{\Sigma}_k| + \log\pi_k\end{aligned}$$

"Quadratic" in QDA: the discriminant function is a quadratic function of x.

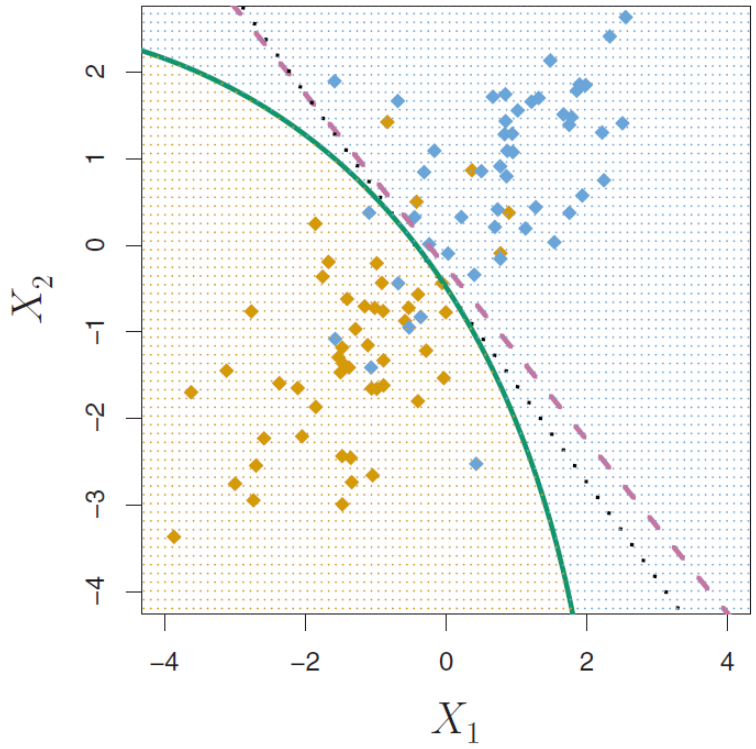
# Which One to Choose? LDA or QDA?

- The answer lies in the bias-variance trade-off
  - QDA allows a separate covariance matrix for each class, thus QDA is more flexible than LDA.
  - QDA may reduce the bias, but its variance might be higher.

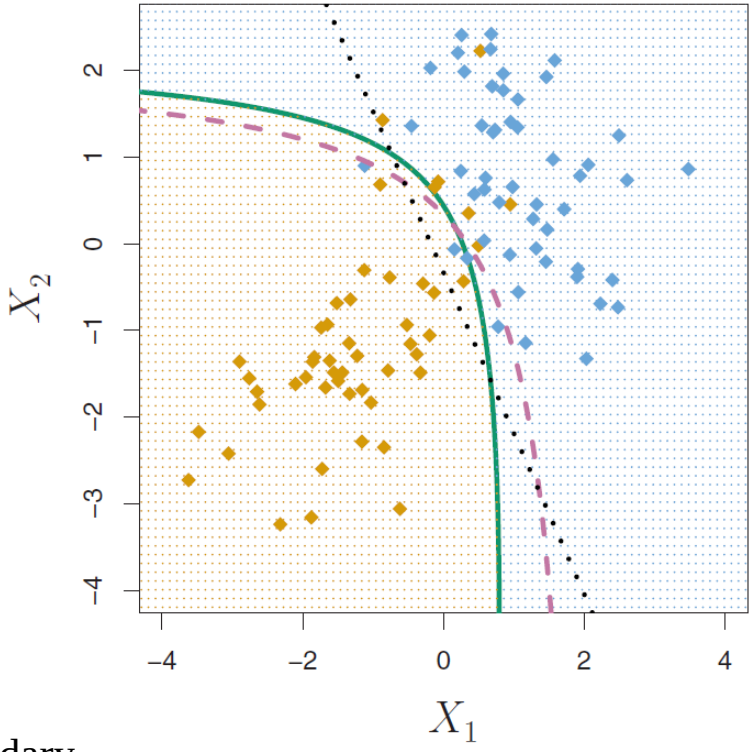
- In general,
  - LDA tends to be better if there are relatively few training observations and so reducing variance is crucial.
  - QDA is recommended if the training set is very large, so that the variance of the classifier is not a major concern, or if the assumption of a common covariance matrix for the $K$ classes is clearly untenable.

# LDA vs. QDA

Covariances of the X are equal across classes
Bayes decision boundary is linear
LDA is a better approximate




Covariances of the X are not equal across classes
Bayes decision boundary is quadratic
QDA is a better approximate




Black dotted: LDA boundary
Purple dashed: Bayes' boundary
Green solid: QDA boundary

# Agenda

- Logistic Regression
- Linear Discriminant Analysis (LDA)
- More Performance Measures
- Quadratic Discriminant Analysis (QDA)
- A Comparison of Classification Methods

# Logistic Regression vs. LDA

- For a two-class setting with p=1, the LDA discriminant function is

$$p_k(x) = Pr(Y = k|X = x) = \frac{\pi_k \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\left(\frac{x-u_k}{\sigma}\right)^2}}{\sum_{l=1}^{K} \pi_l \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\left(\frac{x-u_l}{\sigma}\right)^2}}$$

- Then, we can get:

$$\log\left(\frac{p_1(x)}{1-p_1(x)}\right) = \log\left(\frac{p_1(x)}{p_2(x)}\right) = c_0 + c_1 x$$

- Thus, LDA has the same form as logistic regression.
- The difference is how the parameters are estimated.
- In practice the results are often very similar.

# Logistic Regression vs. LDA

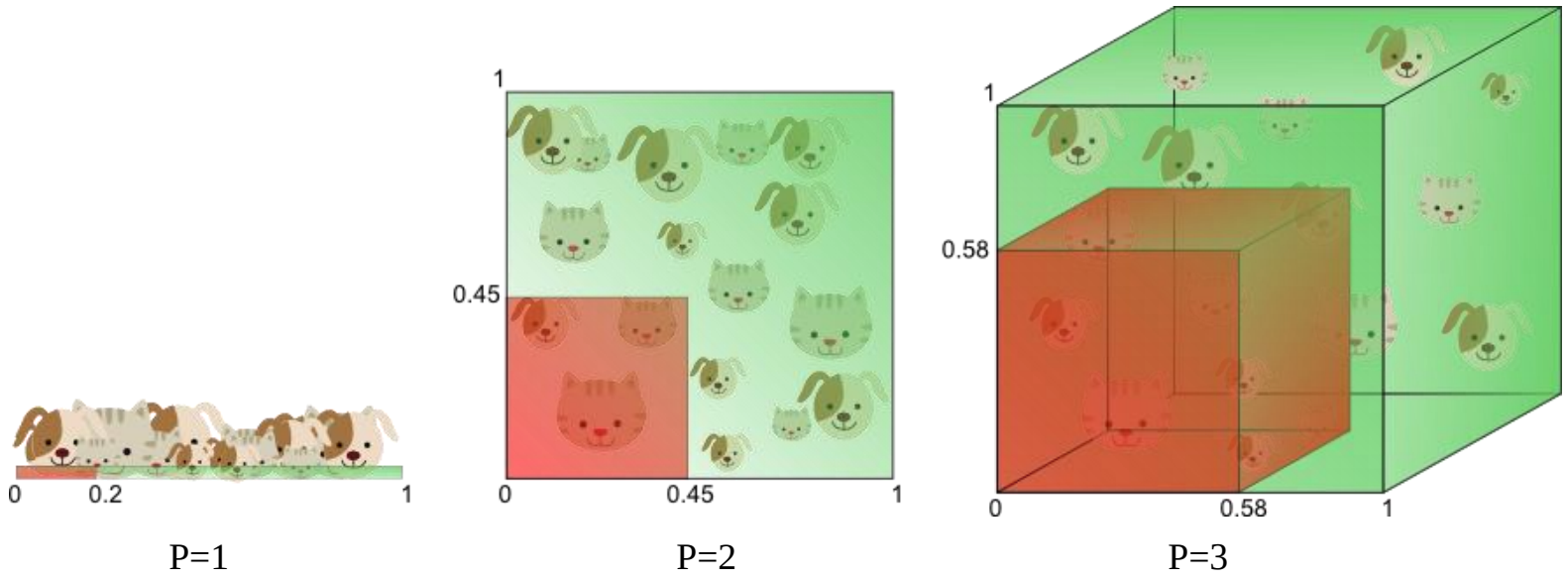
- LDA assumes that the observations are drawn from a Gaussian distribution with a common covariance matrix in each class, and so can provide some improvements over logistic regression when this assumption approximately holds.
- Conversely, logistic regression can outperform LDA if these Gaussian assumptions are not met.

# KNN vs. LDA and Logit

- KNN takes a completely different approach.
- KNN is a completely non-parametric approach: no assumptions are made about the shape of the decision boundary.
- Therefore, we can expect this approach to dominate LDA and logistic regression when the decision boundary is highly non-linear.
- However, KNN does not tell us which predictors are important; we don’t get a table of coefficients.

# KNN Suffers from the Curse of Dimensionality

- As variables are added, the data space becomes increasingly sparse.
- Prediction and classification models fail due to insufficient data for a useful model across so many variables.




P=1 P=2 P=3

The amount of training data needed to cover 20% of the feature range grows exponentially with the number of dimensions.



Image: http://www.visiondummy.com/2014/04/curse-dimensionality-affect-classification/

# QDA vs. KNN, LDA, and Logit

- QDA serves as a compromise between the non-parametric KNN method and the linear LDA and logistic regression approaches.
- Since QDA assumes a quadratic decision boundary, it can accurately model a wider range of problems than can the linear methods.
- No one single method dominates in all situations:
  - True decision boundary is linear: LDA and Logit perform well;
  - True decision boundary is moderately non-linear: QDA performs better;
  - True decision boundary is more complicated: non-parametric KNN is superior.

# Q & A