

Lecture 11

Classification

Readings: Zelterman, 2015, Chapter 10.1-10.4; Izenman, 2008 Chapter 8.1-8.4; ISLR, 2021 Chapter 9; Johnson & Wichern 2007, Chapter 11

DSA 8070 Multivariate Analysis

Whitney Huang
Clemson University

Agenda

Background

Binary Linear
Classification

Support Vector
Machines

1 Background

2 Binary Linear Classification

3 Support Vector Machines

- **Data:**

$$\{\mathbf{X}_i, Y_i\}_{i=1}^n,$$

where Y_i is the class information for the i_{th} observation
 $\Rightarrow Y$ is a qualitative variable

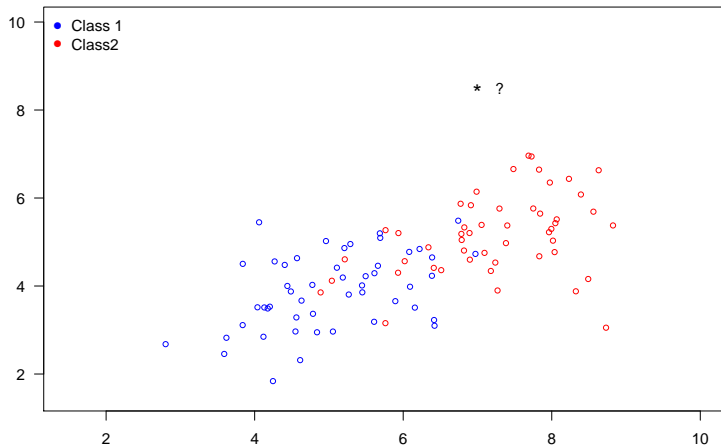
- **Classification** aims to classify a new observation (or several new observations) into one of those classes

Quantity of interest: $P(Y = k_{th} \text{ category} | \mathbf{X} = \mathbf{x})$

- In this lecture we will focus on **binary linear classification**

Toy Example

Wish to classify a new observation $x_i = (x_{1i}, x_{2i})$, denoted by $(*)$, into one of the two groups (**class 1** or **class 2**)



Background

Binary Linear
ClassificationSupport Vector
Machines

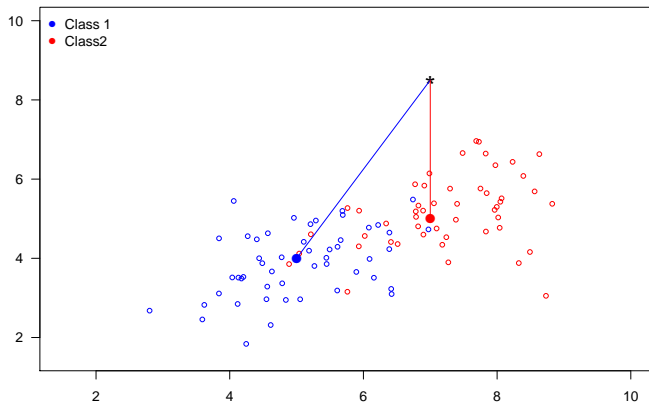
Toy Example Cont'd

We can compute the distances from this new observation $\mathbf{x} = (x_1, x_2)$ to the groups, for example,

$$d_1 = \sqrt{(x_1 - \mu_{11})^2 + (x_2 - \mu_{12})^2},$$

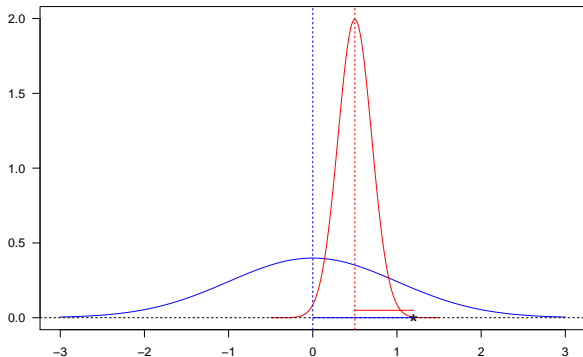
$$d_2 = \sqrt{(x_1 - \mu_{21})^2 + (x_2 - \mu_{22})^2}.$$

We can assign \mathbf{x} to the group with the smallest distance



Variance Corrected Distance

In this one-dimensional example, $d_1 = |x - \mu_1| > |x - \mu_2|$. Does that mean x is “closer” to group 2 (red) than group 1 (blue)?



We should take the “spread” of each group into account.

$$\tilde{d}_1 = |x - \mu_1|/\sigma_1 < \tilde{d}_2 = |x - \mu_2|/\sigma_2$$

General Covariance Adjusted Distance: Mahalanobis Distance

Classification



Background

Binary Linear
Classification

Support Vector
Machines

The [Mahalanobis distance \[Mahalanobis, 1936\]](#) is a measure of the distance between a point x and a multivariate distribution of X :

$$D_M(x) = \sqrt{(x - \mu)^T \Sigma^{-1} (x - \mu)},$$

where μ is the mean vector and Σ is the variance-covariance matrix of X

One can use the Mahalanobis distance, by computing the Mahalanobis distance between an observations x_i and the “center” of the k_{th} population μ_k , to carry out classification

Binary Classification with Multivariate Normal Populations

Assume $X_1 \sim \text{MVN}(\mu_1, \Sigma)$, $X_2 \sim \text{MVN}(\mu_2, \Sigma)$, that is,
 $\Sigma_1 = \Sigma_2 = \Sigma$

- Maximum Likelihood of group membership:

$$\text{Group 1 if } \ell(x, \mu_1, \Sigma) > \ell(x, \mu_2, \Sigma)$$

- Linear Discriminant Function:

$$\text{Group 1 if } (\mu_1 - \mu_2)^T \Sigma^{-1} x - \frac{1}{2}(\mu_1 - \mu_2)^T \Sigma^{-1}(\mu_1 + \mu_2) > 0$$

- Minimize Mahalanobis distance:

$$\text{Group 1 if } (x - \mu_1)^T \Sigma^{-1}(x - \mu_1) < (x - \mu_2)^T \Sigma^{-1}(x - \mu_2)$$

All the criteria above are equivalent in terms of classification

In addition to the observed characteristics of units $\{x_i\}_{i=1}^n$, other considerations of classification rules are:

- Prior probability:

If one population is more prevalent than the other, chances are higher that a new unit came from the larger population. Stronger evidence would be needed to allocate the unit to the population with the smaller prior probability.

- Costs of misclassification:

It may be more costly to misclassify a seriously ill subject as healthy than to misclassify a healthy subject as being ill.

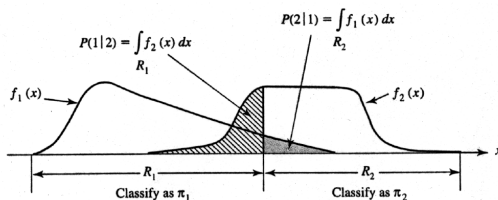
Classification Regions and Misclassifications

- The probability of misclassifying an object into π_2 when it belongs in π_1 is

$$P(2|1) = \mathbb{P}(X \in \mathcal{R}_2 | \pi_1)$$

- The probability of misclassifying an object into π_1 when it belongs in π_2 is

$$P(1|2) = \mathbb{P}(X \in \mathcal{R}_1 | \pi_2)$$



Source: Figure 11.3 from Applied Multivariate Statistical Analysis, 6th Ed (Johnson & Wichern). Visualization is for $p = 1$ variable.

Let p_1 and p_2 denote the prior probabilities of π_1, π_2 , and $c(1|2), c(2|1)$ be the costs of misclassification:

- Then probabilities of the four possible outcomes are:

$$\mathbb{P}(\text{correctly classified as } \pi_1) = \mathbb{P}(\mathbf{X} \in \mathcal{R}_1 | \pi_1) \mathbb{P}(\pi_1) = P(1|1)p_1$$

$$\mathbb{P}(\text{incorrectly classified as } \pi_1) = \mathbb{P}(\mathbf{X} \in \mathcal{R}_1 | \pi_2) \mathbb{P}(\pi_2) = P(1|2)p_2$$

$$\mathbb{P}(\text{correctly classified as } \pi_2) = \mathbb{P}(\mathbf{X} \in \mathcal{R}_2 | \pi_2) \mathbb{P}(\pi_2) = P(2|2)p_2$$

$$\mathbb{P}(\text{incorrectly classified as } \pi_2) = \mathbb{P}(\mathbf{X} \in \mathcal{R}_2 | \pi_1) \mathbb{P}(\pi_1) = P(2|1)p_1$$

- Classification rules are often evaluated in terms of the **expected cost of misclassification (ECM)**:

$$\text{ECM} = c(2|1)P(2|1)p_1 + c(1|2)P(1|2)p_2,$$

and we seek rules that **minimize the ECM**

Classification Rule and Special Cases of Minimum ECM Regions

The regions \mathcal{R}_1 , \mathcal{R}_2 that minimize the ECM are defined by the values of \mathbf{x} for which

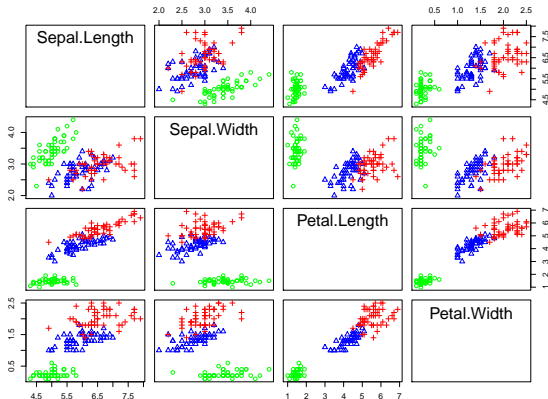
$$\mathcal{R}_1 : \frac{f_1(\mathbf{x})}{f_2(\mathbf{x})} > \left(\frac{c(1|2)}{c(2|1)} \right) \left(\frac{p_2}{p_1} \right)$$

$$\mathcal{R}_2 : \frac{f_1(\mathbf{x})}{f_2(\mathbf{x})} < \left(\frac{c(1|2)}{c(2|1)} \right) \left(\frac{p_2}{p_1} \right)$$

- if $p_1 = p_2 : \frac{f_1(\mathbf{x})}{f_2(\mathbf{x})} > \frac{c(1|2)}{c(2|1)} \Rightarrow \mathcal{R}_1$, otherwise \mathcal{R}_2
- if $c(1|2) = c(2|1) : \frac{f_1(\mathbf{x})}{f_2(\mathbf{x})} > \frac{p_2}{p_1} \Rightarrow \mathcal{R}_1$, otherwise \mathcal{R}_2
- if $c(1|2) = c(2|1)$ and $p_1 = p_2 : \frac{f_1(\mathbf{x})}{f_2(\mathbf{x})} > 1 \Rightarrow \mathcal{R}_1$, otherwise \mathcal{R}_2

Example: Fisher's Iris Data

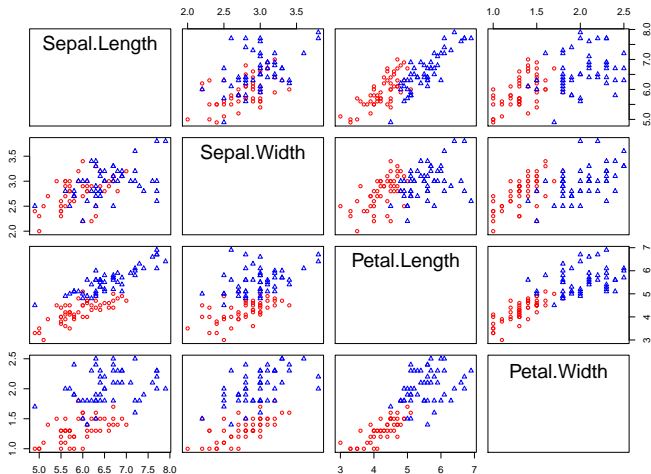
4 variables (sepal length and width and petal length and width),
3 species (**setosa**, **versicolor**, and **virginica**)



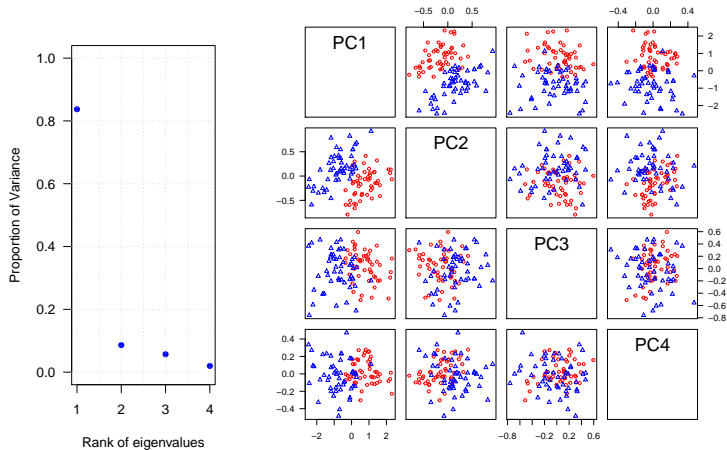
Task: Classify flowers into different species based on lengths and widths of sepal and petal

Fisher's Iris Data Cont'd

Let's focus on the latter two classes (**versicolor**, and **virginica**)



To further simplify the matter, let's focus on the first two PCs of X



Background

Binary Linear
Classification

Support Vector
Machines

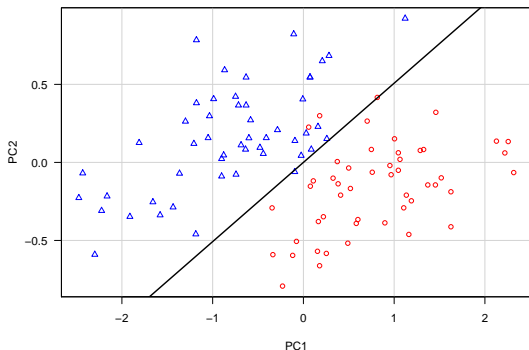
Linear Discriminant Analysis

Main idea: Use Bayes rule to compute

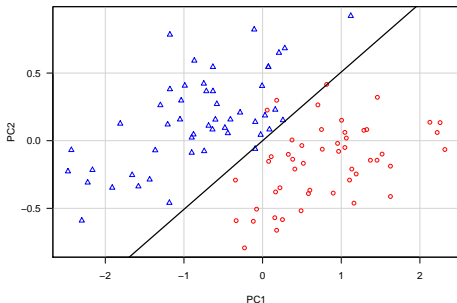
$$P(Y = k | \mathbf{X} = \mathbf{x}) = \frac{P(Y = k)P(\mathbf{X} = \mathbf{x} | Y = k)}{P(\mathbf{X} = \mathbf{x})} = \frac{\pi_k f_k(\mathbf{x})}{\sum_{k=1}^K \pi_k f_k(\mathbf{x})}.$$

Assuming $f_k(\mathbf{x}) \sim \text{MVN}(\boldsymbol{\mu}_k, \Sigma)$, $k = 1, \dots, K$ and use

$\hat{\pi}_k = \frac{n_k}{n} \Rightarrow$ it turns out the resulting classifier is **linear in \mathbf{x}**



Classification Performance Evaluation



```
fit.LDA
```

```
versicolor virginica
```

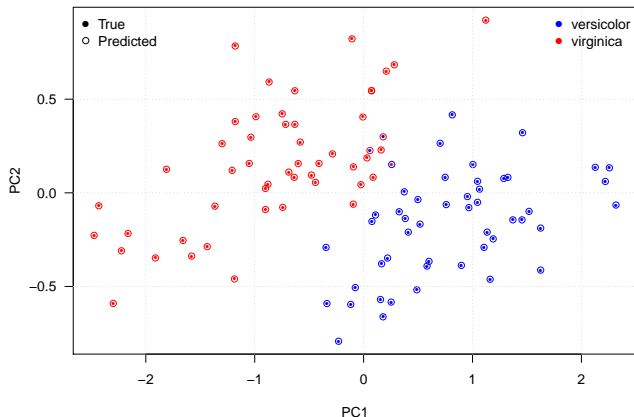
```
versicolor      47      3
```

```
virginica       1     49
```

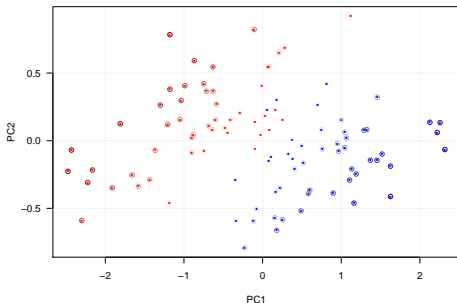
Misclassification rate: $\frac{3+1}{47+3+1+49} = 0.04$

Logistic Regression Classifier

Main idea: Model the logit $\log\left(\frac{P(Y=1)}{1-P(Y=1)}\right)$ as a linear function in \mathbf{x} (PC1 and PC2 in this case)



Logistic Regression Classifier Cont'd



	logisticPred	
	versicolor	virginica
versicolor	48	2
virginica	1	49

Misclassification rate: $\frac{2+1}{48+2+1+49} = 0.03$

Linear Discriminant Analysis Versus Logistic Regression

Classification



Background

Binary Linear
Classification

Support Vector
Machines

For a binary classification problem, one can show that both linear discriminant analysis (LDA) and logistic regression are **linear classifiers**. The difference is in how the parameters are estimated:

- Logistic regression uses the conditional likelihood based on $P(Y|X = x)$
- LDA uses the full likelihood based on multivariate normal assumption on X
- Despite these differences, in practice the results are often very similar

Quadratic Discriminant Analysis

In linear discriminant analysis, we **assume** $\{f_k(\mathbf{x})\}_{k=1}^K$ are normal densities and $\Sigma_1 = \Sigma_2$, therefore we obtain a **linear classifier**.

What if $\Sigma_1 \neq \Sigma_2$? \Rightarrow we get **quadratic discriminant analysis**

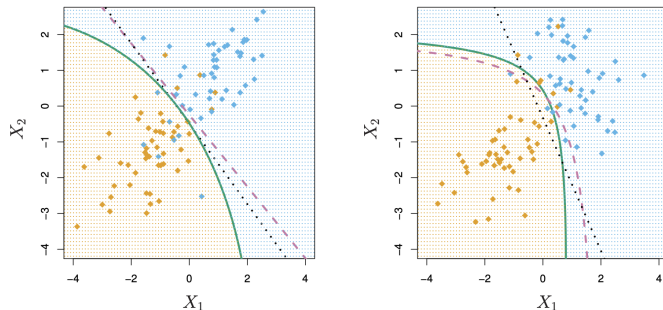
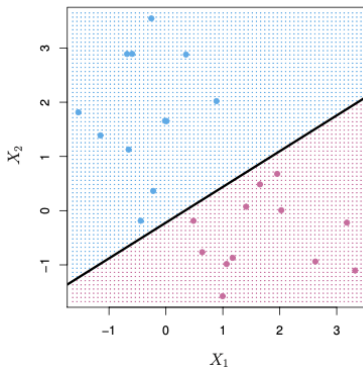
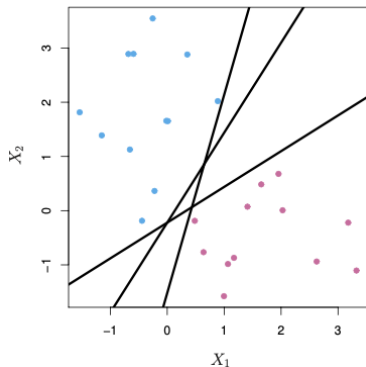


Figure courtesy of [An Introduction of Statistical Learning](#) by G. James et al. pp. 154

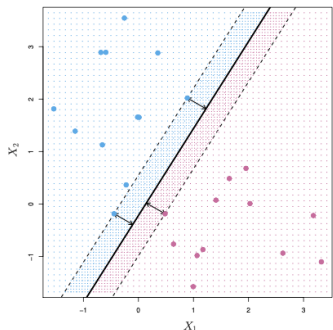
An Algorithmic Approach to Classification

Find a **hyperplane** that “best” separates the classes in feature space

- what we mean by “separateness”?
- what is the feature space?



Main idea: among all separating hyperplanes, find the one that creates the biggest gap (“margin”) between the two classes



doing so leads to the following optimization problem:

$$\text{maximize}_{\beta_0, \beta_1, \beta_2} M$$

$$\text{subject to } \sum_{j=1}^2 \beta_j^2 = 1,$$

$$y_i(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2}) \geq M,$$
$$i = 1, \dots, n$$

This problem can be solved efficiently using techniques from quadratic programming

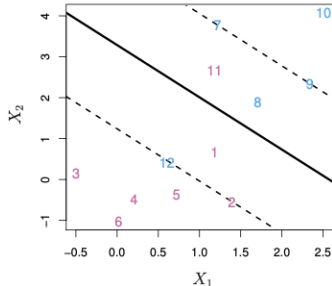
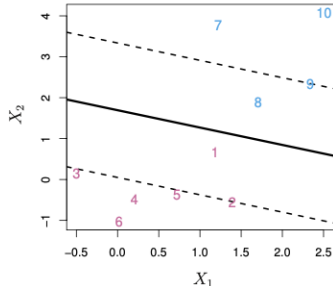
Background

Binary Linear
ClassificationSupport Vector
Machines

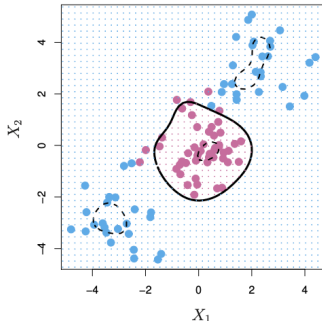
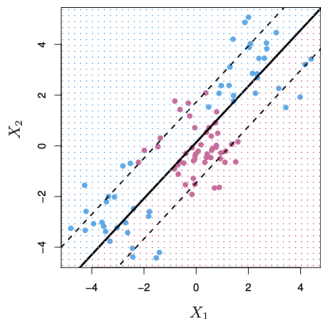
Support Vector Classifier

- Sometimes the data can not be separated by a line
- data can be noisy which leads to unstable maximal-margin classifier

The **support vector classifier** maximizes a “soft” margin



Beyond Linear Classifier



- A linear boundary can fail to separate classes
- Can expand the feature space by including transformations, e.g., $X_1^2, X_2^2, X_1X_2, \dots \Rightarrow$ gives non-linear decision boundaries in the original feature space
- However, polynomials basis can be unstable, a more general way to introduce non-linearities is through the use of kernels, e.g., $f(\mathbf{x}) = \beta_0 + \sum_{i \in \mathcal{S}} \hat{\alpha}_i \exp(-\gamma \sum_{j=1}^p (x_j - x_{ij})^2)$

SVM Versus Logistic Regression (LR) and LDA

- When classes are (nearly) separable, SVM does better than LR and LDA
- Use LR to estimate class probabilities as SVM is a non-probabilistic classifier
- For nonlinear boundaries, kernel SVMs are popular

In this lecture we learned about:

- Some classical classifiers for performing classification
- How to assess the efficacy of a classifier
- Support vector machines (SVMs)

R functions to know

- `lda/qda` from the `MASS` library
- `svm` from the `e1071` library

In the next lecture, we will learn about [Cluster Analysis](#)