THE UNIVERSITY OF SYDNEY

Lecture 5: Classification: overview

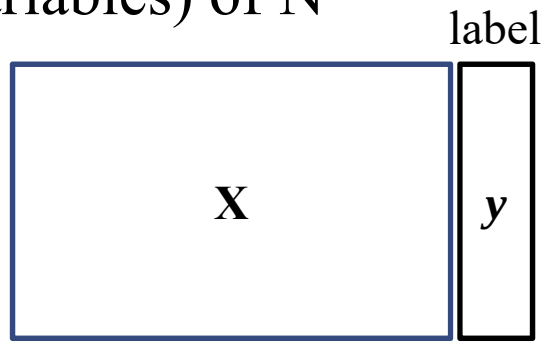
STAT5003

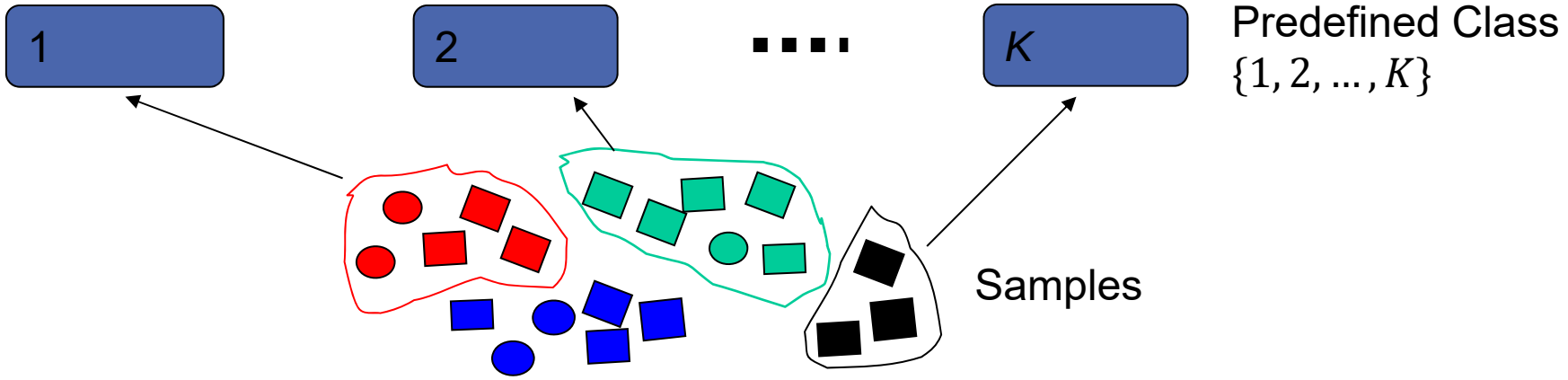
Pengyi Yang

Basic principles of classification

Each object associated with a class label (or response) $y \in \{1, 2, \ldots, K\}$ and a feature vector (vector of predictor variables) of N measurements: $\mathbf{X} = (x_1, \ldots, x_N)$

Aim: classify y using $\mathbf{X}$.





y = Class Label

X = Feature vector
{colour, shape}

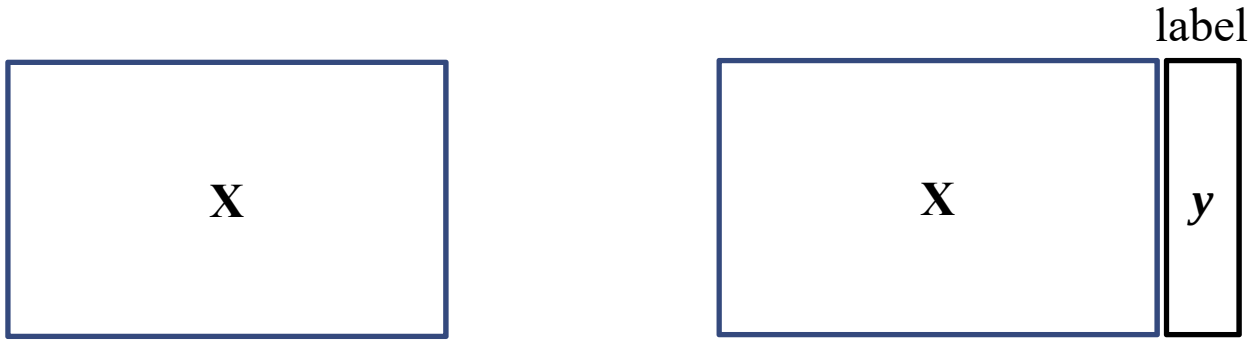
Classification rule ?

X = {red, square}
y = ?

Classification vs Clustering

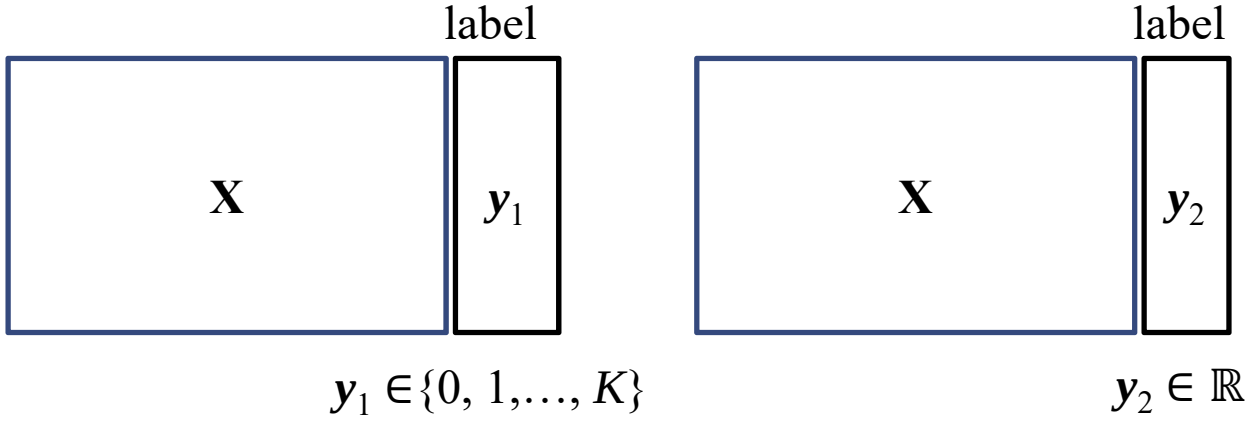
- Clustering: classes unknown, want to discover them from the data (unsupervised)

- Classification: classes are predefined, want to use a (training or learning) set of labeled objects to form a classifier for classification of future observations (supervised)

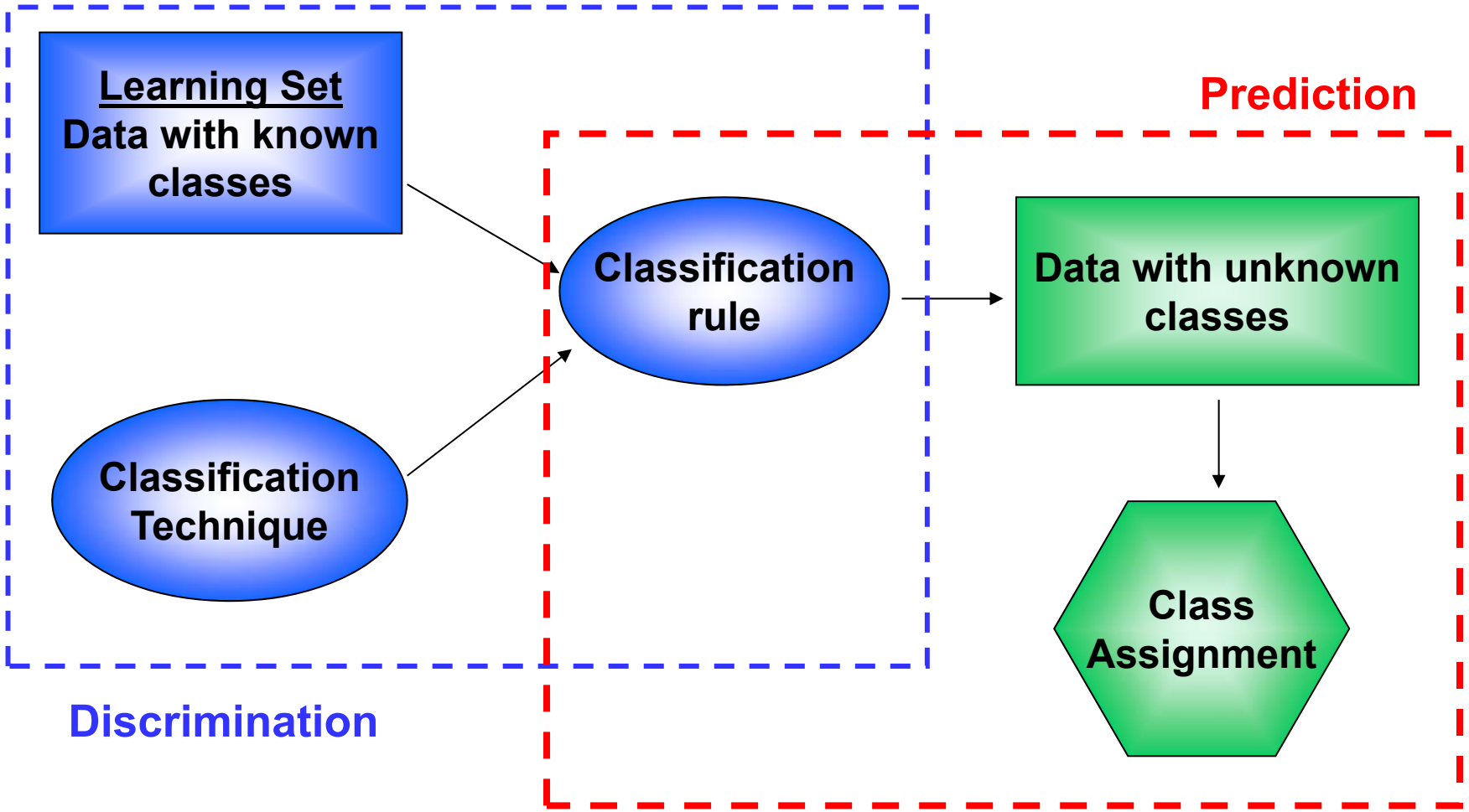


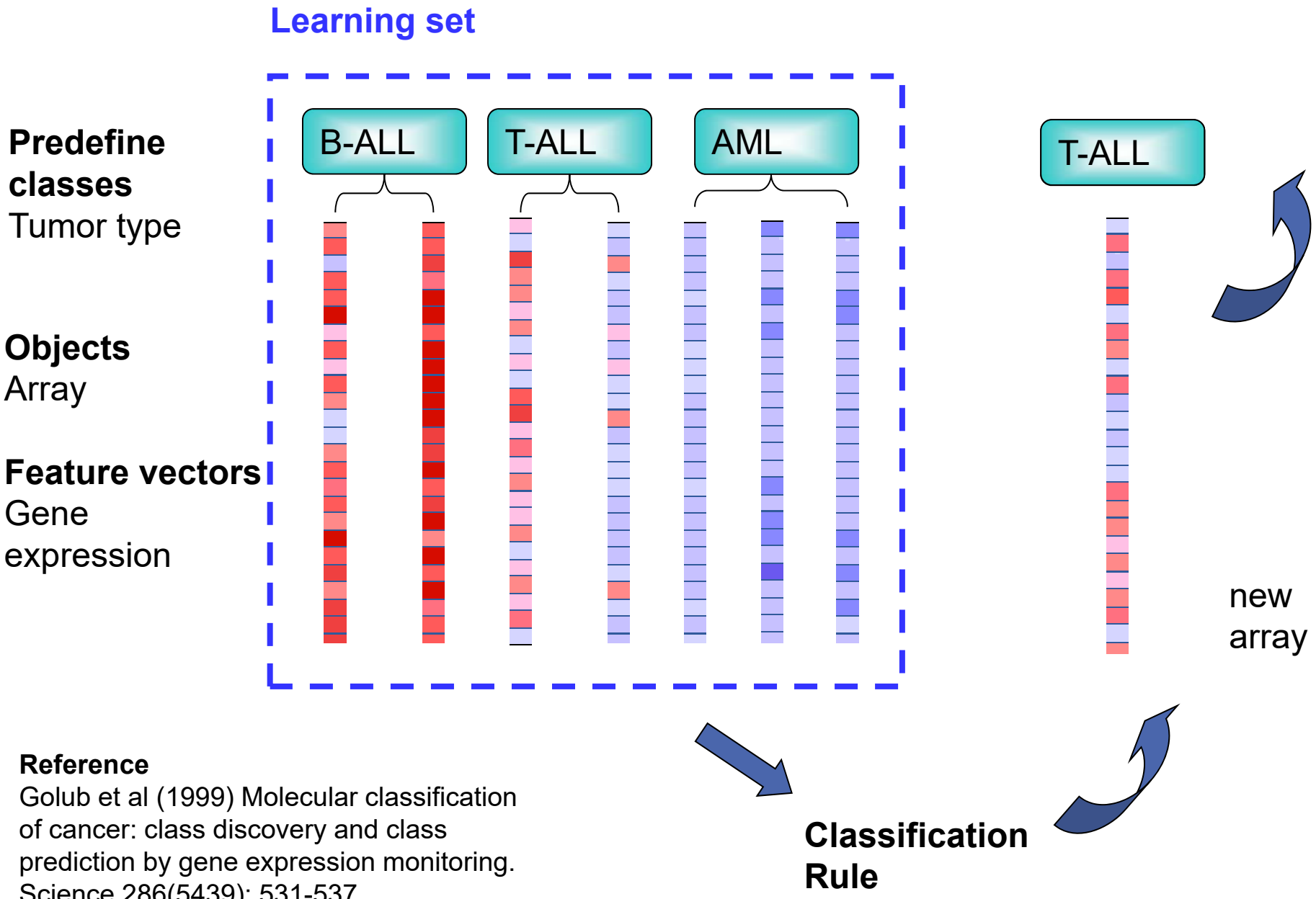
Classification vs Regression

- Regression: no class definition, the response variable is a continues value. Model the relationship between explanatory variables and the response variable.
- Classification: samples are predefined to be from a given class. Classification models produce a continuous valued prediction, which is usually in the form of a probability (i.e. the predicted values of class membership for any individual sample are between 0 and 1 and sum to 1). A predicted class is required in order to make a decision.



Classification

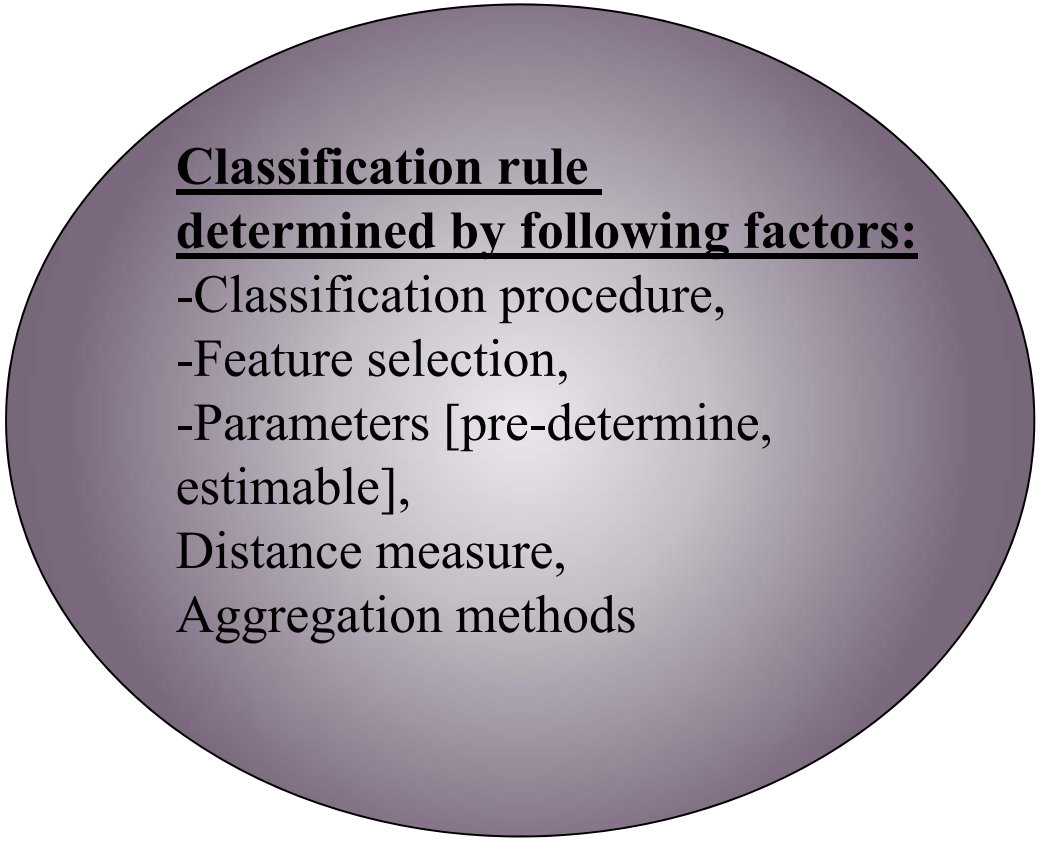




Reference

Golub et al (1999) Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. Science 286(5439): 531-537.

Performance
Assessment
e.g. Cross validation



- One can think of the classification rule as a black box, some methods provides more insight into the box.
- Performance assessment needs to be looked at for all classification rule.

Two class classification

- Classification of two class problem:
  - Email: Spam / Not Spam
  - Tumour: Malignant /Benign

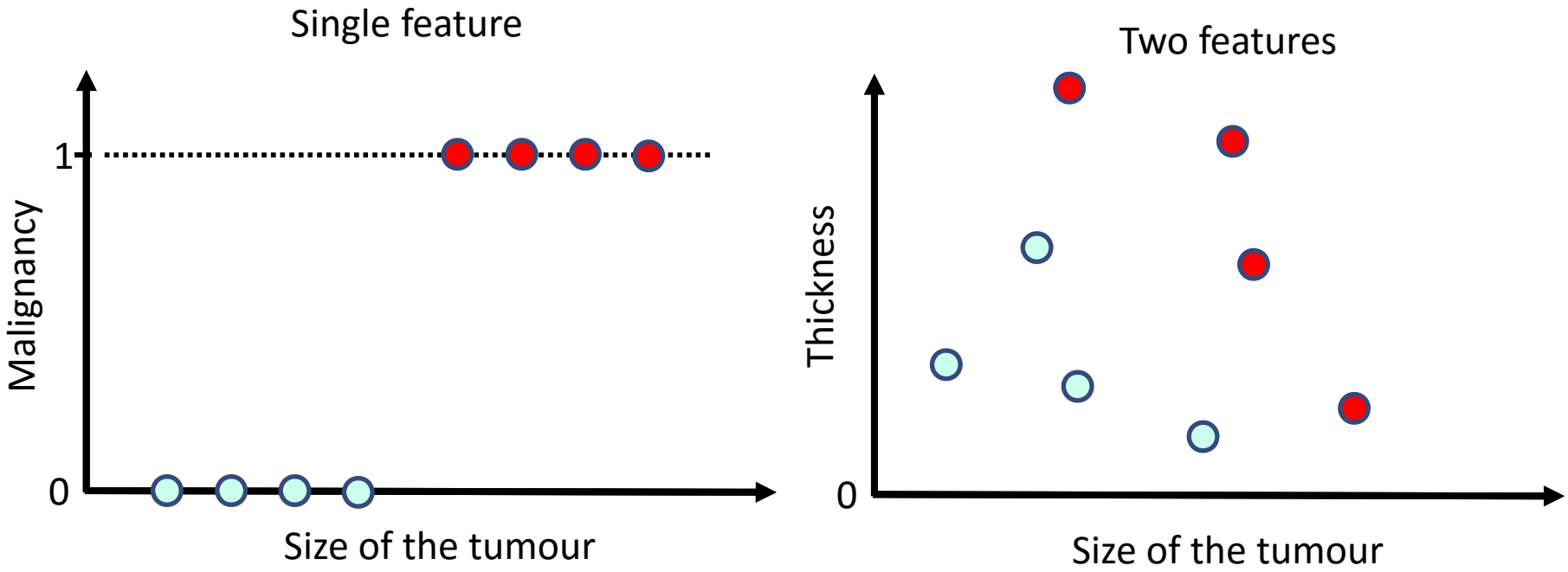
label a sample as:

$\boldsymbol{y} \in \{0, 1\}$

0: “negative class”

1: “positive class”

Problem setup

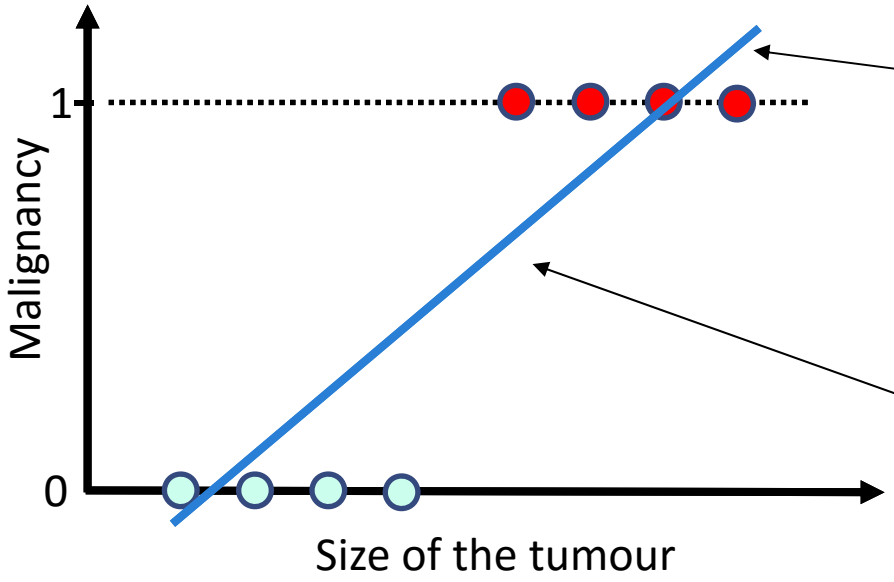


Threshold classifier output $h_\theta(x)$ at 0.5:

if $h_\theta(x) > 0.5$, predict "y=1"
if $h_\theta(x) < 0.5$, predict "y=0"

Why not using simple linear regression?

- When ***y*** only takes on values of 0 and 1, why standard linear regression is inappropriate?



How do we interpret values greater than 1?

How do we interpret values of ***y*** between 0 and 1?

Problems

- The regression line $\beta_0 + \beta_1 x$ can take on any value between negative and positive infinity
- In the tumour diagnosis problem, *y* can only take on two possible values: 0 or 1
- Therefore the regression line almost always predicts the wrong value for *y* in classification problems

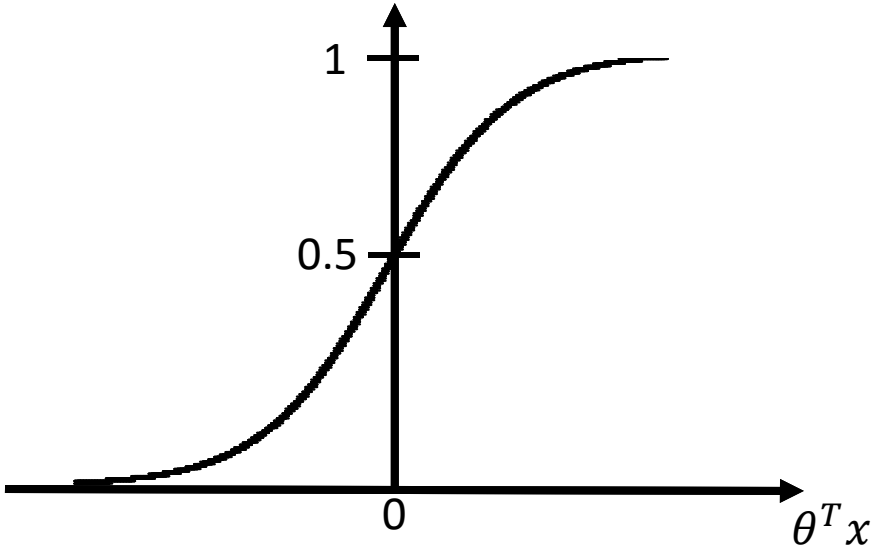
Logistic regression

Logistic regression model:

$$\log\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 x_1 + \cdots + \beta_P x_P = \boldsymbol{\theta}^T \boldsymbol{x}$$

Solve for p

$$p = \Pr(y = 1|\boldsymbol{x}) = h_{\boldsymbol{\theta}}(\boldsymbol{x}) = g(\boldsymbol{\theta}^T \boldsymbol{x}) = \frac{1}{1 + e^{-\boldsymbol{\theta}^T \boldsymbol{x}}}$$



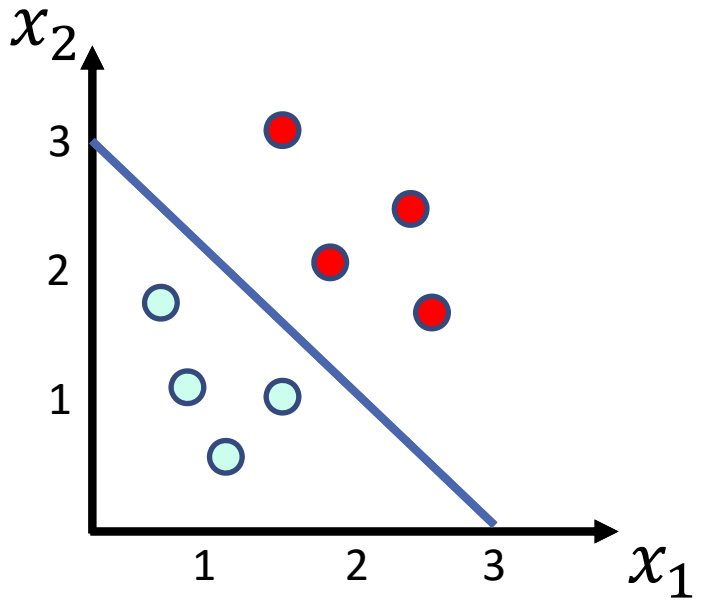
Logistic regression: decision boundary

Decision Boundary

$$\Pr(y = 1|\boldsymbol{x}) = h_{\boldsymbol{\theta}}(\boldsymbol{x})$$

$$= g(\boldsymbol{\theta}^T \boldsymbol{x}) = \frac{1}{1 + e^{-\boldsymbol{\theta}^T \boldsymbol{x}}}$$

Predict $y = 1$ if $\boldsymbol{\theta}^T \boldsymbol{x} \geq 0$



$$h_{\boldsymbol{\theta}}(\boldsymbol{x}) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2)$$

$$\boldsymbol{\theta} = \begin{bmatrix} -3 \\ 1 \\ 1 \end{bmatrix}$$

Maximum likelihood estimation of θ with single predictor

$$\Pr(y = 1|x) = g(\theta_0 + \theta_1 x)$$
$$\Pr(y = 0|x) = 1 - g(\theta_0 + \theta_1 x)$$
$$\Pr(y|x) = [g(\theta_0 + \theta_1 x)]^y [1 - g(\theta_0 + \theta_1 x)]^{1-y}$$

Likelihood function of n samples:

$$L = \prod_{i=1}^{n} [g(\theta_0 + \theta_1 x_i)]^{y_i} [1 - g(\theta_0 + \theta_1 x_i)]^{1-y_i}$$

$$\log(L) = \sum_{i=1}^{n} y_i \cdot \log[g(\theta_0 + \theta_1 x_i)] + (1 - y_i) \cdot \log[1 - g(\theta_0 + \theta_1 x_i)]$$

$$\frac{\partial \log(L)}{\partial \theta_0} = 0, \frac{\partial \log(L)}{\partial \theta_1} = 0$$

Demonstration

Linear Discriminant Analysis (LDA)

- LDA undertakes the same task as Logistic Regression. It classifies data based on categorical variables
  - Malignant or benign
  - Making profit or not
  - Buy a product or not
  - Satisfied customer or not

Logistic Regression vs LDA formulations

- With Logistic Regression we modeled the probability of Y being from the k^{th} class as

$$p(X) = \Pr(Y = k|X = x) = \frac{e^{\beta_0 + \beta_1 X}}{1 + e^{\beta_0 + \beta_1 X}}.$$

- However, Bayes' Theorem states

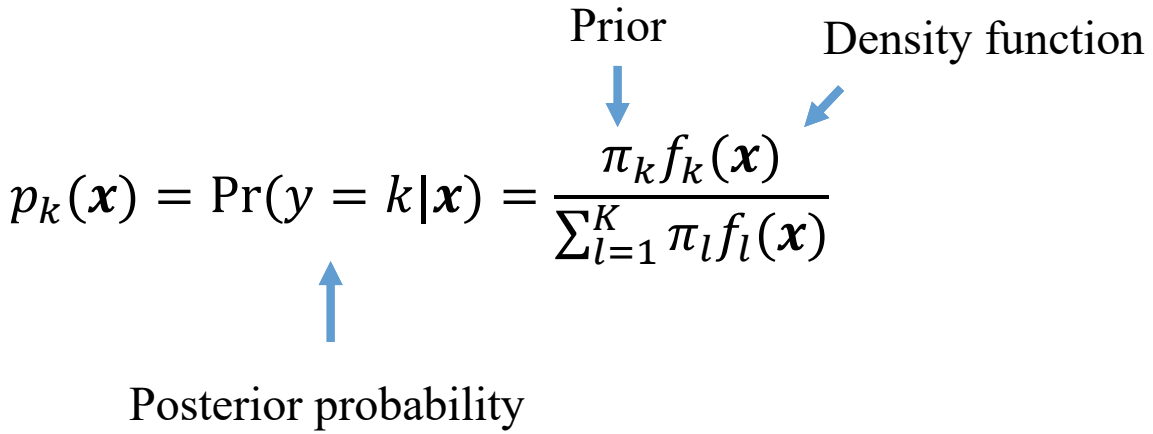
$$p(X) = \Pr(Y = k|X = x) = \frac{\pi_k f_k(x)}{\sum_{l=1}^{K} \pi_l f_l(x)}.$$

π_k : Probability of coming from class k (prior probability)

$f_k(x)$: Density function for X given that X is an observation from class k

Bayes' theorem

Bayes' theorem model the classification as:



LDA Estimates π_k and $f_k(x)$

- We can estimate π_k and $f_k(x)$ to compute $p(X)$

- The most common model for $f_k(x)$ is the *Normal Density* (LDA)

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

- Using the above density, we only need to estimate three quantities to compute $p(X)$

$$\mu_k \quad \sigma_k^2 \quad \pi_k$$

Use training data set for estimation

- The mean μ_k could be estimated by the average of all training observations from the k^{th} class.
- The variance σ_k^2 could be estimated as the weighted average of variances of all k classes.
- And, π_k is estimated as the proportion of the training observations that belong to the k^{th} class.

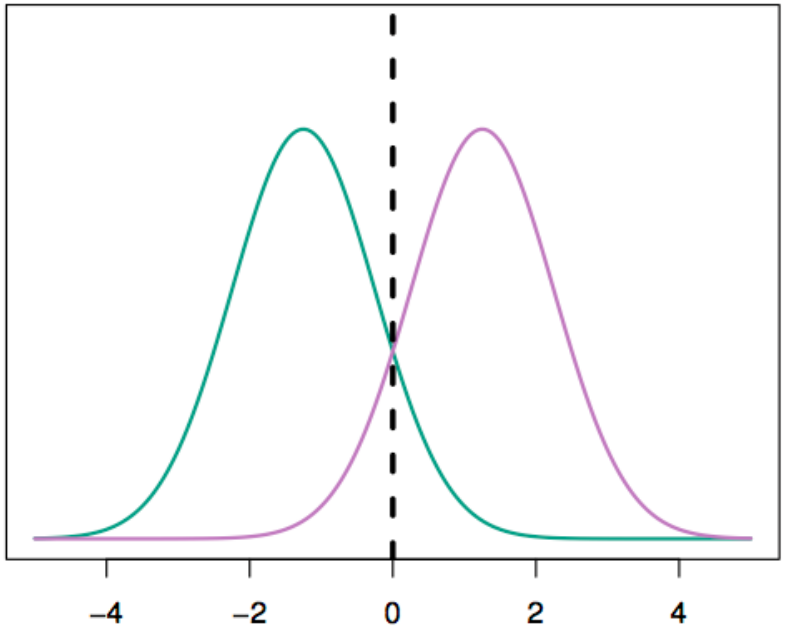
$$\hat{\mu}_k = \frac{1}{n_k} \sum_{i:y_i = k} x_i$$

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

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

A simple example with one predictor

- Suppose we have only one predictor
- Two normal density function $f_1(x)$ and $f_2(x)$, represent two distinct classes
- The two density functions overlap, so there is some uncertainty about the class to which an observation with an unknown class belongs
- The dashed vertical line represents Bayes' decision boundary



Deriving LDA for one predictor

Assuming that we are working with only one predictor

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

$$p_k(x) = \Pr(y = k|x) = \frac{\pi_k \frac{1}{\sqrt{2\pi\sigma_k}} \exp\left(-\frac{1}{2\sigma_k^2}(x-\mu_k)^2\right)}{\sum_{l=1}^{K} \pi_l \frac{1}{\sqrt{2\pi\sigma_l}} \exp\left(-\frac{1}{2\sigma_l^2}(x-\mu_l)^2\right)}$$

$$\log\left(p_k(x)\right) = x \cdot \frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + \log(\pi_k)$$

LDA decision boundary

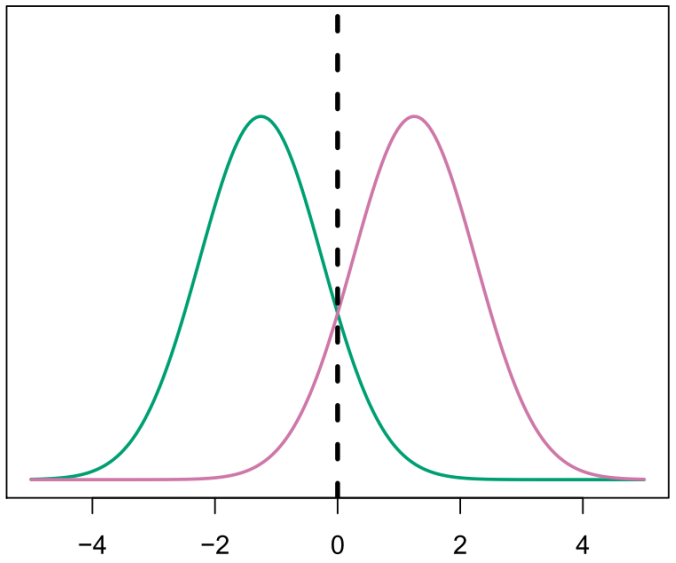
If K=2 and $\pi_1 = \pi_2$, then assigns an observation to class 1 if:

$$\log\left(p_1(x)\right) > \log\left(p_2(x)\right)$$

$$x \cdot \frac{\mu_1}{\sigma^2} - \frac{\mu_1^2}{2\sigma^2} + \log(\pi_1) - x \cdot \frac{\mu_2}{\sigma^2} + \frac{\mu_2^2}{2\sigma^2} - \log(\pi_2) > 0$$

$$2x(\mu_1 - \mu_2) > \mu_1^2 - \mu_2^2$$

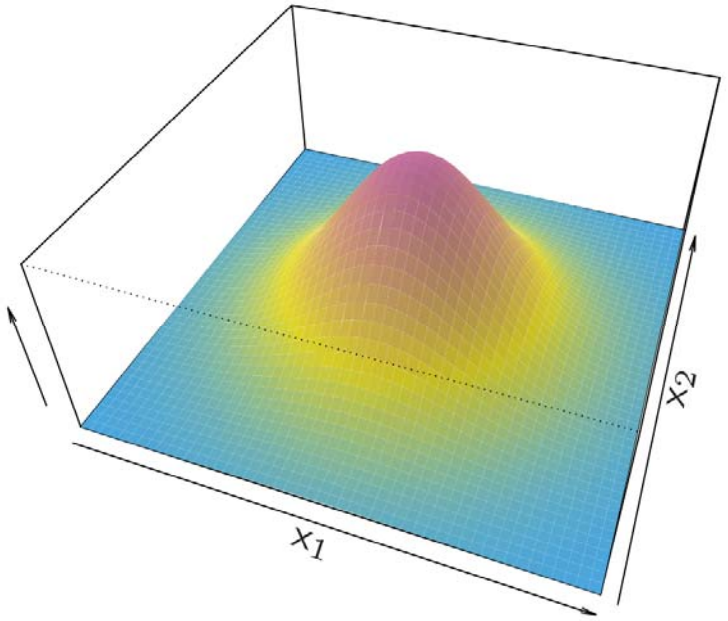
Decision Boundary $x = \dfrac{\mu_1^2 - \mu_2^2}{2(\mu_1 - \mu_2)} = \dfrac{\mu_1 + \mu_2}{2}$

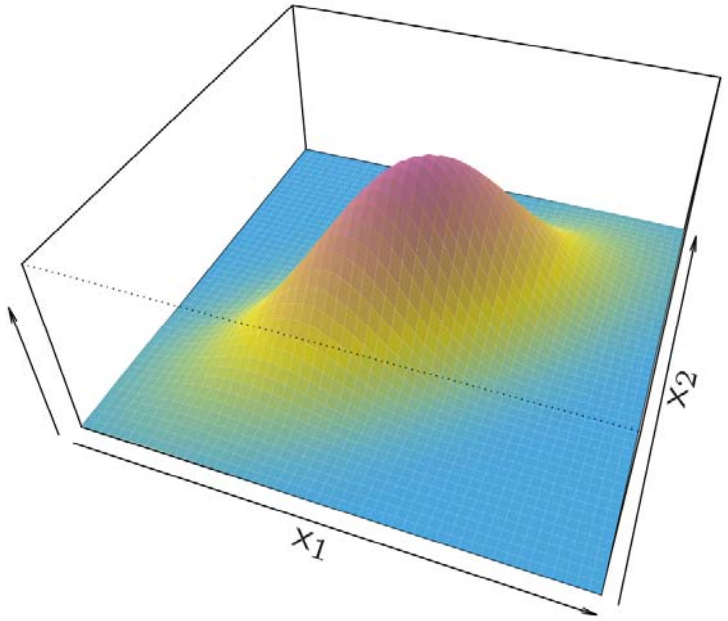


LDA more than one learning features

$$\log\left(p_k(x)\right) = x \cdot \frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + \log(\pi_k)$$ One feature

$$\log\left(p_k(\boldsymbol{x})\right) = \boldsymbol{x}^T \cdot \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_k - \frac{1}{2}\boldsymbol{\mu}_k^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_k + \log(\pi_k)$$ More than one feature





Demonstration

Why not logistic regression?

- Logistic regression is unstable when the classes are well separated
- In the case where n is small, and the distribution of predictors X is approximately normal, then LDA is more stable than Logistic Regression
- LDA is more popular when we have more than two response classes

More on logistic regression vs LDA

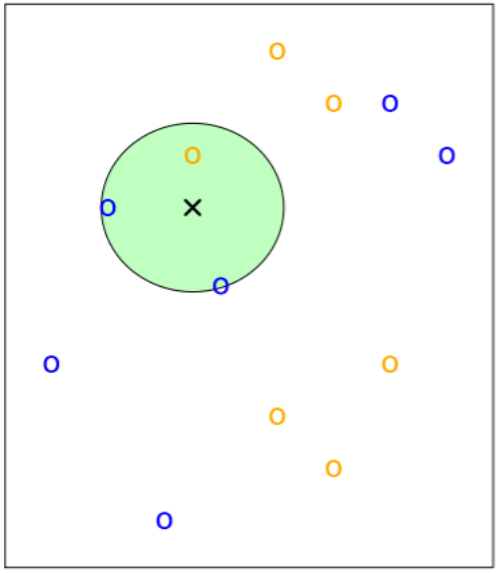
- Similarity: Both Logistic Regression and LDA produce linear boundaries

- Difference: LDA assumes that the observations are drawn from the normal distribution with common variance in each class, while logistic regression does not have this assumption.
- LDA would do better than Logistic Regression if the assumption of normality hold, otherwise logistic regression may outperform LDA

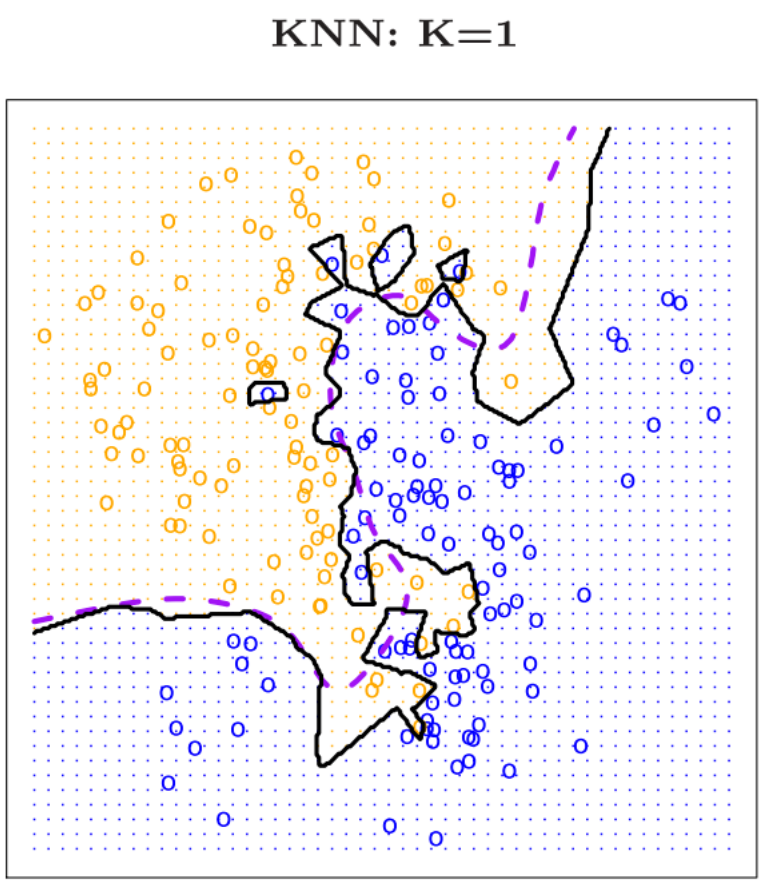
k-Nearest Neighbours (kNN)

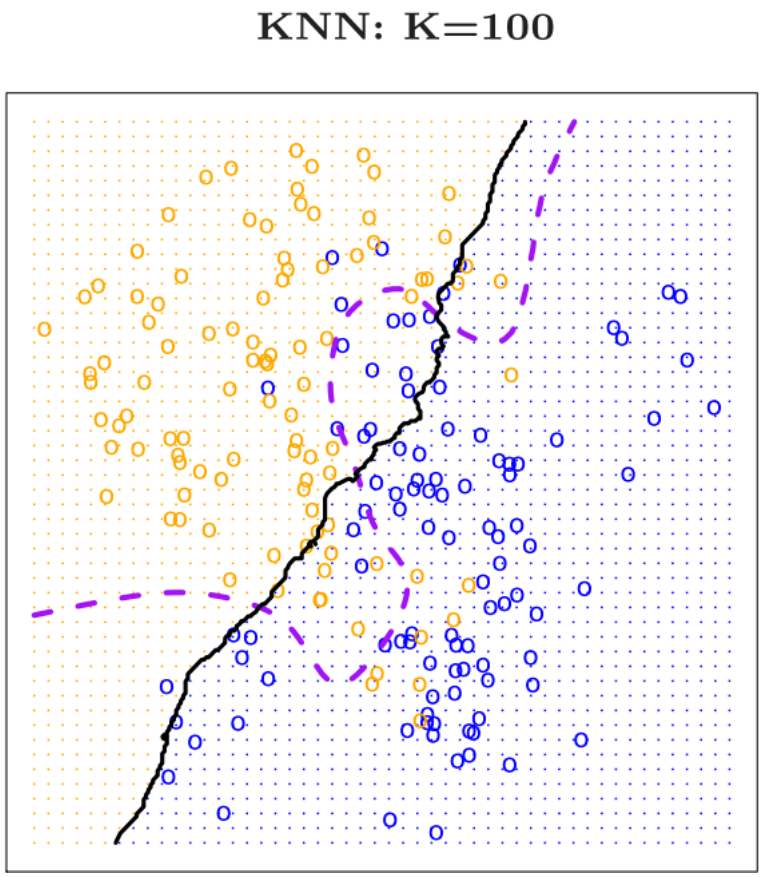
kNN model:

$$\Pr(y = c|\boldsymbol{x}) = \frac{1}{K}\sum_{\mathcal{N}_{\boldsymbol{x}}^{K}} I(y = c)$$



k-Nearest Neighbours (kNN) – continue





Bias-variance trade-off

Suppose we have fit a model $\hat{f}(x)$ to some training data Tr, and let (x_0, y_0) be a test observation drawn from the population. If the true model is $Y = f(X) + \epsilon$ (with $f(x) = E(Y|X = x)$), then

$$E\left(y_0 - \hat{f}(x_0)\right)^2 = \text{Var}(\hat{f}(x_0)) + [\text{Bias}(\hat{f}(x_0))]^2 + \text{Var}(\epsilon).$$

The expectation averages over the variability of y_0 as well as the variability in Tr.

Typically as the *flexibility* of $\hat{f}$ increases, its variance increases, and its bias decreases. So choosing the flexibility based on average test error amounts to a *bias-variance trade-off.*

Refer back to the figure in previous slide, the large the *k* the less flexible the decision boundary (high bias low variance), the smaller the *k* the more flexible the decision boundary (low bias high variance)

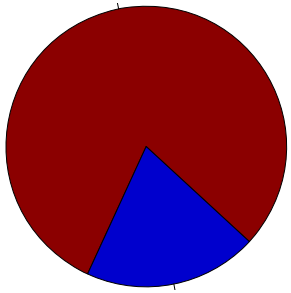
Demonstration of kNN

kNN vs (LDA and Logistic Regression)

- kNN takes a completely different approach
- kNN is completely non-parametric: No assumptions are made about the shape of the decision boundary
- Advantage of kNN: We can expect kNN to dominate both LDA and Logistic Regression when the decision boundary is highly non-linear
- Disadvantage of kNN: kNN does not tell us which predictors are important (no table of coefficients)

Classification evaluation based on class labels
Training and testing errors

N training samples



T test samples

Training error rate: $\frac{1}{N}\sum_{i=1}^{N} I(y_i \neq \hat{y}_i)$

Test error rate: $\frac{1}{T}\sum_{t=1}^{T} I(y_t \neq \hat{y}_t)$

The “best” classifier is the one for which the test error is the smallest.

Test error not training error should be used for classification evaluation