Week 2: Linear Regression & Some Fundamental Notions and Tools in Machine Learning

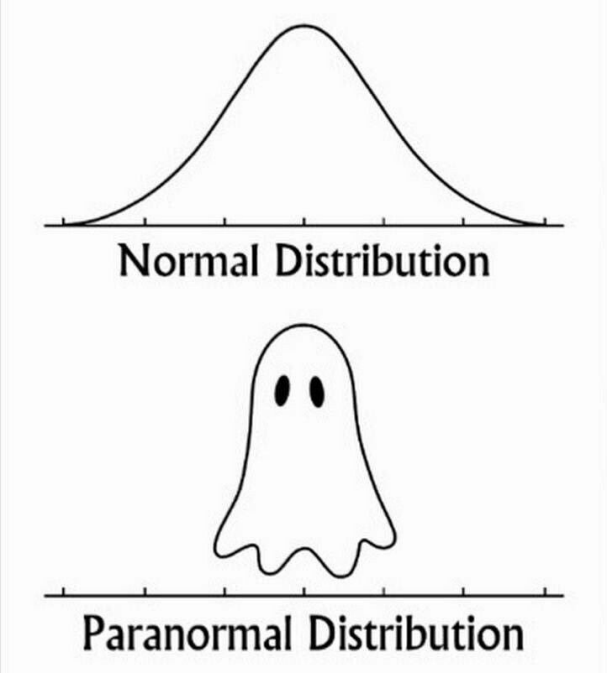
AI 539: Machine Learning for Non-Majors

Alireza Aghasi

Oregon State University

This Lecture ...

– Would be surprised to know how much of Statistics is about Normality!



Some Basic Probability Overview

- For a continuous random variable X we often define a probability density distribution $f_X(x)$ where $\mathbb{P}(a \leq X \leq b) = \int_a^b f_X(x)\mathrm{d}x$.
- A random variable X is normally distributed with mean μ and variance σ^2 (denoted as $\mathcal{N}(\mu, \sigma^2)$), when

$$f_X(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

- Sum of normals: if $x_1, \ldots, x_n$ are normal (**not necessarily independent**) then the weighted sum $\alpha_1 x_1 + \ldots \alpha_n x_n$ is also normal
- Expectation of the weighted sum: if $x_1, \ldots, x_n$ are random variables with mean $\mathbb{E}(x_i) = \mu_i$, then for constants α_i:

$$\mathbb{E}(\alpha_1 x_1 + \ldots \alpha_n x_n) = \alpha_1\mu_1 + \ldots + \alpha_n\mu_n$$

- Variance of the weighted sum: if $x_1, \ldots, x_n$ are **independent** random variables with variance $var(x_i) = \sigma_i^2$, then

$$var(\alpha_1 x_1 + \ldots \alpha_n x_n) = \alpha_1^2\sigma_1^2 + \ldots + \alpha_n^2\sigma_n^2$$

$X_1 : \mathbb{E}X_1 = 1 \quad , \quad \mathrm{Var}(X_1) = 2$

$X_2 : \mathbb{E}X_2 = 2, \quad \mathrm{Var}(X_2) = 4$

$$X = 3 - X_1 + 2X_2$$

$$\mathbb{E}[X] = 3 - 1 + 2 \times 2 = 6$$

$$\mathrm{Var}(X) = 0 + (-1)^2 \times 2 + 2^2 \times 4 = 18$$

A Brief Review of Hypothesis Testing

Hypothesis Testing

- A hypothesis is often a conjecture about one or more populations
- To prove a hypothesis is true we need to examine all the population which is often not practical, **instead we take a random sample** and probabilistically assess if we have enough evidence to support a hypothesis
- **Example**: All human beings respond well to a specific treatment H_0 ?
  - Instead of testing it on all people on the planet, we look into a fraction of people who are infected by the decease and see if the treatment works
- Since we are focused on a limited sample, we can only state our confidence about the conjecture probabilistically

H_1: not everyone respond well

Hypothesis Testing

- Hypothesis testing is often formulated in terms of two hypotheses
  - H_0: the null hypothesis
  - H_1: the alternate hypothesis
- The hypothesis we want to test is if H_1 is likely to be true
- Often, the equality hypothesis is chosen to be the null hypothesis
- For many problems that we encounter in this course, Hypothesis Testing is simply testing the chances of a random variable to be in regions defined by H_0 or H_1 (wait for the numerical example)

Hypothesis Testing

α { is the confidence of the test
smaller is better

① assert H_0

② $1-\alpha \geq 95\%$?

- We want to test if H_1 is likely to be true
- You decide to assert something about the null hypothesis H_0 and be certain what you assert is true with probability at least $1 - \alpha$
- **Example**: We are 90% confident that this drug works on patients with xxx decease
- So normally you decide on how confident you want to make an assertion and determine a value for α

control the confidence

Hypothesis Testing

In hypothesis testing only one of these two cases happens:

1. – You **reject** H_0 and **accept** H_1 since you have enough evidence in favor of H_1
2. – You **fail to reject** H_0, since you don not have enough evidence to support H_1

While you see a lot of documents talking about "*accepting* H_0", technically you should use the term "*fail to reject*"

- It might be the case that H_0 is false, but your data is not enough to reject it (does not mean you should accept it)

Example: H_0: Tim is innocent H_1: Tim is guilty

If you have enough to support Tim is guilty, you reject H_0. If you do not have enough to show that Tim is guilty (failure to reject H_0), that does not mean he is innocent (accepting H_0)

Hypothesis Testing: Types of Error

	Reject H_0 (accept H_1)	Do not reject H_0
H_0 true in reality (probability)	Type I error α	Correct decision $1-\alpha$ $\geq 95\%$
H_1 true in reality (probability)	Correct decision $1-\beta$	Type II error β

- α is a small number that we determine and is called the significance level (the probability of making type I error)
- We decide on how confident we want to make a claim in favor of H_0 and $1-\alpha$ is our confidence about this
- Normally people take α to be 0.05 or 0.01, giving you 95% or 99% chance of validity in making the argument in support of H_0
- We also have a type II error (calling $1-\beta$ the power of the test), but here we do not want to focus on that

Hypothesis Testing Example

- In the context of our linear regression problem we are interested in hypothesis testing problems on the basis of samples

Example: There is a normal distribution with variance 1 and unknown mean μ. We take 10 independent samples x_i of this distribution as:

`1.8978, 1.7586, 2.3192, 2.3129, 1.1351, 1.9699, 1.8351, 2.6277, 3.0933 , 3.1093`

we add up these numbers and divide it by 10 (taking the sample mean) and observe that

$$\frac{x_1 + \ldots x_{10}}{10} = 2.2059.$$ take average

We get a feeling that probably $\mu = 2$, so we decide to test this hypothesis:

$$H_0 : \mu = 2 \qquad vs \qquad H_1 : \mu \neq 2 \text{ (two sided test).}$$

assertion counter assertion

Hypothesis Testing Example

Solution: We generally look into the behavior of the random variable

$$\bar{x} = \frac{x_1 + \ldots + x_{10}}{10}$$

$\bar{X} \sim N(\mu, 0.1)$

which is a normally distributed random variable with mean μ and variance 0.1 [can you say why?]. As a result, $z = \frac{\bar{x}-\mu}{\sqrt{0.1}}$ is a standard $\mathcal{N}(0,1)$ random variable [can you say why?].

We refer to z as the **test statistic**.

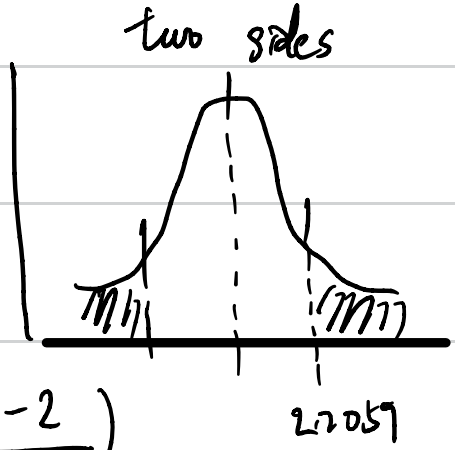
p-value: *is a useful quantity in the analysis of the test and is the probability of obtaining a result equal or "more extreme" than what we have observed, given that the null hypothesis is true. In the case of this example:* **assuming that $\mu = 2$, we perfectly know the distribution of $\bar{x}$ (or z) and want to asses how likely it is to draw a sample this far towards the tails and further (i.e., away from the mean of $\bar{x}$)**:

$$\begin{aligned}\text{p-value} =& \mathbb{P}\left(\bar{x} \geq 2.2059, \bar{x} \leq 2\mu - 2.2059 \mid \mu = 2\right) \\ =& \mathbb{P}\left(|z| \geq \frac{2.2059 - \mu}{\sqrt{0.1}} \,\middle|\, \mu = 2\right) = \mathbb{P}\left(|z| \geq 0.6511\right) = 0.5150.\end{aligned}$$

$\bar{X} \sim N$

$$E(\bar{X}) = E\left[\frac{1}{10}X_1 + \cdots + \frac{1}{10}X_{10}\right] = \frac{1}{10}E[X_1] + \cdots + \frac{1}{10}E[X_{10}] = \frac{1}{10}\mu + \cdots + \frac{1}{10}\mu = \mu$$

$$Var[\bar{X}] = Var\left[\frac{1}{10}X_1 + \cdots + \frac{1}{10}X_{10}\right] = \frac{1}{100}$$



P-value:

$$= P(\bar{X} \geq 2.2059,\ \bar{X} \leq 2\mu - 2.2059 \mid \mu = 2)$$

$$= P\left(\frac{\bar{X}-2}{\sqrt{0.1}} \geq \frac{2.2059-2}{\sqrt{0.1}},\ \frac{\bar{X}-2}{\sqrt{0.1}} \leq \frac{4-2.2059-2}{\sqrt{0.1}}\right)$$

$$Z = \frac{\bar{X}-2}{\sqrt{0.1}} \Rightarrow P\left(|Z| \geq \frac{2.2059-2}{\sqrt{0.1}}\right) = 0.515$$

Hypothesis Testing Example

Suppose our significance level is $\alpha = 0.05$.

> If p-value$\leq \alpha$: reject H_0 (accept H_1)
> If p-value$> \alpha$: fail to reject H_0

For our example p-value $= 0.5150 > 0.05$, so we cannot reject the hypothesis that $\mu = 2$. Not enough prove for $\mu = 2$

- If the value of $\bar{x} = 2.2059$ was obtained by taking the sample mean over 100 samples then we had $z = \frac{\bar{x}-\mu}{\sqrt{0.01}}$ and

$$\text{p-value} = \mathbb{P}\left(|z| \geq \frac{2.2059 - \mu}{\sqrt{0.01}} \,\middle|\, \mu = 2\right) = \mathbb{P}\left(|z| \geq 2.059\right)$$
$$= 0.0395 < 0.05,$$

  then we were able to reject H_0.
- In other words we are more than 95% confident (accurately, 96.05% confident) that it is not possible to take the sample mean over 100 random numbers of mean $\mu = 2$ and variance 1, and get a value as far from 2 as 2.2059! [Lets try it on Matlab]

Hypothesis Testing Example: Unknown Variance

Suppose in the previous example we did not know σ and instead of working with the standard random normal variable $z = \frac{\bar{x}-\mu}{\frac{\sigma}{\sqrt{n}}}$ we work with the random variable

$$\frac{\bar{x}-\mu}{\frac{s}{\sqrt{n}}} \qquad \text{where :} \qquad s = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i-\bar{x})^2}$$

for t-distribution

Note that s is an unbiased estimate of the standard deviation (since we don't know σ this is the best we can use).

If we want to go through a similar hypothesis test, we need to look into the test statistic $t = \frac{\bar{x}-\mu}{\frac{s}{\sqrt{n}}}$ which is no more normally distributed. It has a more complicated distribution called *Student's t distribution*.

So for the probabilities needed to calculate the p-value, we need to refer to t-distribution tables instead of the normal distribution tables.

Your Take Away from Hypothesis Testing

- We have some independent samples from a distribution and we guess something about our data
- We form a hypothesis test with some null and alternate hypotheses
- We fix some value for the significance α, meaning that $1 - \alpha$ is how confident we want to be in making our claim
- We form a test statistic (the resulting random variable can have a very complicated distribution)
- We calculate the p-value:
  - If p-value$\leq \alpha$: reject H_0 (accept H_1)
  - If p-value$> \alpha$: fail to reject H_0

Now Lets Start Linear Regression!

Introduction to Linear Regression

- You remember we had an ideal "regression function" $f(\boldsymbol{x})$, which was the actual function behind our data generation and our observations y were in the form

$$y = f(\boldsymbol{x}) + \epsilon$$

- Note that our input vector $\boldsymbol{x}$ contained all the input features, i.e.,

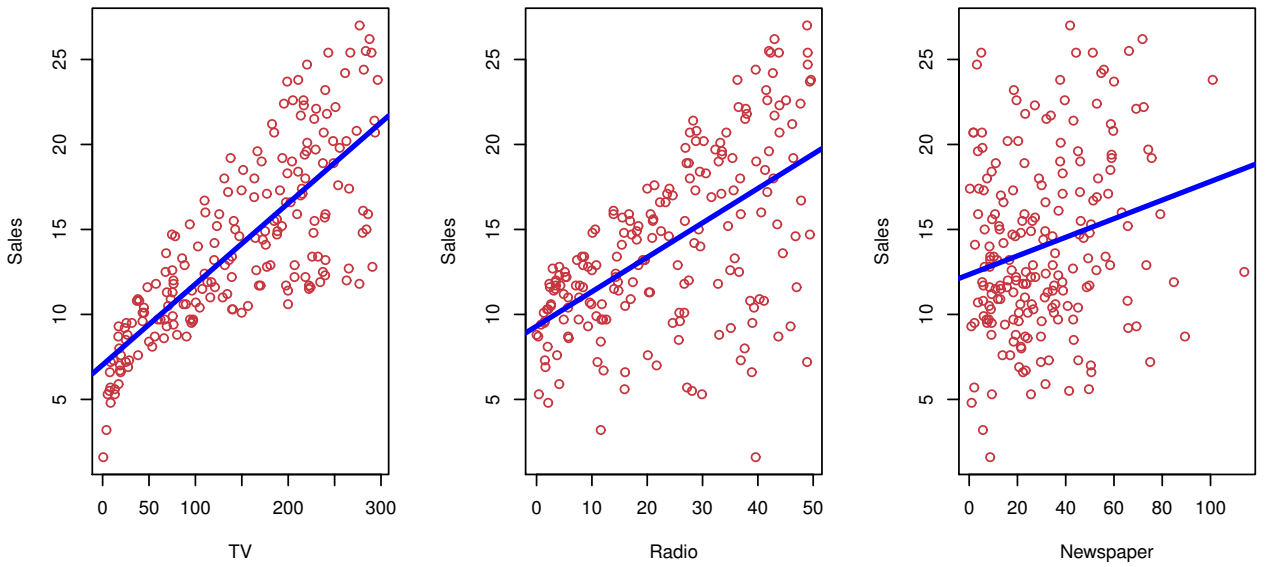
$$\boldsymbol{x} = [x_1, \ldots, x_p]^\top$$

- We had no access to $f(\boldsymbol{x})$, but we wanted to estimate it with some function $\hat{f}$
- In **linear regression** we search for a $\hat{f}$, which takes the following form

$$\hat{f}(\boldsymbol{x}) = \beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p$$

Introduction to Linear Regression

Lets start with a very simple model that only uses one feature



$$\text{Sales} = \beta_0 + \beta_1 \text{TV} + \epsilon$$ → one factor one time

or more generally,

$$y = \beta_0 + \beta_1 x + \epsilon,$$

where x is the only available feature

More on Simple Linear Regression

- We consider the model

intercept slope

$$y = \beta_0 + \beta_1 x_1 + \epsilon$$

  - β_0 and β_1 are two unknown constants that represent the intercept and slope, also known as coefficients and ϵ is the error term.
  - We are given samples of the form $(x_1, y_1), \ldots, (x_n, y_n)$, using which we try to fit some values $\hat{\beta}_0$ and $\hat{\beta}_1$ for the model coefficients
  - After this fit we can predict future responses to a test sample x_t, using

$$\hat{y}_t = \hat{\beta}_0 + \hat{\beta}_1 x_t,$$

Determining the Model Coefficients

$$\hat{f}(x) = \hat{\beta}_0 + \hat{\beta}_1 x_i + \varepsilon_i$$

– Remember that we had samples $(x_1, y_1), \ldots (x_n, y_n)$ and we would like to determine $\hat{f}(x)$ by

$$\min \sum_{i=1}^{n} (y_i - \hat{f}(x_i))^2 = \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_i)^2$$

– Using our simple model we would like to decide β_0 and β_1 such that the **Residual Sum of Squares** (RSS)

$$RSS = \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_i)^2$$

is minimized

minimize distance

find β_0, β_1 to minimize $\sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_i)^2 = L(\beta_0, \beta_1)$

$$0 = \frac{\partial L}{\partial \beta_0} = 2 \cdot \sum_{i=1}^{n} (-y_i + \beta_0 + \beta_1 x_i) \Rightarrow \sum_i \beta_0 + \underbrace{\beta_1 \sum_i x_i}_{n \cdot \bar{x}} = \underbrace{\sum y_i}_{n \cdot \bar{y}}$$

$$0 = \frac{\partial L}{\partial \beta_1} = 2 \sum_{i=1}^{n} x_i (\beta_0 + \beta_1 x_i - y_i) \Rightarrow \beta_0 \sum_i x_i + \beta_1 \sum_i x_i^2 = \sum x_i y_i$$

$$\Rightarrow \begin{cases} n \cdot \beta_0 + \left(\sum_i x_i\right) \beta_1 = \sum_i y_i \\ \left(\sum_i x_i\right) \beta_0 + \left(\sum_i x_i^2\right) \cdot \beta_1 = \sum x_i y_i \end{cases}$$

Determining the Model Coefficients

- Taking the derivative with respect to β_0 and β_1 and setting it to zero, for $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$ and $\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$ we get

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x} \qquad \text{and} \qquad \hat{\beta}_1 = \frac{\sum_{i=1}^{n} x_i y_i - n\bar{x}\bar{y}}{\sum_{i=1}^{n} x_i^2 - n\bar{x}^2}$$

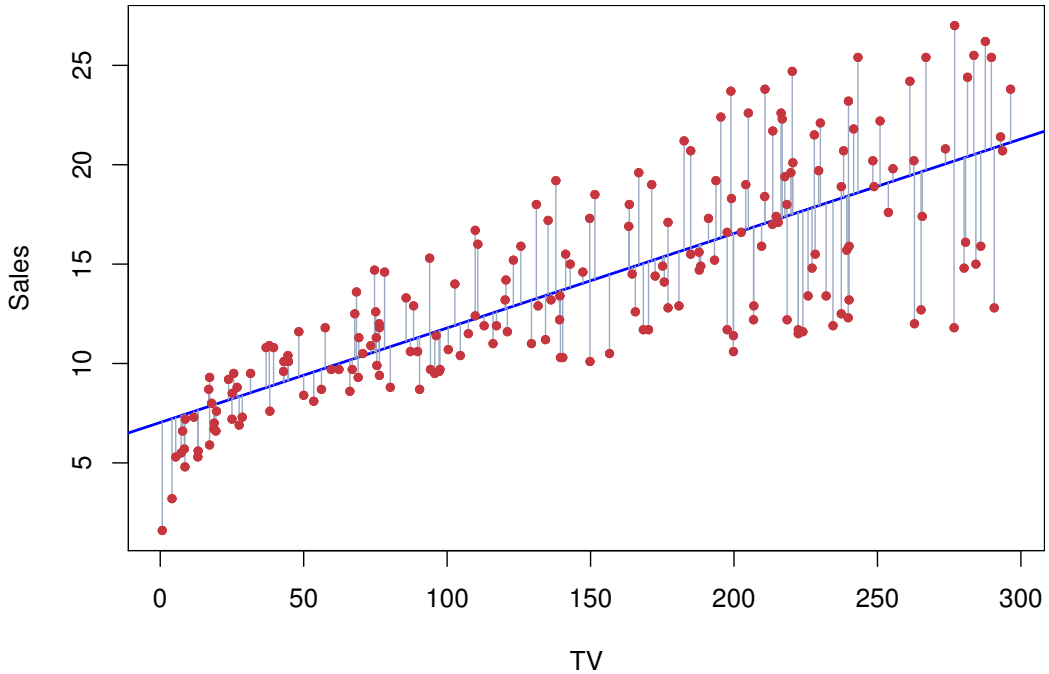
- We may use the simple equalities:

$$\sum_{i=1}^{n} x_i y_i - n\bar{x}\bar{y} = \sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y}), \quad \sum_{i=1}^{n} x_i^2 - n\bar{x}^2 = \sum_{i=1}^{n}(x_i - \bar{x})^2$$

to get the final equations

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2}, \quad \hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}.$$

How Well Did We Do the Fit?



Now that we have our fit we would like to address few questions about it!

(`We code up the answer to each question in the Example Code 1`)

Baseline: $y = \beta_0 + \beta_1 x$

Gold's Model

The estimates are unbiased: $\begin{cases} E[\hat{\beta}_1] = \beta_1 \\ E[\hat{\beta}_0] = \beta_0 \end{cases}$

$$\hat{\beta}_1 = \frac{\sum (x_i - \bar{x}) y_i}{\sum (x_i - \bar{x})^2}$$

$$y_i = \beta_0 + \beta_1 x_i + \varepsilon_i$$

Given $Var(\varepsilon) = \sigma^2 \rightarrow Var(y_i) = \sigma^2$

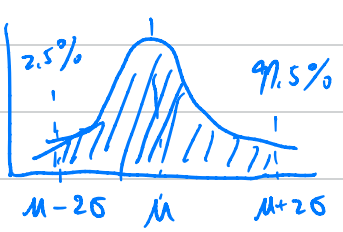
$$Var(\hat{\beta}_1) = Var\left[\frac{\sum (x_i - \bar{x}) y_i}{\sum (x_i - \bar{x})^2} \right]$$

$$= Var\left[\frac{(x_1 - \bar{x}) y_1}{\sum [x_i - \bar{x}]^2} + \cdots + \frac{(x_n - \bar{x}) y_n}{\sum [x_i - \bar{x}]^2} \right]$$

Since all y_i are independent to each other

$$= \sigma^2 \frac{(x_1 - \bar{x})^2}{\left[\sum (x_i - \bar{x})^2\right]^2} + \cdots + \sigma^2 \frac{(x_n - \bar{x})^2}{\left[\sum (x_i - \bar{x})^2\right]^2}$$

$$= \sigma^2 \frac{\sum (x_i - \bar{x})^2}{\left[\sum (x_i - \bar{x})^2\right]^2} = \frac{\sigma^2}{\sum (x_i - \bar{x})^2}$$



What is the Confidence Interval for the Coefficients Obtained?

- Notice that $\hat{\beta}_1$ and $\hat{\beta}_0$ are both normally distributed when the noise is normally distributed
- We can define confidence intervals that the true β_1 and β_0 are in it with 95% confidence
- For $\sigma^2 = \text{var}(\epsilon)$, we define

$$\text{SE}(\hat{\beta}_0)^2 = \sigma^2 \left(\frac{1}{n} + \frac{\bar{x}^2}{\sum_{i=1}^n (x_i - \bar{x})^2} \right), \quad \text{SE}(\hat{\beta}_1)^2 = \frac{\sigma^2}{\sum_{i=1}^n (x_i - \bar{x})^2}$$

- Hint on derivation: since $\sum_i (x_i - \bar{x})\bar{y} = 0$, we can start with the alternative formulation $\hat{\beta}_1 = \frac{\sum_{i=1}^n (x_i - \bar{x}) y_i}{\sum_{i=1}^n (x_i - \bar{x})^2}$ and use the properties stated at the beginning of slides to derive $var(\hat{\beta}_1)$
- The 95% confidence intervals for β_0 and β_1 are

$$\left[\beta_0 - 2\text{SE}(\hat{\beta}_0), \beta_0 + 2\text{SE}(\hat{\beta}_0) \right], \quad \left[\beta_1 - 2\text{SE}(\hat{\beta}_1), \beta_1 + 2\text{SE}(\hat{\beta}_1) \right]$$

- Exercise: Show $\hat{\beta}_0$ and $\hat{\beta}_1$ are unbiased.

$$E[\hat{\beta}_1] = E\left[\frac{\sum (x_i - \bar{x}) y_i}{\sum (x_i - \bar{x})^2}\right] = E\left[\frac{\sum (x_i - \bar{x})(\beta_0 + \beta_1 x_i + \varepsilon_i)}{\sum (x_i - \bar{x})^2}\right]$$

$$= \frac{\beta_0 \cdot \sum (x_i - \bar{x}) + \beta_1 \sum (x_i - \bar{x}) x_i + 0}{\sum (x_i - \bar{x})^2}$$

$$= \beta_1 \frac{\sum (x_i - \bar{x})^2}{\sum (x_i - \bar{x})^2} = \beta_1$$

Is There a Relationship Between x and y?

- We would like to know if there is really a relationship between x and y or if the fit is useless?
- We form a hypothesis testing for β_1 (if it is zero, then x and y are not related):
  - $H_0 : \beta_1 = 0$
  - $H_1 : \beta_1 \neq 0$ → There are some relationships
- Our test statistic is chosen to be $t = \frac{\hat{\beta}_1 - \beta_1}{SE(\hat{\beta}_1)} = \frac{\hat{\beta}_1 - 0}{SE(\hat{\beta}_1)}$
- We look this up in the t-distribution table and find the p-value

(see the code)

If p-value$\leq \alpha$: reject H_0 (accept H_1)
If p-value$> \alpha$: fail to reject H_0

- For the example provided p-value $= 2 \times 10^{-16}$ and we reject H_0 (meaning that x and y are related)

How Well does the Model Explain the Data?

- We can also answer the question of how well our fitted model explains the data by defining another statistic:

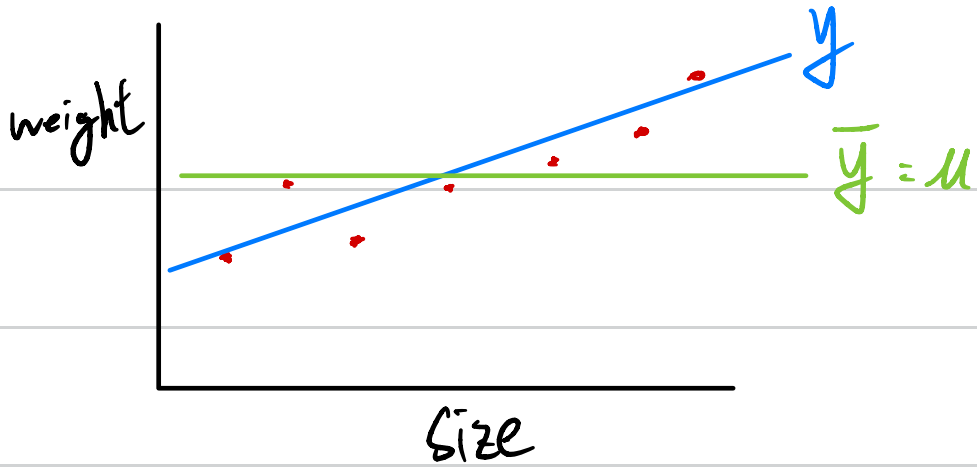
$$R^2 = 1 - \frac{RSS}{TSS} = \frac{\text{explained}}{\sum (y_i - \bar{y})^2}$$

where TSS is the Total Sum of Squares

$$RSS = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_i)^2, \quad TSS = \sum_{i=1}^{n} (y_i - \bar{y})^2 \to n \cdot \sigma^2$$

- For general regression problems (not only the simple one) R^2 statistic measures the proportion of variability in y that can be explained by x
- R^2 close to 1 indicates that our model explains a large proportion of the response variability, and R^2 close to zero indicates that our model cannot explain much of the variability in response

(see the code)



By eye, it looks like the line fits the data better than μ.

How to quantify that difference? R^2

Multiple Linear Regression

Multiple Linear Regression

– It can be the case that we have multiple features $x_1, \ldots, x_p$ and we would like a fit like

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots \beta_p x_p + \epsilon$$

For example: $\text{sales} = \beta_0 + \beta_1.\text{TV} + \beta_2.\text{radio} + \beta_3.\text{newspaper} + \epsilon$

– Suppose that we have n training/response samples $(\boldsymbol{x}^{(1)}, y_1), \ldots, (\boldsymbol{x}^{(n)}, y_n)$ where

$$\boldsymbol{x}^{(1)} = \begin{pmatrix} x_1^{(1)} \\ x_2^{(1)} \\ \vdots \\ x_p^{(1)} \end{pmatrix}, \quad \boldsymbol{x}^{(2)} = \begin{pmatrix} x_1^{(2)} \\ x_2^{(2)} \\ \vdots \\ x_p^{(2)} \end{pmatrix}, \quad \ldots \quad , \boldsymbol{x}^{(n)} = \begin{pmatrix} x_1^{(n)} \\ x_2^{(n)} \\ \vdots \\ x_p^{(n)} \end{pmatrix}$$

Multiple Linear Regression

– We would like to minimize the following squared error

$$RSS = \sum_{i=1}^{n} \left(y_i - \beta_0 - \beta_1 x_1^{(i)} - \beta_2 x_2^{(i)} \ldots - \beta_p x_p^{(i)} \right)^2$$

– Consider using the following matrices/vectors

$$\boldsymbol{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}, \quad \boldsymbol{\beta} = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_p \end{pmatrix}, \quad \boldsymbol{X}_{n \times (p+1)} = \begin{pmatrix} 1 & x_1^{(1)} & x_2^{(1)} & \cdots & x_p^{(1)} \\ 1 & x_1^{(2)} & x_2^{(2)} & \cdots & x_p^{(2)} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 1 & x_1^{(n)} & x_2^{(n)} & \cdots & x_p^{(n)} \end{pmatrix}$$

note that matrix $\boldsymbol{X}$ has the samples along the rows

– Then, it is straightforward to see that

$$RSS = (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})^\top (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}) = \| y - X\beta \|^2$$

$$RSS = (y - X\beta)^T (y - X\beta)$$

$$= (y^T - \beta^T X^T)(y - X\beta)$$

$$= y^T y - 2y^T X\beta + \beta^T X^T X\beta$$

$$\frac{\partial RSS}{\partial \beta} = 0 - 2X^T y + 2X^T X\beta = 0$$

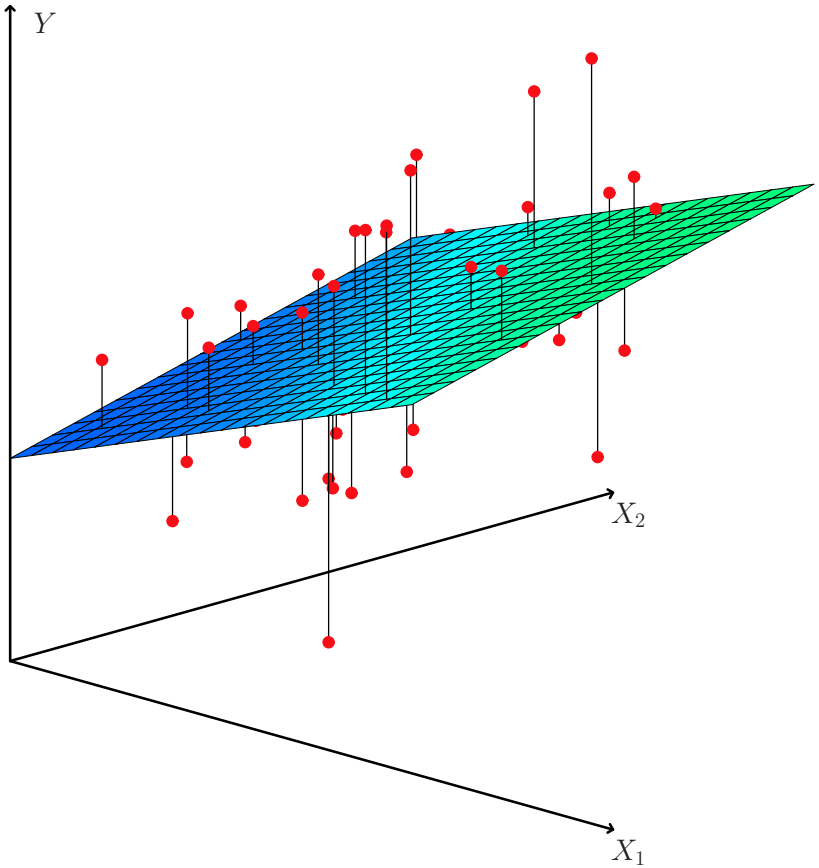
$$\Rightarrow X^T X\beta = X^T y \Rightarrow \beta = (X^T X)^{-1} X^T y$$

Multiple Linear Regression

– Similar to what we did before we can set $\partial RSS/\partial \boldsymbol{\beta} = 0$ (requires little bit of knowing how to do vector/matrix derivatives) and get

$$\hat{\boldsymbol{\beta}} = (\boldsymbol{X}^\top \boldsymbol{X})^{-1} \boldsymbol{X}^\top \boldsymbol{y}$$

minimize $\|y - X\beta\|^2$
β



Are the Features and Response Related?

- We would like to know if at least one of the features $x_1, \ldots, x_p$ is useful in predicting the response.
- We form a hypothesis testing as follows:
  - $H_0 : \beta_1 = \beta_2 = \ldots = 0$
  - H_1 : at least one β_i is non-zero
- For $RSS = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2$ and $TSS = \sum_{i=1}^{n}(y_i - \bar{y})^2$, our test statistic is chosen to be

$$F = \frac{(TSS - RSS)/p}{RSS/(n-p-1)}$$

which turns to have an F distribution

If p-value$\leq \alpha$: reject H_0, If p-value$> \alpha$: fail to reject H_0

- We can either use F-distribution tables and find the p-value; or use this rule: *if F is much larger than 1, we reject H_0; if F is very close to 1, we fail to reject H_0*

(see the code)

F much $> 1 \Rightarrow$ P-value $\downarrow$: Good

F close to $1 \Rightarrow$ P-value $\uparrow$: Bad

Now that we have our fit, again we would like to address few questions about it!

(We code up the answer to each question in the Example Code 2)

Intercept : β_0

If I drop either "indus" or "age"

It's highly possible to improve the other p-value?

Assessing the P-Values and Correlations Among Features

– Sometimes (most of the times) features are correlated and the contribution of one feature can be taken care of by the others

	Coefficient	Std. error	t-statistic	p-value
Intercept	2.939	0.3119	9.42	< 0.0001
TV	0.046	0.0014	32.81	< 0.0001
radio	0.189	0.0086	21.89	< 0.0001
newspaper	−0.001	0.0059	−0.18	0.8599

should be dropped

	TV	radio	newspaper	sales
TV	1.0000	0.0548	0.0567	0.7822
radio		1.0000	0.3541	0.5762
newspaper			1.0000	0.2283
sales				1.0000

(see the code)

What are the Best Selection of Features?

- As noted sometimes some features can be redundant and we would like to find the best subset of features that predicts well and is not redundant
- In general this problem is "NP-hard" (computationally very hard) and we need to assess 2^p models
- There are some heuristics to do this that we will see later:
  - Forward selection: model p regressions each with only one feature, pick the one with least RSS, repeat it with selected feature and combination of others, ...
  - Backward selection: Start with all features and remove variable with largest p-value, run a new regression, remove variable of largest p-value, ...

In general, Backward is better? less time?

How Can We Handle Categorical Features?

- Sometimes our features do not take numerical values, instead they take categorical values
- **Example**: In a regression problem we have a feature called `ethnicity`, which takes possible values of `Asian, Caucasian, African-American`
- We can introduce 2 dummy variables (features) e_A, e_C
  - $e_A = 1,\ e_C = 0$ if Asian
  - $e_A = 0,\ e_C = 1$ if Caucasian
  - $e_A = 0,\ e_C = 0$ if African-American
- Basically, for every categorical feature that has L levels, we need to define $L - 1$ dummy variables

Can We Only Fit Linear Curves with Linear Regression?

Yes, we can!

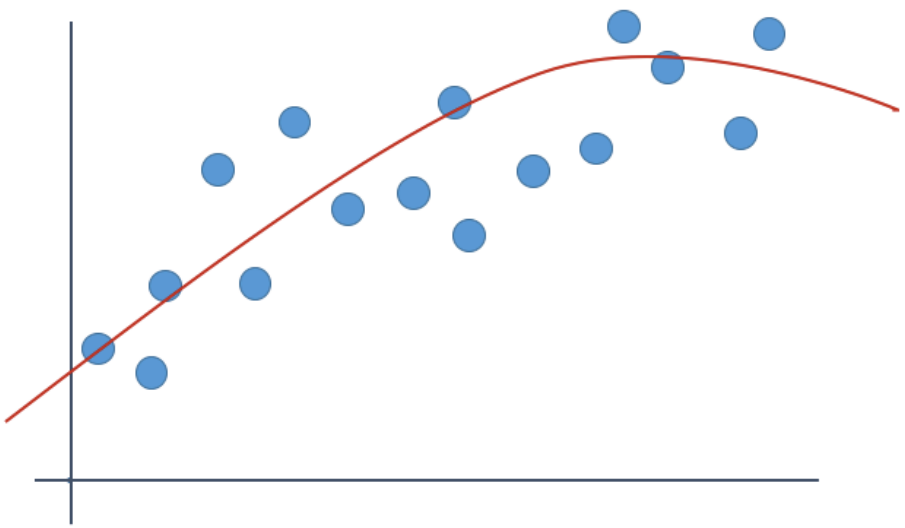
- Based on the equation

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots \beta_p x_p + \epsilon$$

  one might get the impression that linear regression is only good for fitting flat surfaces (linear manifolds)
- If we include powers of a feature, e.g., $x_1, x_1^2, \ldots$ or cross terms between the features, e.g., $x_1 x_2, x_2 x_3 x_5$, etc, then we can also fit nonlinear surfaces
- Of course knowing what powers or what cross terms to include in the feature list is not always clear

Can We Only Fit Flat Curves with Linear Regression?

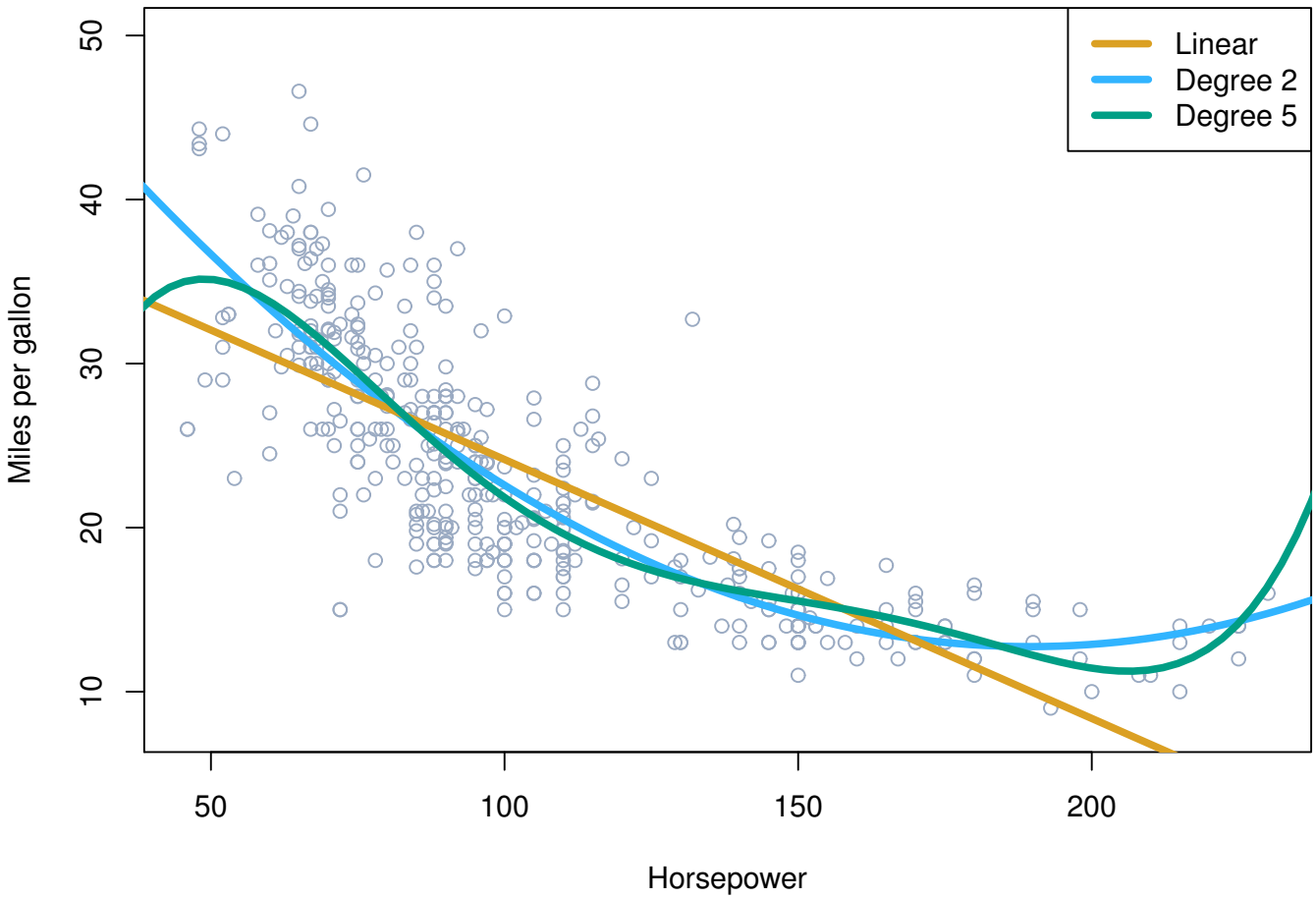
– **Example**: For a problem wtih only one feature, we have a set of point that look lying on a parabola, we use the regression $y = \beta_0 + \beta_1 x + \beta_2 x^2$



Can We Only Fit Flat Curves with Linear Regression?

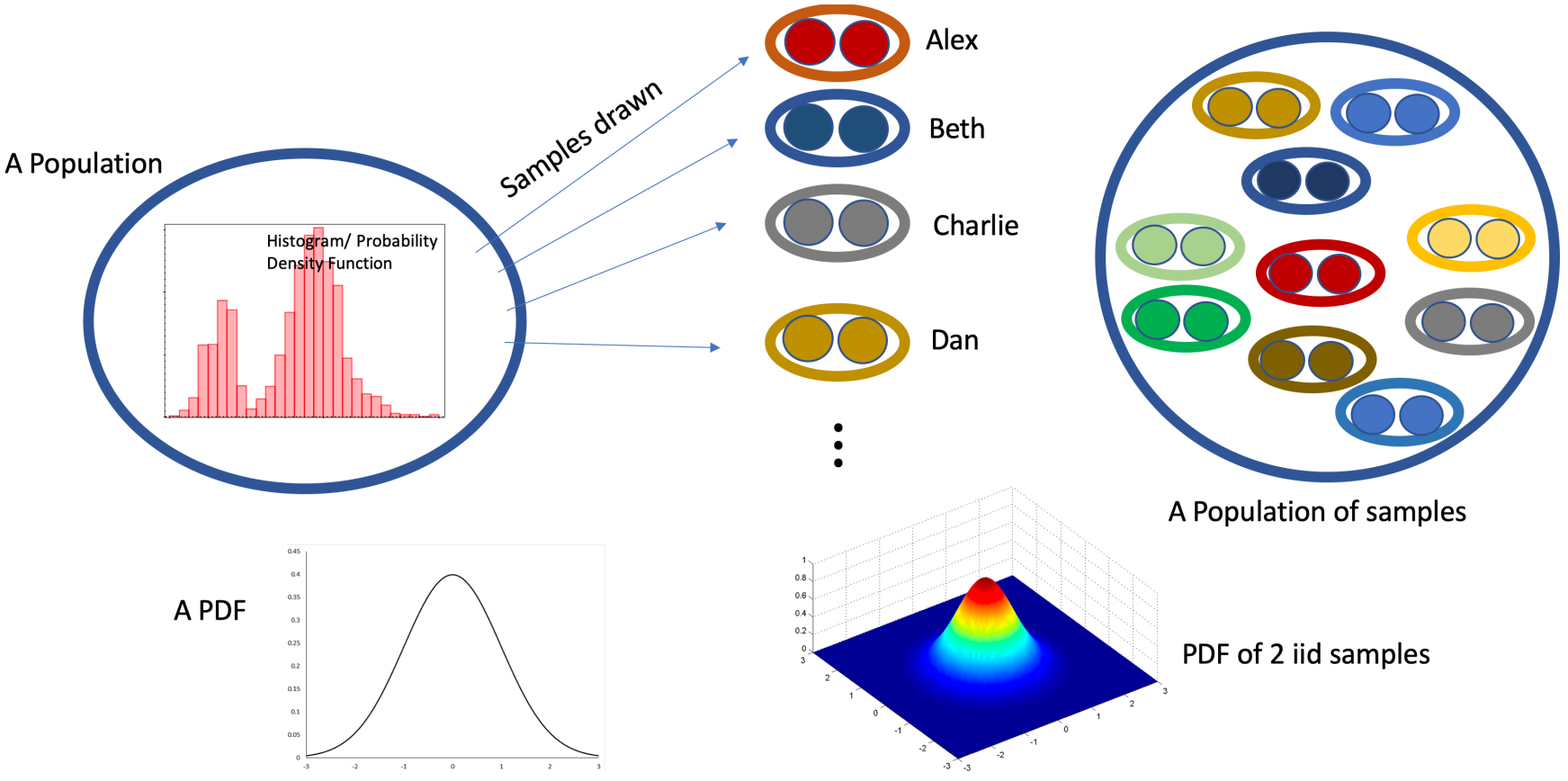
– **Example**: Regressing Mile per Gallon in terms of the Horse Power

$$\texttt{mpg} = \beta_0 + \sum_{i=1}^{p} \beta_i (\texttt{horsepower})^i$$



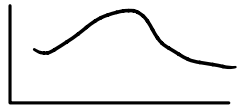
Sampling Distribution

Sampling Distribution



- We often have a reference population (with some PDF or histogram) and would like to draw multiple samples to discover something about the population

$f_X(x)$:



$(x_1^{(1)}, x_2^{(1)})$

$(x_1^{(2)}, x_2^{(2)})$

$(x_1^{(3)}, x_2^{(3)})$

$$f_{X_1,X_2}(x_1,x_2) = f_X(x_1)\, f_X(x_2)$$

$$f_{X_1,X_2,X_3}(x_1,x_2,x_3) = \prod_{i=1}^{3} f_X(x_i)$$

Sampling Distribution

- A useful fact to derive the distribution for multiple independent samples is: *if* $x_1 \sim f_1(x)$, $x_2 \sim f_2(x)$, ..., $x_N \sim f_N(x)$ *are independent random variables, their joint PDF is*

$$f(x_1, x_2, \ldots, x_N) = f_1(x_1) f_2(x_2) \ldots f_N(x_N) = \prod_{i=1}^{N} f_i(x_i)$$

- **Example:** We take two independent samples x_1 and x_2 from a standard normal distribution. What is the joint PDF of x_1 and x_2?

$$\begin{aligned} f(x_1, x_2) &= \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x_1^2}{2}\right) \times \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x_2^2}{2}\right) \\ &= \frac{1}{2\pi} \exp\left(-\frac{x_1^2 + x_2^2}{2}\right) \end{aligned}$$

Given $f_X(x) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2}\right)$

$$\rightarrow f_{X_1,X_2}(x_1, x_2) = \frac{1}{\sqrt{2\pi}} \exp\left(\frac{-x_1^2}{2}\right) \cdot \frac{1}{\sqrt{2\pi}} \exp\left(\frac{-x_2^2}{2}\right)$$

$$= \frac{1}{2\pi} \exp\left(-\frac{x_1+x_2}{2}\right)$$

$$f(x_1, \cdots, x_{10}) = \prod_{i=1}^{10} \frac{1}{\sqrt{2\pi}\,\sigma} \exp\left(-\frac{(x_i-N)^2}{2\sigma^2}\right)$$

$$= \left(\frac{1}{\sqrt{2\pi}\,\sigma}\right)^{10} \exp\left(-\frac{\Sigma(x_i-N)^2}{2\sigma^2}\right)$$

In-Class Exercise

- We take 10 samples from a normal distribution with mean μ and variance σ^2. What is the sampling distribution?

$$f(x_1, x_2, \ldots, x_N) = \ldots$$

Functions of Samples

- Not only we can talk about the joint distribution of the samples, we can also talk about the distribution of a function applied to the samples.
- Finding the distribution for functions of random variables is not a straightforward task.
- **Example**: If x_1 and x_2 have the joint PDF $f_{X,Y}(x, y)$, what is the pdf for $z = 3x + y$?

  Answer. $F_Z(z) = \mathbb{P}(3X + Y \leq z) = \int\int_{3x+y \leq z} f_{X,Y}(x, y) dx\ dy$ and then taking a derivative of the CDF $F_Z(z)$ to acquire the pdf $f_Z(z)$.
- But sometimes there are shortcuts. For example when we know what is the distribution for the sum of two random variables, and all we need to do is estimating the distribution parameters.

If X, Y are normal → Z is normal

In-Class Exercise

- **Example:** We have a normal distribution with mean 2 and variance 9. What is the distribution of of the sample mean acquired by averaging 4 independently drawn samples.
  Hard Way. We obtain the sampling distribution $f(x_1, x_2, x_3, x_4)$, which is the joint distribution, and then use the technique in the previous example to acquire the distribution of $z = (x_1 + x_2 + x_3 + x_4)/4$.
  Easy Way. Using the fact sheet from lecture 2, we know weighted sum of normal random variables is normally distributed, so all we need is to find the mean and the variance of $z = (x_1 + x_2 + x_3 + x_4)/4$.

$$\mathbb{E}(z) = 4 \times (2/4) = 2, \quad \text{var}(z) = 4 \times \frac{9}{16} = \frac{9}{4}.$$

  Therefore $z \sim \mathcal{N}(2, \frac{9}{4})$.
- See `https://en.wikipedia.org/wiki/Relationships_among_probability_distributions` for other shortcuts for the sum of independent random variables.

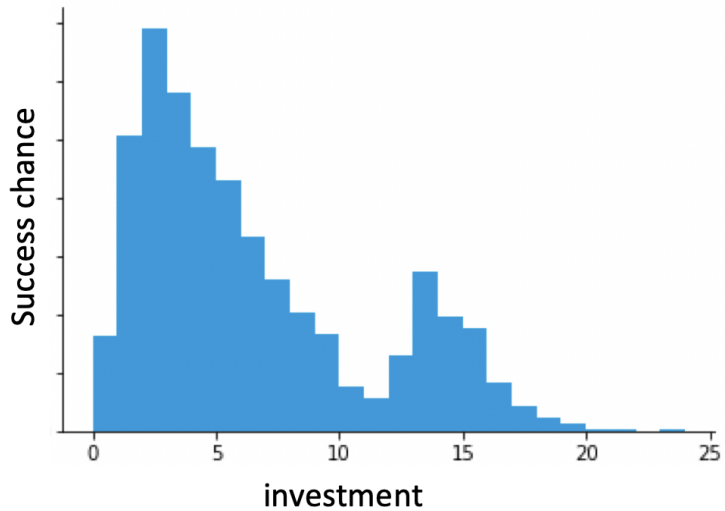
$$X_i \sim N(2, 9)$$

$$Z = \frac{X_1 + X_2 + X_3 + X_4}{4} \longrightarrow Z \sim N\left(\frac{2 \times 4}{4}, \frac{9 \times 4}{16}\right)$$

$$\Rightarrow Z \sim N\left(2, \frac{9}{4}\right)$$

Brief Overview of Maximum Likelihood

– Say you are taking a risky investment and you know the following pdf corresponds to the succcess chance in terms of the investment. How much would you invest?



$\tilde{\chi}_1 = 1.213$

$\tilde{\chi}_2 = -10.7$

$\tilde{\chi}_3 = 3.41$

$\vdots$

, so observation $= \begin{bmatrix} 1.213 \\ -10.7 \\ 3.41 \\ \vdots \end{bmatrix}$

Brief Overview of Maximum Likelihood

- Maximum likelihood (ML) is a statistical estimation technique
- The main goal in ML is often estimating the parameters of the **reference** population from a set of samples
- Let $x_1, x_2, \cdots x_n$ be samples from a distribution with some unknown parameter θ and joint distribution

$$f(x_1, x_2, \cdots, x_n|\theta)$$

- The maximum likelihood estimate of θ based on the observations $\tilde{x}_1, \tilde{x}_2, \cdots \tilde{x}_n$ is

$$\theta_{ML} = \underset{\theta}{\text{argmax}} \; f(\tilde{x}_1, \tilde{x}_2, \cdots, \tilde{x}_n|\theta)$$

- When $x_1, x_2, \cdots x_n$ are i.i.d samples from a distribution $f(\cdot)$, then

$$f(x_1, x_2, \cdots, x_n|\theta) = f(x_1|\theta)f(x_2|\theta)\cdots f(x_n|\theta)$$

$$= \prod_{i=1}^{n} f(x_i|\theta)$$

Brief Overview of Maximum Likelihood

$E[X] = \mu$, $\sigma_X^2 = 1$

Example 1. We have a normal distribution $\mathcal{N}(\mu, 1)$ and we do not know μ. We take 5 independent samples from this distribution and the values turn out to be

$$\tilde{x}_1 = 2.5377,\ \tilde{x}_2 = 3.8339,\ \tilde{x}_3 = -0.2588,\ \tilde{x}_4 = 2.8622,\ \tilde{x}_5 = 2.3188,$$

what is the ML estimate of μ.

Solution. If we take 5 independent samples $x_1, x_2, \cdots x_5$ from a normal distribution $\mathcal{N}(\mu, 1)$, their joint distribution is

$$f(\tilde{x}_1, \tilde{x}_2, \tilde{x}_3, \tilde{x}_4, \tilde{x}_5|\mu) = \prod_{i=1}^{5} \frac{1}{\sqrt{2\pi}} \exp(-\frac{(x_i - \mu)^2}{2}),$$

some basic calculus yields $\mu_{ML} = \frac{\tilde{x}_1 + \tilde{x}_2 + \cdots + \tilde{x}_5}{5} = 2.2587$ (why?)

$$N(\mu, 1) \rightarrow f_X(x) = \frac{1}{\sqrt{2\pi}} \exp\left(\frac{-(x-\mu)^2}{2}\right)$$

1st step: $f(x_1, \cdots, x_5) = \prod_{i=1}^{5} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{\sum_{i=1}^{5}(x_i-\mu)^2}{2}\right)$

$$f(\tilde{x}_1, \tilde{x}_2, \cdots, \tilde{x}_5 | \mu) = \left(\frac{1}{\sqrt{2\pi}}\right)^5 \exp\left(-\frac{\sum(x_i-\mu)^2}{2}\right)$$

$$= \left(\frac{1}{\sqrt{2\pi}}\right)^5 \exp\left(-\frac{(2.5377-\mu)^2}{2} - \frac{(3.8339-\mu)^2}{2} - \cdots\right)$$

μ is the only unknown,

$$l(\mu) = \log f(\tilde{x}_1, \cdots, \tilde{x}_5 | \mu) = 5 \cdot \log \frac{1}{\sqrt{2\pi}} - \frac{\sum(x_i-\mu)^2}{2}$$

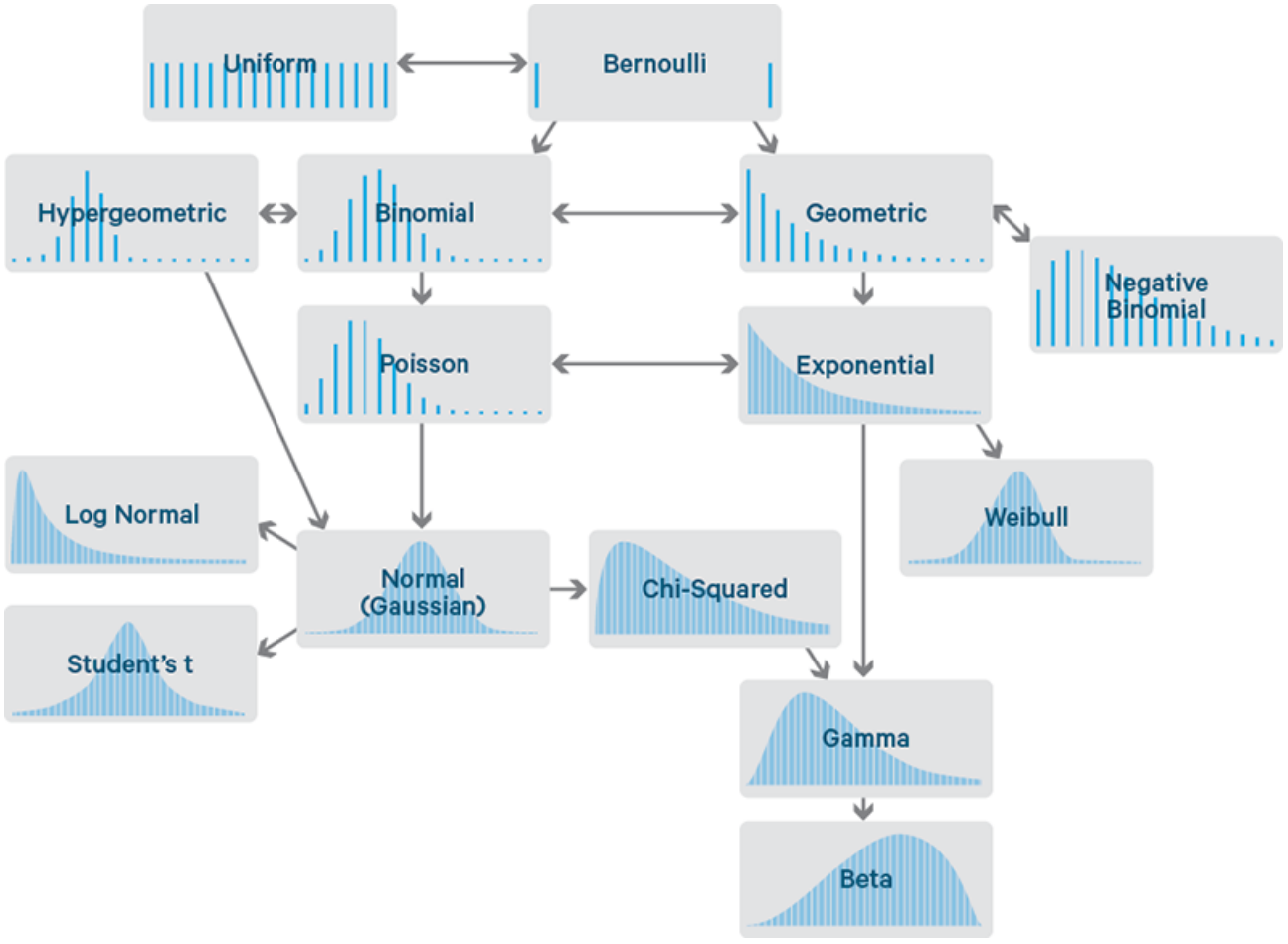
To find max of μ, take partial derivative

$$\frac{\partial l(\mu)}{\partial \mu} = 0 \rightarrow \sum_{i=1}^{5}(\mu - x_i) = 0$$

$$\Rightarrow 5\mu = \sum_{i=1}^{5} x_i \Rightarrow \mu = \frac{\sum_{i=1}^{5} x_i}{5}$$

Brief Overview of Maximum Likelihood

Example 2. We have a random generator which works as a black-box. Use the table below and a maximum likelihood approach to estimate the distribution and its parameters (see the MATLAB code).



[Figure Link]

Example 2:

$$X \sim \exp(\lambda), \quad \text{PDF: } f_X(x) = \lambda \cdot e^{-\lambda x} \quad (\lambda \geq 0,\ x \geq 0)$$

$$f(x_1, \cdots x_{1000} | \lambda) = \lambda^{1000} \cdot \exp\left(-\lambda \sum_{i=1}^{1000} x_i\right)$$

$$\mathcal{L}(\lambda) = \log f(\tilde{x}_1, \cdots \tilde{x}_{1000} | \lambda) = 1000 \log \lambda - \lambda \sum_{i=1}^{1000} \tilde{x}_i$$

$$\frac{\partial \mathcal{L}(\lambda)}{\partial \lambda} = \frac{1000}{\lambda} - \sum_{i=1}^{1000} \tilde{x}_i = 0 \Rightarrow \lambda = \frac{1000}{\sum_{i=1}^{1000} \tilde{x}_i}$$

Brief Overview of Maximum Likelihood

More related to Linear Regression

Example 3. We have a simple linear model in the form of $y = \beta_0 + \beta_1 x + \epsilon$, where $\epsilon \sim \mathcal{N}(0, 1)$. We pass the points $x_1, \ldots, x_n$ to the model and receive the independent random responses $y_1, \ldots, y_n$. Based on the observed samples, what is the ML estimate for β_0 and β_1?

Hint:

$$\begin{aligned} &\arg\max_{\beta_0, \beta_1} \quad f(y_1, \cdots y_n | X_1 = x_1, \cdots, X_n = x_n, \beta_0, \beta_1) \\ &= \arg\max_{\beta_0, \beta_1} \quad \log\left(f(y_1, \cdots y_n | X_1 = x_1, \cdots, X_n = x_n, \beta_0, \beta_1)\right) \end{aligned}$$

Important Note. Similar to the way we treated the RSS minimization, in ML we also need to set the derivative with respect to all the variables to zero. You will do this in the homework.

$$y_i = \beta_0 + \beta_1 x_i + \varepsilon_i \quad \varepsilon_i \sim N(0,1)$$

random

$$\rightarrow \quad y_i \sim N(\beta_0 + \beta_1 x_i, 1)$$

$$f(y_1, y_2 \cdots y_n \mid \beta_0, \beta_1) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{\sum (y_i - \beta_0 - \beta_1 x_i)^2}{2}\right)$$

$$L(\beta_0, \beta_1) = \log f(y_1, \cdots y_n \mid \beta_0, \beta_1)$$

$$= n \log \frac{1}{\sqrt{2\pi}} - \frac{\sum (\hat{y}_i - \beta_0 - \beta_1 x_i)^2}{2}$$

constant

instead of take derivative of whole equation

minimize $\frac{1}{2} \sum (\hat{y}_i - \beta_0 - \beta_1 x_i)^2$

RSS

Brief Intro to Optimization

Setting the Derivative to Zero?

- We have been setting the derivative both for minimization and maximization. How do we know if the resulting point is a minimizer or a maximizer?
- What happens if we cannot solve the equation resulted by setting the derivative to zero?
- Normally distinguishing a minimizer from a maximizer requires the second derivative information
- In this lecture we will introduce methods that are for general minimization
- Also would cover a class of functions that a stationary point corresponds to a minimizer

new methods !

Gradient Descent and I ts Variants

In-Class Exercise

- Find the minimizer of the function

$$\mathcal{C}(p_1, p_2) = (p_1 - p_2 - 3)^2 + p_2^2$$

Can We Always Find the Zero Easily?

- How about the minimizer of the function

$$\mathcal{C}(p_1, p_2) = (1 - p_1)^2 + (1 - p_2)^2 - 2\exp(-3p_1^2 - 3p_2^2)$$

Let's take a look at the function plot in Matlab.

Gradient Descent for Minimization

- We saw that our fitting and ML problems ultimately can be reduced to a minimization

$$\min_{\boldsymbol{p}} \ \mathcal{C}(\boldsymbol{p})$$

  where $\boldsymbol{p}$ includes all the unknown variables.
- Assuming $\boldsymbol{p} \in \mathbb{R}^L$, a numerical way of minimization is to start from a point $\boldsymbol{p}^{(0)}$ and iteratively perform the following steps

$$\boldsymbol{p}^{k+1} = \boldsymbol{p}^{(k)} - \eta \nabla \mathcal{C}\Big|_{\boldsymbol{p}=\boldsymbol{p}^{(k)}} \quad \text{where} \quad \nabla \mathcal{C} = \begin{pmatrix} \partial \mathcal{C}/\partial p_1 \\ \partial \mathcal{C}/\partial p_2 \\ \vdots \\ \partial \mathcal{C}/\partial p_L \end{pmatrix}$$

  parameter η is called the **learning rate**
- Larger learning rate does not necessarily mean faster solve
- Let's go through a simple example to see how gradient descent works (see the MATLAB code and the next slide)

$$p = \begin{pmatrix} p_1 \\ p_2 \end{pmatrix} = \begin{bmatrix} 1.2 \\ 3.4 \end{bmatrix} \qquad \nabla C(p) = \begin{bmatrix} 3.1 \\ 4.2 \end{bmatrix}$$

$\eta = 0.01$

$$p^{(2)} = \begin{bmatrix} 1.2 \\ 3.4 \end{bmatrix} - 0.01 \begin{bmatrix} 3.1 \\ 4.2 \end{bmatrix}$$

Gradient Descent for Minimization

- Please refer to the MATLAB `gradientDescent.m` script
- Lets consider the very simple objective

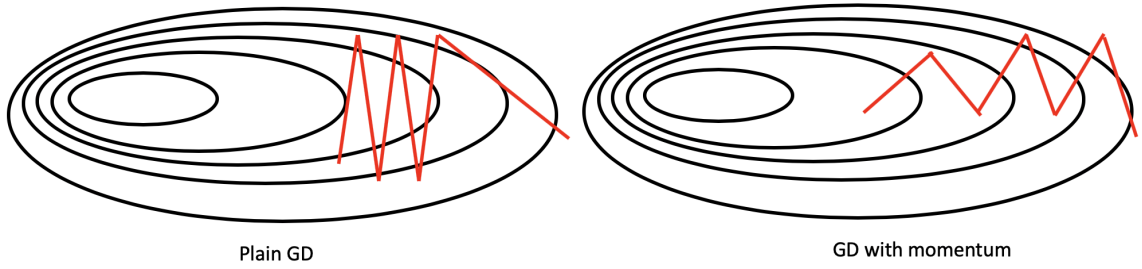
$$\mathcal{C}(p_1, p_2) = (1 - p_1)^2 + (1 - p_2)^2 - 2\exp(-3p_1^2 - 3p_2^2)$$

The gradient can be calculated as

$$\nabla\mathcal{C} = \begin{pmatrix} 2(p_1 - 1) + 12p_1\exp(-3p_1^2 - 3p_2^2) \\ 2(p_2 - 1) + 12p_2\exp(-3p_1^2 - 3p_2^2) \end{pmatrix}$$

- We can see that this objective has multiple local minimizers (two)
- Depending on where we start from we may land in either one
- A too small LR (learning rate) can make the minimization slow
- A too large LR can also make it slow or never converging!
- LR can affect which minimizer we converge to, but this is beyond our control

Gradient Descent with Momentum Makes GD faster!



- Momentum is a method that can dampen the gradient descent oscillations and accelerate it
- It can even help skipping shallow minima and land into deeper minima
- Gradient descent with **learning rate** η and **momentum** γ:

$$\boldsymbol{\theta}_{k+1} = \gamma \boldsymbol{\theta}_k + \eta \nabla \mathcal{C}(\boldsymbol{p}^k)$$
$$\boldsymbol{p}^{k+1} = \boldsymbol{p}^k - \boldsymbol{\theta}_{k+1}$$

- Again refer to the MATLAB code

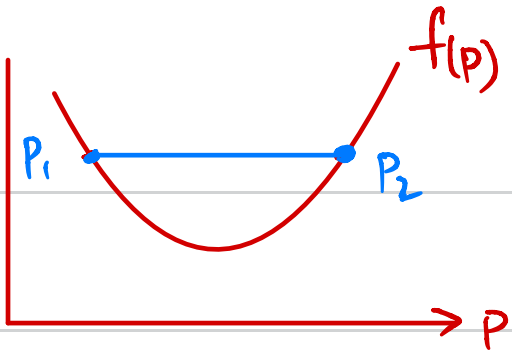
Convex Functions

What Are Convex Functions?

- Convex functions are a class of functions which have computationally attractive properties when it comes to minimization problems
- Suppose that $f(\boldsymbol{p}) : \mathbb{R}^n \to \mathbb{R}$, i.e., a function that operates on multiple variables (a vector) and produces a scalar as the output
- Function f is convex if for all $\boldsymbol{p}_1$ and $\boldsymbol{p}_2$ in the domain, and $0 \leq \theta \leq 1$:

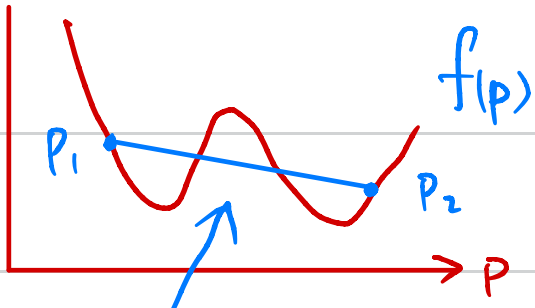
$$f(\theta \boldsymbol{p}_1 + (1-\theta)\boldsymbol{p}_2) \leq \theta f(\boldsymbol{p}_1) + (1-\theta) f(\boldsymbol{p}_2)$$

- Intuitively, a function is convex if the line segment between any two points on the graph of the function lies above (or just touching) the graph between the two points.



If $f(p)$ between $P_1 P_2$ is below line P_1P_2 or equal

→ Convex function



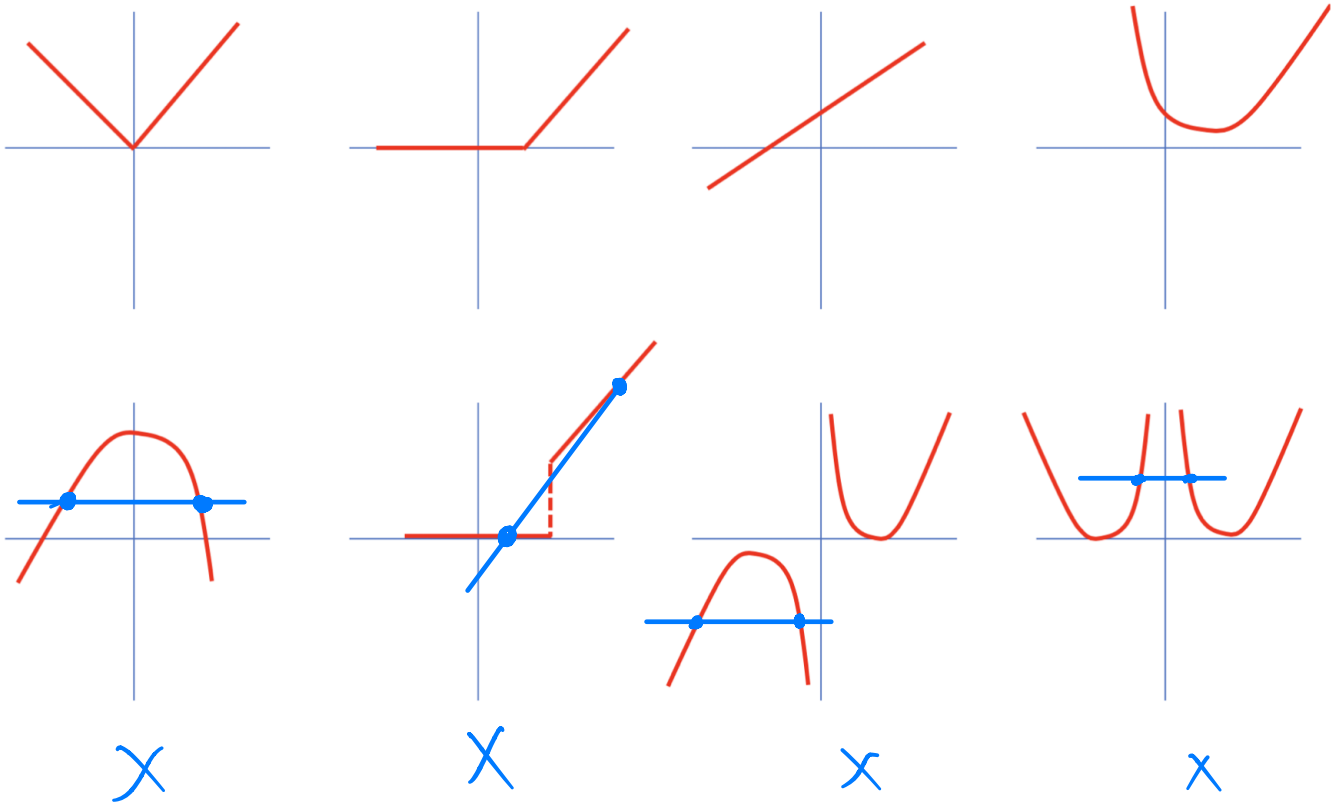
beyound the line P_1P_2

→ Not Convex Function

In-Class Exercise

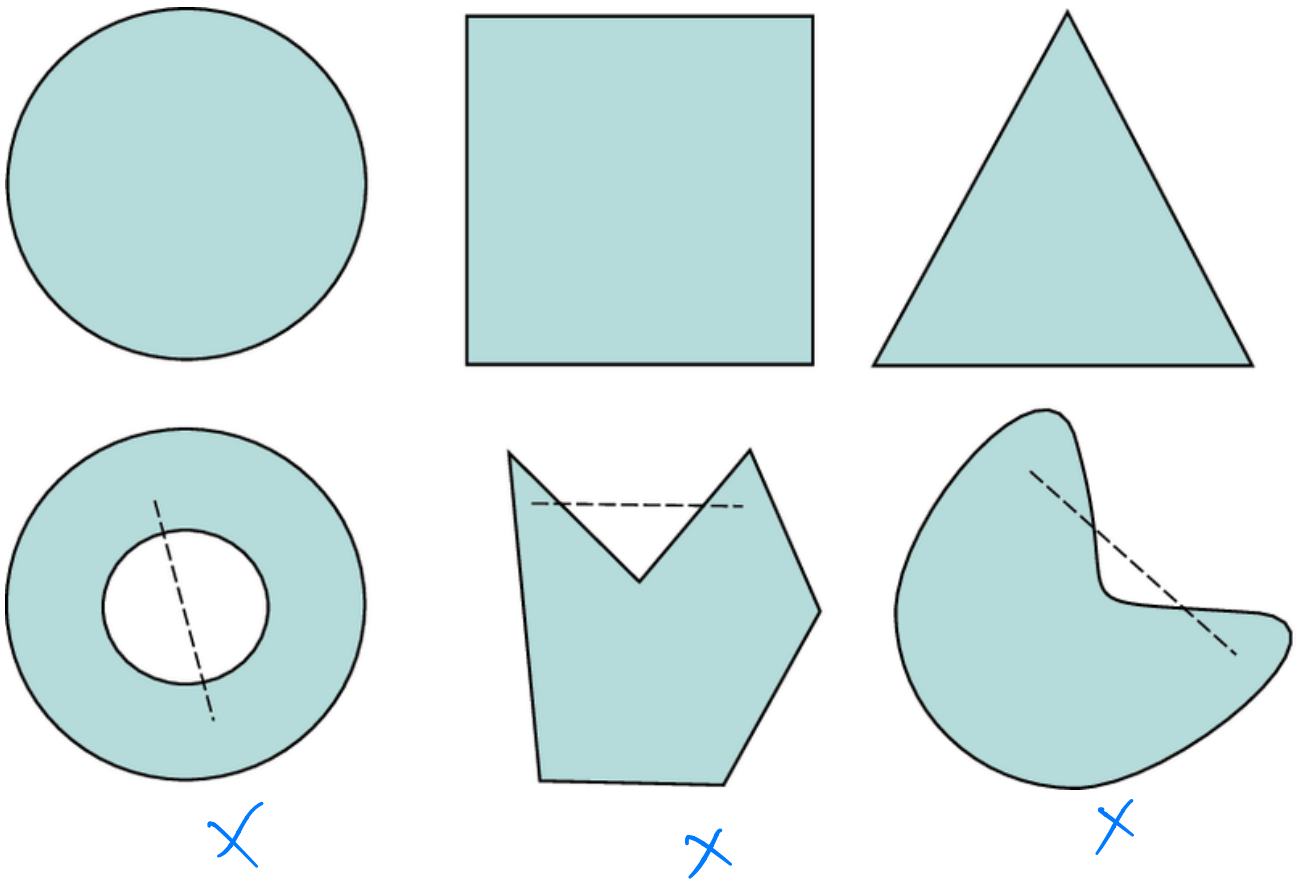
- Identify convex functions:

Any linear function is convex function



Convex Sets

- A set is convex if the line connecting any two points of the set entirely stays within the set



is Convex

Convex Function is not necessarily Continuous Function

If constructing a Convex Function,

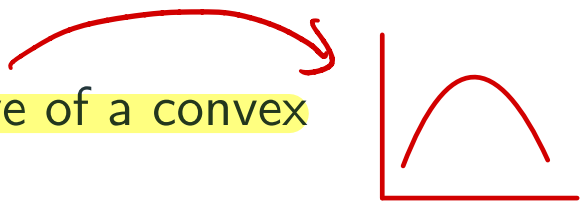
It should be Continuous

More on Convexity

- In optimization theory, convex programs which are in the form

$$\underset{\boldsymbol{p}}{\text{minimize}} \quad f(\boldsymbol{p}) \qquad \text{subject to:} \qquad \boldsymbol{p} \in \text{a convex set}$$

  have very desirable computational properties
- For differentiable convex functions gradient descent always lands to a **global minimizer**
- A function that can be represented as the negative of a convex function is called **concave** function.



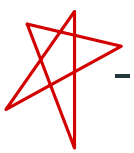
- Verifying the convexity in low dimensions, like 1 or 2, can be done visually. But for high dimensions we need to use the properties and definitions to show the convexity

More Properties and Examples

- One way to show the convexity is using the definition and showing that for all $\boldsymbol{p}_1$ and $\boldsymbol{p}_2$ in the domain, and $0 \leq \theta \leq 1$:

$$f(\theta \boldsymbol{p}_1 + (1-\theta)\boldsymbol{p}_2) \leq \theta f(\boldsymbol{p}_1) + (1-\theta) f(\boldsymbol{p}_2).$$

- **Example.** Show that the function $f(x, y) = x + y$ is convex.



- Generally, all linear functions of the form

$$f(x_1, x_2, \ldots, x_n) = \alpha_1 x_1 + \alpha_2 x_2 + \ldots + \alpha_n x_n$$

are convex.

More Properties and Examples

- Another way to show the convexity is using the following theorem: *A differentiable function is convex if for all $\boldsymbol{p}_1$ and $\boldsymbol{p}_2$ in the domain:*

$$f(\boldsymbol{p}_2) - f(\boldsymbol{p}_1) - \nabla f(\boldsymbol{p}_1)^\top (\boldsymbol{p}_2 - \boldsymbol{p}_1) \geq 0.$$

(we got rid of θ)!

- **Example.** Show that the function $f(x, y) = (x + y)^2$ is convex.

- If $g(z)$ is convex, then g applied to a linear function is also convex, i.e., $g(\alpha_1 x_1 + \ldots + \alpha_n x_n)$ is convex.
- In other words, to show the convexity of $f(x, y) = (x + y)^2$ we only need to show that $f(z) = z^2$ is convex.

$$f\left(\begin{bmatrix} x \\ y \end{bmatrix}\right) = (x+y)^2 = f(x,y)$$

$$\nabla f(x,y) = \begin{bmatrix} 2(x+y) \\ 2(x+y) \end{bmatrix}$$

$$P_i = \begin{bmatrix} x_i \\ y_i \end{bmatrix}, \quad i = 1, 2$$

$$\nabla f(P_1) = 2\begin{bmatrix} x_1+y_1 \\ x_1+y_1 \end{bmatrix}$$

We need to show that

$$P_2 - P_1 = \begin{bmatrix} x_2-x_1 \\ y_2-y_1 \end{bmatrix}$$

$$\nabla f(P_1)^T$$

$$(x_2+y_2)^2 - (x_1+y_1)^2 - 2(x_1+y_1)(x_2-x_1) - 2(x_1+y_1)(y_2-y_1) \geq 0$$

$$\Rightarrow (x_2+y_2)^2 - (x_1+y_1)^2 - 2(x_1+y_1)\left(x_2+y_2-x_1-y_1\right) \geq 0$$

$$\begin{cases} u = x_2+y_2 \\ v = x_1+y_1 \end{cases}$$

$$\Rightarrow u^2 - v^2 - 2v(u-v) \geq 0$$

$$\Rightarrow u^2 + v^2 - 2uv \geq 0 \Rightarrow (u-v)^2 \geq 0 \quad \#$$

Suppose $f(z)$ is convex is $z \in \mathbb{R}$

$\rightarrow f(\alpha_1 x_1 + \alpha_2 x_2 + \cdots + \alpha_p x_p)$ is convex in $x_1, x_2, \cdots, x_p$

$f(z) = z^2$ is convex $\rightarrow$ $f(x, y)$ is convex if $f(x, y)$ is linear.

$f(z)$ is convex is $z \in \mathbb{R}$ if $f''(z) \geq 0 \quad \forall z \in \mathbb{R}$

$\Downarrow$

ex2. $f(z) = z^2$ is convex, since

$f'(z) = 2z$

$f''(z) = 2 \geq 0 \Rightarrow f(z)$ is convex

More Properties and Examples

- Yet, another way to show the convexity is using the following theorem: *A twice differentiable function is convex if at any point in the domain all the eigenvalues of the Hessian matrix are non-negative.* For $f(x_1, x_2, \ldots, x_n)$ the Hessian matrix is defined as

$$\nabla^2 f = \begin{pmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{pmatrix}$$

- In the single variable case, a twice differentiable function $f(x)$ is convex if $f''(x)$ is non-negative for all the points in the domain.
- **Example.** Use this result to show that the function $f(x, y) = (x + y)^2$ is convex.

Prove $f(x,y) = (x+y)^2$ is convex

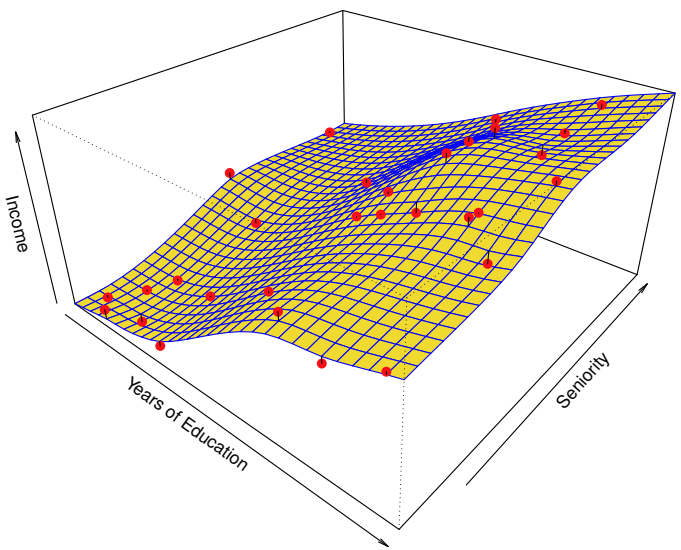
$$\nabla f = \begin{bmatrix} 2(x+y) \\ 2(x+y) \end{bmatrix}$$

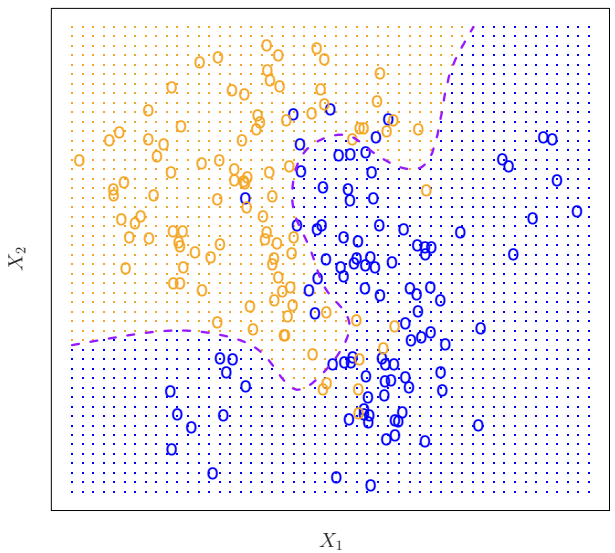
$$\nabla^2 f = \begin{bmatrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x \partial y} \\ \frac{\partial f}{\partial y \partial x} & \frac{\partial^2 f}{(\partial y)^2} \end{bmatrix} = \begin{bmatrix} 2 & 2 \\ 2 & 2 \end{bmatrix} \geq 0 \quad \#$$

Classification

Classification

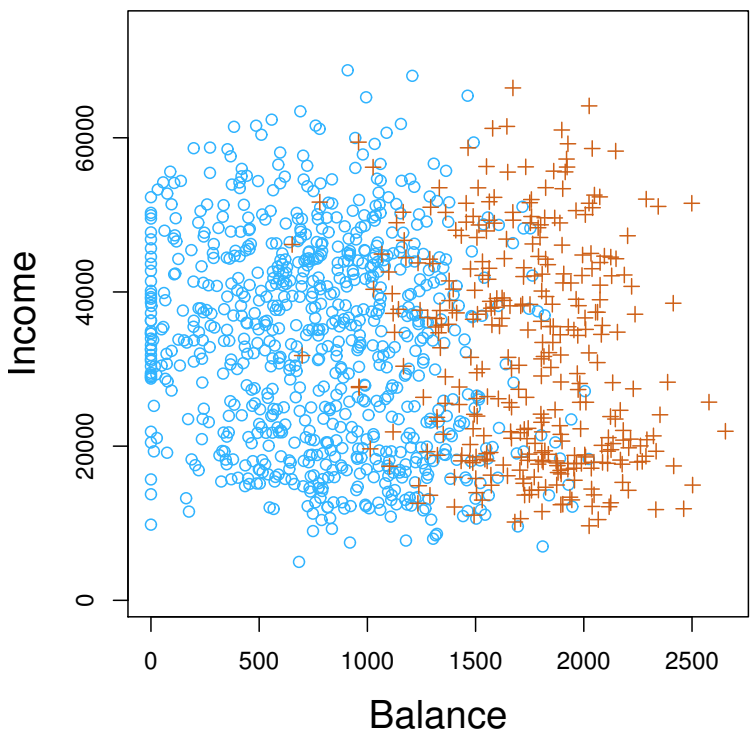
- In many applications, the response is not a quantitative value and instead represents a class, e.g., $y \in \{\text{student, non-student}\}$, $y \in \{\text{while, yellow, green}\}$
- Yet based on the observation of some features, we would like to predict the class (what we refer to as the classification)
- Regression vs classification

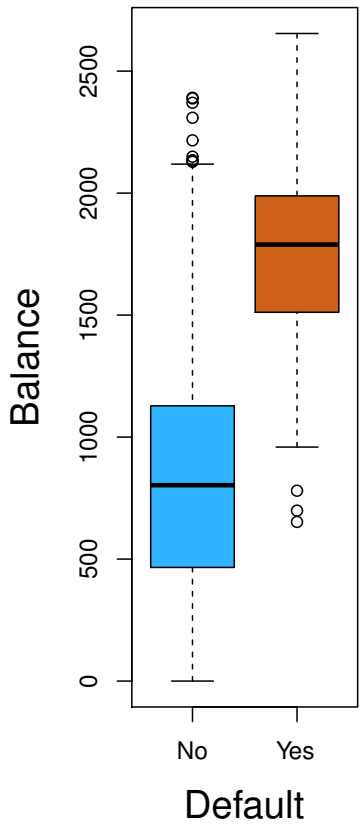


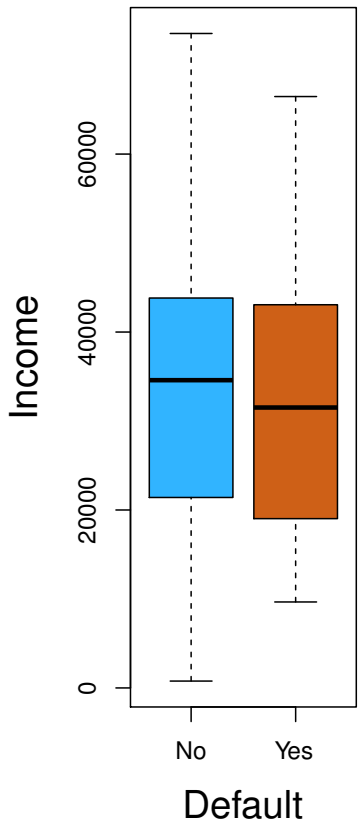


Classification

Example. Predicting default cases on the credit card (unable to pay the credit card), based on the income and current balance







(one immediate observation is probably balance is a more useful feature)

Binary Classification

- In simple regression for a single feature x we fitted a line $y = \beta_0 + \beta_1 x$ to the data
- In binary classification with only one feature, we don't have values any more, but two classes (say class 0 and class 1)
- Can we do the fit in a way that the sign of $\beta_0 + \beta_1 x$ becomes an indicator of the class for us?
- In other words, for a given feature x_t, we make a decision based on the following:

$$y_t = \begin{cases} 1 & \beta_0 + \beta_1 x_t > 0 \\ 0 & \beta_0 + \beta_1 x_t < 0 \end{cases},$$

- A smooth function (called Sigmoid – also inverse Logit) that takes almost binary values $0, 1$ based on the sign of the input z is

Sigmoid for continuous

$$\frac{e^z}{1+e^z} \approx \begin{cases} 1 & z >> 0 \\ 0 & z << 0 \end{cases}$$

Simple Example

input (feature) : $X \in \mathbb{R}$

output (response) : $y \in \{0, 1\}$

$$y = \text{Sign}(\beta_0 + \beta_1 X)$$

problem :

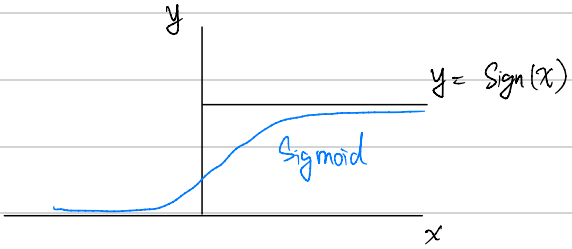
Sign Function is not continuous

Regression : $y = \beta_0 + \beta_1 X$

$(x_1, y_1) \dots (x_n, y_n)$ fit β_0, β_1

such that $y_i \simeq \text{Sign}(\beta_0 + \beta_1 x_i) \quad , i = 1, \dots, n$

minimize $\sum_{i=1}^{n} (y_i - \text{Sign}(\beta_0 + \beta_1 x_i))^2$



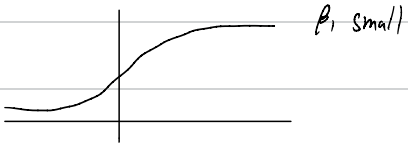
Sigmoid $\begin{cases} \text{Sign}(\beta_0 + \beta_1 x) \\ \sigma(\beta_0 + \beta_1 x) \end{cases}$

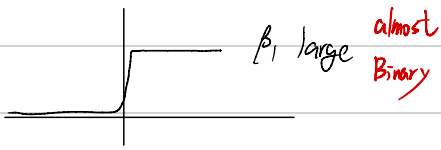
$$P(y=1|x) = \sigma(\beta_0 + \beta_1 x)$$

$$\rightarrow P(y=0|x) = 1 - \sigma(\beta_0 + \beta_1 x)$$

β_0 : offset

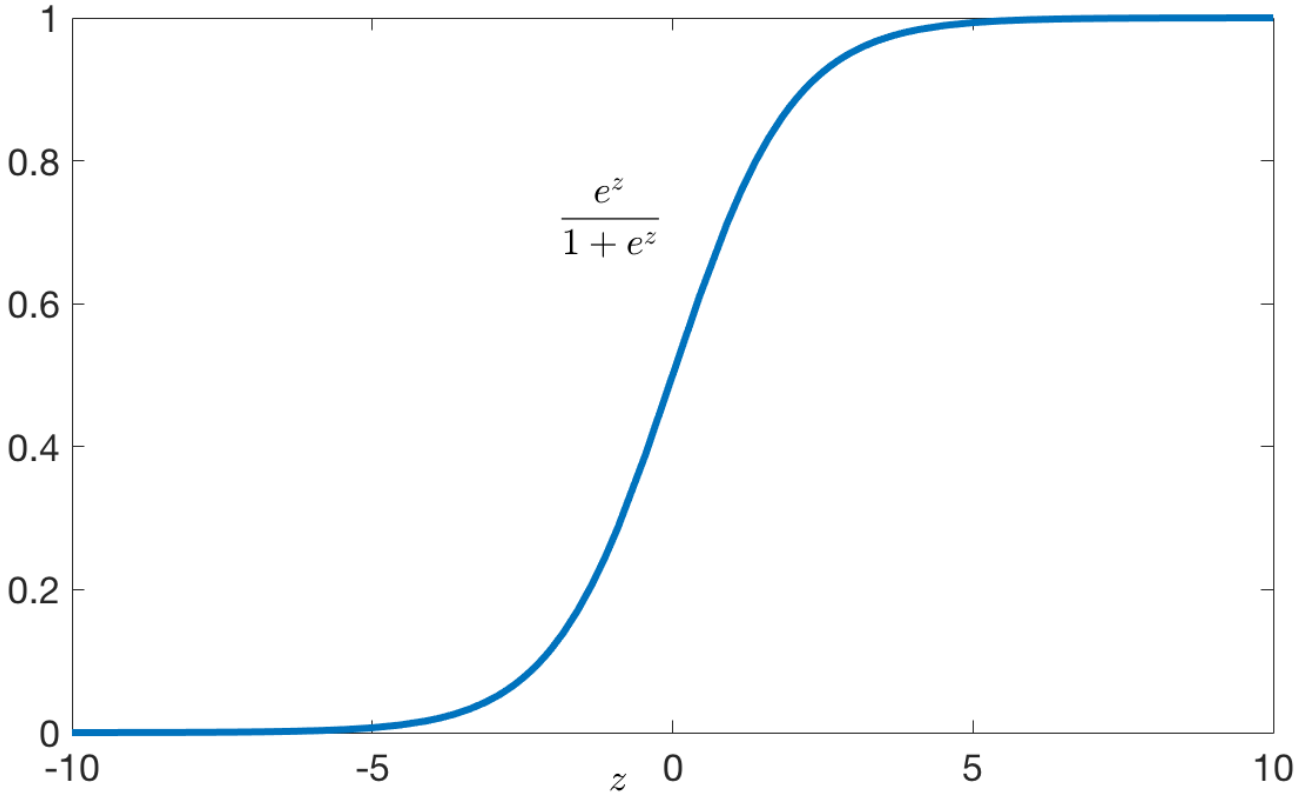
β_1 : slope



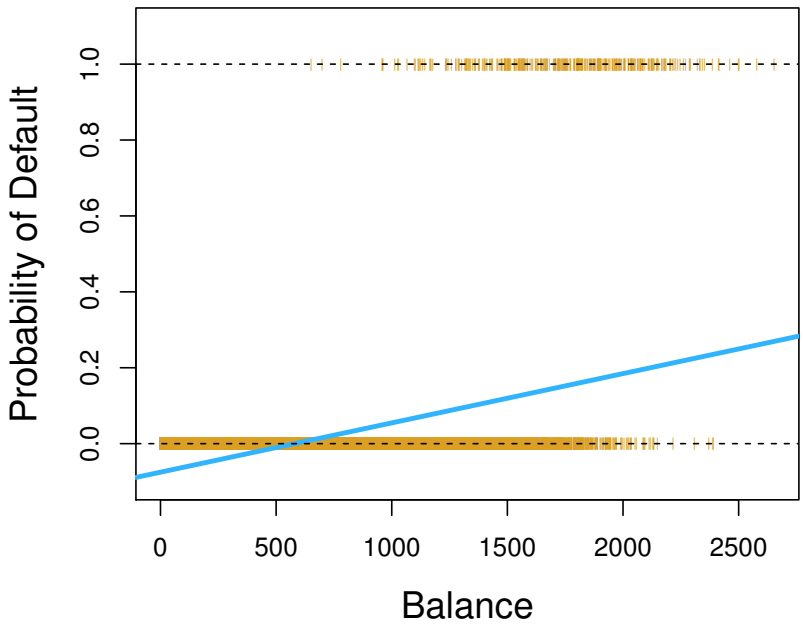


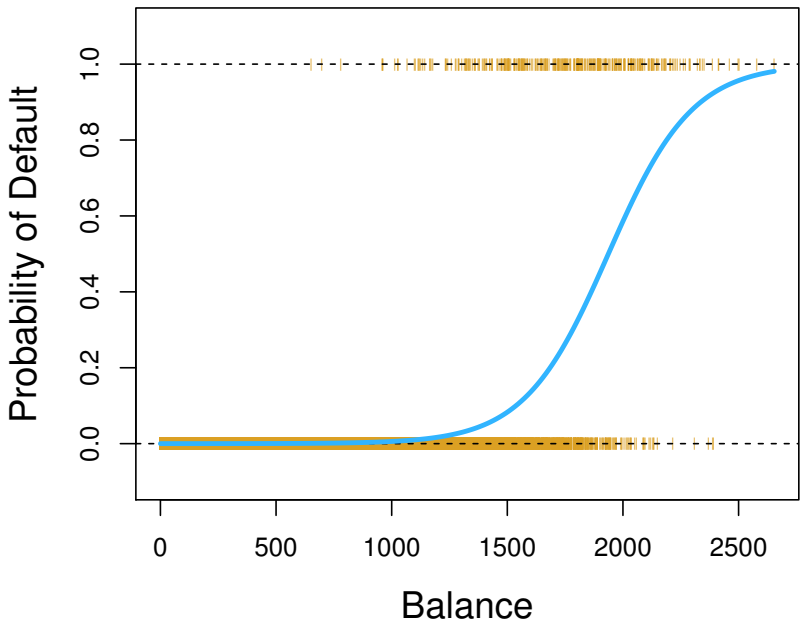
Binary Classification

– When we have a smooth approximation of the sign function, learning the parameters β_0 and β_1 is numerically easier



Binary Classification





Trying to treat the classification problem as a regression problem does not produce reasonable results!

Steps to solve problem

Regression

$f_\theta(y|x)$

↓

$-\log f(y|x)$

↓

Max Likelihood

of find $f_\theta(y|x)$

Classification

$P_\theta(Y=\ell|x)$ (label)

↓

$-\log P(Y=\ell|x)$

↓

Max Likelihood

How Does Binary Classification Work?

- We somehow learn β_0 and β_1 from the training data (will be explained soon)
- We are given a test point x_t, for which we evaluate $\beta_0 + \beta_1 x_t$
- We pass this quantity to our smooth sign approximation

$$p(x_t) = \frac{e^{\beta_0+\beta_1 x_t}}{1 + e^{\beta_0+\beta_1 x_t}}$$

- If $p(x_t)$ was closer to 1 our prediction of the class for x_t is class one (e.g., $p(x_t) = 0.7$) and if $p(x_t)$ was closer to 0 our prediction of the class for x_t is class zero (e.g., $p(x_t) = 0.3$)
- Now that $p(\cdot)$ generates some value between zero and one for us, one immediate interpretation for it is being the probability of label 1

$$p(x_t) = \mathbb{P}(y = 1|x_t) = 1 - \mathbb{P}(y = 0|x_t)$$

so if $p(x_t) = 0.7$, then the test label is 1 with probability 0.7, and 0 with probability 0.3

How to Do the Training for the Simple Logistic Regression?

- Many of the classification techniques you see in this course only differ in the way that we model

$$\mathbb{P}(Y = \ell | x_t)$$

- We observe samples $(x_1, y_1), \cdots (x_n, y_n)$, where $y_i \in \{0, 1\}$
- We want to determine β_0 and β_1 such that the probability of assigning the right labels is maximized

$$\arg\max_{\beta_0, \beta_1} \quad \mathbb{P}(Y_1 = y_1, \cdots Y_n = y_n | X_1 = x_1, \cdots, X_n = x_n, \beta_0, \beta_1)$$

- Basically, we want to find the ML estimates for β_0 and β_1

Model: $P(Y=1|X) = \sigma(\beta_0+\beta_1 X) = \frac{e^{\beta_0+\beta_1 X}}{1+e^{\beta_0+\beta_1 X}}$

$P(Y=0|X) = 1-\sigma(\beta_0+\beta_1 X) = 1-\sigma(\beta_0+\beta_1 X)$

Training Data: $(x_1, y_1), (x_2, y_2) \dots (x_n, y_n)$
(Samples)

To derive ML

→ $P(Y_1=y_1, Y_2=y_2, \dots, Y_n=y_n \mid X_1, \dots x_n, \beta_0, \beta_1)$

Since independent = $\prod_{i=1}^{n} P(Y_i=y_i \mid x_i, \beta_0, \beta_1)$

only estimates β_0, β_1

not x_i

– Since our samples are independent, we get

$$\begin{aligned}\mathbb{P}(Y_1 = y_1, \cdots Y_n = y_n | x_1, \cdots, x_n, \beta_0, \beta_1) &= \prod_{i=1}^{n} \mathbb{P}(Y_i = y_i | x_i, \beta_0, \beta_1) \\ &= \prod_{i:y_i=1} p(x_i) \prod_{i:y_i=0} (1 - p(x_i)) \\ &= \prod_{i=1}^{n} p(x_i)^{y_i} (1 - p(x_i))^{1-y_i}\end{aligned}$$

where the first equality is thanks to

$$p(x_i) = \mathbb{P}(Y = 1 | x_i) = 1 - \mathbb{P}(y = 0 | x_i)$$

– So we ultimately want to find β_0 and β_1 that maximize

$$\prod_{i=1}^{n} p(x_i)^{y_i} (1-p(x_i))^{1-y_i} = \prod_{i=1}^{n} \left(\frac{e^{\beta_0+\beta_1 x_i}}{1 + e^{\beta_0+\beta_1 x_i}} \right)^{y_i} \left(1 - \frac{e^{\beta_0+\beta_1 x_i}}{1 + e^{\beta_0+\beta_1 x_i}} \right)^{1-y_i}$$

Some Notes on The Logistic Regression

$-\log(\,) \rightarrow$ convert Max to min, then we can use GD

- In logistic regression, we end up with a more complex cost function to optimize (after applying the negative log we get)

$$
\begin{aligned}
L(\beta_0, \beta_1) &= -\log\left(\prod_{i=1}^{n}\left(\frac{1}{1+e^{-\beta_0-\beta_1 x_i}}\right)^{y_i}\left(\frac{1}{1+e^{\beta_0+\beta_1 x_i}}\right)^{1-y_i}\right) \\
&= -\sum_{i=1}^{n} -y_i \log\left(1+e^{-\beta_0-\beta_1 x_i}\right) - (1-y_i)\log\left(1+e^{\beta_0+\beta_1 x_i}\right) \\
&= \sum_{i=1}^{n} y_i \log\left(1+e^{-\beta_0-\beta_1 x_i}\right) + (1-y_i)\log\left(1+e^{\beta_0+\beta_1 x_i}\right)
\end{aligned}
$$

- This function is convex and can be nicely minimized using gradient descent. You may see examples in the homework!

Turns out that

$$f(z) = y \cdot \log\left(1+e^{-z}\right) + (1-y)\log\left(1+e^{z}\right) \quad \text{is convex in } z$$

$$z = \beta_0 + \beta_1 x$$

$$f(\beta_0, \beta_1) = y \cdot \log\left(1+e^{-\beta_0-\beta_1 x}\right) + (1-y)\log\left(1+e^{\beta_0+\beta_1 x}\right) \text{ is convex in } \beta_0, \beta_1$$

Logistic Regression has no closed form solution to β_0 and β_1

What Happens for More than One Feature?

- In case of multiple features, only minor modification is required
- We still try to maximize $\prod_{i=1}^{n} p(x_i)^{y_i}(1-p(x_i))^{1-y_i}$, but now we have

$$p(x_t) = \frac{e^{\beta_0+\beta_1 x_1+\beta_2 x_2+\cdots+\beta_p x_p}}{1+e^{\beta_0+\beta_1 x_1+\beta_2 x_2+\cdots+\beta_p x_p}}$$

- We run the maximization to estimate $\beta_0, \beta_1, \cdots, \beta_p$
- In practice you never have to do the maximization and most software such as R, Python and Matlab have packages to do that numerically

① Only one feature $x, y \in \{0, 1\} \Rightarrow$ ML formula is on P.32

② Multiple features , $x_1, x_2 \ldots x_n$, $y \in \{0, 1\}$

$$P(Y=1|X) = \sigma\left(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n\right)$$

What Happens for More than Two Classes?

- Example, based on some features such as city, year of education and number of publications, classify the students of a class into undergrads, Masters, and PhDs
- Recall our method of classification in the binary case, we evaluated $p(x_t)$ which was technically $\mathbb{P}(Y = 1|x_t)$ and if it was closer to 1 then our class prediction was 1, if it was small, then $\mathbb{P}(Y = 0|x_t) = 1 - \mathbb{P}(Y = 1|x_t)$ would be large and our prediction is class zero
- One way of interpreting this is evaluating $\mathbb{P}(Y = k|x_t)$ for $k = 0, 1$ and the k that produces the largest value for $\mathbb{P}(Y = k|x_t)$ is our predicted label
- Now for K labels, we evaluate $\mathbb{P}(Y = k|x_t)$ for $k = 1, 2, \cdots, K$ and the k that produces the largest value for $\mathbb{P}(Y = k|x_t)$ is our predicted label

What Happens for More than Two Classes?

- For K labels, we evaluate $\mathbb{P}(Y = k|x_t)$ for $k = 1, 2, \cdots, K$ and the k that produces the largest value for $\mathbb{P}(Y = k|x_t)$ is our predicted label
- When we have $K > 2$ labels (e.g., $y \in$ {while, yellow, green}) and p features $x_1, x_2, \cdots, x_p$, we fit K models parametrized by

$$\begin{aligned} \text{Label 1:} \quad & \{\beta_0^{(1)}, \beta_1^{(1)}, \cdots, \beta_p^{(1)}\} \\ \text{Label 2:} \quad & \{\beta_0^{(2)}, \beta_1^{(2)}, \cdots, \beta_p^{(2)}\} \\ & \vdots \\ \text{Label K:} \quad & \{\beta_0^{(K)}, \beta_1^{(K)}, \cdots, \beta_p^{(K)}\} \end{aligned}$$

- For this problem we consider the following form:

$$p_k(\boldsymbol{x}) = \mathbb{P}(Y = k|\boldsymbol{x}) = \frac{e^{\beta_0^{(k)} + \cdots + \beta_p^{(k)} x_p}}{e^{\beta_0^{(1)} + \cdots + \beta_p^{(1)} x_p} + \cdots + e^{\beta_0^{(K)} + \cdots + \beta_p^{(K)} x_p}}$$

- What is the sum of all $\mathbb{P}(Y = k|\boldsymbol{x})$ for a fixed $\boldsymbol{x}$?

What if 3 classes ?

Classes $y \in \{0, 1, 2\}$

$\rightarrow P(Y=0|X) + P(Y=1|X) + P(Y=2|X) = 1$

Therefore, for more two classes:

$P(Y=l|X)$ $l \in$ the class labels

Have to deal with each model individually

$$\begin{cases} P(Y=1|X): \beta_0^{(1)}, \beta_1^{(1)}, \cdots \beta_p^{(1)} \\ P(Y=2|X): \beta_0^{(2)}, \beta_1^{(2)}, \cdots \beta_p^{(2)} \\ \vdots \\ P(Y=L|X): \beta_0^{(L)}, \beta_1^{(L)}, \cdots \beta_p^{(L)} \end{cases}$$

Then Make Sure

$$\sum_{Y=1}^{L} P(Y=l|X) = 1$$

Linear and Quadratic Discriminant Analysis

1. Regression Model

Ex: $\begin{cases} \beta_0 + \beta_1 x \\ \beta_0 + \beta_1 x_1 + \dots + \beta_p x_p \end{cases}$ $\quad y = g(x) \quad \beta = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_p \end{bmatrix}$

$PDF = f(y|\beta, x) \xrightarrow{-\log} -\log f(y|\beta, x)$

$\xrightarrow{\text{minimize}}$ to estimate β

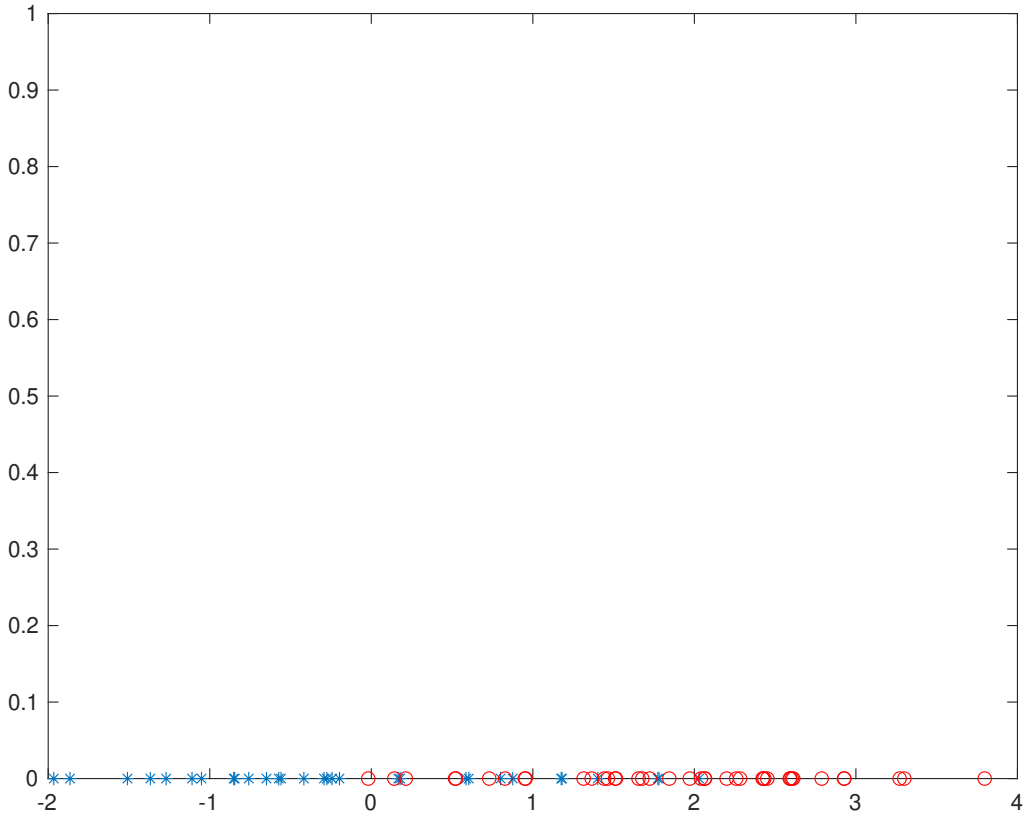
2. Classification $\quad y \in \{0, 1, 2, \dots, L\}$

$P(y=l \mid \beta^{(l)}, x), \quad l = 0, 1, 2, \dots, L$

$\xrightarrow{-\log} -\log P(y=l \mid \beta^{(l)}, x)$

$\xrightarrow{\text{minimize}}$ estimate $\beta^{(l)}$

LDA/QDA Story in Simple Words

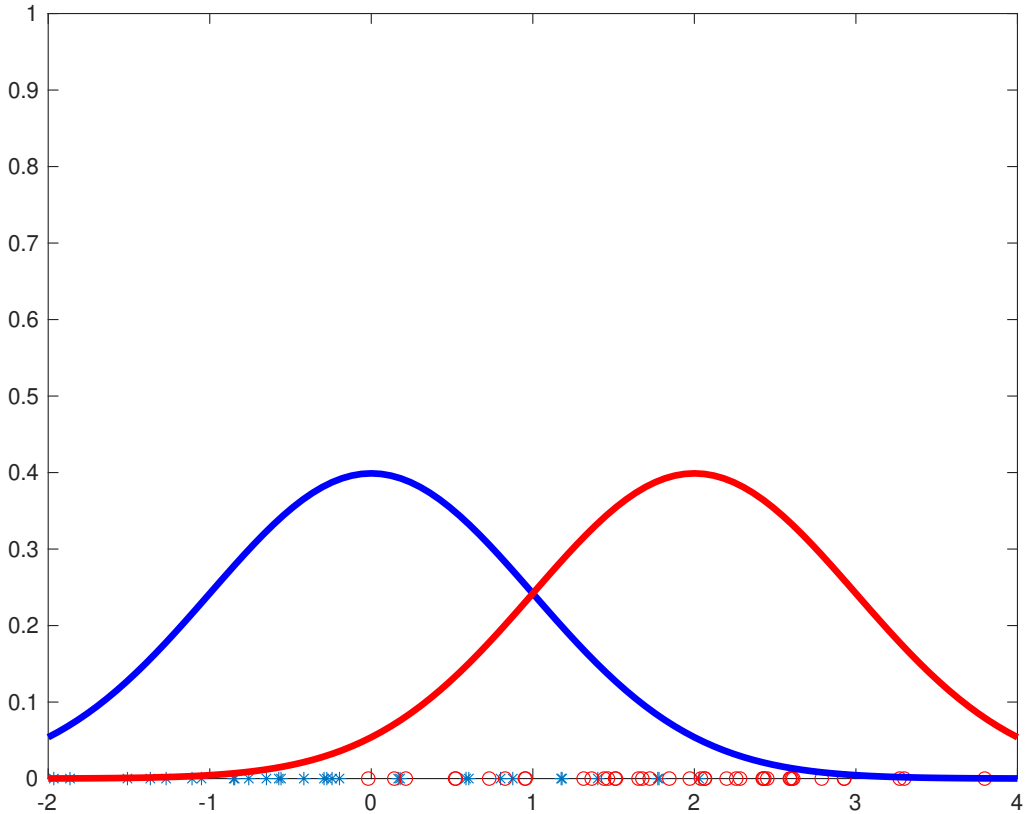


Let's do an exercise. We are given 30 blue points and 40 red points as above. Let's fit a Normal pdf to each cloud

$$X \sim (\mu, \sigma^2)$$

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2}\right)$$

LDA/QDA Story in Simple Words

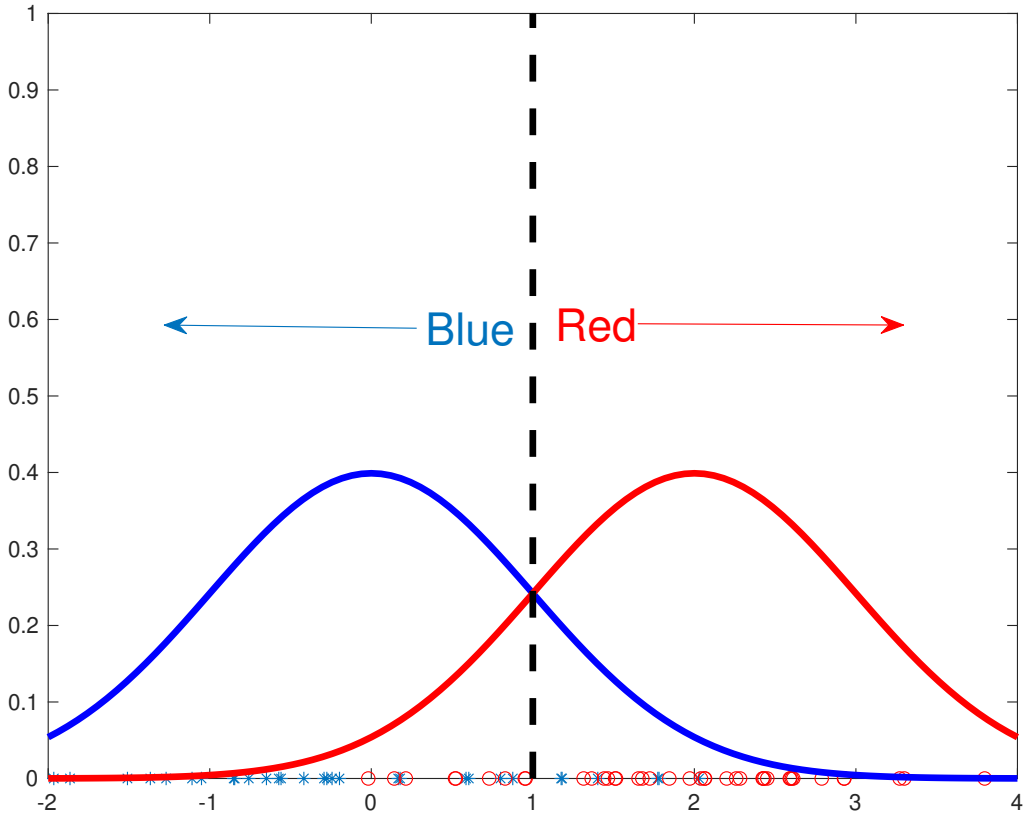


Let's assume both variances turn out similar. Find the intersection point. What can you say about this point?

LDA/QDA Story in Simple Words

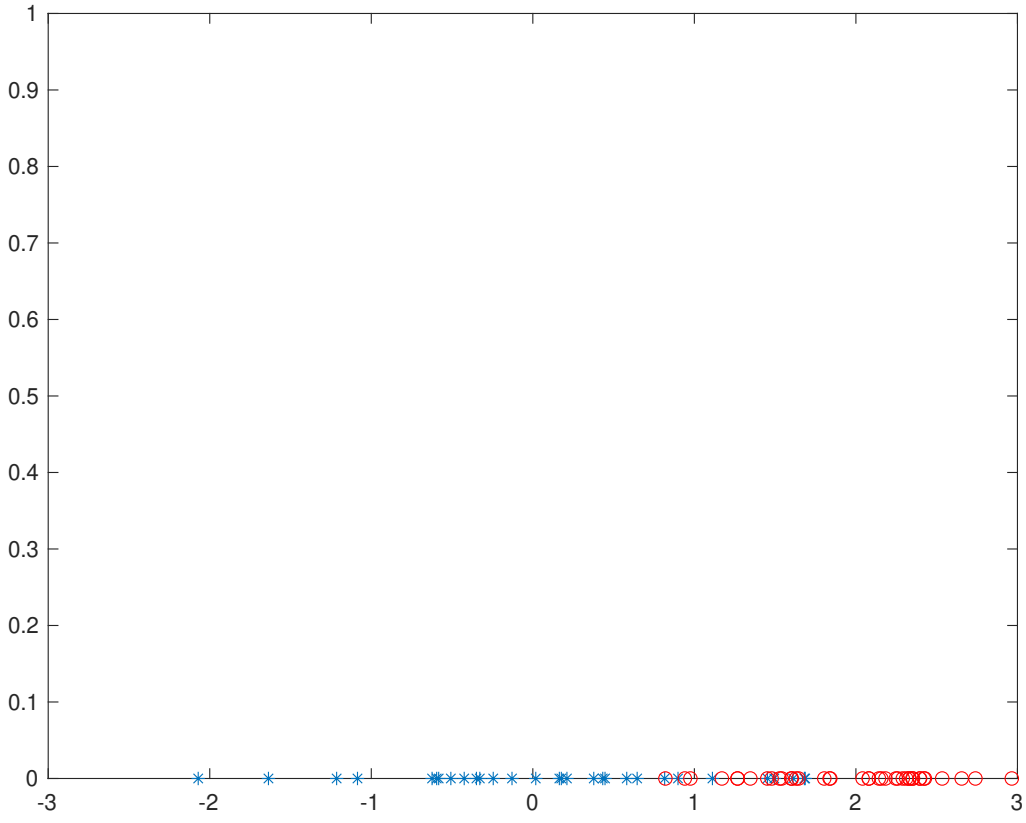
One dimension

1. Just Average the samples



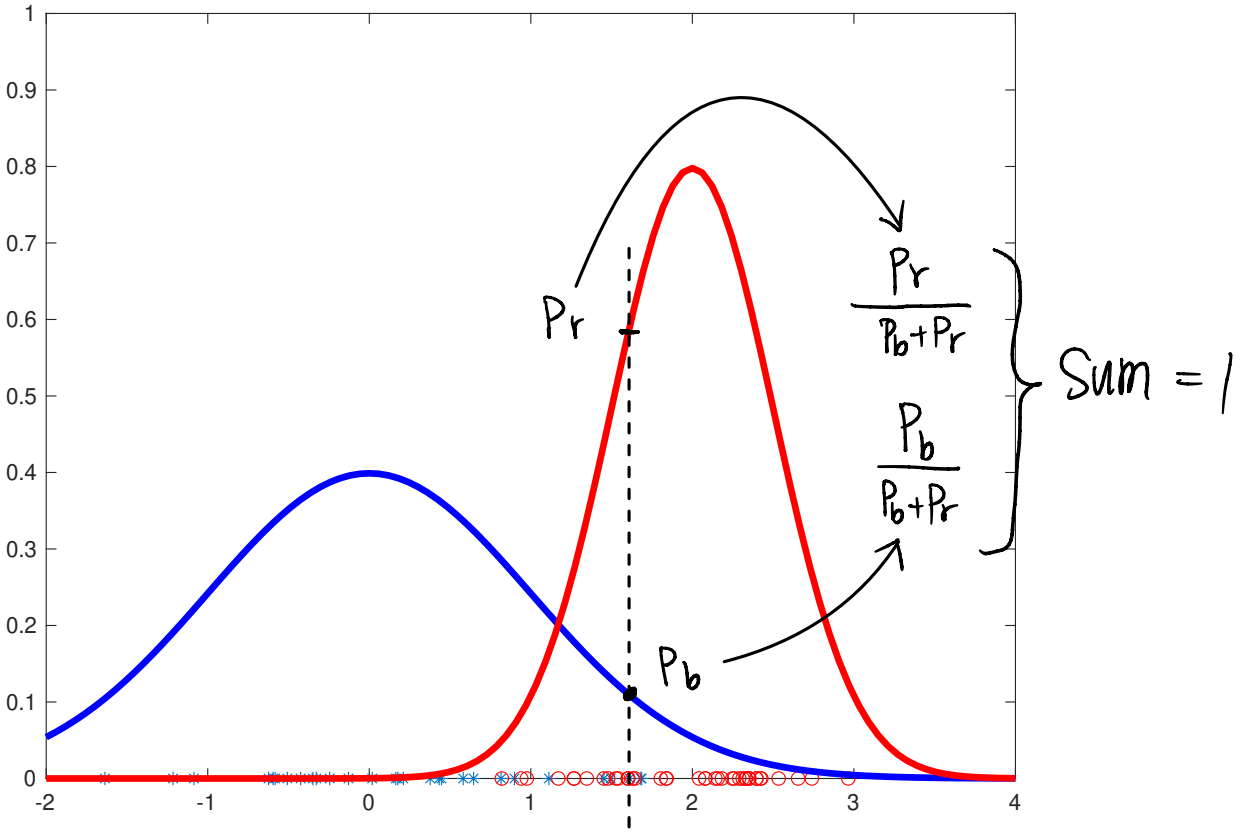
We have found a rule to classify each point on the real line

LDA/QDA Story in Simple Words



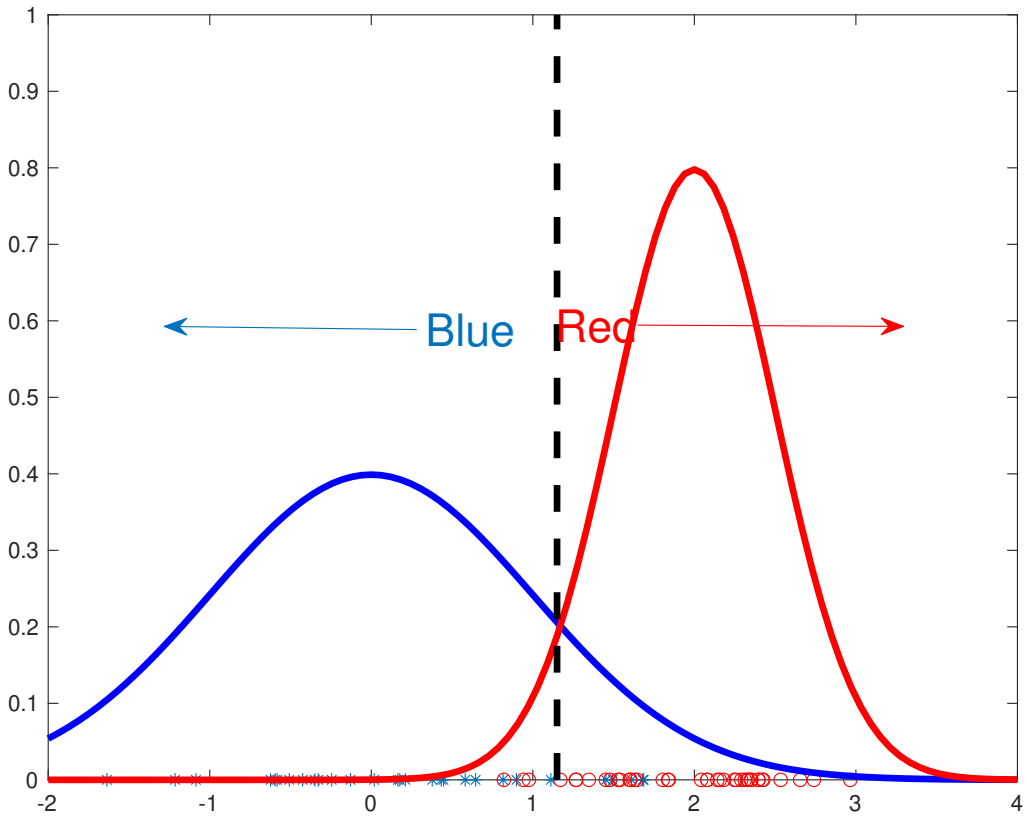
Let's do a different exercise with another set of points

LDA/QDA Story in Simple Words



The intersection point changes since the variances are no more the same.

LDA/QDA Story in Simple Words



Again we have a rule. If you want to assign a probability of being blue and being red to each point, how would you calculate that probability?

$$\mathbb{P}(Y = B|x) = p_B(x) = \frac{f_B(x)}{f_B(x) + f_R(x)}, \quad \mathbb{P}(Y = R|x) = p_R(x) = \frac{f_R(x)}{f_B(x) + f_R(x)}$$

LDA/QDA Story in Simple Words

$f_\ell = f(x|y=\ell)$

- Say you want to incorporate the probability of each class in your calculation of $\mathbb{P}(Y|x)$. We can use the Bayes formula

– Probabilistically, suppose that our y can take K distinct values. By the Bayes' theorem we have

Given Data $f(x|Y=\ell)$ → PDF

$$\mathbb{P}(Y=\ell|\boldsymbol{X}=\boldsymbol{x}) = \frac{\mathbb{P}(Y=\ell)\mathbb{P}(\boldsymbol{X}=\boldsymbol{x}|Y=\ell)}{\sum_{k=1}^{K}\mathbb{P}(Y=k)\mathbb{P}(\boldsymbol{X}=\boldsymbol{x}|Y=k)}$$

π_ℓ is the proportion $f(x|Y=k)$

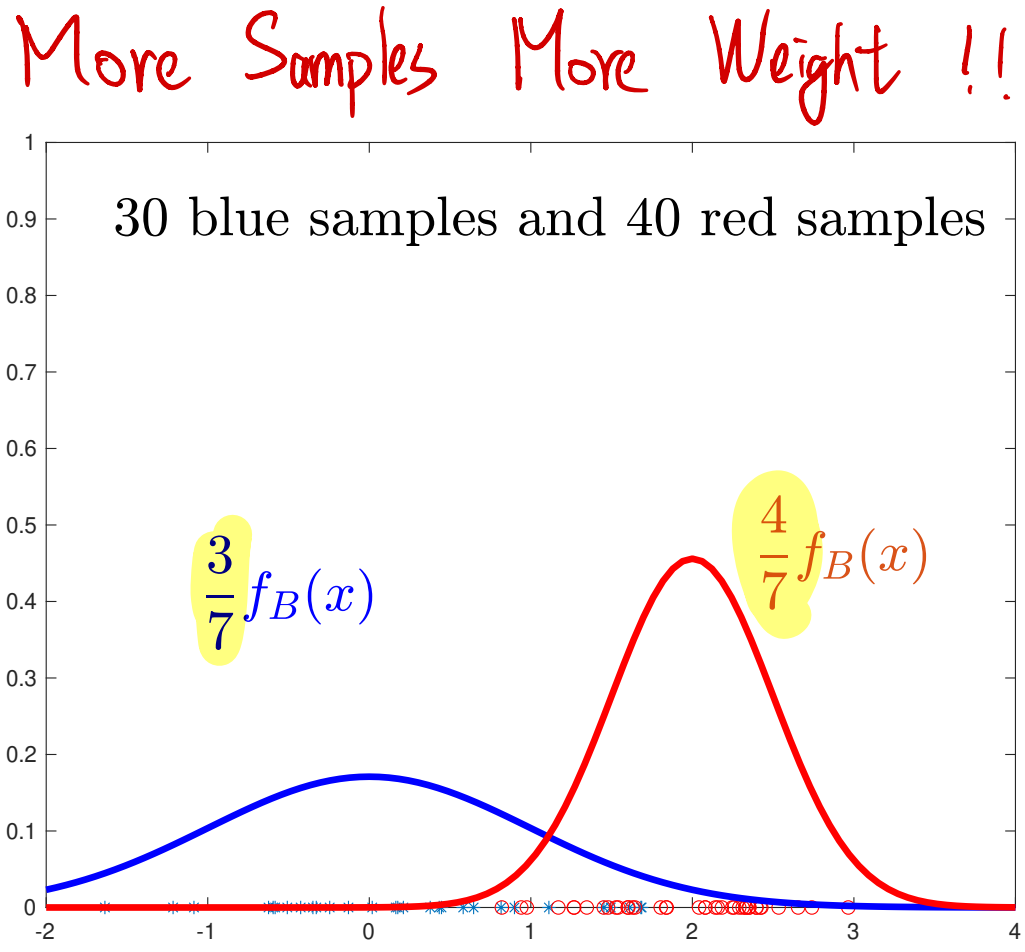
$$= \frac{\pi_\ell f_\ell(\boldsymbol{x})}{\sum_{k=1}^{K}\pi_k f_k(\boldsymbol{x})}$$

– Let's see why this equality holds, knowing the Bayes' equality

$$\mathbb{P}(A|B) = \frac{\mathbb{P}(A\cap B)}{\mathbb{P}(B)}$$

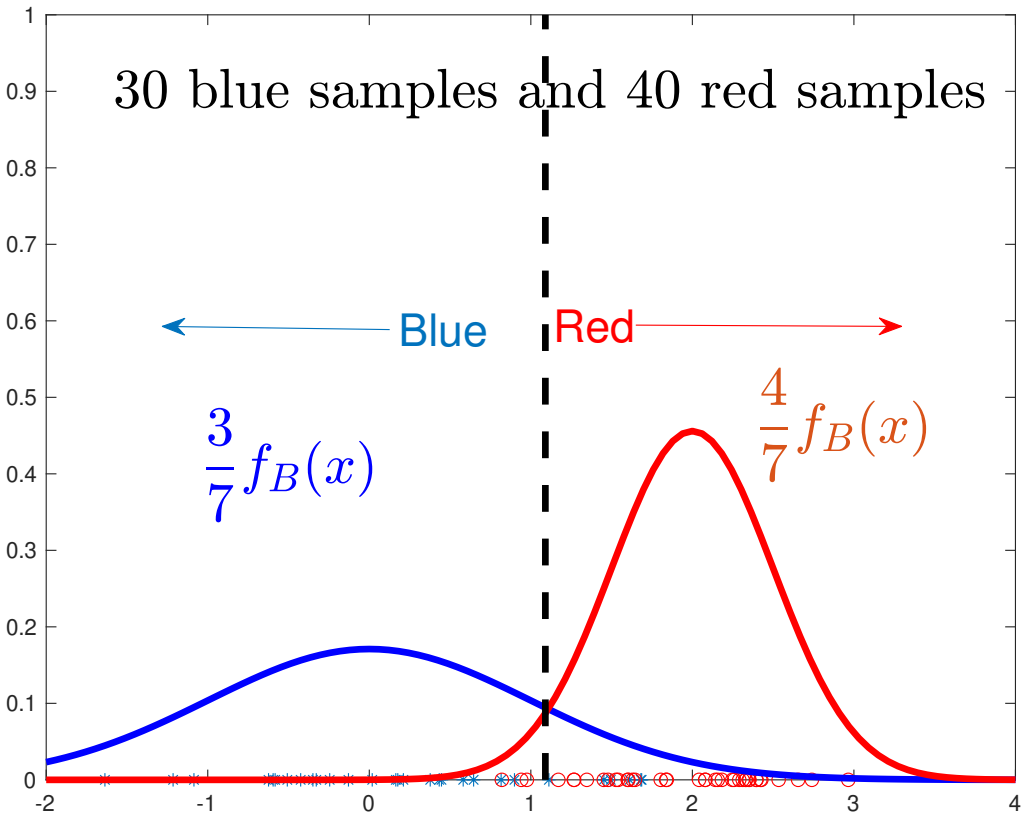
and $A_1 \cup A_2 \cdots \cup A_K$ covering the entire space, where $A_i \cap A_j = \emptyset$.

LDA/QDA Story in Simple Words



The intersection point yet changes

LDA/QDA Story in Simple Words

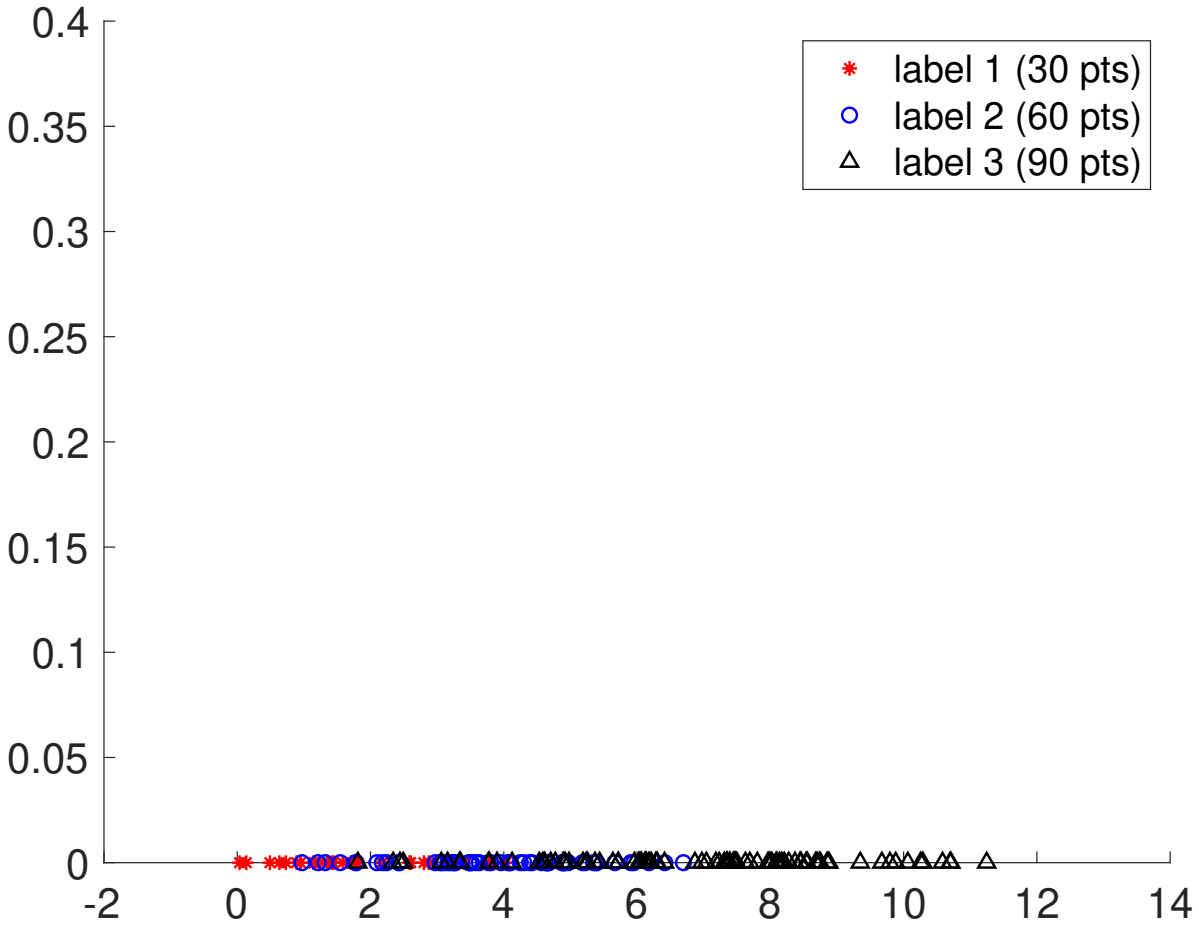


$$\mathbb{P}(Y = B|x) = p_B(x) = \frac{\frac{3}{7}f_B(x)}{\frac{3}{7}f_B(x) + \frac{4}{7}f_R(x)},$$

$$\mathbb{P}(Y = R|x) = p_R(x) = \frac{\frac{4}{7}f_R(x)}{\frac{3}{7}f_B(x) + \frac{4}{7}f_R(x)}$$

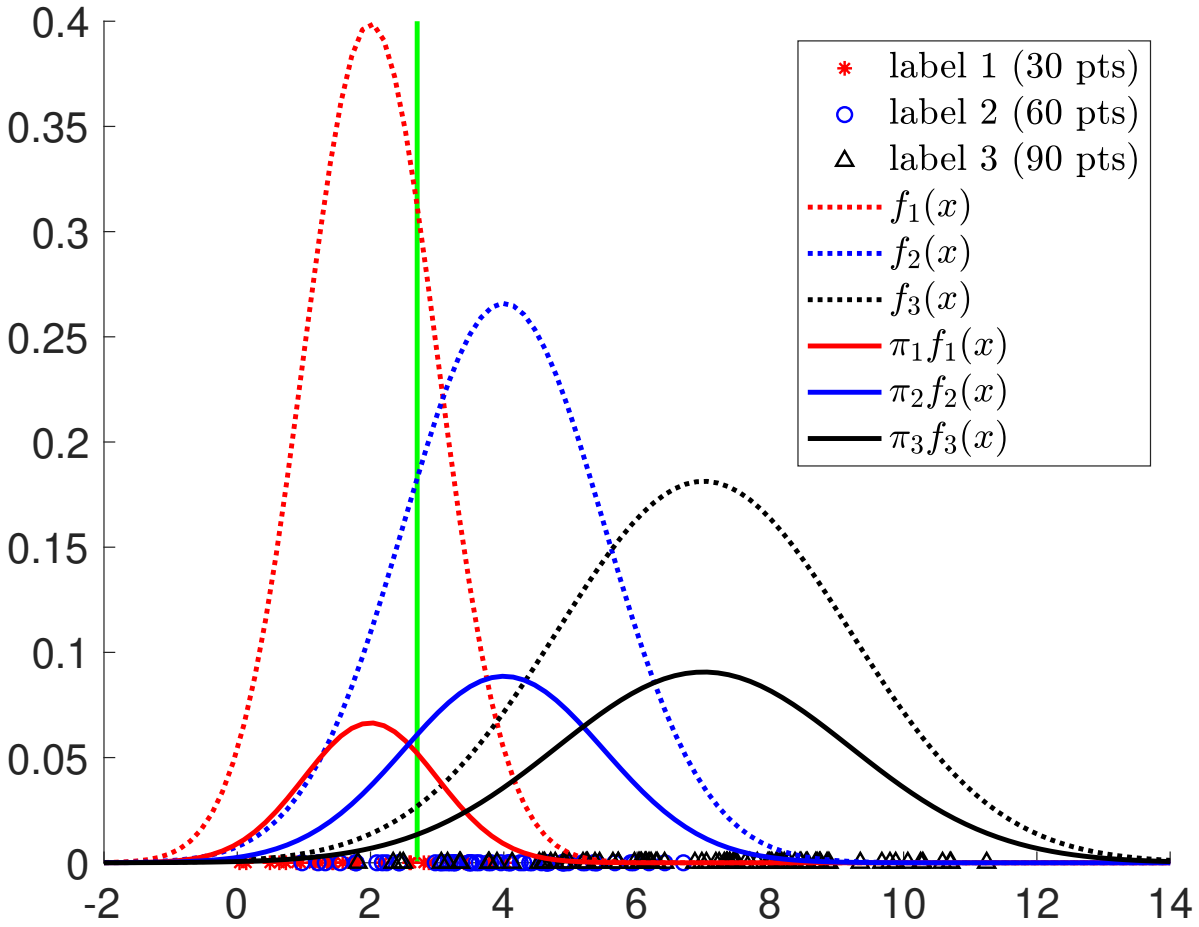
LDA/QDA Story in Simple Words

Question. What if we have more than two classes?



Story in Simple Words

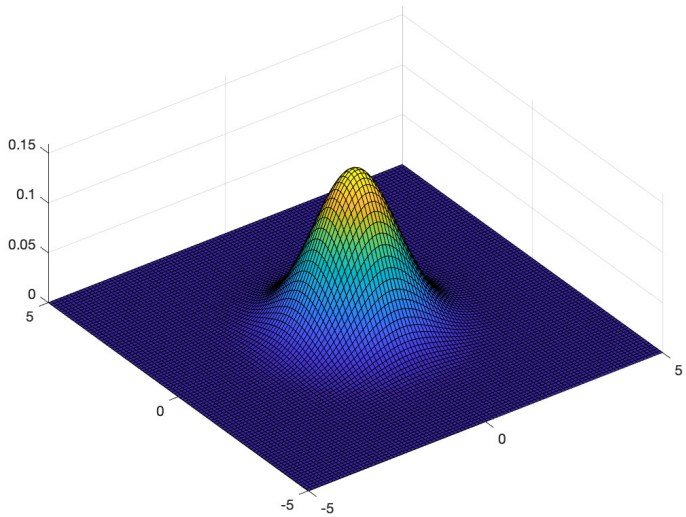
Answer. Same idea!

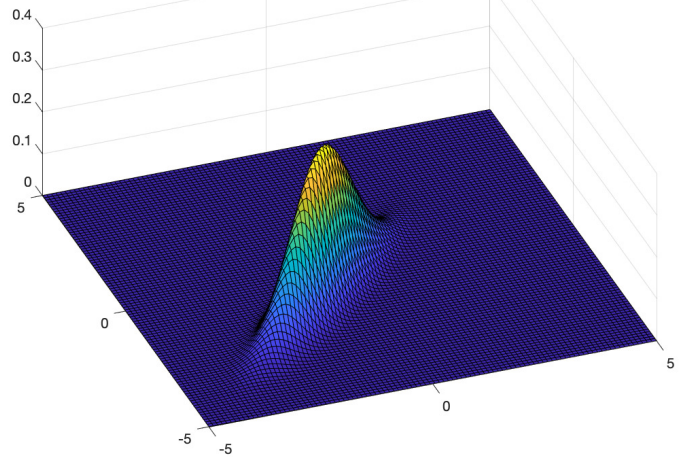


Story in Simple Words

Goal. Higher dimensions,

$$\boldsymbol{\mu}_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \boldsymbol{\Sigma}_1 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \boldsymbol{\mu}_2 = \begin{pmatrix} -1 \\ -1 \end{pmatrix}, \boldsymbol{\Sigma}_2 = \begin{pmatrix} 1 & 0.9 \\ 0.9 & 1 \end{pmatrix}$$





Also, in higher dimensions the intersection between two Normal pdfs is no more just a point, it would be a boundary

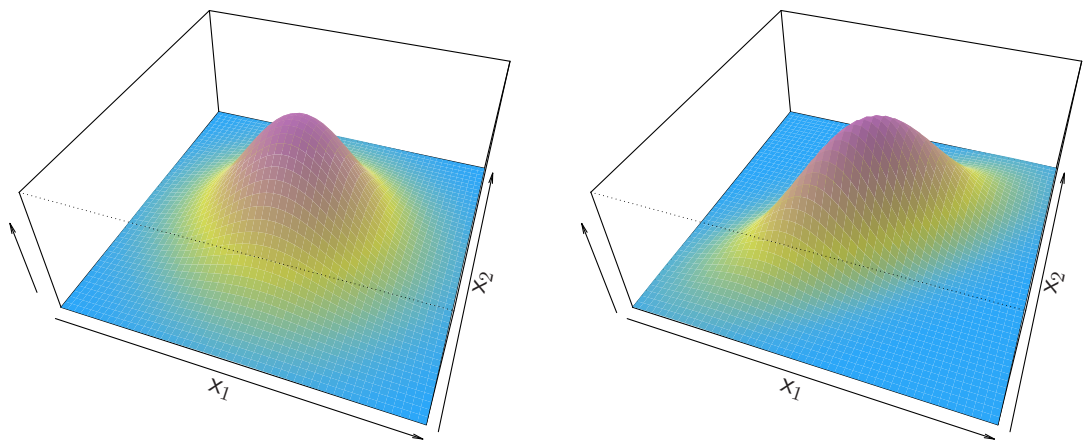
Little Introduction about Multivariate Normal

- Recall the normal distribution for a random variable x:

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp(-\frac{(x-\mu)^2}{2\sigma^2})$$

- Similar to the scalar case, we can define a distribution for the random vector $\boldsymbol{x} = [x_1, \cdots, x_p]^T$ as

$$f(\boldsymbol{x}) = \frac{1}{(2\pi)^{p/2}|\boldsymbol{\Sigma}|^{1/2}} \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right)$$



Linear Discriminant Analysis (LDA)

- Recall

$$\mathbb{P}(Y = \ell | \boldsymbol{X} = \boldsymbol{x}) = \frac{\mathbb{P}(Y = \ell)\mathbb{P}(\boldsymbol{X} = \boldsymbol{x} | Y = \ell)}{\sum_{k=1}^{K} \mathbb{P}(Y = k)\mathbb{P}(\boldsymbol{X} = \boldsymbol{x} | Y = k)} = \frac{\pi_\ell f_\ell(\boldsymbol{x})}{\sum_{k=1}^{K} \pi_k f_k(\boldsymbol{x})}$$

- The purpose of LDA is learning a model for $\mathbb{P}(Y = \ell | \boldsymbol{X} = \boldsymbol{x})$
- In the formulation above, $f_\ell(\boldsymbol{x})$ is in a sense the distribution we consider for the data points in class ℓ, and π_ℓ is the probability that we pick some random sample and it belongs to class ℓ
- In LDA, we assume that all $f_\ell(\boldsymbol{x})$ have a multivariate normal distribution with similar covariances and different means, i.e.

$$f_\ell(\boldsymbol{x}) = \frac{1}{(2\pi)^{p/2}|\boldsymbol{\Sigma}|^{1/2}} \exp\left(-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu}_\ell)^\top \boldsymbol{\Sigma}^{-1}(\boldsymbol{x} - \boldsymbol{\mu}_\ell)\right)$$

- Unlike logistic regression, which involved a rather complicated maximization for learning, in LDA we have closed form expressions for π_ℓ, $\boldsymbol{\mu}_\ell$ and $\boldsymbol{\Sigma}$ and classifying new test points becomes very easy

Linear Discriminant Analysis (LDA)

– Given a training set $(\boldsymbol{x}_1, y_1), \cdots, (\boldsymbol{x}_n, y_n)$ where the responses y can take K distinct class values $1, 2, \cdots, K$, we can easily learn the LDA model by calculating π_ℓ, $\boldsymbol{\mu}_\ell$ and $\boldsymbol{\Sigma}$ via (considering c_ℓ to be the index of samples in class ℓ)

1. Calculate Proportion

$$\hat{\pi}_\ell = \frac{\#\text{ of elements in } c_\ell}{n}$$

2. Just Average samples

$$\hat{\boldsymbol{\mu}}_\ell = \frac{1}{\#\text{ of elements in } c_\ell} \sum_{i \in c_\ell} \boldsymbol{x}_i$$

$$\hat{\boldsymbol{\Sigma}} = \frac{1}{N - K} \sum_{k=1}^{K} \sum_{i \in c_\ell} (\boldsymbol{x}_i - \hat{\boldsymbol{\mu}}_\ell)(\boldsymbol{x}_i - \hat{\boldsymbol{\mu}}_\ell)^\top$$

Covariance matrix

?

– After this point for a new test point $\boldsymbol{x}_t$ we have all that is needed to calculate $\mathbb{P}(Y = \ell | \boldsymbol{X} = \boldsymbol{x}_t)$ for $\ell = 1, \cdots, K$ and pick as the label the one that is largest

X_1	X_2	y
1	2	r
-1	-1	b
0	3	r
-2	1	r

$$\mu_r = \frac{\begin{bmatrix}1\\2\end{bmatrix} + \begin{bmatrix}0\\3\end{bmatrix} + \begin{bmatrix}-2\\1\end{bmatrix}}{3} = \begin{bmatrix}\frac{-1}{3}\\2\end{bmatrix}$$

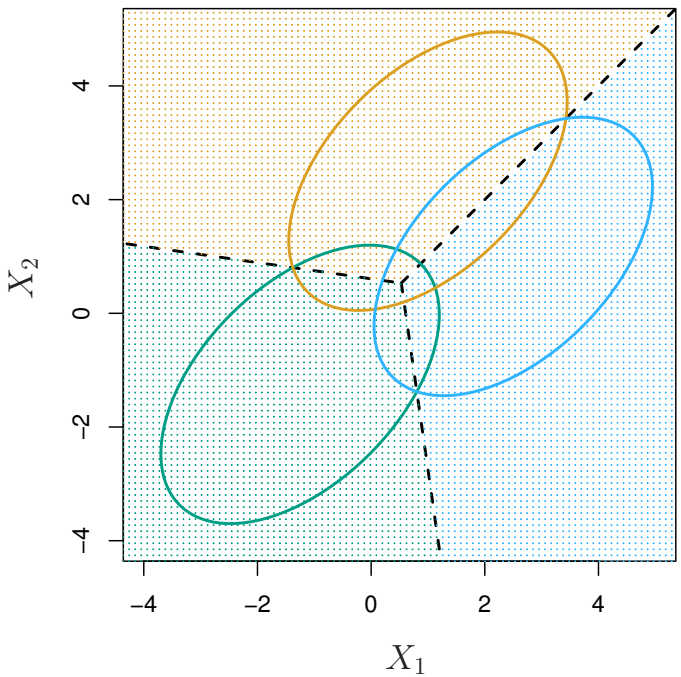
$$\mu_b = \begin{bmatrix}-1\\-1\end{bmatrix}$$

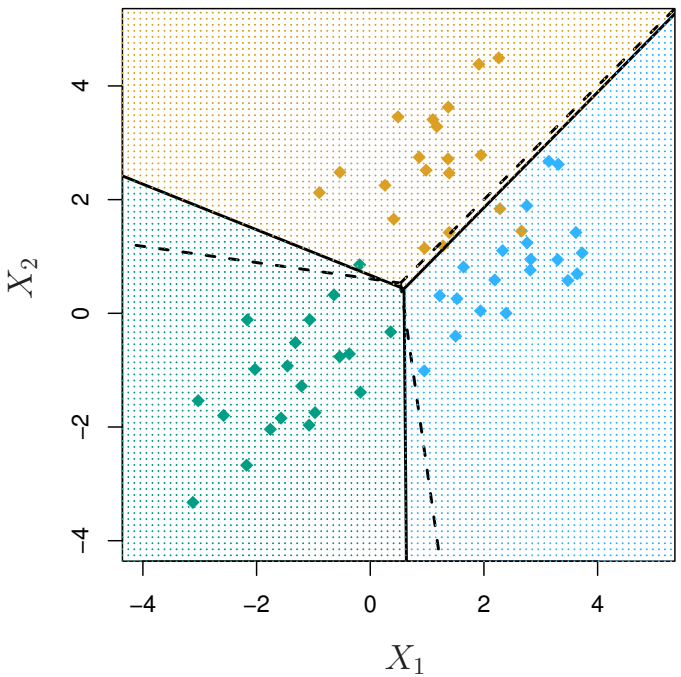
$$\Sigma_r = \left(\begin{bmatrix}1\\2\end{bmatrix} - \begin{bmatrix}\frac{-1}{3}\\2\end{bmatrix}\right)\left(\begin{bmatrix}1\\2\end{bmatrix} - \begin{bmatrix}\frac{-1}{3}\\2\end{bmatrix}\right)^T$$

$$+ \left(\begin{bmatrix}0\\3\end{bmatrix} - \begin{bmatrix}\frac{-1}{3}\\2\end{bmatrix}\right)\left(\begin{bmatrix}0\\3\end{bmatrix} - \begin{bmatrix}\frac{-1}{3}\\2\end{bmatrix}\right)^T$$

$$+ \left(\begin{bmatrix}-2\\1\end{bmatrix} - \begin{bmatrix}\frac{-1}{3}\\2\end{bmatrix}\right)\left(\begin{bmatrix}-2\\1\end{bmatrix} - \begin{bmatrix}\frac{-1}{3}\\2\end{bmatrix}\right)^T$$

Linear Discriminant Analysis (LDA)





Assume there are 2 classes

$$\pi_1 f_1(x) \longrightarrow \frac{\pi_1}{(2\pi)^{p/2}\,|\Sigma|^{\frac{1}{2}}} \exp\left(-\frac{(x-\mu_1)^T \Sigma^{-1} (x-\mu_1)}{2}\right)$$

$$\pi_2 f_2(x) \longrightarrow \frac{\pi_2}{(2\pi)^{p/2}\,|\Sigma|^{\frac{1}{2}}} \exp\left(-\frac{(x-\mu_2)^T \Sigma^{-1} (x-\mu_2)}{2}\right)$$

Decision Boundary : $\{x : \pi_1 f_1(x) = \pi_2 f_2(x)\}$

$$\pi_1 \exp\left(\frac{-1}{2}(x-\mu_1)^T \Sigma^{-1} (x-\mu_1)\right) = \pi_2$$

$\xrightarrow{\log}$ $\log \pi_1 - \frac{1}{2}(x-\mu_1)^T \Sigma^{-1} (x-\mu_1)$

$$\rightarrow \log \pi_1 - \frac{1}{2}\left[\underbrace{x^T \Sigma^{-1} x}_{=} - \mu_1^T \Sigma^{-1} x - x^T \Sigma^{-1} \mu_1 + \mu_1^T \Sigma^{-1} \mu_1\right]$$

$$= \log \pi_2 - \frac{1}{2}\left[x^T \Sigma^{-1} x - \mu_2^T \Sigma^{-1} x - x^T \Sigma^{-1} \mu_2 + \mu_2^T \Sigma^{-1} \mu_2\right]$$

Linear Discriminant Analysis (LDA)

- In practice to assign a label to a given test point $\boldsymbol{x}_t$ we do not need to calculate

$$\mathbb{P}(Y = \ell | \boldsymbol{X} = \boldsymbol{x}_t) = \frac{\pi_\ell f_\ell(\boldsymbol{x}_t)}{\sum_{k=1}^{K} \pi_k f_k(\boldsymbol{x}_t)}$$

  and only comparing $\pi_\ell f_\ell(\boldsymbol{x}_t)$ is enough

- This reduces to evaluate

$$\delta_\ell = \boldsymbol{x}_t^\top \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_\ell - \frac{1}{2} \boldsymbol{\mu}_\ell^\top \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_\ell + \log \pi_\ell$$

  and pick as the class ℓ corresponding to the largest δ_ℓ

- You can find the decision boundary between class i and j by finding the points for which $\delta_i = \delta_j$

- [see the sample Matlab code]

Quadratic Discriminant Analysis (QDA)

- Recall

$$\mathbb{P}(Y = \ell | \boldsymbol{X} = \boldsymbol{x}_t) = \frac{\pi_\ell f_\ell(\boldsymbol{x}_t)}{\sum_{k=1}^{K} \pi_k f_k(\boldsymbol{x}_t)}$$

- The purpose of QDA is learning a model for $\mathbb{P}(Y = \ell | \boldsymbol{X} = \boldsymbol{x})$ in a more flexible way compared to LDA
- In QDA, we assume that all $f_\ell(\boldsymbol{x})$ have a multivariate normal distribution with similar covariances and different means, i.e.

$$f_\ell(\boldsymbol{x}) = \frac{1}{(2\pi)^{p/2} |\boldsymbol{\Sigma}_\ell|^{1/2}} \exp\left(-\frac{1}{2} (\boldsymbol{x} - \boldsymbol{\mu}_\ell)^\top \boldsymbol{\Sigma}_\ell^{-1} (\boldsymbol{x} - \boldsymbol{\mu}_\ell) \right)$$

- The main difference between LDA and QDA is in LDA we consider a single $\boldsymbol{\Sigma}$ for all classes, but in QDA we allow more flexibility by having a different covariance matrix for each class
- Similar to LDA, QDA can be learned easily and we can obtain closed form expressions for π_ℓ, $\boldsymbol{\mu}_\ell$ and $\boldsymbol{\Sigma}_\ell$

Quadratic Discriminant Analysis (QDA)

– Given a training set $(\boldsymbol{x}_1, y_1), \cdots, (\boldsymbol{x}_n, y_n)$ where the responses y can take K distinct class values $1, 2, \cdots, K$, we can easily learn the QDA model by calculating π_ℓ, $\boldsymbol{\mu}_\ell$ and $\boldsymbol{\Sigma}_\ell$ via (considering c_ℓ to be the index of samples in class ℓ)

$$\hat{\pi}_\ell = \frac{\#\text{ of elements in } c_\ell}{n}$$

$$\hat{\boldsymbol{\mu}}_\ell = \frac{1}{\#\text{ of elements in } c_\ell} \sum_{i \in c_\ell} \boldsymbol{x}_i$$

$$\hat{\boldsymbol{\Sigma}}_\ell = \frac{1}{\#\text{ of elements in } c_\ell - 1} \sum_{i \in c_\ell} (\boldsymbol{x}_i - \hat{\boldsymbol{\mu}}_\ell)(\boldsymbol{x}_i - \hat{\boldsymbol{\mu}}_\ell)^\top$$

– After this point for a new test point $\boldsymbol{x}_t$ we have all that is needed to calculate $\mathbb{P}(Y = \ell | \boldsymbol{X} = \boldsymbol{x}_t)$ for $\ell = 1, \cdots, K$ and pick as the label the one that is largest

Assume there are 2 classes

$$\pi_1 f_1(x) \longrightarrow \frac{\pi_1}{(2\pi)^{p/2}\,|\Sigma|^{\frac{1}{2}}} \exp\left(-\frac{(x-\mu_1)^T \Sigma^{-1} (x-\mu_1)}{2}\right)$$

$$\pi_2 f_2(x) \longrightarrow \frac{\pi_2}{(2\pi)^{p/2}\,|\Sigma|^{\frac{1}{2}}} \exp\left(-\frac{(x-\mu_2)^T \Sigma^{-1} (x-\mu_2)}{2}\right)$$

Decision Boundary : $\{x : \pi_1 f_1(x) = \pi_2 f_2(x)\}$

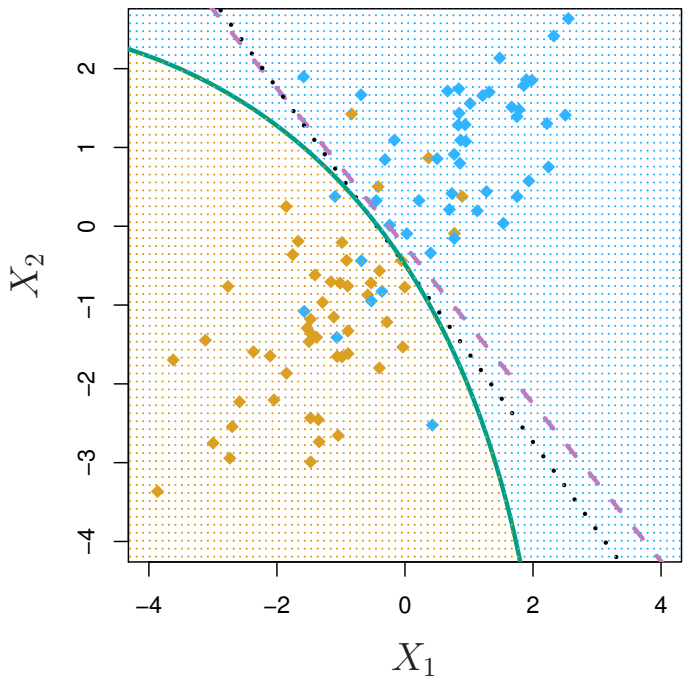
$$\pi_1 \exp\left(\frac{-1}{2}(x-\mu_1)^T \Sigma^{-1}(x-\mu_1)\right) = \pi_2$$

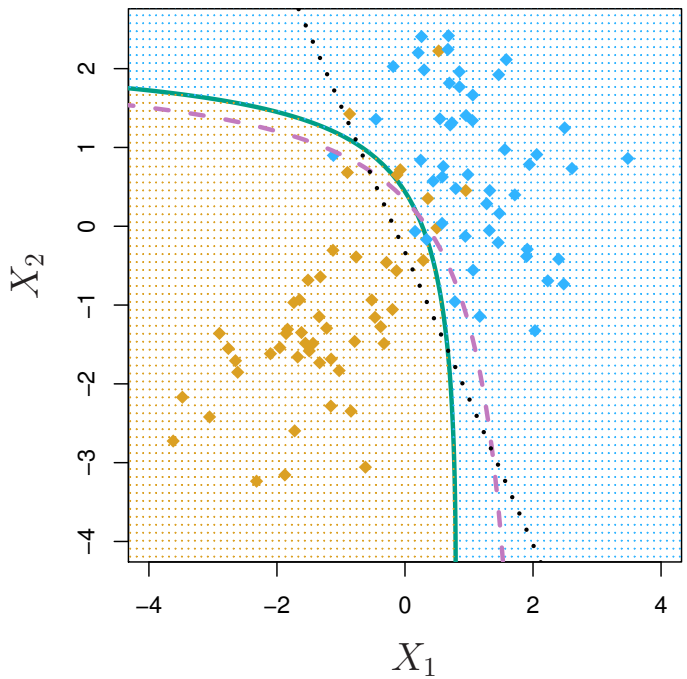
$$\xrightarrow{\log} \log \pi_1 - \frac{1}{2}(x-\mu_1)^T \Sigma^{-1}(x-\mu_1)$$

$$\rightarrow \log \pi_1 - \frac{1}{2}\left[x^T \Sigma_1^{-1} x - \mu_1^T \Sigma_1^{-1} x - x^T \Sigma_1^{-1} \mu_1 + \mu_1^T \Sigma_1^{-1} \mu_1\right]$$

$$= \log \pi_2 - \frac{1}{2}\left[x^T \Sigma_2^{-1} x - \mu_2^T \Sigma_2^{-1} x - x^T \Sigma_2^{-1} \mu_2 + \mu_2^T \Sigma_2^{-1} \mu_2\right]$$

Quadratic Discriminant Analysis (QDA)





Quadratic Discriminant Analysis (QDA)

- In practice to assign a label to a given test point $\boldsymbol{x}_t$ we do not need to calculate

$$\mathbb{P}(Y = \ell | \boldsymbol{X} = \boldsymbol{x}_t) = \frac{\pi_\ell f_\ell(\boldsymbol{x}_t)}{\sum_{k=1}^{K} \pi_k f_k(\boldsymbol{x}_t)}$$

  and only comparing $\pi_\ell f_\ell(\boldsymbol{x}_t)$ is enough

- This reduces to evaluate

$$\delta_\ell = -\frac{1}{2} \log |\boldsymbol{\Sigma}_\ell| - \frac{1}{2} (\boldsymbol{x}_t - \boldsymbol{\mu}_\ell)^\top \boldsymbol{\Sigma}_\ell^{-1} (\boldsymbol{x}_t - \boldsymbol{\mu}_\ell) + \log \pi_\ell$$

  and pick as the class the ℓ corresponding to the largest δ_ℓ

- [see the sample Matlab code]

Summary

- Logistic regression is very popular for classification, especially for binary classification
- LDA is especially useful when $K > 2$, the number of training samples is small, or the classes are well separated, and Gaussian assumptions are reasonable.
- QDA presents more flexibility in shaping the partitions compared to LDA
- Logistic regression can also fit quadratic boundaries like QDA, by explicitly including quadratic terms in the model

Week 4: Resampling/Bootstrap & Model Validation and Selection

AI 539: Machine Learning for Non-Majors

Alireza Aghasi

Oregon State University

Super easy to understand !

Bootstrap

$y = \beta_0 + \beta_1 x + \varepsilon$, $\text{var}(\varepsilon) = \sigma^2$, $(x_1, y_1) \sim (x_n, y_n)$

$\hat{\beta}_0 =$

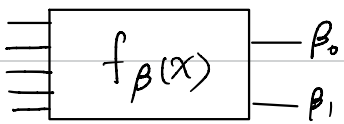
$$\hat{\beta}_1 = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sum (x_i - \bar{x})^2}$$

$E[\hat{\beta}_1] = \beta_1$ # , $\text{Var}(\hat{\beta}_1) = \frac{\sigma^2}{\sum (x_i - \bar{x})^2}$ #

What if $y = \beta_0 + \log|\beta_1 x|$

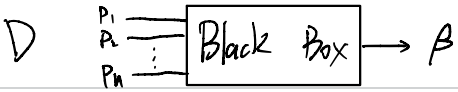
dataset

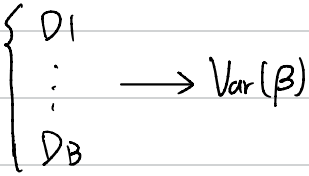
$D \{ (x_1, y_1), \vdots, (x_n, y_n)$

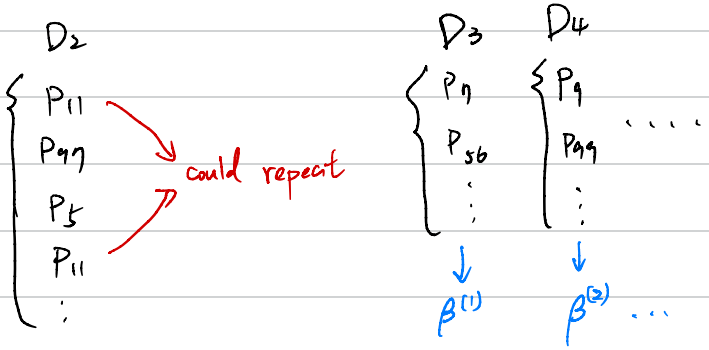


With limited data set, there won't be multiple β_0, β_1

So $\{ D_1 \rightarrow \text{Var}(\beta_1)$, D_2, $\vdots$, D_B







So $\beta^{(1)}, \beta^{(2)} \cdots \beta^{(B)} \rightarrow$

- average
- variance
- histogram
- Confidence Interval

Bootstrap

- The bootstrap is a flexible and very powerful statistical tool that can be used to quantify the uncertainty associated with a given estimator or statistical learning method
- It can provide an estimate of the standard error of a coefficient, or a **confidence interval for that coefficient**, regardless of how complex the derivation of that coefficient is

A Very Simple Example

- Think of a block, which takes 100 randomly generated samples drawn from a distribution and generates their sample mean $\bar{x}$ as the output
- We want to see if we do this many, many times, how the outputs of our block vary
- In principal to do this we would need to run the experiment, for say 1000 times, and then look at the variations of $\bar{x}$, however, this requires constant access to the distribution and a lot of sampling (which in many applications can be expensive to acquire)
- Bootstrap can help us do this without accessing the reference distribution
- Let's see the code

Bootstrap via an Example

- Lets explain bootstrap via an example
- Suppose that we wish to invest a fixed sum of money in two financial assets that yield returns of X and Y (random quantities)
- We will invest α shares in X, and will invest the remaining $1-\alpha$ in Y
- To minimize the risk, we want to minimize $var(\alpha X + (1-\alpha)Y)$
- We can show that (in the class we do it) that the minimizer is

$$\alpha = \frac{var(Y) - cov(X,Y)}{var(X) + var(Y) - 2cov(X,Y)}$$

$X, Y \in RV$, if independent $\Rightarrow Var(X+Y) = Var(X) + Var(Y)$

$X, Y \in RV$, not independent $\Rightarrow Var(X+Y) = Var(X) + Var(Y) + 2\,Cov(X,Y)$

$$V = Var(\alpha X + (1-\alpha) Y)$$

$$= \alpha^2 Var(X) + (1-\alpha)^2 Var(Y) + 2\,Cov(\alpha X, (1-\alpha) Y)$$

$$= \alpha^2 Var(X) + (1-\alpha)^2 Var(Y) + 2\alpha(1-\alpha)\,Cov(X,Y)$$

$$0 = \frac{dV}{d\alpha} = 2\alpha\,Var(X) + 2(1-\alpha)(-1)\,Var(Y) + (2-4\alpha)\,Cov(X,Y)$$

$$\rightarrow \alpha_{opt} = \frac{Var(Y) - Cov(X,Y)}{Var(X) + Var(Y) - 2\,Cov(X,Y)}$$

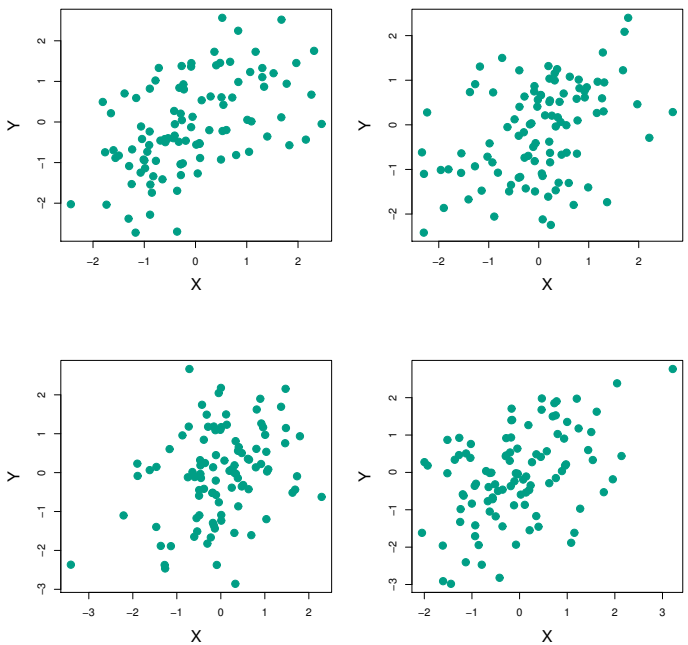
Bootstrap via an Example

- In real-world we do not know $var(X), var(Y), cov(X, Y)$
- Suppose we are given a data set containing pairs of X and Y. We can estimate $var(X), var(Y), cov(X, Y)$ from the sample set and get an estimate $\hat{\alpha}$ for the optimal share
- Ideally we can generate these sample sets many times, and estimate an $\hat{\alpha}$ for each and look into the histogram
- However in a real-world we only have one sample set to use
- Bootstrap yet allows us to generate good estimates of α **using only one sample set!**

Bootstrap via an Example

To see how nicely Bootstrap works, lets compare its outcome with the case that α is generated from many synthetic sample generations

- We generate 1000 sample sets each containing 100 pairs of X, Y
- For the synthetic data generated $var(X) = 1, var(Y) = 1.25$ and $cov(X, Y) = 0.5$ which yield an optimal value of $\alpha = 0.6$

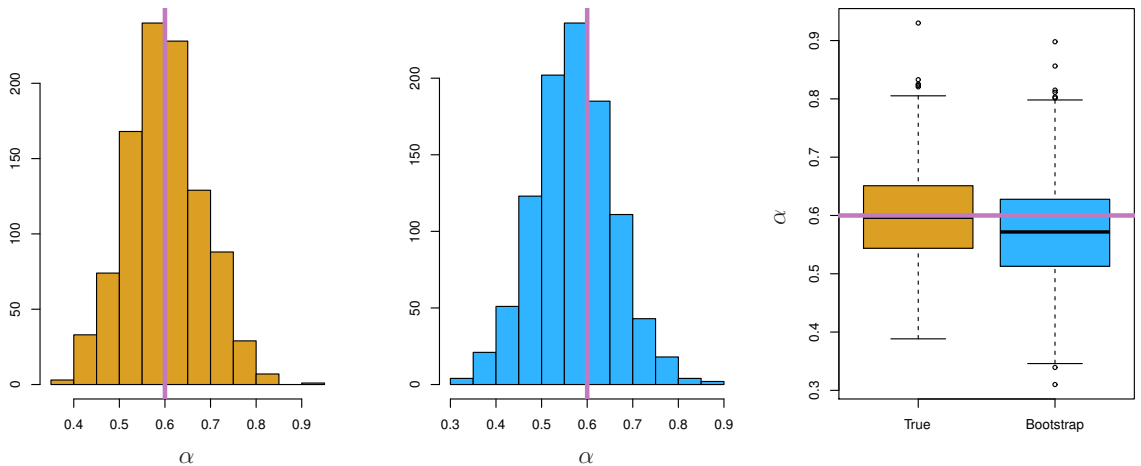


Bootstrap via an Example

- To get the left panel we generate 1000 synthetic sample sets, for each obtain $\hat{\alpha}$ and plot the histogram and calculate

$$\bar{\alpha} = \frac{1}{1000}\sum_{i=1}^{1000}\hat{\alpha}_i, \quad SE(\alpha) = \sqrt{\frac{1}{999}\sum_{i=1}^{1000}(\hat{\alpha}_i - \bar{\alpha})^2}$$

- For the bootstrap we only use one of the sample sets and regenerate new sample set by sampling with replacement
- Surprisingly, the results are very close



Bootstrap General Framework

- Suppose a black-box calculates $\hat{\alpha}$ from a sample set, e.g., a coefficient in linear or logistic regression
- We are interested in estimating the variability of $\hat{\alpha}$ without examining many new sample sets
- Denoting the first bootstrap data set by Z^1, we use Z^1 to produce a new bootstrap estimate for α, which we call $\hat{\alpha}^1$
- This procedure is repeated B (say 100 or 1000) times, in order to produce B different bootstrap data sets, $Z^1, Z^2, \cdots, Z^B$, and the corresponding α estimates, $\hat{\alpha}^1, \cdots, \hat{\alpha}^B$
- We estimate the standard error of these bootstrap estimates using the formula

$$SE_B(\alpha) = \sqrt{\frac{1}{B-1}\sum_{i=1}^{B}(\hat{\alpha}^i - \bar{\hat{\alpha}})^2} \quad \text{where} \quad \bar{\hat{\alpha}} = \frac{1}{B}\sum_{i=1}^{B}\hat{\alpha}^i$$

- This serves as an estimate of the standard error of α estimated from the original data set!

Programming Exercise

Let's go through some programming exercises

- Basic Example
- Linear model example

(x_1, y_1) $\qquad y = \beta_0 + \beta_1 x$

$\vdots$

(x_n, y_n) $\qquad y_i = \beta_0 + \beta_1 x_i + \varepsilon_i$

$$\rightarrow \varepsilon_i = y_i - \beta_0 - \beta_1 x_i \begin{cases} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_{100} \end{cases}$$

$x_1, y_1^{(1)}$	$x_1, y_1^{(2)}$
$\vdots$	$\vdots$
x_{100}, y_{100}	x_{100}, y_{100}

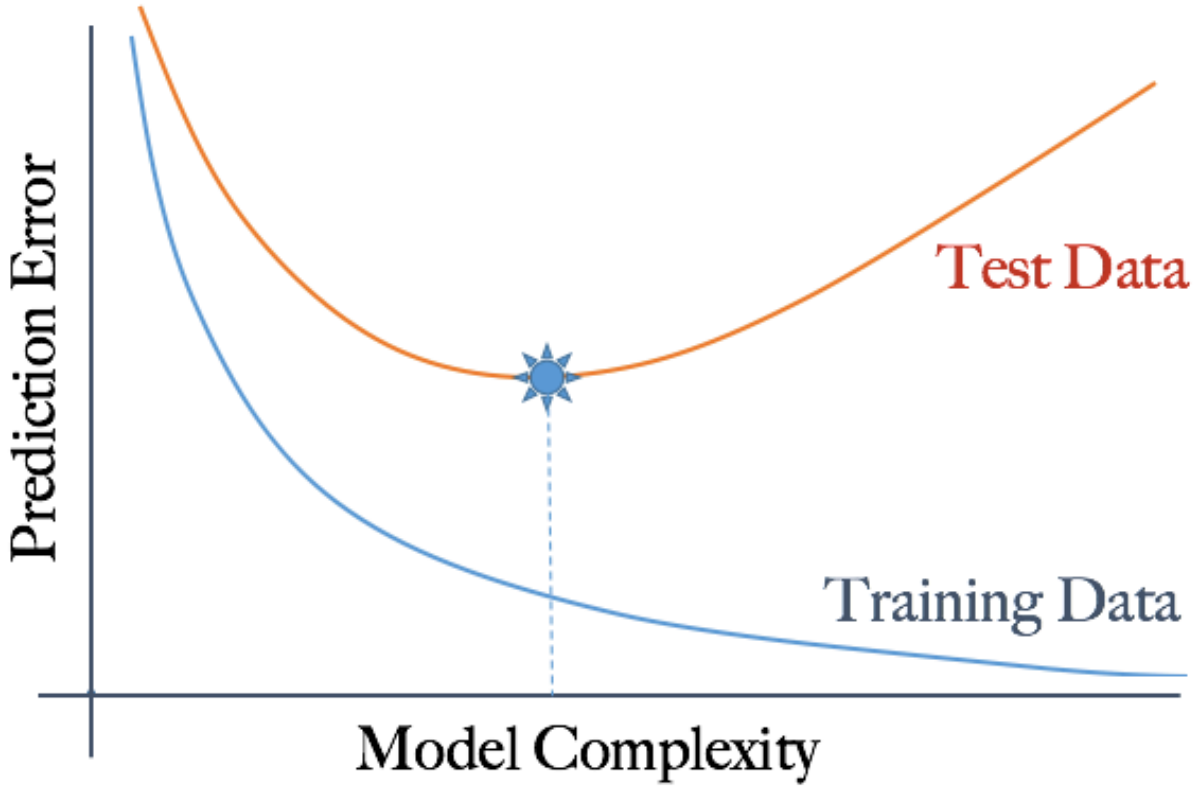
Model Validation and Selection

Introduction

- Recall that we fitted out models using training data and were interested in evaluating the performance with respect to independent test data
- To produce justifiable model reliability arguments, the test data should not be used in the training
- If the model is evaluated against the training data the results can be very distracting
- The training error rate is often quite different from the test error rate, and in particular the former can dramatically underestimate the latter (recall the accuracy vs complexity chart)

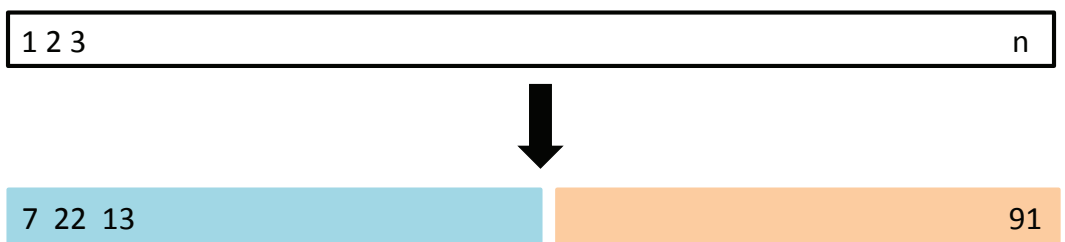
Training vs Test Model Evaluation

Recall this plot from the first session



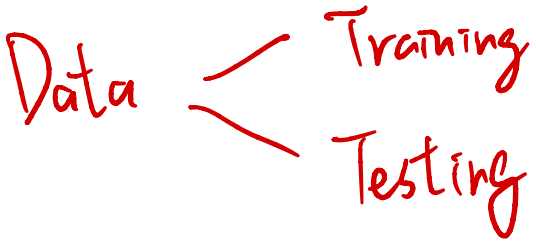
Real-World Data Issues and Test Performance

- A good evaluation is possible when a large test set is available
- Often such set is not available
- We are interested in a class of methods that estimate the test error by holding out a subset of the training observations from the fitting process, and then applying the statistical learning method to those (held out) observations



Validation Set Approach

Approach 1



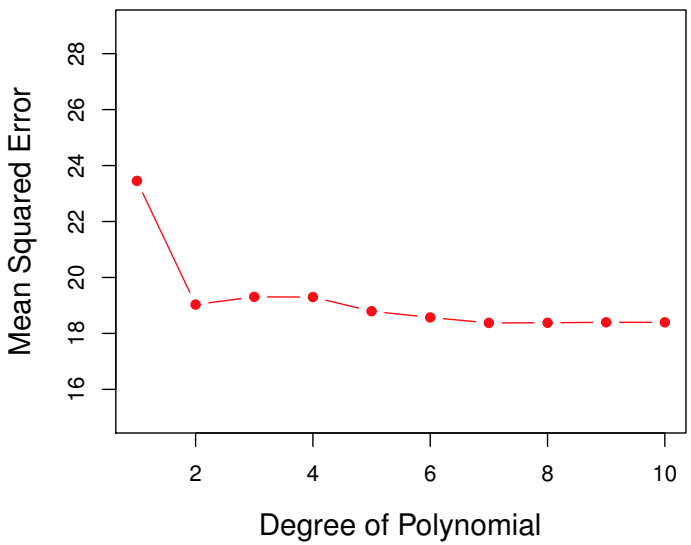
- This is the standard approach we have been using so far
- We randomly divide the available set of samples into two parts: a training set and a validation or hold-out set (sometimes 50%-50% splitting, often 80%-20% splitting)
- The model is fit on the training set, and the fitted model is used to predict the responses for the observations in the validation set
- The error with reference to the hold-out set is an approximation of the test error

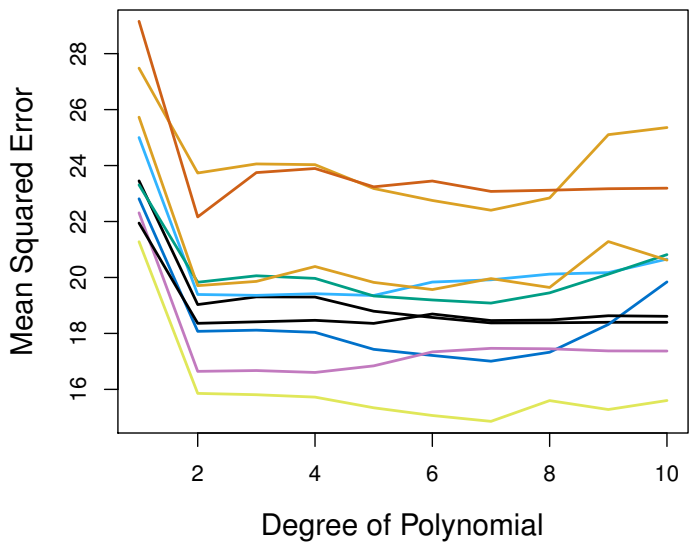
example

- Recall the automobile data: Regressing Mile per Gallon in terms of the Horse Power

$$\texttt{mpg} = \beta_0 + \sum_{i=1}^{p} \beta_i \texttt{(horsepower)}^i$$

- We randomly split the 392 observations into two sets, a training set containing 196 of the data points, and a validation set containing the remaining 196 observations.





Validation Set Cons & Pros

Pro – The procedure is **simple to do** (as we have done so far) and only a subset of the observations — those that are included in the training set rather than in the validation set — are used to fit the model

Con – The validation estimate of the test error can be highly variable, depending on precisely which observations are included in the training set and which observations are included in the validation set

Con – While the estimated test error vary a lot, **finding information such as model selection is still possible**

Con – Since a large portion of the data need to be held aside, the model fits are not accurate enough

Leave-One Out Cross-Validation (LOOCV) Approach 2

- We have n data points $(x_1, y_1), \cdots, (x_n, y_n)$, we use $n-1$ for the training and one instance for the test
- Of course a single test point is no where close to the true test error, but this process is repeated n times, every time $n-1$ points used for training and one point left out for the test
- Considering $MSE_1 = (y_1 - \hat{y}_1)^2, \cdots MSE_n = (y_n - \hat{y}_n)^2$, an approximation of the test error is

$$CV_n = \frac{1}{n} \sum_{i=1}^{n} MSE_i$$

Only one sample for testing each time

Cons & Pros with LOOCV

Pro – It has a very small bias compared to the validation set approach (it almost uses as much data as possible to fit the model)

Pro – The test error overestimation is less than the validation set approach (because of what we mentioned above)

Pro – Its results are reproducible unlike the validation set approach which uses a random subset of the data for test evaluation

Con – It can be computationally very expensive (requires running the algorithm n times)

– For linear models there is a shortcut to calculate CV_n that only requires fitting the model once with the entire data (but **this shortcut only applies to linear models**)

$$CV_n = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{y_i - \hat{y}_i}{1 - h_i}\right)^2$$

where $\hat{y}_i$ are the fitted values of the original least squares problem and h_i are only data dependent

K-Fold Cross Validation

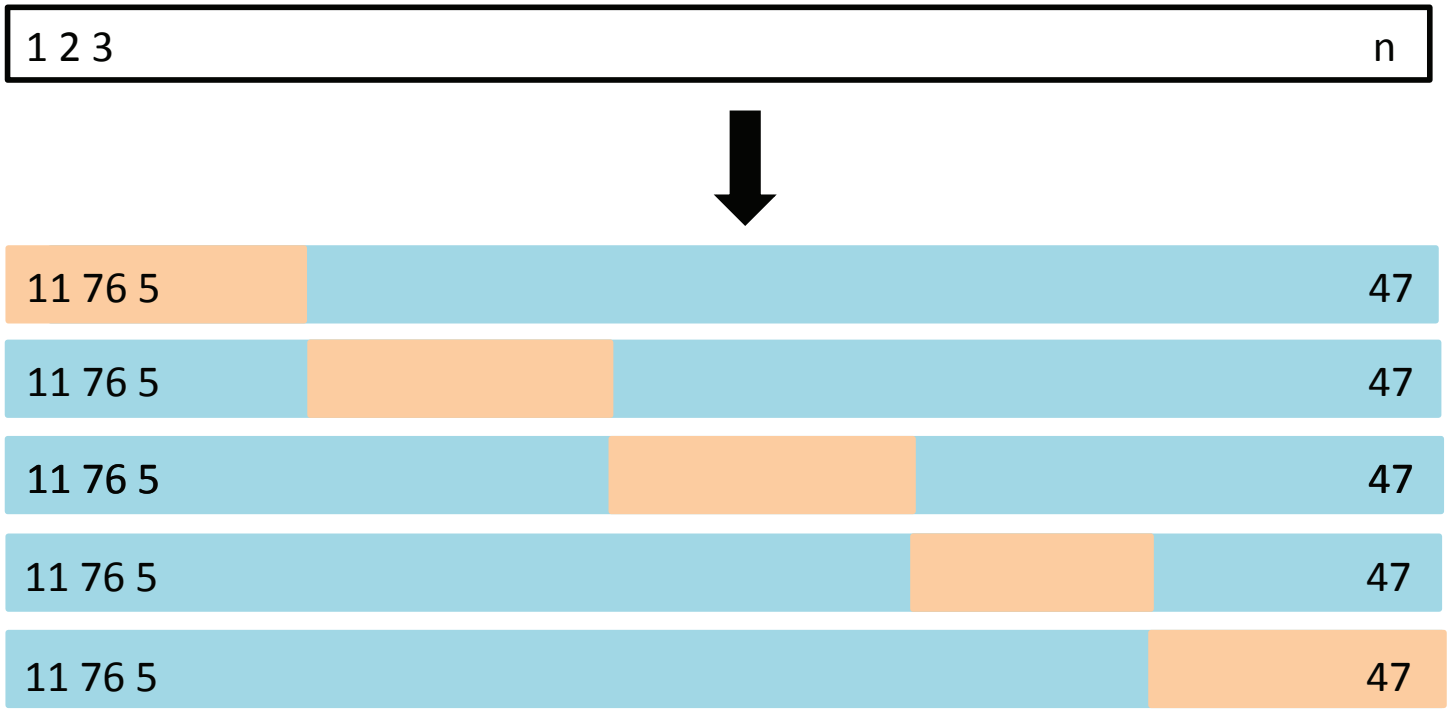
Approach 3

- Widely used approach for estimating test error
- This approach involves **randomly dividing the set of observations into K groups**, or folds, of approximately equal size. The first fold is treated as a validation set, and the method is fit on the remaining $K - 1$ folds
- This is done in turn for each part $k = 1, 2, \cdots, K$, and then the results are combined
- Estimates can be used to select best model, and to give an idea of the test error of the final chosen model

$$CV_K = \frac{1}{K} \sum_{k=1}^{K} MSE_k$$

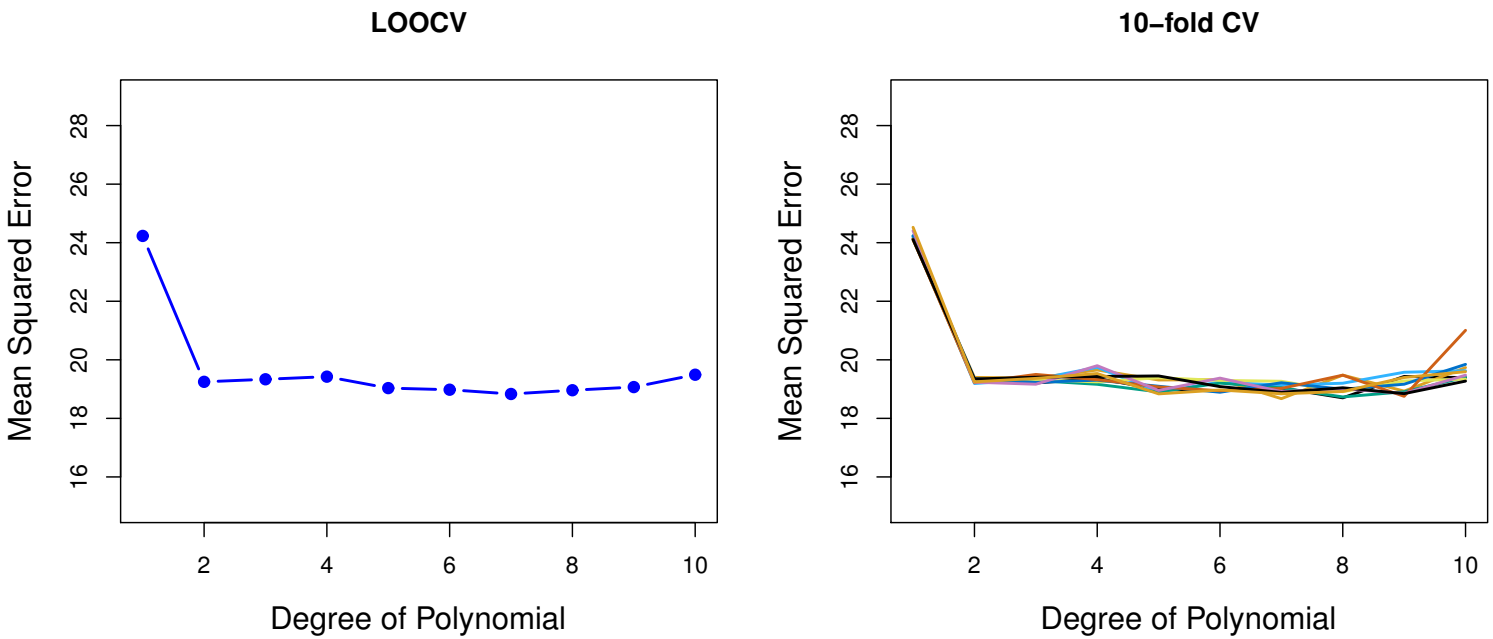
- Often $K = 5$ or $K = 10$ is what is considered in application

K-Fold Cross Validation



K-Fold Cross Validation and LOOCV

- LOOCV is a special case of K-fold CV for $K = n$
- In general K-fold CV is much cheaper than LOOCV because it only requires K model fits vs n model fits
- For model selection, K-fold CV often gives us similar outcomes at a much lower computational cost



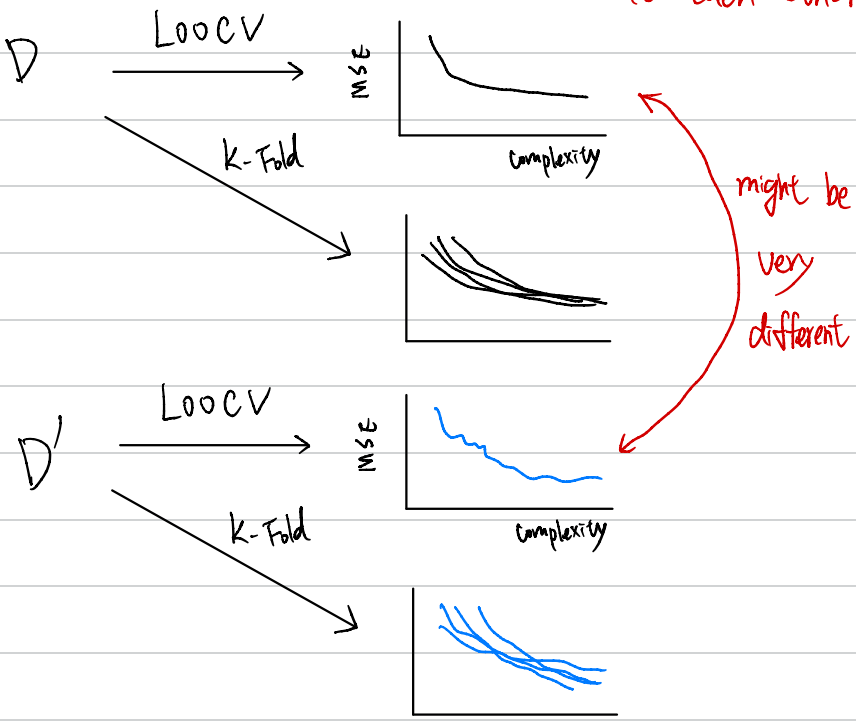
K-Fold Cross Validation and LOOCV

- Aside form the computational issues, even surprisingly K-Fold CV produces better test estimates than the LOOCV
- LOOCV has a lower bias compared to the K-fold CV, since it uses more data to fit the model
- But K-fold CV has a lower variance compared to the LOOCV, since LOOCV is the sum of n highly correlated random variables while the correlation between the MSEs in K-fold CV is lower, recall

$$var(X + Y) = var(X) + var(Y) + 2cov(X, Y)$$

Better than LOOCV in terms of Variance

Since the training sets in LOOCV are highly correlated to each other.



i.g.

$X, Y \in RV$, if independent $\Rightarrow Var(X+Y) = Var(X) + Var(Y)$

$X, Y \in RV$, not independent $\Rightarrow Var(X+Y) = Var(X) + Var(Y) + 2\,Cov(X, Y)$

Suppose $X = Y \Rightarrow Var(X, Y) = Var(2X) = 4\,Var(X)$

CV & Classification

- We divide the data into K roughly equal-sized index sets $C_1, \cdots, C_K$
- Compute

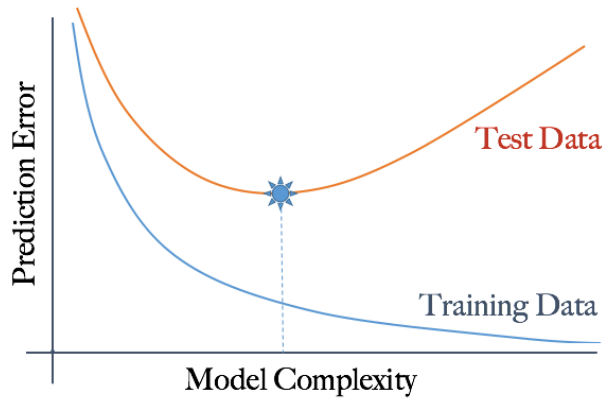
$$CV_K = \frac{1}{K} \sum_{k=1}^{K} Err_k$$

where

$$Err_k = \frac{1}{\#\text{ elements in } C_k} \sum_{i \in C_k} I(y_i \neq \hat{y}_i)$$

Cross Validation Summary

- As mentioned earlier, model selection based on the RSS or R^2 statistics can be misleading, since the training error is not a good representative of the actual test error

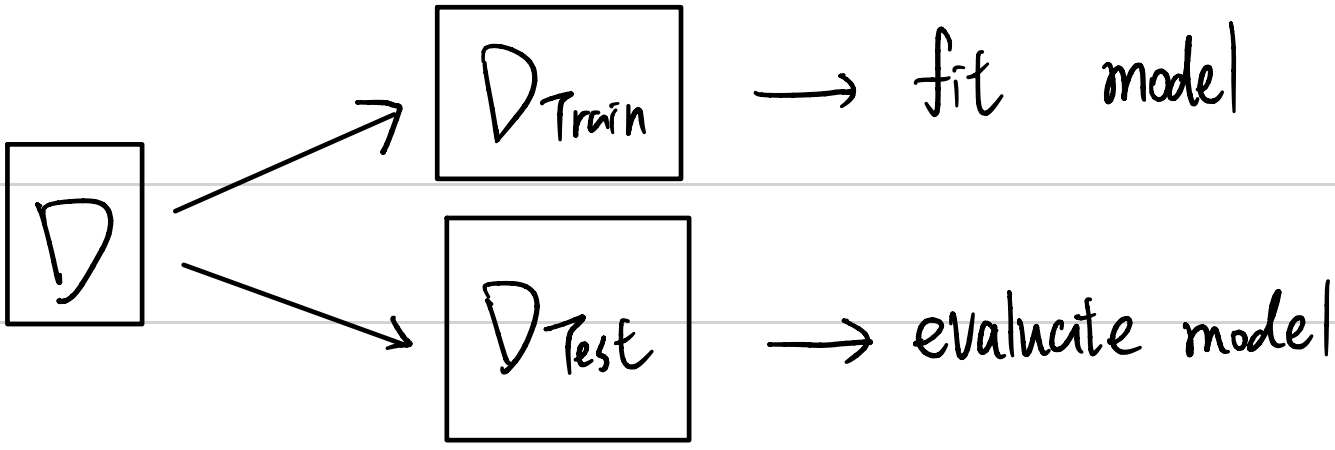


- Instead through a process of splitting the data into training and validations sets, we were able to use LOOCV or K-Fold CV as estimates of the test error
- We discussed why K-Fold CV is a more desirable estimate, computationally and statistically

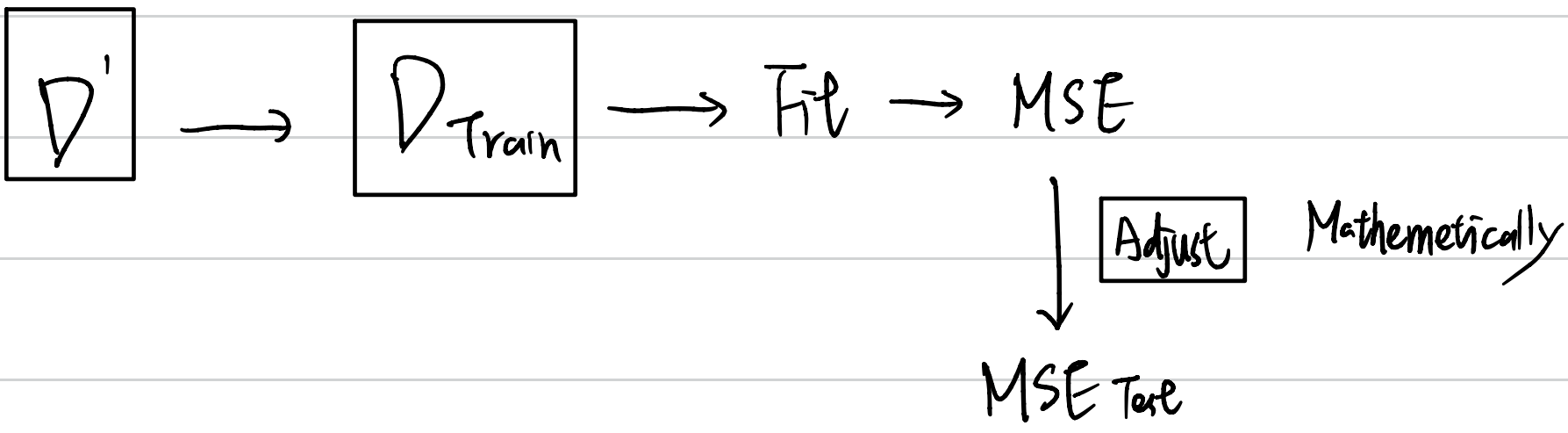
Adjusting the Training Statistics for Test Error Approximation

Adjusting Techniques

- We introduce few other ways of adjusting the training error **to make it a better representative of the test error**
- These adjustments are **not as reliable as Cross validation**, but they are easier to **calculate**
- These quantities were **more widely used before** the widespread use of computers for regression and machine learning
- Now that computers can help performing multiple fits computationally fast enough, often K-Fold CV is considered as the desirable test error approximation
- The test error estimates that we will present in the next couple of slides are **only valid for least-squares models (i.e., linear regression)**, and do not extend to all parametric models



Restricted Classification Model



List of Other Techniques

Only apply to Linear Regression Model

Methods to adjust the training error for the number of variables to estimate the test MSE:

- C_p statistic
- Akaike information criterion (AIC)
- Bayesian information criterion (BIC)
- Adjusted R^2

C_p Statistic

- For a fitted least squares model with d predictors

$$C_p = \frac{1}{n}(RSS + 2d\hat{\sigma}^2)$$

- $\hat{\sigma}^2$ is an estimate of the noise variance
- $\hat{\sigma}^2$ is normally estimated using all the predictors (full model)
- It is an unbiased estimate of the **test MSE**
- The smaller C_p, the better the model (we can pick models with the smallest C_p statistic)
- Becomes a better estimate of the test error as the sample size, n, increases

AIC: Akaike Information Criterion

- Defined for a large class of models based on the maximum likelihood criterion
- When we consider the noise ϵ be of i.i.d Gaussian, the MLE and MSE return identical results and in this case we have

$$AIC = \frac{1}{n\hat{\sigma}^2}(RSS + 2d\hat{\sigma}^2)$$

  which is a multiple of C_p (no preference over using one vs the other)
- $\hat{\sigma}^2$ is an estimate of the noise variance
- $\hat{\sigma}^2$ is normally estimated using all the predictors (full model)
- The smaller AIC, the better the model (we can pick models with the smallest AIC statistic)

BIC: Bayesian Information Criterion

- Takes a Bayesian approach to estimate the test error
- Asymptotically ($n \to \infty$) choosing the model with the highest posterior probability of being the best model
- In the case of least squares the formulation is

$$BIC = \frac{1}{n\hat{\sigma}^2}(RSS + \log(n)d\hat{\sigma}^2)$$

  which takes an *almost* similar form as the previous two statistics
- $\hat{\sigma}^2$ is an estimate of the noise variance
- $\hat{\sigma}^2$ is normally estimated using all the predictors (full model)
- The smaller BIC, the better the model (we can pick models with the smallest AIC statistic)
- When $n < 7$, BIC imposes a smaller penalty on the number of variables, but for $n > 7$ that $\log n > 2$ the penalty is larger
- In other words in standard observation regimes where n is sufficiently large, BIC tends to pick smaller models than AIC or C_p

Adjusted R^2

- Presents a way of making the R^2 statistic dependent on the number of predictors
- Recall the R^2 statistic:

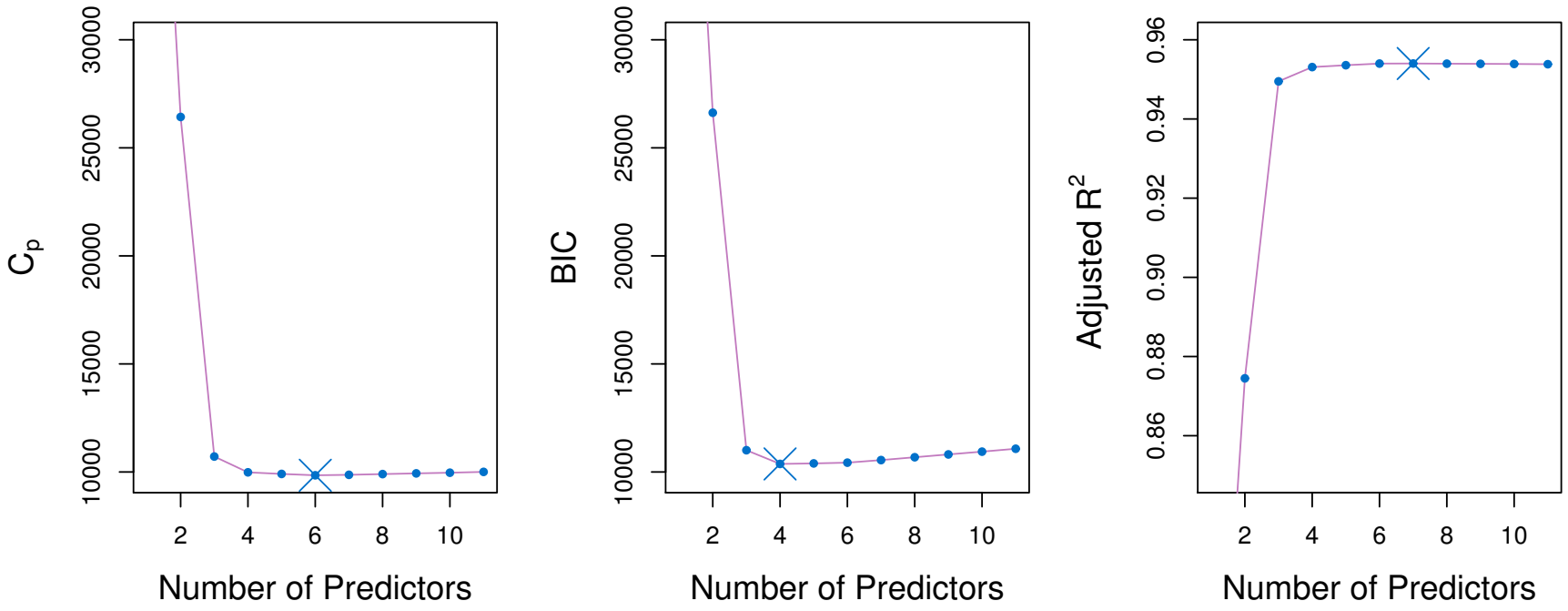
$$R^2 = 1 - \frac{RSS}{TSS}, \qquad \text{where} \quad TSS = \sum_{i=1}^{n}(y_i - \bar{y})^2$$

- The formulation for adjusted R^2 is

$$R^2_{adj} = 1 - \frac{RSS/(n-d-1)}{TSS/(n-1)}$$

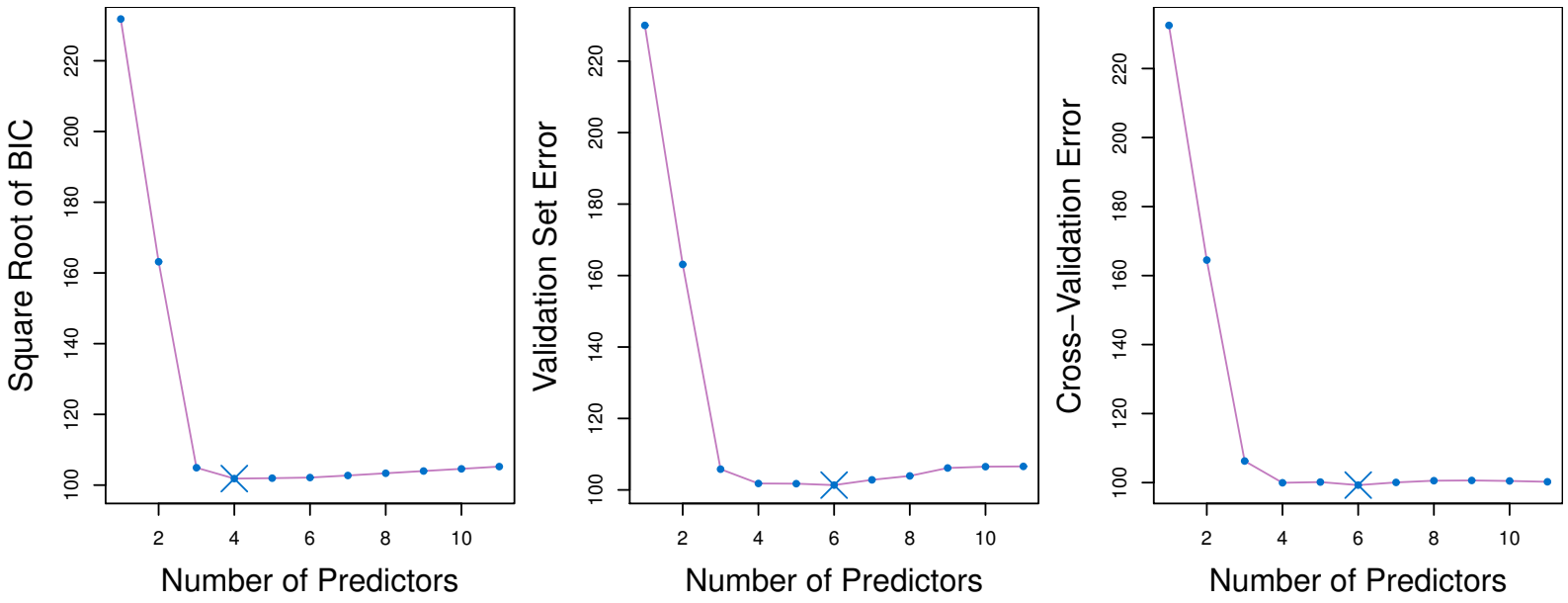
- Unlike the other three statistics that being small indicates a better model, for adjusted R^2 we are interested in models that tend to generate values closer to 1
- The use of C_p, AIC and BIC is more motivated in statistical learning theory than the adjusted R^2

Example Comparing the Performances



C_p, BIC, and adjusted R^2 for the best models of each size for the Credit data set

Comparison Against CV Techniques



- The results are not much different
- Note that nowadays CV methods are computationally fast to implement and regardless of the model can always be used as a reliable selection tool

How to Use These Statistics in Model Selection

– **Best subset selection** formal procedure (NP-hard and computationally not possible for large p)

Algorithm 6.1 *Best subset selection*

---

1. Let $\mathcal{M}_0$ denote the *null model*, which contains no predictors. This model simply predicts the sample mean for each observation.

2. For $k = 1, 2, \ldots p$:

   (a) Fit all $\binom{p}{k}$ models that contain exactly k predictors.

   (b) Pick the best among these $\binom{p}{k}$ models, and call it $\mathcal{M}_k$. Here *best* is defined as having the smallest RSS, or equivalently largest R^2.

3. Select a single best model from among $\mathcal{M}_0, \ldots, \mathcal{M}_p$ using cross-validated prediction error, C_p (AIC), BIC, or adjusted R^2.

$$\{x_1, x_2 \ldots, x_p\}$$

$$\#\text{ of subset} = 2^p$$

$$\sum_{k=0}^{p} \binom{p}{k} = 2^p$$

$M_0 \leftarrow$ Size 0 : no feature : β_0

$M_1 \leftarrow$ size 1 : $\{x_1\}, \{x_2\} \ldots \{x_p\} \rightarrow p$

Best Models

$M_2 \leftarrow$ size 2 : $\{x_1, x_2\}, \{x_1, x_3\}, \ldots \rightarrow \binom{p}{2}$

$\vdots$

M_p

How to Use These Statistics in Model Selection

- **Forward stepwise selection** (computationally tractable)
- At each step the variable that gives the greatest additional improvement to the fit is added to the model

Champions stay

Algorithm 6.2 *Forward stepwise selection*

1. Let $\mathcal{M}_0$ denote the *null* model, which contains no predictors.
2. For $k = 0, \ldots, p-1$:
   (a) Consider all $p-k$ models that augment the predictors in $\mathcal{M}_k$ with one additional predictor.
   (b) Choose the *best* among these $p-k$ models, and call it $\mathcal{M}_{k+1}$. Here *best* is defined as having smallest RSS or highest R^2.
3. Select a single best model from among $\mathcal{M}_0, \ldots, \mathcal{M}_p$ using cross-validated prediction error, C_p (AIC), BIC, or adjusted R^2.

- Forward selection can even be used when $n < p$

$\{X_1\}, \{X_2\} \ldots \{X_p\}$

$\{X_{10}\}$

keep

$\{X_{10}, X_1\}, \{X_{10}, X_2\} \cdots \{X_{10}, X_p\}$

$\{X_{10}, X_8\}$

keep

$\{X_{10}, X_8, X_1\}$ $\{X_{10}, X_8, X_2\} \cdots \{X_{10}, X_8, X_p\}$

$\{X_{10}, X_8, X_{20}\}$

$\vdots$

How to Use These Statistics in Model Selection

3

- **Backward stepwise selection** (computationally tractable)
- Begins with the full least squares model containing all p predictors, and then iteratively removes the least useful predictor, one-at-a-time

---

Algorithm 6.3 *Backward stepwise selection*

---

1. Let $\mathcal{M}_p$ denote the *full* model, which contains all p predictors.
2. For $k = p, p-1, \ldots, 1$:
   (a) Consider all k models that contain all but one of the predictors in $\mathcal{M}_k$, for a total of $k-1$ predictors.
   (b) Choose the *best* among these k models, and call it $\mathcal{M}_{k-1}$. Here *best* is defined as having smallest RSS or highest R^2.
3. Select a single best model from among $\mathcal{M}_0, \ldots, \mathcal{M}_p$ using cross-validated prediction error, C_p (AIC), BIC, or adjusted R^2.

- Backward selection requires $p < n$ (to allow the full model to be fit)