

Statistical Learning

Introduction to Statistics

Outline

- 1. Why Statistics**
- 2. Statistical Methods**
- 3. Types of Statistics - Descriptive and Inferential Statistics**
- 4. Data Sources and Types of Datasets**
- 5. Attributes of Datasets**

Why Statistics is So Important?

Three significant events triggered the current meteoric growth in the use of analytical decision making and *Statistics is central to all of them.*

Event1

- Technological developments, Revolution of Internet and social networks, data generated from mobile phones and other electronic devices, produce large amount of data from which insights will have to be sifted.
- The discovery of pattern and trends from these data for organizations will pave the way for improving profitability, understanding customer expectations, and appropriately pricing their products so that they can gain competitive advantage in the marketplace.

Why Statistics is So Important?

Event 2

- Advances in enormous computing power to effectively process and analyze massive amounts of data
- Sophisticated and faster algorithms for solving problems
- Data Visualization for Business Intelligence and Artificial Intelligence

Why Statistics is So Important?

Event 3

- Large data storage capability
- Parallel computing, and cloud computing coupled with better computer hardware have enabled businesses and other organizations to solve large scale problems faster than ever before without sacrificing

Big Data

Big data

- A set of data that cannot be managed, processed, or analyzed with traditional software/algorithms within a reasonable amount of time.
- Big data revolves around
Volume Velocity Variety Value Veracity

Walmart handles over one million purchase transactions per hour.

Facebook processes more than 250 million picture uploads per day.

Statistics - Methods

Classification

- *Classification* techniques helps in segmenting the customers into appropriate groups based on key characteristics.
- For example, using *appropriate statistical model*, an organization could easily segment the customers into Long Term Customers, Medium Term Customers, and Brand Switchers.
- Another application in this context is classifying customers into “Buyers and Non-Buyers.”
- Classification helps professionals understand the customer behavior and position their products and brands using appropriate strategies.

Statistics - Methods

Pattern Recognition

- “A picture is worth thousand words” and it reveals hidden pattern in the data that could be leveraged by retail professionals. Pattern recognition techniques include *Histogram*, *Box Plot*, *Scatter Plot* and other *Visual Analytics*.
- For example, histogram drawn for income of a particular class of customers may reveal a symmetrical bell curve pattern or may be left or right skewed.
- Relationship between age and expenditure could be captured using a scatter plot.
- *Box Plot* enables identification of outliers (extreme points) apart from providing the distribution pattern.

Statistics - Methods

Association

- *Association Analysis* helps in determining which of the items go together. Association rules include a set of analytics that focuses on discovering relationships that exist among specific objects.
- In this context, market basket analysis refers to an association rule that generates the probability for an outcome.
- For example, market basket analysis may lead to a finding that if customers buy coffee, there is a 40% probability that they also buy bread.
- Association rules can be adapted by organizations to store lay cross-selling among others, discount and sales promotion decisions, and

Statistics - Methods

Predictive Modeling

- Both customer segmentation as well as identifying and targeting most profitable customers can be facilitated by predictive models.
- *Regression* can be used for predicting the amount of expenditure on a particular product based on input variables income, age, and gender.
- Organizations can leverage on other advanced models that comprise *Logistic Regression*, and *Neural Networks* for predicting a target variable as well as classifying and predicting into which group the consumer belongs to.
- For example, these models can classify and predict buyers and non-buyers, and defaulters and non-defaulters on credit card loan.

Classical Definition of Statistics

“ By Statistics, we mean methods specially adopted to the elucidation of quantitative data affected to a marked extent by multiplicity of causes”.

Yule and Kendal

It is interesting to see what *Thomas Davenport* means by Business Analytics and note the similarities and dissimilarities between the two.

“Business Analytics (BA) can be defined as the broad use of data and quantitative analysis for decision making within organizations”.

Types of Statistics

Descriptive Statistics
is concerned with Data
Summarization,
Graphs/Charts, and
Tables

Inferential Statistics is a
method used to talk
about a Population
Parameter from a Sample.

Population, Parameter, Sample, Statistic

A Population is the universe of possible data for a specified object. Example: People who have visited or will visit a website.

A Parameter is a numerical value associated with a population. Example: The average amount of time people spend on a website.

A Sample is a selection of observations from a population. Example: People (or IP addresses) who visited a website on a specific day.

A Statistic is a numerical value associated with an observed sample. Example: The average amount of time people spent on a website on a specific day.

Data Sources

Primary Data are collected by the organization itself for a particular purpose. The benefits of primary data are that they fit the needs exactly, are up to date, and reliable.

Secondary Data are collected by other organizations or for other purposes. Any data, which are not collected by the organization for the specified purpose, are secondary data. These may be published by other organizations, available from research studies, published by the government, web, social media and so on.

Types of Data

Qualitative Data are nonnumeric in nature and can't be measured. Examples are gender, religion, and place of birth.

Quantitative Data are numerical in nature and can be measured. Examples are balance in your savings bank account, and number of members in your family.

Quantitative data can be classified into discrete type or continuous type. **Discrete type** can take only certain values, and there are discontinuities between values, such as the number of rooms in a hotel, which cannot be in fraction. **Continuous type** can take any value within a specific interval, such as the production quantity of a particular type of paper (measured in kilograms).

Types of Data Sets

- **Record**
 - Relational records
 - Data matrix, e.g., numerical matrix, crosstabs
 - Document data: text documents: term-frequency vector
 - Transaction data
- **Graph and network**
 - World Wide Web
 - Social or information networks
 - Molecular Structures
- **Ordered**
 - Video data: sequence of images
 - Temporal data: time-series
 - Sequential Data: transaction sequences
 - Genetic sequence data
- **Spatial, image and multimedia:**
 - Spatial data: maps
 - Image data
 - Video data

	team	coach	pla y	ball	score	game	wi n	lost	timeout	season
Document 1	3	0	5	0	2	6	0	2	0	2
Document 2	0	7	0	2	1	0	0	3	0	0
Document 3	0	1	0	0	1	2	2	0	3	0

TID	Items
1	Bread, Coke, Milk
2	Beer, Bread
3	Beer, Coke, Diaper, Milk
4	Beer, Bread, Diaper, Milk
5	Coke, Diaper, Milk

Data Objects

- Data sets are made up of data objects.
- A data object represents an entity.
- Examples:
 - sales database: customers, store items, sales
 - medical database: patients, treatments
 - university database: students, professors, courses
- Also called *samples*, *examples*, *instances*, *data points*, *objects*, *tuples*.
- Data objects are described by attributes.
- Database rows -> data objects; columns -> attributes.

Attributes

- Attribute (or dimensions, features, variables): a data field, representing a characteristic or feature of a data object.
 - *E.g., customer_ID, name, address*
- Types:
 - Nominal
 - Binary
 - Ordinal
 - Numeric: quantitative
 - Interval-scaled
 - Ratio-scaled

Attribute Types

- Nominal: categories, states, or “names of things”
 - *Hair_color* = {*auburn, black, blond, brown, grey, red, white*}
 - marital status, occupation, ID numbers, zip codes
- Binary
 - Nominal attribute with only 2 states (0 and 1)
 - Symmetric binary: both outcomes equally important
 - e.g., gender
 - Asymmetric binary: outcomes not equally important.
 - e.g., medical test (positive vs. negative)
 - Convention: assign 1 to most important outcome (e.g., HIV positive)
- Ordinal
 - Values have a meaningful order (ranking) but magnitude between successive values is not known.
 - *Size* = {*small, medium, large*}, grades, army rankings

Numeric Attribute Types

- Quantity (integer or real-valued)
- Interval
 - Measured on a scale of equal-sized units
 - Values have order
 - E.g., *temperature in C° or F°, calendar dates*
 - No true zero-point
- Ratio
 - Inherent zero-point
 - We can speak of values as being an order of magnitude larger than the unit of measurement (10 K° is twice as high as 5 K°).
 - e.g., *temperature in Kelvin, length, counts, monetary quantities*

Statistical Learning

Measures of central tendency, dispersion, and correlation

Outline

- 1. Raw Data**
- 2. Frequency Distribution - Histograms**
- 3. Cumulative Frequency Distribution**
- 4. Measures of Central Tendency**
- 5. Mean, Median, Mode**
- 6. Measures of Dispersion**
- 7. Range, IQR, Standard Deviation, coefficient of variation**
- 8. Normal distribution, Chebyshev Rule.**
- 9. Five number summary, boxplots, QQ plots, Quantile plot, scatter plot.**
- 10. Visualization: scatter plot matrix.**
- 11. Correlation analysis**

Data versus Information

When analysts are bewildered by plethora of data, which do not make any sense on the surface of it, they are looking for methods to classify data that would convey meaning. The idea here is to help them draw the right conclusion. Data needs to be arranged into information.

Raw Data

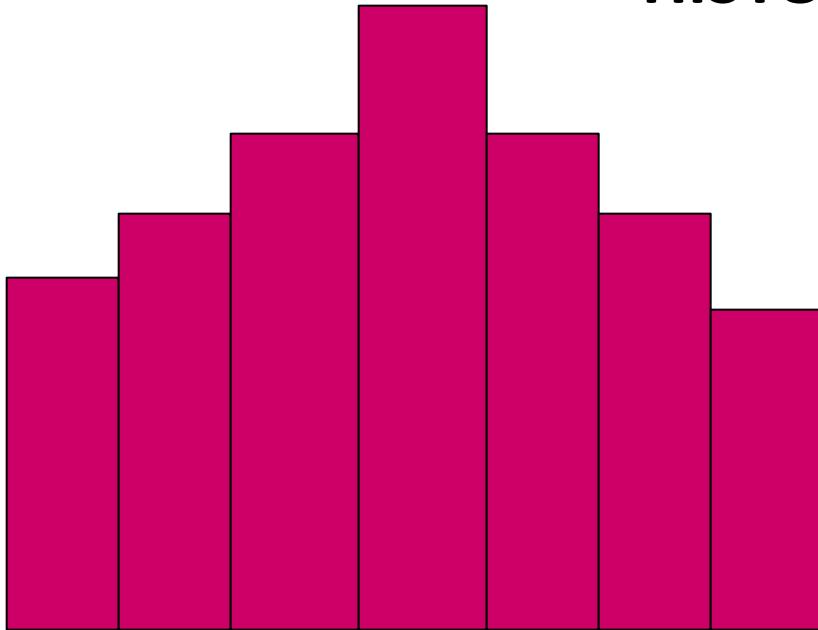
Raw Data represent numbers and facts in the original format in which the data have been collected. We need to convert the raw data into information for decision making.

Frequency Distribution

In simple terms, frequency distribution is a summarized table in which raw data are arranged into classes and frequencies.

Frequency distribution focuses on classifying raw data into information. It is a widely used data reduction technique in descriptive statistics.

HISTOGRAM



Histogram (also known as frequency histogram) is a snap shot of the frequency distribution.

Histogram is a graphical representation of the frequency distribution in which the X-axis represents the classes and the Y-axis represents the frequencies in bars

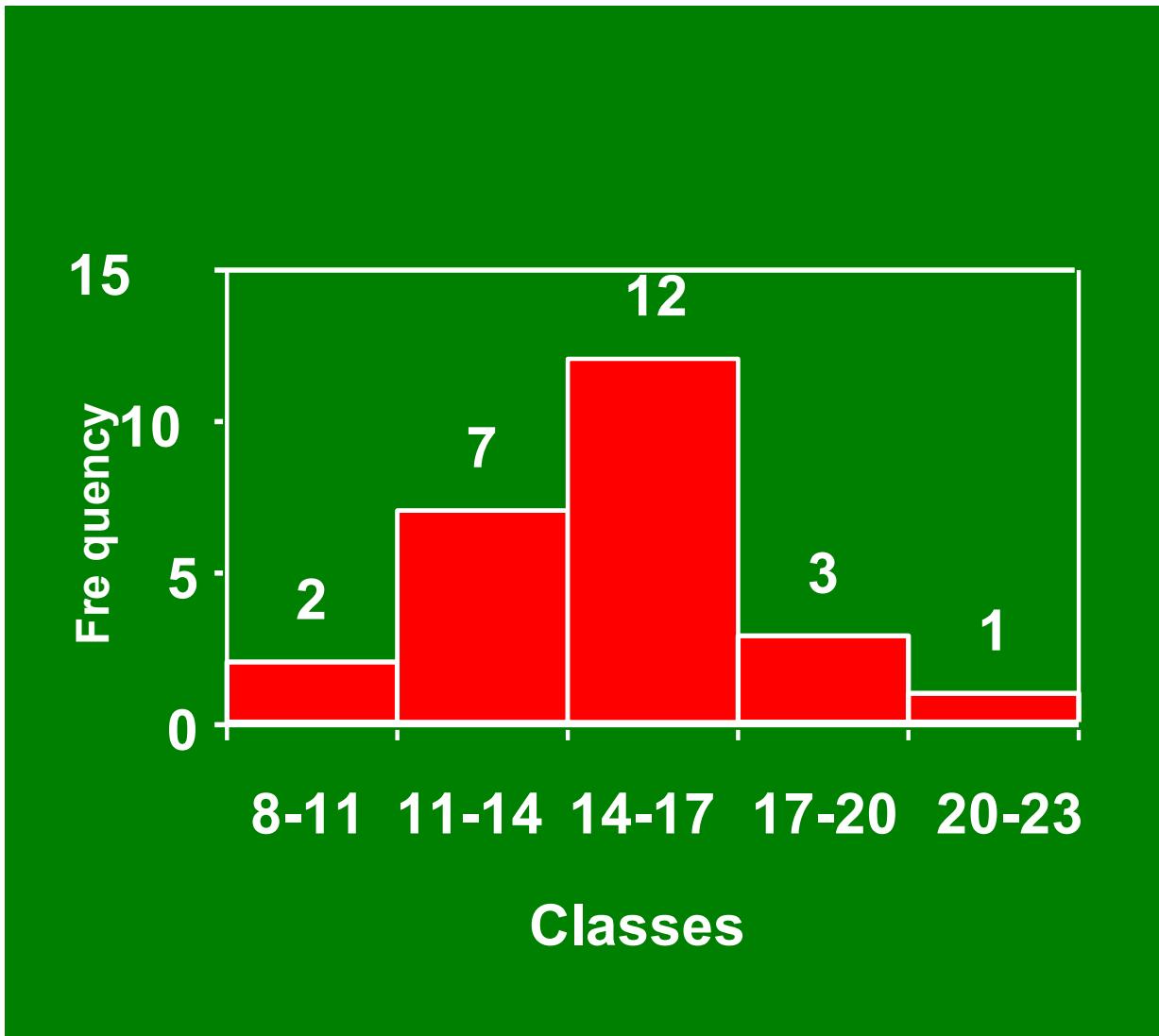
Histogram depicts the pattern of the distribution emerging from the characteristic being measured.

Histogram- Example

The inspection records of a hose assembly operation revealed a high level of rejection. An analysis of the records showed that the "leaks" were a major contributing factor to the problem. It was decided to investigate the hose clamping operation. The hose clamping force (torque) was measured on twenty five assemblies. (Figures in foot-pounds). The data are given below:
Draw the frequency histogram and comment.

8	13	15	10	16
11	14	11	14	20
15	16	12	15	13
12	13	16	17	17
14	14	14	18	15

Histogram Example Solution



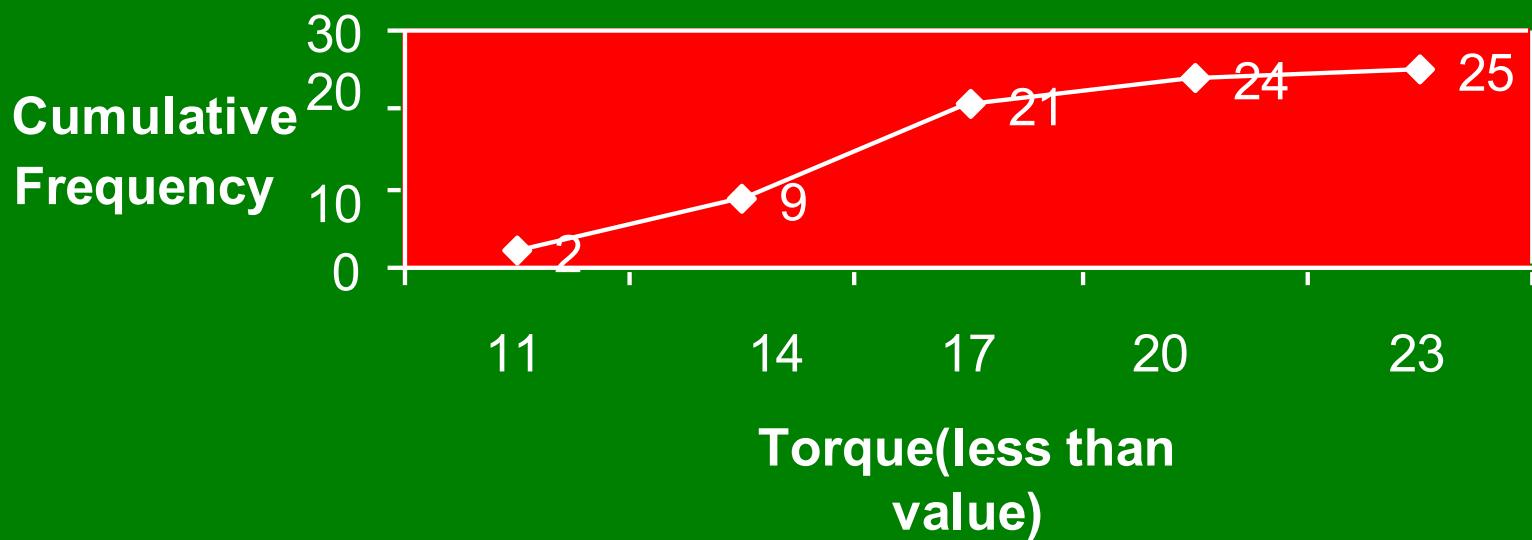
Cumulative Frequency Distribution

A type of frequency distribution that shows how many observations are above or below the lower boundaries of the classes. You can formulate the following from the previous example of hose clamping force(torque)

Class	Frequency	Relative Frequency	Cumulative Frequency	Cumulative Relative Frequency
8-11	2	0.08	2	0.08
11-14	7	0.28	9	0.36
14-17	12	0.48	21	0.84
17-20	3	0.12	24	0.96
20-23	1	0.04	25	1.00
Total	25	1.00		

Cumulative Distribution Function

**Cumulative Distribution (Ogive Curve)
for the Example**



What is Central Tendency?

Whenever you measure things of the same kind, a fairly large number of such measurements will tend to cluster around the middle value. Such a value is called a measure of "Central Tendency". The other terms that are used synonymously are "Measures of Location", or "Statistical Averages".

Arithmetic Mean

Arithmetic Mean (called mean) is defined as the sum of all observations in a data set divided by the total number of observations. For example, consider a data set containing the following observations:

In symbolic form mean is given by $\bar{X} = \frac{\sum X}{n}$

\bar{X} = Arithmetic Mean

$\sum X$ = Indicates sum all X values in the data set

n = Total number of observations(Sample Size)

Arithmetic Mean -Example

The inner diameter of a particular grade of tire based on 5 sample measurements are as follows: (figures in millimeters)

565, 570, 572, 568, 585

Applying the formula $\bar{X} = \frac{\sum X}{n}$

We get mean = $(565+570+572+568+585)/5 = 572$

Caution: Arithmetic Mean is affected by extreme values or fluctuations in sampling. It is not the best average to use when the data set contains extreme values (Very high or very low values).

Median

Median is the middle most observation when you arrange data in ascending order of magnitude. Median is such that 50% of the observations are above the median and 50% of the observations are below the median.

Median is a very useful measure for ranked data in the context of consumer preferences and rating. It is not affected by extreme values (greater resistance to outliers)

$$\text{Median} = \frac{n+1}{2} \text{ th value of ranked data}$$

n = Number of observations in the sample

Median - Example

Marks obtained by 7 students in Computer Science Exam are given below: Compute the median.

45 40 60 80 90 65 55

Arranging the data after ranking gives

90 80 65 60 55 45 40

Median = $(n+1)/2$ th value in this set = $(7+1)/2$ th observation= 4th observation=60

Hence Median = 60 for this problem.

Mode

Mode is that value which occurs most often. It has the maximum frequency of occurrence. Mode also has resistance to outliers.

Mode is a very useful measure when you want to keep in the inventory, the most popular shirt in terms of collar size during festive season.

Mode -Example

The life in number of hours of 10 flashlight batteries are as follows: Find the mode.

340	350	340	340	320	340	330	330
340	350						

340 occurs five times. Hence, mode=340.

Comparison of Mean, Median, Mode

Mean	Median	Mode
Defined as the arithmetic average of all observations in the data set.	Defined as the middle value in the data set arranged in ascending or descending order.	Defined as the most frequently occurring value in the distribution; it has the largest frequency.
Requires measurement on all observations.	Does not require measurement on all observations	Does not require measurement on all observations
Uniquely and comprehensively defined.	Cannot be uniquely determined under all conditions.	Not uniquely defined for multi-modal situations.

Comparison of Mean, Median, Mode Cont.

Mean	Median	Mode
Affected by extreme values. Can be treated algebraically. That is, Means of several groups can be combined.	Not affected by extreme values. Cannot be treated algebraically. That is, Medians of several groups cannot be combined.	Not affected by extreme values. Cannot be treated algebraically. That is, Modes of several groups cannot be combined.

Measures of Dispersion

In simple terms, measures of dispersion indicate how large the spread of the distribution is around the central tendency. It answers unambiguously the question "What is the magnitude of departure from the average value for different groups having identical averages?".

Range

Range is the simplest of all measures of dispersion. It is calculated as the difference between maximum and minimum value in the data set.

$$\text{Range} = X_{\text{Maximum}} - X_{\text{Minimum}}$$

Range-Example

Example for Computing Range

The following data represent the percentage return on investment for 10 mutual funds per annum. Calculate Range.

12, 14, 11, 18, 10.5, 11.3, 12, 14, 11, 9

$$\text{Range} = X_{\text{Maximum}} - X_{\text{Minimum}} = 18 - 9 = 9$$

Caution: If one of the components of range namely the maximum value or minimum value becomes an extreme value, then range should not be used.

Inter-Quartile Range(IQR)

IQR= Range computed on middle 50% of the observations after eliminating the highest and lowest 25% of observations in a data set that is arranged in ascending order. IQR is less affected by outliers.

$$\text{IQR} = Q_3 - Q_1$$

Interquartile Range-Example

The following data represent the annual percentage returns of 9 mutual funds.

Data Set: 12, 14, 11, 18, 10.5, 12, 14, 11, 9

Arranging in ascending order, the data set becomes
9, 10.5, 11, 11, 12, 12, 14, 14, 18

$$\text{IQR} = Q_3 - Q_1 = 14 - 10.75 = 3.25$$

Standard Deviation

To define standard deviation, you need to define another term called variance. In simple terms, standard deviation is the square root of variance.

Example for Standard Deviation

The following data represent the percentage return on investment for 10 mutual funds per annum. Calculate the sample standard deviation.

12, 14, 11, 18, 10.5, 11.3, 12, 14, 11, 9

Solution for the Example

A	B	C	D
1			
2	X	$X - \bar{X}$	$(X - \bar{X})^2$
3	12	-0.28	0.08
4	14	1.72	2.96
5	11	-1.28	1.64
6	18	5.72	32.72
7	10.5	-1.78	3.17
8	11.3	-0.98	0.96
9	12	-0.28	0.08
10	14	1.72	2.96
11	11	-1.28	1.64
12	9	-3.28	10.76
13	Mean =		56.96
14	12.28	Variance =	6.33
15		Standard Deviation =	2.52

Standard Deviation Formula

Coefficient of Variation (Relative Dispersion)

Coefficient Variation (CV) is defined as the ratio of Standard Deviation to Mean.

In symbolic form

$$CV = \frac{S}{\bar{X}} \text{ for the sample data and } = \frac{\sigma}{\mu} \text{ for the population}$$

Coefficient of Variation

Example

Consider two SalesPersons working in the same territory

The sales performance of these two in the context of selling PCs are given below. Comment on the results.

Sales Person 1

Mean Sales (One year average)

50 units

Standard Deviation
5 units

Sales Person 2

Mean Sales (One year average)

75 units

Standard deviation
25 units

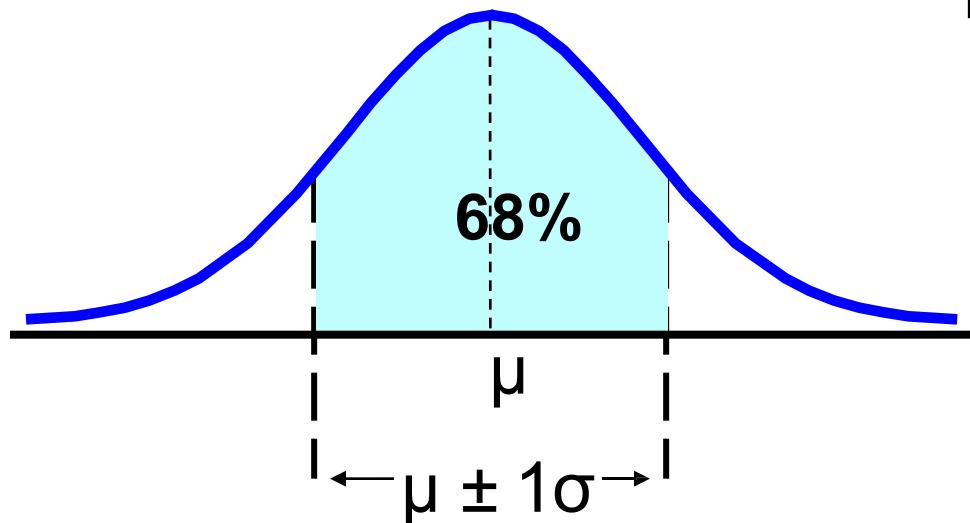
Interpretation for the Example

The CV is $5/50 = 0.10$ or 10% for the Sales Person1 and $25/75=0.33$ or 33% for sales Person2.

The moral of the story is "don't get carried away by averages. Consider variation ("risk").

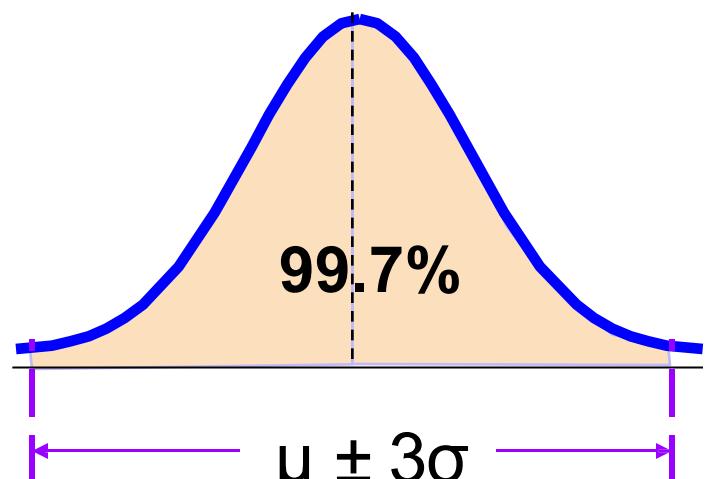
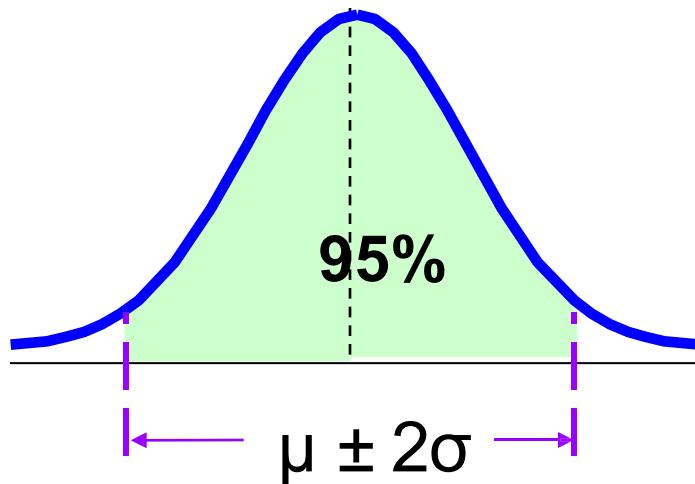
The Empirical Rule

- The empirical rule approximates the variation of data in a bell-shaped distribution
- Approximately 68% of the data in a bell shaped distribution is within 1 standard deviation of the mean or $\mu \pm 1\sigma$



The Empirical Rule

- Approximately 95% of the data in a bell-shaped distribution lies within two standard deviations of the mean, or $\mu \pm 2\sigma$
- Approximately 99.7% of the data in a bell-shaped distribution lies within three standard deviations of the mean, or $\mu \pm 3\sigma$



Chebyshev Rule

- Regardless of how the data are distributed, at least $(1 - 1/k^2) \times 100\%$ of the values will fall within k standard deviations of the mean (for $k > 1$)
- For Example, when $k=2$, at least 75% of the values of any data set will be within $\mu \pm 2\sigma$

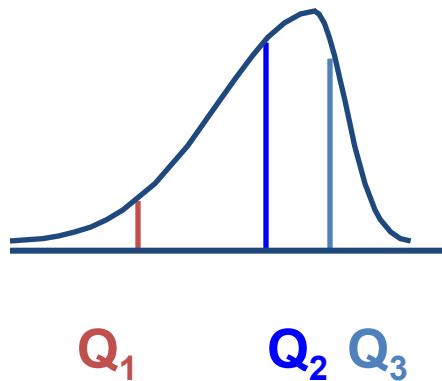
The Five Number Summary

The five numbers that help describe the center, spread and shape of data are:

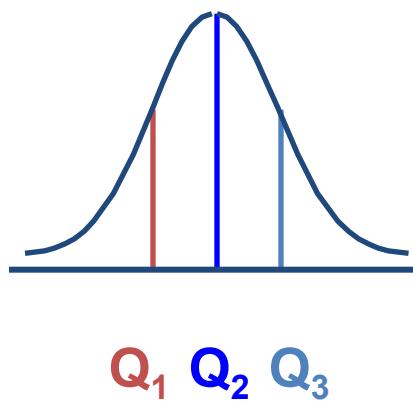
- X_{smallest}
- First Quartile (Q_1)
- Median (Q_2)
- Third Quartile (Q_3)
- X_{largest}

Distribution Shape

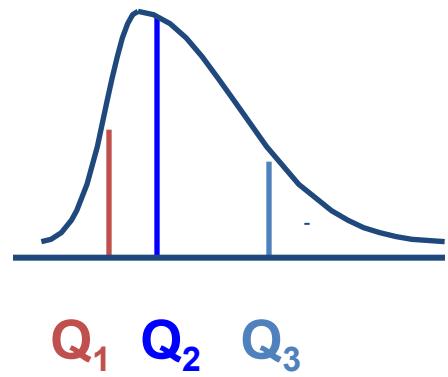
Left-Skewed



Symmetric



Right-Skewed



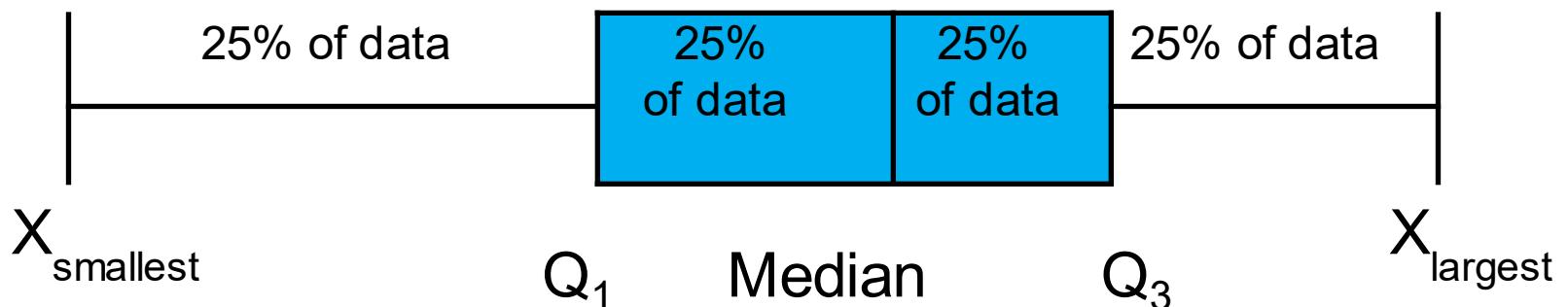
Relationships among the five-number summary and distribution shape

Left-Skewed	Symmetric	Right-Skewed
$\text{Median} - X_{\text{smallest}}$ $>$	$\text{Median} - X_{\text{smallest}}$ \approx	$\text{Median} - X_{\text{smallest}}$ $<$
$X_{\text{largest}} - \text{Median}$ $Q_1 - X_{\text{smallest}}$ $>$	$X_{\text{largest}} - \text{Median}$ $Q_1 - X_{\text{smallest}}$ \approx	$X_{\text{largest}} - \text{Median}$ $Q_1 - X_{\text{smallest}}$ $<$
$X_{\text{largest}} - Q_3$	$X_{\text{largest}} - Q_3$	$X_{\text{largest}} - Q_3$
$\text{Median} - Q_1$ $>$	$\text{Median} - Q_1$ \approx	$\text{Median} - Q_1$ $<$
$Q_3 - \text{Median}$	$Q_3 - \text{Median}$	$Q_3 - \text{Median}$

Five Number Summary and The Boxplot

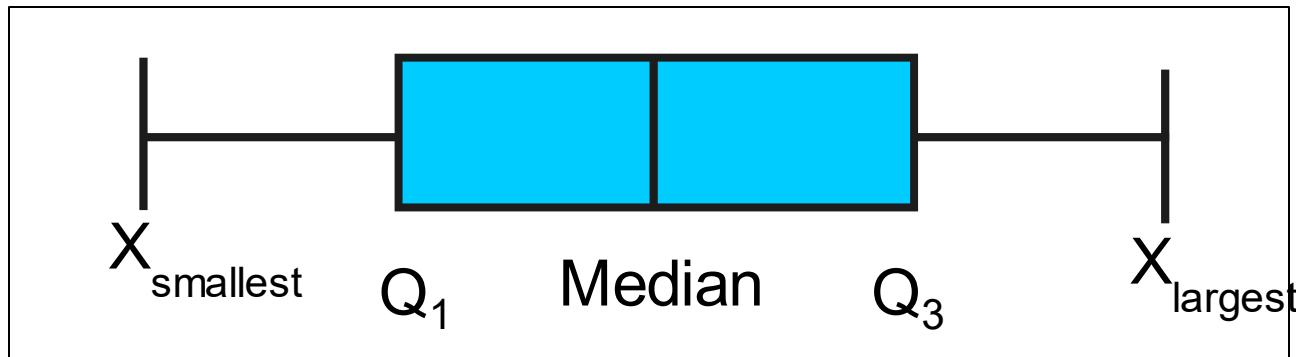
- **The Boxplot:** A Graphical display of the data based on the five-number summary:

Example:



Five Number Summary: Shape of Boxplots

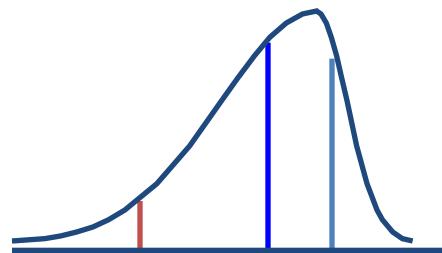
- If data are symmetric around the median then the box and central line are centered between the endpoints



- A Boxplot can be shown in either a vertical or horizontal orientation

Distribution Shape and The Boxplot

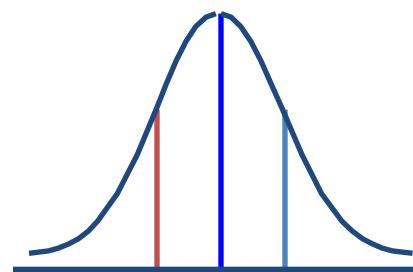
Left-Skewed



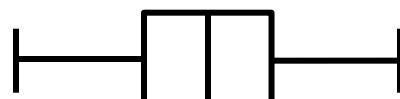
Q_1 Q_2 Q_3



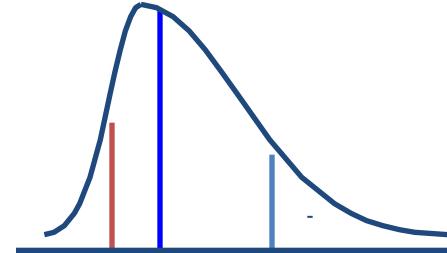
Symmetric



Q_1 Q_2 Q_3



Right-Skewed



Q_1 Q_2 Q_3



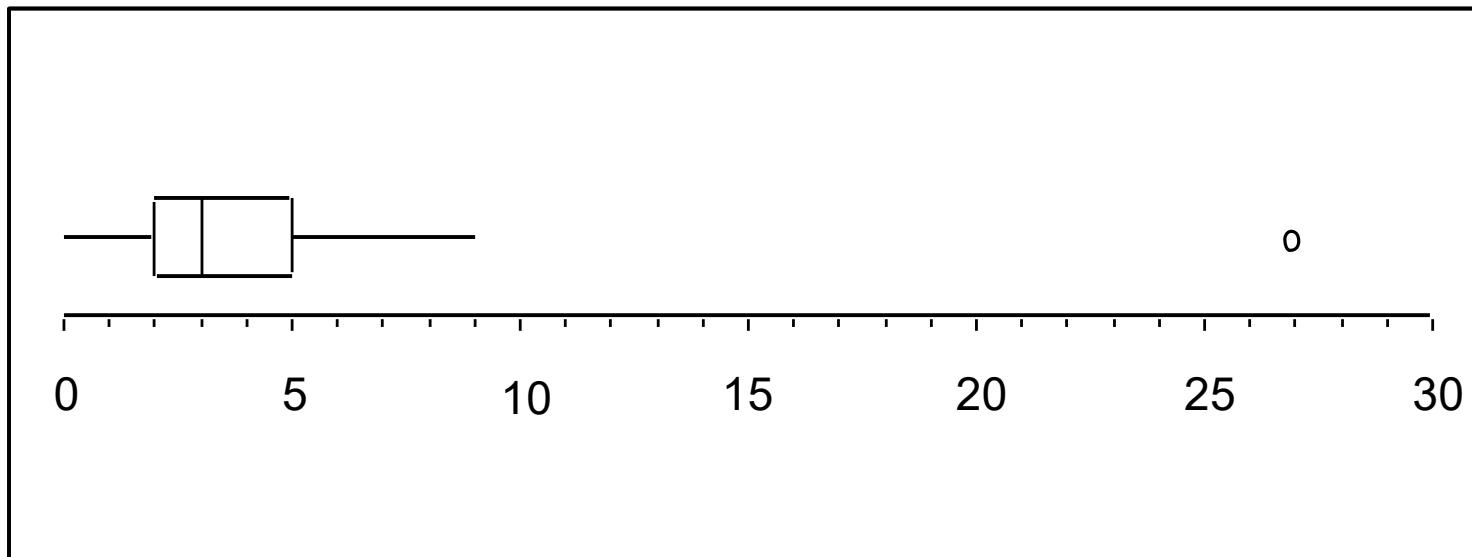
Boxplot Example

The data are right skewed in the following plot



Box plot example showing an outlier

- The boxplot below of the same data shows the outlier value of 27 plotted separately
- A value is considered an outlier if it is more than 1.5 times the interquartile range below Q_1 or above Q_3



Graphic Displays of Basic Statistical Descriptions

Boxplot: graphic display of five-number summary

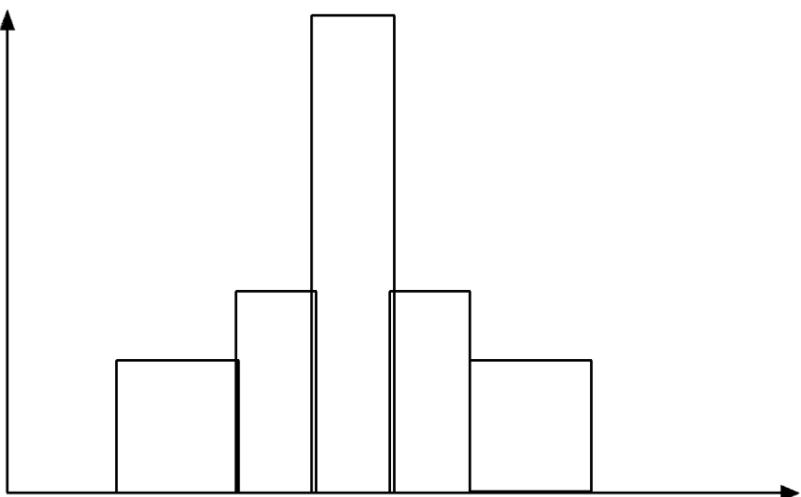
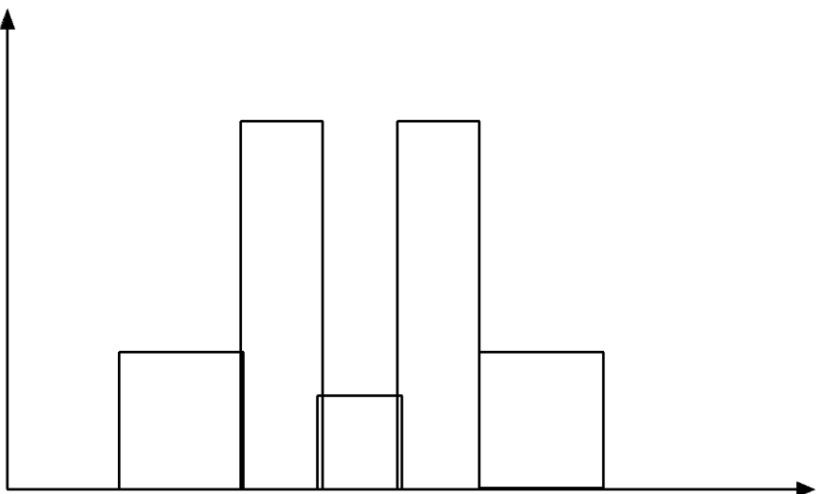
Histogram: x-axis are values, y-axis repres. frequencies

Quantile plot: each value x_i is paired with f_i indicating that approximately $100 f_i \%$ of data are $\leq x_i$

Quantile-quantile (q-q) plot: graphs the quantiles of one univariant distribution against the corresponding quantiles of another

Scatter plot: each pair of values is a pair of coordinates and plotted as points in the plane

Histograms Often Tell More than Boxplots



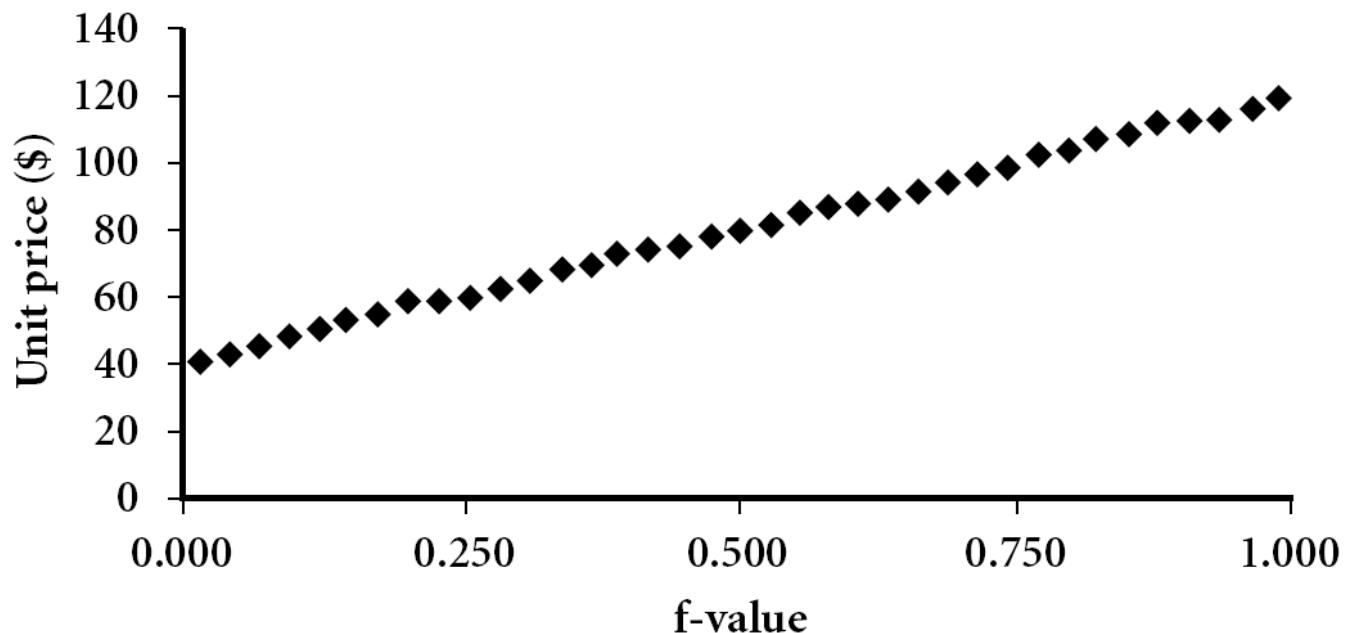
- The two histograms shown in the left may have the same boxplot representation
 - The same values for: min, Q1, median, Q3, max
- But they have rather different data distributions

Quantile Plot

Displays all of the data (allowing the user to assess both the overall behavior and unusual occurrences)

Plots **quantile** information

For a data x_i , data sorted in increasing order, f_i indicates that approximately $100 f_i\%$ of the data are below or equal to the value x_i

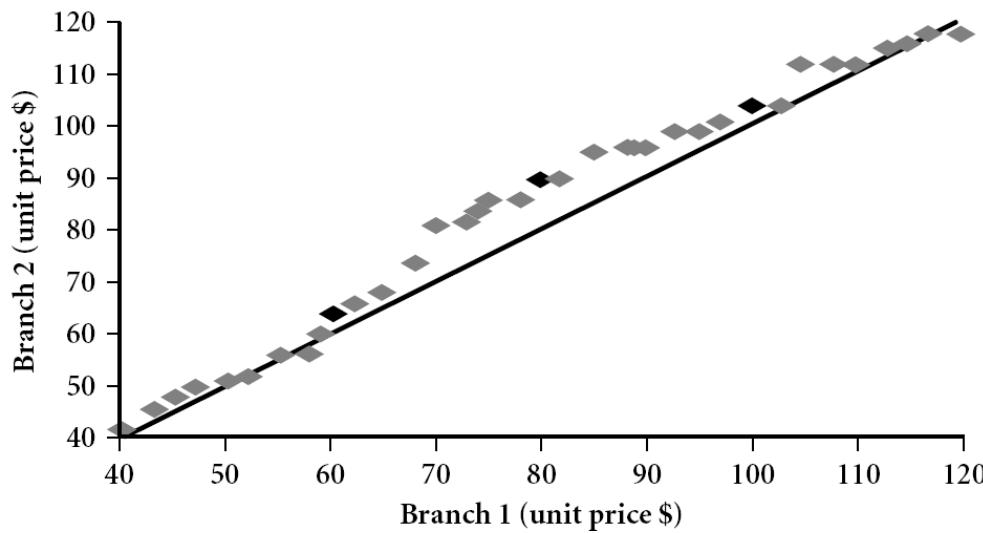


Quantile-Quantile (Q-Q) Plot

Graphs the quantiles of one univariate distribution against the corresponding quantiles of another

View: Is there is a shift in going from one distribution to another?

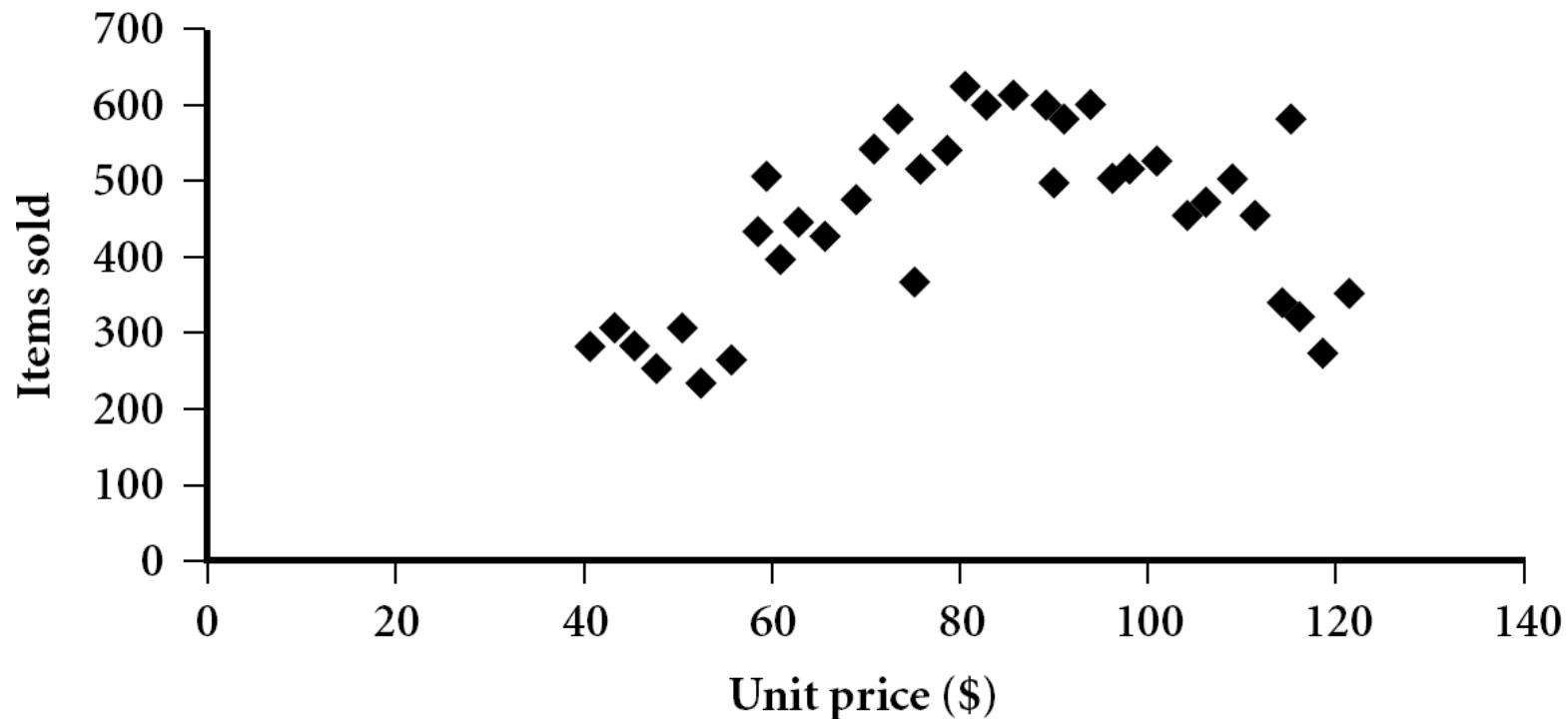
Example shows unit price of items sold at Branch 1 vs. Branch 2 for each quantile. Unit prices of items sold at Branch 1 tend to be lower than those at Branch 2.



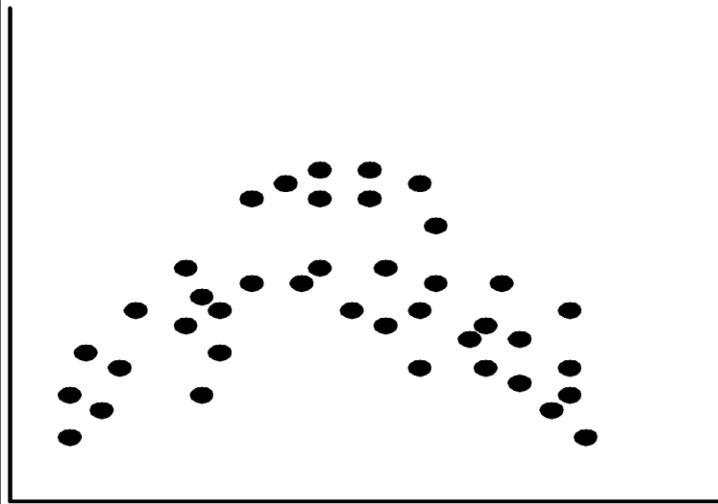
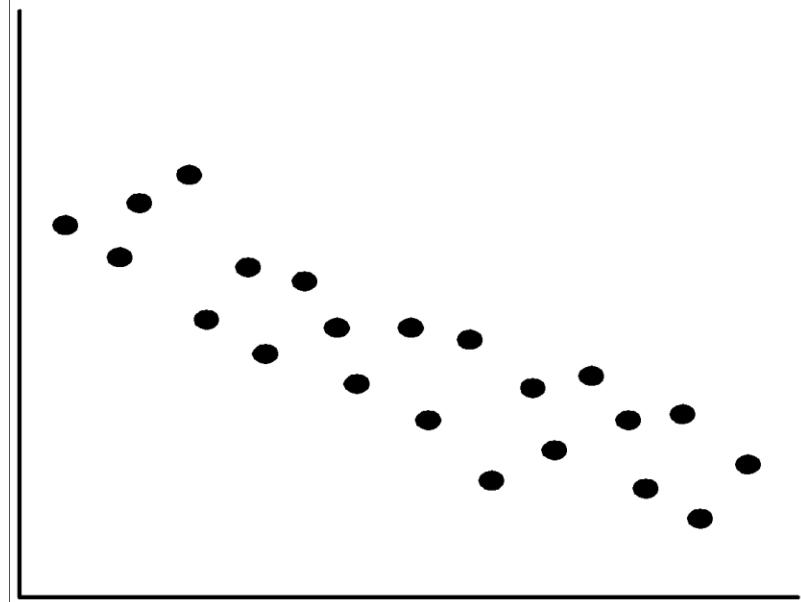
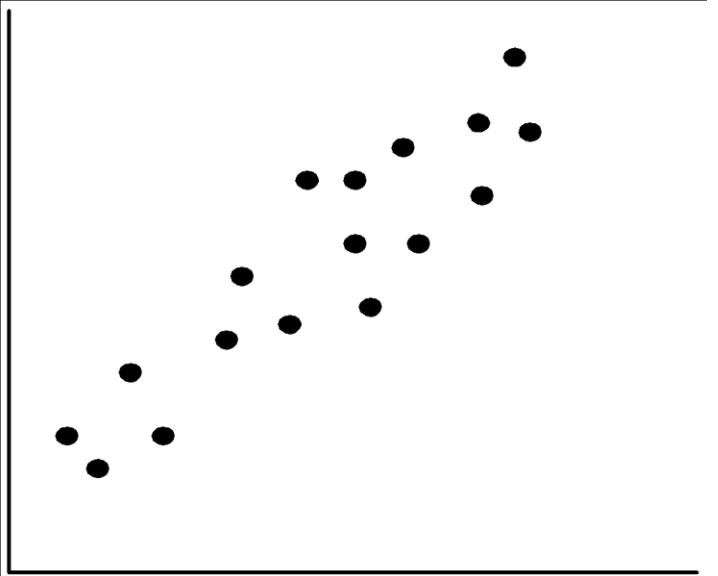
Scatter plot

Provides a first look at bivariate data to see clusters of points, outliers, etc

Each pair of values is treated as a pair of coordinates and plotted as points in the plane



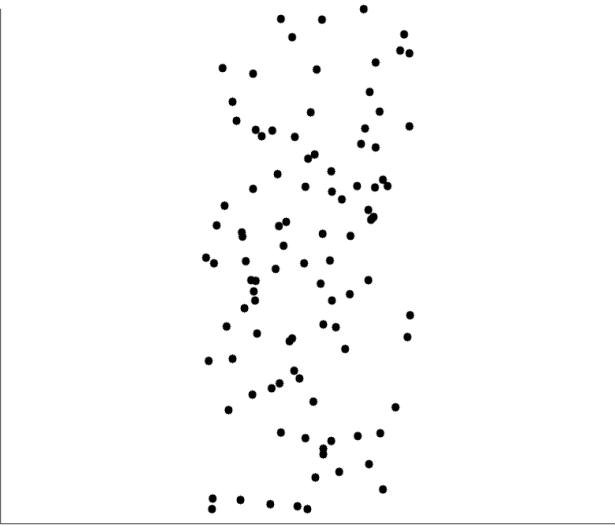
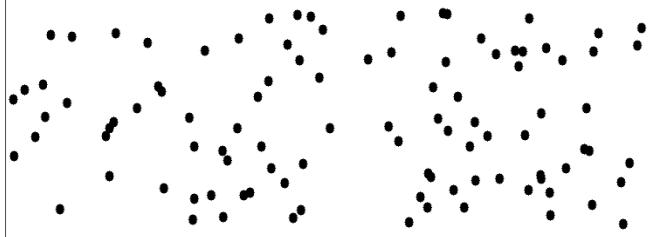
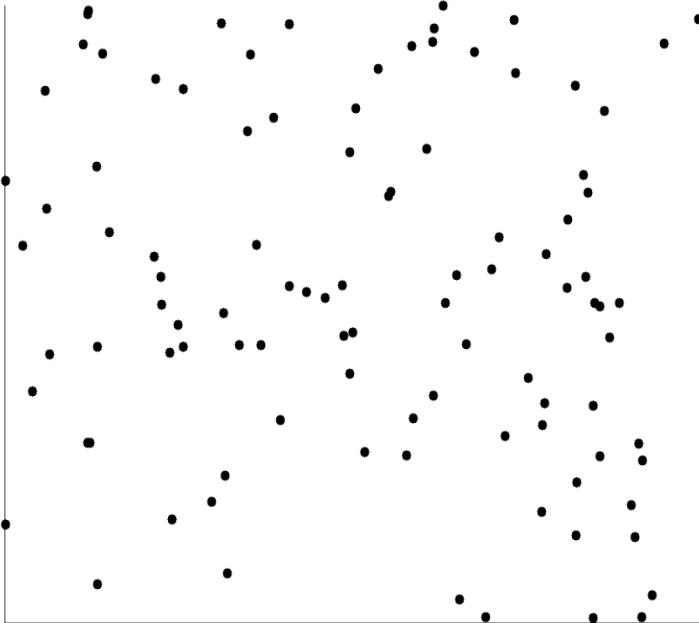
Positively and Negatively Correlated Data



The left half fragment is positively correlated

The right half is negative correlated

Uncorrelated Data



Data Visualization

Why data visualization?

Gain insight into an information space by mapping data onto graphical primitives

Provide qualitative overview of large data sets

Search for patterns, trends, structure, irregularities, relationships among data

Help find interesting regions and suitable parameters for further quantitative analysis

Provide a visual proof of computer representations derived

Categorization of visualization methods:

Pixel-oriented visualization techniques

Geometric projection visualization techniques

Icon-based visualization techniques

Hierarchical visualization techniques

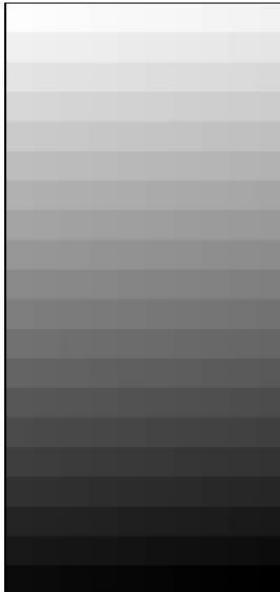
Visualizing complex data and relations

Pixel-Oriented Visualization Techniques

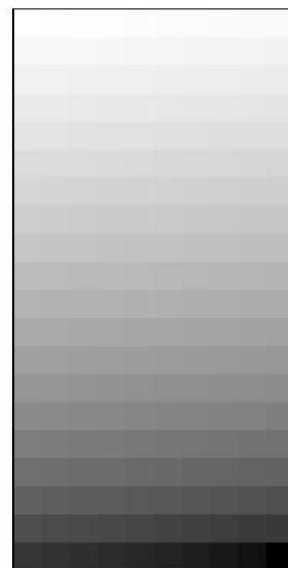
For a data set of m dimensions, create m windows on the screen, one for each dimension

The m dimension values of a record are mapped to m pixels at the corresponding positions in the windows

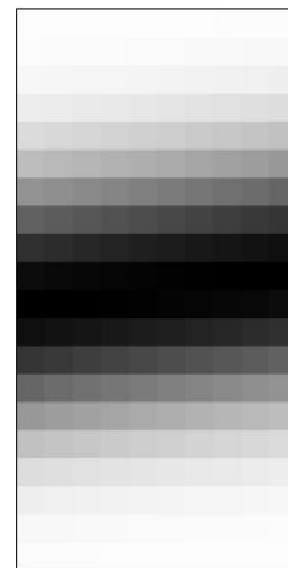
The colors of the pixels reflect the corresponding values



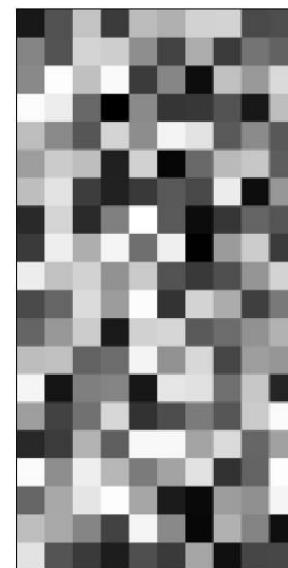
(a) Income



(b) Credit Limit



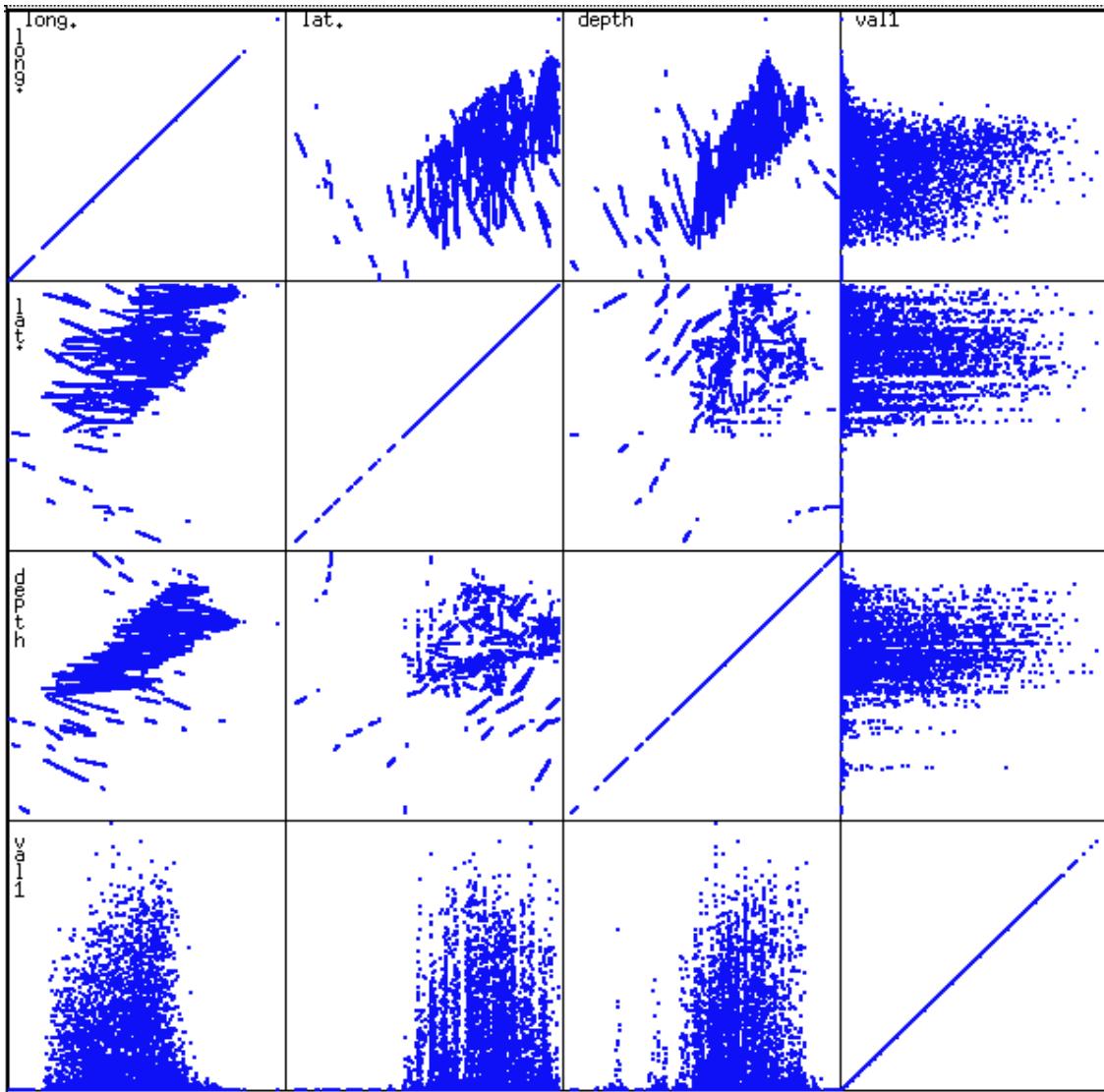
(c) transaction volume



(d) age

Scatterplot Matrices

Used by permission of M. Ward, Worcester Polytechnic Institute



Matrix of scatterplots (x-y-diagrams) of the k-dim. data [total of $(k^2 - k)/2$ scatterplots]

Correlation Analysis (Nominal Data): Chi-Square Test

	Play chess	Not play chess	Sum (row)
Like science fiction	250(90)	200(360)	450
Not like science fiction	50(210)	1000(840)	1050
Sum(col.)	300	1200	1500

$$e_{ij} = \frac{\text{count}(A = a_i) \times \text{count}(B = b_j)}{n}$$

$$e_{11} = \frac{\text{count}(male) \times \text{count}(fiction)}{n} = \frac{300 \times 450}{1500} = 90.$$

For this 2×2 table, the degrees of freedom are $(2 - 1)(2 - 1) = 1$. For 1 degree of freedom, the χ^2 value needed to reject the hypothesis at the 0.001 significance level is 10.828

χ^2 (chi-square) calculation (numbers in parenthesis are expected counts calculated based on the data distribution in the two categories)

$$\chi^2 = \sum \frac{(Observed - Expected)^2}{Expected}$$

$$\chi^2 = \frac{(250 - 90)^2}{90} + \frac{(50 - 210)^2}{210} + \frac{(200 - 360)^2}{360} + \frac{(1000 - 840)^2}{840} = 507.93$$

Shows that like_science_fiction and play_chess are correlated in the group.

Correlation Analysis (Numeric Data)

Correlation coefficient (also called Pearson's product moment coefficient)

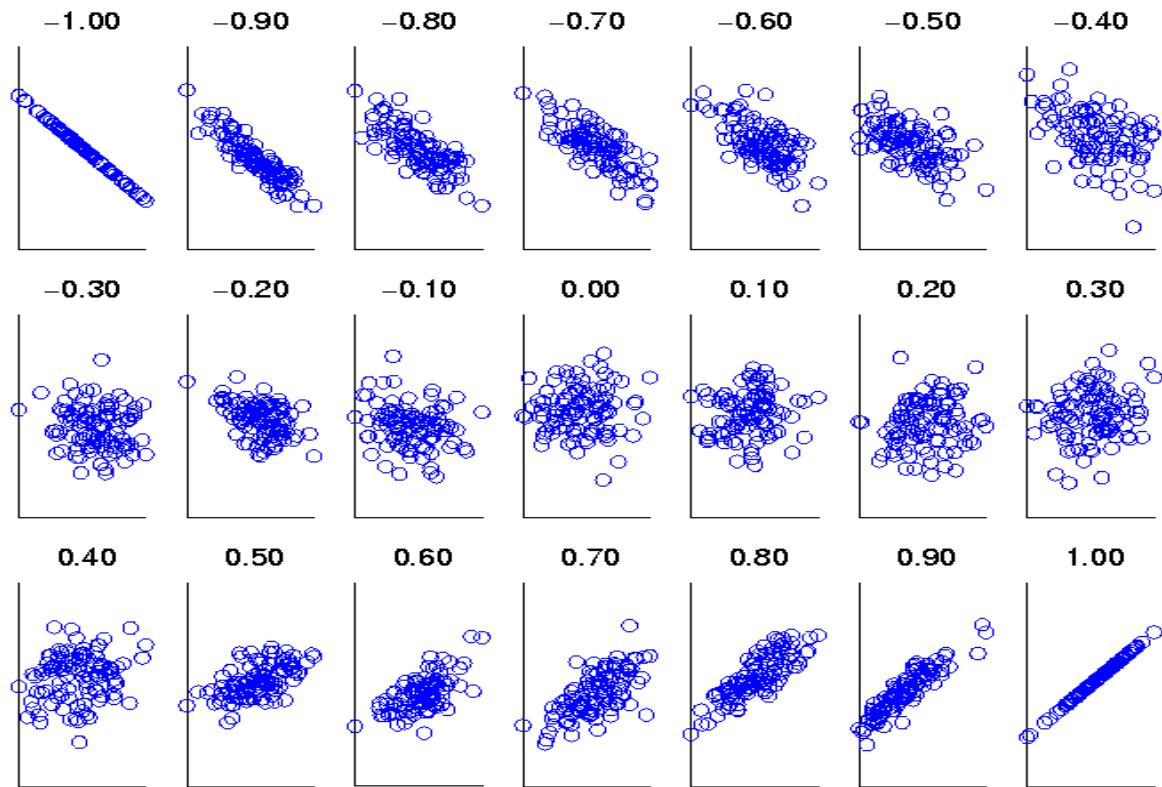
$$r_{A,B} = \frac{\sum_{i=1}^n (a_i - \bar{A})(b_i - \bar{B})}{(n-1)\sigma_A \sigma_B} = \frac{\sum_{i=1}^n (a_i b_i) - n \bar{A} \bar{B}}{(n-1)\sigma_A \sigma_B}$$

where n is the number of tuples, \bar{A} and \bar{B} are the respective means of A and B, σ_A and σ_B are the respective standard deviation of A and B, and $\Sigma(a_i b_i)$ is the sum of the AB cross-product.

If $r_{A,B} > 0$, A and B are positively correlated (A's values increase as B's). The higher, the stronger correlation.

$r_{A,B} = 0$: independent; $r_{AB} < 0$: negatively correlated

Visually Evaluating Correlation



**Scatter plots
showing the
similarity from
-1 to 1.**

Summary

- Histograms
- Measures of central tendency: mean, mode, median
- Measures of dispersion: range, IQR, variance, std deviation, coefficient of variation.
- Normal distribution, Chebyshev Rule.
- Five number summary, boxplots, QQ plots, Quantile plot, scatter plot.
- Visualization: scatter plot matrix, parallel coordinates.
- Correlation analysis.

Statistical Learning - Probability and Distributions

Probability – Meaning & Concepts

- **Probability** refers to chance or likelihood of a particular event-taking place.
- An **event** is an outcome of an experiment.
- An **experiment** is a process that is performed to understand and observe possible outcomes.
- Set of all outcomes of an experiment is called the **sample space**.

Example

- In a manufacturing unit three parts from the assembly are selected. You are observing whether they are defective or non-defective.

Determine

- a) The sample space.
- b) The event of getting at least two defective parts.

Solution

a) Let S = Sample Space. It is pictured as under:

GGG GGD GDG DGG

GDD DGD DDG DDD

D = Defective

G = Non-Defective

b) Let E denote the event of getting at least two defective parts. This implies that E will contain two defectives, and three defectives. Looking at the sample space diagram above, $E = \{GDD, DGD, DDG, DDD\}$. It is easy to see that E is a part of S and commonly called as a subset of S . Hence an event is always a subset of the sample space.

Definition of Probability

Probability of an event A is defined as the ratio of two numbers m and n. In symbols

$$P(A) = \frac{m}{n}$$

where m= number of ways that are favorable to the occurrence of A and n= the total number of outcomes of the experiment (all possible outcomes)

Please note that P (A) is always ≥ 0 and always ≤ 1 .
P (A) is a pure number.

Extreme Values of Probability

The range within which probability of an event lies can be best understood by the following diagram. The glass shows three stages- Empty, half-full, and full to explain the properties of probability.



100% Chance or Certainty



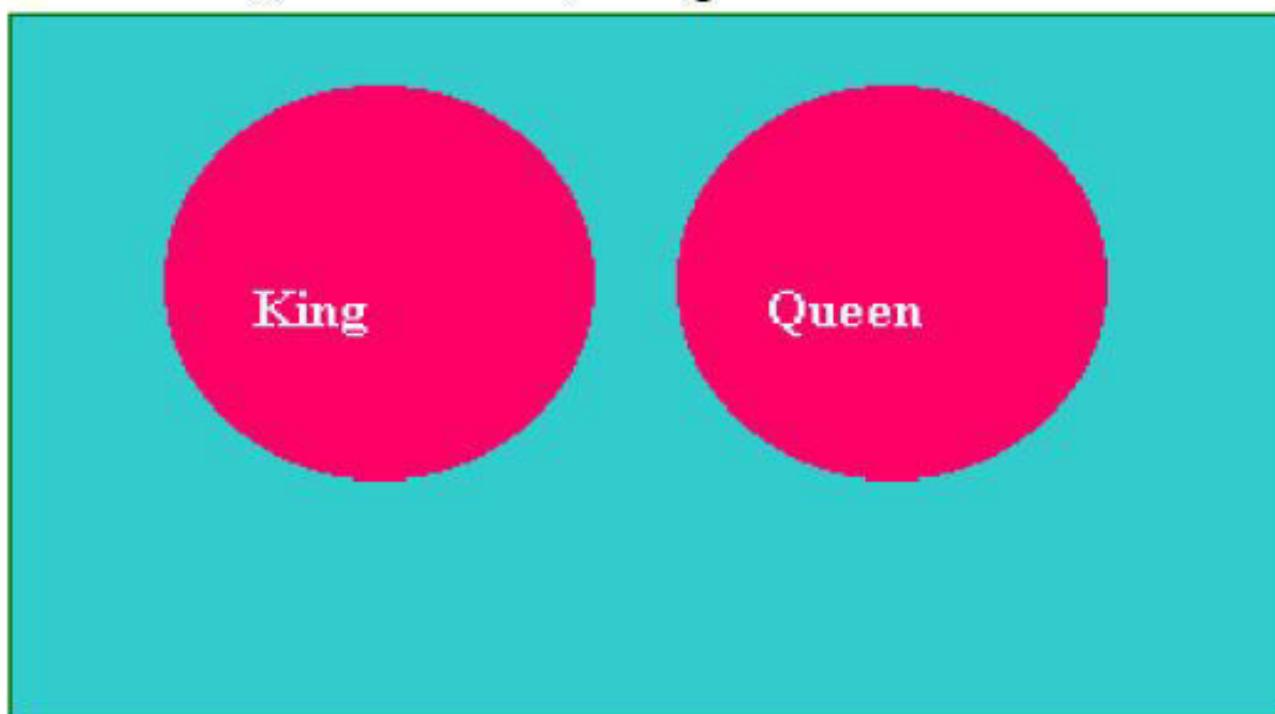
(50% Chance) Equally Likely



(0% Chance) Impossibility

Mutually Exclusive Events

Two events A and B are said to be mutually exclusive if the occurrence of A precludes the occurrence of B. For example, from a well shuffled pack of cards, if you pick up one card at random and would like to know whether it is a King or a Queen. The selected card will be either a King or a Queen. It cannot be both a King and a Queen. If King occurs, Queen will not occur and Queen occurs, King will not occur.



Independent Events

- Two events A and B are said to be independent if the occurrence of A is in no way influenced by the occurrence of B. Likewise occurrence of B is in no way influenced by the occurrence of A.

Rules for Computing Probability

1) Addition Rule -Mutually Exclusive Events

$$P(A \cup B) = P(A) + P(B)$$

This rule says that the probability of the union of A and B is determined by adding the probability of the events A and B.

Here the symbol $A \cup B$ is called A union B meaning A occurs, or B occurs or both A and B simultaneously occur. When A and B are mutually exclusive, A and B cannot simultaneously occur.

Rules for Computing Probability

2) Addition Rule -Events are not Mutually Exclusive

$$P(A \cup B) = P(A) + P(B) - P(A \cap B)$$

This rule says that the probability of the union of A and B is determined by adding the probability of the events A and B and then subtracting the probability of the intersection of the events A and B.

The symbol $A \cap B$ is called A intersection B meaning

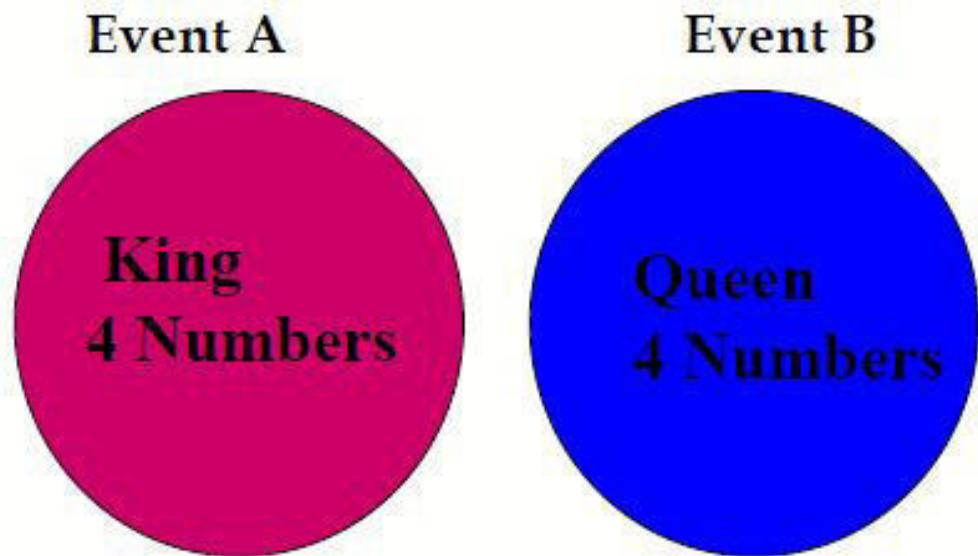
both A and B simultaneously occur.

Example for Addition Rules

- From a pack of well-shuffled cards, a card is picked up at random.
- 1) What is the probability that the selected card is a King or a Queen?
 - 2) What is the probability that the selected card is a King or a Diamond?

Solution to Part 1)

Look at the Diagram:



Let A = getting a King

Let B = getting a Queen

There are 4 kings and there are 4 Queens. The events are clearly mutually exclusive. Applying the formula $P(A \cup B) = P(A) + P(B)$
 $= 4/52 + 4/52 = 8/52 = 2/13$

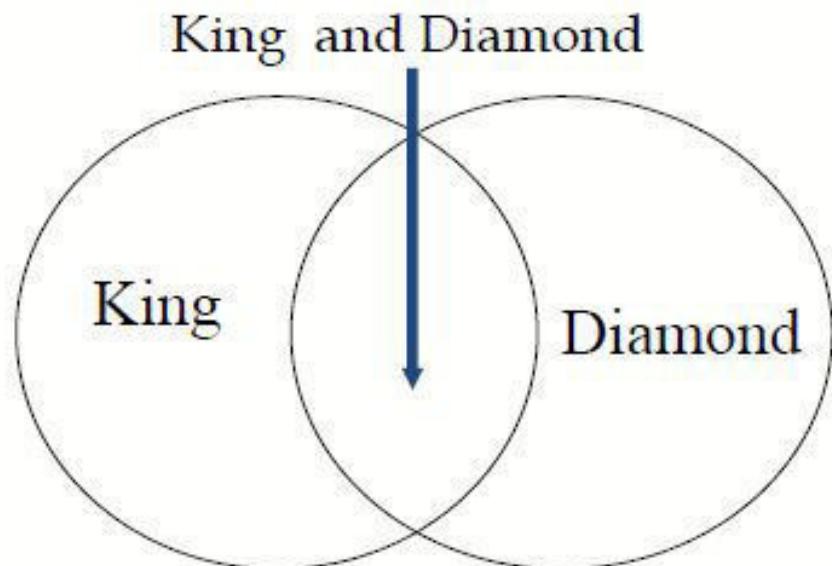
Solution to part 2)

INSTITUTE OF MANAGEMENT, CHEN

Look at the Diagram:

There are totally 52 cards in a pack out of which 4 are Kings and 13 are Diamonds. Let A= getting a King and B= getting a Diamond. The two events here are not mutually exclusive because you can have a card, which is both a King and a Diamond called King Diamond.

$$\begin{aligned} P(K \cup D) &= P(K) + P(D) - P(K \cap D) \\ &= 4/52 + 13/52 - 1/52 = 16/52 = 4/13 \end{aligned}$$



Multiplication Rule

Independent Events

$$P(A \cap B) = P(A) \cdot P(B)$$

This rule says when the two events A and B are independent, the probability of the simultaneous occurrence of A and B (also known as probability of intersection of A and B) equals the product of the probability of A and the probability of B. Of course this rule can be extended to more than two events.

Multiplication Rule

Independent Events-Example

Example:

The probability that you will get an A grade in Quantitative Methods is 0.7. The probability that you will get an A grade in Marketing is 0.5. Assuming these two courses are independent, compute the probability that you will get an A grade in both these subjects.

Solution:

Let A = getting A grade in Quantitative Methods

Let B =getting A grade in Marketing

It is given that A and B are independent.

$$P(A \cap B) = P(A) \cdot P(B) = 0.7 \cdot 0.5 = 0.35.$$

Multiplication Rule

Events are not independent

$$P(A \cap B) = P(A) \cdot P(B/A)$$

This rule says that the probability of the intersection of the events A and B equals the product of the probability of A and the probability of B given that A has happened or known to you. This is symbolized in the second term of the above expression as $P(B/A)$. $P(B/A)$ is called the conditional probability of B given the fact that A has happened.

We can also write $P(A \cap B) = P(B) \cdot P(A/B)$ if B has already happened.

Multiplication Rule

Events are not independent-Example

INSTITUTE OF MANAGEMENT, CHE

From a pack of cards, 2 cards are drawn in succession one after the other. After every draw, the selected card is not replaced. What is the probability that in both the draws you will get Spades?

Solution:

Let A = getting Spade in
the first draw

Let B = getting spade in the second draw.
The cards are not replaced.

This situation requires the use of conditional probability.

$P(A) = 13/52$ (There are 13 Spades and 52 cards in a pack)

$P(B/A)=12/51$ (There are 12 Spades and 51 cards because the first card selected is not replaced after the first draw)

$$P(A \cap B) = P(A).P(B/A) = (13/52).(12/51) = 156/2652 = 1/17.$$

Marginal Probability

- Contingency table consists of rows and columns of two attributes at different levels with frequencies or numbers in each of the cells. It is a matrix of frequencies assigned to rows and columns.
- The term marginal is used to indicate that the probabilities are calculated using a contingency table (also called joint probability table).

Marginal Probability - Example

A survey involving 200 families was conducted. Information regarding family income per year and whether the family buys a car are given in the following table.

Family	Income below Rs 10 Lakhs	Income of Rs. ≥ 10 lakhs	Total
Buyer of Car	38	42	80
Non-Buyer	82	38	120
Total	120	80	200

- What is the probability that a randomly selected family is a buyer of the car?
- What is the probability that a randomly selected family is both a buyer of car and belonging to income of Rs. 10 lakhs and above?
- A family selected at random is found to be belonging to income of Rs 10 lakhs and above. What is the probability that this family is buyer of car?

Solution

- a) What is the probability that a randomly selected family is a buyer of the Car?
 - $80/200 = 0.40.$

- b) What is the probability that a randomly selected family is both a buyer of car and belonging to income of Rs. 10 lakhs and above?
 - $42/200 = 0.21.$

- c) A family selected at random is found to be belonging to income of Rs 10 lakhs and above. What is the probability that this family is buyer of car?
 - $42/80 = 0.525.$ Note this is a case of conditional probability of buyer given income is Rs. 10 lakhs and above.

Bayes' Theorem

- Bayes' Theorem is used to revise previously calculated probabilities based on new information.
- Developed by Thomas Bayes in the 18th Century.
- It is an extension of conditional probability.

Bayes' Theorem

Given a hypothesis H and evidence E , Bayes' theorem states that the relationship between the probability of the hypothesis $P(H)$ before getting the evidence and the probability $P(H | E)$ of the hypothesis after getting the evidence is

$$P(H | E) = \frac{P(E | H)}{P(E)} P(H).$$

Many modern machine learning techniques rely on Bayes' theorem. For instance, spam filters use Bayesian updating to determine whether an email is real or spam, given the words in the email. Additionally, many specific techniques in statistics, such as calculating p-values or interpreting medical results, are best described in terms of how they contribute to updating hypotheses using Bayes' theorem.

Bayes' Theorem

$$P(B_i | A) = \frac{P(A | B_i)P(B_i)}{P(A | B_1)P(B_1) + P(A | B_2)P(B_2) + \dots + P(A | B_k)P(B_k)}$$

- where:
 - B_i = i^{th} event of k mutually exclusive and collectively exhaustive events
 - A = new event that might impact $P(B_i)$

Bayes' Theorem - Example

Bayesian Spam Filtering

One clever application of Bayes' Theorem is in spam filtering. We have

- Event A: The message is spam.
- Test X: The message contains certain words (X)

Plugged into a more readable formula (from Wikipedia):

$$\Pr(\text{spam}|\text{words}) = \frac{\Pr(\text{words}|\text{spam}) \Pr(\text{spam})}{\Pr(\text{words})}$$

Bayes' Theorem - Example

Bayesian filtering allows us to predict the chance a message is really spam given the “test results” (the presence of certain words). Clearly, words like “viagra” have a higher chance of appearing in spam messages than in normal ones.

Spam filtering based on a blacklist is flawed — it’s too restrictive and false positives are too great. But Bayesian filtering gives us a middle ground — we use *probabilities*. As we analyze the words in a message, we can compute the chance it is spam (rather than making a yes/no decision). If a message has a 99.9% chance of being spam, it probably is. As the filter gets trained with more and more messages, it updates the probabilities that certain words lead to spam messages. Advanced Bayesian filters can examine multiple words in a row, as another data point.

What is a Probability Distribution

- In precise terms, a **probability distribution** is a total listing of the various values the random variable can take along with the corresponding probability of each value. A real life example could be the pattern of distribution of the machine breakdowns in a manufacturing unit.
- The random variable in this example would be the various values the machine breakdowns could assume.
- The probability corresponding to each value of the breakdown is the relative frequency of occurrence of the breakdown.
- The probability distribution for this example is constructed by the actual breakdown pattern observed over a period of time. Statisticians use the term “observed distribution” of breakdowns.

Binomial Distribution

- The Binomial Distribution is a widely used probability distribution of a discrete random variable.
- It plays a major role in **quality control** and **quality assurance** function. Manufacturing units do use the binomial distribution for **defective** analysis.
- Reducing the number of defectives using the proportion defective control chart (p chart) is an accepted practice in manufacturing organizations.
- Binomial distribution is also being used in **service organizations** like banks, and insurance corporations to get an idea of the proportion customers who are satisfied with the service quality.

Conditions for Applying Binomial Distribution (Bernoulli Process)

- Trials are independent and random.
- There are fixed number of trials (n trials).
- There are only two outcomes of the trial designated as *success* or *failure*.
- The probability of success is uniform through out the n trials

Binomial Probability Function

Under the conditions of a Bernoulli process,

The probability of getting x successes out of n trials is indeed the definition of a Binomial Distribution. The Binomial Probability Function is given by the following expression

$$P(x) = \binom{n}{x} P^x (1 - P)^{n-x}$$

Where $P(x)$ is the probability of getting x successes in n trials

$\binom{n}{x}$ is the number of ways in which x successes can take place out of n trials

$$= \frac{n!}{x!(n-x)!}$$

P is the probability of success, which is the same through out the n trials.

P is the parameter of the Binomial distribution

x can take values 0, 1, 2, , n

Example for Binomial Distribution

A bank issues credit cards to customers under the scheme of Master Card. Based on the past data, the bank has found out that 60% of all accounts pay on time following the bill. If a sample of 7 accounts is selected at random from the current database, construct the Binomial Probability Distribution of accounts paying on time.

Mean and Standard Deviation of the Binomial Distribution

The mean μ of the Binomial Distribution is given by $\mu = E(x) = np$

The Standard Deviation σ is given by

$$\sigma = \sqrt{np(1-p)}$$

For the example problem in the previous two slides,
Mean $= 7 \times 0.6 = 4.2$.

$$\text{Standard Deviation} = \sqrt{4.2(1 - 0.60)} = 1.30$$

Poisson Distribution

- Poisson Distribution is another discrete distribution which also plays a major role in **quality control** in the context of reducing the number of defects per standard unit.
- Examples include number of defects per item, number of defects per transformer produced, number of defects per 100 m² of cloth, etc.
- Other examples would include 1) The number of cars arriving at a highway check post per hour; 2) The number of customers visiting a bank per hour during peak business period; 3) The number of pixels in the image that are corrupted.

Poisson Probability Function

Poisson Distribution Formula

$$P(x) = \frac{e^{-\lambda} \lambda^x}{x!}$$

where

$P(x)$ = Probability of x successes given an idea of λ

λ = Average number of successes

e = 2.71828(based on natural logarithm)

x = successes per unit which can take values $0, 1, 2, 3, \dots, \infty$

λ is the Parameter of the Poisson Distribution.

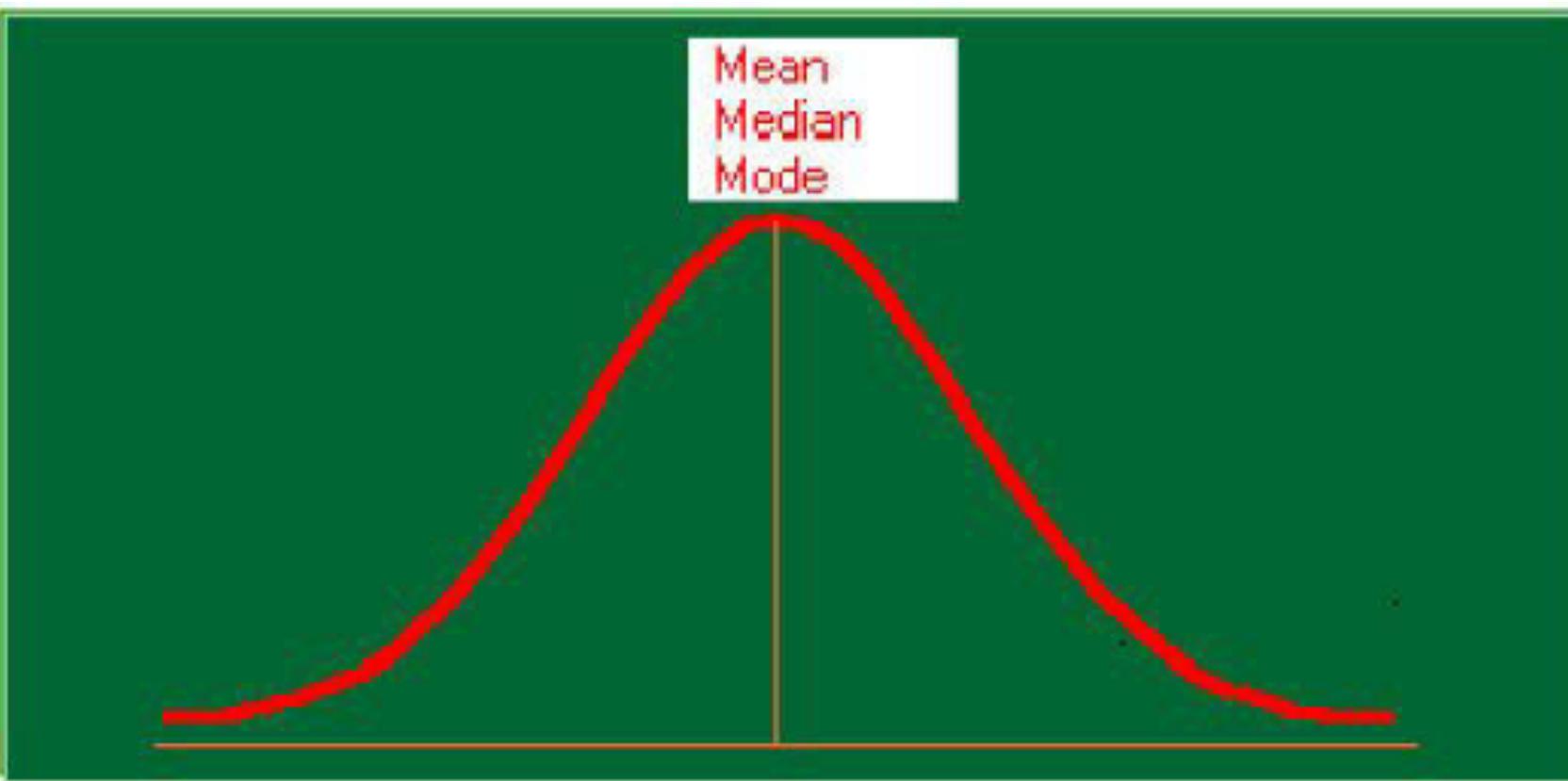
Mean of the Poisson Distribution is = λ

Standard Deviation of the Poisson Distribution is = $\sqrt{\lambda}$

Example – Poisson Distribution

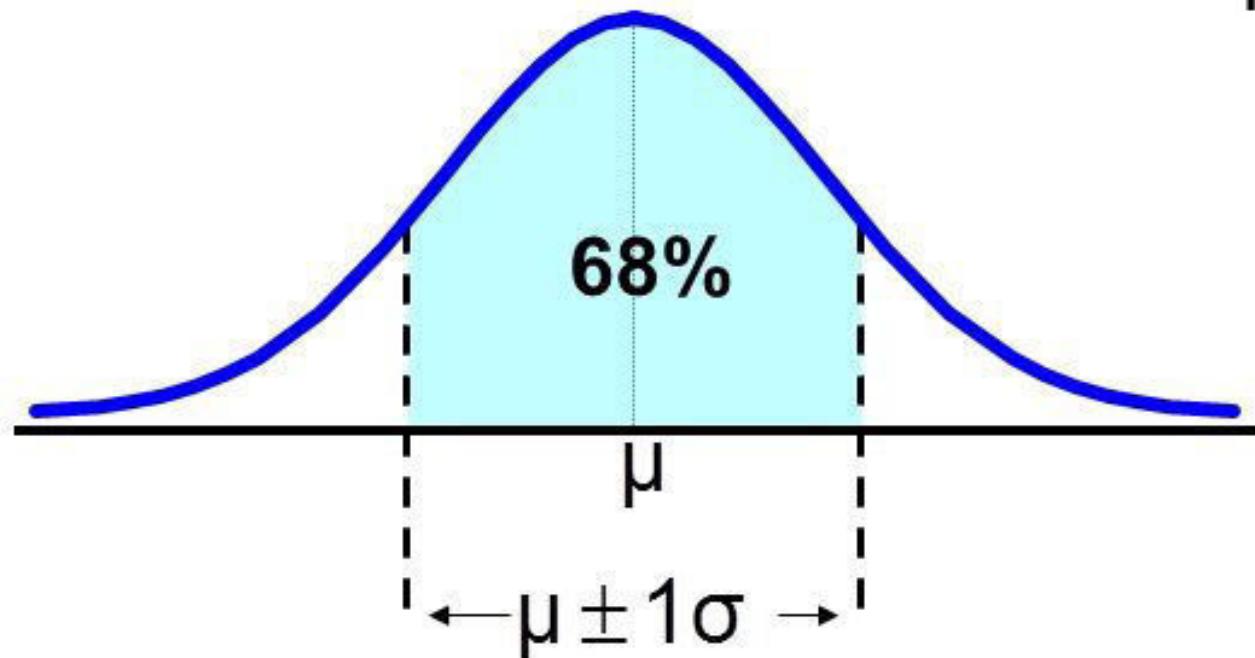
If on an average, 6 customers arrive every two minutes at a bank during the busy hours of working, a) what is the probability that exactly four customers arrive in a given minute? b) What is the probability that more than three customers will arrive in a given minute?

Normal Distribution



Normal Distribution

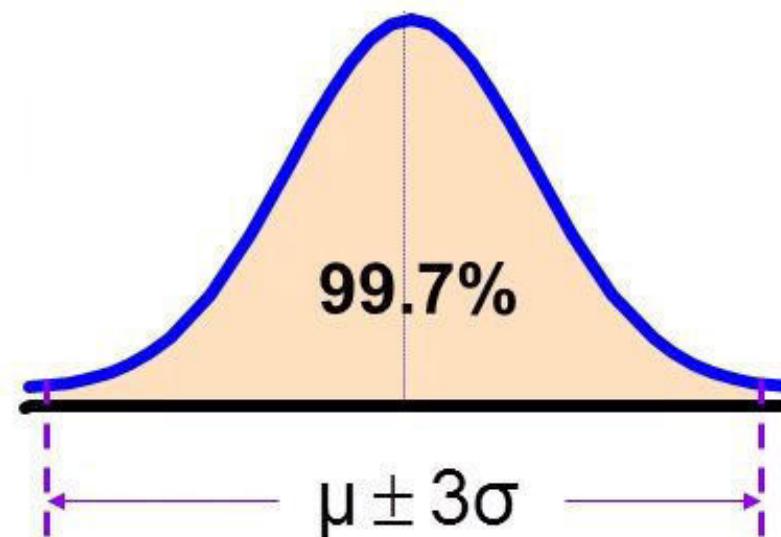
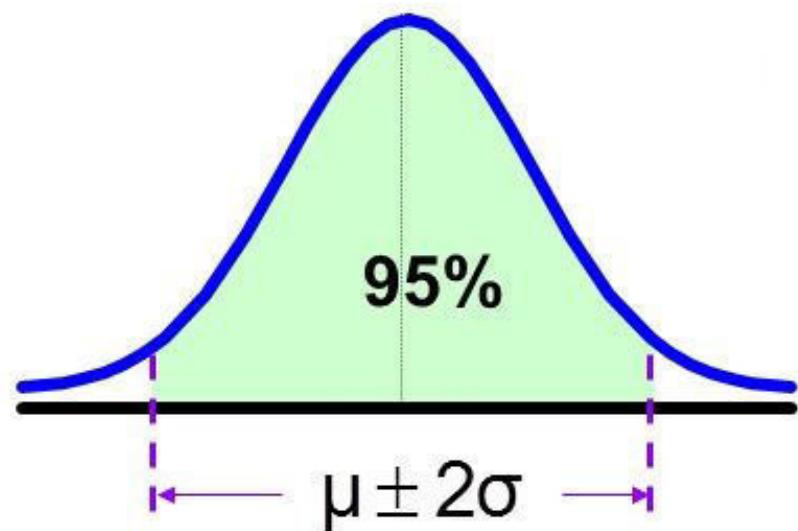
- The empirical rule approximates the variation of data in a bell-shaped distribution
- Approximately 68% of the data in a bell shaped distribution is within 1 standard deviation of the mean or $\mu \pm 1\sigma$



Normal Distribution

INSTITUTE OF MANAGEMENT,

- Approximately 95% of the data in a bell-shaped distribution lies within two standard deviations of the mean, or $\mu \pm 2\sigma$
- Approximately 99.7% of the data in a bell-shaped distribution lies within three standard deviations of the mean, or $\mu \pm 3\sigma$



Properties of Normal Distribution

- The **normal distribution** is a **continuous distribution** looking like a **bell**. Statisticians use the expression “**Bell Shaped Distribution**”.
- It is a **beautiful distribution** in which the **mean**, the **median**, and the **mode** are all equal to one another.
- It is **symmetrical** about its **mean**.
- If the tails of the **normal distribution** are extended, they will run parallel to the horizontal axis without actually touching it. (**asymptotic** to the **x-axis**)
- The **normal distribution** has two parameters namely the **mean** μ and the **standard deviation** σ

Normal Probability Density Function

In the usual notation, the probability density function of the normal distribution is given below:

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

x is a continuous normal random variable with the property $-\infty < x < \infty$ meaning x can take all real numbers in the interval $-\infty < x < \infty$.

Standard Normal Distribution

INSTITUTE OF MANAGEMENT, I

The Standard Normal Variable is defined as follows:

$$Z = \frac{X - \mu}{\sigma}$$

Please note that Z is a pure number independent of the unit of measurement. The random variable Z follows a normal distribution with mean=0 and standard deviation =1.

$$f(Z) = \frac{1}{\sqrt{2\pi}} e^{-\left[\frac{Z^2}{2}\right]}$$

Example Problem

The mean weight of a morning breakfast cereal pack is 0.295 kg with a standard deviation of 0.025 kg. The random variable weight of the pack follows a normal distribution.

- a)What is the probability that the pack weighs less than 0.280 kg?
- b)What is the probability that the pack weighs more than 0.350 kg?
- c)What is the probability that the pack weighs between 0.260 kg to 0.340 kg?

Statistical Learning - Hypothesis Testing

Agenda

- Sampling distribution
- Central Limit Theorem
- Confidence intervals
- Hypothesis Formulation
- Null and Alternative Hypothesis
- Type I and Type II Errors
- Hypothesis Testing
 - One tailed v/s two tailed test
 - Test of mean
 - Test of proportion
 - Test of variance
- Examples

Concepts of sampling distribution

- Why do we need sampling?
- Analyse the sample and make inferences about the population
- Sample statistic vs population parameter
- Sampling distribution – distribution of a particular sample statistic of all possible samples that can be drawn from a population – sampling distribution of the mean

Sampling Distribution: CLT

- If n samples are drawn from a population that has a mean μ and standard deviation σ :
- The sampling distribution follows a normal distribution with:
- Mean: μ
- Standard Deviation: σ / \sqrt{n} (also c/a Standard Error)
- The corresponding z-score transformation is:

$$Z = \frac{\bar{x} - \mu}{\frac{\sigma}{\sqrt{n}}}$$

- If the population is normal, this holds true even for smaller sample sizes.
- However, if the population is not normal, this holds true for sufficiently large sample sizes.

Central Limit Theorem

- “Sampling Distribution of the mean of any independent random variable will be normal”
- This applies to both discrete and continuous distributions.
- The random variable should have a well defined mean and variance (standard deviation).
- Applicable even when the original variable is not normally distributed.
- Assumptions:
 - The data must be randomly sampled.
 - The samples values must be independent of each other.
 - The 10% condition: When the sample is drawn without replacement, the sample size n , should be no more than 10% of the population.
 - In general, a sample size of 30 is considered sufficient.
 - The sample size must be sufficiently large.
 - If the population is skewed, pretty large sample size is needed.
 - For a symmetric population, even small samples are acceptable.

Central Limit Theorem (*contd.*)

Assume a dice is rolled in sets of 4 trials and the faces are recorded. This is repeated for a month (30 days)

Sample	Throw 1	Throw 2	Throw 3	Throw 4	Mean
1	4	1	6	2	3.25
2	1	2	3	2	2
3	5	6	4	6	5.25
4	4	3	6	1	3.5
5	2	2	4	3	2.75
6	4	2	1	6	3.25
7	3	6	6	4	4.75
8	2	4	2	5	3.25
9	2	1	5	6	3.5
10	1	3	6	6	4
11	4	3	3	3	3.25
12	6	5	4	1	4
13	3	3	3	1	3.25
14	2	5	2	6	3.75
15	1	3	1	6	2.75

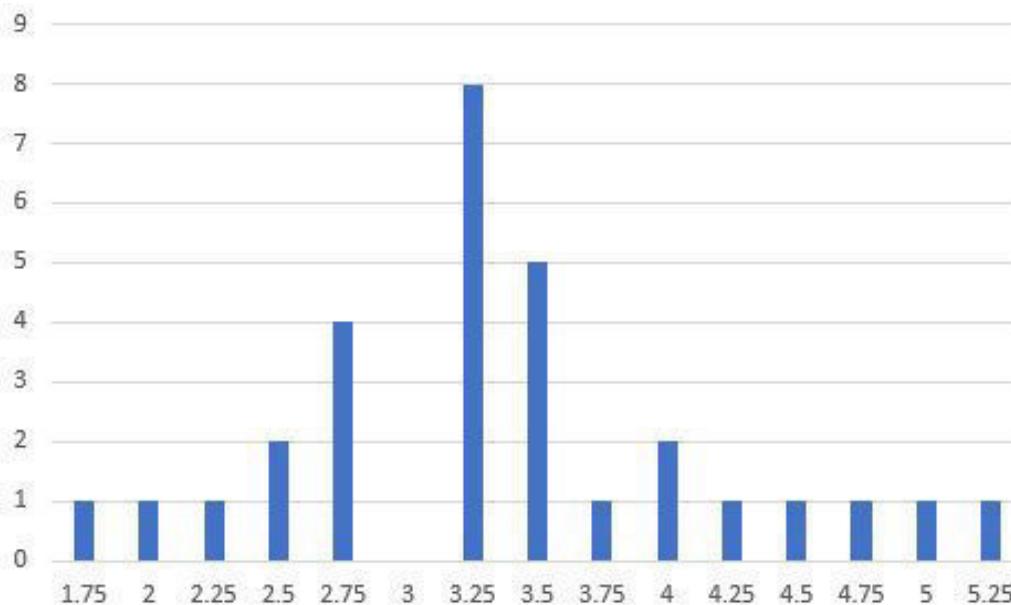
Sample	Throw 1	Throw 2	Throw 3	Throw 4	Mean
16	6	4	5	5	5
17	3	2	3	6	3.5
18	1	3	2	1	1.75
19	6	1	3	3	3.25
20	5	2	5	6	4.5
21	1	2	1	6	2.5
22	3	2	6	2	3.25
23	3	1	3	4	2.75
24	3	2	6	4	3.75
25	6	1	1	5	3.25
26	1	5	2	2	2.5
27	4	2	2	3	2.75
28	4	6	2	5	4.25
29	4	2	3	5	3.5
30	3	1	4	1	2.25

Central Limit Theorem (*contd.*)

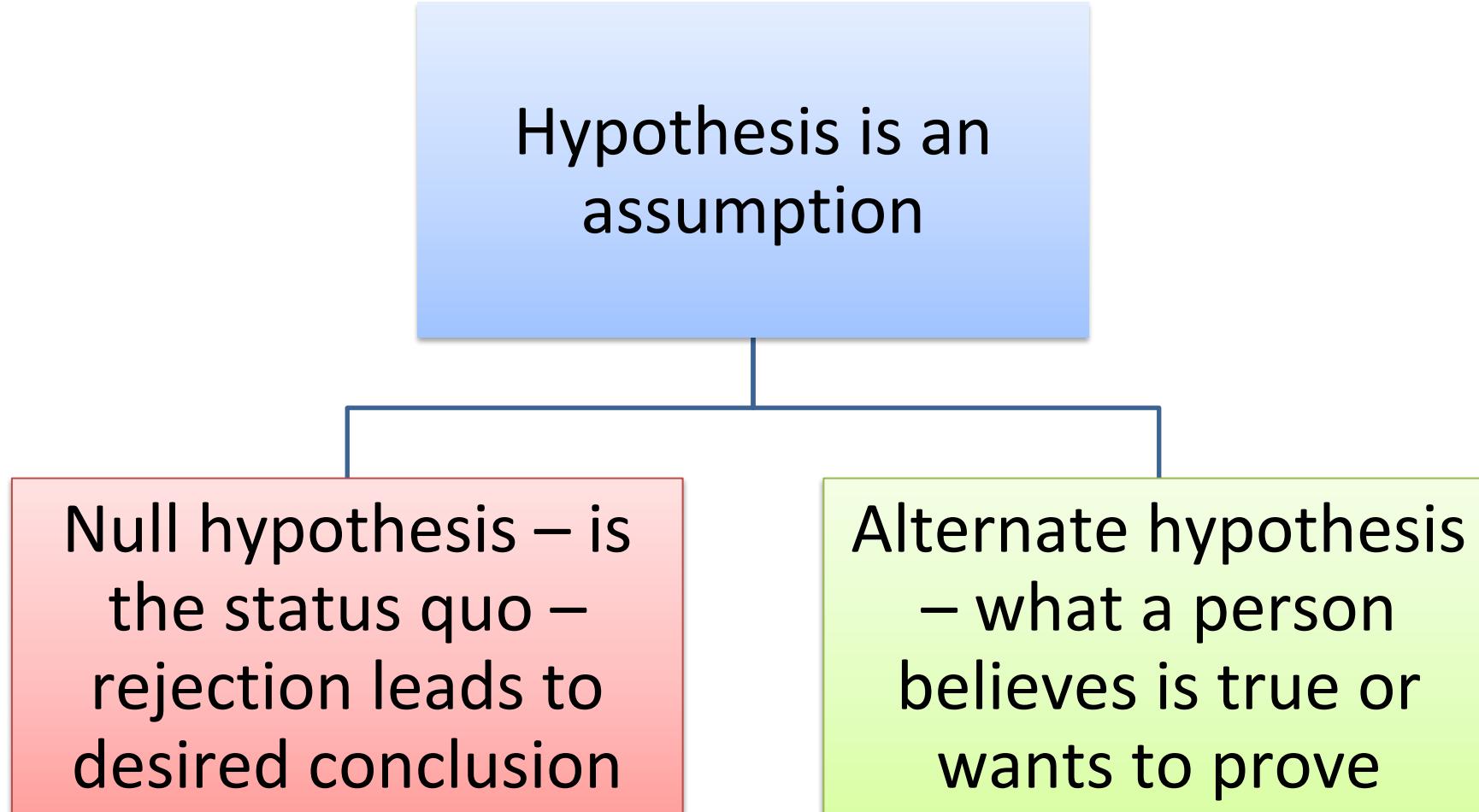
The means of the 30 samples are obtained are recorded in a frequency distribution table:

Mean	1.75	2	2.25	2.5	2.75	3	3.25	3.5	3.75	4	4.25	4.5	4.75	5	5.25
Frequency	1	1	1	2	4	0	8	5	1	2	1	1	1	1	1

Plotting the sample distribution of the sample mean, the following curve is obtained:



Hypothesis



Hypothesis Formulation

Coca Cola's most selling product is the 600ml coke or Coca Cola. Since the 600 ml info is on the label, we assume it to be true. But, is it actually true ?

As a customer, we're concerned that there is at least 600 ml in the bottle. If little more, we're okay.

Under-filling upsets the customers

On an average, is there at least 600 ml coke in every bottle?

Quantity \geq 600 ml
Quantity $<$ 600 ml

As a manufacturer, we would want the volume to be exactly 600 ml.

Overfilling results in higher costs of production.

On an average, is there exactly 600 ml coke in every bottle?

Quantity = 600 ml
Quantity $>$ 600 ml



Hypothesis Formulation (contd.)

collect 100 bottles from all over the country, so that we have a *random* sample.



Measure volume of each bottle in the sample to find the mean of 100 bottles.



Use sample mean to test assumption (status quo)

What is status quo in this scenario



Mean volume = 600ml



This is the key to inferential statistics: making inferences about the population from the sample.

Hypothesis Formulation (contd.)

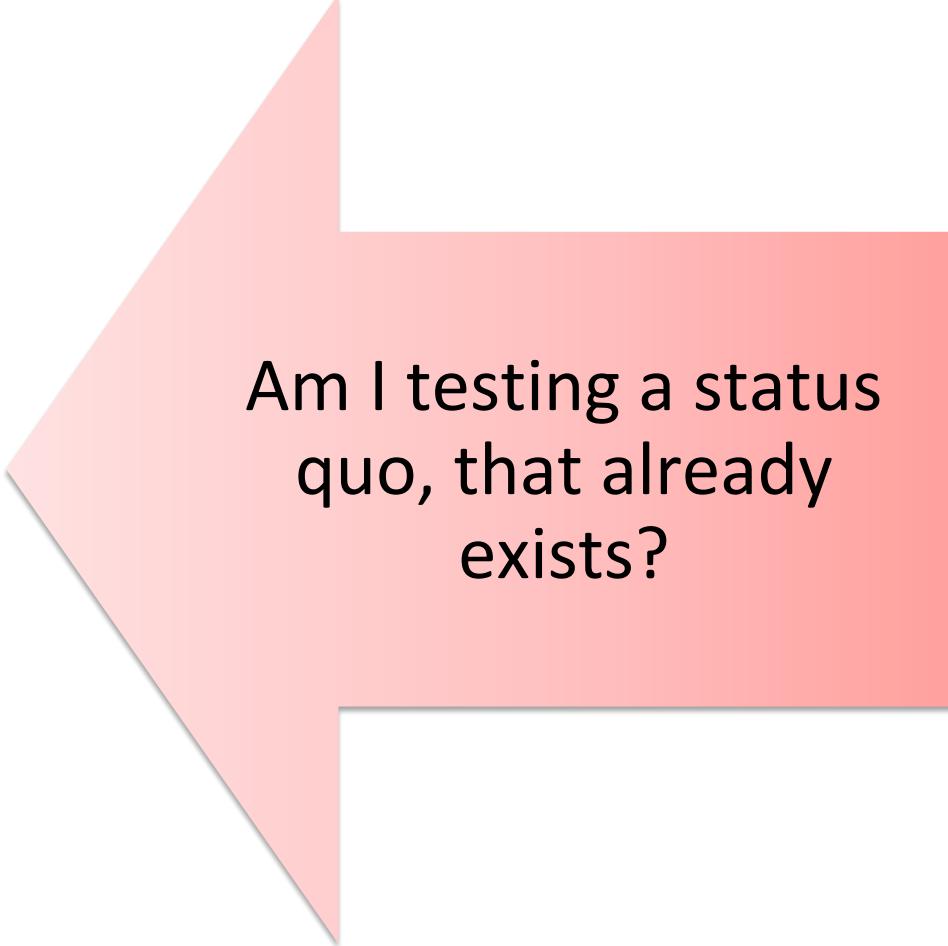
- Company claim: Volume > 600 ml (This may or may not be true)

What is the
claim or
assumption?

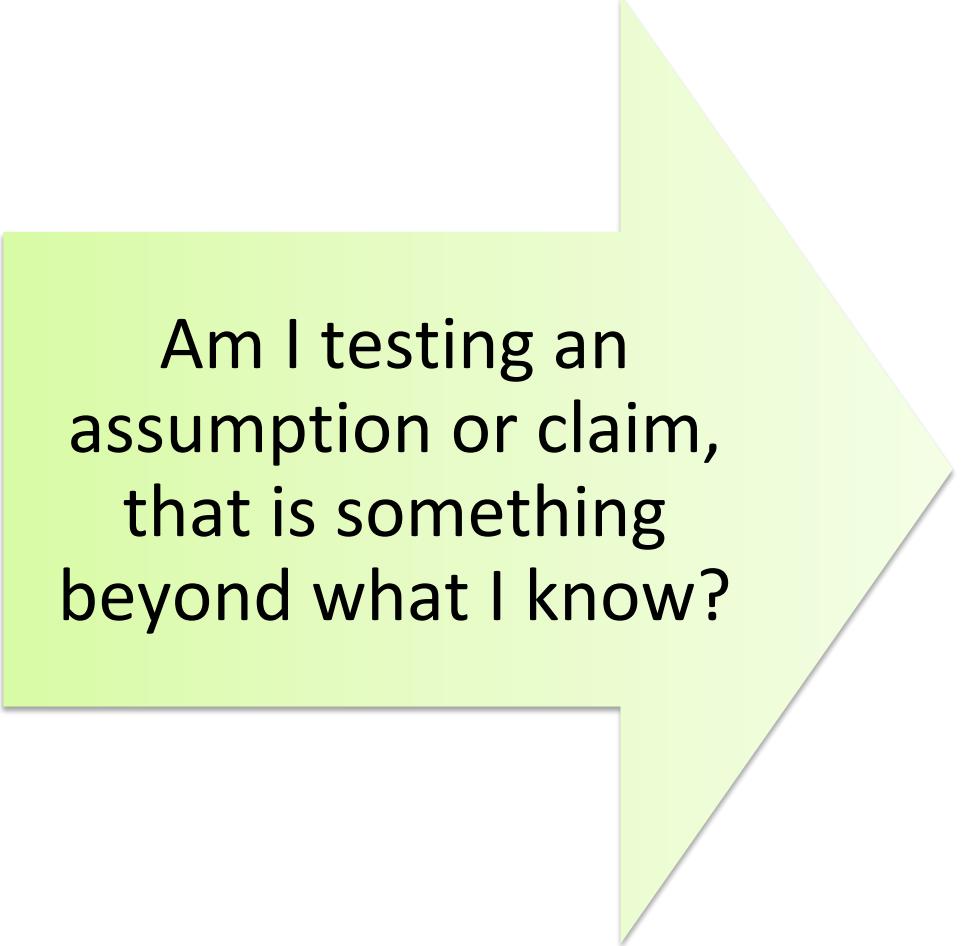


Mean volume
> 600ml

When formulating hypothesis...

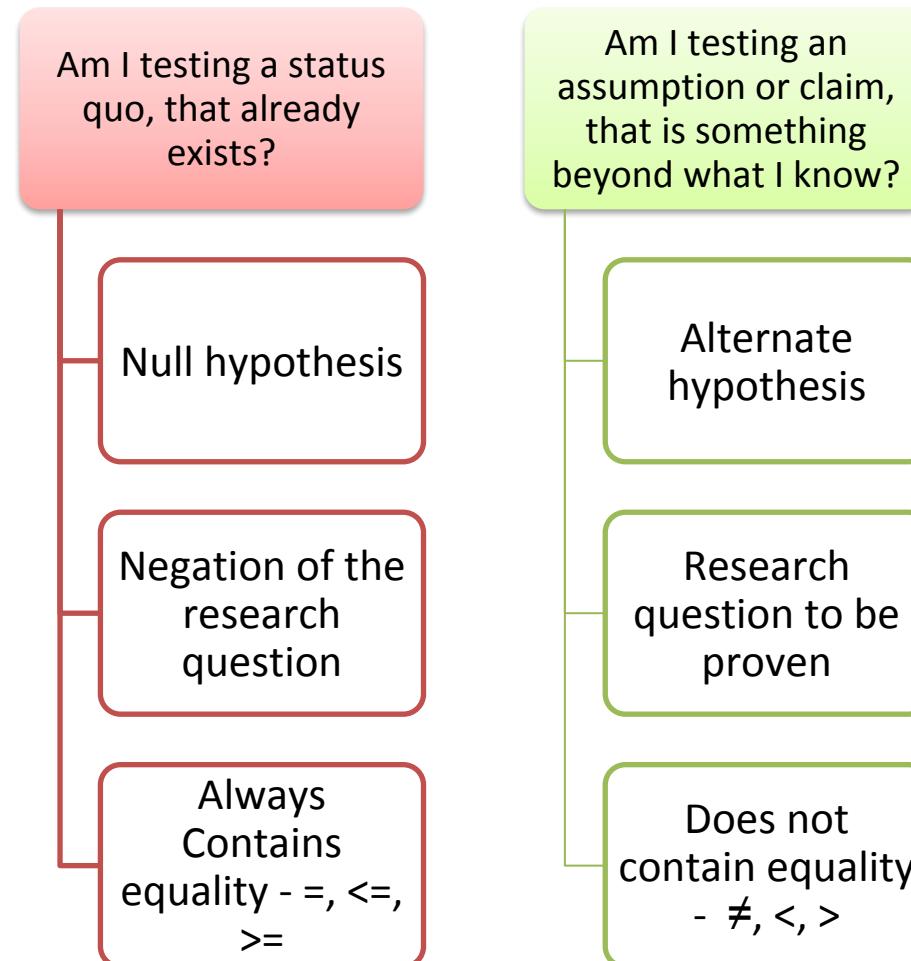


Am I testing a status quo, that already exists?



Am I testing an assumption or claim, that is something beyond what I know?

When formulating hypothesis...



Null and Alternative Hypothesis

All statistical conclusions are made in reference to the null hypothesis.

We either **reject** the null hypothesis or **fail to reject** the null hypothesis; we do not accept the null hypothesis.

From the start, we assume the null hypothesis to be true, later the assumption is rejected or we fail to reject it.

- When we **reject** the null hypothesis, we can conclude that the alternative hypothesis is supported.
- If we **fail to reject** the null hypothesis, it does not mean that we have proven the null hypothesis is true.
 - Failure to reject the null hypothesis does not equate to proving that it is true.
 - It just holds up our assumption or the status quo.

Null and Alternative Hypothesis: Example

Assumption: Volume = 600 ml

$$\Rightarrow H_0: \mu = 600 \text{ ml} \quad H_a: \mu \neq 600 \text{ ml}$$

Case 1: Data indicates that bottles are filled properly.

For example, $\mu = 600.4 \text{ ml}$

\Rightarrow We **fail to reject** the null hypothesis i.e. **fail to reject** the assumption.

We cannot say that the null is proved, we can just say that the assumption has held up.

Case 2: Data indicates that bottles are not filled properly

For example, $\mu = 645 \text{ ml}$

\Rightarrow We **reject** the null hypothesis i.e. **reject** our assumption.

We have statistical evidence to say that alterative hypothesis is valid or supported.

Type I and Type II Errors

Type I Error:

- Rejection of null hypothesis when it should not have been rejected.
- Incorrectly rejecting the null hypothesis.

Type II Error:

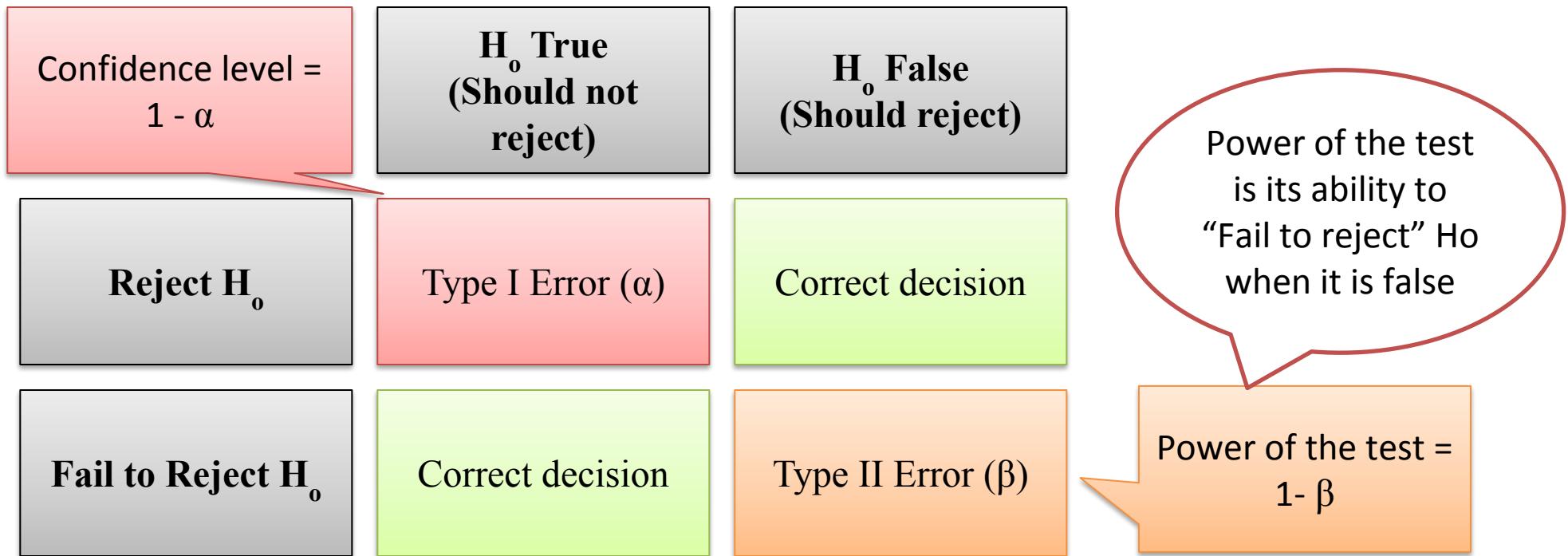
- Failure to reject the null hypothesis, when it should have been rejected.
- Incorrectly not rejecting the null hypothesis.

Decision/ Reality	H_0 True (Should not reject)	H_0 False (Should reject)
Reject H_0	Type I Error (α)	Correct Rejection (No error)
Fail to Reject H_0	Correct Decision (No error)	Type II Error (β)

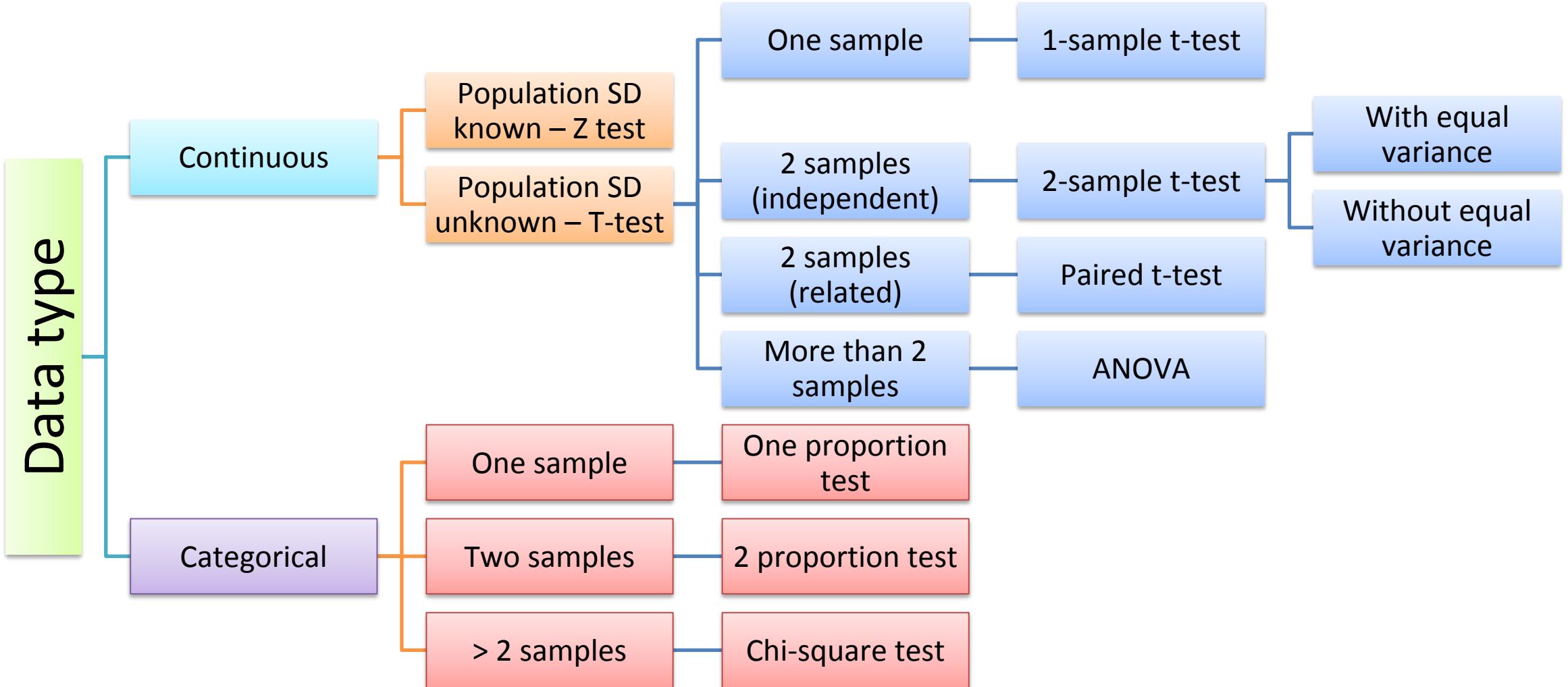
Causes of Type I and Type II Errors:

- By random chance, we may select a sample which is not representative of the population.
- Sampling techniques may be flawed.
- Assumptions in our null hypothesis may be flawed.

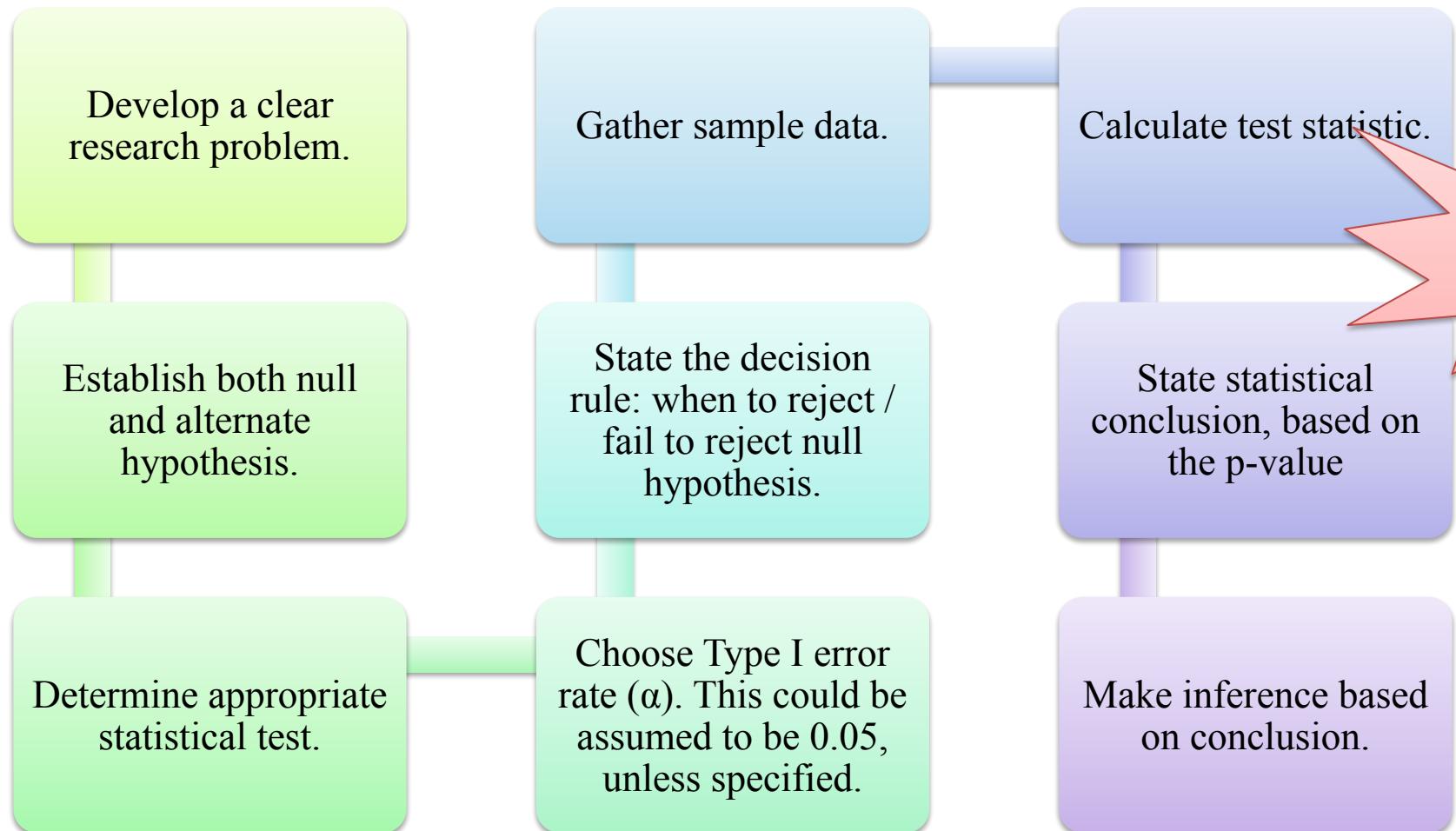
Type 1 & Type 2 errors



Hypothesis testing roadmap



Steps in hypothesis testing



**IF p is low,
NULL will GO**

Type of hypothesis tests

- Single sample or two or more samples
- One tailed or two tailed
- Tests of mean, proportion or variance

One tailed vs two tailed test

Case 1

- A customer complains that the mean volume is not equal to 600 ml
- What is H_0 ?
- What is H_a ?
- Is this one-tailed or two tailed?

Case 2

- Coca Cola official claims that the mean volume in coke bottles is more than 600ml
- What is H_0 ?
- What is H_a ?
- Is this one-tailed or two tailed?



One tailed vs two tailed test

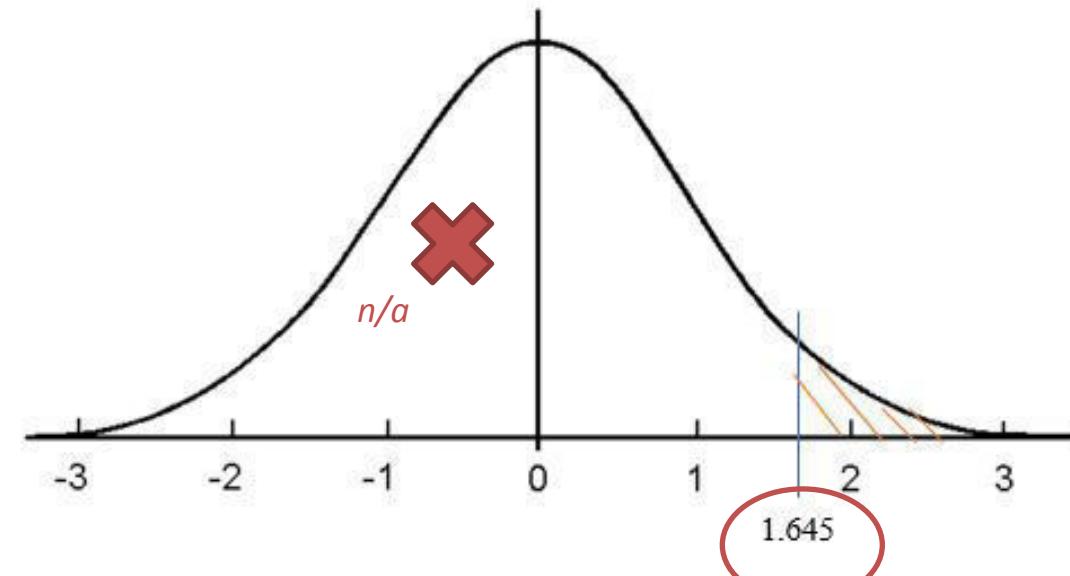
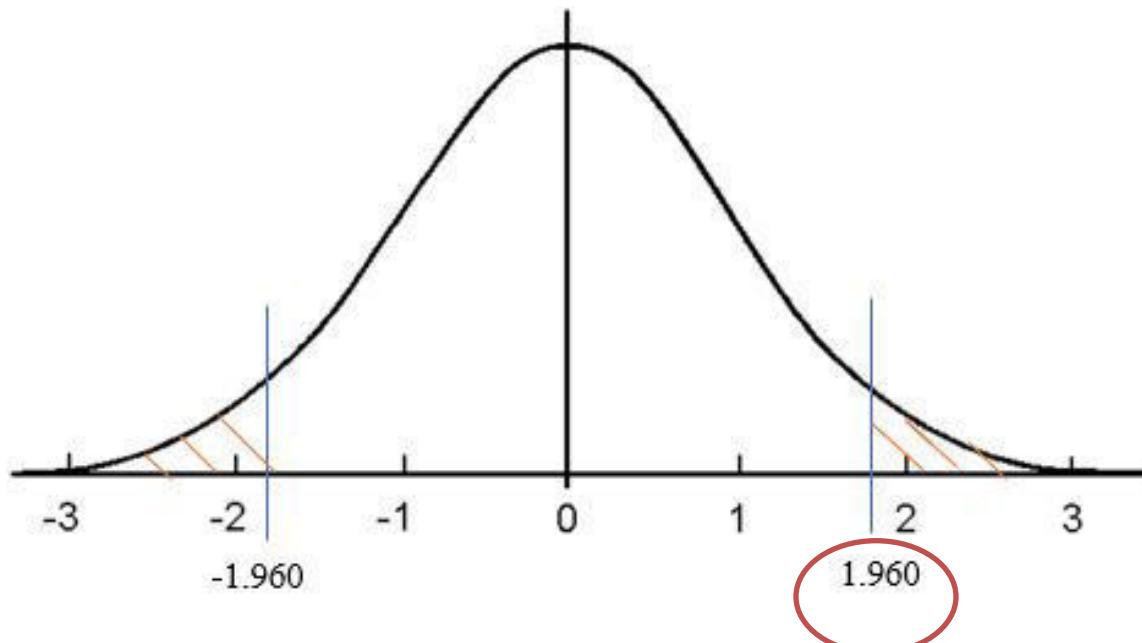
Case 1

- $H_0: \mu = 600\text{ml}$
- $H_a: \mu \neq 600\text{ml}$
- Two-tailed test

Case 2

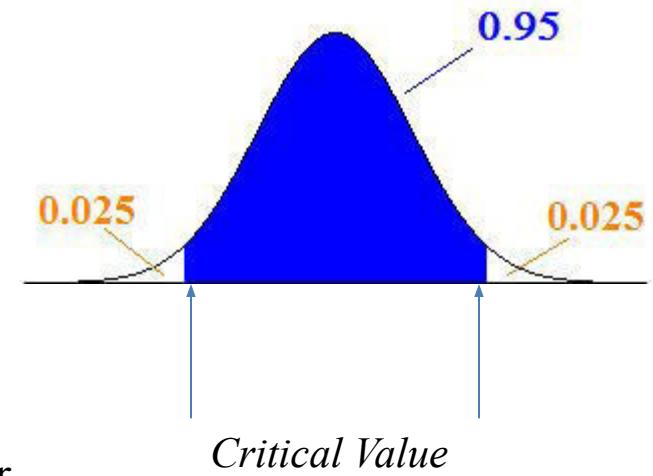
- $H_0: \mu \leq 600\text{ml}$
- $H_a: \mu > 600\text{ml}$
- One-tailed test

One tailed vs two tailed test



Confidence Intervals

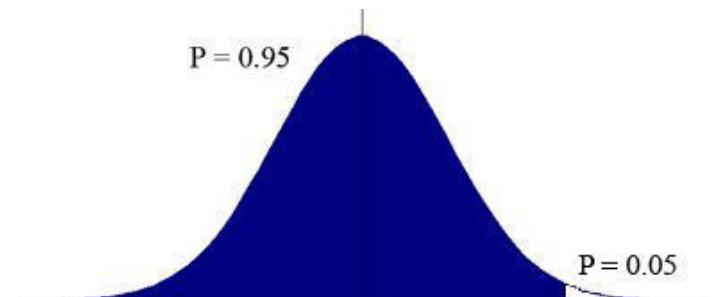
- 95% of all sample means (\bar{x}) are hypothesized to be in this region.
⇒ This is called as 95% confidence interval.
- If sample mean is in the blue region, we fail to reject the null hypothesis
- If sample mean is in the white region, we reject the null hypothesis.
- Here, $\alpha = 0.05$
⇒ α is the level of significance or our tolerance level towards making a Type I error.
- If the null hypothesis is correct, $(\alpha * 100)\%$ of the sample means should lie in the rejection region.



In case of one-tailed situation:

- All of α is in one tail or the other, depending on the alternative hypothesis.
- H_a points to the tail, where the critical value and the rejection region are.

(Case when observed mean > hypothesized mean)



Example – Confidence interval estimation

- A paper manufacturer has a production process that operates continuously. The paper is expected to have a mean length of 11 inches and a standard deviation of 0.02 inches. At periodic intervals, a sample is selected to determine whether the paper length is still equal to 11 inches. You select a random sample of 100 sheets and the mean paper length is 10.998 inches.
- Construct a 95% confidence interval.
- Construct a 99% confidence interval.

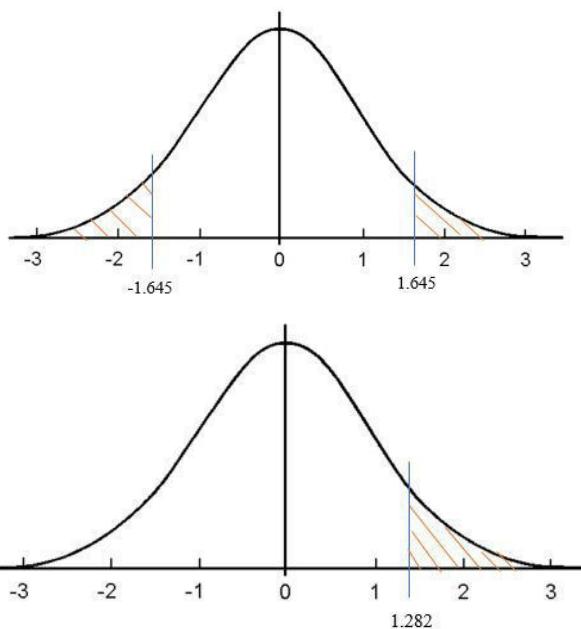
$$C.I. = \bar{X} \pm Z_{\frac{\alpha}{2}} \frac{\sigma}{\sqrt{n}}$$

Single Sample z – test of mean (*known* σ)

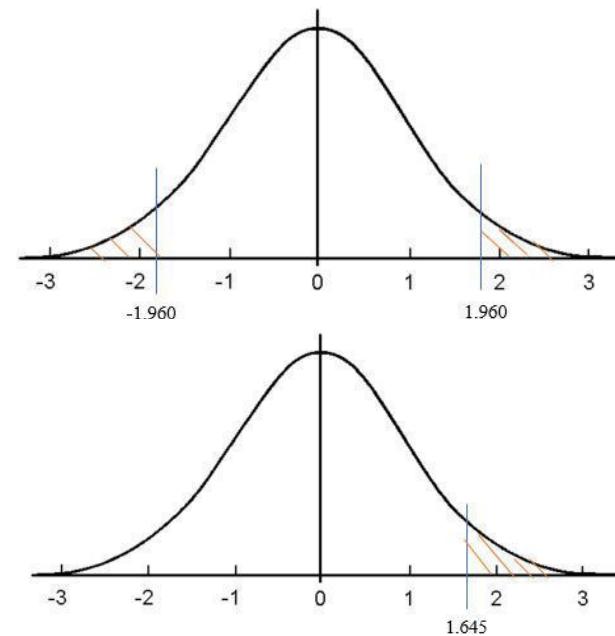
Test Statistic: $z = \frac{\bar{x} - \mu}{\sigma / \sqrt{n}}$

p-value: How much of the area is above the test-statistic? (*Does test statistic fall in the rejection region?*)
If it is less than the specific α , we reject the null hypothesis

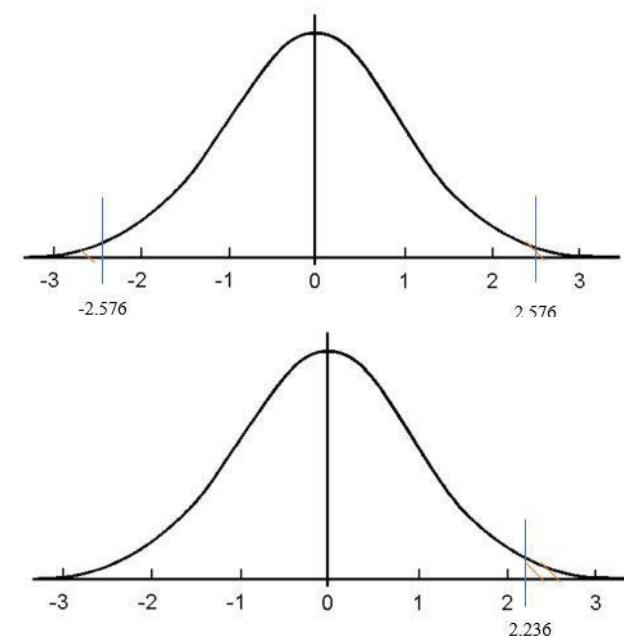
$$\alpha = 0.10$$



$$\alpha = 0.05$$



$$\alpha = 0.01$$



Example problem - Single Sample z – test of mean

- You are the manager of a fast food restaurant. You want to determine if the population mean waiting time has changed from the 4.5 minutes. You can assume that the population standard deviation is 1.2 minutes. You select a sample of 25 orders in an hour. Sample mean is 5.1 minutes. Use the relevant hypothesis test to determine if the population mean has changed from the past value of 4.5.

Steps to solve the problem...

- One-tailed or two-tailed
- What is H_0 and H_a
- Determine Z and Z_{stat}
- Draw the normal curve
- Reject/Fail to reject H_0 ?

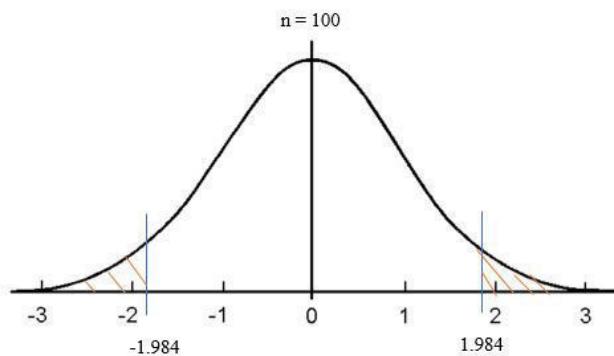
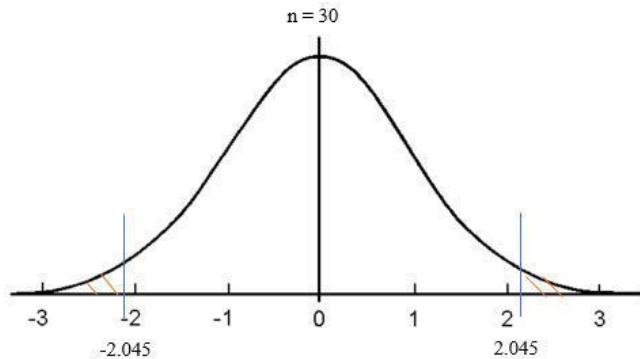
Single Sample t – test of mean (*unknown* σ)

Test Statistic: $t = \frac{\bar{x} - \mu}{s/\sqrt{n}}$

p-value: How much of the area is above the test-statistic? (*Does test statistic fall in the rejection region?*)
If it is less than the specific α , we reject the null hypothesis

* *t-statistic depends on the sample size*

$$\alpha = 0.05$$



Two sample tests of mean

To understand if the mean volume in coke bottle is 600ml, we decide to take two samples from two manufacturing centers.

The assumption would be that the mean difference between the two samples would be zero:
i.e. $\mu_1 = \mu_2 \Rightarrow \mu_1 - \mu_2 = 0$ (*Null hypothesis*)



- When σ is known, use z-distribution

$$z = \frac{(\bar{x}_1 - \bar{x}_2) - D_0}{\sqrt{\left(\frac{\sigma_1^2}{n} + \frac{\sigma_2^2}{n}\right)}} \text{ standardized mean b/w two means (zero in the above example)}$$

- When σ is not known, use t-distribution

$$t = \frac{\bar{x}_1 - \bar{x}_2}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}} \text{ here, } df \text{ is calculated as:}$$

$$df = \left[\frac{\left(\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2} \right)^2}{\frac{s_1^4}{n_1^2(n_1-1)} + \frac{s_2^4}{n_2^2(n_2-1)}} \right]$$

Example – 2 sample t-test of mean

A hotel manager is concerned with increasing the return rate of customers. One aspect that affects this is the time taken to deliver luggage to the guest's room after check-in. A random sample of 20 deliveries were selected from Wing A and Wing B of the hotel. Analyse whether there is a difference in the average time taken by the 2 Wings?

Matched Sample / Paired t-test of mean

It is reported that the caffeine in coke had increased the permissible limit because of manufacturing issues.

A sample of 100 bottles taken reports the average caffeine to be 10.5 µg (Permissible level is 10 µg)

Coca Cola technicians derive a technique using which they would correct caffeine levels in the coke bottles, rather than having to throw them away.

The 100 bottles are made to undergo this technique and caffeine levels are measured in the same bottles.



The t-statistic here is calculated as:

$$t = \frac{\bar{d} - \mu_d}{\frac{s_d}{\sqrt{n}}}$$

Mean difference, μ_d = hypothesized difference (*usually 0*)
Standard deviation of the difference

Bottle	Caffeine before	Caffeine after	Difference
1	10.4	10.2	0.2
2	10.8	10.5	0.3
3	9.8	10	-0.2
...			Calculate \bar{d} and s_d

Example -Paired t-test of mean

- The data represents the compressive strength of 40 samples taken 2 days and 7 days after pouring.
 - AT 0.01 level of significance, is there evidence that the mean strength is lower at 2 days than at 7 days?

z-test of Proportion

In 2010, Coca Cola officials noted that 30% of the bottles were under-filled. They took corrective measures. 5 years later, they sampled 300 bottles and determined that 76 of them were under-filled. At 5% significance level, is this evidence sufficient to show the impact of corrective measures?



$$H_0: p_0 = 0.30$$

$$H_a: p_0 < 0.30$$

$$\hat{p} = 76/300$$

For one sample,

$$z_0 = \frac{\hat{p} - p_0}{\sqrt{\frac{p_0(1-p_0)}{n}}}$$

For two samples,

$$z = \frac{(\hat{p}_1 - \hat{p}_2)}{\sqrt{\hat{p}(1-\hat{p})} \sqrt{\frac{1}{n_1} + \frac{1}{n_2}}}$$

p-value: How much of the area is above the test-statistic? (*Does test statistic fall in the rejection region?*)

If it is less than the specific α , we reject the null hypothesis

Example – One tailed test of proportions

There are 155 banks involved in certain international transactions. A federal agency claims that at least 35% of these banks have total assets of over \$10 billion (In U.S. dollars). An independent agency wants to test this claim. It gets a random sample of 50 out of the 155 banks and finds that 15 of them have total assets of over \$10 billion. Can the claim be rejected?

Test of variance

5 bottles from Manufacturer 1 show the following quantities:

- 607ml, 602ml, 590ml, 603ml, 598ml

5 bottles from Manufacturer 2 show the following quantities:

- 602ml, 597ml, 600ml, 603ml, 598ml

Case 1:

One of the two manufactures contract should be renewed at the end of the year.

Which one do you think should be renewed ? First, second or both ?



Case 2:

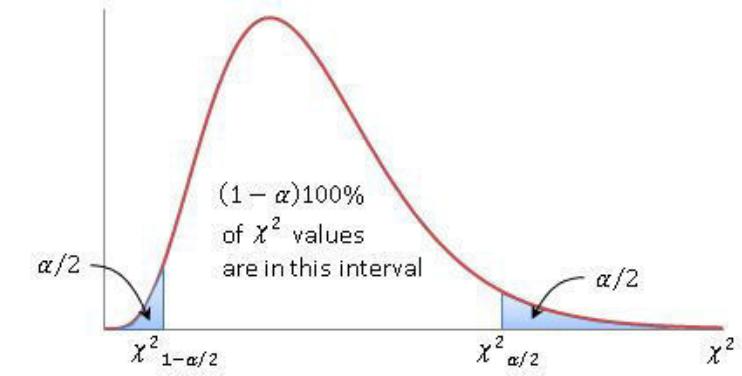
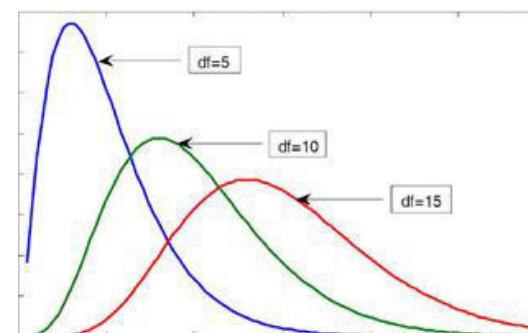
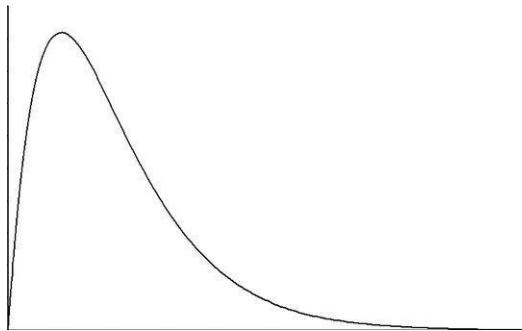
The audit teams wants to ensure that the production patterns should remain equivalent across all manufacturers.

Do you think the two manufactures qualify this constraint ?

Chi square test of variance

When we take many samples of the same size from a normal population and find the sample means, they follow a normal distribution.

When we take many samples of the same size from a normal population and find the sample variances, they DO NOT follow a normal distribution; instead they follow a **chi-square (χ^2) distribution**, which is dependent on the degrees of freedom.



- Area under the curve is always 1.
- Cumulative Probability runs from right to left; 1 is towards the left end, while 0 is towards the right.

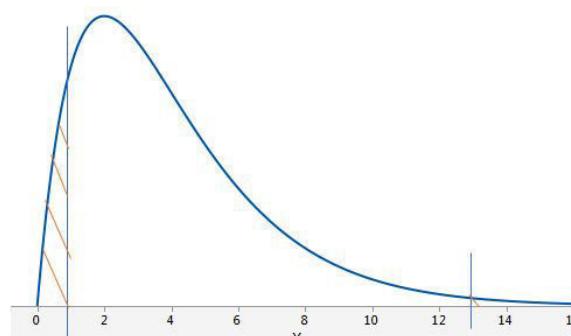
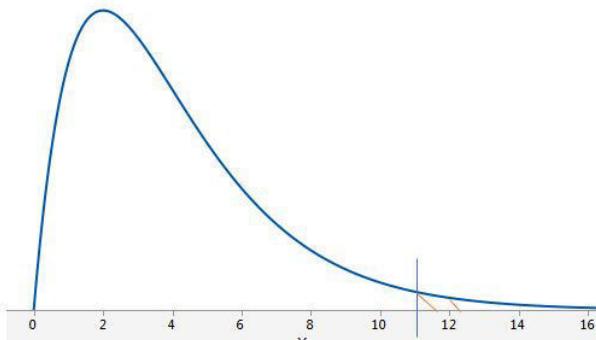
Chi square test of variance

Chi-square (χ^2) test compares the population variance, with the hypothesized variance.

$$\chi^2 = \frac{(n-1) s^2}{\sigma^2} \quad \text{where, } n = \text{sample size}$$

s^2 = sample variance and σ^2 = population variance (which we wish to test)

At $\alpha = 0.05$ and $n = 5$ ($df = 4$)



p-value: How much of the area is above the test-statistic? (*Does test statistic fall in the rejection region?*)
If it is less than the specific α , we reject the null hypothesis

Example – chi-squared test

When new paperback novels are promoted at bookstores, a display is often arranged with copies of the same book with differently colored covers. A publishing house wanted to find out whether there is a dependence between the place where the book is sold and the color of its cover. For one of its latest novels, the publisher sent displays and a supply of copies of the novel to large bookstores in five major cities. The resulting sales of the novel for each city-color combination are given. Numbers are in thousands of copies sold over a three-month period.

F-ratio test of variance

When two independent random samples are taken from normal population(s) with equal variances, the sampling distribution of the ratio of those sample variances follows an F distribution.

- Test of equality of variances: comparison of two sample variances
- The variances are compared using a ratio:

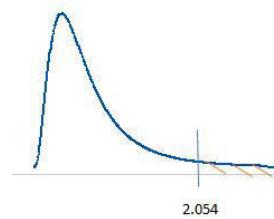
$$F = \frac{s_x^2}{s_y^2} \quad \text{where, } s_x^2 \text{ is the larger sample variance while } s_y^2 \text{ is the smaller sample variance}$$

(Both numerator and denominator have their individual dfs)

F-distribution is only right-tailed:

$$H_0: \sigma_x^2 = \sigma_y^2 \quad H_a: \sigma_x^2 \neq \sigma_y^2$$

For $\alpha = 0.05$ and $df1 = 24$, $df2 = 21$



p-value: How much of the area is above the test-statistic? (*Does test statistic fall in the rejection region?*)

If it is less than the specific α , we reject the null hypothesis *Use FDIST function

Example - F-test

An important measure of the risk associated with a stock is the standard deviation, or variance, of the stock's price movements. A financial analyst wants to test the one-tailed hypothesis that stock A has a greater risk (larger variance of price) than stock B. A random sample of 25 daily prices of stock A gives $s^2_A=6.52$, and a random sample of 22 daily prices of stock B gives a sample variance of $s^2_B=3.47$. Carry out the test at $\alpha=0.01$.

Hypothesis Tests using Python

z-test

```
statsmodels.stats.weightstats.ztest(x1, x2=None, value=0,  
alternative='two-sided')
```

Link to refer -

<https://www.statsmodels.org/stable/generated/statsmodels.stats.weightstats.ztest.html>

t-test

```
scipy.stats.ttest_ind(a, b, axis=0, equal_var=True, nan_policy='propagate')
```

Link to refer -

https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.ttest_ind.html

Chi-square (χ^2) test

```
scipy.stats.chisquare(f_obs, f_exp=None)
```

Link to refer

<https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.chisquare.html>

F-test

```
alpha = 0.05 #Or whatever you want your alpha to be.  
p_value = scipy.stats.f.cdf(F, df1, df2)  
if p_value > alpha: # Reject the null hypothesis that Var(X) == Var(Y)
```

Link to refer - <https://stackoverflow.com/questions/21494141/how-do-i-do-a-f-test-in-python>

Hypothesis Testing Using Python

One Sample Testing

Some important functions:

1. `t_statistic, p_value = ttest_1samp(array, n)`

Here n= sample number , daily_intake= array

2. `z_statistic, p_value = wilcoxon(array - n)`

The Wilcoxon test is used if the data is demonstrably not normally distributed.

Hypothesis Testing Using Python

Two Sample Testing

Some important functions:

1. `t_statistic, p_value = ttest_ind(group1, group2)`
2. `u, p_value = mannwhitneyu(group1, group2)`
3. `t_statistic, p_value = ttest_1samp(post-pre, 0)`
4. `z_statistic, p_value = wilcoxon(post-pre)`
5. `levene(pre,post)`
6. `shapiro(post)`

ANOVA- One Way Classification

- The samples drawn from different populations are independent and random.
- The response variables of all the populations are normally distributed.
- The variances of all the populations are equal.

Hypothesis of One-Way ANOVA

- $H_0: \mu_1 = \mu_2 = \mu_3 = \mu_4 = \dots = \mu_k$
 - All population means are equal
- H_1 : Not all of the population means are equal
 - For at least one pair, the population means are unequal.

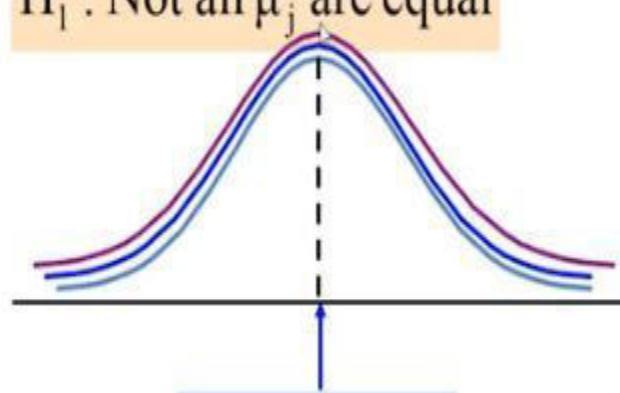
One-Way ANOVA

One-Way ANOVA

Null Hypothesis(H_0 =True)

$$H_0 : \mu_1 = \mu_2 = \mu_3 = \dots = \mu_k$$

$$H_1 : \text{Not all } \mu_j \text{ are equal}$$



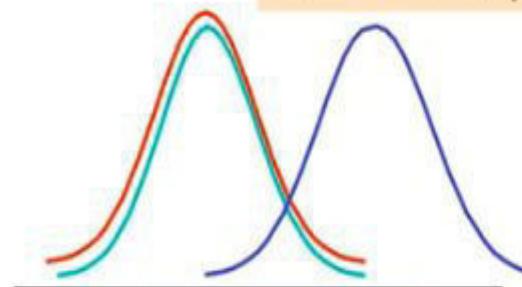
$$\mu_1 = \mu_2 = \mu_3$$

One-Way ANOVA

Alternative Hypothesis(H_1 =True)

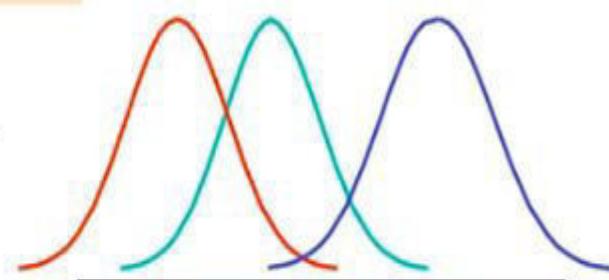
$$H_0 : \mu_1 = \mu_2 = \mu_3 = \dots = \mu_k$$

$$H_1 : \text{Not all } \mu_j \text{ are equal}$$



$$\mu_1 = \mu_2 \neq \mu_3$$

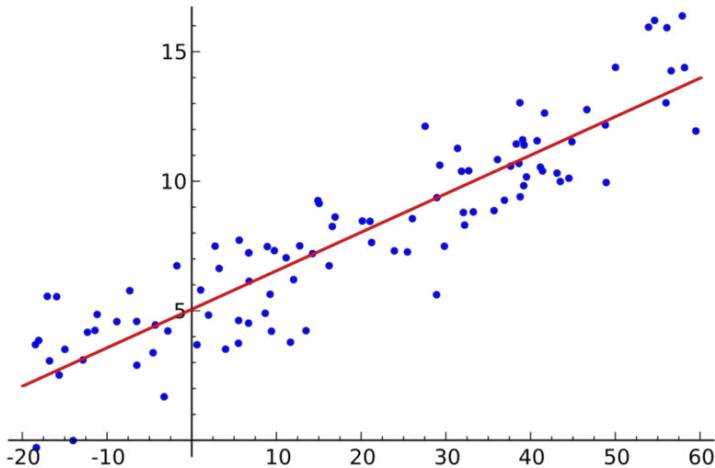
or



$$\mu_1 \neq \mu_2 \neq \mu_3$$

Linear Regression

Before knowing what is linear regression, let us get ourselves accustomed to regression. Regression is a method of modelling a target value based on independent predictors. This method is mostly used for forecasting and finding out cause and effect relationship between variables. Regression techniques mostly differ based on the number of independent variables and the type of relationship between the independent and dependent variables.



Simple linear regression is a type of regression analysis where the number of independent variables is one and there is a linear relationship between the independent(x) and dependent(y) variable. The red line in the above graph is referred to as the best fit straight line. Based on the given data points, we try to plot a line that models the points the best.

The line can be modelled based on the linear equation shown below.

$$Y = \beta_0 + \beta_1 x + \epsilon$$

Isn't Linear Regression from Statistics?

Before we dive into the details of linear regression, you may be asking yourself why we are looking at this algorithm.

Isn't it a technique from statistics?

Machine learning, more specifically the field of predictive modelling is primarily concerned with minimizing the error of a model or making the most accurate predictions possible, at the expense of explainability. In applied machine learning we will borrow, reuse and steal algorithms from many different fields, including statistics and use them towards these ends.

As such, linear regression was developed in the field of statistics and is studied as a model for understanding the relationship between input and output numerical variables, but has been borrowed by machine learning. It is both a statistical algorithm and a machine learning algorithm.

Linear Regression Model Representation

Linear regression is an attractive model because the representation is so simple.

The representation is a linear equation that combines a specific set of input values (x) the solution to which is the predicted output for that set of input values (y). As such, both the input values (x) and the output value are numeric.

The linear equation assigns one scale factor to each input value or column, called a coefficient and represented by the capital Greek letter Beta (β). One additional coefficient is also added, giving the line an additional degree of freedom (e.g. moving up and down on a twodimensional plot) and is often called the intercept or the bias coefficient.

For example, in a simple regression problem (a single x and a single y), the form of the model would be:

$$Y = \beta_0 + \beta_1 x$$

In higher dimensions when we have more than one input (x), the line is called a plane or a hyper-plane. The representation therefore is the form of the equation and the specific values used for the coefficients (e.g. β_0 and β_1 in the above example).

Linear Regression Learning the Model

1. Simple Linear Regression

With simple linear regression when we have a single input, we can use statistics to estimate the coefficients.

This requires that you calculate statistical properties from the data such as means, standard deviations, correlations and covariance. All of the data must be available to traverse and calculate statistics.

2. Ordinary Least Squares

When we have more than one input we can use Ordinary Least Squares to estimate the values of the coefficients.

The Ordinary Least Squares procedure seeks to minimize the sum of the squared residuals. This means that given a regression line through the data we calculate the distance from each data point to the regression line, square it, and sum all of the squared errors together. This is the quantity that ordinary least squares seek to minimize.

3. Gradient Descent

This operation is called Gradient Descent and works by starting with random values for each coefficient. The sum of the squared errors is calculated for each pair of input and output values. A learning rate is used as a scale factor and the coefficients are updated in the direction towards minimizing the error. The process is repeated until a minimum sum squared error is achieved or no further improvement is possible.

When using this method, you must select a learning rate (alpha) parameter that determines the size of the improvement step to take on each iteration of the procedure.

4. Regularization

There are extensions of the training of the linear model called regularization methods. These seek to both minimize the sum of the squared error of the model on the training data (using ordinary least squares) but also to reduce the complexity of the model (like the number or absolute size of the sum of all coefficients in the model).

Two popular examples of regularization procedures for linear regression are:

- **Lasso Regression:** where Ordinary Least Squares is modified to also minimize the absolute sum of the coefficients (called L1 regularization).
- **Ridge Regression:** where Ordinary Least Squares is modified to also minimize the squared absolute sum of the coefficients (called L2 regularization).

Preparing Data for Linear Regression

Linear regression has been studied at great length, and there is a lot of literature on how your data must be structured to make best use of the model. In practice, you can use these rules more as rules of thumb when using Ordinary Least Squares Regression, the most common implementation of linear regression.

Try different preparations of your data using these heuristics and see what works best for your problem.

- Linear Assumption
- Remove Noise
- Remove Collinearity
- Gaussian Distributions
- Rescale Inputs

Summary

In this post you discovered the linear regression algorithm for machine learning.

You covered a lot of ground including:

- The common names used when describing linear regression models.
- The representation used by the model.
- Learning algorithms used to estimate the coefficients in the model.
- Rules of thumb to consider when preparing data for use with linear regression.

Try out linear regression and get comfortable with it.

Credits: <https://towardsdatascience.com/introduction-to-machine-learning-algorithms-linear-regression-14c4e325882a>

Linear Regression

Theory

Regression Model Assumptions

Assumption 1

Linear Regression Model is linear in the parameters though it may not be linear in the variables i.e. the slope coefficients are always raised to power 1. The variables may be raised to any power. The regression model thus, takes the form $y_i = \beta_1 + \beta_2 X_i + u_i$ (β_1 is intercept, β_2 is coefficient, X_i is variable, u_i is disturbance)

The conditional expectation of y , $E(y | X_i)$ is a linear function of the parameters i.e. the β s. y is linearly related to X when the rate of change of y with respect to X (i.e. slope or derivative of y with respect to X , dy/dx) is independent of the value of X . For e.g.

1. if $y = 10x$ then $dy/dx = 10$, which is independent of x i.e. for all values of x , dy/dx is constant.
2. If $y = 10x^2$ then $dy/dx = 20x$ i.e. the rate of change of y depends on the current value of x . It is dependent on X . Hence the function is not linear in X

The deviation u_i in each X_i prediction can be positive or negative. Technically u_i is known as stochastic disturbance or stochastic error term

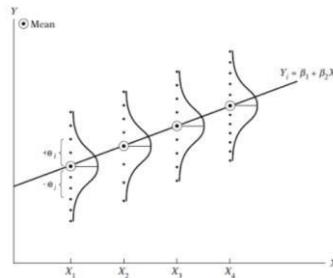
Regression Model Assumptions

Assumption 2

1. X values are independent of the error term. Values taken by the regressor X may be considered fixed in repeated trials / sample. The error in prediction of each trial is independent of the value of X.
2. Error term u_i , represents the impact of the variables not considered for the model. Since the assumption that X predictors are independent of one another, it applies to even those variables not considered

Assumption 3

1. The mean value of disturbance u_i is zero. Given the value of X_i , the means or expected value of the random disturbance $E(u_i | X_i) = 0$ i.e. $E(u_i) = 0$
2. Population of y corresponding to a given X_i is distributed around its mean value, implies no specification bias / error in the model indicating that the model is correctly specified.



3. Specification error results from leaving out important variables or choosing wrong functional form to express relationship between y and X

Regression Model Assumptions

Assumption 4

1. Homoscedasticity or Constant Variance of u_i , the variance of the error / disturbance is the same regardless of the value of X
2. $\text{Var}(u_i) = E[u_i - E(u_i | X_i)]^2 = E(u_i^2 | X_i)$ because of assumption 3 ($E(u_i) = 0$).
3. $= E(u_i^2)$ for a given X_i = constant variance (representation for homoscedasticity)

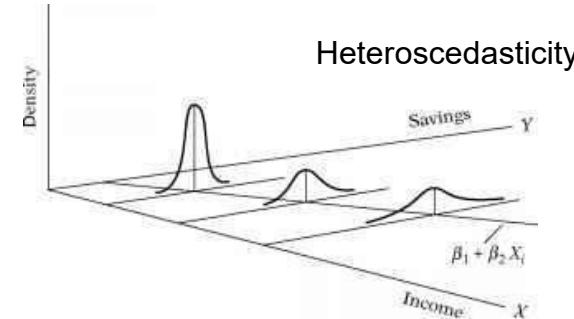
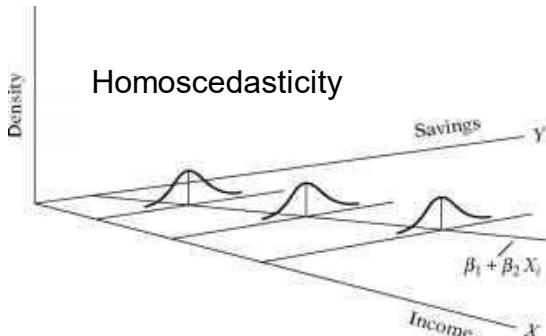


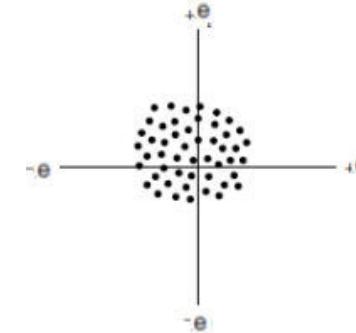
Image source: <https://www.rhayden.us/regression-models/the-nature-of-heteroscedasticity.html>

4. The likelihood that the y observations coming from the population with $X = X_i$ would be closer to the population regression function than those coming from the populations corresponding to $X = X_2, X = X_3$ and so on. The reliability of predicted Y will fall
5. By invoking Assumption 4, we stress equal importance to all y values corresponding to different values of X

Regression Model Assumptions

Assumption 5

1. No autocorrelation between disturbances u_i . Given any two X values, X_i and X_j ($i \neq j$), the correlation between any two u_i and u_j is zero i.e. no serial or auto correlation
2. This assumption is justified when time is not an attribute i.e. the trials / records are not generated in any time-series fashion



Assumption 6

1. The number of observations n must be greater than the number of parameters to be estimated. In data science parlance, the depth should be much greater than breadth i.e. number of records much larger than the number of columns to avoid curse of dimensionality situation.

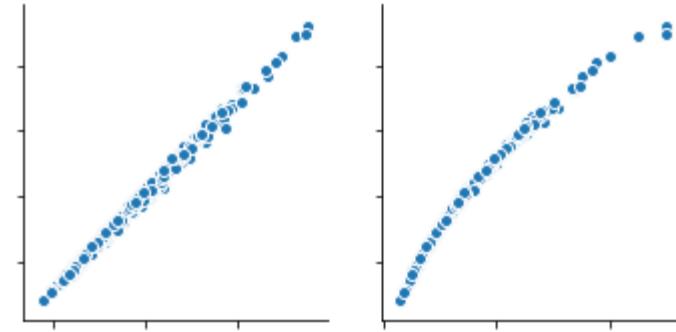
Assumption 7

1. The X should have variance. The values should not be constant. In Data Science parlance, X should have variance. Further the outliers should not exist

Regression Model Assumptions

Assumption 8

1. There no perfect collinearity between the predictor variables X
2. In case of perfect collinearity, the scatter plot will be line
3. Most often we come across less than perfect collinearity



Assumption 9

1. The model is correctly specified i.e. neither overfit or underfit

Assumption 10

The stochastic term u_i is normally distributed. The error term 'e' follows the normal distribution with zero mean and (constant) variance $u_i \sim N(0, \sigma^2)$

where the symbol \sim means distributed as and N stands for the normal distribution, the terms in the parentheses representing the two parameters of the normal distribution, namely, the mean and the variance. If this assumption is violated, the statistical tests such as t, and F in regression may not be valid.

Note: All the assumptions pertain to population regression function only.

Significance of stochastic disturbance term

1. The disturbance term u_i is a surrogate for the variables that are left out of the model but have a collective impact on the output y .
2. The reason why those variables were left out could be many
3. We may not fully understand how those variables impact the output (theoretically weak)
4. Lack of data for those variables. Some variables are not quantitative by nature and we may not have a way to capture such data for e.g. personality of an individual that impacts his/her monthly expenses
5. Peripheral variables – Some variables have a weak influence on the target and their joint influence may be very weak. Such variables can be represented by the u_i
6. Intrinsic randomness in the process. For e.g. personality of individuals may vary significantly even when the most of the measurable attributes are same
7. Principle of parsimony requires that we keep our models as simple as possible (Occam's razor). If significant part of y 's behavior can be captured by a few variables, then why not keep it simple. Let the other variables collective effort be represented by the u_i

Disturbance term expected value

Assumptions 3 (The mean value of disturbance u_i is zero) Why?

1. Let linear regression model be $y_i = \beta_1 + \beta_2X_{2i} + \beta_3X_{3i} + \dots + \beta_kX_{ki} + u_i$ (a data point in K dimensions)
2. Assume $E(U_i | X_{2i}, X_{3i}, \dots, X_{ki}) = W$ (W is a constant , in standard model $W = 0$)
3. Conditional expectations of the equation for y_i can be expressed as
 - a. $E(y_i | X_{2i}, X_{3i}, \dots, X_{ki}) = \beta_1 + \beta_2X_{2i} + \beta_3X_{3i} + \dots + \beta_kX_{ki} + W$
 - b. $\Rightarrow (\beta_1 + W) + \beta_2X_{2i} + \beta_3X_{3i} + \dots + \beta_kX_{ki}$
 - c. $\Rightarrow \alpha + \beta_2X_{2i} + \beta_3X_{3i} + \dots + \beta_kX_{ki}$ where $\alpha = (\beta_1 + W)$
4. Given the training data, the X s are treated as constant while the β s are the variables
5. **If the assumption 3 is not fulfilled, we cannot solve the equation for β_1 !**

Heteroscedasticity of disturbance

Homoscedasticity or Constant Variance of u_i , the variance of the error / disturbance is the same regardless of the value of X .

Violation of this assumption leads to Heteroscedasticity. There are several reasons for this –

1. As the processes mature and stabilize over a number of operations, the variability in the output falls.
For e.g. a new coder may show more variance in coding productivity and an experienced one
2. As one input variable grows, the process outputs vary more for e.g. as monthly household income grows, there is more choice to spend on and hence the savings may fluctuate depending on the household preferences
3. Data collection techniques improve, the data collected first may show more variations than the data collected last
4. Outliers can lead to heteroscedasticity
5. Skewness in data can lead to heteroscedasticity. For e.g. few individuals with extremely high incomes will contribute most to the variations
6. The model specified may not be the correct one. We chose a simple linear model while the model should be relatively more complex

Heteroscedasticity of disturbance (Contd...)

Variance of u_i , homoscedastic or heteroscedastic plays no part in the determination of the coefficients.

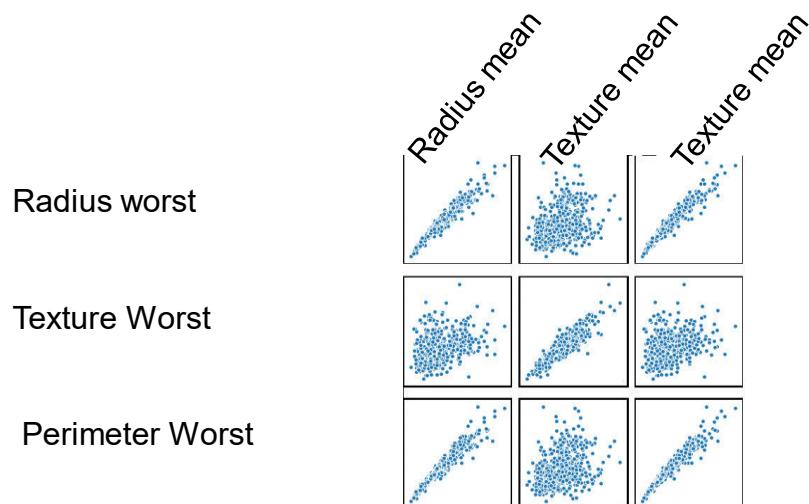
Even with heteroscedasticity the coefficients will converge to the population value of the coefficients. Infact the distribution of the coefficients remains asymptotically normally distributed.

The problems is, the coefficients will have lot of variance and make the overall model less accurate. The OLS method does not take into account the heteroscedastic nature of certain attributes relative to others. It gives same weightage to all attributes. On the other hand GLS (Generalized Least Square) method takes into account such difference and gives lesser weight to attributes with more heteroscedasticity and in the process result in better models

Multi-Collinearity

Multi Collinearity – What is it?

Multicollinearity is that situation where the independent variables in the linear model are not truly independent i.e. they are correlated. For e.g. in the Wisconsin Breast Cancer dataset the first three attributes “radius mean”, “perimeter mean” and “texture mean” is shown below. The first two are strongly correlated



Multi Collinearity – Types of multicollinearity

1. **Structural multicollinearity:** This type occurs when we create features from existing features and build a model using all of the features. For example, using “Radius” and “Area” as two variables. When features are generated, ensure the generated feature and the original features do not strongly correlate, if they do, you may want to drop the original feature as long as the generated feature contains all the information from the original

2. **Data multicollinearity:** This type of multicollinearity is an artifact of the data itself. The nature of the variables is such that they correlate. For e.g. in auto-mpg.csv, the columns “weight” and “horsepower” of a car will correlate positively. In case there are such correlating variables in the data, they may be combined into a composite variable using techniques such as PCA

Multi Collinearity

1. Assumptions 8 (There no perfect collinearity between the predictor variables X) – is technically known as the assumption of no collinearity or no multicollinearity when more than one variables is involved
2. Formally, no collinearity means there exists no two numbers λ_2 and λ_3 such that $\lambda_2x_2 + \lambda_3x_3 = 0$. If such a relationship exists, then X_2 and X_3 are said to be collinear or linearly dependent
3. On the other hand if the equation holds only when λ_2 and $\lambda_3 = 0$, then the variables are non-collinear
4. In simple terms, multicollinearity is the situation when two or more variables used in a model, are related to each other i.e. change in values of one leads to change in values of other
5. The problem with having multicollinearity is in the inability to understand how one variable influences the target. There is no way to estimate separate influence of each variable on target. Thus no way to estimate the partial regression coefficients

Multi Collinearity (Contd...)

6. If multicollinearity is perfect, the regression coefficients of X variables are indeterminate and their standard errors are infinite
7. If multicollinearity is less than perfect, the regression coefficients, although determinate, possess large standard errors, which means the coefficients cannot be estimated with confidence
8. High degree of multicollinearity will not take away the property of being best unbiased linear estimators. It violates none of the regression assumptions. The only problem is that it will result in hard to determine coefficients with small standard errors
9. But the same problem occurs when we have too few observations or the independent variables have small variances

Multi Collinearity – What is the problem?

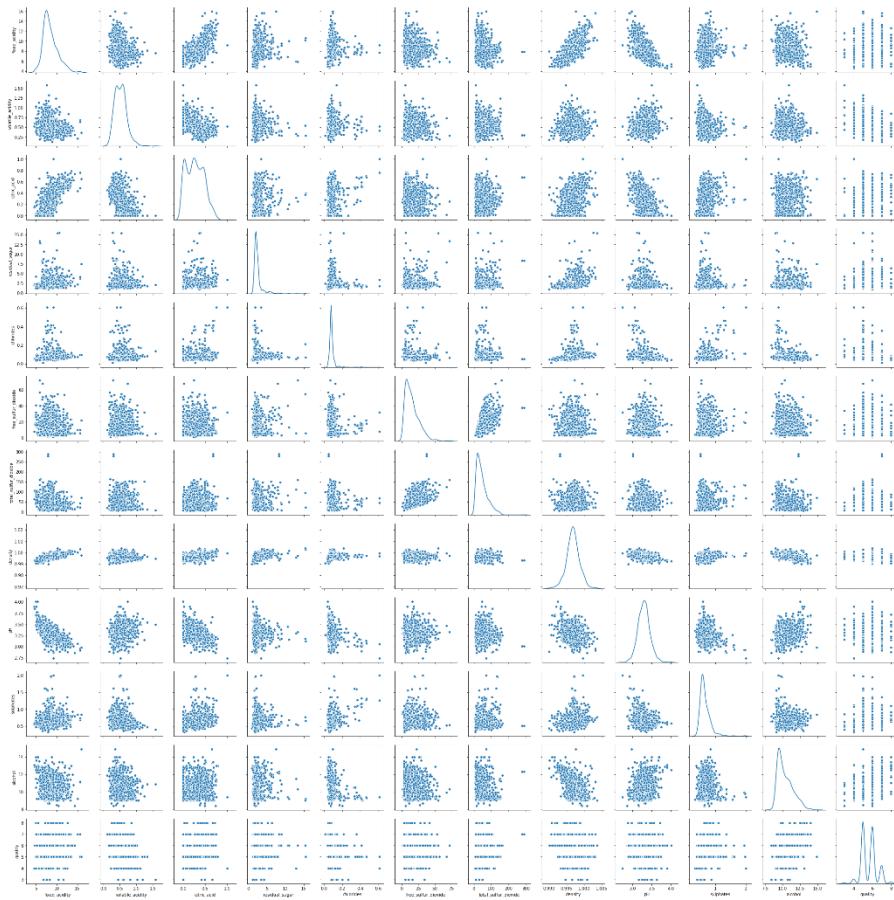
1. Independent variables should be *independent* of one another. Instead, if they correlate strongly, it can lead to sub-optimal model and mislead in terms of statistical results such as P values
2. The main objective of regression analysis is to express the relationship between each predictor variable and the dependent variable independently.
3. The regression coefficient is a measure of mean change in the dependent variable for each 1 unit change in an independent variable keeping all other independent variables constant. However, with collinear independent variables, it will not be possible to change one variable keeping others constant!
4. The coefficient estimates for an independent variable Vs the target variable can swing wildly based on inclusion or exclusion of other correlated independent variables are in the model. The coefficients become very sensitive to changes in the model structure
5. Multicollinearity reduces the precision of the estimate coefficients, which weakens the statistical power of your regression model. You might not be able to trust the p-values to identify independent variables that are statistically significant

Multi Collinearity – Testing for multicollinearity with Variation Inflation Factor (VIF)

1. The variance inflation factor (VIF) identifies correlation between independent variables and the strength of that correlation.
2. Statsmodel based linear models provide a VIF for each independent variable
3. VIFs start at 1 and have no upper limit.
 - a. A value of 1 indicates that there is no correlation between this independent variable and any others
 - b. VIFs between 1 and 5 suggest that there is a moderate correlation, but it is not severe enough to warrant corrective measures
 - c. VIFs greater than 5 represent critical levels of multicollinearity where the coefficients are poorly estimated, and the p-values are questionable.

Testing for multicollinearity with Variation Inflation Factor (VIF)

Red wines dataset, correlation between features and VIF values
Most attributes have very high value of VIF



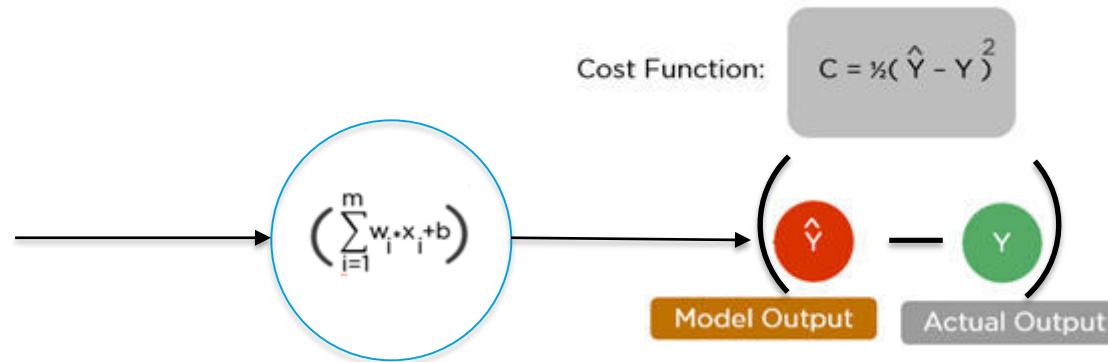
fixed_acidity ---> 80.0219390812402
volatile_acidity ---> 16.57719262878206
citric_acid ---> 9.774864397481704
residual_sugar ---> 4.906541592370461
chlorides ---> 6.529251770198401
free_sulfur_dioxide ---> 6.448644902127925
total_sulfur_dioxide ---> 6.87705611151861
density ---> 1445.240488945372
pH ---> 1037.4662099590764
sulphates ---> 20.65264657492166
alcohol ---> 121.46712238121306



Loss Function & Optimization Algorithm

Loss function (Mean Square Loss)

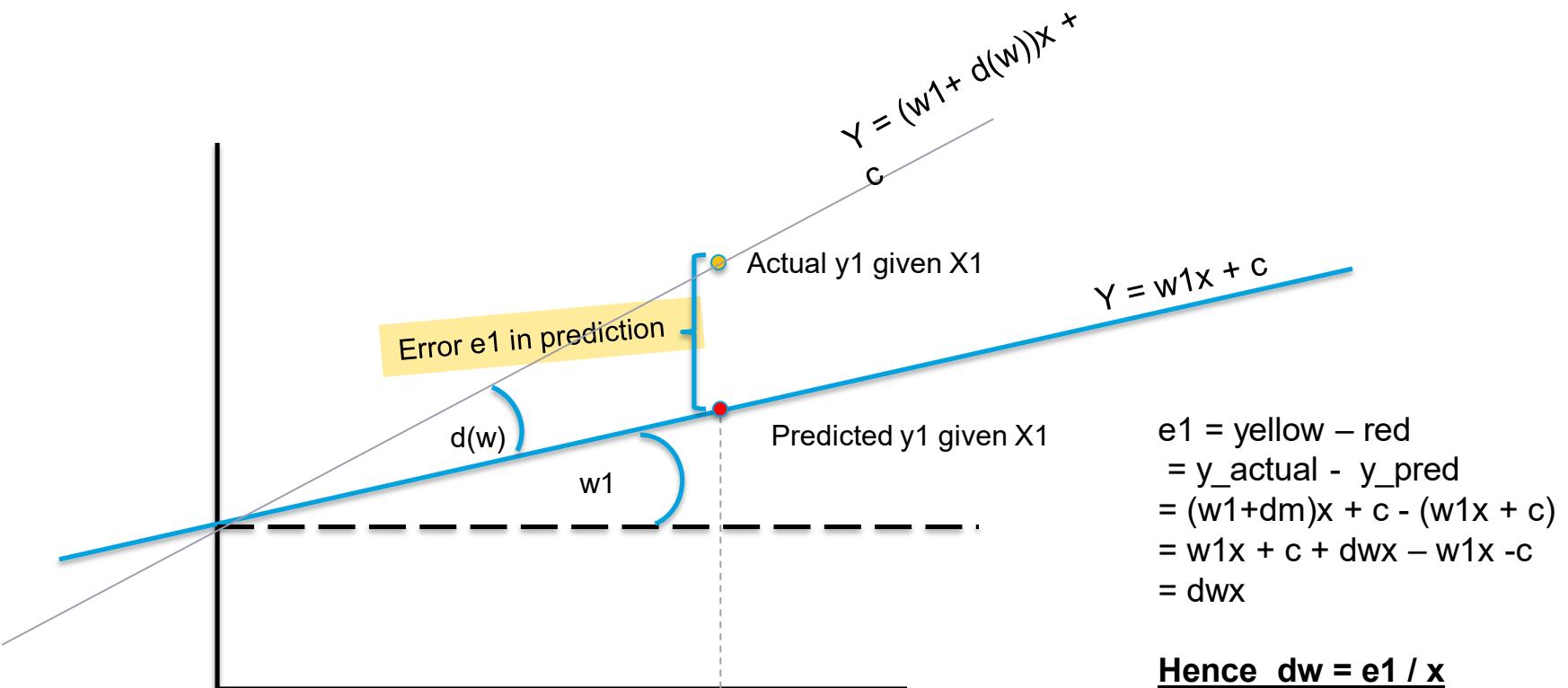
1. What is an optimization algorithm and what is its use? - Optimization algorithms help us to **minimize (or maximize)** an **Objective** function (*another name for Error function*) $E(x)$ which is simply a mathematical function dependent on the Model's internal **learnable parameters** which are used in computing the target values(Y) from the set of *predictors*(X) used in the model



2. $C = \frac{1}{2}((w_i \cdot x_i + b) - y)$. In this expression x_i and y come from the data and are given. What the ML algorithm learns is the weight w_i and bias b . Thus $C = f(w_i, b)$
3. The optimizer algorithms try to estimate the values of w_i and b which when used, will give minimum or maximum C . In ML we look for minimum

Relation between error and change in weights

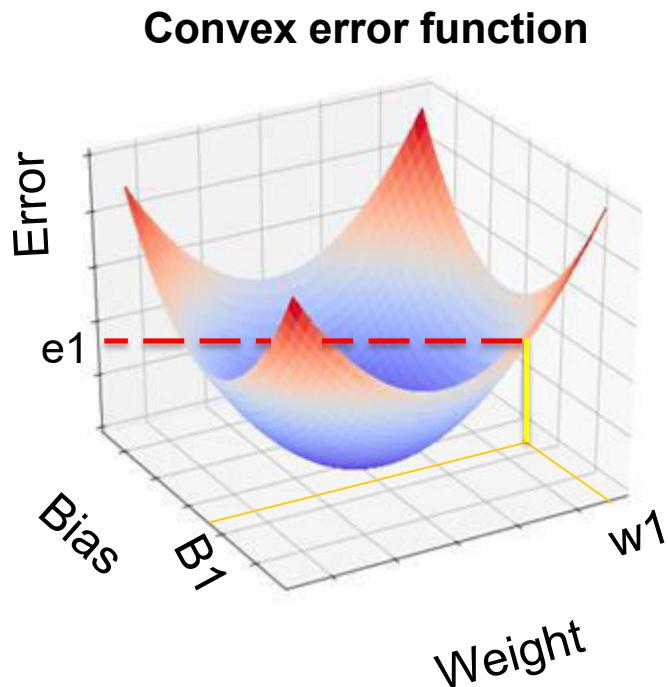
Since part of neuron function is linear equation (before applying the non-linear transformation), the error at each neuron can be expressed in terms of the linear equation.



The change required in m (dw) is e_1/x . However, change required w.r.t another data point may be different. To prevent jumping around with dw , we moderate the change in W by introducing a **learning rate α** . Hence $dw = \alpha(e_1/x)$

Gradient Descent

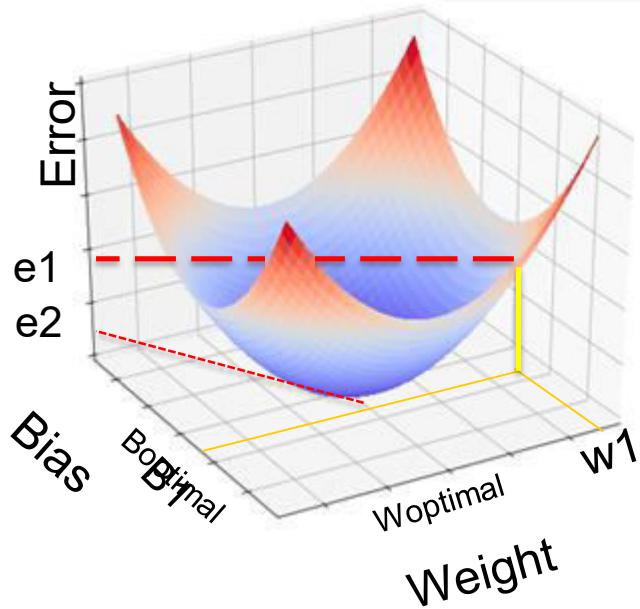
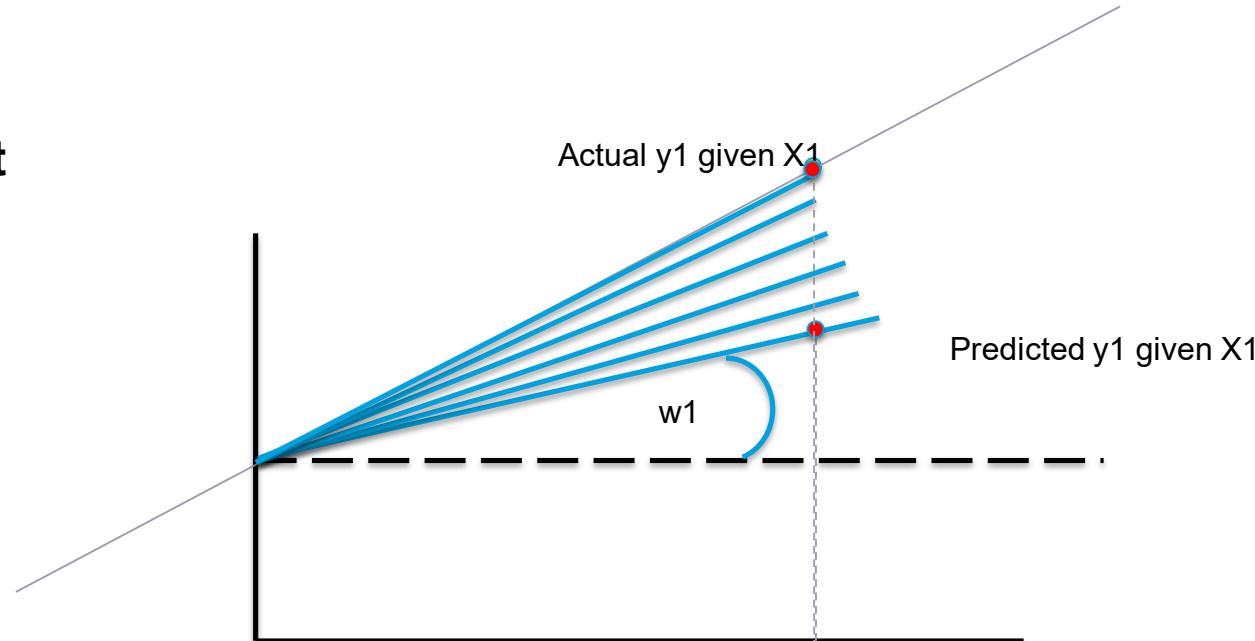
The challenge is, all the weights in all the inputs need to be adjusted. It is not manually possible to find the right combination of weights using brute force. Instead, the machine learning algorithm uses a learning function called gradient descent



1. A random combination of bias B_1 and input weights W_1 (showing only one as more than one is not possible to visualize)
2. Each combination of W_1 and B_1 is one particular linear model in a neuron. That model is associated with proportionate error e_1 (red dashed line).
3. Objective is to drive e_1 towards 0. For which we need to find the optimal weight ($W_{optimal}$) and bias ($B_{optimal}$)
4. The algorithm uses gradient descent algorithm to change bias and weight from starting values of B_1 and W_1 towards the $B_{optimal}$, $W_{optimal}$.

Note: in 3D error surface can be visualized as shown but not in more than 3 dimensions

Gradient Descent



Least error E_2 is at the global minima of the convex function which only one unique combination of weight (w_{optimal}) and bias (b_{optimal}) will fetch us.

Gradient Descent

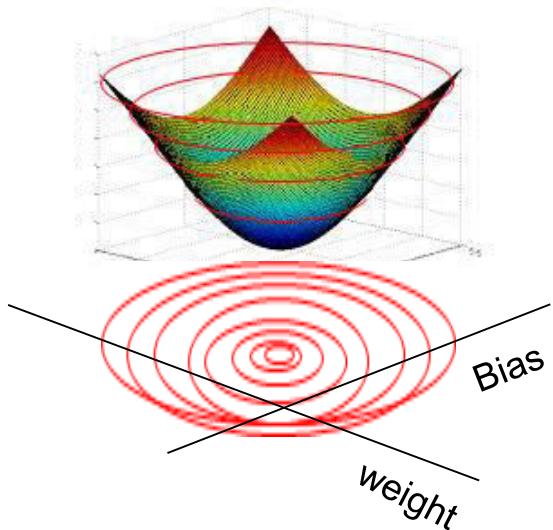
1. Let target value for a training example X be y i.e. The data frame used for training has value X, y
2. Let the model (represented by random m and c) predict the value for the training example X to be \hat{y}
3. Error in prediction is $E = \hat{y} - y$. If we sum all the errors across all data points, some will be positive some negative and thus cancel out
4. To prevent the sum of errors becoming 0, we square the error i.e. $E = (\hat{y} - y)^2$. Note: in squared expression, $\hat{y} - y$ or $y - \hat{y}$ mean the same
5. Sum of $(\hat{y} - y)^2$ across all the X values is called SSE (Sum of Squared Errors)
6. Using gradient descent (descend towards the global minima). Gradient descent uses partial derivatives i.e how the SSE changes on slightly modifying the model parameters m and c one at a time

$$\frac{d(E)}{d(m)} = \frac{d(\text{sum}(\hat{y} - y)^2)}{d(m)}$$

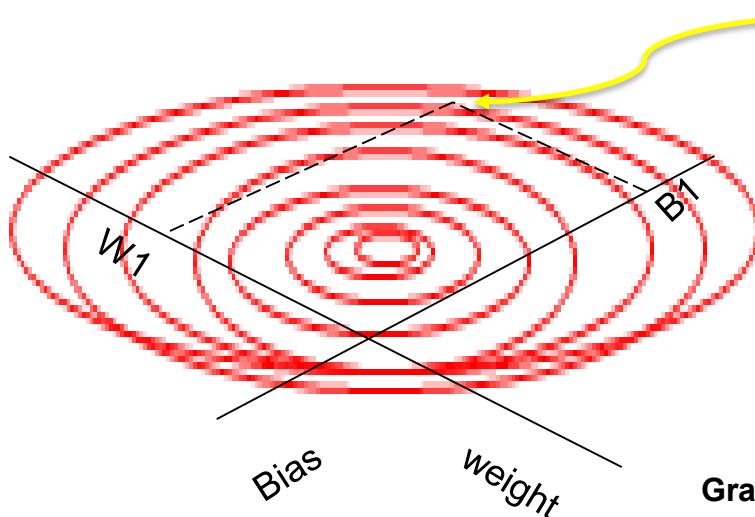
$$\frac{d(E)}{d(c)} = \frac{d(\text{sum}(\hat{y} - y)^2)}{d(c)}$$

Gradient Descent

Transform our error function (which is a quadratic / convex function) into a contour graph. Gradient is always found on the input model parameters only



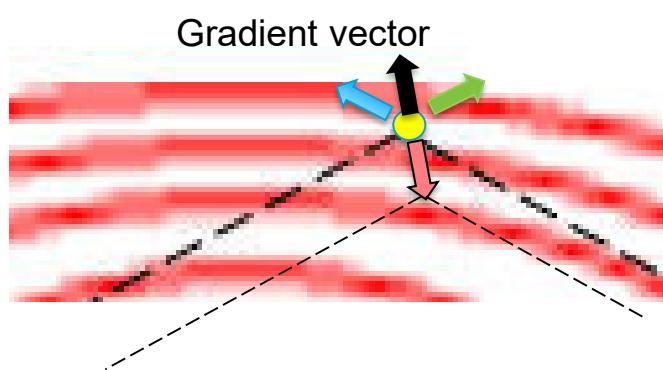
1. Every ring on the error function represents a combination of coefficients (m_1 and m_2 in the image) which result in same quantum of error i.e. SSE
2. Let us convert that to a 2d contour plot. In the contour plot, every ring represents one quantum of error.
3. The innermost ring / bull's eye is the combination of the coefficients that gives the lease SSE



Randomly selected starting point

About the contour graph –

1. Outermost circle is highest error while innermost is the least error circle
2. A circle represents combination of parameters which result in same error. Moving on a circle will not reduce error.
3. Objective is to start from anywhere but reach the innermost circle



Gradient Descent Steps –

1. First evaluate $dy(\text{error})/d(\text{weight})$ to find the direction of highest increase in error given a unit change in weight (Blue arrow). Partial derivative w.r.t. to weight
2. Next find $dy(\text{error}) / d(\text{bias})$ to find the direction of highest increase in error given a unit change in bias (green arrow). Partial derivative w.r.t. to bias
3. Partial derivatives give the gradient in the given axis and gradient is a vector
4. Add the two vectors to get the direction of gradient (black arrow) i.e. direction of max increase in error
5. We want to decrease error, so find negative of the gradient i.e. opposite to black arrow (Orange arrow). The arrow tip is new value of bias and weight.
6. Recalculate the error at this combination and iterate to step 1 till movement in any direction only increases the error

Gradient Descent

1. Gradient descent is a way to minimize an objective function / cost function such as Sum of Squared Errors (SSE) that is dependent on model parameters of weight / slope and bias
2. The parameters are updated in the direction opposite to the direction of the gradient (direction of maximum increase) of the objective function
3. In other words we change the values of weight and bias following the direction of the slope of the surface of the error function down the hill until we reach minima
4. This movement from starting weight and bias to optimal weight and bias may not happen in one shot. It is likely to happen in multiple iterations. The values change in steps
5. The step size can be influenced using a parameter called Learning Rate. It decides the size of the steps i.e. the amount by which the parameters are updated. Too small learning step will slow down the entire process while too large may lead to an infinite loop

6. The mathematical expression of gradient descent

Update Model parameter at e2

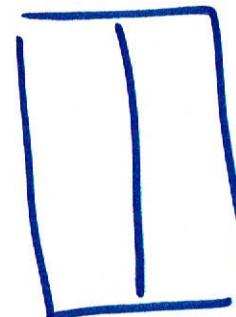
Old Model parameter at e1

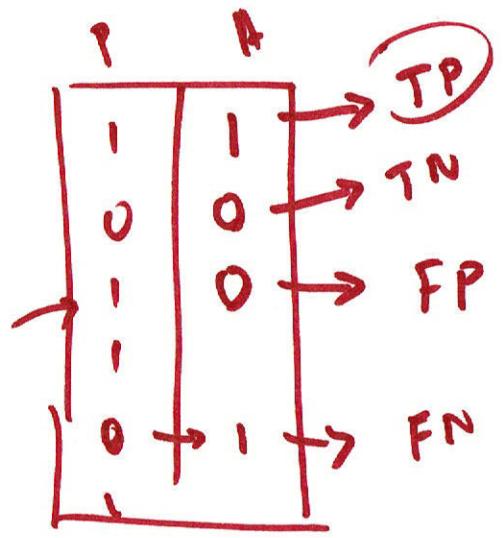
learning step
Gradient descent with learning step

$$\theta = \theta - \eta \cdot \nabla_{\theta} J(\theta)$$

Performance Measures

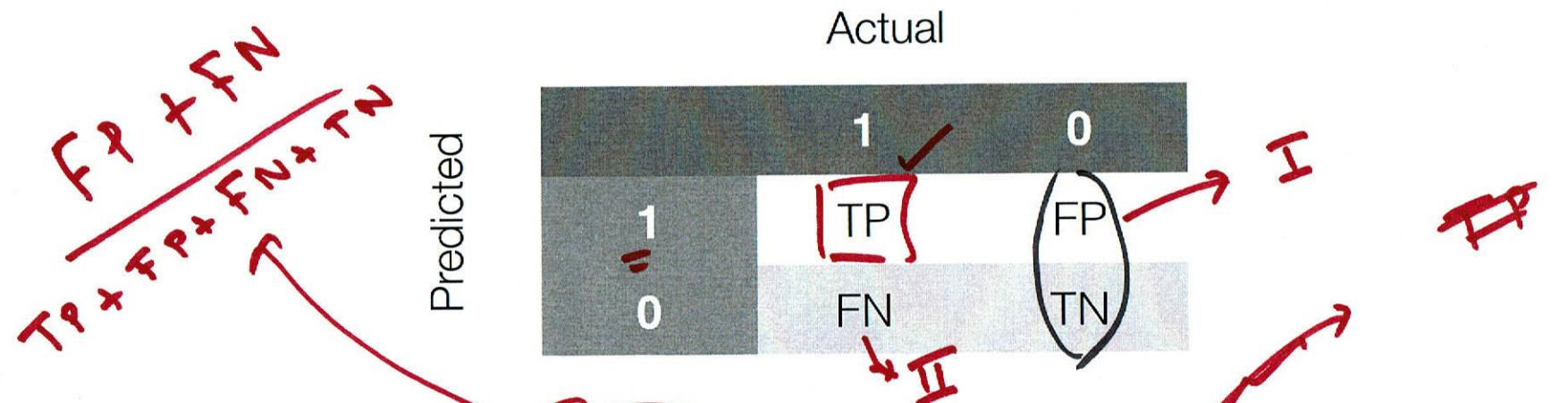
- Confusion Matrix ↵
- ROC Curves, Gini Coefficient ↵
- Gain and Lift Chart ↵
- Kolmogorov-Smirnov (K-S) chart ↵
- Concordance-Discordance ratio ↵
- Root Mean Square Error, Mean Absolute Error ↵





Confusion Matrix

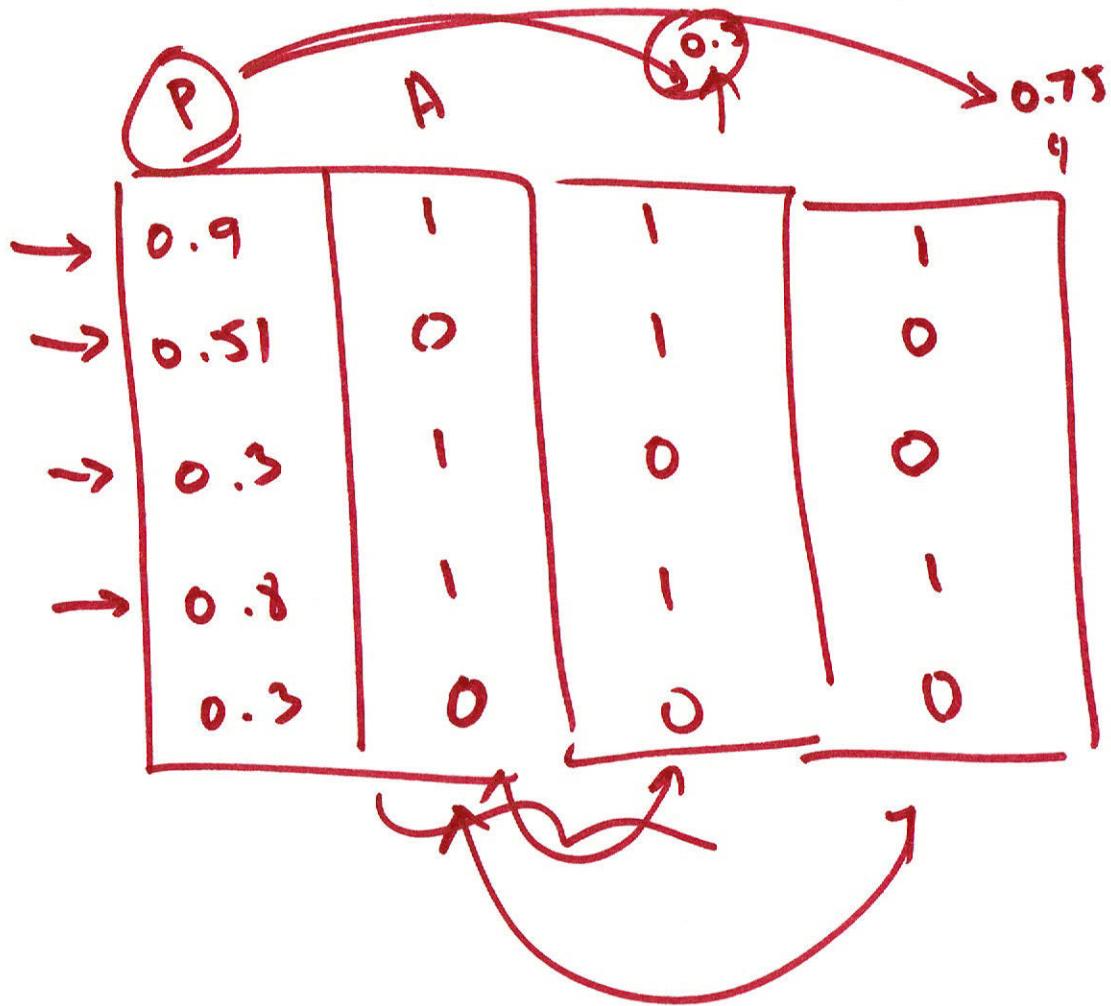
- For classification problem with a class output, the confusion matrix gives the counts of correct and erroneous predictions:

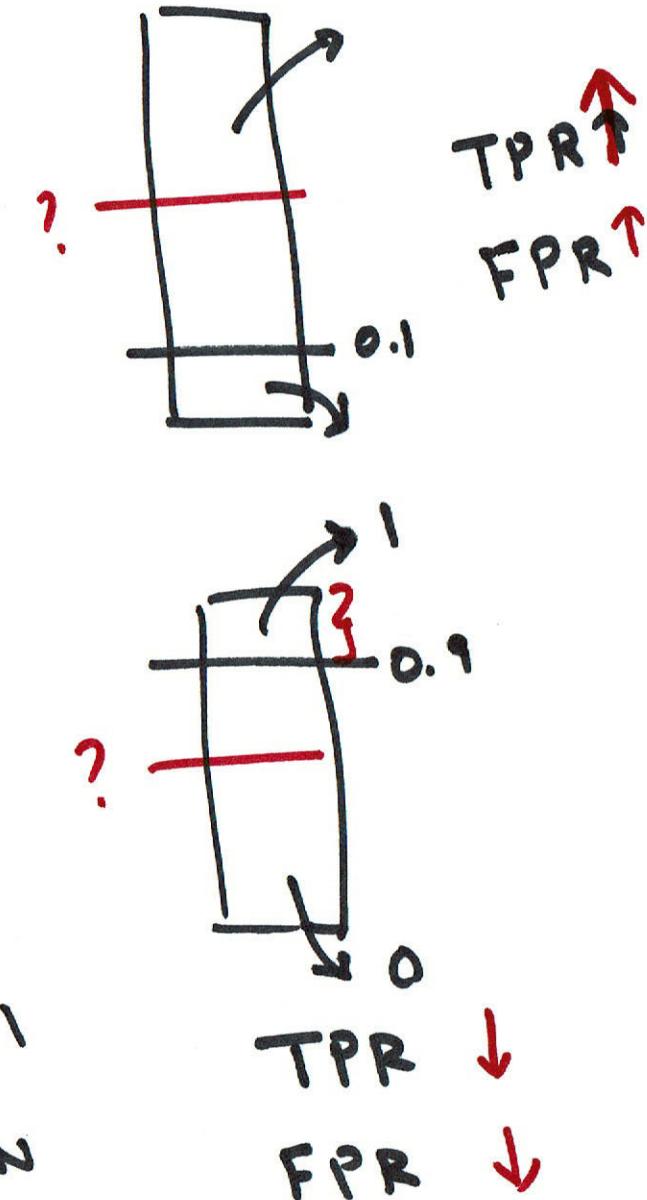
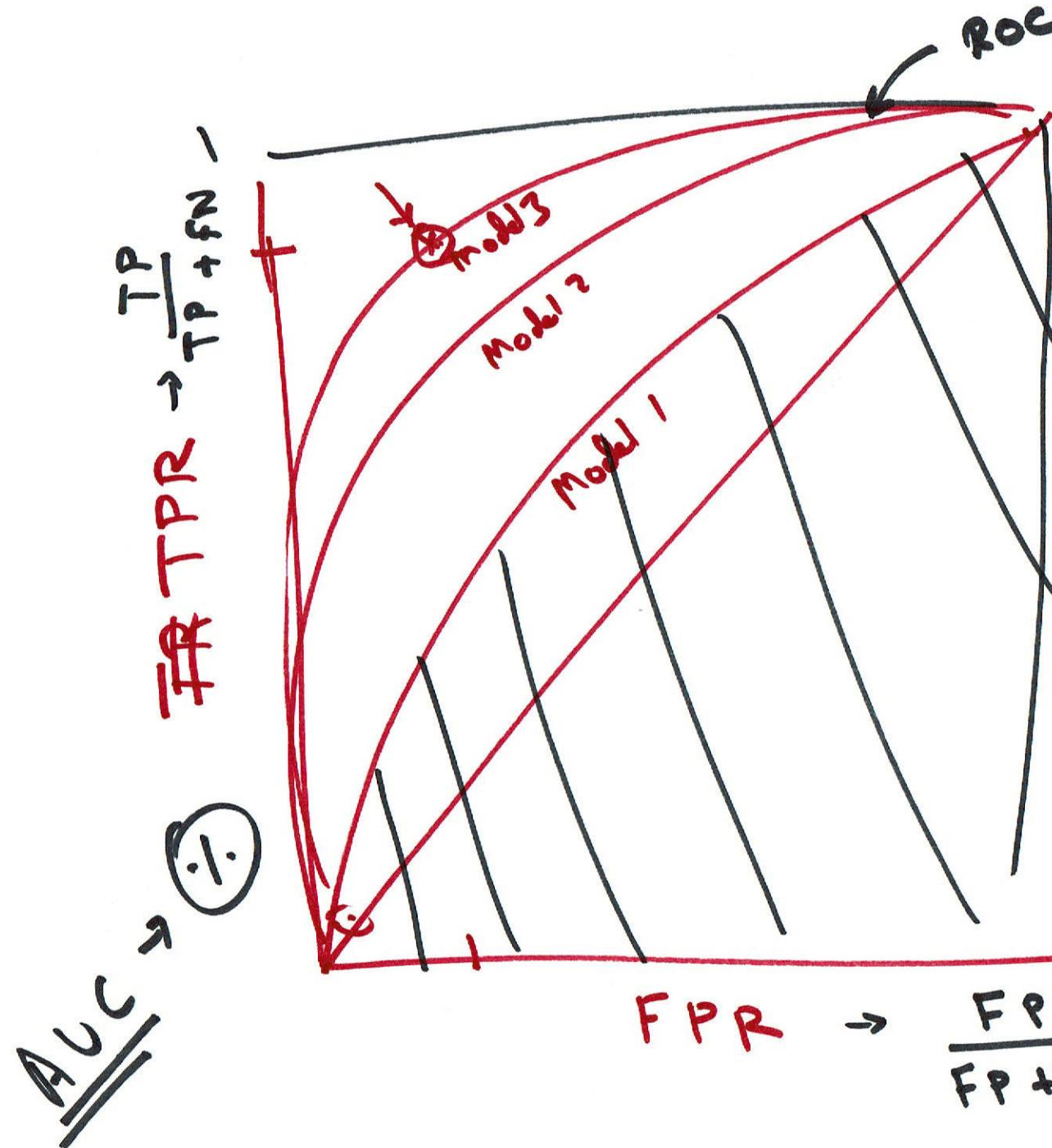


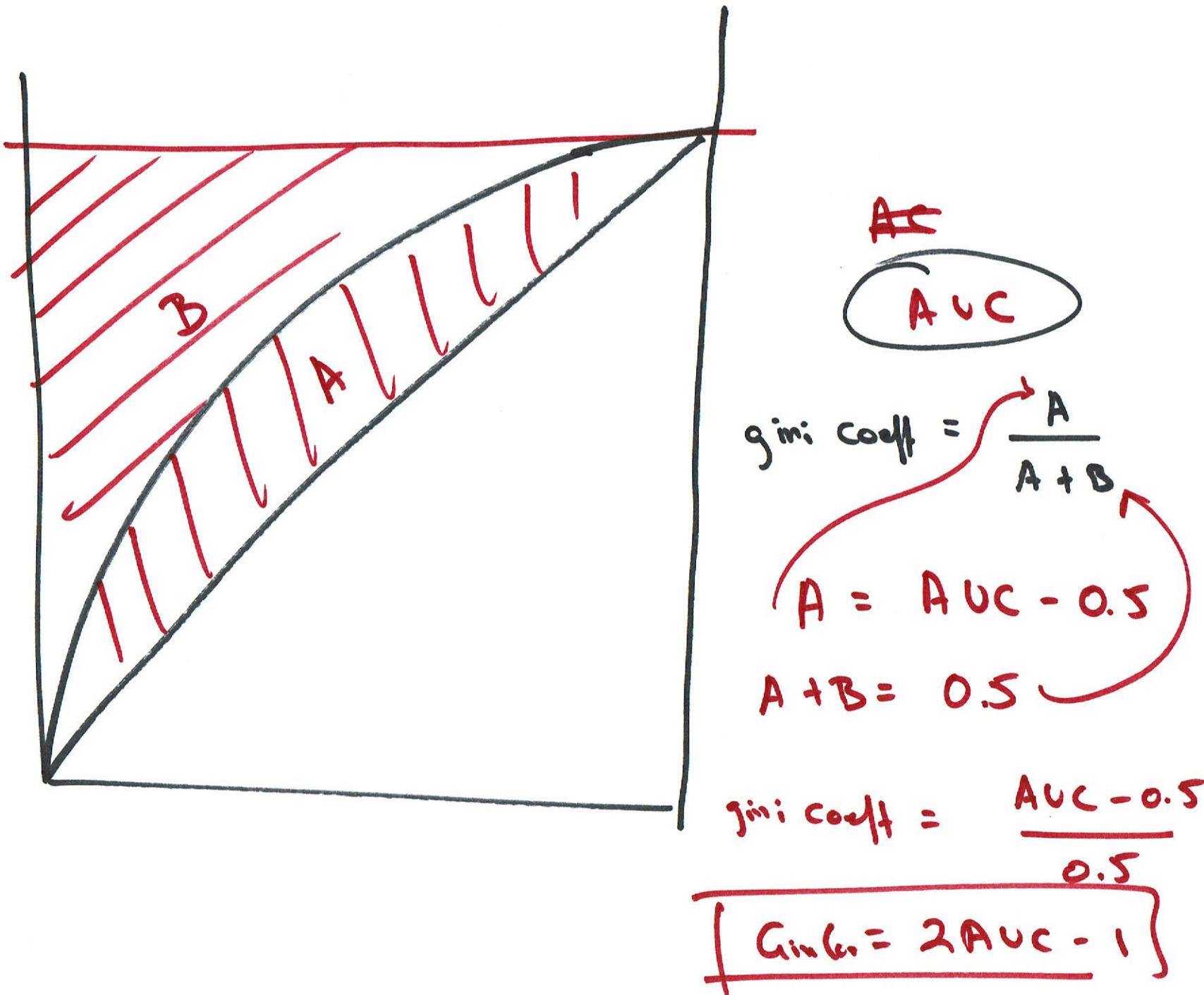
- Classification Error Rate: sum of Type 1 (FP) and Type 2 (FN) Errors (in percentage). Accuracy is $1 - (\text{error rate})$
- Sensitivity (also called Recall or True Positive Rate): proportion of Total Positives that were correctly identified
- Specificity (also called True Negative Rate): proportion of Total Negatives that were correctly identified

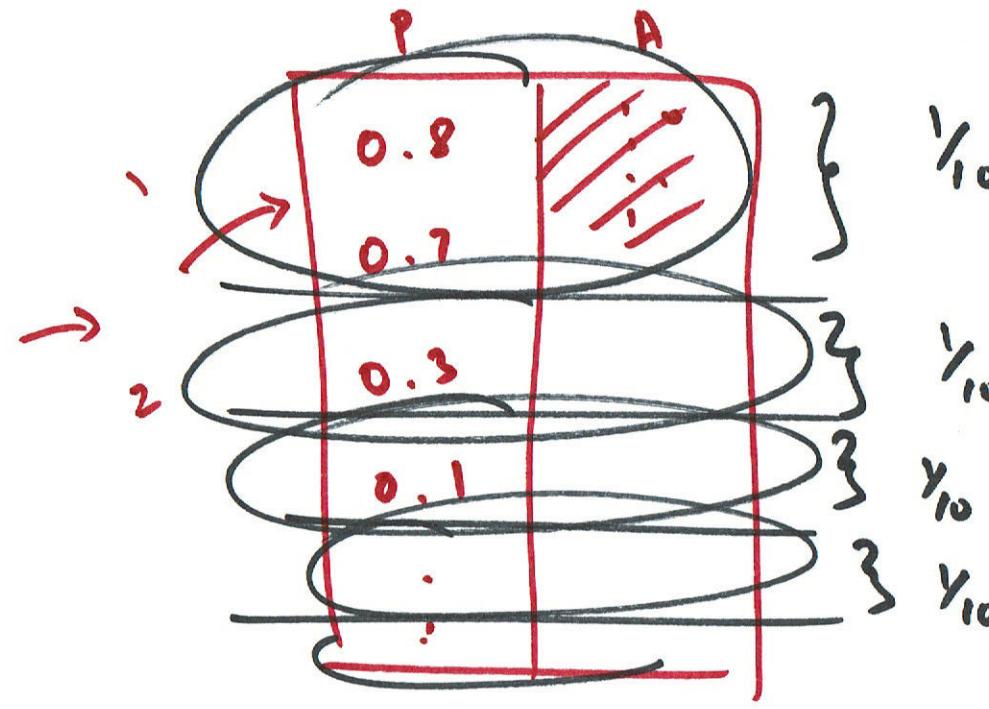
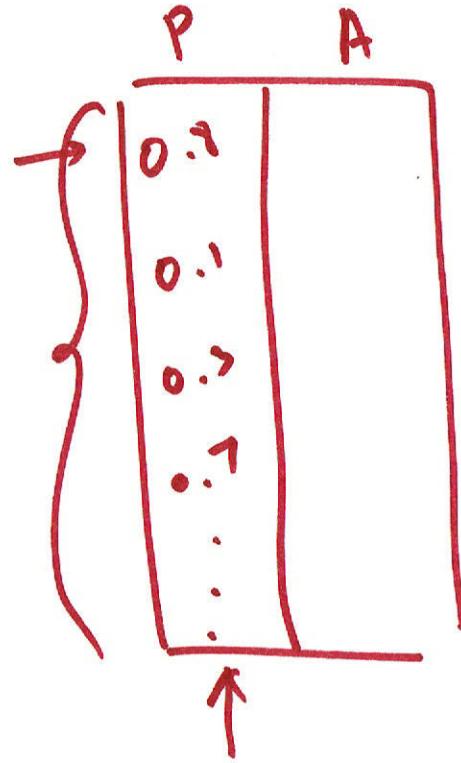
$$\frac{TP}{TP + FN}$$

$$\frac{TN}{TN + FP}$$





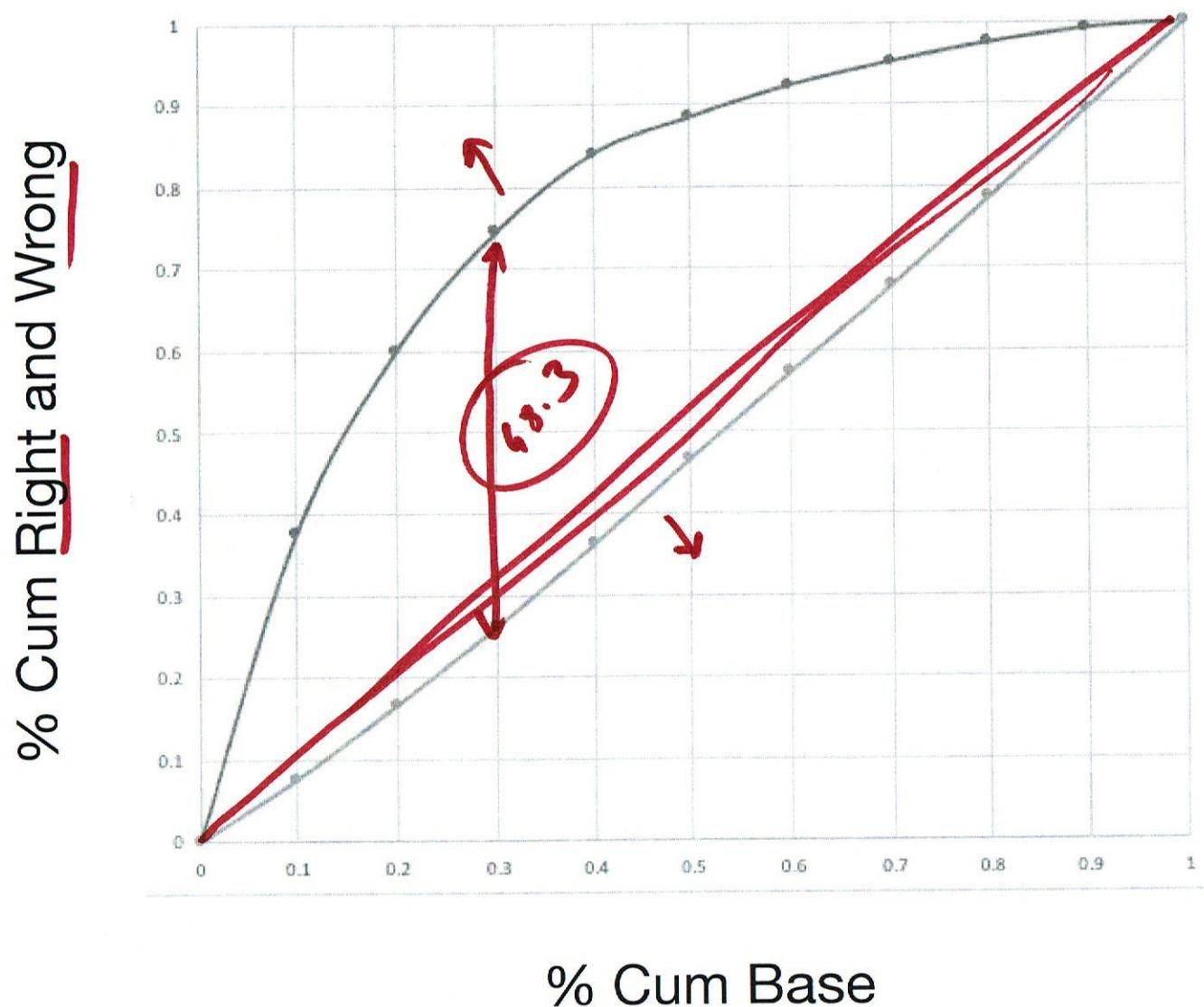




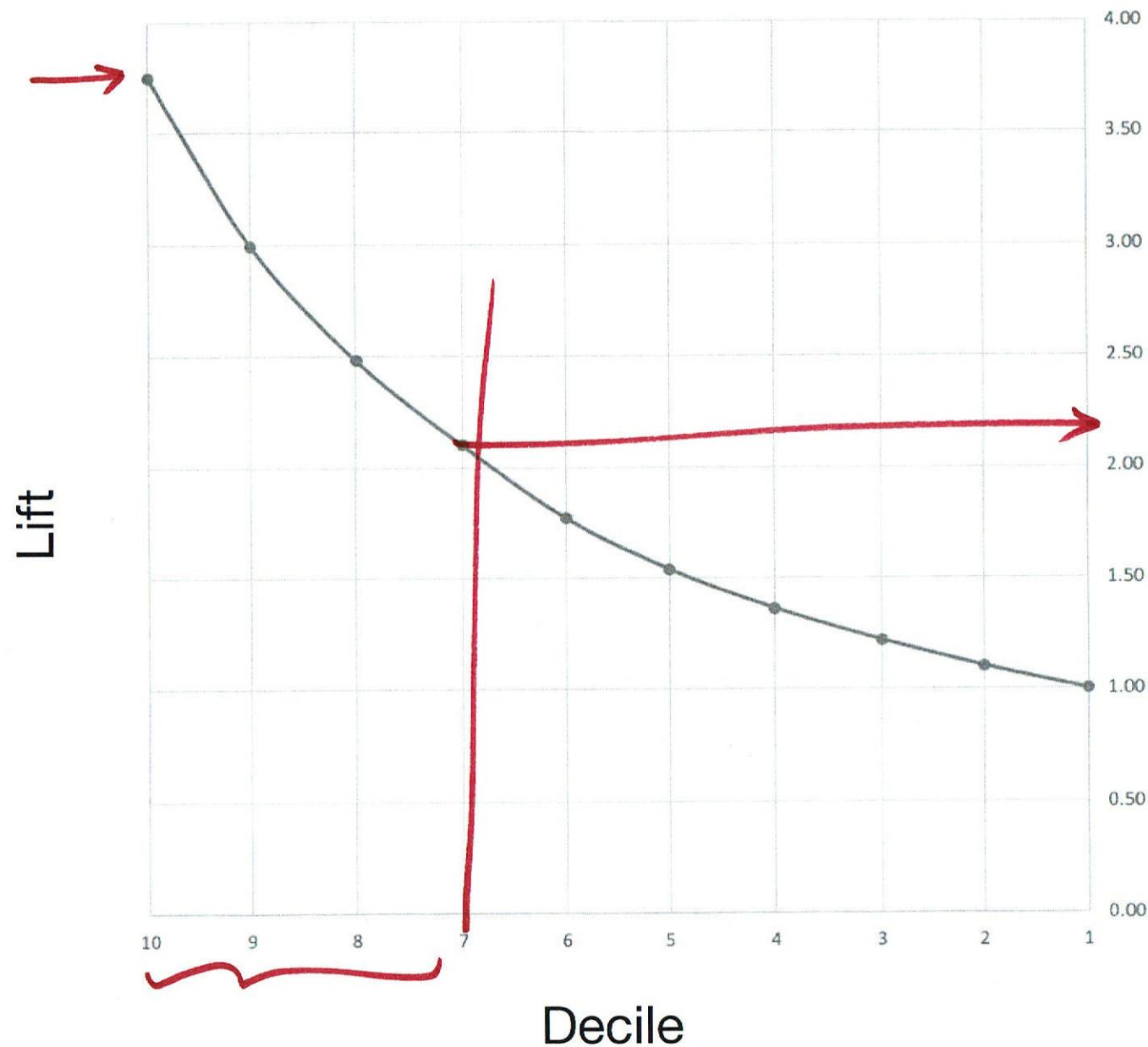
Rank Ordered Table Example

Decile	Base Cnt	#Right	# Wrong	%Right	Cum. Resp.	Cum. Non-	%Cum . Base	%Cum . Resp.	%Cum . Non-	KS	Lift
A	B	C	D=B-C	E=C/B	F = CumSum(C)	G=CumSum(D)	H = CumSum(B)/Total	I = F/Total	J=G/Total	I-J	I/H
10	1000	295	705	29.50%	295	705	10%	37.48%	7.65%	29.83%	3.75
9	1000	176	824	17.60%	471	1529	20%	59.85%	16.60%	43.25%	2.99
8	1000	115	885	11.50%	586	2414	30%	74.46%	26.20%	48.26%	2.48
7	1000	75	925	7.50%	661	3339	40%	83.99%	36.24%	47.75%	2.10
6	1000	35	965	3.50%	696	4304	50%	88.44%	46.72%	41.72%	1.77
5	1000	30	970	3.00%	726	5274	60%	92.25%	57.25%	35.00%	1.54
4	1000	23	977	2.30%	749	6251	70%	95.17%	67.85%	27.32%	1.36
3	1000	18	982	1.80%	767	7233	80%	97.46%	78.51%	18.95%	1.22
2	1000	13	987	1.30%	780	8220	90%	99.11%	89.22%	9.89%	1.10
1	1000	7	993	0.70%	787	9213	100%	100.00%	100.00%	0.00%	1.00
Total	10000	787	9213	7.87%	787	9213					

K-S Chart



Lift Chart



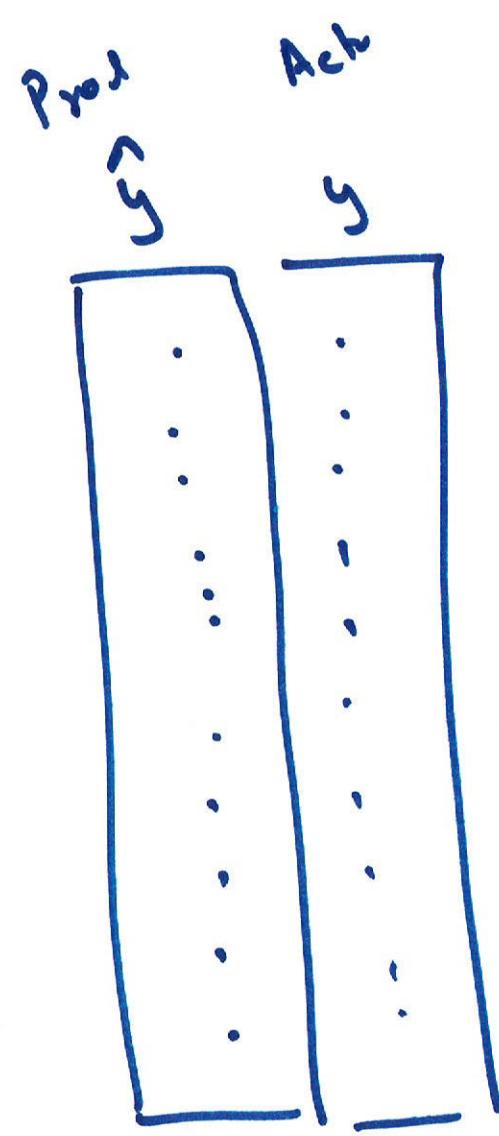
Example

Name	Right?	Prob
A	0	0.056
B	0	0.134
C	0	0.156
D	1	0.512
E	0	0.235
F	0	0.25
G	1	0.25
H	1	0.2
I	0	0.135
J	0	0.089

A P

- D, A $\rightarrow 0.512, 0.056 \rightarrow \checkmark C$
- D, B $\rightarrow 0.512, 0.0134 \rightarrow \checkmark C$
- D, C
- D, E
- D, F
- D, I
- D, J
- G, A
- G, F $\rightarrow 0.25, 0.25 \rightarrow O T$
- H, E $\rightarrow 0.2, 0.235 \rightarrow X D$

$$\text{Concordance Ratio} = \frac{18}{21}$$



$$\frac{1}{n} \sum |y_i - \hat{y}_i|$$

MAE

RMSE

$$\sqrt{\frac{1}{n} \sum (y_i - \hat{y}_i)^2}$$

Machine Learning

Naive Bayes Classifier

Naive Bayes Classification

- Will my flight be on time? It is Sunny, Hot, Normal Humidity, and not Windy!
- Data from the last several times we took this flight

OUTLOOK	TEMPERATURE	HUMIDITY	WINDY	Flight On Time
Rainy	Hot	High	FALSE	No
Rainy	Hot	High	TRUE	Yes
Overcast	Hot	High	FALSE	Yes
Sunny	Mild	High	FALSE	No
Sunny	Cool	Normal	FALSE	Yes
Sunny	Cool	Normal	TRUE	No
Overcast	Cool	Normal	TRUE	Yes
Rainy	Mild	High	FALSE	No
Rainy	Cool	Normal	FALSE	Yes
Sunny	Mild	Normal	FALSE	Yes
Rainy	Mild	Normal	TRUE	Yes
Overcast	Mild	High	TRUE	Yes
Overcast	Hot	Normal	FALSE	Yes
Sunny	Mild	High	TRUE	No

Probability Review

- If A is any event, then the complement of A, denoted by \bar{A} , is the event that A does not occur.
- The probability of A is represented by $P(A)$, and the probability of its complement $P(\bar{A}) = 1 - P(A)$.
- Let A and B be any events with probabilities $P(A)$ and $P(B)$.
 - If you are told that B has occurred, then the probability of A might change. The new probability of A is called the conditional probability of A given B.
 - Conditional probability: $P(A|B) = P(A \text{ and } B) / P(B)$
 - Multiplication rule: $P(A \text{ and } B) = P(A|B) P(B)$

Probabilistic Independence

- Probabilistic independence means that knowledge of one event is of no value when assessing the probability of the other.
- The main advantage to knowing that two events are independent is that in that case the multiplication rule simplifies to: $P(A \text{ and } B) = P(A) P(B)$.

Bayes' Rule

- $P(A|B)$, reads “A given B,” represents the probability of A if B was known to have occurred.
- In many situations we would like to understand the relation between $P(A|B)$ and $P(B|A)$.
- You are planning an outdoor event tomorrow. When it actually rains, the weatherman correctly forecasts rain 90% of the time. When it doesn't rain, he incorrectly forecasts rain 10% of the time. Historically it has rained only 5 days each year. Unfortunately, the weatherman has predicted rain for tomorrow. What is the probability that it will rain tomorrow?

use Bayes' Rule:

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

Bayes' Rule Continued

- Let A_1 through A_n be a set of mutually exclusive outcomes.
- The probabilities of the A s are $P(A_1)$ through $P(A_n)$. These are called prior probabilities.
- Because an information outcome might influence our thinking about the probabilities of any A_i , we need to find the conditional probability $P(A_i|B)$ for each outcome A_i . This is called the posterior probability of A_i .
- Using Bayes' Rule:

$$P(A_i|B) = \frac{P(B|A_i)P(A_i)}{P(B|A_1)P(A_1) + \dots + P(B|A_n)P(A_n)}$$

Bayes' Rule Continued

- In words, Bayes' rule says that the posterior is the likelihood times the prior, divided by a sum of likelihoods times priors.
- The denominator in Bayes' rule is the probability $P(B)$.

$$\text{posterior probability} = \frac{\text{conditional probability} \cdot \text{prior probability}}{\text{evidence}}$$

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

So will our flight be on time?

Outlook

	Yes	No	P(yes)	P(no)
Sunny	2	3	2/9	3/5
Overcast	4	0	4/9	0/5
Rainy	3	2	3/9	2/5
Total	9	5	100%	100%

Temperature

	Yes	No	P(yes)	P(no)
Hot	2	2	2/9	2/5
Mild	4	2	4/9	2/5
Cool	3	1	3/9	1/5
Total	9	5	100%	100%

Humidity

	Yes	No	P(yes)	P(no)
High	3	4	3/9	4/5
Normal	6	1	6/9	1/5
Total	9	5	100%	100%

Wind

	Yes	No	P(yes)	P(no)
False	6	2	6/9	2/5
True	3	3	3/9	3/5
Total	9	5	100%	100%

On Time?	P(Yes)/P(No)
Yes	9
No	5
Total	14
	100%

Naïve Bayes Classifiers

- Probabilistic models based on Bayes' theorem.
- It is called “naive” due to the assumption that the features in the dataset are mutually independent
- In real world, the independence assumption is often violated, but naïve Bayes classifiers still tend to perform very well
- Idea is to factor all available evidence in form of predictors into the naïve Bayes rule to obtain more accurate probability for class prediction
- It estimates conditional probability which is the probability that something will happen, given that something else has already occurred. For e.g. the given mail is likely a spam given appearance of words such as “prize”
- Being relatively robust, easy to implement, fast, and accurate, naive Bayes classifiers are used in many different fields

Naïve Bayes Classifiers - Pros and Cons

- Advantages
 - Simple, Fast in processing and effective
 - Does well with noisy data and missing data
 - Requires few examples for training (assuming the data set is a true representative of the population)
 - Easy to obtain estimated probability for a prediction
- Dis-advantages
 - Relies on and often incorrect assumption of independent features
 - Not ideal for data sets with large number of numerical attributes
 - Estimated probabilities are less reliable in practice than predicted classes
 - If rare events are not captured in the training set but appears in the test set the probability calculation will be incorrect

Gaussian Naive Bayes classifier

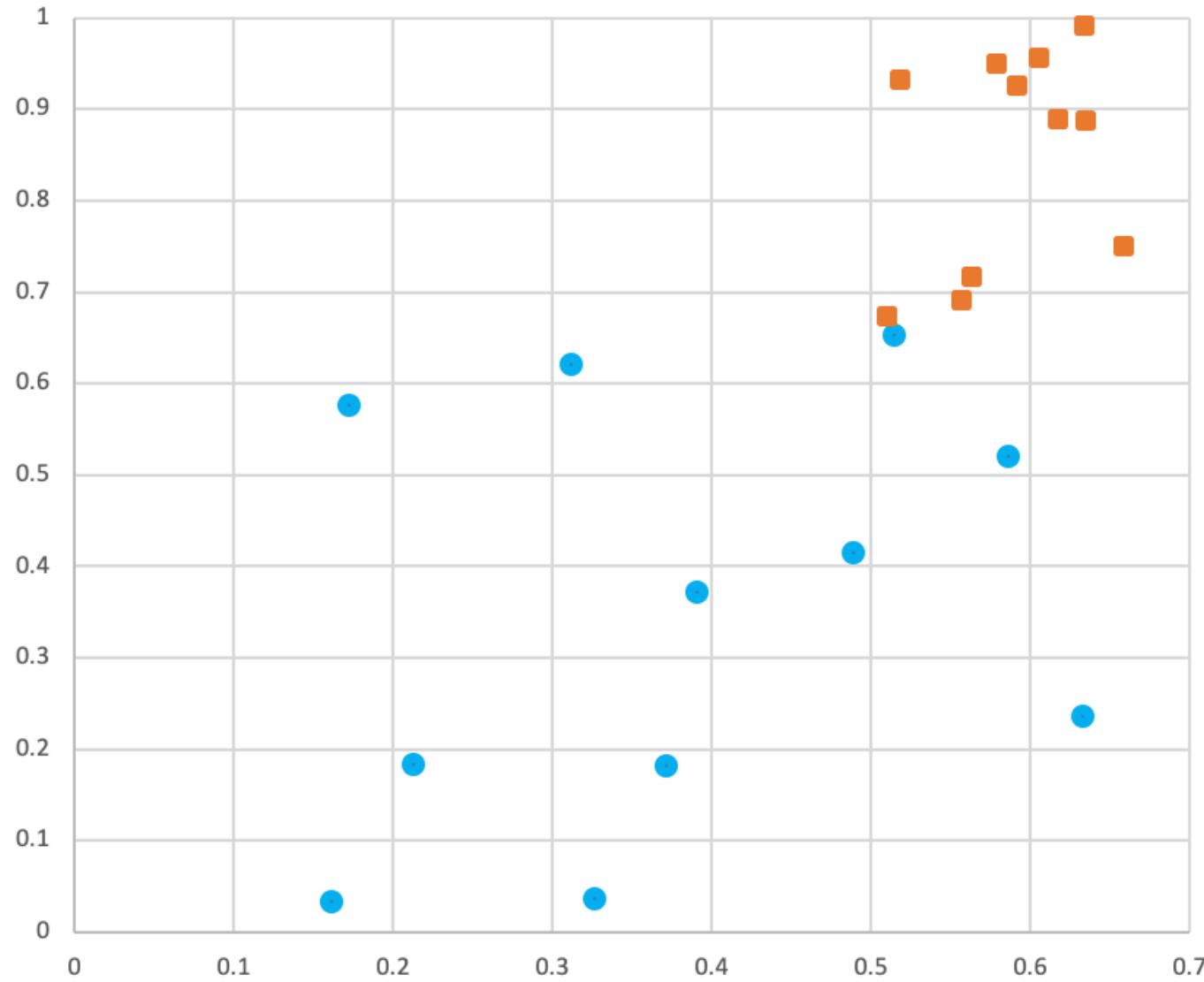
- When some of our independent variables are continuous we cannot calculate conditional probabilities!
- In Gaussian Naive Bayes, continuous values associated with each feature (or independent variable) are assumed to be distributed according to a Gaussian distribution
- All we would have to do is estimate the mean and standard deviation of the continuous variable.

Machine Learning

K-Nearest Neighbors

K-Nearest Neighbors

- Simple! A data point is most similar to its neighbors



Distance measure is important

- Most commonly distance is measured using Euclidian distances
- We should always Normalize data
- Other distance measurement methods include
 - Manhattan distance
 - Minkowski distance
 - Mahalanobis distance
 - Cosine similarity

- The approach to find nearest neighbors using distance between the query point and all other points is called the brute force. Becomes time costly and inefficient with increase in number of points
- Determining the optimal K is the challenge in K Nearest Neighbor classifiers.
 - Larger value of K suppresses impact of noise but prone to majority class dominating
-

Other Variants

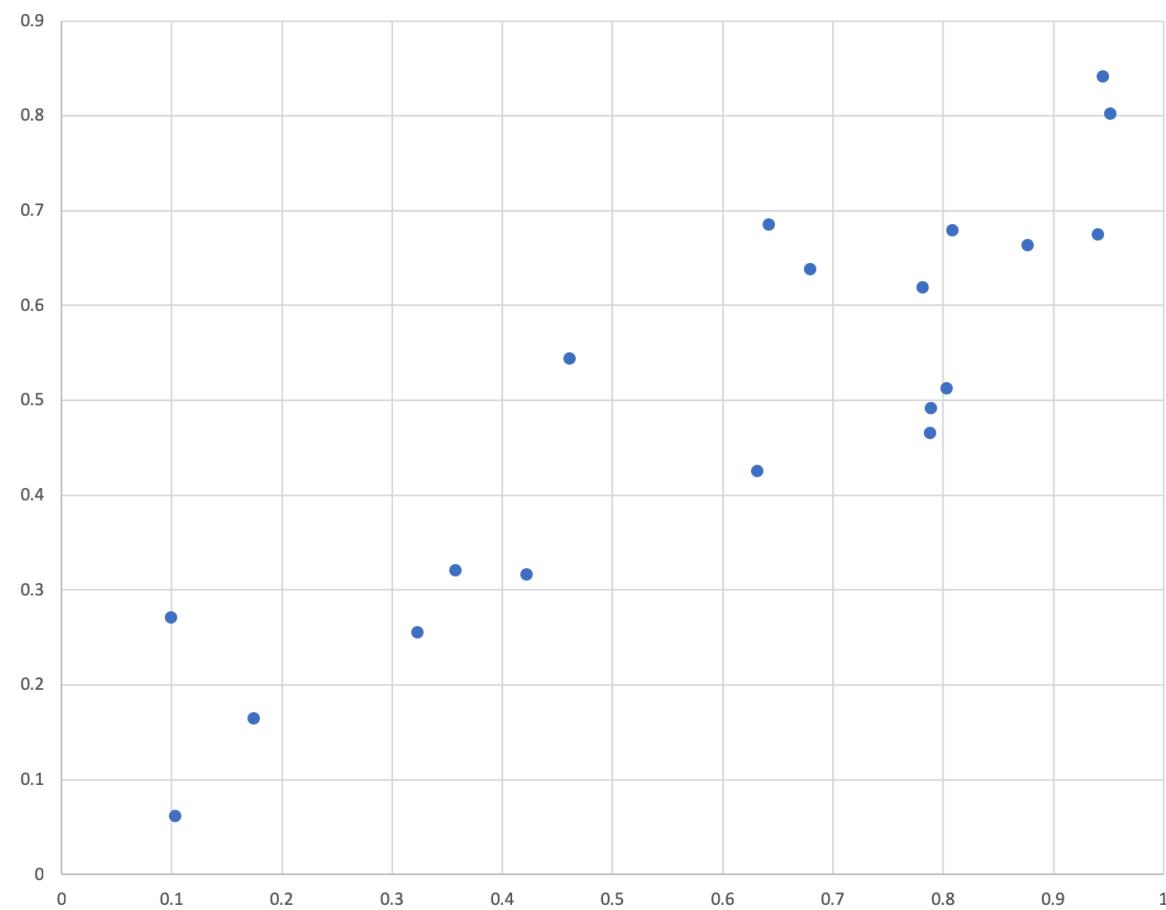
- Radius Neighbor Classifier
 - implements learning based on number of neighbors within a fixed radius r of each training point, where r is a floating point value specified by the user
 - may be a better choice when the sampling is not uniform. However, when there are many attributes and data is sparse, this method becomes ineffective due to curse of dimensionality
- KD Tree nearest neighbor
 - Approach helps reduce the computation time.
 - Very effective when we have large data points but still not too many dimensions

K-NN

- It does not construct a “model”. Known as a non-parametric method.
- Classification is computed from a simple majority vote of the nearest neighbors of each point
- Suited for classification where relationship between features and target classes is numerous, complex and difficult to understand and yet items in a class tend to be fairly homogenous on the values of attributes
- Not suitable if the data is too noisy and the target classes do not have clear demarcation in terms of attribute values
- Can also be used for regression

K-NN for regression

- The Neighbors based algorithm can also be used for regression where the labels are continuous data and the label of query point can be average of the labels of the neighbors



K Nearest Neighbors - pros and cons

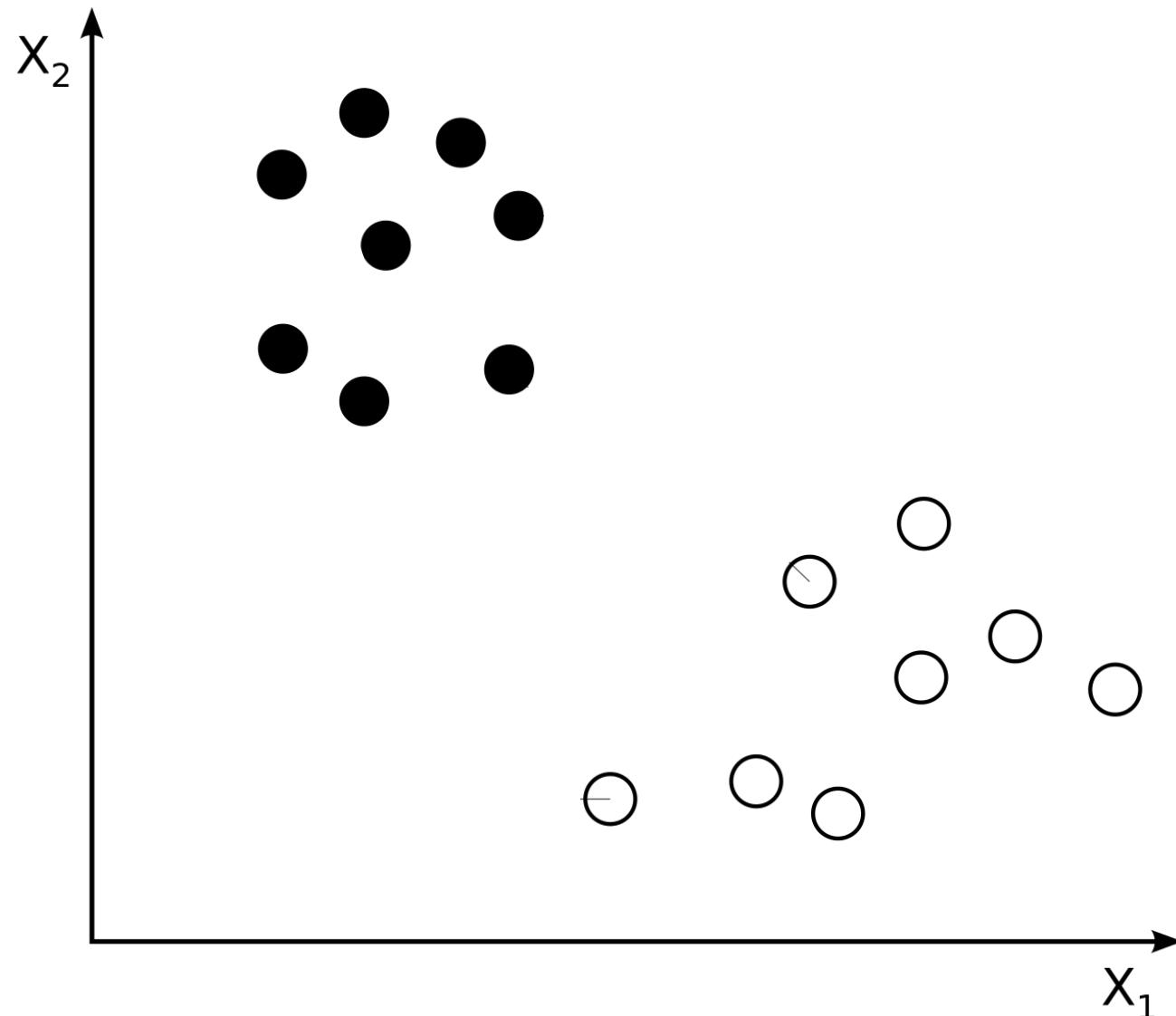
- Advantages
 - Makes no assumptions about distributions of classes in feature space
 - Can work for multi classes simultaneously
 - Easy to implement and understand
 - Not impacted by outliers
- Dis-advantages
 - Fixing the optimal value of K is a challenge
 - Will not be effective when the class distributions overlap
 - Does not output any models. Calculates distances for every new point (lazy learner)
 - Computationally intensive

Machine Learning

Support Vector Machines

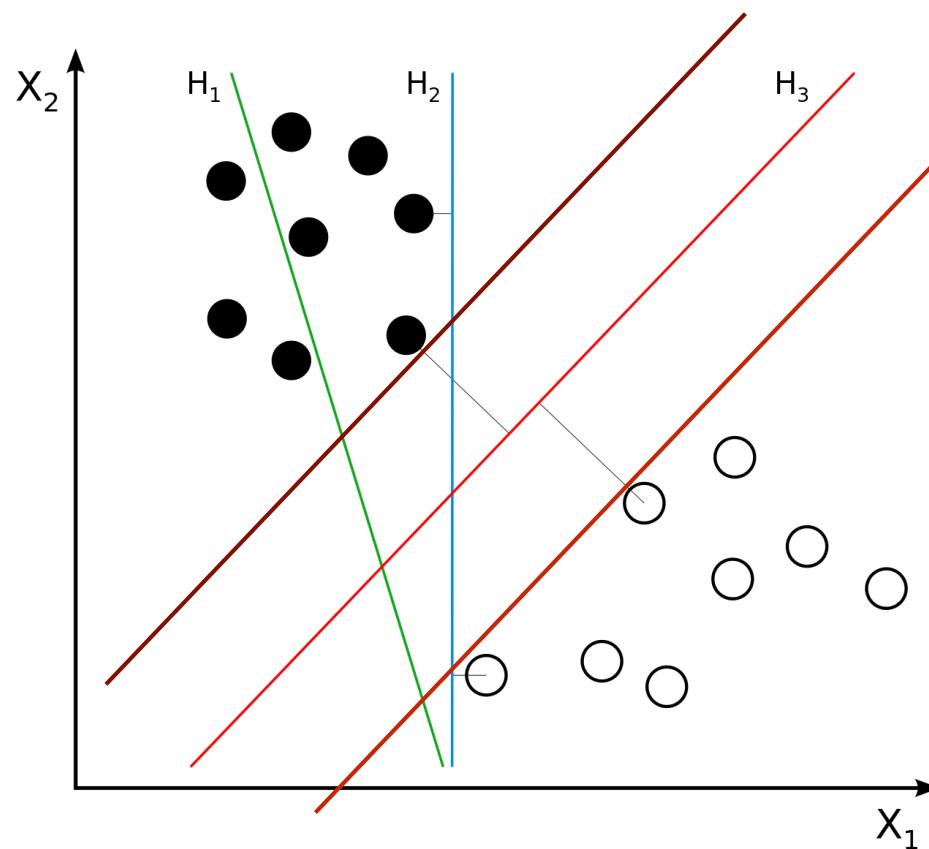
Support Vector Machine

- A Classification Problem
 - Linearly separable



Separating Hyper-planes

- There are many hyperplanes that might classify the data.
- One reasonable choice as the best hyperplane is the one that represents the largest separation, or margin, between the two classes. So we choose the hyperplane so that the distance from it to the nearest data point on each side is maximized.
- If such a hyperplane exists, it is known as the maximum-margin hyperplane



Hard Vs Soft Margin

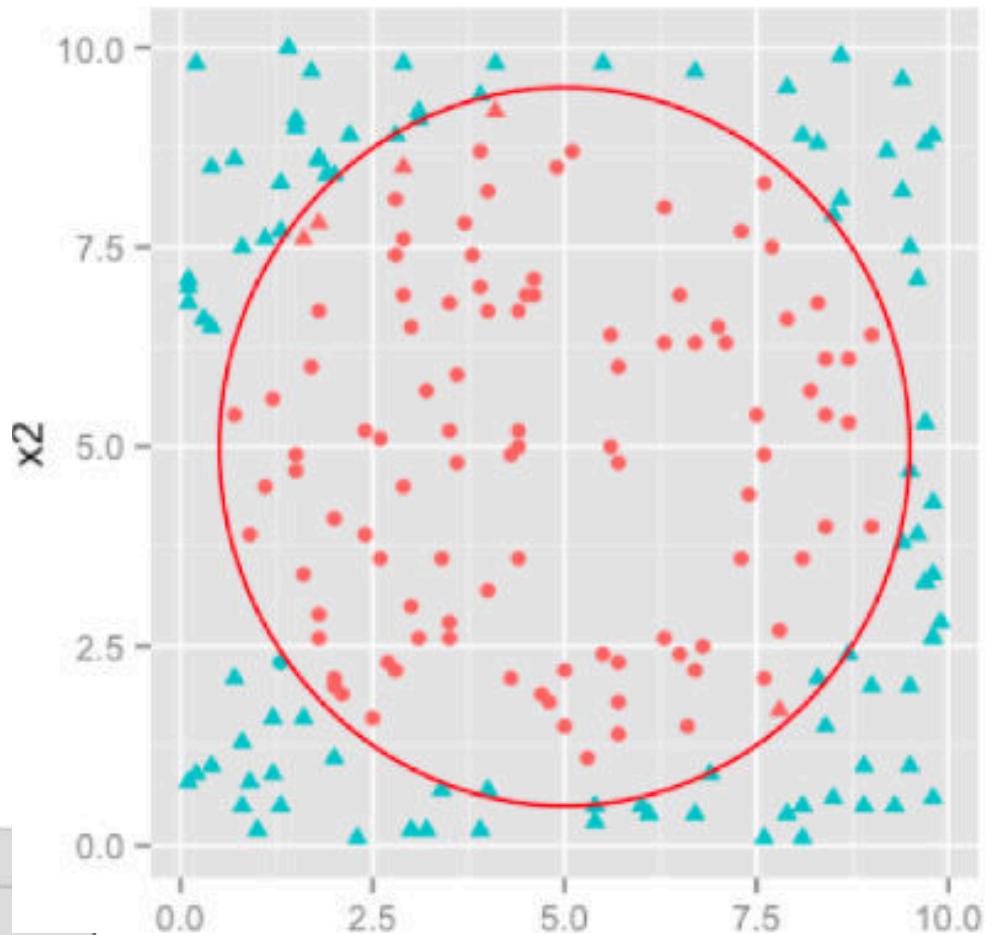
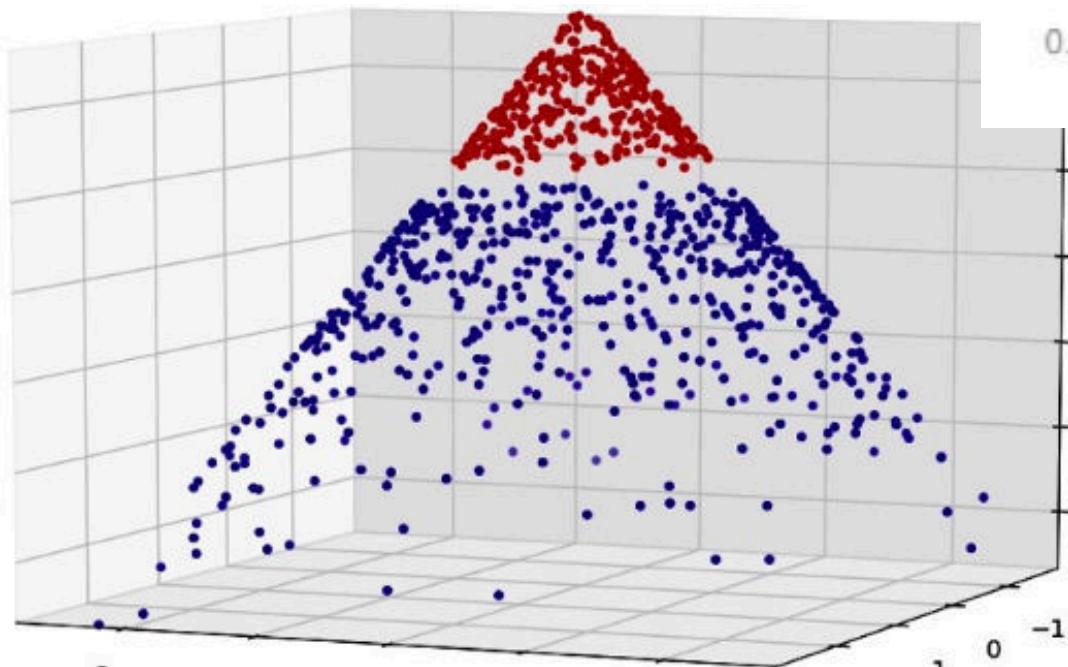
- If the data is linearly separable
 - we can select two parallel hyperplanes that separate the two classes of data, so that the distance between them is as large as possible.
 - The region bounded by these two hyperplanes is called the "margin", and
 - the maximum-margin hyperplane is the hyperplane that lies halfway between them.
- When data are NOT linearly separable
 - Hinge loss function: adds a penalty for crossing over the margin
 - Penalty is proportional to the distance from the margin

Kernel SVM

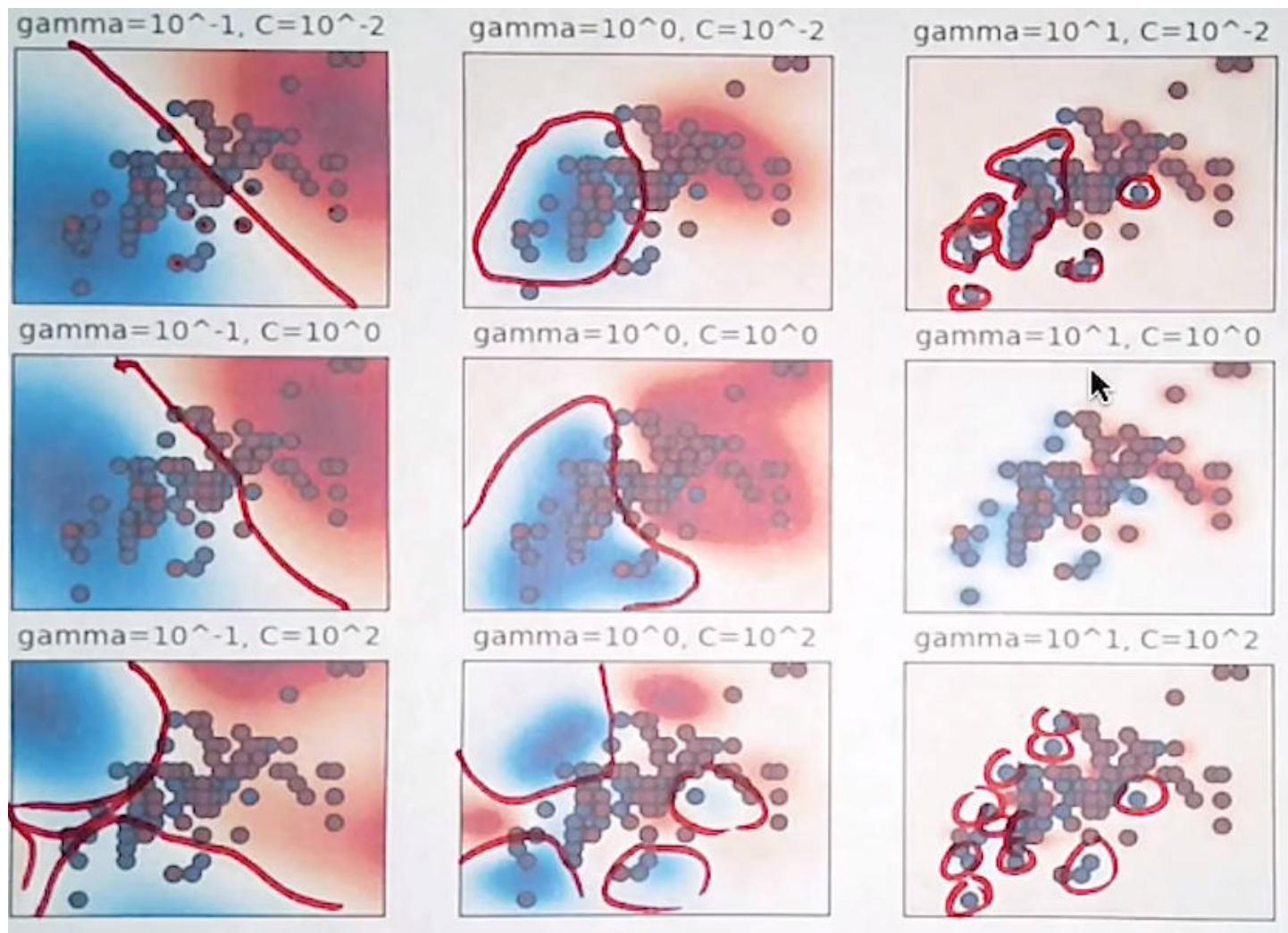
- “A complex pattern-classification problem, cast in a high-dimensional space nonlinearly, is more likely to be linearly separable than in a low-dimensional space, provided that the space is not densely populated” -
- T. Cover

Kernel SVM

- “A complex pattern-classification problem, cast in a high-dimensional space nonlinearly, is more likely to be linearly separable than in a low-dimensional space, provided that the space is not densely populated”
- Tom Cover

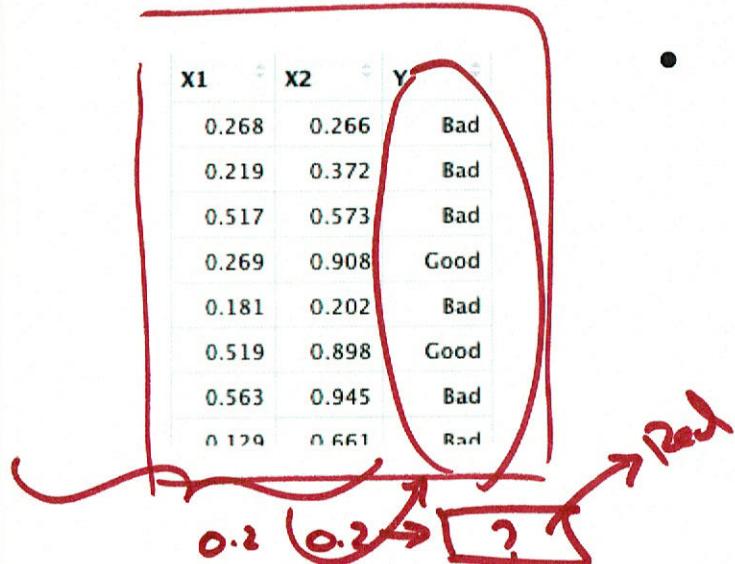


Parameters affecting the classification: C and Gamma



Classification and Regression

- Decision Trees can be used for both



- Classification

- Spam / not Spam
- Admit to ICU /not
- Lend money / deny
- Intrusion detections

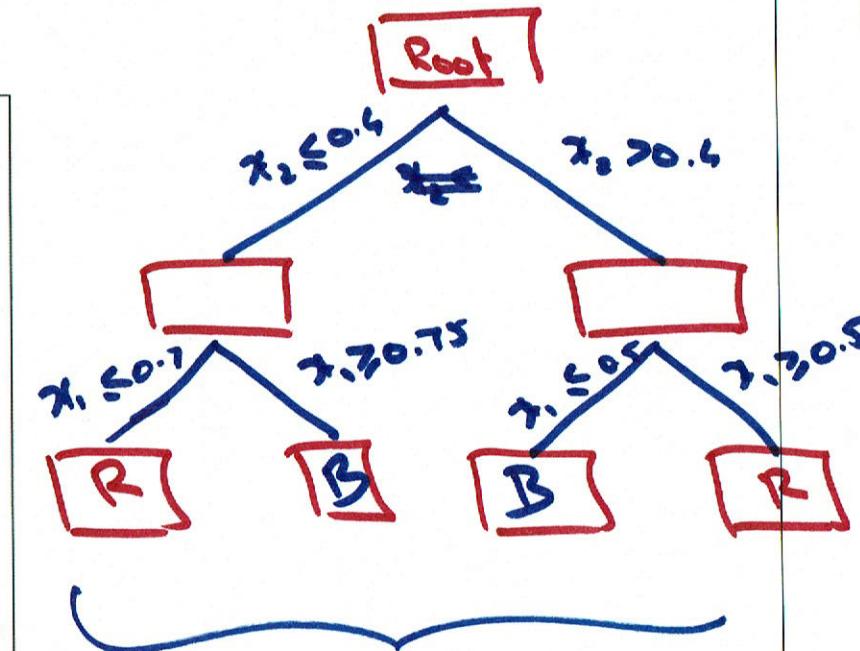
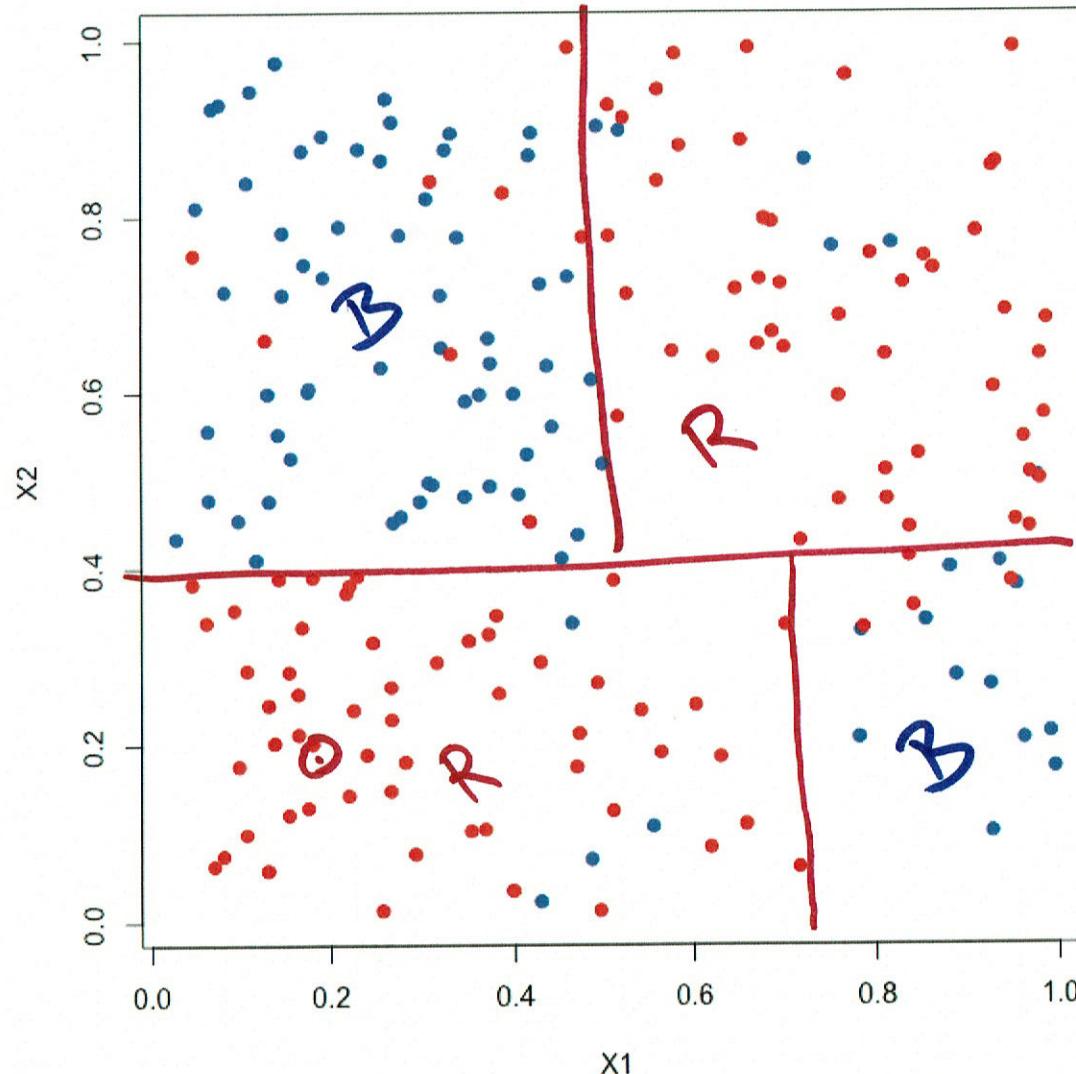
CART

X1	X2	Y
0.268	0.266	64.41
0.219	0.372	28.08
0.517	0.573	95.76
0.269	0.908	15.84
0.181	0.202	41.83
0.519	0.898	25.20
0.563	0.945	9.44
0.129	0.661	82.77

- Regression

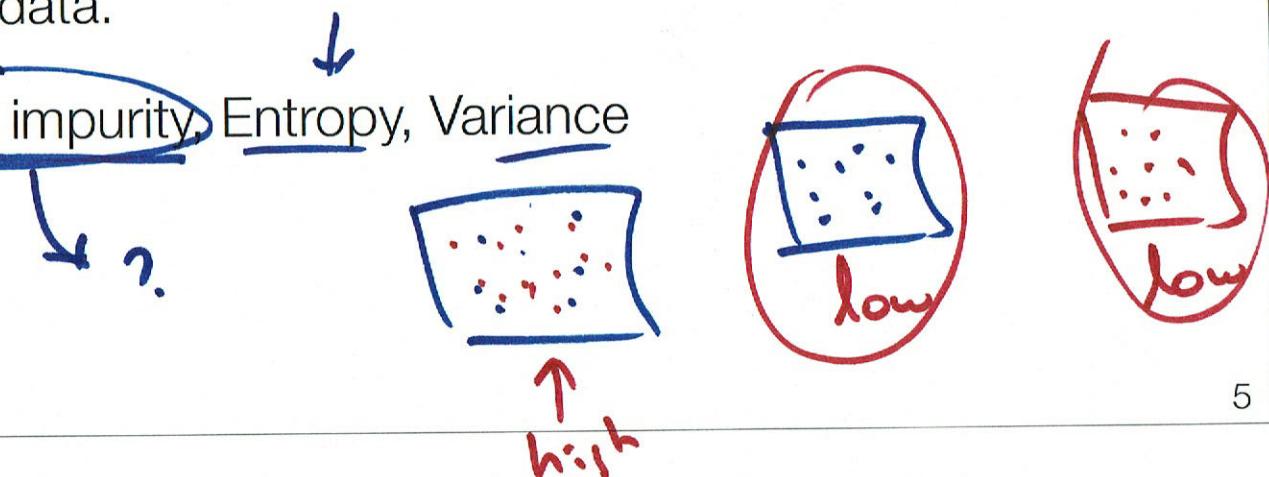
- Predict stock returns
- Pricing a house or a car
- Weather predictions (temp, rain fall etc)
- Economic growth predictions
- Predicting sports scores

Visualizing Classification as a Tree



Metrics

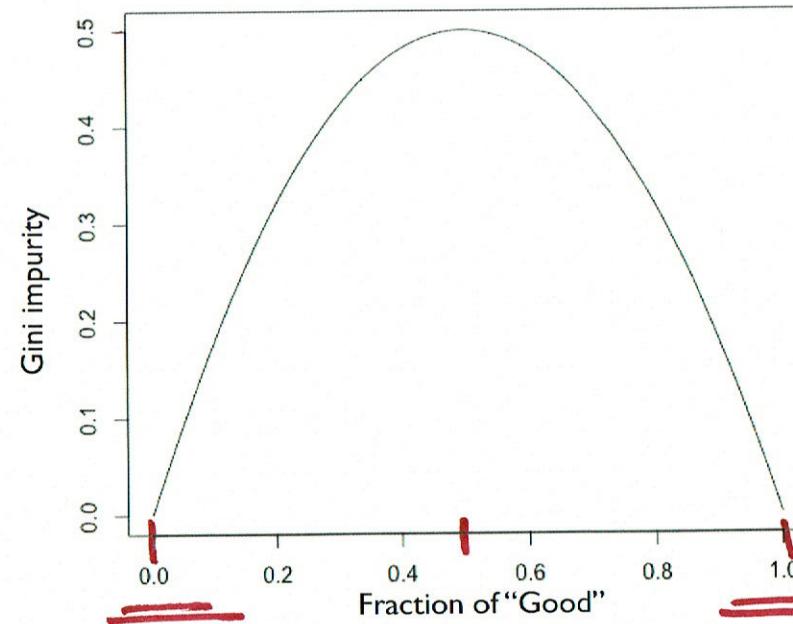
- Algorithms for constructing decision trees usually work top-down, by choosing a variable at each step that best splits the set of items.
- Different algorithms use different metrics for measuring “best”
- These metrics measure how similar a region or a node is. They are said to measure the impurity of a region.
- Larger these impurity metrics the larger the “dissimilarity” of a nodes/regions data.
- Examples: Gini impurity, Entropy, Variance

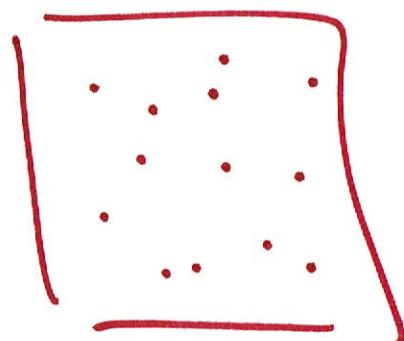
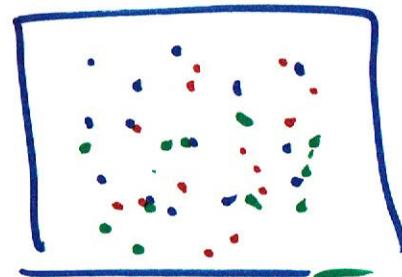


Gini impurity

- Used by the CART
- Is a measure of how often a randomly chosen element from the set would be incorrectly labeled if it was randomly labeled according to the distribution of labels in the subset.
- Can be computed by summing the probability of an item with label i being chosen (p_i), times the probability of a mistake ($1 - p_i$) in categorizing that item.
- Simplifying gives, the Gini impurity of a set:

$$1 - \sum_{i=1}^J p_i^2$$





$$P_1 \mid P_2 \mid P_3$$

$$\xrightarrow{P_1} ① \Rightarrow P_1(1-P_1)$$

$$\leftarrow P_1 P_2 + P_1 P_3$$

$$\xrightarrow{P_2} ② \Rightarrow P_2(1-P_2)$$

$$\leftarrow P_2 P_3 + P_2 P_1$$

$$\xrightarrow{P_3} ③ \Rightarrow P_3(1-P_3)$$

$$\leftarrow \underbrace{P_3 P_1 + P_3 P_2}$$

$$\sum P_i(1-P_i)$$

$$\sum P_i - \sum P_i^2 \Rightarrow$$

$$1 - \sum P_i^2$$

CART: An Example

Handwritten notes:

- Row 1: Cust_ID, Gender, Occupation, Age, Target
- Row 2: M, Sal, 22, 1
- Row 3: M, Sal, 22, 0
- Row 4: M, Self-Emp, 23, 1
- Row 5: M, Self-Emp, 23, 0
- Row 6: M, Self-Emp, 24, 1
- Row 7: M, Self-Emp, 24, 0
- Row 8: F, Sal, 25, 1
- Row 9: F, Sal, 25, 0
- Row 10: F, Self-Emp, 26, 0

Root node : $P_1 = 0.4 \quad P_2 = 0.6$

$$GI = 1 - (0.4)^2 - (0.6)^2 = 0.48$$

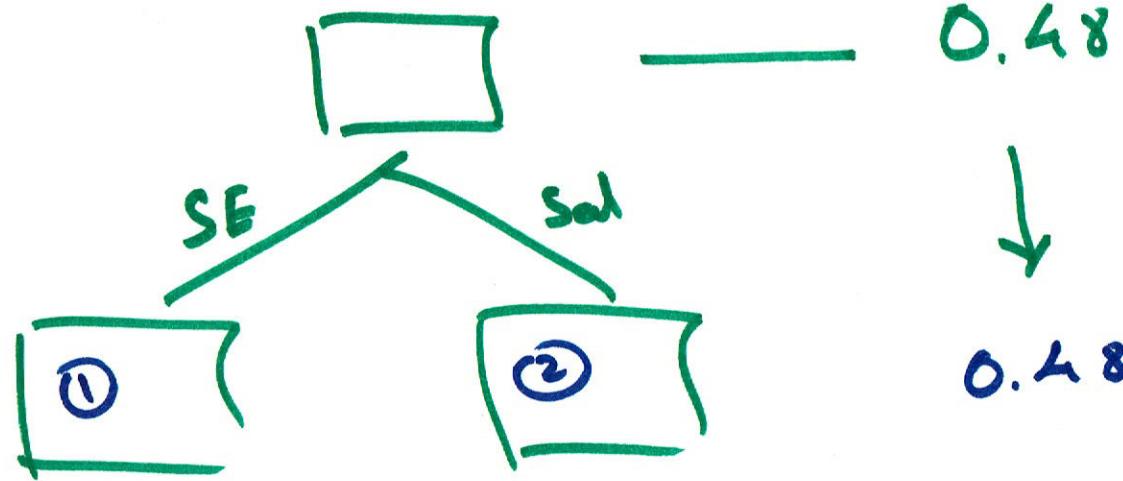
$P_1 = 0.5 \quad P_2 = 0.5$

$$1 - 0.5^2 - 0.5^2 = 0.5$$

$P_1 = 0.25 \quad P_2 = 0.75$

$$1 - 0.25^2 - 0.75^2 = 0.375$$

$$GI = \frac{6}{10}(0.5) + \frac{4}{10}(0.375) \Rightarrow 0.45$$

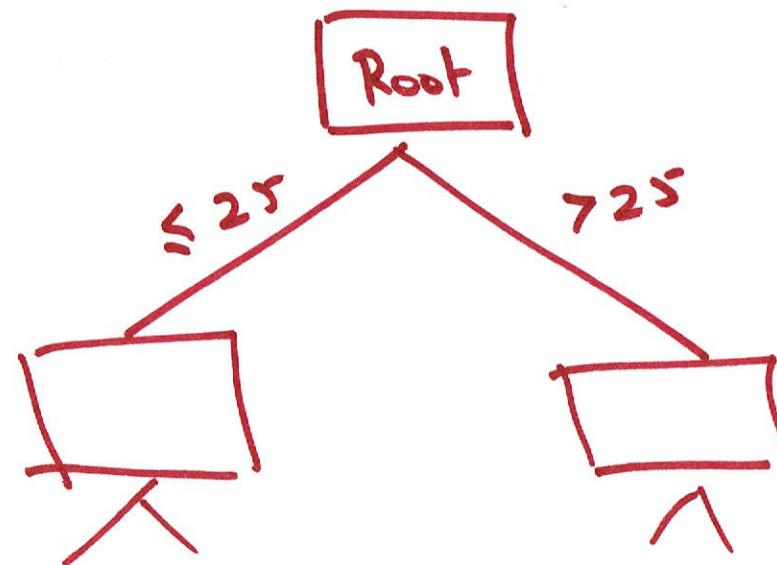


$$\begin{array}{c}
 \textcircled{1} \quad G.I = 1 - 0.4^2 - 0.6^2 \\
 \textcircled{2} \quad G.I = 1 - 0.4^2 - 0.6^2 \\
 = 0.48 \qquad \qquad \qquad = 0.48
 \end{array}$$

$G.I = \frac{\sum}{10} (0.48) + \frac{\sum}{10} (0.48) = 0.48$

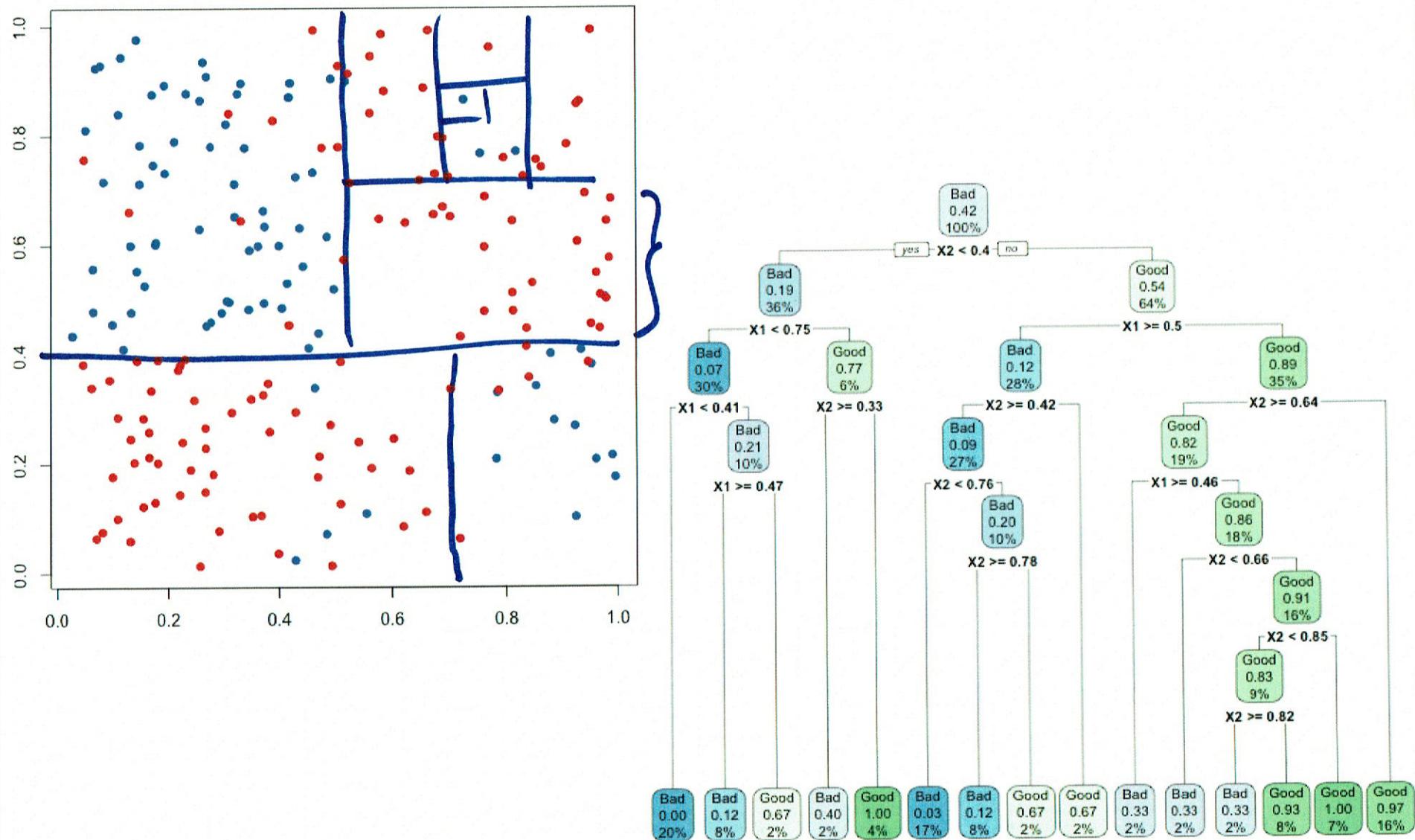
	Left	right	Gini Split
$\leq 22, > 22$	0.5	0.47	0.48
$\leq 23, > 23$	0.5	0.44	0.47
$\leq 24, > 24$	0.5	0.38	0.45
$\leq 25, > 25$	0.5	0	0.40

Gain ≈ 0.08

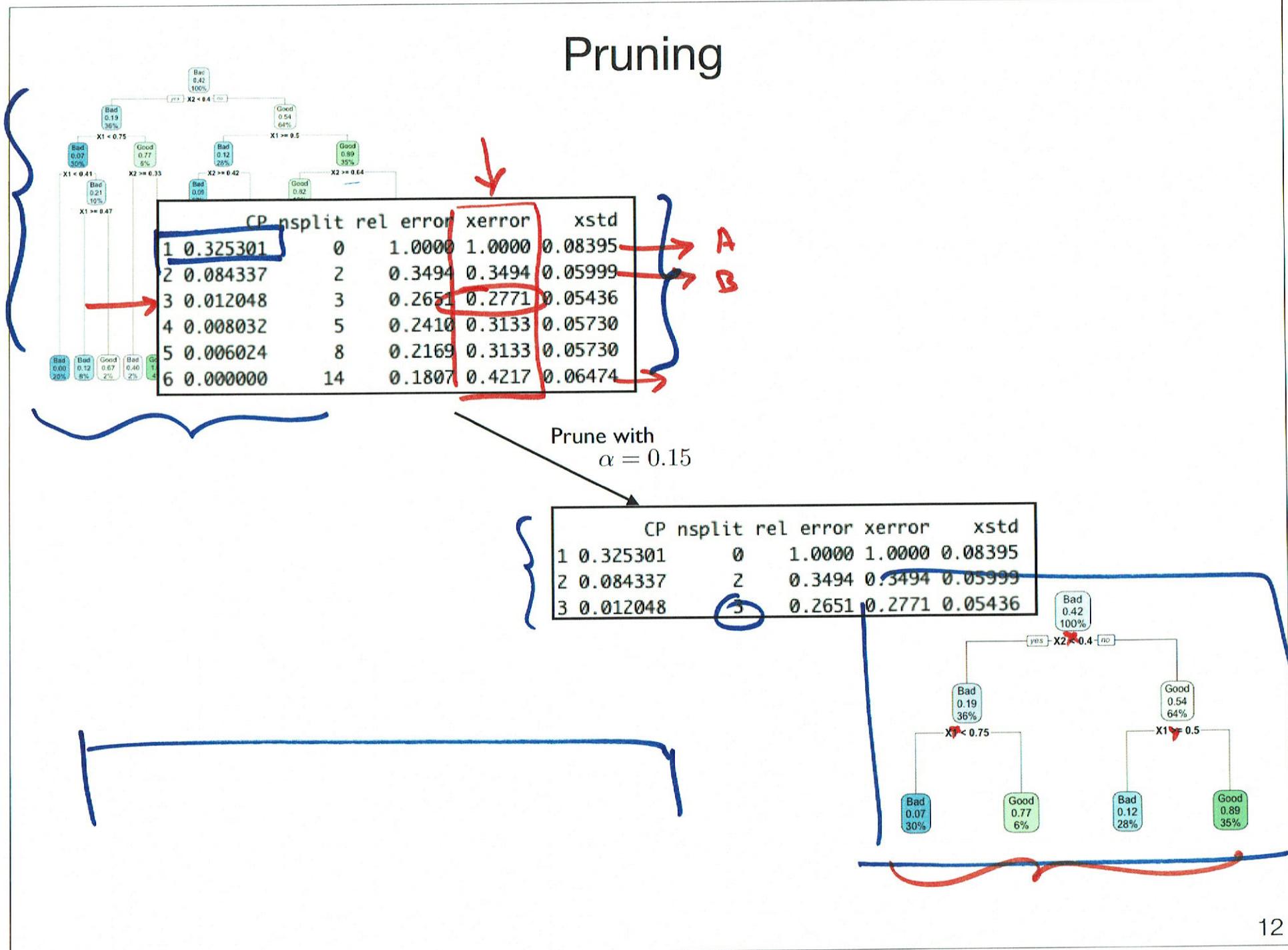


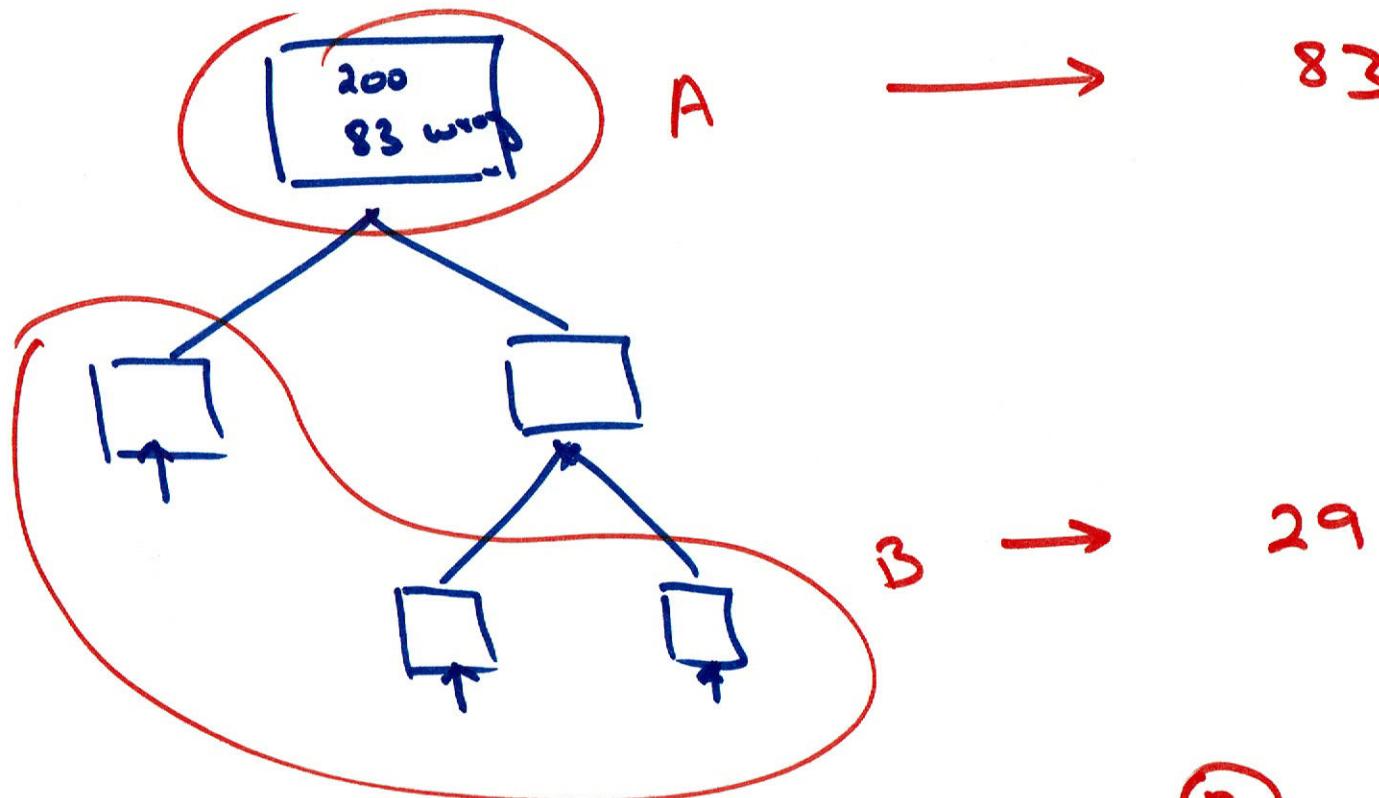
0.48
↓
0.40

Overfitting in Decision Trees



Pruning





(A) 1 node → (B) 3 nodes

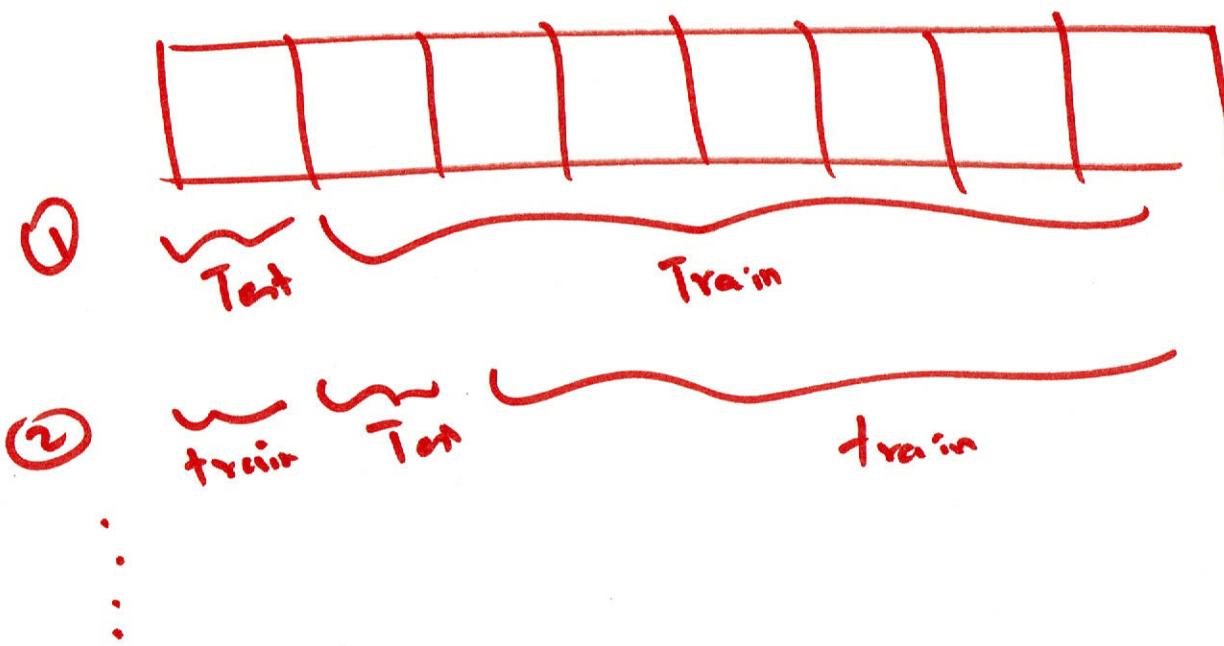
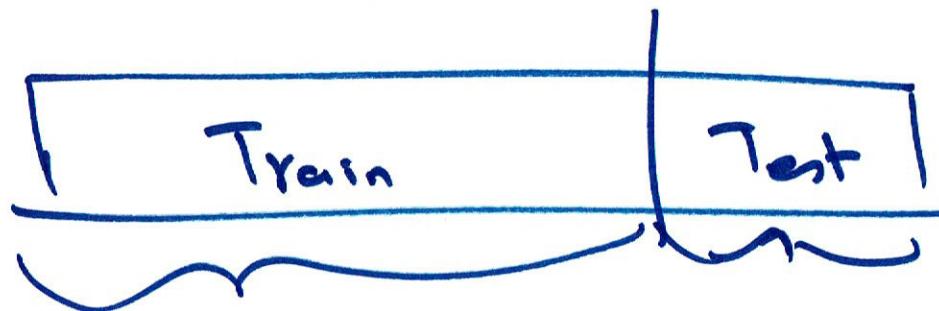
83 → 29

rel. error dec =

$$\left[\frac{54}{83} = 2\alpha \right]$$

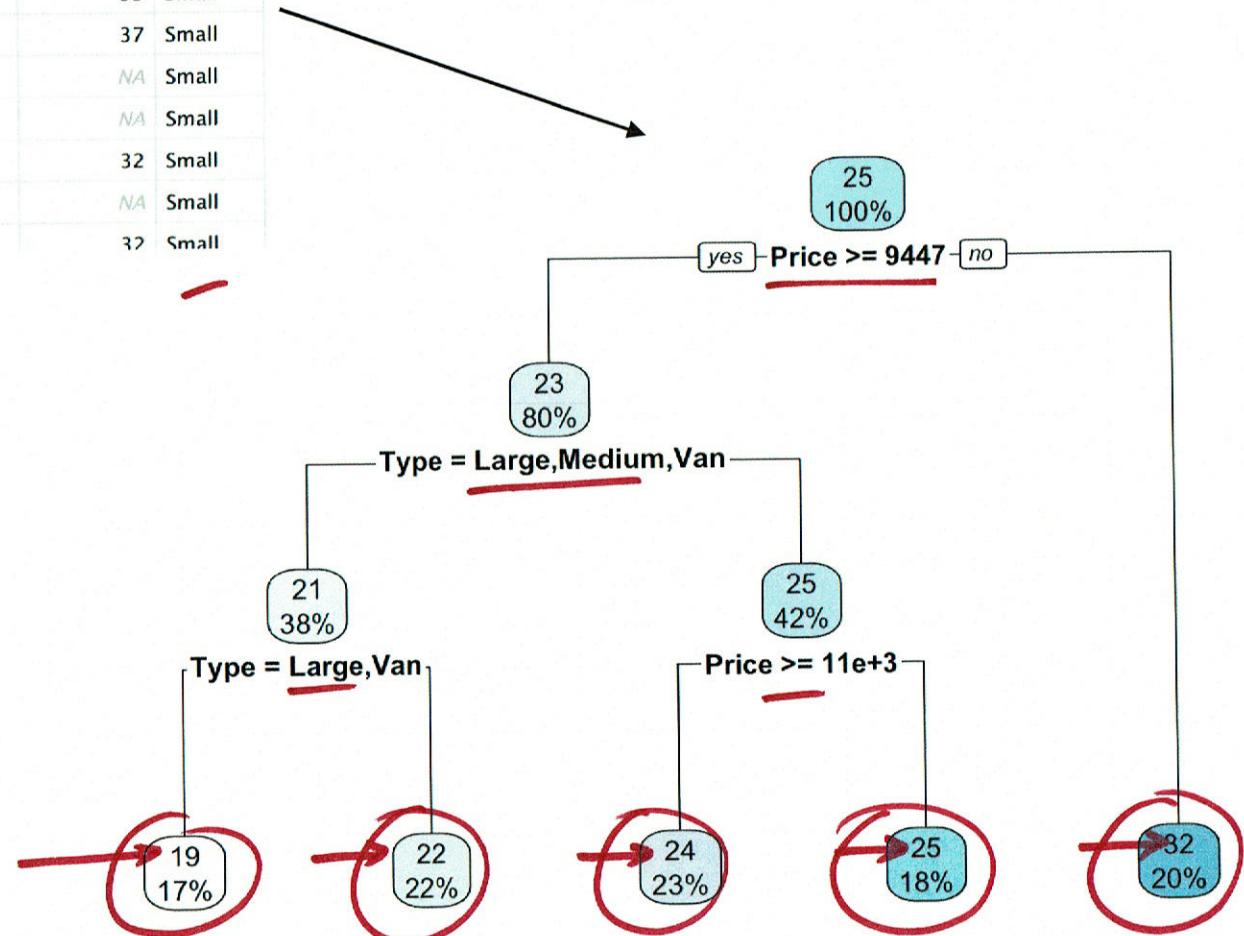
$$(CP) \Rightarrow \alpha = \frac{54}{83} \times \frac{1}{2} = 0.325$$

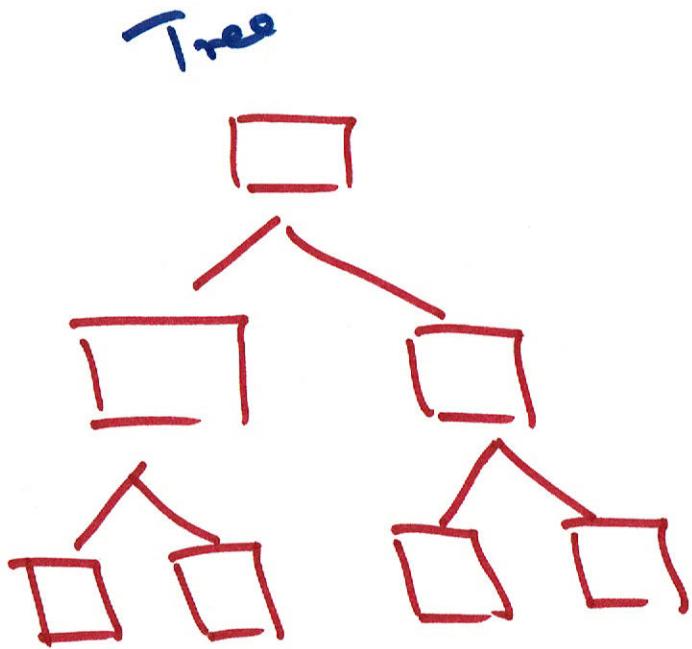
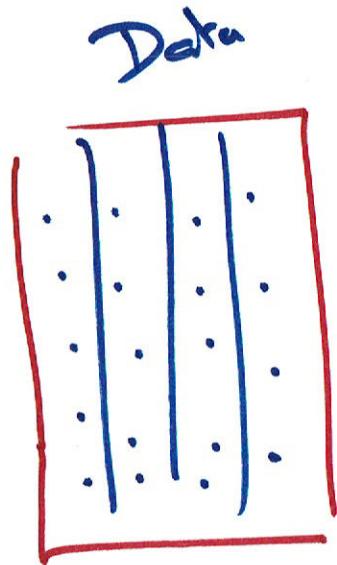
$$d = 0.15$$



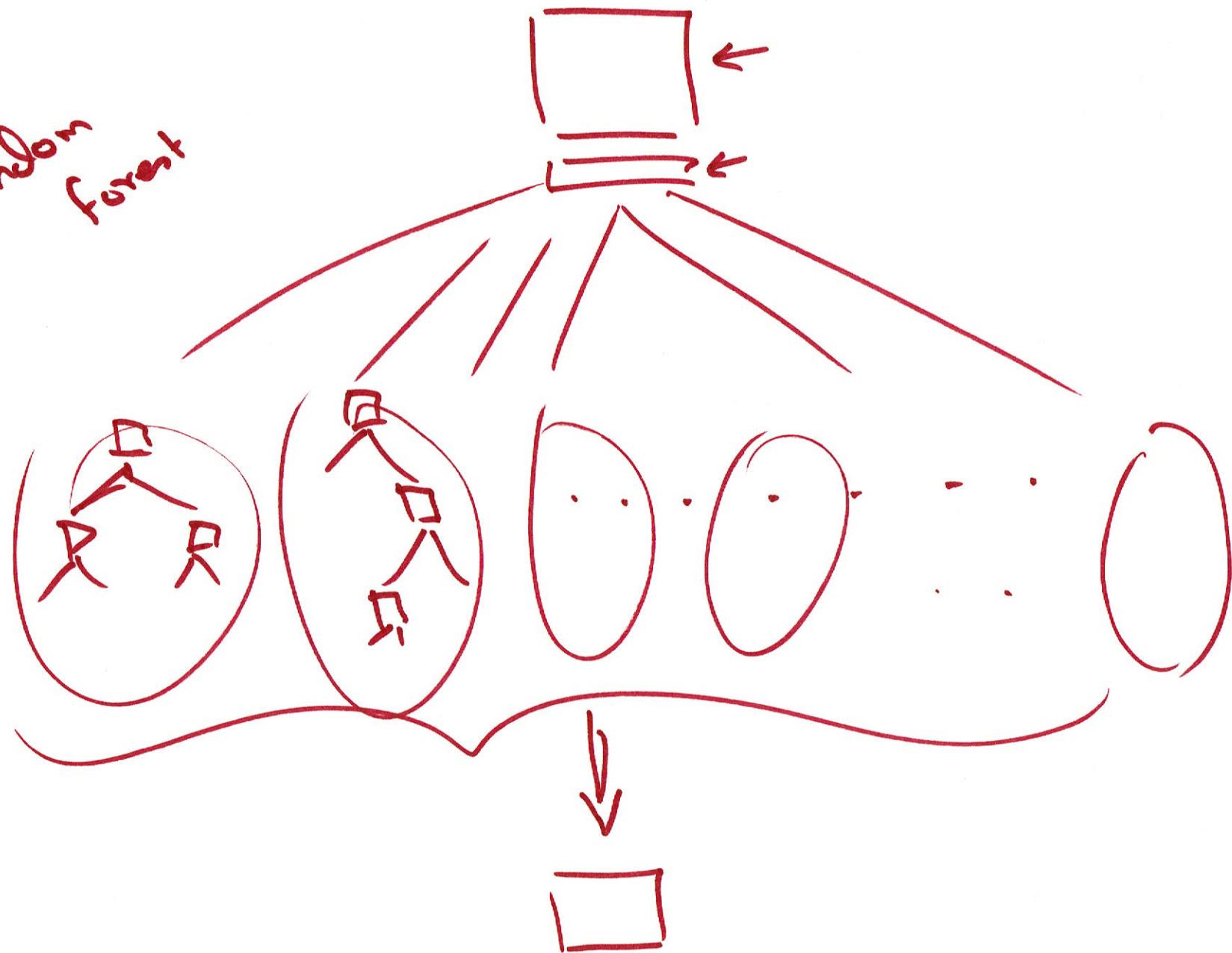
Regression Trees

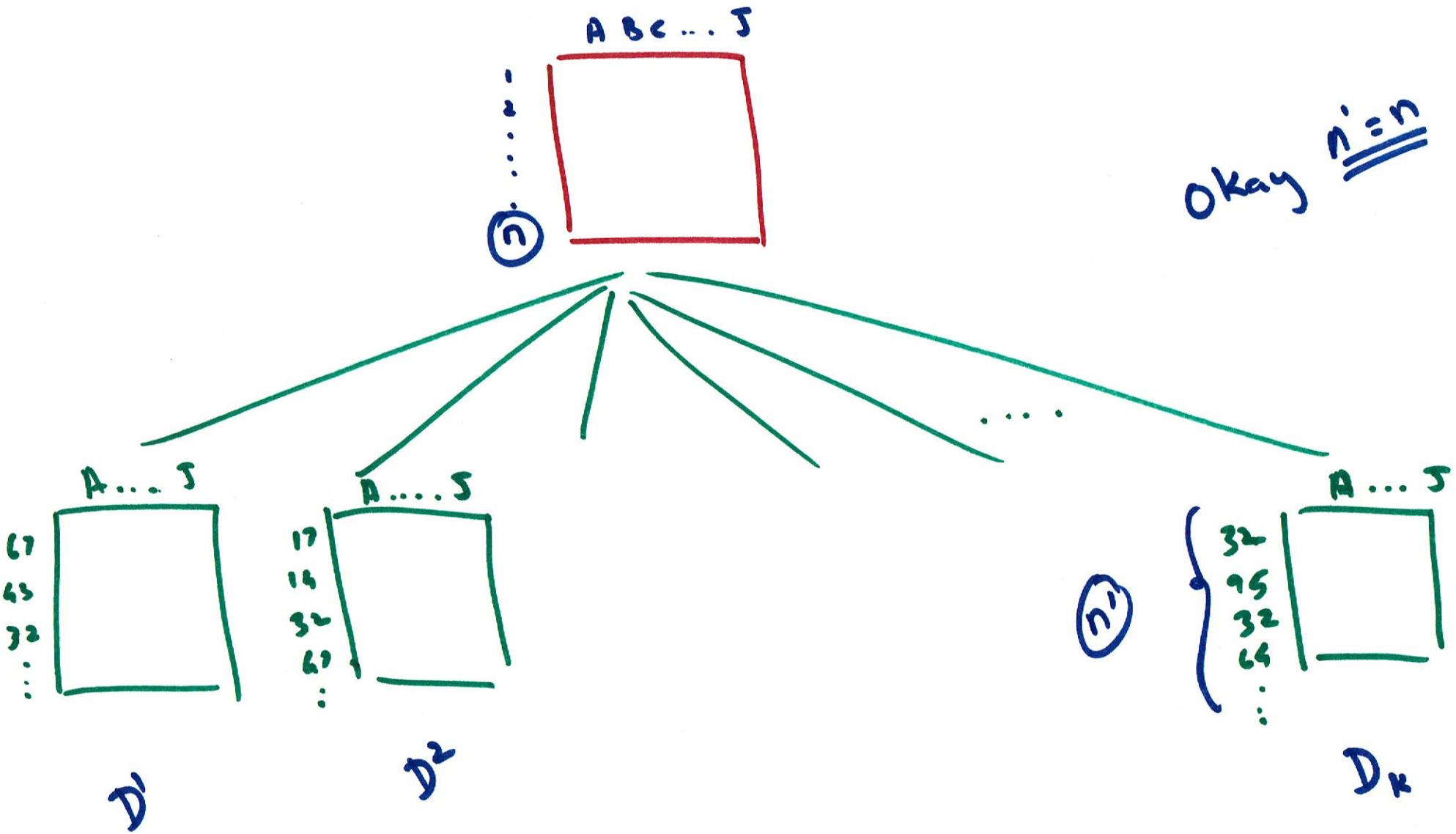
	Price	Country	Reliability	Mileage	Type
Acura Integra 4	11950	Japan	Much better	NA	Small
Dodge Colt 4	6851	Japan	NA	NA	Small
Dodge Omni 4	6995	USA	Much worse	NA	Small
Eagle Summit 4	8895	USA	better	33	Small
Ford Escort 4	7402	USA	worse	33	Small
Ford Festiva 4	6319	Korea	better	37	Small
GEO Metro 3	6695	Japan	NA	NA	Small
GEO Prizm 4	10125	Japan/USA	Much better	NA	Small
Honda Civic 4	6635	Japan/USA	Much better	32	Small
Hyundai Excel 4	5899	Korea	worse	NA	Small
Mazda Protege 4	6599	Japan	Much better	32	Small

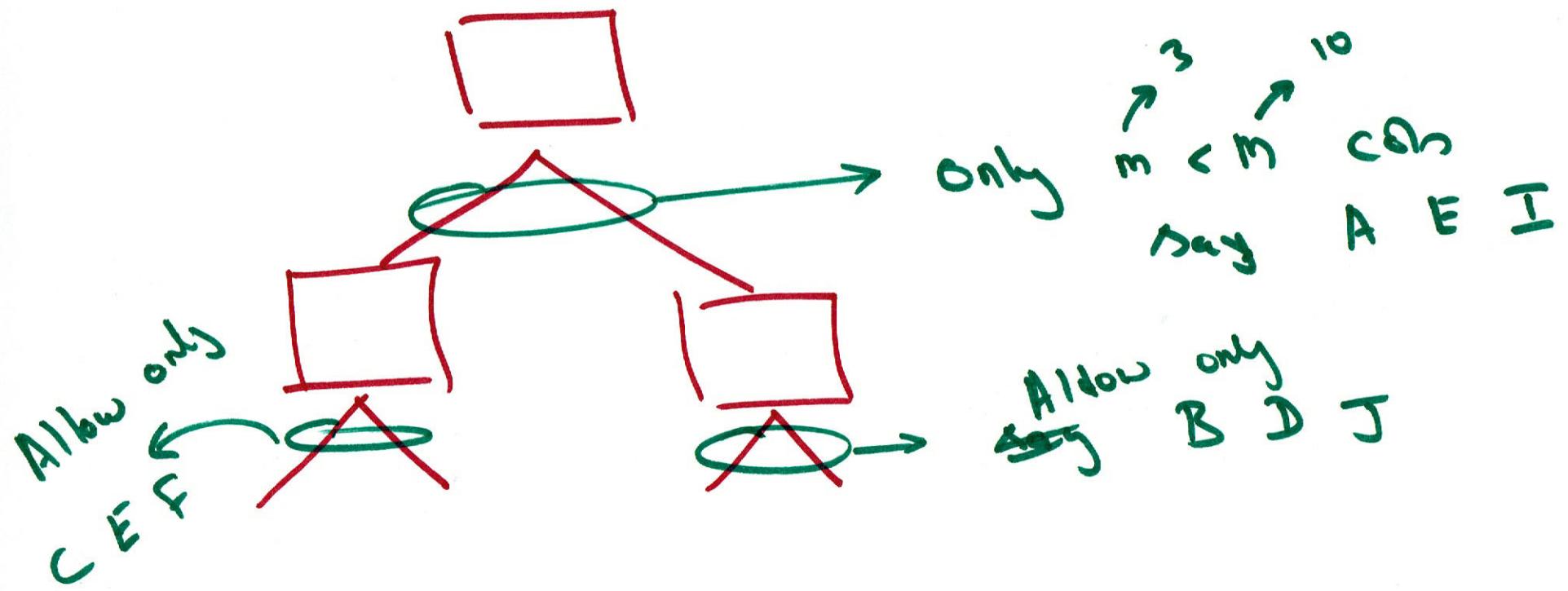
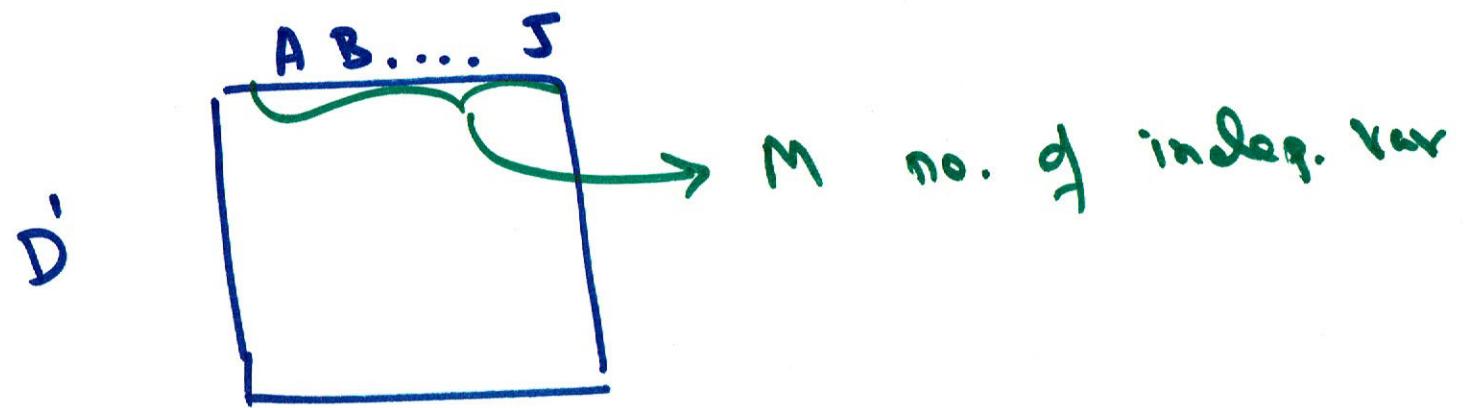




Random
Forest



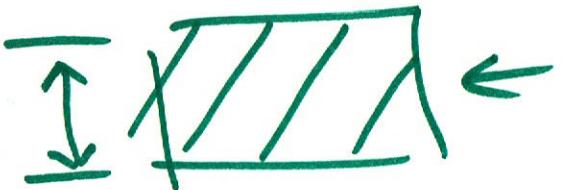




Random forests

- Random Sampling with replacement
- For each subset build a decision tree. However, only use m randomly pick independent variables for each node's branching possibilities.
 - Do not prune
- While predicting:
 - Use each tree to make individual predictions
 - Combine predictions using voting:
 - Means for regression
 - Modes for classification

Say $M = 10 \Rightarrow A \text{ B.C.D.O.O.O. } \textcircled{3}$

- | ① $m = 10 \rightarrow$ high tree correlation
| ↓ 
| ② $m = 2 \rightarrow$ your trees are weak.

Clustering

Machine Learning

Data we will work with

- Customer Spend Data
 - AVG_Mthly_Spend: The average monthly amount spent by customer
 - No_of_Visits: The number of times a customer visited in a month
 - Item Counts: Count of Apparel, Fruits and Vegetable, Staple Items purchased

Cust_ID	Name	Avg_Mthly_Spend	No_Of_Visits	Apparel_Items	FnV_Items	Staples_Items
1	A	10000	2	1	1	0
2	B	7000	3	0	10	9
3	C	7000	7	1	3	4
4	D	6500	5	1	1	4
5	E	6000	6	0	12	3
6	F	4000	3	0	1	8
7	G	2500	5	0	11	2
8	H	2500	3	0	1	1
9	I	2000	2	0	2	2
10	J	1000	4	0	1	7

- Can we cluster similar customers together?

Connectivity Based: Hierarchical Clustering

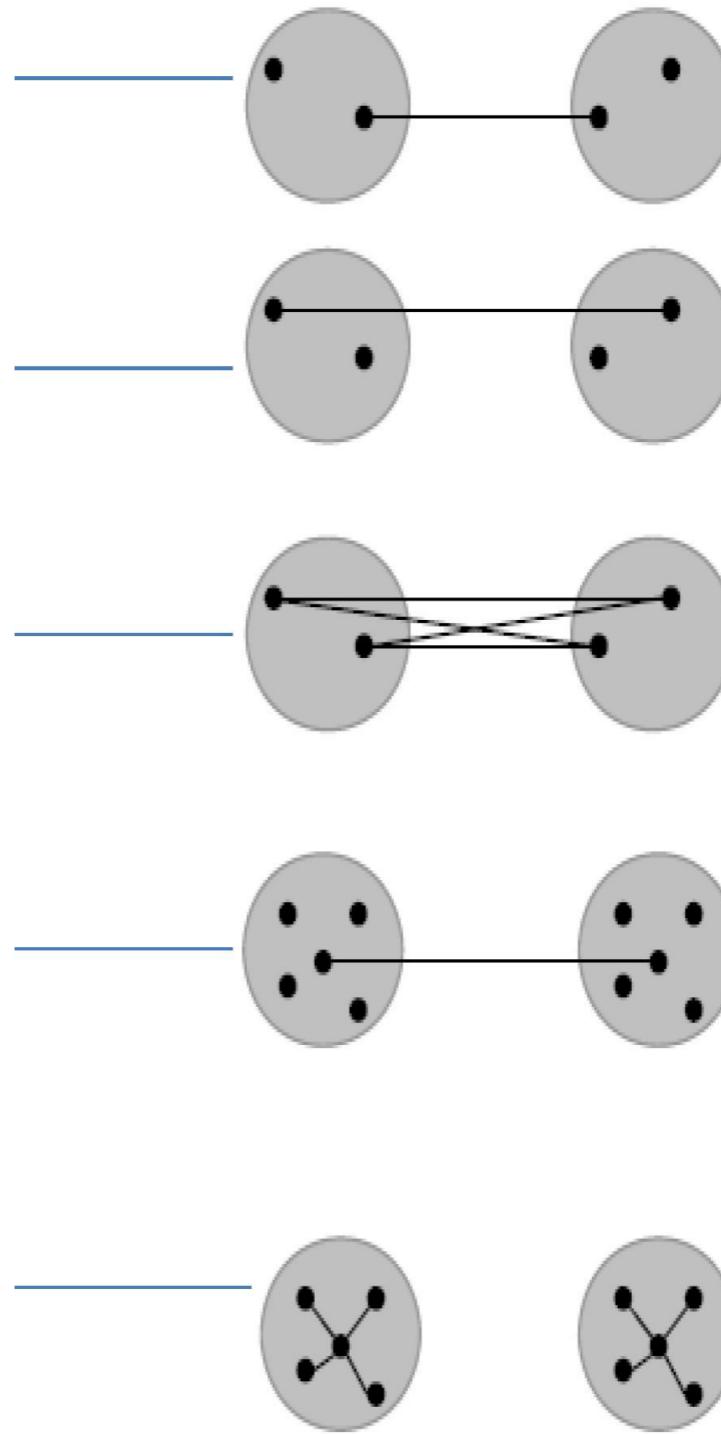
- Hierarchical Clustering techniques create clusters in a hierarchical tree like structure
- Any type of distance measure can be used as a measure of similarity
- Cluster tree like output is called Dendogram
- Techniques either start with individual objects and sequentially combine them (Agglomerative), or start from one cluster of all objects and sequentially divide them (Divisive)

Agglomerative

- Starts with each object as a cluster of one record each
- Sequentially merges 2 closest records by distance as a measure of similarity to form a cluster.
- How would we measure distance between two clusters?

Distance between clusters

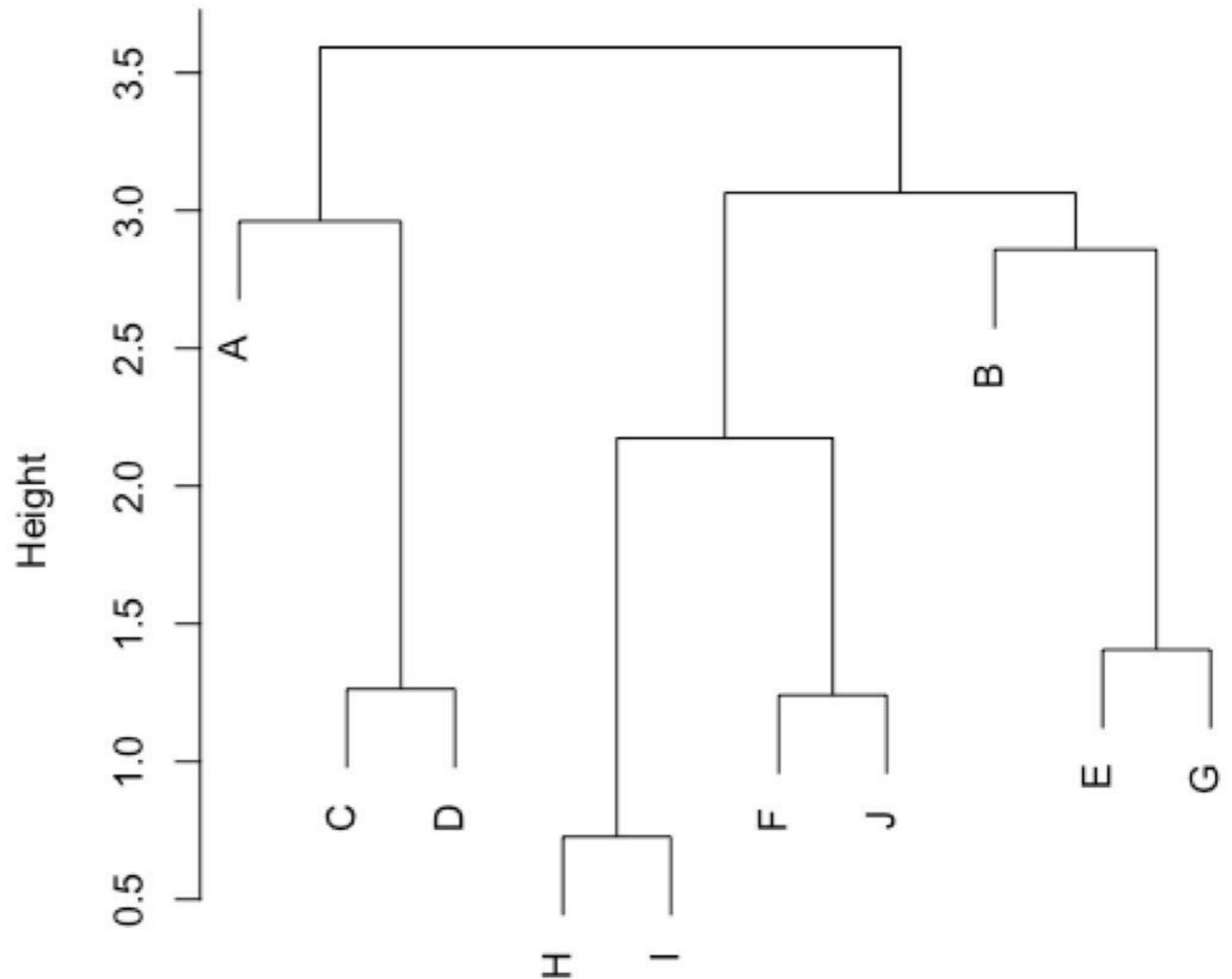
- Single linkage – Minimum distance or Nearest neighbor
- Complete linkage – Maximum distance or Farthest distance
- Average linkage – Average of the distances between all pairs
- Centroid method – combine cluster with minimum distance between the centroids of the two clusters
- Ward's method – Combine clusters with which the increase in within cluster variance is to the smallest degree



Distance between objects

	1	2	3	4	5	6	7	8	9
2	4.252								
3	3.411	3.838							
4	2.512	3.473	1.264						
5	4.268	2.697	2.922	3.204					
6	3.980	2.208	3.579	2.853	3.431				
7	4.378	3.021	3.384	3.345	1.406	3.171			
8	3.396	3.603	3.663	2.927	3.244	2.350	2.457		
9	3.534	3.395	4.054	3.213	3.482	2.175	2.613	0.727	
10	4.550	2.967	3.591	3.041	3.408	1.241	2.800	2.115	2.057

Cluster Dendrogram



Centroid based: K-Means Clustering

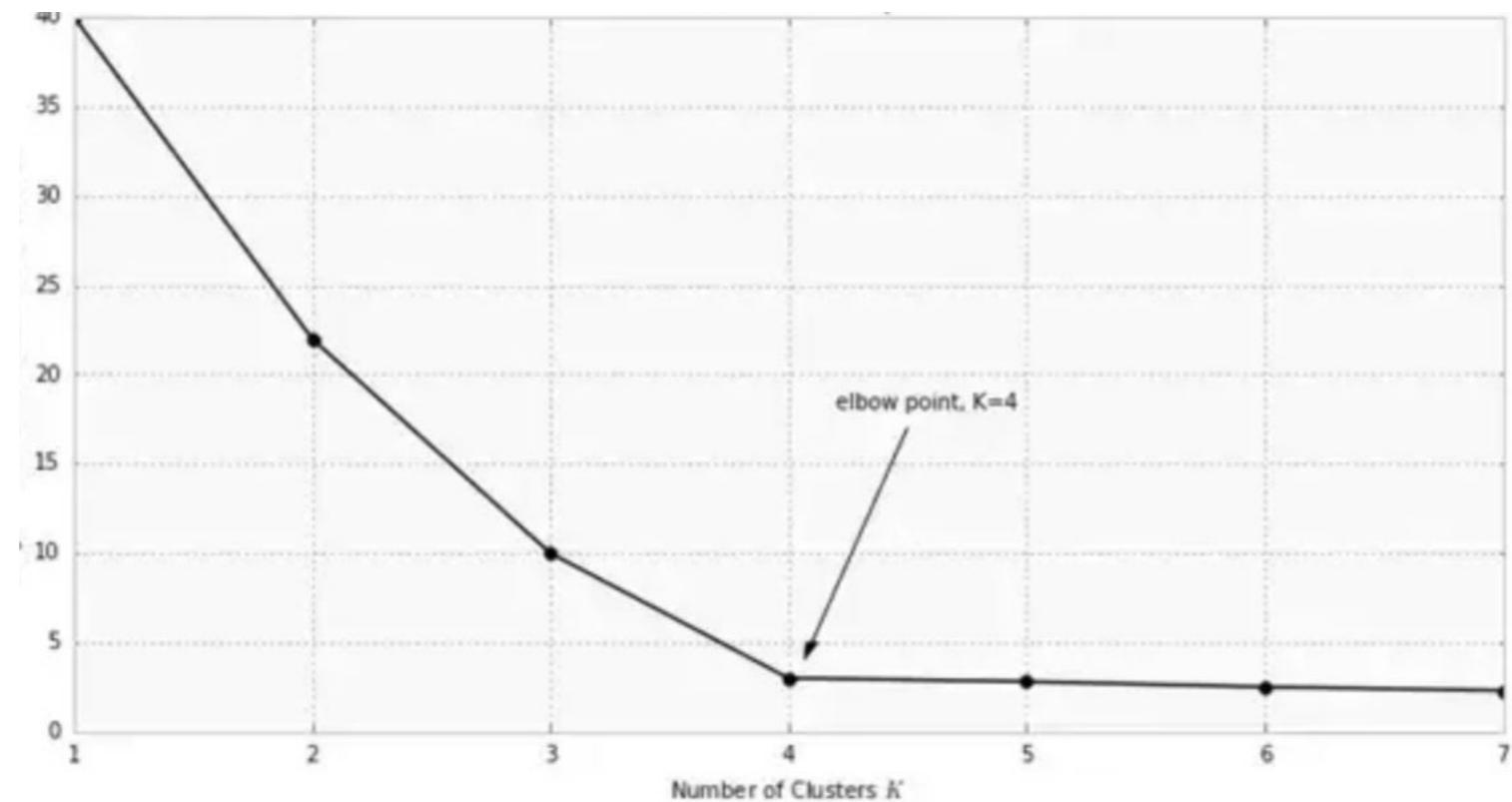
- K-Means is probably the most used clustering technique
- Aims to partition the n observations into k clusters so as to minimize the within-cluster sum of squares (i.e. variance).
- Computationally less expensive compared to hierarchical techniques.
- Have to pre-define K , the no of clusters

Lloyd's algorithm

1. Assume K Centroids
2. Compute Squared Euclidian distance of each objects with these K centroids. Assign each to the closest centroid forming clusters.
3. Compute the new centroid (mean) of each cluster based on the objects assigned to each clusters.
4. Repeat 2 and 3 till convergence: usually defined as the point at which there is no movement of objects between clusters

Choosing the optimal K

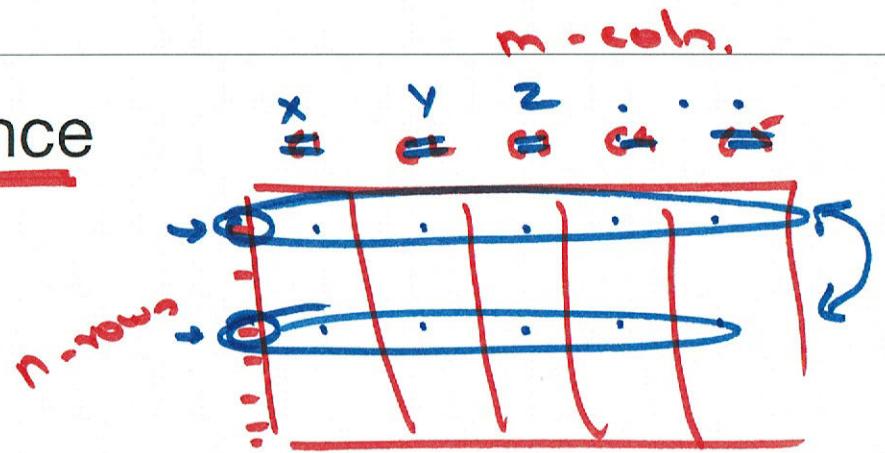
- Usually subjective, based on striking a good balance between compression and accuracy
- The “elbow” method is commonly used



Lloyd's algorithm

1. Assume K Centroids
2. Compute Squared Euclidian distance of each objects with these K centroids. Assign each to the closest centroid forming clusters.
3. Compute the new centroid (mean) of each cluster based on the objects assigned to each clusters.
4. Repeat 2 and 3 till convergence: usually defined as the point at which there is no movement of objects between clusters

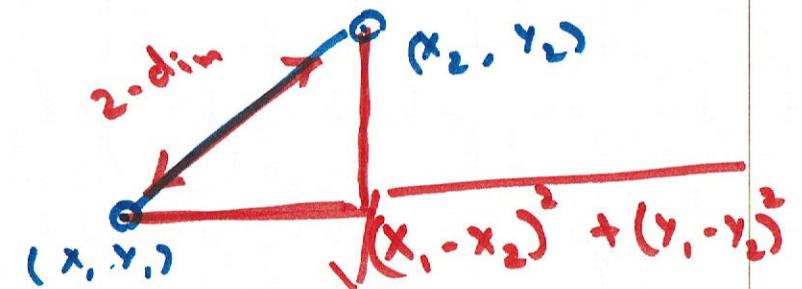
Distance



- To define "similarity" you need a measure of distance
- Examples of common distance measures
 - Manhattan Distance
 - Euclidean Distance
 - Chebyshev Distance

$$|x_1 - x_2| + |y_1 - y_2| + |z_1 - z_2| + \dots$$

2-dim


$$\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

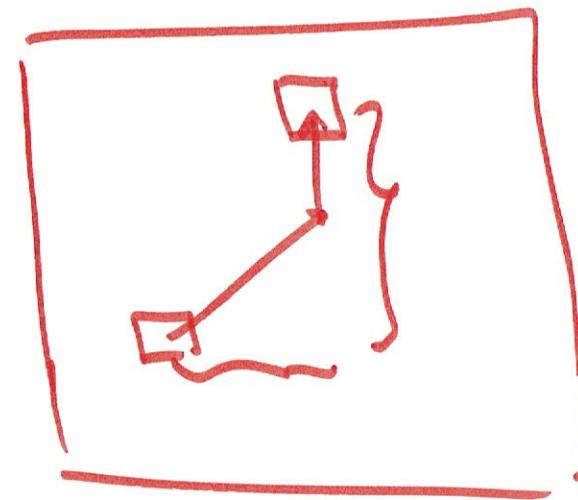
n-dim

$$\sqrt{[(x_1 - x_n)^2 + (y_1 - y_n)^2 + (z_1 - z_n)^2 + \dots]}$$

Chebyshev (or) the Chebboard dist.

m-dim

$$\max(|x_1 - x_2|, |y_1 - y_2|, |z_1 - z_2|, \dots)$$



Minkowski

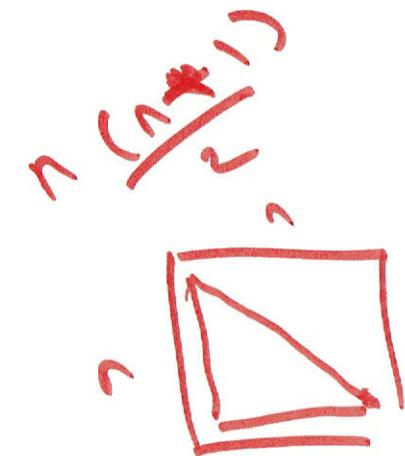
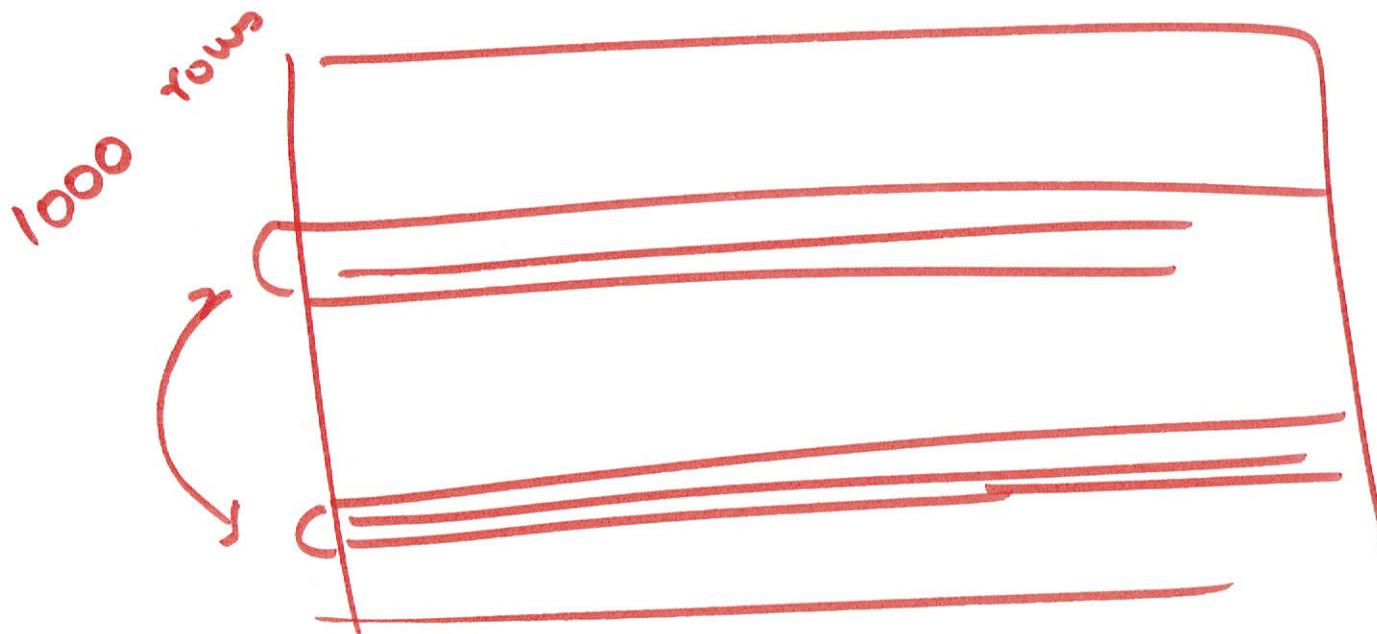


$$\left(\sum_{i=1}^n |x_i - y_i|^p \right)^{1/p}$$

\Rightarrow Euclidean dist.

\Rightarrow Manhattan dist.

\Rightarrow Chebyshev dist.



Connectivity based in roughly begin by computing

$500,000$

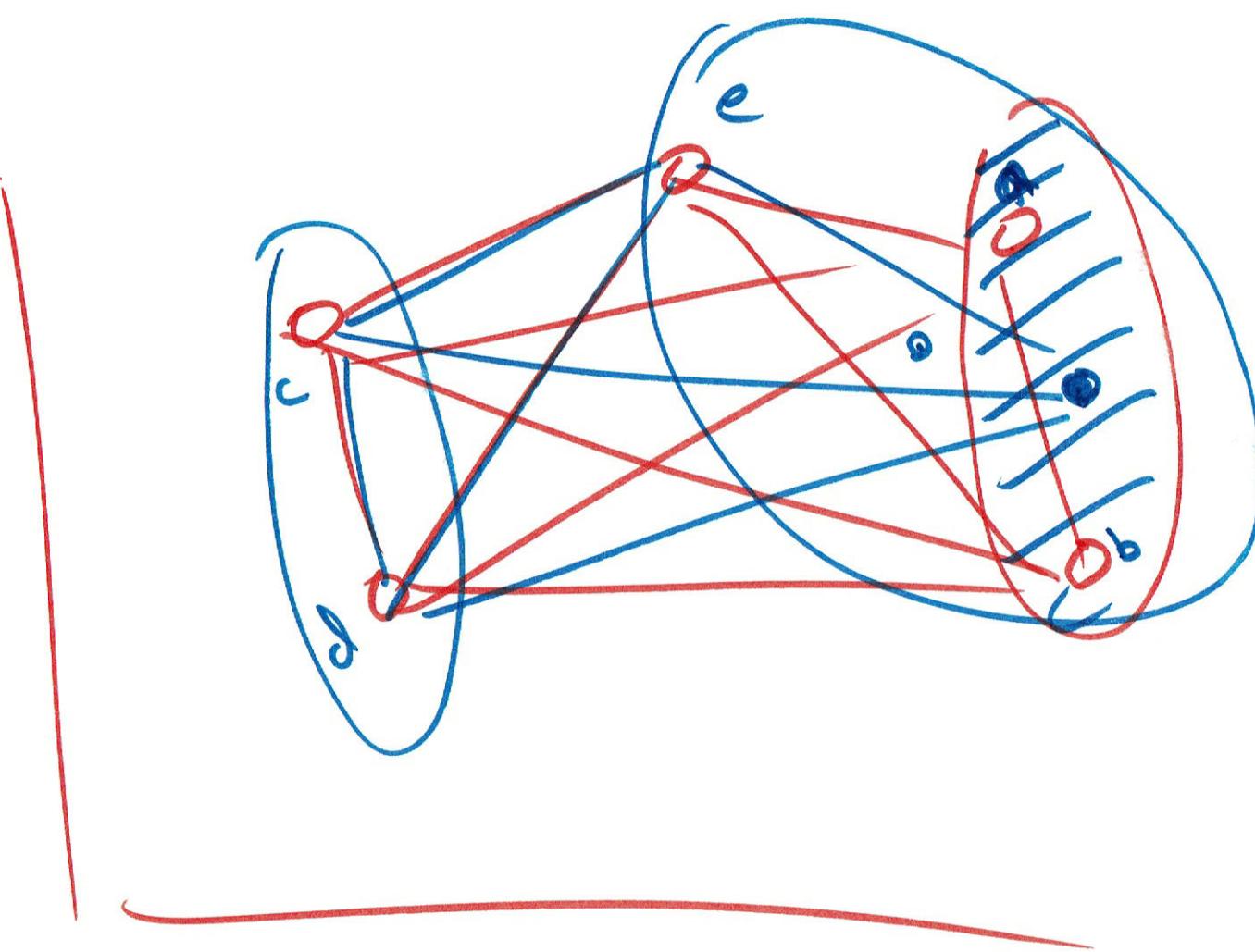
$$\text{dist. } \frac{n \times (n+1)}{2}$$

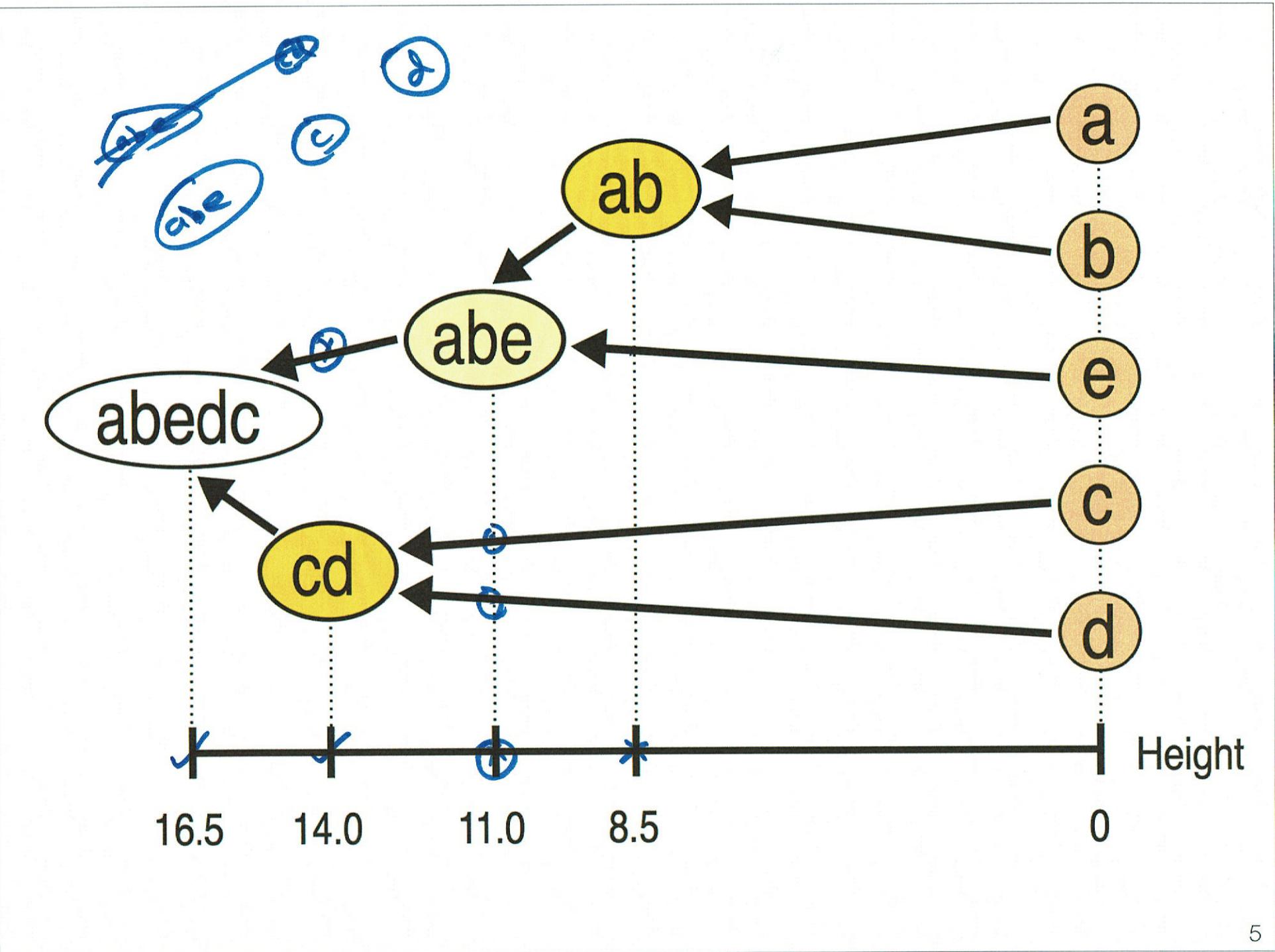
Centroid based in
5-groups

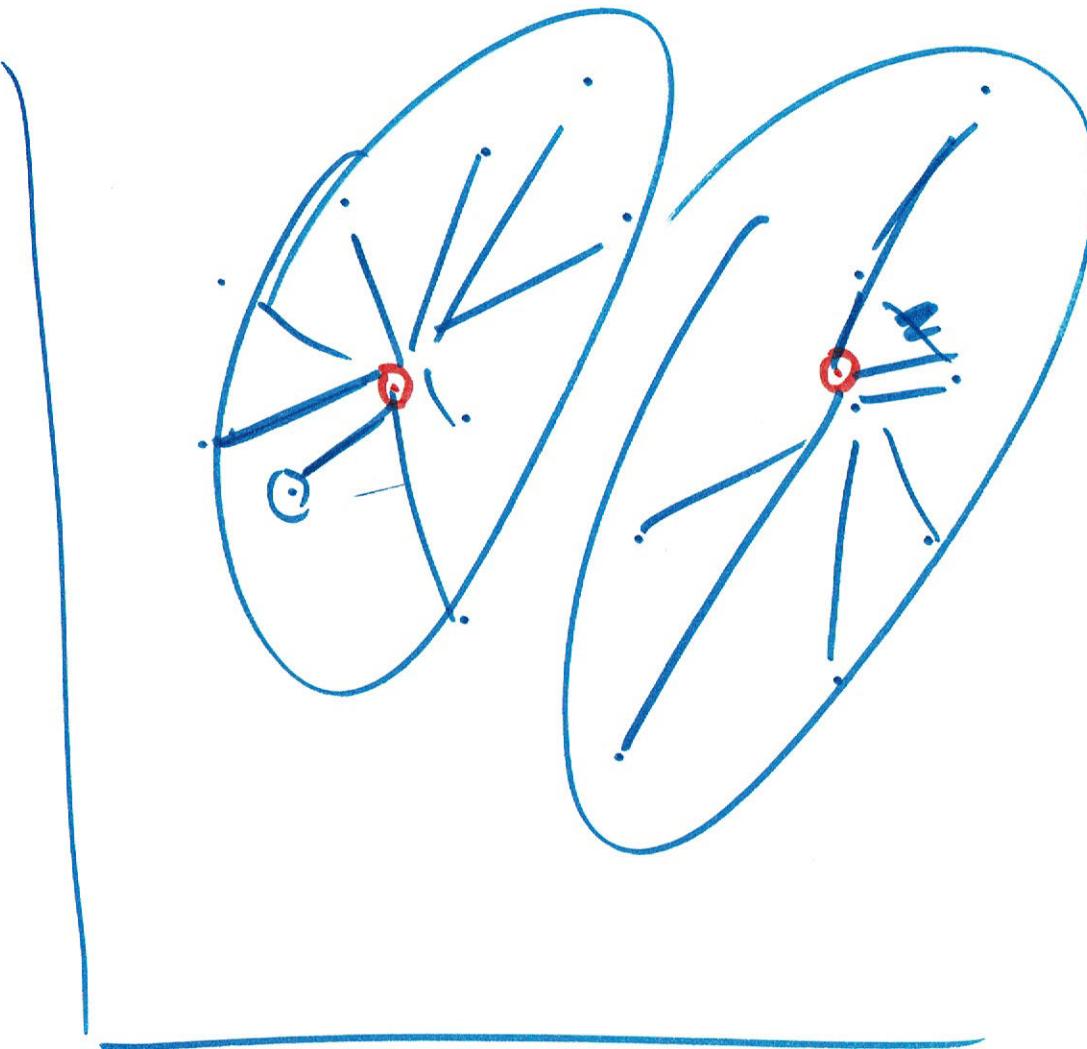
roughly begin by computing

$$5 \times 1000 \text{ dist.}$$

$$5 \times 7$$

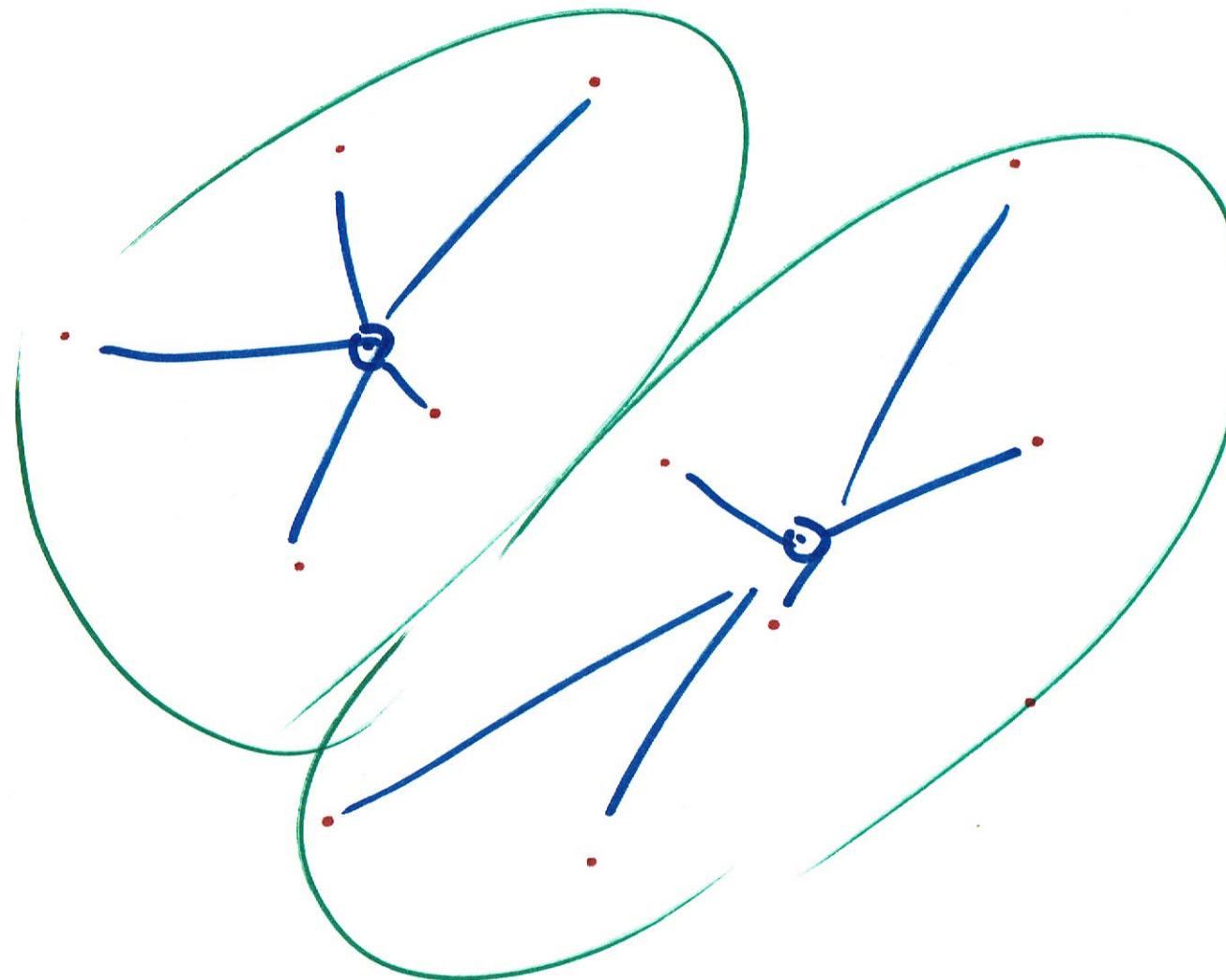






K

$$\frac{n(n+1)}{2}$$

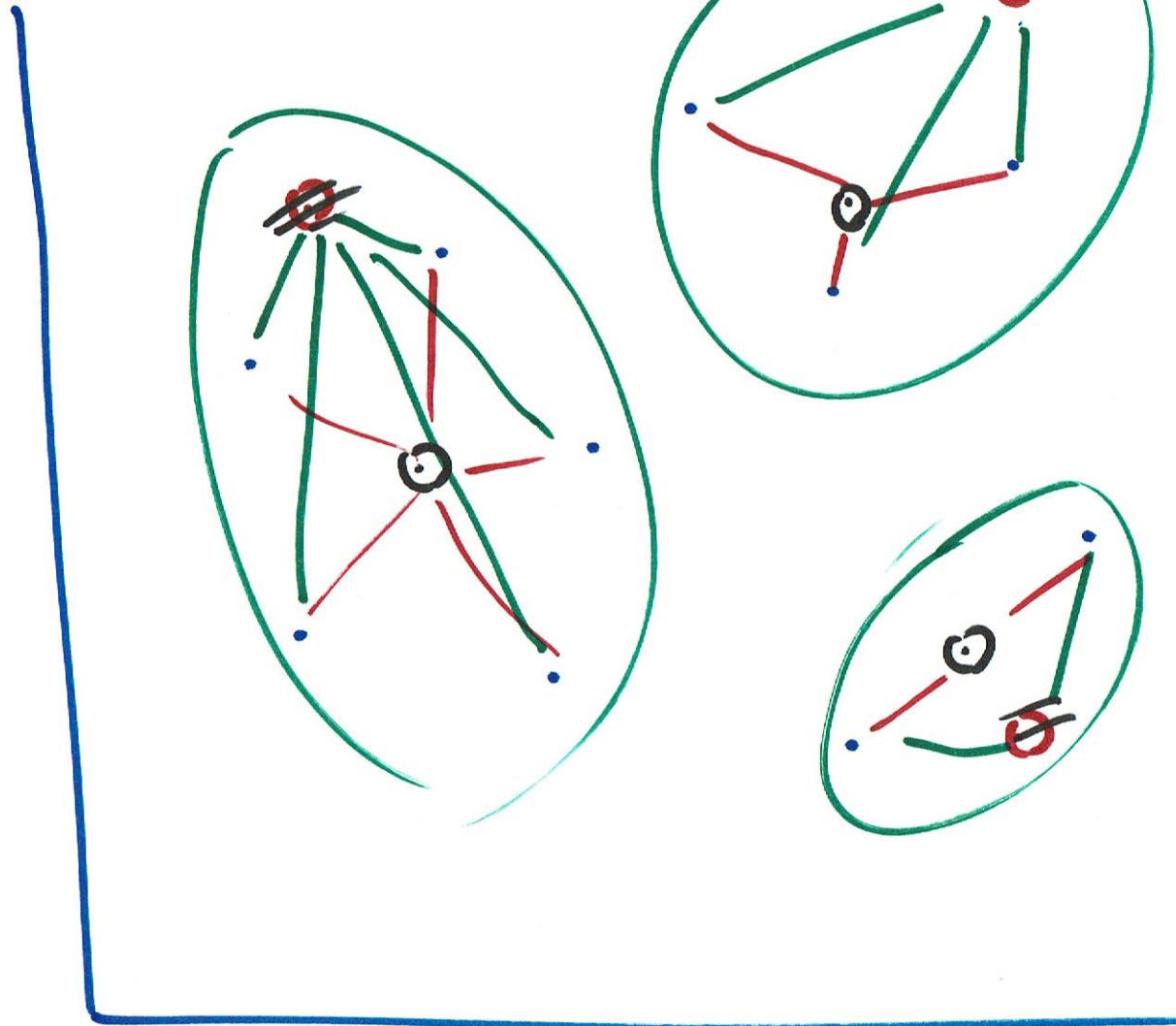


Centroid based: K-Means Clustering

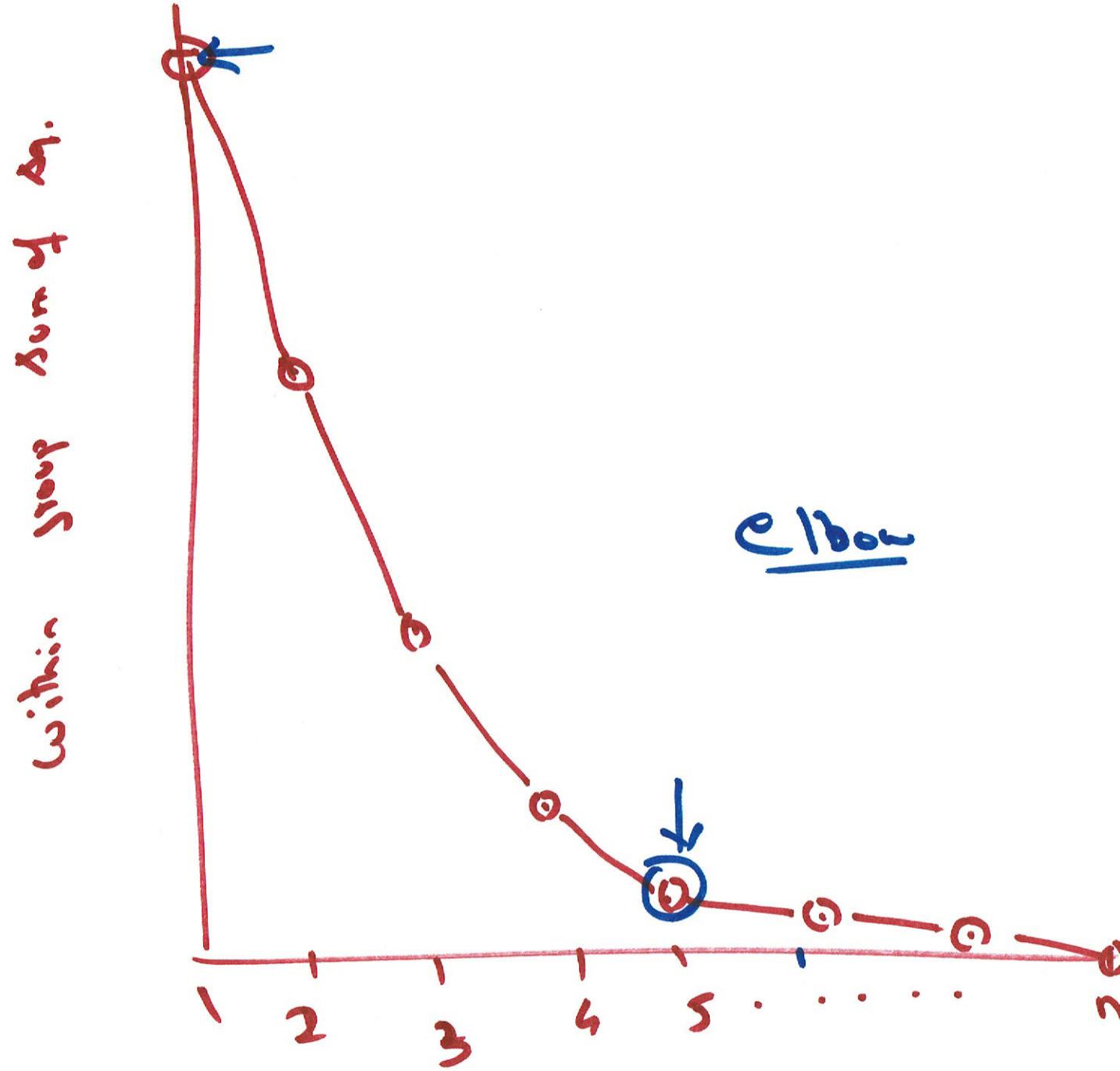
- K-Means is probably the most used clustering technique
- Aims to partition the n observations into k clusters so as to minimize the within-cluster sum of squares (i.e. variance). }
- Computationally less expensive compared to hierarchical techniques.
- Have to pre-define K, the no of clusters

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

$\kappa = 3$



$\kappa \approx$



Data we will work with

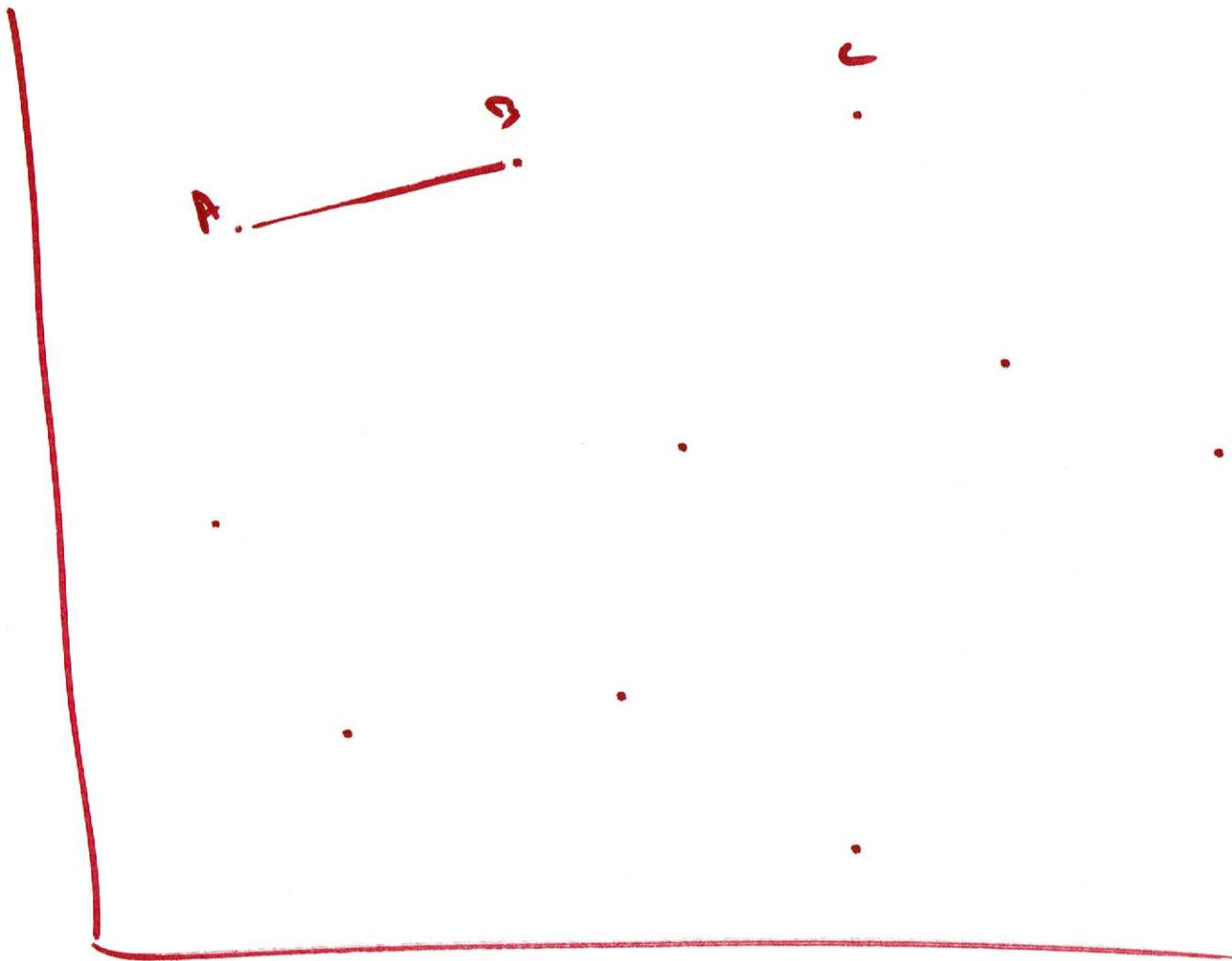
- Customer Spend Data

- AVG_Mthly_Spend: The average monthly amount spent by customer
- No_of_Visits: The number of times a customer visited in a month
- Item Counts: Count of Apparel, Fruits and Vegetable, Staple Items purchased

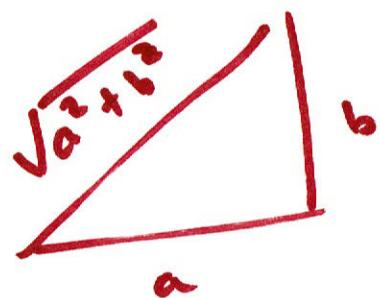


	Cust_ID	Name	Avg_Mthly_Spend	No_Of_Visits	Apparel_Items	FnV_Items	Staples_Items
1	1	A	10000	2	1	1	0
2	2	B	7000	3	0	10	9
3	3	C	7000	7	1	3	4
4	4	D	6500	5	1	1	4
5	5	E	6000	6	0	12	3
6	6	F	4000	3	0	1	8
7	7	G	2500	5	0	11	2
8	8	H	2500	3	0	1	1
9	9	I	2000	2	0	2	2
10	10	J	1000	4	0	1	7

- Can we cluster similar customers together?

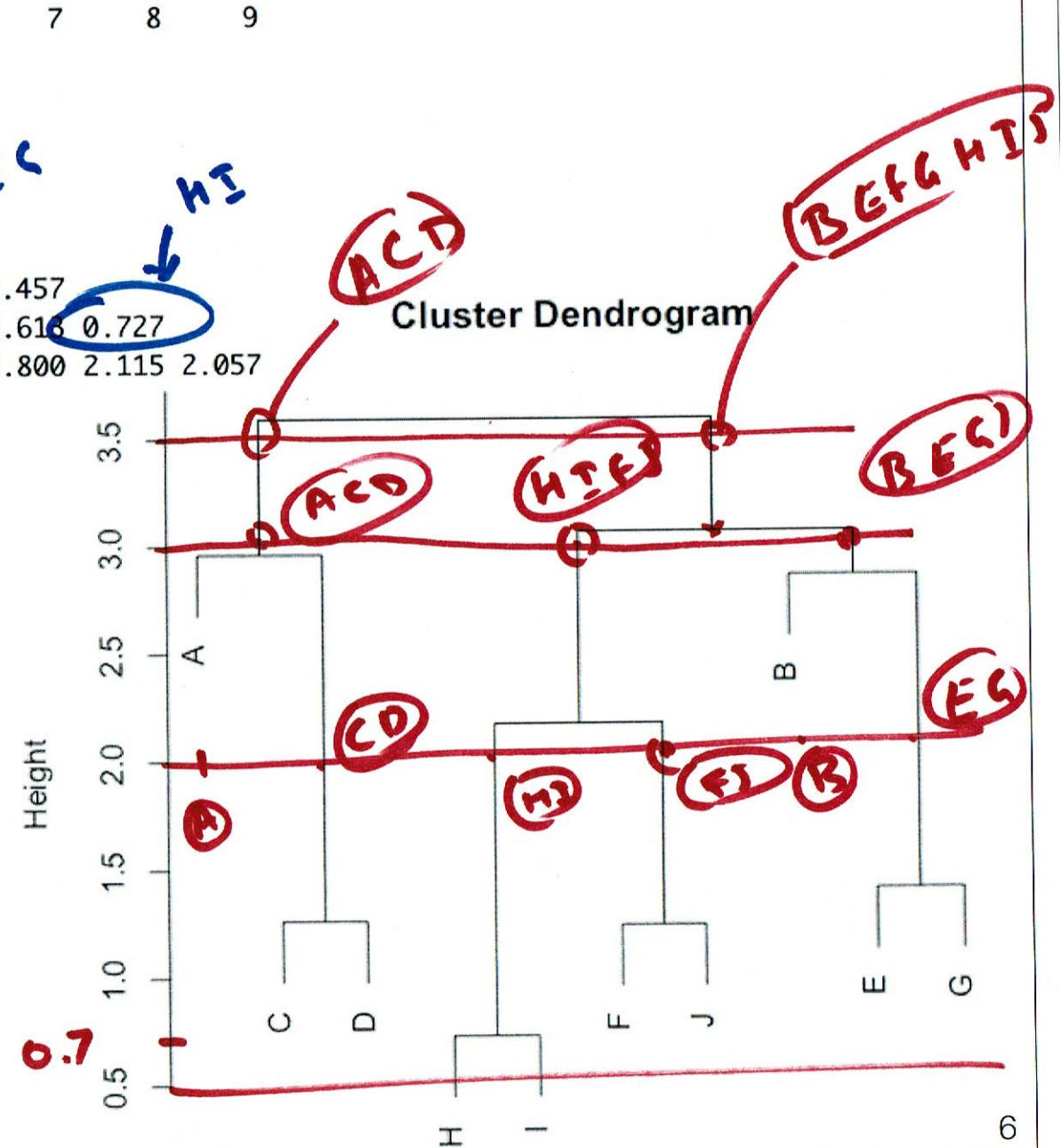


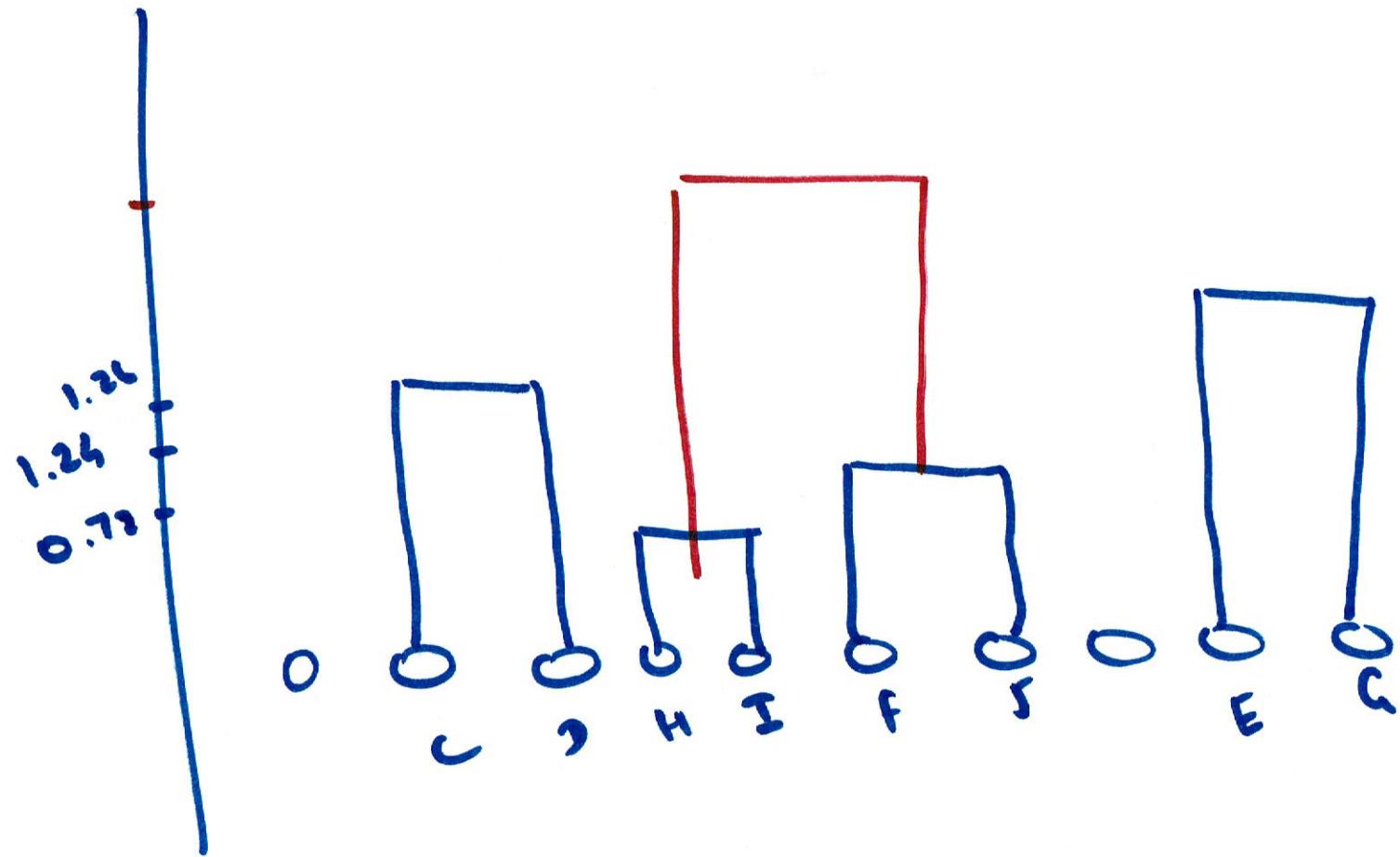
$$\text{dist A + B} = \sqrt{\left((10000 - 7000)^2 + (2-3)^2 + (1-0)^2 \right) + \left((1-10)^2 + (0-9)^2 \right)}$$

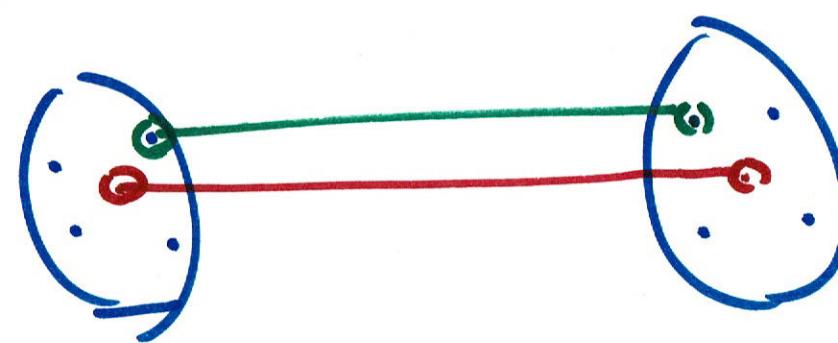
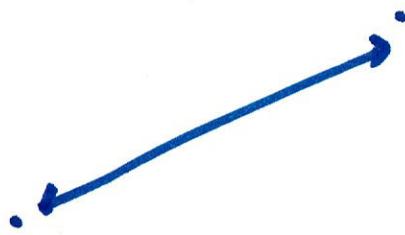


Distance between objects

	1	2	3	4	5	6	7	8	9
2	4.252								
3	3.411	3.838							
4	2.512	3.473	1.264						
5	4.268	2.697	2.922	3.204					
6	3.980	2.208	3.579	2.853	3.431				
7	4.378	3.021	3.384	3.345	1.406	3.171			
8	3.396	3.603	3.663	2.927	3.244	2.350	2.457		
9	3.534	3.395	4.054	3.213	3.482	2.175	2.618	0.727	
10	4.550	2.967	3.591	3.041	3.408	1.241	2.800	2.115	2.057

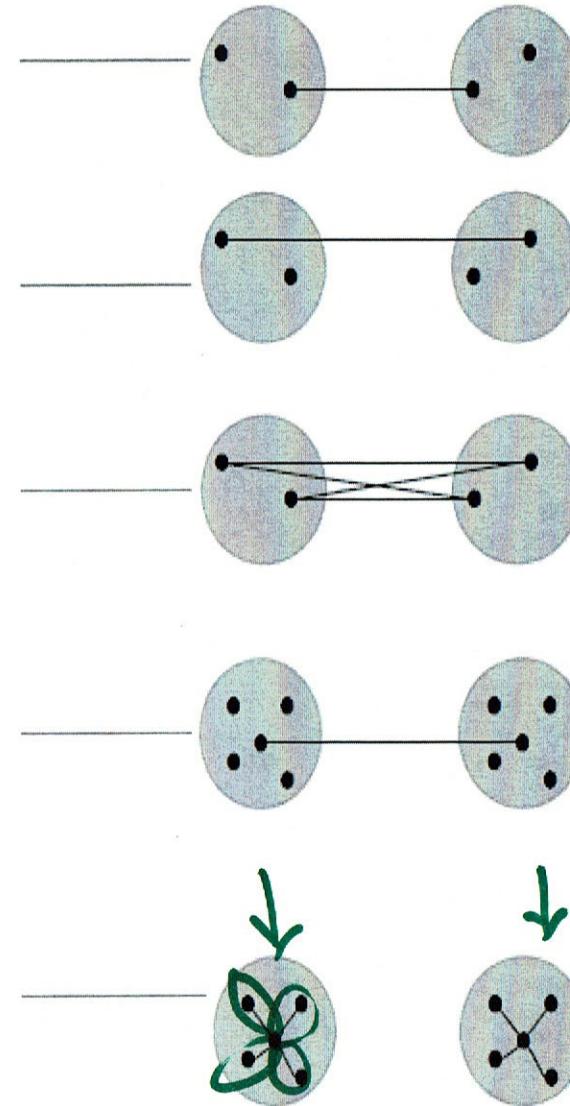


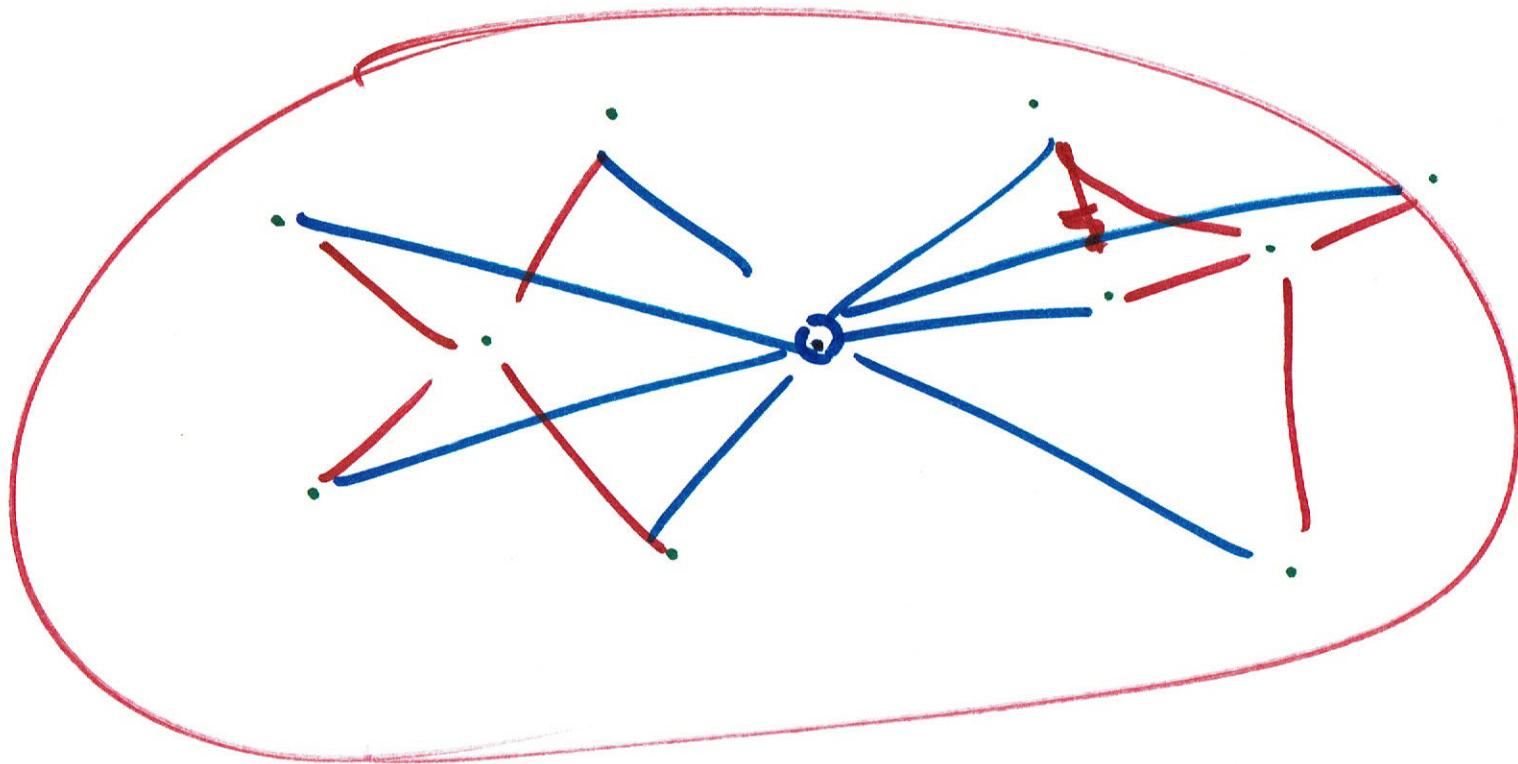




Distance between clusters

- Single linkage – Minimum distance or Nearest neighbor
- Complete linkage – Maximum distance or Farthest distance
- Average linkage – Average of the distances between all pairs
- Centroid method – combine cluster with minimum distance between the centroids of the two clusters
- Ward's method – Combine clusters with which the increase in within cluster variance is to the smallest degree

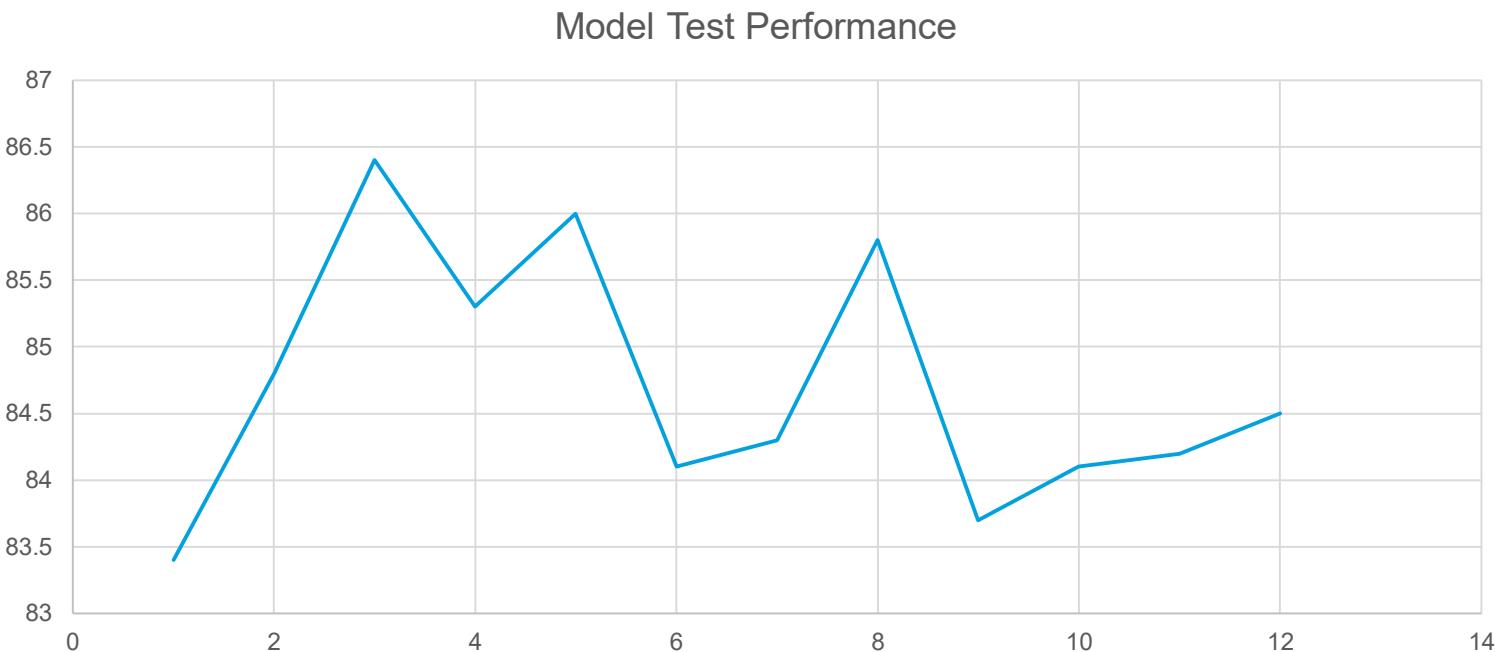




Cross Validation

Need for cross validation?

1. We wish to know how well a ML model is likely to perform in production.
2. Model's performance in training is no guarantee production performance
3. To estimate the model production score, hold a part of the sample data out of training phase. We call it test data which represents the universe
4. Usually the available data is not sufficient to split into training and test set and expect the two to represent the universe
5. Hence the model error on test data may not be good estimate of the model error in the universe
6. In the absence of large data sets, a number of techniques can be employed to estimate the model error in production
7. One of the techniques is cross validation



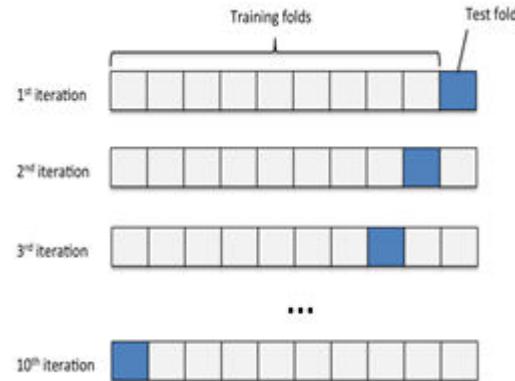
1. Car MPG prediction accuracy trained on 70% and tested on 30%
2. For each run, we get different accuracy scores
3. Simple training/test a.k.a validation approach gives varying results with every run (random state not set)
4. Thus we cannot rely on one round of testing as we would have by chance got the split that gives max score first time
5. To get a more realistic estimate, we have to use more reliable techniques such as cross validation

What is cross validation?

1. Cross-validation is a technique to evaluate / validate a machine learning model and estimate its performance on unseen data
2. The techniques creates and validates given model multiple times
3. The number of times it does so, is dependent on the value selected by the user of the technique. Usually expressed as “K” which is always an integer
4. The sequence of steps used is iterated through as many times as K
5. The process begins by dividing the original data into K parts / folds using random function

Cross validation procedure

1. Shuffle the dataset randomly
2. Split the dataset into k folds
3. For each distinct fold:
 - a. Keep the fold data separate / hold out data set
 - b. Use the remaining folds as a single training data set
 - c. Fit the model on the training set and evaluate it on the test set
 - d. Retain the evaluation score and discard the model
 - e. Loop back
4. The steps 3.a to 3.e will be executed K times
5. Summarize the scores and average it by dividing the sum by K.
6. Analyze the average score, the dispersion to assess the likely performance of the model in the unseen data (production data / universe)



Implementing K Fold cross validation

Visual understanding (example based on scikitlearn guide)

- a) from numpy import array
- b) from sklearn.model_selection import KFold
- c) data = array([10,20,30,40,50,60,70,80,90,100])
- d) kfold = KFold(5, True)
- e) for train, test in kfold.split(data):
- f) print('train: %s, test: %s' % (data[train], data[test]))

Training Data	Test Data
[10 20 30 40 50 60 80 90]	[70 100]
[10 40 50 60 70 80 90 100]	[20 30]
[10 20 30 40 50 70 90 100]	[60 80]
[10 20 30 50 60 70 80 100]	[40 90]
[20 30 40 60 70 80 90 100]	[10 50]

Note : We cannot have K > number of data points.... Why?

Ref: Kfold_introduction.ipynb

Configuring the K

1. K is an integral number. Minimum value of K has to be 2. There will be two iterations in this case
2. Max value of K can be the number of data points. This is also known as Leave One Out Cross Validation or LOOCV
3. Whatever the value of K chosen, the resulting training and test data should be representative of the unseen data as much as possible
4. There is not formula to decide the K but K = 10 is usually considered good
5. Too large a K, means less variance across the training sets thus limit the model differences across iterations
6. For a sample size (N) of n, and K = k, number of records (r) per fold = n/k .

Evaluating the model in an iteration

1. In each iteration, the model is trained on K -1 number of folds and evaluated on the left out fold.
2. The MSE or Mean Squared Error is thus calculated on the left out fold
3. Since the procedure is repeated K times, we will have K MSEs. Total up all the MSE and divide by K to get the overall expected MSE

$$CV_k = \left(\sum(MSE_i) \text{ for } i = 1 \text{ to } K \right) / K$$

Some salient features of K-fold

1. Each record / data point in the sample data before creating the Kfolds, is assigned to a single fold and stays in that fold for the duration of the procedure.
2. This means that each data point is used once in hold-out set and K-1 times in training
3. When hyper parameters are to be tweaked, split the original data into two. Keep one part aside. Use the other to do the Kfold validation. Once the optimal hyperparameters are found, assess the model on the test data
4. Any data transformation done on the whole set outside the loop, may lead to data leakage and overfitting

Implementing K Fold cross validation

K fold in Pima Indian Classification

```
from pandas import read_csv
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import KFold
from sklearn.model_selection import cross_val_score
import numpy as np

filename = 'pima-indians-diabetes.data'
names = ['preg', 'plas', 'pres', 'skin', 'test', 'mass', 'pedi', 'age', 'class']
dataframe = read_csv(filename, names=names)

array = dataframe.values
X = array[:,0:8]
Y = array[:,8]

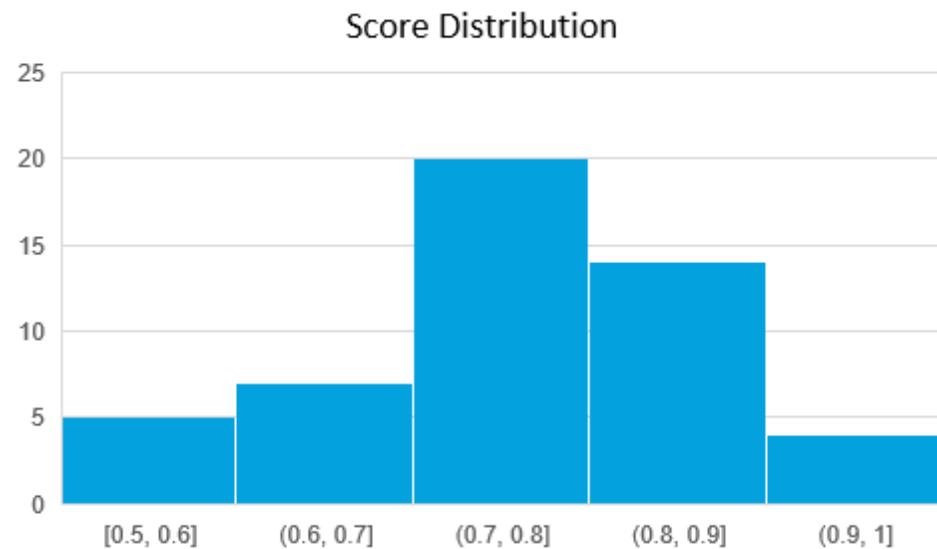
num_folds = 50
seed = 7

kfold = KFold(n_splits=num_folds, random_state=seed)
model = LogisticRegression()
results = cross_val_score(model, X, Y, cv=kfold)
print(results)
print("Accuracy: %.3f%% (%.3f%%)" % (results.mean()*100.0, results.std()*100.0))
```



Kfold_logistic.ipynb

Mean	0.770166667
Standard Error	0.015172553
Median	0.8
Mode	0.8
Standard Deviation	0.107286148
Sample Variance	0.011510317
Kurtosis	0.2246995
Skewness	-0.267638887
Range	0.5
Minimum	0.5
Maximum	1
Sum	38.50833335
Count	50
Confidence Level(95.0%)	0.030490386



1. Distribution of the scores on 50 iterations
2. Model accuracy is likely to be in $0.77 - 0.03$ to $0.77 + 0.03$ i.e. $0.74 - 0.80$ at 95% confidence level

Leave One Out Cross validation (LOOCV) procedure

1. In this method, a single observation (x_1, y_1) is used for the validation set and the remaining $(x_2, y_2), \dots, (x_n, y_n)$ make up the training set
2. The statistical model is fit on the $n-1$ training examples
3. The statistical model prediction \hat{y} is made for the excluded observation using x_1 .
4. $MSE_1 = (\hat{y} - y)^2$ for the excluded point
5. The MSE is unbiased but is poor estimate because it is highly variable
6. We can repeat this by keeping every data point for test one at a time and using rest for training

Leave One Out Cross validation (LOOCV) procedure

7. Repeating this approach n times we get MSE1, MSE2,..., MSEN
8. $CV(n) = \frac{\sum(MSE_i)}{n}$ for $i=1$ to n
9. LOOCV is a special case of Kfold validation with $K = n$

```
# scikit-learn k-fold cross-validation
from numpy import array
from sklearn.model_selection import LeaveOneOut
# data sample
data = array([10,20,30,40,50,60,70,80,90,100])
# prepare cross validation
loocv = LeaveOneOut()
# enumerate splits
for train, test in loocv.split(data):
    print('train: %s, test: %s' % (data[train], data[test]))
```

train: [20 30 40 50 60 70 80 90 100], test: [10]
train: [10 30 40 50 60 70 80 90 100], test: [20]
train: [10 20 40 50 60 70 80 90 100], test: [30]
train: [10 20 30 50 60 70 80 90 100], test: [40]
train: [10 20 30 40 60 70 80 90 100], test: [50]
train: [10 20 30 40 50 70 80 90 100], test: [60]
train: [10 20 30 40 50 60 80 90 100], test: [70]
train: [10 20 30 40 50 60 70 90 100], test: [80]
train: [10 20 30 40 50 60 70 80 100], test: [90]
train: [10 20 30 40 50 60 70 80 90], test: [100]

```
from pandas import read_csv
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import LeaveOneOut
from sklearn.model_selection import cross_val_score
import numpy as np

# prepare cross validation
loocv = LeaveOneOut()
model = LogisticRegression()

filename = 'pima-indians-diabetes.data'
names = ['preg', 'plas', 'pres', 'skin', 'test', 'mass', 'pedi', 'age', 'class']
dataframe = read_csv(filename, names=names)

array = dataframe.values
X = array[:,0:8]
Y = array[:,8]

results = cross_val_score(model, X, Y, cv=loocv)
print("Accuracy: %.3f%% (%.3f%%)" % (results.mean()*100.0, results.std()*100.0))
```

Accuracy: 76.953% (42.113%) (Compare the standard deviation with KFOLD which is much less)

Bias Variance trade-off and cross validation

1. Every machine learning model are impacted by bias, variance and random errors. These errors are analysed in the context of the test environment. Refer to the attached document
2. A simple validation approach consists of a training set and test set where training set is used to build the model and test set is used to validate the model
3. Suppose you create the training and test set in 50:50 ratio using random function multiple times and build and validate the model each time.
4. Since the model is created based on a subset of the data, it is likely to be impacted by bias error and the test errors are likely to vary across iterations
5. The simple validation approach is likely to suffer from both bias and variance errors and the degree of each type of error depends on the ratio of the split

Bias Variance tradeoff and cross validation

6. LOOCV, which gets 90% of the data and multiple iterations, is likely to commit less bias errors. However, since only 10% of the data is left for testing, the variance errors will be high
7. K-Fold cross validation, which lies between simple validation and LOOCV approach, can give more moderate error rates.
8. By selecting right K, we can minimize both the bias and variance errors. The overall error rate is likely to be the model performance in the production

Bootstrap Sampling

1. Also known as sampling with replacement, is a sampling technique used when the amount of data is limited
2. A random function is used to create the sample from the original dataset. That a record has already been picked earlier for the sample is immaterial.
3. Within a sample, there may be repeating records. Could be duplicates or more but two sample sets are unlikely to be 100% same.
4. The records not picked up in an iteration will serve the purpose of test data and test data will always have unique records
5. Suppose we have 10 data points in the dataset. We can create multiple sample sets each with 10 data points or less and corresponding test data.
6. The number of samples created maybe 10, more than 10 or less than 10
7. The more the samples we create from a small size data, more likely we will have samples that are very similar in terms of the data points

Bootstrap Sampling

```
1. from sklearn.utils import resample  
2. import numpy as np  
  
3. # load dataset  
4. data = [10,20,30,40,50,60,70,80,90,100] # original data with 10 data points  
  
5. # configure bootstrap  
6. n_iterations = 50          # Number of bootstrap samples to create = 50  
7. n_size = int(len(data) * 1) # picking only 50 % of the given data in every  
    bootstrap sample  
  
8. # run bootstrap  
9. stats = list()  
10. for i in range(n_iterations):  
11.     # prepare train and test sets  
12.     train = resample(data, n_samples=n_size) # Sampling with replacement  
13.     test = np.array([x for x in data if x not in train]) # picking rest of the data not  
        considered in sample  
14.     print("Train_data ->", train, " " , "Test_data ->", test)
```

Bootstrap sample -

1. Train_data -> [30, 30, 30, 90, 20, 80, 30, 80, 40, 10] Test_data -> [50 60 70 100]
2. Train_data -> [50, 80, 20, 20, 40, 40, 50, 100, 70, 70] Test_data -> [10 30 60 90]
3. Train_data -> [10, 80, 40, 80, 90, 100, 30, 80, 90, 20] Test_data -> [50 60 70]
4. Train_data -> [30, 30, 40, 70, 70, 50, 100, 100, 70, 20] Test_data -> [10 60 80 90]
5. Train_data -> [90, 30, 100, 40, 10, 30, 30, 50, 10, 30] Test_data -> [20 60 70 80]
6. Train_data -> [30, 40, 30, 20, 80, 20, 10, 30, 50, 40] Test_data -> [60 70 90 100]
7. Train_data -> [70, 30, 60, 30, 80, 100, 40, 20, 70, 70] Test_data -> [10 50 90]
8. Train_data -> [100, 30, 70, 100, 90, 90, 10, 60, 60, 70] Test_data -> [20 40 50 80]
9. Train_data -> [40, 40, 30, 90, 90, 30, 90, 10, 60, 100] Test_data -> [20 50 70 80]
10. Train_data -> [10, 90, 70, 70, 30, 20, 70, 70, 90, 40] Test_data -> [50 60 80 100]
11. Train_data -> [100, 40, 100, 90, 100, 50, 30, 50, 20, 40] Test_data -> [10 60 70 80]
12. Train_data -> [50, 20, 10, 50, 50, 20, 60, 20, 40, 100] Test_data -> [30 70 80 90]
13. Train_data -> [10, 20, 10, 40, 60, 30, 20, 30, 80, 80] Test_data -> [50 70 90 100]
14. Train_data -> [50, 90, 50, 60, 50, 90, 40, 30, 40, 50] Test_data -> [10 20 70 80 100]
15. Train_data -> [100, 70, 100, 70, 100, 70, 90, 20, 20, 60] Test_data -> [10 30 40 50 80]
16. Train_data -> [80, 30, 100, 60, 40, 20, 100, 70, 20, 60] Test_data -> [10 50 90]
17. Train_data -> [20, 100, 50, 70, 50, 10, 50, 60, 30, 70] Test_data -> [40 80 90]
18. Train_data -> [10, 60, 90, 40, 50, 100, 50, 50, 20, 80] Test_data -> [30 70]
19. Train_data -> [20, 30, 80, 10, 100, 80, 60, 90, 50, 80] Test_data -> [40 70]
20. Train_data -> [10, 20, 70, 10, 80, 60, 50, 20, 20, 10] Test_data -> [30 40 90 100]
21. Train_data -> [90, 70, 50, 100, 20, 60, 60, 90, 60, 70] Test_data -> [10 30 40 80]
22. Train_data -> [80, 20, 100, 10, 80, 30, 100, 20, 60, 100] Test_data -> [40 50 70 90]
23. Train_data -> [10, 90, 20, 90, 50, 80, 30, 100, 10, 80] Test_data -> [40 60 70]
24. Train_data -> [100, 30, 30, 70, 90, 30, 30, 90, 100, 10] Test_data -> [20 40 50 60 80]
25. Train_data -> [20, 10, 40, 20, 20, 40, 20, 90, 100, 50] Test_data -> [30 60 70 80]
26. Train_data -> [20, 100, 50, 60, 80, 70, 90, 20, 90, 40] Test_data -> [10 30]
27. Train_data -> [70, 20, 100, 20, 40, 60, 30, 80, 80, 70] Test_data -> [10 50 90]

Bootstrap sample (contd...) -

28. Train_data -> [70, 50, 80, 60, 100, 60, 40, 80, 70, 100] Test_data -> [10 20 30 90]
29. Train_data -> [40, 90, 80, 20, 10, 70, 10, 80, 90, 60] Test_data -> [30 50 100]
30. Train_data -> [50, 80, 90, 90, 80, 80, 80, 20, 30, 90] Test_data -> [10 40 60 70 100]
31. Train_data -> [90, 40, 40, 80, 20, 80, 90, 30, 50, 90] Test_data -> [10 60 70 100]
32. Train_data -> [40, 50, 100, 100, 70, 60, 40, 100, 50, 10] Test_data -> [20 30 80 90]
33. Train_data -> [30, 20, 30, 70, 20, 20, 30, 80, 40, 70] Test_data -> [10 50 60 90 100]
34. Train_data -> [10, 10, 70, 50, 60, 40, 70, 70, 100, 10] Test_data -> [20 30 80 90]
35. Train_data -> [10, 40, 50, 100, 30, 100, 20, 10, 80, 70] Test_data -> [60 90]
36. Train_data -> [90, 30, 10, 70, 50, 40, 30, 100, 20, 40] Test_data -> [60 80]
37. Train_data -> [40, 50, 90, 100, 30, 100, 90, 30, 50, 70] Test_data -> [10 20 60 80]
38. Train_data -> [30, 10, 20, 70, 60, 90, 90, 30, 70, 60] Test_data -> [40 50 80 100]
39. Train_data -> [100, 60, 90, 20, 100, 70, 20, 50, 70, 100] Test_data -> [10 30 40 80]
40. Train_data -> [50, 90, 30, 60, 40, 80, 20, 10, 40, 90] Test_data -> [70 100]
41. Train_data -> [10, 30, 20, 10, 80, 60, 20, 40, 20, 70] Test_data -> [50 90 100]
42. Train_data -> [90, 100, 60, 80, 10, 90, 20, 30, 30, 30] Test_data -> [40 50 70]
43. Train_data -> [60, 70, 70, 100, 20, 30, 20, 50, 60, 70] Test_data -> [10 40 80 90]
44. Train_data -> [20, 10, 50, 30, 90, 50, 100, 80, 40, 100] Test_data -> [60 70]
45. Train_data -> [10, 30, 10, 40, 80, 60, 20, 40, 60, 30] Test_data -> [50 70 90 100]
46. Train_data -> [60, 100, 70, 70, 70, 20, 30, 90, 90, 20] Test_data -> [10 40 50 80]
47. Train_data -> [40, 80, 80, 20, 80, 90, 50, 30, 30, 80] Test_data -> [10 60 70 100]
48. Train_data -> [10, 50, 60, 70, 100, 60, 30, 80, 100, 70] Test_data -> [20 40 90]
49. Train_data -> [80, 20, 40, 100, 10, 90, 50, 40, 90, 20] Test_data -> [30 60 70]
50. Train_data -> [10, 70, 60, 50, 50, 100, 40, 50, 80, 50] Test_data -> [20 30 90]

Bootstrap Sampling

1. With the bootstrap samples available, we can create models on the training and test and average out the scores over all the runs. Refer to the attachment
2. When we create and test our model on bootstrapped data set, each iteration will give a performance score
3. When we increase the number of iterations to a large number and plot the frequency curve for the performance scores, one will notice the performance scores tend to follow normal distribution.
4. For very large number of iterations, the distribution becomes almost normal. This is known as the Central Limit Theorem.
5. Central Limit Theorem states “the sampling distribution of the sample means approaches a normal distribution as the sample size gets larger — *no matter what the shape of the population distribution.*

```
1. from pandas import read_csv
2. from sklearn.utils import resample
3. from sklearn.tree import DecisionTreeClassifier
4. from sklearn.metrics import accuracy_score
5. from matplotlib import pyplot
6. import numpy as np

7. # load dataset
8. data = read_csv('pima-indians-diabetes.data', header=None)
9. values = data.values

10. # configure bootstrap
11. n_iterations = 1000          # Number of bootstrap samples to create
12. n_size = int(len(data) * 0.50) # picking only 50 % of the given data in every bootstrap sample

13. # run bootstrap
14. stats = list()
15. for i in range(n_iterations):
16.     # prepare train and test sets
17.     train = resample(values, n_samples=n_size) # Sampling with replacement
18.     test = np.array([x for x in values if x.tolist() not in train.tolist()]) # picking rest of the data not considered in
sample
19.     # fit model
20.     model = DecisionTreeClassifier()
21.     model.fit(train[:, :-1], train[:, -1])
22.     # evaluate model
23.     predictions = model.predict(test[:, :-1])
24.     score = accuracy_score(test[:, -1], predictions) # caution, overall accuracy score can mislead when classes
are imbalanced
25.     print(score)
26.     stats.append(score)
```



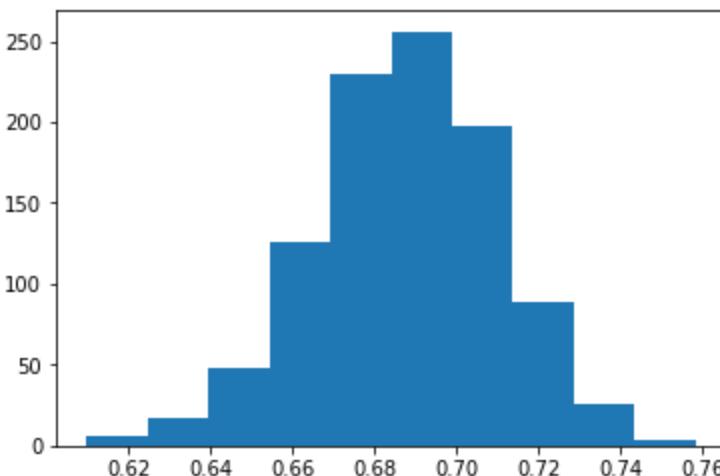
Bootstrapping_Co
nfidence_Level.ipynb

Model performance measures

1. We never give point estimates for model's performance in production. We always give range estimates. For e.g. model accuracy is likely to be in the range of 80% - 95%
2. Range estimates indicate lack of surety. Which means less than 100% sure. Hence, range estimates need to be backed up by confidence level. How confident are we in the range.
3. Larger the range more confident we are. But range cannot be too large. The model becomes unreliable!
4. The general practice is to quote the range at 95% confidence level. If you are working in life critical projects, one needs to be 99.99999% confident
5. The range can be estimated through K-Fold validation or Bootstrap sampling

Model Accuracy estimates using BootStrapping

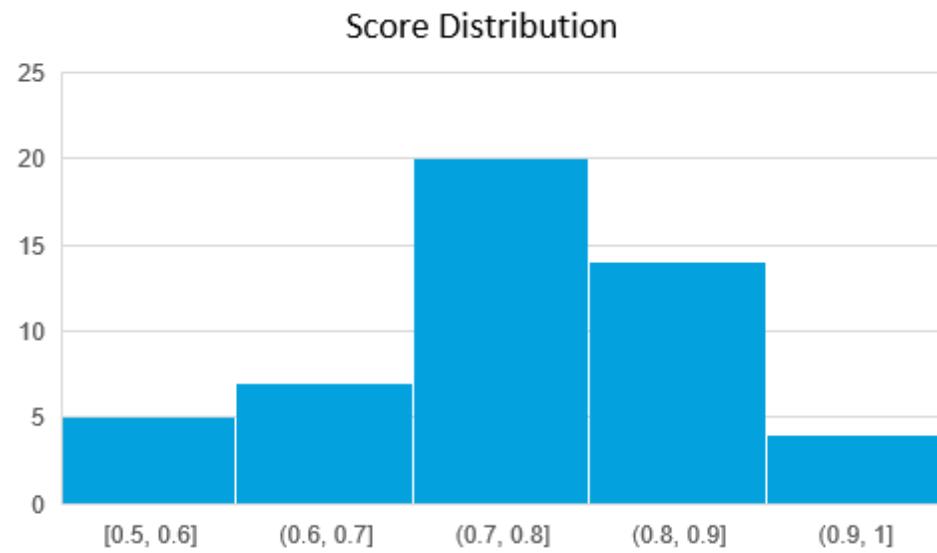
```
1. # plot scores
2. pyplot.hist(stats)
3. pyplot.show()
4. # confidence intervals
5. alpha = 0.95          # for 95% confidence
6. p = ((1.0-alpha)/2.0) * 100      # tail regions on right and left .25 on each side indicated by P value (border)
7. lower = max(0.0, np.percentile(stats, p))
8. p = (alpha+((1.0-alpha)/2.0)) * 100
9. upper = min(1.0, np.percentile(stats, p))
10. print('%.1f confidence interval %.1f%% and %.1f%%' % (alpha*100, lower*100, upper*100))
```



95.0 confidence interval 64.2% and 73.0%

Model Accuracy estimates using Kfold Validation

Mean	0.770166667
Standard Error	0.015172553
Median	0.8
Mode	0.8
Standard Deviation	0.107286148
Sample Variance	0.011510317
Kurtosis	0.2246995
Skewness	-0.267638887
Range	0.5
Minimum	0.5
Maximum	1
Sum	38.50833335
Count	50
Confidence Level(95.0%)	0.030490386



1. Model accuracy is likely to be in the range of 0.74 – 0.80 at 95% confidence level

Model performance measures

Model performance measures

- a. Confusion Matrix – A 2X2 tabular structure reflecting the performance of the model in four blocks

Confusion Matrix	Predicted Positive	Predicted Negative
Actual Positive	True Positive	False Negative
Actual Negative	False Positive	True Negative

- a. Accuracy – How accurately / cleanly does the model classify the data points. Lesser the false predictions, more the accuracy

$$\text{Accuracy} = (TP + TN) / (TP + TN + FP + FN)$$

- a. Sensitivity / Recall – How many of the actual True data points are identified as True data points by the model . Remember, False Negatives are those data points which should have been identified as True.

$$\text{Recall} = TP / (TP + FN)$$

- a. Specificity – How many of the actual Negative data points are identified as negative by the model

$$\text{SPEC} = \frac{TN}{TN + FP}$$

- a. Precision – Among the points identified as Positive by the model, how many are really Positive

$$\text{Precision} = TP / (TP + FP)$$

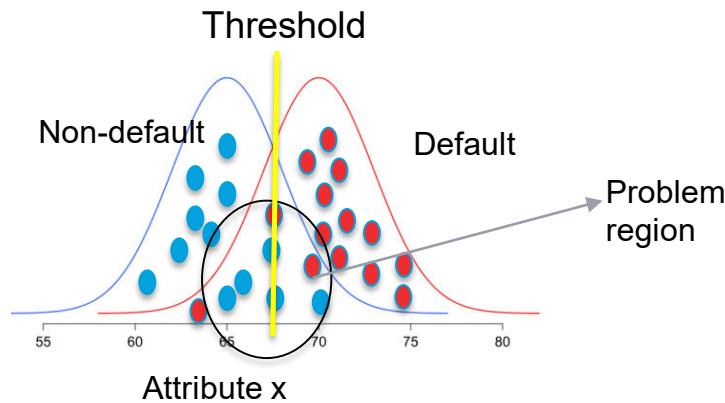
Model performance measures

Assume model is identifying defaulters. In this binary classification defaulter class is class of interest and labeled as +ive (positive - 1) class, other class is – ve(negative - 0)

1. True Positives - cases where the actual class of the data point and the predicted is same. For e.g. a defaulter (1) predicted as defaulter (1)
2. True Negatives – cases where the actual class was non-defaulter and the prediction also was non-defaulter
3. False Positives – cases where actual class was negative (0) but predicted as defaulter (1)
4. False Negatives – cases where the actual class was positive (1) but predicted as non-defaulter (0)
5. Ideal scenario will be when all positives are predicted as positives and all negatives are predicted as negatives

Model performance measures

6. In practical world this will never be the case. There will be some false positives and false negatives
7. Our objective will be to minimize both but the problem is, when we minimize one the other will increase and vice versa!
8. The problem is in the overlap region in the distributions



6. Objective will be to minimize one of the error types, either the false positive or false negative

Model performance measures

10. Minimize false negatives - if predicting a positive case as negative is going to be more detrimental for e.g. predicting a cancer patient (positive) as non-cancer (negative)
11. Minimize false positives – if predicting a negative as positive is going to be more detrimental for e.g. predicting a boss's mail as spam!
12. Accuracy – over all correct predictions from all the classes to total number of cases. Should rely on this metrics only when all classes are equally represented. Not reliable if class representation is lopsided as algorithms are biased towards over represented class
13. Precision - $TP / (TP + FP)$. When we focus on minimizing false negatives, TP will increase but along with it FP will also increase. How much increase in TP starts hurting (due to increase in FP) ?

Model performance measures

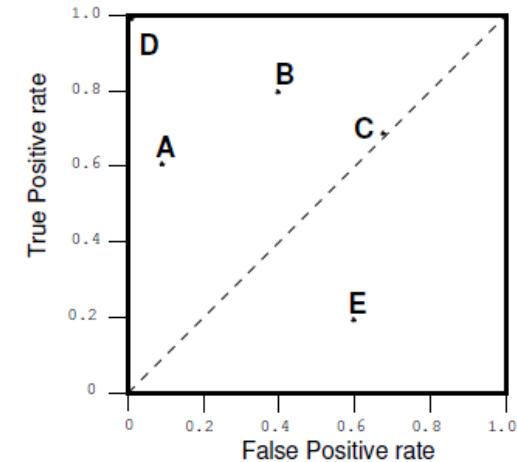
14. Recall – $TP / (TP + FN)$: when we reduce FN to increase TP, how much we gain ? Recall and precision will oppose each other. We want recall to be as close to 1 as possible without precision being too bad

14. To compare models, we use ROC AUC that gives us the optimal combination of these metrics

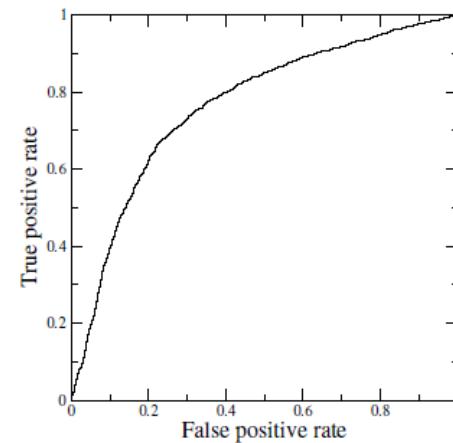
Receiver Operating Characteristics (ROC) Curve

A technique for visualizing classifier performance

- a. It is a graph between TP rate and FP rates
 - I. $\text{TP rate} = \text{TP} / \text{total positive}$
 - II. $\text{FP rate} = \text{FP} / \text{total negative}$
- b. ROC graph is a trade off between benefits (TP) and costs (FP)
- c. The point (0,1) represents perfect classified (e.g. D)
 - I. $\text{TP} = 1$ and $\text{FP} = 0$
- d. Classifiers very close to Y axis and lower (nearer to x axis) are conservative models and strict in classifying positives (low TP rate)
 - a. Classifiers on top right are liberal in classifying positives hence higher TP rate and FP rate



A basic ROC graph showing five discrete classifiers.



Ref:ROC_AUC.ipynb ,

Linear Regression Regularization

Linear Regression Model -

Lab- 1- Estimating mileage based on features of a second hand car

Description – Sample data is available at

<https://archive.ics.uci.edu/ml/datasets/Auto+MPG>

The dataset has 9 attributes listed below that define the quality

1. mpg: continuous
2. cylinders: multi-valued discrete
3. displacement: continuous
4. horsepower: continuous
5. weight: continuous
6. acceleration: continuous
7. model year: multi-valued discrete
8. origin: multi-valued discrete
9. car name: string (unique for each instance)

Sol : Ridge_Lasso_Regression.ipynb

Regularising Linear Models (Shrinkage methods)

When we have too many parameters and exposed to curse of dimensionality, we resort to dimensionality reduction techniques such as transforming to PCA and eliminating the PCA with least magnitude of eigen values. This can be a laborious process before we find the right number principal components. Instead, we can employ the shrinkage methods.

Shrinkage methods attempt to shrink the coefficients of the attributes and lead us towards simpler yet effective models. The two shrinkage methods are :

1. Ridge regression is similar to the linear regression where the objective is to find the best fit surface. The difference is in the way the best coefficients are found. Unlike linear regression where the optimization function is SSE, here it is slightly different

$$\sum_{i=1}^n \left(y_i - \beta_0 - \sum_{j=1}^p \beta_j x_{ij} \right)^2$$

Linear Regression cost function

$$\sum_{i=1}^n \left(y_i - \beta_0 - \sum_{j=1}^p \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^p \beta_j^2$$

Ridge Regression with additional term in the cost function

1. The term λ is like a penalty term used to penalize large magnitude coefficients β_j when it is set to a high number, coefficients are suppressed significantly. When it is set to 0, the cost function becomes same as linear regression cost function

Regularising Linear Models (Shrinkage methods)

Why should we be interested in shrinking the coefficients? How does it help?

When we have large number of dimensions and few data points, the models are likely to become complex, overfit and prone to variance errors. When you print out the coefficients of the attributes of such complex model, you will notice that the magnitude of the different coefficients become large

Large coefficients indicate a case where for a unit change in the input variable, the magnitude of change in the target column is very large.

Coeff for simple linear regression model of 10 dimensions

1. The coefficient for cyl is 2.5059518049385052
2. The coefficient for disp is 2.5357082860560483
3. The coefficient for hp is -1.7889335736325294
4. The coefficient for wt is -5.551819873098725
5. The coefficient for acc is 0.11485734803440854
6. The coefficient for yr is 2.931846548211609
7. The coefficient for car_type is 2.977869737601944
8. The coefficient for origin_amERICA is -0.5832955290166003
9. The coefficient for origin_asIA is 0.3474931380432235
10. The coefficient for origin_eUROPE is 0.3774164680868855

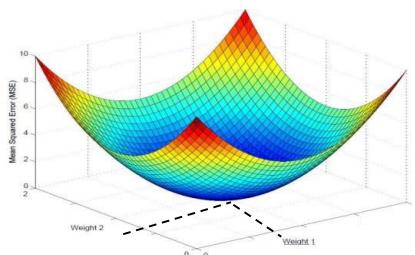
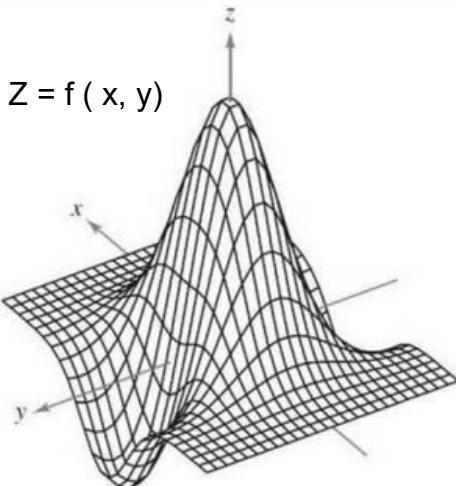
Ref: Ridge_Lasso_Regression.ipynb

Coeff with polynomial features shooting up to 57 from 10

```
-9.67853872e-13 -1.06672046e+12 -4.45865268e+00 -2.24519565e+00 -  
2.96922206e+00 -1.56882955e+00 3.00019063e+00 -1.42031640e+12 -  
5.46189566e+11 3.62350196e+12 -2.88818173e+12 -1.16772461e+00 -  
1.43814087e+00 -7.49492645e-03 2.59439087e+00 -1.92409515e+00 -  
3.41759793e+12 -6.27534905e+12 -2.44065576e+12 -2.32961194e+12  
3.97766113e-01 1.94046021e-01 -4.26086426e-01 3.58203125e+00 -  
2.05296326e+00 -7.51019934e+11 -6.18967069e+11 -5.90805593e+11  
2.47863770e-01 -6.68518066e-01 -1.92150879e+00 -7.37030029e-01 -  
1.01183732e+11 -8.33924574e+10 -7.95983063e+10 -1.70394897e-01  
5.25512695e-01 -3.33097839e+00 1.56301740e+12 1.28818991e+12  
1.22958044e+12 5.80200195e-01 1.55352783e+00 3.64527008e+11  
3.00431724e+11 2.86762821e+11 3.97644043e-01 8.58604718e+10  
7.07635073e+10 6.75439422e+10 -7.25449332e+11 1.00689540e+12  
9.61084146e+11 2.18532428e+11 -4.81675252e+12 2.63818648e+12
```

Very large coefficients!

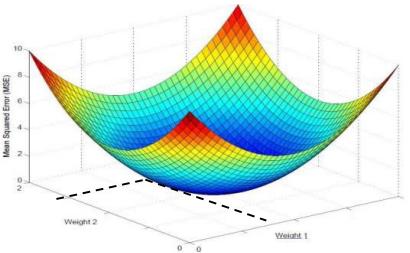
Regularising Linear Models (Shrinkage methods)



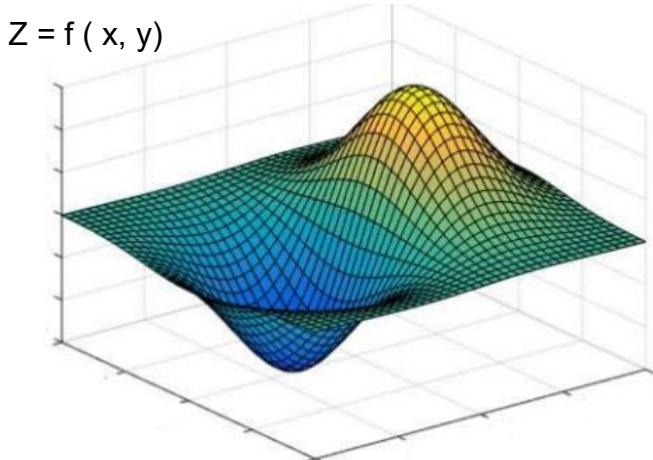
1. Curse of dimensionality results in large magnitude coefficients which results in a complex undulated surface / model.
1. This complex surface has the data points occupying the peaks and the valleys
1. The model gives near 100% accuracy in training but poor result in testing and the testing scores also vary a lot from one sample to another.
1. The model is supposed to have absorbed the noise in the data distribution!
1. Large magnitudes of the coefficient give the least SSE and at times SSE = 0! A model that fits the training set 100%!
1. Such models do not generalize

$$\sum_{i=1}^n \left(y_i - \beta_0 - \sum_{j=1}^p \beta_j x_{ij} \right)^2 = 0$$

Regularising Linear Models (Shrinkage methods)



$$\sum_{i=1}^n \left(y_i - \beta_0 - \sum_{j=1}^p \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^p \beta_j^2$$



1. In Ridge Regression, the algorithm while trying to find the best combination of coefficients which minimize the SSE on the training data, is constrained by the penalty term
 1. The penalty term is akin to cost of magnitude of the coefficients. Higher the magnitude, more the cost. Thus to minimize the cost, the coefficient are suppressed
 1. Thus the resulting surface tends to be relatively much more smoother than the unconstrained surface. This means we have settled for a model which will make errors in the training data
 1. This is fine as long as the errors can be attributed to the random fluctuations i.e. because the model does not absorb the random fluctuations in the data
 1. Such model will perform equally well on unseen data i.e. test data. The model will generalize better than the complex model

Regularising Linear Models (Shrinkage methods)

Impact of Ridge Regression on the coefficients of the 56 attributes

```
Ridge model: [[ 0.  3.73512981 -2.93500874 -2.13974194 -3.56547812 -1.28898893  3.01290805  
 2.04739082  0.0786974  0.21972225 -0.3302341 -1.46231096 -1.17221896  0.00856067  2.48054694  
 -1.67596093  0.99537516 -2.29024279  4.7699338 -2.08598898  0.34009408  0.35024058 -0.41761834  
 3.06970569 -2.21649433  1.86339518 -2.62934278  0.38596397  0.12088534 -0.53440382 -1.88265835  
 -0.7675926 -0.90146842  0.52416091  0.59678246 -0.26349448  0.5827378 -3.02842915 -0.36548074  
 0.5956112 -0.15941014  0.49168856  1.45652375 -0.43819158 -0.20964198  0.77665496  0.36489921  
 -0.4750838  0.3551047  0.23188557 -1.42941282  2.06831543 -0.34986402 -0.32320394  0.39054656  0.06283411]]
```

Large coefficients have been suppressed, almost close to 0 in many cases.

Regularising Linear Models (Shrinkage methods)

1. Lasso Regression is similar to the Ridge regression with a difference in the penalty term. Unlike Ridge, the penalty term here is raised to power 1. Also known as L1 norm.

$$\sum_{i=1}^M \left(y_i - \sum_{j=0}^p w_j \times x_{ij} \right)^2 + \lambda \sum_{j=0}^p |w_j|$$

1. The term λ continues to be the input parameter which will decide how high penalties would be for the coefficients. Larger the value more diminished the coefficients will be.
1. Unlike Ridge regression, where the coefficients are driven towards zero but may not become zero, Lasso Regression penalty process will make many of the coefficients 0. In other words, literally drop the dimensions

Regularising Linear Models (Shrinkage methods)

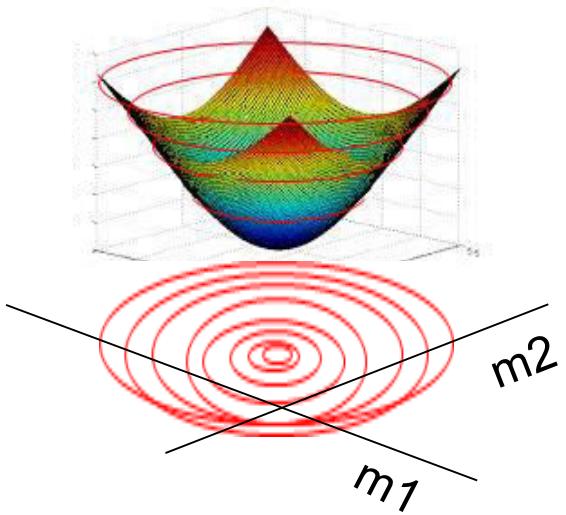
Impact of Lasso Regression on the coefficients of the 56 attributes

```
Lasso model: [ 0. 0.52263805 -0.5402102 -1.99423315 -4.55360385 -0.85285179 2.99044036 0.00711821 -0. 0.76073274 -0. -0. -0.19736449  
0. 2.04221833 -1.00014513 0. -0. 4.28412669 -0. 0. 0.31442062 -0. 2.13894094 -1.06760107 0. -0. 0. 0. -0.44991392 -1.55885506 -0. -0.68837902 0.  
0.17455864 -0.34653644 0.3313704 -2.84931966 0. -0.34340563 0.00815105 0.47019445 1.25759712 -0.69634581 0. 0.55528147 0.2948979 -0.67289549  
0.06490671 0. -1.19639935 1.06711702 0. -0.88034391 0. -0. ]
```

Large coefficients have been suppressed, to 0 in many cases, making those dimensions useless i.e. dropped from the model.

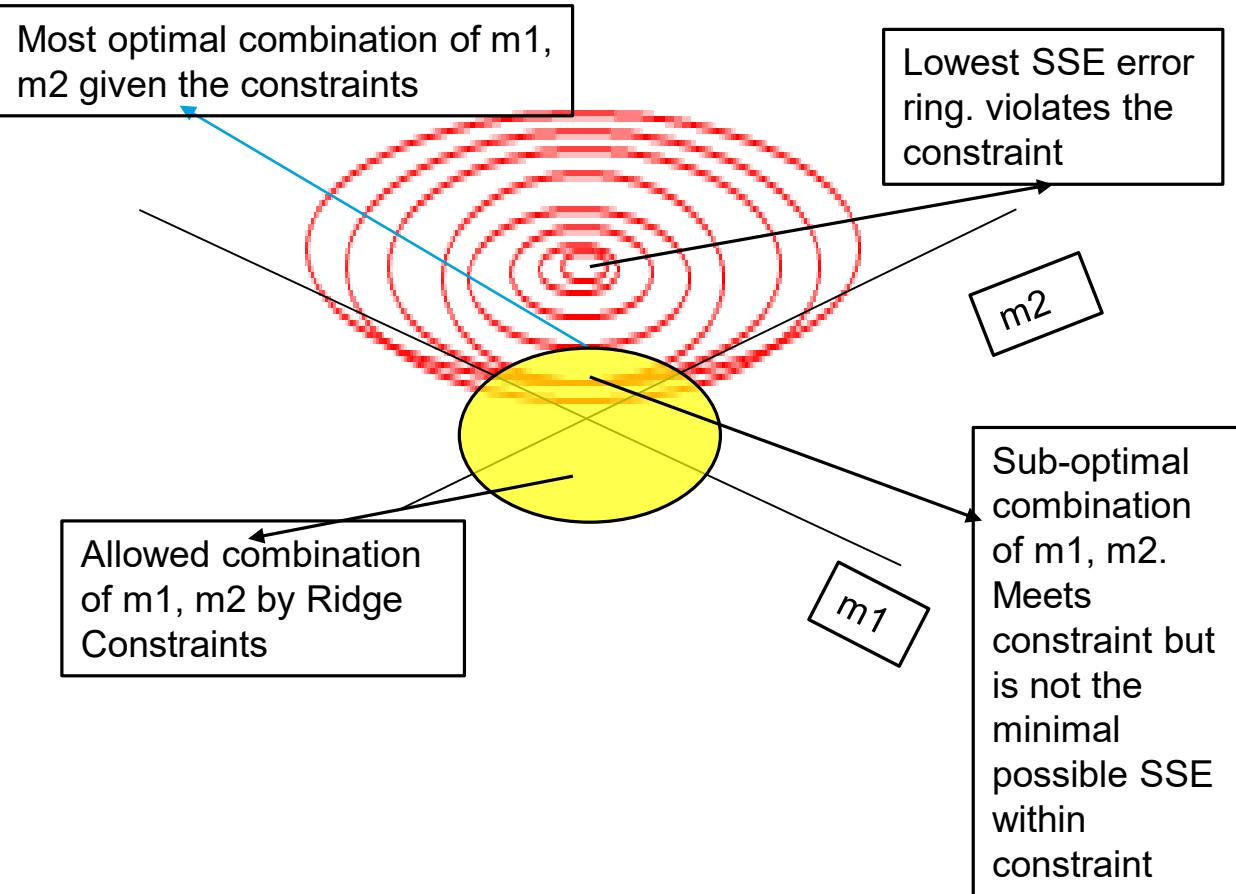
Regularising Linear Models (Comparing The Methods)

To compare the Ridge and Lasso, let us first transform our error function (which is a quadratic / convex function) into a contour graph



1. Every ring on the error function represents a combination of coefficients (m_1 and m_2 in the image) which result in same quantum of error i.e. SSE
1. Let us convert that to a 2d contour plot. In the contour plot, every ring represents one quantum of error.
1. The innermost ring / bull's eye is the combination of the coefficients that gives the least SSE

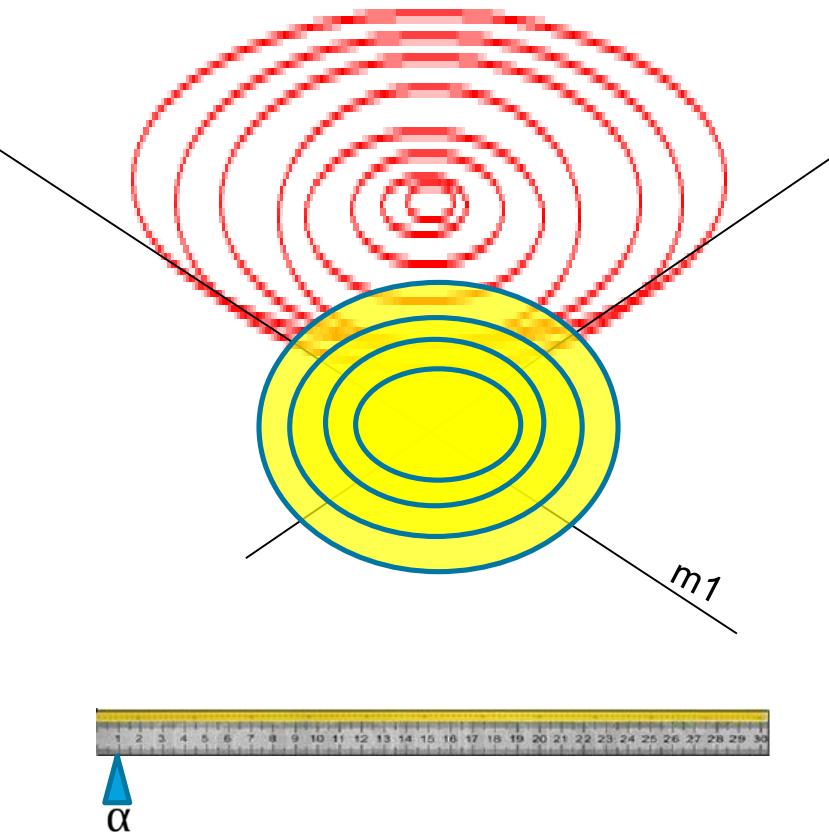
Regularising Linear Models (Ridge Constraint)



1. Yellow circle is the Ridge constraint region representing the ridge penalty (sum of squared coeff)
 1. Any combination of m_1 and m_2 that fall within yellow is a possible solution
 1. The most optimal of all solutions is the one which satisfies the constraint and also minimizes the SSE (smallest possible red circle)
 1. Thus the optimal solution of m_1 and m_2 is the one where the yellow circle touches a red circle.

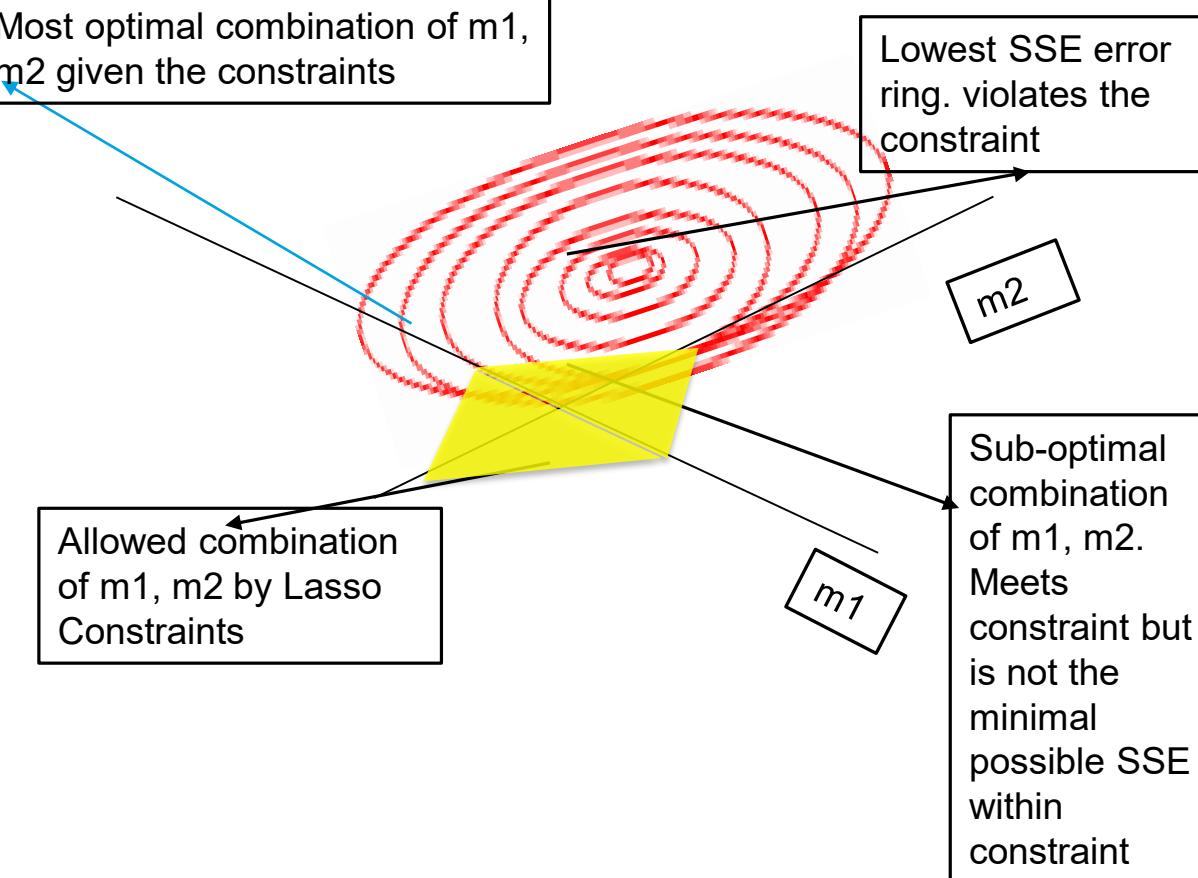
The point to note is that the red rings and yellow circle will never be tangential (touch) on the axes representing the coefficient. Hence Ridge can make coefficients close to zero but never zero. You may notice some coefficients becoming zero but that will be due to roundoff...

Regularising Linear Models (Ridge Constraint)



1. As the lambda value (shown here as alpha) increases, the coefficients have to become smaller and smaller to minimize the penalty term in the cost function i.e. the
 1. The larger the lambda, smaller the sum of squared coefficients should be $\lambda \sum_{j=1}^p \beta_j^2$ as a result the tighter the constraint region
 1. The tighter the constraint region, the larger will be the red circle in the contour diagram that will be tangent to the boundary of the yellow region
 1. Thus, higher the lambda, stronger the shrinkage, the coefficient shrink significantly and hence more smooth the surface / model
 1. More smoother the surface, more likely the model is going to perform equally well in production
 1. When we move away from a model with sharp peaks and valleys (complex model) to smoother surface (simpler models), we reduce the variance errors but bias errors go up.
 1. Using gridsearch, we have to find the right value of lambda which results in right fit, neither too complex nor too simple a model

Regularising Linear Models (Lasso Constraint)



1. Yellow rectangle is the Lasso constraint region representing the Lasso penalty (sum coeff)
 1. Any combination of m_1 and m_2 that fall within yellow is a possible solution
 1. The most optimal of all solutions is the one which satisfies the constraint and also minimizes the SSE (smallest possible red circle)
 1. Thus the optimal solution of m_1 and m_2 is the one where the yellow rectangle touches a red circle.

The beauty of Lasso is, the red circle may touch the constraint region on the attribute axis! In the picture above the circle is touching the yellow rectangle on the m_1 axis. But at that point m_2 coefficient is 0! Which means, that dimension has been dropped from analysis. Thus Lasso does dimensionality reduction which Ridge does not

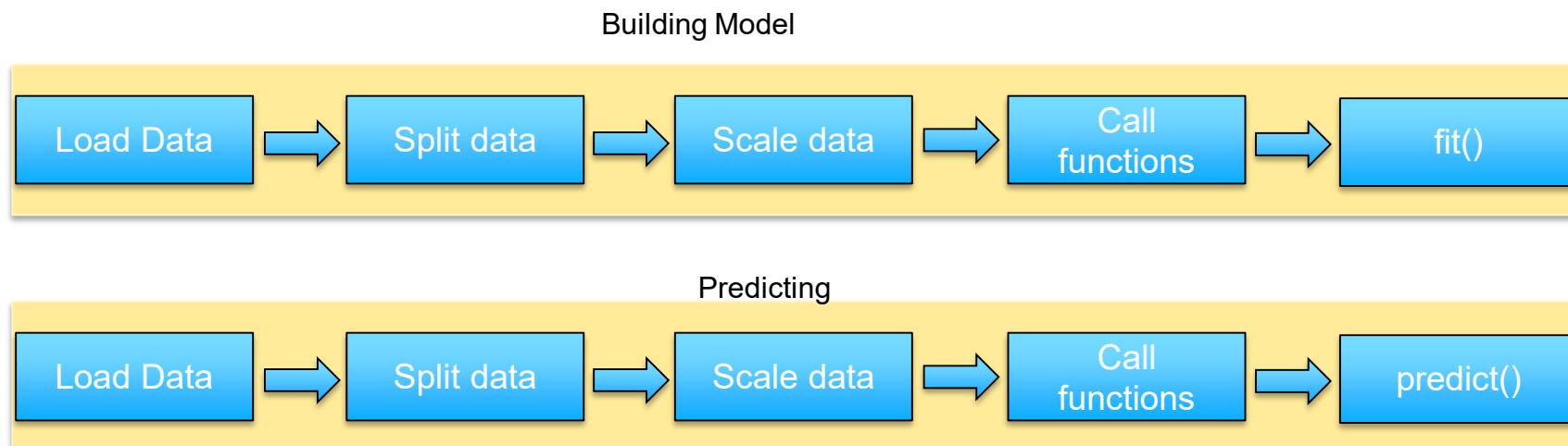
Pipelines

What is a pipeline-

1. Almost always, we need to tie together many different processes that we use to prepare data for machine learning based model
1. It is paramount that the stage of transformation of data represented by these processes are standardized
1. Pipeline class of sklearn helps simplify the chaining of the transformation steps and the model
1. Pipeline, along with the GridsearchCV helps search over the hyperparameter space applicable at each stage

Pipelines

1. Sequentially apply a list of transforms and a final estimator.
2. Intermediate steps of the pipeline must be ‘transforms’, that is, they must implement fit and transform methods.
3. The final estimator only needs to implement fit
4. Helps standardize the model project by enforcing consistency in building testing and production. Ref: <https://scikit-learn.org/stable/modules/compose.html>



Ref:Pipeline_Simple.ipynb, Pipeline_Gridsearch.ipynb

Build a pipeline

1. Import the pipeline class
 - a. `from sklearn.pipeline import Pipeline`
2. Instantiate the class into an object by listing out the transformation steps. In the following example, a scaling function is followed by the SVC algorithm
 - a. `pipe = Pipeline([("scaler", MinMaxScaler()), ("lr", logisticregression())])`
1. Call the fit() function on the pipeline object
 - a. `pipe.fit(X_train, y_train)`
1. Call the score() function on the pipeline object or predict() function
 - a. `pipe.score(X_test, y_test)`

In the step 2b, the pipeline object is created using a dictionary of key:value pairs. The key is specified in strings for e.g. “scaler” followed by the function to be called.

The key is the name given to a step.

`Pipeline_simple.ipynb`

Build a pipeline (Contd...)

1. The pipeline object requires all the stages included to have a “transform()” function except for the last stage which is an estimator.
1. The transform step transforms the input data. The transformed output of a stage is the input to the next stage
1. During the call “pipeline.fit()”, the pipeline calls the fit and transform functions on each step in sequence. For the last step, only the fit function is called
1. While predicting using pipeline, similarly transform function in all the stages followed by a predict function in the last stage is performed
1. The pipeline object does not need to have a predict function. It only needs to have a fit function at least

Build a pipeline (Contd...)

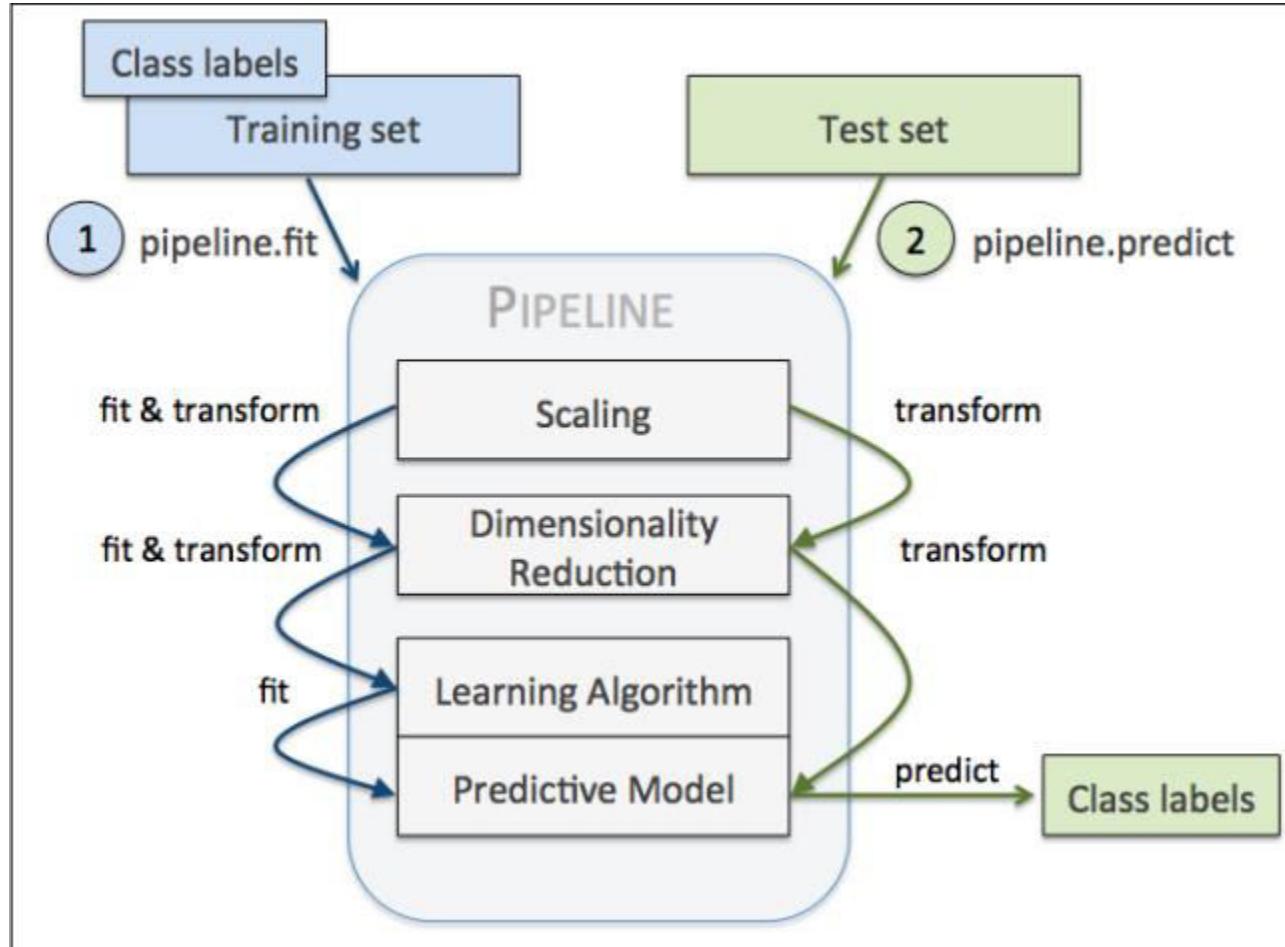


Image Source : Python Deeper Insights Into Machine Learning

make_pipeline

1. Creating the pipeline could be cumbersome. Specifying a name to each stage may not be necessary
1. Alternatively there is a “make_pipeline()” function that will create the pipeline and automatically name each step. We do not need to specify a name
 - a. `from sklearn.pipeline import make_pipeline`
 - b. `pipe = make_pipeline(MinMaxScaler(), (SVC()))`
 - c. `print(" Pipeline steps:\n{}".format(pipe.steps))`
1. Note, we have not specified any name to the stages. The names will be automatically assigned and are usually lowercase of the class names

`pipeline_introduction.ipynb`

make_pipeline

1. Download Wisconsin Breast Cancer dataset from UCI
2. Create dataframe
3. Split into X and y
4. Transform y into numerical using labelencoder
5. Split data into training testing
6. Standardize dataset using standard scalar
7. Use PCA to reduce to 2 dimensions
8. Use logistic regression on the reduced dimensions
9. Create a pipeline
10. Do a fit and score

Pipeline_GridSearchcv_wisc_bc_da
ta_classwork.ipynb

HyperParameter Tuning

Hyper Parameters & Tuning

1. Hyper parameters are like handles available to the modeler to control the behavior of the algorithm used for modeling
1. Hyper parameters are supplied as arguments to the model algorithms while initializing them. For e.g. setting the criterion for decision tree building
`"dt_model = DecisionTreeClassifier(criterion = 'entropy')"`
1. To get a list of hyper parameters for a given algorithm, call the function `get_params()`...for e.g. to get support vector classifier hyper parameters
 1. `from sklearn.svm import SVC`
 2. `svcl= SVC()`
 3. `svcl.get_params()`

Ref: Models_HyperParameters.ipynb
2. Hyper parameters are not learnt from the data as other model parameters are. For e.g. attribute coefficients in a linear model are learnt from data while cost of error is input as hyper parameter. **Ref:** model_parameters.csv



model_parameters

Hyper Parameters & Tuning

5. Fine tuning the hyper parameters is done in a sequence of steps
 1. Selecting the appropriate model type (regressor or classifier such as `sklearn.svm.SVC()`)
 2. Identify the corresponding parameter space (Ref: `model_parameters.csv`)
 3. Decide the method for searching or sampling parameter space;
 4. Decide the cross-validation scheme to ensure model will generalize
 5. Decide a score function to use to evaluate the model
5. Two generic approaches to searching hyper parameter space include
 1. `GridSearchCV` which exhaustively considers all parameter combinations
 2. `RandomizedSearchCV` can sample a given number of candidates from a parameter space with a specified distribution.
Ref: `GridSearchSimpleExample.ipynb`
5. While tuning hyper parameters, the data should have been split into three parts – Training, validation and testing to **prevent data leak**
5. The testing data should be separately transformed * using the same functions that were used to transform the rest of the data for model building and hyper parameter tuning

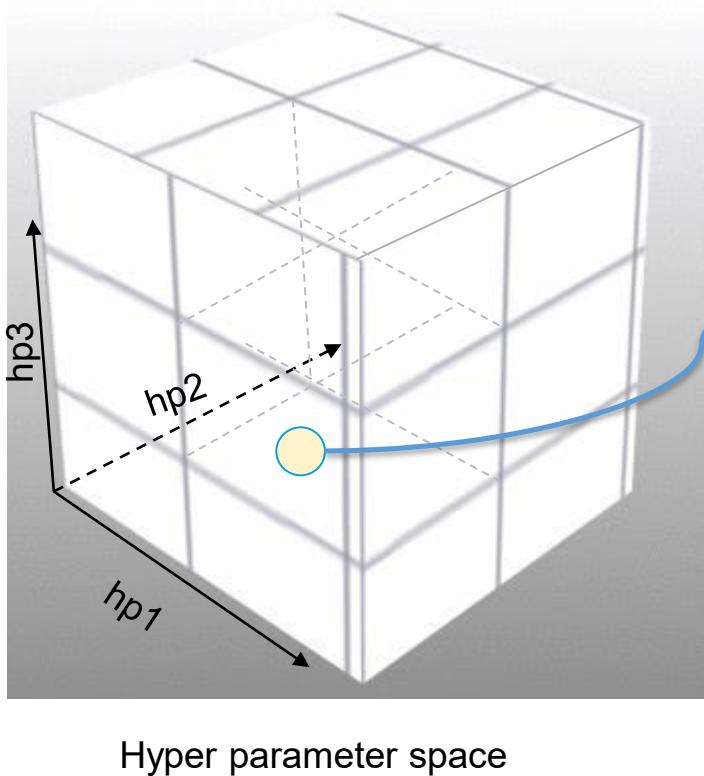
* Any transformation where rows influence each other. For e.g. using `zscore`. `OneHotCode` transformation does not come into this category. It can be done before splitting the data

Hyper Parameters & Tuning (GridsearchCV/ RandomizedSearchCv)

GridsearchCV –

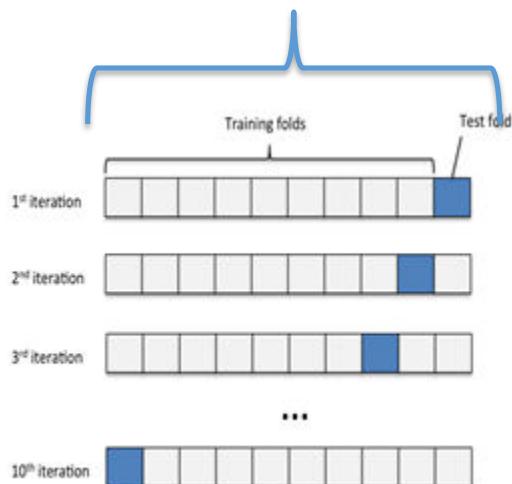
1. Is a basic optimal hyperparameter tuning technique.
2. It builds a model for each permutation of all of the given hyperparameter values
3. Each such model is evaluated and ranked.
4. The combination of hyperparameter values that gives the best performing model is chosen
5. For every combination, cross validation is used and average score is calculated
6. This is an exhaustive sampling of the hyperparameter space and can be quite inefficient

GridSearchCV



One combination of hyper parameters used K times to train and test. The avg score of the K times is the score associated with this combination

This will repeat for all possible combinations i.e. all the cells in the space.



Ref: Simple_Gridsearch.ipynb ,
GridSearchCV_compare_models.ipynb,

Design GridSearch with Pipeline

1. Use the earlier wisc_bc_dataset
2. Standardize dataset using standard scalar
3. Use PCA
4. Use SVC
5. Create a pipeline for scaling, pca and svc
6. Create param grid for PCA and SVC stage separately
7. Use Gridsearchcv with 5 CV
8. Print the best parameters
9. Print the best score

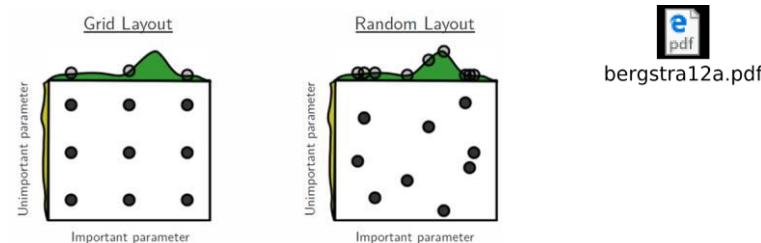
Pipeline_GridSearchcv_wisc_bc_da
ta_classwork.ipynb

Hyper Parameters & Tuning (GridsearchCV/ RandomizedSearchCv)

RandomizedSearchCV –

1. Random search differs from grid search. Instead of providing a discrete set of values to explore on each hyperparameter (parameter grid), we provide a statistical distribution.
1. Values for the different hyper parameters are picked up at random from this combine distribution
1. The motivation to use random search in place of grid search is that for many cases, hyperparameters are not **equally** important.

A Gaussian process analysis of the function from hyper-parameters to validation set performance reveals that for most data sets only a few of the hyper-parameters really matter, but that different hyper-parameters are important on different data sets. This phenomenon makes grid search a poor choice for configuring algorithms for new data sets. - [Bergstra, 2012](#)

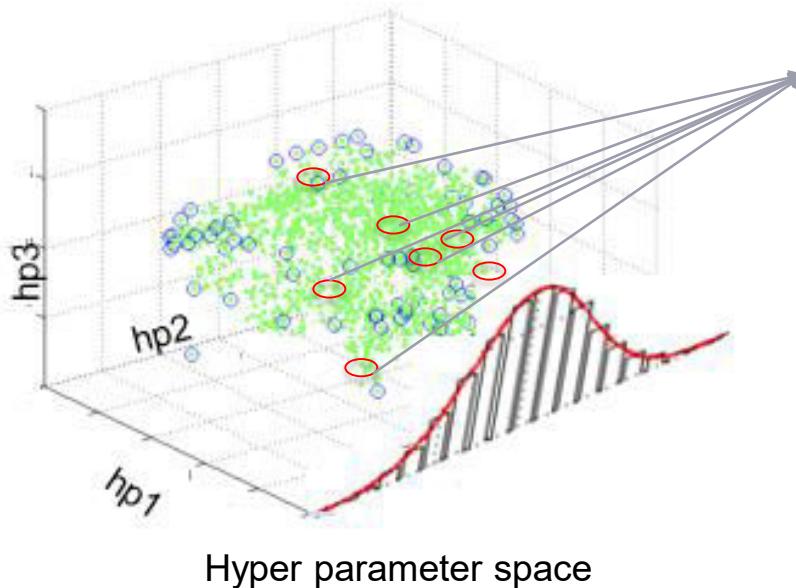


Picture by [Bergstra, 2012](#)

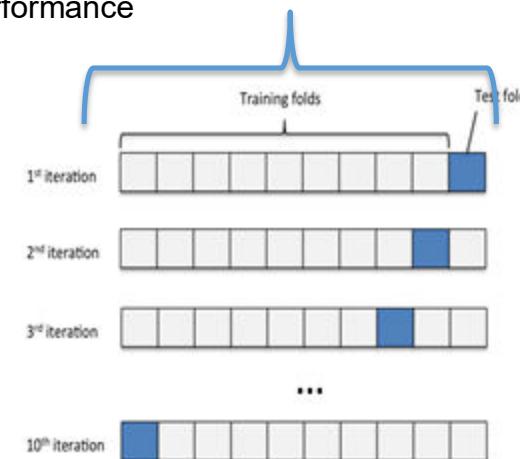


bergstra12a.pdf

RandomizedSearchCV



Randomly pick up n_iter samples from the hyper parameter distribution as sample, Use it K times and find avg performance



Ref: RandomizedSearchCV_GridSearchCV.ipynb

4. In contrast to GridSearchCV, not all combinations are evaluated. A fixed number of parameter settings is sampled from the specified distributions.
5. The number of parameter settings that are tried is given by n_iter
6. If all parameters are presented as a list, sampling without replacement is performed. If at least one parameter is given as a distribution, sampling with replacement is used. It is highly recommended to use continuous distributions for continuous parameters
7. Randomsearch has higher chance of hitting the right combination than gridsearch. Ref: Decision_Tree_Regressor_Concrete_Regularization_GridSearchCV.ipynb

ThankYou

Recommendation systems

Agenda

- Why recommendation systems?
- Real world examples and historical trends
- Basic techniques of recommendation
 - Popularity based
 - Classification
 - Content based
 - Collaborative filtering
 - Hybrid approaches
- Evaluation of recommendation
- Python examples
- Advances in recommendation systems

We are overloaded

- Thousands of news articles and news blogs each day
- Millions of movies, books, music tracks online
- Several thousands of ad messages sent to us each day
- But is it new topic?
 - what is new?



From scarcity to abundance

Limited resources:

- Shelf space
- No of hours for TV shows
- No of theaters for Movies

Web enabled near-zero-cost dissemination of information about products

- Reliance vs Amazon
- **long tail phenomenon**
 - Cut off point
 - Area of the curve



Why Recommendation System?

What we can solve?

Help user find item of their interest.

Help item provider deliver their items to right user.

- Identify products most relevant to the user
- Personalized content
- Eg. Top n offers

Help website improve user engagement.

Recommender system creates a matching between users and items and exploits the **similarity between users/items** to make recommendations.

What can be recommended?

Advertising messages

Jobs

Movies

Research papers

Books

Investment choices

Music tracks

TV programs

News articles

Citations

Restaurants

Cloths

Future friends (Social network sites)

Online mates (Dating services)

Courses in e-learning

Supermarket goods

User and matching items

Amazon **Users**: members, **Items**: products, books etc.

Netflix **Users**: members, **Items**: movies, TV shows

LinkedIn **Users**: members, **Items**: members

Facebook **Users**: members, **Items**: jobs

Power of Recommendations*:

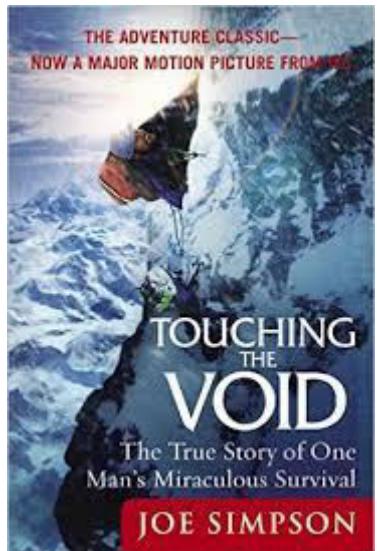
Netflix: – 2/3 rented movies are from recommendation

Google News – 38% more click-through are due to recommendation

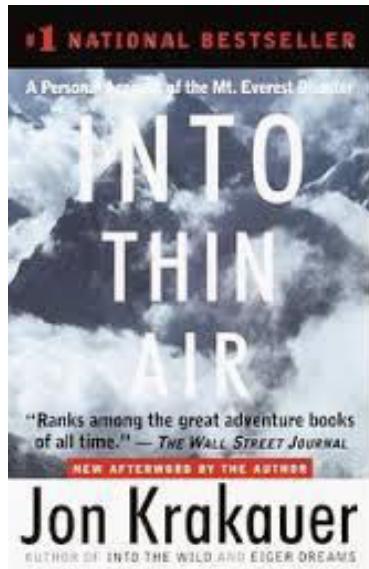
Amazon – 35% sales are from recommendation

*(Celma & Lamere, ISMIR 2007)

Recommendation historical trends



Published in 1988



Published in 1996

In 1988, Mr. Joe Simpson, an English mountain climber, wrote a book called *Touching the Void*. It got good reviews but was considered as modest success and it was soon forgotten.

However, a decade later, an interesting thing happened. Mr. Jon Krakauer wrote a book called *Into Thin Air*, a **thriller** about a mountain-climbing tragedy, which became a publishing sensation. Suddenly, *Touching the Void* started to sell again.

Real world examples

- GroupLens(<https://grouplens.org/>)
 - Helped in development of initial recommender systems by pioneering initial collaborative filtering models
 - Provided various datasets –MovieLens and BookLens
- Amazon –Did lot of work on implementing commercial recommender systems
 - They also implemented lot of computational improvements
- Netflix Prize
 - Pioneered Latent Factor/Matrix Factorization models
- Google –You Tube
 - Hybrid recommendation systems
 - Deep Learning Based Systems
- Social Network Recommendations

Perfect matching item may not exist

I want a laptop with 500GB HD, 8GB memory and i5 processor for \$400



Product #1

- HD: 500 GB
- Memory: 8 GB
- Processor: i7
- Price: \$550

Product #2

- HD: 250 GB
- Memory: 4 GB
- Processor: i3
- Price: \$350

Product #3

- HD: 250 GB
- Memory: 6 GB
- Processor: i5
- Price: \$450

Types of recommendation systems

- Popularity based recommendations
- Classification model based
- Content based recommendations
- Nearest neighbour collaborative filtering:
 - User based
 - Item based
- Hybrid approaches
- Association rule mining

Fundamental concepts of similarity

Euclidean Distance

Cosine Similarity:

Cosine Similarity is the cosine of the angle between the 2 vectors A and B

Closer the vectors, smaller will be the angle and hence larger their cosine value.

$$\text{Inner}(x, y) = \sum_i x_i y_i = \langle x, y \rangle$$

$$\text{CosSim}(x, y) = \frac{\sum_i x_i y_i}{\sqrt{\sum_i x_i^2} \sqrt{\sum_i y_i^2}} = \frac{\langle x, y \rangle}{\|x\| \|y\|}$$

Pearson Similarity:

Pearson similarity is also another measure of finding similarity between two vectors.

Pearson correlation and cosine similarity are invariant to scaling.

Cosine similarity is NOT invariant to shifts. Pearson correlation is invariant to shifts.

$$\begin{aligned}\text{Corr}(x, y) &= \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2} \sqrt{\sum(y_i - \bar{y})^2}} \\ &= \frac{\langle x - \bar{x}, y - \bar{y} \rangle}{\|x - \bar{x}\| \|y - \bar{y}\|} \\ &= \text{CosSim}(x - \bar{x}, y - \bar{y})\end{aligned}$$

Correlation is the cosine similarity between centered versions of x and y & between -1 to 1

Fundamentals

Jaccard Similarity:

Similarity is **defined as the** number of users which have rated item A and B divided by the number of users who have rated either A or B

Intersection over Union

It is used for comparing **both** similarity and diversity of **two** sets

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}.$$

If \mathbf{x}, \mathbf{y} are two vectors with all real x_i, y_i then jaccardian similarity coefficient:

$$J(\mathbf{x}, \mathbf{y}) = \frac{\sum_i \min(x_i, y_i)}{\sum_i \max(x_i, y_i)},$$

The Jaccard distance, which measures dissimilarity between sample sets, is complementary to the Jaccard coefficient and is obtained by subtracting the Jaccard coefficient from 1

Popularity based Recommender System

Popularity based recommendation system works by recommending items viewed/purchased by most people and rated high

Recommendations: **Ranked list of items by their purchase count / viewed count**

“Popular News”

It uses:

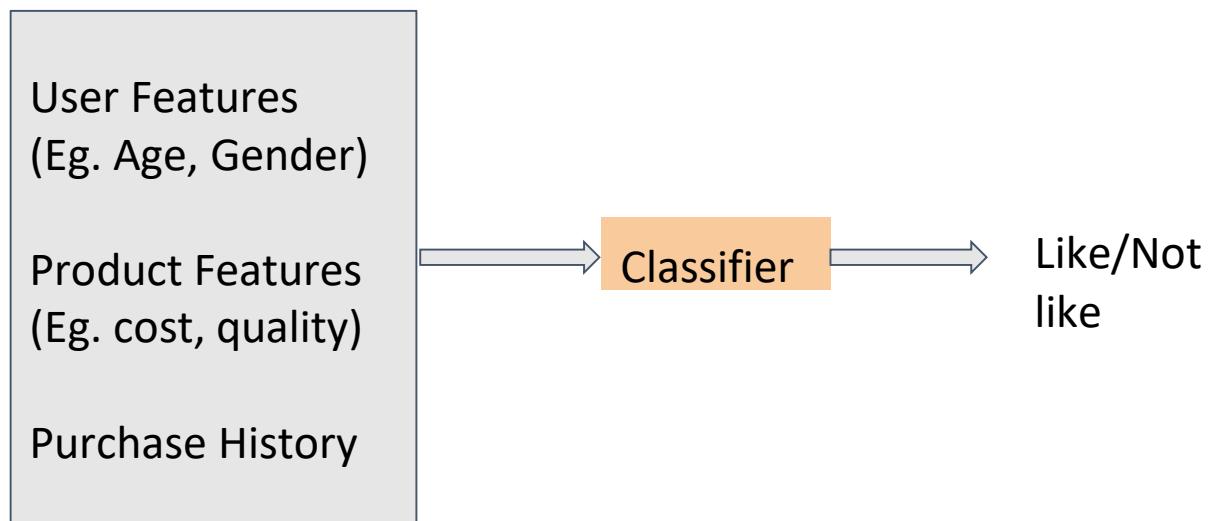
- Can use context
- Purchase history
- User and item features
- Scalable

Not a personalized recommendation

The screenshot shows the Google News homepage at <https://news.google.com/news/headlines?hl=en&gl=US&ned=us>. The interface includes a navigation bar with back, forward, and search functions, and tabs for Headlines, Local, For You, and U.S. The main content area is titled "Top Stories" and displays two news articles. The first article is about Senator Chuck Grassley and the release of transcripts from a Russian lawyer meeting, with related coverage links for Trump Jr. and Kushner. The second article is about Donald Trump's visit to Davos and his efforts to mend strained ties with Britain. A sidebar on the left lists other news sections like World, U.S., Business, Technology, Entertainment, Sports, Science, and Health, along with a "Manage sections" option.

Classification model

Uses features of both products as well as users in order to predict whether a user will like a product or not.



Limitations.

1. It is not easy to collect **quality information** about products and users.
2. Even if we are able to collect good information, they may not be sufficient to make a good classifier
3. Scalability issue

- The outcome can be 1 if the user likes it else 0
- Incorporates personalization

Content based recommendations

Recommendations are based on information on the content of items rather than on other users' opinions.

Main idea:

If a user likes an item then he/she will also like a “similar” item

Recommend items based on their similarity:

- Recommend items to customer x similar to previous items rated highly by x

Uses a machine learning algorithm to induce a profile of the users preferences from examples based on a featural description of content.

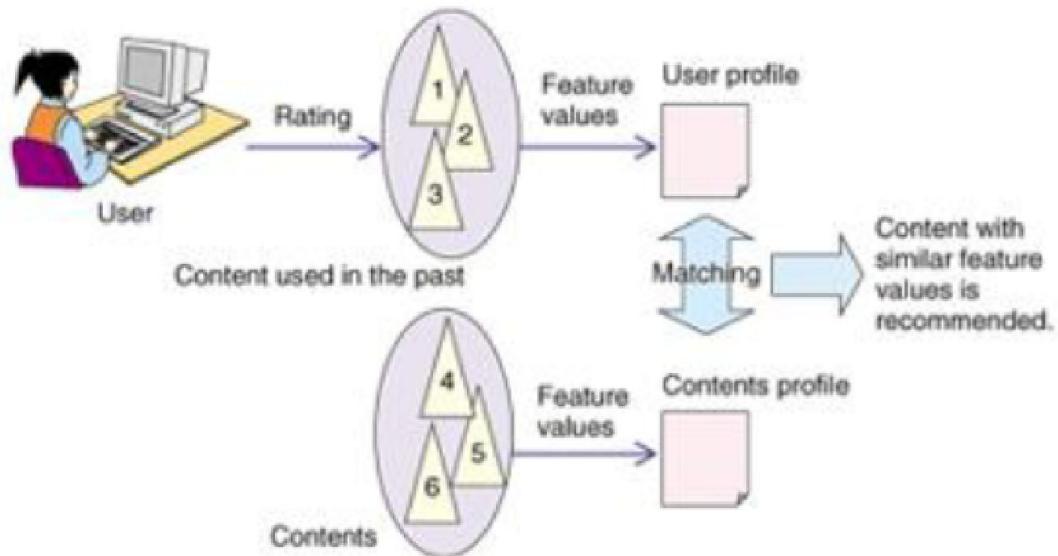
No need for data on other users.

- No cold-start or sparsity problems.

Able to recommend to users with unique tastes.

Techniques that can be used: Cosine similarity

Content based recommendation system



Item Profiles:

- set of features
- vector

User profiles:

- Weighted average of rated item profiles
- Normalize rating using average rating of user

Content based recommendation system

Advantages

- Content-based recommender systems **don't require a lot of user data**.
- We just **need item data** and you're able to start giving recommendations to users.
- Recommendation engine does not depend on lots of user data, so it is possible to give recommendations to even your first customer as long as you have adequate data to build his user profile. **Does not suffer from cold start**
- **Less expensive to build and maintain**

Challenges

- Your item data **needs to be well distributed**
- **Availability of features** which explain Items and user preferences
- Recommendations will likely be **direct substitutes**, and not complements, of the item the user interacted with. This is one of the key reasons why collaborative filtering provides better recommendations
- **Less dynamic**

Singular Value Decomposition (SVD)

Supplementary "Recommendation Systems" Lecture

Dr. Sunil Chinnamgari

Prerequisites

- Matrix Transpose
- Matrix Multiplication
- Identity Matrix
- Orthogonal Matrix
- Orthonormal Matrix
- Diagonal Matrix
- Determinant of a Matrix
- Eigen Values
- Eigen Vectors
- Gram-Schmidt orthonormalization process

What is SVD?

Given a rectangular matrix A, the linear algebra theorem SVD specifies that

$$A_{mn} = U_{mm} S_{mn} V_{nn}^T$$

In other words, A can be broken down into the product of three matrices - an orthogonal matrix U, a diagonal matrix S, and the transpose of an orthogonal matrix V

Additional properties that hold good are :

$U^T U = I$, $V^T V = I$; the columns of U are orthonormal eigen vectors of AA^T , the columns of V are orthonormal eigen vectors of $A^T A$, and S is a diagonal matrix containing the square roots of eigenvalues from U or V in descending order.

Start with the matrix

$$A = \begin{bmatrix} 3 & 1 & 1 \\ -1 & 3 & 1 \end{bmatrix}$$

In order to find U , we have to start with AA^T . The transpose of A is

$$A^T = \begin{bmatrix} 3 & -1 \\ 1 & 3 \\ 1 & 1 \end{bmatrix}$$

so

$$AA^T = \begin{bmatrix} 3 & 1 & 1 \\ -1 & 3 & 1 \end{bmatrix} \begin{bmatrix} 3 & -1 \\ 1 & 3 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 11 & 1 \\ 1 & 11 \end{bmatrix}$$

Next, we have to find the eigenvalues and corresponding eigenvectors of AA^T . We know that eigenvectors are defined by the equation $A\vec{v} = \lambda\vec{v}$, and applying this to AA^T gives us

$$\begin{bmatrix} 11 & 1 \\ 1 & 11 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \lambda \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

We rewrite this as the set of equations

$$11x_1 + x_2 = \lambda x_1$$

$$x_1 + 11x_2 = \lambda x_2$$

and rearrange to get

$$(11 - \lambda)x_1 + x_2 = 0$$

$$x_1 + (11 - \lambda)x_2 = 0$$

Solve for λ by setting the determinant of the coefficient matrix to zero,

$$\begin{vmatrix} (11 - \lambda) & 1 \\ 1 & (11 - \lambda) \end{vmatrix} = 0$$

which works out as

$$(11 - \lambda)(11 - \lambda) - 1 \cdot 1 = 0$$

$$(\lambda - 10)(\lambda - 12) = 0$$

$$\lambda = 10, \lambda = 12$$

our eigenvectors For $\lambda = 10$ we get

$$(11 - 10)x_1 + x_2 = 0$$

$$x_1 = -x_2$$

$$x_1 = -x_2$$

which is true for lots of values, so we'll pick $x_1 = 1$ and $x_2 = -1$ since those are small and easier to work with. Thus, we have the eigenvector $[1, -1]$ corresponding to the eigenvalue $\lambda = 10$. For $\lambda = 12$ we have

$$(11 - 12)x_1 + x_2 = 0$$

$$x_1 = x_2$$

and for the same reason as before we'll take $x_1 = 1$ and $x_2 = 1$

These eigenvectors become column vectors in a matrix ordered by the size

the eigenvector for $\lambda = 12$ is column one

the eigenvector for $\lambda = 10$ is column two

$$\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

Finally, we have to convert this matrix into an orthogonal matrix which we do by applying the Gram-Schmidt orthonormalization process to the column vectors. Begin by normalizing \vec{v}_1 .

$$\vec{u}_1 = \frac{\vec{v}_1}{|\vec{v}_1|} = \frac{[1, 1]}{\sqrt{1^2 + 1^2}} = \frac{[1, 1]}{\sqrt{2}} = \left[\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}} \right]$$

Compute

$$\vec{w}_2 = \vec{v}_2 - \vec{u}_1 \cdot \vec{v}_2 * \vec{u}_1 =$$

$$[1, -1] - \left[\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}} \right] \cdot [1, -1] * \left[\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}} \right] =$$

$$[1, -1] - 0 * \left[\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}} \right] = [1, -1] - [0, 0] = [1, -1]$$

and normalize

$$\vec{u}_2 = \frac{\vec{w}_2}{|\vec{w}_2|} = \left[\frac{1}{\sqrt{2}}, \frac{-1}{\sqrt{2}} \right]$$

to give

$$U = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{bmatrix}$$

The calculation of V is similar.

V is based on $A^T A$,

Find the eigenvalues of $A^T A$

find corresponding eigenvectors

use the Gram-Schmidt orthonormalization process to convert that to an orthonormal matrix.

All this to give us

$$V = \begin{bmatrix} \frac{1}{\sqrt{6}} & \frac{2}{\sqrt{5}} & \frac{1}{\sqrt{30}} \\ \frac{2}{\sqrt{6}} & -\frac{1}{\sqrt{5}} & \frac{2}{\sqrt{30}} \\ \frac{1}{\sqrt{6}} & \frac{\sqrt{5}}{\sqrt{5}} & \frac{-5}{\sqrt{30}} \end{bmatrix}$$

when we really want its transpose

$$V^T = \begin{bmatrix} \frac{1}{\sqrt{6}} & \frac{2}{\sqrt{6}} & \frac{1}{\sqrt{6}} \\ \frac{2}{\sqrt{6}} & -\frac{1}{\sqrt{5}} & 0 \\ \frac{1}{\sqrt{6}} & \frac{\sqrt{5}}{2} & \frac{-5}{\sqrt{30}} \end{bmatrix}$$

For S we take the square roots of the non-zero eigenvalues and populate the diagonal with them, putting the largest in s_{11} , the next largest in s_{22} and so on until the smallest value

$$S = \begin{bmatrix} \sqrt{12} & 0 & 0 \\ 0 & \sqrt{10} & 0 \end{bmatrix}$$

Now we have all the pieces of the puzzle

$$A_{mn} = U_{mm} S_{mn} V_{nn}^T = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{\sqrt{2}}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} \sqrt{12} & 0 & 0 \\ 0 & \sqrt{10} & 0 \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{6}} & \frac{2}{\sqrt{6}} & \frac{1}{\sqrt{6}} \\ \frac{\sqrt{6}}{2} & \frac{-1}{\sqrt{5}} & 0 \\ \frac{\sqrt{5}}{2} & \frac{\sqrt{5}}{2} & \frac{-5}{\sqrt{30}} \end{bmatrix} =$$

$$\begin{bmatrix} \frac{\sqrt{12}}{\sqrt{2}} & \frac{\sqrt{10}}{\sqrt{2}} & 0 \\ \frac{\sqrt{12}}{\sqrt{2}} & \frac{-\sqrt{10}}{\sqrt{2}} & 0 \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{6}} & \frac{2}{\sqrt{6}} & \frac{1}{\sqrt{6}} \\ \frac{\sqrt{6}}{2} & \frac{-1}{\sqrt{5}} & 0 \\ \frac{1}{\sqrt{30}} & \frac{2}{\sqrt{30}} & \frac{-5}{\sqrt{30}} \end{bmatrix} = \begin{bmatrix} 3 & 1 & 1 \\ -1 & 3 & 1 \end{bmatrix}$$

Fundamental concepts of similarity

Euclidean Distance

Cosine Similarity:

Cosine Similarity is the cosine of the angle between the 2 vectors A and B

Closer the vectors, smaller will be the angle and hence larger their cosine value.

$$\text{Inner}(x, y) = \sum_i x_i y_i = \langle x, y \rangle$$

$$\text{CosSim}(x, y) = \frac{\sum_i x_i y_i}{\sqrt{\sum_i x_i^2} \sqrt{\sum_i y_i^2}} = \frac{\langle x, y \rangle}{\|x\| \|y\|}$$

Pearson Similarity:

Pearson similarity is also another measure of finding similarity between two vectors.

Pearson correlation and cosine similarity are invariant to scaling.

Cosine similarity is NOT invariant to shifts. Pearson correlation is invariant to shifts.

$$\begin{aligned}\text{Corr}(x, y) &= \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2} \sqrt{\sum(y_i - \bar{y})^2}} \\ &= \frac{\langle x - \bar{x}, y - \bar{y} \rangle}{\|x - \bar{x}\| \|y - \bar{y}\|} \\ &= \text{CosSim}(x - \bar{x}, y - \bar{y})\end{aligned}$$

Correlation is the cosine similarity between centered versions of x and y & between -1 to 1

Fundamentals

Jaccard Similarity:

Similarity is **defined as** the number of users which have rated item A and B divided by the number of users who have rated either A or B

Intersection over Union

It is used for comparing **both** similarity and diversity of **two** sets

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}.$$

If \mathbf{x}, \mathbf{y} are two vectors with all real x_i, y_i then jaccardian similarity coefficient:

$$J(\mathbf{x}, \mathbf{y}) = \frac{\sum_i \min(x_i, y_i)}{\sum_i \max(x_i, y_i)},$$

The Jaccard distance, which measures dissimilarity between sample sets, is complementary to the Jaccard coefficient and is obtained by subtracting the Jaccard coefficient from 1

Collaborative Filtering

Idea: If a person X likes items A, B, C and Y like B,C,D then they have similar interests and **X should like item D and Y should like item A.**

This algorithm is entirely based on the user's past behaviour and not on the context.

This makes it one of the **most commonly used algorithm** and is not dependent on any additional information.

Basic assumptions:

- Customers who had similar tastes in the past, will have similar tastes in the future
- Users give ratings to catalog items (implicitly or explicitly)

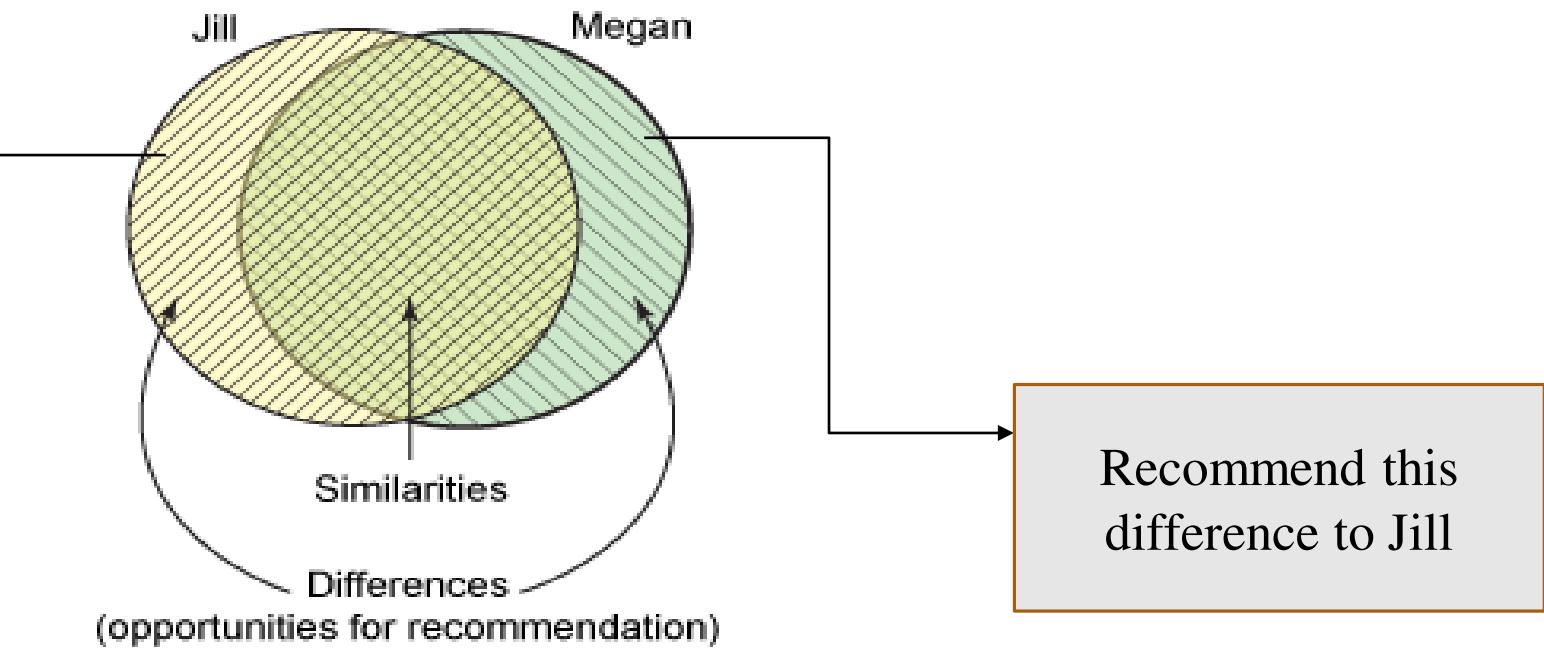
Examples:

- Product recommendations by e-commerce player like Amazon and merchant recommendations by banks like American Express.

User-User Collaborative filtering

Item-Item Collaborative filtering

Collaborative Filtering



User based nearest neighbour Collaborative filtering

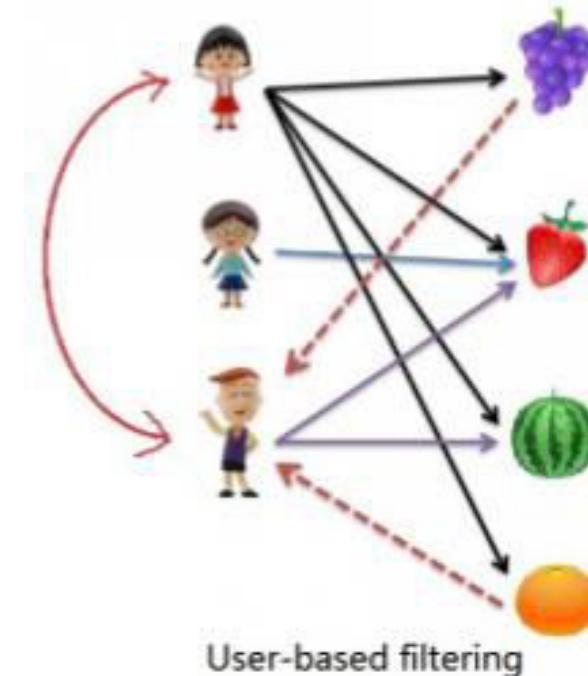
Find out **the users** who have a similar taste of products as the current user chosen.

Similarity is based upon similarity in **their** purchasing behaviour.

“User A is similar to user B because both purchased items X, Y and z.”

Memory-based: the rating matrix is directly used to find neighbours / make predictions

Does not scale well for most real-world scenarios



Source: <http://www.salemmarafi.com/code/collaborative-filtering-with-python/>

User-User collaborative filtering

User/Movie	x1	x2	x3	x4	x5	Mean User Rating
A	4	1	-	4	-	3
B	-	4	-	2	3	3
C	-	1	-	4	4	3

$$r_{AC} = [(1-3)*(1-3) + (4-3)*(4-3)] / [((1-3)^2 + (4-3)^2)^{1/2} * ((1-3)^2 + (4-3)^2)^{1/2}] = 1$$

$$r_{BC} = [(4-3)*(1-3) + (2-3)*(4-3) + (3-3)*(4-3)] / [((4-3)^2 + (2-3)^2 + (3-3)^2)^{1/2} * ((1-3)^2 + (4-3)^2 + (4-3)^2)^{1/2}] = -0.866$$

Item based nearest-neighbour collaborative filtering

Reverse of user based collaborative filtering

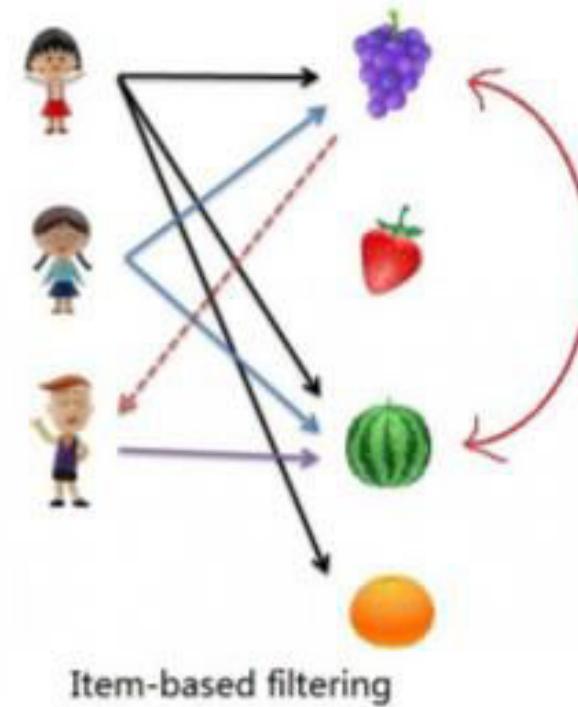
Recommend items to the user that are similar to the items the user has bought.

Similarity is based upon co-occurrence of purchases:

- Use the similarity between items (and not users) to make predictions

“Items X and y were purchased by both users A and B, so they are similar.”

- Item-based CF is an example for model-based approaches



Item based collaborative filtering: Example

History Matrix

	B	A	C
Ted			
Carol			
Bob		?	

Co-occurrence Matrix: Items by Items

A		B		C	
A		1	2		
B	1			1	
C	2	1			

Bob's Recommendations= [C, B]

Item-Item collaborative filtering

User/Movie	x1	x2	x3	x4	x5
A	4	1	2	4	4
B	2	4	4	2	1
C	-	1	-	3	4
Mean Item Rating	3	2	3	3	3

$$C_{14} = [(4-3)*(4-3) + (2-3)*(2-3)] / [((4-3)^2 + (2-3)^2)^{1/2} * ((4-3)^2 + (2-3)^2)^{1/2}] = 1$$

$$C_{15} = [(4-3)*(4-3) + (2-3)*(1-3)] / [((4-3)^2 + (2-3)^2)^{1/2} * ((4-3)^2 + (1-3)^2)^{1/2}] = 0.94$$

Market basket analysis

Discovers co-occurrence relationships among activities performed.

Market basket analysis can be used to **divide customers into groups**

Market basket analysis may provide the retailer with information **to understand the purchase behavior of a buyer**.

“customers who bought book A also bought book B”

When one super market chain discovered in its analysis that male customers that bought diapers often bought beer as well, have put the diapers close to beer coolers, and their sales increased dramatically.

Might tell a retailer that customers often purchase shampoo and conditioner together, so putting both items on promotion at the same time would not create a significant increase in revenue, while a **promotion involving just one of the items** would likely drive sales of the other.

Association rule mining

Rules of the form $x \rightarrow Y$

from a set of sale transactions.

It's common use in shopping behaviour analysis

Can be used as the basis for **decisions about marketing activities** such as, e.g., promotional pricing or product placements.

It is NOT sequence mining: Association rule learning typically does not consider the order of items either within a transaction or across transactions.

Example database with 5 transactions and 5 items

transaction ID	milk	bread	butter	beer	diapers
1	1	1	0	0	0
2	0	0	1	0	0
3	0	0	0	1	1
4	1	1	1	0	0
5	0	1	0	0	0

$$I = \{\text{milk, bread, butter, beer, diapers}\}$$

$$\{\text{butter, bread}\} \Rightarrow \{\text{milk}\}$$

Performance metrics

- RMSE
- MAE –Mean Absolute Error
- Accuracy
- ROC curve
- Precision
- Recall
- Precision Recall Curve: Evaluation of top n recommendations

Evaluation measures

Out of all the recommended items, how many user actually liked the recommendations?

What ratio of items that a user likes were actually recommended.

- **Precision:** Out of all items retrieved, how many are relevant

- It is based on true positives and false positives.

- It shows how many selected items are actually also relevant.

- True positives (TP) are the positive guesses made that were actually correct while the false positives (FP) are the positive guesses made that were incorrect.

$$Precision = \frac{TP}{(TP + FP)}$$

- **Recall:** How many relevant items retrieved from all relevant items

- is based on true positives and false negatives.

- It shows how many of the relevant information is selected.

- The true positives (TP) are again the positive guesses made that were actually correct while the false negatives (FN) are negative guesses while they should have been positive.

$$Recall = \frac{TP}{(TP + FN)}$$

Evaluation measures

- **Accuracy:** Percentage guessed correct from total guesses
 - method that is based on exact matches.
 - This is the most simple evaluation method, but may perform poorly based on the data.
 - If the labels of your data mostly consist of one specific cluster, the classifiers may classify all the data as that cluster. Since the data contains mostly that cluster, its accuracy will be high due to one classification.
 - This may cause promising results, but they are not correct. So for this measure, it is necessary to have information about the labels of your data. The number itself is not enough for performance conclusions.

$$Accuracy = \frac{\text{Good guesses}}{\text{Total Guesses}} \cdot 100$$

RoC and Precision Recall curve

Receiver operating characteristics curve.

A plot of true positive fraction (= sensitivity) vs. false positive fraction (= 1 – specificity) for all potential cut-offs for a test.

A ROC curve plots recall (true positive rate) against fallout (false positive rate) for increasing recommendation set size.

In Top-20 recommender example,

The 20 items you recommend for a user are the Positive items, and the unrecommended items are Negative.

Precision Recall Curve:

A precision-recall curve shows the relationship between precision (= positive predictive value) and recall (= sensitivity) for every possible cut-off.

The main difference between ROC curves and precision-recall curves is that the number of true-negative results is not used for making a PRC.

Curve	x-axis		y-axis	
	Concept	Calculation	Concept	Calculation
Precision-recall	Recall	$TP / (TP + FN)$	Precision	$TP / (TP + FP)$
ROC	1-specificity	$FP / (FP + TN)$	Sensitivity	$TP / (TP + FN)$

Hybrid recommender systems

Multiple recommender systems are combined to improve recommendations

- Although any type of recommender systems can be combined a common approach in industry is to combine **content based approaches and collaborative filtering approaches**
- Content based models can be used to solve **the Cold Start and Gray Sheep problems** in Collaborative Filtering
- Some of the typical methods of Hybridization include
 - **Weighted** –Recommendations from each system is weighted to calculate final recommendation
 - **Switching** –System switches between different recommendation model
 - **Mixed** - Recommendations from different recommenders are presented together

A common approach is to use **Latent Factor models for high level recommendation** and then **improving them using content based systems** by using information on users or items

Summary

- Basics of recommendation system
- Popularity based recommendations
- Classification model based
- Content based recommendations
- Nearest neighbour collaborative filtering:
 - User based
 - Item based
- Hybrid approaches
- Python examples
- Association rule mining

Industry example of e-commerce

Events:-

- Tracks and stores on all consumer activity and behavior
- Each click on product, adding to wishlist, adding to cart, save for later and purchase

Ratings:-

- Assign implicit values on user actions
- Ratings of products from the users
- And Feedback

Filtering:-

Hybrid approach of Collaborative filtering and user based filtering

Considerations

- Sometimes not recommending or simple recommendation is the best option
- Privacy concerns in Recommendation systems
 - The case of Target Corporation
- Computational challenges in recommendation systems, can be costly to implement
 - •The final model built by winning Netflix team could not be implemented due to engineering challenges
 - •Good Data Collection, well thought out metrics are a must

References

1. [1. http://dataconomy.com/an-introduction-to-recommendation-engines/](http://dataconomy.com/an-introduction-to-recommendation-engines/)
2. [2. https://goo.gl/ehBnhf](https://goo.gl/ehBnhf)
3. [3. https://github.com/dvysardana/RecommenderSystems_PyData_2016](https://github.com/dvysardana/RecommenderSystems_PyData_2016)
4. [4. https://brenocon.com/blog/2012/03/cosine-similarity-pearson-correlation-and-ols-coefficients/](https://brenocon.com/blog/2012/03/cosine-similarity-pearson-correlation-and-ols-coefficients/)
5. [5. https://wiki.epfl.ch/edicpublic/documents/Candidacy%20exam/Evaluation.pdf](https://wiki.epfl.ch/edicpublic/documents/Candidacy%20exam/Evaluation.pdf)
6. [6. http://www.quuxlabs.com/blog/2010/09/matrix-factorization-a-simple-tutorial-and-implementation-in-python/](http://www.quuxlabs.com/blog/2010/09/matrix-factorization-a-simple-tutorial-and-implementation-in-python/)
7. Gunawardana, Asela, and Guy Shani. "A survey of accuracy evaluation metrics of recommendation tasks." *Journal of Machine Learning Research* 10.Dec (2009): 2935-2962.
8. Davis, Jesse, and Mark Goadrich. "The relationship between Precision-Recall and ROC curves." *Proceedings of the 23rd international conference on Machine learning*. ACM, 2006.

Further reading

Book: Recommender Systems An Introduction by Dietmar Jannach

Book: Mining Massive Datasets by Jure Leskovec, Anand Rajaraman, Jeff Ullman (www.mmds.org)

Coursera course on Recommender Systems, by University of Washington

Coursera course on Recommender Systems, by University of Minnesota

<https://dl.acm.org/citation.cfm?doid=2959100.2959166>