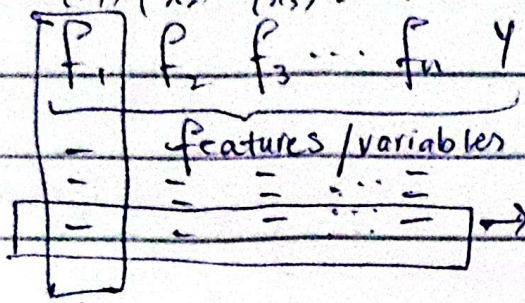
Exploratory Data Analysis

↳ Univariate

one variable

(x_1) (x_2) (x_3) ... (x_n) → label / target



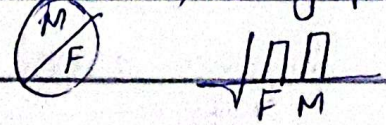
higher → higher analysis

features

Univariate → pick one feature for analysis

Descriptive analysis → we provide summary

↳ Visual analysis → Pi chart, bar graph



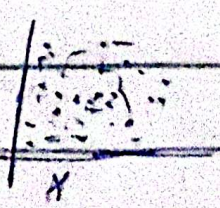
→ Python code

```
import matplotlib.pyplot as plt

fig1 = plt.figure()
figs.addSubplot(2,2,1)
plt.scatter('x_val', 'y_val')
```

row location } Row wise fill

- Scatterplot (X, Y)



grid type:

which = "major" → draw major points

- PIE PLOT (Catagorical Data > Variable)

plt.pie(value, label="label_name") (By default = %. can also do count)

- Erich (Beautify) - Term in DS to add values

- autopct = '%.2f%%' → 2 decimal place + % sign

auto percentage — starting and ending format

- startangle = 90 → default = 0°

value to start the ploting

- explode = [0,0,0,0,0,0] → default=0 means no explode

Number of pics/values in data — feature

- wedgeprops

- BAR PLOT - (Prefered Catagorical Data)

- Labels are not sorted in sort(). It only sort the values count.

① either sort the labels with values count

② or use (sort = False), then it will not sort and show the values.

By Default = count, also do count with slightly change

BAR Chart → catagorical Data (Freq. Distribution)

VS

- HISTROGRAM (Used for ~~discreet~~ discrete value)
  - use bin for discrete data to
    bin = int ((max (values) - min (values))/2)
  - Density plot → For bin (very difficult to evely distribute of real data)
    distribution → (Seaborn is used for density ( ) to show frequency distribution)
    FDF vs CDF → best

import seaborn as sns

① hist ② ecdf Draw: dist. plot

ax = sns.displot (data = values, kind = 'kde')

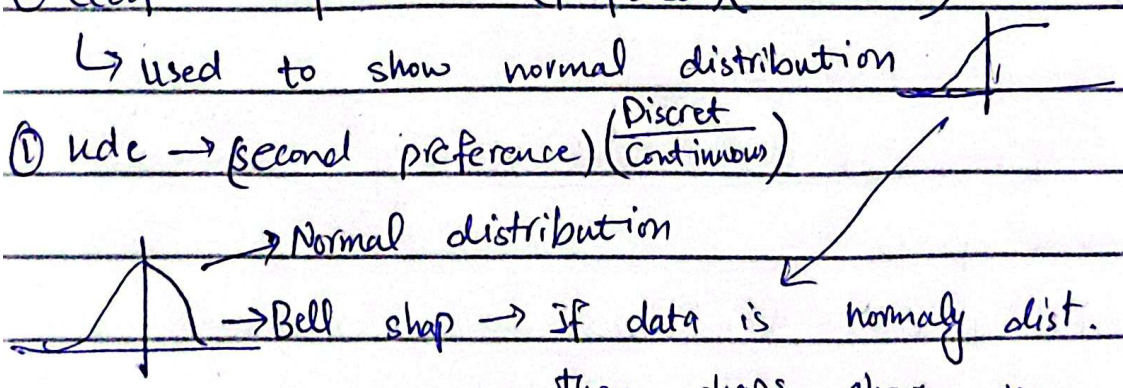
distribution plot

Kernel density estimate

① hist → histogram

② ecdf → empirical (prefered) (Discret / Continuous) variables

↳ used to show normal distribution



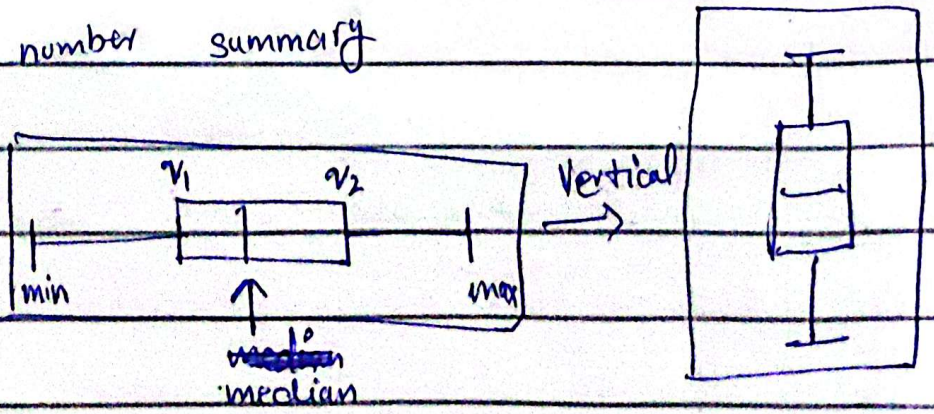
① kde → (second preference) (Discret / Continuous)

→ Normal distribution

→ Bell shap → if data is normaly dist. then shaps shows very goo shap

- BOX PLOT

aka 5 number summary



- Discrete / Continuous

Bivariate Data Analysis

- Scatter, Line, Count, Box, Swarm Plots

① Numerical Data ② Catagorical Data

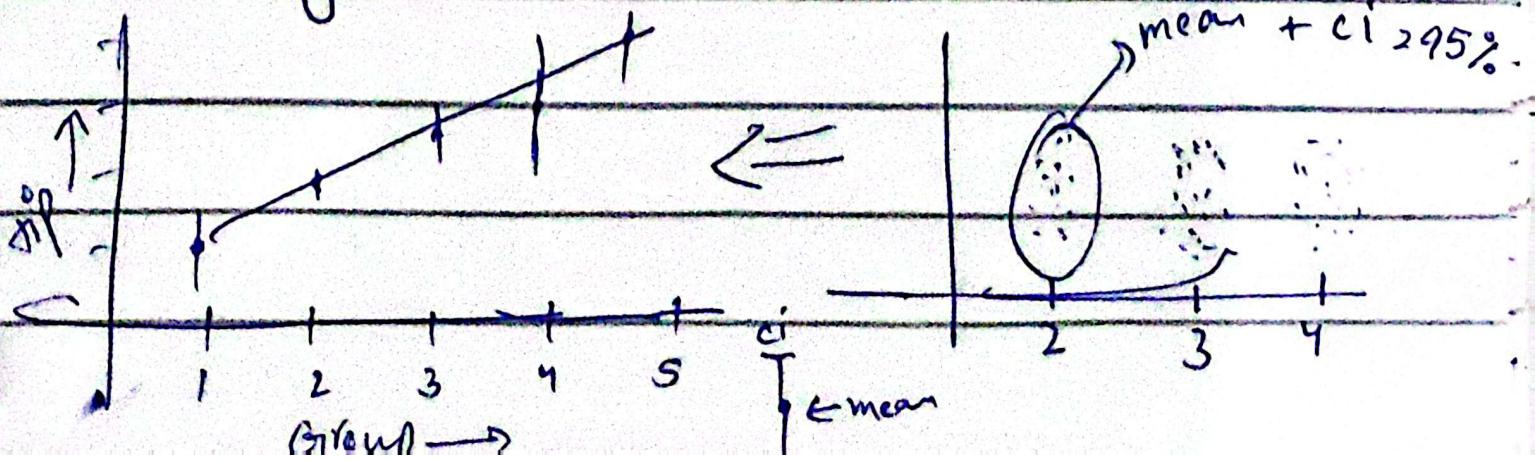
Corrdation → Increase of independent cause increase/decrease in dependent var.

Casuality → How much change in dependent is caused by independent

sns.relplot( ) → relation plot

sns.regplot( ) ⟶ add regression line

- for continous data (x_estimator) mean or average of data at a point is use along with confidence interval

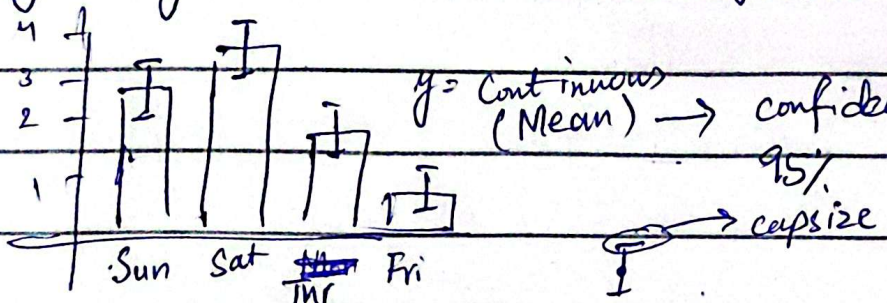


sns. Implot ( ) ⟶ Linear plot

Catagorical ⟶ against ⟷ Continuous

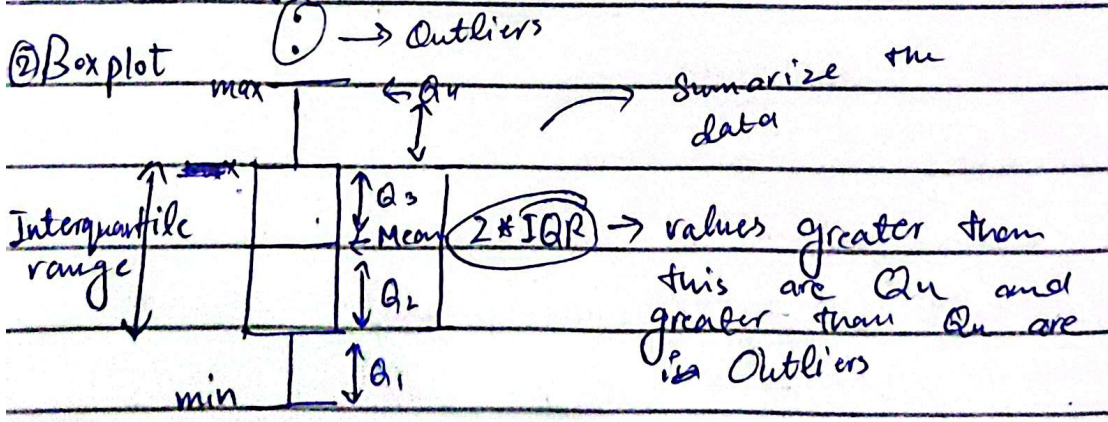
① Countplot ( ) ⟶ countplot used for catagorical with continuous data. e.g how many male, female paid tips.

E.g Day and customer ~~tips~~ group

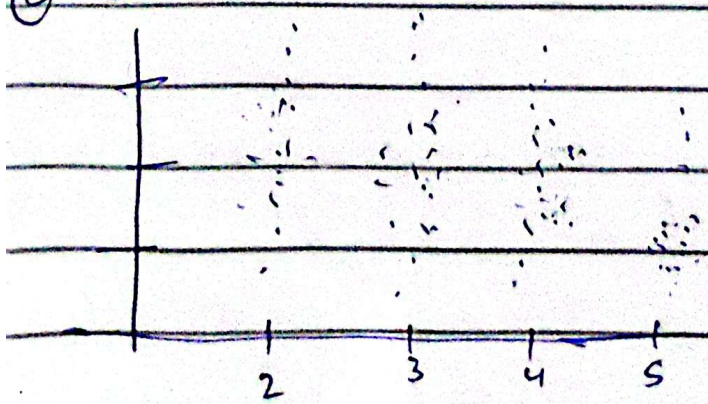


X = catagorical

② Boxplot

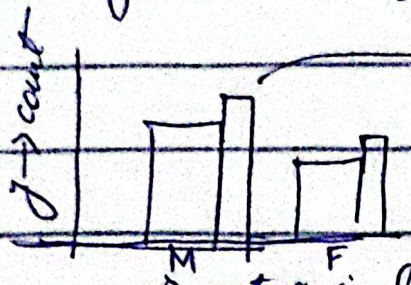


③ Swarm Plot



Catagorical → againt → catagorical

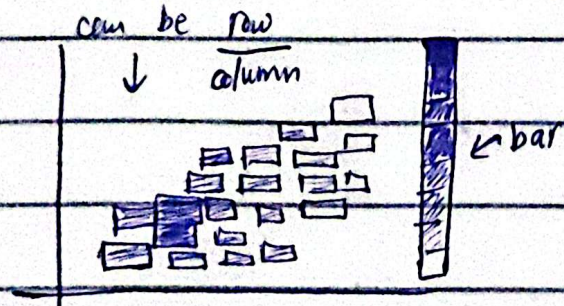
① Count Plot



x → catagorical ⇒ can be divided

② Distribution / Density Plot (Continuous VS catagorical)

No specific condition



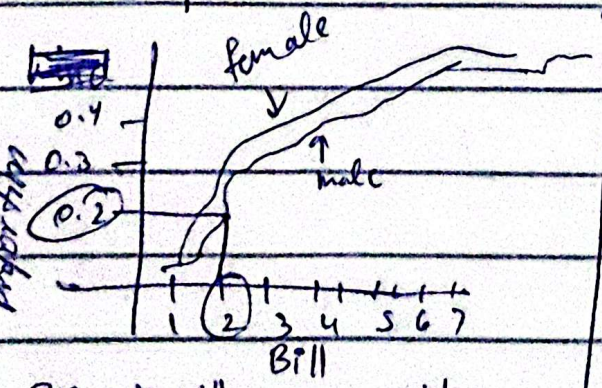
frequency

cdf

pdf

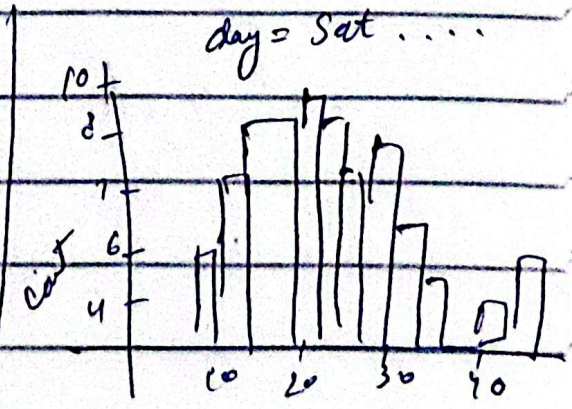
mdf

→ cdf



20 → bill are paid

0.2 percent be femail



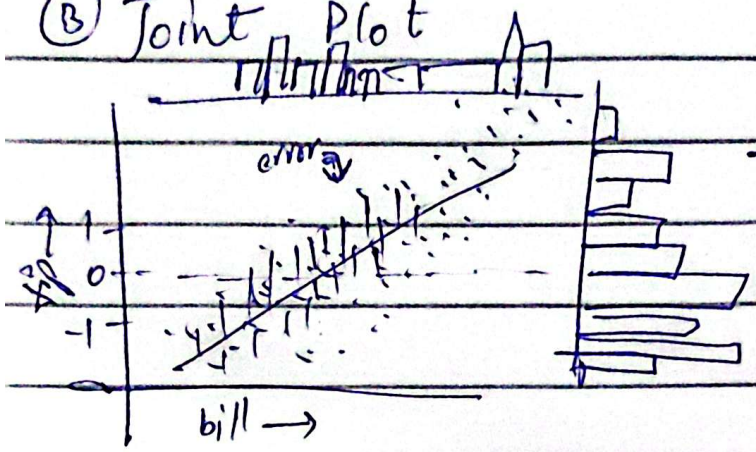
histogram → seperal for every day

Skewness

① Mean > Median > Mode → Positive Skew

② Mean < Median → Negative Skew

(B) Joint Plot



→ frequency

→ histogram on right side and on top.

kind = "~~reg~~", 'scatter', "reg", "hist", "resid"

↓

tells error distribution

→ Choose the graph on the basics on continuous or catagorical data.

Catagorical variable

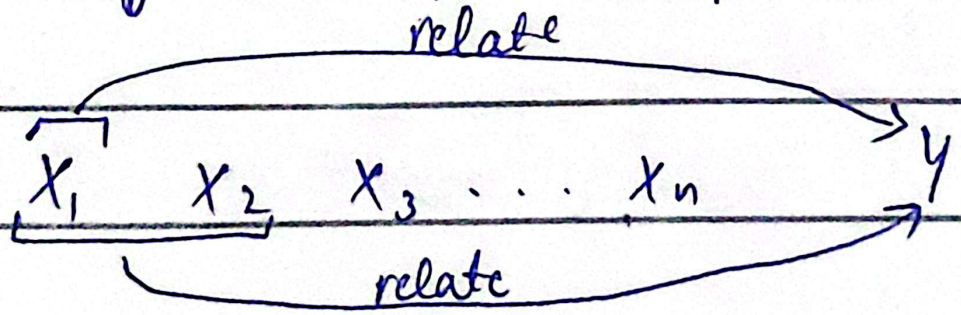
1
2
3
⋮
10

→ pick any number and find its ~~fe~~ frequency. If frequency > 1 then it is catagorical

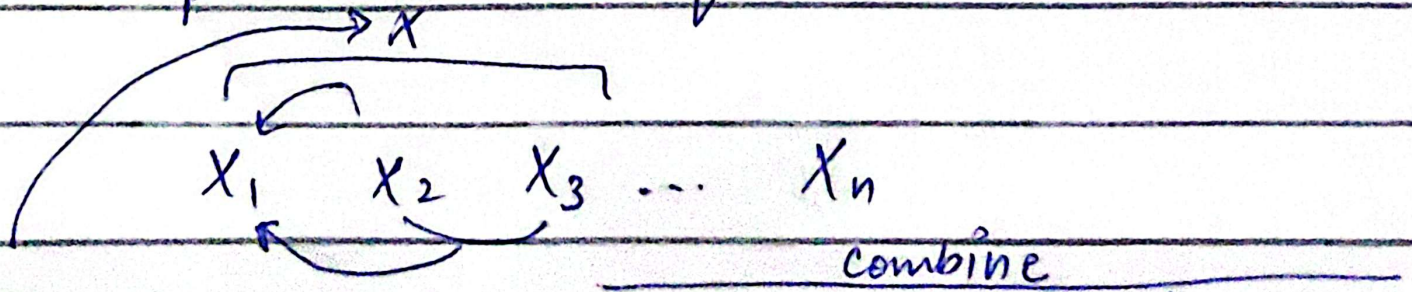
→ can make groups (grouping)

Multivariate Data Analysis

- Dependencye Technique (Dependen 2 Independent)



- Interdependence Technique



① Factor analysis – Reduce the # of variables

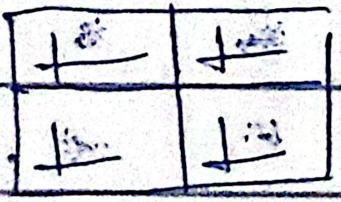
② Cluster analysis – ~~Combine~~ / Study together but not combined

- Scatter plot can be used to show multiple variables in 2D graph or plane.
It uses continuous variables as x and y and rest of any catagorical variable as: ① hue ② style ③ row/colum or both to show multiple variables on 2D graph.

① hue : means different color / shade

② style : (★, o, •, △)

③ row/colums : one graph plan is divided



2 row x 2 col

Thursday IDS 21-2

- Swarm plot
- Playfair plot (9 plots)
  - r values

- Graph Description (Explanation)
  ↳ Explain graph with well-known words and used different words each time even for same meaning
  [Important for IELTS]

Wednesday IDS 6-3

Machine Learning – Automates analytical model

– prediction on previous / existing data

– learn from experience

Train the machine how to learn

Machine Learning	Deep Learning
Structure Data	Unstructured Data
Data that can be represented in tabular form	Data that cannot be represented in tabular form
– Given features from data	– Given data and extract features on their own

(Goodness of Fit)

R-Squared

$$R^2 = 1 - \frac{\sum (y - \hat{y})^2}{\sum (y - \bar{y})^2}$$

$$R^2 = 1 - \frac{SSR}{SST} < 1$$

+1 → positive relation

-1 → negative relation

$$r = \frac{n(\sum xy) - (\sum x)(\sum y)}{\sqrt{n\sum x^2 - (\sum x)^2}\sqrt{n\sum y^2 - (\sum y)^2}}$$

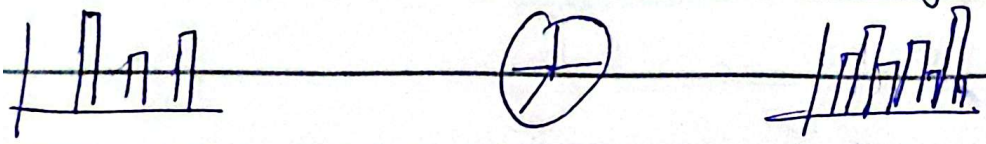
$$y = \frac{n\sum xy - \sum x \sum y}{n(\sum x^2) - (\sum x)^2} \quad / \quad y = \frac{n\sum xy - \sum x \sum y}{n\sum x^2 - (\sum x)^2}$$

$$r = \frac{n\sum xy - \sum x \sum y}{\sqrt{[n\sum x^2 - (\sum x)^2][n\sum y^2 - (\sum y)^2}}$$

$-1 < r < +1$

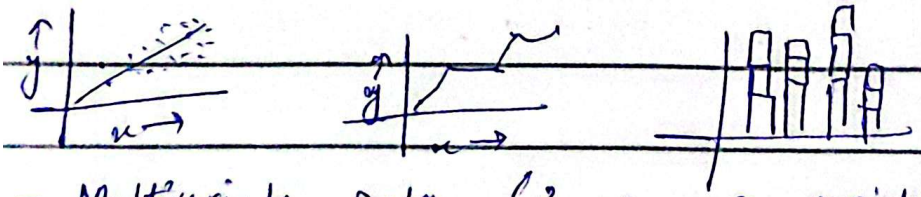
- Univariate Data / ~~Variable~~ (1 Variable)

① Bar chart ② Pi chart ② Histogram



- Bivariate Data (2 variables)

① Scatter Plot ② Line Plot ③ Stacked Bar chart



- Multivariate Data (3 or more variables)

① style • → △ → ★

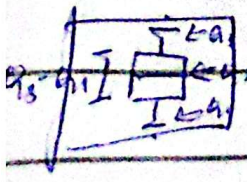
② hue, color, saturation

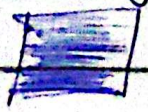
④ Row, column increase

- Univariate Data Analysis

① Bar chart ② Histogram ③ Pie chart

④ Box plot ⑤ Density Plot





- Bivariate Data Analysis

(1) scatter Plot (2) Line Plot (3) Count Plot

(4) Box Plot (5) Swarm Plot

- Multivariate Data Analysis (more than 2 var)

(1) ~~Inter~~dependent Technique

(2) Interdependent technique → understand the structure

↳ Factor analysis → combine highly corelated features together

↳ Cluster analysis → combine similar features

Increasing trend	Decreasing trend
(1) Increased dramatically	Slowed down noticeably
(2) Tripled	Modest — steady
(3) Increased strikingly	Levelling off — steady
(4) Steep upward trajectory	Declined
(5) Surged - increase suddenly	Dwindled
(6) Climbing more slowly	Plummeted
(7) Rising steadily	Tapered off
(8) No sign of leveling off	Decreased gradually
	~~Stumped~~ Fall down
(9) Increased exponentially	Fell off
(10) Increased Rapidly	Sank
(11) Exponential Growth	Waned
(12) Accelerated	To the Ground
(13) Rapid ascent	Going downwards
(14) Substantial Increase	Trend goin down
	Dropped straight down

Graph code

ax. sns.displot(data = values, kind = 'ecdf') # hist, kde

-> hist: This values generate histogram plot.

↳ display the distribution of data through bin and provides insights into the f or pdf

->kde: kernel ~~estimation plot~~ density estimation plot.

↳ used to estimate the pdf of a continuous var.

-> ecdf: empirical cumulative distribution function.

LR ~~$y = \theta_0 x + \theta_1$~~ $y = \beta_1 X + \beta_0 + \varepsilon$ ← error

Gradient Decent ↑ stop ↑ y intercut

$x_{n+1} = x_n + \alpha \Delta F(x_n)$

$F(x_n) = x^2$	Cost Function (MSE)
$x^2_0 = x^2 + \alpha \Delta(x^2)$	error = (guess - actual)
$x = x^2 + 2x$	$MSE = \frac{1}{n} \Sigma (y - \hat{y})^2$
$x = 2x + \alpha \Delta(2x)$	$y = mx + c$
$x = 2x + \alpha 2$	$MSE = \frac{\Sigma (y - \hat{y})^2}{n}$

$f(m,b) = \frac{\Sigma (y_i - (mx_i - b))^2}{n}$

$\frac{\partial f}{\partial m} = \frac{\Sigma -2 - 2(y - (mx+b))}{n}$

$\frac{\partial f}{\partial m} = \frac{1}{n} \Sigma -2 (y - (mx + b))$

$f(x = X)$

$\frac{1}{10}, \frac{2}{10}, =$ PMF (1)t
0.1 0.1
0.2 0.3

PMF → Frequency (individually) / total

PMF | 26 Bar

Discrete

~~$f(-\infty < X)$~~ $f(a < X < b)$

$\frac{x^{n+1}}{n+1}$

PDF | Normal Distribution

cont rang

$= \frac{1}{6\sqrt{2\pi}} e^{\frac{(x-\mu)^2}{26^2}}$

$6 = \sqrt{\frac{\Sigma(x-\mu)}{N}}$

$6 = \sqrt{\frac{(32.14-29.52)^2}{20}}$

$Z = \frac{x-\mu}{6}$

$z = \frac{x-\mu}{6/\sqrt{n}}$

$\frac{\sqrt{\text{sample mean}} - \sqrt{\text{population mean}}}{st / \sqrt{\text{number of sample}}}$

$z = \frac{1600 - 1570}{120/\sqrt{100}} = 2.51 > 1.96$

$b = \frac{n\Sigma xy - \Sigma x \Sigma y}{(n\Sigma x^2 - (\Sigma x)^2}$

$y = mx + b$

$a = \frac{\Sigma y - b\Sigma x}{N}$

$y = -4.4502 + 0.21^x$

$R^2 = 1 - \frac{\Sigma(y - \hat{y})^2}{\Sigma(y - \bar{y})^2}$

$r = \frac{n\Sigma xy - \Sigma x \Sigma y}{\sqrt{(n\Sigma x^2 - (\Sigma x)^2)(n\Sigma y^2 - (\Sigma y)^2)}}$

$MSB = \frac{\Sigma(y - \hat{y})^2}{n}$ $\hat{y} = mx + b$

$MSE = \frac{\Sigma(y - (mx+b))^2}{n}$

$m = m - \alpha \frac{\partial d}{dm}$

$b = b - \alpha \frac{\partial d}{\partial b}$

~~$MSE = \frac{2}{n}\Sigma$~~

$MSE = -\frac{2\Sigma(y - mx+b)^2}{n}$

$Z\text{-score} = \frac{x-\mu}{6} = \frac{\text{sample} - \text{pop}}{st}$

$2\text{-test} = \frac{\bar{x}-\mu}{6/\sqrt{n}}$

$b = \frac{n\Sigma xy - \Sigma x \Sigma y}{n\Sigma x^2 - (\Sigma x)^2}$

$a = \frac{n\Sigma y - b\Sigma x}{N}$

$r = \frac{n\Sigma xy - \Sigma xy}{}$

Wednesday IDS 20-3

kNN → Supervised learning

↳ vote from majority (most similar)
similarity

↳ used for classification

& regression

Binary classification → better

- non-parametric algorithm – don't asume about data
- lazy learner – nearly no learning insted it store the dataset
- Find the distance of all of the points in the plane (1000, ~~1000~~ 1000h) and then sort the distance points. The check the first k points and then predict the classes. k = any odd number.

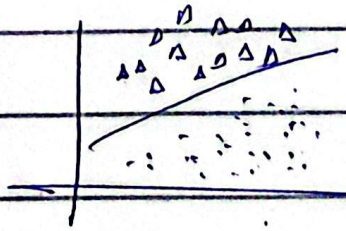
Distance functions

① Euclidean $\sqrt{\sum_{i=1}^{k}(x_i - y_i)^2}$

② Manhattan

③

Linear classify



Example

N.P (3,7) , k = 3

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

d =

Sort = {3, 3.6, 4, 5}

k

Class = B

X	Y	label	d
7	7	A	5
7	4	A	5
3	4	B	3
1	4	B	3.6

Value of k

- Value of k must be odd (3, 5, 7, 11)
- Small values of k can make noise
- larger values of k

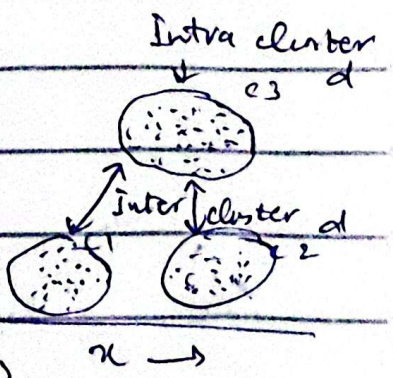
Table

Input Expected	Actual
value[i] = -999	
value = 1000, i=101	

Friday IDS 22-3

K-Means

- Clustring algo
- Intra cluster / within cluster distanst should be minimum



k → no. of cluster {k = 2, 3, ...}

↳ no. of centroids = center points of cluster

e → cluster

→ randomly initialize centroid values

-> find the distance of every point to centroid and the point which have minimum distance with centroid, group them e.g. $|x_i - c_1|, |x_2 - c_2|$

-> find the mean distance within centroid and move the centroid point in the middle of the cluster (update the value)

-> Do this until there is same values of average

no. of cluster

↑

Choosing the k value

- Elbow method — WCSS — Intra cluster ↓
- Expectation Maximization — Inter cluster ↑

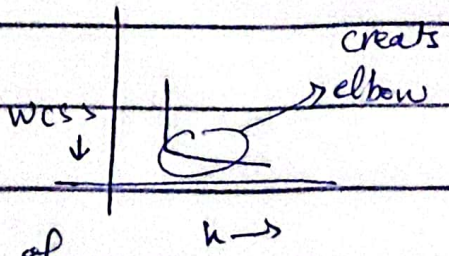
$$d = \sum_{j=1}^{c} \sum_{i=1}^{n} \|x_i - c_j\|^2$$

x_i → data points, c_j → clusters

- Elbow method — find the value of k
  ↳ within cluster sum of squares (WCSS)

$$\sum_{i=1}^{n} \|x_i - c\|^2$$

(within cluster)



- Choose the lower point of (K) → optimal number of clusters

k → integer

- Silhouette Method — used for verification mainly

b → distance of the point from cluter it does not belong

a → distance of the point from cluster it belongs => -1 < value < 1

kmean. inertia — } return sum of squares

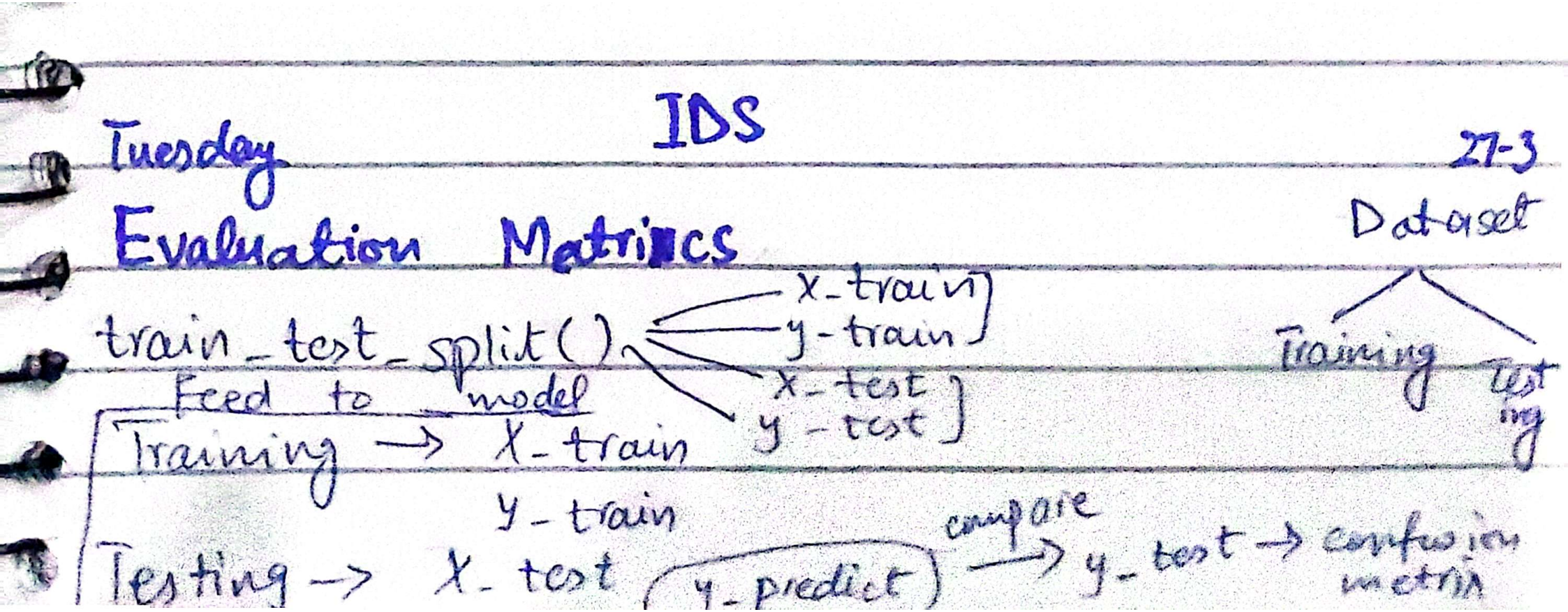
+ve → good, -ve → bad, 0 → overlapping

↓ optimal ↓ not optimal

IDS

Tuesday 27-3

Evaluation Matrics



Classification → Confusion Matrix

Actual \ Predicted	P: Yes	N: No
T	TP	FP
F	FN	TN

FP → Type 1 error

False Positive
False Alarm

FN → Type 2 error

↑ Type 1 → Overfit

↑ Type 2 → Underfit

1) Accuracy = $\frac{\text{Correct}}{\text{Total}}$

Example

Actual: Dog	
3	2
1	4

TP: 3 TN: 4

FN: 1 FP: 2

Accuracy = $\frac{3+4}{10} = \frac{7}{10} = 70\%$

Classification → Confusion Matrix

Actual \ Predicted	P: Yes	N: No
T	TP	FP
F	FN	TN

FP → Type 1 error

False Positive
False Alarm

FN → Type 2 error

① Accuracy = Correct / Total

↑ Type 1 → Overfit
↑ Type 2 → Underfit

Example Actual — Dog

Predicted \ Actual	3	2
	1	4

TP : 3 TN : 4
FN : 1 FP : 2

$$\text{Accuracy} = \frac{3+4}{10} = \frac{7}{10} = 70\%$$

Friday IDS 28-3

Naive Bayes - Bayes Theorm
- **Conditional Probility**

① Independent
② Equal Imp

A = X
B = y

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

$\Pi_{i=1}^{n}$ ← Product

Posterior = Prior × likelihood / evidence

$X = X_1, X_2, X_3, \ldots X_n$

$$P(y|X_1, X_2, X_3 \ldots X_n) = \frac{P(X_1, X_2, X_3 \ldots X_n|y) * P(y)}{P(X_1, X_2, X_3, \ldots X_n)}$$

solv

- Zero frequency problem => Laplacian Smoothing
  ↳ If data not exist of a feature then the probability is zero which makes the whole probability zero

NB Types

① Multinominal ② Bernoulli ③ Gausian

Ex#1

P(write | Infected) =

P(write | clean) =

P(Executable | Infected) =

P(Non-exe |

Total = 7

P(clean) = 3/7

P(Infected) = 4/7

Laplacian Smoothing

	write	executed	large
P(clean)	2/3	1/3	2/3 => 3/4
P(infected)	1/4	4/4 = 1	1 => 1/5

$P(\text{clean} | \text{Yes}) = \frac{2}{3} \times \frac{1}{3} \times \frac{3}{4} \times \frac{3}{7} = 0.071$ ↑

~~P(Infected | Yes)~~

$P(\text{clean} | \text{No}) = \frac{1}{4} \times 1 \times \frac{1}{5} \times \frac{3}{7} = 0.021$

Wednesday IS 3-4

Wednesday IDS 3-4

Spam Filter (Important)

- Types : Bayesian Filter

$P(spam|word)$

$$Pr = \frac{Pr(word|spam)\,Pr(spam)}{Pr(word)}$$

Total = $P_1 * P_2 * P_3$

Ham : Send me review

Spam : Send me password

$$P(S/W) = \frac{P(W/S) * P(S)}{P(W/S \cup h)}$$

$$P(W/S \cup h) = P(W/S) * P(S) + P(W|h) * P(h)$$

$$P = \frac{P_1 P_2 \dots P_n}{P_1 P_2 P_3 \dots P_n + (1-P_1)(1-P_2)\ddots(1-P_n)}$$

Laplace Smoothing {+2 in classes | +1 in features
every word is a feature

- E_n = "Review us now"

$$P(S/E_n) = \frac{P(E_N/S) * P(S)}{P(E_N)}$$

$P(E_N/S)$ zero for word which is no in E_n $(1-p)$

$Pr(\{1, 0, 1, 00, 0\} \mid \text{Spam})$ (train)

one ~~or the~~ for the word which exists in E_n (p)

- Threshold / Tolerance = 5%

Friday **SQE** 5-4

Quiz - 2

Friday **IDS** 5-4

Data Preprocessing (80% Time)

- Raw Data → apply → Prepared data → feed → algorithm

Raw Data → Data Preprocessing → Prepared Data → Feed Algorithm

- **Data Cleaning**

aka cleansing, scrubbing

- remove and fixed data
- remove duplicate data
- Irrelevant Data

affect models

- only knn can work on empty cells

① variance is zero ② Given no information eg email, phone

NaN for → Empty cells

Note :- If all of the data is 100% same or different then remove that column

Python → [NaN] → empty cells

Missing Values

① Missing Completely at Random (MCAR)
↳ missing data cannot be linked/related with other features

② Missing At Random (MAR)
↳ missing data can be related with other features

③ Missing Not At Random (MNAR)
↳ Unobserved data, cannot find any relation

Handling Missing Value

① Deleting the missing values
- If 2% of 100k entries are missing then deleting is a good approach. && MAR and MCAR then delete it.

② Imputing the missing values
- Replace with arbitrary value
- Replace with mean
- Replace with Mode (for categorical data)
- Replace with Median (for outliers)
- Replace with Previous fill, Replace with backward fill

Data Transformation

[String → Number]

Types:
① Label ② One Hot Encoding

Types
① Nominal (Categorical) ② Ordinal

Best for → Catagorical Data

One Hot Encoding (Example)

Color	_red	_green	_blue
Red	1	0	0
Green	0	1	0
Blue	0	0	1

Vectorize form

One Hot encoding Vector

Text → Number → NLP

→ Counter Vectorizer

Feature Scaling

Salary	Age
10k	22
20k	27
50k	39
↑	↑
Very big values affect the model	small values neglected by model

① Min-Max Normalization

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)}$$

② Standardization

$$x' = \frac{x - \bar{x}}{\sigma}$$

must

Normalization vs Standardization

↑ no Gaussian Distribution

↑ follow Gaussian Dist.

Data Preprocessing

- Feature Generation & Selection

- Selection

① Unsupervised

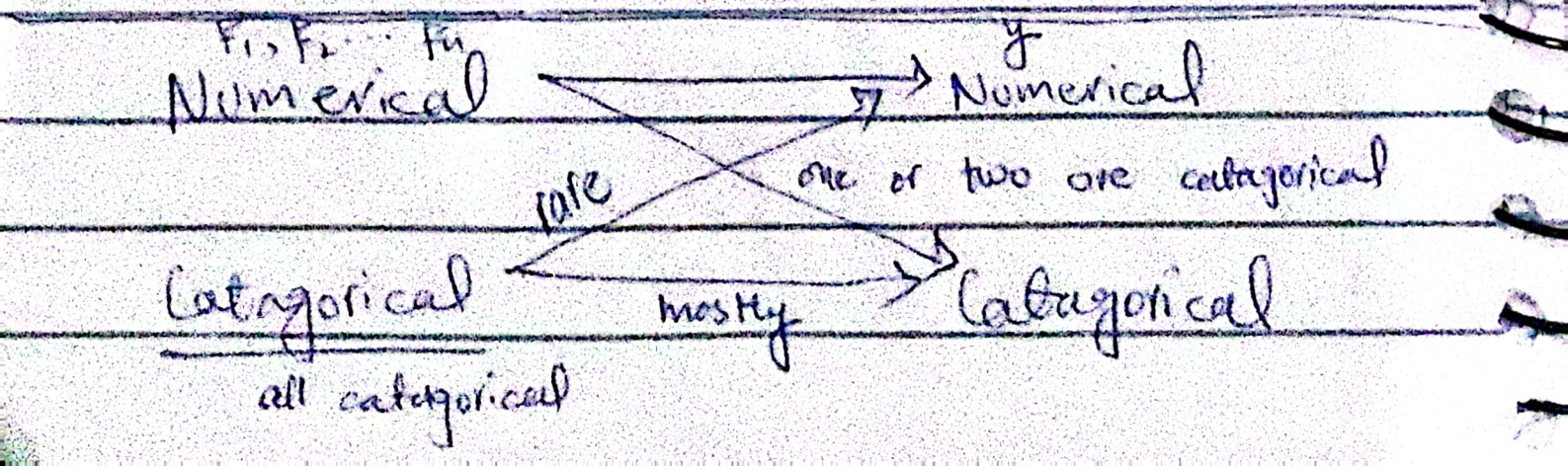
- Remove with zero or constant variance
- Remove missing values column.
- High multicolinerierty => High coorelation cause redundency. - Remove them. less variety

② Supervised

① Filters Method aka Single Factor Analysis

↳ coorelate ~~this~~ each feature with

↳ High coorelation with target / label



Not always increase accuracy

① Numerical → Numerical (Regression)
- Find correlation

② Numerical → Catagorical (Classifications)
↳ ① Information Gain / Mutul Information

② Wrappers
- Greedy Search

1st

$f_1 = 0.25$ (y)
$f_2 = 0.35$
$f_3 = 0.30$

2nd (Iterative)

$(f_3, f_1) = 0.47$ (y)
$(f_3, f_2) = 0.52$

(y) repeat with all features

→ Sequential Forward Slashing → Limit features
↳ cannot go back

→ Backward Eliminate
↳ can go backward and eliminate selected features

② Embedded Method
→ Decision Tree
→ Regression, Classification, Inform Gain

Regression D.T → Classification
→ Information Gain
Selection Model / Algo.
Learning Model / Algo.

Information Gain
Gini Index
Gain Ratio
ASM → Attribute Selection Method

Decision Tree

① Node →
② Edge →
③ Leaf →

Information Gain
↓ Depends
Entropy

Entropy → Randomness

Calculate Entropy for 9 → Yes / No only = 0

$$E(S) = \sum_{i=1}^{c} -p_i \log_2 p_i$$

$\log_2$ → log of base 2

Imp!

- Find target var classes

eg Yes = 9
No = 5
} 14

9, 5

$$E(5,9) = -\frac{9}{14}\log_2\frac{9}{14} - \frac{5}{14}\log_2\frac{5}{14}$$

Yes No

Entropy for multiple ~~Entropy~~ attributes

entropy
entropy
$f_1, f_2, f_3, \ldots f_n$ Y
entropy

$$E(I,X) = \sum_{c \in X} P(c)E(c)$$

$$E(a,b) = \left(\frac{-a}{a+b}\log_2\frac{a}{a+b}\right) + \left(\frac{-b}{a+b}\log_2\frac{b}{a+b}\right)$$

-ve in Entropy:

- $\log_2$ of ratio is always -ve so we put

Random Forest ← DT1, DT2, ... DTN

Information Gain → IG

⇒ Information Gain (T, X) = Entropy (T) − Entropy (X)

T → Total attri

X → single attri

⇒ Information Gain = Entropy (before) − $\sum_{j=1}^{K}$ Entropy (j, after)

− Select with the maximum IG

Wednesday IDS 24-4

Data Transformation

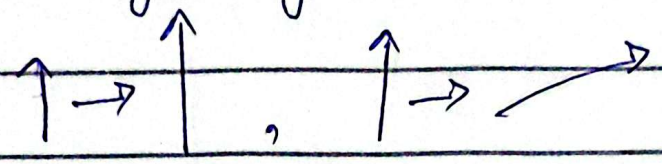
↳ Dimentionality Reduction

⇒ Transformation → change of shap

① Linear Transformation (Geometry)

+ straigh line transformation

- size may change — no curve, no bend

eg 

Eigenvectors and Eigenvalues (Linear algebra)

Matrix = A Vector = $\vec{v}$ → single column

$$\begin{bmatrix} 2 & 3 \\ 4 & 5 \end{bmatrix} * \begin{bmatrix} 4 \\ 9 \end{bmatrix} = \begin{bmatrix} \text{always vector} \end{bmatrix}$$

Dataset — scaled — always vector

- Resultant vector is either scale-up or scale-down without changing ~~in direction~~ linearity.

- Scale-up / Scale-down with an integer and that integer is called eigen value.

→ Eigenvector

$$A.\vec{v} = \lambda.\vec{v}$$

↑ scalar

$$A\vec{v} - \lambda\vec{v} = 0$$

$$\vec{v}(A - \lambda I) = 0$$

$\vec{v} \neq 0$ then $A - \lambda I = 0$

find $\vec{v}$?

λ → eigenvalue

v → eigenvector (Scaled version)

I → Identity matrix

λ → can have more than one values

Use in DS

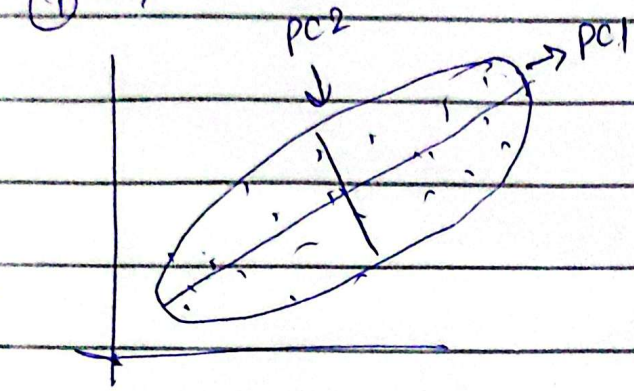
→ Determine a set of comon variables

Dimentionality Reduction

① Principal Component Analysis

② Singular vector Decomposition

① →



$A - \lambda I = 0$

$\lambda = \lambda_1, \lambda_2$

$PC_1 > PC_2$

$V_1 \quad V_2$

Orthogonal
90°

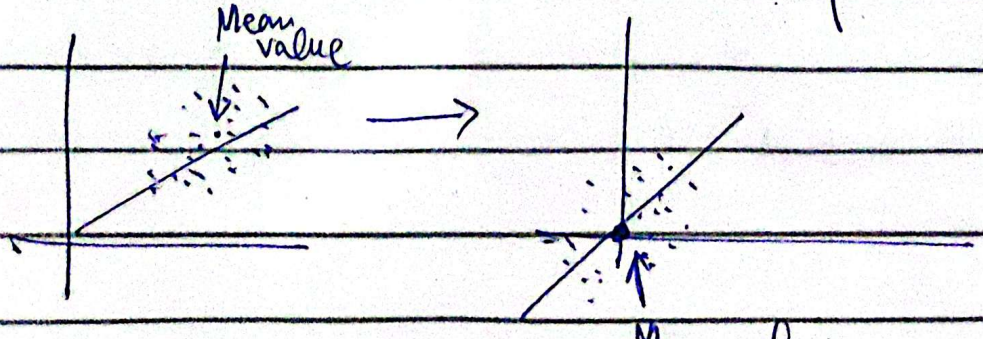
variance	Principal Component
→ each feacture are independance	→ each depends on each other (combine)

Friday SQE 26-4

JMeter - Quiz on Friday

Friday IDS 26-4

PCA ⇒ also calculates co-variance rather than variance of separate features only.



— Move mean value to center (0,0) for 2- Distribution

→ U cannot apply PCA directly / without standardization

→ Mathematically :

Example : Apply PCA to find 2-3 imp vol out of 4.

i) find mean & std of each feature col.

ii) find x_{new} $\quad x_{new} = \frac{x - u}{\sigma}$

iii) find variance

$$\sigma^2 = \frac{\Sigma (x - \mu)^2}{N}$$

sort eign values to find top k eign. value

iv) find co-variance

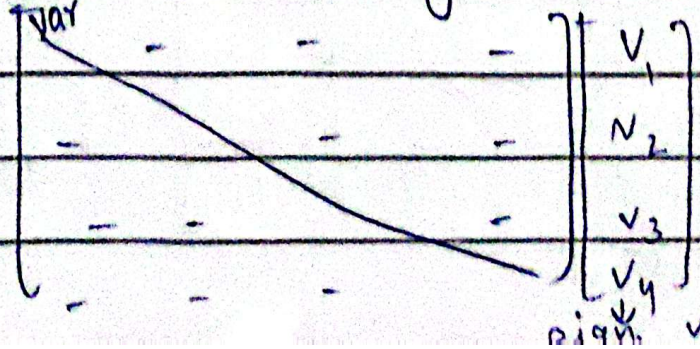
$$COV(X,Y) = \frac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{n-1}$$

& n = no. of obser.

v) calculate cov matrix

	f_1	f_2	f_3	f_4
f_1	V_{f_1}	cov	cov	cov
f_2	cov	V_{f_2}	cov	cov
f_3	cov	cov	V_{f_3}	cov
f_4	cov	cov	cov	V_{f_4}

vi) Calculate eigen values and eign vect



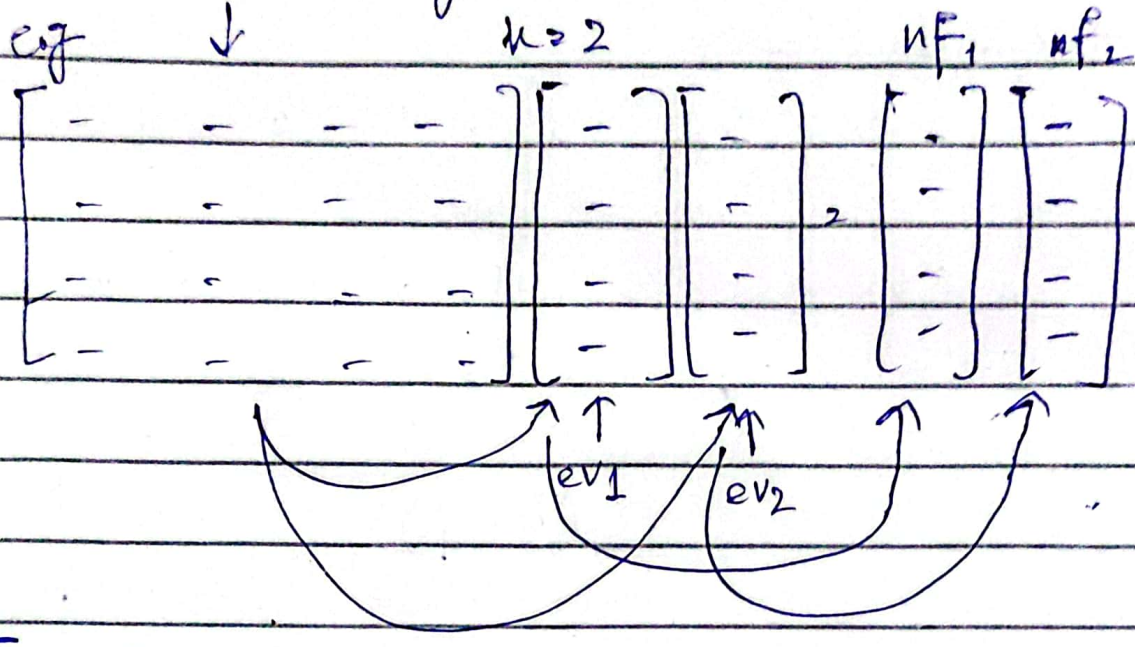
Value of k = no. of columns

standardized form of features

0. - -

no of k

Feature matrix * eign. vector = Tranformed Data



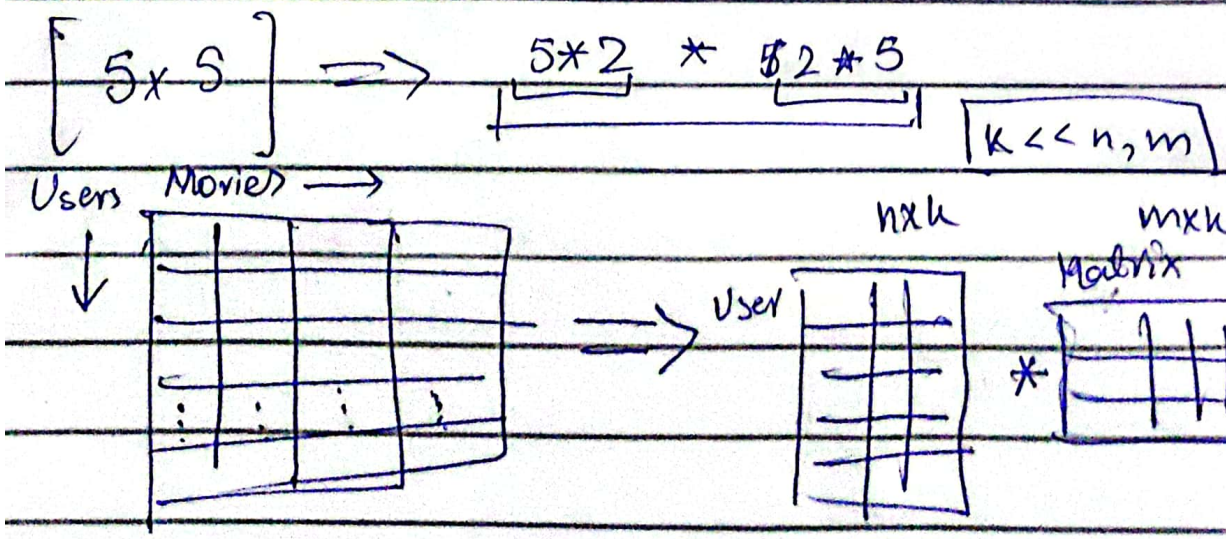
Importance

- Improve accuracy

- less interpratable
- Standardization is necessary

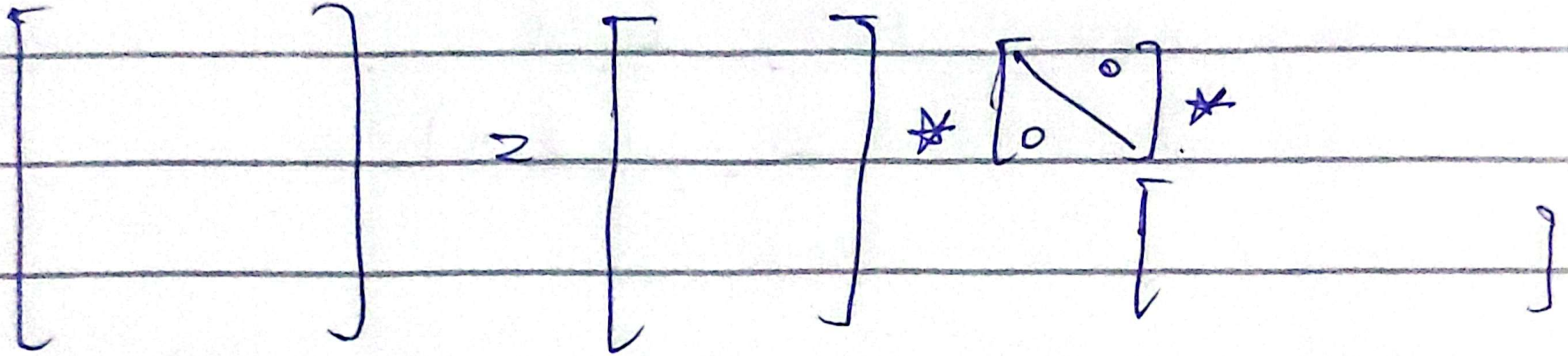
Recommendation System

Matrix Factorization



Singular Value Decomposition (SVD)

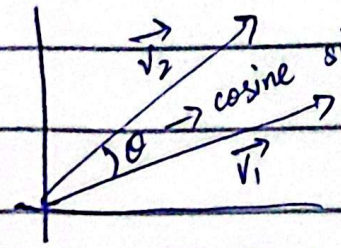
$$R = P \Sigma Q^T$$



Similarity Measure

Object 1 vectors Object 2 vectors

- Cosine Similarity



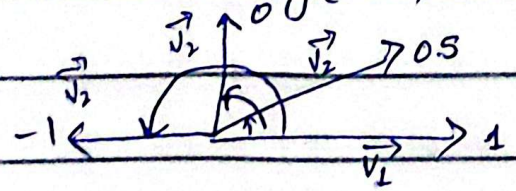
- similarity between 2 vectors in direction
- The angle decides the similarity between vectors

- Convert features into vectors
- Dot Product the vectors

$$\text{similarity}(A,B) = \cos\theta = \frac{A \cdot B}{\|A\| \|B\|}$$

+1 ——— −1

↑ very similar ↑ not similar



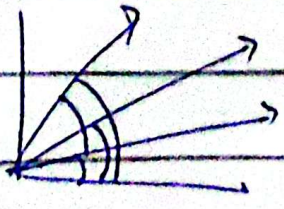
Doc 1: _____ $\Rightarrow \vec{v_1} = [1\ 1\ 0\ 1 \ldots]$

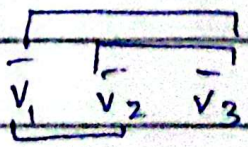
Doc 2: _____ $\vec{v_2} = [1\ 0\ 1 \ldots]$

→ BOW returns the unique words in Doc.

→ words repeating in the doc are shown only one time but their frequency should be in vectorize form

- For 3 and more





Users	Movies		
	M_1	M_2	M_3
U_1	2	3	5
U_2	1	2	5 → For user
U_3	4	3	2

↑ For movie

- Bias

Predict — Actual

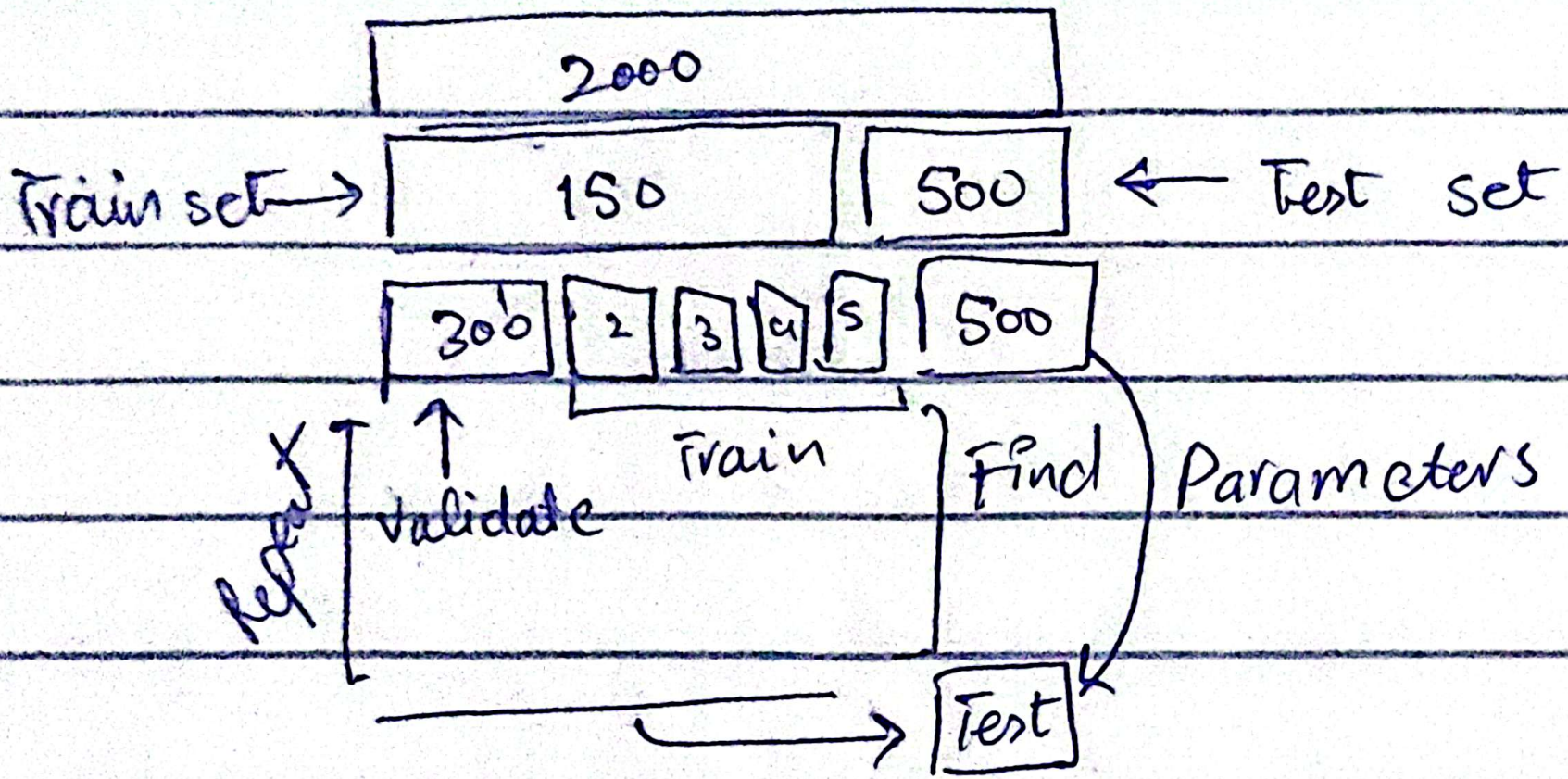
Gap

- Variance = Predicted Values] Spread

• Overfitting	• Underfitting
- Model fail to generalize, instead it specifies the features.	- Model doesn't learn in any way, no feature learnt.
- Model memorize the features	- Model unable to capture features.
- Model performed well on training data but performed poor on ~~train~~ test data.	- Model performed poorly on both training and testing dataset.

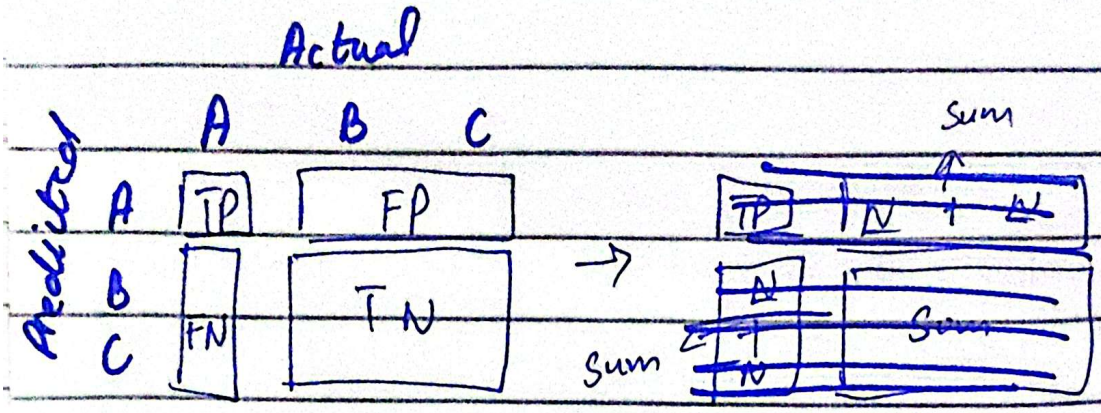
10 - Fold

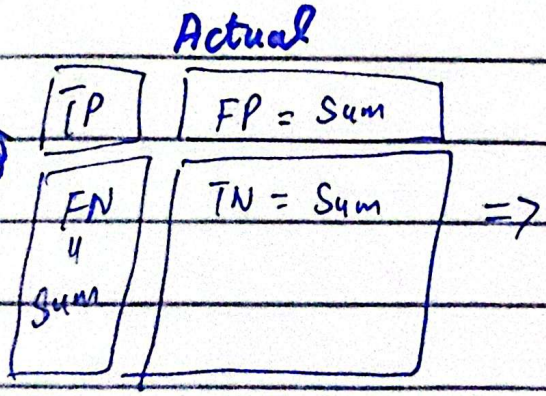
k - Fold Cross - validation



Confusion matrix of multi-class classification

- No. of classes x No. of classes



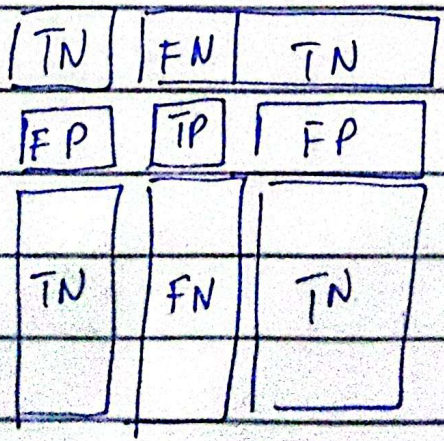


=> 2x2 matrix

→ Multi-class

. < Precision, Recall

> Accuracy

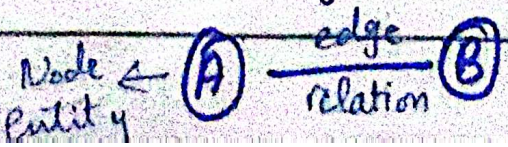


→ Binary-classification

< Accuracy

Graph Data Science

- data presented as graphs
- Social networking e.g.



Centrality Measures in Social Network Analysis

① Degree Centrality

- one hop / jump - find very connected individual

② Betweenness Centrality

$\frac{\sigma st(v)}{\sigma st}$ ← each pair

③ Closeness Centrality

$$CC(i) = \frac{N-1}{\sum d(i,j)}$$

- Distance Matrix

	A	B	C	farness	$\frac{N-1}{\sum d(i,j)}$
A	0			→ sum	$\frac{(3-1)}{sum}$
B		0		→ sum	$\frac{(3-1)}{sum}$
C			0	→ sum	$\frac{(3-1)}{sum}$