Revise :-

1. KNN as a Classification & Regression Algo.

2. Templates use for Classification & Regr Algo.

3. Evaluation Metrics of Classification Algo.

(g) f1 score (a). Precision (d) Confusion Matrix

(b) Recall (e) Error Rate

(c) Accuracy (f) Classification Report

4. KNN on a Regression data.

10 March 2023 19:42

Agenda :-

1. Regression Metrics in python using scikit learn.
2. K-Means Clustering Algorithm.
3. Implementation of K Mean in python using Sklearn.

10 March 2023 19:44

* Regression Evaluation Metrics

1. Mean Absolute Error
2. Mean Squared Error
3. Root Mean Squared Error
4. R^2 Error ✓ ☆
5. Adjusted R^2 Error. ✓ ☆

Very Imp

* Euclidean distance calc from Centroid Point to data points

1st row

C3 :- $(8, 4)$ $(X1, y_1)$:- $(2, 4)$

(x_2, y_2) (x_1, y_1)

ED :-

$$\sqrt{(8-2)^2 + (4-4)^2}$$

$$= \sqrt{6^2+0^2} = \sqrt{36} = \underline{\underline{6}}$$

10 March 2023 20:34

$$C1 :- \overset{(x_1, y_1)}{(2, 4)} \quad \overset{(x_2, y_2)}{(2, 6)} \quad \overset{(x_3, y_3)}{(4, 7)}$$

$$\underline{\underline{CP}} :- \frac{x_1 + x_2 + x_3 + \ldots}{n}, \quad \frac{y_1 + y_2 + y_3 + \ldots}{n}$$

$$= \left(\frac{2+2+4}{3}, \frac{4+6+7}{3} \right)$$

$$\underset{(CP)}{C1} = (2.67, 5.67)$$

Revision :-

1. Dataset looks in a Unsupervised ML Algo (Target is absent)
2. Solved a problem using K-Means Clustering.
3. Define a K-Means Algorithm.
4. K in K-Means represent the number of clusters.
5. Implentation of K-Means in python.

Agenda:-

1. Implementation of K-Means using sklearn in python.
2. Ideal method to select k (no. of clusters)
3. Elbow method.
5. Data preprocessing techniques.

11 March 2023 19:41

Elbow Method :-

$K = 1 \rightarrow C1$

$K = 2 \rightarrow C1, C_2$

$K = 3 \rightarrow C1, C_2, C3$

$K = 4 \rightarrow C1, \ldots\ldots, C4$

$K = 5 \rightarrow C1, \ldots\ldots, C5$

$\vdots$

$K = 20 \rightarrow C1, \ldots\ldots, C20$

WCSS :- Within Cluster Sum of Squares

:- used for judging the optimal K.

Since we have check for each value of k between 1 to 20,

① We have use a for loop

② wcss = [ ]

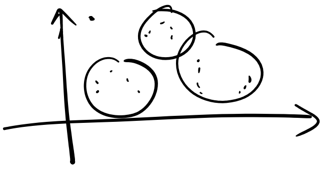
③ loop running

calc wcss and that will be appended to wcss empty list

④ We create a visual to identify the optimal k.

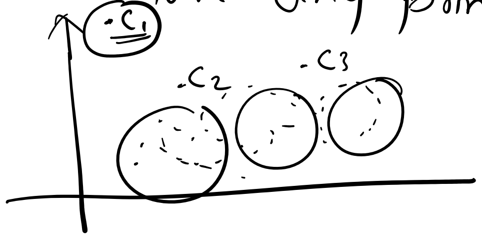
11 March 2023 20:52

init = K-means++



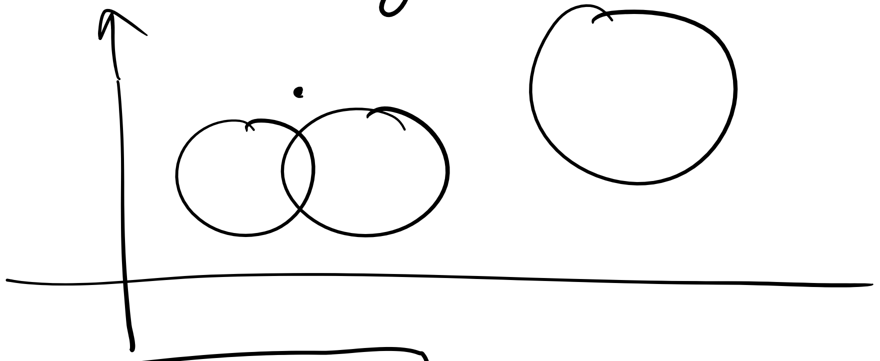
Standard KMeans Algorithm :-

1. Sensitive to the initialization of centroids or the mean points. If the centroid is very far, then that cluster will not have any point associated with it.



11 March 2023 20:56

2. More than 1 cluster might end up linked with single same cluster.



Standard

'init = 'K-means++'

11 March 2023 20:58

K- means++

* K- means++ is the standard K-Means Algorithm coupled with smart initialization of centroid points.

11 March 2023 20:59

*random_ state = 42

* Smply used to make the predictions static

$K=1$

$\boxed{133} \xrightarrow{15} 132.5$ or 133.5 or 134

11 March 2023 21:08

ML Algorithms :-

1. You cannot supply a missing data

2. If any column (variable) is having higher values then ML Algo will give importance to that.

3. ML algo do not understand text data.

   eg :- E-ID E-Name × Quali ✓ Age Salary Yrs. Exp.

   Numerical

13 March 2023 19:36

Revision :-

1. Implementation of K-Means Algorithm in python using sklearn library

2. Elbow Method for optimal (k) no of clusters.

3. Data Preprocessing :-

   a. Missing Value Analysis.

13 March 2023 19:38

Agenda:-

1. Missing Value Treatment techniques.
    1. Mean, Median, Mode, Forward, Backward fill.
    2. Sklearn method for Missing Value imputation.

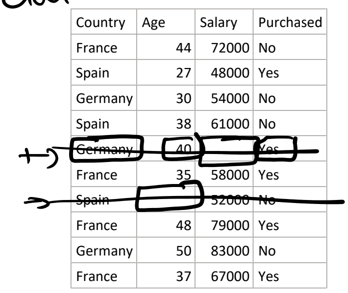
2. Data Encoding Techniques.

13 March 2023 19:40

1) Missing Value Techniques:-

(1) Dropping the rows:-

data =



Country	Age	Salary	Purchased
France	44	72000	No
Spain	27	48000	Yes
Germany	30	54000	No
Spain	38	61000	No
~~Germany~~	~~40~~		~~Yes~~
France	35	58000	Yes
~~Spain~~		~~52000~~	~~No~~
France	48	79000	Yes
Germany	50	83000	No
France	37	67000	Yes

data.dropna()

20%

The entire row gets deleted

(2) Dropping the column :-

data.drop('Age', axis=1)

* Axis: Data is 2-D data.

	Age	Sal	Name	Gend	Qualification.
1 →					
2 →					
3 →	d				
4 →					

```
data.drop('Sal')

data.drop('Sal', axis=1)

data.drop('Gender', axis=1)
```

Country	Age	Salary	Purchased
France	44	72000	No
Spain	27	48000	Yes
Germany	30	54000	No
Spain	38	61000	No
Germany	40	○	Yes
France	35	58000	Yes
Spain	○	52000	No
France	48	79000	Yes
Germany	50	83000	No
France	37	67000	Yes

Age :- Replace all missing with mean of Age.

Salary :- Replace all missing values with mean of Salary

Mode: Most - frequent value in the dataset

eg:- [①Apple, Orange, ②Apple, Mango, Banana, ③Apple]

| Apple - 3 ← Apple is the mode

data

Orange - 1

Mango - 1

Banana - 1

13 March 2023 21:08

① Missing Value Analysis.

② It does not accept Categorical (String) data.

↓

Convert Cate data to Num. data

↓

Data Encoding

13 March 2023 21:10

Data Encoding :- Applicable only on Categorical data

Nominal (No order)

↓

Label Encoding

Ordinal (Specific Order)

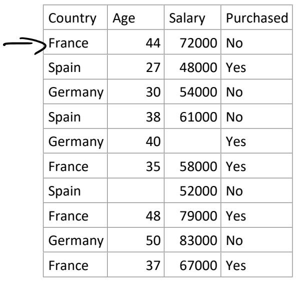
↓

One Hot Encoding

13 March 2023 21:11

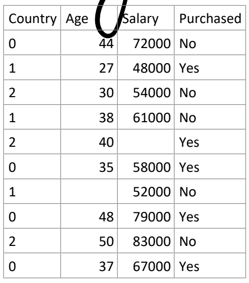
Label Encoding :- For Nomial data.

eg:- Country Column in the Nomial data



Country	Age	Salary	Purchased
France	44	72000	No
Spain	27	48000	Yes
Germany	30	54000	No
Spain	38	61000	No
Germany	40		Yes
France	35	58000	Yes
Spain		52000	No
France	48	79000	Yes
Germany	50	83000	No
France	37	67000	Yes

$\xrightarrow[\text{Enc.}]{\text{Label}}$



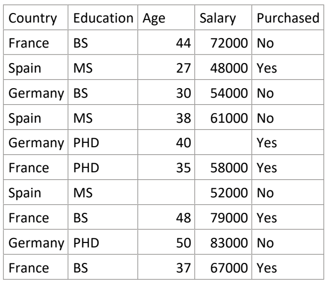
Country	Age	Salary	Purchased
0	44	72000	No
1	27	48000	Yes
2	30	54000	No
1	38	61000	No
2	40		Yes
0	35	58000	Yes
1		52000	No
0	48	79000	Yes
2	50	83000	No
0	37	67000	Yes

France – 0
Spain – 1
Germany – 2

13 March 2023 21:16

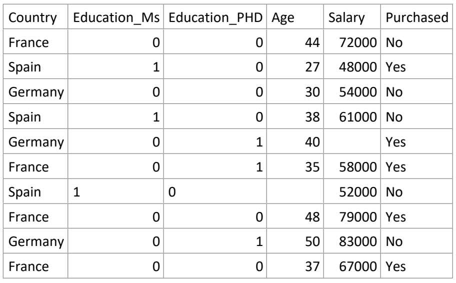
One - Hot - Encoding :- Ordinal data

{ BS - Bachelor
MS - Master
PHD - Doct.



Country	Education	Age	Salary	Purchased
France	BS	44	72000	No
Spain	MS	27	48000	Yes
Germany	BS	30	54000	No
Spain	MS	38	61000	No
Germany	PHD	40		Yes
France	PHD	35	58000	Yes
Spain	MS		52000	No
France	BS	48	79000	Yes
Germany	PHD	50	83000	No
France	BS	37	67000	Yes

OHE →



Country	Education_Ms	Education_PHD	Age	Salary	Purchased
France	0	0	44	72000	No
Spain	1	0	27	48000	Yes
Germany	0	0	30	54000	No
Spain	1	0	38	61000	No
Germany	0	1	40		Yes
France	0	1	35	58000	Yes
Spain	1	0		52000	No
France	0	0	48	79000	Yes
Germany	0	1	50	83000	No
France	0	0	37	67000	Yes

~~Education_BS~~	Education_MS	Education_PHD
~~1~~	0	0
~~0~~	1	0
~~1~~	0	0
~~0~~	1	0
~~0~~	0	1

OHE increases the no. of columns in the data.