Zeal Education Society's

ZEAL COLLEGE OF ENGINEERING AND RESEARCH

Website : www.zealeducation.com Address : Narhe, Pune - 411 041

TABLE OF CONTENTS

Subject : M.L.

Sr. No.	Title	Date	Page No.	Sign	Remark
1).	Data Preparation	23/7/24	1		
2).	Regression Technique	6/8/24	5		
3).	Classification Technique	20/8/24	8		
4).	Clustering Techniques	3/9/24	12		
5).	Association Rule Learning	17/9/24	16		
6).	Multilayered Neural Network Model	1/10/24	20		

ZEAL
INSTITUTES
www.zealeducation.com

Assigment :- 1

Title:

- Data Preparation

Problem:

- Download heart dataset from following link https://www.kaggle.com/zhaoyingzhu/heart.csv Perform the following operation on a given dataset.

a). Find Shape of Data
b). Find Missing values
c). Find data type of each column
d). Finding out zero's
e). Find Mean age of patients.
f). Now extract only Age, Sex, ChestPain, RestBP, Chol, Randomly divide dataset in training (75%) & testing (25%).

Objectives:

- Data preparation is particular to data, the objectives of the projects, and algorithms that'll be used in data modelling techniques. Data Preparation is the process of cleaning and transforming raw data to make predictions accurately through using ML algorithms. Each predictive modeling project with machine learning is

ZEAL
INSTITUTES
www.zealeducation.com

different.

Theory:

- What is Data Preparation:
  - On a predictive modeling project, such as classification or regression, raw data typically cannot be used directly. So, hence we can define that data preparation is one of the important steps in data science domain, in which data collected from multiple sources is cleaned and transformed to improve its quality prior to use in business analytics.
  - As, such the raw data must be pre-processed prior to being used to fit and evaluate a machine learning model.
  - We can define data preparation as the transformation of raw data into a form that is more suitable for modelling.

- Data cleaning:
  - Identifying and correcting mistakes or errors in data

ZEAL
INSTITUTES
www.zealeducation.com

- Feature Selection: Identifying those input variables that're most relevant to the task.
- Data Transforms:
  - Changing the scale or distribution of variables.
- Feature Engineering;
  - Deriving new variables from available data.
- Dimensionality Reduction:
  - Creating compact projections of the data. Each of these is whole field of study with specialise algorithms.

Why do we need Data Processing?

- A real-world data generally contains noises, missing values, and maybe in an unusable format which cannot be directly used for machine learning models.
- It involves below steps:

1). Getting the dataset
2). Importing libraries
3). Important datasets
4). Finding Missing Data
5). Encoding Categorical Data
6). Splitting dataset into training & test set.
7). Feature Scaling.

1). Get the dataset

- The collected data for particular problem in a proper format is known as dataset.
- So each dataset is diff. from another dataset. To use the dataset in our code. we usually put it into a CSV file. However, sometimes, we also use HTML or XLSX.

• .CSV File:

- CSV stands for "Common-Seperated Values" files; it is file format which allows us to save the tabular data, such spreadsheets.

2). Importing Libraries;

- Numpy; Numpy python library is used for including any type of maths operation in code.

- import numpy as nm.
Here, we have used nm, which is short for Numpy, and it will be used in the whole Program.

- Matplotlib: The second library is matplotlib, which is python 2D plotting library and with this sub-library pyplot.
∴ import matplotlib.pyplot as mpt.

ZEAL
INSTITUTES
www.zealeducation.com

- Pandas; The last library is Pandas library, which is one of the most famous Python libraries and used for importanting and managing datasets.

i). Save your Python file in directory which contains dataset.

ii). Go to file explorer option in Syder IDE, & select required factory.

iii). Click on F5 button or run option to execute file.

Confusion Matrix:

- It is matrix of size 2*2 matrix used to describe the performance of a classification model with actual values on one axis & predicted on another.

		Actual	
		Positive	Negative
Predicted	Positive	True Positive	False Positive
	Negative	False Negative	True Negative

- True Positive:

- Model predicted positive and it's true.

- True Negative: models predicted neg. & its true i.e negative in reality.
- False Positive: model predicted positive and its false.
- False Negative: model predicted negative and its false.

$\therefore$ $TPR = \frac{TP}{\text{Actual Posi.}} = \frac{TP}{TP+FN}$

$\therefore$ $FNR = \frac{FN}{\text{Actual Posi.}} = \frac{FN}{TP+FN}$

$\therefore$ $TNR = \frac{TN}{\text{Actual Negative}} = \frac{TN}{TN+FP}$

$\therefore$ $FPR = \frac{FP}{\text{Actual Negative}} = \frac{FP}{TN+FP}$

Calculations using Confusion Matrix:

- <u>Classification Accuracy</u>:

– It is one of important parameters to determine the accuracy of classification problems. It defines how often the model predicts to correct output. It can be calculated as ratio of no. of correct prediction made

$\therefore$ $\boxed{\text{Accuracy}: \frac{TP+TN}{TP+FP+FN+TN}}$

ZEAL
INSTITUTES
www.zealeducation.com

- **Misclassification rate:**
  - It is also termed as Error State, and it defines how often model gives wrong predictions.
  - The formula is:

$$\therefore \text{Error rate} :- \frac{FP + FN}{TP + FP + FN + TN}$$

- **Precision:**
  - It can be defined as no. of correct outputs provided by model or out of all positive classes that have predicted values.

$$\therefore \text{Precision} = \frac{TP}{TP + FP}$$

- **Recall:**
  - It can be defined as no. of correct outputs provided by model or out of all positive classes that have correctly by model.

$$\therefore \text{Precis. / Recall} = \frac{TP}{TP + FN}$$

- F- Measure:
  - If two models have low precision, high recall or vice versa, it is difficult to compare these models. The F-Score is max^m if recall is equal to precision.
  - Formula:

$$\boxed{\text{F-Measure} := \frac{2 * \text{Recall} * \text{Precision}}{\text{Recall} + \text{Precision}}}$$

Null Error Rate:
- How often our model would be incorrect, if it always predicted majority class.

ROC Curve:
- Graph displaying classifiers performance for all possible thresholds.

Conclusion;

→ Most machine learning algorithms require data to be formatted in very specific way, so datasets generally require some amt. of preparation before they can yield useful insights.

21/10/24

In [1]:
```
#Name: Onkar Shirke (T512055)
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [3]:
```
dataFrame=pd.read_csv('heart.csv')
```

In [5]:
```
dataFrame.shape
```

Out[5]: (303, 15)

In [7]:
```
dataFrame.head()
```

Out[7]:

	Unnamed: 0	Age	Sex	ChestPain	RestBP	Chol	Fbs	RestECG	MaxHR	ExAng	O
0	1	63	1	typical	145	233	1	2	150	0	
1	2	67	1	asymptomatic	160	286	0	2	108	1	
2	3	67	1	asymptomatic	120	229	0	2	129	1	
3	4	37	1	nonanginal	130	250	0	0	187	0	
4	5	41	0	nontypical	130	204	0	2	172	0	

In [9]:
```
dataFrame.tail()
```

Out[9]:

	Unnamed: 0	Age	Sex	ChestPain	RestBP	Chol	Fbs	RestECG	MaxHR	ExAng
298	299	45	1	typical	110	264	0	0	132	0
299	300	68	1	asymptomatic	144	193	1	0	141	0
300	301	57	1	asymptomatic	130	131	0	0	115	1
301	302	57	0	nontypical	130	236	0	2	174	0
302	303	38	1	nonanginal	138	175	0	0	173	0

In [11]:
```
dataFrame=dataFrame.drop("Unnamed: 0",axis =1)
```

In [13]:
```
dataFrame.dtypes
```

ZEAL
INSTITUTES
www.zealeducation.com

Assignment: 5

Title:
- Regression Technique

Problem Statement:
- Download temperature data from below link.

This data consists of temperature of INDIA averaging the temperatures of all places month wise. Temperature values are recorded in CELSIUS.

a). Apply linear Regression using suitable library function & predic Month-wise temp.

b). Visualize simple regression model.

Objective:
- In order to predict value of dependent variable for individual effect of some explanatory variable on dependent variable.

Theory:

i). Introduction to Regression:

- It is another important and broadly used Statistical and machine learning tool. The key objective of regression-based tasks is to predict output labels or responses which are continuous numeric values for the input data.

- Linear Regression:
  - It is analysis is used to predict the value of variable based on value of another variable. Also called independent variable.
- Types:

i) Simple Linear Regression:
- Simple linear algorithms have one dependent and independent variable; their relationship is a deterministic relationship if one variable can be expressed accurately by another.

eqn: $\boxed{y = a_0 + a_1 X + \varepsilon}$

ii) Multiple linear Regression:
- In this type of linear regression, we always attempt to discover the relationship between two or more independent variables or input and corresponding dependent variable or output.

iii) MLR eqn:
- In Multiple linear eqn the target variable (Y) is a linear combination of multiple predictor variables $x_1, x_2, x_3, \ldots x_n$.

ZEAL
INSTITUTES
www.zealeducation.com

Since, it is Simple linear Regression, so same is applied for multiple linear regression eqn.

$$\boxed{Y = b_0 + b_1 x_1 + b_2 x_2 + b_3 x_3 + \dots b_n x_n}$$

Implementations of Multipic linear Regression model using Python:

- To implement MLR using python, we have: MLR is dependent variable and other independent variable.
  i). Data Pre-Processing steps
  ii). fitting MLR model to train State.
  iii). Predicting result of test set Application

Application of ML regression algorithms 're as:

- Forecasting or predictive analysis
- Optimization
- Error correction
- Economics
- Finance
- Effectiveness of Independent prediction.
- Predicting impact of changes.

i). Mean Absolute Error:

- MAE is a very simple metric which calculates the absolute difference between

predicted values.

$$\therefore MAE = \frac{1}{N} \sum |y - y|$$

2). Mean squared Error (MSE);

- MSE is most used and very simple metric with a little bit of change in mean absolute error or Mean Squarred error States that finding squared diff. btw. actual & predicted value, it represent sq. dist btw. actual & predicted values.

$$\therefore MSE = \frac{1}{n} \sum (y - \hat{y})^2$$

- Advantage:
- The graph of MSE is differentiable, you can easily loss function.

- Disadvantage:
- If you have outliers in dataset then it penalizes the outliers most and calculated MSE is bigger.

3). Root Mean Squared Error (RMSE);

- As RMSE is clear by the name itself, that

ZEAL
INSTITUTES
www.zealeducation.com

is a simple square root of mean squared error.

$$\therefore RMSE = \sqrt{\frac{1}{n} \sum_{j=1}^{n} (y_j - \hat{y}_j)}$$

- for performing RMSE we have to Numpy Numpy square root function over MSE,

```
print ("RMSE", np.sqrt(mean_squared_error(y_test
                                  - pred)))
```

- Advantage:
- Output value you get in same unit as required output variable which makes interpretation of loss easy.
- Disadvantage:
- It is not that robust to outliers as compared to MAE.

4). Root Mean Squared Long Error (RMSLE);

- Taking the log of RMSE metric slows down the scale of error. The metric is very helpful when you are developing a model without caring the inputs.
- To control this situation of RMSE we take log of calculated RMSE error and resultant we get

as RMSLE
print ("RMSE", np.log (np_sqrt (mean_Squared_error(y_test, y_pred)))

- It is very simple metric i.e. used by most of datasets hosted for M.L competitions.

5). R-Squared (R_2):

- R2 Score is a metric that tells performance of your model, not the loss in an absolute sense that how many wells did your model perform.

- In contrast, MAE and MSE depend on context as we have seen R2 score is independent of context.

- So, with help of R squared we've baseline model to compare a model which none of the other metrics provide.

Hence, R2 Squared is also known as Coefficient of Determination or Sometimes also known as Goodness of fit

ZEAL
INSTITUTES
www.zealeducation.com

$$R2 = \text{Square} = 1 - \frac{SSr}{SSm}$$

```
from sklearn.metrics import r2_score
r2 = r2_score(y_test, y_pred)
print (r2)
```

e). Adjusted R- Squared:

- The disadvantage of R2 Score is while adding new features in data R2 score starts increasing or remains const. but it never decreases because, It assumes that while adding more data variance of the data increases.

- Hence, To control the situation Adjusted R Squared came into existence.

$$R_a^2 = 1 - \left| \left(\frac{n-1}{n-k-1} \right) \times (1-R^2) \right|$$

where,

n = no. of observations

k = no. of independent variables.

R_a^2 = adjusted R^2

n = 40

k = 2

adj-r2 Score = 1 - ((1 - r²) * (n-1)(n-k-1))

Conclusion;

- The system integrates the task of developing a regression model from data, with the technique of searching for logical conditions that enable a better fitting error by model. It continues to be significant asset to many leading sectors starting from finance, education, banking, meteri, medicine, media. etc.

21/1/24

In [1]:
```
#Name: Onkar Shirke (T512055)
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
import seaborn as sns
%matplotlib inline
from sklearn.metrics import mean_squared_error,mean_absolute_error,r2_score
```

In [3]:
```
temp_dataset = pd.read_csv('temperatures.csv')
temp_dataset
```

Out[3]:

	YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	
0	1901	22.40	24.14	29.07	31.91	33.41	33.18	31.21	30.39	30.47	29.97	27.31	2
1	1902	24.93	26.58	29.77	31.78	33.73	32.91	30.92	30.73	29.80	29.12	26.31	2
2	1903	23.44	25.03	27.83	31.39	32.91	33.00	31.34	29.98	29.85	29.04	26.08	2
3	1904	22.50	24.73	28.21	32.02	32.64	32.07	30.36	30.09	30.04	29.20	26.36	2
4	1905	22.00	22.83	26.68	30.01	33.32	33.25	31.44	30.68	30.12	30.67	27.52	2
...	...	...	...	...	...	...	...	...	...	...	...	...	
112	2013	24.56	26.59	30.62	32.66	34.46	32.44	31.07	30.76	31.04	30.27	27.83	2
113	2014	23.83	25.97	28.95	32.74	33.77	34.15	31.85	31.32	30.68	30.29	28.05	2
114	2015	24.58	26.89	29.07	31.87	34.09	32.48	31.88	31.52	31.55	31.04	28.10	2
115	2016	26.94	29.72	32.62	35.38	35.72	34.03	31.64	31.79	31.66	31.98	30.11	2
116	2017	26.45	29.46	31.60	34.95	35.84	33.82	31.88	31.72	32.22	32.29	29.60	2

117 rows × 18 columns

In [5]:
```
temp_dataset.shape
```

Out[5]: (117, 18)

In [7]:
```
temp_dataset.describe().T
```

ZEAL
INSTITUTES
www.zealeducation.com

Assignment:- 3

Aim:
- Classification Technique

Problem Statement:
- Data set Available on kaggle (The last column needs to be changed to 0 or 1). The counsellor of the firm is supposed check whether student will get an admission or not based on his / her GRE Score and Academic Score.

A). Apply Data pre-processing (Label Encoding Data Transwormation .....) Techniques if necessary.
B). Perform data-preparation (Train-Test Split)
C). Apply Machine learning Algorithm.
D). Evaluate Model 4.

Objective:
- The main goal of the classification algorithms is to identify category of given data, these algorithms are mainly used to predict the output for categorical data.

Theory:
- Classification Algorithm:
- The classification algorithm is supervised

learning technique i.e. used to identify the category of new observations on basis of training data.

$\therefore$ $y = f(x)$, where y = Categorical unit

Types of ML Classification Algorithms:

- Linear Models:
  - Logistic Regression
  - Support Vector Machines

- Non-Linear Models:
  - K-Nearest Neighbors
  - Kernel SVM
  - Naive Bayes
  - Decision Tree Classification
  - Random forest Classification.

Decision Tree Classification Algorithm:

- Decision Tree is Supervised Algorithm technique that can be used for both Classification & Regression problems.
- In a Decision Tree, there are two nodes, which are Decision Node and leaf Node.
- Decision nodes are used to make

ZEAL
INSTITUTES
www.zealeducation.com

and have multiple branches, whereas leaf nodes are output of those decisions and do not contain any further branches.

- It is called decision tree, because similar to tree. In order to build a tree, we use CART algorithm, which stands for Classification & Regression Tree algorithm.

Decision Node → Root Node

Decision Node Decision Node

~~Leaf Node~~ leaf Node leaf Node Decision Node

leaf node leaf Node

Decision Trees:

- There are various algorithms in Machine learning, so choosing the best algorithm for the given dataset and problem is main point to remember while creating machine learning model.

- Decision Trees usually mimic human thing ability while making a

decision, so it is easy to understand.

- The logic behind the decision tree can be easily understood because it shows a tree-like structure.

Decision Tree Terminologies;

- Root node: It is from where decision tree starts. It represents entire dataset, which further gets divided.
- Leaf Node: leaf nodes are final output node, and tree cannot be segregated further after getting leaf node.
- Splitting:- Splitting is process of dividing the decision node / root node into sub-nodes according to given conditions.
- Branch / Sub-Tree: A tree formed by splitting tree
- Pruning: Pruning is the process of removing the unwanted branches of tree.
- Parent child node: The root node of the tree is called parent node, and other nodes are called child nodes.

ZEAL
INSTITUTES
www.zealeducation.com

Attribute Selection Measures;

- While implementing Decision tree, the main issue arises that how to select best attribute for root node and for sub-nodes. So, to solve such problems there is technique which is called as Attribute Selection measure or ASM. By this measurment, we can easily select the best attribute for the nodes of tree.
  - Information Gain
  - Gini Index

Information Gain;

- Information gain is measurment of changes in entropy after segmentation of dataset based on attribute.
- It calculates how much information a feature provide us about a class.
- According to the value of information gain, we split the node and build the decision tree.
- A decision tree algorithm always tries to maximize the value of information gain, and node / attribute having the highest information gain

node / attribute having higher info. gain is split first.

Information Gain = Entropy(S) - [(Weighted Avg) * Entropy (each feature)]

Entropy;

- It is a metric to measure the impurity in a given attribute. It specifies randomness in data. Entropy can be calculated as;

$\therefore$ Entropy (S) = $-P(yes)\log_2 P(yes) - P(no)\log_2 P(no)$

Gini Index:

- Gini Index is a measure of impurity or purity used while creating a decision tree in CART (Classification and Regression Tree) Algorithm.

- An attribute with low Gini Index should be preffered as compared to the high Gini Index.

- Gini Index can be calculated using the below formula; $\boxed{\text{Gini Index} = 1 - \Sigma_i P_i^2}$

ZEAL
INSTITUTES
www.zealeducation.com

Prunning : Getting an Optimal Decision Tree ;

- Prunning is process of deleting the unnecessary node from a tree in order to get the optimal decision tree.
- Therefore, a technique that decreases the size of learning tree without reducing accuracy is known as Prunning. There are mainly two types of tree prunning technology used.

  - Cost complexity Prunning
  - Reduced Error Prunning.

Advantages of Decision Tree ;

- It is simple to understand as it follows same process which a human follow while making any decision any decision in real life.
- It can be very useful for solving decision-related problems.
- It helps to think about all the possible outcomes for a problem.

Disadvantages of Decision Tree;

- The decision tree contains lots of layers, which makes it complex.
- It may have an overfitting issue, which can be resolved used in Random forest algorithm.

Python Implementation of Decision Tree:

- Data Preprocessing Step
- Fitting Decision Tree algorithm to the Training set.
- Predicting the test result
- Test accuracy of the results (creation of Confusion matrix).
- Visualizing test set result.

Conclusion;

- A classification is method of predicting similar information from categorical or numerical datasets. Nowadays machine learning algorithms are becoming the more popular for classification problems. It gives an an introduction to most of popular machine learning algorithms used for classification of pattern recognition.

21/10/21

In [2]:
```
#Name: Onkar Shirke (TS12055)

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [4]:
```
#Loading data into dataframe
data = pd.read_csv("Admission_Predict.csv")
```

In [6]:
```
data.head()
```

Out[6]:

	Serial No.	GRE Score	TOEFL Score	University Rating	SOP	LOR	CGPA	Research	Chance of Admit
0	1	337	118	4	4.5	4.5	9.65	1	0.92
1	2	324	107	4	4.0	4.5	8.87	1	0.76
2	3	316	104	3	3.0	3.5	8.00	1	0.72
3	4	322	110	3	3.5	2.5	8.67	1	0.80
4	5	314	103	2	2.0	3.0	8.21	0	0.65

In [8]:
```
data.tail()
```

Out[8]:

	Serial No.	GRE Score	TOEFL Score	University Rating	SOP	LOR	CGPA	Research	Chance of Admit
395	396	324	110	3	3.5	3.5	9.04	1	0.82
396	397	325	107	3	3.0	3.5	9.11	1	0.84
397	398	330	116	4	5.0	4.5	9.45	1	0.91
398	399	312	103	3	3.5	4.0	8.78	0	0.67
399	400	333	117	4	5.0	4.0	9.66	1	0.95

In [10]:
```
data.shape
```

Out[10]:
```
(400, 9)
```

In [12]:
```
data.columns
```

Out[12]:
```
Index(['Serial No.', 'GRE Score', 'TOEFL Score', 'University Rating', 'SOP',
       'LOR ', 'CGPA', 'Research', 'Chance of Admit '],
      dtype='object')
```

In [14]:
```
data.drop("Serial  No.",axis=1,inplace=True)
```

In [16]:
```
data
```

ZEAL
INSTITUTES
www.zealeducation.com

Assignment: 4

Title:
- Clustering Techniques

Problem Statement:
- This dataset gives the data of Income and money spent by customers visiting a Shopping Mall. The data set contains Customer ID, Gender, Age, Annual Income, Spending Score. Therefore, as a mall owner you need to find the group who are the profitable customers for the mall owner. Apply at least two clustering algorithms.

a). Apply ~~Data~~ Pre-Processing (Label Encoding, Data Transmission....) techniques if necessary.
b). Perform data-preparation (Train-Test Split).
c). Apply Machine learning Algorithm.
d). Evaluate model
e) Apply cross-validation & Evaluation Model

Objective:
- The goal of Clustering is to find distinct groups or "Clusters" within a data set. Using a machine language

algorithm, the tool creates groups where items in similar group, will in general have similar characteristic to each other in. In K-Means, each cluster is associated with a centroid. The main objective of K-Means algorithm is to minimize of distances between the points and their respective cluster centroid.

Theory:

- Introducing to Clustering:
  - It is basically type of unsupervised learning method.
  - Generally it is used as a process to find data sets consisting of input data without labeled responses. Generally, it is used as a process to find meaningful structure, explanatory underlying processes, generative features and groupings.

- Clustering:
  - It is task of dividing population or data points into a no. of groups

ZEAL
INSTITUTES
www.zealeducation.com

- It is basically a collection of objects on basis of similarity & dissimilarity between them.

Clustering Methods:

- To cluster your data you'll follow these steps.

1) Prepare data
2). Create similarity topic
3) Run clustering algorithm
4) Interpret results and adjust your clustering.

- Density Based Methods:
  - These methods consider the clusters as the dense region having some similarities and differences from the lower dense region of the space

- Hierarchial Based Methods:
  - The clusters formed in this method from structure based on hierarchy
    - Agglomerative (bottom up)
    - Divisive (top down approach)

- Partitioning Methods:

Zeal Education Institutes

- These methods partition of objects into K clusters and each partition forms one cluster

- Grid - Based Methods:
  - In this method, the data space is formulated into a finite no. of cells that form a grid - like structure.

Workflow:

- Prepare Data
- Create Similarity Metric
- Run Clustering Algorithm:
  - It uses the similarity metric to cluster data. focuses on K-Means.
- Interpret Results and Adjust:
  - Checking quality of your clustering output is iterative & exploratory because clustering lacks "truth" can verify output.
- Finding K Value:
  - Elbow Method, Silhouette Method

ZEAL
INSTITUTES
www.zealeducation.com

Elbow Method:

> It is used to find K for K-Means. For this, we have learn Something known as WSS (Within Sum of squares).

> WSS: It is defined as sum of squared dist. between each member of cluster & its centroid.

$$\therefore WSS = \sum_{i=1}^{M} d(p_i, q(i))^2 = \sum_{i=1}^{m} \sum_{j=1}^{n} (p_q - q_j^{(i)})^2$$

Silhoette Method:

> Silhouette Method: Here in silhoette Method, we will compute silhoette score for every point.

> Silhouette Coefficient for the point $\frac{(= b - a)}{max(a, b)}$

Run the Clustering Algorithm:

- K-means Clustering Algorithm:

> K-means Clustering is an unsupervised algo. learning algorithm that is used

to solve the clustering problems in machine learning or data science.

> The algorithm groups unlabled dataset into different clusters.

> The K-Means algorithm defines the no. of predefined clusters that need to be created in process i.e. for $K = 2$.

Here, the dataset which divided belongs to only one group that has similar properties.

> To cluster data into K-clusters, K-means follows the steps below: 1, 2, 3 & 4.

> Then, check these commonly used metrics as described in following sections:

- Cluster Cardinality
- Cluster Magnitude
- Performance of Downstream System.

1). <u>Cluster Cardinality</u>:

> It is no. of examples per cluster. Plot the cluster cardinality for all clusters & investigate clusters that are major outliers.

ZEAL
INSTITUTES
www.zealeducation.com

eg:- Reference of clusters.

2). Cluster Magnitude:

> It is Sum of distances from all examples to the centroid of cluster. Similar to cardinality, Check how magnitude varies across clusters & investigate anamalies.

3). Magnitude vs Cardinality:

> Notice that higher cluster cardinality tends to result in higher cluster magnitude, which intuitively makes sense.

> Clusters are anamalous when cardinality does int corelate with magnitude relative to other clusters. find anamalous clusters by plotting magnitude against cardinality.

K-Means :

1). Advantages;

- Relatively Simple to implement
- Scales to large data sets.
- Guarantee convergence
- Can - warm - start positions of centroids

Easily adapts to new examples.

- Generalizes to clusters of different shapes & sizes, such as elliptical clusters.

2). <u>Disadvantages:</u>

1). Choosing K - mannually.

2). Being independent on initial values.

> Also involves (k - seeding)

3). Clustering Outliers.

> Centroids are dragged by outliers instead of being ignored.

4). Scaling with no. of dimensions

<u>Conclusion:</u>

— We have learned about K.Means Clustering algorithm from scraten to implementation. Firstly we have seen clustering finding K-value in K.Means. Overall, we have seen importance of K Means Clustering algorithm. Finally, we had implemented code for algorithm in jupyter notebook.

21/10/21

In [2]:
```
#Name: Onkar Shirke (T512055)
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings('ignore')
```

In [4]:
```
df = pd.read_csv("Mall_Customers.csv")
```

In [6]:
```
df.head()
```

Out[6]:

	CustomerID	Genre	Age	Annual Income (k$)	Spending Score (1-100)
0	1	Male	19	15	39
1	2	Male	21	15	81
2	3	Female	20	16	6
3	4	Female	23	16	77
4	5	Female	31	17	40

In [8]:
```
df.tail()
```

Out[8]:

	CustomerID	Genre	Age	Annual Income (k$)	Spending Score (1-100)
195	196	Female	35	120	79
196	197	Female	45	126	28
197	198	Male	32	126	74
198	199	Male	32	137	18
199	200	Male	30	137	83

In [10]:
```
df.shape
```

Out[10]:
```
(200, 5)
```

In [12]:
```
df.columns
```

Out[12]:
```
Index(['CustomerID', 'Genre', 'Age', 'Annual Income (k$)',
       'Spending Score (1-100)'],
      dtype='object')
```

In [14]:
```
df.drop("CustomerID",axis=1,inplace=True)
```

In [16]:
```
df
```

ZEAL
INSTITUTES
www.zealeducation.com

Assignment No. 5.

Title :- Association Rule Learning

Problem statement :- Download Market Basket optimization dataset from below link :- https://www.kaggle.com/hemanthkumar05/market-basket-optimization. It contains 7501 transactions. Find association rules between item.
header= None
a) Following steps b) data processing c) Generate list of transactions d) Train algorithm e) Generated rules depend on values of hyper parameters.

Objective :- Association Rule learning is a rule based ML method Intended to to identify strong rules

Theory

Association Rule Learning

It's a type of unsupervised learning technique that checks for the dependency of one data item on another data item and maps accordingly so that it can be more profitable.

The association rule learning has web usage mining, continuous production etc.

How does Association Rule Learning Work?

It works on If & Else statement. Here If called antencedent & else called consequent. These types of relationships where we can find out some association or correlation between 2 items is called single cardinality. Metrics :-

- support
- confidence
- Lift.

Support :- It's the frequency of A. It defined as fraction of the transaction T that contains the itemset X.

$$Supp(X) = \frac{Freq(X)}{T}$$

Confidence :- It is the ratio of the transaction that contains X and Y to the number of records that contain X.

$$confidence = \frac{Freq(X,Y)}{Freq(X)}$$

Lift :- It's the strength of any rule which can be defined as :-

$$Lift \frac{Supp(X,Y)}{Supp(X) \times Supp(Y)}$$

Types of Association Learning

3 algorithms ⇒ 1) Apriori 2) Eclat 3) F-P Growth.

Apriori Algorithm

It uses frequent datasets to generate association rules. It's designed to work on the databases that contain transactions. It uses frequent itemsets to generate association rules.

What is Frequent itemset ?

The items whose support is greater than the threshold value or user-specified minimum support.

Steps for Apriori Algorithm

Suppose we've following dataset.

TID	ITEMSET
T1	A, B
T2	B, D
T3	B, C
T4	A, B, D
T5	A, C
T6	B, C
T7	A, C

TID	Itemset
T8	A, B, C, E
T9	A, B, C

ZEAL
INSTITUTES
www.zealeducation.com

Itemset	Support_count
A	6
B	7
C	5
D	2
E	1

step 2 :- candidate Generation C2 & L2

Itemset	support_count
{A, B}	4
{A, C}	4
{A, D}	1
{B, C}	4
{B, D}	2
{C, D}	0

Create L3 table, There is only one combination of itemset that has support count equal to the minimum support count. so, the L3 will have only one combination {A, B, C}

step 3 :- candidate Generation C3 & L3
We'll create the pair of the itemsets of L1 in the form of subsets.

Itemset	Support_Count
{A, B, C}	2
{B, C, D}	1
{A, C, D}	0
{A, B, D}	0

There is only one combination of itemset that has support count equal to minimum support count. So, the L3 will've only one combination {A, B, C}

Step4: Finding the association rules of the subset

To generate the association rules, first we will create a new table with the possible rules from the occured combination {A, B, C}.

For all rules, we will calculate the confidence using formula sup (A∪B)/A.

After calculating the confidence value of rules,

we will exclude the rules that have less confidence than the minimum threshold (50%)

ZEAL
INSTITUTES
www.zealeducation.com

Advantages of Apriori Algorithm

- Easy to understand
- The join & prune steps of the algorithm can be easily implemented on large datasets.

Disadvantages

- Overall performance can be reduced as it scans database
- Works slow as compared to other algorithms
- Time complexity is $O(2^D)$ very high. Here, D is horizontal width present in database.

Python Implementation of Apriori Algorithm

Now, we'll see practical implementation. To implement this, we've a problem of retailer, who wants to find the association between his shop's product, so that he can provide an offer of "Buy this & Get that" to his customers.

The retailer has a dataset information that contains a list of transactions made by his customers. In the dataset, each row show

steps:-
- Data pre processing
- Training the Apriori model on dataset
- visualising the result.

Data Pre processing steps

- Import libraries

pip install apyroi
Below is code to implement libraries that will be used for different tasks

```
import numpy as nm
import matplotlib.pyplot as mtp
import pandas as pd.
```

- Importing

All rows of dataset are showing different transactions made by the customers.
The first row is the transaction done by the first customer, which means there is no particular name for each column & have their own individual value or product details. So, we need to mention here in our code that there is no particular name.

ZEAL
INSTITUTES
www.zealeducation.com

```
dataset = pd.read_csv('Market_Basket_data1.csv')
transactions = []
for i in range (0, 7501):
    transactions.append ([str (dataset.values[i,j])
    for j in range (0,20)])
```

- Training Apriori Model on dataset

```
from apyori import apriori
rules = apriori (transactions = transactions,
min_support = 0.003, min_confidence = 0.2,
min_lift = 3, min_length = 2, max_length=2)
```

- Visualizing the result

Now, we'll visualize the output for our apriori model. Here, we will follow some more steps,

- Displaying the result of the rules occurred from apriori function.

```
results = list (rules)
results.
```

Applications of Association Rule learning

- Market Basket Analysis.
- Medical Diagnosis
- Protein sequence.

conclusion

Association Rule Mining collects Interesting Associations And correlation Relationships Among Large sets of Data Items. The Association Rules show Attribute Value Conditions That occur Frequently Together.

In A Given Data set, A simple Ex. of Association Rule Mining is Market Basket Analysis.

21/10/24

In [1]:
```
#Name: Onkar Shirke (T512055)
import numpy as np
import pandas as pd

import os
for dirname, _, filenames in os.walk('/kaggle/input'):
    for filename in filenames:
        print(os.path.join(dirname,  filename))
```

In [3]:
```
!pip install mlxtend
```

```
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: mlxtend in c:\users\student\appdata\roaming\python
\python312\site-packages (0.23.1)
Requirement already satisfied: scipy>=1.2.1 in c:\programdata\anaconda3\lib\site-
packages (from mlxtend) (1.13.1)
Requirement already satisfied: numpy>=1.16.2 in c:\programdata\anaconda3\lib\site
-packages (from mlxtend) (1.26.4)
Requirement already satisfied: pandas>=0.24.2 in c:\programdata\anaconda3\lib\sit
e-packages (from mlxtend) (2.2.2)
Requirement already satisfied: scikit-learn>=1.0.2 in c:\programdata\anaconda3\li
b\site-packages (from mlxtend) (1.4.2)
Requirement already satisfied: matplotlib>=3.0.0 in c:\programdata\anaconda3\lib
\site-packages (from mlxtend) (3.8.4)
Requirement already satisfied: joblib>=0.13.2 in c:\programdata\anaconda3\lib\sit
e-packages (from mlxtend) (1.4.2)
Requirement already satisfied: contourpy>=1.0.1 in c:\programdata\anaconda3\lib\s
ite-packages (from matplotlib>=3.0.0->mlxtend) (1.2.0)
Requirement already satisfied: cycler>=0.10 in c:\programdata\anaconda3\lib\site-
packages (from matplotlib>=3.0.0->mlxtend) (0.11.0)
Requirement already satisfied: fonttools>=4.22.0 in c:\programdata\anaconda3\lib
\site-packages (from matplotlib>=3.0.0->mlxtend) (4.51.0)
Requirement already satisfied: kiwisolver>=1.3.1 in c:\programdata\anaconda3\lib
\site-packages (from matplotlib>=3.0.0->mlxtend) (1.4.4)
Requirement already satisfied: packaging>=20.0 in c:\programdata\anaconda3\lib\si
te-packages (from matplotlib>=3.0.0->mlxtend) (23.2)
Requirement already satisfied: pillow>=8 in c:\programdata\anaconda3\lib\site-pac
kages (from matplotlib>=3.0.0->mlxtend) (10.4.0)
Requirement already satisfied: pyparsing>=2.3.1 in c:\programdata\anaconda3\lib\s
ite-packages (from matplotlib>=3.0.0->mlxtend) (3.0.9)
Requirement already satisfied: python-dateutil>=2.7 in c:\programdata\anaconda3\l
ib\site-packages (from matplotlib>=3.0.0->mlxtend) (2.9.0.post0)
Requirement already satisfied: pytz>=2020.1 in c:\programdata\anaconda3\lib\site-
packages (from pandas>=0.24.2->mlxtend) (2024.1)
Requirement already satisfied: tzdata>=2022.7 in c:\programdata\anaconda3\lib\sit
e-packages (from pandas>=0.24.2->mlxtend) (2023.3)
Requirement already satisfied: threadpoolctl>=2.0.0 in c:\programdata\anaconda3\l
ib\site-packages (from scikit-learn>=1.0.2->mlxtend) (2.2.0)
Requirement already satisfied: six>=1.5 in c:\programdata\anaconda3\lib\site-pack
ages (from python-dateutil>=2.7->matplotlib>=3.0.0->mlxtend) (1.16.0)
```

In [5]:
```
import csv
from mlxtend.preprocessing import TransactionEncoder
from mlxtend.frequent_patterns import apriori, association_rules
```

In [7]:
```
data = []
with open('Market_Basket_Optimisation.csv') as file:
    reader = csv.reader(file, delimiter=',')
```

ZEAL
INSTITUTES
www.zealeducation.com

Assignment No. 6.

Title :- Multilayer Neural Network Model

Problem Statement

Download the dataset of National Institute of Diabetes & Digestive & Kidney Diseases from below link:-
https://raw.githubusercontent.com/jbrownlee/Datasets/master/pima-indians-diabetes.data.csv

a) Load dataset in program. Define ANN model with Keras.
b) Compile the model with parameters.
c) Evaluate performance of model.
d) Evaluate model performance using different activation functions.

Theory

Neural Networks are parallel computing devices, which is basically an attempt to make a computer model of brain.

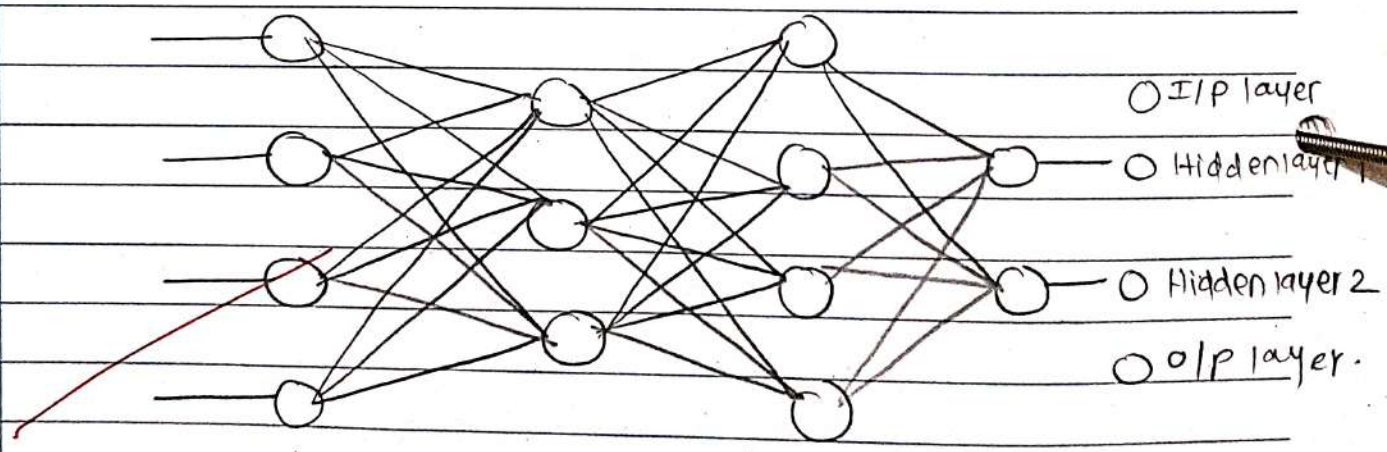
Artificial Neural Network

ANN is an efficient computing system whose central theme is borrowed from the analogy of biological neural networks.

ANNs are also named as "artificial neural systems" or "parallel distributed processing systems" or "connectionist system".

Architecture of ANN

ANN consists of large number of artificial neurons, which are termed units arranged in a sequence of layers.



Input layer

It accepts inputs in several different formats provided by the programmer.

Hidden layer

It presents in between input & output layers. It performs all the calculations to find hidden features & patterns

Output layer

The input goes through a series of transformations using the hidden layer

It finally results in output that is conveyed using this layer.

ANN's computation is represented as :-

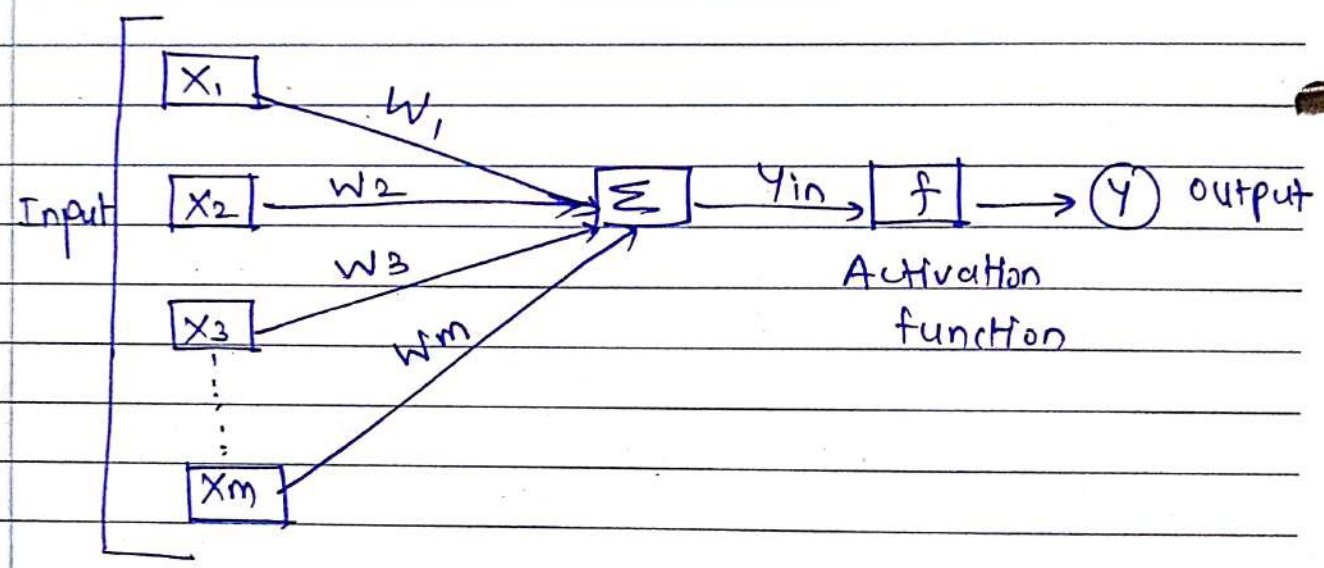
$$\boxed{\sum_{i=1}^{n} W_i * X_i + b}$$

It determines the weighted total is passed as an input to produce the output.

There are distinctive activation functions available that can be applied upon the sort of tasks we're performing.

Model of an ANN

The following diagram represents the general model of ANN followed by its processing.



For above, general model of artificial neural network, the net input can be calculated as follows:-

$$Y_{in} = x1.w1 + x2.w2 + x3.w3 \ldots\ldots xm.wm$$

$$Y_{in} = x1.w1 + x2.w2 + x3.w3 \ldots xm.wm$$ i.e. Net input.

ZEAL
INSTITUTES
www.zealeducation.com

$y_{in} = \Sigma m_i x_i \cdot w_i$

$y_{in} = \Sigma i m x_i \cdot w_i$

The output can be calculated by applying the activation function over the net input.

$Y = F(y_{in})$

Output = function net input calculated

Feed Forward ANN

A feed forward network is a basic neural networks consisting of an input layer, an output layer, & at least one layer of a neuron.

Though assessment of its output by reviewing its input, the intensity of the network can be noticed based on group behavior of the associated neurons & the output is decided.

The primary advantage of this network is that it figures out how to evaluate & recognize input patterns.

Advantage of Artificial Neural Network

- **Parallel Processing Capability**

ANN have a neumarical value that can perform more than one task simultaneously.

- **storing data on the entire network**

After Data that is used in traditional programming is stored on the whole network, not on a database.

The disappearance of a couple of pieces of data in one place doesn't prevent the network from working.

- **capabillity to work with incomplete knowledge**

After ANN training, the information may produce output even with inadequate data.

- **Having a memory distribution**

- **Having Fault Tolerence**

Extortion doesn't prohibit ANN from generating output

ZEAL
INSTITUTES
www.zealeducation.com

Disadvantages of ANN

- **Assurance of proper network structure**

There is no particular guideline for determining the structure of artificial neural networks. The appropriate network structure is accomplished through experience, trial & error.

- **Unrecognized behavior of network**

It's the most significant issue of ANN. When ANN produces a testing solution, it doesnt provide insight concerning why & how.

It decreases trust in the network

- **Hardware dependence**

Artificial neural networks need processors with parallel processing power, as per their structure. Therefore, the realization of the equipment is dependent.

- **Difficulty of showing the issue to network**

ANN can work with numerical data. Problems must be converted into numerical values.

- The duration of network is unknown

The network is reduced to a specific value of error, & this doesn't give us optimum results.

Conclusion

Multilayer perceptrons are the most commonly used types of neural networks. Using the backpropagation algorithm for training, they can be used for a wide range of applications. from the functional approximation to prediction in various fields, such as estimating the load of a calculating system or modeling the evaluating of chemical reactions of polymerization, described by complex systems of differential equations.

21/12/24

In [1]:
```
#Name: Onkar Shirke (T512055)
```

In [2]:
```
import math
import numpy as np
import pandas as pd
import tensorflow as tf
import matplotlib.pyplot as plt
from tensorflow.keras import Model
from tensorflow.keras import Sequential
from tensorflow.keras.optimizers import Adam
from sklearn.preprocessing import StandardScaler
from tensorflow.keras.layers import Dense, Dropout
from sklearn.model_selection import train_test_split
from tensorflow.keras.losses import MeanSquaredLogarithmicError
```

In [3]:
```
df  =  np.loadtxt('https://raw.githubusercontent.com/jbrownlee/Datasets/master/pim
```

In [4]:
```
df
```

Out[4]:
```
array([[  6.   , 148.   ,  72.   , ...,   0.627,  50.   ,   1.   ],
       [  1.   ,  85.   ,  66.   , ...,   0.351,  31.   ,   0.   ],
       [  8.   , 183.   ,  64.   , ...,   0.672,  32.   ,   1.   ],
       ...,
       [  5.   , 121.   ,  72.   , ...,   0.245,  30.   ,   0.   ],
       [  1.   , 126.   ,  60.   , ...,   0.349,  47.   ,   1.   ],
       [  1.   ,  93.   ,  70.   , ...,   0.315,  23.   ,   0.   ]])
```

In [5]:
```
df.shape
```

Out[5]:
```
(768, 9)
```

In [6]:
```
x = df[:,:8]
y = df[:,8]
```

In [7]:
```
from sklearn.model_selection import train_test_split
X_train, X_temp, y_train, y_temp = train_test_split(x, y, test_size=0.2, random_

X_test, X_val, y_test, y_val = train_test_split(X_temp, y_temp, test_size=0.5, r
```

In [8]:
```
print(f"x  train  shape{X_train.shape}")
print(f"y  train  shape{y_train.shape}")
print(f"x test shape{X_test.shape}")
print(f"y  test  shape{y_test.shape}")
print(f"x val shape{X_val.shape}")
print(f"y val shape{y_val.shape}")
```

```
x train shape(614, 8)
y train shape(614,)
x test shape(77, 8)
y test shape(77,)
x val shape(77, 8)
y val shape(77,)
```

In [9]:
```
from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
```