Data Science

Q2.) Difference between covariance & correlation.

Covariance	Correlation
(i) It is an indicator of how two random variable change concerning each other / Variable dependent on each other	(i) It indicates how strongly these two variable are related.
(ii) We can deduct correlation from a covariance.	(ii) It is deduced by dividing the calculated covariance by standard deviation.
(iii) The values of covariance lies in the range of $-\infty$ to $+\infty$	(iii) It is limited to values between the range -1 and +1
(iv) covariance is affected by the change of scale	(iv) It is not affected by the change of scale.
(v) Covariance has a definite unit as deduced by the multiplication of two no.s & their units.	(v) It is a unitless, absolute no. between -1 & +1 including decimal values.
(vi) We can calculate covariance for only two variables	(vi) Correlation can be calculated for multiple sets of numbers.

Q.3] Explain different Data transformation technique.

Ans:- Different data transformation technique are:-

(i) Data Smoothing:- It is a process that is used to remove noise from datasheet using some algorithm. It allows for highlighting important features present in the datasheet. It helps in predicting the patterns. The concept behind data smoothing is that it will be able to identify simple changes to help predict different trends & patterns. The noise is removed from the data using techniques such as

- Binning
- Regression
- Clustering

(ii) Attribute Construction:- The new attributes consult existing attributes to construct new data set that eases data mining. This simplifies the original data and makes the mining more efficient. For eg:- suppose we have a data set referring to measurements of different plots, i.e., we may have height & width of each plot. So here we can construct a new attribute 'area' from existing attributes 'height' & 'width'.

(iii) Data Aggression:- It is the method of storing & presenting data in a summary format. The data may be obtained from multiple data sources to integrate these data sources into a data analysis description. Gathering

accurate data of high quality & a large enough quantity is necessary to produce relevant results.
For e.g.:- we have a data set of sales reports of an enterprise that has quaterly sales of each year. We can aggregate the data to get the enterprise's annual sales report.

(iv) Data Normalization:- Normalizing the data refers to scaling the data values to a much smaller range such as [-1, 1] or [0.0, 1.0].
There are diff. methods to normalize the data :-

- Min-max normalization:- This method implements a linear transformation on the original data.
- Z-score normalization:- This method normalises the value for attribute A using the mean & standard deviation.
- Decimal scaling:- This method normalizes the value of attribute A by moving the decimal point in the value.

(v) Data Discretization:- This is a process of converting continous data into a set of data intervals. Continous attribute value are substituted by small interval labels. This makes the data easier to study and analyze. This improves the efficiency of the task. This method is also called a data reduction mechanism.

Two types of data discretization are:-
- Supervised discretization:- where class information is used.
- Unsupervised discretization:- based on which direction the process proceeds.

For eg:- the values of age attribute can be replaced by the interval labels such as (0-10, 11-20, ...) or (kid, youth, adult, senior).

(VI) Data Generalization:- It converts low level data attributes to high level data attributes using concepts hierarchy. This conversion from a lower level to higher level is useful to get a clearer picture of data.
It is divided into two approaches:
- Data cube process (OLAP) approach.
- Attribute-oriented induction (AOI) approach.

For e.g:- age data can be in the form of (20,30) in a datasheet. It is transformed into a higher conceptual level into a categorical value (young, old).

Q4.) Explain different data reduction technique.

Ans.(i) Dimensionality Reduction:- It eliminates the attributes from the data set under consideration thereby reducing the volume of original data. It reduces the size as it eliminates outdated or redundant features. Here are three methods of dimensionality reduction:-

- Wavelet Transform.
- Principal component Analysis.
- Attribute subset selection.

(ii) Numerosity Reduction :- It reduces the original data volume and represents it in a much smaller form. This technique includes two types parametric and non-parametric numerosity reduction.

- Parametric :- It only stores data parameters instead of the original data. One of its method is regression & long-linear.

- Non-Parametric :- It does not assume any model. It results in a more uniform reduction, irrespective of data size. Types :- Histogram, Clustering, Sampling.

(iii) Data Cube Aggregation :- It is used to aggregate data in a simpler form. It is a multidimension aggregation that uses aggregation at various levels of a data cube to represent original data.

(iv) Data Compression :- It employs modification, encoding or converting the structure of data in a way that consumes less space. Data that can be restored successfully from its compressed form is called Lossless compression. Dimensionality & numerosity are also used for data compression.

Types:
- Lossless Compression:- It uses algorithms to restore the precise original data from the compressed data.
- Lossy Compression:- The decompressed data may differ from the original data but are useful enough to retrieve information from them.

(v) Discretization Operation:- It is used to divide the attributes of the continous nature into data with interval. We replace many constant values of the attribute with label of small intervals. Types:- Top-down discretization, bottom-up discretization.

Q5.) What are web scraping Tools.

Ans.) (i) ParseHub:- is an powerful & elegant tool that allows you to build web scrapers without writing a single line of code.

Key features:
- Clean text & HTML before downloading data.
- Simple to use graphical interface.
- Automatic IP rotation.
- Can extract data from tables & Maps.
- Data is exported in JSON or Excel Format.

Shortcomings:
- Troubleshooting is not easy for larger projects
- The output can be very limiting at times.

(ii) Scrapy:- is a web scraping library used by python developers to build scalable web crawlers.

Key features:
- Open source Tool.
- Extremely well documented
- Easily extensible
- Portable Python

Shortcoming:
- In terms of Javascript support, it is time consuming.

(III) OctaParse:- has a target audience similar to Parsehub, catering to people who want to scrape data without having to write a single line of code.

Key features:
- Site Parser & hosted solution for users who want to run scrapers in the cloud.
- Annonymous Web data Scraping to avoid being banned.

Shortcoming:
- If run the crawler with local extraction instead of running it from the cloud it haults after 4 hours.

(IV) Scraper API :- is designed for designers building web scrapers.

Key features:-
- Helps you render JavaScript.
- Easy to integrate.
- Geolocated Rotating Proxies.

- Great speed & reliability to build web scrapers.

Shortcomings:
- It is a web scraping Tool is not deemed suitable for browsing.

(v) Mozenda :- caters to enterprises looking for a cloud-based serve Web Scraping platform.

Key features:-
- Offers point & click interface to create Web scraping events in no time.
- Best customer support & in-class account management.
- Provide both phone & e-mail support to all the customers.
- Highly scalable platform.
- Allows On-premise Hosting.

(vi) Webhose.io :- is best recommended ~~to~~ for platforms or services that are on the lookout for a completely developed web scraper.

key features:
- Content Indexing is fairly fast.
- Easy Integration with different solution.
- Dedicated support team that is highly reliable.
- Easy to use API

Shortcoming:
- Setup isn't that simplified for non-developers.
- The option for data retention of historical data was not available for a few users.

(VII) Content Grabber:- is a cloud-based web scraping Tool that helps businesses of all sizes with data extraction.

Key features:
- Web data extraction is faster compared to lot of its competitors.
- You can schedule it to scrape information from the web automatically.
- Offers a variety of formats for extracted data like CSV, JSON, etc

Shortcoming:
- Prior knowledge of HTML & HTTP required.
- Pre-configured crawlers for previously scraped websites not available.

(VIII) Common Crawl:- was developed for anyone wishing to explore & analyze data & uncover meaningful insights from it.

Key features:
- Open datasets of raw web page data & text extractions.
- Support for non-code based usage cases.
- Provides resources for educators teaching data analysis.

Shortcoming:
- Support for live data isn't available.
- Support for AJAX based sites isn't available.

Q6.) Difference between supervised, Unsupervised & Reinforcement ML:

Supervised	Unsupervised	Reinforcement
(i) It is trained using labelled data.	(i) It is trained using unlabelled data	(i) Works on interacting with the environment.
(ii) Type of data:- labelled data	(ii) Unlabelled data.	(ii) No predefined data.
(iii) It is categorized in Regression & classification.	(iii) Association and clustering	(iii) Exploitation and Exploration.
(iv) Extra supervision is required	(iv) No supervision	(iv) No supervision.
(v) Various algo. such as Linear regression, Logistic regression, SVM, KNN, etc.	(v) K-Means, C-Means, Apriori.	(v) Q-Learning, SARSA.
(vi) Aim:- Calculate outcomes.	(vi) Discover Underlying patterns.	(vi) Learns a series of action.
(vii) Application:- Risk Evaluation, Forecast sales.	(viii) Recommendation System, Anomaly Detection.	(vii) Self driving Cars, gaming, Healthcare

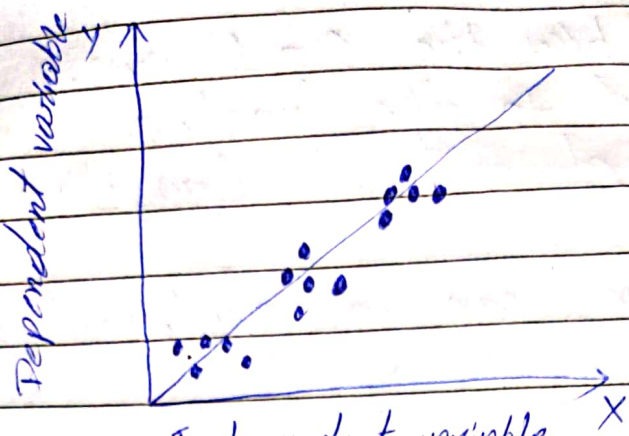
Q7:- Linear and Logist regression.

Ans:- Types of Regression:

- Linear Regression.
- Logistic Regression.
- Polynomial Regression.
- Support Vector Regression.
- Decision Tree Regression.
- Random Forest Regression.
- Ridge Regression.
- Lasso Regression.

Linear Regression:-

- It is a statistical regression method which is used for predictive analysis.
- It is one of the very simple & easy algo. which works on regression & shows relationship between the continous variables.
- It is used for solving regression problem in machine learning.
- It shows the linear relationship between the independent variable (X-axis) & the dependent variable (Y-axis).
- If there is only one input variable (x), then such linear regression is called simple linear regression, And if there is more than one input variable, then such linear regression is called multiple linear regression.
- The relationship between the variable in linear regression can be explained using the below diagram.



Mathematical equation:- $Y = aX + b$

Y = Dependent variable

X = Independent variable

a & b are the linear coefficient.

Popular application:-

- Analysing trends & sales estimates
- Salary forecasting
- Real estate prediction.
- Arriving at ETAs in traffic.

Logistic Regression:-

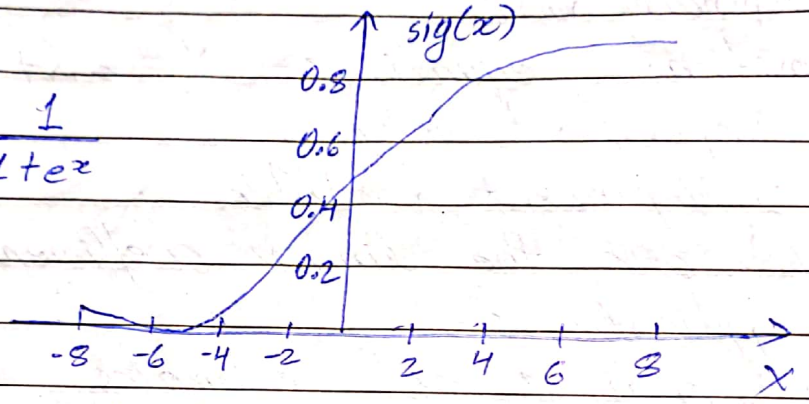
- It is another supervised learning algo which is used to solve the classification problems. In classification problems, we have dependent variable in a binary or discrete format such as 0 or 1.
- It works with the categorical variable such as 0 or 1, Yes or No, True or False, Spam or Not spam, etc.
- Logistic regression is a type of regression but it is different from linear regression algo in the terms how they are used.
- It uses sigmoid function. The sigmoid function can be represented as:

$$f(x) = \frac{1}{1 + e^{-x}}$$

- $f(x)$ = Output b/w the 0 & 1 value.
- x = input to the function.
- e = base of natural logarithm.

- Logistic regression can be explained using below graph:

$$sig(x) = \frac{1}{1+e^{x}}$$



Types of logistic regression:
- Binary (0/1, pass/fail)
- Multi (cats, dogs, lions)
- Ordinal (low, medium, high).

Q8.) Matplotlib pie chart using explode, shadow, legend.

Ans.) Creating Pie charts

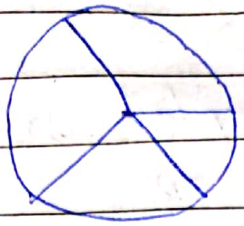
With Pyplot, you can use the pie() function to draw pie charts:

Example:

```
import matplotlib.pyplot as plt
import numpy as np

y = np.array([35, 25, 25, 15])
plt.pie(y)
plt.show()
```

Result:



Labels:
Add labels to the pie chart with the label parameter.

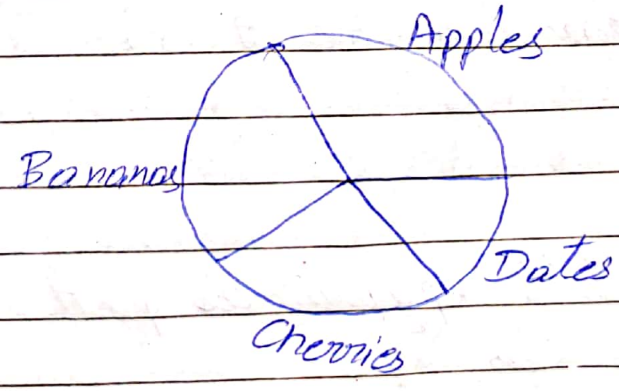
Example:

```
import matplotlib.pyplot as plt
import numpy as np

y = np.array([35, 25, 25, 15])
mylabels = ["Apples", "Bananas", "Cherries", "Dates"]

plt.pie(y, labels = mylabels)
plt.show()
```

Result:



Explode:
Explode parameter allows your one of the wedges to stand out.

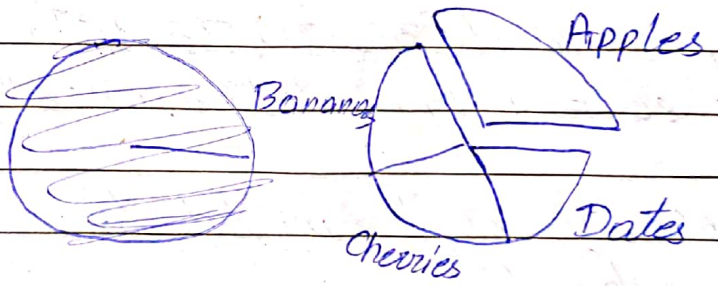
Example:

```
import matplotlib.pyplot as plt
import numpy as np

y = np.array([35, 25, 25, 15])
mylabels = ["Apples", "Bananas", "Cherries", "Dates"]
myexplode = [0.2, 0, 0, 0]

plt.pie(y, labels = mylabels, explode = myexplode)
plt.show()
```

Result:



Shadow:
Add a shadow to the pie chart by setting the shadows parameters to True

Example:

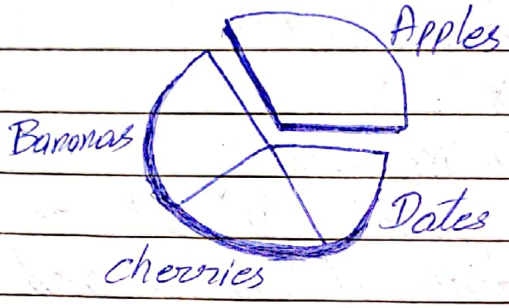
```
import matplotlib.pyplot as plt
import numpy as np

y = np.array([35, 25, 25, 15])
mylabels = ["Apples", "Bananas", "Cherries",
            "Dates"]
explode = myexplode, sh
```

```
myexplode = [0.2, 0, 0, 0]

plt.pie(y, labels = mylabels, explode = myexplode,
        shadow = True)
plt.show()
```

Result:



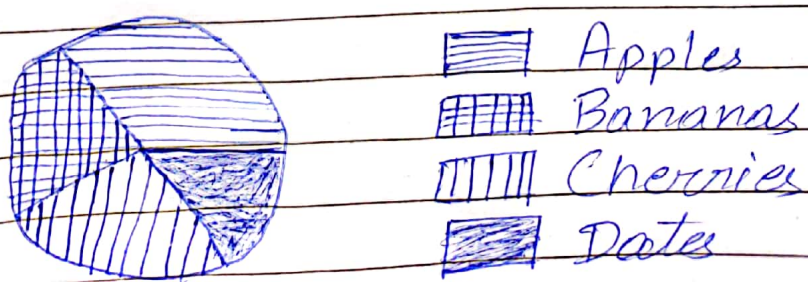
Legend:
To add a ~~list~~ list of explanation for each wedges, use legend() function:

Example:

```
import matplotlib.pyplot as plt
import numpy as np

y = np.array([35, 25, 25, 15])
mylabels = ["Apples", "Bananas", "Cherries", "Dates"]
plt.pie(y, labels = mylabels)
plt.legend()
plt.show()
```

Result:



89.) Difference between Matplotlib & Seaborn.

Ans.)

Matplotlib	Seaborn
(i) It is used for making basic graphs	(i) Seaborn utilises fascinating themes.
(iii) It contains a few plots and patterns for data visualisation.	(ii)
(ii) Datasets are visualised with the assistance of lines, scatter plots, pie charts, etc.	(ii) It contains a few plots and patterns for data visualisation.
(iii) It is very much associated with Pandas and NumPy & goes about as a graphic package	(iii) It is more agreeable in taking care of data frames in Pandas.
(iv) It is profoundly robust and customised.	(iv) It tries not to cover plots with the assistance of its default themes.
(v) It acts productively with data arrays and frames.	(v) It is considerably more organised and functional & treats the entire dataset as a solitary unit.

Q10. Diff. bet. Overfitting & Underfitting.

Ans:

Underfitting	Overfitting
(i) High Bias, Low variance	(i) Low Bias, high variance
(ii) Performs poorly on trained data	(ii) Performs well on trained data.
(iii) Training accuracy and validation accuracy are poor.	(iii) Training accuracy is very good but validation accuracy is poor.
(iv) Happens when we have less amount of data	(iv) Happens when we train our model a lot over noisy datasets.
(v) It is more complex model.	(v) It is more simple model.
(vi) It has Larger quantity of features.	(vii) It has smaller quantity of features.
(vii) It requires less regularization.	(vii) It requires more regularization.
(viii) More data can't help.	(viii) More data can help.