Question bank

Module 3 - Questions

1) **Demonstrate the need for Data visualization. State the advantage of data visualization.**

Data visualization is crucial for conveying insights, trends, and patterns hidden within datasets, making it an essential tool for decision-making, communication, and analysis in various fields. Here's a demonstration of the need for data visualization along with the advantages it offers:

Demonstration of the Need for Data Visualization:
Imagine you're analyzing a dataset containing monthly sales data for a retail company over the past five years. The dataset includes information such as sales revenue, product categories, regions, and customer demographics. Without visualization, you might face several challenges:

Understanding Trends and Patterns: Raw data in tabular form can be challenging to interpret, especially when dealing with large datasets. It's difficult to identify trends, seasonality, or anomalies by looking at rows and columns of numbers.

Spotting Outliers and Anomalies: Detecting outliers or anomalies hidden within the data is difficult without visualization. Anomalies might indicate errors in data collection, significant events, or emerging trends that require attention.

Communicating Insights: Communicating insights and findings to stakeholders or colleagues is challenging without visual aids. Text-based summaries or reports may not effectively convey the richness of the data or the complexity of relationships within it.

Advantages of Data Visualization:
Enhanced Understanding: Data visualization enables users to understand complex datasets more quickly and intuitively. Visual representations, such as charts, graphs, and maps, make it easier to identify trends, relationships, and patterns within the data.

Improved Decision-Making: Visualizing data allows decision-makers to gain insights and make informed decisions more effectively. By visualizing key metrics, trends, and forecasts, decision-makers can identify opportunities, mitigate risks, and optimize strategies.

Facilitates Exploration and Analysis: Data visualization tools provide interactive features that enable users to explore data dynamically. Users can drill down into specific data points, filter information, and uncover insights in real-time, enhancing the analysis process.

Effective Communication: Visualizations are powerful tools for communicating complex ideas and findings to a wide range of audiences. Visual representations simplify complex concepts, making them more accessible and engaging for stakeholders, clients, or team members.

Identifying Patterns and Trends: Visualizing data helps users identify patterns, correlations, and trends that may not be apparent in raw data. By visualizing data over time or across different dimensions, users can uncover valuable insights and actionable intelligence.

Detecting Anomalies and Outliers: Visualizations make it easier to detect outliers, anomalies, or irregularities within datasets. Visual representations highlight unusual data points, enabling users to investigate further and take appropriate actions.

In summary, data visualization is essential for gaining insights, communicating findings, and making informed decisions in various domains. It enhances understanding, facilitates analysis, and enables users to derive actionable insights from complex datasets, ultimately driving better outcomes and improving decision-making processes.

2) **State the significant challenges in visualizing big data and how to overcome these challenges.**

Visualizing big data is a challenging task due to the sheer volume, variety, and complexity of data involved. Some of the significant challenges in visualizing big data are:

- Scalability
- Data variety.
- Data quality
- Data complexity
- High performance requirements

To overcome these challenges, the following steps can be taken:

- Meeting the need for speed
- Simplify the Data
- Filter and Subset Data
- Use Appropriate Visualization Techniques
- Data Cleaning and Quality Improvement

3) **Explain Bootstrapping and state its advantages and disadvantages.**

Cross-Validation is a resampling technique with the fundamental idea of splitting the dataset into 2 parts- training data and test data. Train data is used to train the model and the unseen test data is used for prediction. If the model performs well over the test data and gives good
accuracy, it means the model hasn’t overfitted the training data and can be used for prediction.

Validation Set Approach
We divide our input dataset into a training set and test or validation set in the validation set approach. Both the subsets are given 50% of the dataset

Holdout Method
This method is the simplest cross-validation technique among all. In this method, we need to remove a subset of the training data and use it to get prediction results by training it on the rest of the dataset.

Leave-P-out cross-validation
In this approach, the p datasets are left out of the training data. It means, if there are total n data points in the original input dataset, then n-p data points will be used as the training dataset and the p data points as the validation set. This complete process is repeated for all the samples, and the average error is calculated to know the effectiveness of the model.

Leave one out cross-validation
This method is similar to the leave-p-out cross-validation, but instead of p, we need to take 1 dataset out of training. It means, in this approach, for each learning set, only one datapoint is reserved, and the remaining dataset is used to train the model. This process repeats for each datapoint.

K-Fold Cross-Validation
K-fold cross-validation approach divides the input dataset into K groups of samples of equal sizes. These samples are called **folds**. For each learning set, the prediction function uses k-1 folds, and the rest of the folds are used for the test set.

Stratified k-fold cross-validation
This technique is similar to k-fold cross-validation with some little changes. This approach works on stratification concept, it is a process of rearranging the data to ensure that each fold or group is a good representative of the complete dataset. To deal with the bias and variance, it is one of the best approaches.

Time-series cross-validation: This method is used for time-series data, where the data is split into training and testing sets in a sequential manner, with the testing set always following the training set.

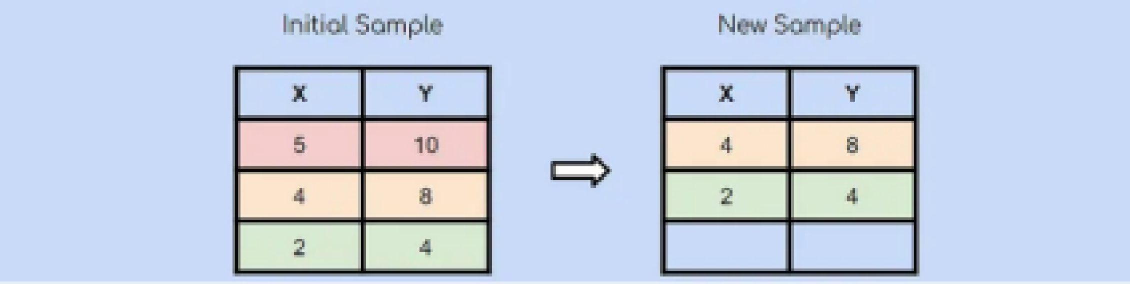
Repeated cross-validation: In this method, the k-fold cross-validation process is repeated multiple times, with different random splits of the data, to obtain a more robust estimate of model performance.

4) Identify the technique used to evaluate the performance of a model on unseen data. Also list all its types.

Bootstrapping is a statistical technique used for random sampling with replacement. It involves taking random samples from a dataset and using those samples to estimate the variability of a statistic of interest, such as the mean or standard deviation. The samples are taken with replacement, meaning that each observation has an equal chance of being selected for each sample.

Here's how bootstrapping works:

1. Starting with the original dataset, a large number of "bootstrap samples" are created by randomly sampling observations from the original dataset with replacement.
2. For each bootstrap sample, the statistical estimator or model is calculated and recorded.
3. The distribution of the estimator or model across all of the bootstrap samples is used to estimate the uncertainty or variability of the estimator or model.

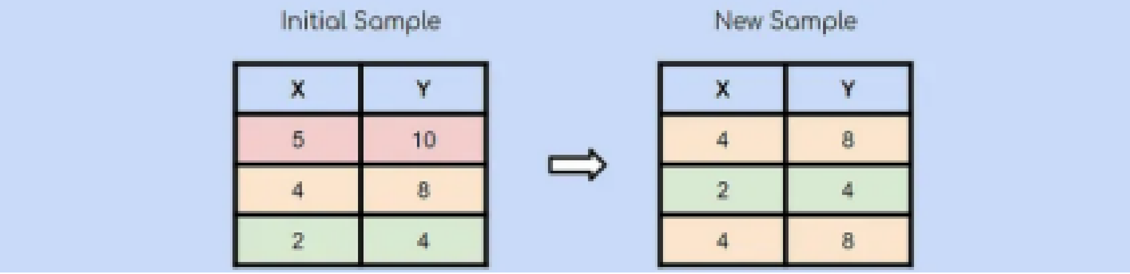


Initial Sample

X	Y
5	10
4	8
2	4

New Sample

X	Y
4	8
2	4



Initial Sample

X	Y
5	10
4	8
2	4

New Sample

X	Y
4	8
2	4
4	8

Advantages of bootstrapping: NSFAR

- Non-parametric.
- Simplicity
- Flexibility
- Accuracy
- Robustness

Disadvantages of bootstrapping: CBSD

- Computationally intensive
- Bias
- Sampling variability
- Dependence on random number generator

4. State the causes of the outlier. MNDSU

Measurement error: Outliers can be caused by errors in data collection or measurement. For example, a sensor malfunction or human error can lead to an extreme value being recorded.

Natural variation: In some cases, outliers may occur due to natural variation in the data. For example, in a distribution with a long tail, there may be some extreme values that are not necessarily errors but reflect the true variability of the data.

Data processing errors: Outliers can also be caused by errors in data processing, such as data entry or data manipulation.

Sampling error: Outliers may also occur due to sampling error, which is the result of selecting a sample that is not representative of the population.

Unusual events: Outliers may be caused by unusual or rare events that are not representative of the typical data. For example, a stock market crash or a natural disaster can lead to extreme values in financial or environmental data.

5) Highlight the importance of data visualization and outline its benefits.(same as 1) repeated
6) Elaborate on Bootstrapping and enumerate its pros and cons.(same as 3) repeated

Module 4-Questions

1) **State the need for anomaly detection? Specify the basic approaches to anomaly detection. Enlist the application of anomaly detection.**

Anomaly detection is an important task in data analysis and machine learning. It involves identifying patterns in data that are unusual or unexpected, and can be used for a variety of applications including fraud detection, intrusion detection, fault detection, and quality control.

The need for anomaly detection arises because anomalies or outliers can have a significant impact on *statistical analyses and modeling*, and may indicate the presence of *unusual or abnormal behavior* that needs to be investigated. By identifying and removing or flagging anomalies, data analysts and machine learning engineers can improve the accuracy of their models and gain insights into potential issues.

There are several approaches to anomaly detection, including:

- Statistical Methods
- Machine Learning Methods
- Rule Based Methods
- Deep Learning Methods
- Density Based Methods
- Distance Based Methods

Some common applications of anomaly detection include: FNQPM

- Fraud detection
- Network intrusion detection
- Quality control
- Predictive maintenance

- Medical diagnosis

2) **Explain outlier detection using Density-based methods.**

Density-based methods use measures of local density to identify outliers. The most commonly used density-based method is the DBSCAN (Density-Based Spatial Clustering of Applications with Noise) algorithm, which identifies outliers as data points that are not part of any cluster or are part of a small, sparse cluster.

Suppose we have 5 data points in two-dimensional space: (1,1), (1,2), (2,1), (2,2), and (10,10). We want to detect the outlier in this data set using DBSCAN.

Here are the steps:

- Choose the parameters: We need to choose the parameters for DBSCAN. Let's choose epsilon (eps) as 1.5 and minimum points (min_samples) as 2. - Calculate distances: Calculate the distances between each data point and all other data points.
- Define core points, border points, and noise points: A point is a core point if it has at least min_samples points within a distance of eps. A point is a border point if it is not a core point but is within a distance of eps from a core point. A point is a noise point if it is neither a core point nor a border point.
- Assign labels: Assign labels to the points. Start with an arbitrary point, and if it is a core point, assign it a new label. Then, recursively expand the cluster by adding border points to the cluster. Finally, mark noise points as outliers.

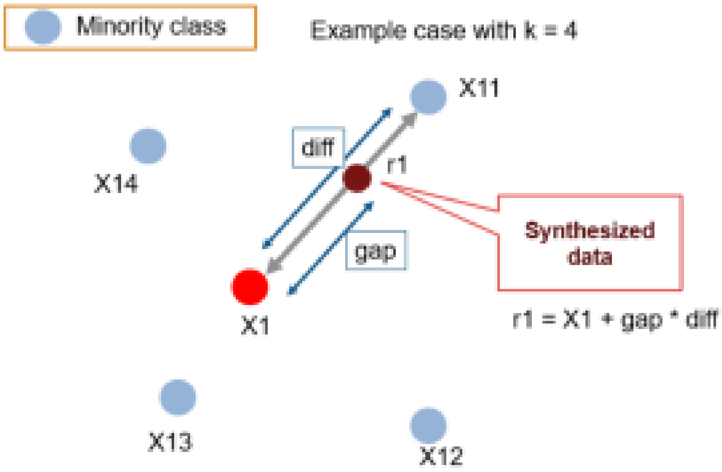
Result: In this example, points (1,1), (1,2), (2,1), and (2,2) are core points because they all have at least 2 other points within a distance of 1.5. The point (10,10) is a noise point because it does not have any other point within a distance of 1.5. Therefore, (10,10) is the outlier in this data set.

3) **Explain SMOTE.**

SMOTE (Synthetic Minority Over-sampling Technique) is an algorithm used for oversampling in imbalanced datasets. It addresses the overfitting problem posed by random oversampling by generating *synthetic samples rather than duplicating existing* ones. SMOTE works by selecting a minority class sample and finding its *k nearest minority class neighbors*. It focuses on the feature space to generate new instances with the help of *interpolation* between the positive instances that lie together. It then generates new minority class samples along the line segments joining these neighbors.

For example, consider a dataset with two classes: Class A (minority) and Class B (majority). The dataset has 100 samples, with 10 samples belonging to Class A and 90 samples belonging to Class B. This is an imbalanced dataset, with Class A being the minority class.

Using SMOTE, new minority class samples are generated by selecting a minority class sample and finding its k nearest minority class neighbors. Suppose k=4, and the algorithm selects sample X1 as the starting point. The 4 nearest minority class neighbors are X11, X12, X13, X14. SMOTE generates new samples along the line segments joining X1 to each of its neighbors, resulting in 4 new synthetic minority class samples.



The advantage of SMOTE over random oversampling is that it *reduces the risk of overfitting* by generating synthetic samples that are not exact copies of existing samples. This can *improve the generalization* of machine learning models trained on imbalanced datasets.

However, SMOTE may also *introduce some noise* into the dataset, especially if the k nearest neighbors are not well-chosen. In addition, SMOTE may not work well for datasets with high dimensionality or non-linear decision boundaries.

To overcome these limitations, variants of SMOTE have been proposed, such as *Borderline SMOTE*, which only generates synthetic samples for borderline instances, and *ADASYN*, which adaptively generates synthetic samples *based on the density* of minority class instances.

4) Discuss the importance of anomaly detection and outline the fundamental methods for anomaly detection. List the various applications where anomaly detection is utilized.(same as 1) repeated
5) Illustrate outlier detection through Distance-based methods (same as 2) repeated

Module 5 questions: Time series forecasting

1) **Explain what is a difference between time series analysis and time series forecasting**

   Time series analysis and time series forecasting are related concepts but serve different purposes in the realm of data analysis and decision-making. Here's a breakdown of the key differences between them:

   Time Series Analysis:

Purpose: Time series analysis focuses on understanding the underlying structure, patterns, and dynamics present in a sequence of data points collected over time. It involves exploring historical data to uncover trends, seasonality, cyclic patterns, and irregularities.

Techniques: Time series analysis techniques include descriptive statistics, data visualization, autocorrelation analysis, decomposition (e.g., trend, seasonality, and noise separation), spectral analysis, and stationarity testing.

Goals: The primary goals of time series analysis are to gain insights into the behavior of the data, identify patterns and relationships, detect anomalies or outliers, and assess the presence of underlying trends or seasonal effects.

Examples: Time series analysis can involve examining historical sales data to identify seasonal peaks, analyzing temperature data to detect long-term trends or cyclical patterns, or investigating stock price movements to uncover correlations or dependencies.

Time Series Forecasting:

Purpose: Time series forecasting involves using historical data to predict future values or trends in a time series. It aims to make informed predictions about future outcomes based on past observations, allowing organizations to plan and make decisions proactively.

Techniques: Time series forecasting techniques include statistical methods (e.g., ARIMA, Exponential Smoothing), machine learning algorithms (e.g., LSTM networks, Prophet), and hybrid approaches that combine multiple models or techniques.

Goals: The primary goal of time series forecasting is to generate accurate predictions of future values or trends, enabling businesses and organizations to anticipate demand, manage resources, optimize operations, and make strategic decisions.

Examples: Time series forecasting can involve predicting future sales volumes based on historical sales data, forecasting stock prices or exchange rates, anticipating electricity demand for energy planning, or projecting future website traffic for capacity planning.

In summary, time series analysis focuses on understanding past data patterns and dynamics, while time series forecasting leverages this understanding to make predictions about future outcomes. Time series analysis provides the foundation for time series forecasting by uncovering insights and relationships in historical data, which are then used to develop and evaluate forecasting models.

2) Describe in detail ARIMA model

ARIMA (AutoRegressive Integrated Moving Average) model is a popular time series forecasting method that combines the autoregressive (AR) and moving average (MA) models with differencing to account for non-stationarity in the time series data. ARIMA models are widely used in various fields such as economics, finance, and engineering to forecast future values of a time series based on its historical data.

The ARIMA model consists of three components:
Autoregressive (AR) component: This component models the linear relationship between the current observation and the previous p observations (p is the order of the autoregressive component). The AR component uses a regression equation that includes the past values of the time series, weighted by their respective coefficients.

$$y_t = c + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \dots + \phi_p y_{t-p} + \varepsilon_t$$

where,
yt is the time series observation at time t
c is a constant
ϕ1, ϕ2, … , ϕp are the autoregressive coefficients
εt is the white noise error term at time t

Moving Average (MA) component: This component models the linear relationship between the current observation and the previous q forecast errors (q is the order of the moving average component). The MA component uses a regression equation that includes the past forecast errors, weighted by their respective coefficients.

$$y_t = c + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \dots + \theta_q \varepsilon_{t-q}$$

where,
θ1, θ2, … , θq are the moving average coefficients

Integrated (I) component: This component models the differencing required to make the time series stationary. Stationarity is a desirable property of a time series that ensures that its statistical properties, such as mean and variance, are constant over time. The I component involves taking the difference between the time series and its lagged value until it becomes stationary.

$$yt = (1-B)^d Z_t$$

where,
B is the backshift operator
Zt is the differenced time series
d is the degree of differencing

The ARIMA model is denoted as ARIMA(p,d,q),

p: the number of lag observations in the model, also known as the lag order. d: the number of times the raw observations are differenced; also known as the degree of differencing.
q: the size of the moving average window, also known as the order of the moving average.
The parameters of the ARIMA model are estimated using the *maximum likelihood estimation* method based on the historical data. Once the parameters are estimated, the ARIMA model can be used to forecast future values of the time series.

The ARIMA model is fitted by estimating the values of the model parameters (ϕ1, ϕ2, … , ϕp, θ1, θ2, … , θq, d) that minimize the sum of squared errors between the actual and predicted values. The estimated model can then be used to forecast future values of the time series.
The ARIMA model has several advantages, including its ability to handle both stationary and non-stationary time series, its flexibility to incorporate external factors, and its ability to provide probabilistic forecasts. However, the ARIMA model also has some limitations, such as its assumption of linear relationships between variables and its sensitivity to outliers.

3) **Discuss use of any 2 machine learning methods for time series forecasting**
   A. ARIMA
   B. LSTM is a type of recurrent neural network (RNN) architecture designed to handle sequence prediction problems, making it well-suited for time series forecasting. Unlike traditional feedforward neural networks, LSTM networks have feedback connections that enable them to retain information over long sequences.

   Key features of LSTM networks include:

   Memory Cells: LSTM networks contain memory cells that can maintain information over time, allowing them to capture long-term dependencies in time series data.

   Gates: LSTM networks have three types of gates (input, forget, and output gates) that regulate the flow of information through the network. These gates enable LSTMs to selectively update and forget information based on the current input and past context.

   Backpropagation Through Time (BPTT): LSTMs are trained using BPTT, a variant of backpropagation specifically designed for sequence data. This allows the network to learn temporal patterns and make accurate predictions.

   LSTMs are particularly effective for capturing nonlinear dependencies and handling complex time series data with irregular patterns, making them suitable for a wide range of forecasting tasks, including stock prices, weather forecasting, and demand forecasting.

4) **Illustrate various smoothing methods applied on the time series data with an example.**

   Naive, Seasonal naive, Average, moving average, weighted average, exponential

average
Naive Method: This is a simple method that involves using the most recent observation as the forecast for the next period. For example, if we have a time series of monthly sales data, the forecast for next month would be the sales value for the current month.

Seasonal Naive Method: Similar to the naive method, but instead of using the most recent observation, it uses the corresponding observation from the previous season. For example, if we have a time series of quarterly sales data, the forecast for next quarter would be the sales value for the same quarter in the previous year.

Simple Average Method: This method involves taking the average of all the past observations and using it as the forecast for the next period. For example, if we have a time series of daily temperature data, we could calculate the average temperature for the past 30 days and use it as the forecast for tomorrow's temperature.

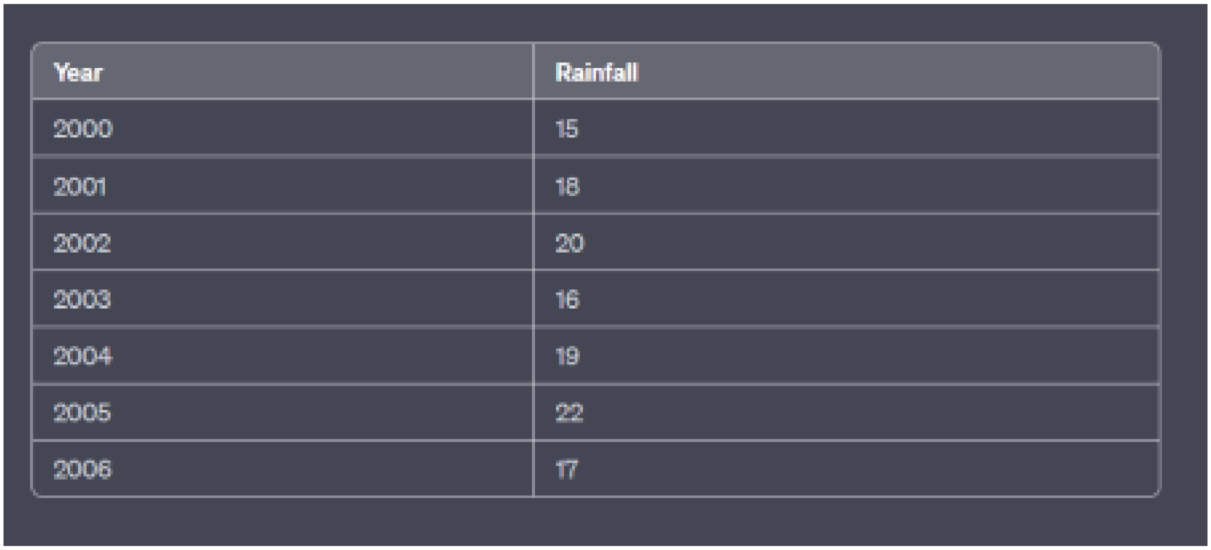
Moving Average Method: Similar to the simple average method, but only considers a window of the most recent observations. For example, if we have a time series of hourly website traffic data, we could calculate a 7-day moving average by taking the average of the traffic values for the current hour and the previous 167 hours.

Weighted Moving Average Method: Similar to the moving average method, but assigns different weights to each observation in the window. The weights can be used to emphasize or de-emphasize certain observations based on their relative importance.

Exponential Smoothing Method: This method involves assigning exponentially decreasing weights to past observations, with the most recent observations given the highest weights. The weights are determined by a smoothing factor, which controls how quickly the weights decrease. This method is particularly useful for data with trend and seasonality.

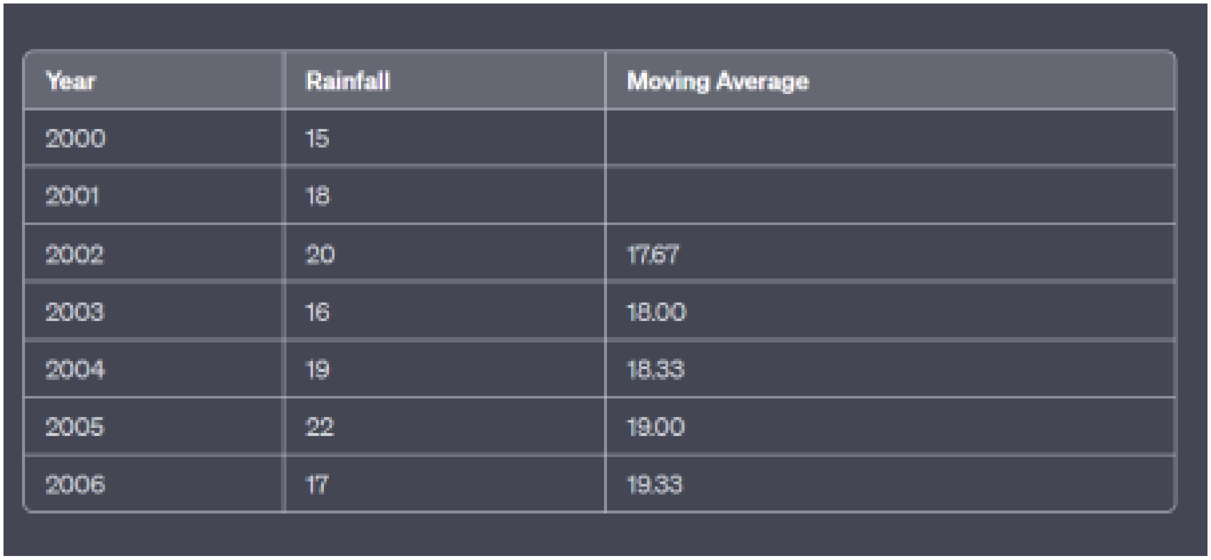
Smoothing methods are commonly used in time series analysis to remove noise and identify underlying trends or patterns in the data. Some of the common smoothing methods used in time series analysis are:

Moving Average (MA) Smoothing: This method involves calculating the average of a fixed number of consecutive data points and using this average value as the smoothed value. The size of the moving window or the number of data points to be averaged is determined by the analyst. For example, a 3-period moving average can be calculated for the following time series data:



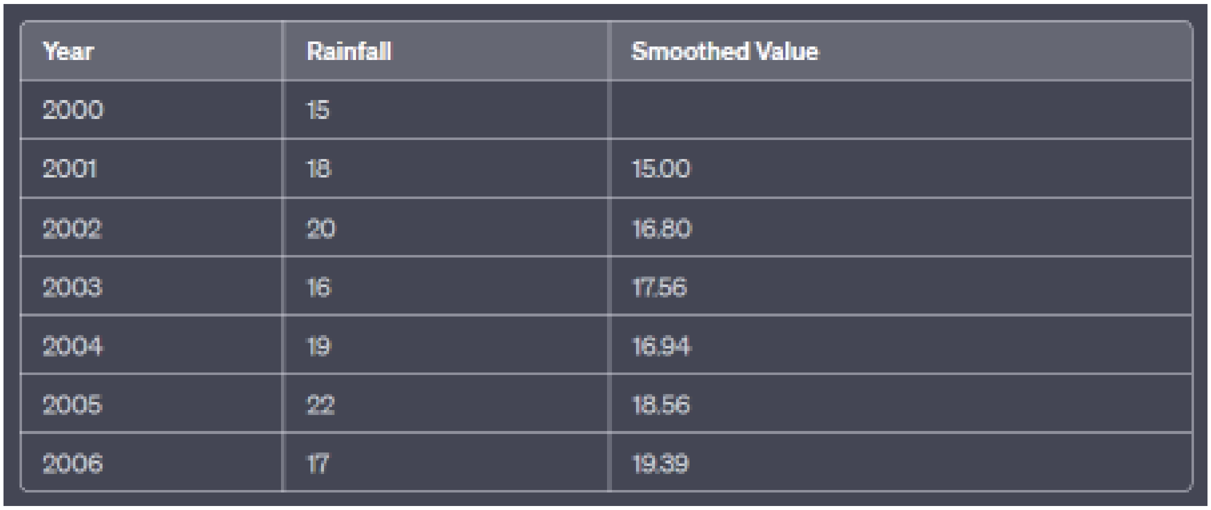
Year	Rainfall
2000	15
2001	18
2002	20
2003	16
2004	19
2005	22
2006	17

The 3-period moving average for this time series can be calculated as follows:



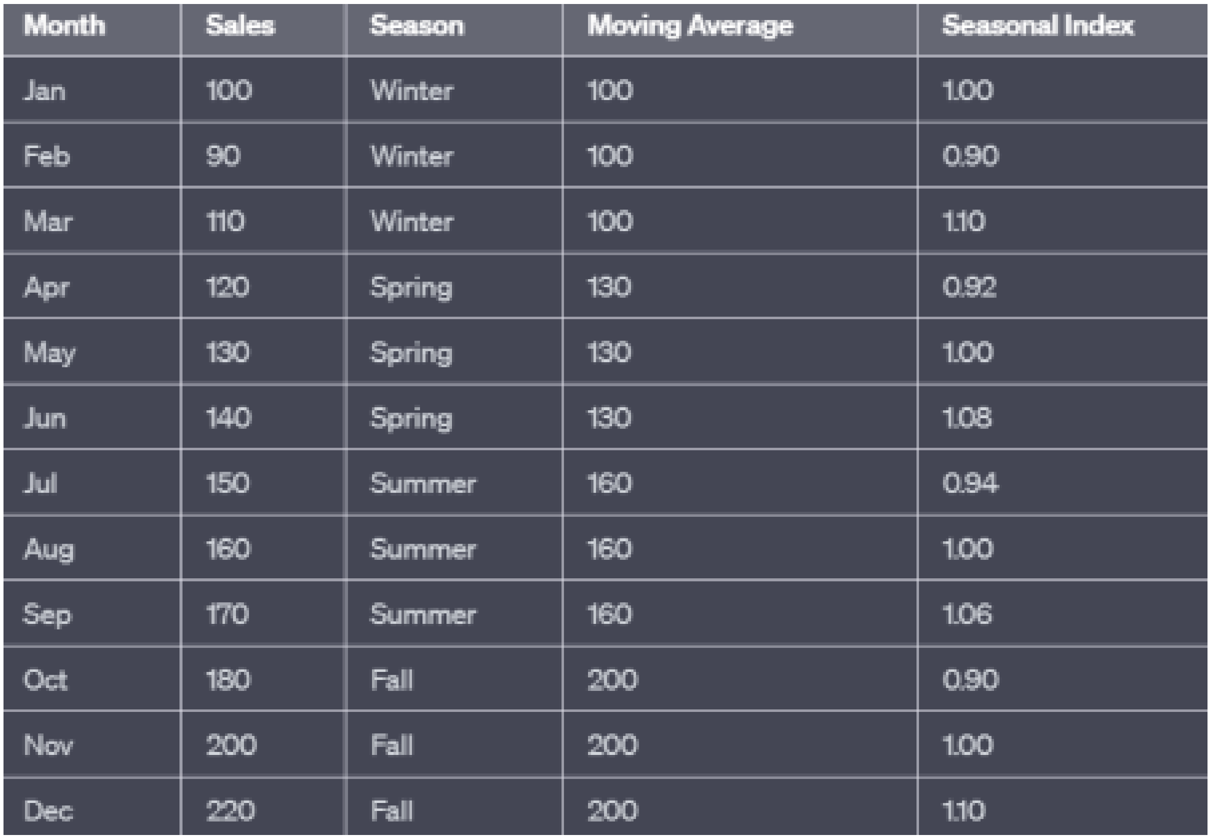
Year	Rainfall	Moving Average
2000	15	
2001	18	
2002	20	17.67
2003	16	18.00
2004	19	18.33
2005	22	19.00
2006	17	19.33

Exponential Smoothing: This method involves calculating a weighted average of past observations, with more weight given to recent observations. The weights are determined by a smoothing parameter, which is usually between 0 and 1. For example, an exponential smoothing model can be applied to the same rainfall time series data with a smoothing parameter of 0.3, as follows:

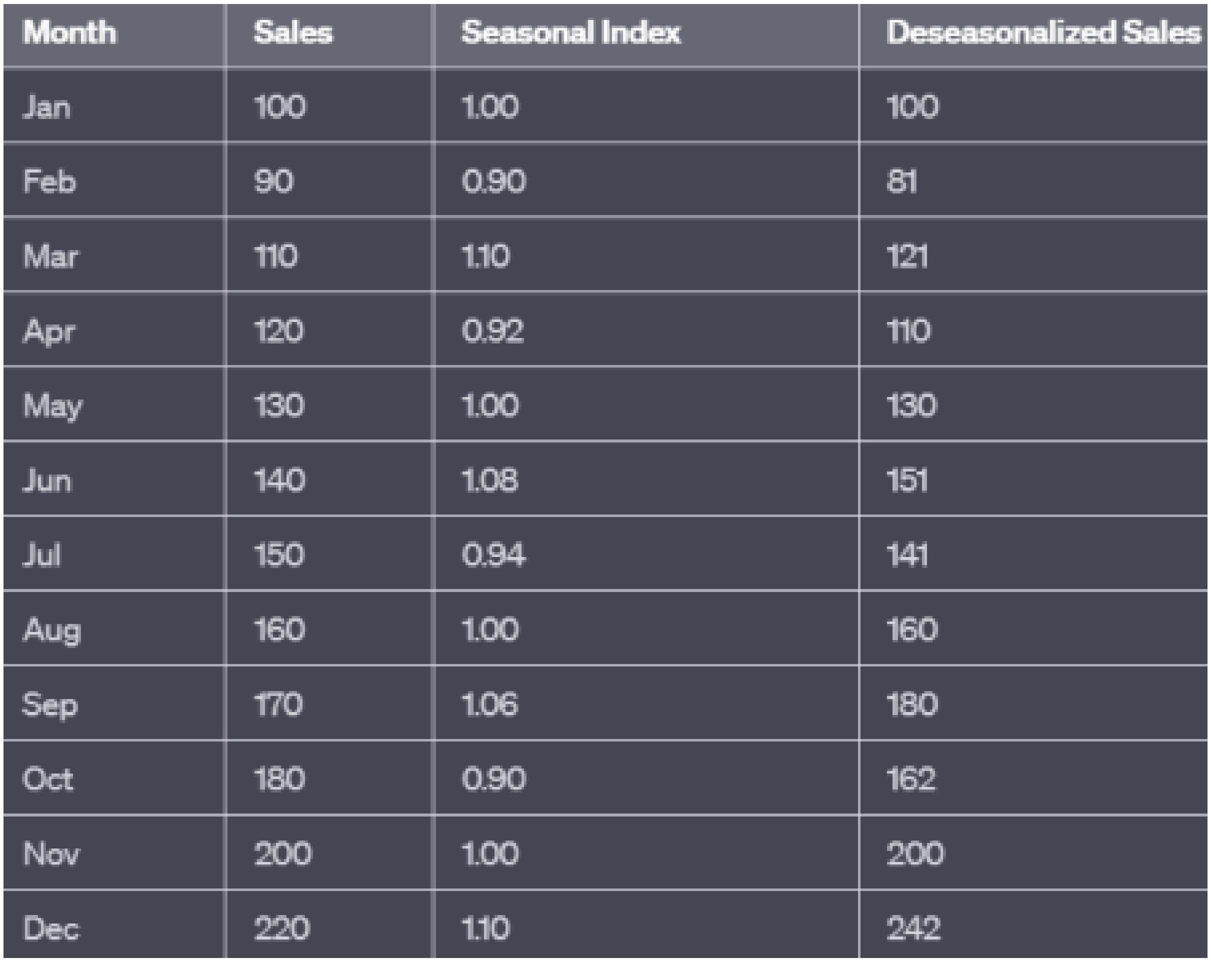


Year	Rainfall	Smoothed Value
2000	15	
2001	18	15.00
2002	20	16.80
2003	16	17.56
2004	19	16.94
2005	22	18.56
2006	17	19.39

Seasonal Smoothing: This method is used to remove seasonality from the time series data. Seasonal smoothing involves taking a moving average of the data within each season. For example, a seasonal smoothing model can be applied to monthly sales data for a retail store to remove the seasonal effect of Christmas sales. The smoothed value for each month would be the average of the same month's sales data over the past few years.



Month	Sales	Season	Moving Average	Seasonal Index
Jan	100	Winter	100	1.00
Feb	90	Winter	100	0.90
Mar	110	Winter	100	1.10
Apr	120	Spring	130	0.92
May	130	Spring	130	1.00
Jun	140	Spring	130	1.08
Jul	150	Summer	160	0.94
Aug	160	Summer	160	1.00
Sep	170	Summer	160	1.06
Oct	180	Fall	200	0.90
Nov	200	Fall	200	1.00
Dec	220	Fall	200	1.10



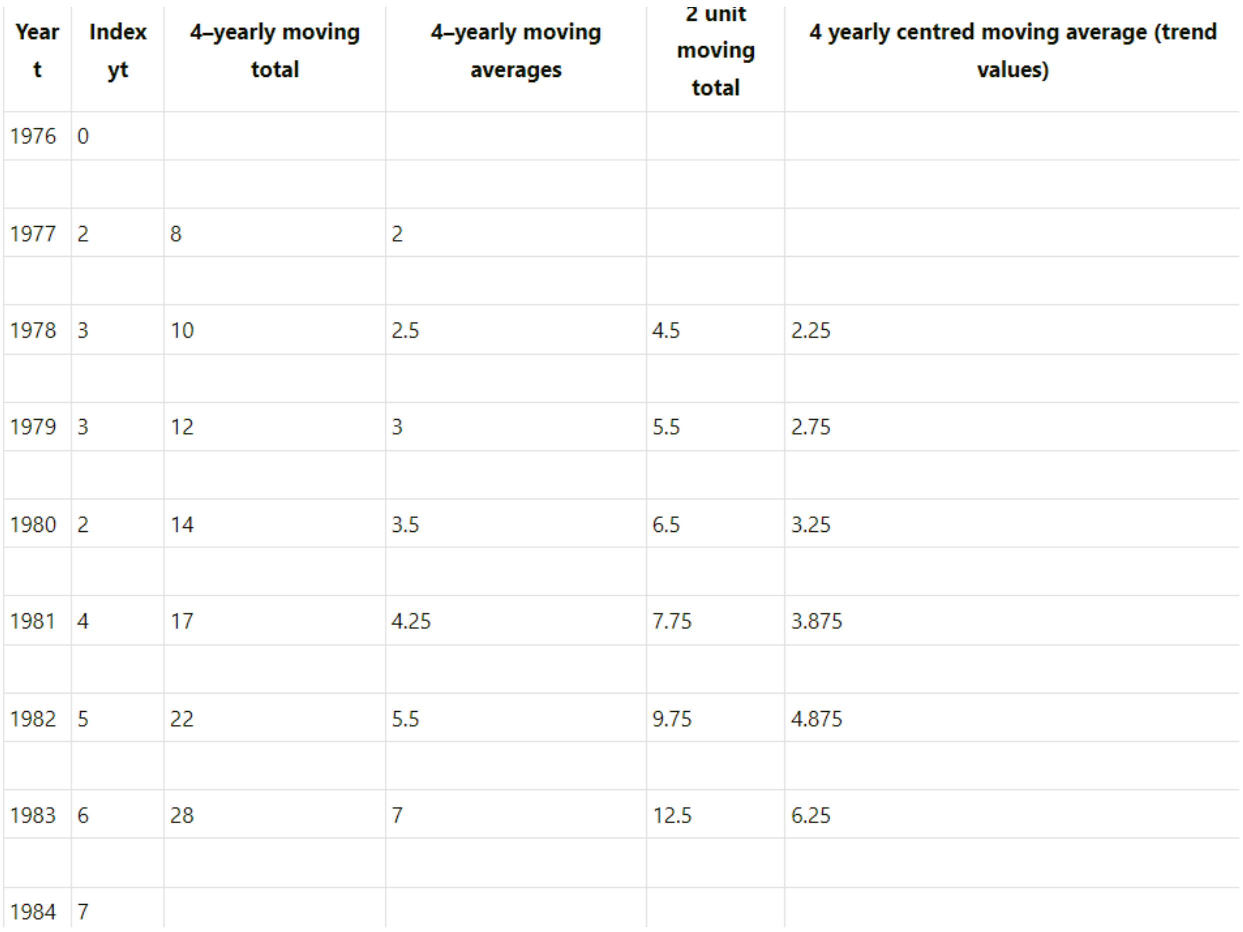
Month	Sales	Seasonal Index	Deseasonalized Sales
Jan	100	1.00	100
Feb	90	0.90	81
Mar	110	1.10	121
Apr	120	0.92	110
May	130	1.00	130
Jun	140	1.08	151
Jul	150	0.94	141
Aug	160	1.00	160
Sep	170	1.06	180
Oct	180	0.90	162
Nov	200	1.00	200
Dec	220	1.10	242

Overall, smoothing methods can help to identify underlying trends and patterns in time series data by removing noise and seasonality. The choice of a particular smoothing method depends on the nature of the time series data and the specific research question at hand.

Link —
https://www.toppr.com/ask/question/construct-3-yearly-moving-averages-from-the-following-data-and/

https://www.shaalaa.com/question-bank-solutions/obtain-the-trend-values-for-the-above-data-using-3-yearly-moving-averages-measurement-of-secular-trend_156481

Year	1976	1977	1978	1979	1980	1987	1982	1983	1984	1985
Index	0	2	3	3	2	4	5	6	7	10

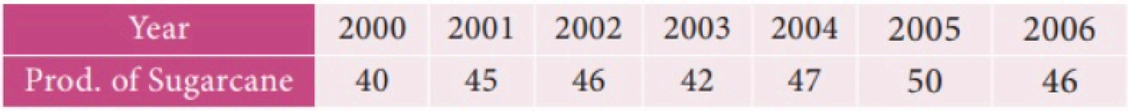


Year t	Index yt	4–yearly moving total	4–yearly moving averages	2 unit moving total	4 yearly centred moving average (trend values)
1976	0				
1977	2	8	2		
1978	3	10	2.5	4.5	2.25
1979	3	12	3	5.5	2.75
1980	2	14	3.5	6.5	3.25
1981	4	17	4.25	7.75	3.875
1982	5	22	5.5	9.75	4.875
1983	6	28	7	12.5	6.25
1984	7				

5. Link - for Method of least square :-- https://www.brainkart.com/article/Measurements-of-Trends--Method-of-Least-Squares_39018/

Given below are the data relating to the production of sugarcane in a district.

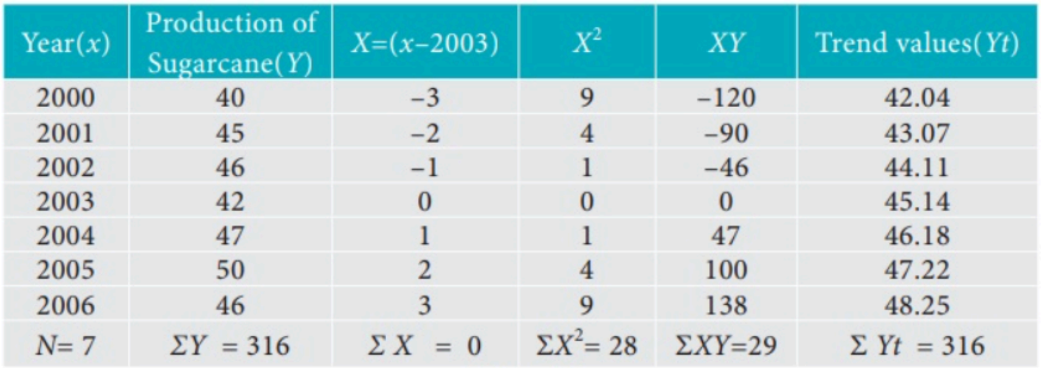
Fit a straight line trend by the method of least squares and tabulate the trend values.



Year	2000	2001	2002	2003	2004	2005	2006
Prod. of Sugarcane	40	45	46	42	47	50	46

Solution:

Computation of trend values by the method of least squares (ODD Years).



Year(x)	Production of Sugarcane(Y)	X=(x–2003)	X^2	XY	Trend values(Yt)
2000	40	–3	9	–120	42.04
2001	45	–2	4	–90	43.07
2002	46	–1	1	–46	44.11
2003	42	0	0	0	45.14
2004	47	1	1	47	46.18
2005	50	2	4	100	47.22
2006	46	3	9	138	48.25
N= 7	$\Sigma Y = 316$	$\Sigma X = 0$	$\Sigma X^2 = 28$	$\Sigma XY = 29$	$\Sigma Yt = 316$

Table 9.5

$$a=\frac{\sum Y}{n}=\frac{316}{7}=45.143\ ;\ b=\frac{\sum XY}{\sum X^2}=\frac{29}{28}=1.036$$

Therefore, the required equation of the straight line trend is given by

Y = a+bX;

Y = 45.143 + 1.036 (x-2003)

The trend values can be obtained by

When X = 2000 , Yt = 45.143 + 1.036(2000–2003) = 42.035

When X = 2001, Yt = 45.143 + 1.036(2001–2003) = 43.071,