

DSA0413 - Fundamentals of Data Science - Lab Questions

1. **Scenario:** You are working on a project that involves analyzing student performance data for a class of 10 students. The data is stored in a NumPy array named `student_scores`, where each row represents a student and each column represents a different subject. The subjects are arranged in the following order: Math, Science, English, and History. Your task is to calculate the average score for each subject and identify the subject with the highest average score.

Question: How would you use NumPy arrays to calculate the average score for each subject and determine the subject with the highest average score? Assume 4x4 matrix that stores marks of each student in given order.

```
import numpy as np

student_scores=np.array([

    [90,88,91,92],

    [76,88,80,90],

    [77,88,90,78],

    [78,88,87,89]])

avg=np.mean(student_scores,axis=0)

high=np.argmax(avg)

highest= ['Math', 'Science', 'English', 'History'][high]

print(avg)

print(highest)
```

2. **Scenario:** You are a data analyst working for a company that sells products online. You have been tasked with analyzing the sales data for the past month. The data is stored in a NumPy array.

Question: How would you find the average price of all the products sold in the past month? Assume 3x3 matrix with each row representing the sales for a different product

```
import numpy as np

sales_data = np.array([

    [100, 120, 150],

    [80, 90, 110],
```

```
[130, 100, 120]

])

average_price = np.mean(sales_data)

print( average_price)
```

3. **Scenario:** You are working on a project that involves analyzing a dataset containing information about houses in a neighborhood. The dataset is stored in a CSV file, and you have imported it into a NumPy array named `house_data`. Each row of the array represents a house, and the columns contain various features such as the number of bedrooms, square footage, and sale price.

Question: Using NumPy arrays and operations, how would you find the average sale price of houses with more than four bedrooms in the neighborhood?

```
import numpy as np

import pandas as pd

house_data = np.array([

    [4, 2000, 350000],

    [3, 1800, 300000],

    [5, 2200, 400000],

    [4, 2400, 380000],

    [6, 2800, 450000]

])

bedroom_column = 0

sale_price_column = -1

houses_more_than_four_bedrooms = house_data[house_data[:, bedroom_column] > 4,
sale_price_column]

average_sale_price = np.mean(houses_more_than_four_bedrooms)

print("Average sale price of houses with more than four bedrooms:", average_sale_price)
```

4. **Scenario:** You are working on a project that involves analyzing the sales performance of a company over the past four quarters. The quarterly sales data is stored in a NumPy array named `sales_data`, where each element represents the sales amount for a specific quarter.

Your task is to calculate the total sales for the year and determine the percentage increase in sales from the first quarter to the fourth quarter.

Question: Using NumPy arrays and arithmetic operations calculate the total sales for the year and determine the percentage increase in sales from the first quarter to the fourth quarter?

```
import numpy as np

data=np.array([10000,20000,25000,30000])

total=np.sum(data)

print(total)

q1=data[0]

q4=data[-1]

per=((q4-q1)/q1)*100

print(per)
```

5. **Scenario:** You are a data analyst working for a car manufacturing company. As part of your analysis, you have a dataset containing information about the fuel efficiency of different car models. The dataset is stored in a NumPy array named `fuel_efficiency`, where each element represents the fuel efficiency (in miles per gallon) of a specific car model. Your task is to calculate the average fuel efficiency and determine the percentage improvement in fuel efficiency between two car models.

Question: How would you use NumPy arrays and arithmetic operations to calculate the average fuel efficiency and determine the percentage improvement in fuel efficiency between two car models?

```
import numpy as np

fuel_efficiency=np.array([20,30,40,23])

avg=np.mean(fuel_efficiency)

print(avg)

car1=fuel_efficiency[0]

car2=fuel_efficiency[2]

per=((car2-car1)/car1)*100

print(per)
```

6. **Scenario:** You are a cashier at a grocery store and need to calculate the total cost of a customer's purchase, including applicable discounts and taxes. You have the item prices and quantities in separate lists, and the discount and tax rates are given as percentages. Your task is to calculate the total cost for the customer.

Question: Use arithmetic operations to calculate the total cost of a customer's purchase, including discounts and taxes, given the item prices, quantities, discount rate, and tax rate?

```
items_price=[6,5,10,2]
```

```
quantities=[3,2,4,1]
```

```
discount=10
```

```
tax=8
```

```
total=sum(price*quantity for price,quantity in zip(items_price,quantities))
```

```
discount_amt=(discount/100)*total
```

```
total_after_dis=total-discount_amt
```

```
tax_amt=(tax/100)*total_after_dis
```

```
total_after_tax=total_after_dis+tax_amt
```

```
print(total_after_tax)
```

```
print(total_after_dis)
```

7. **Scenario:** You are working as a data analyst for an e-commerce company. You have been given a dataset containing information about customer orders, stored in a Pandas DataFrame named `order_data`. The DataFrame has columns for customer ID, order date, product name, and order quantity. Your task is to analyze the data and answer specific questions about the orders.

Question: Using Pandas DataFrame operations, how would you find the following information from the `order_data` DataFrame:

1. The total number of orders made by each customer.
2. The average order quantity for each product.
3. The earliest and latest order dates in the dataset.

```
import pandas as pd
```

```
order_data = pd.DataFrame({
```

```
    'CustomerID': [1, 2, 1, 3, 2],
```

```
    'OrderDate': ['2022-01-01', '2022-01-02', '2022-01-01', '2022-01-03', '2022-01-02'],
```

```

'ProductName': ['ProductA', 'ProductB', 'ProductA', 'ProductC', 'ProductB'],
'OrderQuantity': [3, 5, 2, 1, 4]

})

total=order_data.groupby('CustomerID')['OrderDate'].count()

avg=order_data.groupby('ProductName')['OrderQuantity'].mean()

earliest=order_data['OrderDate'].min()

latest=order_data['OrderDate'].max()

print(total)

print(avg)

print(earliest)

print(latest)

```

8. **Scenario:** You are a data scientist working for a company that sells products online. You have been tasked with analyzing the sales data for the past month. The data is stored in a Pandas data frame.

Question: How would you find the top 5 products that have been sold the most in the past month?

```

import pandas as pd

sales_data = pd.DataFrame({

    'ProductID': [1, 2, 1, 3, 2, 3, 4, 5, 4, 5],

    'ProductName': ['ProductA', 'ProductB', 'ProductA', 'ProductC', 'ProductB', 'ProductC',
'ProductD', 'ProductE', 'ProductD', 'ProductE'],

    'QuantitySold': [10, 15, 8, 20, 12, 18, 5, 25, 6, 22]

})

product_Sales=sales_data.groupby('ProductName')['QuantitySold'].sum()

sorted_sales=product_Sales.sort_values(ascending=False)

five=sorted_sales.head(5)

print(five)

```

9. **Scenario:** You work for a real estate agency and have been given a dataset containing information about properties for sale. The dataset is stored in a Pandas DataFrame named `property_data`. The DataFrame has columns for property ID, location, number of bedrooms, area in square feet, and listing price. Your task is to analyze the data and answer specific questions about the properties.

Question: Using Pandas DataFrame operations, how would you find the following information from the `property_data` DataFrame:

1. The average listing price of properties in each location.
2. The number of properties with more than four bedrooms.
3. The property with the largest area.

```
import pandas as pd
```

```
property_data = pd.DataFrame({  
    'property_ID': [1, 2, 3, 4],  
    'location': ['A', 'B', 'A', 'B'],  
    'noofbedrooms': [5, 4, 6, 4],  
    'areasqft': [1200, 1000, 1300, 1400],  
    'listingprice': [50000, 35000, 55000, 40000]  
})
```

```
# 1. Average listing price of properties in each location
```

```
avg_listing_price_per_location =  
property_data.groupby('location')['listingprice'].mean()  
  
print("Average listing price of properties in each location:")  
  
print(avg_listing_price_per_location)
```

```
# 2. Number of properties with more than four bedrooms
```

```
num_properties_more_than_four_bedrooms =  
property_data[property_data['noofbedrooms'] > 4].shape[0]
```

```
print("\nNumber of properties with more than four bedrooms:",  
num_properties_more_than_four_bedrooms)
```

```
# 3. Property with the largest area
```

```
property_with_largest_area = property_data.loc[property_data['areasqft'].idxmax()]
```

```
print("\nProperty with the largest area:")
```

```
print(property_with_largest_area)
```

10. **Scenario:** You are working on a data visualization project and need to create basic plots using Matplotlib. You have a dataset containing the monthly sales data for a company, including the month and corresponding sales values. Your task is to develop a Python program that generates line plots and bar plots to visualize the sales data.

Question:

1. How would you develop a Python program to create a line plot of the monthly sales data?
2. How would you develop a Python program to create a bar plot of the monthly sales data?

```
import matplotlib.pyplot as plt
```

```
import pandas as pd
```

```
sales_data = pd.DataFrame({
```

```
    'Month': ['Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun'],
```

```
    'Sales': [10000, 12000, 8000, 15000, 13000, 16000]
```

```
})
```

```
plt.plot(sales_data['Month'], sales_data['Sales'], marker='o', color='b', label='Monthly Sales')
```

```
plt.title('Monthly Sales Data')
```

```
plt.xlabel('Month')
```

```
plt.ylabel('Sales')
```

```
plt.show()
```

```
plt.bar(sales_data['Month'], sales_data['Sales'], color='c', label='Monthly Sales')
```

```
plt.title('Monthly Sales Data')
```

```
plt.xlabel('Month')
```

```
plt.ylabel('Sales')
```

```
plt.show()
```

11. **Scenario :** You are a data scientist working for a company that sells products online. You have been tasked with creating a simple plot to show the sales of a product over time.

Question:

1. Write code to create a simple line plot in Python using Matplotlib to predict sales happened in a month?
2. Write code to create a scatter plot in Python using Matplotlib to predict sales happened in a month?
3. Develop a Python program to create a bar plot of the monthly sales data.

```
import matplotlib.pyplot as plt
```

```
import pandas as pd
```

```
sales_data = pd.DataFrame({  
    'Month': ['Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun'],  
    'Sales': [10000, 12000, 8000, 15000, 13000, 16000]  
})
```

```
plt.plot(sales_data['Month'], sales_data['Sales'], marker='o', color='b', label='Monthly Sales')
```

```
plt.title('Monthly Sales Data - Line Plot')
```

```
plt.xlabel('Month')
```

```
plt.ylabel('Sales')
```

```
plt.show()
```

```
plt.scatter(sales_data['Month'], sales_data['Sales'], color='r', label='Monthly Sales')
```

```
plt.title('Monthly Sales Data - Scatter Plot')
```

```
plt.xlabel('Month')
```

```
plt.ylabel('Sales')
```

```
plt.show()
```



```
plt.bar(sales_data['Month'], sales_data['Sales'], color='g', label='Monthly Sales')

plt.title('Monthly Sales Data - Bar Plot')

plt.xlabel('Month')

plt.ylabel('Sales')

plt.show()
```

12. **Scenario:** You are working on a data analysis project that involves analyzing the monthly temperature and rainfall data for a city. You have a dataset containing the monthly temperature and rainfall values for each month of a year. Your task is to develop a Python program that generates line plots and scatter plots to visualize the temperature and rainfall data.

Question:

1. Develop a Python program to create a line plot of the monthly temperature data.
2. Develop a Python program to create a scatter plot of the monthly rainfall data.

```
import matplotlib.pyplot as plt

import pandas as pd

weather_data = pd.DataFrame({

    'Month': ['Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun', 'Jul', 'Aug', 'Sep', 'Oct', 'Nov', 'Dec'],

    'Temperature': [10, 12, 15, 18, 22, 25, 27, 26, 23, 20, 15, 12],

    'Rainfall': [50, 40, 30, 20, 10, 5, 8, 12, 15, 25, 35, 45]

})

plt.plot(weather_data['Month'], weather_data['Temperature'], marker='o', color='b',
label='Temperature')

plt.title('Monthly Temperature Data - Line Plot')

plt.xlabel('Month')

plt.ylabel('Temperature (°C)')

plt.show()

plt.scatter(weather_data['Month'], weather_data['Rainfall'], color='r', label='Rainfall')

plt.title('Monthly Rainfall Data - Scatter Plot')
```

```
plt.xlabel('Month')

plt.ylabel('Rainfall (mm)')

plt.show()
```

10.01.2024

13.Scenario: You are a data analyst working for a company that sells products online. You have been tasked with analyzing the sales data for the past month. The data is stored in a Pandas data frame.

Question: Develop a code in python to find the frequency distribution of the ages of the customers who have made a purchase in the past month.

```
import pandas as pd

sales_data = pd.DataFrame({

    'CustomerID': [1, 2, 3, 4, 5],

    'Name': ['Customer1', 'Customer2', 'Customer3', 'Customer4', 'Customer5'],

    'Age': [25, 30, 28, 35, 40],

    'PurchaseAmount': [100, 150, 120, 200, 180]

})

age_frequency = sales_data['Age'].value_counts().sort_index()

print("Frequency distribution of customer ages:")

print(age_frequency)
```

14. Scenario: You are a data analyst working for a social media platform. As part of your analysis, you have a dataset containing user interaction data, including the number of likes received by each post. Your task is to develop a Python program that calculates the frequency distribution of likes among the posts.

Question: Develop a Python program to calculate the frequency distribution of likes among the posts?

```
import pandas as pd

interaction_data = pd.DataFrame({

    'PostID': [1, 2, 3, 4, 5, 6, 7, 8, 9, 10],
```

```

'Likes': [50, 30, 20, 50, 10, 30, 40, 50, 20, 30]

})

likes_frequency = interaction_data['Likes'].value_counts().sort_index()

print("Frequency distribution of likes among the posts:")

print(likes_frequency)

```

15. **Scenario:** You are working on a project that involves analyzing customer reviews for a product. You have a dataset containing customer reviews, and your task is to develop a Python program that calculates the frequency distribution of words in the reviews.

Question: Develop a Python program to calculate the frequency distribution of words in the customer reviews dataset?

```

import pandas as pd

data = {'customerID': [1, 2, 3, 4, 5],

        'reviews': ['good','bad','good','ok','bad']}

df = pd.DataFrame(data)

distribution = df['reviews'].value_counts().sort_index()

print("review ")

print(distribution)

```

16. **Scenario:** You are a data analyst working for a marketing research company. Your team has collected a large dataset containing customer feedback from various social media platforms. The dataset consists of thousands of text entries, and your task is to develop a Python program to analyze the frequency distribution of words in this dataset. Your program should be able to perform the following tasks:

- Load the dataset from a CSV file (data.csv) containing a single column named "feedback" with each row representing a customer comment.
- Preprocess the text data by removing punctuation, converting all text to lowercase, and eliminating any stop words (common words like "the," "and," "is," etc. that don't carry significant meaning).
- Calculate the frequency distribution of words in the preprocessed dataset.
- Display the top N most frequent words and their corresponding frequencies, where N is provided as user input.
- Plot a bar graph to visualize the top N most frequent words and their frequencies.

Question: Create a Python program that fulfills these requirements and helps your team gain insights from the customer feedback data.

17. Suppose a hospital tested the age and body fat data for 18 randomly selected adults with the following result.

<i>age</i>	23	23	27	27	39	41	47	49	50
<i>%fat</i>	9.5	26.5	7.8	17.8	31.4	25.9	27.4	27.2	31.2
<i>age</i>	52	54	54	56	57	58	58	60	61
<i>%fat</i>	34.6	42.5	28.8	33.4	30.2	34.1	32.9	41.2	35.7

Question:

- Calculate the mean, median and standard deviation of age and %fat using Pandas.
- Draw the boxplots for age and %fat.
- Draw a scatter plot and a q-q plot based on these two variables

```
import pandas as pd
```

```
import matplotlib.pyplot as plt
```

```
import scipy.stats as stats
```

```
age = [23, 23, 27, 27, 39, 41, 47, 49, 50, 52, 54, 54, 56, 57, 58, 58, 60, 61]
```

```
percent_fat = [9.5, 26.5, 7.8, 17.8, 31.4, 25.9, 27.4, 27.2, 31.2, 34.6, 42.5, 28.8, 33.4, 30.2, 34.1, 32.9, 41.2, 35.7]
```

```
data = pd.DataFrame({'Age': age, '%Fat': percent_fat})
```

```
mean_age = data['Age'].mean()
```

```
median_age = data['Age'].median()
```

```
std_age = data['Age'].std()
```

```
mean_fat = data['%Fat'].mean()
```

```
median_fat = data['%Fat'].median()
```

```
std_fat = data['%Fat'].std()
```

```
print("Age: Mean =", mean_age, "Median =", median_age, "Standard Deviation =", std_age)
```

```
print("%Fat: Mean =", mean_fat, "Median =", median_fat, "Standard Deviation =", std_fat)
```

```
data.boxplot(column=['Age', '%Fat'])
```

```
plt.title("Boxplots for Age and %Fat")
```

```
plt.show()
```

```

plt.scatter(data['Age'], data['%Fat'])

plt.xlabel('Age')

plt.ylabel('%Fat')

plt.title('Scatter Plot')

plt.show()

plt.figure()

stats.probplot(data['%Fat'], dist="norm", plot=plt)

plt.title('Q-Q Plot for %Fat')

plt.show()

```

18. a. **Scenario:** You are working with a dataset representing the daily sales of a product over the past month. Calculate the variance of the daily sales to understand how much the sales figures deviate from the mean.

```

import numpy as np

daily_sales = np.array([100, 120, 90, 110, 130, 95, 105, 115, 125, 100, 110, 120, 130, 95, 105])

mean_sales = np.mean(daily_sales)

variance_sales = np.var(daily_sales)

print("Variance of Daily Sales:", variance_sales)

```

b. **Scenario:** In a study, you have collected data on the hours spent studying (variable x) and the corresponding exam scores (variable y) for a group of students. Calculate the covariance between study hours and exam scores to explore if there is a relationship.

```

import numpy as np

study_hours = np.array([2, 3, 1, 4, 5, 2, 3, 4, 5, 1])

exam_scores = np.array([65, 70, 60, 75, 80, 65, 70, 75, 80, 60])

covariance_study_exam = np.cov(study_hours, exam_scores)[0, 1]

print("Covariance between Study Hours and Exam Scores:", covariance_study_exam)

```

19. **Scenario:** You are dealing with a dataset containing the monthly expenses of different departments in a company. Use NumPy functions to efficiently calculate both the variance and covariance matrix of these expenses.

```

import numpy as np

```

```

monthly_expenses = np.array([
    [50000, 55000, 60000, 52000, 58000],
    [40000, 42000, 38000, 41000, 45000],
    [30000, 32000, 31000, 30000, 33000]
])

variance_expenses = np.var(monthly_expenses, axis=1)

print("Variance of Monthly Expenses for Each Department:")

print(variance_expenses)

covariance_matrix_expenses = np.cov(monthly_expenses)

print("\nCovariance Matrix of Monthly Expenses:")

print(covariance_matrix_expenses)

```

20. a. **Scenario:** Analyzing the performance of servers, you want to determine the 25th, 50th, and 75th percentiles of response times to identify potential bottlenecks.

```

import numpy as np

response_times = np.array([20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70])

percentiles_25th = np.percentile(response_times, 25)

percentiles_50th = np.percentile(response_times, 50)

percentiles_75th = np.percentile(response_times, 75)

print("25th Percentile:", percentiles_25th)

print("50th Percentile (Median):", percentiles_50th)

print("75th Percentile:", percentiles_75th)

```

b. **Scenario:** In a medical study, you have collected data on patients' recovery times after a procedure. Calculate the 10th, 50th, and 90th percentiles to understand the distribution of recovery times.

```

import numpy as np

recovery_times = np.array([10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60])

percentiles_10th = np.percentile(recovery_times, 10)

```

```
percentiles_50th = np.percentile(recovery_times, 50)
```

```
percentiles_90th = np.percentile(recovery_times, 90)
```

```
print("10th Percentile:", percentiles_10th)
```

```
print("50th Percentile (Median):", percentiles_50th)
```

```
print("90th Percentile:", percentiles_90th)
```

22. **Scenario:** You are investigating a dataset representing the daily temperatures in a city. Calculate the variance and identify potential outliers that may indicate unusual weather conditions.

```
import numpy as np
```

```
daily_temperatures = np.array([25.5, 26.0, 24.8, 25.2, 25.7, 26.5, 27.0, 28.5, 25.0, 25.5, 26.8, 29.5, 24.0, 26.5, 27.8])
```

```
variance_temperatures = np.var(daily_temperatures)
```

```
print("Variance of Daily Temperatures:", variance_temperatures)
```

```
z_scores = (daily_temperatures - np.mean(daily_temperatures)) / np.std(daily_temperatures)
```

```
z_score_threshold = 2
```

```
potential_outliers = np.abs(z_scores) > z_score_threshold
```

```
print("\nPotential Outliers:")
```

```
print(daily_temperatures[potential_outliers])
```

13.01.2024

23. **Scenario:** You work as a data scientist for a marketing agency, and one of your clients is a large e-commerce company. The company wants to understand the purchasing behavior of its customers and segment them into different groups based on their buying patterns. The e-commerce company has provided you with transaction data, including customer IDs, the total amount spent in each transaction, and the number of items purchased.

Question: Build a clustering model using the K-Means algorithm to group customers based on their spending and purchase behavior and visualize the clusters using scatter plots or other appropriate visualizations to gain insights into customer distribution and distinguish different segments.

24. **Scenario:** You work as a data scientist for a retail company that operates multiple stores. The company is interested in segmenting its customers based on their purchasing behavior to better understand their preferences and tailor marketing strategies accordingly. To achieve this, your team has collected transaction data from different stores, which includes customer IDs, the total amount spent in each transaction, and the frequency of visits.

Question: Your task is to build a clustering model using the K-Means algorithm to group customers into distinct segments based on their spending patterns.

25. **Scenario:** Suppose you are working as a data scientist for a medical research organization. Your team has collected data on patients with a certain medical condition and their treatment outcomes. The dataset includes various features such as age, gender, blood pressure, cholesterol levels, and whether the patient responded positively ("Good") or negatively ("Bad") to the treatment. The organization wants to use this model to identify potential candidates who are likely to respond positively to the treatment and improve their medical approach.

Question: Your task is to build a classification model using the KNN algorithm to predict the treatment outcome ("Good" or "Bad") for new patients based on their features. Evaluate the model's performance using accuracy, precision, recall, and F1-score. Make predictions on the test set and display the results.

26. **Scenario:** You work as a data scientist for a real estate company. The company has collected data on various houses, including features such as the size of the house, number of bedrooms, location, and other relevant attributes. The marketing team wants to build a predictive model to estimate the price of houses based on their features. They believe that linear regression modeling can be an effective approach for this task.

Question: Your task is to write a Python program to perform bivariate analysis and build a linear regression model to predict house prices based on a selected feature (e.g., house size) from the dataset. Additionally, you need to evaluate the model's performance to ensure its accuracy and reliability.

27. **Question:** Classification and Regression Trees (CART) for Car Price Prediction

You are working for a car dealership, and you want to predict the price of used cars based on various features such as the car's mileage, age, brand, and engine type. You have collected a dataset of used cars with their respective prices.

Write a Python program that loads the car dataset and allows the user to input the features of a new car they want to sell. The program should use the Classification and Regression Trees (CART) algorithm from scikit-learn to predict the price of the new car based on the input features.

The CART algorithm will create a tree-based model that will split the data into subsets based on the chosen features and their values, leading to a decision path that eventually predicts the price of the car. The program should output the predicted price and display the decision path (the sequence of conditions leading to the prediction) for the new car.

28. **Scenario:** You work as a data scientist for an e-commerce company that sells a wide range of products online. The company collects vast amounts of data about its customers, including their purchase history, browsing behavior, demographics, and more. The marketing team wants to understand their customer base better and improve their targeted marketing strategies. They have asked you to perform customer segmentation using clustering techniques to identify distinct groups of customers with similar characteristics.

Question: Your task is to use Python and clustering algorithms to segment the customers into different groups based on their behavior and characteristics. The marketing team will use these segments to tailor their marketing campaigns and promotions effectively.

29. **Question:** K-Nearest Neighbors (KNN) Classifier

You are working on a classification problem to predict whether a patient has a certain medical condition or not based on their symptoms. You have collected a dataset of patients with labeled data (0 for no condition, 1 for the condition) and various symptom features.

Write a Python program that allows the user to input the features of a new patient and the value of k (number of neighbors). The program should use the KNN classifier from the scikit-learn library to predict whether the patient has the medical condition or not based on the input features.

30. **Question :** Decision Tree for Iris Flower Classification

You are analyzing the famous Iris flower dataset to classify iris flowers into three species based on their sepal and petal dimensions. You want to use a Decision Tree classifier to accomplish this task.

Write a Python program that loads the Iris dataset from scikit-learn, and allows the user to input the sepal length, sepal width, petal length, and petal width of a new flower. The program should then use the Decision Tree classifier to predict the species of the new flower.

31. **Question :** Linear Regression for Housing Price Prediction

You are a real estate analyst trying to predict housing prices based on various features of the houses, such as area, number of bedrooms, and location. You have collected a dataset of houses with their respective prices.

Write a Python program that allows the user to input the features (area, number of bedrooms, etc.) of a new house. The program should use linear regression from scikit-learn to predict the price of the new house based on the input features.

32. **Question:** Logistic Regression for Customer Churn Prediction

You are working for a telecommunications company, and you want to predict whether a customer will churn (leave the company) based on their usage patterns and demographic data. You have collected a dataset of past customers with their churn status (0 for not churned, 1 for churned) and various features.

Write a Python program that allows the user to input the features (e.g., usage minutes, contract duration) of a new customer. The program should use logistic regression from scikit-learn to predict whether the new customer will churn or not based on the input features.

33. **Question:** K-Means Clustering for Customer Segmentation

You are working for an e-commerce company and want to segment your customers into distinct groups based on their purchasing behavior. You have collected a dataset of customer data with various shopping-related features.

Write a Python program that allows the user to input the shopping-related features of a new customer. The program should use K-Means clustering from scikit-learn to assign the new customer to one of the existing segments based on the input features.

34.Question: Evaluation Metrics for Model Performance

You have trained a machine learning model on a dataset, and now you want to evaluate its performance using various metrics.

Write a Python program that loads a dataset and trained model from scikit-learn. The program should ask the user to input the names of the features and the target variable they want to use for evaluation. The program should then calculate and display common evaluation metrics such as accuracy, precision, recall, and F1-score for the model's predictions on the test data.

35. Question: You are a data scientist working for an e-commerce company. The marketing team has conducted an A/B test to evaluate the effectiveness of two different website designs (A and B) in terms of conversion rate. They randomly divided the website visitors into two groups, with one group experiencing design A and the other experiencing design B. After a week of data collection, you now have the conversion rate data for both groups. You want to determine whether there is a statistically significant difference in the mean conversion rates between the two website designs.

Question:

"Based on the data collected from the A/B test, is there a statistically significant difference in the mean conversion rates between website design A and website design B?"