Tech Saksham

Capstone Project Report

“E-Commerce Analysis”

“University College Of Engineering – Arni”

NM ID	NAME
au513321105033	SATHYA M

Ramar Bose

Sr. AI Master Trainer

ABSTRACT

The project provides an analysis of sales data from an e-commerce dataset to understand key sales trends and patterns for the year 2019. The data consists of various attributes such as order ID, quantity ordered, price, order date, purchase address, product, and others. The analysis includes data preprocessing and cleaning to ensure accurate and relevant insights. The analysis utilizes visualizations such as bar charts and line graphs to present data in an accessible manner. This helps to identify trends and patterns that can guide decision-making for sales strategies, marketing campaigns, and inventory management. Overall, the project offers valuable insights into the e-commerce sales data for 2019 and provides potential opportunities for business improvement.

1. Problem stat.
2. Data collection
3. Existing solution
4. Proposed solution with used models
5. Result

INDEX

CHAPTER 1

INTRODUCTION

1.1 Problem Statement

- The retail business needs to optimize its sales strategies and inventory management to enhance revenue and customer satisfaction.
- By analyzing the sales data, including trends in sales, product combinations, and city-specific and hourly sales distributions, the business can make data-driven decisions.

1.2 Proposed Solution

(I) - Load the CSV files from the specified folder and combine them into a single DataFrame.

(II) - Clean the data by excluding rows with headers and missing values.

(III) - Extract features such as year, month, hour, and city from the data.

(IV) - Analyze the data

(V) - Provide data-driven insights to inform decision-making for sales strategies and inventory management.

1.3 Feature

Data Preprocessing and Cleaning:

- Loading, combining, and filtering CSV files to obtain a comprehensive dataset.
- Removing null values and excluding header rows to clean the data.

Summary Statistics:

- Calculating total orders, products sold, and total sales for the year 2019.
- Providing an overview of sales performance.

Visualizations:

- Plotting monthly sales trends to identify seasonal variations.
- Visualizing hourly sales trends to identify peak sales hours.

1.4 Advantages

The project offers several advantages for businesses and data analysts working with e-commerce sales data:

Comprehensive Insights: Provides a detailed analysis of sales data, including total orders, products sold, and total sales, which can inform strategic decisions and performance evaluation.

Data-Driven Decision-Making: Offers data visualizations and statistical analyses that enable businesses to make informed decisions based on real data, such as optimizing inventory management and adjusting sales strategies.

Identification of Trends and Patterns: Visualizations such as monthly and hourly sales charts reveal trends and patterns in sales, aiding in planning and forecasting.

Location-Based Analysis: City-specific sales data helps identify high-performing locations, which can guide targeted marketing efforts and regional sales strategies.

1.5 Scope

The scope of the project can be defined in terms of the extent of data analysis, potential use cases, and avenues for further exploration and application. The project covers the following areas:

Data Coverage: The project focuses on sales data from an e-commerce dataset for the year 2019.The analysis includes order data, such as order ID, quantity ordered, price each, order date, purchase address, product details, and more.

Time Frame: The analysis is limited to the year 2019, but the methods and techniques can be applied to data from other years for longitudinal analysis.

Feature Extraction and Analysis:Extracting key features such as sales, cities, year, month, hour, and minute from order data.Investigating patterns and trends in sales by time, location, and product.

CHAPTER 2

SERVICES AND TOOLS REQUIRED

2.1 LR - Exiting Models

Standard Logistic Regression: The basic form of logistic regression, available in many machine learning libraries (e.g., scikit-learn, TensorFlow, PyTorch).Suitable for binary classification tasks.

Regularized Logistic Regression: Variants of logistic regression that include regularization techniques such as L1 (Lasso) and L2 (Ridge) regularization.Helps to prevent overfitting by penalizing large coefficients in the model.

Multinomial Logistic Regression: Extends logistic regression to handle multiple classes (multiclass classification).Predicts the probability of each class and can be implemented using one-vs-rest or softmax approaches.

Stochastic Gradient Descent (SGD) Logistic Regression: Uses stochastic gradient descent as the optimization technique for training the logistic regression model.Useful for handling large datasets and can be found in libraries such as scikit-learn.

2.1 Required – System config | Cloud computing

Compute Resources: Virtual Machines: Choose virtual machines with suitable CPU and memory configurations to handle data processing and analysis tasks.

Managed Services: Consider using managed services for data processing (e.g., AWS Glue, Google Dataflow) to simplify infrastructure management.

Storage:

- **Cloud Storage**: Use cloud storage services such as Amazon S3, Google Cloud Storage, or Azure Blob Storage for storing datasets.

- **Data Warehousing:** Consider using a managed data warehouse service (e.g., Amazon Redshift, Google BigQuery, or Azure Synapse Analytics) for efficient data querying and analysis.

Data Processing:

- **Batch Processing**: If the dataset is large, you might consider using distributed data processing frameworks such as Apache Spark on cloud platforms (e.g., AWS EMR, Google Dataproc).
- **Serverless Functions:** Consider using serverless functions (e.g., AWS Lambda, Google Cloud Functions, Azure Functions) for event-driven data processing and analysis tasks.

2.1 Services Used

Data Processing:

- **Data Factory:** Managed data integration service for data transformation and preparation.
- **HDInsight:** Managed service for running big data processing tasks with Apache Spark and other frameworks.

Machine Learning:

- **Azure Machine Learning:** Managed service for building and deploying models.

Analytics:

- **Azure Synapse Analytics:** Data warehousing and analytics service for efficient querying and analysis.

Monitoring and Logging:

- **Azure Monitor:** Monitoring service for tracking resource usage and application performance.

Tools and software used:

- **Python:**A versatile programming language commonly used for data analysis and machine learning**.**

Libraries and packages:

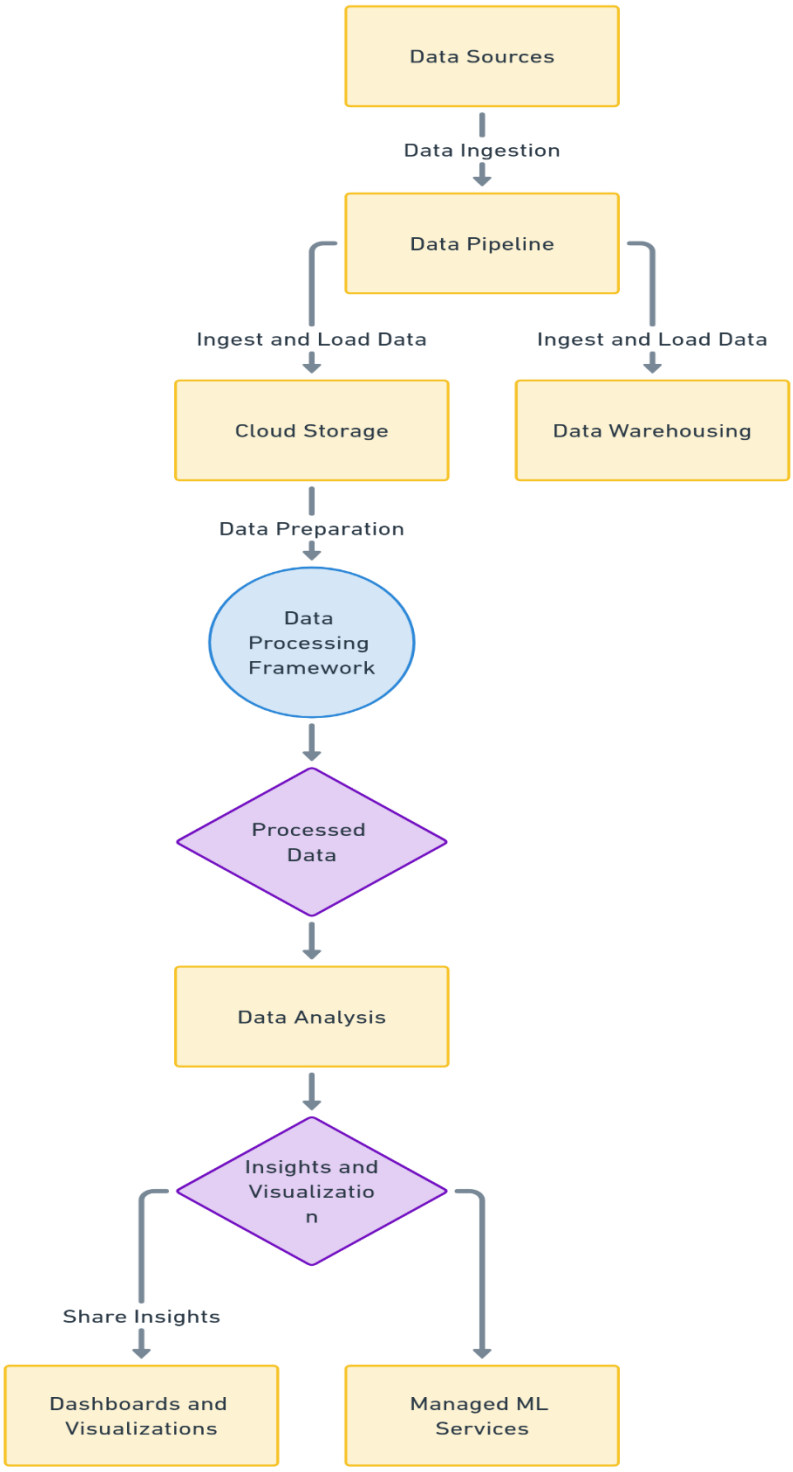
- **Pandas:** For data manipulation, cleaning, and analysis.
- **NumPy**: For numerical computations and array handling.
- **Jupyter Notebooks**:An interactive computing environment for writing and executing code, visualizing data, and documenting the analysis.
- **R:**An alternative language for statistical analysis and data manipulation.
- Popular libraries include dplyr, tidyr, and ggplot2.

CHAPTER 3

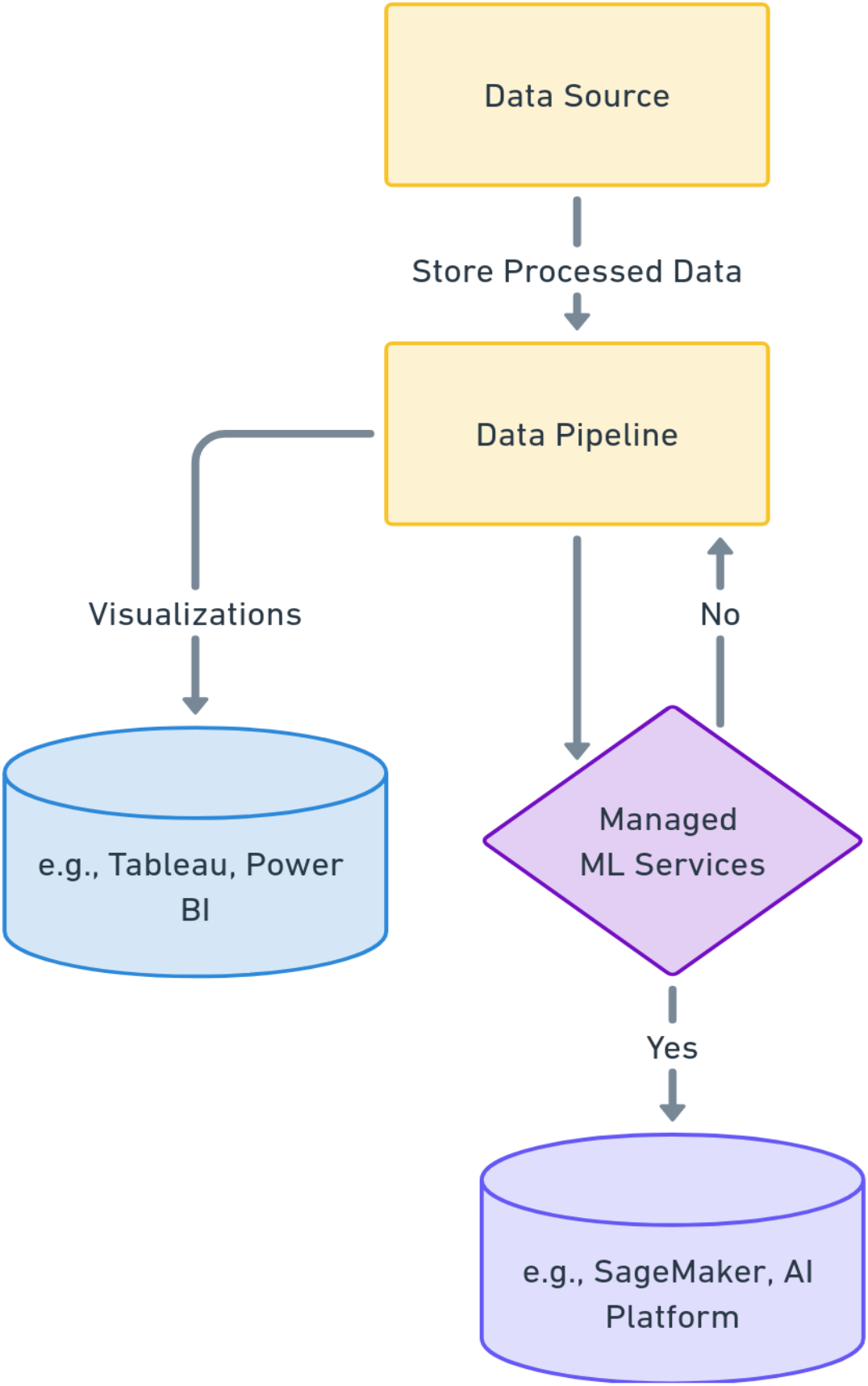
PROJECT ARCHITECTURE

3.1 Architecture

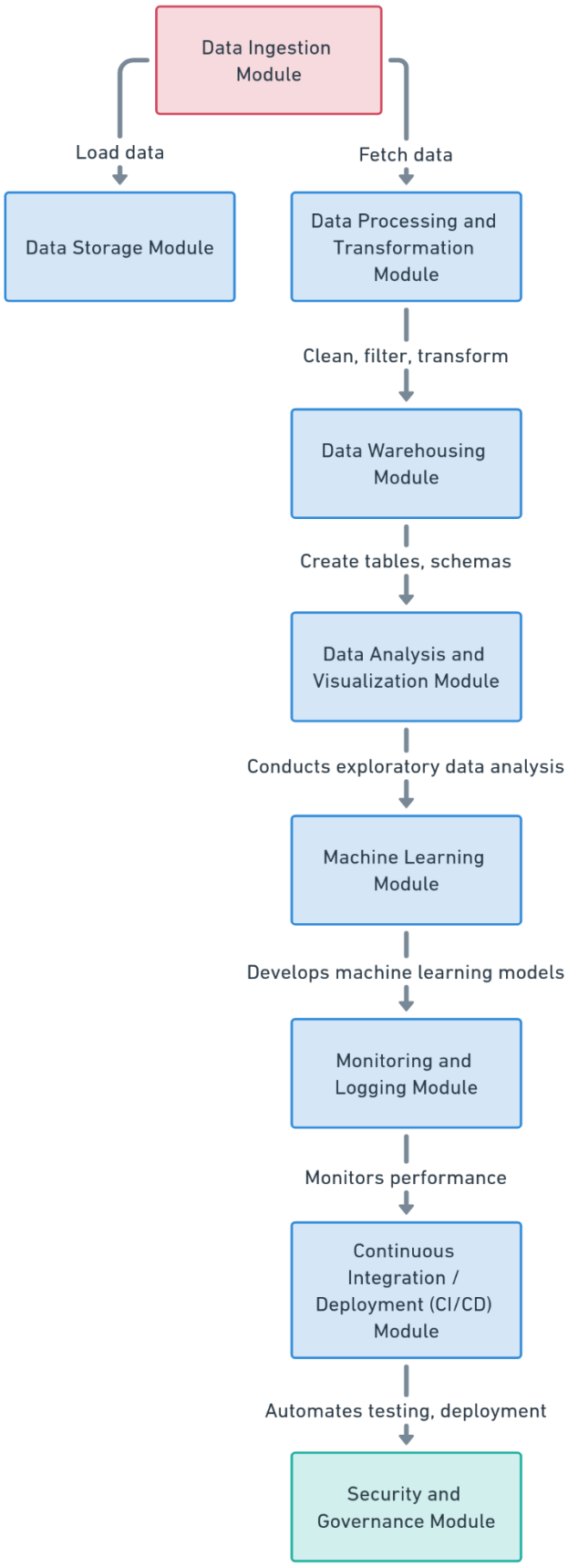
1. System flow diagram



2. Data flow diagram

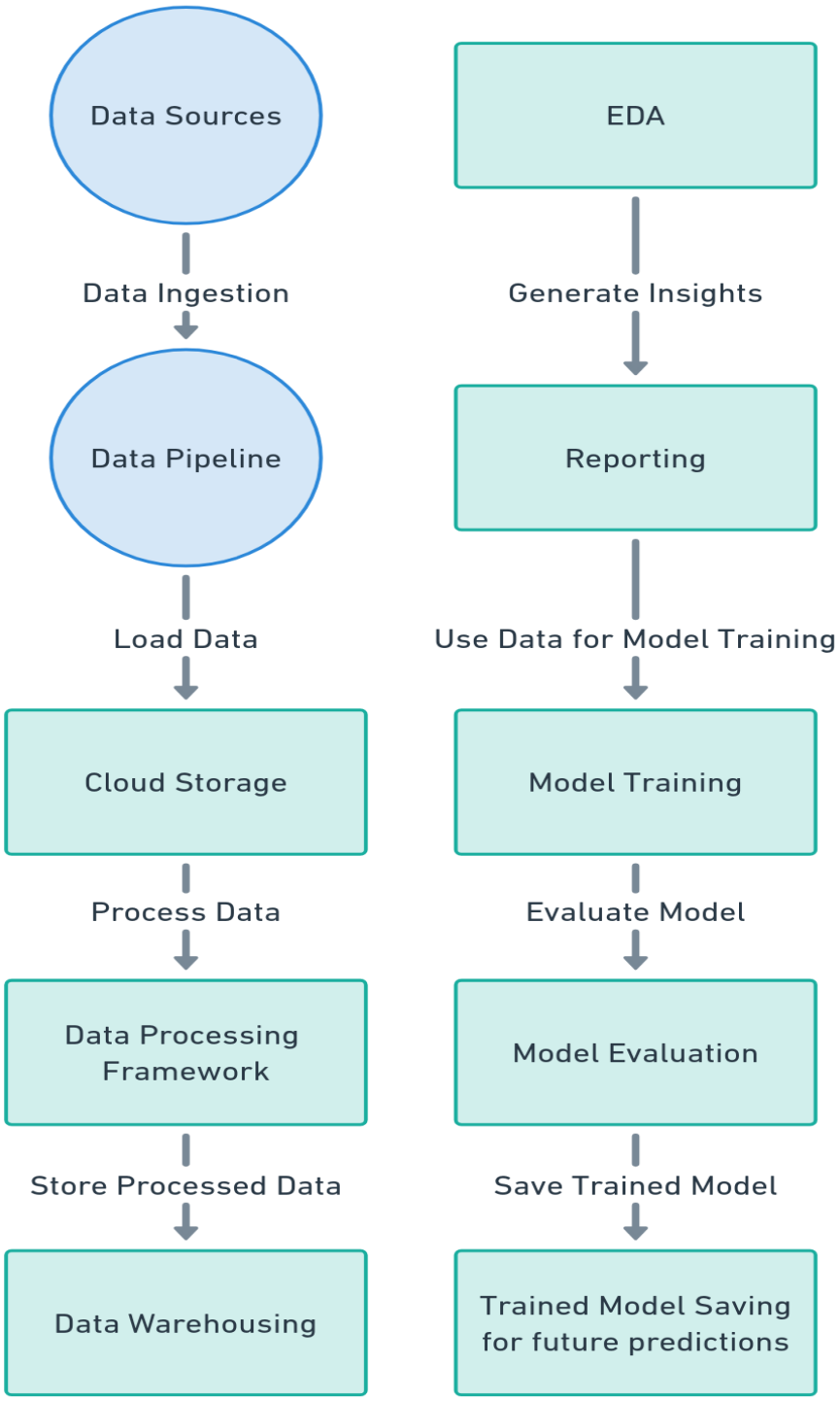


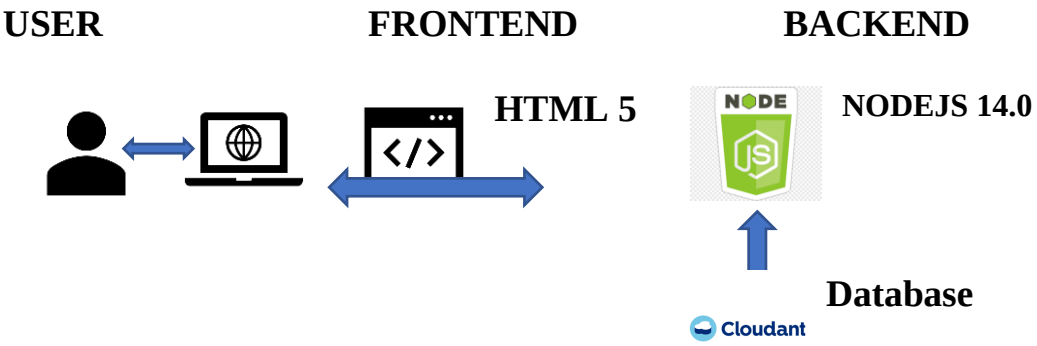
3.Module



4. User interface

- Next Module (EDA) flow diagram
- Training model diagram
- Predicting model's diagram
- Model Performance evaluation models





Here's a high-level architecture for the project:

Data Ingestion and Storage:

- Data is collected from various sources such as online store transactions and customer information.
- The data is ingested and stored in cloud storage solutions like Amazon S3, Google Cloud Storage, or Azure Blob Storage.

Data Processing and Transformation:

- Distributed processing frameworks such as Apache Spark or cloud-native solutions are used to clean, preprocess, and transform the data.
- The processed data is then loaded into a data warehouse such as Amazon Redshift, Google BigQuery, or Azure Synapse Analytics for analysis.

Exploratory Data Analysis (EDA) and Visualization:

- Analysts use tools such as Python, R, or business intelligence platforms to explore and visualize data.
- Visualizations like charts and dashboards communicate findings and insights.

Model Development and Deployment:

- Machine learning models are developed, trained, and validated using the processed data.
- Trained models are deployed as web services or APIs for real-time predictions.

Monitoring and Reporting:

- Models in production are monitored and maintained to ensure consistent performance.
- Insights and reports are shared with stakeholders for strategic decision-making and planning.

CHAPTER 4

MODELING AND PROJECT OUTCOME

Data load:

Code:

```
# Specify the path to the dataset folder
folder_path = '/content/sample_data/dataset'  # Replace
'your_folder_name' with your folder's name
files = [os.path.join(folder_path, f) for f in os.listdir(folder_path)
if f.endswith('.csv')]
# Load and combine CSV files
for file_path in files:
    df = pd.read_csv(file_path)
df.head()
```

Ouput:

	Order ID	Product	Quantity Ordered	Price Each	Order Date	Purchase Address
0	236670	Wired Headphones	2	11.99	08/31/19 22:21	359 Spruce St, Seattle, WA 98101
1	236671	Bose SoundSport Headphones	1	99.99	08/15/19 15:11	492 Ridge St, Dallas, TX 75001
2	236672	iPhone	1	700.0	08/06/19 14:40	149 7th St, Portland, OR 97035
3	236673	AA Batteries (4-pack)	2	3.84	08/29/19 20:59	631 2nd St, Los Angeles, CA 90001
4	236674	AA Batteries (4-pack)	2	3.84	08/15/19 19:53	736 14th St, New York City, NY 10001

EDA – analysis report:

1. Missing Code:

```
    # Check for null values

    df.isna().sum()
    # Check rows with null values

    df[df.isna().any(axis=1)]
```


Output:

	Order ID	Product	Quantity Ordered	Price Each	Order Date	Purchase Address
310	NaN	NaN	NaN	NaN	NaN	NaN
1220	NaN	NaN	NaN	NaN	NaN	NaN
2639	NaN	NaN	NaN	NaN	NaN	NaN
2675	NaN	NaN	NaN	NaN	NaN	NaN
3109	NaN	NaN	NaN	NaN	NaN	NaN
3300	NaN	NaN	NaN	NaN	NaN	NaN
4277	NaN	NaN	NaN	NaN	NaN	NaN
4293	NaN	NaN	NaN	NaN	NaN	NaN
4443	NaN	NaN	NaN	NaN	NaN	NaN
4667	NaN	NaN	NaN	NaN	NaN	NaN
5508	NaN	NaN	NaN	NaN	NaN	NaN
5649	NaN	NaN	NaN	NaN	NaN	NaN
6063	NaN	NaN	NaN	NaN	NaN	NaN
6428	NaN	NaN	NaN	NaN	NaN	NaN
6588	NaN	NaN	NaN	NaN	NaN	NaN
6779	NaN	NaN	NaN	NaN	NaN	NaN
7492	NaN	NaN	NaN	NaN	NaN	NaN
7928	NaN	NaN	NaN	NaN	NaN	NaN
7935	NaN	NaN	NaN	NaN	NaN	NaN
8665	NaN	NaN	NaN	NaN	NaN	NaN
8800	NaN	NaN	NaN	NaN	NaN	NaN
8910	NaN	NaN	NaN	NaN	NaN	NaN
9400	NaN	NaN	NaN	NaN	NaN	NaN

2. Data Visualizations

Code:

```
# Plot monthly sales
df_month = df.groupby('Month')['Sales'].sum()
plt.figure(figsize=(10, 5))
df_month.plot(kind='bar', color=['#0892a5', '#2e9b9b', '#50a290',
'#6fa985', '#8dad7f', '#a9b17e', '#c4b383', '#dbb68f'])
plt.title('Monthly Sales', weight='bold', fontsize=20, pad=20)
plt.ylabel('Sales (in million)')
plt.show()
```

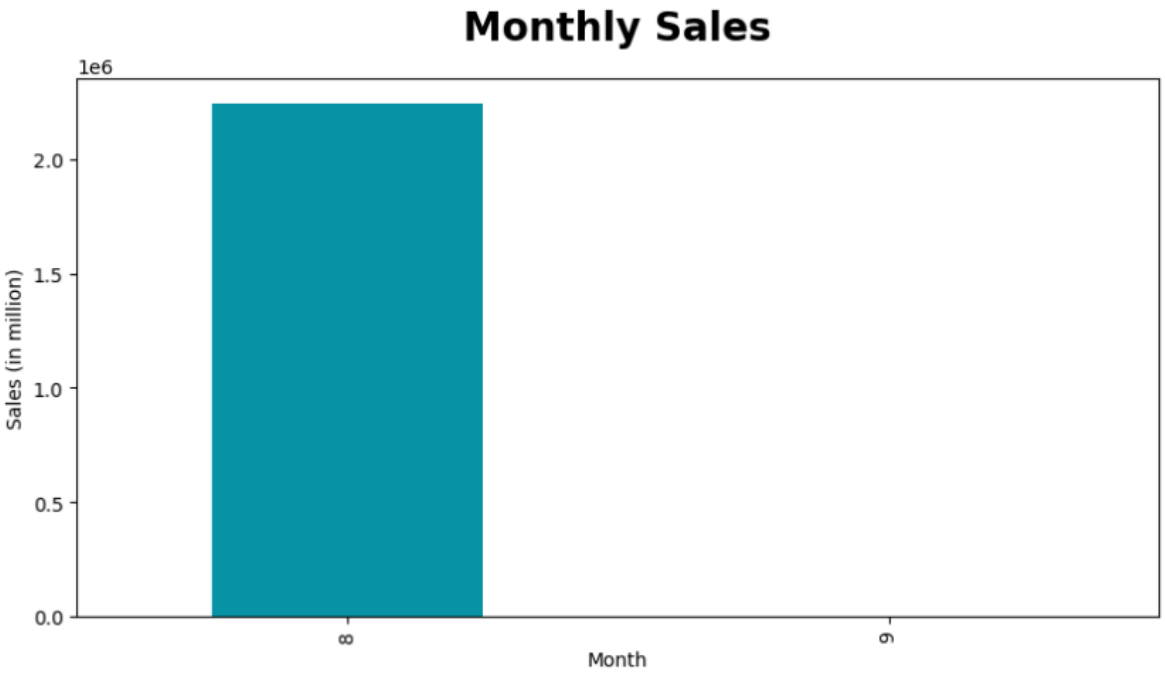
```
# Plot hourly sales
df_hour = df.groupby('Hour')['Quantity Ordered'].count()
plt.figure(figsize=(10, 5))
plt.plot(df_hour.index, df_hour.values)
plt.title('Hour Sales', weight='bold', fontsize=20)
plt.grid(True)
plt.xticks(ticks=df_hour.index)
plt.ylabel('Sales (in million)')
plt.xlabel('Hour')
plt.show()
```

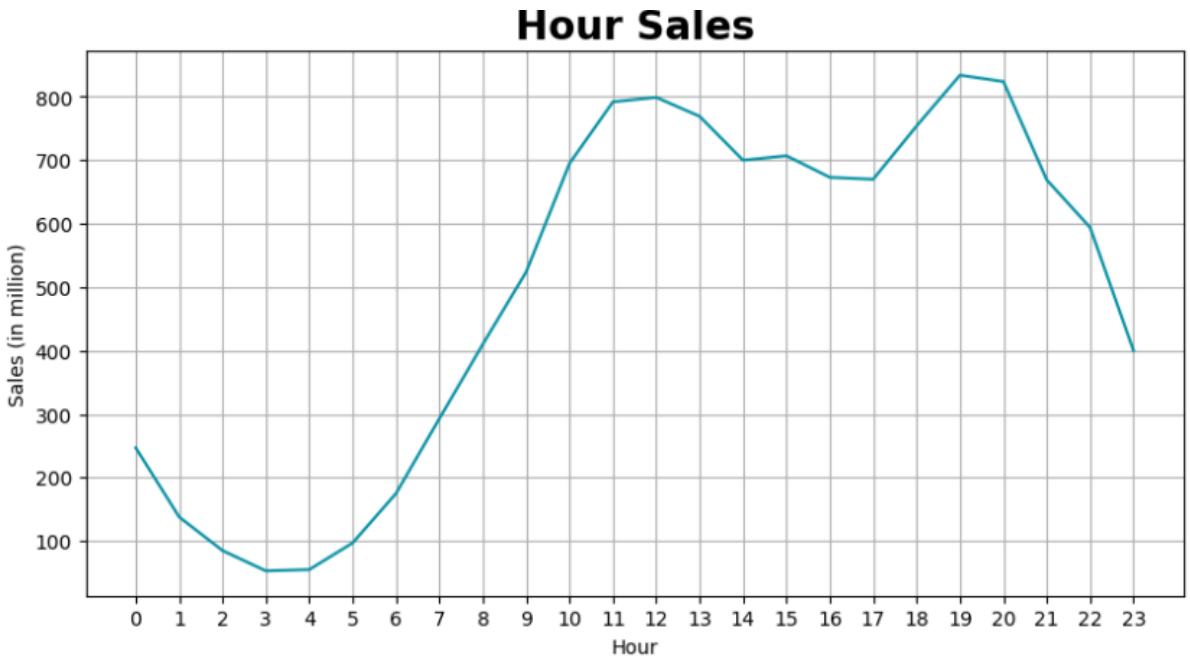
```
# Plot city-specific sales
df_city = df.groupby('Cities')['Sales'].sum()
plt.figure(figsize=(10, 5))
df_city.sort_values(ascending=False).plot(kind='bar', color=['#0892a5',
'#2e9b9b', '#50a290', '#6fa985', '#8dad7f', '#a9b17e', '#c4b383',
'#dbb68f'])
plt.title('Cities Sales', weight='bold', fontsize=20)
plt.ylabel('Sales (in million)')
plt.show()
```

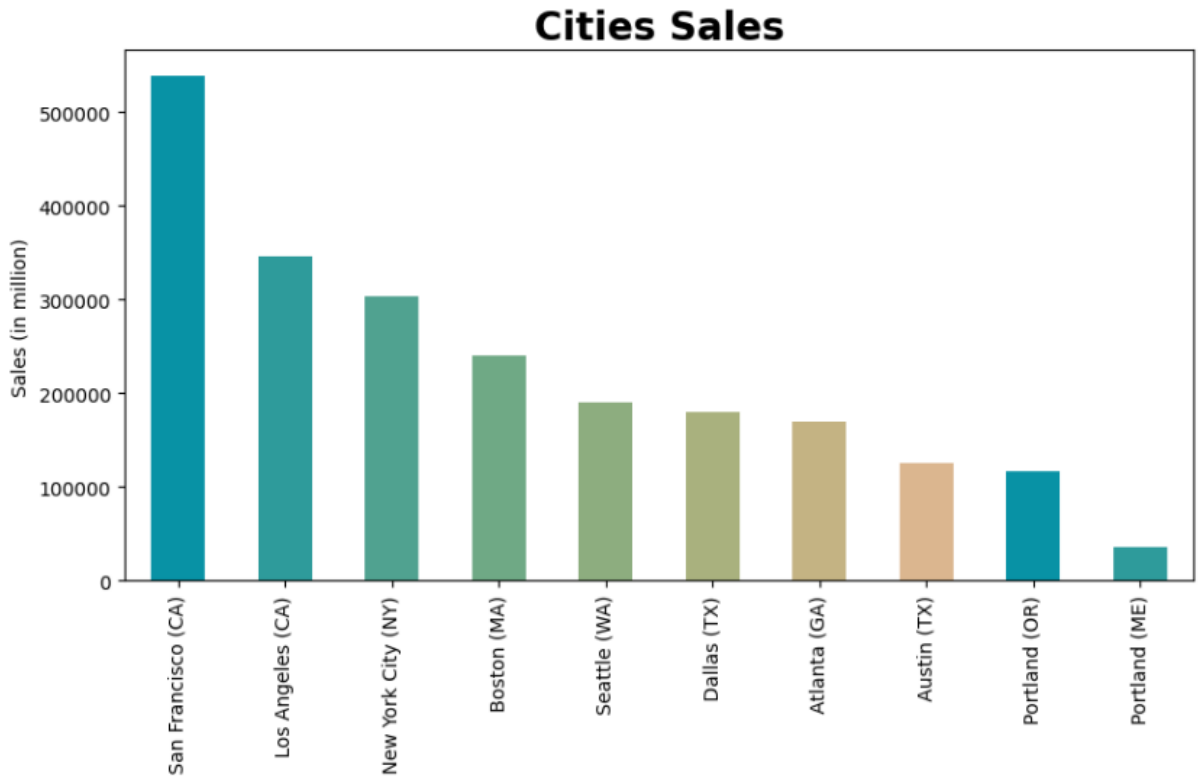
```
# Plot the most popular products by quantity ordered
df_product = df.groupby('Product')['Quantity Ordered'].sum()
df_product = df_product.sort_values(ascending=False)

plt.figure(figsize=(10, 6))
df_product.plot(kind='bar', color=['#0892a5', '#2e9b9b', '#50a290',
'#6fa985', '#8dad7f', '#a9b17e', '#c4b383', '#dbb68f'])
plt.title('Product Quantity Orders', weight='bold', fontsize=20)
plt.ylabel('Quantity (pcs)')
plt.show()
```


Output:







Model output:

Code:

```
def proba_prod(product, df):
    """
    Calculate the monthly and yearly probabilities for a given product.

    Arguments:
    product -- The name of the product to calculate probabilities for.
    df -- The DataFrame containing the data.

    Returns:
    A tuple (prob_month, prob_year):
    prob_month -- A numpy array of monthly probabilities.
    prob_year -- The yearly probability.
    """
    total_rows = df.shape[0]
    product_df = df[df['Product'] == product]
    product_rows = product_df.shape[0]
    prob_year = round(product_rows / total_rows * 100, 2)

    monthly_probabilities = []
    for month in range(1, 13):
        monthly_total = df[df['Month'] == month].shape[0]
        monthly_product = product_df[product_df['Month'] ==
month].shape[0]
        if monthly_total == 0:
            monthly_probabilities.append(0)
        else:
            monthly_probability = round((monthly_product /
monthly_total) * 100, 3)
            monthly_probabilities.append(monthly_probability)
    prob_month = np.array(monthly_probabilities)

    return prob_month, prob_year
# Define the list of products to analyze
products = [
    'USB-C Charging Cable', 'Lightning Charging Cable', 'Google Phone',
'iPhone',
    'Wired Headphones', 'Apple Airpods Headphones', 'Bose SoundSport
Headphones'
]
```



```
# Calculate and display probabilities for each product
for product in products:
    prob_month, prob_year = proba_prod(product, df)
    print(f'Yearly probability for {product}: {prob_year}%')
    print(f'Monthly probabilities for {product}: {prob_month}')
    print()
```

Output:

```
Yearly probability for USB-C Charging Cable: 11.24%
Monthly probabilities for USB-C Charging Cable: [ 0.    0.    0.    0.    0.    0.    0.   11.235 16.667  0.
  0.    0.   ]

Yearly probability for Lightning Charging Cable: 11.33%
Monthly probabilities for Lightning Charging Cable: [ 0.    0.    0.    0.    0.    0.    0.   11.327 16.667  0.
  0.    0.   ]

Yearly probability for Google Phone: 2.99%
Monthly probabilities for Google Phone: [0.    0.    0.    0.    0.    0.    0.    2.997 0.    0.    0.    0.   ]

Yearly probability for iPhone: 3.66%
Monthly probabilities for iPhone: [0.    0.    0.    0.    0.    0.    0.    3.667 0.    0.    0.    0.   ]

Yearly probability for Wired Headphones: 9.97%
Monthly probabilities for Wired Headphones: [0.    0.    0.    0.    0.    0.    0.    9.979 0.    0.    0.    0.   ]

Yearly probability for Apple Airpods Headphones: 8.36%
Monthly probabilities for Apple Airpods Headphones: [ 0.    0.    0.    0.    0.    0.    0.    8.355 16.667  0.
  0.    0.   ]

Yearly probability for Bose SoundSport Headphones: 7.28%
Monthly probabilities for Bose SoundSport Headphones: [0.    0.    0.    0.    0.    0.    0.    7.283 8.333 0.    0.    0.   ]
```

Manage relationship

Managing relationships is an essential aspect of any e-commerce sales data analysis project. The project involves various stakeholders, such as data scientists, business analysts, IT teams, and business executives, as well as third-party vendors and service providers. Here are some key aspects of managing relationships in this project:

Stakeholder Engagement and Communication:

- Maintain clear and open lines of communication with all stakeholders to ensure project goals are understood and aligned.
- Provide regular updates on project progress, milestones, and potential challenges.

Cross-Functional Collaboration:

- Foster collaboration between different teams, such as data science, IT, and business units, to leverage diverse expertise and perspectives.

- Encourage teamwork and shared ownership of project outcomes.

Expectation Management:

- Clearly define project scope, deliverables, and timelines to set realistic expectations for stakeholders.
- Manage expectations regarding the potential impact and limitations of the analysis and models.

Vendor and Service Provider Management:

- Establish strong relationships with cloud service providers, data vendors, and other third-party partners.
- Negotiate contracts and service level agreements (SLAs) that align with project needs and business objectives.

Feedback and Continuous Improvement:

- Gather feedback from stakeholders on project deliverables and processes to identify areas for improvement.
- Use feedback to make iterative adjustments to the project approach and deliver better results.

Data Governance and Compliance:

- Work closely with legal and compliance teams to ensure data privacy and security regulations are followed.
- Establish clear data governance policies to manage data access, usage, and storage.

Change Management:

- Prepare stakeholders for changes resulting from the project, such as new data-driven strategies or technology implementations.
- Provide training and support to help stakeholders adapt to changes effectively.

Performance Measurement and Reporting:

- Measure and report on key performance indicators (KPIs) to demonstrate the project's value and impact.
- Use performance data to justify ongoing investment and support for the project.

By effectively managing relationships with stakeholders, teams, and vendors, the project can achieve its objectives and deliver meaningful insights that drive business success. Strong relationships contribute to a collaborative and productive project environment, enabling efficient execution and optimal outcomes.

Project result:

```
Yearly probability for USB-C Charging Cable: 11.24%
Monthly probabilities for USB-C Charging Cable: [ 0.     0.     0.     0.     0.     0.     0.    11.235 16.667  0.
  0.     0.   ]

Yearly probability for Lightning Charging Cable: 11.33%
Monthly probabilities for Lightning Charging Cable: [ 0.     0.     0.     0.     0.     0.     0.    11.327 16.667  0.
  0.     0.   ]

Yearly probability for Google Phone: 2.99%
Monthly probabilities for Google Phone: [0.    0.    0.    0.    0.    0.    0.    2.997 0.    0.    0.    0.   ]

Yearly probability for iPhone: 3.66%
Monthly probabilities for iPhone: [0.    0.    0.    0.    0.    0.    0.    3.667 0.    0.    0.    0.   ]

Yearly probability for Wired Headphones: 9.97%
Monthly probabilities for Wired Headphones: [0.    0.    0.    0.    0.    0.    0.    9.979 0.    0.    0.    0.   ]

Yearly probability for Apple Airpods Headphones: 8.36%
Monthly probabilities for Apple Airpods Headphones: [ 0.     0.     0.     0.     0.     0.     0.     8.355 16.667  0.
  0.     0.   ]

Yearly probability for Bose SoundSport Headphones: 7.28%
Monthly probabilities for Bose SoundSport Headphones: [0.    0.    0.    0.    0.    0.    0.    7.283 8.333 0.    0.    0.   ]
```


CONCLUSION

In conclusion, the e-commerce sales data analysis project using cloud computing leverages advanced data processing and machine learning techniques to provide valuable insights and enable data-driven decision-making. The project involves analysis sales data, conducting exploratory data analysis(EDA), and using machine learning models for sales forecasting, product recommendations, and customer behavior prediction

FUTURE SCOPE

The scope of the project can be defined in terms of the extent of data analysis, potential use cases, and avenues for further exploration and application. The project cover the following areas:

Data Coverage: The project focuses on sales data from an e-commerce dataset for the year 2019. The analysis includes order data, such as order ID, quantity ordered, price each, order date, purchase address, product details, and more.

Time Frame: The analysis is limited to the year 2019, but the methods and techniques can be applied to data from other years for longitudinal analysis.

REFERENCES

1. Project Github link (https://github.com/Sathuu04/E-Commerce-Analysis), Sathya M, 2024
2. Project video recorded link (https://www.youtube.com/watch?v=o6lQaPQURhc), Sathya M, 2024
3. Project PPT & Report github link (https://github.com/Sathuu04/E-Commerce-Analysis/tree/main/Report%20and%20PPT), Sathya M, 2024
4. Project Dataset Github link (https://github.com/Sathuu04/E-Commerce-Analysis/tree/main/Dataset), Sathya M, 2024

GIT Hub Link of Project Code:

https://github.com/Sathuu04/E-Commerce-Analysis/tree/main/Code

THANK YOU

