# SALES FORECAST, PURCHASE FORECAST MARKETING FORCAST

**MINI PROJECT REPORT**

Submitted by

**FATHIMATHU ALEESHSHA**

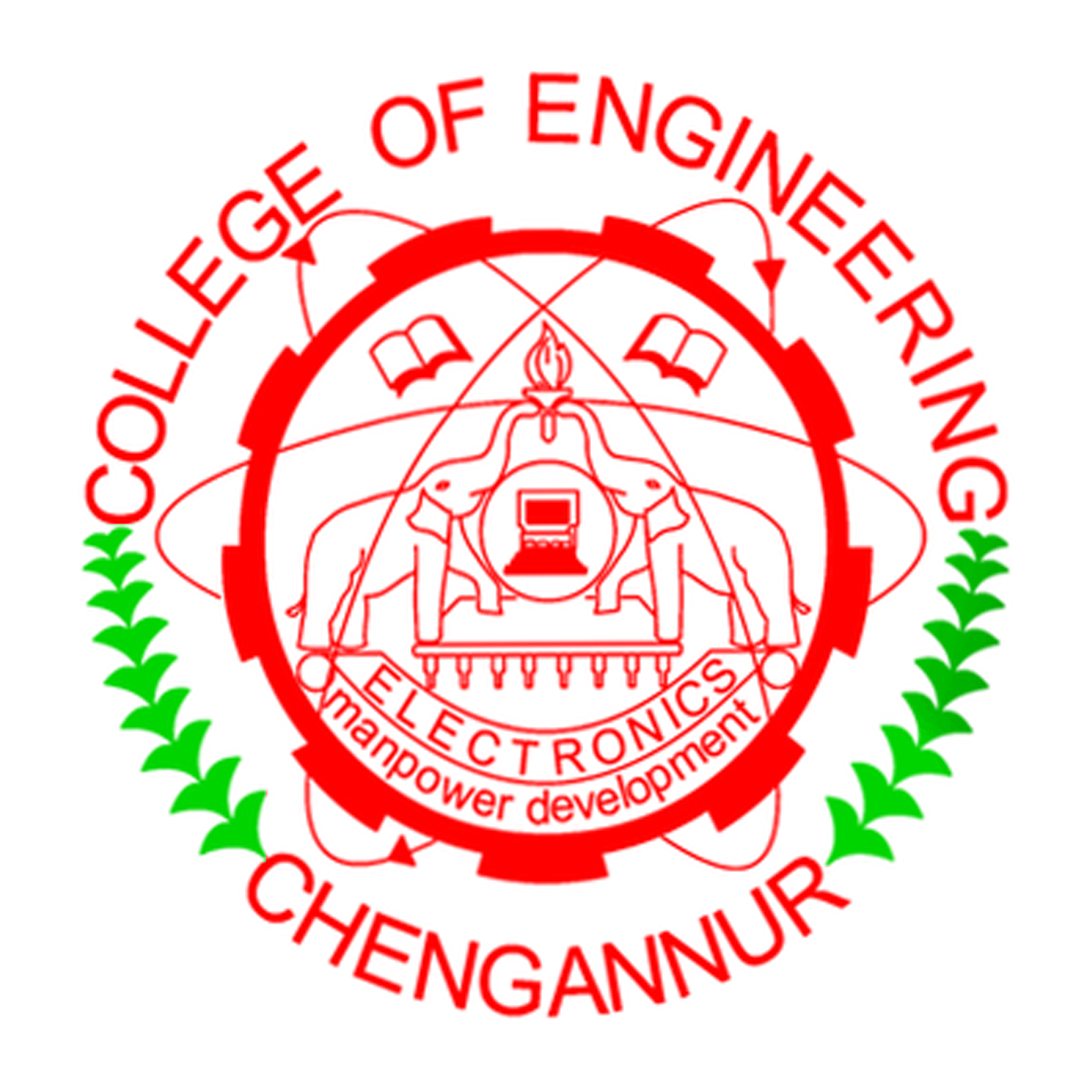
## CHN23MCA2032

to

### APJ Abdul Kalam Technological University

*in partial fulfillment of the requirements for the award of Degree in*

***Master of Computer Application***



## DEPARTMENT OF COMPUTER ENGINEERING COLLEGE OF ENGINEERING CHENGANNUR, ALAPPUZHA

**NOVEMBER 2024**

#### DEPARTMENT OF COMPUTER ENGINEERING COLLEGE OF ENGINEERING CHENGANNUR ALAPPUZHA

**CERTIFICATE**

*This is to certify that the project report titled* ***Sales , Purchase, Market forecasting*** *is a Bonafide record of the* ***CHN23MCA MINI PROJECT*** *presented by (CHN23MCA2032), Fourth Semester* ***FATHIMATHU ALEESHA*** *OF Master of Computer Application student, under my guidance and supervision. This mini project is submitted in partial fulfillment of the requirements for the award of the degree* ***Master of Computer Application*** *of* ***APJ Abdul Kalam Technological University****.*

#### Smt. Angel Thankam Thomas Ameena A

Project Guide Project Coordinator

Assistant Professor Assistant Professor

Dept. of Computer Engineering Dept. of Computer Engineering

**Gopakumar.G** Head of the Dept. Associate Professor

Dept. of Computer Engineering

# DECLARATION

I undersigned hereby declare that the project report “**Sales, Purchase, Market forecasting”-** , submitted for partial fulfillment of the requirements for the award of degree of Master of Computer Application of the APJ Abdul Kalam Technological University, Kerala is a Bonafide work done by us under the supervision of **Smt. Angel Thankam Thomas** ,Assistant Professor, Department of Computer Engineering as project Guide , **Ameena A** Project Coordinator and **Gopakumar G** Head of the department. This submission represents my ideas in my own words, and where ideas or words of others have been included, I have adequately and accurately cited and referenced the original sources. I also declare that I have adhered to the ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in our submission.

I understand that any violation of the above will be a cause for disciplinary action by the institute and/or the University and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been obtained. This report has not been previously formed the basis for the award of any degree, diploma, or similar title of any other University.

**Place:** Chengannur **FATHIMATHU AEESHA**

**Date :** 08/11/2024 **CHN23MCA2032**

#### 

**ACKNOWLEDGEMENT**

This work would not have been possible without the support of many people. First and foremost, I give thanks to Almighty God who gave me the inner strength, resources, and ability to complete my project successfully.

I would like to thank **Dr. Smitha Dharan**, The Principal College of Engineering Chengannur, for providing the best facilities and atmosphere for the project completion and presentation. Special thanks also go to HOD **Gopakumar G**, Associate Professor, Department of Computer

Engineering, for her exceptional support, guidance, and encouragement throughout the project. I would also like to thank our project coordinator, **Smt. Angel Thankam Thomas**, Assistant Professor, Department of Computer Engineering who also took on the role of my project guide, for their extended help and support during the project.

I would like to thank my dear friends and faculties for extending their cooperation and encouragement throughout the project work, without which I would never have completed the project this well. Thank you all for your love and for being very understanding.

**FATHIMATHU ALEESHA**

**CHN23MCA2032**

# ABSTRACT

Machine learning (ML) can greatly enhance the analysis of sales, purchases, and marketing by extracting meaningful patterns from data, predicting trends, and optimizing decision-making processes.

**Sales Analysis** to predict future sales based on historical data and external factors, in sales identify distinct customer segments based on purchasing behavior, demographics, and preferences.

**Purchases analysis** used for demand Forecasting Similar to sales forecasting, Machine learning models predict demand for inventory management to avoid overstocking or stockouts. Supplier Risk Analysis: Using Machine learning algorithms to evaluate supplier performance by analyzing data points related to delivery times, product quality, and reliability Analyzing purchasing patterns to identify cost-saving opportunities, such as bulk purchasing or alternative suppliers.

**Marketing Analysis Targeted Marketing Campaigns**: Using Machine Learning models to personalize marketing efforts by predicting which customers are most likely to respond to certain campaigns.

Machine learning can be used to analyze customer feedback, reviews, and social media posts to understand customer sentiment toward products or services. Machine leatning helps predict the long-term value of a customer using models that analyze purchasing history, frequency, and other key metrics.

**CONTENTS**

1. Declaration
2. Acknowledgement
3. Abstraction
4. Introduction
5. Problem Definition and motivations
6. Literation Review.
7. Data Flow diagram
8. Problem Statement
9. Hardware and Software Review
10. Source Code.
11. Screen Shot.
12. Test Case Result
13. Conclusion.
14. References.

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# INTRODUCTION

**Project Area**

The main project area to cover Analyzing Sales, Purchase and Marketing using machine leaning language.

# Objectives

* + 1. **Implementation of ML Technology:** Sales tracking by Machine language
    2. **Development of Advertiser Dashboard:**  Developed dash board to display Sales, Purchasing , marketing data.
    3. **Enhancement of Stakeholders Engagement:** To create immersive and interactive Machine learning experiences that enhance stake holders engagement and understanding of Sales, Purchase and Marketing products .
    4. **Disruption of Traditional Marketing Methods:** To disrupt traditional marketing methods by offering innovative solutions that transform sales, purchase and marketing practices.

**Problem Definition and Motivations**

**Existing System.**

Defining a problem in machine learning is one of the most crucial steps in developing a successful model. This step sets the scope, approach, and metrics for evaluating solutions. Here’s a breakdown of key aspects to consider in problem definition for machine learning.

The existing system for traditional sales that primarily relies on traditional methods.

The existing system uses traditional purchasing practices.

These static marketing techniques are limited in their ability to engage audiences and provide interactive experiences.

Define the data requirements for the model, including feature selection, data quality, and quantity.

Consider if the data is already available or if it needs to be collected or synthesized. Address data privacy and security concerns as necessary

**Limitations**

* + - Limited interactivity: Traditional sales offer limited opportunities for interaction with consumers, resulting in passive advertising experiences.
    - Limited stock due to traditional purchasing practices.
    - Ineffective measurement: Measuring the effectiveness of traditional marketing advertisements can be challenging, as it relies on indirect methods such as surveys and sales data, which may not provide accurate or real-time insights.

**LITERATURE REVIEW**

1. General Machine Learning

"Pattern Recognition and Machine Learning" by Christopher Bishop

* + A comprehensive introduction to probabilistic graphical models and foundational ML techniques. Recommended for readers who want a solid theoretical understanding.

"Machine Learning: A Probabilistic Perspective" by Kevin P. Murphy

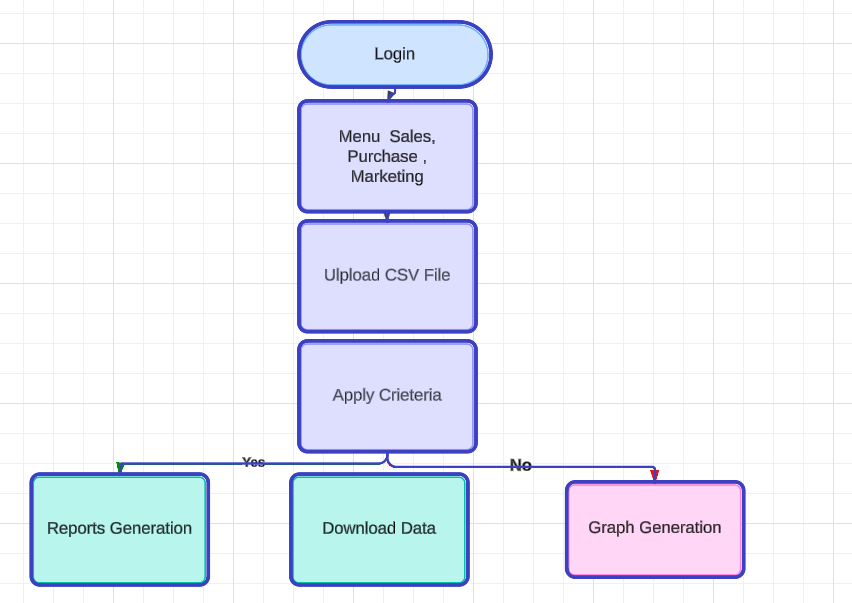
* + Covers ML methods through the lens of probability and statistics, with a strong emphasis on Bayesian approaches. It's more math-intensive but widely regarded as a core text.

2."Understanding Machine Learning: From Theory to Algorithms" by Shai Shalev-Shwartz and Shai Ben-David

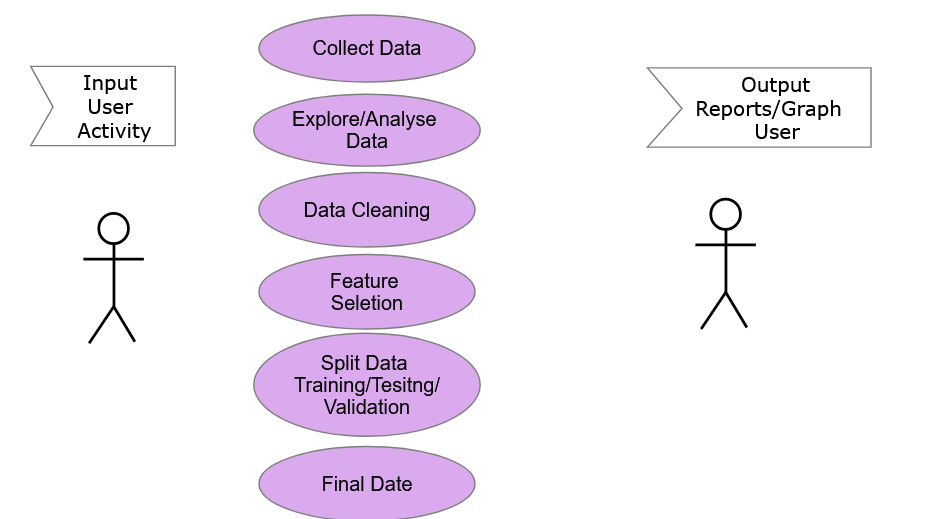
4."Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow" by Aurélien Géron

5. "Machine Learning Yearning" by Andrew Ng

**DATA FLOW DIAGRAM**



**User Case Diagram**



# Problem Statement

To design and implement a solution that maximizes management interactions.

# Proposed System

Aims to revolutionize traditional sales, purchase and marketing practices,. Key features of the proposed model include:

**Hardware Requirements**

* + - **Processor:** Octa-core (or higher) processor with support for ARM64 architecture.
    - **Memory (RAM):** At least 8 GB RAM or higher.
    - **Storage:** Minimum 200 GB internal storage.

**Software Requirements**

* **Pycham.**
* **Chrome Browser**
* **Python Library pandas,matplotlib,numphy,streamlit**

# Source Code

# Main.py

# import streamlit as st ### Page Setup admim\_page = st.Page( page="admin.py", title= "Admin Page", default=True, ) sales\_page = st.Page( page="exceldataset.py", title="Sales Page", ) purchase\_page = st.Page( page="purchase.py", title="Purchase Page", ) marketing\_page = st.Page( page="marketing.py", title="Marketing Page", ) about\_page = st.Page( page="about.py", title = "About Project", ) ##pg=st.navigation(pages=[admim\_page,sales\_page,purchase\_page,marketing\_page,about\_page]) ##pg.run() pg=st.navigation( { "info":[admim\_page], "Projects":[sales\_page,purchase\_page,marketing\_page,about\_page], } ) pg.run()

# sales.py

# import time import concurrent.futures import streamlit as st from streamlit.runtime.scriptrunner import add\_script\_run\_ctx import flet as ft import plotly.express as px import pandas as pd import os import warnings import sys warnings.filterwarnings('ignore') st.set\_page\_config(page\_title="Super Market Project ", page\_icon=":bar\_chart:",layout="wide") st.title(" :bar\_chart: Super Market Project by Fathimathu Aleesha") st.markdown('<style> div.block.container {padding-top:1rem;} </style>',unsafe\_allow\_html=True) fl = st.file\_uploader("file\_folder: Upload a File",type=(["csv","txt","xlsx","xls"])) if fl is not None: filename = fl.name st.write(filename) df = pd.read\_csv(filename,encoding="ANSI") else: os.chdir(r"C:\My Laptop\Munna Notes\MCA\Sem 3\Study Materials\Mini Project\SalesPurchaseMarketing\pythonProject1") df = pd.read\_csv("Superstore1.csv",encoding="ANSI") # "ISO-8859-1" col1, col2 = st.columns((2)) df["Order Date"] = pd.to\_datetime(df["Order Date"]) # Getting the MIn and Max Date startDate = pd.to\_datetime(df["Order Date"]).min() endDate = pd.to\_datetime(df["Order Date"]).max() with col1: date1 = pd.to\_datetime(st.date\_input("Start Date ",startDate)) with col2: date2 = pd.to\_datetime(st.date\_input("End Date",endDate)) df = df[(df["Order Date"] >= date1) & (df["Order Date"] <= date2)].copy() # Side bar filter st.sidebar.header("Choose your filter:") region= st.sidebar.multiselect("Pick Your Region", df["Region"].unique()) ## create for region if not region: df2=df.copy() else: df2=df[df["Region"].isin(region)] ## create for State state=st.sidebar.multiselect("Pick the State",df2["State"].unique()) if not state: df3=df2.copy() else: df3=df2[df2["State"].isin(state)] ## create for city city = st.sidebar.multiselect("Pick the City", df3["City"].unique()) ## Filter based on Region , state and city if not region and not state and not city: filtered\_df = df elif not state and not city: filtered\_df = df[df["Region"].isin(region)] elif not region and not city: filtered\_df = df[df["State"].isin(state)] elif state and city: filtered\_df = df3[df["State"].isin(state) & df3["City"].isin(city)] elif region and city: filtered\_df= df3[df3[df["Region"].isin(region) & df3["City"].isin(city)]] elif region and state: filtered\_df = df3[df["Region"].isin (region) & df3["State"].isin(state)] elif city: filtered\_df = df3[df3["City"].isin(city)] else: filtered\_df = df3[df3["Region"].isin(region) & df3["state"],isin(state) & df3["City"].isin(city)] # ok 08-10-2024 category\_df = filtered\_df.groupby (by=["Category"], as\_index = False )["Sales"].sum() with col1: st.subheader("Category Wise Sales") fig=px.bar(category\_df,x="Category",y="Sales", text= ['${:,.2f}'.format(x) for x in category\_df["Sales"]],template= "seaborn") st.plotly\_chart(fig,use\_container\_width=True,height =200) with col2: st.subheader("Category wise Sales") fig=px.pie(filtered\_df , values= "Sales" , names = "Region", hole =0.5) fig.update\_traces(text = filtered\_df ["Region"],textposition="outside") st.plotly\_chart(fig,use\_container\_width=True) # Download data Codet cl1,cl2 = st.columns(2) with cl1: with st.expander("Category\_View Data"): st.write(category\_df.style.background\_gradient(cmap="Blues")) csv = category\_df.to\_csv(index=False).encode('utf-8') st.download\_button("Download Data",data= csv, file\_name = "Category.csv",mime="text/csv", help="Click Here to Download the data as a CSV file") with cl2: with st.expander("Region\_View Data"): region = filtered\_df.groupby (by="Region",as\_index= False)["Sales"].sum() st.write(region.style.background\_gradient(cmap="Oranges")) csv = region.to\_csv(index=False).encode('utf-8') st.download\_button("Download Data", data=csv, file\_name="Region.csv", mime="text/csv", help="Click Here to Download the data as a CSV file") ### Time series analysis filtered\_df["month\_year"]=filtered\_df["Order Date"].dt.to\_period("M") st.subheader('Time Series Analysis') linechart = pd.DataFrame(filtered\_df.groupby(filtered\_df["month\_year"].dt.strftime("%Y : %b"))["Sales"].sum()).reset\_index() fig2=px.line(linechart, x="month\_year",y="Sales",labels= {"Sales": "Amount"},height=500,width =1000,template="gridon") st.plotly\_chart(fig2,use\_container\_width=True) ### Download time series data with st.expander("View Data of Time Series"): st.write(linechart.T.style.background\_gradient(cmap="Blues")) csv=linechart.to\_csv(index=False).encode("utf=8") st.download\_button('Download Data',data=csv,file\_name="TimeSeries.csv") #### Create a tree based on Region , Category,sub-cateogory st.subheader("Hierarchical view of Sales using tree map") fig3 = px.treemap(filtered\_df,path =["Region","Category","Sub-Category"],values= "Sales", hover\_data=["Sales"],color= "Sub-Category") fig3.update\_layout(width = 800, height = 650) st.plotly\_chart(fig3,use\_container\_width= True) ## Segment wise Sales chart1, chart2 = st.columns((2)) with chart1: st.subheader("Segment WIse Sales") fig = px.pie(filtered\_df, values="Sales", names="Segment", template="plotly\_dark") fig.update\_traces(text=filtered\_df["Segment"],textposition="inside") st.plotly\_chart(fig,use\_container\_width=True) ## Category wise Sales with chart2: st.subheader("Category Wise Sales") fig = px.pie(filtered\_df, values="Sales", names="Category", template="gridon") fig.update\_traces(text=filtered\_df["Category"], textposition="inside") st.plotly\_chart(fig, use\_container\_width=True) ### Show table data in month wise Sales Sammary import plotly.figure\_factory as ff st.subheader(":point\_right: Month wise Sub-Category Sales Summary") with st.expander("Summary Table"): df\_sample = df[0:5] [["Region","State","City","Category","Sales","Profit","Quantity"]] fig = ff.create\_table(df\_sample,colorscale="Cividis") st.plotly\_chart(fig, use\_container\_width=True) ### SHow hcategory wise sales st.markdown("Month wise sub-Category Table") filtered\_df["month"]=filtered\_df["Order Date"].dt.month\_name() sub\_category\_Year = pd.pivot\_table(data=filtered\_df,values="Sales",index=["Sub-Category"],columns="month") st.write(sub\_category\_Year.style.background\_gradient(cmap="Blues")) ## Scatter polotter SHow the relatioship between Sales and Profit data1 = px.scatter(filtered\_df,x="Sales",y="Profit",size="Quantity") data1['layout'].update(title="Relationship between Sales and Profit using Scatter Plot.", titlefont= dict(size=20),xaxis = dict(title="Sales",titlefont=dict(size=19)), yaxis=dict(title= "Profit",titlefont= dict(size=19))) st.plotly\_chart(data1,use\_container\_width=True) ## View Sacattered Plet Data in Table with st.expander("View Data"): st.write(filtered\_df.iloc[:500,1:20:2].style.background\_gradient(cmap="Oranges")) ## Download the data set csv= df.to\_csv(index=False).encode('utf-8') st.download\_button('Download Data',data = csv,file\_name="Data.csv")

# Screen shorts

# 

# 

# 

# 

# A screenshot of a computer screen Description automatically generated

# A pie chart with text on it Description automatically generated

# A screen shot of a graph Description automatically generated

# A screenshot of a computer Description automatically generated

# Testing

# Unit Testing

# Unit testing includes collecting data, explore and analyzing data, data cleaning.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| **Test Case  ID** | **Test Case** | **Test Case Name** | **Steps** | **Pass/Fail** |
| **1** | **Collect Data** | **Pre-processing** | **No Problem in Data** | **Pass** |
| **2** | **Explore and Analyze Data** | **Pre-Processing** | **No Duplication Data** | **Pass** |
| **3** | **Data Cleaning** | **Pre-processing** | **Remove all glitches in Data** | **Pass** |
| **4** | **Data Transformation** | **Data transform into CSV file** | **Converted from XLS to CSV** | **Pass** |
| **4** | **Feature Selection** | **Filter/Wrapper/Embedded** | **Filter Completed** | **Pass** |
| **5** | **Split Data into Training Set** | **60-70% of the data for training** | **Split the data for Model** | **Pass** |
| **6** | **Split data Validation Set** | **15-20% of the data for Validation** | **Split the data for Validation** | **Pass** |
| **7** | **Split Data Test Sets** | **15-20% of thefor Test Data Set** | **Split the data for Testing** | **Pass** |

## 

## System Testing

System testing includes uploading data, view data, download, reports, graph etc. System testing helps identify any inconsistencies or deviations from expected behavior and ensures that the platform functions reliably across different types of dates

# System Test Case Result

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Case  ID** | **Test Case Name** | **Description** | **Expected Result** | **Pass/Fail** |
| **1** | **Uploading Data Set** | **Upload CSV file** | **Visualize the data in Browser** | **Pass** |
| **2** | **View Data of Sales** | **Upload CSV file** | **Data Viewed in Browser** | **Pass** |
| **3** | **Download Data** | **Download Data** | **Download Data to Local folder** | **Pass** |
| **4** | **Reports Generation** | **Report Generation on Crieteria** | **Report Viewed in Broser** | **Pass** |
| **4** | **Graph generated** | **Scatter /Heirachichial View graph** | **Graph Viwed in Browser** | **Pass** |
| **5** | **Time Series Analysis** | **Time Sereis Graph** | **Graph Viwed in Browser** | **Pass** |
| **6** | **Purchase data** | **Viwed the the data** | **Data and Graph viewed in Browser** | **Pass** |
| **7** | **Marketing Data** | **Viwed the the data** | **Data and Grpah viewed in Browser** | **Pass** |

**CONCLUSION AND FUTURE SCOPE**

The development of the Sales, Purchasing and marketing in machine learning platform represents a significant advancement in SALES. By seamlessly integrating digital content with enhancing the sales there by increase the market share of products. The platform enhances user engagement and provides a more immersive experience for both customers and suppliers.

The implementation of Reports and Graph generation and features enhance sakes and costing materials in real-time.

Moving forward, this reporting system platform has the potential to revolutionize traditional marketing methods by offering innovative solutions for sales and purchase.

This Machine learning platform represents a significant step forward in the convergence of digital and physical for supermarkets realms, providing a glimpse into the future of in the digital age.

**REFERENCES**

1. **"Machine Learning: A Probabilistic Perspective" by Kevin P. Murphy**  
   This book gives a comprehensive view of machine learning, emphasizing a probabilistic approach.
2. **"Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow" by Aurélien Géron**  
   A practical, beginner-friendly guide with a focus on Python and popular libraries for hands-on implementation.
3. **"A Few Useful Things to Know About Machine Learning" by Pedro Domingos**  
   An excellent overview of core machine learning concepts and practical insights.
4. **Scikit-Learn**: Scikit-Learn documentation is a go-to resource for understanding machine learning algorithms in Python.