**Product Scale Analysis**

**Phase 3**

Introduction:

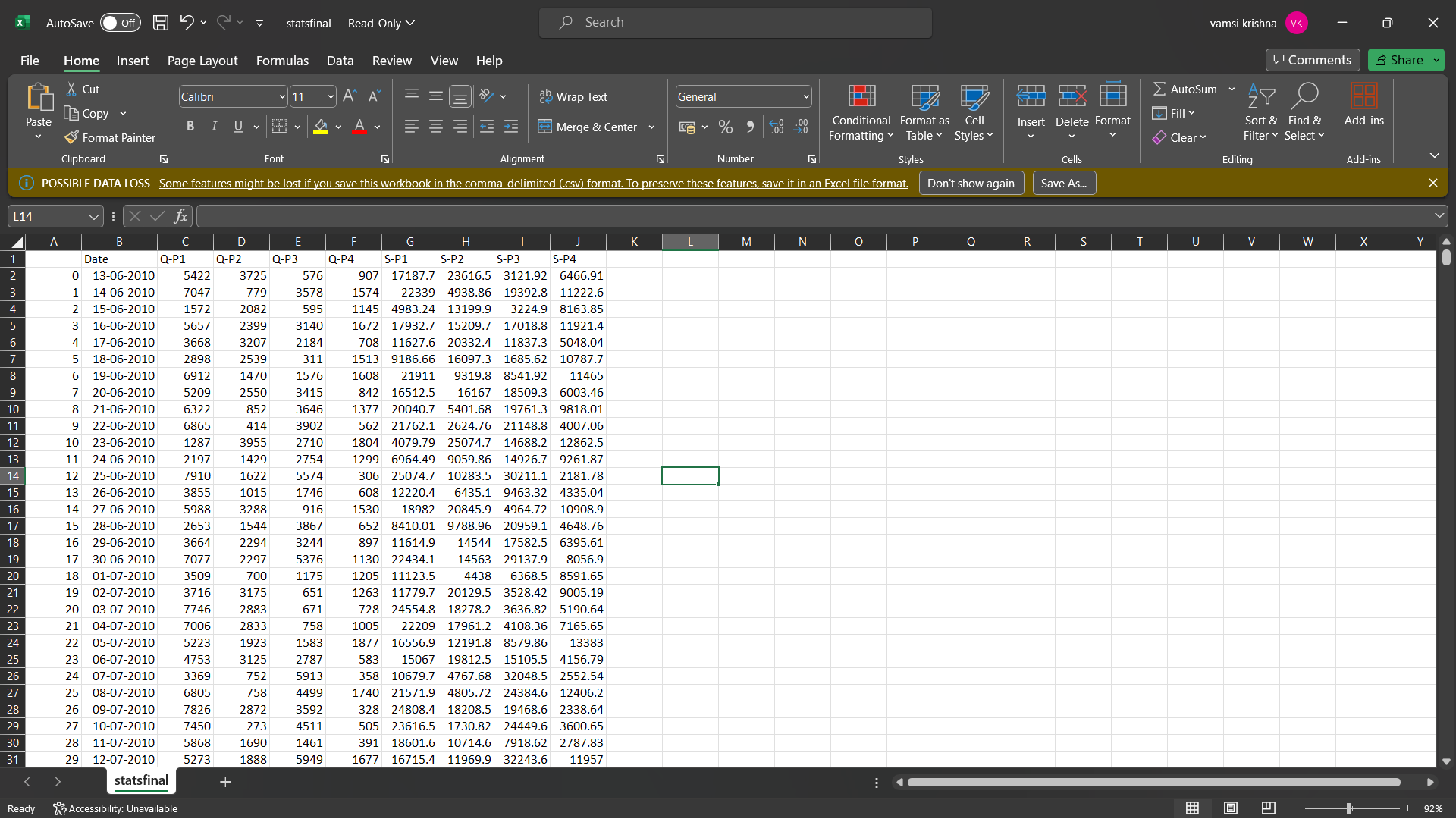
* Product scale analysis is a strategic evaluation process employed by businesses to assess the performance and potential of their products. It holds significant importance in guiding decision-making, resource allocation, and long-term planning.
* To effectively manage their product portfolio, companies need to understand the individual products within it. Product scale analysis helps in gaining insights into each product's strengths and weaknesses.
* Financial metrics are a central part of the analysis. This includes assessing revenue, profit margins, and cost structures associated with the product.
* The product's role in the overall business strategy is crucial. It must fit cohesively with the company's goals and objectives. Product scale analysis informs decisions about where to allocate resources. This includes considerations like marketing budgets, R&D investments, and personnel allocation.

1) Analysis Objectives:

* **Identify the Purpose**:
* Clearly articulate why you are conducting the product scale analysis. Common objectives may include assessing profitability, market share, customer satisfaction, or identifying growth opportunities.
* **Set Specific Goals:**
* Define specific, measurable goals that align with the analysis purpose. For instance, if the goal is to assess profitability, specify the profit margin target you want to achieve.
* **Identify Key Metrics:**
* Determine the key performance indicators (KPIs) that will help you measure progress toward your objectives. This could include metrics like revenue, profit margin, market share, customer acquisition cost, or customer retention rate.

**DATASET LINK:**

[**https://www.kaggle.com/datasets/ksabishek/product-sales-data**](https://www.kaggle.com/datasets/ksabishek/product-sales-data)



**2)** **Collect Sales Data:**

* **Identify Data Sources:**
* Identify the sources of sales data you'll need. This might include internal sources such as sales reports, accounting records, and CRM (Customer Relationship Management) software.
* External sources like market research reports, industry benchmarks, or third-party sales data providers can be valuable as well.
* **Data Collection Tools:**
* If the data is available in digital format, you can use spreadsheet software like Microsoft Excel or data analytics tools like Tableau or Power BI to collect and organize
* For data from physical sources, consider digitizing it through scanning or manual entry.
* **Data Collection Schedule:**
* Determine how often you'll collect sales data. Daily, weekly, monthly, or quarterly data collection may be necessary, depending on the objectives and the nature of the product**.**
* **Data Cleaning and Validation:**
* Ensure that the collected data is accurate and free of errors. Validate the data by cross-referencing it with multiple sources if possible.
* Clean the data by removing duplicates, correcting errors, and addressing any missing or incomplete information.

PROGRAM

* **DATA COLLECTION:**

data = pd.read\_csv(‘[**https://www.kaggle.com/datasets/ksabishek/product-sales-data**](https://www.kaggle.com/datasets/ksabishek/product-sales-data)**’)**

data.head()

import numpy as np

import pandas as pd

import os

for dirname, \_, filenames **in** os.walk('/kaggle/input'):

for filename **in** filenames

* **Product sales:**

import pandas as pd

import numpy as np

import seaborn as sns

import matplotlib.pyplot as plt

pd.options.display.max\_columns=50

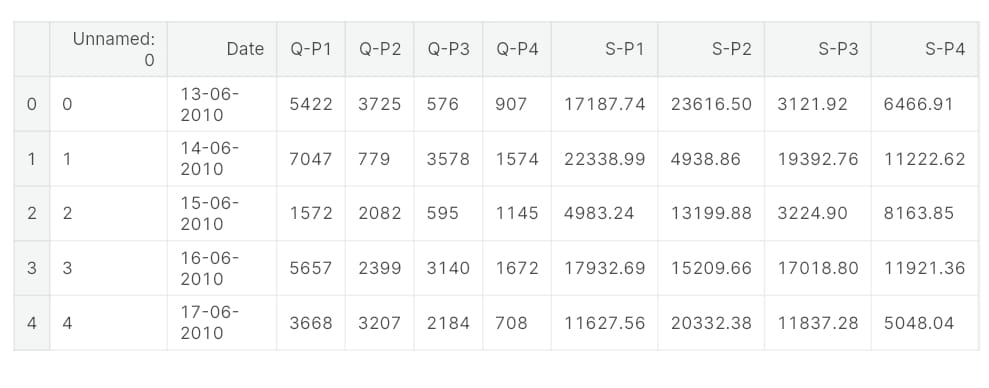
sns.set(style="darkgrid")

#### Importing data

linkcode

df=pd.read\_csv("/kaggle/input/product-sales-data/statsfinal.csv")

df.head(5)



* **UNDERSTANDING DATA:**

df.shape

(4600, 10)

df.columns

Index(['Unnamed: 0', 'Date', 'Q-P1', 'Q-P2', 'Q-P3', 'Q-P4', 'SPP1', 'S-P2',

'S-P3', 'S-P4'],

dtype='object')

df.info()

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 4600 entries, 0 to 4599

Data columns (total 10 columns):

# Column Non-Null Count Dtype

--- ------ -------------- -----

0 Unnamed: 0 4600 non-null int64

1 Date 4600 non-null object

2 Q-P1 4600 non-null int64

3 Q-P2 4600 non-null int64

4 Q-P3 4600 non-null int64

5 Q-P4 4600 non-null int64

6 S-P1 4600 non-null float64

7 S-P2 4600 non-null float64

8 S-P3 4600 non-null float64

9 S-P4 4600 non-null float64

dtypes: float64(4), int64(5), object(1)

memory usage: 359.5+ KB

df.isnull().sum()

Unnamed: 0 0

Date 0

Q-P1 0

Q-P2 0

Q-P3 0

Q-P4 0

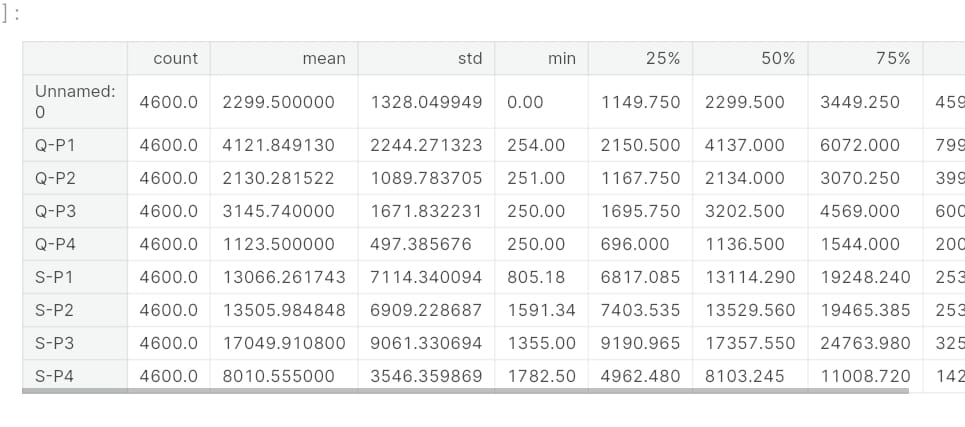
S-P1 0

S-P2 0

S-P3 0

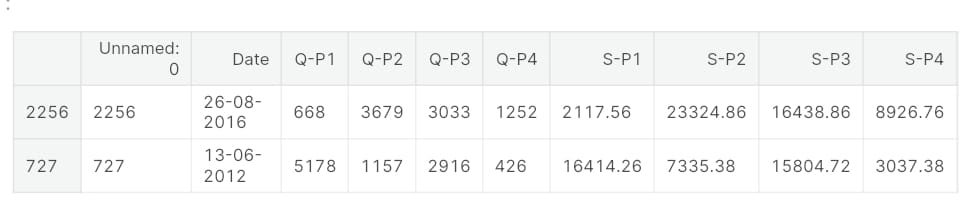
S-P4 0

dtype: int64



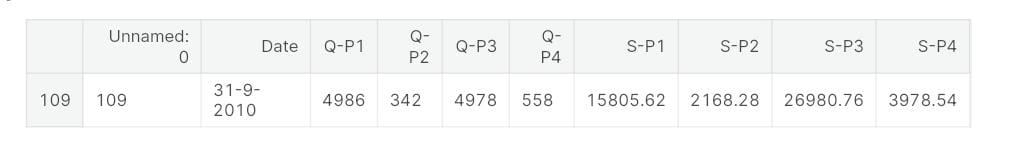
* **DATA CLEANING:**

df.sample(2)

****

from datetime import datetime as dt

df[df["Date"]=="31-9-2010"]

****

df["Date"].fillna(df["Date"].mean(),inplace=True)

df['Date'].isnull().sum()

df.dtypes

Unnamed: 0 int64

Date datetime64[ns]

Q-P1 int64

Q-P2 int64

Q-P3 int64

Q-P4 int64

S-P1 float64

S-P2 float64

S-P3 float64

S-P4 float64

dtype: object

df["month"]=df["Date"].dt.month\_name()

df["day"]=df["Date"].dt.day\_name()

df["dayoftheweek"]=df["Date"].dt.weekday

df["year"]=df["Date"].dt.year

df.sample()



df.drop(columns=["Unnamed: 0"],inplace=True)

df.sample();****

df.corr().T



**3)** **Data Analysis:**

* Once you've collected the sales data, you can start analyzing it to assess the product's performance in line with your defined objectives.
* Use appropriate data analysis tools and techniques to derive insights. This may involve creating charts, graphs, and reports to visualize and interpret the data effectively.
* Monitor the data over time to identify trends and patterns. Be prepared to adjust your analysis and objectives as the data reveals new information.
* Present your findings in a clear and understandable manner, often through reports or presentations, to facilitate decision-making within the organization.
* **EXPLORATORY DATA ANALYSIS:**

df.sample()

****

q = df[["Q-P1","Q-P2","Q-P3","Q-P4"]].sum()

print(q)

plt.figure(figsize=(8,8))

plt.pie(q,labels=df[["Q-P1","Q-P2","Q-P3","Q-P4"]].sum().index,shadow=True,autopct="**%0.01f%%**",textprops={"fontsize":20},wedgeprops={'width': 0.8},explode=[0,0,0,0.3])

plt.legend(loc='center right', bbox\_to\_anchor=(1.2, 0.8));

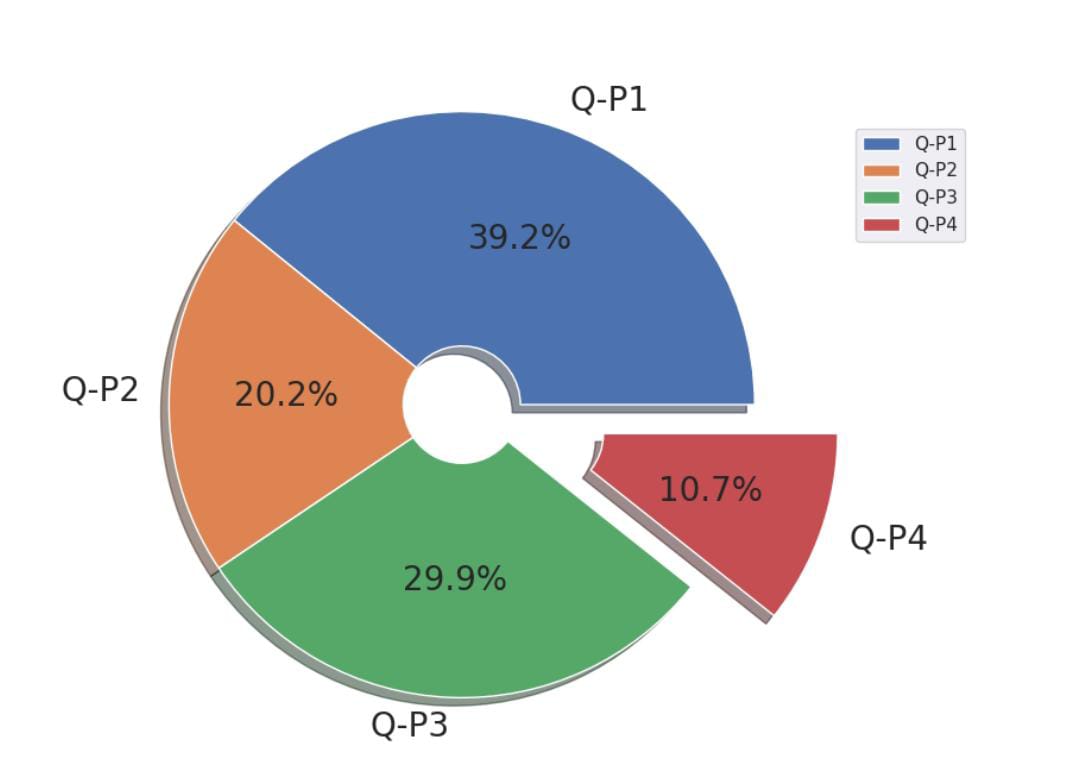
Q-P1 18960506

Q-P2 9799295

Q-P3 14470404

Q-P4 5168100

dtype: int64

****

print(df["month"].value\_counts())

plt.figure(figsize=(10,10))

sns.countplot(x="month",data=df,edgecolor="black")

plt.xticks(rotation=90);

October 411

January 399

July 398

June 385

August 385

September 385

November 385

December 385

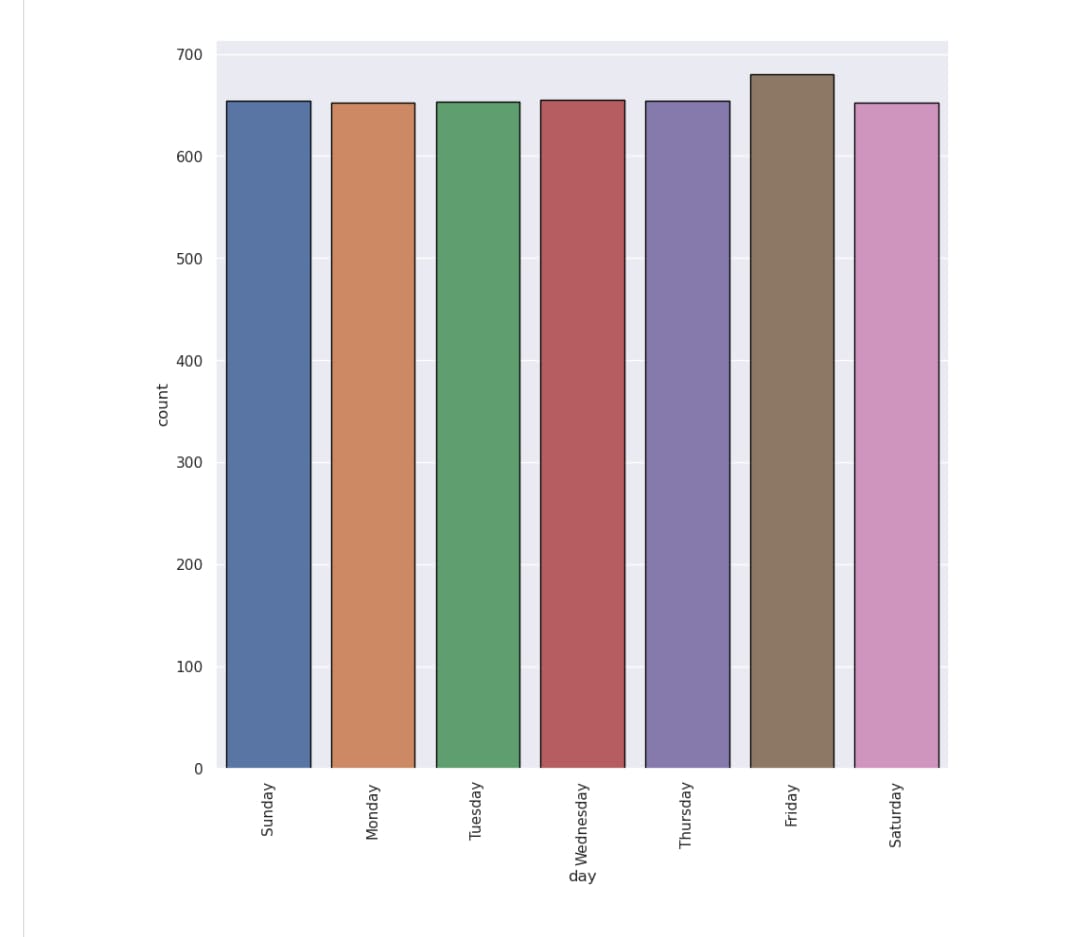
March 380

May 379

April 367

February 341

Name: month, dtype: int64

****

sns.relplot(x="month",y="S-P1",data=df,kind="line",height=10,color="red")

plt.xticks(rotation=90);

sns.relplot(x="month",y="S-P2",data=df,kind="line",height=10,color="blue")

plt.xticks(rotation=90);

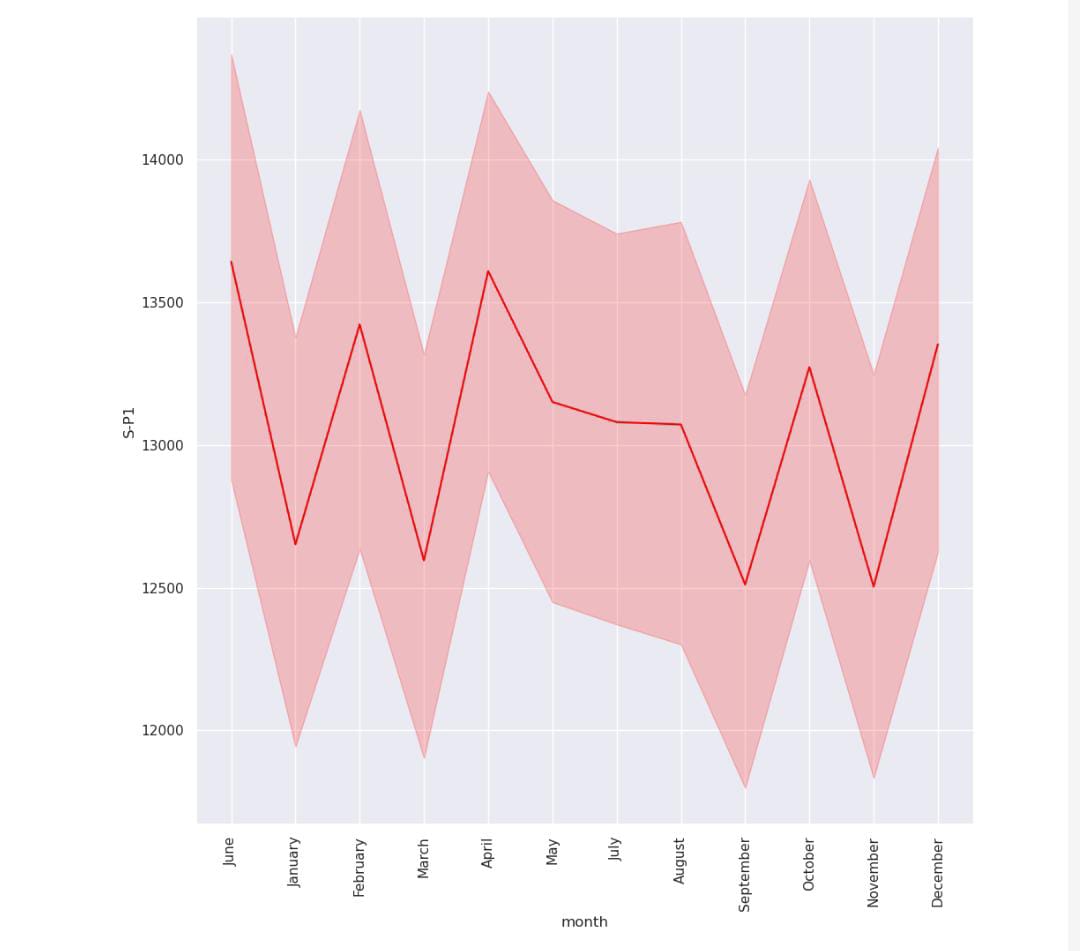
sns.relplot(x="month",y="S-P3",data=df,kind="line",height=10,color="green")

plt.

xticks(rotation=90);

sns.relplot(x="month",y="S-P4",data=df,kind="line",height=10,color="purple")

plt.xticks(rotation=90);



week\_t=df[df["dayoftheweek"]<5]

weekend\_t=df[df["dayoftheweek"]>=5]

print(week\_t.groupby("day")[["S-P1","S-P2","S-P3","S-P4"]].sum())

S-P1 S-P2 S-P3 S-P4

day

Friday 8913637.41 9267831.02 11428877.58 5463169.99

Monday 8636791.80 8864347.08 11064892.06 5292577.61

Thursday 8577981.96 8909481.54 10951554.44 5043013.35

Tuesday 8433525.06 8738326.90 11156338.30 5384854.07

Wednesday 8693537.97 8908067.72 11017830.20 5086827.20

In [36]:

linkcode

plt.figure(figsize=(10,10),dpi=100)

plt.subplot(2,2,1)

sns.barplot(x="day",y="S-P1",data=week\_t,edgecolor="black",estimator=sum)

plt.xticks(rotation=45);

plt.subplot(2,2,2)

sns.barplot(x="day",y="S-P2",data=week\_t,edgecolor="black",estimator=sum)

plt.xticks(rotation=45);

plt.subplot(2,2,3)

sns.barplot(x="day",y="S-P3",data=week\_t,edgecolor="black",estimator=sum)

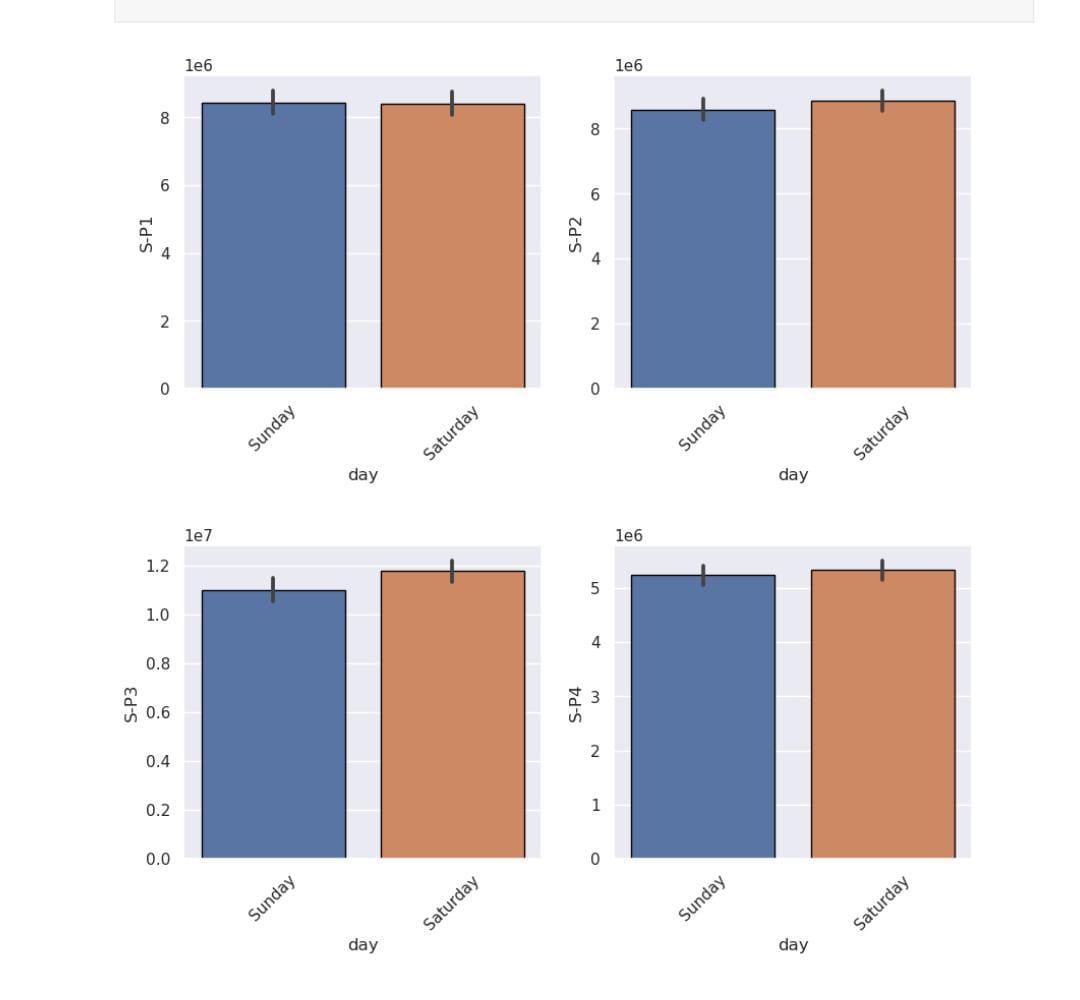
plt.xticks(rotation=45);

plt.subplot(2,2,4)

sns.barplot(x="day",y="S-P4",data=week\_t,edgecolor="black",estimator=sum)

plt.xticks(rotation=45)

plt.subplots\_adjust(hspace=0.5);



* **CONCLUSION:**
* Data-Driven Decision-Making: Product analytics enables data-driven decision-making by providing actionable insights based on real-world data. This approach allows companies to move away from guesswork and make informed choices.
* Customer-Centric Approach: Understanding customer behavior and preferences is crucial. By analyzing product data, businesses can segment their customers, tailor marketing efforts, and create personalized experiences that drive customer satisfaction and loyalty.
* Product Performance Optimization: Businesses can identify top-performing products, optimize pricing strategies, and manage inventory effectively. This leads to increased sales, improved profitability, and a stronger competitive edge.
* Competitive Advantage: By keeping an eye on competitors and market trends, businesses can identify opportunities and gaps in the market. This information allows for strategic product development and differentiation.
* Customer Satisfaction and Retention: Analyzing customer feedback and churn rates helps improve products and services, leading to higher customer satisfaction and increased customer retention.