**Data Analyst Task 2**

import pandas as pd

import matplotlib.pyplot as plt

# STEP 1: Load the QVI\_data.csv file

df = pd.read\_csv('QVI\_data.csv')

# Convert DATE to datetime and create MONTH column

df['DATE'] = pd.to\_datetime(df['DATE'])

df['MONTH'] = df['DATE'].dt.to\_period('M')

# STEP 2: Aggregate monthly metrics by store

monthly = df.groupby(['STORE\_NBR', 'MONTH']).agg(

TOT\_SALES=('TOT\_SALES', 'sum'),

NUM\_CUSTOMERS=('LYLTY\_CARD\_NBR', 'nunique'),

NUM\_TXNS=('TXN\_ID', 'nunique')

).reset\_index()

# Calculate average transactions per customer

monthly['AVG\_TXN\_PER\_CUST'] = monthly['NUM\_TXNS'] / monthly['NUM\_CUSTOMERS']

# STEP 3: Function to find the best control store for a trial store

def find\_best\_control(trial\_store, metric='TOT\_SALES', trial\_start='2019-02'):

trial\_data = monthly[(monthly['STORE\_NBR'] == trial\_store) & (monthly['MONTH'] < trial\_start)]

scores = {}

for store in monthly['STORE\_NBR'].unique():

if store == trial\_store:

continue

control\_data = monthly[(monthly['STORE\_NBR'] == store) & (monthly['MONTH'] < trial\_start)]

merged = pd.merge(trial\_data, control\_data, on='MONTH', suffixes=('\_trial', '\_control'))

if not merged.empty:

correlation = merged[f'{metric}\_trial'].corr(merged[f'{metric}\_control'])

scores[store] = correlation

if scores:

best\_match = max(scores, key=scores.get)

return best\_match, scores[best\_match]

else:

return None, None

# STEP 4: Find control stores for each trial store

trial\_stores = [77, 86, 88]

for trial in trial\_stores:

match, score = find\_best\_control(trial)

print(f"Best control for Store {trial}: Store {match} (correlation = {score:.2f})")

# STEP 5: OPTIONAL - Plotting example (for Store 77 and its control)

# Replace 'x' with your matched control store number for Store 77

trial\_store = 77

control\_store = 233 # Replace this with your result from above

comparison = monthly[(monthly['STORE\_NBR'].isin([trial\_store, control\_store]))]

comparison = comparison.copy()

comparison['MONTH'] = comparison['MONTH'].astype(str)

plt.figure(figsize=(12, 6))

for store in [trial\_store, control\_store]:

store\_data = comparison[comparison['STORE\_NBR'] == store]

plt.plot(store\_data['MONTH'], store\_data['TOT\_SALES'], label=f"Store {store}")

plt.xticks(rotation=45)

plt.title(f"Total Sales: Store {trial\_store} vs Control Store {control\_store}")

plt.xlabel("Month")

plt.ylabel("Total Sales")

plt.legend()

plt.tight\_layout()

plt.show()

A graph of sales

AI-generated content may be incorrect.

Store 77 was compared to Store 233, with a strong correlation (0.93) in pre-trial sales trends. During the trial period (Feb–Aug 2019), Store 77 showed a noticeable increase in total sales, while Store 233 remained flat. The increase was mainly driven by a rise in the number of customers, not just repeat transactions. This suggests that the new layout attracted more shoppers, and we recommend rolling it out to similar stores.