import pandas as pd

from sklearn.linear\_model import LinearRegression

from sklearn.model\_selection import train\_test\_split

from sklearn.metrics import mean\_squared\_error, r2\_score

from sklearn.preprocessing import PolynomialFeatures

import numpy as np

# Filter data for the South Central region and the specific product

df\_product = df\_23[(df\_23['Geography'] == 'Southeast - Standard - Multi Outlet + Conv') &

(df\_23['Product'] == 'GARDEIN FROZEN REGULAR MEAT SUBSTITUTE GROUND 13.7 OZ - 0842234000981')]

# Select relevant columns

X = df\_product[['Price per Unit', 'Price per Unit Feature Only', 'Price per Unit Display Only']]

y = df\_product['Unit Sales']

# Replace NaN values with 0

X.fillna(0, inplace=True)

# Adding interaction terms manually

interaction\_terms = []

for i in range(X.shape[1]):

for j in range(i + 1, X.shape[1]):

interaction\_terms.append(X.columns[i] + ' \* ' + X.columns[j])

X\_inter = np.hstack((X.values, (X.values[:, 0] \* X.values[:, 1]).reshape(-1, 1),

(X.values[:, 0] \* X.values[:, 2]).reshape(-1, 1),

(X.values[:, 1] \* X.values[:, 2]).reshape(-1, 1)))

# Split the data into training and test sets

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X\_inter, y, test\_size=0.2, random\_state=42)

# Fit the regression model

regression\_model = LinearRegression()

regression\_model.fit(X\_train, y\_train)

# Make predictions

y\_pred = regression\_model.predict(X\_test)

# Evaluate the model

mse = mean\_squared\_error(y\_test, y\_pred)

r2 = r2\_score(y\_test, y\_pred)

print(f'Mean Squared Error: {mse}')

print(f'R-squared: {r2}')

# Get the coefficients

coefficients = pd.DataFrame(regression\_model.coef\_, columns=['Coefficient'])

coefficients.insert(0, 'Feature', list(X.columns) + interaction\_terms)

print(coefficients)