Multiple Linear Regression

Code.py

```python
import numpy as np

import matplotlib.pyplot as plt

import pandas as pd

dataset = pd.read_csv(r"C:\Users\omkar joshi\DATA SCIENCE\Databases\Investment.csv")

X = dataset.iloc[:, :-1]

y = dataset.iloc[:, 4]

X = pd.get_dummies(X,dtype=int)

from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2, random_state = 0)

from sklearn.linear_model import LinearRegression

regressor = LinearRegression()

regressor.fit(X_train,y_train)

y_pred = regressor.predict(X_test)

#== we build mlr model

m = regressor.coef_

print(m)

c = regressor.intercept_

print(c)

X = np.append(arr = np.full((50, 1), 42467).astype(int), values = X, axis = 1)

import statsmodels.api as sm

X_opt = X[:,[0,1,2,3,4,5]]

#OrdinaryLeastSquares

regressor_OLS = sm.OLS(endog=y, exog=X_opt).fit()

regressor_OLS.summary()

import statsmodels.api as sm

X_opt = X[:,[0,1,2,3,5]]
```

```python
#OrdinaryLeastSquares

regressor_OLS = sm.OLS(endog=y, exog=X_opt).fit()

regressor_OLS.summary()

import statsmodels.api as sm

X_opt = X[:,[0,1,2,3]]

#OrdinaryLeastSquares

regressor_OLS = sm.OLS(endog=y, exog=X_opt).fit()

regressor_OLS.summary()

import statsmodels.api as sm

X_opt = X[:,[0,1,3]]

#OrdinaryLeastSquares

regressor_OLS = sm.OLS(endog=y, exog=X_opt).fit()

regressor_OLS.summary()

import statsmodels.api as sm

X_opt = X[:,[0,1]]

#OrdinaryLeastSquares

regressor_OLS = sm.OLS(endog=y, exog=X_opt).fit()

regressor_OLS.summary()

bias = regressor.score(X_train, y_train)

bias

variance = regressor.score(X_test, y_test)

variance

import pickle

# Save the trained model to disk

filename = 'multiple_linear_regression_model.pkl'

# Open a file in write-binary mode and dump the model

with open(filename, 'wb') as file:

    pickle.dump(regressor, file)
```

```python
print("Model has been pickled and saved as multiple_linear_regression_model.pkl")
```

app.py

```python
import streamlit as st

import pickle

import numpy as np

import pandas as pd

model = pickle.load(open(r'C:\Users\omkar joshi\DATA
SCIENCE\multiple_linear_regression_model.pkl', 'rb'))

# Load dataset (for feature reference)

dataset = pd.read_csv(r"C:\Users\omkar joshi\DATA SCIENCE\Databases\Investment.csv")

X = dataset.iloc[:, :-1]

X = pd.get_dummies(X, dtype=int)  # handle categorical columns

st.set_page_config(page_title="Investment Prediction App", page_icon="📈",
layout="centered")

st.markdown("""

    <style>

        .main {

            background-color: #0E1117;

            color: #FAFAFA;

            font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;

        }

        h1 {

            text-align: center;

            color: #00C9A7;

        }

        .stButton button {

            background-color: #00C9A7;

            color: white;
```

```
        border-radius: 12px;

        height: 3em;

        width: 100%;

        font-size: 18px;

    }

    .stButton button:hover {

        background-color: #00B894;

        color: white;

    }

    .prediction-box {

        padding: 15px;

        border-radius: 12px;

        background-color: #1B4332;

        color: #D8F3DC;

        text-align: center;

        font-size: 20px;

        font-weight: bold;

    }

    </style>
""", unsafe_allow_html=True)

st.title("📈 Investment Prediction App")

st.markdown("### Predict the **Investment amount** using a **Multiple Linear Regression
Model** trained on financial dataset.")

st.subheader("🔢 Enter Input Features")

user_inputs = {}

for col in X.columns:

    if X[col].nunique() == 2:  # Dummy variable

        user_inputs[col] = st.selectbox(f"{col}", [0, 1])
```

```python
    else:

        user_inputs[col] = st.number_input(f"{col}", float(X[col].min()), float(X[col].max()),
float(X[col].mean()))

# Convert input into DataFrame

input_df = pd.DataFrame([user_inputs])

if st.button(" 🧙  Predict Investment"):

    prediction = model.predict(input_df)[0]

    st.markdown(f"<div class='prediction-box'> 💰  The predicted Investment based on given
inputs is: <br><br> **${prediction:,.2f}**</div>", unsafe_allow_html=True)

st.markdown("---")

st.markdown(" 📊  *This model was trained using a dataset of investment-related
features.*")
```

output:

📈 Investment Prediction App

Predict the Investment amount using a Multiple Linear Regression Model trained on financial dataset.

🔢 Enter Input Features

DigitalMarketing

```
73721.62                                              −    +
```

Promotion

```
121344.64                                             −    +
```

Research

```
211025.10                                             −    +
```

State_Bangalore

```
0                                                          ⌄
```

State_Chennai

```
0                                                          ⌄
```

State_Hyderabad

```
0                                                          ⌄
```

🤖 Predict Investment

💰 The predicted Investment based on given inputs is:

$111,204.78

📊 This model was trained using a dataset of investment-related features.