Sec 1 Lecture 2-Logistic Regression

January 19, 2024

1 1.) Pull in Data and Convert ot Monthly

[1]:
```
import yfinance as yf
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

[2]:
```
apple_data = yf.download('AAPL')
df = apple_data.resample("M").last()[["Adj Close"]]
```

```
[**********************100%%**********************]  1 of 1 completed
```

2 2.) Create columns.

- Current Stock Price, Difference in stock price, Whether it went up or down over the next month, option premium

[13]:
```
# Difference in stockprice
df['Diff'] = df['Adj Close'].diff().shift(-1)

# Target up or down
df['Target'] = np.sign(df['Diff'])

# Option Premium
df['Premium'] = .08 * df['Adj Close']
```

[14]:
```
df.head()
```

[14]:

Date	Adj Close	Diff	Target	Premium
1980-12-31	0.117887	-0.020296	-1.0	0.009431
1981-01-31	0.097592	-0.006045	-1.0	0.007807
1981-02-28	0.091546	-0.006909	-1.0	0.007324
1981-03-31	0.084637	0.013386	1.0	0.006771
1981-04-30	0.098023	0.016409	1.0	0.007842

3 3.) Pull in X data, normalize and build a LogReg on column 2

[15]:
```
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn import metrics
```

[16]:
```
X = pd.read_csv("Xdata.csv", index_col="Date", parse_dates=["Date"])
```

[28]:
```
y = df.loc[:"2023-09-30","Target"].copy()

df = df.loc[:"2023-09-30",:].copy()
```

[30]:
```
logreg = LogisticRegression()

logreg.fit(X,y)

y_pred = logreg.predict(X)
```

[29]:
```
y_pred
```

[29]:
```
array([-1., -1., -1.,  1.,  1., -1., -1., -1., -1.,  1., -1.,  1., -1.,
       -1., -1., -1., -1., -1.,  1.,  1.,  1.,  1.,  1., -1.,  1.,  1.,
       -1.,  1.,  1., -1., -1.,  1., -1., -1., -1.,  1.,  1.,  1., -1.,
        1., -1., -1.,  1.,  1., -1., -1., -1.,  1., -1., -1., -1., -1.,
       -1.,  1., -1., -1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,
       -1.,  1.,  1., -1.,  1.,  1.,  1.,  1.,  1.,  1.,  1., -1.,  1.,
        1.,  1.,  1., -1., -1.,  1., -1.,  1., -1.,  1.,  1.,  1., -1.,
       -1.,  1., -1., -1.,  1., -1., -1., -1.,  1.,  1., -1., -1.,  1.,
       -1.,  1., -1., -1., -1.,  1.,  1., -1.,  1.,  1., -1., -1., -1.,
        1.,  1.,  1.,  1.,  1.,  1., -1., -1., -1., -1.,  1., -1.,  1.,
       -1.,  1.,  1.,  1., -1.,  1., -1., -1., -1., -1., -1.,  1.,  1.,
        1., -1., -1., -1., -1.,  1., -1., -1., -1., -1.,  1.,  1., -1.,
        1.,  1., -1., -1., -1., -1.,  1., -1.,  1., -1.,  1., -1.,  1.,
       -1., -1.,  1.,  1.,  1., -1., -1., -1., -1.,  1., -1., -1., -1.,
       -1.,  1.,  1., -1.,  1.,  1., -1.,  1.,  1., -1., -1., -1.,  1.,
       -1., -1., -1.,  1.,  1., -1., -1.,  1., -1.,  1.,  1.,  1., -1.,
       -1.,  1.,  1., -1.,  1., -1., -1.,  1.,  1., -1.,  1.,  1., -1.,
        1.,  1.,  1., -1.,  1.,  1.,  1.,  1.,  1.,  1., -1., -1.,  1.,
       -1.,  1., -1., -1., -1., -1.,  1., -1.,  1.,  1., -1.,  1., -1.,
       -1., -1.,  1., -1.,  1.,  1., -1.,  1.,  1., -1., -1., -1., -1.,
       -1.,  1., -1.,  1.,  1.,  1., -1.,  1., -1.,  1.,  1.,  1., -1.,
        1., -1.,  1.,  1.,  1.,  1., -1.,  1.,  1., -1.,  1.,  1.,  1.,
        1., -1.,  1.,  1., -1., -1.,  1., -1.,  1.,  1.,  1.,  1.,  1.,
        1.,  1., -1., -1.,  1., -1., -1.,  1., -1.,  1.,  1.,  1.,  1.,
        1.,  1.,  1.,  1.,  1.,  1., -1.,  1.,  1.,  1., -1.,  1., -1.,
```

```
-1.,  1.,  1.,  1., -1., -1.,  1., -1., -1., -1., -1.,  1., -1.,
 1.,  1.,  1., -1.,  1.,  1.,  1.,  1.,  1.,  1., -1.,  1.,  1.,
 1., -1., -1.,  1., -1.,  1.,  1.,  1.,  1.,  1.,  1., -1.,  1.,
-1., -1.,  1., -1.,  1.,  1., -1.,  1.,  1.,  1.,  1., -1., -1.,
 1.,  1.,  1.,  1., -1., -1., -1., -1., -1.,  1.,  1.,  1., -1.,
 1.,  1., -1.,  1.,  1.,  1., -1.,  1.,  1.,  1.,  1.,  1.,  1.,
 1., -1.,  1.,  1., -1.,  1.,  1., -1.,  1.,  1., -1., -1., -1.,
-1.,  1., -1., -1., -1., -1.,  1., -1.,  1., -1.,  1.,  1.,  1.,
 1., -1.,  1.,  1.,  1.,  1., -1.,  1., -1.,  1.,  1., -1.,  1.,
 1., -1., -1.,  1., -1., -1.,  1., -1.,  1.,  1., -1., -1., -1.,
-1.,  1.,  1.,  1.,  1., -1.,  1.,  1., -1.,  1.,  1.,  1.,  1.,
 1., -1., -1.,  1.,  1.,  1.,  1.,  1., -1., -1.,  1.,  1., -1.,
 1.,  1.,  1., -1.,  1.,  1.,  1., -1.,  1.,  1.,  1., -1., -1.,
 1., -1., -1., -1.,  1., -1., -1.,  1., -1., -1.,  1.,  1.,  1.,
-1.,  1.,  1.,  1., -1., -1., -1.])
```

4 4.) Add columns, prediction and profits.

[31]:
```
df['Predictions'] = y_pred
```

[37]:
```
df['Profits'] = 0

# True Positives
df.loc[(df['Predictions'] == 1) & (df['Target'] == 1),"Profits"] = df["Premium"]

# False Positives
df.loc[(df['Predictions'] == 1) & (df['Target'] == -1),"Profits"] = 100 *␣
 ↪df["Diff"] + df["Premium"]

# True Negatives
# = 0

# False Negatives
# = 0
```

[36]:
```
df
```

[36]:

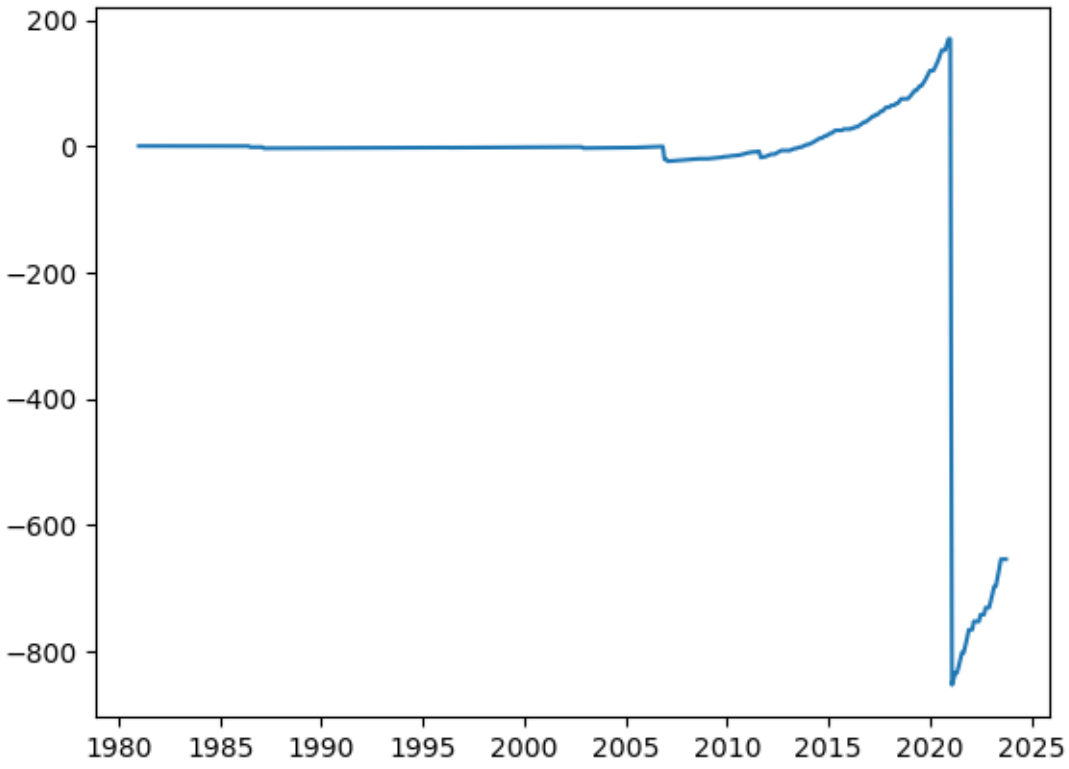
Date	Adj Close	Diff	Target	Premium	Profits	Predictions
1980-12-31	0.117887	-0.020296	-1.0	0.009431	0.000000	-1.0
1981-01-31	0.097592	-0.006045	-1.0	0.007807	0.000000	-1.0
1981-02-28	0.091546	-0.006909	-1.0	0.007324	0.000000	-1.0
1981-03-31	0.084637	0.013386	1.0	0.006771	0.006771	1.0
1981-04-30	0.098023	0.016409	1.0	0.007842	0.007842	1.0
…	…	…	…	…	…	…
2023-05-31	176.778061	16.675507	1.0	14.142245	14.142245	1.0

```
2023-06-30  193.453568   2.473389     1.0  15.476285  15.476285          1.0
2023-07-31  195.926956  -8.304138    -1.0  15.674156   0.000000         -1.0
2023-08-31  187.622818 -16.638077    -1.0  15.009825   0.000000         -1.0
2023-09-30  170.984741  -0.439423    -1.0  13.678779   0.000000         -1.0

[514 rows x 6 columns]
```

5 5.) Plot profits over time

[39]:
```
plt.plot(np.cumsum(df['Profits']))
plt.show()
```



6 Skills from the MQE to help Mr.Lius ventures

1. Ability to analyze user behavior and engagement on StarsArena, optimizing features and pricing strategies.
2. Skilled in evaluating the economic impact of business strategies, crucial for decision-making in the dynamic environment.
3. Strategic Decision Making: Trained in making informed decisions, vital for navigating the competitive landscape of the SocialFi space.

4. Market Research: Expertise in understanding market trends and user preferences, essential for StarsArena's growth and user experience enhancement.

7 6.) Create a loop that stores total profits over time

[ ]:

8 7.) What is the optimal threshold and plot the total profits for this model.

[ ]: