1.) Pull in Data and Convert ot Monthly

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

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

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

2.) Create columns.

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

In [17]:
```
# difference in stock price
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"]
```

In [18]:
```
df.head()
```

Out[18]:

Date	Adj Close	Diff	Target	Premium
1980-12-31	0.117887	-0.020296	-1.0	0.009431
1981-01-31	0.097591	-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.) Pull in X data, normalize and build a LogReg on column 2

In [3]:
```
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
```

In [11]:
```
X = pd.read_csv("Xdata.csv", index_col="Date", parse_dates=["Date"])
```

In [21]:
```
y = df.loc[:"2023-09-30","Target"].copy()
df = df.loc[:"2023-09-30",:].copy()
```

In [22]:
```
logreg = LogisticRegression()
logreg.fit(X,y)
y_pred = logreg.predict(X)
```

4.) Add columns, prediction and profits.

In [23]:
```
df["Predictions"] = y_pred
```

In [32]:
```
df["Profits"] = 0.
```

In [35]:
```
# 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"] +

#True Negative = 0
```

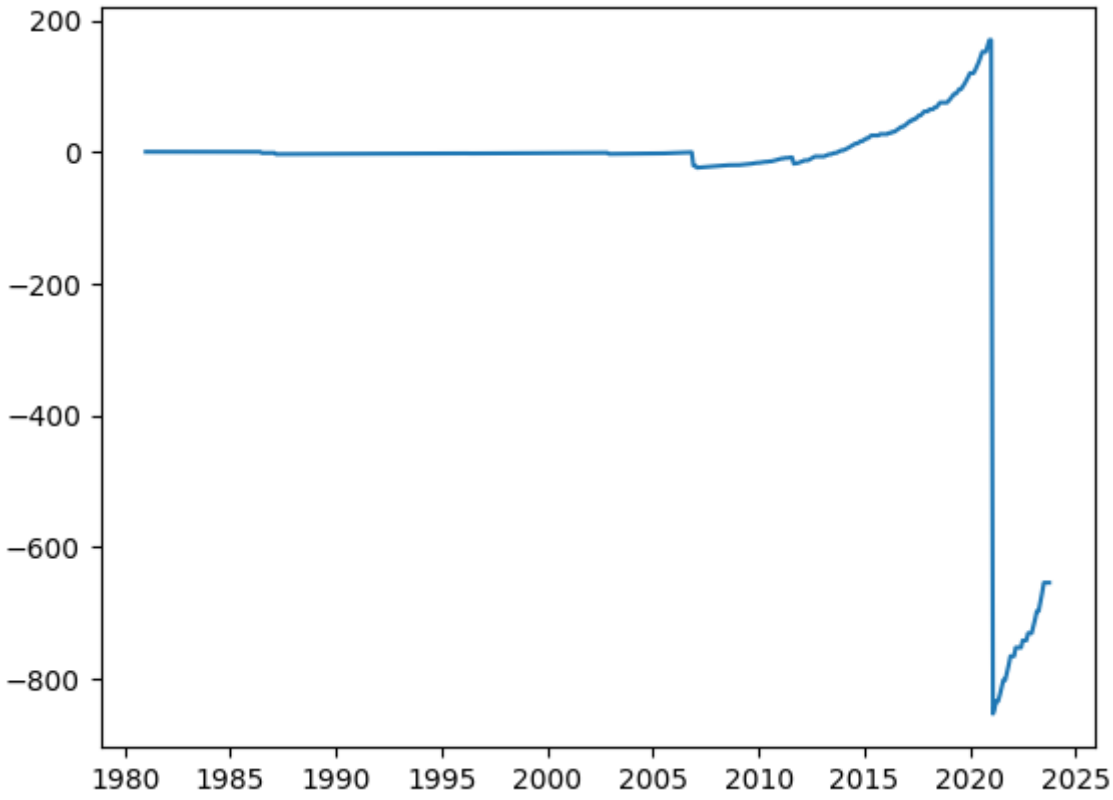
In [38]:
```
df.head()
```

Out[38]:

Date	Adj Close	Diff	Target	Premium	Predictions	Profits
1980-12-31	0.117887	-0.020296	-1.0	0.009431	-1.0	0.000000
1981-01-31	0.097591	-0.006045	-1.0	0.007807	-1.0	0.000000
1981-02-28	0.091546	-0.006909	-1.0	0.007324	-1.0	0.000000
1981-03-31	0.084637	0.013386	1.0	0.006771	1.0	0.006771
1981-04-30	0.098023	0.016409	1.0	0.007842	1.0	0.007842

5.) Plot profits over time

In [37]:
```
plt.plot(np.cumsum(df["Profits"]))
plt.show()
```



5.5 how you see your skills valuable to PJ and/or Philip Liu

Based on the knowledge I learnt from ECON410 guest speaker, I would recommend a start-up blockchain company to improve their security level, incluing conducting regular audits, implementing rigorous security protocols, and establishing a bug bounty program to encourage white hat hackers to find and report vulnerabilities.
Regarding my data analysis ability, I can adopte Python/R and SQL to conduct user behavior analysis, in terms of features engineering and NLP analysis for the features that are most valued by the community.In addition, I can use data analytics skill to help interpret market trends and user sentiment, allowing the platform to adapt quickly to changing market conditions or to anticipate them.

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

In [ ]:

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

In [ ]: