

## Econ 441b HW 3

```
In [1]: import pandas as pd
import sklearn as sk
from sklearn.linear_model import LinearRegression
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import Lasso
```

Intel MKL WARNING: Support of Intel(R) Streaming SIMD Extensions 4.2 (Intel(R) SSE4.2) enabled only processors has been deprecated. Intel oneAPI Math Kernel Library 2025.0 will require Intel(R) Advanced Vector Extensions (Intel(R) AVX) instructions.

Intel MKL WARNING: Support of Intel(R) Streaming SIMD Extensions 4.2 (Intel(R) SSE4.2) enabled only processors has been deprecated. Intel oneAPI Math Kernel Library 2025.0 will require Intel(R) Advanced Vector Extensions (Intel(R) AVX) instructions.

### 0.) Clean the Apple Data to get a quarterly series of EPS.

```
In [2]: y = pd.read_csv('AAPL_quarterly_financials.csv')
y.index = y.name
y = pd.DataFrame(y.loc['BasicEPS', :]).iloc[2:,:]
y.index = pd.to_datetime(y.index)
#check if Nas are no dividend period
y.sort_index().fillna(0.)
```

Out [2]:

	BasicEPS
1985-09-30	0.0
1985-12-31	0.004
1986-03-31	0.002
1986-06-30	0.002
1986-09-30	0.0
...	...
2022-09-30	1.29
2022-12-31	1.89
2023-03-31	1.53
2023-06-30	1.27
2023-09-30	1.47

153 rows × 1 columns

1.) Come up with 6 search terms you think could nowcast earnings. (Different than the ones I used) Add in 3 terms that that you think will not Nowcast earnings. Pull in the gtrends data. Clean it to have a quarterly average.

In [3]: `pip install pytrends`

Requirement already satisfied: pytz in /Users/qiupinxuan/opt/anaconda3/lib/python3.9/site-packages (4.9.2)

Requirement already satisfied: requests>=2.0 in /Users/qiupinxuan/opt/anaconda3/lib/python3.9/site-packages (from pytz) (2.31.0)

Requirement already satisfied: pandas>=0.25 in /Users/qiupinxuan/opt/anaconda3/lib/python3.9/site-packages (from pytz) (1.5.3)

Requirement already satisfied: lxml in /Users/qiupinxuan/opt/anaconda3/lib/python3.9/site-packages (from pytz) (4.9.3)

Requirement already satisfied: python-dateutil>=2.8.1 in /Users/qiupinxuan/opt/anaconda3/lib/python3.9/site-packages (from pandas>=0.25->pytz) (2.8.2)

Requirement already satisfied: pytz>=2020.1 in /Users/qiupinxuan/opt/anaconda3/lib/python3.9/site-packages (from pandas>=0.25->pytz) (2023.3.post1)

Requirement already satisfied: numpy>=1.20.3 in /Users/qiupinxuan/opt/anaconda3/lib/python3.9/site-packages (from pandas>=0.25->pytz) (1.23.5)

Requirement already satisfied: charset-normalizer<4,>=2 in /Users/qiupinxuan/opt/anaconda3/lib/python3.9/site-packages (from requests>=2.0->pytz) (2.0.4)

Requirement already satisfied: idna<4,>=2.5 in /Users/qiupinxuan/opt/anaconda3/lib/python3.9/site-packages (from requests>=2.0->pytz) (3.4)

Requirement already satisfied: urllib3<3,>=1.21.1 in /Users/qiupinxuan/opt/anaconda3/lib/python3.9/site-packages (from requests>=2.0->pytz) (1.26.16)

Requirement already satisfied: certifi>=2017.4.17 in /Users/qiupinxuan/opt/anaconda3/lib/python3.9/site-packages (from requests>=2.0->pytz) (2023.11.17)

Requirement already satisfied: six>=1.5 in /Users/qiupinxuan/opt/anaconda3/lib/python3.9/site-packages (from python-dateutil>=2.8.1->pandas>=0.25->pytz) (1.16.0)

WARNING: There was an error checking the latest version of pip.

Note: you may need to restart the kernel to use updated packages.

In [4]: `from pytz.request import TrendReq`

In [19]: `# Create pytz object  
pytz = TrendReq(hl='en-US', tz=360)  
  
# Set up the keywords and the timeframe  
keywords = ["Iphone", "Covid","Huawei","Interest Rates", "ucla","Tylor Swift tickets", "Apple Share"]  
# Add your keywords here: Covid, Huawei, war, ucla, chip storage, Tylor Swift tickets.  
start_date = '2004-01-01'  
end_date = '2024-01-01'  
  
# Create an empty DataFrame to store the results  
df = pd.DataFrame()  
  
# Iterate through keywords and fetch data  
for keyword in keywords:  
 #time.sleep(5)  
 pytz.build_payload([keyword], cat=0, timeframe=f'{start_date} {end_date}', geo='', gprop='')`

```
interest_over_time_df = pytrends.interest_over_time()
df[keyword] = interest_over_time_df[keyword]
```

```
In [20]: X = df.resample('Q').mean()
X
```

```
Out[20]:
```

	Iphone	Covid	Huawei	Interest Rates	ucla	Tylor Swift tickets	Apple Share
date							
2004-03-31	0.000000	0.000000	0.0	60.000000	88.333333	0.000000	9.666667
2004-06-30	0.000000	0.000000	1.0	64.333333	82.666667	0.000000	8.333333
2004-09-30	0.000000	0.000000	1.0	53.000000	74.333333	22.666667	8.000000
2004-12-31	0.000000	0.000000	1.0	46.333333	85.000000	33.333333	9.333333
2005-03-31	0.000000	0.000000	1.0	48.333333	82.666667	13.333333	12.000000
...	...	...	...	...	...	...	...
2023-03-31	46.000000	7.666667	28.0	83.000000	41.333333	2.666667	53.666667
2023-06-30	44.666667	5.000000	26.0	73.000000	29.000000	6.333333	49.666667
2023-09-30	53.333333	6.000000	28.0	73.666667	29.000000	13.333333	59.000000
2023-12-31	51.000000	6.333333	27.0	67.666667	38.666667	4.333333	51.000000
2024-03-31	47.000000	6.000000	25.0	70.000000	30.000000	4.000000	51.000000

81 rows x 7 columns

```
In [21]: # fix data
temp = pd.concat([y,X],axis =1).dropna()
y = temp[['BasicEPS']].copy()
X = temp.iloc[:,1:].copy()
```

## 2.) Normalize all the X dataImport data.

```
In [22]: scaler = StandardScaler()
```

```
In [23]: X_scaled = scaler.fit_transform(X)
```

## 4.) Run a Lasso with lambda of .1. Plot a bar chart.

```
In [24]: lasso = Lasso (alpha=0.1)
```

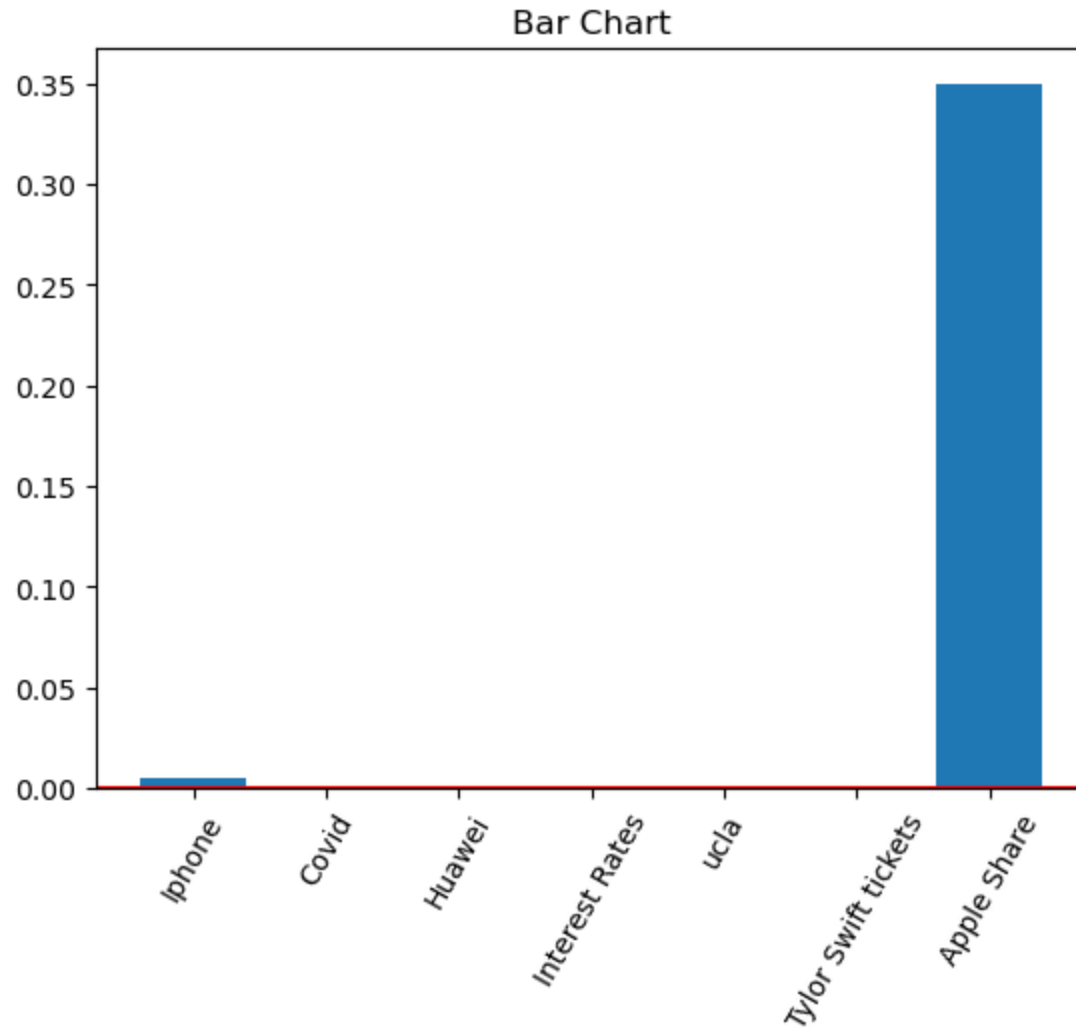
```
In [25]: lasso.fit(X_scaled,y)
```

```
Out[25]: Lasso(alpha=0.1)
```

```
In [26]: coef = lasso.coef_  
coef
```

```
Out[26]: array([ 0.00445588,  0.          ,  0.          ,  0.          , -0.          ,  
                -0.          ,  0.34972203])
```

```
In [27]: import matplotlib.pyplot as plt  
#plt.figure(figsize=())  
plt.bar(range(len(coef)),coef)  
plt.xticks(range(len(coef)), X.columns, rotation=60)  
plt.axhline(0,color='r')  
plt.title('Bar Chart')  
plt.show()
```



## 5.) Do these coefficient magnitudes make sense?

It means when we could consider apple share and iphone as an important search term when nowcasting earnings. Other coefficients of variables like covid, ucla, Tylor Swift tickets that are not related to the earnings are obvious to be zero.