

# 주식 데이터 시각화 및 예측 웹 서비스

- 사용 데이터 : Yahoo Finance 에서 크롤링
- 웹 : Streamlit
- 시각화 : Pyplot
- 예측 : sklearn. LinearRegression

## 1. 데이터 크롤링 및 로딩

```
data_load_state = st.text('Loading data...')

# Load data from yahoo finance.
start = st.sidebar.date_input("Start date",dt.datetime(2017,1,1))
#end = st.sidebar.date_input('End date', datetime(2022,1,1))
# start=dt.date(2021,1,1)
end=dt.date.today()
#st.markdown("원하는 회사를 입력해주세요. *ex)apple=AAPL, google=G00G, samsung=005930.KS*")
stocklist=('AAPL','GOOG','TSLA','MSFT','005930.KS','LPL','000660.KS')
st.markdown("`Apple = 'AAPL', Google = 'GOOG', Tesla = 'TSLA', Microsoft = 'MSFT', Samsung='005930.KS', LG Display = 'LPL', SK Hynix = '000660.KS'`")

option = st.selectbox('원하는 회사를 선택해주세요',stocklist)
#user_input=st.text_input('', 'AAPL')
data=pdr.get_data_yahoo(option,start,end)
#data=pdr.get_data_yahoo("005930.KS", start, end)

#fill nan vale with next value within columns
data.fillna(method="ffill",inplace=True)
```

- 웹에 설정된 체크 박스와 달력을 통해 원하는 종목과 날짜를 선택합니다.
- 선택된 종목과 날짜에 해당하는 데이터를 Yahoo Finance로 부터 크롤링합니다.

Start date

2017/03/01

← March 2017 →

S	M	T	W	T	F	S
		1	2	3	4	
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

## 주식 데이터 시각화 및 예측

Yahoo Finance로부터 가져온 주식 데이터를 시각화 및 분석하고 미래 주가를 예측합니다.

좌측 사이드 바를 이용하여 원하는 시작 날짜를 정할 수 있습니다.

Loading data...done!

Apple = 'AAPL', Google = 'GOOG', Tesla = 'TSLA', Microsoft = 'MSFT', Samsung = '005930.KS', LG Display = 'LPL', SK Hynix = '000660.KS'

원하는 회사를 선택해주세요

TSLA
AAPL
GOOG
TSLA
MSFT
005930.KS
LPL
000660.KS

min	12.4453	11.7993	12.0733	11.9313	29,401,800.0000	11.9313
-----	---------	---------	---------	---------	-----------------	---------

## 2. 데이터 처리 및 데이터 프레임 생성

```
# Notify the reader that the data was successfully loaded.
data_load_state.text('Loading data...done!')
st.markdown("")
# create checkbox
if st.checkbox('Show raw data'):
    st.subheader(f'Raw data of {option}')
    st.write(data)

# show the description of data
st.subheader(f'Detail description about Datasets : {option}')
describ=data.describe()
st.write(describ)

#create new columns like year, month, day
data["Year"]=data.index.year
data["Month"]=data.index.month
data["Weekday"]=data.index.day_name()
```

- pandas를 이용해 데이터를 데이터 프레임 형식으로 저장하고, describe를 통해 요약된 데이터를 보여줍니다.
- 체크박스를 이용하여 raw data의 테이블 여부를 선택하여 볼 수 있습니다.

Start date

2017/03/01

Start date

2017/03/01

좌측 사이트 바를 이용하여 원하는 시작 날짜를 정할 수 있습니다. 📅

Loading data...done!

Apple = 'AAPL', Google = 'GOOG', Tesla = 'TSLA', Microsoft = 'MSFT', Samsung = '005930.KS', LG Display = 'LPL', SK Hynix = '000660.KS'

원하는 회사를 선택해주세요

AAPL

☒ Show raw data

#### Raw data of AAPL

	High	Low	Open	Close	Volume	Adj Close
2017-0	34.3600	34.1750	34.2700	34.2475	93,931,600.0000	32.2429
2017-0	35.0375	34.4000	34.4725	34.9475	145,658,400.0000	32.9019
2017-0	35.0700	34.6900	35.0000	34.7400	104,844,600.0000	32.7066
2017-0	34.9575	34.6475	34.6950	34.9450	84,432,400.0000	32.8996
2017-0	34.9425	34.6500	34.8425	34.8350	87,000,000.0000	32.7960
2017-0	34.9950	34.6975	34.7650	34.8800	69,785,200.0000	32.8384
2017-0	34.9500	34.7050	34.7375	34.7500	74,828,800.0000	32.7160
2017-0	34.6975	34.2625	34.6850	34.6700	88,623,600.0000	32.6407
2017-0	34.8400	34.6600	34.8125	34.7850	78,451,200.0000	32.7489
2017-0	34.8575	34.7050	34.7125	34.8000	69,686,800.0000	32.7630

#### Detail description about Datasets : AAPL

	High	Low	Open	Close	Volume	Adj Close
count	1,397.0000	1,397.0000	1,397.0000	1,397.0000	1,397.0000	1,397.0000
mean	87.5059	85.5531	86.5071	86.5709	116,939,586.9005	85.2446
std	47.7318	46.5102	47.1191	47.1413	54,158,472.1376	47.5977
min	34.3600	34.1750	34.2700	34.2475	41,000,000.0000	32.2429
25%	44.7050	43.7675	44.1050	44.2225	80,819,200.0000	42.4948
50%	65.6975	64.0925	64.5775	65.0350	102,785,600.0000	63.8190
75%	133.5600	130.2400	132.1300	131.9400	136,616,400.0000	130.7801
max	182.9400	179.1200	182.6300	182.0100	426,510,000.0000	181.2599

#### 'Open,High,Low,Close' Line Chart : AAPL

### 3. 데이터 시각화

```
fig_close = px.line(data[['Open','High','Low','Close']])
st.subheader(f"'Open,High,Low,Close' Line Chart : {option}")
st.plotly_chart(fig_close)
# st.subheader('Graph of Close & Open:-')
# st.line_chart(data[['Open','High','Low','Close']])

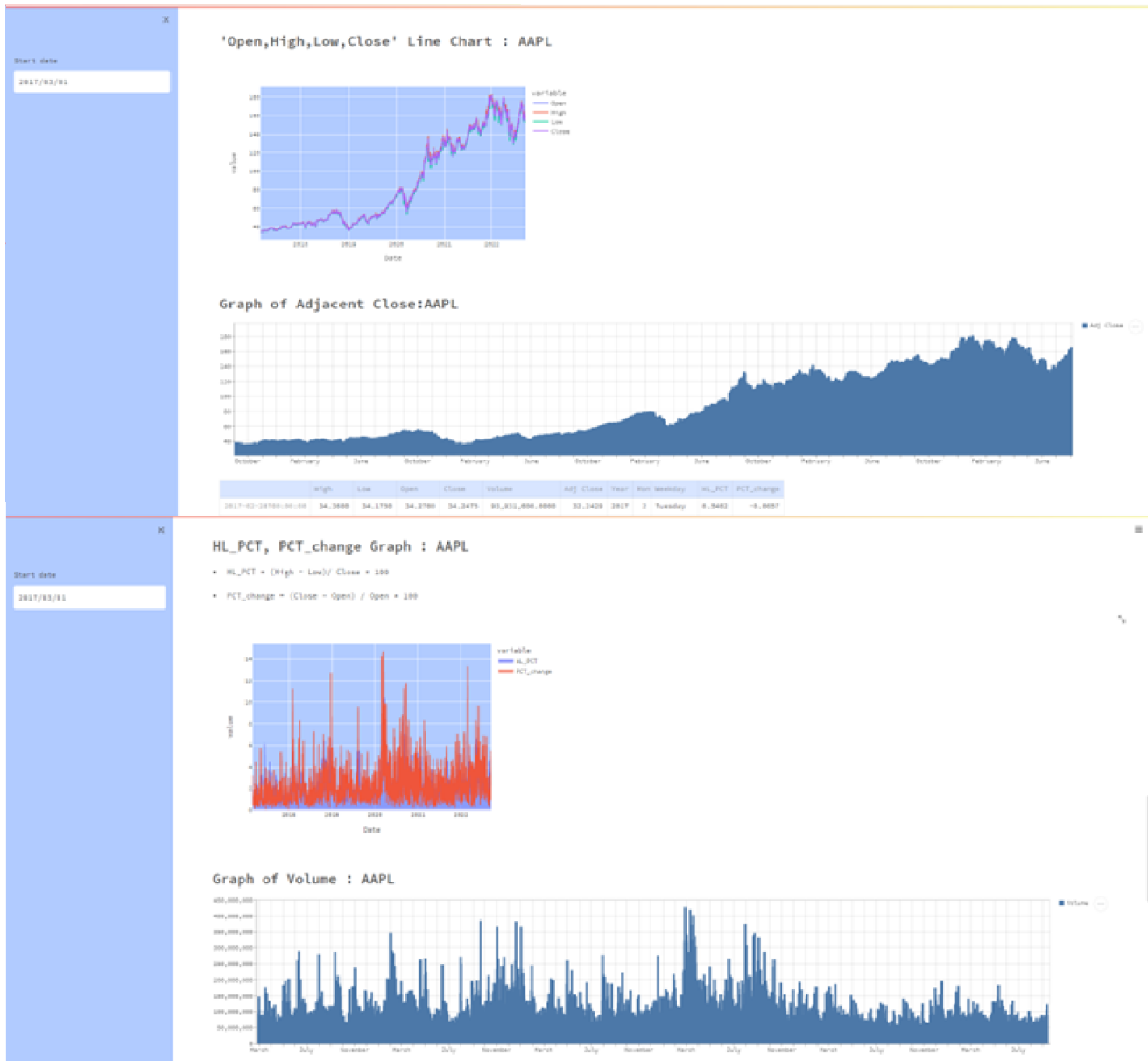
# display plot of Adj Close column in datasets
st.subheader(f'Graph of Adjacent Close:{option}')
st.bar_chart(data['Adj Close'])

#High-low
data['HL_PCT'] = (data['High'] - data['Low']) / data['Close'] * 100.0
data['PCT_change'] = (data['Close'] - data['Open']) / data['Open'] * 100.0
data
st.subheader(f'HL_PCT, PCT_change Graph : {option}')
st.markdown(""" HL_PCT = (High - Low)/ Close * 100""")
st.markdown(""" PCT_change = (Close - Open) / Open * 100""")
fig_1=px.area(data[['HL_PCT','PCT_change']])
st.plotly_chart(fig_1)

# display plot of volume column in datasets
st.subheader(f'Graph of Volume : {option}')
st.bar_chart(data['Volume'])
```

```
# create new cloumn for data analysis.
data['HL_PCT'] = (data['High'] - data['Low']) / data['Close'] * 100.0
data['PCT_change'] = (data['Close'] - data['Open']) / data['Open'] * 100.0
data = data[['Adj Close', 'HL_PCT', 'PCT_change', 'Volume']]
```

- pyplot을 이용하여 dataframe에 저장된 데이터를 시각화 합니다.
- 해당 종목의 'Open', 'High','Low','Close'의 Line chart, 주가 변화율에 대한 Area chart 등을 생성합니다.



## 4. 주가 예측

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2)
clf = LinearRegression(n_jobs=-1)
clf.fit(X_train, y_train)

confidence = clf.score(X_test, y_test)
```

```

# display the accuracy of forecast value.
st.subheader('Accuracy:')
st.write(confidence)

forecast_set = clf.predict(X_lately)
data['Forecast'] = np.nan

last_date = data.iloc[-1].name
last_unix = last_date.timestamp()
one_day = 86400
next_unix = last_unix + one_day

for i in forecast_set:
    next_date = datetime.datetime.fromtimestamp(next_unix)
    next_unix += 86400
    data.loc[next_date] = [np.nan for _ in range(len(data.columns)-1)]+[i]
    last_date = data.iloc[-1].name
    dti = pd.date_range(last_date, periods=forecast_out+1, freq='D')
    index = 1
for i in forecast_set:
    data.loc[dti[index]] = [np.nan for _ in range(len(data.columns)-1)] + [i]
    index +=1

# display the forecast value.
st.subheader(f'Forecast value : {option} with LinearRegression')
st.dataframe(data.tail(50))

# display the graph of adj close and forecast columns
st.subheader(f'Graph of Adj Close and Forecast : {option}')
st.area_chart(data[["Adj Close", "Forecast"]])

```

- Sklearn의 LinearRegression 라이브러리를 이용하여 주가를 예측합니다.
- 실제 주가와 비교하여 정확도(Accuracy)를 측정합니다.

X

Start date

2017/03/01

Accuracy:

0.9768048302143317

Forecast value : AAPL with LinearRegression

	Adj Close	Hi_PCT	PCT_change	Volume	Labels	Forecast
2022-09-12T09:00:00	<NA>	<NA>	<NA>	<NA>	<NA>	158.1544
2022-09-13T09:00:00	<NA>	<NA>	<NA>	<NA>	<NA>	159.8741
2022-09-14T09:00:00	<NA>	<NA>	<NA>	<NA>	<NA>	156.7210
2022-09-15T09:00:00	<NA>	<NA>	<NA>	<NA>	<NA>	155.5606
2022-09-16T09:00:00	<NA>	<NA>	<NA>	<NA>	<NA>	157.8489
2022-09-17T09:00:00	<NA>	<NA>	<NA>	<NA>	<NA>	155.5935
2022-09-18T09:00:00	<NA>	<NA>	<NA>	<NA>	<NA>	158.5462
2022-09-19T09:00:00	<NA>	<NA>	<NA>	<NA>	<NA>	164.4768
2022-09-20T09:00:00	<NA>	<NA>	<NA>	<NA>	<NA>	154.5027
2022-09-21T09:00:00	<NA>	<NA>	<NA>	<NA>	<NA>	156.3692

Graph of Adj Close and Forecast : AAPL

