Stock Market Prediction

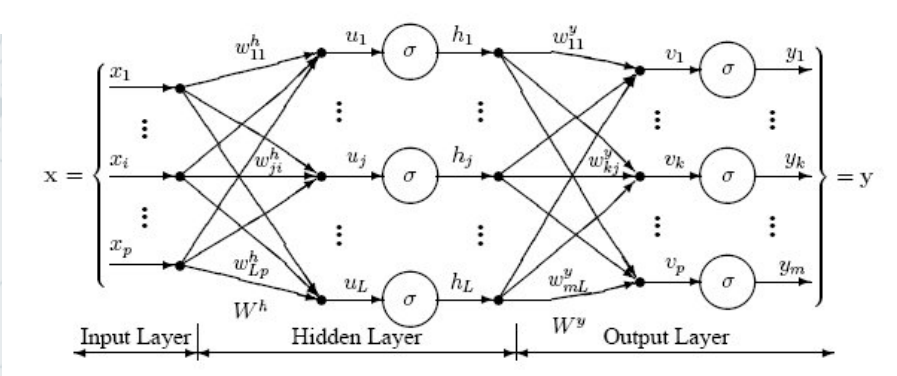
Ning Chen

Agenda

- Overview
- Business Understanding
- Data Collection
- Exploratory Data Analysis
- Classification
- Time Series
- Sentimental
- Frontend
- Next Steps

Overview

Accurate prediction of stock market asset is a significant and challenging task due to complicated nature of the financial stock markets.



Business Goals

Stock market prediction aims to determine the future movement of the stock value of a financial exchange.

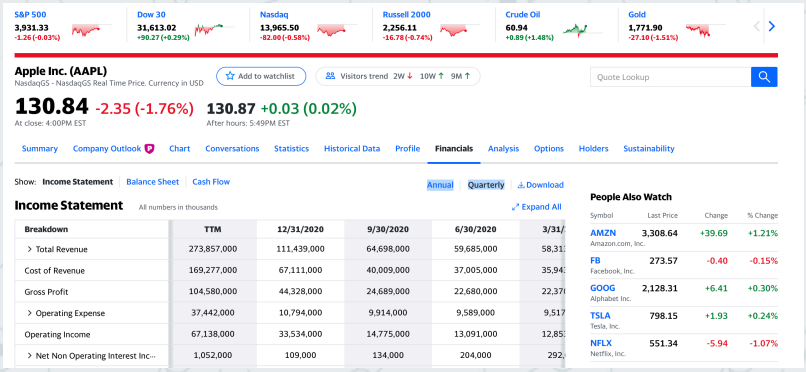
Classification: Trade Strategy (buy sell hold)

random forest, SVM, Xgboost, NN

time series regression model: SARIMAX, Facebook Prophet, LSTM

sentimental analysis: NLP

Data Collection



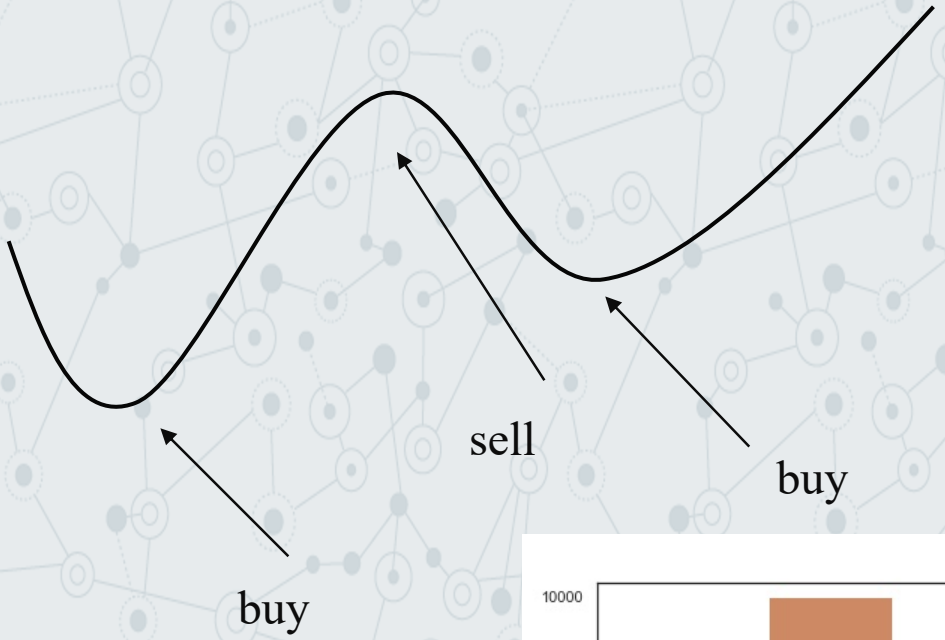
Data was collected from three different web sources by API calls or Web Scraping.

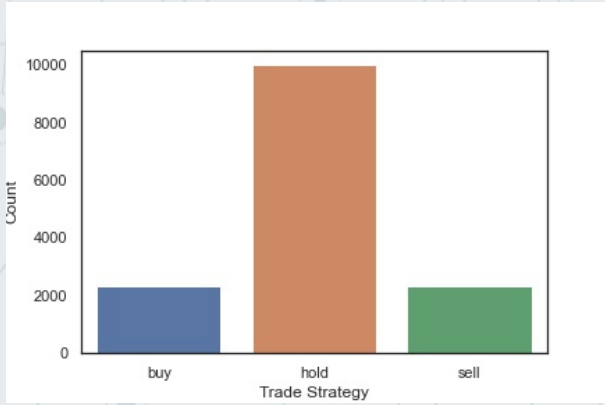
-Quarterly Report for Classification by Web Scrapping.
-Yahoo Finance and IEX API for Time Series by API calls.
-Twitter for sentimental data by VADER.

Exploratory Data Analysis

Trade Strategy

local minimum of the price to buy, local maximum of the price to sell, and all other time to hold.





Exploratory Data Analysis

Missing data

The missing data for weekends and holidays was filled by interpolation method.

The missing data of exogenous features was filled by propagating nearest valid observation backward/forward to next valid observation.

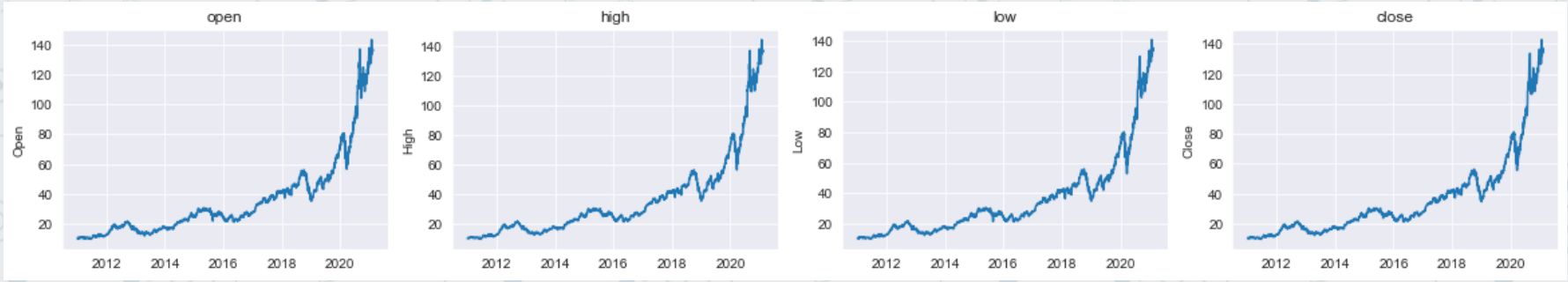
$$P_t = \sqrt[3]{P_{t-1}^{\,2} * P_{t+2}}$$

$$P_{t+1} = \sqrt[3]{P_{t-1} * P_{t+2}^{\,2}}$$

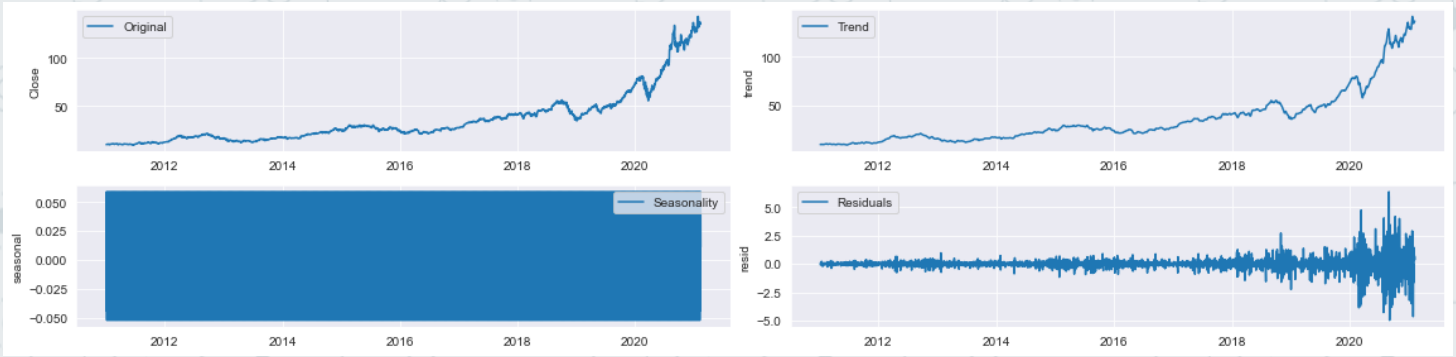
The fact that data are missing should not be neglected—quite often it is an indication of illiquidity. Using an average price results in an underestimate of volatility.

Exploratory Data Analysis

stock price

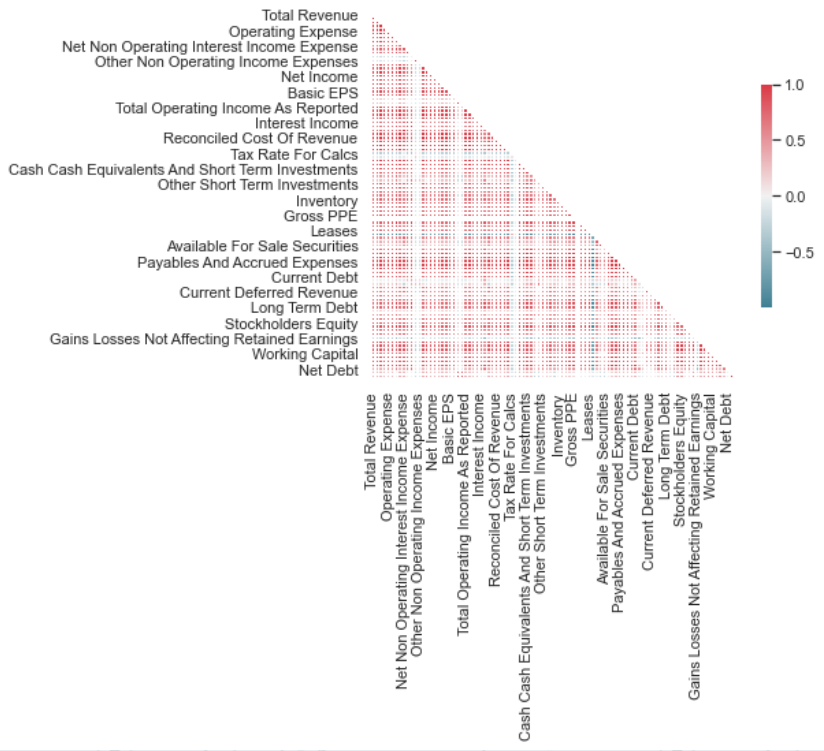


seasonal decompose

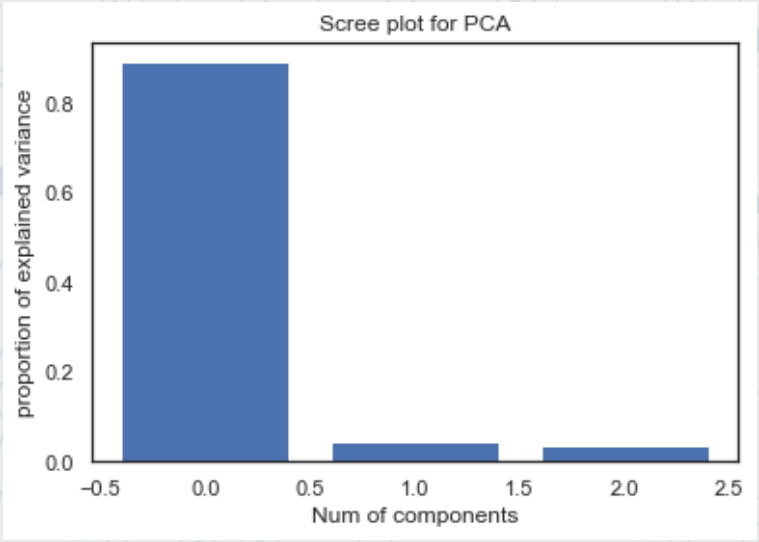


Classification

Correlation



Principle Component Analysis (PCA)



Classification

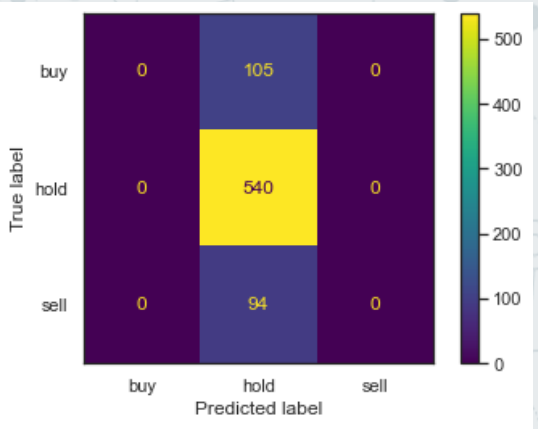
Gridsearch with classification aglorithms

```
Training accuracy score: 0.7339201083276913
Test accuracy score: 0.7307171853856563
Training F1 score: 0.7339201083276912
Test F1 score: 0.7307171853856562
              precision    recall  f1-score   support

         buy       0.00      0.00      0.00       105
        hold       0.73      1.00      0.84       540
        sell       0.00      0.00      0.00        94

    accuracy                           0.73       739
   macro avg       0.24      0.33      0.28       739
weighted avg       0.53      0.73      0.62       739
```

SVM

```
Training accuracy score: 0.6844660194174758
Test accuracy score: 0.6740463215258855
Training F1 score: 0.6844660194174758
Test F1 score: 0.6740463215258855
              precision    recall  f1-score   support

         buy       0.00      0.00      0.00       462
        hold       0.67      1.00      0.81      1979
        sell       0.00      0.00      0.00       495

    accuracy                           0.67      2936
   macro avg       0.22      0.33      0.27      2936
weighted avg       0.45      0.67      0.54      2936
```

random forest

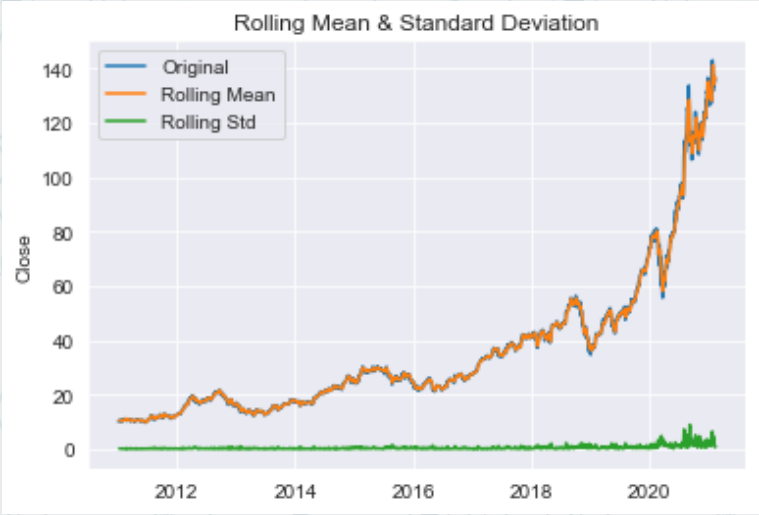



```
Training accuracy score: 0.6844660194174758
Test accuracy score: 0.6740463215258855
Training F1 score: 0.6844660194174758
Test F1 score: 0.6740463215258855
              precision    recall  f1-score   support

         buy       0.00      0.00      0.00       462
        hold       0.67      1.00      0.81      1979
        sell       0.00      0.00      0.00       495

    accuracy                           0.67      2936
   macro avg       0.22      0.33      0.27      2936
weighted avg       0.45      0.67      0.54      2936
```

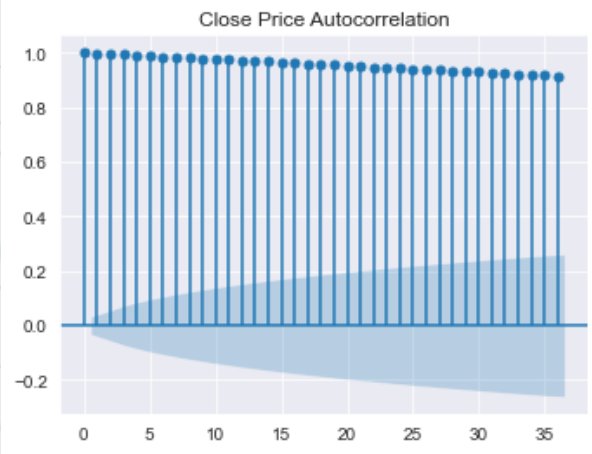
Xgboost

Time Series

Dickey-Fuller Test



ACF & PACF



Time Series

SARIMAX Model with exogenous features


```
SARIMAX RMSE of close price: 1.15
SARIMAX MAE of close price: 0.82
```

Facebook Prophet


```
Facebook Prophet close price train RMSE: 4.34
Facebook Prophet close price test RMSE: 33.89
Facebook Prophet close price train MAE: 2.59
Facebook Prophet close price test MAE: 33.05
```

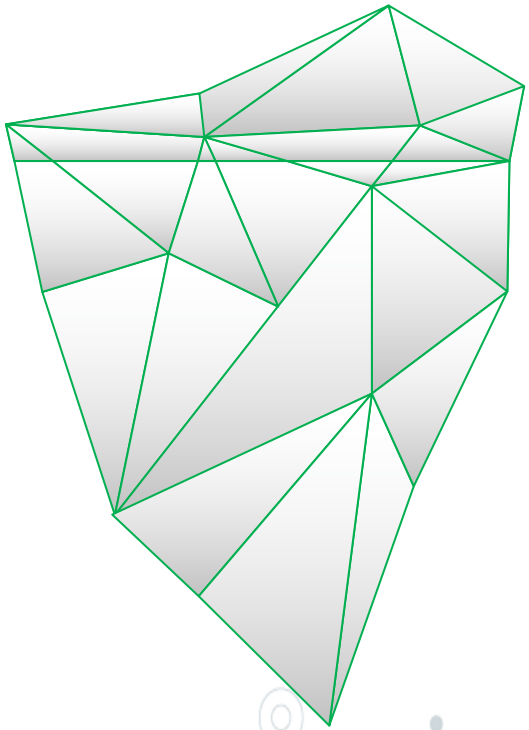
Time Series

LSTM


```
LSTM Networks close price train RMSE: 0.70
LSTM Networks close price test RMSE: 3.28
LSTM Networks close price train MAE: 0.37
LSTM Networks close price test MAE: 2.61
```

Frontend

Streamlit was used to create a frontend to predict stock price with different time range.



```
st.title('Stock Market Prediction')


@st.cache
def data_load(stock):
    '''
    Loads the stock we want to predict
    '''
    tick = yf.Ticker(stock)
    # get historical market data
    data = tick.history(period="max")
    data = data.reindex(pd.date_range(df.index[0], df.index[-1], freq='D')).interpolate()

    return data


def create_dataset(dataset, look_back=1):
    dataX, dataY = [], []
    for i in range(len(dataset)-look_back-1):
        a = dataset[i:(i+look_back), 0]
        dataX.append(a)
        dataY.append(dataset[i + look_back, 0])
    return np.array(dataX), np.array(dataY)
```

Next Step

-To access the updated quarterly reports timely and obtain more important features.

-To tune the hyperparameters (exogenous variables) in Time Series models. Technical indicators such as MACD, Stochastic, RSI, etc can be used.

-Besides Twitter, gathering more relevant sentimental data from other web sources.