Build a Stock Price Predictor

April 18, 2018

1 Capstone Proposal

Dang Le Dang Khoa
April 17th, 2018

2 Domain Background

- Investment firms, hedge funds, and even individuals have been using financial models to better understand market behavior and make profitable investments and trades. A wealth of information is available in the form of historical stock prices and company performance data, suitable for machine learning algorithms to process.
- For this project, the task is to build a stock price predictor that takes daily trading data over a certain date range as input, and outputs projected estimates for given query dates. Note that the inputs will contain multiple metrics, such as opening price (Open), highest price the stock traded at (High), how many stocks were traded (Volume) and closing price adjusted for stock splits and dividends (Adjusted Close); the system only needs to predict the Adjusted Close price.

3 Problem Statement

- Predict(forecast) the Adjusted Close day by day
- Suggest 2 different approaches:
  - Traditional supervised machine learning methods(Regressions)
  - A specific machine learning method for time series forecasting(ARIMA model)
- Challenges of Time series data:
  - Contains a lot of noises
  - Is different from tabular data that each data point related to each other
  - Contains time-dependent structures:
    - Level: the average value in the series
    - Trend: global increasing or decreasing
    - Seasonalities: repeating pattern of the series

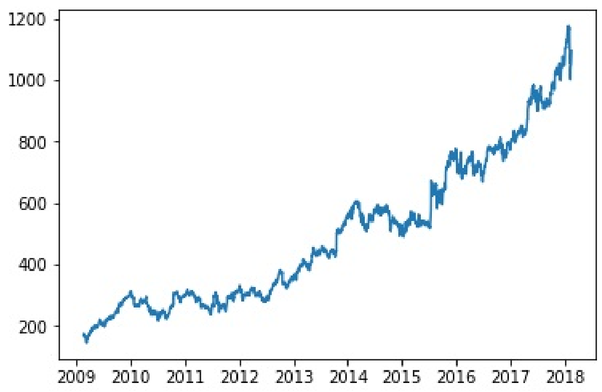


image alt <>

4 Datasets and Inputs

- Dataset downloaded from: http://finance.yahoo.com
- Example: GOOG dataset

Dates	Open	High	Low	Close	Adj Close	Volume
2009-02-17	172.135422	172.423553	168.747467	170.222870	170.222870	11434600
2009-02-18	172.498062	175.548233	169.159775	175.414093	175.414093	12127300
2009-02-19	177.580017	178.737488	169.601898	170.212936	170.212936	10042200
2009-02-20	167.932755	173.332642	166.417618	172.105621	172.105621	12515000
2009-02-21	172.378845	173.769791	163.710220	163.963577	163.963577	10510000

- Inputs: Date and Adjusted Close
- Example: Time series input

Dates	Adj Close
2009-02-17	170.222870
2009-02-18	175.414093
2009-02-19	170.212936
2009-02-20	172.105621
2009-02-21	163.963577

- Output: Prediction Price at the current day
- Optional output: Suggest Buy/Sell/Hold

5 Solution Statement

5.1 Regression Approach

- Choose Linear Regression for traditional machine learning model approach

- For time series Regression, create lagged values as new features

$$lag(n) = f(t-n)$$

Example: lagged values with n = 7

Dates	t	t-1	t-2	t-3	t-4	t-5	t-6	t-7
2011-01-08	1.020005	1.020005	1.015140	1.007810	0.996310	1.000000	1.00000	1.00000
2011-01-09	1.020005	1.020005	1.020005	1.015140	1.007810	0.996310	1.00000	1.00000
2011-01-10	1.016315	1.020005	1.020005	1.020005	1.015140	1.007810	0.99631	1.00000
2011-01-11	1.019293	1.016315	1.020005	1.020005	1.020005	1.015140	1.00781	0.99631
2011-01-12	1.020716	1.019293	1.016315	1.020005	1.020005	1.020005	1.01514	1.00781

- Perform grid search to find the optimal lag order based on Root Mean Squared Error(RMSE)

5.2 ARIMA Approach

5.2.1 Definition

- ARIMA stands for Autoregressive Integrated Moving Average (Alternative name: Box-Jenkins Model)
- ARIMA is a forecasting technique that projects the future values of a series based on its own inertia
- Its main application is short-term forecasting requiring at least 40 historical data points
- ARIMA works best when data:
  - Exhibits a stable or consistent pattern
  - Have a minimum amount of outliers

5.2.2 Models parameter

- ARIMA attempts to describe the movement in a stationary time series as a function of "autoregressive and moving avg"
  - AR(autoregressive)
  - MA(moving avg)
- Autoregressive Models:

$$X(t) = A(1) * X(t-1) + A(2) * X(t-2) + ... + A(n) * X(t-n) + E(t)$$

  - X(t): the time-series
  - X(t-n): time series lagged n
  - A(n): autoregressive parameters
  - E(t): the error term of the model
- Moving Average Models:

$$X(t) = -B(1) * E(t-1) + E(t)$$

  - B(1): MA of order 1
  - E(t): current error term
  - E(t-1): error in the previous period

5.2.3 Approach

- Mixed ARIMA model is built on 3 parameters (p,d,q)
  - p: lag order
  - d: degree of differencing
  - q: size of moving average window(order of moving average)
- Perform grid search to find the optimal orders (p,d,q) based on Root mean squared error(RMSE)

6 Evaluation Metrics

- RMSE(Root Mean Squared Error)

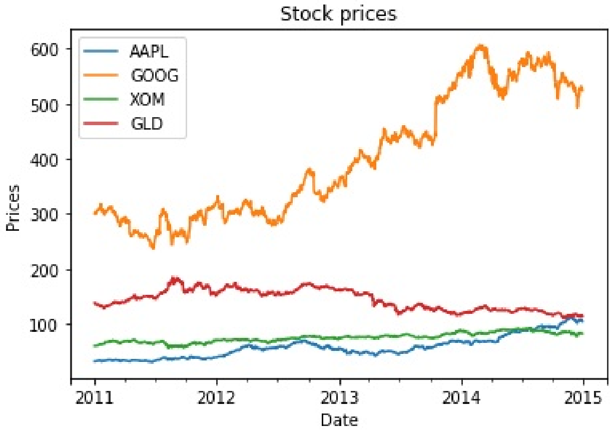
$$RMSE = \sqrt{\frac{\sum_{t \in N}(y_t - \hat{y}_t)^2}{N}}$$

- Reasons to choose RMSE:
  - Squaring error to have positive values
  - Putting more weight on large errors
- Cons of choosing RMSE:
  - Our data may have many outliers that affect the perfomance evaluation

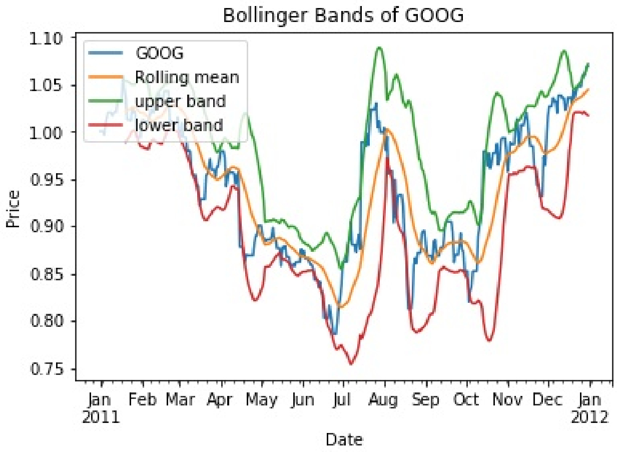
7 Project Design

7.1 Data Exploratory

- Perform statical analysis
  - Mean
  - Standard Deviation
  - Median
  - Sum
- Visualize time series data
  - Visualize time-series data



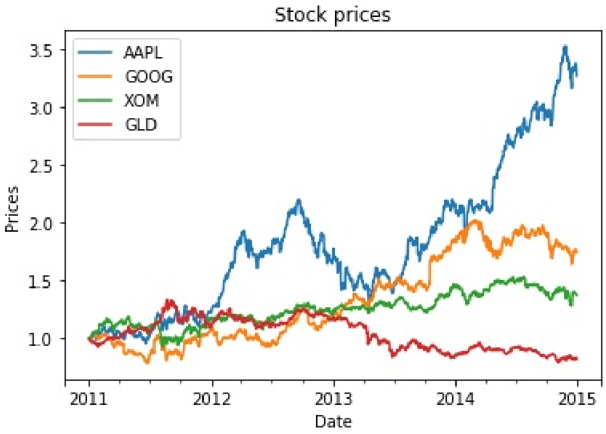
- Bollinger Bands - Rolling stats



7.2 Data Preprocessing

- Normalize data

$$f(t) = \frac{f(t)}{f(0)}$$

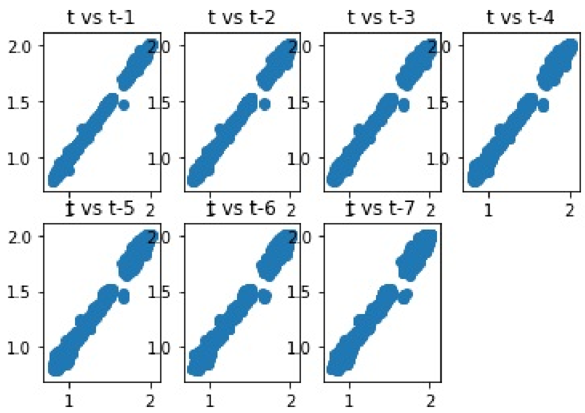


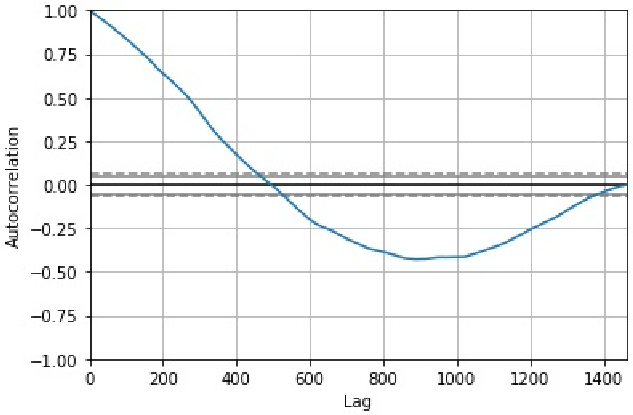
- Remove Trend if nesscessary

7.3 Model Prediction

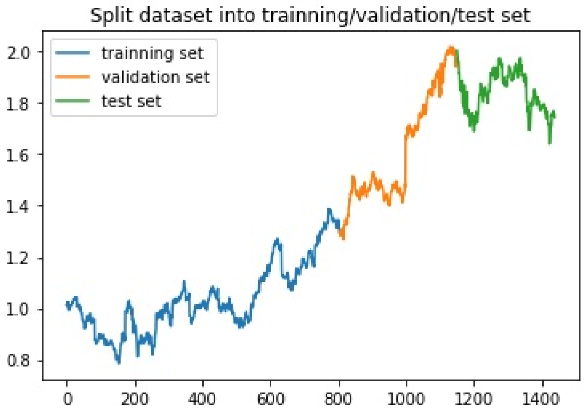
7.3.1 Linear Regression

- Feature engineering
  - Create lagged value
  - Examine correlation between lagged datapoints





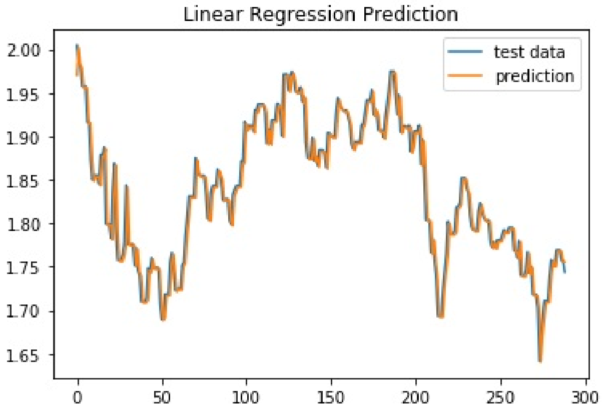
- Split data into trainning/validation/test set



- Perform grid search to find optimal parameters

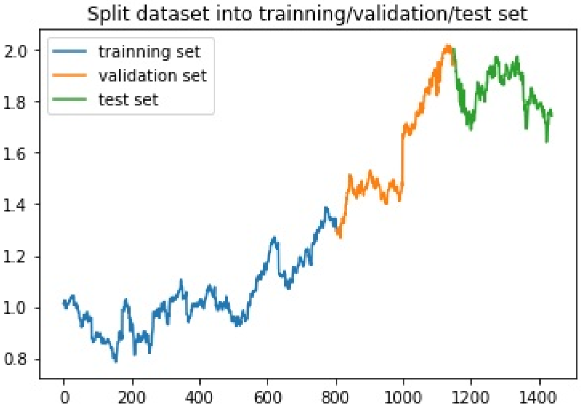
```
for lag in lag_values:
    model = fit(trainning_set(X, y, lag))
    y_hat = model.predict(validation_set(X, lag))
    error = RMSE(y - y_hat)
    best_params = params with minimum error
return lag
```

- Evaluate based on RMSE and visualization



7.3.2 ARIMA

- Split data into trainning/validation/test set



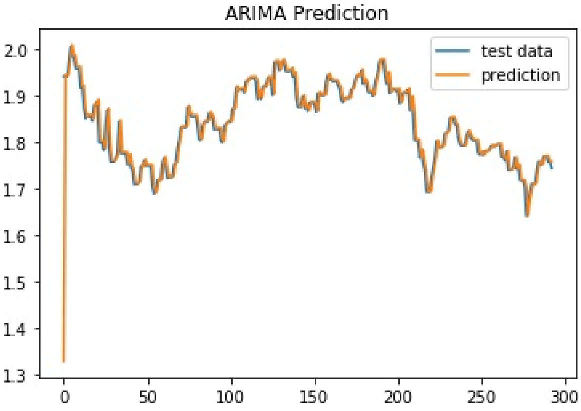
- Perform grid search to find optimal parameters

```
for each (p,q,d) in order_values:
    model = fit(trainning_set(X, y, p, q, d))
    y_hat = model.predict(validation_set(X, p, q, d))
    error = RMSE(y - y_hat)
    best_params = params with minimum error
return best_params(p,q,d)
```

- Evaluate based on RMSE and visualization

7.4 Model Evaluation

- Compare RMSE of 2 different approaches
- Explain the results based on visualizations



8 Reference

https://www.quora.com/What-is-ARIMA