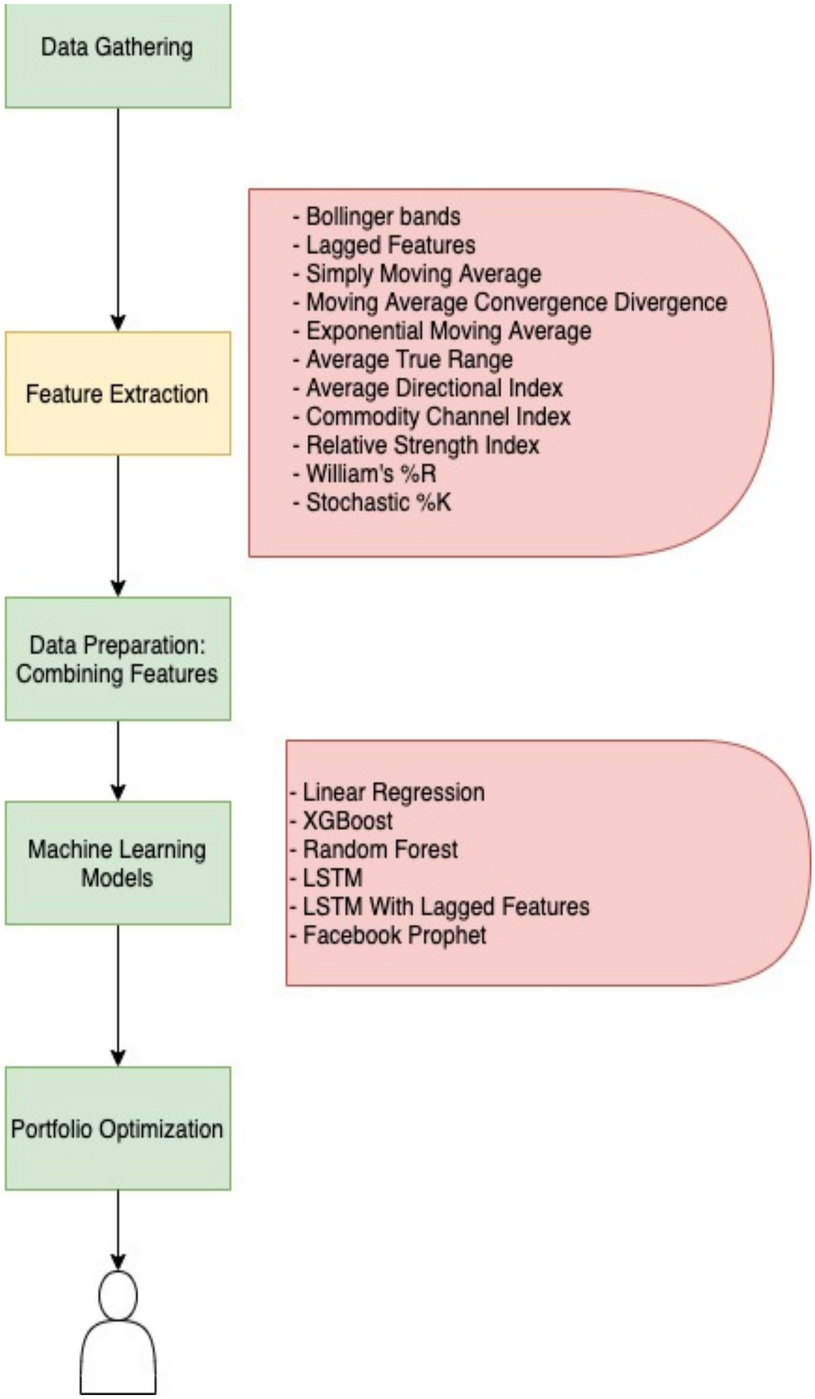
This is a very interesting project where we will be doing extensive feature extraction and model deployment pipeline for Stocks Data. At the end we will be deploying these models into Heroku using streamline.
App link: https://stock-prediction-dashboard.herokuapp.com/
Blogpost URL: https://medium.com/@nikhilkohli1992/a97afc55d926



Data Gathering

Feature Extraction

- Bollinger bands
- Lagged Features
- Simply Moving Average
- Moving Average Convergence Divergence
- Exponential Moving Average
- Average True Range
- Average Directional Index
- Commodity Channel Index
- Relative Strength Index
- William's %R
- Stochastic %K

Data Preparation: Combining Features

Machine Learning Models

- Linear Regression
- XGBoost
- Random Forest
- LSTM
- LSTM With Lagged Features
- Facebook Prophet

Portfolio Optimization

ABSTRACT

The ubiquity of data today enables investors at any scale to make better investment decisions. The challenge is ingesting and interpreting the data to determine which data is useful, finding the signal in this sea of information.

Stock market prediction is the act of trying to determine the future value of a company stock or other financial instrument traded on an exchange. The successful prediction of a stock's future price could yield significant profit. The efficient-market hypothesis suggests that stock prices reflect all currently available information and any price changes that are not based on newly revealed information thus are inherently unpredictable. Others disagree and those with this viewpoint possess myriad methods and technologies which purportedly allow them to gain future price information.

We fall in the second categories and in this project, we will be using the power of Machine Learning and Artificial Intelligence to predict the stock prices.

Forecasting Future Stock prices is a very hard problem to solve. An efficient Predictive model to correctly forecast future trend is crucial for Hedge funds and algorithmic trading. Especially in the case of Algorithmic Trading where error should me minimal as millions of dollars are at stake for each trade.

Portfolio Optimization strategies needs to be back tested on historical data after predicting future stock prices.

Stock prices depends upon many factors like the Market behaviors, other stocks, Index funds, Global news etc. We will try to capture many of these in our features.

In this project, we will look at this problem in many ways to Predict the Closing Prices -

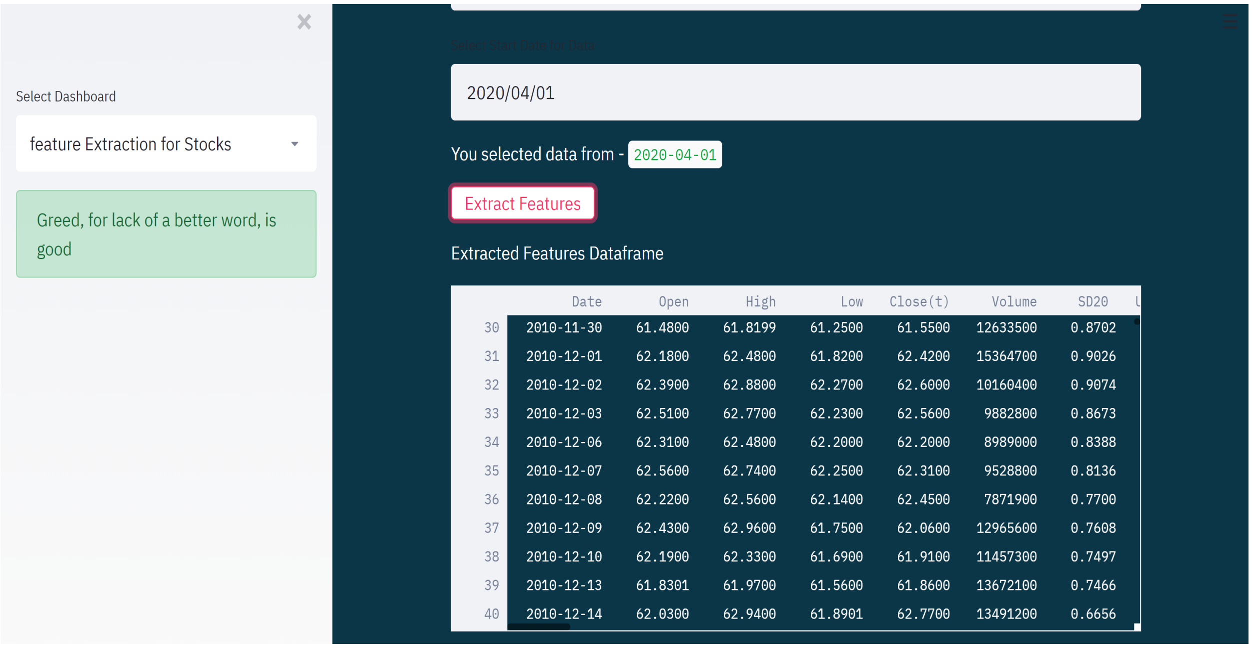
- We will start with Extracting Features and see which performs well for predicting each stock. We will extract various Technical Indicators described below.
- Then check correlation and Perform feature selection using RFECV Recursive feature Elimination using Random Forest to select best features.
- Then we will create a pipeline for this feature extraction and convert the entire code into Pipeline so anyone can easily run it and get the extracted features data for each stock.
- Next we will use Time Lagged data as a feature and create features based on previous day closing prices, Previous days Index funds prices.
- Then we will train 4 different Algorithms - Linear Regression, Random Forest, XG Boost, LSTM and GRU for forecasting next day price and test and evaluate it on historical stock data.

- We will also create a **Pipeline** for this to train many stocks with many algorithms in just one go.
- We will Evaluate the data on around 2 years of data which is a long period, so if our models are closer overall, means we are doing great. Metrics we will use are MAE, MAPE, R2 and RMSE. Final Metrics which we will look at to compare models is MAE (Mean Absolute Error)
- We will also check feature importance of various features using Random forest and XG boost in this.
- We will pick the best algorithm from these and will tune the Number of lagged days to consider for forecasting for each type like Stock price, other index Funds previous prices.
- For LSTM, we will use Lagged previous days prices for a lookback period of 30-60 days.
- Then we will create a Portfolio of these stocks and will build a strategy using Sharpe ratio to optimize the portfolio and allocate the money of a fund effectively.
- As a future scope, we will also try to create a dashboard to Show the comparison of 2 portfolios before and after optimization.

For a user who is running this, please follow the folder structure

1. **Data Source**
2. **Feature Extraction**
3. **Feature Extraction Pipeline**
4. **Stock Prediction**
5. **Stock Prediction Pipeline**
6. **Portfolio Optimization**

End Result: A Heroku app



Greed, for lack of a better word, is good

Select Start Date for Data

2020/04/01

You selected data from - 2020-04-01

Extracted Features Dataframe

	Date	Open	High	Low	Close(t)	Volume	SD20
30	2010-11-30	61.4800	61.8199	61.2500	61.5500	12633500	0.8702
31	2010-12-01	62.1800	62.4800	61.8200	62.4200	15364700	0.9026
32	2010-12-02	62.3900	62.8800	62.2700	62.6000	10160400	0.9074
33	2010-12-03	62.5100	62.7700	62.2300	62.5600	9882800	0.8673
34	2010-12-06	62.3100	62.4800	62.2000	62.2000	8989000	0.8388
35	2010-12-07	62.5600	62.7400	62.2500	62.3100	9528800	0.8136
36	2010-12-08	62.2200	62.5600	62.1400	62.4500	7871900	0.7700
37	2010-12-09	62.4300	62.9600	61.7500	62.0600	12965600	0.7608
38	2010-12-10	62.1900	62.3300	61.6900	61.9100	11457300	0.7497
39	2010-12-13	61.8301	61.9700	61.5600	61.8600	13672100	0.7466
40	2010-12-14	62.0300	62.9400	61.8901	62.7700	13491200	0.6656