ORIE 5255 Presentation

- Explore the portfolio strategy with ML technique in Pharmaceuticals industry

Team member: Xuanyu Ding, Bangjie Xu, Shaohan Wang

Problem Setup

Company

iShares U.S. Pharmaceuticals ETF (IHE):

- JOHNSON & JOHNSON
- PFIZER INC
- MERCK & CO INC
- ZOETIS INC CLASS A
- ELI LILLY
- CATALENT INC
- VIATRIS INC
- BRISTOL MYERS SQUIBB
- JAZZ PHARMACEUTICALS
- PERRIGO PLC

Background

Under the pandemic, pharmaceuticals industry companies played a key role in developing the vaccine and providing medical care program to the society. Our team hope to take a step into exploring their stock performance and related portfolio construction

Problem statement

Our team hope to explore the usage of ML into portfolio construction, specifically to capture the noisy trading signal in daily basis. To better interpret the contribution of the ML technique into constructing the portfolio, we check the portfolio performance with the plot

https://www.ishares.com/us/products/239519/ishares-us-pharmaceuticals-etf

Challenges and Solutions

Challenge 1

Challenge 2

Challenge 3

What to predict

Inspired by Prof. Aldridge's book "Big Data Science in Finance", our team try to set the stock one-day ahead return as the target to determine the trading signal

What factor to use

The chosen input variables are 10-day stock price average, 10-day price maximum minus minimum, 10-day price volatility and the volume of transactions

What model to fit

We set linear regression as the baseline model. We also fit the Random Forest tree model and Neural Network model to compare the portfolio performance

Linear Model

- Set baseline

- Select training and testing window
- Train: 2018.1.1 - 2020.1.1
- Test: 2020.1.2 --- 2021.9.1

- Evaluate factor
- corr

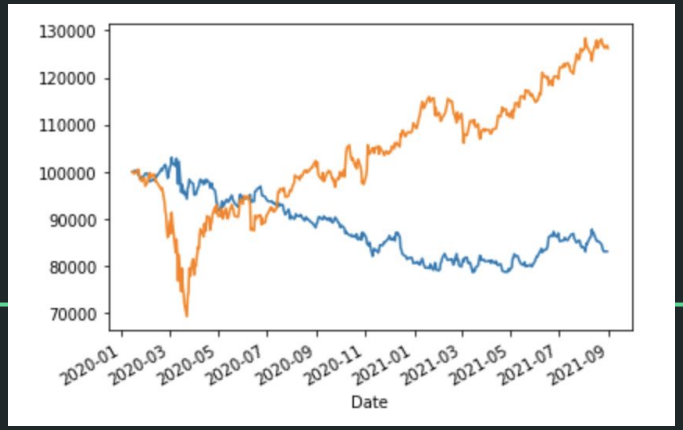
- Regularization

Date	ma	max_min	volm	volt	one_day_ahead_return	prediction	signal
2020-01-15	61.241993	3.598354	10937400	1.386469	0.189384	-0.177125	-1
2020-01-16	61.567738	3.787739	8435800	1.438397	-0.056824	-0.220536	-1
2020-01-17	61.940829	3.598351	14375000	1.305340	0.672325	-0.119805	-1
2020-01-21	62.362215	3.380547	12519400	1.150245	0.000000	-0.094087	-1
2020-01-22	62.693641	3.380547	15186700	1.038061	-0.530277	-0.038146	-1
...	...	...	...	...	...	...	...
2021-09-23	60.926509	2.955910	9393700	1.049361	-0.396770	-0.093388	-1
2021-09-24	60.638853	2.668255	9541700	0.919926	-0.386845	-0.052734	-1
2021-09-27	60.341278	2.251652	8231800	0.782366	0.000000	-0.021607	-1
2021-09-28	60.116113	2.142540	10812500	0.696624	0.773693	0.028784	1
2021-09-29	59.979228	1.458118	9043300	0.502045	-0.959999	0.077133	1

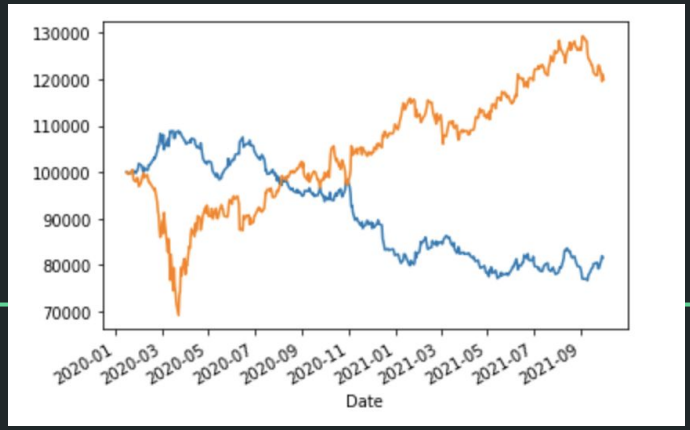
	ma	max_min	volm	volt	one_day_ahead_return
ma	1.000000	-0.262183	-0.364318	-0.285978	-0.093213
max_min	-0.262183	1.000000	0.142822	0.972253	-0.161822
volm	-0.364318	0.142822	1.000000	0.123472	0.009950
volt	-0.285978	0.972253	0.123472	1.000000	-0.173259
one_day_ahead_return	-0.093213	-0.161822	0.009950	-0.173259	1.000000

LR Performance (1)

- Train once



- Rolling window = 20



LR Performance (2)

- Training start from 2018
- 2019
- 2020

Random Forest

- Set baseline
- Select training and testing window
- Hyperparameter tunning
- Run model and backtest

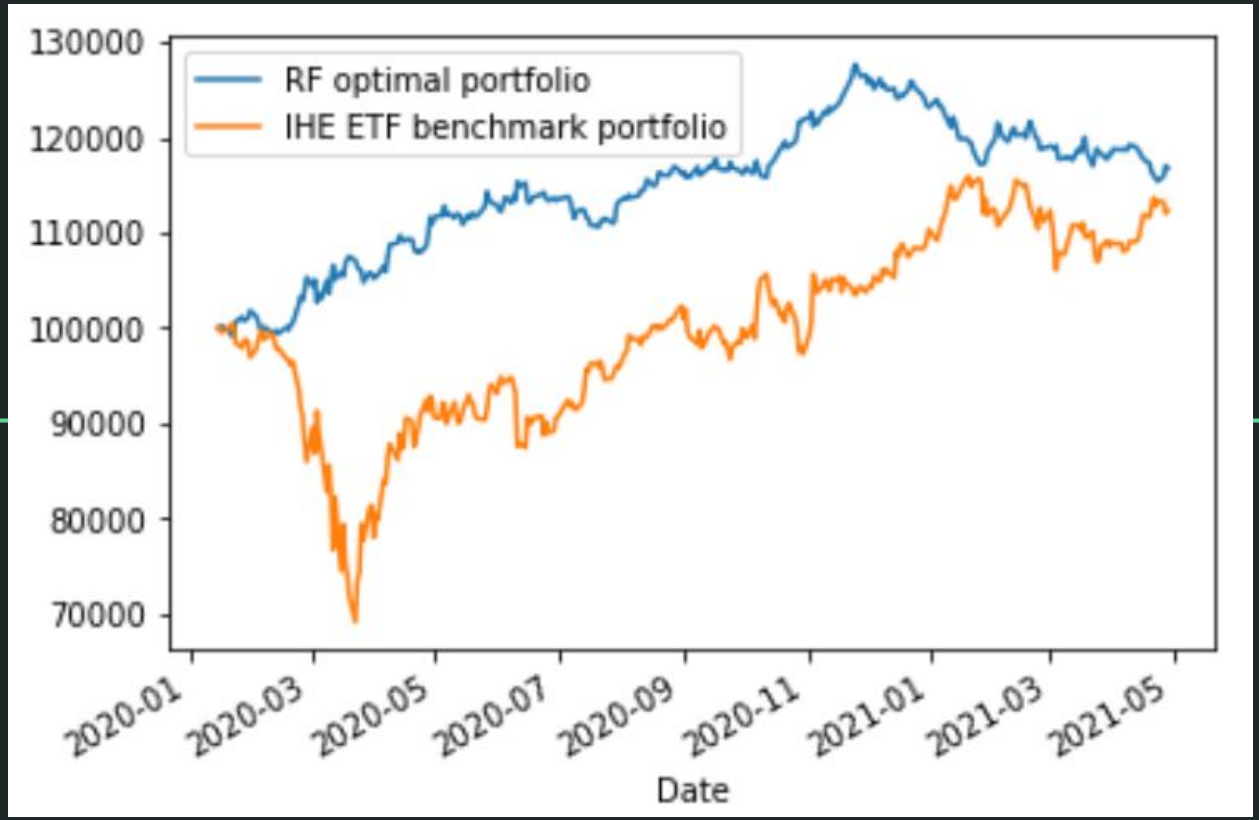
Date	ma	max_min	volm	volt	one_day_ahead_return
2020-01-15	57.680000	4.860001	875100	1.837396	0.630001
2020-01-16	58.167001	5.490002	652000	2.139839	-0.590000
2020-01-17	58.657001	5.480000	774500	2.144414	0.639999
2020-01-21	59.210001	5.230000	387300	2.092643	0.230000
2020-01-22	59.756001	5.070000	361100	1.953812	-0.180000

- Data Preprocessing
  - Drop nan values
  - Add and compute x-variables and one-day ahead returns into every stock's dataframe
  - Training set: 2018.1.1 --- 2020.1.1
  - Testing set: 2020.1.2 --- 2021.4.30

```
Root Mean Squared Error of BMY is 0.9983908559098249
Root Mean Squared Error of CTLT is 2.159625895077728
Root Mean Squared Error of ELAN is 0.8120800378743563
Root Mean Squared Error of JAZZ is 3.7469883621298132
Root Mean Squared Error of JNJ is 2.6908073080258954
Root Mean Squared Error of LLY is 3.85229614516501
Root Mean Squared Error of MRK is 1.307759784882788
Root Mean Squared Error of PFE is 0.669089001894527
Root Mean Squared Error of VTRS is 0.4699786366255631
Root Mean Squared Error of ZTS is 3.052106599818873
```

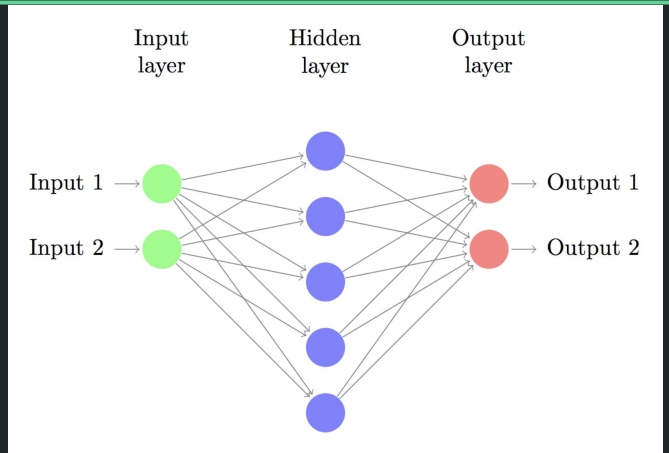
- Hyperparameter tunning
  - Grid search to find best parameters
    - ➔ max_depth
    - ➔ N_estimators
    - ➔ min_samples_split

Random forest optimal portfolio performance



Neural Network (NN)

- Neural Network as one of the most popular ML/DL techniques in the industry can capture the non-linear relationships between input and target variables.
- By adding different user-specified layers to account for the internal interactions and transformations. To train NN model, one common method is to apply the gradient descent approach to optimize the hyper-parameters in each layer.
- Our application here is to predict target variable (one-day ahead return) with the specified input variables in rolling basis with the hope to capture the trading signal with future movement influence and also to avoid overfitting problems under this framework.
- The selected test period for NN model is from 01/01/2020 to today.



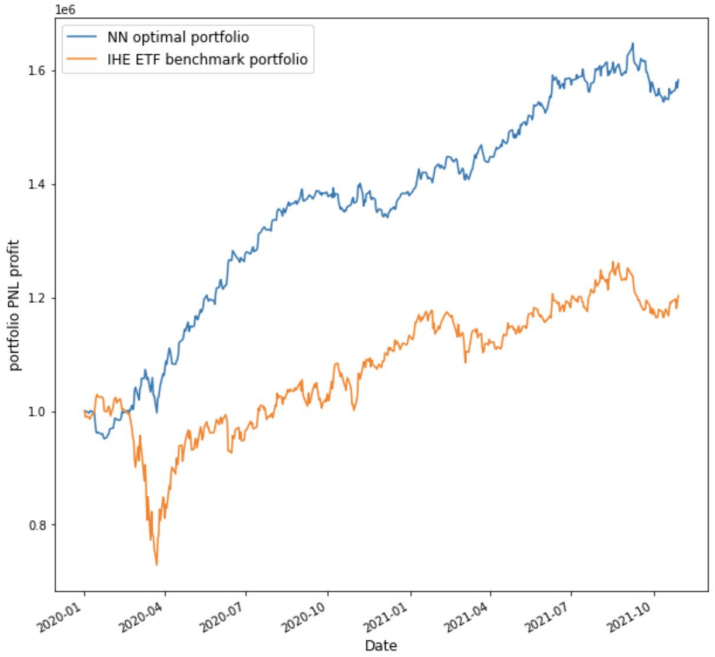
NN Layer Selection

- One of the most popular layer for predicting stock returns is the Hyperbolic Tangent activation function because it can capture the value between -1 and 1.
- We may also add more hidden layers in between to account for most sophisticated variable interactions.
- Here, I choose to perform a 6-layer neural network model with the first hidden layer and output layer as Linear function; second and fourth hidden layer as Hyperbolic Tangent function; Third hidden layer as Rectifier Linear Unit function.

Rolling window selection

- The choice of the rolling window size is extremely important because the NN model will give us different prediction based on different window length. Common window length is between 5 and 200.
- Here, for each selected ticker, I determine the optimal rolling window size separately given the largest backtest single ticker PNL profit.

NN result (PNL & rolling window size)



	optimal rolling window size
BMY	120
CTLT	200
JAZZ	20
JNJ	140
LLY	140
MRK	60
PFE	80
PRGO	100
VTRS	120
ZTS	200

	return given optimal rolling window
BMY	0.331058
CTLT	1.489754
JAZZ	0.464530
JNJ	0.195313
LLY	0.560097
MRK	-0.015935
PFE	0.349664
PRGO	1.142031
VTRS	0.920168
ZTS	0.391682

```
The return of the entire portfolio under optimal window sizes for each ticker is 0.5828361620679109
```

Conclusion and Future Works

- NN >> RF >> LR

- Future works
- Explanation of performance
- Factor selection
- Model selection
- Practical usage

Thank you!