# Assignment 4

I use the data of HSBC stock and year1 is 2018 year2 is 2019. All data in this file is rounded to two decimal points.

## Week labeling

My strategy to label is compare the open price of the first day with the close price of the last day. If open bigger than close label ‘red’, else label ‘green’. But there is a special case, when label is green if the close price of this Friday is smaller than the close price of last Friday, the label will turn to red. Also, if the close price of this Friday is bigger than the close price of last Friday, the red label will turn to green label. (at here Friday means the last day of the week not exactly Friday)

The calculation process of the following results is displayed in the file(stock\_data\_label.py)

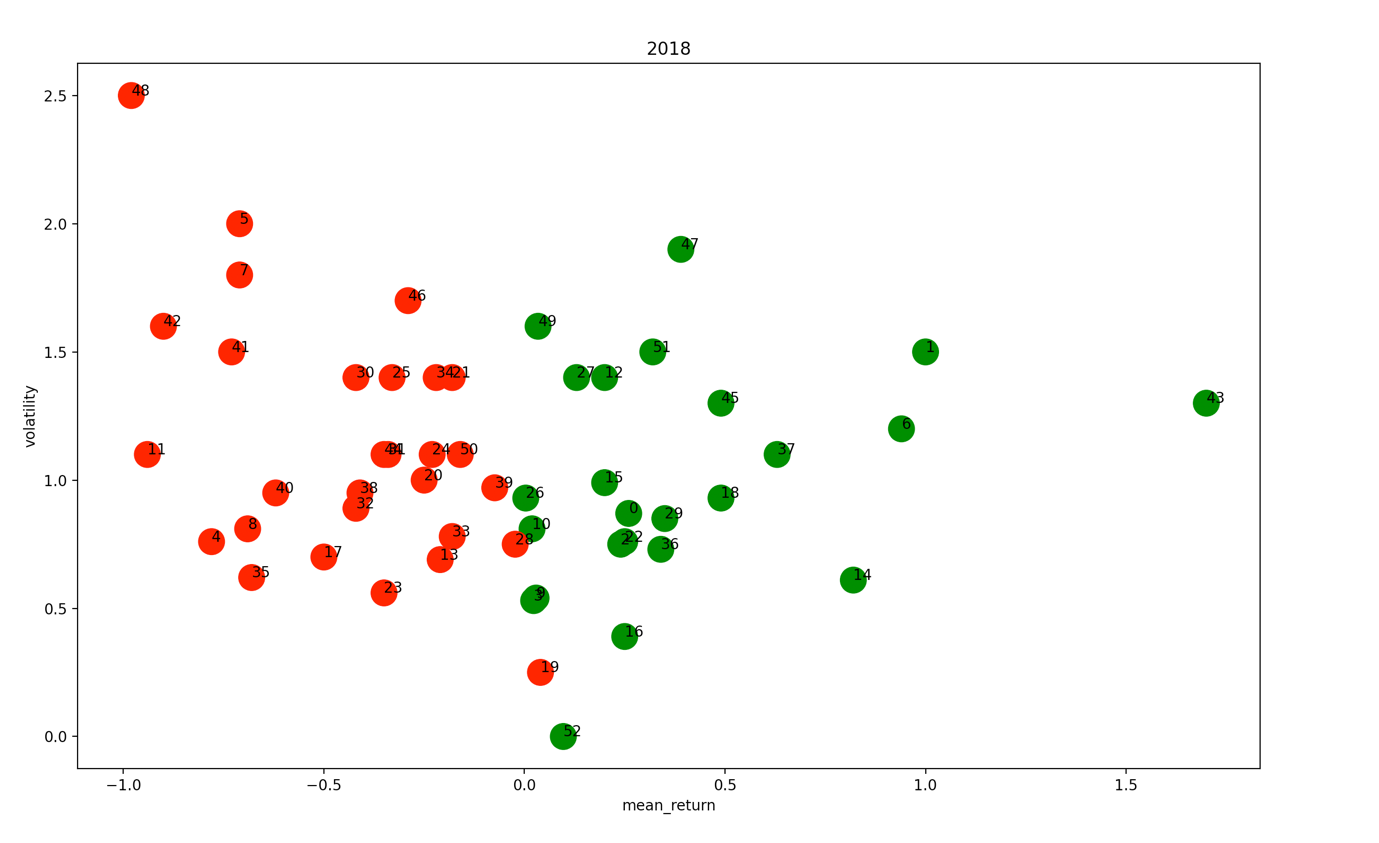
* Question1

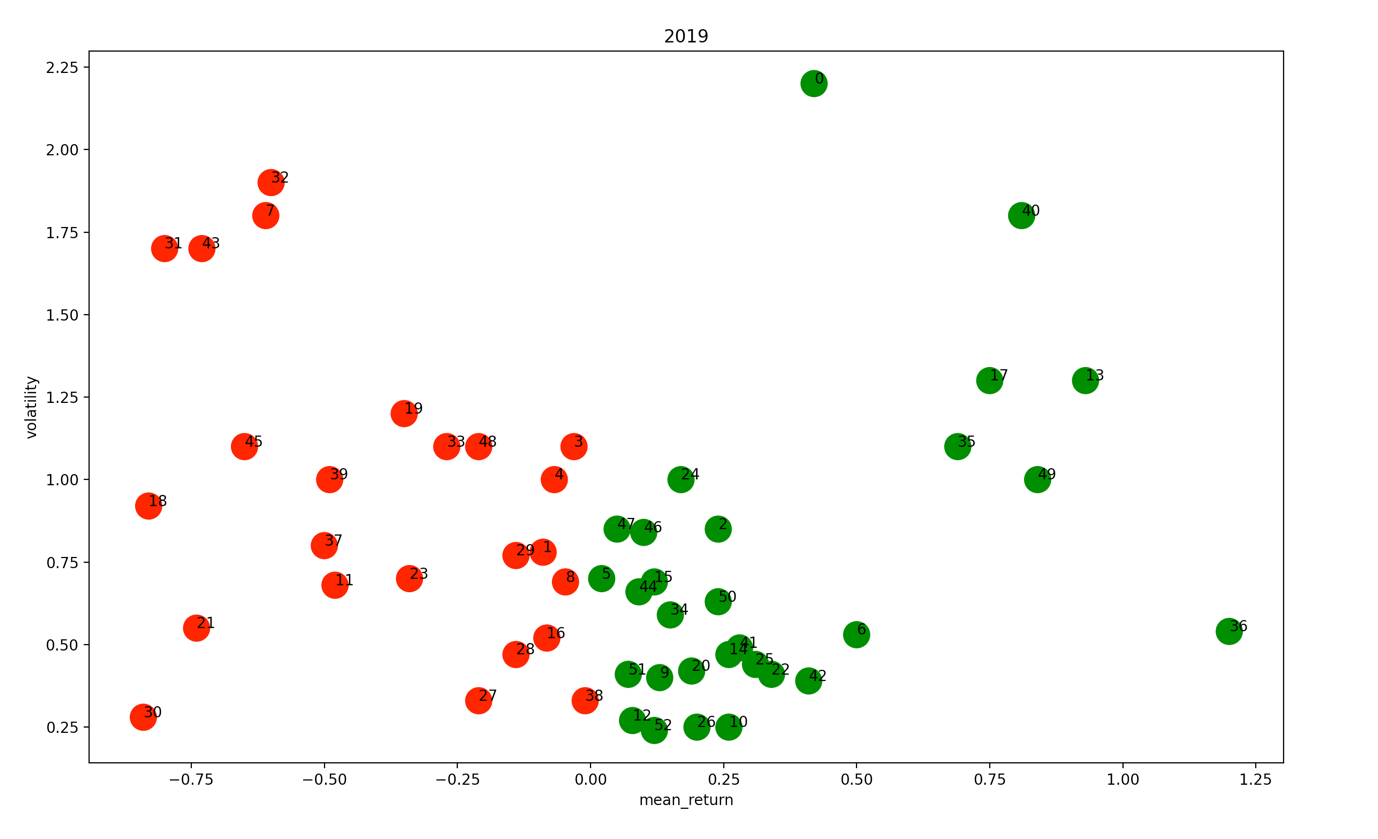
The final price is 96.83

* Question2

The final price is 157.38. My strategy’s result in a larger amount at the end of the year.

## Examine labels

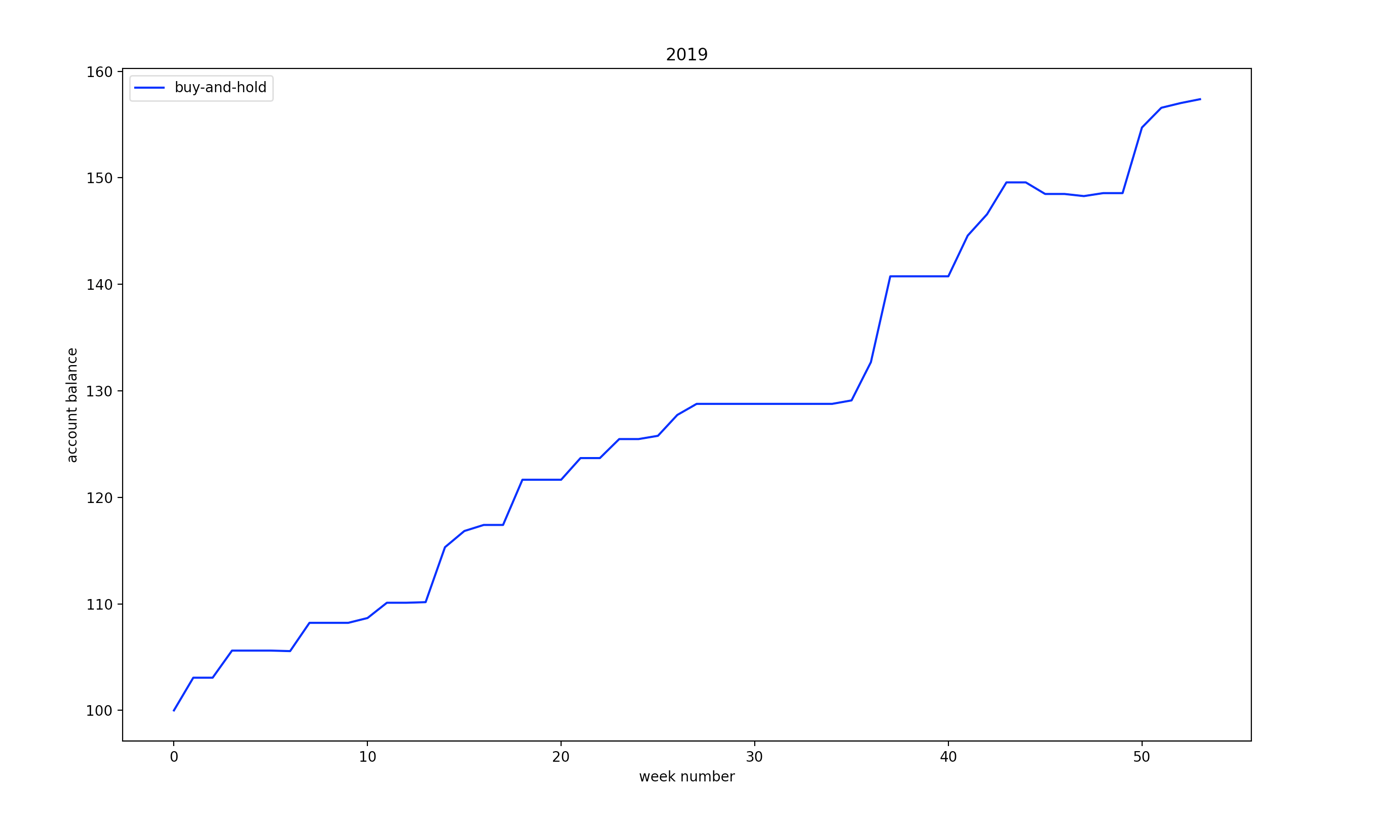
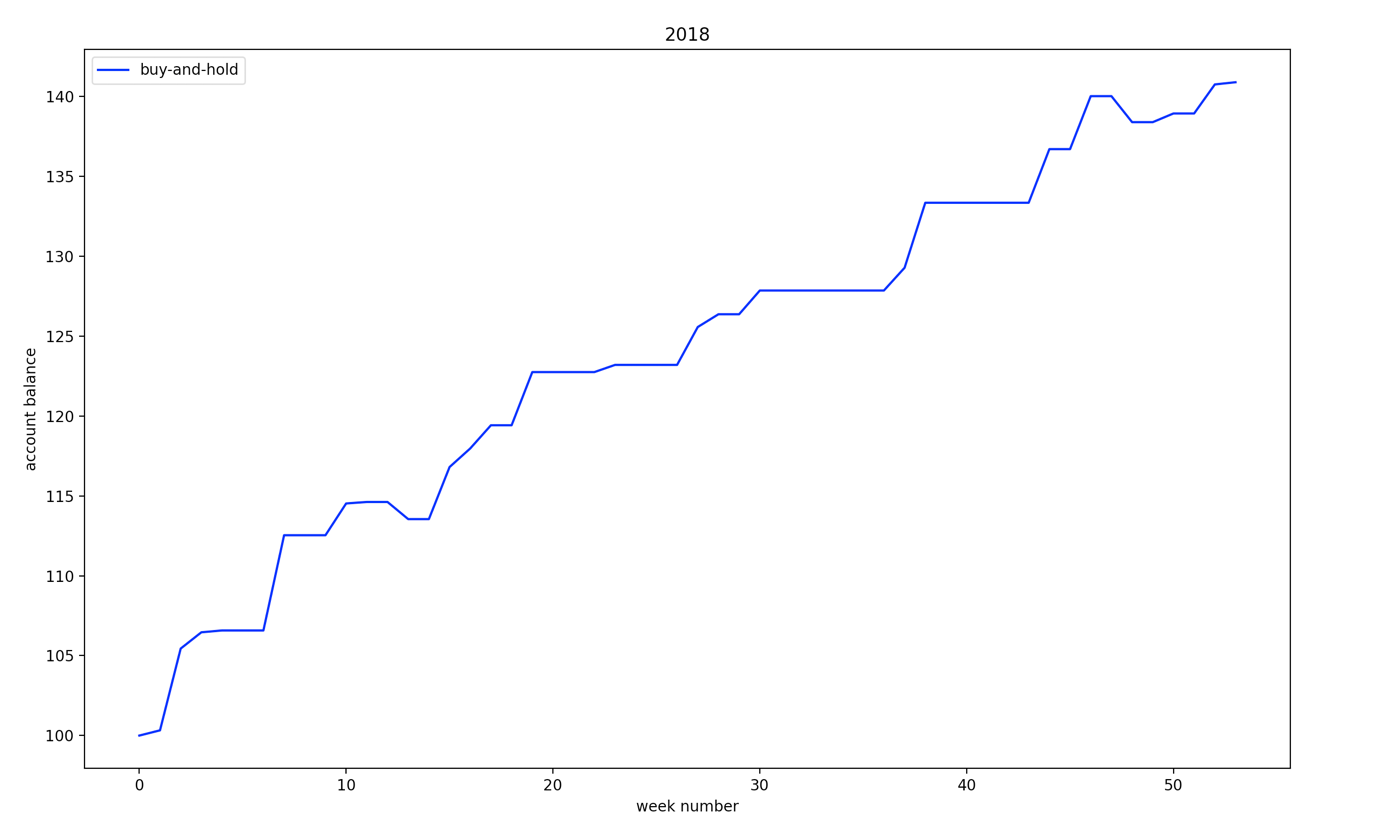




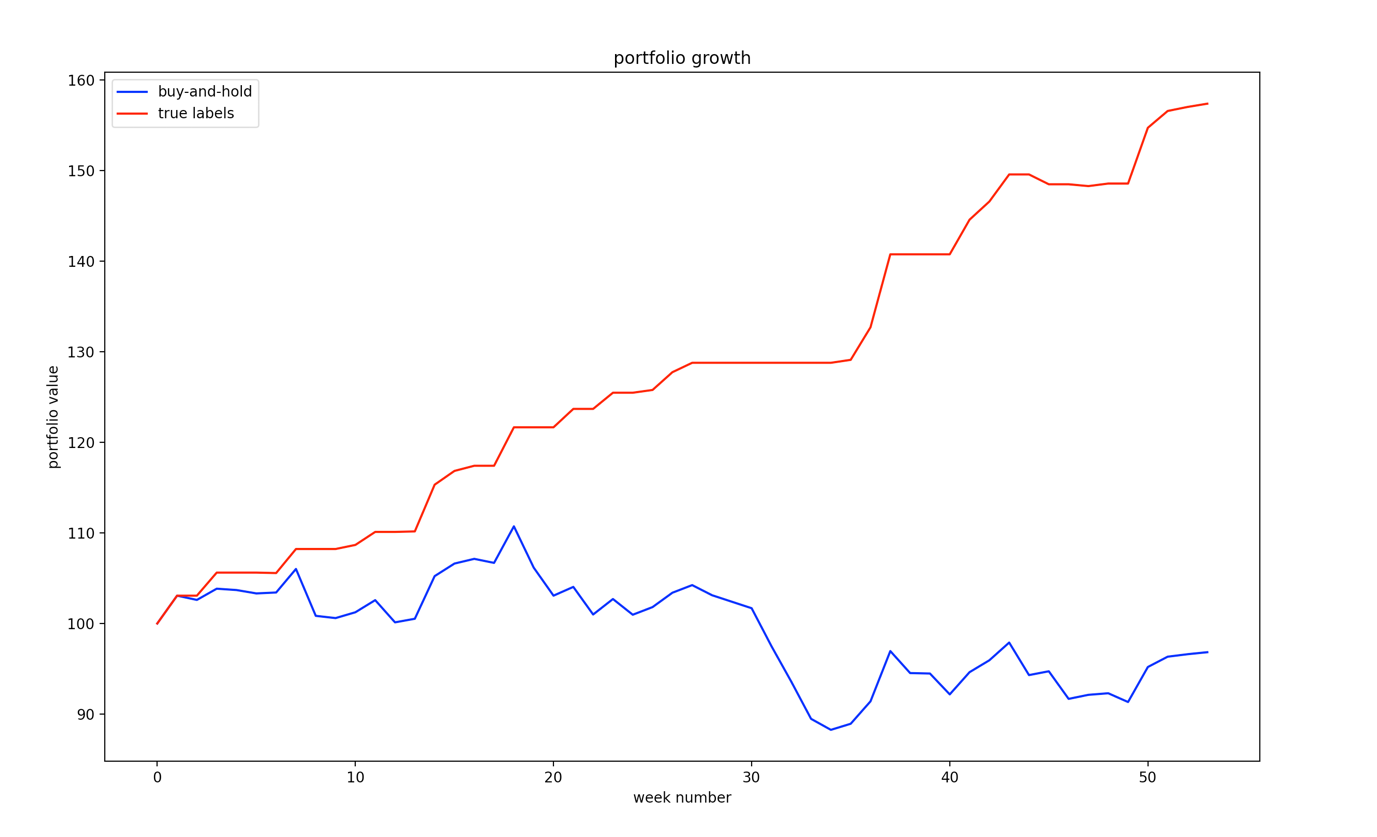
The values in both charts are after removing the percent sign

1. For both 2018 and 2019, most green points have a mean value greater than 0. Because the x-axis left to right is negative to positive, so most red points are on the left of the green points. And the y-axis is the standard deviation, so it has nothing to do with the color of this point. The higher the position of the point, the farther the data distribution for this week is from the average. As can be seen from the above two charts, most of the standard deviations in 2018 are between 0.5-1.5, while most of the standard deviations in 2019 are below 1.25, and most are clustered below 1. It can be seen that the data spread in 2019 is smaller.
2. Of course, the same colors will be together because the left side is all red and the right side is all green.
3. The color distribution of the two charts in 2018 and 2019 is the same, but the location distribution is not the same. I said before about the distribution of standard deviation. Regarding the distribution of the mean, In 2018, most of the points have an average value between -0.5 and 0.5, while in 2019, a small number of points have an average value between -0.25 and 0.25, and the rest are unevenly distributed. In general, the amount of red in the remaining points is greater than that of green.
4. If the nearest neighbor classifier can be adjusted according to the actual situation after the first year of training, it will have better performance in the second year.

## Trading with labels

1. Year 2018 mean: 123.89, volatility: 11.20; year 2019 mean: 127.58, volatility: 16.70
2. 
3. Because I trade with label so the balance will grow week by week, so the min should be the first day of the year and the max should be the last day of the year
4. 2018 final value is 140.88, 2019 final value is 157.38
5. For both 2018 and 2019 the maximum duration for growing is 4 and decreasing is 1.

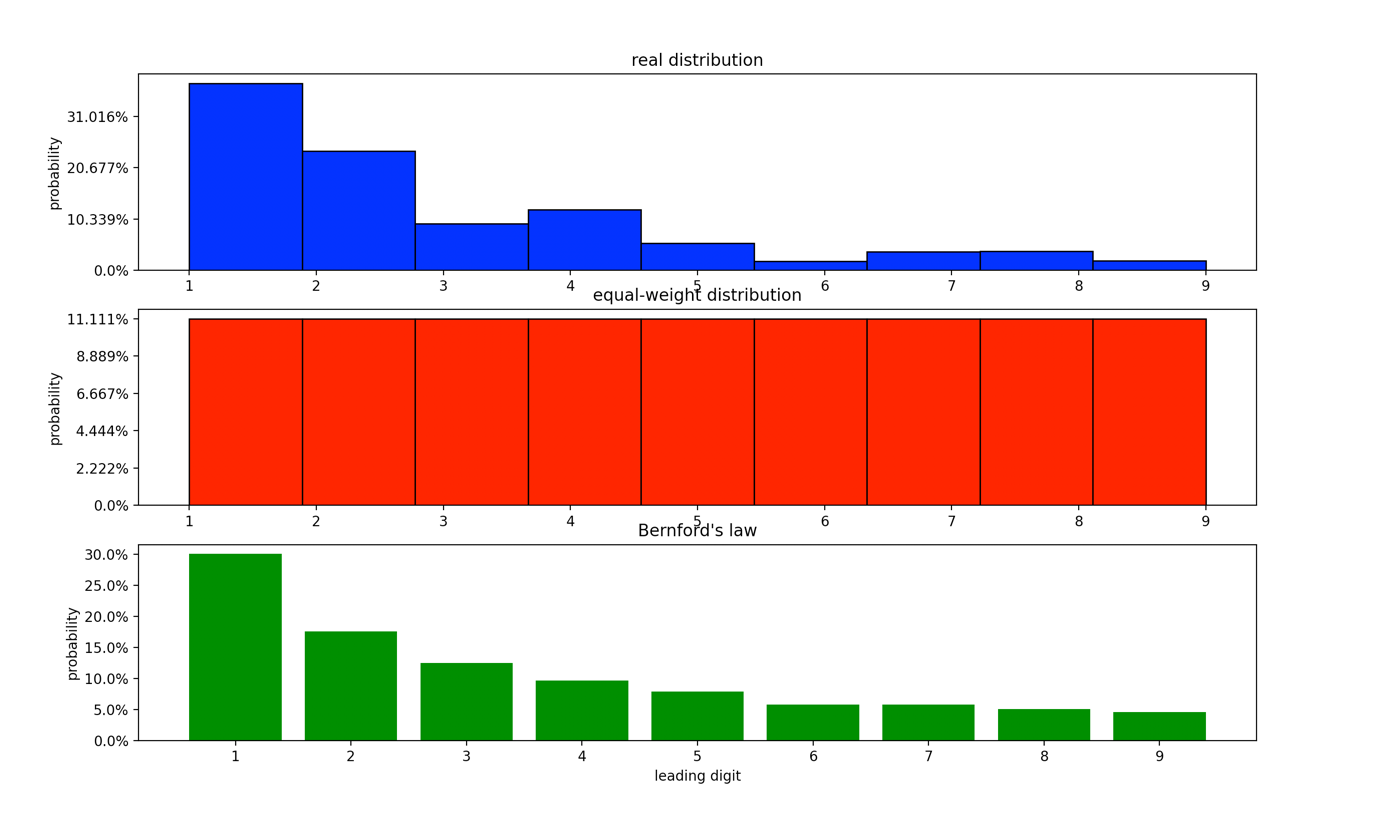
## Plot portfolio growth



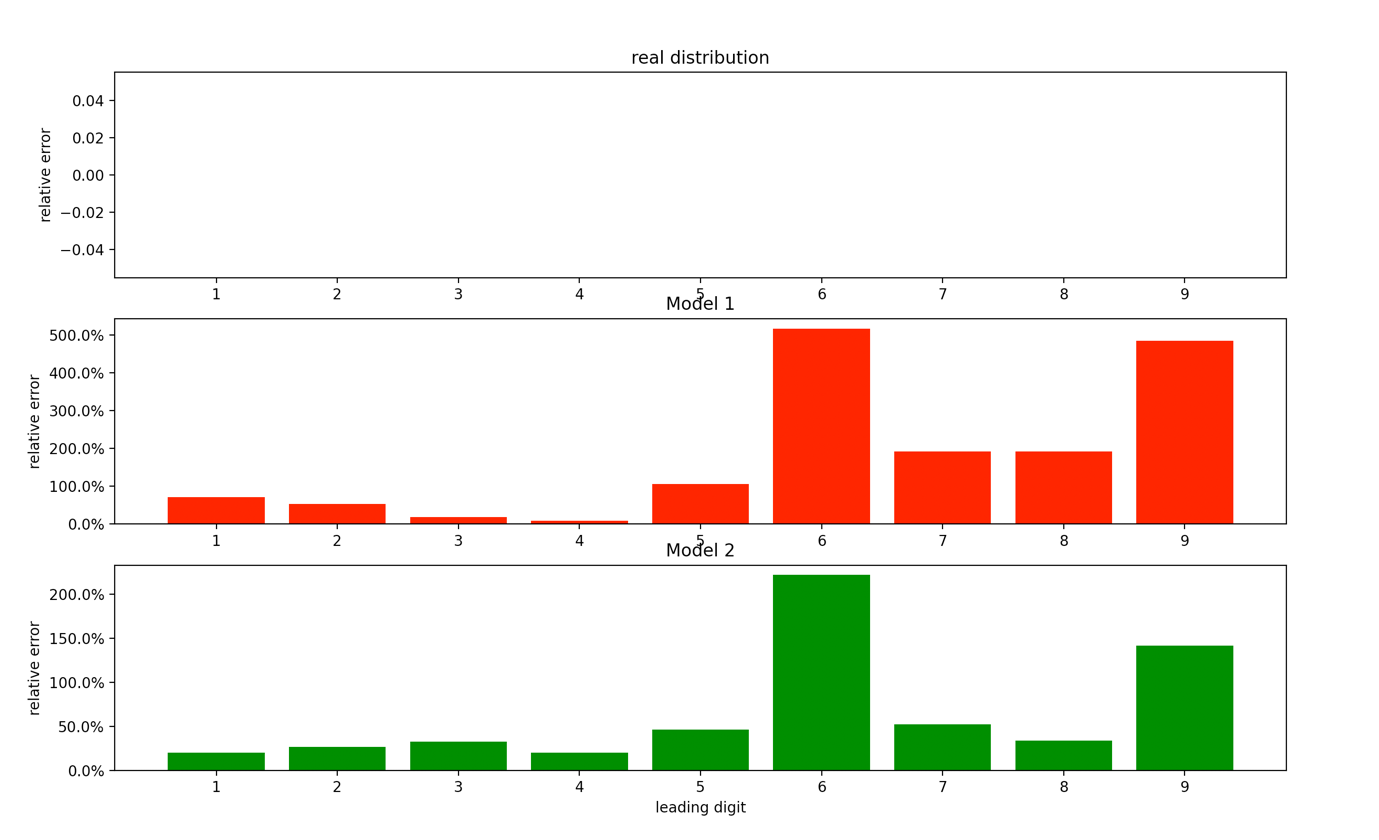
In this assignment the portfolio value is calculated by . And the end price is if next week is ‘red’, so it is the end price of the last day of this week; the open price is the first day’s open price we enter the market. Also the price is the initial capital which we enter the market.

## Retail price data

* Question 1: plot shown by run the code /Part2/question1.py



* Question 2: plot shown by run the code /Part2/question2.py and for the first plot (relative error for the real distribution) is empty, because the values for 1-9 is 0.



* Question 3: model 1: 0.11; model 2: 0.04. So, model 2 is closer to the real distribution. (both results are rounded to two decimal points, code shown in /Part2/question3.py)
* Question 4:
  + Asia: Japan

F: 0.421, 0.339, 0.082, 0.099, 0.018, 0.012, 0.006, 0.018, 0.006

RMSE: 0.155

* + Europe: France

F: 0.504, 0.248, 0.062, 0.072, 0.023, 0.009, 0.034, 0.033, 0.013

RMSE: 0.148

* + Middle East: United Arab Emirates

F: 0.429, 0.321, 0.0, 0.179, 0.036, 0.0, 0.036, 0.0, 0.0

RMSE: 0.153

So, the closest country is France.

* Question 5: