

1

1

I use last available value to clean the dataset and ensure that there is no gap in the dataset.

2

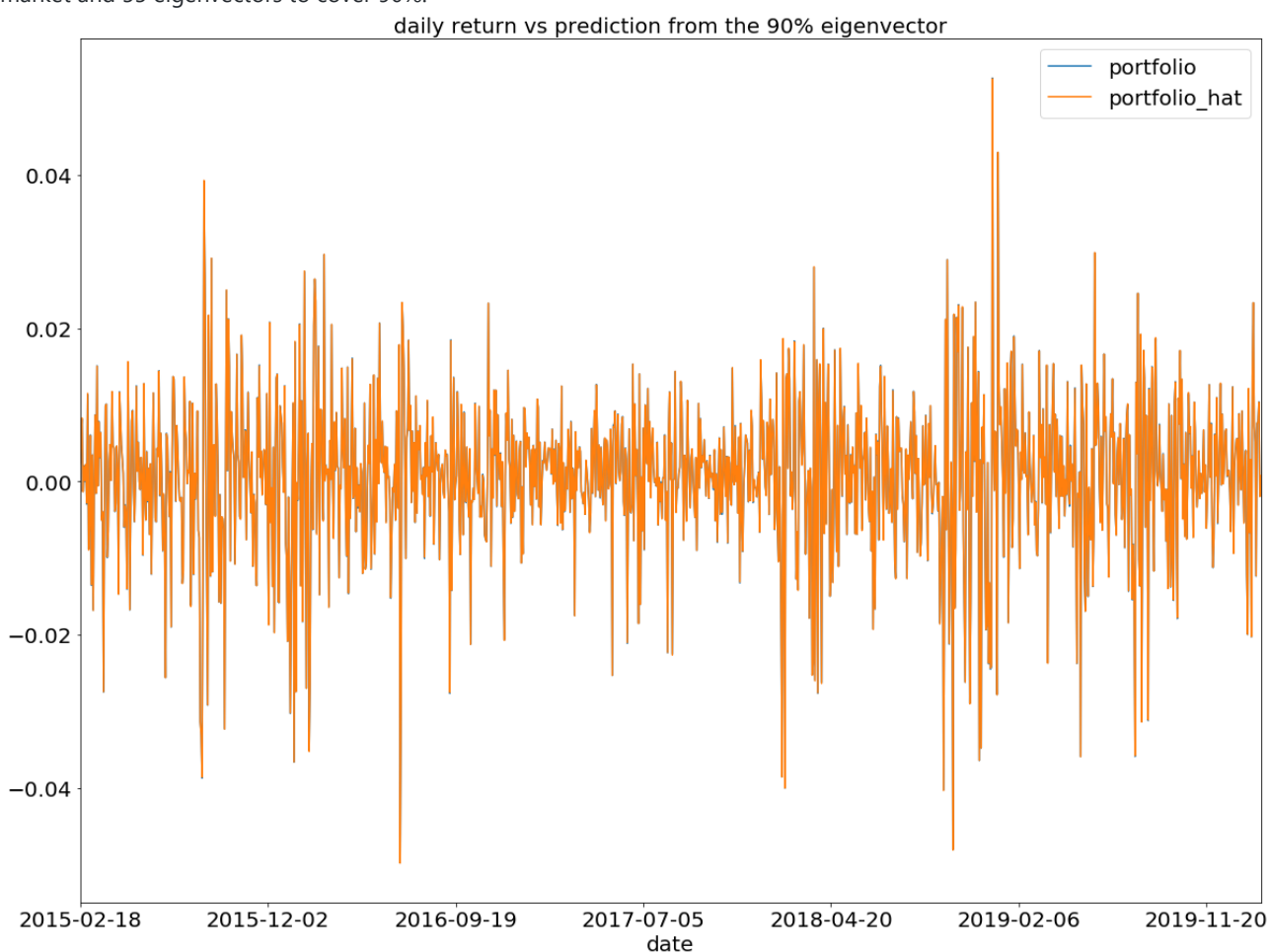
I generate the log return as daily return from the price dataset instead normal percentage change.

3

i use the package `numpy.linalg.eig` to generate the eigenvector from the covariance matrix and sort them based on their eigenvalue. It is interesting to notice that there is no eigenvector with negative eigenvalue. However, if there is a negative eigenvalue, we should double check the valid and accuracy of the dataset.

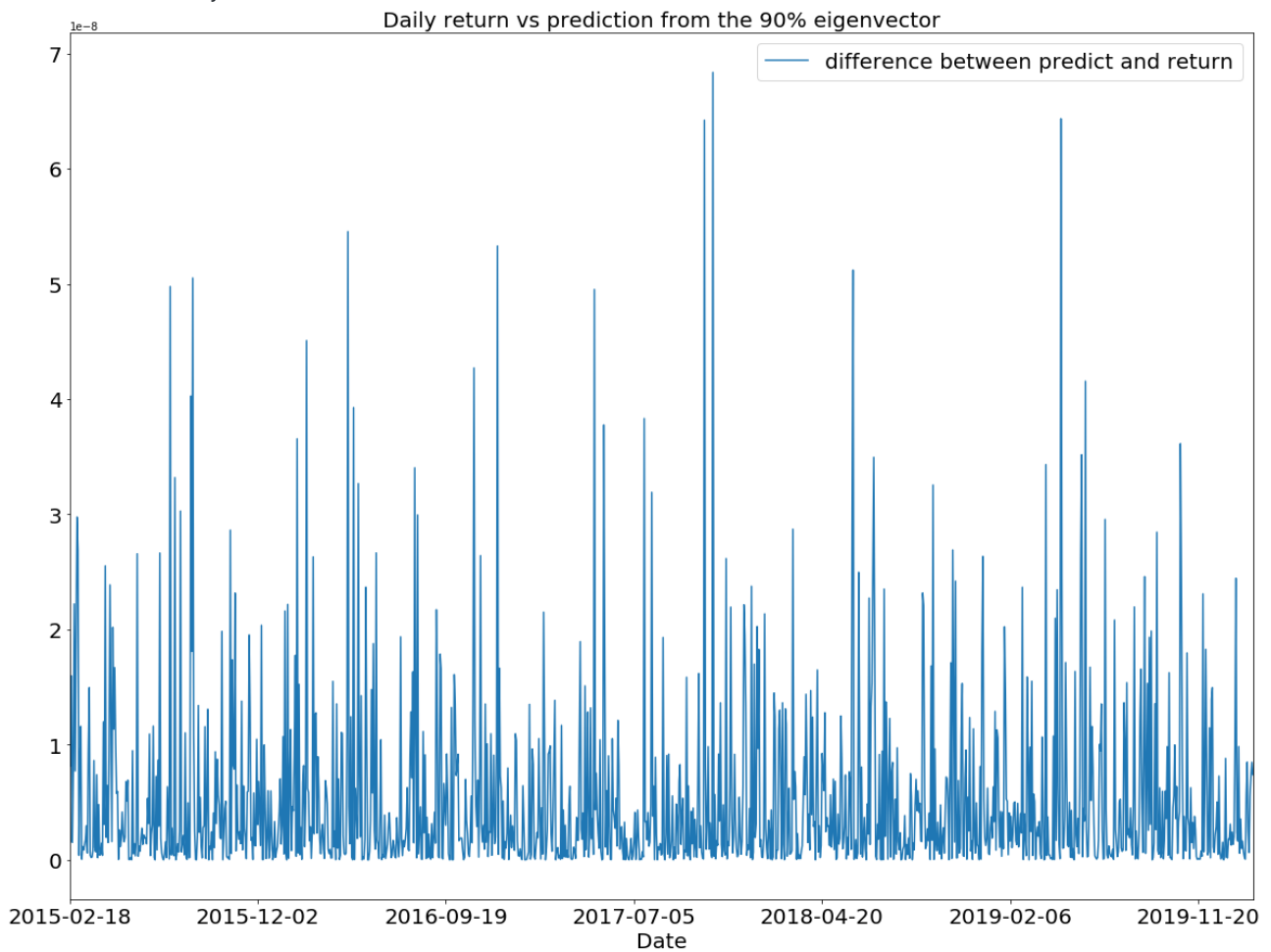
4

Using $\text{eigenvalue}/\text{sum}(\text{eigenvalue})$ as the weight, for my dataset, I need first 8 eigenvector to cover 50% information of the market and 55 eigenvectors to cover 90%.



5

After plotting the residual return stream from difference of 90% eigenvector dataset and original dataset, we can notice that the difference is every minimal.



2

1

I use SVD decomposition and pseudo-inverse to calculate the C^{-1} from the covariance matrix. Then I use it to calculate the $GC^{-1}G$. G is a constrain matrix that set the weight = 1 and the 10% of the portfolio is allocated to first 17 securities.

2

The portfolio looks good compare with the bench mark. However, it does a lot of short position. Thus, for a mutual fund, It maynot be a good portfolio since, unlike hedge fund, mutual fund does not short usually

market vs portfolio

