Portfolio Optimization with Deep portfolio theory and Unsupervised Machine Learning

Team members:

Yang Tao U40255851

Chengyu Xing U24930840

Yihe Liu U43817287

Siqi Jiang U83278011

Introduction:

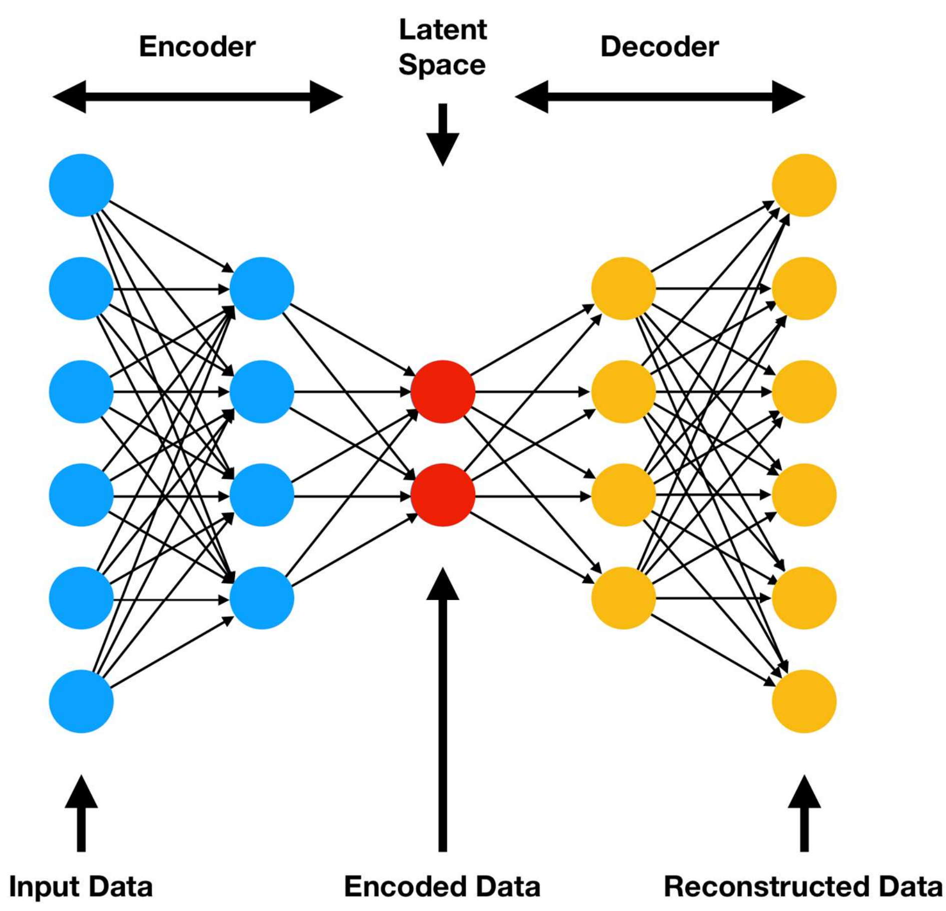
In our portfolio management project, we aim to train the ANN with 500 stocks' data from the US financial market to find out the outperformed set of stocks and then give proper weights to maximize the net worth, respectively. For the stocks sifting process, one traditional way is to use Principal Component Analysis (PCA), a linear technique for dimensionality reduction, to find out the few 'powerful' stocks which could represent the majority performance of the system in order to realize the purpose of stocks selection. However, this type of traditional method can only analyze data set in a linear manner, which is not efficient in recognizing more complex relationships, non-linear relationships, beneath our data set. Therefore, in order to improve efficiency in portfolio optimization, our project will apply deep learning theory, combines with the neural network, a method of machine learning, to model our data in a non-linear way to realize our financial goal.

Data

There are two crucial factors of a good portfolio: stock selection and stock allocation. First, we will download 500 stocks from S&P500 using yahoo finance or Bloomberg. Then, we will use the last good value method to clean the data to make sure there are no gaps between the data. Also, to ensure that the data is complete and accurate, we will use the yfinance package mainly. In order to analyze ten years of data, we will use eight years of data for training and two years for testing. Moreover, we will use Keras and Tensorflow to build our model for Autoencoder.

Autoencoder

Autoencoder is a popular unsupervised learning method for dimensionality reduction, which contains two processes: using the encoder to compress input into latent features, then reconstructing data through latent features. The optimal goal is to minimize the reconstruction error to optimize the encoder and decoder. Therefore, we can finally obtain representative lower-dimension code from initial input data. This method is permanently using input data as both input and output of the neural network, which provides wide versatility. Another advantage of the Autoencoder is that it can express non-linear characteristics compared to traditional PCA. As we mentioned before, Autoencoder and PCA are two similar techniques for dimensionality reduction, but they do have differences in some ways. PCA features are linearly uncorrelated with each other because of the orthogonal basis; in the meantime, autoencoder latent features could be correlated to others.



Moreover, the Autoencoder has better performance when at a lower dimension. In other words, with the same level dimension reduction, the Autoencoder makes a more accurate prediction than the PCA. Thus, Autoencoder, which can reduce the full and non-linear data down to 2 or 3 dimensions that contians all the necessary information, can save way more space then PAC. Nevertheless, Autoencoder has drawbacks on the computations and implementation. PCA requires less computing power and has a much easier implementation. Also, the overfitting, which exists for every machine learning problem, will taint the prediction by Autoencoder. However, there are differences that lead us to do this project, to explore the potential of the neural network, and to apply a new technology on the financial problem.

Algorithm

starting with the built of Autoencoder network, we will use Keras and TensorFlow, which are two most popular packages for machine learning problems, to construct the neural network. We will test different loss functions, such as mean square error and optimizers, such as Adam, sgd, to find the best algorithm for our problem. After the construction of Autoencoder, we will use it to decode the daily return of the index to latent features and explore with different levels of latent features. Then, we use latent features to reconstruct the daily returns of every individual stock. By sorting the stock name from lowest different to highest from their actual returns, we will know which stocks are the market maker, and which stocks do not follow the market. After gathering all necessary information, we will apply few optimizer theories to construct the portfolio to beat the market, such as mean-variance optimization.

By the end of project

In the end, after exciting exploration, we hope to find the answers to two questions: would the portfolio be profitable in the real financial market? And will Autoencoder show its advantages against the PCA when it is applied to financial problems?

Reference

Heaton, J. & Polson, N. & Witte, J.. (2016). Deep Portfolio Theory.

Blogs, Edgar Press. "Financial Portfolio Management with Deep Learning." The Intelligence of Information, December 21, 2016.