Project Description:
In the final project we do 2 parts:
The first part is using the neutral network to do the classifier of cat and non-cat.
The second network is using the deep neural network to predict the stock price.

Part I:
This part of the project is what we propose in the project concept.
The motivation of doing this part is after we watched the iPhone X presentations by apple we find it interests to do a project that can tell apart the different categories of pictures. So we do the Cat and None-Cat binary classifier.
Current Techniques: The market of the picture recognition is mature. And company like apple can do the 3D modeling of a object using infrared sensor. Between 1990s-2012 the main technique of picture is SVM and when in 2012 AlexNet using CNN got a massive win on SVM.

What we do: What we use in the lab is using the library of Keras and Tensorflow. So when we doing our project we try to build our own network with functions. We try to build each different simple part of sigmoid, Relu, cost evaluation, etc into functions and capsule it into a main function.By our project a user can realize a simple neural network that can be used to classify picture category. And we test it with 2 kinds of pictures (cat and animal) . And the result using binary classifer. First we do the 1 hidden layer test and using the learning rate of 0.0075 the test accuracy is 70 % and when we build a more deeper neural network using L hidden layer. L is 5 in our model. And the hidden layer unites is 7. The test accuracy is 69.0 %

Evaluation: We can see that when the model becomes more complex the training result is more better. But compare with what we have done in lab8 the open source package for tensorflow the accuracy is over 94%. Which means we need to add the complexity of our model to get a more accurate result: the probably improvement of the accuracy is:
1.The complexity of the model: more hidden layers and units
2.The data: we use the label **cat** and **animal** and animal could probably include the cat. So we need more distinguished categories.
3.Need to find a fit iteration number and learning rate to have a better accuracy.

Presentation: If we have 2 categories of pictures and anyone can use the function to tell apart these 2 categories by using the function.

What we learned: From this part of the project we have a deeper understand of how a neural network is built. Compare to using the package from keras and tensorflow we can train our own model.

Because what we've done have some overlap with the lab8 and we didn't know it when we start doing the project. We want to move on to do some interesting model that have more react with the daily life. So we think of to build a model to do the stock predict and compare the model.

Part II:
Method we use:
- Generate 1 factor synthetic return data (To compare the PCA with neural network method in terms of the ability to uncover market factors.)
- Increase the Num of factors (To test the feasibility of Auto-encoder)
- Apply Auto-encoder to real data (SPY 500 from bllomberg) and generate trading strategy.

Generate synthetic stock return data:

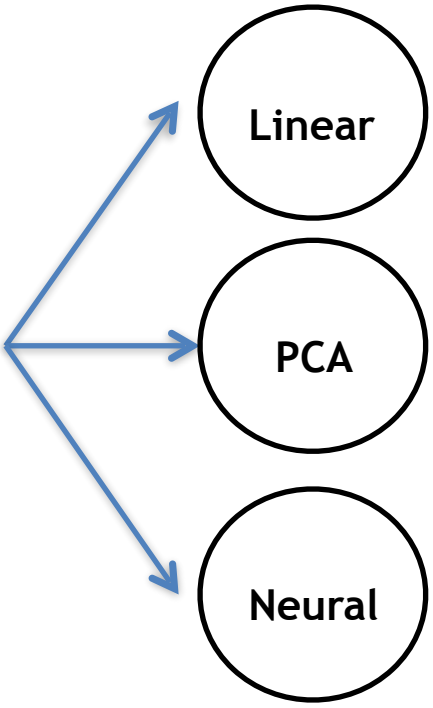
Basic Model Formula (J. Yeo and G. Papanicolaou)

$$R_{(N,\ T)} = L_{(N,\ P)} F_{(P,\ T)} + U_{(N,\ T)}$$

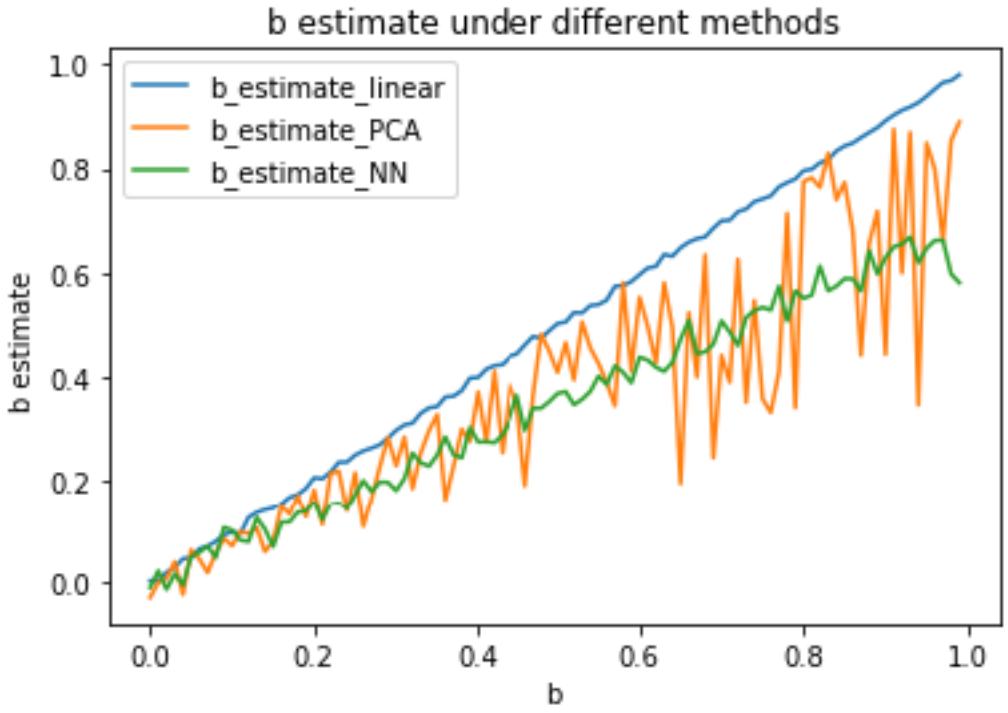
$$U_{(N,T)} = I^{1/2}_{(N,N)} \varepsilon_{(N,T)} B^{1/2}_{(T,T)}$$

$$\varepsilon_{(N,T)} \sim N(0,1)$$

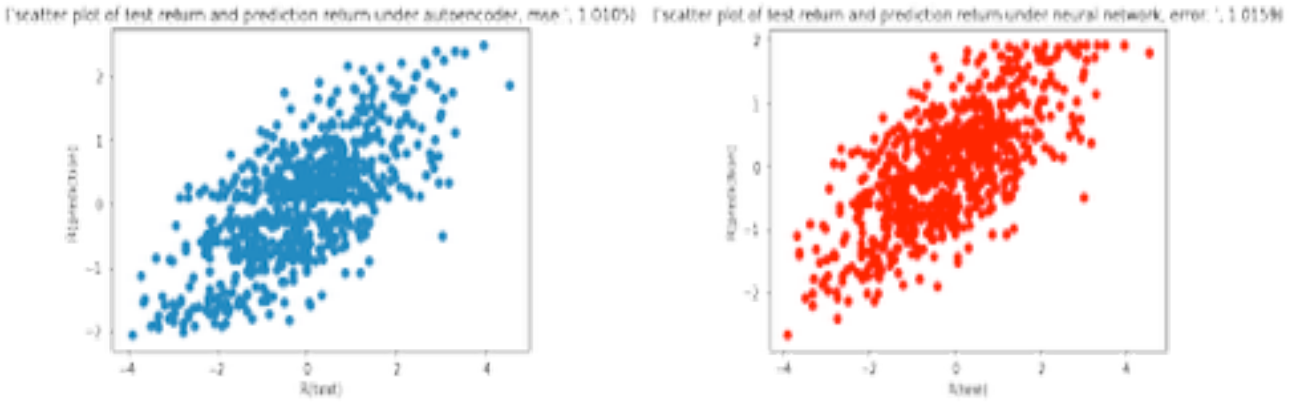
$$B_{(i,j)} = b_{estimation}^{\ |i-j|}$$



R: Synthetic returns
L: Factor Loadings
F: Factors
U: Model Residuals
* Here ‘b’ denotes the mean reversion and volatility of residuals



And then we increase the factor to 50:
to compare the dimension reduction through Auto-encoder and the pure neural network. we use the Relu as the Activation Function.



We find that the Autoencoder has a better performance than the pure neural network.

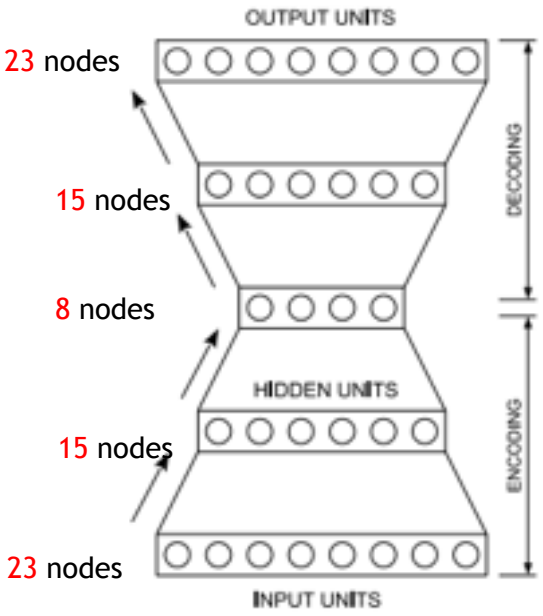
Then we use the real data log returns of SPY500 for last 10 years from Bloomberg.

And select 23 indicators of the market (L.Troiano, E. Mejuto, and P. Kriplani):

TABLE I
LIST OF INDICATORS

Indicator name	Type of Indicator
Absolute Price Oscillator (APO)	Type of Indicator
Aroon	Momentum
Aroon Oscillator	Momentum
MESA Adaptive Moving Average (MAMA)	Overlap studies
Average Directional Movement Index (ADX)	Momentum
Average Directional Movement Index Rating	Momentum
Average True Range (ATR)	Volatility
Balance of Power (BOP)	Momentum
Bollinger Bands (BBANDS)	Overlap studies
Bollinger Bandwidth	Overlap studies
%B Indicator	Overlap studies
Chaikin A/D Oscillator	Volume
Chande Momentum Oscillator (CMO)	Momentum
Commodity Channel Index (CCI)	Momentum
Directional Movement Index	Momentum
Double Exponential Moving Average (DEMA)	Overlap studies
Exponential Moving Average (EMA)	Overlap studies
Kaufman's Adaptive Moving Average (KAMA)	Overlap studies
Minimum and Maximum value over period	-
Moving Average (MA)	Momentum
Moving Average Convergence/Divergence (MACD)	Momentum
Momentum	Momentum
Money Flow Index (MFI)	Momentum
On Balance Volume	Volume
Percentage Price Oscillator (PPO)	Momentum
Plus Directional Indicator	Momentum
Plus Directional Movement	Momentum
Relative Strength Index (RSI)	Momentum
Relative Vigor Index (RVI)	Momentum
Rate of change ratio (ROC)	Momentum
Parabolic SAR	Overlap studies
Stochastic Oscillator	Momentum
Triple Exponential Moving Average (TEMA)	Overlap studies
Triangular Moving Average (TRIMA)	Overlap studies
1-day ROC of a Triple Smooth EMA (TRIX)	Momentum
Ultimate Oscillator	Momentum
Weighted Moving Average (WMA)	Overlap studies
Williams' Percent Range (%W)	Momentum

Autoencoder mode:



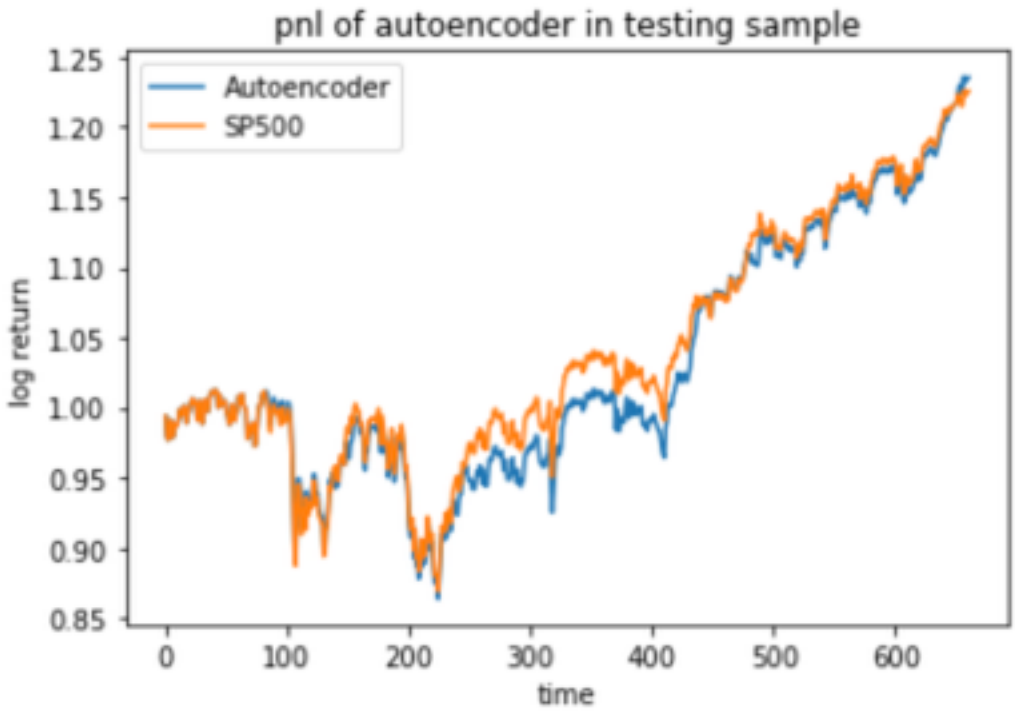
Classification: SVM

Activation Function: Sigmoid

Loss Function: MSE

Input units(1 Rise
0. Hold
-1 drop)

The result:



Here auto encode is using Autoencoder+SVM, we can see that the predict figure is close to the real data.

Conclusion: The stock price predict using neural network is a new field and have no standard solutions , we do this interdisciplinary project intended to expand the application of the
The improvement we can make:
1.Simply classify the trend we predicted into up or down patterns sometime is not accurate enough to predict the stock market.
2. For real data, we only use technical indicators as features. The real meaning for the features remain unclear.