Using Machine Learning for medium frequency derivative portfolio trading

Abhijit Sharang, Chetan Rao

Stanford University, Netflix

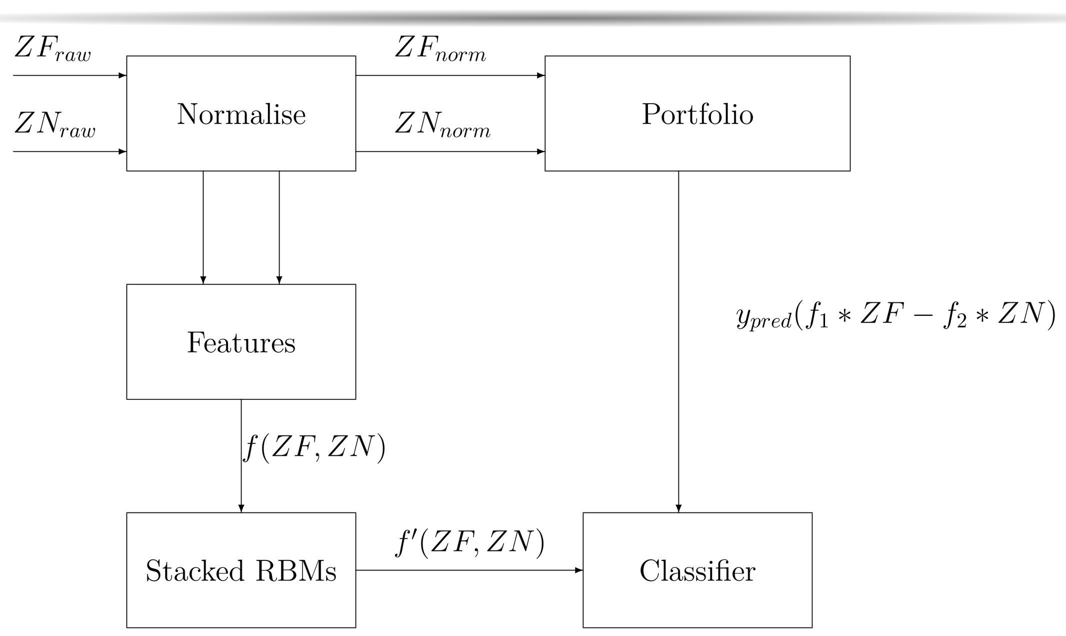
Introduction

We use machine learning for designing a medium frequency trading strategy for the portfolio of 5 year and 10 year US Treasury Notes derivative. We formulate the problem as a classification problem where we predict the direction of movement of the portfolio and use the information to place a buy trade or a sell trade. The experimentation shows that this pipeline can be helpful in obtaining a profitable trade.

Motivation

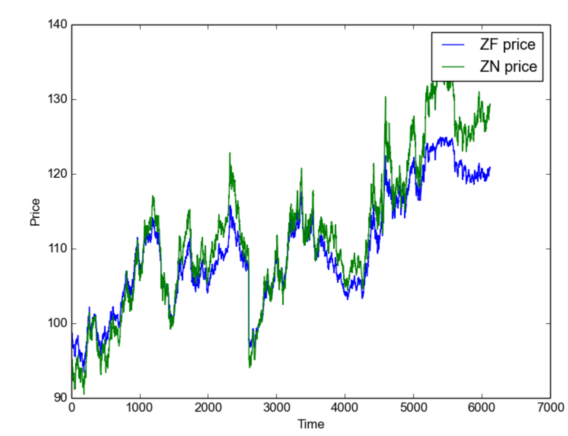
Using Machine Learning in finance is challenging because the signal to noise ratio is low. A good amount of domain expertise is required for engineering features which can assist in solving a meaningful classification or regression problem. However, with deep learning techniques, it is possible to use raw features at the beginning of the pipeline and construct meaningful features from the raw features, assuming that there are latent features which explain patterns in the input.

ML pipeline

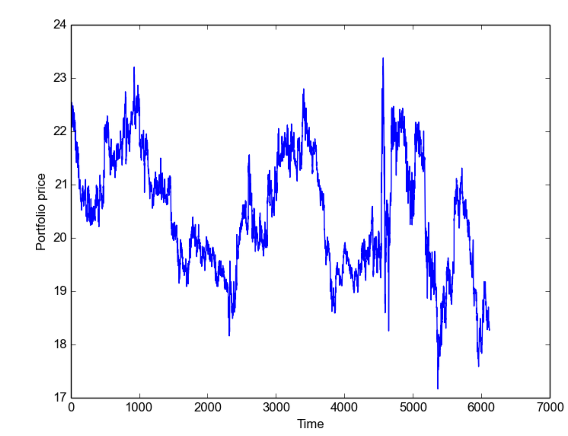


Data

The daily time series for individual instruments and the spread are displayed.



a: Instrument price



b: Portfolio price

Portfolio price is computed by buying 1 unit of ZF and selling hedge adjusted 1 unit of ZN.

- Description of X and Y
  - dependent variable - 1 if the portfolio rises at the end of the week, -1 otherwise
  - features - weekly and biweekly trends of ZF and ZN
- Description of training,validation and test sets.
  - Training : 80% of total data - 2325 +ve and 2608 -ve.
  - Validation : Next 10% of total data - 230 +ve and 250 -ve.
  - Test : Final 10% of the total data - 295 +ve and 315 -ve

Classification Algorithms

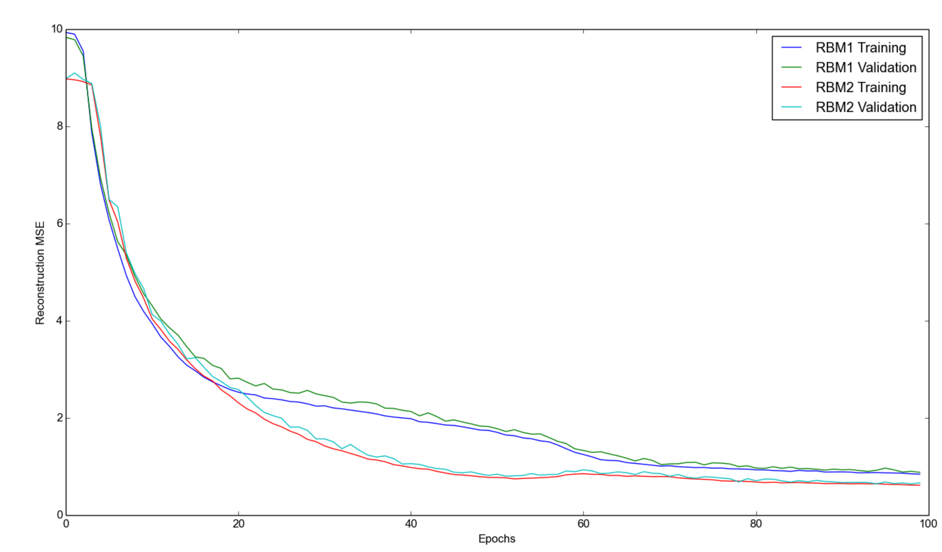
We use three versions of classification algorithms. For neural network, we perform fine-tuning by initialising the weights with those learned during belief propagation.

- Logistic regression - log likelihood maximisation
- Support Vector Machine - margin maximisation
- Neural Network - MSE minimisation

Restricted Boltzman machine

- We use belief network to learn the latent representations of the input features.
- Reconstruction weights for each RBM can be learned greedily.
- First RBM - Gaussian-Bernoulli. Second RBM - Bernoulli
- The number of epochs, mini-batch size and the learning rate are chosen via a validation set.

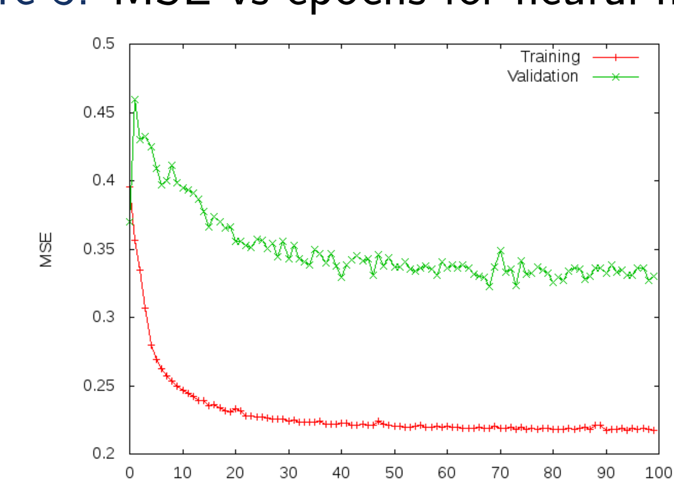
Figure 2: Reconstruction MSE vs epochs for two stacked RBMs in the belief network



Fine tuning

- Validation set used to decide the number of training epochs for neural network and logistic regression and constant C for SVM.

Figure 3: MSE vs epochs for neural network



Results

Table 1: Recall rate for each direction

Algorithm	Actual ⇑	Actual ⇓
SVM	60%	61.69%
Logistic Regression	58.73%	62.37%
Neural Network	57.77%	59.66%

Table 2: Precision rate for each direction

Algorithm	Actual ⇑	Actual ⇓
SVM	62.58%	58.59%
Logistic Regression	62.5%	59.09%
Neural Network	60.46%	56.95%

Figure 4: ROC curve for SVM

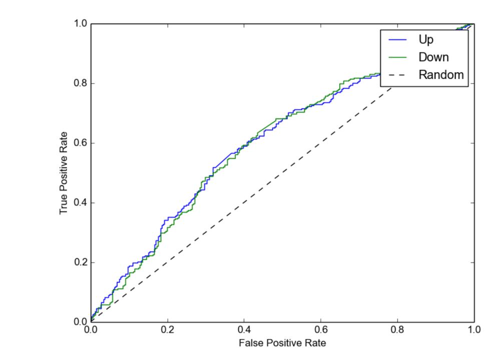
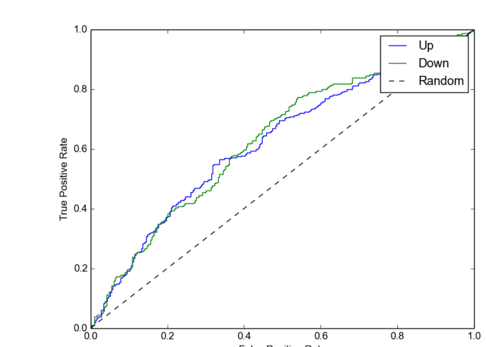


Figure 5: ROC curve for Neural Network



PNL performance