**Machine Learning Engineer Nanodegree**

**Capstone Project**

Zilvinas Marozas  
July 14th, 2018

**I. Definition**

**Project Overview**

Predicting financial instruments prices using machine learning has been attempted by many researchers with varying degree of success. Predicting Commodity Futures market is very important not only for speculators, money managers, etc. It is vital for many companies (food producers, oil refineries, farmers) to estimate these prices with certain degree of accuracy since profitability and sometimes very survival of the company and many jobs could depend on correct estimation of commodity prices. Commodities Futures market is very important to US economy and is highly regulated by US government. Various US government agencies like **Commodity Futures Trading Commission** (CFTC) collect lots of data and makes it freely available to the public. The Commitment of Traders (COT) Report is conducted by the Commodity Futures Trading Commission (CFTC) detailing the open interest in each futures and options on commodities markets containing 20 or more traders holding position sizes large enough to meet the CFTC’s reporting level. The purpose of this report is to provide traders with transparency in regards to the open interest in various futures markets and the sizes of those positions for different groups of traders. Therefore, this additional data could be used to supplement pricing information (closing price, and volume) while predicting commodities prices.

This project seeks to utilize Deep Learning models, Long-Short Term Memory (LSTM)

Neural Network algorithm, to predict prices of Corn Commodity Futures. I will use Time Series data (Closing Price and Volume) and data from COT report. Classical neural networks called Multilayer Perceptrons, or MLPs for short, can be applied to

sequence prediction problems i.e. MLPs do offer great capability for sequence prediction but still suffer from key limitation of having to specify the scope of temporal dependence between observations explicitly upfront in the design of the model. Recurrent Neural Networks, or RNNs for short, are a special type of neural network designed for sequence problems. There are a number of RNNs, but it is the LSTM that delivers on the promise of RNNs for sequence prediction. LSTMs have internal state, they are explicitly aware of the temporal structure in the inputs, are able to model multiple parallel input series separately. Given all these advantages of LSTM for sequence predictions , I will use Keras and LSTM for this project.

**Problem Statement**

In this project we will try to predict closing weekly price of Corn Commodity Futures. In order to perform this prediction we will create a dataset that includes weekly Corn Futures closing prices as well as Long Open Interest and Short Open Interest of Processors/Users( sometimes they are called Commercials) from COT reports and by using this dataset we will try to predict next week’s prices. Why do we need to complicate our model and add additional data from COT report instead of just using Pricing information. I do believe that using only historical prices alone is not sufficient to predict prices of commodities and especially it is very hard to predict turning points, when commodity reverses the previous trend and starts moving to the opposite direction. Therefore, we need to take into account “Fundamentals data” in order to predict these prices with some degree of consistency. However, obtaining “Grains Fundamentals Data” (area planted, area harvested, exports, etc.) is difficult and expensive and it will make machine learning model much more complicated. My hypothesis is that we can substitute “Fundamentals Data” with data from COT report by tracking open interest(long open interest and short open interest) of Processors/Users. This group of traders represents well-funded enterprises that have “deep pockets” in order to conduct research and also have inside information since they are typically large users of grains. For this project I will use a Long Short Term Memory networks (LSTM).

**Metrics**

With forecasts and actual values in their original scale, we can then calculate an error score for the model. In this case, we calculate the Root Mean Squared Error (RMSE) that gives error in the same units as the variable itself.

Benchmark model we will evaluate on the validation dataset. We do this using the walk-forward validation method. In essence, we step through the validation dataset time step by time step and get predictions. Once predictions are made for each time step in the validation dataset, they are compared to the expected values and a Root Mean Squared Error (RMSE) score is calculated. Once we have RMSE for both models we will compare them and decide whether or not LSTM model is good choice for our problem.

**Analysis**

*(approx. 2-4 pages)*

**Data Exploration**

In this section, you will be expected to analyze the data you are using for the problem. This data can either be in the form of a dataset (or datasets), input data (or input files), or even an environment. The type of data should be thoroughly described and, if possible, have basic statistics and information presented (such as discussion of input features or defining characteristics about the input or environment). Any abnormalities or interesting qualities about the data that may need to be addressed have been identified (such as features that need to be transformed or the possibility of outliers). Questions to ask yourself when writing this section:

* *If a dataset is present for this problem, have you thoroughly discussed certain features about the dataset? Has a data sample been provided to the reader?*
* *If a dataset is present for this problem, are statistics about the dataset calculated and reported? Have any relevant results from this calculation been discussed?*
* *If a dataset is****not****present for this problem, has discussion been made about the input space or input data for your problem?*
* *Are there any abnormalities or characteristics about the input space or dataset that need to be addressed? (categorical variables, missing values, outliers, etc.)*

**Exploratory Visualization**

In this section, you will need to provide some form of visualization that summarizes or extracts a relevant characteristic or feature about the data. The visualization should adequately support the data being used. Discuss why this visualization was chosen and how it is relevant. Questions to ask yourself when writing this section:

* *Have you visualized a relevant characteristic or feature about the dataset or input data?*
* *Is the visualization thoroughly analyzed and discussed?*
* *If a plot is provided, are the axes, title, and datum clearly defined?*

**Algorithms and Techniques**

In this section, you will need to discuss the algorithms and techniques you intend to use for solving the problem. You should justify the use of each one based on the characteristics of the problem and the problem domain. Questions to ask yourself when writing this section:

* *Are the algorithms you will use, including any default variables/parameters in the project clearly defined?*
* *Are the techniques to be used thoroughly discussed and justified?*
* *Is it made clear how the input data or datasets will be handled by the algorithms and techniques chosen?*

**Benchmark**

A benchmark in forecast performance provides a point of comparison If a model achieves performance at or below the benchmark, the technique should be fixed or abandoned. Three properties of a good technique for making a benchmark forecast are:

* Simple: A method that requires little or no training or intelligence.
* Fast: A method that is fast to implement and computationally trivial to make a prediction.
* Repeatable: A method that is deterministic, meaning that it produces an expected output given the same input.

A common algorithm used in establishing a baseline performance for time series forecasting is the persistence algorithm. The persistence algorithm uses the value at the current time step (t) to predict the expected outcome at the next time step (t+1). This satisfies the three above conditions for a baseline forecast. Due to simplicity of benchmark model it will take into account only the price and will disregard other feature like Open Interest from COT report.

**III. Methodology**

*(approx. 3-5 pages)*

**Data Preprocessing**

In this section, all of your preprocessing steps will need to be clearly documented, if any were necessary. From the previous section, any of the abnormalities or characteristics that you identified about the dataset will be addressed and corrected here. Questions to ask yourself when writing this section:

* *If the algorithms chosen require preprocessing steps like feature selection or feature transformations, have they been properly documented?*
* *Based on the****Data Exploration****section, if there were abnormalities or characteristics that needed to be addressed, have they been properly corrected?*
* *If no preprocessing is needed, has it been made clear why?*

**Implementation**

In this section, the process for which metrics, algorithms, and techniques that you implemented for the given data will need to be clearly documented. It should be abundantly clear how the implementation was carried out, and discussion should be made regarding any complications that occurred during this process. Questions to ask yourself when writing this section:

* *Is it made clear how the algorithms and techniques were implemented with the given datasets or input data?*
* *Were there any complications with the original metrics or techniques that required changing prior to acquiring a solution?*
* *Was there any part of the coding process (e.g., writing complicated functions) that should be documented?*

**Refinement**

In this section, you will need to discuss the process of improvement you made upon the algorithms and techniques you used in your implementation. For example, adjusting parameters for certain models to acquire improved solutions would fall under the refinement category. Your initial and final solutions should be reported, as well as any significant intermediate results as necessary. Questions to ask yourself when writing this section:

* *Has an initial solution been found and clearly reported?*
* *Is the process of improvement clearly documented, such as what techniques were used?*
* *Are intermediate and final solutions clearly reported as the process is improved?*

**IV. Results**

*(approx. 2-3 pages)*

**Model Evaluation and Validation**

In this section, the final model and any supporting qualities should be evaluated in detail. It should be clear how the final model was derived and why this model was chosen. In addition, some type of analysis should be used to validate the robustness of this model and its solution, such as manipulating the input data or environment to see how the model’s solution is affected (this is called sensitivity analysis). Questions to ask yourself when writing this section:

* *Is the final model reasonable and aligning with solution expectations? Are the final parameters of the model appropriate?*
* *Has the final model been tested with various inputs to evaluate whether the model generalizes well to unseen data?*
* *Is the model robust enough for the problem? Do small perturbations (changes) in training data or the input space greatly affect the results?*
* *Can results found from the model be trusted?*

**Justification**

In this section, your model’s final solution and its results should be compared to the benchmark you established earlier in the project using some type of statistical analysis. You should also justify whether these results and the solution are significant enough to have solved the problem posed in the project. Questions to ask yourself when writing this section:

* *Are the final results found stronger than the benchmark result reported earlier?*
* *Have you thoroughly analyzed and discussed the final solution?*
* *Is the final solution significant enough to have solved the problem?*

**V. Conclusion**

*(approx. 1-2 pages)*

**Free-Form Visualization**

In this section, you will need to provide some form of visualization that emphasizes an important quality about the project. It is much more free-form, but should reasonably support a significant result or characteristic about the problem that you want to discuss. Questions to ask yourself when writing this section:

* *Have you visualized a relevant or important quality about the problem, dataset, input data, or results?*
* *Is the visualization thoroughly analyzed and discussed?*
* *If a plot is provided, are the axes, title, and datum clearly defined?*

**Reflection**

In this section, you will summarize the entire end-to-end problem solution and discuss one or two particular aspects of the project you found interesting or difficult. You are expected to reflect on the project as a whole to show that you have a firm understanding of the entire process employed in your work. Questions to ask yourself when writing this section:

* *Have you thoroughly summarized the entire process you used for this project?*
* *Were there any interesting aspects of the project?*
* *Were there any difficult aspects of the project?*
* *Does the final model and solution fit your expectations for the problem, and should it be used in a general setting to solve these types of problems?*

**Improvement**

In this section, you will need to provide discussion as to how one aspect of the implementation you designed could be improved. As an example, consider ways your implementation can be made more general, and what would need to be modified. You do not need to make this improvement, but the potential solutions resulting from these changes are considered and compared/contrasted to your current solution. Questions to ask yourself when writing this section:

* *Are there further improvements that could be made on the algorithms or techniques you used in this project?*
* *Were there algorithms or techniques you researched that you did not know how to implement, but would consider using if you knew how?*
* *If you used your final solution as the new benchmark, do you think an even better solution exists?*