# Machine Learning Engineer Nanodegree

## Capstone Proposal

Moustafa Banbouk
December 16th, 2017

## Proposal

### Domain Background

The scope of this project is to predict the price of Cryptocurrencies, a relatively new type of currency that started in 2008 with Bitcoin. The hype of cryptocurrencies returned back in 2017 as we witnessed an increase of some cryptocurrencies like bitcoin by more than 2400% as demonstrated in the following graph.


Cryptocurrencies belong to a highly vulnerable market that changes on daily basis. The frequency of price change for cryptocurrencies is very high as we are seeing a 30% increase or decrease in one day in some cases. The increase or decrease of cryptocurrencies depend on various parameters including marketcap,

liquidity, value proposition, number of coins in circulation but at the end all these factors are influenced by the number of buyers and sellers competing for a particular cryptocurrency coin. Similar to stock prices, if we can predict high increase or decrease, we will be able to invest in a smarter manner and optimize our trading strategy. We can run our prediction algorithm on multiple currencies and decide which coins we should, which should we buy and which to keep.

There are numerous papers written on Cryptocurrency trading, the most famous of which is a reddit post by "joskye", a cryptocurrency investor and holder.

- Title: The Intelligent Investors Guide to Cryptocurrency
- Link: https://np.reddit.com/r/Particl/comments/7f28ja/the_intelligent_investors_guide_to_cryptocurrency/

As for the academic research in the field, I would like to refer to the following videos on which my algorithms were based:

The project will be based on the following videos from Siraj Raval and Sandeep Sharma.

Predicting Stock Prices - Learn Python for Data Science #4

How to Predict Stock Prices Easily - Intro to Deep Learning #7

Predicting Stock Price: A Machine Learning Project

As for the referenced academic paper, the project is based on the "Stock Prediction using Machine Learning a Review Paper" with the below details:

- Paper Title: "Stock Prediction using Machine Learning a Review Paper"
- Institute: Information Technology (I.T.), Vidyalankar Institute of Technology, Mumbai, Maharashtra, India
- Published in: The International Journal of Computer Applications in April 2017

Regarding the data source, we will be downloading fresh cryptocurrency historical data from the well known coinmarket cryptocurrency market capitalizations monitoring website http://www.coinmarketcap.com through the site's JSON APIs.

## Problem Statement

As highlighted in the previous paragraph, the main problem with cryptocurrency is their high volatility and therefore, high fluctuations in their value. As an investor, we need to know when to buy, sell or hold a particular investment in a

particular cryptocurrency coin. The main decision differentiator is the expected value of such cryptocurrency that can be predicted using machine learning algorithms.

## Datasets and Inputs

To be able to trade safely, we need a mechanism to predict the cryptocurrency closing value based on available and well known input parameters.

### Input Features

Regarding the feature inputs, we will be using the following features per coin:

1. Closing value of all the previous 8 days (8 Features)
2. Closing value before 2 weeks (1 Feature)
3. Closing value before 3 weeks (1 Feature)
4. Closing value before 4 weeks (1 Feature)

In total, we will be having 8 + 1 + 1 + 1 = 11 Feature per each coin and since I am planning to derive the above features for the best performing 5 altcoins and therefore, we will be having 11 x 5 = 55 Features. While developing the solution, if I felt that the performance of my program is not up to the required level due to the high number of features, I may select 2 or 3 altcoins to work with instead of 5.

### Output Feature

I am planning to have as an output feature a vector of including the expected value of the altcoins I am analyzing. The plan is to have a vector of 5 values carrying the predicted value of the altcoin for the next day.

### Datasets

Datasets for the historical values of cryptocurrencies can be easily obtained from online cryptocurrency exchanges and for this project we will be deriving the required cryptocurrency historical values from http://www.coinmarketcap.com through the site's JSON APIs.

Number of examples in the dataset: we are planning to use the values of the last 365 days without taking into consideration the last day to overcome lookahead bias. Therefore, we should be having 365 rows worth of data with each row having 55 features if we took 5 altcoins (Bitcoin, Ethereum, Ripple, Bitcoin Cash and Litecoin). If we found that the algorithm is slow, we will decrease the number

of altcoins to be analyzed. It should be noted that we will consecutively divide the dataset to training, validation and testing datasets with ratios of 80%, 10%, 10%.

The output variable is a vector of 5 values carrying the predicted value of the altcoin for the next day.

## Solution Statement

We will be machine learning to predict the expected cryptocurrency value based on historical values of the same altcoin and other altcoins. To achieve this goal, we will be using machine learning algorithms and compare among their performance. This will allow us to derive the best algorithm that can be used in such an endeavor.  As we are dealing with continuous numerical data, the machine learning problem at hand is a pure regression problem and therefore, we will be using supervised learning regression algorithms to predict expected cryptocurrency future values.

## Benchmark Model

## Evaluation Metrics

The evaluation metric in our project is the cryptocurrency value vs USD (ex. ETH/USD is the value of 1 Ether altcoin with respect to USD). Our benchmark will be available forecast from one of the best known online sites WalletInvestor (http://walletinvestor.com/forecast) as this website provides long and short term predictions for each cryptocurrency.

To derive the performance of our system, we will compare the predicted cryptocurrency value from our algorithm to that of WalletInvestor website using the short term online forecasts.

As a benchmark, we will be using the Linear Regression with only input features as the cryptocurrency value for the last 8 days as a baseline for comparison with our algorithm.

To quantify the performance of our model, we will be the mean square error as metric.

## Project Design

As the project at hand is a supervised learning problem then we will be taking the below Steps in building our solution:

1. Data Loading: Using coinmarketcap.com API, download a fresh historical data of the main cryptocurrencies
2. Data Exploration and Anomaly detection: Investigate the dataset to determine missing information, anomaly values and the quantity/quality of each feature.
3. Data Normalization / Pre-processing: We will scale the input and output data by dividing the value of each cryptocurrency by the initial value before one year
4. Identify Features and Target Columns: In this step, we will select the features as highlighted in the "Datasets and Inputs" paragraph:

   > The target columns are the predicted cryptocurrency values for the next day. Features normalization should also be employed in this step to insure the best performance

5. Visualize Data: Visualize features to determine correlations and importance
6. Training and Testing Data Split: We will split the data into training, validation and testing subsets (Ration: 80-10-10) without shuffling them
7. Training and Evaluation Models: Based on the provided data, we will analyze:
   - strengths/weaknesses models and compare them
   - Fit various models to varying sizes of trained data
   - measure the mean squared error and select the best model, in addition, we will produce multiple tables for each model including set size, training time, prediction time, mean squared error on training data, mean squared error on test data.

   Using sci-kit learn, we will be testing various models (using their default parameters) including linear regression, logistic regression, gaussian naïve bayes, decision trees, random forests, SVMs and KNN

8. Comparison of top models: Select the top models after comparing their performance with each other
9. Select the best model: choose the best performance, faster and cost CPU/memory effective model
10. Model Tuning: Optimize the selected model hyper-parameters

---