# Machine Learning Engineer Nanodegree

## Capstone Proposal

Optimizing scalp trading with Reinforcement Learning.

Mukesh Yadav

Aug 12th, 2019

## Proposal

## Domain Background

Scalping is a trading style that specializes in profiting off small price changes, generally after a trade is executed and becomes profitable. It requires a trader to have a strict exit strategy because one large loss could eliminate the many small gains the trader worked to obtain. Having the right tools such as a live feed, a direct-access broker and the stamina to place many trades is required for this strategy to be successful.

Scalping is based on an assumption that most stocks will complete the first stage of a movement. But where it goes from there is uncertain. After that initial stage, some stocks cease to advance while others continue.

[1] https://www.investopedia.com/articles/trading/05/scalping.asp

## Problem Statement

While doing a scalping trading it's crucial to know when to enter and some extent when to exist. Doing scalping is a stressful task and it requires dedication, one has to sit in front of the system to make the decision in a few minutes. Since this is a very time consuming and stressful way of trading an automated system can do a better job than compared to a trader who needs to look at the screen always.

## Datasets and Inputs

To do this project one requires an OHLC candle of minutes data. The data source is taken from Kaggle and its open for use.

This data comprises of 200 instruments data from NSE stocks exchange.

Data URL.

https://www.kaggle.com/muke5hy/starter-nifty-200-minutes-data

## Solution Statement

The solution to this problem is to when deciding when to buy and when to sell. Though it is very difficult to come up with the model which is always in profit, my objective for the model is to have overall profitability.

## Benchmark Model

Total PnL should be the **Buy & Hold** return. If the model starts trading at the start of the month with a stock value of 1000 if the end of the month its value goes to 1300 then 300 is a profit in buy and hold strategy. The model should give a profit of 300+ to be used in production or on live trading system.

## Evaluation Metrics

Since this is a trading system evaluation metrics is limited to total Profit and Loss of said stock ticker.

The system will show daily, weekly and Monthly PnL which can act as its metrics.

## Project Design

Design of the project is crucial for readability and maintenance, any project which is not readable or properly modularized will eventually become a tech Dept. For this purpose, I have tried to design it as modular as possible.

The project is divided into the following packages.

1. Actions: This package contains the method taking a trade and deciding price and adding slippage in case if we are doing backtest.

2. Environments: A Gym based environment for reinforcement learning.
3. Exchanges: For creating a trading bot it is necessary to have backtesting and live environment. For this project, this package will contain the simulated data provider.
4. Features: Implementation for Sklearn's pipeline.
5. Rewards: Reinforcements core component is rewarding which has been designed in this package.
6. Slippage: Simple package to calculate the Slippage while doing backtesting.
7. Strategies: Training strategy is package contains the functionality for tuning and training.

The project will run from the train.py which will pass the data to agents and give results.