Capstone Project

Martin Piják

Bratislava, Slovakia

Abstract

This is project for using Machine Learning methods in futures trading.

Keywords: Machine Learning, Trading, Udacity, Nanodegree

1. Domain

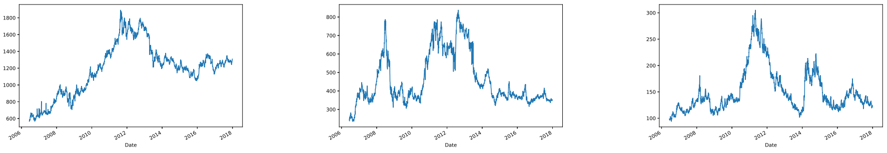
This project investigates possible futures trading strategies on Chicago Mercantile Exchange (CME) and Intercontinental Exchange (ICE) markets with Machine Learning methods. The goal is to find the trading strategy mostly based on the price, Commitment of Traders report (COT) and seasonality pattern. We will compare this trading strategy to commonly used investing approaches as returns of Nasdaq.

Following commodities were investigated:

- Gold
- Corn
- Coffee

2. Datasets and Inputs

I used data from Quandl. Data contains Open, High, Low, Close and volume (OHLCV) and commitment of traders (COT). Continuous data was generated by taking contract with the highest volume for the trading day. For details please see **data-preparation.ipynb**.



(a) Gold (b) Corn (c) Coffee

Figure 1: Close price graphs

3. ML Tools

Following tools were used:

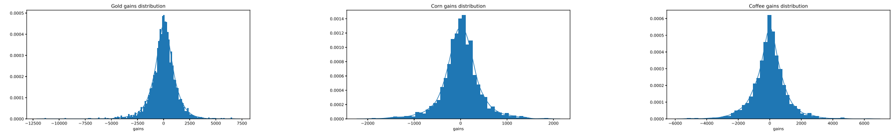
- General ML tools
  - Scikit
- Visualisation
  - matplotlib
  - seaborn
- Model building tools
  - Keras
  - TensorFlow (keras backend)
  - H2O
  - LightGBM

4. Data analysis

In the proposal I wanted to investigate classification of volatility. Below is 95% of daily volatility. 95% was selected as a simulation of slippage. General trading idea is to keep trade for a day.

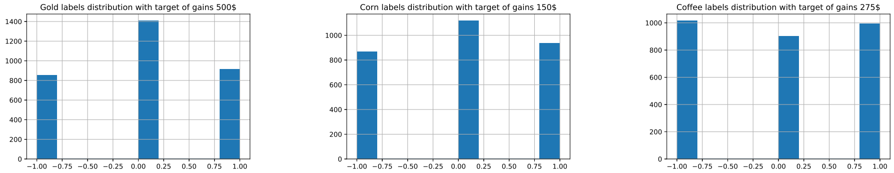
Labels were generated based on the trading target. Target -1 is for short trade. 0 for no trade and 1 for long trade.

I decided to have a look at regressor as well.



(a) Gold (b) Corn (c) Coffee

Figure 2: Gains distribution



(a) Gold (b) Corn (c) Coffee

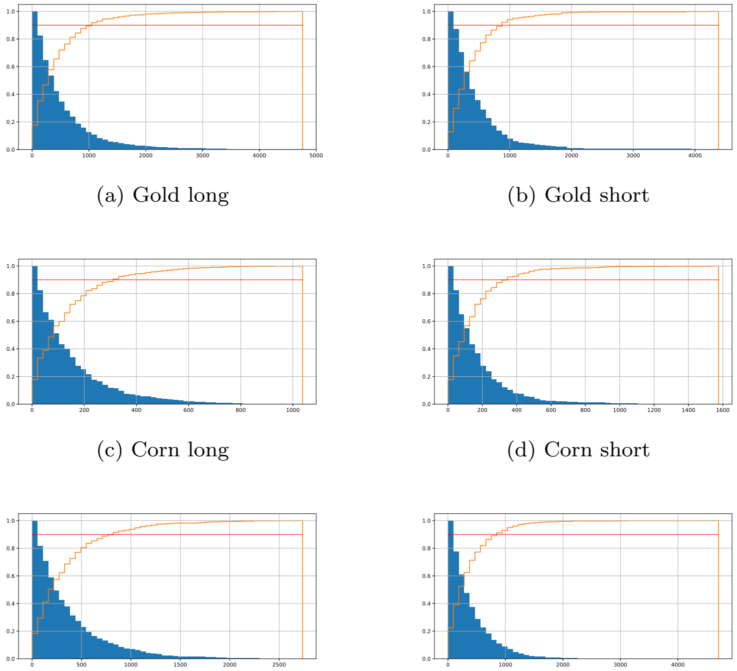
Figure 3: Labels distribution

Based on the past 2 years of trading data (OHLCV and COT) classifier is deciding whether to trade or not.

Approach to training classifier/regressor and evaluation does not take into account that stop-loss can still make trade unsuccessfull. From Machine Learning perspective the best approach is to have result of classification regression as close to the desired outcome as possible.

Stop-loss was selected so that 90% of trades will be successfully executed (exited on close).

Commodity	Long Stop Loss	Short Stop Loss
Gold	1000	800
Corn	300	300
Coffee	800	800



(a) Gold long

(b) Gold short

(c) Corn long

(d) Corn short

(e) Coffee long

(f) Coffee short

Figure 4: Stop Loss analysis counter position move

5. Feature Engineering

We transformed last $2 * 252$ trading days into vector with 2520 scalars.

In order to capture cyclicality I have transformed features as trading day of month, day of week and quarter of year into *sin* and *cos* values. I am not sure whether this transformation grants subsequent PCA usage. After adding data features vector now contains 2528 scalars.

We transformed COT to reflect extreme in the index. COT 1 of industrials corresponds to the maximum of traders positions throughout the last 2 years. Respectively 0 to minimum. We have included COT for commercial and industrial users from the last 8 weeks which is 16 values. Training vector now contains $2528 + 16 = 2544$ values.

5.1. PCA

We try to capture about 80% of variance of the data. For corn this corresponds to about 160 components. For gold it is about 180 components. Case of coffee is very strange.

It is interesting Corn is best explained by PCA transformation. It is probably due to clear seasonal patterns in trading.

I am surprised that gold is better explained by PCA transformation than Coffee. I would expect that coffee has stronger seasonal trading patterns than gold because of the growth cycle. Maybe gold mining is subject to the weather in similar way as agricultural commodities. Gold is still mostly recycled and new production has limited impact on total amount of traded gold.

Possible explanations:

- corn traded on CME is mostly US produced with stable harvest season
- production of coffee is very unpredictable depending on the conditions of a given year
- there are multiple producers around the world (coffee is more of a global market with limited US production) with different harvest periods Coffee Harvest
- important difference between coffee and corn is price per unit corn is much less efficient to transport
  - 1 kg of corn is worth about 15
  - 1 kg of coffee is worth about 230

Based on the PCA variance graph I think that PCA transformation is not suitable for coffee. Information in components is growing linearly. If we don't see sharp increase of cumulative explained variance with few first components, then PCA transformation is not suitable. Therefore, coffee should not be considered for trading with PCA transformation. I will continue with coffee as well but based on this transformation I would not go ahead with trading unless I would find different transformation.

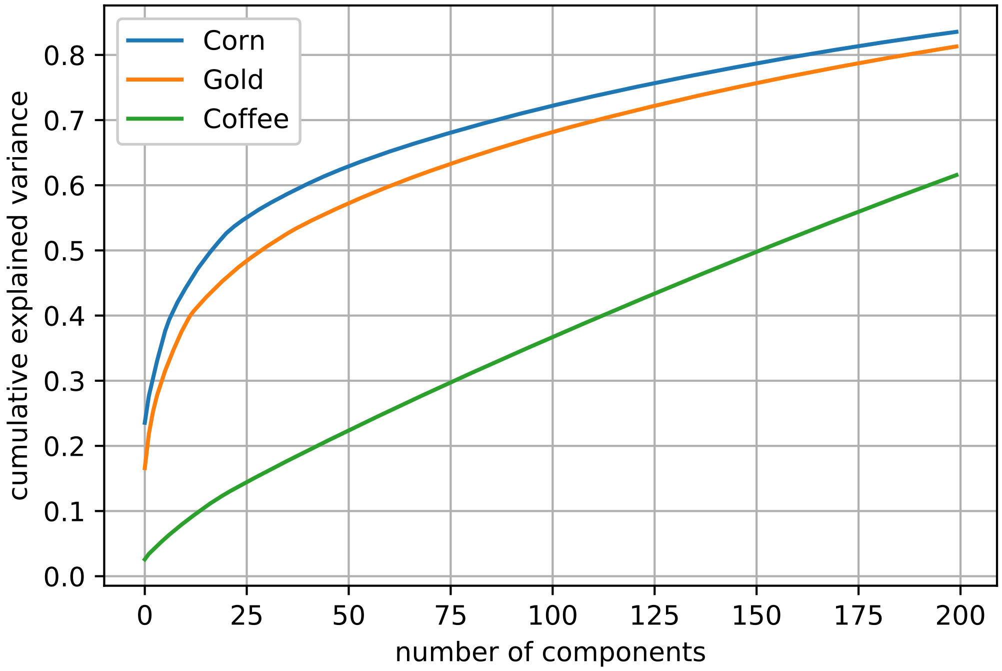
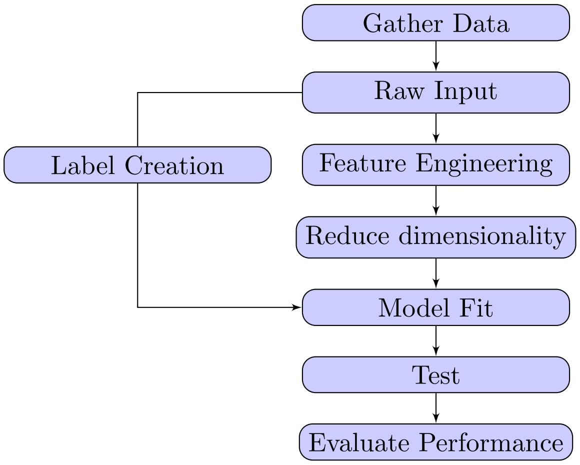


Figure 5: PCA explained variance

5.2. Project Design

Following schema is showing high level project pipeline.



I will look into commodities that's why I want to capture seasonality. I plan to use daily data, because of that my trading plan is to hold the position (either long/short) for 1–3 days or until stop loss was hit.

I will investigate different features that have impact on trading.

- feature engineering
  - PCA (dimensionality of input vector is very high)
  - COT
  - price
  - volume
  - date (modify to find seasonality)
- training
  - Random Forest
  - Neural Network
  - XGBoost
  - LightGBM
  - ...

How will be stop-loss selected?

- fixed
  - try different fixed stop-loss values
- floating
  - based on the volatility of previous days

5.3. Raw Input

Raw input will have approximately following shape:

$$\text{2 years data}$$

$$2 * 252 \text{ trading days} = 504 \text{ trading days}$$

$$504 \text{ trading days} * 7 \text{ (number of features in a day)} = 3528 \text{ features}$$

5.4. Feature Engineering

Reduce dimensionality:

- PCA
- accumulation
  - trading day of month
  - average
  - try different min/max of last trading day in relationship to 2years of data
  - ...

5.5. Label Generation

Labels will be generated as,

$$fee = 1.5 \tag{1}$$

$$t_{treshold} = 30 \tag{2}$$

$$v_{volatility} = (\text{close} - \text{open}) * \delta \text{ where } \delta \approx 0.95 \tag{3}$$

$$labels \begin{cases} |v_{volatility}| > t_{treshold} + fee \implies \begin{cases} v_{volatility} > 0 \implies 1 \text{ (long)} \\ v_{volatility} < 0 \implies -1 \text{ (short)} \end{cases} \\ |v_{volatility}| < t_{treshold} + fee \implies 0 \text{ (no trade)} \end{cases} \tag{4}$$

δ constant is used for simulating slippage. 1.5 that we are subtracting is a trading fee. We can see in the label function if volatility is too close to 0 it will be rounded to 0 — meaning don't trade.

Output of model will be classification of input into three categories trade (long/short position) or do not trade.

If we will look into longer trading than one day. Labels can be constructed in the similar way as above. But we will have to consider stop-loss value. We might have to end the trade sooner than we wanted.

6. Benchmark Model

I will use three benchmarks for my model

- fixed percentage (bank/bond deposit comparison)
- Dow Jones Industrial Average performance
- mean reversal trading on given commodity markets

7. Evaluation of trading

Compare trading strategy to Nasdaq performance and mean reversal strategy.

8. Conclusion

Trading is a difficult ML problem. Out of three compared commodities gold, corn and coffee we were able to predict performance with gold. Other commodities behaved randomly with approximately 0 correlation to the actual volatility.

In the beginning of project, I was thinking of a classifier (short, no trade, long) because it is closer to the usage of model. I tested regressor as well. Regressor works better because there is more information. I tried different loss function when training regressor. I decided to use weighted MSE. This could be further modified for better function omitting errors below threshold.

Gold trading is the most capital intensive with very big stop losses (1000 long, 800 short). This can be problem for trading with 10000$ account.

In case of corn and coffee the data was almost impossible to classify. I suspect more data transformation is needed to get better results.