International Journal of xxxxxx
Vol. x, No. x, (20xx), pp. xx-xx

Using Machine Learning to develop Short term Trend Trading Strategies

Ricky Sambo Macharm

WorldQuant University
ricky.macharm@gmail.com

Abstract

This paper presents a research on a trading strategy on some selected best performing FOREX currency pairs after considering realistic trading scenarios and analyzing the performance on out of sample data. Machine Learning models like Random Forest, eXtreme Gradient Boosting (XGBoost), and Deep learning methods (using fast.ai's tabular data) were tested. Technical indicators derived from the traditional ones like moving averages, Average True Range (ATR) and pandas date were used to generate features for training. The best performing algorithm is then deployed as a webapp using streamlit.

Keywords: Machine Learning, Deep Learning, Traditional Indicators, algorithm, webapp, FOREX, XGBoost, features, labelling

1. Introduction - Problem Definition

Python has become the de factor programming language for algorithmic trading and different platforms (quantopian, numerai, quantconnect) have risen to cater for traders who want to create their own signal generating algorithms. Unfortunately, some traders do not seem to have the skills nor the aptitude to learn how to code in Python so they can use some of these platforms.

This project aims at leveraging on the power of Python for algorithmic trading to create a standalone signal generating web app traders can access to use for their trading purposes without knowing how to write a single line of Python code.

2. Literature Review

2.1. Introduction

A market is said to be trending if you can look at the chart and see which direction it seems to be headed (up or down). Experienced traders have long observed that markets do not trend all the time - sometimes a market seems to be stuck in a range with no clear direction.

Charles Dow co-founded The Wall Street Journal in 1889 primarily to cover business and financial news[1]. He began to publish a series of periodicals on his views of the markets using two market averages he had earlier developed called the Dow Jones Industrial Average (DJIA) and the Dow Jones Transportation Average (DJTA)

The Dow theory can be summarized thus: when the market is trending upwards and if one of the averages (DJIA or DJTA) breaches a previous important high then it is expected that the other will follow suit within a reasonable period of time. Dow theory also explains the movement of the market as consisting of a top, a bottom, an uptrend, and a downtrend. His principles are the basis of what we now call Technical Analysis.

After his death, a lot of people tried to organize and collectively represent the Dow theory based on Dow's published editorials. Many other theorists and speculators later also

ISSN: XXXX-XXXX IJXX

proposed theories to explain their observations of the behaviour of the markets. Notable of which are the Wyckoff Methodology[2] and Elliot Waves[3] just to name a few.
In the early 1980's, commodity traders Richard Dennis and William Eckhard decided to test to see if traders were born or they were made. They recruited and trained 23 individuals (calling them "Turtles") by teaching them for two weeks and then tested with real money. The trading strategy they used was a trend following technique that used new four-week highs as an entry signal for longs.
Since the complete rules that made up the Turtle strategy have been made public, it has and still is possible to backtest them and see the performance on recent market activity. From some of the backtests it has been reported that the CAGR which was about 216% between 1970 to 1986 when Dennis and Eckhardt were developing the system, and eventually when his Turtle Traders learnt and traded the system to 10.5% between 1986 and 2009[4].

The market is constantly changing from a period of movement to a period of rest and back to a period of movement. This interchange between the phases of motion and rest is constantly taking place[5]. Understanding this simple explanation can aid us in developing trading strategies that can be used to exploit the forex markets. Machine learning techniques can be mobilized to identify movements in the market for profits that are sustainable.

2.2. Technical Analysis

The use of past price data to guide a trader's decision today on how to engage the market (either to go long, short, exit a position or to stand aside) is known as Technical Analysis[6]. In this work, I will use technical analysis to generate my signals. I will start by generating buy and sell signals from some technical indicators out some indicators.

2.2.1. Technical Indicators

We will list some traditional indicators that have been used in stock and commodities trading for some time. The list is endless and some of them have seen a fair use in trading the forex market.

- Trend Indicators: Moving Averages are the most deployed trend indicators on a technical analysts chart. They are usually used to smooth out the price of an asset over a specified length of time by calculating and displaying a constantly updated mean value of the closing price. Simple Moving Averages use Arithmetic means while the Exponential moving (EMA) is a weighted average that is meant to give greater importance to recent price data which tends to it more responsive to new information.
- Momentum Indicators: These help the technical analyst to get a better understanding of the rate of change of price. Some commonly used momentum indicators include: Moving Average Convergence Divergence (or the MACD which is really derived from the difference between two moving averages), the Relative Strength Index (RSI) and the Average Directional Index (ADX) which is derived from the smoothed averages of the -DI and +DI, which are themselves derived from the comparison of two consecutive lows and their respective highs.

2.2.2. Trend Trading

Trend trading is a style of trading where technical analyst attempts to make profit on the price movement of a particular asset in the market in a particular direction. When the price of the asset is moving up indicated by the price bars on the chart making consistent higher-highs and higher-lows, the trend is said to be up. Conversely, if the price bars on the chart are making consistent lower-lows and lower highs then the trend is said to be down.

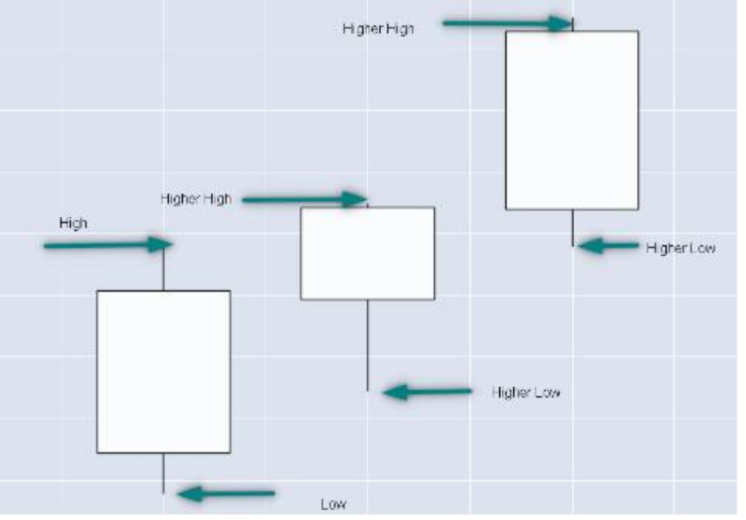

Figure 1. Candles making higher highs and higher lows.

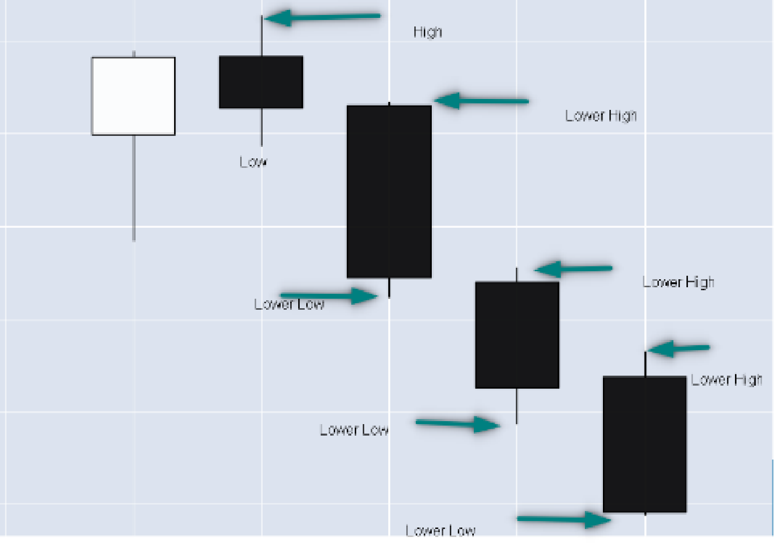

Figure 2. Candles making lower lows and lower highs

One common way of generating signals for trend trading using the moving average is to consider a long trade when a fast moving average crosses above a slow one and for shorts we wait for the fast moving average to cross down. Another signal generating technique is for the closing price of the asset be above or below the moving average in question. The choice of technique is at the discretion of the trader. We will use a similar technique as the latter, however using less traditional indicators most of which have seen some great success in the forex market.

Figure 3. Simple moving average cross over system

2.2.3. Machine Learning

Since Arthur Samuel popularized the term Machine Learning in the late 1950's the field has seen tremendous progress in recent years. The financial industry especially algorithmic traders around the world also try to benefit from these technological advances[7]. Arthur Samuel's idea was instead of programming a computer step by step on how to solve a particular problem, it is shown examples of problems and their solutions thus allowing it to figure it out by itself using statistical and Machine Learning techniques[8]. Like Statistical techniques, Machine Learning techniques also seeks to map the elements of a particular domain (in our case features we have selected from our dataset) with those of another domain (the direction of the market up or down).
Wang *et al.* [9] used volume as a feature to help in forecasting the direction of S&P 500 and DJI of which they found no strong correlation. Using a Machine Learning technique called support vector machine (SVM) model on stock prediction,[10] made a comparison between multi-layer perceptron (MLP) and SVM. They showed that in many cases, SVM seemed to have outperformed MLP on the datasets they used. [11] combined machine learning and deep learning techniques that have been utilised by prior research approaches, to get new training features for prediction.

2.3. Competitor Analysis

With the rise of machine learning and the interest that trading equities, CFDs, Stocks and the foreign exchange markets, so many researchers and retail traders are interested in merging the two to find out how they could beat the market. Platforms like Quantopian, QuantConnect and NumerAI (to mention a few) have risen to fill that gap. Despite the success of these platforms in bringing great traders and technicians together, a large niche of non-techie traders are not involved as learning Python, C# or R stands as a barrier for many of them.

This project aims on concentrating on such traders. When fully launched, the traders would be heavily assisted from receiving signals while they spend time on trading using whatever money management approach they prefer. Waiting by a computer screen for a trade setup can cause traders to start imagining chart patterns or to be affected by paralysis of analysis.

2.3.1. Strengths

- This project aims to be designed to provide signals generated from machine learning algorithms that have been found to better than chance (>50%) to non-tech traders via a simple webapp.
- It will encourage traders to concentrate more on entries and exit making them less prone to paralysis of analysis.

2.3.2. Weakness

- This project will not include money management techniques as that would be left to the discretion of the trader interacting with it.

2.3.3. Opportunities

- This will bring in more players from the trading community into the machine learning by lowering the barriers to entry.
- Since it will be an open source project, it would also bring in other machine learning enthusiast to contribute there by increasing the popularity.

2.3.4. Threats

- There are big corporations with deep pockets who are interested in this domain.
- Some trading platforms already have signal generation via dashboards.

3. Methodology

This project uses python (and libraries like Pandas, Numpy, Matplotlib, Scikit Learn, FastAI and Pytorch) to import the data, visualize it, generate features, and create a Machine Learning algorithm for our webapp. Streamlit will be used to host the webapp for the public to access.

3.1. Data Collection

We will use the Foreign Exchange (FOREX) data collected from the Oanda API[12]. The best instruments to trade are the ones that are trending (moving in the same direction) most strongly over a given period. In sideways environments, it is usually best to watch out for those which have shown the highest volatility over recent days. Since we will be working on short term trending strategies, we would use our system to select those that show great performances during our test period to display on out webapp.

We intend to use the daily timeframe when accessing the data from the broker. This will not be a day trading webapp. The signals would be generated once a day so the trader can spend the rest of his time on something else.

We will choose to use for our algorithm currency pairs selected from the collection of currencies shown in the bale below.

Table 1. Table of currencies and % of Daily Trades April 2019

Rank	Currency	ISO 4217 code(symbol)	% of daily trades(bought or sold)(April 2019)
1	United States dollar	USD (US$)	88.3%
2	Euro	EUR (€)	32.3%
3	Japanese yen	JPY (¥)	16.8%
4	Pound sterling	GBP (£)	12.8%
5	Australian dollar	AUD (A$)	6.8%
6	Canadian dollar	CAD (C$)	5.0%
7	Swiss franc	CHF (CHF)	5.0%
8	Renminbi	CNY (元 / ¥)	4.3%
9	Hong Kong dollar	HKD (HK$)	3.5%
10	New Zealand dollar	NZD (NZ$)	2.1%

Figure 4. The 10 most traded currencies in 2019 [13], [14]

We will not use the Chinese Renminbi and the Hong Kong dollar because of the wide spreads and the fact that the brokers that offer any currency pair involving them are rare. The pairing of the remaining 8 will form a basket of 28 currency pairs that we can choose from based on performance.

3.2. Feature Generation

Once we have set the API from the broker, we use the pandas package to load up the data. We now begin to use some traditional technical analysis indicators to generate features that tell us to go long, short or to stand aside. We will explain some of them following.

3.2.1. Daily Log Returns.

We will use the daily log returns to generate features and shift that by one day. The log returns were chosen over the simple returns for the reason that they are non-desicrete, continuous and time additive. We want to see if the previous N day's price could help us to predict the price of the following day in question.

$$r_t = ln\frac{y_t}{y_{t-1}} \tag{1}$$

Where,

r_t = returns

y_t = todays close price

y_{t-1} = yesterday's close price

3.2.2. Technical Indicators

We will select a few non-traditional technical indicators for some the features we would generate. These technical indicators have found favor in the eyes of many forex traders who claim they perform better than the traditional ones even though many of them were derived from the so-called old school traditional indicators.

- TTM Squeeze: John F. Carter introduced the TTM Squeeze indicator in his book Mastering the Trade [15]. It is really a combination of several traditional Indicators. In this work we will not use it as it was presented in the original work. Our indicator will be a combination of the Bollinger Bands, the Keltner Channel and the ADX. We will want the Bollinger Bands to be found within the Keltner Channel and the ADX to be below the value of 20. The moment that the Bollinger bands leave the Keltner Channel then we have a signal.

- Supertrend: This is a trend following indicator that was developed by Olivier Seban and has been moving around the FOREX trading forums for a while. The indicator is based off the Average True Range and it is usually overlaid on the price chart; it has been used as a trailing stop indicator as well as used to determine the direction of the trend. When the close price of the instrument is above the indicator we say the market is in an uptrend and when close price is below we say that it is in a downtrend.
- SSL Channel Indicator: The Semaphore Signal Level channel Indicator is a price chart overlay indicator that is made from a pair of moving averages that have equal periods: one of the highs of the price and the other of the lows. When the close price of the instrument is above the indicator, we say the market is in an uptrend and when close price is below, we say that it is in a downtrend.
- Lag Log Returns: We will calculate the log returns of the closing prices of our data and then arrange the previous log returns of the previous days as features to be fed into our machine learning in our attempt to predict the next day's position up or down, making it a classification problem. The lag returns could be made categorical or simply left as continuous variables.

3.2.3. Handling Dates

Sometimes dates handle market participation which can affect how the prices move for a particular instrument. Training our model by letting it know what month or day of the weeks or if it's a holiday or the end of the quarter could greatly help. Part of the features would be to add these as features for the model to train with. This concept is part of the fastai methodology and is part of the course offered online [16].

3.3. Classification using Machine Learning

Since our intention is to predict if the market is going up or down, it will seem appropriate to tackle this as a binary classification problem. A simple classifier is some rule, or preferably a function, that maps to a domain x a class label $\tilde{y}$:

$$\tilde{y} = f(x) \tag{2}$$

The predicted class is expected to be the one with the highest probability of occurrence, mathematically represented as:

$$\tilde{y} = arg\, max_y Pr(Y = y|X) \tag{3}$$

We will choose from a few supervised learning models provided by the library scikit-learn which runs on Python. After the initial training we will apply meta-labelling as introduced by Marcos de Prado in [17].

3.3.1. Random Forest

A common supervised learning approach for tackling regression and classification in known a Decision Tree. In this approach, the goal is predicting a target by using decision rules shaped from the features available in the dataset in question. A large number of Decision trees makes a Random forest which generally averages the predictions of the trees.

3.3.2. XGBoost

eXtreme Gradient Boosting (or XGBoost) also uses Decision Trees just like the Random Forest model, however the difference here is the fact that weaker learners from the decision trees are boosted. This is believed to help to reduce bias and it is based on the a question posed by Kearns and Valiant in [18]: “Can a set of weak learners create s single strong learner…?”

3.3.3. LightGBM

LightGBM (LGBM), like XGBoost, is an open-source gradient boosting library for GPU computing used for regression and classification problems. In LGBM, we can enable bagging alongside boosting, and speed up training using GPUs.

4. Project Work Plan and Deliverables

The table below displays the work plan this entire capstone project will follow:

Table 2. Work Plan

Using Machine Learning to develop Short term Trend Trading Strategies											
Task	Deliverable	Duration									
		Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10
Project	Project Proposal										
Analysis	Literature Review and Competitor Analysis										
Project Design	Project proposal and Methodology										
Peer Review	Refined Project Proposal based on Feedback										
Project	Draft Project Report										
Implementation and Presentation	Final Project Report and Presentation of Findings										

5. Results

The steps taken to carry out the development of the web app and the results obtained are outlined and explained below. After the data has been collected and the features generated as explained, the resulting data frame is divided into a training set, a validation set and a test set.

```
[ ]  df_train = dataset.iloc[:-1000,:]
     train = df_train.copy()
     df_test = dataset.iloc[-1000:,:]
     test = df_test.copy()
```

```
[ ]  cond = df_train.Year<2015
```

```
[ ]  train_idx = np.where(cond)[0]
     valid_idx = np.where(~cond)[0]
```

```
[ ]  splits = (list(train_idx),list(valid_idx))
```

5.1. Training and Testing

The data collected are from the 28 FOREX pairs gotten from the 8 major currencies already discussed. The modelling was done on the training, the validation and the test set on all the choses 28 pairs before the results are displayed. I will sample out some of the pairs instead of displaying for all of them.

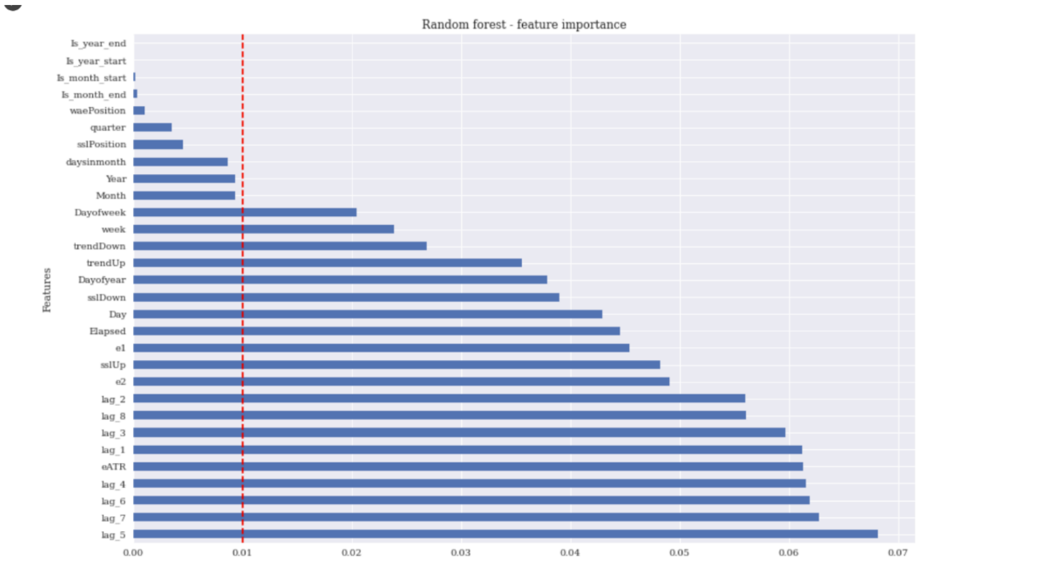

Fig. 5 Feature importance

Using the Random Forest feature importance function, we were able to view the chart above and using the red dashed line shown (0.01) we get rid of all the features that have a value less than that indicated by the red dashed line.

```
[ ]  cluster_columns(xs_imp)
```

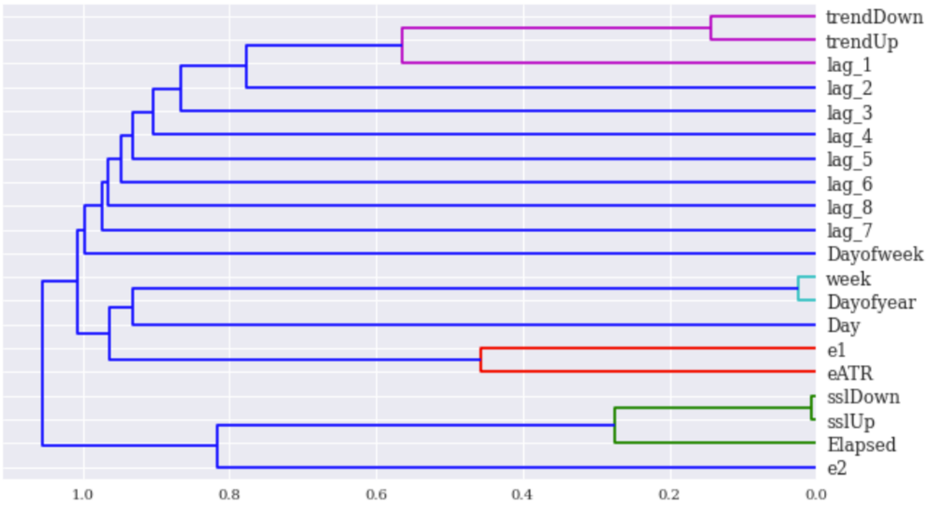

Fig. 6 Cluster columns

The cluster columns is to see if some features are best not used together as they are repeating what the other is doing.

```
[ ]  dataset[['ret', 'strategy']].cumsum().apply(np.exp).plot(
                                                      figsize=(10, 6))
```

<matplotlib.axes._subplots.AxesSubplot at 0x7f575e2a37d0>

Fig 7 chart of strategy performance (out-of-sample) against buy and hold

The strategy chart (fig. 7) shows us how it performed on the during the test period

```
[ ]  valid_pred = m.predict(valid_xs_final)
     valid_acc = accuracy_score(valid_y, valid_pred)
     print(f'Validation Set Accuracy: {valid_acc}')
     print(confusion_matrix(valid_y, valid_pred))
     print(classification_report(valid_y, valid_pred))
```

```
Validation Set Accuracy: 0.5127768313458262
[[188  88]
 [198 113]]
              precision    recall  f1-score   support

        -1.0       0.49      0.68      0.57       276
         1.0       0.56      0.36      0.44       311

    accuracy                           0.51       587
   macro avg       0.52      0.52      0.50       587
weighted avg       0.53      0.51      0.50       587
```

Fig. 8 Validation set accuracy score

```
[ ]  test_pred = m.predict(test_imp_final)
     test_acc = accuracy_score(test['dir'], test_pred)
     print(f'Test Set Accuracy: {test_acc}')
     print(confusion_matrix(test['dir'], test_pred))
     print(classification_report(test['dir'], test_pred))
```

```
Test Set Accuracy: 0.497
[[284   0 214]
 [  0   0   1]
 [288   0 213]]
              precision    recall  f1-score   support

        -1.0       0.50      0.57      0.53       498
         0.0       0.00      0.00      0.00         1
         1.0       0.50      0.43      0.46       501

    accuracy                           0.50      1000
   macro avg       0.33      0.33      0.33      1000
weighted avg       0.50      0.50      0.49      1000
```

Fig. 9 Test set accuracy score

What we are showing here is the performance of one of the currency pairs from the basket of 28 currencies pairs after training. The validation and test sets were kept aside and used only when training was completed. This is out-of-sample data. By using two sets, we are checking the robustness on the currency pair in question.
Fig 10 shows the simple tear sheet for the said currency pair gotten from the pyfolio package.

```
create_simple_tear_sheet(df_test['ret']*test_pred)
```

Start date	2017-04-03
End date	2021-03-29
Total months	47
	Backtest
Annual return	1.7%
Cumulative returns	6.9%
Annual volatility	7.2%
Sharpe ratio	0.27
Calmar ratio	0.08
Stability	0.05
Max drawdown	-22.1%
Omega ratio	1.05
Sortino ratio	0.38
Skew	-0.16
Kurtosis	3.69
Tail ratio	1.01
Daily value at risk	-0.9%

Fig. 10 Tear sheet

lgb_results_valid

statistic	Annual return	Cumulative returns	Annual volatility	Sharpe ratio	Calmar ratio	Stability	Max drawdown	Omega ratio	Sortino ratio	Skew	Kurtosis	Tail ratio	Daily value at risk
AUDCAD	-0.007652	-0.030023	0.084188	-0.049147	-0.044033	0.055104	-0.173787	0.991904	-0.067420	-0.277135	0.903384	0.934395	-0.010623
AUDCHF	0.057857	0.250064	0.135737	0.484730	0.276138	0.171244	-0.209522	1.100410	0.609551	-5.789234	103.515171	1.030809	-0.016840
AUDJPY	-0.011053	-0.043147	0.126372	-0.024699	-0.045043	0.625443	-0.245385	0.995808	-0.033742	-0.347970	1.967797	0.992771	-0.015934
AUDNZD	0.017935	0.073086	0.078118	0.266613	0.122905	0.034389	-0.145923	1.045830	0.368994	-0.276315	1.713710	0.910833	-0.009759
AUDUSD	-0.040273	-0.150512	0.104632	-0.340536	-0.161875	0.318498	-0.248793	0.946291	-0.467353	-0.115465	0.272634	0.922973	-0.013324
CAD	-0.047595	-0.175941	0.077192	-0.593138	-0.162038	0.639339	-0.293730	0.906412	-0.832555	0.172331	1.018124	0.922640	-0.009907
CADCHF	0.101835	0.469364	0.126366	0.833915	0.494112	0.706058	-0.206097	1.196763	1.029297	-8.485876	173.853118	1.245941	-0.015502
CADJPY	0.103744	0.479491	0.117161	0.901051	1.072539	0.856564	-0.096727	1.169018	1.394877	0.423612	2.824462	1.121590	-0.014342
CHF	-0.075232	-0.266823	0.120493	-0.585524	-0.268221	0.868056	-0.280486	0.877835	-0.689684	-9.421944	202.430997	0.893654	-0.015461
CHFJPY	-0.077827	-0.274953	0.131982	-0.543672	-0.205769	0.451679	-0.378227	0.883049	-0.639442	-9.832691	217.255473	0.955025	-0.016913
EURAUD	-0.031213	-0.118240	0.106238	-0.245335	-0.127223	0.007692	-0.245341	0.959706	-0.334298	-0.230208	1.434289	0.856468	-0.013488
EURCAD	0.014625	0.059309	0.094323	0.201076	0.101498	0.004612	-0.144095	1.035088	0.283466	-0.123976	2.425824	1.001208	-0.011808
EURCHF	-0.058886	-0.214030	0.101562	-0.542000	-0.257047	0.766706	-0.229085	0.826445	-0.592847	-19.097697	520.052004	1.030204	-0.013014
EURGBP	0.000792	0.003148	0.089602	0.053445	0.004602	0.017663	-0.172192	1.009501	0.079875	0.944189	9.847852	1.090414	-0.011270
EURJPY	0.063327	0.275914	0.105340	0.635457	0.518530	0.682874	-0.122129	1.120226	0.958825	0.578746	6.920226	1.067085	-0.013006
EURNZD	-0.045087	-0.167295	0.108203	-0.372197	-0.156360	0.271495	-0.288355	0.939016	-0.507674	-0.320703	3.026101	1.045691	-0.013792
EURUSD	-0.097234	-0.333636	0.090431	-1.085696	-0.243819	0.902658	-0.398795	0.827565	-1.436449	-0.201738	3.187922	0.869466	-0.011783
GBPAUD	-0.072134	-0.257028	0.108617	-0.635060	-0.210324	0.454548	-0.342968	0.898522	-0.878959	0.617748	7.280856	0.852778	-0.013958
GBPCAD	0.028047	0.116016	0.093667	0.341894	0.113018	0.002050	-0.248162	1.061141	0.525159	1.218395	12.496699	1.028446	-0.011674
GBPCHF	-0.127839	-0.418871	0.126133	-1.017527	-0.275367	0.848000	-0.464249	0.791472	-1.153242	-9.399425	184.510685	0.867010	-0.016401
GBPJPY	0.058056	0.250999	0.128021	0.505505	0.226177	0.783079	-0.256684	1.100501	0.685898	-1.884345	26.589422	1.098946	-0.015872
GBPNZD	-0.014846	-0.057626	0.114986	-0.072439	-0.048710	0.211627	-0.304779	0.987688	-0.098857	-0.663299	6.314240	1.040519	-0.014520
GBPUSD	0.065360	0.285620	0.091873	0.734700	0.365428	0.627798	-0.178859	1.145583	1.205662	2.388036	30.014565	1.140100	-0.011307
JPY	0.082074	0.367541	0.104159	0.809399	0.572701	0.508679	-0.143310	1.152245	1.198333	0.155990	3.206651	1.019248	-0.012788
NZDCAD	-0.022790	-0.087421	0.095378	-0.194023	-0.094970	0.002092	-0.239967	0.967827	-0.269346	-0.111106	1.622263	0.978615	-0.012090
NZDCHF	0.064394	0.281000	0.135976	0.525072	0.477472	0.850505	-0.134864	1.107106	0.913159	5.251619	88.783287	1.083142	-0.016848
NZDJPY	-0.000941	-0.003729	0.129828	0.057552	-0.004094	0.004061	-0.229862	1.009920	0.081799	0.155253	2.696330	0.981058	-0.016327
NZDUSD	0.047910	0.204066	0.112808	0.471207	0.342941	0.377820	-0.139702	1.079222	0.683391	0.047105	0.419974	1.068199	-0.014002

Fig. 11 performance result for 28 pairs

Finally, we have the performance results for all the 28 pairs under consideration.

5.2 Deployment online

After carrying out the tests above, the best models are converted into pickle files and saved for deployment on the internet for traders who have access to the link. Different services were tried but it seems like Heroku meets our needs for this instance.

deploy_v2 · Streamlit

Not secure | 564469258683.ngrok.io

Today's Signals: 2021-04-14

	probability	signal	probability	signal
AUDCAD	0.6149	Stand Aside	0.5104	Stand Aside
AUDCHF	0.7555	Go Long	0.8110	Go Long
AUDJPY	0.5104	Stand Aside	0.5569	Stand Aside
AUDNZD	0.8761	Go Short	0.8973	Go Short
AUDUSD	0.5430	Stand Aside	0.5645	Stand Aside
CAD	0.7332	Go Short	0.7885	Go Short
CADCHF	0.6313	Stand Aside	0.5487	Stand Aside
CADJPY	0.5453	Stand Aside	0.5513	Stand Aside
CHF	0.8642	Go Short	0.7588	Go Short
CHFJPY	0.4996	Stand Aside	0.6392	Stand Aside
EURAUD	0.7651	Go Short	0.8575	Go Short

Fig. 12 Dashboard deployed for possible trades for the date shown

```
df
```

pair	probability	signal	probability	signal
AUDCAD	0.614944	Stand Aside	0.510403	Stand Aside
AUDCHF	0.755539	Go Long	0.811047	Go Long
AUDJPY	0.510395	Stand Aside	0.556935	Stand Aside
AUDNZD	0.876118	Go Short	0.897291	Go Short
AUDUSD	0.543039	Stand Aside	0.564512	Stand Aside
CAD	0.733243	Go Short	0.788532	Go Short
CADCHF	0.631281	Stand Aside	0.548682	Stand Aside
CADJPY	0.545323	Stand Aside	0.551307	Stand Aside
CHF	0.864171	Go Short	0.758795	Go Short
CHFJPY	0.499585	Stand Aside	0.639214	Stand Aside
EURAUD	0.765052	Go Short	0.857505	Go Short
EURCAD	0.549323	Stand Aside	0.526042	Stand Aside
EURCHF	0.541320	Stand Aside	0.640929	Stand Aside
EURGBP	0.549590	Stand Aside	0.637477	Stand Aside
EURJPY	0.552767	Stand Aside	0.617067	Stand Aside
EURNZD	0.600196	Stand Aside	0.525762	Stand Aside
EURUSD	0.714377	Go Long	0.737431	Go Long
GBPAUD	0.711568	Go Short	0.539645	Stand Aside
GBPCAD	0.575008	Stand Aside	0.620227	Stand Aside
GBPCHF	0.552319	Stand Aside	0.523653	Stand Aside
GBPJPY	0.511230	Stand Aside	0.596263	Stand Aside
GBPNZD	0.768763	Go Short	0.826644	Go Short
GBPUSD	0.611479	Stand Aside	0.650093	Stand Aside
JPY	0.670280	Stand Aside	0.647922	Stand Aside
NZDCAD	0.551724	Stand Aside	0.541771	Stand Aside
NZDCHF	0.778707	Go Long	0.944802	Go Long
NZDJPY	0.636873	Stand Aside	0.688373	Stand Aside
NZDUSD	0.579917	Stand Aside	0.515035	Stand Aside

Fig. 13 Dashboard of all currency pairs to be deployed for traders

For us to prepare online deployment, we use streamlit, an open-source app framework used for creating apps for Machine Learning and Data Science projects in pure Python. The Dataframe dashboard and shown created in fig. 13 is passed to the streamlit application before being deployed on Heroku.

6

6. Conclusion and further work

Various machine learning models were trained on some chosen technical indicators and some pandas datetime as independent variables or features. The whole problem was a classification problem where we needed to predict if the market would go up or down or to stand aside.

The models successfully identified with relatively good, and consistent performance in validation and test set (out-of-sample) were converted into pickle files and used on the deployed webapp.

Further work will be done on this to make it even more robust:

- More optimizations would be done on a pair-by-pair basis in order to get models for each pair instead of generalizing the models and choosing the best performing pairs.
- Currency pair analysis would be added in the decision-making processes of selecting which pairs to trade irrespective of the signals generated by the models.
- Money management practices would be incorporated to make it a complete trading system
- Paid hosting would be considered and a more attractive UI (User Interface) included.

Disclaimer

This paper was created as part of a capstone project for WorldQuant University degree program in MSc in Financial Engineering and it is for educational purposes only. It does not constitute investment advice. You can lose some or all of your money trading stocks, CFDs, Cryptocurrencies and Forex.

7. References

[1] "Dow theory." [Online]. Available: https://en.wikipedia.org/wiki/Dow_theory.

[2] R. D. Wyckoff and T. R. D, *Wyckoff method of trading in stocks: a course of instruction in tape reading and active trading*. New York: Wyckoff Associates, 1932.

[3] A. J. Frost and R. R. Prechter, *Elliott wave principle: key to market behavior*. Chichester: Wiley, 2004.

[4] J. Liberty, "Were the Turtles just lucky?...: Au.Tra.Sy blog - automated trading system." 2010, [Online]. Available: http://www.automated-trading-system.com/turtles-just-lucky/.

[5] T. Crabel, *Day trading with short term price patterns and opening range breakout*. Greenville, SC: Traders Press, 1990.

[6] C. J. Neely and P. A. Weller, "Technical analysis in the foreign exchange market," *Fed. Reserv. Bank St. Louis Rev.*, 2011.

[7] Y. J. Hilpisch, *Artificial intelligence in finance: a Python-based guide*. Sebastopol, CA: O'Reilly Media, 2021.

[8] J. Howard and S. Gugger, *Deep learning for coders with fastai and PyTorch: AI applications without a PhD*. Sebastopol, CA: O'Reilly Media, Inc, 2020.

[9] X. Wang, P. Kang, H. Phua, and W. Lin, "Stock market prediction using neural networks: Does trading volume help in short-term prediction?," 2003.

[10] H. Ince and T. B. Trafalis, "Short term forecasting with support vector machines and application to stock price prediction," *Int. J. Gen. Syst.*, vol. 37, no. 6, pp. 677–687, 2008.

[11] G. Liu and X. Wang, "A new metric for individual stock trend prediction," *Eng. Appl. Artif. Intell.*, vol. 82, pp. 1–12, 2019.

[12] Oanda, "Which API should I use?" Oanda Corporation, [Online]. Available: https://developer.oanda.com/rest-live-v20/introduction/.

[13] Wikimedia, “Most traded currencies by value,” *Wikipedia*. 2019, [Online]. Available: https://en.wikipedia.org/wiki/Template:Most_traded_currencies#cite_note-1.
[14] BIS, “Triennial Central Bank Survey of Foreign Exchange and Over-the-counter (OTC) Derivatives Markets in 2019,” *Triennial Central Bank Survey*. Bank of International Settlements, 2019, [Online]. Available: https://www.bis.org/statistics/rpfx19_fx.pdf.
[15] J. F. Carter, *Mastering the trade proven techniques for profiting from intraday and swing trading setups*, 3rd ed. New York: NYC.
[16] J. Howard, “Practical Deep Learning for Coders.” [Online]. Available: https://course.fast.ai/.
[17] P. M. L. de, *Advances in financial machine learning*. New Jersey: Wiley, 2018.
[18] M. Kearns and L. Valiant, “Crytographic limitations on learning Boolean formulae and finite automata,” *Symp. Theory Comput.*, pp. 433–444, 1989.

Code Reference

This project adopts collaborative GitHub environment and the source code for the project is available at:

https://github.com/African-Quant/WQU_MScFE_Capstone_Grp9