Project Diamond Hands

Supervised learning techniques to forecast Bitcoin price returns

Contributors:

Lee Copeland

Vishnu Kurella

Ahmad Sadraei

Ling Zhou

Goal and Recipe

Goal: Design an algorithmic trading strategy using supervised learning techniques to decide on the optimal position for the next 1hr return of Bitcoin

Inputs: Utilize various observable market data points, including prices and technical indicators for crypto and non-crypto assets

Data

Curation and pre-processing:

- Download and slice time series data by hourly interval (Messari and Genesis Volatility APIs)
- Process and clean data set (merge, interpolate for market closures)
- Scale Data and then split data into train and test groups
- Build, train and compare multiple models to determine optimal BTC position for the next hour

Data

Independent variables: All time series data intervals are hourly

Spot Price % Change:

- Bitcoin (BTC)
- Ethereum (ETH)
- DXY (Bloomberg USD Index)
- S&P 500 futures
- Gold futures
- 10yr UST futures

Technical Indicators:

- Volume
- BTC/ETH Ratio
- 50 hour + 200 hour moving averages

Option Metrics:

- BTC and ETH at-the-money implied volatility
- BTC and ETH 25-delta implied skew

Technologies

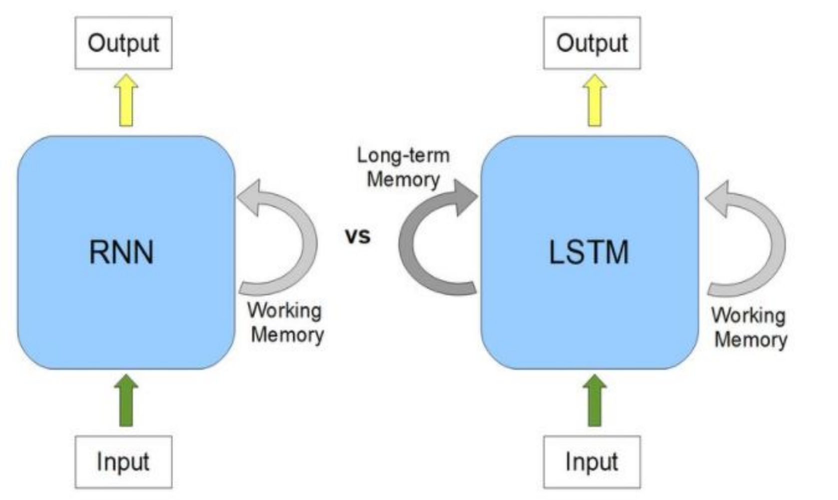
- Tensorflow
- Sklearn
- Google Colab
- APIs (Messari and Genesis)
- Numpy
- Pandas
- Quantitative Easing

Model

- Supervised learning can consider multiple factors and known past outcomes to make predictions about future outcomes
- We chose a classification model over a regression model in order to focus on the decision of having a position
- There are 5 classification fields: large short, small short, no position, small long; large long
- Recurrent Neural Networks (RNN) use output from previous intervals to inform the current interval
- **We chose a bidirectional LSTM recurrent neural network for time series forecasting in TensorFlow**

Model

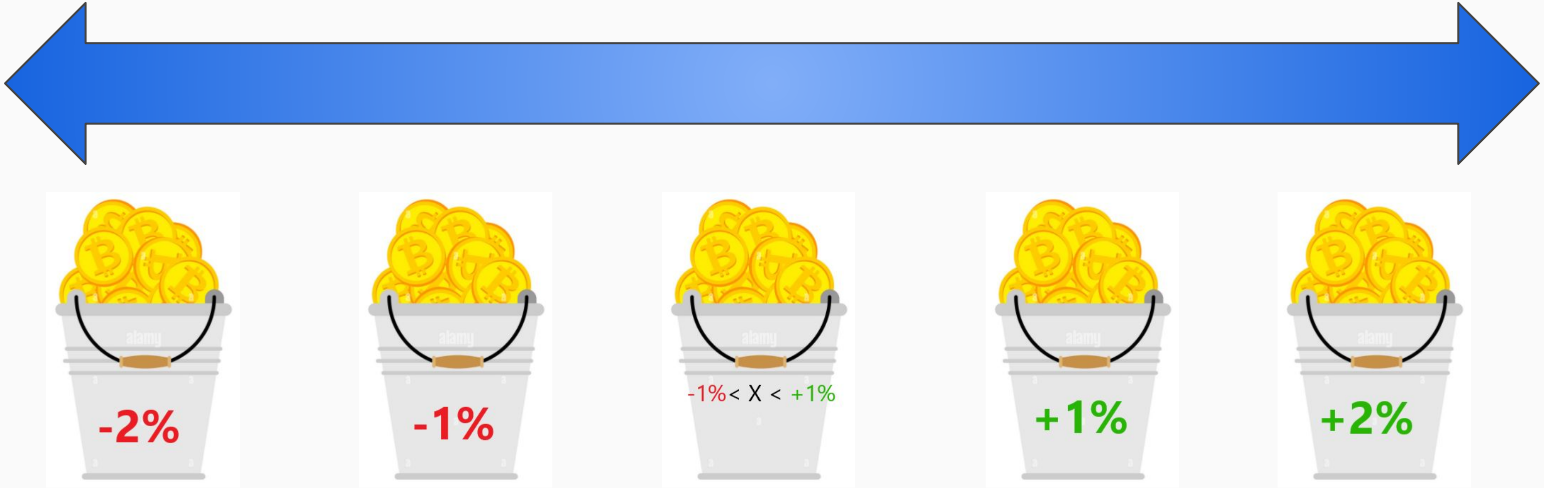
- LSTM: “Long Short-Term Memory” is a type of RNN model
- Vanilla RNN models use their operating state memory (short-term)
- LSTM is a variant of RNN that uses operating state memory from historical intervals (long-term) to inform the present interval decision



Source: *LSTM Based Automatic Plant Root Anatomization Systems*, Oxford University

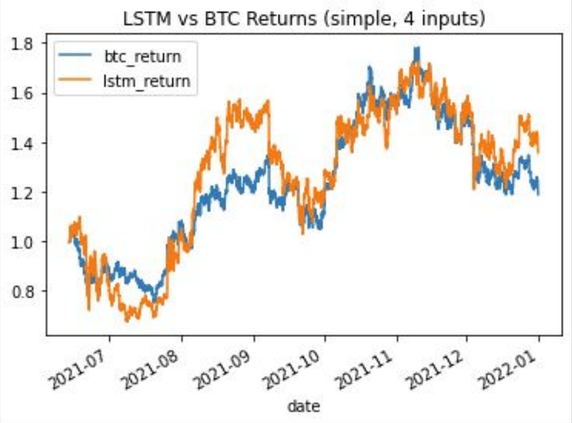
Multi-label Classification

- Trading strategy seeks to scale position size based on signal strength



Model 1

```
BTC Annualized Return: 51.696043%
BTC Annualized Standard Deviation: 66.393471%
BTC Sharpe Ratio: 0.7786314196077307
LSTM Annualized Return: 93.351096%
LSTM Annualized Standard Deviation: 90.159028%
LSTM Sharpe Ratio: 1.035404866614756
```



The basic iteration of our model with just 4 inputs:

- btc_close_pct
- btc_volume
- eth_close_pct
- eth_volume

Buy 71% of time, buy 2x 19% of time, sell 1x 10% of the time

```
-2  -1  0  1  2
0   0   0  1  0     3452
           0  1      908
    1   0  0  0      474
```

Model 1

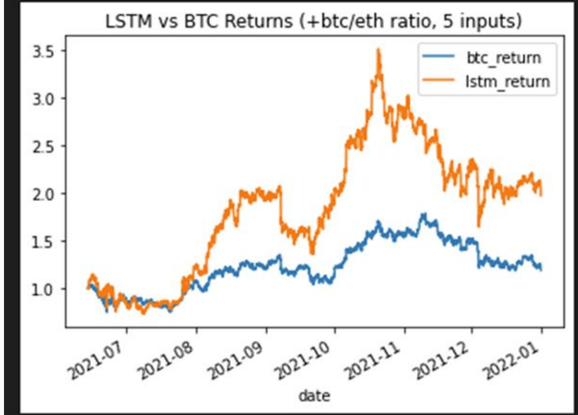
```
Model: "sequential_15"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 lstm_45 (LSTM)              (None, 4, 4)              96

 dropout_45 (Dropout)        (None, 4, 4)              0

 lstm_46 (LSTM)              (None, 4, 4)              144

 dropout_46 (Dropout)        (None, 4, 4)              0

 lstm_47 (LSTM)              (None, 4)                 144

 dropout_47 (Dropout)        (None, 4)                 0

 dense_15 (Dense)            (None, 5)                 25

=================================================================
Total params: 409
Trainable params: 409
Non-trainable params: 0
_________________________________________________________________
```

	precision	recall	f1-score	support
0	0.00	0.00	0.00	851
1	0.24	0.10	0.14	1172
2	0.00	0.00	0.00	754
3	0.27	0.77	0.39	1197
4	0.21	0.22	0.21	860
micro avg	0.25	0.25	0.25	4834
macro avg	0.14	0.22	0.15	4834
weighted avg	0.16	0.25	0.17	4834
samples avg	0.25	0.25	0.25	4834

- 409 Parameters
- 0.14 average precision

Model 2

```
BTC Annualized Return: 51.696043%
BTC Annualized Standard Deviation: 66.393471%
BTC Sharpe Ratio: 0.7786314196077307
LSTM Annualized Return: 169.179319%
LSTM Annualized Standard Deviation: 101.676085%
LSTM Sharpe Ratio: 1.6639047281429602
```



Added the %change in BTC/ETH ratio:

- btc_close_pct
- btc_volume
- eth_close_pct
- eth_volume
- btc_eth_ratio

This model was much more “aggressive”: Bought 2x 27% of the time, Sold 2x 2% of the time

Better returns + higher standard deviation, net result in higher sharpe ratio

Model 2

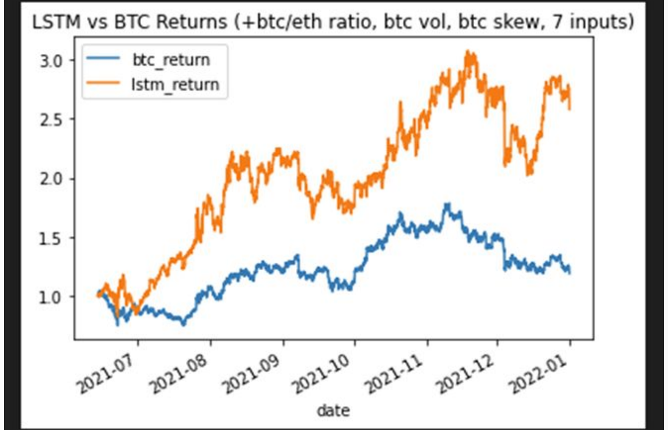
```
Model: "sequential_16"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 lstm_48 (LSTM)              (None, 5, 5)              140

 dropout_48 (Dropout)        (None, 5, 5)              0

 lstm_49 (LSTM)              (None, 5, 5)              220

 dropout_49 (Dropout)        (None, 5, 5)              0

 lstm_50 (LSTM)              (None, 5)                 220

 dropout_50 (Dropout)        (None, 5)                 0

 dense_16 (Dense)            (None, 5)                 30

=================================================================
Total params: 610
Trainable params: 610
Non-trainable params: 0
_________________________________________________________________
```

	precision	recall	f1-score	support
0	0.21	0.02	0.04	851
1	0.30	0.08	0.12	1172
2	0.00	0.00	0.00	754
3	0.26	0.69	0.38	1197
4	0.24	0.36	0.29	860
micro avg	0.26	0.26	0.26	4834
macro avg	0.20	0.23	0.17	4834
weighted avg	0.22	0.26	0.18	4834
samples avg	0.26	0.26	0.26	4834

- 610 Parameters
- 0.20 average precision

Model 3

```
BTC Annualized Return: 51.696043%
BTC Annualized Standard Deviation: 66.393471%
BTC Sharpe Ratio: 2.250914047071839
LSTM Annualized Return: 205.407788%
LSTM Annualized Standard Deviation: 91.255278%
LSTM Sharpe Ratio: 2.250914047071839
```



Added the %change in BTC/ETH ratio and BTC vol + skew:

- btc_close_pct
- btc_volume
- eth_close_pct
- eth_volume
- btc_eth_ratio_pct
- btc_twentyFiveDelta30DayExp
- btc_atm30

This model was much more “aggressive”: Bought 1x only 54% of time, Bought 2x 28% of time, Sold 1x 18% of time

Better returns + same standard deviation, net result in higher sharpe ratio

Model 3

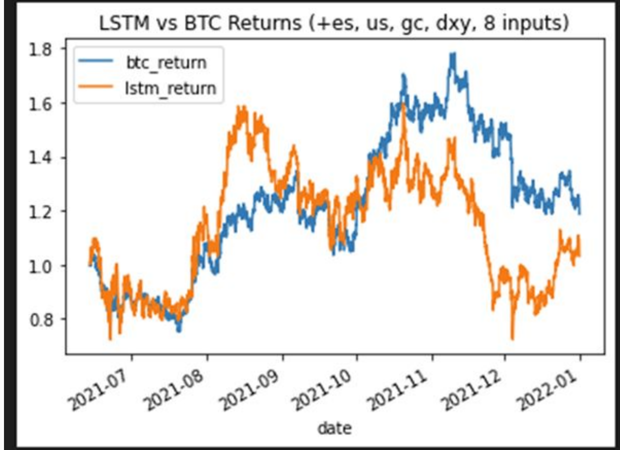
```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 lstm (LSTM)                 (None, 7, 7)              252

 dropout (Dropout)           (None, 7, 7)              0

 lstm_1 (LSTM)               (None, 7, 7)              420

 dropout_1 (Dropout)         (None, 7, 7)              0

 lstm_2 (LSTM)               (None, 7)                 420

 dropout_2 (Dropout)         (None, 7)                 0

 dense (Dense)               (None, 5)                 40

=================================================================
Total params: 1,132
Trainable params: 1,132
Non-trainable params: 0
_________________________________________________________________
```

	precision	recall	f1-score	support
0	0.00	0.00	0.00	851
1	0.26	0.30	0.27	1172
2	0.00	0.00	0.00	754
3	0.27	0.59	0.37	1197
4	0.26	0.26	0.26	860
micro avg	0.26	0.26	0.26	4834
macro avg	0.16	0.23	0.18	4834
weighted avg	0.17	0.26	0.20	4834
samples avg	0.26	0.26	0.26	4834

- 1132 Parameters
- 0.16 average precision

Model 4

```
BTC Annualized Return: 51.696043%
BTC Annualized Standard Deviation: 66.393471%
BTC Sharpe Ratio: 0.7786314196077307
LSTM Annualized Return: 55.021245%
LSTM Annualized Standard Deviation: 99.722304%
LSTM Sharpe Ratio: 0.5517446236967012
```



Uses S&P, US 10Y, Gold, and USD futures instead of additional BTC/ETH data:

- btc_close_pct
- btc_volume
- eth_close_pct
- eth_volume
- dxy_close_pct
- es_close_pct
- gc_close_pct
- us_close_pct

This model was much more “aggressive”: Bought 1x only 47% of time, Bought 2x 32% of time, Sold 1x 21% of time

Worse returns, higher standard deviation, worse sharpe ratio

Model 4

```
Model: "sequential_7"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 lstm_21 (LSTM)              (None, 8, 8)              320

 dropout_21 (Dropout)        (None, 8, 8)              0

 lstm_22 (LSTM)              (None, 8, 8)              544

 dropout_22 (Dropout)        (None, 8, 8)              0

 lstm_23 (LSTM)              (None, 8)                 544

 dropout_23 (Dropout)        (None, 8)                 0

 dense_7 (Dense)             (None, 5)                 45

=================================================================
Total params: 1,453
Trainable params: 1,453
Non-trainable params: 0
_________________________________________________________________
```

	precision	recall	f1-score	support
0	0.00	0.00	0.00	851
1	0.23	0.20	0.22	1172
2	0.00	0.00	0.00	754
3	0.27	0.52	0.36	1197
4	0.20	0.36	0.26	860
micro avg	0.24	0.24	0.24	4834
macro avg	0.14	0.22	0.17	4834
weighted avg	0.16	0.24	0.19	4834
samples avg	0.24	0.24	0.24	4834

- 1453 Parameters
- 0.14 average precision

Pitfalls

- We started with too much data and that actually diluted the signal
  - Reducing the number of features our model took actually yielded better results
  - Needing to make sure the number of features ended up around %10 of parameters
- Spent majority of time running the model using various features sub sets to find the optimal ones to use
- Our model yielded different result on each run (given same input)
  - The random initialization of the weight for the model makes the outcome of the model non-deterministic
  - Could use random seed to counter the issue

Follow Ups

- Utilize higher precision data, such as every five minutes instead of hourly
- Adjust the parameters for the LSTM model for Models 2 and 3 to identify a model with better precision
- Model for an output position in ETH or an alt-coin instead of bitcoin because the other coins may be more efficient
- Transaction Costs

Questions