1.ipnyb

pred variable : log\_ret

sigma 1

4 hidden layers with 1024 , 512 , 256 , 128 neurons

cost function : tf.reduce\_mean(tf.squared\_difference(out, Y))

Adam optimizer to reduce mse

batch size 256

epochs 100

split : 80 20

**MSE final : 0.135828**

**Ground truth in blue , predicted in red**

**A close up of a device

Description generated with high confidence**

1.ipnyb

pred variable : log\_ret , considered from Aug 2016 ( when prices started to increase , therefore chances of log return being ~0 are low

sigma 1

4 hidden layers with 1024 , 512 , 256 , 128 neurons

cost function : tf.reduce\_mean(tf.squared\_difference(out, Y))

Adam optimizer to reduce mse

batch size 256

epochs 100

split : 80 20

**MSE final : 0.0122917**

**A close up of a device

Description generated with high confidence**

2.ipnyb

pred variable : log return , considered from Aug 2016 ( when prices started to increase , therefore chances of log return being ~0 are low

RNN with 2 LSTM layers with dropouts 0.4 and 0.5

loss function : mse

optimizer : rmsprop , metrics : accuracy

window size 20 for LSTM , batch size 768

30 epochs

70 10 10

**"Train Score: 0.95 MSE (0.97 RMSE)**

**Test Score: 0.14 MSE (0.38 RMSE)"**

**A close up of a device

Description generated with high confidence**

**1.ipnyb**

**pred variable : log\_ret , considered from Aug 2016 ( when prices started to increase , therefore chances of log return being ~0 are low**

**sigma 1**

**4 hidden layers with 1024 , 512 , 256 , 128 neurons**

**cost function : tf.reduce\_mean(tf.squared\_difference(out, Y))**

**Adam optimizer to reduce mse**

**batch size 128**

**epochs 100**

**split : 80 20**

**MSE final : 0.0105706**

**A close up of a colorful background

Description generated with high confidence**

2.ipnyb

pred variable : log return

RNN with 2 LSTM layers with dropouts 0.4 and 0.5

loss function : mse

optimizer : rmsprop , metrics : accuracy

window size 20 for LSTM , batch size 512

100 epochs

70 10 10

**Train Score: 0.79 MSE (0.89 , Test Score: 0.14 MSE (0.38 RMSE)**

**A close up of a device

Description generated with high confidence**

**1.ipnyb**

**pred variable : log\_ret , considered from Aug 2016 ( when prices started to increase , therefore chances of log return being ~0 are low**

**sigma 1**

**4 hidden layers with 2048 , 1024 , 512 , 256 neurons**

**cost function : tf.reduce\_mean(tf.squared\_difference(out, Y))**

**Adam optimizer to reduce mse**

**batch size 128**

**epochs 100**

**split : 80 20**

**MSE final for test : 0.0135728**

**A picture containing object

Description generated with very high confidence**

**2.ipnyb**

**pred variable : log return**

**RNN with 2 LSTM layers with dropouts 0.4 and 0.5**

**loss function : mse**

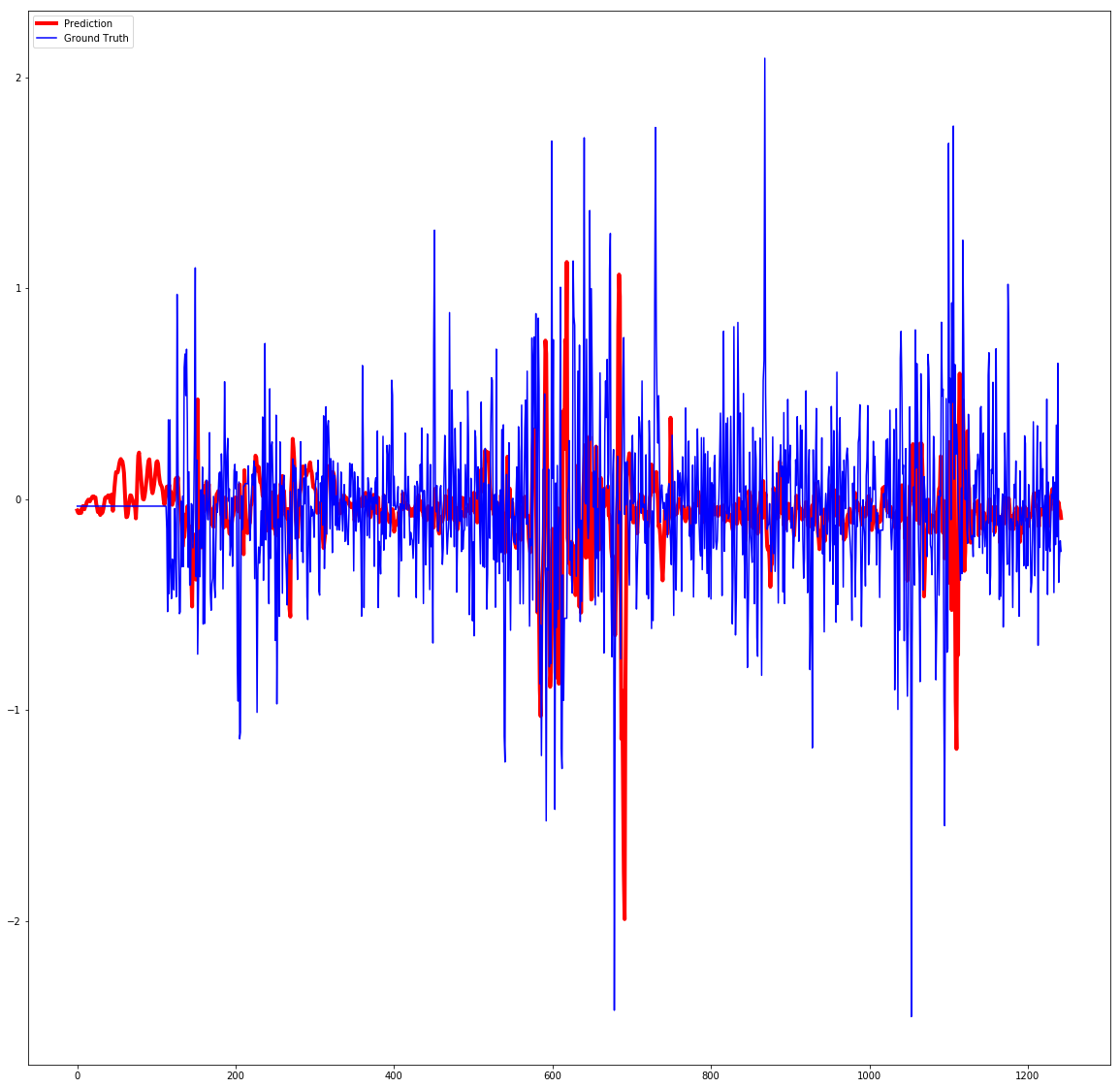
**optimizer : rmsprop , metrics : accuracy**

**window size 20 for LSTM , batch size 512**

**500 epochs**

**70 10 10**

**Train Score: 0.27 MSE (0.52 RMSE) Test Score: 0.18 MSE (0.43 RMSE)**

****

**1.ipnyb**

**pred variable : log\_ret , considered from Aug 2016 ( when prices started to increase , therefore chances of log return being ~0 are low**

**sigma 1**

**4 hidden layers with 512 , 256 , 128, 64 neurons**

**cost function : tf.reduce\_mean(tf.squared\_difference(out, Y))**

**Adam optimizer to reduce mse**

**batch size 128**

**epochs 100**

**split : 80 20**

**MSE final for test : 0.0126264**

**A close up of a colorful background

Description generated with high confidence**

**1.ipnyb**

**pred variable : log\_ret , considered from Aug 2016 ( when prices started to increase , therefore chances of log return being ~0 are low**

**sigma 1**

**4 hidden layers with 512 , 256 , 128, 64 neurons**

**cost function : tf.reduce\_mean(tf.squared\_difference(out, Y)) , with weight initializer mode set to fan\_out**

**Adam optimizer to reduce mse**

**batch size 128**

**epochs 100**

**split : 80 20**

**MSE final for test :0.0610254**

**A close up of a device

Description generated with high confidence**

**1.ipnyb**

**pred variable : log\_ret , considered from Aug 2016 ( when prices started to increase , therefore chances of log return being ~0 are low**

**sigma 1**

**4 hidden layers with 512 , 256 , 128, 64 neurons**

**cost function : tf.reduce\_mean(tf.squared\_difference(out, Y)) , with weight initializer mode set to fan\_in**

**Adam optimizer to reduce mse**

**batch size 128**

**epochs 100**

**split : 80 20**

**MSE final for test : 0.0352366**

**A close up of a device

Description generated with high confidence**

**2.ipnyb**

**pred variable : log return**

**RNN with 2 LSTM layers with dropouts 0.3 and 0.1**

**loss function : mse**

**optimizer : rmsprop , metrics : accuracy**

**window size 20 for LSTM , batch size 512**

**500 epochs**

**70 10 10**

**Train Score: 0.20 MSE (0.45 RMSE) Test Score: 0.18 MSE (0.42 RMSE)**

**A close up of a device

Description generated with high confidence**

**1.ipnyb**

**pred variable : log\_ret , considered from Aug 2016 ( when prices started to increase , therefore chances of log return being ~0 are low**

**sigma 1**

**4 hidden layers with 512 , 256 , 128, 64 neurons**

**cost function : tf.reduce\_mean(tf.squared\_difference(out, Y)) , with weight initializer mode set to fan\_avg , distribution set to normal**

**Adam optimizer to reduce mse**

**batch size 128**

**epochs 500**

**split : 80 20**

**MSE final for test : 5.06339**

**A close up of a flower

Description generated with high confidence**

**2.ipnyb**

**pred variable : log return**

**RNN with 2 LSTM layers with dropouts 0.1 and 0.1**

**loss function : mse**

**optimizer : rmsprop , metrics : accuracy**

**window size 20 for LSTM , batch size 512**

**500 epochs**

**70 10 10**

**Train Score: 0.14 MSE (0.38 RMSE) Test Score: 0.23 MSE (0.48 RMSE)**

**A close up of a device

Description generated with high confidence**

**1.ipnyb**

**pred variable : log\_ret , considered from Aug 2016 ( when prices started to increase , therefore chances of log return being ~0 are low**

**sigma 1**

**4 hidden layers with 512 , 256 , 128, 64 neurons**

**cost function : tf.reduce\_mean(tf.squared\_difference(out, Y))**

**Adam optimizer to reduce mse**

**batch size 64**

**epochs 500**

**split : 80 20**

**MSE final for test : 0.130275**

**A close up of a flower

Description generated with high confidence**

**2.ipnyb**

**pred variable : log return**

**RNN with 2 LSTM layers with dropouts 0.1 and 0.1**

**loss function : mse**

**optimizer : rmsprop , metrics : accuracy**

**window size 20 for LSTM , batch size 256**

**500 epochs**

**70 10 10**

**Train Score: 0.10 MSE (0.31 RMSE) Test Score: 0.24 MSE (0.49 RMSE)**

**A close up of a device

Description generated with high confidence**

**1.ipnyb**

**pred variable : log\_ret , considered from Aug 2016 ( when prices started to increase , therefore chances of log return being ~0 are low**

**sigma 1**

**4 hidden layers with 512 , 256 , 128, 64 neurons**

**cost function : tf.reduce\_mean(tf.squared\_difference(out, Y))**

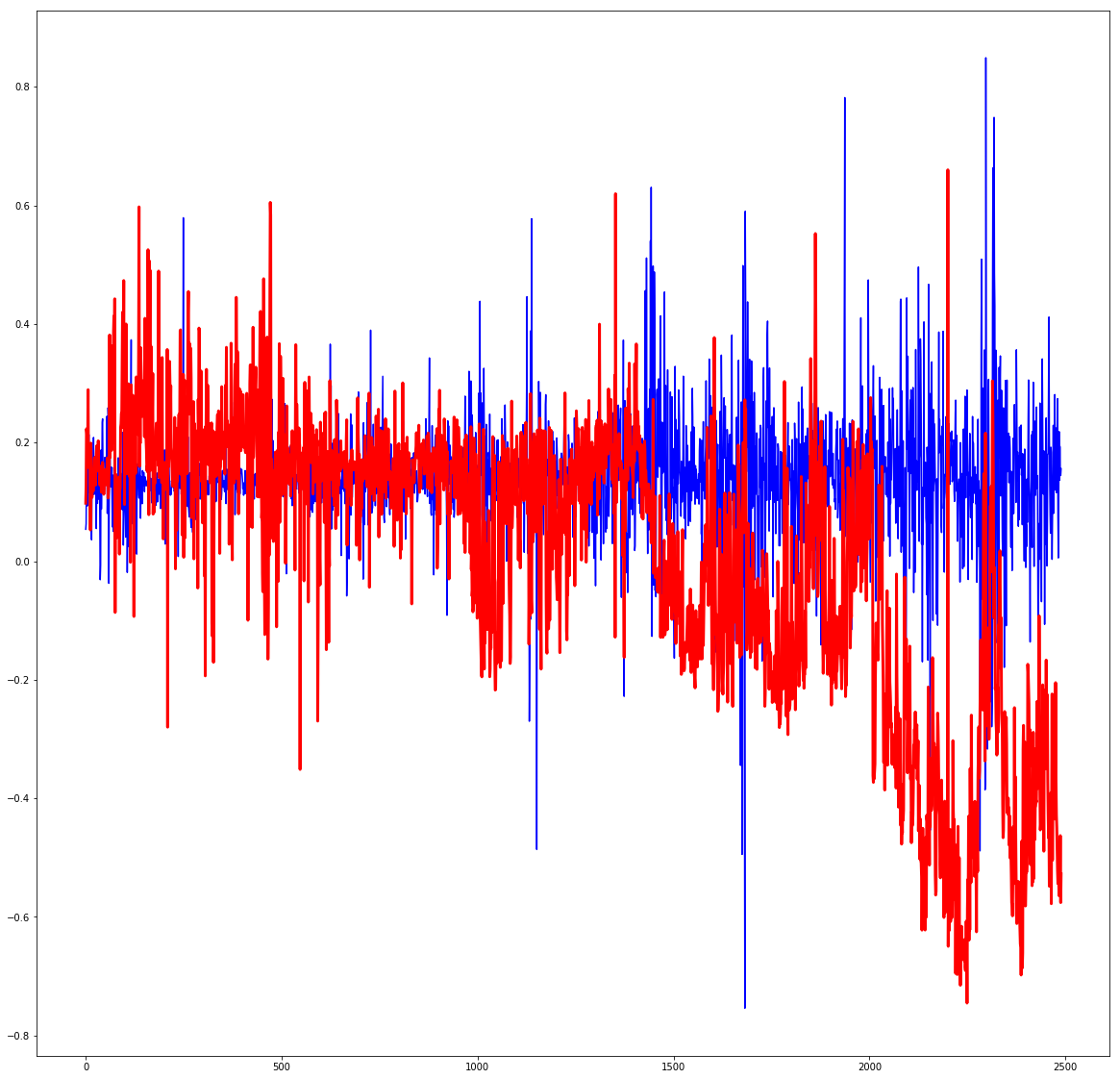
**Adam optimizer to reduce mse**

**batch size 512**

**epochs 500**

**split : 80 20**

**MSE final for test : 0.141722**

****

**1.ipnyb**

**pred variable : log\_ret , considered from Aug 2016 ( when prices started to increase , therefore chances of log return being ~0 are low**

**sigma 1**

**4 hidden layers with 256 , 128, 64 , 32 neurons**

**cost function : tf.reduce\_mean(tf.squared\_difference(out, Y))**

**Adam optimizer to reduce mse**

**batch size 64**

**epochs 100**

**split : 80 20**

**MSE final for test : 0.78542**

**A close up of a flower

Description generated with high confidence**

**2.ipnyb**

**pred variable : log return**

**RNN with 2 LSTM layers with dropouts 0.1 and 0.1**

**loss function : mse**

**optimizer : rmsprop , metrics : accuracy**

**window size 20 for LSTM , batch size 64 , sequence length + 5**

**500 epochs**

**70 10 10**

**Train Score: 0.06 MSE (0.25 RMSE) Test Score: 0.23 MSE (0.48 RMSE**

**A close up of a device

Description generated with high confidence**