

Prediction of Foreign Exchange Rates

using a variety of **Machine Learning Algorithms**



By.
Benita Rego
Meghshanth Sara

Overview

- **Foreign Exchange Rate Market** is the most valuable market when it comes to buying, selling and trading.
- The capacity to anticipate the foreign exchange rate is an useful talent.
- Forecasting the foreign exchange rate is a challenging time series issue.
- **Deep learning models** proven to be very efficient in the prediction of complex financial analytics problems.
- **Recurrent Neural Networks (RNNs)** - ANN has proven to be favorable for the provided **time series data**.



Problem Statement

Predicting the foreign exchange rates using various ML algorithms.

→ Recent currency data from <https://in.investing.com/currencies>

Data Scrapped for 01/2016 - 01/2023

→ Data Processing and Visualization

→ Training and testing various ML algorithms:

- ANN
- LSTM
- GRU

Machine Learning Algorithms **Tested:**

ANN

- Artificial neural networks (ANNs) used in supervised learning problems in which we know the target labels of the data.

LSTM

- Long Term Short Memory (LSTM) capable of picking up long-term dependencies, in issues involving sequence prediction.

GRU

- Gated Recurrent Unit (GRU) uses links through a series of nodes to carry out memory and clustering-related ML tasks.

Models:

ANN



```
from keras.layers import LSTM
K.clear_session()
model_lstm = Sequential()
model_lstm.add(LSTM(50, input_shape=(1, X_train.shape[1]),
                    activation='relu',
                    kernel_initializer='lecun_uniform',
                    return_sequences=False))
model_lstm.add(Dense(1))
model_lstm.summary()
```

Model: "sequential"

Layer (type)	Output Shape	Param #
=====		
lstm (LSTM)	(None, 50)	10400
dense (Dense)	(None, 1)	51
=====		
Total params: 10,451		
Trainable params: 10,451		
Non-trainable params: 0		

LSTM



```
K.clear_session()
model = Sequential()
model.add(Dense(12, input_dim=1, activation='relu'))
model.add(Dense(1))
model.summary()
```

Model: "sequential"

Layer (type)	Output Shape	Param #
=====		
dense (Dense)	(None, 12)	24
dense_1 (Dense)	(None, 1)	13
=====		
Total params: 37		
Trainable params: 37		
Non-trainable params: 0		

GRU



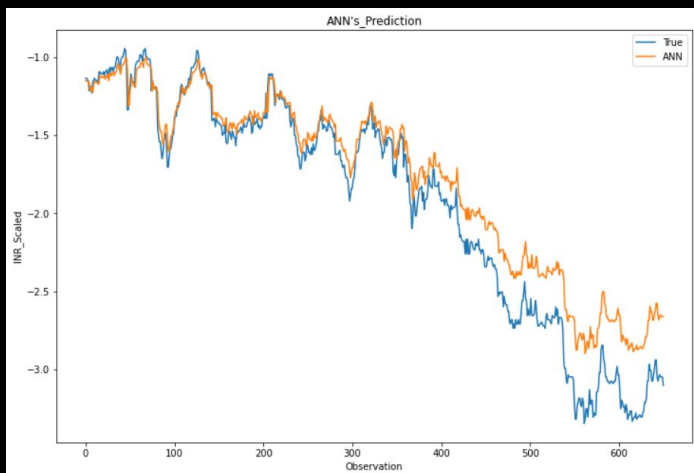
```
from keras.layers import GRU
K.clear_session()
model_gru = Sequential()
model_gru.add(GRU(7, input_shape=(1, X_train.shape[1]),
                  activation='linear',
                  kernel_initializer='lecun_uniform',
                  return_sequences=False))
model_gru.add(Dense(1))
model_gru.summary()
```

Model: "sequential"

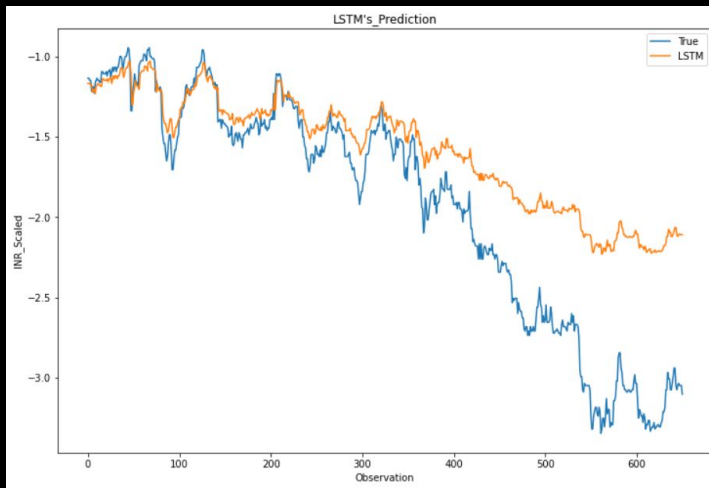
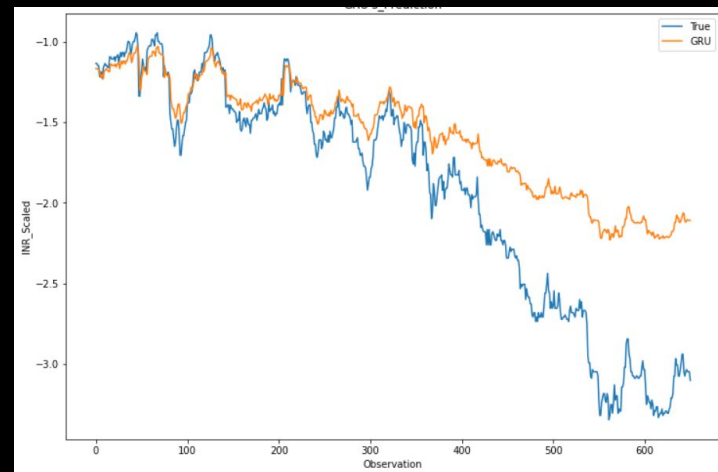
Layer (type)	Output Shape	Param #
=====		
gru (GRU)	(None, 7)	210
dense (Dense)	(None, 1)	8
=====		
Total params: 218		
Trainable params: 218		
Non-trainable params: 0		

Prediction Graphs:

ANN



GRU



LSTM