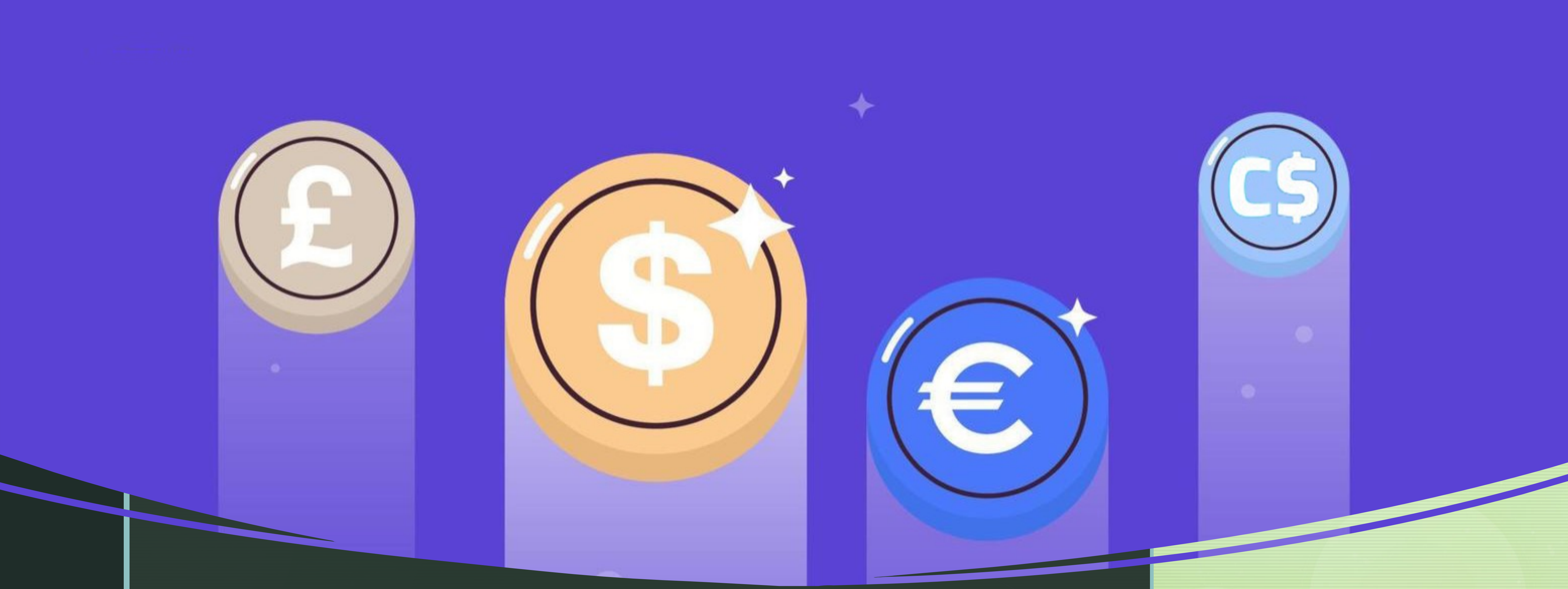


Forecasting Exchange Rates

Ly Le Thi
M.Sc. Econometrics

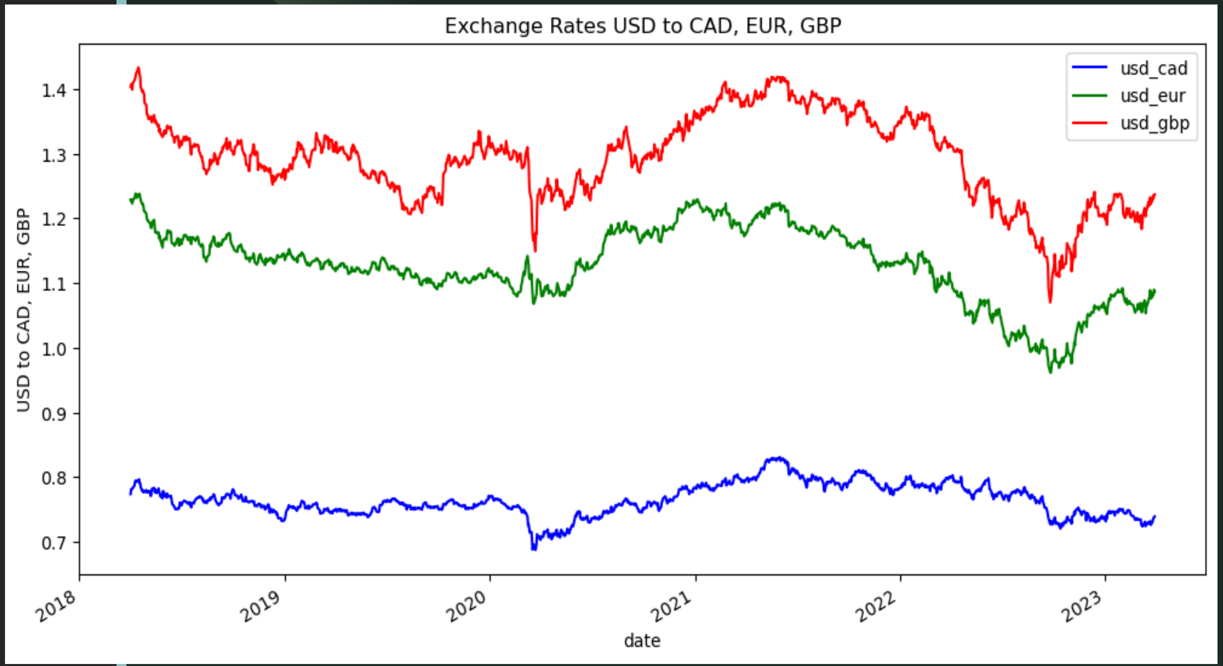
Contents

Introduction

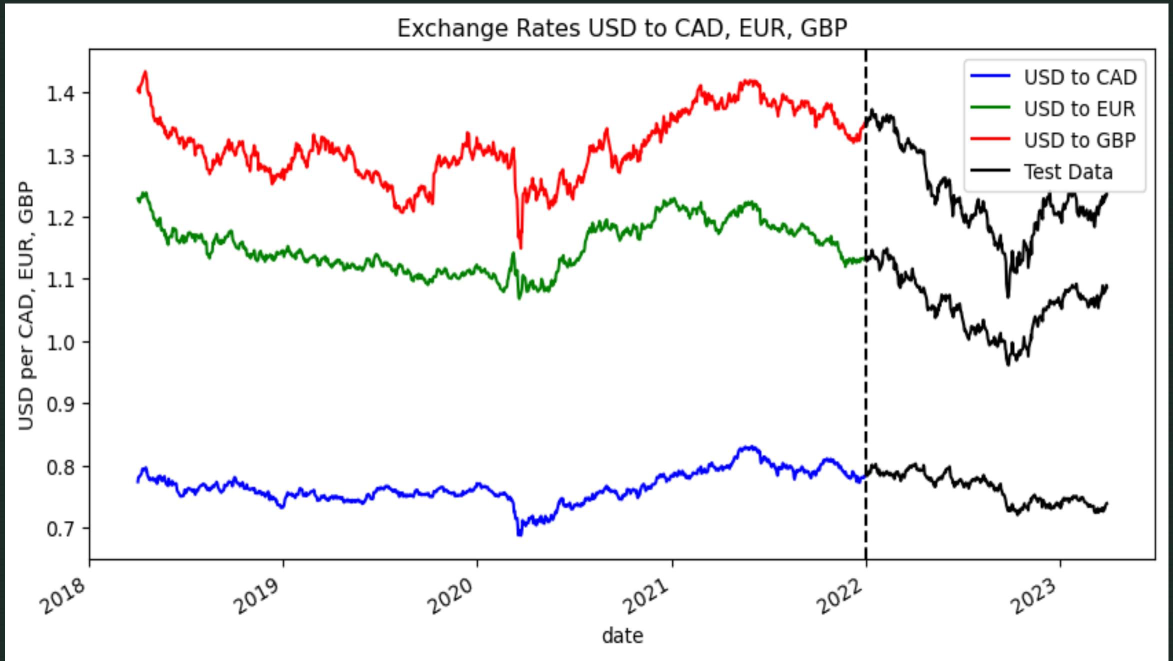
- Exchange rates are a critical component of the global economy, impacting businesses, investors, and policymakers => accurate forecasting is crucial for making informed decisions and managing risks
- Main Targets: Forecasting 3 exchange rates: USD per EUR, USD per CAD and USD per GBP
- Machine learning and Deep learning - capture complex patterns and relationships in the data, increased predictive performance
- Two machine learning methods (Random Forest and eXtreme gradient boosting (Xgboost)) and one deep learning method (Long-Short term memory) are employed in this research

Data

Main dataset: Daily data of 3 exchange rates: 1) USD per CAD, 2) USD per GBP, 3) USD per EUR



Data visualization and descriptive



Splitting point at 01.01.2022 (75% training, 25% for testing)

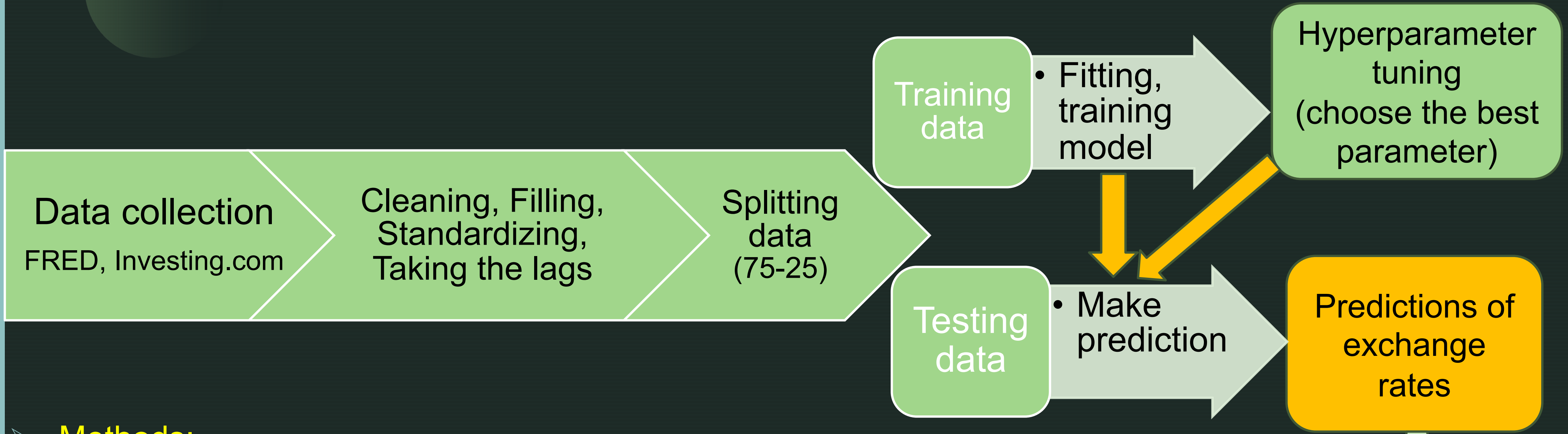
	USD per CAD	USD per EUR	USD per GBP
Mean	0.765193	1.127909	1.293042
Standard Deviation	0.025135	0.057725	0.067342
Minimum	0.687805	0.9616	1.0703
Maximum	0.831186	1.2384	1.433200

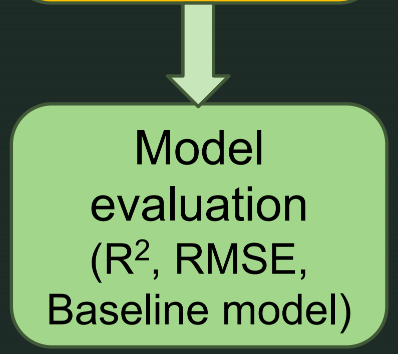
Full sample period: April 02, 2018 to March 31, 2023
(1824 data points)

Fundamental Factors Data

- Exchange rates: Taking lags (1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 19, 20, 21, 22, 23 days lag)
- Stock data: S&P 500 Index (for the U.S.), the S&P TSX Composite Index (for Canada), the STOXX Europe 600 (for Eurozone) and the FTSE 100 (for the U.K.), lags up to 10 trading days. Daily data
- Interest rates: Monthly data
- Money supply: Aggregates M1 and M2 (monthly frequency)
- Macroeconomic indexes: Monthly CPI, Unemployment, and Quarterly GDP data
- Economic Policy Uncertainty Index: Monthly basis
- =>Totally 35-time series with different lengths and frequencies and its lag
- Most of the data taken from FRED and Investing.com (stock indices)

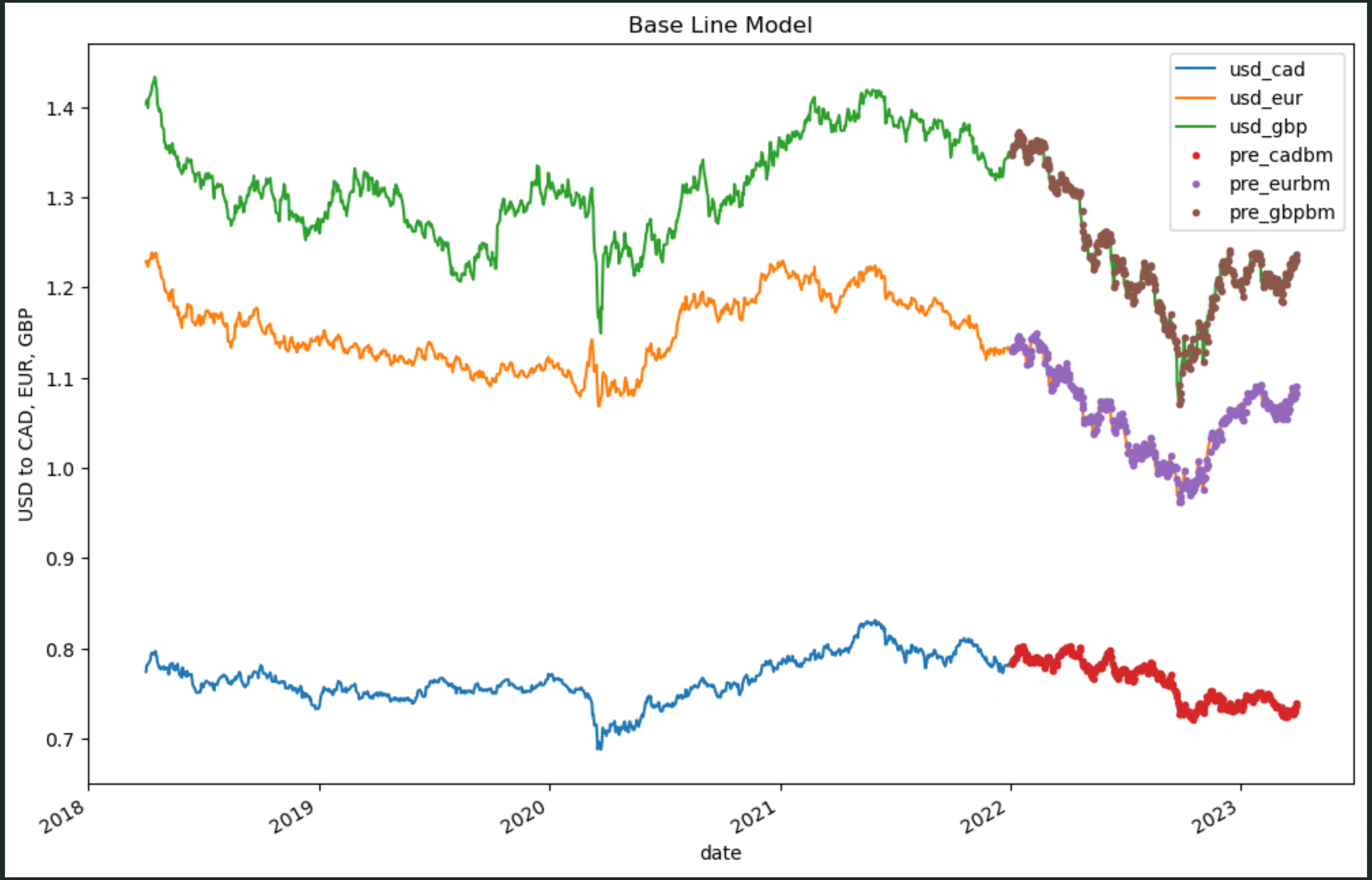
Methodology





- Methods:
  - Supervised machine learning:
  - Random Forest, Extreme Gradient Boosting (XGboost)
  - Deep Learning: Long-Short term memory
- Evaluation: R-square and Roots Mean Square error (RMSE) and Baseline model ($Y_t = Y_{t-1}$)

Performance of baseline model



R^2 close to 1(min 0.975, max 0.983), RMSE small (min 0.003, max 0.47)

Random Forest

Random forest model with 5000 trees, max deep 10

3 separate models. Use RandomForestRegressor()

Hyperparameter tuning

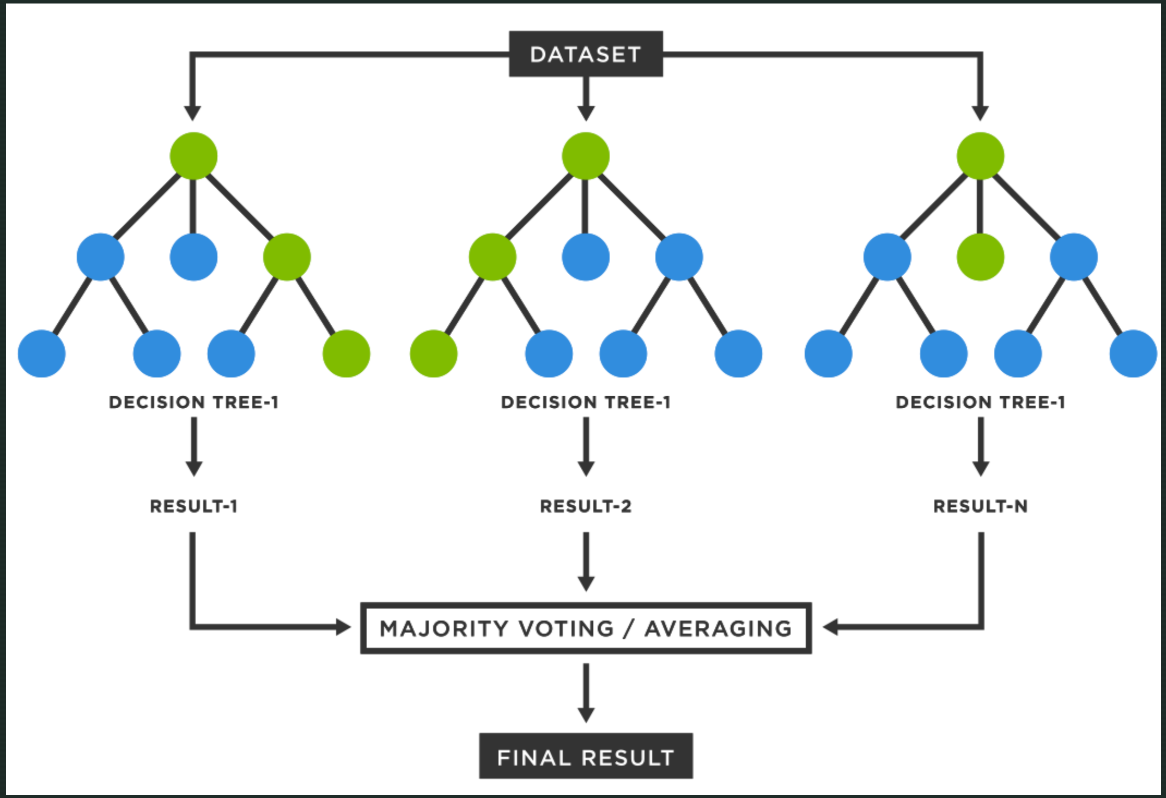
Optimize model's performance. Use RandomizedSearchCV() with 3 fold cv

Fit model with training data

Training the model using traing data

Forecasting

Use trained model with best hyperparameter for prediction



Source: Tibco.com, 2023

XGboost

3 separate models Use XGBRegressor() for gradient boosting model

Random model with 10000 trees, max deep 10

Optimizing gradient boosting models UseRandomizedSearch CV() with 3 fold cv

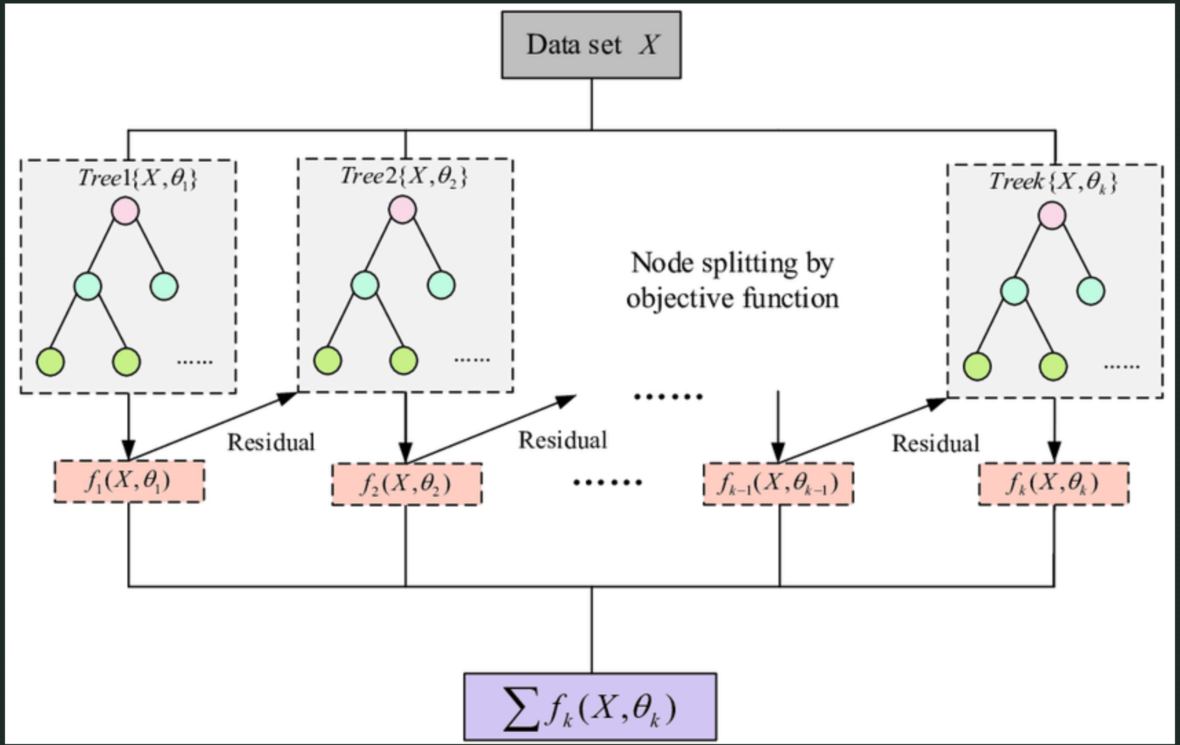
Hyperparameter tuning

Training the model using traing data

Fit model with training data

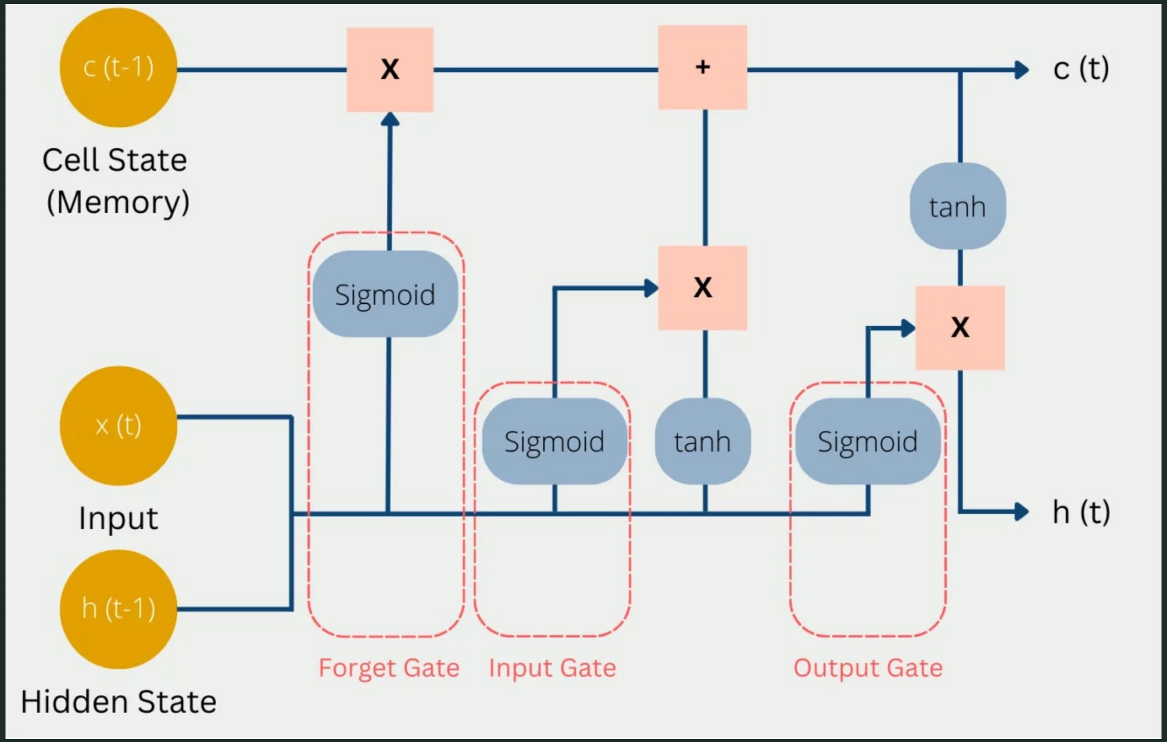
Use trained model with best hyperparameter for prediction

Forecasting



Source: (Guo et al., 2020)

Long Short Term Memory



Source: (Brownlee, 2017)

```python
# Define your LSTM model using Keras
def build_lstm_model(hp):
    model = keras.Sequential()
    model.add(keras.layers.LSTM(units=hp.Int('units', min_value=32, max_value=512, step=32),
                                input_shape=(X_train_reshaped.shape[1], X_train_reshaped.shape[2])))  # LSTM layer
    model.add(keras.layers.Dense(1, activation='linear'))
    model.compile(optimizer=keras.optimizers.Adam(learning_rate=hp.Float('learning_rate', min_value=0.001, max_value=0.01, sampling='log')),
                  loss='mean_squared_error')
    return model

tuner_cad = RandomSearch( build_lstm_model, objective='val_loss', max_trials=50, directory='my_dir_cad', project_name='my_lstm_tuning_cad')
tuner_cad.search(X_train_reshaped, y_train_cadlstm, epochs=30, validation_data=(X_val_lstm, y_val_cadlstm), batch_size=64)
best_model_cad.fit(X_train, y_train_cad, epochs=50, batch_size=64)
```

Hyperparameter results

Random Forest	Optimal Hyperparameter
Model for CAD	{'n_estimators': 5000, 'max_features': 'auto', 'max_depth': 10}
Model for EUR	{'n_estimators': 3666, 'max_features': 'auto', 'max_depth': 20}
Model for GBP	{'n_estimators': 2777, 'max_features': 'auto', 'max_depth': 20}
Xgboost	
Model for CAD	{'n_estimators': 2000, 'max_depth': 1, 'learning_rate': 0.01}
Model for EUR	{'n_estimators': 4000, 'max_depth': 1, 'learning_rate': 0.05}
Model for GBP	{'n_estimators': 2000, 'max_depth': 1, 'learning_rate': 0.01}
LSTM	
Model for CAD	Optimal number of units in the first LSTM layer is 192 Optimal learning rate for the optimizer is 0.005063713408586714.
Model for EUR	Optimal number of units in the first LSTM layer is 480 Optimal learning rate for the optimizer is 0.0012281783364939784.
Model for GBP	Optimal number of units in the first LSTM layer is 384 Optimal learning rate for the optimizer is 0.004325833653269832.

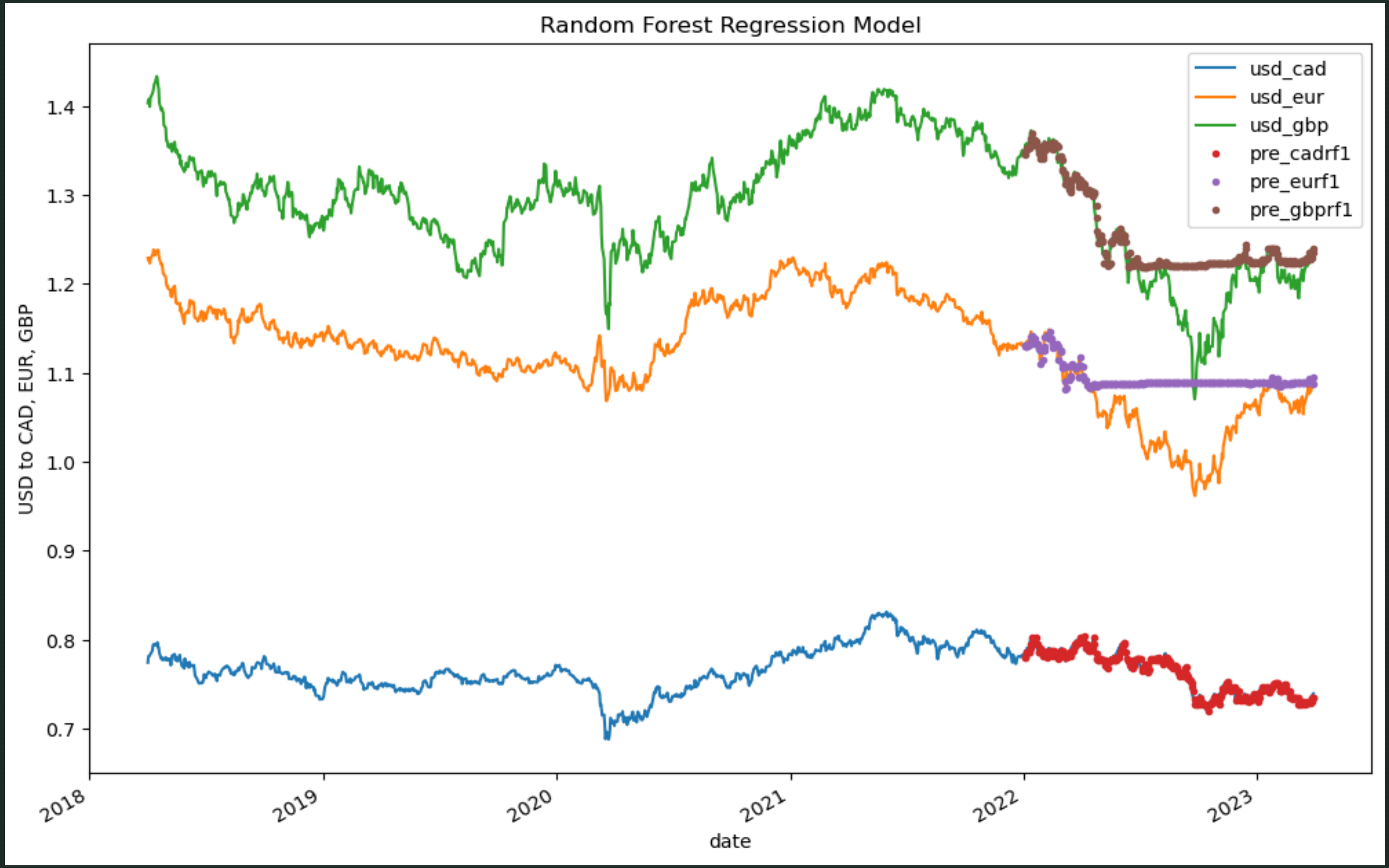
Feature Importance: Historical exchange rates data and Stock market has most contribution. Otherwise, uncertainty of economic policy in U.K., the CPI of U.S. market, Unemployment rates in Europe has large contribution for model also.

Results on Testing data

		USD per CAD	USD per EUR	USD per GBP
R^2	Baseline model	0.9750307713908	0.9796218062956	0.9831046758051
	Random Forest	0.9705185256032	-0.3406085240396	0.6825534239628
	XGboost	0.9641572350851	-0.20346504263931	0.7795793869424
	LSTM	-1.3864030105356	-0.5035408962331	0.5529629576827
RMSE	Baseline model	0.00377388874799093	0.29685244080211	0.47302749630991
	Random Forest	0.004100728193732	0.05259352453577	0.03864119956181 3
	XGboost	0.004521547904647	0.3297491594942	0.48621161451222
	LSTM	0.0368942153720	0.3413867733888	0.4935872804441

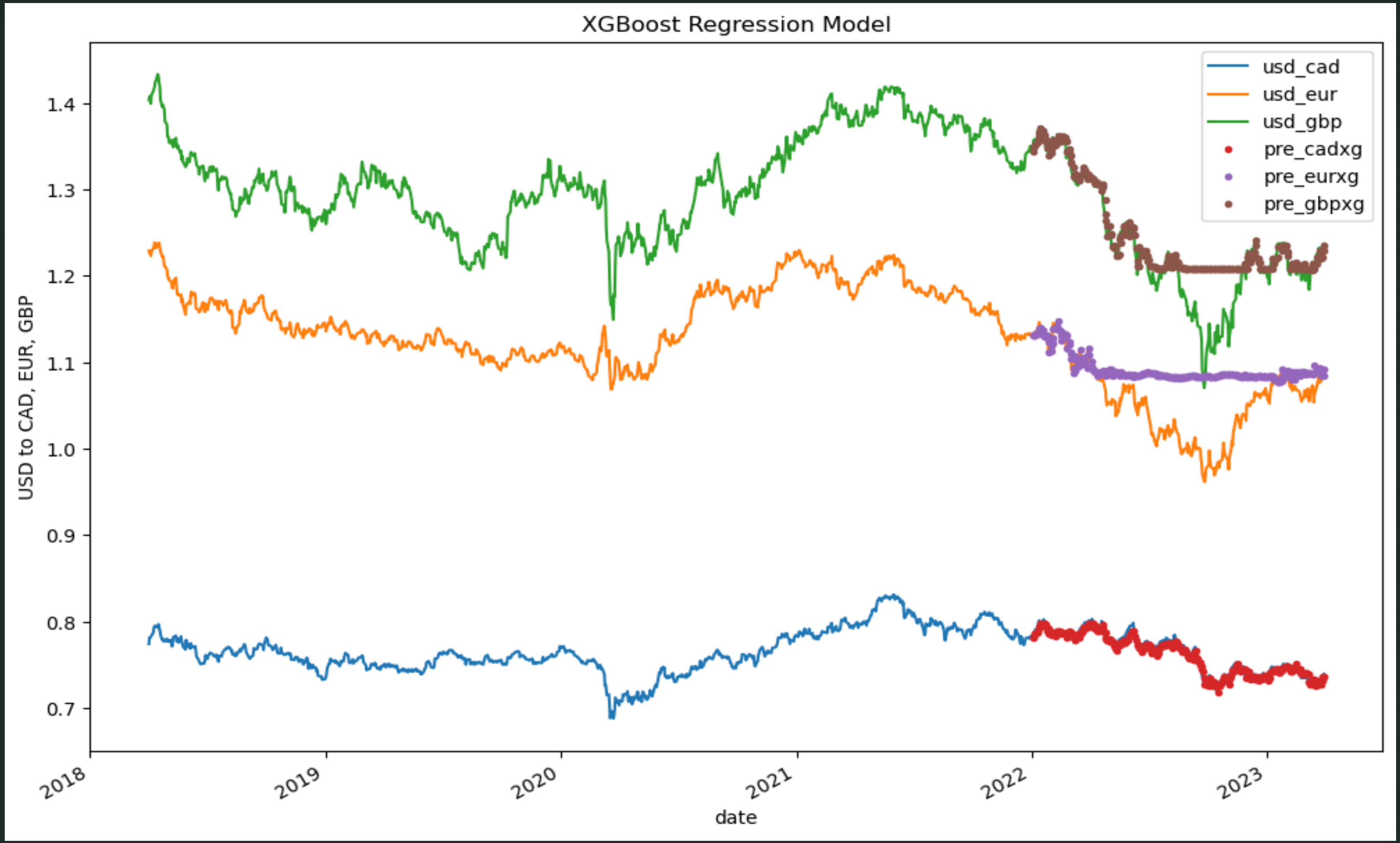
R^2 is highest for USD-CAD exchange rate comparable with baseline model (Xgboost and random forest , but LSTM perform poorly for USD-CAD. 3 models have average fitting for USD-GBP and poorly performed for USD-EUR.

Visualization Random Forest



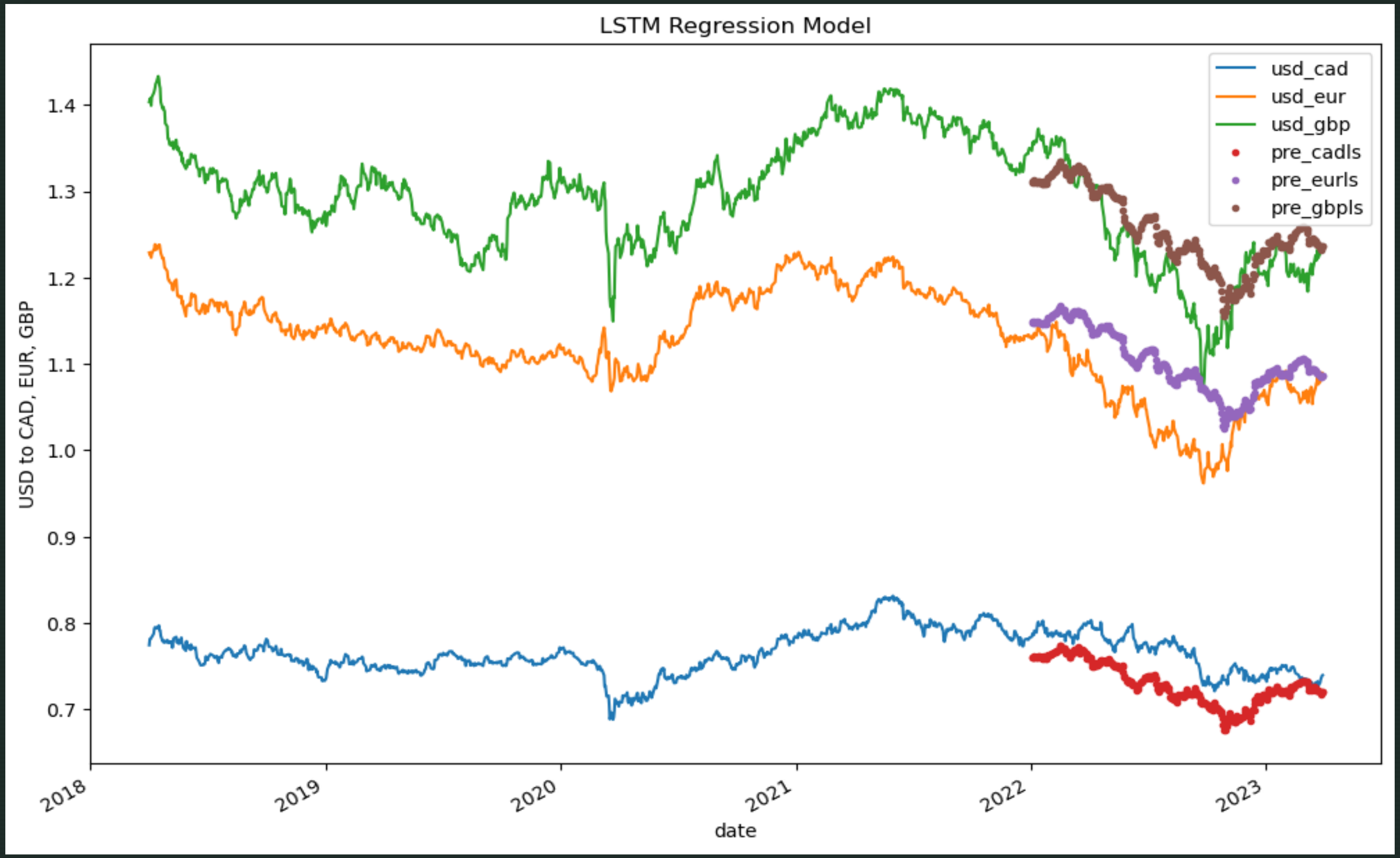
Random Forest can capture pretty precise of Echange rates movement is CAD-USD and average preformance in GBP-USD exchnage rates but not quite good in forcasting EUR-USD

Visualization Xgboost



XGboost can capture pretty precise of Echange rates movement is CAD-USD and average preformance in GBP-USD exchnage rates but not quite good in forcasting EUR-USD. But fitting better than Random forest

Visualization LSTM



LSTM can capture the pattern of exchange rates movement but not presice in forcasting

Limitation of 3 models

Overfitting

For Xgboost and random forest model has strong performance on the training data compare to testing data (higher R^2 (>0.989, and very small RMSE on training set)

Underfitting

For LSTM model has avearge to poor preformance on both testing and traning set. Learned the pattern but not precise in prediction

Data Limitation

Lack of data to extend the training set further than the deep droping point

Conclusion

- XGBoost outperforms the other two models in 3 exchange rate predictions (especially for CAD)
- Historical exchange rates data and Stock market has most contribution for the model, uncertainty of economic policy in UK, the CPI of US market, Unemployment rates in Europe also have large effect.
- 3 machine learning models in the report perform worse than a very simple baseline model
- Future research: explore enhanced cross-validation techniques to time series data, more extensive datasets to bolster the LSTM model's predictive capabilities.

Thanks for your attention

References

1. Brownlee, J. (2017). *Long short-term memory networks with python: develop sequence prediction models with deep learning*: Machine Learning Mastery

2. Brownlee, J. (2016). *XGBoost With python: Gradient boosted trees with XGBoost and scikit-learn*: Machine Learning Mastery.

3.Brownlee, J. (2021). *Ensemble learning algorithms with Python: Make better predictions with bagging, boosting, and stacking*: Machine Learning Mastery.

4. Guo, R., Zhao, Z., Wang, T., Liu, G., Zhao, J., & Gao, D. (2020). Degradation state recognition of piston pump based on ICEEMDAN and XGBoost. *Applied Sciences, 10*(18), 6593.

5. Federal Reserve Bank of St. Louis (FRED). (2023). Federal Reserve Economic Data | FRED | St. Louis Fed. Retrieved 20.08.2023, from https://fred.stlouisfed.org/

6. Tibco.com. (2023). Retrieved 05.09.2023, from https://www.tibco.com/reference-center/what-is-a-random-forest