Portfolio Analytic Group – Screening Test
Forecast Kurs Rupiah (IDR)

Tujuan:
Melakukan ramalan kurs IDR dengan menggunakan metode multivariate linear regression dengan variabel berupa indeks USD (*USDTW*) dan neraca berjalan (*CA*).

Metodologi:
Utamanya akan digunakan metode multivariate linear regression dengan persamaan
$IDR = \beta_0 + \beta_1 * USDTW + \beta_2 * CA$
Untuk mencapai konvergensi, digunakan metode ordinary least square (OLS) untuk menemukan local minima metrik error berupa R-squared dan Root Mean Square Error

Kemudian, penulis juga menggunakan metode Random Forest Regressor dengan metrik error yang sama. Metode ini diharapkan dapat menangkap relasi non-linear pada variabel yang tidak dapat digambarkan oleh metode OLS

Contoh Data:

	Month-Year	USDTW	CA	IDR
0	1986-03-01	62.9331	-721.0	1132.75
1	1986-06-01	62.3251	-1174.0	1136.00
2	1986-09-01	61.1942	-1371.0	1492.80
3	1986-12-01	62.4267	-833.0	1655.40
4	1987-03-01	60.4773	-673.0	1651.53

.
.
.

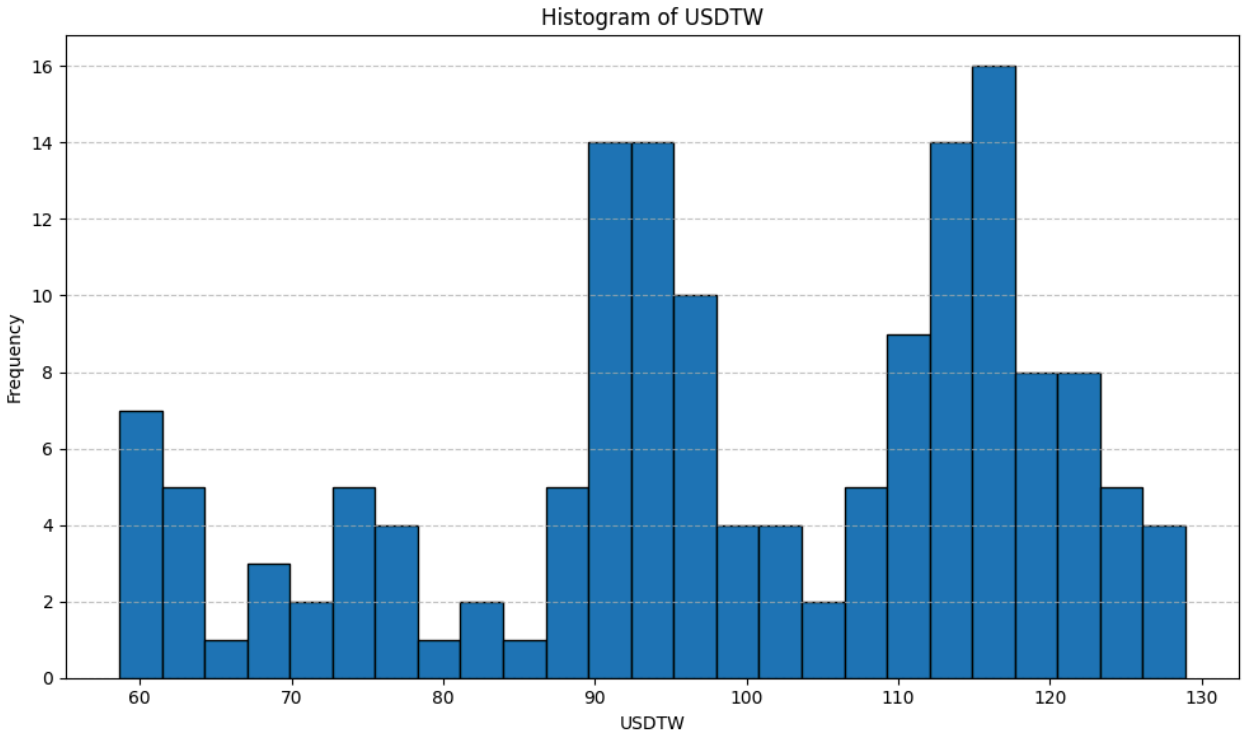
	Month-Year	USDTW	CA	IDR
148	2023-03-01	120.8071	2775.03	15062.0
149	2023-06-01	119.5789	-2363.76	15026.0
150	2023-09-01	121.9827	-1170.89	15526.0
151	2023-12-01	120.1585	-1119.97	15416.0
152	2024-03-01	121.0413	-2160.81	15853.0

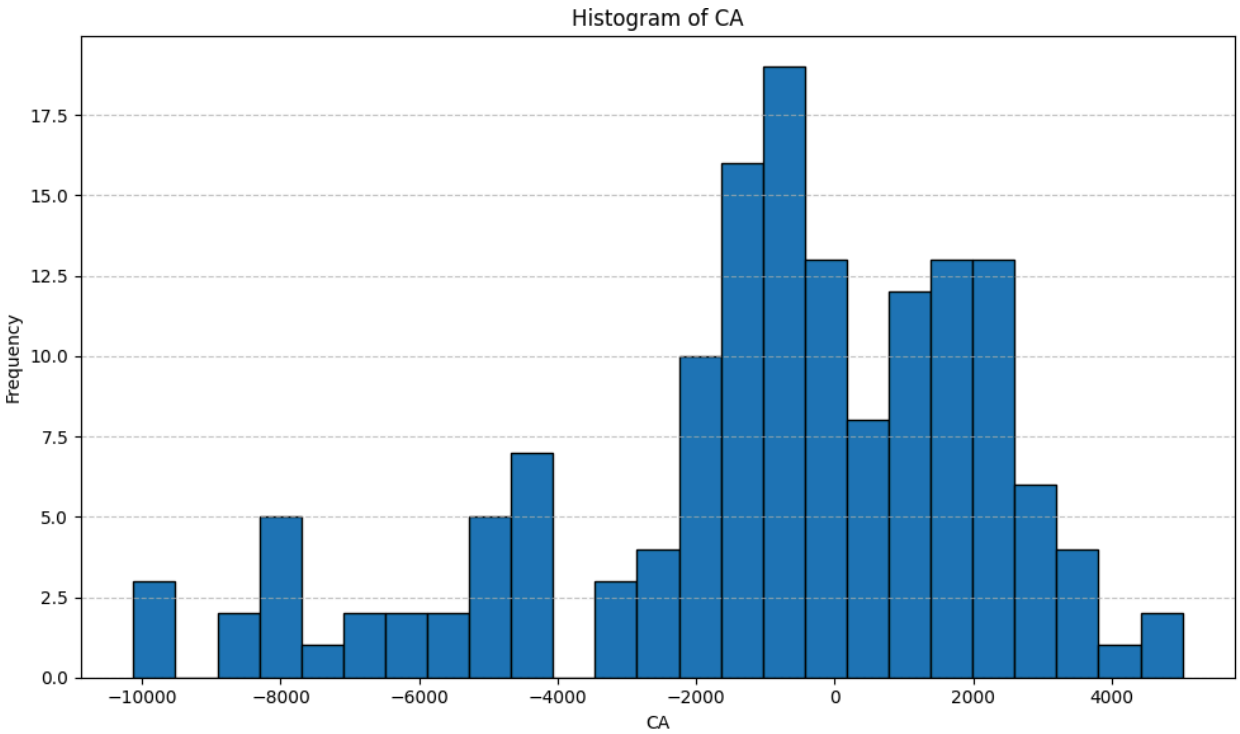
Tercantum merupakan awal dan akhir dari sumber data yang akan digunakan. Data memiliki 4 kolom yaitu triwulan, USDTW, CA, dan IDR yang merupakan target predikisi. Data ini memiliki 152 data yang masing-masing menggambarkan satu triwulan

Analisa dan Data:
Pertama, kita dapat melihat statistik dari USDTW dan CA.

	USDTW	CA	IDR
count	153.00	153.00	153.00
mean	99.48	-966.42	8,345.44
std	19.03	3,202.75	4,788.97
min	58.64	-10,125.65	1,132.75
25%	90.01	-2,034.00	2,276.00
50%	99.66	-637.00	9,118.00
75%	115.61	1,477.00	12,440.00
max	128.94	5,020.45	16,367.01

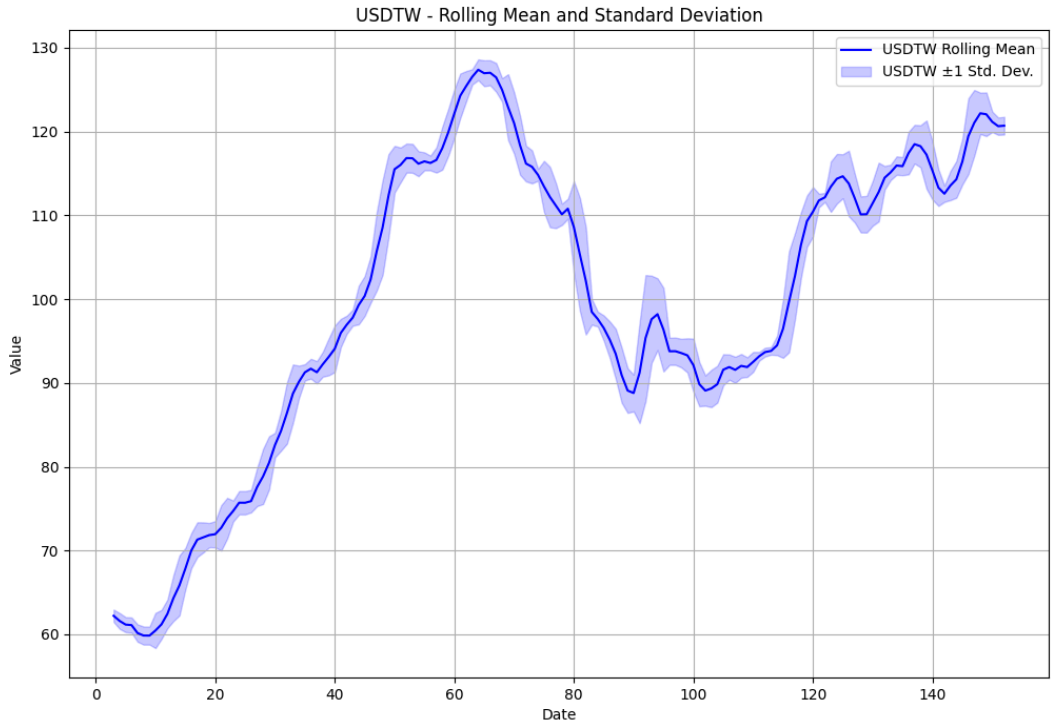
Terlihat bahwa masing-masing kolom memiliki statistik yang cukup baik, tanpa adanya outlier yang berlebihan. Lalu, penulis akan melihat distribusi dari fitur yaitu USDTW dan CA

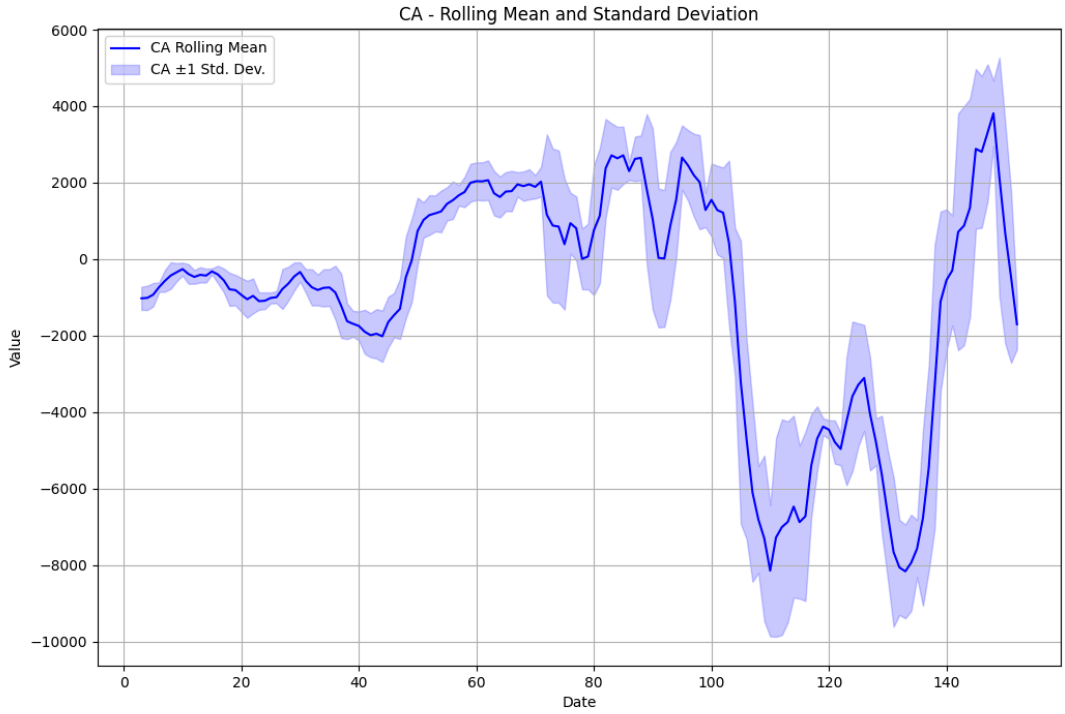




Terlihat bahwa walaupun kedua fitur ini memiliki outlier yang minimal, distribusi keduanya cukup jauh dari jenis-jenis distribusi umum seperti distribusi normal, log-normal, ataupun decay. Data pun terlihat. Oleh karena itu, penulis akan mencoba mencari fitur turunan untuk mengubah data menjadi normal terutama untuk penggunaan metode-metode nonlinear.

Selanjutnya, penulis akan memeriksa stasioneritas data





Karena nilai rata-rata dan nilai standar deviasi USDT dan CA memiliki variasi yang signifikan, maka dapat diambil kesimpulan data tidaklah stasioner. Maka dari itu, penulis akan mencoba metode differencing pada data, dan memutuskan untuk memakai metode Random Forest

Regressor pada hubungan nonlinear dan bukan metode time series seperti ARIMA, Var, atau Long-short term memory yang membutuhkan data dengan stasionerirtas tinggi.

Penulis juga melakukan metode z-normalization yang menyebabkan data USDTW dan CA memiliki nilai tengah 0 dan standar deviasi 1. Normalisasi ini membantu dalam memastikan bahwa semua variabel memiliki skala yang sama, sehingga mencegah satu variabel mendominasi yang lain dalam proses analisis. Dengan z-normalization, kita dapat membandingkan perubahan relatif dari variabel-variabel ini secara lebih konsisten dan menghindari bias yang mungkin timbul karena perbedaan skala asli.

Setelah melakukan normalisasi dan satu metode differencing untuk mendapatkan data yang lebih stasioner, maka didapatkan data berupa

Month-Year	USDTW	CA	IDR	pct_change_idr	month	month_scaled	pct_change_usdtw	pct_change_ca	usdtw_scaled	ca_scaled	pct_change_usdtw_scaled	pct_change_ca_scaled
2023-03-01	120.8071	2,775.0300	15,062.0000	-0.0118	3.0000	0.2500	-0.0118	-0.2069	1.1247	1.1720	-0.5878	-0.1571
2023-06-01	119.5789	-2,363.7600	15,026.0000	-0.0102	6.0000	0.5000	-0.0102	-1.8518	1.0599	-0.4377	-0.5301	-0.9031
2023-09-01	121.9827	-1,170.8900	15,526.0000	0.0201	9.0000	0.7500	0.0201	-0.5046	1.1867	-0.0641	0.5489	-0.2921
2023-12-01	120.1585	-1,119.9700	15,416.0000	-0.0150	12.0000	1.0000	-0.0150	-0.0435	1.0905	-0.0481	-0.7008	-0.0830
2024-03-01	121.0413	-2,160.8100	15,853.0000	0.0073	3.0000	0.2500	0.0073	0.9293	1.1370	-0.3742	0.0942	0.3582

Pemodelan Multivariate Linear Regression:
Dengan menggunakan Ordinary Least Square untuk menemukan koefisien beta terbaik dengan dua jenis error independen dan recursive step untuk menemukan panjangnya data yang memberikan error terkecil. Pada model ini kami menggunakan nilai data fitur mentah tanpa adanya normalisasi atau transformasi untuk untuk menjaga kemudahan pemahaman model.

1. Dengan error R-Squared terbaik
   R-squared terbaik diraih dengan menggunakan data dari 2007-06-01 sampai data 2024-03-01, dengan nilai 0.918124. Persamaan yang diraih ialah
   $IDR = -7971.9658 + 191.4956 * USDTW - -0.0712 * CA$

   Selain itu, model ini mendapatkan mean squared error sebesar 687.77.

   Nilai R-squared yang mendekati 1.0 menunjukkan bahwa model ini memiliki kemampuan yang sangat baik dalam menjelaskan variasi data yang diamati. Dengan kata lain, model dapat menjelaskan sebagian besar dari variasi dalam variabel target berdasarkan fitur-fitur yang digunakan dalam model. Ini menunjukkan bahwa model memiliki kecocokan yang baik dengan data dan dapat memprediksi hasil dengan akurasi yang tinggi.

   Nilai RMSE sebesar 687.77, yang relatif kecil dibandingkan dengan nilai rata-rata asli 8,345.44 atau dengan error rata-rata sebesar 8.23%, menunjukkan bahwa model memiliki performa yang baik dalam memprediksi nilai target, dengan kesalahan prediksi rata-rata yang rendah dibandingkan dengan skala data.

2. Dengan error RMSE terbaik
   RMSE terbaik diraih dengan menggunakan data dari 2017-09-01 sampai data 2024-03-01, dengan nilai 432.59. Persamaan yang diraih ialah
   IDR = 151.7252 + 124.5660 * USDTW - -0.0066 * CA

   Selain itu, model ini mendapatkan R-Squared sebesar 0.618096.

   Nilai RMSE sebesar 432.59, yang relatif kecil dibandingkan dengan nilai rata-rata asli 8,345.44 atau dengan error rata-rata sebesar 5.18%, menunjukkan bahwa model memiliki performa yang baik dalam memprediksi nilai target, dengan kesalahan prediksi rata-rata yang rendah dibandingkan dengan skala data.

   Namun, dengan R-squared 0.618096, yang lebih rendah dari model sebelumnya, yang memiliki nilai R-squared lebih tinggi, menunjukkan bahwa meskipun model ini memiliki RMSE yang lebih baik, kemampuannya dalam menjelaskan variasi data masih kurang dibandingkan dengan model yang lebih kuat sebelumnya. Hal ini mungkin mengindikasikan bahwa meskipun kesalahan prediksi rata-rata lebih rendah, model ini belum sepenuhnya menangkap hubungan kompleks dalam data.

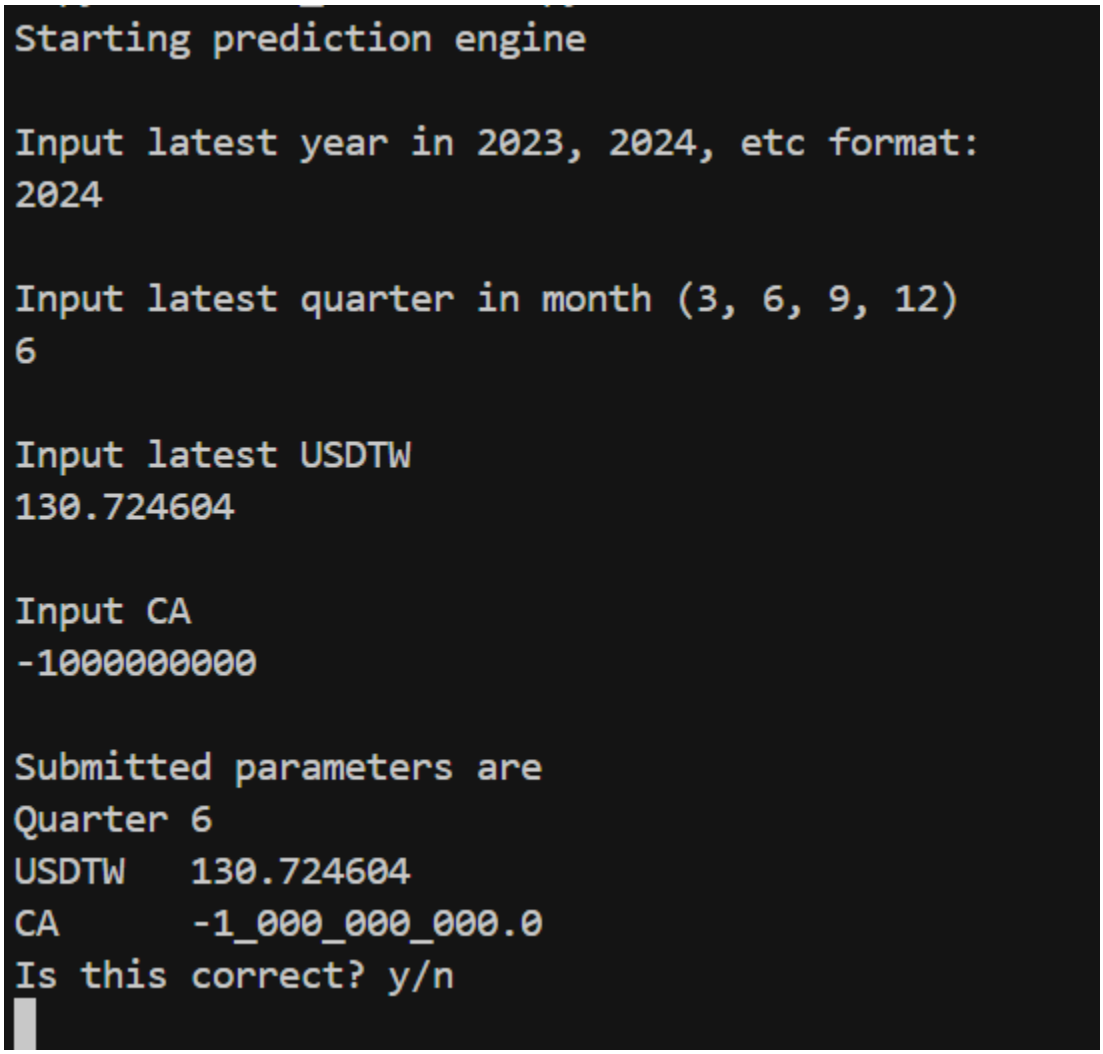
Pemodelan Random Forest Regressor:
Penulis mencoba menggunakan random forest regressor untuk menangkap hubungan non-linear yang mungkin ada pada data. Pun, kali ini kami menggunakan data yang telah ditransformasi dan dinormalisasi untuk mengurangi bias pada model machine learning. Metode penilaian model yang digunakan sama dengan model sebelumnya.

1. Dengan error R-Squared terbaik
   R-squared terbaik diraih dengan menggunakan data dari 2007-12-01 sampai data 2024-03-01, dengan nilai 0.987367. Selain itu, model ini mendapatkan mean squared error sebesar 266.63
2. Dengan RMSE terbaik
   RMSE terbaik diraih dengan menggunakan data dari 2019-03-01 sampai data 2024-03-01, dengan nilai 153.86. Selain itu, model ini mendapatkan R-squared 0.946078

Dengan nilai R-squared dan RMSE yang lebih baik di kedua extremum dibandingkan dengan model OLS, dapat ditarik kesimpulan bahwa model Random Forest memiliki performa yang lebih baik dalam memprediksi nilai target. Model ini lebih efektif dalam menangkap hubungan non-linear dalam data dan memberikan prediksi yang lebih akurat. Hasil ini menunjukkan bahwa Random Forest dapat menjadi pilihan yang lebih baik daripada OLS untuk data dengan hubungan kompleks, terutama setelah data diolah melalui transformasi dan normalisasi yang tepat.

Contoh Penggunaan Prediksi:
Pengguna dapat mengikuti langkah-langkah instalasi dan menyalakan sistem yang akan tertera pada dokumen *readme*. Setelah Sistem berjalan, maka pengguna akan disajikan sebuah interface dimana pengguna dapat memasukkan nilai USDTW dan CA, dan mendapatkan hasil prediksi IDR

Setelah sistem beroperasi, pengguna akan diberikan beberapa pertanyaan dan input data



Terlihat bahwa sistem menanyakan triwulan terakhir beserta data USDTW dan CA terbaru. Pada kasus ini penulis mengisi dengan kenaikan 8% pada USDTW dari nilai terakhir dan defisit 1.000.000.000 pada CA.

Setelah itu, sistem akan mengeluarkan empat prediksi, yaitu dengan menggunakan Multivariate Linear Regression dengan metrik R-Squared dan RMSE terbaik, dan Random Forest Regressor dengan dua metrik yang serupa, sehingga diperoleh

```
Predicting using Ordinary Least Square Regressor with best R-squared metric
Prediction result is 71_261_351.76

Predicting using Ordinary Least Square Regressor with best Root Mean Squared Error metric
Prediction result is -6_623_034.37

Predicting using Random Forest Regressor with best R-squared metric
Prediction result is 15_762.63

Predicting using Random Forest Regressor with best Root Mean Squared Error metric
Prediction result is 15_803.38
```

Dari hasil analisis, didapatkan bahwa kedua model Linear Regression memberikan hasil prediksi yang tidak realistis, seperti harga rupiah sebesar 71.000.000 dan -6.000.000. Masalah ini disebabkan oleh sensitivitas model linear terhadap outlier dan ketidakmampuannya untuk menangkap hubungan non-linear dalam data. Misalnya, parameter defisit sebesar 1.000.000.000 pada neraca berjalan, yang merupakan outlier signifikan dibandingkan dengan nilai maksimum terbaru yang hanya sebesar 5.000, dapat menyebabkan hasil prediksi yang jauh dari nilai yang sebenarnya. Hal ini menunjukkan bahwa Linear Regression mungkin tidak cocok untuk data dengan outlier atau hubungan yang kompleks tanpa penyesuaian tambahan.

Di sisi lain, model-model Random Forest memberikan hasil yang lebih masuk akal, dengan pergeseran nilai rupiah tetap berada di sekitar angka 15.000. Hal ini mendukung asumsi bahwa Random Forest lebih mampu menangkap hubungan non-linear antara USDTW, CA, dan IDR. Selain itu, hasil ini juga menegaskan pentingnya normalisasi dan stasionaritas data dalam meningkatkan akurasi model. Normalisasi dan stasionaritas membantu model dalam memahami pola yang lebih kompleks dan membuat prediksi yang lebih konsisten dan dapat diandalkan.

Saran:
Dengan analisa diatas, penulis menyarankan beberapa tahap tahap di kemudian hari.

Tahap Pertama meliputi pengumpulan data tambahan yang mungkin mempengaruhi variabel target, seperti faktor-faktor eksternal atau indikator ekonomi lainnya. Pertimbangkan pengembangan model kombinasi atau ensemble, yang menggabungkan berbagai model untuk meningkatkan akurasi prediksi. Integrasikan model ke dalam sistem produksi atau alur kerja analisis yang ada dan otomatisasikan proses pelatihan serta evaluasi model. Lakukan analisis sensitivitas untuk memahami bagaimana perubahan pada variabel input mempengaruhi hasil prediksi dan identifikasi serta kelola risiko yang mungkin terkait dengan penggunaan model.

Tahap Kedua ialah menggunakan rekayasa data untuk mentransformasi data sehingga ternormalisasi dan memiliki stasionaritas tinggi. Hal ini dapat dilakukan dengan metoda

differencing seperti yang telah dilakukan, menggunakan indikator temporal seperti moving averages, atau gabungan dari keduanya.

Tahap Ketiga ialah menggunakan model time series seperti ARIMA, Vector Autoregression (VAR), dan Long Short-Term Memory (LSTM) lebih sesuai untuk data time series karena mereka dirancang untuk menangkap pola temporal dan ketergantungan jangka panjang dalam data. Berbeda dengan model regresi steady-state yang menangkap hubungan statis antar variabel, model time series mempertimbangkan dinamika perubahan antar waktu waktu yang data, seperti tren dan musiman, serta autokorelasi. Stasionaritas, yaitu memiliki statistik yang konsisten sepanjang waktu, sangat penting untuk model time series seperti ARIMA dan VAR. Data harus distasionerkan melalui differencing atau transformasi lain untuk memastikan akurasi model dan kemampuan prediksi yang baik.

Tahap Kelima adalah review dan iterasi, dengan meninjau hasil dan proses secara berkala. etelah fitting model, penting untuk mengevaluasi kinerja model menggunakan teknik seperti cross-validation atau dengan membagi data menjadi set pelatihan dan pengujian. Periksa residual model untuk memastikan tidak ada pola yang tidak tertangkap dan model mampu memprediksi dengan baik. Lakukan pengujian dengan data baru dan sesuaikan hyperparameter atau model berdasarkan hasil evaluasi. Terapkan model ke data nyata, monitor performa secara terus-menerus, dan lakukan pemeliharaan atau pembaruan jika diperlukan.

Tahap Keenam adalah implementasi dan integrasi model. Gunakan API atau front-end GUI untuk memudahkan interaksi dengan model, daripada terminal. Integrasikan model dengan database internal untuk akses yang lebih cepat dan efisien vs dengan metode penyimpanan data seperti file CSV. Ini akan meningkatkan kemudahan penggunaan, otomatisasi, dan skalabilitas sistem prediksi.