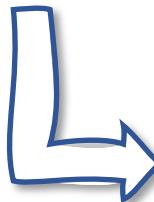




Kajian Literatur Model Hybrida *Transformer Bidirectional Long Short-Term Memory – XGBoost* dengan Koreksi Galat pada Deret Waktu



Model hibrida yang dikembangkan guna mengatasi keterbatasan model dalam menghadapi situasi nonlinieritas, multi-musiman, dan heteroskedastisitas yang umum terjadi pada data deret waktu dengan amatan banyak.

Ghardapaty G. Ghiffary



Today Report Outline

.....

- Latar Belakang – Lingkup Batasan
- Ekstrak, Transformasi dan Eksplorasi Data
 - Pengenalan Pola
 - Uji T Dampak Kejadian
 - Interaksi Bulan dan Hari
 - Jendela Rataan-Stdev Bergerak
 - Analisis Gap dan Anomali
- Pemodelan Transformer BiLSTM
- Pemodelan Galat dengan Boosting (XGB)
- Pemodelan Hybrida dan Evaluasi Model

Latar Belakang

1.

Data dengan pola musiman, anomali, dan volatilitas heteroskedastik terkadang terkendala asumsi pada ARIMA, SARIMA dan GARCH.

$$\text{Var}(P_t) \neq \text{Var}(P_{t+h}), \quad \forall h > 0$$

$$P_t = \mu + \sum_{j=1}^k s_t^{(j)} + R_t$$

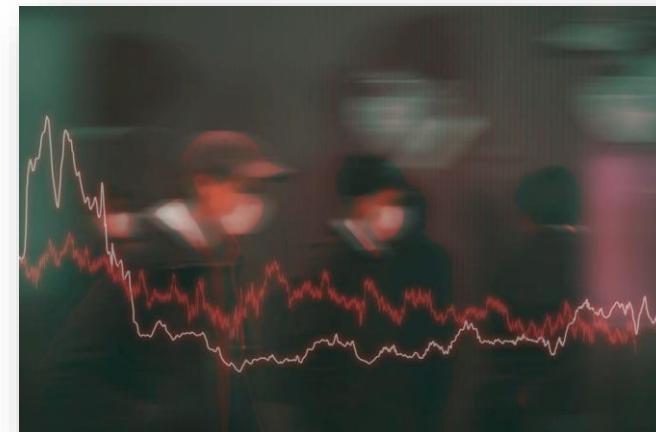
2.

Keterbatasan dapat diatasi dengan model non parametrik melalui 2 tahap yaitu:

- Penangkapan pola jangka Panjang
- Koreksi galat pada model dasar



$$\hat{y}_t^{base} = f\theta(P_{t-\tau:t}, X_{t-\tau:t})$$



?



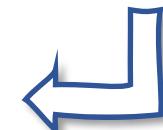
3.

Pengembangan lebih lanjut menggabungkan Transformer, BiLSTM dan XGBoost dalam memodelkan deret waktu.

4.

3 Kunci pada Pengembangan model ini meliputi:

- Transformer sbg dependensi jangka panjang
- BiLSTM melihat dua arah
- XGB penanganan ragam tak konstan pada galat



$$\hat{\tau}_t = g\phi(\hat{y}_t - \hat{y}_t^{base})$$

Latar Belakang

1. Transformer

- Positional Encoding

Arsitektur untuk memberikan informasi urutan temporal kepada model guna model mempelajari hubungan relatif antar posisi melalui linear kombinasi.

- Multi-Head Self Attention

Membagi attention menjadi beberapa head kemungkinan kejadian (pola musiman, tren jangka Panjang, dampak kejadian antar amatan, dll)

- Feed-Forward Network (FFN)

Menangani perubahan non-linier antar token setelah proses *self-attention*, bertindak seperti "pengolah lokal" untuk setiap posisi dalam urutan, independen terhadap posisi lain.



$$PE_{t,2i} = \sin\left(\frac{t}{10000^{\frac{2i}{d}}}\right),$$

$$PE_{t,2i+1} = \cos\left(\frac{t}{10000^{\frac{2i+1}{d}}}\right),$$

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

$$Q = XW^Q; K = XW^K; V = XW^V$$

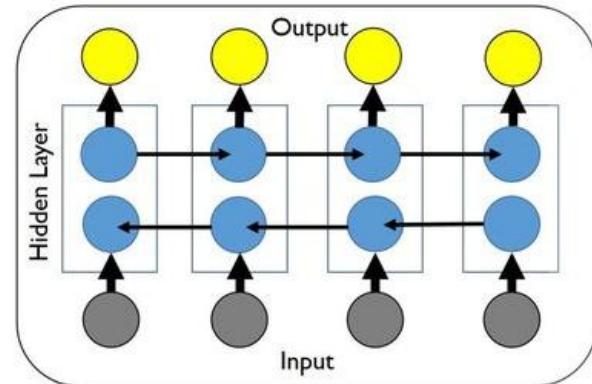
$$FFN(x) = W_2\phi(W_1x + b_1) + b_2$$

ϕ dengan ReLU atau GELU



Latar Belakang

2. BiDirectional LSTM



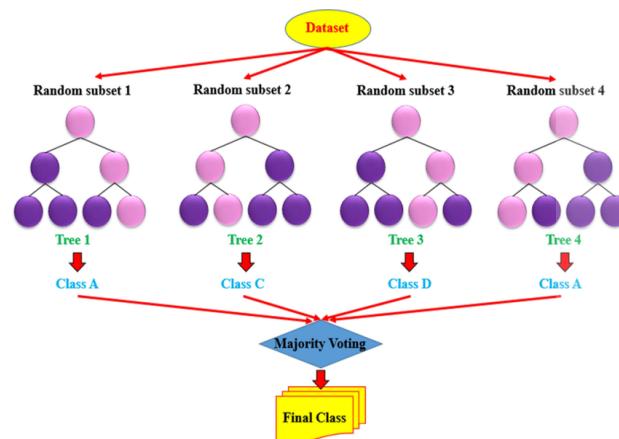
BiLSTM Architecture
Graves & Schmidhuber, 2005

Model neural network sekuensial yang menangkap pola jangka panjang dari masa lalu (forward) dan masa depan (backward) secara simultan.

$$\begin{aligned} f_t &= \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \\ i_t &= \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \\ o_t &= \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \\ C_t &= f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \end{aligned}$$

Forget Gate
Input Gate
Output Gate
Cell State Update

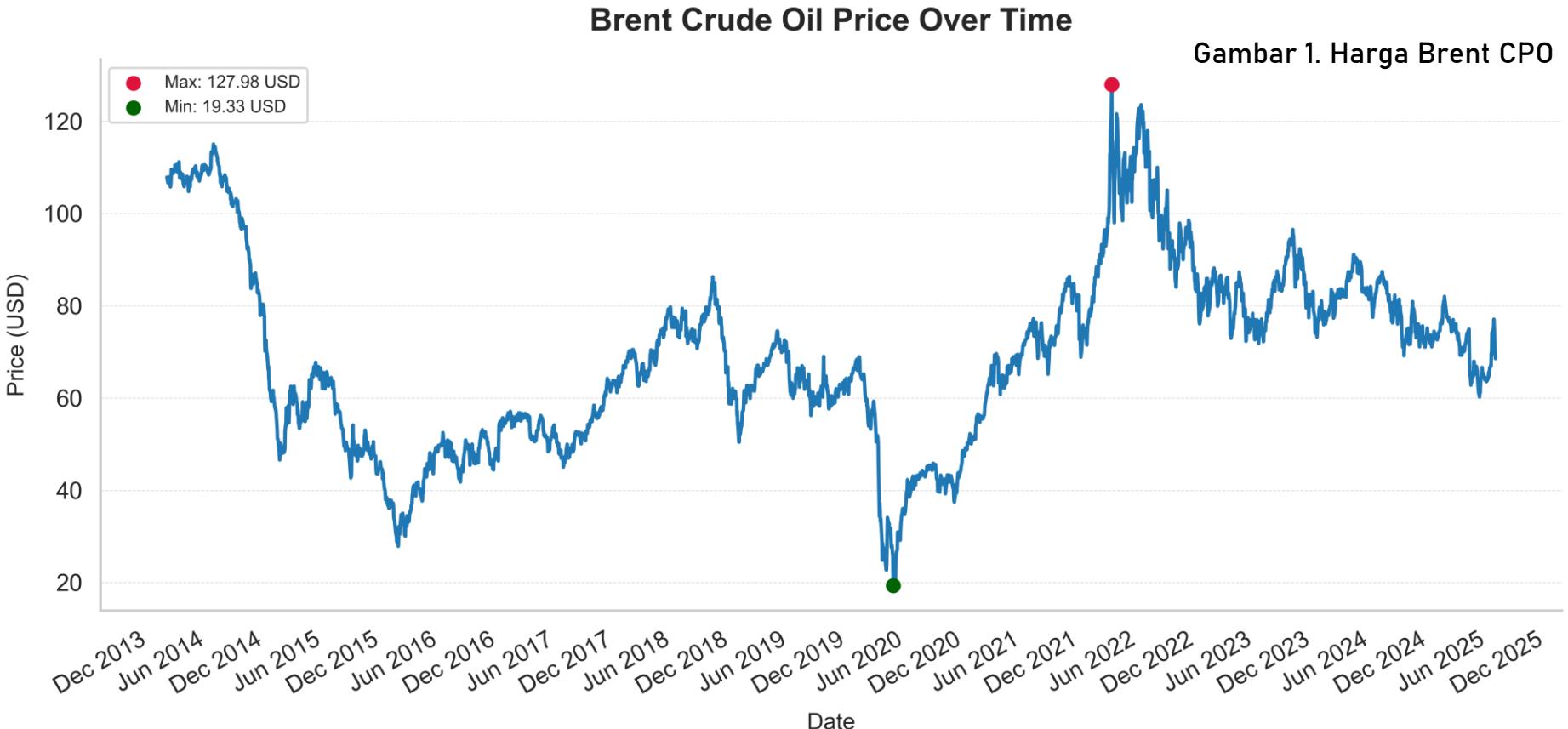
3. XGBoost



Model ensemble berbasis pohon keputusan yang mengoptimalkan fungsi loss melalui *boosting*. Cocok untuk menangkap pola non-linieritas pada suatu data serta bersifat kekar atau *robust*.

$$\mathcal{L}^{(t)} = \sum_{i=1}^n [l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i))] + \Omega(f_t)$$

Latar Belakang



Data yang digunakan merupakan data harian dengan frekuensi hari kerja. Periode pengambilan data adalah 24/06/2014 – 24/06/2025.

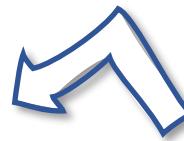


Data dengan Volatilitas Tinggi

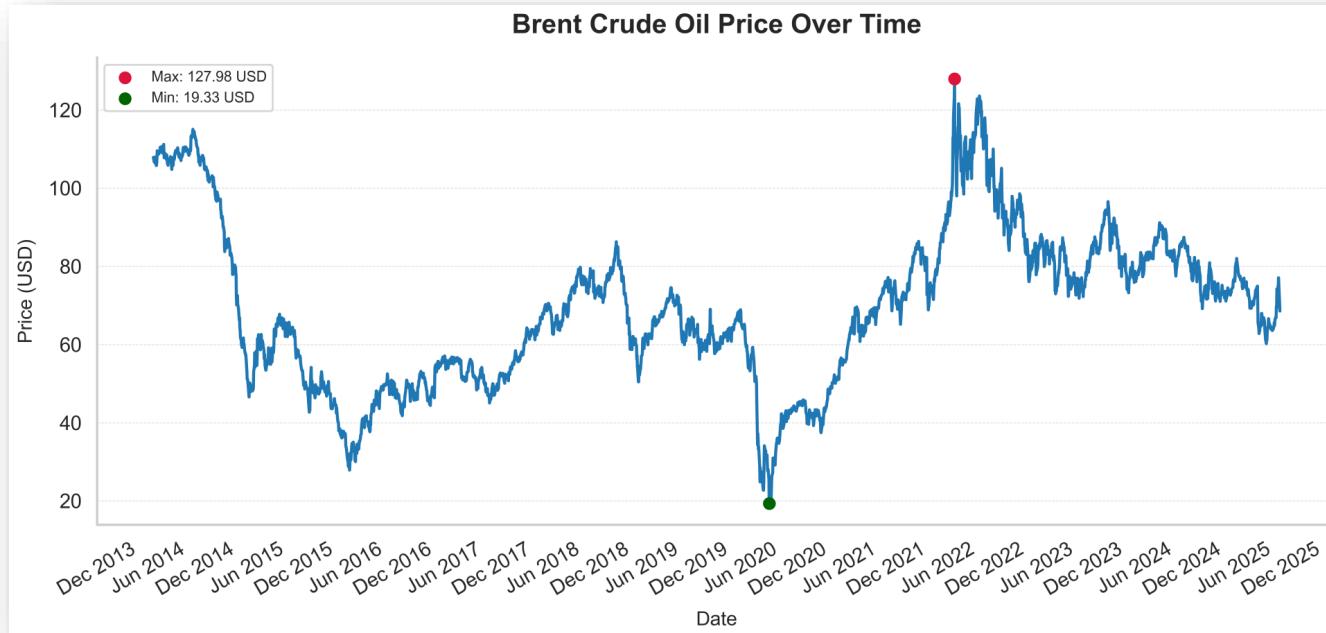
Terdapat 2978 amatan hari

ETL dan EDA

Perlu peubah dummy untuk intervensi kejadian di rentang waktu terkait?



Terdapat pencilan dan nilai ekstrem, pertanda efek waktu lainnya?



Apakah terdapat kondisi efek tahunan, bulanan, mingguan, harian atau tanggal unik kalender?



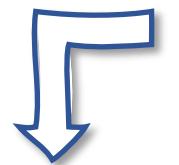
Seberapa banyak kejadian besar di rentang tahun ini? ?

Terdapat 17+ Fenomena Ekonomi yang terjadi



ETL dan EDA

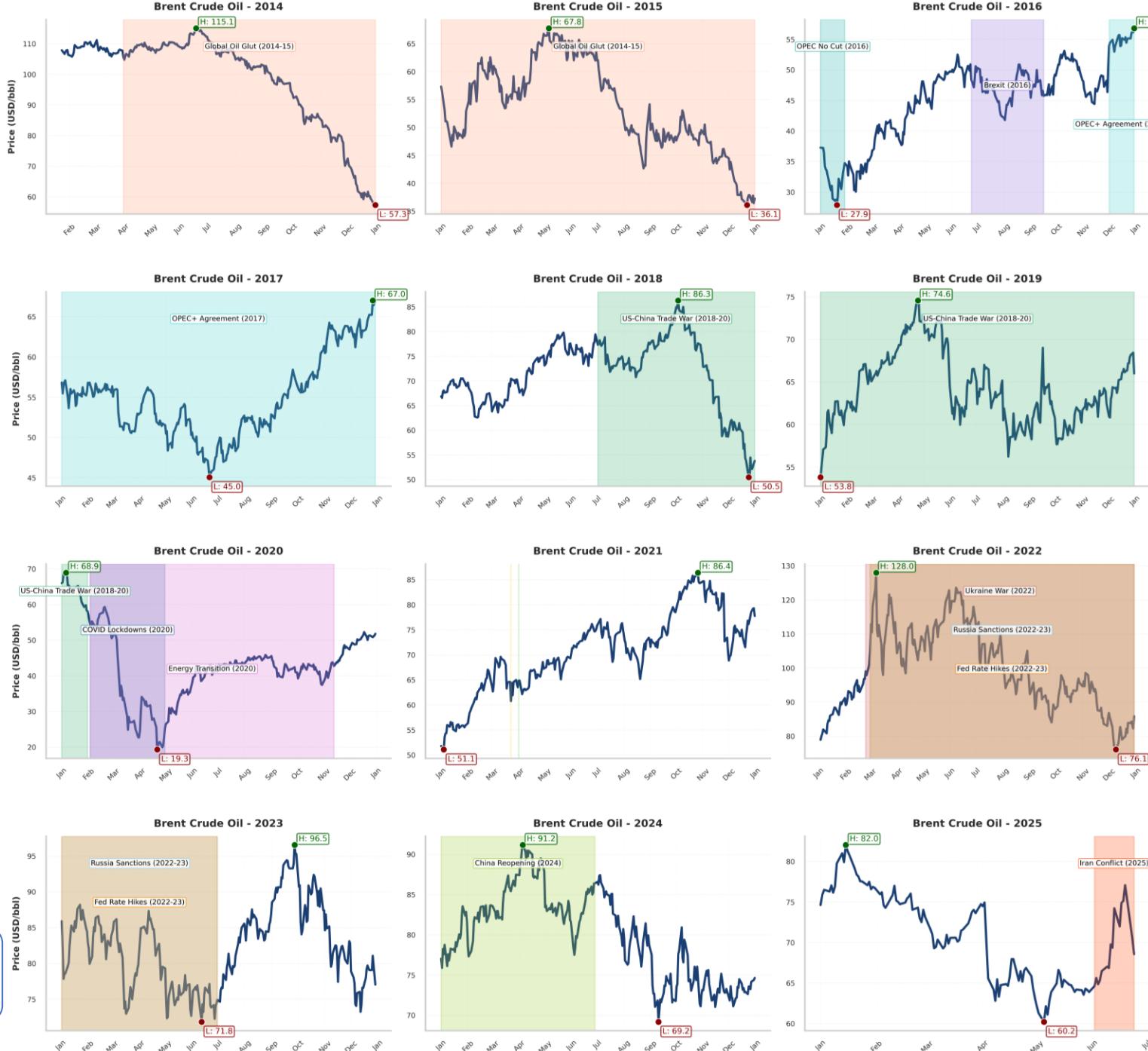
Efek dan pola tahunan
di rentang amatan
pada waktu terkait.



17 Fenomena Ekonomi ini meliputi perang dagang, kondisi buruk geopolitik negara luar, kejadian luar biasa dan transisi ekonomi hijau.



*Mungkinkah ada pola
tahunan pada data?*

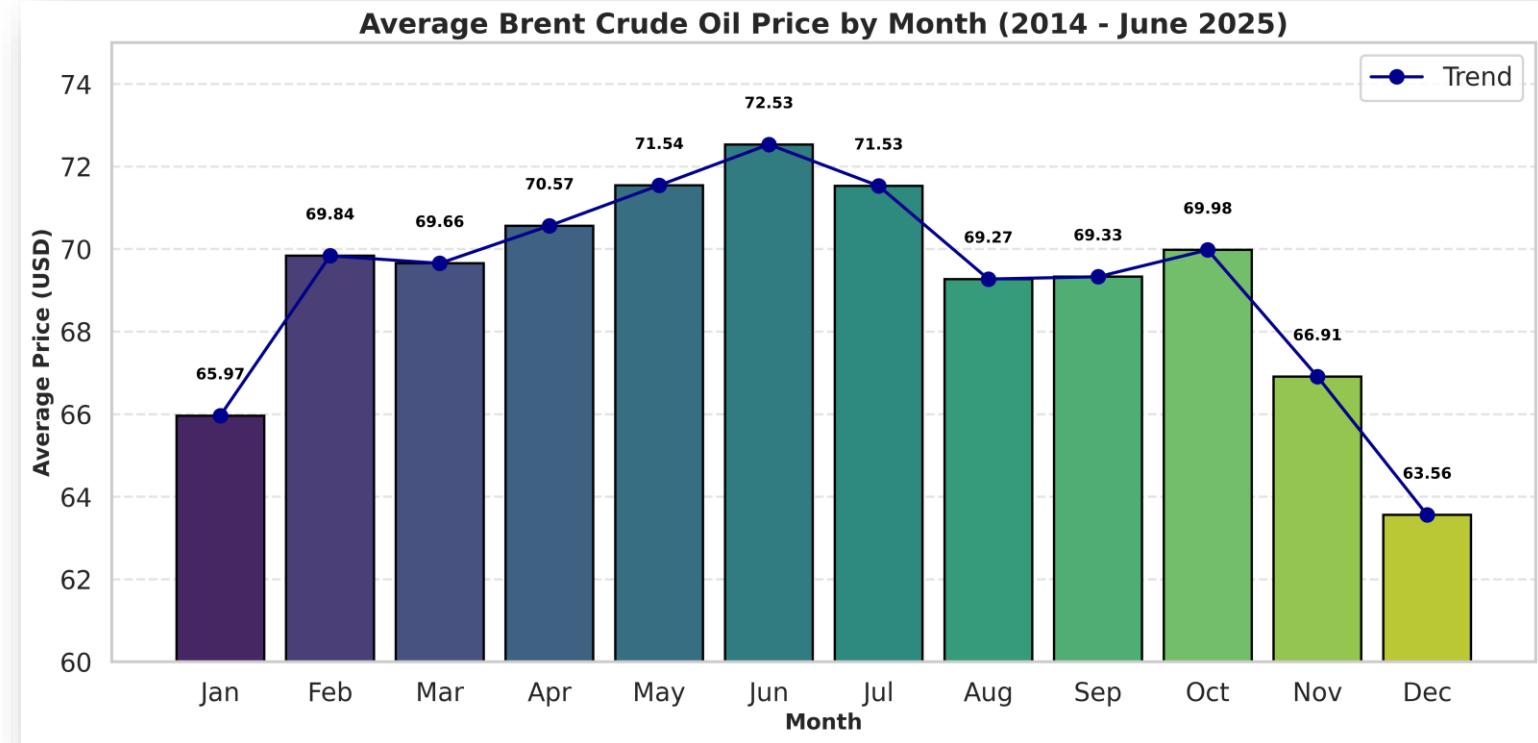


ETL dan EDA

Nampak rata-rata harga tiap bulan sangat berbeda



Gambar 3. Rata Rata Harga Bulanan Brent CPO



Indikasi tiap awal dan akhir tahun harga CPO turun, pertengahan tahun naik.



Perbedaan harga rata rata bulana merupakan insight! ?

Setiap 21 – 22 amatan
memungkinkan perubahan pola

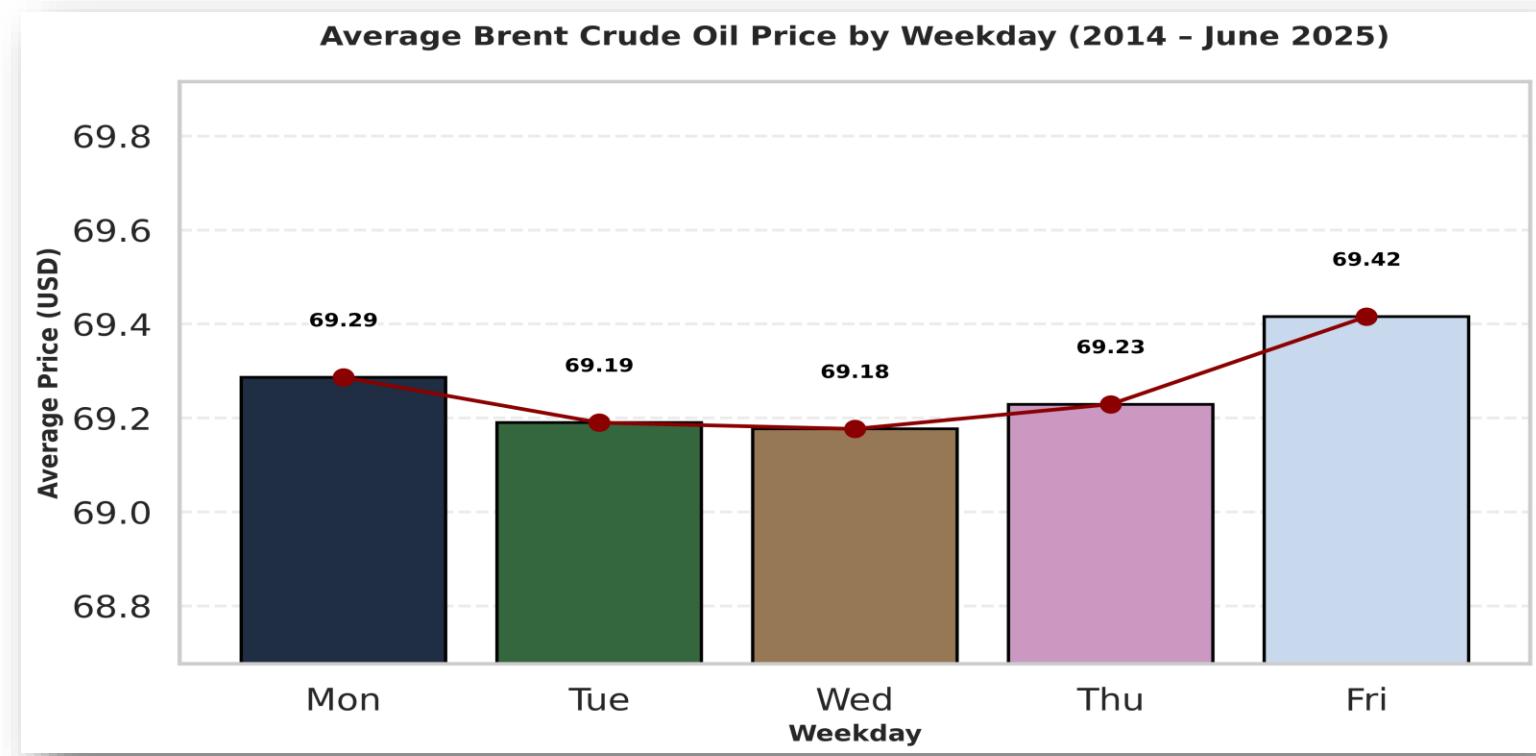


ETL dan EDA

Nampak rata-rata harga per hari relative sama



Gambar 4. Rata Rata Harian Brent CPO



Tidak terdapat pola naik dan turun serta fluktuasi pada harga rata rata harian Brent CPO.



Tidak ada beda harga rata rata harian merupakan insight!

?

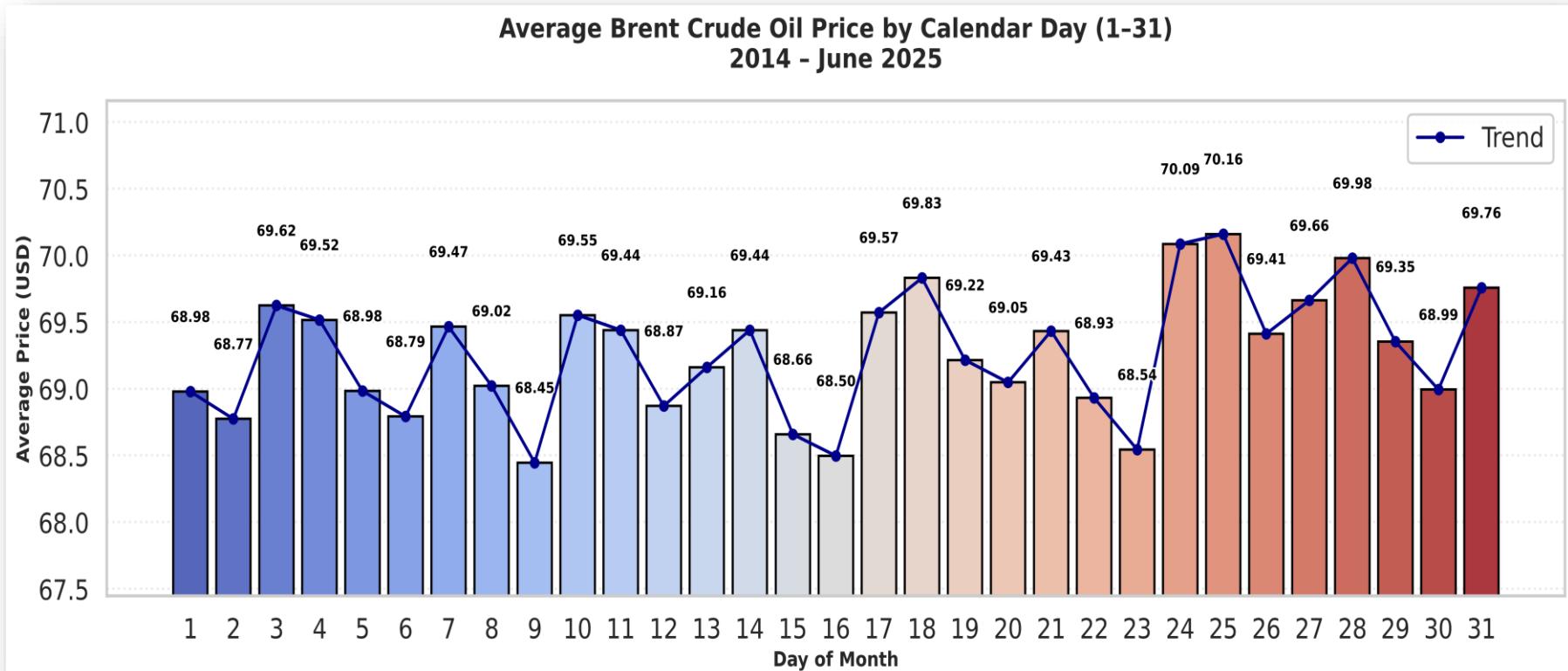
Kemungkinan tidak perlu melihat 1-2 hari amatan lain

ETL dan EDA

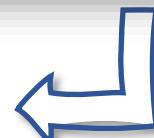
Nampak rata-rata harga per tanggal relative beda



Gambar 5. Rata Rata Tanggal Unik Brent CPO



Terdapat indikasi pola naik dan turun serta fluktuasi pada harga rata rata tanggal unik Brent CPO.



Perubahan harga antar beberapa tanggal merupakan insight! ?

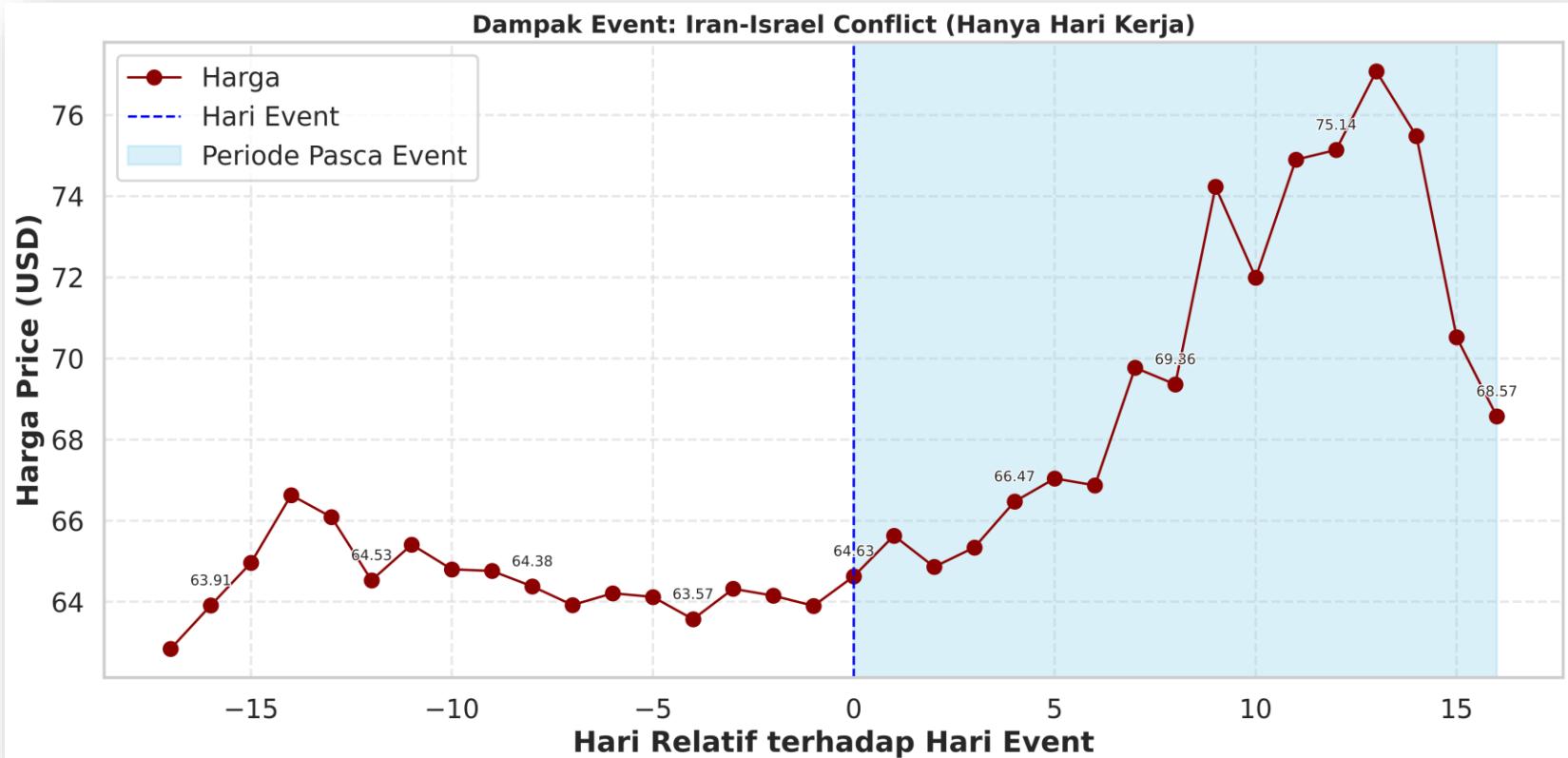
Kemungkinan terdapat pola ulangan tiap berapa hari/minggu

ETL dan EDA

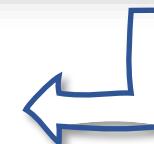
Uji T dengan mengambil 17 Amatan sesudah dan sebelum kejadian.



Gambar 6. Uji T Iran vs Israel 2025



Uji T dan hasil plot kejadian menunjukkan terdapat beda rata rata harga CPO sebelum dan sesudah kejadian.



Terdapat pola kenaikan pada setelah kejadian, pertanda hal baru!



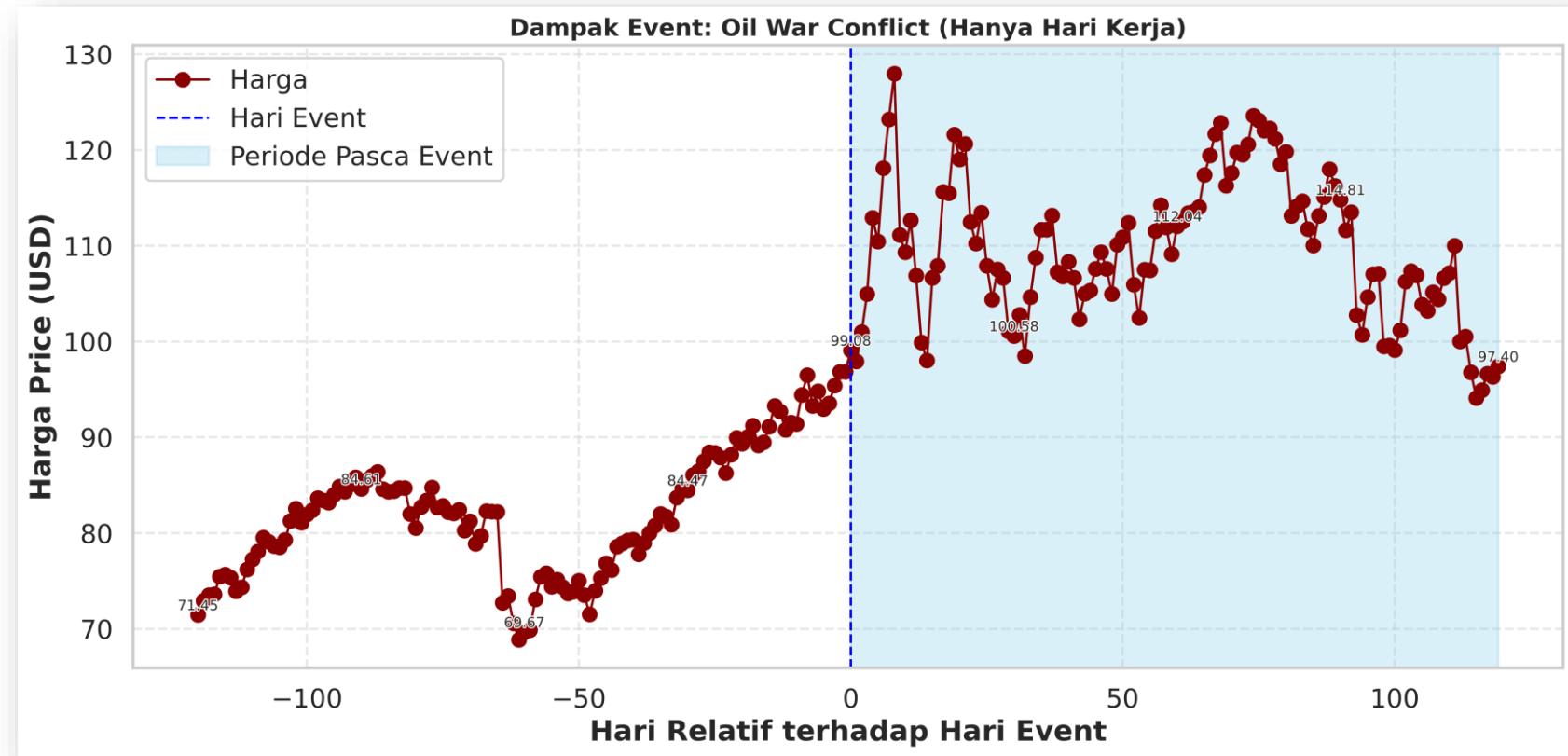
Kemungkinan terdapat pengaruh dari kejadian ini untuk Harga CPO

ETL dan EDA

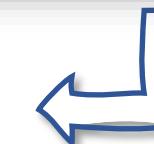
Uji T dengan mengambil 120 Amatan sesudah dan sebelum kejadian.



Gambar 7. Uji T Ukraina vs Rusia



Uji T dan hasil plot kejadian menunjukkan terdapat beda rata rata harga CPO sebelum dan sesudah kejadian.



Terdapat pola kenaikan pada setelah kejadian, pertanda hal baru!

?

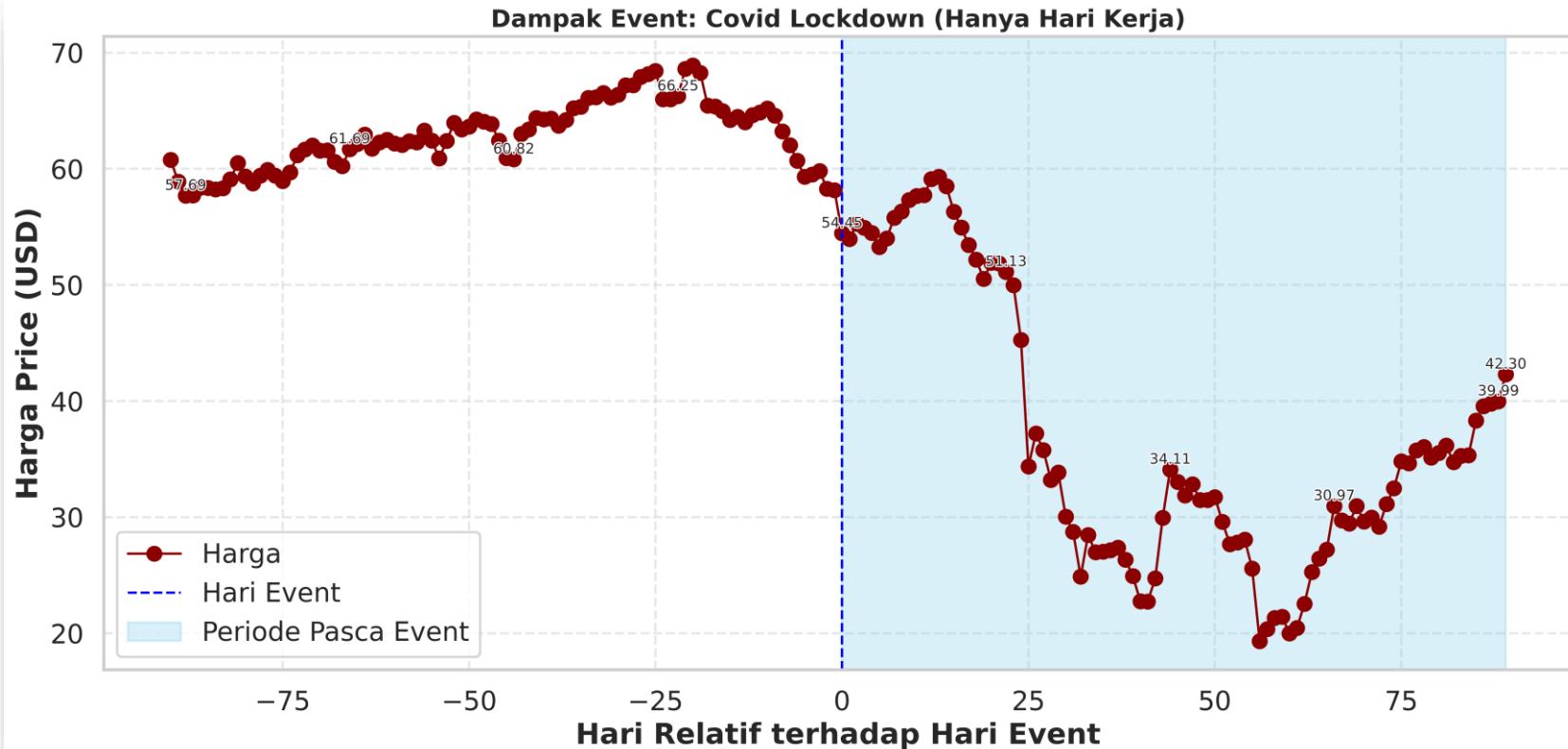
Kemungkinan terdapat pengaruh dari kejadian ini untuk Harga CPO

ETL dan EDA

Uji T dengan mengambil 90 Amatan sesudah dan sebelum kejadian.



Gambar 8. Uji T Pembatasan Covid



Uji T dan hasil plot kejadian menunjukkan terdapat beda rata rata harga CPO sebelum dan sesudah kejadian.



Terdapat pola penurunan pada setelah kejadian, pertanda hal baru!



Kemungkinan terdapat pengaruh dari kejadian ini untuk Harga CPO



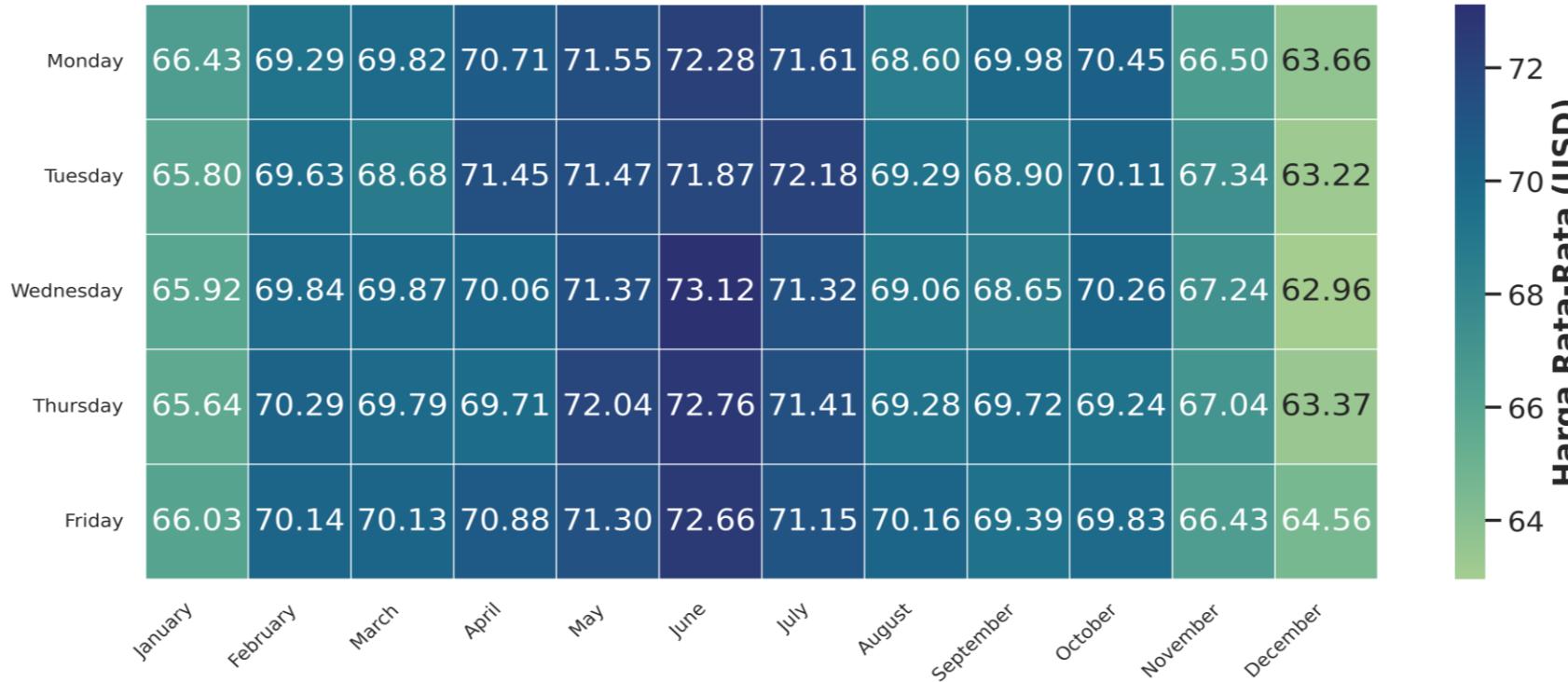
ETL dan EDA

Tampak interaksi Bulan dan Hari tidak cukup menjelaskan pengaruh harga.



Gambar 9. Interaksi Bulan dan Hari

Heatmap Harga Rata-Rata berdasarkan Hari dan Bulan



Pola musiman Bulanan Q2-Q3 cenderung meningkat, sementara Rabu-Kamis hari dengan harga rata rata tertinggi.



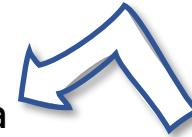
Tidak terdapatnya pola konsisten antara Hari dan Bulan pada data.



Kemungkinan tidak terdapat pengaruh dari interaksi ini

ETL dan EDA

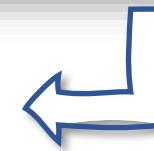
Terdapat amatan yang berada pada luar interval, tanda amatan pencilan.



Gambar 9. Rataan & S. Baku Bergerak



Volatilitas (simpangan baku bergulir) bersifat stasioner, sehingga dapat dimodelkan tanpa transformasi sementara. Harga aktual mungkin perlu pembedaan.



Perlunya pembagian menjadi konsep tahunan agar lebih jelas.

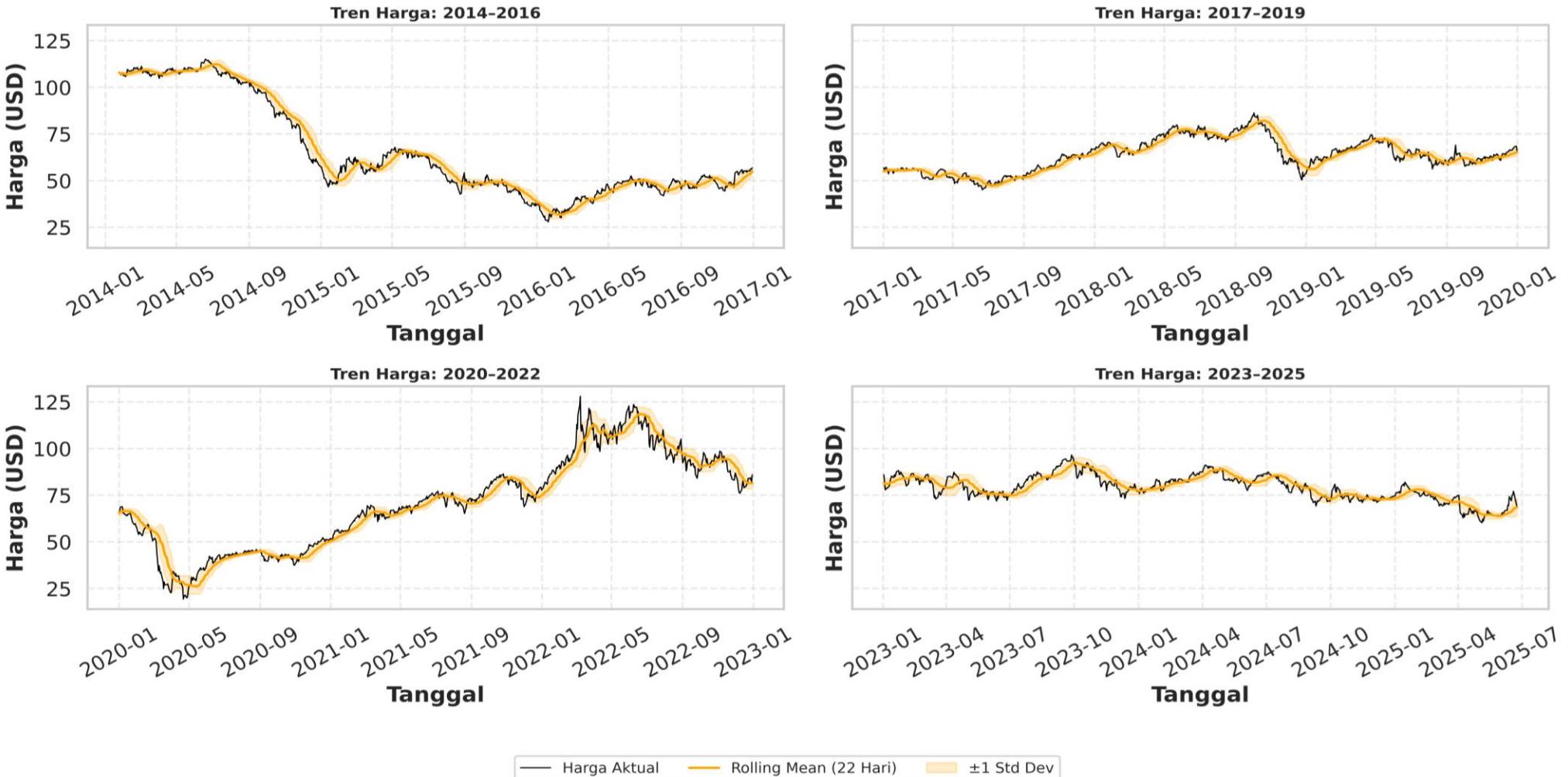
?

Penting lakukan ini, agar jelas mengenali pencilan data waktu



ETL dan EDA

Rataan dan Simpangan Baku Bergulir



Model Box-Jenkins (ARIMA dan GARCH) sulit menangkap volatilitas ekstrem ini, sehingga pendekatan hybrid (Transformer + XGBoost) lebih sesuai.



Volatilitas (2020-2022) bersifat Episodik dibanding periode lain. Pembeda dengan 22 hari dinilai cukup untuk memodelkan pola deret waktu.

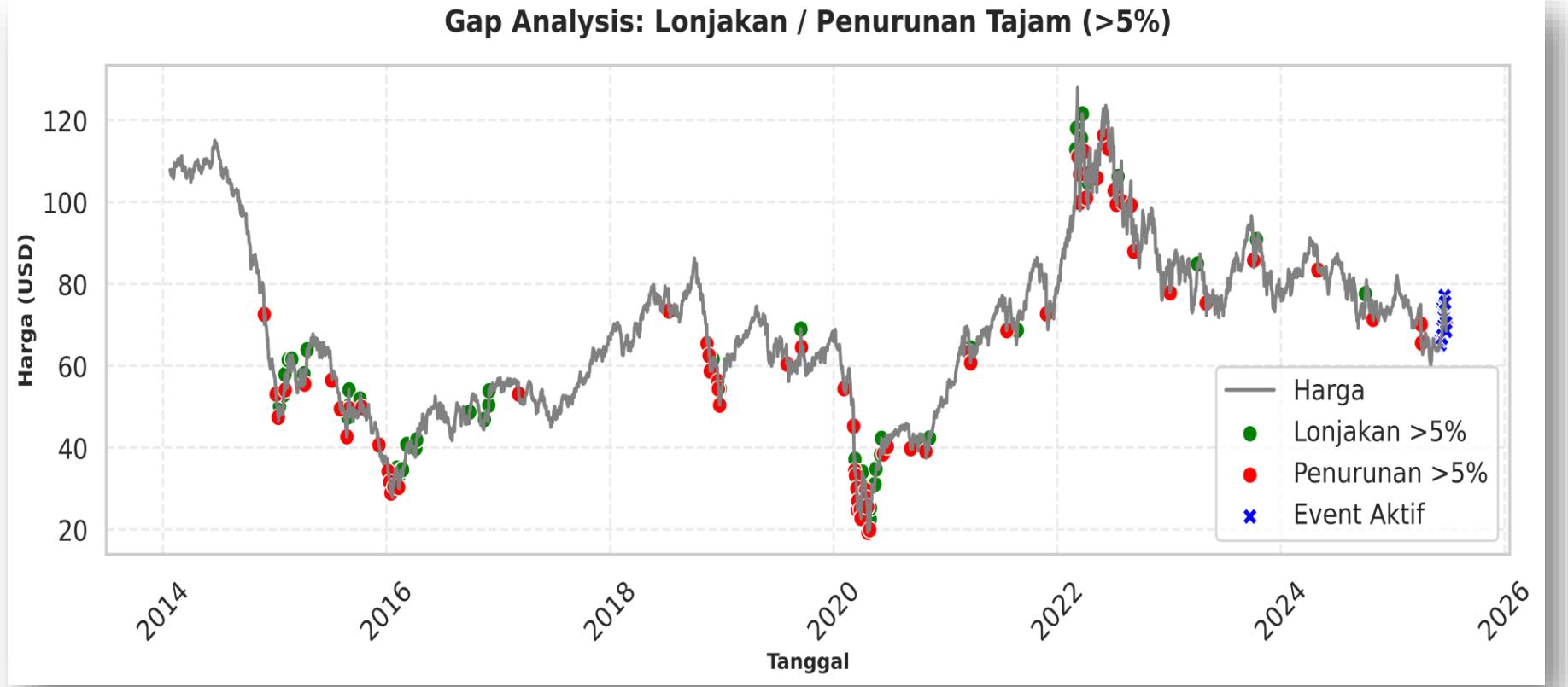


ETL dan EDA

Gap cenderung terjadi berkelompok (misal: 3-5 gap dalam 2 minggu selama dampak kejadian).



Gambar 11. Analisis Gap Ekstrem



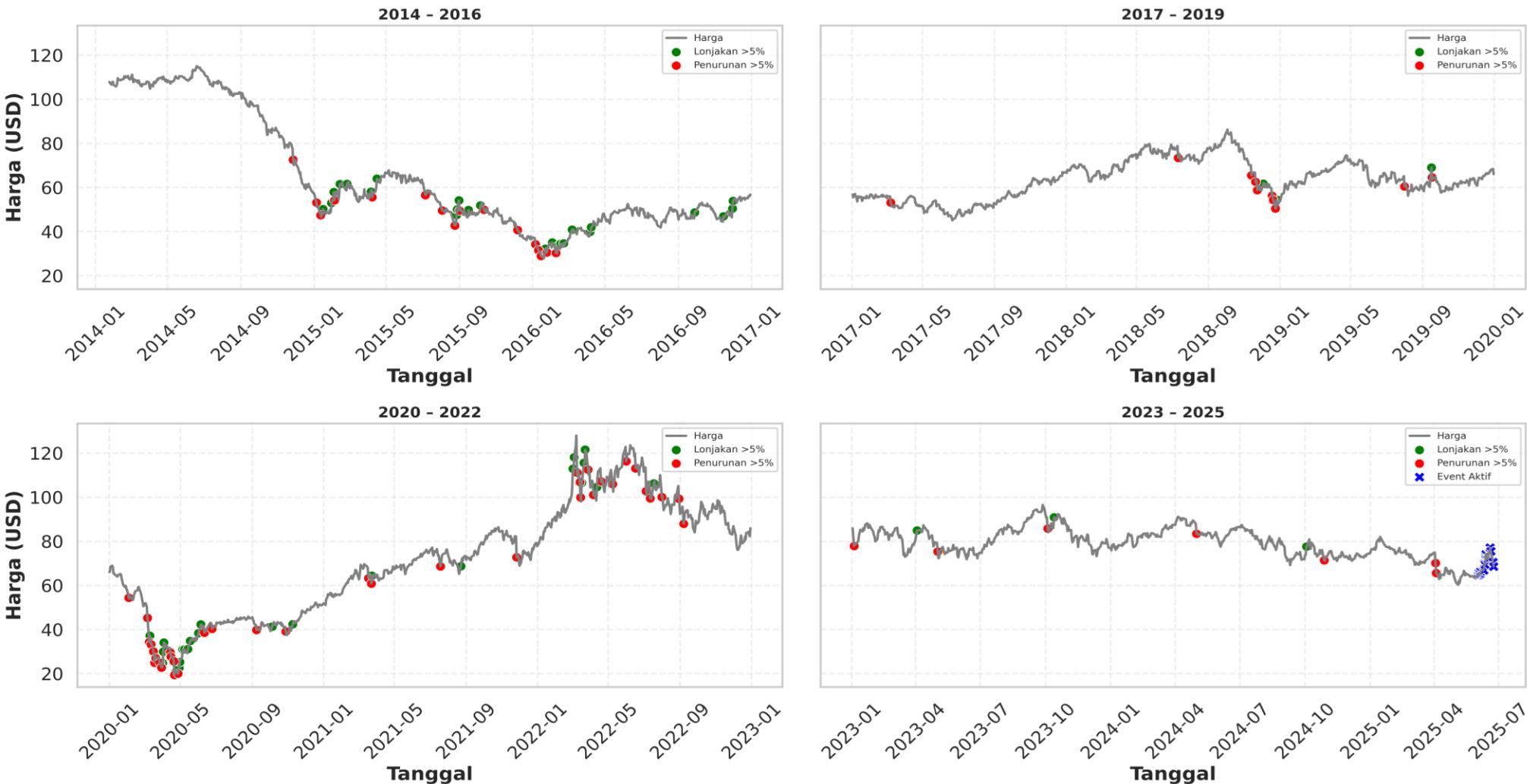
November-Desember selalu cenderung harga turun dan Mei-Juli cenderung harga naik. Artinya Gap 5% ini didominasi oleh faktor eksternal diluar model aslinya.



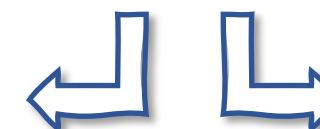
Uji Granger (Kausalitas) bersifat Taraf Nyata, apa artinya ini. ?

Informasi amatan peubah lain dapat membantu peubah lainnya

ETL dan EDA



Guna menangkap pola atau amatan gap ekstrem, perlu eksogen baru berupa intervensi 1=Gap dan 0=Tanpa Gap

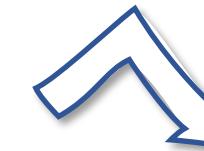


Nilai amatan Gap Ekstrem (2020-2022) cenderung lebih banyak dibandingkan dgn interval tahun lainnya. Ini juga pertanda bahwa akan ada pola musiman sendiri.



ETL dan EDA

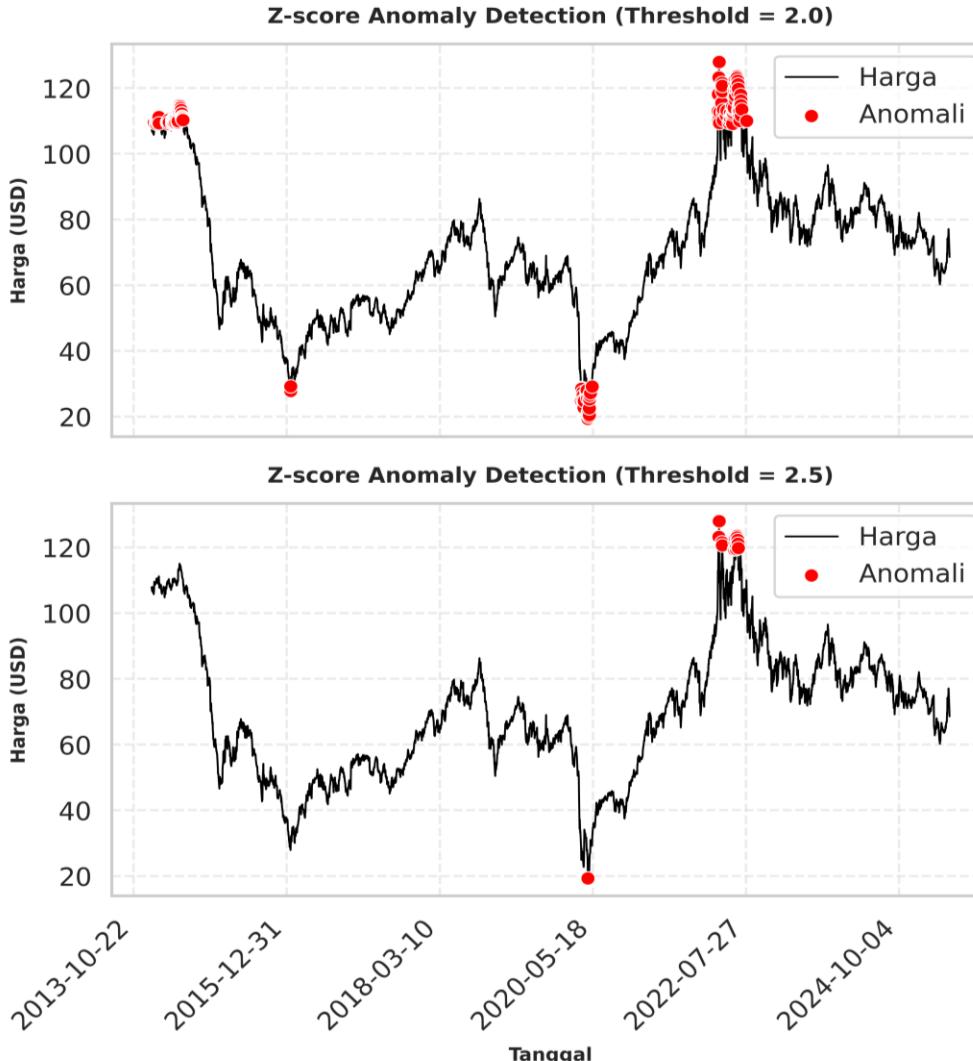
Pada sebaran normal, kita ambil nilai-Z berada pada rentang 2-3 kali meyimpang dari rata-rata.



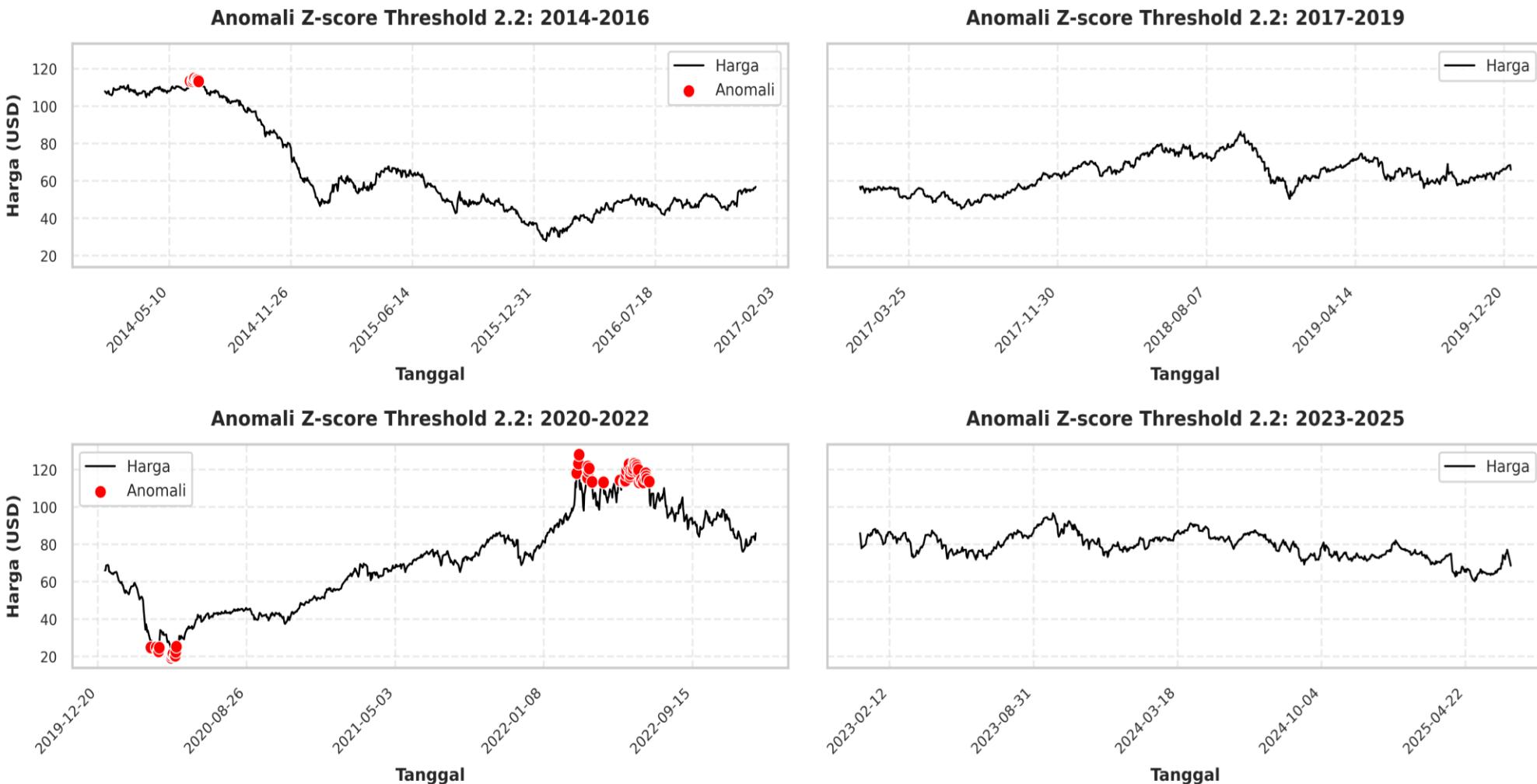
Anomali atau Penculan

$$z = \frac{x - \mu}{\sigma}$$

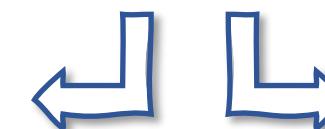
Gambar 13. Skenario Batas Z-Score



ETL dan EDA



Sebelumnya kita ambil batas nilai-Z data ini adalah 2.2 dengan pertimbangan masih menangkap pola dan nilai tidak terlalu menyimpang dari rata-rata.



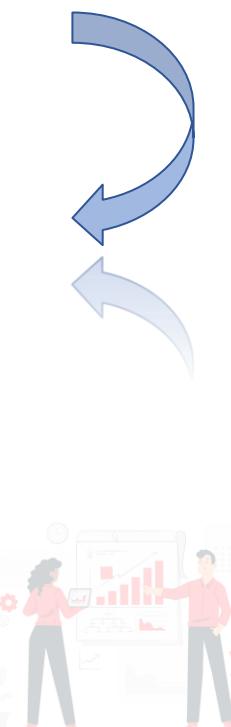
Nilai amatan Penciran Anomali (2020-2022) cenderung lebih banyak dibandingkan dgn interval tahun lainnya. Ini juga pertanda bahwa akan ada pola musiman sendiri.



ETL dan EDA

Tahapan	Sub-Bagian	Insight	Keterangan
Pola deret Waktu	Tahunan	Pola <i>time_step</i> pemodelan menggunakan kombinasi (2, 3, 21, dan 22)	<i>Time_step</i> (2,3) inisiasi dari pola pembeda tanggal unik.
	Bulanan		<i>Time_step</i> (21, 22) inisiasi dari pola pembeda bulanan.
	Harian		
	Tanggal Unik		
Kejadian	Uji-T	17 peubah kejadian dapat digunakan dalam pemodelan deret waktu	17 peubah memiliki rataan berbeda antara sebelum dan sesudah kejadian
Interaksi	Hari * Bulan	Tidak terdapat insight baru	Tidak ada struktur interaksi

Tahapan	Sub-Bagian	Insight	Keterangan
Rataan dan Simpangan Baku bergulir	Gap Ekstrem	Intervensi dummy (0,1) peubah sebagai eksogen tambahan	Penangkapan pola untuk pembelajaran mesin statistika.
	Pencilan	Intervensi dummy (0,1) peubah sebagai eksogen tambahan	Penangkapan pola untuk pembelajaran mesin statistika.



Pemodelan Mesin Pembelajaran

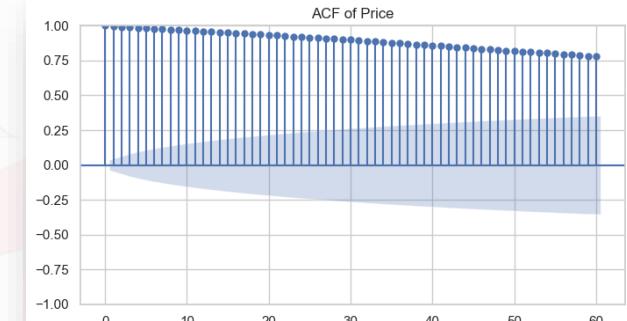
Tahap 1. ACF dan PACF

```
def plot_acf_pacf(df, lags=60):
    price = df['Price'].values
    fig, axes = plt.subplots(1, 2, figsize=(14,4))
    sm.graphics.tsa.plot_acf(price, lags=lags, ax=axes[0])
    axes[0].set_title('ACF of Price')
    sm.graphics.tsa.plot_pacf(price, lags=lags, ax=axes[1])
    axes[1].set_title('PACF of Price')
    plt.tight_layout(); plt.show()
```

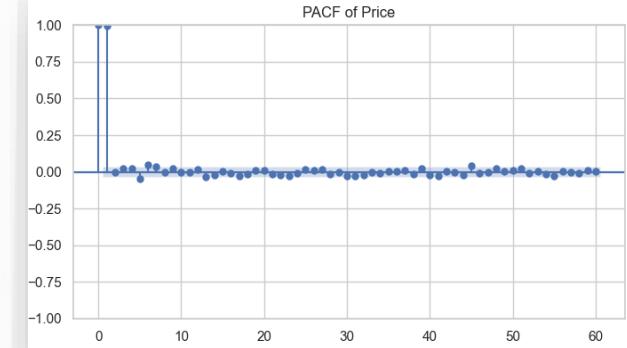
Catatan

1. Digunakan untuk membentuk plot fungsi autokorelasi dan autokorelasi parsial.
2. Lag pada waktu ke-t dapat digunakan sebagai scenario awal dalam time_step model.
3. Pada data tidak stasioner, cobalah lakukan pembedaan 1-2 kali hingga stasioner lalu buat Kembali plot autokorelasi dan autokorelasi spasial.
4. Periksa apakah deret waktu memiliki komposisi tren dan musiman, lakukan dekomposisi untuk mengetahuinya.
5. Jika menggunakan model Box-Jenkins, wajib melakukan ini.

Plot ACF



Plot PACF





Pemodelan Mesin Pembelajaran

Tahap 2. Pemodelan Fitur Waktu

```
def load_and_fe(path, lags=[7,22], mas=[7,22],
fourier=[5,21]):
    df = pd.read_csv(path, parse_dates=['date'],
index_col='date').sort_index()
    # lag features
    for lag in lags:
        df[f'lag_{lag}'] = df['Price'].shift(lag)
    # moving averages
    for w in mas:
        df[f'ma_{w}'] = df['Price'].rolling(w).mean()
    # fourier terms
    t = np.arange(len(df))
    for p in fourier:
        df[f'sin_{p}'] = np.sin(2*np.pi*t/p)
        df[f'cos_{p}'] = np.cos(2*np.pi*t/p)
    df.dropna(inplace=True)
    return df
```

Catatan

1. Fitur “lag” berfungsi melihat pola perubahan dan predictor lampau
2. Fitur “Moving Average” berfungsi menangkap tren local pada data yang berubah
3. Fitur “Sin/Cos” berfungsi untuk menangkap dan evaluasi pola yang terjadi pada waktu

\downarrow

$$lag_k(t) = Peubah(t - k)$$

$$MA_w(t) = \frac{1}{w} \sum_{i=0}^{w-1} Peubah(t - i)$$

$$\sin_p(t) = \sin\left(\frac{2\pi t}{p}\right)$$

$$\cos_p(t) = \cos\left(\frac{2\pi t}{p}\right)$$

\downarrow

Fitur waktu (lag7, lag_22,
ma_22 sin_22, cos_22)



Pemodelan Mesin Pembelajaran

Tahap 3. Pembagian Data

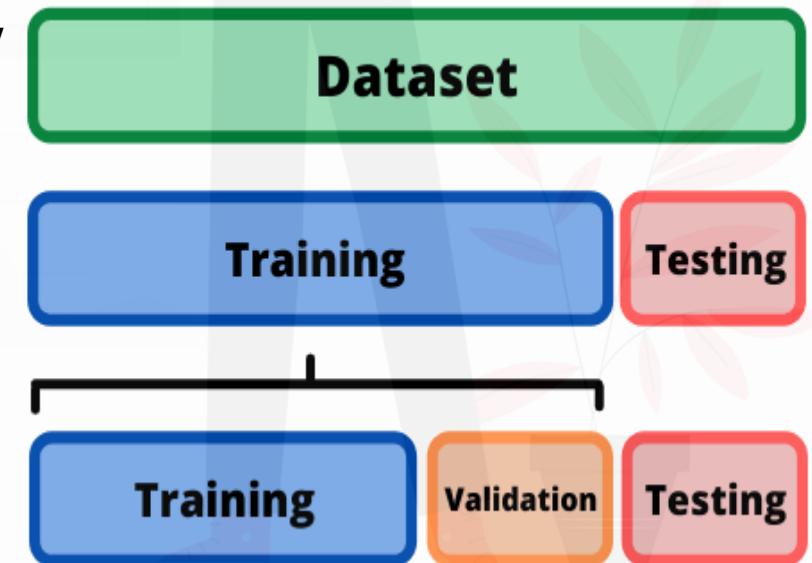
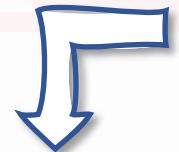
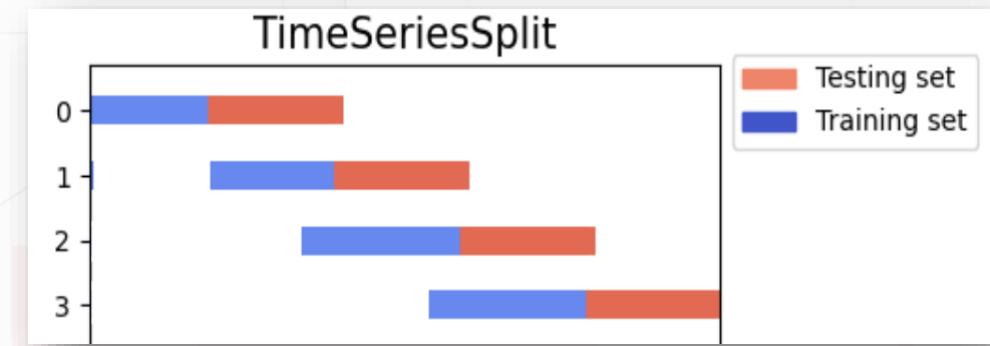
```
def train_test_split_time(df, test_size=0.2):  
    idx = int(len(df)*(1-test_size))  
    return df.iloc[:idx], df.iloc[idx:]
```

Catatan

1. Membagi data latih dan data uji menjadi 80:20 dari data asli.
2. Pada kajian atau pemodelan ini belum dilakukan pembagian waktu berdasar jendela bergerak (*sliding window*) dan jendela membesar (*expanding window*)

Perlunya pembagian data secara sekuensial pada deret waktu!

Data sekuensial dan tidak boleh diacak karena ini deret waktu





Pemodelan Mesin Pembelajaran

Tahap 4. Preprocessor, Jendela, Pembalikan Skala

Atribut	Fungsi	Luaran
<code>__init__(time_steps=1)</code>	Inisialisasi parameter <code>time_steps</code> dan dua jenis skala (<i>MinMax</i> & <i>StandardScaler</i>)	Objek <code>preprocessor</code> dengan 2 <code>scaler</code>
<code>fit_transform()</code>	Menskalakan target () dan fitur lainnya	<code>y_s</code> : target diskalakan, <code>X_s</code> : fitur diskalakan
<code>transform()</code>	Transformasi data baru dengan skala dari data latih	<code>y_s</code> , <code>X_s</code> (hasil transform, tanpa fit)
<code>create_sequences(y_s, X_s)</code>	Membuat urutan (sequence) dari data diskalakan	<code>Xp</code> : lag target, <code>Xe</code> : lag fitur, <code>Ys</code> : target t
<code>inv_y(y_s)</code>	Mengembalikan hasil prediksi ke skala asli	<code>y</code> : nilai prediksi dalam skala asli

Tujuan: Mengubah data asli menjadi urutan terstandar untuk digunakan sebagai input model *deep learning* berbasis deret waktu dan transformasi balik model ke skala aslinya.



Scaling Fitur dan Target

Transformasi Data

Jendela Deret Waktu
(*Sequencing*)

Pembalikan Skala Data



`n_sample, time_steps,`
`n_features`

Pemodelan Mesin Pembelajaran

Tahap 4. Preprosessor, Jendela, Pembalikan Skala

Catatan

1. LSTM merupakan 3 dimensi sehingga perlu transformasi data menjadi bentuk 3 dimensi.
2. Standarisasi dan Normalisasi – Jendela Data (Sequence) – Pembalikan skala data adalah inti dari tahap 4.
3. Scaler data harus fit atau pada data latih.

1

Min-Max Scalling

$$y^s = \frac{y - y_{min}}{y_{max} - y_{min}}$$

Standarisasi

$$x^s = \frac{x - \mu}{\sigma}$$

Konsep Jendela Deret Waktu

$$\begin{aligned} Xp_i &= \{y_{i-ts}, \dots, y_{i-1}\}, \text{dimensi} = (n, ts, 1) \\ Xe_i &= \{X_{i-ts}, \dots, X_{i-1}\}, \text{dimensi} = (n, ts, d) \\ Ys_i &= y_i, \text{dimensi} = (n,) \end{aligned}$$

Keterangan dimana 1) Xp : urutan nilai amatan Ys; 2) Xe : urutan fitur dan; 3) Ys : output/prediksi.

2



Konsep Pembalikan Skala

$$\hat{y} = y^s \cdot (y_{max} - y_{min}) + y_{min}$$

Semua array harus punya dimensi yang sama

$$Xp.shape[0] = Xe.shape[0] = Ys.shape[0]$$

3





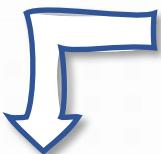
Pemodelan Mesin Pembelajaran

Tahap 5. Positional Encoding Layer - Transformer

Atribut	Fungsi	Luaran
<code>_init__(length, depth)</code>	Menginisialisasi posisi dan kedalaman dimensi	tensor <code>self.pe</code> (1, length, d)
<code>angle[:, 0::2]</code>	Aplikasi fungsi sinus untuk dimensi genap	Nilai sinus musiman
<code>angle[:, 1::2]</code>	Aplikasi fungsi cosinus untuk dimensi ganjil	Nilai cosinus musiman
<code>angle</code>	Menghitung sudut berdasarkan posisi dan dimensi	Matriks angle

Catatan

- Positional Encoding (PE) memungkinkan Transformer memodelkan musiman atau siklus dalam data deret waktu.
- PE dapat digunakan untuk peramalan multistep karena pola sinus dan cosinus tetap terdefinisi ulangannya di amatan ke depannya.



Indeks Genap

$$PE_{(p,2i)} = \sin\left(\frac{p}{10000^{2i/D}}\right)$$

Indeks Ganjil

$$PE_{(p,2i+1)} = \cos\left(\frac{p}{10000^{2i/D}}\right)$$



Ind. Genap sbg pola siklik lambat, Ind. Ganjil sbg pola siklik cepat.



Pemodelan Mesin Pembelajaran

Tahap 6. Transformer - BiLSTM Model

Atribut	Fungsi	Luaran
PositionalEncoding(...)	Menambahkan informasi posisi temporal ke input	tensor dengan encoding posisi
Bidirectional(LSTM(...))	Menangkap pola urutan dua arah (forward & backward)	Sequence tensor (ts, units*2)
LayerNormalization()	Menstabilkan distribusi layer setelah LSTM dan setiap blok	tensor yang dinormalisasi
MultiHeadAttention(...)	Mencari hubungan antar waktu (self-attention)	Tensor atensi
tf.reduce_mean(x, axis=1)	Merata-ratakan output sequence jadi 1 representasi per sample	Tensor (None, units)
Add() + LayerNormalization()	Residual connection + normalisasi ulang	Arsitektur stabil: seperti Transformer encoder block

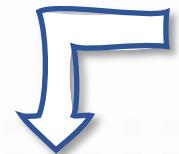
Tujuan: Membangun arsitektur Transformer-BiLSTM guna menangkap pola historis dan hubungan antar waktu juga menghasilkan model prediktif yang kekar terhadap perubahan pola dan peubah eksogen.

Pemetaan Ukuran dan Ekstraksi Pola Waktu

Agregasi Global dan Prediksi Hasil

Catatan

1. *Windowing (time_steps)* harus konsisten disesuaikan dengan pola pada data.
2. Transformer cocok dan dapat menangkap pola deret waktu jangka Panjang.





Pemodelan Mesin Pembelajaran

Tahap 7. Arsitektur Transformer - BiLSTM

Atribut	Fungsi	Luaran
build_transformer_bilstm()	Membangun model dengan parameter yang sedang diuji	Model neural network dengan parameter berubah ubah.
optuna.create_study()	Membuat studi hyperparameter tuning	Objek study yang memminimumkan MAPE dan MAE
study.optimize(..)	Menjalankan proses pencarian hyperparameter terbaik	Optuna melakukan eksplorasi dan eksplorasi ruang parameter
study.best_params	Mengambil kombinasi terbaik setelah seluruh percobaan	Kamus parameter optimal

Tujuan: Menentukan kombinasi hyperparameter terbaik untuk model dengan cara mengoptimasi kinerja validasi MAPE dan MAE, sehingga model memiliki akurasi prediksi yang tinggi dan lebih stabil.

Pendefinisian Fungsi
(lstm_units, dropout, lr,dll)

Split Data dan Latih Model

Optimasi dan Tuning Parameter Terbaik

Catatan

1. Gunakan *trials* yang banyak agar mencari kombinasi parameter terbaik tetapi tetap perhatikan apakah model nantinya akan *overfitting* atau tidak.





Pemodelan Mesin Pembelajaran

Tahap 8. Pelatihan Model Dasar Berbasis Validasi

Atribut	Fungsi	Luaran
pre.fit_transform()	Melakukan standarisasi fitur dan normalisasi target	y_s, X_s (MinMax Scaler untuk target, Standard Scaler untuk fitur eksogen)
pre.create_sequences(...)	Membentuk input 3D untuk model time series	Xp, Xe, y_seq (lag target, lag fitur, dan target saat ini)
build_transformer_bilstm()	Membangun model dengan parameter terbaik dari tuning	Model mengikuti hasil tuning (jumlah layer, dropout, dsb.)
model.fit(...)	Melatih model dengan data train dan validasi internal	Model latih dengan EarlyStopping dan pengaturan learning rate adaptif (Reduce LROn)

Tujuan: Melatih model Transformer-BiLSTM menggunakan seluruh data latih dengan validasi internal berdasarkan urutan waktu, menggunakan parameter terbaik hasil tuning, untuk menghasilkan model yang optimal dan generalisasi baik.

Preprocessing dan Windowing

Split Data Internal untuk Validasi Model

Bangun dan Latih Model dari Parameter Terbaik

Catatan

1. Gunakan *fit_transform* hanya di data latih. Data test gunakan *transform* agar tidak terjadi *data leakage*
2. Kombinasi *early stopping* dan *reduce LR* adalah cara terbaik mencegah model terlalu fit ke data latih





Pemodelan Mesin Pembelajaran

Tahap 9. Fitur Engineering pada Galat

Atribut	Fungsi	Luaran
pre.transform(df)	Menstandardisasi data baru menggunakan skala dari data latih	y_s, X_s
pre.create_sequences(...)	Membentuk input sekuensial	Xp, Xe, y_seq (Input 3D untuk prediksi model)
res_s = y_seq - y_base_s	Menghitung galat prediksi	res_s sebagai target model residual
.rolling(5).mean(), std(), ...	Menghitung fitur jedela bergulir dari galat	Fitur rm, rs, skew, kurt (statistik basis galat 5 amatan sebelum)
last_x = Xe[:, -1, :]	Mengambil fitur eksogen terakhir	last_x sebagai fitur prediktor untuk model residual
last_y = Xp[:, -1, 0]	Mengambil harga terakhir dari window prediktor	Last_y sebagai salah satu fitur tambahan residual

Tujuan: Menghasilkan prediksi awal dan menghitung galat, lalu membentuk kumpulan fitur baru berbasis galat prediksigauna pemodelan lanjutan (residual modeling / stacking).

Preprocessing dan Windowing

Split Data Internal untuk Validasi Model

Bangun dan Latih Model dari Parameter Terbaik

Catatan

1. Gunakan *fit_transform* hanya di data latih. Data test gunakan transform agar tidak terjadi *data leakage*
2. Kombinasi *early stopping* dan *reduce LR* adalah cara terbaik mencegah model terlalu fit ke data latih





Pemodelan Mesin Pembelajaran

Tahap 10. Pemodelan Galat dan Fitur Penting

Atribut	Fungsi	Luaran
param_dist	Menentukan pencarian ruang hyper-parameter XGBoost	Parameter: tree depth, learning rate, regularisasi, subsample, dll.
RandomizedSearch CV(...)	Tuning model dengan pencarian acak terhadap parameter	Objek GridSearch dengan 3 – 5-fold validasi silang
search.best_estimator_	Mengambil model XGBoost terbaik	XGBRegressor untuk prediksi galat
get_booster().get_score()	Mengambil importance dari booster internal XGBoost	Rangkuman <i>gain</i> per fitur (Gain = peningkatan rata-rata dari tiap fitur dalam split tree)

Tujuan: Melatih model koreksi residual berbasis XGBoost untuk menangkap pola non-linear yang belum dipelajari oleh model dasar lalu mengidentifikasi fitur penting dalam menjelaskan galat model dasar melalui visualisasi gain fitur penting.

Pelatihan Model Galat (XGBoost)

Evaluasi dan Seleksi Model Terbaik

Visualisasi Fitur Penting

Catatan

1. Kontribusi fitur penting XGBoost dihitung dari peningkatan informasi pada tiap *split tree*. Meskipun XGBoost kekar, tetap gunakan validasi silang dan regularisasi agar tidak *overfitting* pada model.
- 2.





Pemodelan Mesin Pembelajaran

Tahap 11. Evaluasi Model dan Plot Hasil

Atribut	Fungsi	Luaran
get_residuals_and_f eats()	Menghitung prediksi base, galat, dan gabungannya	y_seq, y_base_s, res_s, X_res
res_m.predict()	Memperkirakan galat dengan model XGBoost	yres_tr_s, yres_te_s (galat tambahan dari model galat)
yh = yb + yres	Hybrid = prediksi dasar + koreksi galat	yh_tr_s, yh_te_s (prediksi gabungan)
calc_metrics()	Menghitung metrik evaluasi model	MAPE, RMSE, MAE, R ²

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right| \cdot 100$$

$$MAE = \frac{1}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right|$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}$$

$$\hat{y}_{hybrid} = \hat{y}_{T.BiLSTM} + \hat{e}_{galat}$$

Tujuan: Mengevaluasi dan memvisualisasikan performa model hybrida pada data latih dan uji, baik dari segi akurasi numerik (MAPE, RMSE, MAE, R²) maupun secara visual terhadap data aktual.

Prediksi Data Latih dan Data Uji

Pembalikan Skala dan Evaluasi Model

Visualisasi Plot Hasil

Catatan

1. Pada data dengan skala kecil, MAPE bisa eksplosif jadi sebaiknya periksa juga MAE. Validasi dengan visual karena kemungkinan nilai evaluasi baik tapi bisa jadi plot telat menangkap pola perubahan.
- 2.



Pemodelan Mesin Pembelajaran

Tahap 12. Peramalan dan Komparasi Model

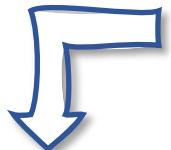
Atribut	Fungsi	Luaran
base_m.predict([Xp, Xe])	Prediksi model dasar	y_bs Prediksi dasar pada data penuh
get_residuals_and_feats(...)	Menghitung fitur galat untuk seluruh data	Input untuk model galat XGBoost
rolling std dev	Hitung standar deviasi galat dengan jendela bergulir	Digunakan untuk menghitung selang kepercayaan 95%
CI (Confidence Interval)	Hitung batas bawah dan atas CI 95% dari prediksi hybrid	Selang interval untuk prediksi model.

$$\text{Bawah} = \hat{y}_{hybrid} - 1.96 \cdot \sigma$$

$$\text{Atas} = \hat{y}_{hybrid} + 1.96 \cdot \sigma$$

1.96 karena menggunakan CI 95% (Sebaran Normal)

Tujuan: Melakukan peramalan pada keseluruhan data dengan model hybrida, menghitung interval kepercayaan 95% (batas atas dan batas bawah) serta memvisualisasikan hasil peramalan beserta ketidakpastian prediksi.



Persiapan Data dan
Prediksi Hybrid

Pembalikan Skala dan Hitung
Selang Kepercayaan 95%

Visualisasi Plot Waktu

Catatan

1. Asumsikan bahwa galat model menyebar normal.
2. Prediksi harus dikembalikan ke skala asli agar hasil dapat diinterpretasi secara nyata.



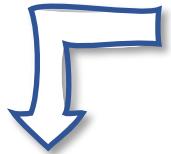


Pemodelan Mesin Pembelajaran

Pipeline Model Hybrida *Transformer BiLSTM - XGBoost*

Atribut	Fungsi	Luaran
TimeSeriesPreprocessor	Inisialisasi preprocessor untuk skala dan urutan	persiapan data untuk model deret waktu
preproc.create_sequences(y_s, X_s)	Membuat urutan data untuk model input	Input model Transformer-BiLSTM
get_residuals_and_feats(base_model, train_df, preproc)	Menghitung galat dan fitur residual	Fitur tambahan untuk pemodelan galat
tune_base(Xp, Xe, y, trials=n)	Hyperparameter tuning model dasar dengan Optuna	Mendapatkan konfigurasi parameter optimal
forecast_full_and_save(base_model, xgb_model, df_full, preproc)	Prediksi penuh data dan visualisasi peralaman dengan CI	Peramalan data dengan selang kepercayaan 95%

Tujuan: Membuat pipeline pemodelan deret waktu dimulai dari pemrosesan data, eksplorasi pola lag, pelatihan model Transformer-BiLSTM, pelatihan model galat (XGBoost), evaluasi performa model, hingga peramalan beserta visualisasi dan interpretasi fitur.



Model Transformer - BiLSTM

Model Koreksi Galat XGBoost

Hasil dan Evaluasi Model Hybrida

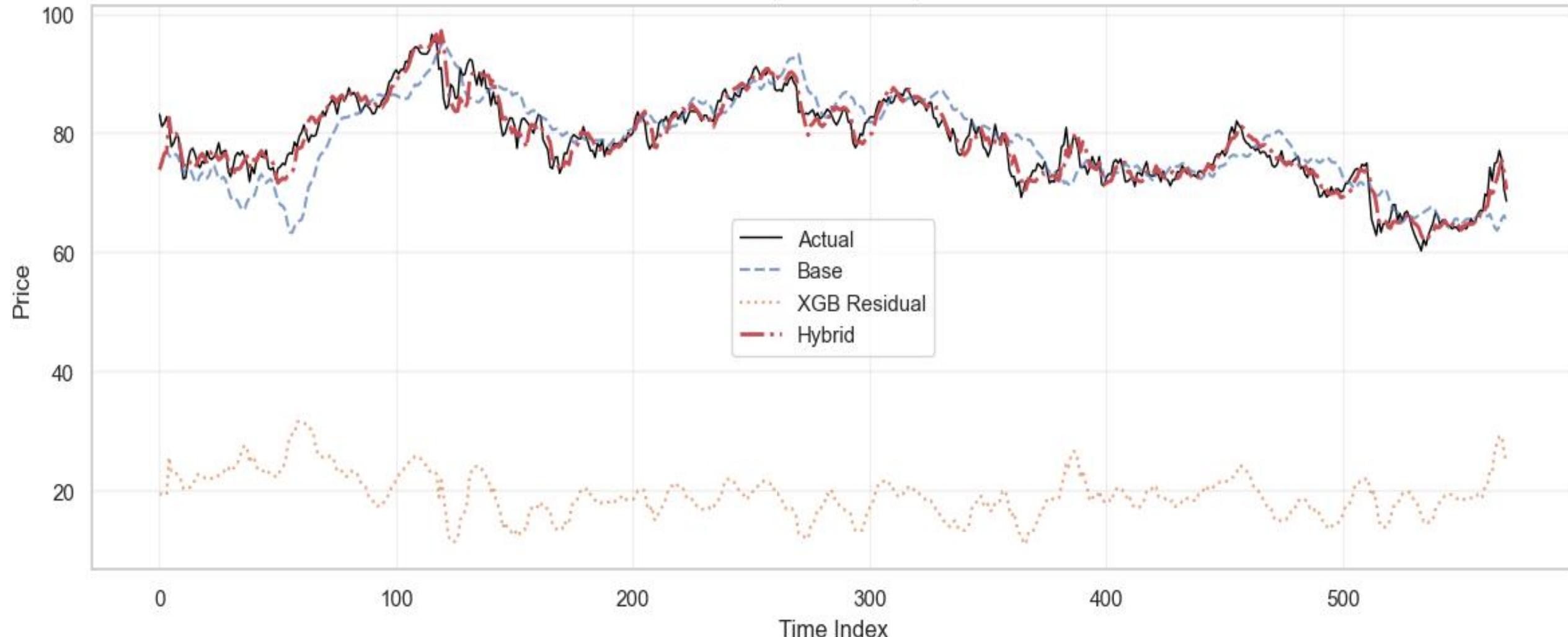
Catatan

1. Gunakan metrik MAPE, MAE, RMSE dan visualisasi untuk deteksi overfitting dan performa model.
2. Pemilihan arsitektur time_step memengaruhi input model (dapat melihat ACF/PACF atau dari pola waktu di EDA).



Plot dan Metrik Evaluasi Data Uji

Test Set Comparison
MAPE=1.83%, RMSE=1.87, MAE=1.43

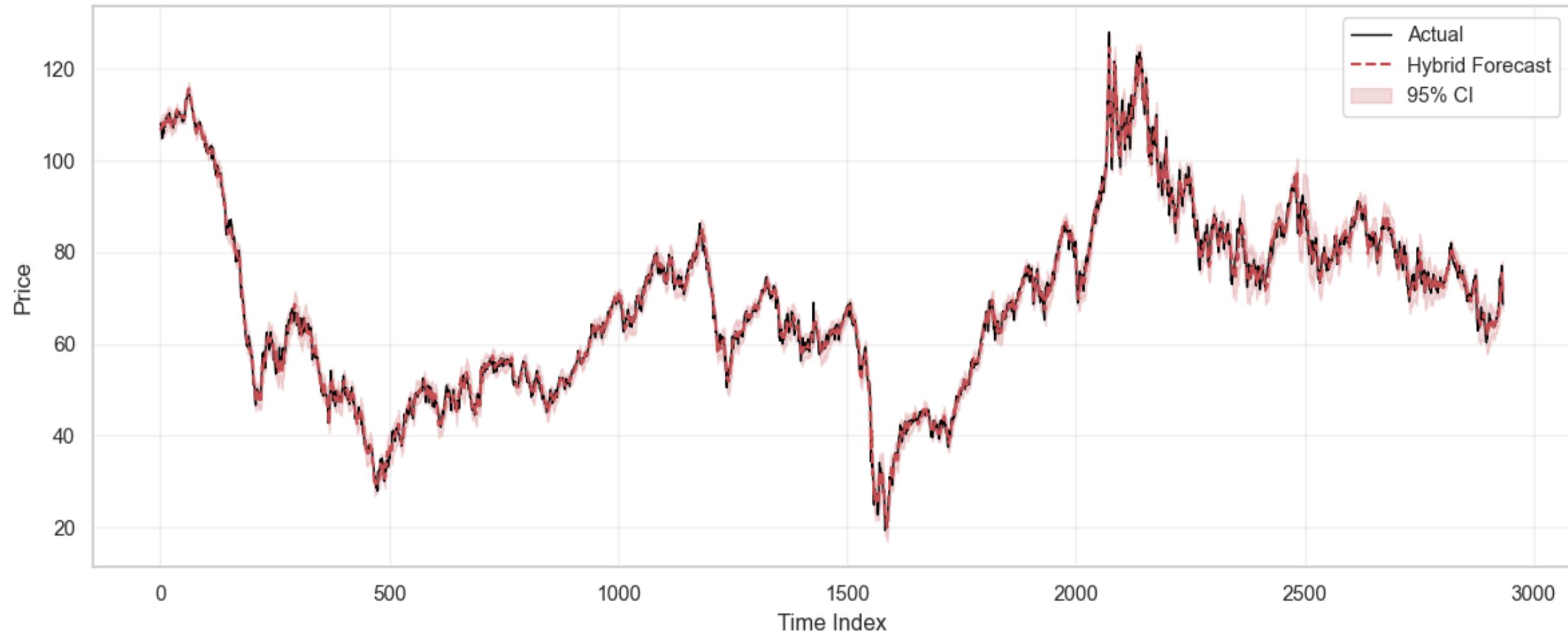


Diberikan visualisasi untuk 1 data actual dan 3 pendekatan berbeda dalam prediksi data actual (berd. garis pada legenda)



Model Hybrid dengan kombinasi Transformer BiLSTM dan koreksi galat XGBoost memberikan hasil prediksi yang cukup baik dan mengikuti pola data actual.

Plot Peramalan Data Asli + CI 95%

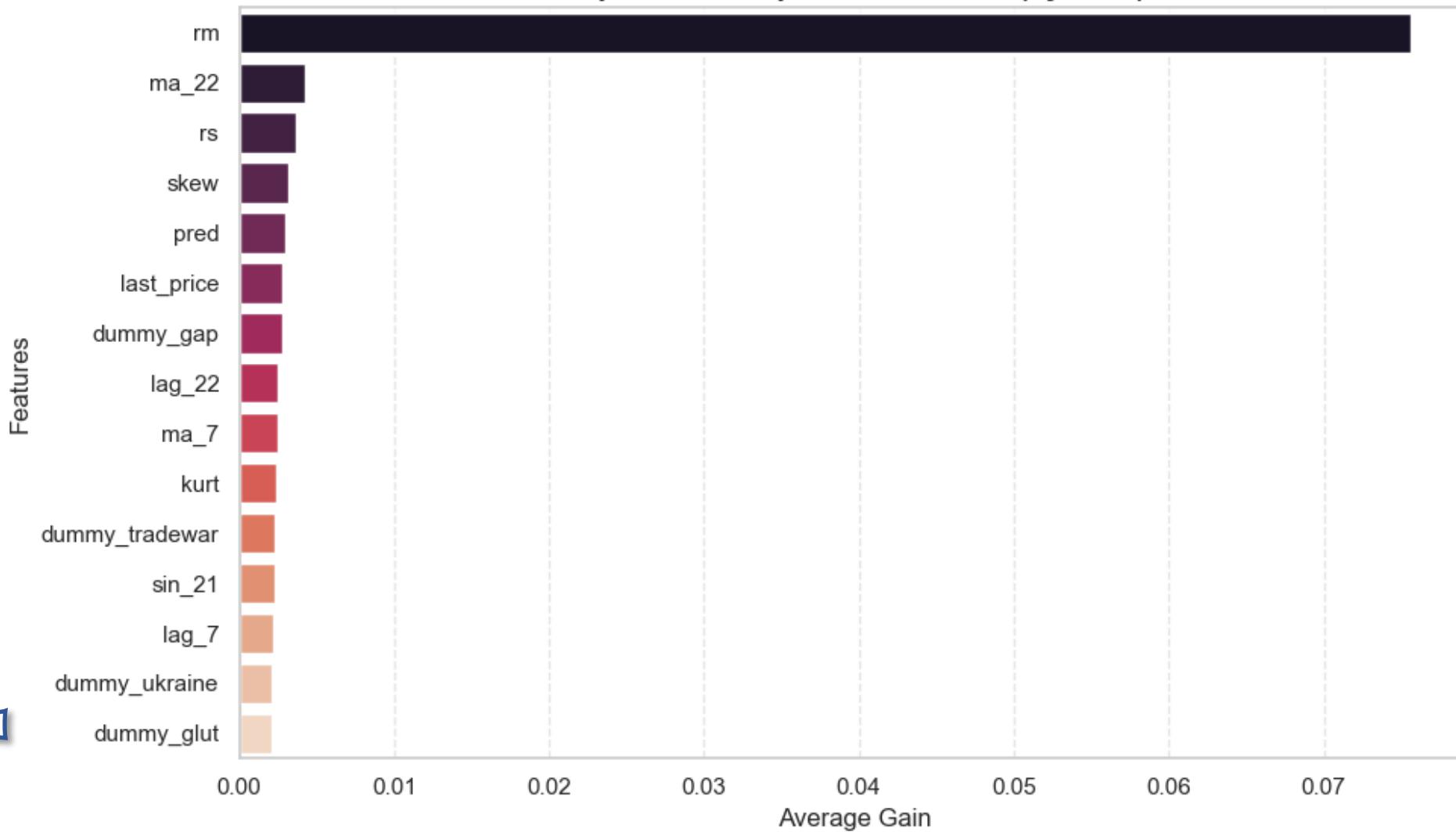


Diberikan visualisasi untuk 1 data actual dan 2 pendekatan prediksi beserta selang kepercayaannya.



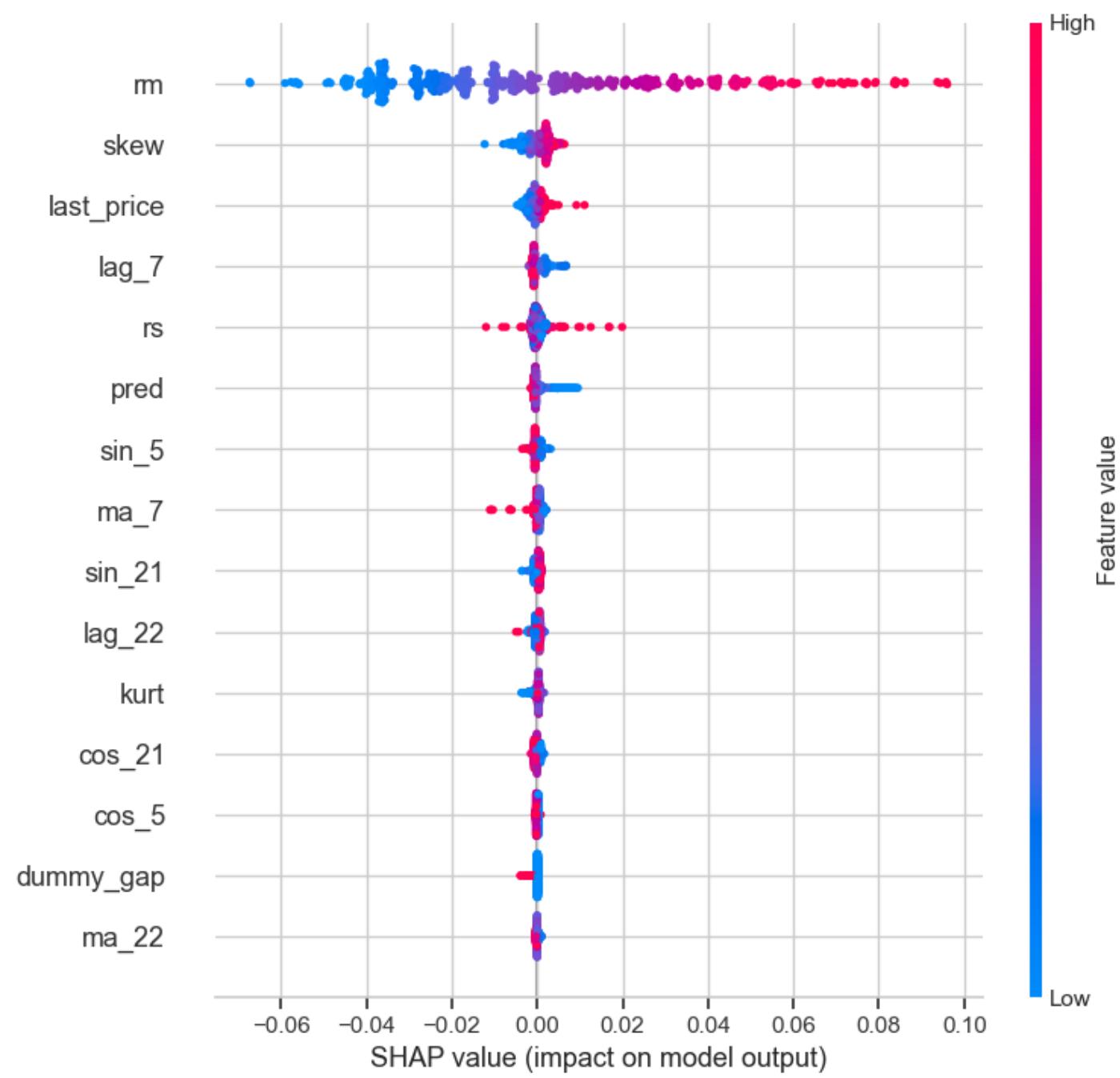
Model Hybrid dinilai mampu menangkap pola perubahan pada deret waktu dan titik nilai prediksinya terlihat tidak keluar dari selang kepercayaan.

Top 15 Most Important Features (by Gain)



1 Fitur teknikal (rm, ma_22, dll) lebih berpengaruh, pertanda terjadi volatilitas dan tren. Fitur eksternal (dummy_gap, dummy_tradewars, dll) menandakan bahwa faktor makroekonomi dan geopolitik memengaruhi galat model, saat pasar tidak stabil.

2 Fitur waktu seperti sin_21 dan lag_22 menangkap pola musiman, sedangkan skew dan kurt menunjukkan sebaran galat yang tidak normal. Pendekatan hybrid lebih mengenali pola kompleks dan non-linearitas.



High



1

Fitur *last_price* dan *rs* memiliki kontribusi SHAP tertinggi, menandakan harga terkini dan volatilitas sebagai faktor utama model. Nilai SHAP positif *last_price* menunjukkan kenaikan harga mendorong prediksi naik, sedangkan *rs* bersifat dual: dapat mencerminkan peluang atau risiko tergantung tingkat volatilitasnya.

2

Fitur *sin_21*, *cos_21*, dan *skew* berpengaruh signifikan, mengindikasikan bahwa pola musiman 1 bulanan (hari kerja) memengaruhi prediksi model. Nilai SHAP negatif pada *skew* menunjukkan bahwa pencilan dan gap harga berdampak besar secara tidak proporsional terhadap output model.

3

Melalui SHAP dikonfirmasi bahwa model sensitif terhadap harga, volatilitas, dan pola waktu, sekaligus memberi masukan terhadap penanganan untuk pencilan (*skew/kurt*).

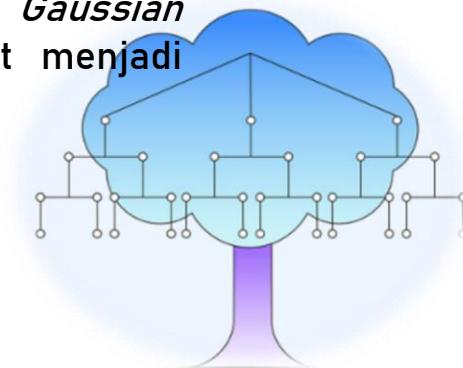


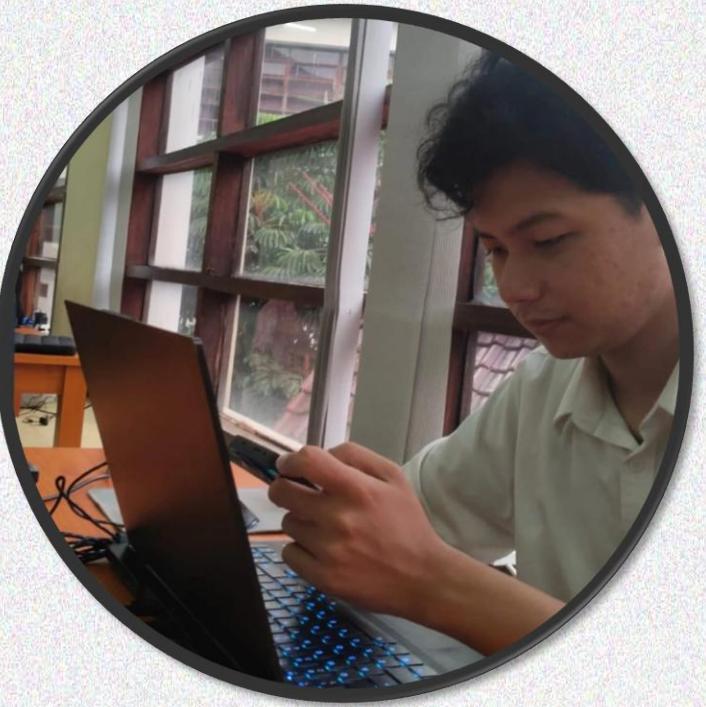


Penutup Kajian

Saran dan Kekurangan Model

- 1 Pendekatan ini mampu menangkap pola kompleks dan non-linearitas galat dengan cukup baik. Namun, pemisahan data latih dan uji masih menggunakan pembagian proporsi tetap yang dapat menyebabkan bias dalam evaluasi model time series. Ke depannya, penggunaan *rolling forecasting origin* atau *sliding window cross-validation* disarankan agar prediksi lebih stabil dan representatif terhadap dinamika waktu.
- 2 Meskipun Transformer dapat mengenali dependensi jangka panjang, kombinasi layer dan hyperparameter saat ini masih ditentukan secara eksploratif. Pendekatan statistik seperti *Bayesian optimization* atau *probabilistic neural architecture search* dapat diterapkan untuk pemilihan struktur model yang lebih efisien dan berbasis inferensi.
- 3 Secara komputasi, model hybrid ini cukup kompleks dan memerlukan sumber daya besar, terutama pada tuning dan pelatihan model sekuensial. Pada penelitian selanjutnya, pengembangan eksplorasi model statistik berbasis probabilistik seperti *Gaussian Process* atau *state-space model* (*Bayesian Structural Time Series*) dapat menjadi alternatif yang lebih ringan namun tetap kuat secara teoritis dan interpretatif.





Ghardapaty Ghaly Ghiffary
Python | Excel | R | QGIS | Power BI



Ghardapaty Ghaly Ghiffary



<https://github.com/ghiffahry>



ghiffary_17



@ghiffary_17



ghiffaryankh@gmail.com



<https://medium.com/@17611083>

