Evaluasi Model Peramalan Time Series dan Pemilihan Model Terbaik

Setelah membangun beberapa model peramalan time series (misalnya, ARIMA dan Holt-Winters), langkah selanjutnya adalah mengevaluasi kinerja masing-masing model dan memilih model yang terbaik.

# Matrik Evaluasi Peramalan Time Series

Beberapa matrik umum yang digunakan untuk mengevaluasi model peramalan time series adalah:

* **Mean Absolute error (MAE):** Rata-rata nilai absolut dari kesalahan prediksi.
* **Mean Squared Error (MSE):** Rata-rata kuadrat dari kesalahan prediksi.
* **Root Mean Squared Error (RMSE):** Akar kuadrat dari MSE.
* **Mean Absolute Percentage Error (MAPE):** Rata-rata persentase kesalahan prediksi.

## Memilih Matrik Evaluasi

Pemilihan matrik evaluasi tergantung pada tujuan dan skala data Anda.

* **MAE dan RMSE:** Lebih mudah diinterpretasikan karena satuannya sama dengan satuan data asli. **RMSE** memberikan penalti yang lebih besar pada kesalahan yang lebih besar.
* **MAPE:** Berguna untuk membandingkan kinerja model pada dataset dengan skala yang berbeda.

# Evaluasi Model dengan Scikit-Learn

Scikit-learn menyediakan fungsi mean\_absolute\_error, mean\_squared\_error, dan r2\_score

untuk menghitung matrik evaluasi.

## Pemilihan Model Terbaik

Tidak ada model peramalan time series yang terbaik untuk semua kasus. Model terbaik adalah model yang paling sesuai dengan karakteristik data Anda dan memberikan kinerja terbaik pada matrik evaluasi yang Anda pilih.

Beberapa faktor yang perlu dipertimbangkan dalam memilih model terbaik:

* **Kompleksitas Model:** Model yang lebih kompleks (seperti ARIMA dengan banyak parameter) mungkin overfitting pada data latih tetapi memiliki kinerja yang buruk pada data baru.
* **Interpretabilitas:** Beberapa model (seperti Exponential smoothing) lebih mudah diinterpretasikan daripada yang lain (seperti model jaringan saraf tiruan).
* **Waktu Pelatihan:** Beberapa model membutuhkan waktu pelatihan yang lebih lama daripada yang lain.

**Tugas Hari 16:**

Lanjutan dengan model ARIMA dan Holt-Winters yang telah anda buat untuk dataset AirPassengers.

1. **Evaluasi Model:**

* Hitung MAE, MSE, RMSE, dan MAPE untuk model ARIMA dan Holt-Winters pada data yang digunakan untuk membuat prediksi (out-of-sample data).
* Tampilkan hasil evaluasi.

1. **Pilih Model Terbaik:**

* Bandingkan kinerja kedua model berdasarkan matrik evaluasi yang telah Anda hitung.
* Jelaskan model mana yang Anda pilih sebagai model terbaik dan berikan alasannya.

**contoh Kode (Scikit-learn dan Statsmodels)**

from sklearn.metrics import mean\_absolute\_error, mean\_squared\_error

# ... (kode untuk membaca, mempersiapkan data, dan membuat model ARIMA dan Holt-Winters)

# Prediksi ARIMA (misalnya, 12 langkah ke depan)

forecast\_arima = model\_arima\_fit.get\_forecast(steps=12)

# Prediksi Holt-Winters (misalnya, 12 langkah ke depan)

forecast\_hw = model\_hw\_fit.forecast(steps=12)

# Evaluasi ARIMA

mae\_arima = mean\_absolute\_error(df['#Passengers'][-12:], forecast\_arima.predicted\_mean)

mse\_arima = mean\_squared\_error(df['#Passengers'][-12:], forecast\_arima.predicted\_mean)

rmse\_arima = np.sqrt(mse\_arima)

# Evaluasi Holt-Winters

mae\_hw = mean\_absolute\_error(df['#Passengers'][-12:], forecast\_hw)

mse\_hw = mean\_squared\_error(df['#Passengers'][-12:], forecast\_hw)

rmse\_hw = np.sqrt(mse\_hw)

# Cetak hasil evaluasi

print('MAE (ARIMA):', mae\_arima)

print('MSE (ARIMA):', mse\_arima)

print('RMSE (ARIMA):', rmse\_arima)

print('MAE (Holt-Winters):', mae\_hw)

print('MSE (Holt-Winters):', mse\_hw)

print('RMSE (Holt-Winters):', rmse\_hw)

**Catatan:**

* Kode di atas menggunakan 12 data terakhir sebagai data uji untuk evaluasi out-of-sample.
* Anda perlu mengganti bagian prediksi ARIMA dan Holt-Winters dengan kode yang Anda buat sebelumnya.
* Anda dapat menghitung MAPE secara manual menggunakan rumus: MAPE = (1/n) \* Σ(|actual - forecast| / |actual|) \* 100

**Selamat Mengerjakan Tugas! 🙂**

Tugas:

1. **Evaluasi Model:**

**from sklearn.metrics import mean\_absolute\_error, mean\_squared\_error**

**# Prediksi ARIMA (misalnya, 12 langkah ke depan)**

**forecast\_arima = model\_arima\_fit.get\_forecast(steps=12)**

**forecast\_arima\_mean = forecast\_arima.predicted\_mean**

**# Prediksi Holt-Winters (misalnya, 12 langkah ke depan)**

**forecast\_hw = model\_hw\_fit.forecast(steps=12)**

**# Sesuaikan indeks forecast\_hw agar sesuai dengan data aktual**

**forecast\_index = pd.date\_range(start=df.index[-1], periods=12, freq='MS')**

**# Pastikan data aktual untuk periode evaluasi (misalnya, 12 bulan terakhir)**

**actual\_values = df['#Passengers'][-12:].values**

**# Evaluasi ARIMA**

**mae\_arima = mean\_absolute\_error(actual\_values, forecast\_arima\_mean)**

**mse\_arima = mean\_squared\_error(actual\_values, forecast\_arima\_mean)**

**rmse\_arima = np.sqrt(mse\_arima)**

**# Evaluasi Holt-Winters**

**mae\_hw = mean\_absolute\_error(actual\_values, forecast\_hw)**

**mse\_hw = mean\_squared\_error(actual\_values, forecast\_hw)**

**rmse\_hw = np.sqrt(mse\_hw)**

**# Tampilkan hasil evaluasi**

**print('MAE (ARIMA):', mae\_arima)**

**print('MSE (ARIMA):', mse\_arima)**

**print('RMSE (ARIMA):', rmse\_arima)**

**print('MAE (Holt-Winters):', mae\_hw)**

**print('MSE (Holt-Winters):', mse\_hw)**

**print('RMSE (Holt-Winters):', rmse\_hw)**

1. **Pilih Model Terbaik:**

Model ARIMA lebih baik daripada Model Holt-Winters karena memiliki nilai yang lebih rendah.