BAB III

METODE DEKOMPOSISI *SEASONAL TREND BASED ON LOESS* (STL)

3.1 Pendahuluan

Metode dekomposisi klasik menggunakan pendekatan prosedur *moving average* sebagai pemulus data untuk mengestimasi komponen musiman dan komponen trend-siklus. Ketika deret waktu memiliki unsur trend yang cenderung naik atau turun, digunakan rata-rata dibobоti untuk mengestimasi trend-siklus. Kurva estimasi akan cenderung “rata” ketika mendekati akhir deret, yang seharusnya terus naik atau turun. Bias ini merupakan efek dari semua jenis metode rata-rata bergerak, ketika data runtun waktu mengandung trend atau memiliki siklus yang kuat (Makridakis,1997).

Untuk mengatasi hal demikian, dapat digunakan metode pemulus lain, salah satunya adalah pemulus regresi lokal. Pemulus regresi lokal menggunakan pendekatan titik ke kurva sehingga semua observasi dapat terestimasi. *Loess* merupakan salah satu metode dari pemulus regresi lokal, dimana menggunakan teknik pembobotan pada titik-titik ketetanggaan dan pencocokan derajat polinomial untuk mendapatkan nilai pemulusan pada data yang diestimasi.

.

3.2 Metode Dekomposisi STL

Metode dekomposisi STL diusulkan oleh Cleveland dkk. pada tahun 1990. STL merupakan akronim dari *Seasonal Trend Based on Loess*. Seperti metode dekomposisi lain, dekomposisi STL merupakan prosedur iteratif yang mengestimasi trend-siklus dan musiman, dimana pemulus *Loess* menjadi pendekatan utama dalam pemulusan data. Dekomposisi STL terdiri dari 2 prosedur berulang dan saling bersarang. Kedua prosedur tersebut adalah *inner loop* dan *outer loop*. Dalam setiap *inner loop*, komponen musiman dan trend-siklus diestimasi kemudian diperbaharui. Sebuah iterasi *outer loop* terdiri dari satu atau dua iterasi *inner loop* diikuti dengan identifikasi terhadap observasi ekstrim. Pengerjaan *inner loop* selanjutnya dilakukan sampai bobot pada observasi ekstrim mengecil seperti yang telah diidentifikasi pada *outer loop* sebelumnya. Terdapat beberapa parameter yang memerlukan pemilihan dalam proses dekomposisi STL. Parameter tersebut di antaranya sebagai berikut. (Cleveland, 1990).

1. $n_{(p)}$ (banyaknya observasi dalam setiap siklus pada komponen musiman),
2. $n_{(i)}$ (banyaknya jalan yang melewati *inner loop*),
3. $n_{(o)}$ (banyaknya iterasi robust *outer loop*),
4. $n_{(l)}$ (parameter pemulus untuk filter deret musiman sementara),
5. $n_{(t)}$ (parameter pemulus untuk komponen trend), dan
6. $n_{(s)}$ (parameter pemulus untuk komponen musiman).

Pembahasan model dekomoposisi STL berdasarkan model aditif, yaitu

$$Y_t = S_t + T_t + E_t\ ,\ t = 1 \text{ sampai } t = N$$

dimana

Y_t adalah nilai deret waktu (data sebenarnya) pada periode t

S_t adalah komponen musiman pada periode t

T_t adalah komponen trend-siklus pada periode t, dan

E_t adalah komponen kesalahan pada periode t

Langkah awal sebelum melakukan proses dekomposisi adalah mengidentifikasi efek kalender (*trading day*). Efek kalender dapat terjadi ketika jumlah hari dalam suatu bulan memiliki pengaruh yang berbeda secara signifikan terhadap data. Untuk pembahasan dekomposisi STL saat ini, data harus tidak memiliki efek kalender. Untuk mengetahui data memiliki efek kalender atau tidak, dapat digunakan uji F dari analisis varians (Shiskin, 1967).

a. Hipotesis:

H_0 : $\sigma_1 = \sigma_2 \ldots = \sigma_k$

H_1 : Paling sedikit ada satu tanda "=" yang tidak berlaku.

atau

H_0: Data runtun waktu tidak mengandung variasi efek kalender

H_1: Data runtun waktu mengandung variasi efek kalender

b. Statistik Uji:

$$F_{hitung} = \frac{A_y/(k-1)}{D_y/\Sigma(n_i-1)}$$

Dengan

A_y : jumlah kuadrat-kuadrat dari sumber variasi antar kelompok;

$(k - 1)$: derajat kebebasan dari A_y;

D_y : jumlah kuadrat-kuadrat dari sumber variasi dalam kelompok;

$\Sigma(n_i - 1)$: derajat kebebasan dari D_y.

c. Kriteria Uji:

Tolak H_o jika $F_{hitung} > F_{tabel} = F_{(1-\alpha)(v_1,v_2)}$ dengan $v_1 = (k-1)$ dan $v_2 = \Sigma(n_i - 1)$. Pengujian hipotesis di atas dapat juga diuji dengan menggunakan kriteria pengujian tolak H_0 jika nilai *Sig.* < α pada output program SPSS.

3.2.1 Langkah-langkah Dekomposisi STL

3.2.1.1 *Inner loop*

Setiap iterasi pada *inner loop* terdiri dari sebuah pemulusan musiman yang memperbaharui komponen musiman, diikuti sebuah pemulusan trend yang memperbaharui komponen trend. Misal $S_t^{(k)}$ dan $T_t^{(k)}$ untuk posisi waktu $t = 1$ sampai $t = N$ adalah komponen musiman dan komponen trend pada iterasi ke-k. Kedua komponen ini didefinisikan pada semua posisi waktu $t = 1$ sampai N termasuk posisi waktu dimana terdapat *missing value* pada Y_t. Komponen terbaharui pada iterasi ke $(k+1)$, $S_t^{(k+1)}$ dan $T_t^{(k+1)}$, dihitung dengan prosedur berikut.

3.2.1.1.1 Langkah 1: Pemisahan Trend

Pemisahan deret dihitung, yakni $Y_t - T_t^{(k)}$. Jika Y_t terdapat *missing value* pada posisi waktu tertentu, maka pada pemisahan deret juga akan terjadi *missing* pada posisi waktu tersebut. Pada langkah ini, $k = 0$ dan diasumsikan $T_t^{(0)} = 0$.

3.2.1.1.2 Langkah 2: Penghitungan Deret Musiman Sementara

Misal digunakan data bulanan, maka $n_{(p)} = 12$. Nilai pemisahan trend untuk setiap bulan dikumpulkan untuk membentuk subderet siklus. Setiap subderet siklus pada pemulusan trend dimuluskan dengan pemulus *Loess*, dengan banyaknya titik ketetanggaan $q = n_{(s)}$ dan derajat polinomial lokal $d = 1$. Nilai pemulusan dihitung pada semua posisi waktu pada subderet siklus, termasuk pada *missing value*, dan pada posisi sebelum posisi awal subderet dan sesudah posisi akhir subderet. Misalkan digunakan rentang subderet siklus Januari, yaitu dari Januari 1943 sampai Januari 1985 dengan *missing value* pada Januari 1960. Maka nilai pemulusan dihitung pada semua posisi dari Januari 1942 sampai Januari 1986. Kemudian nilai pemulusan untuk semua subderet siklus dikumpulkan kembali. Kumpulan ini disebut deret musiman sementara dan dinotasikan dengan $C_t^{(k+1)}$.

3.2.1.1.3 Langkah 3 : Filter Deret Musiman Sementara

Filter ini digunakan pada $C_t^{(k+1)}$. Filter terdiri dari rata-rata bergerak dengan panjang $n_{(p)}$, diikuti dengan rata-rata bergerak dengan panjang $n_{(p)}$, dan diikuti rata-rata bergerak dengan panjang 3. Jika $n_{(p)} = 12$, maka rata-rata bergerak yang digunakan menjadi MA 3x12x12. Hasilnya kemudian dimuluskan oleh pemulus *Loess* dengan derajat polinomial lokal $d = 1$ dan banyaknya titik ketetanggaan $q = n_{(l)}$. Output dinyatakan dengan $L_t^{(k+1)}$dan didefinisikan pada posisi waktu $t = 1$ sampai N.

Nilai *loss* pada awal dan akhir deret hasil dari rata-rata bergerak, diantisipasi dan diatasi oleh ekstrapolasi komponen musiman sementara pada langkah 2. Maksud dari langkah ini adalah mengidentifikasi trend-siklus yang mungkin terkontaminasi komponen musiman sementara pada langkah 2. Jika ada trend-siklus kecil dalam komponen musiman sementara, hasil dari pemulusan akan jadi deret dengan semua nilainya mendekati nol.

3.2.1.1.4 Langkah 4: Penghitungan Komponen Musiman

Komponen musiman pada jalan ke $(k+1)$ adalah $S_t^{(k+1)} = C_t^{(k+1)} - L_t^{(k+1)}$ untuk $t = 1$ sampai N. $L_t^{(k)}$ dikurangkan agar mencegah kekuatan frekuensi rendah dari masuknya komponen musiman (Cleveland, 1990).

3.2.1.1.5 Langkah 5: Pemisahan Komponen Musiman

Pemisahan komponen musiman dihitung, yakni $Y_t - S_t^{(k+1)}$. Jika Y_t terdapat *missing value* pada posisi waktu tertentu, maka pada deret pemisahan musiman juga akan terjadi *missing* pada posisi waktu tersebut.

3.2.1.1.6 Langkah 6 : Penghitungan Komponen Trend

Deret pemisahan musiman dimuluskan oleh pemulus *Loess* dengan banyaknya titik ketetanggaan $q = n_{(t)}$ dan derajat polinomial lokal $d = 1$. Nilai pemulusan dihitung pada semua posisi waktu $t = 1$ sampai $t = N$ termasuk *missing value*. Komponen trend dari iterasi ke $(k+1)$ dinyatakan dengan $T_t^{(k+1)}$ untuk $t = 1$ sampai $t = N$, merupakan himpunan nilai hasil pemulusan.

Jadi, porsi pemulusan musiman pada *inner loop* terdapat pada langkah 2, 3, dan 4. Sedangkan porsi untuk pemulusan trend pada langkah 6.

3.2.1.2 *Outer loop*

Estimasi komponen trend-siklus dan komponen musiman yang dihasilkan, digunakan untuk menghitung komponen *irregular*

$$R_t = Y_t - T_t - S_t \tag{3.1}$$

Jika nilai $|R_t|$ cukup besar, maka mengindikasikan terdapat observasi ekstrim. Keadaan ini diidentifikasi dan kemudian dilakukan perhitungan bobot. Setiap jalan dari *outer loop* terdiri dari beberapa iterasi *inner loop* kemudian diikuti dengan penghitungan bobot robust. Semakin besar nilai $|R_t|$, maka data akan mendapat bobot robust semakin kecil.

Bobot robust pada titik t adalah

$$\rho_t = B\left(\frac{|R_t|}{h}\right) \tag{3.2}$$

dimana

- $h = 6\,MAD\,(|R_t|)$
- B adalah fungsi bobot bisquare

$$B = \begin{cases} (1-u^2)^2 & untuk\ 0 \le u \le 1 \\ 0 & untuk\ u > 1 \end{cases} \tag{3.3}$$

Kemudian iterasi *inner loop* diulangi. Namun pada pemulusan *Loess* di langkah 2 dan langkah 6 bobot ketetanggaan untuk nilai pada waktu t dikalikan dengan bobot robust ρ_t. Pada langkah 1, tidak digunakan asumsi $T_v^{(0)} \equiv 0$, melainkan menggunakan kompoenen trend yang dihasilkan langkah 6 pada *inner loop* sebelumnya. Iterasi robust pada *outer loop* dilakukan sebanyak $n_{(o)}$ kali.

Jika data mengandung efek kalender, maka akan terdapat 1 langkah tambahan pada *inner loop* di atas. Pada langkah ke 7 ini, suatu komponen regresi F_t diestimasi dengan mengurangkan komponen trend-siklus dan komponen musiman dari data Y_t. Ft juga dikurangkan pada langkah 1 dan langkah 5 pada *inner loop* iterasi kedua, serta pada penghitungan komponen *irregular* pada *outer loop*. Jika iterasi robust digunakan, bobot robust yang dihasilkan akan digunakan pada pencocokan kuadrat-terkecil pada estimasi komponen regresi.

.

3.2.2 Pemilihan Parameter pada Dekomposisi STL

Seperti yang telah disebutkan sebelumnya bahwa dalam proses dekomposisi STL terdapat beberapa parameter yang harus dipilih. Pemilihan didasarkan pada informasi data awal. Parameter tersebut di antaranya sebagai berikut. (Cleveland, 1990).

3.2.2.1 Banyaknya observasi dalam setiap siklus pada komponen musiman ($\boldsymbol{n_{(p)}}$)

Suatu deret waktu dapat memiliki dua atau lebih komponen periodik. Misalnya deret waktu tahunan dapat dibuat sebagai periode bulanan $n_{(p)} = 12$, dan sebagai deret harian $n_{(p)} = 365$.

3.2.2.2 Banyaknya jalan yang melewati *inner loop* $(n_{(i)})$ dan banyaknya iterasi robust pada *outer loop* $(n_{(o)})$

Estimasi robust STL diperlukan ketika informasi awal data mengindikasikan perilaku non-Gaussian dalam deret waktu cenderung ekstrim atau dengan kata lain mengandung *outlier*. Untuk mendeteksi bahwa data memilki *outlier*, dapat digunakan metode *spatial statistics Z test*.

$$Z_{X_T} = \left|\frac{X_T - \mu_T}{\sigma_T}\right| > \theta \tag{3.4}$$

μ_T dan σ_T menyatakan rata-rata dan standar deviasi dari runtun waktu X_T dengan tingkat signifikasi 5% dan nilai $\theta = 2$,. Jika $Z_{X_T} > \theta$, maka data dideteksi sebagai *outlier*.

Jika runtun waktu tidak mengandung *outlier*, iterasi robust dapat diabaikan $(n_{(o)} = 0)$. Sehingga pada kasus seperti demikian, tidak diperlukan iterasi *outer loop*. Jika iterasi robust diabaikan, pilih $n_{(i)}$ cukup besar sehingga pembaruan komponen trend dan komonen siklus menjadi konvergen. Namun untuk beberapa kasus, kekonvergenan dapat terjadi dengan cepat. Biasanya dengan memilih $n_{(i)} = 1$ atau $n_{(i)} = 2$. Sedangkan jika iterasi robust dipakai, pilih $n_{(o)} = 5$ atau $n_{(o)} = 10$ sehingga estimasi robust pada komponen trend dan komponen musiman akan konvergen.

3.2.2.3 Parameter Pemulus untuk Filter Deret Musiman Sementara $(n_{(l)})$

Pemilihan $n_{(l)}$ adalah bilangan bulat ganjil dan lebih besar dari $n_{(p)}$. Pemilihan $n_{(l)}$ berkontribusi mencegah komponen trend dan komponen musiman tidak saling bersaing pada keadaan kesamaan variansi dalam data.

3.2.2.4 Parameter pemulus untuk komponen musiman $(\boldsymbol{n_{(s)}})$

Semakin besar $n_{(s)}$, maka subderet siklus akan menjadi lebih mulus. Selalu dipilih bilangan bulat ganjil dan paling kecil 7.

3.2.2.5 Parameter pemulus trend $(\boldsymbol{n_{(t)}})$

Semakin besar parameter yang dipilih, komponen trend akan menjadi mulus. Pemilihan menggunakan rumus berikut.

$$n_{(t)} = \left[\frac{1{,}5n_{(p)}}{\left(1 - \frac{1{,}5}{n_{(s)}}\right)}\right]_{ganjil} \tag{3.5}$$

3.3 Peramalan dengan menggunakan Metode Dekomposisi STL

Tehnik kombinasi dan prosedur dekomposisi telah diaplikasikan terhadap peramalan runtun waktu untuk meningkatkan akurasi prediksi dan untuk memfasilitasi analisis data. Marina Theodosiu dari *Central Bank of Cyprus* (2010) telah menguji manfaat dan melakukan pembatasan dekomposisi dan tehnik kombinasi pada peramalan dengan mengembangkan suatu metode peramalan baru.

Metode baru ini berdasarkan disagregasi komponen runtun waktu melalui prosedur dekomposisi STL, kemudian mengkombinasi-linearkan ekstrapolasi komponen disagregasi, dan reagregasi ekstrapolasi tersebut untuk membentuk estimasi pada runtun waktu keseluruhan. Metode ini mengapliksikan empat tehnik peramalan, yaitu ARIMA, Theta, Holt-Winters (HW) and Holt’s Damped Trend (HDT) terhadap data dari deret NN3 *competition* dan M1 *competition*. Hasil dari metode baru ini kemudian diinvestigasi menjadi kombinasi sederhana pada keempat metode peramalan tersebut dan metode peramalan dekomposisi klasik (dalam pembahasan Tugas Akhir ini digunakan model aditif). Kekuatan metode terletak pada kemampuan untuk memprediksi runtun waktu periode panjang dengan tingkat akurasi relatif tinggi, dan untuk menampilkan kekonsistenan untuk runtun waktu dengan *range* lebar, terlepas dari karakter, struktur dasar dan tingkat *noise* data (Theodosiu, 2010).

3.3.1 Ekstrapolasi Komponen Disagregasi

Komponen trend akhir, musiman akhir, dan *irregular* akhir yang didapat dari dekomposisi STL sebelumnya diasumsikan merupakan tiga runtun waktu terpisah yang kemudian akan dicari nilai peramalannya. Untuk memperoleh petunjuk dalam menentukan metode peramalan mana yang cocok bagi ekstrapolasi masing-masing komponen, empat metode peramalan yaitu ARIMA, Theta, Holt-Winters (HW) and Holt's Damped Trend (HDT) digunakan berdasarkan performa masing-masing dalam memprediksi eror dan kerelatifannya terhadap komponen dominan dan tingkat *noise* dalam data.

Metode peramalan baru ini diaplikasikan pada data NN3 *competition* untuk memprediksi 18 observasi ke depan, dengan hanya menggunakan 36 observasi pertama (3 tahun) dalam sampel data. Kemudian, hanya 54 observasi pertama dari masing-masing runtun waktu digunakan dalam analisis. Metode peramalan terbaik pada masing-masing runtun waktu kemudian diuji terhadap komponen struktural dalam runtun waktu dengan menggunakan *mean absolute scaled error* (MASE).

Pertama, untuk menentukan kekuatan masing-masing komponen dalam deret waktu, dicari koefisien determinasi dengan cara komponen diregresikan terhadap data awal. Koefisien determinasi R^2 mengindikasikan kekuatan masing-masing komponen.

$$x_t = \alpha_T + \beta_T T_t + \epsilon_{t,T}$$

$$x_t = \alpha_S + \beta_S S_t + \epsilon_{t,S}$$

$$x_t = \alpha_R + \beta_R R_t + \epsilon_{t,R} \qquad (3.6)$$

Kedua, deret diklasifikasikan ke dalam empat kelompok berdasarkan metode peramalan terbaik untuk tiap deret waktu. Adapun metode peramalan dinotasikan M, dengan $M = 1,2,3,4$ (1 menyatakan HW, 2 menyatakan HDT, 3 menyatakan Theta, dan 4 menyatakan ARIMA). Masing-masing runtun waktu kemudian diramalkan dengan keempat metode tersebut kemudian dievalusi dengan MASE sebagai metode terbaik, untuk membentuk grup G_M. Masing-masing grup diasosiasikan dengan matriks ukuran $nx3$ dengan entri-entri koefisen determinasi komponen diagregasi. n merupakan banyaknya runtun waktu dimana metode M memiliki MASE terkecil.

$$G_M = \begin{pmatrix} R^2_{S,1} & R^2_{T,1} & R^2_{R,1} \\ \cdots & \cdots & \cdots \\ R^2_{S,n_M} & R^2_{\text{换},n_M} & R^2_{R,n_M} \end{pmatrix} \qquad (3.7)$$

Tujuan pengklasifikasian ini adalah untuk menentukan hubungan antara performa setiap metode peramalan. Dari analisis tersebut dapat ditarik beberapa kesimpulan sebagai berikut.

- Untuk deret waktu dengan tingkat musiman tinggi, metode peramlan terbaik adalah ARIMA dan HW.
- Untuk deret waktu dengan tingkat trend tinggi, metode peramalan yang terbaik adalah HDT, HW, Theta, dan ARIMA.
- Untuk deret waktu dengan tingkat *irregular* tinggi, metode permalan terbaik adalah ARIMA.

3.3.2 Ekstrapolasi Komponen *Irregular*

Langkah terpenting dalam aplikasi metodologi baru ini, terletak pada estimasi komponen *irregular*. Namun sampai saat ini belum terdapat literatur yang menjelaskan tentang ekstrapolasi komponen *irregular* dari aplikasi prosedur dekomposisi secara statistik. Informasi yang masih dapat digambarkan dari komponen yang kemudian dibuang, dapat berefek negatif pada akurasi estimasi. Informasi dalam komponen *irregular* mungkin dalam bentuk autokorelasi residual dalam deret atau bentuk kondsisonal bergantung pada komponen lain dari deret awal.

Berdasarkan intuisi tersebut, komponen *irregular* dilibatkan dalam estimasi deret keseluruhan melalui tehnik kombinasi, berdasarkan ekstraksi komponen *irregular* dari deret pemisahan trend teresktrapolasi $\widehat{TR}_{t+1}$ dan deret pemisahan musiman terekstrapolasi $\widehat{SR}_{t+1}$. Keduanya secara berurutan diperoleh dari menambahkan trend dengan *irregular*, dan dari menambahkan musiman dengan *irregular*.

$$TR_t = T_t + R_t$$

$$SR_t = S_t + R_t \tag{3.8}$$

3.3.3 Peramalan

Berdasarkan penjelasan sebelumnya, peramalan komponen trend, komponen musiman, komponen kombinasi trend-*irregular*, dan komponen kombinasi musiman-*irregular*, dilakukan sesuai dengan karakter komponen tersebut. Dengan mengkombinasikan keempat komponen di atas, dapat diperoleh estimasi peramalan untuk runtun waktu keseluruhan sebagai berikut.

$$\hat{x}_{t+1} = \left(\hat{T}_{t+1} + \hat{S}_{t+1} + \widehat{T}R_{t+1} + \widehat{S}R_{t+1}\right)/2 \tag{3.9}$$

Himpunan pengukuran kesalahan digunakan untuk mengevaluasi performa metode peramalan tersebut. Penghitungan keempat ukuran tersebut disajikan dalam tabel 3.1 berikut.

Tabel 3.1
Ukuran Eror Untuk Evaluasi Performa Metode Peramalan

MAE	*Mean Absolute Error*	mean $(\lvert\epsilon_t\rvert)$
MASE	*Mean Absolute Scaled Error*	mean $(\lvert q_t\rvert)$
sMAPE	*Symetric Mean Absolute Perc.Error*	mean $\left(200\frac{\lvert Y_t - F_t\rvert}{\lvert Y_t + F_t\rvert}\right)$

Sumber : Theodosiu, 2010

$$\epsilon_t = Y_t - F_t \tag{3.10}$$

$$q_t = \frac{\epsilon_t}{\frac{1}{n-1}\sum_{i=2}^{n}\lvert Y_i - Y_{i-1}\rvert} \tag{3.11}$$

Y_t menyatakan observasi sebenarnya dan F_t menyatakan peramalan pada periode ke t.