Contents

1. Time series decomposition: TSD, STL

Typical TSD (time series decomposition) looks like:

$$y_t = T_t + S_t + R_t$$

T_t — trend, S_t — seasonality component, R_t — random fluctuations, a.k.a. noise.

The decomposition can also take on the following forms:

$$y_t = T_t S_t R_t, \text{ or } y_t = (T_t + S_t) R_t.$$

1. 1. Classical TSD (using moving averages)

Moving average (MA) is given by the following expression:

$$\mathrm{MA}(y_t; m) = \frac{1}{m} \sum_{j=-k}^{k} y_t,$$

where $m = 2k + 1$ is called *window size* and has to be odd. Backward formula:

$$\mathrm{MA}(y_t; m) = \frac{1}{m} \sum_{j=-m}^{0} y_t,$$

Forward formula:

$$\mathrm{MA}(y_t; m) = \frac{1}{m} \sum_{j=0}^{m} y_t.$$

For $m = 4$:

$$\mathrm{MA}(y_t; 4) = \frac{1}{4}(y_{t-1}, y_t, y_{t+1}, y_{t+2}).$$

Moving average over moving average:

$$\begin{aligned}
\mathrm{MA}(\mathrm{MA}(y_t, 4); 2) &= \frac{1}{2}[\mathrm{MA}(y_{t-1}; 4), \mathrm{MA}(y_t; 4)] = \\
&= \frac{1}{2}\left[\frac{1}{4}(y_{t-2}, y_{t-1}, y_t, y_{t+1}) + \frac{1}{4}(y_{t-1}, y_t, y_{t+1}, y_{t+2})\right] = \\
&= \frac{1}{8} y_{t-2} + \frac{1}{4} y_{t-1} + \frac{1}{4} y_t + \frac{1}{4} y_{t+1} + \frac{1}{8} y_{t+2}.
\end{aligned}$$

MAs are used to: 1) smooth out the data; 2) extranct the trend.

Weighted moving average (WMA):

$$\mathrm{WMA}(y_t; m) = \sum_{j=-k}^{k} y_{t+j} \cdot w_j,\;\; w_j \geq 0,\;\; \sum w_j = 1.$$

The classical TSD algorithm is given as follows:

1. Compute trend component using $2 \times m$-MA if m is even and m-MA if it is odd.

$$\hat{T}_t = \begin{cases} \mathrm{MA}(y_t; m), \text{ if } m \text{ is odd,} \\ \mathrm{MA}(\mathrm{MA}(y_t; m); 2), \text{ if } m \text{ is even.} \end{cases}$$

2. Detrend the time series (TS):

$$y_t - \hat{T}_t = S_t + R_t.$$

3. Compute $\hat{S}_t$ by averaging detrended TS for a season (assuming that S_t does not change from season to season).
4. $\hat{R}_t = y_t - \hat{S}_t - \hat{T}_t$.

Note: TSD assumes that S_t is constant throughout the seasons and that the trend line itself is not sensitive to sharp fluctuations.

1. 2. STL decomposition

An alternative to classical TSD would be *STL decomposition* (Seasonal Trend decomposition via LOESS). Here LOESS (locally estimated scatterplot smoothing) is type of local regression for modeling and smoothing data $(x_i, y_i)_{i=1}^{m}$. Its key components are:

1. Kernel function. For example, Gaussian kernel

$$w_i = \exp\left(-\frac{(x_i - x)^2}{2\tau^2}\right).$$

2. Smoothing parameter τ. Smaller τ leads to narrower windows and more flexible models, larger τ — to wider windows and less flexible models and $\tau \to +\infty$ means that $w_i = 1$, hence model becomes a simple linear regression.

Given data $(x_i, y_i)_{t=1}^{m}$ or $(t, y_t)_{t=1}^{T}$, the LOESS algorithm step-by-step:

1. Choose a kernel function $\mathcal{F}$ and set smoothing parameter τ.

2. For all x_i:

   2.1. Calculate $w_i = \mathcal{F}(x_i, x, \tau)$

   2.2. Build weighted regression model. For example, weighted least squares:

$$L = \sum_{i=1}^{n} w_i (y_i - \Theta^T x_i)^2,$$

   where $\Theta = (X^T W X)^{-1} X^T W y$.

   2.3. Make predictions $\hat{y}(x)$ for x only.

   2.4. "Forget" the model.

1. 2. 1. STL algorithm

Input: $Y = \{y_1, ..., y_\tau\}$.

Parameters: n_p — # of outer iterations (1-2)

n_i — # of inner iterations (1-2)

n_l — trend smoothing parameter (smoothing parameter for LOESS)

n_s — seasonality smoothing parameter

n_o — residual smoothing parameter (optional, for residues R_t).

0. Outer loop: repeat the following steps n_p times.

1. Initialization:

   1.1. set trend $T^{(0)} = 0$ or other initial approximation (MA for example);

   1.2. set weights $w = \{1, 1, ...1\}$ (optional, for residues).

2. Inner loop: repeat n_i times

   2.1. Detrend time series: $D = Y - T$.

   2.2. Compute seasonal component:

   2.2.1. Split D subseries by seasons;

   2.2.2. For each subseries apply the LOESS smoothing with $\tau = n_s$ and weights w.

   2.2.3. Assemble the smoothed subseries into a seasonal component C.

2.2.4. Center the seasonal component C by subtracting moving average.

2.3. Update seasonal component $S = C$.

2.4. Deaseasonalize the data: $Y_{\text{desd}} = Y - S$

2.5. Update the trend: apply LOESS for Y_{desd} with $\tau = n_l$ and "robust" weights w (obtain T).

3. Compute the residuals $R = Y - T - S$.
4. Update weights: recompute weights w based on residues R to reduce the influence of outliers usually using Tuikey's biweight function.

Post-processing:

1. Normalize seasonality: mean value of S for each season should be zero.
2. Smoothen the trend if needed.

Result: trend T, seasonality S, residual noise R

Pros:

- *flexiblity:* it is robust to outliers;
- *robustness:* it can model non-linear trends;
- *arbitrary period:* it can work with any seasonality.

1. 2. 2. Tuikey's biweight function

Tuikey's biweight function is used to update the weights w using the following algorithm:

1. Obtain the residuals $R = Y - S - T$
2. Compute MAD (median absolute deviation):

$$\mathrm{MAD} = \mathrm{median}(|r_i - \mathrm{median}(R)|).$$

Normalize: $S \approx 1.4826 \cdot \mathrm{MAD}$, since $\sigma = 1.4826$

3. Compute the normalized residuals:

$$u_i = \frac{r_i}{C \cdot S},$$

where C is a tuning constant ($C = 4.685$).

4. Bisquare function

$$w_i = \begin{cases} (1 - u_i)^2, \ |u_i| < 1, \\ 0, \ |u_i| \geq 1. \end{cases}$$

5. If $S = 0$, then $w_i = 0$ (all residuals are the same). If $\text{MAD} = 0$, but the residuals are not the same, we use standard deviation instead of MAD.

For example, if $R = [0.1, -0.2, 3.0, -0.1, 10.0]$:

1. $\text{median}(R) = 0.1$, hence $\text{MAD} = \text{median}(|R - 0.1|) = 0.3$
2. $S = 0.3 \cdot 1.4826 \approx 0.4448$
3. $C = 4.685 \Rightarrow C \cdot S = 2.083$
4. $r_3 = 3.0 : |u_3| = |\frac{3.0}{2.083}| \approx 1.44 > 1 \Rightarrow u_3 = 0$
5. $r_3 = 10.0 : |u_5| = 4.801 > 1 \Rightarrow u_5 = 0$
6. $r_1 = 0.1 : |u_1| \approx 0.04821 \Rightarrow w_1 \cdot (1 - 0.048^2)^2 \approx 0.995$

2. Weak, strong stationarity. Stationarity tests: DF, ADF, KPSS. Reduction to stationary time series.

2. 1. Stationarity and Ergoticity

Stationarity is a key feature of time series. There are several kinds of stationarity:

- *Strict stationarity:* joint distribution of any segment of time series $\left(y_{t_1}, y_{t_2}, ..., y_{t_k}\right)$ is equivalent to $\left(y_{t_1+\tau}, y_{t_2+\tau}, ..., y_{t_k+\tau}\right)$ $\forall \tau$.
- *Weak stationarity:*

   1. $\forall t \ \mathbb{E}[y_t] = \mu$,

   2. $\forall t \ \mathbb{D}[y_t] = \sigma^2 < +\infty$,

   3. $\forall t, s, \tau \ \ \mathrm{cov}(y_t, y_s) = \mathrm{cov}(y_{t+\tau}, y_{s+\tau}) = \gamma(|t - s|)$. Here $\gamma(\cdot)$ is a function that depends on distance between points.

2. 1. 1. Non-stationary time series examples

1. Time servies with determinitstic trend:

$$y_t = \alpha + \beta t + \varepsilon_t, \ \varepsilon_t \sim \mathcal{N}(0, \sigma^2).$$

Here, $\mathbb{E}[y_t] = \alpha + \beta t$ which is not a constant value.

2. $y_t = \sin t + \varepsilon_t, \ \varepsilon_t \sim \mathcal{N}(0, \sigma^2)$. Here

$$\mathbb{E}[y_t] = \begin{cases} 1, \ t = \frac{\pi}{2} + 2\pi k \\ -1, \ t = -\frac{\pi}{2} + 2\pi k \end{cases}$$

and since it depends on t the TS is non-stationary.

3. Random Walk: $y_t = y_{t-1} + \varepsilon_t, \ \varepsilon_t \sim \mathcal{N}(0, \sigma^2), \ \ \mathrm{cov}(\varepsilon_t, \varepsilon_s) = 0, \ t \neq s$. Let us write out values of this TS:

$$\begin{aligned} y_1 &= y_0 + \varepsilon_1, \\ y_2 &= y_1 + \varepsilon_2 = y_0 + \varepsilon_1 + \varepsilon_2, \\ &\dots \\ y_t &= y_0 + \sum_{i=1}^{t} \varepsilon_i \end{aligned}$$

Therefore, $\mathbb{E}[y_t] = y_0, \ \mathbb{D}[y_t] = t\sigma^2$.

2. 1. 2. Stationary time series examples

1. $y_t = \varepsilon_t, \varepsilon_t \underset{\text{iid}}{\sim} \mathcal{N}(0, \sigma^2)$ – white noise. In this case,

$$\forall t, s : t \neq s,\ \mathbb{E}[y_t] = 0,\ \mathbb{D}[y_t] = \varepsilon^2 < \infty \to \text{stationary}$$

2. $y_t = \beta_1 y_{t-1} + \varepsilon_t,\ \beta \in (-1, 1),\ \varepsilon_t \underset{\text{iid}}{\sim} \mathcal{N}(0, \sigma^2)$

$$\begin{aligned} y_t &= \beta_1 y_{t-1} + \varepsilon_t = \beta_1(\beta_1 y_{t-2} + \varepsilon_{t-1}) + \varepsilon_t = \\ &= \beta_1^t y_0 + \sum_{i=1}^{t} \beta_1^{t-i} \varepsilon_i. \end{aligned}$$

Here, since ε_i are independant from each other:

$$\begin{aligned} \mathbb{E}[y_t] &= \mathbb{E}\left[\beta_1^t y_0 + \sum_{i=1}^{t} \beta_1^{t-i} \varepsilon_i\right] = \beta_1^t y_0 + \sum_{i=1}^{t} \beta_1^{t-i} \mathbb{E}[\varepsilon_i] = \\ &= \beta_1^t y_0 \quad \text{if} \quad t \to \infty,\ \beta_1^t \to 0. \\ \mathbb{D}[y_t] &= \mathbb{D}\left[\beta_1^t y_0 + \sum_{i=1}^{t} \beta_1^{t-i} \varepsilon_i\right] = \sum_{i=1}^{t} \beta_1^{2(t-i)} \mathbb{D}[\varepsilon_i] = \\ &= \left(\beta_1^{2t-2} + \beta_1^{2t-4} + \ldots + 1\right) \cdot \sigma^2 \\ \operatorname{cov}(y_t,\ y_{t+1}) &= \operatorname{cov}\Big(\beta_1 y_{t-1} + \varepsilon_t,\ \beta_1 y_t + \varepsilon_{t_1}\Big) \\ &= \operatorname{cov}\left(\beta_1^t y_0 + \sum_{i=1}^{t} \beta_1^{t-i} \varepsilon_i,\ \beta_1^{t+1} y_0 + \sum_{i=1}^{t+1} \beta_1^{t+1-i} \varepsilon_i\right) = \\ &= \beta_1 \operatorname{cov}(\varepsilon_t,\ \varepsilon_t) + \beta_1^3 \operatorname{cov}(\varepsilon_{t-1},\ \varepsilon_{t-1}) + \ldots + \beta_1^{2t-1} \operatorname{cov}(\varepsilon_1,\ \varepsilon_1) = \\ &= \sum_{i=1}^{t} \beta_1^{2i-1} \mathbb{D}[\varepsilon_{t+1-i}] \to \frac{\beta_1}{1 - \beta_1^2} \cdot \sigma^2 = \text{const.} \end{aligned}$$

A random stochastic process is called *ergodic* if its statistical properties can be estimate using a sample from it.

Note: any ergodic process is stationary and almost any stationary process is ergodic.

2. 2. Stationarity tests

2. 2. 1. Unit root

Time series with unit root do not have a constant average level and have stochastic trends.

Let us consider a simple model: $y_t = \varphi \cdot y_{t-1} + \varepsilon_t,\ \varepsilon_i \overset{\text{iid}}{\sim} \mathcal{N}(0, \sigma^2),\ \varphi$ is constant.

1. $|\varphi| < 1$ means that the process is stationary;
2. $|\varphi| > 1$ is a non-stationary or explosive time series;
3. $|\varphi| = 1$ is the unit root case, not stationary, since:

$$y_t = y_{t-1} + \varepsilon_t = y_0 + \sum_{i=1}^{n} \varepsilon_i \Rightarrow \mathbb{D}[y_t] = t\sigma^2.$$

Why unit root?

Let us define a lag operator $Ly_t = y_{t-1}$. Then, $y_t = \varphi y_{t-1} + \varepsilon_t$ can be rewritten as $y_t = \varphi L y_t + \varepsilon_t$ hence $y_t(1 - \varphi L) = \varepsilon_t$.

Taking this into account, the characteristic equation would be

$$(1 - \varphi z) = 0 \Rightarrow z = \frac{1}{\varphi}$$

and if $\varphi = 1$ then $z = 1$ and $y_t = y_{t-1} + \varepsilon_t$.

2. 2. 2. Dickey-Fuller test (unit root test)

1. Consider a time series $y_t = \varphi y_{t-1} + \varepsilon_t$. Let $\Delta y_t = y_t - y_{t-1}$, then:

$$\Delta y_t = (\varphi - 1) y_{t-1} + \varepsilon_t = \gamma y_{t-1} + \varepsilon_t.$$

2. Formulate the hypotheses:

$$H_0 : \gamma = 0\ (\varphi = 1) \Rightarrow \text{unit root} \Rightarrow \text{non-stationary time series.}$$
$$H_1 : \gamma < 0\ (\varphi < 1) \Rightarrow \text{no unit root} \Rightarrow \text{stationary time series.}$$

3. Evaluate γ by fitting regression: $\Delta y_t = \gamma y_{t-1} + \varepsilon_t$. Estimate standard t-statistic for γ:

$$t_{\text{stat}} = \frac{\hat{\gamma}}{\text{SE}(\hat{\gamma})}$$

4. Dickey-Fuller distribution: if H_0 is correct, t_{stat} does not follow the standard t-distribution, it follows Dickey-Fuller distribution.

Significance level	**Critical value**
1%	-3.43
5%	-2.86
10%	-2.57

5. If $t_{\text{stat}} <$ crit. val. $\rightarrow H_0$ is rejected,

If $t_{\text{stat}} >$ crit. val. $\rightarrow H_0$ is not rejected.

2. 2. 3. Modification of DF test

Basic regression is very simple model. Instead, it is often epxanded:

$$\Delta y_t = \alpha + \beta t + \gamma y_{t-1} + \varepsilon_t.$$

This model is able to perform stationarity checks around deterministic trends.

2. 2. 4. Augmented Dickey-Fuller test

DF test assumes that ε_t are not correlated. This issue can be solved by adding lagged differences to the regression. Those lagged differences will reduce auto-correlation in error terms ε_t.

$$\Delta y_t = \alpha + \beta t + \gamma y_{t-1} + \sum_{i=1}^{p} \delta_i \Delta y_{t-i}$$

How does the choice of p impact the model:

- if p is too small, then the correlation issue will not be solved,
- if p is too big, then the power of test decreases.

How to choose p:

1. $p \approx \sqrt[3]{T},\ p \approx \sqrt{T}$.
2. Test different p, choose p which gives you the “best” regression: BIC, AIC, MQIC.

Interpretation of ADF is exactly the same.

2. 2. 5. KPSS (Kwiatkowski-Phillips-Schmidt-Shin) test

1. KPSS assumes that the time series can be decomposed into the following sum:

$$y_t = \xi_t + r_t + \varepsilon_t,$$

where:

- ξ_t is deterministic trend,
- r_t is stochastic trend such that $\mathbb{D}[r_t] = \sigma_r^2$,
- ε_t – white noise.

2. H_0: time series is stationary $\Rightarrow \sigma_r^2 = 0 \Rightarrow y_t = \xi_t + \varepsilon_t$,

H_1: time series is not stationary $\Rightarrow \sigma_r^2 > 0 \Rightarrow r_t \neq 0$.

3. Fit regression:

   3.1. $y_t = \alpha + \beta t + \varepsilon_t \Rightarrow$ residuals $e_t = y_t - \hat{\alpha} - \hat{\beta} t$.

   3.2. Accumulation of residuals $S_t = \sum_{i=1}^{t} e_i$.

   3.3. Calculate KPSS value:

$$\text{KPSS} = \sum_{i=1}^{T} \frac{S_t^2}{T^2 \sigma_\varepsilon^2},$$

   where σ_ε^2 is the variance of ε_t estimated using Newey-West method.

4. Decision logic: if $\text{KPSS} < \text{crit. value}$, reject H_0. Otherwise, H_0 is not rejected.

3. Filtration problem. Deterministic methods of filtration: MA, SMA, EMA, polynomial smoothing.

4. Filtration and smoothing using Fourier analysis: Fourier series, Fourier transform, DFT, FFT, SSFT.

5. Time series forecasting problem. Multi-step ahead forecasting: two main approaches.

6. Exponential smoothing, Holt's linear model, ETS models.

7. Autocorrelation and partial autocorrelation. AR, MA, ARMA, ARIMA models.

8. Predictive clustering

9. Predictive clustering for trajectory forecasting

10. Clusterization for time series: DBSCAN, Wishart, metrics

11. Time series forecasting with neural networks