**Main methods of reduction to stationary time series**

1) Trend

2) Nonconsistent dispersion

1. Taking difference of time series:

$$y_i \to \Delta y_i,\ \Delta y_i = y_i - y_{i-1},\ i = 2, ..., \tau.$$

2. Substract the trend component:

2.1. TSD $\to$ Trend $\to y_i -$ Trend;

2.2. Polynomial regression.

2.* Lagged difference:

$$y_i \to \Delta_k y_i,\ \Delta_k y_i = y_i - y_{i-k}$$

and adjust the seasonality.

2.** Subtract the seasonal component:

$$\text{TSD} \to \text{Seasonal component} \to y_i - \text{Season}$$

**Dispersion stabilization.**

1. Box-cox transformation:

$$y = \{y_1, ..., y_\tau\},\ y_i > 0,$$
$$\tilde{y}_i = \begin{cases} \frac{y_i^\lambda - 1}{\lambda}, & \text{if } \lambda \neq 0, \\ \log y_i, & \text{if } \lambda = 0. \end{cases}$$

**Note:**

$$\lambda = \begin{cases} 1 \to \text{no transformation}; \\ 0.5 \to \text{square root, i.e. softer than log}; \\ 0 \to \log. \end{cases}$$

**Using maximum likelyhood function.**

Given a likelyhood function for $N$:

$$L = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(z_i - \mu)^2}{2\sigma^2}\right),$$

$z_i = \tilde{y}_i = \text{Box-Cox}(y_i; \lambda)$, hence

$$L = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(\tilde{y}_i - \mu)^2}{2\sigma^2}\right) \prod_{i=1}^{n} y_i^{\lambda-1}$$

$$\log L = l = -\frac{n}{2}\log\pi - \frac{n}{2}\log\sigma^2 - \sum_{i=1}^{n}\log\left(-\frac{(\tilde{y}_i - \mu)^2}{2\sigma^2}\right) +$$

$$+(\lambda - 1)\sum_{i=1}^{n}\log y_i.$$

If $\delta y_i \leq 0 : \ (y_i + \alpha) > 0, \ i = 1, \ldots\tau.$

When to apply Box-Cox:

1. Graphical test: plot variance against mean. Use Box-Cox if there is a clear dependance.
2. Distribution is asymmetric (skewed idk).

# Autocorrelation and partial autocorrelation.

Autocorrelation function (ACF, denoted as $\text{ACF}(k)$) shows correlation of $y_i$ with log-ed component of time series $y_{t-k}$ for different $k$'s. It is given by the following expression:

$$\text{ACF}(k) = \text{corr}(y_t, y_{t-k}) = \frac{\text{Cov}(y_t, y_{t-k})}{\sigma(y_t)\sigma(y_{t-k})} \approx \frac{\sum_{\tau=k}^{T}(y_k - \overline{y})(y_{t-k} - \overline{y})}{\sum_{t=1}^{T}(y_t - \overline{y})}.$$

1. Trend;
2. Menoly(???) of the process;
3. Seasonality.

In order to get rid of (...)'s influence partial autocorrelation function $\text{PACF}(k)$ is used. It shows correlation between $y_t$ and $y_{t-k}$ but removes the effect of all other internal(?) lags $(y_{t-1}, y_{t-2}, \ldots, y_{t-k+1})$.

$$\text{PACF}(k) = \text{Corr}(y_t, y_{t-k} | \; y_t, y_{t-1}, \ldots, y_{t-k+1}).$$

PACF is calculated as follows:

1. Fit a regression:

$$y_t = \varphi_{k_1} y_{t-1} + \varphi_{k_2} y_{t-2} + \ldots + \varphi_{k_k} y_{t-k} + \varepsilon_t$$
$$\varphi_{k_k} = \text{PACF}(k)$$

Linear models we may look up: $\text{AR}(k)$, $\text{MA}(k)$, $\text{ARMA}(p, k)$, $\text{ARIMA}(k)$.

# Data filtering and smooting

Data filtering is **not** smooting. Rather smoothing is a tool used in data filtering. Filtering is a timse series transformation aimed at highlinghting, analyzing or supressing certain characterstics (components) of time series.

Main goals of filtering:

1. Trend extraction;
2. Noise supression;
3. Artifact removal;
4. Time series decomposition.

A problem that may arise during filtering is finding a compromise between precision and smoothing.

**I. Determinate methods of filtration.**

1. $\text{SMA}(y_t; 2m + 1) = \frac{y_{t-m} + y_{t-m+1} + \ldots + y_{t+m}}{2m+1}$ – central(?) mean. the problem with this one is that the last m observations are not really

there. In order to takle this problem “left” and “right” variations of this formula are used:

$$\mathrm{SMA}_{\text{left?}}(y_t, m) = \frac{y_{t-m} + y_{t-m+1} + \ldots + y_t}{m+1}.$$

2. $\mathrm{WMA} = \frac{\sum w_i y_i}{\sum w_i}$.
3. $\mathrm{EMA}(y_t) = \alpha y_t + (1 - \alpha)\ \mathrm{EMA}(y_{t-1})$.

Polynomial (Savitzky-Golay) filter

Given data points, choose a window of size $n = 2m + 1$ and fit a polynomial line of a low degree then choose its value at $i$ as TS value at $i$. Algorithm step-by-step (at point $i$):

1. Choose the window of size $n = 2m + 1$.
2. Fit a polinomial $P(i) = \alpha_0 + \alpha_1 i + \alpha_2 i^2 + \ldots + \alpha_k i^k$.
3. Least squares minimization:

$$\sum_{i=-m}^{m} (P(i) - y_i)^2 \to \min_{X_j}$$

4. $P(0) = \hat{\alpha}_0 \to$ smoothed value for current $y_t$.

Downside: polynomials fitted for each point, which is suboptimal.

$\hat{\alpha}_0$ can be expressed as weighted combination of all $y_i$ inside the window:

$$\hat{\alpha}_0 = c_{-m} y_{-m} + c_{-m+1} y_{-m+1} + \ldots + c_m y_m,$$

where $c_j$ are coefficients of Savitzky-Golay filter, which depend on window size and degree of polynomial.

How to compute $c_j$:

1. $P(i) = \alpha_0 + \alpha_1 i + \ldots + \alpha_k i^k$

2. $$P(-m) = \alpha_0 + \alpha_1 \cdot (-m) + ... + \alpha_k \cdot (-m)^k \approx y_{-m},$$
$$...$$
$$P(0) = \alpha_0,$$
$$...$$
$$P(m) = \alpha_0 + \alpha_1 m + ... \alpha_k m^k.$$

3. $$X\alpha \approx y,\ X = \begin{pmatrix} 1 & -m & (-m)^2 & ... & (-m)^k \\ 1 & -m+1 & (-m+1)^2 & ... & (-m+1)^k \\ ... & ... & ... & ... & ... \\ 1 & m & m^2 & ... & m^k \end{pmatrix},$$
$$\|X\alpha - y\|^2 \to \min_{\alpha}$$

$$\hat{\alpha}_0 = c_0^T \hat{\alpha} = c_0^T \left(X^T X\right)^{-1} X^T y,\ c_0 = [1, 0, ..., 0]^T$$
$$\hat{\alpha}_0 = C^T y = c_{-m} y_{-m} + ... + c_m y_m$$

How to deal with harder points:

1. Asymmetric window
2. Use polynomials calculated for the first and last full window.

# Fourier transform

Fourier series is a decomposition of a function $f \in C[a, b]$ with a orthogonal function system $g_{k(x)}$ in some euclidean space:

$$f(x) = \sum_{k=1}^{\infty} c_k g_{k(x)},\ (f, g_k) = \int_a^b f(x) g_k(x)\ \mathrm{dx} = 0$$

If $g_k(x)$ is a trigonometric system:

$$g_k \in \left\{ \frac{1}{2l}, \frac{1}{\sqrt{l}} \cos\left(\frac{\pi x}{l}\right), \frac{1}{\sqrt{l}} \sin\left(\frac{\pi x}{l}\right), ... \right\}$$

Then f(x):

$$f(x) = \frac{a_0}{2} + \sum_{k=1}^{\infty}\left[a_k \cos\left(\frac{k\pi x}{l}\right) + b_k \sin\left(\frac{k\pi x}{l}\right)\right],$$

$$a_k = \frac{1}{l}\int_{-l}^{l} f(x)\cos\left(\frac{k\pi x}{l}\right)\ \mathrm{dx},$$

$$b_k = \frac{1}{l}\int_{-l}^{l} f(x)\sin\left(\frac{k\pi x}{l}\right)\ \mathrm{dx},\ b_0 = 0,\ b_{-k} = -b_k.$$

In a more general case:

$$f(x) = \sum_{k=-\infty}^{\infty} c_k e^{iw_k x},\ w_k = \frac{\pi k}{l},\ c_k = \frac{1}{2l}\int_{-l}^{l} f(x)e^{-iw_k x}\ \mathrm{dx}$$

Since $\sin(kx) = \frac{e^{ikx}-e^{-ikx}}{2i}$, $\cos(kx) = \frac{e^{ikx}+e^{-ikx}}{2}$,

$$f(x) = e^{iw_0 x}\cdot\frac{a_0}{2} + \sum_{k=1}^{\infty}\left[a_k\frac{e^{iw_k x}+e^{-iw_k x}}{2} + b_k\frac{e^{iw_k x}-e^{-iw_k x}}{2i}\right] =$$

$$= \frac{a_0}{2}e^{iw_0 x} + \frac{1}{2}\sum_{k=1}^{\infty}[a_k e^{iw_k x} + a_k e^{-iw_k x} - ib_k e^{iw_k x} + ib_k e^{-iw_k x}] =$$

$$= \frac{a_0}{2}e^{iw_0 x} + \frac{1}{2}\sum_{k=1}^{\infty}(a_k - ib_k)e^{iw_k x} + \frac{1}{2}\sum_{k=1}^{\infty}(a_k + ib_k)e^{-iw_k x} =$$

$$= \sum_{k=-\infty}^{\infty} c_k e^{iw_k x}.$$