Model-Based Decomposition

ESTP training

0. Outline

- Unobserved component models
  - STS, AMB
- SEATS algorithm
- Estimation
  - MMSE estimators, algorithms
- Analysis
  - Finite / infinite approaches
- SEATS in JD+
  - Parameters, non decomposable models, estimation

0.1 Some notations

- Operators
  - Backward operator: $B^a y_t = y_{t-a}$
  - Forward operator: $F^a y_t = y_{t+a}$
  - Differencing: $\Delta y_t = y_t - y_{t-1} = (1 - B)y_t$
  - Seasonal differencing: $\Delta_f y_t = y_t - y_{t-f} = (1 - B^f)y_t$
  - Annual sum: $S_f y_t = y_t + \cdots + y_{t-f+1}$
- ARIMA
  - $P(B)x_t = x_t + p_1 x_{t-1} + \cdots + p_n x_{t-n}$
  - $\nabla(B)\Phi(B)y_t = \Theta(B)\varepsilon_t$

0.2 Useful tools

- Wold representation of an ARMA model

$$\Phi(B)y_t = \Theta(B)\varepsilon_t \Leftrightarrow y_t = \Psi(B)\varepsilon_t, \Psi(B) = \frac{\Theta(B)}{\Phi(B)}$$

- Auto-covariance generating function (acgf)

$$acgf(y_t) = \sigma^2\Psi(z)\Psi(z^{-1}) = \sigma^2\frac{\Theta(z)\Theta(z^{-1})}{\Phi(z)\Phi(z^{-1})}$$

- Spectrum ≡ Fourier transform of the acgf

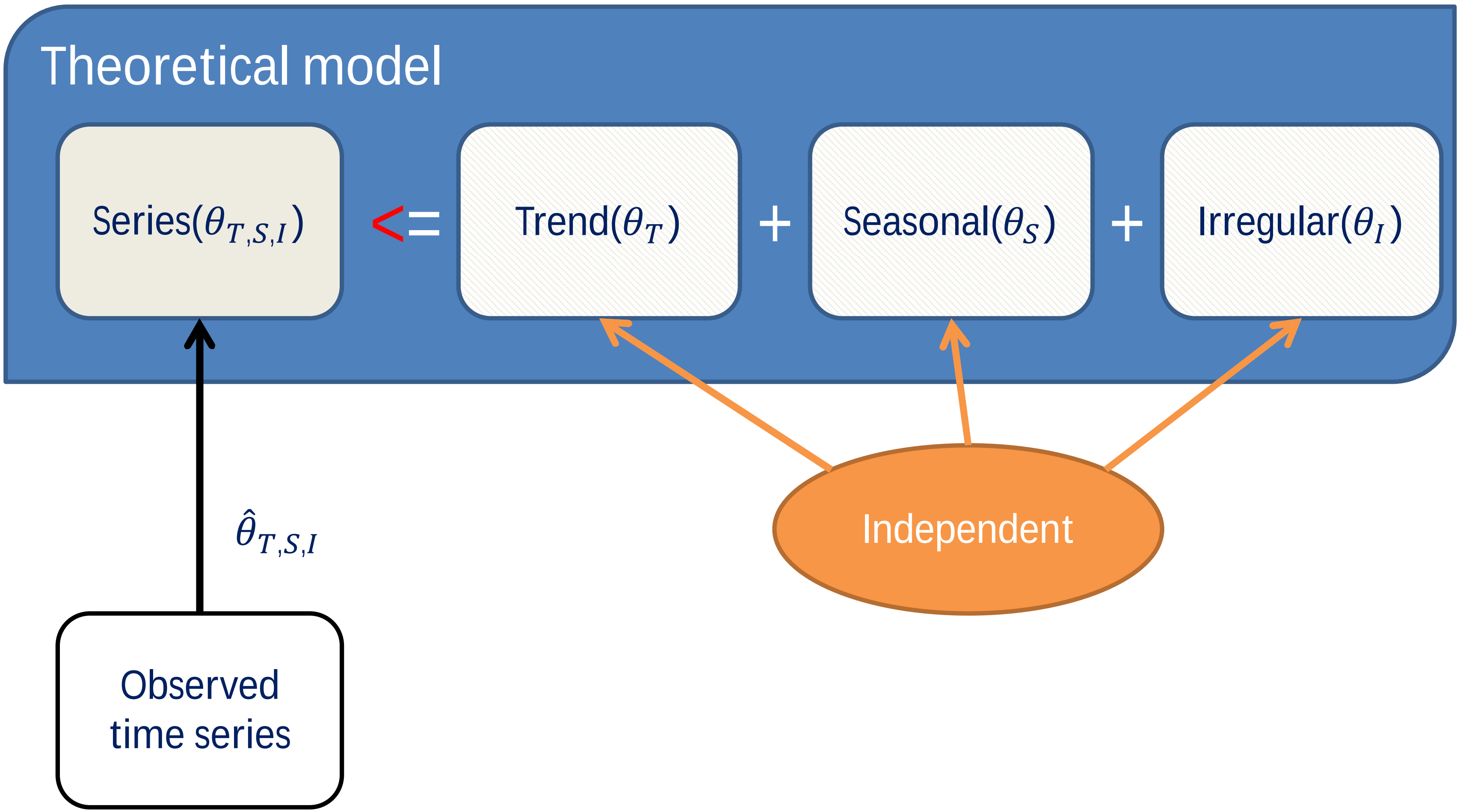
$$g_y(\lambda) = \frac{\sigma^2}{2\pi}\frac{|\Theta(e^{-i\lambda})|^2}{|\Phi(e^{-i\lambda})|^2}$$

- Extension to the non-stationary cases

1. Unobserved components model

- Principle
  - The different components of the series are “defined” by [ARIMA] models
  - The components are independent
  - The sum of the model of the components is the model of the series

1.1 Unobserved components model (STS)



1.1.1 Basic structural model (STS)

- Local linear trend (l_t)

$$\Delta l_t = m_t + \mu_t, \quad \mu_t \sim N(0, \boldsymbol{\sigma_\mu^2})$$
$$\Delta m_t = \nu_t, \quad \nu_t \sim N(0, \boldsymbol{\sigma_\nu^2})$$

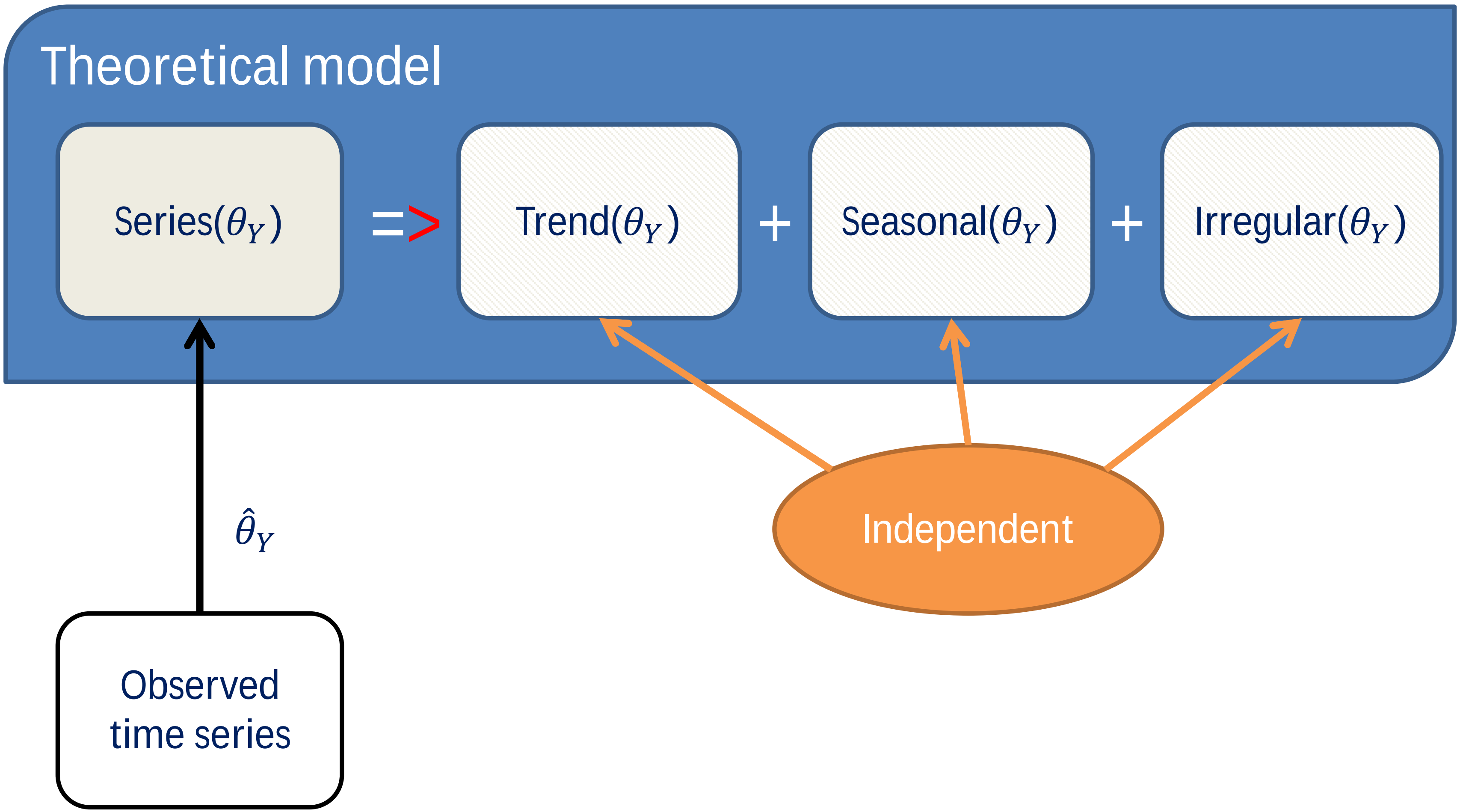
- Seasonal component (i_t)*(dummy model)*

$$S_f s_t = \gamma_t, \quad \gamma_t \sim N(0, \boldsymbol{\sigma_\gamma^2})$$

- Noise (i_t)

$$i_t = \varepsilon_t, \quad \varepsilon_t \sim N(0, \boldsymbol{\sigma_\varepsilon^2})$$

1.2 Unobserved components model (AMB)



1.2.1 Airline model

$$\Delta\Delta_f y_t = \Theta(B)\,\Theta\left(B^f\right)\varepsilon_t, \quad \varepsilon_t \sim N(0, \sigma_\varepsilon^2)$$

Or

$$y_t = y_{t-f} + (y_{t-1} - y_{t-f-1}) + \varepsilon_t + \boldsymbol{\theta_f}\varepsilon_{t-f} + \boldsymbol{\theta}(\varepsilon_{t-1} + \boldsymbol{\theta_f}\varepsilon_{t-f-1}),$$

$$\varepsilon_t \sim N\left(0, \boldsymbol{\sigma_\varepsilon^2}\right)$$

1.2.2 Airline decomposition

Model

D: 1.00000 - B - B^12 + B^13

MA: 1.00000 - 0.638958 B - 0.401037 B^12 + 0.256246 B^13

sa

D: 1.00000 - 2.00000 B + B^2

MA: 1.00000 - 1.57954 B + 0.604993 B^2

Innovation variance: **0.50128**

trend

D: 1.00000 - 2.00000 B + B^2

MA: 1.00000 + 0.0721892 B - 0.927811 B^2

Innovation variance: **0.01558**

seasonal

D: 1.00000 + B + B^2 + B^3 + B^4 + B^5 + B^6 + B^7 + B^8 + B^9 + B^10 + B^11

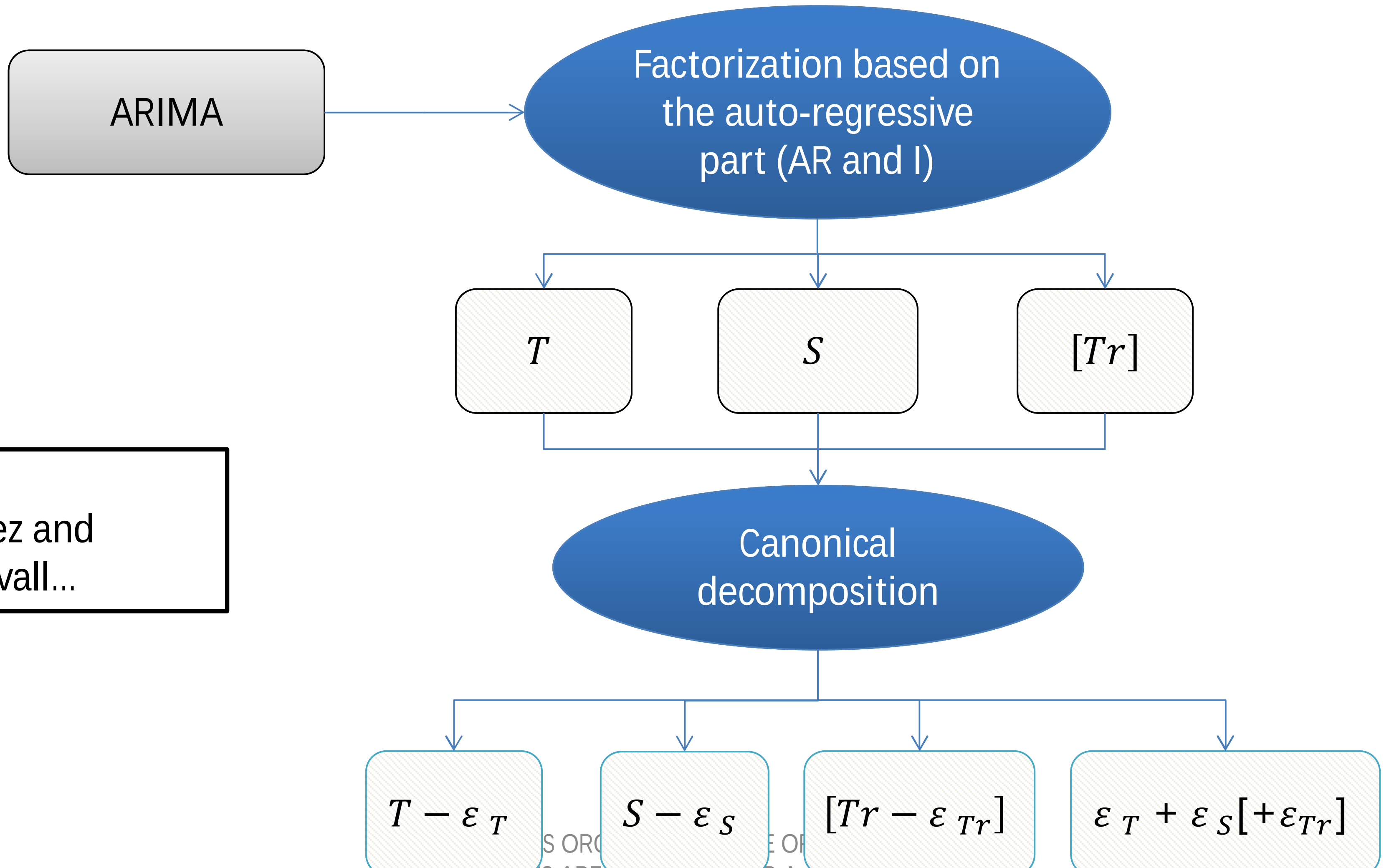
MA: 1.00000 + 0.808535 B + 0.559003 B^2 + 0.297153 B^3 + 0.0540774 B^4 - 0.150773 B^5 - 0.307157 B^6 - 0.412051 B^7 - 0.468058 B^8 - 0.482247 B^9 - 0.465546 B^10 - 0.432937 B^11

Innovation variance: **0.10862**

irregular

Innovation variance: **0.31773**

2. SEATS algorithm



See:
Gomez and Maravall...

2.1 SEATS Algorithm. Details (I)

- ARIMA:

$$\Phi(B)\Delta(B)y_t = \Theta(B)\varepsilon_t$$

- Factorization of the AR polynomial:

$$\Phi(B)\Delta(B) = \prod(1 - \alpha_i B)$$

TrendCycleSelector.java

```java
public boolean accept(final Complex root) {
    Complex iroot = root.inv();
    if (root.getIm() == 0) {
        return iroot.getRe() >= m_bound;
    } else {
        if (iroot.abs() >= m_bound) {
            double arg = Math.abs(iroot.arg());
            if (arg <= m_lfreq) {
                return true;
            }
        }
        return false;
    }
}
```

  - Trend-cycle
    - α_i real,
      - $\alpha_i \geq k$
    - α_i complex,
      - $|\alpha_i| \geq k$, $\arg(\alpha_i) \leq c$

$\left(c = \frac{\pi}{s}\right) \sim cycle\ length \geq two\ years$

– Seasonal

- α_i real,
  - $\alpha_i < -l$
- α_i complex,
  - $|\arg(\alpha_i) - f_s| \leq e$

f_s *seasonal frequency*

– Transitory (I)

- All other roots

SeasonalSelector.java

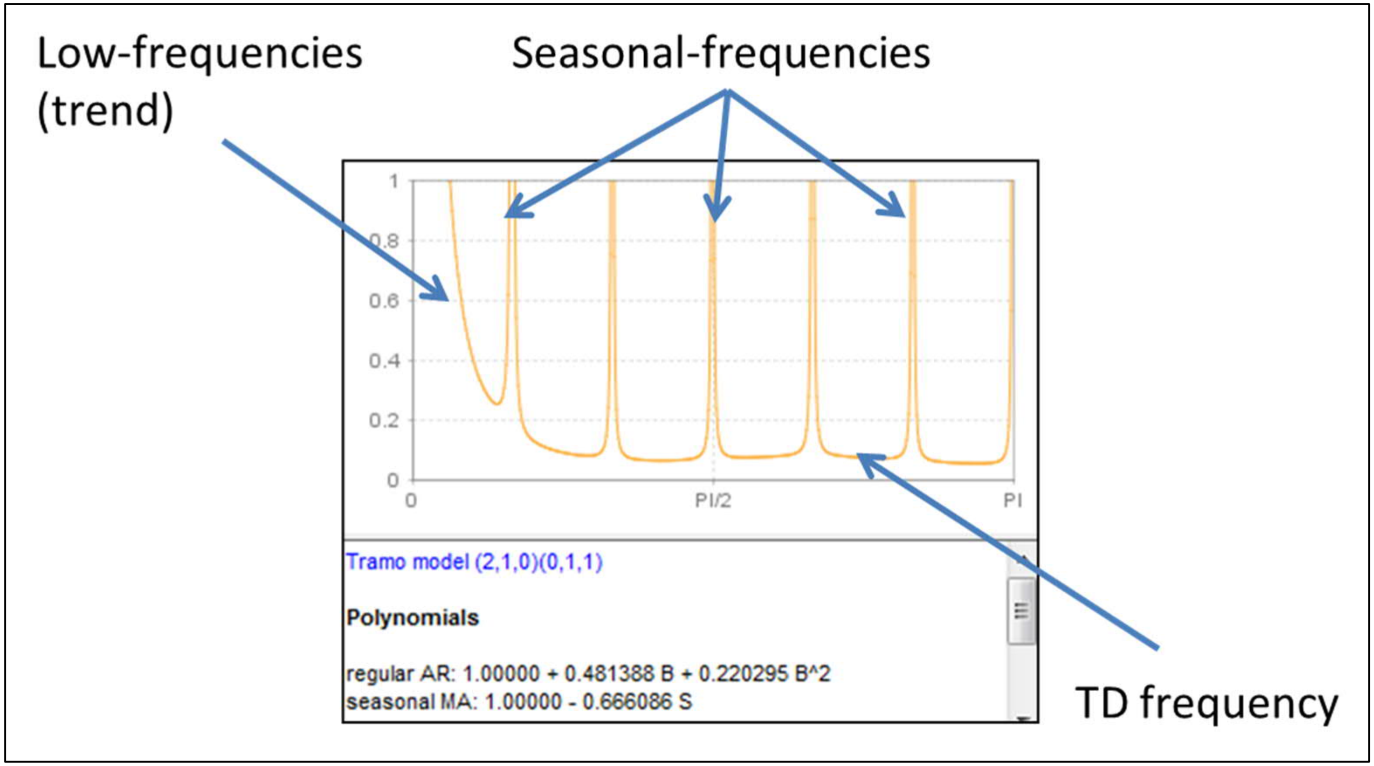
```java
public boolean accept(final Complex root) {
    if (Math.abs(root.getIm()) < 1e-6) {
        if (1/root.getRe() < -m_k)
            return true;
        else
            return false;
    }

    double pi = 2 * Math.PI / m_freq;
    double arg = Math.abs(root.arg());
    double eps=m_epsphi/180*Math.PI;
    for (int i = 1; i <= m_freq / 2; ++i) {
        if (Math.abs(pi * i - arg) <= eps)
            return true;
    }
    return false;
```

Trend boundary	*k*
Seasonal tolerance (degree)	*e*
Seasonal boundary	*l*
Seas.boundary (unique)	*l (no seasonal part)*

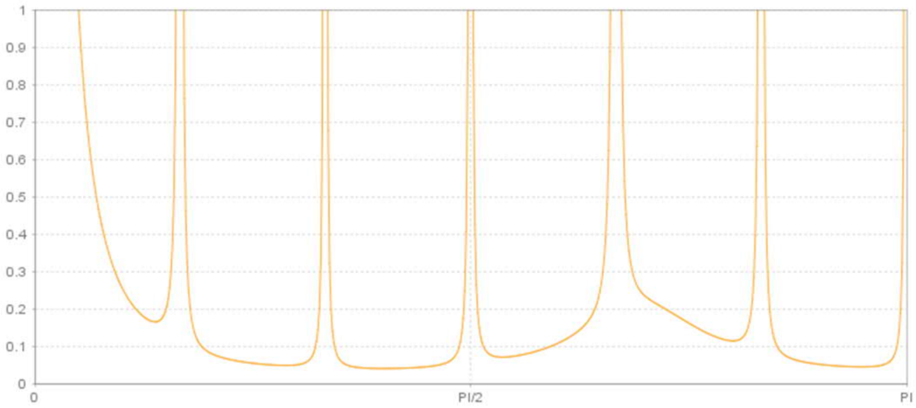
2.2 SEATS spectral decomposition

- Splitting of the spectrum following the different « types » of frequencies



CONTRACTORS ORGANISING SOME OF THE COURSES ARE ACTING UNDER A FRAMEWORK CONTRACT CONCLUDED WITH THE COMMISSION

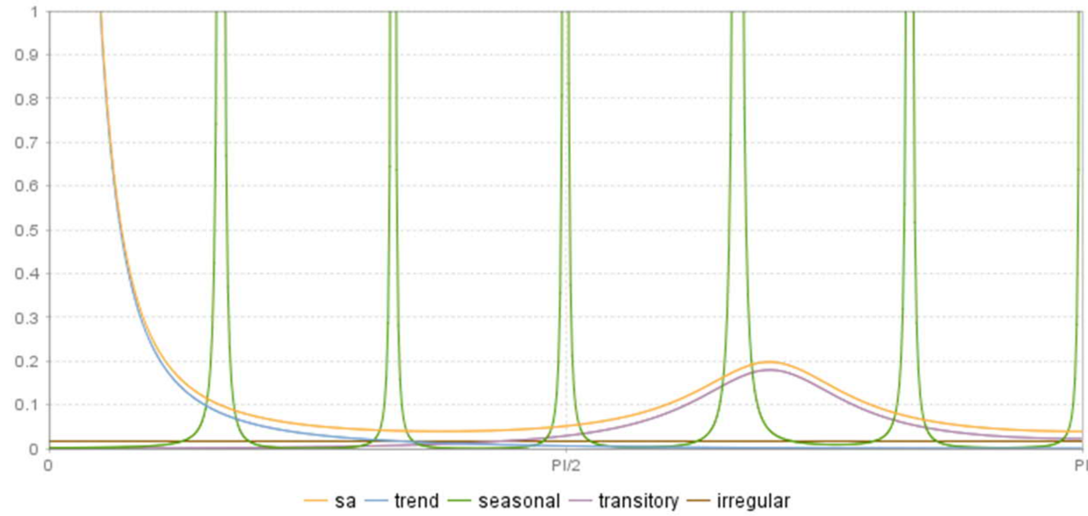
2.2 SEATS spectral decomposition



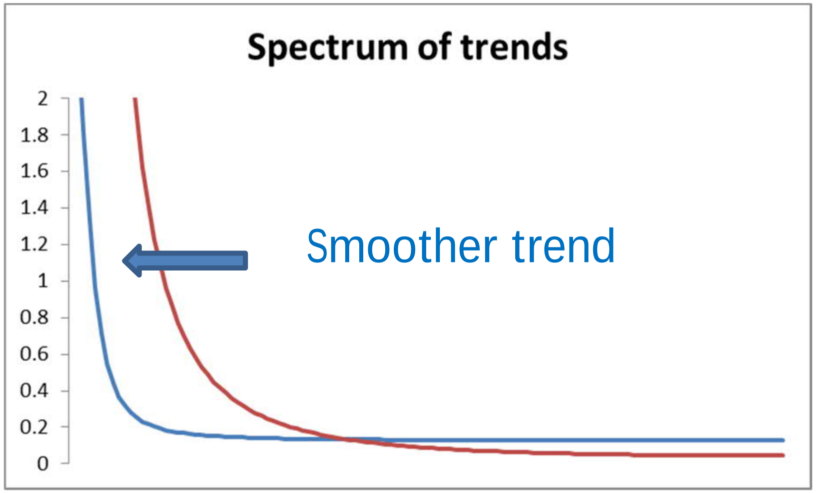
$$S_Y = S_T + S_S + S_{Tr} + S_I$$

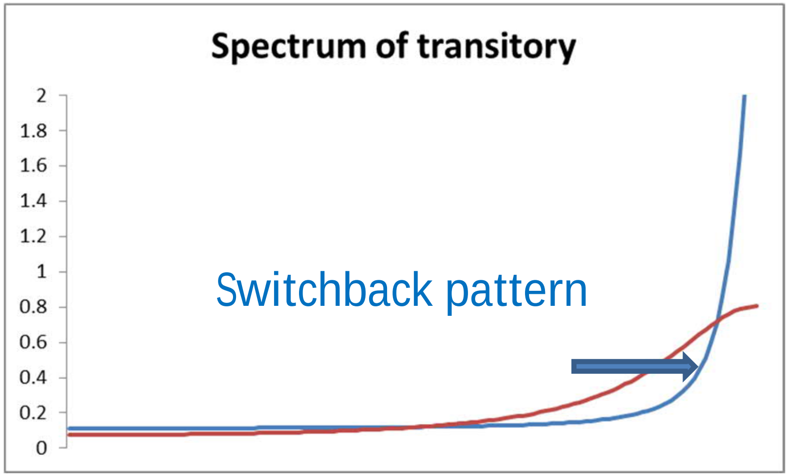
$$S_{SA} = S_T + S_{Tr} + S_I = S_Y - S_S$$

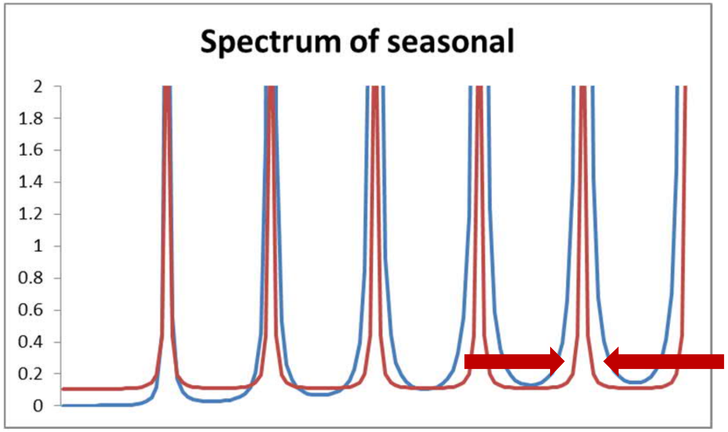
Independent components !



2.2 SEATS spectral decomposition







More stable seasonal

3. Estimation of the model

Components model

- Independant components
- Sum of cmps <=> y

- Estimators = Linear transformations of y
- MMSE criterion

Components estimators

- $\hat{S} = L_S Y, \quad \widehat{N} = L_N Y$
- Correlated estimators

- Estimation by means of KF, Burman's, McElroy's algorithms...

Components estimates

- $(\hat{s}, \hat{n}) = KF(y)$
- ...

3.1 Basic case

Simple derivation for finite stationary models

$$Y_t = S_t + N_t$$

$$\boxed{\begin{aligned} \hat{S}_t &= L_S Y_t \\ E((S_t - \hat{S}_t)Y_t') &= 0 \end{aligned}}$$

$$\Leftrightarrow \Sigma_{SS} - L_S \Sigma_{YY} = 0$$

$$\Leftrightarrow L_S = \Sigma_{SS} {\Sigma_{YY}}^{-1}$$

$$\hat{S}_t = \Sigma_{SS} {\Sigma_{YY}}^{-1} y_t, \quad \hat{N}_t = \Sigma_{NN} {\Sigma_{YY}}^{-1} y_t$$

$$cov(\hat{S}_t, \hat{N}_t) = \Sigma_{SS} \Sigma_{YY}^{-1} \Sigma_{NN} \neq 0$$

3.2 Wiener-Kolmogorov filters

MMSE estimator for infinite series (see Whittle[1963], also valid for non stationary series)

Using MA representations:

$$\phi(B)X_t = \theta(B)\epsilon_t \Leftrightarrow X_t = \psi(B)\epsilon(t), \quad \psi(B) = \frac{\theta(B)}{\phi(B)}$$

$$\hat{S}(t) = k_S \frac{\psi_S(B)\psi_S(F)}{\psi(B)\psi(F)} y(t) = \nu_S(B,F)y(t)$$

$$\nu_S(B,F) = k_S \frac{\theta_S(B)\phi_n(B)\theta_S(F)\phi_n(F)}{\theta(B)\theta(F)}$$

3.3 Estimation of the components

- STS and AMB → UCARIMA models
  - Same estimation algorithms
  - Common analysis tools
- 3 solutions, strictly equivalent (except for SD)
  - Burman algorithm (WK filters): legacy and default solution, fastest
  - Kalman smoother: more stable, exact SD
  - [Matrix computation]: Not available in JD+ 3.x

4. Analysis

Components estimators

McElroy:

$L_S = M_S \; (n \times n \; matrix)$

$M_S[i,.] = weights \; for \; S[i]$

Maravall (whittle...)

$L_S = \psi_S(B)\psi_S(F)$ [WK filters]

$\hat{S}(t) = \xi_S^-(B)\varepsilon(t) + \xi_S^+(F)\varepsilon(t)$ [PsiE-weights]

Error analysis based on the PsiE-weights

- Analysis of the final and preliminary estimators
- Revision analysis

4.1 Analysis in the frequency domain

Gain of the filter: $\nu_S(\omega) = \frac{g_S(\omega)}{g_y(\omega)}$

$$\begin{aligned} g_{\hat{S}}(\omega) &= \left[\frac{g_S(\omega)}{g_y(\omega)}\right]^2 g_y(\omega) \\ &= \nu_S(\omega)^2 g_y(\omega) \\ &= \frac{g_S(\omega)}{g_y(\omega)} g_S(\omega) \end{aligned}$$

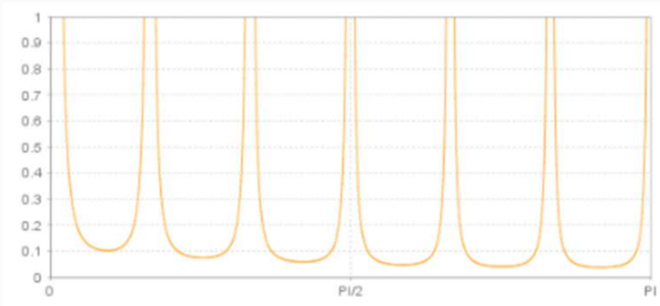
Remarks:

- $g_{\hat{S}}(\omega) \leq g_S(\omega)$
- "Dips" in the spectrum of the estimator (roots of ϕ_n)

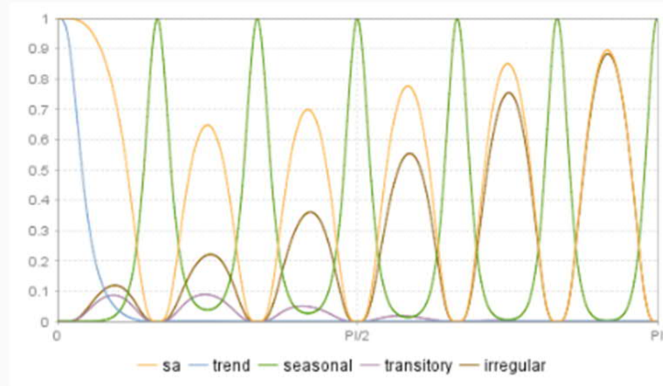
4.1 Analysis in the frequency domain

- (Squared-) gain of the filters:
  - What are the frequencies that are kept/removed by the filter
- The estimators “underestimate” the variance of the components
- Similar analysis for the preliminary estimators
  - Non-symmetric filters → phase effects

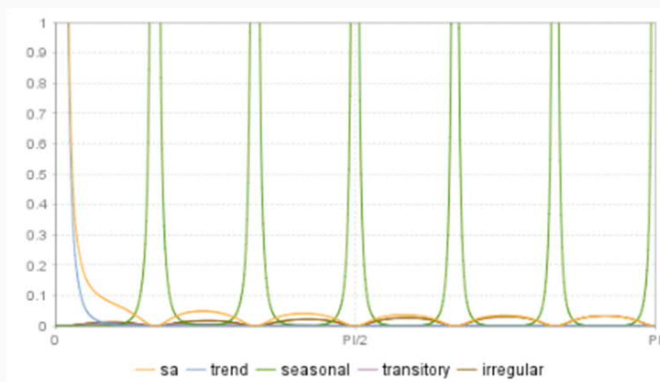
4.2 Graphical representations



(a) $g(\omega)$



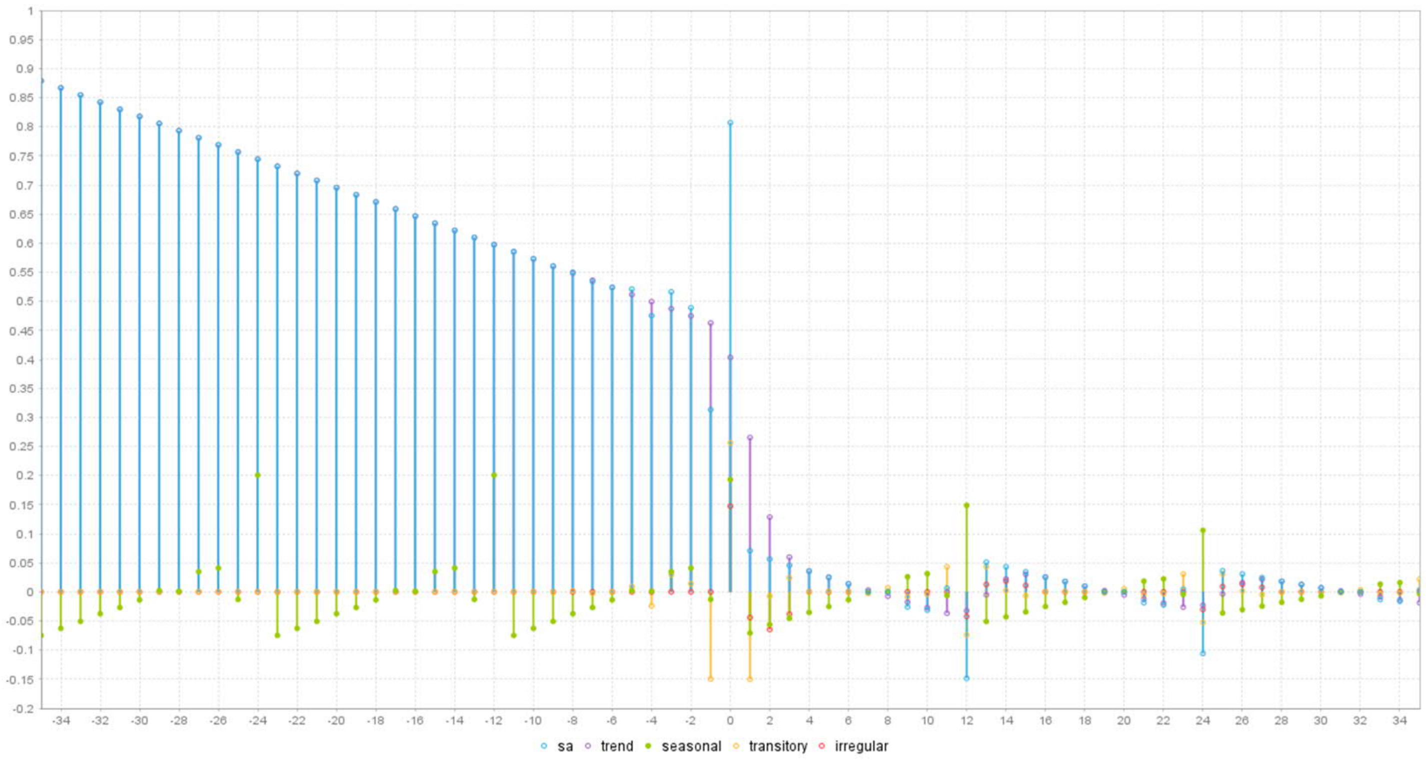
(b) $\nu_X{}^2(\omega)$



(c) $g_{\hat{X}}(\omega)$

4.2 PsiE-weights

- =Psi weights for estimators



4.3 Analysis

- What matters?
  - Understanding the differences between the “theoretical components” and their “estimators”
    - For instance: “dips” in the spectrum of the estimator
  - Understanding the properties of the estimators
    - Model of Irregular ≠ white noise, negative ac(1) in many SA estimators, …
  - Understanding PsiE-weights

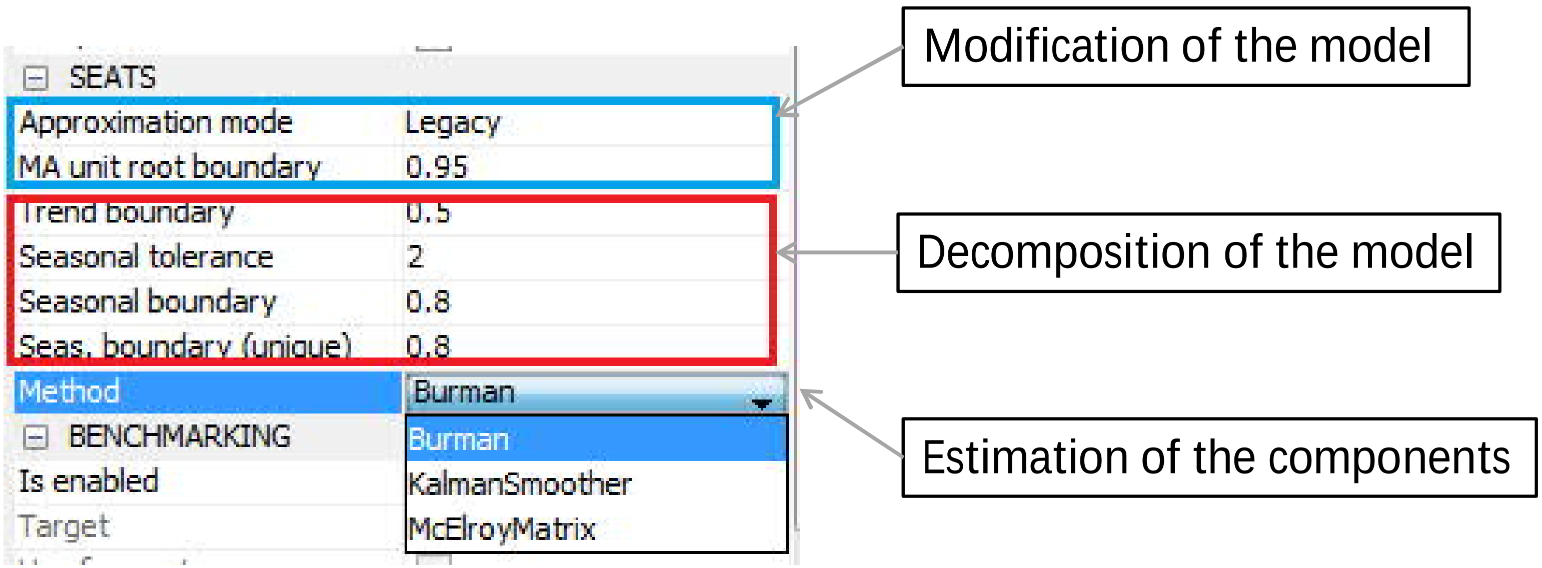
$$y_t = \sum_{i\leq t} \psi_i \varepsilon_i \Rightarrow \hat{s}_t = \nu(B,F) y_t = \nu(B,F) \sum_{i\leq t} \psi_i \varepsilon_i$$

$$\Rightarrow \hat{s}_t = \sum_{i\leq t} \xi_{s,i}^{-} \varepsilon_i + \sum_{i>t} \xi_{s,i}^{+} \varepsilon_i$$

$$\Rightarrow \hat{s}_{t|T} = \sum_{i\leq t} \xi_{s,i}^{-} \varepsilon_i + \sum_{t<i\leq T} \xi_{s,i}^{+} \varepsilon_i$$

$$\Rightarrow \hat{r}_{t|T} = \sum_{t>T} \xi_{s,i}^{+} \varepsilon_i$$

5. SEATS in JD+



CONTRACTORS ORGANISING SOME OF THE COURSES ARE ACTING UNDER A FRAMEWORK CONTRACT CONCLUDED WITH THE COMMISSION

5.1 SEATS in JD+ (cont.)

- Impact of the parameters
  - Only in case of AR polynomials

  - k
    - Small: possible « noisy » trend
    - k ≈ 1: more stable trend
  - e
    - Large (>5): possible short term cycle in the seasonal (for instance, stochastic TD)→erratic seasonal
  - l
    - Small (<.8): higher risk of erratic seasonal

- General consideration: threshold effects are unavoidable

5.2 Non decomposable models

- Some models are not decomposable (often due to complex models)
  - Best solution: change yourself the model (→airline)
  - Otherwise:
    - Legacy: Seats search for another SARIMA model, as similar as possible to the original model
    - Noisy: Seats add noise in the initial model (→ I = Tr[= 0])

6. Final remarks

- What matters?
  - Impact of the model on
    - SA/S “smoothness” ⇒ Be very careful with stationary AR roots
    - Revisions
  - Use PsiE-weights to understand/anticipate revisions
  - Model-based diagnostics
    - Variance estimators<> variance estimates
      - Should not happen if the original model is well defined
      - Be careful with non decomposable models / fixed models / “bad” models
  - To go further, see:
    - “SEASONAL ADJUSTMENT AND SIGNAL EXTRACTION IN ECONOMIC TIME SERIES”, by R. Gomez and A. Maravall