European Commission

Strumenti per rilevare anomalie e relazioni informative in dati del commercio internazionale

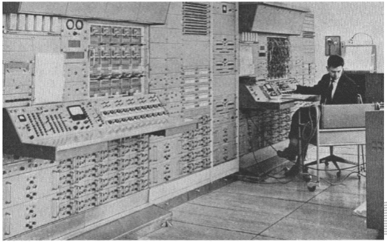
Domenico Perrotta and Francesca Torti

MATLAB in ambito aziendale, universita e policy research, November 8th 2024

Joint Research Centre

The Joint Research Centre of the European Commission

The JRC provides independent, evidence-based knowledge and science, supporting EU policies to positively impact society.



Digital Transformation and Data Directorate

- Digital economy
- Cybersecurity
- Algorithmic Transparency
- Data Governance
- *Text and data mining*
- Advanced computing & ICT

- *Anti-fraud and international trade analysis*
- *Text Mining for Democracy: disinformation, political intelligence*
- *Anticipation: foresight technology emergence, epidemics intelligence*
- *Web Text Mining: media monitoring and analytics*


Podcast Newsletter Shop

il POST

ECONOMIA | MARTEDÌ 6 SETTEMBRE 2022 QUESTO ARTICOLO HA PIÙ DI UN ANNO

È vero che le sanzioni alla Russia non stanno funzionando?

Example 1: Sanctions monitoring

- **Data:** information recorded by the trade operators in the customs declaration collected from the national EU authorities, including transaction weight (quantity or supplementary units), value, origin and destination of the consignment.

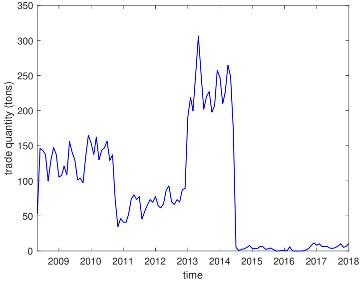
- **Product classification:** The international Harmonised System and/or the EU-internal TARIC.

- **Objective 1:** Identify structural breaks and outliers in trade time series, pointing to possible circumvention of restrictions on export to Russia.

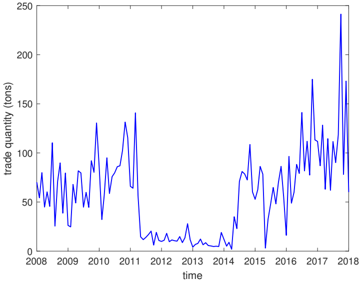
- **Objective 2:** Summarise numeric tables (possibly sparse) containing count or continuous data elements: use of co-clustering for the ranking of signals.

Example 1: Robust Monitoring of Time Series

Imports of plants from KE to UK



Imports of sugars from UA to LT

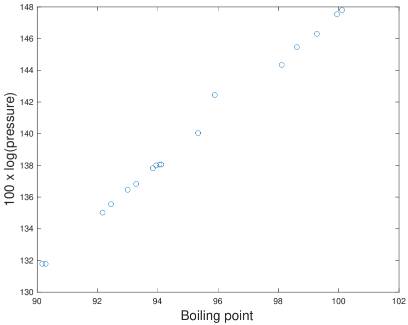


Risk-analysis/anti-fraud/monitoring purposes: identify sudden reductions or increases in trade volumes/values (*structural changes and groups of outliers*).

Statistical purpose: provide a robust unified framework to treat simultaneously outliers, level shifts, trends and seasonality $\longrightarrow$ *statistically sound signals ranking approach*.

The concept of robustness in regression

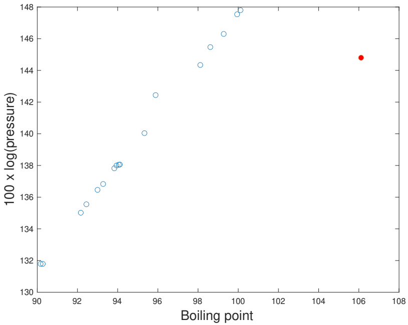
```
load('forbes.txt');
y=forbes(:,2);
X=forbes(:,1);
X = (X - 32) * 5/9; % Convert
to Celsius
plot(X,y,'o');
xlabel('Boiling
point','Fontsize',16);
ylabel('100 x
log(pressure)','Fontsize',16);
f1 = gcf ; figure(f1);
```



Forbes data: 17 observations about water boiling point (x axis) at different altitudes and therefore pressures (y axis)

The concept of robustness in regression

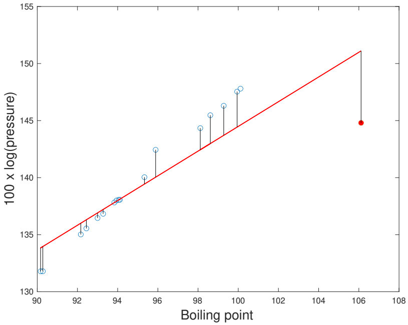
```
yc = y; yc(end) = yc(end)-3;
Xc = X; Xc(end) = Xc(end)+6;
hold on
plot(Xc(end),yc(end),'o',
'MarkerFaceColor','r');
figure(f1);
```



Forbes data + a clear outlier

The concept of robustness in regression

```
int = ones(size(Xc,1),1);
Xic = [int Xc];
beta0 = (Xic’*Xic)(Xic’*yc);
beta1 = Xic \ yc;
beta2 = regress(yc,Xic);
beta3 = fitlm(Xc,yc,’y 1+x1’);
b = beta1;
fit = @(z) b(1) + b(2)*z;
hold all
plot(Xc, fit(Xc), ’r’);
plot([Xc Xc]’ , [fit(Xc)yc]’);
```

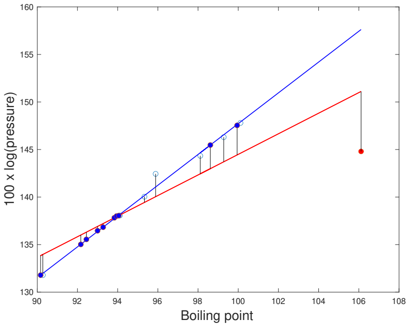


The outlier produces a considerable deviation of the Ordinary Least Squares line and therefore distorts the estimates of the model parameters.

European Commission

The concept of robustness in regression

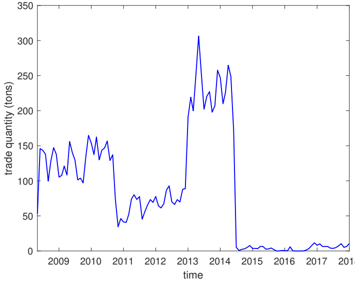
```
outLTS = LXS(yc,Xc);
% La retta di regressione LTS
b = outLTS.beta;
plot(Xc,b(1)+b(2)*Xc,'b');
% La h unita' utilizzate per
il fit da LTS
in = outLTS.weights;
plot(Xc(in),yc(in),'o',
'MarkerFaceColor' , 'b')
```



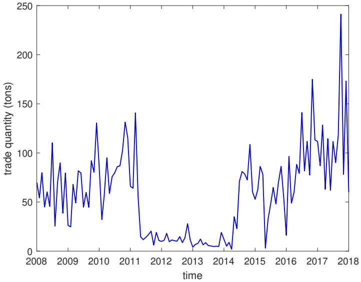
Least Trimmed Squares minimizes the sum of the squared residuals of a subset of the data: the outlier does not influence the regression line

Example 1: Robust Monitoring of Time Series

Imports of plants from KE to UK



Imports of sugars from UA to LT



Risk-analysis/anti-fraud/monitoring purposes: identify sudden reductions or increases in trade volumes/values (*structural changes and groups of outliers*).

Statistical purpose: provide a robust unified framework to treat simultaneously outliers, level shifts, trends and seasonality $\longrightarrow$ *statistically sound signals ranking approach*.

Example 1: the `LTSts.m` function

Documentation

Search Help

CONTENTS Close

‹ All Products

‹ **Flexible Statistics and Data Analysis (FSDA)**

‹ **Flexible Statistics and Data Analysis Toolbox**

‹ **Functions**

LTSts

ON THIS PAGE

◀ logmvnpdfFS LTStsLSmult ▶

LTSts

LTSts extends LTS estimator to time series

expand all in page

Syntax

```
out=LTSts(y)                         example
out=LTSts(y,Name,Value)              example
[out, varargout]=LTSts(___)          example
```

Description

It is possible to set a model with a trend (up to third order), a seasonality (constant or of varying amplitude and with a different number of harmonics) and a level shift (in this last case it is possible to specify the window in which level shift has to be searched for).

`out =LTSts(y)` Simulated data with linear trend and level shift. example

`out =LTSts(y, Name, Value)` Airline data: linear trend + just one harmonic for seasonal component. example

`[out, varargout] =LTSts(___)` Model with linear trend and six harmonics for seasonal component. example

Example 1: the `LTSts.m` model

$$y_t = \underbrace{\sum_{a=0}^{A} \beta_{a,0} t^a}_{\text{trend}} + \underbrace{\left(1 + \sum_{g=1}^{G} \gamma_g t^g\right) \left[\sum_{b=1}^{B} \left(\beta_{b,1} \cos\left(\frac{2\pi b}{12} t\right) + \beta_{b,2} \sin\left(\frac{2\pi b}{12} t\right)\right)\right]}_{\text{seasonality}} + \underbrace{\delta_1 I(t \geqslant \delta_2)}_{\text{level shift}}$$

terms originally introduced by Rousseeuw, Perrotta, Riani, Hubert (2019)

↓

new terms introduced in 2022 to address our problem

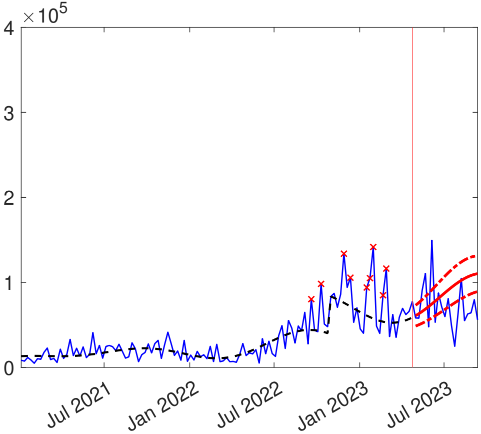
$+ \sum_{e=1}^{E} \beta_{e,3} x_{t,e}$ **covariates term**, added to incorporate multiple level shifts and other trade factors

$+ \sum_{r=1}^{R} \phi_r y_{t-r}$ **autoregressive term**, as the current value may depend on the previous ones

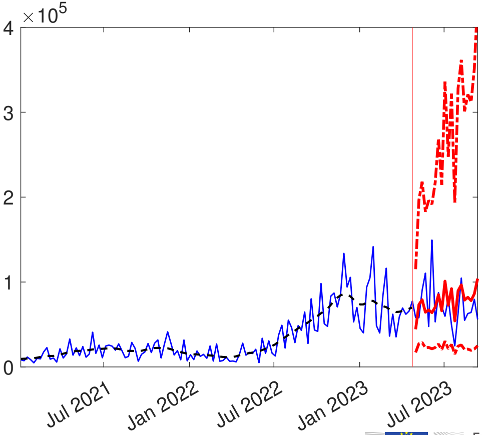
$+ \varepsilon_t$

Example 1:stability of `LTSts.m` & `forecastTS.m` under small perturbations

Export quantities (Kg) of parts and accessories of motor vehicles from EU to Kazakhstan.



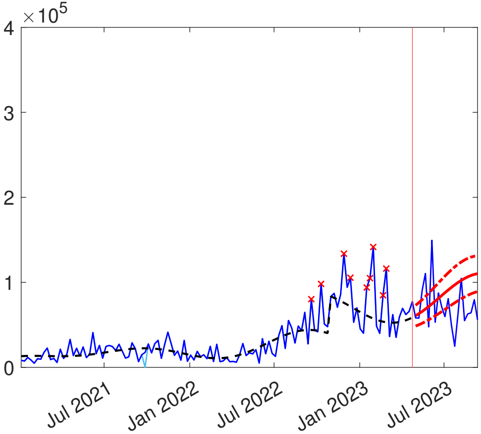
LTSts



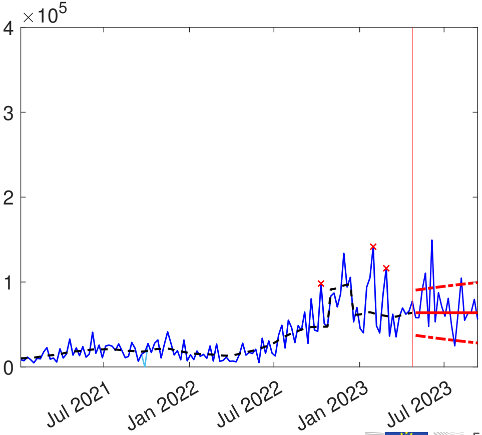
TAXUD-like

Example 1:stability of `LTSts.m` & `forecastTS.m` under small perturbations

Export quantities (Kg) of parts and accessories of motor vehicles from EU to Kazakhstan.



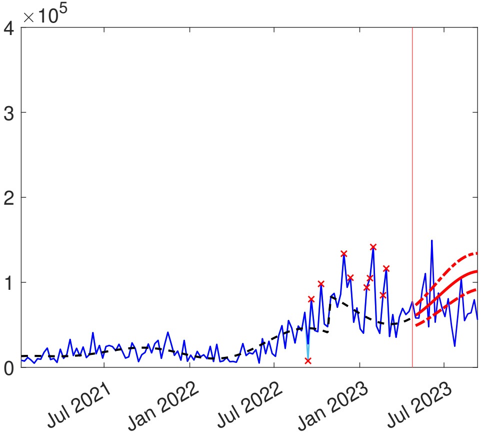
LTSts



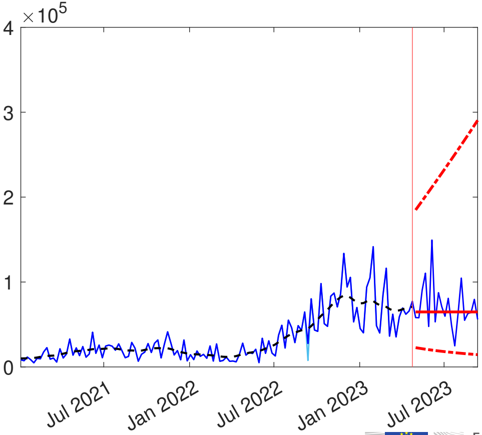
TAXUD-like

Example 1:stability of `LTSts.m` & `forecastTS.m` under small perturbations

Export quantities (Kg) of parts and accessories of motor vehicles from EU to Kazakhstan.



LTSts

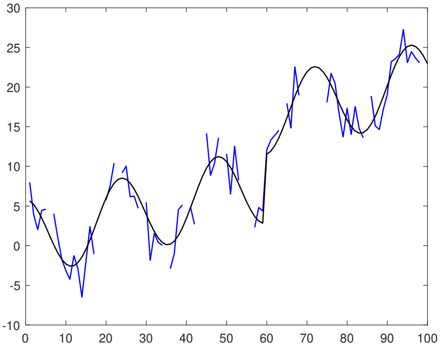


TAXUD-like

Example 1: missing values with the `LTSts.m`

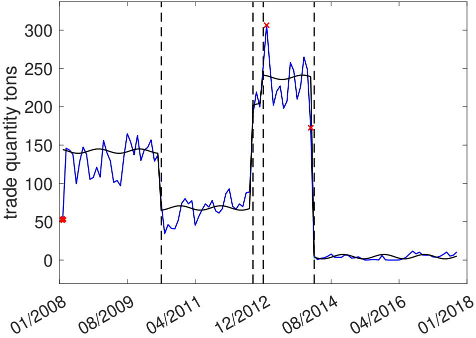
Robust estimation methods cleverly trim a fraction of data elements, excluding outliers that may severely distort results.

We exploited this property to account for missing values, by simply excluding also the missing values from the estimate.

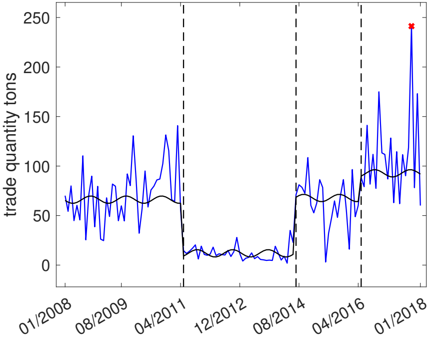


Simulated data with missing values

Example 1: multiple level shifts with `LTStsLSmult.m`



Plants from Kenya to UK



Sugars from Ukraine to Lithuania

Iterative procedure stops when the current level shift is not significant. At step t^*, the level shifts found at steps $< t^*$ are included as step functions in the **additional covariates**

Example 1: variable selection with `LTStsVarSel`

Motivation: Each product-origin series has its own complexity and requires its own model.

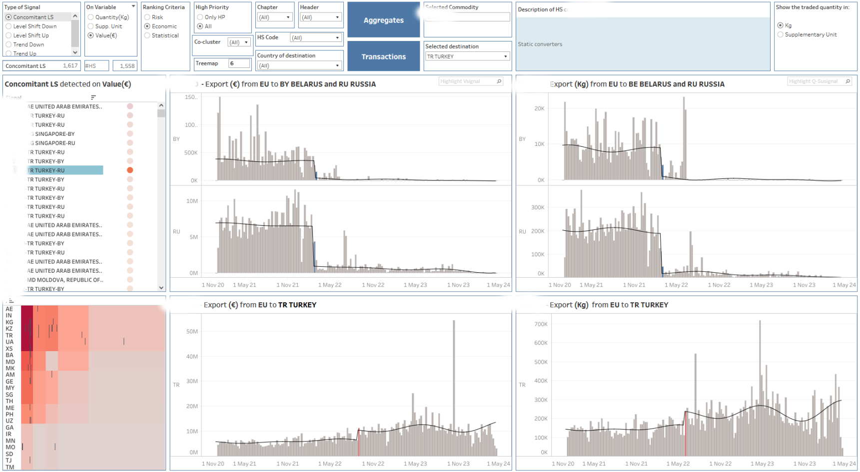
Objective: Select automatically the optimal number of model terms (A, B, G, E, R) for each trade time series.

$$
\begin{aligned}
y_t = & \sum_{a=0}^{\textcircled{A}} \beta_{a,0} t^a \quad + \delta_1 I(t \geqslant \delta_2) \\
& + \left(1 + \sum_{g=1}^{\textcircled{G}} \gamma_g t^g\right) \left[\sum_{b=1}^{\textcircled{B}} \left(\beta_{b,1} \cos\left(\frac{2\pi b}{12} t\right) + \beta_{b,2} \sin\left(\frac{2\pi b}{12} t\right)\right)\right] \\
& + \sum_{e=1}^{\textcircled{E}} \beta_{e,3} x_{t,e} \quad + \sum_{r=1}^{\textcircled{R}} \phi_r y_{t-r}
\end{aligned}
$$

Iterative procedure based on backward variable elimination:

1. we start from an over parameterized model,
2. we eliminate the least significant component,
3. we stop when no more component can be removed based on step 2.

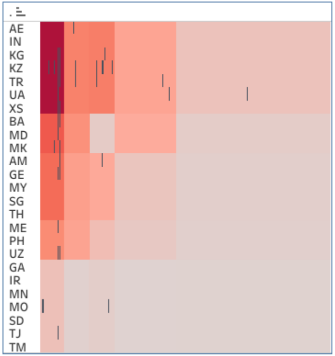
Example 1: application of `LTSts.m`



Concomitant level shift (LF): export of EU27 to RU (LS down) and Turkey (LS up) of a monitored product

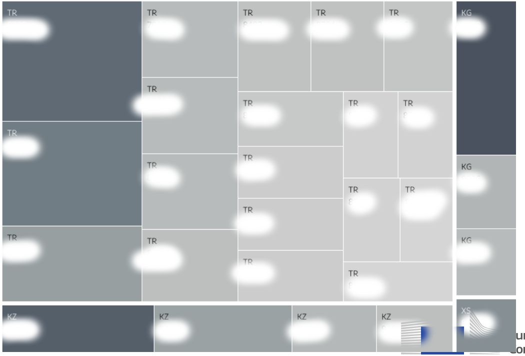
Example 2: ranking signals using co-clustering

Purpose: understand who is “facilitating” circumvention and for which commodities



CLuster group with Ranking = 1 Lambda = 8.059 (Lambda = -1 indicates an outlier)
The selected Header [illegible] and monitored Country **KZ** have **13** Signals.

Note: Only groups with at least 12 signals are dispalyed.



Example 2: co-clustering tables of signals

Country-product contingency table, containing counts of signals detected with LT-Sts.

The new robust co-clustering is to group simultaneously the rows and columns, and detect anomalous cells.

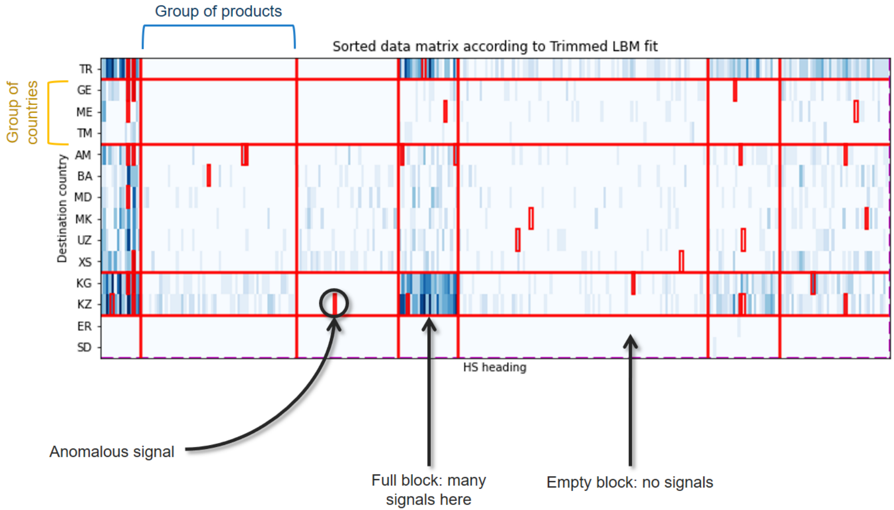
Products (HS Headings)

Destination Countries

AM	0	0	0	0	0	1	0	0	0	0	...	0	0	1	0	1	5	1	0	0	0
BA	0	0	1	0	0	2	0	1	2	0	...	0	0	0	1	0	1	0	0	1	0
ER	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
GE	0	0	0	0	2	3	0	0	0	0	...	0	0	0	0	3	1	1	0	1	0
KG	0	2	1	0	0	0	0	0	0	0	...	2	0	1	1	7	8	0	4	1	0
KZ	1	0	0	1	0	2	0	0	1	0	...	4	1	1	0	8	10	1	5	0	0
MD	0	0	0	0	0	1	1	0	0	0	...	0	0	0	0	0	2	0	0	0	0
ME	0	0	1	0	0	0	0	0	0	0	...	0	0	0	0	1	0	0	0	0	0
MK	0	0	0	0	0	1	0	0	0	0	...	0	0	0	1	1	0	0	0	0	0
SD	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
TM	0	0	0	0	1	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
TR	0	0	0	2	1	1	0	0	0	0	...	0	0	0	0	7	3	0	1	0	1
UZ	0	2	0	0	1	1	0	0	0	2	...	0	0	0	0	0	0	0	0	0	0
XS	0	0	0	0	0	3	0	0	0	2	...	3	0	0	0	0	0	0	0	1	0

Number of signals for
Country-Product combination

Example 2: co-clustering tables of signals

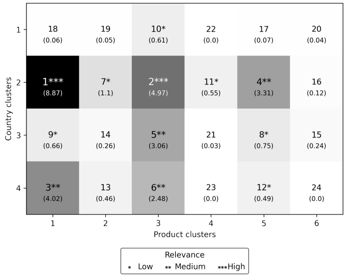


Example 2: the co-clustering model

New robust co-clustering with trimmed Latent Block Models

$$\mathcal{L}(\boldsymbol{\pi}, \boldsymbol{\rho}, \boldsymbol{\Lambda}|X) = \sum_{ik} z_{ik} \log \pi_k + \sum_{jl} w_{jl} \log \rho_l + \sum_{ijkl} z_{ik} w_{jl} M_{ij} \log f(x_{ij}|\lambda_{kl})$$

- $\boldsymbol{X} = \{X_{ij}\}_{ij}$ $n \times p$ data matrix
- $f(\cdot|\lambda_{kl})$ density of $X_{ij}|\{z_{ik}, w_{jl}\}$
- $Z \in \{0,1\}^{n \times g}$ s.t. $\sum_i z_{ik} = 1$ $\forall k$ (row partition matrix)
- $W \in \{0,1\}^{p \times m}$ s.t. $\sum_j w_{jl} = 1$ $\forall l$ (column partition matrix)
- $\Lambda = \{\lambda_{kl}\}_{kl}$: block parameters
- $\boldsymbol{\pi} \in \Delta^{g-1}$, $\boldsymbol{\rho} \in \Delta^{m-1}$: row and column mixing proportions (Δ^d: d-simplex)
- $M \in \{0,1\}^{n \times p}$ (mask matrix, $M_{ij} = 0$ means x_{ij} is excluded)



Ranking of blocks based on the estimated block parameters (inside parentheses)

Example 2: use of the MATLAB Engine

```
% # pip3 install matplotlib # to install the library from bash
% # pip3 install scipy
% pyenv(Version="/Library/Frameworks/Python.framework/Versions/

% tupl = variable to be assigned in output, i.e. tpl:=tupl
[tpl , Zuseless]= pyrunfile("call_py.py", ["tupl" , "Z"]);

htmp = heatmap(double(tpl{1}), 'ColorLimits', [0 30]);
htmp.YDisplayLabels = tpl{2};
htmp.XDisplayLabels = tpl{3};
```

Co-clustering code is being ported from Python to MATLAB, in FSDA. But it is already operational thanks to the MATLAB Engine

```
import numpy as np
import pandas as pd
import matplotlib
matplotlib.use('Agg')   # use 'Agg' non-
import matplotlib.pyplot as plt
from TCoClust.Methods import *
from TCoClust.Utils import *

# load data
df = pd.read_csv("tab.csv", index_col=0)

print("\nPreview of loaded data:\n")
print(df.head())

# transform to numpy array
X = df.to_numpy()
# get row and column lables as lists (us
row_labels = list(df.index)
col_labels = list(df.columns)
```