

Computational aspects of robustified mixed LS-TLS estimation

Jiri Franc

Czech Technical University in Prague
Faculty of Nuclear Sciences and Physical Engineering
Department of Mathematics

January 21, 2014



Introduction

Let us consider the overdetermined set of linear equations ($n > p$)

$$\mathbf{Y} \approx \mathbf{X}\beta^0,$$

where

- ▶ $\mathbf{Y} \in \mathbb{R}^{n \times 1}$ is vector of response (dependent variable),
- ▶ $\mathbf{X} \in \mathbb{R}^{n \times p}$ is matrix of predictors (independent variables) with full column rank,
- ▶ $\beta^0 \in \mathbb{R}^{p \times 1}$ is unknown vector of parameters (coefficients).



Introduction

Let us consider the overdetermined set of linear equations ($n > p$)

$$\mathbf{Y} \approx \mathbf{X}\beta^0,$$

where

- ▶ $\mathbf{Y} \in \mathbb{R}^{n \times 1}$ is vector of response (dependent variable),
- ▶ $\mathbf{X} \in \mathbb{R}^{n \times p}$ is matrix of predictors (independent variables) with full column rank,
- ▶ $\beta^0 \in \mathbb{R}^{p \times 1}$ is unknown vector of parameters (coefficients).

The aim is to

- ▶ estimate the parameter β^0 and find such a linear model that describes the structure best fitting the bulk of the data.
- ▶ find such a robust estimator, which can cope with the situation when some predictors are not error free and data set contains outliers.

Let use the theory of mixed least squares - total least squares, robust estimation based on trimming or on downweighting the influential points and mix it all together.



Mixed Least Squares - Total Least Squares problem

$$\mathbf{Y} \approx \mathbf{X}\beta, \quad \mathbf{Y} \in \mathbb{R}^n, \quad \mathbf{X} \in \mathbb{R}^{n \times p}, \quad n > p,$$

$$\begin{aligned} \text{partition } \mathbf{X} &= [\mathbf{X}^{(1)}, \mathbf{X}^{(2)}] & \mathbf{X}^{(1)} &\in \mathbb{R}^{n \times p_1}, \quad \mathbf{X}^{(2)} \in \mathbb{R}^{n \times p_2} \\ \beta^T &= [\beta^{(1)T}, \beta^{(2)T}] & \beta^{(1)} &\in \mathbb{R}^{p_1}, \quad \beta^{(2)} \in \mathbb{R}^{p_2} \end{aligned}$$

and assume that the columns of $\mathbf{X}^{(1)}$ are error free and $p_1 + p_2 = p$ then **LS-TLS estimation** is given as

$$\begin{aligned} \hat{\beta}^{(LS-TLS)} &= \min_{\beta \in \mathbb{R}^p, [\varepsilon, \Theta] \in \mathbb{R}^{n \times (p_2+1)}} \|[\varepsilon, \Theta]\|_F \\ &\text{subject to } \mathbf{Y} + \varepsilon = \mathbf{X}^{(1)}\beta^{(1)} + (\mathbf{X}^{(2)} + \Theta)\beta^{(2)}, \end{aligned}$$

where $\|\cdot\|_F$ is the Frobenius norm and rows in $[\varepsilon, \Theta]$ are *i.i.d.* and normal.

By varying p_1 from zero to p , the mixed LS-TLS problem can handle also with any ordinary LS or ordinary TLS problem.



Mixed Least Squares - Total Least Squares problem

$\mathbf{X} = [\mathbf{X}^{(1)}, \mathbf{X}^{(2)}]$ has full column rank, columns of $\mathbf{X}^{(1)}$ are error free, and $0 < p_1 < p$.
QR factorization:

$$[\mathbf{X}^{(1)}, \mathbf{X}^{(2)}, \mathbf{Y}] = \mathbf{Q} \begin{bmatrix} \mathbf{R}_{11} & \mathbf{R}_{12} & \mathbf{R}_{\mathbf{Y}_1} \\ 0 & \mathbf{R}_{22} & \mathbf{R}_{\mathbf{Y}_2} \end{bmatrix}.$$

Ordinary TLS solution $\hat{\beta}^{(TLS, p-p_1)}$ of $\mathbf{R}_{\mathbf{Y}_2} \approx \mathbf{R}_{22}\beta$ gives the last p_2 components of $\hat{\beta}^{(LS-TLS)}$.

The first p_1 components are obtained from the solution of

$$\mathbf{R}_{11}\hat{\beta}^{(LS, p_1)} = \mathbf{R}_{\mathbf{Y}_1} - \mathbf{R}_{12}\hat{\beta}^{(TLS, p-p_1)}.$$

The mixed LS-TLS solution is $\hat{\beta}^{(LS-TLS)} = [\hat{\beta}^{(LS, p_1)}, \hat{\beta}^{(TLS, p-p_1)}]$.

Unfortunately this universal estimator is not robust and gives misleading results when outliers occur.



Total Least Squares

Total Least Squares (TLS)

minimizes the sum of the squared orthogonal distances from the data points to the fitting hyperplane:

$$\hat{\beta}^{(TLS)} = \arg \min_{\beta \in \mathbb{R}^p} \frac{1}{1 + \|\beta\|^2} \sum_{i=1}^n |Y_i - X_i \beta|^2 = \arg \min_{\beta \in \mathbb{R}^p} \frac{\|\mathbf{Y} - \mathbf{X}\beta\|}{\sqrt{1 + \|\beta\|^2}}.$$

The basic stable algorithm used to solve the problem is based on the SVD (see Golub and Van Loan (1980), for generalization Van Huffel and Vandewalle (1991), Paige and Strakoš (2006)), and Hnětynková, M. Plešinger etc. (2011)



Total Least Squares

Total Least Squares (TLS)

minimizes the sum of the squared orthogonal distances from the data points to the fitting hyperplane:

$$\hat{\beta}^{(TLS)} = \arg \min_{\beta \in \mathbb{R}^p} \frac{1}{1 + \|\beta\|^2} \sum_{i=1}^n |Y_i - X_i \beta|^2 = \arg \min_{\beta \in \mathbb{R}^p} \frac{\|\mathbf{Y} - \mathbf{X}\beta\|}{\sqrt{1 + \|\beta\|^2}}.$$

The basic stable algorithm used to solve the problem is based on the SVD (see Golub and Van Loan (1980), for generalization Van Huffel and Vandewalle (1991), Paige and Strakoš (2006)), and Hnětynková, M. Plešinger etc. (2011)

Total Least Trimmed Squares (TLTS)

minimizes the sum of the h smallest squared orthogonal distances of data points to the manifold given by β :

$$\hat{\beta}^{(TLTS)} = \arg \min_{\beta \in \mathbb{R}^p} \sum_{i=1}^h d_{(i)}^2, \quad d_j = \frac{|Y_j - X_j^T \beta|}{\|[-1, \beta^T]\|},$$

where h is an optional parameter satisfying $\frac{n}{2} \leq h \leq n$ and $d_{(i)}^2$ is the i th least squared orthogonal distance, i.e. for any $\beta \in \mathbb{R}^p$ $d_{(1)}^2(\beta) \leq d_{(2)}^2(\beta) \leq \dots \leq d_{(n)}^2(\beta)$.



Robustified mixed LS-TLS

The idea to compute the robustified mixed LS-TLS estimation of the problem

$$\mathbf{Y} \approx [\mathbf{X}^{(1)}, \mathbf{X}^{(2)}] (\beta^{(1)T}, \beta^{(2)T})^T$$

is the same. We need to identify the influential points from both parts and trim (down-weight) them.

Squared orthogonal distance of j th observation from the manifold represented by $\beta^{(2)}$:

$$d_j = \frac{\left| \left(Y_j - (X_j^{(1)})^T \beta^{(LS)} \right) - (X_j^{(2)})^T \beta^{(TLS)} \right|}{\| [-1, (\beta^{(TLS)})^T]^T \|}$$

Squared vertical distance of j th observation from the manifold represented by $\beta^{(1)}$:

$$r_j = \left(\left(Y_j - (X_j^{(2)})^T \beta^{(TLS)} \right) - (X_j^{(1)})^T \beta^{(LS)} \right)^2$$



Mixed Least Trimmed Squares - Total Least Trimmed Squares

Let us denote by q_i the sum of both distances d_i (orthogonal) and r_i (vertical).

Mixed Least Trimmed Squares - Total Least Trimmed Squares (LTS-TLTS)

minimizes the sum of the h smallest distances q_i

$$\hat{\beta}^{(LTS-TLTS)} = \arg \min_{\beta \in \mathbb{R}^p} \sum_{i=1}^h q_{(i)}(\beta),$$

where h is an optional parameter satisfying $\frac{n}{2} \leq h \leq n$ and $q_{(i)}$ is the i th least mixed distance, i.e. for any $\beta \in \mathbb{R}^p$

$$q_{(1)}(\beta) \leq q_{(2)}(\beta) \leq \dots \leq q_{(n)}(\beta), \quad q_j(\beta) = d_j^2(\beta) + r_j^2(\beta), j \in \{1, 2, \dots, n\}$$



Mixed Least Weighted Squares - Total Least Weighted Squares (LWS-TLWS)

minimizes the sum of the re-weighted squared distances by some weights from $\langle 0, 1 \rangle$. The estimator is inspired by the Least Weighted Squares (see Víšek 2011).

$$\begin{aligned}\hat{\beta}^{(LWS-TLWS, w)} &= \arg \min_{\beta \in \mathbb{R}^p} \sum_{i=1}^n w \left(\frac{i-1}{n} \right) q_{(i)}(\beta) = \\ &= \arg \min_{\beta \in \mathbb{R}^p} \sum_{i=1}^n w \left(\frac{\pi(\beta, i)-1}{n} \right) q_i(\beta),\end{aligned}$$

where $\pi(\beta, i)$ is the random rank of the i -th residual, weights w_i are defined by the weight function $w : \langle 0, 1 \rangle \rightarrow \langle 0, 1 \rangle$ and satisfy certain conditions.



Properties of Mixed LTS-TLTS estimator

Proven properties:

- ▶ The existence of the LTS-TLTS problem is given by the existence of the classical LS-TLS estimation for any subsample of size h .
- ▶ TLTS is so called half-sample estimator and it has 50% breakdown points (LTS-TLTS should be too).
- ▶ Objective function is continuous, nonconvex, non-differentiable and has multiple local minima, whose number commonly rises with the number of observations and unknowns.

Expected properties:

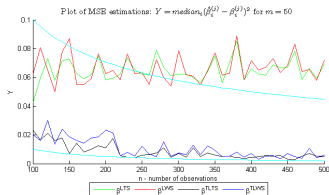
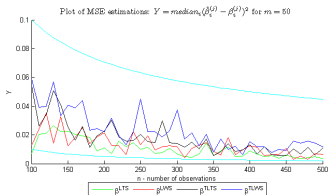
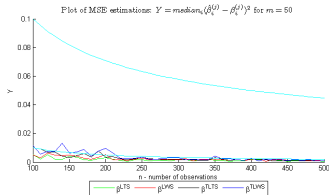
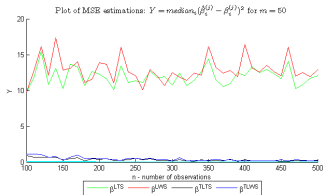
- ▶ It is supposed that the LTS-TLTS estimator is consistent, because Empirical Mean Square Errors from simulated examples tends to zero. Theoretically it will be hopefully proven soon.



Properties of Mixed LTS-TLTS estimator

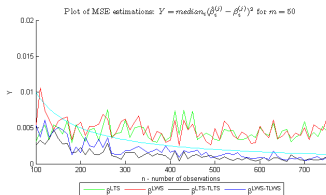
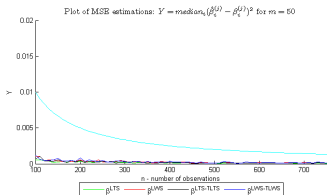
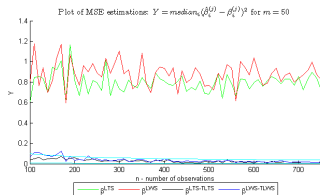
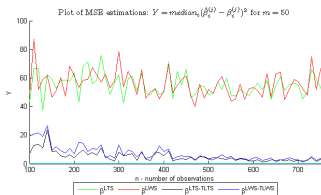
behavior of the estimators for large n

Plot of median squares errors $Y = \text{median}_i(\hat{\beta}_i^{(j)} - \beta_i^{(j)})^2$ for varying n , $m = 50$. Outliers = 25%, Trimming = 30%, Regressors without error free columns.



Behavior of the estimators for large n

Plot of median squares errors $Y = \text{median}_i(\hat{\beta}_i^{(j)} - \beta_i^{(j)})^2$ for varying n , $m = 50$. Outliers = 25%, Trimming = 30%, Regressors with error free columns.



Algorithms for robustified mixed LS-TLS estimation

LTS-TLTS:

- ▶ Exact algorithm based on evaluation of all $\binom{n}{h}$ computations of LS-TLS works in practice only if the number of observations is less than 20.
- ▶ Implemented non-exhaustive exact algorithms are Branch-and-Bound algorithm (BAB) and Borders Scanning Algorithm (BSA).
- ▶ Approximative algorithms for larger data sets (circa $n > 60$) with more observations and unknowns, with best ratio between achievement and price (computation time) is FAST LTS-TLTS algorithm based on Rousseeuw and Van Driessen resampling algorithm for LTS and k-opt algorithm.
- ▶ Another approximative algorithms are based on theory of simulating annealing (Metropolis-Hastings algorithm) or genetics algorithms.

LWS-TLWS:

- ▶ Exhaustive algorithm for LWS-TLWS needs $n!$ steps and is very impractical.
- ▶ For LWS-TLWS we do not have any non-exhaustive exact algorithms.
- ▶ Among approximative algorithms, the re-weighted concentration algorithm (based on Rousseeuw algorithm) is the fastest and most accurate one.

All these algorithms were implemented in MATLAB.



FAST LTS-TLTS algorithm

approximative algorithm

For $k=1$ to number of iteration do:

1. Pick randomly $(p + 1)$ data points and compute ordinary LS-TLS estimate $\hat{\beta}^{(LS-TLS, p+1)}$.
2. Compute the distance q_i for all n data points.
3. Select the h data points with the smallest distances.
4. Compute ordinary LS-TLS estimate $\hat{\beta}^{(LS-TLS, h)}$ for selected h data points.
5. Repeat steps 2-4 until convergence.
6. If the value of the objective function is the smallest one among the values, that have been reached up to this moment, store the appropriate estimation as a LS-TLS estimate.

Properties:

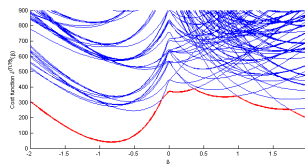
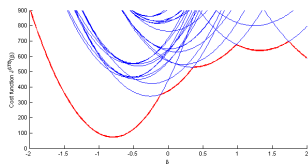
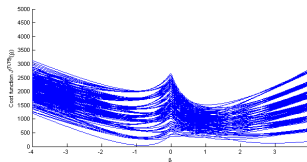
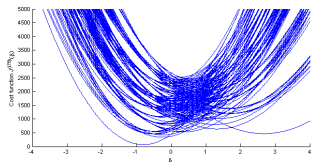
- ▶ The algorithm usually finds a local minimum which is close to the global minimum, but not necessarily equal to that global minimum.
- ▶ Hawkins and Olive (2002) showed that elemental concentration algorithms, where the number of concentration steps is finite, are zero breakdown and resampling estimators are zero breakdown and inconsistent.



BSA - Borders scanning algorithm

exact algorithm

The BSA algorithm was firstly introduced for LTS by Karel Klouda in 2007 and is based on scanning of the objective function.



The graph of optional function (red bold line) for LTS and TLTS estimation on data with $n = 10$ observations, $\rho = 1$ and trimming parameter $h = 6$.



Borders scanning algorithm

exact algorithm

The idea of the algorithm is to find all compositions of the objective function, in given part find the local minimum and the global minimum must be among them. We want to find set

$$\mathcal{H} = \{ \beta \in \mathbb{R}^p \mid \exists i, j \in \{1, 2, \dots, n\}, q_{(h)}(\beta) = q_i(\beta) = q_j(\beta) = q_{(h+1)}(\beta) \} .$$

where again $q_j = d_j^2 + r_j^2$ is the sum of the j th squared orthogonal distance d_j and the j th squared vertical distance r_j and

$$q_{(1)}(\beta) \leq q_{(2)}(\beta) \leq \dots \leq q_{(n)}(\beta).$$

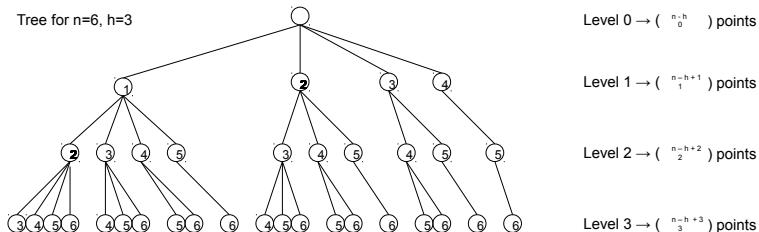
We are looking for a set containing such a β 's that give a hyperplanes which divide the distance between the h th and $(h+1)$ th most distant points from a given hyperplane into two halves.



BAB - Branch and Bound algorithm

exact algorithm

The algorithm is inspired by BAB algorithm for LTS presented by José Agulló (2001) and guarantees global optimality. The algorithm passes through the tree with h levels, $(n - h + 1)$ roots and $\binom{n}{h}$ terminal nodes. The tree has at the level m , where $m < h$, $\binom{n-h+m}{m}$ number of nodes.



Tree for 6 observations and coverage equal to 3.



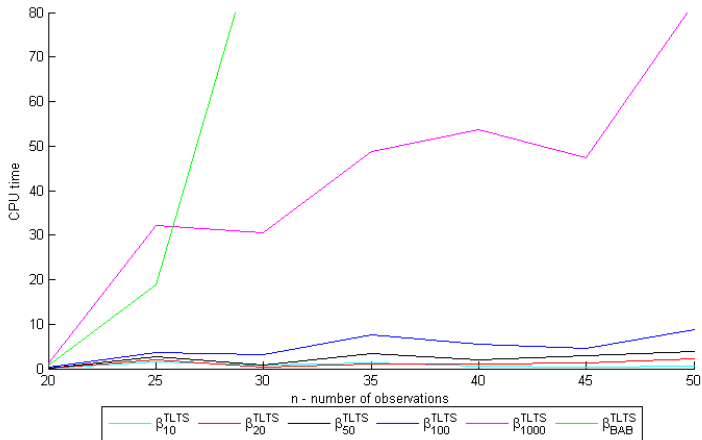
BAB - Branch and Bound algorithm

1. Compute initial LTS-TLTS estimation $\hat{\beta}^{(init,n)}$ via any fast approximative algorithm and evaluate its objective function $S(\hat{\beta}^{(init,n)}) = \sum_{i=1}^h q_{(i)}$ and set the $S(\hat{\beta}^{(LTS-TLTS,n)}) = S(\hat{\beta}^{(init,n)})$.
2. For given number of nodes at the level g , $j \in 1 \dots (n-h+g)$ we compute ordinary LS-TLS estimate and evaluate its objective function $S(\hat{\beta}^{(LTS-TLTS,g,j)}) = \sum_{i=1}^g q_i$.
3. If $S(\hat{\beta}^{(LTS-TLTS,g,j)}) > S(\hat{\beta}^{(LTS-TLTS,n)})$ then cut all children of given node, move to the next node and repeat step 2.
4. If $S(\hat{\beta}^{(LTS-TLTS,g,j)}) < S(\hat{\beta}^{(LTS-TLTS,n)})$ and $g < h$ then move to the child of given node and repeat step 2.
5. If $S(\hat{\beta}^{(LTS-TLTS,g,j)}) < S(\hat{\beta}^{(LTS-TLTS,n)})$ and $g = h$ then set $S(\hat{\beta}^{(LTS-TLTS,n)}) = S(\hat{\beta}^{(LTS-TLTS,h,j)})$ and move to the next upper node.



Speed and quality comparison

Median of CPU time depending on different algorithms and number of observations.



Simulation study - comparing BSA with others algorithms

Results of simulation study of the experiment with $p = 3$, $p_1 = 2$, $p_2 = 1$, data contains intercept, n is varying, 25% outliers, trimming level $h = 0.7n$. FAST operates with 1000 iterations and starting level of BAB is $\lceil h/4 \rceil$, number of repetition for each n is only 5.

n	times_median 1.0e+3			n_tls_comps_median			EMSE 1.0e+2	
	FAST	BSA	BAB	FAST	BSA	BAB	FAST	exact
15	0.0068	0.0037	0.0006	40407	5148	501	8.6300	0.2811
20	0.0070	0.0092	0.0010	41432	8442	1674	3.0361	0.1320
25	0.0080	0.0237	0.0043	47180	20214	19467	1.1849	0.6110
30	0.0084	0.0471	0.0468	49266	32382	263979	0.6026	0.2911
35	0.0103	0.0902	0.0835	59821	60876	472936	0.2579	0.1803
40	0.0092	0.1480	0.2758	52831	79434	1596996	0.3416	0.1635
45	0.0102	0.2370	4.6032	58022	116460	16330600	0.1765	0.1016

Computation was running in MATLAB on one core of Intel i5-3210M CPU.



Simulation study - comparing BSA with others algorithms

Results of simulation study of the experiment with $p = 5$, $p_1 = 3$, $p_2 = 2$, with intercept, varying n , 25% outliers, trimming level $h = 0.7n$. FAST operates with 1000 iterations and starting level of BAB is $\lceil h/4 \rceil$, number of repetition for each n is only 5.

n	times_median 1.0e+4			n_tls_comps_median			EMSE 1.0e+3	
	FAST	BSA	BAB	FAST	BSA	BAB	FAST	exact
15	0.0006	0.0095	0.0000	32957	211596	279	3.3199	2.2971
20	0.0007	0.0404	0.0001	41741	661485	2300	0.1345	0.0714
25	0.0007	0.1781	0.0007	41339	2780778	33727	2.6977	0.0835
30	0.0009	0.4620	0.0026	49372	5367450	141243	0.2759	0.0799
35	0.0009	1.2064	0.0245	51742	12978999	1401634	1.0107	0.0382
40	0.0010	2.6130	0.0338	56216	23414160	1933190	0.8178	0.0273
45	0.0012	5.5081	0.5177	60711	45259467	29251432	0.0395	0.0196

Computation was running in MATLAB on one core of Intel i5-3210M CPU.



Real data sets analysis

Real data sets are from Leroy (1987) and let us denote by

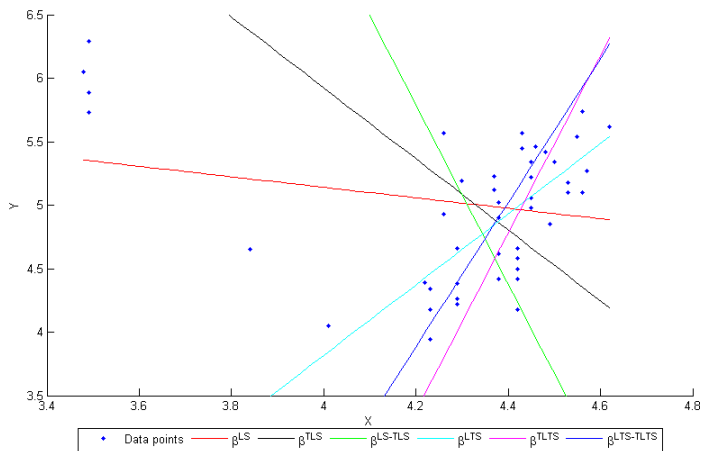
- ▶ **"Stars"** - the Hertzsprung-Russell Diagram of the Star Cluster CYG OB1.
- ▶ **"Wood"** - the modified Wood Gravity Data with five independent variables and intercept.
- ▶ **"Brain"** - we denote Mammal brain weights data with 28 observations.

Computational time of LTS-TLTS for real data sets					
Data	n	p	h	time in seconds	
				BSA	BAB
Stars	47	2	0.8n	4.042	4.973
Wood	20	6	0.6n	235.546	0.187
Brain	28	2	0.8n	2.044	0.515



Real data sets analysis

Estiamtion of the Hertzsprung-Russell Diagram of the Star Cluster CYG OB1



Conclusion

From the simulations and analysis we can conclude following statements:

- ▶ Mixed LTS-TLTS is very powerful robust estimator for linear problems, where some predictors are measured with random errors and outliers occur.
- ▶ We have several algorithms for computation of LTS-TLTS estimation.
- ▶ BSA algorithm can be used for data sets with $p < 4$ and $n < 70$, and the computation time depends more on the number of predictors p than BAB.
- ▶ BAB algorithm can be used for data sets with $n < 60$ and the computation time grows rapidly with growing number of observations.

Future work:

- ▶ Proofs of theoretically properties of LTS-TLTS.
- ▶ Parallelizing of mentioned algorithms and improved computational speed.
- ▶ Research into categorical variables.



Thank you for attention.

Literature:

- ▶ Golub, G. and Van Loan, C., *An analysis of the total least squares problem*, 1980, SIAM J. Numerical Analysis, 17.
- ▶ Van Huffel, S. and Vandewalle, J., *The Total Least Squares Problem: Computational Aspects and Analysis*, 1991, SIAM, Philadelphia.
- ▶ Agulló, J., *New algorithms for computing the least trimmed squares regression estimator*, 2001, Computational Statistics and Data Analysis", volume 36, nr. 4.
- ▶ Hawkins, D. M., and Olive, D. J., *Inconsistency of Resampling Algorithms for High Breakdown Regression Estimators and a New Algorithm*, Journal of the American Statistical Association, 2002.
- ▶ A. Kukush and S. Van Huffel. *Consistency of elementwise-weighted total least squares estimator in a multivariate errors-in-variables model $ax \approx b$* . Metrika, 59(1):75-97, 2004.
- ▶ Rousseeuw, P. J. and Van Driessen, K., *Computing LTS regression for large data sets*, Data Mining and Knowledge Discovery, 2006.
- ▶ Paige, C. C. and Strakoš, Z., *Core problems in linear algebraic systems*, SIAM Journal on Matrix Analysis and Applications 27, 2006.
- ▶ Klouda, K., *BSA - exact algorithm computing LTS estimate*, arXiv:1001.1297, 2010.
- ▶ Víšek, J. A., *Consistency of the least weighted squares under heteroscedasticity* Kybernetika, 2011.
- ▶ I. Hnětynková, M. Plešinger, Sima M., Z. Strakoš, and S. Van Huffel. *The total least squares problem in $ax \approx b$. a new classification with the relationship to the classical works*. SIAM Journal on Matrix Analysis and Applications, 32:748-770, 2011.
- ▶ Cook W. J. *In Pursuit of the Traveling Salesman: Mathematics at the Limits of Computation*. Princeton University Press, Princeton, New Jersey, 2011.



Computational aspects of robustified mixed LS-TLS estimation

Jiri Franc

Czech Technical University in Prague
Faculty of Nuclear Sciences and Physical Engineering
Department of Mathematics

January 21, 2014

