LEAST SQUARES BASED METHODS FOR THE ESTIMATION OF A SPECTRUM BACKGROUND

V. Mazet [*] **J. Idier** [**] **D. Brie** [*] **B. Humbert** [***]
C. Carteret [***]

[*] *CRAN, CNRS UMR 7039,*
Université Henri Poincaré, Nancy 1,
BP 239, F-54506 Vandœuvre-lès-Nancy Cedex
Tel.: +33 3 83 68 44 61, Fax: +33 3 83 68 44 62
e-mail: vincent.mazet@cran.uhp-nancy.fr
[**] *IRCCyN, CNRS UMR 6597, École Centrale de Nantes,*
1 rue de la Noë, BP 92101, F-44321 Nantes Cedex 3
[***] *LCPME, CNRS UMR 7564, Université Henri Poincaré,*
405 rue de Vandoeuvre, F-54600 Villers-lès-Nancy

Abstract: This article presents different methods to estimate the background of spectra (in particular Raman or infrared spectra). These methods are based on the least squares estimation: reweighted least squares that enables to minimise a cost function which is not a quadratic (we choose an hyperbolic function, the Cauchy function and a truncated quadratic) and least trimmed squares (which computes a least squares estimation on a subset of the sample set). For the last approach, we also propose a new algorithm which performs well but faster than the FAST-LTS of (Rousseeuw and Van Driessen, 1999). The performances of the different methods are compared on simulated and real data.

Keywords: spectroscopy, background estimation, least squares, reweighted least squares, least trimmed squares.

1. INTRODUCTION

In spectroscopy (like Raman inelastic diffusion or infrared absorption spectroscopy), the interpretation can be hampered by the background (see figure 3 for a simulation example, and figures 6 and 7 for a real spectrum). This work proposes some methods to estimate the background by a low-order polynomial. Subtracting the background yields a more interpretable signal.

Spectra are the sum of three signals:

- a sum of pikes (Gaussians, Lorentzians, ...) with different locations, widths and amplitudes, which contains the relevant information for chemists;
- a background (*e.g.* in the case of a Raman spectrum: Rayleigh diffusion signals, fluorescence phenomenons), modelled by a p-order polynomial;
- a noise, supposed to be Gaussian and additive, which gathers misfit errors and model uncertainties.

In the sequel, $\boldsymbol{b} \in \mathbb{R}^N$ will denote the background, and $\boldsymbol{n} \in \mathbb{R}^N$ the sum of the two other signals (pikes and noise). The measured data are given by $\boldsymbol{y} = \boldsymbol{b} + \boldsymbol{n}$. The polynomial coefficients representing

the background are gathered into $\boldsymbol{a} = (a_0 \cdots a_p)^T$. N is the signal length, and $\boldsymbol{t} = (t_1 \cdots t_N)^T$ the samples of the evolution variable (corresponding to wavenumber in our application). Then,

$$\boldsymbol{b} = \boldsymbol{T}\boldsymbol{a},$$

where

$$\boldsymbol{T} = (\boldsymbol{t}^0 \cdots \boldsymbol{t}^p) = \begin{pmatrix} 1 & t_1 & t_1^2 & \cdots & t_1^p \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & t_N & t_N^2 & \cdots & t_N^p \end{pmatrix}.$$

The aim of the presented methods is to find an estimate $\widehat{\boldsymbol{a}}$ of $\boldsymbol{a}$, which yields an estimated background $\widehat{\boldsymbol{b}} = \boldsymbol{T}\widehat{\boldsymbol{a}}$.

There are few methods for background estimation from measured spectra. (Kneen and Annegarn, 1996), for example, use digital filtering. Other papers use cubic splines and Bayesian approach (Gulam Razul *et al.*, 2003; von der Linden *et al.*, 1999), but these methods are hard to implement and require a high computational burden.

This paper presents in section 2 the least squares method, in order to introduce notations and the following methods. In section 3 we develop the reweighted least squares which enables to minimise a cost function which is not a quadratic. Then, section 4 presents the least trimmed squares: first the FAST-LTS algorithm developed by (Rousseeuw and Van Driessen, 1999), and second, a new algorithm which performs better (in terms of simplicity and time computation) for our problem. Section 5 shows results on simulated spectra and real data, and compares the different methods. At last section 6 concludes the paper and gives future directions of research.

2. LEAST SQUARES

The *least squares* (LS) method is the simpliest; in return the estimations are not satisfactory. This is due to the fact that this method tends to minimise the mean square error of the whole data points, including those corresponding to pikes. It is presented here to introduce the other methods.

The LS estimator minimises:

$$\mathcal{J}(\boldsymbol{a}) = ||\boldsymbol{y} - \boldsymbol{b}||^2 = (\boldsymbol{y} - \boldsymbol{T}\boldsymbol{a})^T(\boldsymbol{y} - \boldsymbol{T}\boldsymbol{a}),$$

which yields the explicit solution:

$$\widehat{\boldsymbol{a}} = (\boldsymbol{T}^T\boldsymbol{T})^{-1}\boldsymbol{T}^T\boldsymbol{y}.$$

A probabilistic interpretation shows that the LS estimator is the maximum likelihood estimator assuming that $\boldsymbol{n}$ is a white gaussian noise:

$$\widehat{\boldsymbol{a}} = \arg\max_{\boldsymbol{a}} p(\boldsymbol{y}|\boldsymbol{a})$$

where $p(\boldsymbol{y}|\boldsymbol{a}) = p(\boldsymbol{n})$. This shows the limits of the LS approach because $\boldsymbol{n}$ is clearly not a white Gaussian noise (recall it is a mixture of pikes and Gaussian noise).

3. REWEIGHTED LEAST SQUARES

The *reweighted least squares* (RLS) method is an iterative technique that enables to optimise a cost function which is not a quadratic.

Indeed, a quadratic gives a too high cost to pikes. This explains why the LS performs poorly because in our application, the pike cost should not be too high, unless the background estimation will be greatly affected by these points. That is why we propose to use three other cost functions (see figure 1):

- the hyperbolic function:
$$\varphi(x) = 2\left(\sqrt{s^2 + x^2} - s\right);$$
- the Cauchy function:
$$\varphi(x) = \ln\left(s^2 + x^2\right) - \ln s^2;$$
- the truncated quadratic:
$$\varphi(x) = \begin{cases} x^2 & \text{if } |x| < s, \\ s^2 & \text{otherwise.} \end{cases}$$

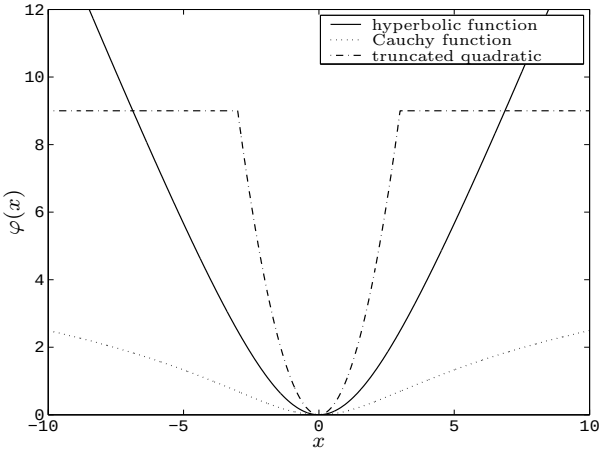

Figure 1. Cost Functions for the RLS.

All these functions are similar to a quadratic function in the neighbourhood of zero (corresponding to a Gaussian noise) but they grow more slowly than a quadratic as x increases, giving a lower cost to high values. In return, contrary to the LS estimator, the optimisation is not straightforward. We will consider separately the case of the hyperbolic and Cauchy functions, and the case of the truncated quadratic.

3.1 Case of the Hyperbolic and Cauchy Functions

The criterion to minimise is [1]:

[1] In the sequel, $\boldsymbol{v}_i$ will represent the i^{th} element of vector $\boldsymbol{v}$ and $\boldsymbol{M}_{i,j}$ the element (i, j) of the matrix $\boldsymbol{M}$.

$$\mathcal{J}(\boldsymbol{a}) = \sum_{i\in\{1,\dots,N\}} \varphi(\boldsymbol{y}_i - (\boldsymbol{T}\boldsymbol{a})_i) = \sum_{i\in\{1,\dots,N\}} \varphi(\boldsymbol{r}_i)$$

where φ is either the hyperbolic function or the Cauchy function.

To perform the minimisation of $\mathcal{J}(\boldsymbol{a})$, we follow (Idier, 2001) by using *half-quadratic regularisation* which introduces an auxiliary variable $\boldsymbol{\delta}$ in order to make the optimisation of the criterion easier. This yields a new criterion $\mathcal{K}(\boldsymbol{a}, \boldsymbol{\delta})$, which has the same global minimum as $\mathcal{J}$:

$$\inf_{\boldsymbol{\delta}} \mathcal{K}(\boldsymbol{a}, \boldsymbol{\delta}) = \mathcal{J}(\boldsymbol{a}),$$

if and only if the function φ satisfies the following conditions:

φ is even;
φ is continuous near zero and $\mathcal{C}^1$ on $\mathbb{R}^*$;
$\varphi(\sqrt{\cdot})$ is concave on $\mathbb{R}^+$.

The checking of these conditions for Cauchy and hyperbolic functions is trivial. For the third condition, we recall that a function $f(\cdot)$ is concave if $f''(\cdot) < 0$.

This allows us to use as an augmented version of the criterion $\mathcal{J}$ termed as the Geman & Reynolds criterion (Idier, 2001; Geman and Reynolds, 1992):

$$\begin{aligned}\mathcal{K}_{\mathrm{GR}}(\boldsymbol{a}, \boldsymbol{\delta}) &= \sum_i \left(\boldsymbol{\delta}_i(\boldsymbol{y}_i - (\boldsymbol{T}\boldsymbol{a})_i)^2 + \psi(\boldsymbol{\delta}_i)\right) \\ &= \sum_i \left(\boldsymbol{\delta}_i \boldsymbol{r}_i^2 + \psi(\boldsymbol{\delta}_i)\right)\end{aligned}$$

where ψ is defined from φ through:

$$\psi(\delta) = \sup_{\boldsymbol{r}_i}(\varphi(\boldsymbol{r}_i) - \delta \boldsymbol{r}_i^2).$$

In (Charbonnier *et al.*, 1997), the iterative algorithm ARTUR is proposed to perform the minimisation of $\mathcal{K}_{\mathrm{GR}}$. It estimates alternatively $\boldsymbol{a}$ and $\boldsymbol{\delta}$, and works as follows:

1. $\mathcal{K}_{\mathrm{GR}}$ is minimised as a quadratic function of $\boldsymbol{a}$, $\boldsymbol{\delta}$ being constant. The resulting equation is:
$$(\boldsymbol{T}^T \boldsymbol{\Delta} \boldsymbol{T})\boldsymbol{a} = \boldsymbol{T}^T \boldsymbol{\Delta} \boldsymbol{y}$$
where $\boldsymbol{\Delta} = \mathrm{diag}\{\boldsymbol{\delta}_1 ... \boldsymbol{\delta}_N\}$;
2. $\mathcal{K}_{\mathrm{GR}}$ is minimised as a function of $\boldsymbol{\delta}$, which yields:
$$\boldsymbol{\delta}_i = \begin{cases} \delta_\infty = \lim_{\boldsymbol{r}_i \to 0} \varphi'(\boldsymbol{r}_i)/2\boldsymbol{r}_i & \text{if } \boldsymbol{r}_i = 0, \\ \varphi'(\boldsymbol{r}_i)/2\boldsymbol{r}_i & \text{otherwise.} \end{cases}$$

The convergence is supposed to be reached when the criterion gradient becomes lower than a predefined value ε, and then the algorithm stops. The criterion gradient is simply calculated by making the difference between the value of the criterion at the last and current iterations.

The expression of $\mathcal{K}_{\mathrm{GR}}$ shows that the first part of the criterion corresponds to a weighted LS estimator. The iterative resolution of the optimisation problem yields a *reweighted* least squares method since the RLS estimator is defined as a weighted LS where the weights depends on residuals from a preliminary estimation (Dollinger and Staudte, 1991).

3.2 Case of the Truncated Quadratic

Because the function φ does not satisfy the conditions on $\mathbb{R}$ for using ARTUR, we propose to minimise the criterion

$$\mathcal{J}(\boldsymbol{a}) = \sum_{i\in\{1,\dots,N\}} \varphi(\boldsymbol{y}_i - (\boldsymbol{T}\boldsymbol{a})_i)$$

by using a *graduated non convexity* (GNC) principle, introduced by (Blake and Zisserman, 1987). It consists in a deterministic relaxation algorithm, which approximates the function φ with a sequence of functions φ_c such that $c_i \in [0, 1]$ follows a deterministic function.

- for $c = 0$, φ_0 is convex;
- for $c = 1$, $\varphi_1 = \varphi$;
- for $c_i \in]0, 1]$, a relaxed solution $\widehat{x}^i$ is calculated by minimizing locally, initialized by the previous solution $\widehat{x}^{i-1}$, *i.e.*:
$$\widehat{x}^i = \underset{x\in V(\widehat{x}^{i-1})}{\arg\min}\ \varphi_{c^i}(x),$$
where $V(\widehat{x}^{i-1})$ means neighbourhood of $\widehat{x}^{i-1}$.

Note that there is no theoretical ground about GNC, however it seems to works in many applications (Blake and Zisserman, 1987).

We propose to use the sequence of functions:

$$\varphi_c(x) = \begin{cases} x^2 & \text{if } |x| < s, \\ 2(1-c)(s|x| - s^2) + s^2 & \text{otherwise.} \end{cases}$$

These functions are convex for $c = 0$ and equal to φ for $c = 1$ (see figure 2).

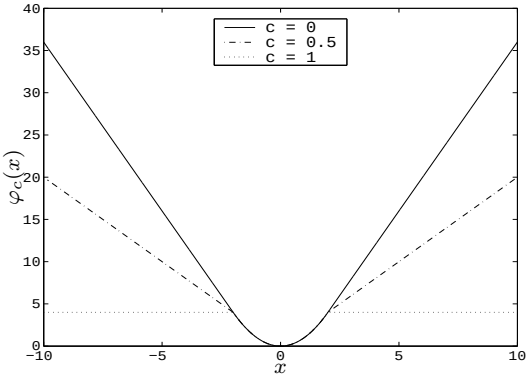

Figure 2. Plot of φ_c for Different c ($s = 2$).

To perform the minimization of the criterion for each c, we propose to use ARTUR. Indeed, in the neighbourhood of $\widehat{x}^{i-1}$ the function φ_c is expected to verify the three conditions. Moreover, $\widehat{x}^{i-1}$ is first

initialised with ARTUR on $\mathbb{R}$ as the minimum of a convex criterion. The case of a truncated quadratic corresponds also to a RLS estimator.

4. LEAST TRIMMED SQUARES

The *least trimmed squares* (LTS) estimator (Rousseeuw and Van Driessen, 1999; Rousseeuw and Bassett Jr., 1991), introduced by Rousseeuw in 1984, is a LS estimator on a h-subset H ($(N+p+1)/2 \leq h \leq N$).

The LTS method is an iterative algorithm, which, starting from a subset H, computes the LS estimation, then redefines H to eliminate the samples far from the background. The associated cost function appears to be a quadratic for samples in H, and a constant function for the others, *i.e.* a truncated quadratic. In fact, the main difference between RLS and LTS approach comes from the way the criterion optimisation is achieved. While the RLS method is a functional optimisation approach, the LTS method uses a combinatory optimisation approach.

Two algorithms are presented here: the FAST-LTS algorithm (Rousseeuw and Van Driessen, 1999), and a new method, simpler and faster in the case of our application.

4.1 The FAST-LTS Algorithm

One of the keys of this algorithm is the fact that starting from any LTS estimation, it is possible to compute a better estimation in the sense of a cost function Q. Q is a function of the residuals $\boldsymbol{r}_i = \boldsymbol{y}_i - \boldsymbol{b}_i$:

$$Q = \sum_{i \in H} \boldsymbol{r}_i^2.$$

This step is called *C-step* for "concentration step", and is described as follows:

- starting from a h-subset H_1,
- compute the LS estimator on H_1,
- compute the residuals: $\forall i = \{1, ..., N\}, \boldsymbol{r}_i = \boldsymbol{y}_i - \boldsymbol{b}_i$,
- define the new h-subset H_2 as the the h-subset corresponding to the lowest residuals.

The FAST-LTS algorithm works as follows:

1. `create a great number (say 500)` h-`subset` H`,`
2. `carry out two C-step for each subset,`
3. `select the ten best results (that is those with lowest` Q`),`
4. `compute C-step for these ten results until convergence,`
5. `report the solution with lowest corresponding` Q`.`

(Rousseeuw and Van Driessen, 1999) indicates that two C-steps are enough in step 2 to select the best subsets. Moreover, they propose a method for step 1. Instead of drawing a random subset H, they use another method, detailed in (Rousseeuw and Van Driessen, 1999, 3.1(b)).

4.2 An Other Algorithm

We present now a new algorithm, well-suited for our application, simpler and faster than FAST-LTS. The key idea is to compute the LS estimator on a subset H whose length reduces at each iteration. The algorithm works as follows:

Initialisation

H `is initialised as the whole set` ($\mathrm{Card}(H) = N$)`,`

While $\mathrm{Card}(H) \geq h$**:**

1. `compute the LS estimator on` H`,`
2. `compute the residuals for` $i \in \{1, ..., N\}$`,`
3. H `is updated,` *i.e.* `it contains only the samples with lowest residuals, such as` $\mathrm{Card}(H)$ `decreases of one element at each iteration.`

Because it involves only $N - h$ LS estimations, this algorithm is very fast as compared to the FAST-LTS.

4.3 How to Choose h?

h corresponds to the sample number used for the LS estimation. Then it is obvious to choose h as the sample number corresponding to the background, the $N-h$ other samples being pikes. Following (Rousseeuw and Van Driessen, 1999), we have chosen $h = (N+p+1)/2$ for the two algorithms because this value corresponds to the highest breakdown value.

5. RESULTS

5.1 Simulation Results

Some simulations were carried out to compare the different methods. The spectrum was simulated as it was described in the introduction. Figure 3 shows a realisation of the simulation. The different methods have to estimate a 6-order polynomial, while the simulation created a 5-order polynomial background.

We do not show here the LS estimation, because of its bad result. On the contrary, figure 4 shows the RLS estimations for an hyperbolic cost function, a Cauchy function and a truncated quadratic. The

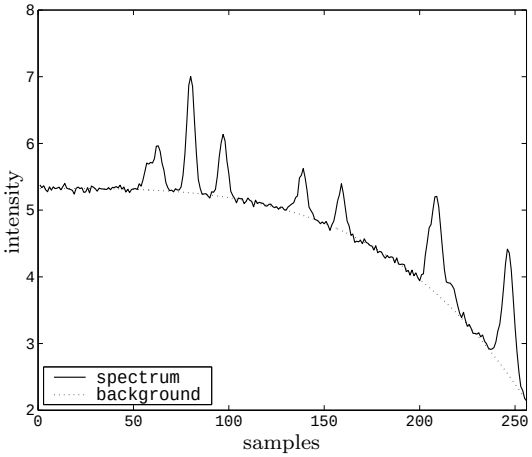

Figure 3. Realisation of a Simulated Spectrum.

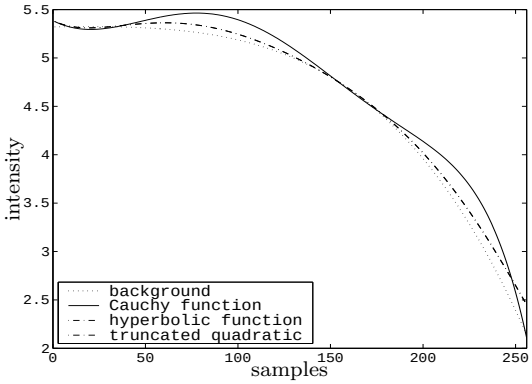

Figure 4. RLS Estimations.

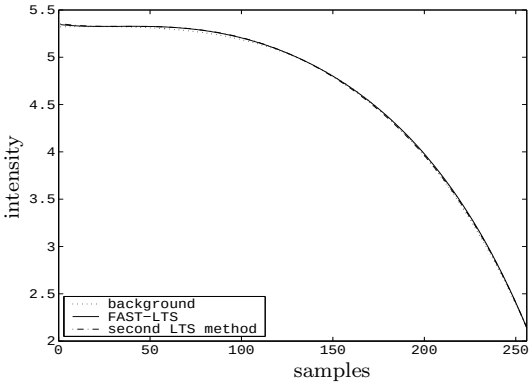

Figure 5. LTS Estimations.

MSE of each estimates are respectively 1.97, 8.61 and 1.98. The estimation are better than the LS estimator, and the algorithms are relatively fast.
At last, figure 5 shows the estimations for the FAST-LTS and the second LTS method. The MSE are respectively 76.7×10^{-3} and 62.6×10^{-3}. Here we can see that these estimations are the best. Furthermore, the second method is very fast because it is designed especially for this type of application, contrary to FAST-LTS which is the slowest method of our work.

Yet, a problem is remaining for these estimations: the spectra whose estimated background have been subtracted may have negative values, which is not acceptable for this kind of spectra.

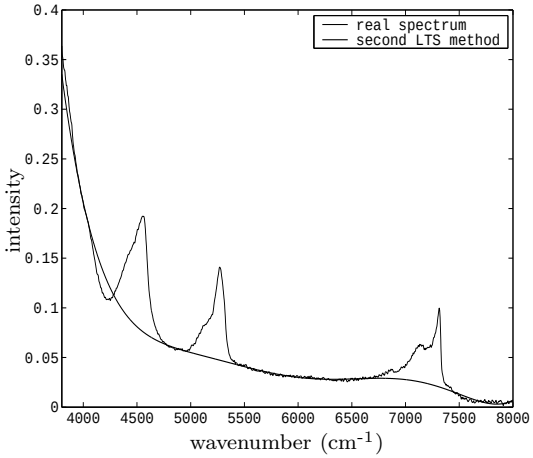

Figure 6. Amorphous Silica.

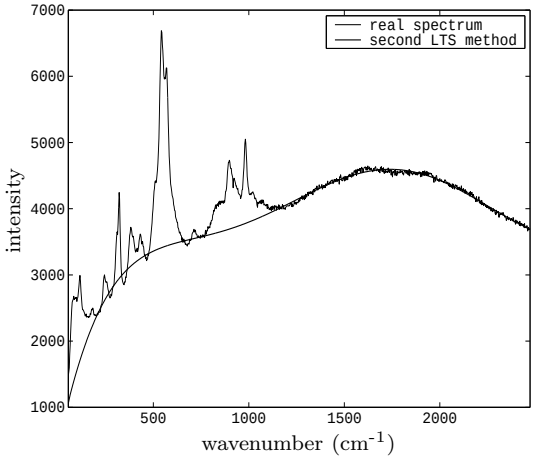

Figure 7. Gibbsite.

5.2 Application to Real Data

These methods have also been applied on a real data signals. The first one (figure 6) represents an infrared spectrum of amorphous silica, and the second one (figure 7) a Raman spectrum of gibbsite AlO(OH). Only the second LTS estimation is plotted, to make the figures more readable.

We see that the background is almost correctly estimated, except around 4200 cm^{-1} on figure 6. Nevertheless, the second LTS method remains the best among the others in this paper.

6. CONCLUSION

This article has presented several methods based on least squares estimation to estimate a spectrum background. Some of these methods give satisfactory results (*e.g.* the trimmed least squares), and may be simple and fast. Future works will be directed toward the respect of the spectra positivity,

and will be able to be coupled with a deconvolution method to obtain an ideal spectrum, *i.e.* a pike signal.

ACKNOWLEDGMENTS

This work was supported by the région Lorraine, France, and by the CNRS (*Centre National de la Recherche Scientifique*).

REFERENCES

Blake, A. and A. Zisserman (1987). *Visual Reconstruction*. MIT Press.

Charbonnier, P., L. Blanc-Féraud, G. Aubert and M. Barlaud (1997). Deterministic edge-preserving regularization in computed imaging. *IEEE Transaction on Image Processing*.

Dollinger, M.B. and R.G. Staudte (1991). Efficiency of reweighted least squares iterates. In: *Directions in Robust Statistics and Diagnostics*. Vol. 1. pp. 61–65. Springer-Verlag.

Geman, D. and G. Reynolds (1992). Constrained restoration and the recovery of discontinuites. *IEEE Transaction on Pattern Analysis and Machine Intelligence* **14**(3), 367–383.

Gulam Razul, S., W.J. Fitzgerald and C. Andrieu (2003). Bayesian model selection and parameter estimation of nuclear emission spectra using RJMCMC. *Nuclear Instruments and Methods in Physics Research A* **497**, 492–510.

Idier, J. (2001). Convex half-quadratic criteria and interacting auxiliary variables for image restoration. *IEEE Transaction on Image Processing*.

Kneen, M.A. and H.J. Annegarn (1996). Algorithm for fitting XRF, SEM and PIXE X-ray spectra backgrounds. *Nuclear Instruments and Methods in Physics Research B* **109/110**, 209–213.

Rousseeuw, P.J. and G.W. Bassett Jr. (1991). Robustness of the *p*-subset algorithm for regression with high breakdown point. In: *Directions in Robust Statistics and Diagnostics*. Vol. 2. pp. 185–194. Springer-Verlag.

Rousseeuw, P.J. and K. Van Driessen (1999). Computing lts regression for large data sets. Technical report. Technical report of the University of Antwerp.

von der Linden, W., V. Dose, J. Padayachee and V. Prozesky (1999). Signal and background separation. *Physical Review E*.