An Automatic Method for Extracting Chemical Impurity Profiles of Illicit Drugs from Chromatoraphic-Mass Spectrometric Data and Their Comparison Using Bayesian Reasoning

Tuomas Salonen
Department of Mathematics and Statistics
University of Helsinki

February 27, 2017

HELSINGIN YLIOPISTO — HELSINGFORS UNIVERSITET — UNIVERSITY OF HELSINKI

Tiedekunta/Osasto — Fakultet/Sektion — Faculty	Laitos — Institution — Department
Faculty of Science	Department of Mathematics and Statistics

Tekijä — Författare — Author
Tuomas Salonen

Työn nimi — Arbetets titel — Title
An Automatic Method for Extracting Chemical Impurity Profiles of Illicit Drugs from Chromatoraphic-Mass Spectrometric Data and Their Comparison Using Bayesian Reasoning

Oppiaine — Läroämne — Subject
Applied Mathematics

Työn laji — Arbetets art — Level	Aika — Datum — Month and year	Sivumäärä — Sidoantal — Number of pages
Master's thesis	February 2017	76 s.

Tiivistelmä — Referat — Abstract

In this work, an automated procedure for extracting chemical profiles of illicit drugs from chromatographic-mass spectrometric data is presented along with a method for comparison of the profiles using Bayesian inference. The described methods aim to ease the work of a forensic chemist who is tasked with comparing two samples of a drug, such as amphetamine, and delivering an answer to a question of the form "Are these two samples from the same source?" Additionally, more statistical rigour is introduced to the process of comparison.

The chemical profiles consist of the relative amounts of certain impurities present in seized drug samples. In order to obtain such profiles, the amounts of the target compounds must be recovered from chromatographic-mass spectrometric measurements, which amounts to searching the raw signals for peaks corresponding to the targets. The areas of these peaks must then be integrated and normalized by the sum of all target peak areas.

The automated impurity profile extraction presented in this thesis works by first filtering the data corresponding to a sample, which includes discarding irrelevant parts of the raw data, estimating and removing signal baseline using the asymmetrical reweighed penalized least squares (arPLS) algorithm, and smoothing the relevant signals using a Savitzky-Golay (SG) filter. The SG filter is also used to estimate signal derivatives. These derivatives are used in the next step to detect signal peaks from which parameters are estimated for an exponential-Gaussian hybrid peak model. The signal is reconstructed using the estimated model peaks and optimal parameters are found by fitting the reconstructed signal to the measurements via non-linear least squares methods. In the last step, impurity profiles are extracted by integrating the areas of the optimized models for target compound peaks. These areas are then normalized by their sum to obtain relative amounts of the substances.

In order to separate the peaks from noise, a model for noise dependency on signal level was fitted to replicate measurements of amphetamine quality control samples non-parametrically. This model was used to compute detection limits based on estimated baseline of the signals.

Finally, the classical Pearson correlation based comparison method for these impurity profiles was compared to two Bayesian methods, the Bayes factor (BF) and the predictive agreement (PA). The Bayesian methods used a probabilistic model assuming normally distributed values with normal-gamma prior distribution for the mean and precision parameters. These methods were compared using simulation tests and application to 90 samples of seized amphetamine.

Avainsanat — Nyckelord — Keywords
predictive agreement, Bayes factor, locally linear regression, chromatography, R software

Säilytyspaikka — Förvaringsställe — Where deposited
The digital repository HELDA

Muita tietoja — Övriga uppgifter — Additional information

Acknowledgement

This thesis is the condensed end result of almost two years of hard work which included encountering many dead ends, reading dozens of articles, writing thousands of lines of code and many hours of enlightening discussions. It would never have been finished without contributions from the people around me who supported me and made me believe I could do it.

Much of the work was done in co-operation with Sami Huhtala from the National Bureau of Investigation (NBI) of Finland. I would like to thank the NBI for financially supporting this thesis and allowing access to their data. I would especially like to thank Sami for all the inspiration, ideas and unwavering support even when time was scarce and for helping me understand some basics of analytical chemistry. Without him, this thesis would not be what it is today.

I further wish to thank my supervisor, prof. Jukka Corander, for having such strong faith in me from the very beginning, even when there was little evidence that faith would be rewarded, and for offering me such a brilliant opportunity for my Master's thesis. His insightful counsel was most valuable and allowed me to stay steady with my course all the way through to the end. This thesis might never have been finished had he not reminded me that not everything always has to be perfect. I also thank him for making me believe that hard work and dedication do pay off in the end.

Finally, I wish to thank my partner Paula Kyyrö for being there when I needed her, for reminding me of the important things in life and for always pushing me to finish this work. Whenever I wavered or hesitated, she was there to give me the encouragement I needed to get things done and reach the finish line.

Contents

Chapter 1

Introduction

In the forensic context there often arises a situation where a forensic chemist is asked if two samples of a drug come from the same source. For this purpose, the samples are usually analyzed using hyphenated methods, such as gas chromatography-mass spectrometry (GC-MS)[1][2] and from the raw data provided by these methods, *impurity profiles* are extracted. The answer to the question regarding the source of the samples is then reduced to comparing their corresponding profiles. These profiles consist of the amounts of certain residual compounds in the sample that result from imperfections in the manufacturing stage, believed to be unique enough to allow for separation of similar and dissimilar samples from each other. This is, in essence, a hypothesis comparison problem with the first hypothesis being "These samples are similar / come from the same source." and the second being "These samples are dissimilar / come from difference sources."

This comparison problem is currently solved by using similarity metrics, such as the Pearson correlation coefficient, combined with a simple threshold for judging if two samples are similar. These metrics are usually not based on any statistical reasoning, but rather rely simply on empirical experimentation such as in the case of amphetamine [3]. In this work, a more rigorous approach utilizing the Bayesian framework (see e.g. [4]) for the hypothesis comparison is introduced based on work done on similar data for oil samples[5]. The resulting Bayesian similarity measures are formulated and applied to both real and simulated chromatographic data of amphetamine samples and their properties are analyzed.

In addition, this work also covers the problem of actually obtaining the impurity profiles necessary for sample comparison in an automated way. While software and systems exist for analyzing data (such as the free OpenChrom software [6]), they are often not designed for the purpose of processing data in forensic context, where the samples are often dirty and contain many interfering components. Furthermore, in forensics it can be useful to obtain quick preliminary results that help guiding the investigative process of a

crime and, as such, there is interest in having a method that can produce results directly from raw data with minimal effort from the analyst, prompting the automatization of the extraction of relevant features from data. Work on such automatization has been done before in works such as [7][8], but these methods often assume overly optimistic settings where the noise can be easily controlled. This cannot, in general, be assumed for the kinds of samples often studied in the forensic context. Therefore, in this work a method is developed for automatically processing chromatographic-mass spectrometric raw data in order to extract the relevant features from which the sample impurity profiles can be comprised. To this end the methodology introduced in [9] is taken as the basis and expanded to develop an algorithm for handling the entire process from the raw data to the impurity profiles.

Finally, for the automated feature extraction to work, an important issue is the analysis of the noise present in chromatographic sample, which is often ignored in the development of such methods. A procedure is introduced applying non-parametric regression techniques such as kernel regression [10] to estimate the noise standard deviation as a function of signal level. This is motivated by the strong relationship between the two properties as noted in, for example, [11]. This is of great importance as no single detection limit for chromatographic peaks is valid for every region of a signal where the baseline may fluctuate radically.

This work is structured as follows. In chapter 2, detailed descriptions of the general procedure and methods used for noise analysis, the feature extraction algorithm and the impurity profile comparison are presented. As much generality is retained as possible, and it should be stressed, that the methods used here should in theory be applicable with minor modifications to any situation where data similar to that produced by GC-MS is handled.

In chapter 3, a case study on amphetamine data is conducted using two datasets consisting of quality control measurements and data from actual confiscated amphetamine samples provided by National Bureau of Investigation in Finland. Furthermore, simulated data is used to determine the performance of the different similarity measures developed in chapter 2. The process of simulation is discussed and described in detail.

The final chapter contains discussion of the results obtained and includes suggestions for future research. Possible improvements for the current method are also explored.

Implementation of the methods was carried out in R[12] and C++ as facilitated by the powerful `Rcpp` interface [13]. In cases where ready made functions were applied, appropriate citations are included.

Chapter 2

Methods for Noise Analysis, Automatic Impurity Profile Extraction and Impurity Profile Comparison

This chapter consists of three distinct parts. In the first section the matters related to noise analysis and constructing a non-parametric model for signal level and noise standard deviation dependence are discussed. In the second part, the algorithm for extracting impurity profiles is developed and in the last part the different methods for comparison are described and their theoretical backgrounds are presented.

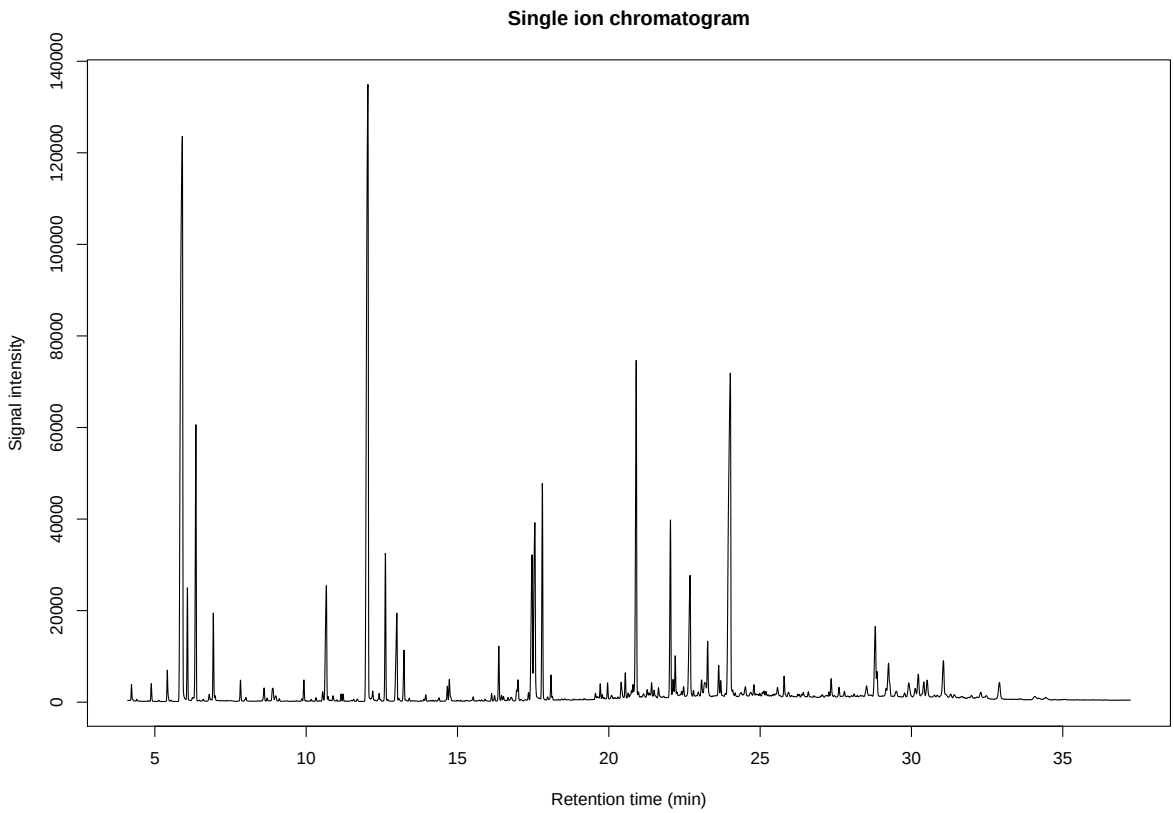


Figure 2.1: A chromatogram corresponding to signal from single ion channel.

2.1 Noise Analysis of Chromatographic-Mass Spectrometric Data

A chemical sample impurity profile consists of the integrated chromatographic peaks corresponding to the compounds relevant to the profile. In order to detect and identify these peaks and separate them from baseline noise it is essential to determine reasonable detection limits for each chromatogram. These detection limits necessarily depend on the estimated level of noise in the chromatogram and, as such, it is important to determine this noise level accurately.

Some classical methods for determining this level are to use a blank sample or regions of baseline from the chromatogram to compute a standard deviation for the noise. This value can then be multiplied by a constant, such as 3 or 10, to establish a limit of detection. Unfortunately, these approaches are not sufficient when the noise is heavily dependent on the signal level of a heavily varying baseline, which is often the case in chromatographic measurements (see [14] for detailed discussion on chromatographic noise). In addition, identifying regions of baseline from a non-blank chromatogram by any automated method is notoriously difficult. Thus, a more sophisticated noise model can be useful for inferring the relevant detection limits.

In this section, a general methodology is described which allows estimating the effect of signal intensity on the noise level by using repeated chromatographic measurements of single sample. Since this process is based on locally linear kernel regression, it is non-parametric and does not require prior specification of an explicit model for signal-noise dependence (for details on non-parametric regression, see e.g. [15]). The resulting model allows estimating the noise standard deviation given baseline intensity level.

In the first part, the necessary preprocessing of analytical measurements from samples is discussed. The second part discusses in detail the actual methods for obtaining a regression model for the data.

2.1.1 Preprocessing the Data from Repeated Measurements

In order to use repeated chromatographic measurements from, for example, quality control samples to estimate a model between signal mean and noise standard deviation, the raw data must first be preprocessed. In the case of chromatography-mass spectrometry, the raw analytical data consists of mass spectra measured against retention time. The mass spectra can be represented as vectors of intensity values with each component corresponding to a specific mass to charge ratio value, which uniquely characterize specific ions. Thus, the data from one sample forms a matrix where each row corresponds to a specific retention time or *scan* and each column corresponds to an *ion channel*.

More formally, suppose that a sample has been repeatedly measured $N_r \in \mathbb{N}$ times and the corresponding chromatographic runs are indexed by $n_r \in \{1, \ldots, N_r\}$. Then, the data corresponding to nth run is given by

$$\mathbf{X}_n = \left[\mathbf{x}^1_{n_r}, \ldots, \mathbf{x}^I_{n_r}\right],$$

where $I \in \mathbb{N}$ is the total number of ion channels considered and $\mathbf{x}^i_{n_r} = (x^i_{n_r,1}, \ldots, x^{(i)}_{n_r,T})^{\mathrm{T}}$ is a column vector corresponding to the intensity values of ion i measured at the retention times $t_1, \ldots, t_T \in \mathbb{R}^+$. In reality, the number $T \in \mathbb{N}$ of the total amount of scans can vary between chromatographic runs n and, as such, the data should be trimmed so that the retention times t_1 and t_T match between runs.

Assuming this trimming has already been done, the data can be rearranged to take the form

$$\mathbf{X}^i = \left[\mathbf{x}^i_1, \ldots, \mathbf{x}^i_{N_r}\right],$$

where $\mathbf{x}^i_{n_r}$ corresponds to a column vector consisting of the measurements of ion i in run n_r. Since each of these vectors are repeated measurements of the same sample, it can roughly be assumed that, for a given scan $t \in \{1 \ldots, T\}$ and ion i, the values $\{x^i_{1,t}, \ldots, x^i_{N,t}\}$ should only differ due to some level of random noise. Assuming no systematic bias, this noise can be considered to have zero mean and, as is common in chromatography, the amount of noise can be represented by its standard deviation.

However, chromatographic measurements are well known to be prone to shifts in retention time caused by the physical aspects of the measurement process. While signal alignment methods such as *parametric time warping* (PTW)[16] could be applied, they cannot be guaranteed to function in regions of pure baseline. As such, a simpler way to alleviate this issue is proposed here.

The total amount of scans T included in the analysis can be chosen so that it can be divided into windows of length $w \in N$, with w chosen so that $T = Pw$, $P \in N$. Then new P-dimensional vectors $\mathbf{z}^i_{n_r}$ of mean intensities are obtained by setting the pth component of the vector to the mean of the values in window $p \in \{1, \ldots, P\}$. That is, for window p, if $l \in \{1, \ldots, T\}$ is an index such that $l = (p-1)w$ and $\{x_{l+1}, \ldots, x_{l+w}\}$ are the values corresponding to the window, the pth component of $\mathbf{z}^i_{n_r}$ is given by

$$z^i_{n_r,p} = \bar{x}_{n_r,p} = \frac{1}{w} \sum_{t=l+1}^{l+w} x_t. \tag{2.1}$$

This leaves us with the reduced data set

$$\mathbf{X}^{i,win} = \left[\mathbf{z}^i_1, \ldots, \mathbf{z}^i_{N_r}\right],$$

where $\mathbf{z}^i_{n_r} = (\bar{x}_{n_r,1}, \ldots, \bar{x}_{n_r,P})^{\mathrm{T}}$.

From this reduced data set the mean and standard deviation for each row, corresponding to some window p, are computed, and these are taken as estimates of realized signal intensity levels and noise standard deviations. For each ion i, this results in two vectors,

$$\mathbf{m}_i = (m^i_1, \ldots, m^i_P)^{\mathrm{T}} \tag{2.2}$$

and

$$\mathbf{s}_i = (s^i_1, \ldots, s^i_P)^{\mathrm{T}}, \tag{2.3}$$

where for each window $p \in \{1, \ldots, P\}$

$$m^i_p = \frac{1}{N} \sum_{n_r=1}^{N_r} \bar{x}^i_{n_r,p} \tag{2.4}$$

and

$$s^i_p = \sqrt{\frac{1}{N-1} \sum_{n=1}^{N} (m^i_p - \bar{x}^i_{n,p})^2}. \tag{2.5}$$

This results in I vectors of signal means and standard deviations. The vectors corresponding to different ion channels can be concatenated if no difference in noise behaviour between channels is found. The resulting data set then consists of pairs of window means and standard deviations $(m_j, s_j) \in \mathbb{R}^+ \times \mathbb{R}^+$, where $j \in \{1, \ldots, J\}$ and $J = IP$.

As the retention time windows used to obtain these data points should be kept small, it is probable that the size of the resulting data set becomes fairly large, which may pose a computational difficulty. If this is the case, data binning can be used to reduce the number of data points as most of them are bound to come from low baseline regions and are therefore close to each other. Binning will introduce some estimation error, but with large amount of data and small bin width, the error should be negligible. [17]

For the purposes of this study, the binning was executed according to the simple binning rule given in [18] as follows. Suppose that the values m_j all lie between a and b with $0 < a < b < \infty$. Then a grid of $K \in \mathbb{N}$ points $g_k, k \in \{1, \ldots, K\}$ can be set in the interval $[a, b]$, with $g_1 = a$ and $g_K = b$. Then each observation (m_j, s_j) is assigned to the grid point it is closest to and for each grid point g_k the amount of points assigned to the grid point c_k and the total sum of values assigned to the point d_k are collected. These form new binned data points of the form (g_k, c_k, d_k). Thus, after binning only up to K data points remain and as the bins with zero counts can be discarded, this can result in massive savings in terms of dataset size.

2.1.2 Locally Linear Regression Using Gamma Kernels and Binned Data

Given a dataset of points $(m_j, s_j), j \in \{1, \ldots, J\}, J \in \mathbb{N}$, where the first component corresponds to signal mean and second component to corresponding standard deviation, the task of estimating the relationship between the quantities m and s can be interpreted a regression problem of the form

$$s_j = f(m_j) + \varepsilon_j,$$

where f is the unknown regression function describing the relationship and ε_j is random zero mean residual with variance that is allowed to depend on the value of m_j. Since $m_j \geq 0$ for all j, the support of f can be assumed to be $[0, \infty)$. In order to keep the method as general as possible, no prior information is claimed of the relationship described by f and, as such, non-parametric methods are used to obtain regression function estimate $\hat{f}$.

A commonly used non-parametric regression method is the Nadaraya-Watson (NW) kernel regression, which is essentially a weighted average smoother with weights provided by a continuous function known as kernel. However, it has been noted that the NW kernel regression suffers from considerable bias near boundaries of the regression function and from inability to model perfectly linear relationships[19]. Furthermore, the usual symmetric kernels themselves create bias near boundaries due to assigning weight outside the support of the regression function. The first problem can be solved by using locally linear regression, where for each point of estimation the weights are given by solving a local linear equation and the second problem is avoided by using asymmetric gamma kernels. A detailed description of the impressive theoretical properties of this method is given in [20].

Formally, the gamma kernel centered at $m \geq 0$ is defined for $x \geq 0$ as

$$K_{m,b}(x) = \frac{m^{x/b}}{\Gamma\left(x/b + 1\right) b^{x/b} e^{m/b}}, \tag{2.6}$$

where $b > 0$ is a smoothing parameter. The form of the gamma kernel in fact corresponds to a gamma density with parameters $x/b + 1$ and b with respect to the kernel center m and the shape of the kernel becomes more asymmetric as x approaches the boundary. This asymmetry is what allows the kernel to properly handle proximity to boundary. The gamma kernel further has the attractive property that the amount of smoothing is adaptive and depends on the value of x. In Figure 2.2 is illustrated a kernel with parameters $m = 7$ and $b = 1$. More illustrations are given in appendix A.

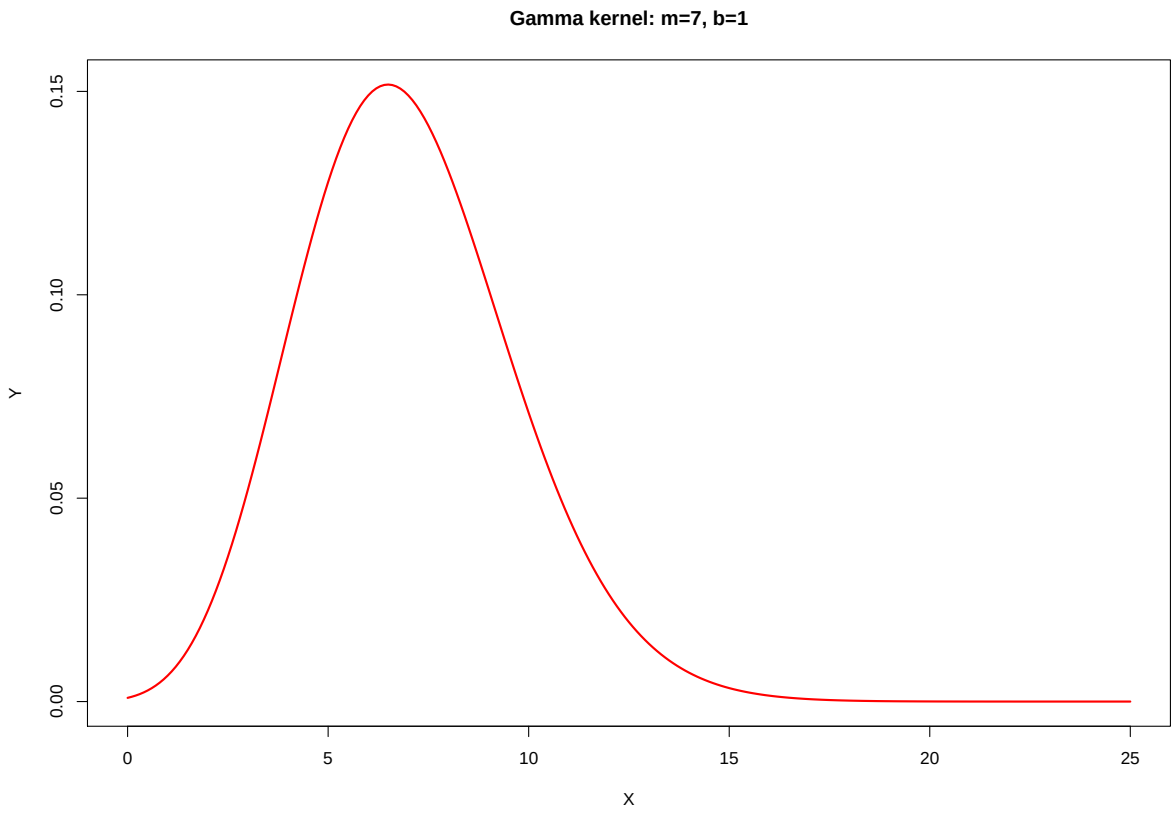


Figure 2.2: A gamma kernel with center $m = 7$ and smoothing parameter $b = 1$.

The locally linear regression estimate at x using this kernel is obtained by minimizing

$$\sum_{i=1}^{N}[s_k - v - \beta(x - m_k)]^2 K_{m_k,b}(x), \tag{2.7}$$

which corresponds to solving a weighted linear least squares problem. This essentially means locally fitting a line $l(x_0) = \hat{v} + \hat{\beta}(x - x_0)$, with x kept fixed and $\hat{v}, \hat{\beta}$ denoting the solution to the least squares equation. The problem is local in the sense that the kernel function assigns noticeable weights only to data points close to x. The regression estimate at x is therefore given by $\hat{f}(x) = l(x) = \hat{v}$ and, as such, we are only interested in the estimate $\hat{v}$ for the purpose of the regression. It should be noted that $\hat{\beta}$ could be used to estimate the derivative of f, but this is not necessary in the current work.

Finally, the estimate $\hat{f}(x) = \hat{v}$ is given by the formula

$$\hat{f}(x) = \frac{\sum_{j=1}^{J} L_j(x) s_j}{\sum_{j=1}^{J} L_j(x)}, \tag{2.8}$$

with local linear weights

$$L_j(x) = \frac{1}{J}\left[S_2(x) - S_1(x)(x - m_j)\right] K_{m_j,b}(x), \tag{2.9}$$

where

$$S_i(x) = \frac{1}{J}\sum_{j=1}^{J}(x - m_j)^i K_{m_j,b}(x), i = 1, 2. \tag{2.10}$$

However, in the current application, binned data of the form $(g_k, c_k, d_k), k \in \{1, \ldots, K\}$, where g_k is the grid point, c_k count of original data points assigned to the grid point and d_k is the total sum of s_j assigned to the grid point, are used. For such data, the corresponding binned estimate is given by

$$\hat{f}^{bin}(x) = \frac{\sum_{k=1}^{K} d_k L_k^{bin}(x)}{\sum_{k=1}^{K} c_k L_k^{bin}(x)}, \tag{2.11}$$

where the weights are now defined by

$$L_k^{bin}(x) = \frac{\left[S_2^{bin}(x) - S_1^{bin}(x)(x - g_k)\right] K_{g_k,b}(x)}{\sum_{k=1}^{K} c_k} \tag{2.12}$$

with

$$S_i^{bin}(x) = \frac{\sum_{k=1}^{K}(x - g_k)^i c_k K_{g_k,b}(x)}{\sum_{k=1}^{K} c_k}, i = 1, 2. \tag{2.13}$$

To make use of this estimate, it is necessary to specify of the smoothing parameter b. In the context of non-parametric regression methods, leave-one-out cross-validation can be used to minimize the mean squared cross-validation error[21]. The basic idea of this method is to leave one sample out from the fit at a time and then use the remaining data sets to estimate the value of the missing sample. In this case the error function is defined for estimator $\hat{f}$ and measured values s_i as

$$MSE(\hat{f}) \coloneqq \sum_{j=1}^{J} \left(\frac{s_j - \hat{f}(m_j)}{1 - L_j(m_j)} \right)^2 \tag{2.14}$$

and, for binned data, the equivalent binned mean square error (BMSE)[22] is defined as

$$BMSE(\hat{f}^{bin}) \coloneqq \sum_{k=1}^{K} \frac{D_k - 2\hat{f}^{bin}(g_k)d_k + \hat{f}^{bin}(g_k)^2 c_l}{(1 - L_k^{bin}(g_k))^2}, \tag{2.15}$$

where D_k are the sums of squared values s_i^2 corresponding to grid point g_k. The optimal smoothing parameter b is then the one that minimizes (2.14) for normal locally linear regression and (2.15) for the binned version.

2.2 Automatic Extraction of Impurity Profiles

While obtaining data through chromatography-mass spectrometry is routine work in most laboratories these days, extracting useful features from this data is not so straightforward. In the case of forensic drug sample comparison, the samples are characterized by impurity profiles consisting of the relative amounts of specific chemical compounds in the samples. These compounds are impurities in the sense that they are the byproducts of the manufacturing process of the substance and, as such, comprise only a small portion of the sample making detection difficult. The detection and quantification of these compounds by hand is a time consuming process and it is therefore desirable to automatize it as far as possible.

In this section, a general procedure for this automation is presented utilizing recent advances in the field of chemometric. In the first part, the developed algorithm is outlined as a whole. Then the sub-tasks of the algorithm, data filtering, peak detection, peak deconvolution and feature extraction are each examined separately.

2.2.1 Basic Concepts and Method Outline

The problem of feature extraction in the case of chromatography-mass spectrometry data can be formalized as follows. A measured sample $\mathbf{X}$ is a S by I matrix, with $S, I \in \mathbb{N}$

where each row corresponds to a single scan and each column corresponds to a measured ion channel. Explicitly,

$$\mathbf{X} = \begin{bmatrix} x_{1,1} & \dots & x_{1,i} & \dots & x_{1,I} \\ \vdots & \ddots & \vdots & & \vdots \\ x_{s,1} & \dots & x_{s,i} & \dots & x_{s,I} \\ \vdots & & \vdots & \ddots & \vdots \\ x_{S,1} & \dots & x_{S,i} & \dots & x_{S,I} \end{bmatrix}, \tag{2.16}$$

where $s \in \{1, \dots, S\}$ and $i \in \{1, \dots, I\}$. Each scan s is associated with a retention time r_s, which represents the time since beginning of measurement when the scan was made.

The goal is to extract from this raw data information related to $N_t \in \mathbb{N}$ *target compounds*. A compound $n_t \in \{1, \dots, N_t\}$ produces signal peaks in several ion channels depending on its mass spectrum around a retention time R_t. This retention time is not assumed to be exact, but rather a system dependent parameter with the actual compound elution occurring within some retention time window around it.

One of the ion channels, in which elution of the compound causes a peak, is assumed to have been chosen as the *target ion channel* based on such things as chemical stability and ability to characterize the amount of the compound in the sample. One or more of the remaining ion channels are assumed to be chosen as *qualifier ion channels* and are used to verify that a peak is indeed produced by the compound of interest.

This verification is based on the idea that signal peaks corresponding to the compound should occur simultaneously in the target and qualifier ion channels. Furthermore, the relative areas of these peaks should match the known mass spectrum of the compound. When a peak has been verified as being caused by the target compound, the area of the target ion peak is used to quantify the amount of the compound in the sample. Thus, in order to extract the impurity profile, these peaks must be identified from raw signal for each target compound and their areas calculated.

To this end the suggested method proceeds first by filtering the raw data in order to discard irrelevant information and improve signal to noise ratio. This is accomplished by extracting only a small retention time window centered at retention time R_t for each compound. Furthermore, for each compound, only the ion channels relevant to it are retained. For the remaining data, the baseline is corrected and, from baseline signal, noise levels are estimated and signal likely to be noise is set to zero. Finally, the signals for each ion channel are smoothed and the first and second derivatives of the signals are computed numerically.

In the peak detection step, these derivatives are used to identify all the peaks present in the signal for each ion channel. Parameters relevant to a peak model are estimated from the detected peaks and peaks occurring in different ion channels at same time are matched and assumed to come from a single compound.

The deconvolution step utilizes a mathematical peak model and parameters estimated in previous step to compute a non-linear least squares fit to the signal. In this step peaks matched as being produced by the same compound are assumed to share certain parameters to enhance robustness of the fitting. This step is important in order to solve issues caused by potentially overlapping peaks. The step is concluded by discarding compounds missing either the target or any qualifier peaks.

After deconvolution, for each retained compound, the peak model fitted for the target and qualifier ion signals is integrated, producing a vector of peak areas. The ratio of the qualifier peak areas to the target peak area is computed and compared to values derived from the known mass spectrum of the target compound. If the values match within predetermined tolerance, the peak is assumed to represent the target compound.

Each step is critical for the successful extraction of the impurity profile. However, any one step can be modified separately and adjusted to fit the needs of the current application.

2.2.2 Data Filtering

The point of the data filtering step is to extract only the information relevant to the target compounds and help separate noise from signal by improving signal to noise ratio. The implementation carried out in this study is described here in detail.

Let $\mathbf{X}, n_t$ and R_t be defined as in Section 2.2.1. The signal peaks corresponding to the compound n_t should appear within a system dependent retention time tolerance window near R_t with radius $r_{\text{tol}} > 0$, which is a parameter provided by the user. Furthermore, to avoid a situation where a peak is cut off from this window because it appeared right at the border, an external window margin $r_{\text{ext}} > 0$ should be added to r_{tol}. Thus, the total signal window radius to be considered is given by $w \coloneqq r_{\text{tol}} + r_{\text{ext}}$. Let $S, E \in \mathbb{N}$ be the scans corresponding to measurement with smallest retention time above $R_t - w$ and largest retention time below $R_t + w$ respectively. Then the signal relevant to this compound is given by

$$\mathbf{X}^t \coloneqq \begin{bmatrix} x_{S,i_1} & \dots & x_{S,i_q} & \dots & x_{S,i_Q} \\ \vdots & \ddots & \vdots & & \vdots \\ x_{s,i_1} & \dots & x_{s,i_q} & \dots & x_{s,i_Q} \\ \vdots & & \vdots & \ddots & \vdots \\ x_{E,i_1} & \dots & x_{E,i_q} & \dots & x_{R_e,i_Q} \end{bmatrix}, \tag{2.17}$$

where $S \leq s \leq E, q \in \{1, \dots, Q\}$ and $i_1, \dots, i_Q \in \mathbb{N}$ are all the ion channels associated with the compound with $Q \in N, Q > 1$. Note that the ion channels i_q need not be in any specific order.

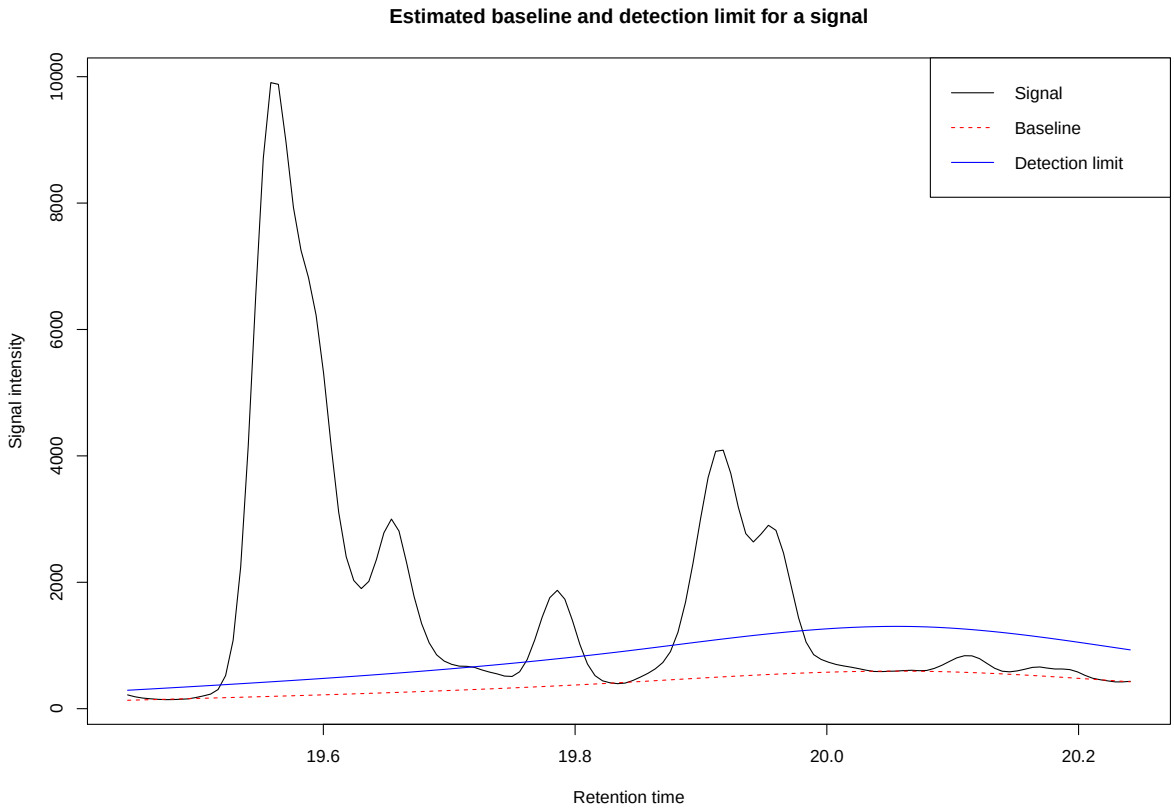


Figure 2.3: Example of a fitted baseline and the corresponding detection limit with $c = 3$.

For each ion channel i_q, the signal within the window is first baseline corrected. The baseline is estimated by the asymmetric re-weighted penalized least squares algorithm (arPLS) [23], which has been developed as an improvement to the popular asymmetric least squares (ALS) algorithm [16]. The advantages of arPLS include that it is able to deal with pure baseline areas, which can be encountered in this context and it is more flexible than ALS in the sense that it allows signal level to be below baseline due to noise. The estimated baseline $\mathbf{z}$ is subtracted from the signal, resulting in baseline correction.

Once the baseline has been computed, the noise S_s^t is estimated at the baseline for each point $s \in \{S, \ldots, E\}$. This can be accomplished using the model described in Section 2.1. The signal peak threshold, that is the minimum intensity that can be considered a peak, is defined at each point s as

$$\mathrm{thr}_{h,s}^t \coloneqq cS_s^t, \tag{2.18}$$

where $c > 0$ is a multiplier supplied by the user. Typical choices for c would be 3, 5 or 10.

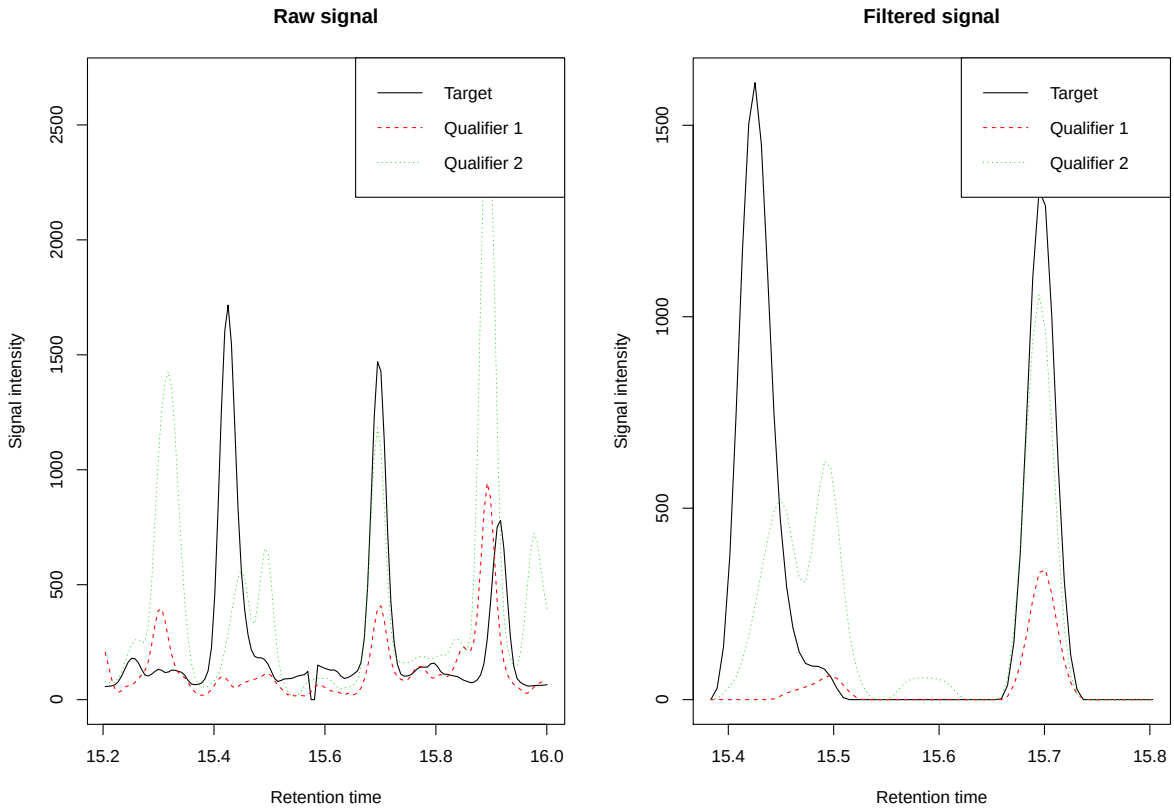


Figure 2.4: Side by side comparison of raw signal and completely filtered signal including the target ion channel and two qualifier ion channels.

By having the threshold depend on the index s, different parts of the signal are allowed to have different detection limits. Furthermore, for the raw signal, the threshold would be $\text{thr}_{h,s}^{t,q} + z_s$, where $z_s \in \mathbb{R}$ is the value of the baseline at s. Thus, the defined threshold corresponds to the classical detection limit definition in chromatography where limit is set as noise standard deviation at baseline level multiplied by a factor, resulting in a certain minimum signal to noise ratio.

The signal is further trimmed by setting all values below certain proportion, such as 10 %, of the threshold to zero in order to further reduce presence of baseline areas in signal. This results in a signal with some disjoint areas with positive values. Of these areas, any that do not contain at least a single value above the peak threshold are also set to zero.

The remaining signal is smoothed using the Savitzky-Golay polynomial filter[24], im-

plemented in R package `signal`[25], which is widely used in chromatography[14]. This requires specification of the polynomial degree and filter window width and in this work approach similar to [9] is adopted with the degree of the polynomial left as user defined parameter and the window width found by minimizing the *Durbin-Watson criterion* [26] within a range of possible window widths given by user. The DW criterion is given by

$$DW = \frac{\sum_{i=2}^{n}[(y_{\text{exp},i} - y_{\text{smd},i}) - (y_{\text{exp},i-1} - y_{\text{smd},i-1})]}{\sum_{i=1}^{n}(y_{\text{exp},i} - y_{\text{smd},i})^2} \times \frac{n}{n-1} \tag{2.19}$$

for experimental signal y_{exp} and smoothed signal y_{smd} with n measurements. The criterion measures how often consecutive differences between smoothed and original signals have the same sign. If they have the same sign, then this implies systematic difference which is undesirable as in that case the smoothing did not remove only noise. The optimal smoothing is achieved when $DW = 2$. As such, the best window width is found by finding the one that results in DW value closest to 2. In addition, the first and second derivatives of the signal are also computed by the Savitzky-Golay filtering algorithm. Appendix A contains illustrations of the effect of the window width on smoothing as well as examples of derivatives produced by the filter.

The noise of these derivatives is also computed similar to [9]: Let $d_j, j = 1, 2$ be the jth derivative of the signal and $L \coloneqq 2w + 1$ be the width of the signal window. Then define the residual

$$s_k^j \coloneqq \left| d_j(k) - \frac{d_j(k-1) + d_j(k+1)}{2} \right|, k \in \{2, \ldots, L-1\}, \tag{2.20}$$

where $d_j(k)$ is the value of the derivative at index k. Then, the noise of the derivative is given by

$$\varepsilon_j \coloneqq \text{median}(\mathbf{s}^j), , \tag{2.21}$$

where $\mathbf{s}_{q,j}$ is the vector containing the residuals. The detection thresholds for the derivatives are given by

$$\text{thr}_{\text{fd},q}^t = a\varepsilon_1 \tag{2.22}$$

for the first derivative and

$$\text{thr}_{\text{sd},q}^t = b\varepsilon_2 \tag{2.23}$$

for the second derivative. Here $a, b \in \mathbb{N}$ are user given multipliers.

The end result of these operations is a smoothed signal with greatly reduced noise (see Figure 2.4). However, as the window width was chosen so that it is wider than the actual retention time tolerance of the peak location, the window can be further trimmed by removing from its beginning and end sections that surely do not contain relevant signal. This can be beneficial in reducing computational complexity in later stages of

processing as it results in less non-relevant peaks to be fitted. In the current application, this trimming was accomplished by finding the scans t_1, t_2, with retention times r_{t_1}, r_{t_2} such that $R_t <= r_{t_1} <= R_n - r_{\text{tol}}$ and $R_S + r_{\text{tol}} <= r_{t_1} <= R_E$, that were respectively the largest and smallest scans, for which signal in all ion channels was zero after processing, and then reducing the window to contain only the scans in the range $[t_1, t_2]$.

2.2.3 Peak Detection

The peak search algorithm presented here is heavily based on the method presented in [9]: the peaks are assumed to conform to a specific parametric model, and the parameters are estimated from peaks identified using the numerical first and second derivatives of the signal. As suggested in the article, the third derivative of the signal is not used as it is assumed no replicate measurements of the signal are available. In the current method, however, instead of the polynomially exponentially modified Gaussian model (PMG)[27], a simpler exponential Gaussian hybrid (EGH)[28] model is used. EGH peak model is similar to exponentially modified Gaussian (EMG)[29] model but it is more numerically stable. The EGH model is given by

$$f_{\text{egh}}(t, H, t_R, \sigma, \tau) = \begin{cases} H \exp \frac{-(t-t_R)^2}{2\sigma^2 + \tau(t-t_R)}, & 2\sigma^2 + \tau(t-t_R) > 0 \\ 0, & 2\sigma^2 + \tau(t-t_R) \leq 0 \end{cases}, \tag{2.24}$$

where the parameters H, t_R and σ are positive numbers specifying the height, peak center location and peak width respectively, and τ is a real number specifying peak asymmetry. The parameters are very similar to those of the PMG model used in [9], so the method described in that article is easily adapted to the EGH model with minor modifications. As typically the retention time interval for scans is fixed, t_R can be taken to be the retention time when peak maximum is reached, but just as well the scan number can be used. The actual retention time is important only when the compound is identified in the peak area integration phase.

Given this peak model, the objective of the peak detection procedure is to find all the peaks occurring in the filtered signal $\mathbf{X}_f^t$ of a target compound t, obtained by the algorithm described in Section 2.2.2, and to estimate the values of the model parameters for each found peak. Thus, the end result is a collection of peak parameters for each peak. Furthermore, as the ultimate goal is to identify peaks corresponding to specific compounds, peaks occurring simultaneously in different ion channels are matched based on their estimated retention times and grouped together as single compound. Once the peaks have been detected and corresponding peak parameter estimates and their upper and lower limits have been obtained, peak deconvolution is performed by fitting the model peaks to the data through nonlinear least squares. This allows separation of overlapping peaks in order to improve accuracy of peak area calculation.

Suppose that the filtered data $\mathbf{X}_f^t$,the derivatives $\mathbf{D}_j^t, j = 1, 2$ and the thresholds $\text{thr}_h^{t,q}$, $\text{thr}_{fd}^{t,q}$ and $\text{thr}_{sd}^{t,q}$ have been obtained by applying the algorithm from Section 2.2.2 and let q be the ion channel in which peaks are being searched. For brevity, the superscripts indicating the ion channel and compound are dropped. The peak detection procedure works by first identifying peaks for each ion channel separately. When a peak occurs in signal this results in two regions in the first derivative: a positive region to the left of the peak center and a negative region to the right, with a zero crossing at the maximum value. Thus, a peak region is identified as the region of signal starting when the first derivative first crosses the threshold thr_{fd} and ending when, after crossing the zero to negative values, it reaches $-\text{thr}_{\text{fd}}$. This is illustrated in Figure 2.5. These regions are collected and analyzed separately with the basic assumption that each region contains at least one peak.

For each detected peak region, the corresponding part of the second derivative of the signal is then inspected. The second derivative is used to detect highly overlapping peaks, which may produce separate negative areas in the second derivative at corresponding peak centers, even when only a single maximum is seen in the corresponding region of the original signal and only one zero crossing is observed in the first derivative. Thus, the algorithm searches for all negative regions in the second derivative, which are then assumed to indicate separate peak components. A negative region is here defined as starting when the second derivative crosses the threshold $-\text{thr}_{n,q}^{\text{sd}}$ from above and ending when it crosses the threshold from below. Furthermore, the region should be included in a sequence of negative values of the second derivative with width of at least $w_{\text{sd}} > 1$. These conditions should guard against spikes caused by noise being identified as peaks. The value of the parameter w_{sd} can be set by the user and should depend on the resolution of the measurement process. In the current implementation, minimum width of 3 was found to provide satisfactory results. Furthermore, the signal maximum value within the negative region must exceed the corresponding threshold value of thr_h in order to be included in further analysis.

Once these negative regions have been identified, several characteristic values are extracted from the signal. Firstly, the beginning of the negative region t_1 is set as the scan preceding the second derivative crossing the threshold $-\text{thr}_{\text{sd}}$ from above. Similarly, the end of the region t_2 is taken as the first scan where the second derivative crosses the threshold from below (see Figure 2.6). Additionally, the retention times corresponding to t_1 and t_2 provide natural lower and upper bounds for the peak center t_R respectively.

Secondly, the peak center t_R, the measured peak height $h_1 \in \mathbb{R}^+$ at t_R, and the measured second derivative absolute value $h_2 \in \mathbb{R}^+$ at t_R are determined by one of two different methods depending on the value of the signal at t_1 and t_2. If values at t_1 and t_2 are less than the measured signal maximum at $t_{\max} \in [t_1, t_2]$, t_R is determined by fitting a quadratic polynomial using least squares to the measurements corresponding to $t_{\max}, t_{\text{left}}$

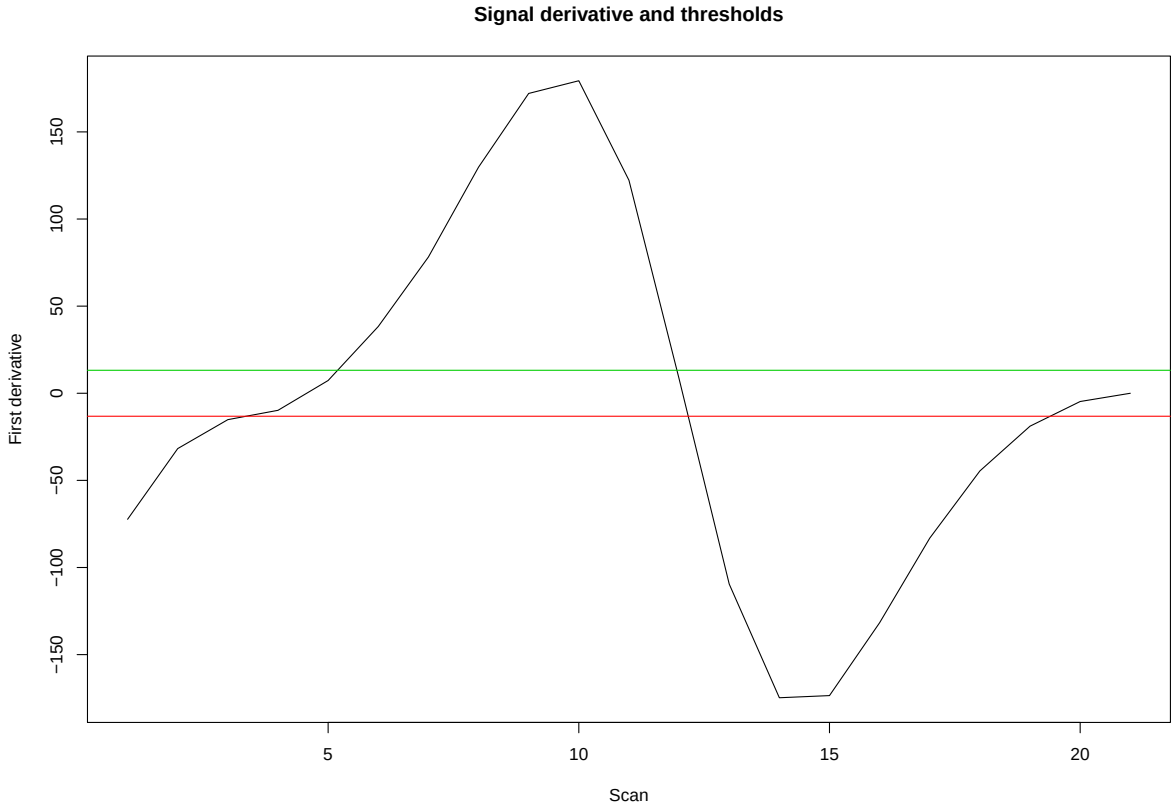


Figure 2.5: Illustration of the peak region detection. The green line is the threshold on the positive side and the peak region starts when the derivative first crosses this line from below. The red line is the threshold on the negative side and the peak region ends when derivative crosses this line from above.

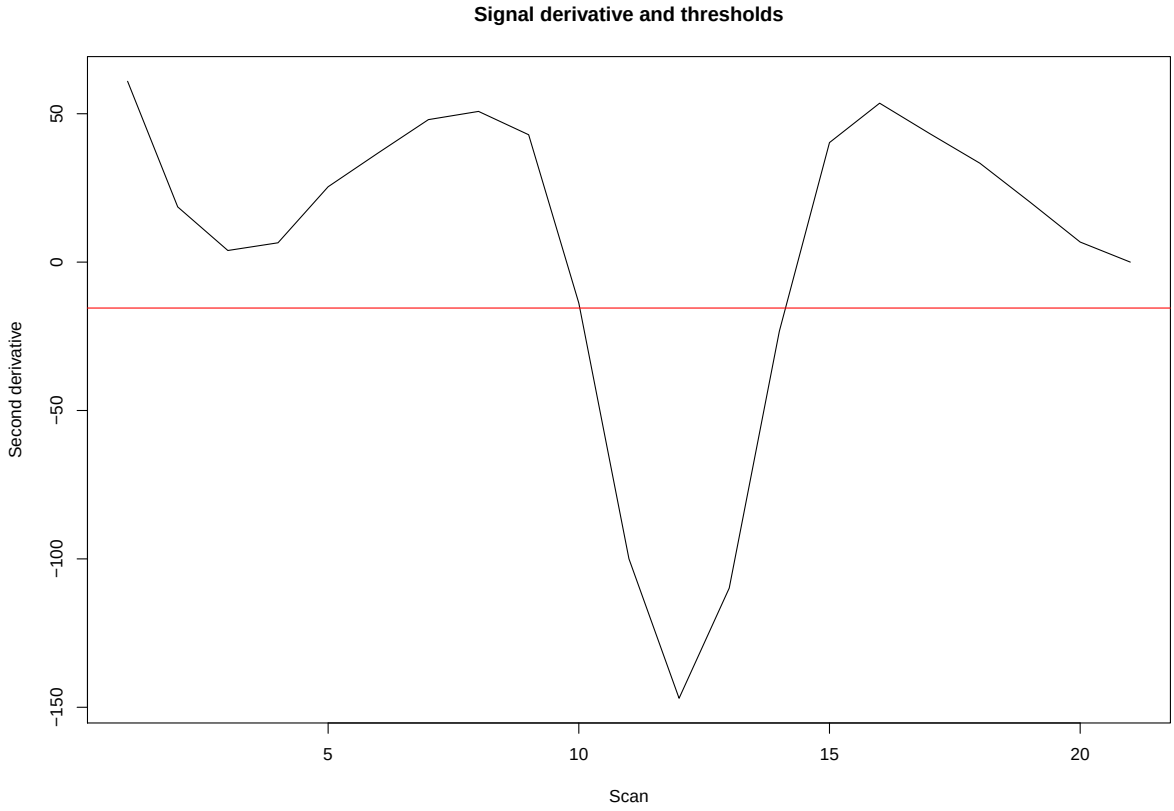


Figure 2.6: Illustration of the negative region detection. The red line is the negative threshold. Part of the second derivative below this signal is considered to be relevant.

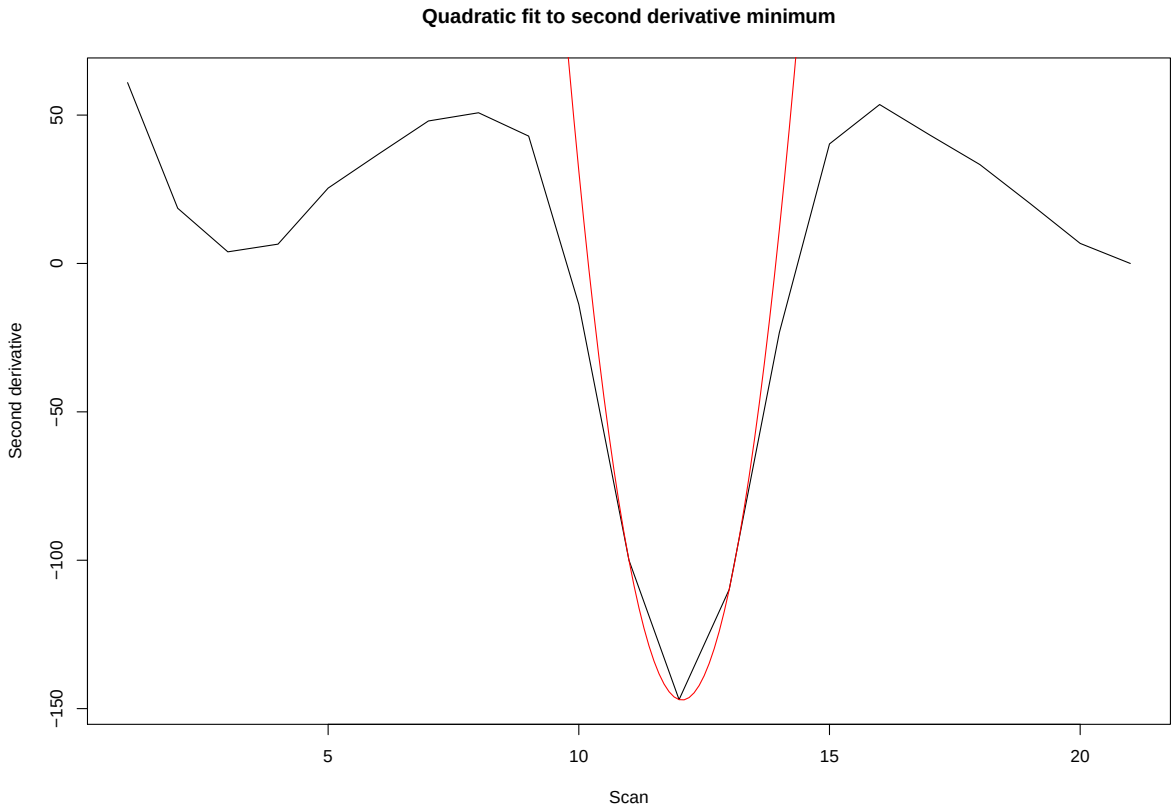


Figure 2.7: A quadratic fit around the minimum of the second derivative of a signal.

and t_{right}, with the last two retention times referring to the signal values to the left and right from the maximum. The peak center t_R is then set as the retention time where the polynomial reaches its maximal value, as it has been shown that this procedure can improve the accuracy of the estimation of the retention time of peak maximum[30]. The parameters h_1 and h_2 are simply set as the measured value of the signal and the absolute value of the second derivative at t_{max} respectively.

If instead the signal maximum in interval $[t_1, t_2]$ is found at either t_1 or t_2, implying a high level of overlap, the scan corresponding to the minimum of the second derivative $t_{\text{min}} \in [t_1, t_2]$ is used. The quadratic fit is then done using the values of the second derivative at this and the two adjacent points to the left and right of the minimum. Consequently, the retention time t_R is set as the location of the minimum of the fitted curve and h_1 and h_2 are set as the signal value and the second derivative absolute value at t_{min} respectively. An example of this fitting is given in Figure 2.7.

Thirdly, the peak height parameter H can be estimated by first defining its maximum and minimum values

$$H_{\max} \coloneqq \max\{h_1, \left(\frac{(t_2 - t_1)}{2}\right)^2 (h_2 + \text{thr}_{n,q}^{\text{sd}}\} \text{ and } H_{\min} \coloneqq min\{h_1, \left(\frac{(t_2 - t_1)}{2}\right)^2 (h_2 - \text{thr}_{n,q}^{\text{sd}}\}, \tag{2.25}$$

where the above expressions are based on a theoretical Gaussian peak as described in the article [9]. Given these bounds, H is simply estimated as their mean by taking

$$H \coloneqq \frac{H_{\max} + H_{\min}}{2}. \tag{2.26}$$

Furthermore, in order to estimate the asymmetry parameter τ, define the peak shape parameters

$$A_v \coloneqq |t_A - t_R| \text{ and } B_v \coloneqq |t_B - t_R|, \tag{2.27}$$

where t_A and t_B are the scans for which signal attains value vH to the left and right of the peak center t_R respectively. These values characterize the skewness of the detected peak (see Appendix A for illustrations), and from them the asymmetry parameter is computed by the formula suggested in [28] as

$$\tau = \frac{-(B_v - A_v)}{\ln(v)}. \tag{2.28}$$

This estimation is ignored if the measured maximum signal value in the interval $[t_1, t_2]$ is at the end points. Otherwise, the values of the signal at t_1 and t_2, denoted by V_{t_1} and V_{t_2} respectively, are determined with their ratios to the peak height defined by

$$v_1 = \frac{V_{t_1}}{H} \text{ and } v_2 \coloneqq \frac{V_{t_2}}{H}. \tag{2.29}$$

Then, to ensure that the peak is actually separable from any overlapping signal at the height corresponding to vH, the ratio is set as the maximum

$$v = \max\{v_1, v_2\}. \tag{2.30}$$

Finally, the retention times t_A and t_B are obtained as

$$t_A = \underset{t<t_R}{\arg\min} |x_{f,s} - vH| \tag{2.31}$$

and

$$t_B = \underset{s>t_R}{\arg\min} |x_{f,s} - vH|, \tag{2.32}$$

where $x_{f,s}$ are the values of $\mathbf{X_f}$ corresponding to the current ion channel q and scan s. Lastly, the parameter τ is obtained by applying equations (2.27) and (2.28).

The last parameter σ corresponding to peak width is computed in similar way as in [9], but without the error term depending on the third derivative, which was deemed too unstable for this application. The formula for the parameter is derived by the reasoning that for a Gaussian peak, its standard deviation is given by both $s_1 \coloneqq (t_2 - t_1)/2$ and $s_2 \coloneqq \sqrt{|h_1/h_2|}$, with $s_1 = s_2$. While it is not assumed any encountered peak is actually Gaussian, these values are nevertheless used to obtain a rudimentary estimate of the peak width as

$$\sigma = \frac{s_1 + s_2}{2} \tag{2.33}$$

with upper and lower bounds given by

$$\sigma_{\min} \coloneqq \min\{t_R - t_1, t_2 - t_R, s_2\} \tag{2.34}$$

and

$$\sigma_{\max} \coloneqq \max\{t_R - t_1, t_2 - t_R, s_2\} \tag{2.35}$$

respectively.

Thus, for each detected peak, a vector of parameters along with their bounds can be obtained. These vectors are then used to characterize the peaks as follows. Suppose $M_q \in \mathbb{N}$ peaks have been found in ion channel q. Then the m_qth peak $\mathbf{p}^{m_q}$, with $m_q \in \{1, \ldots, M_q\}$, is given by

$$\mathbf{p}^{m_q} = (t_R^{m_q}, t_1^{m_q}, t_2^{m_q}, H^{m_q}, H_{\min}^{m_q}, H_{\max}^{m_q}, \sigma^{m_q}, \sigma_{\min}^{m_q}, \sigma_{\max}^{m_q}, \tau^{m_q}). \tag{2.36}$$

Peaks characterized in this way require determining four parameters each. However, peaks corresponding to a single compound should share parameters as their retention time and asymmetry should be nearly identical due to the characteristics of the measurement process. To build on this idea, suppose that the target ion channel is given by $q = 1$. Then the compounds corresponding to the peaks $\mathbf{p}^{m_1}$ are given by

$$C^{n_c} = \{\mathbf{p}^{m_q}, q \in \{1, \ldots, Q\} \mid |t_R^{m_q} - t_R^{m_1}| < p_{\text{tol}}\}, \tag{2.37}$$

where p_{tol} is a tolerance parameter given by the user defining how close peak retention times must be to be considered to have appeared at the same time and n_c is the index of the compound, starting with 1 and increasing by one for each new compound. Only one peak from each ion channel can be matched to a compound, if two peaks in the same ion channel are close enough to the first peak in compound then the one that is closest is picked. Since there are likely peaks not corresponding to the ones found this process is repeated for each remaining "free" peak in the next ion channel. This is continued until all

peaks are assigned to a compound, even if that compound contains only the single peak, resulting in $N_c \in \mathbb{N}$ peaks. Then, for each compound C^{n_c} and for each peak $\mathbf{p}^q \in C^{n_c}$, the compound retention time is computed as

$$t_R^{n_c} = \frac{\sum_q t_R^q}{|C^{n_c}|}, \tag{2.38}$$

and the compound asymmetry parameter as

$$\tau^{n_c} = \frac{\sum_q \tau^q}{|C^{n_c}|}. \tag{2.39}$$

Here $|C^{n_c}|$ denotes the number of peaks in the compound and summation is calculated over all peaks in C_n. For the retention time, the compound upper bound and lower bound are set as the maximum and the minimum of the single peak retention times associated with the compounds respectively. These compound parameter estimates are then used to replace the corresponding parameters for each peak.

2.2.4 Peak Deconvolution

Suppose $N_c \in \mathbb{N}$ compounds C^{n_c} have been detected in the signal $\mathbf{X}$ corresponding to a target compound according to the method in Section 2.2.3. A reconstruction of the measured signal can then be computed by applying the peak model defined by (2.24) to parameters contained in the compounds as follows. Let t_s and t_e be the scans corresponding to the retention time window relevant to the target compound and, for each compound, let $\mathbf{z}^{n_c,q} = [z_{t_s}^{n_c,q}, \ldots, z_{t_e}^{n_c,q}]^{\mathrm{T}}$ be a vector with elements given by

$$z_t^{n_c,q} = f_{\text{egh}}\left(t, t_R^{n_c,q}, H^{n_c,q}, \sigma^{n_c,q}, \tau^{n_c,q}\right), t \in \{t_s, \ldots, t_e\}, \tag{2.40}$$

where the parameters $t_R^{n_c,q}, H^{n_c,q}, \sigma^{n_c,q}$ and $\tau^{n_c,q}$ correspond to a peak detected in ion channel q in compound C^{n_c}. In case no peak in ion channel q was associated with the compound, the elements of the vector $\mathbf{z}_{n_c}^q$ can be considered to be zero. These vectors then correspond to reconstructed signal within the relevant retention time window in the ion channel q caused by a peak. Together, these signals form the matrix

$$\mathbf{Z}^{n_c} = \left[\mathbf{z}^{n_c,1}, \ldots, \mathbf{z}^{n_c,Q}\right], \tag{2.41}$$

representing the contribution of the compound C^{n_c} to the total signal. The total signal matrix is then given by the sum of all compound signals

$$\mathbf{Z} = \sum_{n_c=1}^{N_c} \mathbf{Z}^{n_c} = \begin{bmatrix} z_{t_s}^1 & \cdots & z_{t_s}^q & \cdots & z_{t_s}^Q \\ \vdots & \ddots & \vdots & & \vdots \\ z_t^1 & \cdots & z_t^q & \cdots & z_t^Q \\ \vdots & & \vdots & \ddots & \vdots \\ z_{t_e}^1 & \cdots & z_{t_e}^q & \cdots & z_{t_e}^Q \end{bmatrix}. \tag{2.42}$$

As many peaks may be overlapping, it is highly probable the estimated parameters are not optimal. As such, it is of interest to try and tune the parameters so that $\mathbf{Z}$ corresponds to the measured signal $\mathbf{X}$ as closely as possible. Similarly to [27], this is treated here as an optimization problem. The objective function to be optimized is here considered to be the sum of squares across all ion channels given by

$$S^2 = \sum_{q=1}^{Q} \sum_{t=t_s}^{t_e} (x_t^q - z_t^q)^2. \tag{2.43}$$

Thus, the goal is to find the set of parameters that result in the reconstructed signal matrix $\mathbf{Z}$ minimizing S^2.

Due to the shape of the peak function, this is not a linear least squares problem and as such, non-linear iterative least squares methods must be used. In this work, a ready-made optimization tool utilizing the powerful Levenberg-Marquardt method [31] for solving non-linear least squares in R package `nloptr` [32] is used. Fit to the signal of a single compound obtained using this method is given in Figure 2.8. This method was chosen due to its speed and its ability to handle the constraints in many of the parameters.

The result of this procedure are the optimized peak parameters for detected compounds. Since a compound must have peaks in all ion channels to be considered a candidate for the target compound, only parameters corresponding to such compounds are retained. The rest can be discarded as the optimization process should have taken care of any overlapping in peaks. These remaining parameters are arranged in matrices

$$\theta_d \coloneqq \begin{bmatrix} t_{R,d} & \tau_d & H_d^1 & \sigma_d^1 \\ \vdots & \vdots & \vdots & \vdots \\ t_{R,d} & \tau_d & H_d^q & \sigma_d^q \\ \vdots & \vdots & \vdots & \vdots \\ t_{R,d} & \tau_d & H_d^Q & \sigma_d^Q \end{bmatrix}, \tag{2.44}$$

where $d \in \{1, \ldots, D\}$ denotes the index of the candidate compound and $D \in \mathbb{N}$ is the total number of candidates found. The final result for a single compound is thus a list of such

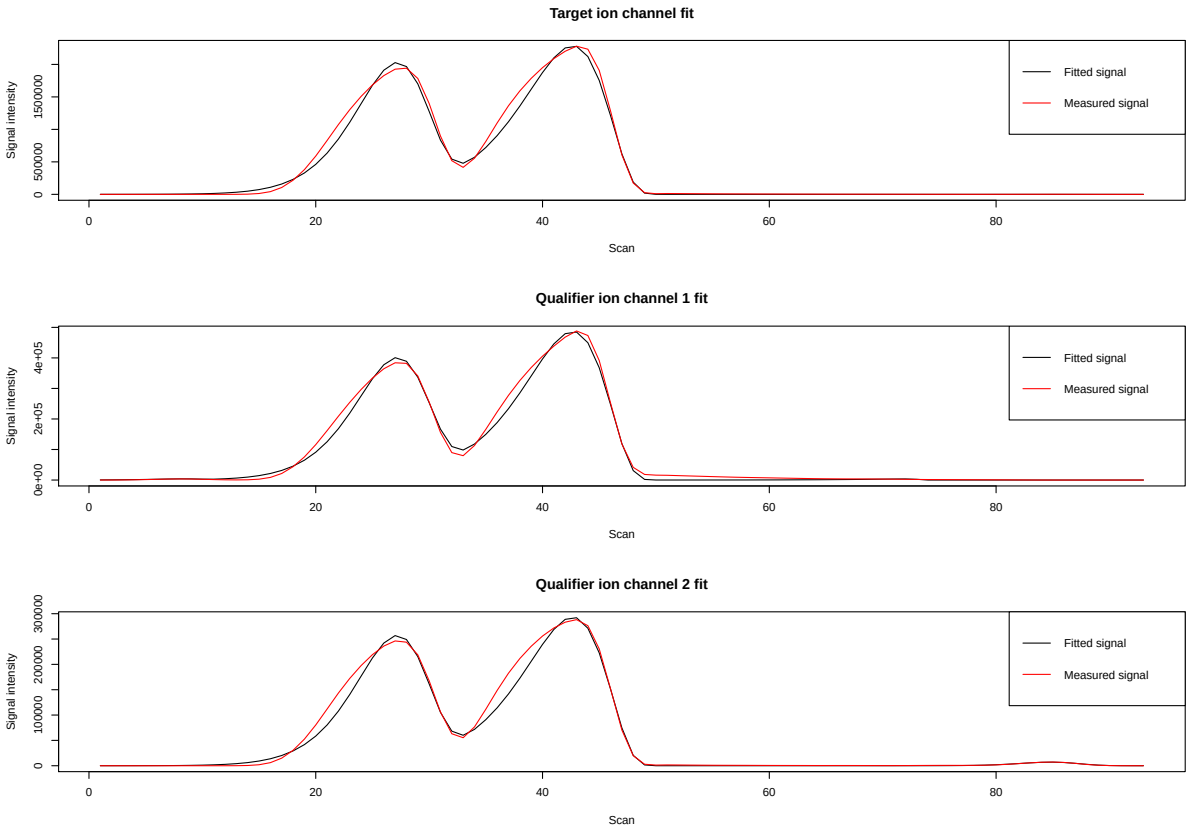


Figure 2.8: Fitted signal (black) and measured signal (red). Severe asymmetry in the measured peak causes some difficulties with peak fit due to rigidity of the EGH peak model.

matrices containing optimized parameters. Note that the peak center and asymmetry parameters are identical for each peak.

2.2.5 Feature Extraction by Peak Integration

The last step of obtaining features that comprise the impurity profile of a sample is the integration of the peak areas. In this step, the candidate compounds found in the deconvolution step are inspected and the areas of the target ion channel peaks are compared to those of the qualifier ion channel peaks. The ratio of the target peak area to the qualifier peak areas must match the known spectrum of the target compound closely enough for a compound to be considered to match the target.

For a given target compound, let

$$P = \{\theta_1, \ldots, \theta_D\} \tag{2.45}$$

be the set of parameter vectors corresponding to candidate compounds found in the signal, with vectors θ_d provided by the deconvolution step as given in (2.44). Then, for each θ_d the area corresponding to the peak in ion channel q is obtained by computing the integral

$$A_q \coloneqq \int_{t_s}^{t_e} f_{\text{egh}}(t, t_{R,d}, H_d^q, \sigma_d^q, \tau_d)\,\mathrm{dt}\,, \tag{2.46}$$

where t_s, t_e are the first and last scans of the retention time window considered for the compound. In practice this is done by numerical integration and in the current implementation the `integrate` function utilizing adaptive quadrature from R `base` package is used.

Suppose further that A_1 is the area of the target peak. The relative areas of the qualifiers with respect to the target are obtained by calculating for each $q > 2$ the ratio

$$v_q \coloneqq \frac{A_q}{A_1}. \tag{2.47}$$

Denote by v_q^T the ratio of the qualifier to the target expected based on the spectrum of the target compound. Then, a compound is accepted as a match for the target compound if

$$|v_q - v_q^T| < f_{\text{tol}}(v_q^T), \tag{2.48}$$

where f_{tol} is a function giving the tolerance corresponding to the ratio v_q^T. This function should be defined by the user and is technically arbitrary. However, it may be necessary to allow the tolerance to change with respect to the value of the ratio, as larger ratios may introduce higher absolute error. This is done in the current implementation by setting

$$f_{\text{tol}}(x) = \max 0.3, 0.3x, \tag{2.49}$$

to allow for higher tolerance for higher ratios, but to keep the tolerance at 0.3 in minimum. The expected ratio is often actually below 0.3, but in such cases the mere presence of the small qualifier peak should be an indicator for a matching compound. The exact values given here were hand tuned based on quality control data and they should be determined separately for each application as the tolerance is highly dependent on the nature of the data.

Furthermore, an additional requirement for a compound to be accepted is that the peak center retention time should be within a user defined tolerance from the expected retention time of the compound. While technically only a retention time window of interest is considered for analysis in the filtering step, the external tolerances for the retention time window make it a possibility that compounds outside this range are detected.

Finally, it is possible that more than one match is found among the detected compounds. In such a case a choice needs to be made and for the current application, the detected compound with largest target ion channel peak area is chosen to represent the target compound. Such a situation should, however, be very uncommon, although further investigation into such occurrences and how to deal with them might be warranted.

Let A^n then be the target peak area for the target compound $n \in \{1, \ldots, N\}$, where $N \in \mathbb{N}$ is the total number of targets. The end result of this step is to obtain for each sample a vector of areas

$$\mathbf{A} = [A^1, \ldots, A^N]^{\mathrm{T}}. \tag{2.50}$$

These vectors are called impurity profiles and can be used for sample comparison as described in the next section.

2.3 Comparison of Chemical Impurity Profiles

This section describes the methods used for preprocessing and comparing chemical samples through their impurity profiles obtained by the process defined in the previous section. These methods are largely based on the harmonized approach, which was developed for amphetamine in [3] and applied to cocaine samples in [33], that utilizes the Pearson correlation coefficient [34] for comparison. However, Bayesian methods similar to ones described in [5] are also considered. To this end, an explicit probabilistic model is introduced and applied to the current data and the data preprocessing steps necessary for applying this model are also described.

First, the preprocessing steps for the data are explored and discussed. Followed by description of the different methods used in this work to compare the samples. For comparison, similarity measures are produced along with equivalent dissimilarity measures.

2.3.1 Preprocessing Area Data

In order to use the area data provided by feature extraction process in the previous section for comparison, some preprocessing of the data is necessary. This preprocessing includes data normalization, in order to make features comparable between samples, and data weighing to ensure each feature will have sufficient impact on the result. More formally, suppose that for each sample indexed by $n_s \in \{1, \ldots, N_s\}, N_s \in \mathbb{N}$, a vector of target peak areas $\mathbf{A}^{n_s}$ is obtained through the integration step described in the previous section. As the samples have different concentrations, the total summed area of all components of $\mathbf{A}^{n_s}$ varies between samples. As such, the values themselves are not necessarily comparable as is. In this application, the standard method of treating these vectors established in [3] is used to obtain the final impurity profiles.

Firstly, any value in these vectors below 1 % of the area of the internal standard is considered to be not detected in order to remove very small peaks from comparison. The area of the internal standard peak is obtained by applying the previous steps to the corresponding target and qualifier ions. As is established in [3], these values are set to an arbitrarily small, positive number, such as 200 in the current implementation. Additionally, some areas may be combined depending on the chemical properties of the corresponding compounds, in which case the total number of variables is reduced slightly. To make the features comparable across samples, the impurity profiles are normalized by their sums, resulting in vectors of proportions of each impurity in the samples given by

$$\mathbf{v}^{n_s} = \frac{\mathbf{A}^{n_s}}{\sum_{n_f=1}^{N_f} A_{n_f}^{n_s}}, \tag{2.51}$$

where $n_f \in \{1, \ldots, N_f\}$ is the index of the impurity and N_f is the total number of features.

While these proportions are technically comparable, the fact that some compounds tend to have much higher concentrations in the sample than others causes the unfortunate effect that the proportions of less concentrated compounds are reduced very close to zero. In order to reduce this effect, *weighing* is introduced as suggested by [3] using the 4th root resulting in the weighed vectors of proportions defined as

$$\mathbf{v}_W^{n_s} = [(v_1^{n_s})^0.25, \ldots, (v_{N_f}^{n_s})^0.25]^{\mathrm{T}}. \tag{2.52}$$

It should be noted that taking the 4th root is in some sense *ad hoc* and other transformations might be considered. However, this has been established as the standard way to treat these profiles and as such it is used here.

The weighed proportion vectors $\mathbf{v}_W^{n_s}$ can then be used for the comparison by, for example, the Pearson correlation coefficient, which is the current standard method in amphetamine profile comparison. However, in this work more flexible probabilistic modeling

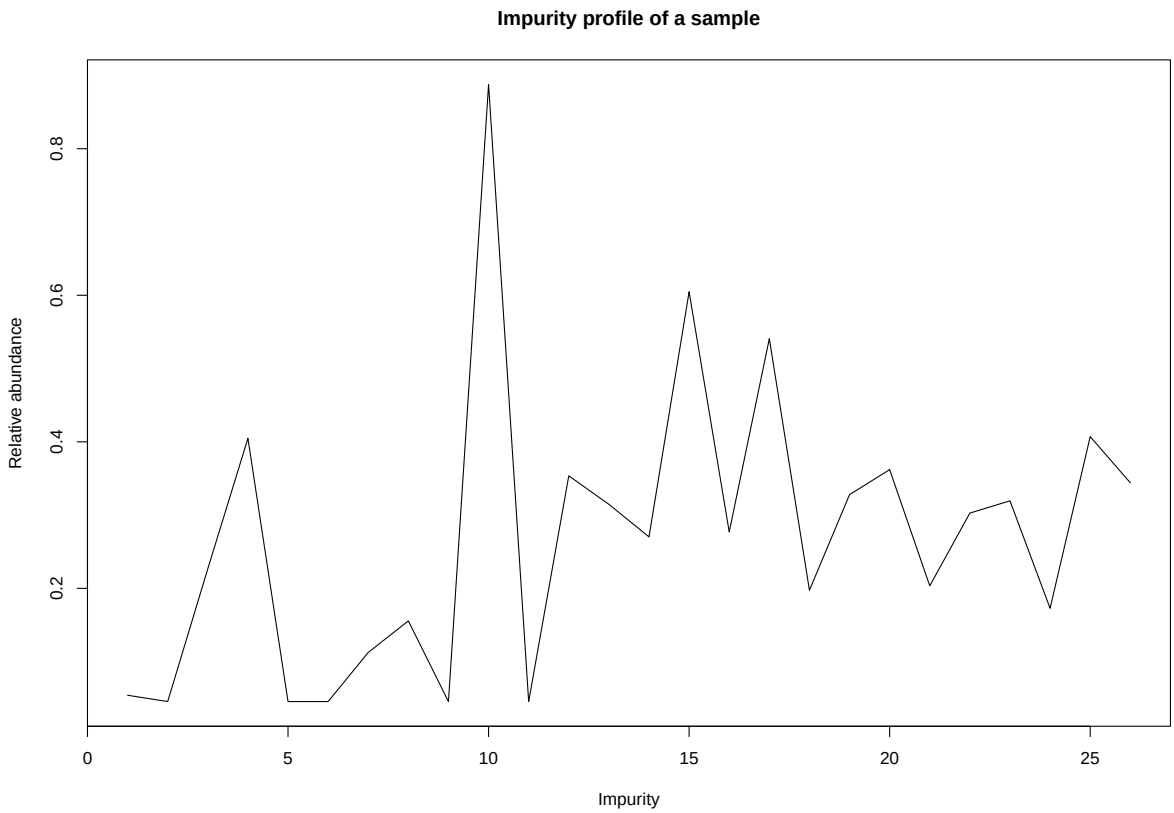


Figure 2.9: An example of an impurity profile after normalization and applying 4th root.

is considered and, to that end, the features are further transformed from the $(0, 1)$ interval to entire real line. This transformation is achieved using the *logit transformation* given by

$$\text{logit}(x) = \log\left(\frac{x}{1-x}\right), x \in (0, 1). \tag{2.53}$$

These transformed features are then denoted as

$$\mathbf{v}_L^{n_s} \coloneqq [\text{logit}(v_{W,1}^{n_s}), \ldots, \text{logit}(v_{W,N_f}^{n_s}]^{\mathrm{T}}. \tag{2.54}$$

In order to make probabilistic modeling more straightforward, each logit transformed feature is centered around zero by subtracting its mean. That is, given the feature corresponding to N_f, define its mean by

$$m_{N_f}^{v_L} = \frac{\sum_{n_s=1}^{N_S} v_{L,N_f}^{n_s}}{N_s} \tag{2.55}$$

and the vector of the means of all features by

$$\mathbf{m}^{v_L} = [m_1^{v_L}, \ldots, m_{N_f}^{v_L}]^{\mathrm{T}}. \tag{2.56}$$

Then the zero centered logistically transformed feature vectors are given by

$$\mathbf{v}_{L_0}^{n_s}) = \mathbf{v}_L^{n_s} - \mathbf{m}^{v_L}. \tag{2.57}$$

The weighed impurity profiles given by the vectors $\mathbf{v}_W^{n_s}$ and the logistically transformed zero centered profiles given by $\mathbf{v}_{L_0}^{n_s}$ can then be used for the comparison by Pearson correlation or the Bayesian methods presented in the following section. As a side note, while outside the scope of the current study, many other transformations and weighing schemes could be considered such as standardizing the features or taking the logarithm instead of the 4th root.

2.3.2 Comparison by Pearson Correlation Coefficient

The current standard measure for chemical impurity profiles in the case of amphetamine is the Pearson correlation coefficient. During a development project for a harmonized method for amphetamine comparison, this measure was found[3] to have the best performance out of several measures considered [33]. However, no statistical framework was designed around this method and as such, the choice is based purely on empirical experimentation. Regardless, this method has been found to provide satisfactory performance and is included here for comparison.

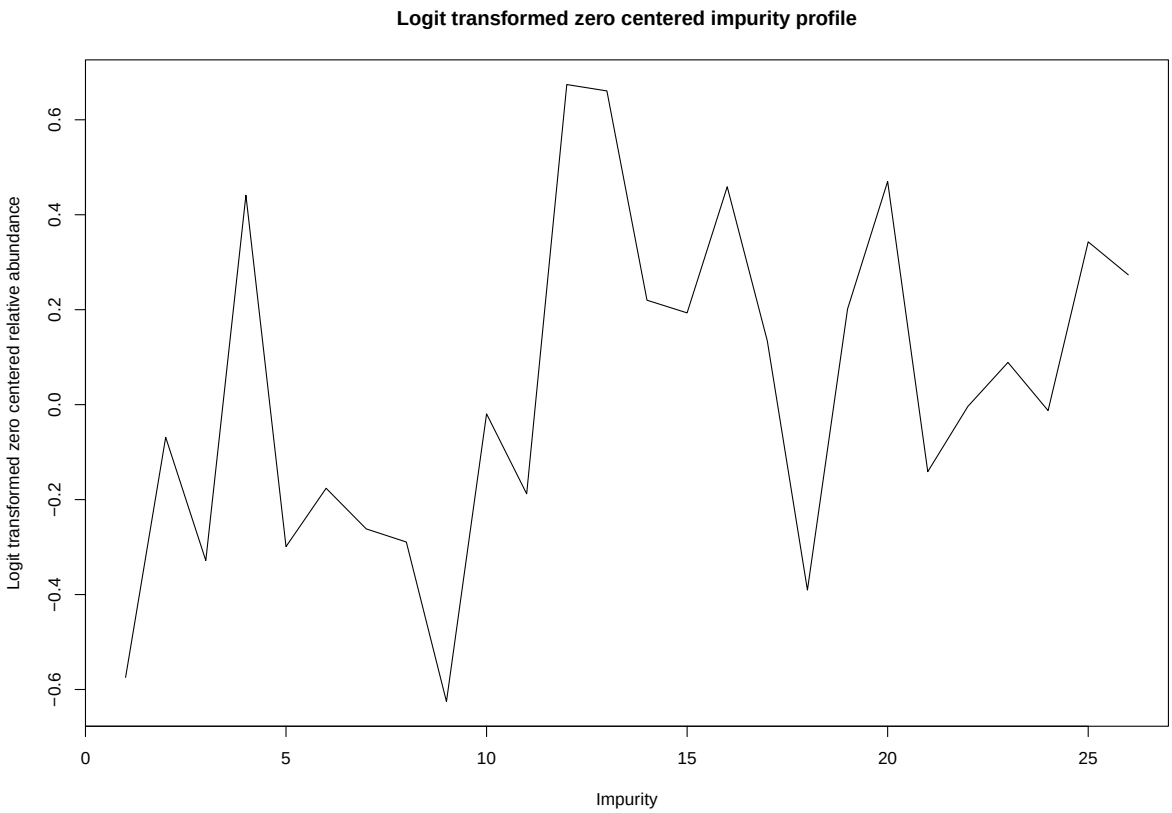


Figure 2.10: An example of an impurity profile after logit transform and zero centering.

Given two random variables X and Y, the Pearson correlation coefficient [34] between them is defined as

$$\rho_{XY} \coloneqq \frac{\operatorname{cov}(X,Y)}{\sigma_X \sigma_Y}, \tag{2.58}$$

where $\operatorname{cov}(X,Y)$ is the covariance between the variables defined by the formula

$$\operatorname{cov}(X,Y) = \mathbb{E}[(X-\mu_X)(Y-\mu_Y)] \tag{2.59}$$

and σ_X, σ_Y are the standard deviations of the variables given by

$$\sigma_X = \sqrt{\mathbb{E}[(X-\mu_X)^2]} \tag{2.60}$$

and μ_X, μ_Y are the means of the variables given by

$$\mu_X = \mathbb{E}(X). \tag{2.61}$$

The values of ρ_{XY} range from -1 to 1 and they measure various levels of linear correlation between the variables. Value of 1 means that an increase in X always coincides with an equal increase in Y relative to the variable means and conversely -1 indicates a decrease in X coincides with an equal increase in Y. As only samples x_i, x_i, with $i \in 1, \ldots, n, n \in \mathbb{N}$, from the variables are available, in practice the sample correlation coefficient

$$r_{\mathbf{x},\mathbf{y}} \coloneqq \frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i-\bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i-\bar{y})^2}} \tag{2.62}$$

is used, with $\bar{x}, \bar{y}$ denoting the sample averages and $\mathbf{x}, \mathbf{y}$ denoting the vectors of samples.

In the case of the impurity profiles, the values x_i and y_i correspond to the features of the profiles $\mathbf{x}$ and $\mathbf{y}$. Thus high correlation means that for the two impurity profiles, the same features are above the mean and below the mean in similar proportions. However, it should be stressed that the measurement does not take into account the absolute differences in the features and as such it discards information to some extent.

The final correlation score, adjusted so that its values lie in the range $[0,1]$ is obtained as

$$\operatorname{COR}(\mathbf{x},\mathbf{y}) \coloneqq \frac{1+r_{\mathbf{x},\mathbf{y}}}{2}. \tag{2.63}$$

When dissimilarities are required, the correlation distance can be obtained by simply computing

$$\operatorname{COR}_D(\mathbf{x},\mathbf{y}) \coloneqq 1 - \operatorname{COR}(\mathbf{x},\mathbf{y}). \tag{2.64}$$

2.3.3 Comparison by Bayesian Methods

The use of Bayesian reasoning in comparison of the impurity profiles is attractive because it allows for more rigorous statistical and mathematical basis for comparison. To this end, a probabilistic model is presented here to facilitate the comparison through statistical methods. Before defining the model, it should be noted that for two impurity profiles to be similar, the features should match closely enough. What is enough depends on the uncertainty of the obtained values of the features.

Model Specification

As in [5], one possibility is to use replicate measurements to estimate distributions of the features. However, in the current application, a problem arises that usually no replicate measurements are available for each sample. Instead, quality control samples with several replicate measurements are available. In this work, it is proposed that these replicate measurements can be used to estimate the overall uncertainty of the obtained feature, the idea being that the measurement uncertainty should not depend so much on the sample itself, but rather on the equipment and methodology used to obtain the features.

To facilitate this, it is assumed that, given a logistically transformed and zero centered impurity profile or feature vector $\mathbf{v}$, each feature v_i follows a normal distribution with the density function

$$f_i(v \mid \mu, \tau) = \sqrt{\frac{\tau}{2\pi}} \exp\left\{\frac{-\tau(v-\mu)^2}{2}\right\} \tag{2.65}$$

with the mean μ and precision τ. Since these parameters are both unknown, a normal-gamma prior distribution defined by

$$\pi_i(\mu, \tau \mid \mu_0, \kappa, \alpha, \beta) = \frac{\beta^\alpha \sqrt{\kappa}}{\Gamma(\alpha)\sqrt{2\pi}} \tau^{\alpha-0.5} \exp\{-\beta\tau\} \exp\left\{\frac{-\kappa\tau(\mu-\mu_0)^2}{2}\right\} \tag{2.66}$$

is assumed with hyperparameters μ_0, κ, α and β. This density corresponds to assuming a $\mathcal{N}(\mu_0, \kappa\tau)$ distribution for μ_0 conditional on τ and Gamma(α, β) distribution for τ. Care should be taken not to mistake the hyperparameter μ_0 for the mean μ of the measurement model. The parameters may vary depending on the feature, but subscripts are dropped to improve readability. This prior distribution is chosen because it is conjugate to the normal model with unknown mean and precision, which allows for easy computation of the marginal likelihoods and posterior predictive distributions needed for comparison. For details on the derivation of the marginal likelihood and posterior predictive distribution as well as the conjugacy property of the normal-gamma prior, see e.g. [35].

The selection of the hyperparameters $\mu_0, \kappa, \alpha, \beta$ for each feature is an important issue and, while specifying a universal method for choosing them is beyond the scope of this

work, one approach that is taken in this work, and discussed more in the next chapter, is to use replicate measurements of a sample to estimate how high precision can be expected of the measurement equipment.

The hyperparameter μ_0 is the prior estimate of the feature mean and, when zero centered features are used, it can be set to 0. The parameter κ can be thought of as how certain we are of our prior mean estimate μ_0, with high values indicating confidence that features do not stray far from μ_0. Since in this context little knowledge is available *a priori* of the features given the sample, this parameter can be set to be very small. Thus, the core of the model used here comes down to the parameters α and β, which specify the distribution of the precision of the obtained features. The parameters reflect how accurate the observed values are believed to be with respect to the true value and, additionally, how certain this estimate of accuracy is. These parameters can be easiest understood through the mean and variance of the parameter τ as they are given by

$$\mathbb{E}(\tau) = \frac{\alpha}{\beta} \text{ and } \mathrm{Var}(\tau) = E((\tau - \mathbb{E}(\tau))^2) = \frac{\alpha}{\beta^2} \tag{2.67}$$

based on the properties of the gamma distribution (see e.g. [36]). These quantities mean that, whenever α increases, the expected value of the precision τ increases implying higher expected accuracy of observed feature. If β is increased, the expected value is decreased, but the variance decreases as well and much more quickly, corresponding to certainty in the precision. Thus, this model should at least in principle be able to capture the characteristics of the measurement uncertainty with respect to the features.

Bayesian Similarity and Dissimilarity Measures for Sample Comparison

The two Bayesian methods for sample comparison considered here, the Bayes factor [37] and the predictive agreement [38], require knowledge of the marginal likelihood and the posterior predictive distribution of the statistical model, with the former specifying the total probability of the data under a model and latter specifying the distribution of a new observation. In order to state the corresponding formulae, the posterior update rules for the hyperparameters of the model given observations are first defined. To this end, suppose n measurements of feature v, denoted by vector $\mathbf{v} = [v_1, \ldots, v_n]^{\mathrm{T}}$, have been made with mean $\bar{v}$. Then the update rules of the model hyperparameters given these

observations are given by the equations

$$\begin{aligned}\mu_n =& \frac{\kappa\mu_0 + n\bar{v}}{\kappa + n}\\ \kappa_n =& \kappa + n\\ \alpha_n =& \alpha + \frac{n}{2}\\ \beta_n =& \beta + \frac{1}{2}\sum_{k=1}^{n}(v_k - \bar{v})^2 + \frac{\kappa n(\bar{v} - \mu_0)^2}{2(\kappa + n)}.\end{aligned} \tag{2.68}$$

Given the posterior parameters, the marginal likelihood can be shown to be

$$p_{\text{marg}}(\mathbf{v}) = \int_{\Theta}\prod_{k=1}^{n} f_i(v_k \mid \mu, \tau)\pi_i(\mu, \tau)\mathrm{d}\mu\mathrm{d}\tau = \frac{\Gamma(\alpha)}{\Gamma(\alpha_n)}\frac{\beta^{\alpha}}{\beta_n^{\alpha_n}}\sqrt{\frac{\kappa}{\kappa_n}}(2\pi)^{-0.5n}, \tag{2.69}$$

with Θ denoting the parameter space, and the posterior predictive distribution for a single new measurement v^* can be expressed as

$$p_{\text{pred}}(v^* \mid \mathbf{v}) = \frac{1}{\sqrt{\pi}}\frac{\Gamma(\alpha_n + 0.5)}{\alpha_n}\sqrt{\frac{\kappa_n}{2\beta_n(\kappa_n + 1)}}\left(1 + \frac{\kappa_n(v^* - \mu_n)^2}{2\beta_n(\kappa_n + 1)}\right)^{-(\alpha_n + 0.5)}. \tag{2.70}$$

These explicit formulae allow for quick computation of the Bayesian comparison methods presented in the following.

The problem of comparing two samples can be expressed as determining which of two hypotheses is more probable. In this case, these hypotheses can be framed as "the two samples have the same underlying distribution", denoted by H_0, and "the two samples have different distributions", denoted by H_1. The Bayesian framework is a natural choice for this (see [4]) and the Bayes factor given by

$$BF(\mathbf{x}) = \frac{p(\mathbf{x}, \mathbf{y} \mid H_0)}{p(\mathbf{x}, \mathbf{y} \mid H_1)}, \tag{2.71}$$

where $\mathbf{x}, \mathbf{y}$ are the observations so far of two samples and $p(\mathbf{x}, \mathbf{y} \mid H_0), p(\mathbf{x}, \mathbf{y} \mid H_1)$ denote the data marginal likelihood under the different hypotheses, is an often used method for such comparisons praised for being objective in the sense that it ignores some of the influence of prior distributions[39]. However, for the comparison by Bayes factors to be reliable, sufficient data should be available, which is often not the case. The absence of data may cause such model evaluation to overestimate the support of a match provided by the evidence as shown in [38]. It should be noted the Bayes factor is also known as the likelihood ratio in the forensic setting (see e.g. [40]), but the former terminology is

preferred here to stress the fact that the marginal likelihoods are results of integration rather than maximization of a likelihood function.

In the current case, it is assumed only one replicate measurement of two different samples corresponding to a single feature are available denoted by v_1 and v_2. The marginal likelihoods for this data under the different hypotheses are given by

$$p(v_1, v_2 \mid H_0) = \int_\Theta \prod_{k=1}^{2} f_i(v_k \mid \mu, \tau)\pi_i(\mu, \tau)\mathrm{d}\mu\mathrm{d}\tau \tag{2.72}$$

$$= p_{\text{marg}}(v_1, v_2) \tag{2.73}$$

and, assuming independence of the parameters for the distributions of the two samples under H_1,

$$p(v_1, v_2 \mid H_1) = \int_\Theta f_i(v_1 \mid \mu_1, \tau_1)\pi_i(\mu_1, \tau_1)\mathrm{d}\mu_1\mathrm{d}\tau_1 \int_\Theta f_i(v_2 \mid \mu_2, \tau_2)\pi_i(\mu_2, \tau_2)\mathrm{d}\mu_2\mathrm{d}\tau_2 \tag{2.74}$$

$$= p_{\text{marg}}(v_1)p_{\text{marg}}(v_2). \tag{2.75}$$

From these likelihoods, the Bayes factor in favor of H_0 regarding feature i is then given by

$$\text{BF}_i(v_1, v_2) = \frac{p_{\text{marg}}(v_1, v_2)}{p_{\text{marg}}(v_1)p_{\text{marg}}(v_2)}. \tag{2.76}$$

Then, for entire impurity profiles $\mathbf{v}_k = [v_k^{(1)}, \ldots, v_k^{(N_f)}]^{\mathrm{T}}, k = 1, 2$, define the total Bayes factor by taking the mean of each individual factor

$$\text{BF}(\mathbf{v}_1, \mathbf{v}_2) = \frac{\sum_{i=1}^{N_f} BF_i(v_1^{(i)}, v_2^{(i)})}{N_f}. \tag{2.77}$$

The values given by the Bayes factor can vary anywhere on the non-negative real line and, as such, converting it to a reasonable dissimilarity measure can be difficult. For the sake of comparison with the other methods the dissimilarity measure is in this work defined by

$$\text{BF}_{\text{d}}(\mathbf{v}_1, \mathbf{v}_2) = 2 \max_{\mathbf{v},\mathbf{w}} BF(\mathbf{v}, \mathbf{w}) - \text{BF}(\mathbf{v}_1, \mathbf{v}_2), \tag{2.78}$$

where the maximum is taken over all pairs of profiles $\mathbf{v}, \mathbf{w}$. This guarantees a non-negative measure that decreases as the support for H_0, and thus similarity, increases. It is not suggested that this is in any way reasonable metric, and it is only used for comparison.

An alternative, proposed in [38], is to compare the predictive distributions directly, as these distributions summarize all the knowledge available regarding the unknown distribution of the variables[41]. This comparison is done by integrating over the minimum

of the two predictive distributions induced by the two samples resulting in the measure called the predictive agreement given by

$$\mathrm{PA}(\mathbf{v}) = \int \min\{p(z \mid \mathbf{x}), p(z \mid \mathbf{y})\} \, \mathrm{dz}\,, \tag{2.79}$$

where the integral is computed over the entire space of possible observations z and $\mathbf{x}, \mathbf{y}$ denote the data at hand. The predictive agreement has been shown to be a conservative measure of similarity of distribution and furthermore it provides similarity scores within the interval $[0, 1]$ due to both predictive distributions being positive and integrating to 1.

Assuming the same setting as with the Bayes factor, the predictive agreement for feature i between two samples can be explicitly expressed as

$$\mathrm{PA}_i(v_1, v_2) = \int_{\mathbb{R}} \min\{p(v^* \mid v_1), p(v^* \mid v_2)\} \mathrm{d}v^*. \tag{2.80}$$

Similarly as in the case of the Bayes factor, the total predictive agreement between two samples is measured as the mean of predictive agreements over all features

$$\mathrm{PA}(\mathbf{v_1}, \mathbf{v_2}) = \frac{\sum_{i=1}^{N_f} PA_i(v_1^{(i)}, v_2^{(i)})}{N_f}. \tag{2.81}$$

The corresponding dissimilarity measure can be simply defined as

$$\mathrm{PA_d}(\mathbf{v_1}, \mathbf{v_2}) = 1 - \mathrm{PA}(\mathbf{v}_1, \mathbf{v}_2). \tag{2.82}$$

Finally, it should be noted that evaluating the predictive agreement is computationally intensive as it requires computing an integral for N_f features for each $N_o \in \mathbb{N}$ observations resulting in calculating

$$N_f \binom{N_o}{2} = \frac{N_f N_o (N_o - 1)}{2} \tag{2.83}$$

integrals in total. To speed up the computation, the algorithm for evaluating the predictive agreement was implemented in C++ utilizing the numerical integration routines provided by the `RcppNumerical` [42] R package.

Chapter 3

Testing the Methods on Real and Simulated Data

In this chapter, the methods developed previously are applied to real data and the results are described. Firstly, the available datasets are described in detail along with the methods used for simulating data. Secondly, the results of applying noise analysis as described in Section 2.1 are presented. Thirdly, the performance of the automated feature extraction presented in Section 2.2 is examined based on quality control samples and, lastly, the performance of the different comparison methods described in Section 2.3 is tested on the simulated data and their properties analysed by applying them to real data. The details of how the hyperparameters for the probability model used in the Bayesian comparisons were chosen are also discussed.

3.1 Real Data

In this study, two datasets of Gas Chromatography-Mass Spectrometry (GC-MS) [1] measurements of amphetamine samples were provided by the Forensic Laboratory of National Bureau of Investigation in Finland. The first one, consisting of measurements of two different quality control samples with 35 replicates for each, was used to estimate the noise model for the signal as well as the precision of computations. In addition, the performance of the impurity profile extraction algorithm was tested on this dataset. The second dataset consisted of GC-MS measurements of 90 amphetamine samples with no prior knowledge of association between any of the samples. These were used to produce parameters for data simulation as well as examine the differences between different comparison methods.

The raw data consisted of 301 ion channels monitored 5584 scans resulting in matrices of 5584 rows and 301 columns for each sample. The raw data was provided in the Analyti-

cal Data Interchange (ANDI) format, from which the signal matrices were extracted using tools given in R package `xcms`[43]. Also, information on the relevant target compounds was provided, containing for each compound the corresponding mass spectra, target and qualifier ion channels and their expected retention times.

3.2 Simulated Data

Simulated data was generated based on properties collected from the second dataset. While an attempt was made to produce as realistic data as possible, it should be noted that the underlying process is much more complex than suggested by the procedure here, which is unavoidably quite *ad hoc*. As such, the simulated data is only used to demonstrate the properties of the comparison methods. Regardless, the simulated data was generated so that each simulated sample consisted of signals generated in a retention time window relevant to each compound and only including the target and qualifier ion channels. This was done instead of generating entire chromatograms in order to save computation time since the filtering stage automatically discards any extraneous data in its first phase.

In order to simulate the data, some distributions had to assigned to characteristic features derived from the dataset. These distributions were fit through the simple method of moments, which involves estimating the sample moments and setting parameters of a distributions to match these. In the current setting, the moments estimated were the mean and the variance of the data.

The EGH peak model was assumed for all peaks and as such, an integral part of the simulation was to simulate corresponding peak parameters. For the asymmetry parameter τ, a normal distribution was assigned by setting the mean and variance as estimated from observed peaks. The retention time parameter or peak location t_R was generated by estimating the mean of the the distances from expected retention time of identified target peaks and using this value as standard deviation of a normal distribution centered at the expected retention time.

Many of the other parameters for simulation were generated from gamma distributions. In order to obtain the shape α and rate β parameters for these distributions, the sample mean m and variance v were first estimated from the data. Since these moments of a gamma distributed random variable X are given by

$$\mathbb{E}(X) = \frac{\alpha}{\beta} \text{ and } \mathrm{Var}(X) = \frac{\alpha}{\beta^2} \tag{3.1}$$

the method of moments allows estimating the distribution parameters by assigning these quantities equal to the sample moments. From this, the equation for mean allows us to

solve α in terms of β as

$$\begin{aligned} \frac{\alpha}{\beta} &= m \\ \Leftrightarrow \alpha &= \beta m. \end{aligned} \tag{3.2}$$

Using this with the equation obtained for variance gives

$$v = \frac{\alpha}{\beta^2} = \frac{\beta m}{\beta^2} = \frac{m}{\beta} \tag{3.3}$$

from which it is obtained that

$$\beta = \frac{m}{v}. \tag{3.4}$$

Inserting this to the equation for α gives

$$\alpha = \frac{m^2}{v} \tag{3.5}$$

and thus the method of moments estimators for α and β are obtained. Using this method, the distributions for simulating the peak width σ and peak height H were obtained based on observed peaks. The distributions for total sum of peak areas S_A, the ratio of internal standard peak areas to total sum of peak areas R_{IS}, the initial impurity quantities and the distribution for peak heights were obtained the same way . The total amount of peaks for each window was generated from poisson distribution based on the number of peaks detected in total in each window on average.

The actual simulation process begins by simulating for each target compound a value based on the observed target ion peak areas in the second dataset. These values are normalized by their sum to obtain the proportions of different compounds in the sample. Given this profile, the target peak areas are obtained by generating the total area S_A from the corresponding distribution and the internal standard peak area is simulated by multiplying the total area with a realization of the distribution for the ratio R_{IS}.

Then, simulation proceeds by generating for each compound the relevant target and qualifier ion channel signals. To this end, for each ion channel, target compound peaks are generated by simulating the peak width, asymmetry and retention time from their estimated distributions. The height is then obtained by finding the value that minimizes the squared difference between the area covered by the resulting peak function and the previously generated peak area. This value is found by using the basic optimization routines provided by R.

Additional peaks are simulated by first generating the total amount of peaks in the current ion channel and then separately acquiring the parameters for each peak with peak heights simulated from the gamma distribution fitted to all observed peak heights. The

height was restricted to be higher than 1% of and lower than 3 times the height of the target compound peak in the channel. In order to prevent these peaks from completely covering the target compound peaks, their retention times are restricted so that they are always at least 3 time units away from the target compound peak center. This setting was found to still allow for large amount of overlap while not completely drowning the "true" signal.

The contributions to the total signal from all peaks are computed by applying the EGH peak model to the simulated parameters and computing the values of the corresponding peak functions at each scan within the retention time window and summing them to a blank signal. This results in a "perfect" signal for a single ion channel.

In order to make the signal more realistic, baseline and noise are added. The baseline is simulated by first computing a baseline mean parameter μ_{bl} by calculating the median of simulated target compound peak heights and multiplying this value by a factor generated from Gamma$(50, 1000)$ distribution, parametrized by shape and rate. The distribution is assigned so that the multiplying factor is usually close to 0.05, but is allowed to occasionally be much larger or smaller in order to emulate random levels of baseline fluctuation. Once the mean has been generated, five random values $c_i, i = 1, \ldots, 5$ from uniform distribution in the interval $(0, 0.5)$ are generated and five corresponding variance parameters are computed as $\sigma^i_{\text{bl}} = (c_i * \mu_{\text{bl}})^2, i = 1, \ldots, 5$. Using the method of moments, these are used to obtain five shape and rate parameters for gamma distributions. From each of these five distributions a single value is generated resulting in five random numbers. These are assigned to five equally spaced scans within the window, starting and ending with the first and last scan of the window. Then spline interpolation facilitated by the `spline` function included in the basic R installation is applied to the square roots of these values and, thus, the basic shape of the baseline is obtained by squaring these values. The reason for taking a square root is to avoid negative values in the resulting curve. This process and possible baselines are illustrated in Figure 3.1.

Finally, noise is added to the baseline by generating it from a normal distribution centered at the baseline with standard deviation depending on the values of the baseline according to the model developed via the method from Section 2.1. This noisy baseline is added to the signal resulting in the simulated signal corresponding to a single ion channel. This process is repeated for each ion channel and each compound until entire sample has been simulated.

In Figure 3.2 can be seen a comparison of simulated and real signal. The simulated data is more noisy and has higher tendency to produce overlapping peaks which makes it more difficult to process than real signal, but for simulation purposes this should be sufficient and it would appear the simulated data captures the characteristics of the real data reasonably well.

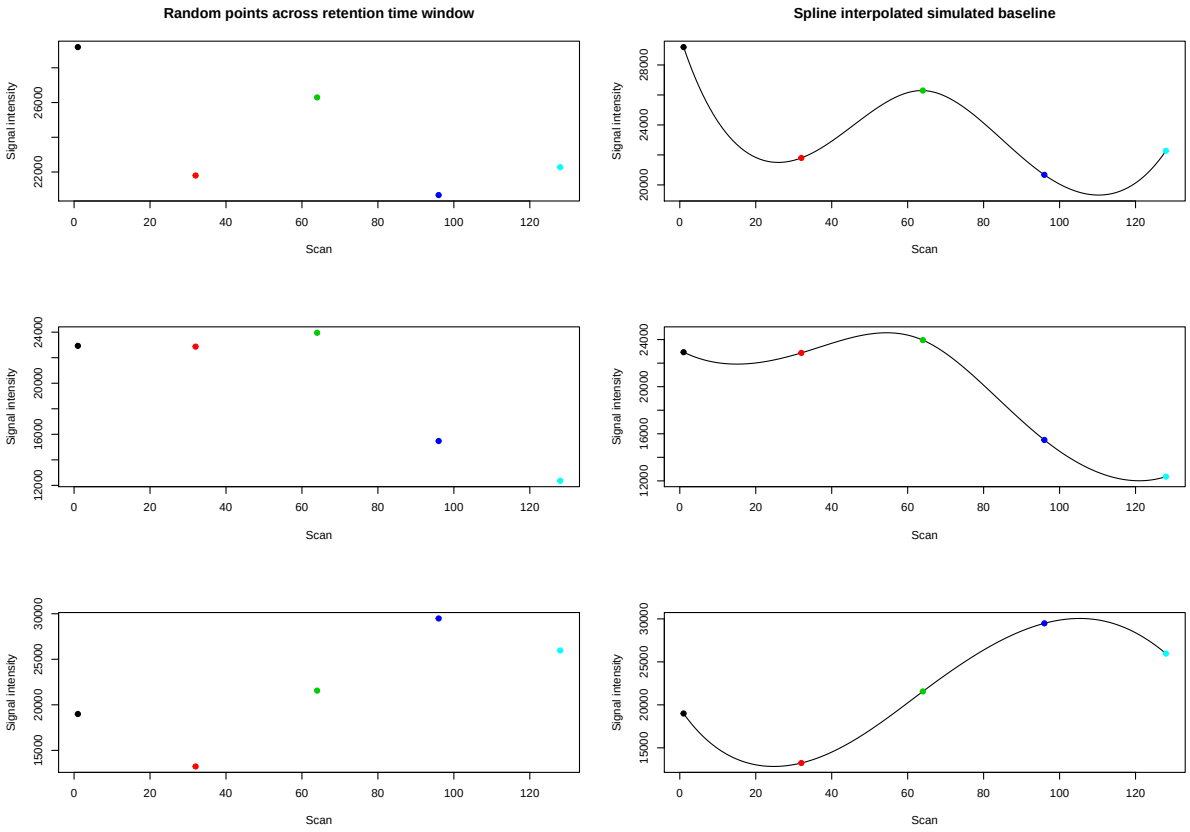


Figure 3.1: Randomly simulated baselines with baseline mean randomly set to 23435.92 and with baseline variance randomly generated for each case. Each row corresponds to one instance of generating the 5 points through which a spline interpolation is performed to obtain a baseline.

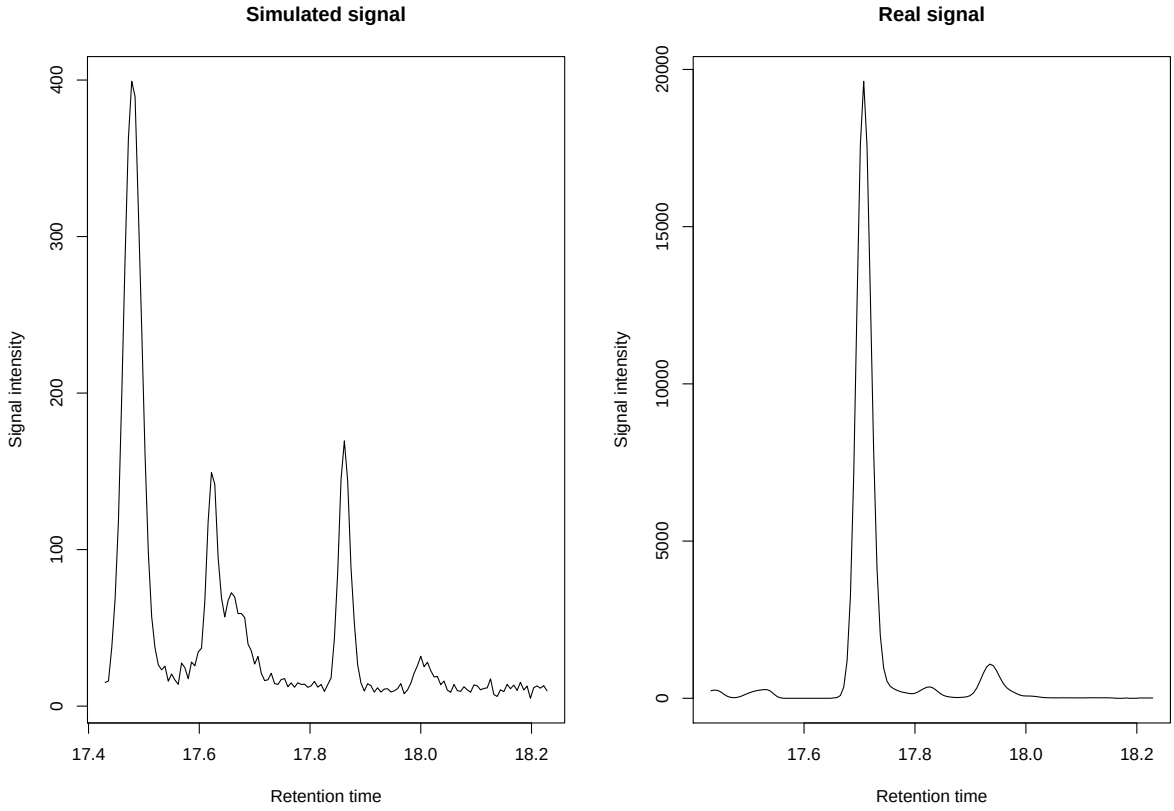


Figure 3.2: Comparison of simulated and real signal. Simulated signal is noisier and in this case has quite low intensity level.

3.3 Noise Analysis of Quality Control Samples

The noise analysis described in Section 2.1 was applied to the quality control samples and the resulting noise model was used in both the data processing and simulation. The extraction of signal mean level and standard deviation pairs was done separately for each of the two sets of 35 replicates corresponding to different samples and, as no difference between distributions of the obtained pairs was found, these datasets were combined to a single set of points.

The resulting dataset for noise analysis consisted of over 60000 points making direct application of kernel methods relatively slow. To this end the binning method described in Section 2.1 was applied to reduce the data set to 5469 binned points when grid spacing of 1 was used. Using these points, leave-one-out cross-validation was used to identify the best smoothing parameter for the kernel regression.

The results of the cross-validation are given in Figure 3.3 and it can be seen that a clear minimum is achieved in mean squared error. The smoothing parameter value minimizing the cross-validation error for the regression function was found to be approximately 2477. The resulting fit is plotted in Figure 3.4.

From the figure, it would appear a reasonably good fit has been obtained. Also, it is evident that the data indeed contains some structure that does not appear to be entirely linear. Interestingly, there would also appear to be considerable levels of heteroscedasticity meaning that the variance of the noise standard deviation seems to increase as the mean signal level increases.

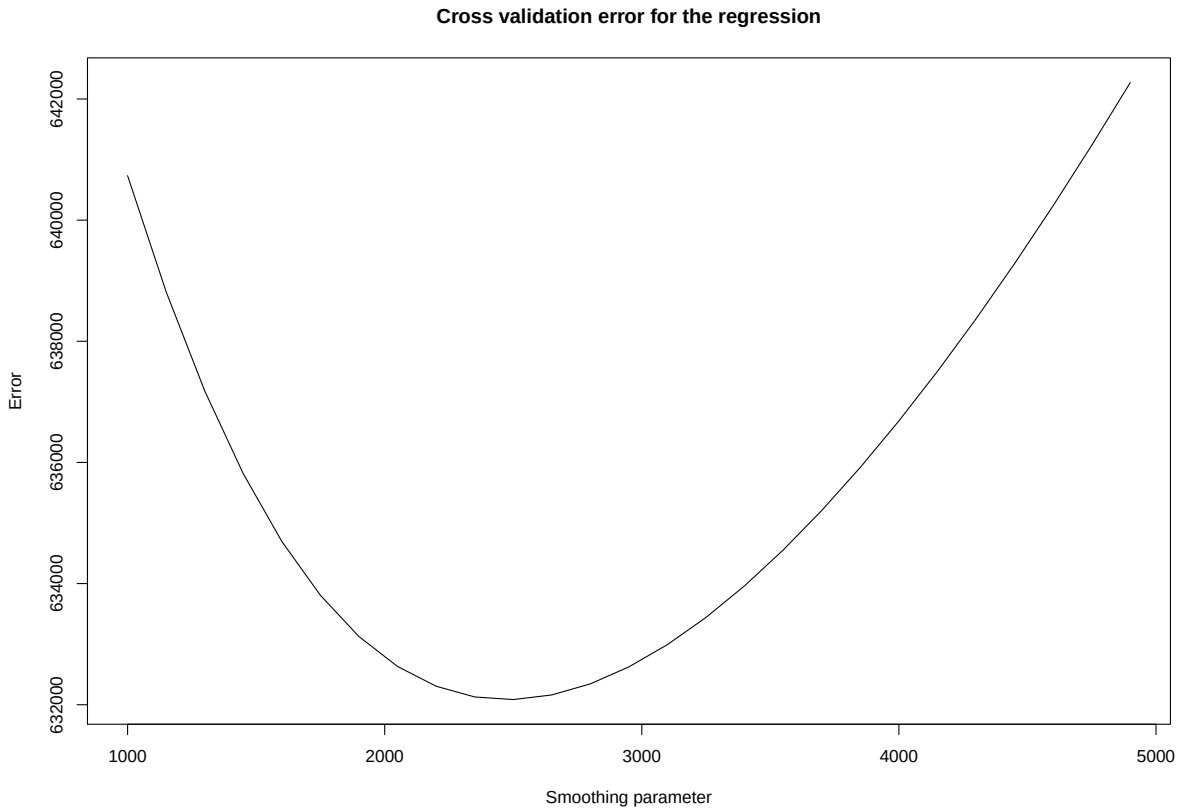


Figure 3.3: Cross-validation errors for the regression.

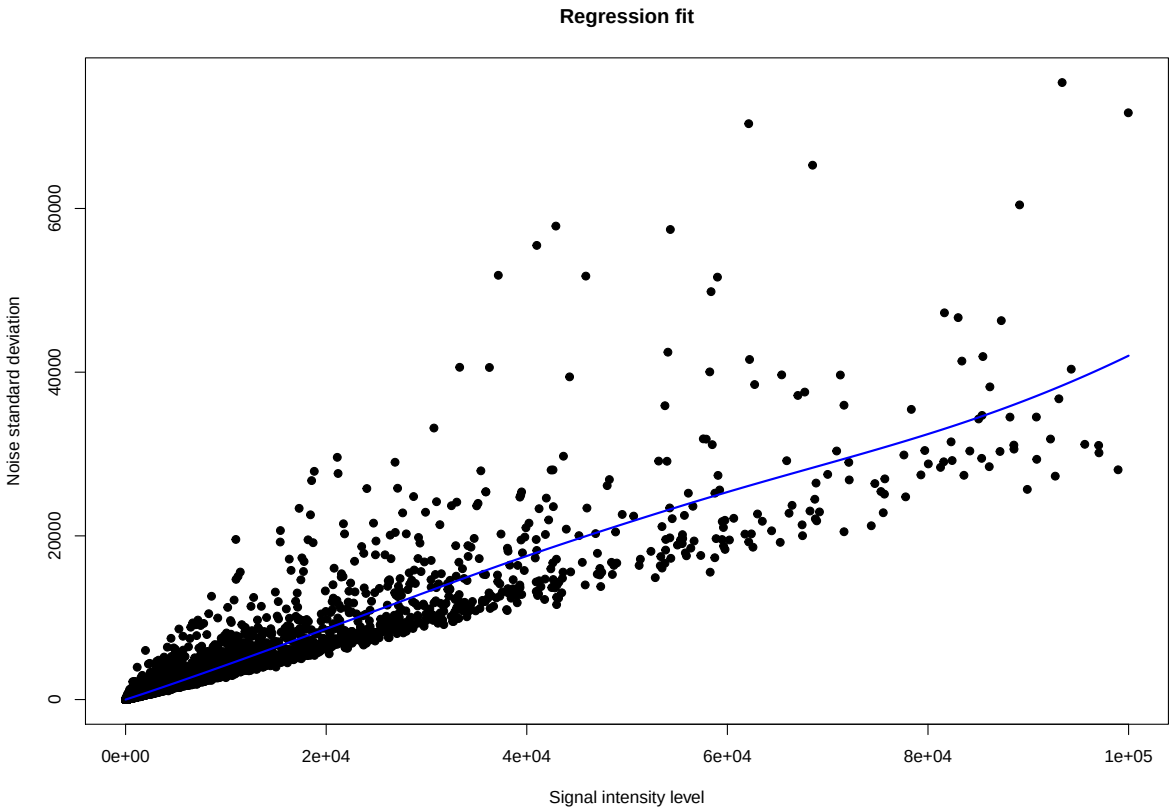


Figure 3.4: The regression fit to data.

3.4 Performance of the Impurity Profile Extraction

The feature extraction method described in Section 2.2 was tested using the quality control samples by inspecting the repeatability of the results based on the two sets of 35 replicates. An important aspect of tuning the method is to strike a balance between false positive and negative identifications of compounds. In excess, the former would result in inaccurate matches or irrelevant source of dissimilarity between samples, while the latter would result in there being too few identified compounds to reliably compare samples. Truth of the matter is, however, that an automated system is not likely to handle every situation perfectly.

Figures 3.5 and 3.6 illustrate, that, for each replicate of the samples, similar profile is consistently extracted. However, for both samples there is some variation on some of the smaller features as can be seen in the figure. This can be explained by the fact that the quality control sample measurements used as raw data here are used to test the current state of the equipment. As such, in some cases the equipment is simply dirty which results in either noisy measurements, which drown out the relevant signal, or even residual compounds from previous samples, which result in incorrectly identified compounds. Furthermore, the quality control samples used here have relatively low concentration and the reliability of the system could be improved by using higher concentrations. The effect of concentration on the results was not considered in this work, however. As expected, the comparison methods were able to separate these samples from each other perfectly.

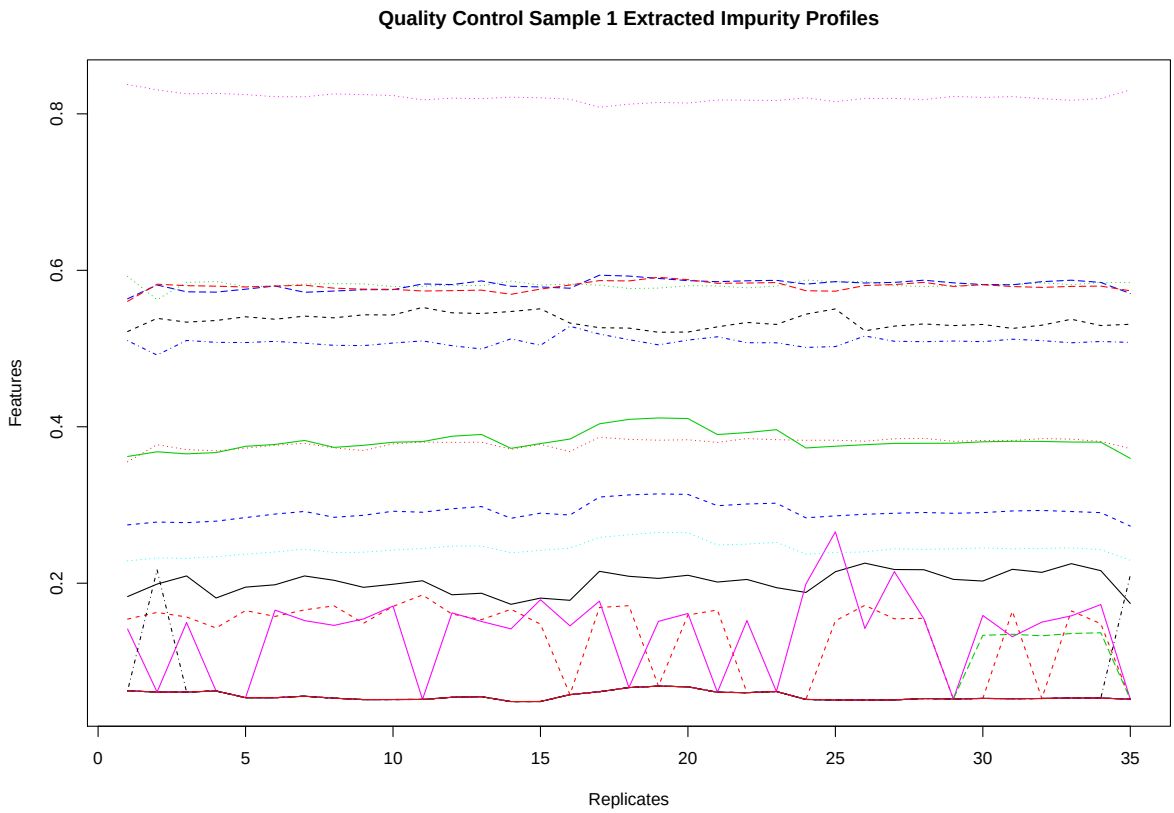


Figure 3.5: Profiles for quality control sample 1.

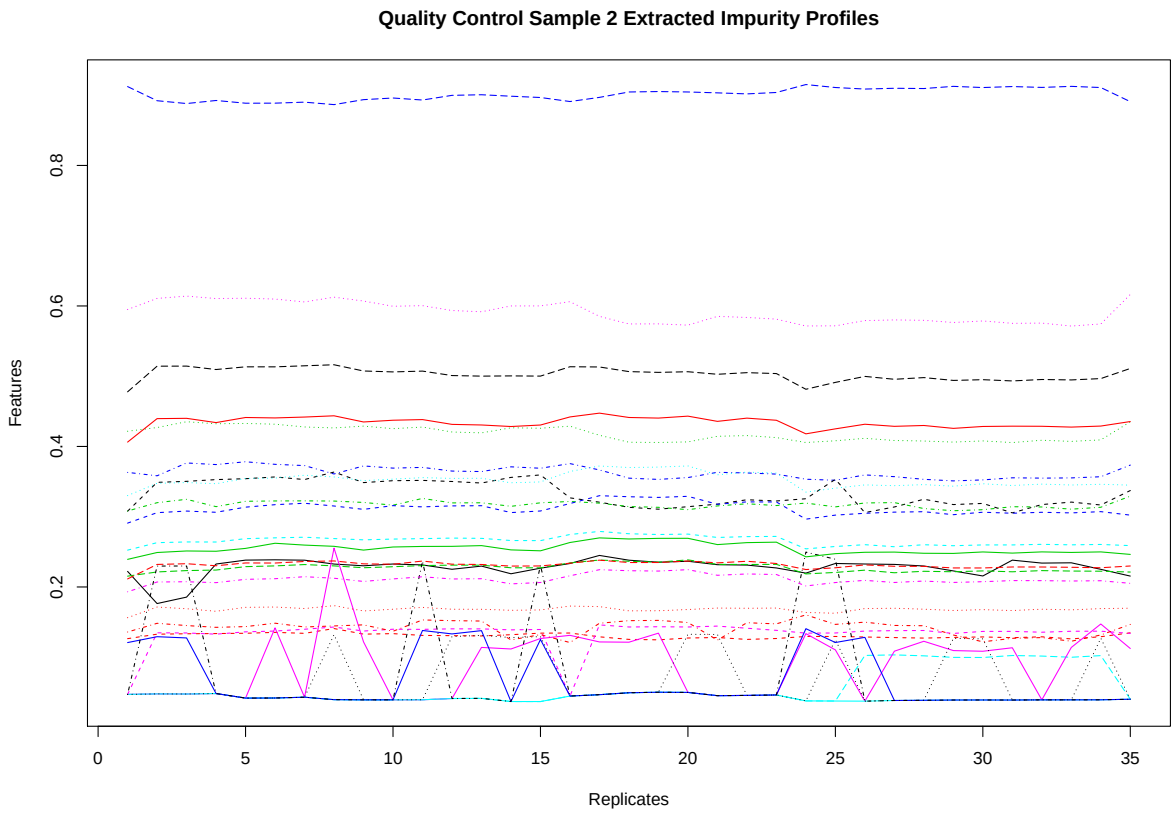


Figure 3.6: Profiles for quality control sample 2.

3.5 Simulation Study of the Impurity Profile Comparison Methods

In order to apply the Bayesian methods, the hyperparameters had to be estimated first as discussed in Section 2.3.3. Since features were zero centered, mean could simply be taken as zero and, in order to remain uninformative about the mean, the parameter κ was set to 10^{-6} to limit the effect of this choice on the comparison. The challenge came from the fact that replicate measurements were only available from two samples. Because of this, it was not possible to analyze each feature separately as several features were not present in the available samples. Regardless, after discarding all features that were not present in every replicate for either of the two samples, it was found that the zero centered logistically transformed impurity profiles showed very little difference between features in terms of standard deviation as is demonstrated in Figure 3.7. It can also be seen that even the differences between the samples are relatively minor.

Thus, a naive assumption was made here that the variance, and thus its reciprocal precision, was the same for each feature. Furthermore, it was assumed the variance does not depend on the sample. While it can be seen that reasonable results could be obtained with these assumptions, it should be emphasized that more rigorous estimation of the hyperparameters should be done, but this would require kind of data that was not available for the current work. In any case, considering the zero centered features share a mean, these assumptions allowed combining the results for each feature resulting in $n = 1120$ replicate measurements of a single combined feature. From this the hyperparameters were estimated using the rules (2.68) for the posterior parameters of the normal-gamma model, by further assuming an improper prior distribution with $\alpha = 0$ and $\beta = 0$ for the gamma component and $\mu_0 = 0, \kappa = 0$ for the normal component. This resulted in choosing the hyperparameters α_p and β_p as

$$\alpha_p = \frac{1120}{2} = 560 \quad \text{and} \quad \beta_p = 0.5 \sum_{i=1}^{n} (x_i - \bar{x}) \approx 1.91, \tag{3.6}$$

where x_i are the feature values of the n measurements and $\bar{x}$ is their mean and zero terms were dropped.

As there is no guarantee these hyperparameters are optimal and the expected value for the precision implied by these parameters was optimistically high, the α_p parameter was scaled down by different factors with β_p kept fixed to reduce the mean and increase the variance of the prior distribution for the precision parameter, thus making the corresponding Bayesian similarity measures more lenient. For more details on how the choice of hyperparameters affects the predictive distribution, see the illustration in Appendix A.

Simulation was run for 1000 rounds, with three samples were generated each round.

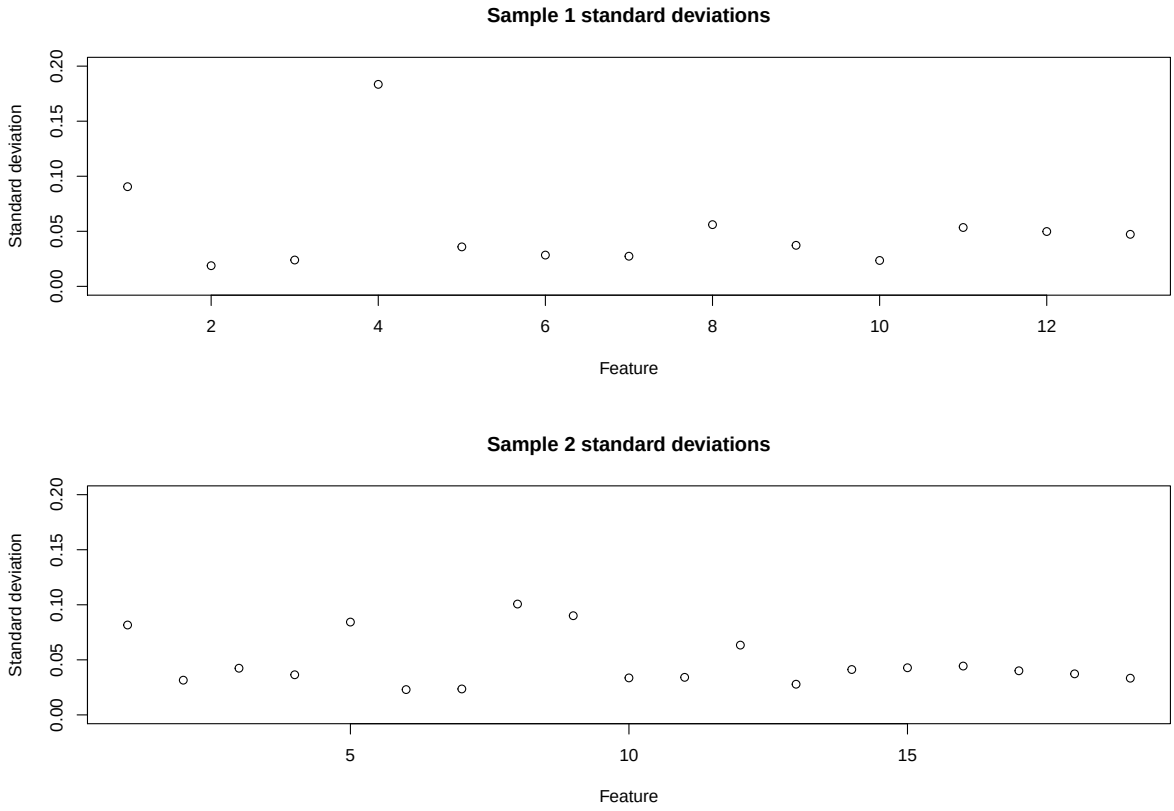


Figure 3.7: Standard deviations of features with no missing peaks for the two quality control sample replicate measurements. The differences in standard deviations are relatively low, even between samples.

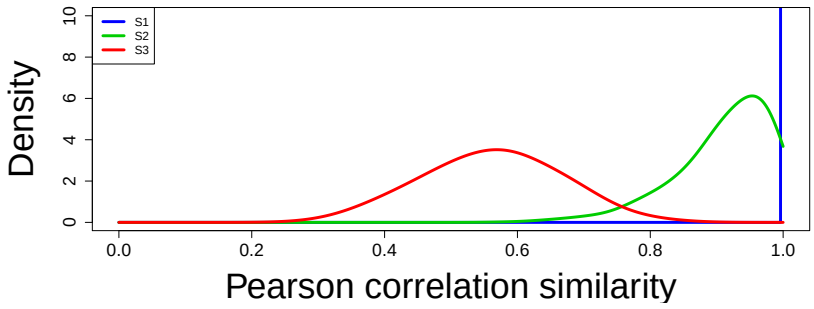


Figure 3.8: Densities of values produced by the Pearson correlation similarity measure under different scenarios.

Sample 1 and 2 were simulated using the same impurity profile and sample 3 using a different profile. It should be noted that in reality, samples from the same source would have very similar chromatograms, but the current simulation method could result in generating vastly different signals for each simulated sample even when the original profiles were the same. As in [5], three different scenarios were considered:

S1 Identical samples, with sample 1 compared to sample 1.

S2 Samples from same source, with sample 1 compared to sample 2.

S3 Samples from different source, with sample 1 compared to sample 3.

The first scenario is not very realistic, but was included mainly to anchor the untransformed Bayes factor, as well as to see how well different methods separate perfectly identical and similar samples.

The Pearson correlation similarity COR , and the predictive agreement PA and Bayes factor BF, using the different hyperparameters, were applied to each scenario each round. Densities for the similarity values in each scenario and with each method obtained with the R function `density` are shown in Figures 3.8, 3.9 and 3.10 for correlation, predictive agreement and Bayes factor respectively. The minimum, maximum and mean values as well as the standard deviation and second and third quantiles of the comparison regarding scenarios 2 and 3 are also presented in the Tables 3.1 and 3.2.

The results show that the correlation measure seems unable to give a score below 0.5 and there seems to be more overlap between the scenarios. The Bayesian methods manage to separate the scenarios **S1** and **S2** better, but the influence of the hyperparameter α_p is considerable. When $\alpha = 56$, the predictive agreement seems to offer the most satisfying result, as similar samples consistently receive scores well above 0.5 and dissimilar samples

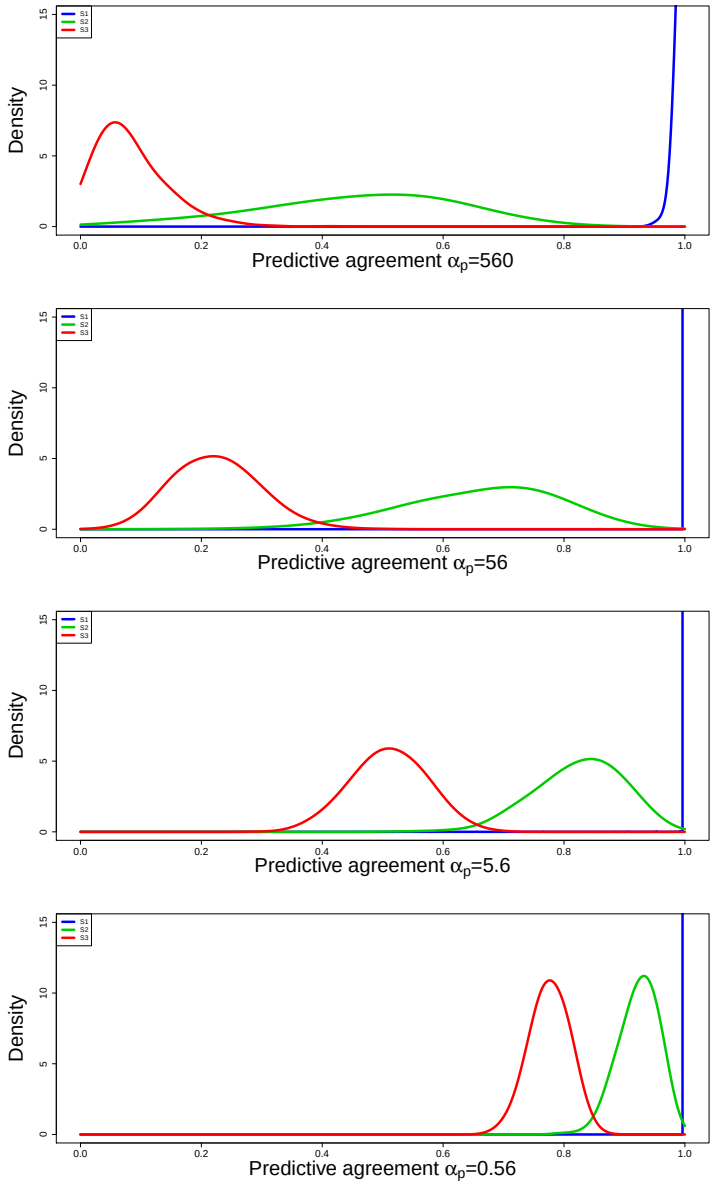


Figure 3.9: Densities of values produced by the predictive agreement under different scenarios using different values for α.

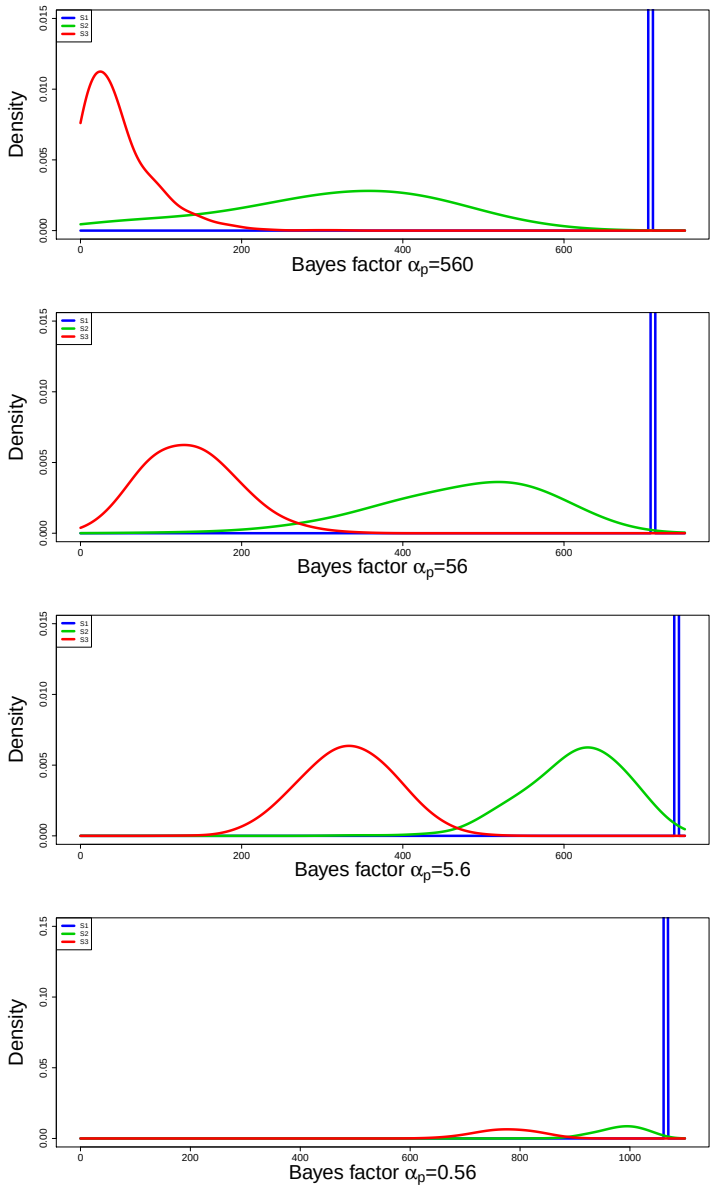


Figure 3.10: Densities of values produced by the Bayes factor under different scenarios using different values for α.

S2							
Method	α_p	Min	1st quantile	Mean	3rd quantile	Max	SD
COR	N/A	0.58	0.88	0.91	0.97	1.00	0.07
PA	560	0.01	0.35	0.46	0.58	0.84	0.16
BF	560	0.34	232.07	315.50	414.51	626.74	132.84
PA	56	0.15	0.58	0.66	0.75	0.94	0.12
BF	56	66.82	411.43	474.94	551.25	697.69	103.31
PA	5.6	0.51	0.78	0.82	0.88	0.98	0.07
BF	5.6	329.86	576.72	614.30	659.07	737.80	60.31
PA	0.56	0.78	0.90	0.92	0.95	0.99	0.03
BF	0.56	789.39	958.32	983.11	1014.33	1064.37	42.87

Table 3.1: Statistics from applying comparison methods to simulated data. Scenario 2.

S3							
Method	α_p	Min	1st quantile	Mean	3rd quantile	Max	SD
COR	N/A	0.28	0.49	0.56	0.63	0.84	0.10
PA	560	0.00	0.04	0.08	0.11	0.43	0.06
BF	560	0.00	15.38	46.65	65.34	319.82	43.26
PA	56	0.03	0.18	0.22	0.27	0.50	0.07
BF	56	2.40	93.80	136.21	172.50	342.29	56.70
PA	5.6	0.30	0.47	0.51	0.55	0.69	0.06
BF	5.6	132.89	294.78	332.42	372.95	506.11	55.92
PA	0.56	0.67	0.75	0.78	0.80	0.87	0.03
BF	0.56	600.93	739.09	775.62	815.05	927.26	53.84

Table 3.2: Statistics from applying comparison methods to simulated data. Scenario 3.

are well below 0.5, allowing an intuitive interpretation of these values as probabilities of similarity. While the results from the Bayes factor seem to be consistent with ones from predictive agreement interpretation seems more difficult. Bayes factor gives under all scenarios values well above 1, in the order of hundreds in fact, theoretically implying support for the hypothesis that the samples are similar in almost every case.

To evaluate how well the different measures can classify the samples to similar and dissimilar given a sample to compare to, receiver operating characteristic (ROC) [44] curves are also plotted in Figures 3.11 and 3.12 with relevant areas under ROC curve (AUC) presented in Table 3.3. The AUC is a common measure for estimating classifier performance and the closer its value is to 1 the better the measure is at indicating similarity. The AUC value can be interpreted as the probability of ranking a randomly selected similar

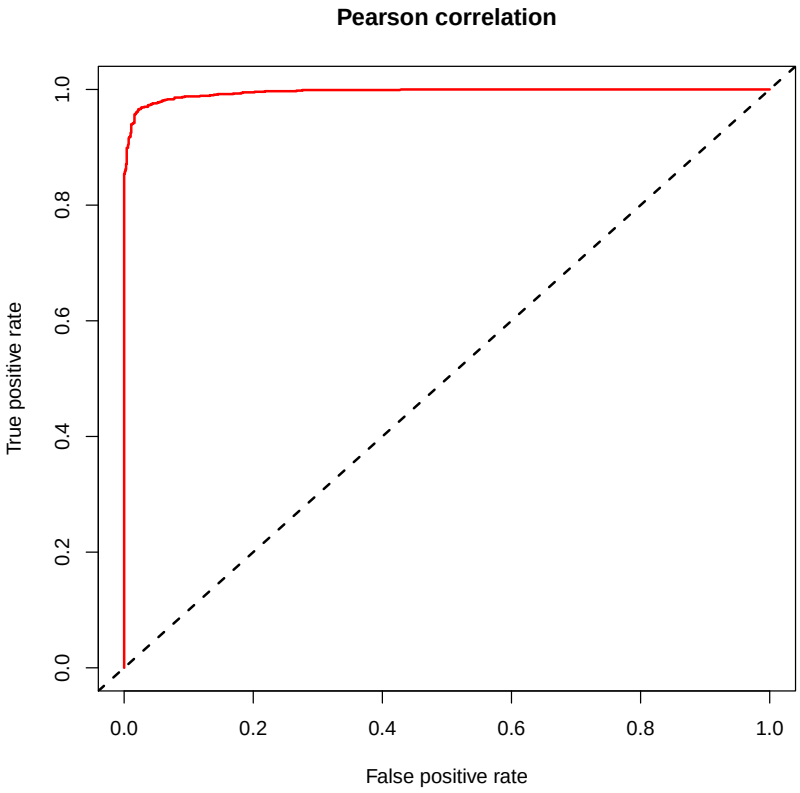


Figure 3.11: ROC curve for Pearson correlation similarity.

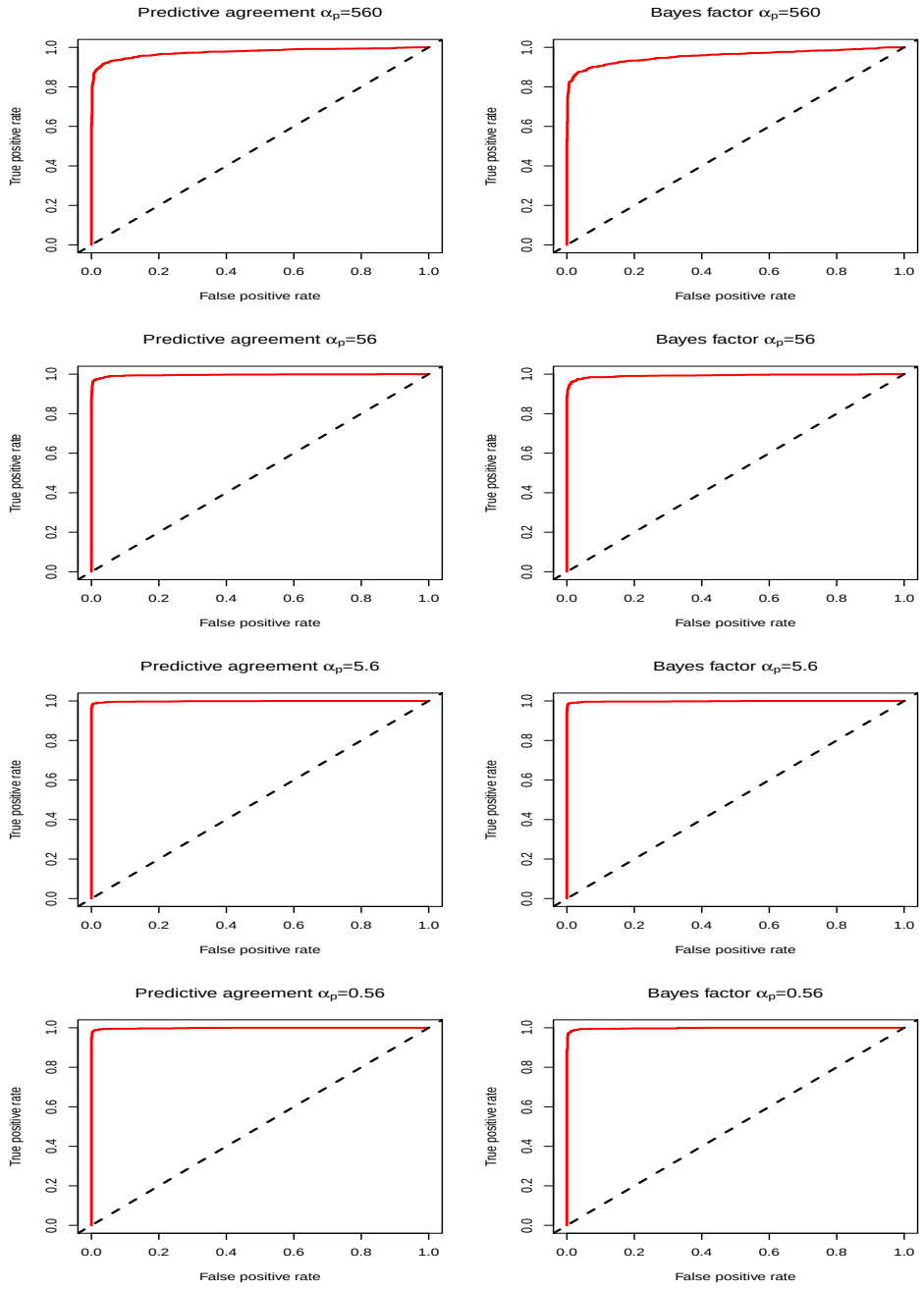


Figure 3.12: ROC curves for predictive agreement and Bayes factor for different values of parameter α.

Measure	α_p	AUC
Correlation	N/A	0.99540
PA	560.0	0.97561
PA	56.0	0.99621
PA	5.6	0.99851
PA	0.6	0.99847
BF	560.0	0.95694
BF	56.0	0.99312
BF	5.6	0.99833
BF	0.6	0.99792

Table 3.3: Areas under ROC curve (AUC) for different measures.

pair higher than randomly selected dissimilar pair. The basic idea of the ROC curve is that a possible classification bound is varied across a grid and for each bound the false positive rate, which is the proportion of dissimilar pairs erroneously labelled as similar, and the true positive rate, which is the proportion of similar pairs correctly labelled as similar, are collected and plotted against each other. For the Bayes factor the grid was set based on the minimum and maximum values of the measure while for correlation and predictive agreement grid set between 0 and 1 was used. The closer the curve is to the upper left corner, the better.

The ROC curves show that each measure is capable of separating similar and dissimilar samples given a reference sample reasonably well. The behaviour of the Bayesian methods is almost identical, although slight difference can be seen, especially with the highest value of α_p, which seems to offer the poorest performance for both Bayes factor and predictive agreement. However, correctly selected α_p results in both the predictive agreement and the Bayes factor outperforming the correlation measure. This is also evident from the AUC measures although the differences are not massive. This would however suggest that the Bayesian methods are able to perform at least as well as the correlation measure and with careful modelling, they could be able to even outperform the classical method.

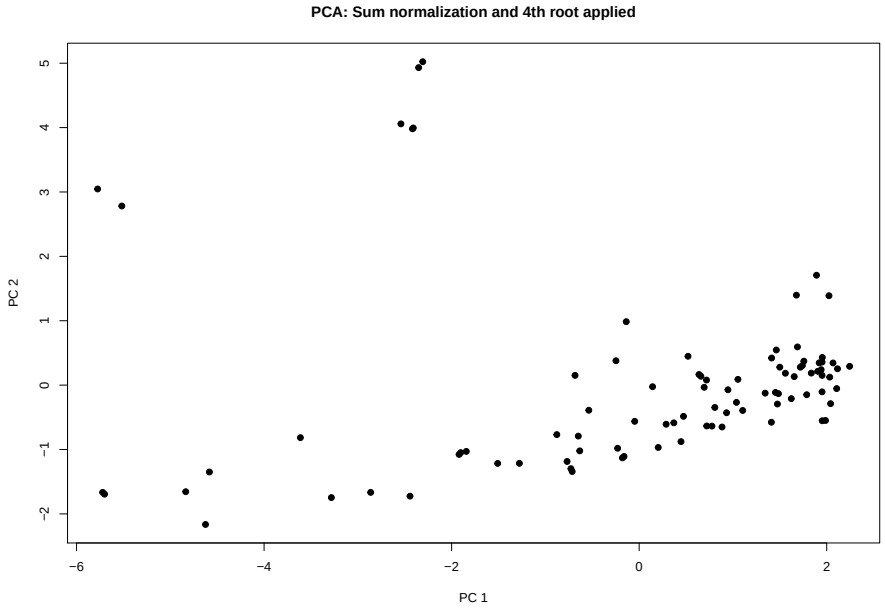
3.6 Applying the Methods to Real Data

The feature extraction and comparison methods were also applied to the second dataset containing 90 real samples of confiscated amphetamine. While no ground truth of these samples is known, they can nevertheless be used for illustrative purposes to examine how the different comparison methods treat the data. To this end, the dissimilarity measures corresponding to the different comparison methods from Section 2.3 were used

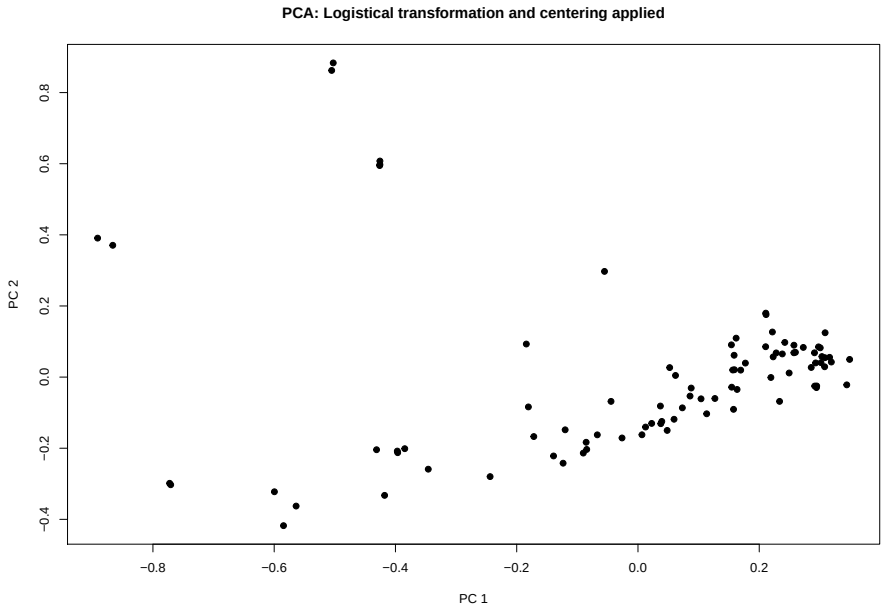
to obtain dissimilarity matrices of the data. As the hyperparameter value $\alpha_p = 5.6$ seemed to produce the best results based on the analysis of the previous section, only the corresponding Bayesian measures are used in this section.

For visualization of the results, classical multi-dimensional scaling (MDS) [45] was used to obtain a 2D representation of the data. Principal component analysis (PCA) (see e.g. [46]) corresponding to multi-dimensional scaling on euclidean distance matrix was also used for the sake of comparison. The basic idea of the classical metric MDS method is to construct, in this case, a 2D representation of given dissimilarities by finding coordinates for data points that would produce similar euclidean distances. This is useful for visualizing data in low dimensions when a dissimilarity matrix can be obtained. PCA, on the other hand, uses the eigenvectors of the covariance matrix of the data to produce a set of uncorrelated coordinates, called principal components, for the data, with each explaining as much of the variance in the data as possible. By selecting only the most important components, data dimensionality can be reduced. The R functions `cmdscale` and `princomp` were used to perform these transformations. Lastly, PCA was applied to both the impurity profiles that were treated by normalizing by their sums before taking 4th root and to the logistically transformed zero centered profiles.

Figures 3.13a and 3.13b show the results of applying PCA to the differently pretreated impurity profiles. From these images it can be seen that the additional logistic transformation has little effect on PCA, although the points do appear bit more clumped. In Figures 3.14, 3.15 and 3.16 are the results of multi-dimensional scaling on the distance measures. The structure of the data seems quite similar to that produced by PCA but some differences can be seen. The correlation distance seems to produce an even more cramped representation than PCA while predictive agreement and Bayes factor cause the data points to be more spread out. It would appear that the Bayesian methods are able to produce a bit more separation than the correlation measure.



(a) PCA applied on the impurity profiles used for correlation computation.



(b) PCA applied to the impurity profiles used for Bayesian measures.

Figure 3.13: PCA applied to differently pretreated impurity profiles.

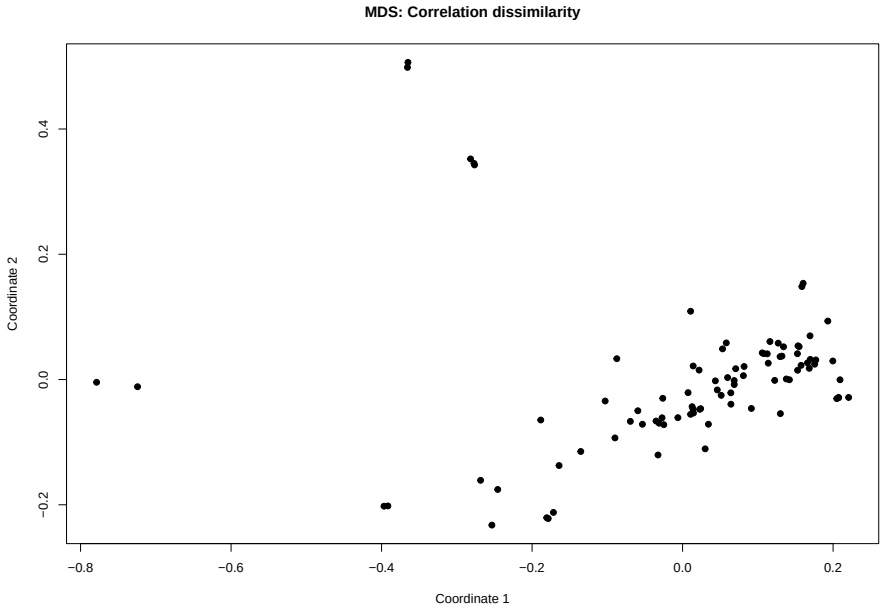


Figure 3.14: MDS applied to correlation distances.

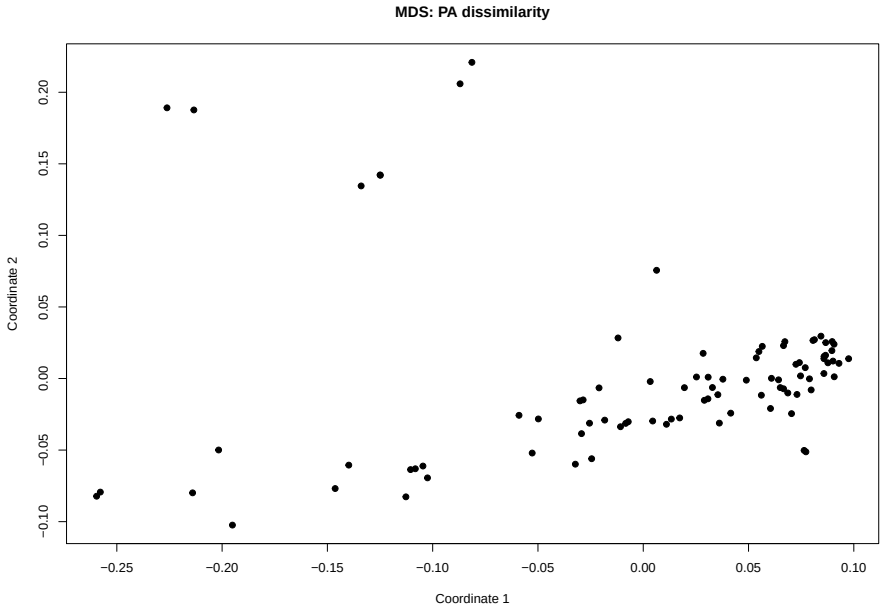


Figure 3.15: MDS applied to PA distances.

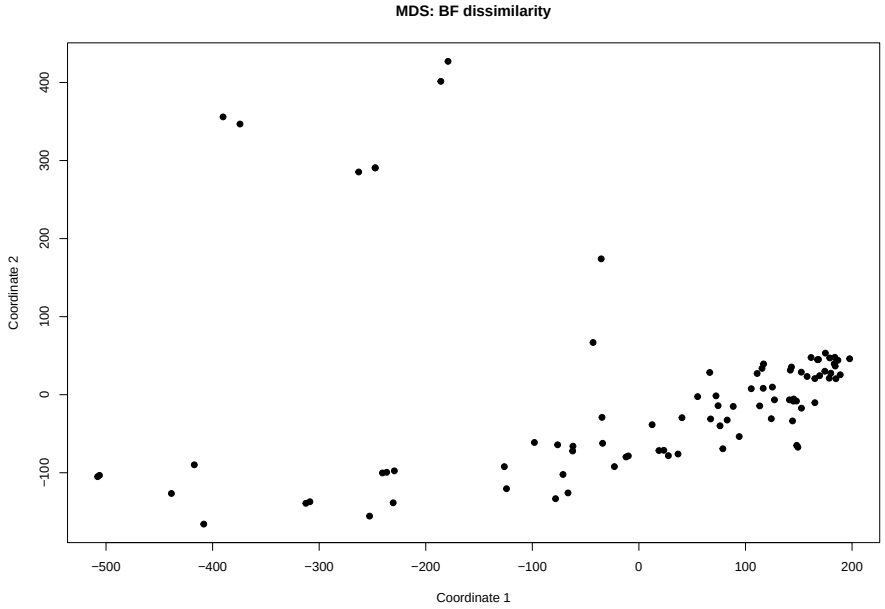


Figure 3.16: MDS applied to BF distances.

Chapter 4

Discussion

In this work methods for noise analysis, automatic impurity profile extraction given target compounds and impurity profile comparison for chemical data produced by chromatography-mass spectrometry have been presented. These methods were applied to real and simulated data and their performance was analysed.

The noise analysis methods introduced were used to identify a clear and non-linear relationship between the signal level and noise standard deviation and local linear regression was used to obtain a satisfactory model for this relationship. Although more study might be warranted on how to deal with the apparent heteroscedasticity in the data, the current method was able to capture the characteristics of the data well, even when approximative binning methods were used. Cross-validation was found to be a useful tool for finding a decent smoothing parameter for the non-parametric regression.

The obtained noise model was successfully applied in the automatic impurity profile extraction. The impurity profile extraction algorithm was found to produce highly consistent estimates with rather noisy data. The performance of the method was not perfect, however, and more comprehensive study should be conducted regarding identifying the peaks. A method similar to one seen in [47] might be used to introduce a more probabilistic approach to identifying peak regions instead of the simple detection limits as used in this study. Regardless, the process was found to be reasonably reliable and was able to produce requested variables from the data, thus limiting the need for human effort in the process.

The Bayesian comparison methods implemented in this work were found to produce similarity measures on par with the correlation measure that is the current standard in the comparison of chemical profiles of amphetamine in forensics. The Bayesian methods would appear to have the potential to surpass the classical method, while also introducing more rigorous statistical basis to the comparison, as long as the model is correctly chosen. It is important to note that the probabilistic model used for the comparison was relatively

crude and, as seen from the tests on the simulated data, much care should be taken to the selection of hyperparameters. Additional research would be needed to develop a robust method for selecting these hyperparameters in this context, and it might be of interest to study the possibility of applying other probabilistic models besides the simple normal-gamma model used in this work. Furthermore, the predictive agreement was found to be an intuitive and accurate measure of similarity. It overcomes the problems of the Bayes factor in that it has bound range, and it was also found more conservative in the sense that, while Bayes factor tends to always result in support for a match, the predictive agreement more readily gave low scores to dissimilar samples.

In conclusion, it is hoped that this work will provide a basis from which further research into forensic drug comparison can be conducted. The methods here were applied only to amphetamine data, but their versatility should allow them to be applied to similar cases in cocaine and methamphetamine comparison quite naturally. Further applications could be found in oil sample comparison and analysis of residues from suspected arson cases.

Appendix A

Illustrations

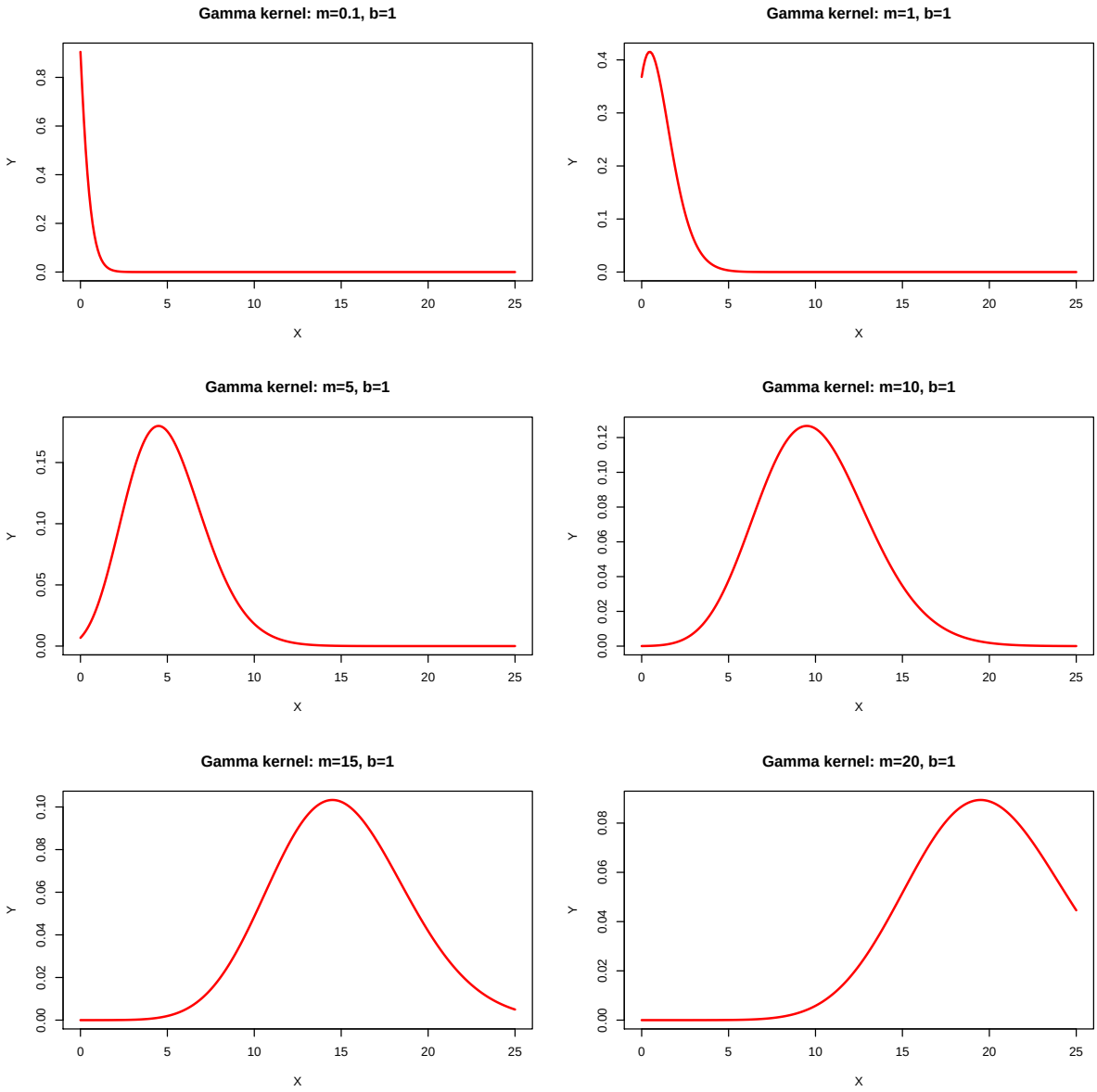


Figure A.1: Gamma kernels with different center parameters m and fixed smoothing parameter $b = 1$. As can be seen, the gamma kernel becomes wider as m increases. Changing b simply scales the kernel.

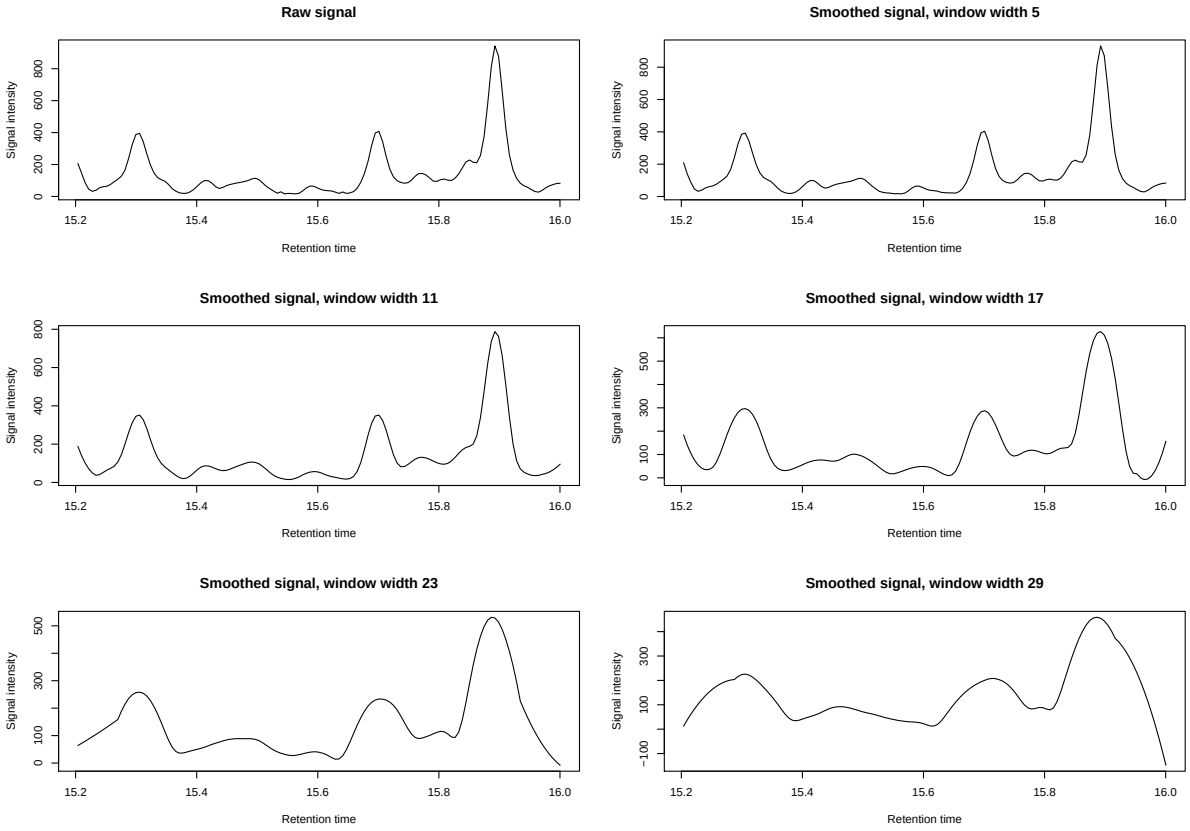


Figure A.2: Effect of window width on smoothing done by the Savitzky-Golay filter.

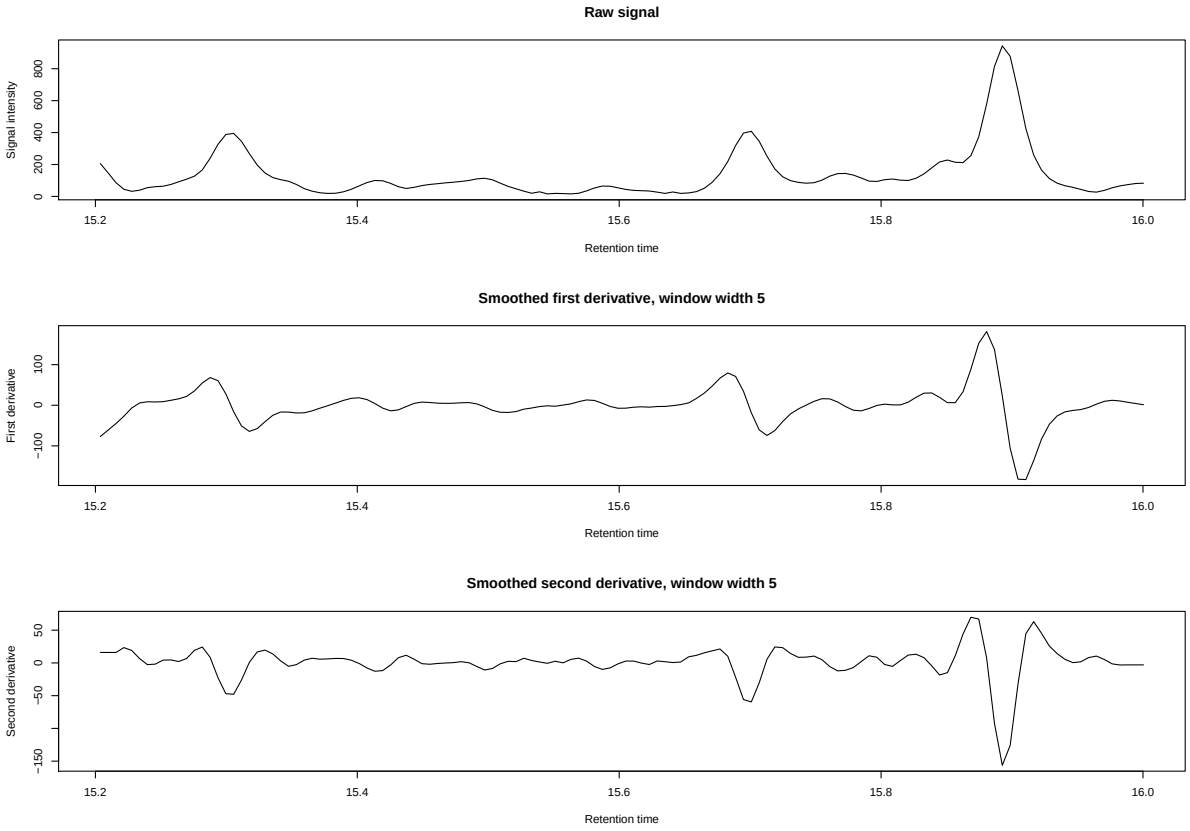


Figure A.3: Signal derivatives given by Savitzky-Golay filter with window width 5.

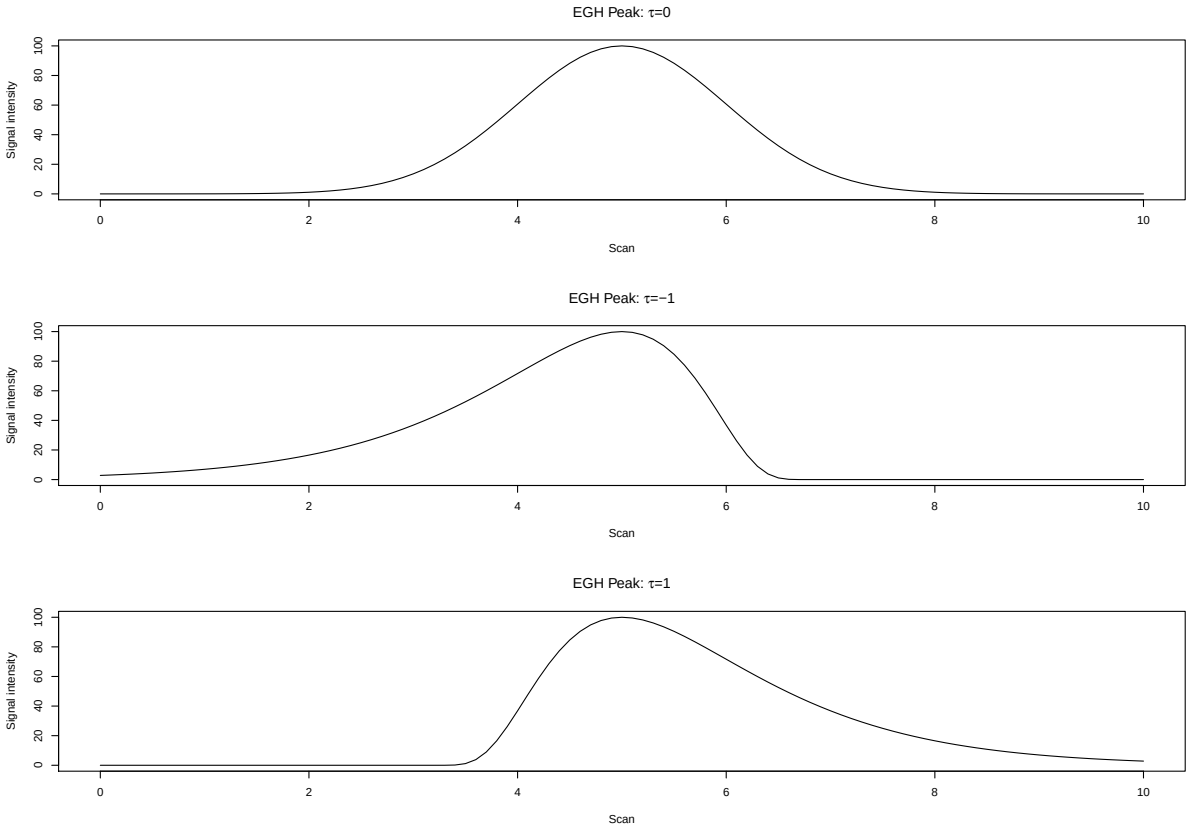


Figure A.4: Illustration of the effect of the asymmetry parameter on the EGH peak shape.

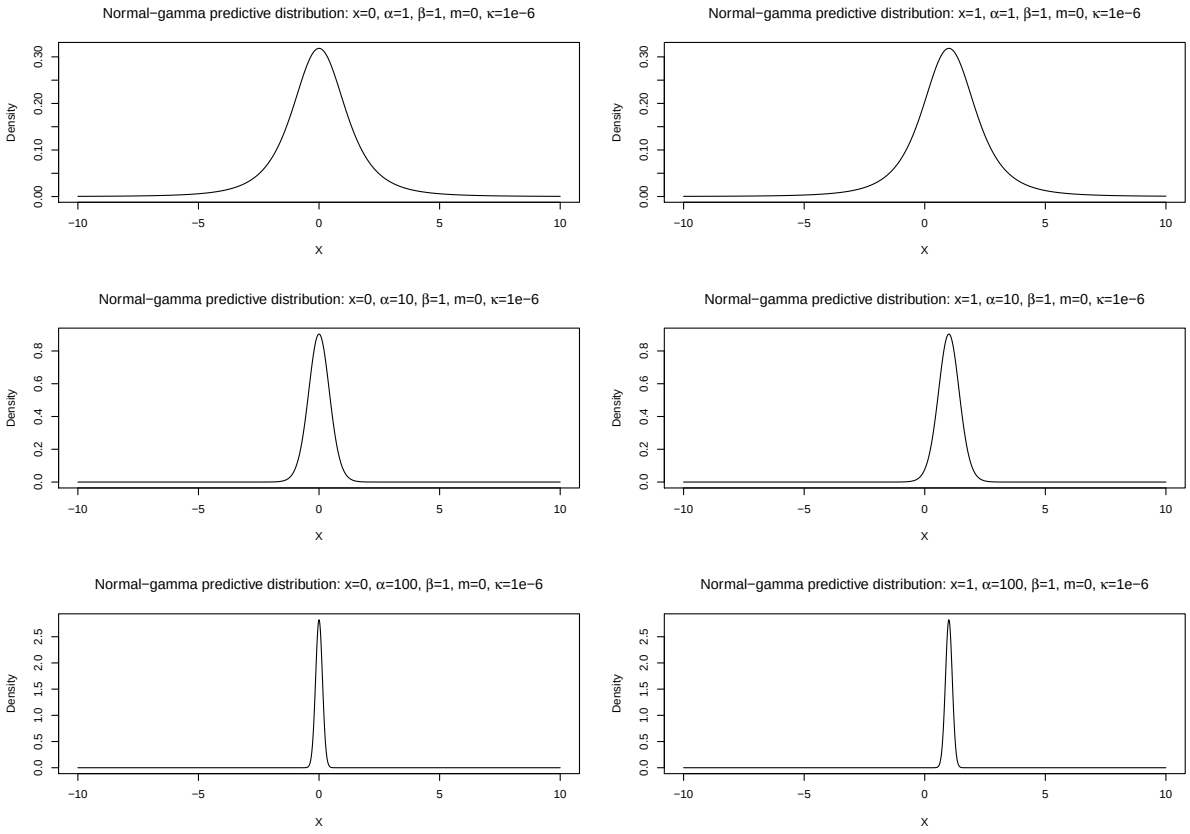


Figure A.5: Illustration of the effect of changing the parameter α on the predictive distribution derived from the normal gamma model. As can be seen, high levels of gamma cause the distribution to center more quickly around the observation x.

Bibliography

[1] M. M. Houck and J. A. Siegel, *Fundamentals of forensic science*. Elsevier Ltd., 2006, pp. 131–149.

[2] K. Dettme-Wilder and W. Engewald, *Practical gas chromatography*. Springer, 2014.

[3] K. Andersson, E. Lock, K. Jalava, H. Huizer, S. Jonson, E. Kaa, A. Lopes, A. Poortman-van der Meer, E. Sippola, L. Dujourdy, and J. Dahlén, "Development of a harmonised method for the profiling of amphetamines VI: Evaluation of methods for comparison of amphetamine", *Forensic Science International*, vol. 169, no. 1, pp. 86–99, 2007.

[4] A. O'Hagan and F. J., *Kendall's advanced theory of statistics : Vol. 2b, bayesian inference*. Arnold, 2004.

[5] P. Blomstedt, R. Gauriot, N. Viitala, T. Reinikainen, and J. Corander, "Bayesian predictive modeling and comparison of oil samples", *Journal of Chemometrics*, vol. 28, no. 1, pp. 58–59, Nov. 2014.

[6] P. Wenig and O. Juergen, "Openchrom: A cross-platform open source software for the mass spectrometric analysis of chromatographic data", *BMC Bioinformatics*, 2010.

[7] S. J. Dixon, R. G. Brereton, H. A. Soini, M. V. Novotny, and D. J. Penn, "An automated method for peak detection and matching in large gas chromatography-mass spectrometry data sets", *J. Chemometrics*, vol. 20, pp. 325–340, 2006.

[8] T. Skov and R. Bro, "An automated method for baseline correction, peak finding and peak grouping in chromatographic data", *The Analyst*, vol. 138, pp. 3502–3511, 2013.

[9] G. Vivó-Truyols, J. Torres-Lapasió, A. van Nederkassel, Y. V. Heyden, and D. Massart, "Automatic program for peak detection and deconvolution of multi-overlapped chromatographic signals part I: Peak detection", *Journal of Chromatography*, pp. 133–145, 2005.

[10] M. Wand and M. Jones, *Kernel smoothing*, 1st ed. Chapman and Hall, 1995.

[11] R. Moroni, P. Blomstedt, L. Wilhem, T. Reinikainen, E. Sippola, and J. Corander, "Statistical modelling of measurement errors in gas chromatographic analyses of blood alcohol content", *Forensic Science International*, vol. 202, pp. 71–74, 2010.

[12] R Core Team, *R: A language and environment for statistical computing*, R Foundation for Statistical Computing, Vienna, Austria, 2016. [Online]. Available: `https://www.R-project.org/`.

[13] D. Eddelbuettel and R. Francois, "Rcpp: Seamless R and C++ integration", *Journal of Statistical Software*, vol. 40, no. 8, pp. 1–18, 2011.

[14] A. Felinger, *Data analysis and signal processing in chromatography*. Elsevier Ltd., 1998.

[15] J. J. Faraway, *Extending the linear model with r*. Chapman & Hall, 2006, p. 232.

[16] P. H. Eilers, "Parametric time warping", *Anal. Chem.*, vol. 76, pp. 404–411, 2004.

[17] F. Hoti and L. Holmström, "On the estimation error in binned local linear regression", *Nonparametric statistics*, vol. 15, no. 4-5, pp. 625–642, Aug. 2003.

[18] M. Wand and M. Jones, *Kernel smoothing*, 1st ed. Chapman and Hall, 1995, pp. 182–192.

[19] T. Hastie and C. Loader, "Local regression: Automatic kernel carpentry", *Statistical Science*, vol. 8, no. 2, 120:143, 1993.

[20] S. X. Chen, "Local linear smoothers using asymmetric kernels", *Ann. Inst. Statist. Math.*, vol. 54, no. 2, pp. 312–323, 2002.

[21] Y. Xia, "Asymptotic behavior of bandwidth selected by the cross-validation method for local polynomial fitting", *Journal of Multivariate Analysis*, vol. 83, pp. 265–287, 2002.

[22] B. A. Turlach and M. P. Wand, "Fast computation of auxiliary quantities in local polynomial regression", *Journal of Computational and Graphical Statistics*, vol. 5, no. 4, pp. 337–350, 1996.

[23] S.-J. Baek, A. Park, Y.-J. Ahn, and J. Choo, "Baseline correction using asymmetrically reweighted penalized least squares smoothing", *Analyst*, vol. 140, p. 250, 2015.

[24] A. Savitzky and M. J. E. Golay, "Smoothing and differentiation of data by simplified least squares procedures", *Anal. Chem.*, vol. 36, no. 8, pp. 1627–1639, 1964.

[25] signal developers, *signal: Signal processing*, 2014. [Online]. Available: `http://r-forge.r-project.org/projects/signal/`.

[26] J. Durbin and G. S. Watson, "Testing for serial correlation in least square regression: I", *Biometrika*, vol. 37, no. 3, pp. 409–428, 1950.

[27] G. Vivó-Truyols, J. Torres-Lapasió, A. van Nederkassel, Y. V. Heyden, and D. Massart, "Automatic program for peak detection and deconvolution of multi-overlapped chromatographic signals part II: Peak model and deconvolution algorithms", *Journal of Chromatography*, pp. 146–155, 2005.

[28] K. Lan and J. W. Jorgenson, "A hybrid of exponential and gaussian functions as simple model of asymmetric chromatographic peaks", *Journal of Chromatography*, vol. 915, pp. 1–13, 2001.

[29] J. P. Foley and J. G. Dorsey, "A review of the exponentially modified gaussian (EMG) function: Evaluation and subsequent calculation of universal data", *Journal of Chromatographic Science*, vol. 22, Jan. 1984.

[30] S. E. Stein, "An integrated method for spectrum extraction and compound identification from gas chromatography/mass spectrometry data", *Journal of the American Society for Mass Spectrometry*, vol. 10, no. 8, pp. 770–781, Aug. 1999.

[31] J. J. Moré, "The levenberg-marquardt algorithm: Implementation and theory", *Numerical Analysis*, vol. 630, pp. 105–116, 2006.

[32] S. G. Johnson, "The nlopt nonlinear-optimization package", *?*, vol. ?, no. ?, ?, ?

[33] P. Esseiva, L. Gaste, D. Alvarez, and F. Anglada, "Illicit drug profiling, reflection on statistical comparisons", *Forensic Science International*, vol. 207, no. 1-3, pp. 27–34, Apr. 2011.

[34] W. K. Härdle and L. Simar, *Applied multivariate statistical analysis*, 4th ed. Springer-Verlag Berlin Heidelberg, 2015, pp. 84–89.

[35] K. P. Murphy. (2007). Conjugate bayesian analysis of the gaussian distribution, [Online]. Available: `https://www.cs.ubc.ca/~murphyk/Papers/bayesGauss.pdf` (visited on 02/24/2017).

[36] K. M. Ramachandran and C. P. Tsokos, *Mathematical statistics with applications in r*, 2nd ed. Elsevier Ltd., 2015, pp. 734–744.

[37] R. E. Kass and A. E. Raftery, "Bayes factors", *Journal of the American Statistical Association*, pp. 773–795, 1993.

[38] P. Blomstedt and J. Corander, "Posterior predictive comparisons for the two-sample problem", *Communications in Statistics - Theory and Methods*, vol. 44, no. 2, pp. 376–389, Jun. 2013.

[39] J.-M. Marin and R. C. P., *Bayesian core: A practical approach to computational bayesian statistics*. Springer, 2007.

[40] G. Zadora, A. Martyna, D. Ramos, and C. Aitken, *Statistical analysis in forensic science*. John Wiley & Sons, Ltd, 2014.

[41] G. S., *Predictive inference: An introduction*. Chapman and Hall, 1993.

[42] Y. Qiu, S. Balan, M. Beall, M. Sauder, N. Okazaki, and T. Hahn, *Rcppnumerical: 'rcpp' integration for numerical computing libraries*, R package version 0.3-1, 2016. [Online]. Available: `https://CRAN.R-project.org/package=RcppNumerical`.

[43] C. Smith, E. Want, G. O'Maille, R. Abagyan, and G. Siuzdak, "XCMS: Processing mass spectrometry data for metabolite profiling using nonlinear peak alignment, matching and identification", *Analytical Chemistry*, 2006.

[44] C. D. Brown and H. T. Davis, "Receiver operating characteristics curvves and related decision measures: A tutorial", *Chemometrics and Intelligent Laboratory Systems*, vol. 80, no. 1, pp. 24–38, 2006.

[45] I. Borg and P. Groenen, *Modern multidimensional scaling*. Springer, 1997.

[46] K. P. Murphy, *Machine learning: A probabilistic perspective*. The MIT Press, 2012.

[47] M. Woldegebriel and G. Vivó-Truyols, "Probabilistic model for untargeted peak detection in LC-MS using bayesian statistics", *Analytical Chemistry*, 2015.