A thesis presented for the degree of

Doctor of Philosophy in Pharmaceutical and Biomedical Sciences

Of Impurities, Drugs and Metabolites – ToF-SIMS Application Studies for Use in the Pharmaceutical Sciences

by

Michael Markus Chrubasik

Supervisors:

Prof. Blair Johnston

Dr. Alison Nordon

Prof. Ian Gilmore

Strathclyde Institute of Pharmacy and Biomedical Sciences, University of Strathclyde, Glasgow
in collaboration with the National Physical Laboratory, Teddington
and the EPSRC Future Manufacturing Research Hub for Continuous Manufacturing and Crystallisation,
University of Strathclyde, Glasgow

December 2020

This thesis is the result of the author's original research. It has been composed by the author and has not been previously submitted for examination which has led to the award of a degree.

Signed:

Date: 15/12/2020

Acknowledgements

The author would like to thank the UK Engineering & Physical Sciences Research Council (EPSRC), the National Physical Laboratory (NPL), the University of Strathclyde and the EPSRC Future Manufacturing Hub for Continuous Manufacturing and Crystallisation (CMAC) (EP/P006965/1) for the continuous support and awarded studentship. This work was funded by the Academy of Medical Sciences (AMS) Springboard award (MRC, RW, RG, SBF001-1009) and Tenovus Scotland (MRC, S15/09).

I would like to take this opportunity to thank everyone who was part of this long journey, thank you..

- To Prof. Blair Johnston, for the many opportunities granted, guidance and enduring support as my mentor and first supervisor.
- To Dr. Alison Nordon, for fruitful discussions and help with spectral data and PLS, and for taking the position as my second supervisor.
- To Prof. Ian Gilmore, for his input and guidance related to ToF-SIMS and for taking the position as my NPL supervisor.
- To my collaborators Dr. Sara Ottoboni and the Price Research group, for our collaboration on paracetamol impurities.
- To my collaborators Dr. Rachel Wood, Rebecca Gilchrist, Ben Veerman and Dr. Margaret Cunningham, for our collaboration on human coronary artery endothelial cells.
- To my collaborators Laia Castano Espriu and Dr. Katherine Duncan, for our collaboration on agar-based bacteria.
- To all staff and colleagues at CMAC, for their patience, help and collegiality.
- To all colleagues at the data science department at NPL, for the warm welcome and for being so patient with me about this thesis business.
- To the many friends and acquaintances I've made along the way, for the joys and adventures, WEBs and FEBs, friendship and a feeling of belonging.

- To Aruna, Tony and Murray, for all your contributions and guidance throughout my time as a student, great help in taming this thesis and all the fantastic travels together.
- To my family, Isa, Lothar, Janusz, Gerhard, Liz and Julia, for all the love and support.
- And finally, to my partner Maria, for always having my back, your unwavering patience, kindness and support and without whom none of this would be possible. Thank you.

Abstract

This thesis aims to explore the use of Time-of-Flight Secondary Ion Mass Spectrometry (ToF-SIMS) for applications in the pharmaceutical sciences through three separate studies with different pharmaceutical applications.

The first study investigated the use of ToF-SIMS for the surface characterisation of pharmaceutical crystals using 4-nitrophenol and 4-acetamidobenzoic acid impurity incorporations in acetaminophen (paracetamol) crystals. A range of impurity deposition scenarios were examined to study the impurity intensity and distribution patterns as well as impurity-induced changes to the crystal surfaces.

In the second study, the characteristics of human coronary artery endothelial cells (HCAECs) before and after application of an anti-thrombotic drug were analysed using ToF-SIMS. A sample preparation method was developed to analyse the cells as well as identify and image the drug on the cell surface. Subsequently, untreated and treated cells were prepared using the trialled procedure to investigate the effects of the drug on the cells.

The final study explored the use of ToF-SIMS for bacterial analysis, in particular agar-based bacteria and the tracking of bacterial metabolites. In this scenario, various sample preparation and drying methods were trialled and the most successful method applied to attempt the identification and tracking of tetracycline, a known streptomyces metabolite and antibiotic, in the bacterial growth medium.

In addition to the ToF-SIMS experimental studies, a software tool for the selection of spectral pre-processing methods for NIR and other spectral applications was developed and assessed. The software toolbox enables a design-of-experiment-centred approach to selecting viable pre-processing methods to correct spectral data prior to further usage in applications such as regression modelling. Two data use-cases are presented that stem from the chemical and pharmaceutical sciences and demonstrate the applicability of the tool.

Contents

Chapter 1

Introduction

The pharmaceutical industry is a multi-billion US dollar business in which in 2019 the top 30 pharmaceutical companies made a combined global revenue of $770 billion[1]. However, the development of new drugs is linked to significant risks, upfront costs and time investments stemming among others from extensive research and development phases (R&D) as well as drug authorization requirements. Due to these factors, a new drug launch in 2019 was shown to cost upwards of $4.5 billion[1]. Between 2014 and 2018 only 87 drugs were approved on average per year by the European drug agency, again, largely limited by extensive drug approval and testing procedures[2].

In recent years, the pharmaceutical industry has shown a trend towards leaner enterprise models and precision medicine. This is mostly driven by an industry-wide push away from high-risk, high-cost, high-profit business models centred around 'blockbuster' drugs. To prepare for the future, many companies are trying to strengthen and bolster their R&D pipelines to be more dynamic and enable accelerated development methods in-line with the shift towards more personalised medicines[3,4]. Continuous development of products is crucial for the long-term survival of these companies and relying solely on the previous high-risk-reward business model is unsustainable[5]. The time it takes to release a drug from initial discovery to market currently averages 12 years[6]. It is hoped that the obstacles felt and foreseen in the markets can be anticipated and counteracted by embracing new technologies and heavily investing in research that will aid in accelerated drug-to-market times as well as the development of more bespoke medicines[3,5].

The introduction of modern analytical and high-throughput methodologies had a major impact on the development of new drugs and therapeutically active ingredients, aiding scientists in the field to screen for more potential medicines while also enabling the analysis of potential drug candidates in more intricate ways[7]. In the last two decades, mass spectrometry imaging, has proven to be a very useful and valuable analytical technique for drug discovery and development with applications ranging from drug and analyte distribution research[8,9], toxicological and drug delivery studies[10,11] and analysis of pharmaceutical materials and drug products[12,13]. Matrix assisted laser desorption ionisation (MALDI) and desorption electrospray ionisation (DESI) have been commonly found among the MSI techniques applied for these purposes. However, in recent years, time-of-flight secondary ion mass spectrometry (ToF-SIMS) has seen an increased uptake in pharmaceutical applications, in particular in biological contexts. This was largely driven by technical improvements of the ion sources allowing for polyatomic ion beams which cause less surface damage and thus aid the analysis of biological material[14,15]. ToF-SIMS is a powerful analytical technique for investigating elemental and molecular variations of samples across surfaces and through sub-surface layers. The main advantages of ToF-SIMS in comparison to the other stated MSI techniques are increased spatial resolution, allowing the analysis of unmodified samples and 3D-imaging capabilities[16]. While the use of ToF-SIMS has seen an increased interest, the number of publications has been limited in contrast to other mass spectrometry techniques such as MALDI or DESI. To extend this limited body of knowledge this thesis aims to investigate useful avenues of application of ToF-SIMS in the field of pharmaceutical sciences. This is achieved by exploring three case studies:

- Pharmaceutical material characterisation through the investigation of paracetamol crystal surfaces with three distinct impurity loadings,

- Analysis of drug localisation in cells to support pharmacological studies that examine suspected off-target effects of a drug in human coronary artery endothelial cells,

- Method development for the analysis of bacteria and their growth media for metabolite identification, including samples preparation and measurement optimisation.

An important aspect to most spectroscopic and spectrometric analyses, especially in the context of quantitative analyses and regression modelling, is the pre- and post-processing of data. Finally, in addition to work pertaining to ToF-SIMS, this thesis also presents a software tool to aid in the pre-processing method selection for near-infrared and other spectroscopy data. The thesis is split into the following chapters:

- Chapter 2 will introduce mass spectrometry and time-of-flight secondary ion mass spectrometry as well as the instrument used for experimental measurements in many of the further chapters, the IONTOF TOF.SIMS 5.
- Chapter 3 will present some background to spectral pre-processing as well as a software tool to aid in the pre-processing method selection for near-infrared and other spectroscopy data. This chapter will introduce statistical methods used in some of the ToF-SIMS related chapters.
- Chapter 4 will explore the usage of ToF-SIMS for pharmaceutical material characterisation in which paracetamol and paracetamol related impurity crystals and their surfaces are analysed.
- Chapter 5 will introduce a sample preparation and analysis method for the analysis of human coronary artery endothelial cells and employ this method to investigate changes to these cells prior to and after application of an anti-thrombotic drug.
- Chapter 6 probes sample preparation methods for agar-based bacterial samples for analysis by ToF-SIMS and trials the most promising sample preparation method with a test-case.
- Lastly, Chapter 7 summarises the previous chapters and presents an outlook based on the results shown.

Chapter 2

Background to Time-of-Flight Secondary Ion Mass Spectrometry

2.1 Introduction to Mass Spectrometry

Mass spectrometry (MS) is the field of study dealing with the analysis of samples based on the separation of their atomic and molecular constituent masses. MS is used to identify, quantify and examine analytes using the mass-to-charge ratio (m/z) of ions that have been produced from samples[17]. To date, a wide variety of mass spectrometry analysis techniques has been developed but all methods follow a basic experimental structure[18]:

- Introduction of a sample
- Ionisation of the analytes
- Separation of ions (mass analyser)
- Detection of separated ions
- Processing and interpretation of data

An ionisation step is required to produce ions from neutral atoms and molecules where ions are produced by loss or gain of a charge from their neutral precursors. The technique applied is dependent on the state of the sample in which the analyte molecules are available. These can range from gaseous samples and liquid or solution samples to solid samples[18]. The level of sample preparation required varies depending on the mass spectrometry system employed and analytes of interest to be measured.

In addition to the different methods for analyte ionisation, a further distinction can be made by the amount of fragmentation produced during ionisation, with a higher degree representing a larger amount of ion fragments from the parent molecular ion to be observed[18,19]. Techniques with high amounts of fragmentation are called hard ionisation methods, while techniques with low fragmentation rates are termed soft ionisation methods. Once ions are formed, they are introduced into a mass analyser. Here they are separated based on their mass-to-charge ratio (m/z), where mass is the molecular weight of the ion and charge is the number of electric charges present on the ion. Some mass analysers are equipped with collision cell modules that allow for further fragmentation of the parent ions. This technique is termed tandem mass spectrometry or (MS/MS) which has applications in many different areas, including the analysis of large biomolecules[19,20].

After their separation, ions are directed towards a detector that creates the signal, which is then translated into a mass spectrum, plotting the intensity of the signal from the detected ions versus their m/z[18,19]. The quality of a spectrum is typically measured in terms of mass resolution and mass accuracy obtained. Mass accuracy describes the difference between the measured and the true mass of an ion and is dependent on the resolution of the mass analyser. Mass resolution on the other hand, is defined as $\mathrm{m}/\Delta\mathrm{m}$, where m is the mass (here m/z) and Δm is commonly measured as the peak width assessed at full-width half-maximum (FWHM). Δm can also be measured as the space, or valley, between two peaks of equal intensity[18,19].

Various forms of mass spectrometers are available that can be chosen depending on the desired properties, from ionisation methods to analyser types as well as sensitivity, accuracy and mass resolution. The “right” instrument depends on the type of samples and analyses necessary. Many mass spectrometers can be coupled to other instruments and can include interchangeable ion sources or multiple analyser types.

2.2 Time-of-Flight Secondary Ion Mass Spectrometry

2.2.1 Introduction

Time-of-Flight Secondary Ion Mass Spectrometry (ToF-SIMS) is a surface analytical ultra-high vacuum technique that has found extensive application in material sciences, earth sciences and, particularly within the last decade, biosciences[39,41]. It enables the mass spectrometric analysis of the topmost atomic or molecular layers of surfaces (1-2 nm depth), bulk materials and material interfaces with up to 100 nm lateral resolution. ToF-SIMS is also used for the identification and imaging of molecular and elemental distributions in organic and inorganic samples. Analyses can typically be carried out without significant sample preparation, unless samples are likely to be affected by the ultra-high vacuum or require additional treatment to enhance ionisation. Among many others, application examples of the technique can be found in many fields such as the pharmaceutical industry, where Furudate et al. used ToF-SIMS to monitor and better understand the process of granulation by observing the amount of a binder molecule on the surface of granules[21]; the material sciences, where Tortora et al. applied ToF-SIMS to study paint artifacts from the 17th century, trying to identify the components used to achieve certain colours[22]; and bio-interfaces, where Perkins et al. employed ToF-SIMS to distinguish and map herbicides within leaf surfaces as well as study their behaviour[23].

Advances in cluster ion beam techniques in the last two decades have significantly improved the sensitivity of depth profiling, particularly in the field of biomaterial research, and have enabled depth resolutions of up to 5 nm, thus allowing more delicate probing of sample layers[26,31,33,39,41]. There are a number of limitations that can make ToF-SIMS less suitable than similar other mass spectrometry imaging techniques for specific scenarios. Even though SIMS-based instruments render themselves great candidates for qualitative surface assessments, using them for quantitative analysis is very difficult due to a number of reasons such as sample variability and matrix and charging effects[24]. For quantitative analyses, the addition of internal calibration standards that can be applied consistently and do not influence the sample or sample surface is necessary. Depending on the type of sample, method development can be complex and time consuming, particularly for samples that need modification to withstand the high vacuum. Biological samples, for example, need extensive method development in the preparation stages to make them vacuum-compatible and optimise signal responses[41]. Furthermore, the different ionisation potentials of elements and molecules can make the multicomponent and bulk analysis of materials difficult. In addition, post-measurement data processing of the results can be very complex, with the consequence that other faster, more accurate, and more high-throughput techniques are often preferred[41]. Further details of these phenomena and challenges will be presented in the following chapters.

2.2.2 Secondary Ion Mass Spectrometry: Principles

In SIMS, sample surfaces are bombarded using a highly energetic primary ion beam (such as Cs^+, Au^+ and Ga^+). The impact of these ions with the surface and underlying layers results in the charge transfer between the primary ions and the elements and molecules at the sample surface, creating both positive and negative secondary ions. The ions are ejected from the surface into the ultra-high-vacuum (UHV) and extracted into a time-of-flight mass spectrometer for separation and detection[34,43]. It has to be noted that a only a small fraction of the species become ionized and a large percentage of the surface-ejected species are neutral molecules and elements[25]. There are various theories that try to explain this process of ion bombardment and material release, which is also known as sputtering. The most widely shared explanation is the linear cascade theory[26,27].

Linear cascade theory describes the process of a monoatomic, high-energy primary ion impacting and interacting with a target sample surface and thus releasing its energy to atoms in the top-most monolayers through inelastic collisions[26,27]. This process triggers a collision cascade (Figure 2.1) of other atoms, imparting excess energy to neighbouring atoms through further collisions, thus resulting in the release of atoms, fragments and molecules from the surface. These can be released as neutrals and secondary ions[26,27] Sputtered surface material can undergo a number of ionisation pathways such as ionisation due to collision with highly energetic material, heterogenous bond cleavage or cationisation of neutrals via the attachment of small ions (e.g. H^+) or metal ions. The latter effect is frequently observed with sodium or potassium as these are common constituents, impurities or contaminants in many materials[28,29].

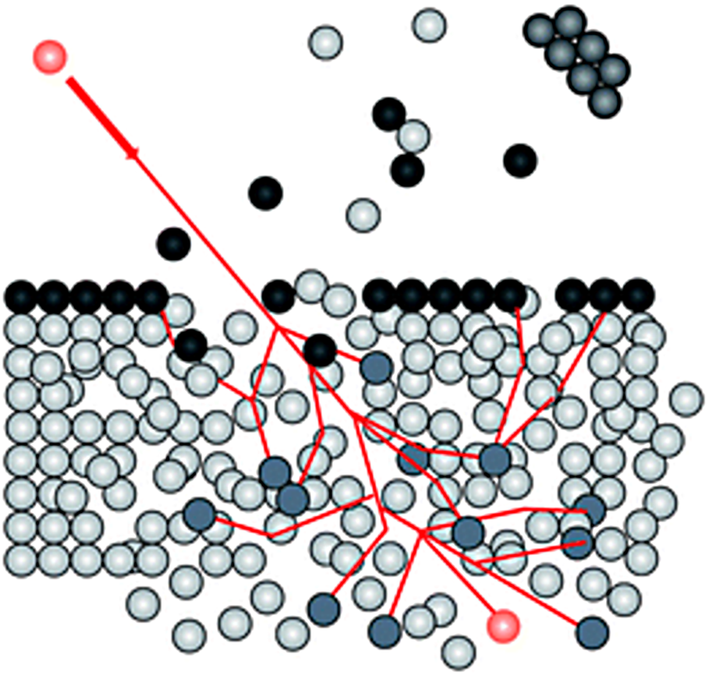


Figure 2.1: A primary ion beam hits a sample surface resulting in collision cascade mixing and the generation of secondary particles (such as atomic and cluster ions, molecules and atoms). Figure obtained from Senoner et al.[30].

Older-generation SIMS instruments typically used primary ions such as Cs^+, Au^+ and Ga^+, with high impact energies of up to 30 keV. The impact of monoatomic primary ions on surfaces can be described well for these using the linear cascade theory[26,27,31]. However, a major shift towards the use of cluster ion species (e.g. $Bi_n{}^+$, $C_{60}{}^+$), particularly in biological applications of SIMS, has been observed in the last two decades. It has been found that the use of cluster ion beams enhances secondary ion (SI) yields by multiple orders of magnitude and produces secondary ions with less fragmentation[32].

This is particularly desirable when trying to identify larger molecules and structures directly, where the identification of ion peaks in a spectrum, after strong fragmentation of the sample, would be a particular challenge, even with lengthy post-processing procedures.

Atomic primary ions penetrate surfaces much deeper than their cluster counterparts resulting in a collision cascade occurring further below the surface, thus reducing the amount of sputtered material and causing substantial damage at and around the site of impact. Cluster ions, on the other hand, are thought to transfer their energy much closer to the surface region of samples causing wider and more surface-focused impact craters, hence significantly increasing the sputter yield of secondary ions produced. Furthermore, because these ions do not penetrate the surface as deeply, sub-surface layers of the sample are less affected, severely reducing the damage to the surface through continued analysis of a similar area or depth profiling[15,32,33]. This effect can be observed viewing the molecular dynamic simulations by Postawa et al. in Figure 2.2[34].

Garrison, et al. present and compare molecular dynamic simulation results for atomic and cluster ion beams as they strike a surface (Figure 2.2)[35]. The simulations clearly demonstrate the differences in impact depth and width between the two beams. The authors compare the atomic primary ion interactions with sample surfaces to a "game of billiards", very much in-line with the linear cascade theory. The amount of primary ion interactions with the surface are concentrated in a smaller area and reach further into the bulk. On the other hand, cluster ion interactions of large clusters (¿1,000) are compared to a "washing mechanism" where the cluster beam strikes a surface and pushes the surface molecules in its path to the sides and out of the impact path, thus generating a larger crater with more surface-based sputter damage that does not impact the bulk as much as a high-energy atomic primary ion beam.

Static versus Dynamic SIMS

There are two distinct methods to perform SIMS analysis: in a static or a dynamic regime. Static SIMS is a mode of analysis applied when the topmost monolayer of

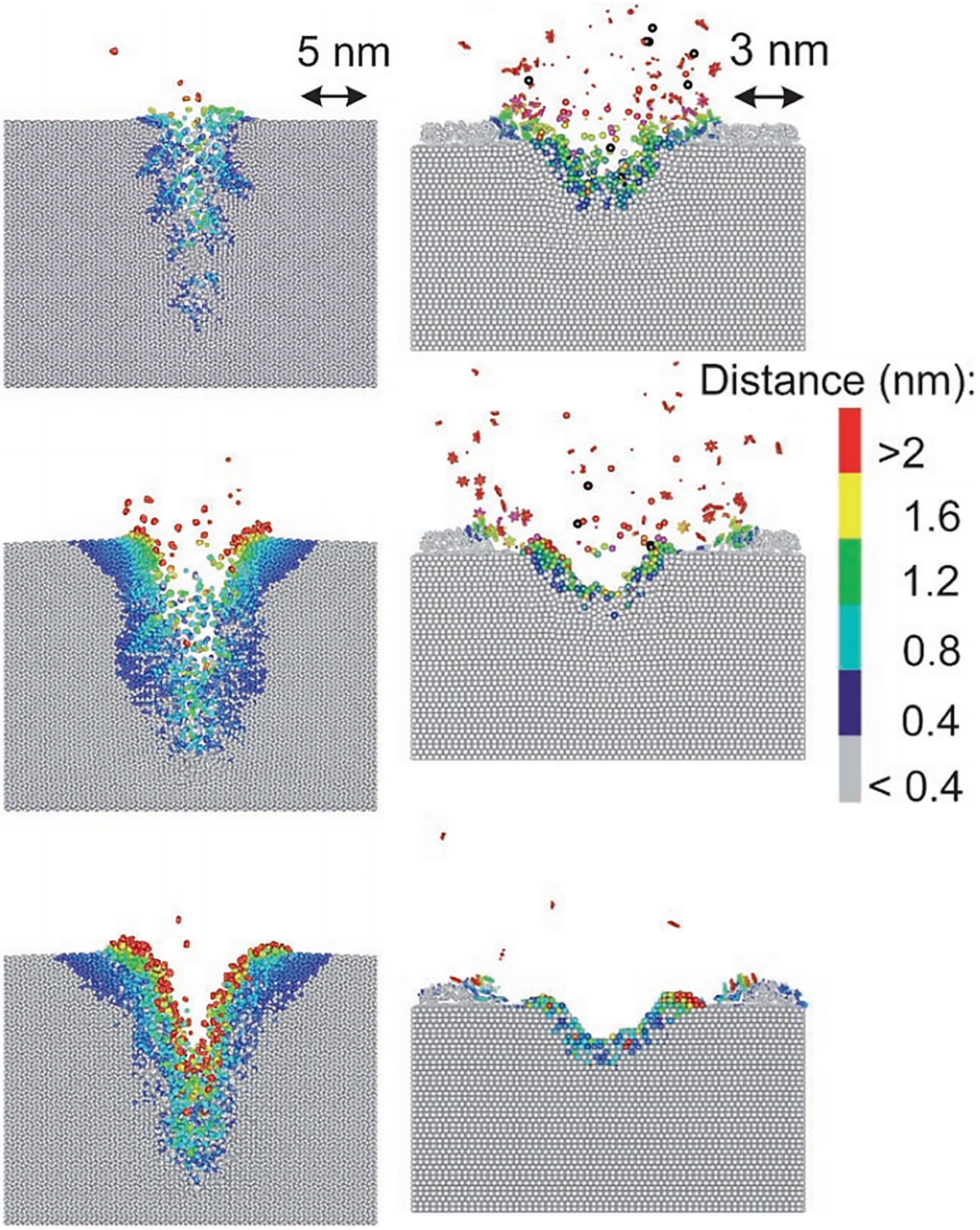


Figure 2.2: 1.5 nm slice comparison between a 15 keV C_{60} (left, time spacing between images from top to bottom: 1 ps, 3 ps, 26 ps) and a 5 keV Au (right, time spacing between images from top to bottom: 1 ps, 4 ps, 36 ps) bombardment of benzene on Ag. The colouring is based on the displacement amount of particles from the original position[35].

a sample is of interest. It allows for the extraction of molecular information of said layer and enables characterisation and exploration of the sample surface. Static SIMS is achieved by limiting the ion beam energy and not exposing the surface to a primary ion dose above 10^{12} ions/cm^{2}[36]. With such limits in place, the damage to the surface is kept to a minimum, making SIMS a surface-sensitive technique. The idea of a static limit was established by Benninghoven in 1969[37] and first systematically tested by Briggs in 1989[38]. In dynamic SIMS, on the other hand, the sample is eroded using a high energy beam continually bombarding the sample. This mode of analysis yields information from the "bulk" of the material and usually causes more severe fragmentation of the surface molecules but in return can provide significantly more information about the sample[36].

General SIMS Equations

A measure of this process is the sputter yield (Y). It can be defined as the amount of material released from the surface per primary ions. The number of ionised atoms, molecules and fragments is significantly lower than the total released material and can vary between 0.0001 and 10%[25]. This sputter yield can be related to the number of secondary ions using the basic SIMS equation (Equation 2.1):

$$I_m = I_p Y_m \alpha \sigma_m \eta \tag{2.1}$$

where

$\mathrm{Y_m}$ is the sputter yield of sputtered particles of a species with mass m,

$\mathrm{I_m}$ is the current of secondary ions with mass m,

$\mathrm{I_p}$ is the primary particle current,

α is the ionisation probability,

σ_m is the fractional concentration of m in the surface layer and

η is the transmission of the analysis system.

It is important to note that the sputter yield, $\mathrm{Y_m}$, includes all sputtered particles of m

- neutral and ionic - and together with the ionisation probability is dependent on the primary ion beam energy and the type of molecule that is being ionised.

In addition to the primary ion source and the material to be sputtered, the sample matrix plays a significant role in the sputter- and secondary ion-yields. As ionisation is thought to occur very close to the sample surface, the electronegativity of the species present on and in the surface material can influence the ionisation probability of different fragments in the same sputtered material[39]. This phenomenon can make sample analysis very difficult and can be the reason why certain secondary ions are particularly enhanced or completely surpressed within resulting mass spectra, thus causing significant issues with quantitative analysis in SIMS[39,40].

2.2.3 Instrumentation

IONTOF TOF.SIMS 5

The work detailed here has been carried out on a TOF.SIMS 5 instrument (IONTOF GmBH, Munster, Germany) at the CMAC Future Manufacturing Research Hub. A typical schematic of the instrument is depicted in Figure 2.3.

Primary ions are produced within the primary ion column where they are accelerated towards the target surface. Inside the primary ion column, ions are generated in the ion gun from where they pass through a pulsing device and focussing lenses to become bunched in time and focused. These steps facilitate the near-simultaneous (approximately 1ns) arrival of hundreds of primary ions at the sample area of interest. The secondary ions that are generated from the surface are then exposed to an extraction voltage (± 2000 V depending on the polarity selected), accelerating them into the time-of-flight mass analyser[33]. In this section, the ions are separated according to their mass-to-charge ratio (*m/z*) and detected, thus generating a mass spectrum.

Analyzer

To enable the analysis and detection of secondary ions, SIMS instruments are equipped

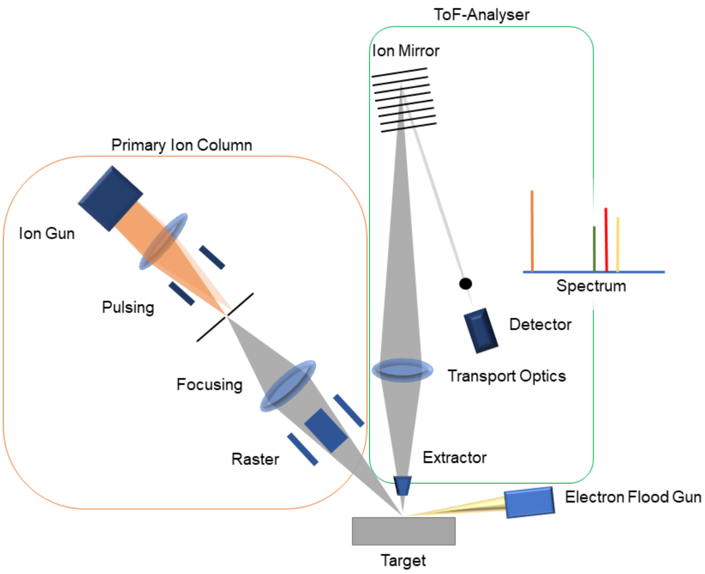


Figure 2.3: IONTOF TOF.SIMS 5 schematic (Provided with instrument by IONTOF, adjusted for own purposes)

with a mass analyser through which secondary ions of different m/z are separated before being propelled towards a detector. There are four main types of mass analysers for ionised species[41]:

- Quadrupole mass analyser
- Ion trap mass analyser
- Fourier-transform ion cyclotron mass analyser
- Time-of-flight mass analyser

Modern SIMS instruments are commonly fitted with Time-of-Flight (ToF) mass analysers, but other types can be found in use as well[26]. The TOF.SIMS 5 is equipped with a time-of-flight analyser and as such will be the only one subject to further in-detail explanation. Capabilities and usage of ToF mass analysers for SIMS were first described by Chait and Standing in 1981[42]. Unlike most other mass analysers, ToF analysers have the capability of detecting at high mass resolution of around 10,000 while also having the ability to simultaneously detect secondary ions over a large mass range[26]. A number of analysers are able to achieve significantly higher mass resolutions but typically come with other disadvantages such as reduced speed or lower lateral resolution.

In the ToF analyser, secondary ions are separated according to their different mass-to-charge ratios (m/z). This is achieved by applying a common accelerating voltage during extraction with which the ions enter a field-free zone and travel a set distance to the detector[43]. Ions of different masses but with the same charge, and given the same kinetic energy, will travel at a different velocity through the flight tube and will therefore arrive at the detector at different times, the lighter ions travelling faster than the heavier ions. This can be described through equation 2.2:

$$E = zU = (mv^2/2) = m(L^2/2t^2) \tag{2.2}$$

where
E is the energy of the secondary ion,
z is the ion charge,
U is the extractor potential,
m is the mass of the secondary ion,
L is the distance travelled through the mass analyser flight tube,
t is the flight time through the mass analyser tube,
and v the velocity of the secondary ion.

Given that the time and distance travelled through the analyser flight tube and the extraction voltage are known, it is possible to deduce the mass-to-charge ratio of the detected ions. The mass resolution is the measure of the ability to separate minor differences in ion energy. Factors that influence mass resolution include the time of ion formation, the distance of the ions from the extraction field and the energy spread of ions of equal mass[43].

The more accurately the time of ionisation can be determined the better the time of flight can be assessed, which is why for high mass resolution, very short high-energy beams are preferred. In the TOF.SIMS 5 this is achieved by “bunching” the primary ions together, creating a high-density ion beam. Due to this bunching process, however, the primary ions arriving at the sample have a larger breadth of kinetic energies. Such a beam is more difficult to focus, thus reducing the spatial resolution but greatly improving mass resolution through higher ion density and beam intensity. On the other hand, increasing the pulse width (i.e. not bunching the primary ions together) will increase the ability to focus the primary ion beam, in turn increasing spatial resolution at the cost of mass resolution[41].

The distance of ions from the extraction field is dependent on the sample topography. To achieve an evenly distributed location of sample surface ions, the surface of a sample should be as flat as possible. A rough sample topography can result in the broadening of mass-to-charge signals and a reduced mass resolution[43–45]. The initial energy and velocity distributions from all secondary ions are counteracted by use of a one-stage ion mirror (or reflectron) which can be found towards the top of the mass analyser assembly. The accelerated ions are “cushioned” by an increasing potential that compensates for any differences in ion velocity or distance travelled. Faster ions penetrate deeper into the ion-mirror than their slower counterparts resulting in a bunching effect and thus allowing for ions of equal *m/z* to arrive at the detector at the same time[41,43,46].

The ToF analyser in the TOF.SIMS 5 includes a single stage reflectron (see Figure 2.4), which effectively extends the length of the flight tube, improving ion separation. Secondary ions are extracted from the stage, focussed and directed into the flight tube. Subsequently they hit the potential grid of the reflectron, which acts like a mirror, directing ions towards the microchannel plate detector.

2.2.4 Ion Sources

Liquid Metal Ion Gun

A liquid metal ion gun (LMIG) is frequently used as a primary ion source for SIMS due to its small beam spot size down to 10 nm and high ionisation yield, giving it excellent brightness and spatial resolution capabilities of sub-100 nm[32,41,43]. The LMIG works

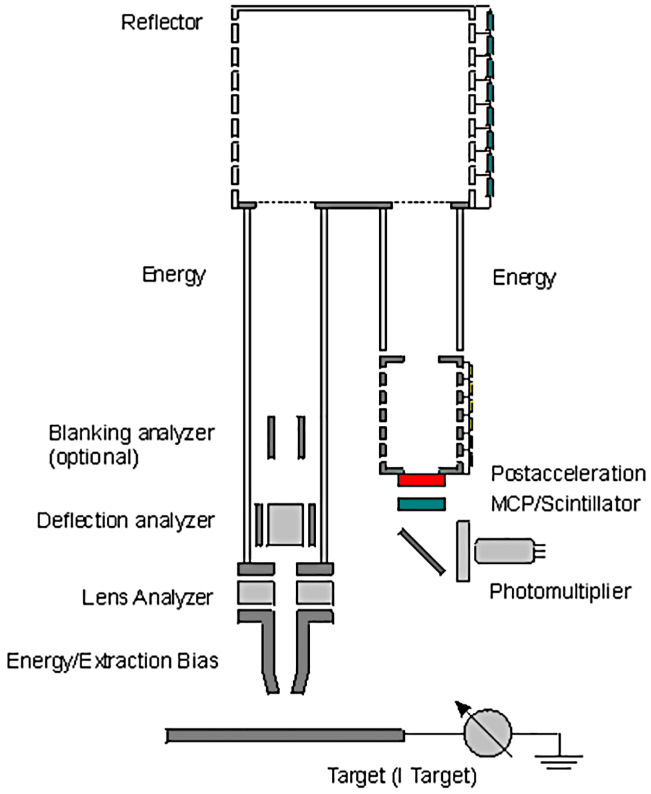


Figure 2.4: TOF.SIMS 5 mass analyser schematic taken from the instrument manual.

by continuously drawing heated liquid metal along a needle via a high-energy electric extraction field which ionises and extracts the metal[41,43]. During this process, atomic and cluster ion species can be formed. With the correct adjustments to the primary ion beam column, i.e. adjustments to the mass filtering via the pre-chopper and chopper, the LMIGs output can be filtered and adjusted to produce a chosen primary ion output; atomic or cluster ions[15,43]. A typical schematic can be seen in Figure 2.5 that shows this setup.

Examples of metal sources for LMIGs include Al, Ga, In and Bi. In addition, when ion gun sources are based on metal alloys (e.g. BiMn alloys), heating them to higher temperatures allows for the release of their higher-temperature alloy components thus enabling multiple ion beams (e.g. Bi^+, Mn^+). The TOF.SIMS 5 is equipped with a BiMn LMIG which can produce both Bi and Mn primary ion beams.

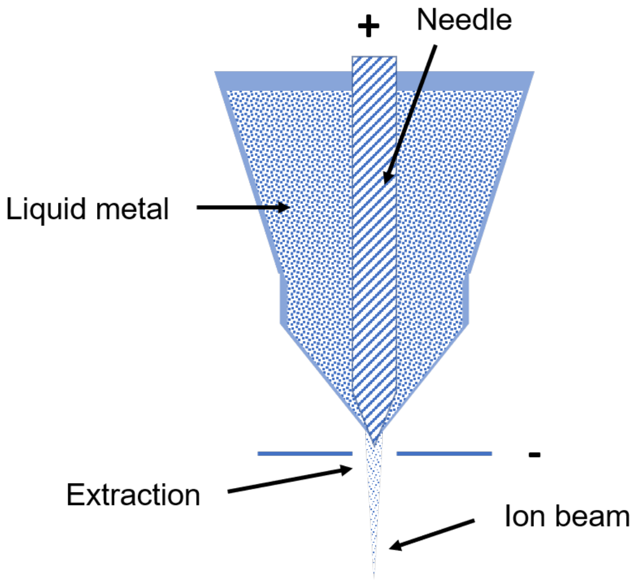


Figure 2.5: Liquid metal ion gun schematic adapted from the instrument manual.

Gas Cluster Ion Beams

Gas Cluster Ion Beams (GCIB) are an alternative ion beam source based on larger ion clusters, such as $C_{60}{}^{+}$ or argon clusters. They are predominantly used for sputtering larger amounts of material away from surfaces in conjunction with another primary ion source for analysis and thus their application is mostly found in depth profiling and 3D imaging[44]. Used as a primary ion source, GCIBs provide a softer method for ionisation (less fragmentation) and can therefore be used for the analysis of larger organic molecules, however this comes at a price of poorer mass resolution and accuracy and a reduction in secondary ion yield[47–49].

Sources Available on the TOF.SIMS 5

The TOF.SIMS 5 model is equipped with a variety of ion sources: a BiMn liquid metal ion gun (LMIG) producing both bismuth atomic- and bismuth cluster-ions, a caesium source, an oxygen source and an argon cluster source. This diversity of primary ion sources enables the analysis of a wide range of samples without any drawbacks of having to choose an inadequate ion source due to non-availability. Table 2.1 describes the ion sources available on the TOF.SIMS 5 and the type of samples each one is recommended for. In this thesis only organic materials were analysed and as such only the bismuth and argon cluster sources were used. Both oxygen and caesium sources are too hard for usage with organic material and do not produce the desired secondary ion yield for such samples.

Table 2.1: Ion sources available on the IONTOF TOF.SIMS 5 at CMAC.

Ion Sources	
Bismuth: Bi^+ $Bi_3{}^+$ $Bi_3{}^{2+}$	Inorganic and organic samples
Caesium: Cs^+	Inorganic samples (electronegative)
Oxygen: $O_2{}^+$	Inorganic samples (electropositive)
Argon clusters: $Ar_{1500}{}^+$ to $Ar_{3000}{}^+$	Organic samples

Flood Gun

Some samples may experience a charge build-up on the surface, known as “charging”. The primary ion beam is a charged particle beam that affects the target surface and causes surface charging via the release of secondary electrons as well as charge transfer to the surface. This can result in local deviations in the extraction voltage required to pull the secondary ions into the analyser, with the consequence being peak broadening and peak shifting, thereby reducing mass accuracy and resolution in the resulting mass spectrum[50]. To compensate for this charging effect, electrons of low energy are applied to the sample area using an electron flood gun to counteract the charge when no ion extraction is taking place[43]. In addition, changing the surface potential of the sample holder can also be used to compensate for the sample charging effects, instead of or as well as the flood gun. Charging effects usually occur on insulating samples, such as plastics or rubber, and may need to be adjusted prior to each measurement taken for samples with a rough or varied surface, as these surfaces exacerbate the effects of charging.

Operational Modes

The TOF.SIMS 5 instrument offers four operational modes:

- Spectrometry analysis
- Surface imaging
- Depth profiling
- 3D-imaging

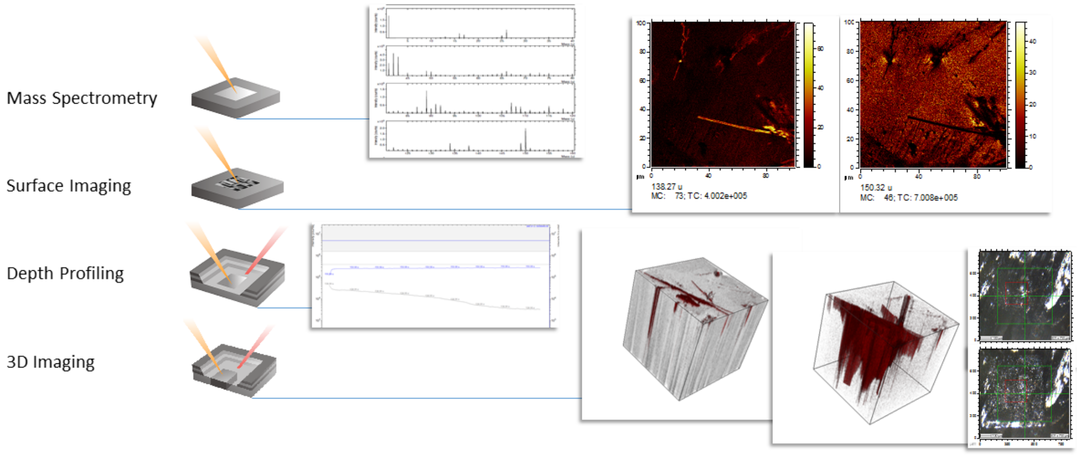
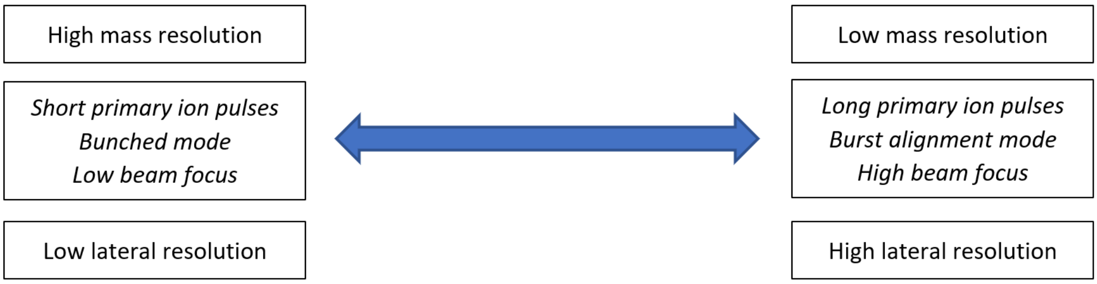


Figure 2.6: Operational Modes of the TOF.SIMS 5 instrument.

The different modes are able to interrogate samples in various ways, from surface through bulk as can be seen in Figure 2.6. ToF-SIMS imaging is achieved by rastering the primary ion beam across the sample, collecting a mass spectrum at each raster point and forming a pixelized matrix of measured data. Due to physical limitations it is possible to achieve either very high mass resolution, with up to 0.001 u separation, or very high spatial resolution with a resolution of up to 100 nm and only achieving nominal mass resolution[51], see Figure 2.7.

Using a hybrid mode called ‘delayed extraction’, it is possible to get a good balance between high lateral and good mass resolution at the same time. The mode combines a focused long-pulse primary ion beam with high spectral resolution and is facilitated by delaying the extraction of secondary ions after the primary ion pulse impact which compensates for the ion energy spread of the long-pulse ion beam[31,51]. A spectral comparison between these modes is provided by Claus et. al[51] and can be seen in

Figure 2.8.



High mass resolution

Short primary ion pulses
Bunched mode
Low beam focus

Low lateral resolution

Low mass resolution

Long primary ion pulses
Burst alignment mode
High beam focus

High lateral resolution

Figure 2.7: Figure depicting trade-off between short primary ion pulses with bunched mode and low beam focus versus long primary ion pulses with burst alignment mode and high beam focus.

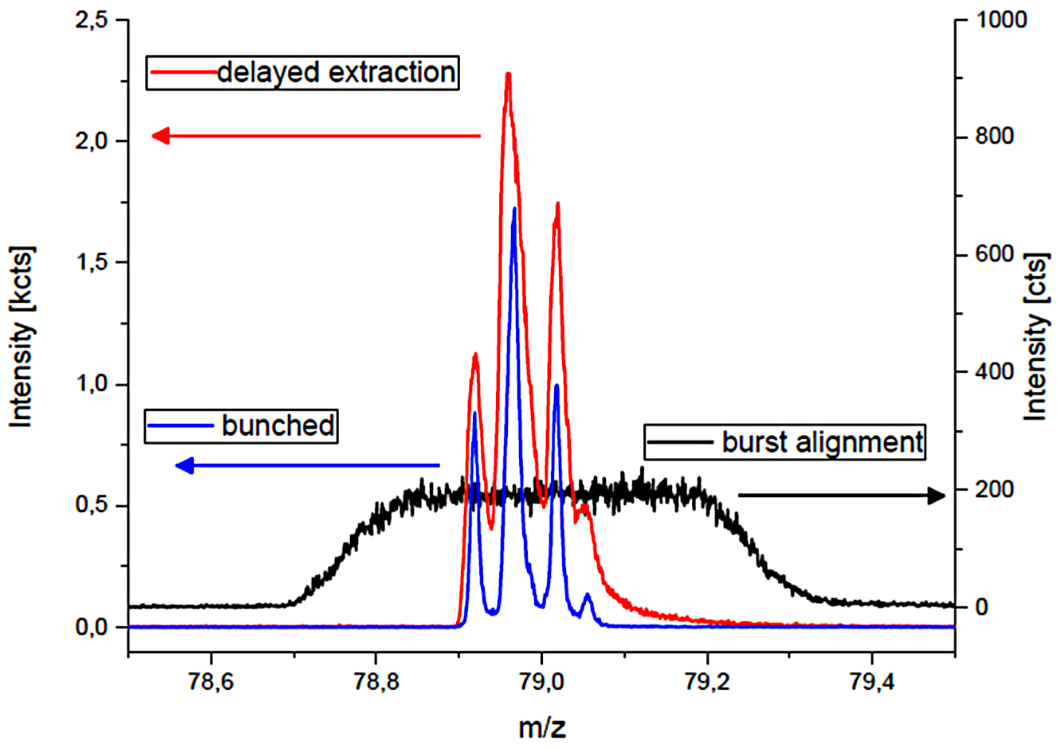


Figure 2.8: Spectral comparison between ”bunched” mode (blue), ”delayed extraction” (red) and ”burst alignment” (black, right scale). Figure used originates from supplementary information from Claus et. al.[51].

There are two major drawbacks to using delayed extraction; firstly, low molecular weight particles below *m/z* 15 cannot be detected as they escape the extraction field too quickly and are therefore lost. Secondly, the extraction delay causes an energy spread in the extracted secondary ions. This effect can be adjusted for to a certain extent, but ultimately limits the mass resolution to approximately 5,000.

A problem generally faced in the imaging of organic materials is the generation of sufficient molecular signal. Even though it is possible to get up to 100 nm resolution, achieving an adequate yield of secondary ions is often the limiting factor in organic mass spectrometry imaging using ToF-SIMS[33,46].

Depth profiling is achieved by applying a ‘dual beam technique’, continuously switching between surface analysis mode (typically using a liquid metal ion gun (LMIG)) and ion beam sputter mode (typically using a $\mathrm{Ar_{1500}}^+$ or $\mathrm{C_{60}}^+$ cluster source). Using this technique, the sample layers can be analysed using the LMIG and then sputtered away with the argon cluster beam, layer-by-layer, creating a depth profile. Using spectrometry mode, the analysis times are kept short and a high mass resolution depth profile is generated. Using analysis beam settings more catered towards high resolution imaging, a 3D image of the sample can be created with high lateral resolution but limited mass resolution in the process.

The caveats of depth profiling and 3D imaging are the very long measurement times depending on the sample depth and depth resolution required. Furthermore, the sputter source used during analysis is required to output an ion dose of at least two orders of magnitude higher than the primary source to be able to remove any damage caused by the primary ion surface analysis[44].

2.2.5 Other Mass Spectrometry Imaging Techniques (MSI)

ToF-SIMS is one of many mass spectrometry imaging techniques in use. This subchapter presents a number of other common mass spectrometry techniques that can be used for imaging. In line with the sample types that this thesis considers, particular focus is placed on the analysis of biological and/or pharmaceutical samples.

Nano-SIMS

In Nano-SIMS, reactive primary ion beams are used for the analysis of samples. Here, Cs^+ and O^- are employed for negative and positive secondary ion generation, respectively. In contrast to typical dynamic ToF-SIMS scenarios, Nano-SIMS offers simultaneously high sensitivity, high mass resolution as well as high lateral resolution[52,53]. The caveats of this technique are a significantly restricted number of secondary ions that can be observed concurrently (i.e. below 10) and the strong fragmentation of molecules in the sample that occurs under the dynamic conditions of the analysis. It is common to use heavy and isotopically-labelled molecules to improve traceability of specific molecules and bioprocesses[52,53].

OrbiSIMS

The OrbiSIMS is a hybrid apparatus that combines high mass resolution (upwards of 240,000 at *m/z* 200) and tandem MS (MS/MS) capabilities of Orbitrap instruments with the high spatial resolution available in SIMS instrumentation. The instrument was developed with bio-applications in mind and enables metabolic imaging with high lateral and mass resolutions. As the name suggests, the hybrid system has two analysers; the traditional ToF analyser and an Orbitrap analyser, which share the same extraction optics and thus can analyse the same point of interest of sample surfaces. Two ion beam sources can be utilised, either a Bi^+ LMIG source or an $Ar_n{}^+$ GCIB source. This kind of instrument currently exemplifies the most advanced of mass spectrometry imaging instruments due to its capabilities but comes with a high cost, limited availability and long measurement and analysis times[54].

Matrix-Assisted Laser Desorption Ionisation Mass Spectrometry

Matrix-Assisted Laser Desorption Ionisation (MALDI) uses a light-absorbing sample matrix which is applied to the sample of interest. A laser is then used to irradiate the sample in order to desorb and ionise sample surface bound molecules[55,56]. The matrix-assists by absorbing the laser radiation, with matrix internal crystals desorbing from the surface and forming a plasma-like area above the sample that promotes ionisation of the upper sample layers. MALDI is known as a soft ionisation technique, producing less fragmentation while the lower mass ranges are often dominated by matrix-related peaks and therefore it is ideal for analysing larger biomolecules, such as intact proteins and lipids[26,55,56]. This is in stark contrast to the hard ionisation of the SIMS mechanism where molecules are much more likely to be split into smaller fragments, resulting in high mass limits of above m/z 100,000 for MALDI and below m/z 5,000 for SIMS[26]. MALDI is thus often the preferred method for ionising large intact proteins as well as the identification of unknown biomolecules.

Another major difference between the techniques can be observed in the lateral resolution. While modern ToF-SIMS instruments, not including Orbi- or Nano-SIMS, can reach lateral resolutions of sub-100 nm, most modern MALDI instruments can reach resolutions of less than 20 µm[57]. This resolution limit forms one of the key shortcomings of MALDI, compared to SIMS, together with the more intensive sample preparation of applying a matrix prior to sample analysis. More in-depth comparisons between MALDI and SIMS can be found in Vickerman[56] and Spraker et al.[58].

Desorption Electrospray Ionisation Mass Spectrometry

In Desorption Electrospray Ionisation (DESI), an ionised solvent spray is applied directly to a sample surface with the subsequent droplet-sample-surface interactions leading to the capture of analyte molecules in the charged droplets, therby transferring the charge from the droplet to the analyte creating ions. The droplets are then dissipated entailing a gas-phase ion extraction to the mass analyser. Unlike SIMS or MALDI, DESI allows for open-air, ambient sample surface imaging without prior sample preparation requirements making it a comparatively simple and versatile MSI technique[26,55]. DESI, like MALDI, is a soft ionisation technique that can target a wide range of mass-to-charge ratios up to m/z 25,000 with spatial resolutions around 200 µm[26,55,59].

MSI Technique Summary

The reported techniques offer varied and complementary attributes that can all be used for mass spectral inquiries and imaging. While DESI and MALDI are the most commonly used due to their ease-of-use and ability for compound identification, SIMS still offers the best spatial and depth resolution available. Advanced SIMS instruments, i.e. Nano-SIMS and OrbiSIMS, offer further specific advantages compared to typical ToF-SIMS usage scenarios but are highly specialised instrumentation which are more expensive and potentially require deeper knowledge of the sample systems at hand. The choice of instrument thus depends on the specific scientific question and the resources available. With regards to advanced SIMS instrumentation, the use of ToF-SIMS could in many scenarios be considered as a first preliminary analysis step prior to analysis with the more advanced and complex systems. ToF-SIMS can also be used in conjunction with other non-imaging techniques, to provide a deeper comparison and interrogation of the samples. For biological and/or organic samples in particular, techniques such as liquid chromatography coupled with tandem mass spectrometry (LC-MS/MS) can be used to provide high quality, high resolution accurate mass information that can be used to confirm or identify compounds in the sample. Given similar fragmentation patterns these can be directly compared to the SIMS imaging data which in turn can provide information about the compound distribution.

Chapter 3

Design-of-Experiment Based Toolbox for Comparing Pre-Processing Methods for Spectral Data

3.1 Introduction

The work described in this chapter was carried out as part of a larger collaboration with the information and communications technology group at the EPSRC Hub for Continuous Manufacturing and Crystallisation (ICT-CMAC). The work performed by ICT-CMAC aimed to measure the mixing of two solvents at steady state within a continuous capillary-centred static mixer using a near-infrared (NIR) - hyperspectral imaging (HSI) probe. This novel method permitted the building of regression models based on the mixing throughout the vessel that, based on the studied conditions, allowed the survey of mixing behaviours of different solutions. Pre-processing the data prior to modelling was required and, due to the large amount of spectra to be analysed, an automated chemometrics approach was sought after. No universal pre-processing method that can deal with the potential variation of samples and sampling scenarios could be determined requiring a procedure that automatically tests and suggests multiple pre-processing steps and methods. In this context a graphical user-interface MATLAB toolbox for comparing pre-processing methods for spectral data was developed that allows users to brute-force test their data using a number of standard pre-processing methods and rate them based on partial-least-squares regression (PLSR) models.

The pre-processing of data is an integral part of chemometric modelling and is frequently applied in scientific fields that make use of spectral techniques. Pre-processing refers to the cleaning and preparing of data prior to modelling or analysis by reducing unwanted effects from them. In the context of spectral measurements, examples of these effects include instrument noise and scattering. This results in lowering the unmodelled variability in the data, allowing subsequent modelling procedures to focus on the wanted features[60,61]. Regression techniques offer better results if the data used have been standardised and pre-processed. If not, noise effects, measurement deviations and systemic variations (e.g. light scattering) can impose on the model, resulting in poorer predictions[62]. Rinnan notes that while many pre-processing methods have been developed with near-infrared (NIR) spectroscopy in mind, further spectroscopic techniques such as Raman and IR have also gained from the development of these procedures[62]. This allows most tools built for any of these techniques to be applicable for all of them. Exploratory data analysis and chemometrics form the tip of the data processing procedure and from the pharmaceutical industry to food and material research, the pre-processing of data is a requirement rather than an option[60].

Surprisingly, given its importance, it is infrequent that data cleaning methods and the processing steps involved prior to modelling are discussed in depth, compared or optimised. Instead, a laissez-faire approach of trial-and-error is often used, based on previous knowledge as well as quick application testing. This can result in non-transferable and non-robust methods that are applicable for a single set of results but fail to capture a larger body of experiments[61,63]. Regulatory bodies and industry demand standardised, consistent and streamlined data for future applications, such as machine learning, and necessitate robust and optimised methods that treat, store and output data and results in a reliable and future-proof way[64]. These interrogative and demanding approaches are taken and applied for most analytical results but are usually not applied to the pre-processing of data[60].

Originally presented by Flaten and Walmsley in 2003[65] and published with an improved method by Gerretzen et al. in 2015[63], a design-of-experiment-based approach to pre-processing was suggested that would enable a brute-force interrogation and testing of the best-to-use pre-processing methods for the data problems at hand. Design of experiments (DoE) is a method to systematically ascertain relationships between variables and factors that influence a process or experiment and its outputs. Using such a design aids in understanding the influences the processing of spectra can have on the model performance and also give insights into what pre-processing methods might be required. On the subject of the paper outputs by Flaten and Walmsley as well as Gerretzen et al., a more in-depth comparison is offered in the discussion of this chapter[63,65]. It has to be noted that neither publication made their code publicly available. The proposed solutions to the pre-processing problem, however, were of significant interest to the CMAC community. At the research hub, frequent processing of spectral data is required and standardised, quick and helpful solutions for the pre-processing of data are not yet readily available. A tool that enables standardised processing of data with comparative reporting could allow more facile data exploration, easier standardised data access for further data research and an informed approach to pre-processing method usage that moves away from a trial-and-error style.

This chapter will introduce a DoE-based toolbox for comparing pre-processing methods for spectral data using partial least squares regression model testing. First, methods and algorithms used in the context of the toolbox are presented. This is followed by the steps the software takes to process and present the data. Finally, two example use-cases are shown exemplifying how the toolbox operates and the results it produces.

3.1.1 Objectives

The objectives of this work were the development and testing of a usable pre-processing tool that could recommend optimized and robust ways to pre-treat spectral data prior to further modelling. The tool should contain typical pre-processing methods applied in basic near-infrared spectroscopy, aid users in understanding why these methods are useful, help with outlier detection and modelling choices as well as output the data in a streamlined and reusable way. If possible, a graphical user interface should be present to reduce the barrier for new users and make it viable to be used as a web interface system.

3.2 Theory and Background

This sub-chapter is tasked with explaining the background to standard classification and regression methods, typical pre-processing methods used when treating spectral data and the background to standard spectral data sets.

3.2.1 Spectral Data Set

Spectral data typically consists of a matrix (X) containing instrument readings of intensity per sample (n) versus wavelength (k) observed. Matrix Y describes concentrations or other observational parameters of samples; these can be known or unknown quantities that are to be determined through the experiment. Together they build a spectral data set (Figure 3.1).

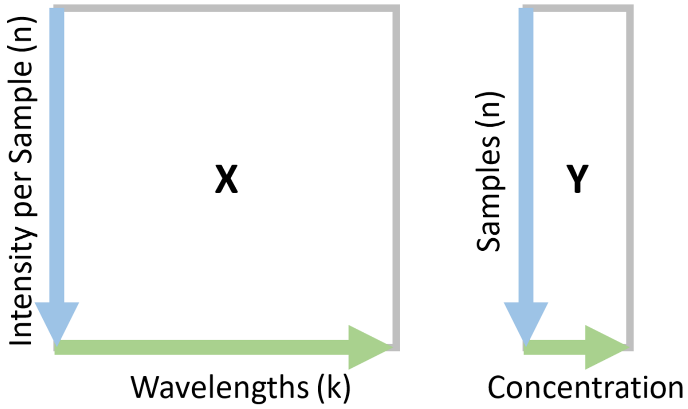


Figure 3.1: Matrix representation of a spectral data set.

In simpler terms, the X-components in data are the pure spectral components, intensity versus wavelengths, whereas Y-components are the known parameters of the samples, such as concentration of a compound in a sample.

3.2.2 Beer-Lambert Law

The Beer-Lambert law is essential for the analysis of solutions using spectral techniques. It explains how the absorbance of a solution is directly proportional to the concentration of the solution's constituents.

$$A_\lambda = -log(T) = \epsilon_\lambda \times l \times c \tag{3.1}$$

Here, A$_\lambda$ is absorbance
(depending on wavelength lambda of the incoming radiation),
T is the light transmittance,
epsilon is the molar absorptivity (depending on wavelength lambda),
l is the path length of the radiation going through the solution and
c is the concentration of the solution's constituent(s)[60,66].

Using the linear relationship between absorbance and concentration, concentrations can be directly calculated from spectral responses. The Beer-Lambert law thus builds the fundamental basis for typical spectral modelling calculations, such as building calibration models from sample spectral responses in combination with the known concentrations of the sample constituents[60,66].

3.2.3 Principal Component Analysis

Principal component analysis (PCA) is a dimensionality reduction technique that is typically used in predictive modelling and data exploration. In its simplest form, for a sample set in any dimensional space (see Figure 3.2), a line of best fit that minimises the average squared distance from each data point can be described. This line of best fit is called a principle component and depicts the maximum spread/variance of the data. When this process is repeated, the next line of best fit should be perpendicular to the previous line, thus building an uncorrelated description of the data between the different principal components that are generated[67].

The technique aids in the analysis of complex data by transforming and projecting a highly dimensional space into a lower dimensional space described by these principle components. By using the PCs the presentation of the data is simplified which helps to highlight similarities, clusters samples into groups and emphasises trends and patterns in the data[67,68].

The application of PCA is very common in ToF-SIMS analyses and is particularly popular to discriminate between spectra of similar biological materials such as cell and tissue types, which can be very difficult to analyse. Robinson et al., for example, applied PCA to analyse lipids of eight human breast cancer cell lines using ToF-SIMS and was able to differentiate the breast cancer cell lines based on their different fatty acid and lipid compositions[69]. Similarly, using ToF-SIMS and PCA, Baker et al. distinguished between prostate cancer cells and non-malignant cells based on their surface chemical composition emphasising that the use of multivariate techniques enabled the separation and analysis[70].

With regards to this thesis, PCA is primarily used as a data exploration technique to highlight and understand differences in human coronary artery endothelial cells before and after a certain drug treatment, (see Chapter 5). As previously explained, the first PC is generated by finding the minimum total distance between the available data points and the projected PC. The effect of this minimisation is that the PC describes the maximum possible variance in the data. This can be seen in Figure 3.2 and the sample distances from the blue line. Comparing PC1 with the x and y axes, PC1 covers a larger variation of the data (or range in this case) while also being the best-line-of-fit to describe the data.

Every following PC must be uncorrelated to the previous PCs, which can be visualised as plotting the variables orthogonal to each other, as Figure 3.2 shows[67,68]. The available variance in the data decreases with each subsequent PC, commonly making the first few variables the most descriptive and interesting with regards to finding patterns in the data. Using the previously described **X**-component of a spectral data set as an input, the method separates (decomposes) the relationships between the samples of **X**

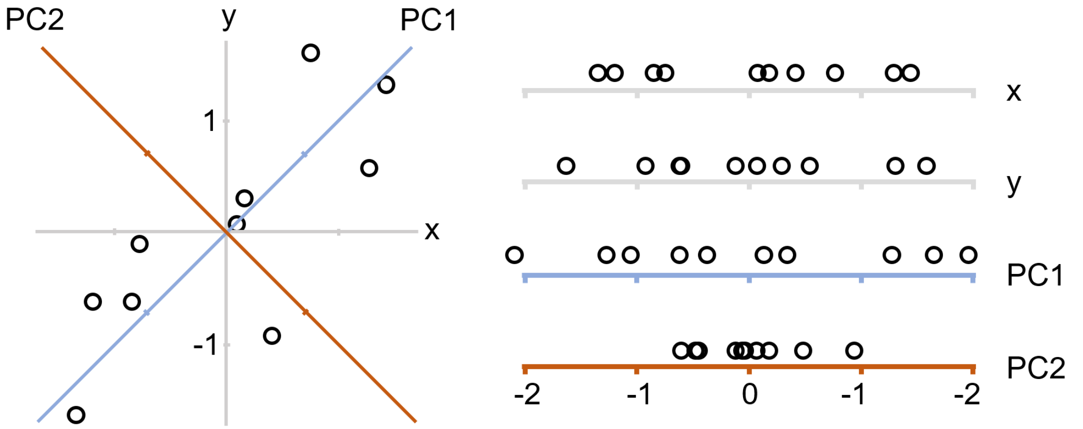


Figure 3.2: Values projected from 2D space onto 1D space along the x- and y- axis as well as along the lines representing the first two principle components (PC1 and PC2). Figure redrawn from Lever at al.[67].

into scores (**T**) and the variables of **X** into loadings (**P**), with all variance and noise that cannot be described via the scores and loadings being described as the residual (**E**). Each principle component is defined by these three sets of values, **T** (scores), **P** (loadings) and **E** (residuals).

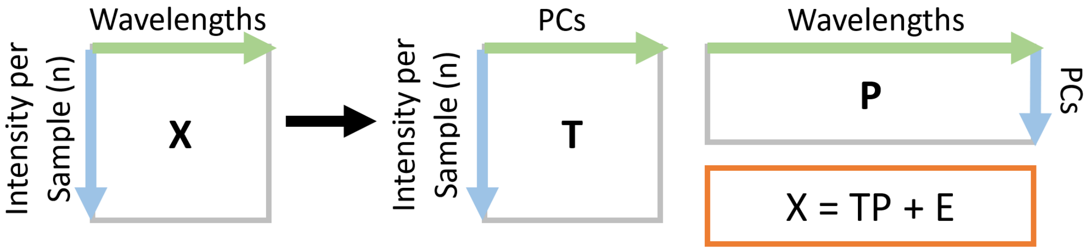


Figure 3.3: Matrix representation of principal component analysis decomposition from a spectral input matrix **X** where the relationships between the samples of **X** are described in scores (**T**) and the variables of **X** are described in loadings (**P**), with all variance and noise that cannot be described via the scores and loadings being described as the residual (**E**).

The results of a PCA should be analysed by viewing the scores and loadings of the various generated PCs. The scores describe the relationships between the different samples included within the model and typically indicate patterns within the data such as clustering of certain sample groups or specific samples being outliers. The loadings on the other hand describe which variables contribute towards the correlation of the samples with a certain principal component. Applied to a mass spectrometry example, different cell lines could be distinguished by using the PC scores while the loadings would indicate which changes in variables, i.e. the mass-to-charge ratios, are most correlated with the respective cell lines.

3.2.4 Partial Least Squares Regression

Regression analysis is used to establish and model the relationship between measurable variations in a system and its matching responses. These 'variations' (e.g. **X** matrix, such as spectra) can be measured, and based on them the 'responses' can be deducted (e.g. **Y** matrix, such as sample concentrations). As a simple example, given a water tap, the variation could be attributed as the amount of water released from the tap while the "response" equals to the extent of opening it. The example shows that variation and response are not necessarily physical attributes but correspond to the components that are measured and known. Building on this connection between variation and response, regression models can be built that assess change in the variations and output the corresponding responses. Data used for the creation of such models are called calibration data while data used to validate the correctness of the model are called validation data.

Numerous regression methods can be used for the analysis of multivariate systems, but this thesis will solely focus on the application of partial least squares regression (PLSR), in particular the SIMPLS algorithm applied through the MATLAB "plsregress" function. PLSR is similar to PCA but has an additional regression element where both the X (e.g. sample spectra matrix) and Y (e.g. sample concentration matrix) components of a data set are used in the modelling process.

PLSR is one of the most used and versatile regression methods in process analytical applications[71,72] and is preferred due to the model maximising correlation between the **X** and **Y** matrices by explicitly linking them in the model output variables while also maximising variance based on its PCA heritage. Due to the former, PLSR accounts for errors originating from both the spectral source (X matrix, e.g. the instrument used for the measurement) and the reference source (Y matrix, e.g. the prepared samples and their ‘known’ concentrations). This furthermore results in the models being more noise resistant, producing very robust model outputs[73].

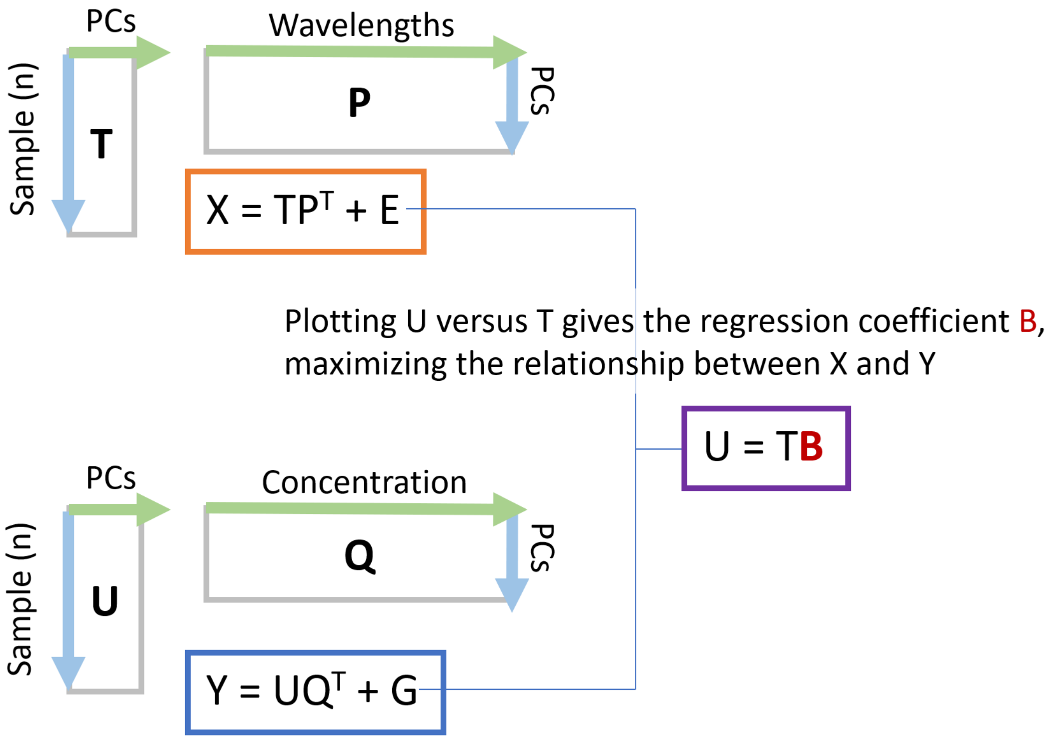

PCs
Wavelengths
Sample (n)
T
P
PCs
X = TP^T + E
Plotting U versus T gives the regression coefficient B, maximizing the relationship between X and Y
U = TB
PCs
Concentration
Sample (n)
U
Q
PCs
Y = UQ^T + G

Figure 3.4: Matrix representation of principal component analysis decomposition from a spectral input matrix **X** where the relationships between the samples of **X** are described in scores (**T**) and the variables of **X** are described in loadings (**P**), with all variance and noise that cannot be described via the scores and loadings being described as the residual (**E**).

Starting from the previously described **X** (sample spectra data set) and **Y** (sample concentration data set) matrices, the method produces variables that describe the relationships between the samples of **X** (**T**, X scores), the wavelengths of **X** (**P**, X loadings), the samples of **Y** (**U**, Y scores) and the concentrations of **Y** (**Q**, Y loadings). These estimations run in parallel and are used to regress **T** on U resulting in the regression coefficient **B**. This last step creates a direct relationship between **X** and **Y**. **B** can now be applied to a new raw set of data resulting in the prediction of an unknown set of **Y**[71,74,75].

The PLS algorithm starts by assuming an initial scores vector of Y, called u1, as Y. Hence, equation 1 starts with vector **u**1 (scores of matrix Y) that is assumed as a first estimate to be Y (Equation 2).

$$\mathbf{u}1 = \mathbf{Y} \tag{3.2}$$

Based on u1, w (weights for variables of the data expressed in X) are calculated by regressing u on X (Equation 3).

$$\mathbf{w}1^T = \mathbf{u}1^+\mathbf{X} \tag{3.3}$$

Then, t is calculated (scores of matrix X) by multiplying X and w (weights for variables of the data expressed in X) (Equation 4).

$$\mathbf{t}1 = \mathbf{X}\mathbf{w}1 \tag{3.4}$$

Followed by q, which is calculated (loadings of matrix Y) by regressing t (scores of matrix X) on Y (Equation 5).

$$\mathbf{q}1^T = \mathbf{t}1^+\mathbf{Y} \tag{3.5}$$

Based on Y and q (loadings of matrix Y), a new scores vector designated as u2 (an updated estimate of scores of Y) is then calculated (Equation 6).

$$\mathbf{u}2 = \mathbf{Y}\mathbf{q}1 \tag{3.6}$$

Comparing u2 together with q (loadings of matrix Y) as an estimate of Y against the original Y, results in the residual error G (Equations 7 and 8).

$$\mathbf{Y}_{original} = \mathbf{u}2\mathbf{q}1 + \mathbf{G}1 \tag{3.7}$$

$$\mathbf{G}1 = \mathbf{Y}_{original} - \mathbf{u}2\mathbf{q}1 \tag{3.8}$$

From here on the algorithm is looped using new values of u to minimise the residual error G. This means that the scores vector u has converged. The previous denominator of 1 is now replaced by j, indicating the j-th loop of the above calculation for which u has converged. When u has converged, p (loadings of matrix X) are calculated by regressing t on X (Equation 9).

$$\mathbf{p}_j^T = \mathbf{t_j}^+\mathbf{X} \tag{3.9}$$

B (the regression coefficient) can be calculated by regressing T on U (equation 10).

$$\mathbf{b}^T = \mathbf{t}^+\mathbf{u} \tag{3.10}$$

The model is summarised in the regression coefficient b which is used for making predictions based on the model. From here on, the regression coefficients b can be calculated. To make better predictions, it can be helpful to use a multiple regression model with multiple latent variables.

To obtain a multiple regression model, we now calculate the difference between the data and the obtained scores and loadings for the matrices of X and Y based on the model as residuals E and F. In this case, X and Y are now replaced by E and F and the process is restarted with the remainder of the variance that has not been described in the original latent variable, being expressed through the residual matrices of E and F (Equations 11 and 12).

$$\mathbf{E} = \mathbf{X} - \mathbf{T}\mathbf{P}^T \tag{3.11}$$

$$\mathbf{F} = \mathbf{Y} - \mathbf{U}\mathbf{Q}^T \tag{3.12}$$

The model creation process can be assumed to be complete when all information in X has been described by the model through an estimated Y, i.e. when E and F have been minimised. To assess this, cross-validation (as described in Section 3.2.4) and component selection (as described in Section 3.2.4) can be employed.

Partial Least Squares Regression Model Validation

Regression models need to be tested for accuracy and robustness of their predictions. Model prediction parameters such as the root mean square error of calibration (RMSEC), root mean square error of cross-validation (RMSECV) and root mean square error of validation (RMSEV, can also be found as error of prediction) are typically used to assess the accuracy and robustness of predictions of regression models[75,76]. These methods measure the squared mean differences between known and predicted Y-values, also called the root mean square error.

$$RMSE = \sqrt{(MSE(\widehat{y}))} = \sqrt{(sum(predicted - known)^2)} \tag{3.13}$$

The RMSEC uses the same data employed in the model creation to see how well the calibration performed. Used by itself this method can lead to overly optimistic prediction values as all training cases are known to the model. It is, however, a good first indicator of the successful creation of a model[75,76].

The RMSECV also uses the same data employed in building the model but in this case the data is divided into a training and a test set. Two typical ways of cross-validation (CV) are presented; leave-one-out CV and n-fold CV. In leave-one-out CV a model is built after removing one of the samples available in the calibration set. Using that sample as a test sample to trial the model an error value is extracted. This process is repeated until all samples in the calibration set have been tested. The cumulative error is averaged and the RMSECV calculated.

In n-fold CV the same method is applied but in this case with a group of n-samples. The group is removed from the calibration set and used as test samples with the model error values obtained being used for the RMSECV calculation.

Leave-one-out CV is the simpler method and applicable to most calibration sample sets. N-fold CV, on the other hand, requires a large number of samples in the calibration set which can be difficult to acquire. If enough data is available, n-fold CV is the preferred method but leave-one-out CV is a functioning and very compatible compromise.

CV is an improved method to test the predictive power of a model and its robustness. It is typically used for models with a limited number of samples and if no separate validation test set is available. Plotting the RMSECV versus the number of latent variables is a good indicator of the required number of latent variables and is often used for this purpose[75–77]. When the error plateaus and does not decrease anymore, it is clearly visible at which point an increase in the number of latent variables does not equate in a meaningful reduction in error and the improved expression of variance of the system. A slightly more in-depth explanation for this can be found in section Section 3.2.4.

The ideal way to test the validity of a model is to acquire a separate test set which can be used to calculate the RMSEP. The sample set should preferably be designed to span the entirety of the model predictive space and should not contain sample values present in the calibration set. The prediction error of this test depicts a real-world test scenario and represents the potential model predictive error best[75,76].

Finally, the accuracy of the model can be tracked using the coefficient of determination, R^2, which measures the closeness of the data to the fitted regression:

$$R2 = 1 - SSres/SStol = \sum (y_i - y_{ipredicted})^2 / \sum (yi - y_{mean})^2 \tag{3.14}$$

SSres is the 'sum of squares due to regression' in this context, while SStol is the 'total sum of squares'. SSres describes the closeness of the regression model values to the data that has been modelled while SStol describes the total variation of the calibration data.

R^2 can attain values between 0 and 1, with 1 representing maximum correlation between the predicted and measured values and 0 meaning no correlation.

Outlier Detection

Outlier detection is concerned with the identification of samples or data points that do not agree with the general trend and vary significantly from the general data observations. There are many reasons for outliers to appear, for example instrument errors, measurement errors, human errors and data processing issues. Such erroneous data points can cause problems in further statistical analysis and modelling and must be identified and investigated prior to their potential removal. It is imperative to find out what these samples are, where they come from and to understand whether they should be used in the model building process. There are three major cases of outliers:

1. Instrument errors
2. Sample belonging to a different sample population
3. Errors in the reference source

Case 1: Instrument errors can occur at random and can affect singular samples or entire sets depending on the defect. Case 2: Samples that belong to a different population than the standard samples, such as analysing a water/isopropanol (IPA) mixture and accidentally mixing up the solutions with a water/ethanol mix will result in outliers in the spectrum. Both cases result in different relationships between the intensities per sample versus the wavelength recorded (X-matrix effect). Case 3 on the other hand would result in a very different outcome due to the disparity between the X- and Y-matrices thus causing a mismatching regression. This would occur for example if the samples were made up incorrectly and thus labelled wrongly resulting in the Y-matrix being wrong in comparison to the spectral responses[75].

Further differentiation can be made between outliers obtained during calibration and those formed during prediction. Calibration outliers are outliers that are involved in the prediction equation and as such could lead to a systematic error during subsequent predictions. Thus, it is imperative to filter these outliers out and have a clean and robust set of calibration samples before modelling. Prediction outliers are outliers that are not involved in the prediction equation but are likely to result in flawed Y prediction values. In this case, only calibration outliers are tackled.

The toolbox is concerned with finding an optimal pre-processing method for the data at hand. To ensure best results, the calibration data must be tested for outliers as these might cause detrimental model prediction results. To detect such outliers, a Monte Carlo based outlier detection algorithm published in the libPLS toolbox by Li et al. (2014) is used[78]. Monte Carlo methods are a class of algorithms that calculate mathematical problems by repeated random sampling[79]. A graphical schema of the calculation steps involved can be seen in Section 3.5.

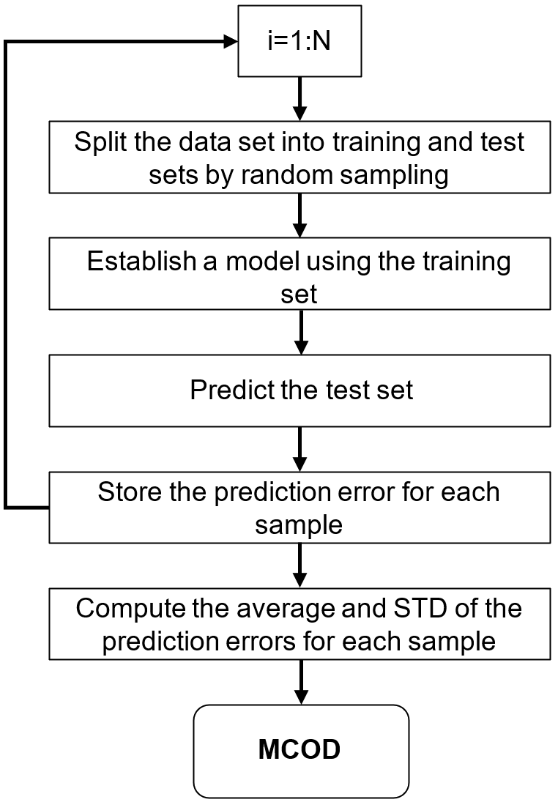

i=1:N
Split the data set into training and test sets by random sampling
Establish a model using the training set
Predict the test set
Store the prediction error for each sample
Compute the average and STD of the prediction errors for each sample
MCOD

Figure 3.5: Monte Carlo based outlier detection workflow, reproduction from[80]

As the algorithm is based on brute-force PLS model parameter assessment the complexity of the sample system dictates the number of latent variables required for outlier assessment. The samples are being randomly split into training (calibration) and test sets (validation). A simple model is created to test how good the training set performs, and the errors of prediction are stored for each sample used. This method is repeated N times with different combinations of samples (currently set to 1000)(see Figure 3.5). The mean standard deviation for each sample is calculated and presented to the user in graph form. The method forecasts that a sample with a significantly higher mean prediction error has a high likelihood of being an outlier. Based on the output of the algorithm the user can check flagged samples and chose to continue with or without the sample(s) in question.

Component Selection

The complexity of a system often dictates the number of latent variables (components) required to describe the changes in the system. It is usually preferred to use as few components as necessary to only model the relevant changes within the data and not cause the model to be overfit. Overfitting a model to its data in this context means that a model is overly specialized in the data set it is trained on, predicting trends and data points specific to the samples it has been given. This can lead to bad prediction performance and decreases robustness with respect to the actual model application. As previously stated, to choose the optimal number of latent variables (LVs) the RMSECV is plotted against the number of LVs used. The RMSECV typically drops off significantly within the first few LVs used, then starts to plateau and increase again. The optimal model can usually be found at the start of the RMSECV plateau[77]. At this point the model describes the optimal variance in the data without overfitting while keeping the model as simple as possible.

3.2.5 Pre-Processing Steps and Methods

Pre-processing is a critical step during spectral data preparation and should be performed prior to any further chemometric analysis. It aids in the removal of instrument and scatter effects that can vary with every analysis and thus makes data more comparable and robust[60]. Spectral pre-processing techniques in NIR can be largely divided into spectral derivatives and scatter correction method[60,65]. The toolbox currently uses the most common pre-processing techniques, but further methods can be added, if necessary.

Spectral Derivatives

Spectral derivatives belong to the standard repertoire of pre-processing methods applied and are used to improve spectral resolution and to remove baseline related issues within the spectra[81,82]. This is accomplished by using first and higher orders of derivatives of spectral intensity with respect to their wavelength equivalent unit. The most commonly used algorithms are the Norris-Williams algorithm and the Savitzky-Golay (SG) polynomial derivative filter with the latter being the only technique applied and thus discussed in this thesis. The filter functions by regressing and deriving small subsequent segments of data to a fitted polynomial function using a linear least squares function. The width of segments must be predetermined and determines the smoothing and binning of the filter. The higher the degree of the polynomial applied the better the fit to the data but subsequently noise sensitivity will be increased. The derivative step is optional and as previously mentioned is primarily used for the removal of baseline effects[60,76,83].

Scatter Correction Methods

Scatter correction methods are primarily used to reduce or eliminate inconsistencies between sample measurements due to light scattering and particle size differences in samples. Scattering occurs when emitted radiation from samples does not follow a predictable path but rather random direction thus resulting in erroneous readings. It is more likely to occur when analysing solutions with smaller particulate matter or high differences in refractive indices[68]. Scatter correction is particularly important for biological and powder samples due to their large variability in uniformity, size and area, making them very susceptible to undesired scatter effects[60,76].

The two most commonly used scatter correction techniques are multiplicative scatter correction (MSC) and standard normal variate (SNV) correction. Both MSC and SNV try to account for offset and scaling effects[84]. In MSC a mean spectrum is generated, typically from the calibration data set, which subsequently is used to normalise the spectra to be corrected (column vector of each variable) by regressing them against each other. Due to a mean spectrum being used in the original calibration correction the same mean spectrum must be applied to all succeeding corrections[76,84] The first fitting step can be described through Equation 3.15 where a, the intercept, describes additive effects, b, the slope, describes multiplicative effects and the residuals, e, which are effects otherwise unaccounted for[68].

$$x_{ik} = a + b\ m_i + e \tag{3.15}$$

The intercept and slope variables are calculated via least-squares regression of each spectrum $\mathbf{x_i}$ versus the mean spectrum $\mathbf{m_i}$ over each wavelength **k**. Additive effects typically originate from inter-sample path-length differences and cause baseline shifts leading to offsets in the y-axis. Multiplicative effects on the other hand classically stem from concentration and particle size differences that cause deviations in light scatter behaviour and thus are responsible for changes in the spectra[60,68]. After the first step is complete the raw spectra are then corrected using Equation 3.16.

$$x_{ik}(MSC) = \frac{x_{ik} - a}{b} = \ m_i + \frac{e}{b} \tag{3.16}$$

SNV on the other hand is a set-independent method[85] where sample spectra are scaled and centred individually based on the mean and standard deviation of each sample, making them independent from the set and reducing spectral intensity variation for each spectrum[76].

$$x_{ik}(SNV) = \frac{(x_ij - m_i)}{SD_i} \tag{3.17}$$

Equation 3.17 describes the mathematical base for SNV. The corrected spectrum $\mathrm{x_{ik}(SNV)}$ is obtained by subtracting the mean spectrum mi, based on all samples, from each spectral quantity of sample i at wavelength k followed by the subsequent division using the sample i standard deviation. Due to this procedure the resulting spectra are independent from the original intensity values and have a variance of 1 centred around a mean value of zero[68]. According to Dhanoa et al.[85] both methods are interconvertible and related making them very similar. The major difference is the set-dependent versus set-independent calculation making MSC susceptible to introducing bias to the corrections through the inclusion of the set. As previously stated, both methods are widely used and as such are included in the toolbox.

Data Normalization, Scaling and Centring

Scaling and data normalization are used to counteract disproportionate spectral responses from samples that experience scaling effects. In these cases, the true value of some measured variables can be affected by positive or negative multiplicative factors which can impact the information gained from the spectra[86,87]. Such effects can occur for a multitude of reasons, including effects related to the instrument, source, scattering and pathlength, which are typically counteracted by finding a standard or reference within the sample to which the rest of the samples can be normalized to[84]. After the application of scatter correction methods, which can account for some of the negative effects, further processing steps may be required.

Mean-centring is the gold standard centring method used in pre-processing and is applied to most modelling data. Here, an average spectrum is generated from the data set and subtracted from each individual spectrum. The effect is a general centring of the data around zero improving interpretability and readability of the data for regression modelling[68,86]. In addition to centring the data, two scaling methods are discussed here, namely, Pareto scaling and auto scaling.

Pareto scaling takes mean-centred data points and divides them by the square root of the standard deviation as the scaling factor of the spectrum. Using this scaling method, the relative importance of large intensities is reduced and the data centred to zero while still keeping the variance[86]. Auto scaling is similar to Pareto scaling. Here the mean-centred data points are scaled using the standard deviation as a scaling factor. It is not commonly applied in spectral scaling due to it possibly inflating measurement errors and a chance of loss of sensitivity regarding small changes in the intensity of spectra. On the other hand, it can be very useful if variables of small and large intensities are important and should be accounted for. The data is centred to zero and the variance of the data is lost[86].

3.2.6 Experimental Design Approach

The design-of-experiment (DoE) approach is a concept introduced by Fisher in 1936 that envisions an experimental approach that is faster, more informative and more scientific than a 'classical' approach with one experiment at a time changing one variable at a time[88,89].

In its practical application during an experiment, multiple variables are changed systematically and simultaneously at pre-set varying levels[89]. Using this method allows to not only measure the effects of the variables on the experimental results but also gather insight on the interactions between the changing variables and their levels[65].

Over the years a multitude of DoE methods have been developed and applied. In this case, a full factorial design is applied in which all different variables (pre-processing methods per pre-processing step) are tested against each other using pre-defined levels. All possible combinations and levels are tested against each other. For example, all available methods of scatter correction are applied, as they have no additional levels of adjustments, each will be applied with 1 level. Baseline correction on the other hand can be further adjusted with varying levels of polynomials and derivatives, hence a number of preselected levels are applied. Tables 3.1 and 3.2 below describe the toolbox available pre-processing methods and options. To test the toolbox, we will use all available methods in our DoE and combine each combination of separate pre-processing steps available, an example of which is shown in Table 3.2. The method is based on the leading paper by Flaten and Walmsley from 2003[65] and resulted in the design of an easy-to-use, fast and adaptable MATLAB toolbox that follows their initial DoE pre-processing idea combining it with effective outputs presented by Gerretzen et al. in 2015[63]. These will be further discussed in the discussion section of this chapter (Section 3.6).

Table 3.1: Available pre-processing methods in the toolbox with their differing levels. Smoothing performed using Savitzky-Golay, order in table denotes order of polynomial applied.

Baseline	Scatter	Smoothing	Scaling
1st Derivative 2nd order	MSC	Smoothing 2nd order	Mean-Center
1st Derivative 4th order	SNV	Smoothing 4th order	Pareto
2nd Derivative 2nd order	none	none	Auto-Scaling
2nd Derivative 4th order			
None			

Table 3.2: Full factorial design example: Each pre-processing method is assigned a number and a matrix of all available combinations is created. This matrix is then later used to apply each of those combinations of pre-processing steps to the data.

Baseline	Scatter	Smoothing	Scaling	Latent Variables
0	0	0	1	1
1	0	0	1	1
2	0	0	1	1
3	0	0	1	1
4	0	0	1	1
...	...	...	...	...
4	2	2	3	Nth variable

3.2.7 Toolbox Working Principle

Data Requirements

The toolbox is based on Mathworks MATLAB and is not a standalone product. As such, MATLAB is a requirement. Tested and supported versions include MATLAB 2014 to MATLAB 2019 a and b. Further requirements are MATLAB-readable raw spectral data sets for calibration and, if available, validation data.

Toolbox Stages

Figure 3.6 represents the steps involved in the toolbox prior to presenting the user with the calculated results and figures.

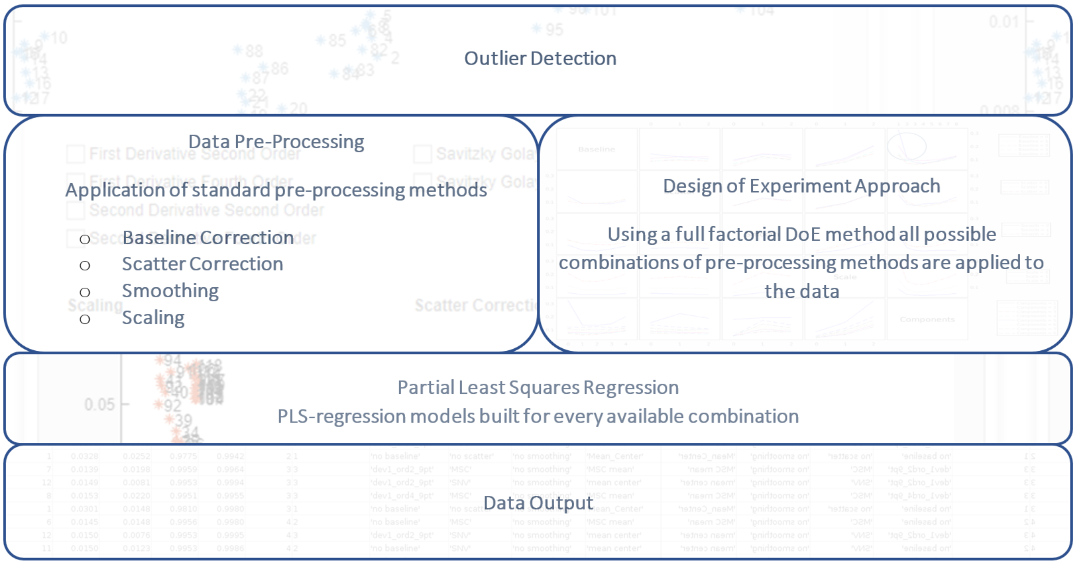

Outlier Detection
Data Pre-Processing
Application of standard pre-processing methods
○ Baseline Correction
○ Scatter Correction
○ Smoothing
○ Scaling
Design of Experiment Approach
Using a full factorial DoE method all possible combinations of pre-processing methods are applied to the data
Partial Least Squares Regression
PLS-regression models built for every available combination
Data Output

Figure 3.6: This figure outlines the step-by-step process of the toolbox.

The toolbox initially aids the user with outlier detection, which ensures that erroneous samples are removed prior to the pre-processing assessment. Outliers can be directly identified, and improvements assessed prior to further steps being triggered. The next step is the creation of a design-of-experiment-based pre-processing step and method matrix which ensures all available combinations of pre-processing methods are applied to the data. This includes typical baseline correction, scatter correction, smoothing and scaling methods.

Once this full factorial experimental design has been defined, models for each of the applied combinations of pre-processing methods are built to brute-force assess the performance of each of those models. The data outputs from those generated models are then compared in various figures to assess what level of complexity is required for the models, which pre-processing methods are helpful, work best and possibly interfere with each other. While the PLS model output can suffice in simple cases, it is advised using the toolbox as a guide to choose the correct pre-processing methods prior to using external tools for modelling. Currently, only PLS1 is implemented and will likely fail to deliver meaningful model outputs beyond pre-processing method advice if used on data that require multivariate modelling.

3.3 Use Case #1: Mixing of Two Solvents – IPA versus Water

3.3.1 Experimental Background

In a proposed experiment ICT-CMAC aimed to measure the mixing of two solvents at steady state within a continuous capillary-centred static mixer using a NIR-HSI probe. In their experiment, Dziewierz et al. used a concentric capillary continuous static mixer built from a Polyether-Ether-Ketone capillary of 0.51 mm internal and 1.588 mm outer diameter which was surrounded by a quartz glass tube of 3.0 mm internal and 5.0 mm outer diameter. Solvent A is pumped through the ether capillary while solvent B is pumped through the quartz tube. Coriflow mass flow meters are used to control flow rates of both solvents (Bronkhorst USA Inc.). Both solvents are set to different mass flow rates to guarantee turbulent mixing in the area of mixing Figure 3.7.

A container was installed around the experiment fitted with Zenith Polymer White Diffuser 95% R (Pro-Lite Technology LTD, Cranfield, England). Three 100 W halogen light bulbs controlled by a pulse-width modulation dimmer were used as a light source. This boxed setup ensures only standardized light would pass through the static mixer and hit the camera optics. A square silver mirror (Thorlabs Inc.) in connection with a brushless DC motor was used as a scanning mirror to scan the entire area of the cuvette from top to bottom. This setup is represented in Figure 3.7.

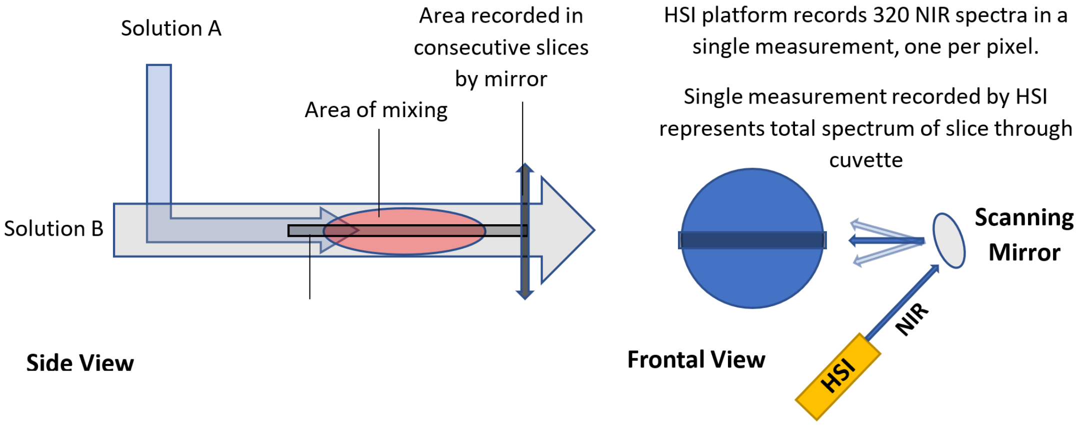

Solution A
Area recorded in consecutive slices by mirror
HSI platform records 320 NIR spectra in a single measurement, one per pixel.
Area of mixing
Single measurement recorded by HSI represents total spectrum of slice through cuvette
Solution B
Scanning Mirror
NIR
HSI
Side View
Frontal View

Figure 3.7: This figure represents the HSI experimental setup with a side and frontal view of the cuvette. The frontal view shows the scanning mirror that is used to record the NIR spectra using the HSI probe, one spectral slice at a time. The side view shows the mixing solutions in the setup and the area of mixing where the HSI is set to record.

Materials and Software

An Innospec RedEye 1.7 hyper-spectral imaging (HSI) camera with an InGaAs sensor was used with spectral settings adjusted to 900-1700 nm (3.2 nm resolution and frame rate of 330 frames/s). Software required for data analysis, chemometrics and imaging were MATLAB 2015b, MATLAB 2016a (The MathWorks Inc.) and PLS Toolbox v7.9.5 (Eigenvector Research Inc.). Chemicals used were isopropanol (IPA; Fisher Scientific UK. Ltd, 99.5% purity) and water, purified using the Integral 15 Milli-Q water purification system (Merck Millipore KGaA).

Data

The HSI camera records 320 pixels (single row recorded by HSI, Figure 3.7) at 256 wavelengths between 900 and 1700 nm (3.2 nm resolution). The mechanical mirror scanner adds further 151 slices resulting in a total hypercube image with 12 million data points. The mirror and camera were set to record 4 images per minute. Each pixel in the image (320x151) contains a NIR spectrum between 900 and 1700 nm with

256 data points which requires pre-processing and analysis for future model building and information extraction purposes. All samples were recorded by initially flushing the system with pre-mixed sample solutions of a given IPA/water concentration followed by recording hyperspectral images. Each sample acquisition and analysis were performed in triplicate for calibration (training data) and validation (testing data) runs. The experimental data thus represents the recorded changes in NIR spectra upon adding and mixing IPA (with an over-time increasing concentration) into a flowing water solution. Calibration and validation sets were recorded with 118 and 102 samples respectively. Calibration and validation data sets were recorded on differing dates.

Issues

Due to a round capillary tube used in the experiment, ray tracing problems throughout the tube were revealed. The refractive index between air, water and IPA differ and hence, rays that are not passing through the central horizontal axis will be diffracted and become unusable without a very advanced ray tracing algorithm that could not be completed as part of the original project. Consequently, for this set of experiments only the central axis was used. To circumvent this issue the usage of a rectangular capillary tube has been suggested for future experiments.

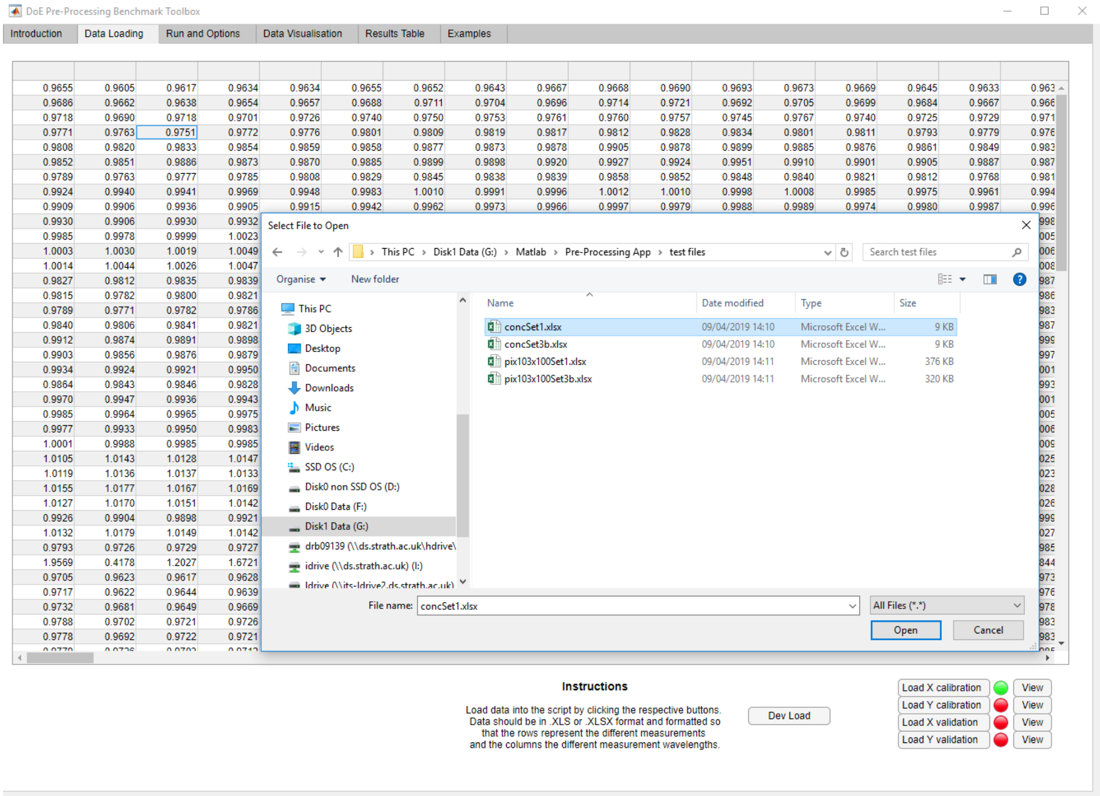
Extraction of Useful Data For Pre-Processing Toolbox

Calibration and validation data acquired during the experiments make for simple example data sets for exemplifying the capabilities of the pre-processing toolbox. Data from a single pixel of the central horizontal axis, over the course of all calibration and validation runs, was extracted and prepared for usage with the MATLAB based toolbox. This pixel was chosen as it represents all pixels with a low amount of ray tracing issues.

3.3.2 Results

This sub-chapter will present the toolbox and its features based on the NIR data previously described. The toolbox has been developed with IR and NIR data in mind but could likely be applied to general spectral data which is to be used in regression modelling. This section can be seen as a high-level manual for the usage of the DoE pre-processing benchmark toolbox and will follow the step-by-step application on the data mentioned above.

Step 0: Data import



0.9655	0.9605	0.9617	0.9634	0.9634	0.9655	0.9652	0.9643	0.9667	0.9668	0.9690	0.9693	0.9673	0.9669	0.9645	0.9633	0.963
0.9686	0.9662	0.9638	0.9654	0.9657	0.9688	0.9711	0.9704	0.9696	0.9714	0.9721	0.9692	0.9705	0.9699	0.9684	0.9667	0.966
0.9718	0.9690	0.9718	0.9701	0.9726	0.9740	0.9750	0.9753	0.9761	0.9760	0.9757	0.9745	0.9767	0.9740	0.9725	0.9729	0.971
0.9771	0.9763	0.9751	0.9772	0.9776	0.9801	0.9809	0.9819	0.9817	0.9812	0.9828	0.9834	0.9801	0.9811	0.9793	0.9779	0.976
0.9808	0.9820	0.9833	0.9854	0.9859	0.9858	0.9877	0.9873	0.9878	0.9905	0.9878	0.9899	0.9885	0.9876	0.9861	0.9849	0.983
0.9852	0.9851	0.9886	0.9873	0.9870	0.9885	0.9899	0.9898	0.9920	0.9927	0.9924	0.9951	0.9910	0.9901	0.9905	0.9887	0.987
0.9789	0.9763	0.9777	0.9785	0.9808	0.9829	0.9845	0.9838	0.9839	0.9858	0.9852	0.9848	0.9840	0.9821	0.9812	0.9768	0.981
0.9924	0.9940	0.9941	0.9969	0.9948	0.9983	1.0010	0.9991	0.9996	1.0012	1.0010	0.9998	1.0008	0.9985	0.9975	0.9961	0.994
0.9909	0.9906	0.9936	0.9905	0.9915	0.9942	0.9962	0.9973	0.9966	0.9997	0.9979	0.9988	0.9989	0.9974	0.9980	0.9987	0.996
0.9930	0.9906	0.9930	0.9932													998
0.9985	0.9978	0.9999	1.0023													005
1.0003	1.0030	1.0019	1.0049													006
1.0014	1.0044	1.0026	1.0047													008
0.9827	0.9812	0.9835	0.9839													987
0.9815	0.9782	0.9800	0.9821													986
0.9789	0.9771	0.9782	0.9786													983
0.9840	0.9806	0.9841	0.9821													987
0.9912	0.9874	0.9891	0.9898													999
0.9903	0.9856	0.9876	0.9879													997
0.9934	0.9924	0.9921	0.9950													001
0.9864	0.9843	0.9846	0.9828													993
0.9970	0.9947	0.9936	0.9943													001
0.9985	0.9964	0.9965	0.9975													005
0.9977	0.9933	0.9950	0.9983													006
1.0001	0.9988	0.9985	0.9985													009
1.0105	1.0143	1.0128	1.0147													025
1.0119	1.0136	1.0137	1.0133													023
1.0155	1.0177	1.0167	1.0169													028
1.0127	1.0170	1.0151	1.0142													026
0.9926	0.9904	0.9898	0.9921													999
1.0132	1.0179	1.0149	1.0142													027
0.9793	0.9726	0.9729	0.9727													985
1.9569	0.4178	1.2027	1.6721													844
0.9705	0.9623	0.9617	0.9628													973
0.9717	0.9622	0.9644	0.9639													976
0.9732	0.9681	0.9649	0.9669													978
0.9788	0.9702	0.9721	0.9726													983
0.9778	0.9692	0.9722	0.9721													983

Select File to Open

This PC › Disk1 Data (G:) › Matlab › Pre-Processing App › test files

Name	Date modified	Type	Size
concSet1.xlsx	09/04/2019 14:10	Microsoft Excel W...	9 KB
concSet3b.xlsx	09/04/2019 14:10	Microsoft Excel W...	9 KB
pix103x100Set1.xlsx	09/04/2019 14:11	Microsoft Excel W...	376 KB
pix103x100Set3b.xlsx	09/04/2019 14:11	Microsoft Excel W...	320 KB

File name: concSet1.xlsx

All Files (*.*)

Instructions

Load data into the script by clicking the respective buttons.
Data should be in .XLS or .XLSX format and formatted so
that the rows represent the different measurements
and the columns the different measurement wavelengths.

Figure 3.8: Graphical user interface to load data into the toolbox.

Data can be loaded into the toolbox via a graphical user interface (Figure 3.8) advising the user which sets of data have been imported and are still required. Furthermore, the numerical data can be previewed in a table viewer. The only data formats currently available are .XLS and .XLSX files. For the X-components file (spectra) rows represent samples and columns represent the wavelengths . For Y-components (concentrations) a separate file is necessary, where rows represent samples and columns represent the measurement variables.

Step 1: Outlier Detection using Monte Carlo Based Cross-Validation

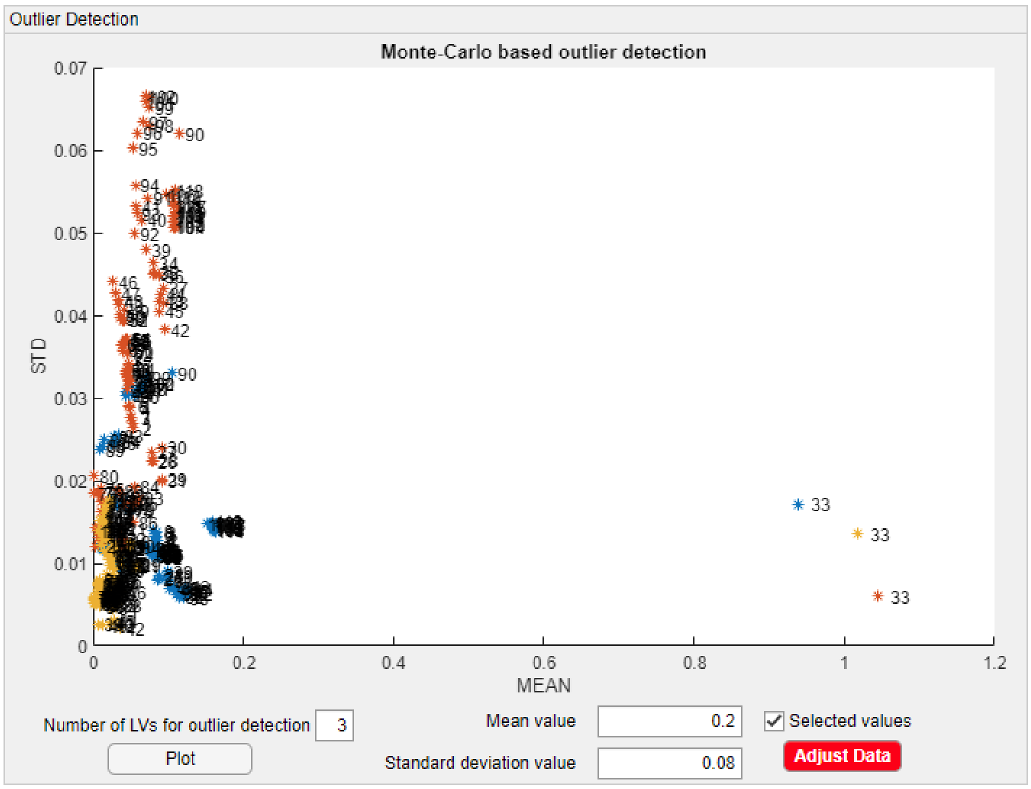
As previously stated, the toolbox is concerned with finding an optimal pre-processing method for the data at hand. For this purpose, outliers must be found prior to the pre-processing and modelling steps. Using the previously described Monte Carlo-based outlier detection, the calibration samples of the data set were used to check for outliers. The outlier detection algorithm allows for fine-tuning with regards to the complexity of the models and number of latent variables with which the outliers will be tested with (see bottom left of Figure 3.9).

In the case of this calibration set with 118 samples 3 latent variables were chosen to assess the data for outliers. The number of latent variables chosen here is arbitrary but for more complex systems of data can help reveal more intricate outliers by modelling the basic variation of the system. These 3 LVs are represented using different colours, LV1 in blue, LV2 in orange and LV3 in yellow.

Figure 3.9 shows the algorithm output with the standard deviation of prediction errors of each sample plotted against the mean prediction error of each sample. Examining the figure, sample 33 appeared to have a much higher mean prediction error compared to all other samples hinting at a potential problem of the measurement. To assess the issue, NIR spectra for representative calibration samples, including sample 33, were further compared (Figure 3.10).

Sample 33 is easily recognized in the plot as the erratic green spectral signal. It seems an instrument error or extreme scatter event occurred during acquisition rendering this sample unusable. Using the toolbox and based on the outlier analysis, all samples with a mean prediction error larger than 0.2 were deselected for further modelling purposes and the calibration sample set corrected from 118 to 117 samples. A value of 0.2 was chosen as the cut off to remove sample 33.

Viewing Figure 3.11 and comparing it to Figure 3.9 a near sevenfold decrease in standard deviation of prediction errors can be observed in addition to a larger amount of clustering of samples after removal of the outlier. This single erroneous sample drastically increased the standard deviation of prediction errors and reduced the accuracy of the entire model. This can be seen in particular when comparing the RMSEC of la-



Number of LVs for outlier detection 3

Mean value 0.2

Standard deviation value 0.08

☑ Selected values

Figure 3.9: MC outlier detection output showing the standard deviation of prediction errors of each sample (STD)versus the mean prediction error of each sample (MEAN). Error detection was run with three latent variables, blue being LV 1, orange LV 2 and yellow LV 3. Sample 33 shows a significantly higher mean prediction error than all other samples and thus requires further attention.

tent variables 1-3 with and without outlier detection (Table 3.3). The outlier increased the error of the model with one latent variable by 1.6%. Subsequently more complex models with more latent variables suffered an increase of 0.2% in RMSEC. For more complex systems and more potential outliers it is apparent that outlier detection is not an option but a necessity.

Step 2: Pre-Processing Method Selection and DoE Application

The toolbox allows the selection and easy addition of pre-processing methods that should be benchmarked against the data.

Figure 3.12 shows the GUI selection options to choose from prior to the toolbox gener-

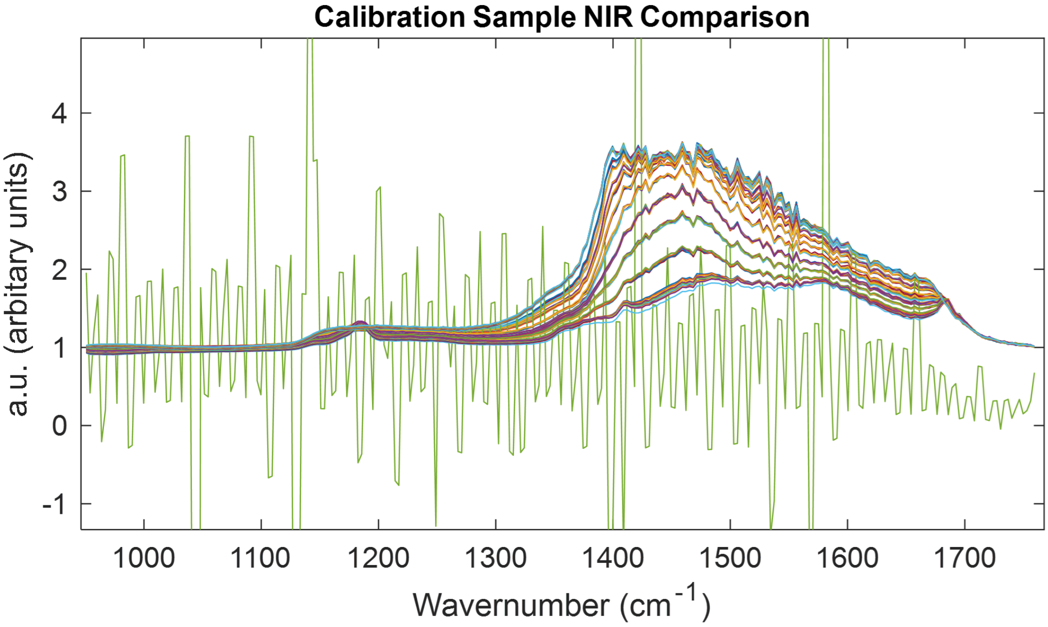
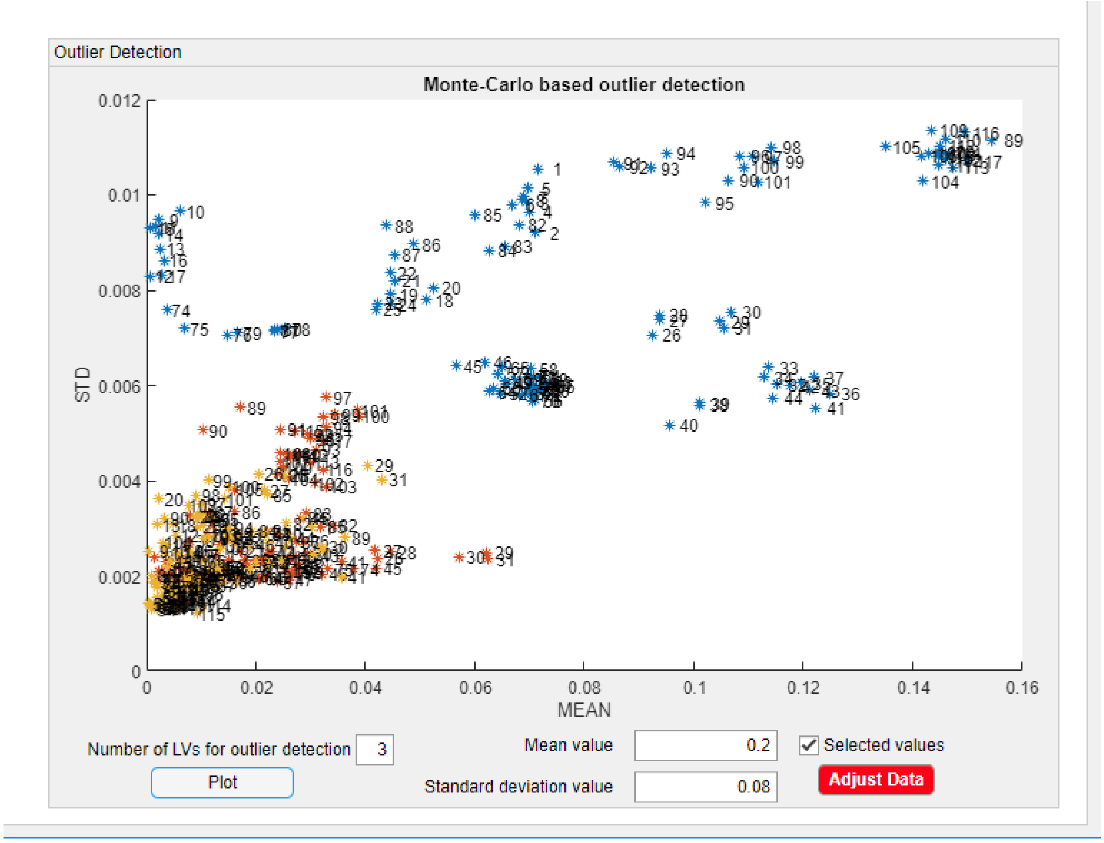


Figure 3.10: Comparison between all 118 calibration NIR spectra.

Table 3.3: Outlier Detection Comparison, best results for the first 3 latent variables with and without outlier detection.

	1 LV	2 LV	3 LV
Without OD	4.7%	2.2%	1.6%
With OD	3.1%	2%	1.4%



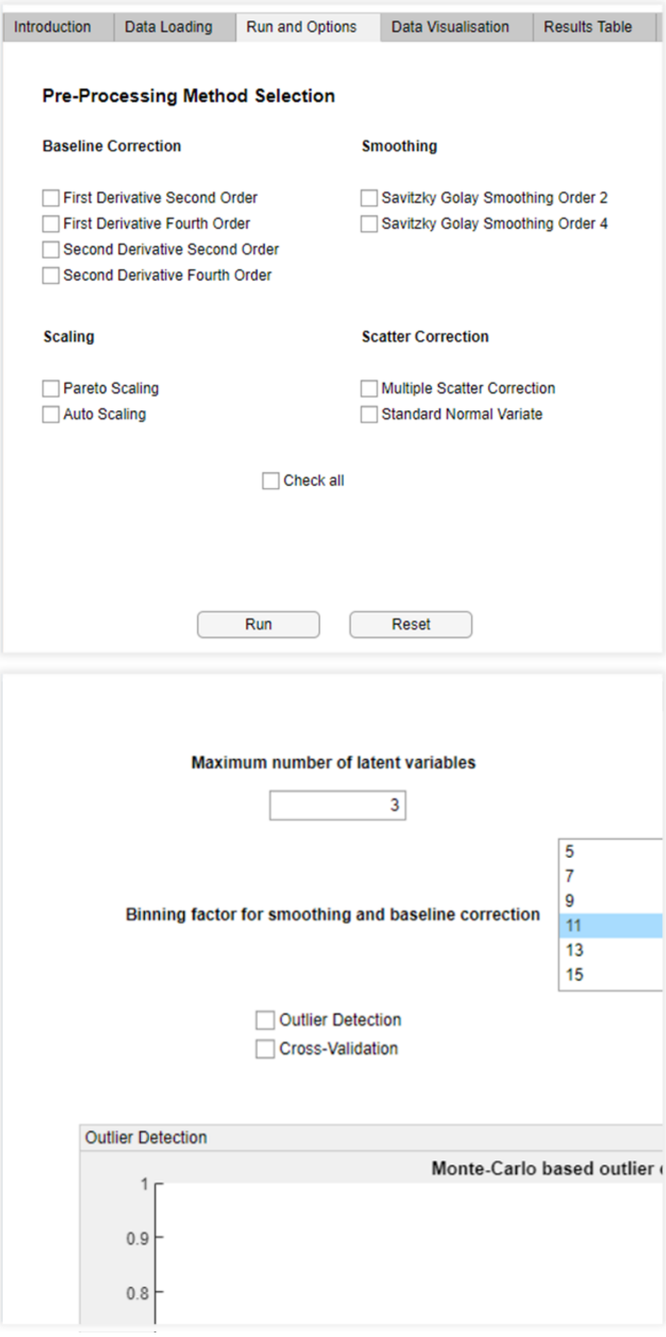
Number of LVs for outlier detection 3

Mean value 0.2

Standard deviation value 0.08

☑ Selected values

Figure 3.11: MC outlier detection output showing the mean prediction error of each sample (MEAN) versus the standard deviation of prediction errors of each sample (STD) for all calibration samples after outlier removal.


Introduction
Data Loading
Run and Options
Data Visualisation
Results Table
Pre-Processing Method Selection
Baseline Correction
□ First Derivative Second Order
□ First Derivative Fourth Order
□ Second Derivative Second Order
□ Second Derivative Fourth Order
Smoothing
□ Savitzky Golay Smoothing Order 2
□ Savitzky Golay Smoothing Order 4
Scaling
□ Pareto Scaling
□ Auto Scaling
Scatter Correction
□ Multiple Scatter Correction
□ Standard Normal Variate
□ Check all
Run
Reset
5
7
9
13
15
□ Outlier Detection
□ Cross-Validation
Outlier Detection
Monte-Carlo based outlier
1
0.9
0.8

Maximum number of latent variables: 3

Binning factor for smoothing and baseline correction: 11

Figure 3.12: GUI representations of the different pre-processing methods and selection options available.

ating the testing matrix. As previously described typical spectral pre-processing steps include;

- Baseline correction
- Scatter correction
- Smoothing of spectra
- Scaling and centring

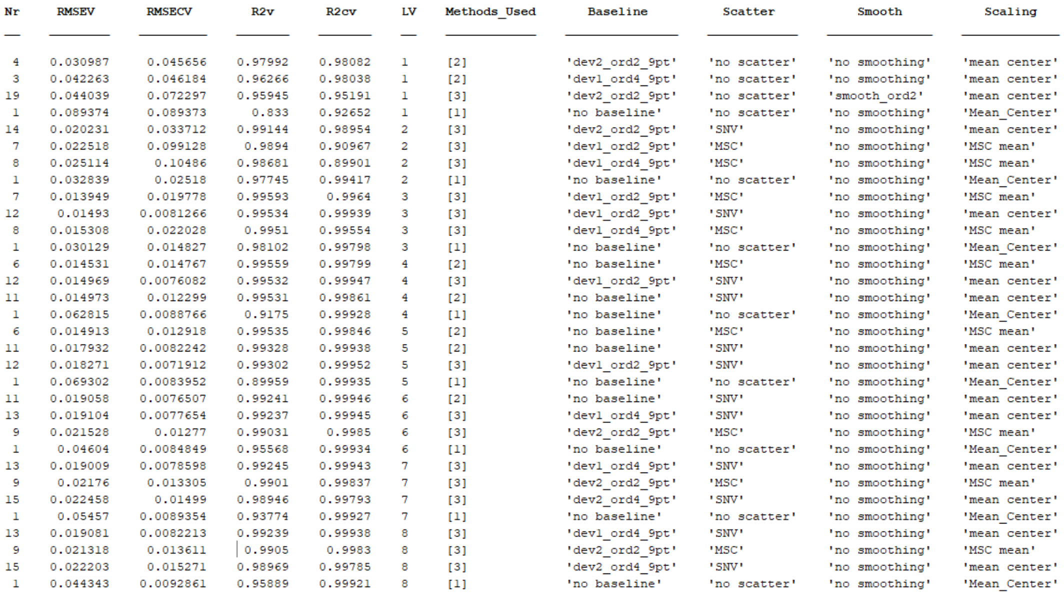
as well as the binning frame length for the spectral baseline correction and smoothing methods. This factor describes how many data points are used for each smoothing and baseline correction calculation.

After selection of the desired methods a full factorial experimental design is applied and a matrix built with each pre-processing method and step combination possible. Each of the defined combinations in the matrix are then applied to the data and saved. In the case of our data, the modelling problem does not seem complex but as a showcase, 8 latent variables are chosen. Taking into account all pre-processing methods previously shown and assuming 8 latent variables a total of 1080 experimental calibration sets are built.

Step 3: PLS Model Building

All 1080 previously saved experimental calibration sets are now used to build their corresponding 1080 PLSR models. All data, including the original calibration set, every built experimental calibration set and validation set (if available) and all generated model parameters are saved in a single data structure for easy post-processing access.

After model generation, the model parameters, including RMSE for calibration and validation, RMSECV, R^2 for calibration, validation and cross-validation models as well as all applied pre-processing methods are displayed in an easy-to-access table, together with the best 3 experimental calibration methods for every latent variable for comparison purposes. An example model output can be seen in Figure 3.13. This figure is a powerful tool to compare and understand the different calibration sets in conjunction with the component selection (Step 4) and total results outputs (Step 6).



Nr	RMSEV	RMSECV	R2v	R2cv	LV	Methods_Used	Baseline	Scatter	Smooth	Scaling
4	0.030987	0.045656	0.97992	0.98082	1	[2]	'dev2_ord2_9pt'	'no scatter'	'no smoothing'	'mean center'
3	0.042263	0.046184	0.96266	0.98038	1	[2]	'dev1_ord4_9pt'	'no scatter'	'no smoothing'	'mean center'
19	0.044039	0.072297	0.95945	0.95191	1	[3]	'dev2_ord2_9pt'	'no scatter'	'smooth_ord2'	'mean center'
1	0.089374	0.089373	0.833	0.92652	1	[1]	'no baseline'	'no scatter'	'no smoothing'	'Mean_Center'
14	0.020231	0.033712	0.99144	0.98954	2	[3]	'dev2_ord2_9pt'	'SNV'	'no smoothing'	'mean center'
7	0.022518	0.099128	0.9894	0.90967	2	[3]	'dev1_ord2_9pt'	'MSC'	'no smoothing'	'MSC mean'
8	0.025114	0.10486	0.98681	0.89901	2	[3]	'dev1_ord4_9pt'	'MSC'	'no smoothing'	'MSC mean'
1	0.032839	0.02518	0.97745	0.99417	2	[1]	'no baseline'	'no scatter'	'no smoothing'	'Mean_Center'
7	0.013949	0.019778	0.99593	0.9964	3	[3]	'dev1_ord2_9pt'	'MSC'	'no smoothing'	'MSC mean'
12	0.01493	0.0081266	0.99534	0.99939	3	[3]	'dev1_ord2_9pt'	'SNV'	'no smoothing'	'mean center'
8	0.015308	0.022028	0.9951	0.99554	3	[3]	'dev1_ord4_9pt'	'MSC'	'no smoothing'	'MSC mean'
1	0.030129	0.014827	0.98102	0.99798	3	[1]	'no baseline'	'no scatter'	'no smoothing'	'Mean_Center'
6	0.014531	0.014767	0.99559	0.99799	4	[2]	'no baseline'	'MSC'	'no smoothing'	'MSC mean'
12	0.014969	0.0076082	0.99532	0.99947	4	[3]	'dev1_ord2_9pt'	'SNV'	'no smoothing'	'mean center'
11	0.014973	0.012299	0.99531	0.99861	4	[2]	'no baseline'	'SNV'	'no smoothing'	'mean center'
1	0.062815	0.0088766	0.9175	0.99928	4	[1]	'no baseline'	'no scatter'	'no smoothing'	'Mean_Center'
6	0.014913	0.012918	0.99535	0.99846	5	[2]	'no baseline'	'MSC'	'no smoothing'	'MSC mean'
11	0.017932	0.0082242	0.99328	0.99938	5	[2]	'no baseline'	'SNV'	'no smoothing'	'mean center'
12	0.018271	0.0071912	0.99302	0.99952	5	[3]	'dev1_ord2_9pt'	'SNV'	'no smoothing'	'mean center'
1	0.069302	0.0083952	0.89959	0.99935	5	[1]	'no baseline'	'no scatter'	'no smoothing'	'Mean_Center'
11	0.019058	0.0076507	0.99241	0.99946	6	[2]	'no baseline'	'SNV'	'no smoothing'	'mean center'
13	0.019104	0.0077654	0.99237	0.99945	6	[3]	'dev1_ord4_9pt'	'SNV'	'no smoothing'	'mean center'
9	0.021528	0.01277	0.99031	0.9985	6	[3]	'dev2_ord2_9pt'	'MSC'	'no smoothing'	'MSC mean'
1	0.04604	0.0084849	0.95568	0.99934	6	[1]	'no baseline'	'no scatter'	'no smoothing'	'Mean_Center'
13	0.019009	0.0078598	0.99245	0.99943	7	[3]	'dev1_ord4_9pt'	'SNV'	'no smoothing'	'mean center'
9	0.02176	0.013305	0.9901	0.99837	7	[3]	'dev2_ord2_9pt'	'MSC'	'no smoothing'	'MSC mean'
15	0.022458	0.01499	0.98946	0.99793	7	[3]	'dev2_ord4_9pt'	'SNV'	'no smoothing'	'mean center'
1	0.05457	0.0089354	0.93774	0.99927	7	[1]	'no baseline'	'no scatter'	'no smoothing'	'Mean_Center'
13	0.019081	0.0082213	0.99239	0.99938	8	[3]	'dev1_ord4_9pt'	'SNV'	'no smoothing'	'mean center'
9	0.021318	0.013611	0.9905	0.9983	8	[3]	'dev2_ord2_9pt'	'MSC'	'no smoothing'	'MSC mean'
15	0.022203	0.015271	0.98969	0.99785	8	[3]	'dev2_ord4_9pt'	'SNV'	'no smoothing'	'mean center'
1	0.044343	0.0092861	0.95889	0.99921	8	[1]	'no baseline'	'no scatter'	'no smoothing'	'Mean_Center'

Figure 3.13: This table depicts 4 model results per number of latent variables used. This includes the three best results with pre-processing methods applied and a comparative result with no pre-processing methods applied. To compare the models RMSEV (if available, otherwise read as RMSEC), RMSECV, R2v and R2cv are visible to assess the model performance.

Using this table, direct number comparisons between the model performances can be viewed and the respective pre-processing methods used to achieve the result. Viewing the figure, it is discernible that the lowest possible model error is achieved with 3 latent variables at 1.4% RMSEV combination Nr7. This combination employs a second order first derivative with multiple scatter correction. As the scatter correction already uses mean-centring this is reflected in the methods used.

Step 4: Component Selection

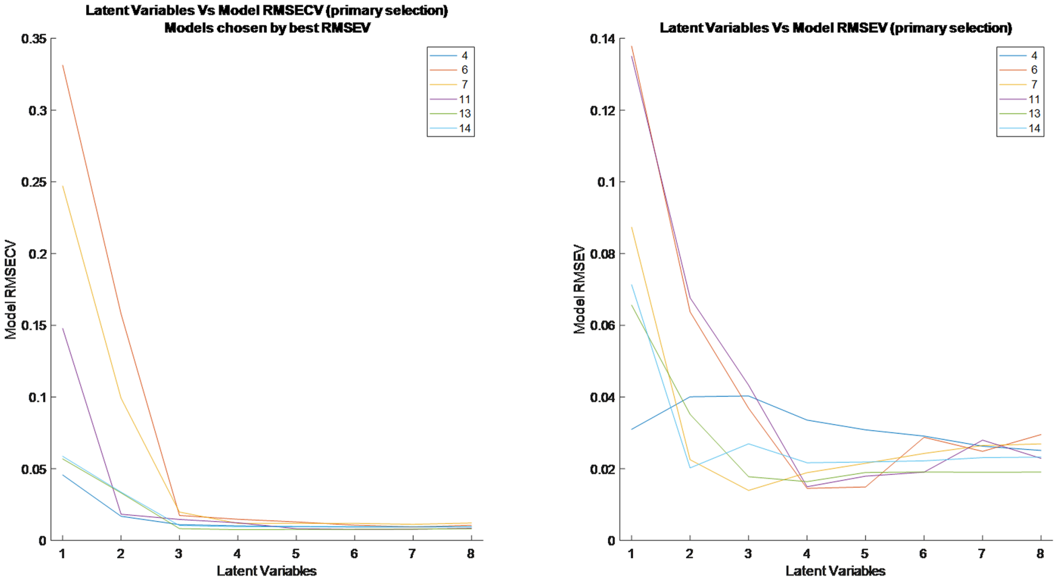


Figure 3.14: Best models selected from all latent variables and model RMSECV / validation plotted against number of latent variables.

The simplest way to understand the number of latent variables necessary to model the underlying changes in the data is to plot the RMSE of the model versus the number of latent variables. The toolbox offers two plots to compare these values: the RMSECV versus LVs plot and the RMSEC versus LVs. If a validation data set is available, the algorithm will automatically select the RMSEV instead of the RMSEC, which is more informative. With an increasing number of latent variables, more variation in the data is described in the model. In other words, the less complex the sample system the less latent variables are required to be able to predict a response to the spectral input. As previously stated, the experimental data used for this example models the interaction between a two-component system, which is not very complex.

Viewing Figure 3.14 the simplicity of the system is apparent in the low number of LVs required to achieve a very low (below 5% / RMSE below 0.05) prediction error in the model. The left plot shows that most models do not require more than 2 to 3 latent variables to capture the most important changes in the system and then plateau. From this point on, any further increase in latent variables only adds to overfitting the model to the training data. This claim is solely based on looking at the RMSECV plot. Comparing the RMSECV and RMSEV plots we can see that some of the models struggle to deliver the previous results with real world data while others hold their ground. Without a validation data comparison this kind of assessment is not possible. Finally, it could be established that the model should not use more than 2 or 3 LVs due to the mentioned overfitting and no real gain being made from an increase in complexity. The final calibration model choice can only be made later with knowing the background for the future application of the models and the amount of pre-processing required to get these calibration results. Less pre-processing is usually preferred.

Step 5: Interaction and Main Effects Plots

An interaction of two variables is defined as the change of one variable affecting the level of another. The tool uses interaction plots to describe how different pre-processing step interactions influence the RMSEV of the built models. It helps to dissect what kind of pre-processing is necessary and deepens the understanding of the data by visualizing which methods support each other or are counterproductive[90]. This is achieved by plotting the different steps against each other and using the mean RMSEV established from all models used with the respective methods as an indicator for changes between the methods applied. The column of the pre-processing steps indicates which step is currently depicted as the x-axis. For example, row1-column1 would show the RMSEV mean for all baseline correction steps versus itself. Moving down one row, row2-column1, shows the various baseline corrections steps (x-axis) and how they interact with the various scatter correction methods (y-axis) applied to the data while row1-column2 displays the same interactions with the scatter corrections methods as the x-axis. These results have to be taken as an indicator and not a final result as they are influenced by all models and as such only indicate possible best relationships.

A brief summary of the interaction plot in Figure 3.15 for each of the pre-processing steps shows that baseline correction method 2 (first derivative with a 4^{th} order polynomial), scatter correction method 2 (MSC), smoothing method 0 (no smoothing), scaling method 0 (only mean-centering) and a latent variable number between 3 and 4 had the best method interactions with the lowest RMSEV possible. The results show that smoothing has no positive effect on the data, suggesting low noise in our original data. Furthermore, mean-centering is the only scaling and centering step that is useful with both other methods being largely detrimental to the RMSEV. Comparing the scatter correction data (column 2 in the Figure 3.15), no large differences between the methods are visible suggesting that no scatter correction is required. A general slight improvement of RMSEV could be attributed to the additional mean-centering effects of the MSC method but they do not appear to be significant.

To visualize the usefulness of the interaction plot in more detail, Figure 3.16 shows a zoom-in on the number of latent variables versus baseline graph (row1-column5) from Figure 3.15. Here, details about the number of latent variables required, the usefulness of applying baseline correction past a certain number of latent variables and the best available baseline correction method can be extracted.

Firstly, viewing the RMSEV for baseline corrections with one LV it is apparent that the application of baseline correction methods coincides with a reduction in model error. Thus, it can be safely assumed that some baseline problems are present in the data and or that the modelling algorithm prefers the corrected spectra, i.e. that the complexity of the data was reduced without requiring a larger number of latent variables to do so. Secondly, using the first derivative is preferred and only minor differences between the application of the second and fourth polynomials are found. The fourth polynomial is preferred with a lower number of LVs while the second polynomial is preferred with a higher number of LVs, meaning that spectral noise is accounted for after LV 2 and is increased using a higher number of polynomials from here on.

Furthermore, it is visible that from LV 4 onwards the baseline error appears to be incorporated in the model and only slight improvements are made still applying any baseline correction method while the overfitting of the data increases the model error. Lastly, this graph reconfirms previous assessments regarding the model complexity required to model the process at hand, showing that going beyond 3 LVs increases model error likely due to overfitting of the data.

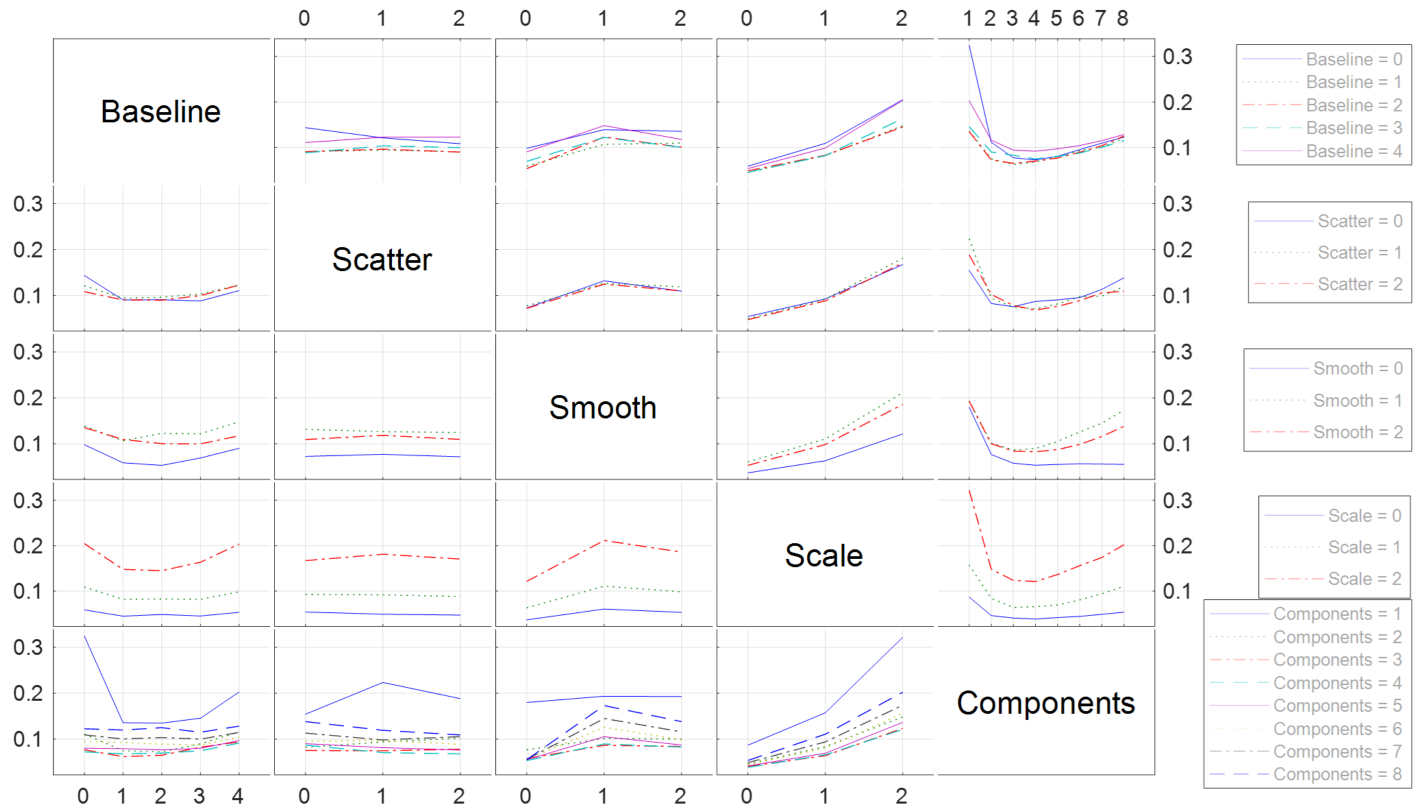


Figure 3.15: Interaction plot describing pre-processing method interactions as RMSEV. The numbers to the individual methods can be found in 3.1, Each column describes the displayed pre-processing step as the x-axis. Moving rows up and down describes the mean model results of the employed pre-processing step interactions as RMSEC (e.g. row-1 column-5 describes the baseline corrections methods (x-axis) vs the number of latent variables (y-axis)).

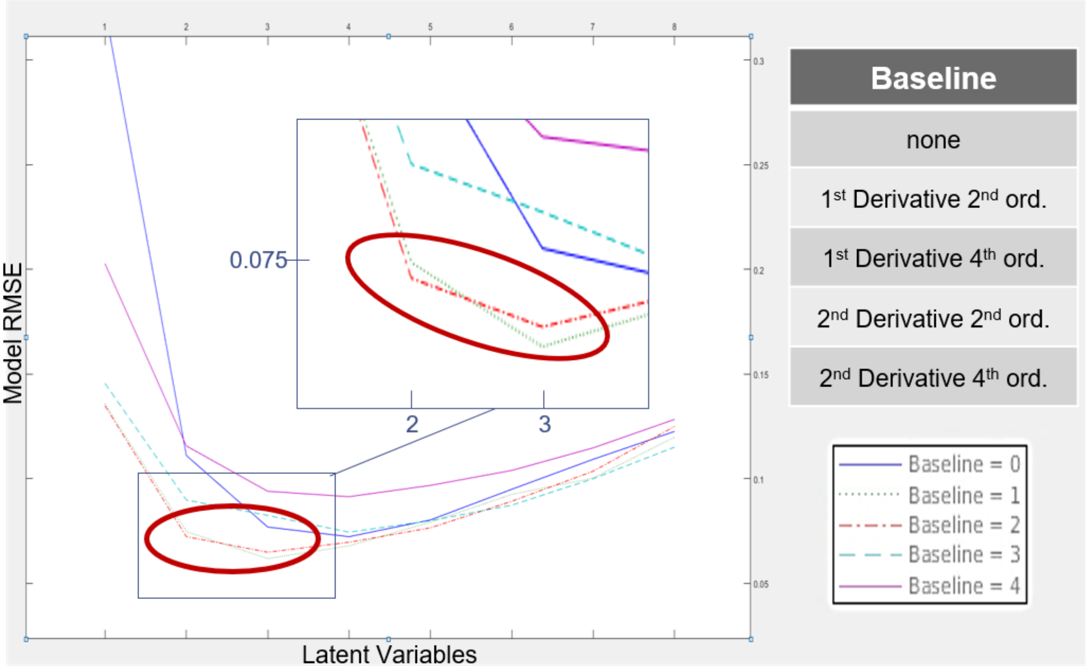


Figure 3.16: Interaction plot describing pre-processing method interactions of the number of latent variables based on varying baseline correction methods as the RMSEV.

An addition to the interaction plots is the main effects plot which averages the effect of all variable levels (pre-processing methods) in a pre-processing step and shows which one method per step demonstrates the maximum effect with regards to the data analysed. Using this plot, an additional layer of information is given to allow an assessment to be made about whether a specific pre-processing method during one of the pre-processing steps was more effective than others. Care must be taken though as it does not automatically equate to the best pre-processing method available as it is an average effect of all models[90].

Figure 3.17 depicts the main effects plot showing similar results to the previous findings. Baseline corrections methods 1 and 2 are very similar, but on the total average of all methods the first derivative with a second order polynomial performed better. Scatter correction, smoothing and scaling confirm the previously observed trends seen in Figure 3.15, while it seems that for a larger number of models 4 latent variables was the optimal number.

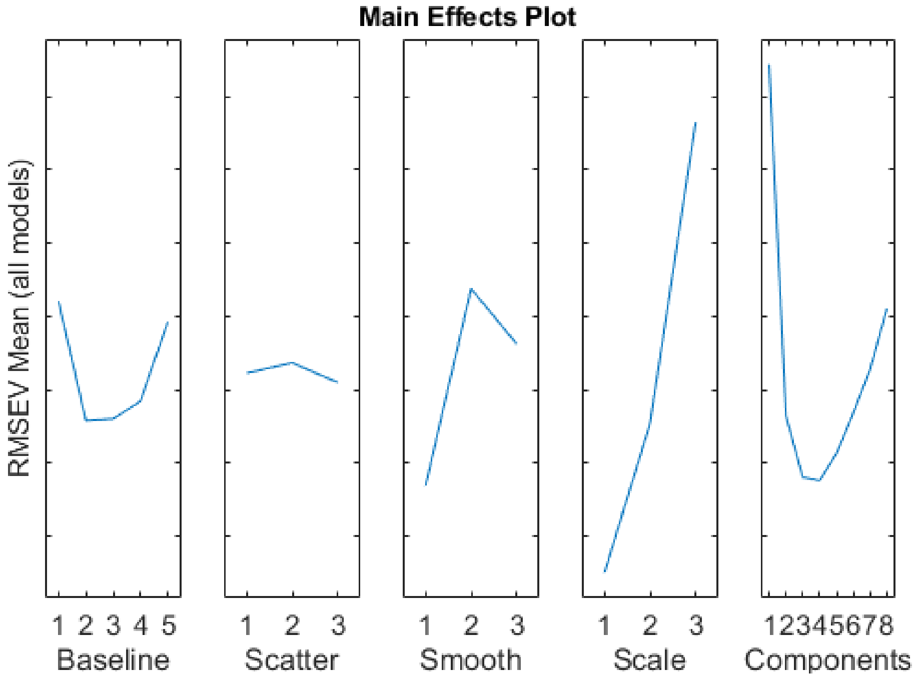


Figure 3.17: Main effects plot describing the averaged effect of the pre-processing methods applied on all models.

Step 6: Final Results Plot

Figure 3.18 depicts the final results plot, an accumulation of all experimental calibration model RMSEC results versus the number of latent variables (RMSEV is used if a validation set was provided). The best three results per number of latent variables are named and coloured showing the number of pre-processing steps and methods used, a comparison bare-bones processing sample with just mean-centring is displayed in black (number 1) with all other computed results being shown in grey. This image is interactive in the toolbox and can be zoomed-in and viewed in more detail if necessary.

This graph is a visual summary of the tabular output shown in Figure 3.13. It quickly summarises and focusses the results of the table while displaying all additionally calculated results.

Without longer analysis and no background knowledge required it is visible that the optimum models for this process should use 3 latent variables and required 3 pre-processing steps. The results with four latent variables come within 0.5 RMSEV of

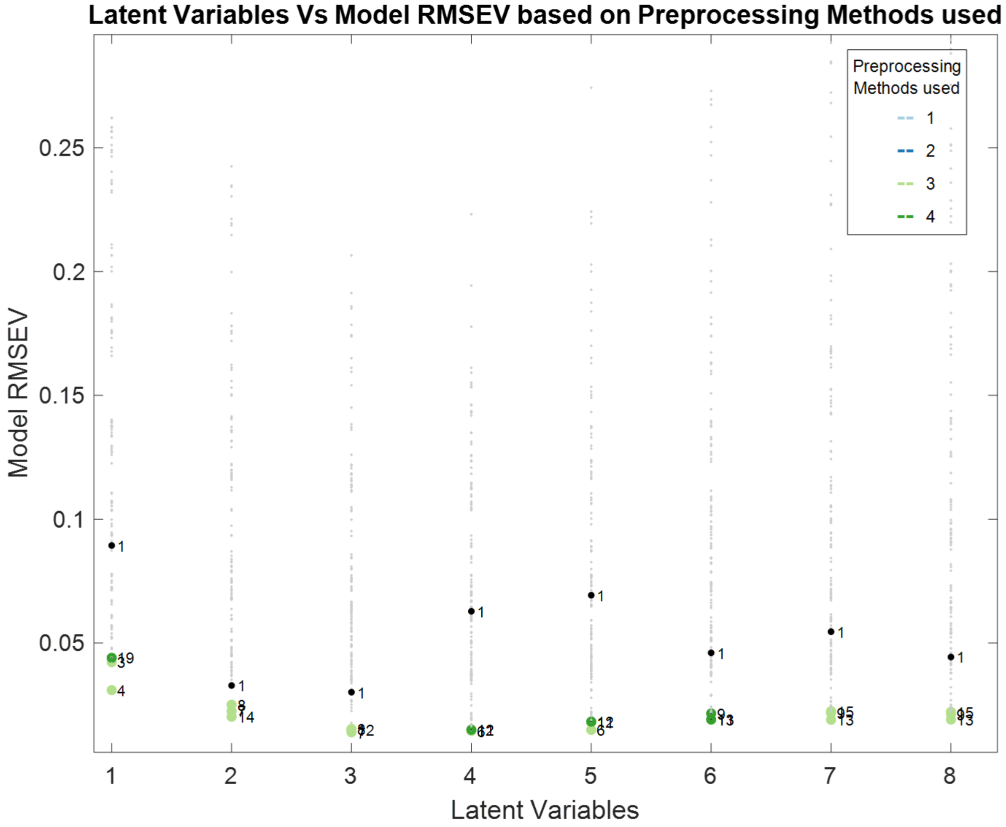


Figure 3.18: Model RMSEV of all models VS number of latent variables. Shows best three models (coloured by number of steps used), models with minimal pre-processing (black, only mean-centring) and all other results (in grey).

the best 3 LV solution while only using 2 pre-processing steps. If a lower number of latent variables is required for the process, with minor losses in accuracy, also two latent variables are acceptable. Using the toolbox for exploratory analysis and model creation the model error decreased from bare-bones processing values of 3.0 RMSEV to 1.4 RMSEV (baseline correction using 1st derivative 2nd order polynomial in conjunction with multiple scatter correction (mean-centring is implied in MSC), effectively halving the error. These RMSEV results have been taken from the toolbox table output in Figure 3.13.

3.4 Use Case #2: Pharmaceutical Tablet Analysis Using NIR

3.4.1 Experimental Background

Experimental data for use case number 2 was taken from an open data set by Dyrby et al. (2002)[91]. In their paper the authors assess the content uniformity of pharmaceutical tablets through the use of NIR and Raman spectroscopy surface measurements. Content uniformity tests are used to assure the quality of drug product batches and thus ensure that the strength of the active pharmaceutical ingredient within the dosage form is compliant with acceptance and efficacy limits. The open data offered is based on the following experimental method provided by Dyrby et al.[91].

The target tablets used for this study were Escitalopram tablets produced by H. Lundbeck A/S and were supplied in four dosage strengths (5, 10, 15 as well as 20 mg tablet weight). Due to the different active pharmaceutical ingredient (API) content, the general weight, shape, and size of the four categories differed. Unlike the 5mg tablets, tablets of sizes 10, 15 and 20 mg were dose-proportional, meaning their total weight was proportionally larger to the amount of API content added.

As described in Table 3.4, tablets of types 10, 15 and 20 mg were produced in 8 batches while tablets of the 5 mg type only had 7 batches produced. For each batch, 10 tablets were individually weighed out and used for analysis, resulting in a total of 310 samples based on 31 batches.

Materials and Software

The authors recorded NIR transmittance spectra in the range of 4000-14000 cm^{-1} with a resolution of 16 cm^{-1} and an average of 128 scans per sample. The spectrometer used was an ABB Bomem FT-NIR MB-160 with an attached tablet sampling device running an InGaAs 1.7µm detector. Using an internal ceramic standard (Spectralon 99%), background transmittance spectra were recorded and used to convert the acquired spectra of the tablets into absorbance units[91]. For the Raman measurements, Dyrby et al. state that during their model building, issues were identified with the sampling

Table 3.4: Table describing the various produced sample batches and their properties per tablet type. The table was reproduced from the original publication by Dyrby et al.[91].

Nominal content of active substance per tablet (mg)	Nominal tablet weight (mg)	Nominal weight percent (%)	Number of batches
5	90	5.6	1 full and 3 pilot scale
10	125	8	2 full and 3 pilot scale
15	188	8	2 full and 3 pilot scale
20	250	8	2 full and 3 pilot scale
4.3-5.7	90	4.8-6.3	3 laboratory scale
8.3-11.4	125	6.9-9.1	3 laboratory scale
12.9-17.1	188	6.9-9.1	3 laboratory scale
17.3-22.8	250	6.9-9.1	3 laboratory scale

performance. Therefore, these results will not be included in this study. Furthermore, in addition to using NIR and Raman, the paper stated that high performance liquid chromatography (HPLC) was applied as a reference method to measure the API content of the tablets, which was given as the weight in milligrams of Escitalopram per tablet. Using the individual tablet weights, a weight percent was calculated (%w/w)[91]. The open data set does not include any HPLC data or the individual tablet weights but states each tablet's %w/w.

The original model building and analysis in 2002 employed Unscrambler version 7.5 (CAMO A/S, Trondheim, Norway) for PLS model building and pre-processing. For further analysis 'MATLAB version 5.3 (MathWorks, Inc., Natick, MA) with PLS_Toolbox version 2.0 (Wise and Gallagher, Eigenvector Research, Manson, WA) and The Graphical User Interface iPLS Toolbox for MATLAB version 2.1 (www.models.kvl.dk/source/) were used for iPLS calculations. MATLAB was also used for pre-processing with SNV[91].

Data

Data provided in the online repository contained a total of 310 tablet spectra depicted as rows in a matrix. The first 3 columns describe the %weight/weight, the tablet type (i.e. 5, 10, 15 and 20 mg respectively) as well as the batch number the samples were taken from. In addition, each row contains the NIR spectrum per sample from wavelengths 7400 to 10500 cm^{-1}. While the original analysis contained spectra from 4000 to 14000 cm^{-1}, Dyrby et al. removed parts outside of the 7400 - 10500 cm^{-1} range due to noise concerns and redundancy. Sadly, no specific information has been provided on which samples were used for model generation and validation. As such, in our analysis, for each tablet type, the samples in the data matrix were sorted by ascending nominal %weight/weight content. Following, every 4th sample row was selected as a test sample, with the remainder of the data set being used for calibration purposes. This resulted in an even spread of validation samples across the various batches provided.

3.4.2 Results

The paper offered 4 tablet types and their different model outputs. This result section will focus on the results based on the 5 mg tablet type and summarise the model results of the 10, 15 and 20 mg tablet types at the end of Section 3.4.2.

Step 0: Data Import

70 available samples for the 5 mg tablet type were separated into 53 calibration and 17 test samples and loaded into the toolbox using the %w/w values of the tablets as the Y variable. The 10, 15 and 20 mg tablet types each had 80 samples and were thus split into 60 calibration and 20 test samples.

Step 1: Outlier Detection

Figure 3.19 depicts the outlier detection results for the 5 mg tablet type. As a reminder, the Monte Carlo outlier detection method builds 1000 models splitting the calibration samples into test groups and recording a mean prediction and standard

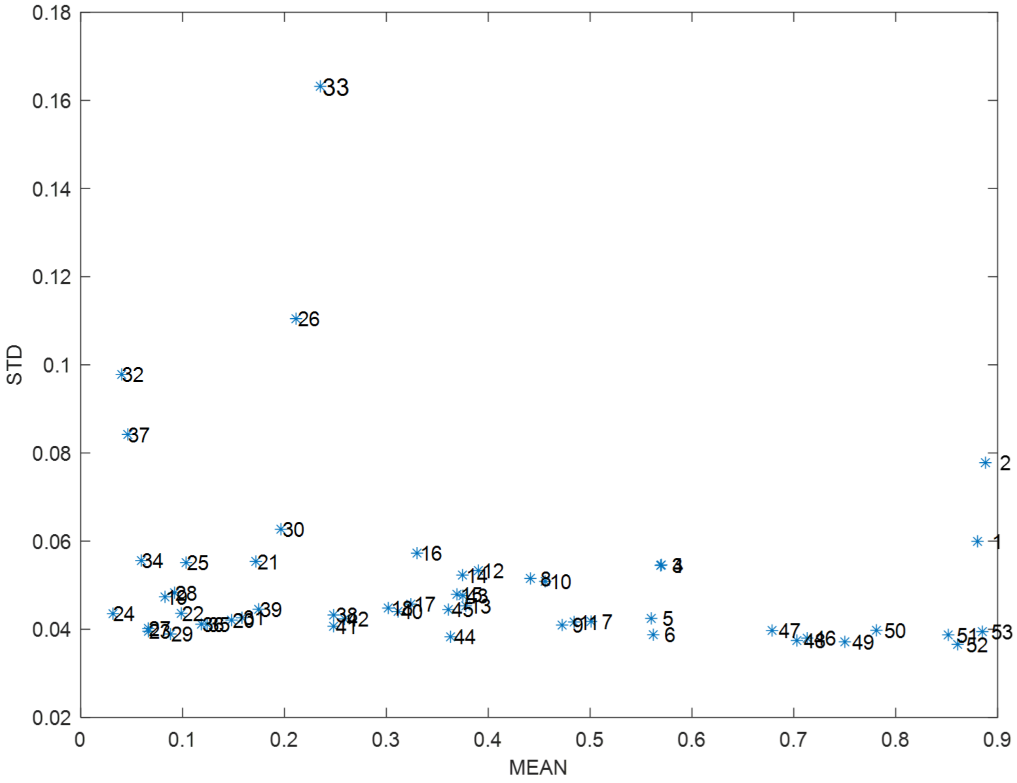


Figure 3.19: Outlier detection for the 5 mg tablet type showing the 53 calibration samples

deviation prediction error for each sample. The sample standard deviation prediction errors are plotted against the sample mean prediction errors. In this use case, samples of a similar batch should band together as the deviations from the sensor reading would likely be similar while a large spread of %w/w content values will likely lead to a spread of predicted values in general. Suspicious samples are numbers 1, 2, 26, 32, 33 and 37. Sample 33 in particular shows a high standard deviation prediction error. Looking at the spectrum in Figure 3.20 the sample appears to have lower intensity values compared to similarly concentrated samples. Interestingly, in this particular set of samples the spectra intensity does not increase linearly with increasing %w/w which appears to be partly due their different batch numbers. In addition, it is also worth noting that Samples 32 and 33 (bottom two samples in Figure 3.20 are from the same batch, while samples 31, 34 and 35 (top three samples in Figure3.20 are from another one. While sample 33 does appear to be suspicious due to its lower intensity, it is not entirely clear whether it is batch related or due to other reasons. Typically, this would be cause for further investigation into the data to understand the origin of this issue but as no further information is available in this case, the sample will not be removed. Similarly to sample 33, the other identified potential outliers did not appear suspicious enough when comparing their spectra and as such were left in the calibration set.

No removable outliers were found for the 10, 15 and 20 mg tablet data sets and as such, these results are not shown here.

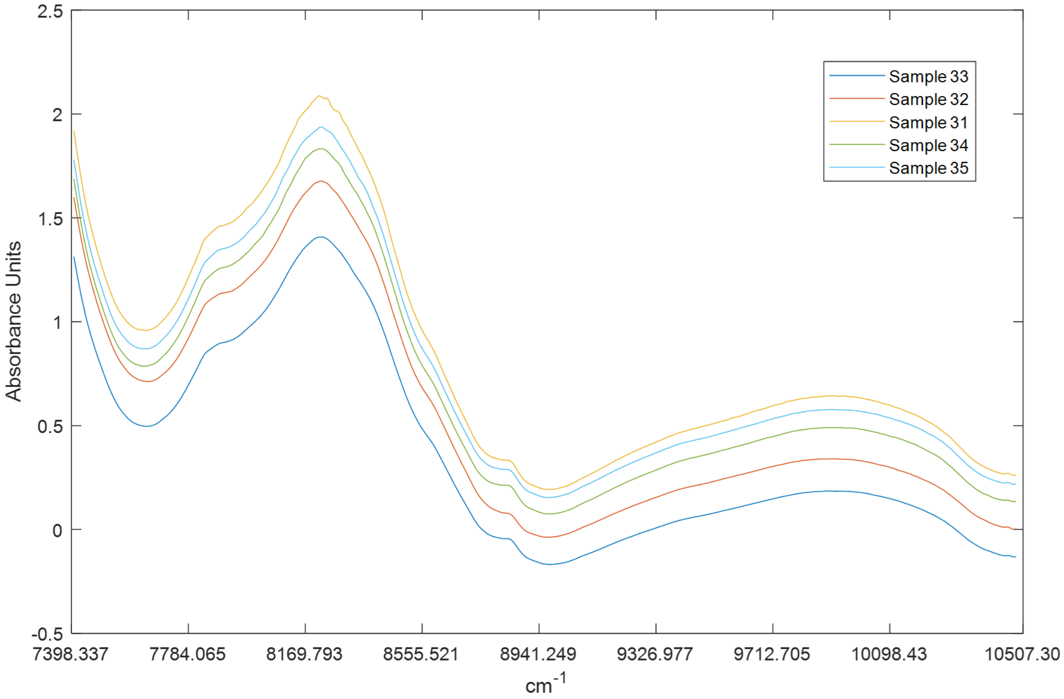


Figure 3.20: NIR spectra of calibration samples with similar %w/w as calibration sample 33. Samples 31 to 35 have increasing %w/w content ranging from 5.47 to 5.50 %w/w. While samples 32 and 33 stem from one batch, samples 31, 34 and 35 stem from another.

Step 2: Pre-Processing Method Selection and DoE Application

Dyrby et al.[91] applied various pre-processing methods to test their models, including derivatives, smoothing and scatter correction, though not in many combinations and variations. To test the toolbox, all potential pre-processing steps and methods available from the toolbox were tested. In their paper the authors found their ideal number of latent variables to be between 2 and 4. To image the full range of model responses 8 latent variables were chosen for this case. The same method was applied for all 4 tablet data sets.

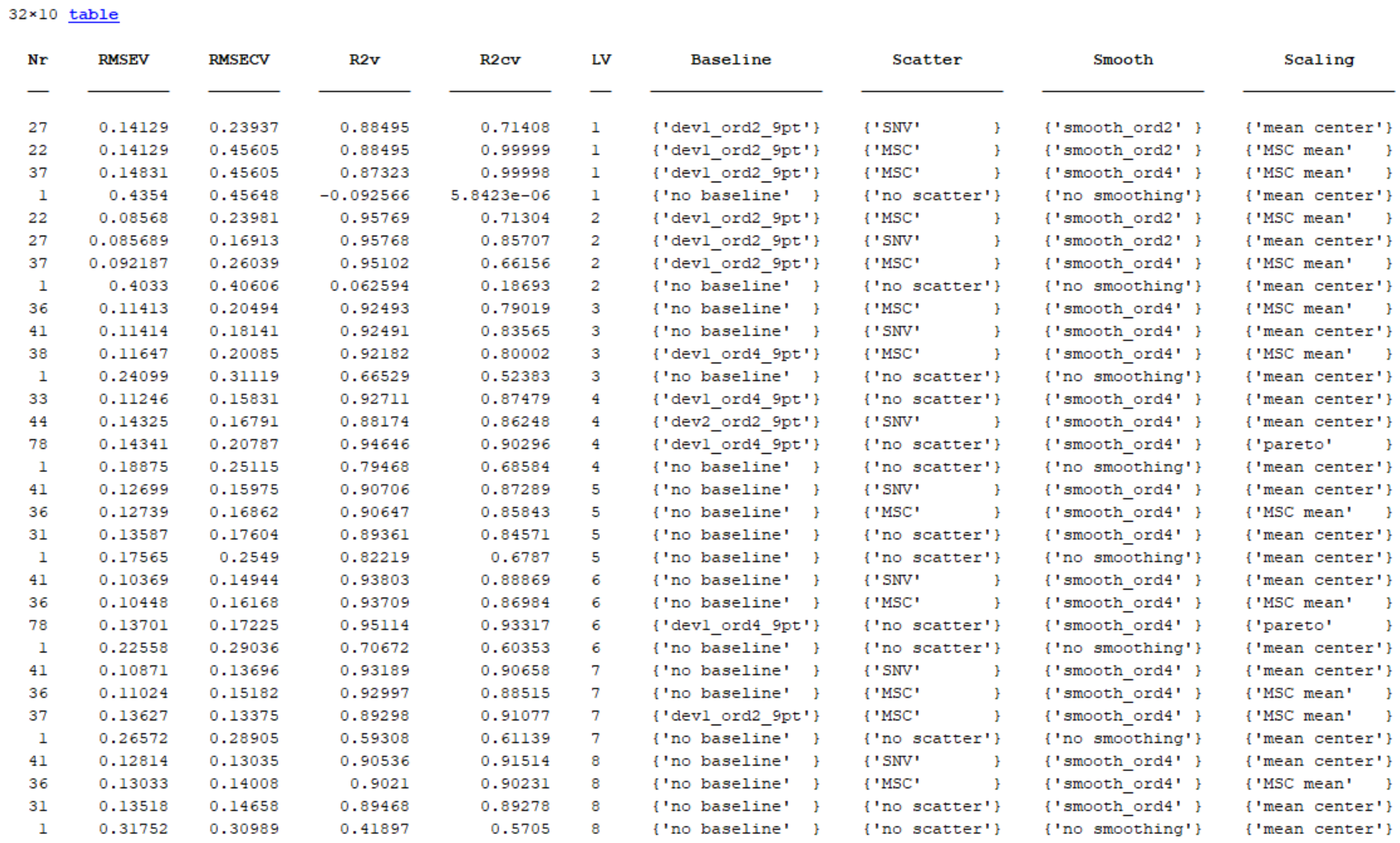
Step 3: PLS Model Building

With similar testing parameters to the previous use case, 1080 different calibration sets were generated in this use-case as well. This is based on 135 pre-processing method combinations multiplied by the number of tested latent variables. Figure 3.21 depicts four model results per number of latent variables used from the generated results table. This includes the three best results with applied pre-processing methods and a comparative result with no pre-processing methods applied apart from mean-centring the data. To compare and assess the model performance, RMSEV, RMSECV, R2v and R2cv are displayed. As the data was split into calibration and validation sets, the RMSEC is not used.

Viewing the 5 mg tablet data set results shown in Figure 3.21 and comparing the general best outcomes for each latent variable, the best 1st and 2nd latent variable models are achieved using model numbers 22, 27 and 37 which all employ a mix of first derivative baseline corrections, scatter corrections methods as well as smoothing and mean-centring (if not applied by the scatter correction method). For models with a 3rd latent variable or more, the preferred model numbers show more variation. While the majority of those models still employ a variety of smoothing, scatter correction and mean centring, interestingly baseline correction models are less successful at a higher number of latent variables.

The best performing models are found with 2 latent variables. The three best models are very similar in predictive performance, giving between 0.089 and 0.092 RMSEV.

Viewing their RMSECV though, they all appear to have significantly higher values, between 0.17 and 0.26 RMSECV. Model Nr. 27 shows the overall best result with 0.089 RMSEV and 0.17 RMSECV and closeness of fit values for r2v and r2cv of 0.95 and 0.85, respectively. Dyrby et al. used the RMSECV and RMSEV values to calculate relative prediction errors[91]. These were defined by dividing the respective root mean square error values by the nominal %w/w content value for the tablet type tested and multiplying the resulting value by 100. Performing this calculation for model Nr. 27, relative prediction errors of 1.59% and 3.04% for the RMSEV and RMSECV respective values are achieved. Model Nr. 27 employed a first order derivative with a second order polynomial baseline correction, standard normal variate scatter correction and a second order polynomial Savitzky-Golay smoothing filter.



```
32×10 table

Nr    RMSEV      RMSECV       R2v          R2cv     LV       Baseline              Scatter              Smooth               Scaling
__    ________   _______    __________   __________   __    ___________________   _________________   ___________________   _________________

27     0.14129   0.23937       0.88495      0.71408    1    {'dev1_ord2_9pt'}    {'SNV'       }    {'smooth_ord2' }    {'mean center'}
22     0.14129   0.45605       0.88495      0.99999    1    {'dev1_ord2_9pt'}    {'MSC'       }    {'smooth_ord2' }    {'MSC mean'   }
37     0.14831   0.45605       0.87323      0.99998    1    {'dev1_ord2_9pt'}    {'MSC'       }    {'smooth_ord4' }    {'MSC mean'   }
 1      0.4354   0.45648     -0.092566   5.8423e-06    1    {'no baseline'  }    {'no scatter'}    {'no smoothing'}    {'mean center'}
22     0.08568   0.23981       0.95769      0.71304    2    {'dev1_ord2_9pt'}    {'MSC'       }    {'smooth_ord2' }    {'MSC mean'   }
27    0.085689   0.16913       0.95768      0.85707    2    {'dev1_ord2_9pt'}    {'SNV'       }    {'smooth_ord2' }    {'mean center'}
37    0.092187   0.26039       0.95102      0.66156    2    {'dev1_ord2_9pt'}    {'MSC'       }    {'smooth_ord4' }    {'MSC mean'   }
 1      0.4033   0.40606      0.062594      0.18693    2    {'no baseline'  }    {'no scatter'}    {'no smoothing'}    {'mean center'}
36     0.11413   0.20494       0.92493      0.79019    3    {'no baseline'  }    {'MSC'       }    {'smooth_ord4' }    {'MSC mean'   }
41     0.11414   0.18141       0.92491      0.83565    3    {'no baseline'  }    {'SNV'       }    {'smooth_ord4' }    {'mean center'}
38     0.11647   0.20085       0.92182      0.80002    3    {'dev1_ord4_9pt'}    {'MSC'       }    {'smooth_ord4' }    {'MSC mean'   }
 1     0.24099   0.31119       0.66529      0.52383    3    {'no baseline'  }    {'no scatter'}    {'no smoothing'}    {'mean center'}
33     0.11246   0.15831       0.92711      0.87479    4    {'dev1_ord4_9pt'}    {'no scatter'}    {'smooth_ord4' }    {'mean center'}
44     0.14325   0.16791       0.88174      0.86248    4    {'dev2_ord2_9pt'}    {'SNV'       }    {'smooth_ord4' }    {'mean center'}
78     0.14341   0.20787       0.94646      0.90296    4    {'dev1_ord4_9pt'}    {'no scatter'}    {'smooth_ord4' }    {'pareto'     }
 1     0.18875   0.25115       0.79468      0.68584    4    {'no baseline'  }    {'no scatter'}    {'no smoothing'}    {'mean center'}
41     0.12699   0.15975       0.90706      0.87289    5    {'no baseline'  }    {'SNV'       }    {'smooth_ord4' }    {'mean center'}
36     0.12739   0.16862       0.90647      0.85843    5    {'no baseline'  }    {'MSC'       }    {'smooth_ord4' }    {'MSC mean'   }
31     0.13587   0.17604       0.89361      0.84571    5    {'no baseline'  }    {'no scatter'}    {'smooth_ord4' }    {'mean center'}
 1     0.17565    0.2549       0.82219       0.6787    5    {'no baseline'  }    {'no scatter'}    {'no smoothing'}    {'mean center'}
41     0.10369   0.14944       0.93803      0.88869    6    {'no baseline'  }    {'SNV'       }    {'smooth_ord4' }    {'mean center'}
36     0.10448   0.16168       0.93709      0.86984    6    {'no baseline'  }    {'MSC'       }    {'smooth_ord4' }    {'MSC mean'   }
78     0.13701   0.17225       0.95114      0.93317    6    {'dev1_ord4_9pt'}    {'no scatter'}    {'smooth_ord4' }    {'pareto'     }
 1     0.22558   0.29036       0.70672      0.60353    6    {'no baseline'  }    {'no scatter'}    {'no smoothing'}    {'mean center'}
41     0.10871   0.13696       0.93189      0.90658    7    {'no baseline'  }    {'SNV'       }    {'smooth_ord4' }    {'mean center'}
36     0.11024   0.15182       0.92997      0.88515    7    {'no baseline'  }    {'MSC'       }    {'smooth_ord4' }    {'MSC mean'   }
37     0.13627   0.13375       0.89298      0.91077    7    {'dev1_ord2_9pt'}    {'MSC'       }    {'smooth_ord4' }    {'MSC mean'   }
 1     0.26572   0.28905       0.59308      0.61139    7    {'no baseline'  }    {'no scatter'}    {'no smoothing'}    {'mean center'}
41     0.12814   0.13035       0.90536      0.91514    8    {'no baseline'  }    {'SNV'       }    {'smooth_ord4' }    {'mean center'}
36     0.13033   0.14008        0.9021      0.90231    8    {'no baseline'  }    {'MSC'       }    {'smooth_ord4' }    {'MSC mean'   }
31     0.13518   0.14658       0.89468      0.89278    8    {'no baseline'  }    {'no scatter'}    {'smooth_ord4' }    {'mean center'}
 1     0.31752   0.30989       0.41897       0.5705    8    {'no baseline'  }    {'no scatter'}    {'no smoothing'}    {'mean center'}
```

Figure 3.21: Result table for the 5 mg tablet type depicts four model results per number of latent variables used. This includes the three best results with pre-processing methods applied and a comparative result with no pre-processing methods applied. To compare the models, RMSEV, RMSECV, R2v and R2cv are visible to assess the model performance.

Step 4: Component Selection

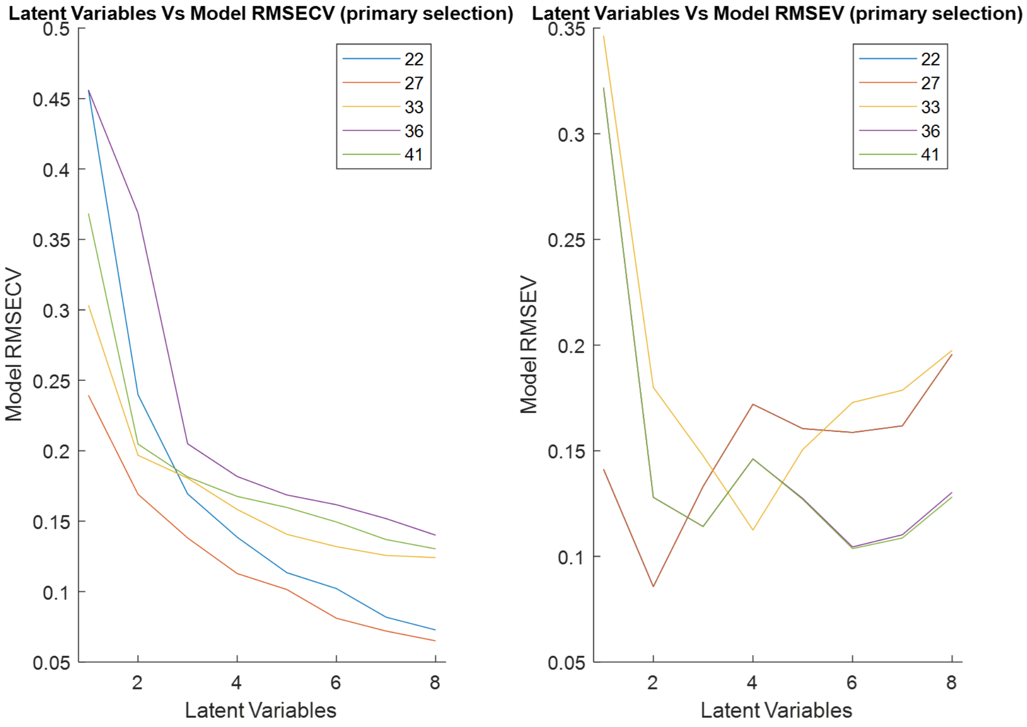


Figure 3.22: Best models selected from all latent variables and model RMSECV (left) and RMSEV (right) plotted against the number of latent variables.

Figure 3.22 depicts the component selection graphs for the 5 mg tablet results. Unlike the previous use case, a larger variation between the models can be seen for both the RMSECV and RMSEV results. Neither plot illustrates a steady low plateau with an increasing number of latent variables. While the RMSECV-based results are all signalling a slow decline in model error, the opposite is the case for the test data results. Here, varying for the different models after the 2nd to 4th latent variable, model errors start rising again, implying the models have been overfitted based on the original calibration set. This confirms that, depending on the model, 2 to 4 latent variables should be preferred.

Step 5: Interaction and Main Plots

Figure 3.24 shows the interaction plots for the 5 mg tablet results. Due to the large amount of data available only a few examples of interesting interactions will be described. Again, the interaction plot attempts to describe the different interactions between the applied pre-processing methods based on the RMSEV of the models produced. Starting with the baseline vs number of components plot (row1-column5) it is visible that baseline correction methods 1 and 2 are preferred (first derivative with 2$^{\text{nd}}$ and 4$^{\text{th}}$ order polynomials, respectively) with a lower number of latent variables employed. Once 5 latent variables are employed, not using any kind of baseline correction method leads to the best model result. This could suggest that from this point on the baseline is corrected by the model and using additional correction factors does not improve the prediction. Looking at the same graph past 5 latent variables however, the model appears to be overfit, gaining in RMSEV with every increase in the number of latent variables. With an increased number of latent variables, small errors and calibration sample specific features are incorporated into the model, making it very specific to the calibration data. Thus, with an increasing number of latent variables the validation data does not fit the model description showing that the model is overfit.

Interestingly, moving to row3-column5, this overfitting behaviour is particularly visible when no smoothing is applied to the data. This suggests that after 3 latent variables, with no smoothing applied, the model starts describing noise and spectral roughness as features of interest thus improving with more latent variables when these features are removed. Thus, a stronger smoothing effect, as applied through a higher order of polynomial, appears to be more effective as can be seen throughout row3. The main effects plot summarises and averages the general findings from all models. It is important to note that the numbering of the different methods is different here with the case of ‘no pre-processing’ now depicted by the number 1 (instead of 0 as in Fig. 3.24), with all other pre-processing method numbers also moving up by 1.

Viewing the main effects plot in Figure 3.23, the previously described effects are visible again. Baseline correction via a first derivative is most effective (2 and 3 in baseline) as well as a 4$^{\text{th}}$ order polynomial (3 in smooth) for smoothing. Due to the granular, rough and solid surface, light scattering appears to be a problem within these samples and SNV (1 in scatter) as well as MSC (2 in scatter) seem to be effective, with SNV showing generally better results. On the other hand, scaling seems entirely ineffective for the sample set at hand.

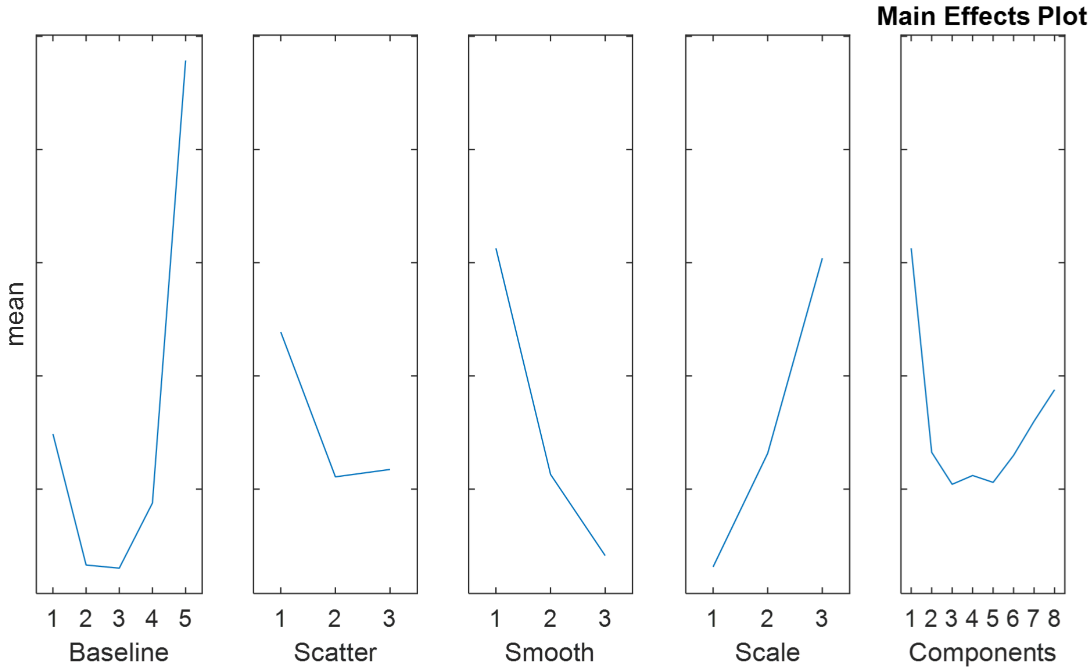


Figure 3.23: Main effects plot describing the averaged effect of the pre-processing methods applied on all models.

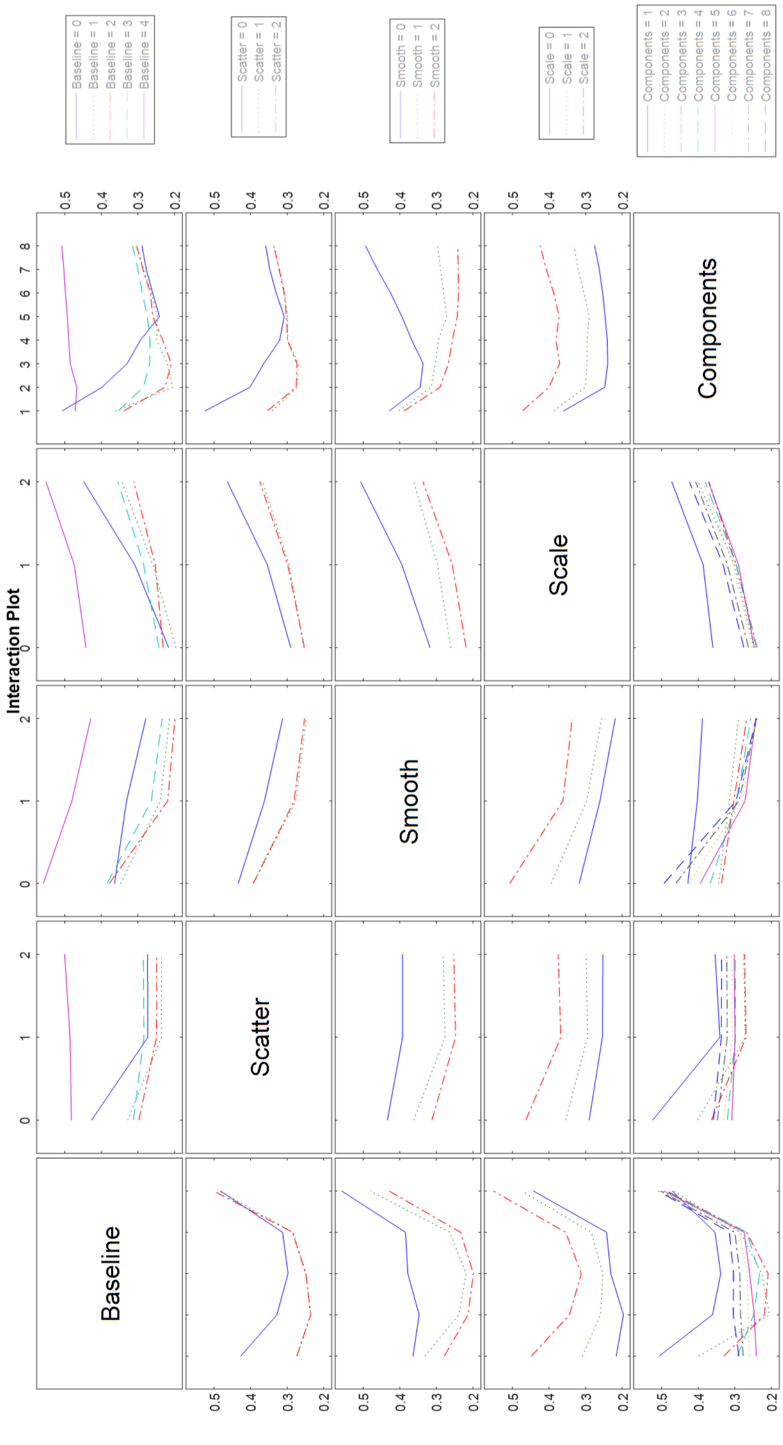


Figure 3.24: Interaction plot describing pre-processing method interactions as RMSEV. The numbers to the individual methods can be found in Table 3.1. Each column describes the displayed pre-processing step as the x-axis. Moving rows up and down describes the mean model results of the employed pre-processing step interactions as RMSEC (e.g. row-1 column-5 describes the baseline corrections methods (x-axis) vs the number of latent variables (y-axis).

Step 6: Final Results Plot

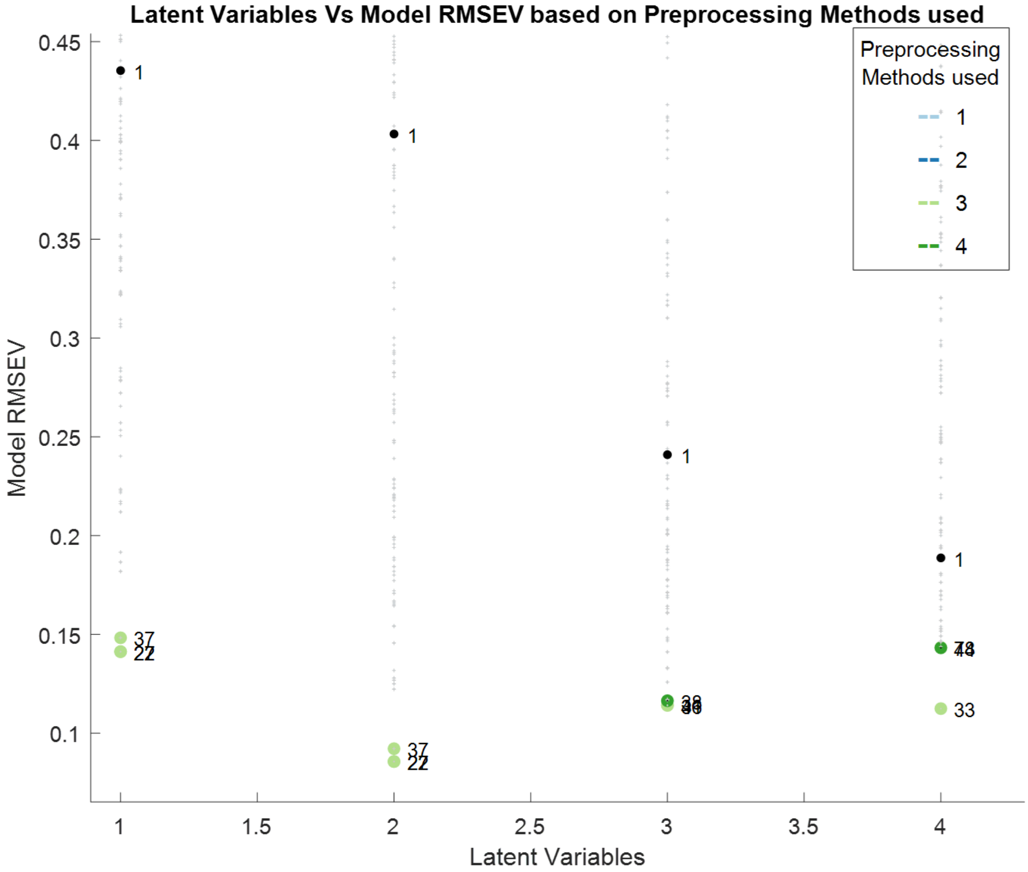


Figure 3.25: Model RMSEV of all models VS the number of latent variables. The plot shows the best three models (coloured by number of pre-processing methods used), models with minimal pre-processing (black, only mean-centring) and all other results (small grey dots).

Figure 3.25 depicts the final results plot of the 5 mg tablet test case, an accumulation of all experimental calibration model RMSEV results versus the number of latent variables. As before, the best three results per number of latent variables are named and coloured showing the number of pre-processing steps and methods used, a comparison bare-bones processing sample with just mean-centring is displayed in black (number 1) with all other computed results being shown in grey. Overlapping features are difficult to circumvent on a static graph when models show very similar performance. The interactive output in the toolbox can be zoomed-in and viewed in more detail if necessary and thus avoids this issue.

Viewing the results in Figure 3.25, all model outcomes are visible and depict a wide range of performance. Unlike the previous case example, samples without pre-processing fare significantly worse until a larger number of latent variables is employed (model number 1). While the best pre-processing method combinations all demonstrate results below 0.15 RMSEV, none of the mean-centred only samples manage to achieve such values. As previously seen, it is evident that the models with two latent variables are the best-performing and appear very close in performance. Interestingly, the best 2 models up to two latent variables perform significantly better than next best other results at the same number of latent variables.

Additional Tablet Type Results

Equal methods were applied for the 10, 15 and 20 mg tablet types. Table 3.5 summarises the model results for all tablet types. An interesting observation is that in comparison to the other tablet types, the 5 mg tablets show a better result in the relative prediction error based on the RMSEV values (row1-column3) versus the RMSECV values. As the cross-validated values are based on the original calibration data, better results are typically expected, which is not the case for the 5 mg tablet type. The other types do follow this convention though, and show an increase between 0.9 and 1.1% relative prediction error going from RMSECV to RMSEV values. It is unclear where this unusual result comes from but will be discussed in the discussion section of this chapter.

It has to be noted that the paper by Dyrby et al. does not provide the relative prediction errors based on their test sets but rather their cross-validated models[91]. However, in all cases, the toolbox-based relative prediction errors (below 1.9% RPE for the 10-20 mg tablet types and 3.04% RPE for the 5mg tablet type) beat the reported results by Dyrby et al. stated to be between 2.7 and 3.7% where the latter is the 5 mg tablet type result[91].

Table 3.5: Summary of modelling results for 5 mg and the additional tablet types including relative prediction errors (RPE) for RMSEV and RMSECV values.

Tablet Type (in mg)	Model Conditions	RPE RMSEV (%)	RPE RMSECV (%)
5	Nr. of Latent Variables: 2 Baseline Correction: 2^{nd} order derivative with a 2^{nd} order polynomial Scatter Correction: SNV Smoothing: Savitzky-Golay 2^{nd} order polynomial	1.59	3.04
10	Nr. of Latent Variables: 2 Baseline Correction: 2^{nd} order derivative with a 2^{nd} order polynomial Scatter Correction: MSC Smoothing: Savitzky-Golay 2^{nd} order polynomial	2.81	1.88
15	Nr. of Latent Variables: 3 Baseline Correction: no baseline correction Scatter Correction: MSC Smoothing: Savitzky-Golay 2^{nd} order polynomial	2.63	1.79
20	Nr. of Latent Variables: 3 Baseline Correction: 1^{st} order derivative with a 2^{nd} order polynomial Scatter Correction: SNV Smoothing: Savitzky-Golay 2^{nd} order polynomial	2.85	1.73

3.5 Discussion and Future Work

3.5.1 Use Case 1

Disregarding the difficult experimental premise that the Case 1 data are based upon, this use case presented a good, low-complexity example to demonstrate some of the strengths and challenges of the presented toolbox: a two-solution-mixing system of IPA and water which results in a measurable change in IPA concentration and thus a single variable input. Sufficient calibration and validation data were prepared to build test models and try the various pre-processing methods built into the toolbox. Starting with outlier detection, the algorithm suggested a sample within the calibration set to be particularly different from the bulk (Figure 3.9). As was shown with sample 33 in this use case, outlier detection has presented itself as a relevant feature. With the toolbox, a faulty measurement which would have normally been easily overlooked within the set of 117 samples was identified. At the same time, it was essential to follow any outlier detection results with a thorough investigation of the physical / chemical causes, as use case 2 shows in more detail.

Moving to component selection (Figure 3.14), an initial assessment of the required number of latent variables based on the RMSECV versus the number of latent variables indicated that 2-3 latent variables were sufficient to capture most of the variance in the model. Viewing the RMSEV versus the number of latent variables, this assessment was not as easily visible, suggesting some models struggled with the validation samples. While a trend was visible, some models showed their weak fit to the test data thus being less favourable.

Viewing the interaction and main plots, signs of overfitting can be seen when observing the RMSEV values versus an increasing number of latent variables (Figure 3.15 row1-column5). As these plots are based on validation sample results, the models appear to struggle to describe samples at a higher number of latent variables which is represented in a rising RMSEV. This is caused by the models describing features present solely in the calibration samples, such as noise or other non-Y matrix related features, which cannot be found in the validation samples. This, again, suggests that no more than 2 or 3 latent variables should be used in this scenario. It appears there is no significant noise in the data as smoothing operations do not help to improve model performance and such, the more successful models do not employ it (Figure 3.15 Row3).

In the current version, all plots use the validation data results for plotting purposes when they are available. However, employing calibration data results could reveal further information about the raw data used and the issues present. For example, it might be easier to spot the required number of latent variables to incorporate baseline errors or whether smoothing and other correcting factors are needed. On the other hand, heavy overfitting of the models would not be as easy to spot. Currently, users have to run the process twice to get both sets of results, even though the data is available. This should be fixed in a future software update.

Overall, the toolbox-suggested model effectively halved the error compared to a basic mean-centring approach (from 3.0 RMSEV to 1.4 RMSEV). To achieve this, three latent variables and baseline correction using 1st derivative 2nd order polynomial in conjunction with multiple scatter correction were employed. On the premise of mixing two solvents and trying to predict their concentration, a less complex model could have been expected, requiring a lower number of latent variables. However, using a scanning mirror to scan a single line HSI-camera across a round capillary tube to capture the mixing of the solvents through the tube did likely complicate the description and modelling of the data. This setup introduced, among others, ray tracing problems, potentially rendering each non-horizontal scan inaccurate. A scanning motion of a mirror system requiring significant additional calibration to secure the motor would return to the same position at a known time as well as scanning through an entire horizontal slice, resulting in a reading that depends on the mixing status and position of a HSI pixel read-out across a slice of tube.

The experimental setup was not fit for the task of recording a timed snapshot of a mixing process across a capillary tube. It was, however, fit to report a concentration of a solvent after having modelled the experimental range of values possible. The latter scenario has been shown to work successfully. The toolbox provided a solution to the problem at hand, suggesting an effective pre-processing method that works even-though the original experiment might not have been ideal.

3.5.2 Use Case 2

Use Case 2 is based on an open data set made available by Dyrby et al. based on a chemometric paper published in 2002[91]. The authors assessed the content uniformity of pharmaceutical tablets through the use of NIR and Raman spectroscopy surface measurements[91]. The target tablets used for this study were supplied in four dosage strengths (5, 10, 15 as well as 20 mg tablet weight).

In comparison to Use Case 1, the available data and modelling requirements have more layers of complexity in this case, making it an excellent test scenario for the toolbox. Four different tablet sample sets, established on their API content, are independently modelled. Each tablet type data set has been built on three different batches where the individual batches demonstrate different tablet characteristics but remain the same with regards to their API content. While this makes for an interesting data set in itself, the results obtained from the toolbox analysis can furthermore be compared to the modelling efforts presented by Dyrby et al.[91]. However, as the authors did not state which exact samples were used for calibration and validation set building, the main comparison has been set against the performance of the paper's reported cross-validation model results. The 5 mg tablet type was chosen as the main comparison sample, with the other tablet types, 10-20 mg, being compared in less detail.

Beginning with the outlier detection toolbox output, Figure 3.19 showed a number of suspicious samples that warranted further investigation. While the suspect sample did show a lower overall intensity to similarly concentrated samples, it could not be proven that this sample was erroneous. As previously stated, this example showed how important it is to assess whether an outlier can or cannot be removed from the sample population pool. However, against the statements of Dyrby et al., during the outlier investigation, it did appear as if the differences in the batch products do have ramifications with regards to the respective sample spectra. Dyrby et al. state in their paper that batches did not show large differences in spectra due to the API content remaining the same. Nevertheless, viewing Figure 3.20, it appears as if samples from the same batch and their similar manufacturing characteristics might influence the spectra more than originally anticipated. Samples of the same batch have spectra with similar intensities banding them together. One reason for this could be that samples of the same batch have a similar baseline compared to samples of another batch however, such a depth of sample analysis is outside the scope of the toolbox. Nevertheless, this could lead to more inaccurate model results and could warrant further checking of the batch characteristics in relation to changes in the NIR spectra. Unfortunately, the open data set does not define which batch number in the data set corresponds to which batch used in the paper.

Moving on to the component selection graphs, Figure 3.22 shows a typical steep decrease in model error which can be observed across the first few latent variables. The majority of the selected models, however, show that after the 2$^{\text{nd}}$ or 3$^{\text{rd}}$ latent variable, increases in the predictive ability of the model become smaller with each increase in the number of latent variables suggesting two or three are the ideal number of variables. Similarly to Use Case 1, viewing the RMSEV component selection figure, depending on the model selected, it appears that the RMSEV increases between 2 to 4 latent variables, suggesting an overfitting of the model, again confirming 2-3 latent variables should be chosen. Likely following a similar logic, this is also the number of model variables chosen by Dyrby et al.[91].

Continuing with the interaction and main effect plots found in Figures 3.24 and 3.23, the different necessary pre-processing steps and their interaction were assessed. The results indicated that scatter correction, baseline correction using first derivatives and smoothing of the data significantly improved model results. The smoothing outputs from the interaction plots in particular showed that, without smoothing, noise in the data would be incorporated into the model leading to overfitting, which can be seen viewing row3-column5 in Figure 3.24. In the interaction plots in Figure 3.24 column 5 in general, the various pre-processing methods all show a trend of an increase in RMSEV past the 2$^{\text{nd}}$ or 3$^{\text{rd}}$ latent variable. This again confirms that this is the ideal model complexity range as overfitting takes place past that point. In comparison with the results presented by Dyrby et al., similar pre-processing methods have been chosen by the toolbox, including first derivatives and scatter correction. However as Dyrby et al. only used a single pre-processing method and step per model, the model results are not entirely comparable.

A peculiar result was found in the most successful 5 mg model regarding the test sample set, where significantly better model outputs than the cross-validated sample set were demonstrated (0.089 RMSEV versus 0.17 RMSECV). Models for the other tablet types do not follow this trend and show better results for the RMSECV tested models. There are a number of possible reasons for such an unusual behaviour: the samples randomly selected for this procedure may, for example, be particularly similar to the calibration set and thus falsely promise better results than actually achieved in an independent test. No other model in the test responded as well to the test samples making this a suspiciously good result that would require further testing. On the other hand, the cross validation for the 5 mg tablets had 70 instead of 80 samples to work with which could have negatively impacted the RMSECV results in general. Given the outstanding model performance, the former is a more likely reason.

In addition to the local models built and presented, Dyrby et al. also tested a global model including all tablet types and variables. Given the suspected differences in batches and the differences in tablet types in general, the reported performance of the global model reaching a best of 4.1% relative prediction error is impressive. However, additional pre-processing could have improved this further, particularly if scaling methods would have been employed. While scaling did not help in the local models in the tests performed in this chapter, the potential differences between the batches could have normalised thus aiding a global model leading to overall better results.

In conclusion, while all model results from the pre-processing with this toolbox were leading in RMSECV values in comparison to the presented values in the paper, it has to be noted that too much pre-processing is not always welcome and can make future predictions less robust[62]. This is highly dependent on the calibration data having covered enough of the potential experimental design space, the differing sample characteristics and variables that future data sets could show. Finally, it is interesting to note that the validation test set performance from the toolbox achieved similar results to the values achieved by Dyrby et al. using their RMSECV samples sets. Such an observation suggests that the additional pre-processing steps indeed improve the model and are thus useful in this shown scenario.

3.5.3 Toolbox Discussion Points

This toolbox started off as an add-on to the licensed PLS-toolbox software for MATLAB but was reprogrammed to run as a separate non-licensed software bound to MATLAB. The code is provided in the appendix and will be further tested, validated and developed on a continued basis at the CMAC Hub. After additional third-party validation and testing (i.e. CMAC), the toolbox will be openly published via repositories, such as GitHub. While the code is being published as part of this PhD, it is believed that additional validation and incorporation into a larger framework will greatly expand its reach.

The advantages of the toolbox in comparison to the previously discussed and published options are accessibility (though currently limited to CMAC) as well as ease of use. Many researchers and experimentalists shy away from large code with command line input requirements whereas graphical user interfaces are more familiar and easier to commandeer. With the current push to digitalisation and the creation of digital carbon copies of laboratories in the form of digital twins, more process models, standardisation and the robustness of models with well-reasoned pre-processing methods will be useful[92,93]. Data that is treated and saved in a standardised way, likely in a centralised location, available to other researchers, is the first step to not only make it findable and accessible but also interoperable and, with the addition of user rights and further added metadata, reusable. This toolbox will hopefully aid in speeding-up the process of pre-treating data while also being helpful in approaching unknown data sets and identifying underlying issues, even with less experienced users.

The potential for upgradeability is high as the design-of-experiment matrix simply needs to be expanded with additional methods to be applied to the data. However, validation and future proofing of the toolbox are required before this can properly occur as it is currently bound to MATLAB's versioning issues thus potentially leading to bugs and inaccessibility. The toolbox is currently built on an early build of the MATLAB internal App-building platform which has been updated in the recent years. Newer releases of this platform required changes to the original code that manifest themselves in incompatibilities which in turn can be addressed with a few bug fixes and feature proofing prior to a larger release.

3.5.4 Comparison With Other Toolboxes

This toolbox was heavily inspired by Walmsley and Flaten's original paper from 2003 where they presented the use of 'design of experiments to select optimum calibration model parameters'[65]. The authors advocated that using model influencing variables, such as pre-processing methods, as factors in an experimental design, interactions and major effects of those variables could be easily determined. Although other model influencing aspects are mentioned, such as variable sub-selection, more in-detail optimisations of the calibration set and outlier detection, the algorithm presented in their paper is, similarly to the toolbox presented in this chapter, only concerned with testing various pre-processing methods and the model performance improvements after their application[65]. The DoE variables tested against were: type of regression model, where either PLS or principle component regression were used; scaling of the data, where no mean-centring versus mean-centring were used; as well as derivatives, where first and second derivative were employed.

In addition, orthogonal signal correction (OSC) and Box-Cox transformations were also available in Walmsley and Flaten's tool. OSC is a PLS-related pre-processing method that tries to remove variation in the X-data that is unrelated to the Y-data[94], such as baseline or scattering errors, while claiming to be removing a minimum amount of information in the spectra pertaining to the Y-data[94]. Svensson et al. argued that the application of OSC in their analyses did not provide a notable improvement in the PLS calibration models generated but rather potentially aided interpretation and understanding during the data analysis steps[95]. Box-Cox transformations, on the other hand, are used to transform available data that might not show a normal distribution into normality. Calibration data sets that do not follow a normal distribution and are used for regression modelling purposes can potentially lead to bias in the model due to some over- or underrepresented samples having more weight. OSC as well as Box-Cox transformations are not applied in the toolbox presented throughout this chapter as they either require more specialist intervention or are not typically used in the spectral applications at CMAC. Instead, scatter correction methods, separate baseline and smoothing corrections and additional scaling methods have been added, which are more commonly used but were not included in Walmsley and Flaten's approach.

Regarding the graphical output of the tools, the interaction and main effects plot outputs have been based on the original paper by Walmsley and Flaten as they convey a great depth of information and can be used to inform users about their data[65]. This original paper formed the basis of the toolbox described in this chapter which can be seen as an expanded and easier-to-use version which mostly focuses on accessibility and usability to non-experts. No exact time requirements per sample set are given by Walmsley and Flaten, however, it is hinted that given a powerful enough CPU (in 2003), analysis times should not be excessive and in the minutes rather than dozens-of-minutes range. Comparing to the toolbox presented in this chapter, on all tested examples, typical calculation times typically took less than 1-3 minutes. To calculate these results a modern multi-core processor was used. It is likely that, were both tools compared side by side, similar results would be achieved today, given the advances in processing power over the last 17 years.

Gerretzen et al. also presented the use of DoE for data pre-processing in 2015[63]. Again, DoE was used to generate a matrix of various potential pre-processing method combinations to be tested. However, Gerretzen et al. then applied a check whether certain pre-processing steps are found to be "useful", i.e. improve the model outcomes[63]. If found to be successful, a method is then optimised further in individual steps while unsuccessful methods are deemed irrelevant and not pursued. To support this interpretation, significance and effectiveness factors were introduced that calculated how well the various pre-processing steps performed against the model data. Similarly to the interaction and main effect plots presented in this chapter and also shown in Walmsley and Flaten's work[65], these factors were also used to calculate interactions between the various pre-processing methods, informing the users on which combinations of methods showed synergies and which ones had detrimental effects.

The authors furthermore expanded on their testing method by introducing pre-processing stages, where baseline correction was followed by scatter correction, then noise removal and finally scaling[63]. This stage-wise approach was taken as the basis for the toolbox described in this chapter, where similarly baseline correction was followed by scatter correction, smoothing and then scaling.

The available pre-processing steps are expansive with many additional methods per stage that are not currently available in the toolbox presented here, examples of which include detrending for baseline correction, additional scatter correction methods such as robust normal variate transformation similar to SNV, and additional scaling options such as Poisson scaling. A factor which was not applied in the testing of pre-processing methods was the number of latent variables which instead was assessed automatically. In a simplified manner, this was done by cross-validating the models and at each increasing number of LVs statistically checking whether the RMSECV had notably improved. This larger, complicated, and more in-depth approach, with an increased number of methods and testing however came at a price. The authors report that typical processing times of data sets are between 15-30 minutes depending on the data sets at hand. However, it is argued that the "smart" approach of pre-checking the viability of pre-processing methods significantly reduced analysis times in comparison to using all of their available methods in a brute force approach, which would have required approximately one day per analysis instead. Either option is significantly more time-consuming than the toolbox presented in this chapter. In addition to applying DoE methodology to understand and inform on the choices of pre-processing techniques to treat spectral samples prior to modelling, both papers have another major commonality. While their specific approaches might differ, the authors all make a case for a more systematic, easier-to-use and more informative way to pre-processing samples that offers optimised strategies while aiding scientists in their analytical investigations.

This toolbox takes the approach presented by Walmsley and Flaten and adds additional ease-of-use features such as a graphical user interface, outlier detection and a small selection of typically used pre-processing methods applied in the CMAC laboratories. This is combined with some similar result figures inspired by Gerretzen et al., building a toolbox that aids and informs non-specialist and experienced users alike in the pre-processing decision making prior to building larger models. Depending on the sample sizes and whether cross-validation is activated or not, the model building process can take less than a minute to a few minutes when run on a modern multi-core CPU. In today's lab and work environment, extreme processing times of more than 10 minutes are not favoured due to the fast pace and time requirements of researchers. However, as previously discussed, this toolbox, in its current state has been built primarily to aid the pre-processing method selection process. While the data output is standardised and basic models are built, better and more complex model building tools are available and will likely be needed to build, maintain, and improve modelling efforts.

3.6 Conclusion and Future Applications

This chapter demonstrates a quick, automated and robust pre-processing method selection and benchmark tool for NIR spectral data. Two case studies have been presented in which optimised pre-processing methods have been identified to treat samples. Basic preliminary PLSR models to test these pre-processing methods showed successful prediction results and revealed the pre-processing improvements necessary to yield optimal modelling outcomes based on the raw data provided. Use Case 1 demonstrated the use of the toolbox for pre-processing method selection for an experimentally challenging two-solution-mixing system of IPA and water. Using the basic modelling test environment provided in the software to verify the output, the best pre-processing method selected resulted in a 53% error decrease versus using a non-pre-processed data set. Use Case 2 depicted the application of the toolbox to assess the correct pre-processing methods required to model content uniformity of differing batches of pharmaceutical tablets. In this case, the toolbox was used to compare against published results by Dyrby et al.[91] in which researchers used standard pre-processing methods to treat the used sample data prior to their modelling attempts. The toolbox-suggested pre-processing methods and comparative models indicated approximately 30% lower relative predictive errors in comparison to the results published by Dyrby et al.[91], demonstrating the usefulness of the tool. While simple model building has been provided for quick assessment of the success of the chosen pre-processing methods via the use of PLS1 regression, it is recommended that the toolbox be used hand-in-hand with more specialist modelling tools for advanced modelling purposes.

Compared to similar tools, such as those published by Flaten and Walmsley or Gerretzen et al.[63,65], the toolbox presented here is a hybrid solution with easier access, quicker turn-around times, convenience functions and future upgradeability in mind. This tool has been designed for the typical data processing tasks performed at CMAC and could help future analyses with demonstrably robust pre-processing method development. No previous knowledge about spectral pre-processing methods is necessary but, if engaged, the tools provided will greatly improve understanding of the data at hand, aiding further analyses. This is achieved by offering graphical outputs that show the effects of the various pre-processing methods indicating existing problems in the data. The outputs of the toolbox are kept accessible and simple with all raw and processed data being saved for future down-stream data processing. The currently available graphical user interface and thus easy accessibility to the software make it an ideal candidate for future web-based toolbox implementation for wider and easier access.

As model building is becoming more and more popular, the toolbox could become part of a larger processing pipeline, where data could be pointed towards an automated pipeline, cleaned, pre-processed and readily modelled using various modelling strategies, from typical regression models such as PLSR, to more sophisticated machine learning models such as random forest regression. Implementing the toolbox or a similar design into a standard IR and NIR processing workflow with aggregated previous pre-processing data from similar experiments would be a great start to building a house library of standardised and easily comparable spectra. A pre-processing method library of typical methods used for a known set of experiments could help scientists to save time in future analyses identifying what kind of characteristic processing steps are required for certain types of analyses. This could be achieved by building machine learning models which suggest such pre-processing methods based on historical data that could be built on the back of standardised data outputs from this toolbox.

With standardised pre-processed data, a standardised spectral library for all spectra produced at CMAC could follow. This could help researchers to understand their pre-processing requirements earlier, identify their spectra faster and gain deeper understanding about their chemical process. While these are ambitious and long-term future research proposals, smarter working, the interconnectedness of data, and the move towards industry 4.0 are being discussed in every large industry.

Chapter 4

Investigating Paracetamol Single Crystal Impurities

4.1 Introduction

Although the pharmaceutical industry has been exploring many forms of drug delivery systems to administer medication, oral dosage forms, such as tablets and capsules are still the preferred administration route for most patients[96]. Succeeding in the effective delivery of the drug is a difficult endeavour; the active ingredient must pass through the inhospitable conditions of the gastrointestinal tract unaffected before delivering the drug load to the appropriate area of the body.

Tablets and other oral dosage forms, such as capsules, typically contain a mix of different compounds that are designed to help in the manufacturing, administration and delivery of the drug. There are many additives that help enhance drug stability, assist in the production of the medicine as well as help to get the drug to the correct place within the body. This can include binders that help keep different materials together, lubricants that aid tabletting, disintegrants that encourage the break-up of the oral dosage form within the body as well as coatings for protection. The most important constituent of a drug, however, is the active pharmaceutical ingredient (API), which is the bioactive compound within a medicine that causes the desired effect to the patient.

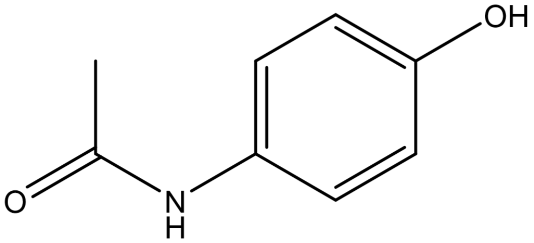


Figure 4.1: Molecular structure of paracetamol

Most APIs are produced by means of chemical synthesis. These processes often require numerous steps involving a large variety of reagents which should, ideally, be removed entirely from the final product through crystallisation and filtration processes. Components include solvents and starting materials to intermediates, catalysts and by-products[97]. Elimination of these impurities can be very difficult, particularly if the solubilities of these compounds are similar to the final product. Any residual components that cannot be removed from the final product are called impurities. Some impurities, such as degradation products or by-products can be structurally related to the API and have the ability to influence the behaviour of the final drug product, such as crystallisation characteristics and solubility.

Acetaminophen, commonly known as paracetamol, is a popular painkiller with analgesic and antipyretic effects[98]. The structure of paracetamol consists of a hydroxyl- and a methylamine-group attached to a benzene ring in para position (as shown in Figure 4.1). Paracetamol is a simple and low-cost drug with few production steps required, making it a common target for pharmaceutical manufacturing studies. Some important aspects of these studies include the identification and quantification of impurities as well as the impact that they may have on the process of API production[99].

Various studies into paracetamol production are available that assess the impact of structurally related impurities on nucleation, crystal growth, surface, texture and morphology[99–105]. As Ottoboni et al. (2018) summarised, there are three principal ways in which structurally-related additives can impact the nucleation and growth of crystals[99,100,106]:

1. Inhibition of the nucleation process via disruption of the emerging nuclei; as shown for metacetamol by Hendriksen et al.[100,106], where nucleation is the process of self-assembling ions or atoms to become a crystalline solid. Typically, initial nucleation sites become the location of further deposition of particles and crystal growth.

2. Incorporation of the impurity into the crystal lattice without further deterring additional API molecules from attaching to the growing crystal; Hendriksen et al.[100,106] investigated the effect of metacetamol as paracetamol docking agent.

3. Morphological changes induced through the disruption and blocking of adsorption of solute molecules such as described for p-acetoxyacetanilide inhibiting crystal growth by blocking certain crystal facets[99,100,106].

Many of these studies also examine the impact of such additives on crystal surface texture and morphology in the form of surface roughening, addition of new crystal faces to the crystal form and changes to general surface character of crystals[99–105].

Instruments and techniques used for the analysis of these features include; optical microscopy (OM) and scanning electron microscopy (SEM) for crystal morphology and surface texture studies, energy-dispersive X-ray spectroscopy (EDX) and X-ray photoelectron spectroscopy for surface elemental composition analysis[106] and atomic force microscopy (AFM) for surface roughness studies[106]. Through applying these and other techniques, a number of interesting findings have been reported on the impact of impurity incorporation on paracetamol crystals. Examples include studies on the presence of structurally-related impurities affecting morphological and chemical properties of paracetamol crystal faces[102]. The incorporation of p-acetoxyacetanilide causing changes to shape and defects of paracetamol was described by Chow et al. (1985) and Prasad et al. (2001)[103,104], while the integration of metacetamol inhibiting the growth of certain crystal faces resulting in a different crystal shape was reported by Saleemi et al. (2013)[105,106].

One of the challenges encountered during these previous studies was “extracting information about both the chemical and topographic character of individual crystal faces using only one technique"[105,106]. This chapter details the investigations carried out using Raman microscopy mapping and Time-of-Flight Secondary Ion Mass Spectroscopy (ToF-SIMS), for the simultaneous examination of chemical and topographic information in the form of impurity localisation and distribution on crystal surfaces, as well as surface modifications caused by impurity incorporation.

Compared to other currently more prominent fields of application, such as mass spectral imaging of biological materials[107], ToF-SIMS has not been utilised as much in published pharmaceutical material characterisation[13,21,108], even though it is very applicable[109,110]. Some of such examples include studies by Mahoney and Fahey (2008) who used 3D imaging to analyse the drug distribution on drug-eluting stents, showing that increased drug loadings lead to a heterogeneous distribution of the drug substance in the stent[111]; Rafati et al. (2012) who investigated the use of partially porous protein-loaded microspheres as controlled drug delivery biomaterials and used ToF-SIMS to define the 3D distribution of protein at the surface as well as within the porous microspheres[112]; and Chan et al. (2012) who reported a novel inhalable anti-tuberculosis drug formulation utilised ToF-SIMS measurements to characterise the surface composition of spray-dried powder particles. In the latter case, the measurements confirmed that when a hydrophilic solvent is employed during the spray-drying manufacturing process, hydrophobic and heavier molecules would form the outer layer of the final dried particle product. The desired distribution was mostly substantiated, though unexpected surface coverage inconsistencies were revealed through ToF-SIMS imaging likely created during the manufacturing process[113].

In a review paper discussing the state of ToF-SIMS in the pharmaceutical industry from 2011, Barnes et al. called the current state of usage of the technique "in its infancy" but predicted that the three-dimensional mapping of drug distributions and quality control in dosage forms would likely see increased application in the future given improvements of the measurement technique[108].

So far, a simultaneous exploration of chemical and surface character of individual pharmacologically relevant crystal faces using ToF-SIMS has not been published. A similar study analysing the "face specific surface properties of pharmaceutical crystals", though, was performed by Muster and Prestidge in 2002[114]. The authors applied the technique to characterise the molecular arrangement of pharmaceutical crystals. The study confirmed that the fragmentation patterns of crystals analysed were strongly crystal face dependent resulting in face-specific spectra even when analysing the same compound. This was previously only inferred from molecular models[114]. Together with atomic force microscopy and contact angle measurements, the authors were thus able to determine the face-specific structure, chemistry and wettability of their crystals.

In this chapter, ToF-SIMS has been used to further explore pharmaceutical crystal characterisation and to assess the usefulness of the technique with regards to the concurrent analyses of impurity-induced surface modifications as well as surface and bulk impurity distribution in and on crystals.

In order to gauge the sensitivity and effectiveness of the technique, three different crystallisation methods were investigated to determine the effects of an impurity on the molecular distribution of crystal surfaces; cooling crystallisation, solvent drop and epitaxial growth. The study further assesses the viability of ToF-SIMS to study the chemical as well as structural character of crystal surfaces using a single technique.

An initial impurity screen of ten paracetamol-related impurities revealed that only two of these, specifically 4-nitrophenol and 4-acetamidobenzoic acid, were suitable candidates for this study due to their unique fragment ions which did not overlap with any paracetamol fragment ions. 4-nitrophenol was chosen for further study due to the better availability of its crystallisation products.

Work has been performed to evaluate different kinds of impurity loading scenarios and how well the impurity could be detected directly on crystal surfaces using the ToF-SIMS instrument. These scenarios include directly dropping and recrystallising the impurity onto the surface of a pure paracetamol crystal, epitaxial deposition of an impurity on-top of a pure paracetamol crystal where the impurity is grown on top of the crystal surface, as well as crystallising an impure paracetamol crystal from a solution of paracetamol with 4%mol impurity.

Publication

Parts of the work presented in this chapter resulted in the following publication:

Impact of Paracetamol Impurities on Face Properties: Investigating the Surface of Single Crystals Using ToF-SIMS; Sara Ottoboni, Michael Chrubasik, Layla Mir Bruce, Thai Thu Hien Nguyen, Murray Robertson, Blair Johnston, Iain D. H. Oswald, Alastair Florence, and Chris Price; Crystal Growth & Design 2018 18 (5), 2750-2758; DOI: 10.1021/acs.cgd.7b01411

4.1.1 Aims

The aims of this study were to:

i) Assess whether ToF-SIMS could be a useful technique to detect and image the presence of paracetamol-related impurities at low concentrations in crystals formed via cooling crystallisation, and

ii) Use the technique to compare the crystallisation product results to other impurity loading scenarios.

Within a larger context, this work was part of a collaboration with Dr Sara Ottoboni in which the use of ToF-SIMS was compared to more classical analysis techniques typically employed for the investigation of crystal surface textures and structures as well as chemical characterisation, such as optical and scanning electron microscopy, Raman imaging and X-ray diffraction (XRD).

4.2 Methods

4.2.1 Sample Preparation

All crystals were prepared by Professor Chris Price's research group based in the Chemical and Process Engineering Department at the University of Strathclyde. Crystallisation methods used in this chapter include drop-deposition of 4-nitrophenol onto a paracetamol crystal, 4-nitrophenol epitaxial growth on a paracetamol crystal, as well as impure crystals with 4%mol 4-nitrophenol and 4-acetamido benzoic acid. The cooling crystallisations of pure paracetamol and pure 4-nitrophenol were also carried out for comparison. Measurements from analyses other than ToF-SIMS have been collated and provided by Dr Sara Ottoboni. These include XRD, Raman as well as optical and scanning electron microscopy measurements and images.

Materials

Materials purchased from Sigma Aldrich included:

- Paracetamol (4-actamidophenol, Bioxtra, ≥99%),
- 4-nitrophenol (≥ 99%),
- Methyl-4-hydroxybenzoate (97%),
- 4-acetamidobenzoic acid (≥98%),
- 4'chloroacetanilide (97%),
- Acetanilide (99%),
- 4-hydroxy acetophenone (99%),
- Orthocetamol (97%),
- 4-aminophenol(98%),
- Metacetamol (≥99%),
- Absolute ethanol (GC grade ≥99.8%) and
- n-hexane (≥95%).

Acetaminophen acetate (99%) was supplied by Tokyo Chemical Industries.

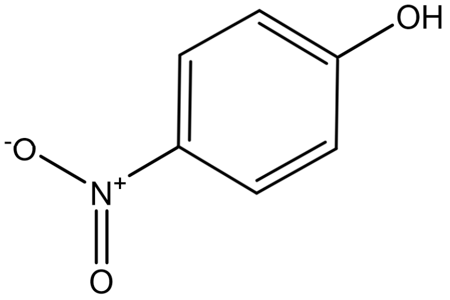


Figure 4.2: Molecular structure of 4-nitrophenol.

4.2.2 Impurity of Choice

4-nitrophenol is a reagent in paracetamol synthesis and structurally similar to paracetamol. The difference to 4-acetaminophen is the replacement of the methylamine group with a nitro group as can be seen in Figure 4.2.

Cooling Crystallisations

Pure paracetamol crystals (PP) were created by cooling a supersaturated solution of pure paracetamol (0.88 g paracetamol in 5 g absolute ethanol (6.3 mL)) from 50°C to 5°C and maintaining the solution at a steady 5°C until a single crystal appears. These pure paracetamol crystals were subsequently used for the drop-deposited and epitaxial impurity analyses[115]. The pure 4-nitrophenol crystal (PN) was grown using the same crystallisation procedure using a supersaturated solution of 4-nitrophenol.

An impure crystal with 4% mol 4-nitrophenol was obtained (P4%N) using a similar method, by cooling a supersaturated solution of paracetamol (0.88 g paracetamol with 4% mol 4-nitrophenol in 5 g absolute ethanol (6.3 mL)) from 50°C to 5°C and maintaining the solution at a steady 5°C until a single crystal appears. During a later phase of the experimental procedures, 4% mol 4-acetamidobenzoic acid paracetamol crystals (P4%A) were produced also using the same crystallisation method as the P4%N crystal.

Crystal with Drop-Deposited 4-Nitrophenol (PDN)

A saturated solution of 4-nitrophenol in ethanol (1500 mg/g ethanol) was prepared and deposited via syringe onto the main face of a pure paracetamol crystal. The ethanol was allowed to evaporate at room temperature leaving a layer of 4-nitrophenol crystals on top of the paracetamol crystal surface[115].

Crystal with 4-Nitrophenol Epitaxy (PEN)

4-nitrophenol was dissolved in hexane at 50°C and a pure paracetamol crystal immersed into it. The solution was then cooled to 5°C resulting in the formation of 4-nitrophenol crystals in the solution and epitaxially grown 4-nitrophenol crystals on the surface of the pure paracetamol crystal[115].

4.2.3 Microscopy

Three techniques were applied to image and assess the surface features of the crystals. Stereomicroscopy images were taken using a BMDZ zoom microscope (Brunel Microscopes Ltd, Chippenham, UK). Differential interference contrast (DIC) optical microscopy images were taken at a 10x magnification factor using a Leica DM6000M microscope (Leica Microsystems, UK), while scanning electron microscopy images were taken using a Hitachi TM-1000 (Hitachi High-Tech, UK). The SEM measurement conditions were as follows: back scattered detector, accelerating voltage 15000 V, magnification 200x, working distance 6700 µm, emission current 57.4 mA, scan speed slow, vacuum conditions 15.0 kV, WD 6.7 mm[106]. All microscopy images were provided by Ottoboni and colleagues.

4.2.4 Further Analysis

In the context of the wider study, additional measurements and analyses were performed by Ottoboni et al. (2018) using atomic force microscopy (AFM), Raman microscopy, high-performance liquid chromatography (HPLC) and X-ray diffraction. As these results are not within the scope of this project, the measurement and instrument parameters have not been discussed in detail.

4.2.5 ToF-SIMS Analysis Method

The sample mounting, measurements and analysis of all ToF-SIMS data was carried out by the author and discussed with Dr Sara Ottoboni in the context of the wider study.

Sample Mounting

As emphasised by the limited number of published articles, ToF-SIMS does not appear to be a common analysis technique for crystal analysis and as such, to the author's knowledge, no available papers are published that describe the challenging sampling method for large-scale crystals.

API crystal products used in pharmaceutical manufacturing are typically small (sub-100 µm), as further processing steps in the manufacturing procedure do not require a larger size. To facilitate measurements and provide a larger area for analysis, the crystals for this study were grown over a period of several days in order to produce larger crystal products. Asymmetrical and/or difficult crystal shapes as well as surface roughness provided some complications during surface measurements. Multiple crystal faces needed to be analysed depending on the information required and insertion into the instrument required precise adjustments to the stage and accurate sample placement. The instrument offers two possible methods for sample mounting, a top-mounted and a back-mounted sample holder, both of which were assessed. The following section describes the mounting procedure utilised for preparing the crystals in this study.

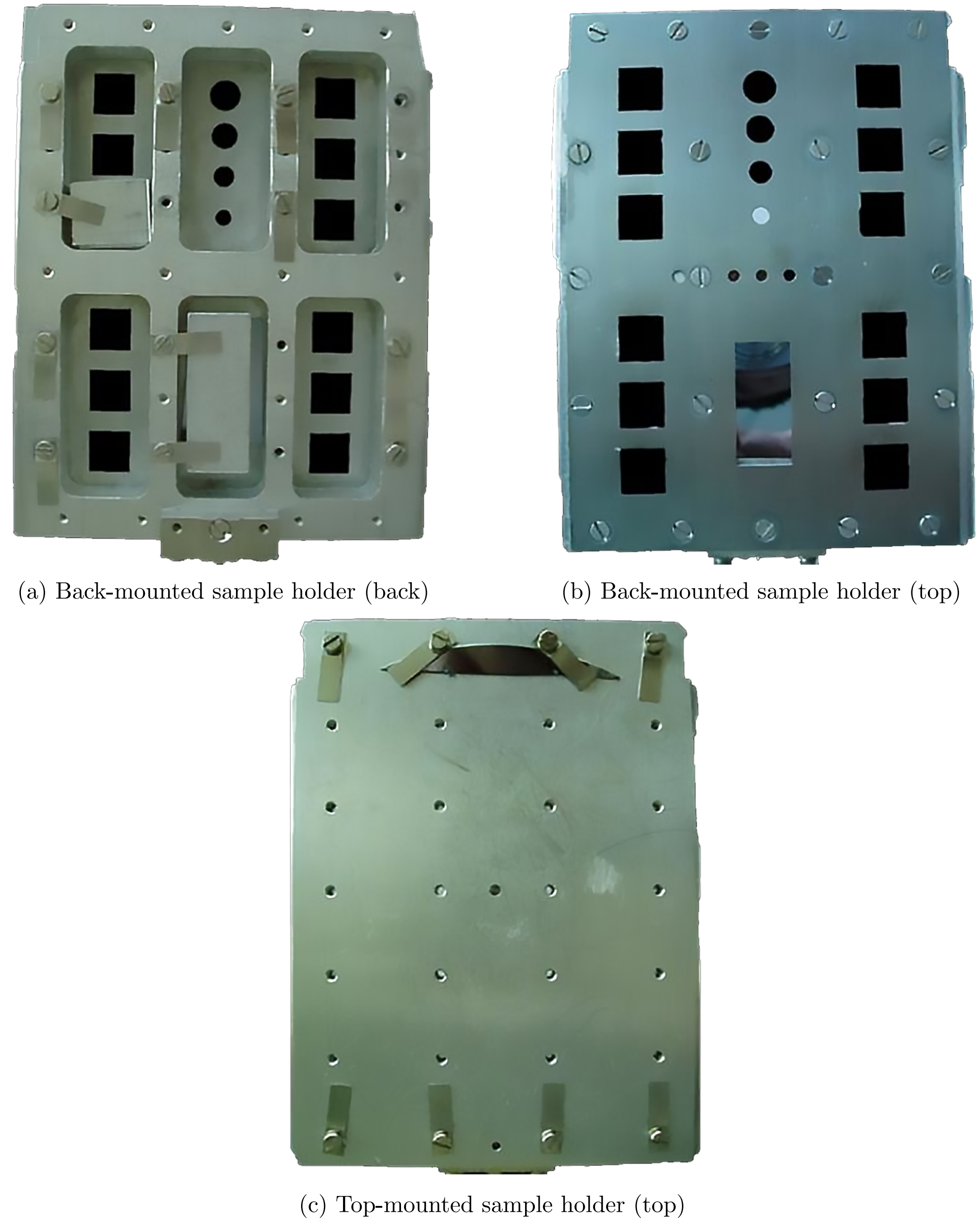
Top-Mounted versus Back-Mounted Sample Holder The top-mounted sample holder (Figure 4.3c) is a flat aluminium stage that allows for the mounting and securing of samples via screws and bendable metal plates. Mounting a sample on the top-mounted sample holder allows for maximum flexibility in sample size, orientation and movement but also increases the risk of potential damage to the instrument due to the manual handling requirements of the sample stage. The crystals were large, some approximately 10 mm tall and/or wide. Adjustment of the crystals on the top-mounted stage could have proven difficult, particularly with respect to the evenness or angle of the analysis surface and the automation of measurements.

The back-mounted sample holder (Figure 4.3b) has a number of sample cut-outs of differing size with height and width restrictions. The samples are held in place with an aluminium spacer and secured using metal clips and screws. Where possible, this allows samples to be pressed evenly against the top of the sample holder to create a flat surface. Samples are mounted below the surface of the sample holder reducing the risk of instrument damage due to sample protrusions. This in turn allows for automated sample holder movement thus significantly reducing the time for manual stage adjustments. However, this sample holder is designed for small, flat samples, and proves difficult for mounting larger, uneven samples such as crystals.

Both sample holders are suboptimal for the work required and the topographically challenging and uneven crystals. To circumvent the issues of manual stage adjustments and mitigate the risks of instrument damage, a method to mount the crystals into the back-mounting sample holder was developed.

As depicted in Figure 4.4, a single crystal (shown in yellow) was mounted on a small 1x1x0.1 cm^3 aluminium spacer (grey) and secured using double-sided sticky tape (shown in orange). If required, the height could be raised using more sticky tape to achieve an even analysis sample surface. The sample was then attached to additional aluminium spacers to build up the sides to the same level as the crystal in order to secure the sample into the back-mounted sample holder without the crystal protruding above the surface of the holder.

Using this method, a relatively even analysis surface was achieved, crystals could be mounted and later moved to other instruments for further analysis or be reintroduced for re-analysis without disturbing the sample. In addition, the procedure allowed for automation of the sample stage with the sample holder inside the instrument for easier calibration and movement around the sample. This method of mounting crystal samples is preferential due to the ease of use, automated sample movements and rapid analysis times possible, in addition to the reduced risk of damage to the instrument that would



(a) Back-mounted sample holder (back)

(b) Back-mounted sample holder (top)

(c) Top-mounted sample holder (top)

Figure 4.3: Back- and top-mounted sample holders for sample introduction into the instrument. Aluminium spacers, as seen in the images, are used to mount and keep samples in place. Image source (G. Trindade[28]).

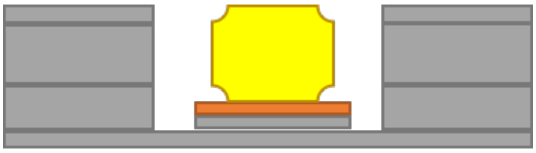


Figure 4.4: Mounting scheme for crystals prior to their introduction onto the back-mounted sample holder. The grey areas represent aluminium spacers, the orange area represents double-sided sticky tape while the yellow object represents a single crystal.

be more likely when using the top-mounted sample holder. Once mounted using the described method, the samples were introduced into the instrument for analysis.

Settings

Using the IONTOF TOF.SIMS 5, all samples were analysed in both positive and negative polarity. Apart from the initial screening of samples using bunched mode, all further measurements were performed using the burst alignment mode (delayed extraction) to compensate for the surface roughness. Negative polarity was chosen as the preferred analysis setting due to a significantly higher intensity of the 4-nitrophenol molecular ion fragment. Surface imaging was performed using the $\mathrm{Bi_3}^{2+}$ cluster ion using settings suitable for a high lateral resolution: burst alignment mode, 30 keV base setting with a 100 ns pulse width and a 0.05 nA beam current.

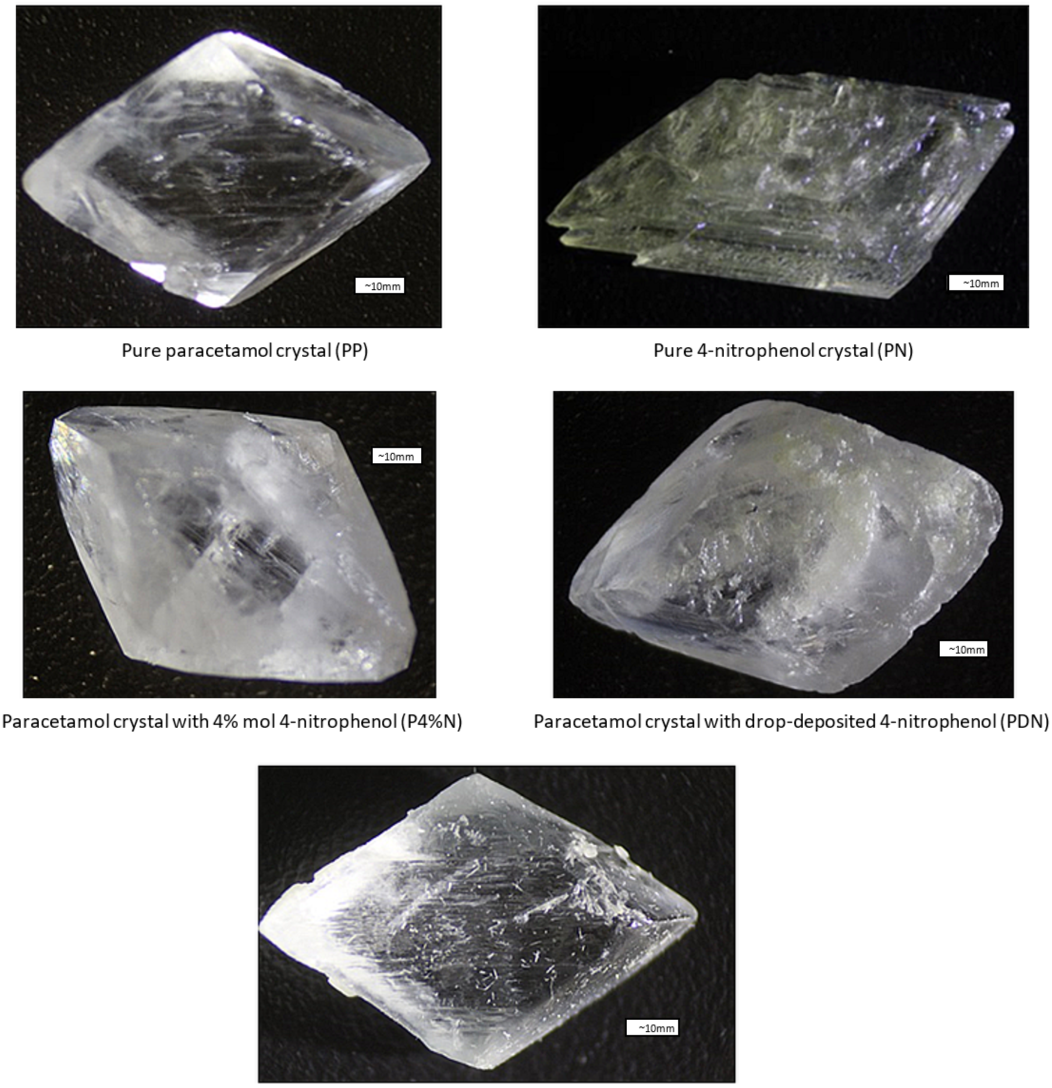
The electron flood gun was used to adjust for charging effects. The area of interest for the analysis was set to 100 µm^2. PP and PDN crystals were recorded at a raster size of 256 px x 256 px while the P4%N crystals were recorded at a 1024 px x 1024 px raster size and the resulting images were then pixel-binned to 256 px x 256 px to improve the intensity per pixel and the comparability with the other crystal sets. The total dose density threshold was set to 4e+11 ions/cm^2 for the PP and PN crystals and 9e+11 ions/cm^2 for the P4%N crystals, both of which are within the static limit of analysis[116]. Bulk studies were performed utilising spectrometry mode using 30 keV $\mathrm{Bi_3}^+$ primary ions for analysis and a 10 keV $\mathrm{Ar_{1500}}^+$ cluster beam for sputtering the surface. The analysis area was set to 150 µm x 150 µm while the sputter area was set to 450 µm x 450 µm. The current for these analyses was expected to be around 0.5 nA, however the value hasn't been recorded in the datafile.

4.3 Results

4.3.1 Optical Characterisation Methods and Comparison

Pure Crystals of Paracetamol (PP) and 4-Nitrophenol (PN)

The crystals produced were compared optically to identify different surface characteristics. Viewing the stereomicroscopy images in Figure 4.5, clear structural differences between the pure paracetamol and pure 4-nitrophenol crystals can be noted. While paracetamol displays larger step-like areas with apparent edges, the 4-nitrophenol crystal has more curved and smaller area steps altogether. These observations are additionally confirmed when viewing the optical microscopy as well as SEM images of the PP and PN crystals (Figures 4.6a and 4.6b, as well as 4.7a and 4.7b, respectively)[106]. Viewing the stereomicroscopy image of the P4%N crystal, as shown in Figure 4.5), it is not directly apparent that the addition of 4-nitrophenol has affected the crystal shape. Single Crystal X-Ray Diffraction (SC-XRD) was used by Ottoboni et al. to confirm the growth of an additional crystal face[106]. Further changes were observed through a rougher surface and an increase in the number of steps, as well as additional defects that can be seen in detail by viewing the OM and SEM images in Figures 4.8a and 4.8b. These findings are consistent with observations by Prasad and Thompson[102,104,106].



Pure paracetamol crystal (PP)

Pure 4-nitrophenol crystal (PN)

Paracetamol crystal with 4% mol 4-nitrophenol (P4%N)

Paracetamol crystal with drop-deposited 4-nitrophenol (PDN)

Paracetamol crystal with 4-nitrophenol epitaxy (PEN)

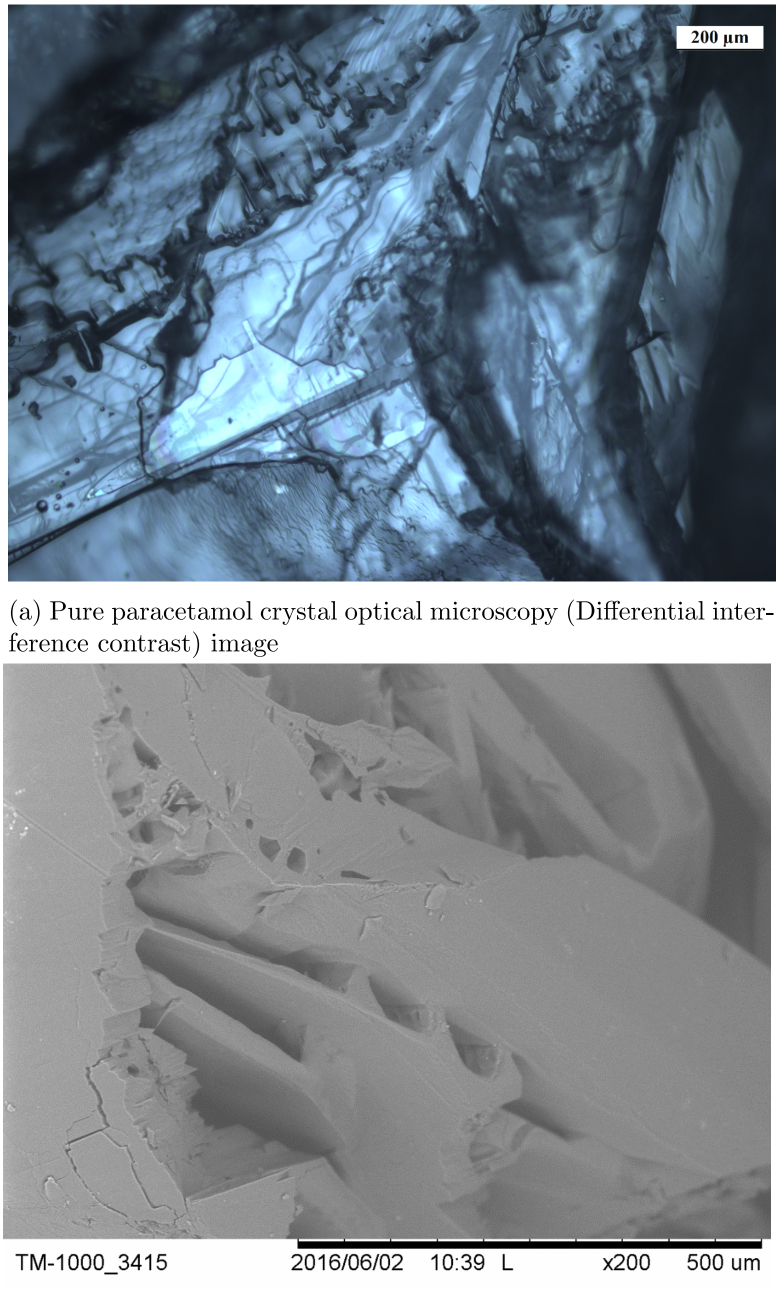
Figure 4.5: Stereomicroscopy images of the paracetamol crystals produced for this study with an included bar contained in each image depicting approximately 10 mm for scale.

PDN Crystal

As shown in the stereomicroscopy image in Figure 4.5, a boundary area on the top surface of the paracetamol crystal can be noted where the 4-nitrophenol solution was dropped onto the crystal. This area appears to be rougher, featuring surface defects not present on the rest of the crystal surface. Viewing the OM and SEM images, this roughness can be explained to be 4-nitrophenol that has crystallised on the paracetamol crystal surface after evaporation. The OM image (Figure 4.9a) prominently features the impurity drop boundary (dark versus light area) with needle-like crystals growing within, while the SEM image shows how crystals towards the centre appear to grow more chaotically, pointing towards all directions, with the outer perimeter crystals growing away from the drop source, mostly pointing outwards, as can be seen around the interfaces between the darker and brighter areas on the SEM image[106].

PEN Crystal

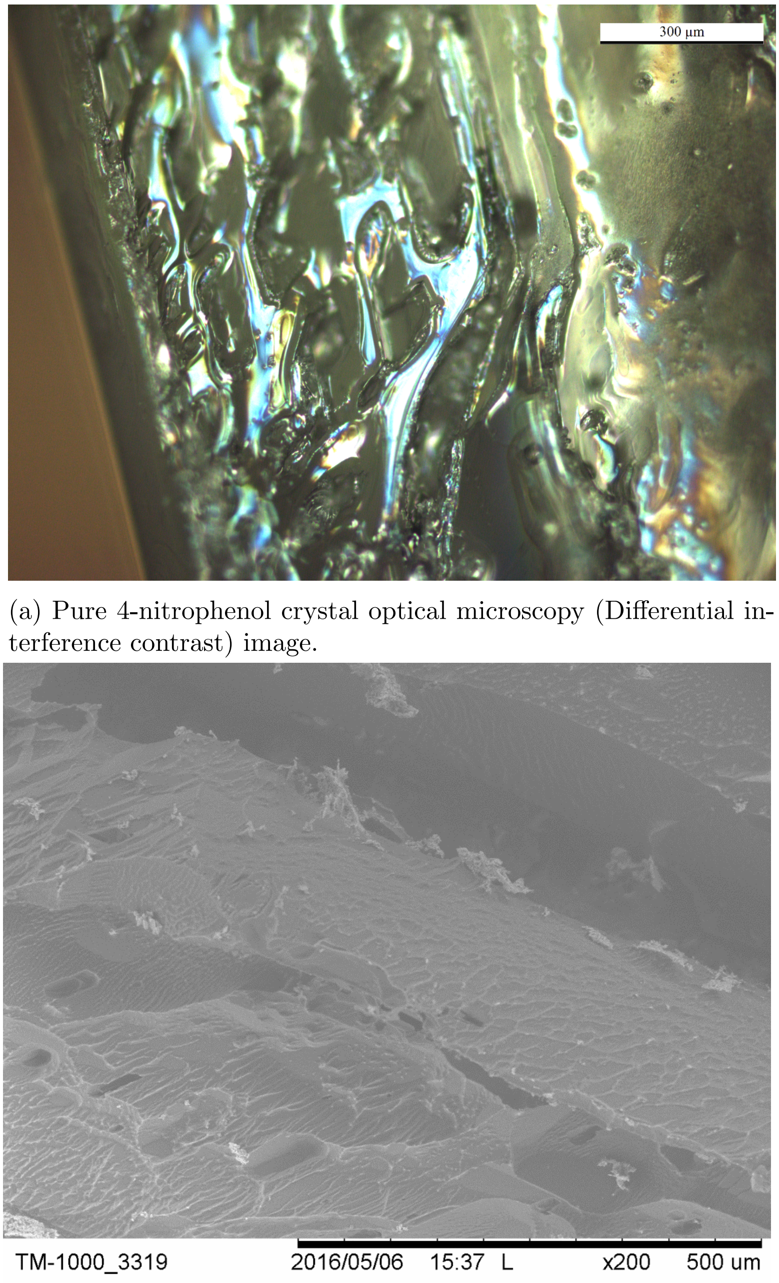
The PEN crystal stereomicroscopy image (Figure 4.5) highlights a typical paracetamol crystal with atypical needle-like features that appear as bright lines. While difficult to observe in the SEM image, these crystal growths can be easily distinguished against the bright paracetamol step structures as darkened needle-like features that have crystallised on top of the paracetamol crystal surface[106].



(a) Pure paracetamol crystal optical microscopy (Differential interference contrast) image

(b) Pure paracetamol crystal SEM image

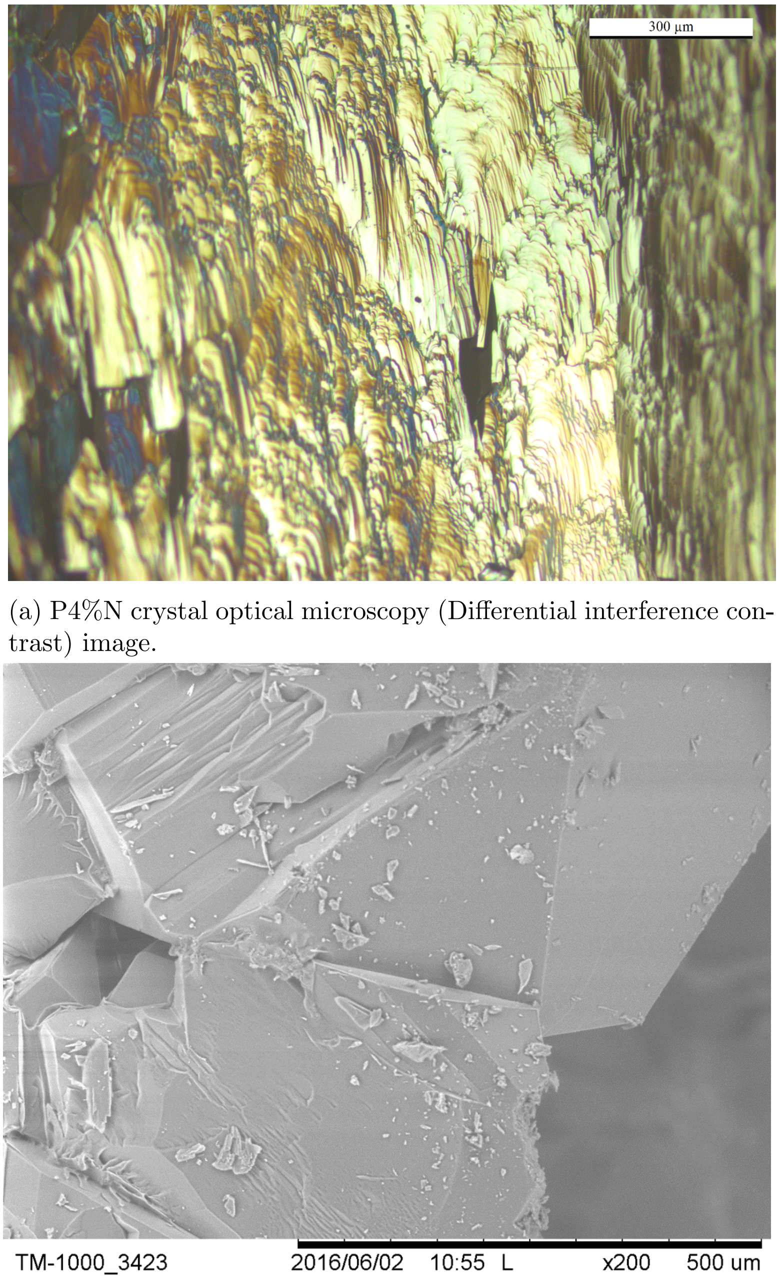
Figure 4.6: Pure paracetamol crystal OM and SEM images.



(a) Pure 4-nitrophenol crystal optical microscopy (Differential interference contrast) image.

(b) Pure 4-nitrophenol crystal SEM image.

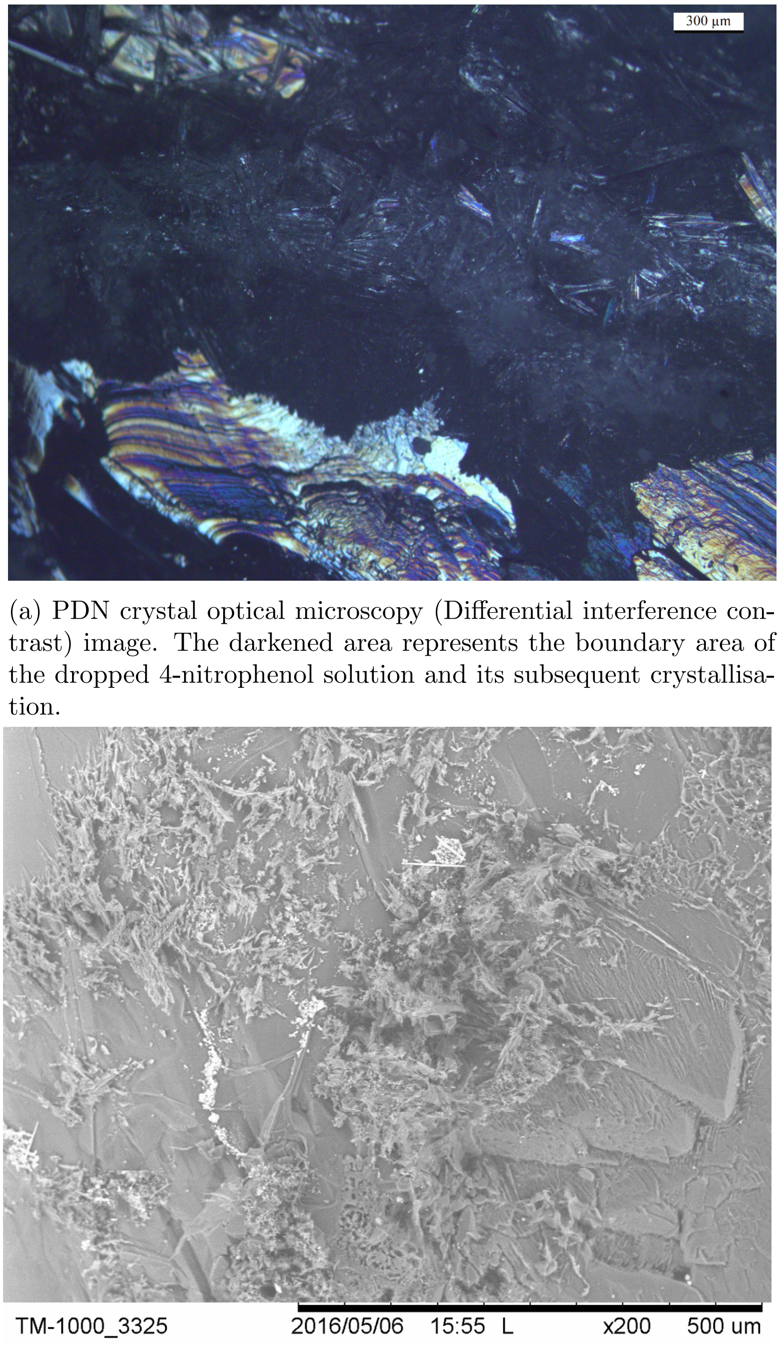
Figure 4.7: Pure 4-nitrophenol crystal OM and SEM images.



(a) P4%N crystal optical microscopy (Differential interference contrast) image.

(b) P4%N crystal SEM image.

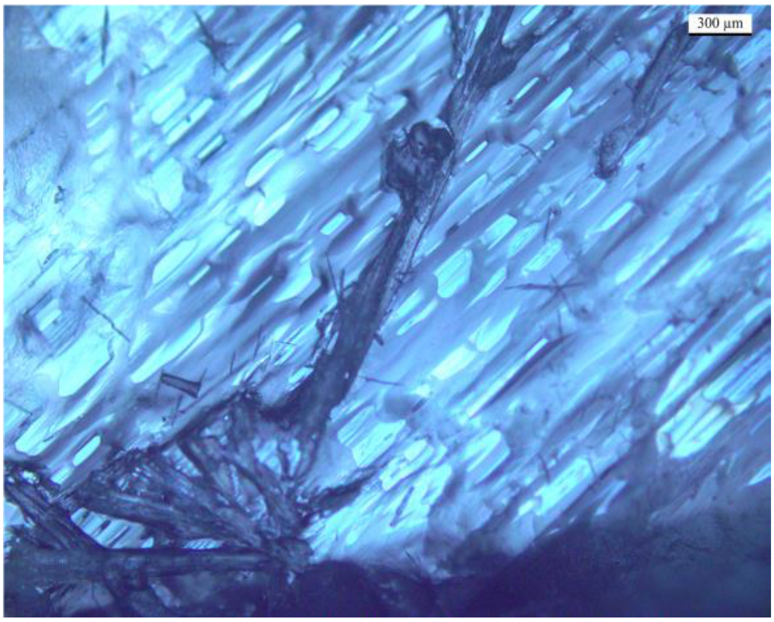
Figure 4.8: P4%N OM and SEM images.



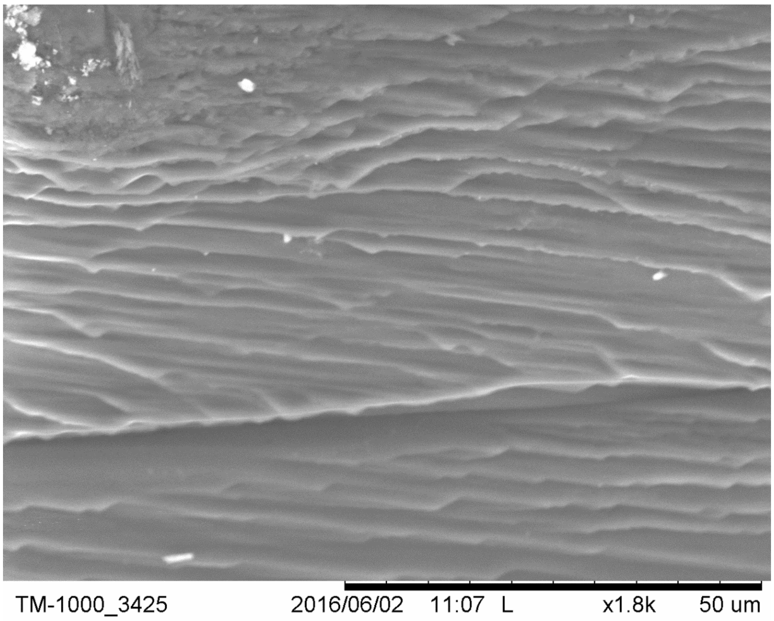
(a) PDN crystal optical microscopy (Differential interference contrast) image. The darkened area represents the boundary area of the dropped 4-nitrophenol solution and its subsequent crystallisation.

(b) PDN crystal SEM image. It shows needle-like growths on top of the paracetamol crystal surface.

Figure 4.9: PDN OM and SEM images.



(a) PEN crystal optical microscopy (Differential interference contrast) image. Needle-like darkened features based on 4-nitrophenol epitaxy can be observed.



(b) PEN crystal SEM image. It displays the paracetamol steps described in the pure paracetamol crystal segment with a few small brighter structures possibly depicting 4-nitrophenol crystals.

Figure 4.10: PEN OM and SEM images.

4.3.2 ToF-SIMS

Impurity Loading Studies

Before confirming 4-nitrophenol as the impurity of choice, a preliminary impurity screening was carried out. To examine the viability of the available impurities, powder sample mass spectra were taken. In addition to the 4-nitrophenol, the other impurities analysed were acetanilide, 4-chloroacetanilide, methyl-4-hydroxybenzoate, acetamidophenol, 4-hydroxyacetophenon, 4-acetamidobenzoic acid, metacetamol and 4-acetoxyacetanilide (see Appendix B).

To aid in the identification of the available fragments from each compound, spectra from other sources were accessed for comparison: spectra were obtained using the NIST 08 MS Demo and AMDIS (Automated Mass Spectral Deconvolution and Identification System) 2.6, which both are offered by the National Institute of Standards and Technology (NIST). The NIST mass spectrometry data centre develops and provides tools related to mass spectral data sharing, spectra and fragment identification for spectra based on gas and liquid chromatography. Although the data are not directly comparable to the ToF-SIMS spectra acquired, they are good indicators for potential fragmentation patterns.

As previously stated, this screening analysis helped to choose 4-nitrophenol and 4-acetamidobenzoic acid as the most suitable impurities to study, since the crystals in question were easy to produce, readily available and were the only impurities that did not have critical fragment ion overlap with the observed paracetamol ions. The spectra of all nine impurity compounds from the screening are included in Appendix B. For the impurity loading scenarios, 4-nitrophenol was chosen over 4-acetamidobenzoic acid as it is a reagent in the synthesis of paracetamol and therefore more prevalent.

Using AMDIS 2.6, the ten most intense peaks from paracetamol and 4-nitrophenol were identified and are shown in Table 4.1. All the given fragments and molecular ions are presented in the neutral state. These peaks were used for the identification of peaks in the ToF-SIMS spectra.

Table 4.1: AMDIS 2.6 output for paracetamol and 4-nitrophenol for the 10 largest predicted neutral fragments and molecular ions.

Paracetamol Peaks	Possible Assignment	4-Nitrophenol Peaks	Possible assignment
43	CHNO	38	C_2N
52	C_3H_2N	39	C_2HN
53	C_3H_3N	53	C_3H_3N
80	C_5H_4O	62	$C_4H_5NO_3$
81	C_5H_6O	63	CH_5NO_2/C_4HN
108	$C_6H_4O_2/C_6H_6NO$	65	C_4H_3N
109	C_6H_7NO	81	C_5H_6O
110	$C_5H_4NO_2/C_6H_6O_2$	93	C_6H_5O
151	Paracetamol	109	$C_5H_3NO_2/C_66H_5O_2$
152	Paracetamol(C^{13})	139	4-Nitrophenol

Viewing the positive and negative spectra for pure paracetamol (blue) and pure 4-nitrophenol (orange) in Figures 4.11 and 4.12, the molecular ion peaks for 4-nitrophenol can be clearly seen at *m/z* 140 for $[M+H]^+$ and *m/z* 138 for $[M-H]^-$, respectively. The same can be said for the molecular ion peaks for paracetamol at *m/z* 152 for $[M+H]^+$ and *m/z* 150 for $[M-H]^-$, respectively. Comparing the spectra in closer detail, most of the more intense peaks overlap between the compounds. Regrettably, a later analysis revealed that in both polarities a very weak peak can be observed at the molecular ion peak of 4-nitrophenol (*m/z* 140 (+), *m/z* 138 (-)) in the paracetamol spectra, however it was not possible to identify the source of these overlapping peaks. With the majority of peaks showing strong overlap in both polarities, the negative polarity was chosen due to the slightly higher intensity of the 4-nitrophenol molecular ion, as well as easier calibration and acquisition behaviour typically observed in negative ion mode.

Due to the overlapping peak issues a comparison between representative spectra of pure paracetamol and the P4%N crystal was required. As can be seen in Figure 4.13, the peak intensity at *m/z* 138, representing the molecular ion peak of 4-nitrophenol in the orange spectrum, is more than four times higher for the impure P4%N crystal spectrum than the pure paracetamol crystal version (4451 au versus

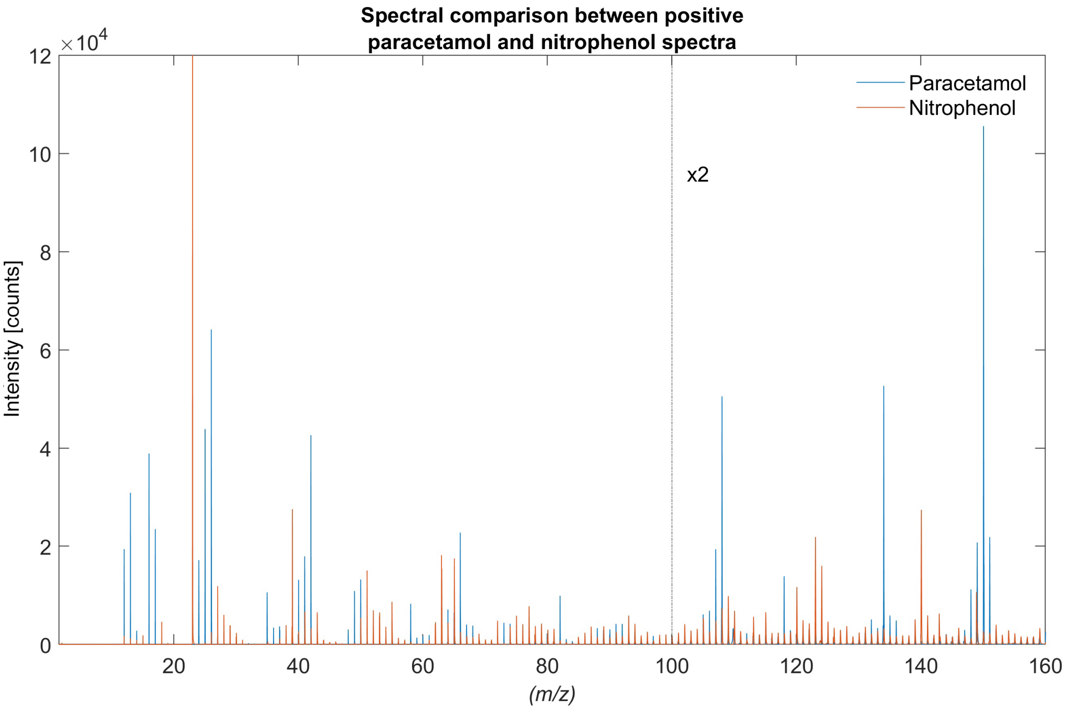


Figure 4.11: Overlay of representative positive spectra of pure paracetamol (blue) and pure 4-nitrophenol (orange).

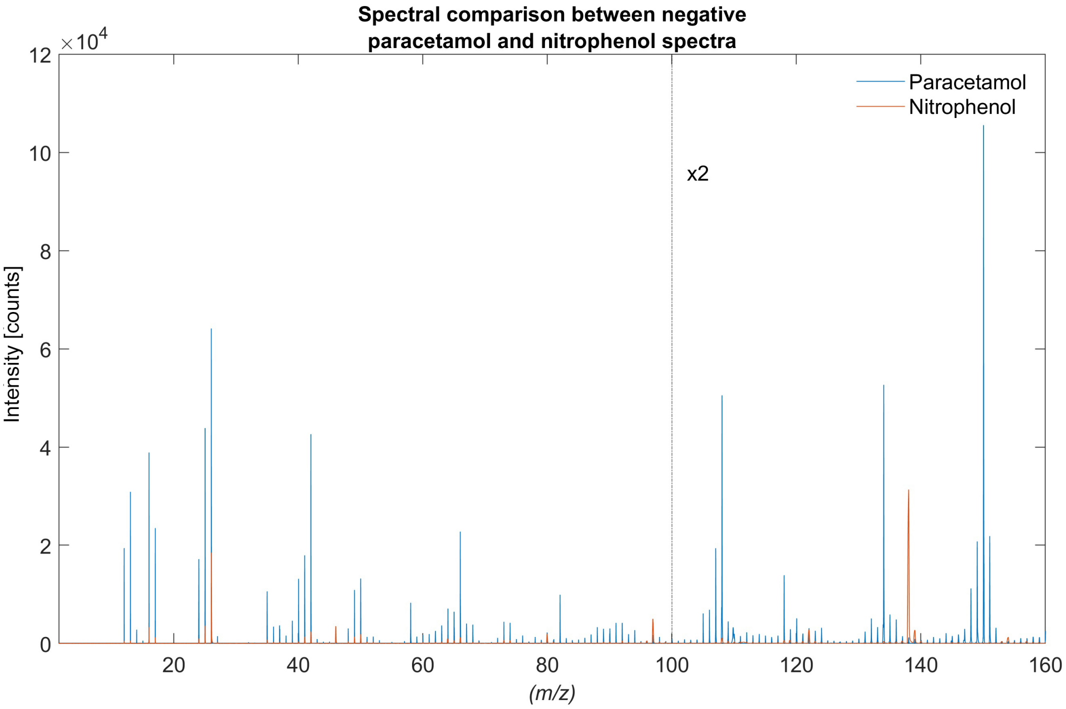


Figure 4.12: Overlay of representative negative spectra for pure paracetamol (blue) and pure 4-nitrophenol (orange).

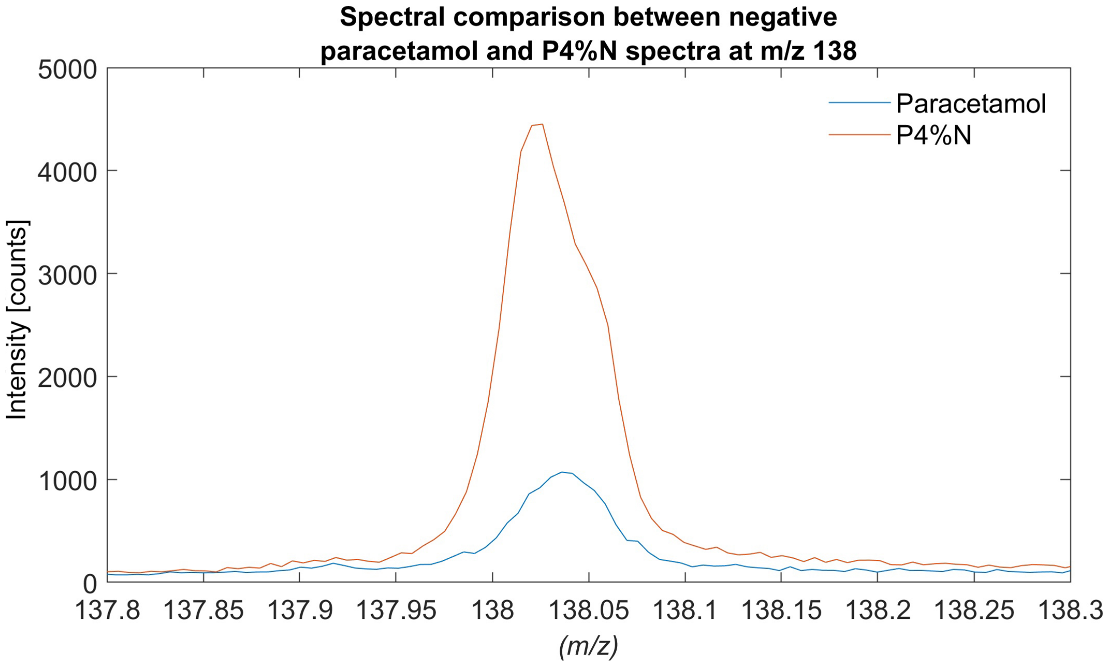


Figure 4.13: Overlay comparison between representative negative spectra for a pure paracetamol (blue) and an impure paracetamol crystal with 4%mol 4-nitrophenol (orange) at m/z 138, the molecular ion peak of 4-nitrophenol.

1070 au, where au stands for arbitrary units) with the latter falling close to baseline levels. Comparing the peak heights to the paracetamol molecular ion peak at m/z 150, the m/z 138 signals can be expressed as 2.5% and 0.6% of the paracetamol peak height, respectively. The 4-nitrophenol intensity level in the impure crystal spectrum is thus sufficiently intense to easily track the impurity at the given concentration level.

Impurity Load 1: Paracetamol crystal with a drop of 4-nitrophenol

Figure 4.14 shows images of the intensity of the molecular ion peaks of 4-nitrophenol and paracetamol as well as an overlay of both acquired from the surface of the PDN crystal. From the 4-nitrophenol ion image, it is clearly visible that the impurity has mostly covered the pure paracetamol crystal and only small areas of pure paracetamol crystal are directly visible where lower ion intensity counts of 4-nitrophenol are observed. Comparing this to the paracetamol ion image, the inverse areas of high 4-nitrophenol ion intensity counts show high intensities for paracetamol. This can be seen even

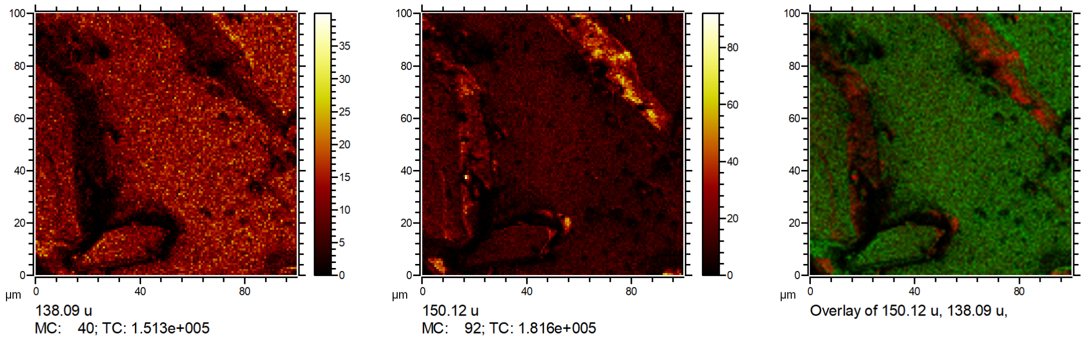


Figure 4.14: Pure paracetamol crystal with a concentrated drop of 4-nitrophenol: Images from left to right, ion image for 4-nitrophenol at m/z 138, ion image for paracetamol at m/z 150 and RGB overlay of both previous images, paracetamol in red, 4-nitrophenol in green.

more clearly when comparing the ion images in an overlay of paracetamol (red) and 4-nitrophenol (green). A number of explanations can account for these observations: first, the dropped and recrystallised impurity layer was very thin and did not cover the whole surface when it dried; second, parts of the pure paracetamol crystal might have dissolved and mixed with the dropped 4-nitrophenol solution and creating pockets of paracetamol as it recrystallised; and third, the crystal surface was rough enough to keep some elevated ridges of paracetamol free from the impurity as the dropped 4-nitrophenol solution settled in the crevasses.

The images were obtained using burst alignment mode specifically catered towards high lateral resolution at the expense of spectral peak resolution as can be seen by the broad peaks in the corresponding mass spectra shown in Figure 4.15 (0.05 u versus 1 u). The intensities on the mass spectrum for the peaks at m/z 138 (4-nitrophenol) and m/z 150 (paracetamol) show that both paracetamol and 4-nitrophenol are present at high intensities which in this case can likely be equated to high surface concentrations.

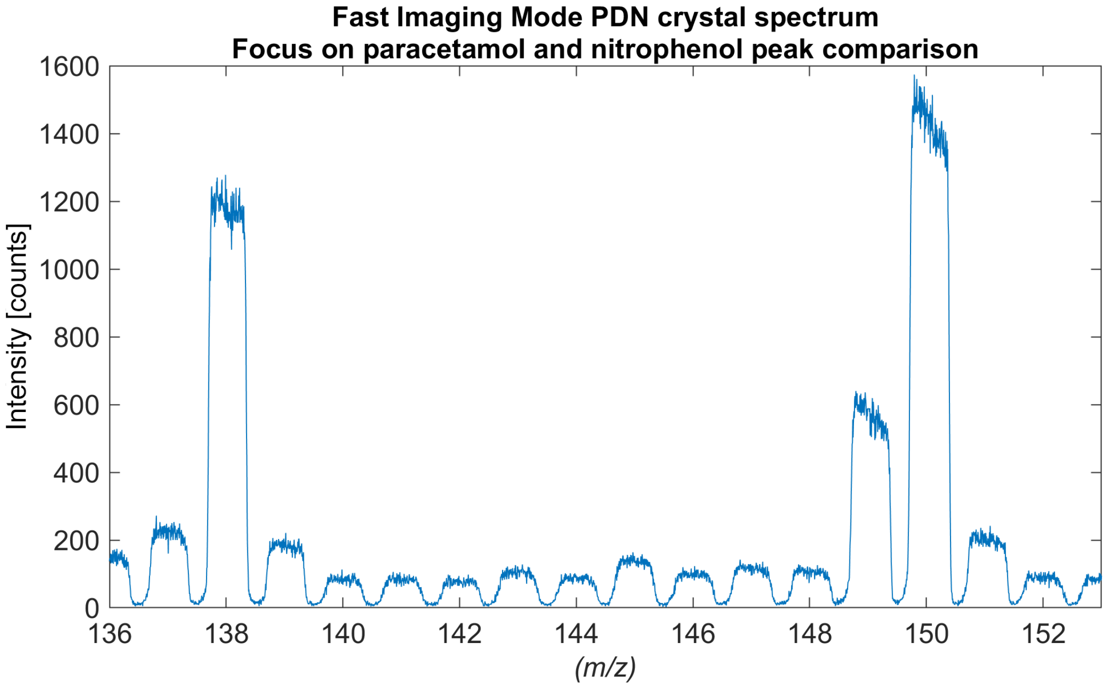
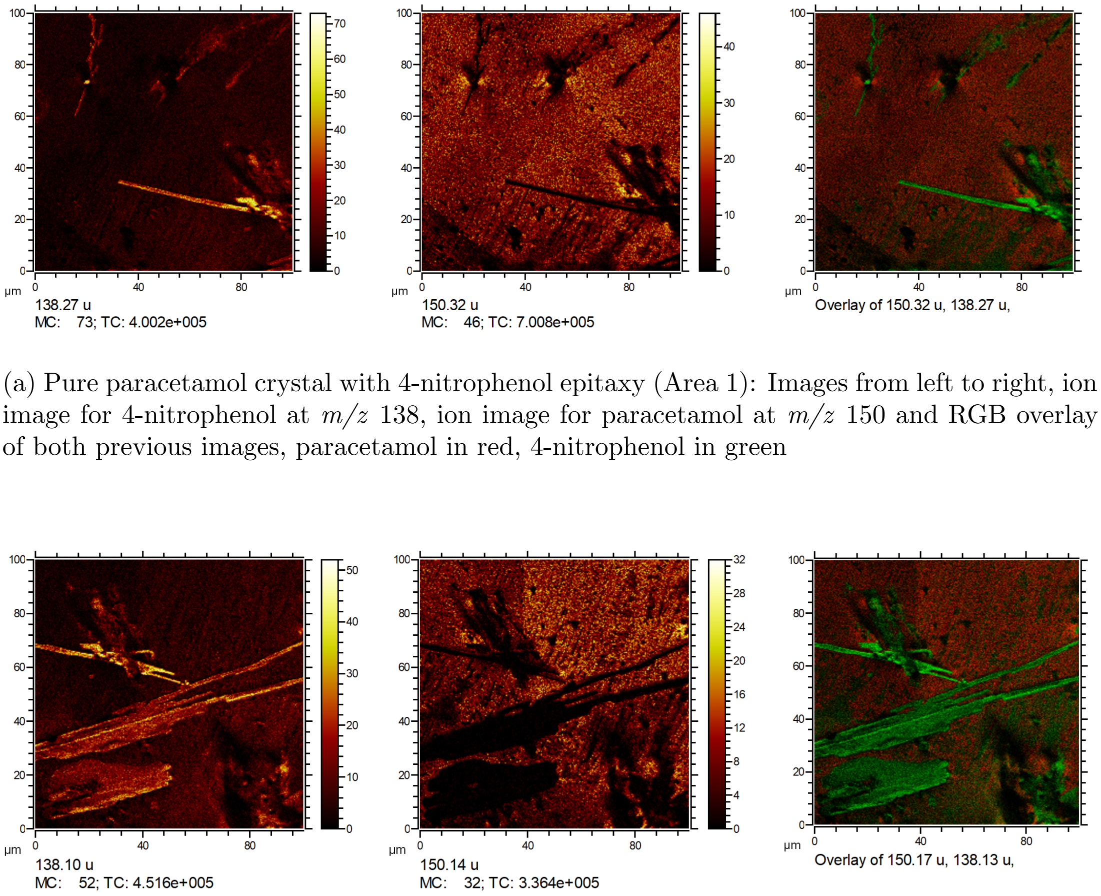


Figure 4.15: ToF-SIMS high lateral mode mass spectrum of PDN crystal.

Impurity Load 2: Paracetamol crystal with epitaxial growth of 4-nitrophenol
Figures 4.16a and 4.16b depict the ion images for the molecular ions of paracetamol and 4-nitrophenol as observed from the paracetamol crystal with a 4-nitrophenol epitaxy. Two different areas on the crystal surface are shown.

From the ion images for the two distinct molecular ions of paracetamol and 4-nitrophenol, one can observe that the high-intensity areas of 4-nitrophenol and paracetamol are complementary for both surface areas under investigation. This is visually represented in the RGB overlay image of both molecular ions. Interestingly, comparing Figures 4.16a and 4.16b, the mass spectra as well as the distribution of 4-nitrophenol differ. While Figure 4.16a displays lower intensity counts for 4-nitrophenol as shown in the blue spectrum in Figure 4.17, Figure 4.16b displays a much larger spread of 4-nitrophenol over the surface of the paracetamol crystal, leading to a significant reduction in paracetamol peak intensity but interestingly only a minor increase in the 4-nitrophenol peak intensity (as shown in the orange spectrum in Figure 4.17). While a decrease in paracetamol is expected when viewing the spread of the impurity, one



(a) Pure paracetamol crystal with 4-nitrophenol epitaxy (Area 1): Images from left to right, ion image for 4-nitrophenol at m/z 138, ion image for paracetamol at m/z 150 and RGB overlay of both previous images, paracetamol in red, 4-nitrophenol in green

(b) Pure paracetamol crystal with 4-nitrophenol epitaxy (Area 2): Images from left to right, ion image for 4-nitrophenol at m/z 138, ion image for paracetamol at m/z 150 and RGB overlay of both previous images, paracetamol in red, 4-nitrophenol in green

Figure 4.16: PEN ion image overlays.

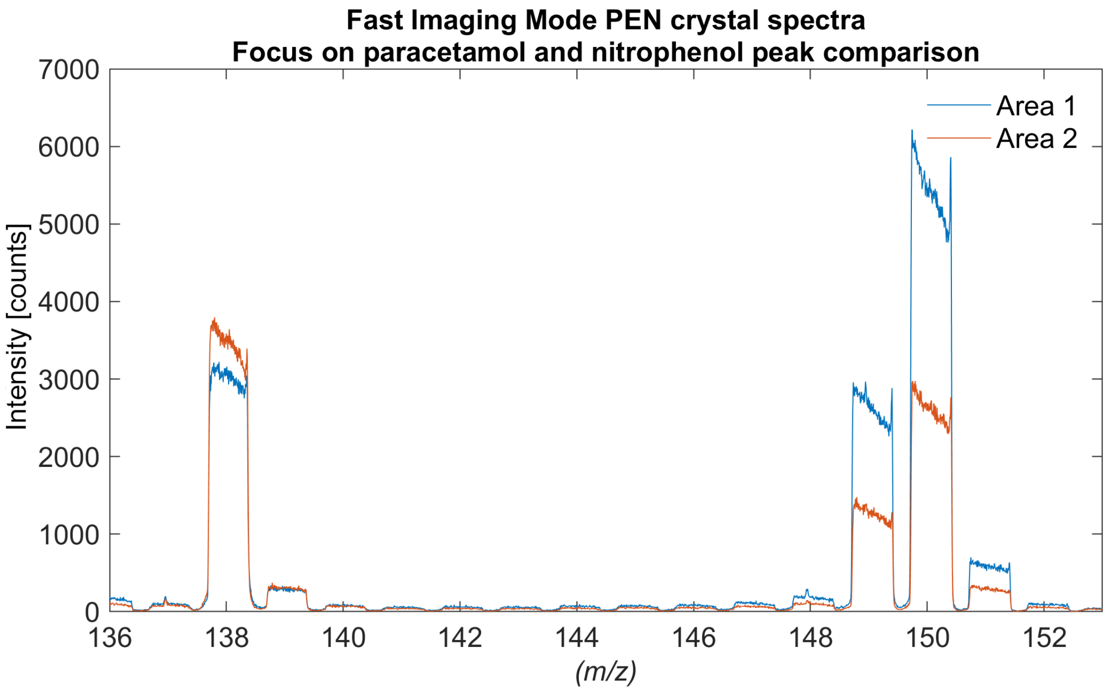


Figure 4.17: ToF-SIMS high lateral mode mass spectra of PEN crystal area 1, (blue) and area 2, (orange)

would also expect a larger increase in the impurity than is observed.

In comparison to the PDN crystal, where a highly concentrated 4-nitrophenol solution was purposefully dropped onto a pure paracetamol crystal area, here a pure paracetamol crystal was dipped into a 4-nitrophenol solution and epitaxial crystal growth could be observed and imaged using ToF-SIMS. Additionally, residue from the 4-nitrophenol was observed on most of the surface of the paracetamol crystal stemming from the epitaxy crystallisation. In this context it is interesting to see that the intensity of the residue around the impurity crystal areas is higher than in the regions further away from the crystal (4.16a and 4.16b).

Impurity Load 3: Impure paracetamol crystal with 4%mol 4-nitrophenol Figure 4.18 illustrates the ion images for the molecular ions of paracetamol and 4-nitrophenol as acquired from the P4%N crystal. Viewing the molecular ion image of 4-nitrophenol, one can see a much lower mass count detected in comparison to the previous images (mass count values: PDN 40, PEN 73/52, P4%N 12) and in particular in comparison to the paracetamol ion image (mass count values: PDN 92, PEN 46/32, P4%N 106). This is to be expected as, contrasting to the previous samples, only a fraction of 4-nitrophenol has been used for the crystallisation of the sample. Nevertheless, 4-nitrophenol appears to be present over the entire surface area of the crystal.

The severe decrease in 4-nitrophenol present on the surface is very notable looking at the mass spectrum of the sample at hand. The representative peak at m/z 138 is barely visible but still present, showing at approximately 2.8% of the paracetamol peak height, similar to the previous P4%N peak ratio observed during the initial peak comparison, suggesting this peak indeed represents the 4-nitrophenol impurity.

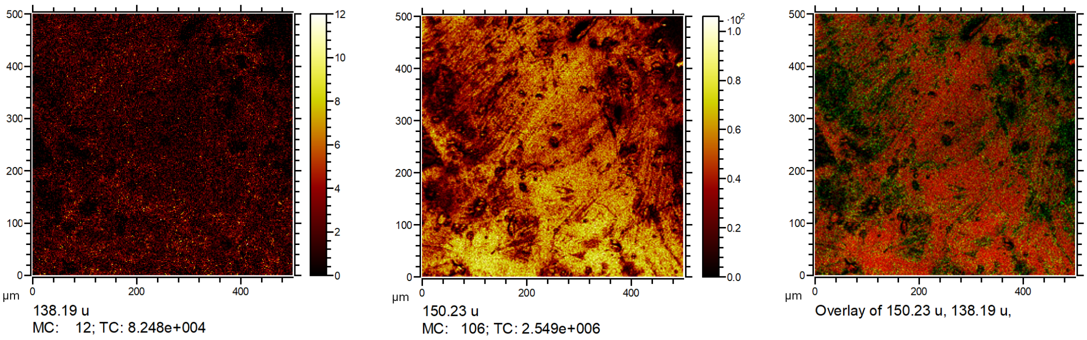


Figure 4.18: P4%N crystal images from left to right, ion image for 4-nitrophenol at m/z 138, ion image for paracetamol at m/z 150 and RGB overlay of both previous images, paracetamol in red, 4-nitrophenol in green (image intensity of green increased).

Figure 4.18 (138.19 u) shows an even 4-nitrophenol distribution which is expected to be consistent throughout the crystal. This may be confirmed via bulk analysis through depth profiling or 3D-imaging.

The results in this section show that, using ToF-SIMS, not only was it possible to identify the impurity but also graphically represent the distribution of the impurity on the crystal surfaces. With impurities that show distinct ion fragments, ToF-SIMS

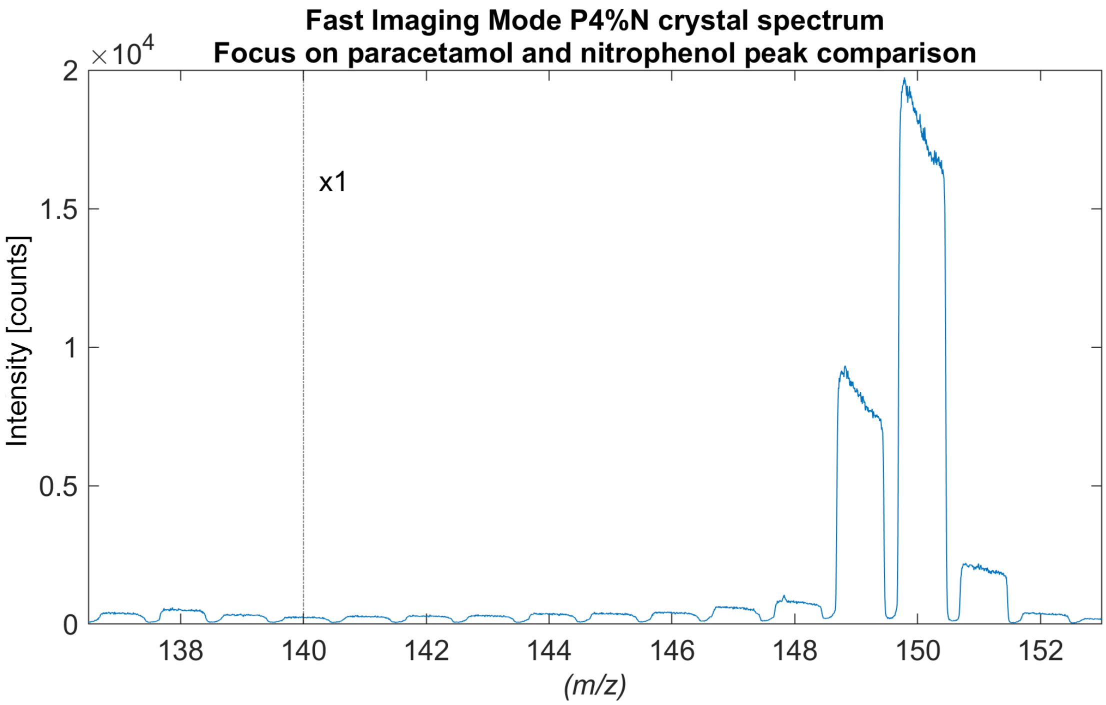


Figure 4.19: ToF-SIMS high lateral mode mass spectrum of P4%N crystal. The intensity scale was halved past m/z 140 due to the large intensity differences between the paracetamol-related molecular ion fragment peaks and the 4-nitrophenol molecular ion fragment peak.

can thus be a very useful exploratory tool for both chemical characterisation and distribution assessments. Structural assessments, while not quantitative, can be possible using the ion-induced secondary electron imaging function which, unfortunately, was not attempted in this study. With additional development in crystal mounting for ToF-SIMS analysis, easier ways to analyse multiple crystal surfaces of a single crystal may be possible, and ToF-SIMS can become a suitable technique to examine not only crystal surfaces but also investigate surface-to-bulk properties.

Bulk Viability Study

With the intent to enable future bulk studies of crystals, preliminary analyses to study the impurity distribution in crystals between the surface and bulk were carried out. For this analysis, two different faces of a paracetamol crystal with 4%mol 4-acetamidobenzoic acid were chosen and analysed using depth profiling. It is also a common impurity found in paracetamol synthesis, and the mass spectra showed little-to-no overlap among the characteristic peaks between the two compounds.

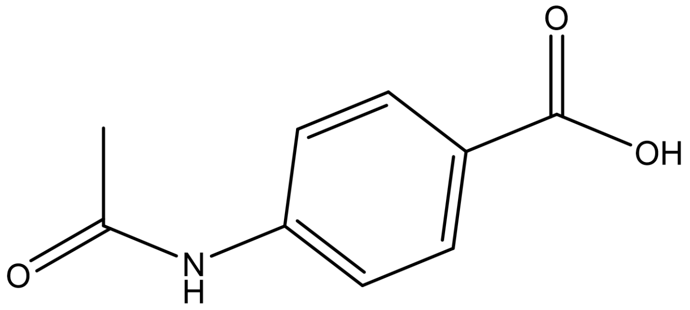
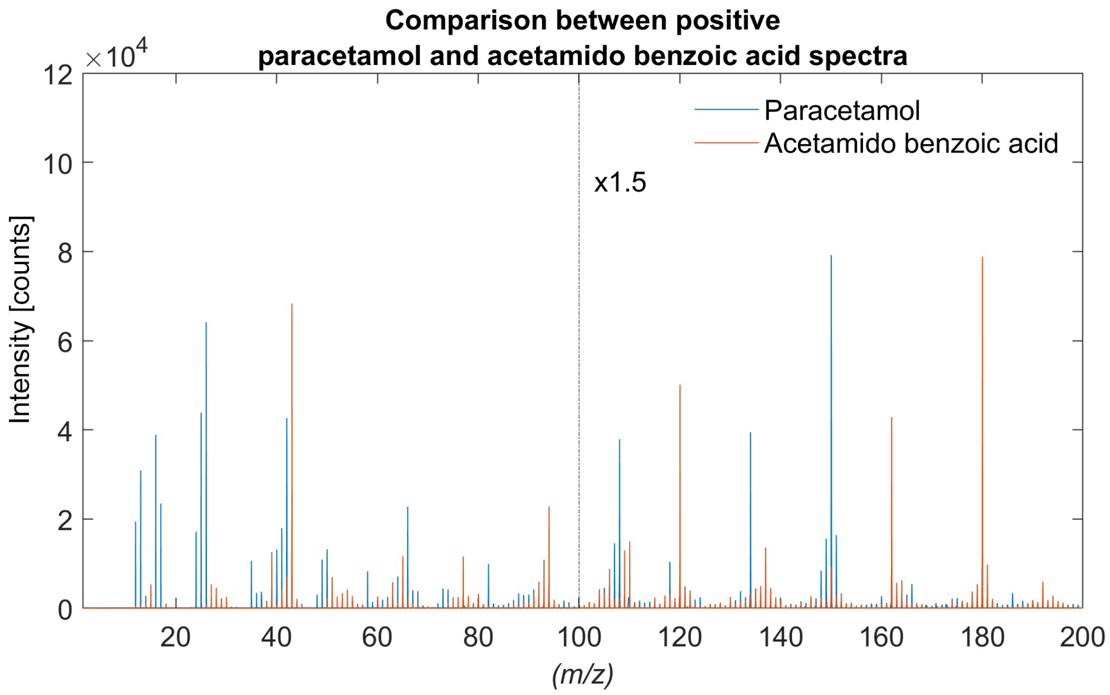


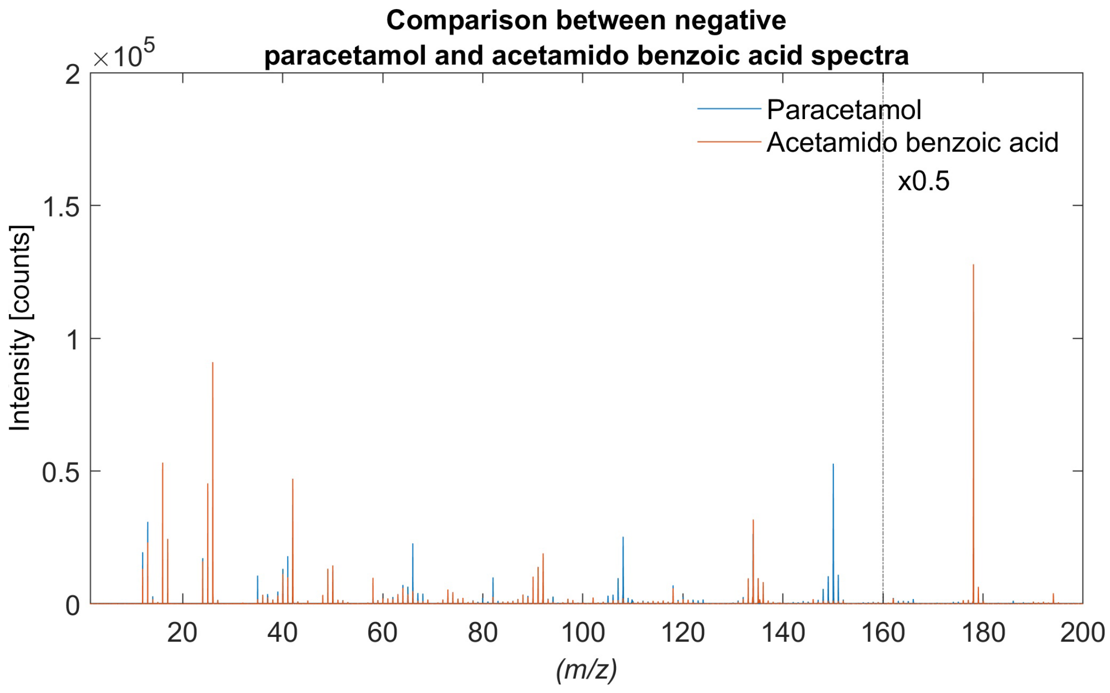
Figure 4.20: Molecular structure of 4-acetamidobenzoic acid.

As the positive and negative mass spectra comparisons of pure paracetamol and 4-acetamidobenzoic acid samples in Figure 4.21a show, the paracetamol and 4-acetamidobenzoic acid molecular ion peaks, *m/z* 152 and 180 respectively, can be well distinguished between the two substances. With a significantly stronger ionisation potential of 4-acetamidobenzoic acid in the negative mode, the intensity of the peaks in positive ion mode were more comparable and therefore this mode was chosen for further studies using delayed extraction imaging and sputter analysis.

With the molecular ion peaks at *m/z* 152 and 180 so clearly distinct and intense, these are the only peaks that have been focused on in this study. A spectral analysis of a crystal comprising paracetamol with 4%mol 4-acetamidobenzoic acid impurity appears very different to the analysis of 4%mol 4-nitrophenol impurity containing crystal (Figure 4.22). The 4-acetamidobenzoic acid molecular ion peak at *m/z* 180 is showing a substantially higher intensity than was anticipated (approximately half the intensity height of paracetamol) based on the small fraction of added impurity, suggesting a large amount of the impurity seen on top of the surface and/or the impurity ionising better than 4-nitrophenol. This behaviour is evident on both faces of the analysed crystal (Figure 4.22).



(a) Overlay of positive spectra for paracetamol (blue) and 4-acetamidobenzoic acid(orange).



(b) Overlay of negative spectra for paracetamol (blue) and 4-acetamidobenzoic acid(orange).

Figure 4.21: 4-acetamidobenzoic acid spectral overlays.

It can be observed even more clearly viewing the ion images obtained for the paracetamol and 4-acetamidobenzoic acid peak additions for both crystal faces. Peak additions are combinations of the most intense molecular ion attributable paracetamol peaks (*m/z* 151, 152, 153) and 4-acetamidobenzoic acid peaks (*m/z* 179, 180, 181). While paracetamol is distributed uniformly, as can be expected from a paracetamol crystal, the impurity appears to be deposited in larger high intensity patches across the surfaces of the crystal.

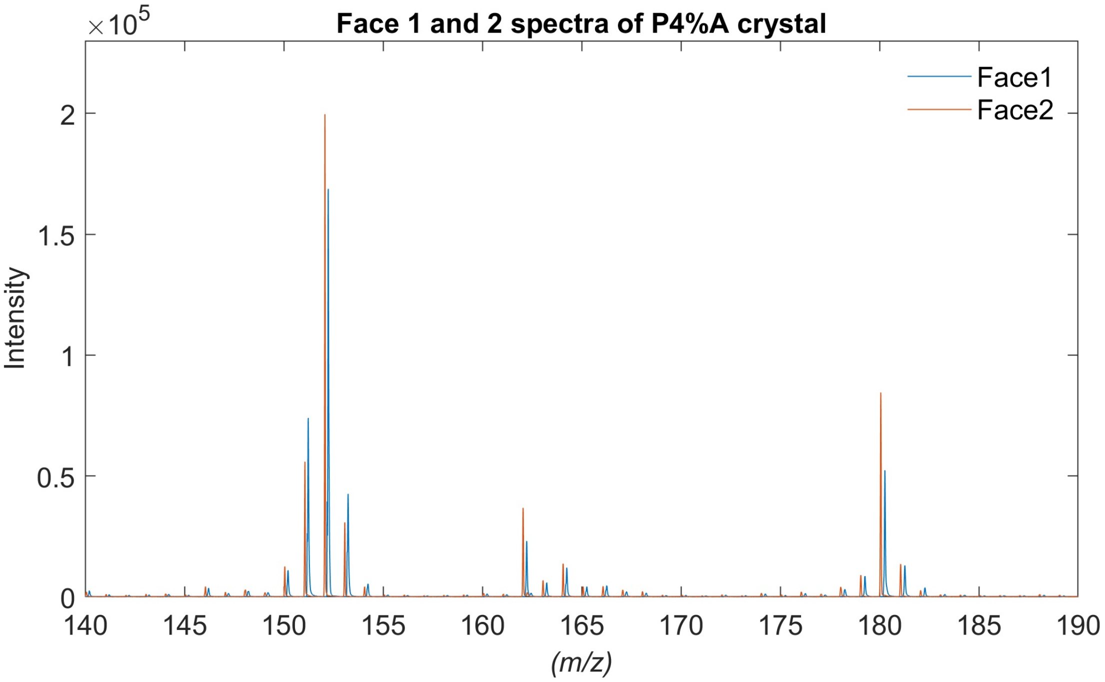


Figure 4.22: Positive Face 1 and 2 spectra from a P4%A crystal. Detail between *m/z* 140 to *m/z* 190 is shown to emphasise the molecular ion peaks of paracetamol (*m/z* 152) and acetamidobenzoic acid (*m/z* 180).

Using an $Ar_{1500}{}^{+}$ sputter beam, one of the crystal faces was depth profiled. The same molecular ion peaks were used to trace the distribution of paracetamol and 4-acetamidobenzoic acid at different crystal depths - paracetamol peaks *m/z* 151, 152, 153, 4-acetamidobenzoic acid peaks *m/z* 179, 180, 181. The profile in Figure 4.24 confirms what was implicated in the ion images: the impurity appears in larger concentrations on the crystal surface but after a few sputtering scans the impurity intensity decreases by at least one order of magnitude. Due to an instrument error, the second

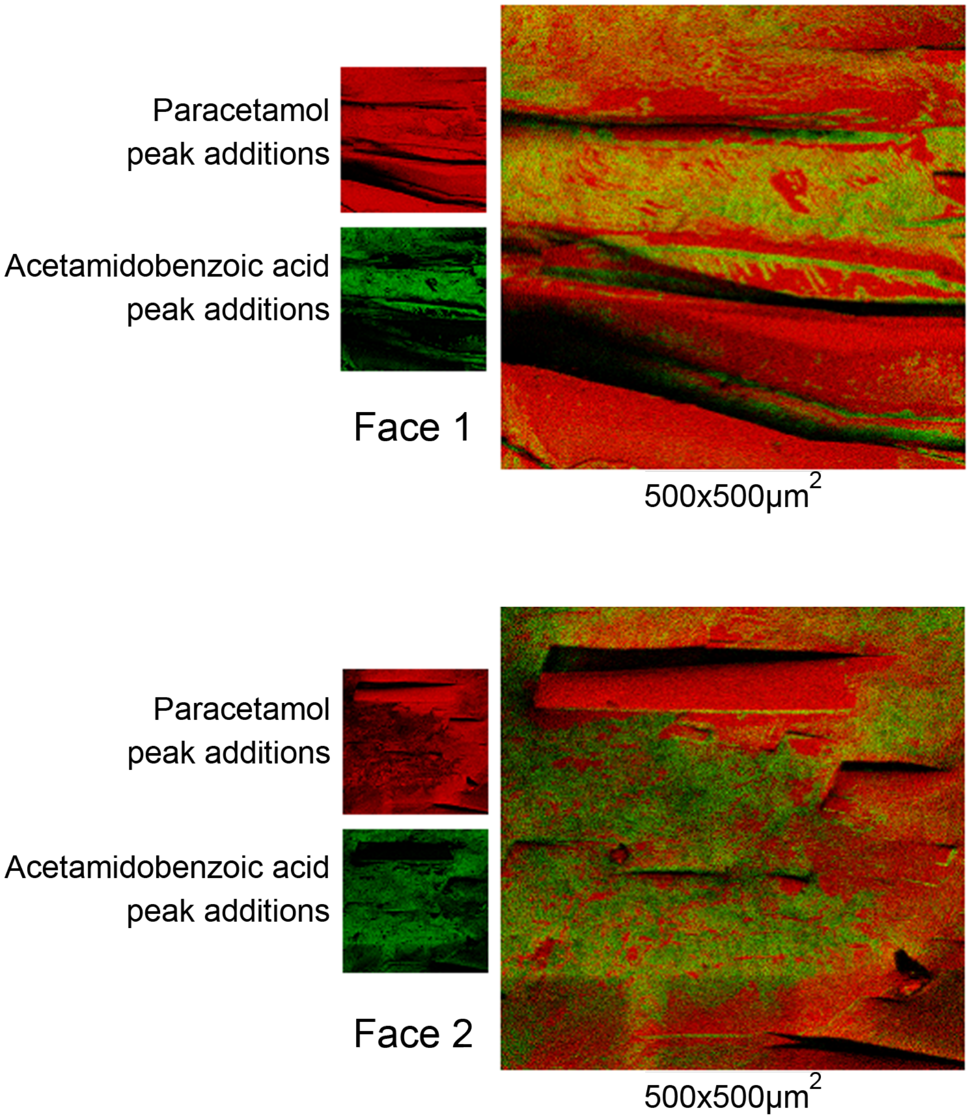

Paracetamol peak additions
Acetamidobenzoic acid peak additions
Face 1
500x500µm^2
Paracetamol peak additions
Acetamidobenzoic acid peak additions
Face 2
500x500µm^2

Figure 4.23: Positive Face 1 and 2 overlay ion images from a P4%A crystal analysis. Peak additions of the most intense molecular ions attributable to paracetamol (m/z 151, 152, 153) and acetamidobenzoic acid peaks (m/z 179, 180, 181) were combined into an overlay where they are presented in red and green respectively. Face 2 exhibits 2 small previously sputtered areas that did not influence the results of the test.

crystal face could not be depth profiled. Instead, at a later time, a second P4%A (P4%A.2) crystal was analysed with depth profiling, measuring two of the crystal faces (Face 1, Face 2), the results of which are shown as a depth profile in Figure 4.25.

Both crystal depth profiles show the same behaviour. After a short sputter duration, the impurity intensity decreases drastically, while the paracetamol intensity increases with the impurity being sputtered away from the surface. This indicates that the large impurity deposit was mostly surface-based as a short sputter duration only removes a small amount of material. No studies have been performed to analyse the distribution of 4-nitrophenol through the P4%N crystal however it would make for a good comparison if future studies were attempted.

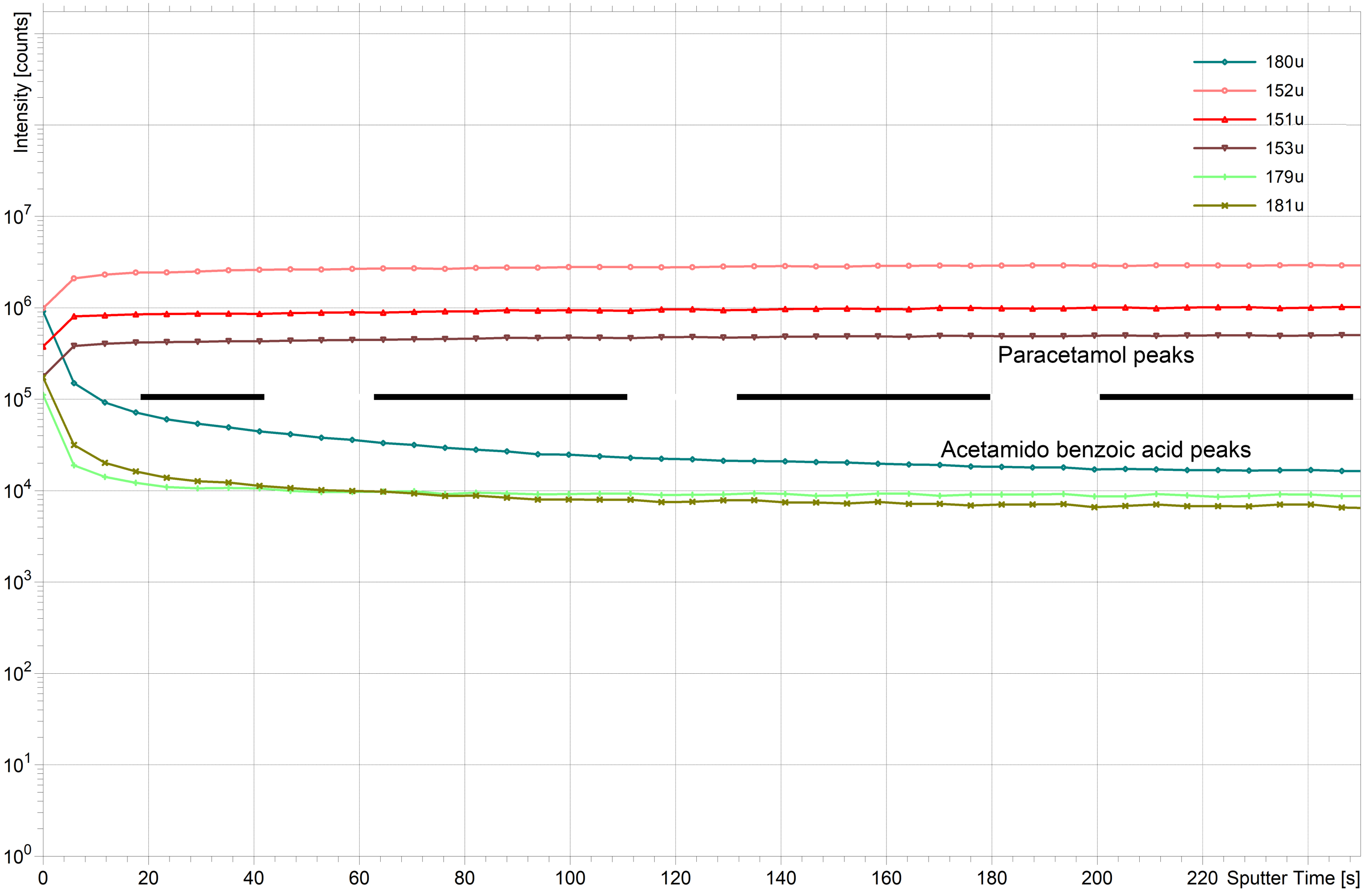


Figure 4.24: Positive Face 1 P4%A depth profile. Peak additions of the most intense molecular ion attributable paracetamol peaks (m/z 151, 152, 153) and 4-acetamidobenzoic acid peaks (m/z 179, 180, 181) were used to track paracetamol and 4-acetamidobenzoic acid distributional differences from surface to bulk.

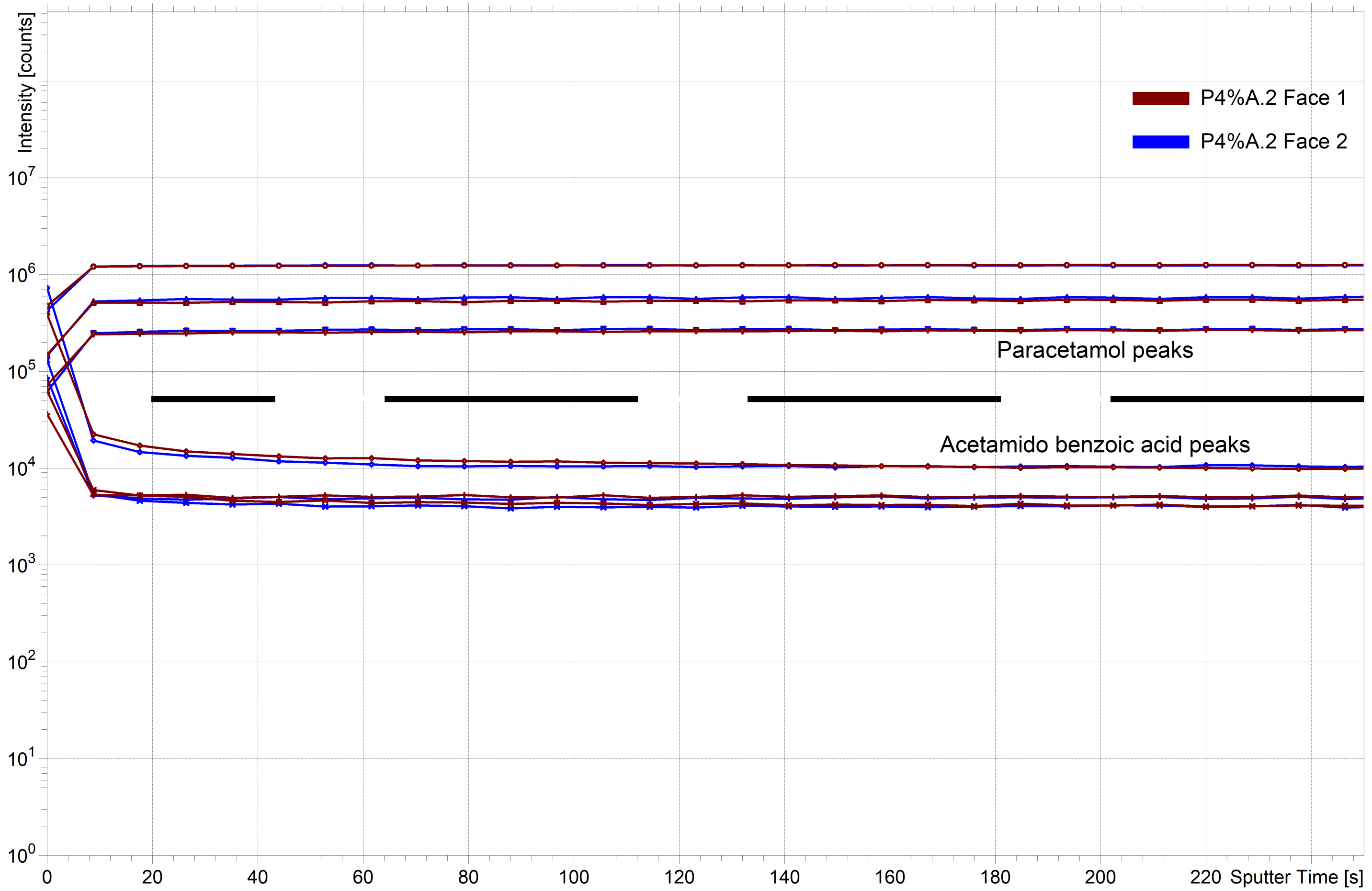


Figure 4.25: Positive Face 1 and Face 2 P4%A depth profiles. Peak additions of the most intense molecular ion attributable paracetamol peaks (m/z 151, 152, 153) and 4-acetamidobenzoic acid peaks (m/z 179, 180, 181) were used to track paracetamol and 4-acetamidobenzoic acid distributional differences from surface to bulk.

4.4 Discussion and Future Work

ToF-SIMS has been used as part of a wider study comparing different analytical techniques applied to pharmaceutical material characterisation. In this context, ToF-SIMS was used to analyse the surfaces of paracetamol crystals with different impurity loadings and demonstrate the feasibility of the technique with regards to its ability to provide chemical and topographical characterisation simultaneously. To extend this study from the surface into the bulk, depth profiling of an impure paracetamol crystal was performed, displaying the usefulness and ability of the instrument to not only perform surface but also bulk characterisation of pharmaceutical materials.

Initially, the analysis of ten common paracetamol impurities indicated only two impurities that generated unique ions that did not overlap with the paracetamol ions, 4-nitrophenol and 4-acetamidobenzoic acid, making them ideal analysis targets for this study. The remaining impurities had significant overlap with paracetamol, which can largely be attributed to the fact that most of those compounds represent building blocks, fragments or by-products of paracetamol. Studies into those impurities would likely require isotopic labelling of either paracetamol or the impurities prior to crystallisation.

In the first part of the study, three paracetamol crystals with different impurity depositions methods were tested, a pure paracetamol crystal with 4-nitrophenol impurity (PDN) drop-deposited on the surface, a pure paracetamol crystal with 4-nitrophenol epitaxial growth (PEN) and an impure paracetamol crystal with 4%mol 4-nitrophenol impurity integrated into the crystal structure (P4%N). While ToF-SIMS should be able to qualitatively identify impurities at lower concentrations than exemplified in this study, the impurity concentration was chosen to allow for more significant changes to the crystal surface and morphology due to impurity inclusion as well as more facile analysis and comparison using the other tested characterisation techniques.

Initial challenges with mounting the crystal samples into the instrument were overcome, as previously described, and allowed for significant time savings due to an increase in speed when moving between samples and instrument parameter measurements. Furthermore, using this method the crystal surfaces were more easily aligned with the ion beam. Although alternative mounting methods exist, the described method was found to be the easiest and most risk-free method available and thus was preferred. In addition to challenging crystal shapes, some crystal surfaces showed difficult topographical features with large height differences making standard spectral and imaging analysis difficult. Without adjustment this led to peak shifting and loss of information. The use of the delayed extraction mode for analysis reduced or solved some occurrences of these issues because it resulted in a less severe energy spread of secondary ions of equal mass, greatly reducing the consequences of harsh topography.

Analysis of the crystals using optical microscopy methods and SEM also showed large differences in surface texture between PP, PN and the impurity loaded crystals PDN, PEN and P4%N, respectively. Some of these changes are visible using the ToF-SIMS measurements, e.g. rougher surfaces, but the optical microscopy and SEM results are clearer and offer a more focussed view. The XRD analysis performed by Dr Sara Ottoboni further revealed that the impurity inclusion modified the crystal faces, thus changing the crystal shape[106]. The results concerning modifications to surface texture and shape are in line with published results[104,105]. The impurity additions via dropping and epitaxial growth did not cause such extreme differences, with the PDN crystal exhibiting impurity crystal growth within the dropped area, appearing to grow outwards from the point of incident and the PEN crystal showing small impurity crystals growing on the surface of the paracetamol crystal, as expected. These details could have been missed without comparison with SEM and OM methods, showing that the techniques are complementary. Overall, the combined results are very consistent and show the expected results. Regarding the ToF-SIMS analysis, the impurity could be identified and imaged in all three chosen impurity loadings. While original tests revealed an overlap of a peak at the 4-nitrophenol molecular ion value of m/z 138 in negative mode, the intensity of that peak was significantly smaller than the impurity intensity in question (75% less intense). While 4-nitrophenol can be used in paracetamol synthesis, the procured paracetamol samples were of very high purity making it unlikely that the unknown peak was 4-nitrophenol. Nevertheless, this issue was relevant for the P4%N crystal with its lower 4-nitrophenol loading. Peak ratio measurements of the unknown in a pure paracetamol crystal and measurements taken on P4%N crystals indicate that the data for the P4%N sample corresponds to 4-nitrophenol and not the unknown peak.

Comparing the data from the different crystals, the ion images produced by ToF-SIMS displayed expected surface loadings for the impurity given the respective impurity dosages. The PDN crystal measurements show the impurity largely covering the crystal surface. While no impurity crystals are visible via the ToF-SIMS measurement, optical microscopy results showed crystallisation of the dropped impurity on top of the crystal surface. Interestingly, the molecular ions of both paracetamol and the impurity are visible on the crystal surface. There are multiple explanations for this behaviour. First, paracetamol at the surface could have, upon application of the drop, been partly dissolved and mixed with the impurity to form this new surface cover. Second, the coverage of the impurity layer is very thin which can lead to two possibilities, the ion beam breaches into the paracetamol layer revealing subsurface paracetamol ions or the coverage is so thin that paracetamol can be seen through the lattice cover, which is unlikely.

The PEN crystal measurements show significant 4-nitrophenol epitaxial growth on the paracetamol crystal. Of particular interest in the ion images of this sample are the impurity residues visible surrounding the larger 4-nitrophenol epitaxial structures and also in the rough surface features of the paracetamol crystal. The P4%N crystal measurements displayed a very homogenous impurity distribution over the crystal surface, where impurity molecular ion peaks largely coincide with the paracetamol molecular ion peaks. Although depth profiling has not yet been carried out on this crystal, the surface results would suggest that the introduction of the impurity into the initial supersaturated solution results in an even distribution of the impurity throughout the crystal. This differs to the other impurity loadings analysed which formed crystals on the paracetamol crystal surface.

In addition to the impurity loading study, results from the bulk studies revealed interesting impurity surface behaviour. Ion images and depth profiling analyses performed with the P4%A impurity crystals showed a highly surface localised distribution of the impurity that, after a few sputtering scans, was reduced by at least one order of magnitude. A second P4%A crystal was analysed to confirm these results. This behaviour was very different from the P4%N impurity crystals where the impurity was seemingly distributed uniformly over the surface. Moving from the surface to the bulk of the P4%A crystals, similar impurity distribution and intensities previously observed in the analysis of P4%N crystals were seen in the form of evenly distributed paracetamol and impurity ions. These results have been confirmed across two crystals and two separate faces suggesting this distribution behaviour is similar across the crystals. While no explanation can be given to what exactly has caused the behaviour, factors that could influence this result could stem from the impurity crystallisation behaviour or the crystallisation process itself such as speed of temperature adjustment. This has interesting implications for API and crystal production under conditions known to possibly include impurities. Some impurities might be prone to surround crystal product surfaces at higher concentrations versus the bulk which could help to design better post-purification processes.

Challenges that have been encountered during analysis were, as previously described, topographical effects, overlapping peaks from the paracetamol-related impurities as well as charging effects experienced during analysis of the large crystals. Rough surfaces with varying height make the analysis of materials very difficult but can be addressed with the use of delayed extraction which can drastically improve the signal from topographically challenging samples. However, this problem can be exacerbated when analysing insulating materials such as the crystals analysed. Here, surface height differences can lead to even larger variations in electron field intensity resulting in reduced signal intensity. These effects can be corrected for using the electron flood gun but can be very extreme and at times difficult to handle.

Another challenge encountered was a strong overlap between the characteristic peaks from some of the impurities of interest and paracetamol. Distinguishing between compounds in a mixture can be very difficult if the species involved have largely overlapping peaks. This is particularly true for impurities which are fragments of the target compound or chemically-related to the target compound (e.g by-products, dimers, etc.). In this case, without further changes to the methodology such as isotopic labelling, it is not possible to discriminate between the different compounds.

In the context of the wider study of combining complementary analysis techniques, ToF-SIMS was compared to other techniques typically used to characterise API crystals, as described in the paper published by Ottoboni et al.[106]. The conclusion found was that the technique was an excellent addition and complementary analytical tool. While optical, electron and atomic force microscopy can be used to show surface texture changes and Raman microscopy can be used to show some elemental surface composition, ToF-SIMS is uniquely qualified to not only chemically characterise samples in question but also relay detailed information about the surface distribution of their constituents and expose surface texture details[106]. It has to be noted, though, that without significant additional work such as experimental series to produce relative reference standards or suitable internal standards, ToF-SIMS cannot be used to quantitatively analyse crystal surfaces. Other mass spectrometry techniques and/or high-performance liquid chromatography would be more suited for such endeavours despite the clear advantages that ToF-SIMS can offer, such as surface chemical imaging and bulk distribution information. The solution to this seems to be the application of multiple techniques for comparison, each one adding another piece to the puzzle.

A number of avenues can be explored for future work based on the results acquired during this study. As has been shown, ToF-SIMS can be used to analyse both surface and bulk of pharmaceutical API crystals, elucidating distributions and concentration gradients of impurities within the products. Given that many impurities are often structurally related, further work could be performed by isotopically labelling the API or impurities of interest. Isotopic labelling could also enable crystal growth and interface studies of API isomers such as paracetamol and its isomers ortho- and metacetamol. Further work could be performed along the lines of the bulk studies performed in this chapter. It would be interesting to analyse all possible paracetamol impurities and understand whether surface to bulk variations can be found and elucidate how this observation came to be.

The crystals produced for this study do not necessarily represent standard crystals used in pharmaceutical manufacturing; they are larger, have enhanced and more emphasised impurity loadings and represent samples engineered for fundamental studies of crystal changes. These studied crystal behaviours though are representative for all crystal sizes and thus are relevant. ToF-SIMS has the capability to analyse samples with a detection limit in the ppb – ppm range and could be very illuminating on many typical pharmaceutical samples, such as characterisation of cooling crystallisation products, blended and multi-component dosage forms and many more. The ability to analyse both surfaces and bulk revealing both chemical composition and distribution of compounds of interest offers fantastic opportunities and confirms Barnes' statement from 2011[108], "ToF-SIMS adds great value to pharmaceutical material characterisation options and bids unique perspectives on sample surfaces".

4.5 Conclusion

This chapter presented the application of ToF-SIMS for pharmaceutical material characterisation in the context of analysing paracetamol crystals with various impurity deposits. The aim of the project was to assess whether ToF-SIMS could be a useful technique to detect and image the presence of paracetamol-related impurities at low concentrations in crystals formed via cooling crystallisation, and use it to compare the crystallisation product results to other impurity loading scenarios. Three 4-nitrophenol impurity loadings scenarios, a surface based droplet application (PDN), an impurity epitaxial growth (PEN) as well as the integration of the impurity via a cooling crystallisation process (P4%N) were compared using ToF-SIMS. The comparison shows that 4-nitrophenol could be clearly identified and localised in all impurity loading scenarios. Ion images of the three samples emphasised the surface differences with the PDN sample indicating the impurity covering most of the crystal surface, the PEN sample exhibiting epitaxially grown impurity crystals and the P4%N sample displaying a homogenous paracetamol as well as 4-nitrophenol distribution on the crystal surface. While moderately visible using the ToF-SIMS, the results facilitated by OM and SEM analyses revealed further significant differences in surface texture between pure paracetamol and the impurity-loaded sample crystals of PDN, PEN and P4%N. Further work included the bulk analysis of paracetamol crystals with 4-acetamidobenzoic acid as an impurity. Here, the results showed a very different impurity behaviour at the crystal surface where the impurity appeared to be more concentrated in comparison to the bulk itself (P4%A), which on the contrary appeared to be more similar to the P4%N crystal analysed. While the experiments did demonstrate some limitations of ToF-SIMS, such as difficulties analysing fragments that are similar to the original analyte and thus show overlapping mass spectral peaks, they also clearly showed the value of the technique. The ability to analyse and image the chemical and topographical variations of samples across surfaces as well as the capability to create 3D-images and depth profiles of sub-surface layers can be of great benefit to pharmaceutical material studies.

Chapter 5

Detection of PZ-128 in Human Coronary Artery Endothelial Cells

5.1 Introduction

Cardiovascular disease (CVD) is a generalised term for a number of ailments that affect blood vessels and the heart. According to the World Health Organization (WHO), CVD and its linked conditions are the number one cause of deaths globally accounting for approximately 40% of annual deaths in the European Union[117–119].

Considerable research efforts are undertaken to further the understanding, prevention and regression of CVD-linked illnesses, with significant progress being made over the last two decades with regards to research outputs and results[119–122].

The majority of CVD-related deaths can be attributed to acute events in the form of heart attacks and strokes caused by arterial hardening and obstruction (atherosclerosis), as well as the formation of blood clots blocking blood flow to the heart, brain and other areas of the body. Atherosclerosis is caused by the build-up of fat, cholesterol and other constituents found in the circulatory system that accumulate in arteries to form plaque. When plaque becomes unstable and ruptures or breaks off, blood can coagulate at the site of a rupture forming blood clots, a condition also known as thrombosis, that can lead to blockages[117,119,123]. The coagulation of blood is caused by a complex cascade of events that ultimately culminate in haemostasis, the stopping or reduction of bleeding. One vital protein involved in this process is called prothrombin, one of four protease-activated receptors (PAR) which, in addition to other functions, play a key role in the control and regulation of haemostasis and inflammatory response. This particular coagulation factor, also known as PAR-1, is highly expressed in platelets and endothelial cells[124].

As part of the on-going research into this important field, different approaches have been studied to treat patients that exhibit risk of thrombosis via the use of anti-platelet agents that reduce or stop blood coagulation[122,123,125]. A recent approach is the use of PAR-1 inhibitors, which inhibit the thrombin-mediated activation of platelets and thus halt blood coagulation and arterial thrombosis[125].

PZ-128 is a membrane-tethered, cell-penetrating peptide that is being tested for its PAR-1 blocking abilities and anti-platelet function. According to Gurbel et al. PZ-128 "modulates platelet function by inhibiting signalling at [receptor protein] interface[s]". In their paper, the authors claim "that PZ-128 selectively inhibits the protease-activated receptor-1 receptor in subjects with coronary artery disease or risk factors"[126]. The drug passed its first phase of clinical trials (trial ID NCT01806077) in 2016 and is currently awaiting further testing[126]. Although the capabilities of PZ-128 are promising, numerous acute adverse effects have been reported; from allergic reactions to transient low blood pressure[126,127].

The Cunningham group at the University of Strathclyde has been investigating PAR-1 inhibitors, in particular PZ-128, and its alleged target specificity. Findings have recently been published that investigated this off-target activity[128]. Despite the claimed target specificity, these studies show evidence of PZ-128 affecting cardiovascular cells, which may explain the adverse reactions observed in patient cohorts[126,127]. This means that, in addition to the drugs inhibitory function, further effects are triggered that can be detrimental to cells and the host.

In an attempt to support the understanding of the mechanism and extent of the off-target activity of PZ-128 on cells, further investigation has been conducted using ToF-SIMS. The structure of the PZ-128 compound is shown in Figure 5.1. It consists of a palmitate conjugated to a peptide of 7 amino acids (lysine, lysine, serine, arginine, alanine, leucine and phenylalanine).

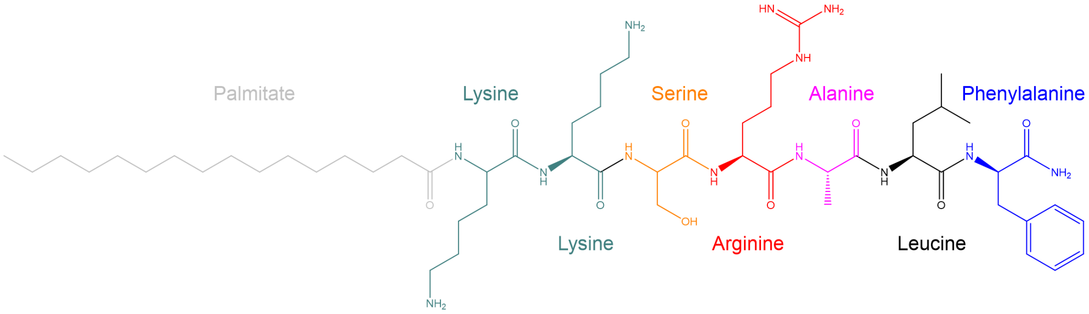


Figure 5.1: Structure of PZ-128 including amino acid and palmitate components, chemical formula $C_{55}H_{99}N_{13}O_{9}$, monoisotopic mass of 1085.77u.

The primary use of PZ-128 is in its function as an anti-platelet drug during stent procedures. Human coronary artery endothelial cells (HCAECs) have been chosen for this study as they would be directly exposed to the drug while travelling through the human circulatory system, thus making them an excellent target for this investigation. This chapter details efforts to complete a number of objectives regarding the analysis of PZ-128 application to HCAECs. The objectives of the study were:

- to optimise a cell preparation method for ToF-SIMS analysis.
- to identify drug-specific peaks and peaks of interest for the HCAECs.
- to identify PZ-128 within drug-treated cell spectra.
- to compare untreated and PZ-128 treated cells with regards to their peaks of interest in order to identify any significant changes.

Different cell preparation methods have been trialled to successfully complete the objectives set by the study and to enable the identification and execution of experiments to investigate the cell-drug interactions. Using literature and knowledge of the key biochemical processes in cells, a list of compounds and their respective mass peaks was generated. This list included alkali and alkali earth metals, such as sodium, potassium and calcium, amino acids such as histidine as well as lipids, such as phosphocholine and cholesterol. The list-included alkali metals are particularly important for cardiovascular functions. Any changes in cellular regulation of these ions could lead to significant adverse effects such as cardiac arrhythmias and other issues. The intensity and distribution of these compounds were compared between the untreated HCAECs and the PZ-128-treated cells with the only difference between the untreated and treated cells being the PZ-128 treatment of 30 µM exposure of the drug for 1 hour).

Considering the importance of this research, any significant changes in ion trends should be cause for concern for the impending phase 2 medical trials. Given the limited number of samples and high requirement of experimental repeats of the system to account for biological variability, additional studies with a larger number of repeats and samples should be carried out to confirm the results from this study. These are discussed towards the end of this chapter. It is imperative that PZ-128 and its off-target effects are studied in further detail prior to further use within human patients.

5.1.1 Biological Cells

Cells have been a great topic of interest in the ToF-SIMS community since Chandra et al. presented the possibility of detecting localised diffusible elements within cells in 1986, with follow-up papers proving the effectiveness and feasibility of different cell preparation methods[129–131].

The following decade yielded a number of interesting studies such as a publication by Colliver et al. that exhibited the first-in-kind atomic and molecular imaging of single cells using ToF-SIMS[132], or Pacholski et al. presenting a method to successfully image and detect phospholipids in cells[133].

Today, ToF-SIMS and its related techniques are used for a wide range of cellular mass spectrometry imaging (MSI) analyses, such as the investigation of intracellular uptake of drugs[134], the study of lipid changes in cells after application of nanoparticles[135], or the examination of drug-induced effects on cells[136]. The biological applications for ToF-SIMS analyses of cells are diverse, and always increasing and improving.

As emphasized in many publications, great care must be taken regarding the cell preparation prior to ToF-SIMS analysis[137–140].

Among the various methods such as freeze-fracturing or frozen-hydrated cell measurements, an assessment of the methods described suggested the two viable options available to our laboratory were chemical fixation with alcohol drying and cryofixation with freeze-drying[138].

Chemical fixation entails cell samples being exposed to chemical components to chemically preserve and fixate them in place using the compounds such as glutaraldehyde or paraformalin/formaldehyde. The advantages in this approach are the possibility of working at room temperatures throughout the analysis as well as the preserving of chemical compartments within the cell[138,140]. However, chemical fixation can lead to a loss in signal of the cellular membrane and thus lead to a change in distribution of diffusible ions on the cell surface[138,140].

Cryofixation involves the flash-freezing of cell samples using isopentane or propane cooled by liquid nitrogen. The advantages of the flash-freezing process are a minimum amount of damage due to water crystallisation leading to cellular structures maintaining their integrity[141].

Subsequently, samples are freeze-dried by slowly increasing the temperature under vacuum extracting most of the residual water[56,140]. As the water is being extracted from within the cell, care has to be taken to ensure cell rupturing is kept under control via a slow increase in temperature during the drying process. Freeze-drying can, if done too quickly, lead to a rearrangement of molecules within the cell[138,140].

In a review by Malm et al. these two methods have been compared. The conclusion suggests that cryofixation followed by freeze-drying is a good ”general purpose method for preparing well-preserved cell samples” while chemical fixation may be of use for small cell membrane feature extraction[138].

Both methods have been applied and presented in this chapter.

5.2 Experimental Methods

5.2.1 Cell Sample Preparation

Single-sided polished silicon wafers (14x11 mm, 500 µm thick, intrinsic, undoped, Mi-Net Technology Ltd) were sterilised using 70% ethanol, air dried in a laminar flow hood and stored in a sterile 12-well plate (ThermoFisher Scientific Ltd., United Kingdom). Human coronary artery endothelial cells (HCAECs) were obtained from the European Collection of Authenticated Cell Lines, purchased through Sigma-Aldrich (Poole, United Kingdom). The cells were preserved in MesoEndo cell growth medium (Cell Applications, San Diego, USA) under conditions suitable for light-sensitive compounds to hinder any possible degradation through this route. They were then washed and passaged using TrypLe Express reagent (Gibco, ThermoFisher Scientific Ltd., United Kingdom) before collection. Subsequently, the cells were centrifuged and transferred into a culture flask where they were incubated at 37°C at a humidified atmosphere with 5% CO_2[128].

For final usage, the HCAECs were seeded within the 12-well plate at a cell density of $1x10^4$ cells/mL.

There, the ’treated’ cell samples were exposed to 30 µM PZ-128 for 1 hour at 37°C with both untreated and treated cells experiencing final washing steps prior to fixation.

5.2.2 Chemical Fixation and Drying

Cell samples were washed with phosphate-buffered saline (PBS; 0.02 M NaH_2PO_4, 0.02 M Na_2HPO_4, 0.15 M NaCl, 5.4 mM KCl, pH 7.2, reverse-osmosis (RO) water). Cells were fixed with 2.5% glutaraldehyde (GA) (Sigma-Aldrich, United Kingdom) in PBS (25% glutaraldehyde in water, diluted 1:10 in PBS) for 15 min in 37°C. Any remaining fixative solution was washed off using more PBS. Post-fixation of the cells was achieved using 1% OsO_4 in RO water followed by repeated RO water rinsing steps. This procedure represents a secondary fixation step for the sample and is typically used to fixate lipids. The samples were then dried by gradually increasing the concentration of ethanol in the rinsing solution (40%, 50%, 60%, 70%, 90%, 96%, and 100%) and thus replacing the water content. Samples were then removed from the solution and air-dried in a laminar flow hood for 20 minutes. Post-drying, the samples were stored in a 12-well plate at room temperature.

5.2.3 Cryofixation and Freeze-Drying

Cell samples seeded onto silicon wafers were cryofixed by dipping the samples into liquid isopentane (kept at -196°C by liquid nitrogen) for 5 seconds. After fixation the samples were then freeze dried at 110°C for 3 hours and left at room temperature prior to analysis.

5.2.4 Scanning Electron Microscopy

Scanning electron microscopy (SEM) was utilised in order to confirm the integrity and usability of the cell samples after the sample preparation process.

The cell samples on the silicon wafer pieces were secured onto SEM aluminium stubs using carbon tape and then coated with gold to a thickness of 10 nm. Gold nanoparticles were sputtered onto the sample using an EM ACE 200 sputter coater (Leica Microsystems, Wetzlar, Germany).

The gold-coated samples were then placed into the Keysight 8500B SEM (Keysight Electronics, Santa Rosa, California, USA) and the focus of the SEM was adjusted to obtain the best image quality of the cells on the silicon substrate.

A range of cells were then imaged at two magnifications (2500x and 10506x) using a topology-focussed mode of the SEM. The topology mode is used to better image surface topology and roughness in the final images.

These images were taken at a resolution of 2048 px x 2048 px and each final image was the result of averaging two individual images taken from the same location on the sample.

Due to issues with the SEM instrument, heavy line-scanning artefacts were present in the images. The open source software Gwyddion (version 2.50) was used to remove some of the imaging artefacts and produce clearer images.

5.2.5 ToF-SIMS

All analyses were performed using an TOF.SIMS 5 instrument (IONTOF GmBH, Muenster, Germany). The instrument was mostly operated using a 30 keV Bi_3^+ primary ion beam. Different parameters and modes of operation were applied for different samples as laid out in Table 5.1 in order to determine the optimal parameters for analysis. Based on the results from the initial samples, a field-of-view of 500 µm^2 with a typical raster size of 512 px x 512 px was applied to subsequent samples by default.

Table 5.1: ToF-SIMS measurement conditions for all sample set used. Mode refers to measurement mode of the TOF.SIMS 5 instrument with 'Spec' signifying spectrometry mode and 'DE' signifying delayed extraction (See Chapter 2). 'Analysis relevant' indicates whether the measurement was used in the final analysis.

Samples	Set	Condition	Repeats per Set	Mode	Polarity	Raster Size (px)	Field of View (µm)	Ion Source	Analysis Relevant?
Set 1	A	Untreated	6	Spec.	+	128x128	500x500	Bi_3^+	Yes
		Treated							Yes
Set 2	A	Untreated	3	DE	+	512x512 binned to match Set 1	500x500	Bi_3^{2+}	No
		Treated							No
Set 3	A/B	Untreated	6	Spec.	+/-	512x512 binned to match Set 1	500x500	Bi_3^+	Yes
		Treated							Yes
Set 4	A/B	Untreated	6	Spec.	+/-	512x512 binned to match Set 1	500x500	Bi_3^+	Yes
		Treated							Yes

5.3 Method Optimisation and Results

5.3.1 Analysis of PZ-128

A small amount of pure PZ-128 in crystalline form was deposited onto a piece of double-sided tape that had been attached to an aluminium block. Excess sample was removed from the tape via tapping and a gentle stream of nitrogen. The sample was then introduced into the instrument and, using the previously described spectrometry mode of the TOF.SIMS 5, positive and negative spectra of the compound were acquired.

Figure 5.2 shows representative ToF-SIMS spectra of PZ-128 in positive and negative ion modes. As can be seen, while both positive and negative ion acquisitions resulted in presentable spectra that show larger fragments, the positive spectra exhibited better resolved and more intense molecular ion peaks of the compound. Easy identification and selection of the molecular ion makes positive mode acquisition the preferred choice. Some of the larger fragment peaks of the molecular ion in positive ion mode have been proposed in Table 5.2.

Table 5.2: Table of proposed larger fragments observed from the positive PZ-128 spectrum.

Fragment	Mass (m/z)	Fragment Formula
F1	505.32	$C_{24}H_{40}N_8O_4$+H^+
F2	561.34	$C_{27}H_{44}N_8O_5$+H^+
F3	592.35	$C_{27}H_{45}N_9O_6$+H^+
F4	646.43	$C_{32}H_{55}N_9O_5$+H^+
F5	720.44	$C_{33}H_{57}N_{11}O_7$+H^+
PZ-128+H	1086.78	$C_{55}H_{99}N_{13}O_9$+H^+
PZ-128+Na	1108.76	$C_{55}H_{99}N_{13}O_9$+Na^+
PZ-128+K	1124.73	$C_{55}H_{99}N_{13}O_9$+K^+

Figure 5.3 depicts the molecular ion peaks observed in the positive spectrum. The protonated ion $[M+H]^+$, observed at *m/z* 1086.78, is the most intense molecular ion peak, with the sodium and potassium-adduct ions, at *m/z* 1108.76 and *m/z* 1124.73, falling off in intensity.

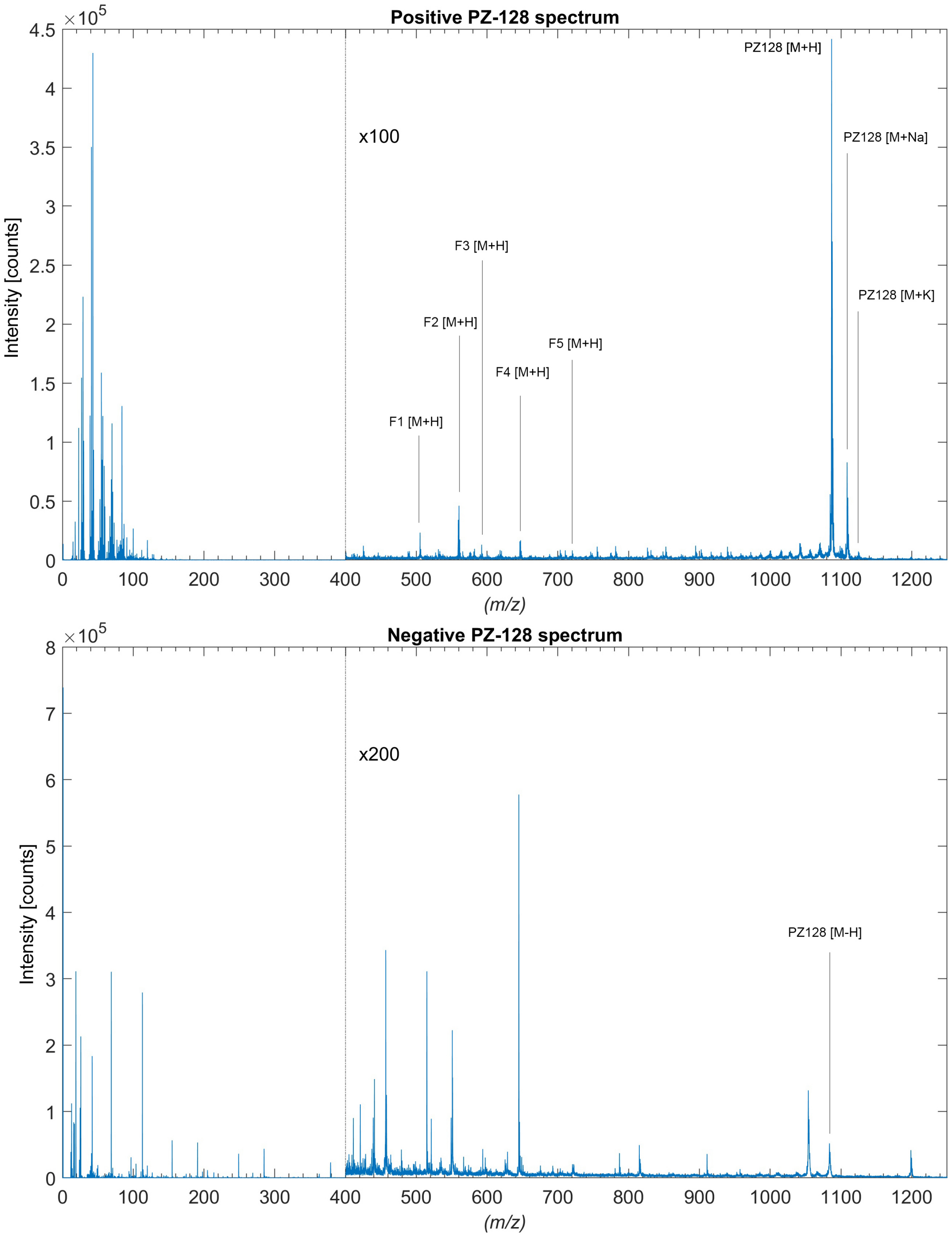


Figure 5.2: Positive (top) and negative (bottom) ion spectra for the pure PZ-128 compound. Magnification of the higher m/z regions has been included to allow lower intensity peaks to be observed.

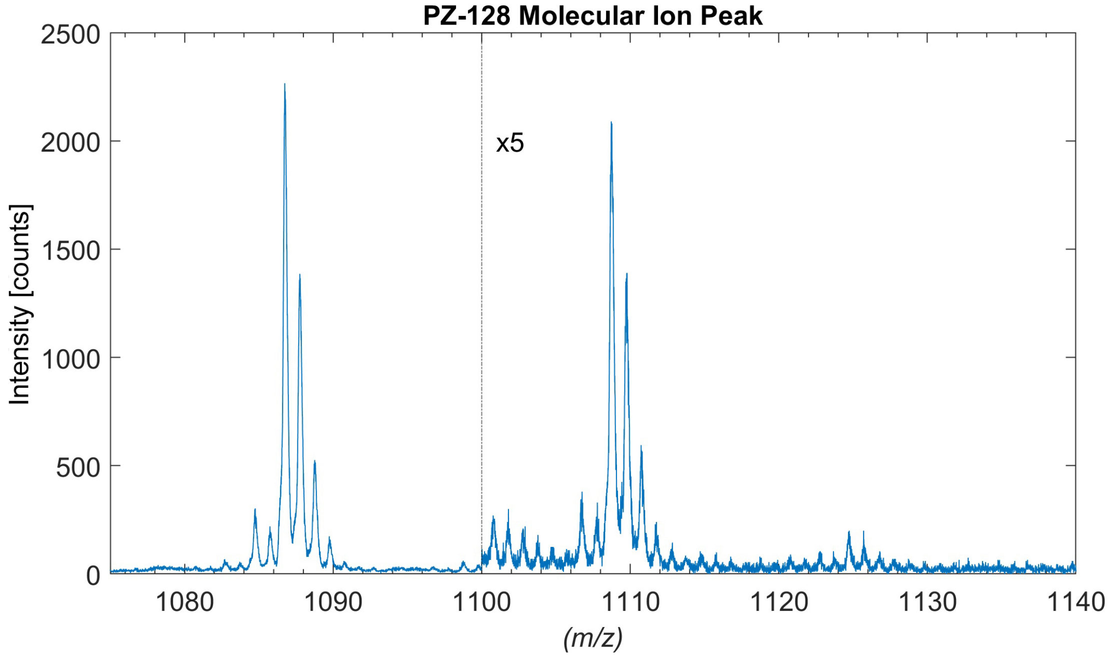


Figure 5.3: Representative positive ion spectrum of pure PZ-128 compound, from left to right, $[M+H]^+$ at m/z 1086.78, $[M+Na]^+$ at m/z 1108.76 and $[M+K]^+$ at m/z 1124.73.

These results confirm the suitability of ToF-SIMS for analysing PZ-128 in its pure form and illustrates how the compound ionises and fragments under SIMS conditions. The characteristic peaks identified here have been used to identify and locate PZ-128 in subsequent HCAEC samples.

5.3.2 Chemical Fixation and Drying versus Cryofixation and Freeze-Drying

Two cell sample preparation methods, namely chemical fixation and alcohol drying (**CD**) versus cryofixation and freeze-drying (**FD**) have been compared, and their feasibilities assessed for future experiments. As discussed in Chapter 2, sample preparation can significantly influence the chemical information obtained from SIMS and therefore the most suitable technique must be determined for the successful continuation of the study and to establish an HCAEC preparation method for analysis using the ToF-SIMS instrument.

To be able to assess the quality of the acquired spectra and understand what kind of results to expect, a literature search of published known and assigned peak fragments for positive spectra of cell samples was performed. Table 5.3 shows a selection of peaks identified during the literature review. The full table, including the assigned peaks for spectra acquired in negative ion mode, can be found in the appendix C.

Preliminary analysis was carried out on two replicate sets of samples which were prepared using **CD**. These samples showed several cell-associated peaks below *m/z* 150, such as $C_3H_8N^+$ at *m/z* 58.07 or $C_4H_8N^+$ at *m/z* 70.07, but none in the higher *m/z* region as can be seen in Figure 5.4. As SIMS is a hard ionisation technique, it is normal to observe an abundance of smaller fragments and elements but this extreme reduction in signal suggests some unwanted effects from the sample preparation method.

Comparing the assigned peaks from Table 5.3 to the spectrum shown in Figure 5.4, typical amino acid related fragment peaks and smaller lipid related fragment peaks below *m/z* 160, associated to the plasma membrane of cells, could be detected and assigned.

Upon further examination of the literature, this was an underwhelming and unacceptable result, as peaks related to the cell biology should be observed beyond *m/z* 160[138]. For example, literature suggests identifying cells via the typically present phosphatidylcholine headgroup (PCH) fragment at *m/z* 184 (corresponding to $C_5H_{15}NO_4P^+$), which is abundant in the plasma membranes of cells and should be detectable by ToF-SIMS[134,138]. No such fragment could be distinguished at large enough intensities to

Table 5.3: Proposed peak assignments for positive ions and fragments in the cell spectra.

Ion	Mass (m/z)	Name
$NH_4{}^+$	18.04	Ammonium
Na^+	23.00	Sodium
$CH4N^+$	30.04	Glycine Fragment
K^+	38.97	Potassium
Ca^+	39.97	Calcium
$^{41}K^+$	40.96	Potassium 41
$C_2H_4N^+$	42.04	Alanine Fragment
$C_2H_6N^+$	44.05	Alanine Fragment
$C_3H_4N^+$	54.04	Valine-, Leucine Fragment
$C_3H_6N^+$	56.05	Valine-, Leucine and Iso-Leucine Fragment
$C_3H_8N^+$	58.07	PCH-, Glutamic Acid Fragment
$C_2H_6NO^+$	60.05	l-Serine Fragment
$C_4H_6N^+$	68.05	Proline Fragment
$C_4H_8N^+$	70.07	Proline Fragment
$C_5H_6N^+$	80.06	Leucine-, Iso-Leucine Fragment
$C_5H_5O^+$	81.02	DNA Ribose Sugar
$C_5H_8N^+$	82.07	Histidine Fragment
$C_4H_6NO^+$	84.04	Glutamic Acid Fragment
$C_5H_{10}N^+$	84.08	Lysine Fragment
$C_5H_8N3^+$	110.08	Arginine-, Histidine Fragment
$C_8H_{10}N^+$	120.08	Phenylalanine fragment
$C_2H_6O4P^+$	125.00	
$C_5H_{13}NO3P^+$	166.06	PCH Fragment
$C_5H_{15}NO4P^+$	184.09	PCH Fragment
$C_8H_{19}NO4P^+$	224.11	PCH Fragment
$C_{27}H_{45}{}^+$	369.35	Cholesterol Fragment
$C_{27}H_{45}O^+$	385.34	Cholesterol Fragment

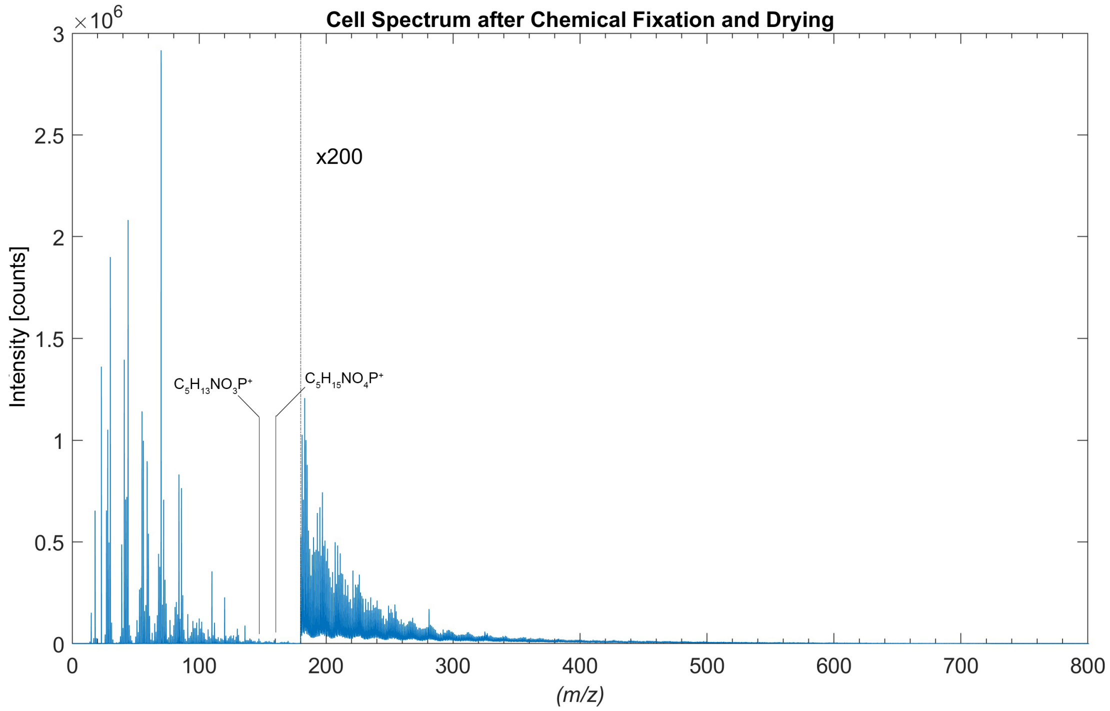


Figure 5.4: CD spectrum, Bi_3^+, 512x512 px, at 500x500 µm^2, total dose density 7e+11 ions/cm^2, 600 s, initially large intensities below m/z 100 with significant intensity drop-off after.

be associated with the cells and thus an optimized **CD** method was required.

To rule out prior experimental glitches a second run of **CD** cells was prepared and compared to a sample set of **FD** cells. In addition, pieces of clean silicon wafer were treated to the same experimental conditions for chemical fixation and cryofixation and analysed by ToF-SIMS in order to establish a list of background ions from the substrate.

Figure 5.5 illustrates the spectral differences between the sample preparation methods. First and foremost, there is a significant difference in ion intensities that can be seen between the samples; the **CD** spectrum shows approximately double the maximum intensity versus the **FD** spectrum. As before the **CD** samples peak intensities drop off above *m/z* 160 with no major biological peaks appearing beyond this point. All visible larger peaks are also present in the baseline spectrum of a **CD** method treated silicon wafer and can therefore be discounted. Looking at the **FD** samples it should be noted that a relative drop-off in intensity can also be seen but larger peaks are still detected past the previously perceived "end of detection".

Further differences can be observed in the lower *m/z* regions. While the **CD** samples exhibit comparatively small salt cation peaks for Na^+ (*m/z* 23) and K^+ (*m/z* 40) and show a generally higher signal intensity for smaller fragments below *m/z* 150 the opposite behaviour can be observed for **FD** samples. Here the salt cation peaks dominate the spectrum and the small fragments are not as emphasized.

It is likely that for the **CD** samples the majority of the salts have been washed off the sample surface and cells and thus only small amounts can be detected. The chemical fixation and alcohol drying procedure also seem to promote more fragmentation, decreasing the likelihood of detection of larger fragments. On the other hand, the **FD** sample surfaces likely still contain larger amounts of surface attached salts. It is not known whether these mostly come from the preparation media or whether they are largely present from the cell environment. It does appear though, that larger fragments are generated from this method, suggesting that the sample preparation method influences the stability or fragility of the molecules. Lastly, the peaks found in the FD spectra do not overlap with the baseline spectrum of an **FD** method treated silicon wafer and hence will be investigated further.

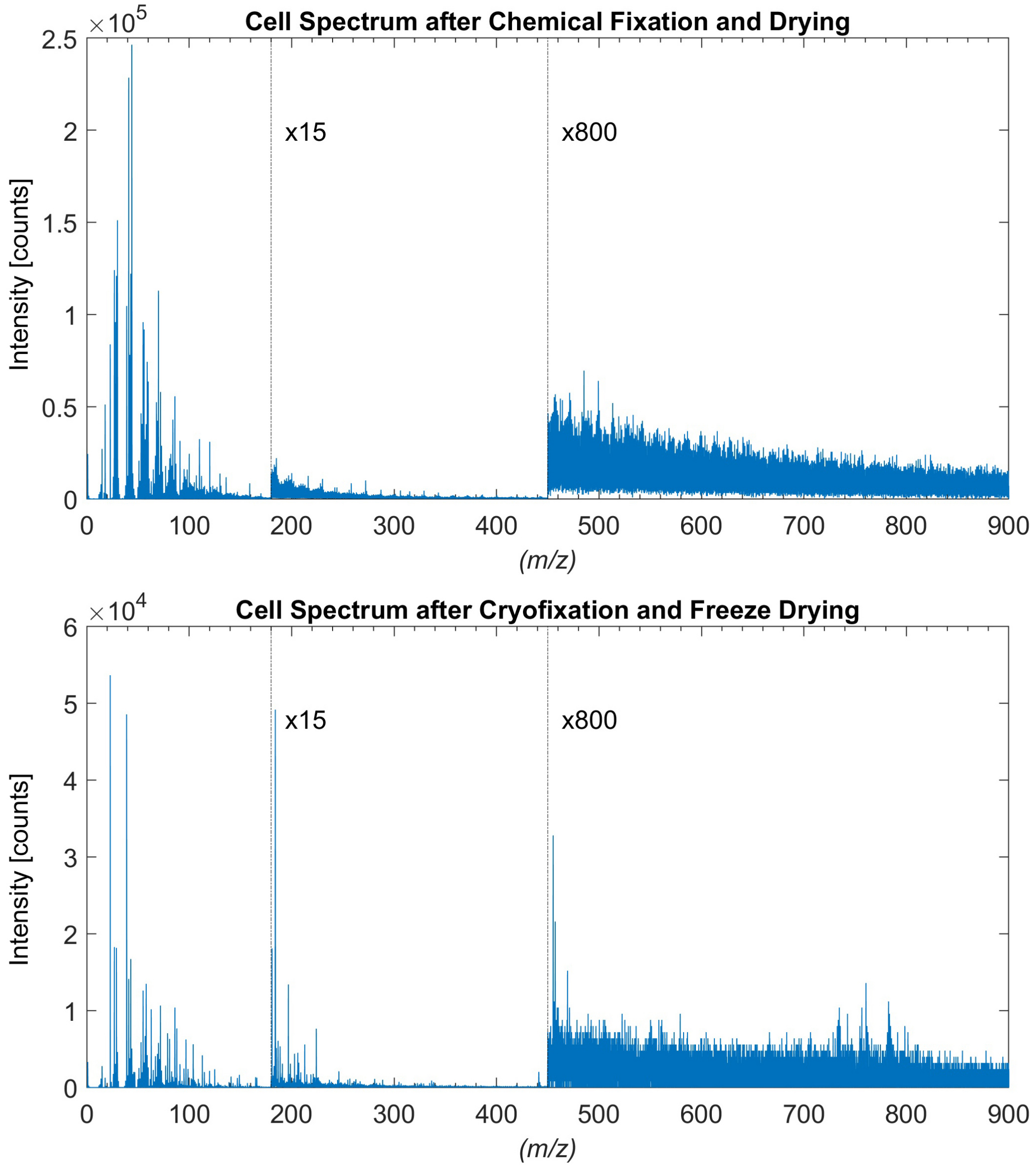


Figure 5.5: Comparison between representative positive sample spectra of the **CD** and **FD** methods over a mass range of m/z 0-900 with a 15x magnifying factor applied to all signals past m/z 180 and an 800x magnification applied to all peaks beyond m/z 450, both spectra were acquired using bunched mode with a $Bi_3{}^+$ ion source in positive mode over a 500x500 µm^2 field of view, a raster size of 256 px x 256 px and experienced the same dose density of 6.13e^{10} ions/cm^2.

5.3.3 Scanning Electron Microscopy Analysis of Freeze-Dried Cells

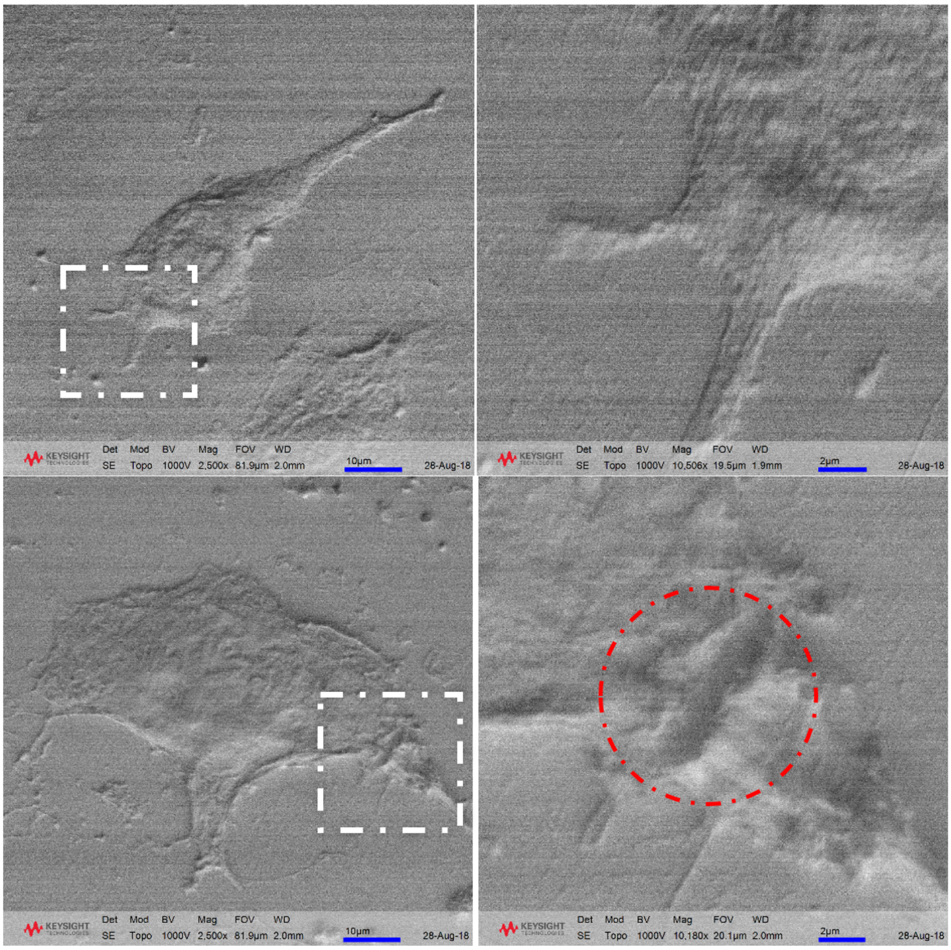


Figure 5.6: SEM images for untreated (top row) and drug-treated (bottom row) **FD** cell samples. Left images show entire cells at 81.9 µm^2, right images show zoomed in areas at 21.2 µm^2. The red circle indicates a potential cell rupture site on a treated cell.

SEM images of gold coated untreated and treated **FD** cell samples were acquired and compared (Figure 5.6).

The goal of the SEM analysis was to confirm the integrity and usability of the cell samples after the **FD** sample preparation process. 87.5% of the cells (7 out of a total of 8 analysed cells) imaged appeared to be intact with none of the untreated cells showing any damage while a single treated cell indicated potential signs of cell rupture (Figure 5.6). The damage could have occurred from several factors, including prior stress to the cell, the application of the drug or simply too much strain from the sample preparation method. The low number of damaged cells found suggests the validity of the chosen preparation method.

5.3.4 Untreated versus Treated Samples

To assess whether the used **FD** method is applicable to observe signals of the drug on the cells and to compare untreated versus PZ-128 treated cells, a preliminary comparative set of **FD** samples was prepared. The cell samples were treated and exposed to the same chemical environments with the only difference being a 30-minute exposure of the treated cells with the PZ-128 compound. The untreated cells were exposed to the same solvent at the same temperatures for an equal amount of time, but without the drug in order to minimise all possible variables.

Six repeat measurements were taken from each sample plate and the results were compared. Figure 5.7 illustrates the differences between untreated and PZ-128 treated cell samples after **FD** treatment. All spectra were acquired using bunched mode with a Bi_3^+ primary ion beam at 0.7 pA in positive mode over a 500x500 µm^2 field of view, a raster size of 256 px x 256 px and experienced the same dose density of 6 e+10 ions/cm^2

Firstly, comparing the spectra visually, it is apparent that the treated sample exhibits a much larger signal intensity. These changes could be the result of the drug altering cell surfaces and thus inducing changes to the surface chemistry of the cells, or of the drug molecule acting as an ionisation promoting matrix. Further sample repeats are required to answer this question.

Even after drug treatment, the Na$^+$ (*m/z* 23) and K$^+$ (*m/z* 40) salt ion peaks still dominated the spectra with most other similarly sized fragments displaying smaller intensity. This suggests the high salt loadings in the sample are inherent and are likely not removable without a change in sample preparation. As with the untreated samples, in addition to the smaller sized fragments below the *m/z* 160-mark, larger fragments are visible, with some reaching the *m/z* 700 region, a drastic improvement in comparison to the **CD** samples. More importantly, the molecular ion peaks of the PZ-128 drug ([M+H]$^+$ at *m/z* 1087, [M+Na]$^+$ at *m/z* 1109 and [M+K]$^+$ at *m/z* 1125) are clearly

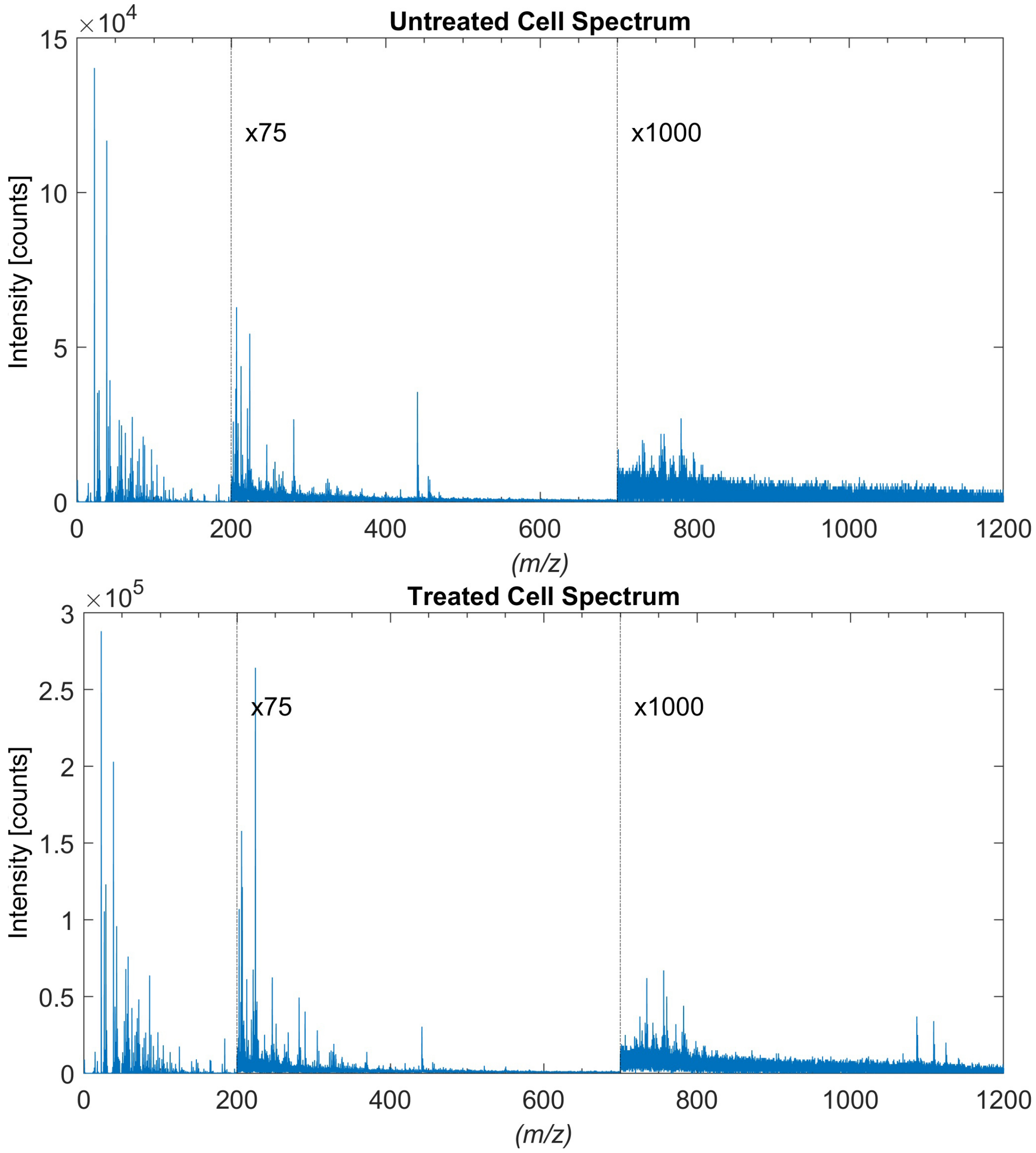


Figure 5.7: Shows representative spectra for untreated and PZ-128 treated **FD** cell samples. The mass range displayed is m/z 0-1300 with a 75x magnification factor applied to all peaks past m/z 200 and a 1000x magnification factor applied to all signals past m/z 700.

visible, suggesting the current method meets the requirements set by the objectives and can be utilized to monitor drug distribution on the cells. It must be noted though that in the higher *m/z* regions the signal-to-noise ratio appears to suffer and that the overall peak intensities are weak, suggesting that any reduction in the ion beam current would likely result in a reduction or even loss of the signals of interest. Figure 5.8 shows all peaks assigned after comparison with the literature-based Table 5.3. The assigned peaks contain salt ions, amino acid fragments and lipid fragments.

This signals further objective milestones completed for the study, namely the identification of a suitable sample preparation method, the assignment of a range of peaks of interest on the cell samples and the ability to identify PZ-128 drug related peaks in the treated cell spectra.

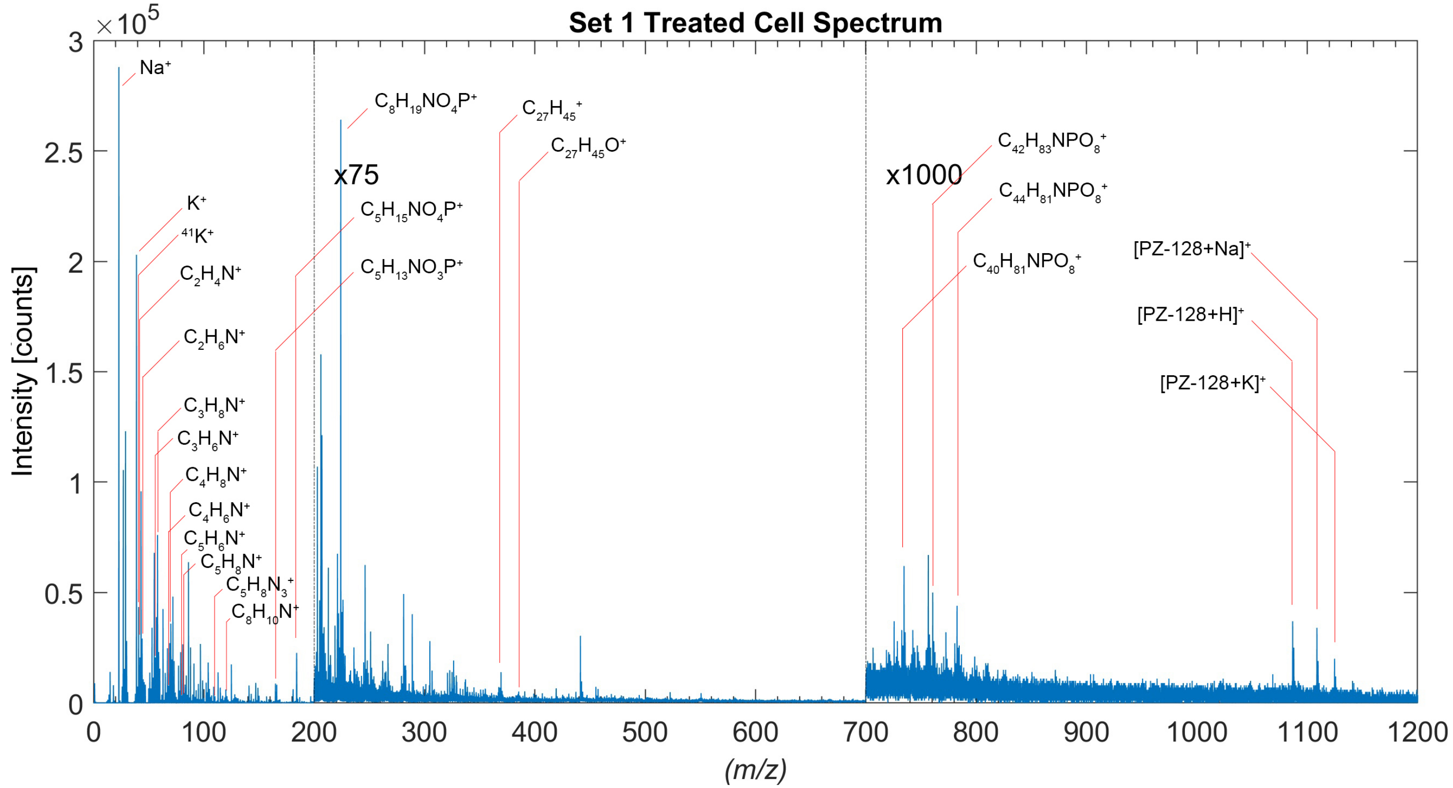


Figure 5.8: Annotated spectrum of a treated HCAEC prepared using the **FD** method.

5.3.5 Comparison of Spectrometry Mode Images of Untreated versus Treated Samples

Spectrometry mode images are not typically used for displaying ToF-SIMS imaging capabilities as they feature very poor lateral resolution (µm resolution versus imaging mode lateral resolution of up to 100 nm). For the purpose of comparison though, and to understand whether the chosen method is capable of visualising the drug being localised on the cells, spectrometry-mode based images are functional. To review, an ion image is produced by "scanning" a region of interest pixel by pixel and acquiring a mass spectrum for each pixel scanned. Peaks of interest can then be chosen and an intensity distribution of these peaks of interest can be reproduced into an ion image.

Figures 5.9 and 5.10 depict representative ion overlay images of untreated and treated **FD** cell samples. Here 3 ion images are overlaid to show the localised distribution of peaks of interest, namely; a phosphocholine headgroup fragment peak (PCH) representing the cell plasma membrane (red, *m/z* 184), PZ-128 molecular ion peaks (green, additions of *m/z* 1087, *m/z* 1088, *m/z* 1109, *m/z* 1110 and *m/z* 1125 and a background related peak, likely related to the silicon wafer to which the cells are fixed (blue, *m/z* 71).

Inspection of Figure 5.9 shows several cell outlines (red) surrounded by a blue background, representing the silicon wafer. A number of green dots of differing intensities seem to be spread throughout the overlay image, but no structured pattern of distribution can be discerned.

Figure 5.10 displays a similar image. Again, several cell outlines (red) are visible and appear to be surrounded by a blue background. In this overlay image though, there are numerous green intensity spots co-located with the PCH signal indicating localisation of the drug molecules on or in the cells.

As stated, the green colour in the presented figures is associated with signal intensities of the PZ-128 molecular ion peak additions. Hence, since no drug is present in the untreated sample, there should not be any PZ-128 peaks in the sample. The green dots observed in the untreated sample are proposed to be noise artefacts stemming from the peak selection and spectra comparison process.

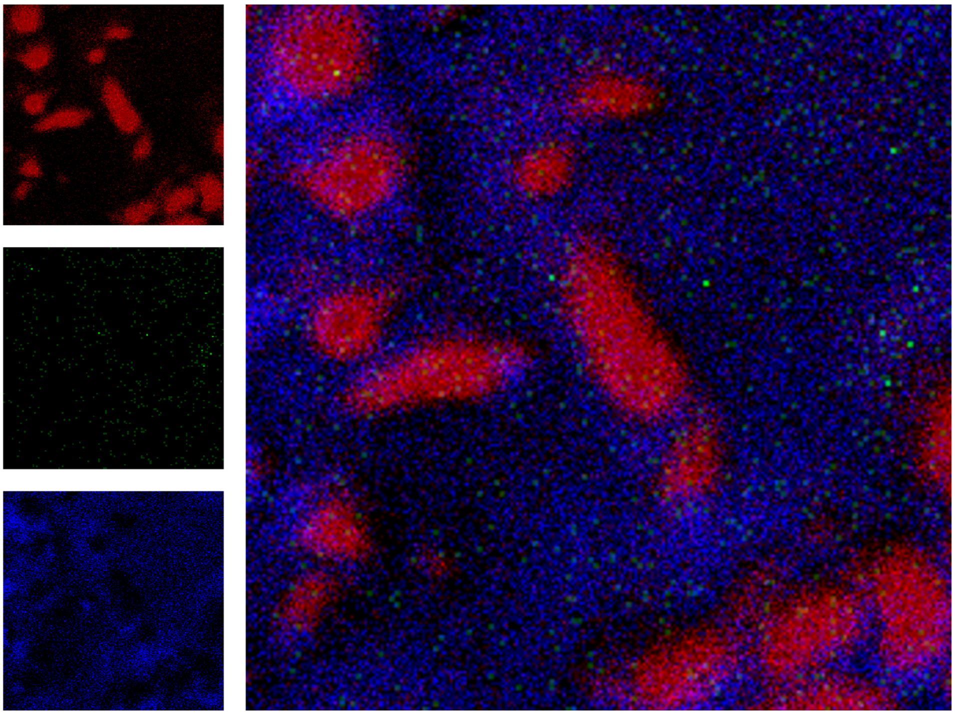


Figure 5.9: Ion RGB overlay image of a representative untreated **FD** cell sample; phosphocholine headgroup fragment peak (PCH) representing the cell plasma membrane (red), PZ-128 molecular ion peak additions (green), and a silicon wafer related peak (blue), selected for better contrast and visibility. The individual ion images are displayed on the left with the overlay being shown on the right. No minimum intensity was set for pixels to show colour resulting in noise peaks showing in the PZ-128 ion channel.

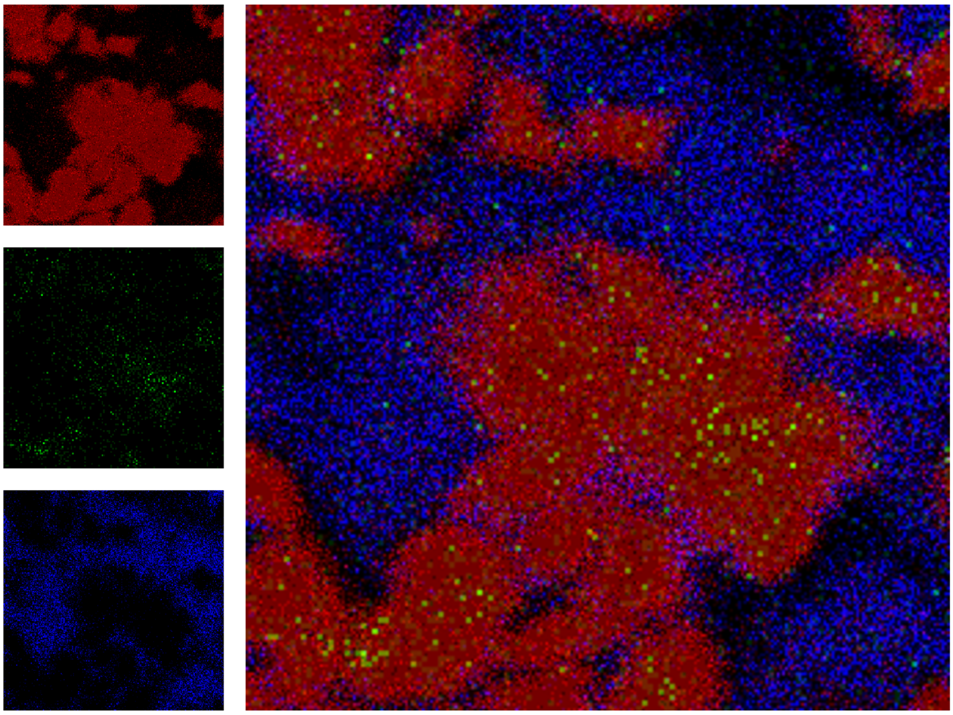


Figure 5.10: Ion RGB overlay image of a representative treated **FD** cell sample; phosphocholine headgroup fragment peak (PCH) representing the cell plasma membrane (red), PZ-128 molecular ion peak additions (green), and a silicon wafer related peak (blue), selected for better contrast and visibility. The individual ion images are displayed on the left with the overlay being shown on the right.

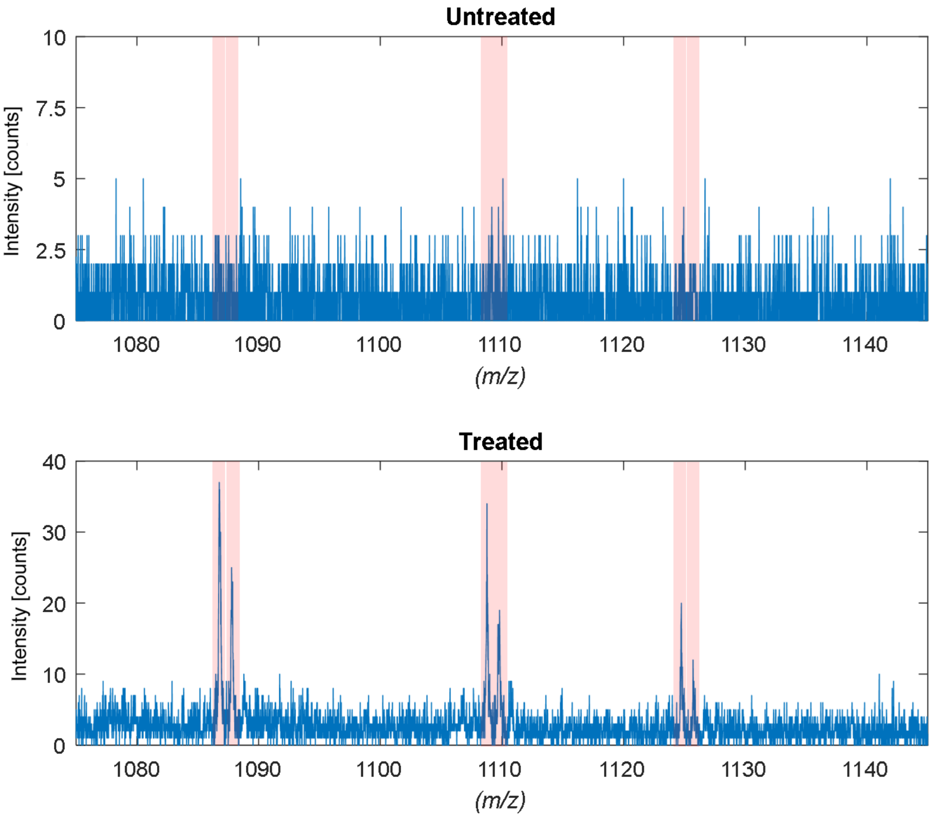


Figure 5.11: Spectra of the untreated and treated spectrometry mode images shown in Figures 5.9 and 5.10 in the region of m/z 1075 -1145.

A single peak list was created to compare all samples, so that the peaks of interest were selected in all spectra and the intensities of those could then be used for data extraction, the plotting of spectra and the visualisation of ion images. While this facilitates the analysis of multiple spectra, where no peak exists in the spectrum this method will still select the spectral background at the designated m/z, creating unwanted artefacts in the image. This is demonstrated by Figure 5.11 where it is clear that there are no PZ-128 related peaks present (m/z 1087, m/z 1088, m/z 1109, m/z 1110, m/z 1124 and m/z 1125) in the untreated sample, and that the green pixels shown in Figure 5.10 are background noise artefacts.

The spectra and images shown are representative of the 5 areas of a sample that were analysed, with similar results observed. All treated samples showed drug-associated peaks directly localised within the cell membrane areas, suggesting the drug treatment of the cells was successful, i.e. the drug interacted with the cell membranes and was successfully washed off from the substrate. It must be noted that this does not indicate how much of the drug traversed into the membranes and cells. However, this confirms that the cryofixation sample preparation method allows for meaningful spectrometry results and provides useful information in the form of lateral distribution of masses of interest.

5.3.6 Principal Component Analysis and Peak Comparisons

With the exception of semi-quantitative comparisons between untreated and treated cell samples using the **FD** method, every original objective of the study has been met so far. This study is still in the preliminary stage, as it is difficult to form theories about sample changes and elucidate trends from only one set of samples, but an initial assessment about PZ-128 treatment and cell analysis procedures can be attempted.

The sample preparation method for **FD** samples was maintained during the production of all samples. Untreated and treated cell samples underwent near identical treatment (except incubation with the drug molecule during the exposure step). Therefore, the assumption can be made that the sample preparation method does not cause the changes in the intensities between untreated and treated samples, but instead the drug treatment influences the cell biology and thus causes the observed changes in the samples.

Initially a principal component analysis (PCA) was performed for sample exploration purposes and to guide further analysis. Subsequently, peaks of interest were selected and the changes between the samples were tracked.

In this section, 4 different sample sets are compared; Set 1, Set 3B, Sets 4 A and B (see Table 5.1). Two further samples sets were excluded from PCA modelling; Set 2 has been acquired using delayed extraction and thus shows too large spectral differences while inclusion of Set 3A resulted in the breaking of the model which will be analysed at a later point. Within both removed sets, differences between untreated and treated samples can still be analysed.

For this purpose, a single peak list containing all positive peaks found between *m/z*

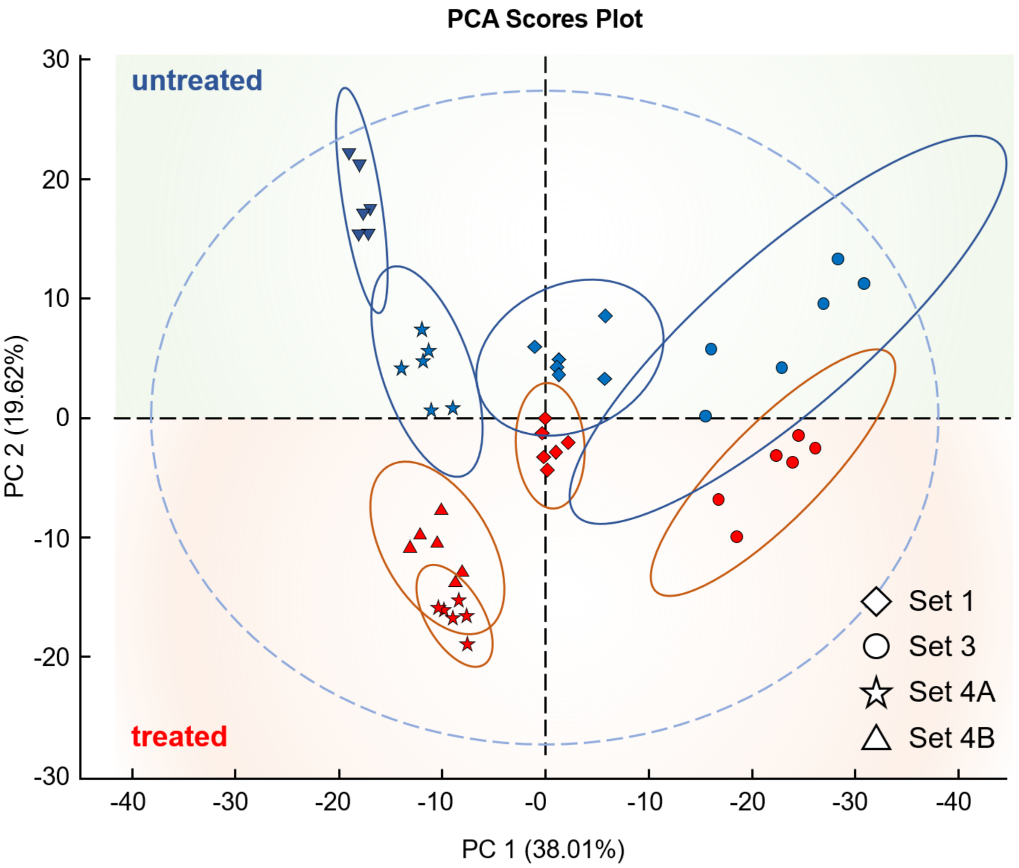


Figure 5.12: PCA scores for PC1 vs PC2 for sample set 1, 3B, 4 A and B. PC1 separates the different sample sets while PC2 separates untreated from treated samples, with treated samples being positively correlated to the PC and vice versa.

18 and m/z 900 have been selected and assigned where possible. A region of interest (ROI) for each sample was selected using the observed cell outlines demonstrated by the PCH-related peaks (m/z 104 and m/z 184) and the ROI spectra were extracted using the peak statistics function in the IONTOF SurfaceLab 7.0 software. Each spectrum was normalised by its specific total ion intensity. Spectra were loaded into MATLAB version 2014b and analysed using the EIGENVECTOR PLS_TOOLBOX, version 7.9.5.

The spectra were loaded into the principal component analysis tool and pre-processed by mean-centring and scaling each variable to their unit standard deviation followed by mean-centring the spectra. A single principal component analysis model was generated. Principal components (PC) 1 (38.01%) and 2 (19.62%) account for 57.63% of the variability in the model. The ovals surrounding each sample are the 95% confidence intervals.

As shown in Figure 5.12 PC1 can be attributed to the separation between the different sample sets, with Set 3 being positively correlated and Sets 4 showing negative correlation. Set 1 does not exhibit a large correlation to PC1 and thus sits between Sets 3 and 4. Small differences in the samples can lead to their separation in the PCA. Reasons for differences in the samples can arise from the sample sets having been prepared on different days, thus leading to slight differences the preparation. Furthermore, the ToF-SIMS analysis was performed as soon as the samples became available, again giving rise to different background conditions during the measurements.

PC2 shows good separation within each of the sample sets depending on the presence of the drug treatment. All untreated samples are positively correlated while all treated samples show a negative correlation to the PC. Set 1 seems to be less strongly correlated to the PC component than most other sets as well as a small number of samples from Set 3 and 4, which appear to have a particularly low correlation to the PC. Overall, the PCA seems to achieve full sample separation across multiple sets of samples suggesting there are difference after treatment.

To understand on which grounds the PCs separated the samples it is important to view the PCA loadings. The loadings describe which variables are associated to the different PC's and what kind of responses they have to the positive and negative scores. Figure 5.13 depicts all positive and negative peak loadings associated with PC2 (blue) and a subset of all known and assigned peaks and their respective loadings in orange.

Viewing the loadings in Figure 5.13 it is clear that no single variable is primarily responsible for the output of the PC. Instead the defining factor appears to be an amalgamation of all peak variables and their associated change between untreated and treated samples. As seen in the scores, a positive loading is related to untreated samples while a negative loading can be related to treated samples. The majority of assigned peaks (orange) are amino acid and lipid related peaks which appear to be negatively associated with PC2, meaning that the treated samples exhibit larger intensities for these peaks. Particularly noteworthy are the PCH associated peaks; the $C_2H_6PO_4^+$ at

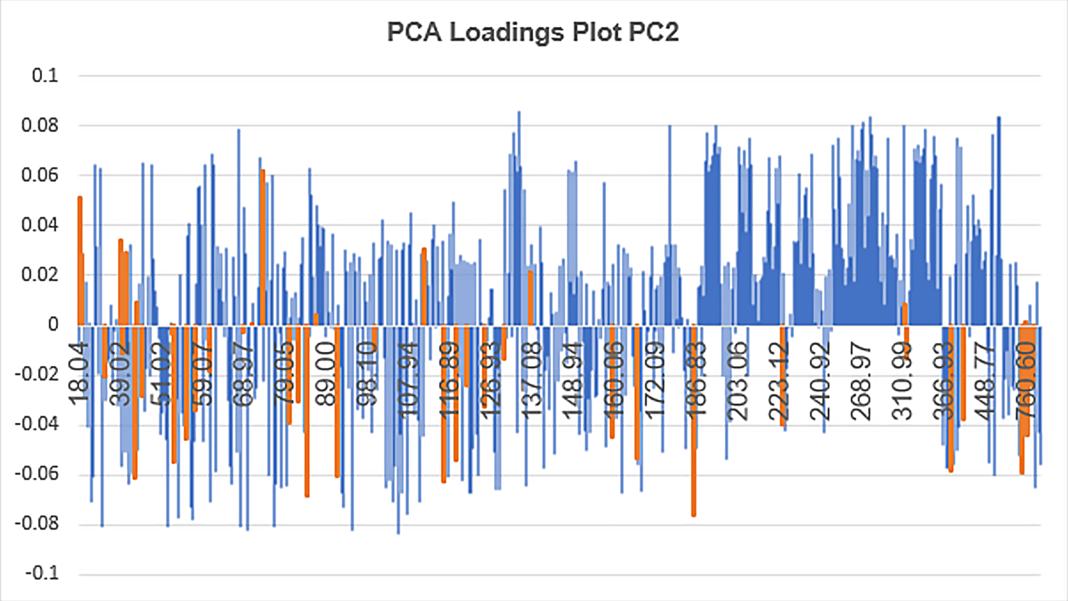


Figure 5.13: PC2 loadings for all 593 peak variables (blue) involved in the PCA model making. A second series is plotted representing all known positive peaks and their respective loadings. PC2 represents 19.62% of all variance within the model.

m/z 125, $C_5H_{15}NO_4P^+$ at m/z 184.11, the larger associated PCH headgroup peaks in the m/z 700+ region as well as the cholesterol associated peaks $C_{27}H_{45}^+$ and $C_{27}H_{45}O^+$ at m/z 369.37 and 385.35 respectively. Changes in these peaks suggest changes in the concentration of the plasma membrane of the cells.

The few assigned positively associated peaks appear to be NH_4^+ at m/z 18.03, $C_2H_6N^+$ at m/z 43.03, $C_2H_7N_3^+$ at m/z 73.06, $C_5H_{12}N^+$ at m/z 86.09, $C_5H_8N_3^+$ at m/z 110.07, $C_8H_{10}NO^+$ at m/z 136.07, $C_{19}H_{35}O_3^+$ at m/z 311.27, $C_{42}H_{81}NPO_8^+$ at m/z 758.57 as well as the salt cation peaks of Na^+, K^+ and $^{41}K^+$ at m/z 22.99, 38.97 and 40.96 respectively. The largest positively associated contributors in the region between m/z 190 – 340 remain unidentified. Here it is of particular interest that a higher intensity of the salt cation peaks appears to be correlated with the untreated peaks suggesting a decrease in intensity after drug treatment.

With the principal component analysis suggesting meaningful deviations and a traceability of those via cell membrane associated as well as the salt ion peaks, a natural continuation of the study is the inter-spectra comparison of known peaks of interest.

Firstly, peaks for the Na^+, K^+, $^{41}K^+$ and Ca^+ ions are compared because of their particular importance for cardiovascular function and changes in cellular regulation of these ions could lead to significant adverse effects such as cardiac arrhythmia.

To assess what changes can be found between untreated cells and cells after treatment, peaks related to the cell membrane and cell functions have been compared.

As seen in the loadings plot in Figure 5.13 the most positively correlated salt cation peaks, with the exception of Ca^+, appear to show a decrease after application of the drug, which can be seen across all sets of samples. Unlike Na^+ and K^+, due to larger intensity fluctuations and contradictory changes occurring in one sample set, no clear pattern can be attributed to the Ca^+ data. Including the two sample sets not part of the model, the results in Set4B would be the only one out of six sets showing an increase in the treated sample.

Focussing on the amino acid and lipid related peaks found in the spectrum, a general increase in intensity can be observed in all sample sets after the application of the drug. Peaks proposed to correspond to amino acids or fragments of amino acids are observed at *m/z* 42 (alanine), *m/z* 56 (valine, leucine and iso-leucine), *m/z* 58 (glutamic acid and PCH), *m/z* 80 (leucine and iso-leucine), *m/z* 84.04 (glutamic acid) and *m/z* 84.09 (lysine). The glutamic acid peak at *m/z* 84.04 shows the opposing trend in Set4B, with a decrease in intensity observed after drug application. As this unexpected trend only occurs in Set4 and only for the glutamic acid peak (*m/z* 84.04), this is thought to be an anomaly. All other amino acid related peaks are seen to have a clear upwards trend after drug treatment.

Similar changes can be observed in the PCH-related lipid peak at *m/z* 184 as well as the cholesterol associated peak at *m/z* 369. After application of the drug both peaks of interest exhibit increased intensity among all sample sets, with Set4A displaying outlier-like behaviour with an order-of-magnitude intensity increase over the untreated sample. Other peaks in the samples of Set4A are not observed to have a similar order-of-magnitude increase in intensity, therefore this observation is as yet unexplained.

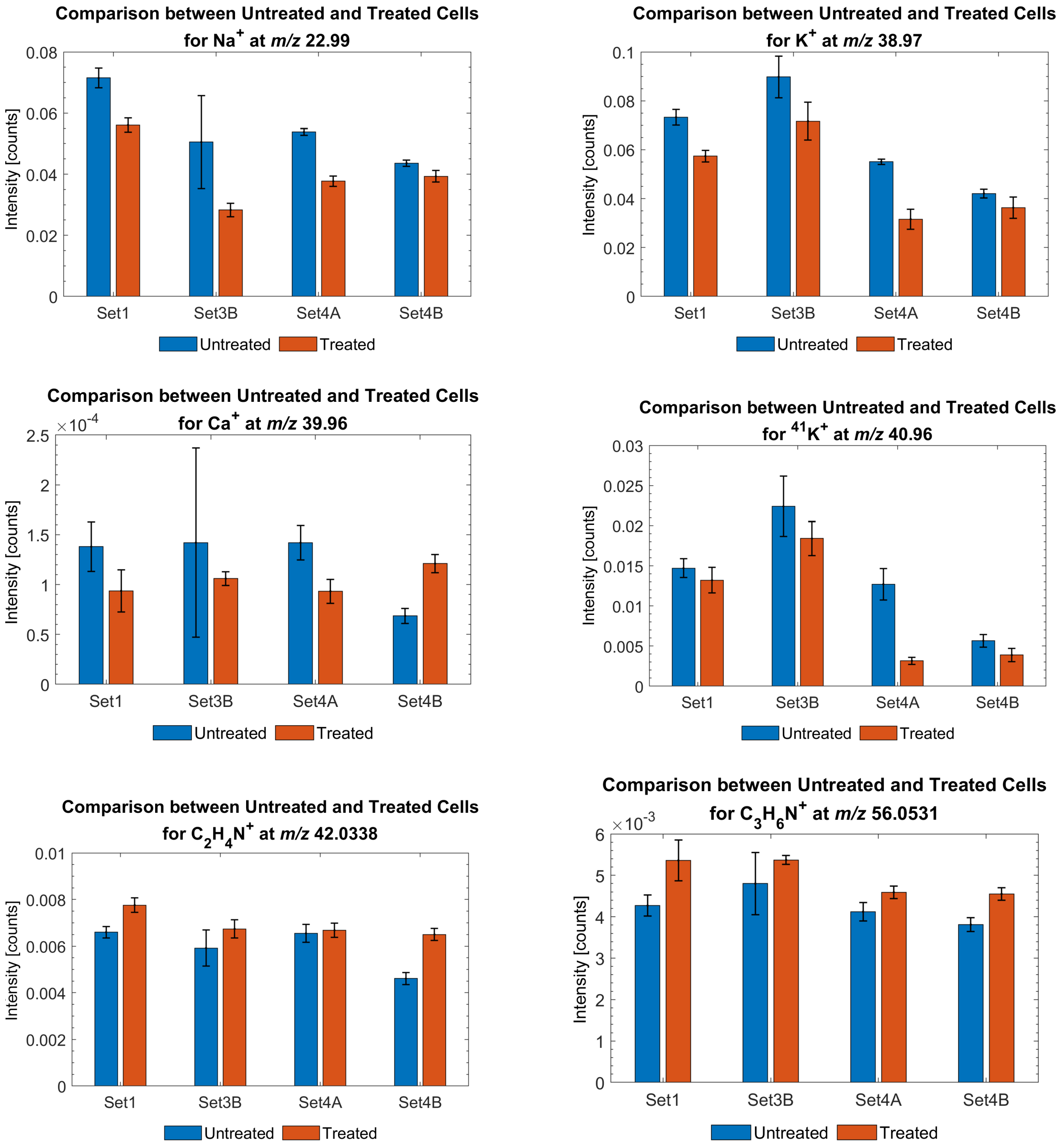
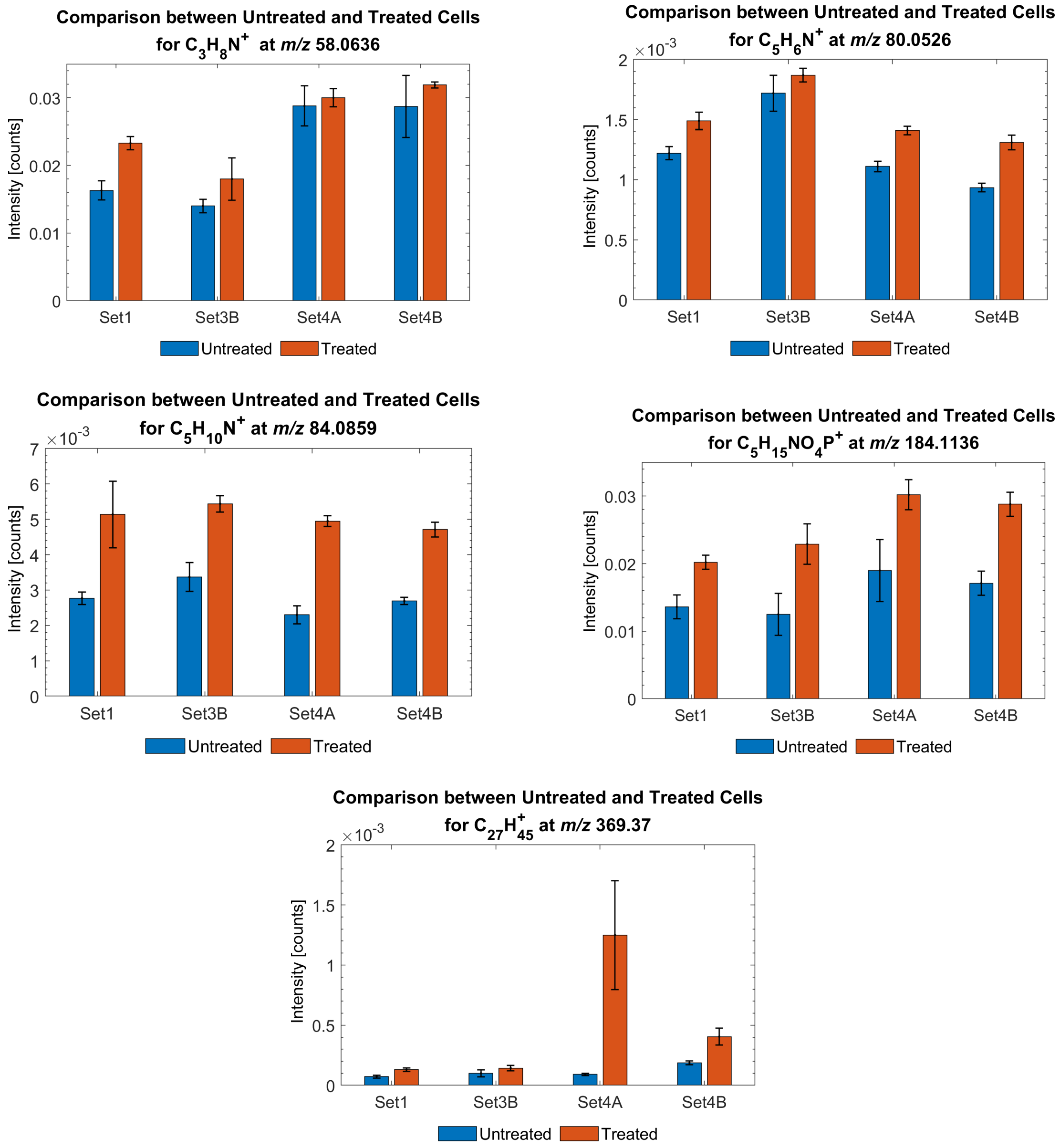


Figure 5.14: Barplot comparisons of ions of interest across the four modelled sample sets of untreated and treated cells based on positive ion mode measurements.


Comparison between Untreated and Treated Cells for $C_3H_8N^+$ at *m/z* 58.0636

Comparison between Untreated and Treated Cells for $C_5H_6N^+$ at *m/z* 80.0526

Comparison between Untreated and Treated Cells for $C_5H_{10}N^+$ at *m/z* 84.0859

Comparison between Untreated and Treated Cells for $C_5H_{15}NO_4P^+$ at *m/z* 184.1136

Comparison between Untreated and Treated Cells for $C_{27}H_{45}^+$ at *m/z* 369.37

Figure 5.15: Barplot comparisons of ions of interest across the four modelled sample sets of untreated and treated cells based on positive ion mode measurements.

5.4 Discussion and Future Work

In the previous section, changes in salt, amino acid and phospholipid peak intensities before and after treatment of HCAECs were compared. The sample range selected for this comparison was based on the same samples that were used in building an investigative PCA model to explore the data. The results from these peak comparisons, together with the PCA analysis suggest that there are changes between the cell samples before and after application of PZ-128. As for each set, both untreated and treated samples have been subject to the same experimental conditions, apart from exposure to the drug compound itself, the differences are therefore likely to be present due to changes induced by the PZ-128 compound. Even though, as previously stated, PZ-128 could act as a matrix, increasing the general secondary ion yield, the reduction in salt ion intensities and no general increase in all ions across the spectrum after treatment suggests otherwise.

Experiments performed in the Cunningham laboratory and partially published by Brouck and Cunningham under the title "Investigating the off-target effects of the clinical trial candidate PZ-128"[128] suggest the effects of PZ-128 do not appear to be PAR-1 specific, as originally stated by Gurbel et al., instead potentially activating a number of other signalling pathways[126,128]. As these experiments have largely been performed on other cell types (HEK293), they might not be directly indicative of the effects on HCAECs, though they strongly suggest possible off-target activity that could be responsible for some of the effects observed in this study. Neither study directly monitors or examines an increase in phospholipid concentration after drug treatment, therefore based on the results described herein, future work should investigate this effect and its causes.

Starting with the PCH-related lipid peaks and cholesterol observations, the increase in cholesterol response after treatment is a very important detail. Cholesterol is a cell component of the plasma membrane and as such is partly responsible for the regulation of endothelial cell functions and the mechanical properties of the cell. Changes and disruption of the cholesterol equilibrium can lead to malfunctions of endothelial cells which build the partition between blood flow and neighbouring tissue and can have wide-ranging effects[142]. All samples show an increased cholesterol response after treatment, indicating that the application of the drug is directly related to this increase.

Newman et al. studied the intracellular drug uptake of amiodarone, an antiarrhythmic drug, on four different cell lines using ToF-SIMS. In their work, one cell line expressed increased lipid concentrations after treatment with the tested drug[134]. Amiodarone and its class of compounds, cationic amphiphilic drugs, are known to cause phosholipidosis, a condition in which cells accumulate excess phospholipids internally, leading to inflammation and histopathological changes[143,144]. While PZ-128 does not belong to this class of drugs, nor does it contain any of its main functional groups, i.e. an aromatic ring or a halogenic compound, the response observed in the treated HCAECs appears to be very similar to the phospholipidosis-affected cell line reported by Newman; large increases in cholesterol and lipid levels after treatment with the drug. The cause of this outcome is currently unclear, but it is certain that further biological tests should be performed to understand whether the drug is directly responsible for this effect.

Focussing on the alkali metals, the reduction in electrolyte levels after treatment could point towards a disruption of ion gradients within the ion channels in the cells. Deviations in electrolyte concentration in cells can lead to arrhythmogenesis, the onset of arrythmia, particularly in the cases of lower potassium and calcium levels[145]. Due to sodium and potassium being the dominating peak features in the spectrum, it is difficult to quantitatively assess this phenomenon, as the disproportionate levels recorded in the spectra could possibly stem from the cell preparation process used. It is thus proposed for future studies to confirm and quantify these results in comparison with an alternative method, such as frozen hydrated cells and improved washing preparation. Nevertheless, the outcomes summarised here imply changes to the electrolyte levels in the cells that could lead to dangerous effects in patients and thus it is evident that determining the exact mechanism of the drug and its effects is vital.

There were a number of key challenges identified during this study. Initially, an appropriate sample preparation method had to be established for use with HCAECs that could be applied with the resources available and would offer suitable results through ToF-SIMS measurements. Based on a number of publications, chemical fixation using glutaraldehyde and cryofixation were attempted[134,138]. The results obtained using glutaraldehyde fixation and alcohol drying of the cells, however, were insufficient due to a lack of biologically relevant peaks beyond *m/z* 160 and the method was thus deemed unfit for purpose. Cryofixation and freeze-drying on the other hand did produce results consistent with observations reported in literature while also enabling the actual identification and signal intensity required and was thus chosen as the primary sample preparation method[26,138]. Other methods for sample preparation do exist, such as frozen hydration or freeze-fracturing, but were not trialled due to a limited supply of cells, time and resources. Sample preparation methods that enable the analysis of cells under ultra-high vacuum conditions cause a departure from their natural environment, yet the results that can be achieved with ToF-SIMS measurements directly depend on these methods[26]. Further work should be carried out to explore using a frozen hydrated method for cell preparation due to potentially improved secondary ion yields and reportedly better integrity of the samples[140].

Another challenge arising from the analysis of cells is the biological variability potentially resulting in inconsistent measurements and can only be adjusted for with increased experimental repeats. As reported by Newman et al. in their paper "Intracellular Drug Uptake—A Comparison of Single Cell Measurements Using ToF-SIMS Imaging and Quantification from Cell Populations with LC/MS/MS", variations in cellular drug uptake could be seen amongst cells of the same sample. The authors state one reason for such behaviour could be that the cells analysed were at different stages of their life cycles. Hence, to achieve statistically significant results, a higher number of measurement repeats and an increase in cells per sample should be aimed for.

Moving on to the ToF-SIMS measurements detailed here, different modes of analysis were trialled. Due to the low intensity of the PZ-128 peaks and higher-mass fragments such as cholesterol, low lateral resolution spectrometry mode measurements were chosen to ensure that peaks resulting from the drug and cell-relevant compounds would be observed in the same analysis. Using imaging mode, higher mass peaks could not be observed or if so only at greatly diminished intensities. In addition, a small number of untreated cells were analysed using delayed extraction and sputter analysis to assess the viability of the cells and potentially enable future research using sputter analysis. The resulting images show (see Appendix 3) that the cell core is visible and that the cells appear to be intact, as has already been assessed using SEM analysis. Delayed extraction did offer significantly better image quality however, the reduction in current also resulted in some higher-mass peaks, such as cholesterol, to be obscured by the background noise of the spectrum. Thus, for the reasons mentioned and due to the significant amount of time required for sputter measurements, further delayed extraction and dynamic SIMS measurements were not pursued.

As has been discussed, the analysis of biological samples via ToF-SIMS brings a number of challenges. These can also be expressed as sources of variability and error such as:

- Variability of cell life cycle stage: this may lead to different readings depending on the cells that are measured and may also lead to variability in the drug uptake per cell;
- Variability in preparation method: this could lead to changes to the cells and can have various downstream effects, such as different drug uptake rates, different survivability as well as altered sensitivity to ionisation, depending on whether preparatory changes occurred in the last stages of fixation and drying;
- Variability in ToF-SIMS conditions: including, but not limited to, different pressure within the chamber, differences in mounting of the samples as well as altered instrument conditions during measurements possibly leading to variations in secondary ion yield, measured peak intensities and image quality;
- Variability in cell numbers per image leading to potential variances in peak intensity differences reported.

Though greatest care has been taken in the preparation, measurement and analysis of the cell samples, variability in the preparation and measurement stages cannot be completely excluded. Such small variations can lead to deviations in the analysis results and could be controlled and tested for with increased biological and technical replicates. To reduce some of the variability from the analysis, all samples were normalised to the total ion intensity of the cell regions of interest while the PCA model spectra were pre-processed in the same fashion.

Further work in this area should include additional steps to develop and widen the scope of this study and enhance the understanding of the outcomes reported.

As previously discussed, the number of analysed samples and repeats was limited due to material and time constraints. Future work should include a larger number of cell samples per set and more sample sets to increase the statistical validity of the study.

In addition, an expanded method development stage could be performed together with the application of other mass spectrometry techniques, such as tandem MS, to identify peaks in the spectra resulting from the HCAEC samples or the drug. Together with further systematic testing of conditions to improve sample preparation (i.e. freeze-drying time, other fixation methods, range of cell washing conditions) it could be assessed whether the high intensity salt peaks originally reported in this work might be reduced with an improved preparation, i.e. comparing hydrated versus freeze-dried cells. With better preparation leading to improved secondary ion yield, different modes on the TOF.SIMS 5 could be used for analysis, i.e. delayed extraction with improved lateral resolution. This could enable 3D-image analysis without relying on more advanced SIMS systems such as Nano- or Orbi-SIMS. Furthermore, a deeper understanding of the peaks detected in healthy cells, using methods described above, could lead to an improved awareness of the changes that occur to cells after drug treatment.

Localisation of the PZ-128 drug on and in the cell could be further explored using Nano-SIMS, by heavy-isotope labelling of the drug molecule prior to cell treatment and performing dynamic SIMS and 3D-image analysis. Such a study could confirm the locality of the drug, whether it penetrates the cell and if it concentrates in specific parts of treated cells potentially clarifying the understanding of the drug mechanism. Finally, further pharmacological studies and assays could be performed to better understand the disruption of ion gradients observed and the cause of increased phospholipids and amino acid related peaks after application of the drug. As the clinical trials suggest, the drug is effective but the observed side effects and unexplainable behaviour provide cause for concern. Given the results presented in this chapter and other evidence surrounding the unexplained biological activity of PZ-128[128], which does not appear to be fully specific to PAR-1, further research regarding this compound should be undertaken before further clinical trials are carried out. Brouck suggests that "elucidating the on- and off-target activity of PZ-128 may offer the opportunity to design more selective drugs that achieve potent PAR1 inhibition while limiting the extent of adverse effects"[128]. A continuation and improvement of the current research could thus yield a potent and valuable addition to the available anti-arithmetic drug field.

5.5 Conclusion

A sample preparation and measurement method for the analysis of human coronary artery endothelial cells (HCAECs) has been tested and presented in this chapter. Out of the available methods, cryofixation and freeze-drying of the cells proved to be the most suitable approach resulting in a significantly better secondary ion yield of peaks of interest related to the cells. Scanning electron microscopy images confirmed that the sample preparation method used did not cause damages to the cells, suggesting the preparation method was valid. Using the trialled sample preparation method, untreated and PZ-128 drug-treated HCAECs have been prepared and analysed using ToF-SIMS.

The measurements illustrated that employing the tested methods the PZ-128 molecular ion could be identified as well as imaged on the cells. Four sets of untreatead and treated samples with 6 repeats per sample were used to generate a principle component analysis (PCA) model based on a single peak list containing all positive peaks found between m/z 18 and m/z 900. The resulting model separates the different sample sets on PC1 while PC2 shows good separation of the untreated from the treated samples. The PCA model loadings suggest that amino acid and lipid related peaks exhibit an increase in intensities after treatment of the drug while salt-ion peak show a decrease in intensity. As all samples were prepared equally with exception to the drug treatment, this indicates that the drug treatment is indeed causing these effects. A more in-detail comparison of assigned ion peaks of salt-ions, lipid and amino acid fragments between the untreated and treated samples confirms these trends. The observed changes to electrolyte and lipid intensity levels could be responsible for a number of adverse effects such as cell inflammation and arrhythmogenesis. However, due to this study only featuring a limited number of samples tested and based on the results gathered and presented, it is highly recommended to extend this study with a larger number of samples and additional tests to further validate these results.

Chapter 6

Agar-Based Bacterial Sample Method Development for Metabolite Tracking

6.1 Introduction

The word “antibiotic” was first used in 1941 by Selman Waksman, describing small molecules produced by micro-organisms that antagonize the development and growth of microbes[146]. In the 1940s and 1950s, the production of antibiotics based on fungi and soil bacteria enabled the treatment and prevention of bacterial infections worldwide, launching a new age for medicinal treatment[146]. However, human pathogens and bacteria quickly evolved, making the original antibiotic treatments less and less effective while requiring newer, more successful antibiotics to combat these resistant bacterial infections[146].

Today, antibiotic resistance is a major global issue that could affect the therapy of millions of patients by threatening the efficacy of antibiotic treatments[147]. Major factors driving this issue are the evolutionary response of microbes based on the overuse of antibiotics in both the food industry, where antibiotics are regularly used for livestock, as well as often unnecessary therapeutic purposes in humans[148,149]. The discovery and development of antibiotics is not a cost-effective process, resulting in the reduction in and the lack of new developments by the pharmaceutical industry[147,149–153]. This lack of new product is likely to become a major problem in the near future[147,149–153].

There are many different pathways to antibiotic discovery, however in the last two decades, metabolomics has been highlighted as a very effective method[149,154–156]. Metabolomics is the field of research relating to the analysis and study of metabolites, which are unique products and intermediates of specific cellular processes[157,158]. Metabolites can be seen as messages or answers by a biological system responding to its current state or the environment that the system is experiencing[158,159]. For example, in the presence of certain organisms, bacteria have evolved to produce metabolites to protect themselves from those organisms. Employing various analytical methods, in particular combining mass spectrometry with additional chromatographic separation techniques (e.g. LC-MS, GC-MS), metabolomics is used to identify and quantify large numbers of metabolites in biological systems, helping to map and unravel metabolite functions in relation to specific cells, bacteria and other organisms[158]. Some of these metabolites, based on fungi and bacteria, act as natural bioactive ingredients, including antibiotics, and are a major supply source for the pharmaceutical industry[160,161].

Microbial antagonism can be a formidable opportunity for such natural product research[162–164]. Here, microbial cultures are “pitted” against each other, invoking hidden natural responses through the simulation of “naturally occurring interactions”[165]. Co- and tri-culture experiments are used to elicit these responses from the many available microbe cultures and find previously unknown metabolites[154,165–168]. For example, Vinale et al. prepared co-culture experiments using two fungi species thus provoking the production of a to-date unidentified metabolite[165]. The metabolite belongs to a compound class known to reduce cancer proliferation showing that co-cultivation can be used for the identification and production of new bioactive metabolites and viably even medicinal treatments[165].

The use of high-throughput mass spectrometry techniques, metabolite libraries, databases as well as analytical and statistical methods is key to metabolomics research. Through their use, researchers are enabled to profile countless biological samples in the hope of uncovering and understanding the various analytes produced by microbial systems[155,156,169]. Desorption electrospray ionization (DESI) and matrix-assisted laser desorption ionization (MALDI) mass spectrometry are some of the leading surface analytical systems in metabolomics research and are widely applied due to their soft ionization and in comparison to other compatible surface analytical techniques high throughput[170–175]. In addition, the application of mass spectrometry imaging has facilitated the analysis of metabolic distributions and exchange patterns in microbial samples, thus allowing the "observation of the invisible"[176] and the possible discovery of unknown metabolites[173,176,177].

Some examples for MSI in metabolomics include Watrous who used DESI imaging to study and monitor the metabolite exchange of two distinct bacterial cultures after imprinting the co-culture samples on filter membranes[177]. In doing so the authors demonstrated a facile sampling method and its usefulness for metabolite research using DESI-MS in addition to offering a functional case study[177]; Gonzalez used MALDI-imaging, again in co-culture experiments, to monitor metabolic exchange patterns of various fungi and bacteria species after growing them on agar substrates[176]. Agar is a common substrate for culture media and is often employed in microbiological work. Lanni combined MALDI-imaging with ToF-SIMS to visualise metabolites in bacterial biofilms taking advantage of the high lateral resolution made available through SIMS while also enabling MALDI MS/MS analysis to assign and validate peaks of interest in regions significant to the samples[170]. Another example demonstrating the use of ToF-SIMS for microbial analysis is presented by Dunham et al.[178], who developed a method to quantitatively image agar-based microbial communities. Here, the authors first prepared standards based on the metabolites to be observed in the bacterial colonies and verified the standard deposition on the sample surfaces using additional analytical techniques such as SEM and MS/MS. The bacterial samples were then analysed and using the known reference standards to relatively quantify the measured analytes, quantitative SIMS imaging of bacterial biofilms was demonstrated[178].

Building on these published studies, a collaboration was initiated with the Duncan research group based at the Strathclyde Institute for Pharmaceutical and Biomedical

Sciences to trial methodologies for sample preparation and analysis of agar-based coculture bacterial samples employing ToF-SIMS. This chapter will describe the method development and initial application of a developed method to understand the difficulties faced when approaching this field using ToF-SIMS alone.

6.2 Experimental Methods

6.2.1 Bacterial Samples

During the method development, various *Streptomyces* strains were used to assess the viability of the preparation methods. All bacterial samples and work relating to their preparation was performed by Laia Castano Espriu. The bacterial strains were obtained from the Duncan lab strain collection at the University of Strathclyde. The *Streptomyces* strains were originally isolated from the Scotland-based subsurface of Trallee Bay (marine sediment). ISP2 Agar was purchased from Sigma Aldrich while all additional media and materials used for the sample preparation were obtained from the Strathclyde Institute of Pharmacy and Biomedical Sciences and if not in-house produced, procured from Sigma Aldrich.

Streptomyces

Streptomyces is a bacterial genus with over 500 described species[179]. They can typically be found in soil but also composts, water and plants[180,181]. This genus of bacteria is well-known for being able to produce bioactive secondary metabolites such as antibiotics and antivirals[180]. *Streptomyces* are responsible for 39% of all microbial metabolites that are exploited for medicinal purposes, embodying a major supply source of natural bioactive ingredients used in pharmaceutical products[160,182]. While this genus has already been successful in providing the basis for countless bioactive products, drugs and medicines, modern approaches could unlock further potentially vital pharmaceutical compounds hidden in *streptomyces* bacteria, thus making it an interesting sample species for method development[182].

6.2.2 Preparation Methods

The bacterial samples were prepared using a variety of methods, including oven-drying, freeze-drying, bacterial imprinting on cellulose membranes as well as nitrogen-drying. For each of the following four preparation methods, experiments included agar reference and *streptomyces* samples. All preparation methods used ISP2 agar sample plates. ISP2 agar (one litre of deionised water, 4 g Yeast extract powder, 10 g Malt extract powder, 4 g Dextrose, 20 g Agar and 18 g Instant Ocean) plates (10 mL) were prepared in triplicate.

Oven-Drying

Sample plates were incubated for 7 days at 30°C. After bacterial growth, a slice (10 mm x 5 mm) containing bacterial lawn was cut, transferred to a glass microscope slide for further assessment and dried for two hours at 45°C. The agar samples were typically adhesive enough to not require any further fixation. The second step involved cutting the culture into smaller sections (10 mm x 2 mm) and placed sideways as transversal slices. An example image of an oven-dried sample can be seen in Figure 6.1.

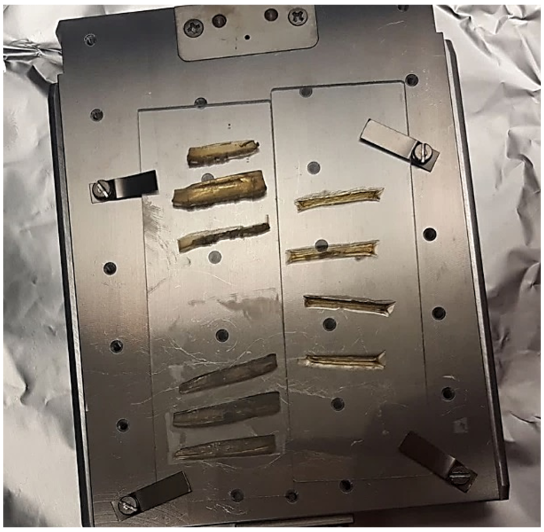


Figure 6.1: Oven-dried samples on the top-mounted sample holder.

Freeze-Drying

The sample plates were incubated for 7 days at 30°C. After bacterial growth, the cultures were, similarly to the oven-drying method, cut into smaller sections (10 mm x 2 mm) and placed sideways as transversal slices on a glass microscope slide. Samples were frozen overnight (-80°C) and lyophilised (Thermo Electron micro modulyo-230 freeze-drier) for four hours. An example image of a freeze-dried sample can be seen in Figure 6.2.

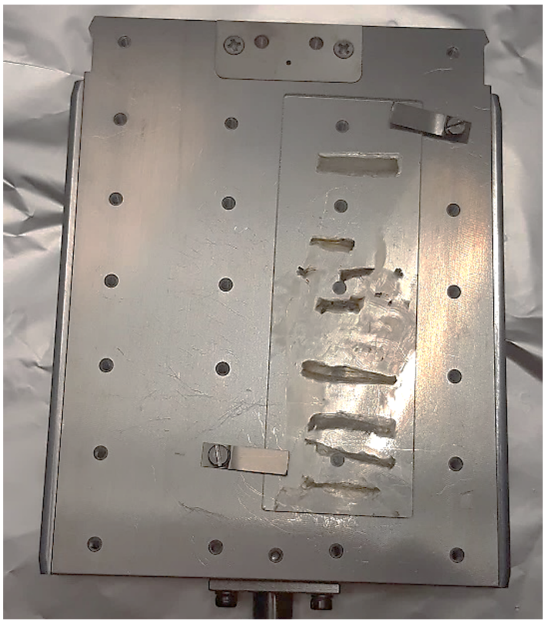


Figure 6.2: Freeze dried samples on the top-mounted sample holder.

Bacterial Imprinting

The sample plates were incubated for 7 days at 30°C. After incubation, cellulose membranes (Fisherbrand™ Grade 111 Cellulose Fast Qualitative Filter Paper) were pressed against the bacterial strains for 30 seconds after which the membranes were oven-dried for ten minutes. An example image of a membrane-based sample can be seen in Figure 6.3. This method is based on a paper published by Watrous et al. in 2010[177].

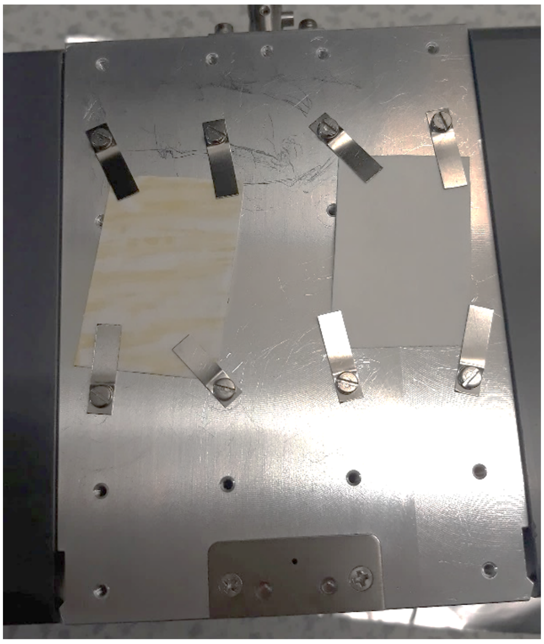


Figure 6.3: Bacterial imprint samples on the top-mounted sample holder.

Nitrogen-Drying

The sample protocol used for nitrogen-dried samples is based on a publication by Dunham et al.[178]. ISP2 agar sample plates were prepared with 7.5 mL solution instead of 10 mL and incubated for 7 days at 30°C. A slice from the bacterial culture was cut out (10 mm x 5 mm) and transferred to a microscope slide. Prior to attaching the removed culture sample onto the microscopic slide, double sided sticky tape was placed onto the slide to hold and stabilise the samples while drying. The samples were then dried under nitrogen for one hour. An example image of a nitrogen dried sample can be seen in 6.4.

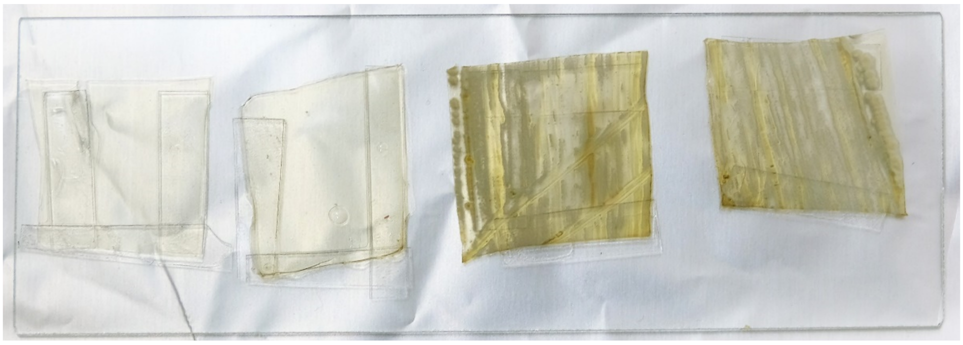


Figure 6.4: Nitrogen dried samples. Agar reference samples (2 left sample slices) and bacterial samples (2 right sample slices).

6.2.3 ToF-SIMS

Work regarding the ToF-SIMS instrument usage and analysis presented in this chapter has been performed by the author of this thesis. Analyses were performed using a TOF.SIMS 5 instrument (IONTOF GmBH, Muenster, Germany). The instrument was operated using a 30 keV $Bi_3{}^+$ primary ion beam. The analyses presented in this work have been done using both spectrometry and delayed extraction modes, employing a field-of-view of 500 µm^2 with a typical raster size of 512 px x 512 px. The total dose density was kept at 2e+11 ions/cm^2 to stay within the static limit.

6.3 Method Optimisation and Results

6.3.1 Bacterial Peaks of Interest

The initial method development was required to obtain a better understanding of the necessary sample handling procedures to process agar matrices and bacterial samples and prepare them for ToF-SIMS analysis. Prior to processing and drying, the agar matrix cannot be analysed within the ToF-SIMS instrument due to excess moisture prohibiting a high vacuum to be reached.

Hence, a first processing method attempt comprised oven drying and preparing agar and bacterial samples as described in 6.2.2. This resulted in very dry and brittle agar sample cuts that often deformed during the drying process. Drying the samples for shorter periods of time, however, resulted in the agar sample cuts being too wet for analysis, not allowing the vacuum to reach acceptable levels of below 10^{-4} bar in the loadlock chamber.

Once samples had been dried to a sufficient level using an optimised drying time and were successfully introduced into the instrument, the brittle sample surface as well as large bacterial growth areas on top of the sample proved to be topographically challenging. However, as the final goal was not to analyse the bacterial growth areas themselves but rather to identify secondary metabolites formed by the bacteria, these larger growth sections could be circumvented and measurements taken in the agar areas adjacent to the growth.

To identify initial regions of interest in the sample spectra, reference spectra of pure agar samples as well as bacterial-growth-adjacent areas were produced and measured. The employed ISP2 agar is a rich medium based on yeast, malt, dextrose and agar, a polysaccharide mixture. Viewing Figure 6.5, the orange spectrum represents a pure ISP2 agar sample with a complex spectrum of low mass peaks below *m/z* 200 and smaller islands of peaks up until the region around *m/z* 700 that could possibly represent repeating units. As the identification of peaks in the agar sample was not of priority this was not further attempted, however, as agar forms large polymeric structures, it is thought that the repeating units represent differently sized agar polymer units.

Instead, the *streptomyces*-growth-adjacent areas were measured, and spectra were produced. A representative sample can be viewed as the blue spectrum in Figure 6.5. As the analysed area contained largely ISP2 agar, a strong overlap of peaks with the pure ISP2 agar sample was observed. While some areas in the *m/z* 200 – 250 region appear to be *streptomyces*-specific peaks, most of the signals in this region also show overlapping agar peaks.

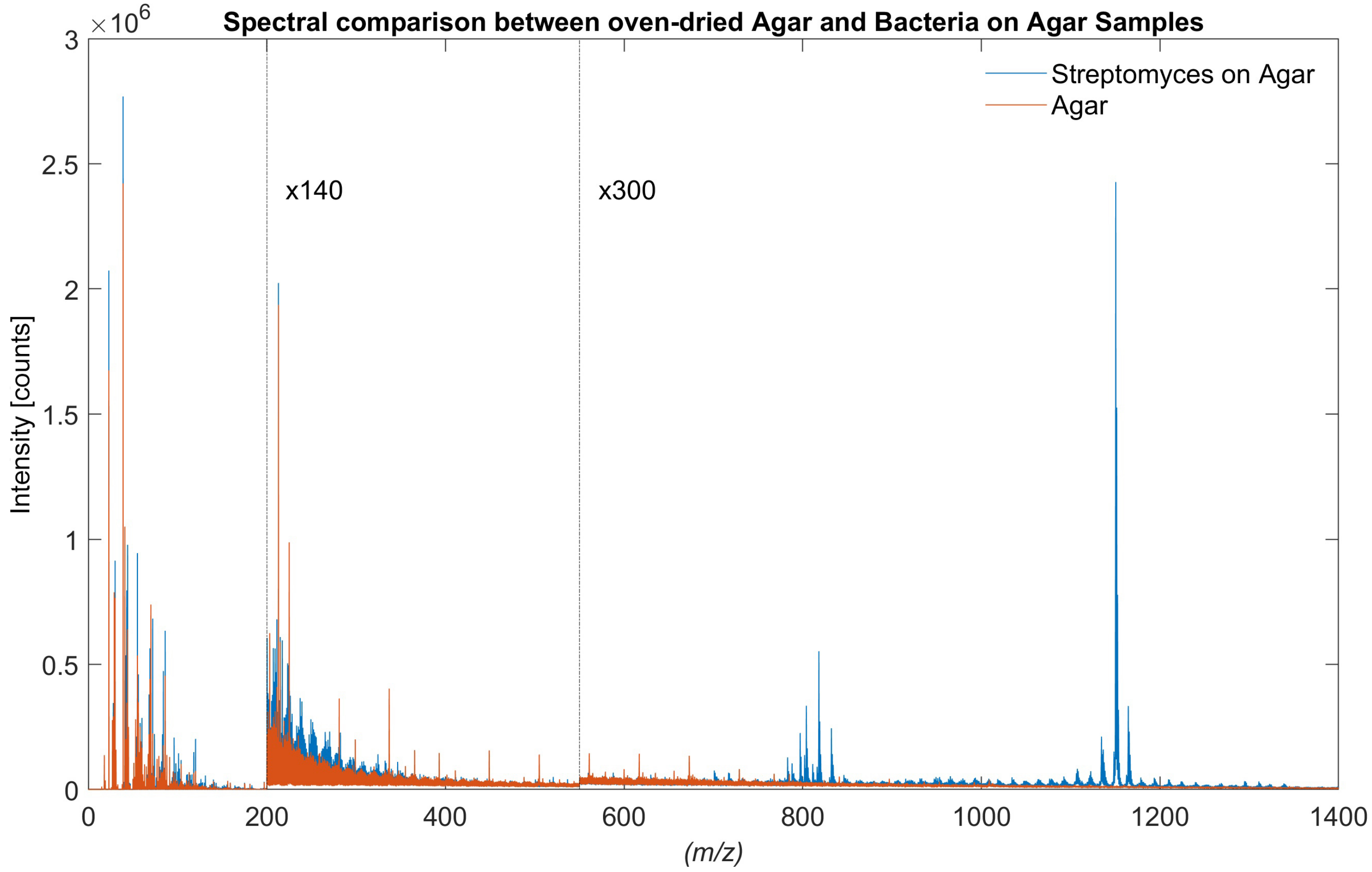


Figure 6.5: This graph depicts a spectral comparison between pure agar samples and samples of *streptomyces* on agar (oven-dried) in the regions between m/z 0 and m/z 1400. The dotted lines with multiplicators indicate the magnification factors by which the signals on the right side of the multiplicators have been enhanced for better readability.

Viewing the rest of the spectrum, three distinct spectral regions can be identified that show *streptomyces*-specific peaks representing bacteria-associated localised growth areas that appear to be embedded in the agar surface. These distinct regions can be seen in the spectral areas centred around *m/z* 700, *m/z* 820 and *m/z* 1150, with the latter two displaying particularly high intensities. For better visibility, magnified versions of these regions can be seen in Figures 6.6 and 6.7.

Regrettably, the TOF.SIMS 5 instrument does not have MS/MS capabilities (see Chapter 2) as some more modern SIMS instruments do and therefore a deeper analysis into the peaks produced by the bacteria was not possible. Nonetheless, these initial assessments show that flat agar areas adjacent to larger *streptomyces* growth zones can be analysed and directly distinguished from the agar matrix. Taking measurements of the bacterial samples thus appears to be possible and potentially viable. However, the initial sample preparation method used required further optimisation and the brittleness and topographically challenging surfaces of the samples made the acquisition of spectra difficult. In addition, further tests were required to see whether the application of delayed extraction could aid with some of the encountered issues while also enabling the direct imaging of the samples at hand.

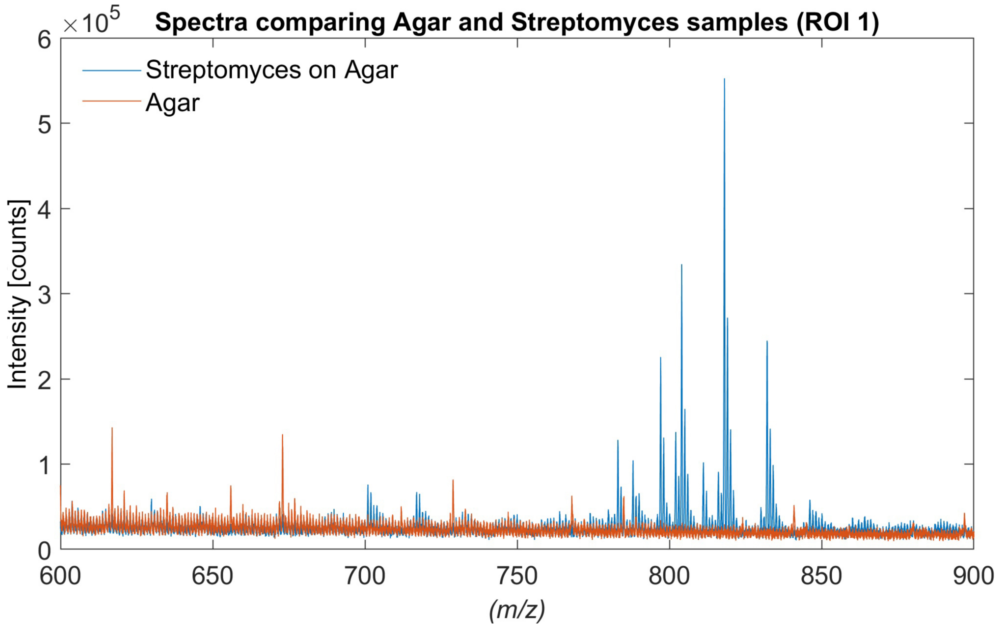


Figure 6.6: This graph depicts a spectral comparison between pure agar and *streptomyces* on agar samples (oven-dried) in the region between m/z 600 and m/z 900.

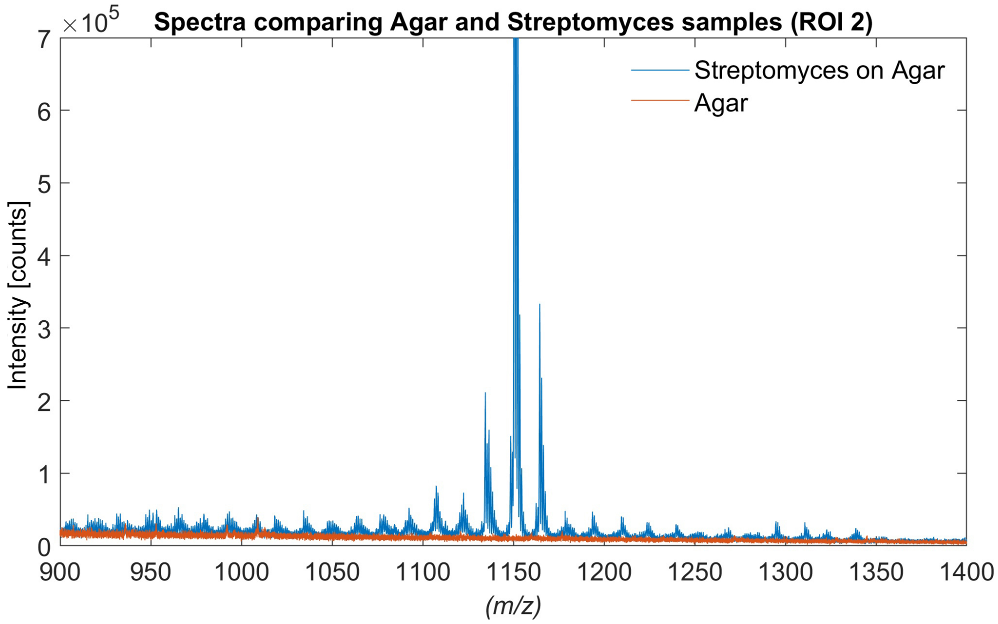


Figure 6.7: This graph depicts a spectral comparison between pure agar and *streptomyces* on agar samples (oven-dried) in the region between m/z 900 and m/z 1400.

6.3.2 Delayed Extraction versus Spectrometry Mode

In addition to applying spectrometry mode for simple spectra acquisition and quick sample analysis, the delayed extraction mode of the ToF-SIMS was used to aid with samples which required imaging and or were topographically challenging[183]. Agar-based *streptomyces* sample measurements were taken to assess the spectral differences between spectrometry mode and delayed extraction mode spectra for the bacterial samples at hand.

Figure 6.8 depicts a spectral overlay of a bacteria-specific peak region based on both delayed extraction and spectrometry modes. It shows that while higher intensity is achieved using the delayed extraction mode (blue), the peak resolution does suffer, reducing peak resolution values from about 3500 to 2500. These results were found to be highly dependent on the analysis area, though, and other regions on the sample surface showed a more significant gain in intensity when using delayed extraction while reaching peak resolution levels more similar to those of the spectrometry mode with values of up to 3000. A major disadvantage of the spectrometry mode was observed in regions with large height differences and topographical features on the sample surface, as these often resulted in a loss of resolution and peak splitting making measurements particularly challenging without the application of delayed extraction. However, on the other hand, spectrometry mode offers higher ion currents that could yield higher ion yields while also causing more fragmentation. This could be particularly helpful when needing to ionise smaller, lowly concentrated analytes.

The benefits offered by delayed extraction are, however, only achieved by significantly prolonging measurement times thus resulting in fewer potential sample measurements being taken per day. More details about the delayed extraction mode and benefits and disadvantages of using it in comparison to spectrometry mode can be found in Chapter 2. Delayed extraction was applied when imaging appeared of interest and time was not critical, otherwise spectrometry mode was a viable alternative, if the topography of the sample was flat enough to allow its use.

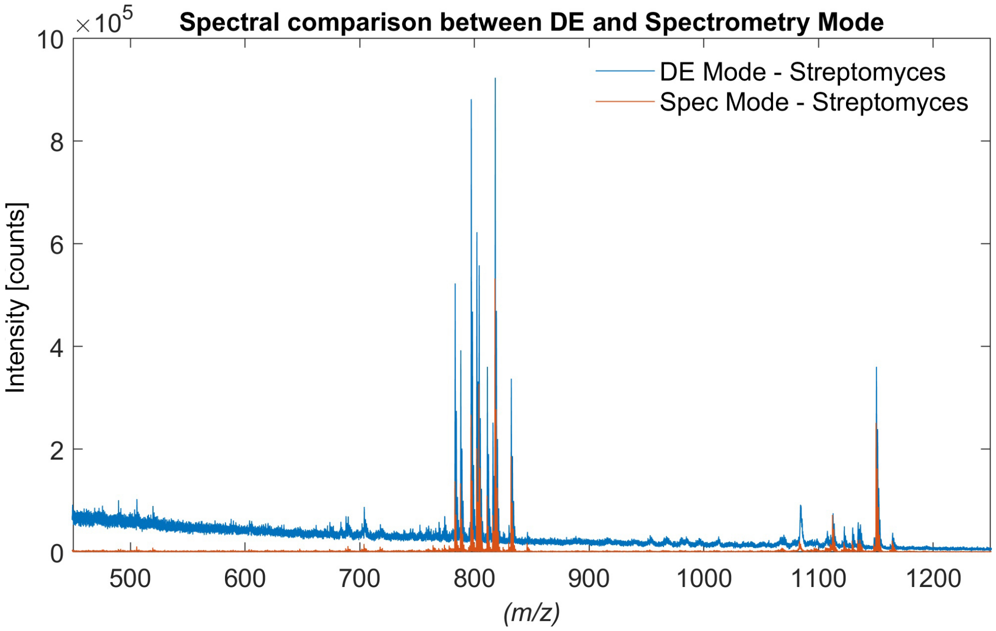


Figure 6.8: This figure shows spectra using delayed extraction (DE, blue line) and spectrometry modes (orange line) for areas of interest based on the same sample of *streptomyces* on agar.

6.3.3 Comparison of Various Sample Preparation Methods

With suitable ToF-SIMS settings identified, the sample preparation was further optimised. This section compares the results using samples produced with four different sample drying methods, namely oven-dried, freeze-dried, membrane-based and nitrogen-dried samples. For more details about the drying methods, please refer back to Section 6.2.2. Based on the results from Section 6.2.1, a *streptomyces*-specific m/z region (m/z 1080 – 1220) was chosen to compare spectra produced from samples obtained using the various drying methods (see Figure 6.9).

The initial comparison focuses on the spectral features while other aspects of the methods are compared towards the end of this section. All spectra were acquired using the same delayed extraction mode instrument settings outlined in Section 6.2.3. While typically at least 5 measurements per sample were taken, a single representative spectrum per sample has been chosen to be displayed in this chapter.

Figure 6.9 depicts a spectral comparison of the four sample preparation methods for the region of m/z 1080 – 1220. An additional magnification is applied in the lower part of the figure, focussing on the peaks between *m/z* 1150 – 1154 and emphasising the intensity differences between the samples presented. Viewing the figure closely, it is clear that the nitrogen-dried samples have the highest intensity. Although the oven-dried samples show similar signal strength, their spectra intensities are slightly lower and they also miss several additional peaks that can be seen in the nitrogen-dried samples.

To be more specific, between *m/z* 1080 and 1140, a number of additional peaks stands out that cannot be seen in any of the other methods. Some areas in this region show small intensity peaks produced by the oven-dried samples, however, these are significantly smaller than the peaks produced from the nitrogen-dried samples. Viewing the most expressive *streptomyces* peaks in the *m/z* 1150 – 1154 region, the membrane and freeze-dried samples indicate a lack in intensity but are comparable in their performance. They are, however, significantly outperformed by both oven-dried and nitrogen-dried samples.

Based on the acquired spectra, the nitrogen-drying method is the most successful, leading to higher-intensity peaks and more peaks of interest not expressed in the other samples. To choose an optimised preparation procedure, the methods are also compared with regards to their ease of handling and effort in preparation. The oven-drying method was the simplest drying method available. Samples, once prepared, are cut out, positioned on a glass-slide and placed in an oven. Once dried out, the glass slide is mounted directly onto the top-mounted sample holder and introduced into the ToF-SIMS instrument. However, the prolonged heat and air-drying results in the samples becoming very dry and brittle. After treatment, samples appear deformed and crumble when touched, producing sample sections that are difficult to handle. This also leads to the previously mentioned height differences and extensive topographical effects, making these samples overall difficult to measure.

Moving on to freeze-drying, the agar slices require drying in a dedicated instrument and additional care when handling. However, the freeze-dried samples suffer from similar effects as oven-dried samples. During the freeze-drying procedure, the samples also become extremely brittle and show even more deformation than samples prepared using the oven-dried method. This renders them even more difficult to analyse resulting in less acceptable measurement outputs. Besides the previously mentioned drying methods, a suggested mounting method by Dunham et al. was attempted[178]. In nitrogen-drying, a steady non-heated flow of nitrogen is applied to the sample surfaces as a drying mechanism, after placing the agar cut-outs on double-sided sticky tape to keep their form during the drying process. The flow of nitrogen is applied until the agar slice appears to be dried, with a typical drying time of one hour. Issues were encountered due to variable thicknesses of the agar slices and thus different drying time requirements resulting in the need to remount some samples after further nitrogen application[178]. The application of double-sided sticky tape did indeed aid the agar slices to stay in shape during the drying process resulting in less deformation and better samples.

In addition to the three drying methods, membrane imprinting was also tested as a fourth technique. As described in 6.2.2, a cellulose membrane is pressed onto a bacterial sample, transferring material from the bacterium and the agar surface onto the membrane. The method was simple to carry out and removed the need to dry and prepare larger agar slices before transferring them to the instrument. However, during the membrane imprint measurements some challenges were encountered. Firstly, imprinting from wet agar and bacteria samples resulted in uncertainty surrounding how much material was transferred during the contact. Secondly, when applying pressure to the membrane manually an additional factor of variability was added, and a more secure and replicable pressure system would be recommended for further studies. While the cellulose membranes initially appeared to be flat, the process of handling, imprinting and wetting the cellulose membrane also resulted in slight deformations of the material. In particular, after drying the membranes tension was required from all sides to provide an even analysis surface. Finally, it was difficult to identify which area to analyse within the ToF-SIMS spectrometer as the outlines of the bacterial colonies were not visible on the membrane.

After trialling these various preparation methods, the results suggested that nitrogen-drying was the most reliable method with respect to sample handling, surface topography and/or deformation of the agar slices and the resulting spectra. In addition, samples analysed using nitrogen-drying showed peaks not visible through other preparation methods, while peaks that were barely visible in any of the other trialled methods were observed at significantly high levels, making this the sample preparation method of choice.

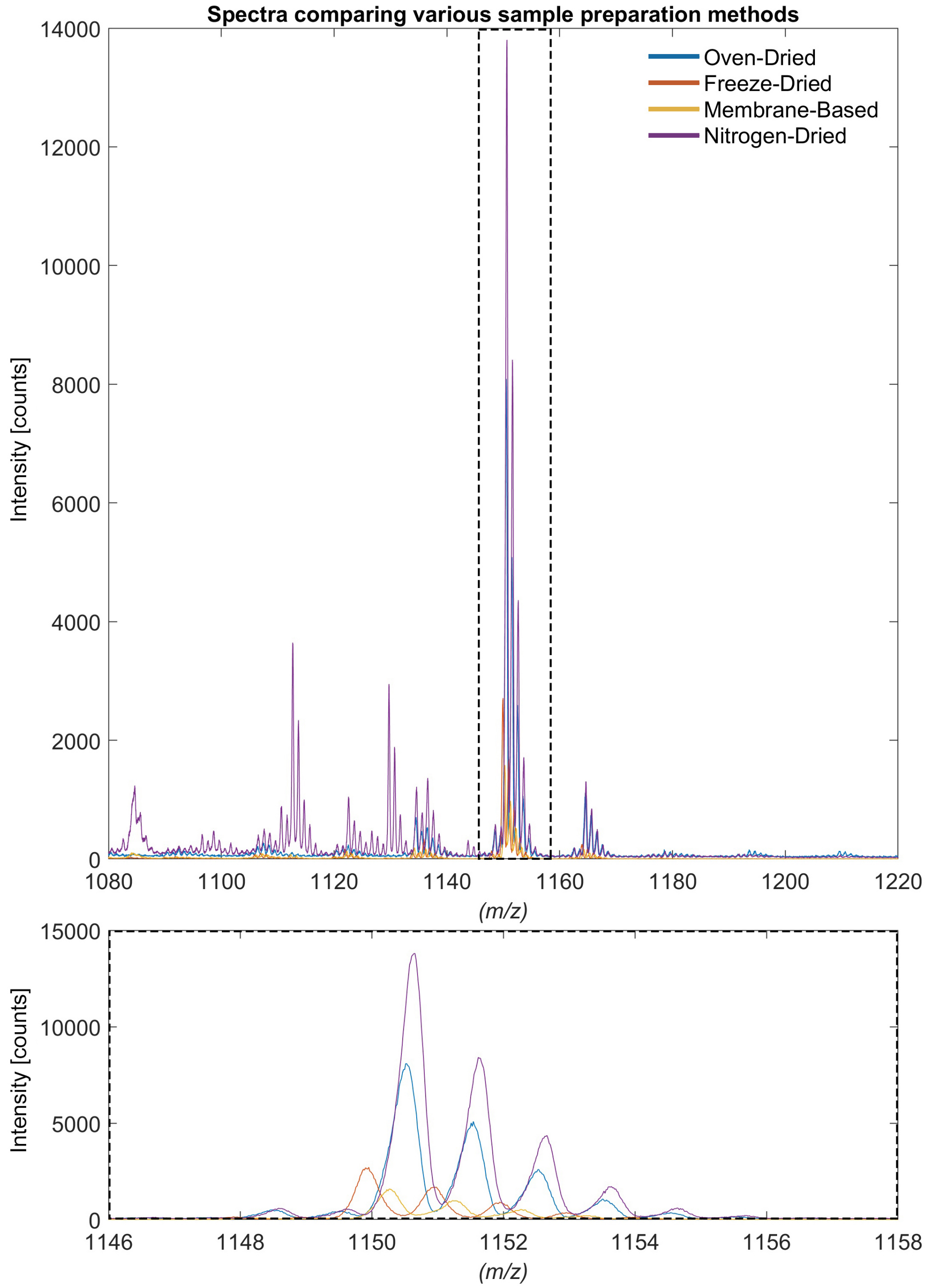


Figure 6.9: This figure depicts a spectral comparison of the four trialled sample preparation methods focussing on a m/z region of 1080 – 1220 which shows *streptomyces*-specific peaks. For better visibility, the dashed area in the upper region of the figure has been magnified and shown in the lower region of the figure.

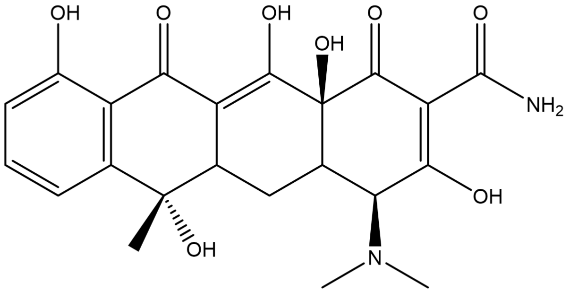


Figure 6.10: Structure of tetracycline with the chemical formula $C_{22}H_{24}N_2O_8$ and a molecular weight of 444.43 u.

6.3.4 Tetracycline Case Study

Prior to applying the sample preparation method to co- and tricultures, further work was necessary to confirm whether bacterial metabolites could be identified within the agar matrix.

Streptomyces rimosus was selected as a test case for further study, due to its well-known ability to produce tetracycline, an antibiotic widely used in the treatment of bacterial infections. Tian et al have previously used ToF-SIMS to measure and image tetracycline within single E. coli cells[48], identifying characteristic ions of tetracycline as *m/z* 445.2 $[M + H]^+$, *m/z* 427.2 $[M + H - H_2O]^+$, and *m/z* 410.1 $[M - (OH)^2]^+$. Building upon this work, experiments have been carried out in order to determine whether tetracycline could be identified in agar samples, initially analysing tetracycline on its own before progressing to tetracycline-spiked agar and *Streptomyces rimosus* samples.

Preliminary experiments were carried out on tetracycline without any media. A 100 mg/mL tetracycline stock solution was prepared in 70% ethanol (in water) and additional dilutions of 0.5 mg/mL and 0.01 mg/mL were also prepared. Adhesive tapes were spiked by using 22 µm syringe filters to deposit 10 µL drops of the stock and dilution solutions, achieving tetracycline deposits of 1 mg, 50 µg and 1 µg on tape, respectively. Additional reference spectra of the adhesive tape were taken for comparative purposes. The results in Figure 6.12 show the mass region *m/z* 443 – 455, focussing on the reported molecular ion peak by Tian et al. An additional large peak can be seen at *m/z* 449.2 which is proposed to be $[M + Na - H_2O]^+$. To better see the differences between the different concentrations of tetracycline, a magnified view showing the mass region *m/z* 443 – 455 is shown in Figure 6.12. Here it is clear that tetracycline could not be identified in the lowest concentration of 1 µg, however it could be observed in the two higher concentration samples.

With the tetracycline molecular ions identified, as well as pure reference spectra obtained, the next step involved trying to identify tetracycline within the agar matrix. Agar samples spiked with tetracycline at equal concentrations as shown in the previously reported adhesive tape samples were prepared to verify the antibiotic could be identified. Here, 7.5 mL of ISP2 agar were spiked with 1 mg/mL and 50 µg/mL of tetracycline. The results can be viewed in Figure 6.12, which highlights the mass region *m/z* 443 – 455. As a reference, the 1 mg/mL adhesive tape tetracycline sample was used to see whether any overlapping peaks can be seen. However, even when viewing the entire spectrum, in none of the samples, across the two tested concentrations, could any tetracycline-specific peaks be identified in the agar-based tetracycline samples.

As no tetracycline could be identified in the agar matrix, even at elevated concentrations, it was deemed unlikely that the lower concentrations produced by bacteria could be observed if they were released into the agar growth medium. As a result, further work using the *streptomyces rimosus* were postponed until a better method of analysis could be found.

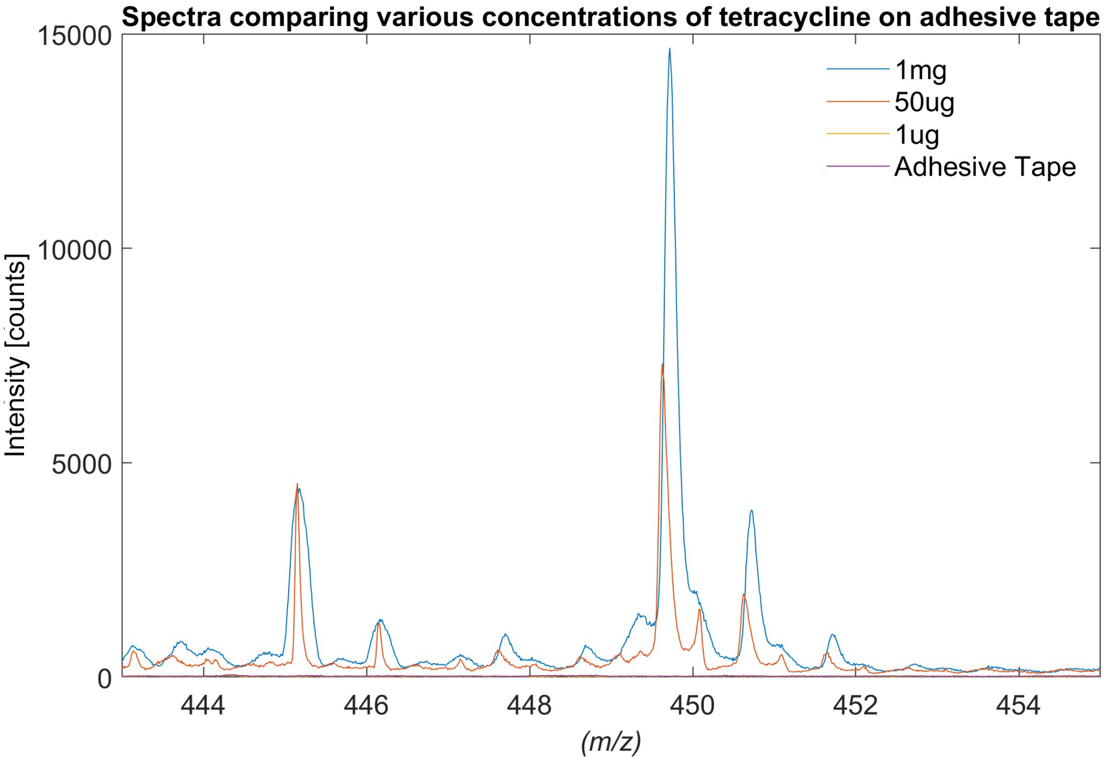


Figure 6.11: This figure depicts spectra of adhesive tape laced with tetracycline drops of different concentrations. The mass region is chosen to highlight the [M + H] and likely [M + Na – H_2O] peaks of the antibiotic at m/z 445.2 and 449.2.

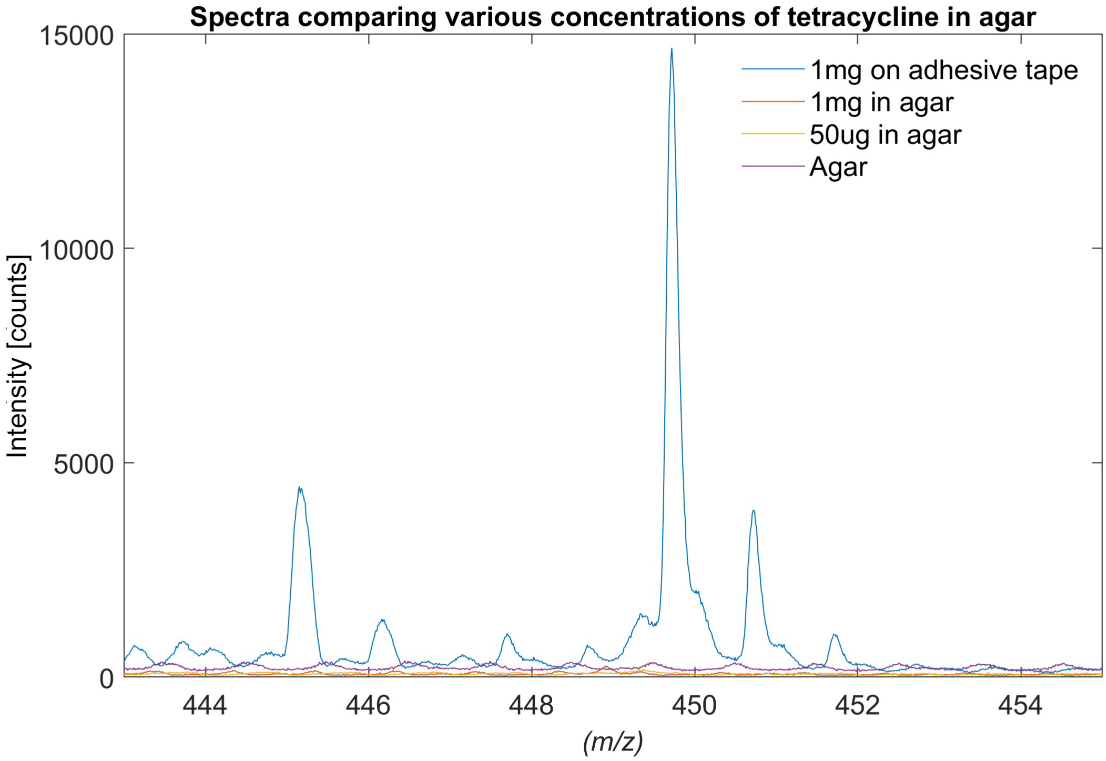


Figure 6.12: This figure depicts spectra of agar laced with different concentrations of tetracycline. The mass region is chosen to highlight the [M + H] and likely [M + Na − H_2O] peaks of the antibiotic at m/z 445.2 and 449.2.

6.4 Discussion

This section discusses the drying and sample preparation methods as well as the ToF-SIMS results presented in this chapter. With regard to sample preparation methods, oven-drying was found to be a very simple technique, did not require any complex setups and was therefore the first method trialled. The results showed bacteria-specific peaks could be identified on the agar surface. However the dryness, brittleness, and deformation encountered during the preparation and handling caused some issues. This method has potential for more optimisation, in particular if the deformation and brittleness could be controlled, such as by using lower drying times and temperatures which could potentially improve the process. Furthermore, an adhesive tape for mounting the agar sample slice, similar to the method employed with the nitrogen-dried samples, could have been utilised to reduce the surface deformation. Finally, it is possible that the exposure to added heat through the drying process affects the agar matrix, modifying or degrading the agarose polymer and making it more brittle. It would be interesting to see how lower drying times and temperatures during the agar sample drying would affect the peak intensities and resolution.

Following on from the oven-drying, freeze-drying was the next method trialled as it had shown promising results for the cellular samples described in Chapter 5. However, similar issues as with oven-drying were encountered using this method, where the agar also became very dry, brittle and deformed. After freeze-drying treatment, many samples were so dry that upon transfer parts of the sample slices would break off and crumble. Due to the extreme drying temperature and the removal of water from the sample matrix, the structure of the agar changed significantly. Similar to the extreme high temperature from oven-dried samples, the extreme low temperature used here is shown to result in reduced peak intensity and resolution. Due to these underwhelming results, this method was discontinued. The dehydration caused too many changes to the sample matrix without any gain with regards to spectral performance. It is likely that a less denaturing drying method should be preferred versus these extreme temperature conditions when it comes to bacterial samples.

In comparison, the imprinting method presented by Watrous et al. appealed for many reasons[177]. Samples did not require additional preparation, did not undergo change due to the drying process and no larger agar sample slices needed to be introduced into the vacuum. Thus, disregarding the imprinting procedure itself, the sample system was not changed[177]. Instead, after imprinting, cellulose sheets containing potential analytes were transferred and could be analysed within the instrument. However, some issues were found with this method, such as the unknown amount of pressure required to transfer material onto the cellulose sheets. It was also undetermined whether all materials of interest, such as metabolites, would transfer to the cellulose, as they may not be localised at the top of the agar surface where the imprint occurs. Furthermore, it was not easy to identify visually what area on the cellulose corresponded to which area of the bacterial colony, especially after the slight deformation that occurred after drying, thus making the ToF-SIMS measurement difficult. Finally, the cellulose membrane material lead to significantly more surface charging compared to the agar samples, thus requiring the application of very high surface potential in addition to the electron flood gun to counter this.

Many of these problems could possibly be accounted for using a similar method suggested by Debois et al. in which a silicon wafer was used for imprinting. Silicon wafers are flat, easily cleanable and are conductive, reducing surface charging of the samples[184]. Still, only a surface-based imprint of the bacterial sample can be made using this technique, thus not revealing anything about the potential material stored within the sample matrix. However, when the agar sample matrices are introduced into the system, bulk analysis is possible, in contrast to imprinting the sample surface. If further experiments were to be trialled with the goal to measure or image the surface of the agar, the results presented by Watrous et al. and Debois et al. showed promising findings and could be trialled further[177,184]. Nevertheless, using cellulose paper introduced too many issues in the study at hand and compared to the oven-dried samples did not deliver better results, leading to the discontinuation of the method at the time.

Moving on to the final drying method, nitrogen-drying, as originally proposed for bacterial samples by Dunham et al.[178]. Here, agar slices were mounted on double-sided sticky tape prior to the drying procedure, reducing the deformation of the sample surface. A steady flow of nitrogen over the samples at room temperature was used to dry out most samples successfully. However, due to variation in agar thickness between samples, some took longer to dry than others, an issue that was only identified when introducing them into the ToF-SIMS and being unable to pump down the vacuum.

In their paper, Dunham et al. suggested a shrinkage factor of the agar samples of approximately 26%, with the largest shrinking occurring at the outer edges of the samples[178] while still maintaining the original surface features. Although not confirmed quantitatively, the same observation was also made in this study using the nitrogen drying, compared with the oven-drying and freeze-drying methods which appeared to cause more shrinkage, possibly leading to a loss of surface features.

In the methods compared in this chapter, nitrogen-drying thus lead to significantly less deformation of the samples resulting in easier handling and less brittle surfaces. The ToF-SIMS signal intensity and resolution of the analysed samples also showed better results than any other method employed. This is in line with the findings presented by Dunham et al., where the authors also asserted that this drying method was preferred to other methods such as oven-drying and freeze-drying[178].

Employing the optimised preparation method, a case study was attempted to analyse agar-based *streptomyces* bacteria and the measurement of their tetracycline metabolite production. However, the measurements did not produce any known peak response for the compound itself. As tetracycline could be measured in its pure form as shown in the reference sample experiments presented in this chapter, as well as in the results shown by Tian et al., this result was unexpected[48].

A major issue encountered in ToF-SIMS analyses is the matrix effect, as explained in Chapter 2. Here, the chemical environment that a sample is analysed in can lead to significant suppression or enhancement of secondary ion yields of analytes, one of the main difficulties encountered when applying ToF-SIMS in quantitative analysis. It appears this is one of the possible reasons that no tetracycline signal can be detected in the samples. The agar sample matrix could thus be causing undesirable matrix effects, leading to no tetracycline peaks being detected in the spectra due to suppression of the signal. While the tetracycline would be far more embedded in the agar sample in comparison to the direct deposition on the reference sample, not detecting any signal indicates problems with the sample matrix or mistakes in the experimental setup. With two repeats at different concentrations performed, it is unlikely that the latter is the cause, however if future experiments were to be conducted, it is advised to increase the concentrations used in the agar samples to explore this path. As previously stated, agar is a polymer that, in addition holds many other compounds which only adds to the complexity of analysing compounds within. Another feasible reason for the lack of a tetracycline signal could be that the complex and rich agar sample matrix reacts with the compound, incorporating it into the sample matrix. However, as no unique peaks could be found that would suggest combinations of tetracycline with other compounds in the agar, this cannot be proved or disproved.

Finally, Dunham et al. presented a method to quantitatively measure bacterial metabolites on an agar surface. In their method, the authors used chemical inkjet printing to print reference standard on top of an agar surface to normalise response intensities of analytes that were produced by bacterial samples[178]. Unfortunately, this methodology could not be applied here, in particular because a first step must show that small concentrations of tetracycline can be identified within the agar sample matrix at all which was not achieved in this series of experiments. Overall, it is possible that the ToF-SIMS instrument used for the study presented in this chapter was not ideal and more sensitive instrumentation with MS/MS capabilities and better detection limits could be more successful.

Due to time restrictions, no further experiments with agar and tetracycline could be performed. If the agar matrix is the leading obstacle, other preparation methods exist that may be used to completely circumvent the usage of this bacterial lawn such as liquid cultures, though they also represent a rich and complex sample matrix. A precursor study to the Dunham et al. paper, presented by Baig et al. showed the successful application of liquid culture analysis with ToF-SIMS, identifying the same analytes used in the later study by Dunham et al.[178,185].

When working with biological samples in ToF-SIMS, one of the most important steps for successful analysis is an appropriate and effective sample preparation method, as this study clearly showed. Identifying the correct technique for the sample at hand can be a difficult and, due to method development requirements, a lengthy process. Even when good preparation methods are found, matrix effects, challenging sample topography and difficult post-processing and data analysis can still complicate measurements and the production of useful data. In addition, biological samples themselves, such as cells and bacteria in particular, are highly complex and the identification of peaks of interest using ToF-SIMS alone is exceptionally difficult without the use of additional techniques such as MALDI-TOF, LC-MS/MS and/or high resolution and mass accuracy instruments.

6.5 Conclusion and Future Work

Overall, technical difficulties, short project times, and a complex sample system made this a very difficult project. It is clear that metabolomics and the search for metabolites in bacterial and co-cultures samples are important undertakings and that the use of modern mass spectrometry techniques are key to unlocking many still hidden metabolites and co-factors. While it is an interesting topic to pursue, extensive method development and in-depth background knowledge of the systems to be analysed and the analytical techniques employed are required.

In this chapter we tested four different bacterial sample drying and preparation methods and found one to work significantly better than the others, namely nitrogen-drying. In comparison to nitrogen-drying, oven and freeze-drying were found to extract too much moisture from the samples, resulting in deformed surfaces and difficult sample handling. This effect was minimised with the nitrogen-drying method leading to improved sample handling and better spectral results. Bacterial imprinting, while showing promise, was found to require further work and adjustments to the method. The use of cellulose membranes as the imprinting material, in particular, caused many of the difficulties and issues encountered, such as surface charging and sample handling; in future work this could likely be solved through the use of silicon wafers to be imprinted on.

Using nitrogen-drying as the preferred sample preparation and drying method, the analysis of tetracycline producing *streptomyces* bacteria was trialled as a case study. While the analyte was identified in a pure reference sample, tetracycline could not be measured within samples with agar as a medium. It is assumed that matrix effects, such as the high levels of salt and other compounds in the agar medium, causes the ionisation of the analyte to be suppressed. As no tetracycline could be identified in multiple spiked samples, the experiment was stopped at this point. As proof exists of other groups identifying metabolites in agar using ToF-SIMS, it is likely that the issues encountered were very analyte-dependent and further work with other analytes should be continued. However, for future work, it is recommended that ToF-SIMS should be applied in combination with other mass spectrometry techniques to aid the identification of compounds and back up the resulting measurements. To help identify whether ToF-SIMS and the sample matrices were the limiting factor in the analysis, a complementary analysis using MALDI-MS/MS could be applied. ToF-SIMS can provide added value to any analysis that requires or benefits from high lateral resolutions and can further be employed for the analysis of bulk samples.

To extend the work performed in this study, the next steps should involve testing liquid cultures as a sample medium to identify whether tetracycline can be detected in other types of media. Otherwise, a larger systematic study with known metabolites of bacteria of interest could be performed aiming to find metabolites that can be observed in both solid and liquid cultures. This would lay the groundwork for future experiments before moving on to more complex co-culture studies.

Chapter 7

Summary and Outlook

7.1 Summary

This thesis probes the suitability of ToF-SIMS for applications in the field of pharmaceutical research & development by investigating three different areas of interest. The work also introduces a software toolbox for pre-processing method selection of spectral data.

In the first study, the use of ToF-SIMS for pharmaceutical material characterisation was investigated by examining the surfaces of paracetamol crystals with three distinct 4-nitrophenol impurity loadings, a surface based droplet application (PDN), an impurity epitaxial growth (PEN) as well as the integration of the impurity via a cooling crystallisation process (P4%N). The measurements show that 4-nitrophenol could be clearly identified and localised in all impurity loading scenarios. Ion images of the PDN sample indicated the impurity covering most of the surface of the crystal at the drop impact site, the PEN sample exhibited epitaxially grown impurity crystals and the P4%N sample displayed a homogenous impurity distribution on the crystal surface. Evaluation of the crystals using optical microscopy methods and SEM revealed significant differences in surface texture between the pure paracetamol samples and the impurity loaded crystals of PDN, PEN and P4%N, respectively. While moderately visible using the ToF-SIMS, the optical microscopy and SEM results were clearer and offered a more focussed view. However, the use case demonstrated the feasibility of the system with regard to its capability to provide chemical as well as topographical characterisation simultaneously. To extend this work from the surface into the bulk, depth profiling of paracetamol crystals with another embedded impurity, namely 4% mol 4-acetamido benzoic acid (P4%A), was performed. The findings indicated that contrary to the P4%N crystals, the P4%A samples exhibited a highly surface localised distribution of the impurity that, after a few sputtering scans, was reduced by at least one order of magnitude in intensity. Once in the bulk however, the impurity behaviour was found to be similar to the P4%N crystals previously analysed, a homogenous distribution of both the paracetamol as well as the impurity molecules. The experiments raised interesting questions regarding the preferred settlement of impurities on the surface versus the bulk, which could lead to the design of better post-purification processes. This small use case thus exhibited the value and potential of the instrument to not only perform surface but also bulk characterisation of pharmaceutical materials for future studies.

In the second study, ToF-SIMS was applied to aid pharmacological studies investigating suspected off-target effects of PZ-128, an antithrombotic drug undergoing clinical trials. To facilitate the work, method development to identify a suitable sample preparation and analysis method was performed. Out of the available tested methods for preparing the human coronary artery endothelial cells (HCAECs), cryofixation combined with freeze-drying were identified to be the most successful. Untreated and drug-treated cells produced using this sample preparation method were analysed using SEM which confirmed the integrity and viability of the samples. Using ToF-SIMS, PZ-128 could further be identified and imaged on the surface of drug-treated HCAECs. A comparison between untreated and drug-treated HCAECs was performed focussing on changes in salt, amino acid and phospholipid peak intensities. It was found that cells treated with the drug expressed a reduction in electrolyte levels as well as an increase in both phospholipid and amino acid peak intensities. The reported intensity variations after treatment are indicative of changes to the cells that could lead to malfunctions and inflammation of the HCAECs as well as an increased likelihood to trigger arrhythmia. The ToF-SIMS study findings strengthen the original hypothesis of the pharmacological studies, suggesting that PZ-128 does indeed cause off-target effects which can be responsible for adverse outcomes as reported in Phase 1 of the clinical trials.

The third pharmaceutical application study explored the use of ToF-SIMS to identify and localise secondary metabolites produced by agar-based bacterial cultures. While this is a common application for other MSI techniques such as DESI and MALDI, there are not many cases reported in literature attempting such work using ToF-SIMS. Initial method development was required to identify the correct sample preparation procedure that would allow the analysis of bacterial samples on the agar matrix. Comparing a number of different preparation methods by viewing spectra of an agar-based streptomyces bacterial strain, the best results were observed when nitrogen-drying the samples at room temperature using double-sided tape to prevent deformation of the agar during the drying process. Using this method, the samples showed the least deformation while also offering higher peak intensities and peak resolution in comparison to the other methods tested, including oven-drying, freeze-drying and membrane imprinting. A test case of the method was conceived attempting to identify tetracycline, a known streptomyces metabolite, in an agar matrix. Reference samples of the pure metabolite indicated a good fit for the experiment, however, measurements of an agar matrix spiked with high concentrations of the metabolite failed to deliver any metabolite signal. It is assumed that the tetracycline signal was suppressed by the agar sample matrix. Future studies are recommended to try other metabolites and bacteria combinations and possibly investigate other bacterial lawns that might not be as detrimental to the studies.

In addition to ToF-SIMS experimental studies, this thesis also introduces a MATLAB-based toolbox for selecting pre-processing methods for spectral data. Pre-processing refers to cleaning and preparing data prior to its further usage for analysis and modelling and is the first step to making data comparable. Yet, the application of pre-processing steps and methods is often not standardised leading to laissez-faire and trial-and-error style usage of these methods. Within this context, a toolbox was developed that applies a design-of-experiment brute-force approach to test available pre-processing methods using partial-least-squares regression. Applied to a specific spectral data set, the toolbox proposes the most successful combination of pre-processing methods based on comparing all available methods against their resulting PLSR model parameters. It has a graphical user interface and presents plots to examine the data in more detail as well as suggesting the best pre-processing methods available. Two use cases for pre-processing NIR measurement data were presented, one based on the mixing of two solvents and the other founded on an open dataset of a content uniformity measurement study of pharmaceutical tablets. In case one, the pre-processing strategy proposed by the toolbox led to a 53% error reduction compared to basic mean-centring of the data whereas for case two, following the toolbox pre-processing methods led to a 30% reduction in relative predictive errors in comparison to the published results this data was based on.

Summarising the different studies, this thesis showed that ToF-SIMS is a very versatile and valuable analytical platform that can be used for many applications throughout pharmaceutical science. From pharmaceutical material characterisation and stability surveys to drug detection in single-cells and biological studies, a major advantage of ToF-SIMS is its variety of operation modes and analytical paths. Mass spectrometry, high resolution mass spectrometry imaging, depth profiling as well as 3D-analysis of materials are all available via a single instrument and with the exception of the latter their successful usage has been demonstrated throughout the use cases presented in this thesis. However, there are some drawbacks that make working with ToF-SIMS challenging. Ionisation probabilities between different materials vary largely and sample-induced issues such as rough topography, differential charging as well as matrix interferences can make the analysis of samples difficult. Measurements and analysis are time-intensive, in particular, as has been shown, when method development is required. Not all samples can tolerate ultra-high vacuum conditions and finding a fitting sample preparation method can be laborious while often being crucial for successful analyses. While the first two studies discussed in this thesis are good examples of successful applications, the third study shows that success is not guaranteed for every ToF-SIMS analysis.

ToF-SIMS is a very specialised analytical technique and depending on the question at hand there might be other analytical platforms that can offer easier solutions. As stated, due to its more qualitative nature, ToF-SIMS requires the use of additional techniques to corroborate and validate its results. However, when it comes to high-resolution chemical imaging and related applications, ToF-SIMS offers great potential to solve niche problems. Combinations with other complementary techniques, such as MS/MS enabled systems for peak identification, SEM for higher-resolution images and data fusion purposes as well as LC- and GC-MS systems for quantitative analysis, were demonstrated in various occasions in the studies presented here and proved to improve in many cases the in-depth sample investigation. Newer SIMS systems with in-built MS/MS capabilities as well as hybrid solutions such as the Orbi-SIMS are expected to simplify such comprehensive investigations further, thus continuously pushing the capabilities of SIMS and its application possibilities in the pharmaceutical industry.

7.2 Outlook

Despite the varying complexity and success of the studies presented in this thesis, future studies in all three areas of investigation are recommended, namely material characterisation, analysis of drugs in cells as well as drug discovery. For material characterisation and impurity analysis, a particularly interesting area of further investigation would be the study of isotopically labelled isomers of paracetamol introduced as impurities into crystals to further investigate the influence of those on crystal growth and structure. In addition, moving on from singular API crystals, it is suggested to perform surface and bulk analysis studies of pharmaceutical drug product undergoing enhanced stability testing. These samples are likely to be of interest and such studies could prove to have high impact.

Further investigations with regard to PZ-128 studies are limited by scheduling of clinical trials and other investigation avenues around the drug. If more time was available, it is recommended to run extended studies with larger numbers of cells to build a more statistically valid sample set. In addition, depth profiles and 3D-analysis of cells are recommended to more accurately localise the drug within the cells. In general, the use of ToF-SIMS and SIMS-based techniques was found to be a good fit for investigations on the subject of drug localisation within cells. It would be interesting to extend this research to other medicines and should be done in a complementary manner with other analytical techniques to validate and quantify the findings.

Drug discovery research, especially metabolomics centred around agar-based bacterial samples was very challenging. Issues that require further attention are matrix interference of the bacterial lawn and sample preparation methods that would allow to circumvent such issues. Bacterial antagonistic behaviour studies are a great fit for ToF-SIMS analysis as the localisation of bacterial metabolites and the analysis of the bacterial responses is of exceptional interest. It is thus recommended to employ ToF-SIMS as a complementary technique to trial the support of such studies. However, it is believed that the limitations of the technique hinder ToF-SIMS to be used as a stand-alone analytical technique for such cases.

Finally, regarding the pre-processing method selection toolbox a number of future work areas have been identified. First of all, further pre-processing methods and regression algorithms should be added to not only allow researchers to identify the right pre-processing methods but also directly use the tool to model their data accordingly. at this point a Git-based publication should be sought after. Next, it is recommended that the toolbox be transferred to a web-based platform and integrated with the CMAC NIR data generation and modelling workflow for further testing as well as to make it more accessible to more users. Ultimately, it is recommended to use standardised data created by software, like the toolbox, in combination with a metadata model to support FAIR data (Findable, Accessible, Interoperable, Reusable) creation and establish a CMAC based spectral library for all samples.

Appendix A

Appendix Chapter 2: Toolbox Code

This code contains all contents of the graphical user interface toolbox described in chapter 3. It is written for Matlab 2018b and must be inserted into the AppDesigner to function. Updated versions of the Matlab AppDesigner might not work fully so it is advised to use Matlab2018a until an updated version is released.

A.1 Raw Code

```
classdef outlierdetectionFunctional190523 < matlab.apps.AppBase

    % Properties that correspond to app components
    properties (Access = public)
        DoEPreProcessingBenchmarkToolboxUIFigure  matlab.ui.Figure
        TabGroup                       matlab.ui.container.TabGroup
        IntroductionTab                matlab.ui.container.Tab
        DoEBasedPreProcessingOptimizationToolLabel  matlab.ui.control.Label
        Label_2                        matlab.ui.control.Label
        DataLoadingTab                 matlab.ui.container.Tab
        LoadXcalibrationButton         matlab.ui.control.Button
        DataPreviewTable               matlab.ui.control.Table
        xCalView                       matlab.ui.control.Button
        yCalView                       matlab.ui.control.Button
        xValView                       matlab.ui.control.Button
        yValView                       matlab.ui.control.Button
        xCalLamp                       matlab.ui.control.Lamp
        yCalLamp                       matlab.ui.control.Lamp
        xValLamp                       matlab.ui.control.Lamp
        yValLamp                       matlab.ui.control.Lamp
        LoadYcalibrationButton         matlab.ui.control.Button
        LoadXvalidationButton               matlab.ui.control.Button
        LoadYvalidationButton               matlab.ui.control.Button
        Label                               matlab.ui.control.Label
        InstructionsLabel                   matlab.ui.control.Label
        DevLoadButton                       matlab.ui.control.Button
        DevLoad2Button                      matlab.ui.control.Button
        DevLoad3Button                      matlab.ui.control.Button
        RunandOptionsTab                    matlab.ui.container.Tab
        FirstDerivativeSecondOrderCheckBox  matlab.ui.control.CheckBox
        FirstDerivativeFourthOrderCheckBox  matlab.ui.control.CheckBox
        SecondDerivativeSecondOrderCheckBox  matlab.ui.control.CheckBox
        SecondDerivativeFourthOrderCheckBox  matlab.ui.control.CheckBox
        BaselineCorrectionLabel             matlab.ui.control.Label
        PreProcessingMethodSelectionLabel  matlab.ui.control.Label
        SavitzkyGolaySmoothingOrder2CheckBox  matlab.ui.control.CheckBox
        SavitzkyGolaySmoothingOrder4CheckBox  matlab.ui.control.CheckBox
        SmoothingLabel                      matlab.ui.control.Label
        ParetoScalingCheckBox               matlab.ui.control.CheckBox
        AutoScalingCheckBox                 matlab.ui.control.CheckBox
        ScalingLabel                        matlab.ui.control.Label
        MultipleScatterCorrectionCheckBox  matlab.ui.control.CheckBox
        StandardNormalVariateCheckBox       matlab.ui.control.CheckBox
        ScatterCorrectionLabel              matlab.ui.control.Label
        OutlierDetectionCheckBox            matlab.ui.control.CheckBox
        CrossValidationCheckBox             matlab.ui.control.CheckBox
        RunButton                           matlab.ui.control.Button
        ResetButton                         matlab.ui.control.Button
        MaximumnumberoflatentvariablesEditFieldLabel  matlab.ui.control.Label
        MaximumnumberoflatentvariablesEditField  matlab.ui.control.NumericEditField
        BinningfactorforsmoothingandbaselinecorrectionListBoxLabel  matlab.ui.control.Label
        WidthBinning                        matlab.ui.control.ListBox
        StatusLabel                         matlab.ui.control.Label
        LabelStatus                         matlab.ui.control.Label
        OutlierDetectionPanel               matlab.ui.container.Panel
        FigureODplot                        matlab.ui.control.UIAxes
        MeanvalueEditFieldLabel             matlab.ui.control.Label
        MeanvalueEditField                  matlab.ui.control.NumericEditField
        StandarddeviationvalueEditFieldLabel  matlab.ui.control.Label
        StandarddeviationvalueEditField  matlab.ui.control.NumericEditField
        RunOD                               matlab.ui.control.Button
        NumberofLVsforoutlierdetectionEditFieldLabel  matlab.ui.control.Label
        NumberofLVsforoutlierdetectionEditField  matlab.ui.control.NumericEditField
        SelectedvaluesCheckBox              matlab.ui.control.CheckBox
        AdjustDataButton                    matlab.ui.control.Button
        CheckallCheckBox                    matlab.ui.control.CheckBox
        DataVisualisationTab                matlab.ui.container.Tab
        FigureData                          matlab.ui.control.UIAxes
        LVvsRMSEButton                      matlab.ui.control.Button
        InteractionPlotsButton              matlab.ui.control.Button
        MainEffectsPlotButton               matlab.ui.control.Button
        BestOverallResultsButton            matlab.ui.control.Button
        InteractionplotLegendPanel          matlab.ui.container.Panel
        ABaselinecorrectionLabel            matlab.ui.control.Label
        BScattercorrectionLabel             matlab.ui.control.Label
        CSmoothingLabel                     matlab.ui.control.Label
        DScalingLabel                       matlab.ui.control.Label
        ENumberoflatentvariablesLabel       matlab.ui.control.Label
        ActivateLegendCheckBox              matlab.ui.control.CheckBox
        PButton_2                          matlab.ui.control.Button
        PButton                            matlab.ui.control.Button
        PButton_3                          matlab.ui.control.Button
        ResultsTable                       matlab.ui.container.Tab
        TableResults                       matlab.ui.control.Table
        UpdateTableButton                  matlab.ui.control.Button
        ExamplesTab                        matlab.ui.container.Tab
    end


    %% Disclaimer

% Some functions used here have not been written by the author of this
% package. They were found in other working packages and distributions and
% have been referenced below.
% All foreign code has been referenced before usage.

% Savitzky Golay Derivative code

%#  AUTHOR:       Luisa Pasti
%#                Copyright(c) 1997 for ChemoAc
%#                FABI, Vrije Universiteit Brussel
%#                Laarbeeklaan 103 1090 Jette
%#                Modified program of
%#                Sijmen de Jong
%#                Unilever Research Laboratorium Vlaardingen
%#
%# VERSION: 1.1 (28/02/1998)
%#
%#  TEST:         Kris De Braekeleer


% Standard Normal Viarate Transformation

%#  AUTHOR:       Andrea Candolfi
%#                Copyright(c) 1997 for ChemoAC
%#                FABI, Vrije Universiteit Brussel
%#                Laarbeeklaan 103 1090 Jette
%#
%# VERSION: 1.1 (28/02/1998)
%#
%#  TEST:         Roy de Maesschalck



% libPLS: an integrated library for partial least squares regression and discriminant
    analysis

%#  AUTHORs:      Li H.-D., Xu Q.-S., Liang Y.-Z.
%#
%#  Li H.-D., Xu Q.-S., Liang Y.-Z. (2014) libPLS: an integrated library
%#  for partial least squares regression and discriminant analysis. Chemom. Intell. Lab. Syst
    ,
%#  2018, 176,34-43
%#
%# VERSION: 1.98 (2018)

%# All scaling functions rewritten and optimised by Michael Chrubasik 2017
%# This includes Pareto, Mean Center and Autoscaling
%#



    properties (Access = private)
    %% options window

    %enhanced options
    mc_outlier = 0;
    cross_validation = 0;

    %baseline correction
    act_dev1_ord2=0;
    act_dev1_ord4=0;
    act_dev2_ord2=0;
    act_dev2_ord4=0;

    %scatter correction
    act_MSC=0;
    act_SNV=0;

    %smoothing
    act_smooth_ord2=0;
    act_smooth_ord4=0;

    %scaling methods
    act_pareto=0;
    act_autoscale=0;

    %binning factor
    width_size=9;
    %number of latent variables
    nr_latent_variables=3;
    %number of latent variables for outlier detection
    MClv=3;

    %% hidden variables that are required

    % run multiple spectra baseline correction
    MSBC_optim = 0;

    % a preprocessing method is added automatically except for scaling
    % (see total_methods and fullfact calculation below)

    %change when additional methods are added to the program
    baseline_methods = 4;
    % set number of scatter methods
    scatter_methods = 2;
    % set number of smoothing methods
    smoothing_methods = 2;
    % set number of scaling methods
    scaling_methods = 3;

    %% running variables
    xc=NaN;
    xv=NaN;
    xc_temp=NaN;
```

```
        xv_temp=NaN;
        xc1=NaN;
        xv1=NaN;
        yc_temp=NaN;
        yv_temp=NaN;
        yc=NaN;
        yv=NaN;
        yc1=NaN;
        yv1=NaN;

        xcOc=NaN;
        xvOc=NaN;
        ycOc=NaN;
        yvOc=NaN;

        total_methods=NaN;
        DoE=NaN;
        nr_methods=NaN;
        DoE_fix=NaN;
        DoE_fixed=NaN;
        analysis=NaN;
        pls_cv_results=NaN;
        pls_cv_r2cv=NaN;
        pls_cv=NaN;
        pls_cv1=NaN;
        meaned_yc=NaN;
        meaned_yv=NaN;

        best_three = NaN;
        best_ten = NaN;
        best_ten_big = NaN;
        best_three_big = NaN;

        status = NaN;
        temp_image=NaN;

        mc_mcoutlier=NaN;
        F=NaN;
        MCmean_value=0;
        MCstd_value=0;
        end

        methods (Access = private)

            function plotmcs_appedit(app,F,threshMEAN,threshSTD,ID)

                if nargin<4;ID=[1:length(F.MEAN)]';end;
                if nargin<3;threshSTD=0;end;
                if nargin<2;threshMEAN=0;end;

                %ID=[1:length(F.MEAN)]';
                %threshSTD=0;
                %threshMEAN=0;



                MEAN=F.MEAN;
                STD=app.F.STD;
                N=length(STD);
                hold(app.FigureODplot,"on");
                h=plot(app.FigureODplot,MEAN,STD,'*');
                set(h(1),'marker','*');

                text(app.FigureODplot,MEAN*1.01,STD,num2str(ID));
                xlabel(app.FigureODplot,'MEAN');ylabel(app.FigureODplot,'STD');
            end
        end


        % Callbacks that handle component events
        methods (Access = private)

            % Value changed function: OutlierDetectionCheckBox
            function OutlierDetectionCheckBoxValueChanged(app, event)
               value = app.OutlierDetectionCheckBox.Value;
               if value
                   % if set to one MC outlier detection will be applied
                   app.mc_outlier = 1;
               else
                   app.mc_mcoutlier = 0;
               end
            end

            % Value changed function: CrossValidationCheckBox
            function CrossValidationCheckBoxValueChanged(app, event)
                value = app.CrossValidationCheckBox.Value;
               if value
                   % if set to one leave-one-out cross validation will be applied
                   app.cross_validation = 1;
               else
                   app.cross_validation = 0;
               end
            end

            % Value changed function: FirstDerivativeSecondOrderCheckBox
            function FirstDerivativeSecondOrderCheckBoxValueChanged(app, event)
                value = app.FirstDerivativeSecondOrderCheckBox.Value;
               if value
                   app.act_dev1_ord2=1;
               else
                   app.act_dev1_ord2=0;
               end
            end

            % Value changed function: FirstDerivativeFourthOrderCheckBox
            function FirstDerivativeFourthOrderCheckBoxValueChanged(app, event)
                value = app.FirstDerivativeFourthOrderCheckBox.Value;
               if value
                   app.act_dev1_ord4=1;
               else
                   app.act_dev1_ord4=0;
               end
            end

            % Value changed function:
            % SecondDerivativeSecondOrderCheckBox
            function SecondDerivativeSecondOrderCheckBoxValueChanged(app, event)
                value = app.SecondDerivativeSecondOrderCheckBox.Value;
               if value
                   app.act_dev2_ord2=1;
               else
                   app.act_dev2_ord2=0;
               end
            end

            % Value changed function:
            % SecondDerivativeFourthOrderCheckBox
            function SecondDerivativeFourthOrderCheckBoxValueChanged(app, event)
                value = app.SecondDerivativeFourthOrderCheckBox.Value;
               if value
                   app.act_dev2_ord4=1;
               else
                   app.act_dev2_ord4=0;
               end
            end

            % Value changed function:
            % SavitzkyGolaySmoothingOrder2CheckBox
            function SavitzkyGolaySmoothingOrder2CheckBoxValueChanged(app, event)
                value = app.SavitzkyGolaySmoothingOrder2CheckBox.Value;
               if value
                   app.act_smooth_ord2=1;
               else
                   app.act_smooth_ord2=0;
               end
            end

            % Value changed function:
            % SavitzkyGolaySmoothingOrder4CheckBox
            function SavitzkyGolaySmoothingOrder4CheckBoxValueChanged(app, event)
                value = app.SavitzkyGolaySmoothingOrder4CheckBox.Value;
               if value
                   app.act_smooth_ord4=1;
               else
                   app.act_smooth_ord4=0;
               end
            end

            % Value changed function: ParetoScalingCheckBox
            function ParetoScalingCheckBoxValueChanged(app, event)
                value = app.ParetoScalingCheckBox.Value;
               if value
                   app.act_pareto=1;
               else
                   app.act_pareto=0;
               end
            end

            % Value changed function: AutoScalingCheckBox
            function AutoScalingCheckBoxValueChanged(app, event)
                value = app.AutoScalingCheckBox.Value;
               if value
                   app.act_autoscale=1;
               else
```

```
                app.act_autoscale=0;
            end
        end

        % Value changed function: MultipleScatterCorrectionCheckBox
        function MultipleScatterCorrectionCheckBoxValueChanged(app, event)
             value = app.MultipleScatterCorrectionCheckBox.Value;
            if value
                app.act_MSC=1;
            else
                app.act_MSC=0;
            end
        end

        % Value changed function: StandardNormalVariateCheckBox
        function StandardNormalVariateCheckBoxValueChanged(app, event)
             value = app.StandardNormalVariateCheckBox.Value;
            if value
                app.act_SNV=1;
            else
                app.act_SNV=0;
            end
        end

        % Value changed function:
        % MaximumnumberoflatentvariablesEditField
        function MaximumnumberoflatentvariablesEditFieldValueChanged(app, event)
             value = app.MaximumnumberoflatentvariablesEditField.Value;
             app.nr_latent_variables = value;
        end

        % Button pushed function: RunButton
        function RunButtonPushed(app, event)

%%% reset all variables used for calculations to circumvent errors

    %% running variables
    %app.analysis=0;
    %app.analysis=NaN;
tic
%%% Start of function part of script
%%%
%%%
%%%
%%%

if app.mc_outlier == 0

    app.xc_temp=app.xc;
    app.yc_temp=app.yc;
    app.xv_temp=app.xv;
    app.yv_temp=app.yv;


end

if app.mc_outlier == 1

```

```
    app.xc_temp=app.xcOc;
    app.yc_temp=app.ycOc;
    app.xv_temp=app.xv;
    app.yv_temp=app.yv;


end
%% Monte Carlo Outlier Detection

%if app.mc_outlier == 1
%    run MCoutlier_newDoE_p1.m
%end

%% Mutliple Spectra Baseline Correction Optimizer

%if app.MSBC_optim == 1
%    if app.baseline_methods == 5
%        run msbc_optim.m
%    end
%end

%% do not change these variables

% temp x_calibration (xc) and x_validation (xv)
xc1 = app.xc_temp;
xv1 = app.xv_temp;

% temp v_calibration (vc) and v_validation (vv)
yc1 = app.yc_temp;
yv1 = app.yv_temp;

debug_test = 0;
%% DoE Matrix Creation

% calculates the number of total methods to be calculated (as DoE matrix
% starts from 0 and the zero(th) method is reserved for no preprocessing a
% (+1) is added to all methods.
app.total_methods = ((app.baseline_methods+1) * (app.scatter_methods+1) * (app.
    smoothing_methods+1) * (app.scaling_methods));

% builds a full factorial DoE matrix based on the input numbers for each
% method
app.DoE = fullfact([app.baseline_methods+1 app.scatter_methods+1 app.smoothing_methods+1 app.
    scaling_methods app.nr_latent_variables+1]);
app.DoE = app.DoE -1;
app.DoE = app.DoE((app.total_methods+1):end,:);

app.nr_methods = size(app.DoE,1)/app.nr_latent_variables;
analysis1.methods_applied = cell(app.total_methods,5);

%%% need to keep analysis conversion appified
app.analysis=analysis1;

%% Count number of preprocessing methods applied per row

methods_applied = zeros(app.nr_methods,1);

app.DoE_fix = repmat([0, 0, 0, 1,0],size(app.DoE,1),1);
app.DoE_fixed = app.DoE+app.DoE_fix;

for hh = 1:app.nr_methods
    for ii = 1:4
        if app.DoE_fixed(hh,ii) > 0
            methods_applied(hh,1) = methods_applied(hh,1) + 1;
            app.analysis.methods_applied{hh,5} = methods_applied(hh,1);
        end
    end
end

%% Read in DoE factorial design and apply corrections
% if further preprocessing steps should be applied to the data, simply
% add them to the respective method and increase the numbering

%%%         This addtional code would add a 5th baseline correction step
%%%         as column 1 from DoE has been chosen, equalled to 5 and thus
%%%         whenever a 5 appears in the DoE variable in column 1, this
%%%         method will be called.
%%%
%%%         -------------------------------------------------------------
%%%         elseif DoE(ii,1) == 5 % Number 5: example;
%%%         analysis.methods_applied{ii, 1} = 'example';
%%%         % example code what to do with calibration and validation data
%%%         xc1 = do_something(xc1);
%%%         xv1 = do_something(xv1);
%%%         -------------------------------------------------------------
%%%

% create structures to hold RMSEV and RMSEC data

if (size(yc1,2)) == 1

    app.analysis.RMSEC = zeros(app.nr_methods, app.nr_latent_variables);
    app.analysis.RMSEV = zeros(app.nr_methods, app.nr_latent_variables);

end

if (size(yc1,2)) ~= 1

    app.analysis.RMSEC = cell(app.nr_methods, app.nr_latent_variables);
    app.analysis.RMSEV = cell(app.nr_methods, app.nr_latent_variables);

end

for ii = 1:app.nr_methods

% resets variables after each run
% x_calibration (xc) and x_validation (xv)
    xc1 = app.xc_temp;
    xv1 = app.xv_temp;

% v_calibration (vc) and v_validation (vv)
    yc1 = app.yc_temp;
    yv1 = app.yv_temp;

    %% Baseline Correction Methods

    % debug code to show if scatter correction applied

    % WIP
    % to extend code allow singular method selection: this will be achieved
    %   by adding an extra check whether the method was activated
    %   specifically. If so, do the correction, if not, do nothing and move
    %   on. At the end, delete all "none" edited methods and dont count.
    %   NOT YET IMPLEMENTED

    if app.DoE(ii,1) == 0
        app.analysis.methods_applied{ii, 1} = 'no baseline';
    end

    if app.DoE(ii,1) ~= 0 % Number 0: none

        if app.DoE(ii,1) == 1 && app.act_dev1_ord2==1% Number 1: dev1_ord2_9pt;
        app.analysis.methods_applied{ii, 1} = 'dev1_ord2_9pt';
        xc1 = deriv(xc1,1,app.width_size,2);
        xv1 = deriv(xv1,1,app.width_size,2);

        elseif app.DoE(ii,1) == 2 && app.act_dev1_ord4==1% Number 2: dev1_ord4_9pt;
        app.analysis.methods_applied{ii, 1} = 'dev1_ord4_9pt';
        xc1 = deriv(xc1,1,app.width_size,4);
        xv1 = deriv(xv1,1,app.width_size,4);

        elseif app.DoE(ii,1) == 3 && app.act_dev2_ord2==1 % Number 3: dev2_ord2_9pt;
        app.analysis.methods_applied{ii, 1} = 'dev2_ord2_9pt';
        xc1 = deriv(xc1,2,app.width_size,2);
        xv1 = deriv(xv1,2,app.width_size,2);

        elseif app.DoE(ii,1) == 4 && app.act_dev2_ord4==1% Number 4: dev2_ord4_9pt;
        app.analysis.methods_applied{ii, 1} = 'dev2_ord4_9pt';
        xc1 = deriv(xc1,2,app.width_size,4);
        xv1 = deriv(xv1,2,app.width_size,4);

        elseif app.DoE(ii,1) == 5 % Number 5: asymmetric least squares, second derivative, 0.001 smoothening;
        app.analysis.methods_applied{ii, 1} = 'multi_spec_base_corr';
        [zc,bgxc] = MSBC(xc1,MSBC_doe_val(best_MSBC,1),MSBC_doe_val(best_MSBC,2),MSBC_doe_val(best_MSBC,3));
        xc1 = xc1-zc;
        [zv,bgxv] = MSBC(xv1,MSBC_doe_val(best_MSBC,1),MSBC_doe_val(best_MSBC,2),MSBC_doe_val(best_MSBC,3));
        xv1 = xv1-zv;
        end


    end

    %% Scatter Correction Methods

    % debug code to show if scatter correction applied

    if app.DoE(ii,2) == 0
        app.analysis.methods_applied{ii, 2} = 'no scatter';
    end

    if app.DoE(ii,2) ~= 0 % Number 0: none
```

```

        if app.DoE(ii,2) == 1 && app.act_MSC==1% Number 1: MSC (mean)
        app.analysis.methods_applied{ii, 2} = 'MSC'     ;
        [xc1, alpha1, beta1, xv1] = MultipleScatterCorrectionMulti(xc1,xv1);

        elseif app.DoE(ii,2) == 2 && app.act_SNV==1% Number 2: SNV
        app.analysis.methods_applied{ii, 2} = 'SNV';
        [mc,nc]=size(xc1);
        xc1=(xc1-mean(xc1')'*ones(1,nc))./(std(xc1')'*ones(1,nc));
        [mc,nc]=size(xv1);
        xv1=(xv1-mean(xv1')'*ones(1,nc))./(std(xv1')'*ones(1,nc));

        end

    end

    %% Smoothing Methods

    % debug code to show if smoothing applied

    if app.DoE(ii,3) == 0
        app.analysis.methods_applied{ii, 3} = 'no smoothing';
    end

    if app.DoE(ii,3) ~= 0 % Number 0: none

        if app.DoE(ii,3) == 1 && app.act_smooth_ord2==1% Number 1: SavGol smooth_ord2_9pt
        app.analysis.methods_applied{ii, 3} = 'smooth_ord2' ;
        xc1 = sgolayfilt(xc1,2,app.width_size);
        xv1 = sgolayfilt(xv1,4,app.width_size);

        elseif app.DoE(ii,3) == 2 && app.act_smooth_ord4==1% Number 2: SavGol smooth_ord4_9pt
        app.analysis.methods_applied{ii, 3} = 'smooth_ord4'    ;
        xc1 = sgolayfilt(xc1,4,app.width_size);
        xv1 = sgolayfilt(xv1,4,app.width_size);

        end

    end


    %% Scaling Methods

    % debug code to show if scaling applied

    if app.DoE(ii,4) == 0

        if app.DoE(ii,2) ~=1 % no MSC in DoE, mean center
        app.analysis.methods_applied{ii, 4} = 'mean center';
        [xc1, xv1] = mean_center_multi(xc1, xv1);
        [yc1, yv1] = mean_center_multi(yc1, yv1);

        elseif app.DoE(ii,2) ~=0 && app.act_MSC==1 % MSC in DoE, do not mean center
        app.analysis.methods_applied{ii, 4} = 'MSC mean';

        elseif app.DoE(ii,2) ~=0 && app.act_MSC==0% MSC in DoE, but deactivated, mean center
        app.analysis.methods_applied{ii, 4} = 'mean center';
        [xc1, xv1] = mean_center_multi(xc1, xv1);
        [yc1, yv1] = mean_center_multi(yc1, yv1);

        end

    end


    if app.DoE(ii,4) ~= 0 % Number 0: no_scaling

        if app.DoE(ii,4) == 1 && app.act_pareto==1 && app.act_MSC==0% Number 1: Pareto mean center
        app.analysis.methods_applied{ii, 4} = 'mean center and pareto' ;
        [xc1, xv1] = mean_center_multi(xc1, xv1);
        [yc1, yv1] = mean_center_multi(yc1, yv1);
        [xc1, xv1] = pareto_scaling(xc1, xv1);
        [yc1, yv1] = pareto_scaling(yc1, yv1);

        elseif app.DoE(ii,4) == 1 && app.act_pareto==1 && app.act_MSC==1% Number 1: Pareto MSC
        app.analysis.methods_applied{ii, 4} = 'mean center and pareto' ;
        [xc1, xv1] = pareto_scaling(xc1, xv1);
        [yc1, yv1] = pareto_scaling(yc1, yv1);

        elseif app.DoE(ii,4) == 2 && app.act_autoscale==1 && app.act_MSC==0% Number 2: Autoscale mean center
        app.analysis.methods_applied{ii, 4} = 'mean center and autoscale';
        [xc1, xv1] = mean_center_multi(xc1, xv1);
        [yc1, yv1] = mean_center_multi(yc1, yv1);
        [xc1, xv1] = auto_scaling(xc1, xv1) ;
        [yc1, yv1] = auto_scaling(yc1, yv1);

        elseif app.DoE(ii,4) == 2 && app.act_autoscale==1 && app.act_MSC==1 % Number 2: Autoscale MSC
        app.analysis.methods_applied{ii, 4} = 'mean center and autoscale';
        [xc1, xv1] = auto_scaling(xc1, xv1) ;
        [yc1, yv1] = auto_scaling(yc1, yv1);

        end

    end


    app.analysis.calibration_data{ii,1} = xc1;
    app.analysis.validation_data{ii,1} = xv1;
    app.analysis.calibration_data{ii,2} = yc1;
    app.analysis.validation_data{ii,2} = yv1;


end


%% PLS Regression Model Building for each method saving all values

for ii = 1:app.nr_methods
    for jj = 1:app.nr_latent_variables

        [XL,yL,XS,YS,beta,PCTVAR] = plsregress(app.analysis.calibration_data{ii,1},app.analysis.calibration_data{ii,2},jj);

        app.analysis.XL{ii,jj} = XL;
        app.analysis.yL{ii,jj} = yL;
        app.analysis.XS{ii,jj} = XS;
        app.analysis.YS{ii,jj} = YS;
        app.analysis.beta{ii,jj} = beta;
        app.analysis.PCTVAR{ii,jj} = PCTVAR;

    end
end

%% PLS Regression Model Application to Validation Data

for ii = 1:app.nr_methods
    for jj = 1:app.nr_latent_variables

    yfit_val = [ones(size(app.analysis.validation_data{ii,1},1),1) app.analysis.validation_data{ii,1}]*app.analysis.beta{ii,jj};
    yfit_cal = [ones(size(app.analysis.calibration_data{ii,1},1),1) app.analysis.calibration_data{ii,1}]*app.analysis.beta{ii,jj};

        if (size(yc1,2)) == 1

            app.analysis.RMSEV(ii,jj) = sqrt(mean((yfit_val - app.analysis.validation_data{ii,2}).^2));
            app.analysis.RMSEC(ii,jj) = sqrt(mean((yfit_cal - app.analysis.calibration_data{ii,2}).^2));

        end

        if (size(yc1,2)) ~= 1

            app.analysis.RMSEV{ii,jj} = sqrt(mean((yfit_val - app.analysis.validation_data{ii,2}).^2));
            app.analysis.RMSEC{ii,jj} = sqrt(mean((yfit_cal - app.analysis.calibration_data{ii,2}).^2));

        end

    yc1 = app.analysis.calibration_data{ii,2};
    yv1 = app.analysis.validation_data{ii,2};
    app.meaned_yc = mean(yc1);
    app.meaned_yv = mean(yv1);

        if (size(yc1,2)) == 1

            app.analysis.SSTc(ii,jj)=sum((yc1-app.meaned_yc).^2);
            app.analysis.SSRc(ii,jj)=sum((yfit_cal-app.meaned_yc).^2);
            app.analysis.SSEc(ii,jj)=sum((yc1-yfit_cal).^2);
            app.analysis.R2c(ii,jj)=1-app.analysis.SSEc(ii,jj)/app.analysis.SSTc(ii,jj);

            app.analysis.SSTv(ii,jj)=sum((yv1-app.meaned_yv).^2);
            app.analysis.SSRv(ii,jj)=sum((yfit_val-app.meaned_yv).^2);
            app.analysis.SSEv(ii,jj)=sum((yv1-yfit_val).^2);
            app.analysis.R2v(ii,jj)=1-app.analysis.SSEv(ii,jj)/app.analysis.SSTv(ii,jj);

        end

        if (size(yc1,2)) ~= 1

            app.analysis.SSTc{ii,jj}=sum((yc1-app.meaned_yc).^2);
            app.analysis.SSRc{ii,jj}=sum((yfit_cal-app.meaned_yc).^2);
            app.analysis.SSEc{ii,jj}=sum((yc1-yfit_cal).^2);
            app.analysis.R2c{ii,jj}=1-app.analysis.SSEc{ii,jj}/app.analysis.SSTc{ii,jj};

            app.analysis.SSTv{ii,jj}=sum((yv1-app.meaned_yv).^2);
            app.analysis.SSRv{ii,jj}=sum((yfit_val-app.meaned_yv).^2);
            app.analysis.SSEv{ii,jj}=sum((yv1-yfit_val).^2);
            app.analysis.R2v{ii,jj}=1-app.analysis.SSEv{ii,jj}/app.analysis.SSTv{ii,jj};

        end


    %analysis.SSTc(ii,jj)=sum((yc-meaned_yc).^2);
    %analysis.SSRc(ii,jj)=sum((yfit_cal-meaned_yc).^2);
    %analysis.SSEc(ii,jj)=sum((yc-yfit_cal).^2);
    %analysis.R2c(ii,jj)=1-analysis.SSEc(ii,jj)/analysis.SSTc(ii,jj);

    %analysis.SSTv(ii,jj)=sum((yv-meaned_yv).^2);
    %analysis.SSRv(ii,jj)=sum((yfit_val-meaned_yv).^2);
    %analysis.SSEv(ii,jj)=sum((yv-yfit_val).^2);
    %analysis.R2v(ii,jj)=1-analysis.SSEv(ii,jj)/analysis.SSTv(ii,jj);

    end
end


%% External Cross-Validation Code if CV wanted

if app.cross_validation == 1
    for ii = 1:app.nr_methods
        app.pls_cv1{ii,1} = plscv(app.analysis.calibration_data{ii,1},app.analysis.
             calibration_data{ii,2},app.nr_latent_variables);
    end

        if (size(yc1,2)) == 1

            for ii = 1:app.nr_methods
                pls_cv_results1(ii,:) = app.pls_cv1{ii,1}.RMSEcv;
                pls_cv_r2cv1(ii,:) = app.pls_cv1{ii,1}.R2cv;
            end

        end

        if (size(yc1,2)) ~= 1

            for ii = 1:app.nr_methods
                pls_cv_results1{ii,:} = app.pls_cv1{ii,1}.RMSEcv;
                pls_cv_r2cv1{ii,:} = papp.ls_cv1{ii,1}.R2cv;
            end

        end
    app.pls_cv=app.pls_cv1;
    app.pls_cv_results = pls_cv_results1;
    app.pls_cv_r2cv = pls_cv_r2cv1;
    %for ii = 1:nr_methods
    %    pls_cv_results(ii,:) = pls_cv{ii,1}.RMSEcv;
    %    pls_cv_r2cv(ii,:) = pls_cv{ii,1}.R2cv;
    %end
    clear analysis_temp_cv;
end

app.analysis.RMSECV = app.pls_cv_results;
app.analysis.R2cv = app.pls_cv_r2cv;
%analysis2 = analysis;
%app.analysis = analysis2;
%clear analysis2;
%% Clearing not needed variables

toc

%%% This part of the script is responsible for making the results data table and making the results available for plotting
%%
%%

for i = 1:app.nr_latent_variables
[app.analysis.bestMethod.val(:,i),app.analysis.bestMethod.nr(:,i)]  = sort(app.analysis.RMSEV(:,i),1,'ascend');
end

% best three methods per component matrix including 4th column with zero
% preprocessing methods applied (except mean centering)

app.best_three_big = zeros(app.nr_latent_variables,3);
app.best_three_big(:,4) = 1;
app.best_ten = zeros(app.nr_latent_variables,10);

Nr = [];RMSEV = [];LV = [];RMSECV = [];R2cv = [];R2v = [];Baseline = [];
Scatter = [];Smooth = [];Scaling = [];Methods_Used = [];

for ii = 1:app.nr_latent_variables
    %disp(['Preprocessing methods used and RMSEV for best three methods at ' num2str(ii)  ' LVs'])
    %disp(['and value for only mean centering.'])
    app.best_three = app.analysis.bestMethod.nr(1:3,ii);
    app.best_ten_big(ii,1:10) = app.analysis.bestMethod.nr(1:10,ii);
    app.best_three_big(ii,1:3) = app.best_three;
    %Preprocessing_Methods_Used = analysis.methods_applied(best_three,:);
    %modelStruct.bestMethod.val(1:3,i)
    app.analysis.RMSEV((app.best_three_big(ii,:)),ii);

    Nr = [Nr;(app.best_three_big(ii,:)')];
    RMSEV = [RMSEV;app.analysis.RMSEV((app.best_three_big(ii,:)),ii)];
    LV = [LV;[ii;ii;ii;ii]];
    R2v = [R2v;app.analysis.R2v((app.best_three_big(ii,:)),ii)];
    Methods_Used = [Methods_Used;app.analysis.methods_applied(app.best_three,5);1];
    Baseline = [Baseline;app.analysis.methods_applied(app.best_three,1);'no baseline'];
    Scatter = [Scatter;app.analysis.methods_applied(app.best_three,2);'no scatter'];
    Smooth = [Smooth;app.analysis.methods_applied(app.best_three,3);'no smoothing'];
    Scaling = [Scaling;app.analysis.methods_applied(app.best_three,4);'no scaling'];

    if app.cross_validation == 1
        R2cv = [R2cv;app.analysis.R2cv((app.best_three_big(ii,:)),ii)];
        RMSECV = [RMSECV;app.analysis.RMSECV((app.best_three_big(ii,:)),ii)];
    else
        R2cv = NaN;
        RMSECV = NaN;
    end

end

    if app.cross_validation == 1
        app.analysis.best_results_perLV = table(Nr, RMSEV, RMSECV, R2v, R2cv, LV,Baseline,Scatter,Smooth,Scaling);

        % taken out Methods_Used due to wrong way of calculating the number of methods used
        %app.analysis.best_results_perLV = table(Nr, RMSEV, RMSECV, R2v, R2cv, LV, Methods_Used,Baseline,Scatter,Smooth,Scaling);
    else
        app.analysis.best_results_perLV = table(Nr, RMSEV, R2v, LV,Baseline,Scatter,Smooth,Scaling);
    end

    %%%%%%
    %%%%%%
    %%%%%%

    %%%%%%

        end

        % Button pushed function: LoadXcalibrationButton
        function LoadXcalibrationButtonPushed(app, event)
            [file,path] = uigetfile;
            if isequal(file,0)
                disp('User selected Cancel')
            else
            disp(['User selected ', fullfile(path, file)])
            app.xc = xlsread(fullfile(path,file))
            end
            %file = strcat(path,file)

            if isnan(app.xc)
                app.xCalLamp.Color ="red";
            else
                app.xCalLamp.Color ="green";
            end
        end

        % Button pushed function: LoadYcalibrationButton
        function LoadYcalibrationButtonPushed(app, event)
            [file,path] = uigetfile
            file = strcat(path,file)
            app.yc = xlsread(file)
            if isnan(app.yc)
                app.yCalLamp.Color ="red";
            else
                app.yCalLamp.Color ="green";
            end
        end

        % Button pushed function: LoadXvalidationButton
        function LoadXvalidationButtonPushed(app, event)
            [file,path] = uigetfile
            file = strcat(path,file)
            app.xv = xlsread(file)
            if isnan(app.xv)
                app.xValLamp.Color ="red";
            else
                app.xValLamp.Color ="green";
            end
        end

        % Button pushed function: LoadYvalidationButton
        function LoadYvalidationButtonPushed(app, event)
            [file,path] = uigetfile
            file = strcat(path,file)
            app.yv = xlsread(file)
            if isnan(app.yv)
                app.yValLamp.Color ="red";
            else
                app.yValLamp.Color ="green";
            end
        end

        % Button pushed function: xCalView
        function xCalViewButtonPushed(app, event)
            app.DataPreviewTable.Data = app.xc;
        end

        % Button pushed function: yCalView
        function yCalViewButtonPushed(app, event)
            app.DataPreviewTable.Data = app.yc;
        end

        % Button pushed function: xValView
        function xValViewButtonPushed(app, event)
            app.DataPreviewTable.Data = app.xv;
        end

        % Button pushed function: yValView
        function yValViewButtonPushed(app, event)
            app.DataPreviewTable.Data = app.yv;
        end

        % Button pushed function: DevLoadButton
        function DevLoadButtonPushed(app, event)
            app.xc = xlsread("C:\Users\micha\OneDrive - University of Strathclyde\Thesis\
                Chemometric\PLSToolboxWorkMichael\pix103x100Set1.xlsx")
            app.yc = xlsread("C:\Users\micha\OneDrive - University of Strathclyde\Thesis\
                Chemometric\PLSToolboxWorkMichael\concSet1.xlsx")
            app.xv = xlsread("C:\Users\micha\OneDrive - University of Strathclyde\Thesis\
                Chemometric\PLSToolboxWorkMichael\pix103x100Set3b.xlsx")
            app.yv = xlsread("C:\Users\micha\OneDrive - University of Strathclyde\Thesis\
                Chemometric\PLSToolboxWorkMichael\concSet3b.xlsx")
        end

        % Button pushed function: UpdateTableButton
        function UpdateTableButtonPushed(app, event)
```

```
            clear app.TableResults

            if app.cross_validation == 1
                % Create TableResults
                app.TableResults = uitable(app.ResultsTable);
                app.TableResults.ColumnName = {"Nr";"RMSEV";"RMSECV";"R2v";"R2cv";"LV";"Baseline";"Scatter";"Smooth";"Scaling"};
                app.TableResults.RowName = {};
                app.TableResults.Position = [14 50 1294 846];
            else
                 % Create TableResults
                app.TableResults = uitable(app.ResultsTable);
                app.TableResults.ColumnName = {"Nr";"RMSEV";"R2v";"LV";"Baseline";"Scatter";"Smooth";"Scaling"};
                app.TableResults.RowName = {};
                app.TableResults.Position = [14 50 1294 846];
            end
            app.TableResults.Data = app.analysis.best_results_perLV;
        end

        % Value changed function: WidthBinning
        function WidthBinningValueChanged(app, event)
           value = app.WidthBinning.Value;
           app.width_size = str2num(value);
        end

        % Button pushed function: LVvsRMSEButton
        function LVvsRMSEButtonPushed(app, event)

%% plot latent variables vs RMSEC for all methods

            clear app.FigureData
            app.FigureData = uiaxes(app.DataVisualisationTab);
            app.FigureData.Position = [9 88 1284 810];

hold(app.FigureData, "on")
%for ii = 1:nr_methods
plot(app.FigureData,[1:app.nr_latent_variables], app.analysis.RMSEV(unique(app.best_ten_big(:,1)),:))
legend(app.FigureData,int2str(unique(app.best_ten_big(:,1))))
xlim(app.FigureData,[0.8 app.nr_latent_variables+0.2]);
%savefig(valplot.fig)
%end
hold(app.FigureData, "off")
title(app.FigureData,'Latent Variables Vs Model RMSEC (primary selection)','FontSize',10)
xlabel(app.FigureData,'Latent Variables');
ylabel(app.FigureData,'Model RMSEC');


        end

        % Button pushed function: InteractionPlotsButton
        function InteractionPlotsButtonPushed(app, event)

            clear app.FigureData
            app.FigureData = uiaxes(app.DataVisualisationTab);
            app.FigureData.Position = [9 88 1284 810];

```

```
interactplot = zeros(app.nr_methods*app.nr_latent_variables,1);
%interactplot(1:nrMethods,1) = modelStruct.RMSEV(:,1);

for i = 1:app.nr_latent_variables
    interactplot(app.nr_methods*(i-1)+1:app.nr_methods*i,1) = app.analysis.RMSEV(:,i);
end

h1=figure('visible','off');
set(h1,'Position',[9 52 900 600])
%interactionplot(interactplot(:,1), dFF) %'varnames', group_names
%f = figure('visible','off');
interactionplot(interactplot(:,1), app.DoE,'varnames', {'A','B','C','D','E'})
%title('Interaction Plot')

saveas(h1,"temp_image.png")


% Remove title, axis labels, and tick labels
title(app.FigureData, []);
xlabel(app.FigureData, []);
ylabel(app.FigureData, []);
app.FigureData.XAxis.TickLabels = {};
app.FigureData.YAxis.TickLabels = {};
% Display image and stretch to fill axes
I=imshow("temp_image.png", 'parent', app.FigureData,'XData',[1 app.FigureData.Position(3)],'YData',[1 app.FigureData.Position(4)]);
% Set limits of axes
app.FigureData.XLim = [0 I.XData(2)];
app.FigureData.YLim = [0 I.YData(2)];


        end

        % Button pushed function: PButton_2
        function PButton_2Pushed(app, event)

interactplot = zeros(app.nr_methods*app.nr_latent_variables,1);
%interactplot(1:nrMethods,1) = modelStruct.RMSEV(:,1);

for i = 1:app.nr_latent_variables
    interactplot(app.nr_methods*(i-1)+1:app.nr_methods*i,1) = app.analysis.RMSEV(:,i);
end

%interactionplot(interactplot(:,1), dFF) %'varnames', group_names
figure('visible','on');
interactionplot(interactplot(:,1), app.DoE,'varnames', {'A','B','C','D','E'})
%title('Interaction Plot')


        end

        % Value changed function: ActivateLegendCheckBox
        function ActivateLegendCheckBoxValueChanged(app, event)
            value = app.ActivateLegendCheckBox.Value;
           if value
               app.InteractionplotLegendPanel.Visible = "on";
           else
               app.InteractionplotLegendPanel.Visible = "off";
            end
        end

        % Button pushed function: MainEffectsPlotButton
        function MainEffectsPlotButtonPushed(app, event)

            clear app.FigureData
            app.FigureData = uiaxes(app.DataVisualisationTab);
            app.FigureData.Position = [9 88 1284 810];

interactplot = zeros(app.nr_methods*app.nr_latent_variables,1);
%interactplot(1:nrMethods,1) = modelStruct.RMSEV(:,1);

for i = 1:app.nr_latent_variables
    interactplot(app.nr_methods*(i-1)+1:app.nr_methods*i,1) = app.analysis.RMSEV(:,i);
end

h1=figure('visible','off');
set(h1,'Position',[9 52 900 600])
%interactionplot(interactplot(:,1), dFF) %'varnames', group_names
%f = figure('visible','off');
maineffectsplot(interactplot(:,1), app.DoE,'varnames', {'Baseline','Scatter','Smooth','Scale','Components'})
%title('Interaction Plot')

saveas(h1,"temp_image.png")


% Remove title, axis labels, and tick labels
title(app.FigureData, []);
xlabel(app.FigureData, []);
ylabel(app.FigureData, []);
app.FigureData.XAxis.TickLabels = {};
app.FigureData.YAxis.TickLabels = {};
% Display image and stretch to fill axes
I=imshow("temp_image.png", 'parent', app.FigureData,'XData',[1 app.FigureData.Position(3)],'YData',[1 app.FigureData.Position(4)]);
% Set limits of axes
app.FigureData.XLim = [0 I.XData(2)];
app.FigureData.YLim = [0 I.YData(2)];

        end

        % Button pushed function: PButton
        function PButtonPushed(app, event)
interactplot = zeros(app.nr_methods*app.nr_latent_variables,1);
%interactplot(1:nrMethods,1) = modelStruct.RMSEV(:,1);

for i = 1:app.nr_latent_variables
    interactplot(app.nr_methods*(i-1)+1:app.nr_methods*i,1) = app.analysis.RMSEV(:,i);
end

%interactionplot(interactplot(:,1), dFF) %'varnames', group_names
figure('visible','on');
maineffectsplot(interactplot(:,1), app.DoE,'varnames', {'Baseline','Scatter','Smooth','Scale','Components'})
%title('Interaction Plot')
        end

        % Button pushed function: BestOverallResultsButton
        function BestOverallResultsButtonPushed(app, event)

            clear app.FigureData
            app.FigureData = uiaxes(app.DataVisualisationTab);
            app.FigureData.Position = [9 88 1284 810];

%% using gscatter, new version, using best three and one processing method

% builds quick structure to draw best three and zero processing methods
% values

val_best3_zero = zeros(4, app.nr_latent_variables);
val_best3_zero = app.analysis.bestMethod.val(1:3,1:app.nr_latent_variables);
val_best3_zero(4,:) = app.analysis.RMSEV(1,1:app.nr_latent_variables);


% hold and for loop required to put all 4 graphs into single plot:
% first three are the best methods, 4 is the zero processing methods value

index_scatter = zeros(app.nr_methods,app.nr_latent_variables);

index_scatter(1,:) = 1;
for ii = 1:app.nr_latent_variables
    for jj = 1:3
        if app.analysis.methods_applied{(app.best_three_big(ii,jj)'),5}(1) == 1
            index_scatter((app.best_three_big(ii,jj)'),ii) = 1;
        elseif app.analysis.methods_applied{(app.best_three_big(ii,jj)'),5}(1) == 2
            index_scatter((app.best_three_big(ii,jj)'),ii) = 2;
        elseif app.analysis.methods_applied{(app.best_three_big(ii,jj)'),5}(1) == 3
            index_scatter((app.best_three_big(ii,jj)'),ii) = 3;
        elseif app.analysis.methods_applied{(app.best_three_big(ii,jj)'),5}(1) == 4
            index_scatter((app.best_three_big(ii,jj)'),ii) = 4          ;
        end
    end
end

index_latent = repmat([1:app.nr_latent_variables],app.nr_methods,1);

%colormap1 = [0.8000,    0.8000,    0.8000];

colormap2 = [0.8000,    0.8000,    0.8000
            0.0000,    0.0000,    0.0000];

colormap3 = [0.6510,    0.8078,    0.8902
             0.1216,    0.4706,    0.7059
             0.6980,    0.8745,    0.5412
             0.2000,    0.6275,    0.1725];

h1=figure('visible','off');
set(h1,'Position',[9 52 900 600])

hold on
for ii= 1:app.nr_latent_variables
    gscatter(index_latent(:,ii), app.analysis.RMSEV(:,ii),index_scatter(:,ii),colormap3,'
        ......',[0.1 0.1 18 18 18 18],'off');
    gscatter(index_latent(:,ii), app.analysis.RMSEV(:,ii),index_scatter(:,ii),colormap2,'+.....',[0.1 12 0.1 0.1 0.1 0.1 0.1],'off');
end


%for ii= 1:nr_latent_variables
%     gscatter(index_latent(:,ii), analysis.RMSEV(:,ii),index_scatter(:,ii),colormap3,'......',[0.1 0.1 18 18 18 18],'off');
%end

for ii = 1:app.nr_latent_variables
    for jj = 1:4
        text(ii+0.05, val_best3_zero(jj,ii), num2str(app.best_three_big(ii,jj)), 'FontSize', 9)
    end
end

title('Latent Variables Vs Model RMSEV based on Preprocessing Methods used')
xlabel('Latent Variables');
ylabel('Model RMSEV');
xlim([0 app.nr_latent_variables+1]);
text(0.90,0.85, {'Preprocessing','Methods used','{\bf \fontsize{15} \color[rgb]{0.6510, 0.8078, 0.8902} — } 1', '{\bf \fontsize{15} \color[rgb]{0.1216, 0.4706, 0.7059} — } 2', '{\bf \fontsize{15} \color[rgb]{0.6980,0.8745,0.5412} — } 3','{\bf \fontsize{15} \color[rgb]{0.2000,0.6275,0.1725} — } 4'},'HorizontalAlignment','center', 'EdgeColor', 'k','units','normalized');

hold off
saveas(h1,"temp_image.png")

% Remove title, axis labels, and tick labels
title(app.FigureData, []);
xlabel(app.FigureData, []);
ylabel(app.FigureData, []);
app.FigureData.XAxis.TickLabels = {};
app.FigureData.YAxis.TickLabels = {};
% Display image and stretch to fill axes
I=imshow("temp_image.png", 'parent', app.FigureData,'XData',[1 app.FigureData.Position(3)],'YData',[1 app.FigureData.Position(4)]);
% Set limits of axes
app.FigureData.XLim = [0 I.XData(2)];
app.FigureData.YLim = [0 I.YData(2)];

        end

        % Button pushed function: PButton_3
        function PButton_3Pushed(app, event)


val_best3_zero = zeros(4, app.nr_latent_variables);
val_best3_zero = app.analysis.bestMethod.val(1:3,1:app.nr_latent_variables);
val_best3_zero(4,:) = app.analysis.RMSEV(1,1:app.nr_latent_variables);


% hold and for loop required to put all 4 graphs into single plot:
% first three are the best methods, 4 is the zero processing methods value

index_scatter = zeros(app.nr_methods,app.nr_latent_variables);
```

```

index_scatter(1,:) = 1;
for ii = 1:app.nr_latent_variables
    for jj = 1:3
        if app.analysis.methods_applied{(app.best_three_big(ii,jj)'),5}(1) == 1
            index_scatter((app.best_three_big(ii,jj)'),ii) = 1;
        elseif app.analysis.methods_applied{(app.best_three_big(ii,jj)'),5}(1) == 2
            index_scatter((app.best_three_big(ii,jj)'),ii) = 2;
        elseif app.analysis.methods_applied{(app.best_three_big(ii,jj)'),5}(1) == 3
            index_scatter((app.best_three_big(ii,jj)'),ii) = 3;
        elseif app.analysis.methods_applied{(app.best_three_big(ii,jj)'),5}(1) == 4
            index_scatter((app.best_three_big(ii,jj)'),ii) = 4           ;
        end
    end
end

index_latent = repmat([1:app.nr_latent_variables],app.nr_methods,1);

%colormap1 = [0.8000,      0.8000,      0.8000];

colormap2 = [0.8000,      0.8000,      0.8000
             0.0000,      0.0000,      0.0000];

colormap3 = [0.6510,      0.8078,      0.8902
              0.1216,      0.4706,      0.7059
              0.6980,      0.8745,      0.5412
              0.2000,      0.6275,      0.1725];

figure('visible','on');
hold on
for ii= 1:app.nr_latent_variables
    gscatter(index_latent(:,ii), app.analysis.RMSEV(:,ii),index_scatter(:,ii),colormap3,'......',[0.1 0.1 18 18 18 18],'off');
    gscatter(index_latent(:,ii), app.analysis.RMSEV(:,ii),index_scatter(:,ii),colormap2,'+.....',[0.1 12 0.1 0.1 0.1 0.1 0.1],'off');
end


%for ii= 1:nr_latent_variables
%    gscatter(index_latent(:,ii), analysis.RMSEV(:,ii),index_scatter(:,ii),colormap3,'......',[0.1 0.1 18 18 18 18],'off');
%end

for ii = 1:app.nr_latent_variables
    for jj = 1:4
        text(ii+0.05, val_best3_zero(jj,ii), num2str(app.best_three_big(ii,jj)), 'FontSize', 9)
    end
end

title('Latent Variables Vs Model RMSEV based on Preprocessing Methods used')
xlabel('Latent Variables');
ylabel('Model RMSEV');
xlim([0 app.nr_latent_variables+1]);
text(0.90,0.85, {'Preprocessing','Methods used','{\bf \fontsize{15} \color[rgb]{0.6510, 0.8078, 0.8902} -- } 1', '{\bf \fontsize{15} \color[rgb]{0.1216, 0.4706, 0.7059} -- } 2', '{\bf \fontsize{15} \color[rgb]{0.6980,0.8745,0.5412} -- } 3','{\bf \fontsize{15} \color[rgb]{0.2000,0.6275,0.1725} -- } 4'},'HorizontalAlignment','center', 'EdgeColor', 'k','
units','normalized');

hold off


        end

        % Button pushed function: RunOD
        function RunODPushed(app, event)

app.FigureODplot = uiaxes(app.OutlierDetectionPanel);
title(app.FigureODplot, 'Monte-Carlo based outlier detection')
xlabel(app.FigureODplot, 'Mean')
ylabel(app.FigureODplot, 'Std')
app.FigureODplot.Position = [16 61 677 454];

% pretreatment method (autocenter ('autocenter') or mean center ('center')
method = 'center';
% Number of MC sampling
N = 1000;
% The ratio of samples randomly selected to build a PLS model, default 0.75.
ratio = 0.7;


addpath('C:\Users\micha\OneDrive - University of Strathclyde\Thesis\Chemometric\PLSToolboxWorkMichael\libPLS_1.95')
rmpath('C:\Users\micha\OneDrive - University of Strathclyde\Thesis\Chemometric\PLSToolboxWorkMichael\libPLS_1.95')


%% -- Outlier detection Matlab forums based on 3 std deviations

%all_idx = 1:length(x)
%outlier_idx = abs(x - median(x)) > 3*std(x) | abs(y - median(y)) > 3*std(y) % Find outlier idx
%x(outlier_idx) = interp1(all_idx(~outlier_idx), x(~outlier_idx), all_idx(outlier_idx)) % Linearly interpolate over outlier idx for x
%y(outlier_idx) = interp1(all_idx(~outlier_idx), y(~outlier_idx), all_idx(outlier_idx)) % Do the same thing for y


%% -- Monte Carlo Outlier Detection Algorithm
% taken from Li H.-D., Xu Q.-S., Liang Y.-Z. (2014) libPLS: An Integrated Library for Partial Least Squares Regression and Discriminant Analysis. PeerJ PrePrints 2:e190v1, source codes available at www.libpls.net.

% parameters if not defined previously

% pretreatment method (autocenter ('autocenter') or mean center ('center')
%method = 'center';
% Number of MC sampling
%N = 1000;
% The ratio of samples randomly selected to build a PLS model, default 0.75.
%ratio = 0.7
% number of latent variables for MC
%MClv = 4;


```

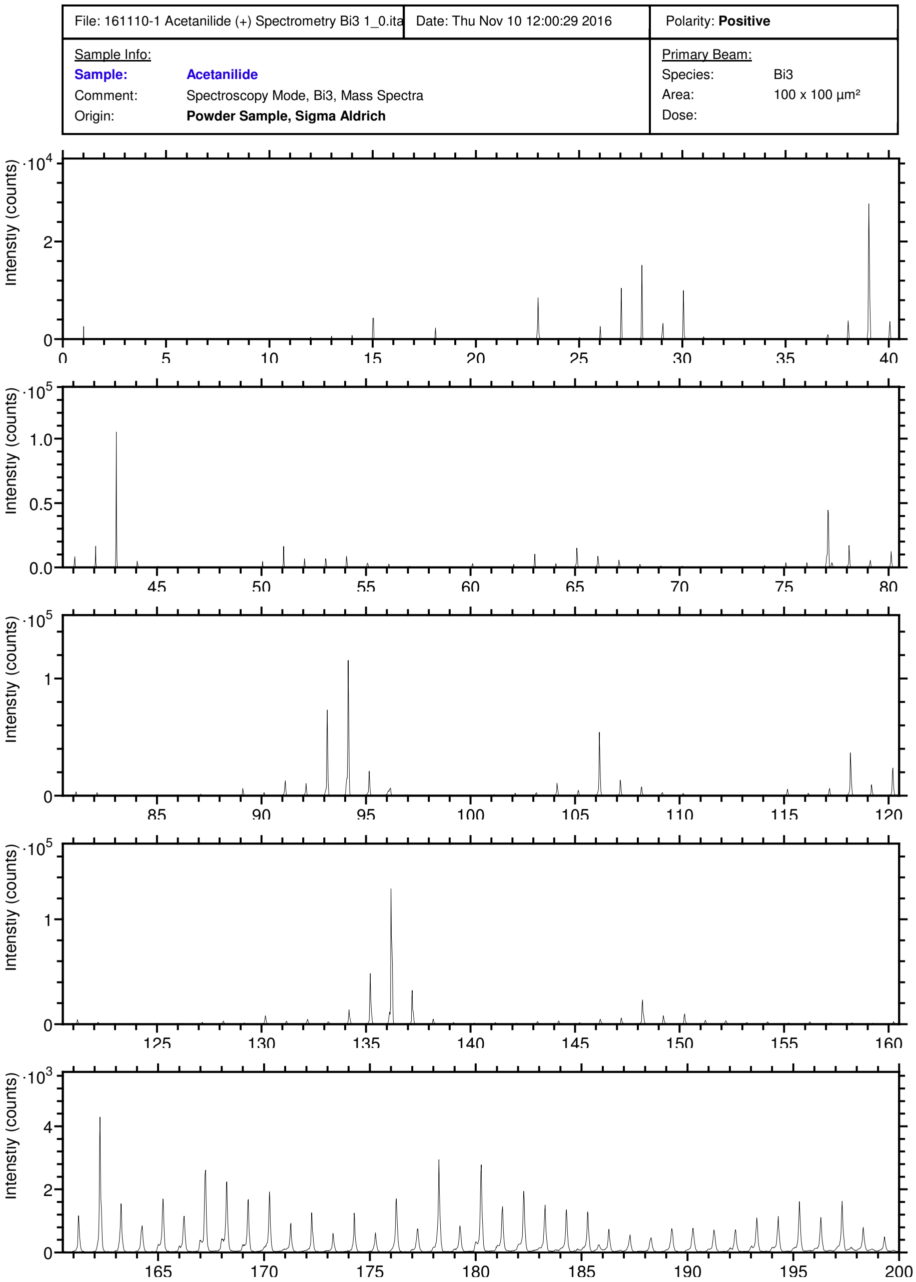
```
for ii = 1:app.MClv
A=ii;
app.F=mcs(app.xc_temp,app.yc_temp,A,method,N,ratio)
app.plotmcs_appedit(app.F);
%app.analysis.MonteCarloPredError = app.F.predError;
end
hold(app.FigureODplot,"off");


        end

        % Value changed function: CheckallCheckBox
        function CheckallCheckBoxValueChanged(app, event)
            value = app.CheckallCheckBox.Value;

            if value

                app.FirstDerivativeSecondOrderCheckBox.Value=1;
                    app.act_dev1_ord2=1;
                app.FirstDerivativeFourthOrderCheckBox.Value=1;
                    app.act_dev1_ord4=1;
                app.SecondDerivativeSecondOrderCheckBox.Value=1;
                    app.act_dev2_ord2=1;
                app.SecondDerivativeFourthOrderCheckBox.Value=1;
                    app.act_dev2_ord4=1;
                app.SavitzkyGolaySmoothingOrder2CheckBox.Value=1;
                    app.act_smooth_ord2=1;
                app.SavitzkyGolaySmoothingOrder4CheckBox.Value=1;
                    app.act_smooth_ord4=1;
                app.ParetoScalingCheckBox.Value=1;
                    app.act_pareto=1;
                app.AutoScalingCheckBox.Value=1;
                    app.act_autoscale=1;
                app.MultipleScatterCorrectionCheckBox.Value=1;
                    app.act_MSC=1;
                app.StandardNormalVariateCheckBox.Value=1;
                    app.act_SNV=1;
            else

                app.FirstDerivativeSecondOrderCheckBox.Value=0;
                    app.act_dev1_ord2=0;
                app.FirstDerivativeFourthOrderCheckBox.Value=0;
                    app.act_dev1_ord4=0;
                app.SecondDerivativeSecondOrderCheckBox.Value=0;
                    app.act_dev2_ord2=0;
                app.SecondDerivativeFourthOrderCheckBox.Value=0;
                    app.act_dev2_ord4=0;
                app.SavitzkyGolaySmoothingOrder2CheckBox.Value=0;
                    app.act_smooth_ord2=0;
                app.SavitzkyGolaySmoothingOrder4CheckBox.Value=0;
                    app.act_smooth_ord4=0;
                app.ParetoScalingCheckBox.Value=0;
                    app.act_pareto=0;
                app.AutoScalingCheckBox.Value=0;
                    app.act_autoscale=0;
                app.MultipleScatterCorrectionCheckBox.Value=0;
                    app.act_MSC=0;
                app.StandardNormalVariateCheckBox.Value=0;
```

```
                    app.act_SNV=0;
            end
        end

        % Value changed function:
        % NumberofLVsforoutlierdetectionEditField
        function NumberofLVsforoutlierdetectionEditFieldValueChanged(app, event)
            value = app.NumberofLVsforoutlierdetectionEditField.Value;
            app.MClv = value;
        end

        % Button pushed function: AdjustDataButton
        function AdjustDataButtonPushed(app, event)

if app.mc_mcoutlier == 0
    end
if app.mc_outlier == 1

    %% inputs for thresholds

    %if app.MCstd_value == 0
    %     uialert(app.FigureODplot,'No standard deviation error reduction in data detected. Change value to something else than 0','Invalid input');
    %end

    app.xcOc = app.xc;
    app.ycOc = app.yc;

    if app.MCmean_value == 0 || app.MCstd_value == 0
    end
    if app.MCmean_value ~= 0 && app.MCstd_value ~= 0


        % get index of rows that exceed threshold for outliers
        idx_mean = find(app.F.MEAN > app.MCmean_value);
        idx_std = find(app.F.STD > app.MCstd_value);
        % nan rows that exceed threshold

        if any(idx_mean)
            app.xcOc(idx_mean, :) = nan;
            app.ycOc(idx_mean, :) = nan;
        end

        if any(idx_std)
            app.xcOc(idx_std, :) = nan;
            app.ycOc(idx_std, :) = nan;
        end

        app.xcOc(any(isnan(app.xcOc),2),:) = [];
        app.ycOc(any(isnan(app.ycOc),2),:) = [];

        app.xc_temp = app.xcOc;
        app.yc_temp = app.ycOc;
    end
else
end

        end

        % Value changed function: SelectedvaluesCheckBox
        function SelectedvaluesCheckBoxValueChanged(app, event)
            value = app.SelectedvaluesCheckBox.Value
            if value
                app.AdjustDataButton.Visible = "on"
            else
                app.AdjustDataButton.Visible = "off"
            end
        end

        % Value changed function: MeanvalueEditField
        function MeanvalueEditFieldValueChanged(app, event)
            value = app.MeanvalueEditField.Value;
            app.MCmean_value = value;
        end

        % Value changed function: StandarddeviationvalueEditField
        function StandarddeviationvalueEditFieldValueChanged(app, event)
            value = app.StandarddeviationvalueEditField.Value;
            app.MCstd_value = value;
        end

        % Button pushed function: DevLoad2Button
        function DevLoad2ButtonPushed(app, event)
            app.xc = xlsread("F:\Thesis\Matlab\x40c_cal.xlsx")
            app.yc = xlsread("F:\Thesis\Matlab\y40c_cal.xlsx")
            app.xv = xlsread("F:\Thesis\Matlab\x40c_val.xlsx")
            app.yv = xlsread("F:\Thesis\Matlab\y40c_val.xlsx")
        end

        % Button pushed function: DevLoad3Button
        function DevLoad3ButtonPushed(app, event)
            app.xc = xlsread("F:\Thesis\Matlab\x50c_cal.xlsx")
            app.yc = xlsread("F:\Thesis\Matlab\y50c_cal.xlsx")
            app.xv = xlsread("F:\Thesis\Matlab\x50c_val.xlsx")
            app.yv = xlsread("F:\Thesis\Matlab\y50c_val.xlsx")
        end
    end

    % Component initialization
    methods (Access = private)

        % Create UIFigure and components
        function createComponents(app)

            % Create DoEPreProcessingBenchmarkToolboxUIFigure and hide until all components
                are created
            app.DoEPreProcessingBenchmarkToolboxUIFigure = uifigure('Visible', 'off');
            app.DoEPreProcessingBenchmarkToolboxUIFigure.Position = [100 100 1335 940];
            app.DoEPreProcessingBenchmarkToolboxUIFigure.Name = 'DoE Pre-Processing Benchmark
                 Toolbox';

            % Create TabGroup
            app.TabGroup = uitabgroup(app.DoEPreProcessingBenchmarkToolboxUIFigure);
            app.TabGroup.Position = [1 9 1325 932];

            % Create IntroductionTab
            app.IntroductionTab = uitab(app.TabGroup);
            app.IntroductionTab.Title = 'Introduction';
            app.IntroductionTab.BackgroundColor = [1 1 1];

            % Create DoEBasedPreProcessingOptimizationToolLabel
            app.DoEBasedPreProcessingOptimizationToolLabel = uilabel(app.IntroductionTab);
            app.DoEBasedPreProcessingOptimizationToolLabel.FontSize = 16;
            app.DoEBasedPreProcessingOptimizationToolLabel.FontWeight = 'bold';
            app.DoEBasedPreProcessingOptimizationToolLabel.Position = [42 800 355 22];
            app.DoEBasedPreProcessingOptimizationToolLabel.Text = 'DoE Based Pre-Processing
                Optimization Tool';

            % Create Label_2
            app.Label_2 = uilabel(app.IntroductionTab);
            app.Label_2.Position = [42 711 699 70];
            app.Label_2.Text = {'This toolbox is a MATLAB based pre-processing optmization
                tool. '; 'Spectral data that is loaded into the program will receive the
                selected pre-processing treatments and all such data will be saved.'; 'The
                saved data will then be modelled using partial least squares regression and
                the most optimal method displayed.'; ''; 'Optional cross-validation and
                outlier detection are available. '};

            % Create DataLoadingTab
            app.DataLoadingTab = uitab(app.TabGroup);
            app.DataLoadingTab.Title = 'Data Loading';
            app.DataLoadingTab.BackgroundColor = [1 1 1];

            % Create LoadXcalibrationButton
            app.LoadXcalibrationButton = uibutton(app.DataLoadingTab, 'push');
            app.LoadXcalibrationButton.ButtonPushedFcn = createCallbackFcn(app,
                @LoadXcalibrationButtonPushed, true);
            app.LoadXcalibrationButton.HorizontalAlignment = 'left';
            app.LoadXcalibrationButton.Position = [1088 115 112 22];
            app.LoadXcalibrationButton.Text = 'Load X calibration';

            % Create DataPreviewTable
            app.DataPreviewTable = uitable(app.DataLoadingTab);
            app.DataPreviewTable.ColumnName = {''};
            app.DataPreviewTable.RowName = {};
            app.DataPreviewTable.Position = [12 154 1284 733];

            % Create xCalView
            app.xCalView = uibutton(app.DataLoadingTab, 'push');
            app.xCalView.ButtonPushedFcn = createCallbackFcn(app, @xCalViewButtonPushed, true
                );
            app.xCalView.Position = [1228 115 47 22];
            app.xCalView.Text = 'View';

            % Create yCalView
            app.yCalView = uibutton(app.DataLoadingTab, 'push');
            app.yCalView.ButtonPushedFcn = createCallbackFcn(app, @yCalViewButtonPushed, true
                );
            app.yCalView.Position = [1228 94 47 22];
            app.yCalView.Text = 'View';

            % Create xValView
            app.xValView = uibutton(app.DataLoadingTab, 'push');
            app.xValView.ButtonPushedFcn = createCallbackFcn(app, @xValViewButtonPushed, true);
            app.xValView.Position = [1228 73 47 22];
            app.xValView.Text = 'View';

            % Create yValView
            app.yValView = uibutton(app.DataLoadingTab, 'push');
            app.yValView.ButtonPushedFcn = createCallbackFcn(app, @yValViewButtonPushed, true);
            app.yValView.Position = [1228 52 47 22];
            app.yValView.Text = 'View';

            % Create xCalLamp
            app.xCalLamp = uilamp(app.DataLoadingTab);
            app.xCalLamp.Position = [1204 117 18 18];
            app.xCalLamp.Color = [1 0 0];

            % Create yCalLamp
            app.yCalLamp = uilamp(app.DataLoadingTab);
            app.yCalLamp.Position = [1204 96 18 18];
            app.yCalLamp.Color = [1 0 0];

            % Create xValLamp
            app.xValLamp = uilamp(app.DataLoadingTab);
            app.xValLamp.Position = [1204 75 18 18];
            app.xValLamp.Color = [1 0 0];

            % Create yValLamp
            app.yValLamp = uilamp(app.DataLoadingTab);
            app.yValLamp.Position = [1204 54 18 18];
            app.yValLamp.Color = [1 0 0];

            % Create LoadYcalibrationButton
            app.LoadYcalibrationButton = uibutton(app.DataLoadingTab, 'push');
            app.LoadYcalibrationButton.ButtonPushedFcn = createCallbackFcn(app, @LoadYcalibrationButtonPushed, true);
            app.LoadYcalibrationButton.HorizontalAlignment = 'left';
            app.LoadYcalibrationButton.Position = [1088 94 112 22];
            app.LoadYcalibrationButton.Text = 'Load Y calibration';

            % Create LoadXvalidationButton
            app.LoadXvalidationButton = uibutton(app.DataLoadingTab, 'push');
            app.LoadXvalidationButton.ButtonPushedFcn = createCallbackFcn(app, @LoadXvalidationButtonPushed, true);
            app.LoadXvalidationButton.HorizontalAlignment = 'left';
            app.LoadXvalidationButton.Position = [1088 73 112 22];
            app.LoadXvalidationButton.Text = 'Load X validation';

            % Create LoadYvalidationButton
            app.LoadYvalidationButton = uibutton(app.DataLoadingTab, 'push');
            app.LoadYvalidationButton.ButtonPushedFcn = createCallbackFcn(app, @LoadYvalidationButtonPushed, true);
            app.LoadYvalidationButton.HorizontalAlignment = 'left';
            app.LoadYvalidationButton.Position = [1088 52 112 22];
            app.LoadYvalidationButton.Text = 'Load Y validation';

            % Create Label
            app.Label = uilabel(app.DataLoadingTab);
            app.Label.HorizontalAlignment = 'center';
            app.Label.Position = [558 38 324 80];
            app.Label.Text = {'Load data into the script by clicking the respective buttons.'; 'Data should be in .XLS or .XLSX format and formatted so'; 'that the rows represent the different measurements '; 'and the columns the different measurement wavelengths.'};

            % Create InstructionsLabel
            app.InstructionsLabel = uilabel(app.DataLoadingTab);
            app.InstructionsLabel.HorizontalAlignment = 'center';
            app.InstructionsLabel.FontSize = 14;
            app.InstructionsLabel.FontWeight = 'bold';
            app.InstructionsLabel.Position = [677 117 86 22];
            app.InstructionsLabel.Text = 'Instructions';

            % Create DevLoadButton
            app.DevLoadButton = uibutton(app.DataLoadingTab, 'push');
            app.DevLoadButton.ButtonPushedFcn = createCallbackFcn(app, @DevLoadButtonPushed, true);
            app.DevLoadButton.Position = [906 81 100 22];
            app.DevLoadButton.Text = 'Dev Load';

            % Create DevLoad2Button
            app.DevLoad2Button = uibutton(app.DataLoadingTab, 'push');
            app.DevLoad2Button.ButtonPushedFcn = createCallbackFcn(app, @DevLoad2ButtonPushed, true);
            app.DevLoad2Button.Position = [906 52 100 22];
            app.DevLoad2Button.Text = 'Dev Load2';

            % Create DevLoad3Button
            app.DevLoad3Button = uibutton(app.DataLoadingTab, 'push');
            app.DevLoad3Button.ButtonPushedFcn = createCallbackFcn(app, @DevLoad3ButtonPushed, true);
            app.DevLoad3Button.Position = [906 23 100 22];
            app.DevLoad3Button.Text = 'Dev Load3';

            % Create RunandOptionsTab
            app.RunandOptionsTab = uitab(app.TabGroup);
            app.RunandOptionsTab.Title = 'Run and Options';
            app.RunandOptionsTab.BackgroundColor = [1 1 1];
            app.RunandOptionsTab.Scrollable = 'on';

            % Create FirstDerivativeSecondOrderCheckBox
            app.FirstDerivativeSecondOrderCheckBox = uicheckbox(app.RunandOptionsTab);
            app.FirstDerivativeSecondOrderCheckBox.ValueChangedFcn = createCallbackFcn(app, @FirstDerivativeSecondOrderCheckBoxValueChanged, true);
            app.FirstDerivativeSecondOrderCheckBox.Text = 'First Derivative Second Order';
            app.FirstDerivativeSecondOrderCheckBox.Position = [33 770 180 22];

            % Create FirstDerivativeFourthOrderCheckBox
            app.FirstDerivativeFourthOrderCheckBox = uicheckbox(app.RunandOptionsTab);
            app.FirstDerivativeFourthOrderCheckBox.ValueChangedFcn = createCallbackFcn(app, @FirstDerivativeFourthOrderCheckBoxValueChanged, true);
            app.FirstDerivativeFourthOrderCheckBox.Text = 'First Derivative Fourth Order';
            app.FirstDerivativeFourthOrderCheckBox.Position = [33 749 174 22];

            % Create SecondDerivativeSecondOrderCheckBox
            app.SecondDerivativeSecondOrderCheckBox = uicheckbox(app.RunandOptionsTab);
            app.SecondDerivativeSecondOrderCheckBox.ValueChangedFcn = createCallbackFcn(app, @SecondDerivativeSecondOrderCheckBoxValueChanged, true);
            app.SecondDerivativeSecondOrderCheckBox.Text = 'Second Derivative Second Order';
            app.SecondDerivativeSecondOrderCheckBox.Position = [33 728 198 22];

            % Create SecondDerivativeFourthOrderCheckBox
            app.SecondDerivativeFourthOrderCheckBox = uicheckbox(app.RunandOptionsTab);
            app.SecondDerivativeFourthOrderCheckBox.ValueChangedFcn = createCallbackFcn(app, @SecondDerivativeFourthOrderCheckBoxValueChanged, true);
            app.SecondDerivativeFourthOrderCheckBox.Text = 'Second Derivative Fourth Order';
            app.SecondDerivativeFourthOrderCheckBox.Position = [33 707 192 22];

            % Create BaselineCorrectionLabel
            app.BaselineCorrectionLabel = uilabel(app.RunandOptionsTab);
            app.BaselineCorrectionLabel.FontWeight = 'bold';
            app.BaselineCorrectionLabel.Position = [33 812 119 22];
            app.BaselineCorrectionLabel.Text = 'Baseline Correction';

            % Create PreProcessingMethodSelectionLabel
            app.PreProcessingMethodSelectionLabel = uilabel(app.RunandOptionsTab);
            app.PreProcessingMethodSelectionLabel.FontSize = 15;
            app.PreProcessingMethodSelectionLabel.FontWeight = 'bold';
            app.PreProcessingMethodSelectionLabel.Position = [33 854 246 22];
            app.PreProcessingMethodSelectionLabel.Text = 'Pre-Processing Method Selection';

            % Create SavitzkyGolaySmoothingOrder2CheckBox
            app.SavitzkyGolaySmoothingOrder2CheckBox = uicheckbox(app.RunandOptionsTab);
            app.SavitzkyGolaySmoothingOrder2CheckBox.ValueChangedFcn = createCallbackFcn(app, @SavitzkyGolaySmoothingOrder2CheckBoxValueChanged, true);
            app.SavitzkyGolaySmoothingOrder2CheckBox.Text = 'Savitzky Golay Smoothing Order 2';
            app.SavitzkyGolaySmoothingOrder2CheckBox.Position = [292 770 207 22];

            % Create SavitzkyGolaySmoothingOrder4CheckBox
            app.SavitzkyGolaySmoothingOrder4CheckBox = uicheckbox(app.RunandOptionsTab);
            app.SavitzkyGolaySmoothingOrder4CheckBox.ValueChangedFcn = createCallbackFcn(app, @SavitzkyGolaySmoothingOrder4CheckBoxValueChanged, true);
            app.SavitzkyGolaySmoothingOrder4CheckBox.Text = 'Savitzky Golay Smoothing Order 4';
            app.SavitzkyGolaySmoothingOrder4CheckBox.Position = [292 749 207 22];

            % Create SmoothingLabel
            app.SmoothingLabel = uilabel(app.RunandOptionsTab);
            app.SmoothingLabel.FontWeight = 'bold';
            app.SmoothingLabel.Position = [292 812 68 22];
            app.SmoothingLabel.Text = 'Smoothing';

            % Create ParetoScalingCheckBox
            app.ParetoScalingCheckBox = uicheckbox(app.RunandOptionsTab);
            app.ParetoScalingCheckBox.ValueChangedFcn = createCallbackFcn(app, @ParetoScalingCheckBoxValueChanged, true);
            app.ParetoScalingCheckBox.Text = 'Pareto Scaling';
            app.ParetoScalingCheckBox.Position = [33 615 100 22];

            % Create AutoScalingCheckBox
            app.AutoScalingCheckBox = uicheckbox(app.RunandOptionsTab);
            app.AutoScalingCheckBox.ValueChangedFcn = createCallbackFcn(app, @AutoScalingCheckBoxValueChanged, true);
            app.AutoScalingCheckBox.Text = 'Auto Scaling';
            app.AutoScalingCheckBox.Position = [33 594 89 22];

            % Create ScalingLabel
            app.ScalingLabel = uilabel(app.RunandOptionsTab);
            app.ScalingLabel.FontWeight = 'bold';
            app.ScalingLabel.Position = [33 657 48 22];
            app.ScalingLabel.Text = 'Scaling';

            % Create MultipleScatterCorrectionCheckBox
            app.MultipleScatterCorrectionCheckBox = uicheckbox(app.RunandOptionsTab);
            app.MultipleScatterCorrectionCheckBox.ValueChangedFcn = createCallbackFcn(app,
                @MultipleScatterCorrectionCheckBoxValueChanged, true);
            app.MultipleScatterCorrectionCheckBox.Text = 'Multiple Scatter Correction';
            app.MultipleScatterCorrectionCheckBox.Position = [292 615 163 22];

            % Create StandardNormalVariateCheckBox
            app.StandardNormalVariateCheckBox = uicheckbox(app.RunandOptionsTab);
            app.StandardNormalVariateCheckBox.ValueChangedFcn = createCallbackFcn(app,
                @StandardNormalVariateCheckBoxValueChanged, true);
            app.StandardNormalVariateCheckBox.Text = 'Standard Normal Variate';
            app.StandardNormalVariateCheckBox.Position = [292 594 153 22];

            % Create ScatterCorrectionLabel
            app.ScatterCorrectionLabel = uilabel(app.RunandOptionsTab);
            app.ScatterCorrectionLabel.FontWeight = 'bold';
            app.ScatterCorrectionLabel.Position = [292 657 110 22];
            app.ScatterCorrectionLabel.Text = 'Scatter Correction';

            % Create OutlierDetectionCheckBox
            app.OutlierDetectionCheckBox = uicheckbox(app.RunandOptionsTab);
            app.OutlierDetectionCheckBox.ValueChangedFcn = createCallbackFcn(app,
                @OutlierDetectionCheckBoxValueChanged, true);
            app.OutlierDetectionCheckBox.Tooltip = {'Activates "Outlier Detection Tool"'};
            app.OutlierDetectionCheckBox.Text = 'Outlier Detection';
            app.OutlierDetectionCheckBox.Position = [727 615 111 22];

            % Create CrossValidationCheckBox
            app.CrossValidationCheckBox = uicheckbox(app.RunandOptionsTab);
            app.CrossValidationCheckBox.ValueChangedFcn = createCallbackFcn(app,
                @CrossValidationCheckBoxValueChanged, true);
            app.CrossValidationCheckBox.Text = 'Cross-Validation';
            app.CrossValidationCheckBox.Position = [727 594 109 22];

            % Create RunButton
            app.RunButton = uibutton(app.RunandOptionsTab, 'push');
            app.RunButton.ButtonPushedFcn = createCallbackFcn(app, @RunButtonPushed, true);
            app.RunButton.Position = [159 423 100 22];
            app.RunButton.Text = 'Run';

            % Create ResetButton
            app.ResetButton = uibutton(app.RunandOptionsTab, 'push');
            app.ResetButton.Position = [283 423 100 22];
            app.ResetButton.Text = 'Reset';

            % Create MaximumnumberoflatentvariablesEditFieldLabel
            app.MaximumnumberoflatentvariablesEditFieldLabel = uilabel(app.RunandOptionsTab);
            app.MaximumnumberoflatentvariablesEditFieldLabel.HorizontalAlignment = 'right';
            app.MaximumnumberoflatentvariablesEditFieldLabel.FontWeight = 'bold';
            app.MaximumnumberoflatentvariablesEditFieldLabel.Position = [673 803 215 22];
            app.MaximumnumberoflatentvariablesEditFieldLabel.Text = 'Maximum number of latent variables';

            % Create MaximumnumberoflatentvariablesEditField
            app.MaximumnumberoflatentvariablesEditField = uieditfield(app.RunandOptionsTab, 'numeric');
            app.MaximumnumberoflatentvariablesEditField.Limits = [2 100];
            app.MaximumnumberoflatentvariablesEditField.RoundFractionalValues = 'on';
            app.MaximumnumberoflatentvariablesEditField.ValueChangedFcn = createCallbackFcn(app, @MaximumnumberoflatentvariablesEditFieldValueChanged, true);
            app.MaximumnumberoflatentvariablesEditField.Position = [737 772 100 22];
            app.MaximumnumberoflatentvariablesEditField.Value = 3;

            % Create BinningfactorforsmoothingandbaselinecorrectionListBoxLabel
            app.BinningfactorforsmoothingandbaselinecorrectionListBoxLabel = uilabel(app.RunandOptionsTab);
            app.BinningfactorforsmoothingandbaselinecorrectionListBoxLabel.HorizontalAlignment = 'right';
            app.BinningfactorforsmoothingandbaselinecorrectionListBoxLabel.FontWeight = 'bold';
            app.BinningfactorforsmoothingandbaselinecorrectionListBoxLabel.Position = [622 692 313 22];
            app.BinningfactorforsmoothingandbaselinecorrectionListBoxLabel.Text = 'Binning factor for smoothing and baseline correction';

            % Create WidthBinning
            app.WidthBinning = uilistbox(app.RunandOptionsTab);
            app.WidthBinning.Items = {'5', '7', '9', '11', '13', '15'};
            app.WidthBinning.ItemsData = {'5', '7', '9', '11', '13', '15'};
            app.WidthBinning.ValueChangedFcn = createCallbackFcn(app, @WidthBinningValueChanged, true);
            app.WidthBinning.Position = [948 647 100 112];
            app.WidthBinning.Value = '11';

            % Create StatusLabel
            app.StatusLabel = uilabel(app.RunandOptionsTab);
            app.StatusLabel.FontSize = 15;
            app.StatusLabel.FontWeight = 'bold';
            app.StatusLabel.Position = [250 359 51 22];
            app.StatusLabel.Text = 'Status';

            % Create LabelStatus
            app.LabelStatus = uilabel(app.RunandOptionsTab);
            app.LabelStatus.HorizontalAlignment = 'center';
            app.LabelStatus.Position = [132 328 286 22];
            app.LabelStatus.Text = 'none';

            % Create OutlierDetectionPanel
            app.OutlierDetectionPanel = uipanel(app.RunandOptionsTab);
            app.OutlierDetectionPanel.Title = 'Outlier Detection';
            app.OutlierDetectionPanel.Position = [598 12 707 538];

            % Create FigureODplot
            app.FigureODplot = uiaxes(app.OutlierDetectionPanel);
            title(app.FigureODplot, 'Monte-Carlo based outlier detection')
            xlabel(app.FigureODplot, 'Mean')
            ylabel(app.FigureODplot, 'Std')
            app.FigureODplot.Position = [16 61 677 454];

            % Create MeanvalueEditFieldLabel
            app.MeanvalueEditFieldLabel = uilabel(app.OutlierDetectionPanel);
            app.MeanvalueEditFieldLabel.HorizontalAlignment = 'right';
            app.MeanvalueEditFieldLabel.Position = [327 33 68 22];
            app.MeanvalueEditFieldLabel.Text = 'Mean value';

            % Create MeanvalueEditField
            app.MeanvalueEditField = uieditfield(app.OutlierDetectionPanel, 'numeric');
            app.MeanvalueEditField.Limits = [0 Inf];
            app.MeanvalueEditField.ValueChangedFcn = createCallbackFcn(app, @MeanvalueEditFieldValueChanged, true);
            app.MeanvalueEditField.Position = [410 33 100 22];

            % Create StandarddeviationvalueEditFieldLabel
            app.StandarddeviationvalueEditFieldLabel = uilabel(app.OutlierDetectionPanel);
            app.StandarddeviationvalueEditFieldLabel.HorizontalAlignment = 'right';
            app.StandarddeviationvalueEditFieldLabel.Position = [257 4 138 22];
            app.StandarddeviationvalueEditFieldLabel.Text = 'Standard deviation value';

            % Create StandarddeviationvalueEditField
            app.StandarddeviationvalueEditField = uieditfield(app.OutlierDetectionPanel, 'numeric');
            app.StandarddeviationvalueEditField.Limits = [0 Inf];
            app.StandarddeviationvalueEditField.ValueChangedFcn = createCallbackFcn(app, @StandarddeviationvalueEditFieldValueChanged, true);
            app.StandarddeviationvalueEditField.Position = [410 4 100 22];

            % Create RunOD
            app.RunOD = uibutton(app.OutlierDetectionPanel, 'push');
            app.RunOD.ButtonPushedFcn = createCallbackFcn(app, @RunODPushed, true);
            app.RunOD.Position = [72 7 100 22];
            app.RunOD.Text = 'Plot';

            % Create NumberofLVsforoutlierdetectionEditFieldLabel
            app.NumberofLVsforoutlierdetectionEditFieldLabel = uilabel(app.OutlierDetectionPanel);
            app.NumberofLVsforoutlierdetectionEditFieldLabel.HorizontalAlignment = 'right';
            app.NumberofLVsforoutlierdetectionEditFieldLabel.Position = [21 30 191 22];
            app.NumberofLVsforoutlierdetectionEditFieldLabel.Text = 'Number of LVs for outlier detection';

            % Create NumberofLVsforoutlierdetectionEditField
            app.NumberofLVsforoutlierdetectionEditField = uieditfield(app.OutlierDetectionPanel, 'numeric');
            app.NumberofLVsforoutlierdetectionEditField.Limits = [0 10];
            app.NumberofLVsforoutlierdetectionEditField.RoundFractionalValues = 'on';
            app.NumberofLVsforoutlierdetectionEditField.ValueChangedFcn = createCallbackFcn(app, @NumberofLVsforoutlierdetectionEditFieldValueChanged, true);
            app.NumberofLVsforoutlierdetectionEditField.Position = [215 30 27 22];
            app.NumberofLVsforoutlierdetectionEditField.Value = 3;

            % Create SelectedvaluesCheckBox
            app.SelectedvaluesCheckBox = uicheckbox(app.OutlierDetectionPanel);
            app.SelectedvaluesCheckBox.ValueChangedFcn = createCallbackFcn(app, @SelectedvaluesCheckBoxValueChanged, true);
            app.SelectedvaluesCheckBox.Text = 'Selected values';
            app.SelectedvaluesCheckBox.Position = [525 33 107 22];

            % Create AdjustDataButton
            app.AdjustDataButton = uibutton(app.OutlierDetectionPanel, 'push');
            app.AdjustDataButton.ButtonPushedFcn = createCallbackFcn(app,
                @AdjustDataButtonPushed, true);
            app.AdjustDataButton.BackgroundColor = [1 0 0];
            app.AdjustDataButton.FontWeight = 'bold';
            app.AdjustDataButton.FontColor = [1 1 1];
            app.AdjustDataButton.Visible = 'off';
            app.AdjustDataButton.Position = [538 9 82 22];
            app.AdjustDataButton.Text = 'Adjust Data';

            % Create CheckallCheckBox
            app.CheckallCheckBox = uicheckbox(app.RunandOptionsTab);
            app.CheckallCheckBox.ValueChangedFcn = createCallbackFcn(app,
                @CheckallCheckBoxValueChanged, true);
            app.CheckallCheckBox.Text = 'Check all';
            app.CheckallCheckBox.Position = [212 540 71 22];

            % Create DataVisualisationTab
            app.DataVisualisationTab = uitab(app.TabGroup);
            app.DataVisualisationTab.Title = 'Data Visualisation';
            app.DataVisualisationTab.BackgroundColor = [1 1 1];
            app.DataVisualisationTab.Scrollable = 'on';

            % Create FigureData
            app.FigureData = uiaxes(app.DataVisualisationTab);
            title(app.FigureData, 'Title')
            xlabel(app.FigureData, 'X')
            ylabel(app.FigureData, 'Y')
            app.FigureData.Position = [9 88 1284 810];

            % Create LVvsRMSEButton
            app.LVvsRMSEButton = uibutton(app.DataVisualisationTab, 'push');
            app.LVvsRMSEButton.ButtonPushedFcn = createCallbackFcn(app, @LVvsRMSEButtonPushed
                , true);
            app.LVvsRMSEButton.Position = [954 59 124 22];
            app.LVvsRMSEButton.Text = 'LV vs RMSE';

            % Create InteractionPlotsButton
            app.InteractionPlotsButton = uibutton(app.DataVisualisationTab, 'push');
            app.InteractionPlotsButton.ButtonPushedFcn = createCallbackFcn(app,
                @InteractionPlotsButtonPushed, true);
            app.InteractionPlotsButton.Tooltip = {'Shows interaction plots between the
                different pre-processing and model options. (Note that if more than 10 LVs
                are required'; ' the other interaction plot option should be chosen).'};
            app.InteractionPlotsButton.Position = [1109 34 105 22];
            app.InteractionPlotsButton.Text = 'Interaction Plots';

            % Create MainEffectsPlotButton
            app.MainEffectsPlotButton = uibutton(app.DataVisualisationTab, 'push');
            app.MainEffectsPlotButton.ButtonPushedFcn = createCallbackFcn(app,
                @MainEffectsPlotButtonPushed, true);
            app.MainEffectsPlotButton.Position = [1109 59 105 22];
            app.MainEffectsPlotButton.Text = 'Main Effects Plot';

            % Create BestOverallResultsButton
            app.BestOverallResultsButton = uibutton(app.DataVisualisationTab, 'push');
            app.BestOverallResultsButton.ButtonPushedFcn = createCallbackFcn(app, @BestOverallResultsButtonPushed, true);
            app.BestOverallResultsButton.Position = [954 34 124 22];
            app.BestOverallResultsButton.Text = 'Best Overall Results';

            % Create InteractionplotLegendPanel
            app.InteractionplotLegendPanel = uipanel(app.DataVisualisationTab);
            app.InteractionplotLegendPanel.Title = 'Interactionplot Legend';
            app.InteractionplotLegendPanel.Visible = 'off';
            app.InteractionplotLegendPanel.Position = [46 41 611 40];

            % Create ABaselinecorrectionLabel
            app.ABaselinecorrectionLabel = uilabel(app.InteractionplotLegendPanel);
            app.ABaselinecorrectionLabel.Tooltip = {'No correction=0'; 'First derivative Second Order=1'; 'First derivative Fourth Order=2'; 'Second derivative Second Order=3    '; 'Second derivative Fourth Order=4'};
            app.ABaselinecorrectionLabel.Position = [4 -4 129 22];
            app.ABaselinecorrectionLabel.Text = 'A = Baseline correction';

            % Create BScattercorrectionLabel
            app.BScattercorrectionLabel = uilabel(app.InteractionplotLegendPanel);
            app.BScattercorrectionLabel.Tooltip = {'No correction = 0'; 'Multiple Scatter Correction = 1'; 'Standard Normal Variate Correction = 2'};
            app.BScattercorrectionLabel.Position = [143 -4 121 22];
            app.BScattercorrectionLabel.Text = 'B = Scatter correction';

            % Create CSmoothingLabel
            app.CSmoothingLabel = uilabel(app.InteractionplotLegendPanel);
            app.CSmoothingLabel.Tooltip = {'No correction = 0'; 'Savitzky-Golay smoothing second order = 1'; 'Savitzky-Golay smoothing fourth order = 2'};
            app.CSmoothingLabel.Position = [274 -4 85 22];
            app.CSmoothingLabel.Text = 'C = Smoothing';

            % Create DScalingLabel
            app.DScalingLabel = uilabel(app.InteractionplotLegendPanel);
            app.DScalingLabel.Tooltip = {'No scaling = 0'; 'Pareteo scaling = 1'; ' Autoscaling = 2'};
            app.DScalingLabel.Position = [368 -4 67 22];
            app.DScalingLabel.Text = 'D = Scaling';

            % Create ENumberoflatentvariablesLabel
            app.ENumberoflatentvariablesLabel = uilabel(app.InteractionplotLegendPanel);
            app.ENumberoflatentvariablesLabel.Tooltip = {'Number of latent variables used in the model'};
            app.ENumberoflatentvariablesLabel.Position = [444 -4 167 22];
            app.ENumberoflatentvariablesLabel.Text = 'E = Number of latent variables';

            % Create ActivateLegendCheckBox
            app.ActivateLegendCheckBox = uicheckbox(app.DataVisualisationTab);
            app.ActivateLegendCheckBox.ValueChangedFcn = createCallbackFcn(app, @ActivateLegendCheckBoxValueChanged, true);
            app.ActivateLegendCheckBox.Text = 'Activate Legend';
            app.ActivateLegendCheckBox.FontSize = 8;
            app.ActivateLegendCheckBox.FontWeight = 'bold';
            app.ActivateLegendCheckBox.Position = [1123 15 94 14];

            % Create PButton_2
            app.PButton_2 = uibutton(app.DataVisualisationTab, 'push');
            app.PButton_2.ButtonPushedFcn = createCallbackFcn(app, @PButton_2Pushed, true);
            app.PButton_2.FontSize = 8;
            app.PButton_2.FontWeight = 'bold';
            app.PButton_2.Tooltip = {'Popout graph for additional editability and functionality'};
            app.PButton_2.Position = [1215 34 14 22];
            app.PButton_2.Text = 'P';

            % Create PButton
            app.PButton = uibutton(app.DataVisualisationTab, 'push');
            app.PButton.ButtonPushedFcn = createCallbackFcn(app, @PButtonPushed, true);
            app.PButton.FontSize = 8;
            app.PButton.FontWeight = 'bold';
            app.PButton.Tooltip = {'Popout graph for additional editability and functionality'};
            app.PButton.Position = [1215 59 14 22];
            app.PButton.Text = 'P';

            % Create PButton_3
            app.PButton_3 = uibutton(app.DataVisualisationTab, 'push');
            app.PButton_3.ButtonPushedFcn = createCallbackFcn(app, @PButton_3Pushed, true);
            app.PButton_3.FontSize = 8;
            app.PButton_3.FontWeight = 'bold';
            app.PButton_3.Tooltip = {'Popout graph for additional editability and functionality'};
            app.PButton_3.Position = [1079 34 14 22];
            app.PButton_3.Text = 'P';

            % Create ResultsTable
            app.ResultsTable = uitab(app.TabGroup);
            app.ResultsTable.Title = 'Results Table';
            app.ResultsTable.BackgroundColor = [1 1 1];

            % Create TableResults
            app.TableResults = uitable(app.ResultsTable);
            app.TableResults.ColumnName = {'Column 1'; 'Column 2'; 'Column 3'; 'Column 4'};
            app.TableResults.RowName = {};
            app.TableResults.Position = [14 50 1294 846];

            % Create UpdateTableButton
            app.UpdateTableButton = uibutton(app.ResultsTable, 'push');
            app.UpdateTableButton.ButtonPushedFcn = createCallbackFcn(app, @UpdateTableButtonPushed, true);
            app.UpdateTableButton.Position = [1173 13 100 22];
            app.UpdateTableButton.Text = 'Update Table';

            % Create ExamplesTab
            app.ExamplesTab = uitab(app.TabGroup);
            app.ExamplesTab.Title = 'Examples';

            % Show the figure after all components are created
            app.DoEPreProcessingBenchmarkToolboxUIFigure.Visible = 'on';
        end
    end

    % App creation and deletion
    methods (Access = public)

        % Construct app
        function app = outlierdetectionFunctional190523

            % Create UIFigure and components
            createComponents(app)

            % Register the app with App Designer
            registerApp(app, app.DoEPreProcessingBenchmarkToolboxUIFigure)

            if nargout == 0
                clear app
            end
        end

        % Code that executes before app deletion
        function delete(app)

            % Delete UIFigure when app is deleted
            delete(app.DoEPreProcessingBenchmarkToolboxUIFigure)
        end
    end
end
```

Appendix B

Appendix Chapter 3: Spectra

This appendix contains all representative spectra obtained during the paracetamol impurity screening process. All measurements have been performed using the Bi_3^+ ion source in spectrometry mode at 100x100 µm^2 and 128x128 pixels. The spectra displayed here are based on the original .pdf outputs and might contain misspellings, however the information described above should account for these.

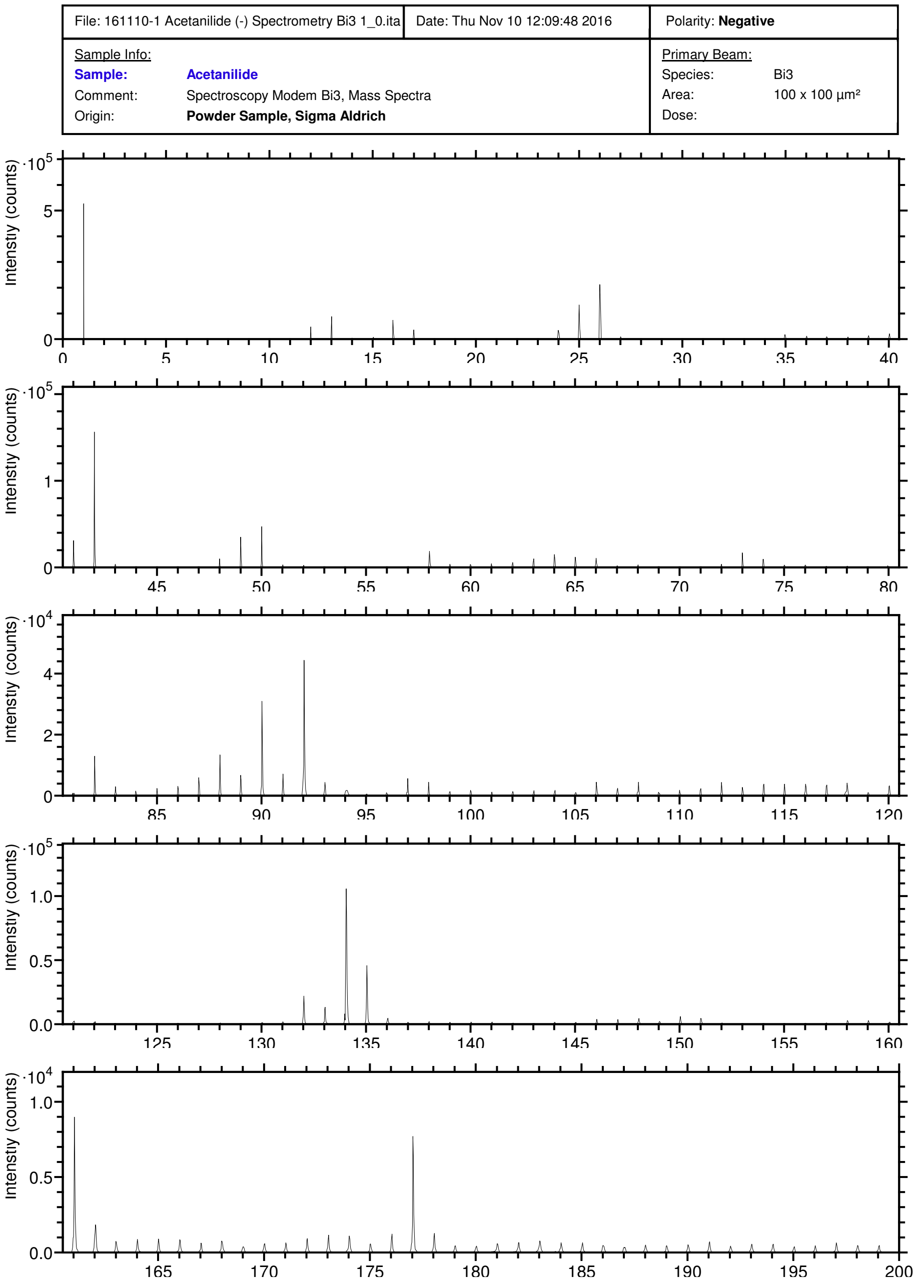


File: 161110-1 Acetanilide (+) Spectrometry Bi3 1_0.ita | Date: Thu Nov 10 12:00:29 2016 | Polarity: **Positive**

Sample Info:
Sample: **Acetanilide**
Comment: Spectroscopy Mode, Bi3, Mass Spectra
Origin: **Powder Sample, Sigma Aldrich**

Primary Beam:
Species: Bi3
Area: 100 x 100 µm²
Dose:

Figure B.1: Acetanilide spectrum - positive ion mode.

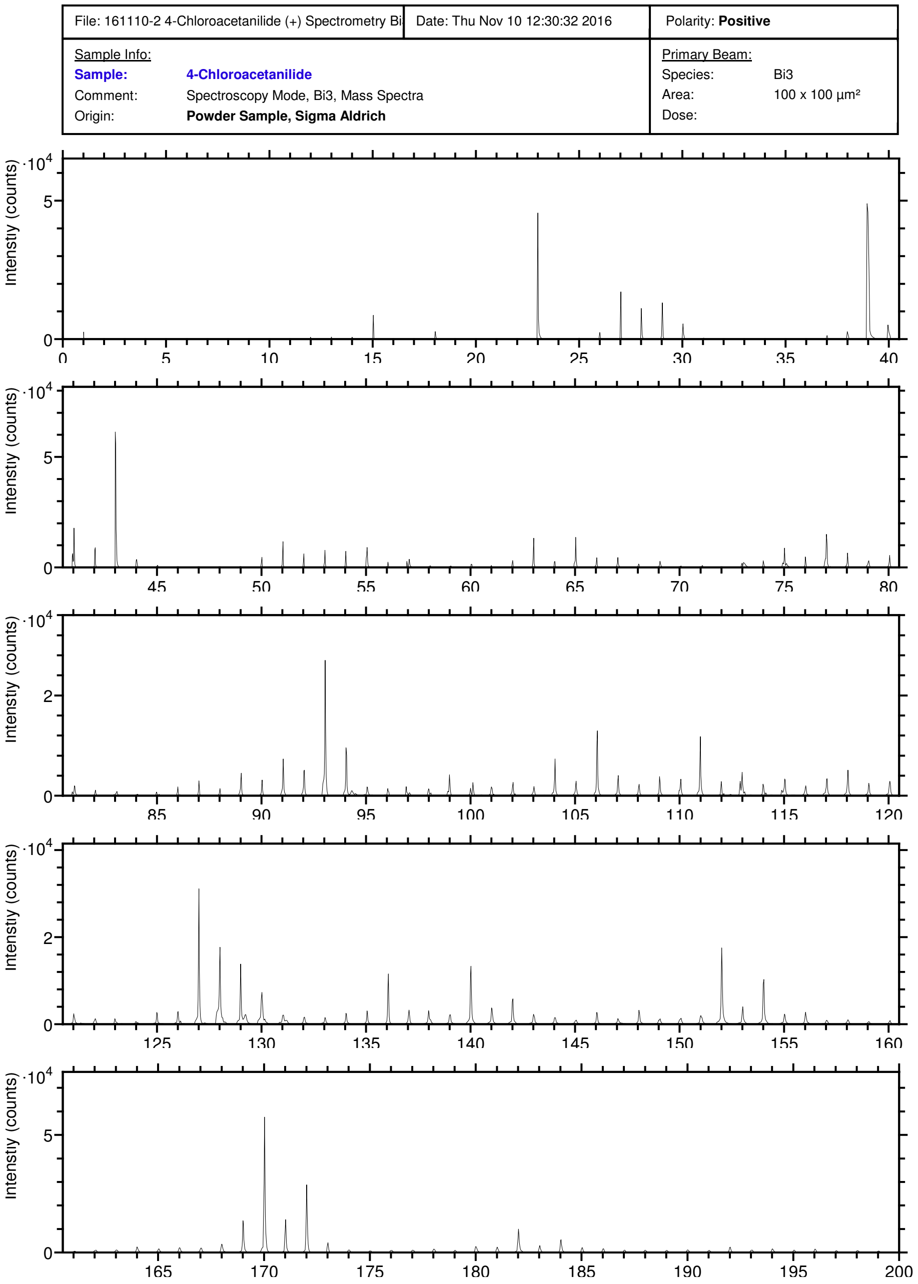


File: 161110-1 Acetanilide (-) Spectrometry Bi3 1_0.ita | Date: Thu Nov 10 12:09:48 2016 | Polarity: **Negative**

Sample Info:
Sample: **Acetanilide**
Comment: Spectroscopy Modem Bi3, Mass Spectra
Origin: **Powder Sample, Sigma Aldrich**

Primary Beam:
Species: Bi3
Area: 100 x 100 µm²
Dose:

Figure B.2: Acetanilide spectrum - negative ion mode.

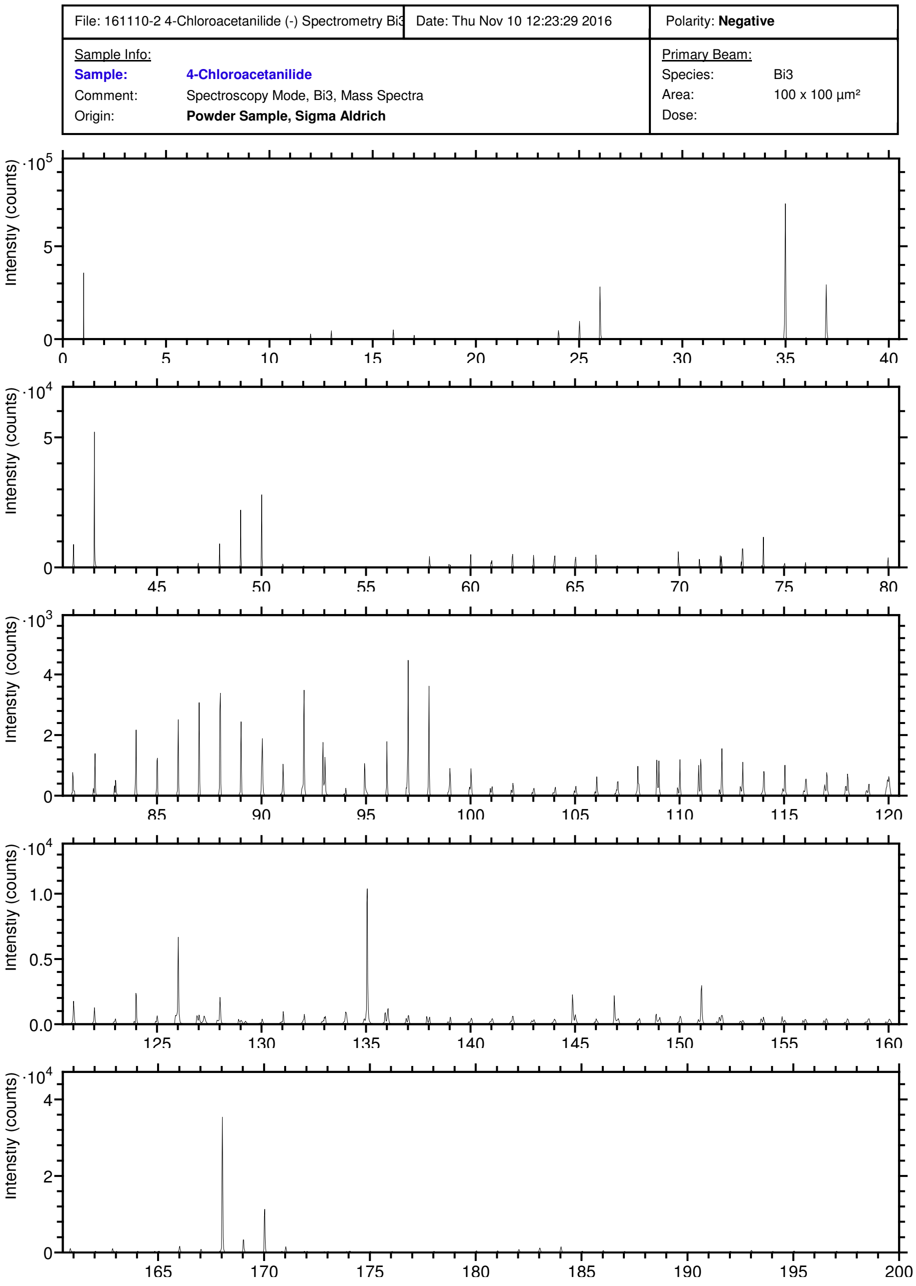


File: 161110-2 4-Chloroacetanilide (+) Spectrometry Bi | Date: Thu Nov 10 12:30:32 2016 | Polarity: **Positive**

Sample Info:
Sample: **4-Chloroacetanilide**
Comment: Spectroscopy Mode, Bi3, Mass Spectra
Origin: **Powder Sample, Sigma Aldrich**

Primary Beam:
Species: Bi3
Area: 100 x 100 µm²
Dose:

Figure B.3: 4-Chloroacetanilide spectrum - positive ion mode.

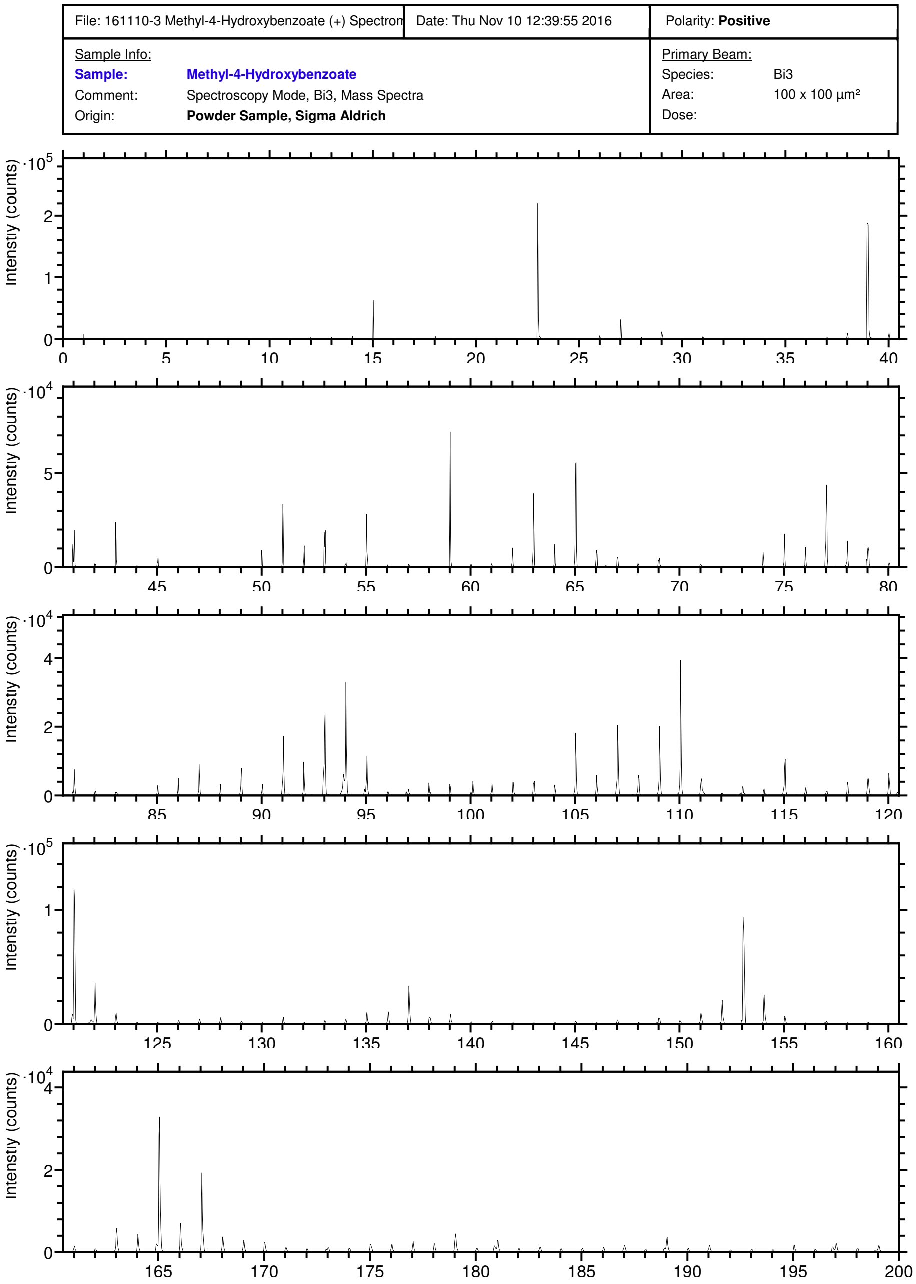


File: 161110-2 4-Chloroacetanilide (-) Spectrometry Bi3 | Date: Thu Nov 10 12:23:29 2016 | Polarity: **Negative**

Sample Info:
Sample: **4-Chloroacetanilide**
Comment: Spectroscopy Mode, Bi3, Mass Spectra
Origin: **Powder Sample, Sigma Aldrich**

Primary Beam:
Species: Bi3
Area: 100 x 100 µm²
Dose:

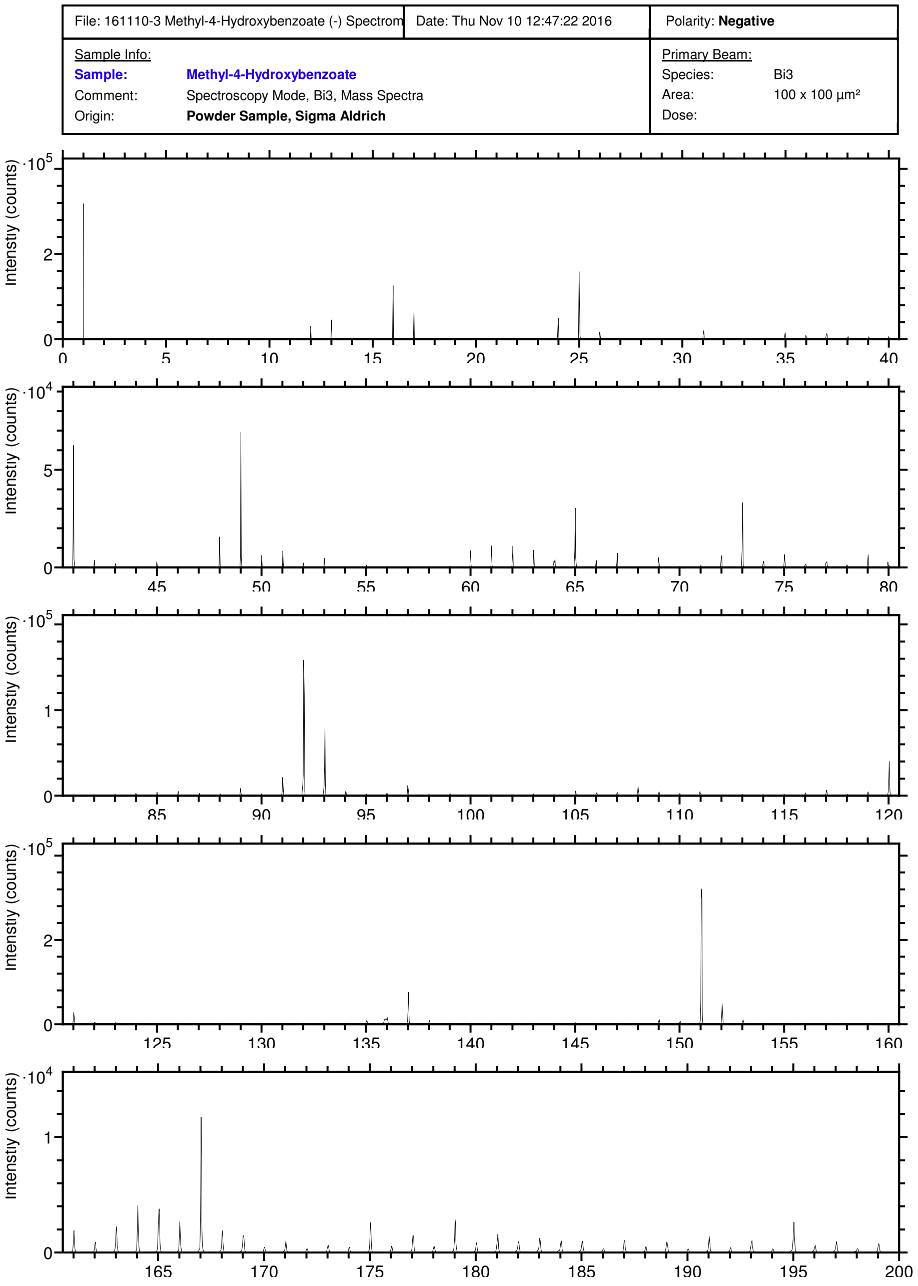
Figure B.4: 4-Chloroacetanilide spectrum - negative ion mode.



File: 161110-3 Methyl-4-Hydroxybenzoate (+) Spectron	Date: Thu Nov 10 12:39:55 2016	Polarity: **Positive**

Sample Info:		Primary Beam:	
Sample:	**Methyl-4-Hydroxybenzoate**	Species:	Bi3
Comment:	Spectroscopy Mode, Bi3, Mass Spectra	Area:	100 x 100 µm²
Origin:	**Powder Sample, Sigma Aldrich**	Dose:	

Figure B.5: Methyl-4-Hydroxybenzoate spectrum - positive ion mode.

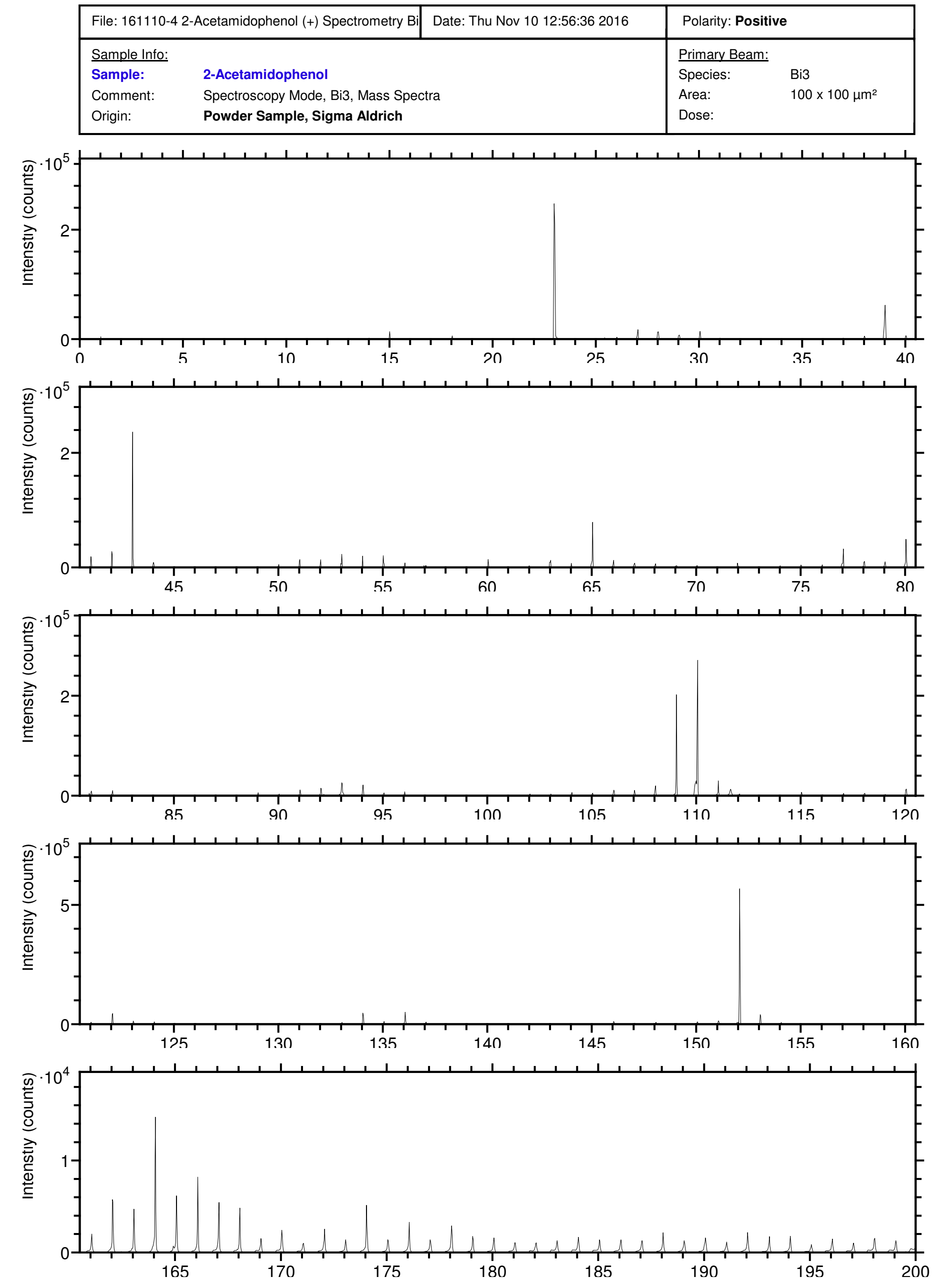


File: 161110-3 Methyl-4-Hydroxybenzoate (-) Spectrom | Date: Thu Nov 10 12:47:22 2016 | Polarity: **Negative**

Sample Info:
Sample: **Methyl-4-Hydroxybenzoate**
Comment: Spectroscopy Mode, Bi3, Mass Spectra
Origin: **Powder Sample, Sigma Aldrich**

Primary Beam:
Species: Bi3
Area: 100 x 100 µm²
Dose:

Figure B.6: Methyl-4-Hydroxybenzoate spectrum - negative ion mode.

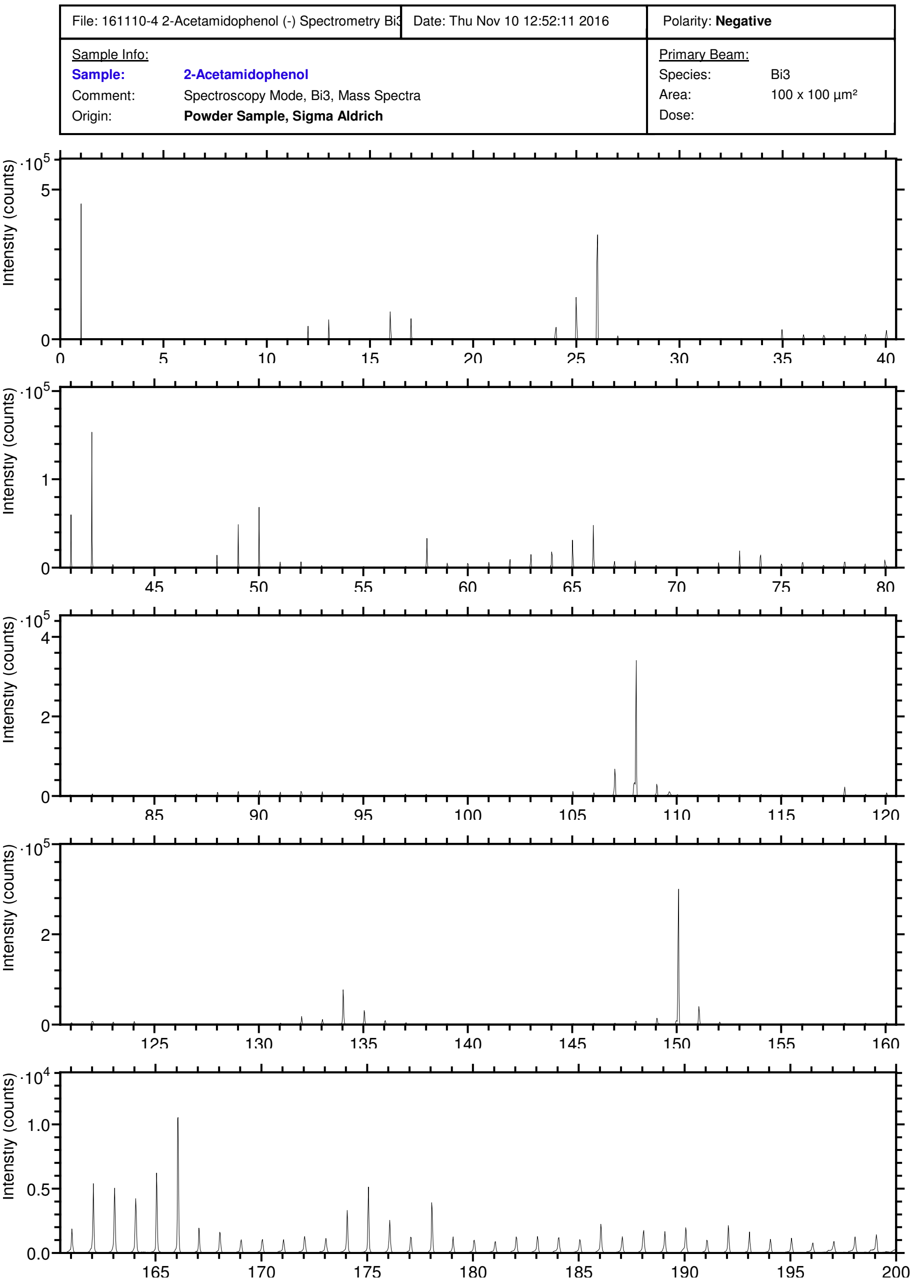


File: 161110-4 2-Acetamidophenol (+) Spectrometry Bi | Date: Thu Nov 10 12:56:36 2016 | Polarity: **Positive**

Sample Info:
Sample: **2-Acetamidophenol**
Comment: Spectroscopy Mode, Bi3, Mass Spectra
Origin: **Powder Sample, Sigma Aldrich**

Primary Beam:
Species: Bi3
Area: 100 x 100 µm²
Dose:

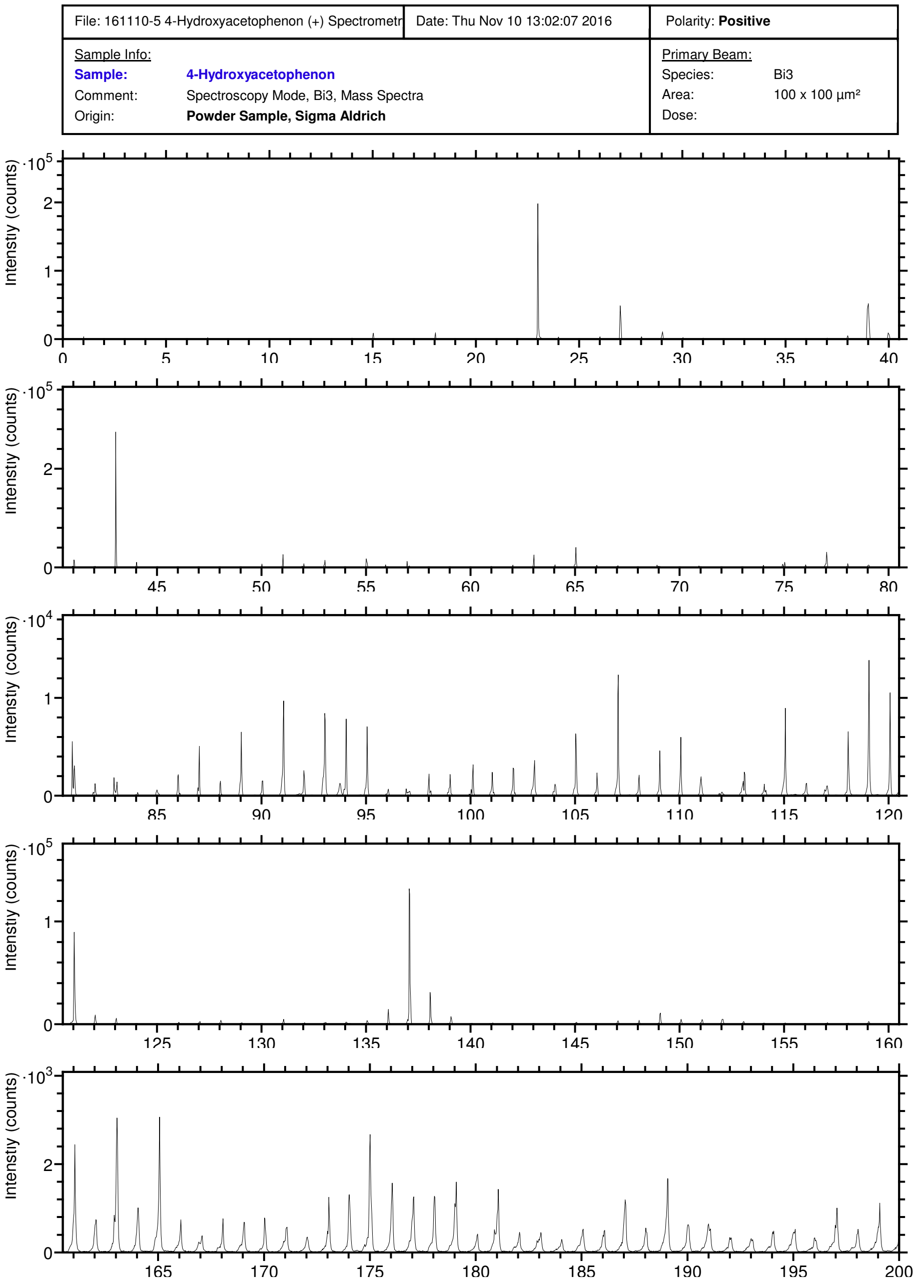
Figure B.7: 4-Chloroacetanilide spectrum - positive ion mode.



File: 161110-4 2-Acetamidophenol (-) Spectrometry Bi3	Date: Thu Nov 10 12:52:11 2016	Polarity: **Negative**

Sample Info:		Primary Beam:	
Sample:	**2-Acetamidophenol**	Species:	Bi3
Comment:	Spectroscopy Mode, Bi3, Mass Spectra	Area:	100 x 100 µm²
Origin:	**Powder Sample, Sigma Aldrich**	Dose:	

Figure B.8: 4-Chloroacetanilide spectrum - negative ion mode.

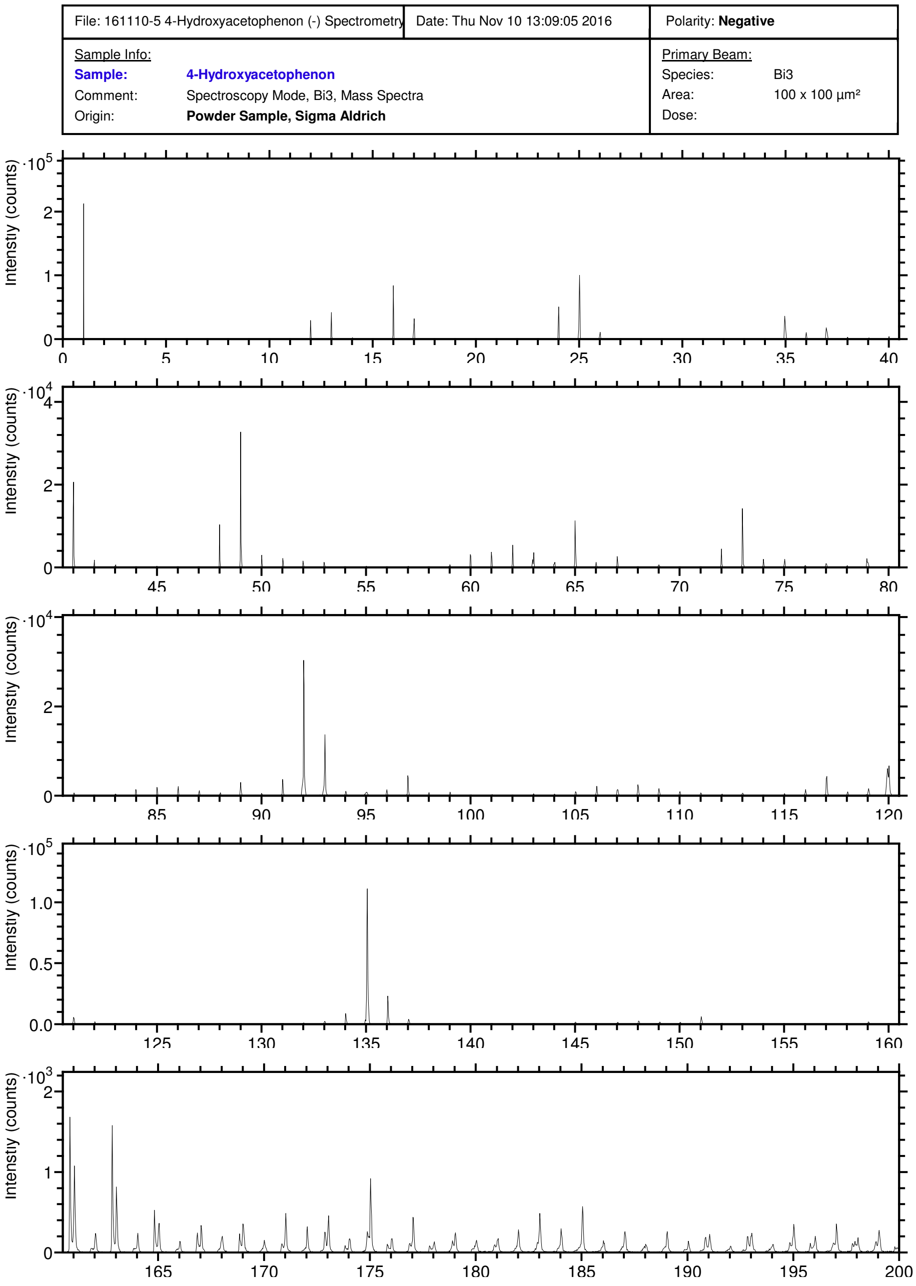


File: 161110-5 4-Hydroxyacetophenon (+) Spectrometr | Date: Thu Nov 10 13:02:07 2016 | Polarity: **Positive**

Sample Info:
Sample: **4-Hydroxyacetophenon**
Comment: Spectroscopy Mode, Bi3, Mass Spectra
Origin: **Powder Sample, Sigma Aldrich**

Primary Beam:
Species: Bi3
Area: 100 x 100 µm²
Dose:

Figure B.9: 4-Hydroxyacetophenon spectrum - positive ion mode.

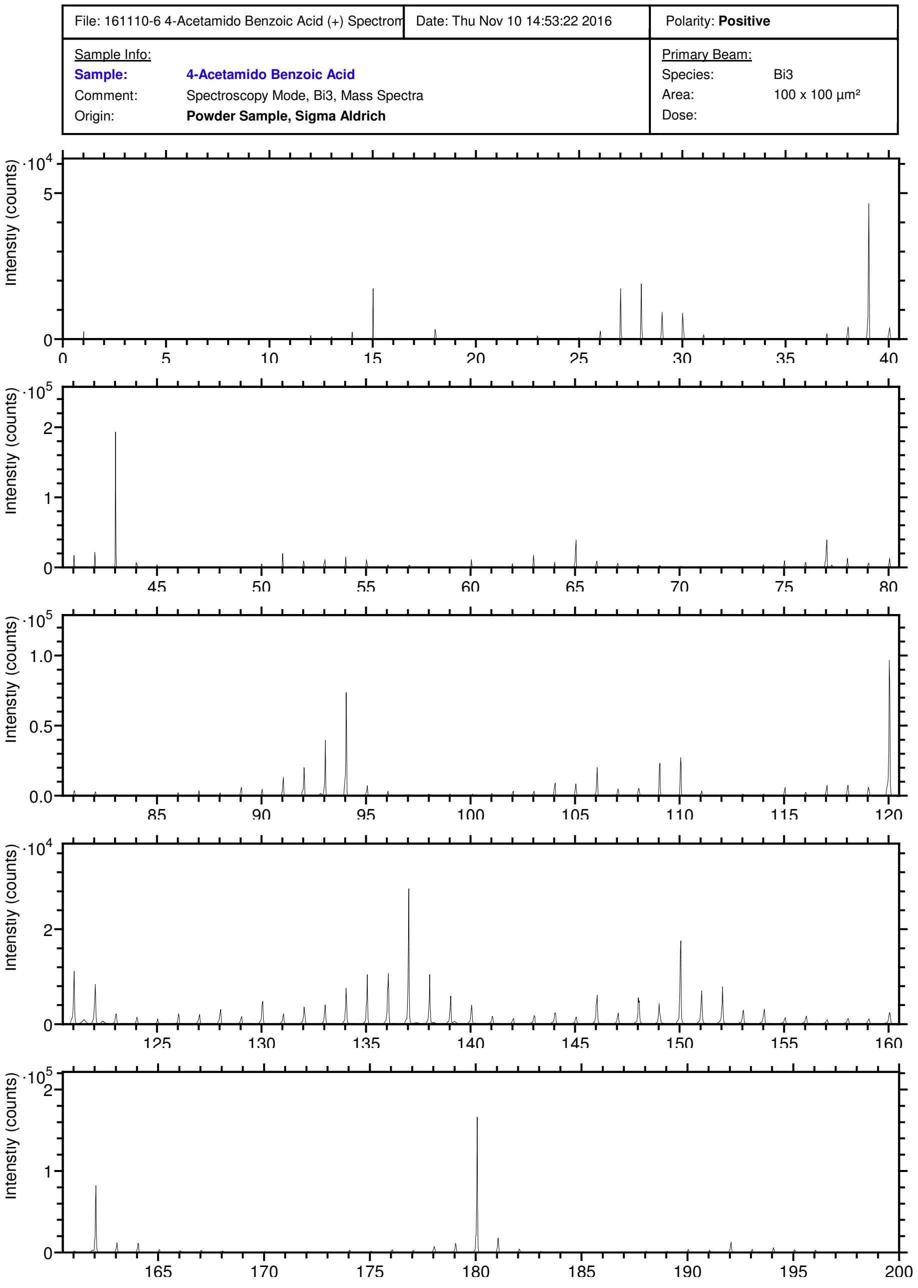


File: 161110-5 4-Hydroxyacetophenon (-) Spectrometry | Date: Thu Nov 10 13:09:05 2016 | Polarity: **Negative**

Sample Info:
Sample: **4-Hydroxyacetophenon**
Comment: Spectroscopy Mode, Bi3, Mass Spectra
Origin: **Powder Sample, Sigma Aldrich**

Primary Beam:
Species: Bi3
Area: 100 x 100 µm²
Dose:

Figure B.10: 4-Hydroxyacetophenon spectrum - negative ion mode.

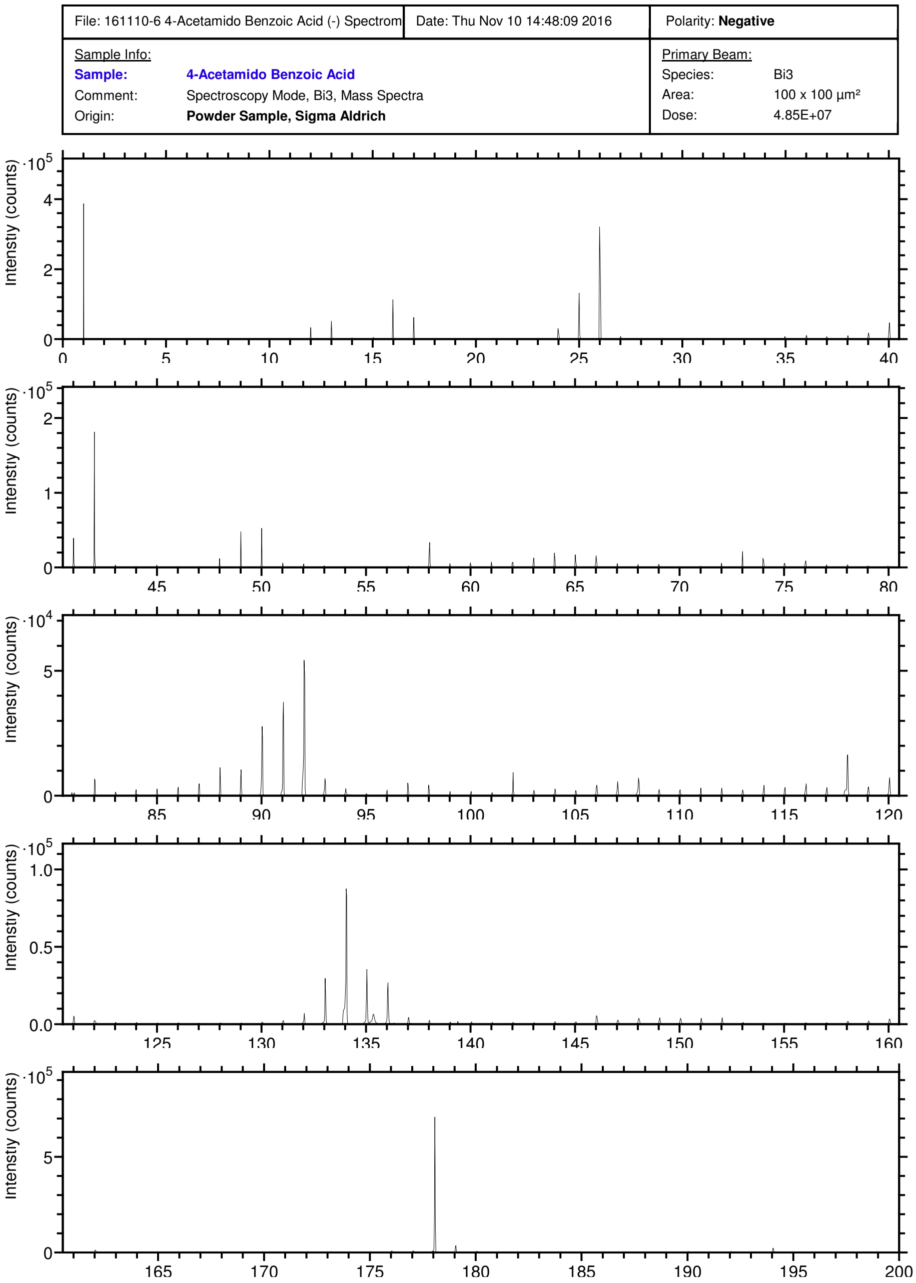


File: 161110-6 4-Acetamido Benzoic Acid (+) Spectrom
Date: Thu Nov 10 14:53:22 2016
Polarity: **Positive**

Sample Info:
Sample: **4-Acetamido Benzoic Acid**
Comment: Spectroscopy Mode, Bi3, Mass Spectra
Origin: **Powder Sample, Sigma Aldrich**

Primary Beam:
Species: Bi3
Area: 100 x 100 µm²
Dose:

Figure B.11: Acetamidobenzoic acid spectrum - positive ion mode.



File: 161110-6 4-Acetamido Benzoic Acid (-) Spectrom	Date: Thu Nov 10 14:48:09 2016	Polarity: **Negative**

Sample Info:

Sample: **4-Acetamido Benzoic Acid**

Comment: Spectroscopy Mode, Bi3, Mass Spectra

Origin: **Powder Sample, Sigma Aldrich**

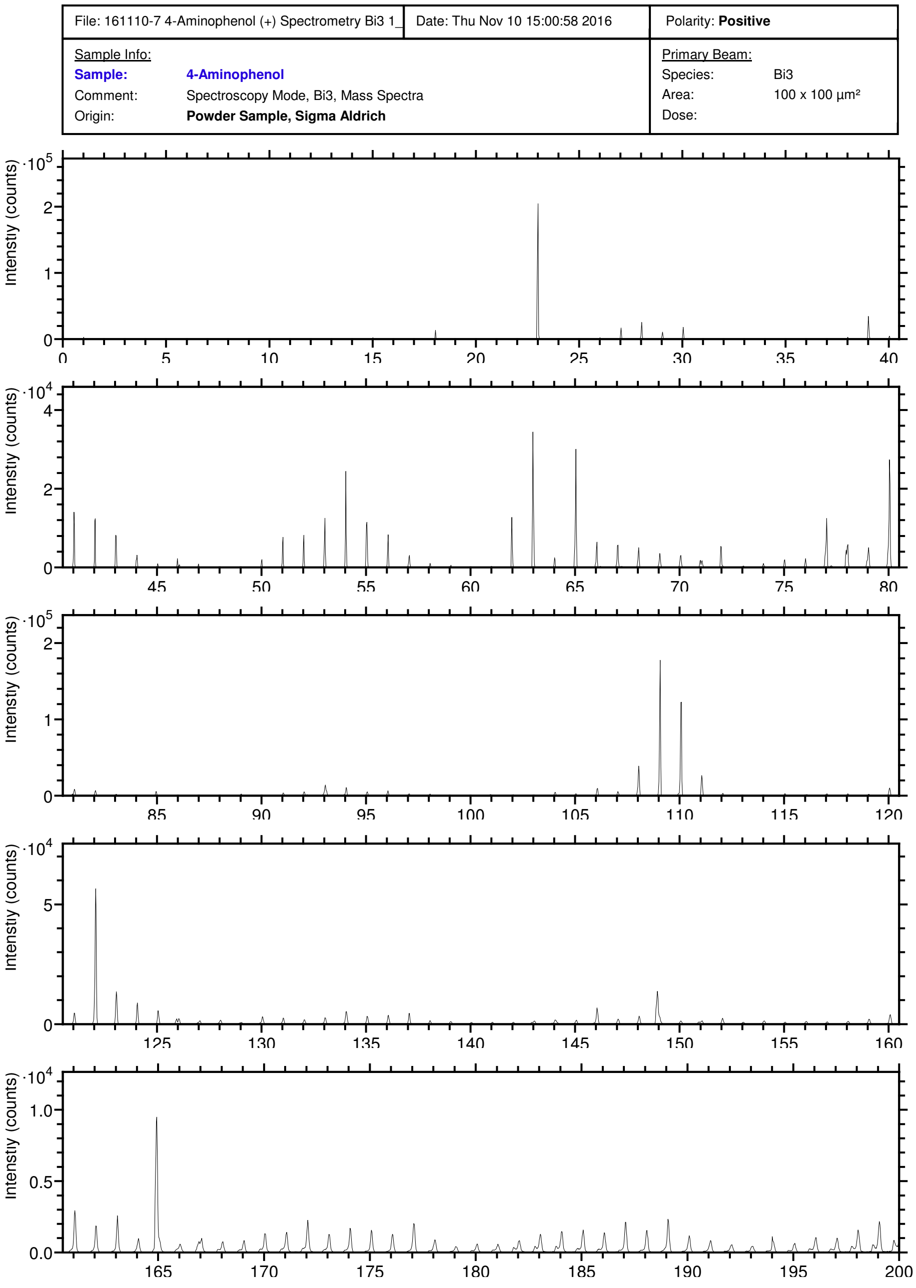
Primary Beam:

Species: Bi3

Area: 100 x 100 µm²

Dose: 4.85E+07

Figure B.12: Acetamidobenzoic acid spectrum - negative ion mode.



File: 161110-7 4-Aminophenol (+) Spectrometry Bi3 1_	Date: Thu Nov 10 15:00:58 2016	Polarity: **Positive**

Sample Info:

Sample: **4-Aminophenol**

Comment: Spectroscopy Mode, Bi3, Mass Spectra

Origin: **Powder Sample, Sigma Aldrich**

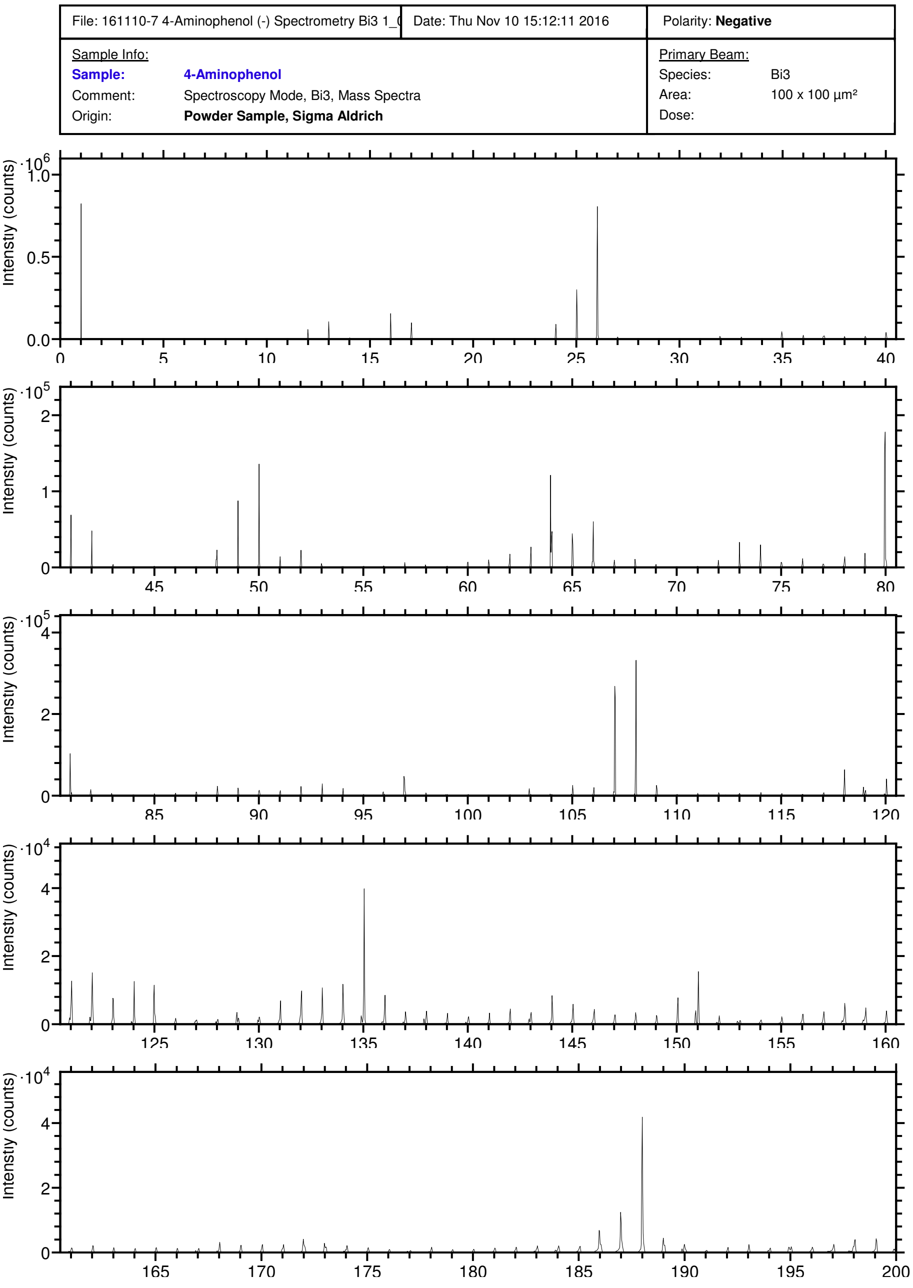
Primary Beam:

Species: Bi3

Area: 100 x 100 µm²

Dose:

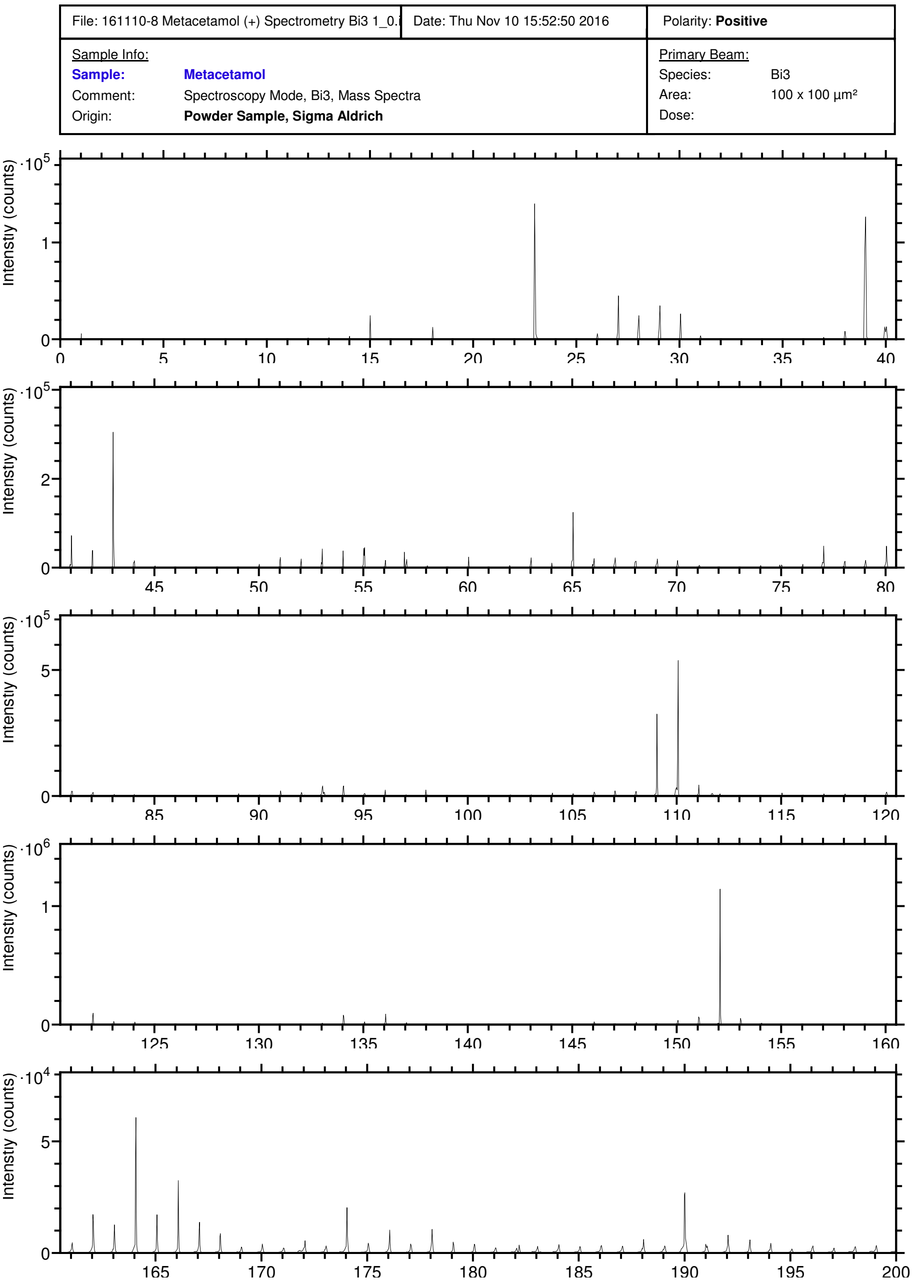
Figure B.13: 4-Aminophenol spectrum - positive ion mode.



File: 161110-7 4-Aminophenol (-) Spectrometry Bi3 1_(	Date: Thu Nov 10 15:12:11 2016	Polarity: **Negative**

Sample Info:		Primary Beam:	
Sample:	**4-Aminophenol**	Species:	Bi3
Comment:	Spectroscopy Mode, Bi3, Mass Spectra	Area:	100 x 100 µm²
Origin:	**Powder Sample, Sigma Aldrich**	Dose:	

Figure B.14: 4-Aminophenol spectrum - negative ion mode.

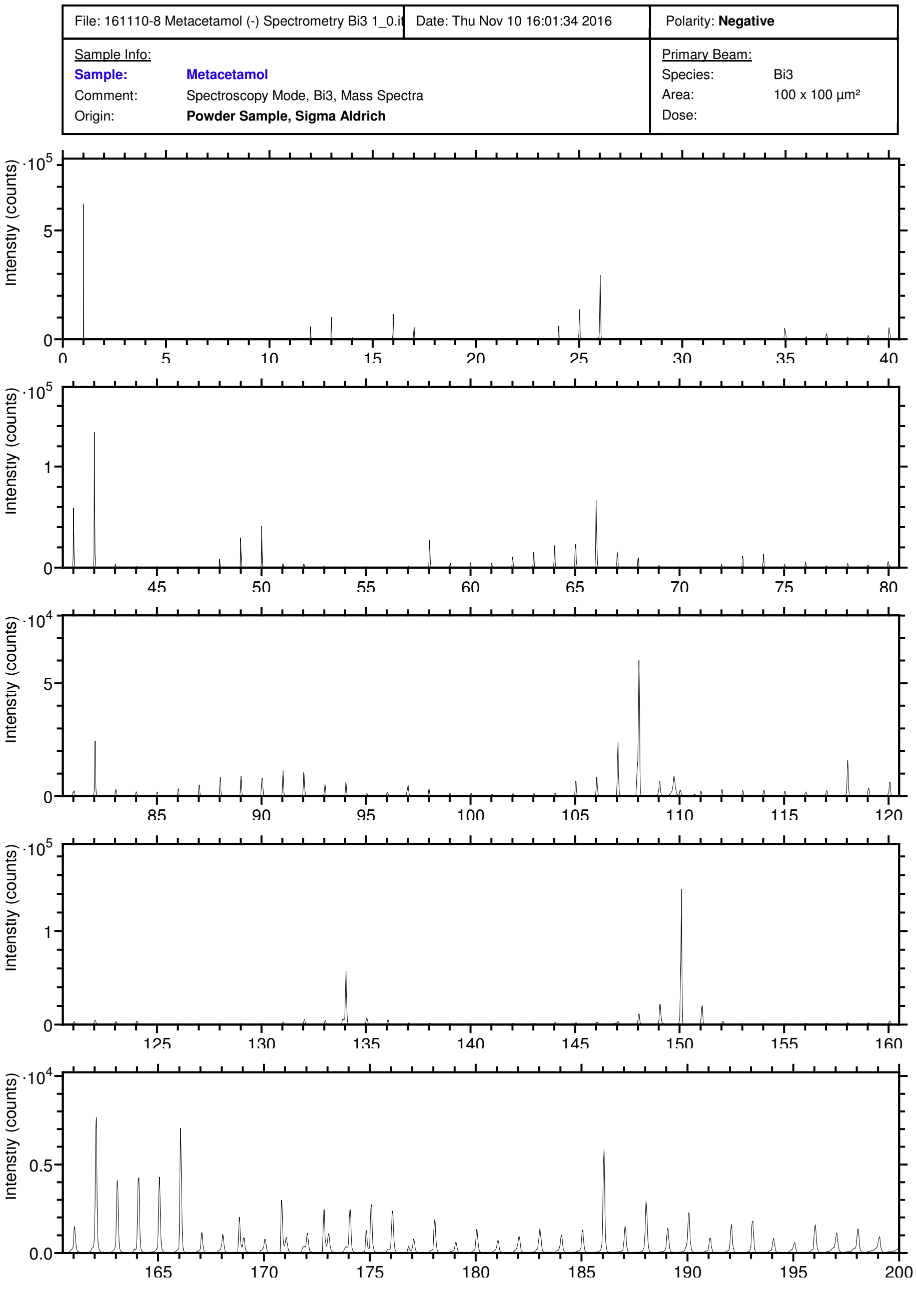


File: 161110-8 Metacetamol (+) Spectrometry Bi3 1_0.i	Date: Thu Nov 10 15:52:50 2016	Polarity: **Positive**

Sample Info:
Sample: **Metacetamol**
Comment: Spectroscopy Mode, Bi3, Mass Spectra
Origin: **Powder Sample, Sigma Aldrich**

Primary Beam:
Species: Bi3
Area: 100 x 100 µm²
Dose:

Figure B.15: Metacetamol spectrum - positive ion mode.

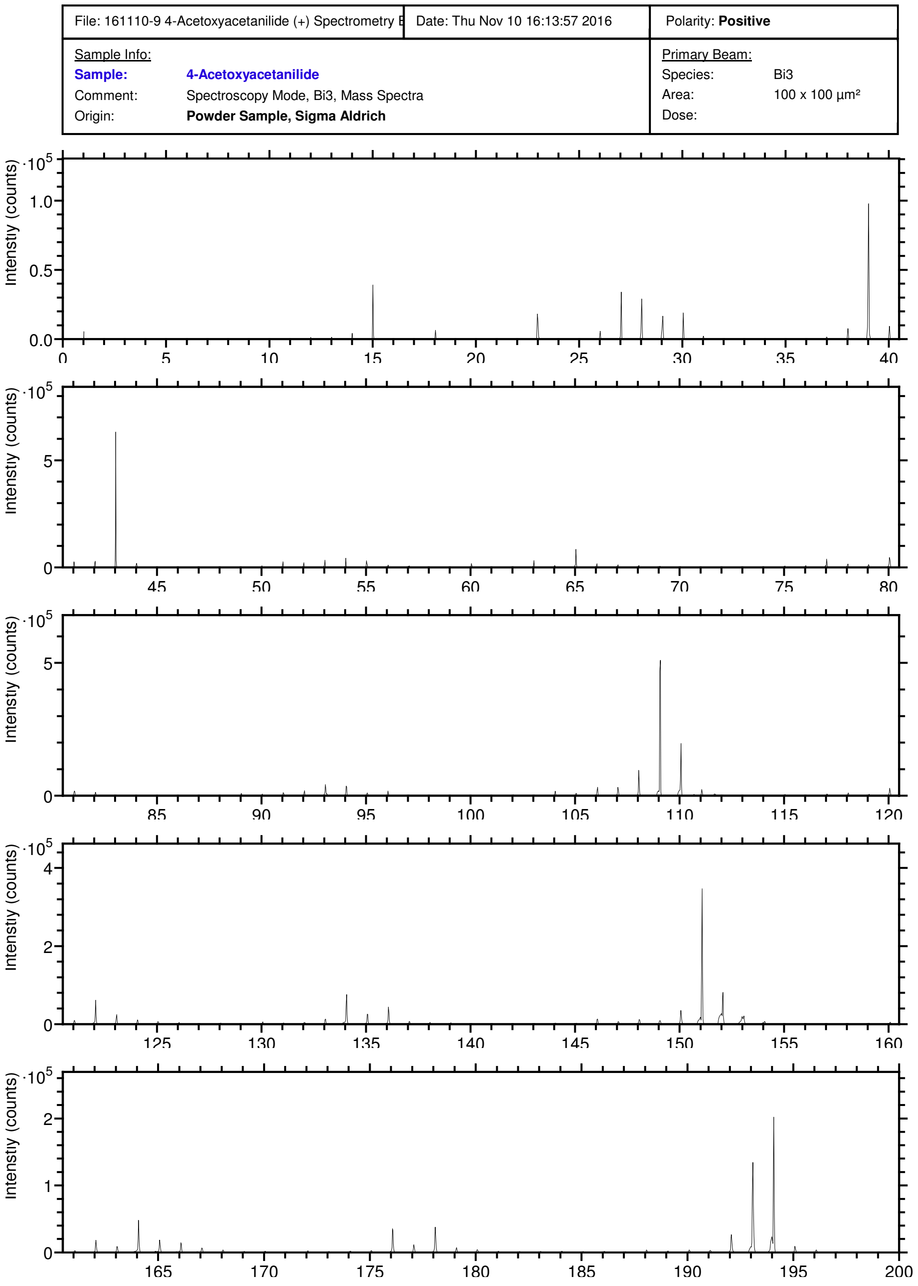


File: 161110-8 Metacetamol (-) Spectrometry Bi3 1_0.it | Date: Thu Nov 10 16:01:34 2016 | Polarity: **Negative**

Sample Info:
Sample: **Metacetamol**
Comment: Spectroscopy Mode, Bi3, Mass Spectra
Origin: **Powder Sample, Sigma Aldrich**

Primary Beam:
Species: Bi3
Area: 100 x 100 µm²
Dose:

Figure B.16: Metacetamol spectrum - negative ion mode.



File: 161110-9 4-Acetoxyacetanilide (+) Spectrometry E

Date: Thu Nov 10 16:13:57 2016

Polarity: **Positive**

Sample Info:

Sample: **4-Acetoxyacetanilide**

Comment: Spectroscopy Mode, Bi3, Mass Spectra

Origin: **Powder Sample, Sigma Aldrich**

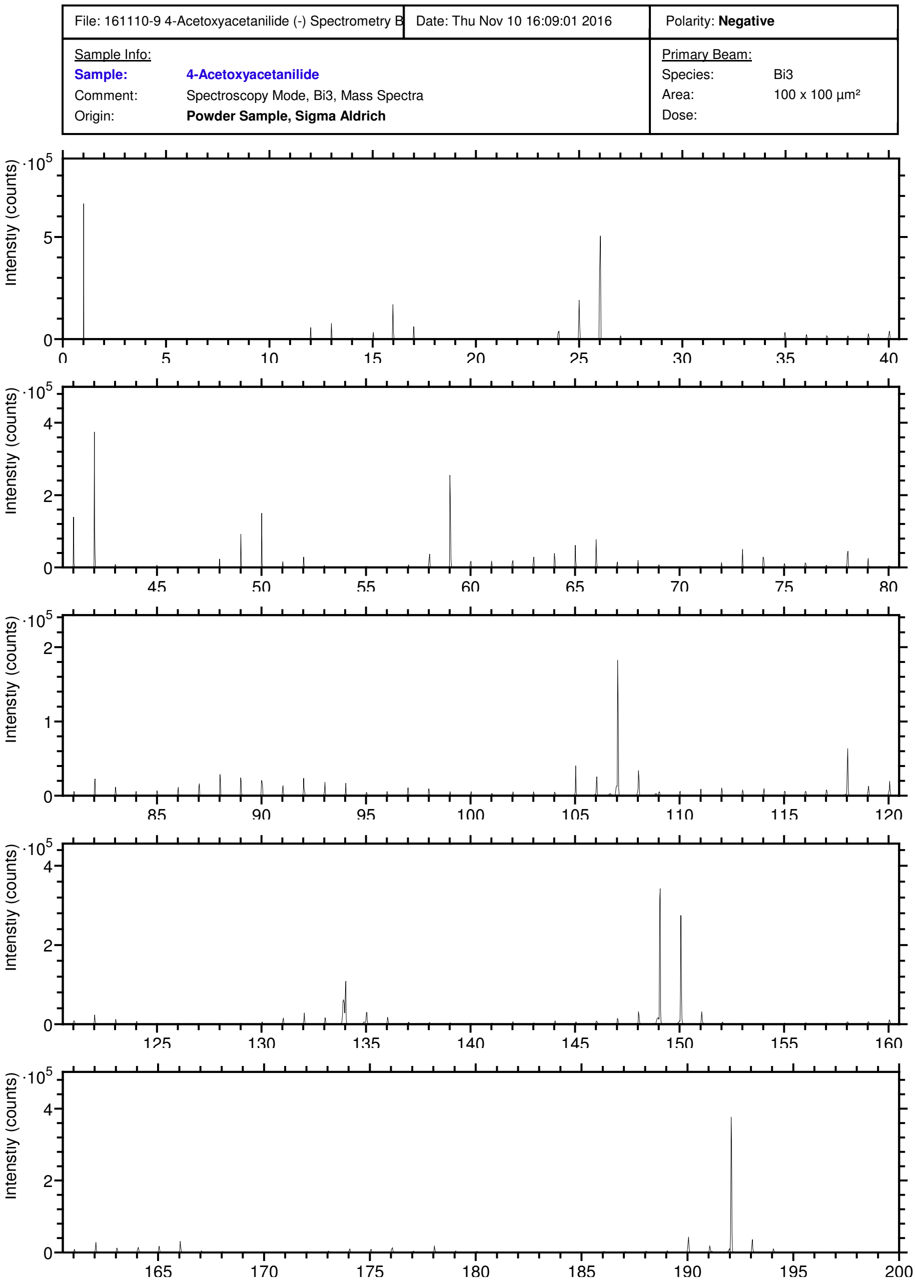
Primary Beam:

Species: Bi3

Area: 100 x 100 µm²

Dose:

Figure B.17: 4-Acetoxyacetanilide spectrum - positive ion mode.

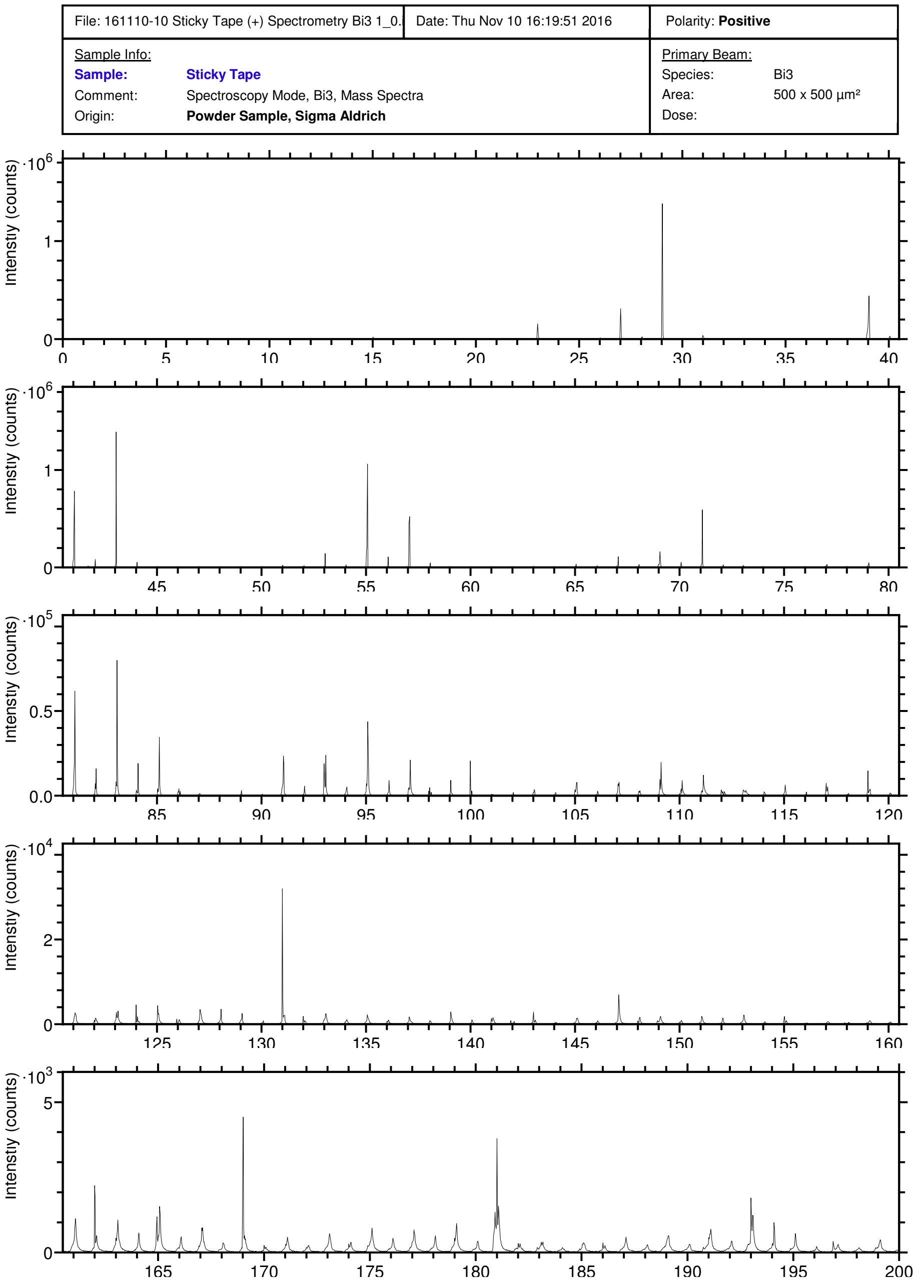


File: 161110-9 4-Acetoxyacetanilide (-) Spectrometry B | Date: Thu Nov 10 16:09:01 2016 | Polarity: **Negative**

Sample Info:
Sample: **4-Acetoxyacetanilide**
Comment: Spectroscopy Mode, Bi3, Mass Spectra
Origin: **Powder Sample, Sigma Aldrich**

Primary Beam:
Species: Bi3
Area: 100 x 100 µm²
Dose:

Figure B.18: 4-Acetoxyacetanilide spectrum - negative ion mode.

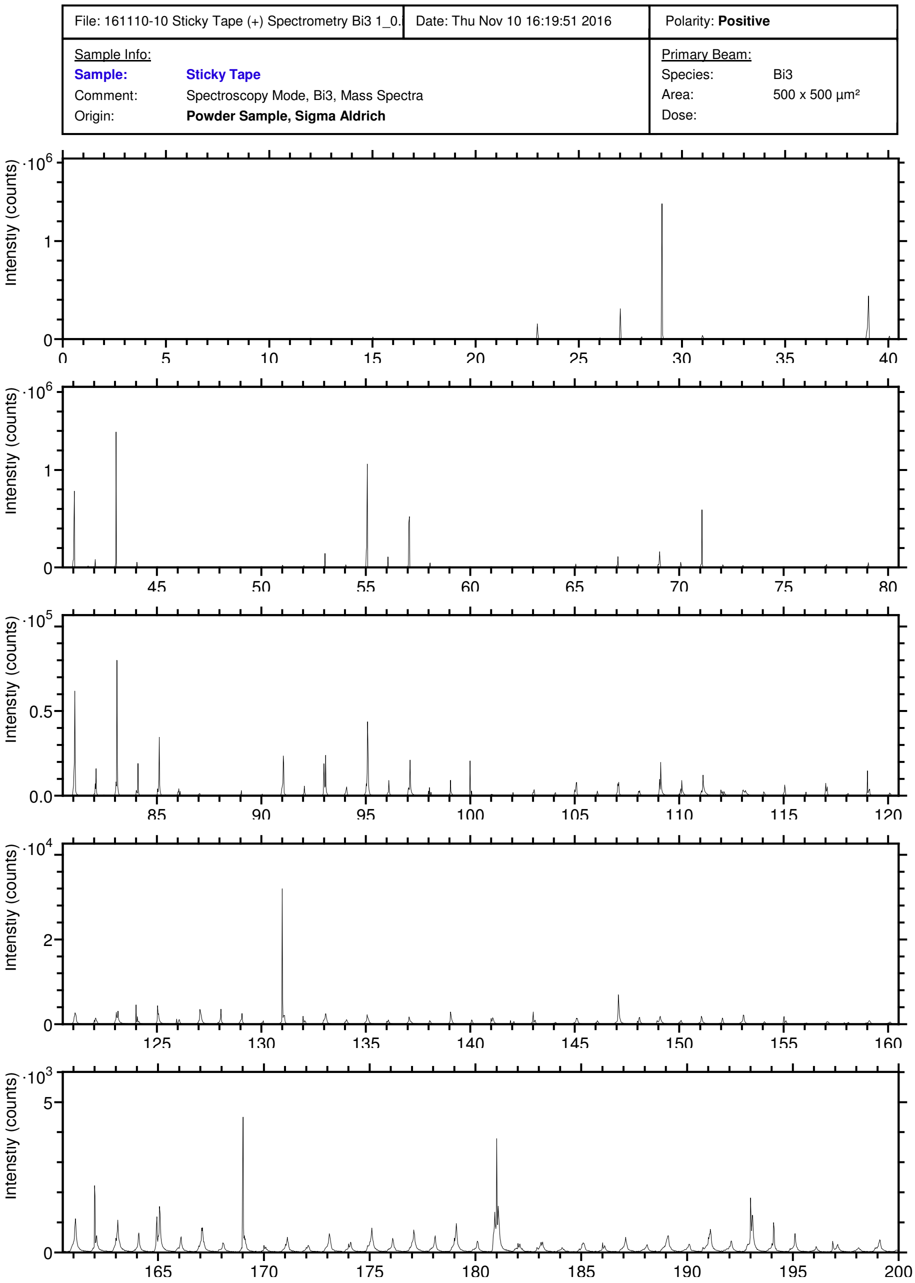


File: 161110-10 Sticky Tape (+) Spectrometry Bi3 1_0. Date: Thu Nov 10 16:19:51 2016 Polarity: **Positive**

Sample Info:
Sample: **Sticky Tape**
Comment: Spectroscopy Mode, Bi3, Mass Spectra
Origin: **Powder Sample, Sigma Aldrich**

Primary Beam:
Species: Bi3
Area: 500 x 500 µm²
Dose:

Figure B.19: Sticky tape spectrum - positive ion mode.



File: 161110-10 Sticky Tape (+) Spectrometry Bi3 1_0.	Date: Thu Nov 10 16:19:51 2016	Polarity: **Positive**

Sample Info:		Primary Beam:	
Sample:	**Sticky Tape**	Species:	Bi3
Comment:	Spectroscopy Mode, Bi3, Mass Spectra	Area:	500 x 500 µm²
Origin:	**Powder Sample, Sigma Aldrich**	Dose:	

Figure B.20: Sticky tape spectrum - negative ion mode.

Appendix C

Appendix Chapter 4: Tables and Ion Image

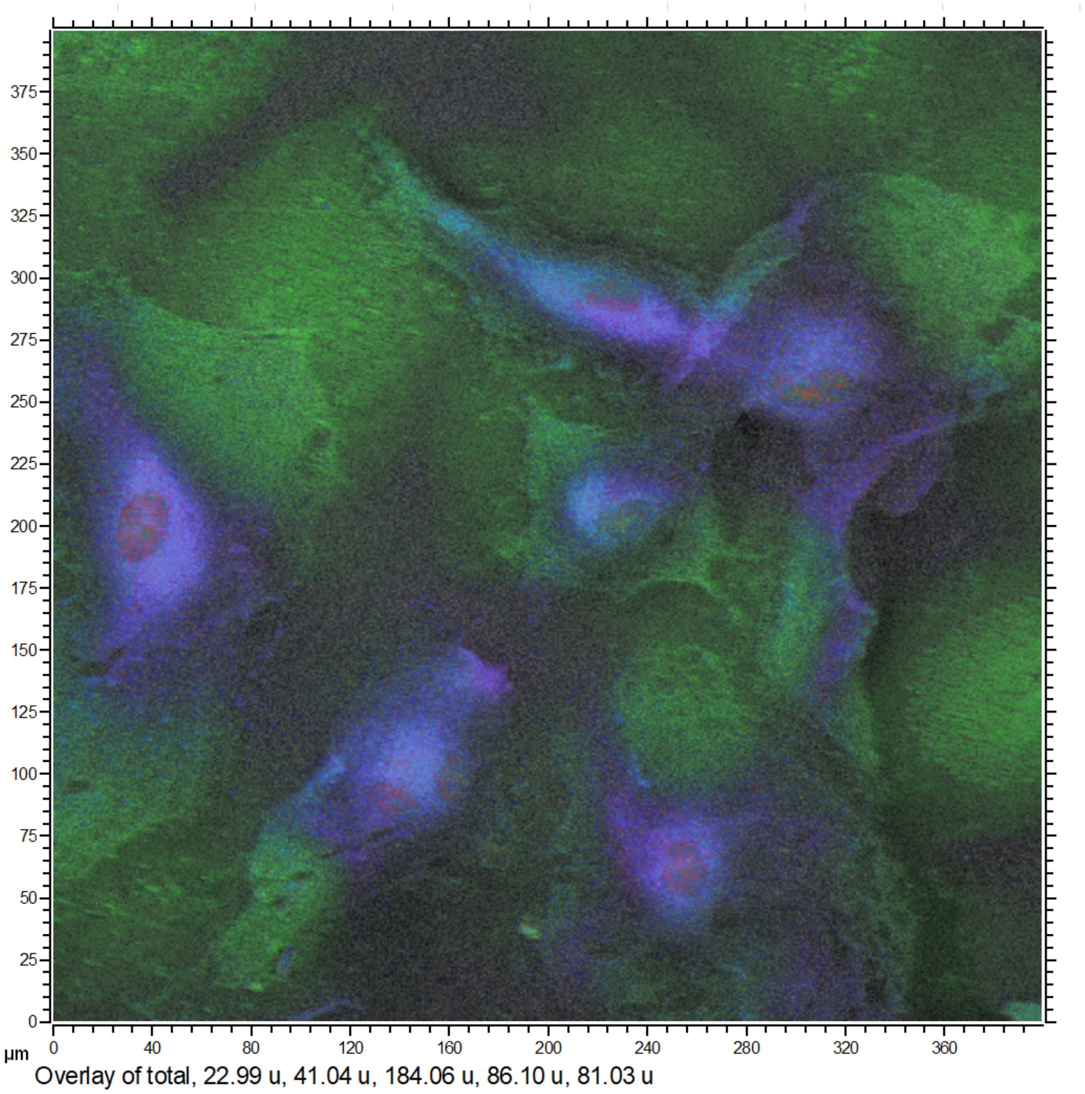


Figure C.1: Ion overlay image of sputtered cells after application of an $Ar_{1500}{}^{+}$ sputter beam and a $Bi_3{}^{+}$ analysis beam. The cell samples were sputtered to completion. The green and blue colours are salt ion related (m/z 23 (sodium) and m/z 41 (^{41}potassium)) with pink and turquoise being cell-membrane related peaks (m/z 86 and m/z 184). Lastly, red signifies the ribose nuclear marker at m/z 81.

Table C.1: Positive ion table with likely peak assignments.

Observed Mass (m/z)	**Molecular Formula ($^+$)**	**Theoretical Mass (m/z)**	**Assignment**
18.04	NH_4	18.0338	Ammonium
23.00	Na	22.9892	Sodium
30.04	CH_4N	30.0338	Glycine Fragment
38.97	K	38.9632	Potassium
39.97	Ca	39.9620	Calcium
40.96	^{41}K	40.9613	Potassium-41
42.04	C_2H_4N	42.0338	Alanine Fragment
44.05	C_2H_6N	44.0495	Alanine Fragment
54.04	C_3H_4N	54.0338	Valine, Leucine Fragment
56.05	C_3H_6N	56.0495	Valine, Leucine-, Isoleucine Fragment
58.07	C_3H_8N	58.0651	PCH-, Glutamic Acid
60.05	C_2H_6NO	60.0444	I-Serine fragment
68.05	C_4H_6N	68.0495	Proline fragment
70.07	C_4H_88N	70.0651	Proline fragment
80.06	C_5H_6N	80.0495	Leucine, Isoleucine
81.02	C_5H_55O	81.0335	DNA ribose sugar
82.07	C_5H_88N	82.0651	Histidine fragment
84.04	C_4H_6NO	84.0444	Glutamic acid fragment
84.08	$C_5H_{10}N$	84.0808	Lysine fragment
110.08	$C_5H_8N^3$	110.0713	Arginine, Histidine
120.08	$C_8H_{10}N$	120.0808	Phenylalanine fragment
125.00	$C_2H_6O_4P$	124.9998	
166.06	$C_5H_{13}NO_3P$	166.0628	PCH fragment
184.09	$C_5H_{15}NO_4P$	184.0733	PCH fragment
224.11	$C_8H_{19}NO_4P$	224.1046	PCH fragment
369.35	$C_{27}H_{45}$	369.3516	Cholesterol fragment
385.34	$C_{27}H_{45}O$	385.3465	Cholesterol fragment

Table C.2: Negative ion table with likely peak assignments.

Observed Mass (m/z)	**Molecular Formula ($^-$)**	**Theoretical Mass (m/z)**	**Assignment**
13.01	CH	13.0084	
24.00	C_2	24.0005	
25.01	C_2H	25.0084	
26.01	CN	26.0036	
27.03	C_2H_3	27.024	
30.97	P	30.9743	
31.02	CH_3O	31.0189	
31.98	S	31.9726	
32.98	HS	32.9804	
34.97	Cl	34.9694	
36.00	C_3	36.0005	
42.01	CNO	41.9985	
48.00	C_4	48.0005	
50.01	C_3N	50.0036	
60.00	C_5	60.0005	
62.97	PO_2	62.9641	
78.97	PO_3	78.9591	
79.97	SO_3	79.9574	
80.98	HSO_3	80.9652	
96.98	HSO_4	96.9601	
122.02	$C_2H_5NO_3P$	122.0013	
140.01	$C_2H_7NPO_4$	140.0118	
172.01	$C_3H_9PO_6$	172.0142	
227.21	$C_{14}H_{27}O_2$	227.2017	
241.04	$C_6H_{10}PO_8$	241.0119	
251.22	$C_{16}H_{27}O_2$	251.2017	FA(16:2)
253.20	$C_{16}H_{29}O_2$	253.2173	Palmitoleic acid
255.23	$C_{16}H_{31}O_2$	255.233	Palmitic acid
259.05	$C_6H_{12}PO_9$	259.0224	
279.22	$C_{18}H_{31}O_2$	279.233	Linoleic acid
281.25	$C_{18}H_{33}O_2$	281.2486	Oleic acid
283.27	$C_{18}H_{35}O_2$	283.2643	Stearic acid
299.08	$C_9H_{16}PO_9$	299.0537	Phosphatidylinositol
303.22	$C_{20}H_{31}O_2$	303.233	Arachidonic acid
305.21	$C_{20}H_{33}O_2$	305.2486	Dihomo-linoleic acid
699.58	$C_{39}H_{72}O_8P$	699.4965	PA(36:2)
701.63	$C_{39}H_{74}O_8P$	701.5121	PA(36:1)
885.64	$C_{47}H_{82}PO_{13}$	885.5499	PI 38:4
886.67	$C_{47}H_{83}PO_{13}$	886.5577	PI 38:3

List of Figures

List of Tables

List of Abbreviations

AFM	Atomic Force Microscopy
API	Active Pharmaceutical Ingredient
CD	Chemical Drying
CMAC	EPSRC Future Manufacturing Research Hub for Continuous Manufacturing and Crystallisation
CV	Cross-Validation
CVD	Cardiovascular Disease
DE	Delayed Extraction
DESI	Desorption Electrospray Ionization
DOE	Design of Experiment
EPSRC	Engineering and Physical Sciences Research Council
FD	Freeze Drying
GCIB	Gas Cluster Ion Beam
GUI	Graphical User Interface
HCAEC	Human Coronary Artery Endothelial Cell
HPLC	High Performance Liquid Chromatography
HSI	Hyperspectral Imaging
ICT	Information and Communications Technology
IR	Infrared
LMIG	Liquid Metal Ion Gun

LV	Latent Variable
MALDI	Matrix Assisted Laser Desorption Ionisation
MS	Mass Spectrometry
MS/MS	Tandem Mass Spectrometry
MSC	Multiplicative Scatter Correction
MSI	Mass Spectrometry Imaging
NIR	Near Infrared
NIST	National Institute of Standards and Technology
NPL	National Physical Laboratory
OM	Optical Microscopy
PAR	Protease-Activated Receptors
PC	Principle Component
PCA	Principle Component Analysis
PCH	Phosphatidylcholine Headgroup
PLSR	Partial Least Square Regression (often also found as PLS)
RMSE	Root Mean Square Error
RMSEC	Root Mean Square Error of Calibration
RMSECV	Root Mean Square Error of Cross-Validation
RMSEP	Root Mean Square Error of Prediction
RMSEV	Root Mean Square Error of Validation
SEM	Scanning Electron Microscope
SI	Secondary Ion
SIMS	Secondary Ion Mass Spectrometry
SNV	Standard Normal Variate
ToF	Time-of-Flight
UHV	Ultra High Vacuum
WHO	World Health Organisation
XRD	X-Ray Diffraction

Bibliography

1. Idea Pharma. *Top pharmaceutical companies for innovation: Roche, AbbVie, Novartis, and more — Fortune* 2020.
2. European Medicines Agency. *Annual Report - European Medicines Agency 2018* tech. rep. (European Medicines Agency, 2018).
3. Gautam, A. & Pan, X. The changing model of big pharma: Impact of key trends. *Drug Discovery Today* **21,** 379–384. doi:`10.1016/j.drudis.2015.10.002` (Mar. 2016).
4. Ford, J., Blair, A., Overman, J. & Naaz, B. *Pharmaceutical industry trends — Deloitte Insights* 2020.
5. Carroll, S. Goodbye blockbuster medicines; hello new pharmaceutical business models — Comment — Pharmaceutical Journal. *The Pharmaceutical Journal* **282** (June 2009).
6. Frazier, K. C. *Pharmaceutical Research And Development: The Process Behind New Medicines* tech. rep. (Pharmaceutical Research and Manufacturers of America (PhRMA), 2015).
7. Mennen, S. M. *et al.* The Evolution of High-Throughput Experimentation in Pharmaceutical Development and Perspectives on the Future. *Organic Process Research and Development* **23,** 1213–1242. doi:`10.1021/acs.oprd.9b00140` (June 2019).

8. Prideaux, B. & Stoeckli, M. Mass spectrometry imaging for drug distribution studies. *Journal of Proteomics* **75,** 4999–5013. doi:10.1016/j.jprot.2012.07.028 (Aug. 2012).

9. Shariatgorji, M., Svenningsson, P. & Andrén, P. E. Mass spectrometry imaging, an emerging technology in neuropsychopharmacology. *Neuropsychopharmacology* **39,** 34–39. doi:10.1038/npp.2013.215 (Jan. 2014).

10. Jungnickel, H., Laux, P. & Luch, A. Time-of-flight secondary ion mass spectrometry (ToF-SIMS): A new tool for the analysis of toxicological effects on single cell level. *Toxics* **4.** doi:10.3390/toxics4010005 (Feb. 2016).

11. Vanbellingen, Q. P. *et al.* Analysis of Chemotherapeutic Drug Delivery at the Single Cell Level Using 3D-MSI-TOF-SIMS. *Journal of the American Society for Mass Spectrometry* **27,** 2033–2040. doi:10.1007/s13361-016-1485-y (Dec. 2016).

12. Hall Barrientos, I. J. *et al.* Fabrication and characterisation of drug-loaded electrospun polymeric nanofibers for controlled release in hernia repair. *International Journal of Pharmaceutics* **517,** 329–337. doi:10.1016/j.ijpharm.2016.12.022 (Jan. 2017).

13. Muramoto, S., Gillen, G., Collett, C., Zeissler, C. J. & Garboczi, E. J. ToF-SIMS depth profiling of oral drug delivery films for 3D visualization of active pharmaceutical particles. *Surface and Interface Analysis* **52,** 76–83. doi:10.1002/sia.6707 (Mar. 2020).

14. Fletcher, J. S. Latest applications of 3D ToF-SIMS bio-imaging. *Biointerphases* **10,** 018902. doi:10.1116/1.4907727 (Mar. 2015).

15. Touboul, D., Kollmer, F., Niehuis, E., Brunelle, A. & Laprévote, O. Improvement of biological time-of-flight-secondary ion mass spectrometry imaging with a bismuth cluster ion source. *Journal of the American Society for Mass Spectrometry* **16,** 1608–1618. doi:10.1016/j.jasms.2005.06.005 (Oct. 2005).

16. Touboul, D. & Brunelle, A. What more can TOF-SIMS bring than other MS imaging methods? *Bioanalysis* **8,** 367–369. doi:10.4155/bio.16.11 (2016).

17. Rifai, N., Horvath, A. R. (R., Wittwer, C. (& Hoofnagle, A. N. *Principles and applications of clinical mass spectrometry : small molecules, peptides, and pathogens* 1st ed., 216 (Elsevier, June 2018).

18. Kool, J. & Niessen, W. M. A. (M. A. *Analyzing biomolecular interactions by mass spectrometry* (ed Kool, J.) 400 (Wiley, May 2015).

19. Rockwood, A. L., Kushnir, M. M. & Clarke, N. J. in *Principles and Applications of Clinical Mass Spectrometry* (eds Rifai, N., Horvath, A. R. & Wittwer, C. T.) 33–65 (Elsevier, 2018). doi:`https://doi.org/10.1016/B978-0-12-816063-3.00002-5`.

20. Mittal, R. D. Tandem Mass Spectroscopy in Diagnosis and Clinical Research. *Indian Journal of Clinical Biochemistry* **30,** 121–123. doi:`10.1007/s12291-015-0498-9` (Apr. 2015).

21. Furudate, T. *et al.* Possibility of monitoring granulation by analyzing the amount of hydroxypropylcellulose, a binder on the surface of granules, using ToF-SIMS. *International Journal of Pharmaceutics* **495,** 642–650. doi:`10.1016/j.ijpharm.2015.09.060` (Nov. 2015).

22. Tortora, L., de Notaristefani, F. & Ioele, M. ToF-SIMS investigation of gilt and painted leather: identification of indigo, oil binder and gold varnish. *Surface and Interface Analysis* **46,** 807–811. doi:`10.1002/sia.5450` (Oct. 2014).

23. Perkins, M. C. *et al.* The application of ToF-SIMS to the analysis of herbicide formulation penetration into and through leaf cuticles. *Colloids and Surfaces B: Biointerfaces* **67,** 1–13. doi:`10.1016/j.colsurfb.2008.04.019` (Nov. 2008).

24. Werner, H. W. Quantitative secondary ion mass spectrometry: A review. *Surface and Interface Analysis* **2,** 56–74. doi:`10.1002/sia.740020205` (Apr. 1980).

25. Terrence Murray, P. & Wayne Rabalais, J. Ejection Dynamics and Electronic Processes Governing Secondary Particle Emission in SIMS. *Journal of the American Chemical Society* **103,** 1007–1013. doi:`10.1021/ja00395a002` (1981).

26. Robinson, M. A. *Chemical Analysis of Cells and Tissues with Time-of-Flight Secondary Ion Mass Spectrometry* PhD thesis (University of Washington, 2013), 237.

27. Sigmund, P. Theory of Sputtering. I. Sputtering Yield of Amorphous and Polycrystalline Targets. *Physical Review* **184,** 383–416. doi:10.1103/PhysRev.184.383 (Aug. 1969).

28. Trindade, G. F. *The development of multivariate analysis methodologies for complex ToF-SIMS datasets : applications to materials science* PhD thesis (University of Surrey, 2018).

29. Spool, A. M. *The practice of TOF-SIMS : time of flight secondary ion mass spectrometry* 192 (Momentum Press, 2016).

30. Senoner, M. & Unger, W. E. SIMS imaging of the nanoworld: Applications in science and technology. *Journal of Analytical Atomic Spectrometry* **27,** 1050–1068. doi:10.1039/c2ja30015j (July 2012).

31. Lee, J. *Time-of-Flight Secondary Ion Mass Spectrometry – Fundamental Issues for Quantitative Measurements and Multivariate Data Analysis Time-of-Flight Secondary Ion Mass Spectrometry – Fundamental Issues for* Doctoral Thesis (University of Oxford, 2011), 1–13.

32. Gunnarsson, A., Kollmer, F., Sohn, S., Höök, F. & Sjövall, P. Spatial-Resolution Limits in Mass Spectrometry Imaging of Supported Lipid Bilayers and Individual Lipid Vesicles. *Analytical Chemistry* **82,** 2426–2433. doi:10.1021/ac902744u (Mar. 2010).

33. Mahoney, C. M. *Cluster Secondary Ion Mass Spectrometry: Principles and Applications* doi:10.1002/9781118589335 (John Wiley and Sons, Apr. 2013).

34. Postawa, Z. *et al.* Depth profiling by cluster projectiles as seen by computer simulations. *Surface and Interface Analysis* **43,** 12–15. doi:10.1002/sia.3417 (Jan. 2011).

35. Garrison, B. J. & Postawa, Z. Computational view of surface based organic mass spectrometry. *Mass Spectrometry Reviews* **27,** 289–315. doi:10.1002/mas.20165 (July 2008).

36. Anderton, C. R. & Gamble, L. J. Secondary Ion Mass Spectrometry Imaging of Tissues, Cells, and Microbial Systems. *Microscopy Today* **24,** 24–31. doi:10.1017/s1551929516000018 (Mar. 2016).

37. Benninghoven, A. Analysis of Submonolayers on Silver by Negative Secondary Ion Emission. *physica status solidi (b)* **34,** K169–K171. doi:10.1002/pssb.19690340267 (Jan. 1969).

38. Briggs, D. Recent advances in secondary lon mass spectrometry (SIMS) for polymer surface analysis. *British Polymer Journal* **21,** 3–15. doi:10.1002/pi.4980210103 (Jan. 1989).

39. Vickerman, J. C. & Briggs, D. *ToF-SIMS : Surface Analysis by Mass Spectrometry* 1st, 789 (IM Publications LLP, 2001).

40. Tian, H. *Visualisation and profiling of lipids in single biological cells using Time-of-Flight Secondary Ion Mass Spectrometry* PhD thesis (The University of Manchester, Manchester, Feb. 2012).

41. Van Der Heide, P. *Secondary Ion Mass Spectrometry: An Introduction to Principles and Practices* 1–365. doi:10.1002/9781118916780 (Wiley Blackwell, Sept. 2014).

42. Chait, B. & Standing, K. A time-of-flight mass spectrometer for measurement of secondary ion mass spectra. *International Journal of Mass Spectrometry and Ion Physics* **40,** 185–193. doi:10.1016/0020-7381(81)80041-1 (Oct. 1981).

43. Nuffel, S. V. *Three-dimensional Time-of-Flight Secondary Ion Mass Spectrometry Imaging of Primary Neuronal Cell Cultures* PhD thesis (University of Nottingham, 2017).

44. Fletcher, J. S., Lockyer, N. P. & Vickerman, J. C. C60, Buckminsterfullerene: its impact on biological ToF-SIMS analysis. *Surface and Interface Analysis* **38,** 1393–1400. doi:10.1002/sia.2461 (Nov. 2006).

45. Terlier, T., Lee, J. & Lee, Y. Investigation of human hair using ToF-SIMS: From structural analysis to the identification of cosmetic residues. *Journal of Vacuum Science & Technology B, Nanotechnology and Microelectronics: Materials, Processing, Measurement, and Phenomena* **36,** 03F131. doi:10.1116/1.5015928 (May 2018).

46. Sodhi, R. N. S. Time-of-flight secondary ion mass spectrometry (TOF-SIMS):—versatility in chemical and imaging surface analysis. *The Analyst* **129,** 483–487. doi:10.1039/B402607C (2004).

47. Shon, H. K., Yoon, S., Moon, J. H. & Lee, T. G. Improved mass resolution and mass accuracy in TOF-SIMS spectra and images using argon gas cluster ion beams. *Biointerphases* **11,** 02A321. doi:10.1116/1.4941447 (June 2016).

48. Tian, H. *et al.* Gas Cluster Ion Beam Time-of-Flight Secondary Ion Mass Spectrometry High-Resolution Imaging of Cardiolipin Speciation in the Brain: Identification of Molecular Losses after Traumatic Injury. *Analytical Chemistry* **89,** 4611–4619. doi:10.1021/acs.analchem.7b00164 (Apr. 2017).

49. Suzuki, M., Nojima, M., Fujii, M., Seki, T. & Matsuo, J. Mass analysis by Ar-GCIB-dynamic SIMS for organic materials. *Surface and Interface Analysis* **46,** 1212–1214. doi:10.1002/sia.5696 (Dec. 2014).

50. Ionoptika, I. B. T. *ToF-SIMS Analysis on Insulating Samples — Ionoptika Ltd* Feb. 2018.

51. Claus, T. K. *et al.* Simultaneous Dual Encoding of Three-Dimensional Structures by Light-Induced Modular Ligation. *Angewandte Chemie International Edition* **55,** 3817–3822. doi:10.1002/anie.201509937 (Mar. 2016).

52. Nuñez, J., Renslow, R., Cliff, J. B. & Anderton, C. R. NanoSIMS for biological applications: Current practices and analyses. *Biointerphases* **13,** 03B301. doi:10.1116/1.4993628 (June 2018).

53. Kilburn, M. R. & Wacey, D. in *Principles and Practice of Analytical Techniques in Geosciences* 1–34 (Royal Society of Chemistry, 2014). doi:10.1039/9781782625025-00001.

54. Passarelli, M. K. *et al.* The 3D OrbiSIMS - Label-free metabolic imaging with subcellular lateral resolution and high mass-resolving power. *Nature Methods* **14,** 1175–1183. doi:10.1038/nmeth.4504 (Dec. 2017).

55. Li, B. *et al.* Analytical capabilities of mass spectrometry imaging and its potential applications in food science. *Trends in Food Science & Technology* **47,** 50–63. doi:10.1016/J.TIFS.2015.10.018 (Jan. 2016).

56. Vickerman, J. C. Molecular imaging and depth profiling by mass spectrometry - SIMS, MALDI or DESI? *Analyst* **136,** 2199–2217. doi:10.1039/c1an00008j (June 2011).

57. Feenstra, A. D., Dueñas, M. E. & Lee, Y. J. Five Micron High Resolution MALDI Mass Spectrometry Imaging with Simple, Interchangeable, Multi-Resolution Optical System. *Journal of the American Society for Mass Spectrometry* **28,** 434–442. doi:10.1007/s13361-016-1577-8 (Mar. 2017).

58. Spraker, J. E., Luu, G. T. & Sanchez, L. M. Imaging mass spectrometry for natural products discovery: A review of ionization methods. *Natural Product Reports* **37,** 150–162. doi:10.1039/c9np00038k (Feb. 2020).

59. Shin, Y. S., Drolet, B., Mayer, R., Dolence, K. & Basile, F. Desorption electrospray ionization-mass spectrometry of proteins. *Analytical Chemistry* **79,** 3514–3518. doi:10.1021/ac062451t (May 2007).

60. Rinnan, Å., van den Berg, F., Balling Engelsen, S., van den Berg, F. & Engelsen, S. B. Review of the most common pre-processing techniques for near-infrared spectra. *TrAC - Trends in Analytical Chemistry* **28,** 1201–1222. doi:10.1016/j.trac.2009.07.007 (2009).

61. Engel, J. *et al.* Breaking with trends in pre-processing? *TrAC Trends in Analytical Chemistry* **50,** 96–106. doi:10.1016/J.TRAC.2013.04.015 (Oct. 2013).

62. Rinnan, Å. *Pre-processing in vibrational spectroscopy-when, why and how* Sept. 2014. doi:10.1039/c3ay42270d.

63. Gerretzen, J. *et al.* Simple and Effective Way for Data Preprocessing Selection Based on Design of Experiments. *Analytical Chemistry* **87,** 12096–12103. doi:10.1021/acs.analchem.5b02832 (Dec. 2015).

64. US Food and Drug Administration. *CBER-CDER Data Standards Strategy FY2018-FY2022* tech. rep. (U.S. Food and Drug Administration, Jan. 2018).

65. Flåten, G. R. & Walmsley, A. D. Using design of experiments to select optimum calibration model parameters. *Analyst* **128,** 935–943. doi:10.1039/b301555f (2003).

66. Swinehart, D. F. *The Beer-Lambert law* 1962. doi:10.1021/ed039p333.

67. Lever, J., Krzywinski, M. & Altman, N. Points of Significance: Principal component analysis. *Nature Methods* **14,** 641–642. doi:10.1038/nmeth.4346 (June 2017).

68. Ferreira, C. *Spatially and angularly resolved diffuse reflectance measurement for in-line analysis of particle suspensions : a multi-sensor approach — University of Strathclyde* PhD thesis (University of Strathclyde, Glasgow, 2019).

69. Robinson, M. A., Graham, D. J., Morrish, F., Hockenbery, D. & Gamble, L. J. Lipid analysis of eight human breast cancer cell lines with ToF-SIMS. *Biointerphases* **11,** 02A303. doi:10.1116/1.4929633 (June 2016).

70. Baker, M. J. *et al.* Discrimination of prostate cancer cells and non-malignant cells using secondary ion mass spectrometry. *The Analyst* **133,** 175–179. doi:10.1039/B712853C (Jan. 2008).

71. José, G. E. *Spatially and angularly resolved diffuse reflectance spectroscopy for in-situ monitoring of suspension polymerisation reactions* PhD thesis (2017).

72. Biancolillo, A. & Marini, F. *Chemometric methods for spectroscopy-based pharmaceutical analysis* Nov. 2018. doi:10.3389/fchem.2018.00576.

73. Wold, S., Sjöström, M. & Eriksson, L. *PLS-regression: A basic tool of chemometrics* in *Chemometrics and Intelligent Laboratory Systems* **58** (Elsevier, Oct. 2001), 109–130. doi:10.1016/S0169-7439(01)00155-1.

74. Nordon, A. *Multivariate Regression Lecture Notes* Glasgow, 2015.

75. Næs, T., Isaksson, T. & Fearn, T. *A user-friendly guide to multivariate calibration and classification - Ghent University Library* 2007th ed. (Chichester : NIR, 2004., 2002).

76. MacFhionnghaile, P. *Quantitative Analysis of the Amorphous Phase and Multiple Polymorphs of Model Sulfa-Drugs; Sulfamerazine and Sulfathiazole* PhD thesis (National University of Ireland Galway, Oct. 2013).

77. Garson, D. G. *Partial Least Squares: Regression & Structural Equation Models* 2016th ed. (ed G. David Garson) (Statistical Associates Publishing, Asheboro, 2014).

78. Li, H.-D., Xu, Q.-S. & Liang, Y.-Z. libPLS: An Integrated Library for Partial Least Squares Regression and Discriminant Analysis. *Chemometrics and Intelligent Laboratory Systems* **176,** 34–43. doi:10.7287/peerj.preprints.190v1 (2018).

79. Kroese, D. P., Brereton, T., Taimre, T. & Botev, Z. I. Why the Monte Carlo method is so important today. *Wiley Interdisciplinary Reviews: Computational Statistics* **6,** 386–392. doi:10.1002/wics.1314 (Nov. 2014).

80. Zhang, L. *et al.* An enhanced Monte Carlo outlier detection method. *Journal of Computational Chemistry* **36,** 1902–1906. doi:10.1002/jcc.24026 (Sept. 2015).

81. Czarnecki, M. A. Resolution enhancement in second-derivative spectra. *Applied spectroscopy* **69,** 67–74. doi:10.1366/14-07568 (Jan. 2015).

82. Owen, A. J. *Uses of Derivative Spectroscopy Application* tech. rep. (Agilent Technologies, 1995).

83. Savitzky, A. & E, M. J. Smoothing and Differentiation of Data by Simplified Least Squares Procedures. *Z. Physiol. Chem* **40,** 1832 (1951).

84. Martens, H. & Næs, T. *Multivariate calibration* 419 (Wiley, 1989).

85. Dhanoa, M., Lister, S., Sanderson, R. & Barnes, R. The Link between Multiplicative Scatter Correction (MSC) and Standard Normal Variate (SNV) Transformations of NIR Spectra. *Journal of Near Infrared Spectroscopy* **2,** 43–47. doi:10.1255/jnirs.30 (Jan. 1994).

86. Van den Berg, R. A., Hoefsloot, H. C. J., Westerhuis, J. A., Smilde, A. K. & van der Werf, M. J. Centering, scaling, and transformations: improving the biological information content of metabolomics data. *BMC genomics* **7,** 142. doi:10.1186/1471-2164-7-142 (June 2006).

87. EigenvectorInc. *Advanced Preprocessing: Sample Normalization - Eigenvector Research Documentation Wiki* Sept. 2019.

88. Fisher, R. A. The Design of Experiments. *Nature* **137,** 252–254. doi:10.1038/137252a0 (Feb. 1936).

89. Anderson, M. J. & Whitcomb, P. J. in *Kirk-Othmer Encyclopedia of Chemical Technology* (John Wiley & Sons, Inc., Hoboken, NJ, USA, Sept. 2010). doi:10.1002/0471238961.0405190908010814.a01.pub3.

90. Lane, D. M. *Online Statistics Education: An Interactive Multimedia Course of Study* 2015.

91. Dyrby, M., Engelsen, S. B., Nørgaard, L., Bruhn, M. & Lundsberg-Nielsen, L. Chemometric Quantitation of the Active Substance (Containing a cyano group) in a Pharmaceutical Tablet Using Near-Infrared (NIR) Transmittance and NIR FT-Raman Spectra. *Applied Spectroscopy* **56,** 579–585. doi:10.1366/0003702021955358 (May 2002).

92. Brown, C. *Digital twins, microfactories and people for future pharmaceutical manufacturing* Nov. 2019.

93. Chen, Y. *et al.* Digital Twins in Pharmaceutical and Biopharmaceutical Manufacturing: A Literature Review. *Processes* **8,** 1088. doi:10.3390/pr8091088 (Sept. 2020).

94. Wold, S., Antti, H., Lindgren, F. & Ohman, J. Orthogonal signal correction of near-infrared spectra. *Chemometrics and Intelligent Laboratory Systems* **44,** 175–185 (1998).

95. Svensson, O., Kourti, T. & MacGregor, J. F. An investigation of orthogonal signal correction algorithms and their characteristics. *Journal of Chemometrics* **16,** 176–188. doi:10.1002/cem.700 (Apr. 2002).

96. Li, C. *et al.* Recent progress in drug delivery. *Acta Pharmaceutica Sinica B* **9,** 1145–1162. doi:10.1016/j.apsb.2019.08.003 (Nov. 2019).

97. Witschi, C. & Doelker, E. Residual solvents in pharmaceutical products: Acceptable limits, influences on physicochemical properties, analytical methods and documented values. *European Journal of Pharmaceutics and Biopharmaceutics* **43,** 215–242. doi:10.1016/S0939-6411(96)00037-9 (June 1997).

98. Ellis, F. *Paracetamol - a curriculum resource* 5–7. doi:10.1007/s10787-013-0172-x (2002).

99. Kuvadia, Z. B. & Doherty, M. F. Effect of structurally similar additives on crystal habit of organic molecular crystals at low supersaturation. *Crystal Growth and Design* **13,** 1412–1428. doi:10.1021/cg3010618 (Apr. 2013).

100. Hendriksen, B. A. & Grant, D. J. The effect of structurally related substances on the nucleation kinetics of paracetamol (acetaminophen). *Journal of Crystal Growth* **156,** 252–260. doi:10.1016/0022-0248(95)00301-0 (Nov. 1995).

101. Hendriksen, B. A., Grant, D. J., Meenan, P. & Green, D. A. Crystallisation of paracetamol (acetaminophen) in the presence of structurally related substances. *Journal of Crystal Growth* **183,** 629–640. doi:10.1016/S0022-0248(97)00488-0 (Feb. 1998).

102. Thompson, C., Davies, M. C., Roberts, C. J., Tendler, S. J. & Wilkinson, M. J. The effects of additives on the growth and morphology of paracetamol (acetaminophen) crystals. *International Journal of Pharmaceutics* **280,** 137–150. doi:10.1016/j.ijpharm.2004.05.010 (Aug. 2004).

103. Chow, A. H., Chow, P. K., Zhongshan, W. & Grant, D. J. Modification of acetaminophen crystals: influence of growth in aqueous solutions containing p-acetoxyacetanilide on crystal properties. *International Journal of Pharmaceutics* **24,** 239–258. doi:10.1016/0378-5173(85)90024-9 (May 1985).

104. Prasad, K. V., Ristic, R. I., Sheen, D. B. & Sherwood, J. N. Crystallization of paracetamol from solution in the presence and absence of impurity. *International Journal of Pharmaceutics* **215,** 29–44. doi:10.1016/S0378-5173(00)00653-0 (Mar. 2001).

105. Saleemi, A., Onyemelukwe, I. I. & Nagy, Z. Effects of a structurally related substance on the crystallization of paracetamol. *Frontiers of Chemical Science and Engineering* **7,** 79–87. doi:10.1007/s11705-013-1308-7 (Mar. 2013).

106. Ottoboni, S. *et al.* Impact of Paracetamol Impurities on Face Properties: Investigating the Surface of Single Crystals Using TOF-SIMS. *Crystal Growth and Design* **18,** 2750–2758. doi:10.1021/acs.cgd.7b01411 (2018).

107. Fletcher, J. S. & Vickerman, J. C. A new SIMS paradigm for 2D and 3D molecular imaging of bio-systems. *Analytical and Bioanalytical Chemistry* **396,** 85–104. doi:10.1007/s00216-009-2986-3 (Jan. 2010).

108. Barnes, T. J., Kempson, I. M. & Prestidge, C. A. Surface analysis for compositional, chemical and structural imaging in pharmaceutics with mass spectrometry: A ToF-SIMS perspective. *International Journal of Pharmaceutics* **417,** 61–69. doi:10.1016/j.ijpharm.2011.01.043 (Sept. 2011).

109. Bennlnghoven, A. & Sichtermann, W. K. Detection, Identification and Structural Investigation of Biologically Important Compounds by Secondary Ion Mass Spectrometry. *Analytical Chemistry* **50,** 1180–1184. doi:10.1021/ac50030a043 (1978).

110. Bugay, D. E. Characterization of the solid-state: Spectroscopic techniques. *Advanced Drug Delivery Reviews* **48,** 43–65. doi:10.1016/S0169-409X(01)00101-6 (May 2001).

111. Mahoney, C. M., Fahey, A. J. & Belu, A. M. Three-Dimensional Compositional Analysis of Drug Eluting Stent Coatings Using Cluster Secondary Ion Mass Spectrometry. *Analytical Chemistry* **80,** 624–632. doi:`10.1021/ac701644j` (Feb. 2008).

112. Rafati, A. *et al.* Chemical and spatial analysis of protein loaded PLGA microspheres for drug delivery applications. *Journal of Controlled Release* **162,** 321–329. doi:`10.1016/j.jconrel.2012.05.008` (2012).

113. Chan, J. G. Y. *et al.* A novel dry powder inhalable formulation incorporating three first-line anti-tubercular antibiotics. *European Journal of Pharmaceutics and Biopharmaceutics* **83,** 285–292. doi:`10.1016/j.ejpb.2012.08.007` (Feb. 2013).

114. Muster, T. H. & Prestidge, C. A. Face specific surface properties of pharmaceutical crystals. *Journal of Pharmaceutical Sciences* **91,** 1432–1444. doi:`10.1002/jps.10125` (June 2002).

115. Ottoboni, S. *Developing strategies and equipment for continuous isolation of active pharmaceutical ingredients (APIs) by filtration, washing and drying* PhD thesis (University of Strathclyde, Glasgow, 2018).

116. Guilhaus, M. in *Encyclopedia of Analytical Science: Second Edition* 412–423 (Elsevier Inc., Jan. 2004). doi:`10.1016/B0-12-369397-7/00347-2`.

117. World Health Organisation. *Cardiovascular diseases (CVDs)* May 2017.

118. Nichols, M. *et al. European Cardiovascular Disease Statistics 2012* tech. rep. (European Heart Network AISBL, Brussels, Nov. 2012).

119. Stewart, J., Manmathan, G. & Wilkinson, P. Primary prevention of cardiovascular disease: A review of contemporary guidance and literature. *JRSM Cardiovascular Disease* **6,** 204800401668721. doi:`10.1177/2048004016687211` (Jan. 2017).

120. Kalanuria, A. A., Nyquist, P. & Ling, G. *The prevention and regression of atherosclerotic plaques: Emerging treatments* 2012. doi:`10.2147/VHRM.S27764`.

121. Bergheanu, S. C., Bodde, M. C. & Jukema, J. W. Pathophysiology and treatment of atherosclerosis: Current view and future perspective on lipoprotein modification treatment. *Netherlands Heart Journal* **25,** 231–242. doi:10.1007/s12471-017-0959-2 (Apr. 2017).

122. Palta, S., Saroa, R. & Palta, A. Overview of the coagulation system. *Indian Journal of Anaesthesia* **58,** 515–523. doi:10.4103/0019-5049.144643 (Sept. 2014).

123. VILESGONZALEZ, J. Atherothrombosis: A widespread disease with unpredictable and life-threatening consequences*1. *European Heart Journal* **25,** 1197–1207. doi:10.1016/j.ehj.2004.03.011 (July 2004).

124. Heuberger, D. M. & Schuepbach, R. A. Protease-activated receptors (PARs): Mechanisms of action and potential therapeutic modulators in PAR-driven inflammatory diseases. *Thrombosis Journal* **17,** 4. doi:10.1186/s12959-019-0194-8 (Mar. 2019).

125. Leonardi, S. & Becker, R. C. in *Handbook of experimental pharmacology* 210, 239–260 (2012). doi:10.1007/978-3-642-29423-5_10.

126. Gurbel, P. A. *et al.* Cell-Penetrating Pepducin Therapy Targeting PAR1 in Supjects with Coronary Artery Disease. *Arteriosclerosis, Thrombosis, and Vascular Biology* **36,** 189–197. doi:10.1161/ATVBAHA.115.306777 (Jan. 2016).

127. Zhang, P. *et al.* Suppression of arterial thrombosis without affecting hemostatic parameters with a cell-penetrating PAR1 pepducin. *Circulation* **126,** 83–91. doi:10.1161/CIRCULATIONAHA.112.091918 (July 2012).

128. Brouck, L. *Investigating the off-target effects of the clinical trial candidate PZ-128* PhD thesis (University of Strathclyde, 2019).

129. Chandra, S., Bernius, M. T. & Morrison, G. H. Intracellular Localization of Diffusible Elements in Frozen-Hydrated Biological Specimens with Ion Microscopy. *Analytical Chemistry* **58,** 493–496. doi:10.1021/ac00293a053 (1986).

130. Chandra, S., Ausserer, W. A. & Morrison, G. H. Evaluation of matrix effects in ion microscopic analysis of freeze-fractured, freeze-dried cultured cells. *Journal of Microscopy* **148,** 223–239. doi:10.1111/j.1365-2818.1987.tb02869.x (Dec. 1987).

131. Ausserer, W. A., Chandra, S. & Morrison, G. H. Morphological and elemental integrity of freeze-fractured, freeze-dried cultured cells during ion microscopic analysis. *Journal of Microscopy* **154,** 39–57. doi:10.1111/j.1365-2818.1989.tb00566.x (Apr. 1989).

132. Colliver, T. L. *et al.* Atomic and Molecular Imaging at the Single-Cell Level with TOF-SIMS. *Analytical Chemistry* **69,** 2225–2231. doi:10.1021/ac9701748 (July 1997).

133. Pacholski, M. L., Cannon, D. M., Ewing, A. G. & Winograd, N. Imaging of exposed headgroups and tailgroups of phospholipid membranes by mass spectrometry. *Journal of the American Chemical Society* **121,** 4716–4717. doi:10.1021/ja983022i (May 1999).

134. Newman, C. F. *et al.* Intracellular Drug Uptake - A Comparison of Single Cell Measurements Using ToF-SIMS Imaging and Quantification from Cell Populations with LC/MS/MS. *Analytical Chemistry* **89,** 11944–11953. doi:10.1021/acs.analchem.7b01436 (Nov. 2017).

135. Hua, X., Li, H. W. & Long, Y. T. Investigation of Silver Nanoparticle Induced Lipids Changes on a Single Cell Surface by Time-of-Flight Secondary Ion Mass Spectrometry. *Analytical Chemistry* **90,** 1072–1076. doi:10.1021/acs.analchem.7b04591 (Jan. 2018).

136. Jiang, H. *et al.* High-resolution sub-cellular imaging by correlative NanoSIMS and electron microscopy of amiodarone internalisation by lung macrophages as evidence for drug-induced phospholipidosis. *Chemical Communications* **53,** 1506–1509. doi:10.1039/c6cc08549k (2017).

137. Berman, E. *et al.* Preparation of single cells for imaging/profiling mass spectrometry. *J. Am. Soc. Mass Spectrom.* **19.** doi:10.1021/jasms.8b03248 (2008).

138. Malm, J., Giannaras, D., Riehle, M. O., Gadegaard, N. & Sjövall, P. Fixation and drying protocols for the preparation of cell samples for time-of-flight secondary ion mass spectrometry analysis. *Analytical Chemistry* **81,** 7197–7205. doi:10.1021/ac900636v (Sept. 2009).

139. Lanekoff, I. *et al. Time of flight mass spectrometry imaging of samples fractured in situ with a spring-loaded trap system* in *Analytical Chemistry* **82** (American Chemical Society, Aug. 2010), 6652–6659. doi:10.1021/ac101243b.

140. Winograd, N. & Bloom, A. Sample preparation for 3D SIMS chemical imaging of cells. *Methods in Molecular Biology* **1203,** 9–19. doi:10.1007/978-1-4939-1357-2_2 (2015).

141. Severs, N. J. & Shotton, D. M. *Rapid freezing, freeze fracture, and deep etching* eng (New York (N.Y.) : Wiley-Liss, 1995).

142. Hong, Z., Staiculescu, M. C., Hampel, P., Levitan, I. & Forgacs, G. How cholesterol regulates endothelial biomechanics. *Frontiers in Physiology* **3,** 426. doi:10.3389/fphys.2012.00426 (Nov. 2012).

143. Halliwell, W. H. *Cationic amphiphilic drug-induced phospholipidosis* in *Toxicologic Pathology* **25** (SAGE Publications Inc., 1997), 53–60. doi:10.1177/019262339702500111.

144. Anderson, N. & Borlak, J. Drug-induced phospholipidosis. *FEBS Letters* **580,** 5533–5540. doi:10.1016/j.febslet.2006.08.061 (Oct. 2006).

145. El-Sherif, N. & Turitto, G. Electrolyte disorders and arrhythmogenesis. *Cardiology Journal* **18,** 233–245 (2011).

146. Clardy, J., Fischbach, M. A. & Currie, C. R. The natural history of antibiotics. *Current Biology* **19,** R437. doi:10.1016/j.cub.2009.04.001 (June 2009).

147. Ventola, C. L. The antibiotic resistance crisis: causes and threats. *P & T journal* **40,** 277–283. doi:Article (2015).

148. Van Boeckel, T. P. *et al.* Global antibiotic consumption 2000 to 2010: An analysis of national pharmaceutical sales data. *The Lancet Infectious Diseases* **14,** 742–750. doi:10.1016/S1473-3099(14)70780-7 (2014).

149. Da Cunha, B. R., Fonseca, L. P. & Calado, C. R. Antibiotic discovery: Where have we come from, where do we go? *Antibiotics* **8.** doi:10.3390/antibiotics8020045 (June 2019).

150. Spellberg, B. *The future of antibiotics* June 2014. doi:10.1186/cc13948.

151. Gould, I. M. & Bal, A. M. New antibiotic agents in the pipeline and how hey can help overcome microbial resistance. *Virulence* **4,** 185–191. doi:10.4161/viru.22507 (Feb. 2013).

152. Michael, C. A., Dominey-Howes, D. & Labbate, M. *The antimicrobial resistance crisis: Causes, consequences, and management* Sept. 2014. doi:10.3389/fpubh.2014.00145.

153. Smith, D. *Antibiotics Are Money-Losers For Big Pharma. How Can We Incentivize The Development Of New Ones?* Jan. 2018.

154. Mazlan, N., Baba, S., Tate, R., Clements, C. & Edrada-Ebel, R. Metabolomics studies of secondary metabolites from co culture of Fusarium sp. and Streptomyces sp. in the search for new potential antibiotics. *Planta Medica* **80,** P1M14. doi:10.1055/s-0034-1394581 (Oct. 2014).

155. Krug, D. & Müller, R. *Secondary metabolomics: The impact of mass spectrometry-based approaches on the discovery and characterization of microbial natural products* 2014. doi:10.1039/c3np70127a.

156. Dettmer, K., Aronov, P. A. & Hammock, B. D. Mass spectrometry-based metabolomics. *Mass Spectrometry Reviews* **26,** 51–78. doi:10.1002/mas.20108 (Jan. 2007).

157. Clish, C. B. Metabolomics: an emerging but powerful tool for precision medicine. *Molecular Case Studies* **1,** a000588. doi:10.1101/mcs.a000588 (Oct. 2015).

158. Roessner, U. & Bowne, J. What is metabolomics all about? *BioTechniques* **46,** 363–365. doi:10.2144/000113133 (Apr. 2009).

159. Jewett, M. C., Hofmann, G. & Nielsen, J. Fungal metabolite analysis in genomics and phenomics. *Current Opinion in Biotechnology* **17,** 191–197. doi:10.1016/j.copbio.2006.02.001 (Apr. 2006).

160. Bérdy, J. *Thoughts and facts about antibiotics: Where we are now and where we are heading* Apr. 2012. doi:10.1038/ja.2012.27.

161. Wright, G. D. Opportunities for natural products in 21st century antibiotic discovery. *Natural Product Reports* **34,** 694–701. doi:10.1039/c7np00019g (July 2017).

162. Burgess, J. G., Jordan, E. M., Bregu, M., Mearns-Spragg, A. & Boyd, K. G. Microbial antagonism: A neglected avenue of natural products research. *Journal of Biotechnology* **70,** 27–32. doi:10.1016/S0168-1656(99)00054-1 (Apr. 1999).

163. Feichtmayer, J., Deng, L. & Griebler, C. Antagonistic microbial interactions: Contributions and potential applications for controlling pathogens in the aquatic systems. *Frontiers in Microbiology* **8,** 2192. doi:10.3389/fmicb.2017.02192 (Nov. 2017).

164. Ramírez, S. R. *et al.* Metabolites produced by antagonistic microbes inhibit the principal avocado pathogens in vitro. **33,** 58–63. doi:10.15446/agron.colomb.v33n1.48241 (2015).

165. Vinale, F. *et al.* Co-Culture of Plant Beneficial Microbes as Source of Bioactive Metabolites. *Scientific Reports* **7.** doi:10.1038/s41598-017-14569-5 (Dec. 2017).

166. Ueda, K. & Beppu, T. *Antibiotics in microbial coculture* Apr. 2017. doi:10.1038/ja.2016.127.

167. Abdalla, M. A., Sulieman, S. & McGaw, L. J. *Microbial communication: A significant approach for new leads* Nov. 2017. doi:10.1016/j.sajb.2017.10.001.

168. Bertrand, S. *et al.* Metabolite induction via microorganism co-culture: A potential way to enhance chemical diversity for drug discovery. *Biotechnology Advances* **32,** 1180–1204. doi:10.1016/j.biotechadv.2014.03.001 (2014).

169. Erhard, M., von Döhren, H. & Jungblut, P. Rapid typing and elucidation of new secondary metabolites of intact cyanobacteria using MALDI-TOF mass spectrometry. *Nature Biotechnology* **15,** 906–909. doi:10.1038/nbt0997-906 (1997).

170. Lanni, E. J. *et al.* MALDI-guided SIMS: Multiscale imaging of metabolites in bacterial biofilms. *Analytical Chemistry* **86,** 9139–9145. doi:10.1021/ac5020222 (Sept. 2014).

171. Dunham, S. J., Ellis, J. F., Li, B. & Sweedler, J. V. Mass spectrometry imaging of complex microbial communities. *Accounts of Chemical Research* **50,** 96–104. doi:10.1021/acs.accounts.6b00503 (2017).

172. Hoffmann, T. & Dorrestein, P. C. Homogeneous Matrix Deposition on Dried Agar for MALDI Imaging Mass Spectrometry of Microbial Cultures. *Journal of the American Society for Mass Spectrometry* **26,** 1959–1962. doi:10.1007/s13361-015-1241-8 (Nov. 2015).

173. Ren, J. L., Zhang, A. H., Kong, L. & Wang, X. J. Advances in mass spectrometry-based metabolomics for investigation of metabolites. *RSC Advances* **8,** 22335–22350. doi:10.1039/c8ra01574k (June 2018).

174. Takáts, Z., Wiseman, J. M. & Cooks, R. G. Ambient mass spectrometry using desorption electrospray ionization (DESI): instrumentation, mechanisms and applications in forensics, chemistry, and biology. *Journal of Mass Spectrometry* **40,** 1261–1275. doi:10.1002/jms.922 (Oct. 2005).

175. Ghyselinck, J., Van Hoorde, K., Hoste, B., Heylen, K. & De Vos, P. Evaluation of MALDI-TOF MS as a tool for high-throughput dereplication. *Journal of Microbiological Methods* **86,** 327–336. doi:10.1016/j.mimet.2011.06.004 (Sept. 2011).

176. Gonzalez, D. J. *et al.* Observing the invisible through imaging mass spectrometry, a window into the metabolic exchange patterns of microbes. *Journal of Proteomics* **75,** 5069–5076. doi:10.1016/j.jprot.2012.05.036 (Aug. 2012).

177. Watrous, J., Hendricks, N., Meehan, M. & Dorrestein, P. C. Capturing bacterial metabolic exchange using thin film desorption electrospray ionization-imaging mass spectrometry. *Analytical Chemistry* **82,** 1598–1600. doi:`10.1021/ac9027388` (Mar. 2010).

178. Dunham, S. J. *et al.* Quantitative SIMS Imaging of Agar-Based Microbial Communities. *Analytical Chemistry* **90,** 5654–5663. doi:`10.1021/acs.analchem.7b05180` (May 2018).

179. Parte, A. C., Sardà Carbasse, J., Meier-Kolthoff, J. P., Reimer, L. C. & Göker, M. List of Prokaryotic names with Standing in Nomenclature (LPSN) moves to the DSMZ. *International Journal of Systematic and Evolutionary Microbiology.* doi:`10.1099/ijsem.0.004332` (July 2020).

180. de Lima Procópio, R. E., da Silva, I. R., Martins, M. K., de Azevedo, J. L. & de Araújo, J. M. *Antibiotics produced by Streptomyces* Sept. 2012. doi:`10.1016/j.bjid.2012.08.014`.

181. Hasani, A., Kariminik, A. & Issazadeh, K. *Streptomycetes: Characteristics and Their Antimicrobial Activities* tech. rep. 1 (2014), 63–75.

182. Sivalingam, P., Hong, K., Pote, J. & Prabakar, K. Extreme environment streptomyces: Potential sources for new antibacterial and anticancer drug leads? *International Journal of Microbiology* **2019.** doi:`10.1155/2019/5283948` (2019).

183. Vanbellingen, Q. P. *et al.* Time-of-flight secondary ion mass spectrometry imaging of biological samples with delayed extraction for high mass and high spatial resolutions. *Rapid Communications in Mass Spectrometry* **29,** 1187–1195. doi:`10.1002/rcm.7210` (July 2015).

184. Debois, D. *et al.* In situ localisation and quantification of surfactins in a Bacillus subtilis swarming community by imaging mass spectrometry. *PROTEOMICS* **8,** 3682–3691. doi:`10.1002/pmic.200701025` (Sept. 2008).

185. Baig, N. F. *et al.* Multimodal chemical imaging of molecular messengers in emerging Pseudomonas aeruginosa bacterial communities. *Analyst* **140,** 6544–6552. doi:`10.1039/c5an01149c` (Oct. 2015).