Master's Thesis

Astrophysical Sciences

Asteroid Spectra and Machine Learning

Hilppa Hietala

March 2020

Supervisors: Antti Penttilä

Karri Muinonen

Censors: Karri Muinonen

Mikael Granvik

UNIVERSITY OF HELSINKI

MASTER'S PROGRAMME IN PARTICLE PHYSICS AND ASTROPHYSICAL SCIENCES

Gustaf Hällströmin katu 2

00560 Helsinki

HELSINGIN YLIOPISTO — HELSINGFORS UNIVERSITET — UNIVERSITY OF HELSINKI

Tiedekunta — Fakultet — Faculty	Laitos — Institution — Department
Faculty of Science	Department of Physics

Tekijä — Författare — Author
Hilppa Hietala

Työn nimi — Arbetets titel — Title

Asteroid Spectra and Machine Learning

Oppiaine — Läroämne — Subject
Astrophysical Sciences

Työn laji — Arbetets art — Level	Aika — Datum — Month and year	Sivumäärä — Sidoantal — Number of pages
Master's Thesis	March 2020	66 + Appendices

Tiivistelmä — Referat — Abstract

The aim of this thesis is to explore applications of machine learning to the study of asteroid spectra, and as such, its research question can be summarized as: How can asteroid spectra be analyzed using machine learning? The question is explored through evaluation of the obtained solutions to two tasks: the optimal locations of spectrophotometric filters for asteroid classification success and the formation of an asteroid taxonomy through unsupervised clustering.

First, background theory for asteroids and particularly spectroscopy of asteroids is presented. Next, the theory of machine learning is briefly discussed, including a focus on the method utilized to solve the first task: neural networks. The first task is executed by developing an optimization algorithm that has access to a neural network that can determine the classification success rate of data samples that would be obtained using spectrophotometric filters at specific locations within the possible wavelength range. The second task, on the other hand, is evaluated through determining the optimal number of clusters for the given dataset and then developing taxonomies with the clustering algorithm k-means.

The obtained results for the first task involving the optimal locations of filters for spectrophotometry seem reliable, and correlate relatively well with well-known mineralogical features on asteroid surfaces. The taxonomic systems developed by the unsupervised clustering also succeeded rather well, as many of the formed clusters seem to be meaningful and follow the trends in other asteroid taxonomies. Therefore, it seems that based on the two investigated tasks, machine learning can be applied well to asteroid spectroscopy. For future studies, larger datasets would be required for improving the overall reliability of the results.

Avainsanat — Nyckelord — Keywords
asteroid, spectroscopy, machine learning, neural network

Säilytyspaikka — Förvaringsställe — Where deposited
Helsinki University Library

Muita tietoja — övriga uppgifter — Additional information

PREFACE

Writing a thesis is rarely a solitary project. While I certainly spent plenty of hours poring over my work alone, there are several people this thesis could not exist without. First of all, I would like to thank my supervisors Antti Penttilä and Karri Muinonen from the University of Helsinki. You made it possible for me to start working on this thesis with a schedule that was unconventional but suited me best, and your guidance ensured that the quality was the best I could produce.

I would also like to thank Francesca DeMeo from Massachusetts Institute of Technology, who kindly provided me with the base asteroid spectra utilized in this thesis and answered my questions regarding them. Similarly, I am grateful to Tampere University, where I finished my bachelor's degree. During the process of writing my bachelor's thesis, I grew fond of the provided LaTeX template, and as such continued to use it as a base for the formatting seen within this work.

Finally, I am grateful for the help of my family. My parents taught me the value of being hard-working while still remaining curious about the world that surrounds me. Needless to say, both of those characteristics were useful during the process of working on this thesis. I am especially grateful to my mother, who continues to cheer me on, no matter what I set my mind to. I would also like to thank my spouse, Kris. Among several feats, you helped me find balance between my studies and my life, and for that, I am forever grateful.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND SYMBOLS

Acronyms

BDM09	Bus-DeMeo dataset
CCD	Charge-coupled-device
DNN	Direct test neural network
ECAS	Eight-Color Asteroid Survey
FC	Framing Camera
FWHM	Full-width-half-maximum
MITHNEOS	MIT-Hawaii Near-Earth Object Spectroscopic Survey
NEAR	Near Earth Asteroid Rendezvous – Shoemaker
ONN	Optimization neural network
PCA	Principal component analysis
PDF	Probability density function
SD	Standard deviation
SDSS	Sloan Digital Sky Survey
SMASS II	Small Main-Belt Asteroid Spectroscopic Survey, Phase II

Glossary

$\mathbf{a}$	Layer output column vector
a	Point in Euclidean space
$a(i)$	Average distance from asteroid spectrum i to those in other clusters
$\mathbf{b}$	Bias vector
b	Point in Euclidean space
$b(i)$	Minimum average distance from asteroid spectrum i to those in other clusters
c	Convoluted data
$\mathbf{d}$	Probability density function vector

$d_{\cos}$	Cosine distance
d_j	Individual value in the probability density function vector
d_{sqeuc}	Squared Euclidean distance
$\mathbf{f}$	Layer activation function
i	Individual asteroid spectrum
k	Number of cluster centers
λ	Wavelength
λ_n	Central wavelength for the nth filter
m	Number of dimensions in Euclidean space
μ	Mean
n	Filter number
$\mathbf{p}$	Input vector
$\mathbf{r}$	Real data vector
s	Number of object features
$s(i)$	Silhouette value for the i spectrum
σ	Standard deviation
$\mathbf{W}$	Weight matrix
$\mathbf{x}$	Example vector for cosine distance
$\mathbf{y}$	Example vector for cosine distance

1 INTRODUCTION

The contents of the Universe have fascinated humankind for millennia. Our Solar System was a natural place to start exploring it, not only because it gives context for our place in the Universe, but also because it is arguably the easiest to observe. There are many objects within this system to consider, but this thesis centers around some of the smallest — the asteroids. Once one has chosen an object to scrutinize, a method for doing so must also be determined. There are several physical aspects of a celestial body to analyze, such as its orbit, size, or mass. However, one characteristic that can reveal vast amounts of information about an asteroid is how it scatters and absorbs light. The phenomena can be measured through spectroscopy, the study of how matter interacts with electromagnetic radiation across a given band of wavelengths. Consequently, asteroid spectra form the main focus of this work.

While asteroids have intrigued astronomers since their discovery, some industries have also developed an interest in them recently. Two fields to note are planetary defense and asteroid mining. As our understanding of the space that surrounds us increases, so does the need to protect our planet against it. This forms the goal of planetary defense, and in order to coordinate plans to defend against a potential asteroid impact, knowledge of the object is crucial; for example, knowing its compositional details would help tremendously in determining how to mitigate the threat it poses. Asteroid mining, on the other hand, is a speculated possibility to gather resources to replace the depleted reserves of our planet. Knowing what minerals an asteroid is composed of is naturally important for efficient collection of these resources. Spectroscopy offers answers to both industries, since it can be used to infer compositional details about any target asteroid, and the observed features in its spectrum can be correlated with specific minerals on its surface.

The ability to group objects together can make using their data more efficient, as one no longer needs to work on a case-by-case basis. It also facilitates easier comparison of different types of objects. Taxonomies are particularly relevant in astronomy, where the amount of objects in a given dataset can be extremely high. However, analyzing vast amounts of data, establishing groupings and following them purely by hand is taxing and time-consuming, so it is natural to look for ways to make the process easier. An excellent candidate for solving the problem is machine learning. It originates from computer science, but has also been a rapidly growing trend in countless other fields due to its adaptability. Industries, especially in the field of technology, have also adopted the use of artificial intelligence in their services and products. Thus, the research question of this thesis is formed: How can asteroid spectra be analyzed using machine learning?

The research question is explored through two distinct tasks. The first is an attempt to provide clarity on the optimal placement of spectrophotometric filters in a way that results in the best asteroid classification success. The task will henceforth be referred to as **Task I**. Interestingly, no previous research seems to exist on the topic, resulting in a relatively wide range of chosen filter locations in studies that require them. The second task is an attempt to remove the human factor in forming classes in asteroid taxonomies by unsupervised clustering of a large dataset. The task will similarly be henceforth referred to as **Task II**.

In order to obtain and evaluate the results of the two tasks, this thesis is structured as follows. First, some background theory for asteroid spectra, available datasets, and classification methods is presented in order to lay a solid foundation to build further discussion upon. Next, machine learning and the methods utilized in this thesis are discussed, including the differences between supervised and unsupervised learning. The supervised learning method of neural networks is then applied to Task I, the optimization of spectrophotometric filters. The utilized method is discussed in detail, along with the obtained results for three, four, and five filters. After the results have been evaluated, the theory, methodology, and results of Task II, the unsupervised classification, are presented and discussed. Finally, conclusions for the whole thesis are drawn.

2 ASTEROIDS AND THEIR SPECTRA

Asteroids are interesting astronomical objects, especially because they offer a unique possibility to look back into the early stages of our Solar System. Not only are most of them relatively unaltered, particularly when compared to the planets [1], but the Earth also collects samples of them in the form of meteorites. However, to know what materials an asteroid is composed of or which asteroid can a meteorite be related to, there must be a way to analyze their characteristics. One way this can be accomplished is through studying how they interact with light.

2.1 Spectroscopy

As mentioned previously, spectroscopy is one of the main ways of studying how a material interacts with electromagnetic radiation. Because so much information about a specific object can be drawn from its spectrum, and because obtaining the spectrum can be done from astonishingly long distances away, it follows that astronomy utilizes spectroscopic methods frequently. In order to make the following sections easier to conceptualize, a brief description of how spectroscopy works, particularly with astronomical objects, shall be presented here.

There are three basic processes a sample of matter can experience when hit by electromagnetic radiation: scattering, absorption, and emission. These specific — and predictable — interaction pathways form the basis of the operation principle for spectroscopy. Typically the concept of spectroscopy involves studying the intensity of light as a function of wavelength, although it is possible to expand this discussion to frequency as well. Wavelengths are particularly important in astronomical spectroscopy, as the features seen on the spectra are caused by specific

transitions in mineral or molecular species [1]. These transitions have characteristic energies required to occur, and therefore specific wavelengths. The correlation of these specific wavelengths to the composition of asteroids is discussed in Section 2.2.3. Studying a spectrum as a continuum facilitates the simultaneous inspection of both the absorption and scattering features, as well as the spectrum's overall slope. It also allows for easy comparisons between different types of asteroids.

The general term of spectroscopy can be divided into several subsections. First and foremost, it can be considered to contain spectrometry, which refers to applying the theory offered by spectroscopy in practice to obtain quantitative measurements. Another important subsection is spectrophotometry. In astronomy, the division between spectrometry and spectrophotometry is rather fine, as the end results are typically used in the same manner and the physical phenomena governing the interactions are the same. What distinguishes them from each other is how their measurements are made; in astronomy, while spectrometry uses spectrographs, spectrophotometry uses photometers that measure the intensity of light that has been passed by a filter with a specific central wavelength and bandpass. Spectrometry is more generally favoured in collecting asteroid spectra, as it can typically measure a great portion of the visible spectrum in a single exposure and does not require as specific weather conditions as spectrophotometry does [2]. However, spectrophotometry still has some important applications, as we shall see in Chapter 5.

2.2 Classification

Since the spectral data can be used to analyze the characteristics of the observed asteroids, it is natural to wonder whether there are several objects with similar features. Grouping asteroids together can, for example, yield indications of them possibly being fragments of one bigger body or facilitate easier discussion and comparison of the abundances of different types of asteroids. As a consequence, several asteroid taxonomies have been developed after we started

collecting spectroscopic data from the asteroids. Each of the taxonomies also noted the specific features seen in the spectra and made interpretations of what their physical meaning might be. It should be noted that it is possible to classify asteroids based on other features they possess as well, but discussion of those topics is outside the scope of this thesis.

2.2.1 Taxonomies

Historically, perhaps the most well-known taxonomic system is the one proposed by David J. Tholen in 1984 [3, 4]. Consisting of 14 classes defined from data between 0.31 and 1.06 microns [4], it forms the baseline that most modern taxonomies still compare themselves to [5]. However, because most of the asteroids fell into only three main groupings in the Tholen system [4], very detailed discussion about the differences between various types of asteroids was not possible.

Consequently, another more detailed taxonomic system was eventually developed by Schelte J. Bus and Richard P. Binzel in 2002 based on the results of the second phase of the Small Main-Belt Asteroid Spectroscopic Survey (SMASS II), defining a total of 26 classes [5]. Although the wavelength range of their data was more limited (0.44 to 0.92 microns), it was of higher quality and spanned much more asteroids than the set Tholen used [5]. The increase in the quality of the data allowed, e.g., the creation of subclasses to some of the primary well-known ones, distinguishing asteroid groupings from each other on a deeper level.

However, the SMASS II taxonomy of Bus and Binzel and those before it were based on data that was either below the infrared range or barely reached it. The problem this presents is that it is more difficult to tell different classes apart with access to limited wavelength ranges, especially because many features that could be important for classification actually lie in the infrared. The complications related to infrared data are discussed in further detail in the next section. Since there was a desire to include the infrared data in asteroid spectroscopy, a system for the purpose was designed by DeMeo et al. in 2009 [6]. The DeMeo system is frequently used in modern asteroid studies [7, 8, 9], owing to its detailed and

extensive nature. The wavelengths that the system is based on range from 0.45 to 2.45 microns [6]. The classes are mainly the same as those defined by Bus and Binzel, with minor alterations and additions. The system currently includes 25 classes: A, B, C, Cb, Cg, Cgh, Ch, D, K, L, O, Q, R, S, Sa, Sq, Sr, Sv, T, V, X, Xc, Xe, Xk, and Xn [10]. This thesis utilizes the DeMeo taxonomy due to the applicability and detail it offers.

2.2.2 Datasets

Asteroids were discovered much later than the planets. In fact, the reason they were discovered is because Kepler calculated that there should be one more planet in between Mars and Jupiter that for some reason had not been observed yet [11]. Their detection taking so long is understandable, as asteroids are hard to get consistent measurements of due to, e.g., their small size, fast sky-plane motion, and possibly dark surface. Consequently, even in the modern age, we do not have much data on them. However, this is soon to change with missions like the European Space Agency's Gaia. Launched in 2013, Gaia has continued to collect data and release it in batches for scientists to scrutinize. For asteroid spectroscopy, the near future is exciting: the Gaia Data Release 3 will be made available in 2021. It will include a significant amount of data on the asteroids in our Solar System, which will help constrain their spectra considerably [12].

One of the most utilized early spectroscopic asteroid datasets is still the aforementioned SMASS II, as it took recordings of 1341 asteroids [13]. The quality of its measurements is far higher than that of its predecessor Eight-Color Asteroid Survey (ECAS) [14], which up until then had been the most used dataset for classification and analysis of asteroid spectra. However, the measurements of neither of them extend into the near-infrared wavelengths. The infrared values are considered to be crucial for modern classification taxonomies, and therefore the focus on datasets that include them has become more prevalent. Therefore, in order to compare any obtained results directly to the established modern taxonomies, a dataset that includes these infrared wavelengths must be obtained.

Acquiring infrared data for asteroids is still complicated due to the hindering effects of the atmosphere. As a consequence, sets that present infrared data for asteroids are either small in size or incorporated into larger sets that have samples that do not necessarily all extend into the infrared. Because of this, no "universally" used large-sized dataset exists that has both visible and infrared data for all asteroid samples. Consequently, for this study the primarily utilized set is a combination of two medium-sized datasets: the set used by DeMeo et al. in their 2009 paper [6] (here referred to as BDM09) and the MIT-Hawaii Near-Earth Object Spectroscopic Survey (MITHNEOS) dataset published in 2019 [10].

In order to develop this new set (referred to from now on as VISNIR), several steps need to be taken. The procedure was planned and executed together with Antti Penttilä from the University of Helsinki and is described in full in the Appendices. VISNIR can be split into two different representations: one that is simulated and one for k-means. The construction of the basic reduced VISNIR set that both are based on begins by combining the sub-classes into their main equivalents. In practice, this means reducing classes like Sa, Sq, Sr, and Sv to simply the S-class.

Out of the main classes, those that have under four samples are removed, as such a low number of samples would represent the class poorly. Additionally, any sample that is considered to be a clear outlier within its class is removed. This brings the base set size to 582 samples. For the simulated set, synthesized samples are added to each main class until each has 200 of them. The synthesized samples are formed by first adding small amounts of noise to the original samples' principal components and then converting them back to the full spectra. The synthesization process is vital, as there would otherwise be too few points to represent each class adequately for machine learning applications. For the k-means dataset, the main classes that were not eliminated due to having too few samples are converted back into the full subclasses. Some additional outliers are also removed in order to improve the algorithm's performance.

Table 2.1 illustrates the abundances of asteroids in the reduced classes of the base VISNIR dataset. It is clear that there are some imbalances in the frequency

of objects in the classes. For example, over half of the samples belong to the S-class, while the two least populous classes, A and T, make up less than 2% of the total population together. The imbalance must be considered when designing the machine learning methods, as the results can develop a strong bias towards the S-class due to its overwhelming majority if one is not careful. This problem is solved in the simulated VISNIR set since each class has the same number of samples, but still remains a concern for the relatively unaltered k-means set.

Table 2.1. *Abundances of asteroid samples in each reduced class in the VISNIR set.*

Reduced Class	Samples	Share (%)
A	7	1.20
B	12	2.06
C	60	10.31
D	21	3.61
K	15	2.58
L	33	5.67
Q	43	7.39
S	309	53.09
T	4	0.69
V	28	4.81
X	50	8.59

Figure 2.1 holds visualisations for the shape of all the spectra in the VISNIR dataset. The figure is attached here not only to illustrate the general shapes of the reduced classes, but also to show the considerable variance between the samples in each of them. Even classes that were not particularly populous, like A and D, show significant differences between the shapes of the objects within those classes. The differences form another important point to keep track of when the machine learning algorithms will be applied to the set — how will they handle within-class variation? The consideration of the characteristics of the utilized dataset is crucial, as machine learning methods are heavily reliant on the quality and abundance of the data they are provided.

What should also be noted in Figure 2.1 is how some classes are naturally more distinctive in shape than others. A, D, and V have shapes that are easy to recognize and that differ significantly from the overall shapes of the other classes. On

the other hand, there are classes that are harder to distinguish by visual inspection. For example, distinguishing B from C is only possible through consideration of the spectra's slopes. C and X are also remarkably similar in shape. These factors will present additional challenges to particularly the unsupervised algorithm in Task II, as it should be able to tell different spectra apart well enough to produce meaningful new classes.

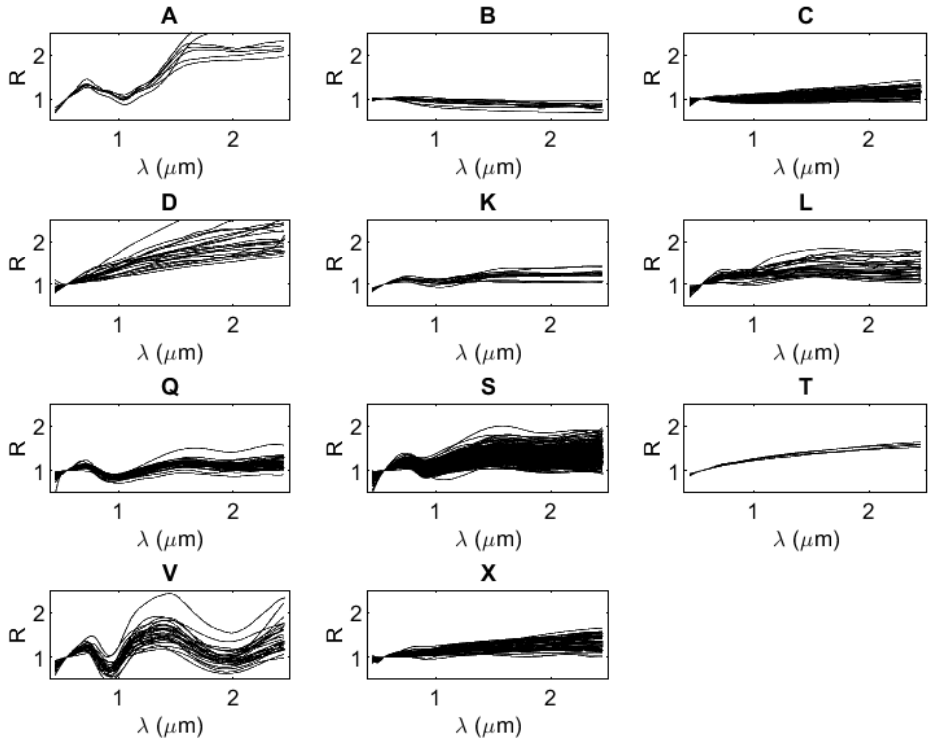

Figure 2.1. *Illustration of the shapes of all spectra in the VISNIR set, split into the correct reduced classes. The X-axis holds the wavelengths from 0.45 to 2.45 microns, while the y-axis is the reflectance normalized at 0.55 microns.*

2.2.3 Features and Mineralogy

The asteroids are split into the taxonomic classes based on the features found in their spectra. But what causes these features, and what are they? As mentioned in Section 2.1, materials have specific energies required to excite electrons within them, and these energies can be correlated to specific wavelengths. In asteroid spectroscopy, the composition is derived from the spectral features seen in the

sunlight that they reflect. For asteroids, these features are primarily caused by absorption and show as dips in reflectivity in the spectrum. While they are not a specific feature, the slopes of the spectra can also be connected to surface details of the asteroids. For example, there is thought to be a correlation between the spectral slope and space weathering [15], which is important to make note of as it transforms the optical properties of the target surface.

The taxonomic system this thesis follows is based on the features listed in Table 2.2. It is easy to notice that most of the significant features are in the near-infrared wavelengths, illustrating the impact they have on how the taxonomic system was formed. Naturally, however, the visible spectrum is still important to include, as it does have some important details and helps determine the overall slope of the spectra better. It is also interesting to see where in the wavelength range the features are generally located, as they can later be compared to where Task I places the optimized filters. The sections that evaluate the underlying reasons behind the obtained results will extend the discussion presented here into what particular elements might cause the significant features that affect the final results.

2.2.4 Classification Methods

While all spectroscopic taxonomies fundamentally have the same objectives, the way the final system is designed varies. There are several methods to make the final decisions on what basis asteroids are clustered together, and how many clusters there should eventually be. The primary method in use today is principal component analysis (PCA). It includes reducing the dimensionality of the dataset by transforming it into principal components, which behave as summaries of features in the set [16]. The transformation is particularly useful for spectroscopic applications, as the datasets they use are often high-dimensional due to the required detailed covering of the wavelength range they utilize. All of the three main taxonomic systems in use today were formed with PCA [3, 6, 17]. They used approximately the first five principal components, particularly the first two, which are plotted against each other in order to investigate groupings in the data.

Table 2.2. *List of features for the classes that have been retained in some form in the VISNIR dataset. Adapted from DeMeo et al. [6] and Binzel et al. [10].*

Class	Features
A	Deep and broad absorption band with minimum near 1 µm, very high slope, may or may not have a shallow absorption band at 2 µm
B	Linear with a negative slope, often has a slight bump around 0.6 µm and/or a slight up concaving curvature between 1 µm and 2 µm
C	Linear with a neutral VIS slope, often has a slight bump around 0.6 µm and low positive slope after 1.3 µm
Cb	Linear with small positive slope beginning around 1.1 µm
Cg	Small positive slope beginning around 1.3 µm with a pronounced UV dropoff
Cgh	Small positive slope beginning around 1 µm and pronounced UV dropoff with a broad shallow absorption band centered near 0.7 µm
Ch	Small positive slope beginning around 1.1 µm and slight UV dropoff with a broad shallow absorption band centered near 0.7 µm
D	Linear with very steep slope, sometimes with slight curvature or kink around 1.5 µm
K	Wide absorption band centered right after 1 µm, left maximum and minimum sharply pointed, walls of the absorption feature are linear with very little curvature
L	Steep slope in VIS leveling suddenly around 0.7 µm, gentle down concaving curvature in IR with maximum around 1.5 µm, may or may not have an absorption band at 2 µm
Q	Deep and rounded absorption feature at 1 µm along with a significant absorption feature at 2 µm
S	Moderate 1 µm and 2 µm features
Sa	Deep and very broad absorption band at 1 µm, otherwise has similar features to A but is less red
Sq	Wide 1 µm absorption band with evidence of a feature near 1.3 µm (like Q)
Sr	Narrow 1 µm feature as well as a feature centered around 2 µm
Sv	Very narrow µm absorption band as well as a feature centered around 2 µm
T	Linear with moderately high slope, often concaves down
V	Very strong and narrow 1 µm absorption band as well as a strong 2 µm absorption feature
X	Linear with medium to high slope
Xc	Low to medium slope, slightly curved and concaved downward
Xe	Low to medium slope as well as an absorption feature shortward of 0.55 µm
Xk	Slightly curved and concaved downward with a faint feature between 0.8 to 1 µm
Xn	Relatively flat with a feature centered at 0.9 µm

When datasets were still limited in size, PCA was not needed. Examples of this are taxonomies that were developed even before that of Tholen's. In 1975, Chapman et al. created the first taxonomy that divided asteroids into two groups based on their composition [18]. The two groups were derived from clusters that were formed when polarimetric, radiometric, and spectrophotometric parameters were plotted against each other in a series of two-parameter plots. Bowell et al. aug-

mented this system in 1978 by increasing the taxonomy's size to five classes [19]. Similarly, they plotted seven optical parameters against each other, alternating between which parameter was plotted against which other one. They then investigated what clusters seemed to appear within the formed figures.

What all the systems have in common is how their formation heavily employs visual inspection. For example, the placement of an object in the PCA plot can clearly show what the potential class for it should be, but sometimes this placement is vague or overlaps with other classes. Therefore, often the spectrum of the object itself must be visually inspected and compared to the potential classes the initial method suggests for it. Human intervention is required for making final decisions on what class seems the most suitable based on the designed system rules, because the rules themselves can be rather vague. Task II outlined in the introduction of this thesis explores removing this human factor from the design process of an asteroid taxonomic system.

3 MACHINE LEARNING

The rapidly developing world has an ever-growing desire for automation. Ever since humanity invented computers, the search for the capability of them doing increasingly complicated tasks has continued. However, ultimately a surprising reversion of expectations was met; while it was relatively easy for computers to do tasks like calculation, they struggled with cognitive tasks that are intuitive for us humans, like recognizing objects and identifying them [20]. In order to solve this problem, a way for computers to "learn" from the data they were presented had to be developed, and as a consequence, the field of machine learning was born. This chapter is included in the thesis to provide a brief overview of the theory and methods behind machine learning. It also introduces the categories that the methods applied in exploring the two tasks belong to. This lays the foundation for the more intricate discussion of the utilized method details that the chapters dedicated to the tasks themselves provide.

3.1 Supervised Learning

Machine learning can broadly be split into two categories: supervised and unsupervised learning. Out of the two, supervised learning is considered to be much easier, which is partly the reason more research has gone into it. It involves the process of attempting to deduce concepts from the provided training data. The training data has two parts: the data samples themselves, as well as the corresponding labels that tell the algorithm what the sample is supposed to be. The algorithm then constructs some kind of mapping function or model that is conditioned to the training set it has been provided [21]. Testing how well the features

in the data have been learnt is possible by giving the algorithm an unlabeled test set after the training process and seeing how well its predictions of the labels match reality.

The main task of supervised learning is classification. One popular example of such a classification process is that of recognizing objects or features in images. Although the implementation of the learning process is easier than that of the unsupervised, a considerable amount of time goes into preparing the data. This is because the data typically must be of very specific form and must have all the corresponding labels so that the classification can succeed. Preparation of massive data sets, particularly of images, with the correct labels is one of the big obstacles machine learning faces today [22].

3.2 Unsupervised Learning

Unsupervised learning, on the other hand, gains its name from the fact that it is not given labels that correspond to the data samples. Therefore, the algorithm must make the decisions on how to divide — and optionally classify — the data independently. Common tasks belonging to unsupervised learning are clustering [23, 24, 25] and outlier detection [26, 27]. The discussion here focuses on clustering due to its popularity. It is also particularly relevant for this thesis, as it is utilized in Task II.

The relative difficulty in implementing unsupervised learning successfully arises from the fact that it is much harder to quantify how well the algorithm succeeded. In supervised learning it is possible to directly determine how the algorithm performed through cross-validation, but with unsupervised learning, it is the user's responsibility to interpret, e.g., the quality of the formed clusters. Often it is also left to the user to determine parameters for the algorithms, such as the number of clusters, initial points, or distance metrics to use. Choosing the most appropriate parameters is often a challenge in itself, and directly affects the performance of the algorithm [21].

4 CLASSIFICATION USING NEURAL NETWORKS

There are several widely used methods for machine learning. One of the most well known today is the neural network, partly due to its sudden rise in popularity some years ago as it started to break records in performance [20]. One of the major advantages of employing neural networks in machine learning is their robust ability to handle several kinds of classification tasks. It is then natural to wonder how well a neural network would be able to classify asteroids under supervised conditions. This forms the focus of this section: A neural network is given the simulated VISNIR dataset along with labels that determine the classification they were given by DeMeo et al. [6].

The dataset is divided so that 70% of the samples are used for training, 15% are used for validation, and 15% are used for testing. What makes this case supervised learning is the fact that a training set, together with its corresponding labels, is provided to the neural network. The neural network then uses the training set to learn the features in the data. Reserving some samples for the validation set is important, as it is used to avoid overfitting in the network, and therefore control the training error [28]. Finally, the set that was kept for testing is used to determine how well the neural network succeeds in classification after it has been trained.

4.1 Utilized Network Structure

There is no automatic way to find the best structure for the neural network; it is left to the user to decide the parameters that define how many layers the net-

work has, what functions those layers utilize, and how many neurons are on each of the layers. The general structure utilized in this thesis is a specific type of feed-forward network, generated by the `patternet` algorithm provided by Matlab's Deep Learning Toolbox. A feed-forward network is a structure in which the connections between the neurons cannot and will not form a cycle, meaning that information only flows in one direction [29]. There are three primary types of layers in this network: the input, the hidden, and the output layers.

The input layer is essentially an $s \times 1$ vector, where s is the number of features each sample has. Each of the feature elements then connects to the neurons on the next layer. Typically some kind of data pre-processing also takes place within the neural network, and these processes can be attributed to take place in the input layer. A common example of such pre-processing is deciding how to handle any possible unknown inputs. The size of the input layer can vary greatly between applications based on the utilized dataset's dimensionality, which consequently affects the design of the hidden layers.

The hidden layers contain the neurons that the user must decide the amount of. These neurons produce an output by utilizing the weighted inputs, biases and activation functions through the equation

$$\mathbf{a} = \mathbf{f}(\mathbf{W}\mathbf{p} + \mathbf{b}), \tag{4.1}$$

where the column vector $\mathbf{a}$ is the output of a layer, $\mathbf{f}$ is the layer's activation function, $\mathbf{W}$ is the weight matrix, $\mathbf{p}$ is the input vector, and $\mathbf{b}$ is the bias vector. The operational power of neural networks lies with the capability to adjust the weights and "learn" the features of the data this way. The adjustment process takes place in the training phase, where the network can fine-tune the weights based on the provided samples with their corresponding labels. The user can also choose to increase the amount of hidden layers, if the classification task seems to require it. In that case, the preceding layer's outputs become the next layer's inputs, and again get multiplied by weights and have biases added to them.

The activation function (or alternatively the transfer function) that all the neurons utilize in the hidden layers in this implementation is the hyperbolic tangent sigmoid function, tansig. It is a specific case of the sigmoid function and mathematically equivalent to $\tanh(x)$

$$f(x) = \tanh(x) = \frac{\mathrm{e}^{x} - \mathrm{e}^{-x}}{\mathrm{e}^{x} + \mathrm{e}^{-x}}, \tag{4.2}$$

which ranges from -1 to 1, making it a scaled and shifted version of the logistic function. Sigmoid functions are some of the most widely used activation functions in neural networks, mostly due to their simplicity and the fact that they are differentiable with a positive derivative everywhere [30]. The differentiability is a key aspect in neural network design, since it facilitates the ability to optimize the performance in a robust manner.

The output layer functions mostly like the hidden layers in this implementation, but unlike the hidden layers, it applies the $\mathrm{softmax}$ function to its inputs. The result is that all the inputs are converted to true probabilities that add up to 1. Based on these values it is possible to choose the value with the highest probability as the assigned class. Because the network structure and the task in this case are relatively simple, in a well-performing case it is typical to see one of the values being close to 1 and the others very close to 0, implying that the network is very sure about the label it is going to assign to that particular sample.

The basic structure for this network is described in Figure 4.1. The figure includes the three outlined layers, as well as the connections between and within them. The illustration here is simplified in the sense that connections are described as vectors or matrices. This allows for the focus to be on the overall processes taking place in the structure instead of the individual components. As mentioned previously, the size of the input layer depends on the number of features in the dataset. Here the number of features is equal to 200. *Each* individual feature connects to *each* of the neurons on the hidden layer with a unique weight. Therefore, an overall weight matrix $\mathbf{W}^1$ of size 40×200 exists "in between" the first two layers. The superscript is included in order to discriminate between the arrays in

different layers. There are 40 rows, since that is the number of neurons on the hidden layer in this study. The fact that it has 40 neurons, and that there is only one hidden layer, is given basis for in Section 4.2.

Each of the hidden layer neurons, which are connected to all the input features as described above, forms a connection to the layer's activation function after it has added up the weights and the bias it receives. These connections together form the 40×1 vector $\mathbf{n}^1$ in Figure 4.1. Since each neuron has its own bias, they can be generalized into a 40×1 vector $\mathbf{b}^1$. Each layer also produces a final output, which is represented by $\mathbf{a}$. The components $\mathbf{p}$, $\mathbf{W}$, $\mathbf{b}$, and $\mathbf{a}$ listed here are the constituents of Equation 4.1.

The output layer has as many neurons as there are classes to place objects into. In this study there are 11 reduced classes, and therefore 11 neurons are utilized on the output layer. Although typically of different size and utilizing a different activation function, the output layer's operation principle is very similar to that of the hidden layers. The individual final outputs are included in vector $\mathbf{a}^2$, which consists of the probabilities between 0 and 1.

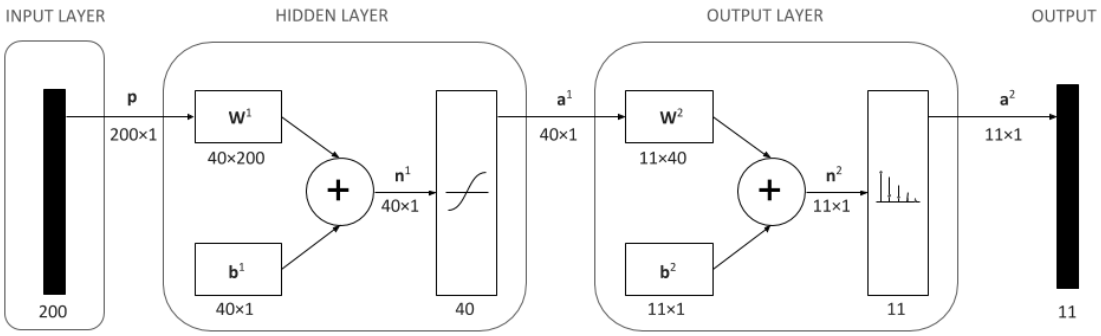

Figure 4.1. *Illustration of the neural network structure utilized in this thesis. The input layer provides the hidden layer 200 features for each sample, connecting through different weights to the 40 neurons. The output layer has as many neurons as there are output classes. In this study this is equal to 11.*

In addition to the layer activation functions, the user must determine the training function the network should use. Its purpose is to facilitate the process of "learning" in the network by minimizing the global error function, which depends on the

weights in the network [31]. In this thesis, the neural networks utilize the scaled conjugate gradient function. Introduced by Martin Møller in 1993, it is well-suited for large-scale problems and functions faster than several other conjugate gradient methods [32]. Another crucial function is the performance function, which determines the network's performance based on the provided targets and actual outputs. In this study, the performance evaluation is made with cross entropy.

4.2 Fine-Tuning the Structure

When considering hidden layer neuron amounts, a general guideline that exists suggests that the ideal number of neurons is typically situated between the values of the input and output [33]. In this study, the size of the input is 200, and the size of the output is 11. The previous section noted that the number of neurons on the single hidden layer is 40, but this is not an obvious choice from the beginning: it must be chosen through some process.

The theory of the optimal neuron amounts being between 11 and 200 is tested in Table 4.1, where the structure of the network is varied and the success of the network in classifying the simulated VISNIR objects is recorded as a mean over 500 runs. The neuron amounts of 5 and 10 are included, even though they are below 11, in order to test the validity of the statement and ensure truly choosing the optimal amount. The same number of neurons is tested in different layer configurations to also determine how many hidden layers the study should utilize. The maximum number of hidden layers to test is determined here to be four.

A trend that can be noted in Table 4.1 is how designs that have split the neurons into several hidden layers succeed worse than their single-layer counterparts. This is unsurprising, since a relatively simple pattern recognition task of this type should succeed well with only one layer. In fact, at least in feed-forward networks, having many layers usually has no advantages [34] and only leads to increasing the training time [35]. Clearly, the best structure for Task I, which uses the simulated VISNIR set, is one with a single hidden layer. Figure 4.2 illustrates the success rates for different numbers of neurons when only one layer is utilized.

Table 4.1. *Comparison of different network structures, in order of increasing network size. Included are the success rate for the number of neurons as a mean after 500 runs, the standard deviation of the rates, and the time elapsed for the repeats. In addition to varying the neuron amounts, different numbers of layers are also tested. It is easy to note that structures with several hidden layers have comparably lower success rates and higher standard deviations.*

Test	Layer 1	Layer 2	Layer 3	Layer 4	Success (%)	Standard Deviation	Time (s)
1	5	-	-	-	90.286	7.114	504.070
2	5	5	-	-	84.581	8.433	621.090
3	10	-	-	-	95.972	3.370	713.234
4	5	10	-	-	89.914	6.837	652.032
5	5	5	5	-	75.311	11.802	706.804
6	15	-	-	-	97.269	1.272	821.021
7	5	5	5	5	64.024	15.609	738.891
8	10	10	-	-	94.834	4.699	871.665
9	20	-	-	-	97.421	3.412	834.320
10	10	10	10	-	93.540	6.410	976.451
11	30	-	-	-	98.008	0.950	933.891
12	10	10	10	10	90.803	7.839	1127.550
13	40	-	-	-	98.084	0.926	992.397
14	50	-	-	-	98.186	0.941	1066.534
15	100	-	-	-	98.161	1.038	1407.042
16	200	-	-	-	97.658	2.141	2290.561

Based on the results in Table 4.1 and the illustration in Figure 4.2, it is easy to see that even with this amount of testing, determining the best structure to use is not straightforward. However, out of the options tested, the network structure of one layer with 40 neurons will be chosen. The reason for this is the fact that it reaches a very good success rate while also having the lowest standard deviation. Values above it have slightly better success rates, but the difference of approximately 0.1 percentage points is not significant enough to warrant choosing them as their running times are longer, not to mention their higher values of standard deviation. There is a clear trend in increasing success rates and decreasing standard deviation in the single-layer structures up until around 40 neurons, but there the trend starts to almost flatline, as can be seen in Figure 4.2.

In addition, the success rate of the 40 neuron structure will form a baseline that all further tests can be compared to. This is possible because it is the "best" success rate that any attempt at classification can produce in this study due to it having

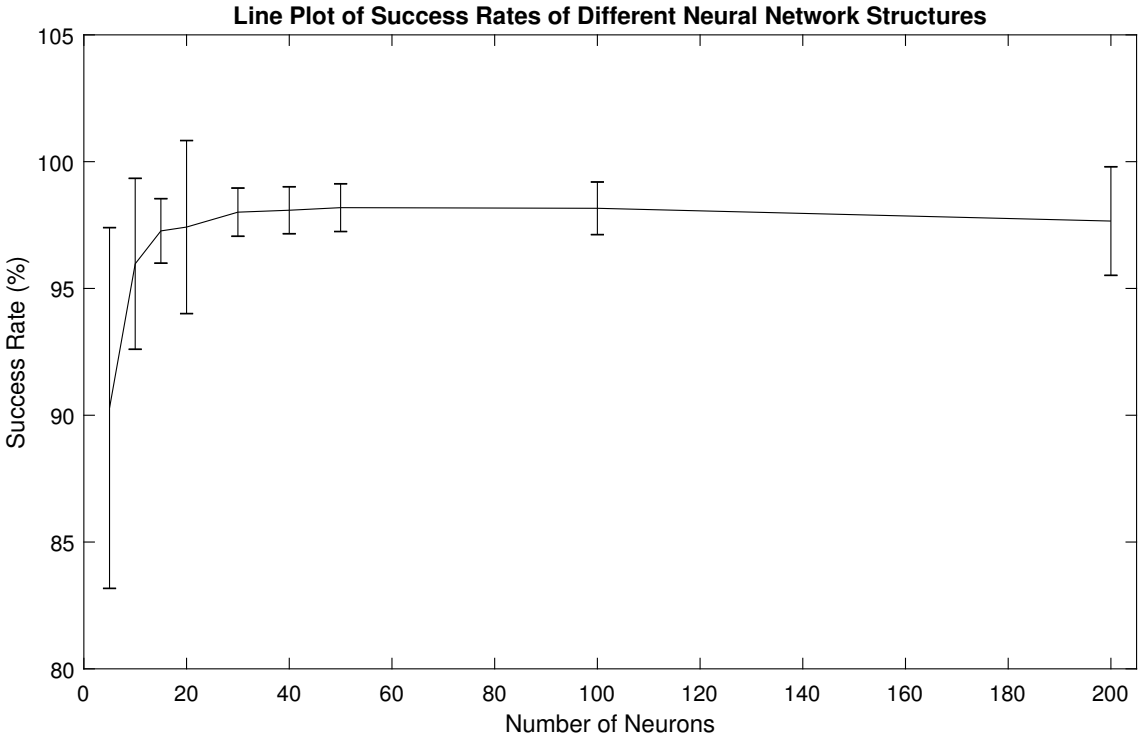

Figure 4.2. *Line plot of the neural network's success rate against the number of neurons on its hidden layer. The error bars come from the calculated standard deviations of each point. Note how they get larger at the beginning and the end of the neuron number range.*

access to the full dataset with the optimal structure. More sizes and distributions of the neurons could have been tested, but the amounts seen here are deemed satisfactory based on the fact that the multilayer structures have lower success rates and the rate of success is not likely to increase far above 98% under any circumstances within this study.

It should be noted that the extremely high success rates reached here are partly due to the simple form of the problem. In addition, the success rate is increased by the fact that the amount of classes has been reduced to 11 and simulated cases have been added to increase the sample size. The large number of samples versus classes makes it easier for the network to succeed well, because it can access many examples of each class in the training process, and many of them are quite similar due to the simulation process. These features make this study somewhat "idealized" when contrasted with reality, but are nonetheless necessary to ensure the quality of the results of any further implementations that require access to the network, as described in the next chapter.

5 TASK I: FILTER OPTIMIZATION

5.1 Motivation for the Use of Filters

As was previously explained, asteroids can be categorized based on their reflectance spectra. These spectra are rather continuous now, but this has not always been the case. Particularly before the charge-coupled-device (CCD) camera became popular and accessible as a spectrometric measurement device, spectrophotometric studies were accomplished through the use of carefully selected filters. The locations for the filters were chosen so that they studied some specific features or a more limited wavelength range. Consequently, there has been a considerable amount of variance in the chosen locations and filter bandpasses in past studies, and none of them were designed to study a large wavelength range for the specific purpose of being able to distinguish different classes of asteroids from each other. In order to illustrate this void in more detail, some example cases are summarized in Table 5.1.

Table 5.1. *Example cases of filter positions, rearranged to be in increasing order from left to right. Each filter location is represented by its central wavelength λ_n, where n is the filter number. Full-width-half-maxima (FWHM) are also given in all applicable cases.*

Study	λ_1 / FWHM$_1$ (µm)	λ_2 / FWHM$_2$ (µm)	λ_3 / FWHM$_3$ (µm)	λ_4 / FWHM$_4$ (µm)	λ_5 / FWHM$_5$ (µm)	λ_6 / FWHM$_6$ (µm)	λ_7 / FWHM$_7$ (µm)	λ_8 / FWHM$_8$ (µm)
ECAS [14]	0.337 / 0.047	0.359 / 0.060	0.437 / 0.090	0.550 / 0.057	0.701 / 0.058	0.853 / 0.081	0.948 / 0.080	1.041 / 0.067
NEAR [36]	0.462 / 0.023	0.554 / 0.024	0.700 / 0.133	0.755 / 0.019	0.900 / 0.033	0.951 / 0.038	0.996 / 0.044	1.033 / 0.051
SDSS [37]	0.350 / 0.060	0.480 / 0.140	0.625 / 0.140	0.770 / 0.150	0.910 / 0.120	-	-	-
FC [38]	0.438 / 0.040	0.555 / 0.043	0.653 / 0.042	0.749 / 0.044	0.829 / 0.036	0.917 / 0.045	0.965 / 0.086	-

Let us discuss the cases from Table 5.1. ECAS is possibly the most famous example of the use of filters for photometric asteroid research. A notable part of its fame rises from the fact that the dataset obtained from the ECAS measurements is what Tholen used to develop the first widely used taxonomic system for asteroids [3], briefly described in Section 2.2.1. However, as seen in Table 5.1, the wavelength range in ECAS did not extend far into the infrared, and therefore can not be robustly used in classifying asteroids with the modern taxonomies. The Near Earth Asteroid Rendezvous – Shoemaker (NEAR) mission was the first to orbit around an asteroid for measuring purposes. Although it had the capability to do continuous spectroscopy, the eight listed filters were included in order to search for the iron-bearing silicates on the surface and to provide a better sensitivity for starfield exposures [36]. As such, its filters were designed for only very limited features.

While the Sloan Digital Sky Survey (SDSS) was primarily designed for detecting stars and galaxies, it also recorded a significant amount of data from asteroids [39]. It is an example of a study that *did* record a sizeable amount of asteroid data, but had filters that were not particularly designed for that specific purpose. Finally, the Dawn mission's Framing Camera (FC) had filters that were specifically designed for studying asteroids. However, once again the mission's objective was to analyze only two asteroids [38], signifying that the filters would not necessarily represent a good general set for studying larger populations with.

An interesting point to note with the placement of the filters themselves seen in Table 5.1 is that they can extend further into the ultraviolet range than the nowadays more widely used CCD, which typically stops around 0.4 microns [2]. The capability to penetrate the ultraviolet is a possible benefit of using filters as opposed to CCD, even though the values in the ultraviolet range are not frequently used in classification systems today. In fact, the focus on longer wavelengths has fundamentally changed the more recent classifications. For example, Bus and Binzel's 2002 classification system had to combine Tholen's F-class and B-class together, because they could not be properly distinguished from each other with the poor quality ultraviolet data that CCD measurements record [7].

Based on the values in Table 5.1, it is indeed clear to see that currently there does not exist a "one-size-fits-all" set of locations for filter placements for asteroid spectrophotometry. While this is partly caused by the different needs the studies have, it still leaves room for considering whether an optimal set of filter locations exists, at least for taxonomic asteroid classification. A way to search for these locations is to utilize the previously introduced neural networks accompanied by an optimization algorithm, as can be seen in the following sections.

5.2 Optimization System

Taking the VISNIR dataset as an example, the range where the filters can be placed runs from 0.45 to 2.45 microns. Let us first consider a scenario where the desired outcome is a set of locations for five filters that can be used to classify asteroids best. Problems instantly arise if determination of the locations is attempted manually: finding sets of locations through considering all the different possible combinations would give results in the millions, which would take unnecessarily long to test. Therefore, in order to find the best locations in a reasonable amount of time, a more intelligent and automated process is required. This leads to the introduction of an optimization algorithm into the system. The algorithm's purpose is to iterate and optimize through several sets of possible locations for the filters and keep refining itself until it can choose the set of locations that is the best for classification purposes. The extent of each set's success is determined with a neural network that the optimization algorithm has free access to.

5.2.1 Optimization Algorithm

Although Matlab has a rather extensive selection of algorithms that can be used for optimization tasks, there are some constraints that the problem it is chosen to solve poses. Primarily, the algorithm must be able to work within a set of constraints, since the applied dataset will only have reflectivities for a certain wavelength range. The algorithm must also be able to simultaneously optimize a set of

values within this range. Based on these criteria, the two algorithms chosen for consideration are `fmincon` and `patternsearch`.

During testing it was quickly discovered that `fmincon` has a tendency to return sets with a relatively high amount of variance in success rates. It also possesses a tendency to sometimes place filters on top of each other. Therefore, for robust operation with `fmincon`, additional constraints have to be introduced into the algorithm to keep the filters apart, which means that freedom of choice for the locations is partly sacrificed. Similar tests were also run with the `patternsearch` algorithm to see how its results would vary from those of `fmincon`. It was discovered that while `patternsearch` takes longer to run by default, it also returns more consistent success rates and almost never places filters on top of each other. Consequently, `patternsearch` is chosen as the optimization algorithm for this task.

The principles of how `patternsearch` works are the following. First of all, the algorithm must be supplied with a function to optimize, the inputs of which it varies so that the output is minimized. Since this study is looking for the highest success rate, the implication in practice is that the function multiplies the result by -1 so that the result is "minimized". In this implementation, the algorithm is given initial starting points, placed at the beginning of the wavelength range. In addition, the beginning and endpoint of the wavelength range — which form the constraints for the optimization process — are determined. The operation principle of the algorithm is such that it computes the objective function's value at several mesh points based on the point it is currently at (e.g., in the beginning, the point is the supplied initial point). From the calculated mesh points it will then choose the one that is smaller than the value at the current point. Finding a smaller value is counted as a success and leads to the mesh size being multiplied by two. However, if no mesh points have a smaller objective function value, the poll is counted as being unsuccessful and the mesh size is multiplied by 0.5 until another success is reached or the algorithm stops running due to meeting one of its end conditions.

In this implementation, it is possible to choose any number of filter locations for the algorithm to optimize. However, for purposes of staying true to reality, the filter amounts are kept between three and five. It is also possible to choose how many repeats of optimizations will be run in a row. This makes it possible to leave the program running without supervision for tests that last a long time or require multiple repetitions for ensuring the quality of the results.

5.2.2 Designed Optimization and Direct Test Neural Networks

The goal of optimizing the filters is to find the best set of locations for use in classification. For this reason, the output of the optimization algorithm's objective function must be the rate of classification success for different location inputs. Because the algorithm itself has no way of deducing the success rate, it must get aid from a neural network that determines and returns this value to it. The optimization neural network (ONN), created purely for this study and as such not a standard term used in machine learning, resides within the algorithm's objective function. However, before any neural network can start classifying the data, several steps must first be taken to prepare said data.

First of all, the continuous initial spectra must be reduced to a form that more closely resembles values that would be obtained with only a few filters. In this study, the filters are simulated as quite ideal, having a Gaussian distribution with a location functioning as the central point and with a predetermined full-width-half-maximum. The utilized FWHM in this study is 0.05 microns, since it is a good simplified average value of the FWHM of filters seen in Table 5.1. The data reduction is executed by first finding the probability density function (PDF) of the normal density function with mean μ and standard deviation σ, evaluated at each of the wavelength range values. The mean is the location of a specific filter. The standard deviation is calculated from the chosen FWHM. To relate the PDFs to the real initial data, a convolution of it and the real data is taken for each sample.

The convolution process can be modeled with

$$c = \frac{\mathbf{r} \cdot \mathbf{d}}{\sum_{j=1}^{200} d_j}, \tag{5.1}$$

where c is the convoluted data for a specific filter and sample, $\mathbf{r}$ is the vector of real data for that sample containing the reflectivity values, $\mathbf{d}$ is the vector containing the probability density function evaluated at each value in the wavelength range for that location, and d_j is an individual PDF value in vector $\mathbf{d}$. Because a dot product is taken between $\mathbf{r}$ and $\mathbf{d}$, the output is a scalar. The convolution process is repeated with all the filter locations for each sample in the original dataset. The result is a new set of data that simulates results that would be obtained using filters placed at the chosen locations.

For the optimization algorithm to be able to use the optimization neural network directly during its operation, some changes to the standard neural network described before have to be introduced. Principal among these is the addition of a system that alters the network so that the returned success rate is not the success rate of a single run, but a mean taken over several ones. This is an absolutely necessary addition, because the success rate of the neural network varies naturally by a few percent if it is allowed to always begin from a different set of starting values. The randomness of starting values is necessary to ensure unbiased results, and therefore must be kept as a feature of the network. If the variance of the success rates is not eliminated or at the very least considerably reduced, the quality of the algorithm's performance suffers, since it cannot determine precisely what locations give better success rates than the ones it has previously encountered. For the main measurements obtained in this study, the amount of repeats done in the ONN are 10, 30, and 50 in order to evaluate how the performance varies. These different amounts are important to explore in the study, since there is no pre-determined way to decide which number of repeats is the optimal amount for computation efficiency and classification success.

The text above described the structure and implementation of the neural network that the optimization algorithm utilizes. However, for testing the validity of the

results, a *direct test* network (DNN) is developed as well. The direct test neural network functions as a method to confirm the success rate of each suggested combination of filter locations over a large number of iterations to determine which combination is actually the best for classification success. It is given either one or several sets of locations to test. It first goes through the same convolution process as the ONN, and then tests the data it obtained for a given set against known labels. The amount of repeats to run before reporting the direct success rate is defined as 500, since it is a good compromise between stabilizing the results to an adequate degree and keeping running time within reasonable bounds. The DNN is functionally very similar to the ONN, but the distinction is drawn here to separate the version of the neural network that the user can utilize freely with any given set of filter locations and the version that the optimization algorithm uses inside its objective function. The DNN also repeats its process considerably more times than the ONN.

In addition to validating the success rate of entire sets of locations, the DNN can be used to investigate which filters in the set are more important than others by simply omitting them from the list of locations given to the network and then seeing how the success rate changes. This allows for acquiring insights into what features the network considers to be of more interest than others and can be used to "sanity check" the values that, e.g., three filter locations drop out from the full set of five. Examples of the direct success for a set and order of importance tests of the filters can be seen in the following sections.

Since both the ONN and DNN use convoluted data, the size of the input vector provided to the neural network is reduced from 200 to the number of utilized filters. In order to evaluate what the best number of neurons to have on the hidden layer is in this case, a similar process to that described in Section 4.2 is undergone. The obtained results are presented in Appendix B. If a set of five well-succeeding filter locations is used, the optimal number of neurons to have on the layer is still 40, as was the case when the full data was used. Therefore, both the ONN and DNN have 40 neurons on their hidden layers.

5.3 Filter Placement Results

This section discusses the results obtained by applying the proposed optimization system design. The running time for some of the tests is fairly long, even with the restrictions that were imposed in order to limit it, causing them to sometimes last for several days. The results for all the tests that are not directly shown in this section can be found in Appendix C and Appendix D with short descriptions. The focus here is on simulations executed with 30 repeats in the ONN, as they are good averages that can optionally be contrasted to the simpler case with 10 repeats or the usually slightly better case with 50 repeats.

The dataset the filters are placed into is the simulated VISNIR set. This is because if the reduced VISNIR set is used, the placement of the filters gains bias from classes that dominate the set, like the S-class. Simulating an equal number of samples for each class ensures that the results are more unbiased. However, what should be considered is that this kind of artificial data makes the classification process easier, which raises the success rate to levels that might not be as easily achievable when working with purely real data.

5.3.1 Results for Five Filters

Let us first evaluate the results obtained when the optimization algorithm is asked to place five filters into the wavelength range. As an example, Table 5.2 lists 10 sets of locations recorded when 30 repeats are made in the ONN. Each test records 10 sets in order to test whether there is variance in the suggested locations between optimization runs. With 30 repeats the algorithm seems to slightly struggle with deciding where to place the first filter, although the majority are placed in the 0.45 – 0.55 microns range, demonstrated by the value of the mean location for that filter. When the success rates of the different sets are compared, it is also clear that the values that start in these shorter wavelengths are superior in success rate to those that start around 0.70 microns. The indecisiveness, however, fades away with 50 repeats, as seen in Table C.2 in Appendix C.

Table 5.2. *Placements of five filters when the optimization neural network takes the mean after 30 repeats. Each filter location is represented by its central wavelength λ_n, where n is the filter number. Sets are listed in order of increasing optimization neural network success rate.*

ONN Repeats: 30, DNN Repeats: 500							
Set	λ_1 (µm)	λ_2 (µm)	λ_3 (µm)	λ_4 (µm)	λ_5 (µm)	ONN Rate (%)	DNN Rate (%)
1	0.700	1.044	1.575	2.138	2.450	93.599	92.766
2	0.763	1.075	1.450	1.075	2.450	93.671	89.335
3	0.450	0.763	0.950	1.075	2.388	96.370	95.852
4	0.450	0.700	0.950	1.106	2.450	96.412	95.900
5	0.450	0.700	0.952	1.106	2.450	96.500	95.945
6	0.454	0.731	0.950	1.071	2.450	96.582	95.984
7	0.450	0.700	1.075	1.450	1.824	96.620	96.049
8	0.450	0.700	1.013	1.200	2.013	96.714	96.093
9	0.513	0.763	1.075	1.450	1.950	96.883	96.147
10	0.550	0.762	0.981	1.200	2.450	97.503	97.064
Mean	0.523	0.794	1.097	1.287	2.287	96.085	95.113
SD	0.116	0.143	0.226	0.332	0.252	1.331	2.315

What should particularly be noted in Table 5.2 are the relatively high values of standard deviation (abbreviated as SD). This indicates that there might be some wavelengths that the optimization neural network does not really need for good classification, and these leftovers are just placed somewhere to fill the criteria the algorithm was given. A hypothesis that can then be formed is that it is likely that less than five filters might succeed well in the classification task, or even perform at a level similar to that of the five filters'. The answer to this hypothesis will be obtained when the results for the four and three filters are evaluated.

To validate the results seen here it is useful to compare them to those obtained for five filters with repeat amounts of 10 and 50 as well. The easiest way to do this is to plot all the locations of the placed filters, their means, and best succeeding sets in a single graph. This is what Figure 5.1 illustrates. The best succeeding sets are determined by comparing the direct success rates of all the sets and are marked with stars. The means are marked with crosses. The clouds of points are split into three layers around each filter number, responding to the number of repeats made in the ONN.

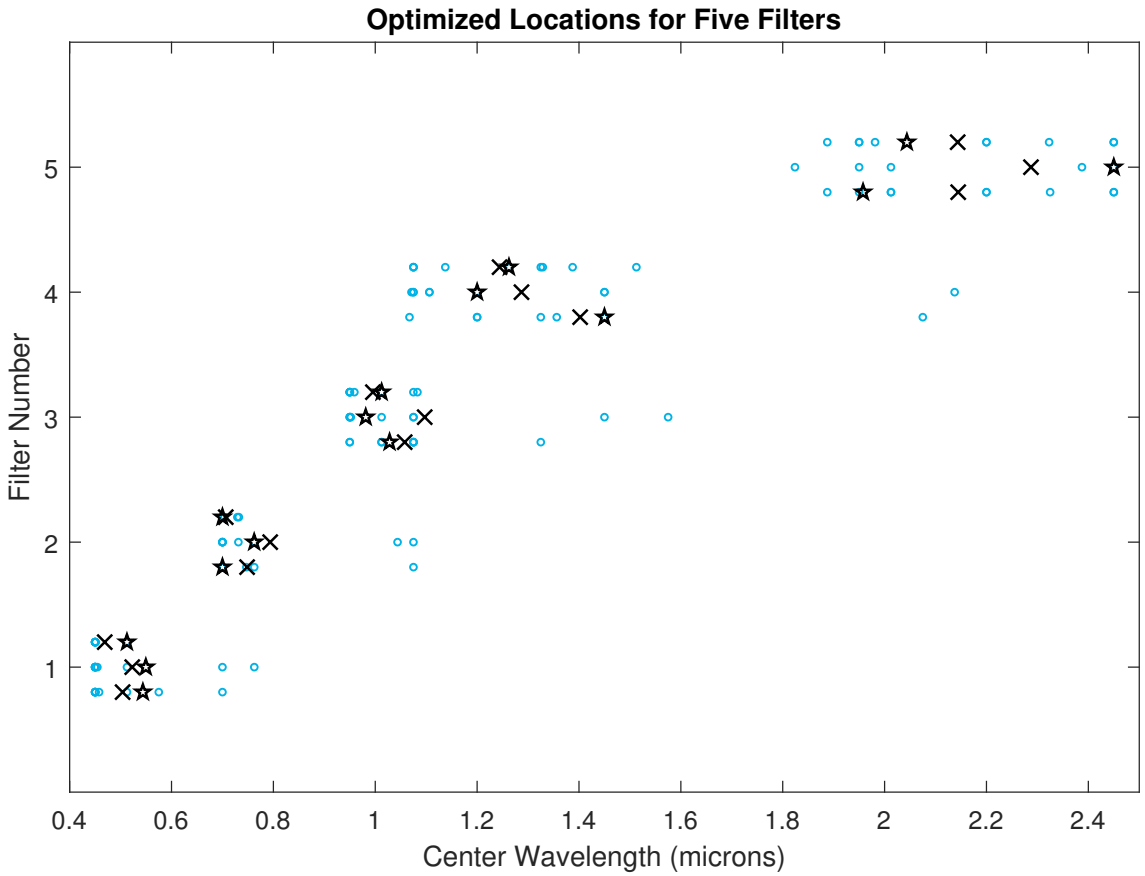

Figure 5.1. *Illustration of how all the recorded locations for five filters lay in the wavelength range. On the y-axis is the filter number. Note that each filter has three layers of values around it. These correspond to the different amounts of optimization network repeats, with 10 being the lowest layer, 30 being the one in the middle, and 50 being the highest. Means for each of the layers are marked with crosses, and the best succeeding locations with stars.*

Looking at how the filter locations lie in the figure, it can be seen that the sets obtained with 50 repeats have the least variance, whereas 10 and 30 almost always have a few runaways. Another clear trend is how the placement of the filters is much more tightly constrained in the lower wavelengths than at the end of the range, demonstrated further by the increasing standard deviations in Table 5.2. The possibility that this is caused by the filters being initialized at 0.45 microns was considered and tested by changing the initial values to 2.45 microns instead. The results are tabulated in Appendix C.2. It was found that the obtained locations are very similar to those that are results of starting at the beginning of the range, leading to discarding this being the root cause of the problem. Therefore, the wide range must mostly be explained by that indecisiveness suspected to be caused by having many filters in the set. However, overall the mean and the best

succeeding location of all the sets are typically quite close to each other. The lack of sets straying too far apart is relieving, as it can be taken as an indication of good optimization success.

The order of importance for these suggested locations can be evaluated with the DNN. In particular, this evaluation is done for the best succeeding set obtained in each complete simulation. Looking at Table 5.2, it is clear that based on both the ONN and DNN success rates, the best performing set is attempt 10 with filters placed at 0.550, 0.762, 0.981, 1.200, and 2.450 microns. Consequently, these values are also very close to the calculated mean locations. The process through which the importances can be determined is described next.

First, a baseline success rate is determined for the chosen locations. Then each filter is removed from the set one by one. This way each filter's importance can be evaluated by observing how much the direct success rate decreases when the filter's location is removed. If the success rate does not decrease radically, it can be concluded that the filter in question was not contributing significantly to the entire success rate of the set. If the success rate falls significantly, on the other hand, that filter must have been relatively crucial to the set. This process yields the results in Table 5.3. The order of importance, from least important to most important, is: 2.450, 1.200, 0.550, 0.762, and 0.981 microns. It is understandable that the 2.450- and 1.200-micron locations are the least important, especially when one recalls how widely spread the point clouds in those ranges are in Figure 5.1. These importances, however, will become more significant when they can be compared to those obtained for different filter amounts as well as when the underlying reasons for their order will be evaluated in Section 5.3.4.

5.3.2 Results for Four Filters

Since there seemed to be some uncertainty in the case of five filters as to where to place all of them, it is natural to consider whether using less filters would solve the problem. Reducing the amount of filters to consider to four results in values seen in Table 5.4, when 30 repeats are used in the ONN. The results for the

Table 5.3. *Determination of the order of importance for the best succeeding set in the five filters and 30 repeats simulation described in Table 5.2. Each filter location is represented by its central wavelength λ_n, where n is the filter number. The success rates are determined with the direct neural network. The resulting order of importance is listed from least important to most important.*

ONN Repeats: 30, DNN Repeats: 500						
Set	λ_1 (µm)	λ_2 (µm)	λ_3 (µm)	λ_4 (µm)	λ_5 (µm)	DNN Rate (%)
1	0.550	0.762	0.981	1.200	2.450	96.997
2	-	0.762	0.981	1.200	2.450	92.958
3	0.550	-	0.981	1.200	2.450	88.569
4	0.550	0.762	-	1.200	2.450	88.511
5	0.550	0.762	0.981	-	2.450	93.468
6	0.550	0.762	0.981	1.200	-	94.289
Resulting Order of Importance: 2.450 < 1.200 < 0.550 < 0.762 < 0.981						

simulations for four filters are again summarized in Figure 5.2, which plots all of them along with the means and best succeeding locations.

First of all, since the four filter case can choose less locations than its predecessor, it is interesting to assess which location it systematically leaves out. Based on the results in Table 5.4, it is clear that the location that *is* included in five filters but not here is 1.200 microns. Based on the importances of the locations for five filters, it is to be expected that values at the bottom of the list would be the target for being dropped out. However, it is somewhat surprising that 1.200 is dropped instead of 2.450, since 2.450 microns is lower in the importance list by almost a full percentage point. It might be that the four filters case deviates from five filters and needs the 2.450-micron location more, for example in order to establish a comparison point for the end of the range. This is supported by the fact that all the other filters are placed in shorter wavelengths, all below 1 micron. The results obtained by starting at the end of the range once again agree with the results described here, shown in Table C.8.

Observing the standard deviations of the locations in Table 5.4 and contrasting them with those of Table 5.2 shows that when using four filters, there is a drastic decrease in the variation of the placements. This observation is supported by how the suggested locations lie when plotted together, as seen in Figure 5.2. Although there are still some outliers, the clouds of points are much more constrained

Table 5.4. *Placements of four filters when the optimization neural network takes the mean after 30 repeats. For further specifications, see Table 5.2.*

ONN Repeats: 30, DNN Repeats: 500							
Set	λ_1 (µm)	λ_2 (µm)	λ_3 (µm)	λ_4 (µm)	λ_5 (µm)	ONN Rate (%)	DNN Rate (%)
1	0.450	0.763	0.903	2.419	-	94.420	93.893
2	0.458	0.763	0.919	2.450	-	94.520	94.084
3	0.575	0.762	0.888	2.450	-	94.564	94.211
4	0.575	0.763	0.888	2.419	-	94.623	94.091
5	0.575	0.763	0.872	2.450	-	94.665	94.085
6	0.544	0.731	0.903	2.450	-	94.830	94.271
7	0.544	0.763	0.903	2.419	-	94.852	94.500
8	0.544	0.732	0.888	2.432	-	94.852	94.317
9	0.544	0.763	0.890	2.450	-	94.958	94.482
10	0.546	0.755	0.911	2.450	-	95.144	94.383
Mean	0.535	0.756	0.896	2.439	-	94.743	94.232
SD	0.045	0.013	0.014	0.015	-	0.222	0.196

in specific regions when compared to the same figure for five filters. The most significant difference is how much more closely the points in the last filter are placed for four filters, indicating stronger certainty in where to place the entire host of given filters. However, this results in a significant gap in the wavelength range, running roughly from 1.2 to 2.2 microns. The underlying reason for this feature is unclear, but would be interesting to focus on in further studies.

Even though the standard deviation in the locations is reduced by using four filters, part of the overall performance is sacrificed; whereas the five filters' best succeeding set has a direct success rate of 97.064% and a direct mean success rate of 95.113%, using four filters results in 94.500% and 94.232% respectively. Although these values are slightly different, using four filters instead of five could be beneficial in real applications. Not only would the optimal placements of the filters be more certain, but data collected by them would still succeed well in classification. Additionally, using less filters would conserve resources and space.

To further evaluate the performance of the four filters, it is useful to inspect Figure 5.2 more closely. As mentioned before, it is easy to note that the clouds of points are much more constrained than before. However, another difference seen when comparing Figure 5.2 to Figure 5.1 is how well the mean and best succeeding

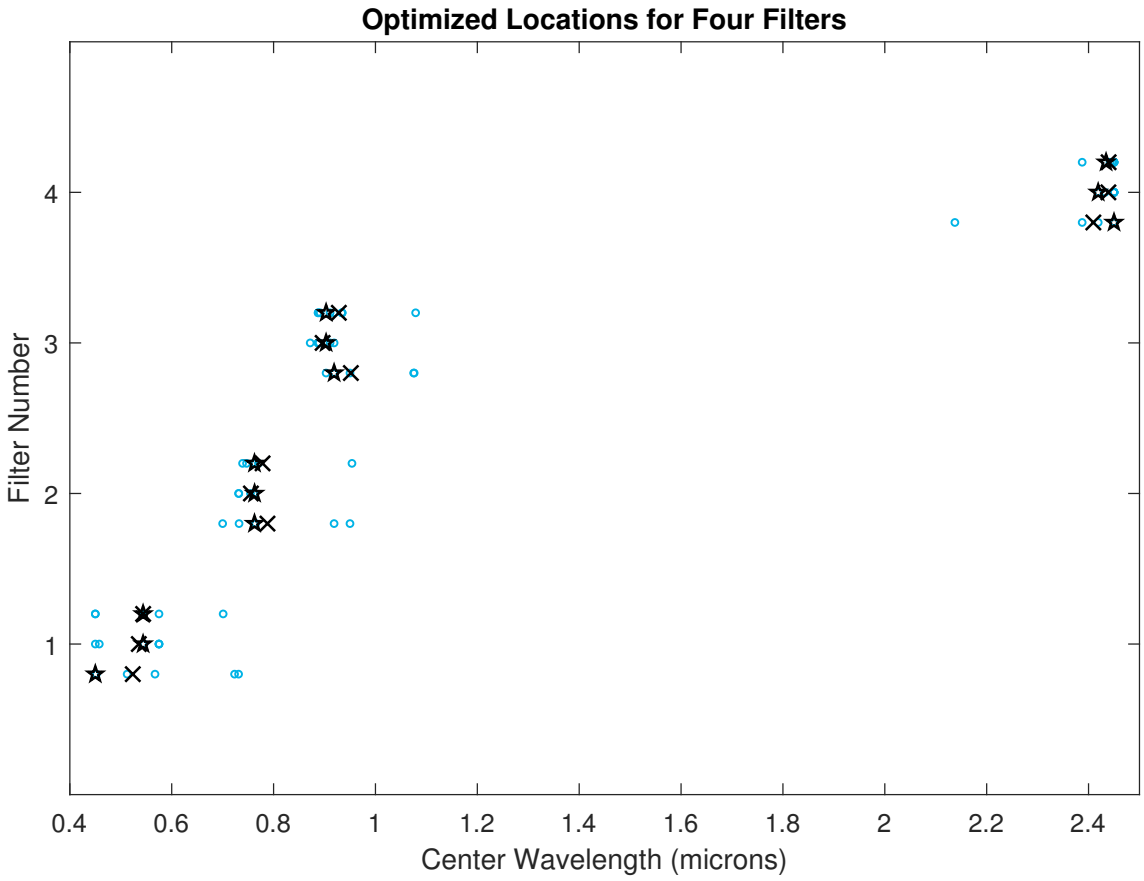

Figure 5.2. *Illustration of how all the recorded locations for four filters lay in the wavelength range. For further specifications, see Figure 5.1.*

locations align for all three different setups for the four filters. Even further, the means and best locations for the different setups align remarkably well with regards to each other. This is a strong indicator that the results given by four filters are rather reliable, particularly more so than those of the five filters'.

Finally for four filters, comparing how the importances of its best locations line up to those of the five filters' is important, as it puts the obtained results in clearer context. Choosing the best set for four filters is slightly more challenging, because due to the success rates being so similar, the ONN's success rate does not directly correlate to those of the DNN. However, the final decision is made based on the direct success rates, because they are less biased and have more runs to reduce the weight of outliers. Therefore, the best locations are in set seven with the direct success rate of 94.500%. The resulting order of importance for these locations is determined in Table 5.5, and shows that it is, in increasing order of importance: 2.419, 0.544, 0.763, and 0.903 microns.

Table 5.5. *Determination of the order of importance for the best succeeding set in the four filters and 30 repeats simulation described in Table 5.4. For further specifications, see Table 5.3.*

ONN Repeats: 30, DNN Repeats: 500						
Set	λ_1 (µm)	λ_2 (µm)	λ_3 (µm)	λ_4 (µm)	λ_5 (µm)	DNN Rate (%)
1	0.544	0.763	0.903	2.419	-	94.437
2	-	0.763	0.903	2.419	-	90.046
3	0.544	-	0.903	2.419	-	84.929
4	0.544	0.763	-	2.419	-	74.564
5	0.544	0.763	0.903	-	-	90.086
Resulting Order of Importance: 2.419 < 0.544 < 0.763 < 0.903						

This result is intriguing, since the order aligns perfectly with that of the five filters (naturally with 1.20 microns removed). The value located around 2.45 microns is still considered to be the least important, while values around 1 micron are the most crucial for classification success. It is particularly interesting from a geophysical viewpoint to see such a pattern forming. Additionally, it helps validate the performance of the optimization algorithm, since it is ultimately returning similar results, even in different implementations.

5.3.3 Results for Three Filters

The final amount of filters that shall be tested in this study is three. Although four filters already reached quite good overall performance, investigating what level of results can be obtained with less filters will elaborate what the optimal amount of filters truly is based on these simulated measurements. If the success rates were to suddenly improve with three filters, further studies into the nature of the filters would have to be done. On the other hand, if the success rates were to decrease, this would be an indication that four filters seems to be the optimal amount.

The results obtained for three filters are listed in Table 5.6. Determining what wavelength is left out from the previous four locations is not as simple as before. The whole set of three filters seems to have shifted to be closer to the middle of the range, behaviour which is easiest to see in Figure 5.3. Consequently, there is

no perfect correspondence in the locations when contrasted with those obtained with four and five filters. The success rates experience a clear drop, with the best direct success rate 90.153% and mean success rate 89.783%, as opposed to four filters' 94.500% and 94.232%. The results with the changed initial points agree with these values, as seen in Table C.9. This decrease of five percentage points in the mean rates might not at first seem significant. However, one must recall that the success rate dropped by just one percentage point when changing from five filters to four, making this change much more drastic in comparison.

Table 5.6. *Placements of three filters when the optimization neural network takes the mean after 30 repeats. For further specifications, see Table 5.2.*

ONN Repeats: 30, DNN Repeats: 500							
Set	λ_1 (µm)	λ_2 (µm)	λ_3 (µm)	λ_4 (µm)	λ_5 (µm)	ONN Rate (%)	DNN Rate (%)
1	0.763	0.950	1.419	-	-	89.541	89.095
2	0.747	0.950	1.403	-	-	89.592	89.094
3	0.763	0.950	1.388	-	-	89.623	89.184
4	0.778	0.934	1.559	-	-	90.082	89.566
5	0.793	0.981	1.259	-	-	90.403	89.850
6	0.778	0.888	2.411	-	-	90.477	90.153
7	0.778	0.919	2.417	-	-	90.592	90.246
8	0.763	0.919	2.434	-	-	90.658	90.231
9	0.825	1.013	1.200	-	-	90.686	90.263
10	0.825	1.013	1.200	-	-	90.726	90.153
Mean	0.781	0.952	1.669	-	-	90.238	89.783
SD	0.026	0.041	0.530	-	-	0.487	0.568

With three filters, the first two locations are still very tightly constrained, as illustrated by their standard deviations and placements in Figure 5.3. However, there is strong variance in where the algorithm places the last filter: the placements extend from approximately 1.2 to 2.4 microns, constituting over half of the total range. None of the measurements executed before showed this degree of uncertainty. The "hesitancy" is likely explained by the fact that there are no remarkable differences in the three filters' success rates, which signifies that the chosen last locations are almost equal in performance. It could then be expected that the best succeeding locations would be in several places within that range for the tests with different repetition amounts. However, this expectation is diverted as all three tests consistently succeed the best when the last filter is placed at approxi-

mately 1.2 microns. This is surprising, as the last time 1.2 microns was included in the results was for five filters, after which it was dropped by the four filters.

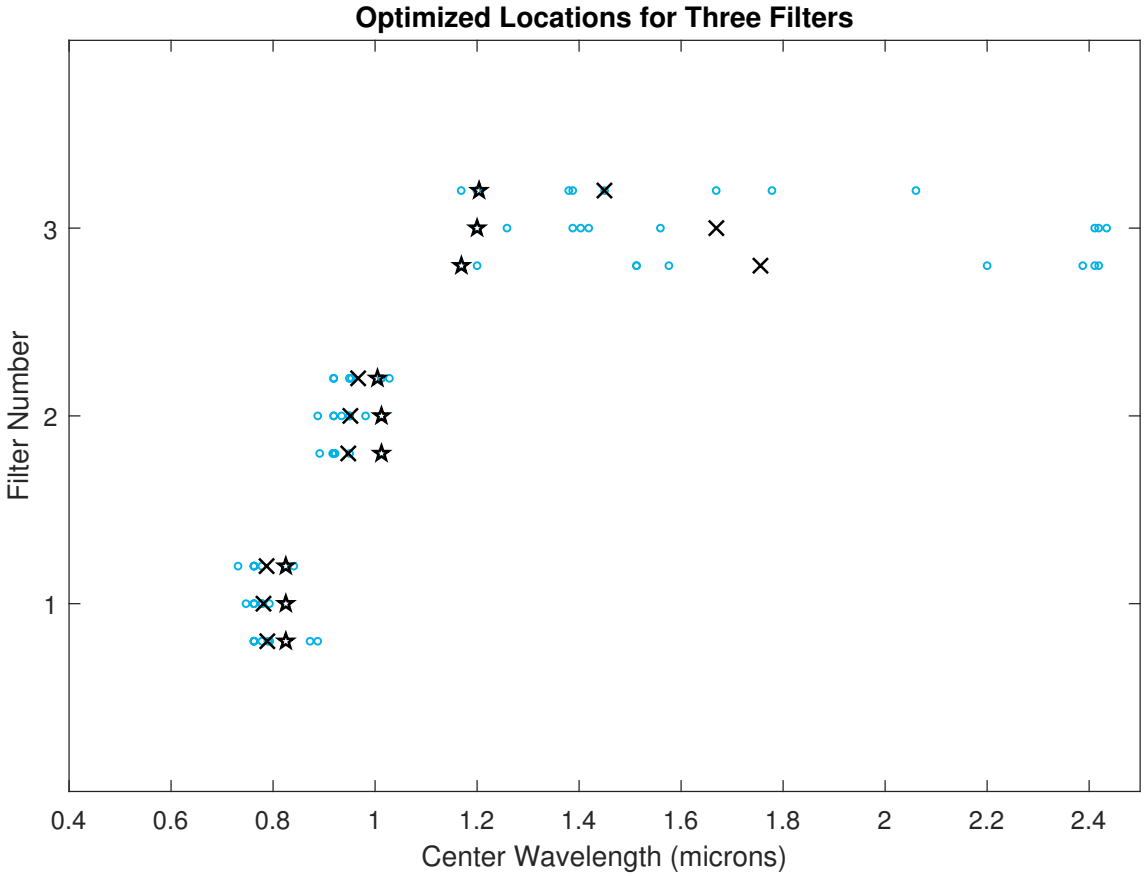

Figure 5.3. *Illustration of how all the recorded locations for three filters lay in the wavelength range. For further specifications, see Figure 5.1.*

The best succeeding set of locations is actually repeated twice for three filters. These are set 9 and set 10 in Table 5.6, with placements at 0.825, 1.013, and 1.200 microns. The evaluation of their order of importance is included in Table 5.7. Most importantly, the obtained results continue the trend of choosing the value near 1 micron as the most significant one. This is relieving to see, as it places the three filters' results at least partly in the same context as those obtained with larger filter amounts. On the other hand, it is rather surprising that the results state that the 0.825-micron wavelength is actually less important to the success than the largely varying last location at 1.200 microns. Based on the collective results in this thesis, it is left somewhat unclear what causes the peculiar phenomenon around the 1.200-micron location, and more extensive studies would have to be done to determine its exact nature.

Table 5.7. *Determination of the order of importance for the best succeeding set in the three filters and 30 repeats simulation described in Table 5.6. For further specifications, see Table 5.3.*

ONN Repeats: 30, DNN Repeats: 500						
Set	λ_1 (µm)	λ_2 (µm)	λ_3 (µm)	λ_4 (µm)	λ_5 (µm)	DNN Rate (%)
1	0.825	1.013	1.200	-	-	90.185
2	-	1.013	1.200	-	-	74.957
3	0.825	-	1.200	-	-	64.983
4	0.825	1.013	-	-	-	70.953
Resulting Order of Importance: 0.825 < 1.200 < 1.013						

5.3.4 Evaluation of the Optimized Locations

The results obtained in the optimization process are encouraging, since clear trends developed in the results for each number of filters. While there are no radical differences in the performance between different amounts of repeats in the ONN, for future studies high values are recommended if possible, since using 50 repeats consistently demonstrates the least variance in the clouds of locations. The choice of focusing on 30 repeats here is still valid, however, as the high values can be inapplicable to some studies due to their time-consuming nature.

Based on the results summarized in the previous sections, the wavelengths that the filter locations seem to prioritize are, approximately: 2.45, 1.20, 0.55, 0.76, and 1.00 microns, in order of increasing importance. These locations are illustrated in Figure 5.4, where the wavelength range is the region between the dashed lines. The filters are modelled as Gaussian distributions around the suggested locations, with the FWHM of 0.05, as explained before. This section analyses what spectral features these locations might correspond to in order to find the reason they were chosen. The discussion shall move from the least important wavelengths to the most important.

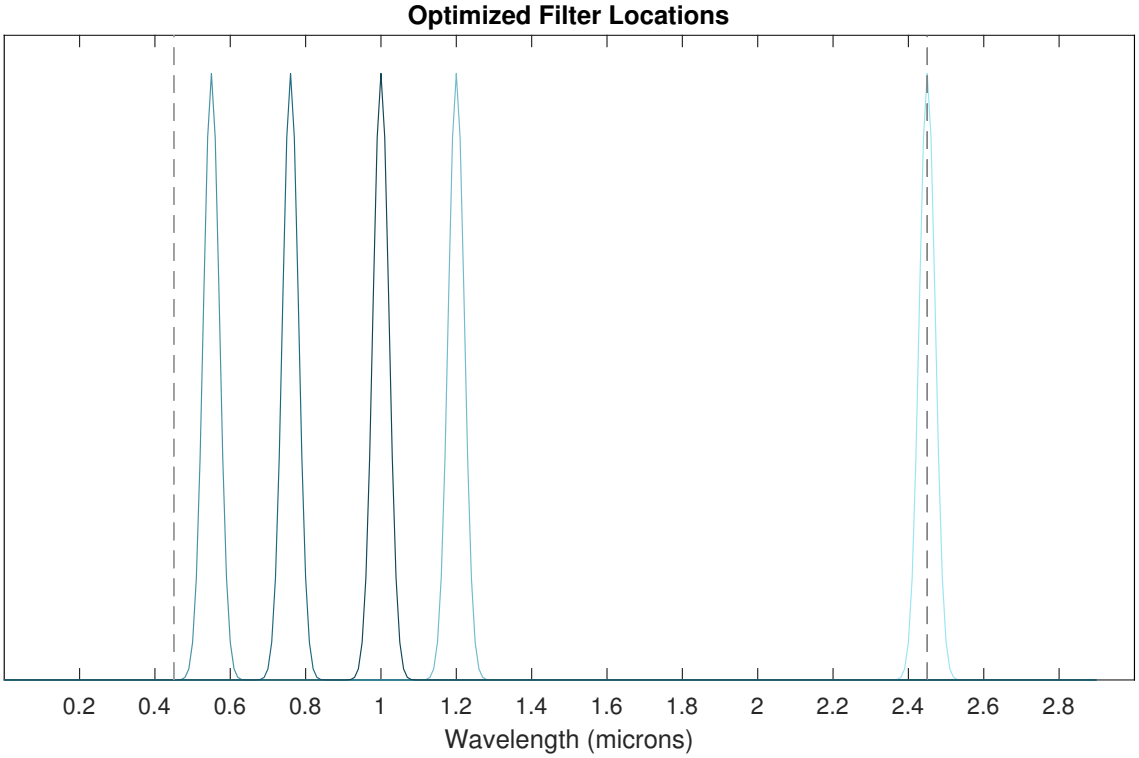

Figure 5.4. *Summary figure of five optimized filter locations. The region within the dashed lines is the wavelength range of the dataset utilized in this study.*

Remembering the visualization of the spectra and Table 2.2, no particular feature exists at 2.45 microns. It is likely that the algorithm chose it in order to simply obtain a better estimation of the overall slope. Similarly, there are no significant features around the enigmatic 1.20 microns. It *is* close to the middle of the range, and many of the classes do not show much deviation in their spectra after that point. However, as was mentioned in the last section, the precise reason why it lingers in the optimization results is slightly unclear.

A favoured location that has not yet been fully discussed is 0.55 microns. It being chosen by both four and five filters is surprising, since all the spectra are normalized at this point. It is also the smallest value that the most successful sets chose. Therefore, it is likely that its purpose is to define the beginning of the range, much like 2.45 microns is used to define the end. Additionally, the fact that the spectra have the same value at that location also likely forms a good comparison point.

All of the tested filter amounts consistently chose to have values close to 0.76 microns in the optimized location sets. The list of features in the DeMeo taxonomy does mention features close to this value, particularly for the C- and L-classes.

Based on visual inspection of the spectral shapes, A, K, Q, S, and V also generally display moderate peaks around the location. However, the wavelength that appears most often in the list of features is 1.00 microns. This is reasonable, as many of the spectra show clear absorption features centered around this point. Consequently, the tests chose the location at 1.00 microns to be most crucial for the classification process. It should be noted that 0.76 and 1.00 microns are the only locations that were included in all the three tests, and both were proven, in general, to be the most important for classification success.

Since both 0.76 and 1.00 microns show strong correlation to actual spectral features, they should have a mineralogical explanation. The 0.7-micron feature listed in Table 2.2 signifies the presence of phyllosilicates that likely formed on the asteroid surface due to aqueous alteration processes [40]. This feature is most closely linked to C-type asteroids, although B-, X-, and T-types have been proven to exhibit it as well, however, not as commonly [41]. The feature at 1.00 microns, on the other hand, is correlated with normal silicates, which are particularly abundant in the S-class [42]. Therefore, both of the most distinct features the optimization algorithm relied upon are silicate-based. Sadly, detection of absorption features caused directly by, for example, water is not possible at these wavelengths, as they lie deeper in the infrared [43].

One more feature to note is the lack of any filters between approximately 1.30 and 2.30 microns. This was already noted during the presentation of the results of the different filters, but is seen with even more clarity in Figure 5.4. It is particularly curious that the gap extends over most of the infrared range, since it is considered to be very important for most modern asteroid taxonomies. It seems to signify, for example, that the neural network does not consider the features that many classes, such as the S-class and V-class, exhibit around 2 microns as being extremely important for classification success. However, the specifics of the underlying reasons for the gap in the infrared still remain unclear.

In order to investigate whether the normalization point of the spectra has an effect on the optimized filter locations, the point was changed from 0.55 to 1.60 microns. The 1.60-micron location was chosen, as there seem to be no significant features

there and the optimizer has not chosen it as a location in previous tests. The five locations suggested by the most successful set are, approximately: 0.61, 0.79, 1.83, 2.20 and 2.45 microns. The fact that locations near the beginning and end of the range are still chosen with the new normalization point strengthens the implication that the optimizer is choosing them in order to estimate the slope of the spectra. The location at 0.79 microns is also very similar to the favoured 0.76-micron location seen before.

However, when the normalization point is changed, there is slightly more variance in the chosen locations and the success rates are on average 1 percentage point lower than those obtained with normalization at 0.55 microns. This seems to imply that the normalization point has an effect on the optimal filter locations, and must therefore be considered in applications similar to the one presented here. In the future, it could be beneficial to consider methods that either let a machine learning algorithm choose the most optimal normalization point or remove the need for one specific normalization point altogether.

6 TASK II: UNSUPERVISED ASTEROID CLASSIFICATION

Even though humans have evolved to recognize patterns in the data our senses provide, when the numbers grow large, it becomes harder to distinguish what is truly meaningful and what is noise. Machines, on the other hand, are unbiased and can be taught to recognize patterns based on vast amounts of provided data. They thrive when the datasets become large. Section 2.2.4 underlined how reliant, so far, asteroid classification has been on human decisions. What would an asteroid taxonomy formed by a machine look like then, and would it outperform those made by humans? This question forms the basis for the second explored task in this thesis, Task II. The task in itself is particularly topical as the release of the full data collected by Gaia draws near, as was discussed in Section 2.2.2.

6.1 K-Means

The list of different unsupervised machine learning methods is long and keeps growing. A classic method that still sees much use today is k-means. Its operation principle is based on the concept of finding k centers that the objects are clustered around. Typically the algorithm begins by choosing k objects from the dataset randomly and assigns them as the initial centroids. Objects are then placed to the closest centroids until none are left. The center is then recomputed based on the formed clusters, and the process begins again. The procedure is considered to be finished when no object is reclassified in two consecutive steps [44].

K-means is widely applied in asteroid analysis [45, 46, 47], although none of the previous studies seem to have explored a large-scale spectroscopic case that extends into the infrared. Not only is it historically favoured for astronomical applications, but k-means is particularly well-suited to solving Task II, as it is computationally simple and robust even when handling high-dimensional data [48], which the VISNIR set is a good example of with its 200 features for each sample.

K-means is extremely sensitive to noise, as ideally the potential clusters should be compact and isolated in order to be clearly distinguished from each other [49]. If the dataset has clear outliers, the algorithm will likely separate them into their own individual clusters. As a consequence, before clustering is attempted on the reduced VISNIR set of 582 samples, visually identifiable outliers are removed. This decreases the total size of the dataset to 578 samples. Another factor to note is that much like a neural network, k-means by default begins from a different set of initial values each time in order to prevent developing a bias and ensure that several different separation methods are attempted. Therefore, running it only a handful of times might provide results that describe the sum of distances within the clusters as longer than they could optimally be. In order to avoid this problem, several repeats are done in each of the following sections.

There is another important parameter the user must determine before k-means can be run: the distance measure it is going to utilize to calculate the distances between the samples and cluster centroids. The most appropriate measure depends on the dataset. Euclidean distance is perhaps the most widely used, and its squared counterpart can be described by the equation

$$d_{\mathrm{sqeuc}} = \sum_{j=1}^{m} (a_j - b_j)^2, \tag{6.1}$$

where a and b are two points with m dimensions in Euclidean space [50]. While it is generally resistant to outliers [50], its performance suffers when the dimensionality of the data increases [51]. Squared Euclidean distance shall be chosen as the first distance measure to investigate owing to its overall reliability, but another one must be chosen to account for the high dimensionality of the data.

A measure that has been experimentally shown to work well with data with high dimensionality is cosine distance [51]. It is equal to the cosine similarity subtracted from 1 [50]

$$d_{\cos} = 1 - \cos\theta = 1 - \frac{\mathbf{x} \cdot \mathbf{y}}{\|\mathbf{x}\|\|\mathbf{y}\|}, \tag{6.2}$$

where $\mathbf{x}$ and $\mathbf{y}$ are vectors, $\|\mathbf{x}\|$ is the Euclidean norm of vector $\mathbf{x}$, and $\|\mathbf{y}\|$ the Euclidean norm of vector $\mathbf{y}$. Conceptually, the Euclidean norm is the length of the vector [52]. Consequently, the cosine distance is the second measure that the clustering shall be investigated with. Neither of the chosen distance measures are metrics, as they do not satisfy the triangle inequality [53].

6.2 Determining Number of Clusters

K-means requires the user to choose how many clusters must be formed. Although it is possible to manually inspect which number of clusters seems to be the most ideal based on the obtained results, the process is time-consuming and introduces a somewhat unsatisfactory amount of human intervention into a task that was developed to reduce its amount. Therefore, methods for attempting to determine the ideal number of clusters mathematically will be explored in the following sections.

6.2.1 Silhouettes

One of the simplest ways to evaluate the ideal number of clusters to use for a specific dataset is through utilizing silhouettes. Their ease of use arises from the fact that the calculation and plotting of their shape is typically automated in most packages. Mathematically, however, the method is based on the following set of rules, which are primarily from Peter J. Rousseeuw's 1987 paper on silhouettes [54]. First, let us take an asteroid spectrum i that belongs to cluster A. After choosing i, the average distance from it to all the other objects in the same cluster

must be calculated. This value is marked as $a(i)$. Then the average distance from that spectrum to objects in all the other clusters than A is calculated. From these, the minimum average distance is chosen and marked as $b(i)$. This signifies the second-best choice to cluster the sample into, other than A. Hence, the silhouette value for the ith object is

$$s(i) = \frac{b(i) - a(i)}{\max[a(i), b(i)]}. \tag{6.3}$$

When cluster A only has one object in it, $s(i)$ is defined as being equal to zero. From the formula it also follows that the silhouette value ranges from -1 to 1, with values closer to 1 indicating that the object i is likely placed in a cluster that represents it well. Calculation of these values for the points can then be done for different clusters amounts, after which one can determine what amount yields the results closest to 1.

Calculation of the silhouette values for different amounts of clusters was executed in this study with Matlab. The range of clusters to evaluate was limited to run from 5 to 25. The lower limit is imposed as 5, since it is already possible to tell with visual inspection of the spectra that any cluster number below it would be an unfair representation of the divisions within the data. The upper limit is defined as 25, as that is the current number of classes in the DeMeo taxonomy [10]. Because the initial values for k-means vary, the silhouettes were calculated 100 times for these 21 cluster amounts, out of which means were taken. The mean silhouette values for both of the distance measures are plotted in Figure 6.1.

The produced silhouette graphs appear visually similar. Both have a peak close to the beginning of the range and then decrease in a manner that resembles exponential decay. With closer inspection it is possible, however, to note that the silhouette values are on average higher for squared Euclidean distance. Both graphs have a maximum at seven clusters, making it the optimal number of clusters to provide k-means with based on the silhouette method.

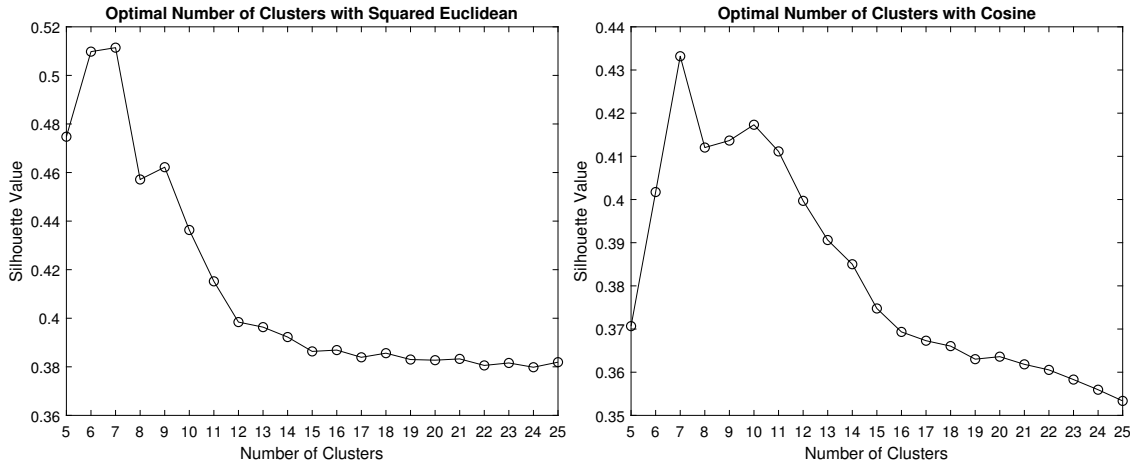

Figure 6.1. *Mean silhouette values after 100 repeats for squared Euclidean and cosine distance. While the silhouette values for squared Euclidean are slightly better on average, both of the graphs peak at seven clusters.*

The graphs produced by the silhouette method are in rather sufficient agreement for both distance measures, as they both indicate that seven is the optimal number of clusters for this dataset. However, before the k-means algorithm is run, it would be beneficial to verify whether seven is truly the number of clusters that represents the divisions within the data with both measures best. The verification is executed in the next section with another popular method for choosing k.

6.2.2 Elbow Method

The elbow method measures the percentage of variance as a function of the number of clusters [55]. The variance is often explained as the quantity of within-cluster distances: the lower the sum of the distances of points in a cluster to the mean point that defines the cluster, the lower the variance in locations. It is intuitively clear that the more clusters there are, the smaller the variance will be. It would be easy to then determine that the more clusters there are, the better the results should be. This is true to an extent, but using too many clusters purely in the hope of reducing the within-cluster distances can lead to poor results. The underlying reason is the fact that the clusters can be split somewhat arbitrarily if the proposed value for k is too large, as they no longer follow "natural" divisions within the data.

A way to determine the good "trade-off" point in the cluster amounts based on the within-cluster distances is to plot them against each other, and determine where the "elbow" in the graph is. The plotting is done with the squared Euclidean and cosine distances in Figure 6.2, where the bars illustrate the obtained sum of within-cluster distances for each number of clusters after 500 repeats are made for each cluster, out of which the shortest sum of distances is chosen. However, as one can see from the shape of the bar graphs, it is not particularly clear where the elbow would lie. Visual inspection and guesswork would indicate the location of the elbow to be anything between 7 to 15 for squared Euclidean and 11 to 16 for cosine, and even those guesses could be optimistic.

Clearly, a more refined method for determining the elbow is needed. One way to accomplish this is to calculate the curvature of the within-class distances, which describes how the sums of these distances evolve [56]. The curvature is the second derivative of the curve formed by the bar graphs. Spikes in the curvature respond to drastic changes in the homogeneity of the associated clusters, and can be therefore correlated to optimal numbers of clusters to provide the k-means algorithm with [56]. Estimating the curvature is done through using the second order central difference approximation. The obtained curvatures, both multiplied by a factor of 25 in order to make them show more clearly, are plotted on top of their respective bar graphs in Figure 6.2.

Before evaluating the obtained spikes, it is important to keep in mind that the within-cluster distances are by definition different for the two distance measures. Therefore, the heights of the peaks in the two graphs should not be directly compared to each other. Looking at the graph for squared Euclidean shows a very distinct spike at 7 clusters, along with several smaller ones at 10, 13, 15, 17, 19, and 23. Out of these, seven clusters is definitely the one to first investigate for squared Euclidean, particularly as seven clusters was also the optimal number suggested by the silhouette method.

For cosine distance, the most distinct peak is at 11 clusters, followed by peaks at 14, 17, 19, and 22 clusters. It is unclear whether the sharp negative slope starting from six clusters indicates that there is a peak before it. However, for

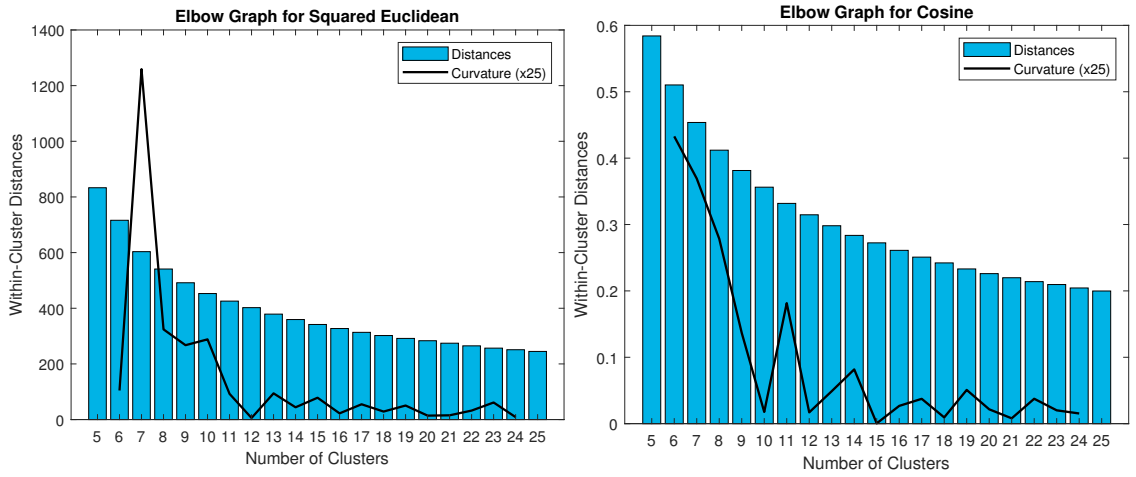

Figure 6.2. *Elbow graphs for squared Euclidean and cosine distance. The bars are simple measures of the total within-cluster distance for the given cluster amounts. The curvature has been multiplied by a factor of 25 in order to make it more distinct, and is formed by applying the second-order central difference approximation to the distance values. For squared Euclidean, a clear peak is seen at seven clusters. For cosine, the most distinct peak appears at eleven clusters.*

the sake of fair comparison, and because the silhouette method for cosine still showed the highest success for seven clusters, this will be the first investigated case for cosine as well, followed by eleven for any possible further studies.

6.3 Clustering Results

Once the number of clusters to provide the described k-means algorithm have been determined, it can finally be run in order to obtain the new classes. For obtaining the following results, the algorithm was once again run 500 times, out of which the result with the best (shortest) sum of distances was chosen.

First, the results for seven classes with the squared Euclidean distance are illustrated in Figure 6.3 and tabulated in Table 6.1. The class numbers themselves do not have any particular meaning, because each iteration of running the k-means would order them differently, even if the objects in the clusters stay the same. Therefore, focus should be kept on the *kind of* objects each new class holds after the clustering. Looking at Figure 6.3, at first glance it seems that based on the shape of the spectra in the classes, the results are rather adequate; for example, class 7 appears consistent, seemingly formed from only V-samples. However,

closer scrutiny reveals several "outliers" in some of the other classes that seem out of place. This is particularly noticeable in classes 3 and 5. Class 4 seems to have combined the high slope spectra from D-class with some A-class samples, however, not all A's and D's are in this new class.

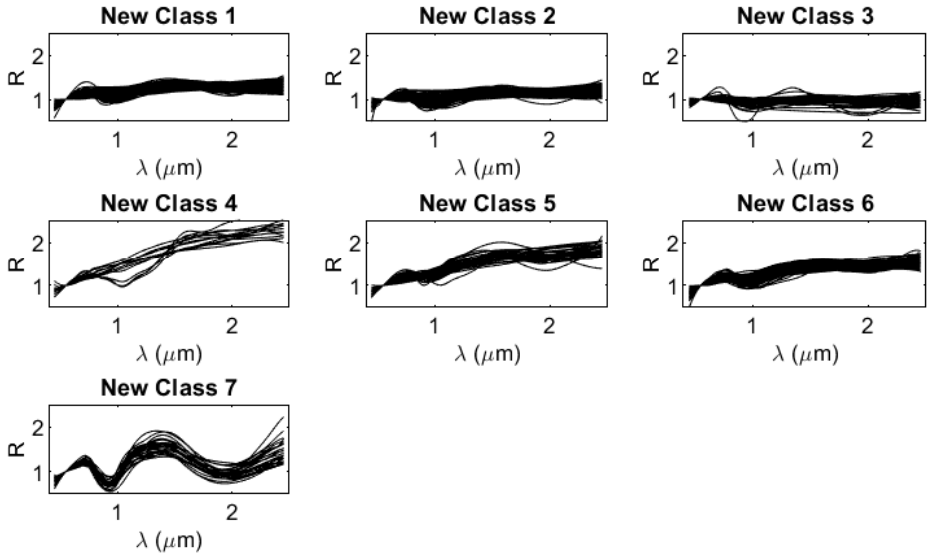

Figure 6.3. *Plotting of original spectral shapes in their new assigned classes, determined by forming seven new clusters with the squared Euclidean distance measure. The x-axis holds the wavelengths from 0.45 to 2.45 microns, while the y-axis is the normalized reflectance.*

A closer look at Table 6.1 illustrates the structure of the formed classes better. The cell for each original class with the highest number of placed samples in the new system is highlighted with blue. The intensity of the colour corresponds to how large of a share that cell holds out of the total number of samples in the original class. In other words, the deeper the colour, the more likely it is that the new cluster is a good representation of the old class as a whole. Most of the coloured cells are on the lighter side, implying that there is some variance in where samples get placed in the new system. This is what was partly seen in Figure 6.3 already. The classes are in general somewhat meaningful, examples of which are the fact that cluster 2 consistently holds a large share of the subclasses of C and cluster 6 collects several samples with particularly high slopes without integrating too many from A or D. However, the fact that such a large share of the original classes become so fractured takes away from the meaning of the results,

as the new classes cannot be fully trusted to be good representations of divisions in the data. In order to investigate whether this is caused by the data itself or the clustering method, the results for seven new classes using the cosine distance must be evaluated.

Table 6.1. *Occurrence rate for full DeMeo classes in the seven new clusters with squared Euclidean distance. In the last column are the total numbers of samples in each original class.*

Class	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5	Cluster 6	Cluster 7	Total
A				4	1			5
B			12					12
C	2	13	10					25
Cb		3	1					4
Cg		2						2
Cgh	1	7	2					10
Ch		14	5					19
D				8	10	2		20
K	5	9	1					15
L	15	6			6	6		33
Q	1	32	9					42
Qw						1		1
S	96	22				28		146
Sa						3		3
Sq	12	24	3			3		42
Sqw	2					15		17
Sr	17	9	1			4		31
Srw					1	7		8
Sv	2					1		3
Svw					1	1		2
Sw					16	41		57
T						4		4
V		1	2				22	25
Vw					1		1	2
X	5	2				8		15
Xc	1	2						3
Xe	5	3				2		10
Xk	7	9				5		21
Xn			1					1

Using the same methodology with only the alteration of changing the distance measure to cosine results in a distribution of spectral shapes illustrated in Figure 6.4. It is possible to see that the silhouette of the shapes of spectra in each cluster appear much more consistent compared to Figure 6.3, even accounting for the fact that class 2 and 6 seem to have some outliers. Similar to squared Euclidean, cosine combines A's and D's into one cluster, although in this case it seems as if a larger portion of both have been recruited. Visually it is also possible to immediately see that V-samples have been recruited to form their own cluster, which was the case in squared Euclidean as well. However, in order to make more precise comparisons between the distance measures and ultimately decide which seems more successful based on the results, the table of occurrences for samples in the new clusters must be consulted.

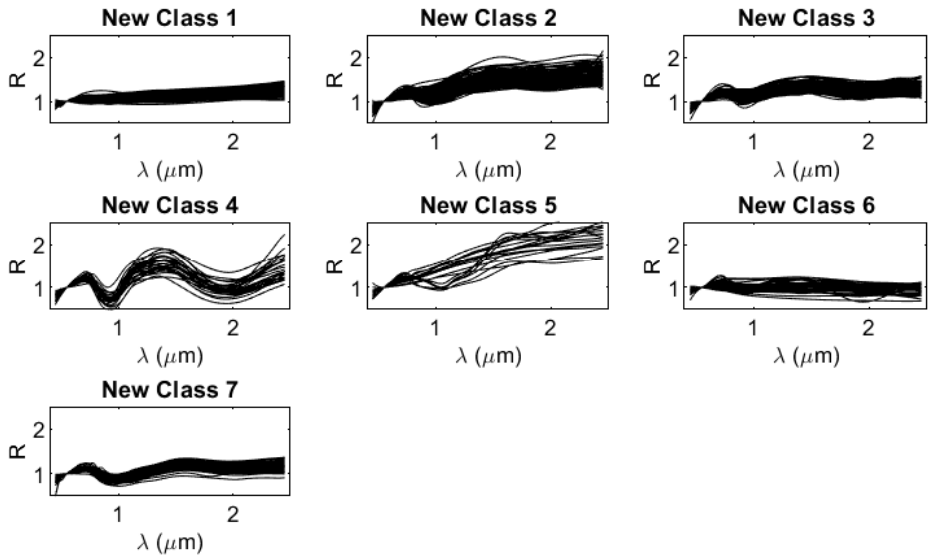

Figure 6.4. *Plotting of original spectral shapes in their new assigned classes, determined by forming seven new clusters with the cosine distance measure. For axes information, see Figure 6.3.*

Table 6.2 was constructed with the same methodology as Table 6.1. When the two are directly compared, the first notable difference is that Table 6.2 has a significantly larger number of darker-toned cells, implying that there is more certainty in where to place old classes in the new system. Like with squared Euclidean, V and Vw stay undisturbed within their own cluster. Both methods choose to group

C, its subclasses, and several samples of X's subclasses in the same cluster, here in cluster 1 and in Table 6.1 in cluster 2. The two distance measures also make a distinction between samples with high slope, denoted by "w" after the original class, although both include several samples that have not been denoted as having particularly high slope in the original classification. However, cosine seems to be slightly sharper in where to draw the line for the distinction between what should be considered as a high slope.

Table 6.2. *Occurrence rate for full DeMeo classes in the seven new clusters with cosine distance. In the last column are the total numbers of samples in each original class.*

Class	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5	Cluster 6	Cluster 7	Total
A					5			5
B						12		12
C	20					5		25
Cb	3					1		4
Cg	2							2
Cgh	10							10
Ch	15					4		19
D		9			11			20
K	9	2				3	1	15
L	5	9	16			3		33
Q							42	42
Qw		1						1
S	1	19	118			3	5	146
Sa		1			2			3
Sq	2	9	6			3	22	42
Sqw		17						17
Sr		1	26			3	1	31
Srw		8						8
Sv			3					3
Svw		1	1					2
Sw		57						57
T		4						4
V				24		1		25
Vw				2				2
X	5	10						15
Xc	3							3
Xe	8	1	1					10
Xk	12	7				2		21
Xn						1		1

A curious difference between the two sets of clusters is how cosine shows a tendency to group the classes Q and Sq together, seen in cluster 7. Both naturally share a similar set of features, but it is interesting to see that Sq is more likely to become clustered together with Q than other S-class members. However, as was the case with squared Euclidean distance, the meaning of some clusters is still unclear when contrasted with the original taxonomy. This behaviour is particularly evident in clusters 3 and 6 in Table 6.2. An attempt to solve the issue would be to increase the number of clusters. Since cosine distance returned a taxonomy that was in general more meaningful than the one produced by squared Euclidean distance, only cosine will be explored with an increased number of clusters.

Returning to the elbow graphs for the distance measures, the most distinct spike for cosine was centered at eleven clusters. This number of clusters also saw relatively good success with the silhouette method, and shall therefore be chosen for further unsupervised classification studies. The eleven new classes are illustrated in Figure 6.5 and their occurrence rates are tabulated in Table 6.3. With the larger number of clusters, the spectral shapes included in the new classes change. Notably, it is possible to see that the previous new class 5 in the cosine system with 7 clusters has now been split into two new classes: 2 and 4. This corresponds to separating the previously combined A and D samples from each other. Another easy visual difference to note is how there now are two clusters of V-samples: classes 8 and 9. In general the shapes of the new classes appear very "clean": there is rather good uniformity in the overall shapes with few outliers, particularly when compared to the cases with seven clusters.

By evaluating the results in Table 6.3, it is possible to see that the original classes A and D are indeed separated into their own distinct classes in the system with eleven clusters. Both are also placed into their clusters with high certainty, and only have a few outliers from other classes in their respective clusters. The basis for this behaviour likely arises from their distinct shapes and high slopes. Some other well separated clusters are 1, 5, 8, and 9. Cluster 1 is primarily composed of Q-samples as well as Sq-samples, which, as we previously saw, have a tendency to be closer to Q than S, particularly with cosine distance. Cluster 5, on the other hand, holds the B-samples, with only two outliers from the C-class. This is a clear

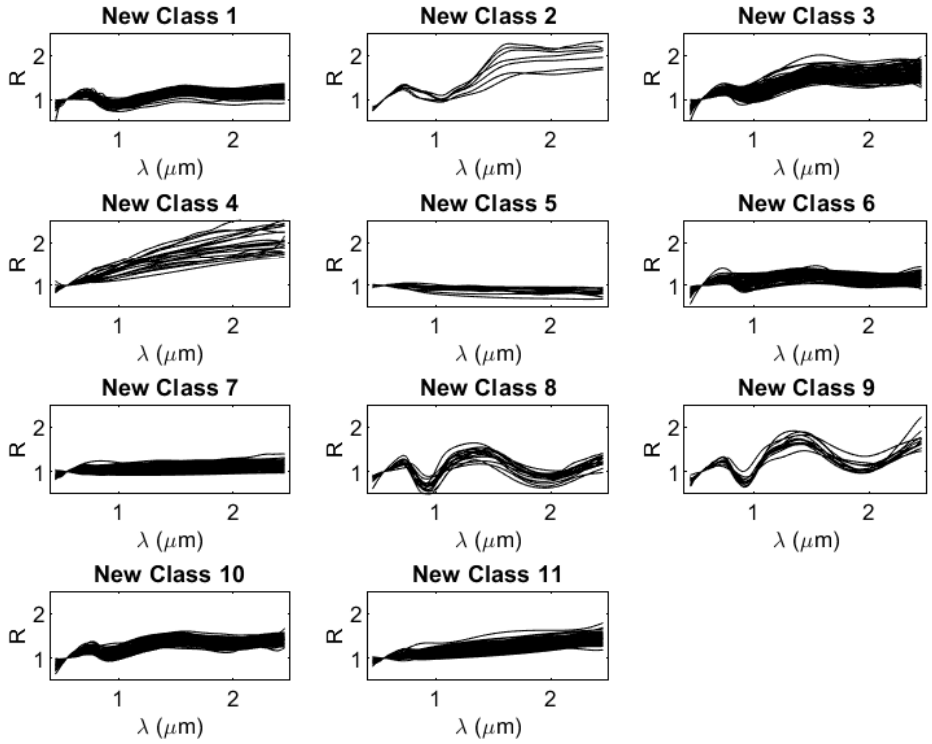

Figure 6.5. *Plotting of original spectral shapes in their new assigned classes, determined by forming eleven new clusters with the cosine distance measure. For axes information, see Figure 6.3.*

distinction from the previous system, where B was integrated into a cluster which had samples from several other classes. Clusters 8 and 9 hold the V-samples. It is curious that the algorithm chose to separate them into their own clusters, particularly because there are no outliers from other original classes in them. A possible explanation for this, however, is how much within-class variance they have in the original dataset. The variance in V-samples is possible to visualize with PCA as well, where they habit a large portion of the formed map [6].

The cluster 1 from the system constructed with seven cosine clusters remains relatively unaltered in cluster 7 in this system. It still holds the majority of the C-samples, along with samples from K- and X-classes. The significance of the remaining clusters 3, 6, 10, and 11 is in comparison somewhat unclear. They contain samples from so many different classes that is hard to tell whether these are truly good separations in the data. The only one with a somewhat clear trend is cluster 3, which combined primarily samples that were listed as having high

Table 6.3. *Occurrence rate for full DeMeo classes in the eleven new clusters with cosine distance. For total numbers of samples in each original class, see Table 6.1 or Table 6.2.*

Class	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5	Cluster 6	Cluster 7	Cluster 8	Cluster 9	Cluster 10	Cluster 11
A		5									
B					12						
C					2		17				6
Cb							2				2
Cg							2				
Cgh							8				2
Ch	1						17				1
D				19							1
K	2						9			1	3
L			5			13	2			8	5
Q	42										
Qw			1								
S	4		5			45				91	1
Sa		2	1								
Sq	17		5			9	1			9	1
Sqw			17								
Sr	1		1			16				13	
Srw			7							1	
Sv						1				2	
Svw			1							1	
Sw			49							8	
T											4
V								15	10		
Vw									2		
X				1			1				13
Xc							2				1
Xe							7			1	2
Xk							10				11
Xn							1				

slopes. However, even with this variance in the clusters, their shapes in Figure 6.5 are still quite uniform. Evaluation of the variance's significance in practice is presented in the next section.

6.4 Evaluation of Suggested Clusters

While the last section described the apparent trends in the formed clusters, the discussion was not extended far into the possible underlying reasons for them and how they compare to the previous taxonomies. Hence, the practical meaning of the results shall be discussed here. The focus will be mainly on the clusters produced with cosine distance, as they seemed to provide the best overall results.

Investigating the quality of the results requires determination of the significance of the clusters, as well as whether they seem to represent the divisions in the data well. This can be accomplished through scrutinizing the spectral graphs that were displayed and discussed in the last section. It is clear that distinctive classes such as A, D, and V, that have clear features and are significantly different from the other classes, are clustered well. This is to be expected, as it should be relatively easy for the algorithm to separate them from the other samples due to these distinctive features.

The spectral shapes for the formed clusters are overall rather uniform, especially when cosine distance is used. However, some intriguing features remain. One of these is the fact that a larger number of clusters begins to separate classes originally defined by the DeMeo taxonomy. This is most notable with V-class samples in the clustering done with eleven cosine clusters. The separation could either indicate a lack of intricacy in the original taxonomy or the clustering itself. Further studies would have to be done to fully determine which component the problem is caused by. Therefore, even though using eleven clusters produces clusters that appear uniform in shape, the meaningfulness of the results is questionable. In addition, it is difficult to tell how uniform the within-cluster shapes are, especially when they include classes that are relatively featureless. It is possible that it is difficult for the algorithm to properly separate samples within these clusters, especially when the number of clusters is small.

The variance in slopes clearly affects the clustering. This is most apparent in the way the algorithm singles out samples that are marked as having particularly high slopes. There are positive and negative sides to this behaviour. On the

positive side, the fact it is able to quite robustly identify samples with high slopes could find applications in searching for asteroids that have likely been affected by space weathering. It could also help in verification of what should be considered a high-slope sample of a specific class. On the negative side, it can also be deemed harmful that the algorithm considers the high-slope samples to be different from their normal counterparts, as they likely should still be in the same class. The strong focus on the slope could also mean that the algorithm prioritizes it over the more delicate features in the data. Further tests where the slope is removed from the data could be done in the future, although this would mean that the algorithm would likely struggle with distinguishing some classes, like B and C.

Another unexpected feature is how the algorithm combines some samples together. A good example of this is how Sq-samples are more readily combined with the Q-class rather than the S-class, even though their original classification indicates that the clustering should naturally go the other way around. While these aspects, together with the divisions formed based on the slopes, could be interpreted as possible improvements to make to the original taxonomy, truly being able to make this claim would once again require much more extensive research and further clustering tests that are outside the scope of this study. In general, however, the obtained clusters *are* mainly very similar to the original classes, as is seen in the intensity of highlights in the tables in the previous section.

It is ironic that even unsupervised learning methods, which are applied in this thesis in order to reduce human interference, often eventually require this human-provided evaluation to ensure the clustering results are logical. While the suggested clusters seem mostly meaningful, there are still some whose significance remains unclear due to the large number of samples from different classes being placed into them. It is clear that the reference taxonomy used for evaluating the formed clusters has a large impact on how the quality of the clusters is interpreted. Improvements in the clustering quality, as well as true confidence in the results, would come from a much larger dataset and more exhaustive study on the behaviour of the algorithm. Therefore, while the algorithm's performance is quite robust and produces results of generally good quality, it might still be a while before these methods can truly rival the original, human-produced, taxonomies.

7 CONCLUSION

The research question "How can asteroid spectra be analyzed using machine learning?" was explored in this thesis through two tasks. Task I resulted in the production of optimized locations for three to five spectrophotometric filters to use in asteroid classification through utilizing an optimization algorithm with access to a neural network. The confidence in placements was good, and there was no significant variance in results when using different amounts of repeats or different initial points. Determining the meaning behind the chosen placements was also made possible by analyzing their contributions to the classification success and correlation to spectroscopic features.

Task II produced new taxonomies through unsupervised learning with k-means. The quality of the resulting clusters, particularly those made with cosine distance, were in general satisfactory. Comparison of the formed clusters to the classes in the original taxonomy highlighted details of the algorithm's decision-making process, as well as distinctions in how a machine-generated taxonomy differs from those made by humans. The results of both Task I and Task II, therefore, overall imply that the explored tasks are valid applications of machine learning to asteroid spectroscopy, and as such answer the presented research question.

Scientifically, the obtained results are important, as they can suggest improvements to the practices in use today. Optimization of the locations for filters in spectroscopy is a novel idea, since so far there exists no standardized method for choosing them. The lack of standardization leads to significant variance between studies, and the quality of the obtained results can never be completely certain, especially if one wants to explore wavelengths outside, e.g., the generally favoured ECAS locations. Not only does the optimization method presented here

remove some of that uncertainty, but it could also be easily adapted to serve as an optimizer for several kinds of filters, classification systems, or spectral ranges. While the unsupervised clustering taxonomy does not necessarily outperform the previous systems in use, it concretely illustrates the benefits machine-generated taxonomies have to offer and what is required for their further development.

However, improvements to the methods developed for the two tasks could be made in the future. First of all, both would benefit from being able to utilize a larger dataset. Obtaining such a dataset was not possible during the time of writing this thesis, but could be feasible in the next few years. The ability to expand the dataset so that no classes have to be left out due to lack of population would yield results that represent reality more accurately, particularly since the need for simulating samples would be reduced. With more time, the machine learning methods could also be improved to higher complexity, which would possibly allow for the data to be handled in a more robust manner. An example of such an improvement would be implementing a better way to regularize the spectra in order to minimize the effect of the choice of normalization point.

Overall, this thesis lands in a time when both the fields of asteroid spectroscopy and machine learning are rapidly evolving. Access to increasingly larger amounts of asteroid data is changing how we view the Universe around us. The machine learning methods we use are being improved at a fast pace, which allows efficient exploration of new research goals. In order to answer the questions these goals evoke, they are adapted to serve new purposes every day; a trend that this thesis comprehensively illustrates.

REFERENCES

[1] M. J. Gaffey, T. H. Burbine, and R. P. Binzel. Asteroid Spectroscopy: Progress and Perspectives. In: *Meteoritics* 28.2 (1993), 161–187.

[2] S. J. Bus, F. Vilas, and M. Barucci. Visible-Wavelength Spectroscopy of Asteroids. In: *Asteroids III*. Ed. by W. Bottke, A. Cellino, P. Paolicchi, and R. Binzel. Tucson, United States: University of Arizona Press, 2002, 169–182.

[3] D. Tholen. Asteroid Taxonomy from Cluster Analysis of Photometry. PhD thesis. University of Arizona, 1984.

[4] D. J. Tholen. Asteroid Taxonomic Classification. In: *Asteroids II*. Ed. by R. P. Binzel, T. Gehrels, and M. S. Matthews. Tucson, United States: University of Arizona Press, 1989, 1139–1150.

[5] S. J. Bus and R. P. Binzel. Phase II of the Small Main-Belt Asteroid Spectroscopic Survey: A Feature-Based Taxonomy. In: *Icarus* 158.1 (2002), 146–177.

[6] F. E. Demeo, R. P. Binzel, S. M. Slivan, and S. J. Bus. An Extension of the Bus Asteroid Taxonomy Into the Near-Infrared. In: *Icarus* 202.1 (2009), 160–180.

[7] M. Delbo, J. Gayon-Markt, G. Busso, A. Brown, L. Galluccio, C. Ordenovic, P. Bendjoya, and P. Tanga. Asteroid Spectroscopy with Gaia. In: *Planetary and Space Science* 73.1 (2012), 86–94.

[8] J. A. Sanchez, V. Reddy, A. Nathues, E. A. Cloutis, P. Mann, and H. Hiesinger. Phase Reddening on Near-Earth Asteroids: Implications for Mineralogical Analysis, Space Weathering and Taxonomic Classification. In: *Icarus* 220.1 (2012), 36–50.

[9] V. Reddy, J. A. Sanchez, E. A. Cloutis, P. Mann, M. R. M. Izawa, L. L. Corre, M. Cuddy, M. Gaffey, and G. Fujihara. Impact Melt Origin of Baptistina Asteroid Family: Lessons from the Chelyabinsk Meteorite Fall. In: *Lunar and Planetary Science Conference*. Vol. 45. 2014.

[10] R. Binzel, F. DeMeo, E. Turtelbloom, S. Bus, A. Tokunaga, T. Burbine, C. Lantz, D. Polishook, B. Carry, A. Morbidelli, M. Birlan, P. Vernazza, B. Burt, N. Moskovitz, S. Slivan, C. Thomas, A. Rivkin, M. Hicks, T. Dunn, V. Reddy, J. Sanchez, M. Granvik, and T. Kohout. Compositional Distributions and Evolutionary Processes for the Near-Earth Object Population: Results from the MIT-Hawaii Near-Earth Object Spectroscopic Survey (MITHNEOS). In: *Icarus* 324 (2019), 41–76.

[11] G. W. Khazanov, ed. *Space Weather Fundamentals*. Boca Raton, United States: CRC Press, 2016, 304.

[12] M. Delbo, C. Avdellidou, and A. Morbidelli. Ancient and Primordial Collisional Families as the Main Sources of X-Type Asteroids of the Inner Main Belt. In: *Astronomy and Astrophysics* 624.A69 (2019).

[13] S. J. Bus and R. P. Binzel. Phase II of the Small Main-Belt Asteroid Spectroscopic Survey: The Observations. In: *Icarus* 158.1 (2002), 106–145.

[14] B. Zellner, D. Tholen, and E. Tedesco. The Eight-Color Asteroid Survey: Results for 589 Minor Planets. In: *Icarus* 61.3 (1985), 355–416.

[15] P. Vernazza, R. P. Binzel, A. Rossi, M. Fulchignoni, and M. Birlan. Solar Wind as the Origin of Rapid Reddening of Asteroid Surfaces. In: *Nature* 458 (2009), 993–995.

[16] J. Lever, M. Krzywinski, and N. Altman. Principal Component Analysis. In: *Nature Methods* 14 (2017), 641–642.

[17] S. J. Bus. Compositional Structure in the Asteroid Belt: Results of a Spectroscopic Survey. PhD thesis. Massachusetts Institute of Technology, 1999.

[18] C. R. Chapman, D. Morrison, and B. Zellner. Surface Properties of Asteroids: A Synthesis of Polarimetry, Radiometry, and Spectrophotometry. In: *Icarus* 25.1 (1975), 104–130.

[19] E. Bowell, C. R. Chapman, J. C. Gradie, D. Morrison, and B. Zellner. Taxonomy of Asteroids. In: *Icarus* 35.3 (1978), 313–335.

[20] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. Cambridge, United States: MIT Press, 2016, 1–18.

[21] T. C. Silva and L. Zhao. Machine Learning. In: *Machine Learning in Complex Networks*. Cham, Switzerland: Springer, 2016, 71–79.

[22] L. von Ahn and L. Dabbish. Labeling Images with a Computer Game. In: *CHI '04 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. Vienna, Austria: ACM, 2004, 319–326.

[23] A. Ball, D. Rye, F. Ramos, and M. Velonaki. Unsupervised Clustering of People from 'Skeleton' Data. In: *2012 7th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. Boston, United States: IEEE, 2012, 225–226.

[24] B. Clarkson and A. Pentland. Unsupervised Clustering of Ambulatory Audio and Video. In: *1999 IEEE International Conference on Acoustics, Speech and Signal Processing. Proceedings*. Vol. 6. Phoenix, United States: IEEE, 1999, 3037–3040.

[25] Y. Zhuang, Y. Rui, T. Huang, and S. Mehrotra. Adaptive Key Frame Extraction Using Unsupervised Clustering. In: *Proceedings 1998 International Conference on Image Processing*. Vol. 1. Chicago, United States: IEEE, 1998, 866–870.

[26] H. Liu, S. Shah, and W. Jiang. On-line Outlier Detection and Data Cleaning. In: *Computers & Chemical Engineering* 28.9 (2004), 1635–1647.

[27] C.-T. Lu, D. Chen, and Y. Kou. Algorithms for Spatial Outlier Detection. In: *Proceedings of the 3rd IEEE International Conference on Data Mining*. IEEE Computer Society, 2003.

[28] L. Prechelt. Early Stopping - But When? In: *Neural Networks: Tricks of the Trade. Lecture Notes in Computer Science*. Ed. by G. Mantavon, G. Orr, and K. Müller. Vol. 7700. Berlin, Germany: Springer, 2012, 53–67.

[29] N. Ganesh and N. G. Anderson. Dissipation in Neuromorphic Computing: Fundamental Bounds for Feedforward Networks. In: *Proceedings of the 17th IEEE International Conference on Nanotechnology*. Pittsburgh, United States: IEEE, 2017, 594–599.

[30] J. Han and C. Moraga. The Influence of the Sigmoid Function Parameters on the Speed of Backpropagation Learning. In: *IWANN 1995: From Natural to Artificial Neural Computation. Lecture Notes in Computer Science*. Ed. by J. Mira and F. Sandoval. Vol. 930. Berlin, Germany: Springer, 1995, 195–201.

[31] F. Guenther and S. Fritsch. Neuralnet: Training of Neural Networks. In: *The R Journal* 2.1 (2010), 30–38.

[32] M. F. Møller. A Scaled Conjugate Gradient Algorithm for Fast Supervised Learning. In: *Neural Networks* 6 (1993), 525–533.

[33] G. Zaccone and R. Karim. *Deep Learning with Tensorflow*. Birmingham, United Kingdom: Packt, 2018, 128.

[34] J. V. Tu. Advantages and Disadvantages of Using Artificial Neural Networks Versus Logistic Regression for Predicting Medical Outcomes. In: *Journal of Clinical Epidemiology* 49.11 (1996), 1225–1231.

[35] R. Lippmann. Pattern Classification Using Neural Networks. In: *IEEE Communications Magazine* 27.11 (1989), 47–50.

[36] A. F. Cheng. Near Earth Asteroid Rendezvous: Mission Summary. In: *Asteroids III*. Ed. by W. B. Jr, A. Cellino, P. Paolicchi, and R. Binzel. Tucson, United States: University of Arizona Press, 2002, 351–366.

[37] M. Fukugita, T. Ichikawa, J. Gunn, M. Doi, K. Shimasaku, and D. Schneider. The Sloan Digital Sky Survey Photometric System. In: *Astronomical Journal* 111 (1996), 1748–1756.

[38] H. Sierks, H. Keller, R. Jaumann, H. Michalik, T. Behnke, F. Bubenhagen, I. Büttner, U. Carsenty, U. Christensen, R. Enge, B. Fiethe, P. G. Marqués, H. Hartwig, H. Krüger, W. K. nad T. Maue, S. Mottola, A. Nathues, K.-U. Reiche, M. Richards, T. Roatsch, S. Schröder, I. Szemerey, and M. Tschentscher. The Dawn Framing Camera. In: *Space Science Reviews* 163.1-4 (2011), 263–327.

[39] Z. Ivezic, M. Juric, R. Lupton, S. Tabachnik, and T. Quinn. Asteroids Observed by the Sloan Digital Sky Survey. In: *Proceedings of SPIE - The International Society for Optical Engineering* 4836 (2002).

[40] F. Vilas and M. Gaffey. Phyllosilicate Absorption Features in Main-Belt and Outer-Belt Asteroid Reflectance Spectra. In: *Science* 246 (1989), 790–792.

[41] D. Morate, J. de León, M. de Prá, J. Licandro, A. Cabrera-Lavers, H. Campins, N. Pinilla-Alonso, and V. Alí-Lagoa. Compositional Study of Asteroids in the Erigone Collisional Family Using Visible Spectroscopy at the 10.4m GTC. In: *Astronomy & Astrophysics* 586 (2015).

[42] F. E. Demeo, C. O. Alexander, K. Walsh, C. Chapman, and R. Binzel. The Compositional Structure of the Asteroid Belt. In: *Asteroids IV*. Ed. by P. Michel, F. E. DeMeo, and W. F. Bottke. Tucson, United States: University of Arizona Press, 2015, 13–42.

[43] A. S. Rivkin, E. Howell, F. Vilas, and L. A. Lebofsky. Hydrated Minerals on Asteroids: The Astronomical Record. In: *Asteroids III*. Ed. by W. Bottke, A. Cellino, P. Paolicchi, and R. Binzel. Tucson, United States: University of Arizona Press, 2002, 235–253.

[44] I. Ordovás-Pascual and J. S. Almeida. A Fast Version of the K-means Classification Algorithm for Astronomical Applications. In: *Astronomy & Astrophysics* 565 (2014).

[45] L. Galluccio, O. Michel, P. Bendjoya, and E. Slezak. Unsupervised Clustering on Astrophysics Data: Asteroids Reflectance Spectra Surveys and Hyperspectral Images. In: *AIP Conference Proceedings* 1082 (2008), 165–171.

[46] R. Honda, Y. Yokota, E. Tatsumi, R. Hayashi, A. Barucci, D. Perna, M. Matsuoka, D. L. Domingue, T. Morota, S. Kameda, T. Kouyama, H. Suzuki, M. Yamada, N. Sakatani, C. Honda, L. Lecorre, M. Hayakawa, K. Yoshioka, Y. Cho, Y. Yamamoto, N. Hirata, Y. Fujii, T. Nakamura, T. Hiroi, H. Sawada, and S. Sugita. Clustering Analysis of Visible Spectra of Asteroid Ryugu and Its Preliminary Global Spectrum Map. In: *Lunar and Planetary Science Conference*. Vol. 50. 2019.

[47] J. de León, N. Pinilla-Alonso, H. Campins, J. Licandro, and G. Marzo. Near-Infrared Spectroscopic Survey of B-Type Asteroids: Compositional Analysis. In: *Icarus* 218.1 (2012), 196–206.

[48] D. Baron. *Machine Learning in Astronomy: A Practical Overview*. 2019. arXiv: 1904.07248.

[49] A. K. Jain. Data Clustering: 50 Years Beyond K-Means. In: *Pattern Recognition Letters* 31.8 (2010), 651–666.

[50] D. J. Bora and A. K. Gupta. Effect of Different Distance Measures on the Performance of K-Means Algorithm: An Experimental Study in Matlab. In:

Journal of Computer Science and Information Technologies 5.2 (2014), 2501–2506.

[51] S. France, J. D. Carroll, and H. Xiong. Distance Metrics for High Dimensional Nearest Neighborhood Recovery: Compression and Normalization. In: *Information Sciences* 184.1 (2012), 92–110.

[52] J. Han, M. Kamber, and J. Pei. 2 - Getting to Know Your Data. In: *Morgan Kaufmann Series in Data Management Systems, Data Mining (Third Edition)*. Ed. by W. Bottke, A. Cellino, P. Paolicchi, and R. Binzel. Waltham, United States: Elsevier, 2012, 39–82.

[53] M. N. Murty and V. S. Devi. *Pattern Recognition: An Algorithmic Approach*. London, United Kingdom: Springer, 2011, 18–19.

[54] P. Rouseeuw. Silhouettes: A Graphical Aid to the Interpreation and Validation of Cluster Analysis. In: *Journal of Computational and Applied Mathematics* 20.1 (1987), 53–65.

[55] P. Bholowalia and A. Kumar. EBK-Means: A Clustering Technique Based on Elbow Method and K-Means in WSN. In: *International Journal of Computer Applications* 105.9 (2014), 17–24.

[56] C. Goutte, P. Toft, E. Rostrup, F. Å. Nielsen, and L. K. Hansen. On Clustering fMRI Time Series. In: *NeuroImage* 9.3 (1999), 298–310.

A DATA

Appendix A describes the preparation procedure for the datasets utilized in this study, as well as the full list of objects used for constructing them including their assigned number, name, source dataset, original classification, and reduced classification. The reduced classification is produced when all subclasses are combined into their main equivalents.

A.1 Preparation

Below is a list of the steps taken in order to prepare the datasets utilized in this study:

1. Initial combination of datasets by picking suitable cases. If the same object exists in both the BDM09 and MITHNEOS sets, only the MITHNEOS case is selected.
2. If the object's spectral range extends fully from 0.45 to 2.45 microns, a spline fit is done to obtain a total of 201 data points. If it does not, the spline fit of the available data is followed by a linear extrapolation to extend to the full range.
3. Normalization of all spectra to unity at 0.55 microns.
4. Removing the data points at 0.55 microns, as they now yield no new information.
5. Removal of cases that greatly stand out from the others in its class as well as unknown cases.

6. Preparing the two distinct sets for the two tasks
   (a) For the simulated set:
      i. If there is uncertainty in the classification of each asteroid, e.g., it could either be S or Sr, the first class is always chosen.
      ii. Reducing asteroids to their "main" class, e.g., Sr is reduced to S.
      iii. Removal of classes that only have few asteroids in them.
      iv. Simulating until all 11 reduced classes have 200 samples.
   (b) For the k-means set:
      i. Full classes are kept, even if there are only few samples in them.
      ii. Removal of cases that still visually stand out from the rest of the samples in each class.

A.2 Table of Objects

Table A.1. *List of all the asteroids in the utilized dataset. Object numbers, names, initial datasets along with the original classification and modified classification are provided whenever possible. * = Number verified from JPL database.*

Number	Name	Source	Original Classification	Reduced Classification
1	Ceres	BDM09	C	C
2	Pallas	BDM09	B	B
3	Juno	BDM09	Sq	S
4	Vesta	BDM09	V	V
5	Astraea	BDM09	S	S
7	Iris	BDM09	S	S
8	Flora	BDM09	Sw	S
10	Hygiea	BDM09	C	C
11	Parthenope	BDM09	Sq	S
13	Egeria	BDM09	Ch	C
14	Irene	BDM09	S	S
15	Eunomia	BDM09	K	K

Continued on next page

Continued from previous page

Number	Name	Source	Original Classification	Reduced Classification
16	Psyche	BDM09	Xk	X
17	Thetis	BDM09	S	S
18	Melpomene	BDM09	S	S
19	Fortuna	BDM09	Ch	C
20	Massalia	BDM09	S	S
21	Lutetia	BDM09	Xc	X
22	Kalliope	BDM09	X	X
24	Themis	BDM09	C	C
25	Phocaea	BDM09	S	S
26	Proserpina	BDM09	S	S
27	Euterpe	BDM09	S	S
28	Bellona	BDM09	S	S
29	Amphitrite	BDM09	S	S
30	Urania	BDM09	S	S
32	Pomona	BDM09	Sw	S
33	Polyhymnia	BDM09	S	S
34	Circe	BDM09	Ch	C
37	Fides	BDM09	S	S
38	Leda	BDM09	Cgh	C
40	Harmonia	BDM09	S	S
41	Daphne	BDM09	Ch	C
42	Isis	BDM09	K	K
43	Ariadne	BDM09	Sq	S
48	Doris	BDM09	Ch	C
49	Pales	BDM09	Ch	C
50	Virginia	BDM09	Ch	C
51	Nemausa	BDM09	Cgh	C
52	Europa	BDM09	C	C
54	Alexandra	BDM09	Cgh	C
55	Pandora	BDM09	Xk	X
56	Melete	BDM09	Xk	X
57	Mnemosyne	BDM09	S	S
58	Concordia	BDM09	Ch	C

Continued on next page

Continued from previous page

Number	Name	Source	Original Classification	Reduced Classification
61	Danae	BDM09	S	S
63	Ausonia	BDM09	Sw	S
64	Angelina	BDM09	Xe	X
65	Cybele	BDM09	Xk	X
66	Maja	BDM09	Ch	C
67	Asia	BDM09	S	S
69	Hesperia	BDM09	Xk	X
70	Panopaea	BDM09	Cgh	C
76	Freia	BDM09	C	C
77	Frigga	BDM09	Xe	X
78	Diana	BDM09	Ch	C
82	Alkmene	BDM09	S	S
84	Klio	BDM09	Ch	C
85	Io	BDM09	C	C
87	Sylvia	BDM09	X	X
90	Antiope	BDM09	C	C
92	Undina	BDM09	Xk	X
93	Minerva	BDM09	C	C
96	Aegle	BDM09	T	T
99	Dike	BDM09	Xk	X
101	Helena	BDM09	S	S
103	Hera	BDM09	S	S
105	Artemis	BDM09	Ch	C
106	Dione	BDM09	Cgh	C
108	Hecuba	BDM09	Sw	S
110	Lydia	BDM09	Xk	X
111	Ate	BDM09	Ch	C
114	Kassandra	BDM09	K	K
115	Thyra	BDM09	Xe	X
119	Althaea	BDM09	S	S
128	Nemesis	BDM09	C	C
130	Elektra	BDM09	Ch	C
131	Vala	BDM09	K	K

Continued on next page

Continued from previous page

Number	Name	Source	Original Classification	Reduced Classification
132	Aethra	MITHNEOS	Xe	X
133	Cyrene	BDM09	S	S
147	Protogeneia	BDM09	C	C
150	Nuwa	BDM09	C	C
151	Abundantia	BDM09	Sw	S
153	Hilda	BDM09	X	X
158	Koronis	BDM09	S	S
160	Una	BDM09	Xk	X
170	Maria	BDM09	S	S
175	Andromache	BDM09	Cg	C
180	Garumna	BDM09	Sr	S
181	Eucharis	BDM09	Xk	X
188	Menippe	BDM09	S	S
191	Kolga	BDM09	Cb	C
192	Nausikaa	BDM09	Sw	S
199	Byblis	BDM09	D	D
201	Penelope	BDM09	Xk	X
205	Martha	BDM09	Ch	C
210	Isabella	BDM09	Cb	C
214	Aschera	BDM09	Cgh	C
216	Kleopatra	BDM09	Xe	X
221	Eos	BDM09	K	K
226	Weringia	BDM09	S	S
233	Asterope	BDM09	Xk	X
234	Barbara	BDM09	L	L
236	Honoria	BDM09	L	L
237	Coelestina	BDM09	Sr	S
243	Ida	BDM09	Sw	S
244	Sita	BDM09	Sw	S
246	Asporina	BDM09	A	A
250	Bettina	BDM09	Xk	X
258	Tyche	BDM09	S	S
264	Libussa	BDM09	S	S

Continued on next page

Continued from previous page

Number	Name	Source	Original Classification	Reduced Classification
266	Aline	BDM09	Ch	C
267	Tirza	BDM09	D	D
269	Justitia	BDM09	D	D
278	Paulina	BDM09	S	S
279	Thule	BDM09	D	D
288	Glauke	BDM09	S	S
289	Nenetta	BDM09	A	A
295	Theresia	BDM09	Sw	S
308	Polyxo	BDM09	T	T
322	Phaeo	BDM09	D	D
337	Devosa	BDM09	Xk	X
345	Tercidina	BDM09	Ch	C
346	Hermentaria	BDM09	S	S
352	Gisela	BDM09	Sw	S
354	Eleonora	BDM09	A	A
359	Georgia	BDM09	Xk	X
371	Bohemia	BDM09	S	S
378	Holmia	BDM09	S	S
387	Aquitania	BDM09	L	L
389	Industria	BDM09	S	S
402	Chloe	BDM09	L	L
403	Cyane	BDM09	S	S
433	Eros	MITHNEOS	Sw	S
434	Hungaria	BDM09	Xe	X
444	Gyptis	BDM09	C	C
446	Aeternitas	BDM09	A	A
453	Tea	BDM09	Sw	S
456	Abnoba	BDM09	S	S
460	Scania	BDM09	L	L
485	Genua	BDM09	S	S
512	Taurinensis	MITHNEOS	Sqw	S
513	Centesima	BDM09	K	K
532	Herculina	BDM09	S	S

Continued on next page

Continued from previous page

Number	Name	Source	Original Classification	Reduced Classification
570	Kythera	BDM09	D	D
579	Sidonia	BDM09	K	K
596	Scheila	BDM09	T	T
599	Luisa	BDM09	L	L
606	Brangane	BDM09	L	L
625	Xenia	BDM09	Sw	S
631	Philippina	BDM09	S	S
653	Berenike	BDM09	K	K
661	Cloelia	BDM09	K	K
670	Ottegebe	BDM09	S	S
673	Edda	BDM09	L	L
675	Ludmilla	BDM09	Sw	S
679	Pax	BDM09	L	L
688	Melanie	BDM09	C	C
699	Hela	MITHNEOS	Sq	S
706	Hirundo	BDM09	Cgh	C
716	Berkeley	BDM09	S	S
719	Albert	MITHNEOS	S	S
720	Bohlinia	BDM09	Sq	S
729	Watsonia	BDM09	L	L
739	Mandeville	BDM09	Xc	X
742	Edisona	BDM09	K	K
773	Irmintraud	BDM09	T	T
776	Berbericia	BDM09	Cgh	C
782	Montefiore	BDM09	Sw	S
785	Zwetana	BDM09	Cb	C
789	Lena	BDM09	Xk	X
793	Arizona	BDM09	S	S
808	Merxia	BDM09	Sr	S
824	Anastasia	BDM09	L	L
847	Agnia	BDM09	S	S
863	Benkoela	BDM09	A	A
887	Alinda	MITHNEOS	S	S

Continued on next page

Continued from previous page

Number	Name	Source	Original Classification	Reduced Classification
908	Buda	BDM09	D	D
913	Otila	BDM09	Sw	S
925	Alphonsina	BDM09	S	S
929	Algunde	BDM09	S	S
944	Hidalgo	BDM09	D	D
984	Gretia	BDM09	Sa	S
985	Rosina	BDM09	S	S
1011	Laodamia	MITHNEOS	Sqw	S
1020	Arcadia	BDM09	Sr	S
1036	Ganymed	MITHNEOS	Sr	S
1065	Amundsenia	BDM09	S	S
1094	Siberia	BDM09	Xk	X
1126	Otero	BDM09	Sw	S
1131	Porzia	MITHNEOS	S	S
1139	Atami	MITHNEOS	Sw	S
1143	Odysseus	BDM09	D	D
1147	Stravropolis	BDM09	Sw	S
1148	Rarahu	BDM09	K	K
1198	Atlantis	MITHNEOS	Sw	S
1204	Renzia	MITHNEOS	Sw	S
1228	Scabiosa	BDM09	Sr	S
1300	Marcelle	BDM09	Cgh	C
1310	Villigera	MITHNEOS	S	S
1329	Eliane	BDM09	Sqw	S
1332	Marconia	BDM09	L	L
1350	Rosselia	BDM09	S	S
1374	Isora	MITHNEOS	Sq	S
1433	Geramtina	BDM09	S	S
1459	Magnya	BDM09	Vw	V
1468	Zomba	MITHNEOS	V	V
1471	Tornio	BDM09	D	D
1494	Savo	BDM09	Sqw	S
1508	Kemi	MITHNEOS	B	B

Continued on next page

Continued from previous page

Number	Name	Source	Original Classification	Reduced Classification
1542	Schalen	BDM09	D	D
1565	Lemaitre	MITHNEOS	Sr	S
1566	Icarus	MITHNEOS	Q	Q
1580	Betulia	MITHNEOS	B	B
1620	Geographos	MITHNEOS	S	S
1627	Ivar	MITHNEOS	Sw	S
1640	Nemo	MITHNEOS	S	S
1642	Hill	BDM09	S	S
1658	Innes	BDM09	Sw	S
1659	Punkaharju	BDM09	S	S
1660	Wood	BDM09	S	S
1662	Hoffmann	BDM09	Sr	S
1667	Pels	BDM09	Sw	S
1685	Toro	MITHNEOS	Sq	S
1747	Wright	MITHNEOS	Sw	S
1751	Herget	BDM09	S	S
1807	Slovakia	BDM09	Sqw	S
1839	Ragazza	BDM09	S	S
1848	Delvaux	BDM09	S	S
1858	Lobachevskij	BDM09	S	S
1862	Apollo	MITHNEOS	Q	Q
1864	Daedalus	MITHNEOS	Sq	S
1865	Cerberus	MITHNEOS	S	S
1866	Sisyphus	MITHNEOS	Sw	S
1903	Adzhimushkaj	BDM09	K	K
1904	Massevitch	BDM09	V	V
1916	Boreas	MITHNEOS	Sw	S
1917	Cuyo	MITHNEOS	Sv	S
1929	Kollaa	BDM09	V	V
1943	Anteros	MITHNEOS	Sw	S
1951	Lick	MITHNEOS	A	A
1980	Tezcatlipoca	MITHNEOS	Sw	S
1981	Midas	MITHNEOS	V	V

Continued on next page

Continued from previous page

Number	Name	Source	Original Classification	Reduced Classification
2035	Stearns	BDM09	Xe	X
2042	Sitarski	BDM09	Sr	S
2045	Peking	BDM09	V	V
2063	Bacchus	MITHNEOS	Sq	S
2064	Thomsen	MITHNEOS	Sqw	S
2074	Shoemaker	MITHNEOS	Sw	S
2078	Nanking	MITHNEOS	S	S
2085	Henan	BDM09	L	L
2099	Opik	MITHNEOS	Ch	C
2100	Ra-Shalom	MITHNEOS	B	B
2102	Tantalus	MITHNEOS	Sr	S
2107	Ilmari	BDM09	Sw	S
2157	Ashbrook	BDM09	S	S
2201	Oljato	MITHNEOS	Sq	S
2212	Hephaistos	MITHNEOS	Q	Q
2246	Bowell	BDM09	D	D
2335	James	MITHNEOS	Sw	S
2340	Hathor	MITHNEOS	Sq	S
2353	Alva	BDM09	S	S
2354	Lavrov	BDM09	L	L
2378	Pannekoek	BDM09	Cgh	C
2386	Nikonov	BDM09	S	S
2396	Kochi	BDM09	S	S
2401	Aehlita	BDM09	S	S
2442	Corbett	BDM09	V	V
2448	Sholokhob	BDM09	L	L
2449	Kenos	MITHNEOS	Xc	X
2501	Lohja	BDM09	A	A
2504	Gaviola	BDM09	Sr	S
2521	Heidi	BDM09	S	S
2566	Kirghizia	BDM09	V	V
2579	Spartacus	BDM09	V	V
2715	Mielikki	BDM09	Sw	S

Continued on next page

Continued from previous page

Number	Name	Source	Original Classification	Reduced Classification
2732	Witt	BDM09	L	L
2851	Harbin	BDM09	V	V
2873	Binzel	BDM09	Sq	S
2875	Lagerkvist	BDM09	S	S
2911	Miahelena	BDM09	Sw	S
2912	Lapalma	BDM09	V	V
2957	Tatsuo	BDM09	K	K
2965	Surikov	BDM09	Sv	S
2977	Chivilikhin	BDM09	S	S
3028	Zangguoxi	BDM09	K	K
3102	Krok	MITHNEOS	Sqw	S
3103	Eger	MITHNEOS	Xe	X
3122	Florence	MITHNEOS	Swq	S
3155	Lee	BDM09	V	V
3198	Wallonia	MITHNEOS	Sqw	S
3199	Nefertiti	MITHNEOS	K	K
3200	Phaethon	MITHNEOS	B	B
3248	Farinella	BDM09	D	D
3255	Tholen	MITHNEOS	S	S
3288	Seleucus	MITHNEOS	Sqw	S
3317	Paris	BDM09	D	D
3352	McAuliffe	MITHNEOS	Sw	S
3361	Orpheus	MITHNEOS	Q	Q
3363	Bowen	BDM09	Sr	S
3395	Jitka	BDM09	S	S
3402	Wisdom	MITHNEOS	S	S
3430	Bradfield	BDM09	S	S
3491	Fridolin	BDM09	S	S
3511	Tsvetaeva	BDM09	Srw	S
3552	Don Quixote	MITHNEOS	D	D
3554	Amun	MITHNEOS	X	X
3635	Kreutz	MITHNEOS	Srw	S
3671	Dionysus	MITHNEOS	Xn	X

Continued on next page

Continued from previous page

Number	Name	Source	Original Classification	Reduced Classification
3674	Erbisbuhl	MITHNEOS	S	S
3691	Bede	MITHNEOS	Xk	X
3701	Purkyne	BDM09	S	S
3734	Waland	BDM09	L	L
3753	Cruithne	MITHNEOS	Q	Q
3788	Steyaert	BDM09	S	S
3833	Calingasta	MITHNEOS	C	C
3844	Lujiaxi	BDM09	L	L
3858	Dorchester	MITHNEOS	Srw	S
3873	Roddy	BDM09	Sw	S
3903	Kliment Ohridski	BDM09	S	S
3908	Nyx	MITHNEOS	V	V
3910	Liszt	BDM09	S	S
3920	Aubignan	MITHNEOS	Sw	S
3949	Mach	BDM09	Sq	S
3988	Huma	MITHNEOS	S	S
4015	Wilson-Harrington	MITHNEOS	B	B
4038	Kristina	BDM09	Vw	V
4055	Magellan	MITHNEOS	V	V
4179	Toutatis	MITHNEOS	Sq	S
4183	Cuno	MITHNEOS	Q	Q
4188	Kitezh	BDM09	V	V
4197	Morpheus	MITHNEOS	Sq	S
4352	Kyoto	BDM09	S	S
4407	Taihaku	BDM09	Sqw	S
4417	Lecar	BDM09	Sw	S
4451	Grieve	MITHNEOS	Svw	S
4486	Mithra	MITHNEOS	Sq	S
4558	Janesick	MITHNEOS	Sr	S
4570	Runcorn	BDM09	Sw	S
4587	Rees	MITHNEOS	Sr	S
4660	Nereus	MITHNEOS	Xe	X

Continued on next page

Continued from previous page

Number	Name	Source	Original Classification	Reduced Classification
4688	1980 WF	MITHNEOS	Q	Q
4713	Steel	BDM09	Sw	S
4737	Kiladze	BDM09	L	L
4775	Hansen	MITHNEOS	L	L
4954	Eric	MITHNEOS	Srw	S
4995	Griffen	MITHNEOS	S	S
5011	Ptah	MITHNEOS	Q	Q
5013	Suzhou-sanzhong	BDM09	Sw	S
5111	Jacliff	BDM09	V	V
5131	1990 BG	MITHNEOS	Sa	S
5143	Heracles	MITHNEOS	Q	Q
5230	Asahina	MITHNEOS	S	S
5261	Eureka	MITHNEOS	Sa	S
5379	Abehiroshi	BDM09	Sr	S
5392	Parker	MITHNEOS	Sv	S
5401	Minamioda	BDM09	Sw	S
5407	1992 AX	MITHNEOS	S	S
5587	1990 SB	MITHNEOS	Sr	S
5604	1992 FE	MITHNEOS	V	V
5626	1991 FE	MITHNEOS	S	S
5641	McCleese	BDM09	Sw	S
5645	1990 SP	MITHNEOS	X	X
5646	1990 TR	MITHNEOS	Q	Q
5660	1974 MA	MITHNEOS	Q	Q
5685	Sanenobufukui	BDM09	S	S
5693	1993 EA	MITHNEOS	S	S
5786	Talos	MITHNEOS	Q	Q
5817	Robertfrazer	MITHNEOS	Sr	S
5836	1993 MF	MITHNEOS	S	S
5840	Raybrown	BDM09	L	L
6037	1988 EG	MITHNEOS	Q	Q
6047	1991 TB1	BDM09	S	S

Continued on next page

Continued from previous page

Number	Name	Source	Original Classification	Reduced Classification
6239	Minos	MITHNEOS	Sq	S
6386	Keithnoll	MITHNEOS	S	S
6411	Tamaga	MITHNEOS	B	B
6455	1992 HE	MITHNEOS	Sqw	S
6456	Golombek	MITHNEOS	Q	Q
6585	O'Keefe	MITHNEOS	S	S
6611	1993 VW	MITHNEOS	V	V
7304	Namiki	MITHNEOS	L	L
7336	Saunders	MITHNEOS	Q	Q
7341	1991 VK	MITHNEOS	Q	Q
7358	Oze	MITHNEOS	Sq	S
7482	1994 PC1	MITHNEOS	S	S
7753	1988 XB	MITHNEOS	Cb	C
7763	Crabeels	BDM09	L	L
7822	1991 CS	MITHNEOS	S	S
7888	1993 UC	MITHNEOS	S	S
7889	1994 LX	MITHNEOS	V	V
8334	1984 CF	BDM09	S	S
8373	Stephengould	MITHNEOS	D	D
8566	1996 EN	MITHNEOS	V	V
8567	1996 HW1	MITHNEOS	Sw	S
9400	1994 TW1	MITHNEOS	S	S
10115	1992 SK	MITHNEOS	S	S
10145	1994 CK1	MITHNEOS	Q	Q
10150	1994 PN	MITHNEOS	S	S
10302	1989 ML	MITHNEOS	X	X
11066	Sigurd	MITHNEOS	S	S
11398	1998 YP11	MITHNEOS	Sr	S
11405	1999 CV3	MITHNEOS	Sq	S
12711	Tukmit	MITHNEOS	Sqw	S
14402	1991 DB	MITHNEOS	Xk	X
15745	Yuliya	MITHNEOS	S	S
16834	1997 WU22	MITHNEOS	S	S

Continued on next page

Continued from previous page

Number	Name	Source	Original Classification	Reduced Classification
16960	1998 QS52	MITHNEOS	Sr	S
17274	2000 LC16	MITHNEOS	D	D
17511	1992 QN	MITHNEOS	B	B
18736	1998 NU	MITHNEOS	S	S
18882	1994 YN4	MITHNEOS	S	S
19127	Olegefremov	MITHNEOS	Srw	S
19356	1997 GH3	MITHNEOS	Sq	S
19764	2000 NF5	MITHNEOS	Sq	S
20786	2000 RG62	BDM09	Sq:	S
20790	2000 SE45	MITHNEOS	S	S
21088	Chelyabinsk	MITHNEOS	Sw	S
22753	1998 WT	MITHNEOS	Q	Q
22771	1993 CU3	MITHNEOS	S	S
24445	2000 PM8	MITHNEOS	Sr	S
24475	2000 VN2	MITHNEOS	Sw	S
25143	Itokawa	MITHNEOS	Sq	S
25330	1999 KV4	MITHNEOS	B	B
25916	2001 CP44	MITHNEOS	Sw	S
26760	2001 KP41	MITHNEOS	C	C
29075	1950 DA	MITHNEOS	L	L
30825	1990 TG1	MITHNEOS	Sq	S
32906	1994 RH	MITHNEOS	S	S
33342	1998 WT24	MITHNEOS	X	X
33881	2000 JK66	MITHNEOS	V	V
35107	1991 VH	MITHNEOS	Sq	S
35396	1997 XF11	MITHNEOS	S	S
36017	1999 ND43	MITHNEOS	S	S
36284	2000 DM8	MITHNEOS	Sq	S
37336	2001 RM	MITHNEOS	S	S
39572	1993 DQ1	MITHNEOS	Sq	S
52340	1992 SY	MITHNEOS	Q	Q
52760	1998 ML14	MITHNEOS	Q	Q
52762	1998 MT24	MITHNEOS	D	D

Continued on next page

Continued from previous page

Number	Name	Source	Original Classification	Reduced Classification
52768	1998 OR2	MITHNEOS	Xk	X
53319	1999 JM8	MITHNEOS	C	C
53435	1999 VM40	MITHNEOS	Srw	S
54690	2001 EB	MITHNEOS	S	S
54789	2001 MZ7	MITHNEOS	Xe	X
63164	2000 YU14	MITHNEOS	S	S
65679	1989 UQ	MITHNEOS	C	C
65803	Didymos	MITHNEOS	S	S
65996	1998 MX5	MITHNEOS	X	X
66063	1998 RO1	MITHNEOS	Sq	S
66146	1998 TU3	MITHNEOS	Q	Q
66251	1999 GJ2	MITHNEOS	Sw	S
68216	2001 CV26	MITHNEOS	S	S
68346	2001 KZ66	MITHNEOS	Sw	S
68350	2001 MK3	MITHNEOS	S	S
68359	2001 OZ13	MITHNEOS	S	S
68372	2001 PM9	MITHNEOS	C	C
68950	2002 QF15	MITHNEOS	S	S
85709	1998 SG36	MITHNEOS	S	S
85774	1998 UT18	MITHNEOS	Cg	C
85818	1998 XM4	MITHNEOS	Srw	S
85867	1999 BY9	MITHNEOS	Q	Q
85989	1999 JD6	MITHNEOS	L	L
85990	1999 JV6	MITHNEOS	S	S
86039	1999 NC43	MITHNEOS	Q	Q
86212	1999 TG21	MITHNEOS	S	S
86450	2000 CK33	MITHNEOS	L	L
86819	2000 GK137	MITHNEOS	Sq	S
87684	2000 SY2	MITHNEOS	Q	Q
88188	2000 XH44	MITHNEOS	V	V
88254	2001 FM129	MITHNEOS	Q	Q
88710	2001 SL9	MITHNEOS	S	S
89355	2001 VS78	MITHNEOS	Sr	S

Continued on next page

Continued from previous page

Number	Name	Source	Original Classification	Reduced Classification
96189	Pygmalion	MITHNEOS	B	B
96590	1998 XB	MITHNEOS	Q	Q
98943	2001 CC21	MITHNEOS	L	L
99799	2002 LJ3	MITHNEOS	Q	Q
99907	1989 VA	MITHNEOS	Sr	S
99942	Apophis	MITHNEOS	Sq	S
100926	1998 MQ	MITHNEOS	Sqw	S
101955	Bennu	MITHNEOS	B	B
102528	1999 US3	MITHNEOS	X	X
108519	2001 LF	MITHNEOS	C	C
136617	1994 CC	MITHNEOS	S	S
136923	1998 JH2	MITHNEOS	Sw	S
136993	1998 ST49	MITHNEOS	Sr	S
137032	1998 UO1	MITHNEOS	Q	Q
137062	1998 WM	MITHNEOS	Sr	S
137126	1999 CF9	MITHNEOS	Sq	S
137170	1999 HF1	MITHNEOS	Xk	X
137199	1999 KX4	MITHNEOS	S	S
137427	1999 TF211	MITHNEOS	S	S
137799	1999 YB	MITHNEOS	Sq	S
138258	2000 GD2	MITHNEOS	Sq	S
138404	2000 HA24	MITHNEOS	S	S
138524	2000 OJ8	MITHNEOS	Sr	S
138846	2000 VJ61	MITHNEOS	Sr	S
138852	2000 WN10	MITHNEOS	Q	Q
138911	2001 AE2	MITHNEOS	L	L
139622	2001 QQ142	MITHNEOS	Sq	S
141018	2001 WC47	MITHNEOS	Sw	S
141052	2001 XR1	MITHNEOS	Sq	S
142040	2002 QE15	MITHNEOS	Sw	S
143381	2003 BC21	MITHNEOS	S	S
143651	2003 QO104	MITHNEOS	Q	Q
144411	2004 EW9	MITHNEOS	L	L

Continued on next page

Continued from previous page

Number	Name	Source	Original Classification	Reduced Classification
145656	4788 PL	MITHNEOS	Srw	S
152560	1991 BN	MITHNEOS	Q	Q
152563	1992 BF	MITHNEOS	S	S
152931	2000 EA107	MITHNEOS	Q	Q
153591	2001 SN263	MITHNEOS	B	B
153814	2001 WN5	MITHNEOS	L	L
154029	2002 CY46	MITHNEOS	S	S
154276	2002 SY50	MITHNEOS	S	S
154302	2002 UQ3	MITHNEOS	Sq	S
154347	2002 XK4	MITHNEOS	S	S
155334	2006 DZ169	MITHNEOS	Sq	S
159402	1999 AP10	MITHNEOS	Sw	S
159635	2002 CZ46	MITHNEOS	L	L
159857	2004 LJ1	MITHNEOS	Sr	S
161998	1988 PA	MITHNEOS	S	S
162058	1997 AE12	MITHNEOS	Q	Q
162149	1998 YQ11	MITHNEOS	Sw	S
162186	1999 OP3	MITHNEOS	Sq	S
162483	2000 PJ5	MITHNEOS	Q	Q
162781	2000 XL44	MITHNEOS	S	S
162911	2001 LL5	MITHNEOS	S	S
162998	2001 SK162	MITHNEOS	D	D
163000	2001 SW169	MITHNEOS	Sw	S
163081	2002 AG29	MITHNEOS	S	S
163249	2002 GT	MITHNEOS	Q	Q
163364	2002 OD20	MITHNEOS	Q	Q
163697	2003 EF54	MITHNEOS	Q	Q
164202	2004 EW	MITHNEOS	Xe	X
170502	2003 WM7	MITHNEOS	C	C
175706	1996 FG3	MITHNEOS	C	C
180186	2003 QZ30	MITHNEOS	X	X
189552	2000 RL77	MITHNEOS	Sr	S
192563	1998 WZ6	MITHNEOS	V	V

Continued on next page

Continued from previous page

Number	Name	Source	Original Classification	Reduced Classification
200840	2001 XN254	MITHNEOS	S	S
214869	2007 PA8	MITHNEOS	Sq	S
217796	2000 TO64	MITHNEOS	Sr	S
217807	2000 XK44	MITHNEOS	Sqw	S
219071	1997 US9	MITHNEOS	Q	Q
236716	2007 FV42	MITHNEOS	S	S
241662	2000 KO44	MITHNEOS	Sw	S
242187	2003 KR18	MITHNEOS	Sqw	S
253841	2003 YG118	MITHNEOS	V	V
267494	2002 JB9	MITHNEOS	X	X
283460	2001 PD1	MITHNEOS	S	S
285263	1998 QE2	MITHNEOS	Ch	C
297418	2000 SP43	MITHNEOS	V	V
301964	2000 EJ37	MITHNEOS	D	D
302311	2002 AA	MITHNEOS	S	S
303174	2004 FH11	MITHNEOS	S	S
308635	2005 YU55	MITHNEOS	C	C
310442	2000 CH59	MITHNEOS	Sq	S
312473	2008 SX245	MITHNEOS	C	C
326290	Akhenaten	MITHNEOS	V	V
329437	2002 OA22	MITHNEOS	Q	Q
345705	2006 VB14	MITHNEOS	Q	Q
350523	2000 EA14	MITHNEOS	Qw	Q
354030	2001 RB18	MITHNEOS	C	C
363067	2000 CO101	MITHNEOS	X	X
365424	2010 KX7	MITHNEOS	Sw	S
399774	2005 NB7	MITHNEOS	Sq	S
401857	2000 PG3	MITHNEOS	D	D
405058	2001 TX16	MITHNEOS	X	X
413038	2001 MF1	MITHNEOS	Sr	S
414586	2009 UV18	MITHNEOS	Svw	S
416186	2002 TD60	MITHNEOS	S	S
	2002 AV	MITHNEOS	S	S

Continued on next page

Continued from previous page

Number	Name	Source	Original Classification	Reduced Classification
	2002 NY40	MITHNEOS	Q	Q
	2002 TP69	MITHNEOS	S	S
	2002 TS67	MITHNEOS	X	X
524516*	2002 UN	MITHNEOS	C	C
	2002 VP69	MITHNEOS	Sq	S
	2003 UB5	MITHNEOS	L	L
	2004 LU3	MITHNEOS	Sr	S
	2004 QD3	MITHNEOS	X	X
	2007 RU17	MITHNEOS	Q	Q
	2008 QS11	MITHNEOS	L	L

B NEURAL NETWORK TESTS

Appendix B summarizes the test results for the optimal number of neurons to include in the ONN and DNN. The procedure to produce these results is fundamentally the same as described in Section 4.2, although multilayer structures have been ignored here since they produced sub-optimal results in the original tests. These tests were made with a sample set of five well-succeeding locations, since they provide a good standard for data that has gone through the convolution process. The DNN was used to find the success rates. Table B.1 presents the obtained results and Figure B.1 plots them in a line graph. As was the case with the full range of data, the ONN and DNN are likely to succeed best with 40 neurons on the hidden layer.

Table B.1. *Success rates of the direct test neural network when using the wavelengths of best succeeding set in Appendix C Table C.2 with different neuron amounts in the hidden layer.*

Test	Neurons	Success (%)	Standard Deviation	Time (s)
1	5	91.761	2.447	341.858
2	10	95.461	1.285	355.482
3	15	96.198	1.338	369.401
4	20	96.593	1.210	367.154
5	30	96.764	1.178	397.210
6	40	96.869	1.160	424.173
7	50	96.733	1.524	458.773
8	100	96.525	1.987	690.442
9	200	95.770	3.064	1010.098

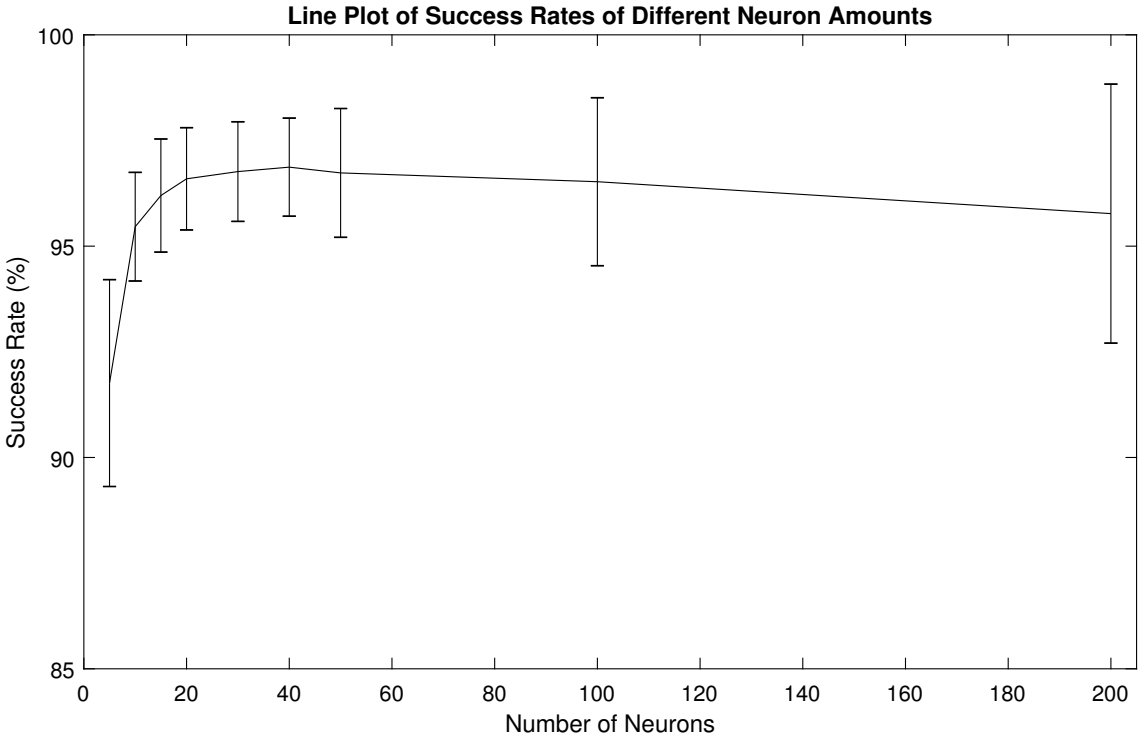

Figure B.1. *Plotting success rate percentages against the number of neurons in the single hidden layer of the direct test and optimization neural network. The values themselves and the error bars come from standard deviation values listed in Table B.1.*

C FILTER PLACEMENT DATA

Appendix C presents the obtained data in the filter tests that is not displayed directly during the discussion of the results of the optimization of the filter placements. Consequently, references to this Appendix are made mainly within Chapter 5.

C.1 Filter Placements with 10 and 50 Repeats

This section includes the results of the different filter amounts for the 10 and 50 repeats in the optimization neural network. Like in the results presented in Chapter 5, all of the tests use 500 repeats in the direct test neural network when verifying the success rate of the optimized locations and the FWHM of 0.05 microns. The locations obtained here are in relatively good agreement with each other as well as the locations presented within the main body of this work.

C.1.1 Five Filters

Table C.1. *Placements of five filters when the optimization neural network takes the mean after 10 repeats. Each filter location is represented by its central wavelength λ_n, where n is the filter number. Sets are listed in order of increasing optimization neural network success rate.*

ONN Repeats: 10, DNN Repeats: 500							
Set	λ_1 (μm)	λ_2 (μm)	λ_3 (μm)	λ_4 (μm)	λ_5 (μm)	ONN Rate (%)	DNN Rate (%)
1	0.700	1.075	1.325	2.075	2.450	94.086	93.202
2	0.450	0.700	0.950	1.200	2.325	96.500	95.709
3	0.450	0.700	1.075	1.325	2.200	96.550	95.516
4	0.450	0.700	1.075	1.450	2.013	96.809	96.082
5	0.450	0.700	1.012	1.450	1.950	96.941	96.150
6	0.459	0.747	0.950	1.067	2.450	96.950	95.955
7	0.575	0.700	1.013	1.200	2.200	97.055	95.693
8	0.451	0.762	1.075	1.450	2.013	97.246	96.011
9	0.513	0.700	1.075	1.356	1.888	97.450	96.525
10	0.544	0.700	1.028	1.450	1.958	97.700	96.870
Mean	0.504	0.748	1.058	1.402	2.145	96.729	95.771
SD	0.083	0.117	0.106	0.272	0.211	1.000	0.988

Table C.2. *Placements of five filters when the optimization neural network takes the mean after 50 repeats. For further specifications, see Table C.1.*

ONN Repeats: 50, DNN Repeats: 500							
Set	λ_1 (μm)	λ_2 (μm)	λ_3 (μm)	λ_4 (μm)	λ_5 (μm)	ONN Rate (%)	DNN Rate (%)
1	0.450	0.700	0.950	1.262	2.200	96.082	95.747
2	0.450	0.700	0.950	1.075	2.323	96.189	95.834
3	0.450	0.700	0.950	1.138	2.450	96.272	95.884
4	0.450	0.732	0.959	1.075	2.200	96.281	95.941
5	0.450	0.700	0.950	1.075	2.450	96.306	95.961
6	0.450	0.704	1.083	1.325	1.950	96.545	96.108
7	0.451	0.729	1.075	1.329	1.981	96.636	96.140
8	0.513	0.700	1.013	1.513	1.888	96.877	96.465
9	0.513	0.700	1.013	1.388	1.950	97.035	96.541
10	0.513	0.700	1.013	1.263	2.044	97.147	96.688
Mean	0.469	0.707	0.995	1.244	2.144	96.537	96.131
SD	0.030	0.013	0.052	0.151	0.212	0.375	0.325

C.1.2 Four Filters

Table C.3. *Placements of four filters when the optimization neural network takes the mean after 10 repeats. For further specifications, see Table C.1.*

ONN Repeats: 10, DNN Repeats: 500							
Set	λ_1 (µm)	λ_2 (µm)	λ_3 (µm)	λ_4 (µm)	λ_5 (µm)	ONN Rate (%)	DNN Rate (%)
1	0.723	0.919	1.075	2.450	-	93.600	92.992
2	0.731	0.950	1.076	2.450	-	93.632	92.956
3	0.450	0.700	0.950	2.450	-	94.014	93.053
4	0.450	0.765	0.919	2.138	-	94.236	93.479
5	0.567	0.763	0.919	2.450	-	94.668	93.927
6	0.450	0.762	0.919	2.388	-	94.755	93.933
7	0.452	0.763	0.903	2.450	-	94.782	94.031
8	0.450	0.765	0.919	2.419	-	94.855	94.022
9	0.450	0.763	0.919	2.450	-	94.905	94.170
10	0.513	0.732	0.919	2.450	-	95.032	94.029
Mean	0.524	0.788	0.952	2.409	-	94.448	93.659
SD	0.114	0.080	0.066	0.098	-	0.536	0.489

Table C.4. *Placements of four filters when the optimization neural network takes the mean after 50 repeats. For further specifications, see Table C.1.*

ONN Repeats: 50, DNN Repeats: 500							
Set	λ_1 (µm)	λ_2 (µm)	λ_3 (µm)	λ_4 (µm)	λ_5 (µm)	ONN Rate (%)	DNN Rate (%)
1	0.701	0.954	1.079	2.388	-	93.116	92.806
2	0.450	0.762	0.934	2.450	-	94.341	94.041
3	0.450	0.763	0.935	2.433	-	94.412	94.054
4	0.575	0.763	0.888	2.446	-	94.505	94.137
5	0.544	0.763	0.934	2.450	-	94.704	94.203
6	0.544	0.763	0.903	2.434	-	94.819	94.557
7	0.544	0.763	0.891	2.442	-	94.848	94.349
8	0.544	0.770	0.888	2.450	-	94.854	94.272
9	0.544	0.739	0.911	2.450	-	94.868	94.429
10	0.544	0.757	0.919	2.450	-	94.935	94.457
Mean	0.544	0.778	0.928	2.439	-	94.540	94.130
SD	0.069	0.062	0.056	0.019	-	0.542	0.497

C.1.3 Three Filters

Table C.5. *Placements of three filters when the optimization neural network takes the mean after 10 repeats. For further specifications, see Table C.1.*

ONN Repeats: 10, DNN Repeats: 500							
Set	λ_1 (µm)	λ_2 (µm)	λ_3 (µm)	λ_4 (µm)	λ_5 (µm)	ONN Rate (%)	DNN Rate (%)
1	0.794	0.950	1.513	-	-	89.664	88.910
2	0.790	0.919	1.513	-	-	89.882	89.134
3	0.778	0.917	1.576	-	-	90.159	89.481
4	0.873	0.891	2.388	-	-	90.409	81.017
5	0.888	1.013	1.200	-	-	90.505	89.209
6	0.763	0.922	2.200	-	-	90.736	90.094
7	0.763	0.919	2.419	-	-	90.741	90.185
8	0.763	0.919	2.411	-	-	90.786	90.214
9	0.763	1.013	1.169	-	-	91.046	89.920
10	0.825	1.013	1.169	-	-	91.214	90.225
Mean	0.789	0.947	1.756	-	-	90.514	88.839
SD	0.040	0.047	0.539	-	-	0.495	2.651

Table C.6. *Placements of three filters when the optimization neural network takes the mean after 50 repeats. For further specifications, see Table C.1.*

ONN Repeats: 50, DNN Repeats: 500							
Set	λ_1 (µm)	λ_2 (µm)	λ_3 (µm)	λ_4 (µm)	λ_5 (µm)	ONN Rate (%)	DNN Rate (%)
1	0.763	0.950	1.450	-	-	89.229	88.863
2	0.731	0.950	1.388	-	-	89.433	88.991
3	0.763	0.954	1.380	-	-	89.481	89.150
4	0.763	0.919	1.669	-	-	89.854	89.467
5	0.778	0.919	1.778	-	-	89.943	89.577
6	0.841	1.028	1.169	-	-	90.346	89.783
7	0.763	0.919	2.060	-	-	90.364	90.112
8	0.825	1.005	1.204	-	-	90.583	90.232
9	0.825	1.013	1.200	-	-	90.593	90.207
10	0.823	1.013	1.200	-	-	90.682	90.222
Mean	0.787	0.967	1.450	-	-	90.051	89.660
SD	0.038	0.044	0.298	-	-	0.537	0.532

C.2 Changed Initial Points

This section lists the results of tests that were made to ensure that the initial points determined in the optimization algorithm do not affect the end results. These tests are presented within this section for the different filter amounts, once again with 30 repeats in the optimization neural network and 500 repeats in the direct test neural network. The obtained results are in good agreement with those yielded by determining the initial points to be at the beginning of the range, increasing the confidence in the results presented within the main body of text.

Table C.7. *Placements of five filters when the initial points have been changed to the end of the range with 30 repeats in the optimization neural network. For further specifications, see Table C.1.*

ONN Repeats: 30, DNN Repeats: 500							
Set	λ_1 (µm)	λ_2 (µm)	λ_3 (µm)	λ_4 (µm)	λ_5 (µm)	ONN Rate (%)	DNN Rate (%)
1	0.696	1.075	1.325	2.083	2.446	93.767	93.203
2	0.450	0.700	0.950	1.216	2.200	96.335	95.779
3	0.450	0.700	1.075	1.325	1.700	96.397	95.785
4	0.450	0.700	1.075	1.450	2.075	96.438	95.853
5	0.450	0.735	0.966	1.122	2.450	96.465	95.993
6	0.450	0.700	1.075	1.388	1.888	96.741	96.110
7	0.450	0.700	1.013	1.200	2.075	96.756	96.155
8	0.544	0.763	0.981	1.263	2.325	97.099	96.540
9	0.543	0.700	1.013	1.341	2.263	97.329	96.774
10	0.551	0.700	1.044	1.326	2.013	97.461	96.957
Mean	0.503	0.747	1.052	1.371	2.143	96.479	95.915
SD	0.081	0.117	0.107	0.268	0.241	1.033	1.039

Table C.8. *Placements of four filters when the initial points have been changed to the end of the range with 30 repeats in the optimization neural network. For further specifications, see Table C.1.*

ONN Repeats: 30, DNN Repeats: 500							
Set	λ_1 (µm)	λ_2 (µm)	λ_3 (µm)	λ_4 (µm)	λ_5 (µm)	ONN Rate (%)	DNN Rate (%)
1	0.450	0.763	0.951	2.450	-	94.152	93.544
2	0.450	0.766	0.888	2.450	-	94.335	93.893
3	0.450	0.763	0.888	2.450	-	94.383	93.807
4	0.450	0.774	0.919	2.419	-	94.424	94.046
5	0.450	0.747	0.903	2.450	-	94.429	93.975
6	0.450	0.755	0.917	2.450	-	94.527	94.069
7	0.450	0.763	0.927	2.450	-	94.555	94.097
8	0.544	0.723	0.888	2.450	-	94.679	94.237
9	0.544	0.731	0.919	2.388	-	94.726	94.182
10	0.544	0.763	0.918	2.450	-	95.133	94.500
Mean	0.487	0.755	0.912	2.441	-	94.534	94.036
SD	0.045	0.016	0.021	0.021	-	0.269	0.257

Table C.9. *Placements of three filters when the initial points have been changed to the end of the range with 30 repeats in the optimization neural network. For further specifications, see Table C.1.*

ONN Repeats: 30, DNN Repeats: 500							
Set	λ_1 (µm)	λ_2 (µm)	λ_3 (µm)	λ_4 (µm)	λ_5 (µm)	ONN Rate (%)	DNN Rate (%)
1	0.778	0.950	1.388	-	-	89.453	89.097
2	0.794	0.950	1.513	-	-	89.497	89.814
3	0.778	0.919	1.602	-	-	90.064	89.592
4	0.770	0.919	1.669	-	-	90.076	89.490
5	0.763	0.919	2.013	-	-	90.405	89.983
6	0.783	0.888	2.442	-	-	90.588	90.155
7	0.778	1.013	1.200	-	-	90.620	90.091
8	0.825	1.013	1.153	-	-	90.659	90.061
9	0.825	1.013	1.153	-	-	90.776	90.201
10	0.825	1.013	1.171	-	-	90.852	90.152
Mean	0.792	0.960	1.530	-	-	90.299	89.764
SD	0.024	0.049	0.426	-	-	0.508	0.493

D FILTER PLACEMENT IMPORTANCES

Appendix D presents the importances of locations that correspond to those presented in the previous Appendix's Section C.1. There are three sections, one for each filter amount. The details of how importances are determined are presented in Chapter 5.

D.1 Importances for Five Filters

Table D.1. *Determination of the order of importance for the best succeeding set in the five filters and 10 repeats simulation described in Table C.1. The success rates are determined with the direct test neural network. Each filter location is represented by its central wavelength λ_n, where n is the filter number. The resulting order of importance is listed from least important to most important.*

ONN Repeats: 10, DNN Repeats: 500						
Set	λ_1 (µm)	λ_2 (µm)	λ_3 (µm)	λ_4 (µm)	λ_5 (µm)	DNN Rate (%)
1	0.544	0.700	1.028	1.450	1.958	96.819
2	-	0.700	1.028	1.450	1.958	92.663
3	0.544	-	1.028	1.450	1.958	90.347
4	0.544	0.700	-	1.450	1.958	84.637
5	0.544	0.700	1.028	-	1.958	88.900
6	0.544	0.700	1.028	1.450	-	88.642
Resulting Order of Importance: 0.544 < 0.700 < 1.450 < 1.958 < 1.028						

Table D.2. *Determination of the order of importance for the best succeeding set in the five filters and 50 repeats simulation described in Table C.2. For further specifications, see Table D.1.*

ONN Repeats: 50, DNN Repeats: 500						
Set	λ_1 (µm)	λ_2 (µ)	λ_3 (µm)	λ_4 (µm)	λ_5 (µm)	DNN Rate (%)
1	0.513	0.700	1.013	1.263	2.044	96.643
2	-	0.700	1.013	1.263	2.044	92.801
3	0.513	-	1.013	1.263	2.044	89.701
4	0.513	0.700	-	1.263	2.044	85.104
5	0.513	0.700	1.013	-	2.044	88.737
6	0.513	0.700	1.013	1.263	-	92.079
Resulting Order of Importance: 0.513 < 2.044 < 0.700 < 1.263 < 1.013						

D.2 Importances for Four Filters

Table D.3. *Determination of the order of importance for the best succeeding set in the four filters and 10 repeats simulation described in Table C.3. For further specifications, see Table D.1.*

ONN Repeats: 10, DNN Repeats: 500						
Set	λ_1 (µm)	λ_2 (µm)	λ_3 (µm)	λ_4 (µm)	λ_5 (µm)	DNN Rate (%)
1	0.450	0.763	0.919	2.450	-	94.055
2	-	0.763	0.919	2.450	-	90.298
3	0.450	-	0.919	2.450	-	86.723
4	0.450	0.763	-	2.450	-	73.679
5	0.450	0.763	0.919	-	-	90.474
Resulting Order of Importance: 2.450 < 0.450 < 0.763 < 0.919						

Table D.4. *Determination of the order of importance for the best succeeding set in the four filters and 50 repeats simulation described in Table C.4. For further specifications, see Table D.1.*

ONN Repeats: 50, DNN Repeats: 500						
Set	λ_1 (µm)	λ_2 (µm)	λ_3 (µm)	λ_4 (µm)	λ_5 (µm)	DNN Rate (%)
1	0.544	0.763	0.903	2.434	-	94.406
2	-	0.763	0.903	2.434	-	90.157
3	0.544	-	0.903	2.434	-	84.889
4	0.544	0.763	-	2.434	-	74.459
5	0.544	0.763	0.903	-	-	90.142
Resulting Order of Importance: 0.544 < 2.434 < 0.763 < 0.903						

D.3 Importances for Three Filters

Table D.5. *Determination of the order of importance for the best succeeding set in the three filters and 10 repeats simulation described in Table C.5. For further specifications, see Table D.1.*

ONN Repeats: 10, DNN Repeats: 500						
Set	λ_1 (µm)	λ_2 (µm)	λ_3 (µm)	λ_4 (µm)	λ_5 (µm)	DNN Rate (%)
1	0.825	1.013	1.169	-	-	90.244
2	-	1.013	1.169	-	-	73.917
3	0.825	-	1.169	-	-	66.746
4	0.825	1.013	-	-	-	71.031
Resulting Order of Importance: 0.825 < 1.169 < 1.013						

Table D.6. *Determination of the order of importance for the best succeeding set in the three filters and 50 repeats simulation described in Table C.6. For further specifications, see Table D.1.*

ONN Repeats: 50, DNN Repeats: 500						
Set	λ_1 (µm)	λ_2 (µm)	λ_3 (µm)	λ_4 (µm)	λ_5 (µm)	DNN Rate (%)
1	0.825	1.005	1.204	-	-	90.271
2	-	1.005	1.204	-	-	73.797
3	0.825	-	1.204	-	-	64.790
4	0.825	1.005	-	-	-	70.063
Resulting order of importance: 0.825 < 1.204 < 1.005						