Applying Machine learning techniques to high contrast images and medium resolution spectra of warm Jupiters.

Presented by

RAKESH NATH RANGA

PhD thesis in SPACE SCIENCES

Université de Liège
STAR Institute, University of Liège, 19 Allée du Six Août, 4000 Liège, Belgium

October 2023

OLIVIER ABSIL	Promoteur
MARC VANDENBROECK	Co-promoteur

Contents

List of Figures

List of Tables

Part I

Machine learning based detection of exoplanets using high contrast image spectra

ABSTRACT

Extracted spectra are an increasing part of exoplanet science, particularly with the advent of the James Webb Space telescope. The scientific goals of such spectra range from molecular characterization, molecular species detection to constraining abundances in exoplanet data. In direct imaging the availability of spectra in high contrast images has allowed astronomers to detect exoplanets and characterize them. With increasing data sizes and the availability of high resolution direct imaging spectra from high resolution spectrographs, there are both computational and accuracy limitations brought by the traditional techniques used to analyze these spectra. The advent of advanced data science algorithms particularly machine learning algorithms have been suggested as an alternative to traditional spectral processing. This part is dedicated to studying the effect of ML algorithms when used directly on spectra extracted from direct imaging data. To that end we have two introductory chapters; in the first one we will first introduce data processing methods for direct imaging spectra and in the second we will then introduce ML algorithms and specifically the classes of algorithms used in this chapter. This is followed by the data chapter in which we will define the science objectives and the data that is used to meet these science objectives. This is followed by two methods chapters; the first of which describes the cross correlation based algorithm and the results it produces on the benchmarking scale and its scientific impact, The second methods chapter describes the ML algorithms and their results The final chapter in this part will discuss the results and derive a conclusion from both the experiments and finally define what next step this piece of research has motivated us to do.

Chapter 1

Introduction to spectral data processing

Spectral data was used in the previous Part by extracting the spectra from SADI data cubes. We used the pixels as samples of spectra that are then cross-correlated with template spectra. In order to make detections we had to define a SNR value by using the cross correlations values generated by cross correlating spectra extracted from different pixels. In the previous part we explored spectra as extractions from individual pixels. We cross correlated these spectra with template spectra and then reconstructed the maps with these replaced spectra. We explored different spatial methods to make these spectra noise free and amenable to data processing. The goal of the previous part was to understand whether spectral processing of spaxels can produce similar or different effects when taking into account the spectral features.

In this part, we will explore the detectability and characterization of exoplanets through spectra alone without considering the image dimension. The goal of this part is to study the detection and characterization of exoplanets whose spectra have been extracted from the spaxels. In the absence of a spatial dimension, we lack the ability to produce SNR maps as in the previous part, we also miss the distinctive 2D Gaussian shape that indicates the PSF of the telescope on the exoplanet. Therefore, the broad scientific goal of this part is to understand how much is detectability and characterization of exoplanets is affected by the absence of these key features. This part will explore the efficacy of ML algorithms of detecting exoplanets and whether they can prove a viable alternative to cross correlation based detection techniques, and whether they can continue to leverage the simultaneous detection and characterization advantage that spectra lend to.

We will describe in this chapter the scientific contributions that spectra alone have produced to the field of exoplanet science, we will then describe briefly the different types of algorithms that have used spectra to make this science possible. We will then situate the importance of our research in this context of algorithms by describing the limitations of already present algorithms. We will then describe the various ML algorithms that are currently in use in Astronomy related to spectra. We will then describe why we are motivated to use ML algorithms and which ones seem to be best suited for this job of identifying spectral features in the data.

1.1 Exoplanetary spectral data from direct imaging data sets

As described in the previous part, spectral data are produced using IFS enabled imaging instruments such as (SINFONI, Bonnet et al., 2004) and (GPI, Macintosh et al., 2014). The disadvantage of having images with spectra is that it leads to a large number of spectra with just noise and just few spaxels with the exoplanet. This sometimes leads to false positive detections in discovery campaigns. In addition, exoplanets are very faint to be easily visible in images, particularly when the image is noisy. However, it is also possible to not have an explicit spatial dimension, but still have extracted spectra through an IFS enabled instrument. This allows us to have fewer but more targeted spectra with just one robust noise spectrum, the standard deviation of which can be used to estimate the noise in the observations. For the purposes of this introductary chapter, we will refer to such spectra as high contrast diffusion spectra (HDC).

Examples of such instruments include (KPIC, Jovanovic et al., 2019) and (Jocou et al., 2022, HARMONI,). KPIC is attached to the Keck telescope and is designed to have $\approx 50\%$ starlight suppression and is designed to detect faint exoplanets which are not visible in the image. Typically, the best way to reduce stellar contamination is to image in wavelength where the stellar flux is low as compared to planetary emissions, the choice of wavelength thus falls typically in the near IR (between 1.4 μm and 1.9 μm that corresponds to the H-band and between 2.0 to 2.4 μ m that corresponds to the K- band. Therefore, these instruments operate in H and/or the K bands. Typically, KPIC as an example, operates in a domain called high dipsersion coronagraphy which combines high contrast imaging and the medium to high resolution spectroscopy. The high contrast portion imaging is achieved by feeding the light from the CCD in the telescope through single mode fibers placed at the exact pixel where the planet is present. The light is then dispersed through medium to high resolution spectrographs with a fixed resolution (R) between $10,000-100,000$. Typically, the R is defined for the lowest wavelength λ_{min} as,

$$R = \frac{\lambda_{\text{min}}}{\Delta\lambda} \tag{1.1}$$

where $\Delta\lambda = \lambda_2 - \lambda_1$ for consecutive wavelength values λ_2 and λ_1. Laboratory studies have shown, for (e.g Calvin et al., 2021), that coupling the single mode fiber to the CCD pixel appropriately produces the best signal to noise (SNR). This SNR, given the context of using just the spectra defines the 'quality' of the spectrum that can be used for scientific processing.

Thus, we have direct imaging that observe a star system where a potential exoplanet is present, the light is gathered on a CCD chip which is then coupled with a single mode fiber to achieve a desired SNR and then dispersed through a spectrograph with a fixed R to produce spectra of potential exoplanets. This spectrum will then be processed similar to what we did in the previous part. However, unlike in the previous part where we use real data, in this part we will simulate this behaviour without any biases on the efficiency of coupling or the fixed R. In order to achieve this we will use synthetic data that is described in Chapter 3.

1.2 Exoplanetary science from HDS

Spectra from direct imaging instruments have been useful in different ways in the field of exoplanet science. Spectra from direct imaging are obtained, usually, through the means integral field spectrographs (IFS) and are typically mounted on larger telescopes that gather photons from the target e.g (e.g Bonnet et al., 2004; Beuzit et al., 2019) or single mode fiber based instruments such as (e.g Jovanovic et al., 2019). This results in gathering of photons in specific wavelength bands that are reflected off the surface of the exoplanet and therefore has the ability to probe interesting surface physics such as temperature, cloud profile etc, when they are appropriately retrieved (e.g Batalha et al., 2019).

There are broadly two ways that these spectra can be used for a) detecting exoplanets based on specific molecules present in the spectrum or b) characterizing exoplanets by identifying the concentration of molecules present in the spectrum. While detection is a fairly well defined term, meaning to be able to sensitively assert that an exoplanet is present based on spectral features alone, whereas characterization is a broader term that covers a broad swathe of definitions. The simplest form of characterizing the exoplanet is to constrain the $\mathrm{T_{eff}}$ and $\log(g)$ of the exoplanet. In this case the parameters are constrained with a clearly defined error bar for both. This is usually the first step that follows any detection and spectra are the only data dimension that we have to perform this step. Usually, we don't need even the spectral absorption lines to estimate $\mathrm{T_{eff}}$ and $\log(g)$ as shown by (e.g Cugno et al., 2023) for PDS70b, they are estimated using continuum modelling or using SED estimation as in the case the first time it was discovered by Mesa et al. (2019). Typically, this is the go-to approach when constraining $\mathrm{T_{eff}}$ and $\log(g)$ is to use the continuum of the exoplanet and derive these with SED, but this only works for exoplanets that are far out from the stellar glare that will be easily characterized. On the other hand for close in companions such as (HD142527b, Christiaens et al., 2018), the continuum is fully dominated by the stellar glow and therefore the molecules present in the atmosphere alone are the indications to the $\mathrm{T_{eff}}$ and $\log(g)$.

Characterization can also be done in terms of molecular abundances Wang et al. (2023) on the surface of the planets, constraining metallicity and surface gravity of the exoplanets Aleman et al. (2023). These type of characterizations are particularly interesting because they give us the following scientific details,

- **Constraining the ages of exoplanet systems:** Molecular abundance ratios are crucial in determining how old an exoplanet and thereby its system is which allows us to lay constraints on its theory formation. The C/O ratio is one of the most common ratios that are inferred and van der Marel et al. (2021) shows that this ratio can be used to constrain the transport rates of icy pebbles within the protoplanetary disks,

- **Constraining the composition of an exoplanet:** the composition of an exoplanet is one of the key attributes that makes exoplanetology truly alluring, according to Madhusudhan (2019) this sort of characterization has resulted in the necessity for new instruments which has in turn lead to a state of 'competitive exoplanetology'. Nevertheless, the abundance ratios, the depth at which these ratios are recovered are the only means that we know currently to infer the composition of an exoplanet.

- **Constraining dynamic processes that define exoplanetary evolution:** molecular species opacities allow astronomers to lay constraints on the dynamic processes within the planetary atmosphere for (Phillips et al., 2020, e.g Brown dwarfs)

In general,characterization using molecular abundances allow us to constrain both the age and rule out potential kinds of exoplanets (e.g Christiaens et al., 2021).

For these kind of studies in addition to observed spectra from planetary systems, we require accurate theoretical models of what planetary spectra look like at different resolutions. It is prudent to model these spectra in the infra red starting for 0.9 μm up to 7 μm where the ratio of planetary flux to stellar flux is higher. A number of detailed spectral models exist starting with high resolution molecular models produced in lab settings such as ExoMol Tennyson and Yurchenko (2012), detailed atmospheric models that are amenable to spectral retrieval such as PetitRadTrans Mollière et al. (2022), specialized cloud and atmosphere models that are suited to studying exoplanetary atmospheres such as PICASO3 Mukherjee et al. (2023) and generalized absorption spectra that are more suited to constraining specific properties of the exoplanet such as $\mathrm{T_{eff}}$ and $\log(\mathrm{g})$, which is what we use in this project named BT-SETTL Allard et al. (1997).

While retrieval is notoriously difficult in practice, it is possible to use theoretical spectra and model specific molecules on the surface of the potential exoplanet and thereby characterize them based on mass and temperature. In order to narrow the scientific scope and to limit the complication of scientific inferences, this thesis will primarily limit itself to characterizing the $\mathrm{T_{eff}}$ and $\log(g)$. This also allows this thesis, to explore the effectiveness of the characterization of exoplanets by using the accuracy of characterization of $\mathrm{T_{eff}}$ and $\log(g)$ as it uses the same features. These features are also the same molecular absorption lines that will allow us for instance to constrain specific molecular abundances. Hence, we use these two parameters as proxies to define the ability of an algorithm to characterize exoplanets.

1.3 Algorithms that detect and/or characterize exoplanets using high contrast diffusion spectra

This thesis will constantly refer to algorithms in two broad categories based on how they produce results from the spectra. These algorithms all either use non-ML based techniques such as cross correlations, forward modelling etc or they use ML based techniques that infer results using ML algorithms that have either been trained with such data or are somehow very specific to using spectra.

Algorithms that use spectra typically work on specific characterizations of a detected exoplanet. Atmospheric characterizations are performed by either retrieval algorithms (e.g Lavie et al., 2017) or by forward modelling (e.g Palma-Bifani et al., 2023) the extracted spectra to derive specific atmospheric properties such as carbon to oxygen C/O ratios in the atmosphere of the exoplanet. Broader characterizations of the exoplanet could be limited to constraining the $\mathrm{T_{eff}}$ and $\log(g)$ with errorbars using log-likelihood ratios obtained through forward modelling (e.g Ruffio et al., 2019) particularly in cases where spectral lines themselves are not well resolved. In this section, we will discuss algorithms that have one processing criteria in common, they take as input raw spectra from instruments such as KPIC and produce scientifically interpretable and publishable results. In their turn, these algorithms have been benchmarked for their performance. In this section we will present a brief summary of the type of algorithms and their science results.

1.3.1 Non-ML based spectral inference algorithms

Post processing for detection and/or characterization of high contrast spectra is different from other spectral post-processing (for example other data acquisition methods such as transit photometry) because of the pecularities unique to high contrast imaging. The first step, though this thesis does not focus on this step, would be to calibrate the extracted spectra from an instrument. This calibration step usually involves solving for the wavelength solution so that the resulting vector from the instrument is mapped to its equivalent wavelength values such that each photon value in the vector corresponds to a wavelength vector. Once this is done, a major problem is the continuous presence of stellar contamination in terms of the stellar continuum in most spectra, particularly those spectra of exoplanets that are at smaller separations from the host star. Thus, processing of spectra requires the equivalent of the PSF subtraction performed for ADI images. This has to be followed, by some way of comparing known exoplanet models to the spectrum of the candidate exoplanet in such a way that any residual stellar contamination does not significantly impact this comparison. Finally, the results of this comparison have to be interpreted so that it has scientific relevance.

Stellar contamination subtraction methods:

The most used stellar contamination subtraction methods is spectral differential imaging (SDI, Sparks and Ford, 2002) which accurately measures the stellar spectrum and subtracts from the data leading to relatively stellar contamination free spectra. SDI creates several high frequency artefacts which are somewhat resolve with the use of high resolution spectral differential imaging (HRSDI, Haffert et al., 2019) and modified HRSDI (mHRSDI, Xie et al., 2020). For spectra in HDS domain, we have continued the same sort reference spectra creation and the adoption of removal of low frequency artifacts. Usually, this leaves behind mis-subtracted residuals, these are known as speckles in the spatial domain, but there is known equivalent or study in the spectral domain. In the spatial domain the speckles are typically tackled using (PCA based algorithms, Xie et al., 2020; Hunziker et al., 2018) where after performing procedures such as SDI, further principal components are computed from the image dimension and then subtracted to remove the unsubtracted residuals. There has also been a few attempts to combine the advantages of having both image and spectral data by combining angular differential imaging (ADI) and spectral differential imaging techniques in an optimal manner Kiefer et al. (2021). This is akin to the analysis we have performed in Part II. Principal component analysis, typically applied to the image domain Amara and Quanz (2012), have been adapted a combination of both spatial and spectral PCA. Thus, while we are able to perform basic subtractions there is no comprehensive subtraction of stellar contamination.

modelling the residual spectral features:

Once the stellar contamination subtraction step is completed, algorithms typically now transform the data making it amenable to interpretation and statistical analysis. There are two broad ways that spectra are processed to characterize them, one would be the so called forward modelling where an analytic function is fit to the data along with a model of noise and the best fit is chosen as the exoplanet model. An example of such modelling is (BREADS, Agrawal et al., 2023) which uses the standard forward modelling principles set out in Ruffio et al. (2019) which has been since then in the detection of water and carbon monoxide in HR8799 (Ruffio et al., 2021). Detailed modeling of the spectral shapes has allowed us to identify orbital parameters as well Wang et al. (2022) where accurate retrieval techniques can be used to constrain the carbon to oxygen ratios as well. In the realm of medium resolution spectroscopy, it is still possible to limit the science question to if specific molecules such as $\mathrm{H_2O}$ and CO are still detectable and this is performed with radial velocity searches through template cross-correlations (e.g Ruffio et al., 2021).

The other type of processing is using cross correlations of templates and the spectra itself as we did in Part I. This type of processing has been particularly popular when detecting specific molecules using (for e.g Molecule maps Hoeijmakers et al., 2018). Cross correlations have an advantage because of their simplicity and ease of interpretations. On the other hand the cross correlation coefficients themselves are noisy and the most consistent interpretations need several noise realizations. These do not easily lend to HDS data, in this thesis we use the formalism established by Ruffio et al. (2019) to define cross correlation noise.

interpreting the results:

Both forward modelling and cross correlations result in produces either mock spectral models or cross correlation coefficients that are subject to interpretation. This interpretation is typically made in the context of statistical SNR which is computed as a ratio of the signal cross correlation coefficients and standard deviation of the noise correlations. For the forward models the model spectrum is cross correlated with model spectrum produced by the forward model whereas it is also possible for instance to directly cross correlate the spectrum with a template directly.

The SNR is then computed for several molecules and thus it is possible to predict if a particular molecule is present if the signal is 5σ over the noise i.e $\mathrm{SNR} \geq 5$. Once the presence of an exoplanet molecules are detected, characterization consists of a few more steps in order to derive properties such as abundance ratios, orbital parameters etc. Typically calculating abundance ratios involves 'retrieving' the spectra from a set of atmospheric models as in (for e.g deriving the C/H ratios Xuan et al., 2022).

1.3.2 Use of machine learning (ML) algorithms in analyzing astrophysical data and spectra

Machine learning in the current age of big data has produced algorithms that are capable of analyzing large amounts of data with remarkably small processing times. They have also proven to be particularly using unsupervised clustering and data mining (e.g Baron, 2019; Ball and Brunner, 2010; Ivezić et al., 2014). They application of ML algorithms have been particularly of note in classifying stellar spectra (e.g Miettinen, 2018; Naul et al., 2018). The analysis of stellar spectra using AstroNN has shown promise even for high resolution spectra Leung and Bovy (2019). Application of artificial neural networks and particularly deep neural networks to exoplanet detection is now regarded as an established method in astrophysics Fluke and Jacobs (2020). The exoplanet community has benefited from the use of deep neural networks for Kepler light curves Pearson et al. (2018), for direct imaging detection Gomez Gonzalez et al. (2018) and to model the PSF model of an instrument Gebhard et al. (2022). ML algorithms such as PCA have shown that they can be quite well trusted to model the noise in data Gomez Gonzalez et al. (2016) and to model the PSF of the instrument which results in high fidelity subtraction of the stellar components Meshkat et al. (2014). In addition, deep learning algorithms such as SODINN Gomez Gonzalez et al. (2018) have demonstrated the ability to detect high contrast companions with fewer false positives. This has motivated their use in new missions and surveys such as the Large interferometer for exoplanets (LIFE) Angerhausen and Quanz (2021). In this context, ML algorithms were considered to be of value to the research question that this chapter addresses. The ML algorithms attempted in direct imaging addresss either the question of PSF subtraction (e.g Gebhard et al., 2022) or try to use the spatial noise variance to discriminate between pixels that contain noise and those that contain the exoplanet (Gomez Gonzalez et al., 2018, e.g). These methods do not take into account the spectral absorption lines that uniquely identify exoplanetary absorption, which in turn can be considered exoplanetary signatures. Some attempts with using spectral features were not particularly successful (e.g Fisher et al., 2020) at learning these spectral features unlike Li et al. (2017) does for stellar spectra. However, at this point there still remains the question of why it is necessary to explore the use of ML with spectral data.

1.4 Limitations of current methods

Molecule maps (Hoeijmakers et al., 2018) are considered suitable for detecting species of molecules of warm Jupiters or cool M-dwarfs Mâlin et al. (2023) and has been used to further characterize the existence of certain molecules in well known companions (e.g Petit dit de la Roche et al., 2018). One of the greatest limitations with applying techniques such as molecule maps is that they are applied when the existence of an exoplanet is well known and though they lend to astrometric characterization of molecules, it does not allow a truly 'blind' search to commence. In other words, while molecule maps detect the presence of molecules they do not detect the companions themselves, or at any rate they are not designed for a blind search for companions. In the previous part we redetected HD142527b and produced SNR maps by taking into account all the molecules present in the spectrum.

In the previous part we also demonstrated that it is possible to produce SNR maps of such extracted spaxels and thereby detect a companion and that such a detected companion can also be characterized through the means of characterization matrix that we demonstrated for HD142527b. Molecule maps and the cross-correlation maps constructed in the previous chapter, rely on some knowledge of spatial noise diversity and the spectral resolution of the spectra play a limited role in detection of the exoplanets. High resolution spectra can be quite advantageous particularly with forward modeling the spectra (Ruffio et al., 2021; Wang et al., 2021, e.g) where spatial data is not available. However, forward modelling is still a method to *choose* a template that would be cross correlated with the sample exoplanet spectrum. This brings into focus the importance of the cross correlation as the main engine of comparing spectra and identifying when a template matches the target spectrum. While there are several algorithms that perform cross correlations and template matching, a unified framework that allows an astronomer to detect and characterize the exoplanet based on the spectrum is still missing. Existing algorithms are also fine tuned to work with non-ML inference algorithms but a comprehensive study into how amenable these processing steps are with ML algorithms is missing.

Data sizes and the number of spectra available are also on the increase with the advent of James Webb, therefore there needs to be a scalable, accurate and somewhat model independent way to use these high resolution spectra to detect exoplanets in direct imaging. Computational speeds also are problematic when a large number of forward models need to be run to rule out various hypotheses. Therefore, the need of the hour is to be able to design an algorithm, which learns specific features which discriminate correctly between the spectra of an exoplanet and those of a star and those of noise. We also need an algorithm that will be perform similarly for both high and low contrast exoplanets and whose computational complexity does not scale out of bounds for higher resolution spectra. In this context, ML algorithms are appropriate to be tested with the current suite of inference algorithms already present in the literature. The question however still remains in terms of why we expect ML algorithms to be effective for this problem.

1.5 Motivation to use ML algorithms for identifying spectral features of exoplanets

In the previous part we discovered that detection of exoplanets using spectral data can be done through cross correlation primarily because of the presence of absorption lines in H and K bands. This was particularly visible when we cross correlated with parts of the spectrum and saw the differential SNR. This clearly indicates that data features are not uniform, but are present in different wavelength ranges in the spectrum, this is also the feature set which is utilized for (for e.g Leung and Bovy, 2019) to classify stellar spectra.

However, how effective is an ML algorithm when compared to a standardized cross-correlation algorithm that is described in the previous chapte? However, is there any relative benefit of using ML algorithms in place of a standard cross correlation and SNR map based detection regime? Consequently, does this have characterization benefits that is naturally produced by the cross correlations? What are the ML algorithms that are best suited to this form of analysis and would these ‘best in class’ algorithms provide any advantage to exoplanet detection? The motivation of this experiment is thus two fold, to begin with we need to define the variables about which we will quantify the effectiveness of an ML algorithm. The second objective is to verify whether the performance of a well trained ML algorithm offers an advantage to cross-correlation for the same data.

Thus we have organized this part to describe the data that is used to make the comparison, the algorithms used and then finally our conclusion. This chapter will describe the methods we use to produce data that can be used to test these questions. We will then describe the control of thse data with the use of parameter ‘knobs’ that allow us to calibrate the effectiveness of the algorithms that we are comparing against. We will then describe the algorithms that were used during these experiments to compare against the cross correlation of template spectra. We will then describe the metrics used to compute the ‘effectiveness’ of an algorithm. This will the be followed up with describing the results produced by cross correlation and producing thereby the benchmark that is set for the ML algorithms to achieve in order to be called at least ‘as effective’ as ML algorithms. We will describe these benchmarks for both detection and characterization. We will then describe results obtained by cross correlation when the ‘knobs’ are at different levels, for both detection and characterization. We will then describe the preliminary results for the ML algorithms that will inform us why ML:algorithms cannot be compared with cross correlations. We will then discuss the reasons for why ML algorithms are not suitable to replace cross correlations in this fashion. Finally, we will conclude this chapter with a discussion on how we could potentially leverage the power of ML algorithms to improve detectability limits, and leverage what we have learnt from this chapter.

Chapter 2

Introduction to machine learning algorithms

Machine learning (ML) algorithms are considered a class of algorithms that learn intrinsic relationships in the data in order to predict specified outputs that depend on these intrinsic relationships in the data. These relationships between specified outputs and the data can be learned in one of three well known ways,

1. **Unsupervised learning:** where the ML algorithm exploits the intrinsic relationships within data to learn features and use such features to derive the desired output without any external interventions. Algorithms in this class are typically used with data where intrinsic relationships in the data are not well established a priory. For example, unsupervised K nearest neighbors algorithms are frequently used to identify clusters within the data that might indicate data grouping which is not easily understood such as looking for data that could be analyzed for transients (e.g Aleo et al., 2022), or finding cluster groups in open cluster data (e.g Deb et al., 2022). Within Astronomy principal component analysis is a very well known unsupervised ML algorithm typically used in identifying structures in the data that can be discarded. Unsupervised ML algorithms are particularly useful for instance in anomaly detection Stinco et al. (2023). Selecting parameters in the data which will enable other ML algorithms to perform better on the data can also be achieved through unsupervised methods Huang et al. (2023). Principal component analysis is particularly useful in detecting such useful parameters in the data and has been proposed for instance to detect oceans in exoplanet data (e.g Ryan and Robinson, 2022). Within the field of high contrast imaging unsupervised ML algorithms have been used in the package (PACO, Chomez et al., 2023) and (VIP, Christiaens et al., 2023) also provides this capability.

2. **Supervised learning:** where the algorithm is shown multiple combinations of relationships between desired outputs for a set of inputs over multiple times and thus 'trained' to learn the relationship between the input and the output. Algorithms in this class are typically used when there is a large amount of labelled data and typ-

ically when the relationship between the desired output and input is well known. This class of algorithms are also very effective when the desired output represents a non-linear combination of inputs for instance inferring for instance that a picture contains a cat from the presence of whiskers, eyes, nose etc in the picture. A famous example of such supervised learning was the handwritten digit recognition where a ML algorithm is trained to recognize handwritten digits in the US postal system and has now been improved on with the latest advances in supervised learning Kussul and Baidyk (2004). This class comprises of a large set of algorithms spanning from (neural networks, Gurney, 1997) to (random forests, Breiman, 2001) to modern day (transformers, Vaswani et al., 2017). The advantage of supervised algorithms is that they are able to learn many complex relationships in the data. However, the intrinsic problems of this class is that it requires a large amount of data, where this is relationship is well known. Supervised learning schemes suffer from the significant problem of learning the unintended relationships in the data when it is not trained appropriately or it is tested with wrong data. In the same example as before, if the algorithm is trained to recognize only cats and we present a dog, it will indeed mis-identify a dog as a cat because of the similarity of the features. It is also possible that while learning the possible relationships between the images and the presence of a cat, the algorithm learns that the any indoor setting is a picture of a cat because of our data contained only indoor cats.

In this thesis, we focused only on supervised learning for the following reasons,

- the goal of the my thesis is to derive the presence of an exoplanet given certain discriminatory conditions in my input such as the spectral features corresponding to the planet. Such ‘conditions’ are typically called features in ML algorithm training.

- a good prior knowledge of such a relationship exists (and is proven using non-ML algorithms such as cross correlation) and a large number of similar datasets exist for me to train a supervised algorithm,

- and finally, supervised algorithms allow us to specify the kind of output we need in astrophysics rather than relying on pre-existing relationships within data.

The act of producing data pairs of input and desired output is known as ‘labelling’ and is typically carried out before training or testing any algorithms. The desired outputs which are the result of a specific input combination are respectively called the ‘label’ and the ‘data’. The data, thus, consists of multiple intrinsic parameters which as stated before are called features and the ML algorithm will typically learn which of these features are important for the desired output. In our case the desired output is whether an extracted spectrum is that of a planet and the features that we use to arrive at this result is the spectral absorption bands.

2.1 Types of supervised algorithms

Supervised learning can be further categorized based on the type of desired output as a regression and classification problems.

Regression: When the output of the supervised learning algorithm is a floating point number such as the distance or the intensity then such problems are called regression problems. Regression problems are usually adopted when the ML algorithm is meant to produce an unbounded output (except by the intrinsic rules set by the case itself for instance distance cannot be negative). This is also useful when we don't know precisely what the expected output is for a specific input, for instance distance between two points in an image can have a infinite values and this changes from each input to another. Within regression problems, we also have types of regression based on the bounds of the output parameter. For instance if the bounds are between 0 and 1, such regression is known as logistic regression. The fundamental limitation of the regression problem is that there is an intrinsic uncertainty in the result. If sufficient information is presented then this uncertainty can be constrained to be within a few percent. However, in some cases the output itself is constrained to be within a few finite values. Such class of problems are called classification problems. Astronomy has many regression problems where for example galaxy photometric parameters have been estimated using machine learning algorithms Yin et al. (2022). Time series forecasting in astronomy has also been framed as regression problem **?**.

Classification: Classification problems are where the output classes are already well known, for example when the input is a set of words the machine learning algorithms classifies these words in to the emotions they are associated with such as happiness, anger etc. Classifiers have been also used to predict for instance liver lesions Prakash et al. (2023) in medical imaging. In astronomy classifiers have been used to differentiate between stars, blazars and quasars Zhao et al. (2023) in WISE data. They have also been famously used to morphologically classify galaxies in the SDSS DR17 Fischer et al. (2019). Classifiers in exoplanet direct imaging have been used by (SODINN, Gomez Gonzalez et al., 2018) and (NA-SODINN Cantero et al., 2023). While classification has the adavantage of knowing the precise set of values that needs to be predicted, it also has the problem of the classes themselves being categorical. For instance, (SODINN, Gomez Gonzalez et al., 2018) wants the class of C^{+}, which is not a numerical value. Typical classifiers work with either neural networks or ensemble classifiers such as (Random Forests, Breiman, 2001) that produce floating point outputs. In order to convert these numerical values we do the following two step process,

1. For an n class classification We first create an $n-1$ sized array. Therefore, a 10 class classifier will output an 9 element array,

2. we then encode the classes using routines such as (One hot encoding Harris and Harris, 2010) that allows to convert a categorical class into a encoded value, where each position in the array is encoded as a 1 for its corresponding class. The case where all array elements are 0 also corresponds to a class, thereby covering all classes

This encoded data is now considered as the value that the ML algorithm needs to predict.

Thus the ML algorithm will predict a value for each position in this array, typically values will range between 0 and 1 for each position following the sigmoid function. Each value is thus a probability of the class and thus the class that the algorithm predicts would be the position with the highest probability. We also at time provide a threshold above which the highest probability is accepted and this becomes the decision boundary of the classifier.

While classifiers are some of the most widely used type of supervised algorithms, they also suffer from the issue where some classes are better represented than the others. They also require large number of labelled examples. These examples have to provide adequate representation of each of the classes, the noise distribution of the noise in each of these class samples should also mimic realistic noise in the data and finally, the samples themselves have to be representative of the data that the algorithm is expected to realistically classify. In order to meet these requirements, we adopt training methodologies to adequately ‘train’ an algorithm and then adopt specific testing methodologies to realistically estimate the performance of such algorithms. These are common for all supervised algorithms, sometimes the testing methodology can still be adopted for unsupervised algorithms as well. In this thesis when we say ML algorithms it is taken to mean supervised ML algorithms.

2.2 Training and testing supervised algorithms

When generating labels for supervised data there is an intrinsic relationship between the data and its corresponding label. This relationship could be either linear or non-linear and can thus be mapped to generate the output y from input data x as,

$$y = f(x) \tag{2.1}$$

where $f(x)$ represents a function mapping x to y A special case of this linear relationship would be a straight line where y and x are related by its slope m and interecept c,

$$y = mx + c \tag{2.2}$$

ML algorithms have been trained in various contexts to derive such relationships between x and y based on large amount of data. Such a process by which the ML algorithm is tuned to mimic $f(.)$ is called ‘training’.

Training requires sufficient number of data points that correspond to the expected output y. The function $f(.)$ is represented by the parameters of the ML algorithm which allow it to produce y given x. These parameters differ based on the ML algorithm in question, their tuning will be explained in detail in the section corresponding to those class of algorithms. In order to train an algorithms, we must first prepare the data so that this training is adequate, but not so targeted that the trained ‘model’ can no longer generalize to new data. The data usually has two characteristic elements, the data or x which consists of images or spectra and the desired result or target output y which is in our case whether the input corresponds to that of a warm Jupiter or not. In this case it is important that ML algorithm learns the generalized features in the data corresponding to the presence of a warm Jupiter but avoids learning noise structures present only along with the warm Jupiter which would be a systematic of data acquisition. Typically, this is solved with acquiring data with enough different types of systematics which would convince our model to learn only the common features, since this is not possible to achieve in practise we try to generate large amounts of data and try to evaluate how generalized the model is. In order to ensure this we first split the data into three parts

1. **Training part**: Usually this comprises of 80% of the data present and is the main engine to train the ML algorithms. The training is usually carried out by passing the x through the ML algorithm and 'comparing' the output of the algorithm to the true label. Usually this comparison is carried out using loss functions such as (cross entropy loss, Zhang et al., 1990). This loss is then minimized over multiple examples by tuning the algorithm until the same error on a part of the dataset not used for training reaches a low minimum value

2. **Validation part**: This is the part of the dataset where the loss is tracked until it reaches a low minimum value to signify the end of training. This comprises of $\approx 10\%$ of the total data and is removed from the data before commencing training. The difference between the output produced by passing the validation data through the algorithm and the true labels is not used to tune the ML algorithms but rather serves to mark the state of the algorithm. When this difference reaches a minimum low value the training is stopped and the generalization of the algorithm is tested.

3. **Testing part:** This part corresponds to $\approx 10\%$ of the total data. This part of the data is never passed through the ML algorithm until the error for the validation dataset reaches a low minumum value. The test data is treated as the final test data which can be used to benchmark how much the model has generalized. This is also serves to inform if the model has not learnt the features and has just learnt specific noise structures present only in the training data but not present in the test data. Note that this does not rule out systematic noise that is present in the entire dataset, and in fact unless the dataset is a very good representation of real variance in data, the results on the test set is usually a clue to the performance of the training.

Thus, the training, validation and test parts make up 100% of the data. Typically, when data sizes are small the validation and test are reduced to 5% each. Once the test part of the data has been passed through the algorithm, the 'model' is now deemed as 'trained'. The quality of the training reflects in the test scores. When the test scores are low but the same scores when computed for the training are high, it indicates that the model has not generalized very well. This is also known as 'overfitting'. This term refers to the idea that model not only trains on the data features but also on the random noise in the data. Such a model has to be retrained once again with appropriate mitigation strategies such as regularization in order to avoid such overfitting. **Regularization:** is defined as small penalty applied to the error so that the error does not easily minimize for small changes in the data, presumably produced when the model fits the noise. The converse issue is also possible where the testing error and the training error both are similar but high. This is a case of 'underfitting' where the data characteristics or features are not sufficiently high to produce a well trained model that generalizes. In such cases, we have numerous strategies to increase the exposure during training of the algorithm to data features.

- **Feature engineering:** is the strategy where we identify aspects of the data which lead to better validation accuracy and lower training error. We then train the ML algorithm with more of those data samples where such features are present or alternatively train the algorithms with only those features present where we have low validation error.
- **hyperparameter tuning:** in this strategy we change the parameters of the ML algorithm so that it produces lower training error and higher validation accuracy. The hyperparameters are subject to which ML algorithm we are training, sometimes tuning such hyperparameters can be difficult to achieve by hand and therefore we undertake a hyperparameter search where we systematically vary the values of the hyperparameters until we achieve the desired accuracy and error combination.
- **data augmentation:** is a method by which we apply specific transforms to the data that systematically provide increased diversity in the features and provide additional noise so that the intrinsic noise is not memorized. Thus it also acts as a regularizer.

In the subsequent subsection we describe the two broad types of algorithms used in this chapter namely ensemble algorithms of which we will explain the random forests and multi layer perceptrons of which we will describe deep neural networks (DNNs) and autoencoders (AE).

2.3 Random forests

Ensemble methods rely on the fundamental idea that an ensemble of methods when polled together produce a reliable and predictive result. These type of learning methods known as ensemble methods or algorithms was initially proposed with the use of (decision trees, Breiman, 1998). This bit of research was followed by the seminal work where it was shown that (Random forests, Breiman, 2001), which are a collection of decision trees indeed produce smaller test errors than the original decision trees algorithm. Since then Random forests have been the most sought after algorithm to perform machine learning tasks. In the field of exoplanets we have instances of the successful use of random forests with (SOFIRF, Gomez Gonzalez et al., 2018) and with (RF, Fisher et al., 2020).

In this thesis we use the random forest algorithm developed with (scikit-learn, Pedregosa et al., 2011). The basic unit of a random forest is a decision tree. As the name suggests a decision tree is step by step evaluation of many decisions, which are made based on the data. Decision trees are frequently used even outside of the ML algorithms to evalaute everyday chocies such as where to eat etc. A decision tree consists of the following parts:

- **root:** which is the start of the decision tree and is typically the first decision that needs to be made for instance, "Do we want to eat out today?"
- **nodes/leaves:** which is typically a new decision that needs to be made but is only influenced by the previous decision step i.e a node is created when we decide that we do want to eat, however a new decision has to be made about "what food do we want to eat"

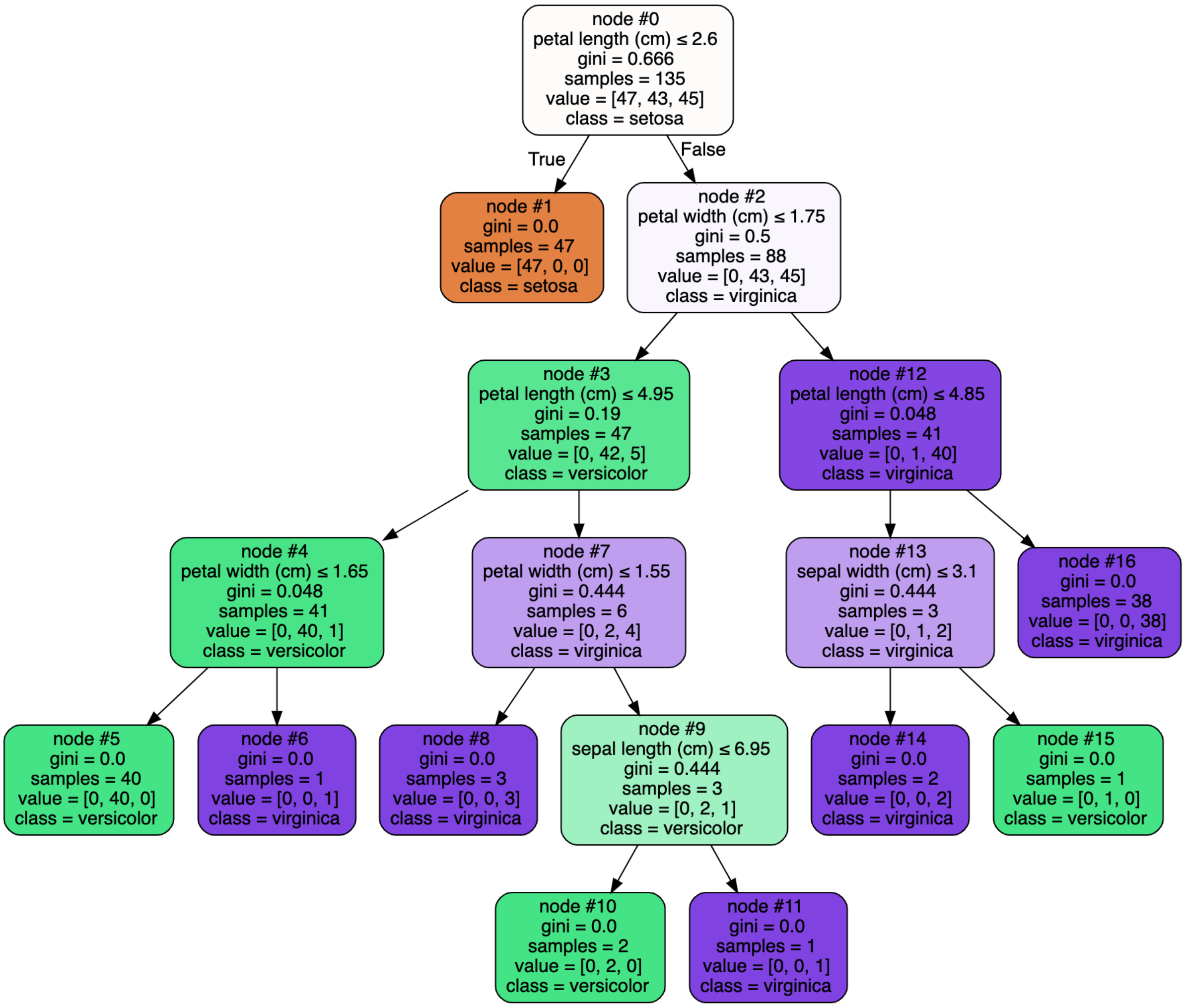


Figure 2.1: A sample decision tree

- **termination leaf/node:** which is the final decision that is reached and is typically the end of the tree.

Depending on how many questions need to be asked and what is the kind of decision that is needed, these trees can be very deep and have many leaves. A sample tree is show in Fig 2.1. This is a typical tree used to evaluate the species of an Iris flower. This decision tree was expressly chosen indeed for its explanatory value where different characteristics of the Iris flower are used to make choice which finally determine the class such as petal length and width. An ensemble of many such decision trees result in a 'forest' of decision trees which are collectively a 'random' forest.

Random forests typically have a number of useful tunable parameters, these parameters are used to both regularize overfitting and improve underfitting. The most common hyper-parameters that are tuned for random forests are the `n_estimators` which are the total number of decision trees in the forest. A key feature of the random forest is its feature explanability, typically the use of its feature importances, which explain the importance of the different data features in predicting the output of the random forest. In the example of the Iris species classification, Fig 2.2 shows the feature importances in predicting the same species as in Fig±2.1. The feature importances are relative measures such that the sum of the feature importances is always 1. As with the decision tree, the random forest uses multiple features to regress/classify the data and arrive at the desired output. As it does this process it uses different features, for example in the case of the Iris flower species classification, it uses the petal width as the most important parameter to make this prediction. This is quite insightful in two ways,

1. it is possible to intuit some understanding on the working of the random forest itself by knowing that some features are more relevant to make the desired predictions. This is usually the case when we know that the output depends on a few parameters but we are not sure which parameters act as features for the random forest.
2. it is possible that we have several parameters (as will be the case in my thesis) that could be used to produce the desired output however, we are not sure if this is necessarily true. For example, we have several spectral bins where the absorption lines corresponding to these of an exoplanet are present but we don't know which of these absorption lines allow us to detect and characterize an exoplanet.

Thus, feature importances form the basis to evaluate the features of the data that were learned by the random forest. This becomes relevant in multiple scenarios,

- when the trained model is overfitting and we can identify and remove those features which produce this overfitting
- when the model is underfitting, the feature importance will allow us to still identify the data features which have a higher relative importance. This can also be problematic because the relative importances can all have very similar values, this is the case when the random forest is not able to fit a generalised model and is the right case for dimensionality reduction
- and finally when the model is fitting well and the test and validation accuracies are similar the feature importances are used to study the the features in the data that allow the model to make accurate predictions.

Random forests are very easily implemented with a two step process with the library `sklearn` (Pedregosa et al., 2011).

2.4 Multi-layer Perceptrons

A well known term even amomg non-experts are the words 'neural network'. As the name suggests neural networks are inspired by the network of neurons that make up the mammalian brain. The basic unit of neural network is a neuron. A neuron consists of an input, output and an activation function that acts on the input similar to $f(.)$ in Eq 2.1. This activation function is a mathematical functiom which operates on the input. Such a unit is

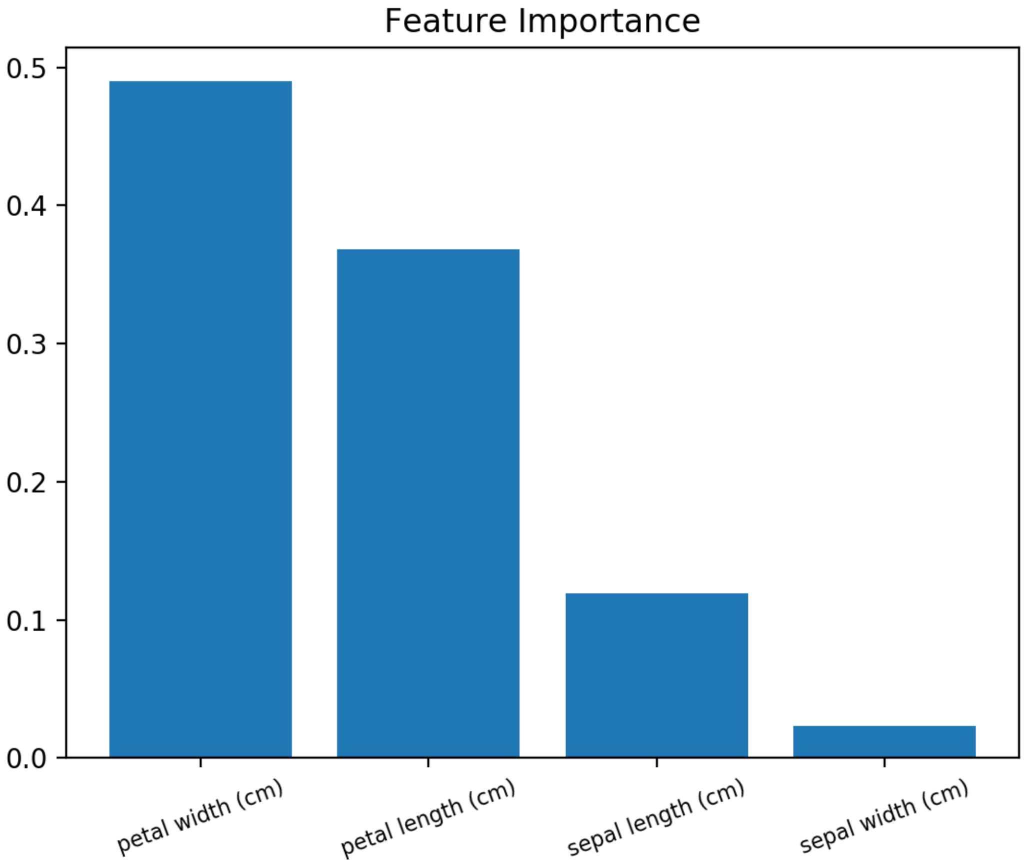


Figure 2.2: RF importances of features used to predict the species of the Iris flower. The y-axis is the relative importance; so a relative importance of 1 implies the most important and 0 is the least important feature to predict the species.

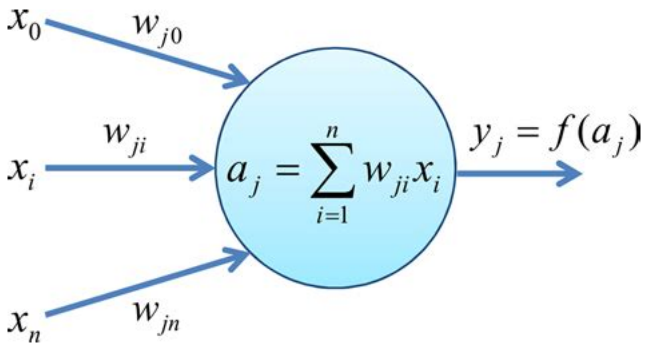


Figure 2.3: A sample percepton consisting of an input vector X_i and an output y is depicted here.

called a perceptron and a sample perceptron is depicted in Fig 2.3. The inputs are combined to form an activation a_j via j weights for each input vector value x_i making the weight matrix (w_{ij}. The output is defined as ,

$$y_j = f(\sum_{i=1}^{n} w_{ji} x_i) \tag{2.3}$$

Many such perceptrons together, producing an output vector y_{jk} for k perceptrons. This is known as neural network. The weights are the neural network parameters whose values can be altered during training. Activation functions $f(.)$ are typically fixed for the duration of training and varied if the validation results have not reached desirable values. This is known as a hyper parameter. The number of neurons in a neural network, the number of weights are also other hyper parameters that can be varied based on the validation results. The output of a neuron can be treated as an input to another set of perceptrons, and they can be in turn connected to another set of neurons and so on. Such a network feeds forward the inputs one layer to the next and such networks are called deep neural networks or multi-layer perceptrons.

Based on the kind of combination of w_{ji} and X_i perceptron networks can be further sub classified as convolutional neural networks (Zhang et al., 1990), recurrent neural networks and so on. Neural networks can also work on multi dimensional input as well long vectors. Configurations of neural networks also vary, for example other deep neural networks we can also have

2.5 Autoencoders

Autoencoders are special cases of multilayer perceptrons where the input and the output remain fixed but the intermediate layers form a mirrored encoder-decoder structure. This was first envisioned to denoise data and thus the (denoising autoencoder, Vincent, 2011) was invented. Typically, the autoencoder has the following main parts,

- **input:** is the input layer which is typically the same shape as the input vector and thus contains as many neurons as the input vector

- **encoder:** the encoder is a set of fully connected neurons which have as input the output of the input layer. These neurons typically are constructed with several layers with the number of neurons in each layer typically reducing in a pyramidal fashion. The last layer is the smallest layer and thus the output of this layer is a sparse representation of the input.
- **decoder:** the decoder is a mirror of the encoder both in the layer construction, the number of layers and the neurons used in each layer. It typically terminates in the same number of neurons as the input of the encoder and this layer connects to the output. The decoder layer typically reconstructs the input from the sparse representation, thereby recovering a noise free version (note this was the original motivation of the autoencoder).
- **output:** as with the input the output is also a vector equal to the size expected for the output.

To wrap this introduction up, we want to make a few points about the limitations of ML algorithms,

- **black box like behavior:** ML algorithms are very useful but also don't particularly lend to great deal of manual fine tuning. Which means that the features that the algorithms learn or not completely controlled and hence to offset this we provide the algorithms with large amount of data.
- **fine tuning training parameters is challenging:** while there are some thumb rules to follow when training to ensure good training, however there is no way to know if the hyperparameters of the neural network are optimal. Standard techniques such as a parameter space search etc are available but are used when the search itself is not very broad and the parameters have well defined limits

Chapter 3

Data generation based on a scientific hypothesis

Spectra have a wide range of properties, such as SNR, line width etc. and we need to choose properties that are in line with science goals of detection and characterization. The goal of this chapter is to define,

- the scientific objective of using spectra with ML algorithms,
- the data that we will use to achieve these objectives and
- the metric that will be used to state whether this chosen objective was met or not. The metric will be common for ML and non-ML algorithms and in principle will be algorithm independent.

In that quest we structure this chapter to start with introducing a science goal by defining the detection and characterization hypothesis. This will then be followed by describing the parameters chosen for data generation followed by a description of the data generation itself. This will be followed by describing the benchmark metric formulation. This chapter aims to set up the basic framework that will be used in the two methods chapters in this part.

3.1 Scientific goals of using spectral data with ML algorithms

Using spectra directly with ML algorithms has not been particularly successful for a specific set of characterization problems (e.g, Fisher et al., 2020). In order to not repeat previous studies and to redefine the goals of spectral data processing with ML algorithms we separate our goals as detection and characterization goals. Detection goals pertain to identifying that a spectrum indeed contains spectral features that pertain to an exoplanet. The characterization goals pertain to using the spectral absorption features to derive constraints on the $\mathrm{T_{eff}}$ and $\log(\mathrm{g})$ of the exoplanet in the spectrum.

3.1.1 Detection hypothesis

The detection hypothesis is expressed as the ability to identify an exoplanet spectrum when it is extracted from a pixel of a high contrast image based on the features of the extracted spectrum alone. In the context of high contrast imaging this means that no matter, the contrast of the exoplanet or the resolution of the spectrograph it is possible to make this fundamental distinction based on the spectral absorption features in the spectrum.

This would prove particularly useful when trying to discriminate between speckles and exoplanets in an residual cube. A well designed algorithm should be able to test and validate this hypothesis. In this part we will develop a ML and a non-ML based algorithm to test this hypothesis. We will use the non-ML algorithm to develop the benchmark values that the ML algorithms need to achieve to validate the detection hypothesis.

3.1.2 Characterization hypothesis

The characterization hypothesis is expressed as the ability to constrain physical exoplanetary parameters with a well defined and repeatable error bar when exoplanet spectral features are present in the spectrum. The exoplanetary parameters that we will consider in my thesis are the $\mathrm{T_{eff}}$ and $\log(g)$. In the context of direct imaging data it means that no matter the spectral resolution of the instrument, if exoplanetary spectral features are present in the spectrum then the hypothesis states that we are able to constrain the $\mathrm{T_{eff}}$ and $\log(\mathrm{g})$ within the stated error bars.

This hypothesis is motivated by the idea that it is possible that detection algorithms produce false positives due to their systematic biases. However, characterization constraints could allow to rule out such false positives when the detection is marginal.

These two hypotheses form the basis of our data generation, algorithm development and interpretation of those results. The detection hypothesis is disproven if we are not able to find (with either the ML or non ML algorithms) the point at which we are not able to make perfect detections no matter the resolution and contrast of the exoplanet present in the spectrum. The characterization hypothesis is also disproven if we are not able to define the minimum error-bar with which we constrain the $\mathrm{T_{eff}}$ and $\log(\mathrm{g})$.

3.2 Data generation to test our hypotheses

We have to test both our hypotheses on the same data that has sufficient variance in fundamental parameters, in this section we will define those paramters and the consequent data generation framework.

3.2.1 Choice of parameters for data generation

We confine ourselves to a smaller set of parameters to study. We choose three parameters, one of which is entirely intrinsic to the exoplanet we are studying and two of them are intrinsic to the the imaging strategy and the choice of spectrograph. The parameters we use in this part of the thesis to test our hypotheses are,

1. **Contrast C:** The contrast of an exoplanet is defined as the flux ratio of the mean flux emitted by the exoplanet to mean stellar flux. This is as such regarded as the crucial marker that will allow us detect faint exoplanets. In principle, the brightness of the exoplanet is a free parameter that can vary depending on the temperature, composition, atmosphere physics (such as presence of clouds) and environment of the exoplanet. The brightness of the star is mostly driven by its spectral type and surface gravity. Therefore, when the brightness of the star is fixed (by knowing accurately its spectral type and surface gravity), the contrast is only influenced by the brightness of the exoplanet in question.

2. **Signal to noise ratio of measured spectrum SNR:** The signal part of a spectrum is a measure of the number of photons in the spectrum that are from an observation (i.e from the star and the planet). The noise can be measured in several ways, but the most basic noise that is present in an observation is the photon noise. In this part of the thesis, we limit our scope to measuring this noise and thus the signal to noise ratio is a ratio of the observed photons and the photon noise. The SNR is typically a function of integration time of the observations, such that longer observing times lead to higher SNR.

3. **Resolution of the spectrograph R:** The spectral resolution is the ratio of a fixed wavelength to the difference between wavelengths of two consequent wavelength bins. R is typically dependent on the instrument and actually changes with the wavelength in consideration and is typically higher for higher wavelengths. To keep the interpretation simple we consider the R as computed for the smallest wavelength in our data.

These parameters have a very specific meaning in literature, the C for instance is the single parameter that defines the sensitivity in the data when computing contrast curves, the SNR is the parameter that is expected to be the limiting factor when developing a new instrument, and the R has long been considered the key factor in using spectra in direct exoplanet detection and (KPIC, Mawet et al., 2016) has prided itself on provide high SNR and high R spectra.

3.2.2 Synthetic spectra library

To test our hypotheses, these parameters have to be sampled over a large sample space to ensure that we are able to rigorously test our hypotheses. Additionally, instrument parameters will require us to have access to instruments with different resolutions but we will need to the exoplanets imaged with these instruments at different SNR and C. This also implies that we cannot control the types of noise that are present in instruments and we cannot limit our study to just the observation noise. Using data from different instruments and different observations will also bring into play, For example, instrument systematics such as the different Strehl ratios produces different amounts of stellar leakages produing variable data. If we choose just one type of instrument choose to drop R we still risk observation systematics such as different seeing on different nights. In addition to other well known effects such as wind halo, these make for a poorly conditioned dataset. While in the previous part we sought to verify that our algorithms work with both real and synthetic data, in this part we seek to verify that with working algorithms are our hypotheses valid.

In order to achieve the desired range in the data without taking into account inter-instrument and site variations, we resort to using synthetic data from the well known template library, (BT-SETTL, Allard et al., 1997, 2011). BT-SETTL conveniently also presents us with a simulation tool (PHOENIX, Allard et al., 2011) that allows us generate accurate atmospheric spectra by specifying the exoplanet properties. These models sample the $\mathrm{T_{eff}}$ range from 1200K corresponding to warm Jupiter type of exoplanets to 7000K corresponding to the B supergiant spectral type. The wavelength range varies from $0.1\ \mu$m up to $16\ \mu$m i.e from the infra red to the near visible spectrum. This allows to simulate the stellar spectrum and the exoplanet spectrum from the same library. Using BT-SETTL, we choose a basic grid of models and we generate synthetic from this grid depending on the requirement. The grid is defined by the following parameters,

$\mathbf{T_{eff}}$: The exoplanet atmosphere is chosen to be between $1200 \leq \mathrm{T_{eff}} \leq 1900$ K with a grid sampled every 100K. This range corresponds to that of warm Jupiters. These temperatures do not lend to very pronounced CO emissions, which are quite prominent at higher temperatures which make those templates somewhat easier to detect for cross correlations. The star is chosen to have a $5000 \leq \mathrm{T_{eff}} \leq 7000$ K surface temperature, the choice of temperature is not so relevant for this problem because beyon a temperature of 4000 K all of the molecules are fully ionized and the $\mathrm{T_{eff}}$ only impacts the continuum. In our processing we remove the continuum and hence it does not play a part in the analysis.

log(g): We choose values for the exoplanet within the existing BT-SETTL model to provide enough range to make an error bar estimate. We choose $2.5 \leq \log(\mathrm{g}) \leq 5.5$ for the exoplanet with a grid sampling rate of 0.5 dex. The stellar $\log(\mathrm{g}) = 2.5$ which is the solar $\log(\mathrm{g})$.

Wavelength λ: We choose wavelength ranges that allow us to probe the full near infra-red region, $1\mu\mathrm{m} \leq \lambda \leq 3\mu\mathrm{m}$. This also includes the Telluric absorption lines between 1.78 and $2.1\ \mu$m. The default spectral resolution of the data $R > 300,000$ and the linewidth is in . We resample the R as needed but we broaden the line width to match an instrumental profile so that the absorption line widths are realistic. We choose this width to be the same as the SINFONI instrumental line width.

3.3 Generating synthetic spectra

The goal of our tests is to ascertain if a) Which type of algorithm is able to satisfy either or both hypotheses b) what are the constraints we can draw on the algorithms themselves when they are tested. Consequently, the goals of generating synthetic spectra are also two fold,

1. to be able to explore a parameter space which is relevant from both the astronomical as well as signal processing point of view. This will allow us to the test the hypotheses, which are purely based on astronomy, but it will also allow us to understand the interplay between the parameters. This is relevant to the community to understand if, of the three parameters we have chosen, are there any which play an important role in validating these hypotheses.

2. To generate a large number of samples that can be used to train, validate and test the machine learning algorithms that are developed to test our hypotheses. In addition to ML algorithms needing a large number of samples to train and generalize well, we also need this parameter space well sampled in order to derive insights into the performance of the ML algorithms on this data. As in the case of a cross correlation based algorithm, the community will benefit if we are able to derive the limits at which these hypotheses were satisfied by ML algorithms.

Thus to generate spectra that are astronomically relevant, having realistic observation noise and finally lend to re derivation of the parameters (i.e now they can be viewed as parameters that can be re-estimated by an observer, we generate the same in three steps. We split the description of this into three subsections, first we create noise-less spectrum from a combination of stellar and exoplanetary spectrum. We then follow this with explaining the noise injection process and finally we describe the re-derivation of the SNR from this noisy spectrum.

3.3.1 Creating astronomically accurate synthetic spectra

The stellar flux (of the spectrum chosen from BT-SETTL) is measured as $F_{\lambda,\mathrm{star}}$. For the exoplanetary spectrum, an exoplanetary template spectrum is randomly chosen with a $1200 \leq \mathrm{T_{eff}} \leq 1900$ K and $2.5 \leq \log(\mathrm{g}) \leq 5.5$. Both spectra are irst re-sampled to a desired R such that $10^3 < R < 10^5$. Re-sampling is a two step process,

1. a wavelength vector is first generated with the desired wavelength range where we want to test our hypotheses. In this case we choose $1 \leq \lambda \leq 3$ μm. The R of this wavelength is set to the value we choose to generate our synthetic spectra.

2. We then interpolate a new synthetic spectrum for the specified wavelength bins based on the flux present in the template library. Note that the maximum resolution cannot exceed the spectral resolution of the BT-SETTL library (3×10^5)

We want to now make sure that every bin has exactly the same relative number of photons so that the sum of all the photons in each of the bins is the same for every spectrum. Once again to achieve this we have a two step proces, the first of which is normalization.

Normalization: Normalization of the flux per wavelength bin (F_λ) is performed so that the average flux in the spectrum is 1.

$$F_\lambda = \frac{F_\lambda}{\sum\limits_\lambda F_\lambda} \tag{3.1}$$

Thus, when we do this for the stellar and the exoplanet spectrum we have,

$$F_{\text{planet},\lambda,\text{normalized}} = F_{\text{star},\lambda,\text{normalized}} = 1 \tag{3.2}$$

Thus both the star and planet are at the same flux.

Scaling the spectra: We rescale both the $F_{\lambda,\text{star}}$ and the $F_{\lambda,\text{planet}}$ with appropriate flux values. Starting with the star, we scale the stellar flux as

$$F_{\text{star},\lambda,\text{new}} = F_{\text{star},\lambda,\text{normalized}} \times \text{SNR}^2 \tag{3.3}$$

The exoplanet usually has a total flux that is proportionally scaled to the stellar flux. The flux from the exoplanet is just a fraction of the stellar flux, expressed usually as a ratio of total planetary flux to the stellar flux known as the contrast C. Thus exoplanetary flux is expressed as,

$$F_{\text{planet},\lambda,\text{new}} = C \times F_{\text{planet},\lambda,\text{normalized}} \times \text{SNR}^2 \tag{3.4}$$

where $C \ll 1$. Once the two fluxes have been computed, they have now to be combined and ‘observed’ by a ‘telescope’. This act of observation will result in noise, this now the second step to generating synthetic spectra.

3.3.2 Noisy spectra generation

In order to simulate an observation where we extract a spectrum from an observed pixel, we combine both the stellar and planetary spectra as,

$$F_{\text{total},\lambda} = (1 - C)F_{\text{star},\lambda,\text{new}} + F_{\text{planet},\lambda,\text{new}} \tag{3.5}$$

Thus, $F_{\text{total},\lambda}$ represents the flux measured at every wavelength bin. The sum of the total integrated flux in the spectrum can be expressed as,

$$F_{\text{total}} = \sum_\lambda F_{\text{total},\lambda} = \text{SNR}^2 \tag{3.6}$$

as the $C \ll 1$. This spectrum still does not contain noise, and so the next step is to introduce realistic noise in each wavelength bin.

The act of observing photons arriving at the instrument is equivalent to counting photons. This results in an intrinsic counting noise which has a Poisson distribution. In order to now produce a noisy spectrum we replace the photon count in each wavelength bin has a value that is chosen from a Poisson distribution with a Poisson parameter k given by,

$$k_\lambda = F_{\text{total},\lambda} \tag{3.7}$$

This means that the flux distribution for each bin is given by,

$$\text{PMF}(k_\lambda) = \mu^{k_\lambda} \frac{\exp(-\mu)}{k_\lambda!} \tag{3.8}$$

A random value is chosen from this PMF, this random value will now represent the signal such that,

$$F_{\text{noisy},\lambda} = \text{random}\left(\text{PMF}(F_{\text{total},\lambda})\right) \tag{3.9}$$

In practice both of these equations are easily replicated with the NUMPY.RANDOM function of POISSON. Note that this function has to be applied repeatedly over every wavelength bin. This is now, one realization of a noisy spectrum. When this repeated many times with many number of spectra we will have spectra each having its own noise realization. This is the final step in generating the synthetic spectrum. [1] We generated spectra with a specific R, where the exoplanet spectrum is computed using a mean contrast C. These spectra maintain a fixed Signal-to-Noise Ratio (SNR), achieved by producing exoplanets with a known flux. Each spectrum produced using this method will have unique values for R, C, and the initial flux. The next subsection will explore how the SNR depends on the initial flux we insert.

3.3.3 Computing the SNR from the noisy spectrum.

In order to compute the SNR of the spectrum we first need to reliably measure noise. An advantage of using purely synthetic data is that we are able to precisely quantify the noise, which can then be used to compute the signal to noise of the cross correlation and the spectra. To start with we compute the amount of noise that is inserted in each wavelength bin. The precise expression of noise in any wavelength bin is

$$N_\lambda = F_{\text{noisy},\lambda} - F_{\text{total},\lambda} \tag{3.10}$$

The noise per bin follows Poisson statistics and therefore, it is the standard deviation of this noise that allows us to estimate the true noise in the spectrum expressed as,

$$\sigma = \sqrt{\frac{1}{\mathrm{N}} \sum_\lambda \left(N_\lambda - \frac{1}{N} \sum_\lambda N_\lambda \right)^2} \tag{3.11}$$

[1] the code for this part of the data generation is available here synthetic data generation

σ is now a generalized measure of the noise that is inserted in the spectrum. The signal inserted in the spectrum is given by Eq (3.6). This can then be turned into a SNR as,

$$\sigma = \sqrt{\frac{1}{N}\sum_{\lambda} N_{\lambda}^{2}} \tag{3.12}$$

and because of the properties of a Poisson distribution, we can simplify the standard deviation of the noise in the spectrum as,

$$\begin{aligned} \sqrt{\frac{1}{N}\sum_{\lambda} N_{\lambda}^{2}} &= \mathrm{SNR} \qquad (3.13) \\ \therefore \sigma &= \mathrm{SNR} \qquad (3.14) \end{aligned}$$

Consequently, we can now express the signal to noise of the spectrum as,

$$\frac{\sum_{\lambda} F_{\text{total},\lambda}}{\sigma} = \frac{\mathrm{SNR}^2}{\mathrm{SNR}} = \mathrm{SNR} \tag{3.15}$$

In other words, starting from the initial flux that we set in the spectrum we can derive the final signal and noise with just two equations,

$$\begin{aligned} \text{signal} &= \sum_{\lambda} F_{\text{total}} \qquad (3.16) \\ \text{noise} &= \sqrt{\sum_{\lambda} F_{\text{total}}} \qquad (3.17) \\ & \qquad (3.18) \end{aligned}$$

These spectra can in principle be sampled over an infinite range of parameter space other than computational limitations placed on R because of the size of the vectors. As a next step, we will need to define a benchmark on which we can test our hypotheses.

3.4 Developing a common benchmark for ML algorithms and non-ML algorithms

Defining a common benchmark for this thesis section is challenging due to three essential criteria: algorithm performance irrespective of scientific goals, a benchmark for the detection hypothesis, and a benchmark for the characterization hypothesis. All algorithms in this section must undergo testing against these benchmarks. Initially, each algorithm is assessed for its general performance. Once an algorithm meets this benchmark, we proceed to evaluate the detection and characterization hypotheses. The goal of this exercise is to understand two things,

1. An algorithm could be well defined and developed (in the case of ML algorithms well trained) but does it pass the basic benchmark to be used in scientific data processing

2. what are the limits that the algorithm places on the scientific hypotheses given that that the algorithms passes the basic performance measures.

Finally, the end goal of this benchmarking procedure is to understand the strengths and limitations of the algorithms developed for this part

3.4.1 Evaluation of the algorithms through confusion matrices

Once our algorithm is developed, we will assess it on a dataset defined by a specific range of (C, R, SNR). The primary objective is to utilize exoplanet-specific absorption features, distinct from stellar emissions and other noise, to differentiate between spectra with and without exoplanets. We set limits on the algorithm's output to ensure accurate discrimination. We impose constraints, such as a false positive rate not exceeding 10^{-4} which is directly related to detection accuracy. In the case of characterization, involving $\mathrm{T_{efF}}$ and $\log(\mathrm{g})$ inference from absorption lines, a false positive entails misrecognizing absorption lines and inferring non-existent parameter values.

This benchmark, applicable to both ML and non-ML algorithms, facilitates false positive evaluation while adjusting thresholds. The consistent and well-designed confusion matrix is employed for this purpose. True positives occur when the algorithm correctly identifies synthetic spectra with exoplanetary features, while false positives arise when spectra without exoplanetary features (i.e., $C = 0$ in Eq 3.5) are incorrectly labeled as having such features. A sample confusion matrix in Tab 3.1 outlines two conditions applied to Eq 3.5, specifically evaluating pure stellar spectra where the algorithm correctly identifies spectra lacking exoplanets. The algorithms adhere to thresholds for counting spectra above or below a fixed threshold in computing this matrix.

Condition	Predictions	
$C > 0$	True positives	False negatives
$C = 0$	False positives	True negatives

Table 3.1: A sample confusion matrix that is the benchmark that is used to evalute the algorithm in consideration. The cells in green are values that the algorithm needs to correctly predict and those in red are those parameters that the algorithm makes mistakes on. The False positives have to be limited to $10^{-4} \leq$ whereas we don't put any constraint on the False negatives.

3.4.2 Quantifying the algorithm as a detection tool

Once an algorithm clears individual confusion matrix tests, it's essential to explore the detection hypothesis limits. This hypothesis can be fully or partially satisfied under specific data constraints, requiring quantification across diverse parameter spaces. To achieve a comprehensive understanding of detection limits, we propose a detection matrix assessing the detectability of a warm Jupiter across varied R and SNR values.

The detection matrix aims to:

1. Define intrinsic detectability of warm Jupiters based on instrument resolution and observation signal-to-noise ratio.

2. Establish the minimum contrast enabling detection with a fixed instrument resolution.

3. Specify the minimum SNR required for observation based on a candidate exoplanet's contrast.

This matrix is valuable for quantifying intrinsic detectability and comparing algorithms operating on spectra. It offers insights into suitable observation types and instruments for detecting specific exoplanet types, aiding in algorithm selection.

The detection matrix features rows representing increasing SNR from bottom to top and columns representing ascending spectral resolution. Each cell, indexed by observing and instrument parameters, defines unique spectral properties for detection limits. The detection limit, defined as the maximum contrast for detectability, is determined using our algorithm. Tab 3.2 presents a sample of this matrix.

SNR of the observation	Resolution of the instrument			
	R_1	R_2	..	R_n
SNR_1	C_{R_1,SNR_1}	C_{R_2,SNR_1}	..	C_{R_n,SNR_1}
SNR_2	C_{R_1,SNR_2}	C_{R_2,SNR_2}	..	C_{R_n,SNR_2}
..	..	..	..	..
..	..	..	..	..
..	..	..	..	..
SNR_n	$C_{R_1,\mathrm{SNR}_\mathrm{n}}$	$C_{R_2,\mathrm{SNR}_\mathrm{n}}$	..	$C_{R_n,\mathrm{SNR}_\mathrm{n}}$

Table 3.2: Sample detection matrix where the rows represent a different SNR numbered from 1 to n where SNR_1 represents the spectrum with the highest SNR. Columns are indexed by the wavelength resolution of the spectra that are processed by the algorithm. Each entry correesponds to the contrast at which the algorithm is able to detect the exoplanet observed with its correspoding SNR amd R. Thus a contrast C_{R1,SNR_1} corresponds to the contrast at which the exoplanet can be detected when observed with a SNR_1 and a resolution R_1 and so on.

The detection matrix above has rows indexed by SNR_1 to SNR_n, representing different signal-to-noise values for generated spectra. In this example, the maximum SNR is SNR_1, and the minimum is SNR_n. Columns are indexed by spectral resolutions, ranging from R_1 to R_n, where R_1 is the smallest resolution, and R_n represents the spectra with the highest resolution. Each entry in the detection matrix is indexed with $C_{R,\mathrm{SNR}}$, representing the contrast at which the exoplanet is detected when synthesized with a specific SNR and R. The ordering of contrast entries is not preferential, but the matrix provides insight into the detectability of increasing contrast with ascending SNR (from the bottom to the top with the same R) and changing R (from left to right along the columns).

3.4.3 Quantifying the characterization hypothesis using the characterization matrix

Spectral features, such as molecular absorption lines, depend on the exoplanet's effective temperature ($\mathrm{T_{eff}}$) and surface gravity ($\log(g)$). Algorithms processing spectra for exoplanet detection can also yield insights into these parameters, aligning with the characterization hypothesis. This hypothesis assumes characterization is possible when exoplanetary features are present in the spectra.

The characterization matrix serves to:

1. Quantify the presence of absorption features for different templates using both ML and non-ML algorithms. 2. Utilize this quantification to derive error bars for both $\mathrm{T_{eff}}$ and $\log(g)$. 3. Visually provide evidence for preferred combinations of $\mathrm{T_{eff}}$ and $\log(g)$.

Unlike the detection matrix, the characterization matrix needs to allow quick and easy analysis, especially when comparing numerous spectra.

This matrix will resemble the one discussed in [refer to Part II Methods chapter]. In that context, it justified the best template for HD142527b and demonstrated the template's negligible impact on PDS70. For quantitative analysis, we will examine the algorithm's output variation for different templates, measuring the dispersion to quantify error in $\mathrm{T_{eff}}$ and $\log(g)$.

The dispersion of the detection parameter across templates serves two purposes:

1. Constrains the algorithm's sensitivity to supplying the closest template to the exoplanet present in the spectrum. 2. Constrains the sensitivity of both $\mathrm{T_{eff}}$ and $\log(g)$ to varying templates.

The ability to produce this characterization matrix relies on sufficient evidence that the algorithm is indeed sensitive to changing templates. In subsequent sections, we will apply this benchmark to both ML and non-ML algorithms and discuss the results obtained.

Chapter 4

Performance of cross correlation based algorithm on the benchmarking metric

In this chapter we focus on the non-ML algorithms designed to test hypotheses using spectra. The primary technique employed to evaluate the spectra is cross correlation. The algorithm encompasses the following steps, detailed in the methods section:

- Subtract the stellar template from the spectrum generated in Eq 3.5 and conduct initial pre-processing (refer to [Part II Methods, pre-processing]).
- Compute cross-correlation using the basic equation (refer to [Part II Methods, cross-correlation]), with a specific velocity dispersion.
- Compute detection and characterization parameters as appropriate.
- Produce detection and characterization matrices from these parameters.

Results of this algorithm with various inputs and exploration of parameter space will be presented in the results section. The chapter concludes by establishing criteria for ML algorithms to be considered: Better than ML algorithms, validating both hypotheses; as good as ML algorithms, validating at least one hypothesis; worse than ML algorithms, not considered appropriate for hypothesis validation. The goal is to summarize the functioning of a basic algorithm utilizing cross-correlations, capable of producing scientifically relevant results. This also forms the basis for evaluating ML algorithms.

4.1 Methods

The cross correlation based non ML algorithm takes as input the spectrum resulting from Eq 3.5. The broad steps of the algorithm are as follows,

1. In the preprocessing stage, spectra are deconvolved from stellar features and undergo continuum subtraction. This resultant spectrum, free from stellar influences, is typ-

ically cross-correlated with a template. However, it finds application beyond this, especially in characterizations unrelated to continuum-based analysis.

2. Post cross-correlation, cross-correlation coefficients emerge at various velocity dispersions between the template and input spectrum. Though non-normalized, these coefficients offer insights into spectrum similarities.

3. Ultimately, detection and characterization hinge on parameters like cross-correlation signal-to-noise ratio and template log-likelihood. These parameters can be leveraged for log-likelihood-based characterizations.

4.1.1 Pre-processing and cross correlation

The spectral preprocessing aligns with the methodology detailed in [See Part II, Methods, pre-processing]. For synthetic data, where the star's pixels are precisely known, we utilize the known stellar spectrum to eliminate stellar features. The procedure involves:

1. Computing the sum of the stellar spectrum in Eq 3.3, serving as the normalization reference for the spectrum in Eq 3.5.

2. Deriving the reference spectrum by applying a Savitzky-Golay filter (order 1, window size 101) to the scaled stellar spectrum.

3. Dividing the reference stellar spectrum from each synthetic spectrum, resulting in a continuum-free spectrum ([Refer to Part II, Methods, pre-processing]). The noise, exoplanet spectrum, and contrast remain unaffected.

The subsequent step entails cross-correlating this spectrum with a template spectrum to generate a cross-correlation vector for different velocity dispersions. Employing Eq [Refer to Part II, Methods, cross-correlation equation], we choose a velocity dispersion range of -50 to 50 km/s. Unlike [Refer to Part II Methods, cross-correlation], the synthetic data allows for a well-measured noise level, eliminating the need for a broad dispersion range for noise computation. As there is no introduced relative velocity shift between the exoplanet and the observer, a small velocity dispersion within -50 to 50 km/s is used for cross-correlation. The velocity resolution (δv) is related to spectral resolution (R) as,

$$\delta v = \frac{1}{R}. \tag{4.1}$$

The resulting cross-correlation vector is then employed to derive both detection and characterization parameters.

4.1.2 Development of the detection parameters and its application

Claiming a detection in the field of exoplanet detection is probably one of the most contentious issues in the field. As we have seen in [Refer Part II methods, SNR] it is possible to use signal to noise from the cross correlation to define a detection threshold. However, computing this signal to noise ratio is computationally intensive. In this part we will re-compute the signal to noise of the cross correlation by taking into account the auto correlation and taking the noisiness of the spectrum into account.

To begin with we define the 'signal' of the signal to noise which is defined as,

$$S(0) = CC(0) \tag{4.2}$$

where $CC(0)$ is the cross correlation value at $v = 0$.This follows from the cross correlation value defined in [Refer Part II cross correlation] The 'noise' comprises of two quantities, the noise measured of the spectrum σ and the auto-correlation value. The auto-correlation value is defined as in the cross correlation function as follows,

$$\mathrm{AC}(v) = \sum_{\lambda} M_{\lambda} \times (M_{v,\lambda} - M_{\mathrm{SG},v,\lambda}) \tag{4.3}$$

$\mathrm{AC}(v)$ represents the auto-correlation value at velocity dispersion v. As previously defined, M_{λ} denotes the model flux in wavelength bin λ, and the addition of the v suffix indicates the wavelength-shifted version resulting from a velocity dispersion of v. The $M_{\mathrm{SG},v,\lambda}$ is the Savitzky-Golay filtered version of the template which also acts as mean subtraction for the cross correlation. We define the cross correlation SNR only for $v = 0$ and is denoted as $\mathrm{SNR}_{\mathrm{ccf}}$. We then compose the noise portion of this signal to noise thus,

$$N(0) = \sigma\sqrt{\mathrm{AC}(0)} \tag{4.4}$$

The next step is to use both these parts to compose the signal to noise metric.

Detecting with $\mathrm{SNR}_{\mathrm{ccf}}$ poses challenges. Unlike established SNR techniques such as Mawet et al. (2014), which have been thoroughly vetted and their limitations well-documented (e.g., STIM Pairet et al., 2019), the robustness of calculating $\mathrm{SNR}_{\mathrm{ccf}}$ lacks comprehensive literature. Consequently, understanding the false positive rate at different thresholds for such measures remains limited. Despite this, setting a threshold is imperative. This study contributes by estimating the false positive rate at $\mathrm{SNR}_{\mathrm{ccf}} \geq 6$. It's crucial to note that this signal-to-noise ratio merely gauges similarity between the target spectrum and its template and does not indicate the significance of the detection. Finally, we use the expression in Ruffio et al. (2019) to compute the cross correlation signal to noise as follows,

$$\mathrm{SNR}_{\mathrm{ccf}} = \frac{\mathrm{CC}(0)}{\sigma\sqrt{\mathrm{AC}(0)}} \tag{4.5}$$

In order to make this a detection parameter we take the following steps,

1. we generate several synthetic spectra with a fixed R and SNR,
2. then we cross correlate these spectra with a template of choice. In order to not be very optimistic in estimating the sensitivity of this technique we choose a spectrum that is a 1 grid point both in $\mathrm{T}_{\mathrm{eff}}$ and $\log(\mathrm{g})$ away from the synthetic spectrum exoplanetary

template to act as the template.

3. Finally we compute the $\mathrm{SNR}_{\mathrm{ccf}}$ for each spectrum. We then choose as the limiting contrast the contrast at which $\mathrm{SNR}_{\mathrm{ccf}} - 6$ is the lowest positive value. We repeat this experiment 100 times and compute the mean contrast with 100 repetitions to avoid any random effects.

4. We fill the value of contrast in the detection matrix to form the detection parameter for the R, SNR combination.

4.1.3 Development of the characterization parameter and its application

The cross correlation coefficient is a measure of similarity between a template spectrum and the target spectrum. The extent of this similarity is quantified by $\mathrm{SNR}_{\mathrm{ccf}}$. While this is a measure of similarity, characterization aims to find the **most** similar template spectrum to the target spectrum. The characterization parameter plays a crucial role in capturing specific information, including:

- Evaluating the degree of similarity between a chosen template (within the parameter space discussed in the previous chapter) and the target spectrum.
- Quantifying the relative dissimilarity of other templates compared to the target spectrum, facilitating the identification of the spectrum that best aligns with the exoplanetary template in the synthetic spectrum.

Therefore, the process of characterization involves optimizing the parameter space to define the most suitable value for the characterization parameter. Our approach involves utilizing the log-likelihood of the cross-correlation to determine the optimal combination of $\mathrm{T}_{\mathrm{eff}}$ and $\log(\mathrm{g})$, accompanied by an assessment of the uncertainty associated with estimating this value. In this subsection, we clarify our derivation and application of the characterization parameter by addressing three key questions: a) What advantages does log-likelihood offer? b) How do we establish the relationship between cross-correlation and log-likelihood? c) How can we use the properties of this log-likelihood to derive error bars for the characterization of an exoplanet?

Why do we need a log-likelihood at all?

The idea of using log-likelihood as a method to estimate parameters inferred from cross correlations was first introduced by Zucker (2003) who derived one of the earliest cross correlation to log-likelihood expressed as,

$$LL \propto \log(1 - \mathrm{CC}^2) \tag{4.6}$$

The use of log-likelihood and cross correlations is justified in Brogi and Line (2019), the use of the negative sign and the dependence on the square of the cross correlation allows for two crucial factors to be considered,

1. using the square of CC will allow us to make a deep likelihood trough only for truly high cross correlation values and any small variations will get ruled out. This will allow us to fit an inverted Gaussian and derive the uncertainty.

2. The negative sign enables to only have true cross correlations of absorption features to absorption features to produce a strong LL, This in turn will let us ensure that the best matching template is the one that truly matches the features of the template to its noisy equivalent.

How do we adapt this concept to our specific case?

Eq 4.6 conceptually defines the relationship between the cross correlation and the log likelihood. However, we still need to take two aspects into consideration when it comes to synthetic data as discussed in Brogi and Line (2019),

1. the nature of the noise. In this case the noise distribution is random with no intrinsic structure to the noise in each bin. Therefore, we can consider σ to be the noise used for LL as well.

2. The effect of the stellar continuum, which has been removed by mean subtraction.

Therefore, we can go about using the derivation of the log-likelihood as described in Ruffio et al. (2019),

$$LL \propto -\frac{\frac{CC^2}{\sigma^4}}{\frac{AC}{\sigma^2}} \tag{4.7}$$

This equation can now be simplified and the proportionality turned into an equality by using a scaling constant a, as While the SNR is derived from Eq(**??**), the LL is computed based on Ruffio et al. (2019) as,

$$\mathrm{LL} = -\mathrm{a}\frac{\mathrm{CC}^2}{\sigma^2\mathrm{AC}} \tag{4.8}$$

This constant of proportionality is related to the mean continuum energy present in the spectrum and Zucker (2003) shows that this constant can be set $a = 1$ for the two conditions of continuum and mean subtraction. Thus, we have the final LL computation is giveny by,

$$\mathrm{LL} = -\frac{\mathrm{CC}^2}{\sigma^2\mathrm{AC}} \tag{4.9}$$

How do we use this log-likelihood to compute the uncertainties?

The characterization matrix consists of rows of T_{eff} and columns of $\log(\mathrm{g})$ and we fill each cell with the LL computed using the spectrum with the corresponding combination of rows and columns. This results in 2D matrix, however the error bars have to be computed separately in $\mathrm{T_{eff}}$ and $\log(\mathrm{g})$. In order to now compute the error bars separately, we first find the cell with the lowest value of LL. This is the initial guess of the parameters that we supply to the inverse Gaussian fit. Then we fit an inverse Gaussian as defined by,

$$LL_g = \frac{A}{\sigma_x\sqrt{2\pi}} \exp\left(-\frac{(x-\mu_x)^2}{2\sigma_x^2}\right) \tag{4.10}$$

where LL_g is the inverse Gaussian we fit to the LL, x is the parameter we are guessing i.e either $\mathrm{T_{eff}}$ or $\log(\mathrm{g})$, μ_x and σ_x are the mean and standard deviation of the Gaussian fit. These represent the estimated value of the parameter and our desired error bar. As guesses we have to specify th starting value of the parameter, we first start with a guess value of T_{eff} or $\log(\mathrm{g})$ at the lowest value cell and for the σ_x we choose the desired error bar.

4.2 Results

It has been stated that for both the ML and non-ML algorithms, the confusion matrices for each algorithm will be computed for specified parameter space. While this is useful to evaluate the algorithm, it does not make statement about the detection and characterization of exoplanets. Therefore, the description of the confusion matrix of the cross correlation algorithms will be discussed in §??. In this section we will discuss the following aspects,

1. the parameter space that was used to generate these experiments and how the cross correlation fares in this parameter space. We will describe the scientific motivation of this parameter space,
2. this is followed by the description of the detection matrix and a brief discussion of what the discussion matrix means for scientific data processing and
3. finally, we will describe the error bars which we were able to compute using the characterization matrix.

We will wrap this section with implications that these results mean for the scientific community in general.

4.2.1 The parameter space that we use to generate data

As a reminder, the parameter space here is the range of values we supply for the contrast (C), the instrument resolution R and the spectral signal to noise ratio (SNR). Each of these parameters have different consequences for the science, the instrument selection and finally observation baseline duration. It is not possible to practically cover every scenario but a good choice of the parameter space selection would be to cover as wide a net as possible for the parameters in question. Alternatively, we could choose to define limits to the parameters which will have scientific relevance. Before we describe the choice of parameters we must first define what is the "sampling" criteria for these parameters. Due to the advantage of using synthetic data, we can freely define our sampling criterion and thereby have as fine a mesh as needed. The sampling criteria has both storage and computation time consequence particularly if we vary these parameters simultaneously. The parameters C and SNR do not have any additional computational requirements when they are changed, however, R has increasing computational constraints. R is directly proportional to the total number of wavelength bins in the data,

$$\mathrm{N} \approx 2\mathrm{R} \tag{4.11}$$

where N is the total number of wavelength bins in the data. While this relationship changes slightly and the constant of proportionality (2) rises with increasing R it will still remain between 2 and 2.5. Therefore, when we generate the parameter space the computational costs of computing cross correlations are significantly higher at higher resolutions. This will become pertinent when using ML algorithms, but for the purposes of this chapter the only limit on the value of R is the resolution limit of the template library.

Parameter ranges of C:
Contrast is defined in the Eq **??**. This contrast varies over each wavelength bin such that different wavelength bins have varying contrasts. However, when referring to contrasts, we will always refer to the mean contrast of the spectrum. The contrasts we choose lie on a logarithmic scale between $C_{\{rmmin}$ and C_{max}. We choose the $C_{\mathrm{min}} = 10^{-2}$ based on the contrasts of HD142527b which has been detected well by the cross correlation algorithm [Refer to Part II Results, HD142527]. The choice of C_{max} based on the detection limits set the warm Jupiter (1000246b Quanz et al., 2015; Cugno et al., 2023) with a contrast of between 10^{-5} and 10^{-6}. Thus with the C_{min} and C_{max} set for the parameter space, we can produce spectra with varying contrasts. We choose a logarithmic sampling rate of 1.2×10^{-3} for each step of contrasts between C_{min} and C_{max}.

Parameter ranges of SNR :
The ranges of SNR were chosen based on the mean SNR per wavelength bin. The signal to noise ratio for each wavelength bin is a statistical average that is expressed as

$$\mathrm{SNR}_{\lambda} = \frac{\sum\limits_{\lambda} F_{\mathrm{noisy},\lambda}}{\mathrm{N}} \tag{4.12}$$

where N is the total number of wavelength bins in the spectrum. We set the value of SNR_{λ} to a certain fixed value so that each of the wavelength bins has a fixed average SNR. This ensures that individual variations in flux level does not bias the study which could lead to the downweighting of random wavelength bins. Note that when R is fixed the relationship between SNR and SNR_{λ} is fixed

$$\mathrm{SNR}_{\lambda} \propto \mathrm{SNR} \tag{4.13}$$

and thus we will always express our results in terms of SNR. We explore the full range of SNR_{λ} between 0.01 to 10^5 for the lowest R and between 10^{-3} to 10^3 for the highest R. In terms of SNR this range is between 100 to 10^8.

For a fixed R we now can generate a large number of spectra with varying SNR with exoplanets templates simulated at different values of C. An example of such an interaction is Fig 4.1. On the x-axis is SNR and on the y-axis is the $\mathrm{SNR}_{\mathrm{ccf}}$. The horizontal dashed line represents the cut off of $\mathrm{SNR}_{\mathrm{ccf}} > 6$ which is our cut off. The orange and blue lines represent those spectra with exoplanets at a contrasts of 10^{-4} and 10^{-6}. The two vertical lines consist of the upper and lower bound of the SNR. The window between the two vertical lines consists now of spectra where very few exoplanets with a contrast of 10^{-6}. An interesting observation is that beyond an $\mathrm{SNR} > 5$ the relationship between $\mathrm{SNR}_{\mathrm{ccf}}$ and SNR becomes linear and thus it validates our cut off criterion.

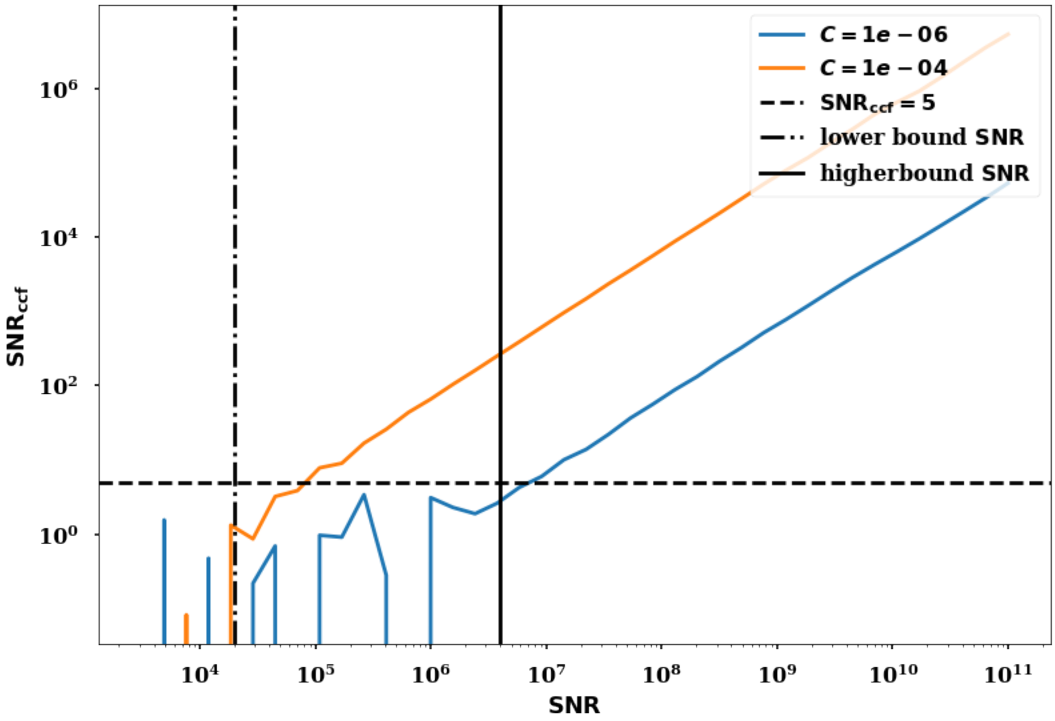


Figure 4.1: Caption

Parameter space selection of R:

The parameter space of R is limited by the resolution of the BT-SETTL library itself and hence we limit ourselves to a $R < 300,000$. We choose a sampling rate of 50 between 10^2 and 10^5 so that we have a total of 20 different resolutions for each spectrum.

4.2.2 The detection matrix

The goal of defining the parameter space and designing the algorithm was to populate the detection and characterization matrices. Given the context here, the detection matrix itself is not very difficult to explain and is indeed somewhat self explanatory. However, it is important to still define what each row and column means in the detection matrix for warm

Jupiters in Fig 4.2 This detection is produced as described in the methods chapter. Each cell

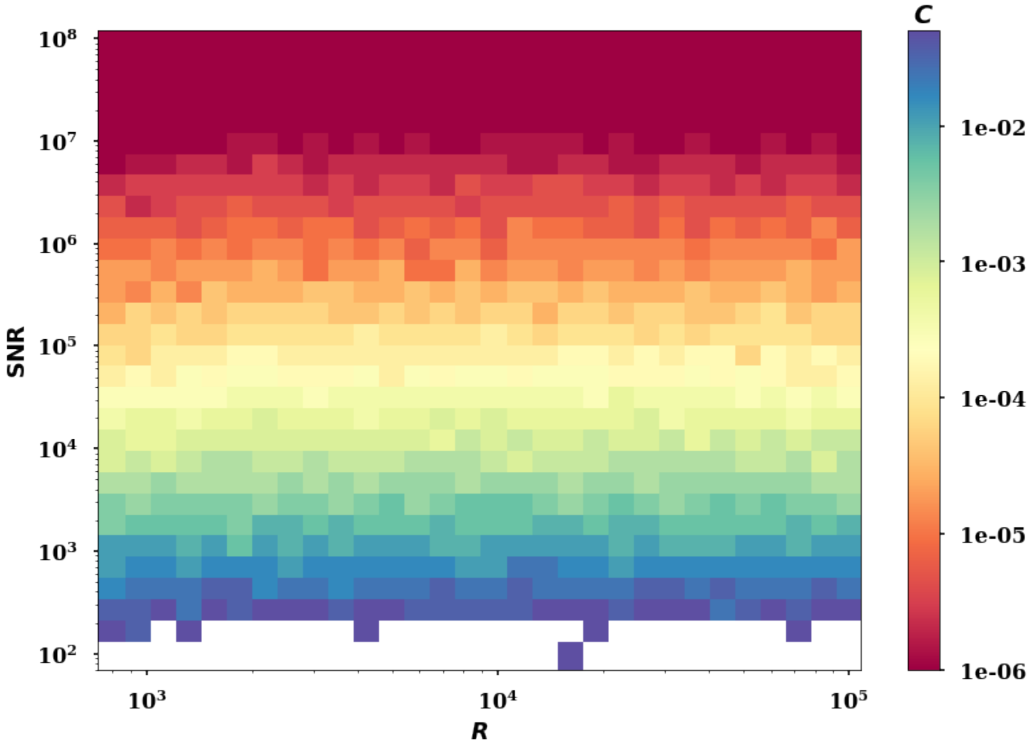


Figure 4.2: Caption

in the matrix represents a contrast between 10^{-1} and 10^{-6}. From the bottom to the top the rows represent SNR values from the lowest to the highest. The bottom two rows i.e $\mathrm{SNR} < 300$ contain a lot of white cells, these are cells where even for the unphysically low contrasts of 0.1 the exoplanets were not detected. These SNR represent cases where no matter what the R and C are the exoplanet is completely undetectable. There are many interesting points to note from this detection matrix. As with Fig 4.1 we saw that a linear relationship exists between the $\mathrm{SNR}_{\mathrm{ccf}}$ and SNR which implied that the SNR seems to directly influence the cross correlation strength. The detection matrix seems to reinforce this idea when detecting faint exoplanets. As the SNR increases the matrix gets redder indicating the detectability of fainter companions. The top 3 rows are at a detection limit of 10^{-6}, this means that the faintest companion is completely detectable for the highest signal to noise of the spectrum.

Another interesting point to be noted in this matrix is when the SNR is fixed for higher R there is no measurable improvement in sensitivity of detection of faint exoplanets. There are a some slight improvements in detection sensitvity at higher SNR particularly at $\mathrm{SNR} \approx 10^5$ and $R \approx 5 \times 10^4$ where we are able to detect a slightly higher contrast exoplanet. But once we increase the SNR we wash out any differences there may be at higher resolution. Note that for a constant SNR as R increases SNR_λ reduces, therefore the advantage of having several wavelength bands is outweighed by lower SNR_λ. Using this detection matrix, it is clear that for warm Jupiters, this trade-off indeed means that increase of R no discernable advantage for detection.

4.2.3 Characterization error bars

The characterization of an exoplanet consists of two subsequent steps,

1. produce the characterization matrix with either the $\mathrm{SNR}_{\mathrm{ccf}}$ or LL where both the matrices have to be mirror images of each other and
2. fit the inverse Gaussian to LL characterization matrix.

For the first part, it is important to first look at a sample plots of the characterization matrices at specific values from the parameter space. We choose a spectrum with a $T_{\mathrm{eff}} = 1500$ and a $\log(\mathrm{g}) = 4.5$. We start with a lower SNR where the exoplanet is not detectable in Fig 4.3. The top left is the characterization matrix with $\mathrm{SNR}_{\mathrm{ccf}}$ as the parameter within the matrix and LL as the parameter on the top right. The bottom of the plot is the shape of the cross correlation function with respect to the velocity. Note that for the contrast involved this exoplanet is not detectable and appropriately it is not possible to quantify the error bar in the effective temperature. This also reinforces the idea of having a characterization hypothesis where the exoplanet is characterizable only if an exoplanet is definitely detectable. Based on Fig 4.2 as we increase the SNR we will be able to detect the exoplanet. At $\mathrm{SNR}_{\mathrm{ccf}} < 5$ the relationship is still not linear and therefore, it is possible that we don't have enough photons in enough wavelength bins to have a detection and characterization. Therefore, we depict a case where $\mathrm{SNR}_{\mathrm{ccf}} > 5$ inf Fig 4.4 A couple of interesting points of comparison between these two last figures,

1. Firstly, when $\mathrm{SNR}_{\mathrm{ccf}} < 5$ we notice that the depth of the LL is fairly shallow. This results in a wild estimate of $\mathrm{T}_{\mathrm{eff}}$ whereas a more reasonable estimate of $\log(\mathrm{g})$ is present but this is because of the limited number of bins.
2. Secondly, the shape of the cross correlation over velocities is strikingly different, which is also reflected in the depth of the LL. This also brings to light the close relationship between detection and characterization. In fact the shape of the cross correlation along velocity seems to indicate the accuracy of fit in the characterization matrix.
3. Finally, the relevant question that is raised when both the plots are compared is whether, increasing the SNR even further will result in a more accurate estimate of $\mathrm{T}_{\mathrm{eff}}$. If that is not the case, then could increasing the resolution lead to better constraints on the effective temperatures and surface gravities?

We answer the first question by cross correlating a spectrum with a much higher SNR. Fig 4.5 shows a case where the cross correlation values are much higher and the distribution of the cross correlation over velocities is much more Gaussian like. This also leads a 'deeper'

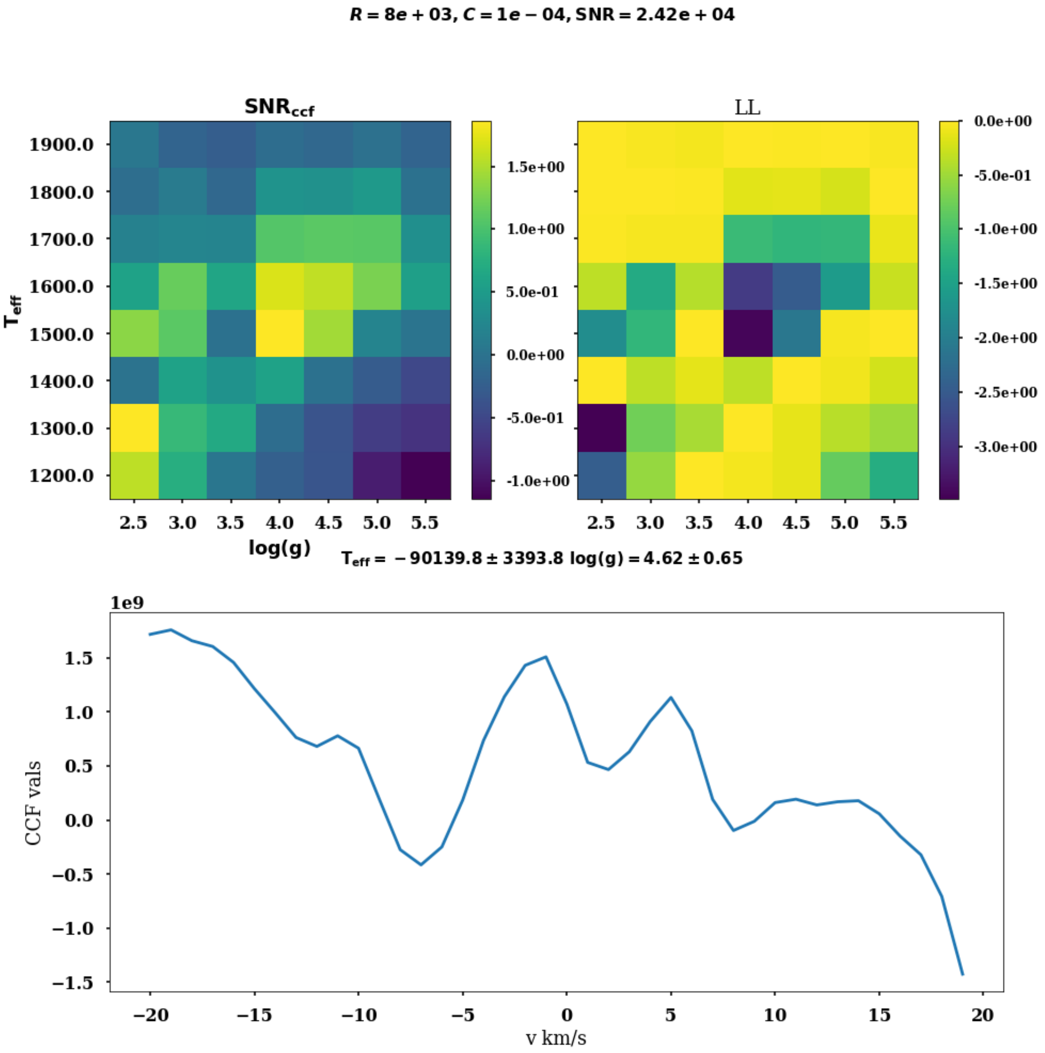


Figure 4.3: Caption

trough in the LL. However, this does not produce a narrower trough because of the velocity resolution. Consequently, the characterization accuracy does not improve. While the $\log(\mathrm{g})$ is slight above one template resolution unit (0.5 dex) the $\mathrm{T_{eff}}$ has an error bar $\approx 200\mathrm{K}$ which is twice the resolution unit of $100\mathrm{K}$. This error bar also is quite close to the error bar at SNR where the exoplanet is just detected. This shows that increasing SNR does not produce a more accurate characterization, whereas for detection a higher SNR does produce fainter detections.

Finally, to answer the question of whether increasing the resolution will change this behaviour, we produce the same plot for a higher R with similar SNR_λ. Note that we lower the C so that it still continues to be comparable in terms of $\mathrm{SNR_{ccf}}$ in Fig 4.6. It is, thus, clear that there is a theoretical limit to how accurate a characterization can be. This is not influenced by any parameters beyond the fact that the exoplanet is detectable in the spectrum. It is also interesting, that the width of the Gaussian of the cross correlation does

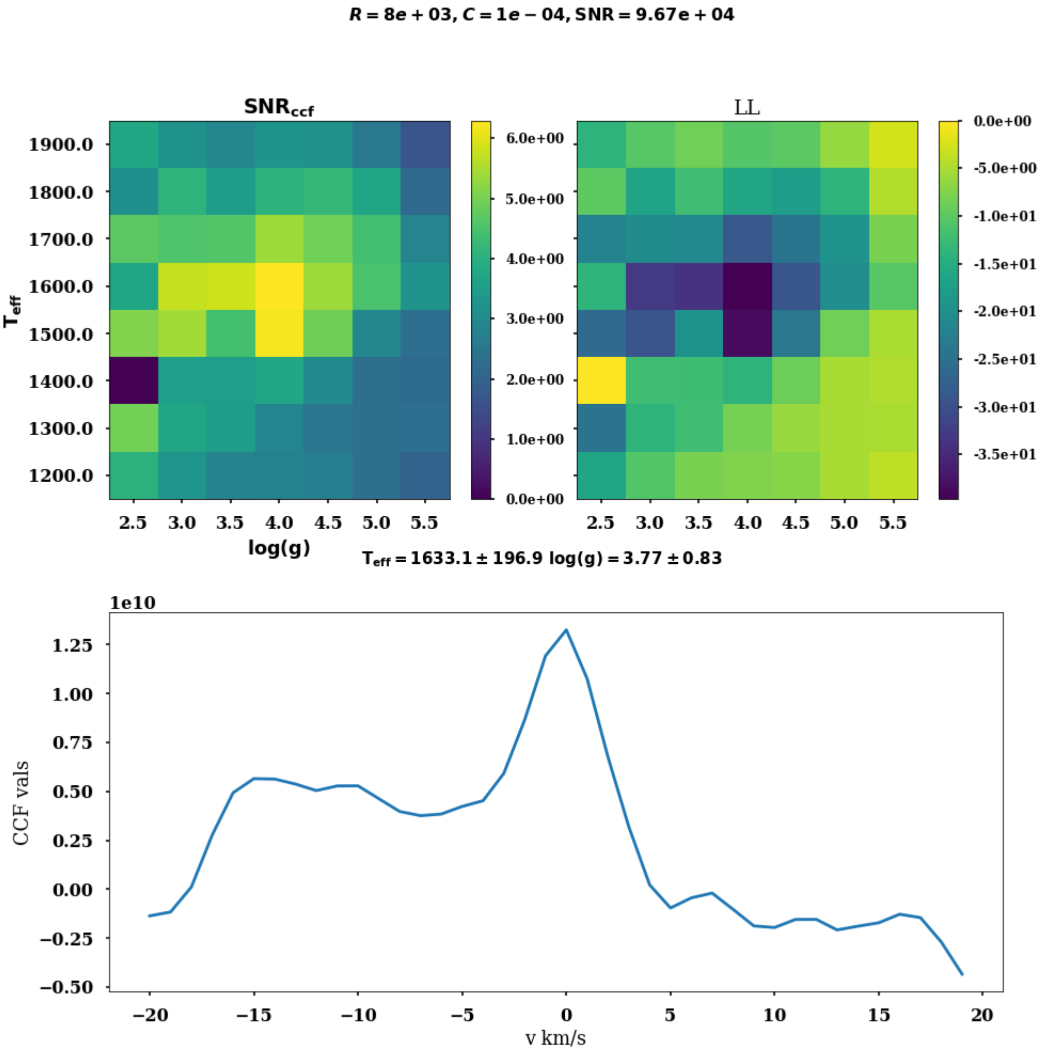


Figure 4.4: Caption

not get narrower and consequently it appears that the width of the LL is also equally wide. This width as we have seen defines the σ_x which is our error bar. In the discussion section of this chapter we will define the benchmark that the cross correlation algorithm sets for the ML algorithms and what steps the ML algorithms need to pass to be considered usable for scientific analysis.

4.3 Benchmark for ML algorithms

In lieu of having a discussion section, we will discuss the results that were produced using the cross correlation algorithm and its consequences for the detection and characterization hypotheses. Given these consequences, we will then define what the benchmark for the ML algorithms are. Finally, we will discuss how these consequences could help the scientific

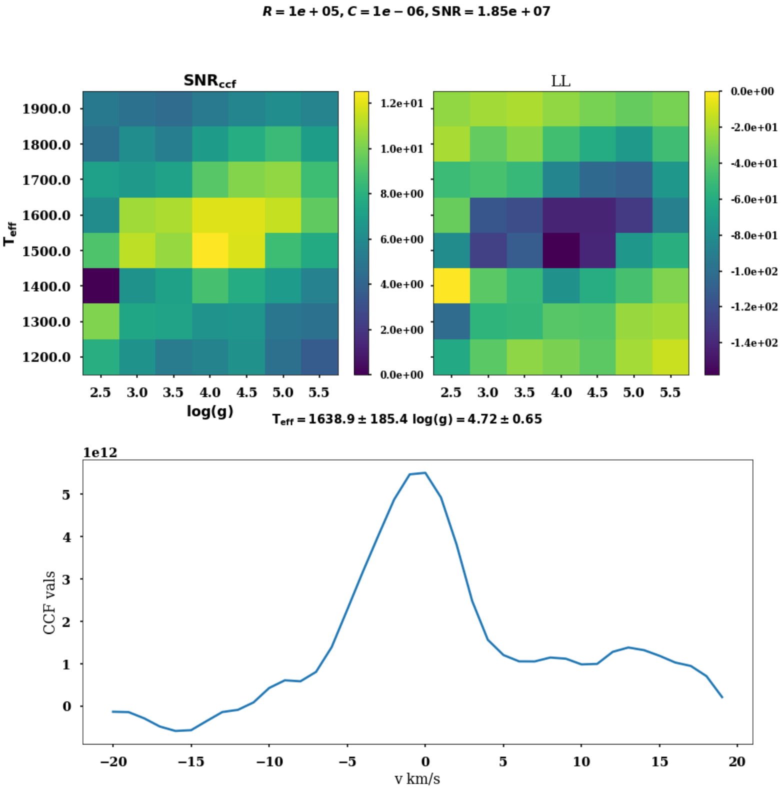


Figure 4.5: Caption

community.

4.3.1 The detection hypothesis and the relevant benchmark

The detection hypothesis states that given the appropriate spectral parameters, it is possible to detect an exoplanet with unlimited sensitivity. We define sensitivity as the ability to detect faint exoplanets while having $< 10^{-4}$ average false positive rate. We then composed a detection matrix to quantify this sensitivity as a function of the observation parameter SNR and the instrumental parameter R. Within this detection matrix we filled the contrast at which the exoplanet was detectable in the spectrum for an $\mathrm{SNR}_{\mathrm{ccf}} \geq 6$ for a defined SNR and R. We found that R had very little effect on the sensitivity of detection, but beyond a $\mathrm{SNR} > 10^5$ the exoplanet at the lowest contrast was detectable at all values of R. This result has some interesting consequences firstly for the next step whereby it will set the benchmark for the ML algorithms and secondly for the scientific goals of using exoplanet spectra in general.

Setting the benchmark for ML algorithms:

The question that this part of my thesis seeks to answer is whether ML algorithms, with their ability to learn patterns, can learn a generalized pattern corresponding the features of

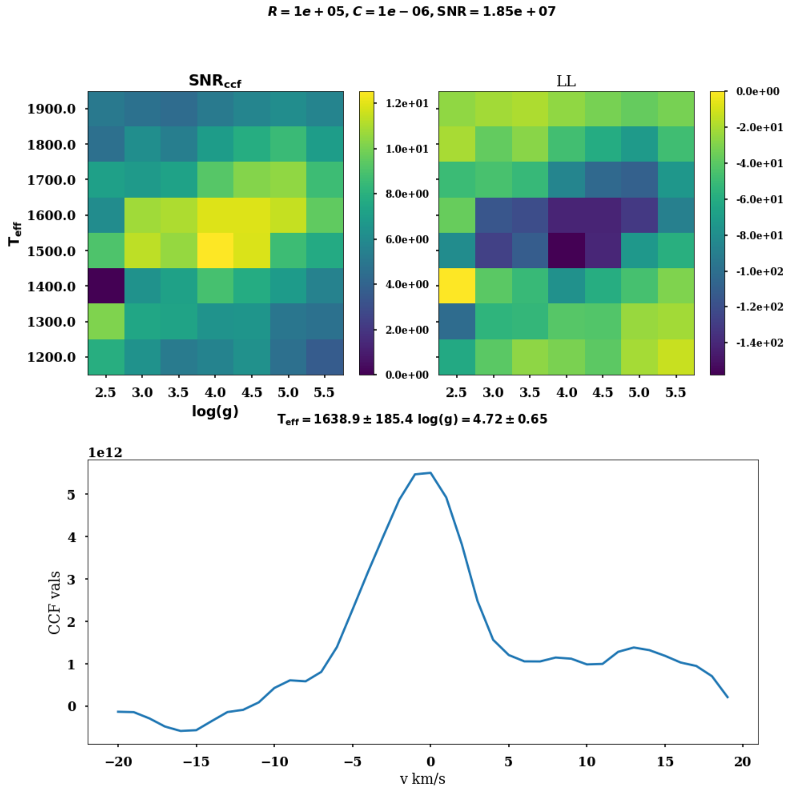


Figure 4.6: Caption

exoplanet spectrum to identify those spectra which contain them. We define this problem to be complicated by three broad parameters, C, R&SNR. The detection matrix allows us to establish where in the parameter space are traditional algorithms most sensitive to the exoplanet. We have now established that to have the highest sensitivity, the cross correlation algorithm requires a minimum $\mathrm{SNR} = 10^5$. This then implies that there definitely is enough planetary features which could be matched with a template. The detection matrix also justifies that R has very minimal to no impact on detection sensitivity. Therefore, the first benchmark goal for ML algorithms would be two fold,

1. within the parameter space that the cross correlation detection algorithm achieves its highest sensitivity, the ML algorithm also needs to perform as well and this will be quantified using confusion matrices to begin with and

2. if ML algorithms achieve a mean false positive rate $< 10^{-4}$ and true positive rate > 0.5 in this parameter space, they will be used to construct a similar detection matrix.

This step would validate that ML algorithms are indeed a reasonable replacement for the cross correlation algorithm. This means that ML algorithms through this methodology can be used to estimate the sensitivity of datasets. Finally, for ML algorithms to be considered better than cross correlation based algorithms it needs to satisfy the following conditions, in addition to the above,

1. they need to be more sensitive to fainter companions, with the same false positive rate, at lower SNR and
2. to be just as flexible with increasing R with the same sensitivity.

Consequence for the scientific goals of spectral data processing to detect exoplanets:

The detection matrix allows us to estimate the parameter space that is ideal for detection sensitivity to be maximized. It also allows us insight into certain parameters that limit the detection sensitivity. One of the key goals of scientific data collection and processing is to identify the values within parameter space that allows us to achieve the scientific goals. The detection matrix aims to be the basis to establish these values for specific scientific cases. The scientific case that we make at this point is the detectability of warm Jupiters using spectra at different SNR and R. More specifically, we aim to define at what contrasts are these warm Jupiters detectable. In the case of warm Jupiters, it is clear the SNR is the most important parameter that limits the sensitivity of detection. An interesting observation is that this sensitivity does not change with increasing R where the SNR_λ decreases for a fixed SNR. The detection sensitivity is thus directly related to the overall SNR than to the SNR_λ for each wavelength bin. Therefore, using this matrix it is clear that for $\mathrm{SNR} = 10^4 - 10^5$, which is the mean SNR that is achieved for an observation such as [Refer to observations in Part II HD142527b] we will be able to detect a warm Jupiter at a $C \approx 10^{-4}$.

Another interesting consequence, is the rate at which sensitivity improves in the detection matrix. In Fig 4.1 it was clear that there is a linear relationship between the SNR and $\mathrm{SNR}_\mathrm{ccf}$ particularly beyond $\mathrm{SNR}_\mathrm{ccf} \geq 5$. This means that beyond this value an increase in SNR produces a an increase on $\mathrm{SNR}_\mathrm{ccf}$ in the detection matrix as it is composed of only contrasts where the $\mathrm{SNR}_\mathrm{ccf} > 6$. Thus for an increase in SNR we should logically see a uniform increase in the detected C. However, what we observe is that this increase is not uniform, whereas it increases quite rapidly for lower contrasts $C < 10^{-4}$ but increases much slower for higher contrasts. This is a very interesting behaviour which seems to indicate that for this sort of data detections beyond 10^{-4} may be difficult to pass. We will see this progress in more detail in [Refer Part II Chapter results].

Finally, we see that for increase in R there is little to no impact on detection. This is particularly interesting, because we would expect that for a constant SNR an increase in R will produce a drop in SNR_λ. This means each of the wavlength bins have fewer signal photons and therefore the cross correlation strength should reach $\mathrm{SNR}_\mathrm{ccf} = 6$ only when more overall photons are available (i.e at higher SNR. But it appears that overall SNR (which is constant along a row) is the only aspect that influences detection and not SNR_λ. We posit that this is the case because of the large number of wavelength bins available, the mean SNR is a bigger factor than individual photons in individual bins. Thus the advantage of higher resolution can be seen for this problem with not losing sensitivity to detection. This might have been a very good case for higher resoluion instruments if the characterization showed improved results with resolution.

4.3.2 The characterization and its relevant benchmark

Characterization, in this thesis, is defined as being able to constrain the $\mathrm{T_{eff}}$ and $\log(\mathrm{g})$ of an exoplanet with consistent error bars. Our characterization hypothesis states that we should be able to characterize any spectrum where an exoplanet exists. The word exists is precisely quantified by the detection algorithm as having a $\mathrm{SNR_{ccf}} \geq 5$. This is evidenced when we use the characterization matrix on spectra at fixed R, C and varying SNR. We notice that the $\mathrm{T_{eff}}$ constraints become consistent after after an $\mathrm{SNR_{ccf}} \approx 3.5$. It is, therefore, clear that the quantification of the characterization parameters are heavily influenced by the ability to detect exoplanet spectra.

We designed a characterization matrix with two things in mind,

1. to constrain the mean $\mathrm{T_{eff}}$ and $\log(\mathrm{g})$ of a spectrum with an exoplanet along with the independent uncertainties of both these quantities and also,
2. to ensure that we are able to do this for all spectra where a exoplanet spectrum can be detected.

The goal of this matrix is not to quantify the evolution of the error bars or the mean of the $\mathrm{T_{eff}}$ and $\log(\mathrm{g})$ with spectral parameters such as SNR and R unlike the detection matrix. However there are still some interesting points to note and these points will define the characterization goals for ML algorithms. This involves three broad ideas,

1. the evolution of the characterization accuracy,
2. the impact of the stellar subtraction on characterization accuracy and
3. the intimate relationship between detection and characterization.

Evolution of the characterization accuracy with the parameter space:
When comparing the error bars over a parameter space we will use the detection matrix in Fig 4.2 for two purposes,

1. to make the link between *detectability* and accurate characterization of the exoplanet spectrum
2. and secondly to demonstrate the handy of the detection matrix when describing the evolution of science goals (in this case detection and characterization) with the parameter space.

When we look at the evolution of both μ_x and σ_x in the characterization matrix in Fig 4.3 to Fig 4.4 the first point is the evolution of the μ_x i.e the mean $\mathrm{T_{eff}}$ and $\log(\mathrm{g})$ does not got any closer to the template $\mathrm{T_{eff}} = 1500\mathrm{K}$ as we move along the rows in the detection matrix. This property is also true for the $\log(g)$. This is a marked diversion from the behaviour of the cross correlation algorithm when used as a detector. While moving along rows we detect fainter exoplanets, it does not seem to translate that this implies the template is closer (i.e accurate mean) to the one that was inserted. As we move further along the rows and we increase the SNR to more than double, we still see little to no effect on the detection accuracy and it appears that even now there is no way to have a perfect characterization. This once again marks a departure with detectability where as we move to unphysically high SNR values in the top rows we get an almost uniformly perfect sensitivity to detection.

Following this line of thought, the evolution in the $\sigma_{\mathrm{T_{eff}}}$ and $\sigma_{\mathrm{log(g)}}$ is also unsurprising. However, what is surprising is the fact that error bars remain for as large as two grid points for the $\mathrm{T_{efF}}$ and slightly more than one grid point for $\log(g)$. The question, therefore, is what is producing this large error bar. The fact that even with higher R this problem is not alleviated shows that intrinsically there is an uncertainty produced by a single quantity left untouched so far i.e C. When we reproduce the same plots for $C = 10^{-1}$ which is not only physical but does not have any other detection analog, we see that we get near perfect characterization. An example is shown if Fig 4.7. We also show the fits to convince ourselves that indeed this fit is an inverse Gaussian fit and the contrast is the parameter the produces a more accurate $\mu_{\mathrm{T_{eff}}}$. But even though the mean $\mathrm{T_{eff}}$ is accurately retrieved there is a fairly large error in the $\log(\mathrm{g})$ characterization. But from this map it is fairly clear what the reason for the inaccurate $\mathrm{T_{eff}}$ characterization is. If indeed the characterization accuracy and thereby its error bars are constrained by the presence of stellar contaminants, would there be a better accuracy when the star is switched off? Talk about the ML characterization goal

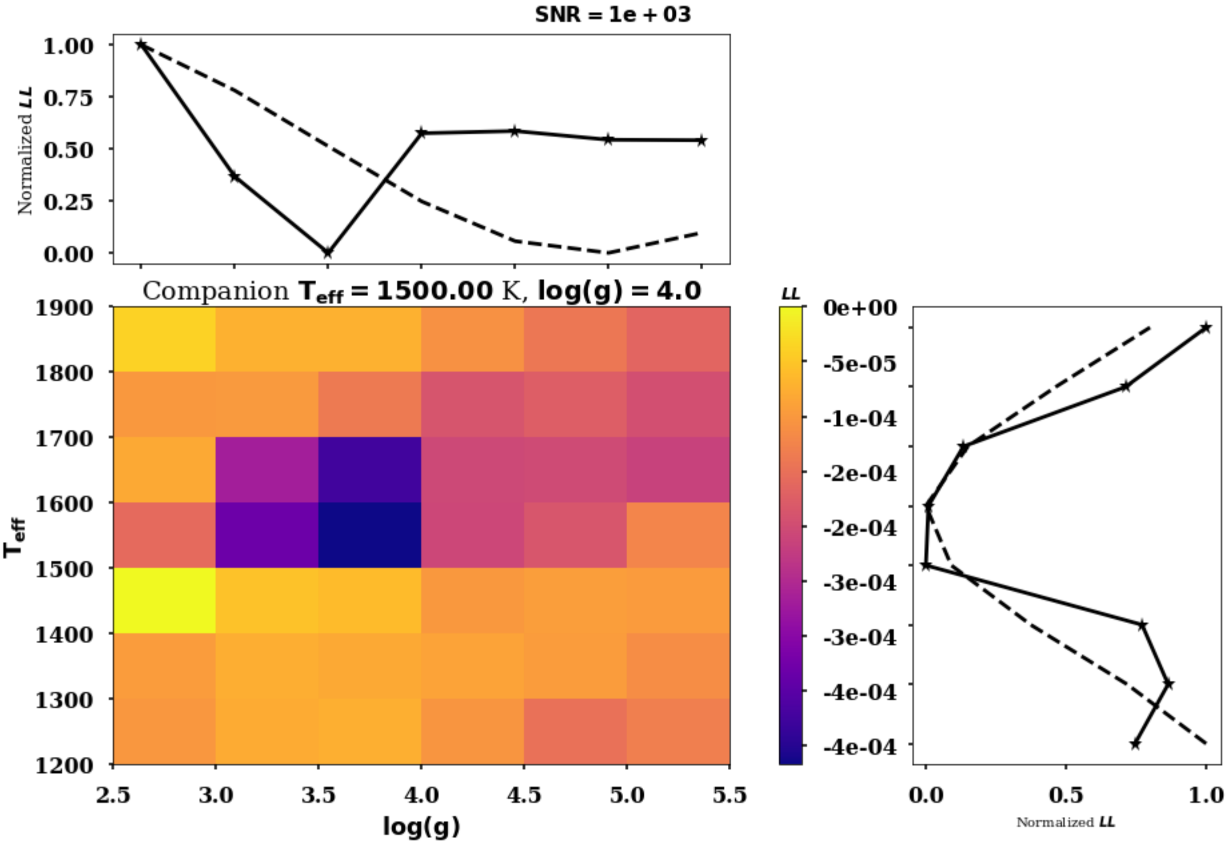


Figure 4.7: Caption

Effect of perfect removal of stellar signal from the data:

While this thesis relies on limited pre-processing and is indeed tuned to removing any stellar features from the data, no pre-processing is perfect and leaves some residual stellar contamination in the data. An easy way to test this is to 'switch off' the star and this is fairly easy to do with synthetic data by setting $C = 1$ in Eq 3.5. This would be the case where the stellar signal is perfectly subtracted. Note that the data still contains observation noise and contains the intrinsic randomness of that noise. We computed different characterization matrices for such spectra for different SNR and R (note that $C = 1$ is now a constant). Fig 4.8 shows the evolution of the estimated $\mu_{\mathrm{T_{eff}}}$ and $\log(\mathrm{g})$ for different values of SNR and R. This is a very interesting plot as the legend seems to indicate that there are different SNR values but the plot just shows one color. The reason for this is that the different SNR values perefectly overlap with no difference in $\mathrm{T_{eff}}$ and $\log(\mathrm{g})$ estimation. The dotted line indicates the original $\mathrm{T_{eff}}$ and $\log(\mathrm{g})$ of the exoplanet. There are at least two interesting takeaways from this plot,

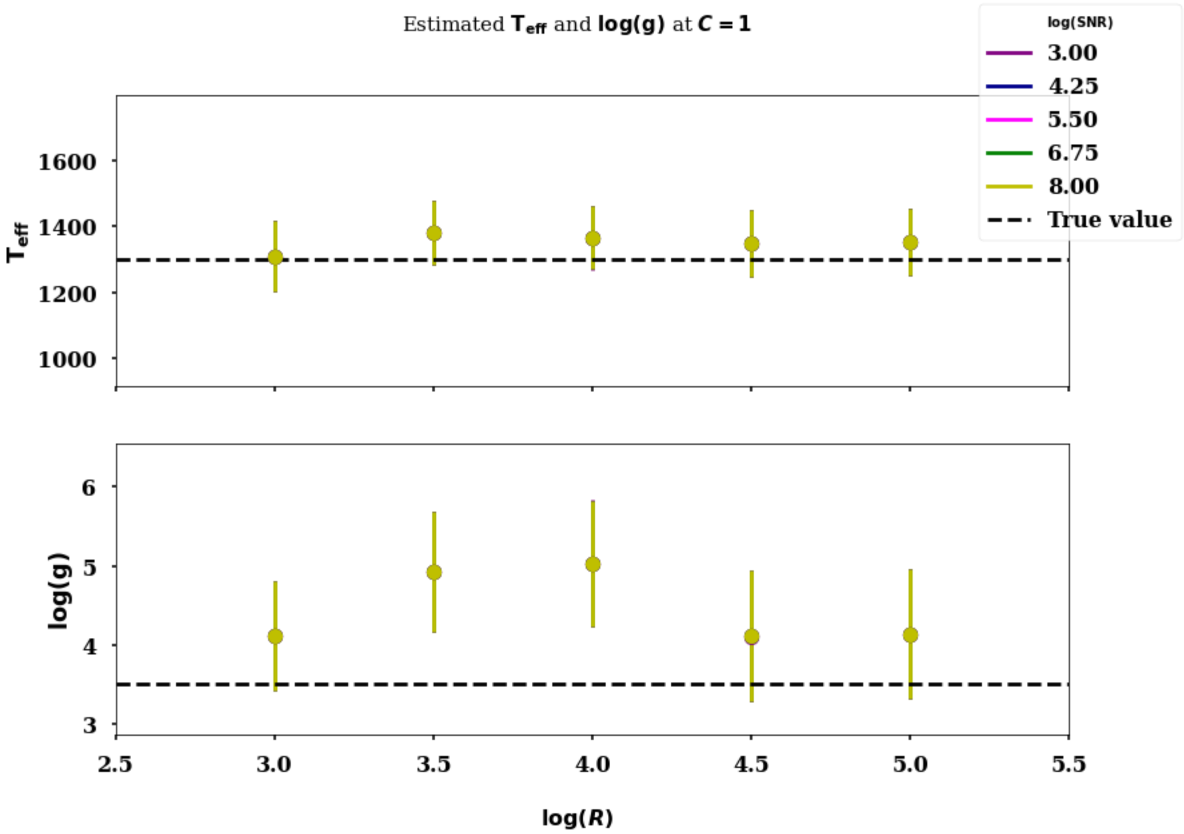


Figure 4.8: Caption

1. unlike the detection matrix, the effect of SNR is absent but the contrast and perfect stellar subtraction is the biggest reason for inaccurate characterization. This is quite interesting because the cross correlation based detection seems to behave as a signal processing problem where with higher SNR we have perfect detection. However, the characterization of an exoplanet seens to behave as an astronomical processing problem whereby we need perfect stellar subtraction to achieve perfect characterization.

2. Secondly, we notice that while the $\mathrm{T_{eff}}$ is perfectly characterized and indeed is defined by a nice LL curve, the $\log(\mathrm{g})$ is not so well characterized. In fact when we see the evolution of characterization accuracy we see the effect of increasing R upto an $R = 10^4$ after which this effect is take over by the presence of more absorption lines.

The second point is fairly well defined in that based on Eq 2.2 the SNR_λ will decrease with increasing R for a fixed SNR and therefore when SNR_λ is the defining parameter to achieve a scientific goal we will see this mitigated with increasing R. Characterization seems like one such goal where the best characterization seems to occur at low R and high R and in between we see a loss of characterization accuracy. This particularly more evident in the case of $\log(\mathrm{g})$ where we work with a smaller baseline and therefore fitting a LL curve is not that straightforward.

This leads us to the limitations of this method and how we expect the ML algorithms to perform better than this. The primary limitation is the need to have a more accurate characterization method whereby the characterization matrix can be populated accurately over a larger range of contrasts. The error bar on this characterization matrix needs to be stable over multiple SNR. However, as we have seen with the cross correlation algorithm, this is not an easy task with spectra. Therefore,the first benchmark for ML algorithms is to identify the region in parameter space of R, SNR where we are able to characterize an exoplanet spectrum with the lowest error for $C < 10^{-2}$. This can the be followed by higher contrast characterizations if it is successful. Note that this is subject to the condition that ML algorithms are able to make high contrast detections, i.e they are able to learn the spectral features of exoplanets at various contrasts.

Link between detection and characterization of spectra:

An outcome of the characterization has been that in many ways the characterization accuracy is unrelated to the detection sensitivity. However, both detection and characterization are being performed with the same spectra and the same algorithm that relies on the same spectral features. Therefore, the question that remains to be explored is whether there is indeed any link between the two operations and if so why does this link disappear so that characterization accuracy does not improve in a lockstep manner with detection sensitivity.

Firstly the link between detection and characterization is most evident when we look at Fig 4.3 and the equivalent detection sensitivity in Fig 4.2. It is clear that when the cross correlation algorithm is not able to detect the exoplanet with the related C, the characterization accuracy is very low. To produce the detection matrix we need to have the following relationship satisfied

$$\frac{\mathrm{CC}(0)}{\sigma} \geq 6 \tag{4.14}$$

allowing that the AC is mostly a constant value and the variation is the σ and the $CC(0)$. At $CC(0)$ there is no relative wavelength shift between the spectrum and the template. Therefore based on [refer to Eq from Part II], we can rewrite it as ,

$$CC = \sum_{\lambda} F_{\lambda_1,\mathrm{noisy}} M_{\lambda_1} + F_{\lambda_2,\mathrm{noisy}} M_{\lambda_2} + \cdots + F_{\lambda_{\mathrm{N}-1},\mathrm{noisy}} M_{\lambda_{\mathrm{N}-1}} + F_{\lambda_\mathrm{N},\mathrm{noisy}} M_{\lambda_\mathrm{N}} \tag{4.15}$$

and therefore we can now re-express the condition with the simplification that $\sigma = \mathrm{SNR}$ as

$$\frac{\sum_{\lambda} F_{\lambda_1,\mathrm{noisy}} M_{\lambda_1} + F_{\lambda_2,\mathrm{noisy}} M_{\lambda_2} + \cdots + F_{\lambda_{\mathrm{N}-1},\mathrm{noisy}} M_{\lambda_{\mathrm{N}-1}} + F_{\lambda_{\mathrm{N}},\mathrm{noisy}} M_{\lambda_{\mathrm{N}}}}{\mathrm{SNR}} \geq 6 \tag{4.16}$$

Thus, for a single entry in the detection matrix the sum of the products of the template and the spectrum has to be $\geq 6 \times \mathrm{SNR}$. Thus the sum of products in Eq 4.15 can be expressed as a limit of the SNR which explains why for increase in SNR we see an increase in detection sensitivity. Thus detection sensitivity can be re-interpreted as the minimum value of the product in Eq 4.15 for which the sum of the photon values in each bin with the template is high enough to be detected. Thus it serves as a mimimum criterion for characterization and hence for values where $\mathrm{SNR} < 5$ the characterization accuracy will provide very unstable errorbars. This is then the fundamental link between detection and characterization. However, why is there not an unlimited improvement to characterization accuracy to reach perfect characterization?

In order to achieve perfect characterization we see that we need to have $C \approx 1$. In such a case we will re-write Eq 3.6 as,

$$F_{\mathrm{total},\lambda} = F_{\mathrm{planet},\lambda} \tag{4.17}$$

and thus the the noisy spectrum is re-written as,

$$F_{\mathrm{noisy},\lambda} = \mathrm{random}(\mathrm{PMF}(F_{\mathrm{planet},\lambda})) \tag{4.18}$$

which means the noisy spectrum does not contain random values from the stellar spectrum. Therefore, when LL is calculated there needs to be higher specificity to template features to produce a deep LL trough. This is provided only when $C \to 1$ than to $C \ll 1$ which is the case for higher contrast. Hence, the characterization accuracy is no longer dependent on SNR but on the intrinsic exoplanetary signal present in the spectrum which is described by C rather than SNR. Naturally, the characterization accuracy would not boundlessly increase as the C is bound for a portion of the detection matrix. On the other hand when we de-link it from the detection matrix we find that we are able to achieve perfect characterization. Thus the detection and characterization are only related until the point when the characterization itself is possible but not to increase the characterization accuracy which is independent of detection.

4.3.3 Why would ML algorithms be an asset if they work?

The goals for the ML algorithms is thus two fold,

1. define the limits at which ML algorithms are able to detect warm Jupiters in either context of the detection matrix or om a part of the detecton matrix parameter space and

2. define the characterization accuracy for this part of the of the detection matrix generated by the ML algorithms.

Before we explain why this thesis posits ML algorithms to be an asset should they work, we need to explain the reasons to test ML algorithms in the context of this chapter. Firstly, the cross correlation algorithm as is defined in this section is limited to processing the data with one sample spectrum at a time. The results however, have to be analyzed for more than one spectrum at a time, for instance we take 10 noise realizations for each cell of the detection matrix and almost 50 different templates have to be cross correlated to produce one characterization matrix is produced. This is a perfect case for batch processing, but also statistical variations between spectra needs to be handled better than by mere averaging. Secondly, there is no unified model for warm Jupiters that is discerned with the use of cross correlation, each template spectrum is treated as a unique spectrum whereas in reality the variation between spectra is not so unique as typified in the characterization matrix in Fig 4.5 where $\mathrm{SNR}_{\mathrm{ccf},\mu}$ is still > 3. This implies that when sufficient photons exist, all the templates provide enough similarity with test spectrum to produce a detetion. Finally, this algorithm is still limited by the interpretation of the cross correlations through a $\mathrm{SNR}_{\mathrm{ccf}}$ or LL which have their own set of biases and therefore, provide only a analysis bias limited result.

This then provides the framework for ML algorithms to be, if successful, a nice alternative to the cross correlation based algorithm. Firstly, ML algorithms are able to analyze batches of spectra and provide batch outputs and thus can populate the detection matrix much faster and with a generalized manner. The need to train ML algorithms is a feature which allows us to train the algorithms with physical parameters that are varied. This in turn will produce an ML algorithm that has learned the 'physics' of the spectrum than working purely from the signal processing stand point. We expect that this is more useful for astronomers, than to fine tune the cross correlation algorithms from a signal processing standpoint. Secondly, the fact that the biases of ML algorithms are well quantifiable, by means of receiver operating characteristic curves for example, means that the algorithmic biases can be well understood. The statistical fluctuations can also be well quantified with the help of a large and varied dataset. This means we don't rely only on averaging the results but will be able to limit these statistical fluctuations by controlling the variance in the data. Finally, the use of ML algorithms will constitute a fast, robust and reliable way to detect and characterize the spectrum simultaneously and we will use the parameter space from the chapter to train the ML algorithms on.

Chapter 5

Development and performance of the ML algorithms

In the previous chapter we presented the cross correlation based detection and characterization algorithms along with the results for a clearly defined parameter space. For the training, validation and testing of ML algorithms in this chapter we continue use the same data set as in the previous chapter. The advantage of being able to generate a practically unlimited number of spectra allows us to extend our scope to try to cover as much of the top five rows of the detecton matrix as possible. We confine ourselves to using the highest SNR cases for two reasons,

1. the top rows allows us a a full range of contrasts to test if the faintest exoplanet detected by the cross correlation algorithm is still reachable by the ML algorithms
2. and to ensure that the lack of photons does not limit our ability to test ML algorithms in the field of high contrast exoplanet detection. This would serve as the best opportunity to test the ability of ML algorithms with a large observing baseline.

The results of that chapter will serve as the basis to develop the ML algorithms used in this chapter. In this chapter, we describe the ML algorithms that are used for this part in my thesis. The goals of this chapter are the following,

1. describe the parameter space and the motivation for the use of parameter space based on the results of the previous chapter. This will form the "data" description in this chapter.
2. Describe the motivation and development of the ML algorithms that will be used in this part of the thesis. The background description of these algorithms are assumed from §2.
3. Describe first the results of the cross correlation algorithm evaluation using the confusion matrices described in §3 followed by the algorithm evaluations of the ML algorithms used.

4. Finally, we end this chapter with a discussion on why the ML algorithms were not used in further science evaluations and its consequences for the use of spectra with ML algorithms.

5.1 Experimental parameter space selection and testing they hypotheses

The parameter space begins with identifying the combination of $(R, \mathrm{SNR}, \mathrm{C}$ for the purposes of training, validation and testing the ML algorithms. The goal of this exercise is to primarily divide the parameter space into two broad categories,

1. identify cases that would serve as the basis to evaluate the ML algorithms and "fail fast" if the algorithms need to be marked as not appropriate for this purpose. A failure to make the appropriate false positive bar would lead the algorithms trained to be marked as inappropriate for this problem. A pass at this stage using confusion matrices indicates that ML algorithms are eligible to check for higher contrast cases which will allow us to compose the detection matrix.
2. The second category would be the highe contrast cases where the contrast is high enough that only higher $\mathrm{SNR} > 10^4$ will be able to detect the exoplanets in these spectra. This second category is the tipping point to define if ML algorithms can indeed be compared with the cross correlation algorithm.

In this section we will begin with describing the problem statement that will be used with ML algorithms first for detection and then for characterization. The hypotheses statements still remain valid and therefore they have to be tested with ML algorithms. We will follow this with describing the parameter space that will be chosen for each category and both the scientific and data challenge posted by this parameter space.

5.1.1 Adapting ML algorithms to test the detection hypothesis

In order to test the detection hypothesis, the first step is to pose the problem accurately so that an ML algorithm can be used to test the hypothesis. The detection hypothesis was tested using the cross correlation algorithm by inferring the conditions (namely R and SNR) that are necessary to achieve perfect detection (namely detecting an exoplanet at $C = 10^{-6}$). The condition for detection was set at $\mathrm{SNR}_{\mathrm{ccf}} \geq 6$ due to its low false positive rate, The $\mathrm{SNR}_{\mathrm{ccf}}$ is defined as the single parameter that will allow us to determine the fitness of the exoplanet to be detected. Thus, we have shown that with the detection hypothesis can be proven with the use of the a detection criterion parameter ($\mathrm{SNR}_{\mathrm{ccf}}$), the detection criterion ($\mathrm{SNR}_{\mathrm{ccf}} \geq 6$) and the detection matrix. The detection parameter will continue to remain the contrast at which we meet the detection criterion when applied to the detection criterion parameter. Following this recipe, we will also define the first two criteria for ML algorithms when validating the detection hypothesis. The detection matrix remains the same to be used for both the ML algorithms and the cross correlation algorithm.

Detection criterion parameter:

The ML algorithms are able to provide two types of outputs as discussed in Chapter 2 namely categorical or a regression output. We explained that when the classes of the output are well known the categorical or classification problem is best chosen. In this case we can have two possibilities for each spectrum, either it contains features for an exoplanet or it does not contain these features. These categories are of course not well separated because a high contrast exoplanet may not be detectable at all. This was the reason for the detection hypothesis, which states that there are indeed conditions where the exoplanet is detectable with an appropriate algorithm. We used the detection matrix to find the conditions where the parameter space separates the spectra as having exoplanet features and therefore detectable even at the highest contrast and not having exoplanet features and therefore not detectable. We use this idea to define the problem of ML algorithms as a classification problem. Given the right conditions the spectra should be clearly separable as containing an exoplanet or not containing one. The detection criterion parameter, therefore is a class of whether an exoplanet exist $y = 1$ or not $y = 0$ and therefore we re-express Eq 3.6 and Eq 3.5 as

$$\begin{aligned} C_y = & \quad y \times C && (5.1)\\ F_{\text{planet},\lambda,\text{new}} = & \quad C_y \times F_{\text{planet},\lambda,\text{old}} \times \text{SNR}^2 && (5.2)\\ F_{\text{total},\lambda} = & \quad (1 - C_y) F_{\text{star},\lambda,\text{new}} + F_{\text{planet},\lambda,\text{new}} && (5.3) \end{aligned}$$

The rest of the equations continue as before from §Chapter 3. We are thus able to generate spectra to be used as training for ML algorithms. As with any classification algorithm, the value of y is universally only 0 or 1 but lies on a sigmoid curve such that when the exoplanet features exist $y \rightarrow 1$ and when they don't exist $y \rightarrow 0$. Therefore, a second criterion to the detection criterion parameter is necessary such that we can quantize y.We choose the criterion as $y = 1$ when $y > 0.5$ and $y = 0$ when $y \leq 0.5$.

In order to understand how the spectral features of the noisy spectrum in Equation (3.5) evolves with different contrasts we present a series of plots where the contrast is reduced and this produces a higher SNR_{ccf}. Note that the SNR_{ccf} is sensitive to changing C exponentially, whereas it is related to SNR linearly. Figure 5.1 shows a spectrum generated with a SNR with two classes, $y = 0$ in pink and $y = 1$ in green. The topmost plot is a sample template spectrum re-sampled to the spectral resolution $R = 1000$. The bottom most plot shows the cross correlation strengths as a function of velocity of both the pink and the green spectra. Note that visually, it is not possible to tell apart the green and pink spectra, but the cross correlation algorithm is able to make this distinction. In order to also illustrate a case where the difference between the pink and green spectra is stark, we choose a very high SNR and low C in Figure 5.2. We can clearly see the shape of the template and its specific features in the K-band and H-band.

Thus, we have the detection criterion parameter y that will serve as the prediction parameter for the ML algorithms with the spectra being inputs.

The detection criterion:

The final part of adapting ML algorithms is to establish the detection criterion that needs

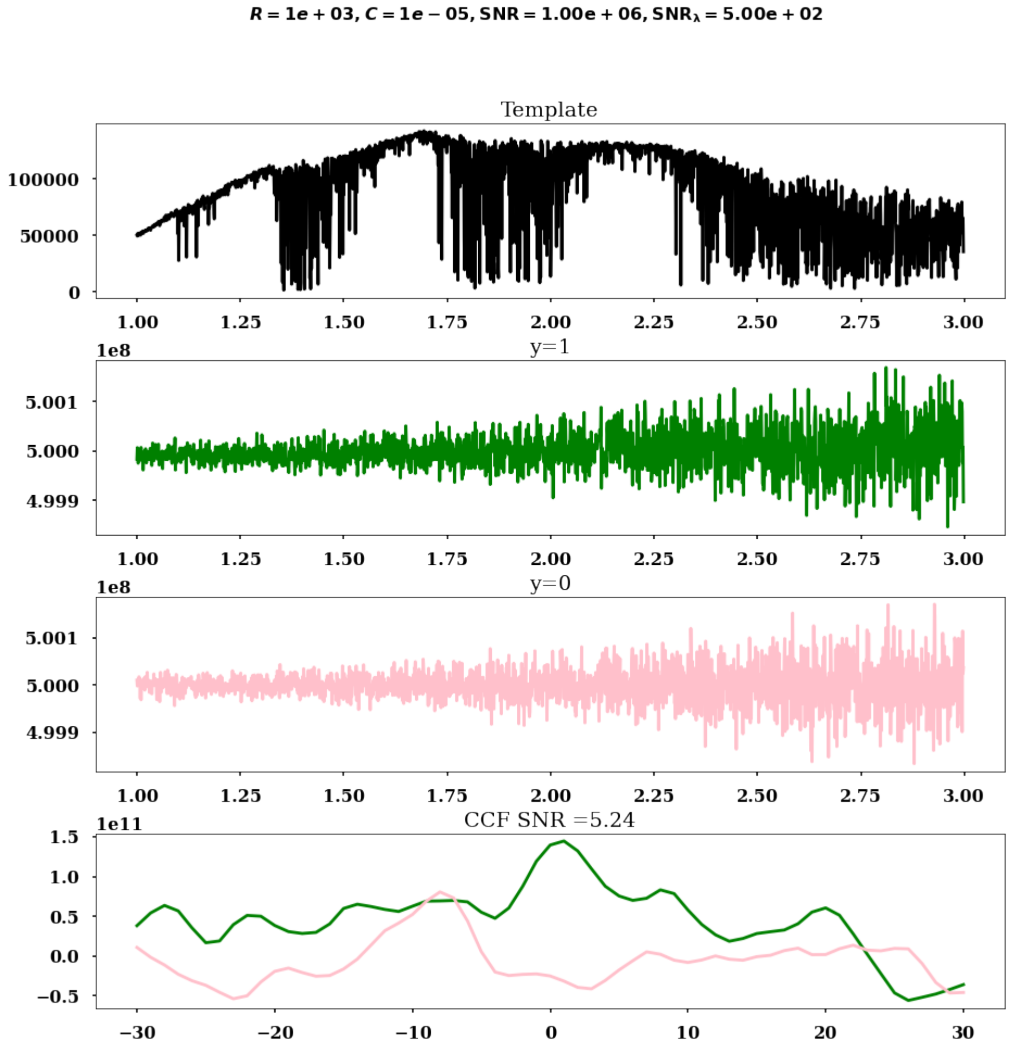


Figure 5.1: Caption

to be met in order to test the detection hypothesis. As stated above when the output of the ML algorithm is between 0.5 and 1, we quantize the output to mean the predicted value of y (y_{pred}) which will then be used to populate the detection matrix. The detection matrix as constructed using the cross correlation algorithm was built by simultaneously analyzing a large number of spectra and then populating the matrix with the results of this analysis. For ML algorithms, as explained in Chapter 2, we need to train, validate and test these algorithms taking care to not overfit the data. A lot of the overfitting will be taken care by providing sufficient variance in the data described in Section 5.1.3. For the purposes of describing generating the detection matrix, it is sufficient to say that we generate over 15000 spectra in total to produce the training, validation and test datasets. The detection criterion still remains the detectable contrast C and we will establish the mean contrast that is detectable for a range of SNR and R, once again chosen with a criterion similar to Figure 4.1.

In order to now establish the detection criterion for a specific parameter values we proceed with the following steps,

1. we first generate a large number of spectra with one specific template with variations being made in SNR and C but a fixed $R = 1000$. We divide these spectra into training, validation and test datasets. We then train the ML algorithms with the training datasets and use the validation dataset to fine tune the following parameters, the learning rate gradient, the momentum, the mix of contrasts involved and the hyper parameters of the ML algorithm being used. The first two are changed continuously during training depending on validation error, but the mix of contrasts is changed so that the validation error is $\approx 10^{-5}$. We then test the algorithms with the test data by producing confusion matrices.

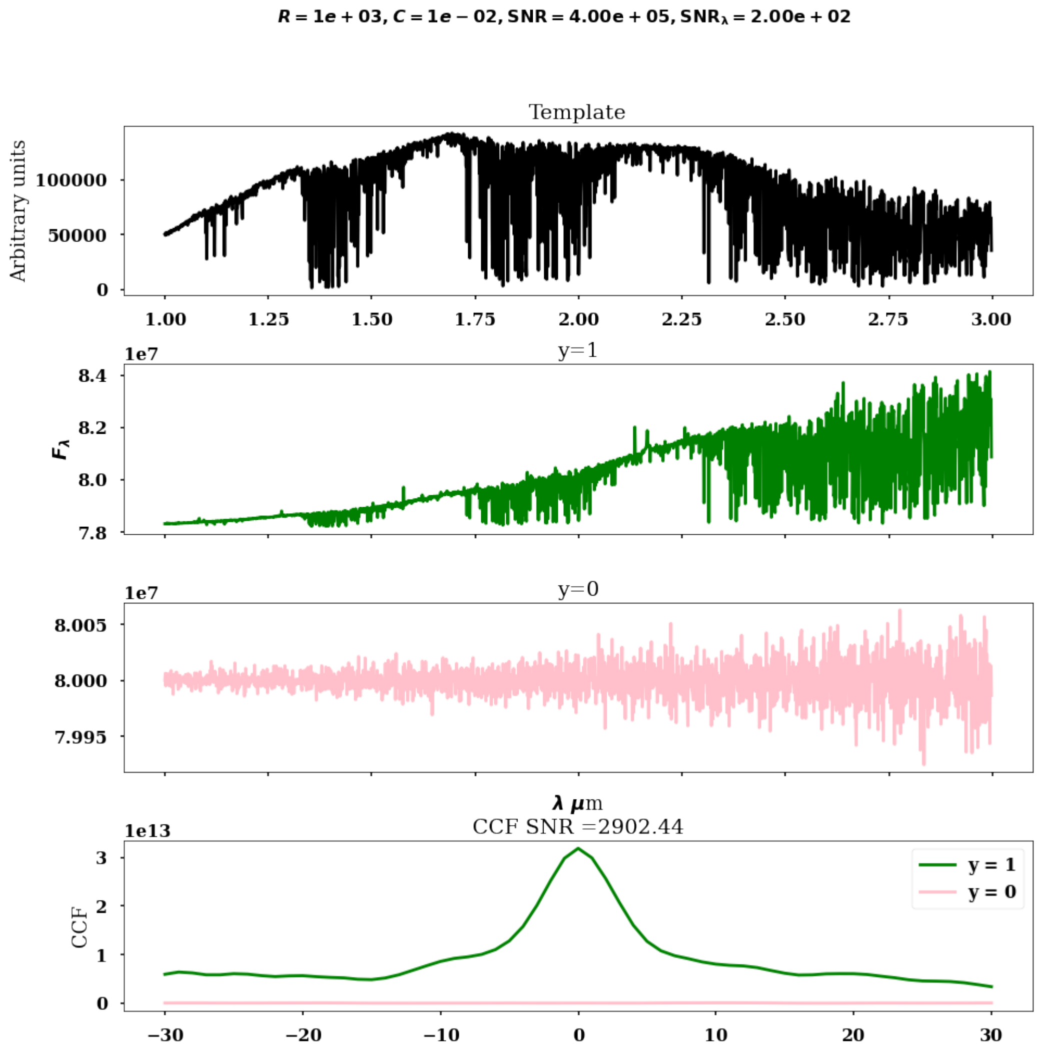


Figure 5.2: Caption

2. When the false positive rate is low enough, we proceed to generate fresh spectra in the contrast ranges used for training (no matter the validation error) and compute the confusion matrices for these. These will be exactly the same as the test confusion matrices but to rule any statistical uncertainties we produce fresh spectra. We use these confusion matrices and identify the highest contrast where we can produce a confusion matrix so that the true positive rate > 0.5 and the false positives produces are 1 for 10000 spectra.

3. We then perform this operation for each of the SNR in the parameter space and identify C to populate the detection matrix.

Note that we will start with highest SNR to ensure that we rule out the possibility of that ML algorithms fail the detection hypothesis.

5.1.2 Adapting ML algorithms to test the characterization hypothesis

The characterization hypothesis is tested using the characterization matrix. As with detection matrix, the characterization matrix is populated by the characterization parameter which was the LL in Chapter 4. The characterization parameter could have also been $\mathrm{SNR}_{\mathrm{ccf}}$, but for reasons explained in Chapter 4 we use LL which is the negative square of the $\mathrm{SNR}_{\mathrm{ccf}}$. We don't have any known literature which connects the value of y_{pred} and any Gaussian distributed parameter that will allow us to derive a χ^2 error bar. Therefore, in this subsection we will briefly describe two different strategies to produce a characterization parameter and the merits and demerits of both. It is to be noted that if the ML algorithms fail the detection hypothesis, there is not enough evidence that they will be able to test the characterization hypothesis.

As already explained in Section 2.1, supervised algorithms are either classification or regression algorithms. In practice this means that either the output of the ML algorithm is completely unbounded or is bounded to remain between 0 and 1. Since we don't have any prior art in this regard, we postulate using both regression and classification as approaches to produce the characterization matrix. The caveat of this section is that neither of these approaches were validated as ML algorithms failed the detection hypothesis, but this section is evidence of clear experimental planning before we explain the ML algorithms.

Use of regression to compute the characterization parameter:

When using regression to compute the characterization parameter, we directly compute the $\mathrm{T}_{\mathrm{eff}}$ and $\log(\mathrm{g})$ from the ML algorithms. The question is how doe we compute the error bars on these quantities. We first define the output of the ML algorithm to be two quantities i.e $\mathrm{T}_{\mathrm{eff}}$ and $\log(\mathrm{g})$ for each spectrum. We then use an appropriate error function such as the mean square error to compute the error on these quantities. As the mean square error will be the quantity minimized, we will have the minimum mean square error on these quantities separately. However, in order to compute the true error bar we had posited the following steps,

1. generate a large number of spectra for a fixed value of (C, R, SNR) so that each spectrum represents only a different noise realization. Based on our results in Section 4.2.3 and Section 4.3.2, it is clear that the biggest effect on characterization error bars is the stellar contamination.

2. fit a non-parametric curve to the results produced from these spectra to compute the means and standard deviations.

3. this will result in one mean and standard deviation for both $\mathrm{T_{eff}}$ and $\log(\mathrm{g})$ for each combination of (C, R, SNR) and

4. finally, we will generate characterization matrices for each value of C, R for several SNR to produce a result akin to Figure 4.7.

This can then be used to verify that the error bars are quantifiable for specific values of C. The advantage of this technique is that we can directly regress to the value of $\mathrm{T_{eff}}$ and $\log(\mathrm{g})$. The disadvantage of this method is that it is possible the distribution of T_{eff} and $\log(\mathrm{g})$ does not lend to accurate error or mean calcuation, in which case it would be pointless. This type of regression does not lend to clean graphical qualification of the performance of different templates. Hence we will also explore a slightly different way of quantifying the mean and standard deviation

Design of a characterization matrix with a logistic characterization parameter:
The goal of the characterization matrix is to compute the mean $\mathrm{T_{eff}}$ and $\log(\mathrm{g})$ and their uncertainties. We also like a nice graphical way that can describe the similarities between different templates. To achieve this we define a new parameter p such that,

$$y_{\mathrm{pred}} = p \tag{5.4}$$

The distribution of p for a single spectrum is Figure 5.3. For every spectrum, we will have a characterization matrix with different p values. The p values are distributed as a 2D Gaussian centered on the $\mathrm{T_{eff}}$ and $\log(\mathrm{g})$ of the template of the exoplanet. A sample map is shown in Figure 5.3 alongside a characterization map with $\mathrm{SNR_{ccf}}$ as the characterization parameter.

We posit that a ML algorithm can learn to recognize feature strengths that vary as a Gaussian centered around the inserted template. The standard deviation of this Gaussian will give us uncertainties of both $\mathrm{T_{eff}}$ and $\log(\mathrm{g})$. We will use a logistic error function such as the categorical cross entropy to minimize the training error. We will generate large number of spectra with constant SNR, R and slightly varying C so that we can have some variance in the data. We will automatically produce these characterization matrices and train the ML algorithms to produce the same matrices for every spectrum. The advantage of this approach is that we get a characterization matrix which has a parameter that can be related to detection. We can also get a very good visual reference for large number of spectra and verify that the error bar is consistent for different types of spectra. Finally, these matrices allow us to infer whether ML algorithms are able to distinguish between templates in one glance. The disadvantage of this approach is that there is no evidence that

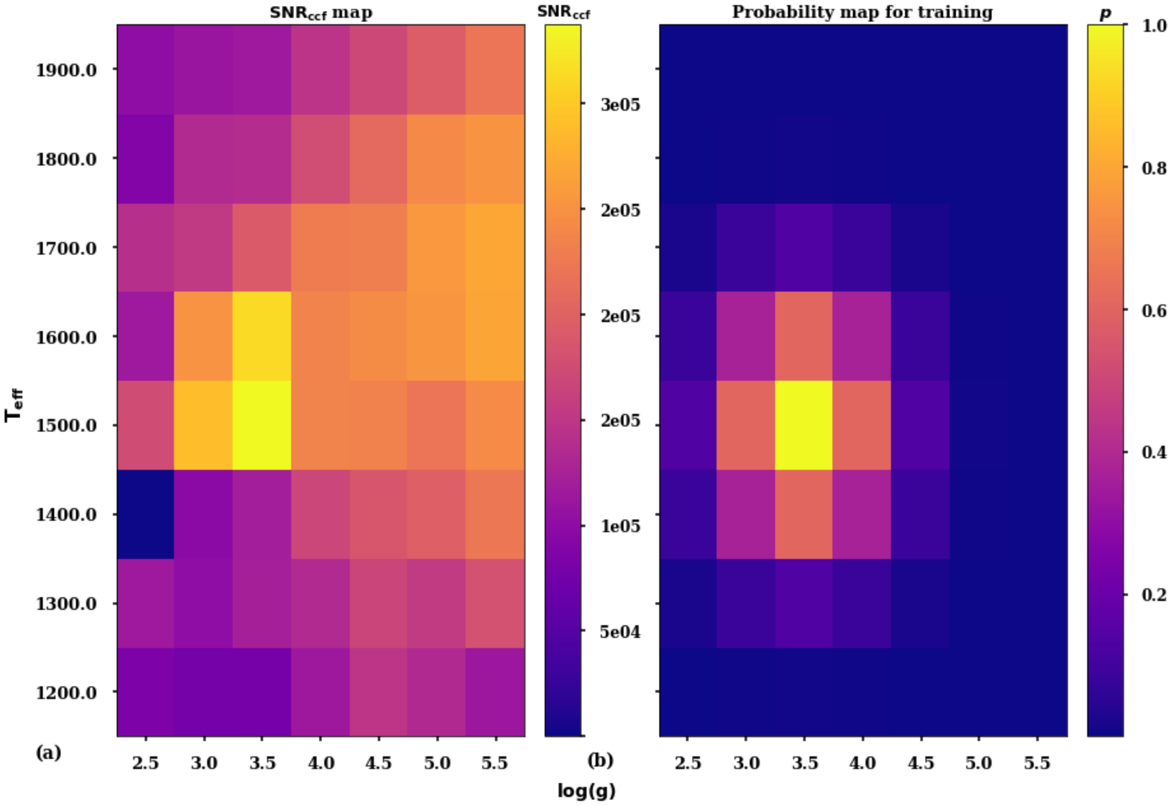


Figure 5.3: Caption

the output of ML algorithms will obey this distribution. There is also little evidence to show that Equation (5.4) holds true for large number of spectra with any bounding condition.

5.1.3 Parameter space selection

The parameter space selection is simplified by the tests performed using the cross correlation algorithm and the resulting detection and characterization matrices. There are two takeaways from these tests that we will continue to work with while choosing the parameter space to train, validate and test the ML algorithms,

- the SNR is the most important parameter to validate the detection hypothesis, in turn this means that for the highest SNR the detection hypothesis has the highest chance of being validated for all contrasts concerned. Note that this does not preclude that ML algorithms will automatically perform exactly as the cross correlation algorithm has performed, but merely chooses the best possible chance for ML algorithms to succeed.

- Secondly, it is clear that detection and characterization are linked through the contrast, where we know that if the contrasts are low and SNR we are able to perform simultaneous detection and characterization. But what needs to be seen if ML algorithms are able to consistently find the spectral features for different contrasts to convince us to use them to characterize spectra.

Point 2 thus raises an important question, i.e will ML algorithms work on both the low and the high contrast space. With this in mind, while keeping $10^5 > \mathrm{SNR} > 10^6$, we divide our parameter space into a low contrast parameter space corresponding to $C > 10^{-3}$ and a high contrast parameter space $10^{-3} > C > 10^{-4}$.

Low contrast parameter space:

The low contrast parameter space is defined where the contrast is between $10^{-2} > C > 10^{-1}$. As described earlier, we choose the highest SNR between $10^5 > \mathrm{SNR} > 10^6$ to give the ML algorithms the best chance to train. We fix the $R = 1000$ for two reasons, firstly that there has not been significant evidence to show that R has much impact on the detection or characterization of an exoplanet particularly when testing our hypotheses. Secondly, higher R means higher number of bins and consequently larger memory and computational requirements. Therefore, we choose a low R to test our hypotheses with ML algorithms. As stated before, we generate a large number of spectra based on Section 5.1.1. In order to keep this fair we also validate the spectra by running them through the cross correlation algorithm and setting a $\mathrm{SNR}_{\mathrm{ccf}} > 5$ to set the $y_{\mathrm{pred,ccf}}$. We sample the parameter space such that the distribution of spectra is fully within the sample space. To illustrate this we plot the spectra generated based on the parameter space, such that a single spectrum generated with a combination of SNR, C is represented as a single point in Figure 5.4. The sample space is filled with three colors pink points corresponding to $y = 0$, black corresponding to spectra undetected by the cross correlation algorithm ($y = 1$) and green points corresponding to those that are detected. The absence of black points in Figure 5.4 shows that there are no spectra which are missed by the cross correlation algorithm. This plot also illustrates that the parameter space is well filled and the balance between $y = 0$ and $y = 1$ spectra is fairly even.

high contrast parameter space:

For the high contrast parameter space we choose a contrast range of $10^5 > C > 10^3$. In principle we could choose the highest contrast, but we also wanted make a parameter space where the true positive rate for the cross correlation algorithm was slightly higher than 0.5. This gives us confidence that majority of the spectra indeed contain features that are detectable by a 'classical' algorithm. At the same time there are a few samples that provide a 'challenge' to the ML algorithms to detect. A similar plot to Figure 5.4 is used to depict the samples drawn from this parameter space. In Figure 5.5 we see some black dots filling up $\approx 30\%$ of the parameter space, whereas the rest of the parameter space is covered by green dots. Note that $\mathrm{SNR}_{\mathrm{ccf}} > 5$ is a somewhat 'high bar' to ensure we don't end up with a high false positive rate and does not mean that those spectra which deliver a $\mathrm{SNR}_{\mathrm{ccf}} \approx 4.5$ are non detectable or that they don't contain features that are detectable.

When reporting the results we will make this discrimination of high and low contrast parameter spaces as the results also are neatly divided in between these parameter spaces. A final word on this parameter space is that while the contrast cut off between the spaces is somewhat arbitrary, there is a clear scientific justification for dividing these contrasts. Firstly, the low contrast cases are what would be really bright targets and therefore, if known they would correspond to the first set of targets that would be used to test new

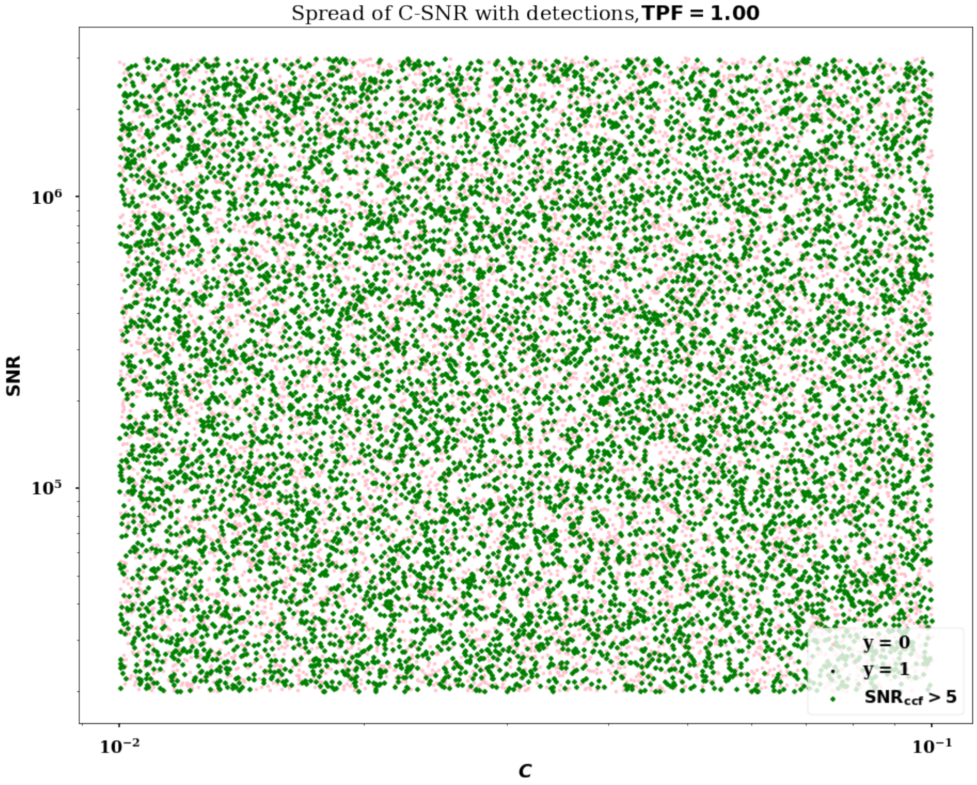


Figure 5.4: Caption

instruments and algorithms. These would also correspond to targets whose properties of $\mathrm{T_{eff}}$ and $\log(\mathrm{g})$ are well defined and so they can be well quantified and verified. In principle, therefore this parameter space represents some of the well defined targets. Secondly, the high contrast space is that which is typical of targets that are the most numerous in the high contrast imaging. Therefore, this represents the targets that ML algorithms will most likely be presented with and therefore the performance on this parameter space is the test of failure of ML algorithms. It is on this parameter that we need to quantify the performance of ML algorithms on.

5.2 ML based algorithms

ML algorithms come in different flavours and types and a brief introduction to this was presented in Chapter 2. As stated earlier, this thesis primarily works with supervised classifiers to test the detection hypothesis and we could choose supervised classifiers or regressors to test the characterization hypothesis. As we shall see in the Section 5.3 the performance of ML algorithms did not provide us with the confidence to try out characterization strategies. Therefore, this section will primarily discuss the algorithms used for testing the detection hypothesis. However, we would be remiss in assuming that these algorithms could not

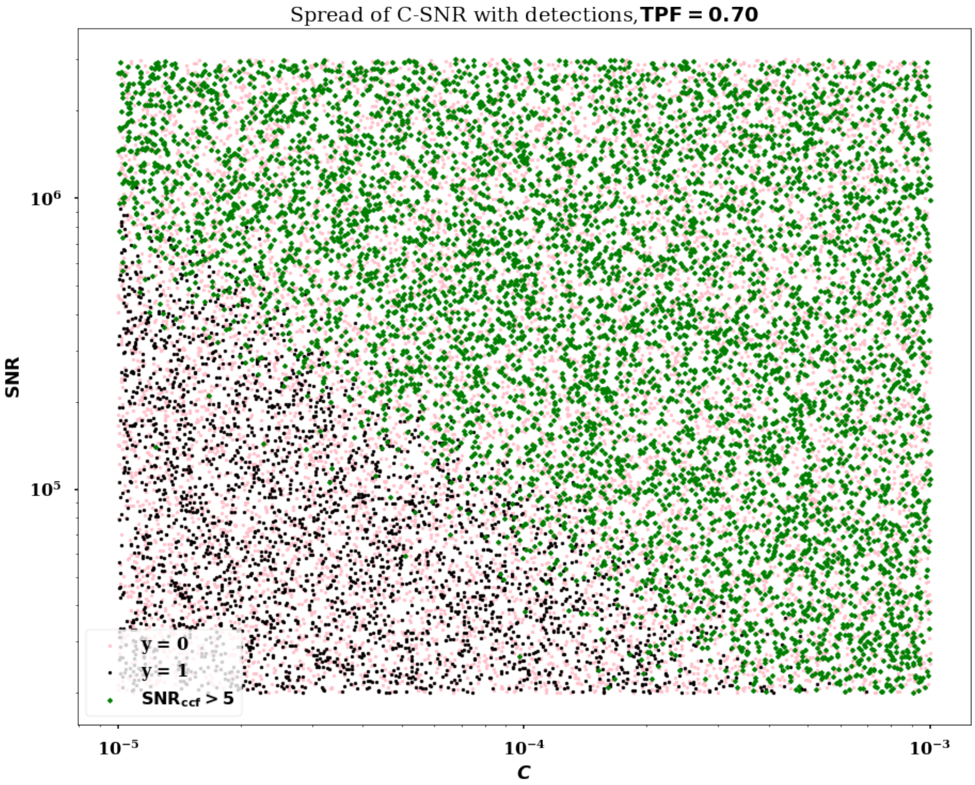


Figure 5.5: Caption

be adapted to test the characterization hypothesis. We will, therefore, also suggest ways to adapt this algorithm for the characterization case. We will discuss later the ways that spectra could be used directly with ML algorithms that is out of the scope of this thesis but could be tried out easily in the future with the framework provided by this thesis.

This is a very crucial paragraph which justifies how what I have done different from 2020 Fisher. Note there are indeed very many similarities and it is possible this is one of the weakest points, that it is not fair to expect any characterization from ML as the Fisher paper exactly reports this. I aim to justify this as 1. I use detection to test if this method can be used with HCI data 2. we then will move to characterization iff we are able to compose the detection matrix with ML algorithms. As described in Chapter 2, supervised classifier algorithms come in different types, in this thesis we use ensemble and deep learning algorithms. The goal of trying two different kinds of algorithms is to verify whether a) are results consistent over different types of algorithms and b) whether any kind of algorithm offers an advantage to this type of analysis. As stated before Fisher et al. (2020) has explored the use of random forests and deep learning algorithms with cross correlation data, which are similar in the sense that they are also large 1D vectors. However, the same algorithms have not worked with spectra directly, when ML algorithms have been used to constrain the metallicity and temperature directly from the spectra. There are two fundamental differences from how we are posing the problem,

1. firstly, we pose this problem as a pure detection problem where we are detecting an exoplanet as an inference based on the presence of an ensemble of exoplanet features in the spectra and

2. secondly, the characterization part of the problem is essentially an extension of the detection where we identify the closest template to the one that is present in the data and use that to infer the properties of the exoplanet.

In addition we choose the parameter space of the spectra after carefully whetting it by running them through the cross correlation algorithm. Finally, we will use the ML algorithms to first the detection hypothesis to define if ML algorithms are able to validate it before validating the characterization hypothesis. Our ML detection and characterization algorithm is a three step algorithm,

1. in the first step we conduct the data generation and normalization where we generate the data specified by the parameters of C, R, SNR this is followed by,

2. the passing of the spectra through the ML algorithms which either detect or could produce a characterization output and finally,

3. this is terminated with a analysis step which is usually either a confusion matrix where we know the truth values or a threshold application.

The final step is the scientific outcome that is relevant to this thesis.

5.2.1 Data generation and pre-processing

We first generate data as in Figure 5.4 for the low contrast parameter space. We choose a range of templates to initially begin with in order to minimize the chance that the ML algorithm will memorize wavelength features specific to the template. As we produce these spectra, we also run them through cross correlation based detection algorithm to ensure that the statistics are inline with Figure 4.2. We generate a total of 12000 spectra within this parameter space. We divide these spectra into ≈ 10000 spectra for training, ≈ 2500 for validation and 500 for testing. We report each of these results in Section 5.3. Our partition of data follows the 80% for training, 15% for validation and 5% to test. The idea being that the validation examples will allow us to provide a rigid buffer against overfitting, at the same time ensuring that we have enough samples to assess if it is underfitting. The test is meant to serve as a failsafe to avoid overfitting the training and validation. Once spectra have been generated, divided into the sample datasets, we first normalize them using the standard scaler which is expressed as,

$$F_{\lambda,\text{norm}} = \frac{F_{\lambda,\text{noisy}} - \mu_{F_{\lambda,\text{noisy}}}}{\sigma_{F_{\lambda,\text{noisy}}}} \tag{5.5}$$

where μ and σ are the mean and standard deviations of the spectrum.

5.2.2 ML algorithms

This section will describe the development of the different ML algorithms starting with the random forest on the basis of the steps of generating data, training the algorithms and validating and testing them.

Random forests:

As stated earlier in Chapter 2,among the ensemble algorithms, random forests have the properties that are most suited to deriving inferences from large vectors. We pose the problem as classification problem to the random forest where the input is a normalized input spectrum and the random forest attempts is trained to classify the result as $y = 0$ or $y = 1$. We then start with the standard number of 1000 trees in the random forest and other default parameters of the SKLEARN implementation. We then progressively increase the number of trees in the forest until we have similar confusion matrices for the training and validation matrices. We will describe the different matrices in results section, but the goal is to have 0 false postives and true positive fraction > 0.5. We found that the best validation matrix is achieved for 3000 trees. We also found that increasing the number of forests, minimum node split etc. had little to no impact on the training loss.

MLP:

In this thesis, we use the MLP to verify whether adding a depth dimension allows our ML algorithm to generalize better and test the detection hypothesis. We started with small neural networks and grew it depth wise, slowly adding depth as the results with validation data on the low contrast dataset. We finally settled on a 11 layer architecture where the activation function is a "Rectifying Linear Unit (ReLU)" with the exception of the last layer which has a sigmoid output to predict the class or another ReLU to predict the value. As before, we first produce spectra which can be processed with the cross correlation algorithm.

We iteratively changed the number of neurons in each layer based on the output of the validation step. Based on the confusion matrices generated from the low contrast dataset, we settled on the following configuration that allowed us to produce identical train, validation and test confusion matrices, $[6000, 3000, 1500, 600, 300, 150, 60, 40, 30, 20, 1]$ for each layer starting from the input to the output class.

Autoencoders:
An autoencoder, as stated in Chapter 2, has been used effectively in classifying stellar spectra. The sparse reconstruction of an autoencoder seems quite ideal for spectra which contain distinct features in just a few wavelength bins whereas the rest of the bins are mostly noise. In our case we use the full wavelength configuration and thereby contain absorption features in each wavelength. In order to use this idea we built an autoencoder with small amount of layers to begin with and then built it up as the results of the validation confusion matrix improved for the low contrast case. We settled on an architecture that consisted of 6 encoding and 7 decoding layers each layer had a ReLU activation. In addition, there was an input layer with ReLU activation with as many neurons as the input vector size N and the output was a single sigmoid neuron to classify the spectra. The autoencoder had a mirrored architecture for the encoding and decoding layers. We start the encoding layer with 1024 neurons, sequentially dividing the number of neurons by 2 until we reach 32 neurons. The decoding layer then starts with 32 neurons and ends with 1024 neurons and one last layer of the same size as the input layer and finally terminating with the output sigmoid.

For both of the deep learning algorithms the loss that we use is the binary cross entropy loss with the ADAM optimization routine. The output of all of these algorithms are subject to two evaluations 1. being the basic algorithmic evaluation where the robustness of the ML algorithms to reproduce the results of cross correlation algorithms will be tested 2. being the scientific validity by constructing the detection matrix using the ML based algorithm.

5.3 Experimental results

The experimental results section of the ML experiments will contain three parts,

1. we will first describe the results of the performance of the cross correlation based algorithm to establish the benchmark of algorithmic performance and to be convinced that the data generated for these experiments were indeed reasonable for the experiment goals.

2. Then we will describe the low contrast results produced by the ML algorithms using confusion matrices

3. and then finally we will end with describing the results produced by ML algorithms when they were trained with high contrast data.

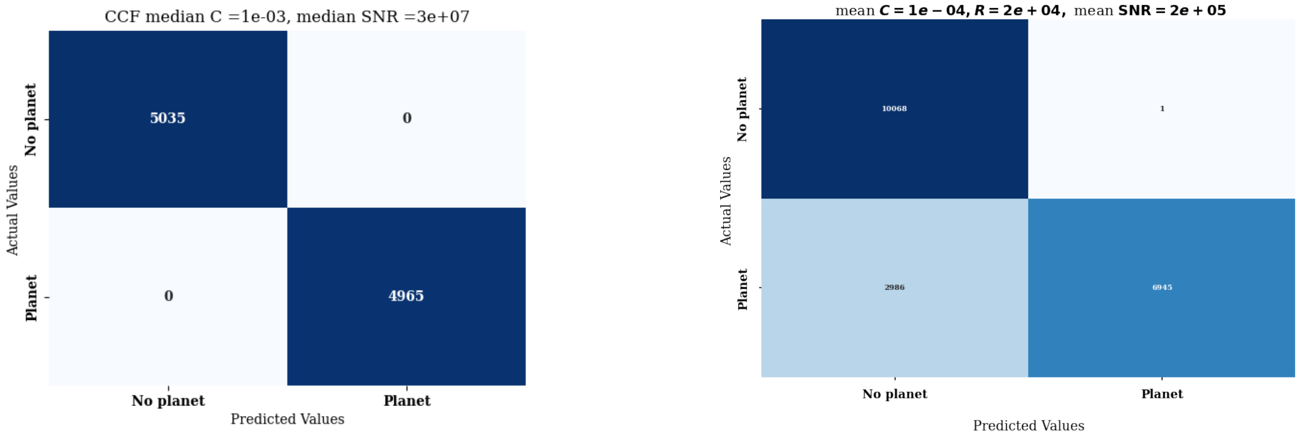
The goal of this section is to clarify the limited scope of ML algorithms but also to showcase that using them in the limited manner will produce somewhat reliable results.

5.3.1 Performance of the cross correlation algorithm as a classifier

The performance of the cross correlation based detection algorithm is defined by the number of false positives and the true positive rate. The true positive rate is given by,

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{5.6}$$

where TP is the total number of $y = 1$ cases that produce a cross correlation with $\mathrm{SNR}_{\mathrm{ccf}} > 5$, FN are the total number of $y = 1$ cases that produce a cross correlation with $\mathrm{SNR}_{\mathrm{ccF}} < 5$. The FP are the $y = 0$ cases that produce $\mathrm{SNR}_{\mathrm{ccf}} > 5$. We run about 10000 spectra through this algorithm to determine the number of FPs produced. Figure 5.6a shows a sample confusing matrix when generating a sample set with a median $C = 10^{-3}$ with lowest $C = 10^{-4}$ and the highest $C = 10^{-2}$ and median $\mathrm{SNR} = 10^{7}$. In this case we have $\mathrm{FP} = 0$ and $\mathrm{TPR} = 1.0$. We also observe a case at higher contrast with mean $C = 10^{-4}$ in Figure 5.6b where we see that while with a $\mathrm{FP} = 1$ we have a low number of false positives for 15000 examples; we have a low $\mathrm{TPR} = 0.67$ owing to both the higher contrast and the lower SNR. Note that for both figures, we are well beyond the detection limit based on Figure 4.2. However, this exercise is meant to illustrate the different ranges of training examples that are used with the ML algorithms. Since higher SNR are more amenable to detection over a range of contrasts, they are the initial range of SNR that will be used for training.



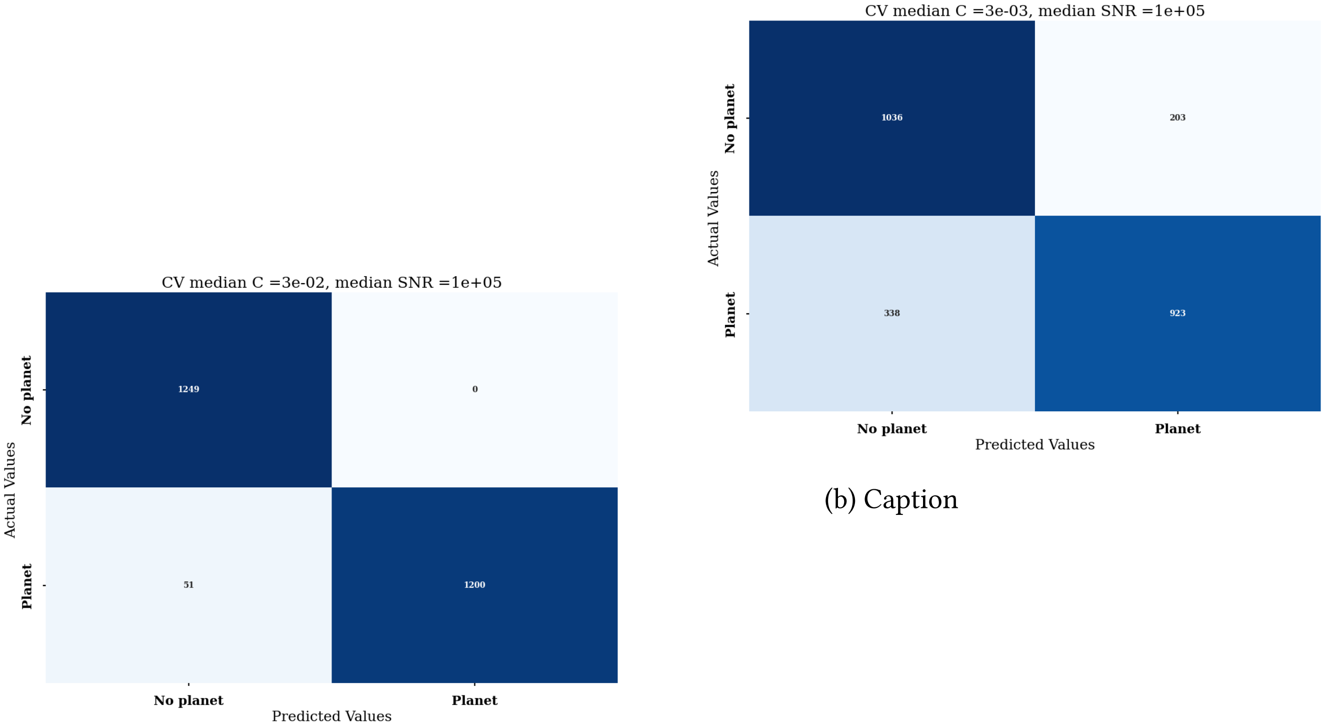
(a) Caption

(b) Caption

Figure 5.6: CCF results

5.3.2 Low contrast results

The low contrast performance was sequentially tested with random forests, MLP and autoencoders. We will present the results here from the test set, the contrast ranges of 10^{-1} to 10^{-3} which is the "low contrast" region for the purpose of our analyses. We present the results of the random forest for this contrast range for the validation data with $\approx$ 2500 examples. These training runs were conducted with two ranges $10^{-1} > C > 10^{-2}$ which are presented in Figure 5.7a and for the range of $10^{-3} > C > 10^{-2}$ in Figure 5.7b. Similarly, for the autoencoder we split the results into the same contrast ranges and this is depicted in Figure 5.8a and $Figure\ 5.8b$



(a) Caption

(b) Caption

Figure 5.7: RF results

When comparing the low contrast results its very interesting to look at Figure 5.6a and Figure 5.7b and Figure 5.8b which are tested around the same contrasts. The number of false positives is 0 for the cross correlation and is $\approx$ 200 for the ML aglorithms. Note that this number is $\approx$ 10 for a lower contrast on the left. The $TPR \approx 0.75$ for the ML algorithms but for the cross correlation based algorithm $\mathrm{TPR} = 1$. This is an interesting feature that when the contrast increases even in the low contrast cases the TPR plummets by a $\approx 1/3$ and the number of false positives increases from 0 to 200. Note that while the TPR remains high enough for the algorithm to be considered successful for mean $C \approx 10^{-3}$, the number of false positives would deem the algorithm unsuitable for scientific usages. At this stage, we can say that ML algorithms have not achieved the necessary false positive requirement for scientific analysis. However, in order to understand at what contrast the ML algorithms no longer learn new features we will also present the high contrast results.

5.3.3 high contrast results

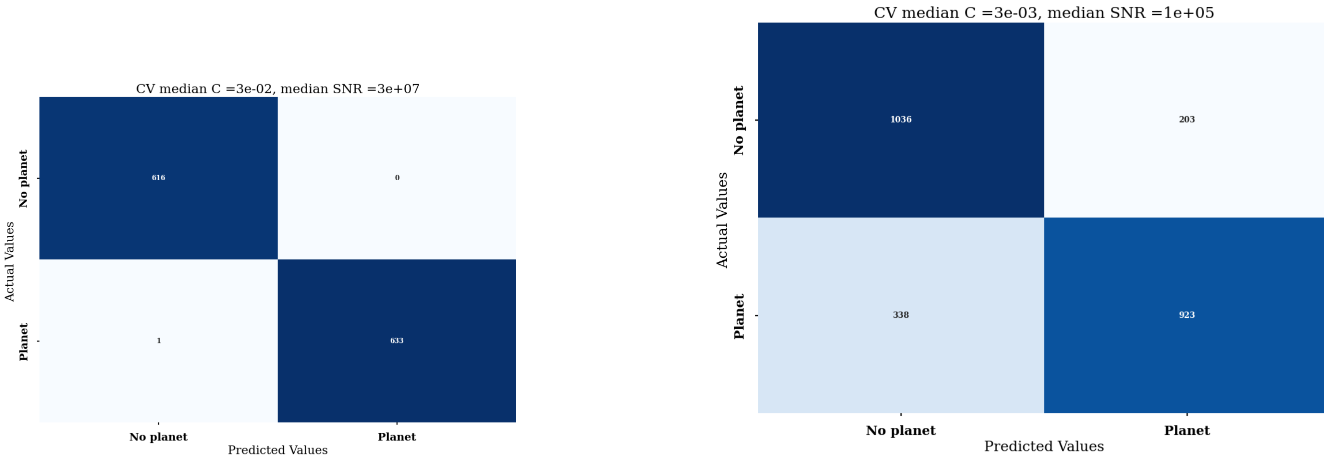
We define a high contrast spectrum as that where $C < 10^{-3}$. We have already such a case of confusio matrix being produced for Figure 5.6b where we see that 1 false positive is identified for 10^4 spectra and the $\mathrm{TPR} \approx 0.7$. We will now train and test this data with ML algorithms and these results are depicted in Figure 5.9a [TODO: 1. Training CM for both low and high contrast for all the algorithms 2. Make sure the CM are in the same SNR 3. Have CM from two different ranges of SNR so its clear that its only contrast and not SNR which is the problem]

5.4 Discussion

In this chapter, we have explored the use of three different ML algorithms in testing the detection hypothesis. We were unable to progress beyond testing the algorithms themselves to subsequently test the hypotheses themselves. In this discussion section we will discuss what are the reasons for the poor results from ML algorithms.

5.4.1 Random forest feature importances

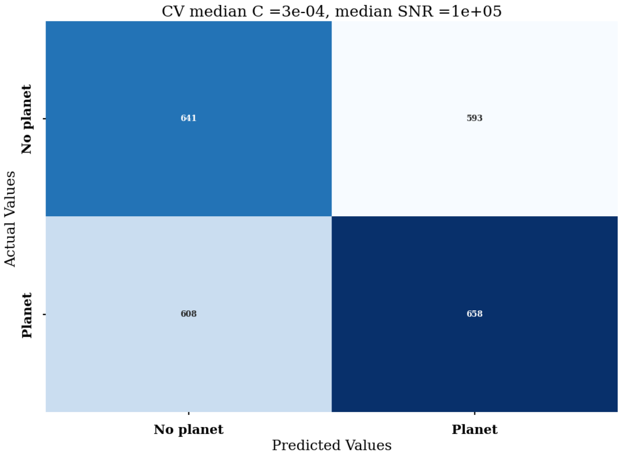
From the confusion matrices Figure 5.8a and Figure 5.7a it is clear that the training on very low contrasts where the planet is about 10th as bright as the host star allows the ML algorithms to train effectively and produce the necessary confusion matrices as desired by our problem statement. From these confusion matrices, it appears that as we increase the con-



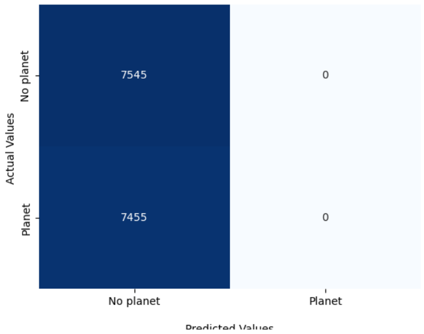
(a) Caption

(b) Caption

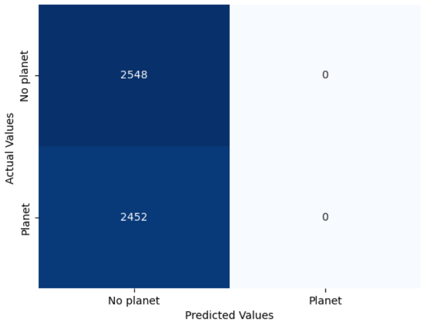
Figure 5.8: AE results



(a) Caption



(b) Caption



(c) Caption

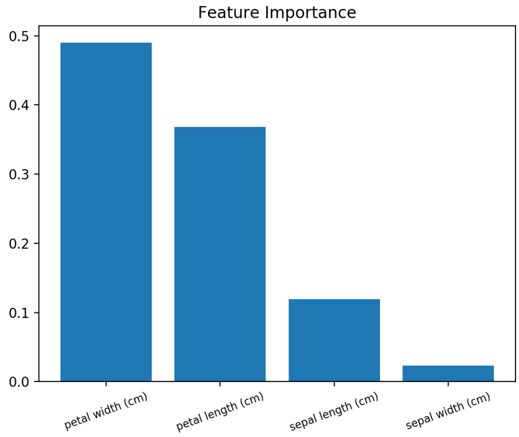


Figure 5.10: Caption

trast the ability to classify drops off quite rapidly. If this is indeed the case, what features does the ML algorithm learn in the lowest contrast that seem to disappear in the higher contrasts. In order to answer this question we performed a very specific experiment by generating spectra for an exoplanet with exactly the same $\mathrm{T_{eff}}$ and $\log(\mathrm{g})$ and train the random forest for different contrasts. Note that even this case we get the exact same confusion matrices. Thus we produce Figure 5.10 using the same SNR and different values C. The feature importances are plotted as function of λ and should therefore track the importance of the wavelength bins. In this figure, the uppermost dark panel indicates the template that was used to generate spectra to train with different contrasts. The lower panels indicate the featrue importances produced by different random forest models. The goal of our training is to generate feature importances that are a mirror image of the first panel in this image. The first panel in this image represents the spectral features which are unique to an exoplanet that are not shared by either the noise or the stellar spectrum. This means that the features that the ML algorithms need to learn are the features uniquely present in the exoplanet spectrum. By training the ML algorithms with different types of spectra, we attempted to produce a plethora of features that would allow an ML algorithm to generalize. However, even when we restrict this problem to exactly one type of exoplanet we see that the features learnt with the changing contrasts are highly limited by the contrast.

One of the first noteworthy things about Figure 5.10 is the low values of the feature importances when compared to more ‘classical’ feature importances such as Figure 2.2. While there is a relative gradient at the lowest contrast, the mean value is quite low. This of course is a function of having a large number of absorption features which means that when no one feature is more important than the others we will have the importances distributed over many wavelengths. Our view on this plot is that while there does seem to be some amount of learning of the relative importances,the extremely low value of each importance actually seems to indicate that the model is unable to train robustly and learn all the features. The second noteworthy point on this graph is the virtual disappearance of importances as the contrast increases and C value decreases. There seems to be some type of overfitting for the noise we can see a slight curving of the importances with the increasing wavelength. When we look at the same feature importances for a low contrast and changing SNR we can see the same kind of behaviour in Figure 5.11. We see that for the highest SNR the

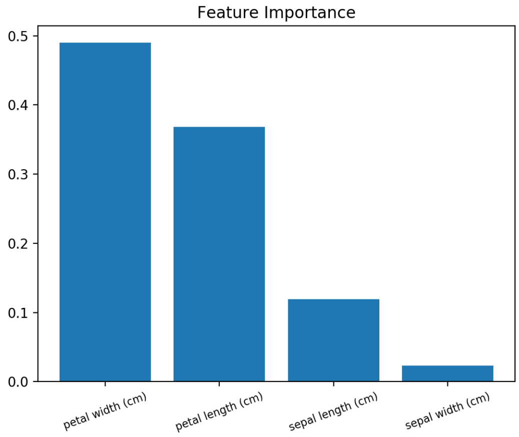


Figure 5.11: Use the correct plot here

feature importances follow the same pattern as in Figure 5.10 and then for the same low contrast spectrum we see the feature importances disappearing as we decrease SNR. This behaviour attests to the most important inference of this part of my thesis, In order for ML algorithms to be able to train and generalize on spectra from direct images the data needs to be both low contrast and high SNR as described in Chapter 3. In the next subsection we will discuss why this form of problem statement is not very appropriate for ML algorithms.

5.4.2 On the unsuitability of ML algorithms to test the detection and characterization hypotheses

At the beginning of this part in Chapter 1, we stated that ML algorithms had the ability to process multiple datasets rapidly and with high precision. We also stated the ability of ML algorithms to draw inferences from diverse data sources. In that sense we have developed our ML based algorithms to train on spectral data from multiple spectral channels. We have identified the 'best' quality data by defining a SNR which is a pure signal metric to define a spectrum. We have also used C as a parameter to inject astrophysics into the problem. But these have served only as scaffolding to the fundamental question, which is whether given diverse spectral features in a high quality astrophysical spectrum can an ML algorithm train and generalize to detect exoplanets and further to characterize them. To this effect we trained an ensemble algorithm and two deep learning algorithms. We found that the best spectra to train were indeed the ones with lowest contrast and highest SNR. We also learnt that the features that for example a random forest based algorithm learns is a mild version of the spectral features shows that ML algorithms are indeed capable of learning some features. However, what is also clear is that ML algorithms are not able to diversify these features to pick up higher contrast or lower SNR exoplanets in spectra.

To verify that the features are not transferrable, we saved the weights of the deep learning algorithms and used them as the starting weights to train higher contrast companions. We found that the confusion matrices are exactly the same. When we tried to use the same weights tuned to detect lower contrast exoplanets we found that the neural network never detects the exoplanet. This means that the features learnt on low contrast or high SNR are not general enough to be used to detect exoplanets. This is the fundamental reason why we cannot compose a detection matrix using ML algorithms. Consequently, we cannot test the detection hypothesis using ML algorithms.

The characterization hypothesis, relies on the ability of an algorithm to primarily detect the spectrum. We saw with the cross correlation based algorithm that when an exoplanet is detectable in the spectrum, it can be characterized with a consistent error bar which in turn can be reduced by perfectly dividing out the stellar contamination. However, ML algorithms have proven to be incompetent in learning features to even test the detection hypothesis and therefore it does not behove to test the characterization hypothesis which relies on the very same features. Note that this thesis has stopped short of composing the characterization matrix for those exoplanet spectra that can be detected due to the lack of time.

Chapter 6

Discussion and conclusion

This part of the thesis has had two broad goals for using direct imaging spectra,

1. to define whether detectability and characterization of exoplanets using spectra was possible and if so what is the biggest factor in this detectability and characterization,
2. and can the use of modern ML algorithms improve this detectability and characterization, and if so can this be quantified using a common metric that could be shared by both ML and non ML algorithms.

We have tested these ideas in different ways and in this chapter we will discuss the different aspects of testing both these ideas. To start we will discuss how the detection and characterization hypotheses are relevant to achieving goal 1 above. In this context we will discuss the relevance of the detection and characterization matrix and how they aid us in exploring idea 1. Then we will discuss how the use of ML algorithms has been ineffective in exploring the scientific idea 1 and how this means that they are unsuitable to use to explore idea 2. Finally, we will discuss the broad reasons around ML algorithms not being very successful and what are the steps that can be taken to mitigate this issue.

6.1 The detection and characterization hypotheses

The broad goal of this part of my thesis was to evaluate whether spectra could be use to detect exoplanets and to characterize them by using the same spectral features. This implies that the absorption features that are typical to an exoplanet will be well discriminated from those of the host star and if the spectrum is of sufficient quality with enough of these absorption lines. The relevant question, therefore, is when the stellar and planetary spectra are combined would a well tuned algorithm be able to detect the exoplanetary features well enough to not ony be able to discover an exoplanet but predict with a known level of uncertainty the kind of planet it is.

Based on these questions, we defined the detection and characterization hypotheses, to study the limits of detection of a warm Jupiter and its corresponding characterization.

These hypotheses allowed us to define the problem statement that we are aiming to test and justify the reason to reject some results and accept others. The hypotheses also validate the benchmark by which we will evaluate algorithms to test these hypotheses. For instance, the detection hypothesis states that so long as there are exoplanet absorption features in a spectrum, the exoplanet is detectable in sufficiently high quality spectrum. We establish that the quality of a spectrum would be defined by the number of photons it gathers, and if they are sufficiently high we would be able to detect an exoplanet. This, however, does not sufficiently define the sensitivity of an algorithm to detect exoplanets. Therefore to define how sensitive an algorithm is we defined the contrast C at which an exoplanet is present with respect to the host star. We then developed a detection matrix to define this sensitivity and test the detection hypothesis. The detection matrix allows us to define if at the lowest contrast (highest value of C) it is still possible to detect the exoplanet for all the different quality spectra. In principle, the contrast at which a detection is claimed is a measure of how an algorithm interprets spectra and not a test of the hypothesis. The hypothesis is tested by checking if an exoplanet can be detected for different quality of spectra. We tested this hypothesis extensively by considering spectra of very low SNR and running it through our cross correlation based detection algorithm. We found that the detection matrix tests the detection hypothesis for even very low values of SNR and for the highest SNR we found that the exoplanet was detected at the highest contrasts $C = 10^{-6}$. For the detection hypothesis, this means that it was tested and verified at a $\mathrm{SNR} \geq 10^5$. For lower SNRs we are limited by the sensitivity of the algorithm. This is one of the major contributions of this part of the thesis that we have produced a detection matrix to test the detection hypothesis which also serves to validate the sensitivity of an algorithm.

The characterization hypothesis defines that the characteristic parameters of the exoplanet, namely $\mathrm{T_{eff}}$ and $\log(\mathrm{g})$ can be estimated through its spectra with well quantified error bars. We defined a characterization matrix that allows us to infer the mean values of these characteristic parameters and estimate their uncertainty. When we run the spectra through our cross correlation based algorithm, we found two salient results,

1. we found that the characterization error bar became consistent at $\mathrm{SNR_{ccf}} \geq 3$ and that this value does not change with increase in SNR but,
2. this uncertainty changed for lower contrasts and in fact this uncertainty became the lowest for $C = 1$ i.e a case of where stellar signal was perfectly removed from the spectrum.

These two results fundamentally validate the ideas that,

1. the characteristic parameters of the exoplanet in the spectrum can be estimated with a known accuracy when the exoplanet is detectable and
2. this characterization is impacted by the amount of residual stellar features that are present in the spectrum.

Both of these derivations are corollaries to the characterization hypothesis which states that *when* the exoplanet is present in the spectrum we are able to characterize the spectra with constant error bars. This thesis thus defines this word *when* to be $\mathrm{SNR_{ccf}} \geq 3$. The second idea that stellar contamination is the fundamental reason for error in the characterization of the exoplanet allows us to appreciate the limitation of characterization algorithms when the characteristic is a broad parameter such as $\mathrm{T_{eff}}$.

Thus, the detection and characterization hypotheses have been well tested and verified with the cross correlation algorithm. This allowed us to rule out the fundamental question of whether this problem is well defined. Secondly it also now allowed us to set a benchmark and parameter space for follow up algorithms. The fact that the sensitivity of the cross correlation based detection algorithm was the highest at $\mathrm{SNR} > 10^5$ pointed to that value being the cut off for the best quality spectra. It also allowed us to state that scientifically it provides reasonable basis to verify that algorithms satisfy the detection hypothesis at these values of SNR before we undertake a study of their sensitivity to detection at lower SNR. Finally, the tests of the characterization hypothesis allowed us to link the detection and characterization at $\mathrm{SNR_{ccf}} > 3$. Note that this still cannot be called a detection but this allows us to make a scientific justification to produce a characterization matrix before testing the detectability of an exoplanet rigorously.

6.2 Difference between the performances of the ML and cross correlation based algorithm

The detection and characterization matrices allow us to test the detection and characterization hypotheses respectively. The goal for ML algorithms was to operate on the same data and the same evaluation criterion to ensure we are able to compare both the types of algorithms. We chose to start training and testing ML algorithms with the highest SNR spectra, with the goal as first establish the sensitivity of ML algorithms on the best quality spectra. This would have allowed us to establish the detection sensitivity of an ML based algorithm at a SNR where the detection hypothesis was satisfied by the cross correlation based for the highest contrast.

As a start we had planned to test the ability of ML algorithms to detect spectra with exoplanet features in them consistently at a specific value of R, SNR for different values of C starting from 10^{-1} down to $C = 10^{-6}$. We aimed to evaluate the detection sensitivity of these algorithms using confusion matrices. As has been clear from the results, the highest contrast that ML algorithms can detect exoplanets is 10^{-3}. An analysis of the results showed that the data features that the ML algorithms need to learn to detect exoplanets disappear beyond this contrast. Naturally, producing a detection matrix is moot at this point because ML algorithms do not seem to be able to test the detection hypothesis. Evidence, also shows that if you are unable to detect the exoplanet it is not possible to test the characterization hypothesis. Consequently, this thesis concludes that ML algorithms are not appropriate to test these hypotheses in this manner. The question therefore is why these ML algorithms failed where a cross correlation based algorithm had succeeded. There are several reasons this could be true we list a few of them below,

1. the lack of significant differentiating features in spectra for ML algorithms to learn from:

One of the major requirements for ML algorithms to separate the $y = 0$ and $y = 1$ cases would be the presence of clear differentiating features between the two classes. When we look for example at Figure 5.1 and Figure 5.2 it is fairly evident that Figure 5.2 contains several spectral bins that contain differentiating features, which may not be so evident in Figure 5.1. This is reflected in confusion matrices produced by the ML algorithms on the same parameter spaces. We clearly observe that the matrices produced with the low contrast exoplanets has very good TPR and low false positives. But as the contrast increases, we notice the confusion matrices producing more false negatives and in the case of random forests more false positives as well. All these indicate that while in the lower contrast cases the algorithms have features that are different between $y = 0$ and $y = 1$, as the contrast increases these features get washed away by stellar signal. This is why even at the higher SNR contrast seems to play such an important role. Thus, we also see that as the contrast increases, deep learning algorithms see only $y = 0$ cases whereas random forests get confused more with the noise.

2. the presence of large number of wavelength bins that do not contain discriminating information:

the large number of wavelength bins are supposed to act as features to ML algorithms which allows them to learn the difference between the $y = 0$ and $y = 1$ cases, provided there are enough wavelength bins with features. The cross correlation based algorithm uses this measure to produce a strong cross correlation signature by accounting for the small amounts of information in each wavelength bin. This has caused problems for many detection problems in the past where false correlations due to atmospheric lines, particularly the Telluric absorption. The assumption at the beginning of this part was that there was enough information in each wavelength bin which allows the cross correlation to detect the exoplanets and therefore should be sufficient for ML algorithms as well. However, this appears to have been an incorrect assumption, because if this was true then either increasing the contrast or increasing the SNR should have had lesser effect than it currently has. If the ML algorithms were purely influenced by the astrophysical features, then we would notice an ML algorithm continuing to detect exoplanets at low SNR or at the very least be less sensitive to changing SNR and a reverse effect would be noticed if the ML algorithms were only affected by the signal in the data. But we notice that both the SNR needs to be high and the C has to be low for ML algorithms to detect exoplanets. This is a sign that that not only are there no differentiating features at higher contrasts and lower SNR but that intrisically the large number of wavelength bins don't contain enough information. A counter point to this is that it is possible that ML algorithms would be trained if we limit the spectra to certain wavelength bins rather than the whole spectrum.

3. finally the presence of photon noise which leads to a lot of variance in features:

astronomical observations have at the very least photon noise. This is very basic and intrinsic noise that is always present in astronomical data. This noise produces intrinsic randomness in the data and also washes out crucial features in the data. We see that both the cross correlation and ML based algorithms are both impacted by this. While the sensitivity of the cross correlation algorithm is limited by this noise, for the ML algorithms it appears to hamper both training and validation of the data. This noise seems to also produce a large amount of confusion in random forests. When the noise levels are decreased at a constant contrast $C > 10^{-3}$ there seems to be minimal impact of noise on the algorithms. We see this as an effect of the nature of photon noise.

6.3 Conclusions and limitations of the study

This study has had several interesting results and contributions to the field of exoplanet detection and characterization. We will first list the notable contributions and then discuss the many limitations of this study.

6.3.1 Conclusions of this study

The conclusions in this study follow three major axis points, the scientific hypotheses developed for this study, followed by the algorithms and finally generalised conclusions of the study.

The detection and characterization hypotheses
In the field of direct exoplanet detection, the use of spectra are well known and somewhat well explored. This part of the thesis primarily explored the idea of whether it was possible to simultaneously detect and characterize exoplanets using their spectra alone. In this part we have defined the detection hypothesis which was tested using the detection matrix and the characterization hypothesis which was tested using the characterization matrix. We defined both these hypotheses and their evaluation metrics to be compatible with both ML and non-ML algorithms. We then generated data that could be used for both types of algorithms and which have a basis in astrophysics.

A cross correlation based detection and characterization algorithm:
We developed an algorithm that cross correlates template spectra with target data spectra, and we interpret these results to compute the detection and characterization matrices. We discovered that the sensitivity of detection of an exoplanet is limited by the SNR of the spectrum. We also found that the detection hypothesis is satisified for all values of contrast at the highest SNR. The characterization of an exoplanet is limited by the same parameters as the detection, but at a lower threshold of the detection parameter we obtain a stable uncertainty but a value higher than what is demanded by the scientific community. We found that the perfect characterization of an exoplanet is limited by the stellar contamination and when stellar contamination is perfectly removed then we have perfect characterization with stable uncertainties.

generalized conclusions:
This study has attempted to understand the interaction between detectability, charac-

terization and the improvement that ML algorithms bring to this interaction. We have concluded that exoplanet detection using cross correlation algorithms is the ultimate test of the presence of an exoplanet and characterization serves as a clue to the presence of the presence of this exoplanet given very specific conditions (for ex: $\mathrm{SNR}_{\mathrm{ccf}} > 3$). This study has also learnt that ML algorithms, no matter how advanced, are severely impacted by both noise and the contrast of the exoplanet. The changing SNR impacts the cross correlation by reducing its sensitivity but in case of the ML based algorithms we notice that both changing SNR and C impacts the detection sensitivity.

6.3.2 Limitations of this study

This study has certain specific limitations fundamentally related to its scope, the data used and extent of exploration of the use of ML algorithms.

an unfair comparison between the cross correlation based and ML based algorithms:
while comparing the performance of the cross correlation based algorithms with those of the ML based algorithms it appears that this kind of problem statement is better suited for a cross correlation kind of algorithm which only evaluates the similarity between spectra. Since we generate large number of spectra which are subject to only random modifications, this problem could be better suited to an unsupervised approach than a supervised one

lack of realistic variation in the data:
so ML doesn't train

the choice of ML algorithms:
maybe a more interesting choice than a safe one could have helped

non use of specific molecules:
Use of molecules is well established but we chose not to.

6.4 Next steps and how this can feature sparsity be countered.

One of the major reasons to not be able to compose a detection matrix with ML algorithms has been the sparsity of features in the spectra. As we saw with Figure 5.11, it was clear that no matter the contrast or the SNR the feature importances never were strong enough to reproduce spectral features. While at lower contrasts we could still some features, these would have disappeared in a more realistic and varying dataset. This makes it unambiguous that the data needs sufficient feature to be able to detect exoplanets.

It is also clear from our analysis that basic characterization of an exoplanet using its spectrum is fundamentally related to its detection. Therefore, it is imperative to study the best way to present ML algorithms data to detect exoplanets at high contrasts. We have also seen that SNR provides a loosely but not wholly reliable metric to evaluate if the data is sufficiently diverse to train a ML algorithm or contains sufficient features. Therefore, we need to find a metric that gives us insight into the data that is presented to ML algorithms for training. In the part that follows this, we will study how ML algorithms behave when provided with a feature rich dataset and when compared using a metric that sees the same features as the ML algorithm.

Bibliography

S. Agrawal, J.-B. Ruffio, Q. M. Konopacky, B. Macintosh, D. Mawet, E. L. Nielsen, K. K. W. Hoch, M. C. Liu, T. S. Barman, W. Thompson, A. Z. Greenbaum, C. Marois, and J. Patience. Detecting Exoplanets Closer to Stars with Moderate Spectral Resolution Integral-field Spectroscopy. AJ, 166(1):15, July 2023. doi: 10.3847/1538-3881/acd6a3.

A. Aleman, B. Macintosh, B. Lacy, T. Groff, N. Zimmerman, and V. Bailey. Constraining Metallicity and Gravity of Young Exoplanets with the Nancy Grace Roman Space Telescope's Coronagraph Instrument. In *American Astronomical Society Meeting Abstracts*, volume 55 of *American Astronomical Society Meeting Abstracts*, page 164.07, Jan. 2023.

P. D. Aleo, K. L. Malanchev, M. V. Pruzhinskaya, E. E. O. Ishida, E. Russeil, M. V. Kornilov, V. S. Korolev, S. Sreejith, A. A. Volnova, and G. S. Narayan. SNAD transient miner: Finding missed transient events in ZTF DR4 using k-D trees. New A, 96:101846, Oct. 2022. doi: 10.1016/j.newast.2022.101846.

F. Allard, P. H. Hauschildt, D. R. Alexander, and S. Starrfield. Model Atmospheres of Very Low Mass Stars and Brown Dwarfs. ARA&A, 35:137–177, Jan. 1997. doi: 10.1146/annurev.astro.35.1.137.

F. Allard, D. Homeier, and B. Freytag. Model Atmospheres From Very Low Mass Stars to Brown Dwarfs. In C. Johns-Krull, M. K. Browning, and A. A. West, editors, *16th Cambridge Workshop on Cool Stars, Stellar Systems, and the Sun*, volume 448 of *Astronomical Society of the Pacific Conference Series*, page 91, Dec. 2011.

A. Amara and S. P. Quanz. PYNPOINT: an image processing package for finding exoplanets. MNRAS, 427(2):948–955, Dec. 2012. doi: 10.1111/j.1365-2966.2012.21918.x.

D. Angerhausen and S. Quanz. The Large Interferometer For Exoplanets (LIFE) mission: status and progress report. In *European Planetary Science Congress*, pages EPSC2021–284, Sept. 2021. doi: 10.5194/epsc2021-284.

N. M. Ball and R. J. Brunner. Data Mining and Machine Learning in Astronomy. *International Journal of Modern Physics D*, 19(7):1049–1106, Jan. 2010. doi: 10.1142/S0218271810017160.

D. Baron. Machine Learning in Astronomy: a practical overview. *arXiv e-prints*, art. arXiv:1904.07248, Apr. 2019. doi: 10.48550/arXiv.1904.07248.

N. E. Batalha, M. S. Marley, N. K. Lewis, and J. J. Fortney. Exoplanet Reflected-light Spectroscopy with PICASO. ApJ, 878(1):70, June 2019. doi: 10.3847/1538-4357/ab1b51.

J. L. Beuzit, A. Vigan, D. Mouillet, K. Dohlen, R. Gratton, A. Boccaletti, J. F. Sauvage, H. M. Schmid, M. Langlois, C. Petit, A. Baruffolo, M. Feldt, J. Milli, Z. Wahhaj, L. Abe, U. Anselmi, J. Antichi, R. Barette, J. Baudrand, P. Baudoz, A. Bazzon, P. Bernardi, P. Blanchard, R. Brast, P. Bruno, T. Buey, M. Carbillet, M. Carle, E. Cascone, F. Chapron, J. Charton, G. Chauvin, R. Claudi, A. Costille, V. De Caprio, J. de Boer, A. Delboulbé,

S. Desidera, C. Dominik, M. Downing, O. Dupuis, C. Fabron, D. Fantinel, G. Farisato, P. Feautrier, E. Fedrigo, T. Fusco, P. Gigan, C. Ginski, J. Girard, E. Giro, D. Gisler, L. Gluck, C. Gry, T. Henning, N. Hubin, E. Hugot, S. Incorvaia, M. Jaquet, M. Kasper, E. Lagadec, A. M. Lagrange, H. Le Coroller, D. Le Mignant, B. Le Ruyet, G. Lessio, J. L. Lizon, M. Llored, L. Lundin, F. Madec, Y. Magnard, M. Marteaud, P. Martinez, D. Maurel, F. Ménard, D. Mesa, O. Möller-Nilsson, T. Moulin, C. Moutou, A. Origné, J. Parisot, A. Pavlov, D. Perret, J. Pragt, P. Puget, P. Rabou, J. Ramos, J. M. Reess, F. Rigal, S. Rochat, R. Roelfsema, G. Rousset, A. Roux, M. Saisse, B. Salasnich, E. Santambrogio, S. Scuderi, D. Segransan, A. Sevin, R. Siebenmorgen, C. Soenke, E. Stadler, M. Suarez, D. Tiphène, M. Turatto, S. Udry, F. Vakili, L. B. F. M. Waters, L. Weber, F. Wildi, G. Zins, and A. Zurlo. SPHERE: the exoplanet imager for the Very Large Telescope. A&A, 631: A155, Nov. 2019. doi: 10.1051/0004-6361/201935251.

H. Bonnet, R. Conzelmann, B. Delabre, R. Donaldson, E. Fedrigo, N. N. Hubin, M. Kissler-Patig, J.-L. Lizon, J. Paufique, S. Rossi, S. Stroebele, and S. Tordo. First light of SINFONI AO-module at VLT. In D. Bonaccini Calia, B. L. Ellerbroek, and R. Ragazzoni, editors, *Advancements in Adaptive Optics*, volume 5490 of *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, pages 130–138, Oct. 2004. doi: 10.1117/12.551187.

L. Breiman. Arcing classifier (with discussion and a rejoinder by the author). *The annals of statistics*, 26(3):801–849, 1998.

L. Breiman. Random forests. *Machine learning*, 45:5–32, 2001.

M. Brogi and M. R. Line. Retrieving Temperatures and Abundances of Exoplanet Atmospheres with High-resolution Cross-correlation Spectroscopy. AJ, 157(3):114, Mar. 2019. doi: 10.3847/1538-3881/aaffd3.

B. Calvin, N. Jovanovic, G. Ruane, J. Pezzato, J. Colborn, D. Echeverri, T. Schofield, M. Porter, J. K. Wallace, J.-R. Delorme, and D. Mawet. Enhancing Direct Exoplanet Spectroscopy with Apodizing and Beam Shaping Optics. PASP, 133(1020):024503, Feb. 2021. doi: 10.1088/1538-3873/abdace.

C. Cantero, O. Absil, C.-H. Dahlqvist, and M. Van Droogenbroeck. NA-SODINN: a deep learning algorithm for exoplanet image detection based on residual noise regimes. *arXiv e-prints*, art. arXiv:2302.02854, Feb. 2023. doi: 10.48550/arXiv.2302.02854.

A. Chomez, A. M. Lagrange, P. Delorme, M. Langlois, G. Chauvin, O. Flasseur, J. Dallant, F. Philipot, S. Bergeon, D. Albert, N. Meunier, and P. Rubini. Preparing an unsupervised massive analysis of SPHERE high contrast data with the PACO algorithm. *arXiv e-prints*, art. arXiv:2305.08766, May 2023. doi: 10.48550/arXiv.2305.08766.

V. Christiaens, S. Casassus, O. Absil, S. Kimeswenger, C. A. Gomez Gonzalez, J. Girard, R. Ramírez, O. Wertz, A. Zurlo, Z. Wahhaj, C. Flores, V. Salinas, A. Jordán, and D. Mawet. Characterization of low-mass companion HD 142527 B. A&A, 617:A37, Sept. 2018. doi: 10.1051/0004-6361/201629454.

V. Christiaens, M.-G. Ubeira-Gabellini, H. Cánovas, P. Delorme, B. Pairet, O. Absil, S. Casassus, J. H. Girard, A. Zurlo, Y. Aoyama, G.-D. Marleau, L. Spina, N. van der Marel, L. Cieza, G. Lodato, S. Pérez, C. Pinte, D. J. Price, and M. Reggiani. A faint companion around CrA-9: protoplanet or obscured binary? *Monthly Notices of the Royal Astronomical Society*, 502(4):6117–6139, 02 2021. ISSN 0035-8711. doi: 10.1093/mnras/stab480. URL https://doi.org/10.1093/mnras/stab480.

V. Christiaens, C. Gonzalez, R. Farkas, C.-H. Dahlqvist, E. Nasedkin, J. Milli, O. Absil, H. Ngo, C. Cantero, A. Rainot, I. Hammond, M. Bonse, F. Cantalloube, A. Vigan, V. Kompella, and P. Hancock. VIP: A Python package for high-contrast imaging. *The Journal of Open Source Software*, 8(81):4774, Jan. 2023. doi: 10.21105/joss.04774.

G. Cugno, T. D. Pearce, R. Launhardt, M. J. Bonse, J. Ma, T. Henning, A. Quirrenbach, D. Ségransan, E. C. Matthews, S. P. Quanz, G. M. Kennedy, A. Müller, S. Reffert, and E. L. Rickman. ISPY: NACO Imaging Survey for Planets around Young stars. The demographics of forming planets embedded in protoplanetary disks. A&A, 669:A145, Jan. 2023. doi: 10.1051/0004-6361/202244891.

S. Deb, A. Baruah, and S. Kumar. Ensemble-based unsupervised machine learning method for membership determination of open clusters using Mahalanobis distance. MNRAS, 515(4):4685–4701, Oct. 2022. doi: 10.1093/mnras/stac2116.

J. L. Fischer, H. Domínguez Sánchez, and M. Bernardi. SDSS-IV MaNGA PyMorph Photometric and Deep Learning Morphological Catalogues and implications for bulge properties and stellar angular momentum. MNRAS, 483(2):2057–2077, Feb. 2019. doi: 10.1093/mnras/sty3135.

C. Fisher, H. J. Hoeijmakers, D. Kitzmann, P. Márquez-Neila, S. L. Grimm, R. Sznitman, and K. Heng. Interpreting High-resolution Spectroscopy of Exoplanets using Cross-correlations and Supervised Machine Learning. AJ, 159(5):192, May 2020. doi: 10.3847/1538-3881/ab7a92.

C. J. Fluke and C. Jacobs. Surveying the reach and maturity of machine learning and artificial intelligence in astronomy. *WIREs Data Mining and Knowledge Discovery*, 10(2):e1349, Jan. 2020. doi: 10.1002/widm.1349.

T. D. Gebhard, M. J. Bonse, S. P. Quanz, and B. Schölkopf. Half-sibling regression meets exoplanet imaging: PSF modeling and subtraction using a flexible, domain knowledge-driven, causal framework. A&A, 666:A9, Oct. 2022. doi: 10.1051/0004-6361/202142529.

C. A. Gomez Gonzalez, O. Absil, P. A. Absil, M. Van Droogenbroeck, D. Mawet, and J. Surdej. Low-rank plus sparse decomposition for exoplanet detection in direct-imaging ADI sequences. The LLSG algorithm. A&A, 589:A54, May 2016. doi: 10.1051/0004-6361/201527387.

C. A. Gomez Gonzalez, O. Absil, and M. Van Droogenbroeck. Supervised detection of exoplanets in high-contrast imaging sequences. A&A, 613:A71, May 2018. doi: 10.1051/0004-6361/201731961.

K. Gurney. *An introduction to neural networks.* CRC press, 1997.

S. Y. Haffert, A. J. Bohn, J. de Boer, I. A. G. Snellen, J. Brinchmann, J. H. Girard, C. U. Keller, and R. Bacon. Two accreting protoplanets around the young star PDS 70. *Nature Astronomy*, 3:749–754, June 2019. doi: 10.1038/s41550-019-0780-5.

D. Harris and S. Harris. *Digital Design and Computer Architecture*. Computer organization bundle, VHDL Bundle. Elsevier Science, 2010. ISBN 9780080547060. URL https://books.google.be/books?id=5X7JV5-n0FIC.

H. J. Hoeijmakers, H. Schwarz, I. A. G. Snellen, R. J. de Kok, M. Bonnefoy, G. Chauvin, A. M. Lagrange, and J. H. Girard. Medium-resolution integral-field spectroscopy for high-contrast exoplanet imaging. Molecule maps of the β Pictoris system with SINFONI. A&A, 617:A144, Oct. 2018. doi: 10.1051/0004-6361/201832902.

P. Huang, Z. Kong, M. Xie, and X. Yang. Robust unsupervised feature selection via data relationship learning. *Pattern Recognition*, 142:109676, Oct. 2023. doi: 10.1016/j.patcog.2023.109676.

S. Hunziker, S. P. Quanz, A. Amara, and M. R. Meyer. PCA-based approach for subtracting thermal background emission in high-contrast imaging data. A&A, 611:A23, Mar. 2018. doi: 10.1051/0004-6361/201731428.

Ž. Ivezić, A. J. Connolly, J. T. VanderPlas, and A. Gray. *Statistics, Data Mining, and Machine Learning in Astronomy: A Practical Python Guide for the Analysis of Survey Data*. 2014. doi: 10.1515/9781400848911.

L. Jocou, P. Rabou, T. Moulin, Y. Magnard, A. Hours, F. Pancher, S. Guieu, A. Carlotti, A. Delboulbé, M. Vérove, E. Stadler, D. Maurel, S. Rochat, F. Henault, K. Dohlen, N. Thatte, B. Neichel, P. Vola, F. Clarke, D. Melotte, M. Tecza, and H. Schnetler. HARMONI at ELT: development of the high-contrast module. In L. Schreiber, D. Schmidt, and E. Vernet, editors, *Adaptive Optics Systems VIII*, volume 12185 of *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, page 121854K, Aug. 2022. doi: 10.1117/12.2626984.

N. Jovanovic, J. R. Delorme, C. Z. Bond, S. Cetre, D. Mawet, D. Echeverri, J. K. Wallace, R. Bartos, S. Lilley, S. Ragland, G. Ruane, P. Wizinowich, M. Chun, J. Wang, J. Wang, M. Fitzgerald, K. Matthews, J. Pezzato, B. Calvin, M. Millar-Blanchaer, E. C. Martin, E. Wetherell, E. Wang, S. Jacobson, E. Warmbier, C. Lockhart, D. Hall, R. Jensen-Clem, and E. McEwen. The Keck Planet Imager and Characterizer: Demonstrating advanced exoplanet characterization techniques for future extremely large telescopes. *arXiv e-prints*, art. arXiv:1909.04541, Sept. 2019. doi: 10.48550/arXiv.1909.04541.

S. Kiefer, A. J. Bohn, S. P. Quanz, M. Kenworthy, and T. Stolker. Spectral and angular differential imaging with SPHERE/IFS. Assessing the performance of various PCA-based approaches to PSF subtraction. A&A, 652:A33, Aug. 2021. doi: 10.1051/0004-6361/202140285.

E. Kussul and T. Baidyk. Improved method of handwritten digit recognition tested on mnist database. *Image and Vision Computing*, 22(12):971–981, 2004.

B. Lavie, J. M. Mendonça, C. Mordasini, M. Malik, M. Bonnefoy, B.-O. Demory, M. Oreshenko, S. L. Grimm, D. Ehrenreich, and K. Heng. HELIOS-RETRIEVAL: An Open-source, Nested Sampling Atmospheric Retrieval Code; Application to the HR 8799 Exoplanets and Inferred Constraints for Planet Formation. AJ, 154(3):91, Sept. 2017. doi: 10.3847/1538-3881/aa7ed8.

H. W. Leung and J. Bovy. Deep learning of multi-element abundances from high-resolution spectroscopic data. MNRAS, 483(3):3255–3277, Mar. 2019. doi: 10.1093/mnras/sty3217.

X.-R. Li, R.-Y. Pan, and F.-Q. Duan. Parameterizing Stellar Spectra Using Deep Neural Networks. *Research in Astronomy and Astrophysics*, 17(4):036, Mar. 2017. doi: 10.1088/1674-4527/17/4/36.

B. Macintosh, J. R. Graham, P. Ingraham, Q. Konopacky, C. Marois, M. Perrin, L. Poyneer, B. Bauman, T. Barman, A. S. Burrows, A. Cardwell, J. Chilcote, R. J. De Rosa, D. Dillon, R. Doyon, J. Dunn, D. Erikson, M. P. Fitzgerald, D. Gavel, S. Goodsell, M. Hartung, P. Hibon, P. Kalas, J. Larkin, J. Maire, F. Marchis, M. S. Marley, J. McBride, M. Millar-Blanchaer, K. Morzinski, A. Norton, B. R. Oppenheimer, D. Palmer, J. Patience, L. Pueyo, F. Rantakyro, N. Sadakuni, L. Saddlemyer, D. Savransky, A. Serio, R. Soummer, A. Sivaramakrishnan, I. Song, S. Thomas, J. K. Wallace, S. Wiktorowicz, and S. Wolff. First light of the Gemini Planet Imager. *Proceedings of the National Academy of Science*, 111(35):12661–12666, Sept. 2014. doi: 10.1073/pnas.1304215111.

N. Madhusudhan. Exoplanetary Atmospheres: Key Insights, Challenges, and Prospects. ARA&A, 57:617–663, Aug. 2019. doi: 10.1146/annurev-astro-081817-051846.

M. Mâlin, A. Boccaletti, B. Charnay, F. Kiefer, and B. Bézard. Simulated performance of the molecular mapping for young giant exoplanets with the Medium Resolution Spectrometer of JWST/MIRI. *arXiv e-prints*, art. arXiv:2301.02116, Jan. 2023. doi: 10.48550/arXiv.2301.02116.

D. Mawet, J. Milli, Z. Wahhaj, D. Pelat, O. Absil, C. Delacroix, A. Boccaletti, M. Kasper, M. Kenworthy, C. Marois, B. Mennesson, and L. Pueyo. Fundamental Limitations of High Contrast Imaging Set by Small Sample Statistics. ApJ, 792(2):97, Sept. 2014. doi: 10.1088/0004-637X/792/2/97.

D. Mawet, P. Wizinowich, R. Dekany, M. Chun, D. Hall, S. Cetre, O. Guyon, J. K. Wallace, B. Bowler, M. Liu, G. Ruane, E. Serabyn, R. Bartos, J. Wang, G. Vasisht, M. Fitzgerald, A. Skemer, M. Ireland, J. Fucik, J. Fortney, I. Crossfield, R. Hu, and B. Benneke. Keck Planet Imager and Characterizer: concept and phased implementation. In E. Marchetti, L. M. Close, and J.-P. Véran, editors, *Adaptive Optics Systems V*, volume 9909 of *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, page 99090D, July 2016. doi: 10.1117/12.2233658.

D. Mesa, M. Keppler, F. Cantalloube, L. Rodet, B. Charnay, R. Gratton, M. Langlois, A. Boccaletti, M. Bonnefoy, A. Vigan, O. Flasseur, J. Bae, M. Benisty, G. Chauvin, J. de Boer, S. Desidera, T. Henning, A. M. Lagrange, M. Meyer, J. Milli, A. Müller, B. Pairet, A. Zurlo, S. Antoniucci, J. L. Baudino, S. Brown Sevilla, E. Cascone, A. Cheetham, R. U. Claudi, P. Delorme, V. D'Orazi, M. Feldt, J. Hagelberg, M. Janson, Q. Kral, E. Lagadec,

C. Lazzoni, R. Ligi, A. L. Maire, P. Martinez, F. Menard, N. Meunier, C. Perrot, S. Petrus, C. Pinte, E. L. Rickman, S. Rochat, D. Rouan, M. Samland, J. F. Sauvage, T. Schmidt, S. Udry, L. Weber, and F. Wildi. VLT/SPHERE exploration of the young multiplanetary system PDS70. A&A, 632:A25, Dec. 2019. doi: 10.1051/0004-6361/201936764.

T. Meshkat, M. A. Kenworthy, S. P. Quanz, and A. Amara. Optimized Principal Component Analysis on Coronagraphic Images of the Fomalhaut System. ApJ, 780(1):17, Jan. 2014. doi: 10.1088/0004-637X/780/1/17.

O. Miettinen. Protostellar classification using supervised machine learning algorithms. Ap&SS, 363(9):197, Sept. 2018. doi: 10.1007/s10509-018-3418-7.

P. Mollière, E. Nasedkin, E. Alei, K. Molaverdikhani, and M. Zilinskas. petitRADTRANS: Exoplanet spectra calculator. Astrophysics Source Code Library, record ascl:2207.014, July 2022.

S. Mukherjee, N. E. Batalha, J. J. Fortney, and M. S. Marley. PICASO 3.0: A One-dimensional Climate Model for Giant Planets and Brown Dwarfs. ApJ, 942(2):71, Jan. 2023. doi: 10.3847/1538-4357/ac9f48.

B. Naul, J. S. Bloom, F. Pérez, and S. van der Walt. A recurrent neural network for classification of unevenly sampled variable stars. *Nature Astronomy*, 2:151–155, Nov. 2018. doi: 10.1038/s41550-017-0321-z.

B. Pairet, F. Cantalloube, C. A. Gomez Gonzalez, O. Absil, and L. Jacques. STIM map: detection map for exoplanets imaging beyond asymptotic Gaussian residual speckle noise. MNRAS, 487(2):2262–2277, Aug. 2019. doi: 10.1093/mnras/stz1350.

P. Palma-Bifani, G. Chauvin, M. Bonnefoy, P. M. Rojo, S. Petrus, L. Rodet, M. Langlois, F. Allard, B. Charnay, C. Desgrange, D. Homeier, A. M. Lagrange, J. L. Beuzit, P. Baudoz, A. Boccaletti, A. Chomez, P. Delorme, S. Desidera, M. Feldt, C. Ginski, R. Gratton, A. L. Maire, M. Meyer, M. Samland, I. Snellen, A. Vigan, and Y. Zhang. Peering into the young planetary system AB Pic. Atmosphere, orbit, obliquity, and second planetary candidate. A&A, 670:A90, Feb. 2023. doi: 10.1051/0004-6361/202244294.

K. A. Pearson, L. Palafox, and C. A. Griffith. Searching for exoplanets using artificial intelligence. MNRAS, 474(1):478–491, Feb. 2018. doi: 10.1093/mnras/stx2761.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, A. Müller, J. Nothman, G. Louppe, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and É. Duchesnay. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, Oct. 2011. doi: 10.48550/arXiv.1201.0490.

D. J. M. Petit dit de la Roche, H. J. Hoeijmakers, and I. A. G. Snellen. Molecule mapping of HR8799b using OSIRIS on Keck. Strong detection of water and carbon monoxide, but no methane. A&A, 616:A146, Aug. 2018. doi: 10.1051/0004-6361/201833384.

M. W. Phillips, P. Tremblin, I. Baraffe, G. Chabrier, N. F. Allard, F. Spiegelman, J. M. Goyal, B. Drummond, and E. Hébrard. A new set of atmosphere and evolution models for cool T-Y brown dwarfs and giant exoplanets. A&A, 637:A38, May 2020. doi: 10.1051/0004-6361/201937381.

N. N. Prakash, V. Rajesh, D. L. Namakhwa, S. Dwarkanath Pande, and S. H. Ahammad. A DenseNet CNN-based liver lesion prediction and classification for future medical diagnosis. *Scientific African*, 20:e01629, July 2023. doi: 10.1016/j.sciaf.2023.e01629.

S. P. Quanz, A. Amara, M. R. Meyer, J. H. Girard, M. A. Kenworthy, and M. Kasper. Confirmation and Characterization of the Protoplanet HD 100546 b—Direct Evidence for Gas Giant Planet Formation at 50 AU. ApJ, 807(1):64, July 2015. doi: 10.1088/0004-637X/807/1/64.

J.-B. Ruffio, B. Macintosh, Q. M. Konopacky, T. Barman, R. J. De Rosa, J. J. Wang, K. K. Wilcomb, I. Czekala, and C. Marois. Radial velocity measurements of hr 8799 b and c with medium resolution spectroscopy. *The Astronomical Journal*, 158(5):200, 2019.

J.-B. Ruffio, Q. M. Konopacky, T. Barman, B. Macintosh, K. K. W. Hoch, R. J. De Rosa, J. J. Wang, I. Czekala, and C. Marois. Deep Exploration of the Planets HR 8799 b, c, and d with Moderate-resolution Spectroscopy. AJ, 162(6):290, Dec. 2021. doi: 10.3847/1538-3881/ac273a.

D. J. Ryan and T. D. Robinson. Detecting Oceans on Exoplanets with Phase-dependent Spectral Principal Component Analysis. *psj*, 3(2):33, Feb. 2022. doi: 10.3847/PSJ/ac4af3.

W. B. Sparks and H. C. Ford. Imaging Spectroscopy for Extrasolar Planet Detection. ApJ, 578(1):543–564, Oct. 2002. doi: 10.1086/342401.

P. Stinco, A. Tesei, and K. D. LePage. Unsupervised active sonar contact classification through anomaly detection. *EURASIP Journal on Applied Signal Processing*, 2023(1):59, Dec. 2023. doi: 10.1186/s13634-023-01016-z.

J. Tennyson and S. N. Yurchenko. ExoMol: molecular line lists for exoplanet and other atmospheres. 425(1):21–33, Sept. 2012. doi: 10.1111/j.1365-2966.2012.21440.x.

N. van der Marel, A. D. Bosman, S. Krijt, G. D. Mulders, and J. B. Bergner. If you like C/O variations, you should have put a ring on it. A&A, 653:L9, Sept. 2021. doi: 10.1051/0004-6361/202141786.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

P. Vincent. A connection between score matching and denoising autoencoders. *Neural computation*, 23(7):1661–1674, 2011.

J. Wang, D. Mawet, N. Jovanovic, J.-B. Ruffio, J.-R. Delorme, T. Schofield, D. Echeverri, E. Morris, L. Finnerty, Y. Xin, K. Horstman, B. Sappey, W. Xuan, and KPIC Team. Surveying Directly Imaged Planets at High Spectral Resolution with KPIC. In *Bulletin of the American Astronomical Society*, volume 54, page 102.347, June 2022.

J. Wang, J. J. Wang, J.-B. Ruffio, G. A. Blake, D. Mawet, A. Baker, R. Bartos, C. Z. Bond, B. Calvin, S. Cetre, J.-R. Delorme, G. Doppmann, D. Echeverri, L. Finnerty, M. P. Fitzgerald, N. Jovanovic, R. Lopez, E. C. Martin, E. Morris, J. Pezzato, S. Ragland, G. Ruane, B. Sappey, T. Schofield, A. Skemer, T. Venenciano, J. K. Wallace, P. Wizinowich, J. W. Xuan, M. L. Bryan, A. Roy, and N. L. Wallack. Retrieving C and O Abundance of HR 8799 c by Combining High- and Low-resolution Data. AJ, 165(1):4, Jan. 2023. doi: 10.3847/1538-3881/ac9f19.

J. J. Wang, J.-R. Delorme, J.-B. Ruffio, E. Morris, N. Jovanovic, D. Echeverri, T. Schofield, J. Pezzato, A. Skemer, and D. Mawet. High resolution spectroscopy of directly imaged exoplanets with KPIC. In S. B. Shaklan and G. J. Ruane, editors, *Techniques and Instrumentation for Detection of Exoplanets X*, volume 11823 of *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, page 1182302, Sept. 2021. doi: 10.1117/12.2596484.

C. Xie, S. Y. Haffert, J. de Boer, M. A. Kenworthy, J. Brinchmann, J. Girard, I. A. G. Snellen, and C. U. Keller. Searching for proto-planets with MUSE. A&A, 644:A149, Dec. 2020. doi: 10.1051/0004-6361/202038242.

J. W. Xuan, J. Wang, J.-B. Ruffio, H. Knutson, D. Mawet, P. Mollière, J. Kolecki, A. Vigan, S. Mukherjee, N. Wallack, J. Wang, A. Baker, R. Bartos, G. A. Blake, C. Z. Bond, M. Bryan, B. Calvin, S. Cetre, M. Chun, J.-R. Delorme, G. Doppmann, D. Echeverri, L. Finnerty, M. P. Fitzgerald, K. Horstman, J. Inglis, N. Jovanovic, R. López, E. C. Martin, E. Morris, J. Pezzato, S. Ragland, B. Ren, G. Ruane, B. Sappey, T. Schofield, A. Skemer, T. Venenciano, J. K. Wallace, and P. Wizinowich. A Clear View of a Cloudy Brown Dwarf Companion from High-resolution Spectroscopy. ApJ, 937(2):54, Oct. 2022. doi: 10.3847/1538-4357/ac8673.

J. E. Yin, D. J. Eisenstein, D. P. Finkbeiner, and P. Protopapas. A Conditional Autoencoder for Galaxy Photometric Parameter Estimation. PASP, 134(1034):044502, Apr. 2022. doi: 10.1088/1538-3873/ac5847.

W. Zhang, K. Itoh, J. Tanida, and Y. Ichioka. Parallel distributed processing model with local space-invariant interconnections and its optical architecture. Appl. Opt., 29(32): 4790–4797, Nov. 1990. doi: 10.1364/AO.29.004790.

G. Zhao, B. Qiu, A.-L. Luo, X. Guo, L. Yao, K. Wang, and Y. Liu. Deep Learning Applications Based on WISE Infrared Data: Classification of Stars, Galaxies and Quasars. *arXiv e-prints*, art. arXiv:2305.10217, May 2023. doi: 10.48550/arXiv.2305.10217.

S. Zucker. Cross-correlation and maximum-likelihood analysis: a new approach to combining cross-correlation functions. MNRAS, 342(4):1291–1298, July 2003. doi: 10.1046/j.1365-8711.2003.06633.x.