Master Thesis : Characterizing the performance of the SPHERE exoplanet imager at the Very Large Telescope using deep learning

Auteur : Bissot, Ludo
Promoteur(s) : Absil, Olivier; Louppe, Gilles
Faculté : Faculté des Sciences appliquées
Diplôme : Master : ingénieur civil en science des données, à finalité spécialisée
Année académique : 2022-2023
URI/URL : http://hdl.handle.net/2268.2/19340

Characterizing the performance of the SPHERE exoplanet imager at the Very Large Telescope using deep learning

Author:

Ludo Bissot

Supervisors:

Prof. O. Absil
Prof. G. Louppe

Thesis submitted for the degree of
Master of Science in Engineering with a
professional focus on "Data Science"

School of Engineering and Computer Science

University of Liège

Academic year 2022-2023

Characterizing the performance of the SPHERE exoplanet imager at the Very Large Telescope using deep learning

Ludo Bissot

Characterizing the performance of the SPHERE exoplanet imager at the Very Large Telescope using deep learning

https://matheo.uliege.be/

Abstract

The high-contrast imaging tool known as Spectro-Polarimetric High-contrast Exoplanet REsearch (SPHERE) represents a second-generation instrument specifically engineered for detecting exoplanets. It has been operational at the Very Large Telescope since 2014.

To harness the extensive dataset generated by SPHERE, enhance future observation scheduling, and further instrument development, it is imperative to gain a comprehensive understanding of how instrumental performance relates to various environmental factors.

This project's principal goal is to use machine learning and deep learning approaches to forecast detection limits in terms of contrast between exoplanets and their host stars. This endeavor will involve the creation of two distinct model types: random forest models and multilayer perceptron models. The ultimate aim is to enhance our comprehension of the connection between input parameters and detection limits, ultimately yielding deeper insights and knowledge within this field.

Furthermore, uncertainties associated with the input features will be captured using a neural network, with the aim of providing confidence intervals in the predictions.

Acknowledgements

I wish to begin by expressing my profound gratitude to my supervisors, Olivier Absil and Gilles Louppe, whose unwavering support, invaluable feedback, and the remarkable opportunity to visit the Laboratoire d'Astrophysique de Marseille have been instrumental throughout the year.

My heartfelt thanks extend to the wonderful individuals I had the privilege of meeting in Marseille. I am deeply appreciative of Elodie Choquet for her unwavering support during Faustine's recovery. To Faustine Cantalloube, I extend my gratitude for graciously hosting me when I was without accommodation, and I wish her a swift and full recovery. I would also like to thank Julien Milli for his dedication, help, and kindness. Additionally, I would like to convey my thanks to all the PhD students and young professionals for their warm welcome in the city.

I am also indebted to my friends Lucas, Martin, and Robin for their incisive critiques and invaluable assistance, which offered diverse perspectives on this highly specialized subject.

Lastly, my heartfelt thanks go out to all my friends and family for their unwavering support throughout this year.

This work has made use of the the SPHERE Data Centre, jointly operated by OSUG/IPAG (Grenoble), PYTHEAS/LAM/CESAM (Marseille), OCA/Lagrange (Nice), Observatoire de Paris/LESIA (Paris), and Observatoire de Lyon. [10] [16] [27]

Contents

List of Figures

List of Tables

Acronyms

ADI Angular Differential Imaging. 18, 34

AO Adaptive Optics. vi, 13, 15, 17, 19

ASM Astronomical Site Monitoring. 43

CDS Centre de Données astronomiques de Strasbourg. 42

DC Data Center. 36, 40

DIMM Differential Image Moption Monitor. 43

ELT Extremely Large Telescope. 32, 35

ESO European Southern Observatory. 13, 32, 43

FITS Flexible Image Transport System. iv, 18, 40

HCI High Contrast Imaging. 1, 11, 69

i.i.d. Independent and Identically Distributed. 22

IAU International Astronomical Union. 5, 6

IRDIS Infrared Dual-band Imager and Spectrograph. 37, 42

KNN K-Nearest Neighbors. 44, 45

LHATPRO Low Humidity And Temperature PROfiling microwave radiometer. 43

MAE Mean Absolute Error. 54, 61

MASS Multi-Aperture Scintillation Sensor. 43

ML Machine Learning. 38, 39

MLP Multi-Layer Perceptron. 26, 50, 69

MSE Mean Squared Error. vii–ix, 21, 30, 54, 56, 58, 59, 61, 62

NASA National Aeronautics and Space Administration. 1

NCPA Non-Common Path Aberrations. 17

PA Parallactic Angle. vii, 18, 32, 33, 39, 45

PCA Principal Component Analysis. 19

PSF Point Spread Function. 13, 15, 18, 19, 33, 38

RDI Reference-star Differential Imaging. 18, 32

ReLU Rectified Linear Unit. 50

RV Radial Velocities. 1, 9

SDI Spectral Differential Imaging. 18

SIMBAD Set of Identifications, Measurements, and Bibliography for Astronomical Data. 42

SLODAR SLOpe Detection And Ranging. 43

SNR Signal-to-Noise Ratio. 37

SPHERE Spectro-Polarimetric High- contrast Exoplanet REsearch. i, 2, 14–16, 35–37, 40, 69

SR Strehl Ratio. 14, 15

VLT Very Large Telescope. i, 2, 13–16, 31, 35, 43

Chapter 1

Introduction

Exoplanets, or planets outside our solar system, were only hypothetical for centuries, appearing exclusively in works of science fiction. The detection of these faraway planets, located light-years away, seems incredibly difficult due to their small size, as they would appear billions of times dimmer than their parent stars. The first definitive confirmation of an exoplanet detection took place in 1992. Exoplanet discoveries cleared the way for an emerging field of astronomy, resulting in a multitude of subsequent detections in the years that followed. The Radial Velocities (RV) approach was crucial in these discoveries at the time.

During the 2000s, further techniques were developed and employed for exoplanet detection. These included the successful transit method, microlensing, and High Contrast Imaging (HCI). These additional techniques expanded the range of methods available for studying and identifying exoplanets. Furthermore, as a result of these developments in the field, the rate of exoplanet discovery has increased rapidly.

According to the National Aeronautics and Space Administration (NASA) there have been 5,506 confirmed exoplanet detections in 4,065 planetary systems as of September 2023 [34]. Among these systems, 878 have multiple planets orbiting their host star [40].
The goal of exoplanet research is not solely to create a catalog of known worlds, but rather to assess their physical properties, and eventually determine whether life can exist elsewhere in the universe.

Detecting exoplanets using direct imaging methods poses significant challenges due to the small angular separation between the planet and its host star and the high contrast between them. These reasons explain why, despite technological advances, only massive planets at large angular separation have been identified so far. Ground-based telescopes use adaptive optics and coronagraphic devices as the primary technique to obtain both high contrast and high angular resolution.

The Very Large Telescope (VLT) is a facility managed by the European Southern Observatory and situated on Cerro Paranal in the Atacama Desert of northern Chile. It comprises four separate telescopes, each featuring a primary mirror with a diameter of 8.2 meters.

The high-contrast imaging instrument Spectro-Polarimetric High-contrast Exoplanet REsearch (SPHERE) is representative of a second generation of instruments designed for exoplanets detection and has been installed at the VLT since 2014. It combines adaptive optics and coronagraphic techniques and allows for direct imaging, spectroscopic analysis, and polarimetric characterization of exoplanet systems. This instrument, which operates in the visible and near infrared ranges, provides higher image quality and contrast for bright targets, although within a limited field of view.

The mission of the High Contrast Data Center is to process raw data on demand and handle public availability of all data gathered by the SPHERE instrument over the last nine years.

To fully exploit the vast SPHERE database, optimize future observation scheduling, and advance instrument development, it is crucial to thoroughly comprehend the relationship between instrumental performance and various environmental parameters. These parameters include atmospheric turbulence intensity, wind velocity, observation duration, pointing direction among others. Understanding these dependencies would allow us to optimize the potential of SPHERE's capabilities.

This project's principal goal is to use machine learning and deep learning approaches to forecast detection limits in terms of contrast between exoplanets and their host stars.
Two types of models will be created to achieve this goal: random forest models and multilayer perceptron models. The goal is to create a greater understanding of the relationship between input attributes and detection limits, resulting in better insights and knowledge in this domain.

Chapter 2 furnishes readers with a background in astronomy, specifically delving into the sub-field of exoplanet detection.

Chapter 3 is dedicated to providing readers with a comprehensive understanding of key machine learning concepts. This understanding is essential for comprehending the methodology employed in this thesis to construct the models.

In Chapter 4, readers can discover the objectives of this research, along with a review of the existing literature and results in the field.

Chapter 5 details the data collection process and the creation of the dataset used as input for the machine learning models.

Chapter 6 elucidates the methodology employed to construct consistent and smoothly operating models.

Chapter 7 presents the results of this work, accompanied by a discussion of their implications.

The concluding remarks can be found in Chapter 8.

Chapter 2

Astronomical background

Astronomy is the study of celestial objects and events via the use of mathematics, physics, and chemistry to explain their origins and changes. It includes visible-to-the-naked-eye objects such as the Sun, Moon, planets, and stars. It also includes objects that must be observed with telescopes or other devices, such as distant galaxies and microscopic particles. Furthermore, astronomy investigates unseen substances such as dark matter and dark energy, which cannot be directly witnessed. Essentially, astronomy studies everything outside Earth's atmosphere, whereas cosmology studies the entire cosmos.

Throughout history various civilizations have made systematic observations of the night sky. In the past, astronomy encompassed diverse areas such as celestial navigation, observational astronomy, and calendar creation.

Professional astronomy nowadays is separated into two main branches: observational and theoretical. Observational astronomy is the collection of data from observations of celestial objects, which is then analyzed using fundamental physics concepts. Theoretical astronomy, on the other hand, is concerned with constructing computer or analytical models to explain and characterize celestial objects and phenomena. These two branches work together in a complementary fashion, as theoretical astronomy seeks to account for observational findings, while observations validate theoretical predictions.

2.1 Exoplanet

A spectacle arises in the night sky as thousands of stars sparkle brightly and appear visible to the naked eye. There are billions more stars beyond this stunning display, their dim brightness escaping our direct awareness.

The sky appears to be a vast expanse abounding with celestial bodies, a magnificent tapestry of light.

Naturally, a series of profound questions emerge: Could these stars host planets in their orbits? Is it possible that among them, some might resemble our own Earth?

For hundreds of years, humanity has been investigating these enigmas, hoping for answers. Today, we are on the edge of discovering the truth, armed with knowledge and technology. And the overwhelming answer is a resounding "yes."

2.1.1 Definition

An exoplanet, by definition, refers to a planet that exists outside of our Solar System. To establish a comprehensive understanding, it is essential to define what constitutes a planet.

The term "planet" comes from the Ancient Greek word "planetes," which means "wanderer." In antiquity, any celestial body that moved across the night sky was considered a planet, even our own Moon.
During those times, the understanding of celestial objects and their behavior was limited, and the distinction between planets and other luminous entities was not clearly defined. Observers assigned the word "planet" to any celestial wanderer that moved around the heavens, frequently recognizing their movements against a backdrop of fixed stars.

The International Astronomical Union (IAU), in its 2006 definition, established a set of criteria that a celestial body must meet to be classified as a planet. This definition, however, is limited to the Solar System, making it inapplicable to exoplanets located outside of our celestial neighborhood. These criteria are as follows:

- Orbiting a Star: A planet should orbit a star, meaning it must revolve around a star as part of its regular motion.
- Spherical Shape: A planet should possess enough mass to generate sufficient self-gravity, allowing it to reach a nearly spherical shape. This condition arises due to the balance between the planet's gravity and the forces acting upon it.
- Clearing its Orbit: A planet must possess enough mass to clear its orbital neighborhood of significant debris or other celestial objects. This means that as a planet orbits its star, it has the gravitational influence to attract or deflect smaller objects in its vicinity and establish a relatively clear orbital path.

The IAU's definition of a planet focuses on smaller planets and excludes the region of transition between giant planets and brown dwarfs. An IAU working group addressed this distinction between exoplanets and brown dwarfs in 2007 by establishing an extra mass-based criteria, as proposed by Boss et al. 2005 [3].

- Objects with true masses below the limiting mass for thermonuclear fusion of deuterium (currently calculated to be 13 Jupiter masses for objects of solar metallicity) that orbit stars or stellar remnants are "planets" (no matter how they formed). The minimum mass/size required for an extrasolar object to be considered a planet should be the same as that used in the Solar System.

However, in August 2018, the IAU's Commission F2: Exoplanets and the Solar System made amendments to this working definition. The updated official definition of an exoplanet is now as follows:

- Objects below the deuterium fusion limit, orbiting stars, brown dwarfs, or stellar remnants, and having a mass ratio with the central object below a specific threshold known as the L4/L5 instability criterion ($M/M_{central} < 2/(25 + \sqrt{621})$) are classified as "planets," regardless of their formation mechanism.
- The minimum mass/size required for an extrasolar object to be considered a planet should align with the criteria used within our Solar System.

It is worth noting that the IAU acknowledged that this definition is subject to evolution as scientific knowledge advances and improves.

2.2 Indirect detection methods

With so many stars in the sky, it's becoming increasingly difficult to believe that our Sun is the solitary designer of planetary systems. Astronomers have been perplexed by this puzzle for a long time, yet the search for such planets is fraught with difficulties.

The most difficult challenge is the elusive character of these planets. They are usually far away from us and dimmed by the brilliance of their parent stars. Seeing such planets through a telescope is like trying to find a firefly located next to a blazing searchlight. Their faintness and close proximity to their host stars make the task even more difficult. Indeed, a natural thought arises: if direct observation of such planets proves difficult, perhaps an indirect approach can reveal their presence ?

2.2.1 Pulsar timing

The first widely accepted detection of extrasolar planets was made by Wolszczan in 1992 [51]. Earth-mass and even smaller planets orbiting a pulsar were detected by measuring the periodic variation in the pulse arrival time. Indeed, this method uses a frequency analysis of the time-periodic pulses released by the pulsar's magnetic poles to detect minor perturbations caused by the presence of a planet.

The planets detected are orbiting a pulsar, a "dead" star, rather than a dwarf (main-sequence) star. What is heartening about the detection is that the planets were probably formed after the supernova that resulted in the pulsar thereby demonstrating that planet formation is probably a common rather than a rare phenomena. However, no information about the observed exoplanet's origin can be gathered, thus the discovery does not reveal much information on planet formation processes.

2.2.2 Radial velocities

A planet orbiting a star has gravity as well, thus it has an influence on the star, causing it to rotate around the system's center of mass. An illustration of this phenomenon is depicted in Figure 2.1.

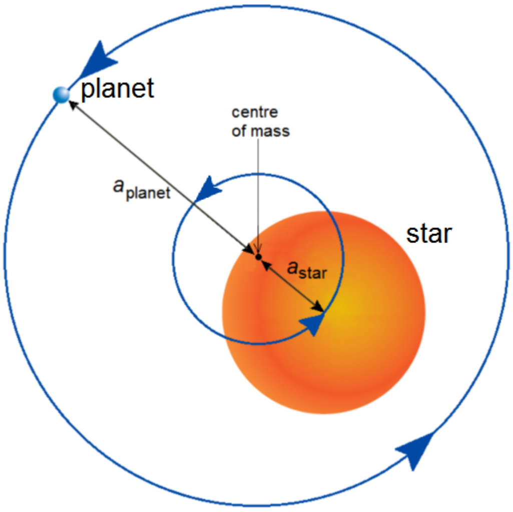


Figure 2.1: Reflexive motion [47]

Although it appears to be a promising idea, astronomers had long been studying this phenomenon in nearby stars. At that time, with the equipment available, the effect proved to be too challenging to be seen. The back-and-forth motion was indeed too subtle to be measured, but this does not imply that the effect was imperceptible.

When an exoplanet is big and/or close enough to its parent star, the center of mass shift becomes large enough to allow Doppler spectroscopy to detect the main star's periodic radial motion as shown in Figure 2.2. It is worth noting that this method also provides a lower bound on the mass of the planet since the orientation of the orbital plane is unknown.

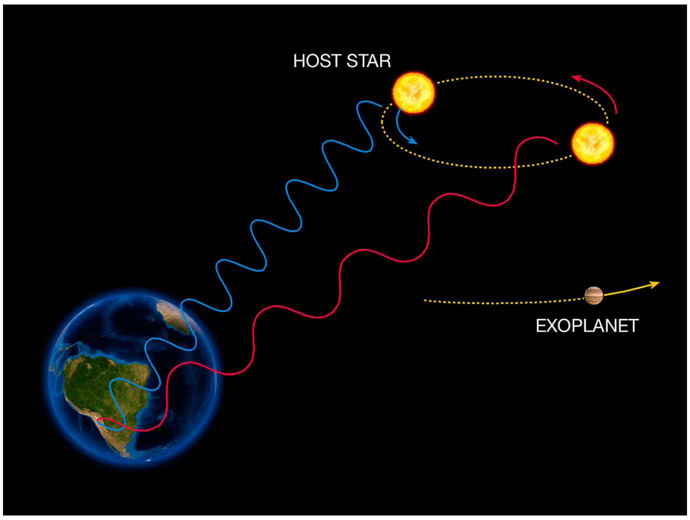


Figure 2.2: Radial Velocity [48]

This technique tends to detect massive planets close to their host star such as hot Jupiters since they induce a larger spectral shift.

It is also worth pointing out that the first exoplanet orbiting a main-sequence star, 51 Pegasi b, has been discovered using this indirect strategy [32].

2.2.3 Transit photometry

Transit photometry is a successful approach for detecting exoplanets that includes tracking the brightness of a star over time. As seen in Figure 2.3, when an exoplanet passes in front of its host star in the observer's line of sight, it temporarily blocks a portion of the star's light, causing a transient decrease in brightness.

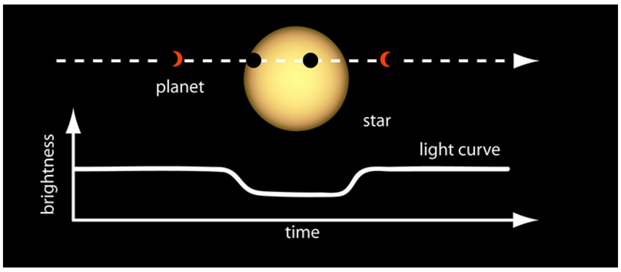


Figure 2.3: Light curve of a planet transiting its star [25]

Astronomers can deduce the presence of an exoplanet, establish its size, and even learn about its orbital period and direction relative to the sky plane by properly detecting these periodic dips, which are proportional to the star-planet surface ratio. Transit photometry has been used to identify thousands of exoplanets, and it is especially successful at spotting planets that are very close to their stars and have orbits that are lined with our line of sight. In fact, the fundamental limitation of the transit approach is the required tilt to observe the eclipse, with the likelihood of aligning with Earth's line of sight reducing as the separation grows.

2.2.4 Astrometry

Astrometry is the exact measuring of an object's location relative to reference background stars. In the case at hand, since the equipment available is becoming more and more sophisticated, the goal is to measure (precisely) the center of mass shift as seen in figure 2.1. The method offers information about some orbital properties of the exoplanet, such as inclination, and can thus be used in conjunction with Radial Velocities (RV) to precisely determine the companion mass. Moreover, similarly to RV, massive exoplanets orbiting low-mass stars are favoured.

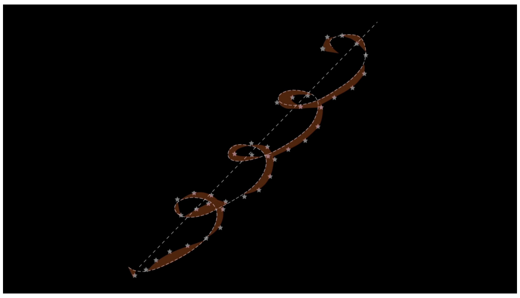


Figure 2.4: Oscillations of the star induced by an orbiting planet over time [11]

The precision required to detect a planet orbiting a star using this technique is exceedingly difficult to obtain and as a result, only one planet has been identified using this method. Nevertheless astrometry has been used to do follow-up studies for planets spotted using other methods.

The GAIA space-telescope, which was launched in 2013 and which is dedicated to astrometry, should drastically increase the number of exoplanets discovered using this method in the years to come.

2.2.5 Gravitational microlensing

While both the radial velocity and transit strategies rely on detecting fluctuations in starlight, a distinct approach incorporates the effect of gravity on light. Gravitational microlensing, which Albert Einstein first proposed in his general theory of relativity, is based on the notion that heavy objects can bend the course of light. When the stars are properly aligned, light traveling from a distant star to an observer can be twisted around an intermediate star, thus acting as a magnifying lens. This causes the background star's light to be amplified, and if a planet circles the lensing star, it causes a visible variation in what would otherwise be a continuous light curve. Figure 2.5 provides a graphical representation of this notion.

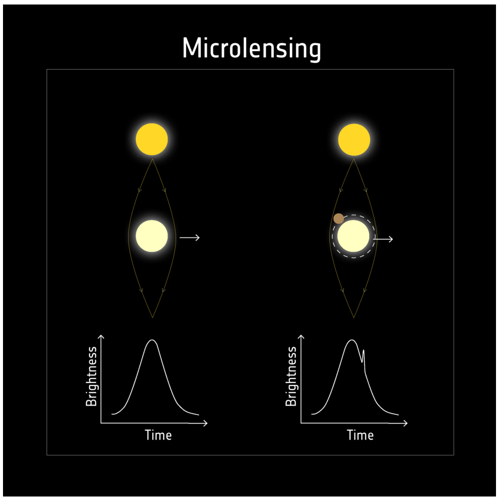
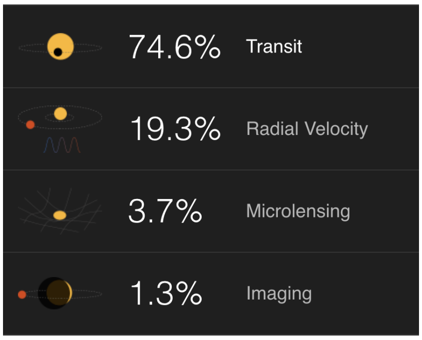


Figure 2.5: Exoplanet detection using microlensing [12]

Nonetheless, due to the rarity of such alignments, only a few dozens of exoplanet detections using this strategy have been made since the first identification in 2000 [2].

2.3 Direct imaging

Despite the fact that indirect detection methods have ruled exoplanet science for decades, recent experimental breakthroughs and novel data processing techniques have contributed to the rapid growth of direct imaging as a viable complementary detection method. Indeed since the method depends on receiving photons emitted by the exoplanet, a wider range of astrophysical parameters can be retrieved thus putting new constraints on planet formation models.
The figure in Figure 2.6 displays the proportion of exoplanets discovered through various methods.



74.6%	Transit
19.3%	Radial Velocity
3.7%	Microlensing
1.3%	Imaging

Figure 2.6: Planets detected using the different methods [34]

When compared to stars, planets are extremely dim, and the tiny light they emit is frequently overpowered by the brilliance of their parent stars. As a result, it is often difficult to directly see and distinguish planets from their home stars. As previously stated, direct imaging seeks to identify photons emitted (mostly in the infrared) or reflected (primarily in the visible) by the exoplanet itself.
The amount of visible light reflected by the exoplanet from the host star is determined by the planet's albedo[1], which is governed by the composition of its surface and atmosphere. When planets orbit at a great distance from their stars and reflect very little starlight, their detection is mostly based on their surface heat radiation which is affected by both the planet age and mass.

Images become more feasible when the star system is close to the Sun and the planet is noticeably massive, located far from its parent star, and has a high temperature, resulting in the emission of intense infrared radiation. The planet's brightness in this spectral range exceeds its visibility in visible wavelengths, hence infrared imaging is used.
Actually planetary evolution models can be used to provide a more accurate characterisation of the region within which astronomers perform their search. During their first phases, the planets' thermal emission peaks in the near-to-mid-infrared wavelength region, and as they mature, it shifts to the mid-infrared range. In High Contrast Imaging (HCI) near-infrared emissions are used as it provides a good trade-off between the noisy (due to the Earth's atmosphere temperature) mid-infrared bands and the visible regime which is more turbulent.

[1] Albedo is the fraction of starlight that is diffusely reflected by an object. It is quantified on a scale ranging from 0, representing a black body that absorbs all incoming radiation, to 1 which represents an object that reflects all incoming radiation.

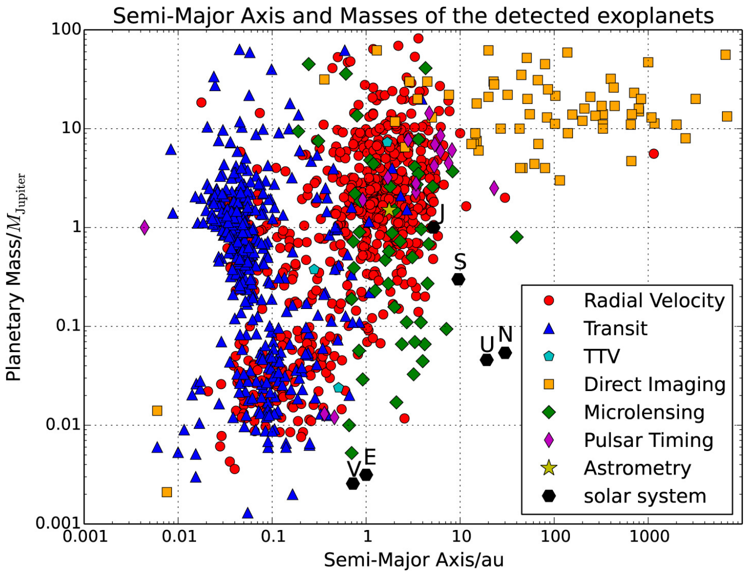


Figure 2.7: The semi-major axes and masses of the detected exoplanets [20]

As depicted in Figure 2.7, direct imaging allows for the finding of young and large exoplanets at separations that indirect approaches do not cover yet.
Actually the first ever planet detected via direct imaging was a giant planet situated near a low temperature brown dwarf [7]. Moreover the two celestial objects were separated by a relatively large angular separation, as shown in Figure 2.8, thus allowing the contrast between the two to be quite small.

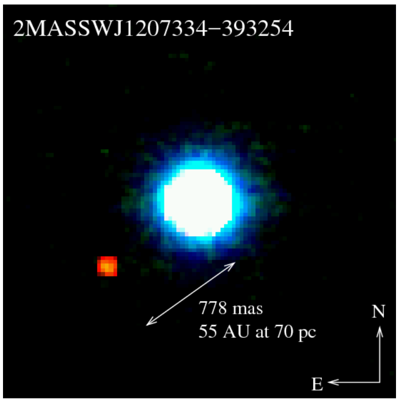


Figure 2.8: In comparison to the color of the brown dwarf 2M1207, the exoplanet is easily recognizable [7]

However, it's important to note that this object does not meet the mass criterion illustrated in Figure 2.7. The mass ratio with respect to Jupiter is approximately 5, whereas a mass ratio of 25 would be required to meet this criterion.
Actually, HR 8799 is recognized as the first main-sequence star where exoplanets orbiting it were directly imaged and identified [29].

2.3.1 Confronting Key Challenges in High-Contrast Imaging

Despite significant technological improvement over the last decade, only massive planets orbiting at great angular separation have been discovered so far. It comes from the fact that directly observing exoplanets is a tremendously challenging task to say the least. On the one hand the planet and its parent star are separated by a very small angular distance, typically ranging from 0.1 to a few arcseconds. On the other hand the contrast between them is quite high, it can goes from 10^{-3} for young massive planets emitting in the infrared to 10^{-10} for Earth-like exoplanets reflecting the light of their star.
In order to achieve both high contrast and high angular resolution, Adaptive Optics (AO) in conjunction with coronagraphic equipments have been used in ground-based telescopes.

Adaptive optics and angular resolution

Images acquired at world-class astronomical observatories such as Paranal, Chile, where ESO's VLT is located, are distorted by atmospheric turbulence. This turbulence causes stars to twinkle, which poets enjoy but annoys astronomers since it blurs cosmic details.

For a given wavelength, the resolution of a telescope is theoretically inversely proportional to the diameter of its primary aperture. When distant point-like sources, such as stars, are observed, they produce a diffraction pattern known as the Point Spread Function (PSF), which is characterized by a center peak surrounded by several lobes for a circular aperture, as specified by the Airy function [2]. In ideal conditions, the resolving power of a telescope is determined by the Rayleigh criterion, which sets a minimum separation between two resolved objects at 1.22 times the wavelength divided by the diameter of the telescope's primary mirror.
However, atmospheric turbulence affects ground-based telescopes, creating fluctuating phase shifts in the wavefront and affecting the sharpness of the PSF. This turbulence blurs the PSF core, lowering the angular resolution limit. The angular resolution in the presence of air turbulence is given by 0.98 times the wavelength divided by the Fried parameter r_0, which depends solely on the strength of atmospheric turbulence, and represents an equivalent telescope diameter for obtaining angular resolution. Because the Fried parameter scales with the wavelength as $\lambda^{6/5}$, the effect of atmospheric turbulence on angular resolution in the near-infrared is less pronounced than in the visible spectrum [9].

[2] Airy was the first to investigate the diffraction theory of point spread functions in the nineteenth century. He devised an equation for the amplitude and intensity of a perfect, aberration-free instrument's point spread function, the so-called Airy disc. The Airy disk and Airy pattern are descriptions of the best-focused spot of light that can be created by a perfect lens with a circular aperture. This spot is limited by the diffraction of light.

As a result, when atmospheric turbulence is taken into account, the angular resolution limit becomes independent of the aperture size which can severely limit the resolving capability of 8-meter class telescopes. While observing from space can solve this problem, the cost of running space telescopes limits the size and power of telescopes we can place beyond Earth's atmosphere.

To tackle this problem, astronomers have used a technique known as adaptive optics [37]. Sophisticated, deformable mirrors controlled by computers can adjust for turbulence in the Earth's atmosphere in real-time, resulting in images nearly as crisp as those gathered in space.

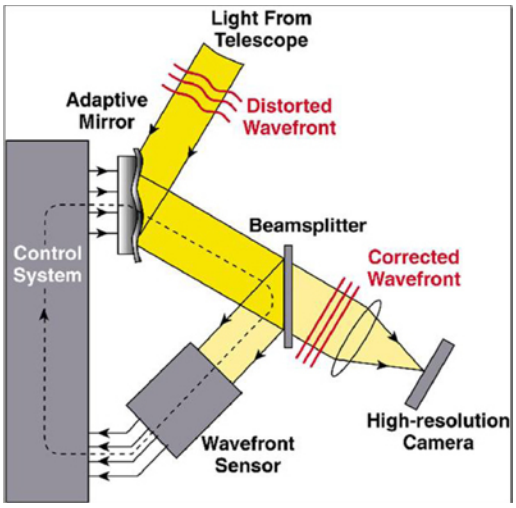
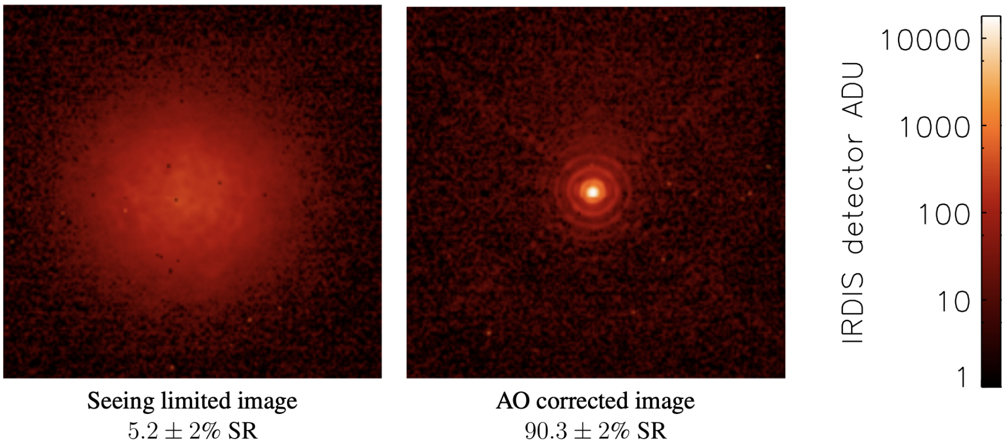


Figure 2.9: Principle of adaptive optics [35]

Figure 2.9 illustrates the main principles of adaptive optics. The deformable mirror, wavefront sensor, and control system, which includes a wavefront reconstructor, are the key adaptive optics components. A beam splitter directs a tiny portion of the light to the wavefront sensor while directing the majority of the light to the science instrument(s).

The VLT/SPHERE instrument, which uses extreme adaptive optics techniques to improve the Strehl Ratio (SR), a measure of optical system performance, representing the ratio of the actual peak intensity of an aberrated image to the maximum intensity achievable in an ideal diffraction-limited system, is the primary data source for this thesis [1]. It accomplishes this by focusing on a small region of the sky, using deformable mirrors with great density, and utilizing quick real-time processing.

In Figure 2.10, a long-exposure image in the H-band was captured by VLT/SPHERE during the assembly, integration, and testing phase under typical conditions of guide star brightness and atmospheric turbulence. The SR is provided at a wavelength of 1.6μm. The left image was taken in seeing-limited mode without AO correction while the right image was obtained with full AO correction under normal turbulence seeing conditions.



Seeing limited image
$5.2 \pm 2\%$ SR

AO corrected image
$90.3 \pm 2\%$ SR

Figure 2.10: Long-exposure image in the H-band with and without AO correction [39]

Coronagraphy and contrast

The second challenge concerns the brightness difference between the planet and its parent star. In order to remedy this issue, a coronagraph is introduced. A coronagraph is an optical device inserted into the telescope (or one of its back-end instruments) that is specifically designed to suppress or obscure the direct light generated by a star or other bright object. This reduction of strong light allows astronomers to examine and analyze adjacent objects that would otherwise be hidden by the central source's overwhelming glare.

In essence, regardless of how advanced and faultless a telescope is, diffraction changes what was once a tiny point of light in space into a circular shape encircled by concentric rings (PSF). Faint planets can be hidden within these rings. To solve this, a coronagraph is introduced to aid in the direct imaging of these dim planets by performing three major duties.

To begin, the coronagraph uses a mask with a dark core area to block the majority of incoming starlight. This mask is precisely engineered to direct the starlight that it does not block toward the beam's outer regions. Then, the coronagraph eliminates diffraction effects. Inserting a undersized stop (referred to as Lyot stop) in a pupil plane downstream of

the focal-plane occulting mask reduces significantly the brightness of the Airy pattern rings, causing the rings to vanish. As a result, the majority of the starlight is eliminated, allowing the sight of objects that are millions of times fainter than the star.
Finally, the light emitted by the companion(s) is not blurred because the telescope is precisely pointed at the star, causing the planet's light to approach at an angle that bypasses the mask and passes through the center of the Lyot stop. Figure 2.11 illustrates this whole process.

Instruments intended for high-contrast imaging usually contain a variety of coronagraphic instruments and several masks to cover a wide range of observational needs and wavelengths. Furthermore, these coronagraphic tools are often equipped with a tip-tilt sensor to precisely position the target and eliminate the possibility of instrument misalignment.
In addition to a differential tip-tilt sensor, the VLT/SPHERE instrument includes a typical Lyot coronagraph [26], a four-quadrant phase mask, and an apodized pupil Lyot coronagraph [31].

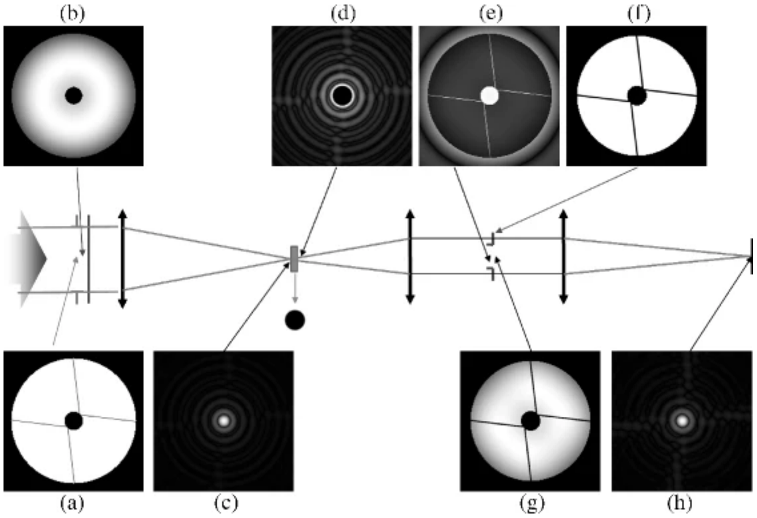


Figure 2.11: The SPHERE Apodized Lyot Coronagraph operates through a sequence of components: entrance pupil (a), apodizer (b), point spread function (c), Lyot occulting coronagraphic mask (d), initial pupil image (e), Lyot stop (f), pupil image with the stop (g), and the enhanced final coronagraphic PSF (h). This system effectively reduces starlight to unveil neighboring objects [19]

Speckle noise

Interference between wavefronts in coherent imaging systems produces the granular noise texture known as speckle, speckle pattern, or speckle noise. Despite the AO and the coronagraph some star light always makes its way to the detector, due in part to residual aberrations caused by uncorrected atmospheric turbulences or instrumental aberrations caused by the telescope and back-end instrument optical train. It is not possible to remove all of these aberrations, especially those that are not seen by the adaptive optics wavefront sensor referred to as Non-Common Path Aberrations (NCPA). Some of these NCPA are time dependent as a result of mechanical stress evolution, heat fluctuations during observation, unfiltered vibrations, or flaws in the instrument's moving parts. Since the variation of those instrumental mechanical characteristics is slow, the speckles induced by these types of aberrations can be viewed as quasi-static and they are not easy to remove.
To address this problem, astronomers rely on different observing strategies and post-processing techniques.

2.4 Observing strategies

The purpose of this section is to describe how the data is collected by the telescope, "Differential imaging" refers to taking multiple pictures of a same target in order to create a model of the speckle field.

Angular differential imaging

A timelapse view of the night sky illustrates that stars seem to be rotating around the (north or south) celestial pole as well as rise and set. In contrast, the field of view of a modern telescope, which tracks stars across the sky by spinning around two axes, remains parallel to the horizon, if not corrected by a specific optical device called a derotator. As a result, the images captured by the telescope appear to rotate while the quasi-static speckles, originating in the telescope and instrument optics (fixed with respect to the detector), maintain their places in the image. Thus if several pictures of a star and its companion are taken, the planet appears to rotate around its parent star.

Using the obtained data and angular diversity, it is feasible to develop a model of the speckle field, generally referred to as the reference PSF or simply background. This reference PSF is then subtracted from the collection of Angular Differential Imaging (ADI) [28] pictures. The remaining frames are then meticulously derotated and added toghether in order to find potential signals from exoplanets or disks. These signals should ideally be unaffected by the reference PSF subtraction, while any residual noise tends to average out incoherently.

Spectral differential imaging

Spectral Differential Imaging (SDI) takes advantage of the fact that light from stars and planets can have different spectral properties [42]. Stars have a relatively smooth and continuous spectrum of light while planets frequently have various spectral properties due to their composition and atmosphere. In addition, it is important to note that speckles exhibit linear wavelength-dependent stretching, whereas the image of an exoplanet remains stationary within the field of view across different wavelengths. SDI primarily uses this characteristic to identify and detect the faint light from exoplanets.

This enables for good speckle noise suppression while avoiding self-cancellation of the planetary signal during subtraction.

Reference-star differential imaging

Reference-star Differential Imaging (RDI) relies on observations of different stars to model the speckle field, whose brightness is then adapted and removed from the image.

RDI attempts to address one of the major limitations of both ADI and SDI observation methodologies, namely the over-subtraction observed, particularly at short separation (due to lower Parallactic Angle (PA) rotation for ADI and less effective rescaling for SDI) [52].

2.5 Data processing

Data processing is an important part of high-contrast exoplanet imaging. Its importance is almost equal to that of selecting a coronagraph or a wavefront control system, and it is linked with the observing technique.

As seen in Figure 2.12 a lot of steps are involved in order to process the data, from a raw FITS file given by the telescope to the residual image. The pipeline can be separated into two main steps: pre-processing and post-processing of the data.

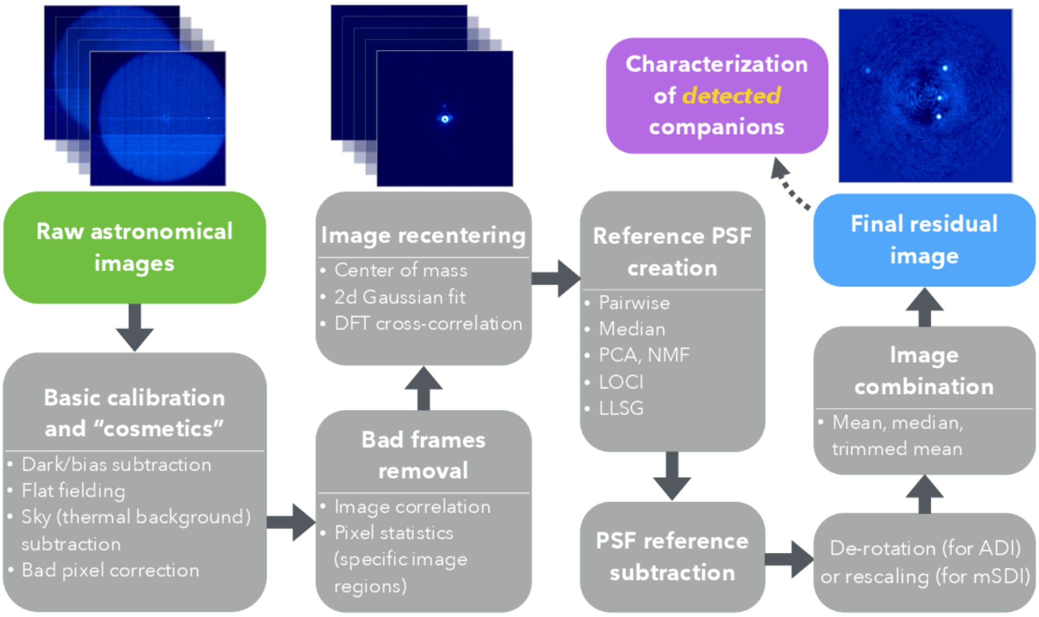


Figure 2.12: Typical differential imaging pipeline, from unprocessed images from the telescope through the development of a residual flux image [17]

The pre-processing steps regroup the calibration of the images, the dark current subtraction, the flat field correction, the thermal background (from the sky) subtraction and the bad pixel correction. Subsequently, poorly captured images resulting from star or coronagraph misalignment, unfavorable observation conditions, or errors in AO correction are identified using image correlation analysis or pixel statistics analysis within a specific portion of the field of view. After that, these undesirable frames are eliminated from the dataset. Finally, the images are recentered in order not to encounter problems when trying to model the reference PSF in the post-processing steps.

The main goal of the post-processing is to maximize the signal to noise ratio of the exoplanet. The later is achieved by tackling the noise introduced by AO correction errors, residuals light from the coronagraph or non-common path aberrations. The post-processing techniques can be divided into three main types :

- **Maximum likelihood techniques :** ANDROMEDA [6], PACO [15], TRAP [38], ...
- **PSF subtraction techniques :** Median subtraction [28], LOCI [24], PCA [43], Non-negative matrix factorization, ...
- **Supervised machine learning techniques :** SODINN [18], ...

Chapter 3

Machine Learning Background

Given the interdisciplinary character of this study, the goal of this part is to provide the reader with critical information for understanding the modeling process. However, it is critical to avoid viewing machine learning as a collection of magical algorithms capable of simply acquiring information from any input. Attempting machine learning without a thorough understanding of its principles and methodology can result in worthless results, misinterpretation of outcomes, and even unexpected consequences, highlighting the significance of acquiring a solid understanding of the area before embarking on these types of projects.

3.1 Definition

Machine Learning is a branch of artificial intelligence that focuses on the development of algorithms and models that allow computer systems to improve their performance at some tasks by learning from data rather than through explicit programming. Furthermore, machine learning approaches are frequently classified into three types based on the type of signal or feedback accessible to the learning system.

In the case at hand and probably the most well-known type of machine learning is called supervised learning. Supervised learning methods create a mathematical representation of a dataset that contains both the input data and the desired outputs. This dataset, which consists of a collection of training cases, is referred to as the training data. Each training instance has one or more inputs as well as the desired output, often known as a supervisory signal. Each training instance is represented as an array or vector within the mathematical model, sometimes referred to as a feature vector, while the training data is described as a matrix.

Supervised learning algorithms acquire knowledge by iteratively optimizing an objective function in order to construct a function capable of predicting the outcomes associated with a given set of inputs.

In opposition to supervised learning, unsupervised learning algorithms operate on datasets that are entirely made up of input data, looking for patterns or structures within the data, such as data point grouping or clustering. These algorithms learn from test data that has not been labeled, classified, or categorized. Unsupervised learning algorithms recognize similarities or shared qualities in data and change their responses based on the presence or absence of these shared features in each new data point, rather than relying on explicit feedback.

Finally, the third category of machine learning sub-field is referred as reinforcement learning. Reinforcement learning consists in a computer program, or agent, who has an objective within a dynamical environment (e.g. drive a car or operating a robot). While exploring its problem space, the program receives input in the form of a reward signal, which it tries to maximize.

3.2 Principles of (supervised) Machine Learning

This section will revisit key principles of statistical learning, specifically the concepts of empirical risk minimization and the distinction between underfitting and overfitting. These principles serve as the guiding principles that enable algorithms to learn, adapt, and make informed predictions from (labeled) data.

3.2.1 Empirical risk minimization

In the context where data samples are independently and identically drawn from an unknown joint probability distribution, denoted as $(\mathbf{x}_i, y_i) \sim p_{X,Y}$, with $\mathbf{x}_i \in \mathcal{X}$, $y_i \in \mathcal{Y}$, and $i = 1, ..., N$, the primary concern is to estimate the conditional probability $p(Y = y|X = \mathbf{x})$.

A learning algorithm typically generates a function represented as $f : \mathcal{X} \to \mathcal{Y}$. To assess how closely the predictions made by this function align with the original data, a loss function can be defined as $l : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$, for example, the Mean Squared Error (MSE).

When the hypothesis space, or the set of functions that can be generated by the selected learning algorithm, is represented as $\mathcal{F}$, and the expected risk is expressed as:

$$R(f) = \mathbb{E}_{(\mathbf{x},y)\sim p_{X,Y}}[l(y, f(\mathbf{X}))]$$

The ultimate objective is to discover a function $f^* \in \mathcal{F}$ that minimizes the expected risk, formulated as:

$$f^* = argmin_{f\in\mathcal{F}}R(f)$$

However, $p_{X,Y}$ is unknown thus the expected risk cannot be evaluated and the optimal model cannot be determined. Nevertheless, the training data $\mathbf{d} = \{(\mathbf{x}_i, y_i)|i = 1, ..., N\}$ is i.i.d. thus it is possible to compute an unbiased estimator of the expected risk called the empirical risk.

$$\hat{R}(f, \mathbf{d}) = \frac{1}{N} \sum_{(\mathbf{x}_i,y_i)\in\mathbf{d}} l(y_i, f(\mathbf{x}_i))$$

In this context, $f_*^d = argmin_{f\in\mathcal{F}}\hat{R}(f, \mathbf{d})$ and, under regularity assumptions, $\lim_{N\to\infty} f_*^d = f^*$.

3.2.2 Under-fitting and over-fitting

The Bayes risk, denoted as R_B, is defined as the minimum expected risk over all possible functions within the hypothesis space:

$$R_B = \min_{f\in\mathcal{Y}^\mathcal{X}} R(f)$$

Here, $\mathcal{Y}^\mathcal{X}$ represents the set of all functions $f : \mathcal{X} \to \mathcal{Y}$.

The Bayes model, which minimizes the Bayes risk, is typically denoted as f_B, and it's evident that no other model can achieve a lower risk. In essence, the Bayes model represents the best achievable performance given the specific problem and data distribution, making it an essential reference point for evaluating the effectiveness of other models.

The capacity of a hypothesis space induced by a learning algorithm intuitively reflects its ability to find an appropriate model, represented by a function $f \in \mathcal{F}$, for any underlying function, regardless of its complexity. In essence, it measures the flexibility and expressive power of the hypothesis space, indicating whether it can effectively capture and represent a wide range of functions.

If the capacity of the hypothesis space $\mathcal{F}$ is too low, it means that the Bayes model f_B may not be within $\mathcal{F}$, and the difference between the expected risk $R(f)$ and the Bayes risk R_B tends to be large for any f in $\mathcal{F}$, including f^* and f_*^d. In this context, models f are considered to underfit the data. Under-fitting occurs when the hypothesis space is not expressive enough to capture the complexity of the true underlying function, leading to poor generalization performance and high training error.

When the capacity of the hypothesis space $\mathcal{F}$ is excessively high, it's possible that f_B is within $\mathcal{F}$ or the difference between the expected risk $R(f^*)$ and the Bayes risk R_B is small. However, due to the high capacity of the hypothesis space, the empirical risk minimizer f_*^d could fit the training data exceptionally well, to the point that:

$$R(f_*^d) \geq R_B \geq \hat{R}(f_*^d, \mathbf{d}) \geq 0 \tag{3.1}$$

In this scenario, f_*^d becomes overly specialized with respect to the true data generating process, and a significant reduction in the empirical risk often comes at the cost of an increase in the expected risk of the empirical risk minimizer, $R(f_*^d)$. In such cases, f_*^d is said to overfit the data. Overfitting occurs when a model captures noise and idiosyncrasies in the training data, resulting in poor generalization to new, unseen data.

A schematic representation of these two phenomenons is depicted in figure 3.1.

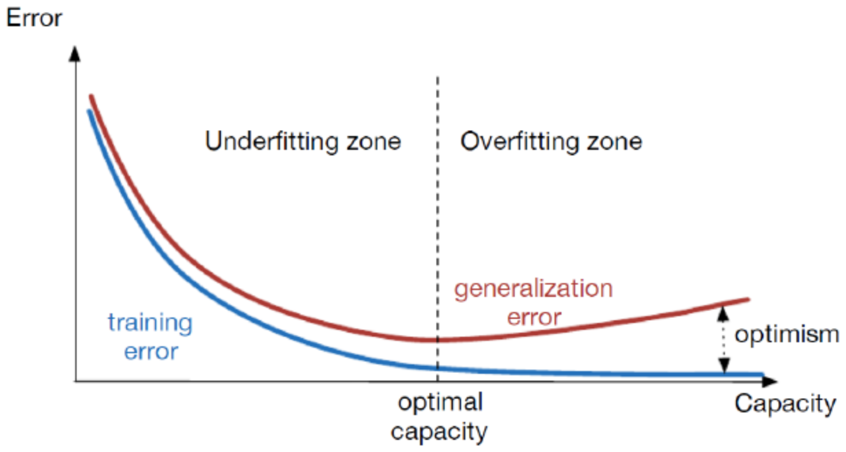


Figure 3.1: Under-fitting and over-fitting [21]

When over-fitting, as seen in equation 3.1, $\hat{R}(f_*^d, \mathbf{d})$ becomes a poor estimator of the expected risk $R(f_*^d)$. Nevertheless, an unbiased estimate of the expected risk can be obtained by assessing the performance of f_*^d on a separate set of data, denoted as $\mathbf{d}_{\text{test}}$, which is independent from the training samples in $\mathbf{d}$. This evaluation on a different dataset helps provide an estimate of how well the model generalizes to new, unseen data and can give a more accurate indication of its true performance.

This process helps mitigate the risk of over-fitting by assessing the model's ability to make predictions on data it hasn't been explicitly trained on.

In practical machine learning applications, it is common to partition the dataset into three distinct subsets: the training, the validation, and the testing sets. The first one is employed to train the machine learning model, allowing it to learn patterns and relationships within the data. The second set serves as a crucial component for assessing the model's generalization performance. It helps in preventing underfitting or overfitting of the model and can also be used for tuning hyperparameters. Finally, the testing set is exclusively reserved for the final evaluation of the trained model's performance. It provides an independent measure of how well the model is expected to perform on new, unseen data.

This three-set division ensures a systematic and rigorous approach to model development and evaluation in machine learning and a proper protocol is presented in figure 3.2.

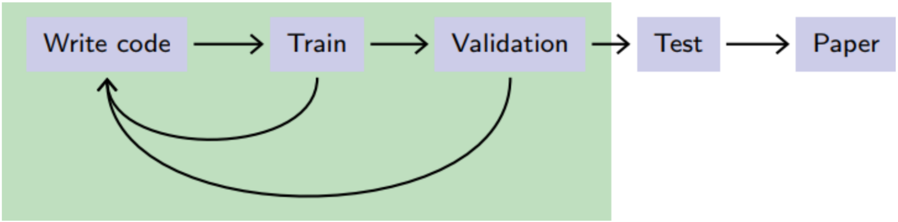


Figure 3.2: Proper protocol to build a machine learning model [13]

3.3 Models

Models play an important role in assisting intelligent decision-making and prediction in the field of machine learning. This section goes into two popular models: Random Forests [5], which are known for their ensemble capabilities, and Neural Networks [23], which are inspired by the complexities of the human brain. Although widely applied across diverse domains, the emphasis here is on their utilization within the framework of supervised machine learning.

3.3.1 Random Forest

A random forest is a machine learning ensemble method used for both classification and regression tasks, it combines the predictions of multiple decision trees to produce a more accurate and robust result.

A decision tree is a hierarchical structure in which each interior node tests an attribute or feature, with each branch corresponding to an attribute value, and each leaf either labeled with a class in the context of classification or a value in the context of regression. An illustrative example of a decision tree is presented in Figure 3.3. In this example, the features are numerical, necessitating the selection of a value to split each node effectively. To achieve the optimal split, the algorithm must identify the best feature and value combination for the split, aiming to maximize the purity of the subsequent nodes or, in other words, minimize the impurity measure. For a given impurity measure (e.g. Shannon entropy), the best splitting attribute is the one which maximizes the expected reduction of impurity.

The primary advantage of decision trees is their ease of use and interpretability. However, they are not robust estimators, making them susceptible to overfitting, and they tend to perform less effectively on average compared to more complex models when applied to similar data.

Random forests address these issues by improving robustness and reducing overfitting. However, to achieve these benefits, they sacrifice the inherent interpretability that decision trees offer. Nevertheless, it is worth noting that random forests can be used to naturally order the relevance of variables in a regression or classification problem [5].

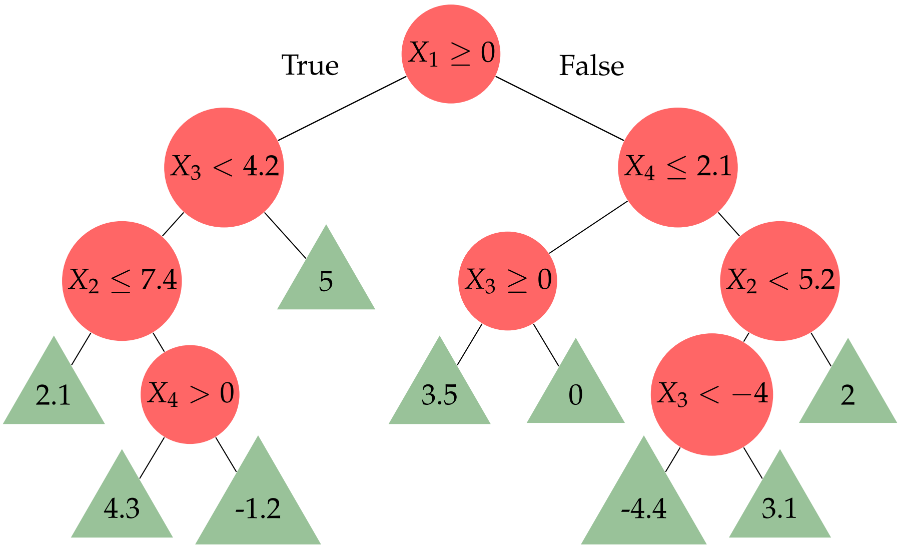


Figure 3.3: Example of a regression tree

3.3.2 Artificial Neural Network

The primary intuition behind artificial neural networks is the idea that, since biological brains can learn and adapt, an algorithm inspired by the functioning of the brain should be able to do the same.

Artificial neural networks attempt to simulate the connections between neurons by assigning varying strengths or weights to these connections, allowing them to become more or less influential in processing information and making predictions. This concept is the foundation for the learning and adaptation capabilities of neural networks.

An artificial neuron is a mathematical function designed to model and simulate the behavior of biological neurons. The first mathematical model of a neuron in history is known as the Threshold Logic Unit, as documented in McCulloch's work in 1943 [30]. Essentially, this function produces an output of one when the weighted sum of its Boolean inputs exceeds zero, and it yields zero otherwise. A more generalized version of this function, which can handle real-number inputs, is called a perceptron, introduced by Rosenblatt in 1957 [36]. The perceptron's classification rule can be expressed as:

$$f(\mathbf{x}) = \sigma(\sum_i w_i x_i + b)$$

Here, σ represents a non-linear activation function, such as the sign function or the sigmoid function. The perceptron unit serves as the fundamental building block for all neural networks, and a schematic representation of this unit can be found in Figure 3.4.

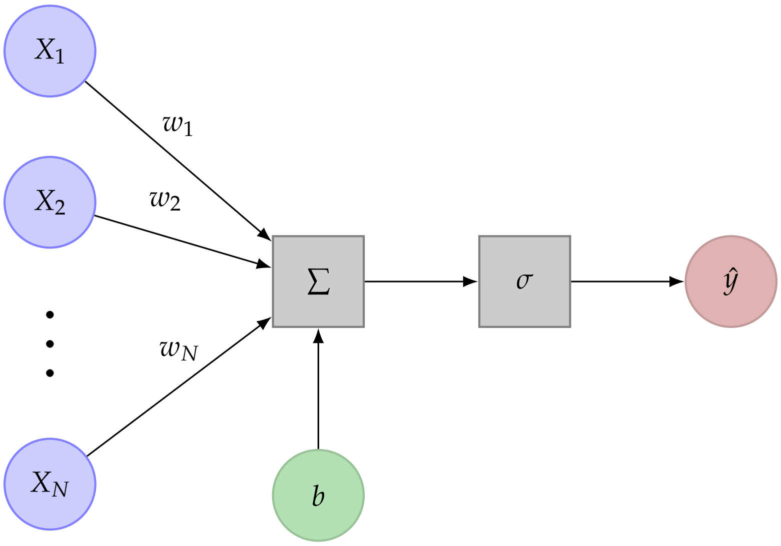


Figure 3.4: Graphical representation of a perceptron

Single neuron models do not offer greater expressiveness compared to linear models. However, they can be interconnected to create a potentially intricate non-linear parametric model known as a Multi-Layer Perceptron (MLP). A basic MLP is illustrated in Figure 3.5, where neurons are organized in layers that are interconnected.

The configuration of the layers, the number of units in each layer, and the number of outputs are specific to the problem at hand and are referred to as hyperparameters.

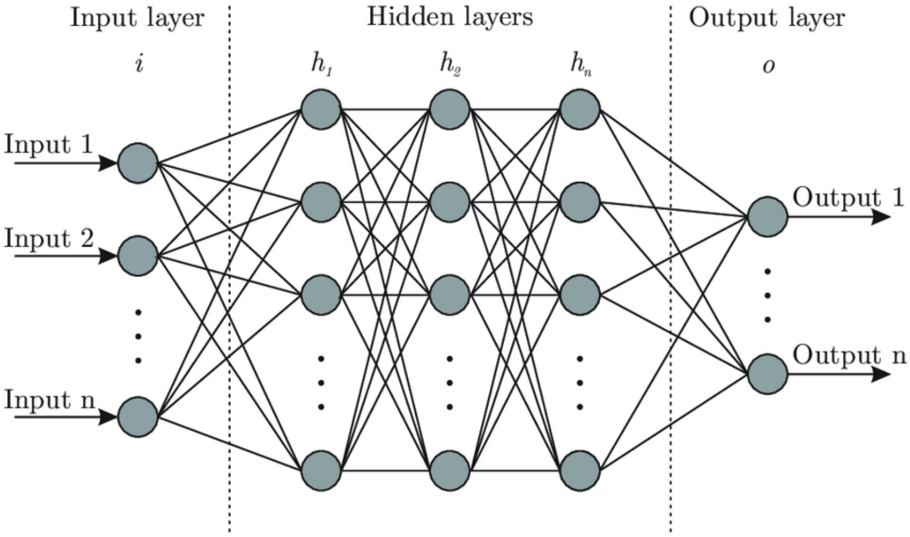


Figure 3.5: Graphical representation of an artificial neural network [4]

The loss function serves as a mathematical tool for quantifying and measuring the discrepancy between the predictions generated by the neural network during the forward pass, denoted as $\hat{y}$, and the true target values, represented as y.
In regression problems, the typical assumption is that the conditional distribution of the target variable y given the input $\mathbf{x}$ follows a normal distribution $p(y|\mathbf{x}) = \mathcal{N}(y; \mu = f(\mathbf{x}, \theta), \sigma^2 = 1)$. In this equation, f is parameterized by a neural network, and it's important to note that the last layer of this neural network does not contain a final activation function. Using maximum likelihood, it comes :

$$\begin{aligned} argmax_\theta p(\mathbf{d}|\theta) &= argmax_\theta \textstyle\prod_{\mathbf{x_i}, y_i \in \mathbf{d}} p(y_i|\mathbf{x_i}, \theta) \\ &= argmin_\theta \textstyle\sum_{\mathbf{x_i}, y_i \in \mathbf{d}} (y_i - f(\mathbf{x}, \theta))^2 \end{aligned} \tag{3.2}$$

The common mean-squared error $l(y, \hat{y}) = (y - \hat{y})^2$ is recovered.

Generally, these loss functions cannot be minimized analytically in a closed form. Nevertheless, numerical minimization methods, such as gradient descent, can be employed to find an optimal solution. Gradient descent works by iteratively adjusting the model's parameters. It starts with an initial parameter set called $\theta_0 \in \mathbb{R}^d$ and creates an approximation of the loss function near this point.

$$\hat{\mathcal{L}}(\epsilon, \theta_0) = \mathcal{L}(\theta_0) + \epsilon^T \nabla_\theta \mathcal{L}(\theta_0) + \frac{1}{2\gamma} ||\epsilon||^2$$

This approximation is a quadratic equation. The key to updating the parameters effectively is to calculate the gradient of this approximation.

$$\nabla_\epsilon \hat{\mathcal{L}}(\epsilon, \theta_0) = \nabla_\theta \mathcal{L}(\theta_0) + \frac{1}{\gamma}\epsilon = 0$$

The best step for improvement is found to be $\epsilon = -\gamma \nabla_\theta \mathcal{L}(\theta_0)$ where γ is the learning rate.

$$\theta_{t+1} = \theta_t - \gamma \nabla_\theta \mathcal{L}(\theta_{t-1})$$

This step is applied repeatedly to update the model's parameters. The choice of the initial parameter set (θ_0) and the learning rate (γ) is crucial for the convergence of the optimization process.

Given that a neural network is a composition of differentiable functions, it becomes possible to compute the total derivatives of the loss by working backward through the network. This process involves recursively applying the chain rule across its computational graph. The specific implementation of this procedure is commonly referred to as reverse automatic differentiation or backpropagation.

Finally, an epoch refers to one complete pass through the entire training dataset during the training phase of a model. During each epoch, the model is exposed to every example in the training dataset exactly once, and it updates its parameters (weights and biases) based on the observed errors or loss on those observations.

3.4 Uncertainty

Uncertainty refers to situations in which there is a lack of complete knowledge or certainty due to poor or partial information. This notion applies to a variety of situations, including future event forecasts, measurements of physical quantities with intrinsic variability, and regions where information is unclear or not fully understood. Uncertainty can develop in partially observable situations or in systems influenced by stochastic processes. Furthermore, it can be caused by a combination of factors such as as ignorance, where relevant information is unknown, and laziness, where one does not make the effort to adequately obtain or analyze accessible information.

Uncertainty can be categorized into two main types: aleatoric and epistemic.

Epistemic uncertainty accounts for uncertainty related to the model itself or its parameters. It arises from our lack of knowledge or understanding about which model can best explain the collected data. Epistemic uncertainty can be reduced or explained away with more data or better model refinement. Essentially, it reflects the uncertainty that can be addressed through improved modeling or increased knowledge.

Aleatoric uncertainty, on the other hand, captures inherent noise in the observations or data. This noise can stem from various sources, such as sensor inaccuracies, measurement errors, or stochastic processes. Unlike epistemic uncertainty, aleatoric uncertainty cannot be reduced by gathering more data. However, it could potentially be reduced by improving the quality of measurements or reducing the sources of noise.

Understanding and distinguishing between these two types of uncertainty is crucial in various fields, including statistics, machine learning, and scientific research, as they have different implications for decision-making and model improvement.

In this context, it's crucial to emphasize that when addressing uncertainty in this work, the specific focus will be on aleatoric uncertainty, as this is the type of uncertainty being targeted for modeling. Furthermore, aleatoric uncertainty can be further categorized into two sub-types: homoscedastic uncertainty, which remains constant for all inputs, and heteroscedastic uncertainty, which depends on the inputs of the model. In the current case, it is important to note that the type of aleatoric uncertainty being dealt with is heteroscedastic, meaning that it varies depending on the specific inputs to the model.

Instead of producing point estimates $\hat{y} = f(\mathbf{x})$, the goal is to model the full conditionnal density $p(y|\mathbf{x})$. Assuming that the distribution is Gaussian it comes :

$$p(y|\mathbf{x}) = \mathcal{N}(y; \mu(\mathbf{x}), \sigma^2(\mathbf{x}))$$

where $\mu(\mathbf{x})$ and $\sigma^2(\mathbf{x})$ are two parametric functions to be learned (by a neural network for instance).

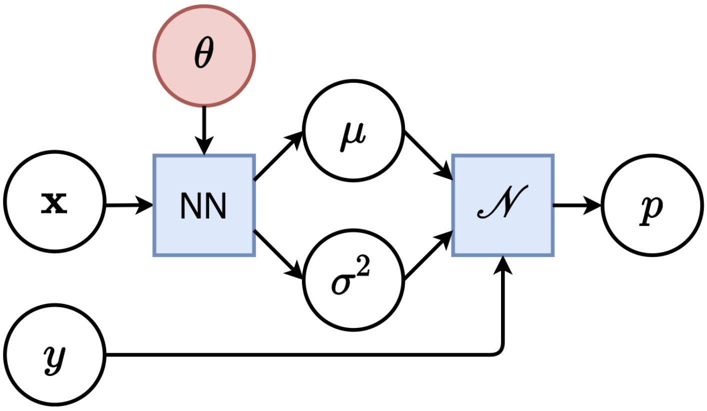


Figure 3.6: Modeling of the heteroscedastic aleatoric uncertainty using a neural network [21]

In order to be trained, this neural network needs an appropriate loss function that can be derived using maximum likelihood. If the training data is denoted by **d** and the parameters of the neural networks are denoted by θ, then it comes

$$\begin{aligned} argmax_\theta p(\mathbf{d}|\theta) &= argmax_\theta \prod_{\mathbf{x_i},y_i \in \mathbf{d}} p(y_i|\mathbf{x_i},\theta) \\ &= argmin_\theta \sum_{\mathbf{x_i},y_i \in \mathbf{d}} \frac{(y_i-\mu(\mathbf{x_i}))^2}{2\sigma^2(\mathbf{x_i})} + \log(\sigma(\mathbf{x_i})) + C \end{aligned} \tag{3.3}$$

It's worth noting how the term $(y_i - \mu(\mathbf{x_i}))^2$ has been reintroduced, it represents the squared difference between the actual value y_i and the predicted mean $\mu(\mathbf{x_i})$, into the classic Mean Squared Error (MSE). In this modified form, the MSE is embellished with additional terms that depend on the standard deviation, providing a more comprehensive representation of the error or uncertainty in the prediction.

Chapter 4

Scope of this work

The primary objectives of this chapter are twofold. Firstly, it aims to provide a concise overview of the key purposes of the research conducted in this thesis. Secondly, it seeks to present the existing work and research findings within the relevant field.

4.1 Problem statement

In the early third millennium, the Very Large Telescope is the showpiece of European astronomical equipment. It is the world's most advanced visible-light viewing facility. The VLT consists of four Unit Telescopes, each with a primary mirror 8.2 meters in diameter, and four Auxiliary Telescopes, each with a 1.8 meter mirror. All of these telescopes can work together to form a massive interferometer known as the VLTI, allowing astronomers to distinguish features with up to 25 times the precision of utilizing the telescopes individually [49].

The current operation method of this facility is suboptimal. A designated astronomer is required to stay awake throughout the night, overseeing the handling of observation requests. These requests, which often necessitate specific atmospheric conditions, are managed through an observation queue, which is manually administered, based on the suggestions from automatic tools that identify relevant observing programs for the current observing conditions. Typically, these observations span approximately one hour each. Furthermore, the quality of the observation can only be assessed once the observation is completed, and in some cases, insufficient data may result in a significant waste of observation time.

Despite the less-than-optimal nature of this observation scheduling approach, it is functional for the VLT, primarily because of the availability of multiple mirrors and other telescopes.

For the Extremely Large Telescope (ELT), it's evident that the current observation scheduling method needs improvement, primarily because there will only be one telescope of this caliber worldwide. Moreover, the estimated cost of a 10-hour observing night is several hundred thousand euros, making it financially significant to ensure a reliable and effective outcome.

Within the European Southern Observatory, a working group has been established to enhance the management of telescope usage and to predict the potential outcomes of observations (in particular for high-contrast imaging programs, for which sensitivity limits are notoriously difficult to predict accurately). To achieve this, the group aims to gain a deeper understanding of the factors that impact observation quality. These factors encompass atmospheric conditions, equipment used, and data processing procedures. In essence, if there exists any bottleneck or limiting factor at any stage of the observation process, the working group intends to identify it.

In this specific context, the objectives of this work are defined. Machine learning techniques will be employed to improve the prediction of detection limits in terms of the contrast between an exoplanet and its parent star. Additionally, the work seeks to better comprehend the parameters that can influence these detection limits.

4.2 State of the art

A similar research effort was conducted by Xuan et al. in 2018, as documented in their paper [53]. The objective of their study was to assess the performance of the vortex coronagraph installed on NIRC2, an instrument mounted on Keck II. This instrument is specifically designed for direct imaging of exoplanets and circumstellar disks in the near- to mid-infrared regime, including L' and M_s.

Their dataset comprises a total of 359 observations, covering 304 distinct targets that were observed between December 26, 2015, and January 5, 2018. This dataset consists of images obtained in both the L' and M_s bandpasses. However, for the purposes of their paper, they have chosen to focus solely on the targets observed in the L' band, which constitutes more than 98% of the data. Notably, approximately two-thirds of their sample consists of stars from surveys that were specifically designed for use with Reference-star Differential Imaging and have limited Parallactic Angle rotation (as it can be seen in figure 4.1).

A list of the explanatory variables for the contrast response is provided in Table 4.1.

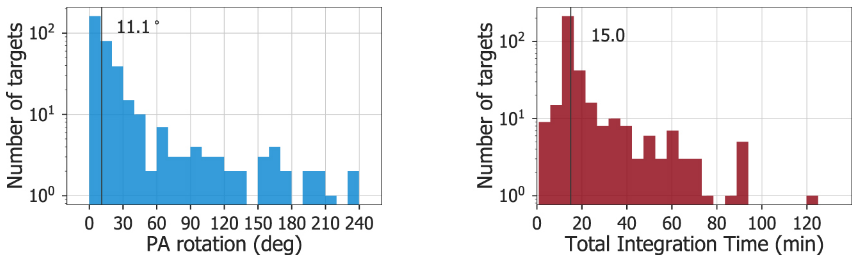


Figure 4.1: Distributions of the amount of PA and total integration time [53]

Variable	**Source**
Observing conditions	
τ_0	AO Telemetry
Seeing	AO Telemetry
WFS Frame Rate	AO Telemetry
Airmass	Fits Header
Primary Mirror Temperature	Keck II sensors
AO Optical Bench Temperature	Keck II sensors
AO Acquisition Camera Enclosure Temperature	Keck II sensors
Dome Temperature	Keck II sensors
Dome Humidity	Keck II sensors
Wind Speed	Keck II sensors
Pressure	Keck Weather Station
Observation Parameters	
PA Rotation	Fits Header
PSF x FWHM	Pipeline Product
PSF y FWHM	Pipeline Product
Total Science Integration Time	Fits Header
RDI Reference Library Size	Pipeline Product
Stellar Magnitudes	
R magnitude	UCAC4
W1 magnitude	WISE All-Sky and AllWISE

Table 4.1: Explanatory variables [53]

An important result in this paper is the differentiation between two contrast regimes: background noise-limited and speckle noise-limited. The boundary at which a target transitions into the background-limited regime depends on the total integration time and the magnitude of the target, resulting in variations among different targets (brighter targets have a larger separation before reaching the background limit).

In the speckle noise-limited regime, the performance is constrained by speckle noise originating from the residual PSF of the star. In contrast to the background noise limit, the speckle noise limit is anticipated to be influenced by a broader array of factors. In other words, the impact of explanatory variables on performance will be minimal in the case of the background noise-limited regime.

They used random forests to try and predict the contrast values using as input features the explanatory variables listed in Table 4.1. They found that the ADI models ($0.2''$ to $1.0''$) reach R^2 values between 69.9% and 82.3%, with RMSE values between 0.25 and 0.37 dex. It is worth pointing out that the researchers noticed a growing challenge in predicting ADI contrasts at larger separations. This result was attributed to the influence of the background limit. At a separation of 1 arcsecond, approximately 93% of ADI contrasts are constrained by the background limit. They hypothesized that background-limited contrasts are heavily impacted by the dynamic extended structures present in the thermal background, a characteristic that is not captured or measured by the explanatory variables used in their analysis. This suggests that the limitations in predicting contrasts at larger separations are related to factors beyond the scope of the chosen explanatory variables.

An example of the predictions achieved by their model can be found in figure 4.2.

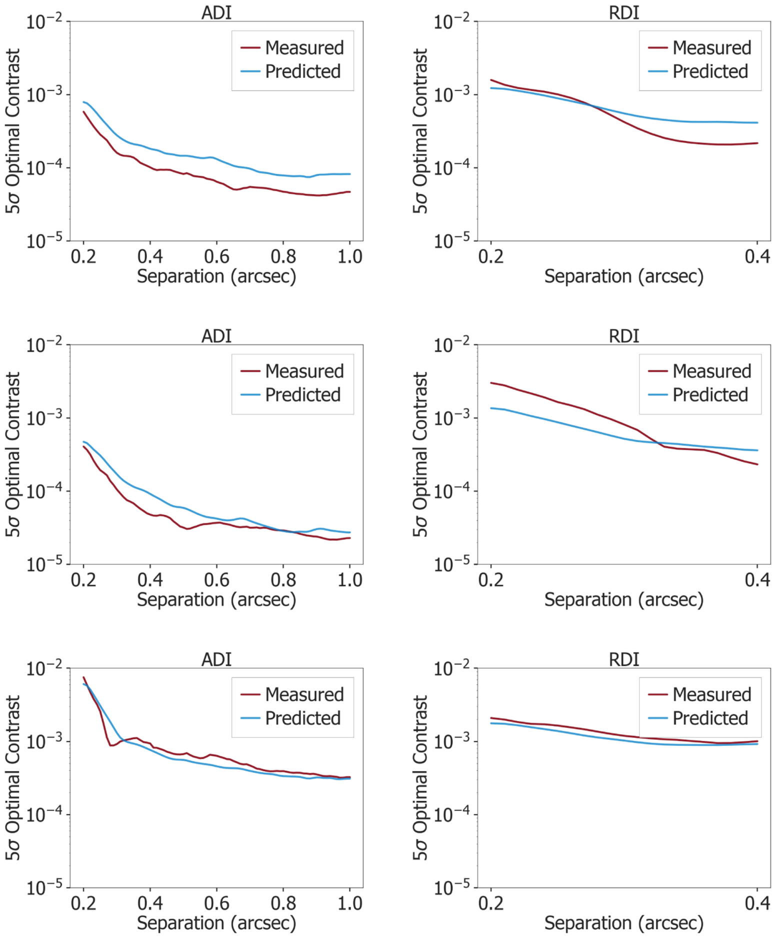


Figure 4.2: Predicted contrast curves and measured contrast curves for three targets [53]

Another interesting result can be found in Dahlqvist et al. 2022 [8]. The aim of his study is to detect and characterize potential exoplanets and brown dwarfs within debris disks. This is accomplished while taking into account a diverse population of stars, including variations in stellar age and spectral type. He presents the analysis of a set of H-band images captured by the VLT/SPHERE instrument as part of the SHARDDS survey. This survey gathers 55 main-sequence stars within 100 parsec, known to host a high-infrared-excess debris disk. This approach potentially offers a better understanding of the intricate interactions between substellar companions and disks.

In this paper, Dahlqvist also investigates the impact of observing conditions and the characteristics of the observing sequence on performance, which is measured in terms of contrast and a summary of his results is depicted in Figure 4.3.

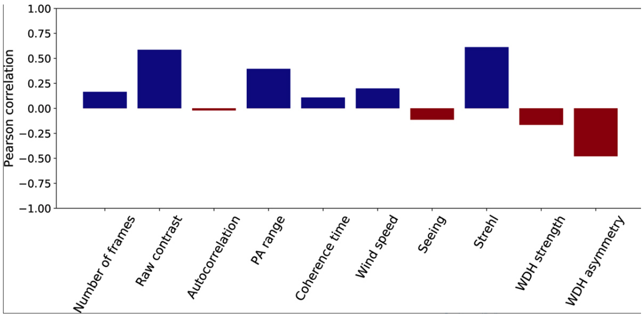


Figure 4.3: Pearson correlations between the median values of the contrast curves and the parameters characterizing the ADI sequences [8]

Finally, the primary distinction between the work of Xuan and this thesis lies in the choice of the instrument used for performance characterization. Indeed in Xuan's work, a different instrument was employed. In contrast, this thesis focuses on the Spectro-Polarimetric High-contrast Exoplanet REsearch instrument, which primarily operates within the speckle noise regime thus being more representative of what is expected to be mounted on the Extremely Large Telescope. The overarching goal is to develop more powerful predictive models.

Chapter 5

Dataset creation

The telescope takes a sequence of pictures during an observational session. As shown in the figure 2.12, these separate snapshots are then assembled into a consolidated master cube, which serves as a store for the raw astronomical images coming from the same observation.
Those data-cubes, accessible through the SPHERE Data Center [44], undergo a series of processing steps. This involves the correction of faulty pixels and the subtraction of the background noise. Afterward, the images are realigned with the center of the star, and the stellar speckle field is calculated using one of the post-processing methods detailed in the previous section before being subsequently removed. The data being processed through this reduction pipeline is consistently accessible through the SPHERE DC at each stage of the reduction process.

5.1 Contrast curves

In this section contrast curves, which will be used as objectives for the models, will be delved into, especially in terms of how they are obtained and the processing applied to the set of contrast curves used in this project.

5.1.1 Definition

A contrast curve is an essential element of characterizing exoplanets and detecting faint companions around a target star. To generate these curves, it is necessary to calculate the sensitivity to off-axis companions in terms of contrast at different angular separations. Contrast is defined as the ratio of the flux in specific regions of the observation to the flux from the central star. To obtain sensitivity limits at different angular separations, circular apertures at various radial distances from the central star are selected.

The Signal-to-Noise Ratio (SNR) can then be calculated as the ratio between the flux in the selected aperture and the estimated standard deviation of the flux from other apertures at the same angular separation, providing an estimate of the background noise.

However, an essential element is missing in this explanation, which is how the flux of an off-axis companion translates into the final image after post-processing. To address this, fake companions are injected into the raw data to determine what fraction of the initially injected flux appears in the final image. This allows for the calculation of the algorithm's throughput, denoted as T_r, which represents the signal attenuation as a function of angular separation. Throughput is empirically calculated using the formula $T_r = F_r/F_{in}$, where F_r is the recovered flux of a fake companion after post-processing, and F_{in} is the initially injected flux of the fake companion.

Thus, the contrast curve, denoted as C_r, is defined as follows:

$$C_r = \frac{k \times \sigma_r}{(T_r \times F_*)}$$

where k is a correction factor (usually set to five for obtaining the five-sigma contrast curve), and F_* represents the flux of the parent star. This definition takes into account the algorithm's throughput, which is crucial for accurate characterization of exoplanets and faint stellar companions [17].

The contrast curves are directly obtained from the SPHERE client. For this project, all of them originate from the SPHERE IRDIS instrument, using the H2H3 bands and the cADI reduction algorithm, an example of such a curve is depicted in figure 5.1.

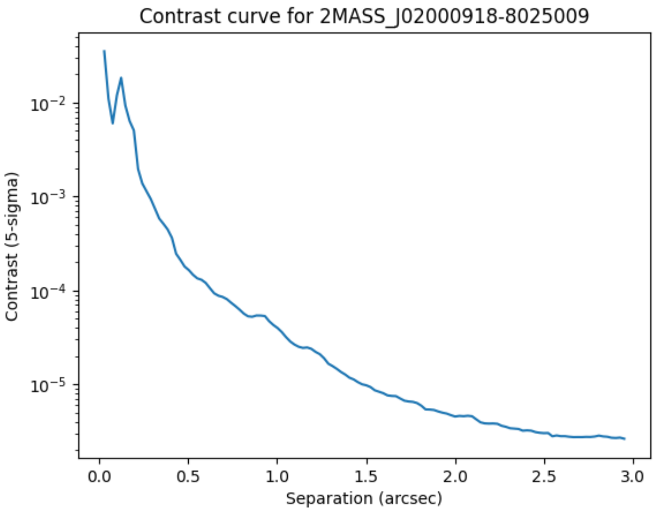


Figure 5.1: Contrast Curve

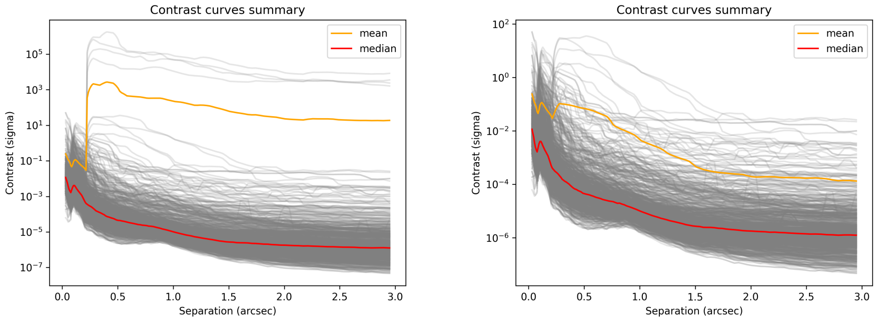
Each observation is associated with its own curve and these curves serve as the objective functions. Since many factors may affect the contrast achieved from an instrument on a given observing night, different curves can be obtained for a same target.

The contrast obtained can be divided into two regimes: background noise-limited and speckle noise-limited. The distance beyond which a target hits the background limit is determined by the total integration time and the magnitude of the target, and hence varies between targets (greater separation for brighter targets). The performance in the speckle noise-restricted domain is limited by speckle noise from the residual PSF of the star. The speckle noise limit is expected to be controlled by a wider range of factors than the background noise limit [53].

5.1.2 Processing

As mentioned in the previous subsection, the speckle noise limit is expected to be influenced by a broader range of factors compared to the background noise limit. Thus, to place greater emphasis on the speckle-noise limited regime, the maximum separation value has been uniformly set to 3 arcseconds for all the curves.

Furthermore, given that the precise type of ML model had not yet been determined at this stage, a consistent separation discretization has been applied to all the curves. To prevent extrapolation, the minimum separation has been selected as the maximum value among all the minimums observed in the curves. Additionally, the number of data points has been set to the median number of points observed in the curves and a logarithmic spacing has been used to prioritize small separation values. Finally, interpolating the curves with the new separation values results in the ones that will be used throughout the rest of this dissertation.



(a) With outliers

(b) Without outliers

Figure 5.2: Contrast curves summary with and without outliers

As depicted in Figure 5.2a, several curves show contrast values that significantly deviate from the typical range one would expect thus causing the mean of the contrast values to be way higher than expected. These curves pose a problem as they likely result from errors in the observations, therefore, they should be removed from the dataset.
To eliminate these problematic curves, the mean log-deviation from the median curve is computed and an histogram of those values can be found in Figure 5.3. After that, an arbitrary threshold above which the curves will be removed is selected (1 in this case).

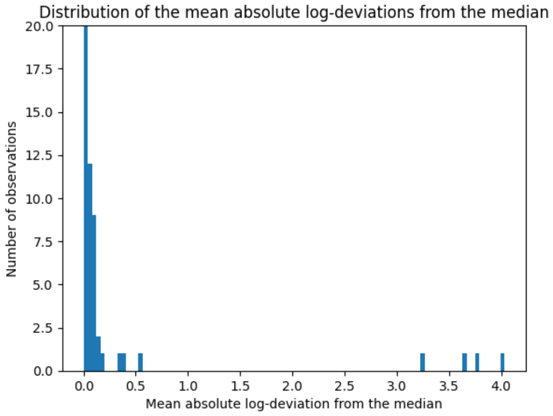


Figure 5.3: Deviations from the median curve

5.2 Features

The goal here is to gather a bunch of features that will help constrain the contrast and thus will serve as inputs to the ML model. In other words variables which might have an impact on the contrast value are sought.

In fact, many factors can influence the contrast achieved by an instrument on a given observing night. Some of these elements are environmental, such as atmospheric properties and optical temperature. Other considerations include the observed target and timing, which determine the airmass during the observation, the amount of time per target, and the degree of Parallactic Angle (PA) rotation. The detection limits are also influenced by the magnitude of the target star and the post-processing algorithm used [53].

5.2.1 FITS file

The Flexible Image Transport System (FITS) is a widely used open standard that defines a digital file format for storing, transmitting, and manipulating data. This format can store data in a variety of formats, including multi-dimensional arrays like a 2D image and tables. In the realm of astronomy, FITS stands as the most widespread digital file format. It was intentionally developed for astronomical data, giving capabilities like the description of photometric and spatial calibration data, as well as metadata that offers information about the provenance of the image [14].

The contrast curves discussed in the previous section are all provided in FITS files. In these files, a distinction can be made between the data, represented as a `numpy` record, and the headers, encoded as a dictionary. It is important to note that the headers can be customized according to the user's preferences when creating the file. Consequently, a significant amount of information is appended during the reduction pipeline from the SPHERE DC.

As previously mentioned, a great number of header keywords have been made accessible by the SPHERE DC. From this extensive set of keywords, a specific subset has been chosen for the purpose of creating the dataset. Table 5.1 provides a concise description of these selected keywords, as provided by the data center.

Because the reduction algorithms are periodically updated, some keywords are only accessible in specific versions of the data products. Consequently, there may be instances of missing data (as seen in table 5.1). While missing information about the effective number of frames or exposure used in the reduction can be handled by simply omitting that data, it is considerably more problematic when essential details such as the observation start and end dates are missing. Indeed, the start and end times of a telescope observation are crucial pieces of information. Without these timestamps it becomes impossible to obtain other vital variables, such as the seeing or the coherence time, during the course of the observation. These time-related details are essential for a comprehensive understanding and analysis of the data.

A substantial effort has been dedicated in order to resolve the issue of missing dates. Given that the contrast curve files are the outcome of a reduction pipeline, it suggests that a parent process utilizing the data cubes from the observation sequence might have access to the timestamps of these images.
However, it appears that the current implementation of the pipeline lacks a mechanism to directly link a specific contrast curve to its corresponding timestamp file using an identifier or a similar method. While the relationship between the processes involved is known, there is no clear way to establish the direct relationship between the outputs of those processes.

Header name	Description	Missing percentage (%)
ESO OBS ID	Observation ID	0.00
DATE-OBS	Average date of full observation	0.00
OBJECT	Original target	0.00
ESO TEL AIRM MEAN	Average airmass during the observation	78.04
EFF_NFRA	Effective number of frames used in reduction	76.74
EFF_ETIM	Effective exposure time used in reduction	76.74
SR_AVG	SPARTA average STREHL	39.20
ESO INS4 FILT3 NAME	Wavefront sensor spectral filter	0.00
ESO INS4 OPTI22 NAME	Wavefront sensor spatial filter	0.00
ESO AOS VISWFS MODE	SPARTA Visible wavefront sensor detection	0.00
ESO TEL AMBI WINDSP	Observatory ambient wind speed	0.00
SCFOVROT	Total field of view rotation	0.00
SC MODE	Reduction algorithm	0.00
ESO TEL AMBI RHUM	Observatory ambient relative humidity	0.00
HIERARCH ESO INS4 TEMP422 VAL	Temperature sensor on HODM case	1.53
HIERARCH ESO TEL TH M1 TEMP	M1 superficial temperature	0.00
HIERARCH ESO TEL AMBI TEMP	Observatory ambient temperature	0.00
OBS_STA	Starting date of observation	78.04
OBS_END	End date of observation	78.04
ESO DET NDIT	Number of Sub-Integrations	0.00
ESO DET SEQ1 DIT	Integration time	0.00

Table 5.1: Header keywords along with their description

In the absence of a direct identifier, a clever solution has been derived to link the contrast curves to the timestamp files. This approach utilizes a combination of three criteria: the target name, the night of observation, and a header keyword ('UTC' or 'LST') that is present throughout the entire reduction pipeline, with a value that is often unique. By employing these three verification criteria, the likelihood of linking files from different observations is greatly minimized thus providing a robust method for pairing contrast curves with their respective timestamp files.

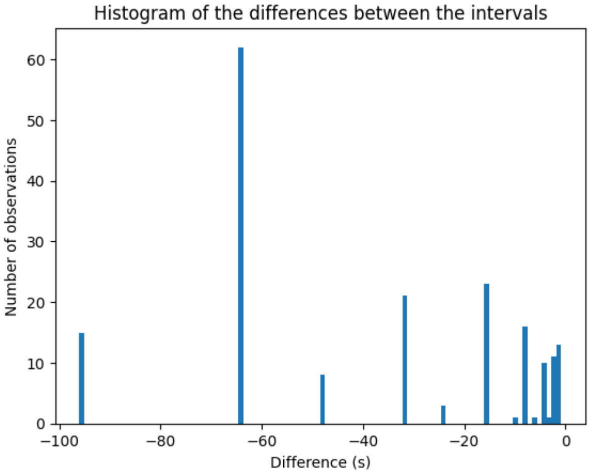


Figure 5.4: Histogram of the differences between the target and recovered time intervals

To assess the effectiveness of this method in estimating the starting and ending times, data from the 21.96% of observations where this information is available has been used. Figure 5.4 illustrates the difference between the two time intervals: the first one being the original data that needs to be recovered, while the second one is the recovered version. This analysis provides insights into the accuracy of the time interval recovery process.

The slight mismatch between the recovered version of the time of observation and the target time is attributed to the fact that not all the images within the data cubes from the parent process are used, as explained in the previous chapter. Some images are exclusively used for telescope calibration or are of poor quality, leading to their removal from the data cubes during the pipeline's pre-processing steps.
However, these estimated intervals are considered satisfactory as they only deviate from the original ones by a few minutes at most. Such a small time difference is unlikely to result in significant variations in factors like seeing or coherence time during the observation.

5.2.2 Simbad

The Set of Identifications, Measurements, and Bibliography for Astronomical Data (SIMBAD) is an astronomical database that catalogs objects beyond the Solar System, and it is managed by the Centre de Données astronomiques de Strasbourg (CDS). To retrieve the flux of the star in the G and H bands (which are respectively representative of the filters used for the SPHERE adaptive optics system and its IRDIS camera), a query is conducted using the time of observation, along with the name and coordinates of the target, as provided in the keywords. This query is executed using Julien Milli's GitHub repository as a resource [22].

The missing data percentages are 1.18% for the G band flux and 4.25% for the H band flux.

5.2.3 Paranal Astronomical Site Monitoring

The Paranal Astronomical Site Monitoring (ASM) is a database maintained by the ESO. Its primary objective is to monitor weather conditions during observations at the Paranal Observatory. To achieve this, sensors are strategically placed near the VLT. Notably, a significant update took place in April 2016. Prior to 2016, data on observation conditions such as seeing or coherence time were collected using the Differential Image Moption Monitor (DIMM). However, in 2016, the Multi-Aperture Scintillation Sensor (MASS) replaced the DIMM, offering more precise measurements of these observation conditions. Additionally, other devices like the SLOpe Detection And Ranging (SLODAR) and the Low Humidity And Temperature PROfiling microwave radiometer (LHATPRO) have also been installed, providing a broader range of observational data.

Using Julien Milli's GitHub repository, a query is conducted on the database to obtain the values of seeing and coherence time. It's important to note that these values can fluctuate during the observation sequence. As a result, statistical estimators such as the median and standard deviation will be employed to characterize these variable features. Additionally, the choice of whether to query DIMM or MASS depends on the observation date. Queries will be directed to the DIMM for observations made before April 2016 and to the MASS for those made after April 2016.

Another challenge arises from the inconsistent time-step used by the ASM to compute these values. This inconsistency makes it uncertain how many data points the database will return for a given time interval. Therefore, handling this variability in time step-size will be an important aspect of the data analysis.

To address the issue caused by the inconsistent time step-size in the ASM database, the following strategy is employed :

- Temporal Extension: An additional 15-minute period is appended to both ends of the time interval designated for the query. This temporal extension ensures that sufficient data is obtained, especially in cases where the observation duration is shorter than the irregular time-step in the database.
- Interpolation : The retrieved values are interpolated, and a uniform time step of 1 minute is applied for this process.

- Estimator Calculation: Within this extended time interval, the values of seeing and coherence time are collected. Since these values typically do not change significantly over a 15-minute interval, they can be used to calculate the median and standard deviation estimators for the interval.

In practical applications, such as predicting expected contrast values, the actual seeing during the observation is unknown. Therefore, only the seeing value observed just before the start of the observation is used as an estimate for the seeing conditions during the observation period. This approach accounts for the fact that the real-time seeing values are not available during the observation itself.

In conclusion, it's worth noting that a considerable number of missing values are encountered, particularly with regard to the estimators for seeing and coherence time. These estimators are absent in approximately 11.57% of the observations

5.2.4 Processing

Now that all the data is gathered, the subsequent phase involves preprocessing before feeding it into the machine learning models. Table 5.2 provides a summary of the input features.

First and foremost, the dataset is divided into three distinct subsets: the training set, the validation set, and the test set. The training set serves as the data used to train the machine learning model. The validation set plays a crucial role in assessing the model's performance, helping to identify whether it is overfitting or underfitting and determining when training should stop. Finally, the test set is reserved solely for evaluating the model's performance. It can be viewed as new data that the model has never encountered before, and it is used to assess how well the model generalizes to unseen examples.

Secondly, it's important to note that most machine learning algorithms, including neural networks, require input data in numerical form. However, real-world datasets often contain categorical features, such as colors, categories, or labels, which are not inherently numerical. To ensure that all data is in numerical form, a transformation is applied to convert categorical values into numerical equivalents.

Another crucial aspect of data preprocessing is dealing with missing values. Missing data can significantly impact the performance of machine learning models. To address this issue, the K-Nearest Neighbors (KNN) algorithm is employed.

Feature	Description	Type
ESO INS4 FILT3 NAME	Wavefront sensor spectral filter	Categorical
ESO INS4 OPTI22 NAME	Wavefront sensor spatial filter	Categorical
ESO AOS VISWFS MODE	Visible wavefront sensor detection	Categorical
ESO TEL AMBI WINDSP	Observatory ambient windspeed	Numerical
ESO TEL AMBI RHUM	Observattory ambient relative humidity	Numerical
HIERARCH ESO INS4 TEMP422 VAL	Deformable mirror temperature	Numerical
HIERARCH ESO TEL TH M1 TEMP	Primary mirror temperature	Numerical
HIERARCH ESO TEL AMBI TEMP	Ambient air temperature	Numerical
ESO DET NDIT	Number of sub-integrations	Numerical
ESO DET SEQ1 DIT	Integration time	Numerical
SIMBAD_FLUX_G	Host star magnitude (G band)	Numerical
SIMBAD_FLUX_H	Host star magnitude (H band)	Numerical
SEEING_MEDIAN	Median seeing during the observation sequence	Numerical
SEEING_STD	Standard deviation of the seeing during the observation sequence	Numerical
COHERENCE_TIME_MEDIAN	Median coherence time during the observation sequence	Numerical
COHERENCE_TIME_STD	Standard deviation of the coherence time during the observation sequence	Numerical
SCFOVROT	Amount of PA	Numerical
SEPARATION	Separation between the exoplanet and its host star	Numerical

Table 5.2: Input features

The KNN algorithm replaces missing values by computing the mean value of their k closest neighbors ($k = 5$ in the case of contrast predictions and $k = 3$ in the case of uncertainty modelling).

This approach is feature-wise, meaning that for each feature with missing values, the nearest neighbors for that specific feature is found and the mean value is calculated. By doing this, the imputed values are more representative of the data distribution and are not simply replaced with arbitrary values.

Furthermore, in the context of neural network applications, such as deep learning, there are additional considerations for data preprocessing. One crucial concern is feature normalization. Normalization is recommended to avoid potential issues such as vanishing or exploding gradients during the training of deep neural networks.

To achieve this, both imputation and normalization parameters are calculated based on the training set. Subsequently, these parameters are utilized to both impute missing values and normalize the features in all three data subsets. This separation is crucial for maintaining the integrity of the model evaluation process. Indeed, it is imperative to follow this approach to prevent the model from being influenced by sets that are exclusively meant for testing purposes.

In summary, proper data preprocessing, including the transformation of categorical values into numerical form, addressing missing values with the KNN algorithm, and performing feature normalization, is essential for preparing the dataset for machine learning or neural network applications. These steps not only improve model performance but also ensure that the model is trained and evaluated with rigor and accuracy.

Chapter 6

Methodology

The goal of this chapter is to present the methodology used for building, training, and selecting the models. It is important to note that, in order to achieve this, a bottom-up approach has been taken. Indeed, in machine learning, there are instances where everything appears to be running smoothly without displaying any errors, but in reality, there might be issues in the code that prevent the model from learning, for example.

To address these types of problems, it is advisable to begin with a small-scale approach. For instance, to ensure that the model is indeed learning, one can start by deliberately overfitting it to a small training batch. Afterward, gradually advance the complexity of the steps while diligently monitoring for smooth operation and proper learning by the model.

Secondly, a building pipeline has been defined, and all the models implemented in this work will adhere to the same pipeline. The pipeline consists of the following steps:

1. Data preparation
2. Model instantiation
3. Model training and validation
4. Model testing

6.1 Two different types of datasets

If the dataset used as input for the models is denoted as $\mathbf{X}$ and has dimensions $(m \times n)$, then m represents the number of observations or, in this context, the number of telescope observations ($m = 843$).

Each observation results in one contrast curve, so there are m contrast curves. On the other hand, n stands for the number of features, where the separation is excluded ($n = 17$). In other words, it's the number of characteristics or variables used to describe each contrast curve, but it does not include the feature related to separation.

As mentioned in the chapter discussing the creation of the dataset, the objective is to perform one regression per observation, or in other words, predict the expected detection limit in terms of contrast (contrast curve) between an exoplanet and its host star. Consequently, all the contrast curves share the same separation vectors or x vectors. Therefore, the resulting contrast vectors or y values have the same dimension, which is $N = 124$ in this specific context. A representation of this dataset can be found in Table 6.1.

Observation ID	$\mathbf{X}$	$\mathbf{y}$
1	x_1^1 ... x_n^1	$[y_1^1, y_2^1, ..., y_N^1]$
2	x_1^2 ... x_n^2	$[y_1^2, y_2^2, ..., y_N^2]$
...	...	...
m	x_1^m ... x_n^m	$[y_1^m, y_2^m, ..., y_N^m]$

Table 6.1: Vector contrast prediction

In the specific context of Table 6.1, the exclusion of separation vectors from the dataset serves two primary purposes. The first reason is that these separation vectors do not vary between observations, meaning they do not provide discriminatory information between different observations. Additionally, since the models output one contrast vector per observation, adding the separation vectors to the dataset **X** would require adding one separation vector per row. This is often not feasible with libraries like `pytorch`.

One potential solution would be to incorporate the separation values as features, making $s_1, \ldots, s_N$ new features in the dataset **X**. However, since N is significantly larger than n, including these separation values as features could potentially confuse the model without providing substantial additional information. Therefore, in this context, the decision was made to exclude separation values from this first type of dataset.

The main drawback of this data representation is that the models are trained to predict contrast values based on the same fixed discretization of separation values. Consequently, it becomes challenging to directly query the model for the contrast value at a specific separation s_i that falls outside of the usual discrete separation steps.
One possible solution to address this issue could involve predicting the entire contrast vector and then simply use linear interpolation to estimate the contrast value for the desired separation s_i.

This approach, while requiring some computation, would provide a reasonable estimation of the contrast value at arbitrary separation points and could be a practical workaround for the limitations of the fixed discretization during model inference.

In that regard, another representation of the dataset will be derived. In this new representation of **X**, the objective is to predict a single contrast value for each data point, so the separation values will be included in the input dataset **X**. For a given observation, the input features (excluding separation) will be replicated N times. A visual representation of this updated dataset is provided in Table 6.2.

Observation ID	**X**				**y**
1	x_1^1	...	x_n^1	s_1	y_1^1
1	x_1^1	...	x_n^1	s_2	y_2^1
1	x_1^1	...	x_n^1	...	...
1	x_1^1	...	x_n^1	s_N	y_N^1
2	x_1^2	...	x_n^2	s_1	y_1^2
2	x_1^2	...	x_n^2	s_2	y_2^2
2	x_1^2	...	x_n^2	...	...
2	x_1^2	...	x_n^2	s_N	y_N^2
...		...		...	...
m	x_1^m	...	x_n^m	s_1	y_1^m
m	x_1^m	...	x_n^m	s_2	y_2^m
m	x_1^m	...	x_n^m	...	...
m	x_1^m	...	x_n^m	s_N	y_N^m

Table 6.2: Single value contrast prediction

Finally, in the context of implementing multiple models, maintaining consistency in the division of data into training, validation, and testing sets is absolutely essential. This consistency is crucial for meaningful model comparisons, ensuring that all models are evaluated on the same testing set and trained on the same training set.

In this specific context, the data must be divided by observations, which gives rise to two important rules. Firstly, a given observation denoted as i must belong to one and only one subset (training, validation, or testing) for all models. This ensures that each observation is consistently used for training, validation, and testing across different models.
Secondly, in cases where the data is represented as shown in Table 6.2, the entire observation, including all data points originating from the same telescope-observation, must remain together in the same subset. Data points from the same telescope-observation should not be split across different subsets. This rule preserves the integrity of the observations during the division process.

Adhering to these rules will enable fair and accurate comparisons between different models while maintaining the integrity and consistency of the dataset.

6.2 Models

The initial model represents a basic implementation of a random forest, where its primary objective is to predict a single contrast value. It utilizes a feature vector as input, which includes the separation value, and uses standard practices for random forest modeling, including the number of trees and the maximal number of feature used.

The second model, which takes the form of a MLP, shares the same input features as the first model and generates similar types of outputs. Parameters like the number of hidden layers and units per layer are configurable and will be delved into in the following chapter. The activation function used between the hidden layers is the Rectified Linear Unit (ReLU).

The third model is again a MLP which adopts a data representation as depicted in Table 6.1, and otherwise closely resembles the second model.

Lastly, the final MLP model, while sharing input features with both the random forest and the initial neural network (as shown in Table 6.2), embarks on a more advanced task. Instead of predicting the contrast value, it focuses on estimating the mean and logarithmic standard deviation of a distribution to capture uncertainty. The loss function is derived straightforwardly from performing maximum-likelihood estimation on the probability distribution function of the chosen distribution type.

6.3 Training

Regarding the random forest model, training is accomplished simply by utilizing the implemented `fit()` function from the `scikit-learn` library [41].

For the neural networks, the training pipeline is consistent across all models and follows a standard procedure. Firstly, a maximum number of epochs is established. Next, the observations are shuffled at the beginning of each new epoch. In the case of the vector output model, shuffling is done without specifying any particular sequence. However, for the single output models, there is an option to specify the size of the sequence or the number of data points by which the shuffle should be performed.

After completing the shuffling step, the dataset is partitioned into small batches of data points. It is crucial to ensure that the total number of data points is evenly divisible by the batch size. This configuration ensures that during each epoch, every data point is processed exactly once, and the impact of each batch on updating the model parameters remains consistent across all batches.

A single forward-pass is performed for a batch, followed by the evaluation of a criterion. The criterion corresponds to the mean squared error when predicting contrast or the expression derived in equation 3.3 when calculating uncertainty is desired. Subsequently, a backward-pass is executed for that batch, leading to an update of the model's parameters. This entire process is repeated for all batches, and when all batches have been processed, one epoch is completed. The loss for a single epoch is essentially the average of the losses calculated across all the batches.

Ultimately, two types of schedulers have been incorporated for managing the learning rate. A scheduler's primary function is to gradually reduce the learning rate after a certain number of epochs. The underlying idea is that initially employing a higher learning rate accelerates the model's convergence, but it must subsequently be reduced to facilitate convergence towards a local minimum, preventing excessive leaps within the loss landscape.
The first scheduler is a continuous and gradual one, defined by the following expression:

$$lr_t = \frac{1}{1 + \text{decay rate} \times \text{epoch}_t} \times lr_0 \tag{6.1}$$

The second scheduler is a step-based approach, characterized by the following expression:

$$lr_t = lr_0 \times \text{decay rate}^{\text{epoch}_t // \text{step size}} \tag{6.2}$$

Here, // represents integer division, ensuring that the learning rate is adjusted at specific intervals defined by the step size.

6.4 Validation

In the context of neural networks, the validation set is employed once at the end of each epoch to gauge the model's performance on examples it hasn't been exposed to during training. This usage of the validation set helps in monitoring whether the model is suffering from underfitting or overfitting.

In this neural network context, a stopping criterion is established. When the validation score reaches a new minimum, the model that achieved this minimum is saved. Subsequently, if the validation score starts to rise, it may indicate that the model is becoming too specialized for the training data, potentially leading to overfitting. Therefore, a threshold on the number of epochs is set, and if the model fails to improve its validation score within this predefined number of epochs, the training process is stopped. This strategy ensures that the model generalizes well and prevents it from memorizing the training data excessively.

The second purpose of the validation set is to fine-tune the models, which includes both random forests and neural networks. Various hyperparameter values will be tested for each model, and the set of hyperparameters retained will be the one that enables the model to attain the lowest validation loss.

6.5 Hyper-tuning

In the case of the random forest, the only hyperparameter that was tested was the number of features used to split a node. The value that yielded the best validation loss was selected and retained.

On the other hand, the training and validation of the neural network models were executed on the Montefiore Alan Clusters [33], and the progress was monitored using Weights and Biases [50]. To streamline the process of altering key parameters for various runs while ensuring clear differentiation between them, a systematic approach was devised.

A powerful feature available on the Weights and Biases website is known as "sweep." This feature enables users to specify multiple hyperparameter values, either as vectors of hyperparameters or as distributions. These specified parameters are then aggregated into a nested dictionary, which serves as a configuration object. Subsequently, multiple models can be trained using various combinations of these hyperparameter values. The primary objective, typically the minimization of the validation loss, is tracked for each specific set of parameter values. This functionality empowers users to fine-tune their models effectively by identifying the optimal hyperparameter settings.
To accomplish this, a method for selecting hyperparameter combinations had to be chosen among the three available methods illustrated in Figure 6.1.

The first is called grid search, which systematically iterates over every possible combination of hyperparameter values, making it computationally expensive. The second method is called random search, which randomly selects hyperparameter values on each iteration based on provided distributions. Lastly, the third method, known as Bayesian search, builds a probabilistic model of a metric score as a function of hyperparameters and selects parameters with a high probability of improving the metric. Bayesian hyperparameter search employs a Gaussian Process to model the relationship between parameters and the model metric, optimizing the probability of improvement. However, this approach requires the specification of the metrickey and works well for a small number of continuous parameters but does not scale well for larger parameter spaces.

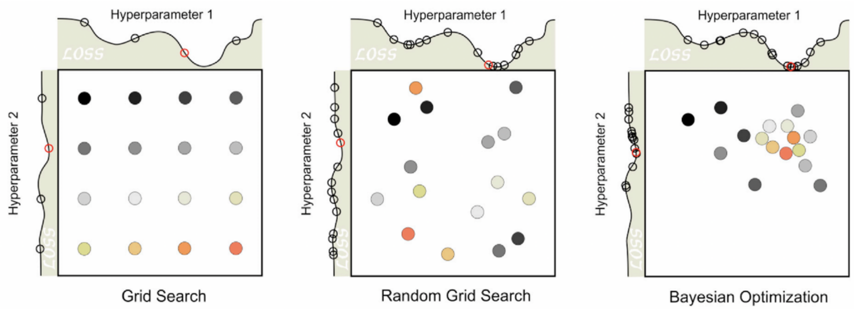


Figure 6.1: Three different methods for hyper-parameters search available on Weights and Biases [46]

The chosen approach involves initially using random search to explore the hyperparameter values and the corresponding performance trends. Once a rough understanding of these values and their impact is obtained, the next step focuses on fine-tuning the neural network. This involves narrowing down the hyperparameter values to the ones that yielded the best results in the random search. If the number of potential combinations is manageable, grid search can be an option. However, in cases where the number of different parameters is not excessive, and the distributions of these parameters are predominantly continuous, the Bayesian approach can be a suitable choice. On the other hand, when the hyperparameters are numerous and don't have continuous distributions, constraining the values and then conducting a combination of random search and grid search can yield effective results.

A helpful feature within the sweep window is the Parameters Importance and Correlation Assessment tool [45]. Correlation measures the linear relationship between a hyperparameter and the selected metric. In other words, a high correlation implies that when the hyperparameter has a higher value, the metric also tends to have higher values, and vice versa. Correlation is a valuable metric to examine, but it may not capture second-order interactions between inputs, and it can be challenging to compare inputs with vastly different ranges.

To address these limitations, an importance metric is also calculated. This involves training a random forest with hyperparameters as inputs and the metric as the target output. Subsequently, feature importance values for the random forest are reported.

This tool was used to conduct another round of random search, considering the correlation between hyperparameters and cross-validation loss, with the hope of finding a slightly better-tuned model.

For those who require more advanced fine-tuning, it's possible to repeat this strategy until a well-constrained range for hyperparameters is found and then perform Bayesian or grid search. However, this approach was not pursued here, as it would entail a significant computational load for potentially marginal gains, and the already obtained results were quite satisfactory.

Finally, it is worth mentioning that when saving a trained neural network, it is the state dictionary that is stored. To successfully load this state dictionary into a model, the model must be instantiated with the correct architecture, including the appropriate number of hidden layers and units per layer. To achieve this, a consistent naming convention, incorporating information from the configuration object, was devised for saving the models.

6.6 Testing

The exclusive purpose of the testing set is to offer a thorough evaluation of the model's performance, as illustrated in Figure 3.2.

The models generate predictions for the test telescope observations, and for each observation, two metrics are computed: the mean absolute error and the mean squared error. To obtain an overall assessment for the entire dataset, all the MSE and MAE values are averaged to calculate their mean, and the median value is also retained.

Chapter 7

Results

In this chapter, the results obtained at the conclusion of this work will be presented and discussed. It's essential to recall that there are a total of 843 observations, or contrast curves, and within each contrast curve, there are 124 data points. These points represent the separation (x) and the detection limit (y) between an exoplanet and its host star in terms of contrast. These observations have been divided into training, validation, and testing sets, with sizes of 672, 85, and 86, respectively.

7.1 Model Selection

First and foremost, to achieve the best possible results, one must select the models that appear to perform optimally on the validation set, as discussed in the previous chapter on methodology.

7.1.1 Contrast Prediction

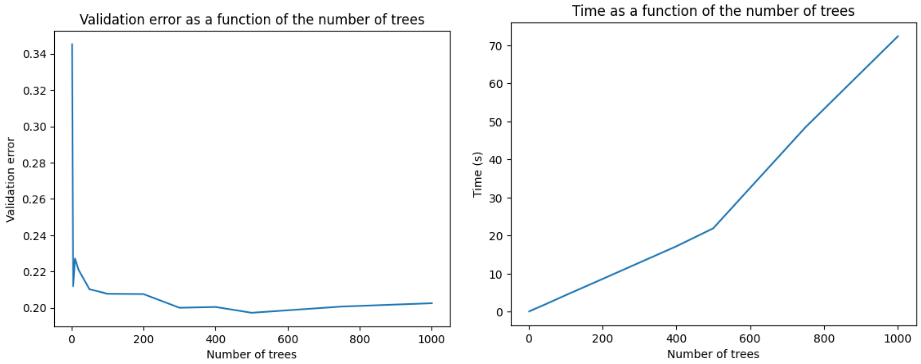
In this section, both the random forest and neural network models, which predict the contrast, will be trained and fine-tuned. The models that achieve the best validation loss will be retained.

Random Forests

In the case of random forests, having more estimators or trees generally leads to better performance. However, this improvement comes at the expense of increased computational resources as depicted[1] in Figure 7.1.

[1] Note that at the end of the validation, the trained random forest is saved as a pth file which can take quite some time as the number of estimators increases.

Therefore, it is essential to strike a good balance between precision and the computational load required.



(a) Validation MSE

(b) Training and validation time

Figure 7.1: Validation loss and execution time as a function of the number of trees in the forest

Once the number of trees is established (in this case, $n = 500$), the next step is to determine the maximum number of features used to split a node. The errors achieved by different models (one for each maximum features value) on the validation set are reported in Table 7.1.

Max. features	0.1	0.3	0.5	0.7	0.9	1
Validation MSE	0.1972	0.2067	0.2076	0.2114	0.2402	0.2839

Table 7.1: MSE on the validation set as a function of the maximal amount of features used to split a node

In this case, it is evident that the clear winner is the random forest model that utilizes 10 percent of the total number of features.

MLP Predicting a single contrast value

Figure 7.2 illustrates the trends in the hyper-parameters of the model, which produces a single contrast value.

The batch size must be a multiple of the separation size; for instance, 124 corresponds to a single observation in the batch, and so on.

The decay rate and step-size parameters correspond to the values from Equation 6.2, which is the step-sized approach to schedule the learning rate. The learning rate value corresponds to lr_0 in the same equation.

The shuffle sequence parameter determines the size of the sequence by which the data points will be shuffled at the start of each epoch before they are processed in batches.

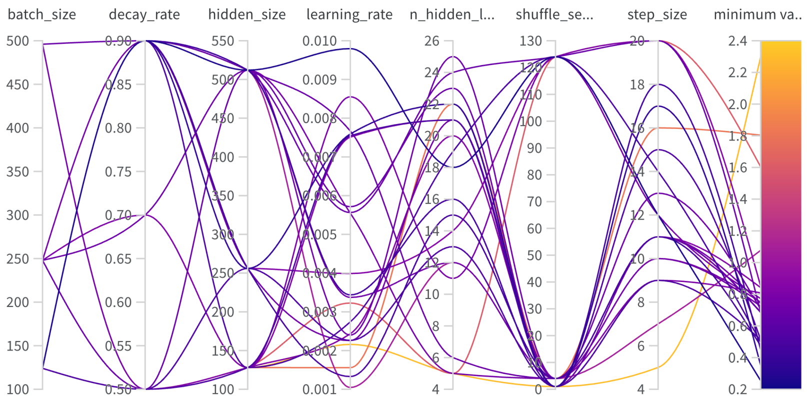
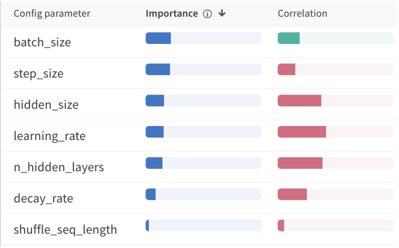


Figure 7.2: Hyper-parameters trends for predicting a single contrast value

Figure 7.3 displays the importance of parameters and their correlation with the validation loss. In this visualization, a red color indicates a negative correlation, meaning that an increase in the value of the hyperparameter corresponds to a decrease in the objective (validation loss). Conversely, a green color signifies the opposite relationship.



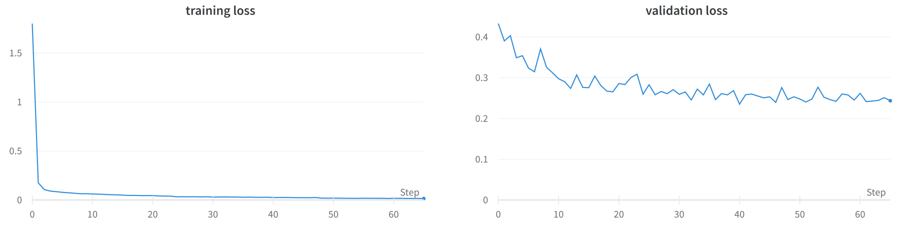
Config parameter	Importance ⓘ ↓	Correlation
batch_size	🟦	🟩
step_size	🟦	🟥
hidden_size	🟦	🟥
learning_rate	🟦	🟥
n_hidden_layers	🟦	🟥
decay_rate	🟦	🟥
shuffle_seq_length	🟦	🟥

Figure 7.3: Parameters importance and correlation with the validation loss

With the information at hand, the parameter ranges are updated accordingly, and another random search is conducted to discover a slightly improved model.

The best-performing model is a Neural Network with 28 hidden layers, each containing 512 hidden units. The batches correspond to a single observation, meaning 124 data points. The initial learning rate value is relatively high at 0.0106, and the corresponding scheduler involves reducing the learning rate by a factor of 0.7556 every 24 epochs. Additionally, the sequences are shuffled with a size of 1, indicating that they are shuffled per data-points.

The reduction in losses during the training of this best-performing model is visualized in Figure 7.4. Training is halted when there has been no improvement in the validation loss for 25 epochs. The model retained is the one that achieved the best validation loss, which is 0.2352.



(a) Training Loss

(b) Validation Loss

Figure 7.4: Training and Validation Losses (MSE) of the best performing model (single)

MLP Predicting the whole contrast vector

Figure 7.5 depicts the trends in hyper-parameter values. These parameters are the same as those found in Figure 7.2, with the only difference being that here, a single data point in the batch corresponds to a single observation since the objective of the network is to predict the entire vector of contrast directly.

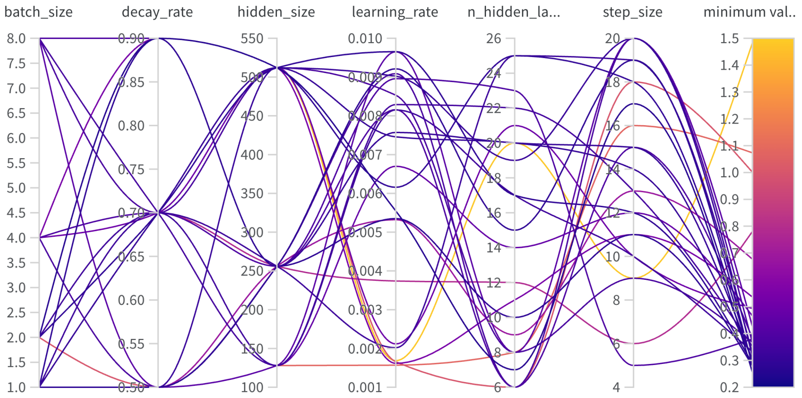
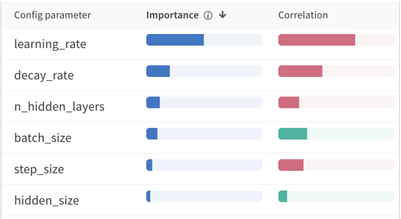


Figure 7.5: Hyper-parameters trends for predicting the whole contrast vector

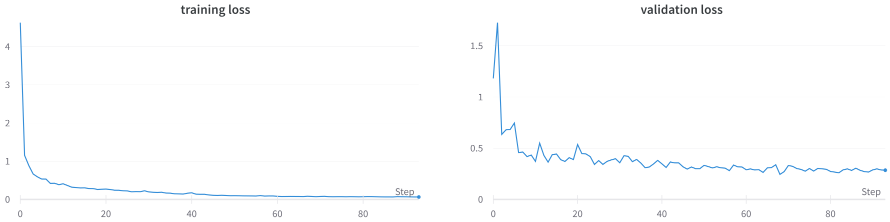
Once more, the information regarding the importance of each parameter and its correlation with the validation loss, as shown in Figure 7.6, is leveraged to conduct another round of parameter random search. This aims to refine and fine-tune the model.

The best-performing model is a Neural Network with 11 hidden layers, each containing 256 hidden units. Two observations are included in each batch. The initial learning rate value is relatively high at 0.0131, and the corresponding scheduler involves reducing the learning rate by a factor of 0.8804 every 12 epochs.



Config parameter	Importance ⓘ ↓	Correlation
learning_rate	🟦	🟥
decay_rate	🟦	🟥
n_hidden_layers	🟦	🟥
batch_size	🟦	🟩
step_size	🟦	🟥
hidden_size	🟦	🟩

Figure 7.6: Parameters importance and correlation with the validation loss



(a) Training Loss

(b) Validation Loss

Figure 7.7: Training and Validation Losses (MSE) of the best performing model (vector)

The reduction in losses during the training of the best-performing model is displayed in Figure 7.7. The model retained is the one that achieved the best validation loss, which is 0.2445.

7.1.2 Capturing Uncertainty

The approach used here is identical to the one employed for tuning the neural networks that predict the contrast. The primary difference lies in the loss function used to train and validate the model, which is not Mean Squared Error but instead the expression found in Equation 3.3.

This expression of the loss function is based on the assumption that $p(y|\mathbf{x})$ follows a normal distribution. While this is a strong assumption that can be challenging to verify, it's important to note that the distributions shown in Figure 7.13 should not be confused with $p(y|\mathbf{x})$. In this context, only the separation feature is fixed, so the histograms in the figure can be considered as some form of marginal distributions of the contrast given a separation value. From a code perspective, changing the assumption about the distribution would not require extensive modifications, making the normal assumption a reasonable starting point for considering uncertainty.

Figure 7.8 presents the trends in the hyperparameters during the second random search[2] and Figure 7.9 shows the parameters influence on the validation score.

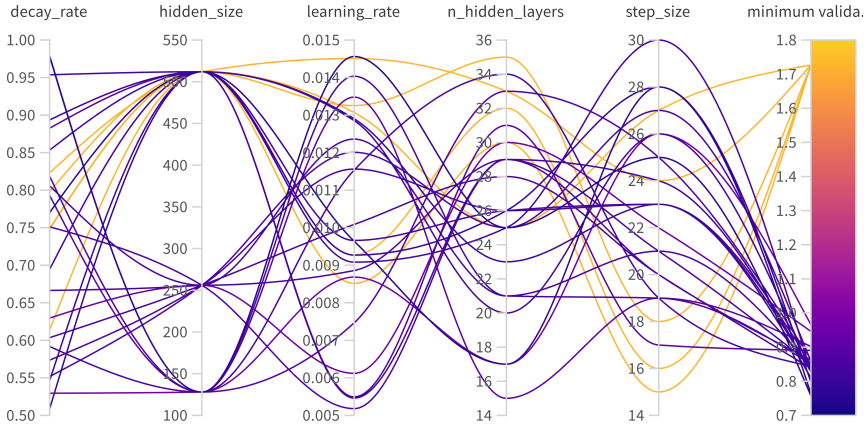
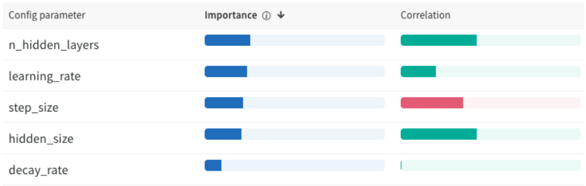


Figure 7.8: Hyper-parameters trends for capturing the uncertainty



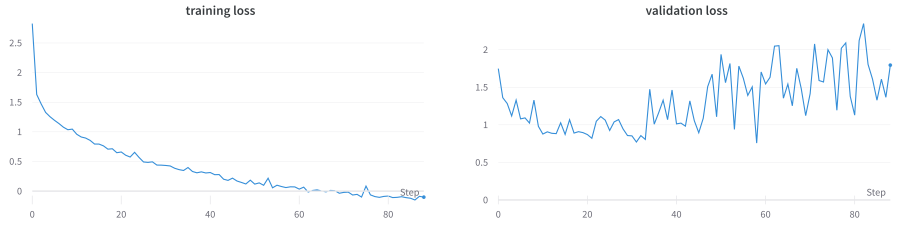
Config parameter	Importance ⓘ ↓	Correlation
n_hidden_layers		
learning_rate		
step_size		
hidden_size		
decay_rate		

Figure 7.9: Parameters importance and correlation with the validation loss

The best overall model is a neural network with 25 hidden layers, each containing 128 hidden units. The initial learning rate is set to 0.0146, and it is multiplied by the decay rate of 0.979 every 25 epochs. Both the batch size and shuffle sequence size are set to 124 data points, which corresponds to one observation. The training and validation losses are displayed in Figure 7.10, and the best validation score (at which training is stopped) is 0.757[3].

[2] Note that the first random search contained some failed runs due to a KNN fitting error, hence the display of the results from the second random search.

[3] Please note that this result is not directly comparable to the ones found in the contrast prediction section, as the expressions of the losses used in the two sections differ.



(a) Training Loss

(b) Validation Loss

Figure 7.10: Training and Validation Losses of the best performing model (uncertainty)

7.2 Results and Discussion

After selecting the optimal models of each type, it is time to present the results they can achieve on the testing set.

7.2.1 Contrast Prediction

First, the results achieved by the different models can be found in Table 7.2.

Model	MSE	MAE
Random Forest (single)	0.1848	0.2540
Neural Network (single)	0.2632	0.3040
Neural Network (vector)	0.2446	0.3264

Table 7.2: Results of the models predicting the contrast on the test set

As anticipated from the validation scores, the random forest outperforms both neural networks on the test set. Interestingly, the neural network that predicts a single contrast value has a higher Mean Squared Error loss value on the test set, suggesting it may be less precise than the neural network that outputs a vector. However, when examining the Mean Absolute Error, it becomes evident that the single-output neural network performs better on average than the one outputting a vector. This is because, on average, the predictions from the single-output network are closer to the actual values. The higher Mean Squared Error results from the penalization of significant errors.

It seems that, given the 18 input features, restricting the random forest to split nodes using only a subset of the features (one feature in this case) is a robust approach for achieving good results. As observed in Table 7.1, when all features are used, the random forest model performs worse than the neural networks. If feature selection had been carried out in the context of the neural networks, it's possible that the results of those models would have been better.

Additionally, as indicated in Equation 3.2, the use of Mean Squared Error as the loss function for regression assumes that $p(y|\mathbf{x})$ follows a normal distribution. However, if this assumption does not hold, a more appropriate expression of the loss could lead to better results.

Lastly, it's important to note that the models were trained and selected on CPUs due to the limitations of the computing node[4]. This significantly slowed down the process, particularly in hyper-tuning the models, which is computationally intensive.

Feature Importance

Random forests offer a practical capability to assess the importance of features in relation to their impact on the model's output. When each node is split using all the features to prevent bias, the feature importance ranking is visualized in Figure 7.11. This ranking provides valuable insights into how different input variables influence the model's predictions.

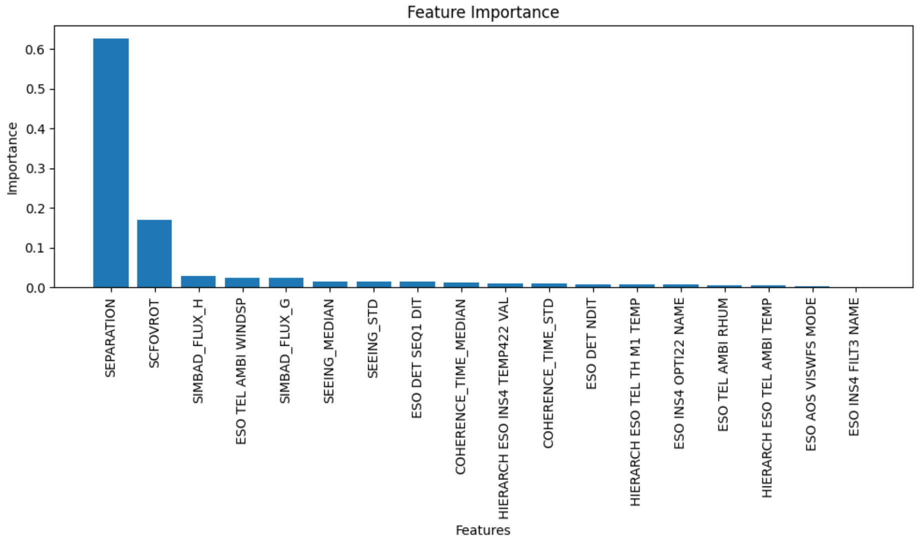
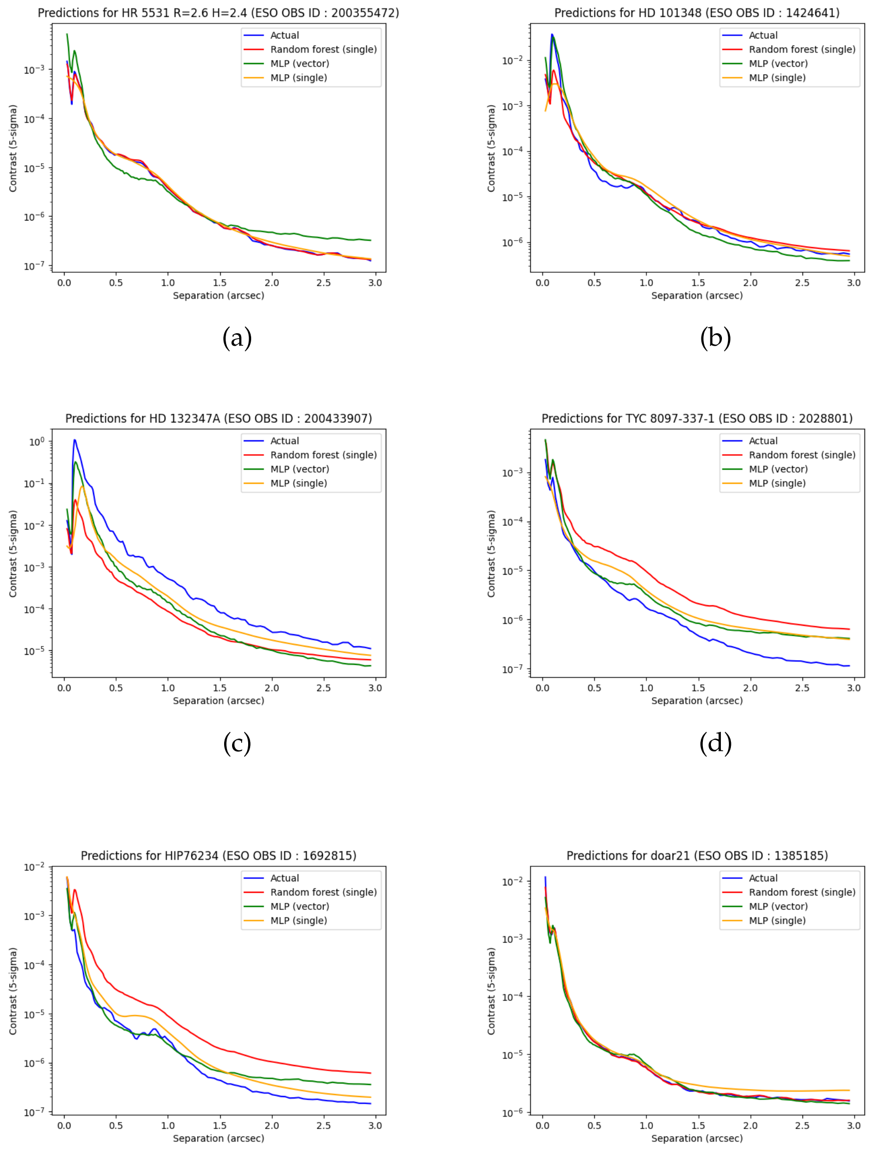


Figure 7.11: Feature importance

Predictions examples

Figure 7.12 displays random examples of predictions made by the predictive models.

[4] For reasons that were not identified, GPUs were unavailable on the cluster node that was utilized.



(a)

(b)

(c)

(d)

(e)

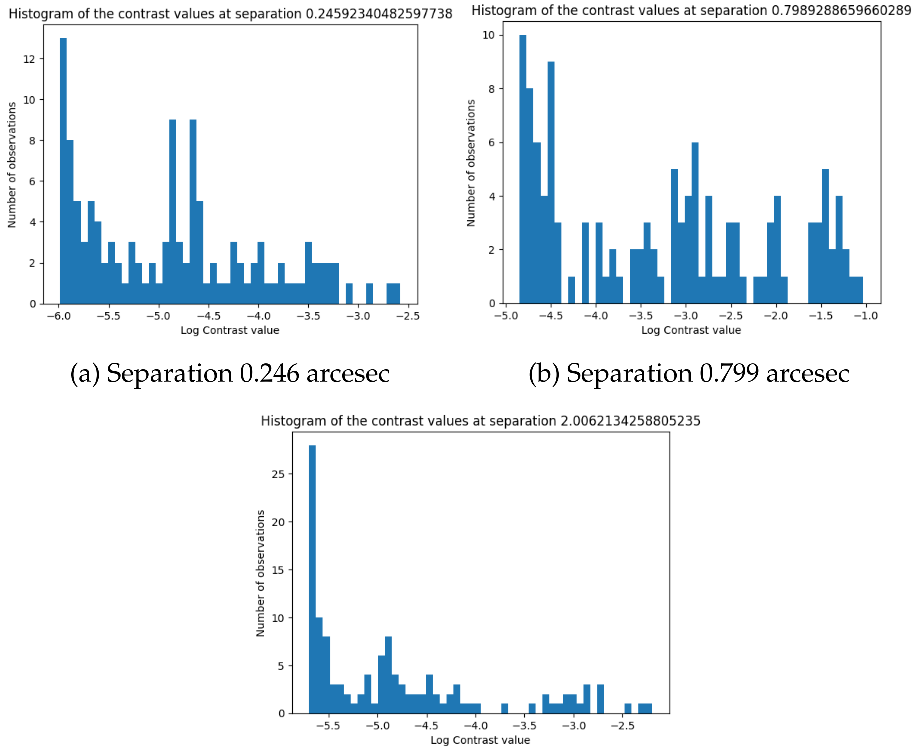
(f)

Figure 7.12: Predictions of the different models (test set)

In some observations, the predicted contrast curves closely resemble the actual ones. This can be attributed to the fact that, in some instances, the same target star appears in both the training and testing datasets. It's important to clarify that this is intentional. The aim is to gain a deeper understanding of the impact of various input features on the contrast value, such as atmospheric conditions and observing strategies. Therefore, in this context, the same target may be observed at different times and under different observing conditions, rendering the observations independent even for the same targets. Nevertheless, in some cases, the conditions are so similar that distinguishing between two contrast curves can be challenging.

Predictions at given separation values

In this section, three separation values of interest will be considered: the first one is 0.25 arcsec, corresponding to the first typical spike in the contrast curves. The second separation of interest is 0.8 arcsec, which is sometimes associated with another small bump in the curve. Finally, the third separation value is 2 arcsec, which usually lies close to the background noise-limited regime of the curve.



(a) Separation 0.246 arcesec

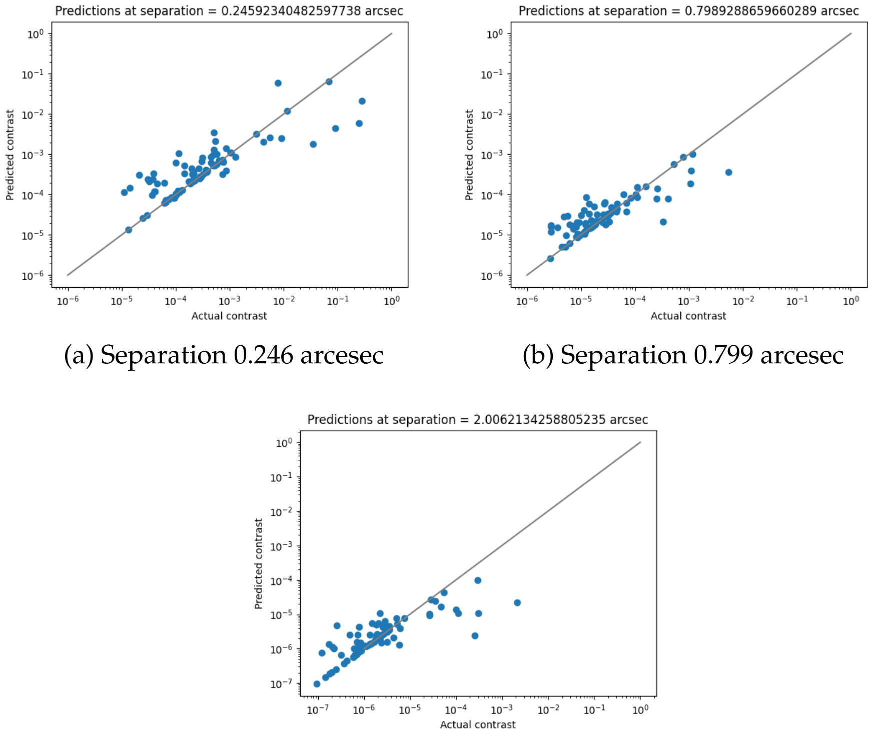
(b) Separation 0.799 arcesec

(c) Separation 2.006 arcesec

Figure 7.13: Histograms of the (log) contrast values at different separations

In the context of these analyses, Figure 7.13 illustrates the marginal distributions of the contrast at those specific separation values. Furthermore, Figures 7.14, 7.15, and 7.16 depict the predicted contrast values against the actual contrast values at those separations for the random forest, the neural network (single), and the neural network (vector), respectively.

A point is positioned above the identity line when the predicted value exceeds the actual one, and below the line when it's the opposite. Points in the lower left corner represent small contrast values, while those in the upper-right corner denote high contrast values.



(a) Separation 0.246 arcesec

(b) Separation 0.799 arcesec

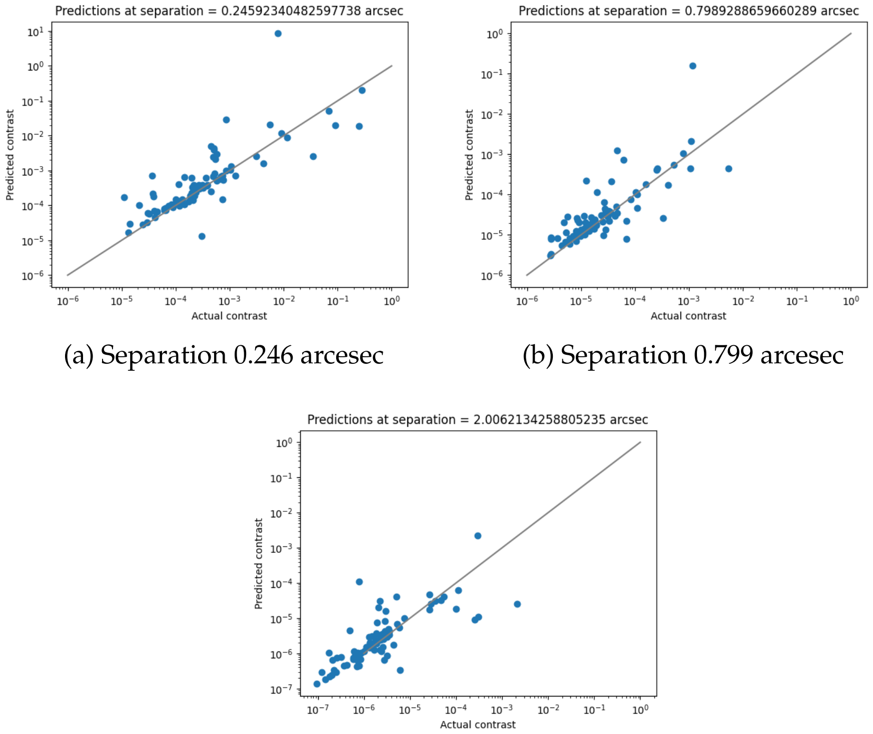
(c) Separation 2.006 arcesec

Figure 7.14: Contrast actual vs predicted values (Random Forest)

Regarding the random forest (Figure 7.14), for typical contrast values at a separation of 0.25 arcsec, the predictions often exceed the actual values, except when the actual values are unusually high, resulting in a spike. At a separation of 0.8 arcsec, the relationship between actual and predicted contrast values is more linear, which is desirable, but the random forest's predictions still tend to be higher than the actual values on average. At a separation of 2 arcsec, the same behavior as at 0.8 arcsec is observed, with the only difference being that the typical contrast values at this separation are smaller.

In general, one limitation of the model is its inability to effectively identify outliers. When the actual contrast values are unusually high for a given prediction, the model often fails to predict a high enough contrast value.

Concerning the neural network that predicts only a single contrast value (Figure 7.15), the points are generally closer to the identity line in all three separation cases. This suggests that this model is less biased at predicting smaller or larger contrast values than the actual ones. However, the points appear to be somewhat more dispersed compared to the random forest, indicating that the predictions of this model are, on average, less precise. This observation aligns with the loss values presented in Table 7.2.



(a) Separation 0.246 arcesec

(b) Separation 0.799 arcesec

(c) Separation 2.006 arcesec

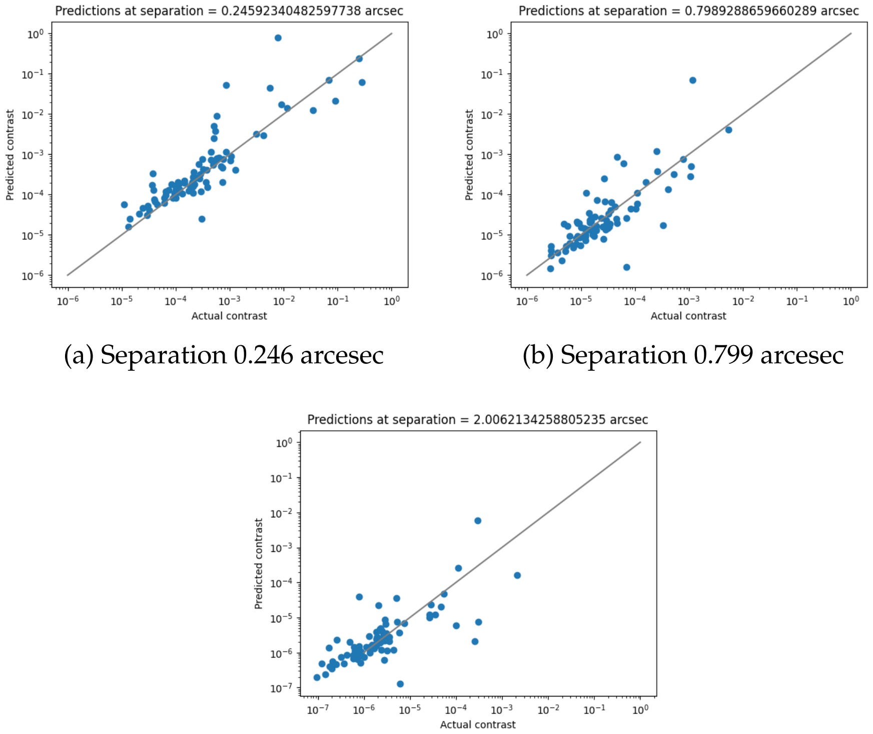
Figure 7.15: Contrast actual vs predicted values (MLP single)

Finally, the behavior of the neural network predicting a vector of contrast (Figure 7.16) closely resembles the behavior of the one that predicts single contrast values (Figure 7.15).

7.2.2 Capturing Uncertainty

Figures 7.17 show the predictions made by the model capturing the uncertainty. The red curve represents the mean of the distribution, which can be identified as the prediction of the model, while the blue area around it represents the interval $[\mu - \sigma; \mu + \sigma]$. In general, the actual value of the contrast is well captured by this interval, but in cases where the contrast curves are unusual, it may not always be the case (as shown in Figure 7.17c). It's worth mentioning that modeling uncertainty is a challenging subfield of machine learning that goes beyond simple value prediction ($\hat{y} = f(\mathbf{x})$).

Having more observations and conducting comprehensive feature selection could likely lead to better results because it would make the distribution $p(y|\mathbf{x})$ easier to model with more accurate feature information. Additionally, exploring different modeling choices beyond the normal assumption might yield better results.



(a) Separation 0.246 arcesec

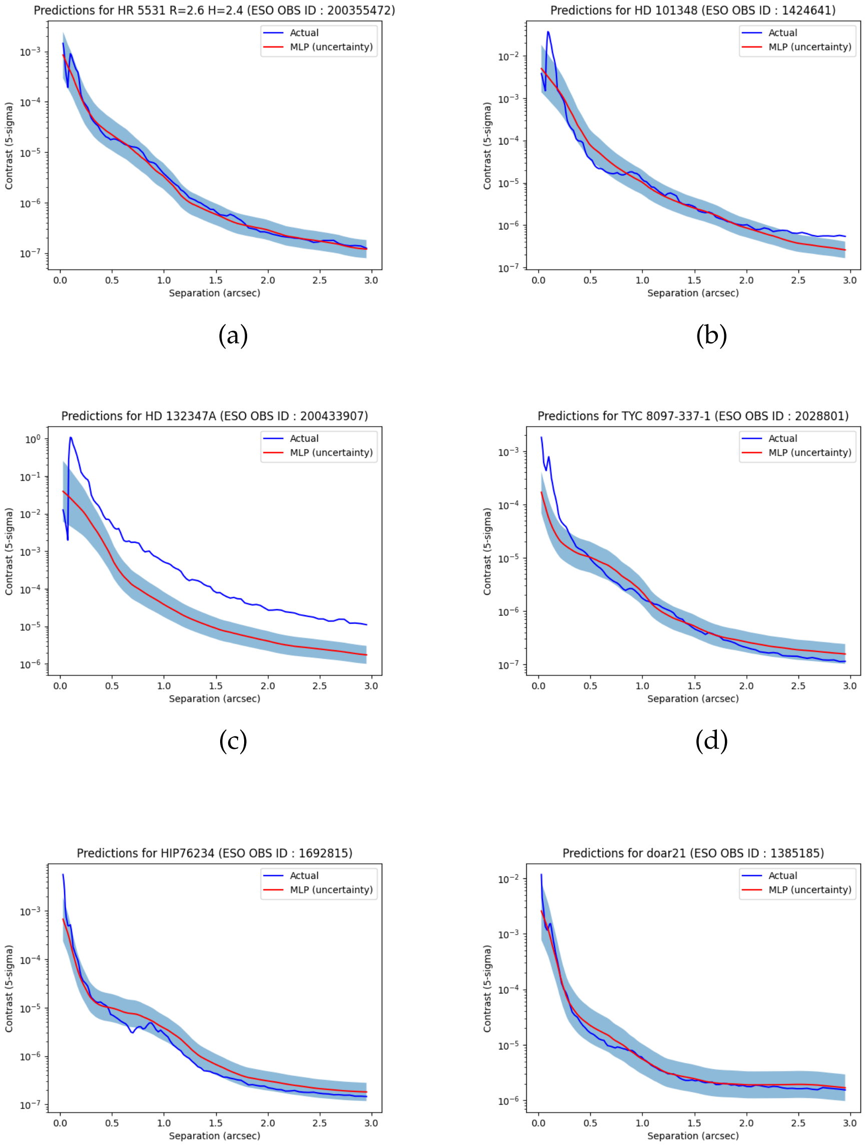
(b) Separation 0.799 arcesec

(c) Separation 2.006 arcesec

Figure 7.16: Contrast actual vs predicted values (MLP vector)

However, the results obtained here serve as a valuable starting point for capturing the uncertainty in contrast, which was the primary goal of this work given the time constraints.

Carl Sagan's quote beautifully emphasizes the importance of studying the uncertainty : "Every time a scientific paper presents a bit of data, it's accompanied by an error bar – a quiet but insistent reminder that no knowledge is complete or perfect. It's a calibration of how much we trust what we think we know."



(a)

(b)

(c)

(d)

(e)

(f)

Figure 7.17: Predictions of the model capturing uncertainty (test set)

Chapter 8

Conclusion and Future Work

The first phase of this thesis involved acquiring familiarity with the field of astronomy, which was entirely new and exciting. An internship was conducted at the Laboratoire d'Astrophysique de Marseille with the objectives of gaining an understanding of the fundamental concepts of High Contrast Imaging and establishing connections with experts in this field. Additionally, the internship aimed to collect a substantial amount of data that could be used to build machine learning models.

To gather the necessary data, numerous meetings were arranged to become familiar with the SPHERE client. In the context of this thesis, access to data not yet available to the public was granted. However, upon closer examination of the obtained data, several issues were identified. Some of these issues were relatively minor and could be resolved easily, while others proved more challenging and time-consuming. The primary issue encountered, as previously mentioned, was the lack of starting and ending times for observing sequences. Addressing these problems required significant effort and consumed several months, leading to project delays. Nonetheless, these challenges were expected when attempting to construct a dataset from scratch.

Once the data issues were resolved, the next step was the development of machine learning models to predict contrast curves. Random forests were initially used to gain insights into feature importance and establish a baseline for comparison with neural networks.

Following the use of random forests, neural networks were created to predict contrast. Due to data limitations, a relatively simple Multi-Layer Perceptron architecture was adopted. The first neural network outputted a vector, showing promising results. However, after discussions with project supervisors, it became evident that this model lacked flexibility as it did not consider separation values. Consequently, the single-output network was developed. Tuning this network, especially without using advanced tools like Sweep, proved to be more challenging.

The final results of this work are somewhat disappointing, as random forests appear to offer slightly better precision compared to neural networks. Nevertheless, the results are very similar and exhibit different behaviors. Failures in some cases of one model do not necessarily imply similar failures in the other.

Given that predicting contrast was not the sole objective of this project, a decision was made to focus on capturing uncertainty, especially in the case of neural networks. The results obtained for this type of neural network are encouraging, and further research in this direction could be fruitful.

In conclusion, it is important to acknowledge that better results might have been attainable with more rigorous data preprocessing, advanced modeling techniques, and extensive fine-tuning. However, due to time constraints, trade-offs had to be made, and it was not possible to maximize all aspects. In the future, this work could serve as a solid foundation for researchers looking to avoid redundant efforts and make incremental improvements. Specifically, for uncertainty prediction, more advanced approaches could be explored, such as outputting a prediction vector comprising the mean and a covariance matrix to capture deeper relationships within the data.

Bibliography

[1] J. -L. Beuzit et al. 'SPHERE: the exoplanet imager for the Very Large Telescope'. In: 631, A155 (Nov. 2019), A155. DOI: 10.1051/0004-6361/201935251. arXiv: 1902.04080 [astro-ph.IM].

[2] IA Bond et al. 'Real-time difference imaging analysis of MOA Galactic bulge observations during 2000'. In: *Monthly Notices of the Royal Astronomical Society* 327.3 (2001), pp. 868–880.

[3] Alan Boss et al. 'Working Group on Extrasolar Planets'. In: *Proceedings of the International Astronomical Union* 1 (Dec. 2005). DOI: 10.1017/S1743921306004509.

[4] Facundo Bre, Juan Gimenez and Víctor Fachinotti. 'Prediction of wind pressure coefficients on building surfaces using Artificial Neural Networks'. In: *Energy and Buildings* 158 (Nov. 2017). DOI: 10.1016/j.enbuild.2017.11.045.

[5] Leo Breiman. 'Random forests'. In: *Machine learning* 45 (2001), pp. 5–32.

[6] F Cantalloube et al. 'Direct exoplanet detection and characterization using the ANDROMEDA method: Performance on VLT/NaCo data'. In: *Astronomy & Astrophysics* 582 (2015), A89.

[7] Chauvin, G. et al. 'A giant planet candidate near a young brown dwarf* - Direct VLT/NACO observations using IR wavefront sensing'. In: *A&A* 425.2 (2004), pp. L29–L32. DOI: 10.1051/0004-6361:200400056. URL: https://doi.org/10.1051/0004-6361:200400056.

[8] C-H Dahlqvist et al. 'The SHARDDS survey: limits on planet occurrence rates based on point sources analysis via the Auto-RSM framework'. In: *Astronomy & Astrophysics* 666 (2022), A33.

[9] Carl-Henrik Dahlqvist. 'Advanced Data Processing Techniques for Exoplanet Detection in High Contrast Images'. Anglais. PhD thesis. ULiège - Université de Liège [Faculté des Sciences], Liège, Belgium, 7September 2022.

[10] Ph Delorme et al. 'The SPHERE Data Center: a reference for high contrast imaging processing'. In: *arXiv preprint arXiv:1712.06948* (2017).

[11] *Detecting exoplanets with astrometry*. 2019. URL: https://www.esa.int/ESA_Multimedia/Videos/2019/12/Detecting_exoplanets_with_astrometry (visited on 12/09/2023).

[12] *Detecting exoplanets with microlensing*. 2019. URL: https://www.esa.int/ESA_Multimedia/Images/2019/02/Detecting_exoplanets_with_microlensing (visited on 12/09/2023).

[13] *EE559 - Deep Learning*. 2023. URL: https://fleuret.org/dlc/ (visited on 12/09/2023).

[14] *FITS file RFC*. (Visited on 12/09/2023).

[15] Olivier Flasseur et al. 'Exoplanet detection in angular differential imaging by statistical learning of the nonstationary patch covariances-The PACO algorithm'. In: *Astronomy & Astrophysics* 618 (2018), A138.

[16] R Galicher et al. 'Astrometric and photometric accuracies in high contrast imaging: The SPHERE speckle calibration tool (SpeCal)'. In: *Astronomy & Astrophysics* 615 (2018), A92.

[17] Carlos Gómez González. 'Advanced data processing for high-contrast imaging-Pushing exoplanet direct detection limits with machine learning'. In: (2017).

[18] CA Gomez Gonzalez, Olivier Absil and Marc Van Droogenbroeck. 'Supervised detection of exoplanets in high-contrast imaging sequences'. In: *Astronomy & Astrophysics* 613 (2018), A71.

[19] Géraldine Guerri et al. 'Apodized Lyot coronagraph for SPHERE/VLT: II. Laboratory tests and performance'. In: *Experimental Astronomy* 30.1 (2011), pp. 59–81.

[20] Ping-hui Huang and Jiang-hui Ji. 'Analogue Simulation and Orbit Solution Algorithm of Astrometric Exoplanet Detection'. In: *Chinese Astronomy and Astrophysics* 41.3 (2017), pp. 399–418. ISSN: 0275-1062. DOI: https://doi.org/10.1016/j.chinastron.2017.08.008. URL: https://www.sciencedirect.com/science/article/pii/S0275106217301017.

[21] *INFO8010 - Deep Learning*. 2023. URL: https://github.com/glouppe/info8010-deep-learning (visited on 12/09/2023).

[22] *Julien Milli's github*. URL: https://github.com/jmilou/sparta (visited on 12/09/2023).

[23] Anders Krogh. 'What are artificial neural networks?' In: *Nature biotechnology* 26.2 (2008), pp. 195–197.

[24] David Lafrenière et al. 'HST/NICMOS Detection of HR 8799 b in 1998'. In: *The Astrophysical Journal* 694.2 (2009), p. L148.

[25] *Light Curve of a Planet Transiting Its Star*. URL: https://exoplanets.nasa.gov/resources/280/light-curve-of-a-planet-transiting-its-star/ (visited on 12/09/2023).

[26] Bernard Lyot. 'The study of the solar corona and prominences without eclipses (George Darwin Lecture, 1939)'. In: *Monthly Notices of the Royal Astronomical Society, Vol. 99, p. 580* 99 (1939), p. 580.

[27] Anne-Lise Maire et al. 'SPHERE IRDIS and IFS astrometric strategy and calibration'. In: *Ground-based and Airborne Instrumentation for Astronomy VI*. Vol. 9908. SPIE. 2016, pp. 975–986.

[28] Christian Marois et al. 'Angular Differential Imaging: A Powerful High-Contrast Imaging Technique'. In: *The Astrophysical Journal* 641.1 (Apr. 2006), pp. 556–564. DOI: 10.1086/500401. URL: https://doi.org/10.1086%2F500401.

[29] Christian Marois et al. 'Direct imaging of multiple planets orbiting the star HR 8799'. In: *science* 322.5906 (2008), pp. 1348–1352.

[30] Warren S McCulloch and Walter Pitts. 'A logical calculus of the ideas immanent in nervous activity'. In: *The bulletin of mathematical biophysics* 5 (1943), pp. 115–133.

[31] Ian S McLean, Suzanne K Ramsay and Hideki Takami. 'Ground-based and Airborne Instrumentation for Astronomy IV'. In: *Ground-based and Airborne Instrumentation for Astronomy IV* 8446 (2012).

[32] Didier Queloz Michel Mayor. 'A Jupiter-mass companion to a solar-type star'. In: *Nature* (1995). DOI: 10.1038/378355a0.

[33] *Montefiore Alan GPU cluster*. URL: https://github.com/montefiore-ai/alan-cluster (visited on 12/09/2023).

[34] NASA. 2023. URL: https://exoplanets.nasa.gov (visited on 01/09/2023).

[35] Thomas Rimmele and Jose Marino. 'Solar Adaptive Optics'. In: *Living Reviews in Solar Physics* 8 (May 2011), p. 2. DOI: 10.12942/lrsp-2011-2.

[36] Frank Rosenblatt. *The perceptron, a perceiving and recognizing automaton Project Para*. Cornell Aeronautical Laboratory, 1957.

[37] G Rousset et al. 'First diffraction-limited astronomical images with adaptive optics'. In: *Astronomy and Astrophysics (ISSN 0004-6361), vol. 230, no. 2, April 1990, p. L29-L32. Research supported by the European Southern Observatory, Ministere de la Recherche et de la Technologie, Ministere de l'Education Nationale, INSU, DRET, and Ministere de la Defense.* 230 (1990), pp. L29–L32.

[38] Matthias Samland et al. 'TRAP: A temporal systematics model for improved direct detection of exoplanets at small angular separations'. In: *Astronomy & Astrophysics* 646 (2021), A24.

[39] Jean-Francois Sauvage et al. 'SAXO: The extreme adaptive optics system of SPHERE (I) system overview and global laboratory performance'. In: *Journal of Astronomical Telescopes, Instruments, and Systems* 2 (May 2016), p. 025003. DOI: 10.1117/1.JATIS.2.2.025003.

[40] J. Schneider. 2023. URL: http://exoplanet.eu/catalog/ (visited on 01/09/2023).

[41] *Scikit Learn Random Forest Regressor*. URL: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html (visited on 12/09/2023).

[42] William Smith. 'Spectral differential imaging detection of planets about nearby stars'. In: *Publications of the Astronomical Society of the Pacific* 99 (Jan. 1988). DOI: 10.1086/132124.

[43] Rémi Soummer, Laurent Pueyo and James Larkin. 'Detection and characterization of exoplanets and disks using projections on Karhunen–Loève eigenimages'. In: *The Astrophysical Journal Letters* 755.2 (2012), p. L28.

[44] *SPHERE Data Center*. URL: https://sphere.osug.fr/spip.php?rubrique16&lang=en (visited on 12/09/2023).

[45] *Sweep : Parameter Importance*. URL: https://docs.wandb.ai/guides/app/features/panels/parameter-importance (visited on 12/09/2023).

[46] *Sweep tutorial from Samuel Cortinhas*. URL: https://www.kaggle.com/code/samuelcortinhas/advanced-wandb-hyper-parameter-tuning-sweeps (visited on 12/09/2023).

[47] *The open university : An introduction to exoplanets*. URL: https://www.open.edu/openlearn/mod/oucontent/view.php?id=87798&section=_unit4.3.1 (visited on 12/09/2023).

[48] *The radial velocity method (artist's impression)*. 2007. URL: https://www.eso.org/public/belgium-fr/images/eso0722e/?lang (visited on 12/09/2023).

[49] *Very Large Telescope*. URL: https://www.eso.org/public/belgium-fr/teles-instr/paranal-observatory/vlt/ (visited on 12/09/2023).

[50] *Weights and Biases*. URL: https://wandb.ai/site (visited on 12/09/2023).

[51] Aleksander Wolszczan and Dail A Frail. 'A planetary system around the millisecond pulsar PSR1257+ 12'. In: *Nature* 355.6356 (1992), pp. 145–147.

[52] Chen Xie et al. 'Reference-star differential imaging on SPHERE/IRDIS'. In: *arXiv preprint arXiv:2208.07915* (2022).

[53] W Jerry Xuan et al. 'Characterizing the performance of the NIRC2 vortex coronagraph at WM Keck Observatory'. In: *The Astronomical Journal* 156.4 (2018), p. 156.