UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

EXTRACTION AND CLASSIFICATION OF OBJECTS FROM ASTRONOMICAL IMAGES IN THE PRESENCE OF LABELING BIAS

TESIS PARA OPTAR AL GRADO DE DOCTOR EN CIENCIAS DE LA COMPUTACIÓN

GUILLERMO FELIPE CABRERA VIVES

PROFESORES GUÍA:
NANCY HITSCHFELD KAHLER
BENJAMÍN BUSTOS CÁRDENAS

PROFESOR CO-GUÍA:
CHRISTOPHER MILLER

MIEMBROS DE LA COMISIÓN:
BÁRBARA POBLETE LABRA
PABLO GUERRERO PÉREZ
PAVLOS PROTOPAPAS

SANTIAGO, CHILE
2015

Resumen

Giga, tera y petabytes de datos astronómicos están empezando a fluir desde la nueva generación de telescopios. Los telescopios de rastreo escanean una amplia zona del cielo con el fin de mapear la galaxia y nuestro universo. Al igual que en otros campos de la ciencia observacional, lo único que podemos hacer es *observar* estas fuentes a través de la luz que emiten y que podemos capturar en nuestras cámaras. Debido a la gran distancia a la que estos objetos se encuentran, aún cuando podemos tener una caracterización estimada de estas fuentes, es imposible conocer las propiedades reales de ellas.

En esta tesis, proponemos un método para la extracción de los llamados perfiles de Sérsic de fuentes astronómicas y su aplicación a clasificación morfológica de objetos. Este perfil de Sérsic es un modelo paramétrico radial asociado con la morfología de galaxias. La novedad de nuestro enfoque es que convierte la imagen 2D en un perfil radial 1D utilizando curvas de nivel elípticas, por lo que incluso cuando el espacio de parámetros de Sérsic es el mismo, la complejidad se ve reducida 10 veces en comparación a ajustes de modelos en 2D de la literatura. Probamos nuestro método sobre simulaciones y obtenemos un error de entre un 40% y un 50% en los parámetros de Sérsic, mientras que obtenemos un χ^2 reducido de 1,01. Estos resultados son similares a los obtenidos por otros autores, lo que sugiere que el modelo de Sérsic es degenerado. A su vez, aplicamos nuestro método a imágenes del SDSS y mostramos que somos capaces de extraer la componente suave del perfil de las galaxias, pero, como era de esperar, fallamos en obtener su estructura más fina.

También mostramos que las etiquetas creadas por los seres humanos son sesgadas en términos de parámetros observables: al observar galaxias pequeñas, débiles o distantes, la estructura fina de estos objetos se pierde, produciendo un sesgo en el etiquetado sistemático hacia objetos más suaves. Creamos una métrica para evaluar el nivel de sesgo en los catálogos de las etiquetas y demostramos que incluso etiquetas obtenidas por expertos muestran cierto sesgo, mientras que el sesgo es menor para etiquetas obtenidas a partir de modelos de aprendizaje supervisado. Aun cuando este sesgo ha sido notado en la literatura, hasta donde sabemos, esta es la primera vez que ha sido cuantificado. Proponemos dos métodos para des-sesgar etiquetas. El primer método se basa en seleccionar una sub-muestra no-sesgada de los datos para entrenar un modelo de clasificación, y el segundo método ajusta simultáneamente un modelo de sesgo y de clasificación a los datos. Demostramos que ambos métodos obtienen el sesgo más bajo en comparación con otros conjuntos de datos y procedimientos de procesamiento de etiquetas ruidosas en la literatura. Mostramos también, a través de simulaciones, que al entrenar sobre un conjunto de datos sesgados, se obtiene un modelo sesgado y menos preciso. Nuestros métodos son capaces de reducir el sesgo y mantener una alta precisión.

Abstract

Giga, tera and petabytes of astronomical data is starting to flow from the new generation of telescopes. Survey telescopes scan a large area of the sky in order to map the galaxy and our universe As in other observational scientific fields, all we can do is to *observe* these sources through the light they emit and that we can capture with our cameras. Due to the big distance these objects are at, though we can have an estimate of their characterization, it is impossible to know their real properties.

In this thesis, we propose a method for extracting the so called Sérsic profiles from astronomical sources and apply it to morphological classification of objects. This Sérsic profile is a radial parametric model usually associated with the morphology of galaxies. The novelty of our approach is that it turns the 2D image into a 1D radial profile using elliptical contour lines, so even when the Sérsic parameter space is the same, the complexity is reduced 10 times as compared to 2D model fit in the literature. We tested our method over simulations and obtained between a 40% and a 50% error in our Sérsic parameters while achieving a reduced χ^2 of 1.01. These results are similar to results achieved by other authors, suggesting that the Sérsic model is degenerate, in the sense that different sets of parameters can give the same light profile. We also apply this method to real images from the SDSS and show that though we are capable of extracting the smooth component of the galaxy profiles, as expected, we miss finer structure.

We also show that labels given by humans are biased in terms of observable parameters: when observing small, faint or distant galaxies, fine structure of these objects is lost, causing a systematic labeling bias towards smoother objects. We create a metric to assess the level of bias in catalogs of labels and show that even labels created by experts show some bias, while a lower bias is obtained from machine learned classification models. Though this bias has been noticed in the literature, to the best of our knowledge, this is the first time it has been quantified. We propose two methods for de-biasing labels. The first method is based on selecting an un-biased sub-sample of the data to train a classification model, and the second method simultaneously fits a bias and classification model to the data. We show that both methods achieve the lowest bias when compared to other data-sets and de-noising procedures in the literature. We also show through simulations that when training over a biased data-set the obtained labels are biased. This bias translates into a lower accuracy in terms of the real *ground truth* labels. Our methods are able to reduce the bias and maintain a high accuracy.

A mi familia

Acknowledgments

I would like to thank my wife Mariajose for being with me at all times (good and bad ones!) helping and supporting me in all I needed, specially during long days and nights of work, always with a smile in her heart.

I would also like to thank my family, including my father, Guillermo, my mother, Ana María, her husband, Alejo, my sisters, Anita, Sole, and Vale, and my brother Javier for all the support they have given me through all my life.

I am specially grateful of Eduardo Vera and Chris Smith for supporting all my crazy ideas and projects, with excellent disposition and always giving me wise advise, specially when taking career decisions.

I would also like to thank my mentor and friend, Chris Miller, for agreeing to work with me several years, permanently teaching me all I asked with a cheerful and friendly disposition (specially the meaning of the word feisty). I also thank Nancy Hitschfeld for being such a great devoted teacher, always having an ear for her students. Thanks to Benjamín Bustos as well for giving me such great advice not only technically, but also in life. I also wish to thank Jaime San Martín for most valuable discussions and his continuous support of our project. Thanks to Alfredo Zenteno also, specially for allowing me to use his computer.

I am specially grateful of the Center for Mathematical Modeling (CMM) of the University of Chile and the AURA Observatory in Chile for their support of my entire research work at the Astroinformatics Laboratory of CMM. I also acknowledge the support from CONICYT through its Scholarship for Ph.D. Studies in Chile, 2011, and its Doctoral Internship Grant, 2012

Powered@NLHPC: This research was partially supported by the supercomputing infrastructure of the NLHPC (ECM-02). Most of the table operations and plots were done using TOPCAT (Taylor, 2005) and matplotlib (Hunter, 2007). Machine learning methods were performed using scikit-learn (Pedregosa et al., 2011).

Contents

Chapter 1

Introduction

The amount of astronomical data, as in many other fields, has increased exponentially during the last couple of years. We have evolved from the Megabyte to the Gigabyte and to the Terabyte regime in less than a decade. Thanks to the special characteristics of the skies in Chile (low humidity, high peaks and plains, low light pollution, and large number of clear nights), during the following years, most of the major observatories will be installed here, such as the Atacama Large Millimeter/submillimiter Array (ALMA, already operative), the Giant Magellan Telescope (GMT, 2021), the Large Synoptic Survey Telescope (LSST, 2021), and the European Extremely Large Telescope (E-ELT, 2022). These instruments will produce petabytes of data. Two examples of the amount of data produced are:

DECam: The Dark Energy Camera is mounted over the Blanco 4-meter telescope at Cerro Tololo Inter-American Observatory (CTIO). Its imager contains 62 2048×4096 science image arrays with a total of 520 megapixels. Each image is approximately one gigabyte producing between 300 and 500 gigabytes per night. During its running life DECam will produce a data volume of between one and five petabytes.

LSST: The Large Synoptic Survey Telescope is an 8.5 meters telescope whose objective is to scan the whole sky every three nights. The 30 terabytes of data obtained every night will open enormous possibilities for new science, specially being able to consider the time domain. These 30 TB per night should be stored, transmitted, processed and mined in real time (more than 1 TB per hour).

When studying and cataloging astronomical objects from survey telescopes, astronomers must face a couple of challenging issues in image processing before getting into the actual analysis. The detection of objects inside an image is not trivial. Apart of having to deal with background, *point spread function* (PSF, defined in detail in

Section 2.2.1) convolution, and noise, two or more objects may cross the line of sight, making the detection, extraction and classification very difficult. This thesis focuses in these three problems: automated detection, extraction, and classification of objects in astronomical images.

1.1 The Problem

Optical survey telescopes scan a large area of the sky in order to observe a lot of objects, including galaxies, stars, asteroids, etc. They usually produce an important number of images from where these sources must be detected and statistically studied. These images are by no means an exact representation of the objects: when the light from these sources crosses the atmosphere and goes into the telescope it gets blurred by the so called Point Spread Function (PSF), it gets sub-sampled by the pixels, and source and image charged-coupled device (CCD) readout noise is added. Furthermore, two objects may be overlapped through the line of sight, causing the distinction between them as separated sources to be another issue. All these effects make the detection, extraction, and classification of astronomical sources an interesting challenge. In this section we will describe these problems in detail.

Detection challenges: One of the most difficult problems when detecting objects in astronomical images is being able to separate which pixels contain photons from the source and which pixels are only noise. This problem gets more important when trying to detect very dim objects. The other important problem when detecting objects is that the PSF convolution may *blend* two objects that are too close (in terms of angular distance from the perspective of the observer).

Extraction challenges: Once objects are detected, it is very important to know their shape and amount of brightness, as accurately as possible. Again, PSF convolution, noise and blending make this a challenging step. In particular, features from dim, small, or blended spiral galaxies become hard to distinguish. Furthermore, the PSF convolution spreads the light of the source through an important amount of pixels around the object, below the level of the noise, so pixels get correlated and light is hard to distinguish below this level.

Classification challenges: Classification is the process of identifying to which category (or categories) an object belongs to. Again, because of PSF convolution and noise, the classification of astronomical sources gets tricky. For example, in the case of spiral galaxies we would expect to see spiral arms, but for low resolution images of galaxies, and/or dim galaxies these spiral arms are difficult to detect, even for the human eye. This means that small galaxies may be confused with

stars (or vice versa), and small spiral galaxies may be confused with galaxies with no arms, or even stars. In that sense, it is of great importance to take into account this bias towards smoother sources when classifying objects.

These problems have been studies for a couple of decades by the astronomy community for relatively small amount of data. During the last couple of years, the exponential growth of data has posed the need for new automated methods. This thesis addresses the problem of automatically analyzing gigabytes of astronomical data in the presence of observable biases, a problem that, though it has been addressed in the literature, still has a lot of space for improvement.

1.2 Main Contribution

Our main contribution can be divided into extraction of objects based on radial parametric models, using these parametric models to automatically classify these objects, and measuring the observational bias in labeled data-sets. Here we give a brief summary of our contribution, and a detailed description of it can be found in section 3.2.

We created a method that fits radial Sérsic models (Sersic, 1968) to elliptical isophotes calculated over astronomical objects. We improved the fitting complexity by a factor of 10 as compared to 2D fits found in the literature.

We also present two metrics for assessing labeling bias in terms of observable parameters. To the best of our knowledge, this is the first time this bias has been measured. We propose two methods for training an un-biased classification model. Using simulations, we show that our methods perform better than state of the art de-noising methods, achieving a similar accuracy as if were trained over the un-biased (latent) labels.

We use the Sérsic profiles to classify objects and achieve over a 93% accuracy when applying this model to morphological galaxy classification (as compared to $\sim$90% in the literature), while keeping a lower observational bias than data-sets in the literature. The Sérsic parameter space is able to accurately recover morphologies while keeping a low observational bias.

1.3 Overview

This thesis has been developed between 2011 and 2014. Its goal is to develop automated methods for astronomical object detection, extraction and classification, able to run on

a high performance computing environment. This thesis is divided into the following chapters:

- Chapter 2 describes all basic concepts needed to read this thesis.
- Chapter 3 describes previous work and explains our contribution in this context.
- Chapter 4 shows the approach we are taking to solve the problem of extraction. This is a two-step iterative approach, where we detect candidate of objects by first running a noise-removal low pass filter over the image, and then we extract sources by modeling a Sérsic radial profile to the candidates. We empirically analyze the PSF-convolved Sérsic profile, and show that it is degenerated, in the sense that similar light profiles can be obtained for different combination of parameters.
- Chapter 5 describes the problem of bias in eyeball labeling of astronomical objects. We create two metrics to asses the amount of bias in labeled data-sets and show that even labels created by experts present some level of bias.
- Chapter 6 describes two approaches for training supervised learning models in the presence of labeling bias. The first approach consists of selecting an unbiased sub-set of the original data and train the model using that sub-set. The second approach consists of training simultaneously a probabilistic bias model and the classification model by using the Expectation-Maximization algorithm that considers the real labels as latent variables. We show that both approaches perform better than previous de-biasing and de-noising algorithms.
- Chapter 7 summarizes this thesis and presents some insights on possible future directions for detection, extraction, and classification of astronomical sources.

Chapter 2

Basic Concepts

2.1 Basic Astronomical Concepts

In this section we briefly describe the basic astronomical concepts needed to understand this thesis. A detailed explanation of these concepts can be found in (Carroll and Ostlie, 2006).

2.1.1 Measuring Light

Astronomical sources, such as stars and galaxies, emit light in different wavelengths. We start by defining the *luminosity* L_ν of an astronomical source (say, a star) as the amount of energy it emits per second for a particular frequency ν. The *flux density* at such frequency F_ν is defined as the amount of energy per unit area. Assuming spherical symmetry, the flux density of an astronomical source at a radius R is

$$F_\nu = \frac{L_\nu}{4\pi R^2}. \tag{2.1}$$

When observing through a telescope different filters can be used. Each filter has different sensitivity to incident radiation, which is not necessarily constant. For example, Figure 2.1 shows the SDSS-III camera filter throughput curves for its different filters. In order to calculate fluxes for a particular filter one integrates F_ν with the response of that filter in terms of the frequency. The *bolometric* flux is the total amount of energy at all frequencies, i.e. the integral of F_ν over ν.

Another definition for measuring light used in astronomy is that of magnitudes. Magnitudes turn fluxes into a logarithmic scale and are defined as

$$m = C_\mathrm{m} - 2.5 \log_{10} F, \tag{2.2}$$

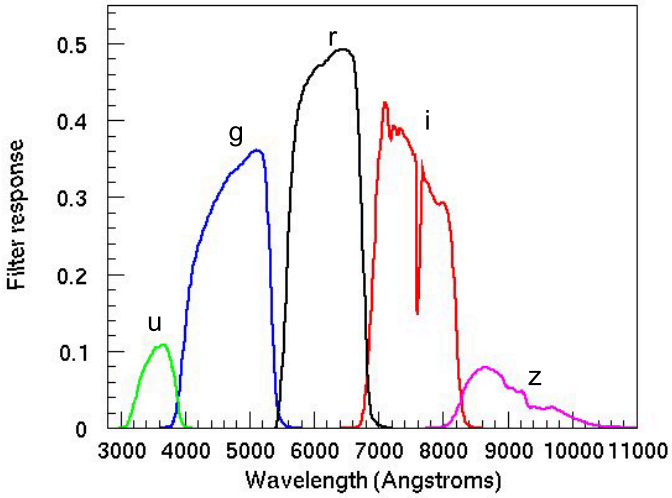


Figure 2.1: Response of the different filters of the SDSS-III camera. The SDSS has three filters: u, g, r, i, and z, each of which have different sensitivity in terms of the radiation received. Image obtained from www.sdss3.org.

where $C_{\rm m}$ is a constant defined by a reference object with flux $F_{\rm ref}$ and magnitude $m_{\rm ref}$ as $C_{\rm m} = m_{\rm ref} + 2.5\log_{10}(F_{\rm ref})$, leading to:

$$m = m_{\rm ref} - 2.5\log_{10}\left(\frac{F}{F_{\rm ref}}\right). \tag{2.3}$$

We call these *apparent magnitudes*, as they represent the light as seen from Earth. Of course, the apparent magnitude is going to depend on the distance of the source from us. In order to compare the intrinsic amount of light of different objects at different distances from us, we use the *absolute magnitude*, which is the magnitude of sources at standard luminosity distance of exactly 10 parsecs.

2.1.2 Redshift

When objects move radially away from the observer, their light increases in wavelength (or moves to the red end of the electromagnetic spectrum). This effect is called redshift. The redshift z is calculated by measuring the relative shift of the light:

$$z = \frac{\nu_{\rm emit} - \nu_{\rm obs}}{\nu_{\rm obs}} = \frac{\lambda_{\rm obs} - \lambda_{\rm emit}}{\lambda_{\rm emit}}, \tag{2.4}$$

where $\nu_{\rm emit}$, and $\lambda_{\rm emit}$ are the emitted frequency and wavelength, and $\nu_{\rm obs}$, and $\lambda_{\rm obs}$ are the observed frequency and wavelength, respectively. Objects with $z > 0$ are moving away from us, and objects with $z < 0$ are moving towards us.

Due to the expansion of the universe galaxies farther away are moving faster from us. In that sense, redshift is directly related to the distance between galaxies and us.

2.1.3 Petrosian Magnitude and Radius

Due to PSF convolution and noise (see next section), the light from observed objects decays smoothly from its center. Because of this, it is not possible to define an exact border for stars and galaxies. A way of measuring the sources' size is by selecting a radius at which some criteria on the amount of light is met.

The *Petrosian ratio* $\mathcal{R}_{\rm P}$ at a radius r from the center of a source is defined as the ratio of the local surface brightness in an annulus (region within two concentric disks) at r to the average surface brightness within r (see Blanton et al., 2001). For the particular case of the Sloan Digital Sky Survey (SDSS), the annulus is defined at radii between $0.8r$ and $1.25r$:

$$\mathcal{R}_{\rm P}(r) = \frac{\int_{0.8r}^{1.25r} dr' 2\pi r' I(r') / [\pi(1.25^2 - 0.8^2) r^2]}{\int_0^r dr' 2\pi r' I(r') / \pi r'^2}, \tag{2.5}$$

where $I(r)$ is the mean radial surface brightness profile.

The *Petrosian radius* $r_{\rm P}$ is defined as the radius at which the Petrosian ratio $\mathcal{R}_{\rm P}(r_{\rm P})$ equals some specified value $\mathcal{R}_{\rm lim}$. For the case of the SDSS they have set $\mathcal{R}_{\rm lim} = 0.2$. In other words, the Petrosian radius is defined as the radius at which the average surface brightness within an annulus is 20% the light of the total surface brightness of the source within that radius.

Using the Petrosian radius, we define the *Petrosian magnitude* as the magnitude associated with the flux of the source within $N_{\rm P}$ times the Petrosian radius. This Petrosian flux is then defined as

$$F_{\rm P} = \int_0^{N_{\rm P} r_{\rm P}} dr' 2\pi r' I(r'). \tag{2.6}$$

For the case of the SDSS they chose $N_{\rm P} = 2$. The SDSS has 5 filters, and they measure magnitudes and radii for each of them.

2.2 Astronomical Images

Astronomical images are taken by a telescope, which contains a charged-coupled device (CCD) camera and may be located in ground or in space. CCD pixels are not completely efficient: only a fraction of the photons received by the pixels release an

electron that is captured within the potential well of the pixel. This is called the *quantum efficiency*, and it depends on the wavelength of the light ($\sim 80\%$ for the visual). Electrons detected by the CCD pixels are converted into counts, or *Analogue-to-Digital conversion Units* (ADUs). The number of ADUs per electron is called the *gain*.

Images are not an exact representation of the light coming from the astronomical objects. This light is convolved with a point spread function (PSF) (caused by the atmosphere, the telescope, and the pixels of the camera), some background intensity is added by close-by bright objects and noise is added from the CCD camera and the signal itself. In this section we will explain in detail each of these artifacts, which have to be taken into account when processing astronomical images.

2.2.1 Point Spread Function

When light from astronomical objects crosses the atmosphere and enters the telescope, some aberration is produced. The response of the instrument to a point source is called the point spread function (PSF), and it is usually a smooth function with a central peak. The shape of the PSF is the result of the convolution of a star (usually assumed to be similar to a Dirac delta function) with the *seeing* (aberration caused by the atmosphere), the diffraction caused by optics, and the pixel sampling of the CCD. The PSF usually resembles a Gaussian, but its exact shape depends on the instrument and the weather conditions at the time the image was observed. Furthermore, the shape of the PSF may vary within the image. In order to determine the PSF, a usual practice is to measure the observed pattern of known stars.

2.2.2 Background

When taking pictures of the sky, one would expect no brightness in zones where there are no objects. This is never true because of light coming from objects close to the pointing (point where the telescope aims at) and Earth sources (like cities) which diffuses through the atmosphere. This luminosity is added to the image, and although it can vary through the image, it is usually very smooth, and sometimes even considered to be constant.

2.2.3 Noise

There are two main sources of noise within CCD pixels: thermal noise produced by the CCD camera (readout noise), and noise from the incoming photons. Astronomers use CCD cameras for most of their ultraviolet to infrared telescopes. These cameras

are composed of CCD sensors that are able to count photons and transform them to digital counts. But these image sensors are also sensible to thermal noise generated by the thermal agitation of electrons. This produces random noise over the image which is usually modeled as Gaussian noise.

At the same time, the rate of incoming photons follows a Poisson distribution: if we expect λ_{p} photons to arrive over a specific time frame, the probability of exactly obtaining N_{p} photons is

$$p(N_{\mathrm{ph}}) = \frac{{\lambda_{\mathrm{p}}}^{N_{\mathrm{p}}} e^{-\lambda_{\mathrm{p}}}}{N_{\mathrm{p}}!}. \tag{2.7}$$

The mean of the Poisson distribution is λ_{p} and its standard deviation is $\sqrt{\lambda_{\mathrm{p}}}$. Incoming photons include photons from the signal, from the background and from the *dark current* (charge in each pixel due to the CCD being at a temperature above zero). When $N_{\mathrm{p}}, \lambda_{\mathrm{p}} \to \infty$ this Poisson distribution can be approximated by a Gaussian distribution with mean λ_{p} and standard deviation $\sqrt{\lambda_{\mathrm{p}}}$.

Both sources of noise are independent, so assuming the readout noise is σ_{r}, the overall noise per pixel will be

$$\sigma^{\mathrm{n}} = \sqrt{\sigma_{\mathrm{r}}^2 + \lambda_{\mathrm{p}}}. \tag{2.8}$$

It is important to notice that though the readout noise may be constant for each pixel, the photon noise will depend on the light being captured, including background and astronomical objects.

2.2.4 Mathematical Representation

Taking into account PSF convolution, background luminosity $b_{i,j}$ and random noise $\varepsilon^{\mathrm{n}}_{i,j}$, the obtained intensity at pixel (i, j) is

$$D_{i,j} = (PSF * I)_{i,j} + b_{i,j} + \varepsilon^{\mathrm{n}}_{i,j}, \tag{2.9}$$

where I is the non-modified light coming from astronomical objects (spacially distributed in the image) and $*$ stands for convolution.

Figure 2.2 shows a high resolution image, its convolution with a PSF and Gaussian noise addition. It can be seen how faint and small features are very hard to distinguish on the final image.

Large survey astronomical images contain different kind of objects. They are usually a large set of point sources (stars), some spiked objects (galaxies), and some complex and diffuse structures (galaxies, nebulae, etc.). As said before, all the light coming from these objects is convolved with the PSF, and some random noise and smooth background is added.

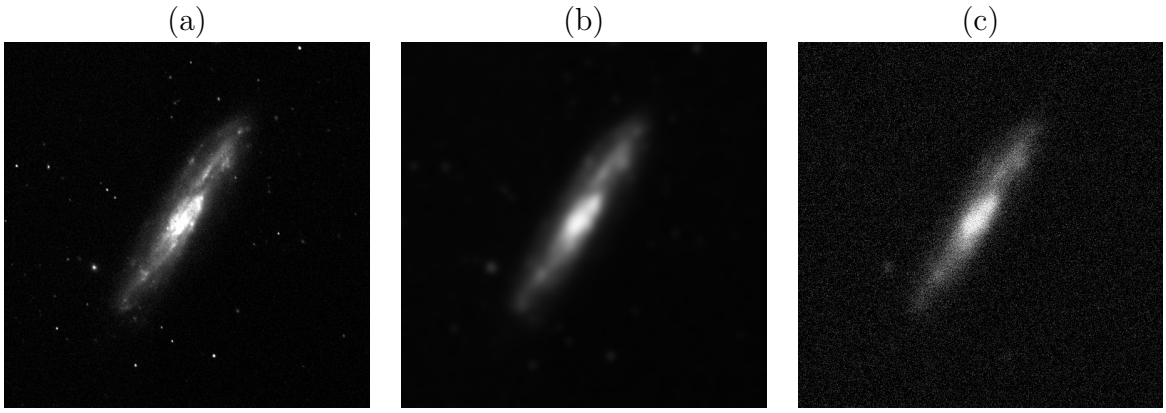


Figure 2.2: A simulation of the formation process behind astronomical images. a) A 512×512 high resolution image. b) Convolution with a Gaussian PSF of 5 pixels standard deviation. c) Addition of Gaussian white noise with a standard deviation of 10% the peak intensity on b.

Considering a set of N_{obj} objects, each with a luminosity $I_k(\mathbf{x})$ at coordinate $\mathbf{x}$, and $k = 1, \ldots, N_{\text{obj}}$, Equation 2.9 can be written as

$$D_{i,j} = (PSF * \sum_k I_k)_{i,j} + b_{i,j} + \varepsilon^{\text{n}}_{i,j}, \tag{2.10}$$

$$= \sum_k (PSF * I_k)_{i,j} + b_{i,j} + \varepsilon^{\text{n}}_{i,j}. \tag{2.11}$$

As mentioned in Starck and Murtagh (2006) a standard method for detecting astronomical sources should perform the following tasks:

1. Background estimation.
2. Convolution with a mask.
3. Object detection.
4. Deblending / merging.
5. Photometry (measurement of the amount of light per object).
6. Classification.

We will group background estimation, convolution, and object detection into the steps required for *detection*, deblending and photometry into *extraction*, and will consider classification on its own. After applying all these steps, we should be able to end with a catalog of sources. This catalog should have information about position, amount of light and type of each object. In this section we will explain each of these steps.

2.3 Detection

2.3.1 Background Estimation

In order to obtain accurately the amount of light coming from each object, an important step to perform is the estimation of the background for each pixel. If background is not calculated correctly, it will bias the amount of light of each object.

If there are no objects on the image, the background should be the average of intensities over its pixels. This is also useful for calculating the distribution of the noise in the image by assuming a noise model and estimating its parameters (such as the standard deviation). Most algorithms calculate the background at different sparse points, and then return a value for each pixel by interpolating on the whole field. Usually they use a window where only light under a certain threshold is considered in order to avoid sources.

2.3.2 Convolution with a Mask (Low Pass Filtering)

Convolving the noisy image with a low pass filter is a helpful way of diminishing the influence of noise on the detection step. Usually a filter similar to the PSF is used. Objects at the image will not have a resolution smaller than the PSF. Stars, which are represented as point sources, in the data image will have the shape of the PSF. So convolving the image with a PSF-like function is a suitable choice to keep signal from objects, as higher frequencies from the source have already been lost by the original PSF convolution. In that sense, the convolution will only remove high frequencies from the noise and keep observed frequencies from the sources.

2.3.3 Detection

Detection of objects is a fundamental step in astronomical image processing. We would like to find precisely the position of objects and distinguish them from unreal artifacts.

Different approaches have been used for detection. Most of them use thresholding, considering pixels with light intensity over a certain value (the threshold) as part of objects. This threshold is usually taken dependent on the noise. If the standard deviation of the background noise is σ, the threshold is usually chosen as $k\sigma$ where k is an arbitrary constant value, usually picked between 1 and 3.

2.4 Extraction

2.4.1 Deblending

When two objects cross the line of sight, light coming from them is added. These stars and galaxies are said to be *blended*. This blending happens more usually than one would think specially on star and galaxy clusters (set of stars or galaxies bounded by gravitational forces). The process of distinguishing how much light comes from each one of them is called *deblending*.

Naive detection algorithms will assume that the light coming from blended systems belong to just one object. These is why an additional deblending step is required in order to find the exact location and amount of light coming from each object.

2.4.2 Photometry

Photometry is the process of measuring the exact flux or intensity coming from each object. This is not a trivial step. The convolution with the PSF implies that the light coming from each object will be distributed over the whole image. Also, it is very important to separate light coming from blended systems.

2.5 Classification

Once brightness from every object is accurately known, astronomers need to know whether those objects are stars, galaxies, asteroids, or anything else. This is when classification algorithms are needed. Most of these algorithms consider the shape of the light profile: stars are supposed to follow the PSF, while galaxies are more diffuse. Objects can be classified in terms of their attributes, such as their roundness or sharpness. These attributes are used within a classification model which takes as input these attributes and outputs what class the object is.

Conceptually classification is a supervised machine learning problem. Consider a typical supervised learning problem, where a labeled data-set $\mathcal{D} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{N}$, is used to determine a learning function $f(\boldsymbol{x}) : \mathcal{X} \to \mathcal{Y}$ which predicts new data. This goal is achieved by using N instances $\boldsymbol{x}_i \in \mathcal{X}$ and their respective labels $y_i \in \mathcal{Y}$ for obtaining this function. When $\mathcal{Y} = \{0, 1\}$ we call it a binary classification problem, when $\mathcal{Y} = \{1, 2, \cdots, K\}$ we call it a multi-class classification problem, and when $\mathcal{Y} = \mathbb{R}$ we call it regression.

2.5.1 Relevant classification models

Different classification models may be used (e.g. logistic regression, support vector machines, random forests, neural networks, etc., see Hastie et al. (2009) and references therein) to approximate $f(\boldsymbol{x})$, and each of them has a different learning algorithm which fits the model to the labeled data-set. In this thesis, we focus mostly on logistic regression (LR) and support vector machines (SVM). LR gives us probabilities of an instance to be of a certain class and it is easy to implement as a proof of concept when proposing new algorithms that use any classification model. On the other hand, though it is not always the model that best fits the data, using SVM is possible to obtain an analytical hypersurface for splitting the parameter space into two, which makes it simpler for scientists to classify objects directly with no need to run a sophisticated model.

Logistic Regression

Logistic Regression (LR) is a binary probabilistic linear classification model, in the sense that it models the probability of a label given its feature vector $\mathrm{P}(y|\boldsymbol{x})$. This probability is modeled through a parametric function

$$\mathrm{P}(y|\boldsymbol{x}) \equiv p_{\boldsymbol{w}}(\boldsymbol{x}) = \frac{1}{1 + e^{-\boldsymbol{w}^\top \boldsymbol{x}'}}, \tag{2.12}$$

where $\boldsymbol{w}$ is a parameter vector which has to be fitted to the data in order to define the *logistic function* $p_{\boldsymbol{w}}(\boldsymbol{x})$, and $\boldsymbol{x}' = (1, \boldsymbol{x})$ is a an extended vector of $\boldsymbol{x}$ by adding 1 as the first coordinate. The reasons for choosing this particular function are the following:

- We classify objects with a probability lower than 0.5 to be of one class and objects with a probability greater than 0.5 to be of the other class (say "0" and "1"). The logistic function is a linear discriminator in the sense that the 0.5 probability threshold occurs when $\boldsymbol{w}^\top \boldsymbol{x} = 0$. This is a linear decision boundary.
- The logistic function is bounded between 0 and 1.
- The logistic function is a monotonic function, for which $\lim_{(\boldsymbol{w}^\top \boldsymbol{x} \to -\infty)} p_{\boldsymbol{w}}(\boldsymbol{x}) = 0$, and $\lim_{(\boldsymbol{w}^\top \boldsymbol{x} \to \infty)} p_{\boldsymbol{w}}(\boldsymbol{x}) = 1$.

Figure 2.3 shows 1D examples of the logistic function. The $\boldsymbol{w}$ vector acts as a scale and translation of the function.

The LR model can be fitted to the data by maximizing its likelihood. The probability of an instance $\boldsymbol{x}_i$ of being of class $y_i = 1$ is $p_{\boldsymbol{w}}(\boldsymbol{x}_i)$, and the probability of being

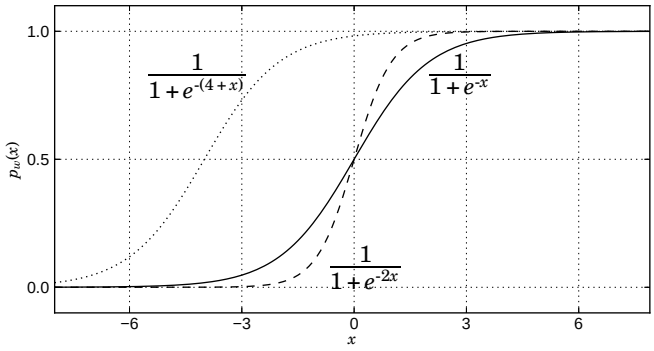


Figure 2.3: Example of logistic function for three different parameter vectors $\boldsymbol{w}$.

$y_i = 0$ is $1 - p_{\boldsymbol{w}}(\boldsymbol{x}_i)$. Then, we define the LR model likelihood of the data $\mathcal{D}$ and the model defined by $\boldsymbol{w}$ as

$$\mathrm{P}(\mathcal{D}|\boldsymbol{w}) = \prod_i p_{\boldsymbol{w}}(\boldsymbol{x}_i)^{y_i}[1 - p_{\boldsymbol{w}}(\boldsymbol{x}_i)]^{1-y_i}. \tag{2.13}$$

In order to get the optimal parameters, we optimize the log-likelihood. When calculating the derivatives of the log-likelihood with respect to the LR parameters, we get a transcendental equation with no closed-form solution. This problem is solved numerically using optimization methods such as Newton-Raphson.

The logistic regression model can be extended to multi-class classification problems with K labels by using a set of $K - 1$ independent binary regressions. This is called *multinomial logistic regression*. Details and properties of this algorithm can be found in Böhning (1992).

Support Vector Machines

The idea behind support vector machines (SVM, Vapnik (1982)) is to divide the attributes space into two by a hyperplane, defined by

$$h(\boldsymbol{x}) \equiv \boldsymbol{b}\boldsymbol{x} + b_0 = 0, \tag{2.14}$$

For each input object, SVM decides its class by finding which side of the hyperplane its attribute vector falls in. In order to develop easier equations, we will define the binary classes as $y \in \{-1, 1\}$. The values for $\boldsymbol{b}$ and b_0 are calculated such that the chosen hyperplane maximizes the distance from it to the closest vectors. We used the *soft margin* modification created by Cortes and Vapnik (1995), which allows training data that cannot be separated without error by introducing a "cost" parameter C.

Parameters $\boldsymbol{b}$ and b_0 are computed by solving the optimization problem

$$\min_{\boldsymbol{b},b_0} \frac{1}{2}||\boldsymbol{b}||^2 + C\sum_i^N \xi_i \tag{2.15}$$

$$\text{subject to } \xi_i \geq 0, y_i(\boldsymbol{x}^\top \boldsymbol{b} + b_0) \leq 1 - \xi_i, \forall i. \tag{2.16}$$

The above procedure uses a hyperplane to classify objects, but usually objects are not linearly separable. SVM can be used as a non-linear classification model by introducing a kernel function

$$K(\boldsymbol{x}, \boldsymbol{x}') = \langle h(\boldsymbol{x}), h(\boldsymbol{x}') \rangle \tag{2.17}$$

which computes the inner products in a transformed space defined by $h(\boldsymbol{x})$. It is not necessary to directly know the transformation $h(\boldsymbol{x})$ as long as we know the kernel function. Some examples of popular Kernel functions are:

linear: $K(\boldsymbol{x}, \boldsymbol{x}') = \langle \boldsymbol{x}, \boldsymbol{x}' \rangle$

d**th degree polynomial:** $K(\boldsymbol{x}, \boldsymbol{x}') = (1 + \langle \boldsymbol{x}, \boldsymbol{x}' \rangle)^d$

radial basis function: $K(\boldsymbol{x}, \boldsymbol{x}') = \exp(-\gamma \langle \boldsymbol{x}, \boldsymbol{x}' \rangle)$

sigmoid: $K(\boldsymbol{x}, \boldsymbol{x}') = \tanh(\gamma \langle \boldsymbol{x}, \boldsymbol{x}' \rangle + \rho)$

Using one of these kernels we can define our classification surface as

$$h(\boldsymbol{x}) = \sum_i^{N_{\text{SV}}} a_i y_i K(\boldsymbol{x}_i, \boldsymbol{x}) + b_0, \tag{2.18}$$

where i runs over the N_{sv} *support vectors*, which are the input vectors for which $a_i \neq 0$. A detalied explanation on how these parameters are obtained can be found in Appendix B. The SVM algorithm obtains for a training set the values for a_i and b_0, but the values for C, γ, and ρ must be defined a priori. In these thesis, these parameters were chosen by griding on them and evaluating our SVM in terms of its accuracy as defined below.

2.5.2 Evaluating a Classification Model

Consider a classification model and a labeled data-set for evaluating such model. We can classify these instances through the model and compare the obtained labels against the original labels. We would expect a perfect model to obtain exactly the same labels as the original ones, but this hardly happens. Usually some of the labels are missed by the model, and it is important to be able to measure how good the model is.

A classification model $f(\boldsymbol{x})$ can be evaluated in different ways. Consider a set of N instances and their ground truth labels $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{N}$ to be used for assessing how good our model is. We define the number of samples classified as k_1 given their ground truth label is k_2 as $N(k_1|k_2)$

$$N(k_1|k_2) = \frac{1}{N}\sum_{i=1}^{N} \mathbb{1}(f(\boldsymbol{x}_i) = k_1)\mathbb{1}(y_i = k_2), \tag{2.19}$$

where $\mathbb{1}(\cdot)$ is the indicator function

$$\mathbb{1}(s) = \begin{cases} 1 & \text{if } s \text{ is true} \\ 0 & \text{if } s \text{ is false} \end{cases} \tag{2.20}$$

Using $N(k_1|k_2)$ we can create a confusion matrix, which we can use to asses which classes are correctly predicted and which ones are confused.

Classification accuracy is a measure of how good a classification model is. The accuracy is defined as the number of correct predictions over the total number of predictions:

$$\text{Accuracy} = \frac{1}{N}\sum_{k=1}^{K} N(k|k), \tag{2.21}$$

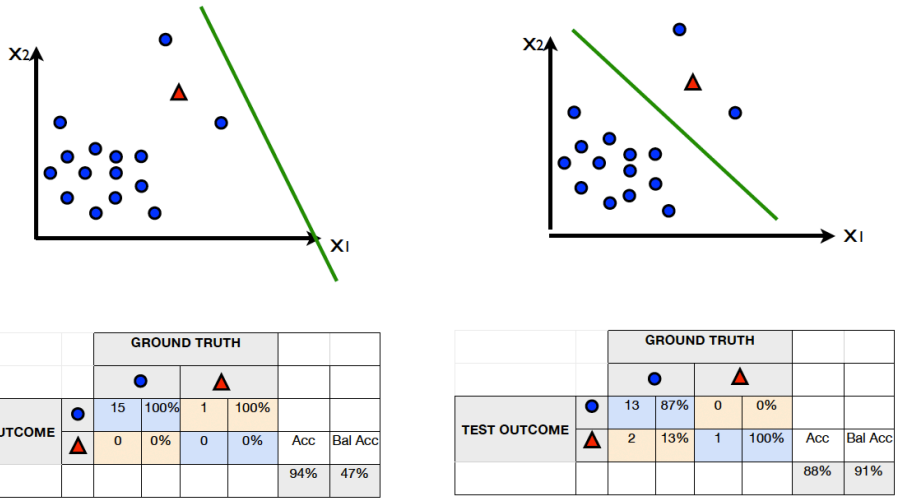
where K is the number of classes. The error rate can be estimated in a similar way by considering the misclassified objects:

$$\text{Error} = 1 - \text{Accuracy} = \frac{1}{N}\sum_{k_1=1}^{K}\sum_{k_2 \neq k_1} N(k_1|k_2). \tag{2.22}$$

Note that the accuracy, as described in Eq. 2.21, assumes that our data-set follows the true distribution of objects, i.e. if more objects of one class are present in our sample, their misclassification will be more important when measuring the accuracy. If our data-set is not balanced (i.e. there are many more objects of one class), the previously described accuracy, could miss all the minor class objects and still evaluate the model as a good one. For this case of unbalanced data-sets, a better accuracy measurement is the *balanced accuracy*, which is the mean of the individual accuracy for each class:

$$\text{balanced accuracy} = \frac{1}{K}\sum_{k=1}^{K} f(k|k), \tag{2.23}$$

where $f(k_1|k_2') = \frac{N(k_1|k_2)}{N(k_2)}$ is the fraction of objects of class k_2 classified as k_1. Figure 2.4 shows an unbalanced scenario and two classification models comparing their confusion matrices, un-balanced accuracy, and balanced accuracy.



		GROUND TRUTH					
		●		▲			
TEST OUTCOME	●	15	100%	1	100%		
	▲	0	0%	0	0%	Acc	Bal Acc
						94%	47%

		GROUND TRUTH					
		●		▲			
TEST OUTCOME	●	13	87%	0	0%		
	▲	2	13%	1	100%	Acc	Bal Acc
						88%	91%

Figure 2.4: Confusion matrix, accuracy and balanced accuracy for a 2D example. Blue circles and red triangles represent two classes (binary classification) in a particularly un-balanced data-set. The green line represents a linear classification model. Left: all instances are classified as circles by the model, obtaining a 94% accuracy, but only a 47% balanced accuracy. Right: 13 circles and 1 triangle are correctly classified, while two circles are incorrectly labeled as triangles, obtaining a 88% accuracy and a 91% balanced accuracy.

Binary Classification

For the case of binary classification ($y_i \in \{0, 1\}$) other evaluation metrics can be defined. We will consider $y = 1$ as *positive* and $y = 0$ as *negative*. We define the number of true positives (TP) $= N(1|1)$, the number of true negatives (TN) $= N(0|0)$, the number of false positives (FP) $= N(1|0)$, and the number of false negatives (FN) $= N(0|1)$. Table 2.1 shows different metrics for evaluating binary classification models.

Care has to be taken when using different metrics. Usually, accuracy or balanced accuracy is used for evaluating a model. For example, sometimes we may wish to

True Positive Rate (sensitivity or recall)	TP / (TP + FN)
True Negative Rate (specificity)	TN / (FP + TN)
False Positive Rate (fall-out)	FP / (FP + TN)
False Negative Rate	FN / (TP + FN)
Positive Predictive Value (precision)	TP / (TP + FP)
Negative Predictive Value	TN / (TN + FN)

Table 2.1: Different metrics for evaluation of binary classification models.

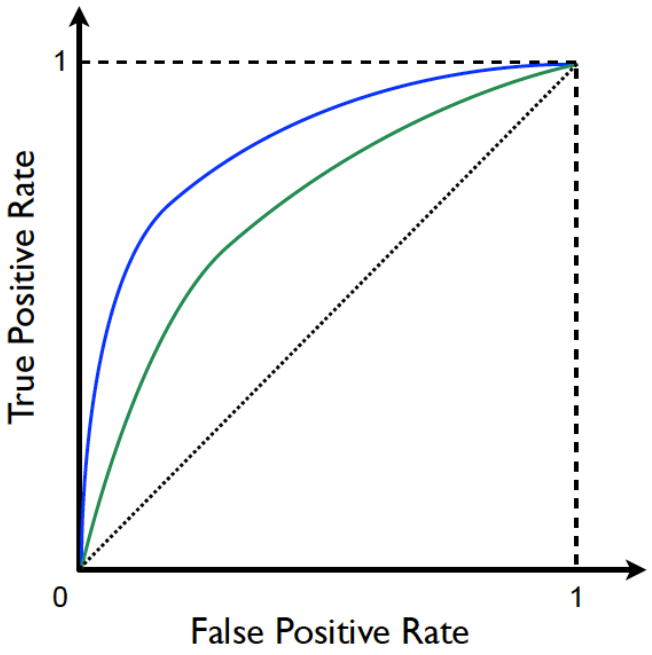


Figure 2.5: Receiver Operating Characteristic curve. Blue and green lines represent two learning models. The blue model performs better than the green model, achieving a higher area under the curve.

have a fixed number of false positives as long as we maintain a certain level of false negatives. Furthermore, errors can be controlled when the classification model allows for a sensitivity parameter that enables us to control the true positive rate (TPR) and false positive rate (FPR). For example, when having a probabilistic model for which we get a probability of an instance to be 1, we can use different probability thresholds to obtain a certain true positive rate instead of 0.5. Of course, there is a trade-off between the TPR and the FPR. This is taken into account by the so called *Receiver Operating Characteristic* (ROC) curve. The ROC curve is created by calculating the TPR and FPR for different values of this control parameter. Figure 2.5 shows an example of two ROC curves for different classification models. The closest the curve gets to a TPR of 1 and a FPR of 0, the best our model is. In particular, when the TPR = 1, and the FPR = 0, we get an area under the curve (AUC) of 1. This way, we get a new metric for evaluating a model that considers sensitivity and fall-out: the closer the AUC is to 1, the better our model is.

2.5.3 Cross-validation

In order to assess the validity of the model (i.e. how the model will generalize to new data) a *cross-validation* technique is used. Cross-validation is a way to limit problems

such as model over-fitting.

Cross validation consists of dividing the labeled data-set into two: a *training set* and a *test set*. The model is fitted to the training set, and then tested against the test set. Furthermore, statistics (such as mean and standard deviation) on a model can be calculated by using different test+train cross validation sets. Different approaches exist for defining these data-sets, which we discuss next.

k-fold cross-validation

k-fold cross-validation consists in splitting the training data-set into k sub-sets. Each of these sub-sets is once retained for testing while the $k-1$ remaining are used for training. The obtained classification model is then evaluated using the retained sub-set. This procedure is repeated using all the k sub-sets, so each of them is used once for testing and $k-1$ times for training. The model validity can be statistically evaluated by calculating the mean and standard deviations of the desired metrics calculated over all these training/test combinations.

A usual practice is to split the training set into two, which would be a 2-fold cross-validation, also called holdout method. On the other hand, when $k = N$ the procedure is also called leave-one-out cross-validation.

Repeated random subsampling

Repeated random subsampling or shuffle split consists in randomly selecting a sub-set for training and a sub-set for testing. This procedure can be repeated as many times as desired, and the size of the training and test sets can be chosen. Again, statistics can be calculated over all the training/test sets.

The advantage of this method over k-fold cross-validation is that the proportion of training/test objects is not dependent on the number of folds. On the other hand, its disadvantage is that it may happen that not all objects are considered, and some of them will be considered more than once.

Stratified cross-validation

Stratification consist in selecting train and test sets that contain the same distribution of objects of each class than the whole data-set. This can be applied to both of the methods described above. In stratified k-fold cross-validation each of the k sub-sets contains the same fraction of objects as the original data-set, while in stratified shuffle split this happens for each randomly selected sub-set.

Chapter 3

Related Work and Contribution

3.1 Previous Work

Semi-automated detection and photometry over astronomical images was born when old photographic plates began to be scanned. Computer programs where created then to analyze these digital data. Current existing methods need some input from the user in order to work. This is specially true for detection and extraction, where sizes, shapes and thresholds are needed to be used (among others) for current algorithms to work. We aim to develop automated methods for extraction and classification of astronomical sources.

3.1.1 Stellar Detection and Photometry

Many reduction techniques were created during the late '70s. Most of these techniques focused on photometry, and detection of stars, and worked in a semi-automatic way, where the interaction with the user was fundamental. The Kitt Peak National Observatory's Interactive Picture Processing System (Wells, 1975) was one of the first approaches for analyzing astronomical images in a semi-automatic way. The system had a monitor where users could select objects and obtain luminosity profiles. Butcher (1977) used this system for photometry of stars from the large Magellanic cloud. Chiu (1976) developed a method for obtaining positions, and Herzog and Illingworth (1977) one for photometry. Green and Morrill (1978) based their work on them creating a method for positions, color index and magnitudes (photometry). Their detection was based on local calculation of the background and thresholding.

The charge-coupled device (CCD) was first conceived in 1970 by Boyle and Smith (1970), and the first CCD image array was produced commercially in 1974 by Fairchild

Electronics, and consisted of 100×100 pixels. The first application of CCDs to astronomy was a 320×512 pixels camera installed on a 1-meter telescope at Kitt Peak National Observatory. This is when the first software packages for automated detection and photometry of stars were created. The most famous of these are RICHFELD (Tody, 1981), ROMAFOT (Buonanno et al., 1983), WOLF (Lupton and Gunn, 1986), STARMAN (Penny and Dickens, 1986) and DAOPHOT (Stetson, 1987). The last one is still frequently used by astronomers. DAOPHOT is mainly focused on stars rather than galaxies. Its detection method uses the PSF of the instrument, tries to fit a star to each pixel in the image, and finds the pixels which are more likely to be stars.

3.1.2 Galaxy Cataloging

At the same time the first stellar detection algorithms were created, the community that was dedicated to detection and classification of galaxies (cataloging) created their own methods. The first works on this area are from Pratt et al. (1975), Kibblewhite et al. (1975), Oemler (1976), Herzog and Illingworth (1977), Benedict and Shelus (1978), and Lorre et al. (1979). However, these methods were created for analyzing specific data, and none of them where meant to work on faint and bright images at the same time.

The first software package created for cataloging different data-sets was FOCAS (Jarvis and Tyson, 1981). Their detection algorithm uses thresholding over the average of a 5×5 window. The APM software (Maddox et al., 1990) and the COSMOS system (Beard et al., 1990) were created later, followed by the PPP package (Yee, 1991). Bertin and Arnouts (1996) created Source Extractor (SE), one of the most famous software packages which is still widely used by astronomers.

SE is meant to optimally detect, deblend, measure and classify sources from astronomical images. The detection of objects uses thresholding: the user specifies a threshold and all pixels over that threshold are selected. At the same time, from this selection, 8-connected contiguous pixels are detected. If there are more than a certain number of contiguous pixels (defined by the user), an object is detected.

Another approach for astronomical image processing is detecting over a wavelet space-frequency decomposition of the image. Bijaoui and Rué (1995) described the Multiscale Vision Model (MVM) idea basing their work on Morlet-Grossman's definition for a 1D wavelet function (Morlet et al., 1982), and the multifrequency channel decomposition by Mallat (1989). The basic idea is to use wavelets to decompose the image in different scales. At each scale, the wavelet coefficients that exceed a certain threshold are detected as structures. Then, sets of connected structures at different scales are defined as objects. Starck and Murtagh (2006) created a software package called *multiresolution* which includes the MVM.

3.1.3 Galaxy Morphologies

Galaxies can be classified according to their morphology. As mentioned above, galaxy shapes have a strong correlation with their global properties such as color, brightness, maximum rotation velocity, and gas content. From these properties, it is possible to obtain important physical properties such as stellar population fraction, surface star density, total mass, and gas to star conversion rates (see Odewahn et al., 2002, and references therein).

The most famous galaxy classification scheme is the one proposed by Hubble (1926) which was later expanded by de Vaucouleurs, Gerard (1959).

Hubble Sequence: Hubble sequence distinguishes between elliptical, spiral, and barred spiral galaxies by assigning the letters E, S, and SB, respectively (see Fig. 3.1). It also considers lenticular (S0) and irregular galaxies (Irr).

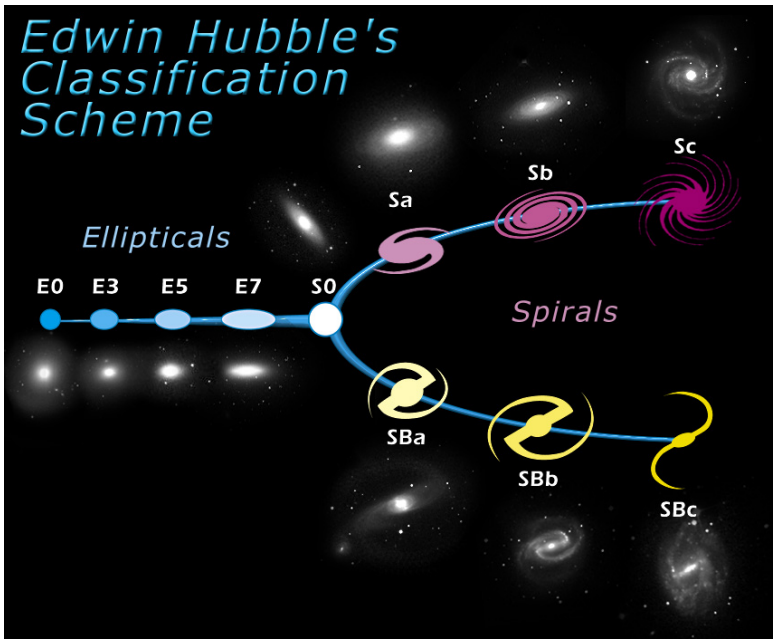


Figure 3.1: Hubble sequence. Image obtained from www.wikipedia.org.

Elliptical galaxies have also assigned a number corresponding to their ellipticities. For example an E7 is an elliptical galaxy with an ellipticity of 0.7.

Spirals, consist of a flat disk and, usually, a strong intensity bulge at the center. The disk contains stars that form spiral arms. Letters a, b, and c are assigned to each spiral galaxy depending on how tight its arms are. Spiral galaxies with very tight arms belong to class Sa, while the ones with loose arms belong to class Sc.

Barred spiral galaxies are similar to spirals, but they have a bar at the center. Its classification follows the one from spiral galaxies.

De Vaucouleurs System: The de Vaucouleurs system (or extended Hubble sequence) is an expansion of the Hubble sequence (see Fig. 3.2). It maintains the separation between elliptical, spirals, barred and irregular galaxies, but it also considers an intermediate class between spirals (S) an barred (SB) for weakly barred spirals (SAB). Their spiral arms tightness is still classified by letters a, b, and c, but a letter d is added for diffuse arms, and a letter m for irregular spiral galaxies with no bulge component.

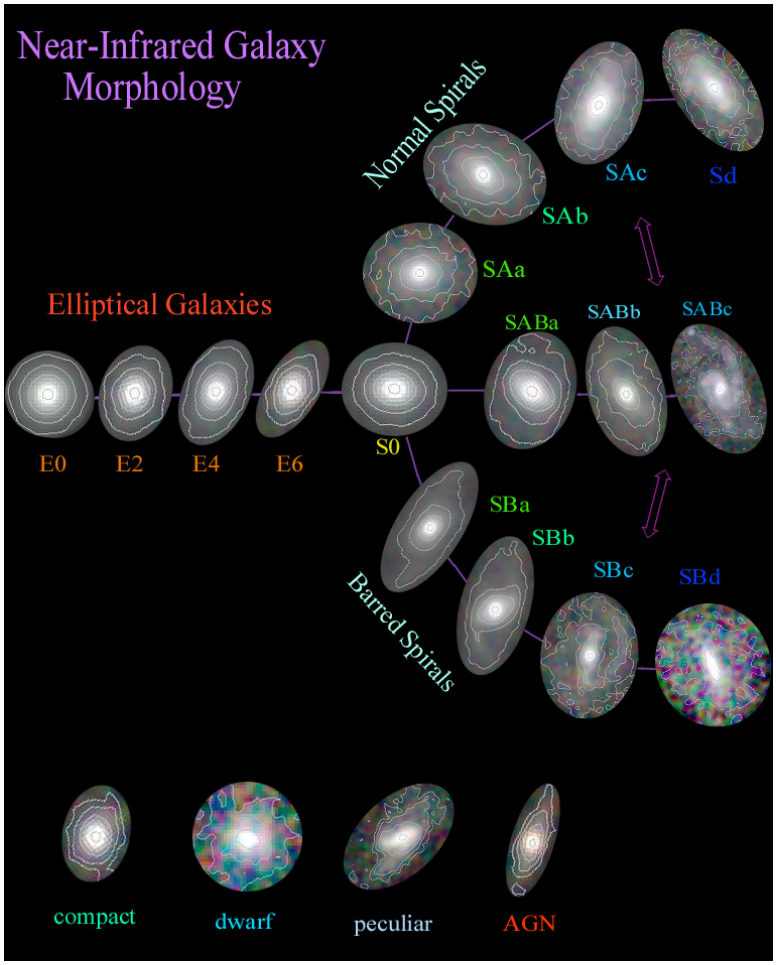


Figure 3.2: de Vaucouleurs system. Image obtained from Jarrett (2000). Images come from the Two Micron All-Sky Survey (Skrutskie et al., 2006), and RGB colors are assigned from K, H, and J filters respectively. Contour lines correspond to the 10%, 20%, 30%,$\cdots$, etc., of the J-band peak flux.

Two new kinds of lenses are added: unbarred (SA0) or barred (SB0). S0 is reserved

for lenses where is not possible to know if a bar exists. The de Vaucouleurs system also considers rings (r), no rings (s), and "transition" galaxies (rs) for faint rings. Finally, irregular galaxies are classified as Im.

3.1.4 Supervised Classification

Supervised classification algorithms have been studied and developed for decades. The basic goal is to classify a set of records (each one containing a set of attributes) into different classes. Classification models are usually trained with a training set and a test set, both for which we know their record classes. The training set is used to train the classification model, which is later tested with the attributes from the test set. Tan et al. (2005) have outlined the most popular algorithm used in the field. These algorithm include decision trees, nearest-neighbor classifiers, Bayesian classifiers, artificial neural networks (ANN), logistic regression (LR), Gaussian process (GP), and support vector machines (SVM).

Decision trees are classification techniques that use tree-like structures to decide which class a record belongs to. Several decision trees induction algorithms have been surveyed by Moret (1982), Buntine (1992), and Murthy (1998). The major difference between algorithms is the way the tree is constructed, depending on the available attributes. Some examples on different decision trees algorithms are CHAID (Kass, 1980), CART (Breiman et al., 1984), ID3 (Quinlan, 1986), and C4.5 (Quinlan, 1992). Decision trees can be used to form ensembles which are very powerful models created by a set of weak predictors. Such is the case of *random forests* (Breiman, 2001) and *boosted trees* (Friedman, 2002).

The basic idea of nearest-neighbor classifiers (Cover and Hart, 1967) is to search for the closest known-class objects to the unknown object in the attributes space. The algorithm uses the classes of these nearest neighbors to determine the class label of the unknown record. Examples of ways for determining the class label include majority vote and weighted vote according to distance.

Bayesian classifiers use Bayes theorem to determine the probability of an object to belong to each class. The class with the highest probability is associated to that object. The naive Bayes classifier assumes independence among the attributes. This way, the probability of the object to be of a certain class can easily be calculated by counting number of occurrences or assuming a certain probability distribution (such as a normal distribution).

Artificial neural networks (ANN) have also been vastly used for classification. They are based on the neurological system, and their main idea is to model the problem through a set of artificial neurons, interconnected between each other. McCulloch and

Pitts (1943) studied how neurons work and proposed a simple mathematical model which they implemented using electric circuits. This idea was further developed by Rosenblatt (1958) who proposed the *perceptron* which was the first artificial neural network implemented. An ANN consists of a set of nodes (artificial neurons) which receive a set of input values to be evaluated by an activation function. The activation function of a neuron defines the output of that node given the set of input values. This output may be connected to another node or may define a class. There are different ANN models, that use different activation functions and can be trained through different algorithms which have been reviewed by Zhang (2000). They have been widely used to solve problems such as bankruptcy prediction, handwriting recognition, speech recognition, product inspection, fault detection, and medical diagnosis, between others (see Zhang (2000) and references therein).

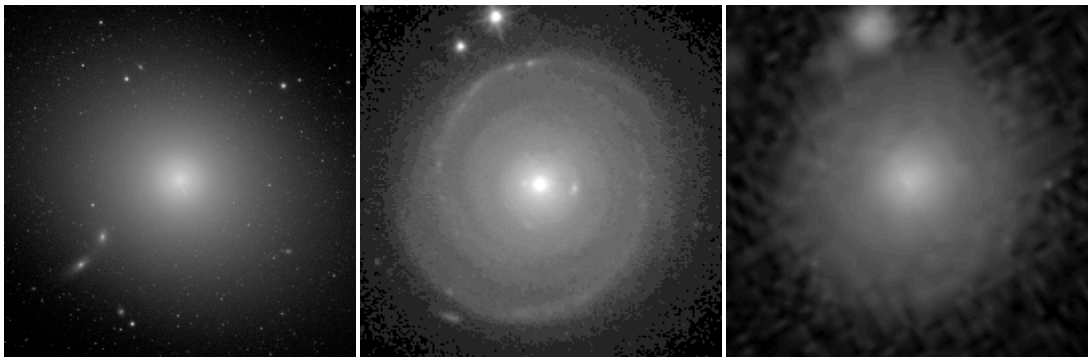
As explained in the previous chapter, logistic regression (McCullagh, 1980; Lemeshow and Hosmer, 1982; Agresti, 1990) is a binary linear classification model that fits a logistic function to calculate the probability of an object to belong to a class. This logistic function is monotonic and works as a linear decision boundary. Classes are defined by choosing a probability threshold, which defines this boundary.

Vapnik (1982) started developing in 1965 the support-vector machine (SVM). The standard SVM is a non-probabilistic binary linear classifier, which divides the attributes space into two by a hyperplane. For each input object, it decides its class by finding which side of the hyperplane its attribute vector falls in. There might be many hyperplanes that separate the data into two classes. The chosen hyperplane is generally the one that maximizes the distance from it to the nearest data points. The first version of SVM only worked on data that could be split by a hyperplane. Cortes and Vapnik (1995) created a *soft margin* modification to SVM which allows training data that cannot be separated without error. Multiclass SVM algorithms have also been developed and tested empirically (see Hsu and Lin (2002), Duan and Keerthi (2005), and references therein).

A Gaussian process (GP) on a data-set is a set of random variables associated to each of the points in the data-set, such that these random variables follow a normal distribution. In the field of machine learning, GP have been used primarily for regression problems (see Rasmussen and Williams, 2005). GP regression can be extended to probabilistic classification by doing regression on the probabilities. As GP is a probabilistic model, it gives a probability distribution for the regressed value of the unseen data, for which it is possible to have confidence intervals.

3.1.5 Labeling Bias

The labeling bias we discuss here is not the same as measurement bias. Millsap and Everson (1993) have reviewed statistical methods for detecting measurement bias in psychological and educational tests. They define this bias as a systematic inaccuracy of measurement. In our case, the measurement by the annotator is "correct". Figure 3.3 shows an example of the biasing problem we are addressing. Figure 3.3c looks elliptical and no spiral arms are present, so it must be classified as an elliptical galaxy (it was classified by $> 95\%$ of annotators as being an elliptical). However, the label is still wrong as a result of the quality of the data itself. Figure 3.3b shows higher quality data, which prove that it is a spiral.



(a) (b) (c)

Figure 3.3: Examples of elliptical and spirals galaxies and biased classifications. (a) An elliptical galaxy with good resolution. (b) A spiral galaxy with good resolution taken from low earth orbit. Notice the spiral arms. (c) The same spiral galaxy as (b), except at worse resolution and taken from the ground through the Earth's atmosphere. Notice that the arms are no longer discernible.

More closely related to our work is the case when test set data have missing values. For example, Saerens et al. (2001) addressed the problem of class balance by using an Expectation-Maximization algorithm (EM, A. P. Dempster and Rubin, 1977) which updates the test class-prior and class-posterior probabilities from some initial estimates. du Plessis and Sugiyama (2012) reformulated their algorithm, and showed their method has higher robustness to noise and outliers, numerical stability, computational efficiency, and accuracy. While our data have wrong labels, they do not have missing labels.

During the last couple of years a lot of work has been done on fixing crowdsourcing data-sets through EM algorithms. Dawid and Skene (1979) have created a method that deals with bad annotators to obtain a better estimate of the ground truth. Raykar et al. (2009) and Raykar et al. (2010) have extended their approach by learning a classification model at the same time they estimate these hidden labels. Simpson et al.

(2013) have developed a Bayesian approach for fixing these uncertainties in Galaxy Zoo Supernovae data. As noted earlier, our annotators are correct in their measurements, but it is the data itself which is the root cause of the incorrect labels.

Another similar problem is that of noisy labels. This has been addressed from a boosting approach (Oza (2004), Gargiulo and Sansone (2010)), from a data filtering approach (Brodley and Friedl (1999), Zhu et al. (2003)), and from a weighting approach (Rebbapragada and Brodley (2007)). Noisy labels are not exactly the same as the systematic bias in terms of specific parameters (e.g. size of objects in terms of resolution) we are trying to address. Nevertheless, these approaches can certainly be used for de-biasing these labels.

3.1.6 Automated Galaxy Classification

Galaxies have been classified by astronomers using the previous defined classes. The Third Reference Catalogue of Bright Galaxies (RC3) was created by de Vaucouleurs, Gerard et al. (1991) and later corrected in Corwin et al. (1994). RC3 is one of the most complete catalogs of galaxies, containing morphology classes for 73,197 galaxies.

Naim et al. (1997a) have used self-organizing maps (SOMs), an unsupervised learning method described in Kohonen (1990), to visualize and classify galaxies obtained with the Hubble Space Telescope (HST). Classification is achieved considering 4 attributes calculated for each galaxy: blobbiness, isophotal center displacement, isophotal filling factor, and skeleton ratio (see Naim et al., 1997b). Their method organizes these attributes onto a 2D space, enabling to visualize the galaxy distribution.

Odewahn et al. (2002) used Artificial Neural Networks (ANN) to create a classification model that follows the revised Hubble system. They used 146 galaxies from the HST and 124 galaxies from earth telescopes which where separated into training and test sets. Attributes inputted to the model where Fourier coefficients adjusted over elliptical isophotals fitted to the galaxies. Principal components where calculated over amplitudes and phases of the Fourier coefficients for further use by the ANN.

A different approach was taken by Lintott et al. (2008). They created *Galaxy Zoo* (GZ), a crowdsourcing web system which allows users to classify galaxies by visual inspection. Galaxies where obtained from the Sloan Digital Sky Survey (SDSS). During its first year they received 5×10^7 classifications from almost $150,000$ people. GZ classification categories are only reduced to elliptical galaxies, clockwise spiral galaxies, anticlockwise spiral galaxies, other galaxies, stars or "do not know", and mergers.

Recently, SVMs have been used to automatically classify galaxies according to their morphologies (Huertas-Company et al., 2011, 2014). They have used features such as the concentration, assymetry and smoothness, as well as magnitudes, achieving an

accuracy of around 90% over GZ data.

3.1.7 HPC over Astronomical Data

Nowadays, astronomical observatories are inconceivable without the use of a datacenter able to process massive volumes of data in a high performance computing (HPC) environment. ALMA uses a supercomputer specially designed for the correlation (Escoffier et al., 2007), but no HPC algorithms are used for image processing.

The LSST, on the other hand, considers the construction of a datacenter with 100-400 TFlops computing power (Ivezic et al., 2008). Most of this computing power will be consumed by the pipelines from the data management system. These pipelines include algorithms for image processing, detection, association, moving object detection, classification, and calibration.

The Dark Energy Survey (DES) (TheDarkEnergySurveyCollaboration, 2005) will produce about 1 TB of raw data per night. The DES data management system (DESDMS) (Mohr et al., 2008; Sevilla,I. et al., 2011) framework integrates data archiving, data processing, and data access. The DESDM is being developed for working in HPC environments, and includes algorithms for image detrending (remove instrumental signatures), astrometric calibration, photometric calibration, coaddition (combination of several images in order to reduce the signal to noise ratio), and cataloging. Most of these routines use SE, specially for the detection and cataloging.

The Square Kilometer Array (SKA) (Dewdney et al., 2009) is a major interferometer which is currently under design. It will consist of thousands of antennas distributed over an area of 3,000 Km^2 in extent. This major facility poses technical challenges in algorithm design including calibration, image reconstruction, and non-imaging processing, such as searching for pulsars and its timing analysis. Wicenec et al. (2011) have outlined the SKA challenges in the areas of data flow, storage design and optimization, database integration into HPC, and low-latency scheduling for ultrascale visualization and analysis on HPC systems. Their goal is to provide the astronomers with a flexible HPC framework for working with multiple TB data-sets.

3.2 Contribution

As described above, in this thesis we address the problems of detection, extraction and classification of astronomical objects. We will divide our contributions into two: extraction and classification.

3.2.1 Extraction

Most detection algorithms use a threshold on the intensity of the images to select pixels as sources or background. This threshold is usually defined in terms of the amount of noise in the images, and discards data that could be used for accurately extracting the light profile of sources. We propose a different approach, in which we reduce the noise level of the images by applying a convolution kernel and then detect local maxima in the images. As the original signal is already convolved with the PSF, all high frequencies are already lost. In that sense, we convolve the CCD image with the same PSF in order to eliminate the high frequency noise while keeping most of the PSF convolved frequencies from the astronomical sources.

In order to extract the light for each object we calculate elliptical isophotes and fit PSF-convolved models to these isophotes over the semi-major axis. In that sense, we reduce the problem from a 2D (pixel image) fit to a 1D fit in term of these isophotes. We do this by fitting the so called single-component Sérsic model, which is typically used to characterize galaxy light distribution. The isophotes are a smooth representation of each object, so it is suitable for smooth light profiles such as the Sérsic model.

When fitting the Sérsic profiles to the 1D isophote intensities, the PSF convolution has to be taken into account. In Trujillo et al. (2001a) and Trujillo et al. (2001b) the effect of the PSF convolution over the Sérsic profile is described, including analytical expressions for it over the semi-major axis. These analytical expressions are numerically complicated to compute, as they include hyper-geometric functions and sums to infinity. In order to sort this problem, we created 2D simulations and obtained the convolved radial profile across the semi-major axis. We obtain reduced χ^2 values of 1.01 ± 0.01, but huge errors over the Sérsic parameters, from which we conclude the PSF-convolved Sérsic model is degenerate. This is consistent with the state of the art results obtained from 2D fits. We show that the complexity of our algorithm is 10 times lower than performing a 2D fit.

3.2.2 Classification

Classical supervised learning assumes that there are known ground truth labels which can be used to train a model. This is not the case of astronomical sources classification: as explained in Chapter 2, light from sources gets convolved with the PSF, and noise is added. This produces biases in labels created by humans by observing images.

In this thesis we extensively address this problem. We created two metrics for assessing this bias. The first is a basic approach that assumes the fraction of ground truth (latent) labels to be uniformly distributed in terms of the observable parameters that bias the observed labels. Then, we extend this approach to the case where the

fraction of objects depend on other intrinsic parameters. To the best of our knowledge this is the first time this bias has been measured. We validate our metric through simulations, and measure the bias of different galaxy morphology data-sets.

We also test how the Sérsic radial profiles of galaxies fit their morphologies and how this can be used to set a limit on when we can trust an object to be a galaxy in terms of its resolution. Furthermore, by using an SVM classification model we create an analytical hypersurface that can be used to simply classify elliptical and spiral galaxies in terms of their Sérsic profiles. This hypersurface is easy to use to classify new Sérsic profile fits, which are abundant in the literature. This is a much simpler way of classifying objects than the models available in the field (which either use complicated classification models or complicated features to asses good results), so almost no expertise in machine learning is needed in order to use it. We check the amount of bias of this new SVM model using the Sérsic parameters as features and show that it is lower than the bias of previous labels (human or machine learned) available in the literature.

Finally, we propose a mixture probabilistic model that simultaneously fits a classification model and a parametric bias to the data-set. This model fits an estimate for the ground truth latent variables and parameters included in the models through an Expectation-Maximization approach. We show our method performs much better over simulations than weighted models in the literature, which are usually used to remove noisy labels.

Chapter 4

Extraction through Parametric Model Fitting

4.1 Introduction

Astronomical images obtained from charged-couple devices (CCD) consist of the light coming from astronomical objects I convolved with the instrument point spread function (PSF) plus a background intensity b and the CCD noise ε. The light obtained can be modeled as:

$$D_{i,j} = (\mathrm{PSF} * I)_{i,j} + b_{i,j} + \varepsilon_{i,j}, \tag{4.1}$$

where $D_{i,j}$ is the data obtained at pixel (i, j). The PSF is the response of the instrument to a point source. As light comes into the atmosphere and into the telescope some aberration is produced. Point sources used in astronomy for calculating the PSF are stars. Knowing the position of a particular star, it is possible to calculate the PSF profile by measuring the light profile of the image at its location.

Consider a set of N_{obj} objects from which the light of the astronomical image comes from. Eq. 4.1 can be written in terms of this objects as

$$D_{i,j} = (PSF * \sum_k I_k)_{i,j} + b_{i,j} + \varepsilon_{i,j}, \tag{4.2}$$

$$= \sum_k (PSF * I_k)_{i,j} + b_{i,j} + \varepsilon_{i,j}. \tag{4.3}$$

where I_k is the intensity of object k.

In this Chapter, we present a new method for detecting and characterizing the light from astronomical objects based on a parametric model called the *Sérsic profile*. Our algorithm detects objects candidates by looking for local maxima over a noise-reduced

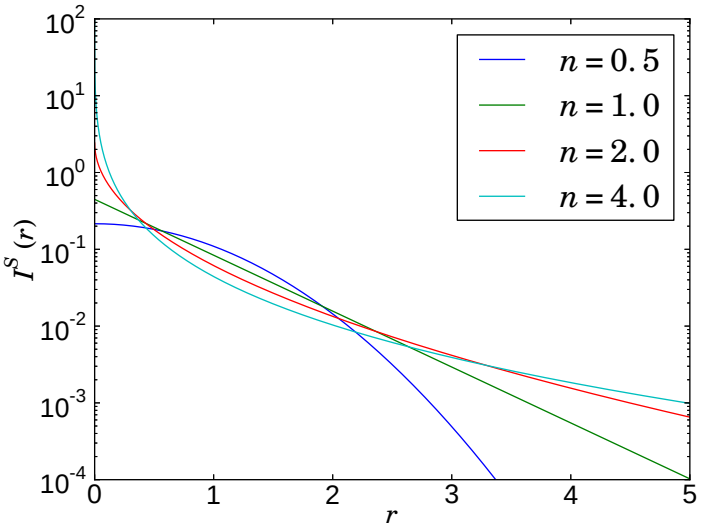


Figure 4.1: Examples of Sérsic profiles for $R_e = 1$ and different n.

image. Then, it obtains a radial profile by calculating elliptical contour lines and fits the Sérsic parameters to such profile.

4.2 The Sérsic Light Profile

To extract their light, we modeled galaxies using a radial parametric model called the *Sérsic profile* (Sersic, 1968). A Sérsic profile is defined as

$$I^{\mathrm{S}}(r) = I_0 e^{-b_n(\frac{r}{R_e})^{1/n}}, \tag{4.4}$$

where I_0 is the intensity at the center of the object, n is called the Sérsic index, and R_e is the effective radius defined as the radius inside which the integrated luminosity equals half the total luminosity of the object. The value of b_n helps defining R_e and it depends on n. We used the approximation used by Ciotti (1991) $b_n = 2 \times n - 0.324$, obtained from a linear interpolation over exact values of b_n.

Figure 4.1 shows radial cuts of Sérsic profiles for different values of n and R_e. All these profiles have the same 2D integrated luminosity of 1. Notice that the bigger the Sérsic index n, the steeper the central light profile, and larger the outer disk.

Early on, $n = 1$ light profiles were associated with the exponential disks of spiral galaxies while $n = 4$ profiles were associated with the DeVaucoulour characterization of ellipticals (see Trujillo et al. (2001a) and references therein). Of course the Universe is not so simple, and numerous examples of spiral galaxies with $n = 4$ Sérsic profiles

exist, likewise for $n = 1$ elliptical galaxies. Therefore, the Sérsic profile has often been described as fitting the structure of the light profile or light structure, but not necessarily the morphology of the object. In Chapter 6 we will assess how well the Sérsic profile recovers morphologies of galaxies.

Due to their shape and rotation angle, most of the galaxies never look completely circular. Elliptical coordinates are very useful for taking into account the ellipticity of the profiles. These coordinates are defined as

$$x = \xi \cos E, \tag{4.5}$$

$$y = \xi(1-\epsilon)\sin E, \tag{4.6}$$

$$\xi = \sqrt{x^2 + \frac{y^2}{(1-\epsilon)^2}}, \tag{4.7}$$

where (x, y) are Cartesian coordinates, ξ is an elliptical radius, E is the eccentric anomaly, and ϵ is called the ellipticity (not to be confused with the eccentricity). Using this elliptical coordinate system described by (ξ, E), the intensity of an elliptical source depends only on ξ, so an elliptical Sérsic profile will be defined as $I^{\mathrm{S}}(\xi)$.

Sérsic profiles model the real light distribution of galaxies before convolution with the PSF. Trujillo et al. (2001a) have analyzed the effects of a PSF Gaussian convolution over the Sérsic profile. In particular, they gave an analytical expression for the brightness distribution over the major axis of the object as

$$\begin{aligned} I^{\mathrm{C}}(\xi, 0) &= (I^{\mathrm{S}} * \mathrm{PSF})(\xi, 0) \\ &= \frac{I_0}{\sqrt{\pi}}(1-\epsilon)e^{-\frac{1}{2}(\xi/\sigma)^2} \sum_{k=0}^{\infty} \frac{(-)^k}{k!} b_n^k \left(\frac{\sqrt{2}\sigma}{R_e}\right)^{\frac{k}{n}} \\ &\quad \times \sum_{l=0}^{\infty} \frac{(2\epsilon - \epsilon^2)^l}{l!} \frac{\Gamma(l+\frac{1}{2})}{\Gamma(l+1)} \Gamma\left(l+1+\frac{k}{2n}\right) \\ &\quad \times M\left(l+1+\frac{k}{2n}, l+1, \frac{1}{2}\left(\frac{\xi}{\sigma}\right)^2\right), \end{aligned} \tag{4.8}$$

where $I^{\mathrm{C}}(\xi, E)$ is the convolved intensity, ϵ is the ellipticity of the object, σ is the Gaussian PSF standard deviation, $\Gamma(x)$ is the gamma function, and $M(\mu, \nu, z)$ are the confluent hypergeometric functions.

4.3 Extraction Procedure

To detect positions for candidates of objects, before fitting the profiles, we convolved the image D with a Gaussian filter G obtaining a convolved image $C = G * D$. The

standard deviation of G was chosen to be the same one of the PSF, so the filter G is similar to the PSF. We chose the PSF as a low-band noise-reduction filter in order to keep most of the frequencies of the sources from the image (after the convolution with the seeing), as the higher resolution is equal to the PSF.

We obtain all the local maxima on the convolved image C and identify them as possible candidates. These local maxima are obtained by comparing each pixel with its '8-connected' neighbors. If the value of the center pixel is higher than the rest of its neighbors, a local maximum is detected.

Furthermore, this reduced noise image C is also useful for obtaining the central intensity I_0 of Eq. 4.4. The central intensity value I_0 can be directly obtained from R_e, n, and ϵ by evaluating Eq. 4.8 at $\xi = 0$, as described by Trujillo et al. (2001a):

$$I^{\mathrm{C}}(0) = I_0(1-\epsilon)\sum_{k=0}^{\infty}\frac{(-)^k}{k!}\left(\frac{\sqrt{2}\sigma}{r_0}\right)\Gamma\left(1+\frac{k}{2n}\right)\times_2 F_1\left(\frac{1}{2},1+\frac{k}{2n};1;2\epsilon-\epsilon^2\right), \quad (4.9)$$

where ${}_2F_1(a,b;c,z)$ is the hypergeometric function. Instead of using directly the noisy data, we used the central value $I^{\mathrm{C}}(0)$ on the convolved image for this,

$$C = G * D = G * (\mathrm{PSF} * I), \quad (4.10)$$

and the property that convolving the image with two Gaussians of standard deviation σ_1 and σ_2 is equivalent to doing one convolution with a Gaussian of standard deviation $\sigma = \sqrt{\sigma_1^2 + \sigma_2^2}$. In our case, G has the same standard deviation as the PSF, so $\sigma = \sqrt{2} \times \sigma_{\mathrm{PSF}}$, where σ_{PSF} is the standard deviation of the PSF. By convolving the data with the Gaussian filter, we obtain the central convolved value $I^{\mathrm{C}}(0)$ equivalent to convolving the original Sérsic profile of Eq. 4.4 with a Gaussian of standard deviation $\sqrt{2} \times \sigma_{\mathrm{PSF}}$, which allows us to determine I_0 by Eq. 4.9. σ_{PSF} is known "a priori" from the data: it comes from the PSF seeing which can be directly measurable from the data or associated data (e.g., a seeing monitor). This way, the only parameters to be fitted are R_e, n, and ϵ.

Sérsic profiles are iteratively modeled at each local maximum starting from the highest intensity value over the convolved image C. At such position, we calculate elliptical isophotes using a modification of the iterative method described by Young et al. (1979), which is described in section 4.3.1. From our routine, we obtain, for different semi-major axis positions r_m, intensity values $I(r_m)$ and ellipticities $\epsilon(r_m) = 1 - \frac{a(r_m)}{b(r_m)}$, where a is the semi-major axis and b the semi-minor axis. We fitted the Sérsic parameters (R_e, n, and ϵ only. I_0 is obtained from C as described above) to these isophotes using Equation 4.8 by minimizing

$$S = \sum_m (I_e(r_m) - I^{\mathrm{C}}(r_m; R_e, n, \epsilon))^2, \quad (4.11)$$

where $I_e(r_m)$ is the intensity associated to isophote m.

The method works iteratively extracting modeled objects at each iteration. Figure 4.2 shows a diagram of how our method works. Summarizing the steps described above, our method starts by convolving the data with a Gaussian low-pass filter (b). In this image, candidates of objects are selected at local maxima (c). Elliptical level curves are fitted to the candidate with highest central intensity (d) and the convolved Sérsic profile of Eq. 4.8 is fitted to these isophotes (e). Finally, the modeled object is extracted from the image (f) and we repeat the procedure again over this residual image. Figure 4.3 shows an example of each step over a mock image created with the method described in 4.4.1. Notice that the convolution (b) is used only for the detection, but steps (d) to (f) are done over the original (residual from previous iteration) image. Below we explain details for each of these step.

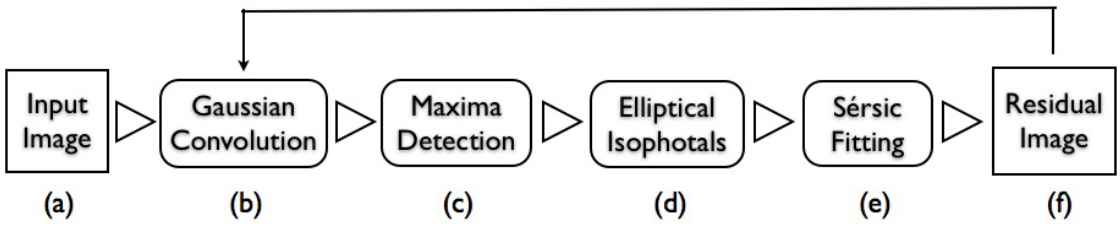


Figure 4.2: Diagram for our method.

4.3.1 Calculating Elliptical Contour Lines

Elliptical isophotes are contours of equal light intensity with elliptical shape. The goal of our elliptical isophote algorithm is to obtain the intensities $I(r_m)$, ellipticities $\epsilon(r_m)$ and rotation angles $\theta(r_m)$ at semi-major axis positions r_m. We used the method described by Kent (1983) (which is similar to the one from Young et al. (1979)), but added an additional step for obtaining initial values for the angles and ellipticities.

The complete procedure consists of calculating an initial approximation for the elliptical parameters (ϵ, θ) at the half-intensity elliptical radius and an iterative procedure for more accurate fitting of a Fourier series, as described in Kent (1983). The initial approximation is calculated at the elliptical radius ξ^* where $I(\xi^*) = \frac{I_0}{2}$ by using a Hough transform. Then, isophotes are fitted iteratively inwards and outwards from that half-intensity isophote. This last step is repeated until getting to the center of the object and then it is increased from ξ^* until the radial gradient relative error $\sigma_{I'}(\xi)/I'(\xi)$ is smaller than a certain value, representing where the object can no longer be identified within the noise, as explained in Busko (1996).

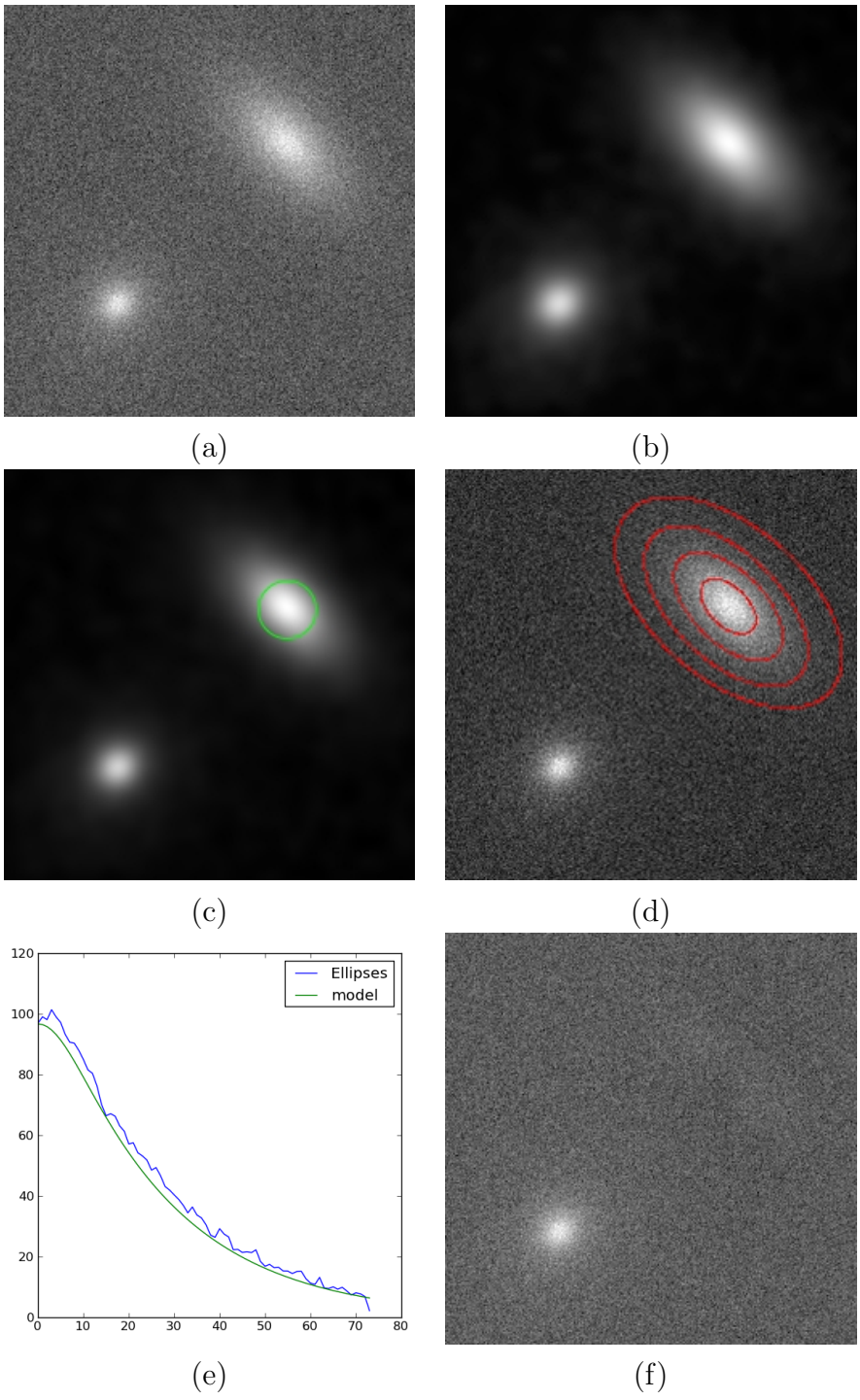


Figure 4.3: Example of our method over a mock image. (a) Original data. (b) Convolution with a low-pass filter. (c) Local maximum detection. (d) Elliptical isophotes fitting. (e) Sérsic profile fitting. (f) Residual image with the modeled profile extracted.

Obtaining Initial Values

For calculating an initial approximation for the ellipticity ϵ and rotation angle θ, we started by calculating a set of points $\{(x_i, y_i)\}$ at $I = \frac{I_0}{2}$ separated by an eccentric anomaly ΔE. This is achieved by searching radially for the first position at which the intensity $I(\xi, E_i) \leq \frac{I_0}{2}$, using a separation distance of 1 pixel. For each of these points, we used the equation of the ellipse

$$(x_i \cos\theta + y_i \sin\theta)^2 + \frac{(x_i \sin\theta - y_i \cos\theta)^2}{(1-\epsilon)^2} = a_0^2, \tag{4.12}$$

where a_0 is the *a priori* semi-major axis calculated as the maximum distance between the center and the points $\{(x_i, y_i)\}$. We used the Hough transform and plotted Eq. 4.12 on coordinates (θ, ϵ) for all (x_i, y_i). The intersection of these curves gives us the estimated value for the ellipticity and rotation angle. Figure 4.4 shows an example of how our algorithm works.

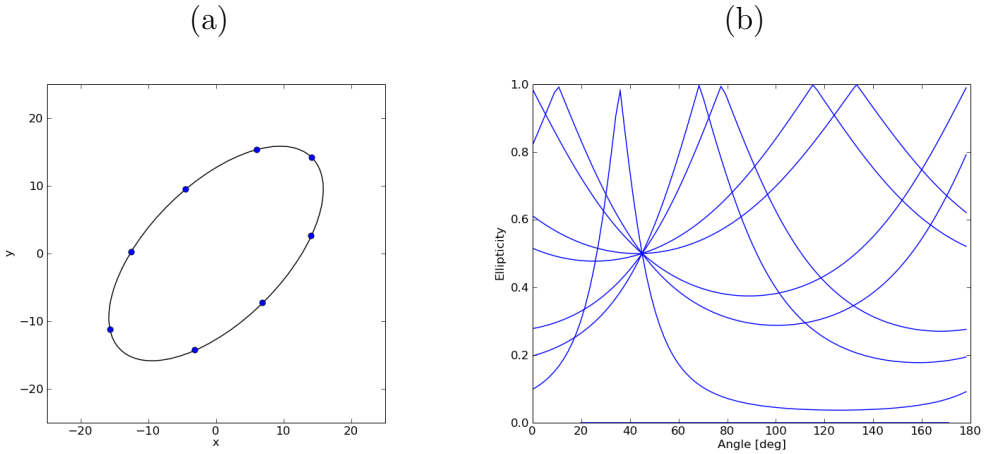


Figure 4.4: Obtaining the angle and ellipticity from a set of points. (a) Set of points distributed on an ellipse in the (x, y) coordinate system. The ellipticity for this particular case is 0.5, an the rotation angle is 45 degrees. (b) The same set of points, but plotted on the (θ, ϵ). The intersection of all curves gives us the ellipticity and rotation angle of the ellipse.

In practice, we created a grid of angles and ellipticities, and added 1 to every cell that each curve intersects. Then, we used a low pass filter, and found the cell with the maximum value, which we associate with the ellipse.

Adjusting Ellipses Parameters

Once obtained an initial approximation for the ellipticity ϵ and the rotation angle θ through the previous described method, it is needed to obtain a more accurate value

for them. This is achieved by describing the ellipse intensity in terms of the elliptical coordinates (ξ, E) (see Eq. 4.7) for an *a priori* ellipse with semi-major axis a_0.

$$\begin{aligned} x &= a_0 \cos E \\ y &= a_0(1-\epsilon)\sin E, \end{aligned} \tag{4.13}$$

This way, using a Taylor expansion and Fourier series, the intensity along the ellipse can be written as

$$\begin{aligned} I(\xi) &= I(a_0) + (\xi - a)I'(a_0) + \cdots \\ &= I_0 + A_1 \cos E + B_1 \sin E + A_2 \cos 2E + B_2 \sin 2E + \cdots, \end{aligned} \tag{4.14}$$

where A_1, B_1, A_2, and B_2 are the Fourier coefficients, which ideally, should be zero. These coefficients are calculated by linear least squares over the set of points around the ellipse. These points are described by their eccentric anomalies E_i. This way, Eq. 4.14 can be written to first order as

$$\Delta I_i = \sum_{m=0}^{3} X_{i,m}\beta_m, \tag{4.15}$$

where

$$\Delta I_i = I(E_i) - I_0, \tag{4.16}$$

$$\boldsymbol{\beta} = [A_1, B_1, A_2, B_2]^{\mathrm{T}}, \tag{4.17}$$

$$\boldsymbol{X} = \begin{bmatrix} \cos(E_1) & \sin(E_1) & \cos(2E_1) & \sin(2E_1) \\ \vdots & \vdots & \vdots & \vdots \\ \cos(E_{N_\mathrm{e}}) & \sin(E_{N_\mathrm{e}}) & \cos(2E_{N_\mathrm{e}}) & \sin(2E_{N_\mathrm{e}}) \end{bmatrix}, \tag{4.18}$$

and N_e is the number of points around the ellipse. Eq. 4.15 can be written in matrix notation as

$$\boldsymbol{\Delta I} = \boldsymbol{X}\boldsymbol{\beta}. \tag{4.19}$$

this system of equations usually has no solution, so the minimum least squares approach solves the quadratic minimization problem

$$\hat{\boldsymbol{\beta}} = \underset{\beta}{\operatorname{argmin}}(||\boldsymbol{\Delta I} - \boldsymbol{X}\boldsymbol{\beta}||^2). \tag{4.20}$$

This problem has a unique solution which can be found by using derivatives, leading us to

$$\hat{\boldsymbol{\beta}} = (\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X})^{-1}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{\Delta I}. \tag{4.21}$$

Once we obtain the values of the Fourier coefficients, we can use them to find the elliptical parameters (θ, ϵ) and the exact center of the ellipses (x_c, y_c). They can be

iteratively computed along the *a priori* ellipse making first order corrections as

$$\Delta x_c = -\frac{A_1}{I'(a_0)} \tag{4.22}$$

$$\Delta y_c = -\frac{B1(1-\epsilon)}{I'(a_0)} \tag{4.23}$$

$$\Delta \epsilon = -\frac{2A_2(1-\epsilon)}{a_0 I'(a_0)} \tag{4.24}$$

$$\Delta \theta = -\frac{2B_2(1-\epsilon)}{a_0 I'(a_0)\left[(1-\epsilon)^2-1\right]} \tag{4.25}$$

At each iteration we modify the elliptical parameter associated to the higher Fourier coefficient. This is done until a convergence criteria is met, based on the value of the highest Fourier coefficient.

4.3.2 Fitting Procedure

One of our goals is to minimize the number of user supplied input parameters. To find the best radial Sérsic model that fitted our isophotes, we did a non linear least squares minimization. A very important issue when using optimization methods is the initial condition. We try to reduce the input parameters to zero by automatically choosing these initial parameters. We started with $\epsilon = \epsilon^{\text{ini}}$ and $R_e = R_e^{\text{ini}}$, where ϵ^{ini} is the ellipticity at the biggest fitted isophotal ellipse and R_e^{ini} is the effective radius calculated over the data. R_e^{ini} is calculated numerically over the elliptical isophotes. This is done by obtaining the integrated brightness over the radial profile and finding the radius at which the integrated brightness is half that value. This leaves just one unknown starting parameter: the Sérsic index n. To solve this issue, we defined a set of initial parameters $n_{\text{ini}} = n_1, ..., n_{N_n}$ and run the least squares minimization N_n times, starting with each of them. We keep the best fit value of all these runs. Ciotti (1991) states that the value of n should be between 0.5 and 10, so we chose values for n_{ini} in that range.

4.3.3 Complexity

As described above and shown in Figure 4.2, our method consists of a Gaussian convolution, the maxima detection, elliptical isophotes fitting, radial Sérsic fitting and calculating the residual image. The overall complexity should then be calculated for each step separately.

Gaussian Convolution The convolution of the data image D with a circular Gaus-

sian filter G can be expressed as:

$$
\begin{aligned}
(D * G)_{i,j} &= \sum_{k,l=0}^{m-1} D_{i-(k-m/2),j-(l-m/2)} G_{k,l}, && (4.26)\\
&= \sum_{k,l=0}^{m-1} D_{i-(k-m/2),j-(l-m/2)} \frac{1}{2\pi\sigma^2} e^{-\frac{(k-m/2)^2+(l-m/2)^2}{2\sigma^2}}, && (4.27)\\
&= \sum_{k,l=0}^{m-1} D_{i-(k-m/2),j-(l-m/2)} \frac{1}{2\pi\sigma^2} e^{-\frac{(k-m/2)^2}{2\sigma^2}} e^{-\frac{(l-m/2)^2}{2\sigma^2}}, && (4.28)\\
&= \sum_{l=0}^{m-1} \left(\sum_{k=0}^{m-1} D_{i-(k-m/2),j-(l-m/2)} G_k^{1\mathrm{D}} \right) G_l^{1\mathrm{D}}, && (4.29)
\end{aligned}
$$

where the data image D has a size of $n_x \times n_y$ the filter G has a size of $m \times m$ ($\frac{m}{2}$ being the mean), a standard deviation of σ, and $G_i^{1\mathrm{D}} = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(i-m/2)^2}{2\sigma^2}}$ is a one dimensional Gaussian filter. It can be seen from Eq. 4.29 that doing a 2D convolution with a circular filter is the same as doing two 1D convolutions on the x axis first and on the y axis after that (or the other way around). Using this approach, the complexity of the convolution would be $\mathcal{O}(n_x \times n_y \times m)$.

There is another improvement for doing this convolution even faster. This is using the fact that the Fourier transform of a convolution of two functions can be represented as the multiplication of the Fourier transform of these functions. In mathematical terms, if $\mathcal{F}$ is the Fourier transform operator, we have

$$\mathcal{F}(D * G) = \mathcal{F}(D) \times \mathcal{F}(G). \tag{4.30}$$

The Fourier transform of an array of size n can be calculated using the Fast Fourier Transform (FFT) (Cooley and Tukey, 1965) in $\mathcal{O}(n \log n)$. Assuming that D and G are 1-dimensional arrays of size n and m respectively, and using the FFT, we can calculate the Fourier transform of D and G in $\mathcal{O}(m \log m + n \log n)$. Multiplication on the Fourier space will take $\mathcal{O}(\max(m, n))$ and the inverse Fourier transform to go back to the real space takes $\mathcal{O}(m \log m + n \log n)$. From these calculations we can conclude that the whole convolution process for the 1D case will take $\mathcal{O}(m \log m + n \log n)$. This is easily extended to our 2D case, giving us an overall complexity for the convolution of $\mathcal{O}((n_x + n_y) m \log m + n_x n_y (\log n_x + \log n_y))$. When working on survey images, the size of the PSF is always much smaller than the image, so we can safely say that this step takes $\mathcal{O}(n_x n_y (\log n_x + \log n_y))$.

Maxima Detection In order to detect the global maxima on the convolved image $C = D * G$, we have to check all pixels. This takes $\mathcal{O}(n_x \times n_y)$.

Elliptical Isophotes The complexity for the computation of the elliptical isophotes has to be calculated for the initial approximation of the elliptical parameters and for the accurate adjustment of them separately (see 4.3.1).

For the initial approximation of ϵ and θ we chose a set of eccentric anomalies E_i separated by ΔE. Thus, the number of angles to be explored are $N_\mathrm{e} = \frac{2\pi}{\Delta E}$. For each of these angles we search radially for the first position at which the intensity $I(\xi, E_i) \leq \frac{I_0}{2}$, using a separation distance of 1 pixel. When running this radial search, the number of pixels to be checked for each angle will never be higher that the object radial size n_obj. This way, the complexity for the initial approximation of the elliptical parameters of the first isophote is $\mathcal{O}(n_\mathrm{obj} \times N_\mathrm{e})$.

For the second step (adjusting ellipses parameters through Fourier coefficients), at each iteration we need to calculate the value of the Fourier parameters by Eq. 4.21. This involves:

- calculating $\boldsymbol{X}_1 = \boldsymbol{X}^\mathrm{T}\boldsymbol{X}$, which requires $\mathcal{O}(4 \times 4 \times N_\mathrm{e})$ ($\boldsymbol{X}$ is of size $N_\mathrm{e} \times 4$),
- calculating $\boldsymbol{X}_2 = \boldsymbol{X}_1^{-1} = (\boldsymbol{X}^\mathrm{T}\boldsymbol{X})^{-1}$, where $\boldsymbol{X}_1$ is a 4×4 matrix. As the size of this matrix is constant, independently of the method to be chosen, the inversion will have a complexity of $\mathcal{O}(1)$,
- calculating $\boldsymbol{X}_3 = \boldsymbol{X}_2 \times \boldsymbol{X}^\mathrm{T}$ takes $\mathcal{O}(N_\mathrm{e})$ returning a $4 \times N_\mathrm{e}$ matrix,
- calculating $\boldsymbol{X}_3\boldsymbol{\Delta I} = (\boldsymbol{X}^\mathrm{T}\boldsymbol{X})^{-1}\boldsymbol{X}^\mathrm{T}\boldsymbol{\Delta I}$ takes $\mathcal{O}(N_\mathrm{e})$.

So the complexity of calculating the Fourier coefficients at each iteration via Eq. 4.21 is $\mathcal{O}(N_\mathrm{e})$. If the method takes N_it number of iterations, we can conclude that for each isophote the complexity will be $\mathcal{O}(N_\mathrm{e} \times N_\mathrm{it})$.

The number of isophotes will also depend on the object size: as explained in 4.3.1 isophotes are calculated by starting at the elliptical radius ξ^* where $I(\xi^*) = \frac{I_0}{2}$ and it is reduced iteratively by one pixel until reaching the center. Then it is increased from ξ^* until the radial gradient relative error $\sigma_{I'}(\xi)/I'(\xi)$ (where $I'(\xi)$ is the radial derivative of the intensity and $\sigma_{I'}(\xi)$ is its standard deviation, see 4.3.1) is smaller than a certain value, representing where the object can no longer be identified within the noise. This is a measurement of the object size n_obj. Using this assumption, the total complexity of the method is $\mathcal{O}(n_\mathrm{obj} \times N_\mathrm{e} + n_\mathrm{obj} \times N_\mathrm{e} \times N_\mathrm{it}) = \mathcal{O}(n_\mathrm{obj} \times N_\mathrm{e} \times N_\mathrm{it})$

Sérsic Fitting The Sérsic fitting is done by finding the Sérsic parameters (n, R_e, ϵ) that fit Eq. 4.8 to the elliptical isophotes by non linear least square minimization. At each iteration of this minimization it is needed to calculate Eq. 4.8 which has a sum that goes to infinity. In practice this sum is calculated until the addition of a new term does not make a significant contribution. Assuming that each calculation of Eq. 4.8 takes N_conv iterations for each radial isophote, and that

the number of iterations for the non linear least squares optimization is N_{nlls}, the complexity for each fitting is $\mathcal{O}(n_{\text{obj}} \times N_{\text{conv}} \times N_{\text{nlls}})$.

Residual Image The residual image is calculated by subtracting an image of the obtained Sérsic profile convolved with the PSF to the data image. Of course, this can be done only over a stamp the size of the object, for which case the size of the PSF is not negligible. Thus, this requires modeling the object over a blank image of size $n_{\text{obj}} \times n_{\text{obj}}$, convolution with the Gaussian PSF of size $m \times m$, and subtraction of $n_{\text{obj}} \times n_{\text{obj}}$ pixels . This takes $\mathcal{O}({n_{\text{obj}}}^2)$, $\mathcal{O}(n_{\text{obj}} \times m \times \log m + {n_{\text{obj}}}^2 \times \log n_{\text{obj}})$, and $\mathcal{O}({n_{\text{obj}}}^2)$ respectively, which gives an overall complexity for calculating the residuals of $\mathcal{O}(n_{\text{obj}} \times m \times \log m + {n_{\text{obj}}}^2 \times \log n_{\text{obj}})$.

Furthermore, in order to get the low pass filtered image it is not necessary to re-filter the residual: we only need to convolve the stamp with the filter and subtract it directly from the filtered image of the previous object detection iteration. This takes exactly the same complexity $\mathcal{O}(n_{\text{obj}} \times m \times \log m + {n_{\text{obj}}}^2 \times \log n_{\text{obj}})$.

Overall Complexity Taking into account the complexities of all the steps described above, and N_{obj} objects per image (each of a mean size of n_{obj}) the overall complexity of our method for each object is

$$\mathcal{O}(n_x n_y(\log n_x + \log n_y) + N_{\text{obj}} n_{\text{obj}}(N_{\text{e}} N_{\text{it}} + N_{\text{conv}} N_{\text{nlls}} + m \log m + n_{\text{obj}} \log n_{\text{obj}})). \tag{4.31}$$

Quantifying Overall Complexity The size of the image defined by n_x and n_y and the number of objects per image N_{obj} will depend on the telescope and camera used. As a reference, SDSS data-release 7 images have 2048 × 1489 pixels and a mean of 1,369 objects (standard deviation of 815). The Petrosian radius of objects with a magnitude smaller than 18 have a mean of 6 arcseconds (approximately 15 pixels with a standard deviation of 10 pixels). As we are not using exactly this size for objects (see Section 4.3.1), we take a conservative approach and consider the size of the objects to be approximately $n_{\text{obj}} \sim 100$ pixels. The SDSS full width half max (FWHM) PSF is approximately 3.5 pixels, so assuming the filter size is 10 times the FWHM, we get to $m = 35$. The number of angles explored are $N_{\text{e}} = \pi a \sim 628$ pixels maximum, where a is the semi-major axis of the biggest ellipse, i.e. ~ 100 pixels. We have empirically tested our method and the number of iterations for calculating the Fourier coefficients is $N_{\text{it}} < 40$, the number of iterations for calculating Eq. 4.8 N_{conv} is between 10 and 50, and the number of non linear least squares iterations N_{nlls} is between 100 and 1000. Using these numbers, the complexity is dominated by the elliptical isophotes and Sérsic fitting.

1D Versus 2D Fit

Software packages for fitting 2D galaxy profiles, such as GIM2D (Simard, 1998) and GALFIT (Peng et al., 2002), use numerical optimization algorithms (Metropolis (Metropolis et al., 1953) and downhill gradient (Press, 2007) respectively) for fitting the PSF-convolved Sérsic model. Consider an optimization method that takes $N_{\rm opt}$ iterations to converge. For each iteration, the following steps are required:

- Represent the Sérsic profile in an $n_{\rm obj} \times n_{\rm obj}$ image. This takes $\mathcal{O}({n_{\rm obj}}^2)$.
- Convolve the image with the PSF. As shown above, this can take $\mathcal{O}(n_{\rm obj} \times m \times \log m + {n_{\rm obj}}^2 \times \log n_{\rm obj}))$ for a symmetric bi-variate Gaussian.
- Subtract the modeled image to the data and evaluate the residual. This takes $\mathcal{O}({n_{\rm obj}}^2)$.

The convolution is the step that takes longer, so the overall complexity per object is $\mathcal{O}(N_{\rm opt} \times n_{\rm obj} \times (m \times \log m + n_{\rm obj} \times \log n_{\rm obj}))$.

Our 1D fitting first calculates the elliptical isophotes in $\mathcal{O}(n_{\rm obj} \times N_{\rm e} \times N_{\rm it})$ and then fits the Sérsic profiles in $\mathcal{O}(n_{\rm obj} \times N_{\rm conv} \times N_{\rm nlls})$, which gives an overall complexity of $\mathcal{O}(n_{\rm obj} \times (N_{\rm e} \times N_{\rm it} + N_{\rm conv} \times N_{\rm nlls}))$. Of course, the detailed comparison between algorithms will depend on the object to be fitted, but we can have an estimate by using the numbers described above. The search Sérsic parameter space (n, R_e, ϵ) is the same for both a 2D fit and our 1D approach. In this sense, for a given optimization algorithm, we would expect to perform the same number of iterations, so we can consider $N_{\rm nlls} \sim N_{\rm opt}$. Using this, we obtain that our proposed method should be approximately 10 times faster than doing a 2D fit.

4.4 Empirical Results

4.4.1 Results over Simulations

Simulating images

In order to create mock data we placed Sérsic profiles into a blank image, then convolved the image with a Gaussian PSF (in order to incorporate atmospheric "seeing" into the mock image) and added white noise following a Gaussian distribution. When sampling Sérsic profiles with $n > 1$ an oversampling problem occurs near the center due to the steepness of the profile at its origin (see Eq. 4.4). In order to solve this, we subsampled pixels close to the center and integrated the intensity over these subpixels.

Figure 4.5 (a) shows a mock image with one profile at the center, with $I_0 = 9678.65$ (chosen to obtain a central intensity of ~ 100 in the convolved image), $R_e = 100$ pixels, $n = 4$, and $\epsilon = 0.5$. Figure 4.5 (b) shows a radial profile of this mock image plus the Sérsic profile and its analytical PSF convolution described by Eq. 4.8. The PSF convolution of the image Sérsic profile is almost the same as the one obtained by the analytical expression. On the other hand, it can be seen that the original Sérsic profile fits the PSF convolution as the radius increases.

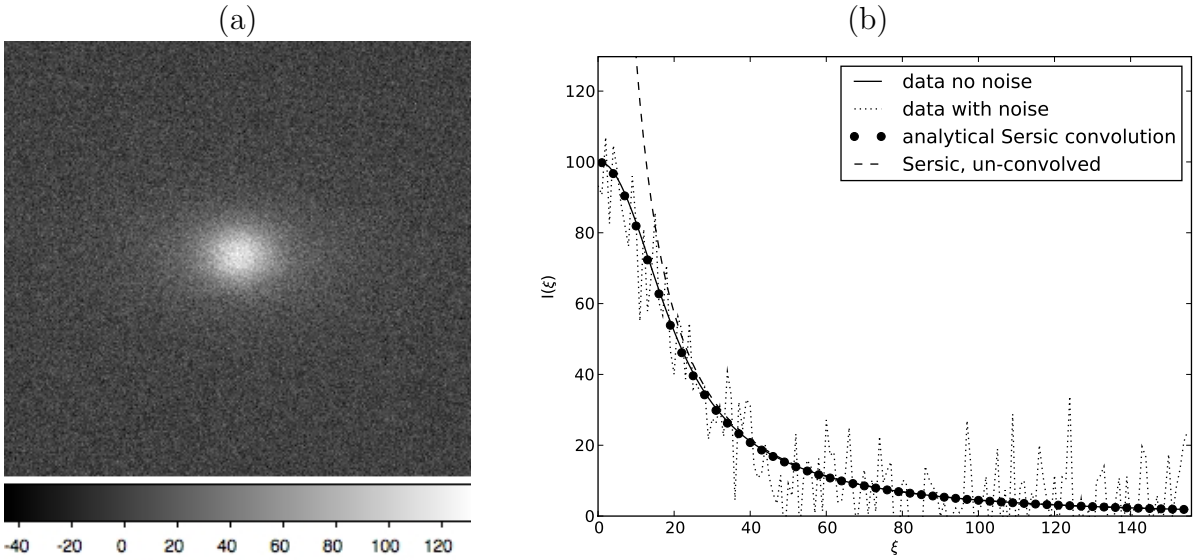


Figure 4.5: Sérsic Profile Mock Image. (a) A Sérsic profile after convolving if with a Gaussian PSF and adding white noise. (b) Radial profiles for the data before adding noise, after adding noise, the analytical convolution and the original Sérsic profile described by Eq. 4.4.

Results

In order to check the correctness of our method, we first created simulated images following the procedure described above for further detection and extraction. We randomly created 100 sets of Sérsic models and simulated 10 images for each of them using different realizations of noise. Each image was 512×512 pixels, and contained only one object with different random Sérsic parameters. Effective radii R_e were chosen between 5 and 100 pixels, ellipticities ϵ between 0 and 0.9, and Sérsic indexes n between 0.5 and 6. Central intensities I_0 were chosen by defining the signal to noise ratio (SNR). The SNR is a measurement of how high is the signal compared to the noise, and it is defined as

$$\mathrm{SNR} = \frac{I_{\mathrm{signal}}}{\sigma_{\mathrm{noise}}}, \tag{4.32}$$

where I_{signal} is the signal (or object) maxima, and σ_{noise} is the standard deviation of the noise. We chose SNR to be between 1 and 10. Once the SNR is chosen, we calculated I_0 using Eq. 4.9 so that the peak of the object with the PSF will match the SNR. The PSF used was a Gaussian with a standard deviation of 10 pixels.

Figure 4.6 shows errors of our algorithm for the 100 Sérsic models averaged over the 10 noise realizations for each of these models. Bars show standard deviation over these 10 error values. Integrated luminosities L are calculated using

$$L = \frac{2\pi I_o R_e^2 n}{b_n^{2n}} \Gamma(2n)(1-\epsilon), \tag{4.33}$$

where Γ is the complete gamma function (see Ciotti (1991)). Errors for luminosities L, effective radii R_e and Sérsic indexes n are calculated as

$$e_\Theta = \frac{|\Theta - \Theta^*|}{\Theta}, \tag{4.34}$$

where Θ is the model quantity to be evaluated (L, R_e or n), and Θ^* is the value obtained using our method. For the case of the errors on the ellipticities e_ϵ, we used $e_\epsilon = |\epsilon - \epsilon^*|$, as we know the maximum value for the ellipticity is 1. It can be seen that we have a mean error of 42.6% for the integrated luminosities, 42% for n, 49.4% for R_e, and 14.5% for the ellipticities.

Although our errors on Sérsic parameters are between 40% and 50%, we get a mean reduced χ^2 value over the simulated image pixels of 1.01 with a standard deviation of 0.01. Being the reduced χ^2 value close to 1, means that, though the parameters does not seem to fit correctly, the images obtained are very similar. In this sense, we can conclude that there are degeneracies in the Sérsic model, i.e. we can obtain similar images with different sets of parameters.

Figure 4.7 shows an example of our results over a 1024×1024 simulated image containing 20 objects. Central intensities were chosen such that SNR are between 3 and 10. Effective radii where chosen between 20 and 100, ellipticities between 0.1 and 0.9. Objects where convolved with a Gaussian PSF of standard deviation 10. Fig. 4.7a shows the objects, Fig. 4.7b shows the complete mock image over which we run our method. Fig. 4.7c shows the modeled objects our algorithm found, and Fig. 4.7d shows the residuals, which are the model convolved with the PSF subtracted from the original data. A perfect model would show only noise on its residuals. It can be seen that though our residuals are not perfect, we obtain a reduced χ^2 of 1.01, meaning the model obtained over the mock image is in agreement with the error variance.

4.4.2 Results over SDSS Images

The Sloan Digital Sky Survey (SDSS) is one of the most important surveys from the last decade. It obtained deep images of more than a quarter of the sky, including more

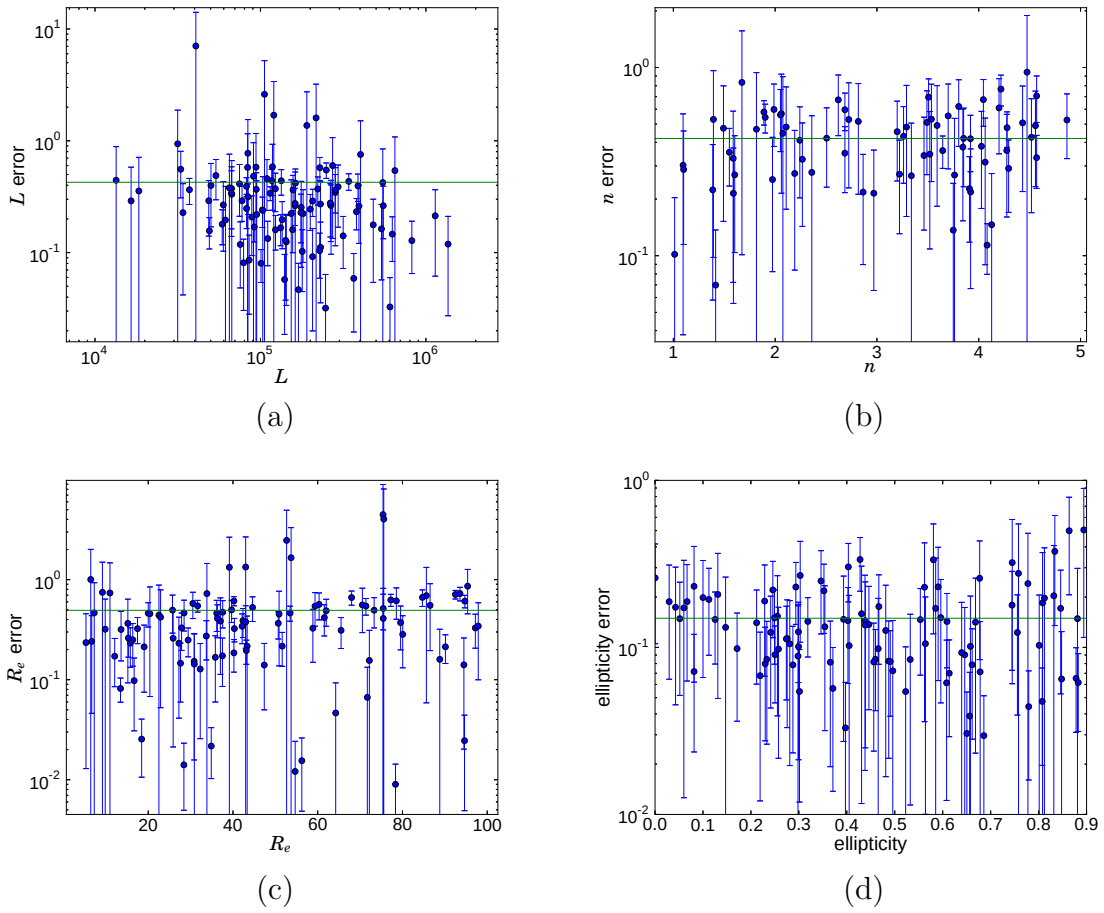


Figure 4.6: Errors calculated running our method over 100 simulations. (a) Errors of the integrated luminosities, with an average of 42.6%. (b) Errors of the Sérsic indexes n, with an average of 42%. (c) Errors of effective radii R_e, with an average of 49.4%. (d) Errors of the ellipticities ϵ, with an average of 14.5%.

than 930.000 galaxies. We used our method over 122 galaxies chosen from the brightest ones of this survey.

Blanton et al. (2005) have also fitted Sérsic profiles to SDSS galaxies and provided a public catalog called the NYU Value-Added Galaxy Catalog. They fitted the Sérsic parameters to circular mean flux annuli calculated by the SDSS pipelines. These fluxes are part of the SDSS's catalogs. They did not use an analytical formula for calculating the PSF convolved Sérsic profile. Instead, they created a grid containing profiles for different PSFs, effective radii and Sérsic indexes. As they fitted to circular annuli, they didn't take into account ellipticities or rotation angles.

We compared our results with the ones obtained by Blanton et al. (2005). Figure 4.8 shows this comparison. It can be seen, that our Sérsic indexes, calculated over

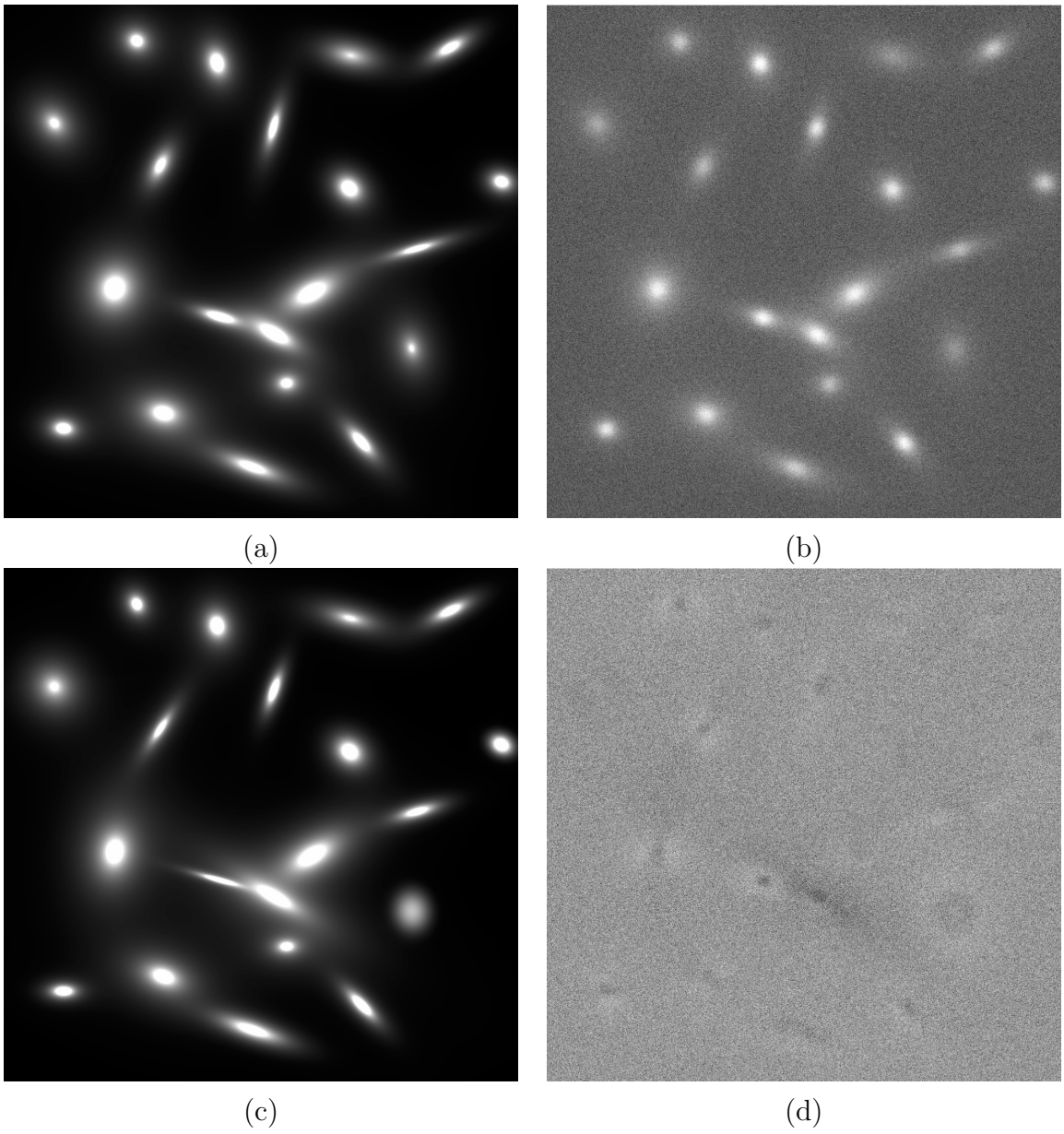


Figure 4.7: Example of our method over simulated data. (a) Objects added to the image. (b) Previous image convolved with the PSF and addition f Gaussian noise. (c) Obtained model. (d) Residuals.

elliptical isophotal profiles, match the ones from Blanton et al. (2005), calculated over circular annuli, within the error bars obtained in Fig. 4.6. Regarding the effective radii, we obtain higher values, mainly because of the fitting over circular annuli of Blanton et al. (2005). Furthermore, Spearman ranked correlations show that the probabilities of variables to be correlated are higher than a 99.9%.

Figure 4.9 shows results over some of these real galaxies obtained from the SDSS

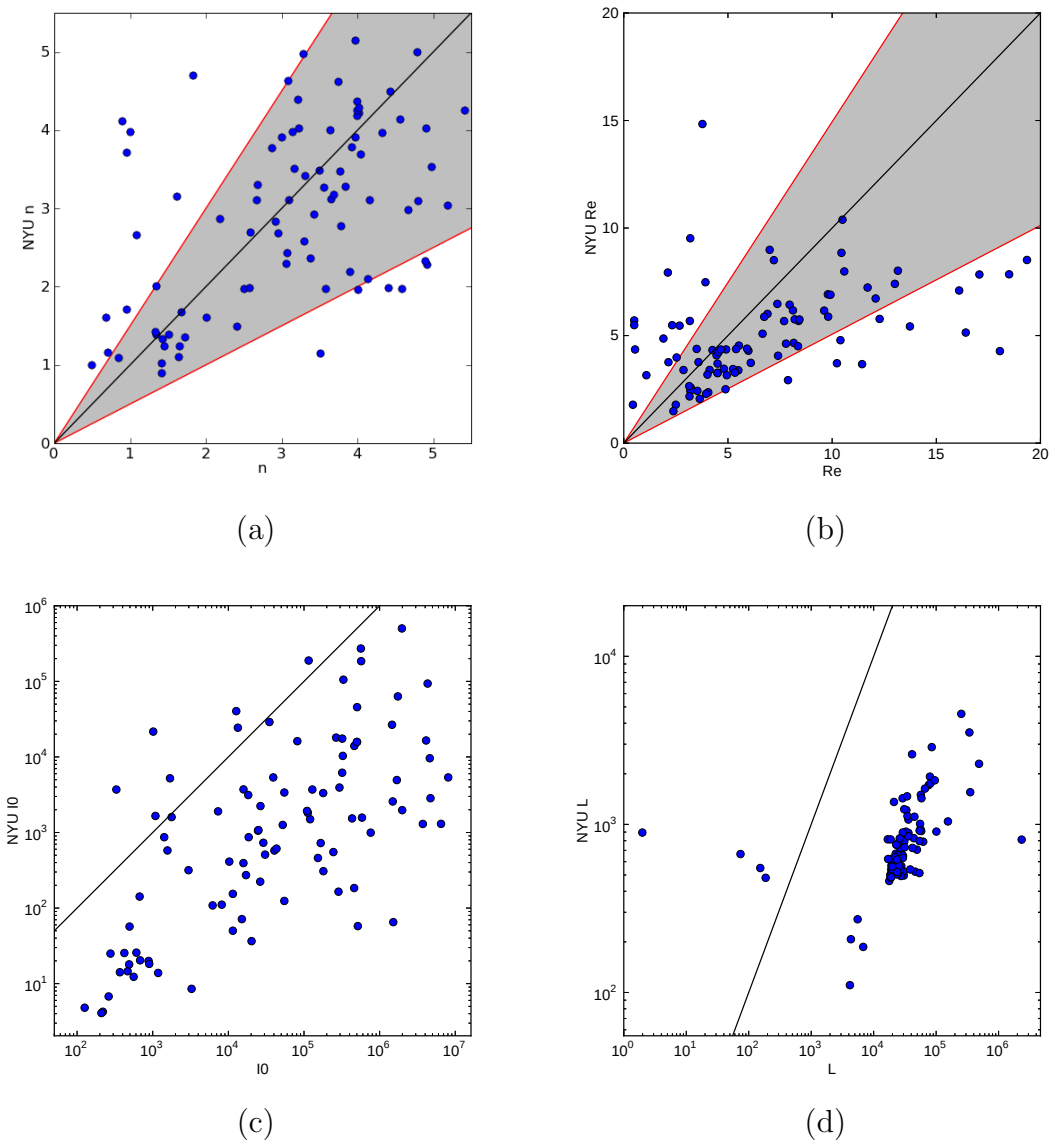


Figure 4.8: Comparison of Sérsic parameters obtained by our method (x axis) over SDSS data versus results from NYU Value-Added Galaxy Catalog (Blanton et al., 2005) (y axis). Black lines show a 1 to 1 relation. (a) Sersic index. Red lines represent a 42% error as described in Fig. 4.6. (b) Effective Radius. Red lines represent a 49.4% error as described in Fig. 4.6. Values by Blanton et al. (2005) underestimate Re due to their circular annulus profiles. (c) Central intensity. (d) Integrated luminosities calculated using Eq. 4.33 ($\epsilon = 0$ for NYU, as they fitted circular annuli).

database. The first row shows these real images, The second row shows residual images calculated using our method, and the third row shows residual obtained using the Sérsic parameters obtained by Blanton et al. (2005). In objects (a) to (c) our residuals look almost free from signal, though it is still possible to see some structure. This is mainly

caused because our radial profile is not able to detect structures such as galaxies' spiral arms. Figure 4.9d shows a pair of galaxies that closely cross the line of sight. This blended system makes our elliptical isophotes fitting to work incorrectly, so our fitted model is not characterizing accurately the light from the central galaxy. Figure 4.9e shows an edge on galaxy, rotated 90 degrees, so we see the edge of the disk. Spiral galaxies have a bulge in the center and a disk around it, with different light profiles. Trying to fit both of them at the same time leads us to this kind of errors. Figure 4.9f is a face on spiral galaxy, where again our method fitted mainly the bulge and failed to fit the disk and the structure of the spiral arms. Figure 4.9g shows a pair of interacting galaxies, where matter from the smallest one is being absorbed by the biggest one. This kind of images are very hard to characterize using a parametric model due to its irregular shape.

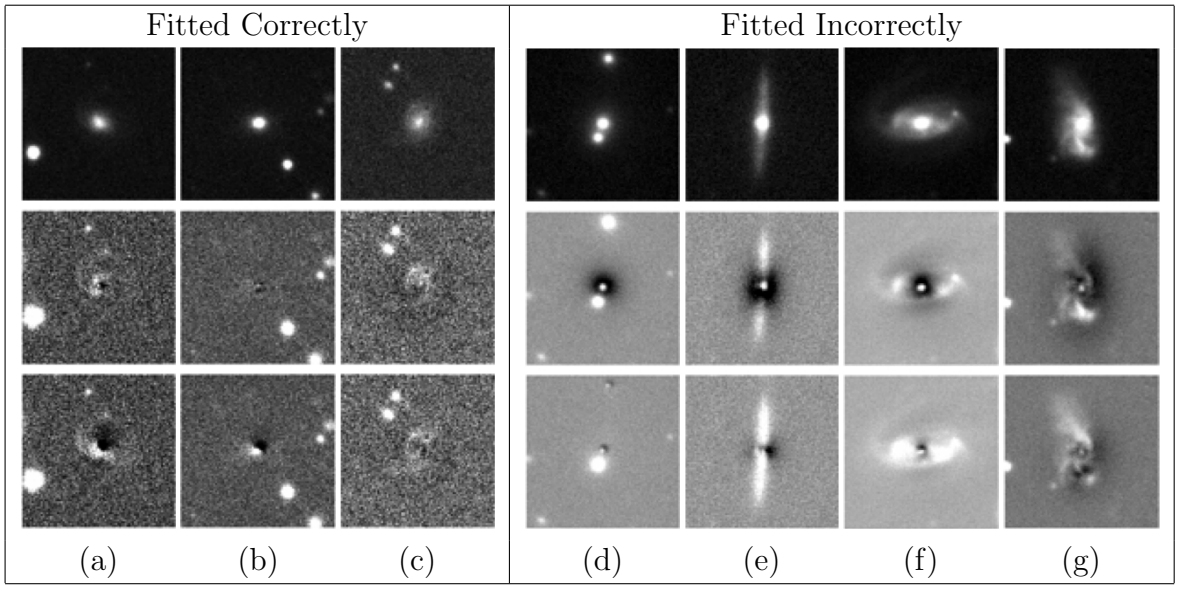


Figure 4.9: Comparison of our method over SDSS data versus results from Blanton et al. (2005). First row are original images obtained from SDSS. Second row are residuals obtained by our method. Third row are residuals from Blanton et al. (2005). (a) - (c) Good fits. (d) Strong blended system. (e) Edge on galaxy. (f) Face on spiral galaxy. (g) Two interacting galaxies.

4.5 Conclusions

We developed a tool for automatically extracting sources from astronomical images by fitting a parametric Sérsic model to them. The Sérsic model is a radial profile widely used in astronomy for characterizing brightness of galaxies. This new method minimizes the amount and complexity of real-time user input with respect to many

commonly used source detection algorithms. Our method consists of five steps:

1. Gaussian convolution of the target image with a Gaussian low-pass filter in order to diminish the noise.
2. Location of local maxima over the reduced noise image as candidates for objects.
3. Elliptical isophotes calculation over the target image around the highest intensity candidate.
4. Sérsic parameters fitting over the elliptical isophotes.
5. Subtraction of the fitted model from the target image and start again from step 1 with this residual image.

When running our method over mock images containing simulated galaxies we obtain good residuals, with a reduced χ^2 of 1.01. Nevertheless, we obtain between a 40% and 50% error on the Sérsic parameters, which indicates that there are degeneracies in the model parameters.

We also run our method over 122 true images obtained from the Sloan Digital Sky Survey. Our models fits the radial light from galaxies but it fails in detecting structure, such as spiral arms (Figure 4.9a and 4.9c), in blended objects (Figure 4.9d) in edge-on galaxies (Figure 4.9e), in galaxies where a clear distinction of the central bulge and the extended disk can be made (Figure 4.9f), and in galaxy mergers (Figure 4.9g) .

As proposed in this thesis, our method detects objects in all local maxima, but a lot of these may be only noise. In this sense, we are still missing a stopping criterion for our detection algorithm. We plan to test model fitting criteria such as the reduced χ^2 or the Bayesian information criterion (BIC, Schwarz et al., 1978) to determine when to stop. We also plan to measure our detection method in terms of false positives and false negatives obtained. At the same time, we need to measure how our method works on blended systems. This will depend mainly on the elliptical contours fitting. For highly blended systems, we would need to fit simultaneous ellipses at different centers for each one of the blended objects.

In order to take into account bulge-disk separation, a two component Sérsic model would be needed. This means fitting two Sérsic profiles at the same time. On the other hand, the problem of detecting structure in galaxies needs a totally different new approach. We first need to characterize the different kind of structures (spiral arms, bars, etc.) including irregularities. As a parametric model for all these features is very hard to find, we plan to create images of each of them and fit a combination of some or all of them to galaxy images.

One of our original goals was to apply our algorithm over the brightest objects in SDSS, but while developing it Simard et al. (2011) fitted 2D Sérsic profiles to these objects using GIM2D. Because of this, instead of addressing the problems described above immediately, we decided to continue with the classification problems using Simard et al. (2011) data.

In Chapter 6 we show how the Sérsic model fits the morphology of galaxies. We also show that stars can be fitted by Sérsic profiles and measure how good this parametric profile helps distinguishing galaxies from PSF-shape objects, which may be stars or unresolved galaxies.

Chapter 5

Labeling Bias

5.1 Introduction

Consider a typical supervised learning problem, where a training set $\mathcal{D}' = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^N$ is used to learn a function $f(\boldsymbol{x}) : \mathcal{X} \to \mathcal{Y}$ which predicts new data. This goal is achieved by using the N instances $\boldsymbol{x}_i \in \mathcal{X}$ and their respective labels $y_i \in \mathcal{Y}$ for obtaining this function. When $\mathcal{Y} = \{0, 1\}$ this is called a binary classification problem, when $\mathcal{Y} = \{1, \cdots, K\}$ this is called a multi-class classification problem, and when $\mathcal{Y} = \mathbb{R}$ we call it regression.

In real life, it may be very hard to obtain the actual real label y_i, which is usually called the *gold standard* or *ground truth* (GT). An estimate of this gold standard may be obtained by using a set of labels created by one or many human annotators. However, real data is never perfect, and labels given by the annotators can be systematically biased due to poor quality of the observed data they are labeling. For example, when labeling data by watching images, or video, human labels may be biased because of image resolution or video frames per seconds. This bias cannot be overcome by "re-training" the annotators, or by incorporating measurement error, which can in principle be very small. This is a significant challenge for fields which require training sets for which an absolute ground truth classification is impossible (e.g., astronomy).

Labeling bias is becoming increasingly important in the era of "Big Data", where it becomes difficult to create estimates of the ground-truth classifications due to the shear size and complexity of the data. In astronomy, researchers have gone from needing ground truth catalogs based on 1000s of galaxies to millions in less than 10 years. One solution has been to evolve from using professional astronomers (e.g. De Vaucouleurs et al. (1992) and Fukugita et al. (2007)) to "citizen scientists" via a crowdsourcing approach, such as Galaxy Zoo (GZ, Lintott et al. (2011)). The ground truth catalogs then include only those classifications where most of the crowd agrees on a single

classification. However, due to the quality of the data, the entire crowd can still be wrong. And of course, when one wants to apply a learning model using these ground truth data, the biases carry through the learning and classification process.

Our scientific motivation comes from the morphological classification of galaxies. The intrinsic shapes of galaxies are determined by their dynamical history and can be modified through events like collisions with other galaxies. By understanding the distribution of galaxy morphologies, we learn about their formation and evolutionary processes. In the simplest scheme, galaxies can be classified as elliptical or spiral (refer to Fig. 3.3a and Fig. 3.3b in Chapter 3). However, as the resolution of the images degrades with respect to the features of the spiral arms, annotators begin to classify spirals as ellipticals (see Fig. 3.3c). Again, this is not the fault of the annotator, since the arms can no longer be detected by the human eye. However, there are classification techniques which can still achieve the correct classification based on other properties. These other classification techniques require an unbiased ground truth data-set for training.

In this chapter, we present a formal definition of this systematic labeling bias, and a metric to assess it without knowing the ground truth.

5.2 Definition of Labeling Bias

Consider an object whose ground truth label is y, and which is labeled as $\hat{y}$ by the annotators. We will assume that we know a set of observables $\boldsymbol{\alpha} = \{\alpha_j\}_{j=1}^{N_\alpha}$ which bias the human labeling process (size of objects compared to resolution and their relative brightness compared to the noise, for example), $\alpha_j \in \mathcal{A}_j$. For a multi-class classification problem, we can define this bias in two ways:

1. **One vs One (OvO):** as the probability of a ground truth label $y = k_1$ to be labeled systematically by the annotators as $\hat{y} = k_2$:

$$\mathrm{P}(\hat{y} = k_2 | y = k_1) = p_{k_2|k_1}(\boldsymbol{\alpha}) \equiv p_{k_2|k_1}. \tag{5.1}$$

1. **One vs Rest (OvR):** as the probability of a ground truth label $y = k$ to be missed by the annotators:

$$\mathrm{P}(\hat{y} \neq k | y = k) = p_{\neg k|k}(\boldsymbol{\alpha}) \equiv p_{\neg k|k}. \tag{5.2}$$

In both of the definitions above, we are assuming that this probability depends only on the observable parameters $\boldsymbol{\alpha}$.

The labeling bias can be represented using a parametric or non parametric probability density function (pdf). In chapter 6 we show how we fitted a parametric function

for this bias. Optimally we would like to model all OvO biases, but this is harder than using the OvR approach: it involves obtaining $K(K-1)$ pdfs versus $K-1$. Furthermore, obtaining the OvR biases from the OvO is straightforward:

$$p_{\neg k|k} = \sum_{k_i \neq k} p_{k_i|k} = 1 - p_{k|k}. \tag{5.3}$$

5.3 Measuring Labeling Bias

As explained above, the label bias we are addressing depends on a set of *observed* properties of the objects (e.g., angular size, brightness, or distance from the observer). We define this set of properties as $\boldsymbol{\alpha} = \{\alpha_j\}_{j=1}^{N_\alpha}$. This bias will cause the fraction of objects of each class to vary in terms of these observable properties. In our galaxy morphology example, the fraction of elliptical galaxies will be larger for smaller galaxies than for larger galaxies because of annotators labeling small spirals as ellipticals. In this sense, for an un-biased data-set we would expect the fraction of objects of each class to be uniformly distributed in terms of the observed properties. In order to measure the amount of bias in a data-set, we calculate the deviation of the fractions in terms of the observable parameters $\boldsymbol{\alpha}$.

Though we expect the fractions to be uniformly distributed in terms of the observables, in real-life scenarios the distribution of fractions may vary in terms of *intrinsic* (or physical) properties. In Sec. 5.3.1 we describe how we measure the bias without considering such parameters, while in Sec. 5.3.2 we extend this approach for considering these intrinsic properties.

5.3.1 Parameter Independent Labels

A way of measuring the classification bias is by calculating the deviation of the observed fraction of objects from their real fraction in terms of the observed parameters $\boldsymbol{\alpha}$. We will create $N_{\mathcal{A}_j}$ single dimensional bins for each of the observed properties α_j and call them $\mathcal{A}_{j,l}$, where j runs over the observed parameter, and l over the bins. For each of these bins we can calculate the *observed class fraction* as

$$r_{j,k,l} = \frac{1}{N_{\mathcal{A}_{j,l}}} \sum_{i|\alpha_{i,j} \in \mathcal{A}_{j,l}} \delta_{\hat{y}_i,k}, \tag{5.4}$$

where $\hat{y}_i$ is the observed (biased) label of object i, $\alpha_{i,j}$ is its value of α_j, $\delta_{\hat{y}_i,k}$ is the Kronecker delta, and $N_{\mathcal{A}_{j,l}}$ is the total number of objects in bin $\mathcal{A}_{j,l}$:

$$N_{\mathcal{A}_{j,l}} = \sum_{k=1}^{K} \sum_{i|\boldsymbol{\alpha}_i \in \mathcal{A}_l} \delta_{\hat{y}_i,k}. \tag{5.5}$$

The observed class fraction as defined by Eq. 5.4 is the fraction of objects of class k at bin l of the observed property α_j. Now consider we know the real fraction of each label k, which we will call the *intrinsic class fraction*: r_k. In the case of a un-biased data-set, we would expect the intrinsic and observed class fractions to be the same. In order to measure the level of bias in a data-set we will use the deviation of the observed fraction from the intrinsic fraction by summing over all the bins for each property α_j

$$\sigma_{j,k} = \sqrt{\frac{1}{N_{\mathcal{A}_j}} \sum_{l=1}^{N_{\mathcal{A}_j}} (r_{j,k,l} - r_k)^2}. \tag{5.6}$$

We would expect an un-biased data-set to have $\sigma_{j,k} \sim 0$ for a large number of objects. This can be extended to all classes and observed properties as

$$L = \sqrt{\frac{1}{K N_\alpha} \sum_{j,k} {\sigma_{j,k}}^2} \sim 0. \tag{5.7}$$

L measures the deviation of the fractions within bins in $\boldsymbol{\alpha}$. In that sense, if the fractions do not depend on $\boldsymbol{\alpha}$, then L will be close to zero.

5.3.2 Parameter Dependent Labels

We will now address the scenario where the fractions of objects depend on physical or *intrinsic* properties (e.g., physical size or luminosity). Consider a set of intrinsic properties $\boldsymbol{\beta} = \{\beta_1, \cdots, \beta_{N_\beta}\}$ on which we define $N_\mathcal{B}$ multi-dimensional bins $\mathcal{B}_q$. Given a set of K labels (e.g. $K = 2$ for spirals and ellipticals), in each bin $\mathcal{B}_q$, we calculate the *intrinsic class fraction* of objects with each label as $r_{k,q}$. For typical galaxy morphology data-sets, we define $\boldsymbol{\beta}_i = (R_i, M_i)$, where R_i is the physical radius (in kpc) and M_i is the absolute magnitude for object i. In other words, given a fixed bin q in galaxy physical size and luminosity, $r_{k=\text{spiral},q}$ defines the *intrinsic fraction* of spirals compared to the total number of galaxies in bin q.

We then consider the set of *observed* properties of the objects (e.g., angular size, apparent magnitude, redshift). We define the set of properties $\boldsymbol{\alpha} = \{\alpha_j\}_{j=1}^{N_\alpha}$ and create single dimensional bins on each observed property for each of the $\mathcal{B}_q$ multi-dimensional bin $\mathcal{A}_{j,l,q}$. Here j defines which property and l defines the range of the bin for that property. For typical galaxy morphological data-sets, we define $\boldsymbol{\alpha}_i = (r_i/\text{PSF}_i, m_i, z_i)$ where r_i is the angular size, PSF_i is the estimated size of the point spread function at the galaxy location in the same units as its angular size, m_i is the apparent magnitude, and z_i is the redshift of galaxy i.

Note the intrinsic properties are treated in multi-dimensional bins, $\mathcal{B}_q$, whereas within each of those bins, the observed properties are treated in individual bins, $\mathcal{A}_{j,l,q}$.

This is because our aim is to study the biases with respect to their observed individual properties.

We then calculate the *observed class fraction*

$$r_{j,l,q,k} = \frac{1}{N_{\mathcal{A}_{j,l,q}}} \sum_{\substack{i|\alpha_{i,j}\in\mathcal{A}_{j,l,q} \\ \boldsymbol{\beta}_i\in\mathcal{B}_q}} \delta_{\hat{y}_i,k}, \tag{5.8}$$

where $N_{\mathcal{A}_{j,l,q}}$ is the total number of objects with the observed property α_j in bin $\mathcal{A}_{j,l,q}$. $\delta_{\hat{y}_i,k}$ is the Kronecker delta given an estimate of each galaxy i true classification ($\hat{y}_i$) for class k. The right-hand sides sums over all galaxies which are simultaneously in the observed single property bin $\mathcal{A}_{j,l,q}$ and the intrinsic property multi-dimensional bin $\mathcal{B}_q$.

For a given classification k and intrinsic property bin $\mathcal{B}_q$, we calculate the ℓ^2- Euclidean difference between the observed class fraction and the intrinsic class fraction and sum over all the $N_{\mathcal{A}_{j,q}}$ bins $\mathcal{A}_{j,l,q}$ for the observed property α_j

$$\sigma^2_{j,k,q} = \frac{1}{N_{\mathcal{A}_{j,q}}} \sum_{l=1}^{N_{\mathcal{A}_{j,q}}} (r_{j,l,q,k} - r_{k,q})^2. \tag{5.9}$$

Equation 5.9 should be ~ 0 for large N and when there is no difference between the intrinsic and observed class fractions, i.e., when the classifications are unbiased with respect to an observable.

We can extend this to all classes and intrinsic and observed properties as

$$L = \sqrt{\frac{1}{K N_{\mathcal{B}} N_{\alpha}} \sum_{j,k,q} {\sigma_{j,k,q}}^2}, \tag{5.10}$$

where K is the number of classes (2 for the case of elliptical versus spirals). We term equation 5.10 the classification bias which quantifies the difference in the *observational class fractions* with respect to the *intrinsic class fractions*.

We note that the *intrinsic class fraction* $r_{k,q}$ can vary for any data-set. For instance, a data-set designed to represent ellipticals might have an inherently low spiral fraction. One can still measure L for such a data-set and find it to have a low bias relative to a broader morphological catalog containing spirals, ellipticals and irregulars. In such a case, a fairer comparison would be to measure L only for the same specific class (e.g., ellipticals) in both data-sets. Alternatively, one might be interested in comparing classification algorithms over a wide range of classes. If so, the data should be joined to remove any selection effects which could influence $r_{k,q}$ and in turn the value of L.

One would hope that $r_{k,q}$ can be measured using an un-biased ("gold standard") data-set or perhaps a subset of the data itself. It is also possible that $r_{k,q}$ could be predicted from theory (e.g. Genel et al., 2014). Here, we take a conservative approach

and assume that *all* observed morphological data-sets have some level of bias in their intrinsic class fractions $r_{k,q}$. We make an estimate $\hat{r}_{k,q}$ by using the observed $r_{j,l,k,q}$ for the bin l in observed property j which is likely to have the least bias. For example, if we are calculating $\sigma_{j,k,q}$ for $\alpha_j = \mathrm{r}/\sigma_{\mathrm{PSF}}$, then we calculate $\hat{r}_{k,q}$ for the bin which includes the largest values of r/σ_{PSF}, since it should contain the least biased classifications.

5.3.3 Measuring Classification Bias in Galaxy Morphology Data-sets

We start with Galaxy Zoo 1 data release (Lintott et al., 2011) and their sample with spectra in SDSS which contains classifications for 667,944 galaxies achieved by crowd-sourcing. We cross-match these data to the SDSS DR7 to obtain their apparent magnitudes (Petrosian r-band), their apparent sizes (Petrosian r-band radii), their redshifts, and estimates of each galaxy's point-spread function (PSF). We used the SDSS field-specific psfWidth_r parameter as an estimate of the FWHM for a Gaussian PSF at the location of each galaxy. When galaxies belong to more than one field we used the smaller PSF. We take only those classifications where the probability of belonging to a class $P_{class} > 80\%$.

In Figure 5.1 we show how L decreases as we place limits on the data, including apparent sizes (relative to the PSF), redshift, and apparent magnitudes of the samples, i.e., $(\frac{r}{\sigma_{\mathrm{PSF}}})_i \geq r^{\mathrm{lim}}$, $z_i < z^{\mathrm{lim}}$, and $m_i < m^{\mathrm{lim}}$ for all i galaxies. As we restrict the data to just nearby large and bright galaxies, the classification bias L gets smaller. From Figure 5.1 it is clear that there are subsets within the data which have much lower bias than the entire sample. We also find that L decreases as more galaxies are used in each bin (squares, triangles, and diamonds). Thus, there is a definitive (but weak) statistical bias in L, unless more than ~ 50 galaxies per bin in $\mathcal{A}_{j,l,q}$ are used to measure equation 5.10.

We now compare L for different data-sets. As we noted earlier, care needs to be taken when comparing L due to variations in the intrinsic class fractions in any given data-set. First, we require the observed properties, the intrinsic properties, and the classification labels for the galaxies. Each morphological data-set we compare is based on the Sloan Digital Sky Survey and so we cross-match to SDSS DR7 to incorporate the same observed Petrosian r-band apparent magnitudes, Petrosian radii, PSFs, and redshifts.

Fukugita et al. (2007) have visually classified 2,275 galaxies, each by three experts. They defined a morphological index T such that $T = 0, 1, 2, 3, 4, 5, 6$ for E, S0, Sa, Sb, Sc, Sd, Im, respectively. In order to measure their bias, we focus on just the ellipticals (+S0) galaxies (N=941) having $0 \leq T < 2$, and the spirals (N=902) having $2 \leq T \leq 5$, since the other data-sets we compare to only use these two classes. We measured the

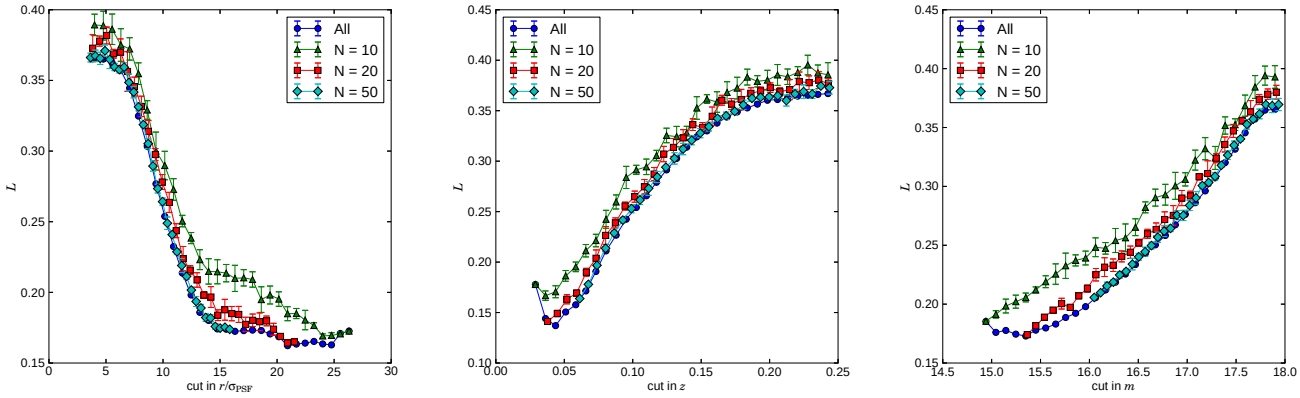


Figure 5.1: The effect of eliminating data from the original Galaxy Zoo classifications one variable at a time. The left panels shows the bias L as a function of the limit on the apparent Petrosian radius over PSF. The middle panel is with respect to limits on the redshift and the right panel is for apparent magnitude. The bias is smallest for sub-samples with large, nearby, and bright galaxies. These plots where calculated using 10 bins in R, M, and $\alpha_i \in (r/\mathrm{PSF}, m, z)$. Circles show results using all data for that particular data sub-set, triangles show results using 10 objects per bin, squares show results using 20 objects per bin, and diamonds show results using 50 objects per bin. Each of these points (except the circles) represent the mean over 5 randomly selected sub samples, and the error bars are the standard deviations.

Table 5.1: **Bias for different data-sets.**

data-set	L for 1,843 objects using $5 \times 5 \times 4$ bins
Fukugita	0.191
Huertas-Company	0.141 ± 0.011
GZ biased	0.318 ± 0.019
GZ debiased	0.156 ± 0.014

bias L (Eq. 5.10) over this sub-set using 4 bins in physical Petrosian radius (kpc) and absolute magnitude ($N_{\mathcal{B}}$ in Eq. 5.10) while choosing 5 bins in angular Petrosian radius over the PSF, apparent magnitude, and redshift ($N_{\mathcal{A}_j}$ in Eq. 5.9). These bins were chosen in such a way that each of them contained approximately the same number of galaxies ($\sim$ 23 galaxies per bin). This is smaller than the $\sim$ 50 galaxies per bin to achieve statistically unbiased measures of L (see above). Therefore, in the other larger comparison samples we randomly sub-selected down to N=1873 galaxies in order to account for any statistical bias in L. Through multiple random sub-sampling we also calculate the sample variance on L.

Having defined the bins and data-sample size based on the Fukugita sample, we then measure L for data provided in Huertas-Company et al. (2011), who used a *support vector machine (SVM)* classification model trained over the data-set from Fukugita et al. (2007). We defined elliptical (+S0s) galaxies as having a probability of being early-type $P(\text{Early}) \geq 0.5$ and spiral galaxies having $P(\text{Early}) < 0.5$. For our third and fourth data-sets, we use both the biased and de-biased Galaxy Zoo data-sets with $P(\text{Early}) \geq 0.8$ and spiral galaxies having $P(\text{Spiral}) \geq 0.8$.

In Table 5.1 we present the comparisons. Recall that Huertas-Company used SVM trained on the Fukugita sample, yet they achieve a lower overall bias then their training sample. This leads us to suggest that the application of machine learning techniques like Support Vector Machines (SVM) can inherently de-bias morphological classifications. We test this idea in the next chapter.

5.4 Conclusions

We analyzed four morphological data-sets for evidence of observational biases, including crowd sourced, expert, and machine learned labels. We developed two simple metrics L to quantify classification bias. The first approach considers the fraction of objects of each class versus observable parameters biasing our labels. We expect an

un-biased data-set to have a low deviation of the fractions with respect to the real fraction. Our second metric extends this idea to the case where the fractions depend on other intrinsic (or physical) parameters. We show that by discarding biased data, our metric diminishes, validating our approach. We also show that even the expert labels morphology catalogs are biased, and that some machine learning models achieve lower bias than these expert labels. In the next chapter we test this idea.

Chapter 6

Un-biased Classification

As stated before, historically Sérsic profiles with $n = 1$ have been associated to exponential disks of spiral galaxies, and $n = 4$ profiles to ellipticals. In this chapter we will investigate how well the Sérsic light profile recovers the morphologies of galaxies. We will pay special attention to biases in morphological classifications.

For obtaining our classification models, we mainly used Galaxy Zoo morphology labels and Sérsic profiles by Simard et al. (2011), as described in Section 6.1. Because of PSF convolution, there are natural limits to where a galaxy can be distinguished from a star in terms of their Sérsic parameters. In Section 6.2 we show a methodology for assessing this. In Section 6.3 we will compare the biases recovered by a SVM classification model against the biases in the data used for training that model using the Sérsic profiles. We will show that even though there is a small correlation between them, this is not statistically significant. In Section 6.4 we propose a method created specifically to model the bias and obtain the latent ground truth labels.

6.1 Data

We analyzed the parameter space of the Sérsic profile using 2D forward fitted models by Simard et al. (2011). We used their r-band single-component Sérsic fits, which were obtained using GIM2D (Simard, 1998) over SDSS images. This table contains 1,123,718 objects, but only 1,108,995 have a valid scale parameter to transform their effective radii from kpc into arcsec. For our PSF comparison, we also used a Gaussian approximation for the PSF at each galaxy. We obtained the standard deviation for such PSF by finding the field at which each galaxy belongs from the SDSS and retrieving its psfWidth_r parameter. Some galaxies belong to more than one field when they fall in overlapping regions. In this case, we used the smaller PSF.

In order to obtain elliptical / spiral classification, we used Galaxy Zoo 1 data release (Lintott et al., 2011). We used their debiased sample, which contains classification of 893,212 galaxies achieved by crowdsourcing. We joined this table with the 1,108,995 GIM2D profiles described above and obtained a total of 775,994 classes vs Sérsic parameters set.

The 1,108,995 GIM2D galaxies have some contamination of stars due to the SDSS classifying these stars as galaxies. In that sense, we also had some Sérsic profiles measured over stars. In order to separate these from the rest of the galaxies, and having a clean set of stars, we used UCAC3 (Zacharias et al., 2009). We obtained RA and DEC coordinates for each galaxy, and searched for objects in UCAC closer than 0.2 arcsec. We obtained 12,458 star candidates. We discarded possible contamination galaxies from UCAC by leaving only objects with absolute proper motion in RA or DEC higher than 2 mas/year. Combining the above tables we obtained a sample of 782,360 objects.

Figure 6.1 shows histograms for n and ϵ. Peaks can be seen at high n and ϵ. As explained in Simard et al. (2011), these peaks correspond to nuclear on-center sources, bar + point source configuration galaxies, or profiles trying to fit bars. We discarded objects with $n > 7.7$ and $\epsilon > 0.84$ from our sample in order to have a clean sample, leaving us with 719,862 objects as our main data-set. Although a peak of objects can be seen at low ϵ too, we decided not to discard these, as it is very likely those are stars contaminating the sample, which will be used to determine the Sérsic parameter space where we can distinguish stars from galaxies later.

6.2 Measurement Limits for the Sérsic Profile

As mentioned in Section 6.1 we discovered stellar contamination in the Simard et al. (2011) data-set. We noticed these stars generally have large n and low R_e. In order to asses when an object can be distinguished from a star in terms of its Sérsic parameters, we calculated the amount of integrated light outside the PSF ratio, $f_{>\mathrm{PSF}}(n, R_e, \epsilon)$, in terms of n, R_e and the galaxy ellipticity, ϵ, for PSF convolved galaxy radial profiles:

$$f_{>\mathrm{PSF}}(n, R_e, \epsilon) = \frac{\sum_i (I^{\mathrm{S}}(\xi_i; n, R_e, \epsilon) - I_{\mathrm{PSF}}(\xi_i))}{\sum_i I_{\mathrm{PSF}}(\xi_i)}, \tag{6.1}$$

where $I^{\mathrm{S}}(\xi; n, R_e, \epsilon)$ are the intensities of the convolved Sérsic profile along the semi-major axis, and $I_{\mathrm{PSF}}(\xi)$ the intensities of the PSF at a radius ξ. We simplified our analysis by using a symmetric bi-variate Gaussian PSF. All our distance measurements

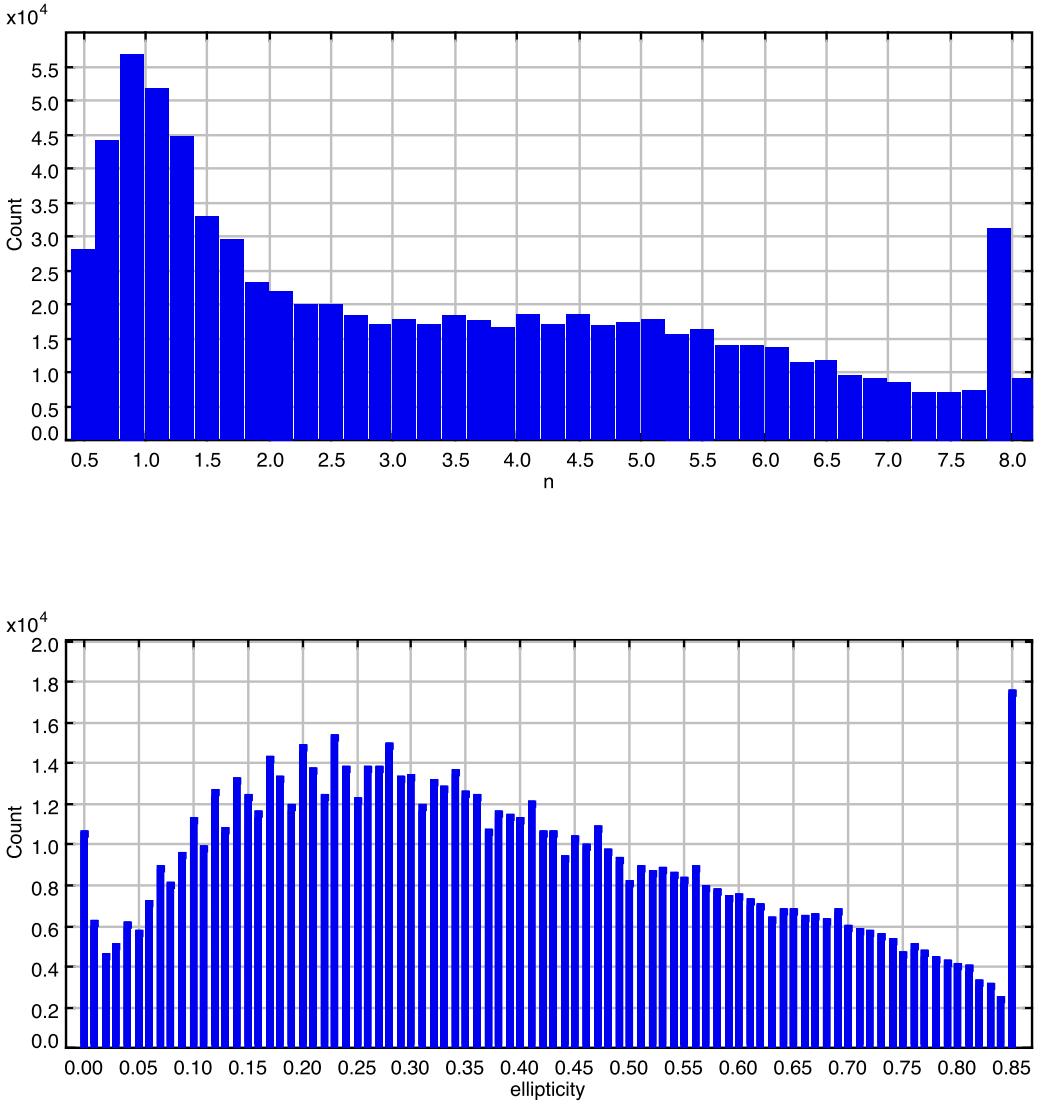


Figure 6.1: Histogram of Simard et al. (2011) fits in terms of their Sérsic index n (top) and ellipticities ϵ (bottom).

for this step were calculated in terms of the standard deviation of this Gaussian PSF, $\sigma_{\rm PSF}$, including ξ and R_e. For example, $R_e = \frac{R_e[\rm arcsec]}{\sigma_{\rm PSF}[\rm arcsec]}$.

Equation 6 from Trujillo et al. (2001a) (see Eq. 4.8) gives an analytical expression for 2D Sérsic profiles convolved with a Gaussian PSF. This equation involves Gamma and confluent hypergeometric functions and we encountered numerous numerical issues when trying to obtain galaxy radial profiles: functions growing too fast and too many sums to achieve convergence. These complications led us to work with image simulated convolved Sérsic profiles.

6.2.1 Simulations of Convolved Sérsic Profiles

We used a 1024×1024 blank image, and for each set of parameters (n, R_e, ϵ) we created a 2D Sérsic profile and convolved it with a symmetric bi-variate Gaussian PSF. This Gaussian had a standard deviation $\sigma_{\rm PSF}$ of 10 pixels, and the effective radius was modified accordingly in order to set it in units of $\sigma_{\rm PSF}$. We then obtained a radial profile along the semi-major axis $I^{\rm S}(\xi_i)$, where ξ_i runs from 0 to 512 pixels ($51.2 \times \sigma_{\rm PSF}$).

In order to obtain a more accurate description of the profile, in the sense of obtaining the correct flux of the galaxy, we sub-sampled every pixel according to

$$\Delta x(\xi) = \Delta y(\xi) = \lceil s \times e^{-b_n(\xi/Re)^{1/n}} \rceil, \tag{6.2}$$

where $\Delta x(\xi)$ is the size of the subsampling in the x axis, $\Delta y(\xi)$ is the size of the subsampling in the y axis, and s is the subsampling at the central pixel. We used $s = 100$ for the purpose of this paper. For each of these 1D galaxy profiles we saved 100 ξ_i and $I_i^{\rm S} = I^{\rm S}(\xi_i)$ values. In order to have higher detail in the center, we used $\xi_i = e^{i\Delta} - 1$, where Δ was chosen such that an intensity of 10^{-3} times the central intensity of the convolved Sérsic profile was obtained at the maximum radius. The values of $I_i^{\rm S}$ where interpolated over the pixel image. For obtaining the value at a different elliptical radius, we interpolated over $I_i^{\rm S}$. Both interpolations were performed using cubic splines. We simulated 50,000 profiles for $n = 0.2, 0.4, \cdots, 10$, $R_e = 0.1, 0.2, \cdots, 10$, and $\epsilon = 0.0, 0.1, \cdots, 0.9$. When getting the profile for a given set of Sérsic parameters, we performed nearest neighbor interpolation to the values described above and retrieve such profile. When calculating $f_{>\rm PSF}$ (Eq. 6.1), the central intensity of $I^{\rm S}$ and $I_{\rm PSF}$ were adjusted to be 1, and ξ_i was chosen uniformly from 0 to $5\sigma_{\rm PSF}$ in intervals of $0.1\sigma_{\rm PSF}$.

6.2.2 $f_{>\rm PSF}$ Contamination Threshold

Figure 6.2 shows the value of $f_{>\rm PSF}$ in terms of n, R_e, and ϵ. It can be seen that as n increases, R_e decreases and ϵ decreases, the amount of light outside the PSF diminishes,

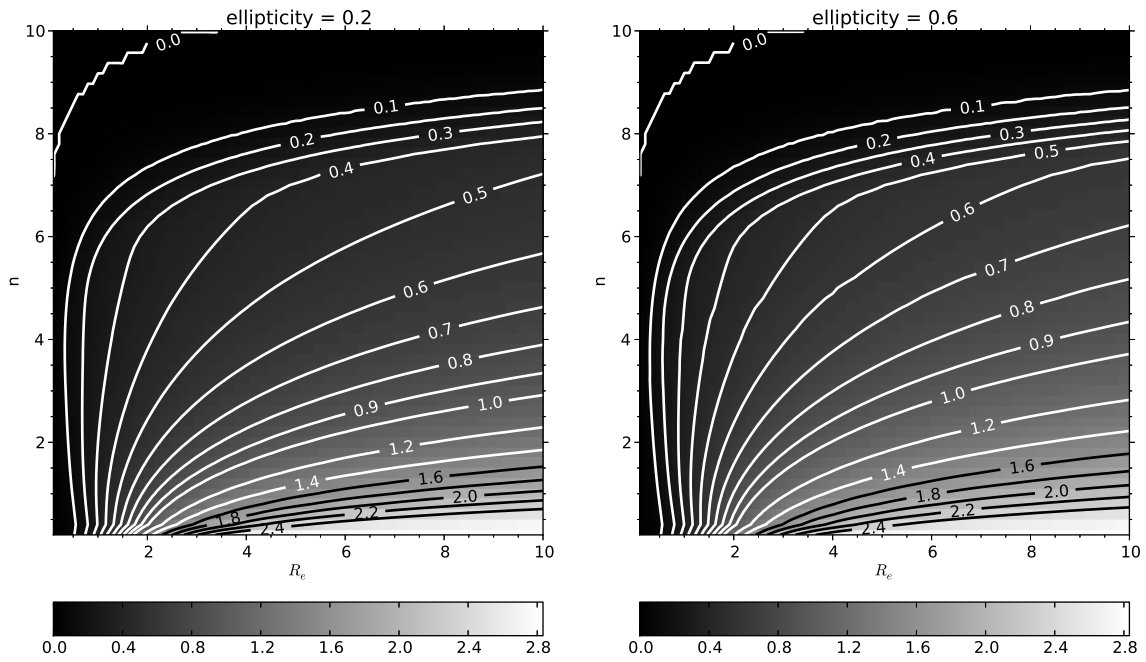


Figure 6.2: Ratio of integrated light outside the PSF in terms of n, R_e for $\epsilon = 0.2$ and $\epsilon = 0.6$. High n and low R_e radial profiles are indistinguishable from a star.

i.e. the light profile from the galaxy resembles a star. This imposes a natural limit in the parameter space where it is possible to distinguish a galaxy from a star.

In order to diminish stellar contamination in our galaxy sample, we calculated a limit for the integrated light outside the PSF ratio $f^{\star}_{>\mathrm{PSF}}$ such that stars cannot be distinguished from galaxies for $f_{>\mathrm{PSF}}(n, R_e, \epsilon) < f^{\star}_{>\mathrm{PSF}}$. We searched objects in the Simard et al. (2011) data that where closer than 0.2 arcsec to stars in UCAC3 (Zacharias et al., 2009) and have a probability of being a galaxy lower than 0.8 according to Galaxy Zoo and absolute proper motion higher than 2 mas/year. When adding the $n \leq 7.7$ and $\epsilon \leq 0.84$ restrictions our star classifications data-set totals 1,065. We then determined the value of $f^{\star}_{>\mathrm{PSF}}$ that best discriminates between stars and galaxies in terms of its balanced accuracy. We present the confusion matrix in Table 6.1, which was generated through cross-validation over 1,000 random sub-set iterations and using two-thirds of the data for training and one third for testing. This table shows the mean and standard deviation of the fraction of objects of class c_j classified as c_i. A 98.2% of the galaxies and a 89.5% of the stars can be distinguished by using only our $f_{>\mathrm{PSF}}$ measurement. Most of the galaxies will be kept, and though percentually a 10.5% of the stars will contaminate our galaxy sample, as the total number of stars is low (1,065), this will be non significant (112 stars versus $\sim 7 \times 10^5$ galaxies).

We then use all of the available data to measure $f^{\star}_{>\mathrm{PSF}} = 0.25$. In other words, the optimal star-galaxy separation occurs when the Sérsic flux is 25% larger than

Table 6.1: **Star/Galaxy Confusion Matrix**

$f(c_i\|c_j)$	star	galaxy
star	$89.5 \pm 1.4\%$	$1.8 \pm 0.1\%$
galaxy	$10.5 \pm 1.4\%$	$98.2 \pm 0.1\%$

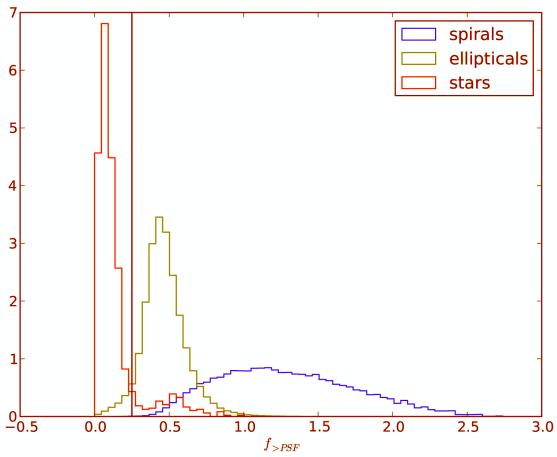


Figure 6.3: Histogram of $f_{>\mathrm{PSF}}$ for stars, elliptical, and spiral galaxies. A black vertical line shows the limit where galaxies are distinguishable from stars at $f^{\star}_{>\mathrm{PSF}} = 0.25$.

the PSF flux. Figure 6.3 shows a normalized histogram of $f_{>\mathrm{PSF}}$ for stars, spiral and elliptical galaxies according to the criteria described above. A black vertical line for the separation threshold $f^{\star}_{>\mathrm{PSF}} = 0.25$ is shown. Figure 6.4 shows the distribution of stars, E (+S0s) and S galaxies in terms of their Sérsic parameters. A line for $f^{\star}_{>\mathrm{PSF}} = 0.25$ is also shown.

6.3 Discarding Bias Labels Procedure

6.3.1 Choosing an Un-biased Sub-sample

In Section 5.3.3 we created a metric L for measuring the labeling bias of a data-set, and showed that by discarding small, dim, and distant galaxies we can get less biased sub-sets. We also showed that L is sensitive to the number of objects used for calculating it. In order to select an un-biased sub-sample, we searched the space of the observed parameters for the threshold limits that obtain the lower values of L. We used such un-biased data-set to train a SVM classification model hoping this model to be be

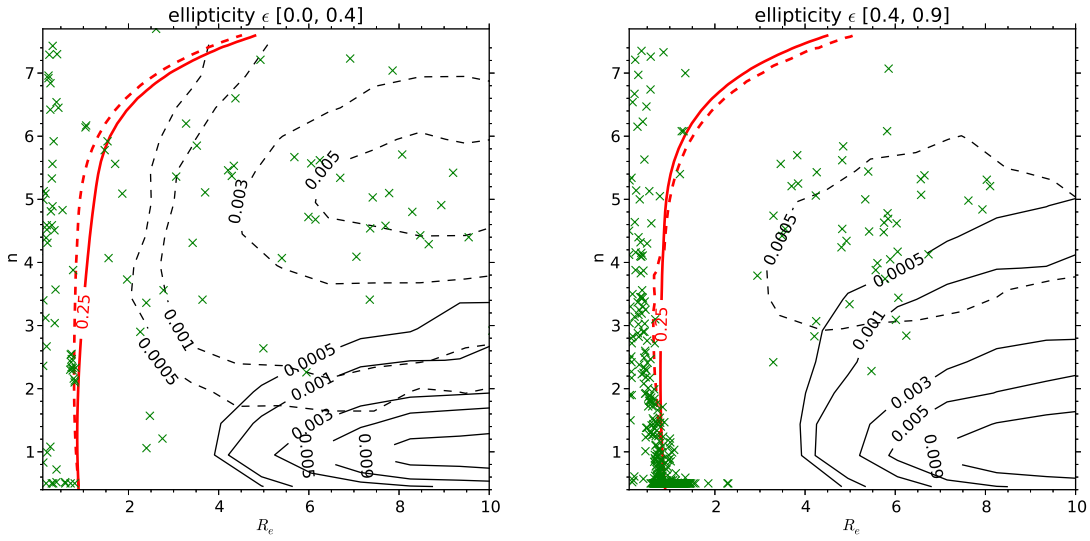


Figure 6.4: Star contamination in terms of $f_{>\mathrm{PSF}}$. Left: objects with ellipticities between 0 and 0.4. Right: objects with ellipticities between 0.4 and 0.84. Green crosses show stars, dashed contour lines show elliptical galaxies and straight contour lines show spiral galaxies. Red lines show the $f^{\star}_{>\mathrm{PSF}}$ threshold, where the straight line represents the lower ellipticity and the dashed line the higher ellipticity.

un-biased.

We examine 2-dimensional projections of L in Figure 6.5. In this case, we used the Galaxy Zoo de-biased sample from Lintott et al. (2011). As before, we only use galaxies with morphological classification probabilities greater than 80%. Notice that the range on L in Figure 6.5 is much smaller than in Figure 5.1 because of the de-biasing procedure applied and described in Bamford et al. (2009) and Lintott et al. (2011). However, there are still ranges in the observational properties of the sample which show much less classification bias than others. We can determine the least biased subset within the data by searching the entire N_α dimensional parameter space for the lowest L. We find $L = 0.074$ when we limit the data to $(\frac{r}{\sigma_{\mathrm{PSF}}}) \geq 11$, $m < 17.7$, and $z < 0.09$.

6.3.2 Elliptical vs Spiral SVM Classification

One of the goals of this work is to establish whether we can develop an elliptical/spiral classification model based on single-component two-dimensional Sérsic profile fits to the galaxy light profiles. To do so, we define a hyperplane

$$h(\boldsymbol{x}) = \boldsymbol{b}\boldsymbol{x} + b_0 = 0, \tag{6.3}$$

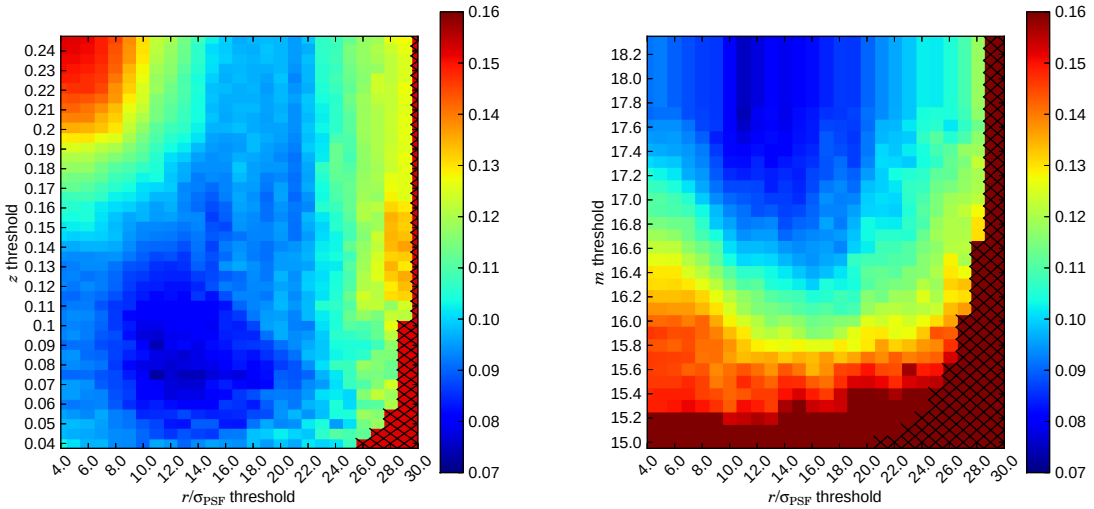


Figure 6.5: Same as in Figure 5.1 except for the 2D projection through a slice in the parameter space. We used 10 bins in R, M, and $\alpha_i \in (r/\mathrm{PSF}, m, z)$. Crosses are drawn over points with less than 5 objects per bin $\mathcal{A}_{j,l,q}$ in R, M, and α_i. We search this 3D parameter space to find the limits for the least biased sub-sample within a data-set in terms of L.

where $\boldsymbol{x} = (n, R_e, \epsilon)$. Galaxies on one side of the hyperplane are elliptical, while galaxies on the other are spirals. The values for $\boldsymbol{b}$ and b_0 are calculated such that the chosen hyperplane maximizes the distance from it to the closest vectors. In this framework, we are solving for the *support vectors* (SVM, Vapnik (1982)) which define the optimal separation between the classes using the Sérsic parameters. We choose to use a SVM classification model in order to be able to obtain an analytical hypersurface that can be directly and simply applied to any catalogue of Sérsic parameters.

In the Sérsic parameter space we do not expect the optimal separation to be a simple linear function. Therefore, we allow $\boldsymbol{bx}$ in equation 6.3 to be rewritten as a linear combination of vectors using a non-linear underlying kernel

$$h(\boldsymbol{x}) = \sum_{i}^{N_{\mathrm{SV}}} a_i \hat{y}_i K(\boldsymbol{x}_i, \boldsymbol{x}) + b_0, \tag{6.4}$$

where i runs over the N_{SV} *support vectors*, $\boldsymbol{x}_i = (n_i, R_{ei}, \epsilon_i)$. The chosen kernel is

$$K(\boldsymbol{x}_i, \boldsymbol{x}) = (\gamma R_{ei} R_e + \rho)^2 + \gamma^2 (n_i n + \epsilon_i \epsilon), \tag{6.5}$$

such that we can express the hypersurface as

$$h(n, R_e, \epsilon) = A R_e^2 + Bn + C R_e + D\epsilon + E. \tag{6.6}$$

We then use a training set to solve equation 6.3 for the support vectors. In the context of SVM, this amounts to solving the optimization problem

$$\min_{\boldsymbol{b},b_0} \frac{1}{2}||\boldsymbol{b}||^2 + \mathcal{C}\sum_{i}^{N}\xi_i \tag{6.7}$$

$$\text{subject to } \xi_i \geq 0, y_i(\boldsymbol{x}\boldsymbol{b} + b_0) \leq 1 - \xi_i, \forall i. \tag{6.8}$$

where we utilize the *soft margin* modification created by Cortes and Vapnik (1995), which introduces an additional "cost" parameter $\mathcal{C}$ in lieu of uncertainties on the classifications themselves.

Note that the kernel in equation 6.5 has tuning parameters γ and ρ, which define the effective kernel used in the analysis. The "cost" $\mathcal{C}$ is also a tuning parameter. These are free parameters in the SVM analysis and so we use cross-validation to estimate them. In the next sub-section, we discuss the details of implementing our procedure to report the coefficients A, B, C, D, E in equation 6.6. These define the surface determined by the non-linear SVM analysis which is then used to classify ellipticals from spirals.

6.3.3 Results

In order to train and test the elliptical / spiral SVM classifications against the Sérsic parameters, we cross-matched the Simard data to Galaxy Zoo 1 data release (Lintott et al., 2011) using their de-biased sample with classification probabilities $\geq 80\%$. We then utilize the analysis presented in Section 6.3.1 and in Figure 6.5 to choose the least biased Galaxy Zoo subset to train on. We note that within this subset, 99.5% of the galaxies with ellipticities > 0.5 are classified as spirals. Therefore we limit the SVM training to galaxies with $\epsilon < 0.5$. In the determination of the bias L and in the validation we classify all galaxies with $\epsilon \geq 0.5$ as spirals.

Before we can use SVM to classify the galaxies with $\epsilon < 0.5$ as either spiral or elliptical, we need to estimate the tuning parameters described in Section 6.3.2. We use cross-validation on 5000 galaxies for training and an independent sample of 5000 galaxies for testing. Ten iterations of the data were drawn using a two-fold shuffle split. We then defined a grid with values of $C \in \{10^{-6}, 10^{-5}, \cdots, 10^{5}\}$, $\gamma \in \{10^{-6}, 10^{-5}, \cdots, 10^{5}\}$, and $\rho \in \{0, 10^{-1}, 10^{0}, \cdots, 10^{5}\}$ to run the SVM classification. For each combination of C, γ, and ρ, we measured the performance of the tuning parameter model based on the mean of the recovery fraction for the ten data iterations. Specifically, the recovery fraction is measured as the average of the percentages that our algorithm correctly recovers for each of the catalog morphologies. Recall that we use independent data-sets for training and testing (to measure these recovery fractions). The optimal tuning parameters are for the model with highest recovery fractions. We find $\hat{C} = 10^{-4}$, $\hat{\gamma} = 100$, and $\hat{\rho} = 10$.

Table 6.2: **Elliptical/Spiral Un-biased Morphological Confusion Matrix**

$f(c_i\|c_j)$	elliptical	spiral
elliptical	$92.1 \pm 0.3\%$	$4.9 \pm 0.1\%$
spiral	$7.9 \pm 0.3\%$	$95.1 \pm 0.1\%$

With estimates for the tuning parameters in-hand, we then conduct the same exercise on a larger sample of $10,000$ training and $10,000$ test galaxies. We find a total mean balanced accuracy of $93.7 \pm 0.1\%$. Table 6.2 shows the mean and standard deviation for the confusion matrix. This confusion metric is balanced in order to avoid inflated rates as a result of imbalances in the distributions of classes. The final classification model is obtained by training our SVM using all data (11,545 early types and 29,733 spirals). The classification surface is then described by $h(n, R_e, \epsilon) = AR_e^2 + Bn + CR_e + D\epsilon + E$ (Eq. 6.6), where $A = 0.0036, B = 1.0741, C = -0.2596, D = -0.9914$, and $E = -1.4135$. Figure 6.6 shows this classification surface in terms of the Sérsic parameters for objects with $\epsilon < 0.5$. Direct classification from new single component Sérsic fits is straightforward:

1. Calculate $f_{>\mathrm{PSF}}$ (Eq. 6.1). Objects with $f_{>\mathrm{PSF}} < 0.25$ cannot be distinguished from a star.
2. Classify galaxies with $\epsilon >= 0.5$ as spirals.
3. For galaxies with $\epsilon < 0.5$ calculate $h(n, R_e, \epsilon)$ (Eq. 6.6). If $h(n, R_e, \epsilon) < 0$, classify as E, otherwise as S.

As our data-set is not uniformly distributed and classes cannot be perfectly separated in terms of their Sérsic parameters, the accuracy of our classification model will depend on the zone of the classification space. This is shown in Figure 6.7, where we plot the cross-validated balanced accuracy in terms of each Sérsic parameter. For this purpose we performed 10 iterations separating the whole un-biased data-set randomly into two (half for training and half for testing). For each of these iterations, we calculated the balanced accuracy over a specified range (the vertical lines in Figure 6.7).

Not surprisingly, we find that balanced accuracy tracks sampling density. This is most apparent in the effective radius, where for $R_e > 15$ (kpc) the accuracy drops to lower than 80 % while peaking at >95% for $R_e = 6$ (kpc) where the sampling is best. In terms of n, we find that our model has its lowest accuracy at $n \simeq 3.5$. This is mainly due to the fact that E and S galaxies are more mixed around this value (see Fig. 6.6).

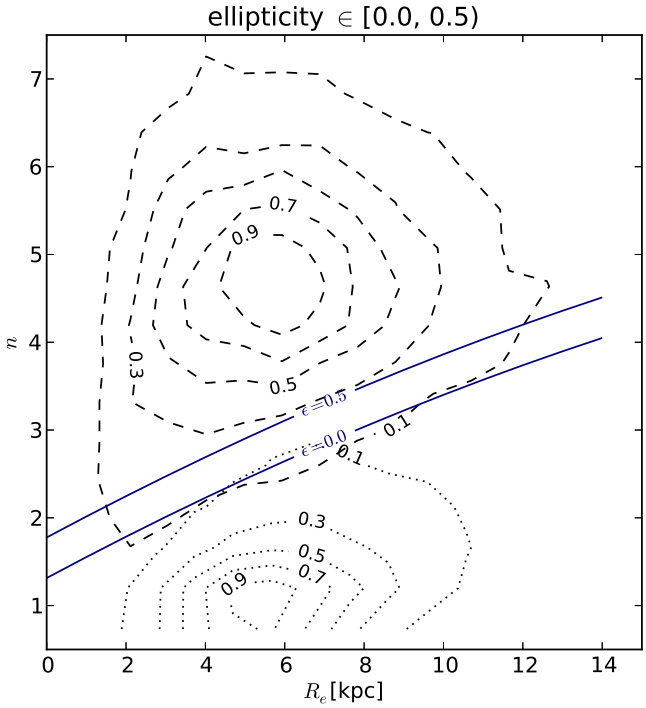


Figure 6.6: SVM classification surface for elliptical and spiral galaxies in terms of the Sérsic parameters and ellipticities for $\epsilon < 0.5$. Dashed contour lines show the distribution of elliptical galaxies and dotted contour lines show the distribution of spiral galaxies for our un-biased data-set. The contour labels denote the fraction of that type. Straight lines show the plane where $h(n, R_e, \epsilon) = 0$ for $\epsilon = 0$ and $\epsilon = 0.5$. This plot contains the 11,545 elliptical (+S0) and 29,733 spiral galaxies of the least biased Galaxy Zoo subset (Section 5.3.3) with $\epsilon < 0.5$.

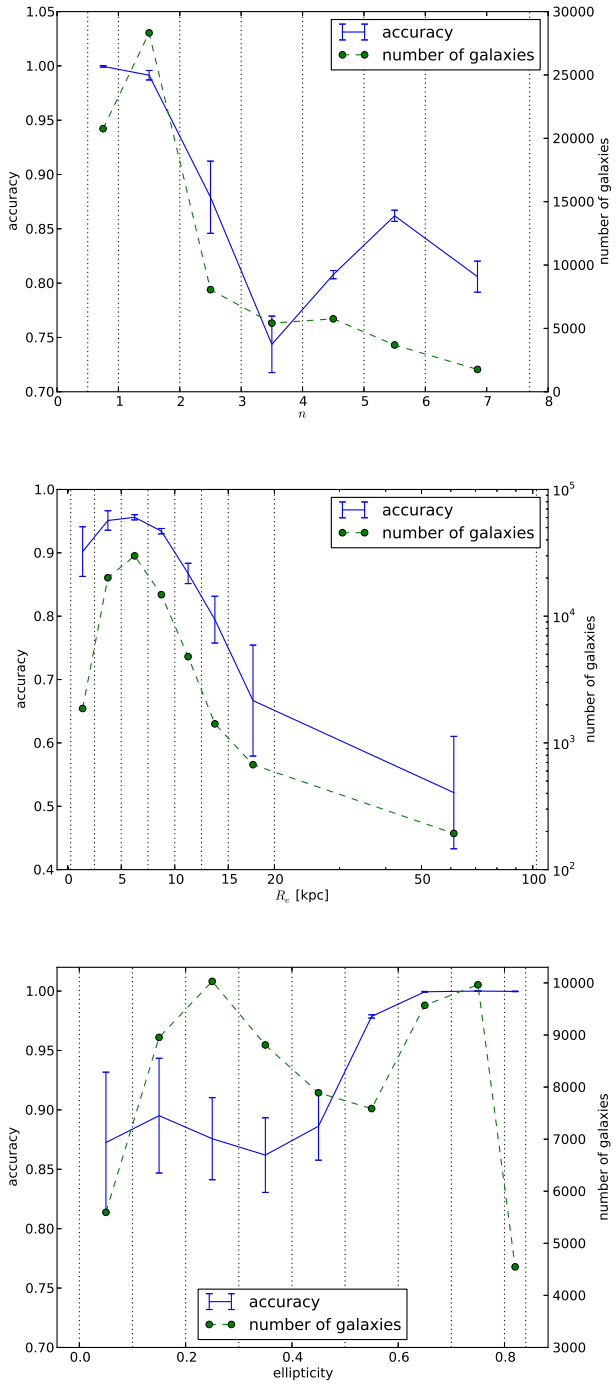


Figure 6.7: Cross-validation accuracy of our classification model versus morphological parameters. Solid blue lines show the accuracy in terms of each parameter. Dashed green lines show the number of objects used for calculating those accuracies. Dotted vertical lines delimit ranges for each parameter used for calculating the accuracies. Top: accuracy versus n. Center: accuracy versus R_e. Bottom: accuracy versus ϵ.

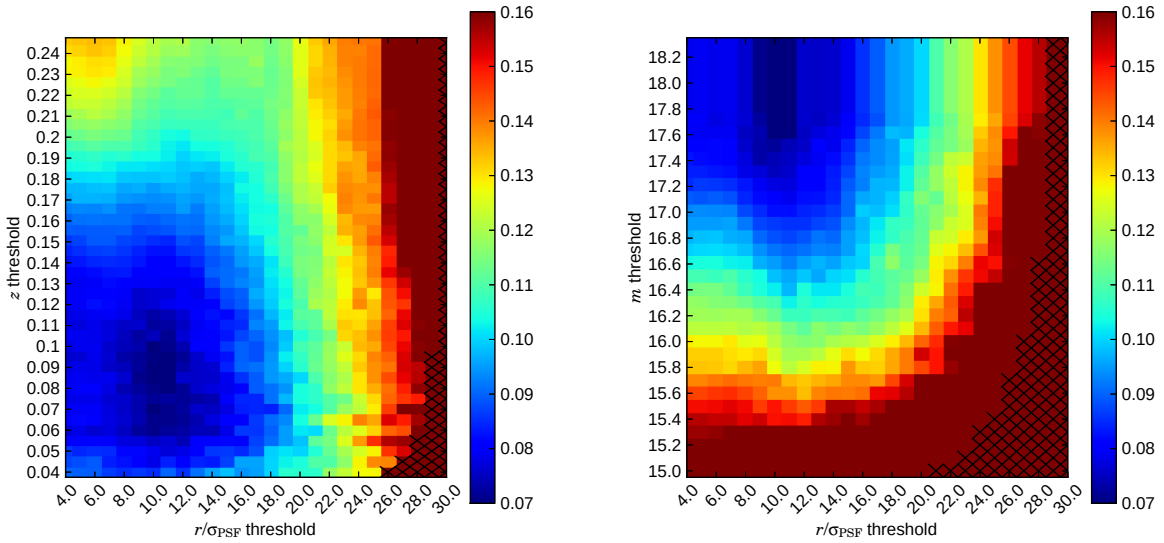
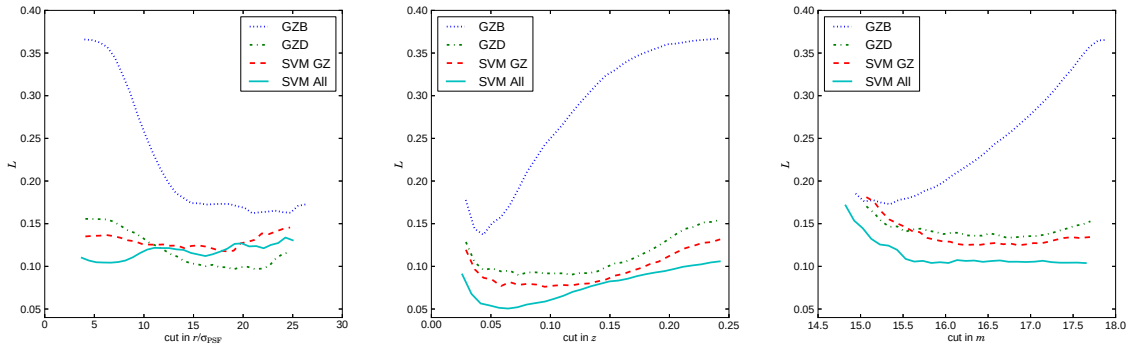


Figure 6.8: The effect of eliminating data over L (color bars) using thresholds over $\frac{R}{\sigma_{\rm PSF}}$, and z at the same time. These plot where calculated using 10 bins in $\bar{R}$, $\bar{M}$, and α_i. Crosses are drawn over points with less than 5 objects per bin in $\bar{R}$, $\bar{M}$, and β_i. The lowest biased subset is for $r/\sigma_{PSF} \leq 11.0$, $z \leq 0.085$,and $m \leq 17.8$ where $L = 0.068$.

After training on the Galaxy Zoo de-biased data-set, we can re-measure the classification bias as discussed in Section 5.3.3. Compare Figure 6.8 to the original de-biased Galaxy Zoo data shown in Figure 6.5. Note that these two figures use identical data-sets to measure L. The only difference is in the classifications themselves, where our morphologies have been determined entirely from the Sérsic parameters after training an SVM classifier on a subset of the Galaxy Zoo de-biased data. Our revised morphologies show reduced bias overall, including out to higher redshifts for the smaller galaxies.

Finally, we go back and apply our classification hypersurface to the entire data-set of Sérsic profiles from Simard et al. Recall that the majority of the galaxies in the Simard et al. data were not visually classified by Galaxy Zoo. We quantify the improvement of our SVM classifications (dashed) to the Galaxy Zoo de-biased classifications (dot-dashed) in Figure 6.9 which is similar to Figure 5.1, except now for a variety of morphological classifications. In terms of all of the observational parameters we study, the Sérsic SVM classifications show the lowest bias.

We matched our classifications to the SVM classified sample from Huertas-Company in order to fairly compare classification bias L (see the discussion in Section 5.3.3 on why this is important). We measured the bias L over this sub-set using 10 bins in physical Petrosian radius and absolute magnitude while choosing 10 bins in angular



(a) (b) (c)

Figure 6.9: Same as Figure 5.1 except now comparing the bias for different morphological classification data-sets. The dotted line is the biased Galaxy Zoo sample (the same as the data shown in Figure 5.1). The dot-dashed line is the de-biased classifications from Galaxy Zoo (also used in Figure 6.5). The dashed line is for the SVM morphologies which were trained on the Galaxy Zoo de-biased data and show an improvement in the overall bias for all redshifts and magnitudes and for small apparent sizes. The solid line shows the bias for the entire Simard et al Sérsic data-set using the SVM classifications. This sample shows the lowest overall bias for all observational parameters.

Petrosian radius over the PSF, apparent magnitude, and redshift. Using these values, we calculated L over five random draws of 50,000 galaxies each (50 galaxies per bin). Table 6.3 shows the mean and standard deviation of L for these sets of labels. It can be seen that although Huertas-Company labels are obtained via a model trained over biased labels (i.e., the Fukugita sample characterized in Table 5.1), they achieve a similar low level of bias compared to our method, which uses a similar machine learning technique except on different parameters. We also calculated the balanced recovery accuracy for our classifications compared to Huertas-Company to find 95.8%. In other words, the Sérsic light profile alone is almost entirely responsible for describing the morphologies in Huertas-Company.

Model Versus Training Bias

We investigate further why both our SVM classifications and the Huertas-Company SVM classifications show similar levels of bias even though they were trained on data-sets with different levels of bias. Recall that our training data uses a sample with limits placed on the sizes, apparent magnitudes, and redshifts of the galaxies. We choose these limits to ensure that the training sample has the lowest value of L (as shown in Figure 6.5).

Table 6.3: **Bias for different data-sets.**

data-set	L using 50,000 objects
Galaxy Zoo De-biased	0.159 ± 0.004
Huertas-Company	0.138 ± 0.004
Sérsic SVM	0.134 ± 0.003

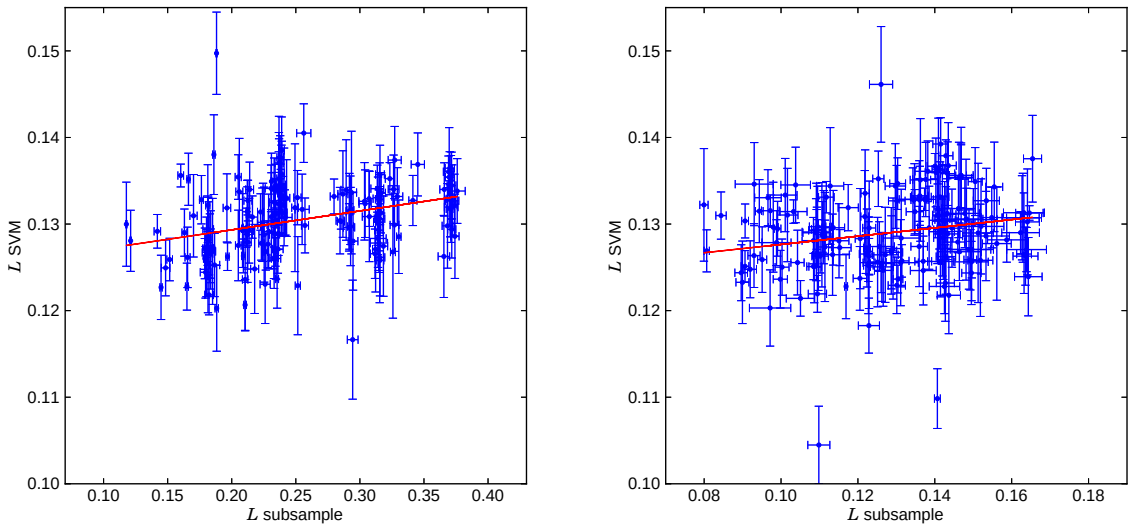
We test how important it is to enforce such training set limits by instead using a large number of training samples each with a varying total L (e.g., using different pixels in Figure 6.5). We then determine new sets of classifications, each using a training data-set with a different set of imposed limits. We measure the post-classification bias L and compare it to the bias in the original training data-sets (the pre-classification bias).

In Figure 6.10, we compare the pre and post Ls for the original (fully biased) Galaxy Zoo data (left) as well as for Galaxy Zoo de-biased data (right). As before, we only use galaxies with $P > 0.80$, and 10 bins in r/PSF, m, and z for each of 10×10 multidimensional bins in (R, M). We only detect a small (and not significant) trend in the pre versus post Ls. This holds true even when we train on the highly biased Galaxy Zoo data (left panel). We conclude that the quality of the initial training set is not crucial when using machine learning techniques like SVM to classify galaxy morphologies over Sérsic profiles. Therefore, it is expected that the Huertas-Company SVM classifications would show a similar level of bias as our own.

Extending to $z = 0.7$ data

We explore how our classification model extends to higher redshift galaxies by applying it to COSMOS data (Koekemoer et al., 2007). We used the single-component GIM2D Sérsic fits obtained by Sargent et al. (2007), and morphologies obtained from the Zurich Estimator of Structural Type (ZEST, Scarlata et al., 2007). The ZEST morphologies are based on a Principle Component Analysis (PCA) using five galaxy structural parameters (and not the Sérsic index). We used as E (+S0) their $T = 1$ labels and as S their $T = 2$ labels. Sérsic effective radii have units of arcsecs, so in order to convert them into kiloparsecs, we used redshifts from the COSMOS Photometric Redshift Catalog (Ilbert et al., 2009) as described by their zp_best parameter. By joining these catalogs, we obtained a total of 26,313 objects: 2,715 early types and 23,598 disk galaxies. McIntosh et al. (2005) note that there should be a small (few percent) size evolution in the effective radii which we ignore.

While both the COSMOS data and the Simard SDSS data used single-component



(a) (b)

Figure 6.10: Comparison of bias over selected sub-sample against SVM labels using 50 galaxies per bin. a) L of our classification method against the value of L of the original sub-set used for training using Galaxy Zoo biased labels. b) L of our classification method against the value of L of the original sub-set used for training using Galaxy Zoo de-biased labels. Straight lines show a total least squares linear regression. Though there is a relation between L in the training and fitted labels, it is not statistically significant.

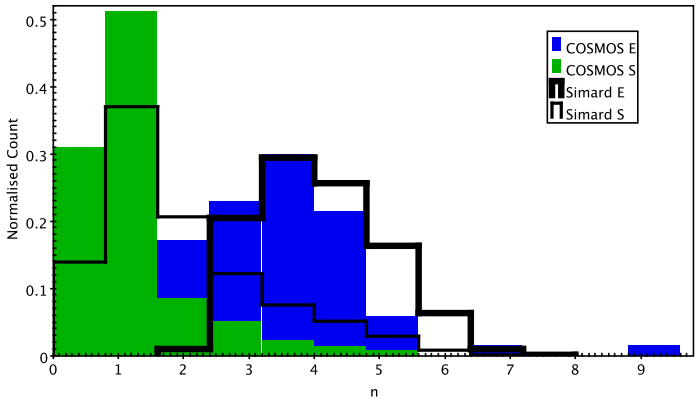


Figure 6.11: Distributions of the Sérsic index for galaxies with $1.5 \leq R_e < 3$, $M < -18$ and $z < 0.2$. We compare the Simard et al. (2011) fits on the SDSS data to the Sargent et al. (2007) fits on the COSMOS data. We use our SVM Sérsic surface morphological classifications. For these limits we obtain 34,590 early types and 86,466 spirals from the Simard data, and 70 early types and 233 spirals from the COSMOS data.

GIM2D Sérsic fits, we first compare the distributions of Sérsic indices for galaxies in similar redshift, absolute magnitude and size ranges. The COSMOS data is constrained to $z < 0.2$. Because the COSMOS data is much deeper in magnitude but much smaller in sky coverage, we use galaxies brighter than an absolute magnitude of $M_r = -18$ in order to have enough data. At this magnitude depth, the Simard et al. (2011) SDSS sample becomes highly incomplete beyond $z \sim= 0.08$ and so our comparison assumes that there is no evolution in the Sérsic index between $z = 0.2$ and $z = 0.1$. To account for any differences in the size distributions of the two samples, we compare at a fixed effective radius $1.5 \leq R_e \leq 3$ [kpc] for both data-sets. In Figure 6.11 we compare the histogram of Sérsic values of Simard et al. data to the COSMOS Sérsic indices from Sargent et al. (2007) after applying these redshift, magnitude, and effective radius constraints. We separate spirals and ellipticals in the COSMOS data using our SVM Sérsic surface for consistency (see below).

Figure 6.11 shows that is generally good agreement between the Sérsic index measurements for the SDSS and COSMOS data. There are more $n < 1$ in the COSMOS data compared the SDSS data, which we attribute to some incompleteness at $z = 0.08$ in the Simard data. The COSMOS data are lacking galaxies with Sérsic indices $n > 4$. We note that the Sargent et al. (2007) data has the same distribution of Sérsic indices as in the ZEST sample, and so the deficiency of $n > 4$ galaxies does not arise from the morphological classifications. The deficiency of $n > 4$ galaxies in the Sargent et al. (2007) could be a result of aggressive star-galaxy separation or perhaps due to cosmic variance of the small area of the COSMOS field.

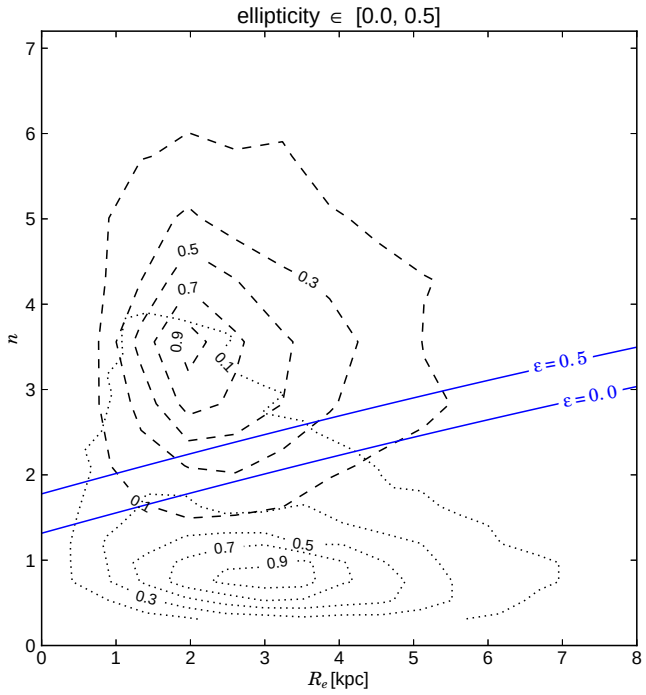


Figure 6.12: Distributions of E and S galaxies in COSMOS and our SVM classification surface for $\epsilon < 0.5$. Dashed contour lines show the distribution of elliptical galaxies and dotted contour lines show the distribution of spiral galaxies for COSMOS' data-set. The contour labels denote the fraction of that type. Straight lines show the surface where $h(n, R_e, \epsilon) = 0$ for $\epsilon = 0$ and $\epsilon = 0.5$. This plot contains 2,648 elliptical (+S0) and 14,061 spiral galaxies from the ZEST data with $\epsilon < 0.5$.

After having established that the Simard et al. (2011) Sérsic fits are statistically similar to the Scarlata et al. (2007) fits, we test how well our low redshift hypersurface works to separate spirals and ellipticals in the higher redshift COSMOS data. In Figure 6.12 and Table 6.4 we show that the $z \sim 0.1$ classifications based on the Galaxy Zoo data recover the $z \sim 0.7$ ZEST PCA-based morphologies to within $\sim 90\%$ accuracy. We note that the ZEST PCA-based morphologies are not "trained" and "validated" as in the SVM procedure we use here. Instead, Scarlata et al. (2007) examine the balanced accuracy using just 80 $z = 0$ galaxies, thus making it difficult assess the overall quality of the classifications. Scarlata et al. (2007) note that they classified face-on S0s as $T = 1$ while more inclined S0 galaxies where classified as $T = 2$, so this balanced accuracy is likely to be lower than it could be since we are calling non-face-on S0s spirals. This analysis shows that the ZEST morphological classifications can be recovered using the hypersurface in Sérsic parameters which separates spirals and ellipticals at low redshift, inferring little evolution in the hypersurface to $z \sim 0.7$.

Table 6.4: **ZEST/SVM Agreement Matrix**

		ZEST	
		E	S
SVM	E	91.90%	13.06%
	S	8.10%	86.94%

Classification based only on Sérsic index n

$n = 1$ exponential disks are usually associated with spiral galaxies, while $n = 4$ de Vaucouleurs profiles are associated with ellipticals. As we note in the Introduction, there are in fact $n = 1$ ellipticals, although many fewer than there are $n = 4$ spirals.

We test the contamination and completeness of ellipticals and spirals based on a single-valued Sérsic index. For instance, after normalizing in terms of the number of galaxies of each class, for $0.9 < n < 1.1$ as spirals and $3.9 < n < 4.1$ as ellipticals, we recover 99.62% of spirals with 0.38% contamination by ellipticals. On the other hand, for $n = 4 \pm 0.1$ we recover only 90.58% of ellipticals and the sample is contaminated by 9.42% spirals. We also test a simple linear discriminator based solely on the Sérsic index n, as opposed to the non-linear SVM using the index, the ellipticity, and the effective radius as described in Section 6.3.2. We find the best separation in the Simard et al. (2011) matched to the Galaxy Zoo de-biases sample is for $n = 2.79$ where we find a total accuracy of $91.63 \pm 0.16\%$. We correctly identified $89.07 \pm 0.73\%$ of the spirals ($10.93 \pm 0.73\%$ confusion) and $94.19 \pm 0.67\%$ of the ellipticals ($5.81 \pm 0.67\%$ confusion). This is lower than the hypersurface based balanced accuracy. In this case, we find that the ellipticals with small effective radii. Thus without the hypersurface, one cannot create a clean and complete sample of spirals.

6.4 Likelihood based De-Biasing

In this Section we present a method for de-biasing labels which simultaneously learns a classification model, estimates the intrinsic biases in the ground truth, and provides new de-biased labels. We test our algorithm on simulated and real data and show that it is superior to standard de-noising algorithms, like instance weighted logistic regression.

We use a parametric function for defining the bias in terms of the biasing attributes (apparent object size related to the resolution, for example), and obtain a maximum-likelihood (ML) estimator for the biasing parameters and a classification model (Sec.

6.4.2). This ML estimator is used to learn at the same time the classification model, the classification bias, and the ground truth labels. We use an expectation-maximization approach (Sec. 6.4.3) to obtain the biasing and classifier parameters. We use the ground truth labels as our unobserved latent variables. Though our proposed methodology is developed for any kind of bias or classification model, in order to test our algorithm, we chose a Gaussian bias and a logistic regression (LR) model, respectively (Sec. 6.4.4). We assess the functionality of our algorithm by testing it over simulations, and applying it to the morphological galaxy classification problem by using GZ majority voting labels (Sec. 6.4.4). As explained in Lintott et al. (2011) some de-biasing has already been done over GZ, but we show that our algorithm is able to further improve their labels.

6.4.1 Parametric Definition of Labeling Bias

The problem we are addressing is that of de-biasing labels at the same time we train our learning function $f(\boldsymbol{x})$. As before, we will assume that we know a set of observables which bias the human labeling process $\boldsymbol{\alpha} = \{\alpha_j\}_{j=1}^{N_\alpha}$ (size of objects compared to resolution and their relative brightness compared to the noise, for example), $\alpha_j \in \mathcal{A}_j$. For a multi-class classification problem, we define the bias $p_{k_2|k_1}$ as the probability of a ground truth label $y = k_1$ to be labeled by the annotators as $\hat{y} = k_2$. This probability will be represented by a parametric bias model defined by some vector of parameters $\boldsymbol{\theta}_{k_2|k_1}$, and will be a function of the observables which bias the classification $\boldsymbol{\alpha}$. In order to simplify the concept, we will use $p_{k_2|k_1}$ for this term, and later we will express our parametric bias model in terms of $\boldsymbol{\alpha}$ and $\boldsymbol{\theta}$:

$$\mathrm{P}(\hat{y} = k_2|y = k_1) = p_{k_2|k_1}(\boldsymbol{\alpha}|\boldsymbol{\theta}_{k_2|k_1}) \equiv p_{k_2|k_1}. \tag{6.9}$$

Binary Classification

Though our algorithm is easily extended to multi-class classification, for ease of exposition, we will focus on binary classification. In this case, the label obtained by annotators $\hat{y}_i \in \{0, 1\}$ will be a distorted version of the ground truth $y_i \in \{0, 1\}$. We will assume that a ground truth label of 1 will be measured by annotators as 0 with a probability of $p_{0|1}$ and that a ground truth label of 0 will be measured by annotators as 1 with a probability of $p_{1|0}$. In other words:

$$\mathrm{P}(\hat{y}_i = 0|y_i = 1) = p_{0|1}, \quad \mathrm{P}(\hat{y}_i = 1|y_i = 0) = p_{1|0}. \tag{6.10}$$

As stated above, $p_{0|1}$ and $p_{1|0}$ will be described later as parametric models, i.e. they are not necessarily constant.

6.4.2 Maximum Likelihood Estimator

We will assume our classification model is trained by adjusting a parameter vector $\boldsymbol{w}$, and gives us a probability $p_{\boldsymbol{w}}(\boldsymbol{x})$ of that instance to be of class 1:

$$\mathrm{P}(y = 1|\boldsymbol{w}, \boldsymbol{x}) = p_{\boldsymbol{w}}(\boldsymbol{x}). \tag{6.11}$$

$p_{\boldsymbol{w}}(\boldsymbol{x})$ can be any probabilistic classification model. Our goal is to obtain the parameters of our bias model and the parameters of this classification model given by $\boldsymbol{w}$ from the original biased data-set $\mathcal{D} = \{(\boldsymbol{x}_i, \hat{y}_i, \boldsymbol{\alpha}_i)\}_{i=1}^N$. For the sake of explaining our methodology, we will group parameters for the bias and the classification model into $\boldsymbol{\Theta}$. Later we will expand this and explain it in detail.

Assuming that the training points are independently sampled, the likelihood function of the parameters $\boldsymbol{\Theta}$ is

$$\mathrm{P}(\mathcal{D}|\boldsymbol{\Theta}) = \prod_{i=1}^{N} \mathrm{P}(\hat{y}_i|\boldsymbol{x}_i, \boldsymbol{\alpha}_i, \boldsymbol{\Theta}). \tag{6.12}$$

We can develop this equation by using

$$\mathrm{P}(\hat{y}_i|\boldsymbol{x}_i, \boldsymbol{\alpha}_i, \boldsymbol{\Theta}) = \mathrm{P}(\hat{y}_i, y_i = 1|\boldsymbol{x}_i, \boldsymbol{\alpha}_i, \boldsymbol{\Theta}) + \mathrm{P}(\hat{y}_i, y_i = 0|\boldsymbol{x}_i, \boldsymbol{\alpha}_i, \boldsymbol{\Theta}), \tag{6.13}$$

and assuming that the observables $\boldsymbol{\alpha}_i$ and the bias are independent of the instances $\boldsymbol{x}_i$, and that the ground truth labels depend only on $\boldsymbol{x}_i$

$$\begin{aligned} \mathrm{P}(\hat{y}_i, y_i|\boldsymbol{x}_i, \boldsymbol{\alpha}_i, \boldsymbol{\Theta}) &= \mathrm{P}(\hat{y}_i|y_i, \boldsymbol{x}_i, \boldsymbol{\alpha}_i, \boldsymbol{\Theta})\mathrm{P}(y_i|\boldsymbol{x}_i, \boldsymbol{\alpha}_i, \boldsymbol{\Theta}) & (6.14)\\ &= p_{\hat{y}_i|y_i}\mathrm{P}(y_i|\boldsymbol{x}_i) & (6.15) \end{aligned}$$

This way, we obtain the likelihood with probabilistic labels written as

$$\mathrm{P}(\mathcal{D}|\boldsymbol{\Theta}) = \prod_{i=1}^{N} \left(a_i p_i + b_i(1 - p_i)\right), \tag{6.16}$$

where we defined

$$\begin{aligned} a_i \equiv p_{\hat{y}_i|1} &= {p_{0|1}}^{1-\hat{y}_i}(1 - p_{0|1})^{\hat{y}_i}, & (6.17)\\ b_i \equiv p_{\hat{y}_i|0} &= {p_{1|0}}^{\hat{y}_i}(1 - p_{1|0})^{1-\hat{y}_i}, & (6.18)\\ p_i &= p_{\boldsymbol{w}}(\boldsymbol{x}_i). & (6.19) \end{aligned}$$

The maximum likelihood estimator may be found by maximizing the log-likelihood:

$$\boldsymbol{\Theta}_{\mathrm{ML}} = \max_{\boldsymbol{\Theta}} \{\ln \mathrm{P}(\mathcal{D}|\boldsymbol{\Theta})\} \tag{6.20}$$

6.4.3 Expectation-Maximization De-biasing Procedure

As we do not know the ground truth of our data, an expectation-maximization algorithm is a good choice to optimize our likelihood. EM is an iterative method for maximizing the likelihood when the model depends on unobserved latent variables. For our problem, we will use the unknown gold standards $\boldsymbol{y} = \{y_i\}_{i=1}^N$ as our hidden variables.

We start by defining our likelihood in terms of our data-set and the true labels as

$$\begin{aligned} \mathrm{P}(\mathcal{D}, \boldsymbol{y}|\Theta) &= \prod_{i=1}^{N} \mathrm{P}(\hat{y}_i, y_i|\boldsymbol{x}_i, \boldsymbol{\alpha}_i, \boldsymbol{\Theta}), && (6.21) \\ &= \prod_{i=1}^{N} \mathrm{P}(\hat{y}_i|y_i, \boldsymbol{x}_i, \boldsymbol{\alpha}_i, \boldsymbol{\Theta})\mathrm{P}(y_i|\boldsymbol{x}_i, \boldsymbol{\alpha}_i, \boldsymbol{\Theta}), && (6.22) \\ &= \prod_{i=1}^{N} \left[(a_i p_i)^{y_i}(b_i(1-p_i))^{1-y_i}\right]. && (6.23) \end{aligned}$$

Each iteration consists of two steps: the expectation (E) step and the maximization (M) step. The E step involves calculating the expectation of the log-likelihood using the current estimate for the parameters. The M step, consists in calculating the parameters that maximize this expected log-likelihood.

E-step

Given the current estimation for the parameters $\boldsymbol{\Theta}$, the expectation of the log-likelihood with respect to the conditional distribution of $\boldsymbol{y}$ given $\boldsymbol{x}$ can be calculated as

$$\mathbb{E}[\ln \mathrm{P}(\mathcal{D}, \boldsymbol{y}|\boldsymbol{\Theta})] = \sum_{i=1}^{N} \mu_i \ln(a_i p_i) + (1-\mu_i)\ln(b_i(1-p_i)), \tag{6.24}$$

where the expectation is calculated with respect to $\mathrm{P}(\boldsymbol{y}|\mathcal{D}, \boldsymbol{\Theta})$, and

$$\mu_i = \mathrm{P}(y_i = 1|\hat{y}_i, \boldsymbol{x}_i, \boldsymbol{\alpha}_i, \boldsymbol{\Theta}). \tag{6.25}$$

Using Bayes' theorem we obtain:

$$\mu_i = \frac{a_i p_i}{a_i p_i + b_i(1-p_i)}. \tag{6.26}$$

M-step

The model parameters $\boldsymbol{\Theta}$ are estimated by maximizing the current estimate of the expected log-likelihood. We used Newton-Raphson method, were $\boldsymbol{\Theta}_j$ is updated by

$\boldsymbol{\Theta}_j^{t+1} = \boldsymbol{\Theta}_j^t - \eta \boldsymbol{H}_{\boldsymbol{\Theta}_j}(\mathbb{E})^{-1} \boldsymbol{\nabla}_{\boldsymbol{\Theta}_j} \mathbb{E}$, where $\nabla_{\boldsymbol{\Theta}_j} \mathbb{E}$ is the gradient of the conditional expectation $\mathbb{E}$ with respect to $\boldsymbol{\Theta}_j$, $\boldsymbol{H}_{\boldsymbol{\Theta}_j}(\mathbb{E})$ is the Hessian matrix of $\mathbb{E}$ with respect to $\boldsymbol{\Theta}_j$, and η is the step length.

Parametric Bias

As explained above, we used a parametric function for our bias. This bias model has some parameters $\boldsymbol{\theta}$ which have to be fitted, and depends on the observables $\boldsymbol{\alpha}$ that bias the human labeling process:

$$p_{0|1} = p_{0|1}(\boldsymbol{\alpha}|\boldsymbol{\theta}_1), \quad p_{1|0} = p_{1|0}(\boldsymbol{\alpha}|\boldsymbol{\theta}_2). \tag{6.27}$$

In that sense, our bias parameters to be fitted are $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$, while the classification model parameter is $\boldsymbol{w}$, so, as defined above, our combined set of parameters to be fitted is $\boldsymbol{\Theta} = \{\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{w}\}$. In Section 6.4.4 we show a concrete example for the definition of this parametric bias.

Initial values for μ_i and $\boldsymbol{\Theta}$

If the biased labels are obtained from majority voting, the initial value for μ_i can be calculated as the fraction of annotators that classified object i as 1, over the total number of annotations. On the other hand, if the biased labels are binary (0 or 1), then μ_i can be initialized as $\hat{y}_i$.

As the bias parameters $\boldsymbol{\theta_1}$ and $\boldsymbol{\theta_2}$ are calculated numerically, an initial value is needed to start the M step. These initial parameters are calculated by fitting $r_{0,l} = r_0 + (1-r_0)p_{0|1}(\boldsymbol{\alpha}_l|\boldsymbol{\theta}_1) - r_0 p_{1|0}(\boldsymbol{\alpha}_l|\boldsymbol{\theta}_2)$, where $r_{0,l}$ is the fraction of objects of class $\hat{y} = 0$ within bins in $\boldsymbol{\alpha}$, r_0 is fitted and represents the real fraction of these objects, and l runs over the bins in $\boldsymbol{\alpha}$.

The initial value for $\boldsymbol{w}$ will depend on the classification model chosen. Our proposed methodology is valid for any classification model, but for illustrating the results on Sec. 6.4.4 we chose a logistic regression model. For this particular case, we chose the initial value for $\boldsymbol{w} = 0$.

Estimating the Ground Truth

The actual ground truth can be directly estimated from the posterior probability μ_i. This posterior probability is a soft probabilistic estimate of the real label. In that sense, the ground truth label can be estimated as 1 if $\mu_i > 0.5$ and 0 otherwise.

Labeling New Data

Once we have learned our classifier, we can label new data. We do so by directly using the resultant model $p_{\boldsymbol{w}}(\boldsymbol{x})$. We define a label to be 1 when $p_{\boldsymbol{w}}(\boldsymbol{x}) > 0.5$ and 0 otherwise.

6.4.4 Results

Classification and Bias Models

Though our method is described for any bias and classification model, in order to test them on real data, we defined our bias and classification model as

$$p_{0|1}(\alpha|\theta) = e^{-\frac{1}{2}\sum_j \alpha_j^2/(2\theta^2)}, \tag{6.28}$$

$$p_{1|0}(\alpha) = 0, \tag{6.29}$$

$$p_{\boldsymbol{w}}(\boldsymbol{x}) = \frac{1}{1+e^{-\boldsymbol{w}^\top \boldsymbol{x}}}. \tag{6.30}$$

These functions where chosen in order to correctly simulate the resolution bias in Galaxy Zoo, where α is the galaxy radius over the point spread function (PSF). For this problem we required $p_{0|1}(0|\theta) = 1$ (very small spiral galaxies are always going to look as ellipticals), and $\lim_{\alpha\to\infty} p_{0|1}(\alpha|\theta) = 0$ (very large galaxies are never going to be confused). On the other hand, elliptical galaxies are not going to be systematically labeled as spirals ($p_{1|0}(\alpha) = 0$). As stated above, $p_{\boldsymbol{w}}(\boldsymbol{x})$ was chosen to be a sigmoid in order to perform logistic regression (LR).

Simulations

In order to further illustrate the problem and our de-biasing method, we created simulations for the biased binary classification problem. The procedure used was the following:

1. Assuming a fraction r_k of labels of class $y = k$, for each class, we randomly created $r_k N$ instances $\{x_1, \ldots, x_{r_k N}\}$ following a bi-variate normal distribution of mean $\boldsymbol{\mu}_k$, and covariance matrix

$$\boldsymbol{\Sigma}_k = \begin{pmatrix} {\sigma_{k,1}}^2 & \rho_k \sigma_{k,1}\sigma_{k,2} \\ \rho_k \sigma_{k,1}\sigma_{k,2} & {\sigma_{k,2}}^2 \end{pmatrix}. \tag{6.31}$$

   To each of these instances we associated the gold standard k. This way we take into account that our gold standard labels are not perfectly separated by instances $\boldsymbol{x}_i$. We used $r_0 = 0.4$, $r_1 = 0.6$, $N = 1000$, $\boldsymbol{\mu}_0 = (0.8, 0)$, $\sigma_{0,1} = 0.3$, $\sigma_{0,2} = 0.4$, $\rho_0 = 0.5$, $\boldsymbol{\mu}_1 = (0.2, 0.8)$, $\sigma_{1,1} = 0.5$, $\sigma_{1,2} = 0.2$, and $\rho_1 = -0.5$.

2. We randomly assigned α_i values to each instance following a log-normal distribution of mean and standard deviation of $\mu_\alpha = 0.2$, $\sigma_\alpha = 0.4$.

3. In order to simulate the bias in terms of α_i, we modified labels $y_i = 0$ with a Gaussian probability $p_{0|1}(\alpha_i|\theta) = \exp\left(-\alpha_i^2/(2\theta^2)\right)$. We created a set of simulated scenarios choosing different standard deviations θ.

4. We added noisy labels by randomly modifying them with a probability of 0.05.

Figure 6.13 shows the distributions of the ground truth and biased labels for a simulation scenario where $\theta = 0.03$. For each of our scenarios, we created 10 train + 10 test data-sets. We fitted our classification models to the train sets and calculated accuracy, area under the ROC curve, and L over the test set. We used $N_{\mathcal{A}} = 10$ bins in α (each of these bins containing equal number of instances).

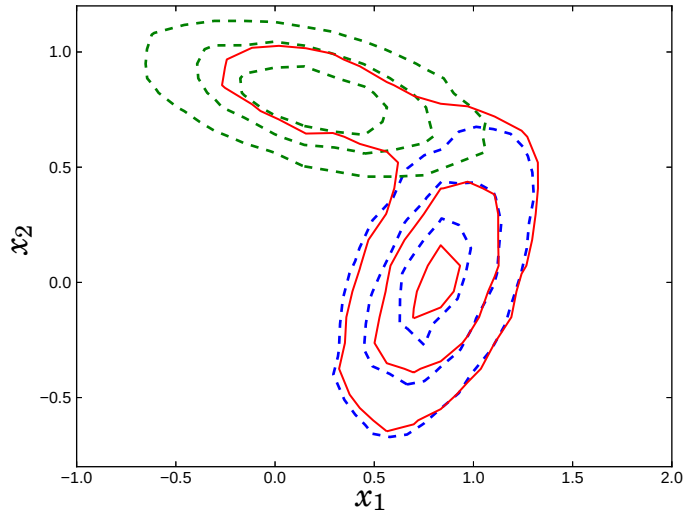


Figure 6.13: Distributions for the simulated ground truth labels y_i, and biased labels $\hat{y}_i$ in terms of $\boldsymbol{x}_i$ for $\theta = 0.03$. The dashed blue line is the ground truth distribution of $y = 0$ objects, the dashed green line is the ground truth distribution of $y = 1$ objects, and the red solid line is the biased distribution of $\hat{y} = 0$ objects. Notice that our simulation moves y=1 objects out of the parameter space in which they are contained and into the space defined by the $y = 0$ objects.

In order to assess the validity of our de-biasing algorithm, we also fitted a LR model over the ground truth labels, and over the biased labels. We also applied the weighted logistic regression (WLR) approach following Rebbapragada and Brodley (2007). For this case, in order to be consistent with the rest of our experiments, we defined weights using clustering (Gaussian mixture model, GMM) and used those weights to train a weighted LR.

Table 6.5 shows accuracy, area under the ROC curve (AUC), and L for the LR trained over the ground truth data, the LR trained over the biased data, the WLR, and our de-biasing EM approach (DEM). These values are shown for three simulations with different levels of bias: $\theta = 0.03$ (17% of the labels are modified), $\theta = 0.05$ (25% of the labels are modified), and $\theta = 0.1$ (35% of the labels are modified). Accuracies and AUCs are calculated using the GT labels. The last row of this table shows L calculated directly over the GT labels and the biased labels. As expected, training directly over the biased data-set gives us a smaller accuracy than training over the GT labels. WLR does better than training over the biased labels, but still does not get the accuracy level of training over the GT labels. We believe this is mainly due to the asymmetry of our bias. Furthermore, the goal of WLR is to work over noisy labels while the problem we are addressing is that of systematic bias in terms of a specific observable parameter. Our DEM procedure performs very similar as training over the GT. Furthermore, accuracies are comparable within their standard deviation. Figure 6.14 shows the accuracy of GT, WLR, and DEM in terms of the amount of bias as defined by θ. As the bias increases DEM maintains its accuracy, while the accuracy of WLR decreases. Another important conclusion from Table 6.5 is that the AUC is not sensitive enough to assess the correctness of our models. Many LR linear models can give similar values for AUC: they only need to be partially parallel to the best LR model. This also means that by choosing a probability threshold for the LR different from 0.5 would give a better accuracy, but without knowing the GT labels choosing this threshold is not straightforward. L (Eq. 5.7) on the other hand, suits perfectly the purpose of measuring bias of labels. The values of L using DEM are comparable to the values obtained over the ground truth labels. Furthermore, Figure 6.15 shows the fraction of $y = 1$ labels in terms of α for $\theta = 0.1$. The bias of $y = 0$ labels as α decreases can be seen on the biased fraction curve. Our proposed EM method is able to reduce the bias significantly, and as α increases, the fractions are not significantly different from the ground truth neither for the bias nor for our DEM method's labels. At the same time, the fractions from WLR are much smaller than the GT fractions, which causes L to increase importantly. This is due to the fact that the WLR model erroneously classifies more objects as being of class 0 than there really are.

Galaxy Zoo

We used 10,000 galaxies classified by Galaxy Zoo through crowdsourcing. These galaxies where chosen so they are uniformly distributed in our parameter space $\boldsymbol{x}$, and $\boldsymbol{\alpha}$. Labels were assigned by thresholding over already unbiased probabilities (see Bamford et al. (2009) and Lintott et al. (2011)). Galaxy labels are assigned by using a 50% probability threshold. Ideally, we would like to chose a higher probability threshold, but doing this results in a lower population of instances, specifically in the biased zone (high $P(y_i = 1|\boldsymbol{w}, \boldsymbol{x})$, and $P(\hat{y}_i = 0|y_i = 1)$).

Table 6.5: Results over Simulations

	$\theta = 0.03$, changed labels = 17%			
	Ground Truth	Biased	WLR	DEM
Accuracy	92.89 ± 0.82	88.75 ± 1.13	91.42 ± 0.82	92.98 ± 0.9
AUC	0.981 ± 0.002	0.977 ± 0.005	0.978 ± 0.005	0.981 ± 0.003
L	0.073 ± 0.032	0.087 ± 0.035	0.073 ± 0.031	0.076 ± 0.034
L data	0.075 ± 0.038	0.218 ± 0.031	-	-
	$\theta = 0.05$, changed labels = 25%			
	Ground Truth	Biased	WLR	DEM
Accuracy	92.49 ± 0.77	78.27 ± 1.76	87.06 ± 0.83	92.22 ± 0.62
AUC	0.979 ± 0.003	0.974 ± 0.003	0.975 ± 0.002	0.979 ± 0.003
L	0.055 ± 0.018	0.204 ± 0.037	0.109 ± 0.032	0.056 ± 0.017
L data	0.056 ± 0.02	0.293 ± 0.027	-	-
	$\theta = 0.1$, changed labels = 35%			
	Ground Truth	Biased	WLR	DEM
Accuracy	93.44 ± 0.83	51.41 ± 2.77	72.88 ± 2.06	93.35 ± 0.9
AUC	0.983 ± 0.003	0.977 ± 0.005	0.976 ± 0.006	0.983 ± 0.003
L	0.072 ± 0.032	0.45 ± 0.041	0.23 ± 0.042	0.069 ± 0.031
L data	0.061 ± 0.023	0.343 ± 0.033	-	-

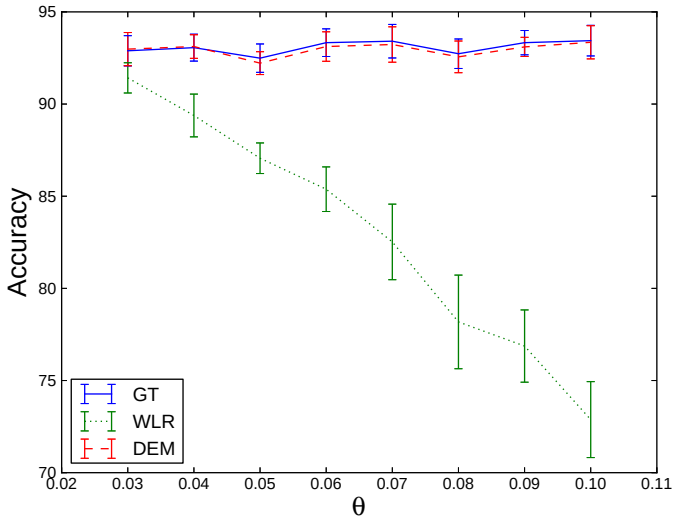


Figure 6.14: Accuracy for training over the GT, and over biased labels using WLR and DEM.

We used the Sérsic parameters as our classification attributes. These Sérsic parameters where obtained from fits by Simard et al. (2011). As our biasing parameter, we used the angular radii of the galaxies over the PSF of the instrument. This way, we take into account the resolution of the telescope and the galaxy apparent size. There are different ways of calculating galaxy radii. We used the Petrosian radius as calculated by the SDSS.

When developing our framework, we assumed that the bias is independent of the feature vector. As we chose the galaxy radius as our biasing parameter, our method does not allow to use R_e for classification. For this reason, we classified our galaxies using only n as our feature.

Figure 6.16 shows the fraction of $y = 1$ labels (spiral galaxies) in terms of α (the angular radius of the galaxies). It can be seen that even though the fraction of original labels, have already been de-biased by the Galaxy Zoo team, a strong bias is still present. Our proposed EM method is able to reduce the bias from $L = 0.18$ to $L = 0.05$, this is around a 72% better. As expected, as α increases, the fractions are similar to the biased sample.

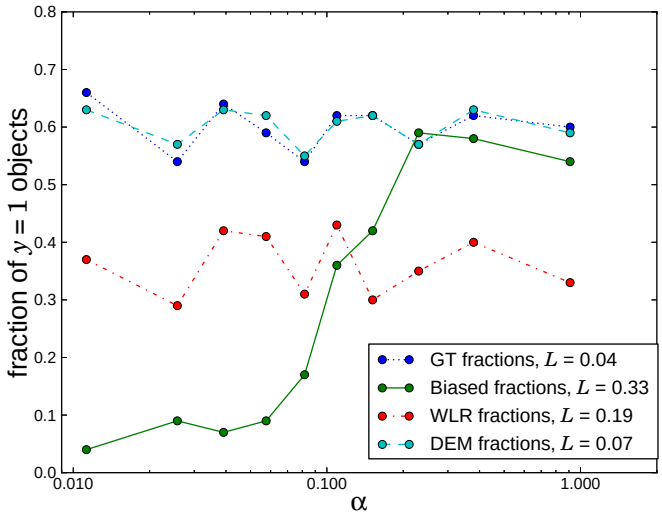


Figure 6.15: Fraction of $y = 1$ labels in terms of α for simulations using $\theta = 0.1$.

6.5 Conclusions

We have developed two methods for supervised learning in the presence of biased labels. Both frameworks can be conceptually applied to any data with bias as described in Chapter 5. We show empirical results when applying our methods over morphological galaxy classification.

Our first approach uses the bias metric described in Chapter 5 to determine an un-biased sub-set which is used for training a classification model. We show that when using the three parameters that describe the Sérsic profile we achieve around a 94% accuracy over galaxy morphology classification, while obtaining the lower amount of bias as compared against labeled data by experts, non-experts, and machine learning models. We also test how the bias in the training set correlates with the bias in the classification model. We find that, though there is a correlation between these biases, it is not statistically significant when using the Sérsic parameters as our classification attributes. We conclude that the Sérsic model by itself is able to reduce the bias of the model independent of the bias of the training set.

Our second approach uses a latent variable model to simultaneously fit a classification and bias model to the data. We use an expectation-maximization method to obtain the expected value of the ground truth labels. We show over simulations that as the bias increases, our method is able to maintain a high accuracy (as compared with the ground truth latent labels) and a low bias, outperforming current state of the art de-noising algorithms. We also show that our method is able to reduce the galaxy morphology bias to around a 30%.

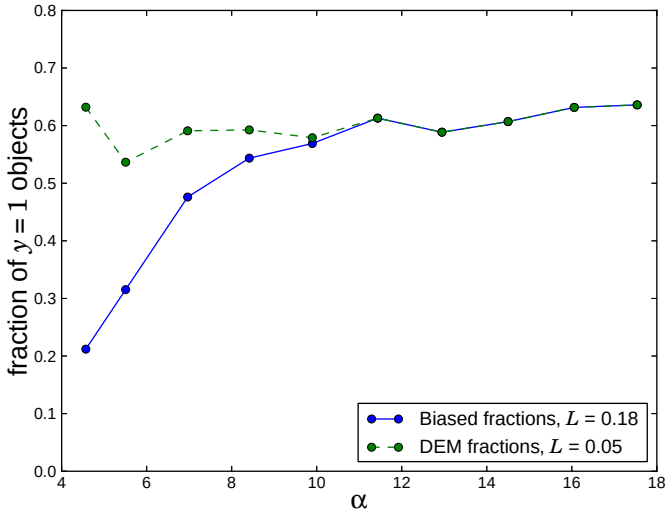


Figure 6.16: Fraction of $y = 1$ labels in terms of α = angular radius for Galaxy Zoo data. The blue line is the fraction of spirals in term of the Petrosian radius for the GZ labels. The green line is the fraction of spirals in term of galaxy radii for our de-biased labels.

Chapter 7

Conclusions

The goal of this thesis was to develop automated methods for astronomical object extraction and classification. This problem has become increasingly important in the era of big-data survey telescopes. In this Chapter we summarize what we have been able to achieve, and what is left to continue developing this field.

7.1 Results Obtained in this Thesis

We developed methods for extracting and classifying objects for astronomical images. For this purpose we consider PSF convolution and noise when analyzing big survey astronomical images.

We created a two-step iterative method for detection and extraction. In each iteration we filter the image with a PSF-shape low pass filter, and search for the maximum in that image. At that coordinate, we calculate elliptical contour levels (or isophotes) and do a 1D fit over those isophotes of a 2D PSF-convolved Sérsic profile over its semi-major axis, then subtract such profile from the image and run the following iteration.

Regarding our Sérsic fits, when running our method over simulations we obtain good fits in terms of their reduced χ^2 (1.01 ± 0.01), but errors between a 40%-50% in their Sérsic parameters. We conclude this is a degeneracy problem of the PSF-convolved Sérsic profile: different parameters reproduce similar light profiles. We also compare our fits over real objects against fits in the literature achieving similar rates of agreement.

In terms of classification, we address the problem of eyeball labels to be biased in terms of observable parameters: the human eye makes mistakes for low resolution, dim, and distant sources. We created two metrics to address this problem. Our first approach uses the deviation of the fraction of objects of each class from the expected

fraction in terms of the observable parameters. We expect the fraction of objects not to vary in terms of the observables. Our second approach extends this metric by considering the case where the fractions depend on intrinsic parameters. For example, for the case of galaxies, the fraction of spiral or elliptical galaxies will depend on their physical radius and intrinsic brightness. We calculate the bias for human labels, and for classification models in the literature. We show that even labels created by experts show a degree of bias, while machine learning models such as SVM are able to get a lower bias, even training over biased labels.

We propose two methods to address the label bias problem. The first one uses the metrics defined above to search for the least biased sub-set by discarding small, dim, and distant sources. This un-biased sub-set is used to train a supervised classification model. We show that when using the Sérsic model as our training features, though there is a correlation between the bias in the training set and the bias of the trained model, it is not statistically significant. This suggests that even in the presence of labeling bias, morphologies cluster in terms of the Sérsic parameters, which helps removing the bias when training. For our second approach, we used a probabilistic model for the bias and the learning model, and used an expectation-maximization method to obtain the expected latent ground truth labels. Through simulations we show that our approach outperforms current state-of-the-art label de-noising algorithms. We also show that when applying it over eyeball labels from Galaxy Zoo we are able to reduce the bias to a 30% of the original one.

7.2 Future Work

We have discussed lines of future work throughout the thesis. In this section we compile them.

1. **Detection / Extraction:** Though we have shown our method works correctly for detecting and extracting Sérsic light profiles, we have not studied a criteria for stopping the detection iterations. In that sense, our method will fit a profile to every local maxima in the low pass filtered image. Future lines of work for this include selecting a model fitting criteria such as the reduced χ^2 or the Bayesian information criterion to determine when to stop. At the same time, the detection procedure has not been evaluated in terms of the number of false positives or false negatives obtained. Though we compared our results for the Sérsic model fits against literature fits over real data, we have not compared them against state-of-the-art 2D fitting routines over simulations. Our algorithm is also missing a way of extracting galaxy non-smooth features such as spiral arms. Also, we have not assessed how well our algorithm works for deblending. We are aware of all

these issues, and we plan to work on them in the future.

2. **Classification in the presence of biased labels:** For the problem of classification in the presence of biased labels, we tried only learning models based on the Sérsic parameters, which, as we showed, gives a low bias model independent of the bias in the training set. We plan to assess if this results holds when using as many features as possible (including the observable parameters). We expect to obtain better accuracies when compared to the biased labels, but a more biased model in terms of our bias metric.

3. **Bias Model:** When simultaneously fitting the bias and classification model we made some assumptions. The strongest assumption we made is that the ground truth latent labels are independent of the observable parameters. This means, for example that the spiral/elliptical galaxies relative distribution will not depend on observables such as the distance of the galaxies from us. We also assumed that the features used for classification are independent of the observable parameters. This restricts our model to a smaller sub-set of features than the ones we could use. Finally, another assumption we made was that there is no selection bias in terms of the observables. For the case of image sources classification this is not important, but, for other data-sets this is an issue to consider. For example, when classifying light curves, we would like to use the signal-to-noise ratio as a biasing parameter, but some variable objects have lower intrinsic brightness, so there is a selection bias in terms of those objects having lower SNR. We are planning to relax these assumptions in the future.

4. **Multi-class Debiasing:** We have extensively tested our de-biasing algorithms over binary labeled data-sets. For the case of the latent variable bias model, turning the bias term into a multi-class bias is not straightforward. When using a one vs one bias model (the probability of being observed as one class given its ground truth label is from another specific class) there is a difficulty to sort: biases may be correlated. On the other hand, we could use a one versus the rest approach, were we define the bias as the probability of getting the observed label wrong. We have not addressed this problem in this thesis, but plan on doing it in the future.

5. **More Sophisticated Simulations:** In the last couple of months, the LSST image simulator has been made publicly available, making more realistic simulations than the ones we made. We plan to use this simulator in order to assess how our methods work on more realistic data. Furthermore, we can be able to simulate the labeling bias by selecting high resolution images and run them through the simulator reducing their brightness and resolution.

Appendix A

Acronyms

ADU	Analogue-to-Digital conversion Unit
ALMA	Atacama Large Millimeter/submillimeter Array
ANN	Artificial Neural Networks
AUC	Area Under the Curve
BIC	Bayesian Information Criterion
CCD	Charged-Coupled Device
CMM	Center for Mathematical Modeling
CTIO	Cerro Tololo Inter-American Observatory
DECam	Dark Energy Camera
DEM	De-biasing Expectation-Maximization algorithm
DES	Dark Energy Survey
E-ELT	European Extremely Large Telescope
EM	Expectation-Maximization
FFT	Fast Fourier Transform
FN	False Negative
FP	False Positive
FPR	False Positive Rate
FWHM	Full Width Half Max
GMM	Gaussian Mixture Model
GMT	Giant Magellan Telescope
GT	Ground Truth
GZ	Galaxy Zoo
HST	Hubble Space Telescope
LR	Logistic Regression
LSST	Large Synoptic Survey Telescope

ML	Maximum-Likelihood
MVM	Multiscale Vision Mode
NLHPC	National Laboratory for High Performance Computing
OvO	One versus One
OvR	One versus the Rest
PCA	Principle Component Analysis
PSF	Point Spread Function
ROC	Receiver Operating Characteristic
SDSS	Sloan Digital Sky Survey
SE	Source Extractor
SKA	Square Kilometer Array
SNR	Signal to Noise Ratio
SOM	Self-Organizing Map
SVM	Support Vector Machine
TN	True Negative
TP	True Positive
TPR	True Positive Rate
WLR	Weighted Logistic Regression
ZEST	Zurich Estimator of Structural Type

Appendix B

Support Vector Machine Training

The idea behind support vector machines (SVM, Vapnik (1982)) is to divide the attributes space into two by a hyperplane, defined by

$$h(\boldsymbol{x}) \equiv \boldsymbol{b}\boldsymbol{x} + b_0 = 0, \tag{B.1}$$

For each input object, SVM decides its class by finding which side of the hyperplane its attribute vector falls in. In order to develop easier equations, we will define the binary classes as $y \in \{-1, 1\}$. The values for $\boldsymbol{b}$ and b_0 are calculated such that the chosen hyperplane maximizes the distance from it to the closest vectors. We used the *soft margin* modification created by Cortes and Vapnik (1995), which allows training data that cannot be separated without error by introducing a "cost" parameter C. Parameters $\boldsymbol{b}$ and b_0 are computed by solving the optimization problem

$$\min_{\boldsymbol{b}, b_0} \frac{1}{2}||\boldsymbol{b}||^2 + C\sum_i^N \xi_i \tag{B.2}$$

$$\text{subject to } \xi_i \geq 0, y_i(\boldsymbol{x}^\top \boldsymbol{b} + b_0) \leq 1 - \xi_i, \forall i. \tag{B.3}$$

The associated Lagrange primal objective function for this optimization problem is

$$L_{\mathrm{P}} = \frac{1}{2}||\boldsymbol{b}||^2 + C\sum_i^N \xi_i - \sum_i^N \lambda_i^1 \xi_i - \sum_i^N \lambda_i^2 [y_i(\boldsymbol{x}^\top \boldsymbol{b} + b_0) - (1 - \xi_i)], \tag{B.4}$$

where λ_i^1 and λ_i^2 are Lagrange multipliers. We optimize L_{P} with respect to $\boldsymbol{\beta}$, β_0, and ξ_i by setting the derivatives to zero obtaining

$$\boldsymbol{b} = \sum_i^N \lambda_i^1 y_i \boldsymbol{x}_i, \tag{B.5}$$

$$\sum_i^N \lambda_i^1 y_i = 0, \tag{B.6}$$

$$\lambda_i^1 = C - \lambda_i^2, \tag{B.7}$$

with the positivity constraints λ_i^1, λ_i^2, $\xi_i \geq 0$. By substituting Equations B.5, B.6, and B.7 we obtain the Lagrangian dual objective function:

$$L_{\mathrm{D}} = \sum_{i=1}^{N} \lambda_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{i'=1}^{N} \lambda_i \lambda_{i'} y_i y_{i'} \boldsymbol{x}_i^\top \boldsymbol{x}_{i'}, \tag{B.8}$$

where, for simplicity of exposure, we have defined $\lambda_i \equiv \lambda_i^1$. L_{D} is maximized subject to $0 \leq \lambda_i \leq C$ and $\sum_i^N \lambda_i^1 y_i = 0$. This is a simpler convex quadratic problem than the primal, and can be solved with standard techniques by including the Karush-Kuhn-Tucker constraints

$$\lambda_i [y_i (\boldsymbol{x}_i^\top \boldsymbol{\beta} + \beta_0) - (1 - \xi_i)] = 0 \tag{B.9}$$

$$\lambda_i^2 \xi_i = 0, \tag{B.10}$$

$$y_i (\boldsymbol{x}_i^\top \boldsymbol{\beta} + \beta_0) - (1 - \xi_i) \geq 0. \tag{B.11}$$

The solution for $\boldsymbol{\beta}$ is given by Equation B.5. But not all observations i are taken into account. Equation B.9 and B.11 say that for all observations i that satisfy $y_i(\boldsymbol{x}_i^\top \boldsymbol{\beta} + \beta_0) - (1 - \xi_i) > 0$, then $\lambda_i = 0$. The observations with $\lambda_i = 0$ define the *support vectors*. The N_{SV} support vectors then define

$$\boldsymbol{b} = \sum_{i}^{N_{\mathrm{SV}}} \lambda_i y_i \boldsymbol{x}_i, \tag{B.12}$$

where i runs over the support vectors. At the same time, we can calculate β_0 from Equation B.9 for any of the margin points ($\xi_i = 0$, $\lambda_i > 0$ according to Equations B.10 and B.7). A usual practice is to average over all these margin points.

The above procedure uses a hyperplane to classify objects, but usually objects are not linearly separable. SVM can be used as a non-linear classification model by introducing a kernel function

$$K(\boldsymbol{x}, \boldsymbol{x}') = \langle h(\boldsymbol{x}), h(\boldsymbol{x}') \rangle \tag{B.13}$$

which computes the inner products in a transformed space defined by $h(\boldsymbol{x})$. It is not necessary to directly know the transformation $h(\boldsymbol{x})$ as long as we know the kernel function. Some examples of popular Kernel functions are:

linear: $K(\boldsymbol{x}, \boldsymbol{x}') = \langle \boldsymbol{x}, \boldsymbol{x}' \rangle$

dth degree polynomial: $K(\boldsymbol{x}, \boldsymbol{x}') = (1 + \langle \boldsymbol{x}, \boldsymbol{x}' \rangle)^d$

radial basis function: $K(\boldsymbol{x}, \boldsymbol{x}') = \exp(-\gamma \langle \boldsymbol{x}, \boldsymbol{x}' \rangle)$

sigmoid: $K(\boldsymbol{x}, \boldsymbol{x}') = \tanh(\gamma \langle \boldsymbol{x}, \boldsymbol{x}' \rangle + \rho)$

Using one of these kernels we can define our classification surface as

$$h(\boldsymbol{x}) = \sum_{i}^{N_{\mathrm{SV}}} \lambda_i y_i K(\boldsymbol{x}_i, \boldsymbol{x}) + b_0, \tag{B.14}$$

where i runs over the N_{sv} *support vectors*, which are the input vectors for which $\lambda_i \neq 0$. The SVM algorithm obtains for a training set the values for λ_i and b_0, but the values for C, γ, and ρ must be defined a priori. In these thesis, these parameters were chosen by griding on them and evaluating our SVM in terms of its accuracy.

Bibliography

A. P. Dempster, N. M. L. and Rubin, D. B. (1977). Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society. Series B*, 39(1):1–38.

Agresti, A. (1990). Categorical data analysis: Probability and mathematical statistics.

Bamford, S. P., Nichol, R. C., Baldry, I. K., Land, K., Lintott, C. J., Schawinski, K., Slosar, A., Szalay, A. S., Thomas, D., Torki, M., Andreescu, D., Edmondson, E. M., Miller, C. J., Murray, P., Raddick, M. J., and Vandenberg, J. (2009). Galaxy Zoo: the dependence of morphology and colour on environment. *Monthly Notices of the Royal Astronomical Society*, 393:1324–1352.

Beard, S. M., MacGillivray, H. T., and Thanisch, P. F. (1990). The Cosmos System for Crowded-Field Analysis of Digitized Photographic Plate Scans. *Monthly Notices of the Royal Astronomical Society*, 247:311–+.

Benedict, G. F. and Shelus, P. J. (1978). Applications of automated inventory techniques to astrometry. In F. V. Prochazka & R. H. Tucker, editor, *IAU Colloq. 48: Modern Astrometry*, pages 295–303.

Bertin, E. and Arnouts, S. (1996). SExtractor: Software for source extraction. *A&AS*, 117(2):393–404.

Bijaoui, A. and Rué, F. (1995). A multiscale vision model adapted to the astronomical images. *Signal Processing*, 46:345–2362.

Blanton, M. R., Dalcanton, J., Eisenstein, D., Loveday, J., Strauss, M. A., SubbaRao, M., Weinberg, D. H., Anderson, Jr., J. E., Annis, J., Bahcall, N. A., Bernardi, M., Brinkmann, J., Brunner, R. J., Burles, S., Carey, L., Castander, F. J., Connolly, A. J., Csabai, I., Doi, M., Finkbeiner, D., Friedman, S., Frieman, J. A., Fukugita, M., Gunn, J. E., Hennessy, G. S., Hindsley, R. B., Hogg, D. W., Ichikawa, T., Ivezić, Ž., Kent, S., Knapp, G. R., Lamb, D. Q., Leger, R. F., Long, D. C., Lupton, R. H., McKay, T. A., Meiksin, A., Merelli, A., Munn, J. A., Narayanan, V., Newcomb, M., Nichol, R. C., Okamura, S., Owen, R., Pier, J. R., Pope, A., Postman, M., Quinn, T., Rockosi, C. M., Schlegel, D. J., Schneider, D. P., Shimasaku, K., Siegmund,

W. A., Smee, S., Snir, Y., Stoughton, C., Stubbs, C., Szalay, A. S., Szokoly, G. P., Thakar, A. R., Tremonti, C., Tucker, D. L., Uomoto, A., Vanden Berk, D., Vogeley, M. S., Waddell, P., Yanny, B., Yasuda, N., and York, D. G. (2001). The Luminosity Function of Galaxies in SDSS Commissioning Data. *Astronomical Journal*, 121:2358–2380.

Blanton, M. R., Eisenstein, D., Hogg, D. W., Schlegel, D. J., and Brinkmann, J. (2005). Relationship between Environment and the Broadband Optical Properties of Galaxies in the Sloan Digital Sky Survey. *Astrophysical Journal*, 629:143–157.

Böhning, D. (1992). Multinomial logistic regression algorithm. *Annals of the Institute of Statistical Mathematics*, 44(1):197–200.

Boyle, W. S. and Smith, G. E. (1970). Charge Coupled Semiconductor Devices. *The Bell System Technical Journal*, 49:587–493.

Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5–32.

Breiman, L., Friedman, J., Stone, C. J., and Olshen, R. (1984). *Classification and Regression Trees*. Chapman and Hall/CRC.

Brodley, C. E. and Friedl, M. A. (1999). Identifying mislabeled training data. *Journal of Artificial Intelligence Research*, 11:131–167.

Buntine, W. (1992). Learning classification trees. *Statistics and Computing*, 2(2):63–73.

Buonanno, R., Buscema, G., Corsi, C., Ferraro, I., and Iannicola, G. (1983). Automated photographic photometry of stars in globular clusters. *Astronomy and Astrophysics*, 126:278–282.

Busko, I. (1996). Error Estimatioon in Elliptical Isophote Fitting. In *Astronomical Data Analysis Software and Systems V*, volume 101, page 139.

Butcher, H. (1977). A main-sequence luminosity function for the Large Magellanic Cloud. *Astrophysical Journal*, 216:372–380.

Carroll, B. W. and Ostlie, D. A. (2006). *An introduction to modern astrophysics and cosmology*, volume 1. Addison-Wesley.

Chiu, L.-T. G. (1976). Scales and Distortion Coefficients of the Lick, KPNO, and Hale Prime-Focus Correctors. *Publications of the Astronomical Society of the Pacific*, 88:803–+.

Ciotti, L. (1991). Stellar systems following the R exp 1/m luminosity law. *Astron. Asrophys.*, 249:99–106.

Cooley, J. and Tukey, J. (1965). An algorithm for the machine calculation of complex Fourier series. *Math. Comput*, 19(90):297–301.

Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20(3):273–297.

Corwin, H. G., Buta, R. J., and de Vaucouleurs, G. (1994). Corrections and additions to the third reference catalogue of bright galaxies. *The Astronomical Journal*, 108:2128.

Cover, T. and Hart, P. E. (1967). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1):21–27.

Dawid, A. P. and Skene, A. M. (1979). Maximum likelihood estimation of observer error-rates using the em algorithm. *Applied Statistics*, pages 20–28.

De Vaucouleurs, G., De Vaucouleurs, A., Corwin Jr, H., Buta, R., Paturel, G., and Fouque, P. (1992). Third reference catalogue of bright galaxies (rc3). *VizieR Online Data Catalog*, 7137:0.

de Vaucouleurs, Gerard (1959). Classification and Morphology of External Galaxies. *Handbuch der Physik*, 53.

de Vaucouleurs, Gerard, de Vaucouleurs, Antoinette, Corwin, Herold G., Jr., Buta, Ronald J., Paturel, Georges, and Fouque, Pascal (1991). Third Reference Catalogue of Bright Galaxies. *Volume 1-3*, 1.

Dewdney, P. E., Hall, P. J., Schilizzi, R. T., and Lazio, T. J. L. W. (2009). The Square Kilometer Array. *Proceedings of the IEEE*, 97(8):1482 – 1496.

du Plessis, M. C. and Sugiyama, M. (2012). Semi-supervised learning of class balance under class-prior change by distribution matching. In *Proceedings of the 29th International Conference on Machine Learning (ICML 2012)*, New York, NY, USA. ACM.

Duan, K.-B. and Keerthi, S. S. (2005). Which Is the Best Multiclass SVM Method? An Empirical Study. In *Multiple Classifier Systems*, pages 732–760. Springer.

Escoffier, R. P., Comoretto, G., Webber, J. C., Baudry, A., Broadwell, C. M., Greenberg, J. H., Treacy, R. R., Cais, P., Quertier, B., Camino, P., Bos, A., and Gunst, A. W. (2007). The ALMA correlator. *Astronomy and Astrophysics*, 462(2):801–810.

Friedman, J. H. (2002). Stochastic gradient boosting. *Computational Statistics & Data Analysis*, 38(4):367–378.

Fukugita, M., Nakamura, O., Okamura, S., Yasuda, N., Barentine, J. C., Brinkmann, J., Gunn, J. E., Harvanek, M., Ichikawa, T., Lupton, R. H., et al. (2007). A catalog of

morphologically classified galaxies from the sloan digital sky survey: north equatorial region. *The Astronomical Journal*, 134(2):579.

Gargiulo, F. and Sansone, C. (2010). Social: self-organizing classifier ensemble for adversarial learning. In *Multiple Classifier Systems*, pages 84–93. Springer.

Genel, S., Vogelsberger, M., Springel, V., Sijacki, D., Nelson, D., Snyder, G., Rodriguez-Gomez, V., Torrey, P., and Hernquist, L. (2014). The Illustris Simulation: the evolution of galaxy populations across cosmic time. *ArXiv e-prints*.

Green, R. F. and Morrill, M. E. (1978). An automated technique for stellar magnitude, color index, and position measurements of astronomical photographs. *Publications of the Astronomical Society of the Pacific*, 90:601–606.

Hastie, T., Tibshirani, R., Friedman, J., Hastie, T., Friedman, J., and Tibshirani, R. (2009). *The elements of statistical learning*, volume 2. Springer.

Herzog, A. D. and Illingworth, G. (1977). The Structure of Globular Clusters. I. Direct Plate Automated Reduction Techniques. *Astrophysical Journal Supplement Series*, 33:55–+.

Hsu, C.-W. and Lin, C.-J. (2002). A comparison of methods for multiclass support vector machines. *IEEE Transactions on Neural Networks*, 13(2):415–425.

Hubble, E. P. (1926). Extragalactic nebulae. *The Astrophysical Journal*, 64:321.

Huertas-Company, M., Aguerri, J. A. L., Bernardi, M., Mei, S., and Sánchez Almeida, J. (2011). Revisiting the Hubble sequence in the SDSS DR7 spectroscopic sample: a publicly available Bayesian automated classification. *Astronomy & Astrophysics*, 525:A157.

Huertas-Company, M., Kaviraj, S., Mei, S., O'Connell, R. W., Windhorst, R., Cohen, S. H., Hathi, . P., Koekemoer, A. M., Licitra, R., Raichoor, A., and Rutkowski, M. J. (2014). Measuring galaxy morphology at $z > 1$. I - calibration of automated proxies. *ArXiv e-prints*.

Hunter, J. D. (2007). Matplotlib: A 2d graphics environment. *Computing In Science & Engineering*, 9(3):90–95.

Ilbert, O., Capak, P., Salvato, M., Aussel, H., McCracken, H. J., Sanders, D. B., Scoville, N., Kartaltepe, J., Arnouts, S., Le Floc'h, E., Mobasher, B., Taniguchi, Y., Lamareille, F., Leauthaud, A., Sasaki, S., Thompson, D., Zamojski, M., Zamorani, G., Bardelli, S., Bolzonella, M., Bongiorno, A., Brusa, M., Caputi, K. I., Carollo, C. M., Contini, T., Cook, R., Coppa, G., Cucciati, O., de la Torre, S., de Ravel, L., Franzetti, P., Garilli, B., Hasinger, G., Iovino, A., Kampczyk, P., Kneib, J.-P., Knobel, C., Kovac, K., Le Borgne, J. F., Le Brun, V., Fèvre, O. L., Lilly, S., Looper,

D., Maier, C., Mainieri, V., Mellier, Y., Mignoli, M., Murayama, T., Pellò, R., Peng, Y., Pérez-Montero, E., Renzini, A., Ricciardelli, E., Schiminovich, D., Scodeggio, M., Shioya, Y., Silverman, J., Surace, J., Tanaka, M., Tasca, L., Tresse, L., Vergani, D., and Zucca, E. (2009). Cosmos Photometric Redshifts with 30-Bands for 2-deg^2. *Astrophysical Journal*, 690:1236–1249.

Ivezic, Z., Tyson, J. A., Acosta, E., Allsman, R., Anderson, S. F., Andrew, J., Angel, R., Axelrod, T., Barr, J. D., Becker, A. C., Becla, J., Beldica, C., Blandford, R. D., Bloom, J. S., Borne, K., Brandt, W. N., Brown, M. E., Bullock, J. S., Burke, D. L., Chandrasekharan, S., Chesley, S., Claver, C. F., Connolly, A., Cook, K. H., Cooray, A., Covey, K. R., Cribbs, C., Cutri, R., Daues, G., Delgado, F., Ferguson, H., Gawiser, E., Geary, J. C., Gee, P., Geha, M., Gibson, R. R., Gilmore, D. K., Gressler, W. J., Hogan, C., Huffer, M. E., Jacoby, S. H., Jain, B., Jernigan, J. G., Jones, R. L., Juric, M., Kahn, S. M., Kalirai, J. S., Kantor, J. P., Kessler, R., Kirkby, D., Knox, L., Krabbendam, V. L., Krughoff, S., Kulkarni, S., Lambert, R., Levine, D., Liang, M., Lim, K.-T., Lupton, R. H., Marshall, P., Marshall, S., May, M., Miller, M., Mills, D. J., Monet, D. G., Neill, D. R., Nordby, M., O'Connor, P., Oliver, J., Olivier, S. S., Olsen, K., Owen, R. E., Peterson, J. R., Petry, C. E., Pierfederici, F., Pietrowicz, S., Pike, R., Pinto, P. A., Plante, R., Radeka, V., Rasmussen, A., Ridgway, S. T., Rosing, W., Saha, A., Schalk, T. L., Schindler, R. H., Schneider, D. P., Schumacher, G., Sebag, J., Seppala, L. G., Shipsey, I., Silvestri, N., Smith, J. A., Smith, R. C., Strauss, M. A., Stubbs, C. W., Sweeney, D., Szalay, A., Thaler, J. J., Berk, D. V., Walkowicz, L., Warner, M., Willman, B., Wittman, D., Wolff, S. C., Wood-Vasey, W. M., Yoachim, P., Zhan, H., and Collaboration, f. t. L. (2008). LSST: from Science Drivers to Reference Design and Anticipated Data Products. page 34.

Jarrett, T. H. (2000). Near-Infrared Galaxy Morphology Atlas. *Publications of the Astronomical Society of the Pacific*, 112:1008–1080.

Jarvis, J. F. and Tyson, J. A. (1981). FOCAS - Faint Object Classification and Analysis System. *Astronomical Journal*, 86:476–495.

Kass, G. (1980). An exploratory technique for investigating large quantities of categorical data. *Applied statistics*, 29:119–127.

Kent, S. M. (1983). CCD photometry of the center of M31. *Astrophysical Journal*, 266:562–567.

Kibblewhite, E. J., Bridgeland, M. T., Hooley, T., and Horne, D. (1975). The Design of the New S.R.C. Automated Photographic Measuring System. In C. de Jager & H. Nieuwenhuijzen, editor, *Image Processing Techniques in Astronomy*, volume 54 of *Astrophysics and Space Science Library*, pages 245–+.

Koekemoer, A. M., Aussel, H., Calzetti, D., Capak, P., Giavalisco, M., Kneib, J.-P., Leauthaud, A., Le Fèvre, O., McCracken, H. J., Massey, R., Mobasher, B., Rhodes, J., Scoville, N., and Shopbell, P. L. (2007). The COSMOS Survey: Hubble Space Telescope Advanced Camera for Surveys Observations and Data Processing. *Astrophysical Journal Supplement Series*, 172:196–202.

Kohonen, T. (1990). The self-organizing map. *Proceedings of the IEEE*.

Lemeshow, S. and Hosmer, D. W. (1982). A review of goodness of fit statistics for use in the development of logistic regression models. *American journal of epidemiology*, 115(1):92–106.

Lintott, C., Schawinski, K., Bamford, S., Slosar, A., Land, K., Thomas, D., Edmondson, E., Masters, K., Nichol, R. C., Raddick, M. J., Szalay, A., Andreescu, D., Murray, P., and Vandenberg, J. (2011). Galaxy Zoo 1: data release of morphological classifications for nearly 900,000 galaxies. *Monthly Notices of the Royal Astronomical Society*, 410:166–178.

Lintott, C. J., Schawinski, K., Slosar, A., Land, K., Bamford, S., Thomas, D., Raddick, M. J., Nichol, R. C., Szalay, A., Andreescu, D., Murray, P., and Vandenberg, J. (2008). Galaxy Zoo: morphologies derived from visual inspection of galaxies from the Sloan Digital Sky Survey . *Monthly Notices of the Royal Astronomical Society*, 389(3):1179–1189.

Lorre, J. J., Benton, W. D., and Elliott, D. A. (1979). Recent developments at JPL in the application of image processing to astronomy. In D. L. Crawford, editor, *Instrumentation in Astronomy III*, volume 172 of *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, pages 394–402.

Lupton, R. H. and Gunn, J. E. (1986). M13 - Main sequence photometry and the mass function. *Astronomical Journal*, 91:317–325.

Maddox, S. J., Efstathiou, G., Sutherland, W. J., and Loveday, J. (1990). The APM galaxy survey. I - APM measurements and star-galaxy separation. *Monthly Notices of the Royal Astronomical Society*, 243:692–712.

Mallat, S. (1989). Multifrequency channel decompositions of images and wavelet models. *IEEE Trans. Acoust. Speech Signal Process*, 37(12):2091–2110.

McCullagh, P. (1980). Regression models for ordinal data. *Journal of the royal statistical society. Series B (Methodological)*, pages 109–142.

McCulloch, W. S. and Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, 5(4):115–133.

McIntosh, D. H., Bell, E. F., Rix, H.-W., Wolf, C., Heymans, C., Peng, C. Y., Somerville, R. S., Barden, M., Beckwith, S. V. W., Borch, A., Caldwell, J. A. R., Häußler, B., Jahnke, K., Jogee, S., Meisenheimer, K., Sánchez, S. F., and Wisotzki, L. (2005). The Evolution of Early-Type Red Galaxies with the GEMS Survey: Luminosity-Size and Stellar Mass-Size Relations Since z=1. *Astrophysical Journal*, 632:191–209.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953). Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21(6):1087–1092.

Millsap, R. E. and Everson, H. T. (1993). Methodology review: Statistical approaches for assessing measurement bias. *Applied Psychological Measurement*, 17(4):297–334.

Mohr, J. J., Adams, D., Barkhouse, W., Beldica, C., Bertin, E., Cai, Y. D., da Costa, L. A. N., Darnell, J. A., Daues, G. E., Jarvis, M., Gower, M., Lin, H., Martelli, L., Neilsen, E., Ngeow, C.-C., Ogando, R. L. C., Parga, A., Sheldon, E., Tucker, D., Kuropatkin, N., and Stoughton, C. (2008). The Dark Energy Survey data management system. In *Proceedings of SPIE*, volume 7016, pages 70160L–70160L–16. SPIE.

Moret, B. M. E. (1982). Decision Trees and Diagrams. *ACM Computing Surveys*, 14(4):593–623.

Morlet, J., Arens, G., Fourgeau, E., and Giard, D. (1982). Wave propagation and sampling theory—part i and ii: Sampling theory and complex waves. *Geophysics*, 47(2):203–236.

Murthy, S. K. (1998). Automatic Construction of Decision Trees from Data: A Multi-Disciplinary Survey. *Data Mining and Knowledge Discovery*, 2(4):345–389.

Naim, A., Ratnatunga, K. U., and Griffiths, R. E. (1997a). Galaxy Morphology without Classification: Self Organizing Maps. *The Astrophysical Journal Supplement Series*, 111(2):357–367.

Naim, A., Ratnatunga, K. U., and Griffiths, R. E. (1997b). Quantitative Morphology of Moderate Redshift Galaxies: How Many Peculiar Galaxies Are There? *The Astrophysical Journal*, 476(2):510–520.

Odewahn, S. C., Cohen, S. H., Windhorst, R. A., and Philip, N. S. (2002). Automated Galaxy Morphology: A Fourier Approach. *The Astrophysical Journal*, 568(2):539–557.

Oemler, Jr., A. (1976). The structure of elliptical and cD galaxies. *Astrophysical Journal*, 209:693–709.

Oza, N. C. (2004). Aveboost2: Boosting for noisy data. In Roli, F., Kittler, J., and Windeatt, T., editors, *Fifth International Workshop on Multiple Classifier Systems*, pages 31–40, Cagliari, Italy. Springer-Verlag.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Peng, C. Y., Ho, L. C., Impey, C. D., and Rix, H.-W. (2002). Detailed Structural Decomposition of Galaxy Images. *Astrophysical Journal*, 124:266–293.

Penny, A. J. and Dickens, R. J. (1986). CCD photometry of the globular cluster NGC 6752. *Monthly Notices of the Royal Astronomical Society*, 220:845–867.

Pratt, N. M., Martin, R., Alexander, L. W. G., Walker, G. S., and Williams, P. R. (1975). The Cosmos Facility at the Royal Observatory Edinburgh. In C. de Jager & H. Nieuwenhuijzen, editor, *Image Processing Techniques in Astronomy*, volume 54 of *Astrophysics and Space Science Library*, pages 217–+.

Press, W. H. (2007). *Numerical recipes 3rd edition: The art of scientific computing*. Cambridge university press.

Quinlan, J. R. (1986). Induction of Decision Trees. *MACH. LEARN*, pages 81 – 106.

Quinlan, J. R. (1992). *C4.5: Programs for Machine Learning (Morgan Kaufmann Series in Machine Learning)*. Morgan Kaufmann.

Rasmussen, C. E. and Williams, C. K. I. (2005). *Gaussian Processes for Machine Learning*. The MIT Press.

Raykar, V. C., Yu, S., Zhao, L. H., Jerebko, A., Florin, C., Valadez, G. H., Bogoni, L., and Moy, L. (2009). Supervised learning from multiple experts: Whom to trust when everyone lies a bit. In *Proceedings of the 26th Annual International Conference on Machine Learning (ICML 2009)*, pages 889–896, New York, NY, USA. ACM.

Raykar, V. C., Yu, S., Zhao, L. H., Valadez, G. H., Florin, C., Bogoni, L., and Moy, L. (2010). Learning from crowds. *J. Mach. Learn. Res.*, 11:1297–1322.

Rebbapragada, U. and Brodley, C. E. (2007). Class noise mitigation through instance weighting. In *Machine Learning: ECML 2007*, pages 708–715. Springer.

Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review,*, 65(6):386–408.

Saerens, M., Patrice, M., and Decaestecker, C. (2001). Adjusting the outputs of a classifier to new a priori probabilities: A simple procedure. *Neural Computation*, 14:21–41.

Sargent, M. T., Carollo, C. M., Lilly, S. J., Scarlata, C., Feldmann, R., Kampczyk, P., Koekemoer, A. M., Scoville, N., Kneib, J.-P., Leauthaud, A., Massey, R., Rhodes, J., Tasca, L. A. M., Capak, P., McCracken, H. J., Porciani, C., Renzini, A., Taniguchi, Y., Thompson, D. J., and Sheth, K. (2007). The Evolution of the Number Density of Large Disk Galaxies in COSMOS. *Astrophysical Journal Supplement Series*, 172:434–455.

Scarlata, C., Carollo, C. M., Lilly, S., Sargent, M. T., Feldmann, R., Kampczyk, P., Porciani, C., Koekemoer, A., Scoville, N., Kneib, J.-P., Leauthaud, A., Massey, R., Rhodes, J., Tasca, L., Capak, P., Maier, C., McCracken, H. J., Mobasher, B., Renzini, A., Taniguchi, Y., Thompson, D., Sheth, K., Ajiki, M., Aussel, H., Murayama, T., Sanders, D. B., Sasaki, S., Shioya, Y., and Takahashi, M. (2007). COSMOS Morphological Classification with the Zurich Estimator of Structural Types (ZEST) and the Evolution Since z = 1 of the Luminosity Function of Early, Disk, and Irregular Galaxies. *Astrophysical Journal Supplement Series*, 172:406–433.

Schwarz, G. et al. (1978). Estimating the dimension of a model. *The annals of statistics*, 6(2):461–464.

Sersic, J. L. (1968). *Atlas de galaxias australes*. Córdoba: Obs. Astronómico.

Sevilla,I., Armstrong,R., Bertin,E., Carlson,A., Daues,G., Desai,S., Gower,M., Gruendl,R., Hanlon,W., Jarvis,M., Kessler,R., Kuropatkin,N., Lin,H., Marriner,J., Mohr,J., Petravick,D., Sheldon,E., Swanson,M.E.C., Tomashek,T., Tucker,D., Yang,Y., Yanny,B., and fortheDESCollaboration (2011). The Dark Energy Survey Data Management System. *eprint arXiv:1109.6741*.

Simard, L. (1998). GIM2D: an IRAF package for the Quantitative Morphology Analysis of Distant Galaxies. In R. Albrecht, R. N. Hook, & H. A. Bushouse, editor, *Astronomical Data Analysis Software and Systems VII*, volume 145 of *Astronomical Society of the Pacific Conference Series*, pages 108–+.

Simard, L., Mendel, J. T., Patton, D. R., Ellison, S. L., and McConnachie, A. W. (2011). A Catalog of Bulge+disk Decompositions and Updated Photometry for 1.12 Million Galaxies in the Sloan Digital Sky Survey. *The Astrophysical Journal Supplement*, 196:11.

Simpson, E., Roberts, S., Psorakis, I., and Smith, A. (2013). Dynamic bayesian combination of multiple imperfect classifiers. In *Decision Making and Imperfection*, pages 1–35. Springer.

Skrutskie, M. F., Cutri, R. M., Stiening, R., Weinberg, M. D., Schneider, S., Carpenter, J. M., Beichman, C., Capps, R., Chester, T., Elias, J., Huchra, J., Liebert, J., Lonsdale, C., Monet, D. G., Price, S., Seitzer, P., Jarrett, T., Kirkpatrick, J. D., Gizis, J. E., Howard, E., Evans, T., Fowler, J., Fullmer, L., Hurt, R., Light, R., Kopan, E. L., Marsh, K. A., McCallon, H. L., Tam, R., Van Dyk, S., and Wheelock, S. (2006). The Two Micron All Sky Survey (2MASS). *Astronomical Journal*, 131:1163–1183.

Starck, J.-L. and Murtagh, F. (2006). *Astronomical Image and Data Reduction*. Springer.

Stetson, P. B. (1987). DAOPHOT: A computer program for crowded-field stellar photometry. *Pub. A. S. P.*, 99:191–222.

Tan, P., Steinbach, M., and Kumar, V. (2005). *Introduction to data mining*. Addison Wesley, 1st edition edition.

Taylor, M. B. (2005). TOPCAT & STIL: Starlink Table/VOTable Processing Software. In Shopbell, P., Britton, M., and Ebert, R., editors, *Astronomical Data Analysis Software and Systems XIV*, volume 347 of *Astronomical Society of the Pacific Conference Series*, page 29.

TheDarkEnergySurveyCollaboration (2005). The Dark Energy Survey. *eprint arXiv:astro-ph/0510346*.

Tody, D. (1981). Richfld photometry program users guide. *Kitt Peak National Obs., Tucson*.

Trujillo, I., Aguerri, J. A. L., Cepa, J., and Gutiérrez, C. M. (2001a). The effects of seeing on Sèrsic profiles. *Mon. Not. R. Astron. Soc.*, 321:269–276.

Trujillo, I., Aguerri, J. A. L., Cepa, J., and Gutiérrez, C. M. (2001b). The effects of seeing on Sérsic profiles - II. The Moffat PSF. *Monthly Notices of the Royal Astronomical Society*, 328:977–985.

Vapnik, V. N. (1982). *Estimation of Dependences Based on Empirical Data*. New York: Springer- Verlag.

Wells, D. (1975). KPNO-CTIO Quart. Bull. pages 12–18.

Wicenec, A., Gerstmann, D. K., Harris, C., and Vinsen, K. (2011). Integrating HPC into Radio-Astronomical data reduction. In *2011 XXXth URSI General Assembly and Scientific Symposium*, pages 1–1. IEEE.

Yee, H. K. C. (1991). A faint-galaxy photometry and image-analysis system. *Publications of the Astronomical Society of the Pacific*, 103:396–411.

Young, P. J., Kristian, J., Westphal, J. A., and Sargent, W. L. W. (1979). CCD photometry of the nuclei of three supergiant elliptical galaxies - Evidence for a supermassive object in the center of the radio galaxy NGC 6251. *Astrophysical Journal*, 234:76–85.

Zacharias, N., Finch, C., Girard, T., Hambly, N., Wycoff, G., Zacharias, M. I., Castillo, D., Corbin, T., Divittorio, M., Dutta, S., Gaume, R., Gauss, S., Germain, M., Hall, D., Hartkopf, W., Hsu, D., Holdenried, E., Makarov, V., Martinez, M., Mason, B., Monet, D., Rafferty, T., Rhodes, A., Siemers, T., Smith, D., Tilleman, T., Urban, S., Wieder, G., Winter, L., and Young, A. (2009). Third U.S. Naval Observatory CCD Astrograph Catalog (UCAC3). *VizieR Online Data Catalog*, 1315:0.

Zhang, G. (2000). Neural networks for classification: a survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 30(4):451–462.

Zhu, X., Wu, X., and Chen, Q. (2003). Eliminating class noise in large datasets. In *ICML*, volume 3, pages 920–927.