Università degli studi di Napoli Federico II

FACOLTÀ DI SCIENZE MATEMATICHE, FISICHE E NATURALI

Corso di Laurea in Astrofisica e Scienze dello Spazio

STraDiWA: a simulation environment for astronomical transient discovery

Relatori

Prof. Giuseppe Longo

Prof. Massimo Brescia

Candidato

Marianna Annunziatella

matr. N91/11

Anno Accademico 2011/2012

Contents

Introduction

Over the past decades, advances in technology are progressively moving us beyond the traditional observational paradigm in which most astronomical studies were made of individual observations of a small sample of objects, usually in a narrow wavelength range, opening the era of multi-epoch digital sky surveys. Nowadays, the advent of a new generation of digital detectors and dedicated surveys has opened the era of digital surveys and transformed astronomy in a data-driven science, where wavelength, multi epoch, high accuracy data are routinely collected for billions of objects. Now and at least for another decade astronomical surveys can be divided in two main types: wide-field or deep-field surveys. Wide-field surveys are usually shallower and cover a large area of the sky, while deep surveys cover smaller areas but go to a much fainter level of signal. This two types of survey will merge in 2018 when the Large Synoptic Sky Survey Telescope (LSST) will produce deep survey of large portion of the Northern Hemisphere. Surveys can be motivated by various scientific goals. They can be used for statistical studies such as the Galactic structure or the Large-Scale Structure in the universe or they can be designed for the research of particular types of objects like Supernovae, high redshift quasar etc. The most famous example of wide-field survey is the SDSS (Sloan Digital Sky Survey) (Gunn et al. 1998, York et al. 1994, Fukugita et al. 1996). The SDSS is an international collaboration which uses a dedicated 2.5 m telescope, located at the Apache Point Observatory in New Mexico, which covered (together with its extension SDSS-II and SDSS-III) an area f $\sim 14,500\,\mathrm{deg}^2$ of the sky. It has two observing modes, imaging and spectroscopy. The imaging survey uses 5 passbands, *ugriz*, with limiting magnitudes of 22.0, 22.2, 22.2, 21.3, and 20.5 mag, respectively.
Among the wide-field surveys a special place is reserved, in recent years, to synoptic surveys, which repeatedly observe the same regions of the sky with a sampling rate sensitive to astronomical phenomena that change over time.

The first synoptic surveys were OGLE (Optical Gravitational Lensing Experiment)[1] and MACHO (Massive Compact Halo Object)[2] projects. Both projects started in 1992 and while OGLE is still running, MACHO ended in 1999. Both were devoted to detecting microlensing events. Microlensing phenomena occurs when a massive compact halo object (macho) lie across the line of sight between the Earth and a distant more luminous star. The presence of the halo causes a temporary brightening of the light of the star. To detect these kind of phenomena, these surveys had to monitor hundreds of millions of stellar sources, and for this reason these projects have allowed the construction of some of the largest catalogs of variable star currently available.
Of the first project we have already had three phases, OGLE-I (1992-1995), OGLE-II (1996-2000), and OGLE-III (2001-2009), while a fourth is still in progress. The main targets of the experiment are the Galactic Bulge, the constellation Carina and toward both the Large Magellanic Cloud and Small the Magellanic Cloud. Details on the first phase of OGLE can be read in Udalski et al. 1992.
The MACHO project used the 1.23m telescope at Mt. Stromlo. The main target was the Large Magellanic Cloud but it also observed the Galactic bulge and the Small Magellanic Cloud, with a total target field of about 90 square degrees. Details on the project can be found in Alcock et al. 1993.
Microlensing searches have triggered the development of hardware and software capable of handling tens of millions of photometric measurements every clear night, archive the data, and perform real time recognition of the very rare microlensing events and the distribution of alerts to all interested observers. MACHO and OGLE have paved the way to the wide variety of projects on sky variability actually running.
There are several synoptic surveys already operating, while even more ambitious projects will begin in the coming years. The most important transient surveys currently working are the Catalina Real Time Transient Survey (CRTS)[3], the Palomar Transient Factory (PTF)[4]. To these should be added several Supernova and asteroids surveys.
The CRTS (Drake et al. 2008, Drake et al. 2012), started in November 2007, uses three wide-field telescopes: the 0.68 m Schmidt at Catalina Station, Arizona, the 0.5 m Uppsala Schmidt at Siding Spring Observatory, Australia, and a 1.5 m reflector located on Mt. Lemmon, Arizona. It actually covers $\sim$ 33.000 deg^2 of the sky in the declination range $-30° < \delta < 70°$. Obser-

[1] http://ogle.astrouw.edu.pl/
[2] http://www.macho.anu.edu.au/
[3] http://crts.caltech.edu/
[4] http://www.astro.caltech.edu/ptf/index.php

vations exclude the Galactic plane within $|b| < 10° - 15°$. The sampling rate is 4 images of the same field, separated by ∼10 min, per night. The observable sky from Arizona and Australia is covered every few weeks with a few exposures.
The PTF (Rau et al. 2009, Law et al. 2009), uses the Samuel Oschin telescope (the 48-inch Schmidt) in the Palomar Observatory. The total area coverage is $\sim 15.000\, deg^2$ in the declination range $-25° < \delta < +25°$. Each night are taken 2 exposures of a given field.
Both projects collect data streams of $\sim$ **0.1 TB /night** and detect $\sim 10-10^2$ transient per night. Projects such as Pan-STARRS(Panoramic Survey Telescope and Rapid Response System) [5], VISTA (Visible and Infrared Survey Telescope for Astronomy)[6], and VST(VLT Survey Telescope)[7] have recently started and collect data streams of $\sim$ **1 TB/night**, with a detection rate of $\sim 10^4$ transient/ night.
Pan-STARRS is a planned array of astronomical cameras and small mirror telescopes and computing facility that will survey the sky visible from Hawaii. It will use four 1.8 m telescopes that will be located either at Mauna Kea or Haleakala in Hawaii. The first telescope prototype, PS1, is already operating.
VISTA is a 4.1m survey telescope, located in the Cerro Paranal Observatory, which has the goal of performing extensive surveys of the Southern skies. There are six large public surveys being conducted with VISTA, which cover different areas of sky at different depths in order to tackle a wide range of scientific problems. One of the surveys will cover the entire Southern hemisphere of the sky.
VST is an alt-azimuthal wide-field survey telescope with a primary mirror diameter of 2.65m. It is the largest telescope in the world designed to exclusively survey the sky in visible light. The VST program is a cooperation between the Osservatorio Astronomico di Capodimonte (OAC), Naples, Italy, and the European Southern Observatory (ESO) that began in 1997. The telescope has only recently started observations and it is already collecting data for a variety of survey projects such as KIDS, VST Voice.
Finally, as already mentioned, in the forthcoming years, as already mentioned, the LSST (Large Synoptic Survey Telescope) [8] will begin to work. LSST (Tyson 2003, Ivezic et al. 2009) is a wide-field telescope that will be located at Cerro Paranal in Chile. LSST will take more than 800 panoramic images each night, with 2 exposures per field, covering the accessible sky twice

[5] http://pan-starrs.ifa.hawaii.edu/public
[6] http://www.vista.ac.uk
[7] http://www.eso.org/public/telesinstr/surveytelescopes/vst/surveys.html
[8] http://www.lsst.org

each week. The total survey area will include $30.000\,\mathrm{deg}^2$ with $\delta < +34.5°$ and will be imaged multiple times in six bands, ugrizy, covering the wavelength range 320-1050 nm. The data reduction of LSST and the detection of transients will move us into the Petascale regime with a detection rate of $\sim 10^5 - 10^6$ transients event per night.
Astronomical variable phenomena, which are the main targets of synoptic surveys, can be divided in two classes:

- Astrometric transients: i.e., objects which change their position in the sky more or less rapidly such as trans Neptunian objects, comets or asteroids.
- Photometric transients: i.e. objects with variable luminosities.

As it will be better discussed in what follows, photometric transients are very useful in many, if not all, fields of astronomy. However, before being useful for science, the survey data need to be processed and understood. Synoptic surveys have to face two major problems: detection and physical classification of the transients. Scope of the classification is to assign at every given event the probability that it belongs to a known class of astrophysical phenomena, in order to guarantee and optimize follow-up observations required for certainly type of variable objects (e.g. short lived transients). The process must be as near real-time as possible, ensure a hight completeness and yet a low contamination. A brief summary of the classification algorithms already developed is given in Chapter 2.
There are many problems to take into account for the classification of variable objects, e.g. how to characterize variable objects (light curves, other statistical indicators), which knowledge base has to be used (base built on the data themselves, or rather on simulated ones), how to solve the computational challenge, how to find the unknown (by throwing away all the known or by searching for intrinsic partitions of the Parameter Space).
This thesis is part of the project STraDiWA (Sky Transient Discovery Web Application), included in the DAME[9] Collaboration, finalized to testing and implementing algorithms based on the machine learning paradigms (Brescia 2012b), for variable objects classification in order to find the one optimized for each type of variable object.
DAME (Data Mining & Exploration) is a collaboration between University Federico II in Naples, the Astronomical observatory of Capodimonte in Naples INAF-OACN, and the California Institute of Technology finalized at implementing an infrastructure for data analysis, exploration, mining and

[9] http://dame.dsf.unina.it/.

visualization tools. The framework makes use of distributed computing environments, available also through the project S.C.o.P.E of the University Federico II[10].
In our opinion the best way to test these classifiers is through templates images (or catalogs). In fact real data are not always suitable since they are often incomplete. A transient, in fact, detected by an increase in brightness is often missing in archival sky surveys and may have just a couple of relatively closely spaced observations in a couple of epochs to go by.
The project includes an automatic workflow to generate astronomical images with an user-defined number and type of variable objects, in order to perform setup and calibration of classification models running on the real images coming from observations.
The original aspects of my work are:

- the design and implementation of the simulation environment, where each type of variable objects is simulated by an independent module;
- the implementation of two such modules, one for Classical Cepheids and the other for type Ia Supernovae;
- the development of a software pipeline for catalog extraction;
- the identification of the best classifier for transient objects by exploiting the machine learning classification models, made available within the DAME infrastructure.

The present thesis outcome has also to be considered as a scientific and technological proposal for the EUCLID space Mission Collaboration[11], to be included between its legacy science toolset.
The thesis is structured as follows.
In the first chapter we give an overview of the photometric transients both galactic and extragalactic, focusing in particular on pulsating stars and Supernovae.
In the second chapter we illustrate the workflow of our project and describe how we have implemented the simulations.
In the third chapter we investigate the problem of catalog extraction and analyze two software, comparing their performances.
In the fourth chapter we describe the algorithms tested for variable objects classification.
In the fifth chapter we summarize our results and report our conclusion, outlining some future developments.

[10] http://www.scope.unina.it/default.aspx
[11] http://www.euclid-ec.org/

Chapter 1

The Astronomical Parameter Space

All the observable quantities from every astronomical observation form the Observable Parameter Space (OPS). This space can be conveniently divided in four main domains.

- The spectrophotometric domain. This include the spectroscopic, the photometric and the polarization sub-domains. The main axes are the wavelength λ, the flux F, the spectroscopic resolution R= $\lambda/\Delta\lambda$ or the Stock parameters.
- The astrometric domain, whose axes include the pairs of coordinates, the astrometric accuracy $\Delta\theta$ and the area coverage Ω.
- The morphological domain, which includes the surface brightness μ and the angular resolution $\Delta\alpha$.
- The time domain, whose axes include the time of the observation (which can be expressed for example in Julian Dates), the time sampling Δt and the number of epochs N_{exp} obtained at each Δt.

Besides the four main domains, we must add the non-electromagnetic channels like neutrinos, gravitational waves or cosmic rays.

The Observable Parameter Space is a N-dimensional space, with N>>100 and steadily increasing, where N is the number of characteristics that can be defined for a given type of observation. Every single observation, as well as each survey, covers only finite portions of this space, due to its own instrumental limits. Some regions of the OPS may result better explored than others. For example visible, NIR (near infrared) or radio domains are better

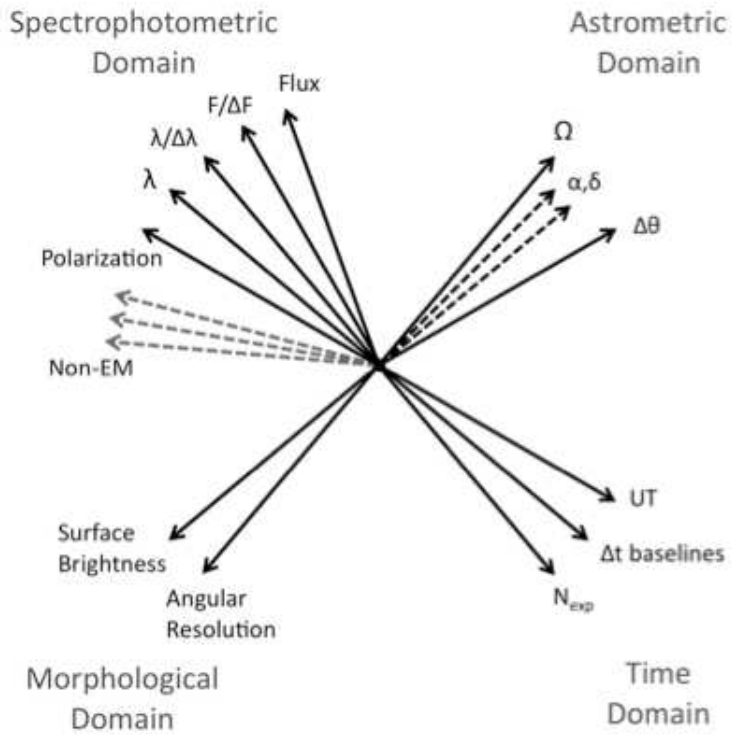


Figure 1.1. A schematic illustration of the Observable Parameter Space, credit to Djorgovski et al.

explored than X-ray and higher energies domains.
There are two more spaces similar to the OPS, the Measurement Parameter Space (MPS) and the Physical Parameter Space (PPS). The first space is constituted by the measured properties of the sources, like the flux, or by derived quantities like colors or surface brightness. Some of the axes of this space are represented by labels, instead of numbers, like the morphological structure (Star/Galaxy). For each detected source a typical survey can measure hundreds of parameters, with a correspondingly high dimensionality of the MPS.
The PPS instead is formed by the physical properties of the detected sources. Physical and measured properties are related by some additional knowledge. Some axes of the PPS are derived parameters like chemical abundances, masses, etc. While the MPS contains the observed properties of a detected source, the PPS contains the astronomical objects. Objects in PPS tend to form clusters and to leave other zones empty. There are many data mining techniques which perform clustering analysis of the PPS. Some of these algorithms have been developed by DAME.

The three spaces usually have common axes which represent distance independent quantities.
New discoveries are often made when we improve the sampling of already known regions, such as when a new wavelength range becomes available, or when we improve the spatial resolution of the instrument. The advent of the synoptic surveys has meant that a region of the OPS, the Time Domain, has been extensively explored in the recent years.

1.1 The Variable Sky

The exploration of the Time Domain allow us to study numerous types of astrophysical phenomena. Targets of Time Domain Astronomy are in fact all of those sources which show some kind of variability. As mentioned before imaging surveys allow us to find and observe observe astrometric and photometric variables. Astrometric variables, also defined transits, are objects whose position in the sky changes with time. Photometric variability instead verifies when a source show a change in brightness at different epochs. In this work we shall focus only on photometric variables.
Photometric variability takes place at any wavelength range. In this thesis, however, we shall focus only on those phenomena which have at least an optical manifestation.
A schematic representation of the different types of photometric optical transients is given in Figure 1.2.
Photometric variability can be both extrinsic or intrinsic.
Extrinsic variables show change in brightness due to the eclipse of one object by another or to the effect of rotation. They can be asteroids or stellar objects. Among the second group there are the microlensing events, the eclipsing binary systems or the Rotating stars. In an eclipsing system a star can change its brightness due to an asteroid occultation, to a planetary transit or to the interaction with another star. in the latter case we talk about Eclipsing Binaries. These systems are formed by physically bound stars having an orbital plane lying near the line-of-sight of the observer. The components periodically eclipse each another, causing a decrease in the apparent brightness of the system as seen by the observer. The period of the eclipse, can range from minutes to years. Rotating stars, instead, show small changes in light that may be due to dark or bright spots on their stellar surfaces.
In this thesis we focus on intrinsic variables. *Intrinsic variables* show brightness variations caused by changes in the physical parameters of the object.

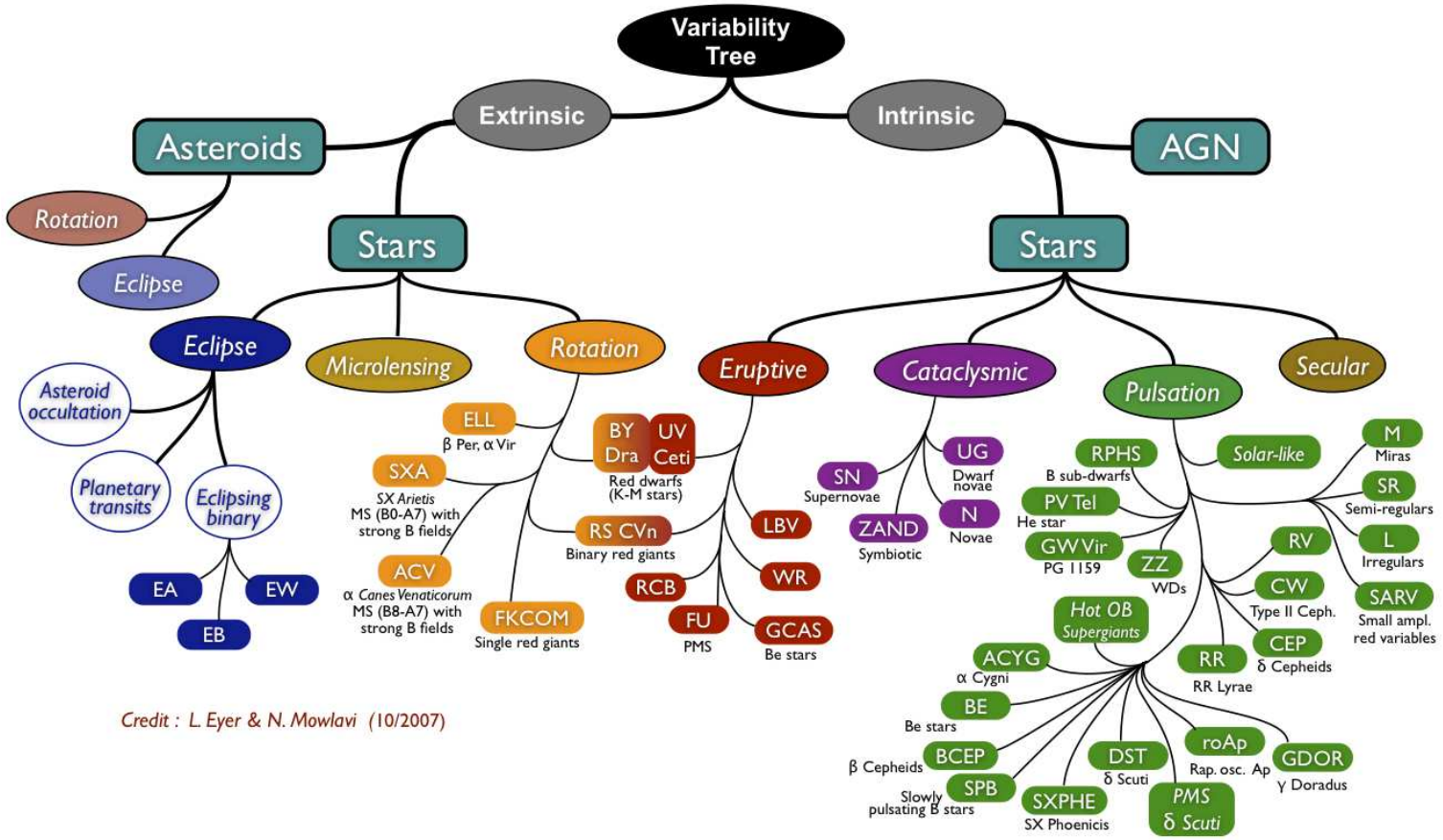


Figure 1.2. Semantic Tree of Astronomical Transient Objects, credit to Eyer & Mowlavi 2007.

This group can be divided in stellar objects, and galaxies.
Intrinsic variable stars are usually divided in three more classes, pulsating, cataclysmic, and eruptive variables depending on which phenomenon is at the origin of their variability. Another class is then formed by stars displaying secular evolution, which are usually stars in the post-AGB (Asymptotical Giant Branch) of the H-R (Hertzsprung-Russell) diagram.
Eruptive variable stars vary in brightness because of violent processes and flares occurring in their chromospheres and coronae. The light changes are usually accompanied by shell events or mass outflow in the form of stellar winds of variable intensity and/or by interaction with the surrounding interstellar medium. The most famous example of eruptive variables are the Wolf-Rayet and the R Coronae Borealis stars. A R Coronae Borealis variable is a luminous, hydrogen-poor, carbon-rich, supergiant star which spend most of its time at maximum light, occasionally fading even nine magnitudes at irregular intervals. Wolf-Rayet stars are very luminous hot Population I stars of effective temperatures between 30000 and 50000 K. They have a characteristic high mass-loss rate ($\sim 10^{-5} M_{\odot} yr^{-1}$). They show light variations with amplitudes of several hundredths of a magnitude and time scales from milliseconds to years.
Of the other two types of intrinsic stellar variables we shall discuss in detail in the next sections focusing in particular on the most representative object of each class, Cepheids for Pulsating variables and Supernovae for Cataclysmic variables.
Galaxies hosting Active Galactic Nuclei (AGNs) are also usually variable. AGNs, however, are very particular variables. In fact they emit strongly over a wide range of wavelengths, from X-ray to radio. Many AGNs vary in brightness by substantial amounts over timescales as short as, months, days, or even hours. AGNs are conveniently divided in two main classes called radio-loud and radio-quiet, depending on whether or not they emit in the radio portion of the electromagnetic spectrum.

1.2 Pulsating variables

A pulsating variable star is characterized by periodic variations of its luminosity. Stellar pulsations can be radial, if the stars expands with spherical symmetry, or non-radial and in this case the shape of the star can result asymmetrically distorted. Pulsations can occur at various frequencies. The lowest allowed frequency is called fundamental mode, and higher frequencies are called overtones. For each oscillation mode, these waves have at least

one node, where the matter remains steady, at the center of the star and an antinode, where the velocity of the gases is maximum, at the surface.

1.2.1 Theory of Stellar Pulsation

The principal categories of pulsating stars are observed to lay in a nearly vertical region of the H−R diagram called Instability Strip. The instability strip defines a range of luminosities, colors, and periods, over which pulsation is a stable mode for the star.

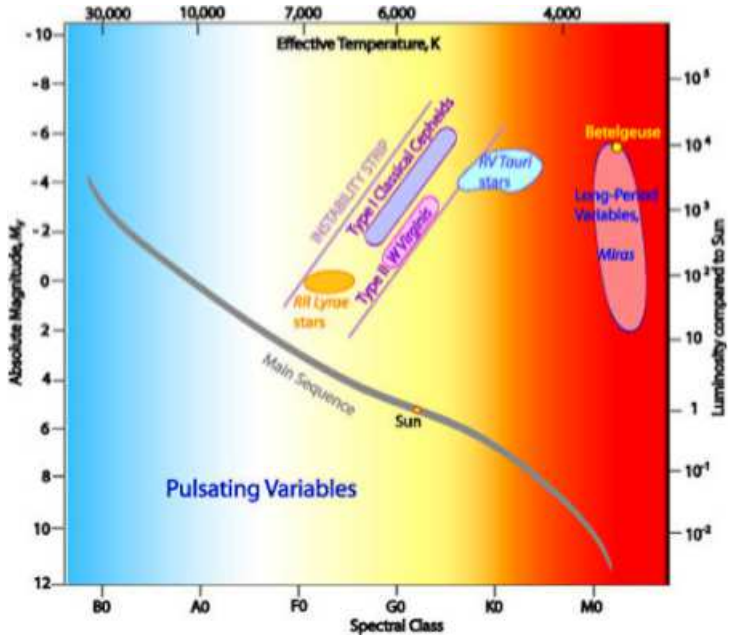


Figure 1.3. Position of some Pulsating Variables in the H−R diagram.

From the physical point of view, radial stellar pulsations can be studied as small perturbations around the hydrodynamical equilibrium state, which can grow to observed amplitudes. This theory is known as linear stability analysis of stellar structure.
To study the phenomenon of radial stellar pulsation we start from stellar structure equations:

$$\frac{\partial^2 r}{\partial^2 t} = -\frac{GM_r}{r^2} - \frac{\partial P}{\partial r}, \tag{1.2.1}$$

$$\frac{\partial r}{\partial M_r} = \frac{1}{4\pi r^2 \rho}, \tag{1.2.2}$$

$$\frac{\partial E}{\partial t} - \frac{P}{\rho^2}\frac{\partial \rho}{\partial t} = \epsilon - \frac{\partial L}{\partial M_r}, \tag{1.2.3}$$

$$L_r = -4\pi r^2 \frac{4ac}{3}\frac{T^3}{\kappa\rho}\frac{\partial T}{\partial r} = -\frac{64\pi^2 ac}{3} r^4 \frac{T^3}{\kappa}\frac{\partial T}{\partial M_r}. \tag{1.2.4}$$

The energy density ϵ and the opacity κ are functions of the density and the temperature. In an equilibrium state the Eq. 1.2.1 - 1.2.4 become:

$$\frac{\partial P_0}{\partial r_0} = \frac{GM_r}{r_0^4}, \tag{1.2.5}$$

$$\frac{\partial r_0}{\partial M_r} = \frac{1}{4\pi r_0^2 \rho_0}, \tag{1.2.6}$$

$$\frac{\partial L_{r0}}{\partial M_r} = \epsilon_0. \tag{1.2.7}$$

To solve the problem of stellar pulsation we can and express all the variables in Eq. 1.2.1-1.2.2 in terms of an equilibrium quantity and a small perturbation: $\mathrm{r} \to \mathrm{r}_0 + \delta\mathrm{r}$, $\mathrm{P} \to \mathrm{P}_0 + \delta\mathrm{P}$, $\rho \to \rho_0 + \delta\rho$, $\mathrm{L} \to \mathrm{L}_0 + \delta\mathrm{L}$. We can put $\zeta = \delta\mathrm{r}/\mathrm{r}_0$, so that
$\mathrm{r} = \mathrm{r}_0(1+\zeta)$, and furthermore we can write a generic Lagrangian quantity f as $f = f_0(1+\delta f/f_0)$. We assume that in case of small perturbation $|\zeta| << 1$ and $|\delta\mathrm{f}/\mathrm{f}_0| << 1$, and neglect all the terms of second and higher orders. With these assumptions, equations 1.2.1 - 1.2.4 are reduced to a single equation in ζ:

$$\begin{aligned}\frac{\partial^2 \zeta}{\partial t^2} = -\frac{1}{r\rho}\frac{\partial \zeta}{\partial t}\frac{d}{dr}\Big[(3\Gamma_1 - 4)P\Big] - \left(\frac{1}{\rho r^4}\right)\frac{\partial}{\partial r}\left(\Gamma_1 P r^4 \frac{\partial \zeta}{\partial r}\right) = \\ = \frac{1}{r\rho}\frac{\partial}{\partial r}\left[\rho(\Gamma_3 - 1)\delta\left(\epsilon - \frac{\partial L_r}{\partial M_r}\right)\right]\end{aligned} \tag{1.2.8}$$

where

$$\Gamma_1 = (dlnP/dln\rho)_{ad} \qquad \text{and} \qquad \Gamma_3 = (dlnT/dln\rho)_{ad}, \tag{1.2.9}$$

are the adiabatic exponents of pressure and temperature. We consider only solutions with the form:

$$\zeta(\mathrm{r}, \mathrm{t}) = \xi(\mathrm{r})\mathrm{e}^{\mathrm{i}\omega\mathrm{t}}, \tag{1.2.10}$$

where $\xi(\mathrm{r})$ is a complex function of the only spatial variable and ω is a

frequency.
In case of adiabatic oscillations the Eq. 1.2.8 become:

$$-\frac{1}{r^4\rho}\frac{d}{dr}\left[\Gamma_1 P r^4 \frac{d\xi}{dr}\right] - \frac{1}{r\rho}\left\{\frac{d}{dr}[(3\Gamma_1 - 4)P]\right\}\xi = \omega^2\xi. \qquad (1.2.11)$$

The solution of this equation requires the adoption of special conditions at the center and at the surface of the star. Eq. 1.2.11 is an eigenvalue equation which admits discrete solutions characterized by eigenfunctions $\zeta_{\rm k}$ and eigenvalues ω_k. Each eigenfunction ζ_k is characterized by k nodes, where $\zeta_{\rm k} = 0$. The frequency ω_0 is the fundamental mode, the higher frequencies are the overtones.
To sustain a pulsating motion, a driving mechanism must be present. If not, pulsations would be damped. For stars located in the Instability Strip this driving mechanism seems to be related to the opacity of the star. Eddington suggested that certain layers of the star, during its compression phase of pulsation, might become quite opaque to radiation.
The increase of opacity causes an accumulation of heat under these layers and eventually brought an increase of pressure that leads to the expansion of the star. At this point, the opacity of these layers decreases and permits the accumulated heat to flow out. Once the pressure has decreased, the star contracts again and a new cycle begins. In most regions of the stars, however, the opacity decreases with the compression. In fact the opacity depends on the density and temperature of the stellar material according to the Kramers law as $\kappa \propto \rho/T^{3/5}$. During the compression both the density and the temperature increase, however the opacity is much more sensitive to the temperature so that the it decreases in this phase of the pulsation. In 1980 J.P. Cox found that the regions of a star in which the mechanism proposed by Eddington can successfully operate are the partially ionization zones. In these layers, where the gas is partially ionized, a compression of the star will produce a further ionization rather than rising the temperature. With a small rise of the temperature, the increase of the density during the compression will produce a corresponding increase in the Kramers opacity. During the expansion, on the other end, there is a small increase of the temperature due to the recombination of the ions with the electron and meanwhile the opacity decreases with decreasing density. Pulsation for star in the Instability Strip is driven by two main ionization zones. The first one is a broad zone where there are both the ionization of the Hydrogen and a first ionization of the Helium and is called hydrogen partial ionization zone. The second one, the He II partial ionization zone, is deeper and involves the ionization of He II

in He III[1].

1.2.2 Types of Pulsating Variables

Types of pulsating variables may be identified on the basis of their pulsation period, mass and evolutionary status of the star, and the characteristics of their pulsations.

- RR Lyrae stars. These are short-period (0.05 to 1.2 days), pulsating, blue giant stars, usually of spectral class A. The amplitude of variation of RR Lyrae stars is generally from 0.3 to 2 magnitudes.

- δ Scuti. This variable stars exhibit variations in their luminosity due to both radial and non-radial pulsations of their surface. Typical brightness fluctuations are from 0.003 to 0.9 magnitudes in V over a period of a few hours, although the amplitude and period of the fluctuations can vary greatly. These stars are usually A0 to F5 type giant or main sequence stars.

- RV Tauri. These stars are yellow supergiants having a characteristic light variation with alternating deep and shallow minima. Their periods, defined as the interval between two deep minima, range from 30 to 150 days. The light variation can be up to 3 magnitudes. Some of these stars show long-term cyclic variations from hundreds to thousands of days. Generally, the spectral class ranges from G to K.

- Pulsating white dwarf. The luminosity of these white dwarf varies due to non-radial gravity wave pulsations. These variables all exhibit small (1%− 30%) variations in light output, arising from a superposition of vibration modes with periods of hundreds to thousands of seconds.

- Long Period Variables. These stars are pulsating red giants or supergiants in which variations in brightness occur over long timescales of months or years. The two major subclasses are Mira and Semiregular variables.

- Irregular Variable Star. These are usually red supergiants with little or no periodicity. They are often poorly studied semi-regular variables that, upon closer scrutiny, should be reclassified.

[1]Astronomers refer to H I as neutral atomic hydrogen and H II as ionized atomic hydrogen. This agreement applies also to other elements.

The main example of pulsating stars are, though, Cepheid variables. They are massive stars of spectral type changing during the pulsation and varying from F at maximum luminosity to a G or K at minimum. These stars are mostly radial pulsators. There are four classes of Cepheid variables:

- Classical Cepheids, or type I Cepheids, fundamental mode pulsators with periods vary form 1 to 70 days.
- Beat Cepheids, which display the presence of two or more simultaneously operating pulsation modes, usually the fundamental and the first overtone. The have periods between 2 and 7 days.
- *S* Cepheids, which are probably first-overtone pulsators, with periods in the same range of Beat Cepheids.
- W Virginis, population II Cepheids with periods between 1 and 30 days. These stars are fundamental mode pulsators.

Although Cepheids exhibit strong correlations between their periods, luminosities and colors, the amplitudes of Cepheids do not appear to correlate with other observables. Cepheids, as well as most of the other pulsating variables, exhibit periodic light curves with a sinusoidal form. An example of light curve for each type of Cepheids are reported in Fig. 1.4, Fig. 1.5, Fig. 1.6 and Fig. 1.7.All the light curves have been produced with the data available on the site of the AAVSO (American Association of Variable Star Observers)[2].

[2] http://www.aavso.org/

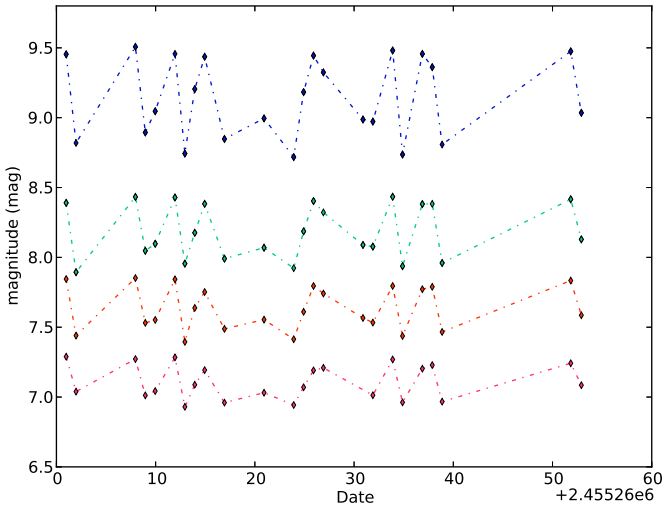


Figure 1.4. Pre-calibrated BVRI light curve for the Classical Cepheid SS Sct. On the y-axis there is the apparent magnitude of the star and on the y-axis the Julian date of the observation. Blu points are the values of the magnitude in B band. Green points are the values of the magnitude in B band. Blue points are the values of the magnitude in B band. Orange points are the values of the magnitude in R band. Red points are the values of the magnitude in I band.

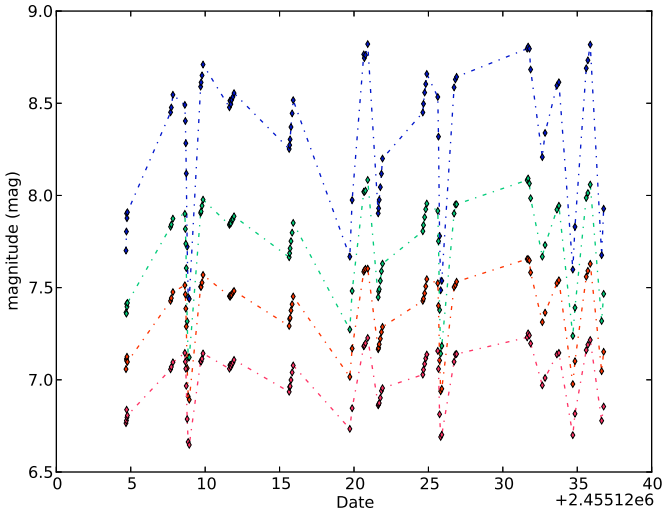


Figure 1.5. Pre-calibrated BVRI light curve for the Beat Cepheid TU Cas. On the y-axis there is the apparent magnitude of the star and on the y-axis the Julian date of the observation. Blue points are the values of the magnitude in B band. Green points are the values of the magnitude in B band. Blue points are the values of the magnitude in B band. Orange points are the values of the magnitude in R band. Red points are the values of the magnitude in I band.

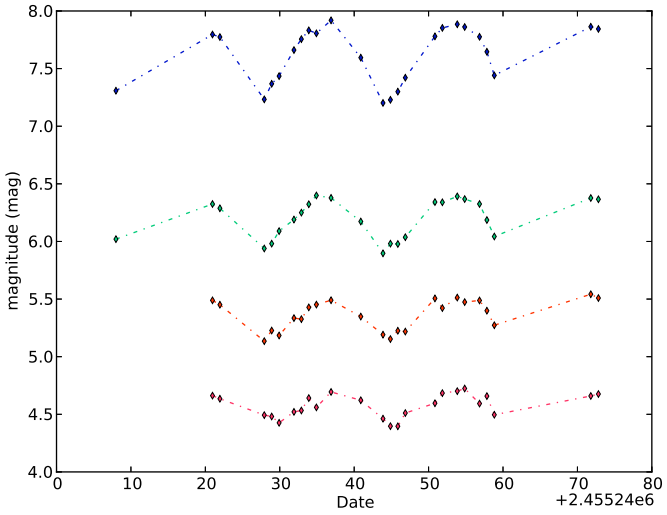


Figure 1.6. Pre-calibrated BVRI light curve for the *S* Cepheid Y Oph. On the y-axis there is the apparent magnitude of the star and on the y-axis the Julian date of the observation. Blue points are the values of the magnitude in B band. Green points are the values of the magnitude in B band. Blue points are the values of the magnitude in B band. Orange points are the values of the magnitude in R band. Red points are the values of the magnitude in I band.

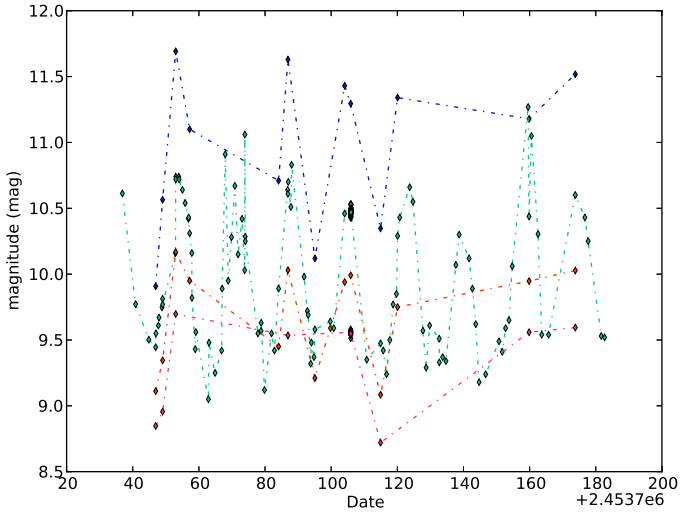


Figure 1.7. Pre-calibrated BVRI light curve for the prototype of W Virginis variables, W Vir. On the y-axis there is the apparent magnitude of the star and on the y-axis the Julian date of the observation. Blue points are the values of the magnitude in B band. Green points are the values of the magnitude in B band. Blue points are the values of the magnitude in B band. Orange points are the values of the magnitude in R band. Red points are the values of the magnitude in I band.

1.2.3 Period-Luminosity relation

In 1912 the American astronomer Henrietta Swan Leavitt found that for a sample of Classical Cepheids in the Large Magellanic Cloud there was a linear correlation between the apparent magnitude of the star and the logarithm of its period. Since all the Cepheids in the LMC, can be considered at the same distance from us, this relation is valid also for the absolute magnitude, up to a zeropoint magnitude. Leavitt's discovery is known as the "Period-Luminosity relation" and can be expressed as:

$$M = a + b * log_{10}P. \tag{1.2.12}$$

The original P-L relationship obtained by Leavitt is shown is Fig. 1.8.
Once it has been properly calibrated, the Period-Luminosity relation allow us to derive from the measured period of a Cepheid, its absolute magnitude

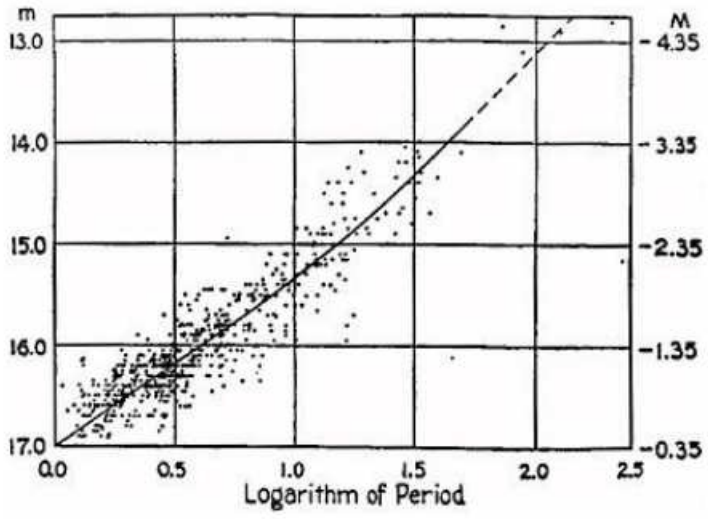


Figure 1.8. The first period-luminosity diagram for the Cepheids. This diagram shows Henrietta Leavitt's graph of data for the Small Magellanic Cloud. On the x-axis there is the Logarithm of the Period of the stars. On the y-axis on the left there is the average apparent magnitude of the variable as observed, on the right the absolute magnitude of the variable stars.

and therefore, via the comparison with the observed one, its distance module.

For completeness we shall just outline a few facts connected with the P-L relationships. The empirical calibration of Period-Luminosity relation presents however several issues. In particular the data have to be corrected for the effects of interstellar reddening. The presence of interstellar grains within our Galaxy, interposed between the observer and a nearby galaxy, or within the galaxy we are studying, adsorbs part of the light coming from a background star or galaxy.. Because of these extinction components, a Cepheid in an external galaxy will appear fainter and redder than it actually is. This will produce systematic errors which will be reported into the distance scale. There is not a single approach to solve the reddening problem in the calibration of the Period-Luminosity relation. However to minimize its effect there are two methods which are commonly adopted: 1) moving to the reddest wavelength allowed to reduce the extinction problem to the level of other systematic and random errors, 2) combining multiwavelength (visual to near-infrared) data for significant numbers of Cepheids in a given galaxy, and determining the averaged extinction using an independently calibrated wavelength-dependent extinction law.
Another problem to take into account in the calibration is the effect of the

metallicity on the Period-Luminosity relation. The main physical mechanisms which contribute to the effect of metallicity on the mean color of Cepheids is the presence of atmospheric metal-line blanketing. The effect of the metallicity is usually smaller at longer wavelengths.
Nowadays Cepheids are continuously studied. In fact the Period-luminosity relation has been calibrated in many ranges of wavelength, and for objects both in our Galaxy and in external galaxies, like the Large and Small Magellanic Clouds. An example of the various calibrations can be found in Tammann et al. (2003), Sandage et al. (2004), Sandage et al. (2008) for Cepheids both in our Galaxy, in LMC and SMC in the photometric bands B, V and I.

1.2.4 Physical basis of the PLC relation

In 1958 A. Sandage discovered a more general relation between luminosity, period and color of a Cepheid star (Sandage A. 1958). The empirical form of this relation is:

$$logP - 1.051(B-V) + 0.230\langle M_V\rangle = logQ + 0.588. \tag{1.2.13}$$

$\langle M_V\rangle$ is the V-band magnitude of the star, (B-V) its color and Q a structural constant. This relation can be approximately understood considering the pulsations of the star as the results of sound waves resonating in the star's interior. The adiabatic sound speed of these waves can be written as:

$$v_s = \sqrt{\frac{\gamma P}{\rho}}. \tag{1.2.14}$$

The pressure can be found from the hydrostatic equilibrium, under the assumption of constant density.

$$\frac{dP}{dr} = -\frac{GM_r\rho}{r^2} = -\frac{G(\frac{4}{3}\pi r^3)\rho}{r^2} = -\frac{4}{3}\pi G\rho^2 r. \tag{1.2.15}$$

The equation 1.2.15 can be integrated using boundary condition that $P = 0$ at the surface, to obtain the expression of the pressure as function of r.

$$P(r) = \frac{2}{3}\pi G\rho^2(R^2 - r^2). \tag{1.2.16}$$

Hence the pulsation period is roughly :

$$\Pi \approx \int_0^R \frac{dr}{v_s} \approx 2\int_0^R \frac{dr}{\sqrt{\frac{2}{3}\pi G\rho^2(R^2-r^2)}}, \tag{1.2.17}$$

or

$$\Pi \approx \sqrt{\frac{3\pi}{2\gamma G\rho}}. \tag{1.2.18}$$

The equation 1.2.18 is also known as Period-density density relation and it is valid for all the pulsating stars. We obtained:

$$\Pi \propto \rho^{-1/2} \propto R^{3/2}, \tag{1.2.19}$$

where R is the radius of the star. Taking into account the black-body relationship:

$$L = 4\pi\sigma R^2 T^4, \tag{1.2.20}$$

we have:

$$\Pi \propto L^{3/4}T^3. \tag{1.2.21}$$

Passing to logarithms this relation becomes linear. Note that in order to get to Eq. 1.2.21 we have not used any relation between the Mass and the Luminosity of the star, because Cepheids are stars in a post-sequence phase. The meaning of Eq. 1.2.21 is that the period of a Cepheid (or more in general of a pulsating star) is determined by luminosity and temperature. The corresponding observing quantities are the magnitude and the color Index. The PLC relation is an equation for a plane. Projection of this plane along each axis give the color-magnitude diagram, the period-color diagram and the period-luminosity diagram.

1.3 Cataclysmic variables: Supernovae

Cataclysmic variables are usually close binary stars in which the most massive component is usually a white dwarf and the companion is commonly a main sequence star. The majority of these systems steadily transfer mass from the companion to the white dwarf through a surrounding accretion disk. This accreted material powers symbiotic activity, including occasional eruptions and jets. Components of this class of objects are:

- Novae. These systems are composed by a white dwarf and a main-sequence low mass star. A classical nova shows an increase of brightness

from 7 to 15 magnitude in a range of 1 to several hundred days.

- Dwarf Novae. These consist of a white dwarf and a red dwarf star slightly cooler of our sun. They show semi-regular outbursts with a typical timescale ranging from weeks to years and a typical amplitude of 4-5 magnitudes.

- Symbiotic Stars. These systems are interacting binary stars composed of an evolved red giant and a hot companion star. The hot component can be a main sequence star, a white dwarf, or a neutron star. Most symbiotics have orbital periods of a few years; some systems orbit over several decades.

The most famous type of cataclysmic variables still remain the Supernovae (SNe).
With the term Supernova we refer to the catastrophic explosion occurring in the later stages of the life of a massive star. During these explosions a mass of $\sim 10 - 100 M_{\odot}$ is ejected with velocities of about 0.01-0.1c. The explosion commonly ejects heavy elements. The burst of radiation in a Supernova often briefly outshines the luminosity of the host galaxy, before fading from view over several weeks or months.
Supernovae are among the most spectacular celestial objects ever observed by humans. If close enough they can be seen even during the day. There have been only eight confirmed Supernovae observed in our Galaxy. Of these objects we have retained several reports. The oldest supernova recorded by humanity was the one observed in 185 AD. Supernovae in 386 and 393 AD are reported only in Chinese records with no precise information about their positions in the sky. The brightest Supernova ever seen was the one exploded in 1006 AD, which reached a visual magnitude of -7.5 mag. It was described by observers in China, Egypt, Iraq, Japan, Switzerland. However, the most famous Supernova is probably the one seen in 1054. This explosion produced the rapidly expanding shell of gas that is now identified as the Crab Nebula. Of this Supernova there are many non European records, the most careful of them by Chinese. We know that it was brighter than Venus ad that remained visible in daylight for 23 days. In the 1181 AD another Supernova was observed by Chinese and Japanese astronomers in the constellation Cassiopeia. Always in the constellation Cassiopeia a Supernova was observed by the Danish astronomer Tycho Brae in the 1572 AD. The last confirmed supernova exploded in our Galaxy was the one observed by Kepler in 1604.
All these Supernova have left behind them the so called Supernova Remnants. Since no Supernova has been observed in our Galaxy in the telescopic era, it

is clear that almost all we know about this phenomenon, has been derived from Supernovae in other galaxies.

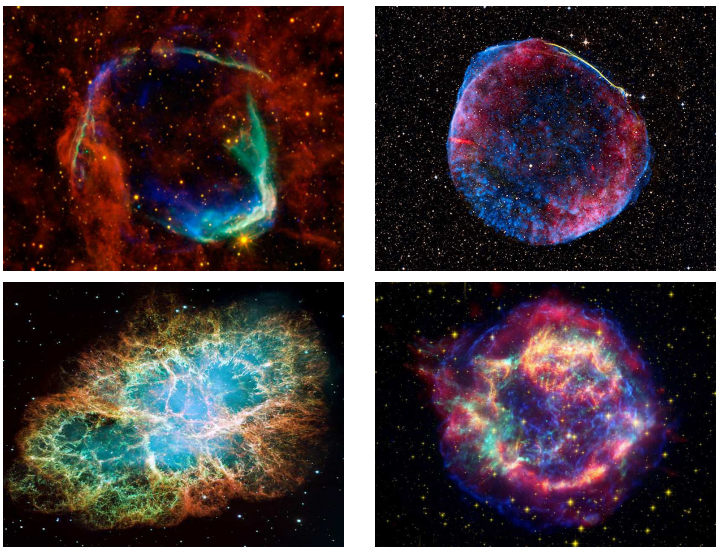


Figure 1.9. Remnants of the Supernovae SN 185, SN 1006, SN 1054, SN 1604.

1.3.1 Classification of Supernovae

Categories of Supernovae are traditionally defined by the features of their optical spectra near the maximum light and, at later stages, by the characteristics of their light curves. Supernovae were first categorized in 1941, by R. Minkowski, in two main types, type I and type II. The main difference between them being the lack of hydrogen emission line, H_α in type I Supernovae. Type Ia Supernovae are further divided in three sub classes: Type Ia, Ib and Ic, according to their spectral characteristics. Type Ia Supernovae show the absorption line of the Si IIλ6355, type Ib show, instead, the absorption line of He Iλ5876 together with emission lines of Oxygen and Calcium, while type Ic Supernovae do not show any of the previous adsorption lines. type II Supernovae are divided in two further categories based on the resulting light curve following the explosion. type II-L show a steady (Linear) decline after the maximum, whereas type II-P display a period of slower decline (a plateau) followed by a normal decay.
Type Ia Supernovae seem to be present in all kind of galaxies, that is ellipticals, spirals and irregulars. They show characteristic elements in their

spectrum, such as magnesium, silicon, sulphur, and calcium near maximum light and iron later on. Their presence in elliptical galaxies, where there is no evidence of stellar formation, means that their progenitors must be long-lived stars.
Type Ib and Ic Supernovae only seem to explode in the arms of spiral galaxies, that is instellar-formation zones. This indicates that their progenitors must be short-lived stars. The composition of these objects is similar to that expected in the core of a massive star that has been stripped of its hydrogen. In the case of Type Ic, most of the helium is gone as well.
Type II Supernovae occur mostly in stellar formation zones, like H II regions of Spiral' s disks, or in Irregular galaxies. Their progenitors are also short-lived stars, hence massive stars.

1.3.2 Supernovae Progenitors

Type Ia and the other types of Supernovae seem to have different progenitors. Type Ib, Ic and II, known as core-collapse Supernovae, are the product of the collapse of a massive, evolved stellar cores, while for type Ia Supernovae there are still different theories.

Core Collapse Supernovae

Type Ib, Ic and II Supernovae are the results of the collapse of different types of stars. Due to the presence of hydrogen in their spectra, the progenitors of type II Supernovae must be stars with masses between $8 - 40 M_{\odot}$. More massive stars, like Wolf-Rayet, loose their envelopes and can result in Supernovae Ib and Ic.
These stars arrive to the pre-Supernova phase, after they have passed the burning stages of hydrogen, helium, carbon, neon, oxygen, and silicon. The end result of the silicon burning stage is the production of an iron core. These process will leave the star with an onion-like structure, as schematized in Figure 1.10.
Because the nuclear binding energy per nucleon has its maximum value for the iron group, no further energy can be released by nuclear fusion. At the temperatures present in the iron cores, the photons have enough energy to destroy heavy nuclei. This process is known as photodisintegration. At this stage there are conditions critical enough that the free electrons that contribute to support the star through the degeneracy pressure, are captured by heavy elements and by protons produced by photodisintegration. At this point most of the core's support, in form of degeneracy pressure, is gone and

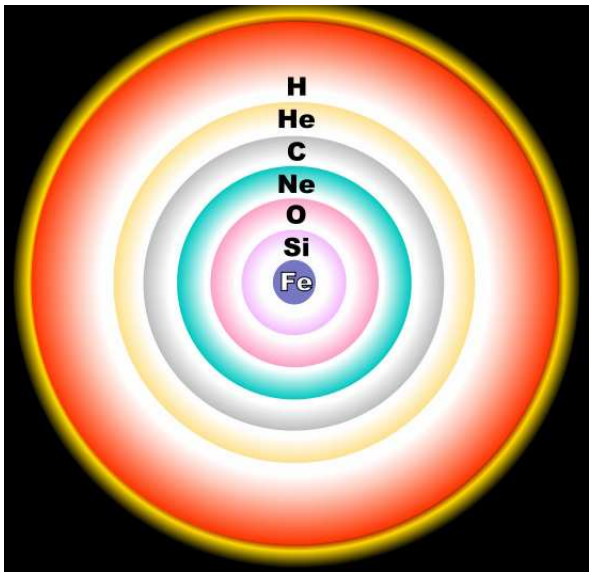


Figure 1.10. Onion-like structure of a star in a pre-Supernova phase.

it begins to collapse. When the collapsing core reaches approximately twice the density of atomic nuclei, $\sim 4 - 5 \times 10^{14} \mathrm{g\,cm^{-3}}$, the repulsive component of the short−range nuclear force halts the collapse. The abrupt halt of the collapse of the inner core will produce a rebound mechanism in which there will form shock waves towards the surface of the star. Not all the energy of these shocks waves can be used to expel the outer envelopes. In fact, a large amount of this energy is lost through neutrinos or is used for iron photodisintegration.
A possible mechanism which can lead to the Supernova explosion is the delayed mechanism. After that the shock waves will fade there will form a sphere of neutrinos with a density of $10^{11} \mathrm{g\,cm^{-3}}$ and a thermal energy of 5 MeV. At such density the opacity is so high that neither the neutrinos can escape. The stalled Supernova shock front will be outside the neutrinos sphere. These neutrinos can provide enough energy so that the stagnating shock can be revived and thus accelerates outward to propagate through the overlying, still collapsing layers of the star and lead to the Supernova explosion. A huge amount of energy is released and the outer layers, those containing calcium, oxygen, carbon, and helium, and any outer envelope of hydrogen are expelled.

Type Ia Supernovae

For type Ia Supernovae, there is no generally accepted picture of their evolutionary origin. The most reliable hypothesis is that these objects form in a binary system containing a carbon-oxygen white dwarf and an evolved star. We know that the accreting material on the white dwarf is the cause of the explosion of the Supernova. However, it is not very clear yet which mechanisms lead to this explosion.

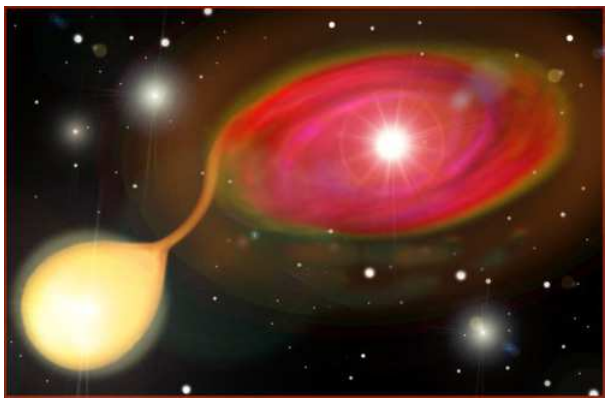


Figure 1.11. Accretion model of a binary system formed by a White Dwarf (upper right) and a companion (lower left).

In the most accredited model the companion star, during the red giant phase, will transfer material onto the white dwarf until it reaches the Chandrasekhar limit, where the degeneracy pressure is no longer able to support the star against the gravity. This causes the white dwarf to contract and subsequently its central temperature and density to rise enough to ignite carbon fusion.
Another model suggests, instead, that when the material from the companion star falls onto the white dwarf, the Helium in the gas will settles on its surface becoming degenerate. When enough Helium has accumulated, there will be the so called Helium flash. Not only this will cause the burning of helium itself on the top of the star but will send a shock wave downward the white dwarf causing the ignition of the degenerate carbon and oxygen.
What happen next is not well understood. If the shock wave produced by the explosion has enough energy to bring the fuel in the near layers above the ignition temperature, we will have a detonation waves which will propagate at supersonic speed. In this case the shock waves will compress and heat up the unburned material until it ignites. If the shock wave doesn't have enough

energy to ignite the fuel in the near layers there will produce a deflagration event which will occur at subsonic speed. In this case the ignition of the unburned material is due to heat transfer through diffusion or turbulent convection. With detonation events all the material will become ^{56}Ni, while in deflagration events part of the material will become ^{56}Ni, the rest in lighter elements, like Si and C. Since the type Ia show all the elements from Fe to C in their spectra the deflagration models are the most accredited one.

1.3.3 Type Ia Supernovae light curves

The light curves of all types of Supernovae, can all be explained by the energy released by the decay of the ^{56}Ni in ^{56}Co and subsequently in ^{56}Fe.
The type Ia light curves in optical and near-infrared bands have all similar shapes. We can identify four phases.

- Rise time. The SN rises to maximum very fast. Only in very lucky occasions have early observations been recorded.
- Maximum phase.
- Second Maximum. A pronounced second maximum has been observed in redder light curves about from 20 to 40 days after the first maximum.
- Late decline. After about 50 days the light curves settle onto a steady decline, which is exponential in luminosity.

The peak luminosity is directly linked to the amount of radioactive ^{56}Ni produced during the explosion. The rise time of the light curve is determined primarily by the explosion energy and by the manner in which the ejecta become optically thin to thermalized radiation, while the late decline of the light curve is governed by the combination of the energy input by the radioactive material and the rate at which this input energy is converted to optical photons in the ejecta.
An example of the light curve of the Supernova SN 1998bu in the M96 galaxy in the filters of the Johnson photometric system is reported below.

1.3.4 Supernovae as distance estimators

All types of Supernovae reach the maximum light 2-3 weeks after the explosion. The type I Supernovae are brighter of one magnitude than the type II. All the type Ia Supernovae have the same peak luminosity. If we can

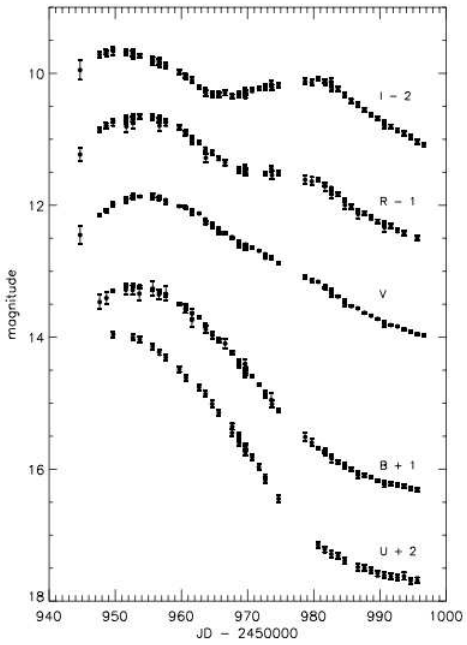


Figure 1.12. UBVRI light curves of SN 1998bu, credit to Jha et al. 1999. On the x-axis there is the Julian date of the observation and on the y-axis the magnitude of the Supernova. The light curves in U,B,R,I have been shifted to avoid overlapping.

measure the absolute magnitude of a Supernova at the maximum, regardless its distance, we can obtain a measure of the Hubble constant.
There have been several studies which demonstrate that the absolute magnitude of a Supernova Ia is related to the width of the light curve. Various studies have been carried out in the 70's to on the light curve shape vs. luminosity relations, but could not be supported by the available data. After acquiring a lot of high quality data in the early 1990s, the breakthrough came with the Phillips relation Phillips (1993) which is a linear relationship between the decline rate parameter Δm_{15}, that is the difference between the magnitude at maximum light and the magnitude after fifteen days, and the absolute peak magnitude of the Supernova. An improved form of this relation, in the Johnson band B, V and I, was given in a later paper, correcting the magnitudes for galaxy reddening. Employing the observed correlation between light curve shape and luminosity has improved the precision of distance estimates derived from SNe Ia significantly and has allowed using SNe Ia for the determination of cosmological parameters.
Given their high luminosity, the Supernovae Ia are powerful distance estimators even for far away objects. This method, however, implies that we can detect the Supernova near its explosion so that we can measure both the

magnitude peak and the decline rate.

Chapter 2

STraDiWA: a simulation environment for astronomical transient discovery

Modern approaches to classification of astrophysical variables can be divided in to time domain based and feature based methods.
The time domain approach makes use of theoretical or empirical models to determine the class of an astrophysical source. An example of this type of classification is the light curve fitting. For each new object we can fit empirical light curves to the data and the fits determine whether or not the object belong to a given class. This technique can be used by restricting to a limited number of cases. For example, assuming that a source is a certain type of supernova, a well sampled group of observed light curves can be used to to classify a new one.
Another type of classification is based on features, that is information derived from time series images and contextual data. The set of features represents a multidimensional space, id est an ideal working environment for machine learning algorithms. Features can be arbitrarily simple or complex. They can belong to time domain or can be contextual information. The most common time domain features are based on the distribution of detected fluxes and can be derived from the light curves of the sources, such as flux ratios, amplitude, skewness or significant frequencies (in case of periodic light curves). Contextual information are instead characteristics of the source which do not change with time, like object coordinates, distance from the nearest detected galaxy and the parameters available for that galaxy, etc. Contextual information are often difficult to include in the classification process, due to the heterogeneity of the data. However, they are useful to discriminate different types of objects (e.g. an event which can be a cataclysmic variable or a supernova is more likely a supernova if it is close to a galaxy). Classification

methods based on features can be either supervised or unsupervised depending on whether or not we have a training sample of labeled data. The former ones aim to assign to a given source its class (or class probabilities), while the latter instead try to recognize clusters in the features space.

2.0.5 Classification methods

There are many supervised methods which can be used for classification of astrophysical variables.
One of the most suitable methods to deal with a sparsely populated knowledge base are Bayesian classifiers. These networks can be used to compute the the probabilities of the object belonging to different classes of transients. Then, by making use of objective criteria, we can determine whether or not the probability for the class of interest is high enough and if the object needs follow-up observations. In order to use the Bayesian approach we have to generate a library of prior distributions. Each distribution has to take into account several factors such as brightness changes in a certain filter over a certain time interval. These distributions need to be estimated for each type of variable astrophysical phenomenon that we want to classify.
Another method for the classification of variable sources is to use Support Vector Machines (SVMs). These algorithms try to find in the features space the hyperplane which best separate the components of each pair of classes. If the two classes are not linearly separable, the SVMs make use of different kernel functions which map points of the input space into a higher dimension space in which the classes become linearly separable. SVMs have been used in recent works for variable stars classification (Willemsen & Eyer 2007, Richards et al. 2011).
Other popular algorithms for supervised classification are the Artificial Neural Networks (ANNs). ANNs are, in their simplest form, non-linear regression algorithms in which the classification is the result of a non linear combination of the input features. These algorithms have been used in particular to separate real transient sources from a variety of data artifacts, with a classification rate of $\sim 90\%$ (C. Donalek et al. 2008).
A new method developed by the CRTS team is based on decision trees (Graham et al. 2012). The use a set of 60 features extracted from the light curve of the object to build a set of decision trees which are able to discriminate between different classes of variable object.
When we don't have a set of labeled data the only classification method we can apply is an unsupervised algorithm. The most famous unsupervised algorithms in time domain astronomy are the Gaussian Mixture Modeling

(GMM) algorithm and the Self Organizing Map (SOM). In case of GMM each cluster is represented by a parametric gaussian distribution and then the entire data set is modeled by a mixture of these distributions. A SOM is an unsupervised type of ANN which aim to map the feature space into a space usually two-dimensional or three dimensional, without loosing the topology of the input space.
From this simple review of the methods above it is clear that the classification of variable objects poses many problems: how to characterize them (using light curves or statistical indicators), in case of supervised method which type of knowledge base use (built on the data themselves or on simulated ones), how to solve the computational challenge, how to find the unknown.
The strategy of our project is to use a hierarchical approach to classification. Different types of classifiers perform better for some event classes than for the others, so we propose to test different classification algorithms in order to find the one optimized for each type of variable object. Our approach has the typical decision tree structure and aims at a classification which becomes finer and finer as we go to higher level of branching.

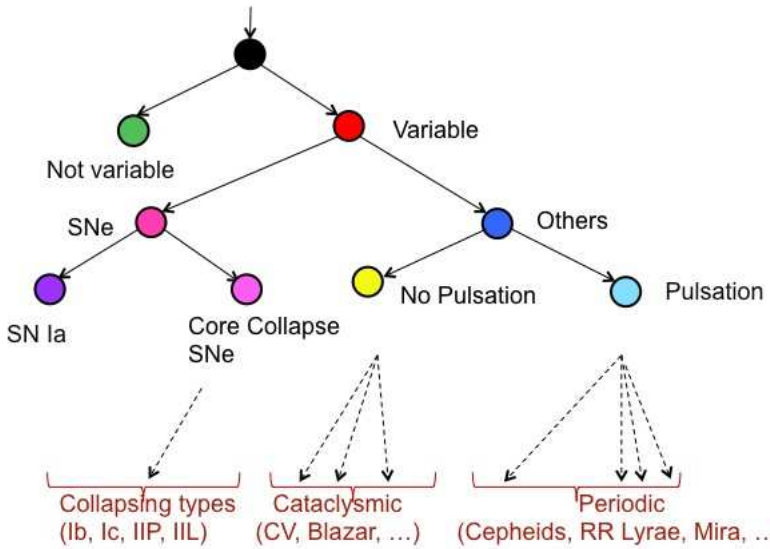


Figure 2.1. Classification Scheme.

In our opinion, the best way to test these classifiers is through simulated data since they allow to better control the various sources of systematics. Real

data are not always suitable because they are often incomplete. A transient, in fact, detected by an increase in brightness is often missing in archival sky surveys and may have just a couple of relatively closely spaced observations in a couple of epochs to go by. A similar approach has been undertaken by the LSST Image Simulation Group. They have implemented a simulation pipeline which is very useful to test detection algorithms. In their first simulations transients were in fact generated randomly, loosing comprehensive theoretical treatment. Our proposal, instead, is produce template images in which variable objects are added according to specific data models.

2.1 Simulation Pipeline

To test the classifiers we developed a simulation pipeline which allows us to build a time series of images and from each image extracts a catalog of sources and their properties. Then, these catalogs have to be merged in a single catalog which contains the information for all the epochs. The flowchart of our simulation pipeline is shown in 2.2. The pipeline makes use of three astronomical software **Stuff**, **SkyMaker**, and **SExtractor**, used in collaboration with **PSFEx**,developed by E. Bertìn and available at http://www.astromatic.net/software. Stuff is responsible for the creation of a catalog of background galaxies, SkyMaker produces an image starting from a catalog of galaxies produced by STUFF, adding a random stellar field, while SExtractor and PSFEx are the software chosen for the catalog extraction. The reasons why we chose SExtractor instead of other similar astronomical software are explained in Chapter 3.

2.2 Setup Phase

In the setup phase we have to set all the information needed to reproduce realistically an astronomical image. These factors are related to the objects distribution and properties, to the survey strategy, to the instrumental setup, and to the observing conditions.
The setup can be defined through a configuration file, named STraDiWA.config, reported in Appendix A. For reasons of comprehensibility in the STraDiWA configuration file we integrated and assembled all the common parameters to Stuff and SkyMaker and SExtractor. Beside these common parameters the user has also to set the configuration files needed for each software in which there are reported their specific parameters.

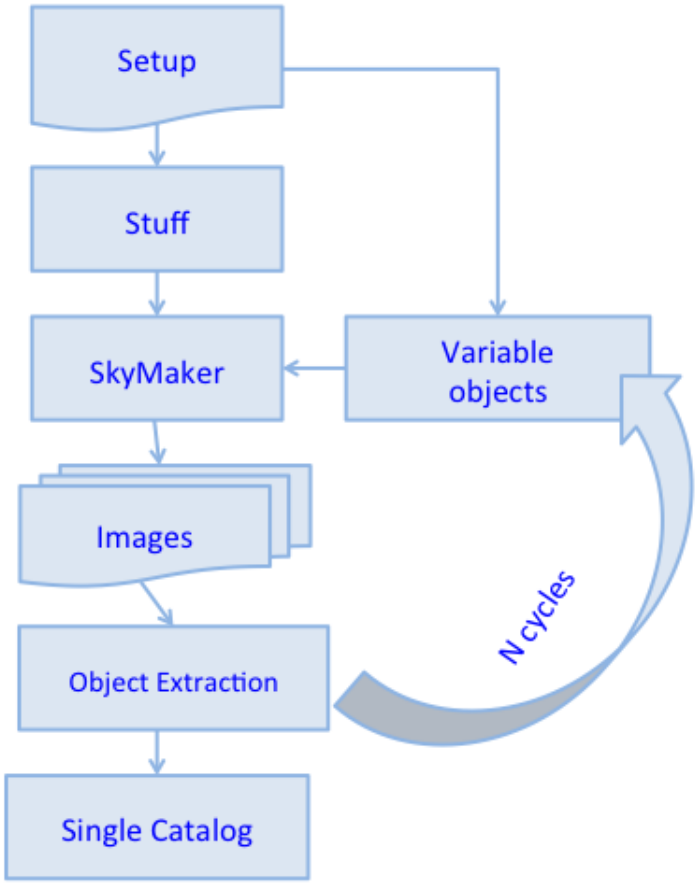


Figure 2.2. Flowchart of the proposed simulation pipeline.

In the setup phase the user must choose mainly the survey strategy and the observing conditions. The survey strategy is determined by the number of the filter in which we want to produce the images (`PASSBAND_OBS`), the limiting in magnitude of these filters (`MAG_LIMITS`) and the sampling rate (`SAMPLING`). When simulating (or observing) a time series of images, the sampling rate can be:

- Uniform. In this case the user can choose the number of days for which the same region of the sky is observed, how often in a night and must specify the length of the night in hours.
- Uneven. In this case the user must provide a time series (in hours).

The observing conditions are ruled by the seeing full with half maximum (`SEEING_FWHM`). During several observations spaced in more days the seeing

can vary following two possible options:

- The SEEING_FWHM can assume each day a different value between a series specified by the user, where each value can be repeated during the simulation. A special case is that of constant seeing, which is however unrealistic.
- The SEEING_FWHM varies randomly each day between a given minimum and maximum.

Last but not least, the user has also to choose the type and distribution of variable objects (VARIABLE). The user can either choose to define for each object its parameters, or can choose to assign them randomly between their range of variation.

2.3 Stuff: creation of the static sky

Stuff is a software that combines spectra, luminosity functions and physical parameters to generate artificial catalogs of the deep extragalactic sky in a standard universe driven by $(\Omega_M, \Omega_\Lambda)$. The program distributes galaxies in redshift space that is subdivided into bins. For every bin the number of galaxies for the Hubble types E, S0, Sab, Sbc, Scd and Sdm/Irr is determined from a Poisson distribution assuming a non-evolving Schechter luminosity function. The different galaxy types are simulated by linearly adding exponential (disk component) and de Vaucouleurs profiles (bulge components) in different ratios. To each galaxy are assigned a random disk inclination angle and a position angle which define the intrinsic ellipticity of the object. The output of the program is a catalog of galaxy positions, apparent magnitudes, semi-minor and major axes, position angles for disks and bulges, de Vaucouleurs type and redshift.
The key input parameters that have to be modified by the user, according to his own purposes, are the image dimensions (IMAGE_SIZE), the pixel size (PIXEL_SIZE), the allowed range of apparent magnitudes (MAG_LIMITS), the detector gain (GAIN), and the required filters (PASSBAND_OBS). There are 120 filters available, covering a wavelength range from 0.29 to 87.74831 μm.The others input keys are mainly related to the cosmological model taken into account, like the value of the Hubble constant, or to the Schechter's functions.

2.4 SkyMaker: Instrumental simulation and image production

SkyMaker started out as a testing tool for the SExtractor source extraction software developed by the same author. It works by taking as input files a list of sources, which can be produced by Stuff, and a setup file. To produce an image this software first build a Point Spread Function model (PSF), that is distribution of the light from a point source, then reads the input catalog and renders sources at the specified pixel coordinates in the frame. Finally SkyMaker provide to add a uniform sky background, with surface brightness provided by the user through the input `BACK_MAG` parameter, and applies to the image Poissonian photon white noise and Gaussian read-out noise of the detector.

PSF Modeling

The PSF used by SkyMaker can be internally generated or loaded through an external fits file. The PSF internal generator has to be able to represent with decent accuracy the PSF of typical astronomical instruments. This means that has to take into account the atmospheric blurring, telescope motion blurring, instrument diffraction and aberrations, optical diffusion effects and intra-pixel response. The produced PSF is a convolution between these components.
We want to focus only on these components that the user can control by modifying the configuration file: the instrument diffraction and aberrations and the optical diffusion. The instrumental PSF become dominated by diffusion beyond a few FWHMs from the center. This effect produces a so called 'aureole" that has to be taken into account when simulating deep and wide galaxy fields, as it reproduces the background variations found on real images around bright stars. The SkyMaker PSF simulator reproduces the effects of diffraction and aberrations in the Frauhofer regime of Fourier optics by manipulating a virtual entrance pupil function $p(\rho, \theta)$. The amplitude part of p is mainly determined by the characteristic of the primary mirror M1, by the effects caused by the presence of spider arms, and by the obscuration of the secondary mirror on the primary.
Optical aberrations may be added by introducing changes of phase $\phi(\rho, \theta)$ of the complex pupil function. SkyMaker can simulate a wide range of aberrations:

- defocus: $\phi_{\text{defocus}} \propto \rho^2$,

- astigmatism: $\phi_{\rm asti} \propto \rho^2 \cos^2(\theta - \theta_{\rm asti})$,
- coma: $\phi_{\rm coma} \propto \rho^3 \cos(\theta - \theta_{\rm coma})$,
- spherical: $\phi_{\rm spher} \propto \rho^4$,
- tri-coma: $\phi_{\rm tri} \propto \rho^3 \cos^3(\theta - \theta_{\rm tri})$, and
- quad-ast: $\phi_{\rm quad} \propto \rho^4 \cos^4(\theta - \theta_{\rm quad})$.

Phase terms are individually normalized following the ESO d_{80} convention: phase coefficients represent the diameter of a circle enclosing 80% of the total flux of an aberrated spot on the focal plane.
A set of input parameters, like the diameters of M1 and the central obscuration, the number, position angle and thickness of the spider arms, makes it possible to simulate with reasonable accuracy the diffraction pattern of most common telescope configurations.

Source Modeling

After build the PSF, SkyMaker deals with the source modeling. So far SkyMaker can model only galaxies and stellar objects. The galaxies are modeled as a sum of a bulge profile and an exponential disk. The bulge follows a de Vaucouleurs profile:

$$\mu_B(r) = m - 2.5log(B/T) + 8.3268\left(\frac{r}{r_{eff}}\right)^{1/4}. \qquad (2.4.1)$$

where μ_B is expressed in $mag/arcsec^2$, m is the apparent magnitude, B/T the apparent bulge-to-total ratio and r_{eff} the effective radius of the spheroid in arcseconds. The disk component is given as exponential profile:

$$\mu_D(r) = m - 2.5log(1 - B/T) + 1.0857\left(\frac{r}{r_h}\right) + 5logr_h + 1.9955, \qquad (2.4.2)$$

where r_h is is the disk scalelength in arcseconds. The parameters m, B/T , r_{eff} , r_h as well as independent aspect ratios and position angles for both components must be read from the input list.
There are many parameters to be set in the SkyMaker configuration file. The most important are related to *i)* pupil features, e.g. the size of the mirrors and the aberration coefficients, *ii)* the detector characteristics, e.g. gain, saturation level and image size, *iii)* PSF model, e.g radius and surface brightness of the aureole and *iv* observing condition, full width half maximum of the

seeing and exposure time.
We wish to stress that the simulations obtained with the combination of Stuff and SkyMaker do not include any artefact such as bad pixels, ghosts or bad columns or other effects which.

2.5 Rules for variable objects

An important aspect of the project was to identify relevant group of variable objects and derive sets of rules for their definition. The models developed for each variable object must be seen as particular instances of a general template. A variable object must be defined by a series of parameters. he number and type of parameters can vary for each class of variable object, for example for a periodic variable we can specify the amplitude, the period, etc., while for a cataclysmic variable can be important to specify the time of the explosion. Furthermore, each class has to take into account the different behavior of the objects at various wavelength. The modules so far implemented are the Classical Cepheids and the type Ia Supernovae. We choose to start implementing the Cepheids because they are the classical example of periodic objects, while Type Ia Supernovae were the following choice according to the classification scheme proposed in Figure 2.1.
Furthermore we have implemented also a module for random objects. These objects in fact were very useful to verify the simulation setup fixed at the beginning of the project. Their magnitude vary randomly in the magnitude limits set in Stuff and SkyMaker in an unrelated way in each band.
In Appendix B we can see the implementation of the abstract class Variable object, and of the class for random variables.

2.5.1 Classical Cepheids

Classical Cepheids are pulsating stars whose magnitude vary periodically. Their light curves is generally approximates as a sinusoid with a constant phase term. The amplitude of their variation ranges from 0.2 to 2 mag, as estimated by American Association of Variable Star (AAVSO)[1]. We begin to simulate Classical Cepheids in our Galaxy. In order to model this class the steps are:

- assign the Period;
- use P-L relation in order to find the mean absolute magnitude;

[1] http://www.aavso.org/

- assign the phase;
- assign the amplitude;
- assume a sinusoidal law and evaluate the temporal evolution of the absolute magnitudes;
- estimate correction for stellar extinction, using the values of absorption coefficients given by Tammann et al. (2003) and extracting randomly the color excess.

Using different calibrations of PL relationships, we can model different types of Classical Cepheid, for example discriminating between pulsation mode or take into account their metallicity. So far, we used the coefficients for the mean PL relation calibrated in Bono et al. , 2010 and reference therein valid for Galactic Cepheids. The number of bands for which we have a calibration for PL relation limits the number of band in which we can model our objects. So far this relation has been calibrated mainly in Johnson bands BVRIJHK. For a classical Cepheid we have four free parameters: the initial apparent magnitude m_i, the period P, the phase ϕ, and the amplitude variation A . These can by defined by the user or extracted randomly in the appropriate ranges. These are:

- [0, 2π] for the phase;
- [0, 2] mag for the amplitude;
- [0, 70] days for the period;
- magnitude limits of Stuff and SkyMaker for the initial magnitude.

Coordinates of the objects are extracted randomly within the image.
Actually simulations of Classical Cepheid have been made in Johnson B, V and I band using coefficients for period-luminosity relation calibrated in Tammann et al. (2003). In Appendix B we can see the implementation of this class.

2.5.2 Type Ia Supernovae

Contardo et al. 2000 used an empirical model to fit the light curve of a sample of type Ia Supernovae in the UBVRI Johnson filter. They found an analytical form of the light curves consisting of a Gaussian (for the peak phase) atop a linear decay (late-time decline), a second Gaussian (to model the secondary

maximum in the V, R, and I band light curves), and an exponentially rising function (for the pre-maximum segment):

$$m(t) = \frac{f_0 + \gamma(t - t_0) + g_0 e^{\frac{(t-t_0)^2}{2\sigma_0^2}} + g_1 e^{\frac{(t-t_1)^2}{2\sigma_1^2}}}{(1 - e^{\frac{\tau - t}{\theta}})}. \tag{2.5.1}$$

We decided to use this expression to model type Ia Supernovae. As we can see in Eq. 2.5.1 we have to set eight parameters (f_0, γ, t_0, g_0, σ_0, g_1, t_1, σ_1, τ, θ) for each band. Actually, some of these parameters are related to each other. To obtain realistic groups of values for the simulations we searched for the range within which those parameters vary in the Contardo PhD Thesis. Analyzing the data we found the variation ranges reported in Tab. 2.1. We excluded the U band because of the lack of information.

Parameter	**B Band**	**V Band**	**R band**	**I Band**
f_0 **(mag)**	[14, 20]	[13, 19]	[12, 18]	[9, 18]
γ **(mag/days)**	[0.01, 0.025]	[0.02, 0.03]	[0.025, 0.04]	[0.015, 0.06]
g_0 **(mag)**	[-3.5, -2]	[-2.5, -1]	[-1.5, -0.5]	[-3, -0.5]
σ_0 **(days)**	[10, 18]	[5, 30]	[5, 10]	[5, 15]
g_1 **(mag)**		[-0.5, -0.1]	[-0.6, -0.2]	[-0.8, -0.5]
σ_1 **(days)**		[4, 10]	[4, 10]	[5, 15]
θ **(days)**	[1, 10]	[1, 10]	[1, 10]	[1, 10]

Table 2.1. Ranges of variation chosen for the parameters f_0, γ, g_0, σ_0, g_1, σ_1, θ of Eq. 2.5.1.

There are not variation ranges for g_1 and σ_1 in the B band since in this band the light curve of a Supernova Ia does not show the secondary maximum. Temporal parameters, t_0 t_1 and τ are not reported in Table 2.1. Instead, we choose to study the correlation between those parameters, focusing in particular on the relation between t_0 in B band and t_0 t_1 and τ in other bands and between t_0 and τ_0 in B band. Relations between temporal parameters result to be linear. Fig. 2.3 - 2.5 show t_0, t_1 τ in each band as function of t_0 in B band. Fig. 2.6 shows t_0 in function of τ in B band. In the caption of each figure is reported the result of the linear regression. All temporal values are measured in Julian Dates.

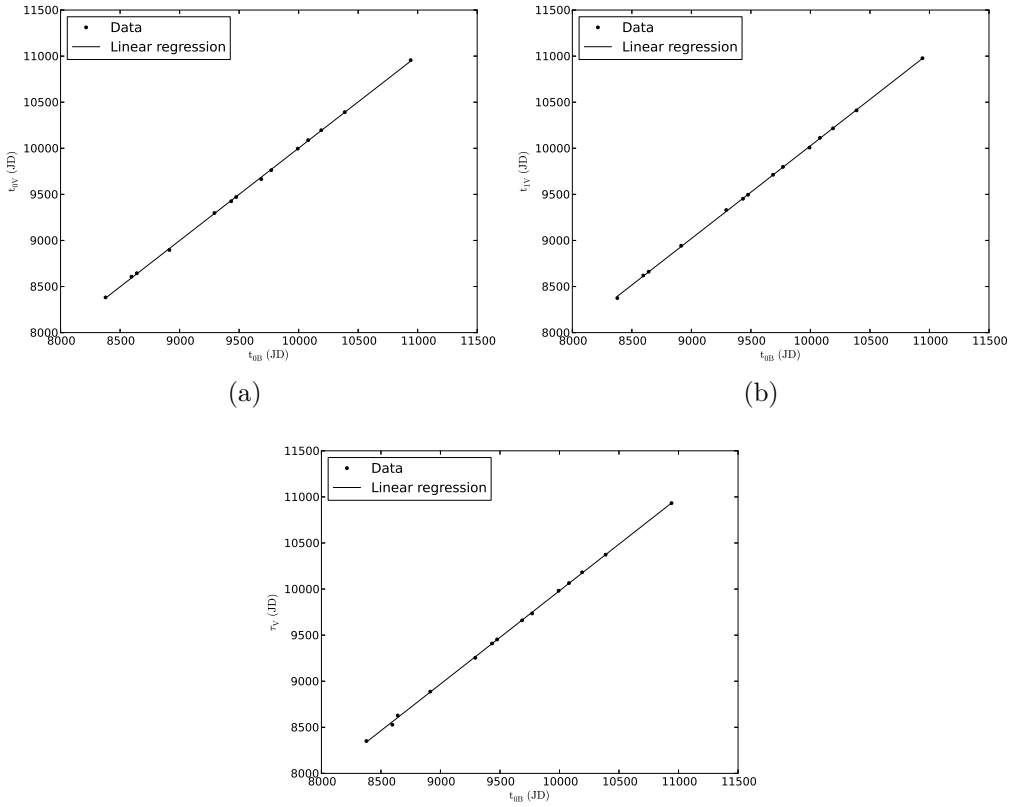


Figure 2.3. On the x-axis: t_0 in B band. On the y-axis t_0 in V band (panel a) , t_1 in V band (panel b), τ in V band (panel c). The equation of the best fits are:
$t_{0V} = 1.003 * t_{0B} - 28.80$. The r-square of the fit is: 0.99981.
$t_{1V} = 1.007 * t_{0B} - 42.09$.The r-square of the fit is: 0.99981.
$\tau_V = 1.012 * t_{0B} - 139.69$. The r-square of the fit is: 0.99970.

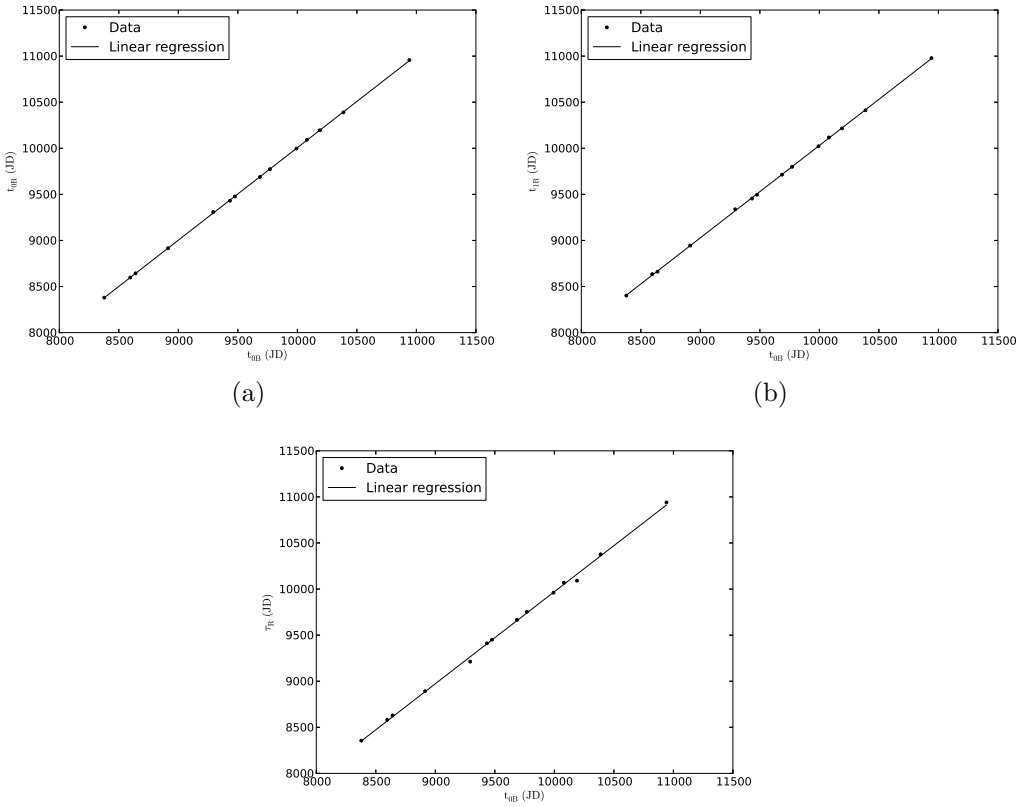


Figure 2.4. On the x-axis: t_0 in B band. On the y-axis t_0 in R band (panel a) , t_1 in R band (panel b), τ in R band (panel c). The equation of the best fits are: $t_{0R} = 1.003 * t_{0B} - 28.23$. The r-square of the fit is: 0.99996.
$t_{1R} = 1.001 * t_{0B} + 15.80$.The r-square of the fit is: 0.99989.
$\tau_R = 0.999 * t_{0B} - 17.25$. The r-square of the fit is: 0.99862.

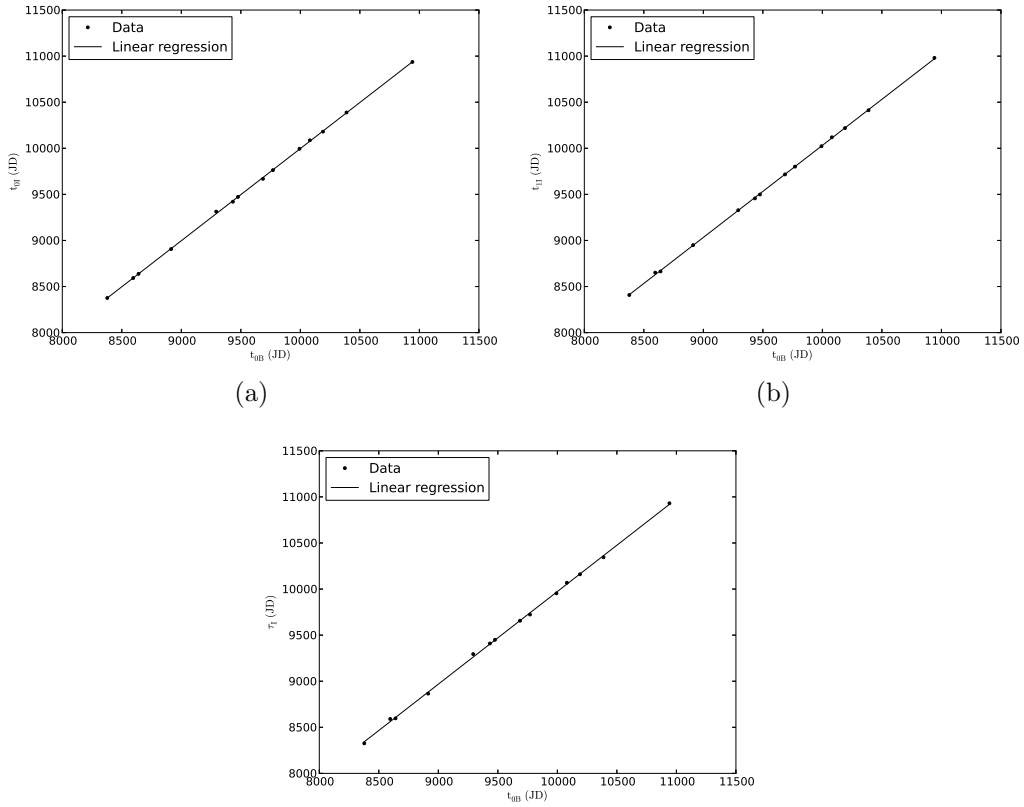


Figure 2.5. On the x-axis: t_0 in B band. On the y-axis t_0 in I band (panel a) , t_1 in I band (panel b), τ in I band (panel c). The equation of the best fits are:
$t_{0I} = 1.000 * t_{0B} - 1.024$. The r-square of the fit is: 0.99985.
$t_{1I} = 0.999 * t_{0B} + 43.54$. The r-square of the fit is: 0.9986.
$\tau_I = 1.005 * t_{0B} - 75.22$. The r-square of the fit is: 0.99948.

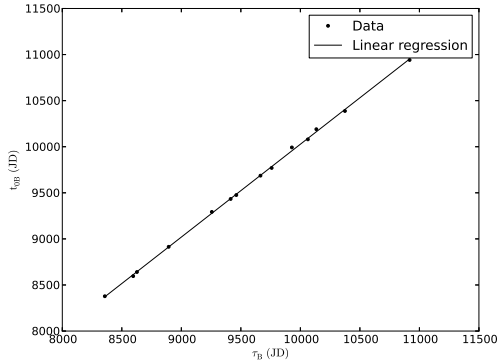


Figure 2.6. On the x-axis: τ in B band, on the y-axis: t_0 in I band.
The equation of the best fit line is: $\tau_B = 1.007 * t_{0B} - 49.06$. The r-square of the fit is: 0.99951.

These relations seem to represent temporal shifts, and in fact all the fits have slopes almost equal to one.
Handling the data, we also found that there seems to be a relationship between f_0 in B and in the other bands and, in first approximation, we can suppose that it is linear. Fig. 2.7 shows f_0 in V, R and I band against f_0 in B band. In the caption of each figure is reported the result of the linear regression.
In practice, in order to to simulate a type Ia Supernova we used the Contardo model with γ, g_0, σ_0, g_1, σ_1, θ extracted randomly from the ranges in Tab. 2.1, fixating τ_B in order to derive other temporal parameters using the relations shown above. At the user is given the possibility to choose how many days the Supernova is before or after the maximum. Furthermore the user can set the value of f_{0B}, while f_{0V}, f_{0R} and f_{0I} result from the previous relations. Once modeled, the Supernova is associated to a galaxy having an integrated magnitude comparable with the maximum luminosity of the SN in the B band.
In Appendix B we can see the implementation of this class.

2.6 Catalog extraction

At the bottom of the simulation flowchart proposed in 2.2, there is the extraction, from each image, of a catalog of sources, containing as much information

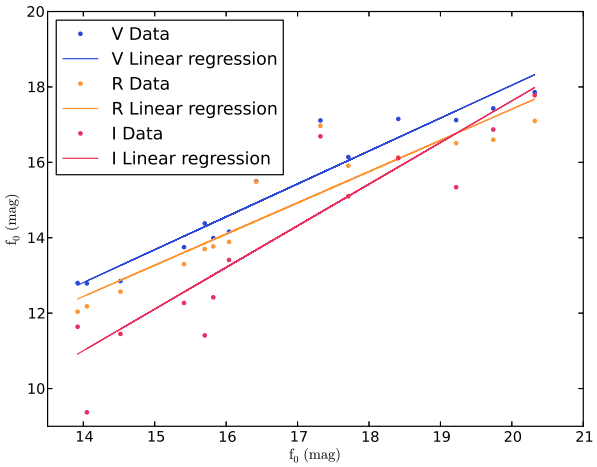


Figure 2.7. On the x-axis: f_0 in B band. On the y-axis f_0 in the other bands. Green points show f_0 in V band. The equation of the best fit line is: $f_{0V} = 0.872 * f_{0B} + 0.595$. The r-square of the fit is: 0.922.
Orange points show f_0 in R band. The equation of the best fit line is: $f_{0R} = 0.828 * f_{0B} + 0.849$. The r-square of the fit is: 0.872.
Red points show f_0 in I band. The equation of the best fit line is: $f_{0I} = 1.10 * f_{0B} - 4.46$. The r-square of the fit is: 0.755.

as possible on their properties. As we said before this task is achieved by SExtractor in combination with PSFEx. The reasons behind this choice are explained in details in Chapter 3.

2.7 Simulation example

As one of the first simulations, we produced 50 images, as observed by the VST[2] (VLT Survey Telescope) telescope and the OmegaCAM camera[3]. The Field of View (FoV) of OmegaCAM@VST is 1 square degree with a pixel scale of 0.213 arcsec/pixel. Therefore the size of the images was set to 16kx16k. The aberration coefficients, the tracking errors, the positions of the spiders were set properly according to the VST technical specifications. The observations are spaced within 90 days, with an uneven sampling rate and with the FWHM of the seeing varying between 0.6 and 1.0 arcsec, according to the ESO statistics at Cerro ParAnal. The magnitude range was set to 14-26 mag and the exposure time of each image to 1500 s.

[2] http://www.eso.org/public/teles-instr/surveytelescopes/vst/surveys.html
[3] http://www.astro-wise.org/~omegacam/index.html

The total number of simulated objects is around 10000. This include non variable stars and galaxy, a sample of classical Cepheid, a sample of type Ia Supernovae with their host galaxy and a sample of randomly variable objects.
Fig. 2.8 and Fig. 2.9 show a section of the B-band image produced at t=0d. The stamps below each image show the evolution of the variable object in the green box (a type Ia Supernova in Fig. 2.8 and a Classical Cepheid in Fig. 2.9).
The B-band light curves of the objects selected in Fig. 2.8 and Fig.2.9 are shown in Fig. 2.10 and Fig. 2.11. The magnitude of the objects are the Kron magnitudes obtained by running SExtractor on each image.

(a) (b) (c) (d) (e) (f) (g)

Figure 2.8. Stamp of the image: the green box in 2.8 is a type Ia Supernova at -9.34 days from its maximum light within its host galaxy. Figs. 2.8b - 2.8g show a close up image of the Supernova at the beginning of the observation (t=0 days), t=7 days, t=18 days, t=34 days, t=59 days, and t=89 days respectively.

Figure 2.9. Stamp of the image: the green box in 2.9a is a Classical Cepheid with a period of 25.39 days. Figs. 2.9b- 2.9g show a close up image of the Cepheid at the beginning of the observation (t=0 days), t=7 days, t=22 days, t=34 days, t=45 days, and t=89 days respectively.

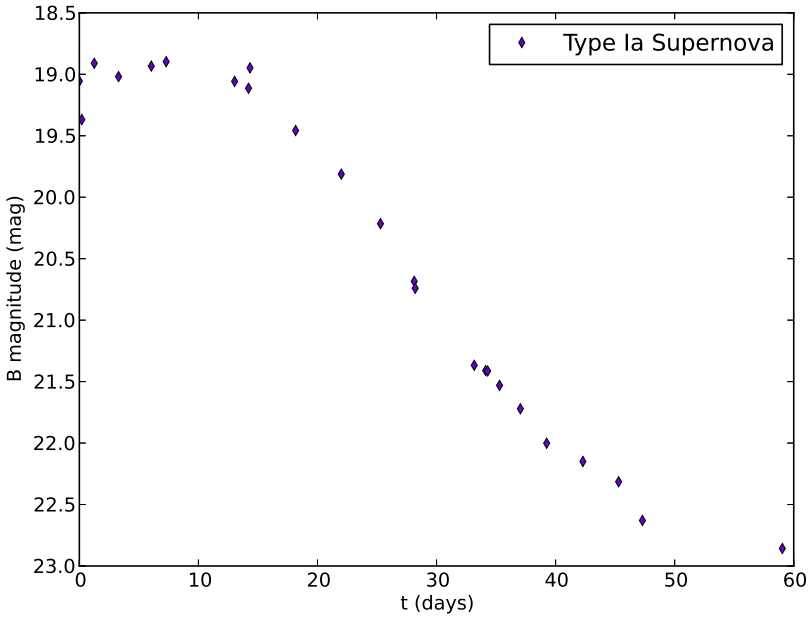


Figure 2.10. Light curve of the type Ia Supernova in Fig. 2.10. On the x-axis there is the time in days of the observation, while on the y-axis there is the B magnitude of the object.

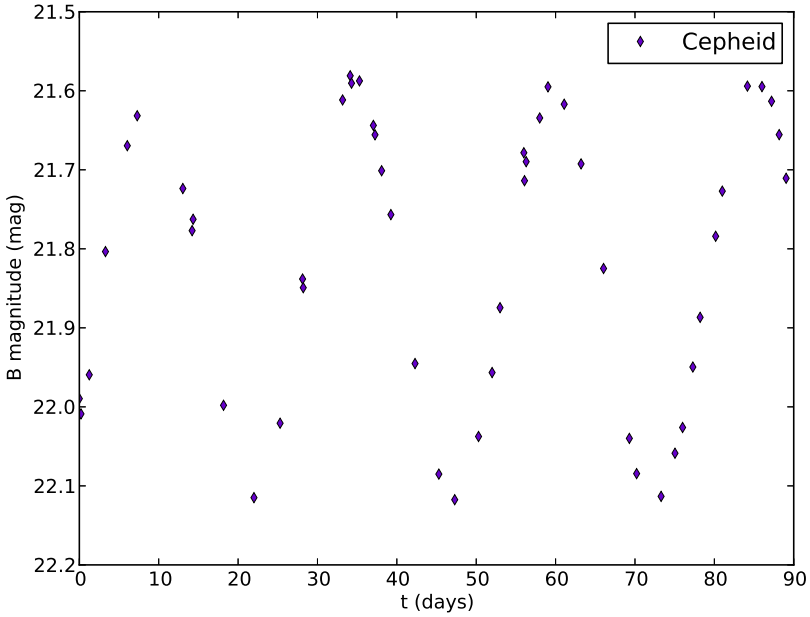


Figure 2.11. Light curve of the Classical Cepheid in Fig. 2.11.On the x-axis there is the time in days of the observation, while on the y-axis there is the B magnitude of the object.

Chapter 3

Comparison between source extraction software

As we said in Introduction, the advent of the new surveys has profoundly changed the needs of scientists in terms of software and data analysis. For instance the sheer size of the raw data makes almost impossible to re-process the raw image data and therefore catalogs are becoming the primary source of information, whereas for catalogs we intend long tables where each object is a row and each column is a different measured property.
The main aspects to take into account when extracting a catalog from an astronomical image are: *i)* to detect as many as possible sources (completeness), *ii)* to minimize the contribution of spurious objects, *iii)*to correctly separate sources resolved/unresolved[1], to produce accurate measurements of astrometric and photometric quantities[2].
Among the main source extraction software used by the astronomical community there are SExtractor (Bertìn & Arnouts, 1996) and DAOPHOT II (Stetson, 1987), which is often used in combination with its companion tool ALLSTAR (Stetson, 1994). SExtractor is commonly used in extragalactic astronomy and has been designed to extract from the images a list of measured properties for both stars and galaxies; while DAOPHOT and ALLSTAR were designed to perform mainly stellar photometry.
Part of my thesis work was therefore to perform a comparison between DAOPHOT and SExtractor photometry in order to find the one best suited to our purposes. So far DAOPHOT II and ALLSTAR have been able to produce more accurate photometry for stellar/unresolved objects using a

[1]For historical reasons, this problem is also known as Star/Galaxy separation.

[2]This chapter is largely extracted from a paper submitted for publication to the journal of the Astronomical Society of the Pacific (Annunziatella et al. 2012.

technique known as Point Spread Function (PSF) fitting . The PSF fitting photometry in SExtractor has, instead, become possible only in recent years. The first attempts were made in the late 90s, when the PSFEx (PSF Extractor) software was made available within TERAPIX "consortium". This tool extracts precise models of the PSF from images processed by SExtractor. However only after the 2010 public release of PSFEx (Bertìn, 2011) [3], and with the recent increases in computer performance, PSF fitting photometry has become actually available in SExtractor.
In this work we want to compare the results obtained using the combination of SExtractor with PSFEx, and DAOPHOT with ALLSTAR, focusing, in particular, on the completeness and purity of the extracted catalog, on the accuracy of photometry and on the determination of centroids, both with aperture and PSF fitting photometry. Previous comparison between extraction software was performed by Becker et al. (2007). They, in pursuit of LSST science requirements, compared DAOPHOT, two versions of SExtractor (SExtractor 2.3.2 and SExtractor 2.4.4) and DoPhot (Mateo & Schechter, 1989). However, differently from the present work, they use as "true" values the measurements obtained with the SDSS imaging pipeline *photo* (Lupton et al., 2001), while we use simulations. Furthermore, we wish to stress that their result was biased by the fact that in 2007 the PSF fitting feature had not yet been implemented in SExtractor.
Also in this case we use image simulations. Image simulations are suitable to test performances various analysis software. Simulations, in fact, allow to know exactly the percentage and the type of input sources and their photometric properties. In this case we used simulated images obtained by using Stuff and SkyMaker setting the instrumental characteristics as in Sect. 2.7. In order to reduce the computational time, we limited our simulations to a FoV of 1/4 of VST. The FWHM of the seeing was set to 0.7 arcsec. We obtain a catalog of input sources of N=4120 down to the input magnitude limit. The stamp of the image to which the results reported in this work refer is shown in Fig. 3.1.
We report only the results obtained in the B-band.

3.1 Source extraction software

In the following section we briefly discuss how DAOPHOT works in combination with ALLSTAR and how SExtractor works in combination with PSFEx,

[3] Available at http://www.astromatic.net/software/psfex.

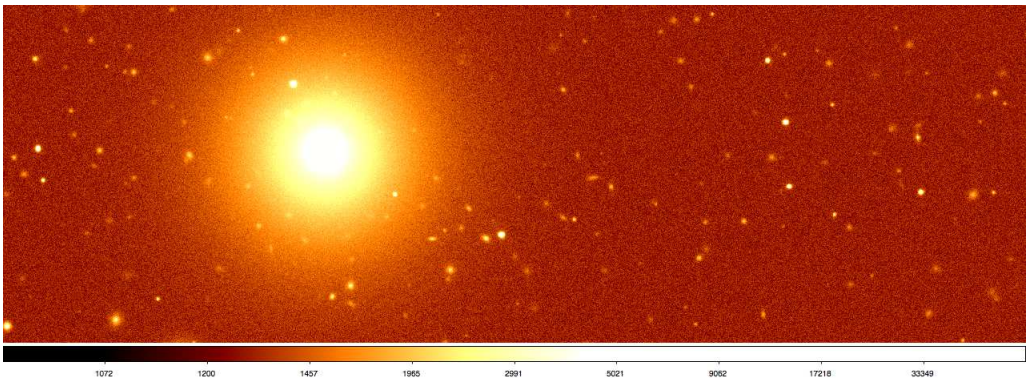


Figure 3.1. Stamp of the B image used to obtain the results reported in this chapter.

and give a brief overview of the main parameters needed to be set in order to optimally run the selected software .

3.1.1 DAOPHOT II

DAOPHOT II is composed by a set of routine mainly designed to perform stellar photometry and astrometry in crowded fields.
The software requires in input several parameters which must be listed in the file *daophot.opt*, including detector gain, readout noise (`GAIN, READ NOISE`), saturation level (`HIGH GOOD DATUM`), approximate size of unresolved stellar sources in the frame (`FITTING RADIUS`), PSF radius (`PSF RADIUS`), PSF model (`ANALYTIC MODEL PSF`), and a parameter designed to allow the user to visually inspect the output of each routine (`WATCH PROGRESS`) .
The first step that DAOPHOT II performs is to estimate the sky background and find the sources above a fixed threshold, given as input parameter, through the FIND routine.
The derived value of this threshold represents the level (in ADU) above the sky background required for a source to be detected. In order to ignore smooth, large-scale variations in the background level of the frame, the image is convolved with a lowered truncated Gaussian function whose FWHM is equal to the value set in input by the `FWHM` parameter. After the convolution the program searches for the local maxima sky enhancement.
Once the sources are detected, DAOPHOT II performs the aperture photometry via the PHOTO routine. Aperture photometry usually requires the definition of at least two apertures. The first one is usually circular, centered on the source and with a radius of a few times its FWHM. The second one is instead ring-shaped, usually is concentric to the first one and with inner

radius equal to the radius of the first aperture. This aperture is used to estimate the sky contribution and usually it covers a number of pixels equal or at least comparable with that of the inner aperture. Then the flux of the source then is obtained by subtracting the sky flux from the aperture flux. The size of the apertures must be chosen thoroughly. In fact if the radius of the inner aperture is too small, there will be a flux loss; while if it is taken too large, too much sky is included and the measurements will become too noisy. The radii of the apertures with DAOPHOT can be specified in a input file: *photo.opt*. An inner and an outer radius of a sky annulus centered on the position of each star must also be specified.
Beside the aperture magnitude the PHOTO routine produce the coordinates of the centroids of the sources, that is the coordinates of the barycenter of the intensity profile around the source.
Aperture photometry performs rather well in the hypothesis of bright and isolated stars. However in crowded fields stars are faint and tend to overlap. In these cases the PSF fitting photometry can produce better results. The last measurement requires that a PSF model has to be derived from the stars in the image. The normalized PSF model is then fitted to each star in the image to obtain the intensity and magnitude.
DAOPHOT II can build a PSF model from a sample of stars obtained with the PHOTO routine in an interactive procedure intended to subtract neighbor stars that might contaminate the profile. Among them, DAOPHOT will exclude stars within one radius from the edges of the image and the stars too close to saturated stars. The analytical formula of the PSF is chosen by the user among available models: a Gaussian function, two implementation of a Moffat function, a Lorentz function and two implementation of a Penny function(Penny, 1995). The PSF routine produces a PSF model and a list of the PSF stars and their neighbors. The modeled PSF stars can be visually inspected by setting properly the `WATCH PROGRESS` parameter.
Although DAOPHOT is designed for stellar photometry extended sources are likely always present in real images and therefore it is required a reliable method to separate galaxies from stars.
As Star/Galaxy classifier it can be used a sharpness parameter (`SHARP`), which describes how much broader the actual profile of the object is compared to the profile of the PSF. The sharpness is therefore dependent on the model of the PSF build and can be easily interpreted by plotting it as a function of apparent magnitude. Objects with `SHARP` significantly greater than zero are probably galaxies.
In this work we indicate simply with DAOPHOT the stand-alone DAOPHOT

II version 1.3-2[4].

3.1.2 ALLSTAR

After having derived PSF models with DAOPHOT, ALLSTAR fits multiple overlapping point-spread functions to all the stars in the image simultaneously. With every iteration, ALLSTAR subtracts all the stars from a working copy of the input image, according to the current best guesses of their positions and magnitudes. Then, it computes increments to the positions and magnitudes from examination of the subtraction residuals around each position. Finally it checks each star to see whether it has converged or it has become insignificant. When a star has converged, its coordinates and magnitude are written in the output file, and the star is subtracted permanently from the working copy of the image; when a star has disappeared, it is discarded.
The input parameters for ALLSTAR, listed in *allstar.opt*, are similar to those in *daophot.opt* and *photo.opt*
It is possible to improve the determination of centroids, by applying a PSF correction and setting the option `REDETERMINE CENTROIDS`.
To improve the Star/Galaxy classification it is possible to use a sharpness measure obtained by ALLSTAR (`SHARP`). This value may be used also in conjunction with another ALLSTAR output parameter, χ, which is the observed pixel-to-pixel scatter from the model image profile divided by the expected pixel-to-pixel scatter from the image profile. In this work we indicate simply with ALLSTAR the software which comes together with DAOPHOT II version v. 1.3-2.

3.1.3 SExtractor

SExtractor is a software mainly designed to produce photometric catalogs for large number of both point-like and extended sources. Sources are detected in four steps: *i)* sky background modeling and subtracting, *ii)* image filtering, *iii)* thresholding and image segmentation, *iv)* merging and/or splitting of detections. The final catalog is extracted according to the input configuration file in which parameters are set by the user.
The first step of background estimation can be skipped if the user gives manually an input estimation of sky background. For the background estimation automatically performed, the most critical input parameters to be set are `BACK_SIZE`, the size of each mesh of the grid used for the determination of

[4]Available at http://starlink.jach.hawaii.edu/starlink

the background map, and BACK_FILTERSIZE, the smoothing factor of the background map.
Once the sky background is subtracted, the image must be filtered. This implies to convolve the signal with a mask, shaped according to the characteristics that the user wants to highlight in the image. In fact, there are different filters available in SExtractor. The more suitable filters are those using "top-hat" functions, which are optimized to detect extended, low-surface brightness objects, Gaussian functions usually used for faint objects detection, and "mexhat" filters, which work with a high value of detection threshold, suitable for bright detections in very crowded star fields.
The detection process is mostly controlled by the thresholding parameters (DETECT_THRESHOLD and ANALYSIS_THRESHOLD). The choice of the threshold must be carefully considered. A too high threshold determines the loss of a high number of sources in the extracted catalog, while a too low threshold value leads to the detection of spurious objects. Hence it is necessary to reach a compromise by setting these parameters according to the image characteristics, the background RMS, and also to the final scientific goal of the analysis.
Two or more very close objects can be detected as an unique connected region of pixels above threshold and, in order to correct for this effect, SExtractor adopts a deblending method based on a multi-thresholding process. Each extracted set of connected pixels is re-thresholded at N levels linearly or exponentially spaced between the initial extraction threshold and the peak value. Also here a compromise is needed to be found since a too low value for the deblending parameters leads to not separate between close sources, while a too high value leads to split extended faint sources in more components. Alternatively it is possible to extract the catalog with different deblending parameters and merge detections for extended sources or close pairs.
Once sources have been detected and deblended, the software starts the measurement phase. SExtractor can produce measurements of position, geometry, and of several types of photometric parameters, including different types of magnitudes. Among photometric quantities, there are the aperture magnitude (MAG_APER), which has the same meaning as explained in Sect. 3.1.1, the Kron magnitude, MAG_AUTO , (Kron, 1980) which is the magnitude estimated through an adaptive aperture, and the isophotal magnitude (MAG_ISO), which is computed by considering the threshold value as the lowest isophote.
Among position parameters there are the barycenter coordinates, (X_IMAGE, Y_IMAGE), computed as the first order moments of the intensity profile of the image, and windowed positional parameters (XWIN_IMAGE,YWIN_IMAGE), computed in the same way as the barycenter coordinates, except that the pixel values are integrated within a circular Gaussian window as opposed to the

object's isophotal footprint.
To separate extended and point-like sources it is possible to use *Stellarity Index* (`CLASS_STAR`) which is the result of a supervised neural network and is used to perform a Star/Galaxy classification. `CLASS_STAR` can assume values between 0 and 1. In theory, SExtractor considers objects with `CLASS_STAR` equal to zero to be galaxies, and those with `CLASS_STAR` equal to 1 to be a star. In practice stars are classified selecting a value for `CLASS_STAR` above 0.9. Two other parameters used often to discriminate between Star and Galaxies are the half-light radius (`FLUX_RADIUS`) and the peak surface brightness above background ($\mu_{\max}$). When plotted against the Kron magnitude, these two parameters identify a so-called *stellar locus*.
In this work we indicate simply with SExtractor the version of the software v. 2.14.7 (trunk.r284).

3.1.4 PSFEx

The last version of SExtractor can work in combination with PSFEx, which builds a model of PSF of the image. The PSF is expressed as a sum of N×N pixel components, where each component is weighted by the appropriate factor in the polynomial expansion (see Mohr et al. 2012). Then SExtractor takes the PSFEx models as input and uses them to carry out PSF corrected model fitting photometry for all sources in the image.
PSFEx accepts as input a catalog produced by SExtractor to build a model of PSF of the image which can be read back in a second run by SExtractor itself. In order to allow PSFEx to work the first catalog produced by SExtractor must contain at least a given number of parameters as we can read in the PSFEx manual[5]. In particular the catalog must contain the parameter `VIGNET`, a small stamp centered on each extracted source used to model the PSF. The size of `VIGNET` must be taken accordingly to the size of the photometric apertures defined by `PHOT_APERTURES`.
PSFEx models the PSF as a linear combination of basis vectors. The basis vectors may be the pixel basis, the Gauss-Laguerre basis, the Karhunen-Loève basis derived from a set of actual point-source images, or any other user-provided basis. The size of the PSF and the number and type of the basis should be specified in the configuration file.
By using SExtractor combined with PSFEx it is possible to obtain various estimates of the magnitude in addition to those described in the previous section: `MAG_PSF`, `MAG_POINTSOURCE`, `MAG_SPHEROID`, `MAG_DISK` and `MAG_MODEL`. `MAG_PSF` is the magnitude resulting from the PSF fitting, `MAG_POINTSOURCE`

[5] https://www.astromatic.net/pubsvn/software/psfex/trunk/doc/psfex.pdf

is the point source total magnitude obtained from fitting, MAG_SPHEROID is the spheroidal component of the fitting, MAG_DISK is the disk component of the fitting and MAG_MODEL is the sum of the spheroid component and the disk component. It is also possible to measure morphological parameters of the galaxies, such as spheroid effective radius, disk aspect ratio, disk scalelength. With a model of the PSF, it is possible to extract a more accurate star/galaxy classification using the new SExtractor classifier, SPREAD_MODEL, which is a normalized simplified linear discriminant between the best fitting local PSF model and a more extended model made by the same PSF convolved with a circular exponential disk model with scalelength = FWHM/16, where FWHM is the full-width-half maximum of the PSF model (Desai et al., 2012). A more detailed description of PSFEx and the new SExtractor capabilities can be found in Bertìn (2011) and Armstrong et al. (Jan 2010).
In this work we indicate simply with PSFEx the version of the software v. 3.9.1.

3.2 Catalog extraction

In this section we provide a general indication of how we set the input parameters in order to extract the catalogs with the software presented above.

3.2.1 DAOPHOT and ALLSTAR

Apart from instrumental parameters, such as gain, saturation level and readout noise, which were set according to the values used for the simulations (Sect. 2), in DAOPHOT detection and photometric options must be configured through input files. In the present analysis, the threshold value was chosen to detect as many possible sources, while avoiding as much as possible spurious detections. In fact, as the threshold decreases, the number of detected sources increases up to a certain value for which the relation change in steepness. Thus it is possible to choose a reasonable value for the threshold by plotting the number of extracted sources for different threshold values and to choose the threshold near the "elbow" of the function. Moreover, in order to avoid spurious detections the extracted catalog was visually inspected.
The FITTING RADIUS was set equal to the FWHM of the image (see Tab. 3.1). To obtain the best aperture radius we have derived the growth curve for input stellar sources. Then, we fixed pixels the aperture radius to 12.5 (see a1 in Tab. 3.1), which produces the better coverage of the sources input magnitude. Thus, values of INNER RADIUS and OUTER RADIUS were chosen

accordingly, smaller and greater than the aperture radius, respectively.
The PSF analytical model was chosen with the higher level of complexity, that is one of the implementation of the Penny function (Penny, 1995). We chose to visually inspect the image of the PSF produced for all the PSF stars by DAOPHOT. ALLSTAR parameters were set accordingly to those established for DAOPHOT, and furthermore we required the redetermination of the centroids.
The main parameters set for DAOPHOT and ALLSTAR are reported in Tab. 3.1.

Parameter	Values
FITTING RADIUS	3.38
THRESHOLD (in sigmas)	5
ANALYTIC MODEL PSF	6
PSF RADIUS	7.5
a1	12.5
INNER RADIUS	10
OUTER RADIUS	20
REDETERMINE CENTROIDS	1.00

Table 3.1. Main input parameters set in DAOPHOT and ALLSTAR configuration files. FITTING RADIUS, PSF RADIUS, a1, INNER RADIUS and OUTER RADIUS are expressed in pixels.

3.2.2 SExtractor and PSFEx

As for DAOPHOT, SExtractor instrumental parameters have been set accordingly to those defined as input in the simulations (see Sect. 2.7).
Concerning the sky background modeling and subtraction we decided to automatically estimate the background within the software. Given the average size of the objects in pixels in our images, we chose to leave BACK_SIZE to the default value 64. The choice of the filter was more complex. We performed several tests with various filters.In our case, better performances have been obtained by gaussian and top-hat masks. However, the choice between the various filters, although changes the number of detected sources, does not affects their measurements.
For the thresholding parameters we followed the same procedure described in Sect. 3.2.1 for DAOPHOT, by choosing a value near to the change in gradient of the relation between the number of extracted sources and the threshold value for detections. Moreover, the catalog was visually inspected to avoid

residual spurious detections and to verify the deblending parameters.
We fixed the size of the aperture for photometry, according to the one set in DAOPHOT, to 25 pixels of diameter (PHOT_APERTURES). For PSFEx parameters we used a set of 20 pixel basis and a size for the PSF image of 25 pixels according with the aperture size. We adopted a 25×25 pixel kernel following PSF variations within the image up to 2nd order. The main values set for SExtractor and PSFEx are reported in Tab. 3.2.

Parameter	Values
DETECT_MINAREA	5
DETECT_THRESH	1.5
ANALYSIS_THRESH	1.5
FILTER_NAME	tophat_3.0_3x3.conv
DEBLEND_NTHRESH	64
DEBLEND_MINCONT	0.001
BACK_SIZE	64
BACK_FILTERSIZE	3
PHOT_APERTURES	25
BASIS_TYPE	PIXEL_AUTO
BASIS_NUMBER	20
PSF_SIZE	25,25

Table 3.2. Main input parameters set in SExtractor and PSFEx configuration file. DETECT_MINAREA, BACK_SIZE, PHOT_APERTURES and PSF_SIZE are expressed in pixels.

3.3 Results

In this section we compare results obtained using the two software, focusing on four aspects, namely: photometric depth and purity of the extracted catalog, accuracy of the derived photometry and the determination of the positions of the centroids.
All the quantities and the statistics shown in this section are obtained by excluding saturated sources. In Fig. 3.2 it is shown $\mu_{\max}$ as a function of the Kron magnitude of the objects extracted by SExtractor. As we can see from the flattening of star sequence, sources with magnitude B≤19 mag are saturated in the simulated images. Starting from Tab. 3.3 and Fig. 3.3 we report the comparison among results obtained for the whole input magnitude

range of unsaturated sources: 19-26 mag. However, since we consider only input stellar sources recovered by both software and since the completeness limit of DAOPHOT star catalog is B=24 mag (see Sect. 3.3.1) , the last two reported magnitude bins are underpopulated and the results may be affected by catalog incompleteness.

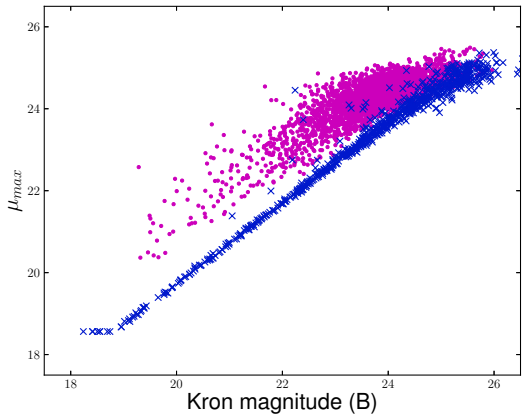


Figure 3.2. $\mu_{\max}$ in function of the Kron magnitude for stars (crosses) and galaxies (points) in the SExtractor catalog.

3.3.1 Photometric depth

The photometric limiting magnitude of the extracted catalog is defined as the magnitude limit below which the completeness drops down to a 90%, where the completeness is the ratio between detected sources, $N_{\rm detected}$, and input sources, $N_{\rm input}$,
With DAOPHOT the photometric depth depends mainly on the threshold applied, while for SExtractor it depends also on the deblending of the sources and on the filter used for the detection (see Sect. 3.1.2). As discussed in Sect 3.2.1 and 3.2.2, in order to fix thresholding and deblending parameters, we performed several tests, visually inspecting extracted sources, and finally we fixed the values reported in Tab. 3.1 and 3.2. Then we compared the results of source extraction obtained using two different filters: a gaussian (dotted line in Fig. 3.3) and a "top-hat" function (continuous line in Fig. 3.3). As shown in Fig. 3.3 using SExtractor with a top-hat filter we can improve the detection of faint sources. In this case the depth of the catalog is ~ 25.0 mag. Hence we refer to this filter in all the tests performed with SExtractor and reported below. Figure 3.3 shows also the percentage of extracted sources per

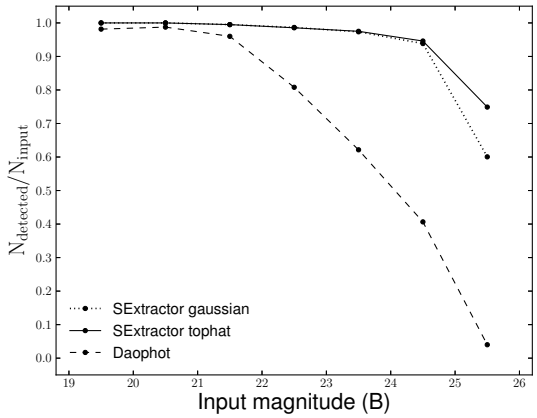


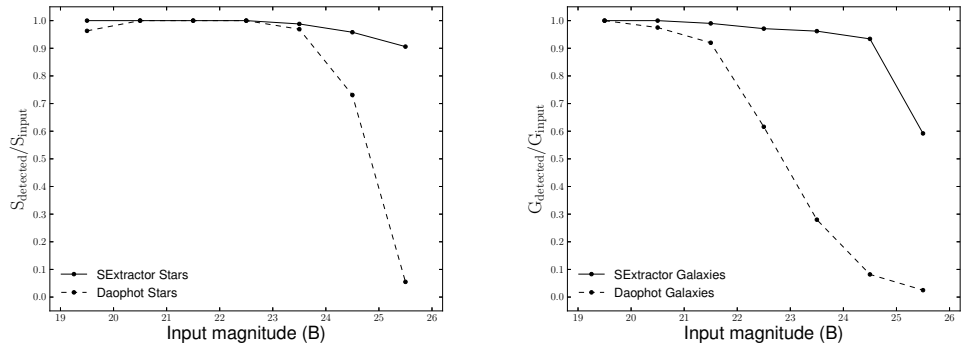
Figure 3.3. Ratio between detected and input sources for different magnitude the bins. The dotted and the solid lines refer to SExtractor used with a gaussian and a top-hat filter respectively, while the dashed line refers to values obtained with DAOPHOT.

magnitude bin obtained using DAOPHOT (dashed line). With this software the completeness drops rapidly to very low values for magnitudes fainter than B = 22.0 mag. However this comparison is misleading. In fact, DAOPHOT is not designed to work with extended sources. For this reason in Fig. 3.4a we report the ratio between the detected sources, which are *a priori* known to be stars ($S_{detected}$), and the input stars (S_{input}). In Fig. 3.4b we show the same quantities but for galaxies ($G_{detected}$, G_{input}).

We can see that the fraction of detected source is higher for stars for both SExtractor (B=26.0 mag) and DAOPHOT (B=24.0 mag). Hence, in conclusion, considering only stars, the final depth returned by DAOPHOT is ~ 2 mag brighter than those produced by SExtractor.

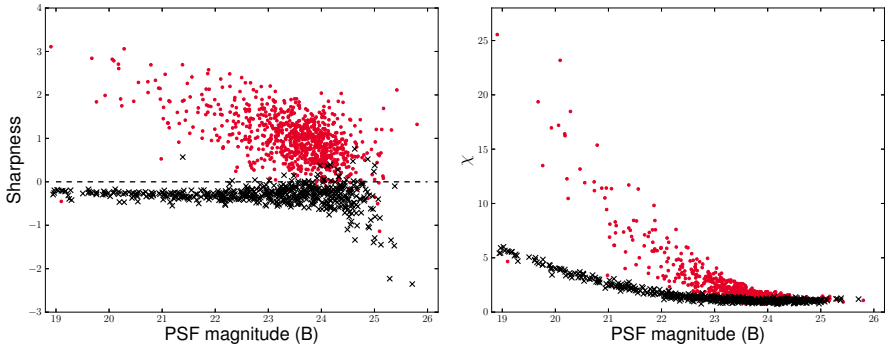
3.3.2 Purity of the catalog

The purity of the catalog is defined as the ratio between the number of the sources well classified and the number of sources detected by the software. For these tests we use only the set of stars detected ($S_{detected}$) and well classified ($S_{classified}$) by both SExtractor and DAOPHOT. We compared results obtained with several methods to classify the sources. In fact, each method



(a) (b)

Figure 3.4. Left panel shows the ratio between detected and input stars as a function of magnitude bins, as obtained by SExtractor (solid line) and by DAOPHOT (dashed line); in the right panel are plotted the same quantities but for galaxies.



(a) (b)

Figure 3.5. Distribution of DAOPHOT sharpness (*left panel*) and χ (*right panel*) as a function of the PSF magnitude for simulated stars (crosses) and galaxies (points). In the left panel the dashed line is the adopted separation limit for the Star/Galaxy classification (see Sect. 3.3.2).

leads to a different estimate of the purity.
For what concern DAOPHOT we used the output parameters `SHARP` (see Fig. 3.5a) and χ (see Fig. 3.5b), redetermined by ALLSTAR (see Sect. 3.1.1). Fig. 3.5a shows the distribution of ALLSTAR sharpness `SHARP` for our data. The separation between the two classes seems to be well defined. On the other hand Fig. 3.5b shows that the use of the χ parameter does not improve the Star/Galaxy classification. For this reason we classified as stars all the sources with `SHARP` lower than 0.

In order to investigate the purity of the catalog with SExtractor we used both traditional methods as well the new parameter SPREAD_MODEL. In Fig. 3.6a we plot CLASS_STAR as a function of the Kron magnitude for our data. As shown the lower is the established limit to separate stars and galaxies, the higher will be the contamination of the stars subsample by galaxies. A reasonable limit for the separation is 0.98.
Figures 3.6b and 3.6c show the locus of stars selected according to the relation between half-light radius and μ_{max}, respectively, as a function of the Kron magnitude. There is an improvement of the source classification compared to the use of CLASS_STAR parameter, allowing a reliable star/galaxy separation down to B = 23.5 mag.
Finally, Fig. 3.6d shows SPREAD_MODEL values as a function of Kron magnitudes. Stars and galaxies tend to arrange themselves in two distinct places of the plot. Also in this case the higher we choose the separation limit, the higher will be the contamination of the stellar sequence from galaxies. A value which can offers a good compromise between a reliable classification and a low contamination is 0.005.
In Fig. 3.7 it is shown the ratio between the sources correctly classified as stars using stellarity index (dotted line), spread model (continuous line) and sharpness parameter (dashed line) described above, as function of input magnitude.
In conclusion if we define a classification with purity of at least 90% reliable, with these methods we can acceptably classify the stars in DAOPHOT down to about 24 mag, that is the photometric depth of the extracted catalog, while in case of SExtractor the classifier SPREAD_MODEL allows to obtain a reliable star/galaxy separation down to B = 26 mag.

3.3.3 Photometry

In this section we compare the results obtained with aperture and PSF photometry on the sample of stars detected by both SExtractor and DAOPHOT. We also investigate the results obtained with Kron, isophotal photometry and model fitting photometry for galaxies detected by SExtractor.
In Tab. 3.3 we report the mean difference and the standard deviation between aperture and PSF magnitude as estimated by DAOPHOT (parts a and b, respectively) and SExtractor (parts c and d, respectively) against input magnitude.
Figure 3.8 shows the residuals between aperture and input magnitudes (top

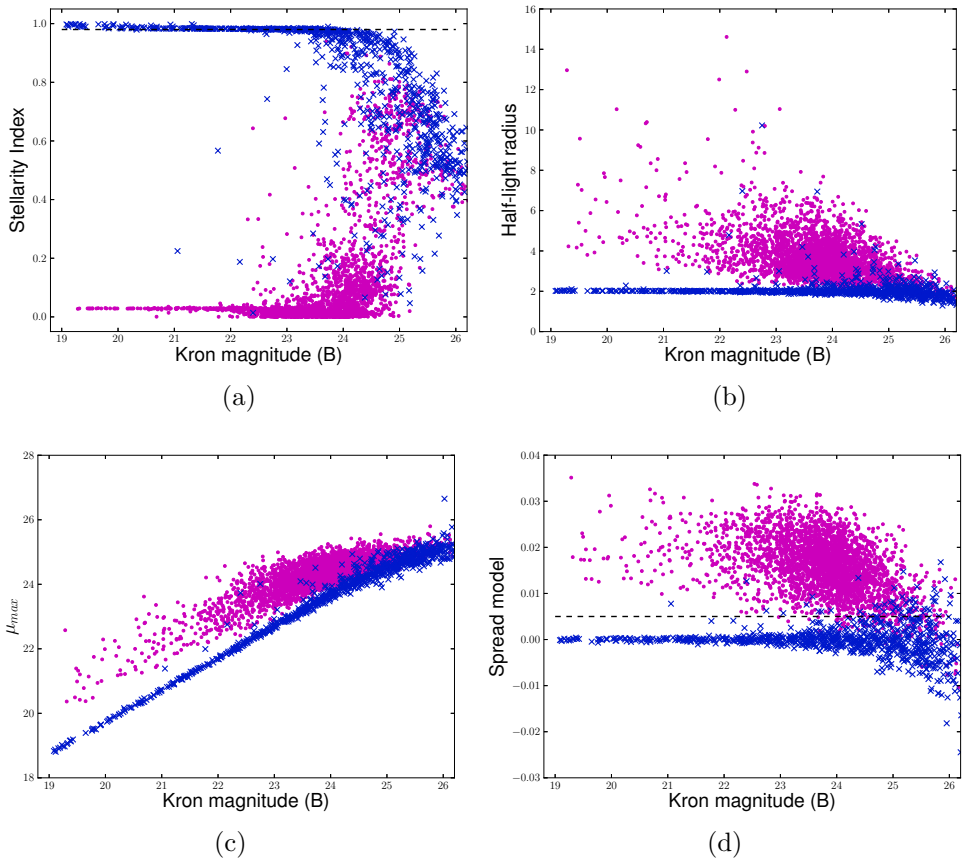


Figure 3.6. Distribution of SExtractor stellarity index (*panel a*), half-light radius (*panel b*), $\mu_{\max}$ (*panel c*) and spread model (*panel d*), as a function of the Kron magnitudes for simulated stars (crosses) and galaxies (points). The dashed line in *panels a* and *d* is the adopted separation limit for the Star/Galaxy classification (see Sect. 3.3.2).

panels), and the residuals between PSF and input magnitudes (bottom panels), as estimated by DAOPHOT (left panels) and SExtractor (right panels). Table 3.3 and Fig. 3.8 show that there is a characteristic broadening of the residuals at fainter magnitudes, as we expect as measurements become sky noise dominated, but the spread in case of PSF photometry remains smaller than for aperture measurements. This behavior was well known (e.g. Becker et al., 2007) for DAOPHOT, but it is worth to underline that SExtractor has reached this level of accuracy in PSF photometry only after the release of PSFEx

In the top part of the Tab. 3.3 we also report the mean difference and the standard deviation between Kron (part a), isophotal (part b) and model

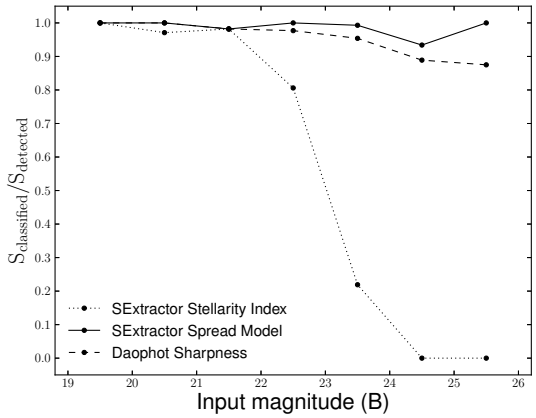


Figure 3.7. Ratio between stars classified by Stellarity Index (dotted line) and Spread Model(solid line) from SExtractor with threshold values respectively to 0.98 and 0.005, and by DAOPHOT sharpness (dashed line) with a threshold value equal to zero and input stars, as function of input magnitude.

magnitudes (part c), respectively, and input magnitudes for stars.
For completeness, since SExtractor is designed also to obtain accurate galaxy photometry, we report in the bottom part of the Tab. 3.4 the mean difference and the standard deviation between Kron (part d), isophotal (part e) and model magnitudes (part f) and input magnitudes for the "true" galaxies detected by the software (See Sect. 5.2).
Considering only stellar photometry, both software are able to deliver acceptable performances in both aperture and PSF photometry, up to a threshold two magnitudes brighter than the limiting magnitudes of input simulated images, which is the completeness limit of the DAOPHOT catalog. Furthermore the Kron magnitude yields $\sim 94\%$ of the total source flux within the adaptive aperture (Bertìn & Arnouts, 1996), so accordingly we see a shift of ~ 0.07 mag even in the brightest magnitude bin. On the other hand, isophotal magnitude depends on the detection threshold and Model magnitudes (obtained through a sum of bulge plus disk) produce also for stars an unbiased estimate of the total magnitude.
In conclusion the new PSF modeling of SExtractor produces photometric measurements as accurate and complete as those obtained with DAOPHOT.

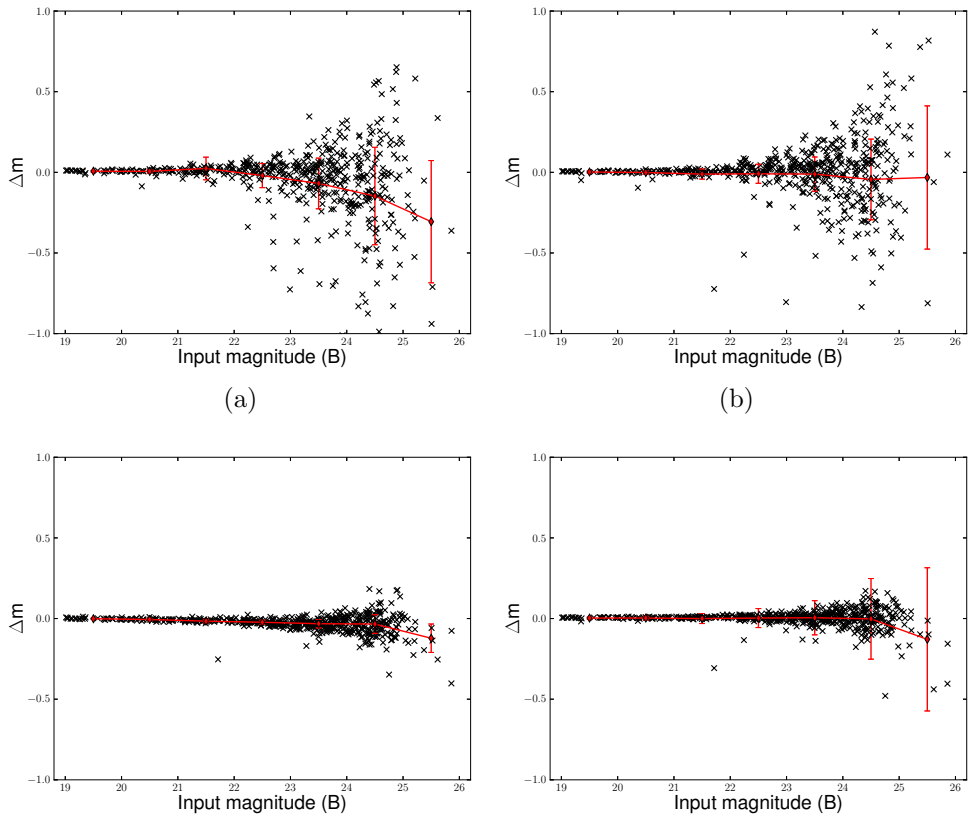


Figure 3.8. *Top panels*: Residuals between aperture magnitudes estimated by DAOPHOT (*left panel*) and by SExtractor (*right panel*), and input magnitudes for detected stars. *Bottom panels*: Residuals between PSF magnitude estimated by DAOPHOT (*left panel*) and by SExtractor (*right panel*), and input magnitude for detected stars. Superimposed red points and solid red lines draw the mean and standard deviation values reported in Tab. 3.3.

3.3.4 Centroids

The last comparison was among extracted positions and input magnitudes, there are different ways to obtain centroids measurements. As stated above, DAOPHOT can provide two different measurements for centroids. The simplest are the coordinates of the barycenter of the source and are derived during the thresholding process. These coordinates can be redetermined by ALLSTAR, once DAOPHOT has build a PSF model, applying a PSF correction.
Concerning SExtractor, we chose to compare the results obtained using the

Bin (mag)	Δm_{mean} (mag)	$\sigma_{\Delta m}$ (mag)	Δm_{mean} (mag)	$\sigma_{\Delta m}$ (mag)	Δm_{mean} (mag)	$\sigma_{\Delta m}$ (mag)	Δm_{mean} (mag)	$\sigma_{\Delta m}$ (mag)
	(a)		(b)		(c)		(d)	
19 - 20	0.007	0.005	0.002	0.006	-0.001	0.003	0.003	0.003
20 - 21	0.006	0.010	-0.001	0.011	-0.006	0.005	0.003	0.004
21 - 22	0.024	0.070	-0.011	0.031	-0.016	0.011	0.000	0.012
22 - 23	-0.020	0.075	-0.009	0.059	-0.023	0.013	0.003	0.014
23 - 24	-0.069	0.157	-0.011	0.106	-0.032	0.026	0.005	0.025
24 - 25	-0.147	0.302	-0.044	0.250	-0.034	0.058	-0.002	0.055
25 - 26	-0.306	0.379	-0.032	0.444	-0.122	0.088	-0.129	0.129

Table 3.3. The table reports, as a function of the magnitude bin (col. 1), the mean difference Δm_{mean} (col.s 2, 4, 6 and 8), and the standard deviation $\sigma_{\Delta m}$ (col.s 3, 5, 7 and 9) between aperture magnitudes as estimated by DAOPHOT in part a and by SExtractor in part b and input magnitudes, and PSF magnitudes obtained by using DAOPHOT in part c and by SExtractor in part d, and input magnitudes.

Bin (mag)	Δm_{mean} (mag)	$\sigma_{\Delta m}$ (mag)	Δm_{mean} (mag)	$\sigma_{\Delta m}$ (mag)	Δm_{mean} (mag)	$\sigma_{\Delta m}$ (mag)
	(a)		(b)		(c)	
19 - 20	0.074	0.005	0.058	0.007	0.057	0.023
20 - 21	0.077	0.009	0.073	0.012	0.048	0.025
21 - 22	0.076	0.027	0.093	0.024	0.028	0.046
22 - 23	0.076	0.046	0.128	0.039	-0.002	0.074
23 - 24	0.087	0.065	0.216	0.052	-0.034	0.091
24 - 25	0.051	0.143	0.389	0.121	-0.098	0.146
25 - 26	0.147	0.195	0.749	0.143	-0.145	0.151

Table 3.4. The table we report, as a function of the magnitude bin (col. 1), the mean difference Δm_{mean} (col.s 2, 4 and 6), and the standard deviation $\sigma_{\Delta m}$ (col.s 3, 5 and 7) between Kron (part a), isophotal (part b), and model (part c) magnitudes obtained by using SExtractor, and input magnitude.

barycenter and the PSF corrected coordinates, as for DAOPHOT, and the results obtained with the windowed position estimates along both axes. These coordinates are obtained by integrating pixel values within a circular Gaussian window. In Tab. 3.5 it is reported the mean difference between barycenter coordinates and PSF corrected coordinates, estimated respectively with DAOPHOT (parts a and b) and SExtractor (parts c and d) and input coordinates.
Finally, Tab. 3.6 shows the difference between windowed estimated by SExtractor and input coordinates.

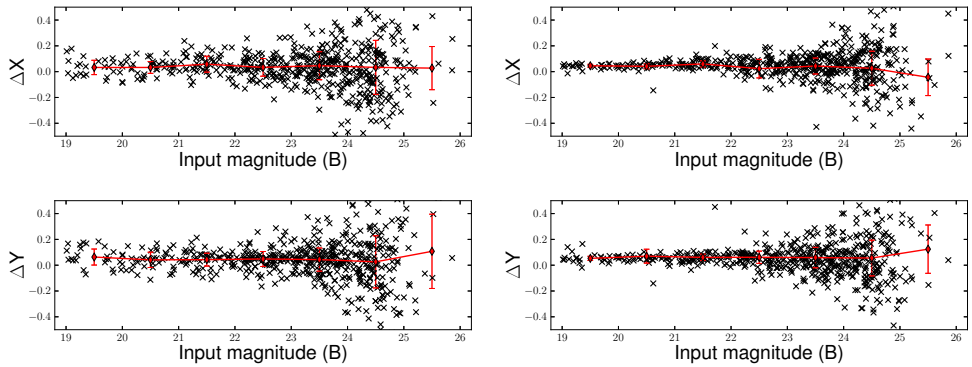
Bin (mag)	ΔX_{mean} (pixel)	$\sigma_{\Delta X}$ (pixel)	ΔY_{mean} (pixel)	$\sigma_{\Delta Y}$ (pixel)	ΔX_{mean} (pixel)	$\sigma_{\Delta X}$ (pixel)	ΔY_{mean} (pixel)	$\sigma_{\Delta Y}$ (pixel)
	(a)				(b)			
19 - 20	0.033	0.055	0.063	0.062	0.046	0.011	0.053	0.016
20 - 21	0.032	0.045	0.041	0.059	0.041	0.019	0.070	0.054
21 - 22	0.058	0.061	0.043	0.050	0.060	0.024	0.061	0.024
22 - 23	0.033	0.068	0.048	0.058	0.023	0.072	0.061	0.047
23 - 24	0.046	0.108	0.043	0.090	0.043	0.062	0.059	0.079
24 - 25	0.033	0.209	0.026	0.202	0.026	0.132	0.055	0.138
25 - 26	0.027	0.167	0.107	0.288	-0.044	0.142	0.124	0.187
	(c)				(d)			
19 - 20	0.050	0.005	0.050	0.004	0.051	0.005	0.050	0.003
20 - 21	0.052	0.009	0.051	0.006	0.051	0.007	0.051	0.006
21 - 22	0.052	0.009	0.053	0.014	0.054	0.009	0.053	0.014
22 - 23	0.044	0.020	0.054	0.022	0.044	0.022	0.052	0.022
23 - 24	0.043	0.048	0.046	0.053	0.044	0.044	0.049	0.052
24 - 25	0.031	0.116	0.043	0.111	0.033	0.114	0.047	0.109
25 - 26	-0.096	0.149	0.097	0.189	-0.007	0.186	0.107	0.269

Table 3.5. The table reports, as a function of the the magnitude bin (col. 1), the mean difference between DAOPHOT X (col. 2),Y (col. 4) barycenter measure and input X,Y and the relative standard deviation (col.s 3 and 5) in the part a, while in the part b there are the mean difference between SExtractor X (col. 6),Y (col. 8) barycenter measure and input X, Y and the relative standard deviation (col.s 7 and 9). In parts c and d are reported the mean difference between X (col.s 2 and 6), Y (col.s 4 and 8) PSF corrected measurements obtained by using DAOPHOT and SExtractor, respectively, and input X,Y, and the relative standard deviation (col.s 3, 5, 7 and 9).

Bin (mag)	ΔX_{mean} (pixel)	$\sigma_{\Delta X}$ (pixel)	ΔY_{mean} (pixel)	$\sigma_{\Delta Y}$ (pixel)
19 - 20	0.050	0.004	0.050	0.004
20 - 21	0.051	0.006	0.051	0.007
21 - 22	0.054	0.010	0.056	0.016
22 - 23	0.032	0.036	0.054	0.025
23 - 24	0.044	0.046	0.052	0.057
24 - 25	0.028	0.127	0.040	0.120
25 - 26	-0.043	0.124	0.134	0.173

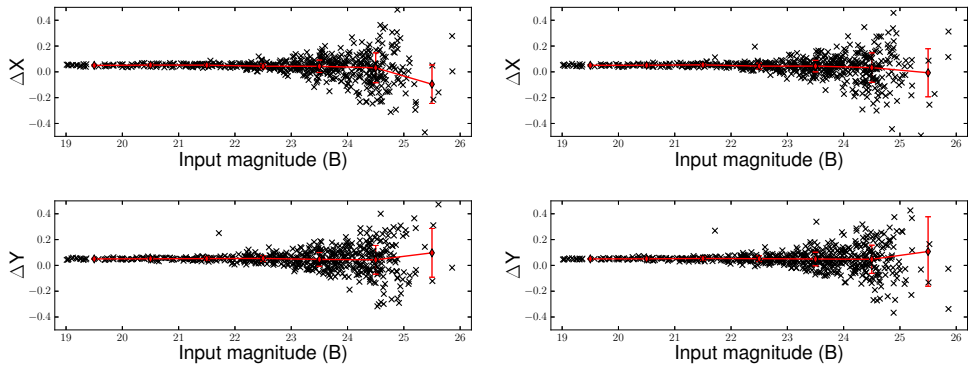
Table 3.6. The table reports, as a function of the the magnitude bin (col. 1), the mean difference between X (col. 2) and Y (col. 4) windowed measurements as estimated by SExtractor and input X,Y and the relative standard deviation (col.s 3 and 5).

Fig. 3.9 and 3.10 show the difference between the input coordinates and barycenter coordinates and between the input coordinates and PSF corrected coordinates.



(a) (b)

Figure 3.9. Difference between barycenter coordinates estimated by DAOPHOT (*left panels*) and by SExtractor (*right panels*), and input coordinate and as a function of input magnitude for detected stars. Superimposed red points and solid red lines draw the mean and standard deviation values reported in top part of Tab. 3.5.



(a) (b)

Figure 3.10. Difference between PSF corrected coordinates estimated by DAOPHOT (*left panels*) and by SExtractor (*right panels*), and input coordinate and as a function of input magnitude for detected stars. Superimposed red points and solid red lines draw the mean and standard deviation values reported in bottom part of Tab. 3.5.

Both software show a bias between output centroid coordinates ≤ 0.01 arcsec (equal to ~ 0.47 pixel) and input X and Y with an average deviation of $\leq$

0.02 arcsec (equal to $\sim$ 0.94 pixel) down to the DAOPHOT completeness magnitude limit. These values are improved in particular in terms of average deviation ($\sigma_{\Delta X(Y)} \leq 0.01$ arcsec) when PSF correction is applied. Hence we can conclude that the results on centroids are satisfactory in both cases.

3.4 Implications

Considering only the number of extracted sources we saw that the limiting magnitude for the extracted catalog is extremely low, in particular the limiting magnitude for DAOPHOT is B=22 mag. Instead, if we limit to consider only stellar sources the photometric depth is improved down to 24 mag for DAOPHOT and 26 mag for SExtractor. This last value is the magnitude limit of input simulated catalogs.
A relevant aspect of the catalog extraction is the capability to discriminate between extended and point-like sources. As we have seen, within the different software, there are various methods to perform the Star/Galaxy classification. In particular the sharpness parameter available with DAOPHOT and improved by the use ALLSTAR returns a reliable Star/Galaxy classification down to the photometric depth of the catalog (B=24 mag). All the traditional methods available in SExtractor, instead, limit the Star/Galaxy classification, at least, one magnitude above the completeness magnitude of the catalog. The new parameter `SPREAD_MODEL`, which is a discriminant between the best fitting local PSF and a more extended model, has largely improved the classification, allowing to separate extended and point-like sources up the completeness limit of the catalog (which is B=26 mag considering only stellar sources).
Since DAOPHOT is mainly designed to perform stellar photometry, in order not to bias the comparison of photometric measurements, we consider only input stellar sources recovered by both software. Considering only stellar photometry, both software are able to deliver acceptable performances in both aperture (with a $\sigma_{\Delta \mathrm{m}} < 0.2$ mag) and PSF photometry (with a $\sigma_{\Delta \mathrm{m}} < 0.03$ mag) , down to B=24 mag, a threshold two magnitude brighter than the limiting magnitudes of input simulated images. This threshold corresponds exactly to the completeness limit of the DAOPHOT catalog. Moreover, since SExtractor allows to derive different estimates of the total magnitudes of sources, we also compare among themselves: Kron, isophotal and model magnitudes. The isophotal magnitude is highly dependent from the detection threshold and in fact we note in [23-24] magnitude bin a higher shift of Δm (up to 0.216 mag) respect to zero than in other magnitudes. The

Kron magnitude yields $\sim 94\%$ of the total source flux within the adaptive aperture at 94% (Bertìn & Arnouts, 1996) so accordingly we see a shift of $\sim$ 0.07 mag even in the brightest magnitude bin. The model magnitude results a good estimate of the input magnitude also for stars, with an error of 0.091 mag in the [23-24] mag bin.
An accurate determination of the object's centroids is crucial in particular for relative astrometry and thus, also for matching sources in different bands or in different epochs. Both software show a bias between output centroid and input X and Y coordinates ≤ 0.01 arcsec with an average deviation of ≤ 0.02 arcsec down to the DAOPHOT completeness magnitude limit of the extracted catalog. These values are improved in particular in terms of average deviation ($\sigma_{\Delta X(Y)} \leq 0.01$ arcsec) when PSF correction is applied. So we can conclude that the results are satisfactory in both cases.
DAOPHOT and ALLSTAR provide a very accurate and reliable PSF photometry, with robust star-galaxy separation. However it is not useful for galaxy characterization. On the other hand SExtractor associated with PSFEx turns competitive in terms of PSF photometry. It returns acceptable aperture photometry and accurate PSF modeling also for faint sources. The windowed centroids are as good as PSF centroids. Moreover SExtractor allows to go very deep in source detection through a properly choose of image filtering masks; the deblending model is very extensible; and the use of neural networking for object classification plus the novel `SPREAD_MODEL` parameter push down to the limiting magnitude, the potentiality of star/galaxy separation. Considering that SExtractor returns accurate photometry also for galaxies, we can conclude that the new version of SExtractor used in combination with PSFEx represents a very powerful software for source extraction with performances comparable to DAOPHOT also for stellar fields.
In the future it would be hopefully to extensively test this SExtractor plus PSFEx on real crowded stellar fields in order to definitively assess the performances of this software. However, an important aspect for the use of PSFEx and SExtractor, we cannot avoid to mention the processing time. Without considering problems such as degradation in performances during periods of heavy disk access, on average, SExtractor requires $0.5s$ per detection to perform PSF photometry and source modeling by using one single CPU with 6GB of RAM. This suggests that actually, the only disadvantage of using SExtractor and PSFEx on wide field images is the processing time. However, on the other hand, although DAOPHOT is more efficient in terms of processing time just for the calculation, it requires more time if the user would like to visually inspect the modeled PSF stars.

Taking in to account the results of all these tests, for the simulation pipeline

proposed in Fig. 2.2 and for the testing phase we used SExtractor with PSFEx for catalog extraction.

Chapter 4

The Classifiers

As we said in Chapter 2 among the scopes of this work there was also to test several data mining algorithms in order to find the one optimized for each type of variable object.
The algorithms we used in this work belong to the rather wide category of Neural Networks which have been used for classification tasks in a variety of scientific and non scientific domains.
The term Neural Network refers to an artificial system of information processing methods that attempt to simulate the functional mechanisms at the base of the human brain (Bishop 2006).
Neural Networks are mathematical models which define a function f: $\mathbf{X} \rightarrow \mathbf{Y}$, between a set of input variables (also called features) and a set of output variables (the targets). This function f(x) can be defined as a composition of other functions $g_i(x)$. A widely used type of composition is the nonlinear weighted sum, $f(\mathbf{x}) = K\left(\sum_i w_i g_i(\mathbf{x})\right)$, where K (commonly referred to as the activation function) is some predefined function.
What is most interesting of the Neural Networks is their possibility to learn. Given a specific task to solve, and a class of functions F, learning means using a set of observations to find $f^* \in F$ which solves the task in some optimal sense. There are two different learning paradigms for a neural network: supervised and unsupervised. We can use a supervised method when we have a training set including typical examples of the inputs and the corresponding outputs: in this way the network can learn how to infer the relation between the input and the output variables. Then, the network is trained by using a variety of suitable learning rules (such as the well known Back Propagation; Bishop 2006), which use the input-output data samples (also called patterns) in order to modify the internal weights and other parameters of the network itself in order to minimize an error function, representing the

training error. If the training is successful, the network learns to recognize the unknown relationship between the input and the output variables, and is therefore able to make predictions on new input samples even if their output is not known a priori (generalization capability).
An unsupervised learning method, instead, is based on training algorithms that modify the weights of the network making reference only to a set of data that includes the only input variables. These algorithms attempt to group the input data by making use of topological or probabilistic methods.

4.1 Multi Layer Perceptron (MLP)

In this work we used an implementation of a Multi Layer Perceptron (MLP; Bishop , 1996). The MLP architecture is one of the most typical feed-forward neural network model. The term feed-forward is used to identify basic behavior of such neural models, in which the impulse is propagated always in the same direction, e.g. from neuron input layer towards output layer, through one or more hidden layers (the network brain), by combining weighted sum of weights associated to all neurons (except the input layer). As easy to understand, the neurons are organized in layers, with proper own role. The input signal, simply propagated throughout the neurons of the input layer, is used to stimulate next hidden and output neuron layers. The output of each neuron is obtained by means of an activation function, applied to the weighted sum of its inputs. Different shape of this activation function can be applied, from the simplest linear one up to sigmoid or hyperbolic tangent (tanh). The number of hidden layers represents the degree of the complexity achieved for the energy solution (training error) space in which the network output moves looking for the best solution (the absolute minimum of the training error). As an example, in a typical classification problem, the number of hidden layers, together with their number of neurons, indicates the number of hyper-planes used to split the parameter space (i.e. number of possible classes) in order to classify each input pattern.
In particular in this work we used the MLP coupled with a particular learning rule, known as Quasi Newton Algorithm (QNA), i.e. the MLPQNA method. From a technical point of view, the MLPQNA differs from more traditional MLP's implementations in the way the optimal solution of the classification problem is found. In recent papers, the analytical characteristics and scientific features of the method have been described in several astrophysical contexts of both classification (Brescia et al. , 2012a; Brescia et al. , 2012c) and regression (Cavuoti et al. , 2012).

More in general, accordingly to Bishop (2006), feed forward neural networks (in their various implementations) provide a general framework for representing non linear functional mappings between a set of input variables (also called features) and a set of output variables (the targets). The training of a neural network can be in fact seen as the search for the function which minimizes the errors of the predicted values with respect to the true values available for a small but significant sub-sample of objects in the same data set. This subset is also called "training set" or "knowledge base". The final performances of a specific neural network depend on many factors: the architecture of the neural network, the way the minimum of the error function is searched and found (learning rule), the way errors are computed, the intrinsic quality (signal-to-noise ratio) of the training data as well the statistical distribution of hidden information within training, validation and test sets derived from the available data (also called the Knowledge Base).
The formal description of a feed-forward neural network with two computational layers is given in Eq. 4.1.1:

$$y_k = \sum_{j=0}^{M} w_{kj}^{(2)} g \left(\sum_{i=0}^{d} w_{ji}^{(1)} x_i \right) \tag{4.1.1}$$

Equation 4.1.1 can be better understood by using a graph as the one shown in Figure 4.1. The input layer (x_i) is made of a number of neurons equal to the number of input variables (d); the output layer, on the other hand, will have as many neurons as the output variables (K). In the general case, the network may have an arbitrary number of hidden layers (also known as perceptrons), each of them can be formed by an arbitrary number of neurons (M). In the depicted case there is just one hidden layer as in most real implementations. In a fully connected feed-forward network each node of a layer is connected to all the nodes in the adjacent layers. Each connection is represented by an adaptive weight $\left(w_{kj}^{l}\right)$ which can be regarded as the strength of the synaptic connection between neurons k and j, while the response of each perceptron to the inputs is represented by a non-linear function g, referred to as the activation function.

Eq. 4.1.1 assumes a linear activation function for the neurons in the output layer.

All the above characteristics of the network, the topology and the weight matrix of its connection, define a specific implementation and are usually referred to as to the "model". The model, however, is only part of the story. In fact, in order to find the model that best fits the data in a specific problem,

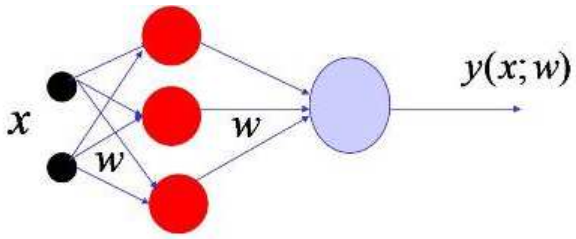


Figure 4.1. The Multi Layer Perceptron general architecture)

one has to provide the network with a set of examples, such as objects for which the final output is known by independent means. These data form the so called training set or Knowledge Base (KB) and through a learning rule are used by the network to find the optimal model.

4.1.1 Learning Rule and Quasi Newton Methods

In our implementation we choose as learning rule the Quasi Newton Algorithm (QNA) which differs from the Newton Algorithm in how the Hessian of the error function is computed. Newtonian models are variable metric methods used to find local maxima and minima of functions (Davidon , 1968) and, in the case of MLPs, they can be used to find the stationary (i.e. the zero gradient) point of the learning function.
Most Newton methods use the Hessian of the function to find the stationary point of a quadratic form. It needs to be stressed, however, that the Hessian of a function is not always available and in many cases it is far too complex to be computed in an analytical way. More often it is easier to compute the function gradient which can be used to approximate the Hessian via N consequent gradient calculations. In order to better understand why QNA are so powerful, it is convenient to start from the classical and quite common Gradient Descent Algorithm (GDA) used for Back Propagation (Bishop , 2006). In GDA, the direction of each updating step for the MLP weights is derived from the error descent gradient, while the length of the step is determined from the learning rate. This method is inaccurate and ineffective and therefore may get stuck in local minima. A more effective approach is to move towards the negative direction of the gradient (line search direction) not by a fixed step, but by moving towards the minimum of the function along that direction. This can be achieved by first deriving the descent gradient and then by analyzing it with the variation of the learning rate (Brescia , 2012b). Let us suppose that at step t, the current weight vector is $w^{(t)}$, and let us consider a search direction $d^{(t)} = -\nabla E^{(t)}$. If we select the parameter

λ in order to minimize $E(\lambda) = E(w^{(t)} + \lambda d^{(t)})$, the new weight vector can be expressed as:

$$w^{(t+1)} = w^{(t)} + \lambda d^{(t)} \tag{4.1.2}$$

and the problem of line search becomes a 1-dimensional minimization problem which can be solved in many different ways. Simple variants are: i) to move $E(\lambda)$ by varying λ by small intervals, then evaluate the error function at each new position, and stop when the error begins to increase, or ii) to use the parabolic search for a minimum and compute the parabolic curve crossing pre-defined learning rate points. The minimum d of the parabolic curve is a good approximation of the minimum of $E(\lambda)$ and it can be derived by means of the parabolic curve which crosses the fixed points with the lowest error values. Another approach makes instead use of *trust region* based strategies which minimize the error function, by iteratively growing or contracting the region of the function by adjusting a quadratic model function which best approximates the error function. In this sense this technique can be considered as a dual to line search, since it tries to find the best size of the region by fixing the step size (while the line search strategy always chooses the step direction before selecting the step size), (Celis et al. , 1985). All these approaches, however, rely on the assumption that the optimal search direction is given at each step by the negative gradient: an assumption which not only is not always true but can also lead to an erroneous convergence. In fact, if the minimization is done along the negative gradient direction, the subsequent search direction (the new gradient) will be orthogonal to the previous one: when the line search finds the minimum, it is:

$$\frac{\partial E}{\partial \lambda}(w^{(t)} + \lambda d^{(t)}) = 0 \tag{4.1.3}$$

and hence,

$$g^{(t+1)T} d^{(t)} = 0 \tag{4.1.4}$$

where $g \equiv \nabla E$. The iteration of the process therefore leads to oscillations of the error function which slow down the convergence process.
The method implemented here relies on selecting other directions so that the gradient component, parallel to the previous search direction, would remain unchanged at each step. Suppose that you have already minimized with respect to the direction $d^{(t)}$ starting from the point $w^{(t)}$ and reaching the point $w^{(t+1)}$, where Eq. 4.1.4 becomes:

$$g(w^{(t+1)})^T d^{(t)} = 0 \tag{4.1.5}$$

by choosing $d^{(t+1)}$ so to preserve the gradient component parallel to $d^{(t)}$ equal to zero, it is possible to build a sequence of directions d in such a way that each direction is conjugated to the previous one in the dimension $|w|$ of the search space (this is known as conjugate gradients method; Golub & Ye (1999)). In presence of a squared error function, the update weights algorithm is:

$$w^{(t+1)} = w^{(t)} + \alpha^{(t)} d^{(t)} \tag{4.1.6}$$

with:

$$\alpha^{(t)} = -\frac{d^{(t)T} g^{(t)}}{d^{(t)T} H d^{(t)}} \tag{4.1.7}$$

Furthermore, d can be obtained for the first time via the negative gradient and in the subsequent iterations, as a linear combination of the current gradient and of the previous search directions:

$$d^{(t+1)} = -g^{(t+1)} + \beta^{(t)} d^{(t)} \tag{4.1.8}$$

with:

$$\beta^{(t)} = \frac{g^{(t+1)T} H d^{(t)}}{d^{(t)T} H d^{(t)}} \tag{4.1.9}$$

This algorithm finds the minimum of a square error function in almost $|w|$ steps but, the at the price of a high computational cost since in order to determine the values of α and β, it makes use of that hessian matrix H which, as we already mentioned is very demanding in terms of computing. A fact which puts serious constraints on the application of this family of methods to medium/large data sets. Excellent approximations for the coefficients α and β can, however, be obtained from analytical expressions that do not use the Hessian matrix explicitly. For instance, β can be calculated through any one of the following expressions (Polak & Ribiere , 1969; Hestenes & Stiefel , 1952; Fletcher & Reeves , 1964):

$$Polak - Ribier : \beta^{(t)} = \frac{g^{(t+1)T}(g^{(t+1)} - g^{(t)})}{g^{(t)T} g^{(t)}} \tag{4.1.10}$$

$$Hestenes - Sitefel : \beta^{(t)} = \frac{g^{(t+1)T}(g^{(t+1)} - g^{(t)})}{d^{(t)T}(g^{(t+1)} - g^{(t)})} \tag{4.1.11}$$

$$Fletcher - Reeves : \beta^{(t)} = \frac{g^{(t+1)T} g^{(t+1)}}{g^{(t)T} g^{(t)}} \tag{4.1.12}$$

These expressions are all equivalent if the error function is square-typed, otherwise they assume different values. Typically the Polak-Ribiere equa-

tion obtains better result because, if the algorithm is slow and subsequent gradients are quite alike between them, it equation produces values of β such that the search direction tends to assume the negative gradient direction (Vetterling & Flannery , 1992).
Concerning the parameter α, its value can be obtained by using the line search method directly. The method of conjugate gradients reduces the number of steps to minimize the error up to a maximum of $|w|$ because there could be almost $|w|$ conjugate directions in a $|w|$-dimensional space. In practice however, the algorithm is slower because, during the learning process, the property *conjugate* of the search directions tend to deteriorate. It is useful, to avoid the deterioration, to restart the algorithm after $|w|$ steps, by resetting the search direction with the negative gradient direction.
By using a local square approximation of the error function, we can obtain an expression for the minimum position. The gradient in every point w is in fact given by:

$$\nabla E = H \times (w - w^*) \tag{4.1.13}$$

where w^* corresponds to the minimum of the error function, which satisfies the condition:

$$w^* = w - H^{-1} \times \nabla E \tag{4.1.14}$$

The vector $-H^{-1} \times \nabla E$ is known as Newton direction and it is the base for a variety of optimization strategies, such as for instance the QNA, which instead of calculating the H matrix and then its inverse, uses a series of intermediate steps of lower computational cost to generate a sequence of matrices which are more and more accurate approximations of H^{-1}.
From the Newton formula (4.1.14) we note that the weight vectors on steps t and $t+1$ are correlated to the correspondent gradients by the formula:

$$w^{(t+1)} - w^{(t)} = -H^{(-1)}(g^{(t+1)} - g^{(t)}) \tag{4.1.15}$$

which is known as *Quasi Newton Condition*. The approximation G is therefore built in order to satisfy this condition. The formula for G is:

$$G^{(t+1)} = G^{(t)} + \frac{pp^T}{p^T\nu} - \frac{(G^{(t)}\nu)\nu^T G^{(t)}}{\nu^T G^{(t)}\nu} + (\nu^T G^{(t)}\nu)uu^T \tag{4.1.16}$$

where the vectors are:

$$p = w^{(t+1)} - w^{(t)}; \nu = g^{(t+1)} - g^{(t)}; u = \frac{p}{p^T\nu} - \frac{G^{(t)}\nu}{\nu^T G^{(t)}\nu} \tag{4.1.17}$$

Using the identity matrix to initialize the procedure is equivalent to consider,

step by step, the direction of the negative gradient while, at each next step, the direction $-Gg$ is for sure a descent direction. The above expression could carry the search out of the interval of validity for the squared approximation. The solution is hence to use the *line search* to found the minimum of function along the search direction. By using such system, the weight updating expression (4.1.6) can be formulated as follows:

$$w^{(t+1)} = w^{(t)} + \alpha^{(t)} G^{(T)} g^{(t)} \tag{4.1.18}$$

where α is obtained by the *line search*.
One of main advantage of QNA, compared with conjugate gradients, is that the *line search* does not require the calculation of α with a high precision, because it is not a critical parameter. Unfortunately, however, again, it requires a large amount of memory to calculate the matrix G ($|w| \times |w|$), for large $|w|$. One way to reduce the required memory is to replace at each step the matrix G with a unitary matrix. With such replacement and after multiplying by g (the current gradient), we obtain:

$$d^{(t+1)} = -g^{(t)} + Ap + B\nu \tag{4.1.19}$$

Note that if the line search returns exact values, then the above equation produces mutually conjugate directions. A and B are scalar values defined as:

$$\begin{aligned} A &= -(1 + \tfrac{\nu^T \nu}{p^T \nu}) \tfrac{p^T g^{(t+1)}}{p^T \nu} + \tfrac{\nu^T g^{(t+1)}}{p^T \nu} \\ B &= \tfrac{p^T g^{(t+1)}}{p^T \nu} \end{aligned} \tag{4.1.20}$$

4.1.2 MLP-QNA

In this work we use our implementation of the Quasi Newton algorithm based on the limited-memory BFGS (L-BFGS; Byrd et al. 1994), where BFGS is the acronym composed of the names of the four inventors (Broyden, 1970; Fletcher, 1970; Goldfarb, 1970; Shanno , 1970).
Summarizing, the algorithm of MLP with QNA is the following:
Let us consider a generic MLP with $w^{(t)}$ the weight vector at time (t).

1. Initialize all weights $w^{(0)}$ with small random values (typically normalized in $[-1, 1]$), set constant ε and $t = 0$;

2. present to the network all training set and calculate $E(w^{(t)})$ as the error function for the current weight configuration;

3. if $t = 0$ then $d^{(t)} = -\nabla E^{(t)}$

4. else $d^{(t)} = -\nabla E^{(t-1)} + Ap + B\nu$, where $p = w^{(t+1)} - w^{(t)}$ and $\nu = g^{(t+1)} - g^{(t)}$;
5. calculate $w^{(t+1)} = w^{(t)} - \alpha d^{(t)}$, where α is obtained by line search equation (4.1.7);
6. calculate A and B for the next iteration, as reported in eq 4.1.20;
7. if $E(w^{(t+1)}) > \varepsilon$ then $t = t + 1$ and goto 2, else STOP.

As it is known, all *line search* methods, being based on techniques searching the minimum error by exploring the error function surface, are likely to get stuck in a local minimum and many solutions have been proposed (Floudas & Jongen , 2005). In order to accelerate the convergence of GDA, Newton's method uses the information on the second-order derivatives. QNA is able to better optimize the convergence time by approximating second-order information with first-order terms (Shanno , 1970).
By having information on the second derivatives, QNA is able to avoid local minima of the error function and to be more precise in the error function trend follow-up, thus revealing a *natural* capability to find the absolute minimum error of the optimization problem.
In the L-BFGS version of the algorithm, in the case of high dimensionality (i.e. input data with many parameters), the amount of memory required to store the Hessian is too big, along with the machine time required to process it. Therefore, instead of using a complete number of gradient values to generate the Hessian, we can use a smaller number of values.
On the one hand, the convergence slows down. On the other hand, the performance could even increase. A statement which only a first sight might seem paradoxical but, while the convergence is measured by the number of iterations, the performance depends on the number of processor's time units spent to calculate the result.
Related to the computational cost there is also the strategy adopted in terms of stopping criteria of the method. As known, the process of adjusting the weights based on the gradients is repeated until a minimum is reached. In practice, one has to decide the stopping condition of the algorithm. More in general, there are several criteria. Among them the most used are: (i) the algorithm could be terminated after the gradient is sufficiently small (by definition the gradient will be zero at a minimum); (ii) based on the error to be minimized, in terms of a fixed threshold; (iii) based on the cross validation. The basic mechanism at the base of any experiment based on machine learning models consists into partitioning data in a train and test set. The network is trained on the training set and its performances are evaluated on the test set.

4.1.3 Training and evaluation of errors

A final series of considerations needs to be made about the training and error evaluation phases. The essence of the test set is to validate the generalization capability of the model on data different from the ones used for training. In the selection of the percentages for the two partitioned subsets we followed what it is considered as a general rule of thumb, indicating the value of 80% and 20%, respectively, for training and test sets (Kearns , 1996). However, by this simple partitioning there could be the possibility that the model may suffer of an overfitting on the validation dataset. The first two criteria mentioned above are mainly sensitive to the choice of specific parameters and may lead to poor results if the parameters are improperly set. The cross validation do not suffer of such drawback. It can avoid overfitting the data and is able to improve the generalization performance of the model. However it is much more computationally expensive. The cross validation can be used to monitor generalization performance during training and to terminate the algorithm when there is no more improvement. Statistically significant results come out by trying multiple independent data partitions and averaging the performance. There are several variants of cross validation methods (Sylvain & Celisse , 2010). We in particular have chosen the k-fold cross validation, particularly suited in presence of a scarcity of known data samples (Geisser , 1975). The mechanism, also known as *leave-one-out*, is quite simple, by dividing the training set of N samples into k subsets ($k > 1$). The model is then trained on $N - 1$ subsets and validated by testing it on the left out subset. This procedure is then iterated each time leaving out a different subset for validation and its squared error is averaged on all cycles.

4.2 The experiments

As we explained in Chapter 2 the strategy of our project is to use a hierarchical approach to variable object classification. This approach has the typical decision tree structure and aims at a classification which becomes finer and finer as we to higher level of branching. The first level of this approach is to perform the crispy classification based on the variable/not variable object dichotomy. We choose to use the MLP-QNA algorithm, since it is the one which provides best results to several astrophysical problems (Brescia et al. , 2012a, Brescia et al. , 2012c) and the one which deals better with poorly populated datasets.
In the following sections there are presented the data used for these tests and the strategy behind the choice of the parameters for the classifier.

4.2.1 The data

To test the MLP-QNA algorithm for the variable object classification we built a set of four simulations, each of them consisting in 50 images, corresponding to 50 different epochs, spaced within 90 days with an uneven sampling rate. The instrumental characteristics are the same for each simulation and fine tuned to the characteristics of VST optics and instrumentation (see Sect. 2.7). The characteristics of the detector, such as the gain and saturation level are chosen equal to those defined in Sect. 2.7, while the image size varied between the simulations. The magnitude range is set to 14-25 mag, in order to remain within the magnitude limit of SExtractor. The seeing FWHM is chosen to vary randomly between 0.6 and 1.0 arcsec (respectively medium and worst conditions at VST site, Cerro Paranal, Chile), while the exposure time is set to 1500s.
The types of objects simulated in our images are: non variable stars, and galaxies, Cepheids, type Ia Supernovae with their host galaxies and random variable objects approximating the behavior of eruptive variables and Active galactic nuclei. As described in Sect. 2.5.2, every SN is associated to a nearby galaxy. This implies that in some cases the SN is so close to the nucleus of the parent galaxy that the extraction software fails in deblending the Supernova. In these cases the whole galaxy appears to be variable due to the contribution of the SN (Fig. 4.2).

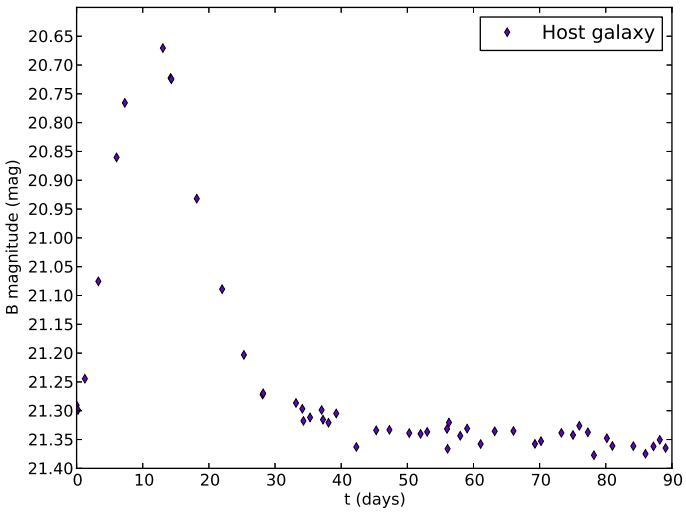


Figure 4.2. Light curve of a Host galaxy, whose Supernova is not detected by the extraction software.

Tables 4.1- 4.4 summarize the number of the objects in each simulation, divided in their categories, i.e. the total catalog, the train and test sets respectively. The number of the objects is obviously different in each simulation, as result of the different image size. Fig 4.3 shows a stamp of the B image at t=0d for the fourth simulation.
For each simulation we obtain a train and a test catalog. The train set contains $\sim$ 80 % of the total number of the objects, being careful to assign a Supernova and its host galaxy to the same set.

OBJECTS	TYPE	FULL	TRAIN	TEST
Variable	SN Ia	80	64	16
	Cepheids	80	64	16
	Random	80	64	16
	Host Galaxy with SN	80	64	16
	Host Galaxy without SN	11	8	3
Not variable	Stars	216	172	44
	Galaxies	1259	1007	252
TOTAL		1806	1443	363

Table 4.1. Number of objects in the first simulation. For each class of objects, the col. 3 shows the quantities in the entire simulation, while col. 4 and 5 show the number of the objects in train and test set respectively.

OBJECTS	TYPE	FULL	TRAIN	TEST
Variable	SN Ia	206	160	46
	Cepheids	200	160	40
	Random	200	160	40
	Host Galaxy with SN	206	160	46
	Host Galaxy without SN	23	18	5
Not variable	Stars	617	493	124
	Galaxies	3510	2808	702
TOTAL		4956	3959	1003

Table 4.2. Number of objects in the second simulation. For each class of objects, the col. 3 shows the quantities in the entire simulation, while col. 4 and 5 show the number of the objects in train and test set respectively.

OBJECTS	TYPE	FULL	TRAIN	TEST
Variable	SN Ia	1081	678	144
	Cepheids	1100	677	173
	Random	1100	702	148
	Host Galaxy with SN	1081	678	172
	Host Galaxy without SN	18	11	3
Not variable	Stars	1374	1093	281
	Galaxies	6580	5245	1333
TOTAL		12334	9084	2254

Table 4.3. Number of objects in the third simulation. For each class of objects, the col. 3 shows the quantities in the entire simulation, while col. 4 and 5 show the number of the objects in train and test set respectively.

OBJECTS	TYPE	FULL	TRAIN	TEST
Variable	SN Ia	1079	681	169
	Cepheids	1099	670	180
	Random	1100	705	145
	Host Galaxy with SN	1079	681	169
	Host Galaxy without SN	19	12	7
Not variable	Stars	1387	1099	288
	Galaxies	6576	5263	1313
TOTAL		12339	9111	2271

Table 4.4. Number of objects in the forth simulation. For each class of objects, the col. 3 shows the quantities in the entire simulation, while col. 4 and 5 show the number of the objects in train and test set respectively.

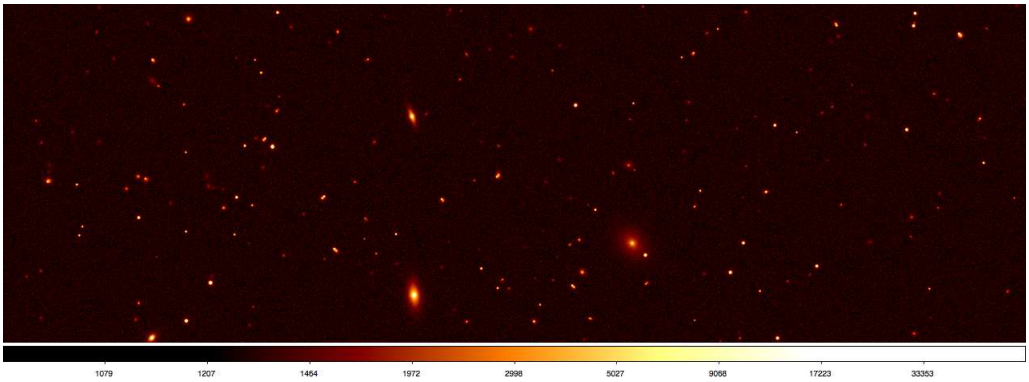


Figure 4.3. Stamp of the B band image at t=0d for the fourth simulation.

4.2.2 Choice of parameters for MLP-QNA

In this first phase of the project we choose to use as parameters for MLP-QNA a set of magnitudes, (Kron magnitudes as estimated by the source extraction software), and times at which they are measured (m_i, t_i) . In order to be more congruent with a real case, and in order to reduce the computational time, we can not use (m_i, t_i) for all the available epochs, but we have to select a subset of epochs. How many epochs, and how they must be chosen, must be carefully evaluated. There are at least three possibilities to take into account:

1. N epochs randomly extracted equal for each object;
2. N epochs randomly extracted different for each objects;
3. N epochs equally spaced equal dor each objects.

Once selected the epochs according to one of these possibilities, we have to be sure that among them each object has at least one measure of magnitude. It is possible, in fact, that the object is not always detected from source extraction software. This happens when the magnitude of the object is near the limit in magnitude of the source extraction software. It is possible that due to their magnitude variation a variable object is detected only in some epochs. If an object, in the train or in test set, does not have any measure for the magnitude in the chosen epochs , it is rejected.

4.3 Results

The results of each test performed by the MLP-QNA on each simulation are discussed below in terms of three evaluation criteria: *accuracy, purity* and *contamination*. The accuracy (CA) is the fraction of objects correctly classified (either variable or not-variable), with respect to the total number of objects in the sample. The purity (CO) is the fraction of variable objects correctly classified as variable. The contamination is the fraction of not variable objects erroneously classified as variable.

Test 1

In the first test we used a simulation dataset of 1806 objects, consisting of 1443 objects as training set and 363 objects as test set, as shown in Tab. 4.1.

In terms of the internal parameter setup of the MLPQNA, we used the following topological parameters:

- **MLP network topology**: a three layer MLP, respectively input, hidden and output layer.
- **input layer**: 20 (corresponding to the number of input features of each pattern);
- **hidden layer**: 41. It is the number of hidden neurons, depending on the number N of input neurons (features in the dataset), equal to $2N + 1$ as rule of thumb;
- **output layer**: 2 (number of classes).

For the QNA learning rule we fixed the following values as best parameters:

- **step**: 0.0001 (one of the two stopping criteria. The algorithm stops if the approximation error step size is less than this value. A step value equal to zero means to use the parameter MaxIt as the unique stopping criterion);
- **res**: 30 (number of restarts of Hessian approximation from random positions, performed at each iteration);
- **dec**: 0.01 (regularization factor for weight decay. The term $dec * ||network\,weights||^2$ is added to the error function, where $network\,weights$ is the total number of weights in the network, directly depending on the total number of neurons inside. When properly chosen, the generalization performances of the network are highly improved);
- **MaxIt**: 3000 (max number of iterations of Hessian approximation. If zero the step parameter is used as stopping criterion);
- **CV(k)**: 10 (k-fold cross validation, with $k = 10$);
- **Error evaluation**: Cross Entropy (statistical evaluation between target and network output, by considering the supervised model outputs as posterior probabilities (Rubinstein & Kroese, 2004)).

The numerical results are shown in the confusion matrices referred to respectively, training phase in Tab. 4.5 and test phase in Tab. 4.6.

As we can see in the training case, we obtain:

	Predicted class 1	Predicted class 2
Target class 1	1173	6
Target class 2	9	255

Table 4.5. Confusion matrix of the training performed with the data of the first simulation (Tab. 4.1). Each column of the matrix represents the instances in a predicted class, while each row represents the instances in an actual class. All correct guesses are located in the diagonal of the table.

- total classification percentage: 98.96%
- class 1 (NOT VARIABLES) classification percentage: 99.49%
- class 2 (VARIABLES) classification percentage: 96.59%

	Predicted class 1	Predicted class 2
Target class 1	280	16
Target class 2	29	30

Table 4.6. Confusion matrix of the test performed with the data of the first simulation (Tab. 4.1). Each column of the matrix represents the instances in a predicted class, while each row represents the instances in an actual class. All correct guesses are located in the diagonal of the table.

As we can see in the test case, in terms of statistical indicators, the *accuracy* of the network, which is the ratio between the number of the objects on the diagonal of the matrix and the total number of the objects of the test set, is ∼88 % , while the *purity* is ∼ 57 %. The *contamination* of the experiment is about ∼ 5%.

Test 2

In the second test we used a simulation dataset of 4956 objects, consisting of 3959 objects as training set and 1003 objects as test set, as shown in Tab. 4.2. The internal parameter setup of the MLPQNA has be set as in Test 1.
The numerical results are shown in the confusion matrices referred to respectively, training phase in Tab. 4.7 and test phase in Tab. 4.8.

As we can see in the training case, we obtain:

- total classification percentage: 98.92%

	Predicted class 1	Predicted class 2
Target class 1	3274	27
Target class 2	16	655

Table 4.7. Confusion matrix of the training performed with the data of the second simulation (Tab. 4.2). Each column of the matrix represents the instances in a predicted class, while each row represents the instances in an actual class. All correct guesses are located in the diagonal of the table.

- class 1 (NOT VARIABLES) classification percentage: 99.18%
- class 2 (VARIABLES) classification percentage: 97.61%

	Predicted class 1	Predicted class 2
Target class 1	766	60
Target class 2	58	123

Table 4.8. Confusion matrix of the test performed with the data of the second simulation (Tab. 4.2). Each column of the matrix represents the instances in a predicted class, while each row represents the instances in an actual class. All correct guesses are located in the diagonal of the table.

The *accuracy* of the network in this case remains ~ 88 % , while the *purity* increases of about $\sim 10\%$ by doubling the dimension of the dataset, reaching a value of ~ 67 %. The *contamination* of the experiment is about $\sim 7\%$.

Test 3

In the third test we used a simulation dataset of 12334 objects, consisting of 9084 objects as training set and 2254 objects as test set, as shown in Tab. 4.3. The internal parameter setup of the MLPQNA has be set as in Test 1.
The numerical results are shown in the confusion matrices referred to respectively, training phase in Tab. 4.9 and test phase in Tab. 4.10.

As we can see in the training case, we obtain:

- total classification percentage: 95.63%
- class 1 (NOT VARIABLES) classification percentage: 91.44%
- class 2 (VARIABLES) classification percentage: 97.44%

	Predicted class 1	Predicted class 2
Target class 1	2511	235
Target class 2	162	6176

Table 4.9. Confusion matrix of the training performed with the data of the third simulation (Tab. 4.3). Each column of the matrix represents the instances in a predicted class, while each row represents the instances in an actual class. All correct guesses are located in the diagonal of the table.

	Predicted class 1	Predicted class 2
Target class 1	384	393
Target class 2	244	1233

Table 4.10. Confusion matrix of the test performed with the data of the third simulation (Tab. 4.3). Each column of the matrix represents the instances in a predicted class, while each row represents the instances in an actual class. All correct guesses are located in the diagonal of the table.

The *accuracy* of the network decreases to $\sim$ 72 % . The *purity* is $\sim$ 83 %. The *contamination* increases to $\sim$ 51%.

Test 4

In the forth test we used a simulation dataset of 12339 objects, consisting of 9111 objects as training set and 2271 objects as test set, as shown in Tab. 4.4.

In terms of the internal parameter setup of the MLPQNA, we used the following topological parameters:

- **MLP network topology**: a four layer MLP, respectively input, hidden1, hidden2 and output layer;
- **input layer**: 20 (corresponding to the number of input features of each pattern);
- **first hidden layer**: 41. It is the number of hidden neurons, depending on the number N of input neurons (features in the dataset), equal to $2N + 1$ as rule of thumb;
- **second hidden layer**: 20;
- **output layer**: 2 (number of classes).

For the QNA learning rule, after several trials, we fixed the following values as best parameters:

- **step**: 0.001 (one of the two stopping criteria. The algorithm stops if the approximation error step size is less than this value. A step value equal to zero means to use the parameter MaxIt as the unique stopping criterion);
- **res**: 30 (number of restarts of Hessian approximation from random positions, performed at each iteration);
- **dec**: 0.01 (regularization factor for weight decay. The term $dec * ||network\,weights||^2$ is added to the error function, where $network\,weights$ is the total number of weights in the network, directly depending on the total number of neurons inside. When properly chosen, the generalization performances of the network are highly improved);
- **MaxIt**: 4000 (max number of iterations of Hessian approximation. If zero the step parameter is used as stopping criterion);
- **CV(k)**: 10 (k-fold cross validation, with $k = 10$);
- **Error evaluation**: Cross Entropy (statistical evaluation between target and network output, by considering the supervised model outputs as posterior probabilities (Rubinstein & Kroese, 2004)).

The numerical results are shown in the confusion matrices referred to respectively, training phase in Tab. 4.11 and test phase in Tab. 4.12.

	Predicted class 1	Predicted class 2
Target class 1	2645	104
Target class 2	83	6278

Table 4.11. Confusion matrix of the training performed with the data of the fourth simulation (Tab. 4.4). Each column of the matrix represents the instances in a predicted class, while each row represents the instances in an actual class. All correct guesses are located in the diagonal of the table.

As we can see in the training case, we obtain:

- total classification percentage: 97.9473%
- class 1 (NOT VARIABLES) classification percentage: 96.22%
- class 2 (VARIABLES) classification percentage: 98.69%

	Predicted class 1	Predicted class 2
Target class 1	439	231
Target class 2	266	1335

Table 4.12. Confusion matrix of the test performed with the data of the fourth simulation (Tab. 4.4). Each column of the matrix represents the instances in a predicted class, while each row represents the instances in an actual class. All correct guesses are located in the diagonal of the table.

The *accuracy* of the network is $\sim$ 78 %, while the *purity* is $\sim$ 65 %. The *contamination* is $\sim$ 40%.
In Tab. 4.13 there are summarized the results obtained in all four simulation. For completeness we report also the results obtained with the train set, although less relevant for the quality evaluation. In fact, it is clear that the only relevant results are those obtained with the test sets, that have not been used for the network training process.

	TRAIN			**TEST**		
	CA %	**CO %**	**CN %**	**CA %**	**CO %**	**CN %**
SIM 1	**98,96**	**96,59**	**0,51**	**87,60**	**56,72**	**5,41**
SIM 2	**98,92**	**97,61**	**0,82**	**88,28**	**67,96**	**7,26**
SIM 3	**95,63**	**97,44**	**8,56**	**71,74**	**83,48**	**50,58**
SIM 4	**97,95**	**96,22**	**1,30**	**78,12**	**65,52**	**39,70**

Table 4.13. Train and test recognition rates for the four simulation described in Sect. 4.2.

As we can see from the previous tests and in Tab. 4.13, when increasing the sample of objects there is a large increase in the contamination, while not obtaining a significant improvement in terms of accuracy and purity of the network. This also by exploring slight differences in the model setup.
These results, although not exalting, are to be considered very preliminary. In particular, what affects the classification is the choice of the features, that in this case is carrying poor information in terms of feature correlation. We also decided to use MLPQNA, one of the more robust existing classification empirical models, based on the machine learning supervised paradigm, in order to exclude, with a high confidence, the possibility that poor results could be associated to the selected model. So far, future developments of the work will consists basically into the investigation whether the classification may

be improved by using more fine statistical features, indirectly derived from the light curves, and not simply the light curves themselves.

Chapter 5

Final results and Future developments

Nowadays, the new generation of observing facilities and dedicated surveys (either wide-field or deep-field) has opened the era of the data-driven astrophysical science, where wavelength, multi epoch, high accuracy data are routinely collected for billions of objects. Among the wide-field surveys a special place is reserved, in recent years, to synoptic surveys, which repeatedly observe the same regions of the sky with a sampling rate sensitive to astronomical phenomena that change over time. Synoptic surveys have to face two major problems: detection and physical classification of both astrometric and photometric transients. The present inter-disciplinary work, ranging from astrophysics to data mining and information technology, was made in the framework of an international Collaboration (DAME), aiming at exposing service-oriented tools for knowledge discovery in astrophysics. Along this work, we discussed the design and development of an automatic workflow to generate astronomical images with an user-defined number and type of variable objects, in order to perform setup and calibration of classification models running on the real images coming from observations. The original contribution obtained by the present work presents several interesting aspects, useful and helpful to engage a virtuous, rigorous and systematic exploration of huge volumes of observed data, enabling the discovery and classification of sky transient objects.

We presented a modular simulation framework, based on transient semantic taxonomy and their physical modeling, on the detection instrument setup and on the exploitation of powerful and reliable software tools, well known to the astronomical community, such as Stuff and SkyMaker, integrated in a workflow specialized to variable object realistic representation. In this context, we have successfully modeled and simulated a preliminary subset of variable objects, for instance Cepheids and type Ia Supernovae, populating

realistic multi-band images, as observed by the VST (VLT Survey Telescope) and OmegaCAM instruments. It is important to stress that this simulation environment has been designed as a modular system, easily configurable and expandable, both in terms of new transient types, different detection instrument and observing condition setup. Such framework included also an integrated pipeline for source catalog extraction, based on well suited available packages, such as SExtractor, PSFEx, DAOPHOT and ALLSTAR, in which we performed a deeper performance analysis and comparison, in order to enhance and optimize their capability to detect transients (Annunziatella et al., 2012).
Another important aspect of the present work is represented by the classification of the sky variable objects, in which we started to investigate the application of machine learning methods, available through the DAME Collaboration.
We presented here some preliminary results obtained by a set of experiments, based on the MLPQNA (Multi Layer Perceptron trained by Quasi Newton Algorithm; Brescia et al. , 2012a, Cavuoti et al. , 2012 and Brescia et al. , 2012c) model. The performed experiments, although in a preliminary stage, in which the classification has been based directly on the light curve information (for instance variability induced by time and magnitude features), revealed important issues. In particular, the information correlation, directly coming from transient light curve analysis, appears as a useful but insufficient base of knowledge to obtain a high performance transient with machine learning methodologies.
Despite of a good classification accuracy and sufficient completeness, the degree of contamination (i.e. spurious objects erroneously classified as variable) is too high. In the realistic case, for instance real synoptic surveys, the contamination could be in principle reduced only by considering a very large set of repeated dedicated observations of the sky, too expensive in terms of observing time and not always feasible in terms of instrument observing strategy or conditions (such as, for instance, space borne survey missions).
Therefore, we demonstrated the need to increase the complexity in the creation of a well suited base of knowledge, by including information extracted from the physical properties of transients as well as from time series images and contextual data, grafted in a fine tuned statistical context (such as a bayesian framework). This is the approach we are going to take in consideration in the further work, together with the application of a wider set of classification models (i.e. genetic algorithms, support vector machine and other neural networks available in the DAME Collaboration), and their performance comparisons.

List of Figures

List of Tables

Appendix A

Configuration files

A.1 STraDiWA configuration file

```
#Default configuration file for STraDiWA Version 1.0


#---------------------------Setup Files-----------------------------------------------------


SETUP_FILES ./default.stuff,./defaultVST.sky,./default.sex
                                                    #Name and path of configuration files
                                                       of Stuff, SkyMaker and SExtractor

#---------------------------Stuff Parameters------------------------------------------------


STUFF_CATALOG_NAME B.list,V.list,I.list
                                                #CATALOG_NAME must be different for each band
PASSBAND_OBS sandage/B,sandage/V,johnson/I
                                                                    #Observed passband(s) in Stuff

#--------------------------Common parameters to Stuff and SkyMaker--------------------------


IMAGE_SIZE 16384                                        #Width,[height] of the output frame
MAG_LIMITS 14 25                                        #Allowed range of apparent magnitudes

#--------------------------Common parameters to Stuff, SkyMaker and SExtractor-------------


GAIN 0.53                                                           #Detector gain in e-/ADU
PIXEL_SIZE 0.213                                                    #Size of pixel in arcsec

#------------------------Common parameters to SkyMaker and SExtractor----------------------


SATUR_LEVEL 65535                                                   #Saturation level (ADU)


#-----------------------SkyMaker Parameters-----------------------------------------------


EXPOSURE_TIME 1500.0                                                                    #(s)
BACK_MAG 22.81,21.81,19.78                                  #Background surface brightness
```

```
                                                    (mag/arcsec2), one for each band

#------------Variable Objects-----------------------------------------------------------

VARIABLE 2,2,1100,0
VARIABLE 1,2,1100,0
VARIABLE 0,2,1100


#VARIABLE                    #First character indicates the type of variable object desired
                             #Second character indicates the number of the distribution:
                                  #1) the User can add the variable object manually
                                    by setting its parameters
                                  #2)the parameters of variable object is chosen randomly
                             #The third parameter is the number of the variable
                               objects of this type. If the  previous option is 1, this
                               parameter must be set to 1
                             #Number of variable objects with mean magnitude below
                               the magnitude limits. The same rule as the previous
                               parameter applies.



SAMPLING A,0,5,30,79,145,175,313,341,344,436,528,607,675,677,796,819,823,847,889,894,914,942,
1015,1087,1135,1207,1248,1272,1344,1346,1351,1392,1417,1466,1517,1585,1663,1685,1759,1801,
1824,1855,1877,1924,1944,2020,2064,2093,2116,2137


#SAMPLING                      #First character identifies the selected behavior between:
                                #A)Time series (in hours) from t = 0
                                #B)Fixed sampling with number of days, time(hours),
                                and length of night (hours). Note that parameter time
                                (hours) has to be consistent to exposure_time;

SEEING_FWHM B,0.5,1
#SEEING_FWHM          #First character identifies the selected behavior between:
                                #A) series of specific values
                                #B) random value between min and max, for each image

INSTR_ZEROPOINT 26,26,26     #Instrumental magnitude zero-point (one value for each band)


#Variable Objects available in the current version of STraDiWA
#0) RANDOM                                                    available in each band
#1) Classical Cepheid                                         available  B,V,I band
#2)Type Ia Supernovae                                      available in B,V,R,I band

#FINE
```

Appendix B

C++ Classes

B.1 Variable Object

```
#ifndef VARIABLEOBJ_H
#define VARIABLEOBJ_H
#include<iostream>
#include<vector>
using namespace std;

class VariableObject
{
public:

    VariableObject(){};
    virtual double magnitude(double t, string  Band)=0;
    virtual double Xpos()=0;
    virtual double Ypos()=0;
    virtual int Code()=0;

private:
};
#endif
```

B.2 Random variable object

```
#ifndef RANDOM_H
#define RANDOM_H
#include"VariableObject.h"
#include<iostream>
#include<vector>
using namespace std;

class RandomObj  : public VariableObject
{
    public:

    RandomObj(double Mmin, double Mmax, double l);

    double magnitude(double t, string Band);
    double Xpos(){return Xpos_ ;}
    double Ypos(){return Ypos_ ;}
    int Code(){return 100;}

    private:

    double Mmin_, Mmax_;
    double Xpos_, Ypos_, l_;

};

#endif
```

B.3 Classical Cepheid

```
#ifndef Cepheid_H
#define Cepheid_H
#include"VariableObject.h"
#include<iostream>

using namespace std;

class Cepheid: public VariableObject
{
public:

    Cepheid(double mbi, double P, double A, double Phi, ↩
       double l);
    double magnitude(double t, string Band);
    double Xpos(){return Xpos_ ;}
    double Ypos(){return Ypos_ ;}
    int Code(){return 100;}


private:

    double P_, Phi_, Amp_, EBV, Rb_, Rv_, Ri_, mod_,mbi_,↩
       mvi_,mii_,Mbi_,Mvi_,Mii_,ab,av,ai,b_b,b_v,b_i;
    double Xpos_,Ypos_,l_,om_,arg_;

};

#endif
```

B.4 Type Ia Supernova

```
#ifndef _supernovae_h
#define _supernovae_h
#include"VariableObject.h"
#include<iostream>
using namespace std;

class Supernovae: public VariableObject
{

public:

    Supernovae(double mB, double tB, double l, string ↩
       list, string pixel);
    double magnitude(double t, string Band);
    double Xpos(){return Xpos_ ;}
    double Ypos(){return Ypos_ ;}
    int Code(){return 100;}

private:

    double mB_,f0B_,tB_,gammaB_,sigma0B_,g0B_,thetaB_,↩
       tauB_,t0B_,tmaxB_;
    double gammaV,g0V,sigma0V,g1V,sigma1V,f0V,thetaV,tauV↩
       ,t0V,t1V;
    double gammaR,g0R,sigma0R,g1R,sigma1R,f0R,thetaR,tauR↩
       ,t0R,t1R;
    double gammaI,g0I,sigma0I,g1I,sigma1I,f0I,thetaI,tauI↩
       ,t0I,t1I;
    double Xpos_,Ypos_,l_;

};
#endif
```

Bibliography

Aihara H., et al., 2011, ApJS, 193, 29

Alcock, C., Allsman, R. A., Axelrod, T. S. 1993, The MACHO Project - a Search for the Dark Matter in the Milky-Way, eds. Soifer, B. T., 43, 291

Annunziatella M., Mercurio M., Brescia M., Cavuoti S., Longo G. 2012, submitted to PASP

Armstrong, B. et al. 2010, Bulletin of the American Astronomical society, vol. 42, pg. 43

Baum, W. A., 1962, in Problems of Extra-Galactic Research, Proceedings from IAU Symposium no. 15, edited by McVittie, G. C., 390.

Becker, A. C., Silvestri, N. M., Owen, R. E. 2007, PASP, 119, 1462

Bertìn, E. & Arnouts, S. 1996, A&AS, 117, 393

Bertìn, E. 2011, ASP Conference Series, eds. Evans I. N., Accomazzi A., Mink D. J., and Rots A.H, 442, 393

Bishop, C.M. 1996. Neural Networks for Pattern Recognition. Oxford University Press, Oxford, England.

Bishop, C. M., 2006, Pattern Recognition and Machine Learning. Springer ISBN 0-387-31073-8.

Bono, G., Caputo, F., Marconi, M., Musella, I. 2000, apj, 715, 277

Bovy, J., et al.; Astrophysical Journal, 749, 41.

Brescia, M., Cavuoti, S., Paolillo, M., Longo, G., Puzia, T., 2012, Monthly Notices of the Royal Astronomical Society, Volume 421, Issue 2, pp. 1155-1165.

Brescia M., 2012, New Trends in E-Science: Machine Learning and Knowledge Discovery in Databases, Horizons in Computer Science Research, Thomas S. Clary (eds.), Series Horizons in Computer Science Vol. 7, Nova Science Publishers, ISBN: 978-1-61942-774-7.

Brescia, M., Cavuoti, S., D'Abrusco, R., Longo, G., Mercurio, A., 2012, Photo-z prediction on WISE-GALEX-UKIDSS-SDSS Quasar Catalogue, based on the MLPQNA model, Submitted to MNRAS.

Broyden, C. G. , 1970, Journ. of the Inst. of Math. and Its Appl., 6, 76.

Byrd, R.H., Nocedal, J., Schnabel, R.B., 1994, Mathematical Programming, 63, 4, pp. 129-156.

Budavari, T., et al., 2001, AJ, 122, 1163

Cavuoti, S., Brescia, M., Longo, G., Mercurio, A. 2012, Submitted at A&A, arxiv:1206.0876v2.

Celis, M., Dennis, J. E., Tapia, R. A., 1985, Numerical Optimization, P. Boggs, R. Byrd and R. Schnabel (eds.), SIAM, Philadelphia USA, pp. 71-82.

Contardo, G., Leibundgut, B., Vacca, W.D 2011, arXiv:0005507

Connolly, A. J., Csabai, I., Szalay, A. S., Koo, D. C., Kron, R. G., & Munn, J. A., 1995, AJ, 110, 2655

Cox, J. P. 1980, Theory of Stellar Pulsation

D'Abrusco et al. 2009, xxxxxxxx

D'Abrusco et al. 2007, xxxxxxxx

Davidon, W.C., 1968, Comput. J. 10, 406.

de Vaucouleurs, G. 1948, Annales d'Astrophysique, 11, 247

Desai, S., Armstrong, R., Mohr, J. J., et al. 2012, ApJ submitted, arXiv:1204.1210

Djorgovski, S.G. and Mahabal, A.A. and Drake, A.J. and Graham, M.J. and Donalek, C.

A. Mahabal, S.G. Djorgovski, S. Marney et al. 2008, New Approaches to Object Classification in Synoptic Sky Surveys, AIP Conf. Ser. 1082, 252

Drake, A.J. and Djorgovski, S.G. and Mahabal, A. 2008, AJ, 696, 870

Drake, A. J. and Beshore, E. and Djorgovski, S. G. 2012, The First Data Release of the Catalina Surveys, 219, 428

Eisenstein, D. 2003, astro-ph, arXiv:0301623

Eyer, L. and Mowlavi, N. 2007, J.Phys.Conf.Ser., 118, 012010

Fletcher, R., Reeves, C. M., 1964, Function minimization by conjugate gradients. Comput. J. 7, 2, 149-154. MR 0187375.

Fletcher, R., 1970, Computer Journal 13: 317.

Floudas, C. A., & Jongen, H. T., 2005, Journal of Global Optimization, Vol. 32, Number 3, 409-415.

Fu, Limin., 1994, Neural Networks in Computer Intelligence. E.M. Munson and L. Goldberg (eds.), McGraw-Hill NY.

Fukugita, M., Ichikawa, T., Gunn, J. E., Doi, M. 1996, AJ, 111, 1748

Gomez, P. L. 2003, AJ, 584, 210

Graham M. J., Djorgovski S. G., Mahabal A., Donalek C., Drake A., Longo A., arXiv:1208.2480

Gunn, L. J. 1996, AJ, 116, 3040

Geisser, S., 1975, Journal of the American Statistical Association, 70 (350), 320-328.

Giannantonio, T., et al., 2006, Phys. Rev. D, 74, 063520

Giannantonio, T., Scranton, R., Crittenden, R. G., Nichol, R. C., Boughn, S. P., Myers, D., & Richards, G. T., 2008, Phys. Rev. D, 77, 123520.

Goldfarb, D., 1970, Mathematics of Computation, 24, 23.

Golub, G. H., & Ye, Q., 1999, SIAM Journal of Scientific Computation, Vol. 21, pp. 1305-1320.

Hennawi, J. F., et al., 2006, AJ, 131, 1.

Hestenes, M. R., Stiefel, E., 1952, Methods of conjugate gradients for solving linear systems. J. Res. Nat. Bur. Standards 49, 6, 409-439. MR 0060307

Ivezic, Z. and Tyson, J. A. and Axelrod, T. 2009, Bulletin of the American Astronomical Society, 41, 460

Jha, S., Garnavich, P., Kirshner, R. 1999, The First Data Release of the Catalina Surveys, 125, 73

Kearns, M., 1996, A Bound on the Error of Cross Validation Using the Approximation and Estimation Rates, with Consequences for Training-Test Split, Neural Information Processing 8, D.S. Touretzky, M,C. Mozer and M.E. Hasselmo (eds.), Morgan Kaufmann, pp. 183-189.

Kron, G. 1980, ApJS, 43, 305

Law, N. M., Kulkarni, S. R., Dekany, R. G. 2009, PASP, 131, 1395

Lawrence A., et al., 2007, MNRAS, 379, 1599

Lupton, R., Gunn, J. E., Ivezic, Z., Knapp, G. R., Kent, S., Yasuda N. 2001, ASP Conference Series, eds. Harnden, Jr., F. R., Primini, F. A., Payne, H. E., vol. 238, pg. 269

Martin D. C., et al., 2005, ApJ, 619, L1

Mateo M. & Schechter P. L. 1989, European Southern Observatory Conference and Workshop Proceedings, eds. Grosbøl, P. J., Murtagh, F., Warmels, R. H., vol. 31, pg. 69

Myers, A. D., Brunner, R. J., Richards, G. T., Nichol, R. C., Schneider, D. P., Vanden Berk, D. E., Scranton, R., Gray, A. G., & Brinkmann, Jon, 2006, ApJ, 638, 622

Myers, A. D., Brunner, R. J., Nichol, R. C., Richards, G. T., Schneider, D. P. & Bahcall, N. A., 2007a, ApJ, 658, 85

Myers, A. D., Brunner, R. J., Nichol, R. C., Richards, G. T., Schneider, D. P. & Bahcall, N. A., 2007b, ApJ, 658, 99.

Mohr, J. J. et al. 2012, The Dark Energy Survey Data Processing and Calibration System

Penny, A. J. 1995, IAU Symposium, eds Philip A. G. D., Janes K. A. and Upgren A. R., vol. 167, pg. 173

Phillips, M. M. 1993, AJ, 413, 105

Polak, E., Ribiere, G., 1969, Note sur la convergence de methodes des directions conjugees. Revue Fr. Inf. Rech. Oper. 16-R1, 35-43. MR 0255025.

Rau, A. and Kulkami, S. R. and Law, N. M. 2009, PASP, 131, 1334

Richards, G. T., et al., 2001a, AJ, 121, 2308

Richards, G. T., et al., 2001b, AJ, 122, 1151

Richards, G. T., et al., 2002, AJ, 123, 2945

Richards, J. W., Starr, D. L., Butler, N. R. 2011, AJ, 733, 10

Rossetto, B. M., Santiago, B. X., Girardi, L., et al. , AJ, vol. 141, pg. 185

Rubinstein, R.Y., Kroese, D.P., 2004, The Cross-Entropy Method: A Unified Approach to Combinatorial Optimization, Monte-Carlo Simulation and Machine Learning. Springer-Verlag, New York NY.

Sandage, A. 1958, AJ, 128, 150

Sandage, A., Tammann, G.A., Reindl, B. 2004, A&A, 424, 43

Sandage, A., Tammann, G.A., Reindl, B. 2008, A&A, 493, 471

Schechter, P. 1976, ApJ, 203, 297

Scranton, R., et al., 2005, ApJ, 633, 589

Sérsic, J. L. 1963, Boletin de la Asociacion Argentina de Astronomia La Plata Argentina, 6, 41

Shanno, D- F. 970, "Conditioning of quasi-Newton methods for function minimization", Math. Comput. 24 (111): 647-656

Stetson, P. B. 1987, PASP, 99, 191

Stetson, P. B. 1994, PASP, 106, 250

Sylvain, A., & Celisse, A., 2010, A survey of cross-validation procedures for model selection. Statistics Surveys, 4, 40-79. doi: 10.1214/09-SS054.

Tammann, G.A., Sandage, A., Reindl, B. 2003, A&A, 404, 423

Tyson, J. A. 2003, Proc.SPIE Int.Soc.Opt.Eng., 4836, 10

Udalski, A., Szymanski, M., Kaluzny, J., Kubiak, M., Mateo, M. 1992, Acta Astronomica, 42, 253

Vetterling, T., Flannery, B. P., 1992, Conjugate Gradients Methods in Multidimensions. Numerical Recipes in C - The Art of Scientific Computing, W. H. Press and S. A. Teukolsky (eds.), Cambridge University Press; 2nd edition.

Willemsen, P.G., Eyer, L. 2007, arXiv:0712.2898

Wright E. L., et al., 2010, AJ, 140, 1868

Wolf, C., et al., 2004, A&A, 421,

York, D. G. 2000, AJ, 120, 1579

Acknowledgments

A conclusione di questo lavoro di tesi non posso far altro che ringraziare chi, in un modo o in un altro, mi ha sostenuto in questo percorso.
Innanzitutto desidero ringraziare il Prof. Giuseppe Longo in particolare per avermi seguito durante questi anni di carriera universitaria e per aver avuto sempre fiducia nelle mie capacità. A lui va anche il merito di avermi inserito in un gruppo di persone con le quali lavorare è stato un vero piacere.
Grazie a Massimo Brescia che con la sua presenza e i suoi consigli è stato per me un punto di riferimento. Grazie a Stefano che si è dimostrato sempre disponibile nei miei confronti, anche quando aveva tutti i motivi per mandarmi a quel paese. Un ringraziamento particolare inoltre va ad Amata che, oltre ad essere una bravissima insegnante, mi ha sempre messo a mio agio e mi ha fatto sentire in qualche modo "protetta". Spero di avere ancora il piacere di lavorare con tutti voi molte altre volte in futuro.
Un particolare ringraziamento va naturalmente alla mia famiglia. A mio fratello, con cui ho la fortuna di poter condividere questo giorno speciale, che mi ha sempre supportata e sopportata e che ha l'abilità, preziosa soprattutto in questo periodo, di farmi ridere anche con poco. A mio padre, sempre pronto ad ascoltare, che ci ha sempre sostenuto, senza farci mai mancare affetto e comprensione. A mia mamma, perché, anche se spesso ci scherziamo su, mi ha trasmesso l'amore per questa disciplina e senza la quale probabilmente ora non avrei intrapreso questo percorso. Vi voglio bene.
Anche se non ci sono tra noi legami di sangue, devo ringraziare due persone che ormai considero parte della mia famiglia. Grazie a Guido che sin dai tempi del liceo si prende cura di me come farebbe un fratello maggiore. Ance sentendoci in un modo o nell'altro tutti i giorni, sento sempre la tua mancanza.
Grazie a Maddalena, che è per me a tutti gli effetti una sorella. In questi anni siamo sempre state il sostegno l'una del' altra nelle situazioni più disparate.

Mi piace pensare che sia per questo motivo che il destino ci ha unite quando ci siamo ritrovate in pullman a parlare di Madagascar. Benché il futuro ormai bussi alle porte, ricordati che sei Hoana, con tutto ciò che questo comporta.
Grazie a tutti gli amici dell'Università con cui ho condiviso tanti momenti di studio matto e disperatissimo ma anche tante, tante risate. In particolare ringrazio Alessia, che mi ha accompagnato durante quasi tutto il percorso della specialistica e che ha affrontato un'epopea per festeggiarne insieme a me la conclusione.
Grazie anche alle amiche del liceo perché, nonostante gli anni passino, l'affetto rimane e continua ad essere un piacere vederci (per colpa mia raramente) per fare due chiacchiere.
Grazie a tutti coloro che hanno contribuito a rendere gli anni universitari un'esperienza indimenticabile. Anche se non sono brava con le parole spero che si sia capito quanto per me siate importanti.