Pontificia Universidad Católica de Chile
Facultad de Matemáticas
Departamento de Estadística

Classification and Modeling of time series of astronomical data

por

Felipe Elorrieta López

Tesis presentada al Departamento de Estadística de la Pontificia Universidad Católica de Chile para optar al grado de Doctor en Estadística

Octubre, 2018

Director de Tesis: **Dr. Susana Eyheramendy Duerr**

FACULTAD DE MATEMÁTICAS
PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE

INFORME DE APROBACIÓN
TESIS DE DOCTORADO

Se informa a la Facultad de Matematícas que la Tesis de Doctorado presentada por el candidato

Felipe Elorrieta López

ha sido aprobada por la Comisión de Evaluación de la Tesis como requisito para optar al grado de Doctor en Estadística, en el examen de Defensa de Tesis rendido el día 12 de Octubre de 2018.

Director de Tesis

Dr. Susana Eyheramendy Duerr _____

Comisión de Evaluación de la Tesis

Dr. Wilfredo Palma Manriquez _____

Dr. Giovanni Motta _____

Dr. Pablo Estevez _____

Dr. Cristian Meza _____

FACULTAD DE MATEMÁTICAS
PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE

Fecha: 12 de Octubre 2018

Autor	:**Felipe Elorrieta López**
Título	:**Classification and Modelling of time series of astronomical data**
Departamento	:**Estadística**
Grado	:**Doctor**
Convocación	:**Octubre** 2018

Se le concede permiso para hacer circular y copiar, con propósitos no comerciales, el título ante dicho para los requerimientos de individuos y/o instituciones.

_____

Firma del Autor

Acknowledgments

Un día de marzo de 2014, empezó esta aventura de más de cuatro años. En este momento, cuando al fin entrego mi trabajo de tesis de Doctorado, quiero agradecer a todas las personas que fueron de vital importancia para lograr este objetivo.

En primer lugar, quiero agradecer a mi compañera de vida Milka Vescovi, por todo él apoyo y paciencia durante todo este tiempo. Te Amo. A mis hijos Iñigo y Julián por darme la fuerza para no flaquear con los inconvenientes que se presentaron en el camino. A mi madre Jovita López por la formación que me llevo a lograr este objetivo, a mi hermano Pablo Elorrieta por el desinteresado apoyo durante todos mis años de estudio, a mi hermano mayor Gonzalo Elorrieta y a mi padre Julio Elorrieta, que se que habría estado muy orgulloso de este logro y su eterno recuerdo sigue siempre presente en mi.

Además, quiero agradecer a mi profesora Guía Susana Eyheremandy, en primer lugar por la carta de recomendación para la postulación al programa de Doctorado en Estadística y a la Beca Conicyt. Luego, por darme la oportunidad de desarrollar este tema de Tesis y por toda la orientación entregada durante estos cuatro años. Al profesor Wilfredo Palma, por su crucial apoyo para sacar este tema adelante. A los profesores de la comisión evaluadora por los comentarios realizados, los cuales ayudaron a hacer de este un mejor trabajo. Al profesor Reinaldo Arellano, por guiar mi tesis de Magister y recomendarme al programa de Doctorado en Estadística y a la Beca Conicyt. Finalmente, quisiera hacer una mención especial al profesor Francisco Torres, quien me incentivó a tomar la decisión de seguir el camino del Doctorado en Estadística. Este trabajo está dedicado a su memoria.

También, a todos los que fueron parte de mi vida durante estos años. A Francisca y Carolina por brindarme su apoyo y asistir a mi exposición de defensa de grado. A mis amigos de la USACH Eto, Ruy y Diego que pese a las pocas veces que nos reunimos, cada una de ellas fue una inyección de energía para terminar este trabajo.

Finalmente, quiero agradecer el apoyo del Instituto Milenio de Astrofisica (MAS) y a la Beca de Doctorado CONICYT No. 21140566.

Contents

List of Figures

List of Tables

Abstract

We are living in the era of Big Data, where several tools have been developed to deal with large amount of data. These technological advances have allowed the rise of the astronomical surveys. These surveys are capable to take observations from the sky and from them generate information ready to be analyzed. Among the observations available there are light curves of astronomical objects, such as, variable stars, transients or supernovae. Generally, the light curves are irregularly measured in time, since it is not always possible to get observational data from optical telescopes. This issue makes the light curves analysis an interesting statistical challenge, because there are few statistical tools to analyze irregular time series. In addition, due to the large amount of light curves available in each survey, automated processes are also required to analyze all the information efficiently. Consequently, in this thesis two goals are addressed: the classification of the light curves from the implementation of data mining algorithms and the temporal modeling of them.

Regarding the classification of light curves, our contribution was to develop a classifier for RR Lyrae variable stars in the Vista Variables in the Via Lactea (VVV) near-infrared survey. It is important to detect RR-Lyraes since they are essential to build a three-dimensional map of the Galactic bulge. In this work, the focus is on RR*ab* type ab (i.e., fundamental-mode pulsators). The final classifier is built following eight key steps that include the choice of features, training set, selection of aperture, and family of classifiers. The best classification performance was obtained by the AdaBoost classifier which achieves an harmonic mean between false positives and false negatives of $\approx 7\%$. The performance is estimated using cross validation and through the comparison with two independent datasets that were classified by human experts. The classifier implemented has already made it possible to identify some RR*ab* in the outer bulge and the southern galactic disk areas of the VVV.

In addition, I worked on modeling light curves. I develop new models to fit irregularly spaced time series. Currently there are few tools to model this type of time series. One example is the Continuous Autoregressive model of order one, CAR(1), however some assumptions must be satisfied in order to use this model. A new alternative to fit irregular time series, that we call the irregular autoregressive model (IAR model), is proposed. The IAR model is a discrete representation of the CAR(1) model which provide more flexibility, since it is not limited by Gaussian time series. However, both the CAR(1) and IAR model are only able to estimate positive autocorrelations. In order to fit negatively correlated irregular time series a Complex irregular autoregressive model (CIAR model) was also developed. For both models maximum likelihood estimation procedures are proposed. Furthermore, the finite sample performance of the parameters estimation is assessed by Monte Carlo simulations. Finally, for both models some applications are proposed on astronomical data. Applications include the detection of multiperiodic variable

stars and the verification of the correct estimation of the parameters in models commonly used to fit astronomical light curves.

Keywords: light curves, variable stars, RR-Lyrae, irregular time series, autoregressive models, data mining algorithms.

Chapter 1

Introduction

In the era of Big Data, Astronomy is undergoing a major revolution. Due to the advances in technology the astronomical surveys have evolved from taking observations of small and focused areas of the sky (for example OGLE [68] and HIPPARCOS [28]) to wide-field surveys (for example VVV [48]).

The Large Synoptic Survey Telescope (LSST) is one of the upcoming big challenge in astronomy. It will take a full picture of the whole sky every three nights. This survey is designed to conduct a ten-year survey of the dynamic universe, from 2022 - 2032. This project will generate a huge amount of data posing important challenges that require the expertise from diverse disciplines such as for example Statistics, Informatics and Astronomy. All this data will be available to the community of astronomers living in Chile.

Therefore, in order to face the challenges of the LSST, the Millennium Institute of Astrophysics (MAS) was created. The MAS has gathered over a hundred researchers and students from five prestigious Chilean Universities.

The MAS is divided into five research lines, where one of them is Astrostatistics & Astroinformatics. The role of astro-statisticians is to provide models and tools to process large datasets and extract valuable knowledge from them. For example, data mining and machine learning algorithms will allow us to automate processes necessary to analyze all fields observed by a specific astronomical survey in a short time.

Some astronomical data available for statistical analysis are light curves, which represent the temporal variations of the brightness of an astronomical object. The light curves can represent the brightness of variable stars, the transit of an extrasolar planet or a supernovae. Light curves analysis offers many astronomical and statistical challenges.

For astronomers, light curves analysis allow to study the dynamic properties of an

object. For example, light curves can differ in the degree of change in magnitude, in the degree of regularity from one cycle to the next and in the length of the cycle (period). These properties can be related to physical properties of the system, like rotation and binary period. These dynamic properties allow the astronomers, for example, to identify the class of a specific variable star only by inspecting the temporal behavior of a star.

For statisticians, light curves consist on a time series of the brightness variation of stars. Generally, this time series are irregularly measured in time, since it is not always possible to get observational data from optical telescopes, because its dependency, for example, on clear skies. Working with irregular time series is an important statistical challenge because there are still few robust statistical tools to analyze unequally spaced time series. Some examples are in the estimation of the spectrum of an irregular time series using the Lomb-Scargle periodogram (Zechmeister & Kürster 2009) [74]) or the continuous-time autoregressive moving average (CARMA) models to fit irregular time series (Kelly et al. (2014) [43]).

In this thesis my focus is to provide new methods to analyze light curves of variable stars. This is performed following two approaches, the classification of light curves from the implementation of data mining algorithms and the temporal modeling of them. The challenges that will be addressed in both approaches are shown in Figure 1.1.

Regarding the classification of variable stars, the main aim is to build automated procedures to classify pulsating variable stars from the VVV survey, such as RR Lyraes and Cepheids. The basic idea is that the classifiers that are implemented must be useful for the astronomers in the process of searching pulsating stars within the VVV observation area. Finding pulsating variable stars in the VVV is particularly interesting, since they are essential to determine the three-dimensional structure of our Galaxy.

In order to build the classifier, I followed the procedure proposed by Debosscher et al. (2007) [23], Dubath et al. (2011) [26], Richards et al. (2011) [56]. The basic idea is to compute characteristics or features of the variable stars from an harmonic model fitted to them. Later, the set of features computed for the variable stars in the training set are used as input in the supervised data mining algorithms.

Some additional aims in the classifier construction were addressed. First, to provide new features specifically designed to better characterize the temporal behavior of the pulsating classes. Second, the data mining algorithm used generally for classification problems in Astronomy is the Random Forest. I looked for an alternative from a wide variety of state-of-the-art data mining algorithms.

Another purpose of this thesis, is to provide new methods to the modeling of irregular time series. Currently, the light curves are fitted using the CARMA family of models. Particularly, using the CAR(1) model is possible to estimate the autocorrelation of a irregularly sampled time series. However, the CAR(1) model have some assumptions, e.g., Gaussian distribution and continuous white noise. In this work, a more flexible alternative to fit irregularly sampled time series is introduced. This model is called the irregular autoregressive model (IAR model), which is a discrete representation of the CAR(1) model. This model is more flexible than the CAR(1) model, since it allows non-Gaussian distributed data.

Furthermore, both CAR(1) and IAR models are only able to estimate positive autocorrelation. That is a limitation compared to the regular autoregressive model which can detect both positive and negative time dependencies. In order to address this constraint, a second model, called the complex irregular autoregressive model (CIAR), is proposed. This model is an extension of the irregular autoregressive model that allows to estimate both positive and negative autocorrelation.

In this work, these models are applied in the analysis of astronomical light curves. The light curves are generally modeled using a parametric model that assumes independent and homoscedastic errors. However, these assumptions are not necessarily satisfied, since in many cases there remain a temporal dependency structure on the errors. Here the aim of the irregular time series models is to verify whether the parametric model is capable to describe all the temporal structure of the light curves.

Consequently, in this thesis we present automated and efficient computational methods under a solid statistical framework applied to solve common problems in the analysis of astronomical data. The structure of this thesis is as follows. In the following section of this chapter the purpose of the study is given. In Chapter 2 the literature on both astronomy and statistical background is reviewed, putting the main emphasis in describing the machine learning algorithms and the methods used currently to model irregular time series.

In Chapter 3 I describe the procedure to build a machine learning classifier for the light curves of variable stars. This procedure is presented in eight key steps. The first half corresponds with the data cleaning and the feature extraction steps. The second half of the procedure corresponds to the implementation of two different classifiers, one for the RR*ab* and other for the Cepheids. In both cases, the optimization of machine learning algorithms, the selection of the most important features and assessing the performance of the trained classifier are presented, putting some emphasis in the challenges to build each classifier in the VVV.

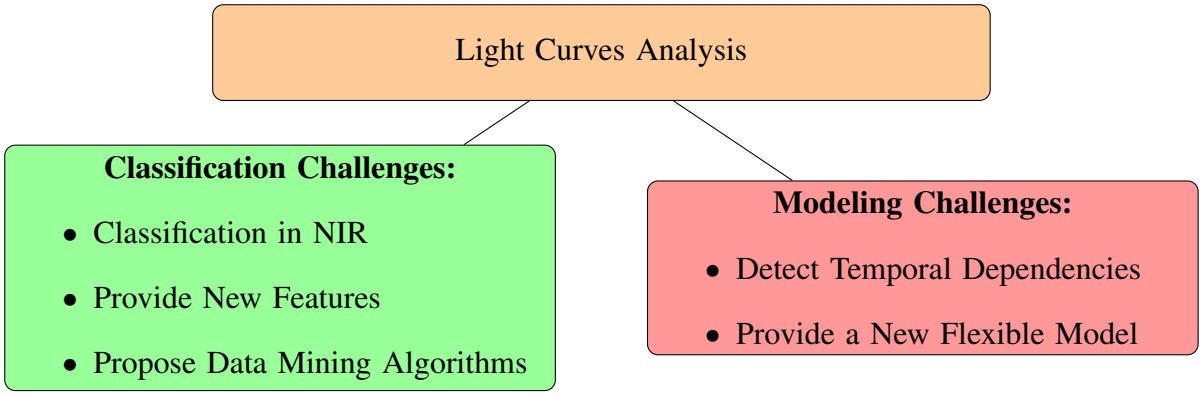


Figure 1.1: Statistical challenges in the light curves analysis addressed in this work

In Chapter 4 we present new and flexible methods to model the irregular time series, which are the irregular autoregressive model (IAR) and the Complex irregular autoregressive model (CIAR). In both cases the estimation procedure is described. In addition, the performance of the maximum likelihood estimators is assessed via Monte Carlo simulations. Some applications in astronomical data of both models are also presented. Finally, in Chapter 5, the conclusions and future works are drawn.

1.1 Purpose of the study

1.1.1 General objective

Provide statistical methods with a robust framework to analyze efficiently a large number of light curves for astronomical data observed using surveys such as OGLE, HIPPARCOS and the VVV.

1.1.2 Specific objectives

1. To build an automated procedure to classify RR Lyrae type ab stars from the VVV survey.

2. To assess the performance of the classifier on different datasets in which flux measurements do not necessarily follow the same conditions in cadence, depth, etc. as our training set.

3. To propose an irregular autoregressive time series model to detect time dependencies on the light curves of astronomical data under a solid statistical framework.

4. To extend the irregular autoregressive model to allow the data to come from other statistical distributions and to detect negative time dependencies.

Chapter 2

Literature Review

2.1 Astronomical Background

Since mankind look at the sky for the first time we have tried to find an explanation to the mysteries of the universe. Over the course of thousand years of our history several scientists, e.g., Aristotle, Nicolaus Copernicus, Johannes Kepler, Galileo, Isaac Newton and Albert Einstein have made significant contributions to explain the astronomical phenomena.

These contributions start in the third century when Aristotle believed that the Earth was at the center of the universe. Later Copernicus proposed that the Sun, not the Earth, was the center of the Solar System. In the early 1600s, Kepler proposed three laws that describe the motion of planets around the Sun. Galileo was the first to use systematically a telescope to observe celestial objects, this allows him to discover the phases of Venus. Sir Isaac Newton improves the Kepler laws of motion and developed the theory of universal gravity. Finally, Einstein developed the theory of general relativity in 1915, which describes how mass and space are related to each other. This theory has been fundamental to better understand astronomical phenomena such as the black holes.

All these contributions have allowed us to better understand how the universe works. Nowadays, we live in the era of the astronomical surveys. Due to the availability of powerful computers, these surveys are capable to take observations from the sky and from them generate a great amount of information ready to be analyzed by the astronomers.

Several scientists from different areas have been attracted by this large amount of information available. Consequently, in this era networks of interdisciplinary collaboration have been created in order to analyze all the available data. These networks are generally composed by astronomers, statisticians, informatics and other related scientists.

All this available data allows us to study and detect patterns of several astronomical objects that are of interest like the stars, planets, supernovas and other variables phenomena. The statistical challenge here, is to provide methods that allow us to analyze efficiently the available information.

Particularly, in this work are addressed two main issues. First, the implementation of an automated procedure to classify variable stars in the VVV survey. The most important challenge related to the classifier is to propose a data mining procedure that considers steps of data cleaning, data transformation and assessment of the implemented classification models. In addition, it is important to test alternative classification methods to the well-known Random Forest algorithm, commonly used to address this problem.

Secondly, the modeling of astronomical time series is another important topic to address. In astronomy it is common to find irregular time series because some conditions must be met to take observations of the sky. Nowadays, there are few methods for modeling irregularly sampled time series. In this work has been made a contribution in this sense, providing new models to fit irregularly sampled time series. The details of each method will be discussed in the following chapters of this thesis. However, some important concepts must be explained previously.

2.1.1 Light curves

Astronomical observations are taken from a region of the sky. From each region observations are obtained several times, which produces a sequence of images in time.

Photometry is the technique of astronomy that allows precise measurement of the brightness of an astronomical object from an image. Historically, several methods have been used to perform the photometry. The last revolution in this sense, came with the rise of the CCD technology.

Using photometry on a sequence of images taken for an astronomical object, a temporal sequence of the brightness measurements can be obtained. This time sequence of brightness measurements is called the light curve of the astronomical object. Consequently, we can define the light curve as a time series of its brightness variations. The light curve allows to follow the behavior of a specific astronomical object through the time.

Additionally, we can define the apparent magnitude as the brightness of an object as it appears to you. Changes in magnitude are in logarithmic scale, i.e., each magnitude means factor of 2.512 in brightness, according to a brightness ranking, originally devel-

oped by Hipparchos (140 AD).

Generally, the light curves are plotted using the apparent magnitude in axis y and the Julian date in axis x. If the astronomical object is periodic, and the period is known, it becomes useful to plot a phased light curve. The phase ϕ of an observation can be computed as,

$$\phi = \left(\frac{t - t_0}{p}\right) - \mathbb{E}(t) \tag{2.1.1}$$

where t_0 is the reference time, t is the time when the observation was taken, p is the period of the light curve and $\mathbb{E}(t)$ is the integer part of $\frac{t-t_0}{p}$, sometimes called as the epoch. The phase generally is expressed as the fraction of the star cycle, taking values in the interval [0,1].

2.1.2 Variable Stars

A variable star is a star whose brightness magnitude fluctuates. Historically, variable stars have been the main tool for determining the content and structure of stellar systems and have had a crucial role in the history of Astronomy. Among these stars, we can differentiate two types depending on whether the process creating the observed variability is inherent to the star (intrinsic variation) or not (extrinsic variation.) The General Catalog of Variable Stars (Samus et al. 2009 [58]) lists over 110 classes and subclasses based on a variety of criteria.

In Figure 2.1 is shown a 'Variability tree' (from Eyer & Mowlavi, 2008 [29]), which gives a visual summary of several of the different types of variable phenomena that may be found in astronomy, in this diagram four division levels are introduced. In the first level are the "classical" division between extrinsic and intrinsic variables. In the second level a distinction is made according to the type of object, being either asteroids, stars, or galaxies.

The third level identifies the phenomenon at the origin of the variability. In the group with extrinsic variability, the phenomena considered are the rotation, microlensing effects and eclipses by a companion or by a foreground object. Among the former, are the eclipsing binaries. The eclipsing binaries are a system in which two stars orbiting a common center of mass. This type of variable stars is composed by the classes Ellipsoidal (ELL), Beta Persei (EA), Beta Lyrae (EB) and the W Ursae Maj (EW).

Among the intrinsic variable objects one finds the eruptive variables, the cataclysmic variables, the rotational variables and finally the pulsating variables. Arguably the most

important group among the intrinsic variables is that of pulsating variables because it contains the RR Lyrae and Cepheids classes, which satisfy a relation between their periods and absolute luminosities that allow to estimate distances, a quantity both fundamental and elusive in Astronomy. For a current review of the physics and phenomenology of pulsating stars, we refer to the recent monograph by Catelan & Smith (2015) [16].

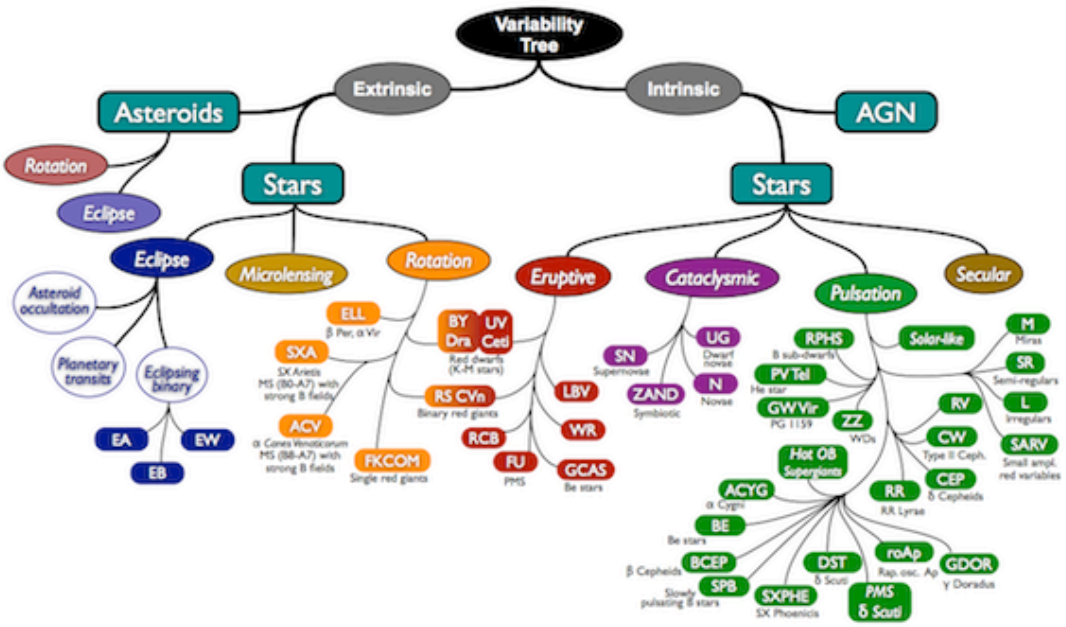


Figure 2.1: 'Variability tree' showing the many different types of stellar (and non-stellar) phenomena that are found in astronomy, (from Eyer & Mowlavi, 2008 [29]).

Many classes of variable stars can also be divided into subclasses. For example, the RR-Lyraes are divided between RR Lyrae types ab and c (sometimes alternatively termed RR0 and RR1, respectively) if the stars pulsate in the fundamental radial mode or in the first-overtone radial mode respectively. The main difference between these subclasses of RR-Lyrae is that the RR*ab* have longer pulsation periods and asymmetric light curves, while the RR*c* have shorter pulsation periods and almost sinusoidal light curves.

The Cepheids can also be divided in the Classical Cepheids (Type I) and the Type II. The different Cepheids types obey different period-luminosity relations. In general, Type I Cepheids are brighter than Type II Cepheids, if both have the same period.

In addition, it is well known that the Cepheids and RR-Lyraes have multi-periodic subclasses such as DMCEP (Double-Mode Cepheid) and RRD (Double-Mode RR-Lyrae) respectively (Moskalik, 2014 [49]). Table 2.1 shows a briefly description of the most common variable stars with its respective observational properties,

Table 2.1: *Most common pulsating variable stars and their amplitudes (in magnitude) and periods (in days) (from Eyer & Mowlavi, 2008 [29]).*

Class name	Period (Days)	Amplitude (mag)
Cepheids	2-70	0.1-1.5
RR Lyrae	0.2-1.1	0.2-2
SR-MIRA	50-1000	up to 8
SPB	0.5-5	up to 0.03
RVTau	30-150	1-3
δ-Scuti	0.02-0.25	up to few 0.1

As mentioned in the previous chapter, in this thesis most of the analysis will be done on the light curves of variable stars, so it is very important to have cleared these concepts to understand the subsequent results. The data of the light curves of several stars (variables and non-variables) can be extracted from different astronomical surveys. In this work the information sources that will be used are the OGLE, HIPPARCOS and the VVV survey whose characteristics will be described below.

2.1.3 Astronomical Surveys

2.1.3.1 OGLE and HIPPARCOS Survey

The Optical Gravitational Lensing Experiment (OGLE) is a ground-based survey from Las Campanas Observatory covering fields in the Magellanic Clouds and Galactic bulge. The OGLE survey began regular sky monitoring on April 12, 1992 as one of the first-generation microlensing sky surveys. The project is now in its fourth phase.

The first phase of the project (OGLE-1) started in 1992 (Udalski et al, 1992 [67]) and observations were continued for four consecutive observing seasons through 1995. The OGLE-II survey collected data from January 1997 to December 2000 (Udalski et al, 1997 [68]). On June 12, 2001 regular observations of the OGLE-III phase began (Udalski 2003b) and ended in May 2009. Finally, the OGLE-IV survey began regular observations of the sky on the night of March 4/5, 2010 (Udalski et al, 2015 [69])). Since 1997 observations have been conducted with the modern automated 1.3 m Warsaw telescope at the Las Campanas Observatory, Chile.

Throughout the 25-year history of OGLE it has been discovered hundreds of thousands of pulsating stars. Most of the observations are collected in the I-band filter with a number of collected epochs between 120-150 in OGLE-IV (Udalski, 2017 [70]).

Hipparcos (The High Precision Parallax Collecting Satellite) Space Astrometry Mission (Perryman et al. 1997 [28]) was an ESA project designed to precisely measure the positions of more than one hundred thousand stars. Launched in August 1989, Hipparcos successfully observed the celestial sphere for 3.5 years before operations ceased in March 1993. Among the 118218 stars measured by Hipparcos, 11597 were found to be (possibly) variable. Of these more than 8000 were new.

2.1.3.2 VVV ESO Public Survey

The Vista Variables in the Via Lactea (VVV) is an ESO public survey that is performing a variability survey of the Galactic bulge and part of the inner disk using ESO's Visible and Infrared Survey Telescope for Astronomy (VISTA), a 4m-class telescope operated by ESO and located at Cerro Paranal, Chile. The VISTA Telescope has started the observations in February 2010 and has finished in October 2015, in this time it took 1929 hours of observation. The sky area covered by the survey was of 520 $\deg^2$ (Fig.1 2.2), where there are 10^9 point sources, an estimated $\sim 10^6 - 10^7$ variable stars, 33 known globular clusters and approximately 350 open clusters.

Unlike optical surveys such as OGLE and HIPPARCOS, the VVV is characterized by using near-infrared filters (Z, Y, J, H and Ks) (NIR). The size of an uniformly covered field (also called a "tile") is 1,501 $\deg^2$, hence the VVV Survey requires a total of 348 such "tiles" to cover the survey area (see (Fig 2.3)), a total of 196 tiles are needed to map the bulge area and 152 tiles for the disk.

Aperture photometry of VVV sources is performed on single detector frame stacks provided by the VISTA Data Flow System (Irwin et al. 2004 [39]) of the Cambridge Astronomy Survey Unit (CASU). A series of flux-corrected circular apertures are used as detailed in previous publications (Catelan et al. 2013 [17]; Dekany et al. 2015 [24]). The smallest 5 apertures, which we denoted as 1, 2, 3, 4, 5, are extracted in aperture radii of $\{0.5, 1/\sqrt{2}, 1, \sqrt{2}, 2\}$ arcsec.

The final products will be a deep NIR atlas in five passbands. One of the main goals is to gain insight into the inner Milky Way origin, structure, and evolution. This will be achieved, for instance, by obtaining a precise three-dimensional map of the Galactic bulge. To achieve this goal, the pulsating stars like Cepheids or RR-Lyrae are of particular importance, for example, there are many RR Lyrae in the direction of the bulge and,

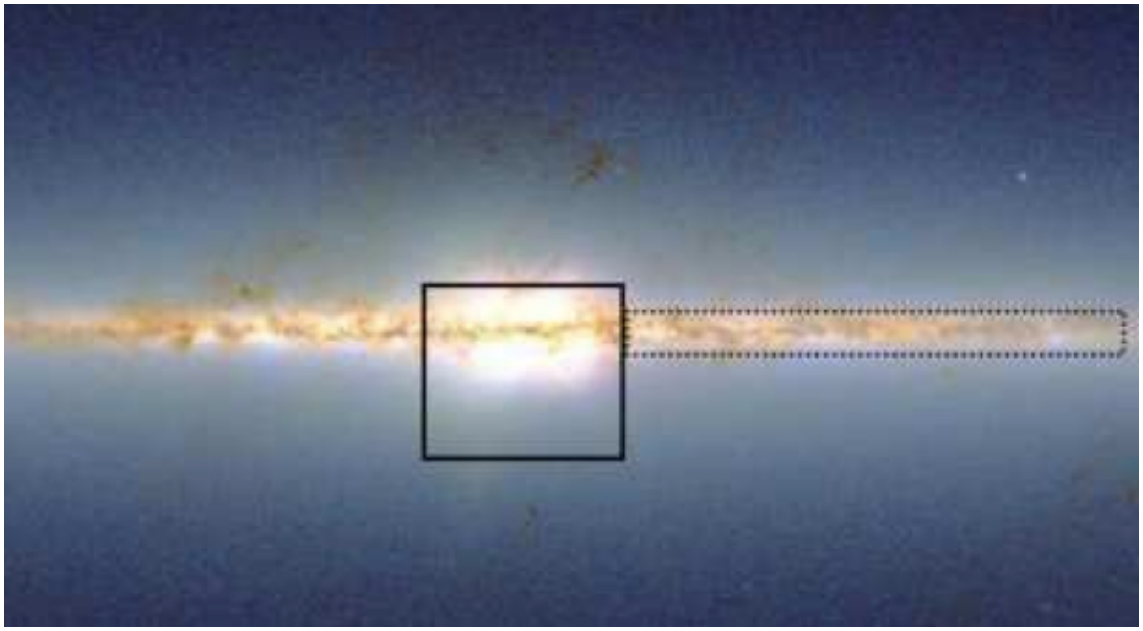


Figure 2.2: 2MASS map of the inner Milky Way showing the VVV bulge (solid box, $-10^o < l < +10^o$ and $-10^o < b < +5^o$) and plane survey areas (dotted box, $-65^o < l < 10^o$ and $-2^o < b < +2^o$), (from Minniti, 2010 [48]).

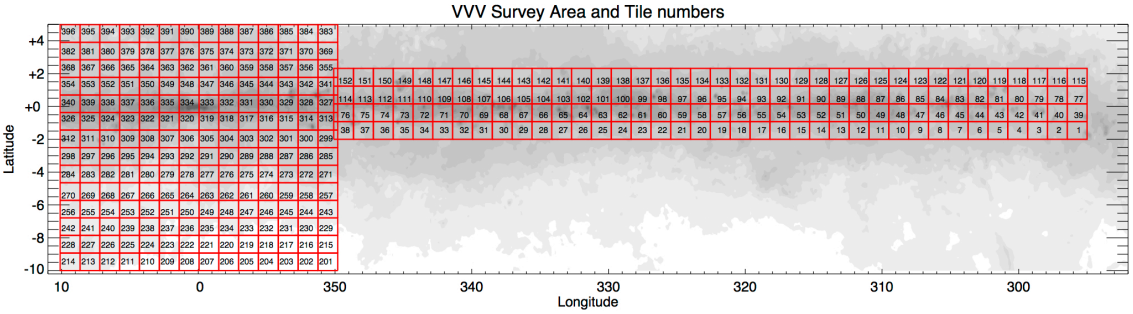


Figure 2.3: VVV Survey Area Tile Numbers Galactic Coordinates (from Catelan et al. 2013 [17]).

because they are very old, they are fossil records of the formation history of the Milky Way. For a detailed account of the VVV see Minniti et al. (2010) [48], and for a recent status updated with emphasis on variability see Catelan et al. (2014) [18].

With the information extracted from the VVV, OGLE and HIPPARCOS surveys, catalogues of known variable stars have been created, which will be useful to test the classifi-

cation and modeling methods that will be proposed in this work. The main idea is to have previously certified tools that allows us to be prepared when future synoptic studies such as the Large Scale Synoptic Telescope (LSST, Ivezic et 2008 [40]) become operational.

The Large Synoptic Survey Telescope (LSST) is the upcoming big challenge in astronomy, this survey is designed to conduct a ten-year survey of the dynamic universe from 2022 - 2032. Among its main goals is to define more precisely the structure and formation of our home galaxy, the Milky Way and cataloging the solar system.

2.2 Time Series Background

In the study of astronomical data, the time series tools have been widely used to explain the temporal behavior of the flux of astronomical objects such as, variable stars, transients or supernovae. This is useful since these objects can be characterized from its temporal behavior. For example, from a suitable time series model, can be derived a set of features able to distinguish between different types of variable stars.

The astronomical observations are generally obtained at irregular time gaps due to some conditions must be met to be able to observe in the optical telescopes, for example, that the sky is clear. This implies several statistical challenges because there are few statistical tools specifically developed to work with unequally spaced time series.

A time series can be defined as a real valued sequence of observations Y_n with $n = 1, \ldots, N$ measured in observational times t_n such that the sequence $t_1, \ldots, t_N$ is strictly increasing. A time series is called regular if the distance of consecutive times $t_j - t_{j-1}$ is constant, whereas if this distance is not constant, the time series is called irregular.

Another basic distinction can be made between the time series tools depending on the domain in which they operate. First the time domain methods will be reviewed.

2.2.1 Analysis in Time Domain

First, I will introduce some basic ideas of the time series analysis. Among the most important concepts in time series are the following:

- **Strict stationarity:** A stochastic process Y_t is strictly stationary (or strongly stationary) if each joint distribution F of a finite sequence of length n $(Y_1, Y_2, \ldots, Y_n)$ is invariant to a translation in k times, i.e.,

$$F(Y_{k+1}, Y_{k+2}, \ldots, Y_{k+n}) = F(Y_1, Y_2, \ldots, Y_n) \quad \forall n, k \in \mathbb{Z}$$

- **Weak stationarity:** A stochastic process Y_t is weakly stationary (or second-order stationarity) if,

  1. $\mathbb{E}[Y_t] = \mu < \infty \quad \forall t \in \mathcal{T}$
  2. $\mathbb{V}[Y_t] = \sigma^2 < \infty \quad \forall t \in \mathcal{T}$
  3. There exists a function $\gamma(.)$ such that $Cov(Y_t, Y_s) = \gamma(t - s) \quad \forall t, s \in \mathcal{T}$

It is easy to check that if Y_t is strictly stationary and $\mathbb{E}(Y_t^2) < \infty \ \forall t$ then Y_t is also weakly stationary [11].

- **Autocovariance function** Let Y_t be a weakly stationary process. The autocovariance function (ACVF) of Y_t at lag k is,

$$\gamma(k) = Cov(Y_t, Y_{t-k})$$

- **Autocorrelation function** The autocorrelation function (ACF) of Y_t at lag k is,

$$\rho(k) = \frac{\gamma(k)}{\gamma(0)}$$

- **White Noise** A weakly stationary sequence ϵ_t is called white noise if all observations of this sequence are uncorrelated. If the mean of ϵ_t are 0 then the sequence can be denoted $\epsilon_t \sim WN(0, \sigma^2)$.

Time series models can be implemented under two scenarios. When the observations are measured regularly or irregularly in time. The basic time series processes are defined in the regular case. The most common model used to fit a weakly stationary time series is the ARMA(p,q) model.

2.2.1.1 ARMA Models

Y_t is an ARMA(p,q) process, if Y_t is weakly stationary and can be written as,

$$Y_t - \phi_1 Y_{t-1} - \ldots - \phi_p Y_{t-p} = \epsilon_t + \theta_1 \epsilon_{t-1} + \ldots + \theta_q \epsilon_{t-q} \tag{2.2.1}$$

where $\epsilon_t \sim WN(0, \sigma^2)$. In addition, let $\Phi(B) = (1 - \phi_1 B - \ldots - \phi_p B^p)$ and $\Theta(B) = (1 + \theta_1 B + \ldots + \theta_q B^q)$ the autoregressive polynomial and the moving-average polynomial respectively. The process is well defined if $\Phi(B)$ and $\Theta(B)$ have no common factors.

A condition for which a stationary solution of 2.2.1 exists is that the zeros of the autoregressive polynomial $\Phi(B) = (1 - \phi_1 B - \ldots - \phi_p B^p)$ are located outside of the unit circle. Some particular cases of the ARMA models can be defined, for example,

- **Autoregressive process (AR)**. If Y_t is an ARMA(p,q) process with $q = 0$, then $Y_t \sim AR(p)$ and can be written as,

$$Y_t - \phi_1 Y_{t-1} - \ldots - \phi_p Y_{t-p} = \epsilon_t \tag{2.2.2}$$

- **Moving-Average process (MA)**. If Y_t is an ARMA(p,q) process with $p = 0$, then $Y_t \sim MA(q)$ and can be written as,

$$Y_t = \epsilon_t + \theta_1 \epsilon_{t-1} + \ldots + \theta_q \epsilon_{t-q} \tag{2.2.3}$$

The ARMA(p,q) process can be estimated by maximum likelihood. Let the observations equally spaced on time $\mathbf{Y} = (Y_1, \ldots, Y_n)'$ have a Gaussian distribution, with the follow covariance matrix,

$$\Gamma_\lambda = (\gamma(i-j))_{i,j=1}^n = (Cov(Y_i, Y_j))_{i,j=1}^n$$

with $\lambda = (\phi_1, \ldots, \phi_p, \theta_1, \ldots, \theta_q, \sigma^2)'$ is the parameter vector of the model. The likelihood of $\mathbf{Y}$ is,

$$L(\lambda) = (2\pi)^{-n/2} |\Gamma_\lambda|^{-1/2} \exp\left\{ -\frac{1}{2} Y' \Gamma_\lambda^{-1} Y \right\}$$

.

From Section 5.2 of Brockwell and Davis [11] the maximum likelihood estimator $\hat{\lambda}$ is asymptotically normal.

2.2.1.2 ARFIMA Models

The ARMA model is a particular case of the general linear process. Another particular class of linear time series is called long memory processes. On the contrary of the ARMA models, the long-memory processes are characterized by an autocovariance function not absolutely summable, i.e. , the autocovariance function $\gamma(k)$ is such that,

$$\sum_{k=-\infty}^{\infty} |\gamma(k)| = \infty$$

A well-known class of long-memory models is the autoregressive fractionally integrated moving-average (ARFIMA) processes. An ARFIMA process Y_t may be defined by,

$$\Phi(B) Y_t = \Theta(B)(1-B)^{-d} \epsilon_t \tag{2.2.4}$$

where $\Phi(B)$ and $\Theta(B)$ are the autoregressive polynomial and the moving-average polynomial respectively. In addition, $(1-B)^{-d}$ is a fractional differencing operator defined by the binomial expansion,

$$(1-B)^{-d} = \sum_{j=0}^{\infty} \eta_j B^j \tag{2.2.5}$$

where

$$\eta_j = \frac{\Gamma(j+d)}{\Gamma(j+1)\Gamma(d)}$$

for $d < \frac{1}{2}$, $d \neq 0, -1, -2, \ldots$ and ϵ_t is a white noise sequence with finite variance. If the process (2.2.4) satisfy that $d \in (-1, \frac{1}{2})$, and the polynomials $\Phi(B)$ and $\Theta(B)$ have no common zeros, then the stationarity, causality and invertibility of an ARFIMA model can be established.

Each time series model reviewed previously may be represented in many different forms. Some examples are the Wold expansion, the autoregressive expansion and the state space systems. Here, I will focus in the state-space system, since it will be very useful later on.

2.2.1.3 State-Space Systems

A linear state space system may be described by the following equations,

$$X_t = F_{t-1}X_{t-1} + V_{t-1} \tag{2.2.6}$$

$$Y_t = GX_t + W_t \tag{2.2.7}$$

where (2.2.6) is known as the state equation which determines a $v-$dimensional state variable X_t and the second equation (2.2.7) is called the observation equation, which expresses the $w-$dimensional observation Y_t. In addition, F_t is a sequence of $v \times v$ called the transition matrix, $G \in \mathbb{R}^{w \times v}$ is the observation linear operator of the observation matrix. Finally, $W_t \sim WN(0, R_t)$, $V_t \sim WN(0, Q_t)$ and V_t is uncorrelated with W_t.

Properties

- **Stability**

  A state space system is said to be stable if F_t^n converges to zero as n tends to ∞. If λ is an eigenvalue of F_t associated to the eigenvector x, then $F_t^n x = \lambda^n x$. Thus, if the eigenvalues of F_t satisfy $|\lambda| < 1$ then $\lambda^n \to 0$ as n increases. Consequently, $F^n x$ also converges to zero as $n \to \infty$.

  In the stable case the equations (2.2.9) have the unique stationary solution given by

$$X_t = \sum_{j=0}^{\infty} F_t^j V_{t-j-1}$$

The corresponding sequence of observations

$$Y_t = W_t + \sum_{j=0}^{\infty} GF_t^j V_{t-j-1}$$

is also stationary.

- **Hankel Matrix**

  Suppose that $\psi_0 = 1$ and $\psi_j = GF^{j-1} \in \mathbb{R}$ for all $j > 0$ such that $\sum_{j=0}^{\infty} \psi_j^2 < \infty$. Then from (2.2.6)- (2.2.7), the process Y_t may be written as the Wold expansion

$$Y_t = \sum_{j=0}^{\infty} \psi_j^2 \epsilon_{t-j}$$

  This linear process can be characterized by the *Hankel matrix* given by

$$H = \begin{pmatrix} \psi_1 & \psi_2 & \psi_3 & \dots \\ \psi_2 & \psi_3 & \psi_4 & \dots \\ \psi_3 & \psi_4 & \psi_5 & \dots \\ \vdots & \vdots & \vdots & \vdots \end{pmatrix}$$

  Since the state space representation of a linear regular process is not necessarily unique, one may ask what the minimal dimension of the state vector is. In order to answer this question it is necessary to introduce the concepts of observability and controllability.

- **Observability**

  Let $\mathcal{O} = (G', F_t'G', F_t'^2G', \dots)'$ be the observability matrix. The system (2.2.6)-(2.2.7) is said to be observable if and only if $\mathcal{O}$ is full rank or, equivalently, $\mathcal{O}'\mathcal{O}$ is invertible.

  The definition of observability is related to the problem of determining the value of the unobserved initial state x_0 from a trajectory of the observed process $\{y_0, y_1, \dots\}$ in the absence of state or observational noise.

- **Controllability**

  Consider the case where the state error is written in terms of the observation so that $V_t = HW_t$ and the state space model can be expressed as,

$$X_{t+1} = F_t X_t + HW_t \tag{2.2.8}$$

$$Y_t = GX_t + W_t \tag{2.2.9}$$

  Let $C = (H, F_t H, F_t^2 H, \ldots)$ be the controllability matrix. The system (2.2.8)- (2.2.9) is controllable if C is full rank or, $C'C$ is invertible.

- **Minimality**

  A state space system is minimal if F_t is of minimal dimension among all representations of the linear process (3.3). A state space system is minimal if and only if it is observable and controllable.

The estimation of the state space models can be performed by the following Kalman recursive equations. For the state-space model (2.2.6)- (2.2.7), the one-step predictors $\hat{X}_t = P_{t-1}(X_t)$ and their error covariance matrices $\Sigma_t = \mathbb{E}[(X_t - \hat{X}_t)(X_t - \hat{X}_t)']$ are unique and determined by the initial conditions

$$\hat{X}_1 = P(X_1|Y_0), \qquad \Sigma_1 = \mathbb{E}[(X_1 - \hat{X}_1)(X_1 - \hat{X}_1)']$$

And the recursions, for $t = 1, \ldots$

$$\Lambda_t = G_t \Omega_t G_t' + R_t \tag{2.2.10}$$

$$\Theta_t = F_t \Omega_t G_t' \tag{2.2.11}$$

$$\Omega_{t+1} = F_t \Omega_t F_t' + Q_t - \Theta_t \Lambda_t^{-1} \Theta_t' \tag{2.2.12}$$

$$\nu_t = Y_t - G_t \hat{X}_t \tag{2.2.13}$$

$$\hat{X}_{t+1} = F_t \hat{X}_t + \Theta_t \Lambda_t^{-1} \nu_t \tag{2.2.14}$$

where $\{\nu_t\}$ is called the innovation sequence.

The optimization of the parameters in Kalman recursion was made by minimizing the reduced likelihood defined as,

$$\ell(\phi) \propto \frac{1}{n}\sum_{t=1}^{n}\left(\log(\Lambda_t) + \frac{\nu_t^2}{\Lambda_t}\right)$$

So far, I have introduced the methods to analyze regular time series. Suppose now a sequence of observational times and values (t_n, Y_n) such that the series $t_1, \ldots, t_N$ is strictly increasing and the distance between consecutive times, $t_j - t_{j-1}$ is not constant $\forall\, j = 1, \ldots, N$, i.e., henceforth it will be assumed that $Y_1, \ldots, Y_N$ was irregularly measured in time. Generally, when a time series is measured in continuous time, the notation changes slightly, writing $Y(t)$ rather than Y_t. An usual approach to fit irregular time series is using the continuous-time autoregressive moving average (CARMA) models

2.2.1.4 CARMA Models

Continuous-time ARMA processes are defined in terms of stochastic differential equations analogous to the difference equations that are used to define discrete-time ARMA processes. The continuous time AR(1) (CAR (1)) process is defined as a stationary solution of the first-order stochastic differential equation.

$$\frac{d}{dt}Y(t) + \alpha_0 Y(t) = \sigma\nu(t) + \beta \tag{2.2.15}$$

where $\nu(t)$ is the continuous time white noise, α_0 and β are unknown parameters of the model. In addition, $\nu(t) = \frac{d}{dt}B(t)$, where $B(t)$ is the standard Brownian motion or Wiener process. The derivative of $B(t)$ does not exist in the usual sense, so equation (2.2.15) is interpreted as an Itô differential equation,

$$dY(t) + \alpha_0 Y(t)dt = \sigma dB(t) + \beta dt, \quad t > 0, \tag{2.2.16}$$

with $dY(t)$ and $dB(t)$ denoting the increments of Y and B in the interval $(t, t+dt)$ and $Y(0)$ a random variable with finite variance, independent of $\{B(t)\}$. The solution of (2.2.16) can be written as,

$$Y(t) = e^{-\alpha_0 t}Y(0) + \sigma\int_0^t e^{-\alpha_0(t-u)}dB(u) + \beta\int_0^t e^{-\alpha_0(t-u)}du$$

or equivalently,

$$Y(t) = e^{-\alpha_0 t}Y(0) + e^{-\alpha_0 t}I(t) + \beta e^{-\alpha_0 t}\int_0^t e^{\alpha_0 u}du \tag{2.2.17}$$

where $I(t) = \sigma\int_0^t e^{\alpha_0 u}dB(u)$ is an Itô integral satisfying $\mathbb{E}(I(t)) = 0$ and $Cov(I(t+h), I(t)) = \sigma^2\int_0^t e^{2\alpha_0 u}du$ for all $t \geq 0$ and $h > 0$. It can be shown that necessary and sufficient conditions for $\{Y(t)\}$ to be stationary are $\alpha_0 > 0$, $\mathbb{E}(Y(0)) = \beta/\alpha_0$ and $\mathbb{V}(Y(0)) = \sigma^2/(2\alpha_0)$. In addition, under these conditions

$$\mathbb{E}(Y(t)) = \beta/\alpha_0 \quad Cov(Y(t+h), Y(t)) = \frac{\sigma^2}{2\alpha_0} e^{-\alpha_0 h}$$

Further, if $Y(0) \sim N\left(\beta/\alpha_0, \sigma^2/(2\alpha_0)\right)$, then the CAR(1) process is also Gaussian and strictly stationary.

If $a > 0$ and $0 \leq s \leq t$, it follows from (2.2.17) that $Y(t)$ can be expressed as,

$$Y(t) = e^{-\alpha_0(t-s)} Y(s) + \frac{\beta}{\alpha_0}\left(1 - e^{-\alpha_0(t-s)}\right) + e^{-\alpha_0 t}(I(t) - I(s)) \tag{2.2.18}$$

or equivalently,

$$Y(t) - \frac{\beta}{\alpha_0} = e^{-\alpha_0(t-s)}\left(Y(s) - \frac{\beta}{\alpha_0}\right) + e^{-\alpha_0 t}(I(t) - I(s)) \tag{2.2.19}$$

We can be extending the model (2.2.15) to a standard continuous time autoregressive model of order (p) (CAR(p)) process $\{Y(t)\}$:

$$D^p Y(t) + \alpha_1 D^{p-1} Y(t) + \ldots + \alpha_p Y(t) = \beta_0 DB(t) \tag{2.2.20}$$

It is useful to represent this equation in operator notation as $\alpha(D)Y(t) = B(t)$ where,

$$\alpha(D) = D^p + \alpha_1 D^{p-1} + \ldots + \alpha_{p-1} D + \alpha_p \tag{2.2.21}$$

A parameterization used by Jones (1981) [41] was based on the zeros $r_1, \ldots, r_p$ of $\alpha(D)$ such that

$$\alpha(D) = \prod_{i=1}^{p} (D - r_i)$$

The necessary and sufficient condition for the stationarity of the model is that all the zeros have negative real parts.

In Belcher et al, (1994) [6] it is defined a new parameterization, where the old parameters α_i may be constructed from the new parameters ϕ_i. The link between the α and ϕ parameters is given by,

$$\beta(D) = \beta_0 D^p + \beta_1 D^{p-1} + \ldots + \beta_{p-1} D + \beta_p = \sum_{i=0}^{p} \phi_i (1 - D/\kappa)^i (1 + D/\kappa)^{p-i} \tag{2.2.22}$$

where κ is a scale parameter and we take $\phi_0 = 1$. Thus the β_i are linear combinations of the ϕ_i. Now let $\alpha(s) = \beta(s)/\beta_0$ so that $\alpha_i = \beta_i/\beta_0$. In a CAR(1) model we can prove that $\alpha_1 = \kappa\frac{(1+\phi_1)}{(1-\phi_1)}$. The function car of the R packages `cts` ([73]) estimates the reparametrized

autoregressive parameters ϕ_i.

Finally, we define a zero-mean CARMA(p,q) process $\{Y(t)\}$ (with $0 \leq q < p$) to be a stationary solution of the pth-order linear differential equation,

$$D^p Y(t) + \alpha_1 D^{p-1} Y(t) + \ldots + \alpha_p Y(t) = \beta_0 DB(t) + \beta_1 D^2 B(t) + \ldots + \beta_q D^{q+1} B(t) \quad (2.2.23)$$

where $D^{(j)}$ denotes j-fold differentiation with respect to t. Since the derivatives $D^j B(t)$, $j > 0$, do not exist in the usual sense, we interpret (2.2.23) as being equivalent to the state-space system.

As mentioned above, the time series methods can be distinguished according to the domain in which works. As the methods which works in time domain were already revised, now the methods that operate in the frequency domain will be seen. Here also the distinction between regular and irregular time series will be made.

2.2.2 Analysis in Frequency Domain

Frequency-domain methods are based in the discrete Fourier transform. The main difference with the time-domain analysis is that these methods are based on the correlation function, while the Frequency-domain methods analyze the response of the process to given set of frequencies. This type of analysis is also called spectral analysis.

2.2.2.1 Spectral Analysis in regular case

Let ω denote the frequency, such that $-\pi < \omega < \pi$, and let P denote the period, such that $P = \frac{2\pi}{\omega}$. Given a time series $\{Y_t\}$, the spectrum is defined to be the Fourier transform of the autocovariance function $\gamma_y(h)$

$$f_y(\omega) = \frac{1}{2\pi} \sum_{h=-\infty}^{\infty} e^{-ih\omega} \gamma_y(h)$$

Now, if we know the spectrum $f_y(\omega)$, from Herglotz Theorem we can compute $\gamma_y(h)$ using the inverse Fourier transform:

$$\gamma_y(h) = \int_{-\pi}^{\pi} e^{ih\omega} f_y(\omega) d\omega$$

These results show that the Fourier spectrum can be directly mapped onto the time-domain autocovariance function, in this sense the frequency and time domain methods

are closely related.

To estimate the spectral density, we may compute the periodogram, which is defined as the squared modulus of the discrete Fourier transform of the autocovariance function, i.e.,

$$I(\omega) = \frac{1}{2\pi} \left| \sum_{h=-\infty}^{\infty} e^{-ih\omega} \gamma_y(h) \right|^2$$

A sampled data set like $\frac{2\pi j}{n}$ with $j = \ldots, -3, -2, -1, 0, 1, 2, 3, \ldots$ contains complete information about all spectral components in a signal Y_t, between the low fundamental frequency $\frac{2\pi}{n}$ and the high Nyquist frequency π.

Fourier analysis has restrictive assumptions, for example, equally spaced time series and sinusoidal shaped variations. As mentioned above, these assumptions are rarely achieved in real astronomical data. There are several methods more appropriate to use in this context, as for example the Lomb Scargle Periodogram.

2.2.2.2 Lomb-Scargle Periodogram

The Lomb-Scargle (LS) periodogram is an extension of the conventional periodogram for unevenly sampled time series. For a time series (t_i, Y_i) with zero mean, the normalized LS periodogram can be computed as,

$$P_{LS}(\omega) = \frac{1}{2\sigma^2} \left(\frac{[\sum_{i=1}^{N} y_i \cos(\omega t_i - \tau)]^2}{\sum_{i=1}^{N} \cos^2(\omega t_i - \tau)} + \frac{[\sum_{i=1}^{N} y_i \sin(\omega t_i - \tau)]^2}{\sum_{i=1}^{N} \sin^2(\omega t_i - \tau)} \right)$$

where $\omega = 2\pi f$ is the angular frequency, f is the ordinary frequency, τ is the phase and σ^2 is the sample variance of y_i. The parameter τ is defined by,

$$\tan(2\omega\tau) = \frac{\sum_{i=0}^{N} \sin(2\omega t_i)}{\sum_{i=0}^{N} \cos(2\omega t_i)}$$

Lomb (1976) [45] showed that P_{LS} is identical to the least-squares fit of a single component stationary sinusoidal model of the form $y(t) = A\sin(\omega t + \tau)$. Consequently, the dominant angular frequency ω is the value that best fit the time series in a least squares sense. This frequency also corresponds to the maximum power in the Lomb-Scargle periodogram. The sinusoidal model can also be expressed by $y(t) = a\cos(\omega t) + b\sin(\omega t)$

where the amplitude A and the phase ψ can be computed from the estimated parameters as $A = \sqrt{a^2 + b^2}$ and $\psi = atan(a/b)$ respectively.

However, the Lomb-Scargle periodogram have two shortcomings. First, this method assumes that the mean of the data is equivalent to the mean of the fitted sine functions. Second, the Lomb-Scargle periodogram does not take into account the measurement errors. The Generalized Lomb-Scargle (GLS) defined by Zechmeister & Kürster (2009) [74] solves these limitations.

2.2.2.3 Generalized Lomb-Scargle Periodogram

This method takes in consideration the measurement error by introducing a weighted sum in the original Lomb-Scargle formulation. Additionally, the GLS introduce an offset constant c to overcome the assumption of the mean of the data. Consequently, let Y_i be the N measurements of a time series at time t_i and measurement errors σ_i, the GLS performs a full sine waves fitting of the form,

$$y(t) = a\cos(\omega t) + b\sin(\omega t) + c$$

where the frequency f is obtained by minimizing the squared difference between the observed data Y_t and the model function $y(t)$ as follow,

$$\chi^2_m(f) = min_\theta \chi^2(f) = \sum_i \frac{(Y_i - y(t_i))^2}{\sigma_i^2}$$

where $\theta = (a, b, c)$ is the parameter vector of the model $y(t_i)$. Let χ^2_0 defined by,

$$\chi^2_0 = \frac{\sum_i [Y_i - \mu]^2}{\sigma_i^2}$$

where $\mu = \frac{\sum_i Y_i/\sigma_i^2}{\sum_i 1/\sigma_i^2}$. The normalized Generalized Lomb Scargle (GLS) periodogram is given by,

$$P_f(f) = \frac{\chi^2_0 - \chi^2_m(f)}{\chi^2_0} \tag{2.2.24}$$

Under Gaussian noise the difference $\chi^2_0 - \chi^2_m(f)$ is χ^2 distributed with 2 degrees of freedom. Alternatively, the periodogram can be normalized by a Gaussian noise level $\frac{2}{N-1}$, so the equation (2.2.25) becomes,

$$P(f) = \frac{N-1}{2}\frac{\chi_0^2 - \chi_m^2(f)}{\chi_0^2} \tag{2.2.25}$$

P is F-distributed with 2 numerator and N-1 denominator degrees of freedom under the null hypothesis of white noise spectrum (Richards et al. (2011) [56]). Note that the power of the periodogram $P_f(f)$ is restricted to $0 \leq P_f(f) < 1$, while the periodogram $P(f)$ is restricted to $0 \leq P(f) < \frac{N-1}{2}$.

An advantage of this generalization with respect to the original Lomb-Scargle periodogram is that the GLS is less susceptible to aliasing, giving more accurate frequencies as a consequence of a better determination of the spectral intensity.

2.3 Machine Learning Background

The machine learning procedures have received particular interest from the astronomical community, since the constant growth of the astronomical surveys and the amount of data that must be analyzed have forced to consider methods that allow to automate processes in a short time. As mentioned above, in this work one of the most important aims is to build an automated classifier using the machine learning methods. First, it is important to state the general classification problem,

2.3.1 The General Classification Problem

When referring to a classification problem, the first distinction to be made is whether there are in the data available a prior knowledge of the class to be predicted. If the response class is known, it is said that a supervised classification will be made. Therefore, in supervised classification there are a set of explanatory variables which have some influence on a discrete response variable. If the discrete response variable takes values 0 or 1, a binary classification algorithm must be used, while if the response takes finite discrete values, a multiclass classification algorithm must be used. In this work, is addressed a binary classification problem. The binary classification methods implemented in this work will be explained briefly below. I refer to Hastie et al. (2009) [38] for more detailed description of the state-of-the-art data mining algorithms.

2.3.2 State-of-the-art Data Mining Algorithms

2.3.2.1 Logistic Regression

Suppose $Y_i \sim Ber(p)$ is a binary response variable, which can be explained by a set of features $x_i = (1, x_{i1} \ldots, x_{ip})$. In logistic regression the main aim is to model directly the conditional probability of the response Y_i given a set of features x_i (i.e., $\mathbb{P}(Y_i = y_i | X_{ij} = x_{ij})$), for that we could assume a particular functional form for link function. The standard logistic function (or Sigmoid) is applied to a linear function of the input features,

$$\mathbb{P}(Y_i = 1 | X_{ij} = x_{ij}) = \frac{1}{1 + \exp(-\theta' x_i)}$$

where $\theta = (\beta_0, \ldots, \beta_p)$. Let $p_i = \mathbb{P}(Y_i = 1 | X_{ij} = x_{ij})$, a simple calculation shows the later expression is equivalent to,

$$\log\left(\frac{p_i}{1-p_i}\right) = \exp(-\theta' x_i)$$

where $\log\left(\frac{p_i}{1-p_i}\right)$ is the logit link function. Logistic regression models are usually fit by maximum likelihood. Note that $\mathbb{P}(Y_i = 1|X_{ij} = x_{ij})$ is a function of θ, which can be expressed as $h_{x_{ij}}(\theta)$, then the log-likelihood function can be written as,

$$\ell(\theta) = -\sum_{i=1}^{n} w_i \left[y_i \log(h_{x_{ij}}(\theta)) + (1 - y_i) \log(1 - h_{x_{ij}}(\theta)) \right]$$

where w_i are the observation weights, which are generally assumed equally distributed (i.e., $w_i = 1/n$). To find the maximum likelihood estimator it is necessary to use optimization methods such as Newton Raphson or Gradient Descent.

2.3.2.2 CART Algorithm

The tree-based methods partition the feature space into a set of rectangles, and then fit a simple model (like a constant $y = c$) in each one. There a lot of tree-based methods which differ in the type of partition and the impurity measure used. The CART algorithm produces binary splits based on the Gini impurity measure, which are defined by,

$$i(k) = \sum_{i=j=1, i \neq j}^{J} p(i|k)p(j|k) = 1 - \sum_{j=1}^{J} p(j|k)^2$$

where $p(j|k)$ is the proportion of class j in node k. Using Gini we first split the space into two regions and model the response by the proportion of the class $y = j$ in each region. We choose the variable and split-point to achieve the best fit. Then one or both of these regions are split into two more regions, and this process is continued, until some stopping rule is applied.

A key advantage of the recursive binary tree is its interpretability. The feature space partition is fully described by a single tree. A disadvantage of the CART algorithm is that they can be extremely sensitive to small perturbations in the data, producing completely different trees.

2.3.2.3 Ensemble Classifiers

The idea of ensemble learning is to build a prediction model by combining the strengths of a collection of simpler base models. The most popular algorithms that follow this methodology are,

- **Bagging (Breiman, 1996 [8]):** Fit many weak classifiers from bootstrap resampling versions of training data and performs the classification by majority vote.

- **Boosting (Freund & Shapire, 1996 [32]):** Fit many weak classifiers to reweighted versions of the training data. The classification is obtained by weighted majority vote.

- **Random Forest (Breiman, 2001 [10]):** Bagging method refinement.

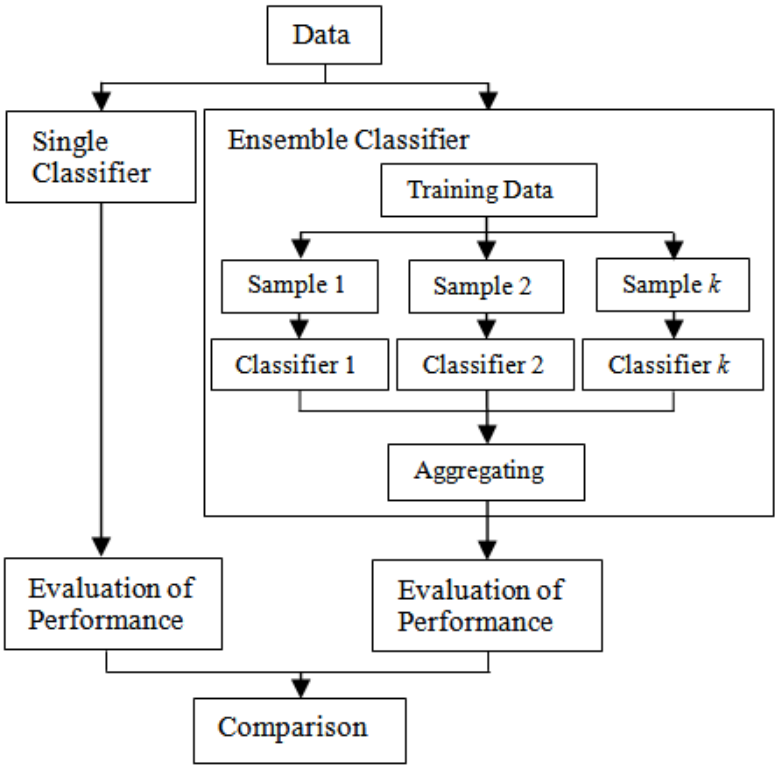


Figure 2.4: Workflow of Ensemble Methods (from Utami, et al., 2014) [72].

2.3.2.4 Bagging

Bagging (Bootstrap aggregation) is a method that combines bootstrapping and aggregating. Usually, the aggregation is performed using the mean in regression problems and majority vote in classification problems. The main idea is reducing the variance of an estimated prediction function through the bootstrap sampling. Bagging seems to work especially well for high-variance, low-bias procedures, such as trees. The procedure is the following.

Algorithm 1 Bagging

Input: Training Data, $(x_{ij}, y_i),\ i = 1, \ldots, N, j = 1, \ldots, P$.

1: Take a bootstrap replicate T_b of the training set T_n.
2: Construct a single classifier $C_b(X_i) = \{1, 2, \ldots, k\}$ in T_b.
3: Combine the basic classifiers $C_b(X_i)$, $b = 1, 2, \ldots, B$ by the majority vote (the most often predicted class).

2.3.2.5 Random Forest

Random forests (Breiman, 2001 [10]) is a substantial modification of bagging that builds a large collection of de-correlated trees, and then averages them. The basic idea is to improve the variance reduction of bagging by reducing the correlation between the trees, without increasing the variance too much. This is achieved in the tree-growing process through random selection of the input variables. Consequently, before each split the algorithm select $m \leq p$ of the input variables at random as candidates for splitting. The procedure is described in the Algorithm 2. Due to the process of sampling the input variables, this method is sensitive to the quality of these inputs.

Algorithm 2 Random Forest

Input: Training Data, $(x_{ij}, y_i),\ i = 1, \ldots, N, j = 1, \ldots, P$.

1: Set the weights $w_m(i) = 1/n, i = 1, 2, \ldots, n$.
2: **for** b in 1 to B
3: Take a bootstrap replicate T_b of the training set T_n.
4: Grow a random-forest tree to T_b, by recursively repeating the followings step for each terminal node of the tree.
   a: Select m variables at random from the P variables.
   b: Pick the best variable/split point among the m.
   c: Split the node into two daughter nodes.
5: Output the ensemble of trees $\{C_b(X_i)\}_1^B$.

2.3.2.6 Boosting

The term "boosting" refers to the process of taking a "weak" learning algorithm (classification or regression) and boosting its performance by training many classifiers and combining them in some way. The main idea of the "boosting" algorithms is to give more weight, in each iteration, to the observations that are harder to classify correctly. Consequently, given initial weights $w_m(i) = 1/n$ the boosting algorithm update these weights after each step based on the error of the classifier in the m-th iteration e_m and the weight

updating coefficient α_m.

There are several versions of the boosting algorithms that differ in the value of α_m and the loss function used. The most popular method for classification is the AdaBoost.M1, which aims to minimize the exponential loss. The Adaboost.M1 algorithm was proposed by Freund and Schapire (1996) [32]. Given a training set $T_n = \{(x_1, y_1), \ldots, (x_i, y_i), \ldots, (x_n, y_n)\}$ where y_i takes values in $1, 2, \ldots, k$, a weak classifier $h_m(x_i)$ is built on this new training set (T_m) and is applied to every training example. Later the error of the classifier in the step m is computed, which it is used to update the weights, which are normalized to sum one. Finally, as in random forest and bagging the majority vote criterion is used to ensemble the weak classifiers. Summarizing, the Adaboost.M1 procedure is shown in the Algorithm 3,

Algorithm 3 Adaboost.M1

Input: Training Data, $(x_{ij}, y_i),\ i = 1, \ldots, N, j = 1, \ldots, P$.

1: Set the weights $w_m(i) = 1/n$, $i = 1, 2, \ldots, n$.
2: **for** m in 1 to M
3: Fit the classifier $h_m(x_i) = \{1, 2, \ldots, k\}$ using weights $w_m(i)$ on T_m
4: Compute $e_m = \sum_{i=1}^{n} w_m(x_i) I\left(h_m(x_i) \neq y_i\right)$ and $\alpha_m = \ln((1 - e_m)/e_m)$
5: Update the weights $w_{m+1}(i) = w_m(i) \exp(\alpha_m I\left(h_m(x_i) \neq y_i\right))$ and normalize them.
6: Output the final classifier $h_m(x_i) = argmax_{j \in Y} \sum_{b=1}^{B} \alpha_b I\left(h_m(x_i) = j\right))$

Note that in the weight updating formula (step 5 in Algorithm 3), the expression $I(h_m(x_i) \neq y_i)$ causes that the weights of the observations wrongly classified increases and the observations rightly classified decreases.

As mentioned above, there are some variations of the "boosting" algorithm which can be obtained by changing the α_m coefficient. As for example, Breiman (1998) [9] propose to use $\alpha_m = 1/2 \ln((1 - e_m)/e_m)$ which is half of α_m proposed originally by Freund and Schapire. Another example is the SAMME algorithm proposed by Zhu [76] which uses $\alpha_m = \ln((1 - e_m)/e_m) + \ln(k - 1)$ where k is the number of classes of the response variable. These three flavours of boosting are implemented in the "adabag" package of R [1].

In addition, note that algorithm 3 uses a link function $\eta(x) = sign(x)$ and an exponential loss function $L(y_i, g) = \exp(y_i g)$. There are variants of the boosting algorithm that uses different loss functions. One example is the L_2 Boost (Friedman et al (2001) [35] that uses the logistic loss function $L(y_i, g) = \log(1 + \exp(-y_i g))$. In addition, the boosting algorithms can also differ in the link function used. For example the Real-AdaBoost uses

$\eta(x) = 0.5 \log\left(\frac{x}{1-x}\right)$ and the Gentle-Adaboost uses $\eta(x) = x$, both algorithms are presented in Friedman et al (2000) [33]. The use of different link function attempts to overcome a problem to find an optimal classification model using as output the object's predicted label, which only gives a partial view of the efficiency of the classifier. These variations of boosting are implemented in the "ada" package of R [22].

2.3.2.7 Support Vector Machine

The Support Vector Machine (Cortes & Vapnik, 1995 [21]) is a generalization of the optimal separating hyperplanes defined when two classes are linearly separable (also called the maximal margin classifier). This optimal separation can be achieved enlarging the feature space in order to accommodate a non-linear boundary between the classes, using kernels (also called the "kernel trick").

Let the hyperplane $< w, x > +b = 0$ where w and b are the weights and bias vector respectively. In addition, the operator $< w, x >$ is the inner product between the vector w and the features matrix x. A maximum margin classifier consists in finding the parameters w and b for an optimal hyperplane. Figure 2.5 illustrates a geometric construction of the corresponding optimal hyperplane in a linear classification problem.

Now, let $\Phi : X \rightarrow H$ denote a nonlinear transformation from the input space $X \subset \mathbb{R}^m$ to the feature space H where the problem can be linearly separable. We may define the corresponding optimal hyperplane as follows:

$$< w, \Phi(x) > +b = 0 \tag{2.3.1}$$

corresponding to the decision function,

$$f(x) = sign(< w, \Phi(x) > +b) \tag{2.3.2}$$

Here, the optimal hyperplane computed in the feature space is,

$$\sum_{i=1}^{n} \alpha_i^* y_i \Phi^T(x_i)\Phi(x_i) = 0 \tag{2.3.3}$$

Consequently, the kernel can be defined as a function $K(x, x') \ \forall \ x, x' \in X \subset R^m$ that satisfy $K(x, x') = \Phi^T(x)\Phi(x')$. The most popular kernels are the linear, polynomial, radial and sigmoid, which are defined in Table 2.2

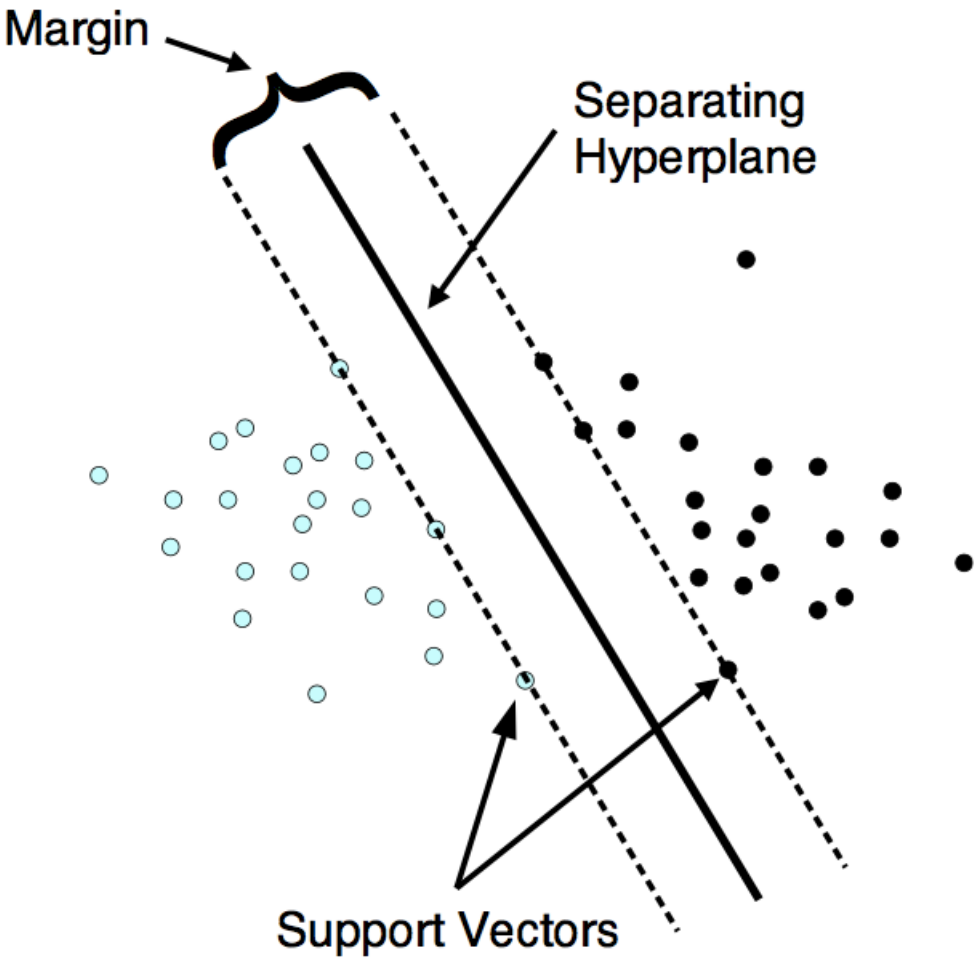


Figure 2.5: Illustration of the optimal hyperplane in SVC for a linearly separable (from Meyer, 2001) [47].

2.3.2.8 Artificial Neural Network

A neural network is a two-stage classification model. This network applies both to regression or classification problems. A single layer perceptron is the simplest type of neural network. But, this method is only capable of solving linearly separable problems.

The limitations of the single layer network have led to the development of multi-layer feed-forward networks (also called Multilayer perceptron (MLP)). Typically, an MLP neural network consists of an input layer, one or more hidden layers, and an output layer, as shown in Figure 2.6.

Table 2.2: *Kernels commonly used in SVMs.*

Kernel	Formula	Parameters
Linear	$K(x, x') =< x, x' >$	
Polynomial	$K(x, x') = (\text{scale} < x, x' > +\text{offset})^{\text{degree}}$	scale, offset and degree
Radial Basis	$K(x, x') = \exp(-\text{scale}\|\|x - x'\|\|^2)$	scale
Sigmoid	$K(x, x') = tanh(\text{scale} < x, x' > +\text{offset})$	scale, offset

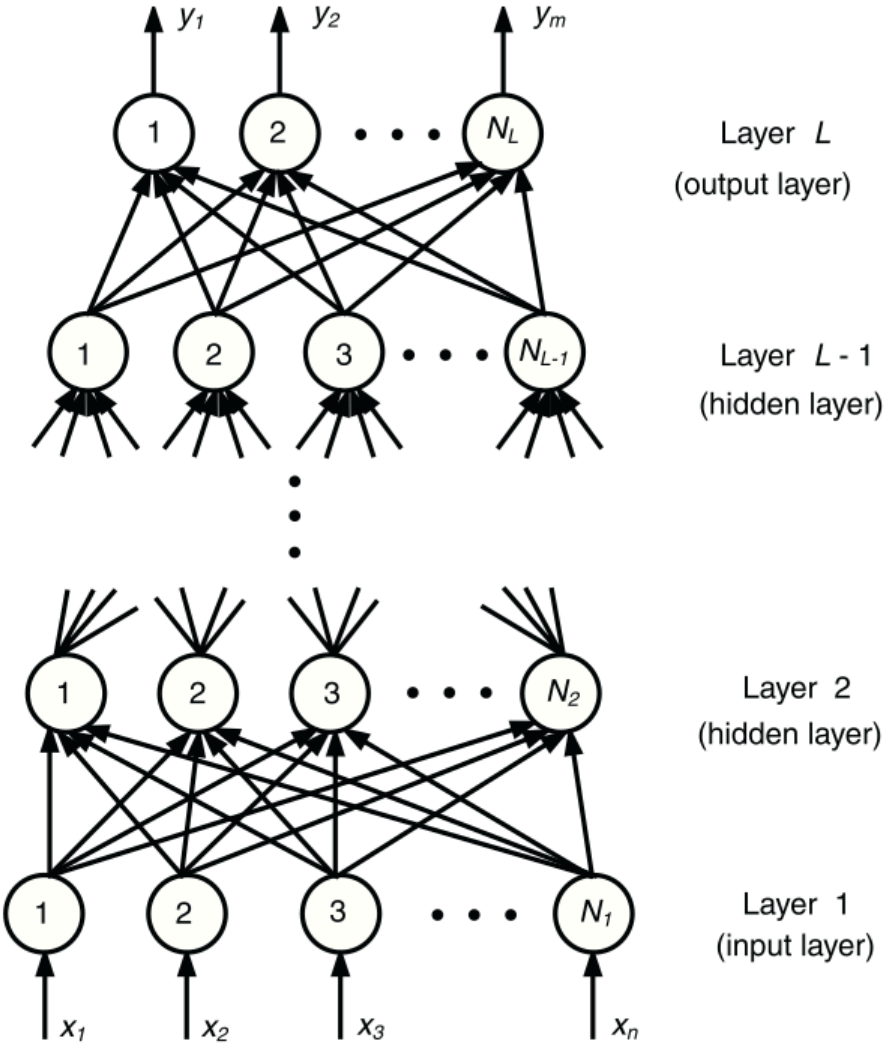


Figure 2.6: Illustration of multi-layer feed-forward networks (from Zhang, 2000) [75].

In this representation, the neurons are ordered into layers so that each neuron in a particular layer is connected with all neurons in the next layer. The connection between the i-th neuron and j-th neuron is characterized by the weight coefficient w_{ij}. Additionally, an

extra weight parameter for each neuron, w_{i0} is introduced, representing the bias for i-th neuron of l-th layer.

Suppose the simplest multi-layer network, with only one hidden layer. The first layer involves M linear combinations of the p-dimension inputs. The output value of the k-th neuron y_k is determined by the following equation,

$$y_k = g\left(\sum_{j=0}^{M} w_{kj}^{(2)} \sigma\left(\eta_i + \sum_{i=0}^{p} w_{ik}^{(1)} x_i\right)\right) \tag{2.3.4}$$

where $w^{(l)}$ are the weights of the l-th layer $l = 1, 2$. $\sigma(.)$ and $g(.)$ are the activation functions for the hidden unit and output respectively. Some examples of activation functions are shown in Table 2.3

Table 2.3: *Activation functions commonly used in MLP.*

Activation Function	Formula
Sigmoid	$\sigma(\gamma) = \frac{1}{(1+e^{-\gamma})}$
Arc-Tangent	$\sigma(\gamma) = \left(\frac{2}{\pi}\right) arctan(\gamma)$
Hyperbolic-Tangent	$\sigma(\gamma) = \left(\frac{e^{\gamma}-e^{-\gamma}}{e^{\gamma}+e^{-\gamma}}\right)$

All these functions are bounded, continuous, monotonic and continuously differentiable (Zhang, 2000). [75]. This neural network can be trained using gradient descent over an error function. The evaluation of the error derivatives proceeds using a version of the chain rule of differentiation, referred to as back-propagation of error.

It is well known that an artificial neural network (ANN), either single layer or multiple layers, is sensitive to the hyperparameters such as the number of hidden layers, the number of neurons at each hidden layer, batch size for stochastic gradient descent.

2.3.2.9 Regularized Regression

Finally, classification algorithms based on regularization techniques will also be used. Among them, the Lasso (Tibshirani 1996 [65]) is a very popular method for regression that uses an ℓ_1 penalization, this implies that the Lasso coefficients are the solutions that minimize the loss function,

$$PRSS(\beta)_{\ell_1} = (y - X\beta)^T (y - X\beta) + \lambda ||\beta||_1 \tag{2.3.5}$$

where $||\beta||_1 = \sum_{j=1}^{p} |\beta_j|$ and λ is a tuning parameter that controls the amount of regularization. An alternative to this method is the Ridge regression which differs to Lasso in that uses a ℓ_2 penalization. The ridge coefficients can be found by minimizing the following loss function,

$$PRSS(\beta)_{\ell_2} = (y - X\beta)^T(y - X\beta) + \lambda||\beta||_2^2 \tag{2.3.6}$$

where $||\beta||_2^2 = \sum_{j=1}^{p} \beta_j^2$. The main difference between these two methods is that unlike ridge regression, the lasso is not a linear estimator, i.e., there is no matrix H such that $\hat{y} = Hy$. In addition, ridge regression is known to shrink the coefficients of correlated predictors towards each other. Lasso, on the other hand, tend to pick one of the correlated predictors and ignore the rest (Friedman et al, 2010 [34]).

Another characteristic of the Lasso is that some of the coefficients are shrunken toward to zero, such solutions are said to be sparse. In order to give flexibility to these methods, Zou and Hastie (2005) [77] defined the elastic net which is a mixture of the lasso and ridge penalties. This method is particularly useful when $p >> n$ or when there are many correlated predictor variables. Here, the elastic net criterion is

$$PRSS(\beta)_{\ell_2} = (y - X\beta)^T(y - X\beta) + \lambda_2||\beta||_2^2 + \lambda_1||\beta||_1 \tag{2.3.7}$$

where λ_1 and λ_2 are fixed and non-negative coefficients. Let $\alpha = \frac{\lambda_2}{\lambda_1+\lambda_2}$ we can rewrite the elastic net penalty as $\alpha||\beta||_2^2 + (1-\alpha)||\beta||_1$, which is a convex combination of the lasso and ridge penalty. In this sense, when $\alpha = 1$ the elastic net becomes simple ridge regression and when $\alpha = 0$ we have the Lasso penalty.

2.3.3 Measuring classifier performance

A confusion matrix is a table that is often used to describe the performance of a supervised classification model (or "classifier"). The main idea is to show the true and predicted classes in the same table. The confusion matrix itself is relatively simple to understand and allows to compute several measures of quality from it.

Suppose a binary classification problem, the 2×2 confusion matrix has de form,

True\Predicted	0	1
0	(TN) n_{11}	(FP) n_{12}
1	(FN) n_{21}	(TP) n_{22}

where,

- TP = True positive. Is the number of observations with true class $y = 1$ classified correctly.
- FN = False negative. Is the number of observations with true class $y = 1$ classified incorrectly.
- FP = False positive. Is the number of observations with true class $y = 0$ classified incorrectly.
- TN = True negative. Is the number of observations with true class $y = 1$ classified correctly.

From this table we can compute several measures of quality as,

Precision

Is the probability that a randomly selected object predicted to be in a target class does belong to the target class. In this work the precision is denoted by P. The precision can be computed as,

$$P = \frac{TP}{TP + FP} \tag{2.3.8}$$

Note that the false-discovery rate, that is, the rate of false positives, is $1 - P$.

Recall

Recall is the probability that a randomly selected object belonging to a target class is indeed predicted to be in that class. In this work the recall is denoted by R. It is also called True positive rate or sensitivity and can be computed as,

$$R = \frac{TP}{TP + FN} \tag{2.3.9}$$

F measure

The Precision-recall F_β measure is defined as the weighted harmonic mean between P and R, that is,

$$F_\beta = \frac{(\beta^2 + 1)PR}{(R + \beta^2 P)} \tag{2.3.10}$$

It is a measure of the accuracy of the classifier, where a perfect accuracy would imply values close to one. In this formulation, the mean is balanced if $\beta = 1$. In this case we call F_β as F1 measure (F_1).

Area under the curve (AUC)

AUC is the area under the so-called receiver operating characteristic (ROC) curve, which shows the true-positive rate as a function of the false-positive rate. Values close to 1, which is the maximum possible, are best because they indicate a classifier that quickly achieves a high true-positive rate with a correspondingly low false-positive rate.

Chapter 3

Light Curves Classification

One of the most important goals of this work is to develop an automated procedure to classify variable stars in the VVV survey. Particularly, it is important to detect the pulsating stars, like for example the RR-Lyraes or Cepheids, since they are essential to build a three-dimensional map of the Galactic bulge. In order to make the classifier, a procedure consisting in eight steps was created, starting with the data cleaning procedure and finishing with the final classifier. To reach this final classifier with the best possible performance in training data, all the state-of-the-art data mining algorithms reviewed in section 2.3.2 were implemented. Each step in the classification is explained in this chapter.

First, to build the classifier the only information that we take into account is the time series of the light curve of each star observed irregularly in time by the VVV survey using the K_s-band. Each time series is composed by the following three attributes,

- **Time:** heliocentric Julian day.
- **Magnitude:** Measure of the brightness of a variable star.
- **Error:** Magnitude Error.

Generally the light curves of variable stars are fitted using an harmonic model. The p-harmonic model is defined by:

$$y(t) = \beta_0 + \sum_{j=1}^{p} (\alpha_{1j} sin(2\pi f_1 jt) + \beta_{1j} cos(2\pi f_1 jt)) + \epsilon(t) \tag{3.0.1}$$

Following Debosscher et al. (2007) [23] and Richards et al. (2011) [56] $p = 4$ is assumed to fit the light curves of variables stars. Before fitting this model, we need to estimate the dominant frequency f_1. We use the generalized Lomb-Scargle periodogram (GLS Zechmeister & Kurster 2009 [74]) to obtain f_1. We restricted the periodogram to frequencies satisfying $f_1 < 5$ day^{-1}. Both the harmonic model and GLS periodogram are

very useful in the first steps of the classifier construction procedure, since for example, they allow us to obtain features to train the classifier and also eliminate outliers and non-periodic light curves.

In each VVV field, millions of stars can be observed, but not all of them will be useful to train the classifier. In this sense the training set must be cleaned in order to use only light curves that make a contribution to the classifier. In addition, the light curves of these stars must also be cleared from outliers and high standard errors.

3.1 Light-curve extraction and pre-processing

One of the most important steps in the classifier building is the pre-processing of the light curves. The aim is to improve the performance of the classifier by removing noisy information of the training set. The data cleaning procedure was performed in the following steps:

Removing Non-Variable Stars: First, the stars that are not variables were removed from the training set. Consequently, stars with light variations were selected using the robust version of the Welch/Stetson variability index J [62, Eq. 1] to the K_s-band light curves. Candidate variable stars have been selected by requiring the value of the Stetson index above the 0.1% significance level corresponding to pure Gaussian noise. The significance level of this statistic was derived from Monte Carlo simulations for several numbers of sample sizes.

Removing Observations with Large Errors: After selecting the curves that showed evidence of variability, we proceeded to eliminate individual observations that have anomalously large magnitude errors, since these observations were measured with much uncertainty due to anomalous observing conditions and therefore they provide little information. Let $\{\epsilon_i\}_{i=1}^n$ be the set of measurement errors estimates for a light curve with n points, then we eliminate the observations with ϵ_i values $> 5\sigma$, where $\sigma = \mathbb{V}(\{\epsilon_i\})$.

Removing Outliers Observations: In addition, an outlier rejection procedure was performed for each light curve. The procedure consists in first estimating the dominant frequency f_1 using the GLS periodogram. This frequency is used to fit the harmonic model (3.0.1) to the light curve. Later, a smooth estimation of the phased light curve was obtained using smoothing splines from the R function `smooth.spline`, where the parameter that controls the smoothing (spar) is set equal to 0.8 (this value was chosen based on a best-fit measure of folded light curves). Then, we performed an iterative σ-clipping procedure to the residuals around the smooth model of the phased light curve. Assuming

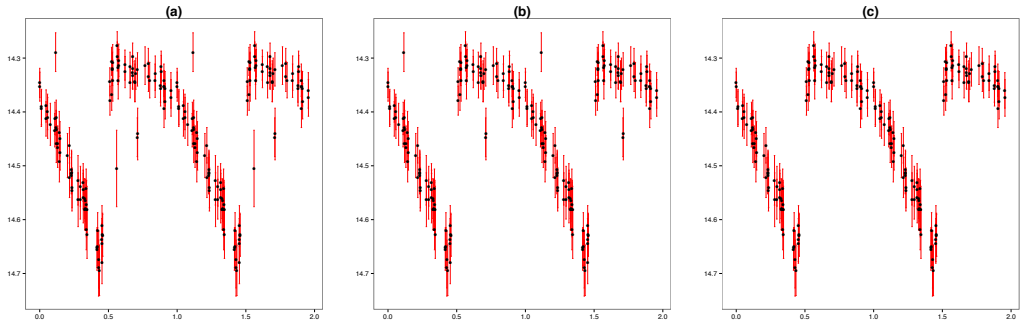


Figure 3.1: a) Original light curve of an RR*ab* star observed in B295 field folded in its period. b) Light curve of the RR*ab* star after the elimination of one observation with large magnitude error . c) Light curve of the RR*ab* star after the elimination of two outliers observations.

Gaussian errors, we removed outliers greater than 4σ, where $\sigma = 1.4826\times$ MAD and MAD is the median absolute deviation (the explicit formula is in Table 3.2).

In Figure 3.1 is shown an example of the last two steps described (i.e., elimination of large magnitude errors and outliers observations) of the data cleaning procedure performed in a RR*ab* variable star from B295 Field. It can be observed that in two steps three observations were removed from the light curve, reaching a perfectly cleaned light curve. Note that two phases are shown in each Figure of 3.1 to appreciate more clearly the shape of the light curve. The same procedure will be followed for each light curve plotted in this work.

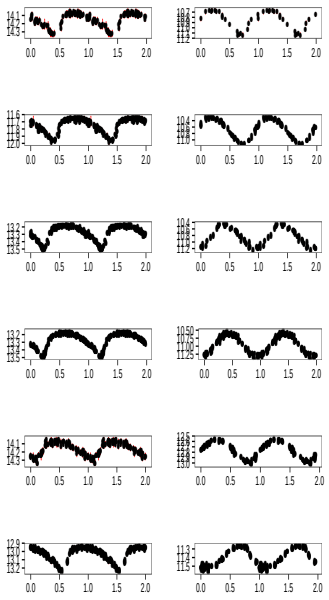
Removing Noisy Light Curves: We also eliminated from our sample light curves with either too low signal-to-noise ratios or irregular phase coverage. Therefore, we eliminated all light curves whose scatter around the phased light curve was not significantly different from the raw light curve. To achieve this, the feature p2p_scatter_pfold_over_mad was used so that 1/p2p_scatter_pfold_over_mad must be greater than 0.8. This feature consists in the median absolute deviation of the phased light curve times the median absolute deviation of the raw light curve (the explicit formula will be shown in Table 3.2).

To eliminate curves with incomplete phase coverage, we eliminated all curves where $1 - \Delta\phi_{\max} < 0.8$, where $\Delta\phi_{\max}$ is the maximum of the consecutive phase differences $\{\phi_{i+1} - \phi_i\}_{i=1}^{N}$, where N is the number of measurements, and we took $\phi_{N+1} - \phi_N \equiv 1 + \phi_1 - \phi_N$.

Removing Non-Periodical Light Curves: Later, in order to select variable stars with periodical behavior, we eliminated all light curves whose highest Lomb-Scargle peak satisfy ≤ 0.3. The threshold was determined using the Lomb-Scargle peaks of known RR*ab*

in our VVV training fields, where none of them has a GLS peak with a value lower that the chosen threshold. Finally, as the VVV observations are clustered, usually in groups of 4, we eliminated all the light curves with ≤ 50 observations, and therefore with < 15 epochs approx.

After we performed the data cleaning procedure, the classifiers are created. Here, we must consider that the observations of the light curves are sampled unequally spaced in time due to different observation constraints. Consequently, the light curves also differ in their sample sizes. This issue makes the classification of light curves a big challenge. To implement the machine learning algorithms, we must represent each light curve as a vector of the same length for all stars composed with features of these light curves, as is represented in figure 3.2 .



$\Rightarrow$

Light Curve	Y	X_1	...	X_p
1				
2				
.				
.				
.				
n				

Figure 3.2: Illustration of the process in which each light curve in the training set is represented as a vector of features.

3.2 Feature Extraction of Light Curves

The features extracted from light curves can be divided in two groups, depending whether they are extracted directly from the parameters of the harmonic model fitted to the light curves or not. Henceforth, the first group will be called as the periodic features and the second one as the non-periodic features.

3.2.1 Periodic Features

After each light curve has gone through the preprocessing step described in Section 3.1, an harmonic model is fitted to them in order to extract the features from the parameters of this model. Following the previous works of Debosscher et al. (2007) [23], Richards et al. (2011) [56] or Elorrieta et al. (2016) [27] the features were extracted from the parameters of the harmonic model with n frequencies and p components where $n = 3$ and $p = 4$.

In Elorrieta et al. (2016) [27] we propose to add the measurement errors in the fitted model. Therefore, we fit the raw light curve using a harmonic model where the parameters are estimated by weighted least-squares with weights equal to the inverse measurement variances σ_i^{-2}. The full model is given by,

$$\hat{y}(t) = \sum_{i=1}^{n}\sum_{j=1}^{p}(\hat{\alpha}_{ij}sin(2\pi f_i jt) + \hat{\beta}_{ij}cos(2\pi f_i jt)) + \hat{a} + \hat{b}t. \tag{3.2.1}$$

This model is fitted using the following procedure. Let $y(t)$ be the observed magnitude at time t from a given variable star. Let $\hat{y}(t) = \hat{a} + \hat{b}t$ be the linear trend estimated from a linear regression model of the light curve. Therefore, $r(t) = y(t) - \hat{y}(t)$ are the residuals of this model. Next, we iterated $n = 3$ times the following two steps.

1. Using the GLS periodogram we determine the dominant frequency f_i of $r(t)$ as the highest peak in the periodogram. Next, the frequency f_i is used to fit the harmonic model (3.0.1) using the method of weighted least-squares. We denote $\hat{z}(t)$ the fitted values of this model

2. We reassigned $r(t)$ through $r(t) \leftarrow r(t) - \hat{z}(t)$

In words, we first subtract the linear trend from the light curve. The intercept and the slope of the linear trend are the first features which are computed. Next, we compute the first frequency f_1 from the largest peak of the GLS periodogram. The frequency and the GLS peak value are also features that are used in the classifier to represent the variable stars. Using this frequency, we fit an harmonic model with p components (3.0.1). The Fourier parameters $\hat{\alpha}_{1j}$ and $\hat{\beta}_{1j}$, with $j = 1, \ldots, 4$ can also be used as features. However, as these parameters are sensitive to the time translations we used a time invariant representation of them. Consequently, we transformed the Fourier coefficients into a set of amplitudes A_{ij} and phases ϕ'_{ij} as follows:

$$\begin{aligned} A_{ij} &= \sqrt{\hat{\alpha}_{ij}^2 + \hat{\beta}_{ij}^2} \\ PH_{ij} &= \arctan(\sin(\phi_{ij}), \cos(\phi_{ij})) \end{aligned}$$

Table 3.1: List of the 40 light-curve periodical features used in this work.

Feature name	Description	Reference[b]
intercept (slope)	Intercept (slope) of a linear regression to the light curve	D07
A_{ij}	Amplitude of the i-th frequency and j-th harmonic	D07
ϕ_{ij}	Phase of the i-th frequency and j-th harmonic	D07
f_i	i-th frequency obtained from GLS	D07
P_i	Peak in the GLS periodogram of the i-th frequency	D07
var_i	Variance left after i-th fit of Fourier model	D07
mse_i	Mean squared error of i-th fit of Fourier model	D07
freq_amplitude_ratio_21 (31)	Amplitude ratio of 2nd (3rd) to 1st component of the Fourier model	R12
freq_frequency_ratio_21 (31)	Frequency ratio of 2nd (3rd) to 1st component of the Fourier model	R12

[a] Note that $i = 1, 2, 3$ and $j = 1, \ldots, 4$. In addition, the phases ϕ_{1j} are not used as features because are set as zero. [b] D07=Debosscher et al. [23]; R12=Richards et al. [57]

where $\phi_{ij} = \arctan(\hat{\beta}_{ij}, \hat{\alpha}_{ij}) - \frac{jf_1}{f_1} \arctan(\hat{\beta}_{ij}, \hat{\alpha}_{ij})$. In this notation, ϕ_{11} was chosen as the reference and was set to zero so that PH_{1j} takes values in the interval $[-\pi, \pi]$.

Later, the fitted harmonic model $\hat{z}(t)$ was subtracted from $r(t)$. Then we proceed with the second iteration. So, we obtain a new frequency f_2 and a new GLS peak periodogram from the residuals and then the process described above is repeated until n frequencies were found and n harmonic models were fitted.

As can be seen, an important set of features was derived from the fitting procedure. Table 3.1 lists all the features that can be computed from the harmonic fit, along with a short description and a reference to the literature when adopted from previous work.

3.2.2 Non-Periodic Features

From the fitting procedure we can get 40 periodical features, to which 28 additional features will be added. They will be called as non-periodic features. Most of them have been proposed by Debosscher et al. (2007) [23], Richards et al. (2011) [56] and Richards et al. (2012) Richards et al. [57]. In Table 3.2 these features are detailed.

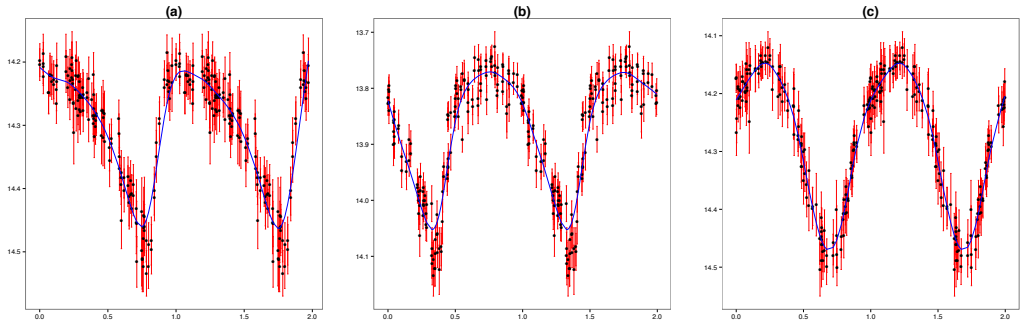


Figure 3.3: a) Folded light curve of an RR*ab* star with $R_1 = 0.46$ observed in B295 field of the VVV. b) Folded light curve of an RR*ab* star with $R_1 = 0.77$ observed in B295 field of the VVV . c) Folded light curve of a variable star with $R_1 = 1.18$ observed in B295 field of the VVV.

However, we propose two new features, which are specifically designed to better distinguish RR*ab* from eclipsing binaries such as those of the W UMa type, which were our most troublesome contaminant in NIR. The basic idea of these features is to represent the asymmetrical behavior typical of the RR*ab* light curves.

We called one of these features R_1, which captures the asymmetrical behavior of the phased light curve by computing the ratio between the elapsed phase until reach the first minimum and the second maximum of the smoothed light curve. In other words, let A_1 be the phase difference of the first maximum and the first minimum and A_2 be the difference in the phases of the first minimum and the second maximum. The R_1 measure can be computed by,

$$R_1 = \frac{A_1}{A_2} = \frac{\phi_{max,1} - \phi_{min,1}}{\phi_{min,1} - \phi_{max,2}}$$

where $\phi_{max(min),i}$ denotes the phase corresponding to the i-th maximum (minimum) of a phased curve. In order to show an example of how this feature works, we compute the feature R_1 for three light curves of the B295 field of the VVV. These three light curves are plotted in Figure 3.3, the first two of them (Figure a)-b)) corresponds to RR*ab* variable stars and its corresponding R_1 is 0.46 and 0.77 respectively. The R_1 for the third light curve is 1.18. Note that the increment in the values of the feature R_1 are consistent with that the light curves have a more symmetrical behavior.

Table 3.2: List of the 28 light-curve non-periodical features used in this work.

Feature name	Description[a]	Reference[b]
skew	Skewness of y	R11
small_kurtosis	Small sample kurtosis of y	R11
std	Standard deviation of y	R11
max_slope	$\max\{(y_{i+1} - y_i)/(t_{i+1} - t_i)\}$	R11
amplitude	$\max(y) - \min(y)$	R11
median_absolute_deviation	$\text{median}(\lvert y - \text{median}(y)\rvert)$	R11
median_buffer_range_percentage	Fraction of points in $\{y\}$ with amplitude within < 0.1 of median(y)	R11
pair_slope_trend	For the set $\{y_{N-29+i} - y_{N-30+i}\}_{i=2}^{30}$ the ratio N_+/N_-	R11
flux_percentile_ratio_mid_k	$F_{50-k/2,50+k/2}/F_{5,95}$	R11
percent_amplitude	$\max(\lvert F - \text{median}(F)\rvert)/\text{median}(F)$	R11
percent_difference_flux_percentile	$F_{5,95}/\text{median}(F)$	R11
freq_model_max(min)_delta_mags	Difference in magnitudes between the two maxima (minima) of y_{2P}	R12
freq_model_phi1_phi2	$(\phi_{min,1} - \phi_{max,1})/(\phi_{min,1} - \phi_{max,2})$ (for $y_{m,2P}$)	R12
freq_rrd	Boolean that is 1 if freq_frequency_ratio_21 (or 31) is within 0.0035 of 0.746	R12
gskew	(median(y) − median(y_0)) + (median(y) − median(y_{1-p})) with $p = 0.03$	R12
scatter_res_raw	$\text{MAD}(y - y_m)/\text{MAD}(y)$	D11
p2p_scatter_2praw	$\sum_{i=2}^{N}(y_{2P,i+1} - y_{2P,i})^2 / \sum_{i=2}^{N}(y_{i+1} - y_i)^2$	D11
p2p_scatter_over_mad	$\sum_{i=2}^{N} \lvert y_{i+1} - y_i\rvert/(N-1)\text{MAD}(y)$	D11
p2p_scatter_pfold_over_mad	$\sum_{i=2}^{N} \lvert y_{P,i+1} - y_{P,i}\rvert/(N-1)\text{MAD}(y)$	D11
fold2P_slope_10percentile (90)	10th (90th) percentile of slopes y_{2P}	R12
R1[c]	$(\phi_{max,1} - \phi_{min,1})/(\phi_{min,1} - \phi_{max,2})$ (for $y_{s,2P}$)	R12, E16
R2	$(y_{s,2P}(\phi_{max,1}) - y_{s,2P}(\phi_{min,1}))/(y_{s,2P}(\phi_{min,1}) - y_{s,2P}(\phi_{max,2}))$	E16

[a] We use the following notation: the light-curve magnitudes at times t_i are denoted by $y(t_i)$ or y_i, the magnitudes phased with period P at phase ϕ_i as $y_P(\phi)$, the harmonic (Fourier) model as y_m, the smooth spline mode as y_s. $\phi_{max(min),i}$ denotes the phase corresponding to the i-th maximum (minimum) of a phased curve, $y(\phi_{max(min),i})$ the corresponding value. N_+ and N_- denote the number of positive and negative members of a set, respectively. $F_{a,b}$ is the difference in flux between the percentile a and b of the fluxes implied by y. $y_{a:b}$ are the subset of y whose members lie between the a-th and b-th percentile. [b] D07=Debosscher et al. [23]; R11=Richards et al. [56]; R12=Richards et al. [57]; D11=Dubath et al. [26];E16=Elorrieta et al. [27] [c] This feature is the same as freq_model_phi1_phi2, but uses y_s instead of y_m.

3.3 A Machine Learned Classifier for RR Lyrae in the VVV Survey

In this section we describe a supervised machine-learned classifier constructed for assigning a score to a K_s-band VVV light curve that indicates its likelihood of being ab-type RR Lyrae. The performance of automated classifiers of variable sources in the optical has been assessed in several previous studies [e.g., 23, 26, 56, 51, 44].

The classifier built in this work differs from those constructed in the referred studies in two aspects. First, this classifier was designed to detect only variable stars of the RR*ab* class. As mentioned in the background section, this is due to the RR Lyrae stars are of particular importance to produce the three-dimensional map of the Galactic bulge. In this specific work, we decided to restrict ourselves only to the RR*ab* stars, since the RR*c* stars have smaller amplitudes (hence noisier light curves) and are frequently very difficult to distinguish from eclipsing binaries.

The second difference regarding previous studies is which these are performed in the optical, while we use the light curves from the VVV survey to build the classifier, which were observed in the near-infrared (NIR). Consequently, it is important to highlight the differences between light curves observed in optical and near-infrared to classify the RR*ab*.

3.3.1 RR*ab* classification in the VVV

The near-infrared (NIR) offers additional challenges as compared to the optical one. First, it is harder to classify RR*ab* in the infrared because their amplitudes are smaller than in the optical. Second, there are not many NIR high-quality light curves with which supervised classifiers can be trained. Third, the NIR light curves have measure with greater errors than the optical. The Figure 3.3.1 shows an example of the light curves observed in the optical and near infrared of a known RR*ab*, where it can be noticed a very symmetric light curve in the infrared (lower panel).

Despite all these difficulties, it is necessary to make the classifier using VVV data, since the near infrared has some desirable properties. For example, in near infrared the distance scale of pulsating stars can be determined most accurately, since the period-luminosity (PL) relations can be determined with lower dispersion (Navarrete et al. (2017) [50]). A precise estimation of distance is essential to build the three-dimensional map of the Galactic bulge. In addition, as the dust is transparent to the NIR filters, it is possible to detect objects that are enshrouded in circumstellar dust. (Matsunaga et al. (2017) [46]).

Consequently, the near-infrared (NIR) surveys can detect objects that cannot be found in the optical.

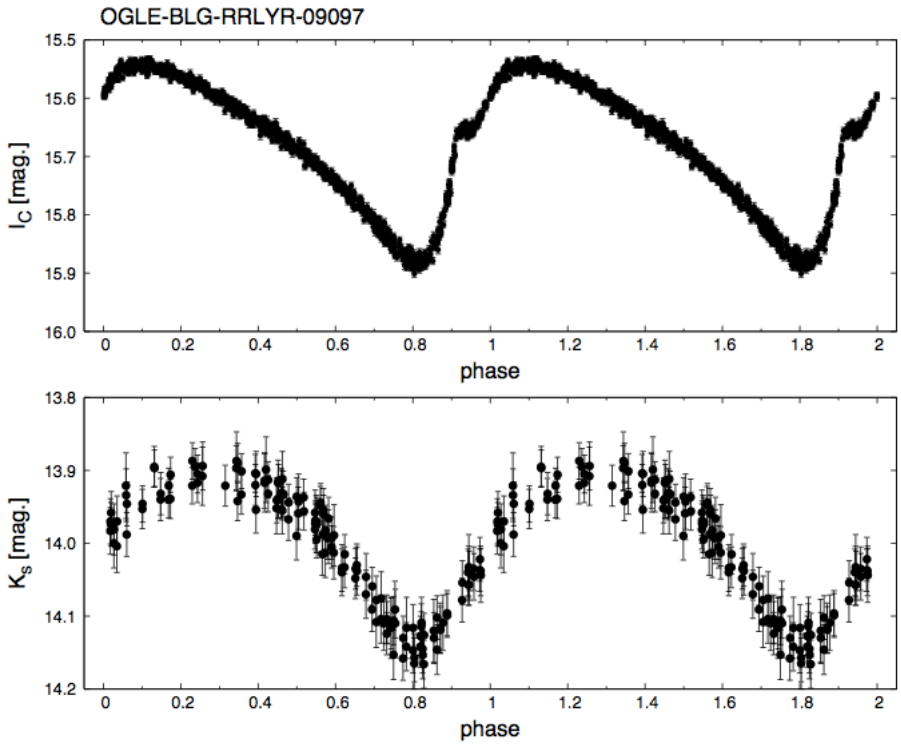


Figure 3.4: Example of a known RR*ab* classified by OGLE using an optical I_C light curve (upper panel). It shows a very symmetric light curve in the infrared (lower panel, K_s light curve from the VVV).

3.3.2 Classification Procedure

The procedure to build the classifier was detailed in Figure 3.5. Here, eight key steps in the construction of the classifier are described. The first four steps were detailed in previous sections, since are related with the procedure in which we take the light curve of each variable star and transform it into a vector of features ready to be used as input in the data mining classification algorithms. These steps are, the pre-processing step, the period estimation by GLS, the light curves modeling and the feature extraction.

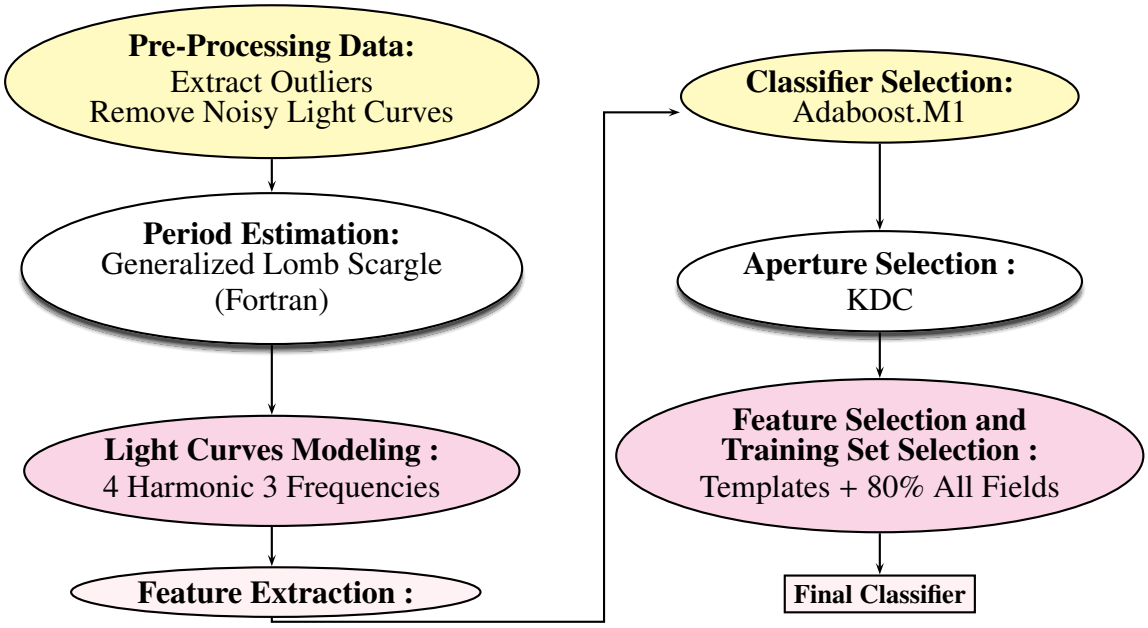


Figure 3.5: Flowchart of the classifier building phase.

From now the next steps will be detailed. These steps are related with the construction of the optimal classifier, choosing the best classification algorithm, the photometry aperture, the most important features and the training sets. Finally, from this analysis we take a decision about the final classifier.

3.3.3 Training sets

Since the aim of this classifier is to detect RR*ab* stars, the response variable which the classifier was trained is defined as,

$$y = \begin{cases} 1, & \text{if the variable star belongs to the RR}ab\text{ class} \\ 0, & \text{if not} \end{cases}$$

Therefore, here a supervised classification scheme must be used. Consequently, the training sets must have known instances of the RR*ab* class, ideally observed with a cadence and precision similar to that of the target data that arises from the VVV. To retrieve a training set from the VVV itself, we used light curves consistent with being variable from the fields B293, B294, and B295, located in the center of the bulge area (around Baade's window, see Figure 2.3) .

Dataset	Class	N
VVV Templates	RR*ab*	1603
	Other	1063
B293 Field	RR*ab*	277
	Other	4869
B294 Field	RR*ab*	207
	Other	5448
B295 Field	RR*ab*	178
	Other	4056

Table 3.3: Number of RR*ab* versus other classes in the training datasets.

These fields were chosen because the OGLE-III survey [63] has also observed in the same field. The OGLE-III catalog is vast and contains known variable stars of several classes. Therefore, we assumed that all RR*ab* in the three chosen fields are present in the OGLE-III catalog. So, can be performed a match between the RR*ab* of the OGLE-III catalog and the light curves extracted from VVV data. In addition to the training set above, we used the NIR light curves belonging to the VVV Templates project [4].

Table 3.3 shows the numbers of RR*ab* light curves versus those belonging to other classes in each of the training datasets considered to build the classifier.

3.3.4 Choice of classifier

To assess the performance of the classifiers, we estimated four measures of quality using ten-fold cross validation: precision, recall, F_1 , and AUC (defined in section 2.3.3). In ten-fold cross-validation, the classifier is trained with nine tenths of the training set, and remaining tenth is used as testing set. The basic idea is to assess the performance of the classifier in new data which not was used in the training process. This process is repeated ten times, and each time, a different tenth of the training set is used as testing set. In each step, the performance measures described above are computed. Finally, each of them is aggregated by the average and the estimated error is computed by the standard deviation.

To choose the best classifier for RR*ab* a several classifiers were tested, which are described in the section 2.3.2. The classifiers were implement using functions in the R language [64] according with the Table 3.4.

To test the classifiers performance, we use as training set the data from VVV templates plus 80% field B293, 80% field B294, and 80% field B295 (we show below that this particular choice of training set is representative of the other fields). The cross-validation estimates of the performance that resulted after training all of the classifiers listed above

Algorithms	Description	R Source Code	Package
Logistic	**Logistic Regression**	glm(formula,data,family=binomial())	stats
Cart	**Classification and Regression Trees**	rpart(formula,data)	rpart
RF	**Random Forest**	randomForest(formula,data,importance=TRUE, proximity=TRUE,ntree=500,mtry=20)	randomForest
SBoost	**Stochastic Boosting**	ada(formula,data,loss=c('exponential'), type=c('discrete'),iter=500)	ada
Ada.M1	**AdaBoost.M1**	boosting(formula,data,boos=TRUE,mfinal=500, coeflearn = 'Breiman'))	adabag
SVM	**Support Vector Machine**	svm(formula,data,kernel='lineal',probability =TRUE)))	e1071
Lasso	**Lasso Regression**	glmnet(trainx,trainy,family="binomial",nlambda =1000)	glmnet
MHNN	**Multiple-hidden-layer trained with backpropagation**	nn.train(trainx,trainy,hidden=c(10),activationfun = "sigm", numepochs=2000,batchsize=1500)	deepnet
DeepNet	**Deep Neural Network**	sae.dnn.train(trainx,trainy,hidden = c(5,5), activationfun = "sigm",numepochs = 2000, batchsize=1500)	deepnet

Table 3.4: State-of-the-art data mining algorithms used to build the classifier for RR*ab*.

using all the features available are summarized in Table 3.5. It is clear from this table that when using all the features we defined, the AdaBoost and SBoost classifiers achieve best performance. It is interesting to note that the performance of the AdaBoost and SBoost classifiers is significantly better than that of random forests, which has been the classifier of choice in the recent literature (e.g., Dubath et al. [26], Richards et al. [57]). While AdaBoost and SBoost are fairly equivalent within the uncertainties, we chose Ada.M1 as our final classifier.

3.4 Optimization of Classification Algorithms

An important step in the selection of the classification algorithms is the tuning (hyper-parameters optimization) of each candidate algorithm. In what follows, the optimization process is described for each classifier implemented,

- Random Forest: The **randomForest** algorithm was used with parameter `ntree=500` (number of trees) and **mtry=20** (number of variables at each tree). Our final classifier includes only 12 out of the 68 original features; for the final classifier the

Table 3.5: Cross-validation performance of classifiers on the templates+B293+B294+B295 training set, using all features

Algorithm	AUC	P	R	F_1 (σ_{F_1})
Logistic	0.9756	0.7869	0.8579	0.8121 (±0.0198)
CART	0.9265	0.8591	0.7373	0.7911 (±0.0177)
RF	0.9811	0.9515	0.8234	0.8804 (±0.0105)
SBoost	0.9939	0.9522	0.9094	0.9298 (±0.0054)
Ada.M1	0.9937	0.9685	0.8974	0.9311 (±0.0046)
SVM	0.9792	0.9036	0.7960	0.8456 (±0.0120)
Lasso	0.9849	0.8599	0.8398	0.8454 (±0.0139)
MHNN	0.9851	0.9190	0.8793	0.8968 (±0.0116)
DeepNet	0.9823	0.9143	0.8762	0.8941 (±0.0102)

parameter **mtry** was set to 3.

To set these parameter values, we assessed the performance of the classifier on a grid of values. The parameter **mtry** has been tested using values in the interval **[1,p]**, where p is the number of features. In Figure 3.6 a) is shown the performance of the Random Forest algorithm under different values of **mtry** using all the features. Note that the Random Forest reach the highest values of the F_1 index in the interval **[20,50]**. In addition, in Figure 3.6 b) is shown the tuning of the same parameter using only the 12 selected features (further explained in Section 3.4.2). In this case the highest values of the F_1 is reached in the interval **[3,4]**. Furthermore, when 12 features have been used, the performance of Random Forest improves significantly, because this algorithm is very sensitive to the quality of the data.

- Stochastic Boosting: was implemented with parameters **loss= "exponential"** and parameter **type="discrete"**. In addition, the "ada" function of R also allows to use a logistic loss function and real and gentle type of boosting. All combinations of loss functions and type of boosting were evaluated, and the `max.iter` parameter was assessed in the interval [100, 1000].

- AdaBoost: is implemented with the **Breiman** weight-updating coefficient and **mfinal=500**. The learning coefficient of the boosting function take different options for weighting the "weak" classifiers based on the misclassification rate. If **coeflearn**="Zhu" implies that the AdaBoost SAMME algorithm was used. Both algorithms have the parameter `boos` set to `TRUE`, and therefore, a bootstrap sample of the training set was drawn using the weights for each observation on that iteration.

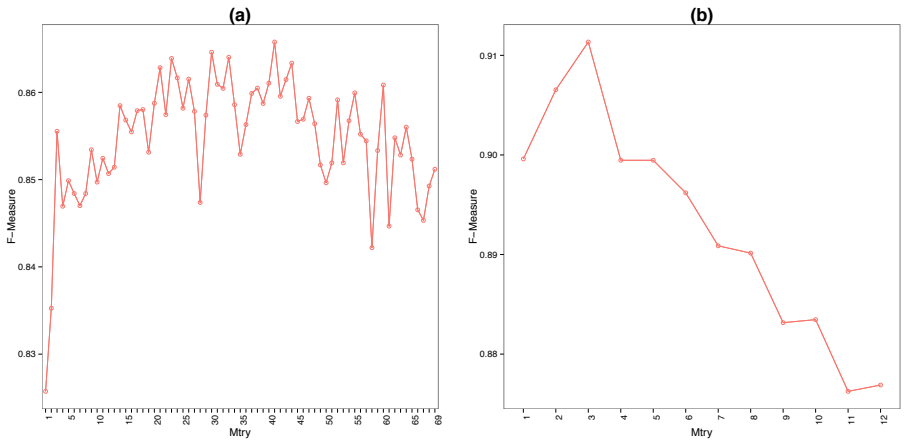


Figure 3.6: Optimization of the number of variables in each tree (mtry parameter) used in Random Forest. In Figure a) is the F-Measure (y-axis) computed for values of the mtry parameter (x-axis) when all the features are used in the classifier. In Figure b) is the F-Measure (y-axis) computed for values of the mtry parameter (x-axis) when only the 12 selected features are used in the classifier.

- Support Vector Machine : is implemented with a **polynomial** kernel and parameters **degree=2** and **nu=0.1.** The "svm" function of R also allows to use the radial basis, lineal, polynomial, and sigmoid kernels. All of them were assessed and we found that the best performing was the polynomial kernel.

- LASSO: The implementation was made with options **family="Binomial"** and **nlambda=1000.** The latter was chosen after testing performance in the range [100,10000]. The parameter α, the elastic net mixing parameter, was tested in the range [0,1] and set to 1 (giving thus a LASSO, $\alpha = 0$ corresponds to ridge regression).

- Multiple hidden neural networks: is implemented with parameters **hidden=10**, **activationfun="sigm"** (a sigmoid activation function), **batchsize=1500** and **numepochs=2000**. It is known that neural networks are sensitive to the number of hidden layers and the batch size for stochastic gradient descent. The number of hidden layers was tested in the interval [1, 20] (see Figure 3.7 a)). The neural network improves its performance when the number of hidden layers is greater than 3. In addition, as can be seen in Figure 3.7 b), the batch size was assessed in the interval [100, 2000] reaching the highest values of F_1 when the batch size is greater than 1000, after this the performance remains stable.

- Deep neural network: is implemented with a sigmoid activation function and ten hidden layers. The remaining parameters are set as **batchsize=1500**, **numepochs=2000**.

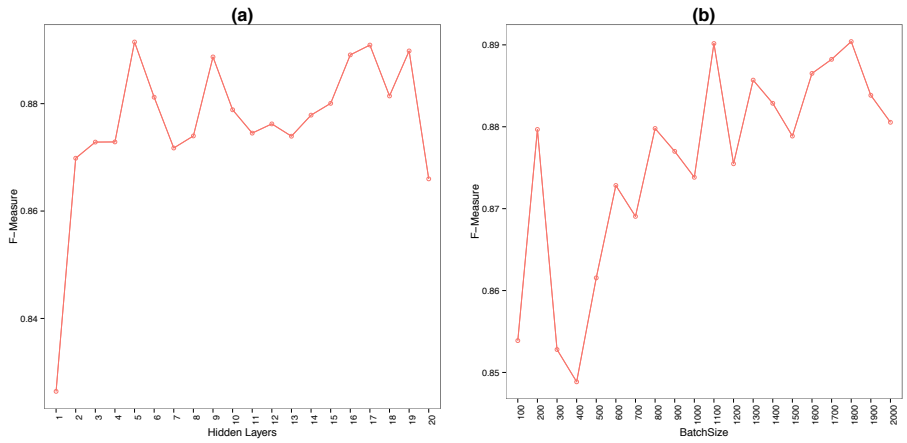


Figure 3.7: Optimization of parameters of Multi Hidden Neural Network. On the left figure is the F-Measure (y-axis) computed for number of Hidden Layers used in the Neural Network (x-axis). On the right figure is the F-Measure (y-axis) computed for the Batch Size used in the Neural Network (x-axis).

> The classifier performance was assessed, testing the parameters **batchsize** in the interval [100,2000] and the number of hidden layers in [1, 20].

Following the optimization of the distinctive parameters of each algorithm, the number of iterations of each of them was assessed. Consequently, the parameters **ntree** of Random Forest, **max.iter** of Stochastic Boosting, **mfinal** of Adaboost.M1 and **numepochs** of Multiple hidden neural networks has been tested in the interval [100,1000]. In Figure 3.8 can be observed that the number of iterations does not vary significantly the performance measure for any of the algorithms assessed. This result has been evaluated in three different scenarios (Figure 3.8 a), b) and c)), depending on the strategy of aperture selection and the feature set used (more details below). In all the figures can be observed that the boosting algorithms (i.e., Stochastic Boosting and Adaboost) consistently get better performance than its two most important competitors (Random Forest and Neuronal Networks). This result is verified even when the Random Forest and Neuronal Network reach its best performance, which happens when only the most important features are used.

3.4.1 Choice of aperture for photometry

After we selected the best classifier, we assessed whether the selection of an aperture from which we estimated the features of the light curves affected the classification performance. We implemented three strategies to select the aperture. The first was to fix

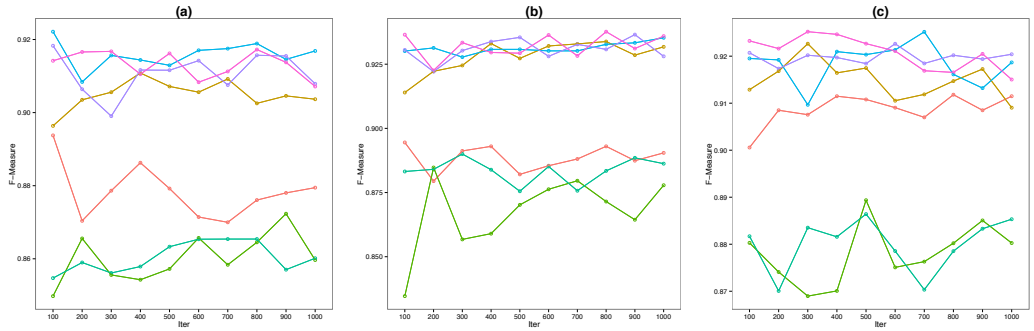


Figure 3.8: Optimization of the number of iterations used in the data mining algorithms implemented this work. The red line is the F-Measure computed for Random Forest, the yellow line corresponds to Stochastic Boosting, the green line is for the Deep Neural Network, the light blue line is for the multi hidden neural network. Finally, the blue, violet and pink lines corresponds to the AdaBoost algorithm with **coeflearn** Breiman, Freund and Zhu respectively. In figure a) the 12 selected features are used and the `minError` strategy of aperture selection in each classifier. In figure b) all the features are used and the `KDC` strategy of aperture selection. In figure c) the 12 selected features are used and the `KDC` strategy of aperture selection.

the aperture size to be equal for all variable stars. We call this strategy `fixAper(i)` for aperture size i, where $i = 1, \ldots, 5$ (this gives us five strategies).

Second, we chose for each light curve the aperture size that achieved the minimum sum of squared errors and called this strategy `minError`. The proportion of light curves assigned under this criterion in each aperture is shown in Figure 3.9.

Third, similarly to Richards et al. [57], we developed a strategy to select the best photometry aperture for each light curve based on a Kernel Density Classifier (KDC). We call this strategy `KDC`. In this method, we select the aperture according to the follow steps:

1. Compute Kernel Density for each aperture. The kernel density for an aperture f_i, with $i = 1, \ldots, 5$ was estimated using the mean magnitudes whose minimal sum of squared measurement errors was achieved at aperture i. Note that the proportion of light curves used in each density estimation comes from the strategy `minError` Figure 3.9. The estimated densities for each aperture are shown in Figure 3.10.

2. Evaluate the median of the magnitudes for a specific light curve x_m in each density.

3. Compute the following vector of probabilities,

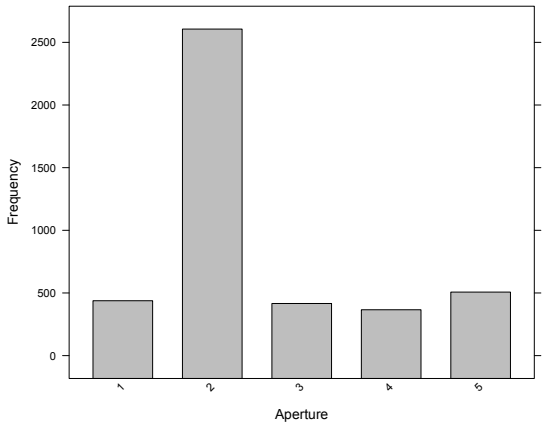


Figure 3.9: Number of light curves with the minimum sum of squared errors at each aperture size.

$$p_i = \frac{\pi_i f_i(x_m)}{\sum_{j=1}^{5} \pi_j f_j(x_m)}$$

where $i = 1, \ldots, 5$ and p_i is the probability to belongs of each density given by the median of magnitudes x_m.

4. For each light curve we select the aperture with the highest p_i.

The main difference with the method proposed by Richards et al. [57] is that we developed a soft thresholding classifier to choose an optimal aperture.

All these seven strategies were evaluated using the boosting classifier algorithms, which were selected previously as the best classification methods. The results are shown in Table 3.6. We used cross-validation on variable stars from the B295 field to estimate the performance of the boosting classifiers under different strategies. The best performance was reached by the strategies `KDC` and `fixAper(2)`. The `KDC` strategy chooses in most cases the aperture 2, therefore it is natural that the `KDC` have a similar performance than the `fixAper(2)`. In addition, note that `KDC` is a better strategy than the `minError`, which is the most commonly used method to choose an aperture. This result is explained by the fact that the strategy `minError` selects the apertures 3, 4 and 5 more frequently than the `KDC` strategy. As can be seen in Table 3.6 the classifier based on any of these apertures has a poor performance.

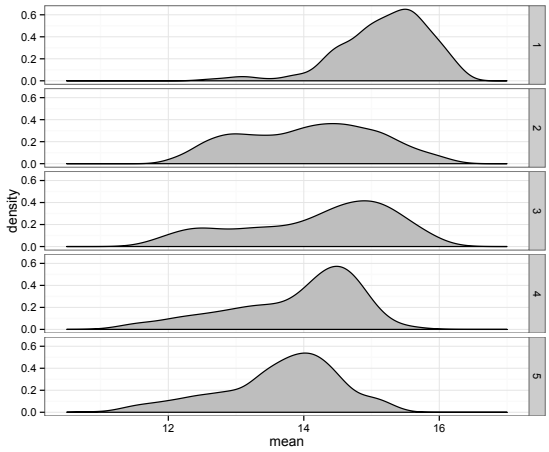


Figure 3.10: Kernel density estimates of the mean magnitude of curves with the minimum sum of squared errors at each aperture size.

Table 3.6: F_1 Measure by Aperture and Classifier Algorithm

Strategy	SBoost_1	SBoost_2	Ada.M1	Ada.SAMME
fixAper(1)	0.9210 (0.0189)	0.9244 (0.0176)	0.9120 (0.0165)	0.9217 (0.0175)
fixAper(2)	0.9359 (0.0090)	0.9363 (0.0080)	0.9308 (0.0099)	0.9303 (0.0092)
fixAper(3)	0.8794 (0.0197)	0.8728 (0.0198)	0.8799 (0.0271)	0.8757 (0.0184)
fixAper(4)	0.8621 (0.0181)	0.8651 (0.0206)	0.8615 (0.0207)	0.8631 (0.0218)
fixAper(5)	0.8055 (0.0249)	0.8019 (0.0278)	0.8146 (0.0194)	0.8166 (0.0251)
KDC	0.9316 (0.0090)	0.9327 (0.0093)	0.9308 (0.0099)	0.9255 (0.0089)
minError	0.9133 (0.0139)	0.9140 (0.0136)	0.9140 (0.0146)	0.9131 (0.0159)

Table 3.7: Cross-validation performance of classifiers on the templates+B293+B294+B295 training set, using the best 12 features

Algorithm	AUC	P	R	F_1 (σ_{F_1})
Logistic	0.8574	0.4624	0.8057	0.4855 (±0.0686)
CART	0.9311	0.8577	0.7342	0.7860 (±0.0141)
RF	0.9902	0.9522	0.8896	0.9194 (±0.0067)
SBoost	0.9942	0.9483	0.9184	0.9326 (±0.0050)
Ada.M1	0.9937	0.9526	0.9154	0.9331 (±0.0055)
SVM	0.9840	0.9073	0.8321	0.8651 (±0.0090)
Lasso	0.9553	0.7618	0.6691	0.6953 (±0.0145)
MHNN	0.9824	0.9365	0.8608	0.8956 (±0.0090)
DeepNet	0.9823	0.9258	0.8775	0.9001 (±0.0079)

3.4.2 Feature selection

Previously we detailed 68 features (Tables 3.1 and 3.2) that were used to find the best classifier. It is clear that not all features have the same effect on the classification. Several data mining algorithms allow measuring the importance of the variables using some criteria. For example, the Adaboost.M1 algorithm uses the gain of the Gini index given by a variable in a tree and the weight of the tree and we can measure how important each one of them is for the classification. In Fig. 3.11 we show the features ordered by importance, with the most important at the top.

As expected, the most important feature is the dominant frequency (f_1). This result is explained since the RR*ab* have periods defined in a known range, between 0.2 and 1 days approx (see section 2.1.2). Consequently, if a RR*ab* variable star has an estimated period out of this range, the classifier will probably label it as "No RR*ab*". Therefore, it is essential to estimate the period of variable stars correctly to obtain an accurate classifier for the RR*ab* stars.

Note in Figure 3.11 that the next three most important features are the ones related to the harmonic fit (i.e., the periodical features). The first non-periodic feature is `p2p_scatter_2praw`, which is related to the noise of the signal. In order to obtain the best performance of the classifier according to the F_1 measure, we decided to use 12 features. This means that the additional features do not add significant information.

We again assess the performance of the data mining algorithms, but now using only the 12 more important features. The cross-validation estimation of the performance measures is summarized in Table 3.7. Note that the performance of the boosting classifiers (Ada.M1 and SBoost) do not vary significantly when using all the features, which shows that the Adaboost.M1 classifier is more stable regarding to the features selected than other

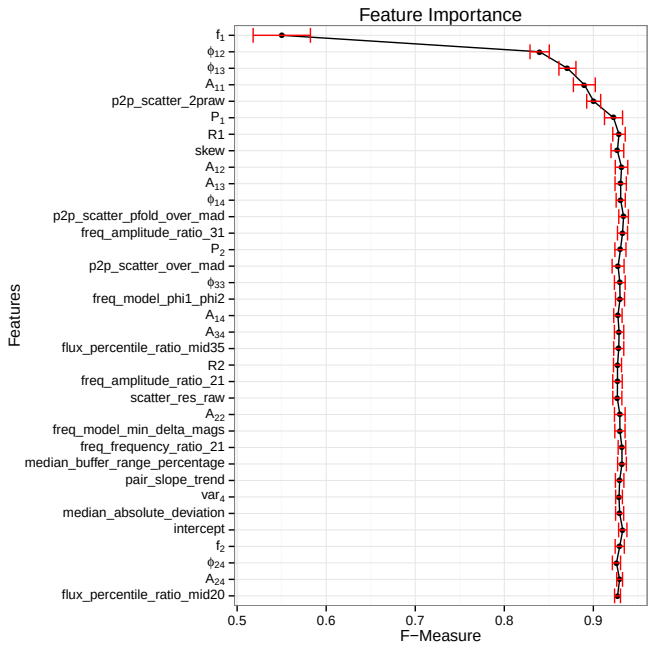


Figure 3.11: Feature importance using the Ada.M1 classifier. Based on this graph, we chose to consider only the 12 most important features in the final classifier.

classification algorithms. On the contrary, the Random Forest algorithm improves its performance significantly after the feature selection, which proves that this algorithm is very sensitive to quality of the features used, since in each step of the Random Forest a set of features is chosen. Despite the increasing performance, the Random Forest has still lower values at the performance measures compared with the Adaboost.M1.

3.4.3 Sensitivity to training set choice

An important step in building a classifier is to select an appropriate training set that captures the variability of the data because when the training set is not representative, the resulting classifier is bound to fail for some types of objects. To test our sensitivity to the training set choice, we trained our classifier using different training sets by taking different subsets out of the four available sets (templates, B293, B294, and B295, see Sect. 3.3.3). In Table 3.8 we show the results of this exercise using the Adaboost.M1 classifier, showing the F_1 measure for different combinations of training and test sets; we note that in this case we measured the performance with a test set that was disjoint from the training. The first row shows the performance for the classifier trained only on templates (T), the following three rows shows the performance for the classifier trained using templates plus B295, B294, or B293 respectively. The next three rows are similar to the previous three,

but now two complete fields are incorporated into the training set. The last two rows show the performance of the classifier with all of the curves from templates plus 90% (80%) of the curves from the three fields. As it is evident from Table 3.8, the performance is best when we include curves from templates and all three fields. It does not vary significantly between having 80% or 90% of the curves over the expected random variations in the F_1 performance, which for Adaboost.M1 was expected to be on the order 1%. We conclude that our choice of training set of templates+ 80% B293 + 80% B294 + 80% B295 does not bias our results in a significant way, as assessed by training the classifier.

Training \Test	B295	B294	B293
Templates	0.8713	0.8905	0.9065
T+B295	-	0.9095	0.9251
T+B294	0.9043	-	0.9270
T+B293	0.9003	0.9150	-
All \B294	-	0.9204	-
All \B293	-	-	0.9290
All \B295	0.9122	-	-
All 90%	0.9267	0.9476	0.9502
All 80%	0.9269	0.9536	0.9304

Table 3.8: F-measure by training set (Adaboost.M1)

3.4.4 Final classifier

After performing all the steps in the classification procedure described above (see Figure 3.5) successfully, we obtain our final classifier. Accordingly, with the results obtained, the final classifier is built using a training set composed by the templates+ 80% B293 + 80% B294 + 80% B295, with a KDC criteria to select the aperture of each light curve, and with the Ada.M1 classifier using the following 12 features of the 68 listed in Tables 3.1 and 3.2 (ordered by its importance).

1. f_1
2. ϕ_{12}
3. ϕ_{13}
4. A_{11}
5. `p2p_scatter_2praw`
6. P_1
7. `R1`

8. skew
9. A_{12}
10. A_{13}
11. ϕ_{14}
12. p2p_scatter_pfold_over_mad

The final classifier has a F_1 measure estimated from cross-validation $\approx$ 0.93. As mentioned above, until now it has never been implemented a classifier for RR*ab* variable stars in the NIR. However, in the optical several classifiers have been implemented. For example, for the ASAS survey, Richards et al. [57] implemented a classifier with a F_1 performance of $\approx$ 0.96 for RR*ab* classification (see their Fig. 5). This means that the expected number of false positives or negatives is about half of what we achieve with our classifier. A better performance in the optical is expected given the larger amplitudes and more asymmetric shape in those bands (see Figure 3.3.1). Therefore, the performance in the optical should be taken as an upper bound of what supervised classification could achieve for the VVV data. In addition, the data is not directly comparable, because the ASAS data have a larger number of epochs (mean 541), whereas the VVV has in most cases close to 100 epochs. Therefore, we are satisfied with the performance achieved, since is very close to the performance obtained in the optical for RR*ab* variable stars.

3.5 Performance on independent datasets

In the procedure to find the final classifier, we measured the performance of several classifiers, using different training set, aperture selection criteria and features. Although the performance was always evaluated using ten-fold cross validation, all the data used in the training set comes from the center of the bulge area of the VVV. In this section, we assess the performance of our classifier in light curves observed in other areas of the VVV, which have been studied by astronomers. These are, (1) a catalog of RR*ab* in the Galactic globular clusters 2MASS-GC 02 and Terzan 10 [2], (2) a catalog of RR*ab* in the outer bulge area of the VVV [37] and (3) a census of RR*ab* stars along the southern galactic disk [25].

This analysis is particularly relevant, since it allows us to evaluate the performance of the classifier on different datasets in which flux measurements do not necessarily follow the same conditions in cadence, depth, etc. as our training set. The final classifier has a F_1 measure of $\approx$ 93% obtained by cross-validation using a score threshold of 0.548, so that if this performance generalizes well, we would expect the harmonic mean of the number of false positives and false negatives to be on the order of 7%.

3.5.1 RR*ab* in 2MASS-GC 02 and Terzan 10

Alonso-García et al. [2] identified, by human inspection of the light curves and color information, 39 RR*ab* variables stars in the globular clusters 2MASS-GC 02 and Terzan 10. Terzan 10 also has been covered by OGLE-IV [61], which allows that the authors at Alonso-García et al. [2] match their results with optical light curves from OGLE. Therefore, the stars labeled as RR*ab* by Alonso-García et al. [2] were classified with great certainty. Our machine-learned classifier is at a disadvantage, since we only had the VVV light curves as input for the classification and not color information.

To assess the performance of the classifier in the light curves of the globular clusters 2MASS-GC 02 and Terzan 10, we compare the scores obtained by the final classifier and the RR*ab* classified by Alonso-García et al. [2] (true positives). In Figure 3.12 the distribution of the scores for true positives, false negatives and false positives are shown. Most of the known RR*ab* are classified correctly by the classifier. Only six light curves, or $\approx 15\%$ of the sample, are false negatives. The periods of these false negatives are consistent with those of RR*ab*, and because they are not symmetrical, they were classified as RR*ab* by Alonso-García et al. [2].

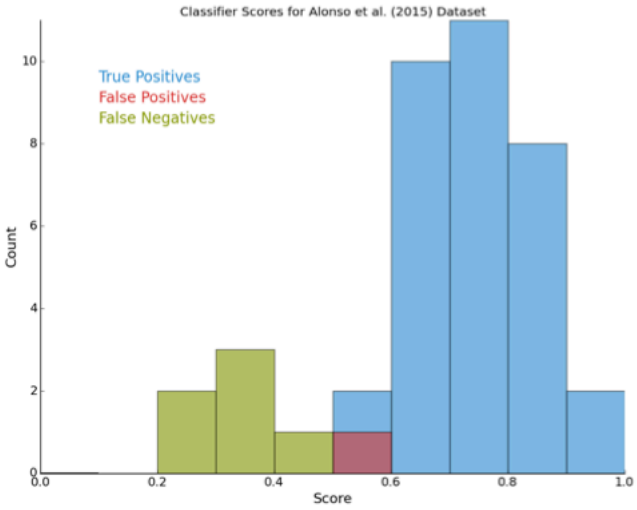


Figure 3.12: Histogram of scores obtained by the classifier for the light curves of the sample presented by Alonso-García et al. [2]. Shown are the true positives (sources classified by Alonso-García et al. [2] as RR*ab*), false positives, and false negatives.

Using a score threshold of 0.548 we found two false positives, or $\approx 5\%$ of the sample. Both light curves are shown in Fig. 3.13. We discuss each in turn. Terzan10_V113, shown in panel (a), was classified as an eclipsing binary in Alonso-García et al. [2] because of its

very symmetric behavior. It is not classified as an RR*ab* by OGLE either, reinforcing its status as a non-RR*ab*. As in the NIR, some RR*ab* have a more symmetrical behavior (see Figure 3.3.1), it is not surprising that variables stars with correct periods and amplitudes are classified as RR*ab* even if they are very symmetric.

One additional variable star was classified as RR*ab*, but was not present as a variable in the Alonso-García et al. [2] catalog. Its internal IDs is 21089, and it is shown in panel (b) of Fig. 3.13. This variable star (273508) was labeled as RR*ab* by OGLE (OGLE_BLG_RRLYR-33508), but that was inadvertently left out in Alonso-García et al. [2]. Probably this light curve is part of a VVV field and not of the cluster Terzan 10, because it is beyond the tidal radius of Terzan 10 and it is also too bright to be part of the cluster.

After studying the false positives in detail, we conclude that one is probably a RR*ab* variable stars and consequently the classifier has only one false positive. Overall, the harmonic mean of false positives and false negatives is 1.71 or $\approx$ 4.4% of the data, better than the performance measures obtained in the training and even better than expected.

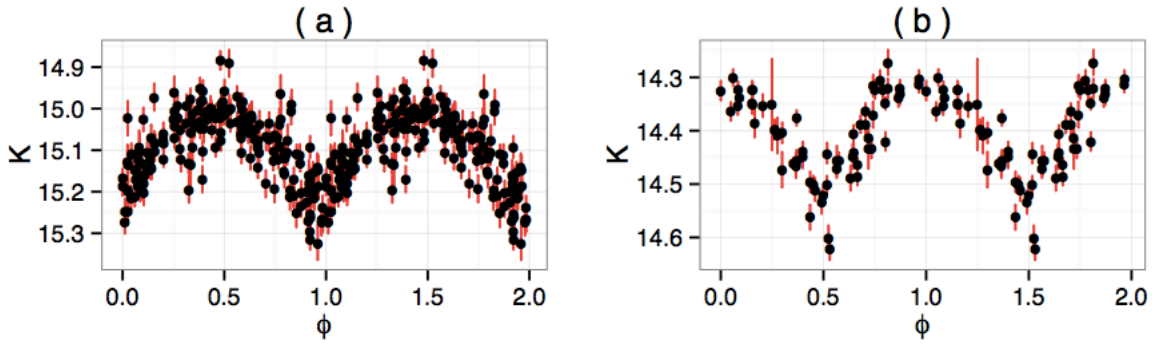


Figure 3.13: Two sources that were nominally false positives: (a) Terzan10_V113; (b) internal identifier 273508. One of them (a) is a bona fide false positive, while the other (b) is a true positive that was not flagged as such in the work of Alonso-García et al. [2] (see text).

3.5.2 RR*ab* in the outer bulge area of the VVV

Gran, F. et al. [37] performed a human classification of RR*ab* variable stars in 7869 light curves previously classified as variable in the outer bulge region of the VVV survey. This region corresponds to the VVV fields b201 through b228 (see Figure 2.3). In order to

assess the performance of our classifier in this dataset, we compare the scores obtained by the classifier with the human classification. There were 1019 light curves classified as RR*ab* by Gran et al., of which 939 passed the cleaning process detailed in Sect. 3.1.

Although the data of the outer bulge area were used to verify the performance of the classifier on independent datasets, the classifier also helped to confirm some RR*ab* that were not classified with certainty in the visual inspection. Among the 1019 RR*ab* found by Gran et al. (2016), the classifier helped to detect the more symmetrical RR*ab* (In Figure 3.14 some examples are shown) where the visual inspection suggested to eliminate as many of them as possible since they could be confused with eclipsing binaries and contaminate the sample. Other important contribution was that our classifier found the RR*ab* stars of high magnitude (more than $K_s \sim 15$ approx.). This result allowed to rule out many false positives selected by analyzing the χ^2 test (Carpenter, 2001 [15]).

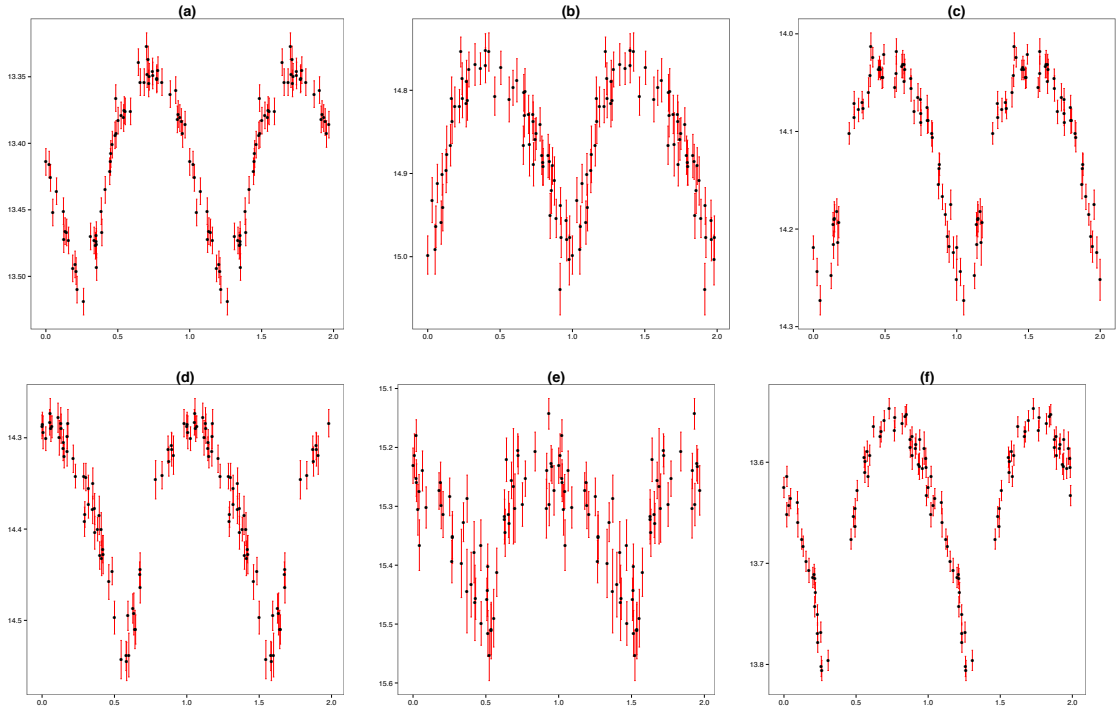


Figure 3.14: Light curves of RR*ab* stars found by Gran et al. (2016) in the outer bulge area of the VVV which were confirmed by the classifier.

After labeling the RR*ab* stars confirmed by the classifier, we could assess the performance obtained by the classifier in the outer bulge data. In the 939 variable stars which

passed the cleaning filters, there were 50 false negatives, and the classifier gave an additional 177 false positives, which gives a total of 1066 stars that were detected as RR*ab* by the classifier. The distributions of the scores for the outer bulge light curves is shown in Fig. 3.15. The harmonic mean of the number of false positives and negatives is ≈78, or ≈ 8% of the sample size, slightly lower but fully consistent with the F_1 measure estimated from cross-validation on the training set.

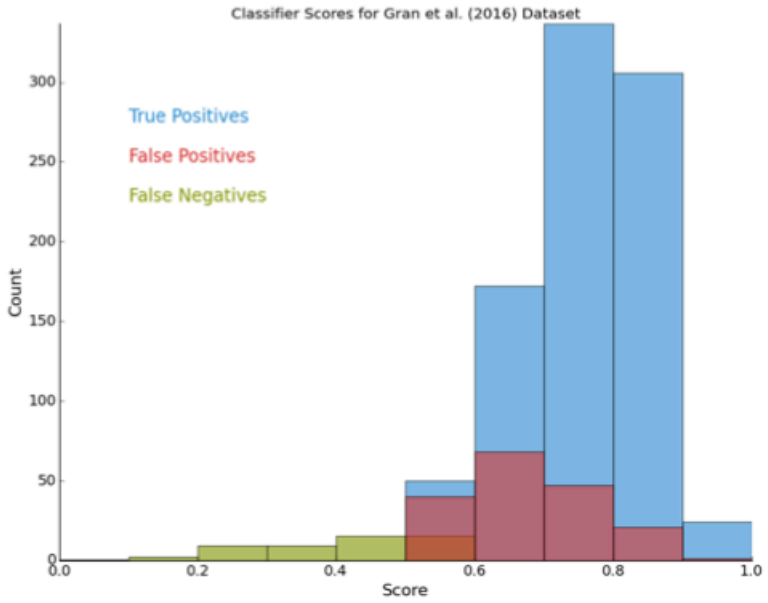


Figure 3.15: Histogram of scores obtained by the classifier for the outer bulge light-curves of the sample used by Gran et al. (2016). Shown are the true positives (sources classified by as RR*ab*), false positives, and false negatives.

3.5.3 Census of RR*ab* stars along the southern galactic disk.

Dekany et al. (2018) [25] performed a census of RR*ab* along the southern galactic disk. The southern galactic disk corresponds to the fields b001-b151 of the VVV survey (see Figure 2.3). It is particularly interesting to assess our classifier in this data, because the light curves from the disk have less observations than the light curves from the bulge area. Therefore, the curves of both data sets differ in their temporal distribution. Furthermore, as there are currently few known RR*ab* in the disk, the classifier was used to discover new RR*ab* stars in this area. Due to the small number of known RR*ab* stars in the disk, the performance of the classifier should be assessed from visual inspections of human experts.

The visual inspection was performed by selecting all the variable stars with a score estimated by the classifier above the 0.5. As a result, they obtained a selection of 3379 RR*ab* candidates. These objects were labeled as RR*ab* or not by a group of human experts. Based in this inspection, Dekany et al. (2018) [25] concluded that 90% of the candidates are consistent with being RR*ab*.

Later, the threshold of the classifier was calibrated to improve the performance on this data. In Figure 3.16 the Kernel Density of the scores of RR*ab* and No RR*ab* are shown. Note that using a score threshold of 0.7 we can obtain a pure sample of RR*ab* candidates. Dekany et al. (2018) decided to use a score threshold of 0.6 with a human-estimated precision of approx 0.9. Using this threshold the classifier found 2147 RR*ab* candidates.

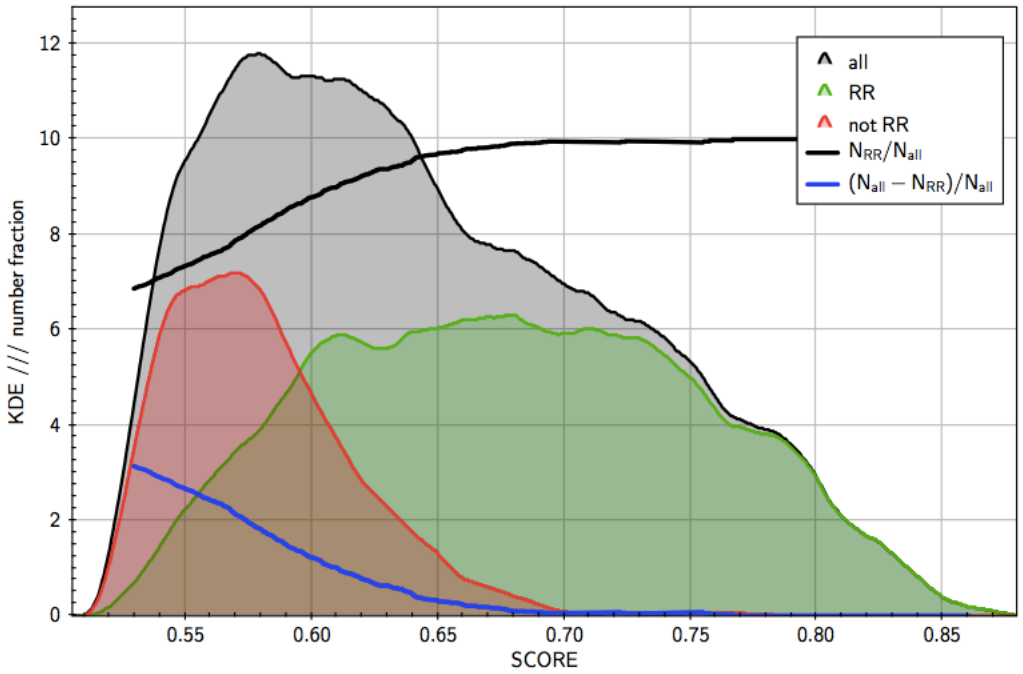


Figure 3.16: Kernel Density estimation of the classification score for RR*ab* (green density) and No RR*ab* (red density) and overall (green density). The blue and black lines correspond to the contamination and precision respectively (from Dekany et al, 2018 [25]).

As can be seen, the classifier was used in several datasets of the VVV survey. Each of them have different characteristics than the dataset used to train the classifier. Particularly, the center of the bulge area was used to train the classifier. This area is distant of the southern galactic disk or the outer bulge area, where the classifier was applied. However,

the results obtained by the RR*ab* classifier were consistent with the performance of the classifier estimated by cross-validation in the training set. The satisfactory results of the classifier allow us to continue looking RR*ab* stars over the full area of the VVV.

Chapter 4

Light Curves Modeling

In the previous sections, I mentioned that the light curves of astronomical objects are measured irregularly in time. Therefore, there are few methods to model the time dependency of the light curves. Some of these methods transform the irregular time series into a regular time series, using interpolation techniques (for a review of such methods see e.g. Rehfeld et al. (2011) [55]).

To model directly the irregular time series, generally the CARMA family of models are used. A particular case of the CARMA models is the CAR(1) model. This model can be used to estimate the autocorrelation of an irregular time series. However, as mentioned in section 2.2.1.4, the CAR(1) models assumes Gaussian distribution and continuous white noise. These constraints provide to the CAR(1) models with little flexibility. For instance, the CAR(1) model is defined by a continuous time white noise, which exists only in the sense that its integral is a continuous time random walk. In addition, the assumption of a Gaussian distribution could be a limitation to model irregular time series with asymmetrical or heavy tailed distributions.

In this section, two new models that fit unequally spaced time series are introduced. The main task of these models is to offer an alternative solution with more flexibility regarding the distribution assumption. This can be achieved using a discrete representation of the continuous time processes and relaxing the distribution assumption. We called these models, the Irregular Autoregressive (IAR) model [30, 31] and the Complex Irregular Autoregressive (CIAR) model. In the following we present both models and its properties.

4.1 Irregular Autoregressive (IAR) model

Let $\{y_t\}$ be an observation measured at irregular times, and therefore $\{y_t\} = \{y_{t_j}\}$, where $\{t_j\}$ is the increasing sequence of observation times with $j = 1, \ldots, n$. The first-order irregular

autoregressive (IAR) process is defined by,

$$y_{t_j} = \phi^{t_j - t_{j-1}} y_{t_{j-1}} + \sigma_y \sqrt{1 - \phi^{2(t_j - t_{j-1})}}\, \varepsilon_{t_j}, \tag{4.1.1}$$

where ε_{t_j} is a white noise sequence with zero mean and unit variance. Note that in a time series measured irregularly, $t_j - t_{j-1}$ is not constant. The initial value of the process is $y_{t_1} = \sigma_y \varepsilon_{t_1}$. Figure 4.1 shows a simulated irregular autoregressive process. Note that the time gaps are not constant in the simulated process.

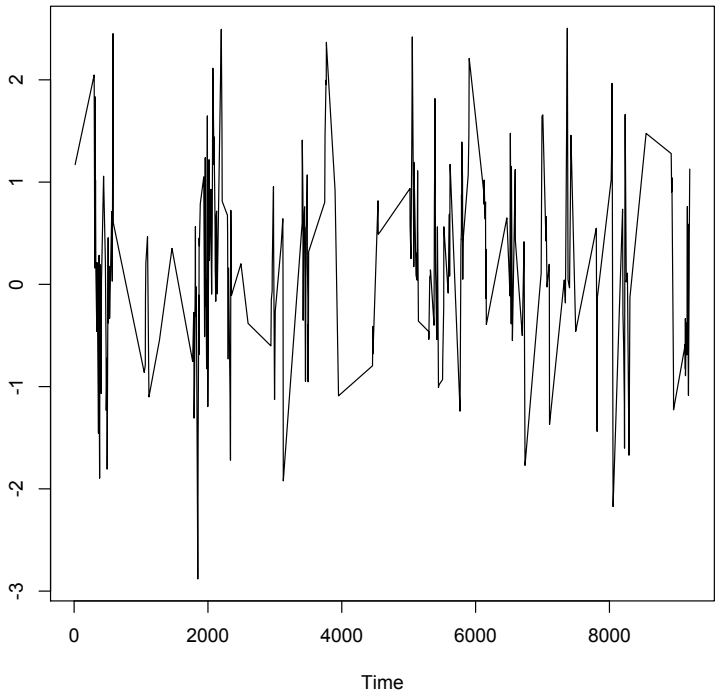


Figure 4.1: Simulated IAR Time Series of length 300 and $\phi = 0.9$.

The IAR model is an extension of the regular autoregressive model in the sense that if $t_j - t_{j-1} = 1$ is assumed, the IAR process becomes the autoregressive model of order 1 (AR(1)) for regularly space data.

Furthermore, both the IAR and CAR(1) process are strongly connected. In equation (2.2.19) we defined the CAR(1) process by,

$$X(t) - \frac{\beta}{\alpha_0} = e^{-\alpha_0(t-s)} \left(X(s) - \frac{\beta}{\alpha_0} \right) + e^{-\alpha_0 t} (I(t) - I(s)) \tag{4.1.2}$$

Setting $\beta = 0$, $\sigma^2 = \frac{\sigma_0^2}{2\alpha_0}$ and replacing $e^{-\alpha_0}$ with ϕ the equation (4.1.2) is equivalently to the equation (4.1.1). Furthermore, if y_{t_j} is described by the equation (4.1.1) then,

$$\begin{aligned} \mathbb{E}(y_{t_j}) &= 0, \\ \mathbb{V}(y_{t_j}) &= \sigma^2, \\ \mathbb{E}(y_{t_k}\, y_{t_j}) &= \sigma^2\, \phi^{t_k - t_j} \end{aligned} \tag{4.1.3}$$

where $k \geq j$. Consequently, the autocovariance and autocorrelation function between two observational times t, s are defined by,

$$\begin{aligned} \gamma(t-s) &= E(y_t\, y_s) = \sigma^2\, \phi^{t-s}, \\ \rho(t-s) &= \frac{\gamma(t-s)}{\gamma(0)} = \phi^{t-s}. \end{aligned} \tag{4.1.4}$$

Furthermore, it can be proved that the first two moments of CAR and IAR processes are the same (see Appendix A), so under gaussianity, both processes are equivalent. In other words, it can be considered that, the IAR process is a discrete representation of the CAR (1) process, where the time dependency is represented in the IAR model by the parameter ϕ and in the CAR(1) model with $e^{-\alpha_0}$.

The results in (4.1.3) prove that the sequence $\{y_{t_j}\}$ corresponds to a second-order or weakly stationary process. Furthermore, under some conditions, the IAR process is strictly stationary and ergodic. This result has been stated in the Theorem 1 of Eyheramendy et al. (2017) [31], which is presented below

Theorem 4.1.1. *Consider the process defined by* (4.1.1) *and assume that the input noise is an i.i.d. sequence of random variables with zero mean and unit variance. Furthermore, suppose that* $t_j - t_{j-n} \geq C \log n$ *as* $n \to \infty$, $\phi^2 < 1$ *such* C *is a positive constant such that* $C \log \phi^2 < -1$. *Then, there exists a solution to the process defined by* (4.1.1)*, and the sequence* $\{y_{t_j}\}$ *is strictly stationary and ergodic.*

Proof: For a given positive integer n we can write

$$y_{t_j} = \phi^{t_j - t_{j-n}}\, y_{t_{j-n}} + \sigma \sum_{k=0}^{n-1} \phi^{t_j - t_{j-k}} \sqrt{1 - \phi^{2(t_{j-k} - t_{j-k-1})}}\, \varepsilon_{t_{j-k}},$$

Notice that under the assumptions of the theorem the first term converges to zero in probability. On the other hand, we have that

$$\phi^{2(t_j - t_{j-k})} \leq k^{\alpha},$$

where

$$\alpha = C \log \phi^2$$

Consequently,

$$\sum_{k=0}^{\infty} \phi^{2(t_j - t_{j-k})} \leq \sum_{k=0}^{\infty} k^{\alpha} < \infty,$$

since $\alpha < -1$ by assumption. Thus, the expression

$$y_{t_j} = \sigma \sum_{k=0}^{\infty} \phi^{t_j - t_{j-k}} \sqrt{1 - \phi^{2(t_{j-k} - t_{j-k-1})}}\, \varepsilon_{t_{j-k}}, \tag{4.1.5}$$

corresponds to a measurable transformation of the i.i.d. sequence $\{\varepsilon_{t_j}\}$. Therefore, due to Theorem 1.7 of Palma (2007) [52], the sequence $\{y_{t_j}\}$ is strictly stationary and ergodic.

Further, it is straightforward to see that the equation (4.1.5) is a solution to the process defined by (4.1.1). This can be shown by plugging-in $y_{t_{j-1}}$, as defined in (4.1.5), into the right-side of equation (4.1.1). After some arithmetic one gets to y_{t_j}, showing that (4.1.5) is indeed a solution to the process defined by (4.1.1).

Note that for a process measured in regular times the assumption of the theorem is satisfied since $t_j - t_{j-n} = n$ and $n > \log(n)$ is achieved. In addition, for a regular AR(1) model $\phi^2 < 1$ is also satisfied. Therefore, the regular AR(1) is also ergodic and stationary.

4.1.1 Estimation of IAR Model

The finite past predictor of the process at time t_j is given by,

$$\widehat{y}_{t_j} = \phi^{t_j - t_{j-1}}\, y_{t_{j-1}}, \text{ for } j = 2, \ldots, n. \tag{4.1.6}$$

where the initial value is $\widehat{y}_{t_1} = 0$. Furthermore, $e_{t_j} = y_{t_j} - \widehat{y}_{t_j}$ is the innovation with variance,

$$v_{t_j} = Var(e_{t_j}) = \sigma_y^2[1 - \phi^{2(t_j - t_{j-1})}] \tag{4.1.7}$$

where the initial values are $e_{t_1} = y_{t_1}$ and $v_{t_1} = Var(e_{t_1}) = \sigma_y^2$.

The estimation of the model parameter $\theta = (\sigma_y^2, \phi)$ can be performed by maximum likelihood. Assuming a Gaussian distribution, minus the log-likelihood of this process can be written as,

$$\ell(\theta) = \frac{n}{2}\log(2\pi) + \frac{1}{2}\sum_{j=1}^{n}\log \nu_{t_j} + \frac{1}{2}\sum_{j=1}^{n}\frac{e_{t_j}^2}{\nu_{t_j}}, \tag{4.1.8}$$

We can obtain the maximum likelihood estimator of σ_y^2 by maximizing the log-likelihood (4.1.8), such that,

$$\hat{\sigma_y}^2 = \frac{1}{n}\sum_{j=1}^{n}\frac{(y_{t_j} - \widehat{y}_{t_j})^2}{\tau_{t_j}}, \text{ where } \tau_{t_j} = \nu_{t_j}/\sigma_y^2. \tag{4.1.9}$$

it is not possible to find a closed form expression for the maximum likelihood estimator of ϕ, but iterative methods can be used.

4.1.2 IAR Gamma

As I mentioned above, the IAR model is equivalent to the CAR(1) model when the data is assumed normally, but the IAR model is more general, since it allows fitting data coming from other distributions. To verify this, we perform an IAR model with Gamma conditional distribution following the procedure of writing the conditional variance as a function of the conditional mean [53]. Let $\mathcal{F}_{t_{j-1}} = \sigma(y_{t_{j-1}}, y_{t_{j-2}}, \ldots)$ the σ-field generated by the information up to instant t_{j-1}, then the conditional mean and variance of IAR model are defined by

$$\begin{aligned}\mathbb{E}(y_{t_j}|\mathcal{F}_{t_{j-1}}) &= \mu + \phi^{t_j - t_{j-1}}\, y_{t_{j-1}} \\ \mathbb{V}(y_{t_j}|\mathcal{F}_{t_{j-1}}) &= \sigma^2\,(1 - \phi^{2(t_j - t_{j-1})})\end{aligned} \tag{4.1.10}$$

Note that these conditional moments are equivalent to those of the Gaussian IAR process, the only difference is the positive parameter μ that corresponds to the expected value of $y_{t_{j-1}}$. If $y_{t_j}|y_{t_{j-1}}$ follow a Gamma distribution, a positive value of μ is required in order to ensure the positivity of the process. However, the process may be shifted, so that $y_{t_j} - \mu$ have a zero mean, like the Gaussian IAR. Under this notation, the initial value of the process y_{t_1} is μ.

In addition, note that under the assumption of stochastic times the marginal mean $\mathbb{E}(y_{t_j}) = \frac{\mu}{1-\mathbb{E}(\phi^{t_j - t_{j-1}})}$ and marginal variance $\mathbb{V}(y_{t_j}) = \sigma^2 + \frac{\mathbb{E}(y_{t_j})^2\mathbb{V}(\phi^{t_j - t_{j-1}})}{1-\mathbb{E}(\phi^{2(t_j - t_{j-1})})}$ are constants.

Now suppose that x_{t_j} follows a gamma distribution with shape α_{t_j} and scale β_{t_j}. It is well known that the expected value and the variance of this distribution are $\mathbb{E}(x_{t_j}) = \alpha_{t_j}\beta_{t_j}$ and $\mathbb{V}(x_{t_j}) = \mathbb{E}(x_{t_j})\beta_{t_j}$ respectively.

xc

Let $\mathbb{E}(y_{t_j}|\mathcal{F}_{t_{j-1}}) = \lambda_{t_j}$, note that $\mathbb{V}(y_{t_j}|\mathcal{F}_{t_{j-1}})$ can be defined as a function g of λ_{t_j} ($g(\lambda_{t_j})$). Consequently, $y_{t_j}|\mathcal{F}_{t_{j-1}} \sim Gamma\left(\alpha_{t_j}, \beta_{t_j}\right)$ with $\beta_{t_j} = \frac{g(\lambda_{t_j})}{\lambda_{t_j}}$ and $\alpha_{t_j} = \frac{\lambda_{t_j}^2}{g(\lambda_{t_j})}$. The log-likelihood are,

$$\ell_j = \log f_\theta\left(\alpha_{t_j}, \beta_{t_j}\right) = -\left(\alpha_{t_j}\right)\log\beta_{t_j} - \log\Gamma\left(\alpha_{t_j}\right) - \frac{1}{\beta_{t_j}} y_{t_j} + \left(\alpha_{t_j} - 1\right)\log y_{t_j}$$

Let $f(y_{t_1}) \sim Gamma(1, 1)$, then the full log-likelihood are,

$$\ell(\theta) = \sum_{j=2}^{N} \ell_j + \ell_1$$

where $\ell_1 = -y_{t_1}$. The unknown parameters of the model are ϕ, μ and σ which can be estimated using iterative methods. From now, this model will be called as IAR-Gamma.

4.1.3 Simulation Results

As mentioned above, the estimation of the parameters of both the IAR and IAR-Gamma is performed by maximum likelihood. In order to implement the estimation procedures of both models several functions were created in R and Python softwares (for more details, see Section 4.4).

It is very important to assess whether the functions were implemented correctly and whether the maximum likelihood estimator proposed is accurate. Consequently, in this section a Monte Carlo experiments is performed in order to assess the finite sample estimation procedure.

The Monte Carlo experiment is based on 1000 repetitions of each simulation. In each repetition, the irregular times were generated using the following mixture of two exponential distributions,

$$f(t|\lambda_1, \lambda_2, \omega_1, \omega_2) = \omega_1 g(t|\lambda_1) + \omega_2 g(t|\lambda_2) \tag{4.1.11}$$

In this representation, the exponential distributions have means $1/\lambda_1$ and $1/\lambda_2$ and ω_1 and ω_2 are its respective weights.

In the following two experiments, the estimation of the ϕ and σ^2 parameters of the IAR model are assessed. In the first of them $\lambda_1 = 130$, $\lambda_2 = 6.5$, $\omega_1 = 0.15$ and $\omega_2 = 0.85$ have been used to generate the irregular times. In Table 4.1 it can be observed that the

Table 4.1: *Maximum likelihood estimation of simulated IAR series with mixture of Exponential distribution for the observational times, with* $\lambda_1 = 130$ *and* $\lambda_2 = 6.5$, $\omega_1 = 0.15$ *and* $\omega_2 = 0.85$.

Case	n	ϕ	$\widehat{\phi}$	SD($\widehat{\phi}$)	$\sigma(\widehat{\phi})$	$\widehat{\sigma}$
1	50	0.900	0.887	0.044	0.034	1.013
2	50	0.990	0.985	0.008	0.008	1.039
3	50	0.999	0.996	0.004	0.003	1.155
4	100	0.900	0.894	0.029	0.024	1.005
5	100	0.990	0.988	0.005	0.006	1.015
6	100	0.999	0.998	0.002	0.002	1.049

estimation of ϕ and σ^2 are close to the real values, even for smaller sample sizes.

It is interesting to assess the sensibility of the estimation procedure to changes in the time distribution. Consequently, we implement a new time distribution with the parameters set to $\lambda_1 = 300$, $\lambda_2 = 10$, $\omega_1 = 0.15$ and $\omega_2 = 0.85$. Table 4.2 confirms the results obtained in the first experiment, since for all combinations of sample size and ϕ the estimation of the parameters is very accurate.

Table 4.2: *Maximum likelihood estimation of simulated IAR series of size n, with Exponential distribution mix observation times,* $\lambda_1 = 300$ *and* $\lambda_2 = 10$.

Case	n	ϕ	$\widehat{\phi}$	SD($\widehat{\phi}$)	$\sigma(\widehat{\phi})$	$\widehat{\sigma}$
1	40	0.900	0.8843	0.058	0.038	1.011
2	40	0.990	0.9854	0.009	0.007	1.037
3	40	0.999	0.9969	0.003	0.002	1.120
4	80	0.900	0.8929	0.034	0.027	1.006
5	80	0.990	0.9876	0.005	0.005	1.018
6	80	0.999	0.9980	0.001	0.002	1.046

Therefore, the Monte Carlo simulations suggest that the finite-sample performance of the proposed methodology is accurate, and is not sensitive to smaller sample sizes and changes in the time distribution. In both cases high values of the parameter ϕ were used in order to have a significant time dependency in the simulated IAR process, taking in consideration that the time distributions that were chosen to have large time gaps.

Now, using the parameters of time distribution $\lambda_1 = 130$ and $\lambda_2 = 6.5$, $\omega_1 = 0.15$ and $\omega_2 = 0.85$ the parameter estimation procedure of the conditionally Gamma IAR process was assessed. In this case it is interesting also to assess the estimation performance of the CAR(1) process. To perform the CAR model we use both the package `cts` of R [73] and

the Python script developed by Pichara et al, (2012) [54].

Table 4.3: *Implementation of IAR and CAR models on simulated Gamma-IAR series in* ***R*** *and Python. For the observational times we use a mixture of two Exponential distributions with parameters* $\lambda_1 = 130$ *and* $\lambda_2 = 6.5$*,* $w_1 = 0.15$ *and* $w_2 = 0.85$*.*

	N	ϕ	σ	$\widehat{\phi}$	SD($\widehat{\phi}$)	$\widehat{\phi^C}$	SD($\widehat{\phi^C}$)	$\widehat{\sigma}$	SD($\widehat{\sigma}$)
R	100	0.9	1	0.899	0.014	0.418	0.306	0.984	0.170
R	100	0.99	1	0.990	0.001	0.890	0.201	0.985	0.161
R	200	0.9	1	0.899	0.010	0.355	0.286	0.993	0.122
R	200	0.99	1	0.990	0.001	0.900	0.184	0.998	0.120
Python	100	0.9	1	0.899	0.013	0.449	0.318	0.990	0.169
Python	100	0.99	1	0.990	0.001	0.919	0.169	0.981	0.200
Python	200	0.9	1	0.899	0.010	0.393	0.299	0.985	0.127
Python	200	0.99	1	0.990	0.001	0.927	0.163	0.996	0.332

In Table 4.3 $\widehat{\phi}$ is the estimation using the conditionally gamma IAR process and $\widehat{\phi^C}$ is the estimation using the CAR process. Evidently, the performance of the CAR(1) model is substantially worse than the one obtained with the IAR model. In addition, note that the performance of CAR(1) model using the Python function does not vary significantly regarding to the results obtained using R.

Another interesting experiment is to assess the ability of the regular time series models to fit an IAR sequence. The standard deviation of the innovations was computed (Eq (4.1.7)) to compare the goodness of fit of each model fitted to the sequence. The IAR sequence was generated with $\phi = 0.99$, $n = 100$ and the observational times was generated using the equation (4.1.11), with $\lambda_1 = 130$ and $\lambda_2 = 6.5$, $\omega_1 = 0.15$ and $\omega_2 = 0.85$. The basic idea is to fit this sequence using the IAR model, the regular autoregressive model of order one (AR(1)) and the ARFIMA(1,d,0) model, where the last two assume regular spaced data.

According to equation (4.1.7), the standard deviation of innovations changes for each observational time, which differs to the regular time series models in which the standard deviation of innovation is constant. Each standard deviation of the IAR model corresponds to the black dots in Figure 4.2. Here we can note that for large time gaps the standard deviation of innovations computed using the IAR model are greater than the ones computed by the AR and ARFIMA models, but if the observations are close, the smallest values are achieved using the irregular model. On average, the standard deviation of innovations is lower when using the IAR model than when using any of the other models.

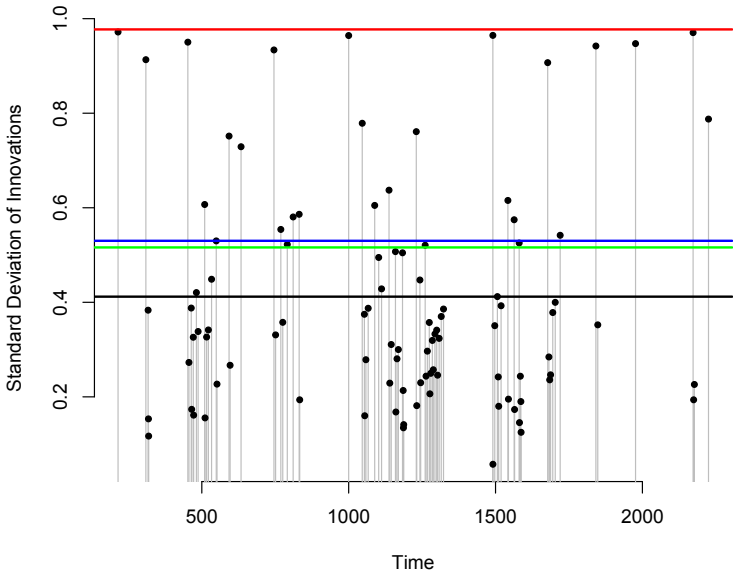


Figure 4.2: Comparison of standard deviation of innovations computed using the IAR model and other time series models that assumes regular times. The red line is the standard deviation of the data, the blue and green lines are the standard deviation of innovations computed using the AR(1) and ARFIMA(1,d,0) model respectively. The black line is the mean of the standard deviation of innovation computed using the IAR(1) model.

This result shows the importance of including the time difference in the definition of the model used to fit an irregular time series. However, both the CAR(1) and the IAR models only allow to estimate positive autocorrelation. That is a limitation of these models regarding to the regular autoregressive model that allows $-1 < \phi < 1$. In the case of the IAR model, the equation (4.1.1) shows that ϕ is powered to the time difference $t_j - t_{j-1}$, which could be a real number. In general a negative ϕ powered to a real number does not exist. Therefore, these models do not have the ability to detect and model negative autocorrelations.

Negative time dependencies appear often in some areas. In financial time series, is common to find significant negative autocorrelation for weekly and monthly stocks returns (further discussion can be found in Sewell (2011) [60]). Several authors agree that these are generally produced in stocks with a high trading frequency [20] [14].

Another well-known example of negatively correlated time series are the antipersistent processes which are characterized for having negative correlation for all positive lags

(For more details see Bondon & Palma (2007) [7]). The best known anti-persistent process is the Kolmogorov's energy spectrum of turbulence [36]. There are several examples in meteorology, e.g., Ausloos,M. and Ivanova,K. (2001) [5] which detect antipersistence in the fluctuations of the Southern Oscillation Index, e.g. sea level pressure. Also, the electricity prices in some Canadian provinces have an antipersistent behavior [71].

The problem of detecting negative time dependencies in irregularly sampled time series has been scarcely addressed in the literature. Chan and Tong (1987) [19] proved that a discrete-time AR(1) process with negative coefficient, always can be embedded in suitably chosen continuous-time ARMA(2,1) process, but this is a low parsimony solution. Alternatively, when an irregular time series have an antipersistent behavior it can be fitted by the CARFIMA process (Tsai H. (2009) [66]) with an intermediate memory, i.e., the Hurst parameter H is such that $0 < H < 1/2$.

In order to detect negative time dependencies, we propose a new alternative model for irregular time series that allows to estimate both positive and negative autocorrelation. This model is an extension of the irregular autoregressive model where now ϕ takes complex values. We call this model a complex irregular autoregressive model (CIAR). In what follows, we describe this model.

4.2 Complex Irregular Autoregressive (CIAR) model

In order to derive a complex extension of the model (4.1.1). we follow the approach of Sekita *et al.* (1991) [59], which build a complex autoregressive model for regular times. Consequently, suppose that x_{t_j} is a complex valued sequence, such that, $x_{t_j} = y_{t_j} + iz_{t_j}\ \forall\ j = 1, \ldots, n$. Likewise, $\phi = \phi^R + i\phi^I$ is the complex coefficient of the model and $\varepsilon_{t_j} = \varepsilon^R_{t_j} + i\varepsilon^I_{t_j}$ is a complex white noise. We define the complex irregular autoregressive (CIAR) process of order 1 as,

$$y_{t_j} + iz_{t_j} = (\phi^R + i\phi^I)^{t_j - t_{j-1}} (y_{t_{j-1}} + iz_{t_{j-1}}) + \sigma_{t_j}(\varepsilon^R_{t_j} + i\varepsilon^I_{t_j}), \tag{4.2.1}$$

where $\sigma_{t_j} = \sigma\sqrt{1 - |\phi^{t_j - t_{j-1}}|^2}$ and $|.|$ is the modulus of a complex number. Furthermore, we assume that only the real part y_{t_j} is observed and the imaginary part z_{t_j} is a latent process. In addition, $\varepsilon^R_{t_j}$ and $\varepsilon^I_{t_j}$, the real and imaginary part of ε_{t_j} respectively are independent with zero mean and variances $\mathbb{V}(\varepsilon^R_t) = 1$ and $\mathbb{V}(\varepsilon^I_t) = c$. The initial values are set to $y_{t_1} = \sigma\varepsilon^R_{t_1}$ and $z_{t_1} = \sigma\varepsilon^I_{t_1}$. In the next lemma are described some properties of this process.

Lemma 1: Let x_{t_j} a complex sequence that satisfies the equation (4.2.1). Then,

a) $\mathbb{E}(x_{t_j}) = 0$

b) $\mathbb{V}(x_{t_j}) = \gamma_x(0) = \sigma^2(1+c)$

c) The autocovariance function of the process is such that $\gamma_x(k) = \mathbb{E}(\overline{x}_{t_{j+k}} x_{t_j}) = \frac{(1+c)\phi^{t_{j+k}-t_j}\left|\sigma_{t_j}\right|^2}{1-\overline{\phi^{\delta_{j+k}}}\phi^{\delta_j}}$

d) The autocorrelation function of the process is such that $\rho_x(k) = \frac{\gamma_x(k)}{\gamma_x(0)} = \frac{\phi^{t_{j+k}-t_j}\left(1-\left|\phi^{\delta_j}\right|^2\right)}{1-\overline{\phi^{\delta_{j+k}}}\delta_j}$

where $\overline{\phi}$ is the complex conjugate of ϕ. In addition, note that under $|\phi| = |\phi^R + i\phi^I| < 1$, the results above prove that the complex sequence x_{t_j} is a weakly stationary process.

Proof: It is straightforward to show that $x_{t_j} = y_{t_j} + iz_{t_j}$ is such that $\mathbb{E}(x_{t_j}) = 0$. Denote $\delta_j = t_j - t_{j-1}$ the time differences and $\gamma_x(0) = \mathbb{E}(\overline{x}_{t_j} x_{t_j}) = \mathbb{V}(x_{t_j})$ the variance of the process, where $\overline{x}_{t_j}$ is the complex conjugate of x_{t_j}, therefore,

$$\begin{aligned}
\overline{x}_{t_j} x_{t_j} &= \left(\overline{\phi^{\delta_j}}\,\overline{x}_{t_{j-1}} + \overline{\sigma}_{t_j}\,\overline{\varepsilon}_{t_j}\right)\left(\phi^{\delta_j}\,x_{t_{j-1}} + \sigma_{t_j}\,\varepsilon_{t_j}\right) \\
&= \left|\phi^{\delta_j}\right|^2 \left|x_{t_{j-1}}\right|^2 + \ldots + \left|\sigma_{t_j}\right|^2 \left|\varepsilon_{t_j}\right|^2
\end{aligned}$$

Now, applying expectation $\mathbb{E}$ and using the properties of the model (4.2.1)

$$\begin{aligned}
\gamma_x(0) &= \left|\phi^{\delta_j}\right|^2 \gamma_x(0) + \left|\sigma_{t_j}\right|^2 (1+c) \\
\gamma_x(0) &= \frac{(1+c)\left|\sigma_{t_j}\right|^2}{1-\left|\phi^{\delta_j}\right|^2} \\
\gamma_x(0) &= \frac{(1+c)\sigma^2\left(1-\left|\phi^{\delta_j}\right|^2\right)}{1-\left|\phi^{\delta_j}\right|^2} = (1+c)\sigma^2 = \mathbb{V}(x_{t_j})
\end{aligned}$$

The covariance of the process is defined such that $\gamma_x(k) = \mathbb{E}(\overline{x}_{t_{j+k}} x_{t_j})$, then,

$$\begin{aligned}
\overline{x}_{t_{j+k}} x_{t_j} &= \left(\overline{\phi^{\delta_{j+k}}}\,\overline{x_{t_{j+k-1}}} + \overline{\sigma_{t_{j+k}}}\,\overline{\varepsilon_{t_{j+k}}}\right)\left(\phi^{\delta_j} x_{t_{j-1}} + \sigma_{t_j}\varepsilon_{t_j}\right) \\
&= \overline{\phi^{\delta_{j+k}}}\phi^{\delta_j}\overline{x_{t_{j+k-1}}} x_{t_{j-1}} + \ldots + \overline{\phi^{\delta_{j+k}}}\,\overline{x_{t_{j+k-1}}}\sigma_{t_j}\,\varepsilon_{t_j}
\end{aligned}$$

Now, applying expectation $\mathbb{E}$ we have,

$$\begin{aligned}\gamma_x(k) &= \overline{\phi^{\delta_{j+k}}}\phi^{\delta_j}\gamma_x(k) + \overline{\phi^{\delta_{j+k}}}\,\sigma_{t_j}\,\mathbb{E}(\overline{x_{t_{j+k-1}}}\varepsilon_{t_j}) \\ \gamma_x(k) &= \frac{\overline{\phi^{\delta_{j+k}}}\,\sigma_{t_j}\,\mathbb{E}(\overline{x_{t_{j+k-1}}}\varepsilon_{t_j})}{1-\overline{\phi^{\delta_{j+k}}}\phi^{\delta_j}}\end{aligned}$$

where we can prove by recursive replacing of the definition of $x_{t_{j-k}}$ that the numerator of the latter expression is equal to $(1+c)\phi^{t_{j+k}-t_j}\left|\sigma_{t_j}\right|^2$. Finally, the autocovariance function is,

$$\gamma_x(k) = \frac{(1+c)\phi^{t_{j+k}-t_j}\left|\sigma_{t_j}\right|^2}{1-\overline{\phi^{\delta_{j+k}}}\phi^{\delta_j}}$$

and therefore, $\rho_x(k) = \frac{\phi^{t_{j+k}-t_j}\left(1-\left|\phi^{\delta_j}\right|^2\right)}{1-\overline{\phi^{\delta_{j+k}}}\delta_j}$

Note that the autocorrelation function $\rho(k)$ of the CIAR process decay in a rate $\phi^{t_{j+k}-t_j}$ (also called exponential decay). This autocorrelation structure makes the difference regarding antipersistent or intermediate memory CARFIMA process, since the autocorrelation function of the latter decays more slowly than an exponential decay. Thus, although both models can fit irregular time series with negative autocorrelation, the appropriate use of these models will depend on the correlation structure of the data.

In this work, we propose an implementation of the CIAR model derived from the State-Space systems. The representation of this model in a state-space system allows us to implement the Kalman filter in order to get the maximum likelihood estimators of the parameters of the model.

4.2.1 State-Space Representation of CIAR Model

To represent the CIAR model in a state-space system the equation (4.2.1) must be rewritten. This is achieved following the results from the next lemma,

Lemma 2: The CIAR process described by (4.2.1) can be expressed by the following equation,

$$y_{t_j} + iz_{t_j} = (\alpha^R_{t_j} + i\alpha^I_{t_j})\,(y_{t_{j-1}} + iz_{t_{j-1}}) + \sigma_{t_j}(\varepsilon^R_{t_j} + i\varepsilon^I_{t_j}), \tag{4.2.2}$$

where $\alpha^R_{t_j} = |\phi|^{\delta_j}\cos(\delta_j\psi)$, $\alpha^I_{t_j} = |\phi|^{\delta_j}\sin(\delta_j\psi)$, $\delta_j = t_j - t_{j-1}$, $\psi = arccos\left(\frac{\phi^R}{|\phi|}\right)$ and $\phi = \phi^R + i\phi^I$.

Proof: The CIAR model follows the equation $y_{t_j} + iz_{t_j} = (\phi^R + i\phi^I)^{t_j - t_{j-1}} (y_{t_{j-1}} + iz_{t_{j-1}}) + \sigma_{t_j}(\varepsilon^R_{t_j} + i\varepsilon^I_{t_j})$. Let's focus on the term" $(\phi^R + i\phi^I)^{t_j - t_{j-1}}$,

$$(\phi^R + i\phi^I)^{t_j - t_{j-1}} = (\phi^R + i\phi^I)^{\delta_j} = |\phi|^{\delta_j} \left(\frac{\phi^R + i\phi^I}{|\phi|} \right)^{\delta_j}$$

where $\delta_j = t_j - t_{j-1}$ and $\phi = \phi^R + i\phi^I$. Using the polar representation for complex numbers we obtain,

$$\begin{aligned} \left(\frac{\phi^R + i\phi^I}{|\phi|} \right)^{\delta_j} &= (\cos(\psi) + i\sin(\psi))^{\delta_j} \\ \left(\phi^R + i\phi^I \right)^{\delta_j} &= |\phi|^{\delta_j} (\cos(\psi) + i\sin(\psi))^{\delta_j} \end{aligned}$$

where $\psi = arccos\left(\frac{\phi^R}{|\phi|}\right)$. Now, using the Moivre property, we have

$$\begin{aligned} \left(\phi^R + i\phi^I \right)^{\delta_j} &= |\phi|^{\delta_j} (\cos(\delta_j \psi) + i\sin(\delta_j \psi)) \\ &= |\phi|^{\delta_j} \cos(\delta_j \psi) + i|\phi|^{\delta_j} \sin(\delta_j \psi) \\ &= \alpha^R_{t_j} + i\alpha^I_{t_j} \end{aligned}$$

where $\alpha^R_{t_j} = |\phi|^{\delta_j} cos(\delta_j \psi)$ and $\alpha^I_{t_j} = |\phi|^{\delta_j} \sin(\delta_j \psi)$. Finally, the Complex IAR model can be represented by the expression,

$$y_{t_j} + iz_{t_j} = (\alpha^R_{t_j} + i\alpha^I_{t_j}) (y_{t_{j-1}} + iz_{t_{j-1}}) + \sigma_{t_j}(\varepsilon^R_{t_j} + i\varepsilon^I_{t_j})$$

Note that following this representation we can express the observed process as $y_{t_j} = \alpha^R_{t_j} y_{t_{j-1}} - \alpha^I_{t_j} z_{t_{j-1}} + \sigma_{t_j} \varepsilon^R_{t_j}$ and the latent process $z_{t_j} = \alpha^I_{t_j} y_{t_{j-1}} + \alpha^R_{t_j} z_{t_{j-1}} + \sigma_{t_j} \varepsilon^I_{t_j}$. Note that the process y_{t_j} is an IAR with parameter ϕ by assuming $\alpha^I_{t_j} = 0$ and $\alpha^R_{t_j} = \phi^{t_j - t_{j-1}}$. In addition, it is straightforward to show that $\alpha^I_{t_j} = 0$ is equivalent to $\phi_I = 0$.

Another important consideration is that the observed process y_{t_j} does not depend directly on $\varepsilon^I_{t_j}$. Consequently, the variance of the imaginary part c is a nuisance parameter, in the sense that can take any value in $\mathbb{R}^+$ and do not cause significant changes in the model.

The equation (4.2.2) can be represented by the space-state system (2.2.6) - (2.2.7) under $t = t_j$ and $X_{t_j} = \begin{pmatrix} y_{t_j} \\ z_{t_j} \end{pmatrix}$. According to the initial assumption of the model, it is only

observable y_{t_j} which implies that $Y_{t_j} = y_{t_j}$. Consequently, $G = \begin{pmatrix} 1 & 0 \end{pmatrix}$ is the observation matrix under this representation.

To complete the specification we define the transition matrix as $F_{t_j} = \begin{pmatrix} \alpha_{t_j}^R & -\alpha_{t_j}^I \\ \alpha_{t_j}^I & \alpha_{t_j}^R \end{pmatrix}$, the noise of both equations as $V_{t_j} = \sigma_{t_j} \begin{pmatrix} \varepsilon_{t_j}^R \\ \varepsilon_{t_j}^I \end{pmatrix}$ and $W_{t_j} = 0$. Finally, the observation and state equations of the state-space representation of CIAR model are,

$$\begin{pmatrix} y_{t_j} \\ z_{t_j} \end{pmatrix} = \begin{pmatrix} \alpha_{t_j}^R & -\alpha_{t_j}^I \\ \alpha_{t_j}^I & \alpha_{t_j}^R \end{pmatrix} \begin{pmatrix} y_{t_{j-1}} \\ z_{t_{j-1}} \end{pmatrix} + \sigma_{t_j} \begin{pmatrix} \varepsilon_{t_j}^R \\ \varepsilon_{t_j}^I \end{pmatrix} \tag{4.2.3}$$

$$y_{t_j} = \begin{pmatrix} 1 & 0 \end{pmatrix} \begin{pmatrix} y_{t_j} \\ z_{t_j} \end{pmatrix} \tag{4.2.4}$$

Note that in this representation, the transition matrix and the variance of noise term of the state equation $Q_{t_j} = |\sigma_{t_j}|^2 \mathbb{V}\left(\varepsilon_{t_j}\right)$ depend on time.

Lemma 3: Let $\alpha_{t_j} = \alpha_{t_j}^R + i\alpha_{t_j}^I$. If $|\alpha_{t_j}| < 1$, the process (4.2.3) is stable and it has a unique stationary solution given by,

$$X_{t_j} = V_{t_j} + \sum_{k=1}^{\infty} V_{t_{j-k}} \prod_{i=0}^{k-1} F_{t_{j-i}}$$

where $V_{t_{j-k}} = \sigma_{t_{j-k}} \begin{pmatrix} \varepsilon_{t_{j-k}}^R \\ \varepsilon_{t_{j-k}}^I \end{pmatrix}$

Proof: Let the transition matrix of CIAR process $F_{t_j} = \begin{pmatrix} \alpha_{t_j}^R & -\alpha_{t_j}^I \\ \alpha_{t_j}^I & \alpha_{t_j}^R \end{pmatrix}$, then for a specific time t_j the eigenvalues of F_{t_j} satisfy the follow equation

$$\begin{aligned} |(F_{t_j} - \lambda I)| &= \left| \begin{pmatrix} \alpha_{t_j}^R & -\alpha_{t_j}^I \\ \alpha_{t_j}^I & \alpha_{t_j}^R \end{pmatrix} - \begin{pmatrix} \lambda & 0 \\ 0 & \lambda \end{pmatrix} \right| = 0 \\ &= \left| \begin{pmatrix} \alpha_{t_j}^R - \lambda & -\alpha_{t_j}^I \\ \alpha_{t_j}^I & \alpha_{t_j}^R - \lambda \end{pmatrix} \right| \\ 0 &= (\alpha_{t_j}^R - \lambda)^2 + \alpha_{t_j}^{I2} \\ 0 &= \lambda^2 - 2\alpha_{t_j}^R \lambda + (\alpha_{t_j}^{R2} + \alpha_{t_j}^{I2}) \end{aligned}$$

$$\Rightarrow \lambda = \frac{2\alpha_{t_j}^R \pm \sqrt{4\alpha_{t_j}^{R2} - 4(\alpha_{t_j}^{R2} + \alpha_{t_j}^{I2})}}{2} = \alpha_{t_j}^R \pm i\alpha_{t_j}^I$$

As $|\alpha_{t_j}^R + i\alpha_{t_j}^I| = |\alpha_{t_j}^R - i\alpha_{t_j}^I| = |\alpha_{t_j}|$, then the process is stable if $|\alpha_{t_j}| < 1$. Therefore, under this assumption the CIAR process has the unique stationary solution (Brockwell & Davis (2002) [12]) given by

$$\begin{aligned} X_{t_j} &= F_{t_j} X_{t_{j-1}} + V_{t_j} \\ &= F_{t_j}(F_{t_{j-1}} X_{t_{j-2}} + V_{t_{j-1}}) + V_{t_j} \\ &= F_{t_j} F_{t_{j-1}} X_{t_{j-2}} + F_{t_j} V_{t_{j-1}} + V_{t_j} \\ &= F_{t_j} F_{t_{j-1}}(F_{t_{j-2}} X_{t_{j-3}} + V_{t_{j-2}}) + F_{t_j} V_{t_{j-1}} + V_{t_j} \\ &= F_{t_j} F_{t_{j-1}} F_{t_{j-2}} X_{t_{j-3}} + F_{t_j} F_{t_{j-1}} V_{t_{j-2}} + F_{t_j} V_{t_{j-1}} + V_{t_j} \end{aligned}$$

Therefore, the general form can be written as,

$$X_{t_j} = X_{t_{j-n}} \prod_{k=0}^{n-1} F_{t_{j-k}} + V_{t_j} + \sum_{k=1}^{n-1} V_{t_{j-k}} \prod_{i=0}^{k-1} F_{t_{j-i}}$$

As $\left|\prod_{k=0}^{n-1} F_{t_{j-k}}\right| = \prod_{k=0}^{n-1} \left|F_{t_{j-k}}\right|$ and $|F_{t_{j-k}}| < 1$ due to the stability of the process, then $\lim_{n\to\infty} \prod_{k=0}^{n-1} F_{t_{j-k}} = 0$. Finally if $n \to \infty$ then the unique stationary solution is given by,

$$X_{t_j} = V_{t_j} + \sum_{k=1}^{\infty} V_{t_{j-k}} \prod_{i=0}^{k-1} F_{t_{j-i}}$$

4.2.2 Estimation of CIAR Model

For the state-space model (4.2.3)- (4.2.4), the one-step predictors $\hat{X}_{t_j} = P_{t_{j-1}}(X_{t_j})$ and their error covariance matrices $\Omega_{t_j} = \mathbb{E}[(X_{t_j} - \hat{X}_{t_j})(X_{t_j} - \hat{X}_{t_j})']$ are unique and determined by the initial values: $\hat{X}_{t_1} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$ and $\Omega_{t_1} = \mathbb{E}[(X_{t_1} - \hat{X}_{t_1})(X_{t_1} - \hat{X}_{t_1})']$. Using the properties of the model (4.2.1) we can rewrite Ω_{t_1} as,

$$\Omega_{t_1} = \sigma^2 \begin{pmatrix} 1 & 0 \\ 0 & c \end{pmatrix}.$$

The Kalman recursions, for $j = 1, \ldots, n-1$ are defined by,

$$\Lambda_{t_j} = G_{t_j} \Omega_{t_j} G_{t_j}' \tag{4.2.5}$$

$$\Theta_{t_j} = F_{t_j} \Omega_{t_j} G_{t_j}' \tag{4.2.6}$$

$$\Omega_{t_{j+1}} = F_{t_j} \Omega_{t_j} F_{t_j}' + Q_{t_j} - \Theta_{t_j} \Lambda_{t_j}^{-1} \Theta_{t_j}' \tag{4.2.7}$$

$$\nu_{t_j} = y_{t_j} - G_{t_j} \hat{X}_{t_j} \tag{4.2.8}$$

c

$$\hat{X}_{t_{j+1}} = F_{t_j}\hat{X}_{t_j} + \Theta_{t_j}\Lambda_{t_j}^{-1}\nu_{t_j} \tag{4.2.9}$$

where $\{\nu_{t_j}\}$ is called the innovation sequence.
The maximum likelihood estimators of the CIAR model parameters ϕ^R and ϕ^I can be obtained by minimizing the reduced likelihood defined as,

$$\ell(\phi) \propto \frac{1}{n}\sum_{j=1}^{n}\left(\log(\Lambda_{t_j}) + \frac{\nu_{t_j}^2}{\Lambda_{t_j}}\right)$$

where Λ_{t_j} and ν_{t_j} comes from the Kalman recursion.

4.2.3 Simulation Results

Similarly to the procedure used in section 4.1.3, the finite sample performance of the estimation procedure with Kalman filter of the CIAR model will be assessed performing Monte Carlo experiments based on 1000 repetitions of each simulation. In each repetition we generate a CIAR sequence corresponding to the equation (4.2.1) using coefficients with different positive and negative values of the real part ϕ^R. In addition, both the imaginary part of the coefficient and the imaginary variance are set to $\phi^I = 0$ and $c = 1$. The irregular times are also generated using the mixture of exponentials distributions defined in (4.1.11). In this case, $\lambda_1 = 15$ and $\lambda_2 = 2$ are chosen as the mean of each exponential distribution $\omega_1 = 0.15$ and $\omega_2 = 0.85$ as its respective weights.

In addition, in order to assess the performance of the IAR model when the CIAR process is generated with a negative correlation, we also estimate the parameter ϕ of the IAR model. In Table 4.4 the results of the Monte Carlo simulations are shown. We can note that the finite-sample performance of the proposed methodology is accurate both for positive and negative values of ϕ^R. In addition, the estimation of the IAR parameter $(\widehat{\phi_{IAR}})$ is close to the value with which the CIAR process was generated ϕ^R, when this value is positive. But when the CIAR process is generated with negative ϕ^R the estimation of the IAR coefficient is close to zero. This result is important since it shows that the IAR model can't identify negative values of ϕ unlike the CIAR model. Finally, note that the accuracy of the estimated values does not depend on the magnitude of the given coefficient.

4.2.4 Comparing the CIAR with other time series models

In order to assess the performance of another time series models in capturing negative time dependences produced by the CIAR model, we generate the sequence of CIAR observations $\{y_1, \ldots, y_n\}$ with $\phi^R = -0.99$, $\phi^I = 0$, $c = 1$, $n = 300$ and the irregular times are defined as in the above example. We fit this sequence using the follow time series models, IAR, AR(1), ARFIMA and CAR(1), and CIAR and compute the root mean squared error

Table 4.4: *Maximum likelihood estimation of complex ϕ computed by the CIAR model in simulated IAR data. The observational times are generated using a mixture of Exponential distribution with $\lambda_1 = 15$ and $\lambda_2 = 2$, $w_1 = 0.15$ and $w_2 = 0.85$.*

Case	N	ϕ^R	$\widehat{\phi^R}$	SD($\widehat{\phi^R}$)	ϕ^I	$\widehat{\phi^I}$	SD($\widehat{\phi^I}$)	$\widehat{\phi_I}$	SD($\widehat{\phi_I}$)
1	300	0.999	0.9949	0.0036	0	0.0009	0.0030	0.9949	0.0036
2	300	0.9	0.8960	0.0187	0	0.0116	0.0413	0.8950	0.0188
3	300	0.7	0.6967	0.0412	0	0.0557	0.0819	0.6948	0.0406
4	300	0.5	0.4942	0.0596	0	0.0849	0.1111	0.4965	0.0569
5	300	-0.999	-0.9984	0.0012	0	0.0001	0.0009	0.0626	0.0265
6	300	-0.9	-0.8991	0.0154	0	0.0014	0.0134	0.0643	0.0299
7	300	-0.7	-0.6991	0.0414	0	0.0061	0.0354	0.0628	0.0289
8	300	-0.5	-0.4971	0.0717	0	0.0091	0.0607	0.0589	0.0283

(RMSE). Figure 4.4 shows that the RMSE of all the models with the exception of the CIAR model has similar values regarding to the standard deviation of the data. On the other hand, the CIAR is the only model that reduce significantly the standard deviation of the sequence.

Further, in order to assess the stability of the above result, this experiment is repeated 1000 times. In addition, we also fitted a CIAR process with positive time dependency ($\phi^R = 0.99$) with the time series models mentioned above. Figure 4.4 a) shows that the RMSE estimated by the irregular time series models are smaller than both the AR and ARFIMA models (both assumes regular sampling). However, as can be seen in Figure 4.4 b) the CIAR model has a significantly better performance in fitting the negatively correlated processes generated than the other models tested. Therefore, we verify that a CIAR process with a large and negative time dependency can not be correctly modeled with the conventional time series models, including those that assume irregular sampling.

For a complete view of the performance of the time series models in fitting the CIAR process, the procedure described above is repeated using different values of ϕ^R. Consequently, in each iteration 100 sequences of the CIAR process are generated taking a total of 20 values of ϕ^R equally spaced in the range (-1,1). In this experiment two distributional times were implemented. Figure 4.4 c) is obtained using times generated by the equation (4.1.11) with $\lambda_1 = 15$ and $\lambda_2 = 2$. Figure 4.4 d) is obtained using times generated by the equation (4.1.11) with $\lambda_1 = 130$ and $\lambda_2 = 6.5$. Figures 4.4 c)-d) show that the largest difference in performance between CIAR and the other time series models fitted, occur for high and negative autocorrelations. This difference tends to be smaller when the autocorrelation parameter approaches 0. Finally, for positive autocorrelations the irregular models have the same performance. The main difference between the two experiments is that for high and positive autocorrelations both the regular and irregular models have the

cii

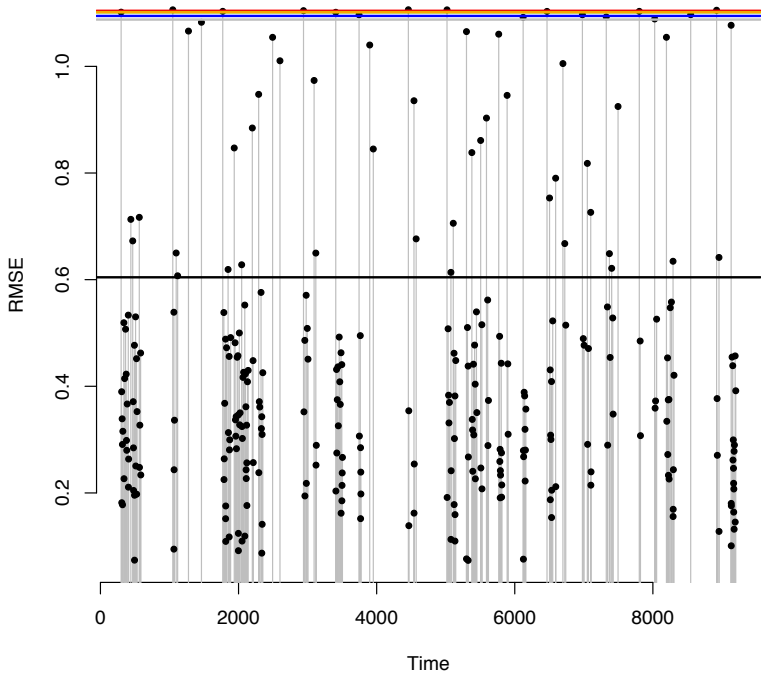


Figure 4.3: Comparison of root mean squared error at each time of a sequence simulated with the IAR model with parameter $\phi^R = -0.99$, $\phi^I = 0$, $c = 0$ and length $n = 300$. The red line corresponds to the standard deviation of the sequence, the blue, green, gray and orange lines correspond to the RMSE computed when the sequence was fitted with IAR(1), AR(1), ARMA(2,1), CAR(1) models respectively. The black line corresponds to the root mean squared error of the CIAR model, where the black dots are the individual RMSE at each time.

same performance when the observational times are closest.

4.2.5 Computing the Autocorrelation in an Harmonic Model

The main advantage of the CIAR process over the other irregular time series models is its ability to model weakly stationary time series with negative autocorrelation. However, another interesting application of this model is to use it to detect if an irregular time series have a negative autocorrelation. A well-known example of a weakly stationary time series

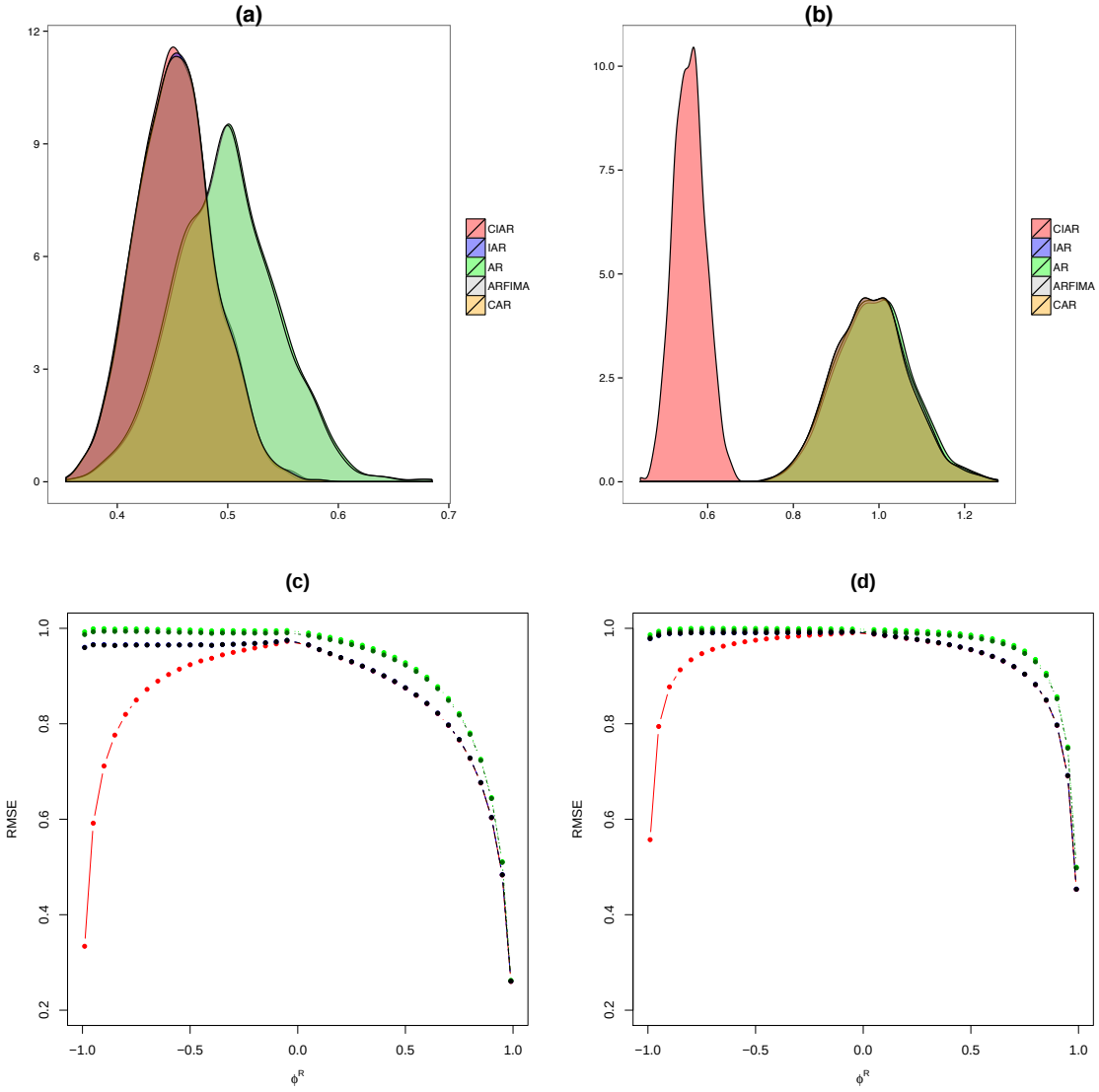


Figure 4.4: In the first row are shown on figures (a) and (b) the kernel Distributions of the root mean squared error computed for the fitted models on the 1000 CIAR sequences simulated. In a) each CIAR process was generated using $\phi^R = -0.99$. In b) each CIAR process was generated using $\phi^R = 0.99$. The other parameters of the models were defined as $\phi^I = 0$, $c = 0$ and length $n = 300$. In the second row are shown on figures (c) and (d) the RMSE computed for different values of the autocorrelation parameter ϕ^R of the CIAR model. The red, blue, green, darkgreen and black lines correspond to the RMSE computed for the CIAR, IAR, AR, ARFIMA and CAR models respectively. In Figure (c) the observational times are generated using a mixture of Exponential distribution with $\lambda_1 = 15$ and $\lambda_2 = 2$, $\omega_1 = 0.15$ and $\omega_2 = 0.85$. In figures (a), (b) and (d) the observational times are generated using a mixture of Exponential distribution with $\lambda_1 = 130$ and $\lambda_2 = 6.5$, $\omega_1 = 0.15$ and $\omega_2 = 0.85$.

that can be negatively correlated is the following harmonic process,

$$y_{t_i} = A\sin(ft_i + \psi) + \epsilon_{t_i} \tag{4.2.10}$$

where f is the frequency of the process and ϵ_{t_i} is a white noise sequence with mean 0 and variance σ^2. In addition, the amplitude A is a fixed parameter and the phase ψ is a random variable with uniform distribution between $-\pi$ and π. Assuming irregular times, the one-step autocorrelation is given by $\rho_1 = \cos(f)$ [13]. It can be proved that this result is also met under irregular times. Note that this autocorrelation is negative for $f \in (\pi/2, \pi)$. In addition, note that for higher frequency values the harmonic process 4.2.10 becomes more anti-persistent, it has been discussed by some authors (e.g., Alperovich, et al (2017) [3]).

A simulation study was performed in order to assess whether the CIAR model can detect the correlation structure of an irregular harmonic model. Suppose that we generated the irregular observational times t_i with $i = 1, \ldots, n$ using the mixture of exponentials distributions (equation (4.1.11)) with $\lambda_1 = 130$ and $\lambda_2 = 6.5$, $\omega_1 = 0.15$ and $\omega_2 = 0.85$. Later, the process y_{t_i} is simulated with length $n = 300$, amplitude $A = 20$ and a unit variance for ϵ_{t_i}. The procedure is repeated $k = 200$ times using k different frequencies taken equally spaced from the interval $(0, \pi)$. We fit the CIAR model to each simulated sequence. In Figure 4.5 the parameters estimated from CIAR model are shown. The theoretical autocorrelation is also added in the plot. Note that, the ϕ^R parameter estimated by the CIAR model is close to the theoretical value.

4.3 Application of Irregular Time Series Models in Astronomical time series

As mentioned previously, in astronomical data is common to observe irregular time series in which these models can be implemented. Particularly, in the analysis of the light curves of variable stars, these models can be useful to characterize the light curves according to its structure of temporal dependence.

In the previous section, I mentioned that the light curves are generally fitted by an harmonic model (see equation (3.0.1)). One application of the irregular time series models in the astronomical context, is to detect whether the harmonic model is sufficient to capture all the temporal dependency in the light curve, i.e., whether a temporal dependency structure remains in the residuals of the harmonic model.

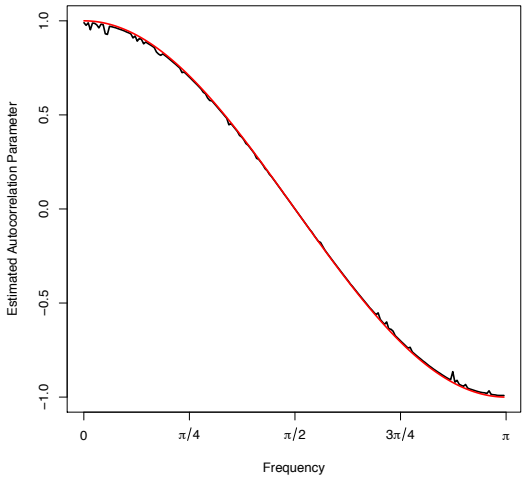


Figure 4.5: Estimated coefficients (y-axis) by the CIAR model in $k = 200$ harmonic processes generated using frequencies (x-axis) in the interval $(0, \pi)$. The black line corresponds to the coefficients estimated by the CIAR model. The red line is the theoretical autocorrelation of the process y_{t_i}

The main idea is to first fit an harmonic model to the light curves of variable stars using only the first dominant period. Later, we fit the irregular time series models (IAR and CIAR) to the residuals of the harmonic fit. The estimated parameters $\hat{\phi}$ and $\hat{\phi}^R$ of the IAR and CIAR models respectively, will indicate whether it remains a time dependency on the residuals. A time dependency structure on the residuals of the harmonic fit can be due to several reasons. One possibility is that the light curve corresponds to a multiperiodic variable stars, and therefore an harmonic model fitted with only one period is not enough to fit the light curve. Another possibility is that the light curves was incorrectly fitted by the harmonic model, as for example using a wrong period in the harmonic fit.

In order to show some applications on real data, in this work the irregular time series models were applied on the light curves of variable stars from the optical surveys OGLE and Hipparcos. We use these surveys, since they have many class of variable stars available and, as these surveys use the optical I-band to make observations, the brightness magnitude of each star is measured with small errors.

All the light curves of the OGLE and Hipparcos surveys are fitted using an harmonic model with one period and 4 harmonic, according to equation (3.0.1). The residuals of each harmonic model are then fitted using IAR and CIAR models. In Figure 4.6 the estimated coefficients for both models are shown. Note that there is a high correlation between the estimated coefficients when both values are positive, which are consistent

with the results of the Monte Carlo simulations.

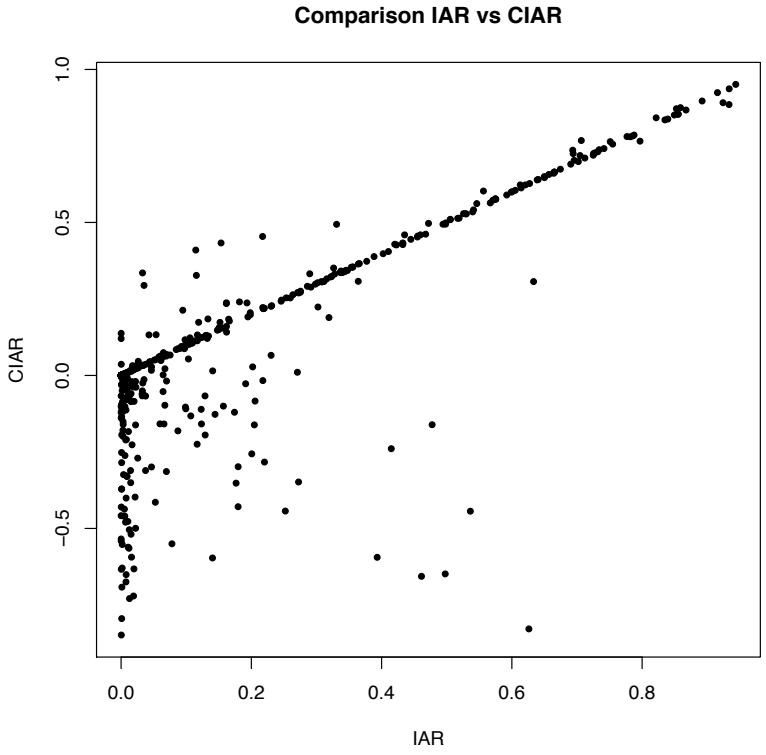


Figure 4.6: Values of the coefficient estimated by the CIAR and IAR models in OGLE and HIPPARCOS light curves.

However, we note several cases identified as uncorrelated or without dependency structure in residuals by the IAR model where the autocorrelation computed using CIAR model is high but negative. In other words, these light curves remain with negative dependency structure in the residuals after the fit of the harmonic model.

In addition, in order to assess the fitting performance of the IAR and CIAR models, we compute the root mean squared error (RMSE) after fitting each irregular model on the residuals of the harmonic model. These results do not vary significantly when ϕ^R is positive. However, if this coefficient is negative, the RMSE computed when we fit the residuals of the harmonic fit using the CIAR model are less than the ones obtained when we fit this data using the IAR model, as can be seen in Figure 4.7.

Therefore, the results obtained for the light curves in OGLE and Hipparcos surveys are consistent with the ones obtained in the Monte Carlo experiments in section 4.2.3,

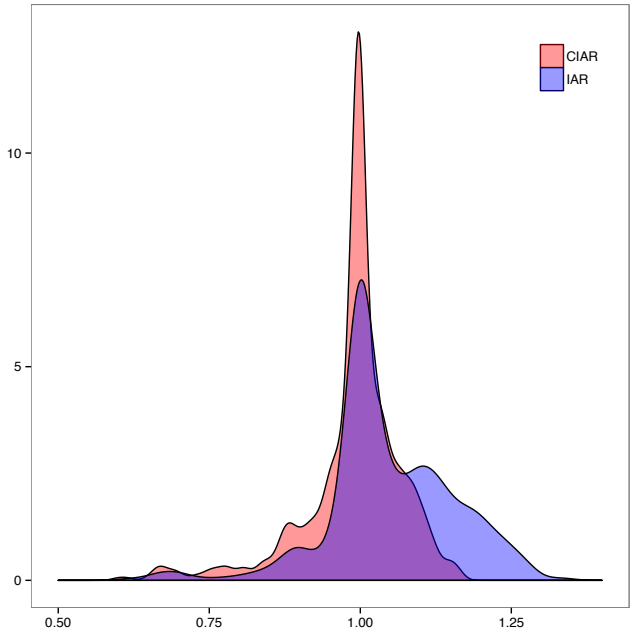


Figure 4.7: Kernel Density of the RMSE computed for the residuals of harmonic fit in the light curves when the CIAR coefficient is negative. The red density corresponds to the RMSE computed using the IAR model, and the green density corresponds to the RMSE computed using the CIAR model.

in the sense that the CIAR process can identify negative time dependencies that the IAR model cannot.

4.3.1 Irregular time series models to detect the harmonic model misspecification

As mentioned above, the time dependencies detected can be due to whether a light curve was incorrectly fitted by the harmonic model or the light curve corresponds to a multiperiodic variable stars. It is said that an harmonic model has been misspecified if it has incorrectly fitted to a light curve due to a bad specification of some of its parameters, for example, the period.

In order to show how the irregular time series models can identify misspecified harmonic models, forty variable stars from OGLE and Hipparcos surveys were selected, for which the harmonic model gives a precise fit of the light curve. These forty light curves are selected by a visual inspection of the 250 light curves with highest R squared value (R^2). The coefficient of determination R^2, is a widely used goodness of fit measure. To

Class	$f_1 \leq 0.1$	$0.1 < f_1 \leq 0.5$	$0.5 < f_1 \leq 1$	$1 < f_1 \leq 2$	$f_1 > 2$
Classical Cepheid (CLCEP)	2	4			
Chem. Peculiar (CP)		1			
Double Mode Cepheid (DMCEP)			1	2	
Delta Scuti (DSCUT)					2
Beta Persei (EA)		1	4	2	
Beta Lyrae (EB)	1		2	2	
W Ursae Maj (EW)		1	1	1	2
Mira (MIRA)	4				
PV Supergiants (PVSG)		1			
RR Lyrae, FM (RRAB)				1	1
RR Lyrae, FO (RRC)					2
Semireg PV (SR)	1				
SX Phoenicis (SXPHE)					1
Total	8	8	8	8	8

Table 4.5: Distribution of the forty selected examples by his frequency range and class of variable stars.

select these light curves a representative sample of the classes and frequencies values observed in OGLE and HIPPARCOS are take in consideration. In order to take a representative sample of the frequencies, five group of frequencies were created and eight light curves from each group were selected. In addition, in each frequency group, the light curves selected were from the most representative classes of each group. Table 4.5 shows the distribution of classes over the different frequency groups.

The hypothesis behind this application is that when a light curve was fitted by an harmonic model with the correct period, the residuals do not have any dependency on time. In other words, the parameter of the IAR (CIAR) model ϕ should be equal to zero. On the other hand, if we fit a model with a wrong period to the light curve, the residuals remain with dependency on time, and therefore $\phi > 0$.

In order to verify this hypothesis we perform an experiment, which consists in to fit wrongly each selected light curve. To fit wrongly these light curves, we used percentual variations of the real frequency in the interval $(f_1 - 0.5f_1, f_1 + 0.5f_1)$, where f_1 is the real frequency. In this interval a total of 38 frequency equally spaced are taken $g_1, \ldots, g_{38}$. Consequently, for each incorrect frequency g_j used to fit the harmonic model we obtain a $\hat{\phi}_j$ from the IAR model. The distribution of $\hat{\phi}$ when the light curve was fitted correctly (on the right) and when not (on the left) is shown in the boxplot of Figure 4.8. The distribution of $\hat{\phi}$ when the light curve was correctly fitted takes small values regarding to the values estimated when the light curve was incorrectly fitted.

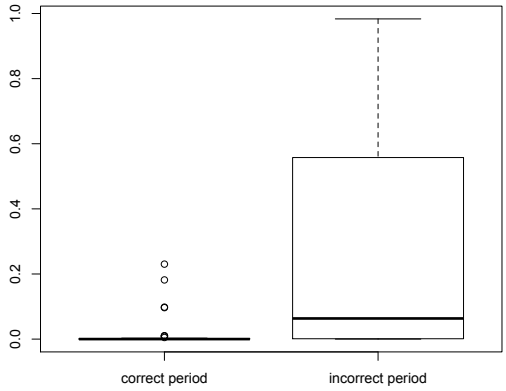


Figure 4.8: Boxplot of the distribution of ϕ, for the light-curves using the correct frequency (on the left) and for the light-curves using the incorrect frequency (on the right).

In addition, in Figure 4.9 three examples among the forty light curves are shown. In the first row (figures (a)-(c)) the folded light curve of a Classical Cepheid (CLCEP), W Ursae Maj (EW) and Delta Scuti (DSCUT) are shown. Each of these light curves is precisely fitted by the harmonic model. On the second row, (figures (d)-(f)) are shown the ϕ values estimated for the IAR model on the residuals of the harmonic model fitted using the percentual variations of the real frequency mentioned above. Note that a percentual variation (x-axis) equal to 0 means that the light curves is correctly fitted. In this case, the estimated values $\hat{\phi}$ are presented in the center of each plot. On the other hand, when the percentual variation is not equal to 0 means that the light curve is fitted using an incorrect period and the estimated ϕ are represented both in left and right of each plot. In all three examples, the smaller values of the estimated ϕ by the IAR model are close to zero and are obtained when the light curve is correctly fitted, while the maximum values of $\hat{\phi}$ are obtained when the light curve is incorrectly fitted. Note that the estimated parameters differ substantially in the three examples. Taking in consideration that the frequency of these three light curves are 0.06, 0.97 and 3.74 respectively, it can be seen that the estimated coefficients depend directly to the frequency of the light curve, which has values around 0.75, 0.18 and 7.5×10^{-5} respectively. In other words, this dependence means that the light curves with short periods have short values of $\hat{\phi}$. Consequently, it is not always possible to discriminate whether $\phi \neq 0$ is significative or not. In order to facilitate this decision, we develop a statistical test for assessing the significance of the parameter ϕ.

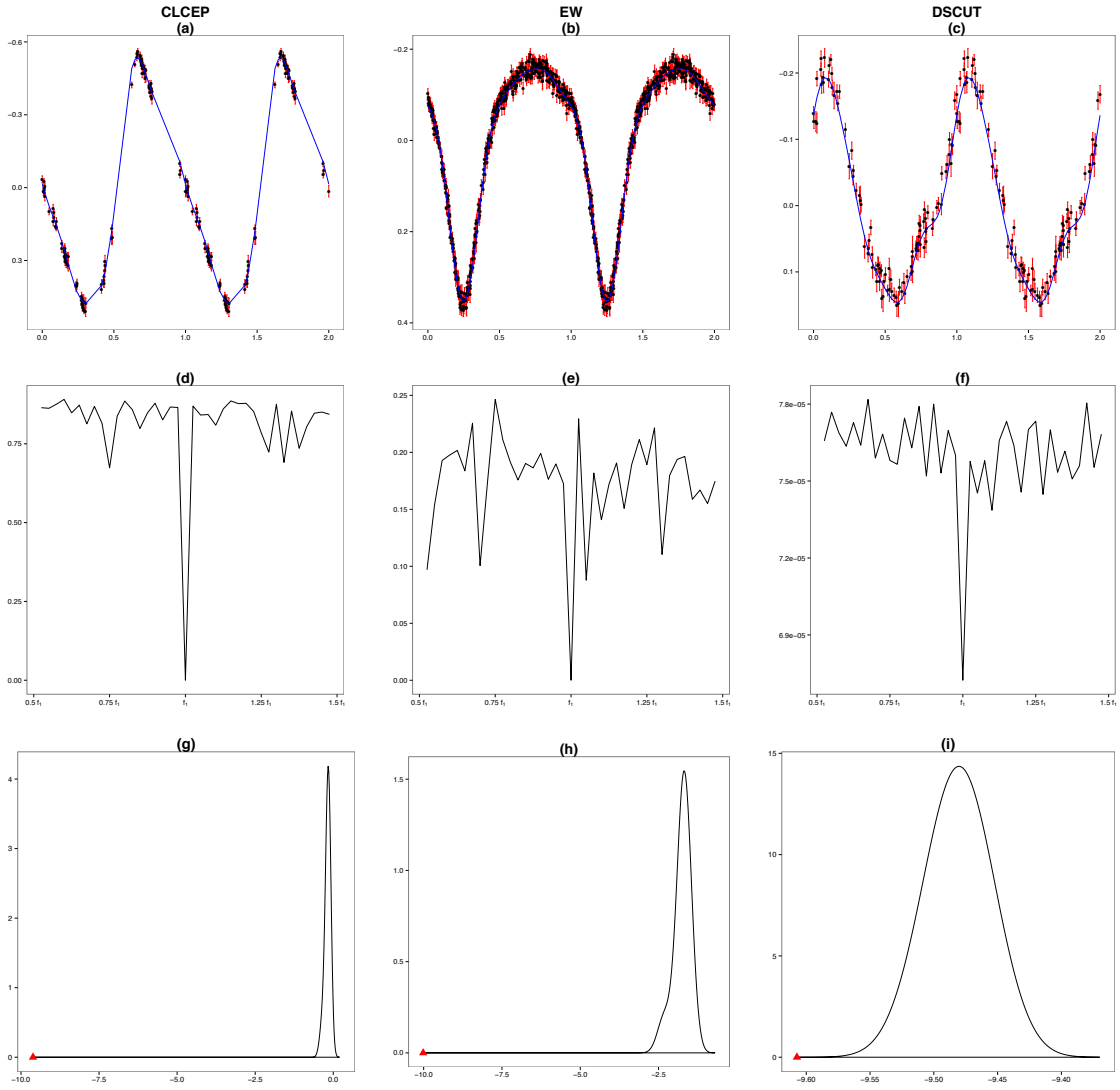


Figure 4.9: In the first row, the light curves of a Classical Cepheid, EW and DSCUT are shown on figures (a)-(c) respectively. The continuous blue line is the harmonic best fit. On the second row (figures (d)-(f)), for each of the variable stars, it is depicted on the x-axis the % of variation from the correct frequency, and on the y-axis is the estimate of the parameter ϕ of the IAR model obtained after fitting an harmonic model with the wrong period (except at zero that corresponds to the right period). On the third row (figures (g)-(i)), the distribution of the parameter ϕ of the IAR model is shown when each light curve is fitted with the wrong period. The red triangle corresponds to the value of ϕ when the correct period is used in the harmonic model fitting the light curves.

4.3.2 Statistical test for the autocorrelation parameter

In this section, a statistical test developed to assess the hypothesis $H_0 : \phi = 0$ vs $H_1 : \phi \neq 0$ is described. Based on the above result, we know that the estimated parameter on the residuals of an harmonic fit is minimized when the real frequency is used. So, in order to assess whether the estimated parameter is significantly different from zero, we must compare it with the remaining estimates. Consequently, the null and the alternative hypothesis can be rewritten as $H_0 : \phi \sim F_1$ vs $H_1 : \phi \nsim F_1$, where F_1 is the distribution of the $\underset{\sim}{\hat{\phi}} = \{\hat{\phi}_1, \ldots, \hat{\phi}_{38}\}$, where each $\hat{\phi}_j$ is computed in the residuals of the harmonic model fitted with an incorrect frequency g_j .

To develop the test for the IAR model, $\log(\hat{\phi})$ is assumed to follow a Gaussian distribution. Figures (g)-(h) shows the density of the $\log(\underset{\sim}{\hat{\phi}})$ computed in the wrongly fitted light curves, and the red triangle shows the $\log(\hat{\phi})$ values when the correct period is used. The p-values in the three examples are $0, 1.62 \times 10^{-281}, 2.86 \times 10^{-19}$ respectively, which is consistent with the precise fit of the light curves.

To perform an equivalent test for the ϕ^R parameter of the CIAR model, $\log(|\hat{\phi^R}|)$ is assumed to follow a Gaussian distribution.

Due to the limitation of the IAR model to detect negative autocorrelations, there are some examples in which the IAR model can't distinguish if the model was correctly fitted or not, but the CIAR model can do it. One example is shown in Figure 4.10. The light curve corresponds to a RR*c* star observed by the HIPPARCOS survey. Figure a) shows the perfect fit of the harmonic model to the light curve. Both the IAR and CIAR model were able to detect this precise fit, since the two models have estimated values closed to zero. However, only the CIAR model gives small estimated value of $|\phi^R|$ when the light curve was correctly adjusted. Consequently, in this case only the CIAR model can distinguish the estimation of ϕ in the correct harmonic fit from the wrong harmonic models (Figures b) and c)).

The light curves from OGLE and Hipparcos have similar cadence and distributions error. An additional interesting experiment would be to replicate the above application using other data with different characteristics. Another light curves examples can be found in the VVV-Templates project (Angeloni et al (2014) [4]). As mentioned in chapter 2, the VVV survey is characterized by using the near infrared to make observations, which allows us to observe more distant objects, but with greater measurement errors. In addition, the VVV light curves also differs from light curves observed in OGLE and HIPPARCOS in the cadence sampling time. The main difference is that in the light curves of the VVV,

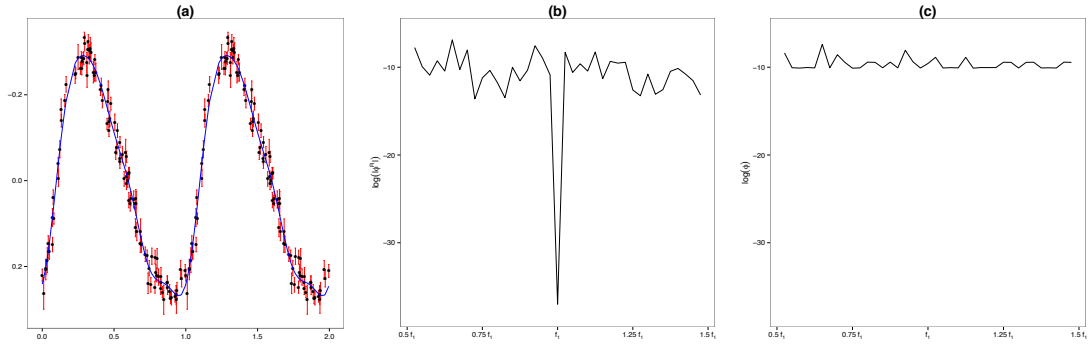


Figure 4.10: a) Light curve of a RR*c* star observed by the HIPPARCOS survey. The continuous blue line is the harmonic best fit. b) Logarithm of the absolute value of the estimated parameter $\hat{\phi}^R$ by the CIAR model on the residuals of the harmonic model fitted with different frequencies. In the x-axis are the percentual variation from the correct frequency, in the y-axis are the logarithm of $\hat{\phi}$. c) Logarithm of the estimated parameter $\hat{\phi}$ by the IAR model on the residuals of the harmonic model fitted with different frequencies. In the x-axis are the percentual variations from the correct frequency, in the y-axis is the logarithm of $\hat{\phi}$.

it is common to see observations spaced at time gaps greater than 100 days, unlike the light curves used so far.

In Figure (4.11) is shown a RR*ab* light curve from the VVV-Templates which was fitted very well by the harmonic model. Again, the CIAR model works well in the detection of the correct modeling, unlike to the IAR model.

4.3.3 Irregular time series models to detect multiperiodic variable stars

As mentioned above, the last application of the irregular time series models, and perhaps the most interesting is the ability to find multi-periodic variable stars. The basic idea is that, when a light curve that corresponds to a multi-periodic variable star with two periods (bi-periodic variable stars) is fitted by an harmonic model using only one period, the residuals remain with temporary dependency structure. Therefore, it is expected that the irregular models are able to detect this temporary dependence.

The difference with the first application is that, the existing temporary dependency in the residuals of the first harmonic model can be explained by an harmonic model using

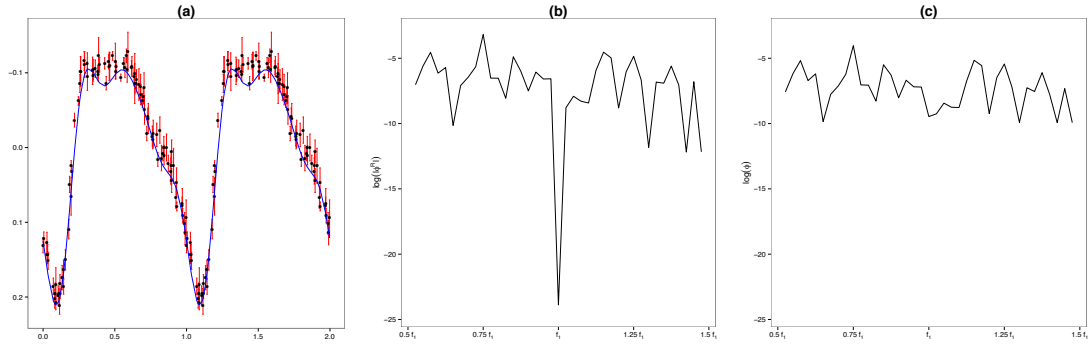


Figure 4.11: a) Light curve of a RR*ab* star observed by the VVV survey. The continuous blue line is the harmonic best fit. b) Logarithm of the absolute value of the estimated parameter $\hat{\phi}^R$ by the CIAR model on the residuals of the harmonic model fitted with different frequencies. In the x-axis are the percentual variation from the correct frequency, in the y-axis are the logarithm of $\hat{\phi}$. c) Logarithm of the estimated parameter $\hat{\phi}$ by the IAR model on the residuals of the harmonic model fitted with different frequencies. In the x-axis are the percentual variations from the correct frequency, in the y-axis is the logarithm of $\hat{\phi}$.

the second most important period. In other words, the time dependency of a light curve of a bi-periodic variable star must be explained using at least the two most important period in the harmonic model.

Consequently, for a bi-periodic variable star, the raw light curve has a temporal dependency. The residuals of the first harmonic model have a temporal dependency. The residuals of the second harmonic model should no longer have autocorrelation. Therefore, it is expected that irregular time series models have an estimated parameter different than zero in the first two cases, and approximately equal to zero for the residuals of the second harmonic model.

In order to illustrate this application, a bi-periodic light curve was simulated using a 2-harmonic model. This model can be derived from the equation 3.2.1. Consequently, the simulated light curve comes from the following equation,

$$y(t) = \sum_{i=1}^{2}\sum_{j=1}^{4}(\alpha_{ij}sin(2\pi f_i jt) + \beta_{ij}cos(2\pi f_i jt)) + \epsilon_t \tag{4.3.1}$$

where ϵ_t follows a Gaussian distribution with mean zero and unit variance. In this case, the bi periodic light curve is generated using the frequencies $f_1 = 1/3$ and $f_2 = 1/12$. Likewise to the previous simulations, the irregular times is generated using the mixture of two exponential distributions (equation (4.1.11)) with parameters $\lambda_1 = 130$ and $\lambda_2 = 6.5$,

$\omega_1 = 0.15$ and $\omega_2 = 0.85$.

In the plot a) of the Figure 4.12 are the residuals obtained after fitting the simulated time series with an harmonic model using one period. The estimated parameter ϕ by the IAR model fitted to these residuals was $\hat{\phi} = 0.5447$. This result is consistent with the temporal dependency observed in the plot a). The residuals after fitting the harmonic model using two periods are in the plot b). The estimated parameter ϕ by the IAR model fitted to these residuals is close to zero $\hat{\phi} \leq 0.0001$, which is consistent with the random behavior observed in plot b).

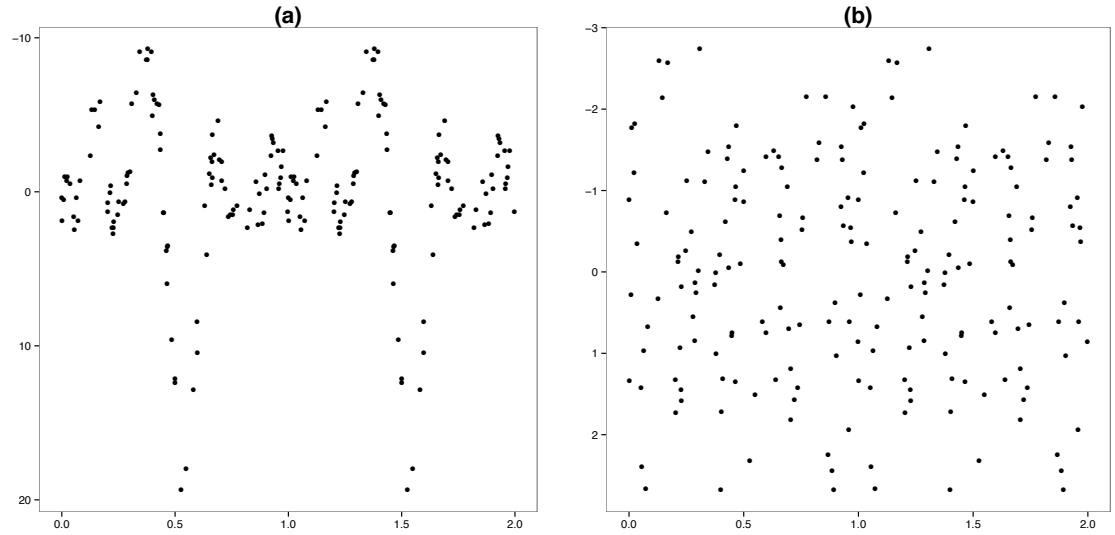


Figure 4.12: (a) Residuals of the best harmonic fit with one frequency for a simulated multiperiodic light curve; (b) Residuals of the best harmonic best fit with two frequencies for the same simulated multiperiodic light curve.

There are several classes of pulsating variable stars with multiperiodic behavior, for example the DMCEP and RRD. From the set of real light curves observed in the OGLE and Hipparcos surveys, a DMCEP that pulses in the first two radial overtones ($1O/2O$ type, $P_2/P_1 \in (0.79 - 0.81)$) was selected.

In Figure 4.13 a) are the residuals of the harmonic model fitted with the fundamental period. In Figure 4.13 b) are the residuals of the harmonic model fitted with the two most important period. It can be observed that after fitting the harmonic model with one period there remains a temporal dependency structure in the residuals plotted in figure a) , while

if we fit this light curve with two harmonics, there is no longer a temporary dependence on residuals (Figure b). The estimated parameter ϕ by the IAR model to the residuals in Figures a) and b) are 0.5411 and 0.033 respectively. If we fit the CIAR model to the same data, the $\hat{\phi}^R$ is also 0.5411. Therefore, both models could detect the multiperiodic behavior of this light curve.

Another example of a bi-periodic light curve also corresponding to a Double mode Cepheid class of variable star are in Figures c) and d) of Figure 4.13. Just like the previous example, it is clearly seen that the residuals of the harmonic model remain with time dependency structure. But, here the estimation using the CIAR model and IAR model differ. While $\hat{\phi}^R$ is -0.561 the $\hat{\phi}$ of the IAR model is 0.011. Therefore, this light curve is an example of negative time dependency structure in the residuals of the harmonic model, which cannot be detected using the IAR model.

4.3.4 Classification Features estimated from the irregular time series models

One of the most important aims in the light curves analysis is to find features that can discriminate one class of variable star from the others. Finding a good feature is the key to building a classifier with good performance to detect stars of a given class. Generally, these features are extracted from the temporal behavior of the brightness of each variable star. In this work, we propose to use the parameters estimated by both the IAR and CIAR models as features. As can be discussed in section 4.3.3, multiperiodic variable stars should have large estimated coefficients, which can distinguish them from other class of variable stars. In the OGLE and HIPPARCOS catalogues there are two classes of multiperiodic variable stars: the Double Mode RR Lyrae (RRD) and the Double Mode Cepheid (DMCEP).

The features that can be extracted from the irregular time series models are the estimated parameters ϕ and ϕ^R of the IAR and CIAR models respectively and the p-value associated to these estimations. It is interesting to assess whether these features can separate the multiperiodic classes from the other RR-Lyraes and Cepheids respectively. Figure a)-b) shows the distribution of the features computed by the CIAR model. As can be seen there are no significant differences between the classes of RR-Lyrae and Cepheids in the distributions of the estimated ϕ^R. However, the p-values computed for the RRD and DMCEP classes take larges values in comparison to the other classes.

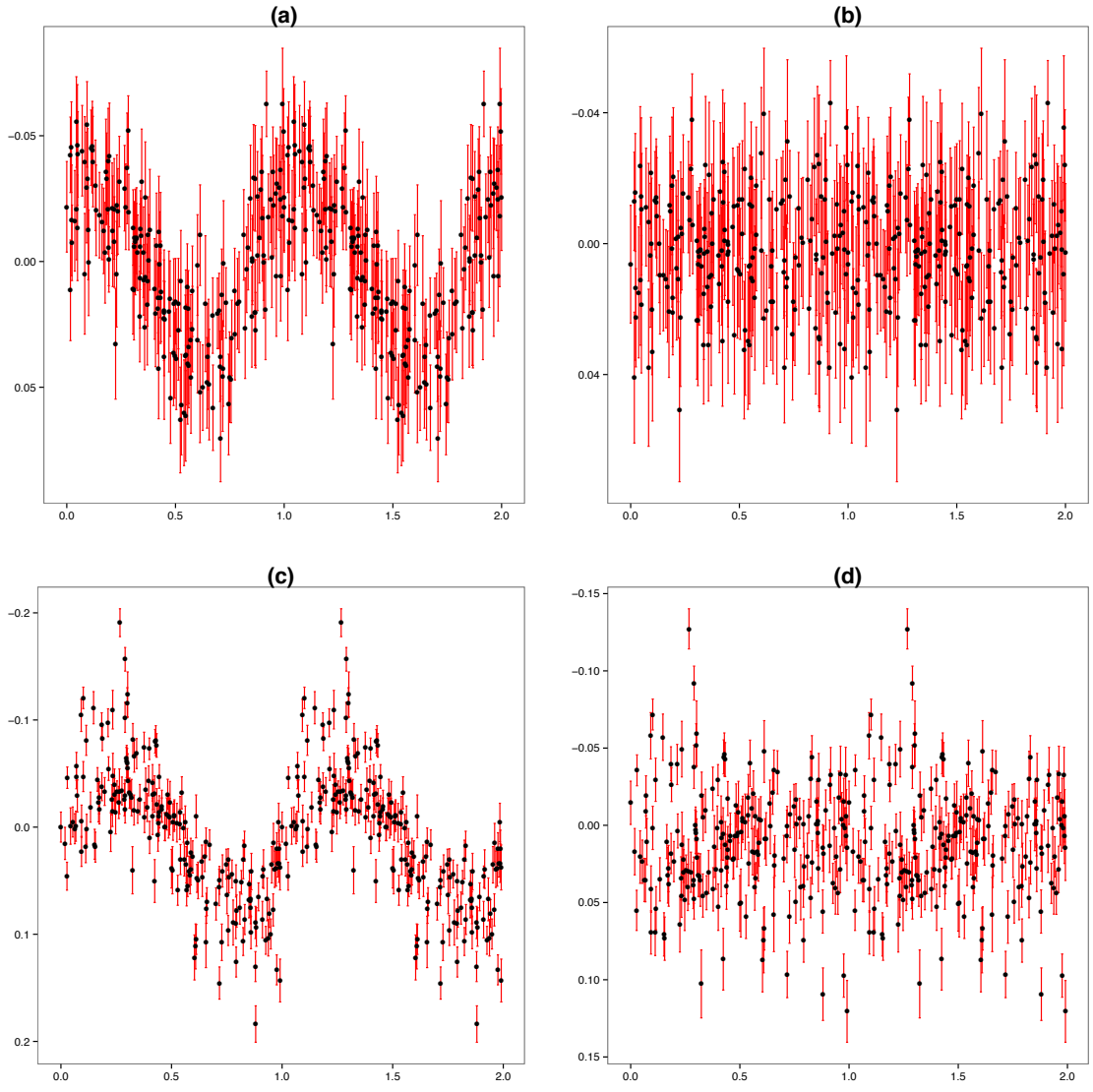


Figure 4.13: In the first column are shown on figures (a) and (c) the residuals after fitting an harmonic model with one period for two double mode Cepheids. On the second column (figures (b) and (d)), the residuals of the same variable stars after fitting an harmonic model with two periods are shown.

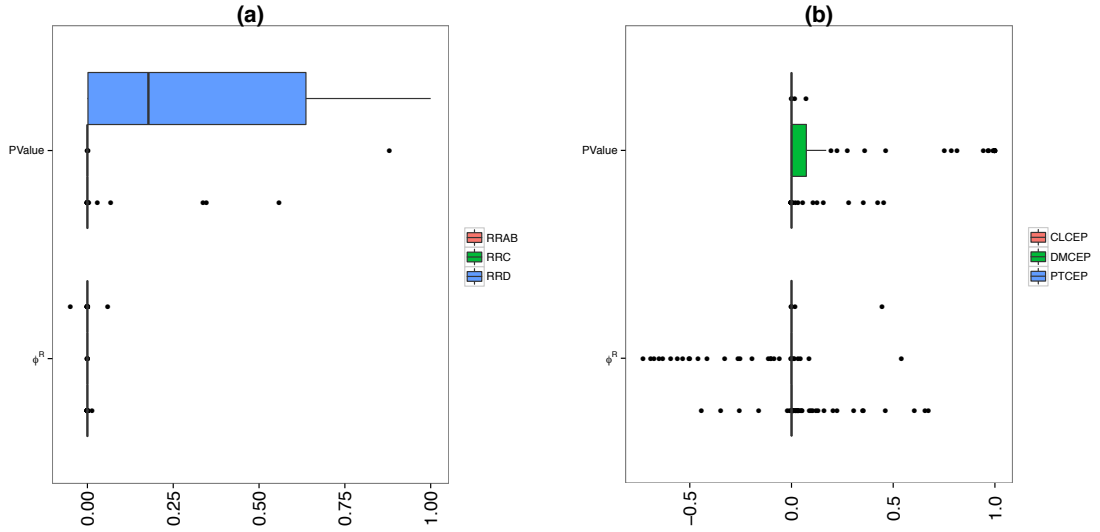


Figure 4.14: a) Boxplot of the ϕ^R and the p-value estimated from the CIAR model in the RR-Lyraes variable stars separated by subclasses. b) Boxplot of the ϕ^R and the p-value estimated from the CIAR model in the Cepheids variable stars separated by subclasses.

This result helps to illustrate that a small coefficient estimated by both irregular models does not necessarily imply that the time series are uncorrelated (as discussed in section 4.3.2). Only the p-value computed reflects the multiperiodic behavior of the RRD and DMCEP classes.

4.3.5 Exoplanet Transit light-curve

So far, all the applications of the irregular time series models developed in this work have been made for the light curves of variable stars. However, for some stars its brightness cannot be seen constantly since they have a planet orbiting around them. If a planet orbiting a star transit in front of it, it will block part of the brightness of the star. The light curve that represents this astronomical phenomenon is generally called as exoplanet transit light-curve.

An exoplanet light curve can be modeled by multiplying the approximately constant flux of the star with the transit signal. Just like the variable stars, the model fitted to the exoplanet transit have the structure $y(t) = g(t, \theta) + \epsilon(t)$. Some differences are that here $y(t)$ represents the logarithm of the brightness magnitude of the star, $g(t, \theta)$ is the sum of a log constant flux and the transiting signal and $\epsilon(t)$ is the independent Gaussian error with zero mean and variance σ^2. It is very common to see that the residuals of this model do

not follow a white noise process.

We test our model in a transit of the exoplanet WASP-6b (Jordan et al, 2013 [42]). They show that the residuals of the fit assuming a Gaussian white noise holds the dependency structure. In Figure 4.15 a) these residuals are shown. To address this, the residuals were adjusted using an ARMA(2,2) model and a $1/f$-model, indicating a long memory time dependency. Both models assume regular times.

In order to fit these residuals with a model that considers the unequally spaced times, the IAR model was used. To detect the temporal dependence in the residuals using this model, the autocorrelation estimator $\hat{\phi}$ and the p-value of the test defined in section 4.3.2 is used. However, this test must be modified since, as the exoplanet transit light curve does not have a periodic behavior, it is not possible to estimate a dominant frequency using the GLS model in this light curve. Therefore, the procedure explained in section 4.3.2 to fit wrongly a light curve cannot be performed.

An alternative is to perform a randomized experiment. This experiment consists in fixing the observation times and shuffle randomly the brightness magnitude a hundred times. To each randomized light curve, the IAR model was fitted in order to estimate the parameter ϕ. Note that the randomized light curve does not have a temporal dependency structure. Likewise to the test described in Section 4.3.2, the ϕ parameter estimated from the residuals is tested whether it belongs to the distribution built using the hundred randomized (and independent) light curves. So, the null hypothesis here is $H_0 : \phi \sim F_1$ vs $H_1 : \phi \nsim F_1$, where F_1 is the distribution of the vector $\underset{\sim}{\hat{\phi}} = \{\hat{\phi}_1, \ldots, \hat{\phi}_{100}\}$, where each $\hat{\phi}_j$ is computed on the $j-th$ randomized light curve.

The p-value computed for the ϕ estimated using the IAR model in the residuals of the fitted model $g(t, \theta)$ is $5.642274e-05$. According with this p-value, these residuals do not have an independent behavior. Therefore, the p-value confirms the existence of temporal dependency on this data. This result is consistent with the results of Jordan et al, 2013 [42]. In Figure 4.15 b) is the distribution of the vector $\underset{\sim}{\hat{\phi}}$, and the red triangle corresponds to $\log(\hat{\phi})$, where $\hat{\phi}$ was computed on the residuals. It can be observed that the $\hat{\phi}$ is greater than the ϕ estimated for the randomized light curves.

4.4 AITS Package in R

In this section, a new package built in the freeware R (R Core Team 2015 [64]) is presented. This package is called **AITS** (**A**nalysis of **I**rregular **T**ime **S**eries). The package

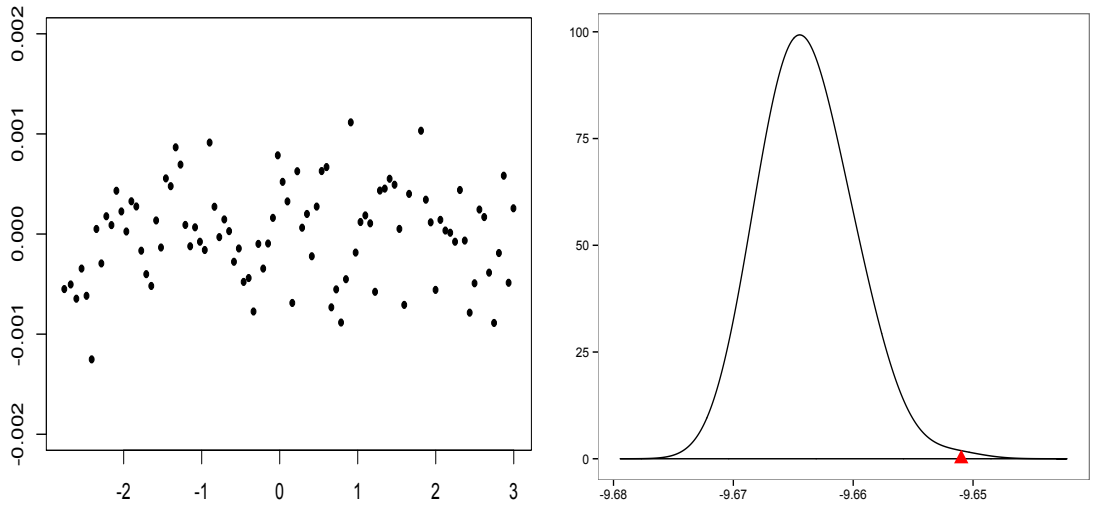


Figure 4.15: (a) Residuals after fitting the model for a transiting exoplanet; (b) The red triangle represents the $\log(\hat{\phi})$, where $\hat{\phi}$ is the parameter of the IAR model. The black line represents the density of the ϕ for the randomized experiment.
.

AITS contains R functions to fit unequally spaced time series from the Irregular Autoregressive (IAR) and the Complex Irregular Autoregressive (CIAR) models. Both models were described previously in this chapter.

This package consists of a total of fifteen functions, five of them for the IAR process, three for the IAR-Gamma Process and four for the CIAR process. The three remaining functions are for generating the irregular times from the mixture of exponential distributions (equation 4.1.11), fitting an harmonic model (equation 3.0.1) and plotting the folded light curve (equation 2.1.1). The functions implemented for the irregular time series models allow to generate observations for each process, fit each model and test the significance of the autocorrelation parameter according with the two tests described previously.

In addition, the package contains four time series which can be used to test the functions. Three of these time series corresponds to light curves of a Classical Cepheid (`clcep`), Delta Scuti (`dscut`) and a Beta Lyrae (`eb`) eclipsing binaries variable stars. These light curves have been used previously in the section 4.3.1 in order to assess the ability of the IAR model to detect the model misspecification. In the R documentation of

each light curve, the frequency estimated by GLS was added. The remaining time series corresponds to the residuals of the fitted exoplanet transit light curve (`Planets`) presented in the section 4.3.5.

4.4.1 The gentime function

The **AITS** package offers several methods to fit temporal data irregularly observed. Furthermore, if you do not have an irregular time series to test these models, the **AITS** package have functions that allow to generate synthetic irregular time series. To do this, first the irregular times must be generated. As mentioned above, in this work it has been proposed to generate the irregular times using a mixture of exponential distributions (equation 4.1.11). The function `gentime` allows to simulate the irregular times. The function can be implemented by the following R command.

```
gentime(n, lambda1 = 130, lambda2 = 6.5, p1 = 0.15, p2 = 0.85)
```

where `n` corresponds to the number of observational times that will be generated. `lambda1` and `lambda2` are the means of each exponential distribution, `p1` and `p2` are its respective weights. The result of this function is an array with the irregularly spaced observations times.

4.4.2 The harmonicfit function

Most of the applications of the irregular time series models that were presented in the previous sections, were performed on light curves of variable stars, which generally have a periodical behavior. If the period of a specific variable star is known, this light curve can be fitted by an harmonic model. An p-harmonic model has been defined previously in the equation (3.0.1). From the function `harmonicfit` this model can be fitted to an irregular time series, using the following R command,

```
harmonicfit(file, f1, nham = 4,weights=NULL,print=FALSE)
```

where `file` is a matrix with two columns, the first of them must have the irregular times and the second the observations. Furthermore, `f1` is the frequency of the time series that will be modeled and `nham` is the number of harmonics that can be used in the fit. The default is 4. In addition, to fit a weighted harmonic model, an array with the weights of each observation must be added in the argument `weights`. Finally, the `print` argument is a boolean. When `print = TRUE` a summary of the harmonic model fitted will be printed. The data `clcep` can be used to test this function. An harmonic model can be fitted to this time series using the follow command,

```
data(clcep)
f1=0.060033386
results=harmonicfit(file=clcep,f1=f1,nham=4)
```

The function `harmonicfit` returns both the residuals of the fitted model and goodness of fit measures, such as the R squared (R2) and the Mean Squared Error (MSE).

4.4.3 The foldlc function

In chapter 2 we mentioned that the light curves of variable stars that have a periodical behavior are generally plotted in its phase. In the phased light curve, the periodic behavior of the brightness a star can be seen much better than in the irregularly measured raw light curve. In equation (2.1.1) the phase of an observation ϕ is defined. The phased light curve also is currently called folded light curve. To make the plot of the folded (phased) light curve with this package, the following code must be used,

```
foldlc(file,f1)
```

where `file` is a matrix with three columns, corresponding to the irregular times, the magnitudes and the measurement errors, and the `f1` is the frequency of the light curve.

The three functions explained above are incorporated into the **AITS** package in order to facilitate the application of the irregular time series models in the light curves of variable stars. The remaining functions are useful to simulate and modeling the irregular time series process. First, the functions that allow to simulate each process are described below.

4.4.4 Simulating the Irregular Time Series Processes

The functions `IAR.sample`, `IARg.sample` and `CIAR.sample` allow to generate observations from the IAR, IAR-Gamma and CIAR process respectively. All these functions work similarly, in the sense that they need as input the length of the generated time series (`n`), the vector with the irregular times (`sT`), which can be generated using the function `gentime`, and the specific parameters of each model. For example, to generate an IAR process the following R command must be used,

```
IAR.sample(phi, n = 100, sT)
```

where `phi` is the value of the autocorrelation parameter of the simulated data. `phi` can take values in the interval (0,1). The R-command to simulate an IAR-Gamma process is,

```
IARg.sample(n, phi, st, sigma2 = 1, mu = 1)
```

as can be seen, the last two commands are very similar. However, the IAR gamma needs to specify two additional parameters according with equation 4.1.10, the scale parameter `sigma2` and the level parameter `mu`. Finally, to generate observations of the CIAR process, the following R command must be used,

```
CIAR.sample(n, sT, phi.R, phi.I, rho = 0, c = 1)
```

where `phi.R` is the real part and `phi.I` is the imaginary part of the complex phi parameter of this model. Both values must be chosen with the condition that $|\phi^R + i\phi^I| < 1$. In addition, it can also be specified the correlation (`rho`) between the real and the imaginary part of the process, and the nuisance parameter `c` related to the variance of the imaginary part. The defaults values are 0 and 1 respectively.

As an example of the use of these functions, the following script generate a IAR process of length 300 with irregular times coming from the mixture of exponential distributions, with parameters $\lambda_1 = 130$, $\lambda_2 = 6.5$, $\omega_1 = 0.15$ and $\omega_2 = 0.85$, and the time dependency parameter $\phi = 0.9$.

```
set.seed(6714)
st<-gentime(n=300)
y<-IAR.sample(phi=0.9,n=300,st)
y<-y$series
plot(st,y,type='l')
rug(st, col = 2)
```

In Figure 4.16, the simulated IAR process are shown. It can be observed a stationary behavior of its mean and variance.

4.4.5 Fitting the Irregular Time Series Processes

The estimation of the three models implemented in the AITS package can be performed by maximum likelihood. However, the estimators of the autocorrelation parameters do not have a closed form, whereby iterative methods must be used. These iterative methods have already been implemented in other packages, for example, the `optimize` function of the `stats` package allows us to find the optimal estimator of the ϕ parameter of the IAR model. Both for the CIAR model and the IAR Gamma it is necessary to optimize more than one parameter. Consequently, the function `nlminb` of the `stats` package allows us to find the optimal solution for these models.

To estimate the IAR model parameter ϕ, the function `IAR.loglik` must be implemented by,

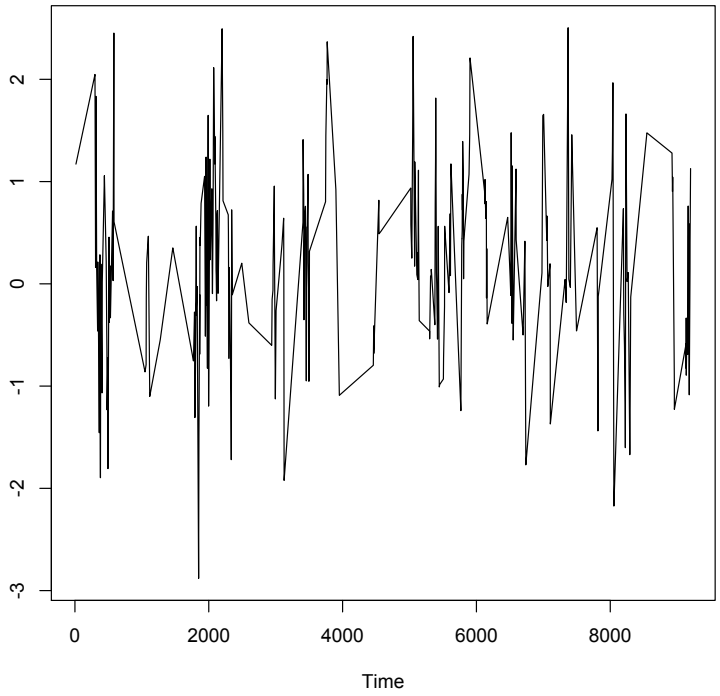


Figure 4.16: Simulated IAR Time Series of length 300 and $\phi = 0.9$. The times was generated by the mixture of exponential distributions with parameters $\lambda_1 = 130$, $\lambda_2 = 6.5$, $\omega_1 = 0.15$ and $\omega_2 = 0.85$.

```
IAR.loglik(y, sT, standarized = "TRUE")
```

where `y` is the array with the values of the sequence, `sT` is the array with the irregular times and the boolean `standarized` must be "TRUE" if the array `y` was standarized. As for example, the ϕ parameter of the IAR process shown in Figure 4.16 will be estimated with the following code.

```
set.seed(6714)
st<-gentime(n=300)
y<-IAR.sample(phi=0.9,n=300,st)
y<-y$series
phi=IAR.loglik(y=y,sT=st)$phi
phi
[1] 0.898135
```

Note that the estimated value was $\hat{\phi} = 0.898$, which was very close to the value with which the IAR process was generated ($\phi = 0.9$). The fitted values of the IAR process and

the maximum likelihood estimator of σ^2 can be obtained with the following code,

```
n=300
d=c(0,diff(st))
phi1=phi**d
yhat=phi1*as.vector(c(0,y[1:(n-1)]))
sigma=var(y)
nu=c(sigma,sigma*(1-phi1**(2))[-1])
tau<-nu/sigma
var.hat=mean(((y-yhat)**2)/tau)
var.hat
[1] 0.9506582
```

where `yhat` is the vector of the fitted values and `var.hat` is the maximum likelihood estimation of the variance of the process.

Similarly to the IAR process, the function `IAR.gamma` allows to estimate the parameters of the IAR-Gamma process using the following command,

```
IAR.gamma(y, sT)
```

where `y` is the array with the values of the sequence and `sT` is the array with the irregular times. In order to test this function, a IAR.gamma process will be generated with the following code,

```
n=300
set.seed(6714)
st<-gentime(n)
y<-IARg.sample(n,phi=0.9,st,sigma2=1,mu=1)
plot(st,y$y,type='l')
rug(st, col = 2)
hist(y$y,breaks=20)
```

In Figure 4.17 a) the IAR.gamma time series is shown. In order to show the asymmetrical behavior of this time series, the histogram of the IAR.gamma observations has been added in figure b).

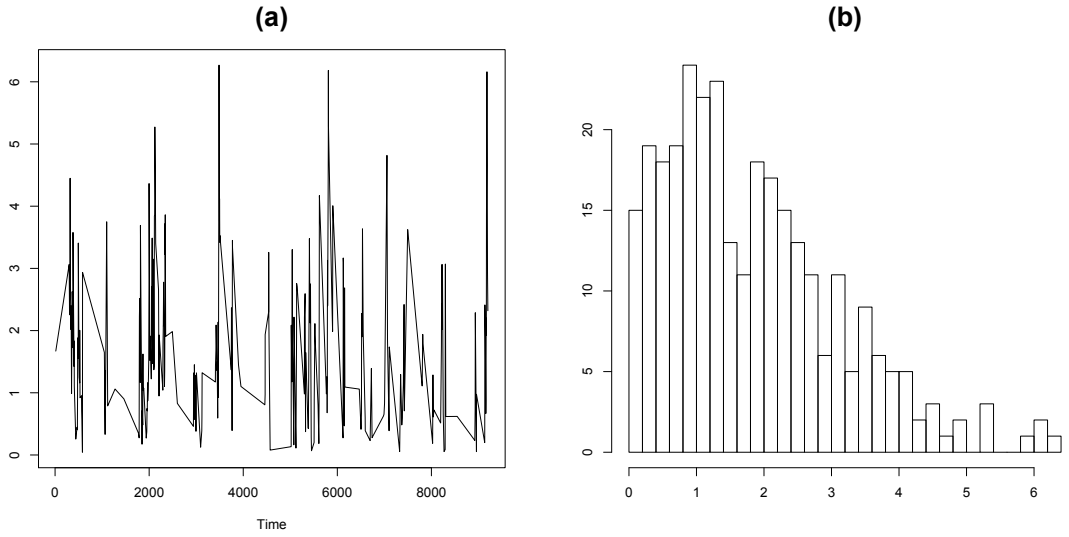


Figure 4.17: Figure a) shows the time series of the Simulated IAR-Gamma Process with length 300 and $\phi = 0.9$. The times was generated by the mixture of exponential distributions with parameters $\lambda_1 = 130$, $\lambda_2 = 6.5$, $\omega_1 = 0.15$ and $\omega_2 = 0.85$. Figure (b) shows the histogram of the IAR-Gamma observations

The maximum likelihood estimation of the IAR-Gamma parameters ϕ, μ y σ (see equation 4.1.10) can be performed on the simulated time series with the following R-command,

```
model<-IAR.gamma(y$y, sT=st)
phi=model$phi
muest=model$mu
sigmaest=model$sigma
phi
[1] 0.8990846
muest
[1] 0.9349599
sigmaest
[1] 0.9854233
```

Note that the estimation of the three parameters $\hat{\phi}$, $\hat{\mu}$, $\hat{\sigma}$ of the IAR-gamma process was very accurate, taking values 0.899, 0.934 and 0.985 respectively.

Finally, the last estimation procedure that will be reviewed is the corresponding to the CIAR model. The parameters of this model can be estimated using the function

`CIAR.Kalman`. This function can be called using the following R-command,

```
CIAR.kalman(y, t, standarized = "TRUE", c = 1, niter = 10, seed =
                                    1234)
```

where `y` is the array with the values of the sequence corresponding to the real part of the complex process, `sT` is the array with the irregular times, `standarized` is a boolean which must be "TRUE" if the array `y` is standarized and `c` is the value of the nuisance parameter equivalent to the variance of the imaginary part of the process. In addition, this function uses the R function `nlminb` to find the optimal maximum likelihood estimators using the Kalman Filter. Using this procedure, the solution found is not always optimal. Therefore, we add the parameter `niter` equivalent to the times that the estimation procedure will be repeated in order to find the optimal estimators. The default value is niter = 10. Finally, a `seed` parameter is specified in order to remove the randomness of the optimal values.

As in the previous examples, this function will be tested in a simulated CIAR process, which can be generated using the following R-command,

```
n=300
set.seed(6714)
st<-gentime(n)
x=CIAR.sample(n=n,phi.R=0.9,phi.I=0,sT=st,c=1)
plot(st,x$y,type='l')
```

In Figure 4.18 is the time series of the real part of the CIAR process generated. To find the maximum likelihood estimators of the parameters of this model, the following code must be used,

```
options(digits=4)
y=x$y
y1=y/sd(y)
ciar=CIAR.kalman(y=y1,t=st)
ciar
[1]  9.108e-01 -9.683e-10
Mod(complex(real=ciar[1],imaginary=ciar[2]))
[1] 0.9108
```

Note that the estimated parameters are $\widehat{\phi^R} = 0.91$ and $\widehat{\phi^I} \approx 0$, which are very closed to the real values of these parameters used to simulate the sequence.

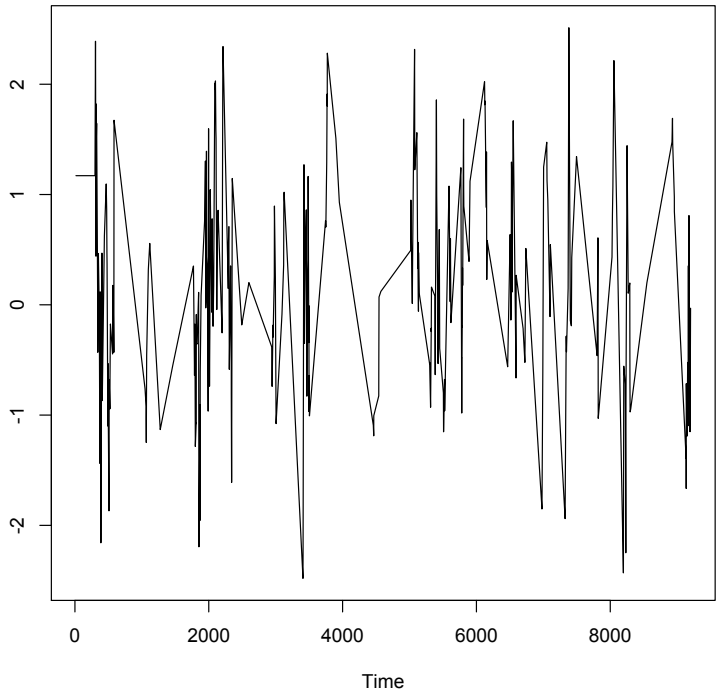


Figure 4.18: Real part of the simulated CIAR process of length 300, $\phi^R = 0.9$, $\phi^I = 0.9$ and the nuisance parameter $c = 1$. The times was generated by the mixture of exponential distributions with parameters $\lambda_1 = 130$, $\lambda_2 = 6.5$, $\omega_1 = 0.15$ and $\omega_2 = 0.85$.

4.4.6 Testing the significance of the parameters of the irregular models

The construction of two statistical test to assess the significance of the autocorrelation parameters was shown previously. The formulated tests differ in that the first of them assumes that the time series have a periodical behavior which can be modeled by an harmonic model. The main idea of this test is to verify whether the harmonic model explain all the time dependency structure in the time series or not. If not, a time dependency structure should remain in the residual of the harmonic fit.

To assess the significance of the autocorrelation parameter, this test uses the dominant frequency (which can be found by GLS (2.2.25)). This frequency is used to fit an harmonic model to the time series. Later, the residuals of the harmonic fitted model are modeled by the irregular time series. The parameter estimated ϕ can be used as an autocorrelation index. However, as mentioned previously a small value of $\hat{\phi}$ does not nec-

essarily mean that there is no temporal dependence on the time series, since this may be due to the dependence between the frequency and the phi value discussed in section 4.3.2. To verify whether the residuals are uncorrelated or not, the test fit an harmonic model to the raw time series using now a percentual variation of the correct frequency. As these models are fitted using a wrong period, $\hat{\phi}$ must have greater values regarding to the ones obtained in the residuals of the correct fitted model.

To perform this test in the **AITS** package for the ϕ parameter estimated by the IAR model, the following command must be used,

```
IAR.Test(y, sT, f, phi, plot = "TRUE", xlim = c(-1, 0))
```

where `y` is the array with the time series observations and `sT` is the array with the irregular observational times. In addition, the dominant frequency `f` and the ϕ parameter estimated by the IAR model (`IAR.loglik`) are needed as input. The argument `plot` is logical, if it is true, the function returns a density plot of the distribution of the $\hat{\phi}$ estimated in the residuals of the wrongly fitted models. The argument `xlim` only works if `plot = "TRUE"`, and define the limits of the x axis. The data `clcep` of this package can be used to exemplify the use of this function. With the following code the example of this test for the IAR model can be run,

```
data(clcep)
f1=0.060033386
results=harmonicfit(file=clcep,f1=f1)
y=results$res/sqrt(var(results$res))
sT=results$t
res3=IAR.loglik(y,sT,standarized='TRUE')
require(ggplot2)
test<-IAR.Test(y=clcep[,2],sT=clcep[,1],f1,res3$phi,plot='TRUE',xlim=c(-10,0.5))
test
```

In this example the ϕ estimated by the IAR model is $\hat{\phi} = 6.67e-05$ and the p-value of the test is 0. According with the hypothesis of the test defined in the section 4.3.2, the ϕ estimated value is not significative. Therefore, the residuals of the harmonic fit do not have a time dependency structure. In Figure 4.19 a) it is shown the density plot returned for the function `IAR.Test`. This plot has the density of the $log(\phi)$ estimated in the residuals when the time series was fitted wrongly, and the red triangle is the $log(\phi)$ of the "correct" ϕ estimation. Evidently, this point does not belong to the distribution of the "wrong" estimations.

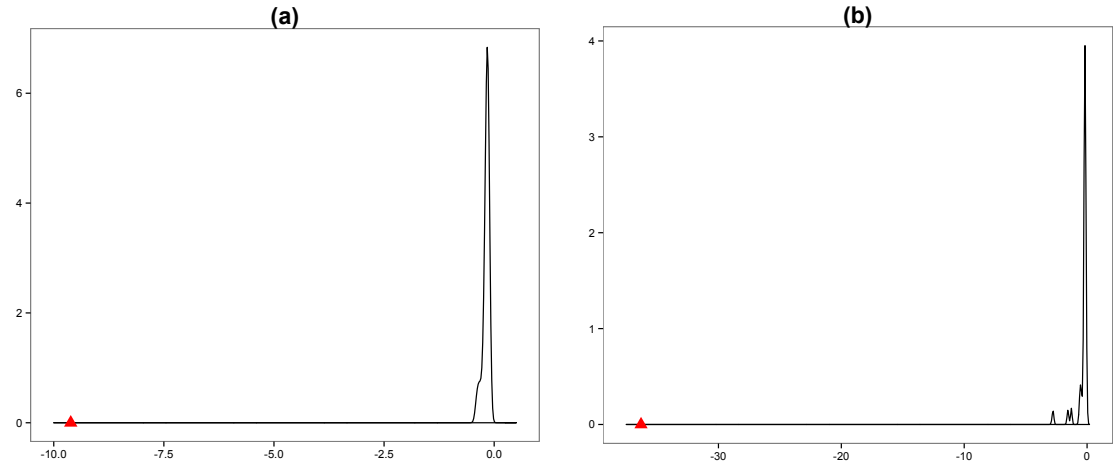


Figure 4.19: Figure a) shows the Density Plot of the $\log(\phi)$, where ϕ was estimated by the IAR model when the time series is fitted using wrong periods. The red triangle is the $\log(\phi)$ estimated when the correct period is used in the harmonic fit. Figure (b) shows the same plot for the CIAR process.

For the CIAR process, the significance of the parameter ϕ^R can be assessed following the same procedure of the IAR.Test. To perform a test for the CIAR process, the following R-command must be used,

```
CIAR.Test(y, sT, c = 1, f, phi, plot = "TRUE", xlim = c(-1, 0), Mod =
                                "False")
```

The arguments of this function are the same of the `IAR.Test` function, with the difference of the argument `c` corresponding to the nuisance parameter of the CIAR model and the argument `Mod` which is a boolean. When `Mod=''False''` the significance of the parameter ϕ^R is assessed. When `Mod=''True''` the significance of the $|\phi^R + i\phi^I|$ is assessed. To perform this test using the same data of the previous example, the following code must be used,

```
data(clcep)
f1=0.060033386
results=harmonicfit(file=clcep,f1=f1)
y=results$res/sqrt(var(results$res))
sT=results$t
res3=CIAR.kalman(y,sT,standarized='TRUE')
res3$phiR
require(ggplot2)
test<-CIAR.Test(y=clcep[,2],sT=clcep[,1],f=f1,phi=res3$phiR,plot='TRUE')
test
```

cxxx

In Figure 4.19 b) it is shown the density plot returned for this function, as can be seen the result of the test of the CIAR process is consistent with the result obtained by the IAR model previously.

As mentioned previously, in this package there are two different kind of test. The second one, do not assumes a periodical behavior of the time series. Therefore, this time series is not able to fit an harmonic model. The main idea of this test is shuffling the time series several times, generating in each case independent samples of the original time series or breaking the time dependency. The ϕ estimated in the "independent" time series must be less than the ϕ estimated in the raw time series. To perform this test with the **AITS** package the following code must be used,

```
IAR.Test2(y, sT, iter = 100, phi, plot = "TRUE", xlim = c(-1, 0))
```

This function works with the same arguments as the `IAR.Test` function, with the exception that it does not require knowing the frequency of the time series. Instead, the function `IAR.Test2` uses the argument `iter` to define the number of independent time series that will be used to create the distribution of $\log(\phi)$. In order to exemplify the use of this function, the code used in the application on the light curve of an exoplanet transit described in section 4.3.5 will be shown below,

```
data(Planets)
t<-Planets[,1]
res<-Planets[,2]
y=res/sqrt(var(res))
res3=IAR.loglik(y,t,standarized='TRUE')[1]
res3$phi
set.seed(6713)
require(ggplot2)
test<-IAR.Test2(y=y,sT=t,phi=res3$phi,plot='TRUE',xlim=c(-9.6,-9.45))
test
```

As mentioned above, the data `Planets` is also in the package `AITS` and corresponds to the residuals of the fitted model by Jordan et al,2013 [42] in an exoplanet transit. In this example, the p-value was ≈ 1, therefore the null hypothesis was accepted. Consequently, it is confirmed that in the Planets time series there is no structure of temporal correlation.

Chapter 5

Discussion

In this work, several tools for the modeling and classification of the light curves have been presented under a solid statistical framework. First, the procedure of a machine learned classifier for RR-Lyrae type ab stars of the VVV survey has been detailed. This classifier was built following eight key steps from the pre-processing of the light curves to the implementation of the data mining algorithms. Throughout this procedure, relevant decisions were taken regarding the photometry aperture, the classification algorithm and the features used.

The best performance among the state-of-the-art data mining algorithms implemented in this work was achieved by the AdaBoost Classifier. This is an interesting result, as the classifiers used for variable stars are usually built from Random Forest. In 3.4 I have found that the Adaboost is consistently better than the Random Forest for this training set. In addition, I have noticed that the Adaboost algorithm is more stable to feature selection than the Random Forest which is sensitive to the quality of the features used.

In addition, the photometry aperture was selected using a method based on a Kernel Density Classifier. Generally, the aperture selection is done by selecting the aperture with the minimum error. However, using this method of aperture selection the performance of the classifier is significantly lower than the reached using the Kernel Selection Method.

In the classification procedure it has also been proposed to make a selection of the more important features for the classifier. This procedure has been rarely used in the literature. However, I have noticed that some classification algorithms are sensitive to the presence of poor quality features. In this work we have chosen the set of features that maximize the F1-measure. I have noticed that 12 features were enough to achieve the best performance of the classifier. As expected, the most important feature was the period, since the periods of RR*ab* take values in a well-known range. In addition, most of the important features come from the harmonic fitted model.

Finally, the performance of the chosen classifier estimated by cross-validation on the training set achieves an F1-Measure of ≈ 0.93 using a score threshold of 0.548. This classifier has a lower performance than the one obtained by Richards et al. (2012) Richards et al. [57] for ASAS, which achieved an F1-Measure of ≈ 0.96. However, the results are not necessarily comparable since our classifier was performed using NIR data, whereas Richards uses data from the optical. As mentioned in 3.3.1, building a classifier in the VVV is more difficult due to the quality of the NIR light curves.

The classifier built in this work has been used in different fields of the VVV. In the globular clusters 2MASS- GC 02 and Terzan 10, the classifier reached an harmonic mean between false positive and negatives of the 4.4% of the data, which are consistent with the performance of the classifier in the training set. In the outer bulge (fields b201–b228) region, the classifier also had a performance consistent with the training, reaching an harmonic mean between false positive and negatives of the 8% of the data. Furthermore, the classifier also helped to confirm some RR*ab* of the outer bulge with a more symmetrical behavior. Finally, the classifier was used to perform a census of RR*ab* along the southern galactic disk (fields b001–b151). After calibrating the classification threshold, Dekany et al. (2018) [25] found 2147 RR*ab* candidates.

In addition to the classifier construction, in this thesis I worked also on providing alternative models to fit irregular time series, as the light curves of astronomical data. The main aim is to provide a more flexible representation of the CAR(1) model, which are currently used for this purpose. To achieve this goal, the first model proposed is a discrete representation of the CAR(1) model called the Irregular Autoregressive Model (IAR). Unlike the CAR(1) model, the irregular autoregressive model allows for Gaussian and non-Gaussian distributed data. Furthermore, it has been proven that the IAR model is strictly stationary and ergodic under conventional conditions. Finally, we propose a maximum likelihood estimation procedure for the parameters of the model.

The IAR model are strongly connected with both the AR and CAR models. Particularly, the IAR model is an extension of the regular autoregressive model of order 1 by assuming irregular times. In addition, the IAR process is also equivalent to the CAR(1) model by assuming Gaussian data. However, the CAR(1) model have a low performance in the fit of non-Gaussian data. This was verified for a Gamma distributed IAR process developed in this work.

A drawback that both the IAR model and the CAR(1) have, is that they only allow to estimate positive autocorrelation. In order to estimate negative autocorrelation, an extension of the IAR model has been proposed. This model is called the Complex Irregular

Autoregressive Model (CIAR), which are characterized by allowing complex coefficients. It has been proven that this model is weakly stationary, and its state-space representation is stable, under some conditions. Like the IAR model, the parameters of the CIAR model are estimated following a maximum likelihood procedure based on the Kalman recursion performed on the state-space representation of the CIAR model. Finally, by assuming a null imaginary part of the autocorrelation parameter, we come back to the IAR process.

Both models proposed in this work were implemented on the light curves of variable stars observed by OGLE and HIPPARCOS surveys. Two applications of these models have been illustrated on this data. First, to identify whether the light curve corresponds to a multiperiodic variable stars. This can be achieved by estimating a significant autocorrelation using these models on the residuals of the harmonic model fitted on the light curve using only the dominant period. The second application is to detect whether a parametric model was misspecified. This application has been tested both in an astrophysical model fitted on a planet and in the harmonic model fitted in the light curves of variable stars.

After fitting both models to the light curves, we have noticed a strong relationship between the IAR and CIAR model estimates, when the coefficient of the CIAR model is positive. However, we have also found several cases of negatively correlated light curves that the IAR model has ignored. In addition, we have shown illustrative examples where the inability to estimate negative correlations of the IAR model prevents finding multiperiodic variable stars or correctly detect whether the harmonic model was misspecified.

In the analysis it was shown that the estimated autocorrelation by both models depend on the frequency used in the harmonic model. This dependency can affect the interpretation of the estimated coefficients. In order to correctly distinguish between a coefficient that indicates significative correlation and another that indicates the opposite, a statistical test has been developed for both models. This test was assessed in forty variable stars selected due to the good harmonic fit obtained in these light curves. For both models, the p-value estimated by the test was consistent with good fit of the selected curves. In addition, we have shown that the p-values obtained from the test proposed in this work for the CIAR model are useful for characterizing the multiperiodic classes of RR-Lyraes (RRD) and Cepheids (DMCEP). This result indicates that the p-value can be an important feature for a machine learned classifier implemented to these classes.

Both IAR and CIAR models are capable to fit irregular time series that have an exponential decay behavior in the autocorrelation function. Another class of irregular time series that have an persistent (antipersisent) behavior can be fitted by the CARFIMA model. For these time series, the CIAR process can be used to identify the sign of the autocorrelation function.

Finally, the package Analysis of Irregular Time Series was developed in R. This package allows to implement both the IAR and CIAR models. The functions of the package allow to simulate each process, estimate its parameters and to assess the significance of the estimated coefficients. The accuracy of the estimation procedures implemented was confirmed using Monte Carlo simulations.

5.1 Future Works

The work developed in this thesis opens up new challenges for future work. One of the most important of them is the construction of a classifier for Cepheids stars. Like the RR*ab*, the Cepheids are pulsating stars essential to build the three-dimensional map of the Galactic bulge. The main idea is to find unknown Cepheids in the VVV disk area. As mentioned in section 3.5.3 to implement a classifier in the VVV disk offers more challenges, due to the small number of observations that light curves have. Furthermore, it is also interesting to classify between type 1 and type 2 Cepheids. An important challenge is to improve the performance of the classifier by finding features that can distinguish between the subclasses of Cepheids.

Regarding the light curves modeling, the discrete representation of irregular time series models has been very useful in terms to extend the regular models to the irregular case. Both the IAR and CIAR models have desirable properties which other models for irregular time series do not have. In this sense, we are currently working to propose new extensions of the very well-known ARMA models in order to relax the regular sampling assumption. In addition, an interesting work for future is to use the autocorrelation coefficients estimated by both models in the next machine learned classifiers implemented.

Since the upcoming astronomical surveys will have continuous stream of data, we start to study methods that allows us to analyze the data while the information is coming in optimal way. The online learning algorithms are characterized by process each training instance once “on arrival” without the need for storage and reprocessing.

Bibliography

[1] Esteban Alfaro, Matías Gámez, and Noelia García. adabag: An R package for classification with boosting and bagging. *Journal of Statistical Software*, 54(2):1–35, 2013. URL http://www.jstatsoft.org/v54/i02/.

[2] J. Alonso-García, I. Dékány, M. Catelan, R. Contreras Ramos, F. Gran, P. Amigo, P. Leyton, and D. Minniti. Variable Stars in the VVV Globular Clusters. I. 2MASS-GC 02 and Terzan 10. , 149:99, March 2015. doi: 10.1088/0004-6256/149/3/99.

[3] Y. Alperovich, M. Alperovich, and A. Spiro. Trends modeling and its impact on hurst exponent at stock market fractal analysis. In *2017 Tenth International Conference Management of Large-Scale System Development (MLSD)*, pages 1–4, Oct 2017. doi: 10.1109/MLSD.2017.8109590.

[4] R. Angeloni, R. Contreras Ramos, M. Catelan, I. Dékány, F. Gran, J. Alonso-García, M. Hempel, C. Navarrete, H. Andrews, A. Aparicio, J. C. Beamín, C. Berger, J. Borissova, C. Contreras Peña, A. Cunial, R. de Grijs, N. Espinoza, S. Eyheramendy, C. E. Ferreira Lopes, M. Fiaschi, G. Hajdu, J. Han, K. G. Hełminiak, A. Hempel, S. L. Hidalgo, Y. Ita, Y.-B. Jeon, A. Jordán, J. Kwon, J. T. Lee, E. L. Martín, N. Masetti, N. Matsunaga, A. P. Milone, D. Minniti, L. Morelli, F. Murgas, T. Nagayama, C. Navarro, P. Ochner, P. Pérez, K. Pichara, A. Rojas-Arriagada, J. Roquette, R. K. Saito, A. Siviero, J. Sohn, H.-I. Sung, M. Tamura, R. Tata, L. Tomasella, B. Townsend, and P. Whitelock. The VVV Templates Project Towards an automated classification of VVV light-curves. I. Building a database of stellar variability in the near-infrared. , 567:A100, July 2014. doi: 10.1051/0004-6361/201423904.

[5] M. Ausloos and K. Ivanova. Power-law correlations in the southern-oscillation-index fluctuations characterizing el Niño. *Phys. Rev. E*, 63:047201, Mar 2001. doi: 10.1103/PhysRevE.63.047201. URL https://link.aps.org/doi/10.1103/PhysRevE.63.047201.

[6] J. Belcher, J. S. Hampton, and G. Tunnicliffe Wilson. Parameterization of continuous time autoregressive models for irregularly sampled time series data. *Journal*

of the Royal Statistical Society. Series B (Methodological), 56(1):141–155, 1994. ISSN Belcher. URL http://www.jstor.org/stable/2346034.

[7] Pascal Bondon and Wilfredo Palma. A class of antipersistent processes. *Journal of Time Series Analysis*, 28(2):261–273, 2007. doi: 10.1111/j.1467-9892.2006.00509.x. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-9892.2006.00509.x.

[8] Leo Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, Aug 1996. ISSN 1573-0565. doi: 10.1007/BF00058655. URL https://doi.org/10.1007/BF00058655.

[9] Leo Breiman. Arcing classifier (with discussion and a rejoinder by the author). *Ann. Statist.*, 26(3):801–849, 06 1998. doi: 10.1214/aos/1024691079. URL https://doi.org/10.1214/aos/1024691079.

[10] Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, Oct 2001. ISSN 1573-0565. doi: 10.1023/A:1010933404324. URL https://doi.org/10.1023/A:1010933404324.

[11] P.J. Brockwell and R.A. Davis. *Time Series: Theory and Methods: Theory and Methods*. Springer Series in Statistics. Springer New York, 1991. ISBN 9780387974293. doi: 10.1007/978-1-4419-0320-4.

[12] P.J. Brockwell and R.A. Davis. *Introduction to Time Series and Forecasting*. Springer-Verlag New York, 2002. ISBN 9780387216577. doi: 10.1007/b97391.

[13] Piet M. T. Broersen. *Automatic Autocorrelation and Spectral Analysis*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006. ISBN 1846283280.

[14] John Y. Campbell, Andrew Wen-Chuan Lo, and Archie Craig MacKinlay. *The Econometrics of Financial Markets*. princeton University press, 1997. URL http://press.princeton.edu/titles/5904.html.

[15] J. M. Carpenter, L. A. Hillenbrand, and M. F. Skrutskie. Near-Infrared Photometric Variability of Stars toward the Orion A Molecular Cloud. , 121:3160–3190, June 2001. doi: 10.1086/321086.

[16] M. Catelan and H. A. Smith. *Pulsating Stars (Wiley-CVH)*. Wiley-VCH, March 2015.

[17] M. Catelan, D. Minniti, P. W. Lucas, I. Dékány, R. K. Saito, R. Angeloni, J. Alonso-García, M. Hempel, K. Helminiak, A. Jordán, R. Contreras Ramos, C. Navarrete, J. C. Beamín, A. F. Rojas, F. Gran, C. E. Ferreira Lopes, C. Contreras Peña, E. Kerins, L. Huckvale, M. Rejkuba, R. Cohen, F. Mauro, J. Borissova, P. Amigo,

S. Eyheramendy, K. Pichara, N. Espinoza, C. Navarro, G. Hajdu, D. N. Calderón Espinoza, G. A. Muro, H. Andrews, V. Motta, R. Kurtev, J. P. Emerson, C. Moni Bidin, and A.-N. Chené. Stellar Variability in the VVV survey. *ArXiv e-prints*, October 2013.

[18] M. Catelan, I. Dekany, M. Hempel, and D. Minniti. Stellar Variability in the VVV Survey: An Update. *ArXiv e-prints*, June 2014.

[19] K. S. Chan and H. Tong. A note on embedding a discrete parameter arma model in a continuous parameter arma model. *Journal of Time Series Analysis*, 8(3):277–281, 1987. doi: 10.1111/j.1467-9892.1987.tb00439.x. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-9892.1987.tb00439.x.

[20] Jennifer S Conrad, Allaudeen Hameed, and Cathy Niden. Volume and autocovariances in short-horizon individual security returns. *Journal of Finance*, 49(4):1305–29, 1994. URL https://EconPapers.repec.org/RePEc:bla:jfinan:v:49:y:1994:i:4:p:1305-29.

[21] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, September 1995. ISSN 0885-6125. doi: 10.1023/A:1022627411411. URL https://doi.org/10.1023/A:1022627411411.

[22] Mark Culp, Kjell Johnson, and George Michailides. ada: An r package for stochastic boosting. *Journal of Statistical Software, Articles*, 17(2):1–27, 2006. ISSN 1548-7660. doi: 10.18637/jss.v017.i02. URL https://www.jstatsoft.org/v017/i02.

[23] J. Debosscher, L. M. Sarro, C. Aerts, J. Cuypers, B. Vandenbussche, R. Garrido, and E. Solano. Automated supervised classification of variable stars. I. Methodology. , 475:1159–1183, December 2007. doi: 10.1051/0004-6361:20077638.

[24] I. Dékány, D. Minniti, G. Hajdu, J. Alonso-García, M. Hempel, T. Palma, M. Catelan, W. Gieren, and D. Majaess. Discovery of a Pair of Classical Cepheids in an Invisible Cluster Beyond the Galactic Bulge. , 799:L11, January 2015. doi: 10.1088/2041-8205/799/1/L11.

[25] István Dékány, Gergely Hajdu, Eva K. Grebel, Márcio Catelan, Felipe Elorrieta, Susana Eyheramendy, Daniel Majaess, and Andrés Jordán. A near-infrared rr lyrae census along the southern galactic plane: The milky way's stellar fossil brought to light. *The Astrophysical Journal*, 857(1):54, 2018. URL http://stacks.iop.org/0004-637X/857/i=1/a=54.

[26] P. Dubath, L. Rimoldini, M. Süveges, J. Blomme, M. López, L. M. Sarro, J. De Ridder, J. Cuypers, L. Guy, I. Lecoeur, K. Nienartowicz, A. Jan, M. Beck, N. Mowlavi, P. De Cat, T. Lebzelter, and L. Eyer. Random forest automated supervised classification of Hipparcos periodic variable stars. , 414:2602–2617, July 2011. doi: 10.1111/j.1365-2966.2011.18575.x.

[27] F. Elorrieta, S. Eyheramendy, A. Jordán, I. Dékány, M. Catelan, R. Angeloni, J. Alonso-García, R. Contreras-Ramos, F. Gran, G. Hajdu, N. Espinoza, R. K. Saito, and D. Minniti. A machine learned classifier for RR Lyrae in the VVV survey. , 595:A82, November 2016. doi: 10.1051/0004-6361/201628700.

[28] ESA, editor. *The HIPPARCOS and TYCHO catalogues. Astrometric and photometric star catalogues derived from the ESA HIPPARCOS Space Astrometry Mission*, volume 1200 of *ESA Special Publication*, 1997.

[29] Laurent Eyer and Nami Mowlavi. Variable stars across the observational hr diagram. *Journal of Physics: Conference Series*, 118(1):012010, 2008. URL http://stacks.iop.org/1742-6596/118/i=1/a=012010.

[30] Susana Eyheramendy, Felipe Elorrieta, and Wilfredo Palma. An autoregressive model for irregular time series of variable stars. *Proceedings of the International Astronomical Union*, 12(S325):259262, 2016. doi: 10.1017/S1743921317000448.

[31] Susana Eyheramendy, Felipe Elorrieta, and Wilfredo Palma. An irregular discrete time series model to identify residuals with autocorrelation in astronomical light curves. *Monthly Notices of the Royal Astronomical Society*, 481(4):4311–4322, 2018. doi: 10.1093/mnras/sty2487. URL http://dx.doi.org/10.1093/mnras/sty2487.

[32] Yoav Freund, Robert E Schapire, et al. Experiments with a new boosting algorithm. In *ICML*, volume 96, pages 148–156, 1996.

[33] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors). *Ann. Statist.*, 28(2):337–407, 04 2000. doi: 10.1214/aos/1016218223. URL https://doi.org/10.1214/aos/1016218223.

[34] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33 (1):1–22, 2010. URL http://www.jstatsoft.org/v33/i01/.

[35] Jerome H. Friedman. Greedy function approximation: A gradient boosting machine. *Ann. Statist.*, 29(5):1189–1232, 10 2001. doi: 10.1214/aos/1013203451. URL https://doi.org/10.1214/aos/1013203451.

[36] Jianbo Gao, Yinhe Cao, Wen-wen Tung, and Jing Hu. *Multiscale Analysis of Complex Time Series: Integration of Chaos and Random Fractal Theory, and Beyond*. Wiley-Interscience, 2007. ISBN 0471654701.

[37] Gran, F., Minniti, D., Saito, R. K., Zoccali, M., Gonzalez, O. A., Navarrete, C., Catelan, M., Contreras Ramos, R., Elorrieta, F., Eyheramendy, S., and Jordán, A. Mapping the outer bulge with rrab stars from the vvv survey. *AA*, 591:A145, 2016. doi: 10.1051/0004-6361/201527511. URL https://doi.org/10.1051/0004-6361/201527511.

[38] Trevor J. Hastie, Robert John Tibshirani, and Jerome H. Friedman. *The elements of statistical learning : data mining, inference, and prediction*. Springer series in statistics. Springer, New York, 2009. ISBN 978-0-387-84857-0. URL http://opac.inria.fr/record=b1127878. Autres impressions : 2011 (corr.), 2013 (7e corr.).

[39] M. J. Irwin, J. Lewis, S. Hodgkin, P. Bunclark, D. Evans, R. McMahon, J. P. Emerson, M. Stewart, and S. Beard. VISTA data flow system: pipeline processing for WFCAM and VISTA. In P. J. Quinn and A. Bridger, editors, *Optimizing Scientific Return for Astronomy through Information Technologies*, volume 5493 of *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, pages 411–422, September 2004. doi: 10.1117/12.551449.

[40] Z. Ivezic, J. A. Tyson, T. Axelrod, D. Burke, C. F. Claver, K. H. Cook, S. M. Kahn, R. H. Lupton, D. G. Monet, P. A. Pinto, M. A. Strauss, C. W. Stubbs, L. Jones, A. Saha, R. Scranton, C. Smith, and LSST Collaboration. LSST: From Science Drivers To Reference Design And Anticipated Data Products. In *American Astronomical Society Meeting Abstracts #213*, volume 41 of *Bulletin of the American Astronomical Society*, page 366, January 2009.

[41] R. H. Jones. Fitting a continuous time autoregression to discrete data. *Applied Time Series Analysis*, pages 651–682, 1981. URL http://ci.nii.ac.jp/naid/10030486300/en/.

[42] A. Jordán, N. Espinoza, M. Rabus, S. Eyheramendy, D. K. Sing, J.-M. Désert, G. Á. Bakos, J. J. Fortney, M. López-Morales, P. F. L. Maxted, A. H. M. J. Triaud, and A. Szentgyorgyi. A Ground-based Optical Transmission Spectrum of WASP-6b. , 778:184, December 2013. doi: 10.1088/0004-637X/778/2/184.

[43] Brandon C. Kelly, Andrew C. Becker, Malgosia Sobolewska, Aneta Siemiginowska, and Phil Uttley. Flexible and scalable methods for quantifying stochastic variability in the era of massive time-domain astronomical data sets. *The Astrophysical Journal*, 788(1):33, 2014. URL http://stacks.iop.org/0004-637X/788/i=1/a=33.

[44] Kim, Dae-Won and Bailer-Jones, Coryn A. L. A package for the automated classification of periodic variable stars. *AA*, 587:A18, 2016. doi: 10.1051/0004-6361/201527188. URL https://doi.org/10.1051/0004-6361/201527188.

[45] N. R. Lomb. Least-squares frequency analysis of unequally spaced data. , 39:447–462, February 1976. doi: 10.1007/BF00648343.

[46] N. Matsunaga. Time-series surveys and pulsating stars: The near-infrared perspective. In *European Physical Journal Web of Conferences*, volume 152 of *European Physical Journal Web of Conferences*, page 01007, September 2017. doi: 10.1051/epjconf/201715201007.

[47] David Meyer and Technische Universität Wien. Support vector machines. the interface to libsvm in package e1071. online-documentation of the package e1071 for quot;r, 2001.

[48] D. Minniti, P. W. Lucas, J. P. Emerson, R. K. Saito, M. Hempel, P. Pietrukowicz, A. V. Ahumada, M. V. Alonso, J. Alonso-Garcia, J. I. Arias, R. M. Bandyopadhyay, R. H. Barbá, B. Barbuy, L. R. Bedin, E. Bica, J. Borissova, L. Bronfman, G. Carraro, M. Catelan, J. J. Clariá, N. Cross, R. de Grijs, I. Dékány, J. E. Drew, C. Fariña, C. Feinstein, E. Fernández Lajús, R. C. Gamen, D. Geisler, W. Gieren, B. Goldman, O. A. Gonzalez, G. Gunthardt, S. Gurovich, N. C. Hambly, M. J. Irwin, V. D. Ivanov, A. Jordán, E. Kerins, K. Kinemuchi, R. Kurtev, M. López-Corredoira, T. Maccarone, N. Masetti, D. Merlo, M. Messineo, I. F. Mirabel, L. Monaco, L. Morelli, N. Padilla, T. Palma, M. C. Parisi, G. Pignata, M. Rejkuba, A. Roman-Lopes, S. E. Sale, M. R. Schreiber, A. C. Schröder, M. Smith, L. S. , Jr., M. Soto, M. Tamura, C. Tappert, M. A. Thompson, I. Toledo, M. Zoccali, and G. Pietrzynski. VISTA Variables in the Via Lactea (VVV): The public ESO near-IR variability survey of the Milky Way. , 15:433–443, July 2010. doi: 10.1016/j.newast.2009.12.002.

[49] P. Moskalik. Multi-mode oscillations in classical Cepheids and RR Lyrae-type stars. In J. A. Guzik, W. J. Chaplin, G. Handler, and A. Pigulski, editors, *Precision Asteroseismology*, volume 301 of *IAU Symposium*, pages 249–256, February 2014. doi: 10.1017/S1743921313014403.

[50] Navarrete, C., Catelan, M., Contreras Ramos, R., Alonso-García, J., Gran, F., Dékány, I., and Minniti, D. Near-ir period-luminosity relations for pulsating stars in centauri (ngc5139). *AA*, 604:A120, 2017. doi: 10.1051/0004-6361/201630102. URL https://doi.org/10.1051/0004-6361/201630102.

[51] Martin Paegert, Keivan G. Stassun, and Dan M. Burger. The eb factory project. i. a fast, neural-net-based, general purpose light curve classifier optimized for eclipsing binaries. *The Astronomical Journal*, 148(2):31, 2014. URL http://stacks.iop.org/1538-3881/148/i=2/a=31.

[52] W. Palma. *Long Memory Time Series: Theory and Methods*. Wiley Series in Probability and Statistics. John Wiley & Sons, Hoboken, New Jersey., 2007. ISBN 9780470131459. URL https://books.google.cl/books?id=HhGa8CcUsWIC.

[53] Wilfredo Palma and Mauricio Zevallos. Fitting non-gaussian persistent data. *Applied Stochastic Models in Business and Industry*, 27(1):23–36, 2011. ISSN 1526-4025. doi: 10.1002/asmb.847. URL http://dx.doi.org/10.1002/asmb.847.

[54] K. Pichara, P. Protopapas, D.-W. Kim, J.-B. Marquette, and P. Tisserand. An improved quasar detection method in eros-2 and macho lmc data sets. *Monthly Notices of the Royal Astronomical Society*, 427(2):1284–1297, 2012. doi: 10.1111/j.1365-2966.2012.22061.x. URL +http://dx.doi.org/10.1111/j.1365-2966.2012.22061.x.

[55] K. Rehfeld, N. Marwan, J. Heitzig, and J. Kurths. Comparison of correlation analysis techniques for irregularly sampled time series. *Nonlinear Processes in Geophysics*, 18(3):389–404, 2011. doi: 10.5194/npg-18-389-2011. URL http://www.nonlin-processes-geophys.net/18/389/2011/.

[56] J. W. Richards, D. L. Starr, N. R. Butler, J. S. Bloom, J. M. Brewer, A. Crellin-Quick, J. Higgins, R. Kennedy, and M. Rischard. On Machine-learned Classification of Variable Stars with Sparse and Noisy Time-series Data. , 733:10, May 2011. doi: 10.1088/0004-637X/733/1/10.

[57] J. W. Richards, D. L. Starr, A. A. Miller, J. S. Bloom, N. R. Butler, H. Brink, and A. Crellin-Quick. Construction of a Calibrated Probabilistic Classification Catalog: Application to 50k Variable Sources in the All-Sky Automated Survey. , 203:32, December 2012. doi: 10.1088/0067-0049/203/2/32.

[58] Nikolay N. Samus, Elena V. Kazarovets, and Olga V. Durlevich. Catalogs of variable stars, current and future. *Proceedings of the International Astronomical Union*, 5 (S264):496–498, 2009. doi: 10.1017/S174392130999319X.

[59] Iwao Sekita, Takio Kurita, and Nobuyuki Otsu. Complex autoregressive model and its properties. Electrotechnical Laboratory, 1991.

[60] Martin Sewell. Characterization of financial time series, 2011.

[61] I. Soszyński, A. Udalski, M. K. Szymański, P. Pietrukowicz, P. Mróz, J. Skowron, S. Kozłowski, R. Poleski, D. Skowron, G. Pietrzyński, L. Wyrzykowski, K. Ulaczyk, and M. Kubiak. Over 38000 RR Lyrae Stars in the OGLE Galactic Bulge Fields. , 64:177–196, September 2014.

[62] P. B. Stetson. On the Automatic Determination of Light-Curve Parameters for Cepheid Variables. , 108:851, October 1996. doi: 10.1086/133808.

[63] M. K. Szymański, A. Udalski, I. Soszyński, M. Kubiak, G. Pietrzyński, R. Poleski, Ł. Wyrzykowski, and K. Ulaczyk. The Optical Gravitational Lensing Experiment. OGLE-III Photometric Maps of the Galactic Bulge Fields. , 61:83–102, June 2011.

[64] R Core Team. R: A language and environment for statistical computing. *R Foundation for Statistical Computing*, 2015.

[65] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society (Series B)*, 58:267–288, 1996.

[66] Henghsiu Tsai. On continuous-time autoregressive fractionally integrated moving average processes. *Bernoulli*, 15(1):178–194, 02 2009. doi: 10.3150/08-BEJ143. URL https://doi.org/10.3150/08-BEJ143.

[67] A. Udalski, M. Szymanski, J. Kaluzny, M. Kubiak, and M. Mateo. The Optical Gravitational Lensing Experiment. , 42:253–284, 1992.

[68] A. Udalski, M. Kubiak, and M. Szymanski. Optical Gravitational Lensing Experiment. OGLE-2 – the Second Phase of the OGLE Project. , 47:319–344, July 1997.

[69] A. Udalski, M. K. Szymański, and G. Szymański. OGLE-IV: Fourth Phase of the Optical Gravitational Lensing Experiment. , 65:1–38, March 2015.

[70] Udalski, Andrzej. Ogle cepheids and rr lyrae stars in the milky way. *EPJ Web Conf.*, 152:01002, 2017. doi: 10.1051/epjconf/201715201002. URL https://doi.org/10.1051/epjconf/201715201002.

[71] Olga Y. Uritskaya and Vadim M. Uritsky. Predictability of price movements in deregulated electricity markets. *Energy Economics*, 49(C):72–81, 2015. doi: 10.1016/j.eneco.2015.02.0. URL https://ideas.repec.org/a/eee/eneeco/v49y2015icp72-81.html.

[72] Iut Tri Utami, Bagus Sartono, and Kusman Sadik. Comparison of single and ensemble classifiers of support vector machine and classification tree. *Journal of Mathematical Sciences and Applications*, 2(2):17–20, 2014. doi: 10.12691/jmsa-2-2-1. URL http://pubs.sciepub.com/jmsa/2/2/1.

[73] Zhu Wang. cts: An r package for continuous time autoregressive models via kalman filter. *Journal of Statistical Software, Articles*, 53(5):1–19, 2013. ISSN 1548-7660. doi: 10.18637/jss.v053.i05. URL https://www.jstatsoft.org/v053/i05.

[74] M. Zechmeister and M. Kürster. The generalised Lomb-Scargle periodogram. A new formalism for the floating-mean and Keplerian periodograms. , 496:577–584, March 2009. doi: 10.1051/0004-6361:200811296.

[75] Q. J. Zhang and K. C. Gupta. *Neural Networks for RF and Microwave Design (Book + Neuromodeler Disk)*. Artech House, Inc., Norwood, MA, USA, 1st edition, 2000. ISBN 1580531008.

[76] Ji Zhu, Hui Zou, Saharon Rosset, and Trevor Hastie. Multi-class adaboost. *Statistics and its Interface*, 2(3):349–360, 2009.

[77] Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2): 301–320, 2005. ISSN 1467-9868. doi: 10.1111/j.1467-9868.2005.00503.x. URL http://dx.doi.org/10.1111/j.1467-9868.2005.00503.x.

Appendix A

Connection Between CAR(1) and IAR process

The CAR(1) model is described by the following equation,

$$\epsilon(t) - \frac{\beta}{\alpha_0} = e^{-\alpha_0(t-s)}(\epsilon(s) - \frac{\beta}{\alpha_0}) + e^{-\alpha_0 t}(I(t) - I(s)) \tag{A.0.1}$$

The IAR model is described by the following equation,

$$\varepsilon_{t_1}, \quad y_{t_j} = \phi^{t_j - t_{j-1}}\, y_{t_{j-1}} + \sigma\, \sqrt{1 - \phi^{2(t_j - t_{j-1})}}\, \varepsilon_{t_j} \text{ for } j = 2, \ldots, n, \tag{A.0.2}$$

Now, setting $\beta = 0$ and $e^{-\alpha_0} = \phi$, the equation (A.0.1) becomes to,

$$\epsilon(t) = \phi^{t-s}\epsilon(s) + \phi^t(I(t) - I(s))$$

To prove the equivalence between the equation (A.0.1) and equation (A.0.2), we can prove that

$$\phi^t(I(t) - I(s)) = \sigma\, \sqrt{1 - \phi^{2(t-s)}}\, \varepsilon_{t_j}$$

Let, $Z(t) = \sigma\, \sqrt{1 - \phi^{2(t_j - t_{j-1})}}$ we know that $\mathbb{E}(Z(t)) = 0$ and $\mathbb{V}(Z(t)) = \sigma^2(1 - \phi^{2(t-s)})$.

Furthermore, if $Z(t) = \phi^t(I(t) - I(s))$, we know that $\mathbb{E}(Z(t)) = 0$, and

$$\begin{aligned}
\mathbb{V}(Z(t)) &= \mathbb{V}(\phi^t(I(t) - I(s))) \\
&= \mathbb{V}(e^{-\alpha_0 t}(I(t) - I(s))) \\
&= e^{-2\alpha_0 t}\mathbb{V}(I(t) - I(s)) \\
&= e^{-2\alpha_0 t}[\mathbb{V}(I(t)) + \mathbb{V}(I(s)) - 2Cov(I(t), I(s))
\end{aligned}$$

Let $t = s + h$, then

$$\begin{aligned}
\mathbb{V}(Z(t)) &= e^{-2\alpha_0(s+h)}[\mathbb{V}(I(s+h)) + \mathbb{V}(I(s)) - 2Cov(I(s+h), I(s))] \\
&= \sigma_0^2 e^{-2\alpha_0(s+h)}\left[\int_0^{s+h} e^{2\alpha_0 u}du + \int_0^{s} e^{2\alpha_0 u}du - 2\int_0^{s} e^{2\alpha_0 u}du\right] \\
&= \sigma_0^2 e^{-2\alpha_0(s+h)}\left[\int_0^{s+h} e^{2\alpha_0 u}du \int_0^{s} e^{2\alpha_0 u}du\right] \\
&= \sigma_0^2 e^{-2\alpha_0(s+h)}\left[\int_s^{s+h} e^{2\alpha_0 u}du\right] \\
&= \sigma_0^2 e^{-2\alpha_0(s+h)}\frac{1}{2\alpha_0}\left[e^{2\alpha_0(s+h)} - e^{2\alpha_0 s}\right] \\
&= \sigma_0^2 \frac{1}{2\alpha_0}\left[1 - e^{-2\alpha_0 h}\right] \\
&= \frac{\sigma_0^2}{2\alpha_0}\left[1 - \phi^h\right]
\end{aligned}$$

Let $\sigma^2 = \frac{\sigma_0^2}{2\alpha_0}$ then we proved that the two first moments of $e^{-\alpha_0 t}(I(t) - I(s))$ and $\sigma\sqrt{1 - \phi^{2(t_j - t_{j-1})}}$ are equivalent.