MACHINE LEARNING METHODS FOR DELAY ESTIMATION IN GRAVITATIONALLY LENSED SIGNALS

by

SULTANAH AL OTAIBI

A thesis submitted to
The University of Birmingham
for the degree of
DOCTOR OF PHILOSOPHY

School of Computer Science
College of Engineering and Physical Sciences
The University of Birmingham
July 2016

Abstract

Strongly lensed variable quasars can serve as precise cosmological probes, provided that time delays between the image fluxes can be accurately measured. A number of methods have been proposed to address this problem. This thesis, explores in detail a new approach based on kernel regression estimates, which is able to estimate a single time delay given several data sets for the same quasar. We develop realistic artificial data sets in order to carry out controlled experiments to test the performance of this new approach. We also test our method on real data from strongly lensed quasar Q0957+561 and compare our estimates against existing results. Furthermore, we attempt to resolve the problem for smaller delays in gravitationally lensed photon streams. We test whether a more principled treatment of delay estimation in lensed photon streams, compared with the standard kernel estimation method, can have benefits of more accurate (less biased) and/or more stable (less variance) estimation. To that end, we propose a delay estimation method in which a single latent non-homogeneous Poisson process underlying the lensed photon streams is imposed. The rate function model is formulated as a linear combination of nonlinear basis functions. Such a unifying rate function is then used in delay estimation based on the corresponding Innovation Process. This method is compared with a more straightforward and less principled baseline method based on kernel estimation of the rate function. Somewhat surprisingly, the overall emerging picture is that the theoretically more principled method does not bring much practical benefit in terms of the bias/variance of the delay estimation. This is in contrast to our previous findings on daily flux data.

ACKNOWLEDGEMENTS

First I would like to thank my supervisor Professor Peter Tiňo for introducing me to this challenging area of machine learning and astrophysics. My heartfelt appreciation and acknowledgement to him for his advice, time, and valuable constructive feedback.

My gratitude goes also to Dr Juan C. Cuevas-Tello for his valuable advice and support.

Special thanks must go to Dr Ilya Mandel and Dr Somak Raychaudhury for sharing their knowledge in astrophysics. It has been a real pleasure and a learning experience to work with them.

I would like to thank my Thesis Group Members, Professor Russell Beale and Dr Iain Styles, for their time, thoughts and interesting research ideas about this thesis development.

Lastly, my special thanks to my husband and my family for their unconditional support and encouragement.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ALGORITHMS

CHAPTER 1

INTRODUCTION

Time delays between images of strongly-lensed distant variable sources can serve as a valuable tool for cosmography, providing an alternative to other tools, such as cosmic microwave background measurements and distance measures based on standard candles [e.g., 43, 70, 108, 125, 129]. Actively studied strong quasars with time-delay measurements include RXJ1131-1231 [e.g., 125, 128] and B1608+656 [e.g., 33, 43, 126]; Q0957+561 [e.g., 32, 46, 92]; SDSS J1650+4251 and HE 0435-1223 [e.g., 19, 59, 132]; SDSS J1029+2623 [e.g., 35]; and SDSS J1001+5027 [e.g., 107]. These have been used to infer Hubble constant measurements with competitive accuracies.

However, time delays are difficult to measure because of the unknown intrinsic source variability, the limited observational cadence, and the measurement noise. A number of methods have been developed to accurately estimate time delays. These include the dispersion spectra (DS) method [19, 100, 132]; the polynomial and curve-fitting methods [30, 131]; the free-knot spline, variability of regression differences (based on Gaussian process regression), and dispersion minimization [127]; Gaussian process (GP) modeling [e.g., 51] and the combined method based on the PRH approach [50]. However, this remains an active area of research, especially in view of the upcoming surveys such as Large Synoptic Survey telescope (LSST), which will provide unprecedented data sets with strongly lensed distant quasars [e.g., 129] [and the recent mock data challenge 27, 69].

A kernel-based method with variable width (K-V) for time delay estimation was pro-

posed by [24]. This was combined with an evolutionary algorithm (EA) for parameter optimization [25]. However, the computational time complexity of EA method is $O(n^6)$ [21]. This restriction makes it inadequate for handling long time series, e.g.(Schild & Thomson) data [116][1]. This complexity is due to matrix inversion in kernel-based methods for weights estimation. Automatic methods for time delay estimation have been proposed to speed up algorithms in order to deal with long time series, based on Artificial Neural Networks [41]; these can be parallelized [22]. Alternatively, a simple hill-climbing optimization has been proposed [23].

1.1 Motivation

Although a great deal of effort has been devoted to estimate the time delay between the two images of Q0957+561, the problem is still open and attracts the interest of researchers. The ongoing debate on the true value of the delay between image A and B of Q0957+56, has been one of the main motivations of this research. We attempted to apply probabilistic models for time delay estimation in the context of kernel methods and machine learning. Our aim was to estimate a single time delay given several data sets for the same quasar.

Daily measurements can be used to predict longer (days and months) delays. However, when countering the problem of shorter (hours or even minutes) delays these measurements are insufficient and one needs to investigate the individual arrival times of photons. We were motivated by our findings on daily flux data to apply more principled methods for delay estimation in lensed photon streams. We studied whether, compared with the standard kernel based baseline, such principled approaches can bring benefits in terms of more stable (less variance) estimation.

[1] http://cfa-www.harvard.edu/∼rschild/fulldata2.txt

1.2 Research questions

In this work a number of important research questions are to be addressed:

- What is the effect of noise and gaps on the performance of any time delay estimators?
- Since the exact time delay of Q0957+561 is unknown, the question to be asked is how the performance of time delay estimation methods can be tested?
- How to design and generate 'realistic' synthetic data sets?
- How to resolve the problem of shorter time delays?
- Whether or not the photon streams can provide sufficient data to estimate shorter delays?
- How beneficial is a more principled treatment of delay estimation in lensed photon streams compared with standard kernel estimation?

1.3 Contribution

1.3.1 Delay estimation for gravitationally lensed fluxes (daily measurements)

The main contribution of this thesis is a new probabilistic method that is efficient, robust to observational gaps, capable of directly incorporating measured noise levels reported for individual flux measurements, and able to estimate a single time delay given several data sets for the same quasar. We also carefully construct synthetic data sets within the framework of multiobjective optimization to reproduce realistic flux variability, observational gaps, and noise levels. This allows us to test our proposed kernel regression estimate method on synthetic as well as real data, in order to measure the bias and variance of the method.

1.3.2 Delay estimation for gravitationally lensed fluxes (shorter delays)

We propose a delay estimation method in which a single latent non-homogeneous Poisson process underlying the lensed photon streams is imposed. The rate function model is formulated as a linear combination of nonlinear basis functions. Such a unifying rate function is then used in delay estimation based on the corresponding Poisson and Innovation Processes. These methods are then compared with a more straightforward and less principled baseline method based on kernel estimation of the rate function. We present a useful study for future developments of alternative methods for the delay estimation in lensed photon streams.

1.4 Publication

- Al Otaibi, S., Tiňo, P., Cuevas-Tello, J.C., Mandel, I. and Raychaudhury, S. Kernel regression estimates of time delays between gravitationally lensed fluxes. *Monthly Notices of the Royal Astronomical Society*, 459(1):573-584, 2016.

- Al Otaibi, S., Tiňo, P. and Raychaudhury, S. Probabilistic Modelling for Delay Estimation in Gravitationally Lensed Photon Streams. *17th International Conference on Intelligent Data Engineering and Automated Learning (IDEAL 2016)*, pp. 552-559, Lecture Notes in Computer Science, Springer-Verlag, LNCS 9937, 2016.

1.5 Thesis Organization

The remainder of this thesis is organized as follows:

Chapter 2 presents the astronomical background of the gravitational lensing phenomenon and the importance of time delay estimation. We discuss some of previously

proposed delay methods, namely, cross correlation and dispersion spectra, to compare with the new approach.

Chapter 3 gives a brief overview of the application of machine learning algorithms in astronomy. We also introduce the kernel-based approach for time delay estimation in this chapter.

Chapter 4 presents the Nadaraya-Watson estimator with known noise levels (henceforth *NWE*). We also extend it to a linear noise model with unknown noise (henceforth *NWE++*). This chapter outlines the experimental results of synthetic and real data.

In chapter 5 we address smaller time delays in gravitationally lensed photon streams and we propose two models: Poisson Process Based Estimation *(PPE)* and Innovation Process Based Estimation *(IPE)* to estimate the time delay in streams of photons. This chapter presents the experimental results for synthetic data.

In chapter 6 we summarize the main contributions of the thesis and give conclusions of the proposed work.

CHAPTER 2

ASTRONOMICAL BACKGROUND AND REVIEW OF RELATED WORK

In this chapter we present the basic concepts and background knowledge necessary to understand the problem of time delay in gravitationally lensed fluxes. We introduce gravitational lensing phenomena and the first discovered gravitational Lens: Q0957+561. We also describe the real data optical and radio. In the second part of this chapter, we present a survey based on Q0957+561 for some of its time delay estimates. Further, a review of the most popular methods in astronomy is introduced at the end of this chapter.

2.1 Gravitational lensing

Einstein's General Theory of Relativity is one of the greatest intellectual achievements of the 20th century. It has explained a number of interesting phenomena such as the expanding universe, black holes and gravitational lenses. Einstein believed that light, which was considered to be massless, is affected by gravity, which results from the distortion of the four-dimensional space-time curvature due to the presence of masses (see Figure 2.1). Light rays move along geodesic paths, i.e. the shortest path between two points; when the space-time is curved as a consequence of the presence of massive objects, these geodesic paths are also curved. This phenomenon is called gravitational lensing [21, 89, 114].

The gravitational lensing system requires a distant source such as quasar and a massive

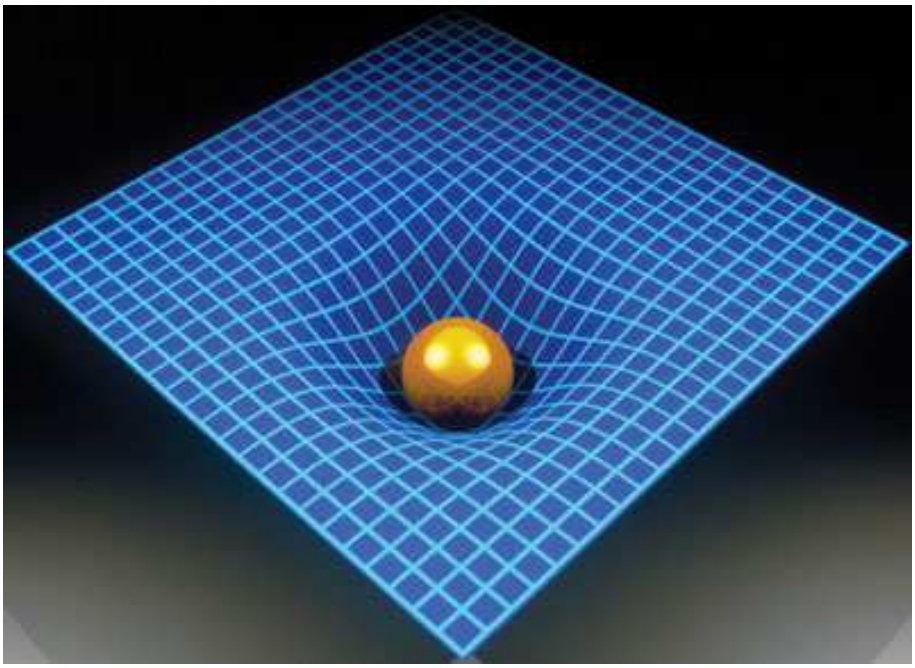


Figure 2.1: Space-time distortion.
Source: Figure obtained from http://www.genetology.net/.

object, which acts as a lens that could be a galaxy or a cluster of galaxies between the source and the observer. Figure 2.2 illustrates the gravitational lensing process in detail. The bright source is located on the left (shaded circle); the the gravitational lens is in the

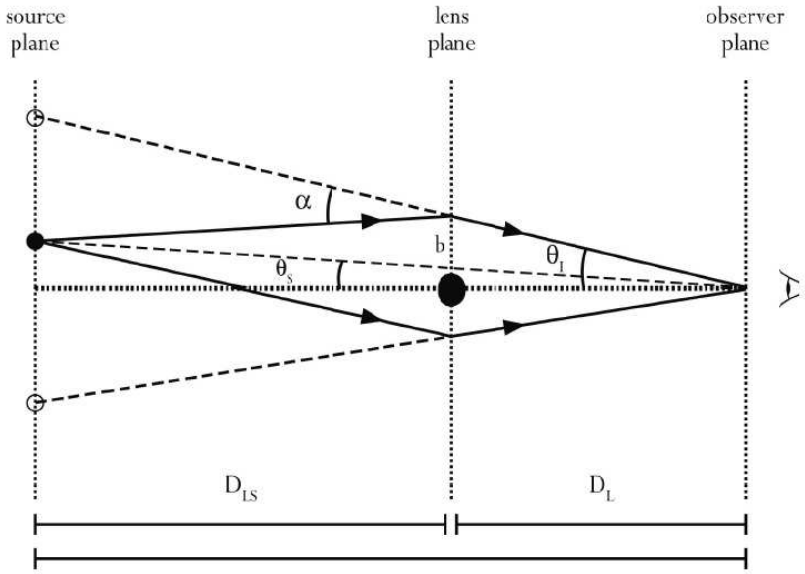


Figure 2.2: Gravitational lensing.
Source: https://en.wikipedia.org/.

middle, (large shaded circle) and the observer is on the right. As indicated in Figure 2.2,

light deflection occurs in the lens plane resulting in the observation of two images of the source [21]. Due to the gravitational lensing effect, photons taking different paths arrive at the observer at different times. One can calculate the angular amount of deflection α according to Einstein's general theory of relativity

$$\alpha = \frac{4GM}{c^2 b} \tag{2.1}$$

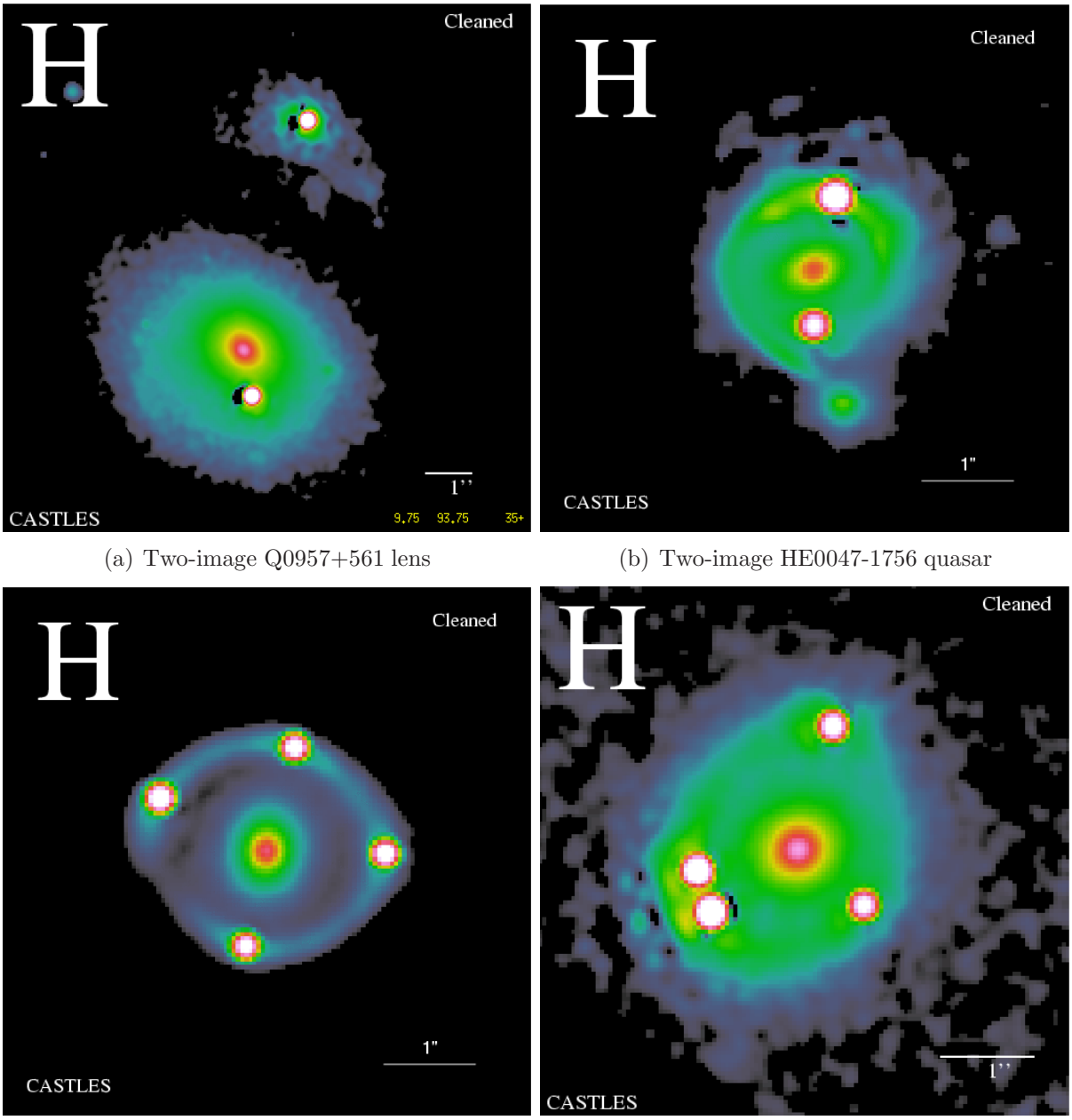
where G is the gravitational constant, M is the lens mass, c is the speed of light and b is the closest distance from the source to the lens [21, 89, 114]. A traveling photon of light from the source to the observer and passing the mass M from a direction θ_I will delayed by

$$-\frac{4GM}{c^3} \ln \theta_I \tag{2.2}$$

This delay is another result of the gravitational lensing phenomena. The arrival times of the photons streams of the two images of the source differ by Δ[21, 114]. This delay between the arrival times of the photons is what this thesis is concerned with.

The observed effect of the gravitational lensing process varies from changing the shape of the image by weak lensing effect to produce multiple images of the source by strong lensing, depending on the mass of the lens and its relative position [114]. Figure 2.3, shows examples of quasars with two and four images taken by the Hubble Space Telescope.

Estimating the time delay between two gravitationally lensed images for the same source is of great importance for astronomical application. The time delay between two light curves depends on the mass of the gravitational lens. It is, therefore, the most direct method for estimating the masses of gravitational lenses (galaxies and clusters of galaxies) and measuring the distribution of matter in the universe. Time delays can be also used to measure universe parameters such as its expansion rate, mass density and the Hubble constant. Such parameters can be used for predicting the age and future of the universe [21, 42, 109, 114, 115].



(a) Two-image Q0957+561 lens

(b) Two-image HE0047-1756 quasar

(c) Four-image PG1115+080 lens

(d) Four-image HE0435-1223 quasar

Figure 2.3: Examples of Gravitational lenses.
Source: H-band cleaned images observed by the Hubble Space telescope.

2.2 Gravitational Lens: Q0957+561

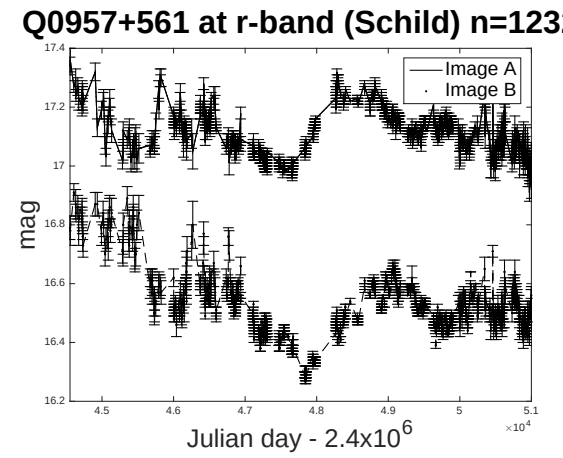
In the early 1960's, quasars were discovered to be a strong source of radio waves. Quasars are extremely bright and distant objects in our universe. Quasar are believed to be produced by super massive black holes surrounded by an accretion disks. They are highly energetic objects that emit huge amounts of electromagnetic energy (radio waves and

visible light) due to the presence of the super massive black holes in the centers of the galaxies in which the quasars are located [118].

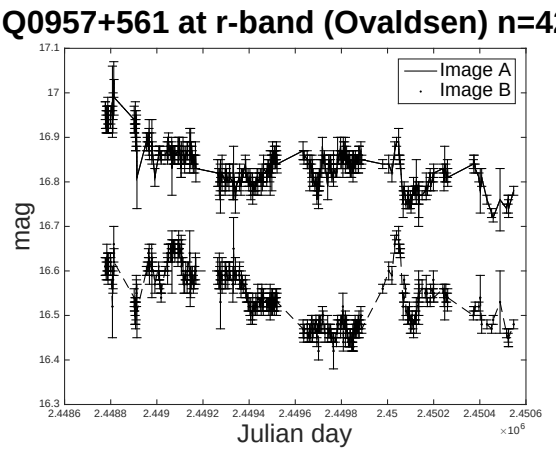
The first discovered gravitationally lensed quasar, Q0957+561, is an extremely bright galaxy with a super massive central black hole. It is, also known as the twin quasar, a double image quasar, that has two images: A and B due to the gravitational lensing effect, that is a lensing galaxy (an intervening mass between the quasar and the observer) distorts the light that traveling from the quasar resulting in, two or multiple images of the same quasar appear in the sky. The fluctuation in the brightness of Q0957+561 can be observed and sampled on a time scale of days [9]. For this specific quasar Q0957+561, the time delay Δ is around 400 days (see Section 2.3 for more details on the long controversy over the value of the delay).

Monitoring campaigns provide us with daily observed data in the form of light fluxes, with each flux capturing the fluctuations in the brightness of an individual image of lensed quasar during a period of time. In other words, the brightness of the images is measured as a function of time. It can be observed at different wavelengths, e.g. radio or optical, and at different observational times. These observations are usually noisy, with different levels of errors, and irregularly sampled, they contain gaps. For our purposes, the real data are available as two irregularly sampled time series of fluxes of the two images A and B. We used six different data sets from Q0957+56. The details on data sets are presented in Table 2.1 and the plots are shown in Figure 2.4. We work only on the final light curves that are reported in [44, 63, 95]. The largest optical data set was provided by Schild, private communication [116]. The whole process of preparing and treating the measured data is beyond the scope of our research. A full description of the data set with explanation of reduction, correction and compilation procedures can be found in [67] for some of the data sets.

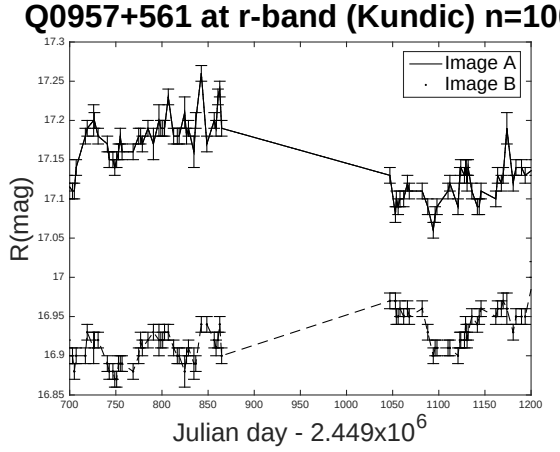
In the following sections, we describe specific aspects of observational radio and optical data.



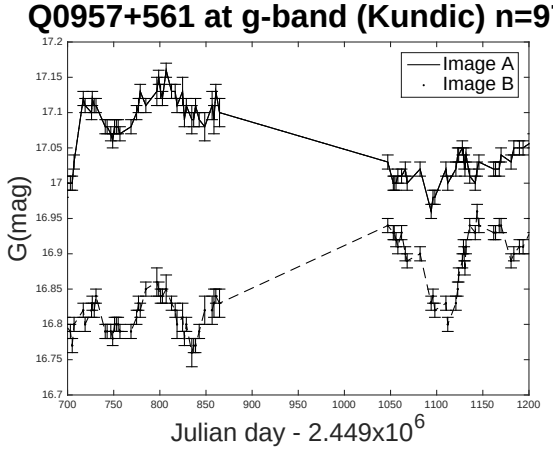
(a) D^1



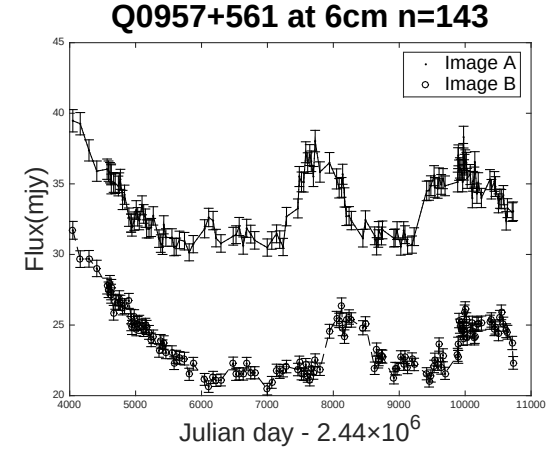
(b) D^2



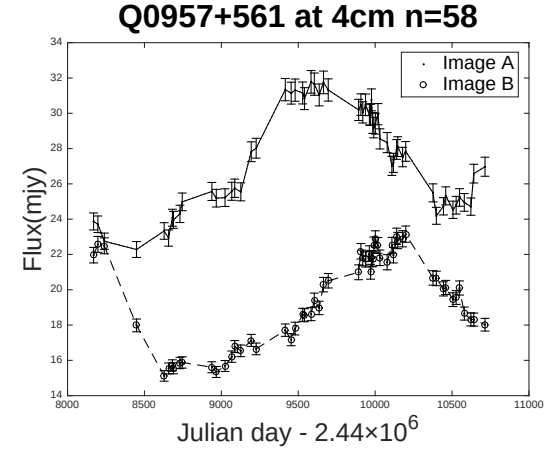
(c) D^3



(d) D^4



(e) D^5



(f) D^6

Figure 2.4: Data set Q0957+561. Image A from D^1 is shifted up by 0.6 magnitudes for clarity; image A from D^2 is shifted up by 0.25 magnitudes; image A from D^4 is shifted up by 0.05 magnitudes. For more details on these data sets see Table 2.1.

Table 2.1: Datasets: Q0957+561

Id	N^{ℓ}	Data	Type	Ratio/Offset	Monitoring range	Ref
D^1	1232	optical	r-band	0.05	16/11/1979 – 4/7/1998	[116]
D^2	422	optical	r-band	0.076	2/6/1992 – 8/4/1997	[95]
D^3	100	optical	r-band	0.21	3/12/1994 – 6/7/1996	[63]
D^4	97	optical	g-band	0.117	3/12/1994 – 6/7/1996	[63]
D^5	143	radio	6cm	1/1.43	23/6/1979 – 6-Oct-1997	[44]
D^6	58	radio	4cm	1/1.44	4/10/1990 – 22/9/1997	[44]

Table 2.2: Radio Data Q0957+561 at 6 cm: The final light curves

Observation	Calendar Date	Julian Day	Image A	Image B
1	23 Jun 1979	4,047.50	39.26	31.71
2	13 Oct 1979	4,160.16	39.26	29.67
3	23 Feb 1980	4,292.79	37.37	29.69
...	...	...	...	...
143	6 Oct 1997	10,728.18	33.06	22.32

2.2.1 Radio data

The gravitational lens Q0957+561 was monitored from 1979 to 1997. Radio observations were collected from the National Radio Astronomy Observatory (NRAO) Very Large Array radio telescope (VLA) at two different wavelengths: 4 cm and 6 cm . The 6 cm data set (hereafter D^5) has143 observations from 23 June 1979 to 6 October 1997. The 4 cm data set (hereafter D^6) has 58 observations from 4 October 1990 to 22 September 1997. These data sets are reported in [44]. The Radio data sets are depicted on the bottom row of Figure 2.4. D^5 is shown in Table 2.2, where the first column shows the observation numbers. The observational times are represented in the calendar date in the second column and in Julian days [1] in the third column. The last two columns have the flux densities of images A and B. The flux densities are reported in millijanskys (mJy) and as in [44], the error involved are assumed to be 2% of the flux densities. In practice, we only need and use the last three columns: the observational times in Julian days and their corresponding fluxes densities for image A and B.

[1] Julian day (JD) is the continuous count of days that have elapsed since the beginning of the Julian Period, which is a chronological interval of 7980 years beginning 4713 BC. This is used primarily by astronomers as a way of representing the date as a continuous real variable.

2.2.2 Optical data

Optical data are also available as time series where the fluxes are measured by imaging devices such as Charge-Coupled Device (CCD) with filters to restrict the range of wavelength/frequency of light observed. The green (g-) and red (r-) bands represent measurements obtained with filters in the wavelength range 400-550 nm and 550-700 nm, respectively. The measurement unit of the flux is known as magnitudes (mag), which is a logarithmic units defined as mag = 2.5 log10 f + constant, where f can be represented in mJy (as radio flux units above) [21].

Table 2.3, shows an example of optical data set. The first column shows the observation numbers. The observational time are represented in the calendar date in the second column and in Julian days in the third column. The fourth and the fifth columns have the flux densities of images A and B reported in (mag). Finally, the last two columns represent the standard deviations of measurement errors at each observation for each flux density (A and B). These errors are assumed to be zero mean Gaussian. Optical data are more accurate than radio data since the errors represent about 0.006% - 0.474% of the flux, i.e 0.001-0.08 mag compared with 2% of the flux in the radio data [21]. Four optical data sets are depicted in he top and middle rows of Figure 2.4 (hereafter referred to as D^1, D^2, D^3 and D^4) and also detailed in Table 2.1:

- D^1 is the largest optical data set of Q0957+561 taken at r-band with 1232 observations from 16 November 1979 to 4 July 1998 [116].
- D^2 refers to optical data at r-band with 422 observations from 2 June 1992 to 8 April 1997 [95].
- D^3 and D^4 are optical data sets at r-band with 100 observations and g-band with 97 observations respectively covering the same period of time from 3 December 1994 to 6 July 1996 [63].

As noticeable from Figure 2.4, specifically plots (a,e and f), there exists a time delay between the fluxes of image A and B which is the quantity to be determined. One simple

Table 2.3: Optical Data Q0957+561 at g-band: The final light curves

Observation	Julian Day	Image A	Image B	Error A	Error B
1	9,689.009	16.9505	16.8010	0.0152	0.0152
2	9,691.007	16.9439	16.7957	0.0111	0.0111
3	9,695.001	16.9356	16.7949	0.0090	0.0090
...	...	...	...	...	
97	10,270.652	17.0928	16.9597	0.0145	0.0119

way of roughly estimating the delay is to fix one of the time series and shift the other (in time and flux) and then evaluate the goodness of the fitting.

2.3 Previous Work

Recent publications on time delays focus on the quasars RXJ1131-1231 and B1608+656 because their photometry allows precise time delays [43, 125]. However, the most studied quasar is the Q0957+561, and it has been adopted the time delay 417.0±1.5 days [46, 92], which was reported by [63]. A number of varied estimates have been proposed for the time delay between the two images of Quasar Q0957+561. A review of the controversy about the time delay estimates has been presented in [45]. Table 2.4 summarizes the estimates for Q0957+561 from 1997 to 2005 [21]. One can easily conclude that it is difficult to accurately estimate the time delay due to the irregular sampling of the noisy data. The uncertainty of estimation increases in proportion to the noise levels and the gap sizes in the data.

The most recent time delay methods include: free-knot spline, variability of regression differences (based on Gaussian process regression), and dispersion minimization [127]. The latter is based on dispersion spectra [100]. They have been tested on synthetic data, based on light curves from quasar HE 0435-1223. They also have been employed to estimate the time delay on RX J1131-1231 [128]. The regression difference method is reported as the most accurate technique [127, 128].

Another method based on Gaussian processes, in particular the PRH method, is the combined method [50]. This method is tested on several real data sets including the

Q0957+561. This method shows different time delays for different data sets on the same quasar. It proposes an automatic way to estimate time delays. They claim that there is a single free parameter (the number of observation pairs for the structure function), but they use smooth polynomial, another parameter to set (polynomial trend). The time delay estimates are sensitive to these parameters. Nevertheless, methods based on a structure function have been found to be accurate [24].

Table 2.4: Review of Time Delay Estimates of Q0957+561;(obtained from [21]).

Data	Year	Method	Delay estimate
1997	Optical(g,r)	Linear, Cross correlation, Dispersion spectra and PHR method	417 ± 3 [64]
1997	Optical(g)	Cross correlation and Dispersion spectra	427 ± 3 [94]
1997	Optical(r)	SOLA	425 ± 17 [103]
1998	Optical(g,r)	Dispersion spectra	416.3 ± 1.7 [101]
1999	Radio(4,6)	PRH method and Dispersion spectra	4.9 ± 30 [45]
1997	Optical(g,r)	Linear, Cross correlation and Dispersion spectra	422.6 ± 0.6 [93]
2001	Optical(r)	$\mathcal{X}^2$ algorithm	423 ± 9 [12]
2003	Optical(r)	PRH method	417.09 ± 0.07 [15]
2003	Optical(r)	Dispersion spectra and $\mathcal{X}^2$	424.9 ± 1.2 [96]
2005	Radio(4,6)	Bayesian method	394.8 ± 0.5 [47]
2005	Optical(r)	Bayesian method	423.5 ± 0.5 [47]

In the following subsections, we review the most popular methods that we used in our experiments for this research.

2.3.1 Cross Correlation

There are two versions of the methods based on cross-correlation: the Discrete Correlation Function (DCF; [28]) and its variant, the Locally Normalized Discrete Correlation Function (LNDCF; [67]). Both calculate correlations directly on discrete pairs of light curves. These methods avoid interpolation in the observational gaps. They are also the simplest and quickest time delay estimation methods.

First, time differences (lags), $\Delta t_{ij} = t_j - t_i$, between all pairs of observations are binned into discrete bins. Given a bin size $\Delta\tau$, the bin centered at lag τ is the time interval $I_\tau = [\tau - \Delta\tau/2, \tau + \Delta\tau/2]$. The DCF at lag τ is given by

$$\mathrm{DCF}(\tau) = \frac{1}{P(\tau)} \sum_{i,j}^{t_i,t_j \in I_\tau} \frac{(y_A(t_i) - \bar{a})(y_B(t_j) - \bar{b})}{\sqrt{(\sigma_a^2 - \sigma_A^2(t_i))(\sigma_b^2 - \sigma_B^2(t_j))}}, \tag{2.3}$$

where $P(\tau)$ is the number of observational pairs in the bin centered at τ, $\bar{a}$ and $\bar{b}$ are the means of the observed data, $y_A(t_i)$ and $y_B(t_j)$, and their variances are σ_a^2 and σ_b^2, respectively.

Likewise,

$$\mathrm{LNDCF}(\tau) = \frac{1}{P(\tau)} \sum_{i,j}^{t_i,t_j \in I_\tau} \frac{(y_A(t_i) - \bar{a}(\tau))(y_B(t_j) - \bar{b}(\tau))}{\sqrt{(\sigma_a^2(\tau) - \sigma_A^2(t_i))(\sigma_b^2(\tau) - \sigma_B^2(t_j))}}, \tag{2.4}$$

where $\bar{a}(\tau)$, $\bar{b}(\tau)$, $\sigma_a^2(\tau)$ and $\sigma_b^2(\tau)$ are the lag means and variances in the bin centered at τ.

The time delay Δ is found when $\mathrm{DCF}(\tau)$ and $\mathrm{LNDCF}(\tau)$, given by equations (2.3) and (2.4), are greatest; i.e., at the best correlation [28, 67].

2.3.2 Dispersion Spectra

The Dispersion Spectra method [99, 100] measures the dispersion of time series of two light curves $y_A(t_i)$ and $y_B(t_j)$ by combining them (given a trial time delay Δ and ratio M) into a single signal, $y(t_k)$, $k = 1, 2, \ldots 2N$. In other words, given the delay Δ, the observed values of signal A, $\{y_A(t_i)\}_{i=1}^N$, and (delayed and rescaled) signal B, $\{\tilde{y}_B(t_i)\}_{i=1}^N$, where $\tilde{y}_B(t) = M y_B(t - \Delta)$, are joined together and re-ordered in time, forming a joint signal $\{y(t_k)\}_{k=1}^{2N}$ of length $2N$. We employ two versions of this method [100]:

$$\mathrm{DS}_1^2(\Delta) = \min_M \frac{\sum_{a=1}^{2N-1} w_a \left(y(t_{a+1}) - y(t_a)\right)^2}{2 \sum_{a=1}^{2n-1} w_a} \tag{2.5}$$

and

$$\mathrm{DS}_{2,4}^2(\Delta) = \min_M \frac{\sum_{a=1}^{2N-1} \sum_{c=a+1}^{2N} H_{a,c} W_{a,c} G_{a,c} \left(y(t_a) - y(t_c)\right)^2}{2 \sum_{a=1}^{2n-1} \sum_{c=a+1}^{2n} H_{a,c} W_{a,c} G_{a,c}}, \tag{2.6}$$

where

$$w_a = \frac{1}{\sigma^2(t_{a+1}) + \sigma^2(t_a)}, W_{a,c} = \frac{1}{\sigma^2(t_a) + \sigma^2(t_c)} \tag{2.7}$$

are the statistical weights taking into account the measurement errors, where $G_{a,c} = 1$ only when $y(t_a)$ and $y(t_c)$ are from different images, and $G_{a,c} = 0$ otherwise, and

$$H_{a,c} = \begin{cases} 1 - \frac{|t_a - t_c|}{\delta}, & \text{if } |t_a - t_c| \leq \delta \\ 0, & \text{otherwise.} \end{cases} \tag{2.8}$$

Compared with DS_1^2, the $\mathrm{DS}_{2,4}^2$ method has an additional parameter, the *decorrelation length* δ, which signifies the maximum distance between observations that we are willing to consider when calculating the correlations [99].

The estimated time delay Δ is found by minimizing DS^2 over a range of time delay trials Δ, as above.

2.4 Summary

We started this chapter by giving a brief description of gravitational lensing phenomena and the first discovered gravitationally lensed quasar Q0957+561. We explained the time delay problem, i.e., the time delay Δ between pairs of images A and B of Q0957+561, where image B is delayed with respect to the image A in time by Δ. The time delay can be directly estimated from the optical or radio observations of the quasar Q0957+561. The significance of accurate prediction of the time delay in cosmology applications has been investigated in Section (2.1).

The real data sets have been described in Section (2.2) and time delay estimates of Q0957+561 from astronomy literature is summarized in Section (2.3). Finally, we reviewed the most popular methods used for time delay estimation and presented some of them in detail, namely, cross correlation and dispersion spectra methods in Section (2.3). These methods have free parameters that are difficult to set objectively based on the given data only. In other words, the values of such parameters cannot be resolved in

a principled manner based on the data. In Chapter (4), we will present the results of the above methods on synthetic data and real gravitationally lensed fluxes in the radio and optical ranges. we will discuss the results, advantages and disadvantages of each method.

CHAPTER 3

MACHINE LEARNING IN ASTRONOMY

In this chapter we present the basic concepts of machine learning. We also provide a review of kernel based methods for time delay estimation in gravitationally lensed fluxes.

3.1 Machine Learning

Machine learning is one of the most challenging research fields. The ongoing debates about whether it is a branch of the Artificial Intelligence (AI) field or whether it is derived from statistical learning theory is still active. Moreover, there is no universal agreement regarding the definition of what machine learning is even among the practitioners of its techniques [48, 79]. Here we review some of the machine learning definitions :

Arthur Samuel defined machine learning as a "Field of study that gives computers the ability to learn without being explicitly programmed".

A recent definition by Tom Mitchell is a " Well-posed Learning Problem: A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E" [79].

It simply refers to the process of making predictions from data by using an automated (algorithms). The aim is to extract 'knowledge' from data, hypothesizing that the questions about the underlying process might be answerable by the data.

Machine learning is now applied to solve large-scale, complex problems. Applications for machine learning include: medical diagnosis, finance, web search, computational biology, and speech recognition. The most popular approaches of machine learning to address real world problems are: neural networks, evolutionary computation, reinforcement learning, Bayesian networks, support vector machines and kernel methods. These approaches are classified into three types of learning: supervised learning, unsupervised learning and reinforcement learning. The choice of learning type is dependent on the problem to be solved. The primary goal of supervised learning is to build models that generalize "accurately predict" the future outcomes rather than predicting the existing one. The models can learn from past examples made of inputs and outputs, then apply what they have learned to unseen inputs in order to predict future outputs. How supervised learning works can be summarized in two simple steps:

- Model training: where the model is learning the relationship between attributes of training data and the outcomes.
- Model testing: by making predictions on new data when the true outcome is unknown.

Usually, data is divided into a training set and testing set. The training set is used to model the system, while the testing set is used to validate that model [48, 79].

Supervised learning tasks can be divided depending on the types of outputs in two categories: classification and regression. The predicted outcomes are discrete valued (finite number of labels) in the former and continuous (real numbers) the in latter [48].

3.2 Machine Learning in Astronomy

Machine learning algorithms can be greatly beneficial in the field of astronomy. Such powerful tools are able to utilize and analyze the increasing amount of collected astronomical data in order to provide significant scientific results. In this chapter, we present a review

of the use of machine learning techniques in the field of astronomy. We provide examples of the application of common machine learning algorithms in the field of astronomy.

3.2.1 Advantages of using machine learning algorithm in astronomy

Here we summarize some of the advantages of applying machine learning techniques in the field of astronomy [4].

- The amount of existing and upcoming data sets in the field is overwhelmingly large. It becomes a necessity to apply automated and intelligent methods in order to extract applicable scientific information from these data.
- The dimensionality of astronomical data is usually high and as widely known, it is difficult, if not impossible, to detect patterns in high dimensional data sets. Machine learning algorithms have proven to be useful in pattern recognition for such data.
- Machine learning algorithms can be used at more than one stage in the whole process. It can be utilized in knowledge discovery in databases as well as the classification/regression tasks.
- Machine learning algorithms can provide prior information about data that can be fully incorporated in the data analysis process. Even though the improvement in terms of final scientific results is not guaranteed when using these methods, they still act as important complementary approaches of data analysis.

3.2.2 Knowledge discovery in databases

Knowledge Discovery in Databases (KDD) is a widely used data mining technique. It can be defined as the process of extracting useful information from a collection of data. As discussed above in machine learning algorithms, KDD can also be applied on astronomical data [4, 55]. In this section, we review some possible applications of well known

machine learning and data mining algorithms on databases (i.e., any machine-readable astronomical data). The KDD process is commonly defined with the following stages:

Data collection

Data collection process includes transforming of all the existing data into a digital format (so that it can be machine-readable), acquiring and archiving new data, and performing any necessary cross-matching between data sets [4, 55].

Data preprocessing

This process is usually dependent on the problem under study and should be approached with caution when used since the final results of many machine learning algorithms can be significantly affected by the input data. The aim of preprocessing is to make data readable, meaningful and prepared for the application of any given algorithm [4, 105].

One of the important steps in the preprocessing stage is the transformation of data. Given an astronomical object, its attributes may need to be transformed. This transformation is usually done in the preprocessing stage using some of the transformation approaches. One common example of these approaches is scalarization, that is transforming categorical data to numerical data by giving each categorical attributes a unique numerical label. Another example of attributes transformations is binning, in which numerical data can be made categorical [4].

In general, this is the stage when bad (incorrect or unreasonable) and missing values should be dealt with. Bad values can be removed, ignored or replace using interpolation for example. Some algorithms cannot deal with missing values in data sets. This can be solved by either the removal of the object with the missing value or interpolation (depending on the problem) of the value from the existing data. Outliers in the data set may be removed depending on their extremity [4].

Another important process in the preprocessing stage is the normalization of the data. Normalization is especially needed in data sets where the attributes are greatly varied in

their ranges. In such cases, normalization can improve the accuracy of any used algorithm. This can be done in many ways e.g. scaling by a given amount, scaling using the minimum and maximum values, or scaling each attribute to have a mean of 0 and a standard deviation of 1 [4].

Attribute selection

Also known as dimensionality reduction, it is the process of automatically selecting a subset of relevant features. The goal is to select as few of attributes as possible in order to retrieve the maximum amount of information. Usually date objects come with a large number of attributes that might not be needed for the problem to be solved. In many cases, using all of object's attributes may lead to poor performance of the algorithm. A carefully applied attribute selection method can enhance the generalization of the model by reducing overfitting along with other advantages such as simplification of the model and shortening the training time. Dimensionality reduction can be very useful for algorithms that are unable to deal with noisy, irrelevant, or redundant attributes [4]. Common examples of sophisticated approaches used in this stage include principal component analysis (PCA) [57], forward selection and backward elimination [4].

3.2.3 Selection and use of machine learning algorithms

Examples of well known machine learning algorithms that gained popularity in astronomical data mining include supervised methods such as: Artificial Neural Network (ANN) [8]; Decision Tree (DT) [110] ; Support Vector Machine [122]; and k Nearest Neighbor (kNN) [117] and unsupervised methods such as: Kernel density estimation (KDE) [117]; K-means clustering [72]; Mixture models [77]; and the Kohonen self-organizing map (SOM) [60].

Selection of the 'optimal' algorithm to use depends on the data set and the actual application of the algorithm. In many cases, one might need to use more than one algorithm in order to reach the desired scientific results particularly for large data sets

[4].

Application of algorithms and some limitations

Some problems are raised by the application of machine learning and data mining algorithms in astronomical data. We are therefore summarizing some of these in this section.

The size of astronomical data sets is often very large and to be able to exploit these data one needs an advanced database technology that can deal with large scale data. Moreover, most astronomical data measurements have an associated error which results in noisy data sets that require a special treatment when using machine learning and data mining algorithms since these errors may affect the performance of such algorithms. Another important issue related to data is that the accuracy of the results from any given algorithm is highly dependent on the quality of the input data. Therefore, the algorithm may suffer in terms of performance when using insufficient, poorly collected or preprocessed input data [4].

Another limitation related to machine learning algorithms is that many of them have a significant number of parameters that need to be optimized. The optimal configuration of these parameters is often not obvious and usually results in further increases in the computational requirements [4].

Although machine learning can be very helpful in data analysis and pattern recognition tasks, it is the scientists role to successfully interpret the results and provide the final conclusions. In addition to that, there is no guarantee that using these algorithms will always produce accurate results. In some cases the results are either statistically invalid or completely wrong despite the fact that they appear reasonable [4].

3.2.4 Uses in astronomy

The field of astronomy produces a huge amount of data that are amenable to the machine learning approach. Examples of projects where astronomical data are used in machine

learning and data mining studies include: Sky Image Cataloging and Analysis System (SKICAT) [4, 135]; the Jet Propulsion Laboratory Adaptive Recognition Tool (JARTool) [4, 11] ; and the Lawrence Livermore National Laboratory Sapphire project [4, 58].

The collaboration between astronomy and machine learning can bring many benefits to both fields. Machine learning experts can employ more advanced and sophisticated algorithms to address astronomical problems with the domain scientists help and guidance regarding the problem details [4].

Here we briefly mention some examples of successful use of machine learning algorithms in astronomical problems. However, a full description of these problems is beyond the scope of this thesis.

Star-Galaxy separation

The number of astrophysical objects in typical surveys is huge (of order 108 or above) [4]. The automated separation of these astrophysical objects into stars, galaxies, and other objects is a classical classification task. Stars are small in size and distant from earth so they appear as point sources while galaxies, which are further away but with a a larger angle, appear as extended sources. Other objects (quasars and supernovae) also appear as point sources [4]. Examples of machine learning algorithms that have been used to successfully perform this separation task include: ANN [4, 90, 91]; DT [4, 5, 134]; and SOM [4, 78].

Morphological classification

This is another example of classification tasks that are needed in the field of astronomy. Galaxy morphology simply means the study of the appearance (shape; size; and structure) of galaxies. There are several systems for the morphological classification of galaxies, the most famous being the Hubble sequence. Hubble's system broadly divides galaxies based on their visual appearance into elliptical, spiral, lenticular, along with various subclasses [4, 54, 130].

ANN has been applied in galaxy morphological classification with a comparable accuracy to human experts [4, 16, 66, 74, 85, 86, 87, 124]. In other cases where the initial distribution of classes is unknown, ANN has been also used in the morphological classification of Hubble Space Telescope images [4, 130]. More examples of using ANN and other supervised algorithms, namely DT and SOM, can be found in [4].

Other galaxy classifications

Beside the morphology, the spectrum of a galaxy can be used for classification [4, 80]. There are a number of studies that used machine learning and data mining algorithms in spectral classification such as PCA [17, 18, 73, 137], ANN [1, 123], and ICA [71].

There are other classification tasks in the literature than that mentioned above. Using machine learning algorithms, such as Bayesian classifier and DT, has significantly increased the new discovery of astrophysical object classes [4, 37] and the known populations of some rare object classes [4, 76].

Further classification examples in which a number of machine learning algorithms have been used, include: ANN and SVM in stellar classifications [4, 6, 136]; and DM and SVM in supernovae detection [3, 4].

Real time processing and the time domain

This simply means the study of changes in astronomical objects with time. It is a very important area of study that needs to be fully explored especially with surveys like the Large Synoptic Survey Telescope (LSST) [4, 56]. Time series analysis techniques in machine learning can be useful for real-time processing and the time domain. However, the exploration of this area comes with many challenges. For example, observations of objects can be irregularly sampled due to weather conditions or equipment availability. One of the suggested solutions for this problem is using probabilistic approaches [4, 75]. Several research studies have been done on the field of time domain. Examples include the classification of variable stars and other solar system objects. The time domain is a promising

research area with great potential of an yet unexplored parameter space that may lead to significant discoveries [4, 26]. Investigations in this field will help to explore new avenues for utilizing other information such as the variability of the objects for classification tasks [4, 31].

3.3 Machine Learning and Time Delay Estimation Problem

Recall the problem of finding the time delay between any given pair of time series obtained from the images of a gravitationally lensed quasar (see Chapter 2 on page 6). We will review the Kernel based method with variable width (K-V) which is one of the most accurate method for delay estimation. It is a novel approach based on kernel methods in the context of machine learning as proposed in [24]. It is combined later with an evolutionary algorithm (EA) for parameter optimization [25]. In this section, we present an overview of these kernel based methods and their performance based on synthetic and real data.

3.3.1 Kernel based approaches for time delay estimation

The observed fluxes of two images A and B of the same distant sources are modeled as two time series:

$$x_A = h_A(t_i) + \varepsilon_A(t_i) \quad \text{and} \quad x_B = h_B(t_i) \ominus M + \varepsilon_B(t_i) \tag{3.1}$$

where $\ominus$ denotes either multiplication or subtraction. Hence M can be either a ratio or an offset between the images, where $\{t_i\}_{i=1}^{n}$ are the observational times and $\varepsilon_A(t_i)$ and $\varepsilon_B(t_i)$ are the observation errors at t_i which are modeled as zero-mean Gaussian distributions

$$N(0, \sigma_A(t_i)) \quad \text{and} \quad N(0, \sigma_B(t_i)) \tag{3.2}$$

for $\varepsilon_A(t_i)$ and $\varepsilon_B(t_i)$ respectively. The "underlying" light curve that underpins image A can be modeled as

$$h_A(t_i) = \sum_{j=1}^{N} \alpha_j K(c_j, t_i) \tag{3.3}$$

Given the delay δ, a time-delayed ,by δ, version of $h_A(t_i)$ underpinning image B can be modeled as

$$h_B(t_i) = \sum_{j=1}^{N} \alpha_j K(c_j + \Delta, t_i). \tag{3.4}$$

The function $K(.,.)$ is a Gaussian kernel of the form

$$K(c,t) = exp\frac{-|t-c|^2}{r_c^2} \tag{3.5}$$

where $r_c > 0$ is the kernel width, $\{c_j\}_{j=1}^N$ and $\{c_j + \Delta\}_{j=1}^N$ are the kernels centers for h_A (3.3) and h_B (3.4) respectively. The functions h_A and h_B are formulated within the generalized linear regression framework [24].

Given the observed data, the likelihood of the model reads:

$$P(\text{Data}|\text{Model}) = \prod_{i=1}^{n} (x_A(t_i), x_B(t_i)|\Delta, \{\alpha_j\}), \tag{3.6}$$

where

$$\begin{aligned} p(x_A(t_i), x_B(t_i)|\Delta, \{\alpha_j\}) = & \frac{1}{2\pi\sigma_A^2(t_i)\sigma_B^2(t_i)} \\ & exp\{\frac{-(x_A(t_i) - h_A(t_i))^2}{2\sigma_A^2(t_i)}\} \\ & exp\{\frac{-(x_B(t_i) - M \ominus h_B(t_i))^2}{2\sigma_B^2(t_i)}\} \end{aligned} \tag{3.7}$$

The negative log likelihood simplifies to:

$$Q = \sum_{i=1}^{n} \left(\frac{(x_A(t_i) - h_A(t_i))^2}{\sigma_A^2(t_i)} + \frac{(x_B(t_i) - M \ominus h_B(t_i))^2}{\sigma_B^2(t_i)} \right) \tag{3.8}$$

In order to avoid extrapolation, Q (3.8) (which represents the 'goodness of fit') should not be evaluated over all observations, (3.8) can be replaced with:

$$Q = \sum_{u=1}^{n-b_1} \frac{(x_A(t_u) - h_A(t_u))^2}{\sigma_A^2(t_u)} + \sum_{v=b_2}^{n} \frac{(x_B(t_v) - M \ominus h_B(t_v))^2}{\sigma_B^2(t_v)} \tag{3.9}$$

where b_1 is the greatest index that satisfying $t_{n-b_1} \leq t_n - \Delta_{max}$ and b_2 is the smallest index that satisfying $t_{b_2} \geq t_1 + \Delta_{max}$ where Δ_{max} is the maximum value of time delay Δ trial values.

This model has N free parameters α_j collected in vector $\boldsymbol{\alpha}$ that need to be determined. Rewriting (3.8) as:

$$Q = \sum_{i=1}^{n} \left(\left[\frac{x_A(t_i)}{\sigma_A(t_i)} - \frac{h_A(t_i)}{\sigma_A(t_i)} \right]^2 + \left[\frac{x_B(t_i)}{\sigma_B(t_i)} - \frac{M \ominus h_B(t_i)}{\sigma_B(t_i)} \right]^2 \right) \tag{3.10}$$

By setting each term of (3.10) equal to zero, and replacing (3.3) and (3.5) into (3.10) , we obtain:

$$\mathbf{K}\boldsymbol{\alpha} = \boldsymbol{x}, \tag{3.11}$$

where $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_N)^\intercal$,

$$\mathbf{K} = \left[\begin{array}{ccc} K_A(c_1,t_1) & \ldots & K_A(c_N,t_1) \\ \ldots & \ldots & \ldots \\ K_A(c_1,t_n) & \ldots & k_A(c_N,t_n) \\ \hline K_B(c_1,t_1) & \ldots & k_B(c_N,t_1) \\ \ldots & \ldots & \ldots \\ K_B(c_1,t_n) & \ldots & K_B(c_N,t_n) \end{array}\right], \qquad \boldsymbol{x} = \left[\begin{array}{c} \frac{x_A(t_1)}{\sigma_A(t_1)} \\ \ldots \\ \frac{x_A(t_n)}{\sigma_A(t_n)} \\ \hline \frac{x_B(t_1)}{\sigma_B(t_1)} \\ \ldots \\ \frac{x_B(t_n)}{\sigma_B(t_n)} \end{array}\right]$$

kernels $K_A(.,.)$ and $K_B(.,.)$ are in the following forms:

$$K_A(c,t) = \frac{K(c,t)}{\sigma_A(t)} \quad \text{and} \quad K_B(c,t) = \frac{M \ominus K(c+\Delta,t)}{\sigma_B(t)} \tag{3.12}$$

Hence,

$$\boldsymbol{\alpha} = \mathbf{K}^{+}\boldsymbol{x} \tag{3.13}$$

where $\mathbf{K}^{+}$ is the pseudo inverse (or Moore-Penrose pseudo-inverse) of $\mathbf{K}$ [21, 39, 40, 65, 102, 104]. The inversion $\mathbf{K}$ is regularized through the singular value decomposition (SVD), with θ is the regularization parameter (the number of singular values to set to zero) [20, 49].

For the Gaussian kernel (3.5), two parameters need to be determined: kernels centers and widths [24]. Centers $\{c_j\}_{j=1}^{N}$ are positioned at the observational times $\{t_i\}_{i=1}^{n}$. According to [24], this approach of positioning the kernels outperformed other approaches such as regular distribution of kernels across the observations period.

It is important to determine the kernel width, since it is the smoothing parameter of the light curves (3.3) and (3.5). Two approaches have been proposed to determine the optimal width r:

- Fixed width r, K-fold cross validation algorithm can be used to determine r.
- Variable widths $\{r_j\}_{j=1}^{N}$, since the center of each kernel is positioned at the observational time, the cumulative kernel width can be determined as follow:

$$r_j = \sum_{d=1}^{k} (t_{j+d} - t_{j-d}). \tag{3.14}$$

  where k is a smoothing parameter referring to the number d of neighboring observations of c_j. The value of k can be optimized using the cross validation algorithm.

This model can be referred to as : K-F and K-V. That is, K-F corresponds to Gaussian kernels centered at observations with fixed width, and K-V has variable width.

To summarize, the aim of this model is to determine the time delay between the two light curves x_A and x_B. For every test value Δ, we determine the model parameters $\boldsymbol{\alpha}$ (3.13) and evaluate Q (3.10). The estimated time delay is the one with the minimal Q for the optimized $\boldsymbol{\alpha}$.

3.3.2 Evolved kernel based approaches

The proposed (EA) in [25] comes from genetic algorithms (GAs) [38, 53] and is based on kernel methods presented in the previous section. The idea behind this algorithm is to optimize the parameters of kernel methods (K-F) and (K-V). EA is used to evolve and optimize the parameters of the kernel-based formulation. Following the kernel-based approach in Section 3.3.1, there are three parameters for the model: the time delay Δ, the smoothing parameter k (3.14) and the regularization parameter θ. As seen in (Section 3.3.1), the inversion of (3.13) is regularized through singular value decomposition (SVD). The singular values less than the threshold λ are set to zero and the regularization parameter θ represents the number of singular values to set to zero. The amount of singular values to keep may vary depending on the value of Δ. The proposed evolutionary algorithm (EA) performs a stochastic global search and optimization methods based on Evolutionary Computation in order to find a proper combination of these three parameters.

For EA, two types of representations are used : real and mixed (real and integer) representation. Each combination of parameters (θ, Δ, k) in addition to the predefined fitness function $f(x)$ represents one individual in the population P1:

$$\boldsymbol{P}_1 = \left[\begin{array}{ccc|c} \Delta_1 & \theta_1 & k_1 & f_1 \\ \Delta_2 & \theta_2 & k_2 & f_2 \\ \cdots & \cdots & \cdots & \cdots \\ \Delta_x & \theta_x & k_x & f_x \\ \cdots & \cdots & \cdots & \cdots \\ \Delta_{np} & \theta_{np} & k_{np} & f_{np} \end{array}\right]$$

The population $\boldsymbol{P}_1$, contains np individuals. The set of parameters Δ_x, θ_x, k_x in each individual is initialized randomly.

The fitness function $f(x)$ is used to evaluate the individuals in $\boldsymbol{P}_1$. Genetic operators, such as selection, crossover and mutation, are used to generate other population

$\boldsymbol{P}_2, \ldots \boldsymbol{P}_{gn}$, where gn is the total number of generations. The best individual is chosen according to its fitness from the last population $\boldsymbol{P}_{gn}$.

Two objective functions are used to measure the fitness of the individuals [25]. :

- Negative log-likelihood (LL), given by (3.10) in Section 3.3.1.

$$Q = \sum_{i=1}^{n} \left(\left[\frac{x_A(t_i)}{\sigma_A(t_i)} - \frac{h_A(t_i)}{\sigma_A(t_i)} \right]^2 + \left[\frac{x_B(t_i)}{\sigma_B(t_i)} - \frac{M \ominus h_B(t_i)}{\sigma_B(t_i)} \right]^2 \right).$$

- The mean squared error (MSE) given by cross validation algorithm (CV) illustrated in Algorithm 1 proposed [25].

Algorithm 1 CV for fitness function

Require: $\mathcal{A}$ the data set of all observations; its cardinality is n.

Ensure: f_x

1: Fix $\mathcal{B} \leftarrow 5$;
2: Fix $\mathcal{L} \leftarrow n/\mathcal{B}$;
3: **for** $i \in \{1, ..., \mathcal{L}\}$ **do**
4: Remove the i^{th} observation of each block and include it in the validation set $\mathcal{V}$;
5: Compute h_A via (3.3) and h_B via (3.4) for the training set $\mathcal{T} = \mathcal{A} - \mathcal{V}$;
6: Obtain MSE_{CV} on the validation set $\mathcal{V}$
7: $\mathcal{R}(i) \leftarrow MSE_{CV}$
8: **end for**
9: $f_x \leftarrow \text{mean}(\mathcal{R})$
10: return f_x

For selection, a roulette wheel method is used and the probability of parents being selected depends on their fitness value. For crossover, linear and double point recombination are used for real and integer respectively. A mutation function from the Genetic Algorithm Toolbox for MATLAB called $Mutbga$ is used as the mutation operator. It is based on Breeder Genetic Algorithm (BGA) [81]. In the BGA mutation, the mutated

variable z_i is determined as follows:

$$z_i = x_i \pm range_i.\delta$$

where x_i is the variable to mutate and $\pm$ sign is selected with probability 0.5 and $range_i$ is the mutation range computed as

$$range_i = 0.5 \times d_i$$

where d_i is the domain of variable x_i. δ is determined as follows

$$\delta = \sum_{i=0}^{m-1} \alpha_i 2^{-i}$$

where $\alpha_i = 1$ with probability 1/m and zero otherwise. Finally, The best individuals (offspring) from the current population are reinserted into the next one [13, 14] cited in [25].

3.3.3 Performance of kernel based approaches

Large scale controlled experiments have been performed on a wide range of (synthetic and real astronomical observations) data sets to compare the accuracy of kernel based methods : K-F, K-V and EAs with that of other methods used in literature for time delay estimation (see Chapter 2). Here we present a summary of their results and final conclusion; more detailed analysis of the experimental results can be found in [21, 24, 25]. These experiments have been conducted on different types of artificially generated data, which include DS-500, DS-5, PHR and Harva data. The results of experiments on synthetic data led to the conclusion that, for all methods included the accuracy of estimation is affected by noise levels and gaps sizes. In other words, "increasing of noise levels and gaps size in the data sets result in increased uncertainty of the time delay estimates" [21]. The performance of Kernel based methods (K-V) and (EA) is more

statistically significant than the other used methods. Kernel based methods have proven to be promising, robust, and accurate time delay estimation methods considering the use of noisy and irregularly sampled data.

Regarding real data, optical and radio data sets from quasar Q0957+561 are used in the experiments (see Chapter 2) for more details on real data). Results are summarized in Tables 3.1 and 3.2 [21]. In Table 3.2 μ and σ denote the mean and standard deviation of estimates from 500 Monte Carlo simulations (MC). The final conclusion, and based on D^4, the time delay for Q0957+561 is claimed to be 419.6 days where a time delay of 417 days was reported on DS1 [64].

Table 3.1: Time delay estimates for real data sets.

Data	K-V	EA
D^2	435	428.8-429.2
D^3	420	418.1-420.3
D^4	420	419.6
D^5	449	494.4-476.4
D^6	409	396.6-397.2

Table 3.2: Results of 500 Monte Carlo simulations.

Data	K-V	EA
D^2	436.6±6.1	432.4±8
D^3	420.9±4	420.5±4
D^4	419.5±0.7	422.3±4
D^5	449.4±27	451.5±25
D^6	408.9±11	393.8±12

As mentioned in Chapter 1, the computational time complexity of K-V and EA methods are $O(n^5)$ and $O(n^6)$ respectively [21]. This restriction makes it unable to deal with long time series. The complexity is due to the inverse matrix in kernel based methods for weights estimation. Automatic methods for time delay estimation have been proposed to speed up algorithms and they are able to deal with long time series, based on ANN [41]. Moreover, a parallel version of these algorithms have been proposed [22].

3.4 Summary

In this chapter, we presented an overview of the main concepts of machine and statistical learning. We have briefly introduced the use of machine learning and data mining algorithms in astronomy and presented many examples of their applications into different astronomical problems. We also listed the advantages of using these algorithms and their limitations.

Then we reviewed the kernel methodology based on kernel linear regression and evolutionary algorithms, proposed by [24, 25], for time delay estimation in gravitationally lensed signals. The basic definitions, notations, and concepts associated with kernel based approach (K-V) for time delay estimation have been presented in Section (3.3.1) . We also reviewed the evolved kernel based approach (EA) in detail. The parameters to evolve, representation, fitness functions and evolutionary operators have been described in Section (3.3.2). The chapter ended with performance analysis of kernel based methods (K-V) and (EA). The kernel based approaches are proven to be the most accurate and stable methodologies for time delay estimation between multiple images of a gravitationally lensed quasar based on the results of the experiments on artificially generated data.

The main disadvantage of these approaches is that they are expensive in terms of computational time complexity due to matrix inversion. This restriction makes large data sets become intractable. This is one of the main concerns to deal with in the extensions of this work. In the next chapter, we will introduce new methods within the kernel regression framework that is faster and more efficient in dealing with large data sets.

CHAPTER 4

ESTIMATING TIME DELAYS IN DAILY OBSERVATION WITH NOISE AND OBSERVATIONAL GAPS

In this chapter, we propose a new approach based on kernel regression estimates, which is able to estimate a single time delay given several data sets for the same quasar. We develop realistic artificial data sets in order to perform controlled experiments to test the performance of this new approach. We also test our method on real data from strongly lensed quasar Q0957+561 and compare our estimates against existing results.

The proposed models in this chapter are based on kernel regression and within the same framework as the previous kernel based approaches that introduced in literature review (see Chapter 3). The main concern with the previously proposed kernel based approaches is expensive from a computational time perspective due to the presences of matrix inversion for weights estimation. Our new approaches addresses this problem and noticeably reduced the computational time as a result of eliminating the matrix computations.

4.1 The Model

We consider a distant point source (e.g. a quasar) with two strongly lensed images[1], referred to as A and B, and one or more time series of flux measurements, possibly taken by different instruments and/or at different frequencies. The entire data collection $D = \{D^1, D^2, \dots D^L\}$ consists of L data sets D^ℓ, $\ell \in [1, L]$, each corresponding to a sequence of measurements taken with a given instrument and at a given frequency. Data sets D^ℓ consist of flux measurements of both images, y^ℓ_A and y^ℓ_B, taken at a non-uniform sequence of N^ℓ observational times times $t^\ell_1, t^\ell_2, ..., t^\ell_{N^\ell}$.

Formally, each set D^ℓ contains N^ℓ three-tuples
$(t^\ell_k, y^\ell_{A,k}, y^\ell_{B,k})$, $k = 1, 2, \dots N^\ell$,

$$D^\ell = \{(t^\ell_1, y^\ell_{A,1}, y^\ell_{B,1}), (t^\ell_2, y^\ell_{A,2}, y^\ell_{B,2}), \dots (t^\ell_{N^\ell}, y^\ell_{A,N^\ell}, y^\ell_{B,N^\ell})\},$$

where $y^\ell_{A,k}$ and $y^\ell_{B,k}$ denote the observed fluxes of image A and B, respectively, in D^ℓ at time t^ℓ_k. We also assume that the standard errors $\sigma^\ell_{A,k}$ and $\sigma^\ell_{B,k}$ are known for each observation $y^\ell_{A,k}$ and $y^\ell_{B,k}$, respectively.

The fluxes corresponding to the two images A and B are collected in sets

$$D^\ell_A = \{(t^\ell_1, y^\ell_{A,1}), (t^\ell_2, y^\ell_{A,2}), \dots (t^\ell_{N^\ell}, y^\ell_{A,N^\ell})\}$$

and

$$D^\ell_B = \{(t^\ell_1, y^\ell_{B,1}), (t^\ell_2, y^\ell_{B,2}), \dots (t^\ell_{N^\ell}, y^\ell_{B,N^\ell})\}.$$

For observations at frequencies above a few tens of MHz, dispersion yields sub-hour arrival time differences, and is not significant relative to typical time-delay measurement accuracy. We therefore assume that the time delay between gravitationally lensed fluxes does not depend on the wavelength at which the observations are taken. We also assume stationarity of the lensing object (e.g., a galaxy) in the sense that the delay does not

[1] generalization to four images is straightforward

change in time; in particular, we ignore micro-lensing contributions.

4.1.1 Nadaraya-Watson Estimator with Known Noise Levels (NWE)

Given a delay Δ, we seek to find a probabilistic model $p(D|\Delta)$ that explains[1] D. Assuming independence of the observation sets D^ℓ, we obtain

$$p(D|\Delta) = \prod_{\ell=1}^{L} p(D^\ell|\Delta).$$

Assuming independent observations at distinct measurement times, we get

$$p(D^\ell|\Delta) = \prod_{k=1}^{N^\ell} p(y^\ell_{A,k}, y^\ell_{B,k}|t^\ell_k, \Delta)$$

and further assumption of independence of measurement noise in images A and B leads to

$$p(y^\ell_{A,k}, y^\ell_{B,k}|t^\ell_k, \Delta) = p_A(y^\ell_{A,k}|t^\ell_k, \Delta)\ p_B(y^\ell_{B,k}|t^\ell_k, \Delta).$$

Modeling the source using image A

It is typically assumed that the measurement uncertainties on fluxes D^ℓ_A and D^ℓ_B are normally distributed, with zero mean Gaussian noise of known standard deviation $\sigma^\ell_{A,k}$ and $\sigma^\ell_{B,k}$ associated with noisy observations $y^\ell_{A,k}$ and $y^\ell_{B,k}$, respectively. We model the mean of image A using Nadaraya-Watson kernel regression [84], [133],

$$f^\ell_A(t) = \frac{\sum_{k=1}^{N^\ell} y^\ell_{A,k}\ K(t, t^\ell_k; h^\ell)}{\sum_{j=1}^{N^\ell} K(t, t^\ell_j; h^\ell)}, \tag{4.1}$$

[1] We slightly abuse mathematical notation as we are actually building conditional models of flux values, given the observation times.

where $f^\ell(t)$ is the predicted flux at time t and $K(t, t_j^\ell; h^\ell)$ is a kernel positioned at t_j^ℓ with bandwidth parameter h^ℓ. We use the Gaussian kernel

$$K(t, t_k; h) = \exp\left\{-\frac{(t-t_k)^2}{\kappa^2(t_k)}\right\},$$

where the kernel scale $\kappa(t_k)$ at position t_k is defined as the distance spanned by the h neighbors (to the left and to the right) of t_k, i.e. $\kappa(t_k) = t_{k+h} - t_{k-h}$. This approach to modeling the noise should work when the autocorrelation length of the observed flux is much longer than any gaps in the data during which the flux is modeled via the Nadaraya-Watson kernel regression estimator. If the autocorrelation length of the observed flux, which can be estimated from a time interval when the observations are relatively closely spaced, is comparable to or larger than a data gap, this approach (or any other approach that does not incorporate a physically accurate flux model) cannot be trusted.

To respect the nature of gravitationally lensed data, we impose that the mean model for image B follows exactly that for image A, up to scaling by a constant[1] $M > 0$ and time shift by Δ:

$$f_B^\ell(t; \Delta) = M\ f_A^\ell(t - \Delta).$$

Since the shift Δ plays no role in modeling image A, we write

$$p(y_{A,k}^\ell, y_{B,k}^\ell | t_k^\ell, \Delta) = p_A(y_{A,k}^\ell | t_k^\ell)\ p_B(y_{B,k}^\ell | t_k^\ell, \Delta), \tag{4.2}$$

where

$$p_A(y_{A,k}^\ell | t_k^\ell) = \frac{1}{\sqrt{2\pi}\ \sigma_{A,k}^\ell}\ \exp\left\{-\frac{1}{2}\frac{(y_{A,k}^\ell - f_A^\ell(t_k^\ell))^2}{(\sigma_{A,k}^\ell)^2}\right\} \tag{4.3}$$

and

[1] assumed known, or easily estimated in a preprocessing stage using the means of the fluxes in D_A^ℓ and D_B^ℓ

$$p_B(y^\ell_{B,k}|t^\ell_k,\Delta) = \frac{1}{\sqrt{2\pi}\ \sigma^\ell_{B,k}}\ \exp\left\{-\frac{1}{2}\frac{(y^\ell_{B,k} - Mf^\ell_A(t^\ell_k-\Delta))^2}{(\sigma^\ell_{B,k})^2}\right\}. \tag{4.4}$$

Note that given Δ, the only free parameter of $p(y^\ell_{A,k}, y^\ell_{B,k}|t^\ell_k,\Delta)$ is the kernel width parameter h^ℓ in the formulation of the mean model (4.1).

Ignoring constant terms and scaling, the negative log likelihood, $-\log p(D^\ell|\Delta)$, forms the approximation error for the set D^ℓ,

$$E^\ell_A(h^\ell;\Delta) = \sum_{k=1}^{N^\ell}\left\{\frac{(y^\ell_{A,k} - f^\ell_A(t^\ell_k))^2}{(\sigma^\ell_{A,k})^2} + \frac{(y^\ell_{B,k} - Mf^\ell_A(t^\ell_k-\Delta))^2}{(\sigma^\ell_{B,k})^2}\right\}. \tag{4.5}$$

Writing down the negative log likelihood for the whole data, $-\log p(D|\Delta)$, and ignoring scaling and constant terms leads to the total approximation error

$$E_A(\boldsymbol{h};\Delta) = \sum_{\ell=1}^{L} E^\ell_A(h^\ell;\Delta),$$

where $\boldsymbol{h} = (h^1, h^2, \ldots h^L)$ is a vector that collects kernel width parameters for all data sets $D^1, D^2, \ldots D^L$ in D.

Modeling the source using image B

One can, of course, start by building a mean flux model $f^\ell_B(t)$ for image B via Nadaraya-Watson kernel regression,

$$f^\ell_B(t) = \frac{\sum_{k=1}^{N^\ell} y^\ell_{B,k}\ K(t,t^\ell_k;h^\ell)}{\sum_{j=1}^{N^\ell} K(t,t^\ell_j;h^\ell)}, \tag{4.6}$$

imposing that the mean model of image A is

$$f^\ell_A(t;\Delta) = \frac{1}{M}\ f^\ell_B(t+\Delta).$$

Crucially, since both images A and B come from the same source, we require that the kernel width h^ℓ for the mean models $f^\ell_A(t)$ and $f^\ell_B(t)$ (and hence for $f^\ell_A(t;\Delta)$ and $f^\ell_B(t;\Delta)$

as well) be the same for the whole data set D^ℓ.

Using the same reasoning as in Section 4.1.1, we obtain an approximation error for the set D^ℓ:

$$E_B^\ell(h^\ell;\Delta) = \sum_{k=1}^{N^\ell} \left\{ \frac{(y_{A,k}^\ell - \frac{1}{M} f_B^\ell(t_k^\ell + \Delta))^2}{(\sigma_{A,k}^\ell)^2} + \frac{(y_{B,k}^\ell - f_B^\ell(t_k^\ell))^2}{(\sigma_{B,k}^\ell)^2} \right\}$$

leading to the total approximation error

$$E_B(\boldsymbol{h};\Delta) = \sum_{\ell=1}^{L} E_B^\ell(h^\ell;\Delta).$$

Estimating the Unique Time Delay across D

Since there is no a-priori reason to prefer one image over the other, we aim to find the unique delay Δ that minimizes both the errors $E_A(\boldsymbol{h};\Delta)$ and $E_B(\boldsymbol{h};\Delta)$ with the same 'level of importance'. In other words, we are looking for Δ and the set of kernel width parameters $\boldsymbol{h} = (h^1, h^2, \ldots h^L)$, one for each data set D^ℓ in D, that minimize the error

$$E(\boldsymbol{h};\Delta) = E_A(\boldsymbol{h};\Delta) + E_B(\boldsymbol{h};\Delta).$$

Note that the imposition that there is a unique delay Δ for the whole data D and that the kernel widths are the same throughout each set D^ℓ for all the corresponding mean models $f_A^\ell(t)$, $f_B^\ell(t)$, $f_A^\ell(t;\Delta)$ and $f_B^\ell(t;\Delta)$, not only makes sense from the point of view of underlying physics, but is also a stabilizing factor in the analysis and modeling of D.

The structure of our problem enables us to use an efficient and practical approach to finding the optimal time delay Δ_*. The error $E(\boldsymbol{h};\Delta)$ to be minimized can be rewritten as

$$E(\boldsymbol{h};\Delta) = \sum_{\ell=1}^{L} E^\ell(h^\ell;\Delta), \tag{4.7}$$

where

$$E^\ell(h^\ell;\Delta) = E_A^\ell(h^\ell;\Delta) + E_B^\ell(h^\ell;\Delta).$$

For every test value Δ we can separately optimise $E^\ell(h^\ell;\Delta)$ for h^ℓ within each set D^ℓ. Note that this boils down into a set of L one-dimensional optimizations of bandwidths $h^1, h^2, \ldots h^L$. In addition, because of the nature of the mean models, the errors $E^\ell(h^\ell;\Delta)$ will behave 'reasonably' with changes in h^ℓ, i.e. the changes will be smooth and we can expect a roughly unimodal shape of cross-validated $E^\ell(h^\ell;\Delta)$. That enables us to use further speed-up tricks (such as halving) in the 1-dimensional optimizations. The estimated time delay is the one with the minimal overall $E(\boldsymbol{h};\Delta)$ for the (cross-validation) optimized kernel width parameters $\boldsymbol{h}$.

4.2 Nadaraya-Watson Estimator with Linear Noise Model (NWE++)

In Section 4.1 only the mean fluxes were modeled, the standard errors on observations were assumed known. Our approach can be extended to full probabilistic modeling by assuming a model for the relationship between the noise level and the observed fluxes. Here, we consider a simple model in which the standard error on the measured flux depends linearly on the observed flux value y, i.e., $\sigma(y) = \nu y$, where the proportionality constant ν depends on the wavelength at which the flux is measured (e.g., ν could be 1% and 0.1% for radio and optical data, respectively). Note that, this general noise model is just an assumption. Assuming that the mean models for data set D^ℓ are fitted reasonably well, so that $y^\ell_{I,k} \approx f^\ell_I(t^\ell_k)$, $I \in \{A, B\}$, then to lowest order $\sigma(y^\ell_{I,k}) \approx \nu^\ell f^\ell_I(t^\ell_k)$.

Most of the material developed in Sections 4.1 will stay unchanged; modifications are required only in the formulation of the noise models (4.3) and (4.4):

$$p_A(y^\ell_{A,k}|t^\ell_k) = \frac{1}{\nu^\ell \sqrt{2\pi}\ f^\ell_A(t^\ell_k)} \exp\left\{\frac{-1}{2(\nu^\ell)^2}\left[\frac{y^\ell_{A,k}}{f^\ell_A(t^\ell_k)} - 1\right]^2\right\} \tag{4.8}$$

and

$$p_B(y^\ell_{B,k}|t^\ell_k,\Delta) = \frac{1}{M\nu^\ell\ \sqrt{2\pi}\ f^\ell_A(t^\ell_k-\Delta)}\ \exp\left\{\frac{-1}{2(\nu^\ell)^2}\left[\frac{y^\ell_{B,k}}{Mf^\ell_A(t^\ell_k-\Delta)}-1\right]^2\right\}. \quad (4.9)$$

This time, however, we can write a full probabilistic model for any time point t and evaluate the likelihood within our model given any observation pair $(y^\ell_A(t), y^\ell_B(t))$ that could have been measured at time t:

$$p_A(y^\ell_A(t)) = \frac{1}{\nu^\ell\sqrt{2\pi}\ f^\ell_A(t)}\ \exp\left\{\frac{-1}{(\nu^\ell)^2}\left[\frac{y^\ell_A(t)}{f^\ell_A(t)}-1\right]^2\right\} \quad (4.10)$$

and

$$p_B(y^\ell_B(t)|\Delta) = \frac{1}{M\nu^\ell\ \sqrt{2\pi}\ f^\ell_A(t-\Delta)}\ \exp\left\{\frac{-1}{(\nu^\ell)^2}\left[\frac{y^\ell_B(t)}{Mf^\ell_A(t-\Delta)}-1\right]^2\right\}. \quad (4.11)$$

The approximation error $E^\ell_A(h^\ell;\Delta)$ to be minimized by the choice of kernel width h^ℓ now reads:

$$E^\ell_A(h^\ell;\Delta) = \frac{1}{(\nu^\ell)^2}\sum_{k=1}^{N^\ell}\left\{\left[\frac{y^\ell_{A,k}}{f^\ell_A(t^\ell_k)}-1\right]^2+\left[\frac{y^\ell_{B,k}}{Mf^\ell_A(t^\ell_k-\Delta)}-1\right]^2\right\}.$$

Following analogous arguments for the case of modeling the source using image B, we have

$$p_A(y^\ell_A(t)|\Delta) = \frac{M}{\nu^\ell\sqrt{2\pi}\ f^\ell_B(t+\Delta)}\ \exp\left\{\frac{-1}{(\nu^\ell)^2}\left[\frac{M\ y^\ell_A(t)}{f^\ell_B(t+\Delta)}-1\right]^2\right\} \quad (4.12)$$

and

$$p_B(y^\ell_B(t)) = \frac{1}{\nu^\ell\sqrt{2\pi}\ f^\ell_B(t)}\ \exp\left\{\frac{-1}{(\nu^\ell)^2}\left[\frac{y^\ell_B(t)}{f^\ell_B(t)}-1\right]^2\right\},$$

which leads to the approximation error

$$E_B^\ell(h^\ell;\Delta) = \frac{1}{(\nu^\ell)^2}\sum_{k=1}^{N^\ell}\left\{\left[\frac{M\ y_{A,k}^\ell}{f_B^\ell(t_k^\ell+\Delta)}-1\right]^2+\left[\frac{y_{B,k}^\ell}{f_B^\ell(t_k^\ell)}-1\right]^2\right\}.$$

Again, the final cost to be minimized is

$$E(\boldsymbol{h};\Delta) = \sum_{\ell=1}^{L} E^\ell(h^\ell;\Delta), \tag{4.13}$$

where

$$E^\ell(h^\ell;\Delta) = E_A^\ell(h^\ell;\Delta) + E_B^\ell(h^\ell;\Delta).$$

4.3 Data

We employ six different data sets from the same quasar Q0957+561, $L = 6$. The data plots are shown in Figure 2.4 in Chapter 2 on page 11 and all the details are presented in Table 2.1 in Chapter 2 on page 12.

In order to consistently compare the performance of different time delay estimation methods in a controlled experimental setting (CS), we also construct synthetic data on the basis of known gravitationally lensed fluxes in the optical and radio ranges, with the given observational noise and gaps structure. The 'ground truth' - the delay - is imposed by us so that the statistics of different delay estimators can be consistently evaluated and compared.

4.3.1 Synthetic data - realistic experimental setting

In this section we construct synthetic signals on which we will test the proposed and some of the existing approaches to gravitational delay estimation in the presence of observational noise and gaps. We constructed synthetic fluxes in the optical range on the basis of D^1 (real r-band optical data of [116]) spanning roughly 10.5 years). In particular, we

used D^1 to fit a distribution of possible fluxes 'compatible' with the data (formulated as a Gaussian process (GP)) and then sampled from this distribution synthetic fluxes of 3,500 observations.

(GP) represents a distribution over functions

$$f(t) \sim GP(\mu_{gp}(t), K_{gp}(t, t')), \tag{4.14}$$

with mean and covariance functions $\mu_{gp}(t)$ and $K_{gp}(t, t')$, respectively. Any sample from the GP corresponding to a finite set of observational times $t_1, t_2, \cdots t_N$ is Gaussian distributed with mean $\mu_{gp}(t_1), \mu_{gp}(t_2), \cdots \mu_{gp}(t_N)$ and covariance matrix

$$K_{gp} = \begin{pmatrix} K_{gp}(t_1, t_1) & K_{gp}(t_1, t_2) & \cdots & K_{gp}(t_1, t_N) \\ K_{gp}(t_2, t_1) & K_{gp}(t_2, t_2) & \cdots & K_{gp}(t_2, t_N) \\ \vdots & \vdots & \ddots & \vdots \\ K_{gp}(t_N, t_1) & K_{gp}(t_N, t_2) & \cdots & K_{gp}(t_N, t_N) \end{pmatrix}. \tag{4.15}$$

For our purposes, we imposed zero mean (the mean of observations in D^1 was shifted to zero) and used the 'squared exponential' kernel function

$$K_{gp}(t, t') = \exp\left\{-\frac{(t - t')^2}{h_{gp}^2}\right\}, \tag{4.16}$$

with scale parameter h_{gp} set using cross validation on D^1.

A vector $(\boldsymbol{y}, \boldsymbol{y}_*)^T$ of observations sampled at observation times $\boldsymbol{t}$ and $\boldsymbol{t}_*$ from the (GP) is distributed as

$$\begin{pmatrix} \boldsymbol{y} \\ \boldsymbol{y}_* \end{pmatrix} \sim N\left(\begin{pmatrix} \boldsymbol{0} \\ \boldsymbol{0} \end{pmatrix}, \begin{pmatrix} K_{gp} & K_{gp*} \\ K_{gp*}^T & K_{gp**} \end{pmatrix}\right), \tag{4.17}$$

where K_{gp}, K_{gp*} and K_{gp**} are kernel matrices corresponding to time instances $\boldsymbol{t} \times \boldsymbol{t}$, $\boldsymbol{t} \times \boldsymbol{t}_*$ and $\boldsymbol{t}_* \times \boldsymbol{t}_*$, respectively. However, given observations $\boldsymbol{y}$ at times $\boldsymbol{t}$, the conditional distribution of $\boldsymbol{y}_*$ at times $\boldsymbol{t}_*$ is given by

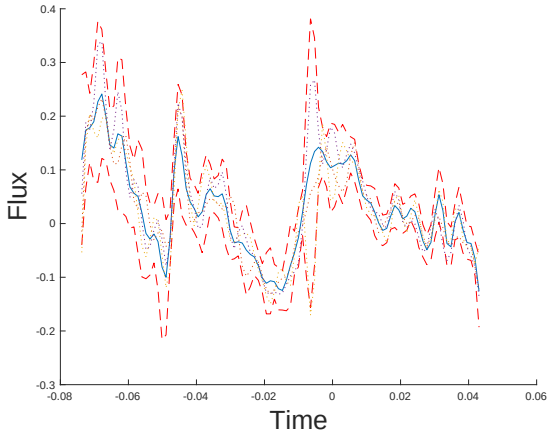


Figure 4.1: Three Gaussian process posterior samples (dotted) based on D^1 (solid). Dashed curves signify $\pm$ 2 standard deviations.

$$p(\boldsymbol{y}_*|\boldsymbol{t}_*, \boldsymbol{y}, \boldsymbol{t}) = N(\boldsymbol{y}_*|\mu_*, \Sigma_*) \tag{4.18}$$

with

$$\mu_* = K_{gp*}^T K_{gp}^{-1} \boldsymbol{y} \tag{4.19}$$

and

$$\Sigma_* = K_{gp**} - K_{gp*}^T K_{gp}^{-1} K_{gp*}. \tag{4.20}$$

We sampled signals $\boldsymbol{y}_*$ from the GP based on D^1 on a regular grid of 3500 time stamps covering the temporal range of D^1. As an example, we show three such signals in Figure 4.1. Dashed curves signify $\pm$ 2 standard deviations. To create a pair of time shifted signals A and B, the smooth long signal (signal A) $y_A = \boldsymbol{y}_*$ was shifted in time by a delay $\Delta = 200$ days to obtain signal B,

$$y_B(t) = y_A(t - \Delta). \tag{4.21}$$

Figure 4.2, shows an example of this generation process.

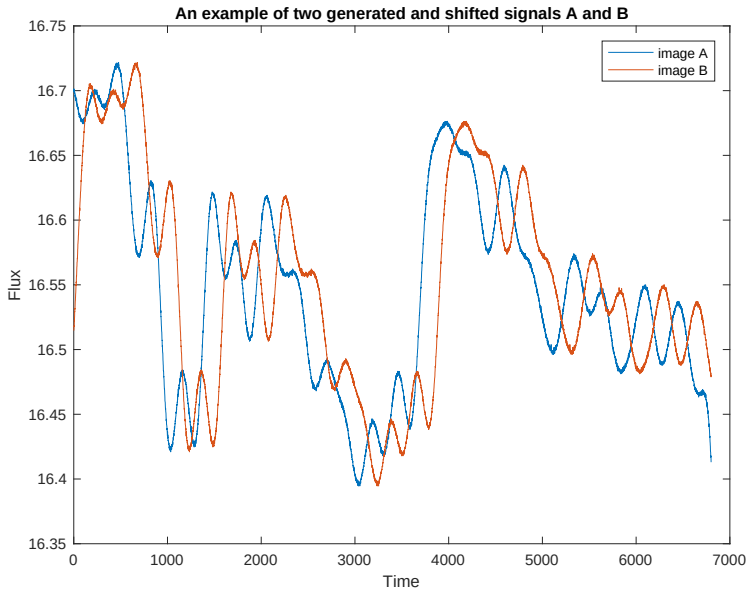


Figure 4.2: An example of two generated signals A and B. Signal B is delayed by 200 days.

Finally, (as explained in greater detail in the following sections), we added observational noise independently to both signals A and B, and imposed observational gaps.

Observational noise

Based on D^1 data, we first calculated the empirical distribution $p(\rho)$ of the ratio ρ of the reported flux levels y_k and their associated standard errors σ_k: $\rho_k = \sigma_k/y_k$. For each observation $y(t)$ in the synthetic stream, we generated an additive observational noise from a zero mean Gaussian distribution with standard deviation $\sigma(t)$, where $\sigma(t) = \rho(t)y(t)$, with $\rho(t)$ generated randomly i.i.d. from the empirical distribution $p(\rho)$. Figure 4.3, shows an example of one data set after adding the noise to the signals.

Observational gaps

Real data are irregularly sampled due to practical considerations such as weather conditions, equipment availability, object visibility, etc. [21, 29]. Gaps in real data are characterized by two important quantities: gap size and gap position. The histogram in Figure

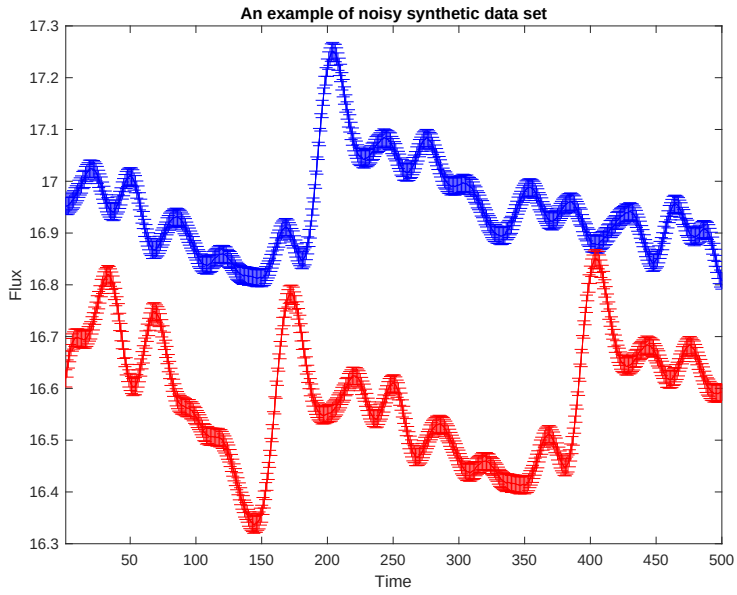


Figure 4.3: An example of the added noise; here error bars are 0.1% of the flux. Signal A has been shifted upwards by 0.4 for visualization.

4.4(a) shows the empirical gap size distribution in D^1. Shorter gaps of 1–5 days are more frequent than longer ones (more than 6 days).

To make the synthetic data more realistic, we would like to respect constraints given by the gap size and inter-gap distance distributions for dominant gap sizes (up to 10 days). Gaps were imposed on the synthetic data by generating their sizes and positions through a multiobjective optimization algorithm. The algorithm incorporated three constraints: (1) closeness of the generated and empirical gap size distributions; (2) closeness of the generated and empirical inter-gap interval distributions for gaps of 1-5 days; (3) closeness of the generated and empirical inter-gap interval distributions for gaps of 6-10 days.

The particular algorithm we used was the computationally efficient *Random Weighted Genetic Algorithm (RWGA)* [36, 61, 82, 83, 138]. It uses a weighted average of normalized objectives for fitness assignment (for diversity imposition the weights are randomized). The procedure is outlined in Algorithm 2.

Algorithm 2 RWGA

1: S = external archive to store non-dominated solutions found during the search so far;

2: n_S = number of randomly selected solutions immigrating from S to the population of potential solutions X_ι in each generation ι.

3: Generate a initial random population X_1, set $\iota = 1$.

4: Assign a fitness value to each individual solution $\chi \in X_\iota$ by performing the following steps:

Step 4.1: Calculate the fitness $z_o(\chi)$ for each objective $o = 1, \ldots O$.

Step 4.2: Generate a random number u_o in $[0, 1]$ for each objective $o = 1, \ldots O$

Step 4.3: Calculate the random weight of each objective o as

$$w_o = \frac{u_o}{\sum_{i=1}^{O} u_i}$$

.

Step 4.4: Update the overall fitness of the solution χ as

$$F(\chi) = \sum_{o=1}^{O} w_o z_o(\chi)$$

.

5: Calculate the selection probability $p_s(\chi)$ of each solution $\chi \in X_\iota$ as follows:

$$p_s(\chi) = \frac{\sum_{\Upsilon \epsilon X_\iota} (F(\Upsilon) - F^{\min})}{F(\chi) - F^{\min}},$$

where $F^{\min} = \min \{F(\chi) \mid \chi \in X_\iota\}$.

6: Select parents using the selection probabilities calculated in Step 3. Mutate offspring with a predefined mutation rate. Copy all offspring to $X_{\iota+1}$.

7: To maintain diversity, randomly remove n_S solutions from $X_{\iota+1}$ and add the same number of solutions from S to $X_{\iota+1}$.

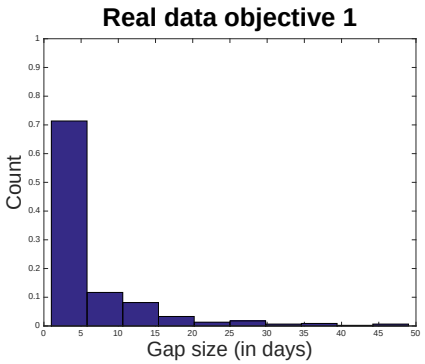
8: If the stopping condition is not satisfied, set $\iota = \iota + 1$ and go to Step 4. Otherwise, return to S.

The genome of each individual contains a suggestion for start positions and sizes of observational gaps. The design of individuals allows for a variable number of gaps and ensures that the gaps are not overlapping. Figure 4.4 shows the results of applying the multi-objective genetic algorithm RWGA based on D^1. Each objective corresponds to a row of two plots in Figure 4.4, left and right plots showing empirical normalized histograms from the real and synthetic data, respectively.

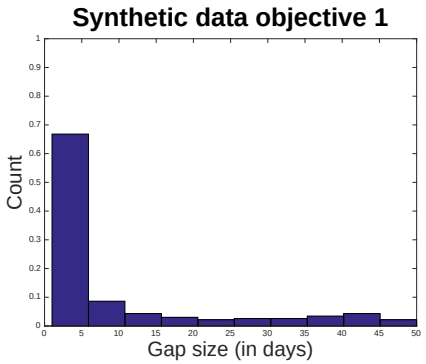
Generation of synthetic 'radio' data proceeded in the same way as described in the previous section for optical data, this time based on data D^5.

4.3.2 Synthetic data - controlled experimental setting

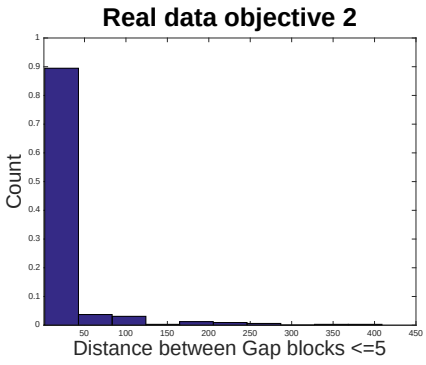
Generation of synthetic fluxes described above was motivated by the desire to preserve realistic gap and noise distributions. We will refer to this approach as the *'realistic' experimental setting* (RS). For comparing delay estimation algorithms in a large-scale controlled setting, we also considered an alternative specification of gap and noise distributions. The synthetic fluxes were first generated from the GP model fitted to D^1, as described in the previous section. The fluxes were then corrupted with observational gaps and noise. The gap sizes g were generated as realizations from a mixture distribution $P_M(g) = \alpha P_B(g; \mu_g) + (1 - \alpha)P_U(g; L_g, U_g)$, where $P_B(g; \mu_g)$ is the Binomial distribution with mean μ_g and $P_U(g; L_g, U_g)$ is the uniform distribution over $[L_g, U_g]$. We used the following settings: $\alpha = 0.95$, $\mu_g = 4, 6, 8$ days, $L_g = 20$ and $U_g = 80$. The gap positions were randomized, subject to the constraint of minimum inter-gap distance of 2 days. The allowed range for gap size was 1 to 80 days. For the additive Gaussian zero mean 'observational' noise, we considered three settings for the standard deviation: 0.1%, 0.2% and 0.3% of the flux level. We will refer to this approach as the *'controlled' experimental setting*.



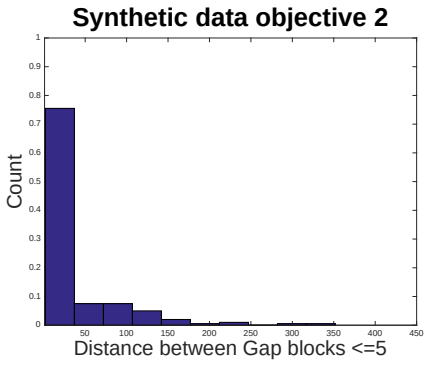
(a) Objective 1



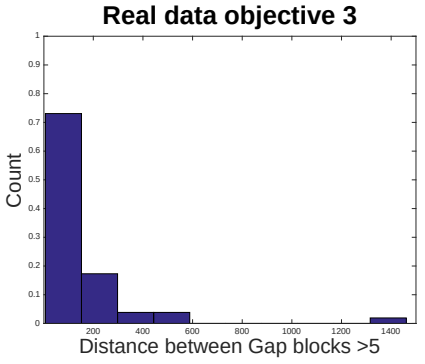
(b) Objective 1



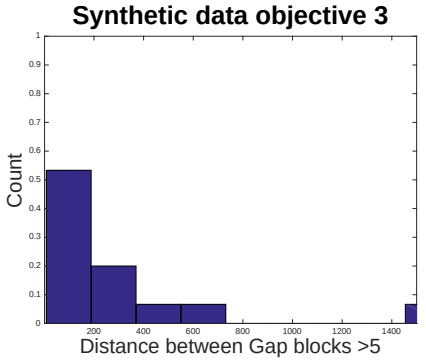
(c) Objective 2



(d) Objective 2



(e) Objective 3



(f) Objective 3

Figure 4.4: Empirical distributions of gap size (a),(b), inter-gap interval for gaps of 1–5 days (c),(d) and inter-gap interval for gaps of 6-10 days (e),(f). Each objective of RWGA corresponds to a row of two plots, left and right plots showing empirical normalized histograms from the real (D^1) and synthetic data, respectively.

4.4 Experimental Results

We performed experiments on synthetic data sets described in Section 4.3, as well as on real gravitationally lensed fluxes in the radio and optical ranges. In the experiments we compared our methods NWE and NWE++, introduced in Sections 4.1 (on page 37) and 4.2 (on page 42), respectively, with two DS approaches, namely DS_1^2 and $\mathrm{DS}_{2,4}^2$ (Section 2.3.2 on page 16) and two cross-correlation approaches DCF and LNDCF (Section 2.3.1 on page 15).

4.4.1 Experiments on synthetic data

As mentioned above, we set the 'true' time delay in the synthetic data to 200 days. The results of all approaches are based on testing time delay values in the range of 175 to 225 days (1 day increment).

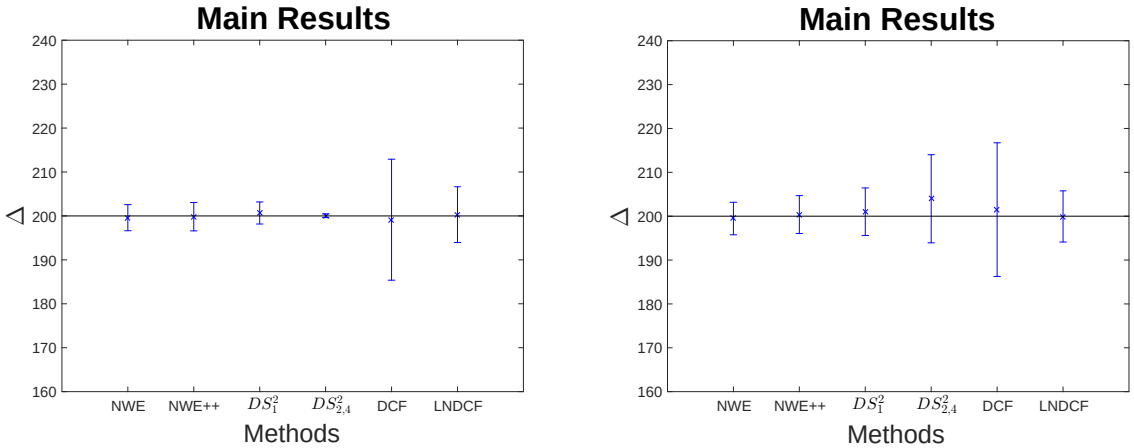
It was found that the best setting for decorrelation length δ in the $\mathrm{DS}_{2,4}^2$ method was 3 days. For NWE and NWE++ the kernel width h was estimated as variable kernel width with $h = 2$ neighbors[1]. The proportionality constant ν for NWE++ is set to 1% and 0.1% of the flux for radio and optical data, respectively. For DCF and LNDCF, the bin size is set to 5 days. (see [24]).

For each method we show the mean (bias) μ and standard deviation σ of the maximum-likelihood delay estimates across experiments. In all plots, the true delay is represented by the horizontal line at $\mu = 200$.

Realistic experimental setting

For synthetic experiments in the realistic setting we generated 500 base signals from the GP fitted to the optical data set D^1, as described in Section 4.3.1. We then ran the RWGA algorithm to generate 500 realizations for observational gap positions and sizes (see Section 4.3.1). Each base signal thus had a corresponding observational gap

[1]Two neighbors came consistently as the favorite option when cross-validating the number of neighbors on several initial data sets.



(a) Optical (b) Radio

Figure 4.5: RS results for optical and radio data.

structure imposed on it. Finally, the signals were corrupted by observational noise (see Section 4.3.1). The same procedure was applied for generating 500 data sets in the radio range.

Summary results for the RS experiments on the 500 optical and radio data sets are presented in Tables 4.1 and 4.2, respectively. We report the mean (μ) and standard deviation (σ) of the delay estimates $\hat{\Delta}_i$, $i = 1, 2, \ldots 500$, the mean absolute error (MAE) of the delay estimates (MAE= $\sum_{i=1}^{500} |\hat{\Delta}_i - 200|/500$), and the 95% Credibility Interval (CI). The overall performance of the methods is also shown in Figure 4.5. On smaller and noisier radio data the NWE is the best performing method, followed closely by NWE++. On optical data, the best performing method is $D_{2,4}^2$. It is important, however, to note that, in contrast to NWE methods, the DS methods (DS) have parameters that are difficult to set objectively based on the given data only. In the experiments, we found the best DS parameter settings by imposing the true delay $\Delta = 200$, which obviously biases the DS results towards over-optimistic better performance levels.

Controlled experimental setting

For each setting of the Binomial gap distribution $\mu_g = 4, 6, 8$ days and for every noise level ratio from 0.1%, 0.2%, 0.3% we generated 100 base signals from the underlying GP fitted

Table 4.1: RS results for optical range.

Method	$\mu\pm\sigma$	MAE	CI range	95% CI
NWE	199.60±2.97	2.19	0.26	[199.34,199.86]
NWE++	199.83±3.23	2.37	0.28	[199.55,200.11]
DS_1^2	200.67±2.51	1.05	0.22	[200.45,200.89]
$DS_{2,4}^2$	200.02±0.40	0.16	0.04	[199.98,200.06]
DCF	199.14±13.77	11.61	1.21	[197.93,200.35]
LNDCF	200.30±6.34	4.47	0.56	[199.74,200.86]

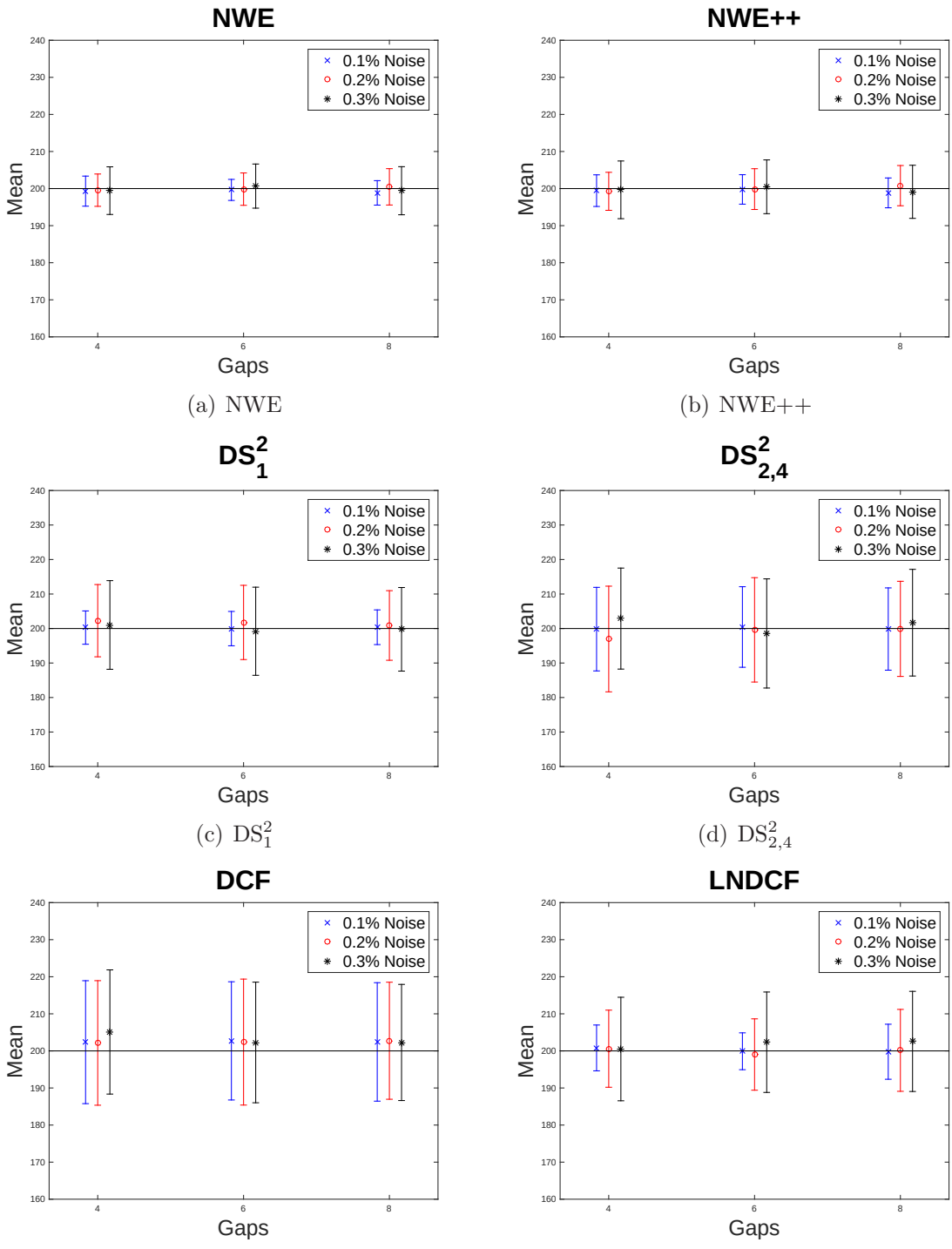
Table 4.2: RS results for radio range.

Method	$\mu\pm\sigma$	MAE	CI range	95% CI
NWE	199.47±3.71	2.95	0.32	[199.15,199.79]
NWE++	200.37±4.31	3.38	0.38	[199.99,200.75]
DS_1^2	201.02±5.42	4.42	0.47	[200.55,201.49]
$DS_{2,4}^2$	204.20±9.98	8.73	0.87	[203.33,205.07]
DCF	201.50±15.23	13.10	1.33	[200.17,202.83]
LNDCF	199.94±5.83	4.73	0.51	[199.43,200.45]

on D^1. We thus obtained 900 data sets. The length of the time series (after applying observational gaps) varied from 800 to 3000 observations.

An analogous procedure was used to generate 900 data sets in the radio range. For each setting of the Binomial gap distribution $\mu_g = 4, 6, 8$ days and for every noise level ratio from 1%, 2%, 3% we generated 100 base signals from the underlying GP fitted on D^5. The overall results across all CS optical and radio data sets are summarized in Tables 4.3 and 4.4, respectively. Figures 4.6 and 4.7 present the results in greater detail, grouped by noise level and gap size.

The kernel-based methods lead to more stable time delay estimates. NWE is the best performing method with respect to all performance measures, followed by NWE++. It is interesting to note that while in general a larger noise level ratio corresponds to a larger standard deviation of the delay estimates, the DCF method seems to be more robust to increased noise levels. For low noise levels and with correlations between time-shifted data streams close to unity, the DCF method is, by construction, relatively insensitive to the level of the noise. However, it is still clearly outperformed by other techniques for the range of noise levels explored in this thesis (see Figures 4.6 and 4.7).

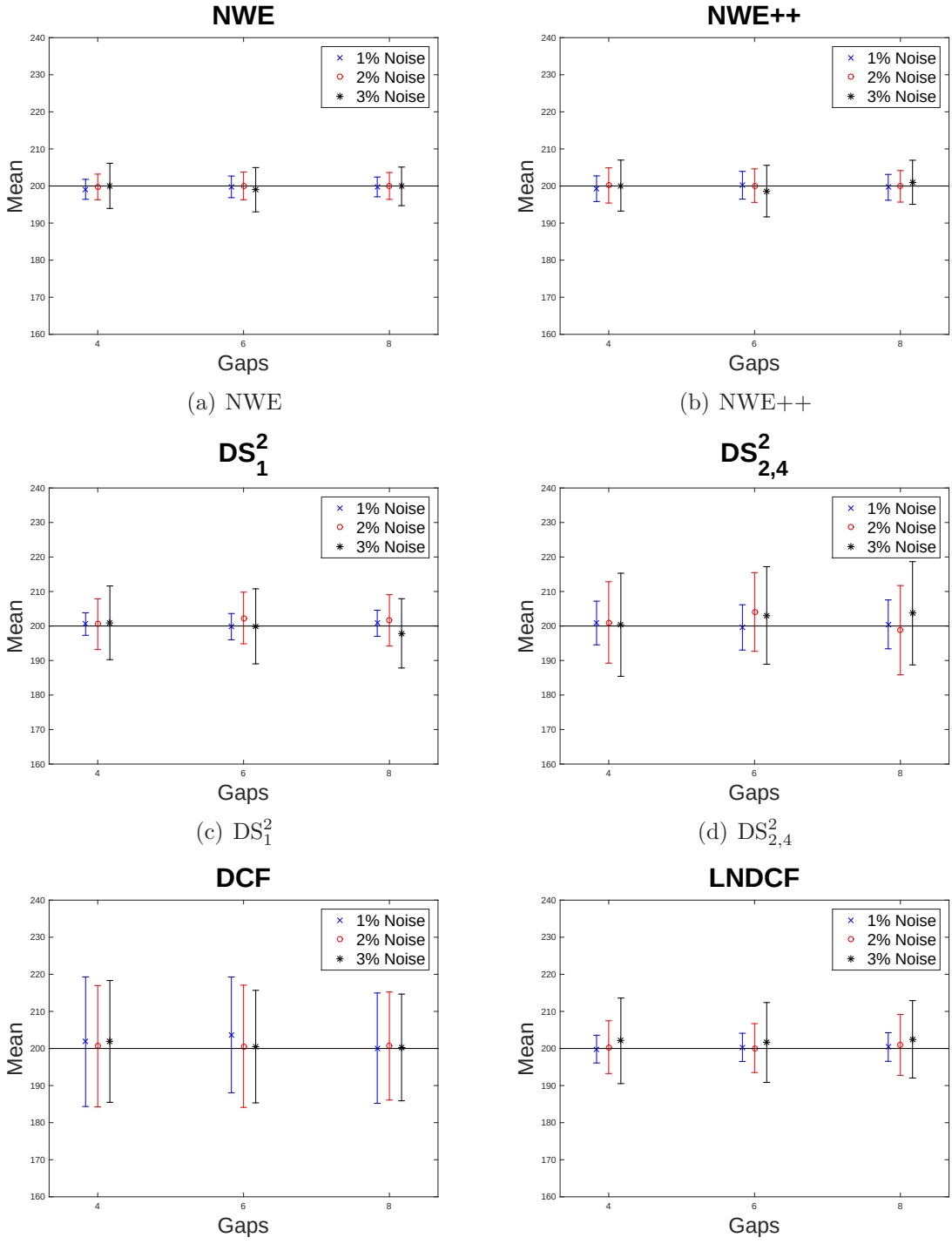


(a) NWE (b) NWE++

(c) DS^2_1 (d) $DS^2_{2,4}$

(e) DCF (f) LNDCF

Figure 4.6: CS optical range results for NWE, NWE++, DS^2_1, $DS2_{2,4}$, DCF and LNDCF methods (plots (a), (b), (c), (d), (e) and (f), respectively) shown as functions of $\mu_g = 4, 6, 8$ days (mean of the binomial gap size distribution) and observational noise level. In each case we present the mean and standard deviation of the delay estimates for the corresponding 100 data sets.



(a) NWE (b) NWE++

(c) DS^2_1 (d) $DS^2_{2,4}$

(e) DCF (f) LNDCF

Figure 4.7: CS radio range results for NWE, NWE++, DS^2_1, $DS2_{2,4}$, DCF and LNDCF methods (plots (a), (b), (c), (d), (e) and (f), respectively) shown as functions of $\mu_g = 4, 6, 8$ days (mean of the binomial gap size distribution) and observational noise level. In each case we present the mean and standard deviation of the delay estimates for the corresponding 100 data sets.

Table 4.3: Overall CS results across all observational gap and noise settings for optical range.

Method	$\mu\pm\sigma$	MAE	CI range	95% CI
NWE	199.69±4.91	3.76	0.32	[199.37,200.01]
NWE++	199.69±5.78	4.41	0.38	[199.31,200.07]
DS_1^2	200.61±9.86	7.62	0.64	[199.97,201.25]
$DS_{2,4}^2$	199.97±14.10	11.98	0.92	[199.05,200.89]
DCF	202.71±16.26	14.22	1.06	[201.65,203.77]
LNDCF	200.63±10.56	8.37	0.69	[199.94,201.32]

Table 4.4: Overall CS results across all observational gap and noise settings for radio range.

Method	$\mu\pm\sigma$	MAE	CI range	95% CI
NWE	199.70±4.23	3.24	0.28	[199.42,199.98]
NWE++	199.89±5.07	3.90	0.33	[199.56,200.22]
DS_1^2	200.49±7.79	5.92	0.51	[199.98,201.00]
$DS_{4,2}^2$	201.31±11.70	9.36	0.76	[200.57,202.09]
DCF	201.13±15.70	13.45	1.03	[200.10,202.16]
LNDCF	200.90±7.92	5.96	0.52	[200.38,201.42]

4.4.2 Experiments on real data

In this section, we present results of methods studied in this thesis using real data - see Table 2.1 and Figure 2.4. Since for real data the noise levels related to observations are available, the NWE++ method was not used.

We have $L = 6$ data sets D^1 — D^6 and for all methods, we test values for time delay on the range of $\Delta = [400, 450]$ (increments of 1 day). The NWE cost to be minimised is $E(\mathbf{h}; \Delta)$ (4.7), with cross-validated kernel scale parameters $\mathbf{h} = (3, 2, 2, 2, 2, 2)$.

For DCF and LNDCF, the bin size $\Delta\tau$ was set to 5, 5, 5, 5, 45, and 30 for D^1, D^2, D^3, D^4, D^5, and D^6, respectively. As mentioned before, unlike in NWE, there is no objective way of setting such parameters based on the data only and we used the setting giving most robust results in the test range of delays 400-450 days. For a fixed delay Δ, the (LN)DCF function values at lag Δ are averaged across the 6 data sets $D^1 - D^6$ and the combined delay estimate is obtained at the maximum of the averaged (LN)DCF curve.

For the Dispersion Spectra method $DS_{2,4}^2$, as argued above, the value of the decorrelation length parameter cannot be resolved in a principled manner based on the data and

Table 4.5: The unique time delay across Q0957+561.

Method	μ (days)
NWE	420
DS_1^2	435
$DS_{2,4}^2$	435
DCF	408.78
LNDCF	426.31

Table 4.6: Q0957+561: Results of 500 Monte Carlo simulations.

Method	μ (d)	σ (d)
NWE	418.65	0.49
DS_1^2	434.98	0.22
$DS_{2,4}^2$	434.92	1.08
DCF	408.77	0.42
LNDCF	431.09	15.04

hence it was set to $\delta = 3$, since at this value DS_1^2 and $DS_{2,4}^2$ have more agreement. Again, for a fixed delay Δ, the $DS_1^2(\Delta)$ and $DS_{2,4}^2(\Delta)$ values are averaged across the six data sets and the combined delay estimate is obtained at the minimum of such averaged curves. The results (unique time delay across Q0957+561) are presented in Table 4.5.

To measure the uncertainty of time delay estimations, following [44, 93, 94, 95], we also performed Monte Carlo simulations by adding white noise generated according to the reported errors to each observation[1]. For each data set, we generated 500 randomized Monte Carlo realizations. The results (mean and standard deviation across the 500 delay estimates) are presented in Table 4.6.

Although we cannot compare these results against a known true value, it is apparent that time delay estimates obtained with different methods are not mutually consistent, unlike estimates on synthetic data. For example, DS_1^2 and DCF estimates appear to lie more than 50 σ apart. Moreover, we find that estimates using different frequency estimates on Q0957+561 data appear to be inconsistent even when the same method is used. This suggests that the claimed measurement errors on the data are significantly

[1]Note that this effectively adds noise to already noisy observations, resulting in a different noise distribution. For example, assuming the original noise is Gaussian, and adding random Gaussian noise from the same distribution, the standard deviation of the noise distribution in this Monte Carlo data will be $\sqrt{2}$ larger than the original one.

under-estimated. Alternatively, there may be unmodeled systematics (e.g., micro-lensing) that lead to varied biases for different analysis techniques.

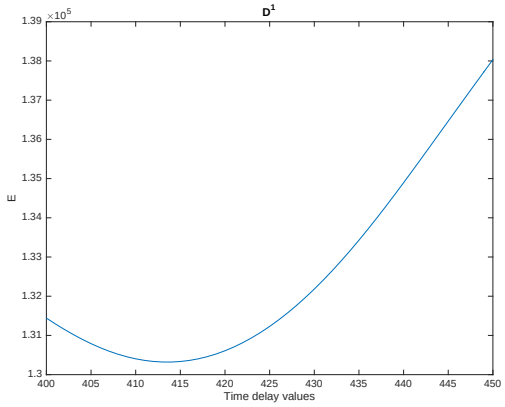
Finally, a summary of results of applying NWE on each data set is shown in Figure 4.8 and Table 4.7. Full flux reconstructions on real data sets are shown in Figure 4.9.

Table 4.7: Q0957+561 Summary of Results using NWE.

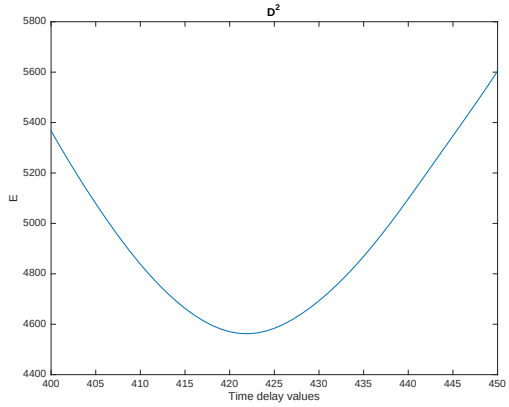
Data	NEW
D^1	414
D^2	422
D^3	428
D^4	422
D^5	450
D^6	418

4.5 Summary

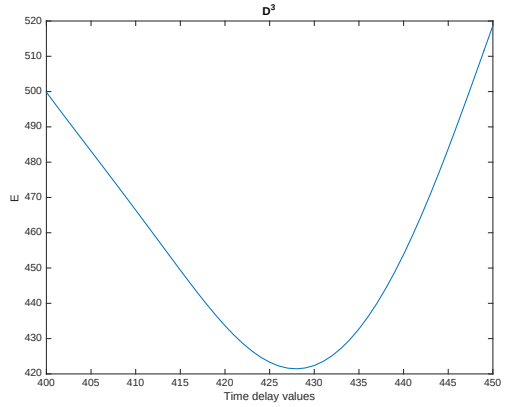
We have introduced a new probabilistic efficient model-based methodology for estimating time delays between two gravitationally lensed images of the same variable point source. The methods were tested and compared on synthetic data sets generated from a GP fitted to the real data. In the controlled experimental setting, the signals were subject to controlled levels of observational noise and gap sizes. In the realistic setting, the data were generated so that multiple aspects of the real data were preserved: noise-to-observed flux ratio, observational gap size distribution and the inter-gap interval distributions. We also performed experiments on real observed optical and radio fluxes from quasar Q0957+561 as a combined data set. Our NWE estimator on the combined optical and radio data suggests a delay of approximately 420 days.



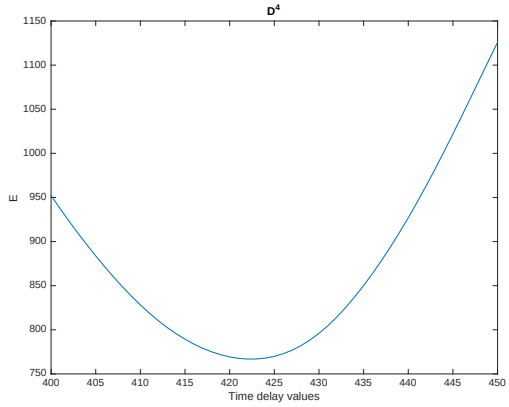
(a) D^1



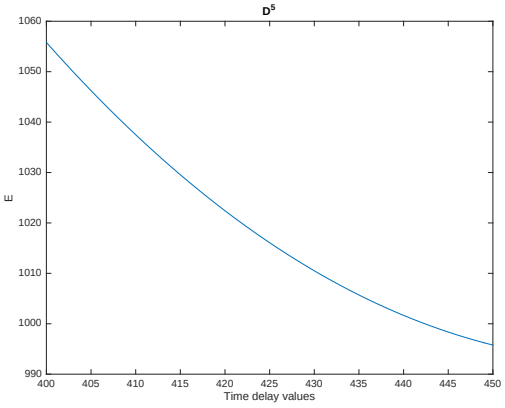
(b) D^2



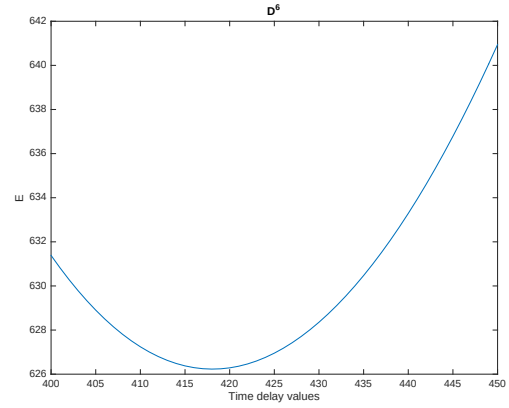
(c) D^3



(d) D^4

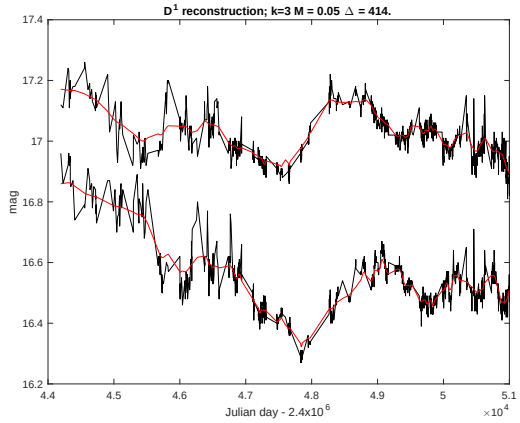


(e) D^5

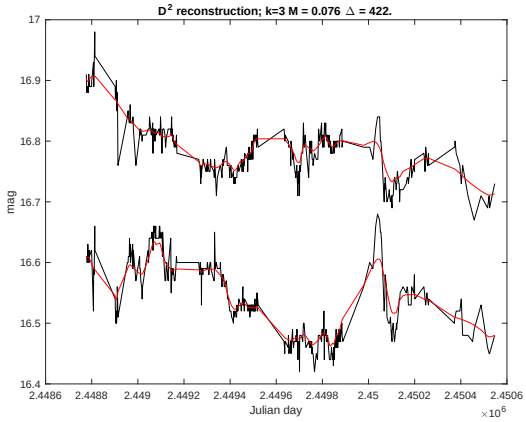


(f) D^6

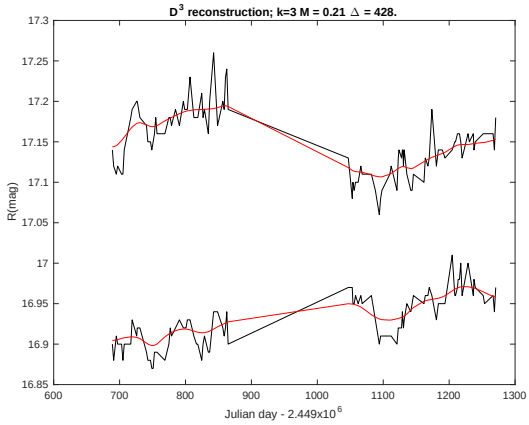
Figure 4.8: Q0957+561 Summary of Results using NWE. Each plot represents E versus Δ for one real data sets.



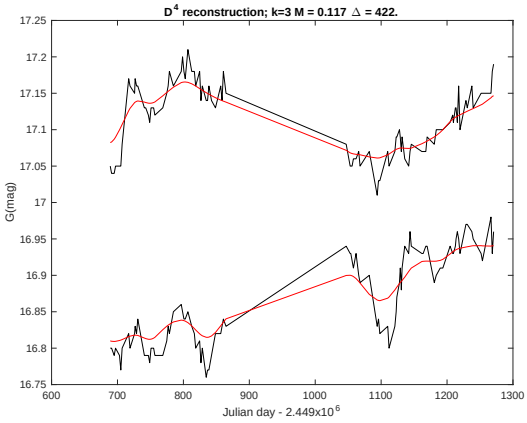
(a) D^1



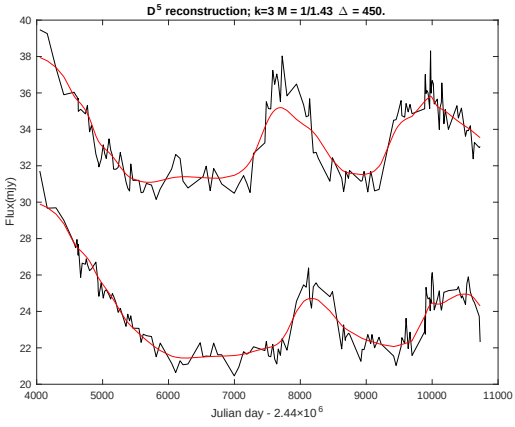
(b) D^2



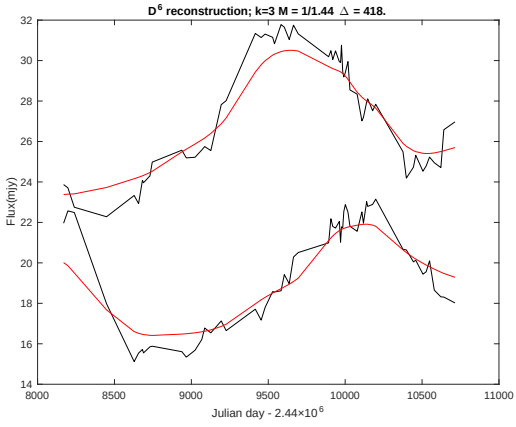
(c) D^3



(d) D^4



(e) D^5



(f) D^6

Figure 4.9: Reconstructions on Real data using NWE.

CHAPTER 5

SMALLER TIME DELAYS - RESOLVING THE TIME DELAY PROBLEM IN STREAMS OF PHOTONS

As seen in the previous chapter, available data are usually in the form of daily measurements which can be used to predict longer (days and months) delays. Current methods in astrophysics are solely rooted in this scenario. However, when countering the problem of shorter delays (e.g. hours), daily measurements are insufficient and one needs to investigate the individual arrival times of photons.

Poisson processes can be applied as a model for photon streams [106]. To resolve the delay in gravitationally lensed photon streams one can use the standard kernel based estimation of the non-homogeneous Poisson process rate function on individual photon streams and then try to time-shift the rate function estimates so the overlap is maximized. Another, more principled alternative is to impose that the source of the delayed photon streams is the same and we simply observe different realizations from the same non-homogeneous Poisson process, gravitationally delayed in time. We study whether, compared with the standard kernel based baseline, such a principled approach can bring benefits in terms of more stable (less variance) estimation.

Normally, delay estimation would be done over streams of photons from a given energy band and then unified over a multitude of energy bands. The baseline and principled delay estimation methods are then compared in a controlled experimental setting using

synthetic photon fluxes with known imposed delay from a variety of non-homogeneous processes assumed to come from a single energy band. To our best knowledge this is the first systematic study that addresses the problem of delay estimation on lensed photon streams. We did not perform experiments on real data, since no large real photon streams from known delayed systems with short time delay are available.

5.1 Kernel Based Delay Estimation in Lensed Photon Streams

For the sake of simplicity we will deal with the case of two lensed photon streams, A and B, from the same source. All techniques presented in this paper can be easily generalized to multiple streams. We assume that the observed photon streams can be accounted for by a Poisson process, the key ingredient of which is the Poisson Distribution - a discrete probability distribution that describes the probability of a number of events occurring in a given period of time:

$$P(X = x) = \frac{e^{-\lambda}\lambda^x}{x!}, \tag{5.1}$$

where $\lambda \geq 0$ is the rate parameter (average number of events in the time period).

Poisson process (e.g. [7, 62, 111, 113]) is a stochastic (point) process that can be used to model arrival times. There are two types of Poisson process: homogeneous Poisson process (HPP), where the rate parameter λ is constant and non-homogeneous Poisson process (NHPP), where $\lambda(s)$ is a function of time s. Given a series of arrival times $s_1, s_2, ..., s_S$, the rate function is commonly estimated by imposing a (Gaussian) kernel of width r on top of each arrival time s_i,

$$K_g(s; s_i, r) = \exp\left\{-\frac{(s - s_i)^2}{2r^2}\right\}. \tag{5.2}$$

The rate function estimate (up to scaling) reads [88, 97, 98]:

$$\hat{\lambda}(s) = \sum_{i=1}^{S} K_g(s; s_i, r) \tag{5.3}$$

We will refer to this method as Kernel Rate Estimation (KRE1).

Suppose that we observe two lensed photon streams $\{s_i^A\}_{i=1}^{S^A}$ and $\{s_i^B\}_{i=1}^{S^B}$ from the same source. On each stream we produce a kernel based estimate of the rate function $\hat{\lambda}^A(s)$, $\hat{\lambda}^B(s)$. Given a suggested time delay Δ, the closeness of the rate estimates (under the delay Δ) can be evaluated e.g. through the mean square difference evaluated on a regular grid of time stamps $\{z_j\}_{j=1}^{Z}$in a relevant interval $[0, T]$,

$$d_2(\hat{\lambda}^A, \hat{\lambda}^B; \Delta) = \frac{1}{Z} \sum_{j=1}^{Z} (\hat{\lambda}^A(z_j) - \hat{\lambda}^B(z_j))^2. \tag{5.4}$$

We will refer to this variant of the method as (KRE1). The delay also can be estimated through minimization of $d_2(\hat{\lambda}^A, \hat{\lambda}^B; \Delta)$ w.r.t the estimated Δ by KRE1 (e.g. via gradient descent).

$$\frac{\partial d_2}{\partial \Delta} = \frac{1}{Z} \sum_{j=1}^{Z} 2(\hat{\lambda}^A(z_j) - \hat{\lambda}^B(z_j)).\frac{\partial(\hat{\lambda}^A(z_j) - \hat{\lambda}^B(z_j))}{\partial \Delta}, \tag{5.5}$$

and we will refer to this variant of the method as (KRE2).

In the following sections we will introduce two types of delay estimation based on Poisson process and its related renewal process.

5.2 Poisson Process Based Estimation (PPE)

Given a suggested delay Δ, the photon steam $\{s_i^B\}_{i=1}^{S^B}$ is shifted in time to the corresponding stream $\{\tilde{s}_i^B\}_{i=1}^{S^B}$, where $\tilde{s}_i^B = s_i^B - \Delta$. The right and left Δ-portions of the streams $\{s_i^A\}_{i=1}^{S^A}$ and $\{\tilde{s}_i^B\}_{i=1}^{S^B}$, respectively, are then cut out to ensure that both streams occur within the same time interval. Assuming the rate function does not change (much)

within interval of length β, we partition the time interval into N_b bins of length β. For each bin i, we denote its mid-time stamp by b_i and the photon counts in steams A and B by C_i^A and C_i^B, respectively. We thus turn the streams of photon arrival times $\{s_i^A\}_{i=1}^{S^A}$ and $\{\tilde{s}_i^B\}_{i=1}^{S^B}$ into the corresponding count streams $\{C_i^A\}_{i=1}^{N_b}$ and $\{C_i^B\}_{i=1}^{N_b}$ associated with bin times $\{b_i\}_{i=1}^{N_b}$.

The crucial aspect of our approach is the imposition of the same (unobserved) Poisson process with rate $\lambda(s)$ capable of accounting for both $\{C_i^A\}_{i=1}^{N_b}$ and $\{C_i^B\}_{i=1}^{N_b}$:

$$P(C^A, C^B|\lambda(s)) = \prod_{i=1}^{N_b} P(C_i^A, C_i^B|\lambda(b_i)). \tag{5.6}$$

Assuming independence (conditional on the rate) of the streams A and B, we have

$$P(C_i^A, C_i^B|\lambda(b_i)) = P(C_i^A|\lambda(b_i)) \cdot P(C_i^B|\lambda(b_i)), \tag{5.7}$$

where

$$P(C|\lambda(b_i)) = e^{-\lambda(b_i)}\frac{\lambda(b_i)^C}{C!}. \tag{5.8}$$

We impose a kernel based model on the common rate function:

$$\lambda(s) = \Psi\left(\sum_{j=1}^{J} w_j K_g(s; c_j, r_b)\right) = \Psi\left(\boldsymbol{w}^\intercal K_g(s; \boldsymbol{c}, r_b)\right), \tag{5.9}$$

with kernels of width r_b, centered at $c_j, j = 1, 2 \ldots J$ and the J free parameters w_j collected in vector $\boldsymbol{w}$. $K_g(s; \boldsymbol{c}, r_b)$ as a vector of kernel evaluations $K_g(s; c_j, r_b)$ at all centers of $\boldsymbol{c} = (c_1, c_2, ..., c_J)$. The function $\Psi(x) = e^x$ is introduced to constrain the model to positive rates. In the experiments we set the kernel centers c_j to the bin mid-points b_j and the kernel width r_b to a multiple of bin width β.

Using (5.8), we obtain

$$P(C_i^A|\lambda(b_i))P(C_i^B|\lambda(b_i)) = e^{-\lambda(b_i)}\frac{\lambda(b_i)^{C_i^A}}{C_i^A!}e^{-\lambda(b_i)}\frac{\lambda(b_i)^{C_i^B}}{C_i^B!} \tag{5.10}$$

leading to negative log likelihood playing the role of error functional:

$$E(\boldsymbol{w}) = -\sum_{i=1}^{N_b} \ln \left[\left(e^{-\lambda(b_i)} \frac{\lambda(b_i)^{C_i^A}}{C_i^A!} \right) \left(e^{-\lambda(b_i)} \frac{\lambda(b_i)^{C_i^B}}{C_i^B!} \right) \right], \tag{5.11}$$

which is equivalent to:

$$E(\boldsymbol{w}) = -\sum_{i=1}^{N_b} -\lambda(b_i) \ln e + C_i^A \ln \lambda(b_i) - \ln C_i^A! - \lambda(b_i) \ln e + C_i^B \ln \lambda(b_i) - \ln C_i^B! \tag{5.12}$$

The negative log-likelihood (without constant terms) simplifies to

$$E(\boldsymbol{w}) = -\sum_{i=1}^{N_b} -2\lambda(b_i) + \ln \lambda(b_i) \left[C_i^A + C_i^B \right] \tag{5.13}$$

where the vector $\boldsymbol{w}$ collects the model parameters w_j. Denoting $\varphi_i = \left[C_i^A + C_i^B \right]$, we have

$$E(\boldsymbol{w}) = 2\sum_{i=1}^{N_b} \Psi(\boldsymbol{w}^\intercal K_g(b_i; \boldsymbol{c}, r_b)) - \sum_{i=1}^{N_b} \varphi_i \ln(\Psi(\boldsymbol{w}^\intercal K_g(b_i; \boldsymbol{c}, r_b))) \tag{5.14}$$

In order to minimize $E(\boldsymbol{w})$, we calculate

$$\frac{\partial \Psi(\boldsymbol{w}^\intercal K_g(b_i; \boldsymbol{c}, r_b))}{\partial w_j} = \Psi^{'}(\boldsymbol{w}^\intercal K_g(b_i; \boldsymbol{c}, r_b)) \phi_j(b_i) \tag{5.15}$$

and

$$\frac{\partial \ln \Psi(\boldsymbol{w}^\intercal K_g(b_i; \boldsymbol{c}, r_b))}{\partial w_j} = \frac{\Psi^{'}(\boldsymbol{w}^\intercal K_g(b_i; \boldsymbol{c}, r_b))}{\Psi(\boldsymbol{w}^\intercal K_g(b_i; \boldsymbol{c}, r_b))} \phi_j(b_i), \tag{5.16}$$

where

$$\phi_j(b_i) = \frac{\partial (\boldsymbol{w}^\intercal K_g(b_i; \boldsymbol{c}, r_b))}{\partial w_j} \tag{5.17}$$

leading to the optimality criterion

$$2\sum_{i=1}^{N_b} \Psi^{'}(\boldsymbol{w}^\intercal K_g(b_i; \boldsymbol{c}, r_b)) \phi_j(b_i) - \sum_{i=1}^{N_b} \varphi_i \frac{\Psi^{'}(\boldsymbol{w}^\intercal K_g(b_i; \boldsymbol{c}, r_b))}{\Psi(\boldsymbol{w}^\intercal K_g(b_i; \boldsymbol{c}, r_b))} \phi_j(b_i) = 0. \tag{5.18}$$

Hence, we arrive at a very intuitive solution - $E(\boldsymbol{w})$ is minimized for the model yielding

the average bin counts:

$$\lambda(b_i) = \Psi(\boldsymbol{w}^\intercal K_g(b_i; \boldsymbol{c}, r_b) = \frac{\varphi_i}{2} = \frac{C_i^A + C_i^B}{2}. \tag{5.19}$$

Denoting the average count $(C_i^A + C_i^B)/2$ in bin i by C_i, we obtain

$$\boldsymbol{w}^\intercal K_g(b_i; \boldsymbol{c}, r_b) = \Psi^{-1}(C_i) \tag{5.20}$$

Defining a vector $\boldsymbol{C}$ as

$$\boldsymbol{C} = \left[\Psi^{-1}(C_1), \Psi^{-1}(C_2), \cdots, \Psi^{-1}(C_{N_b})\right]^\intercal \tag{5.21}$$

we have

$$\boldsymbol{w}^\intercal \mathbf{K}_g = \boldsymbol{C}^\intercal, \tag{5.22}$$

where $\mathbf{K}_g$ is an $N_b \times N_b$ matrix

$$\mathbf{K}_g = \left[K_g(b_1; \boldsymbol{c}, r_b), K_g(b_2; \boldsymbol{c}, r_b), \ldots K_g(b_{N_b}; \boldsymbol{c}, r_b)\right], \tag{5.23}$$

we obtain the model estimate

$$\boldsymbol{w} = \mathbf{K}^+ \boldsymbol{C}, \tag{5.24}$$

where $\mathbf{K}^+$ is the Moore-Penrose pseudo-inverse[1] of $\mathbf{K} = \mathbf{K}_g^\intercal$.

5.3 Innovation Process Based Estimation (IPE)

The previous approach was based on modeling count data within individual time bins. Bin width is a free parameter that needs to be set and the delay estimation can be sensitive to this value. To avoid this problem, we introduce a different approach based on modeling inter-arrival times. It is well known that if event counts can be modeled by

[1] In case of ill-conditioned $\mathbf{K}$ one can use e.g. SVD decomposition to regularize the matrix inversion.

Poisson distribution with mean rate λ, then the inter-arrival times are distributed with exponential distribution with mean λ^{-1}.

As in the previous approach,given a suggested delay Δ, the photon stream $\{s_i^B\}_{i=1}^{S^B}$ is shifted in time to the corresponding stream $\{\tilde{s}_i^B\}_{i=1}^{S^B}$, where $\tilde{s}_i^B = s_i^B - \Delta$. The right and left Δ-portions of the streams $\{s_i^A\}_{i=1}^{S^A}$ and $\{\tilde{s}_i^B\}_{i=1}^{S^B}$, respectively, are then cut out to ensure that both streams occur within the same time interval. We denote the differences between two consecutive arrival times by $d^A = \{d_i^A\}_{i=1}^{D^A}$ and $d^B = \{d_i^B\}_{i=1}^{D^B}$, where $d_i^A = s_{i+1}^A - s_i^A$ and $d_i^B = s_{i+1}^B - s_i^B$, respectively.

5.3.1 IPE1

We aim to find a probabilistic model that maximizes the probability $P(d^A, d^B|\lambda(s))$. Assuming that streams A and B are independent, we have

$$P(d^A, d^B|\lambda(s)) = \prod_{i=1}^{D^A} P(d_i^A|\lambda(s_i^A)) \prod_{i=1}^{D^B} P(d_i^B|\lambda(s_i^B)), \tag{5.25}$$

where

$$P(d|\lambda) = \lambda e^{-\lambda d}. \tag{5.26}$$

As in the previous model, we impose a kernel based model on the common rate function[1]

$$\lambda(s) = \sum_{j=1}^{J} w_j K_g(s; c_j, r_o) = \boldsymbol{w}^\intercal K_g(s; \boldsymbol{c}, r_o), \tag{5.27}$$

with kernels of width r_o, centered at $c_j, j = 1, 2 \ldots J$ and the J free parameters w_j collected in vector $\boldsymbol{w}$. $K_g(s; \boldsymbol{c}, r_o)$ is a vector of kernel evaluations $K_g(s; c_j, r_o)$ at all centers of $\boldsymbol{c} = (c_1, c_2, ..., c_J)$.

[1]In the experiments, we almost never encountered the solution with negative values of λ. Therefore, to simplify presentation, we do not apply the transformation function Ψ.

Using (5.26), we obtain

$$P(d^A, d^B|\lambda(s)) = \prod_{i=1}^{D^A} \lambda(s_i^A) e^{-\lambda(s_i^A) d_i^A} \prod_{i=1}^{D^B} \lambda(s_i^B) e^{(-\lambda(s_i^B) d_i^B)}. \tag{5.28}$$

The error functional (taking the negative log-likelihood of (5.28)) is

$$E = -\Sigma_{i=1}^{D^A}(\log \lambda(s_i^A) - \lambda(s_i^A) d_i^A) - \Sigma_{i=1}^{D^B}(\log \lambda(s_i^B) - \lambda(s_i^B) d_i^B). \tag{5.29}$$

We minimize $E(\boldsymbol{w})$ via gradient descent,

$$\boldsymbol{w}(m+1) = \boldsymbol{w}(m) - \gamma_1 \frac{\partial E}{\partial \boldsymbol{w}}, \tag{5.30}$$

where $\gamma_1 > 0$ is the learning rate controlling the step size and

$$\begin{aligned} \frac{\partial E}{\partial \boldsymbol{w}} = & - \sum_{i=1}^{D^A} \frac{K_g(s_i^A; \boldsymbol{c}, r_o)}{\boldsymbol{w}^\intercal K_g(s_i^A; \boldsymbol{c}, r_o)} - d_i^A K_g(s_i^A; \boldsymbol{c}, r_o) \\ & - \sum_{i=1}^{D^B} \frac{K_g(s_i^B; \boldsymbol{c}, r_o)}{\boldsymbol{w}^\intercal K_g(s_i^B; \boldsymbol{c}, r_o)} - d_i^B K_g(s_i^B; \boldsymbol{c}, r_o). \end{aligned} \tag{5.31}$$

5.3.2 IPE2

In this approach, we performed gradient descent not only on the model parameters $\boldsymbol{w}$, but also on the delay Δ. In this case we do not need to cut the photon streams. For presentation simplicity, we will still use the notation $d^A = \{d_i^A\}_{i=1}^{D^A}$ and $d^B = \{d_i^B\}_{i=1}^{D^B}$ for the inter-arrival times.

Again, our goal is to find a a probabilistic model that maximizes the probability $P(d^A, d^B|\lambda(s))$

$$P(d^A, d^B|\lambda^A(s), \lambda^B(s)) = \prod_{i=1}^{D^A} P(d_i^A|\lambda^A(s_i; \boldsymbol{w})) \prod_{i=1}^{D^B} P(d_i^B|\lambda^B(s_i; \boldsymbol{w}, \Delta)) \tag{5.32}$$

We impose a kernel based model on the common rate function (expressed for stream

A):

$$\lambda^A(s) = \sum_{j=1}^{J} w_j K_g(s; c_j, r_o) = \boldsymbol{w}^\intercal K_g(s; \boldsymbol{c}, r_o). \tag{5.33}$$

We suppose that the rate function of stream B is a time-delayed (by Δ) version of the one for stream A:

$$\lambda^B(s) = \left(\sum_{j=1}^{J} w_j K_g(s; c_j - \Delta, r_o) \right) = \boldsymbol{w}^\intercal K_g(s; \boldsymbol{c} - \Delta, r_o), \tag{5.34}$$

using (5.26), we obtain

$$P(d^A, d^B | \lambda^A(s), \lambda^B(s)) = \prod_{i=1}^{D^A} \lambda^A(s_i) e^{-\lambda^A(s_i) d_i^A} \prod_{i=1}^{D^B} \lambda^B(s_i) e^{-\lambda^B(s_i) d_i^B}, \tag{5.35}$$

leading to the error functional

$$E = -\sum_{i=1}^{D^A} (\log \lambda^A(s_i) - \lambda^A(s_i) d_i^A) - \sum_{i=1}^{D^B} (\log \lambda^B(s_i) - \lambda^B(s_i) d_i^B). \tag{5.36}$$

We will minimize E w.r.t two parameters $(\boldsymbol{w}, \Delta)$. To that end we plug (5.33) and (5.34) into (5.36):

$$\begin{aligned} E = &-\sum_{i=1}^{D^A} \left(\log \sum_{j=1}^{J} w_j K_g(s_i^A; c_j, r_o) - d_i^A \sum_{j=1}^{J} w_j K_g(s_i^A; c_j, r_o) \right) \\ &-\sum_{i=1}^{D^B} \left(\log \sum_{j=1}^{J} w_j K_g(s_i^B; c_j - \Delta, r_o) - d_i^B \sum_{j=1}^{J} w_j K_g(s_i^B; c_j - \Delta, r_o) \right). \end{aligned} \tag{5.37}$$

We have,

$$\begin{aligned} \frac{\partial E}{\partial \boldsymbol{w}} = &- \sum_{i=1}^{D^A} \left(\frac{K_g(s_i^A; \boldsymbol{c}, r_o)}{\boldsymbol{w}^\intercal K_g(s_i^A; \boldsymbol{c}, r_o)} - d_i^A K_g(s_i^A; \boldsymbol{c}, r_o) \right) \\ &- \sum_{i=1}^{D^B} \left(\frac{K_g(s_i^B; \boldsymbol{c} - \Delta \cdot \mathbf{1}, r_o)}{\boldsymbol{w}^\intercal K_g(s_i^B; \boldsymbol{c} - \Delta \cdot \mathbf{1}, r_o)} - d_i^B K_g(s_i^B; \boldsymbol{c} - \Delta \cdot \mathbf{1}, r_o) \right), \end{aligned} \tag{5.38}$$

where **1** is a vector of 1's and

$$\frac{\partial E}{\partial \Delta} = -\sum_{i=1}^{D^B} \left(\frac{1}{\sum_{j=1}^{J} w_j \exp\left\{\frac{-(s_i^B-(c_j-\Delta))^2}{2r_o^2}\right\}} \sum_{j=1}^{J} w_j \exp\left\{\frac{-(s_i^B-(c_j-\Delta))^2}{2r_o^2}\right\} \frac{-2(s_i^B-(c_j-\Delta))}{2r_o^2} \right.$$
$$\left. - d_i^B \sum_{j=1}^{J} w_j \exp\left\{\frac{-(s_i^B-(c_j-\Delta))^2}{2r_o^2}\right\} \frac{-2(s_i^B-(c_j-\Delta))}{2r_o^2} \right) \tag{5.39}$$

Finally Δ is updated as follows:

$$\Delta(m+1) = \Delta(m) - \gamma_2 \frac{\partial E}{\partial \Delta} \tag{5.40}$$

where $\gamma_2 > 0$ is the learning rate and $\boldsymbol{w}$ is updated using (5.30) and (5.31).

5.3.3 IPE3

Finally, in the last variation of our method (IPE3) we optimize E w.r.t the model parameters, delay and kernel band-width r_o.

$$\frac{\partial E}{\partial r_o} = -\sum_{i=1}^{D^A} \frac{1}{\sum_{j=1}^{J} w_j \exp\left\{\frac{-(s_i^A-c_j)^2}{2r^2}\right\}} \sum_{j=1}^{J} w_j \exp\left\{\frac{-(s_i^A-c_j)^2}{2r^2}\right\} \frac{(s_i^A-c_j)^2}{r^3}$$
$$- d_i^A \sum_{j=1}^{J} w_j \exp\left\{\frac{-(s_i^A-c_j)^2}{2r^2}\right\} \frac{(s_i^A-c_j)^2}{r^3}$$
$$- \sum_{i=1}^{D^B} \frac{1}{\sum_{j=1}^{J} w_j \exp\left\{\frac{-(s_i^B-(c_j-\Delta))^2}{2r^2}\right\}} \sum_{j=1}^{J} w_j \exp\left\{\frac{-(s_i^B-(c_j-\Delta))^2}{2r^2}\right\} \frac{(s_i^B-(c_j-\Delta))^2}{r^3}$$
$$- d_i^B \sum_{j=1}^{J} (w_j \exp\left\{\frac{-(s_i^B-(c_j-\Delta))^2}{2r^2}\right\} \frac{(s_i^B-(c_j-\Delta))^2}{r^3}. \tag{5.41}$$

updated as follows

$$r(m+1) = r(m) - \gamma_3 \frac{\partial E}{\partial r} \tag{5.42}$$

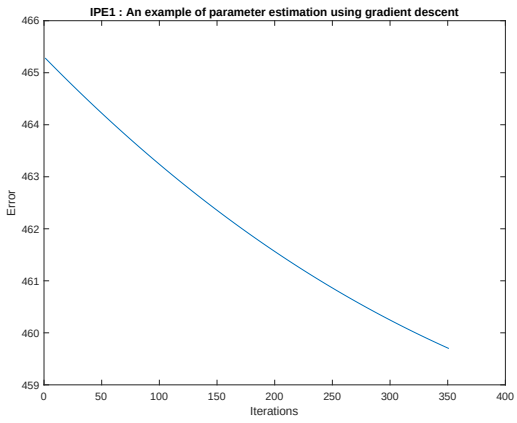
where $\gamma_3 > 0$ is the learning rate. $\boldsymbol{w}$ is updated according to (5.30) and(5.31) where Δ is updated according to (5.39) and (5.40).

5.3.4 Gradient descent parameters

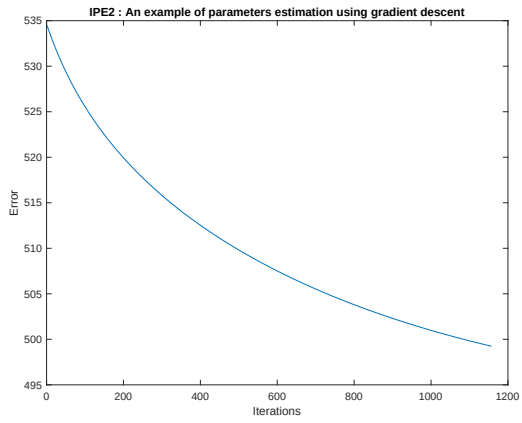
The values of our learning rates γ_1 (5.30) , γ_2 (5.40) and γ_3 (5.42) are chosen based on previous preliminary experiments where we tested a range of values for each parameter. In these experiments, we started by optimizing our models with large learning rates (e.g. 0.1), and then progressively reducing these rates, by an order of magnitude (0.01, then 0.001, 0.0001, etc.). We selected the values that seem to be causing E -(5.29) and (5.36)- to decrease rapidly. For the final experiments (Section 5.6), we set γ_1 , γ_2 and γ_3 to 10^{-6}, 10^{-4} and 10^{-5} respectively. For convergence condition, we used an automatic test that declares convergence if E decreases by less than a small number ϵ in one iteration. In other words, if E goes below a small number ϵ, we stop and declare convergence. The value of ϵ is chosen based on previous preliminary experiments where we tested a range of values for ϵ. For the final experiments (Section 5.6), we set ϵ to 0.1. Examples of parameters estimation using Gradient descent algorithms for IPE1, IPE2 and IPE3 are shown in Figure 5.1.

5.4 Parameters Initialization

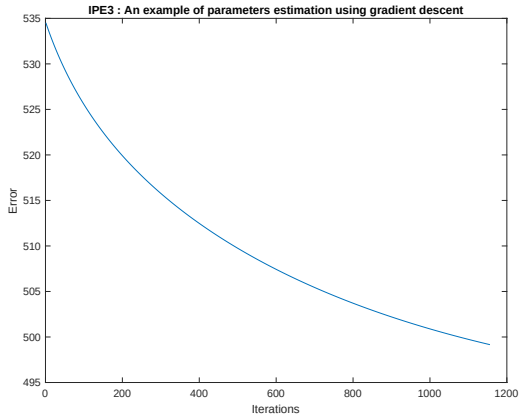
In this section we describe in detail how the free parameters of the proposed methods are initialized.



(a)



(b)



(c)

Figure 5.1: Examples of parameters estimation using Gradient descent algorithm: (a) IPE1, (b) IPE2 and (c) IPE3.

5.4.1 Kernel parameters

Gaussian kernels have two parameters that need to be determined, in particular kernel centers $\{c_j\}_{j=1}^{J}$ and the kernel width r. We use three approaches to position the kernels:

- for KRE, kernels are centered at each photon's arrival time.
- for PPE, kernels are centered at each bin center.
- for IPE, the centers c_j are uniformly distributed across the time period $[0, T]$.

The kernel width determines the degree of smoothing for the underlying rate function. For KRE, we apply a method for selecting the width based on the principle of minimizing

the mean integrated square error (MISE) proposed by [120]. In this method, given a time delay Δ, the photon streams A and B are first superimposed into a single stream $\{s_i\}$. The optimal band-width value r is then found by minimizing

$$C(r) = \frac{1}{2^2} \sum_{i,j} \mathcal{F}(s_i, s_j) - \frac{2}{2^2} \sum_{i \neq j} \mathcal{K}_r(s_i - s_j)$$

where

$$\mathcal{F}(s_i, s_j) = \int_a^b \mathcal{K}_r(s - s_i)\mathcal{K}_r(s - s_j)\,\mathrm{d}s$$

and

$$\mathcal{K}_r(s) = \frac{1}{\sqrt{2\pi}r} \exp\left\{-\frac{s^2}{2r^2}\right\}.$$

To find the kernel width (and bin size) r_b in PPE, the 'optimal' bin width selection method proposed by [119] summarized in Algorithm 3.

Algorithm 3 A method for the bin width β selection ([119]).

1: **for all** $\beta_i \in (L_\beta, U_\beta)$ **do**

2: Divide the observation period T into N^β bins of width β

3: Count the number of arrivals k_i from stream A and B that enter the ith bin.

4: Calculate the mean and of the arrivals count k_i as follows

$$\bar{k} = \frac{1}{N^\beta} \sum_{i=1}^{N^\beta} k_i$$

5: Calculate the variance of the arrivals count k_i as follows

$$var = \frac{1}{N^\beta - 1} \sum_{i=1}^{N^\beta} (k_i - \bar{k})^2$$

6: Calculate the cost function

$$C(\beta) = \frac{2\bar{k} - var}{(2\beta)^2}$$

7: **end for**

8: $\beta \leftarrow \text{argmin}_{\beta_i}(C)$

The algorithm needs a search range $[L_\beta, U_\beta]$ for bin width β. We determine this interval by finding the minimum of (5.14) for a series of trial values of $r_b = \beta$. The search interval $[L_\beta, U_\beta]$ then corresponds to the largest stable delay estimation region of β values - i.e., the interval of β values for which the estimated delay Δ does not change (see Figure 5.2).

For IPE models, kernel width r_o is optimized using cross validation algorithm (CVA) [24, 48]. The algorithm partitions the data into 10 blocks of equal length $\mathcal{L}$. The i-th validation set $\boldsymbol{\mathcal{V}}_i$, $i = 1, 2 \ldots \mathcal{L}$, is obtained by collecting the i-th element of each block. The rest of the data is the "training set". We then fit our models on the training set and use the validation set $\boldsymbol{\mathcal{V}}_i$ to calculate the cost function E over a range of suggested width values $r_o \in (L_{r_o}, U_{r_o})$. This procedure is repeated $\mathcal{L}$ times for each validation set $\boldsymbol{\mathcal{V}}_i$, $i = 1, 2 \ldots \mathcal{L}$. The chosen r_o is the one yielding the smallest average cost E across the

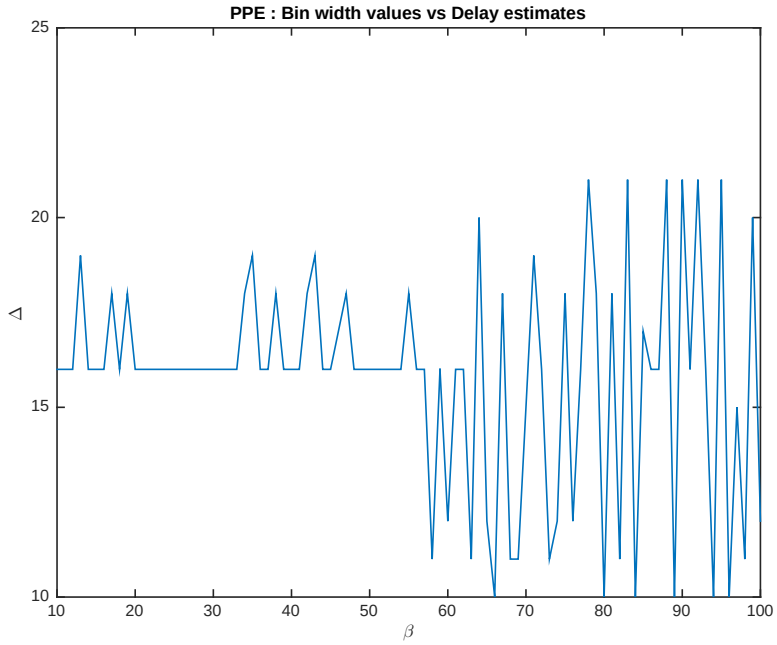


Figure 5.2: β versus Δ for PPE. In each combination (β, Δ) the delay estimate has been plotted; $\Delta = [10, 40]$ with 1 unit increment; The range is at $\beta \in [21, 32]$.

folds $i = 1, 2 \ldots \mathcal{L}$.

5.4.2 IPE parameter initialization using KRE1

The IPE weight vector $\boldsymbol{w}$ is initialized using the rate function estimates readily provided by the KRE1 model. However, the rate functions obtained by KRE1 on streams A and B need to be scaled to represent the underlying rate of the non-homogeneous Poisson process. Note that for the delay detection task for which the KRE1 method is used, no such scaling was needed - the delay is invariant to scaling the estimated rate functions by the same factor. In contrast, the IPE methods need to operate with the non-scaled estimates of the true rate function.

Given the KRE1-estimated rate functions on streams A and B, $\hat{\lambda}^A(s)$, $\hat{\lambda}^B(s)$, respectively, the overall KRE1 rate function is their average

$$\hat{\lambda}(s) = \frac{\hat{\lambda}^A(s) + \hat{\lambda}^B(s)}{2}. \tag{5.43}$$

The scaling factor ϑ is obtained as follows: By imposing the rate function

$$\lambda(s) = \vartheta\hat{\lambda}(s) \tag{5.44}$$

the ϑ value is found using maximum likelihood (minimizing negative log-likelihood) see (5.29) in Section (5.3.1) on page 69. By replacing $\lambda(s)$ with $\vartheta\hat{\lambda}(s)$ in (5.29), we have,

$$E = -\sum_{i=1}^{D^A}(\log(\vartheta\hat{\lambda}(s_i^A)) - \vartheta\hat{\lambda}(s_i^A)d_i^A) - \sum_{i=1}^{D^B}(\log(\vartheta\hat{\lambda}(s_i^B)) - \vartheta\hat{\lambda}(s_i^B)d_i^B) \tag{5.45}$$

with

$$\frac{\partial E}{\partial \vartheta} = -\sum_{i=1}^{D^A}(\frac{\hat{\lambda}(s_i^A)}{\vartheta\hat{\lambda}(s_i^A)} - \hat{\lambda}(s_i^A)d_i^A) - \sum_{i=1}^{D^B}(\frac{\hat{\lambda}(s_i^B)}{\vartheta\hat{\lambda}(s_i^B)} - \hat{\lambda}(s_i^B)d_i^B). \tag{5.46}$$

Denoting $\hat{\lambda}(s_i^A)d_i^A$ and $\hat{\lambda}(s_i^B)d_i^B$ by q_i^A and q_i^B, respectively, we obtain

$$\begin{aligned}\frac{\partial E}{\partial \vartheta} &= \frac{1}{\vartheta}\Big(-\sum_{i=1}^{D^A}(1-\vartheta q_i^A) - \sum_{i=1}^{D^B}(1-\vartheta q_i^B)\Big) \\ &= \frac{1}{\vartheta}\Big(-D^A + \vartheta\sum_{i=1}^{D^A} q_i^A - D^B + \vartheta\sum_{i=1}^{D^B} q_i^B\Big).\end{aligned} \tag{5.47}$$

Setting the derivative to zero, we get

$$\vartheta = \frac{D^A + D^B}{\sum_{i=1}^{D^A} q_i^A + \sum_{i=1}^{D^B} q_i^B}. \tag{5.48}$$

Setting of IPE weights to match the rate function $\lambda(s)$ can then be done by imposing a regular $(s_1, s_2, ..., s_N)$ grid on $[0, T]$, evaluating the rate values on the grid,

$$\boldsymbol{x} = (\hat{\lambda}(s_1), \hat{\lambda}(s_2) \ldots \hat{\lambda}(s_N))^\intercal, \tag{5.49}$$

and solving

$$\boldsymbol{w} = \boldsymbol{K}^{\intercal+}\boldsymbol{x}, \tag{5.50}$$

where $\mathbf{K}$ is an $N \times N$ matrix

$$\mathbf{K} = [K_g(s_1; \boldsymbol{c}, r_o), K_g(s_2; \boldsymbol{c}, r_o), \dots K_g(s_N; \boldsymbol{c}, r_o)]\,. \tag{5.51}$$

and $\mathbf{K}^{\intercal +}$ is the Moore-Penrose pseudo-inverse[1] of $\mathbf{K}^{\intercal}$.

5.5 Data

To test and compare different methodologies suggested above, we performed controlled experiments on synthetic data generated from non-homogeneous Poisson processes. From each given non-homogeneous Poisson process we generated two series A and B of arrival times, the series B was then time-shifted by a known delay.

The rate functions defining non-homogeneous Poisson processes were obtained by superimposing G Gaussian functions of fixed width r_g positioned on a regular grid $\{c_g\}_{g=1}^{G}$ in $[0, T]$,

$$\lambda(s) = \sum_{g=1}^{G} w_g \cdot \exp\left\{\frac{-(s - c_g)^2}{2r_g^2}\right\}, \tag{5.52}$$

where $w_g \in \mathbb{R}$ are the mixing weights generated randomly from uniform distribution on $[L_w, U_w]$. The kernel widths were set to a multiple of the kernel separation (distance between the two consecutive kernel centers) d_g, $r_g = \alpha_g \cdot d_g$. We used $T = 400$, $G = 80$, $\alpha_g = 3$, $L_w = -1$ and $U_w = 1$. The synthetic rate functions were then rescaled to the interval $[0, 2]$. Figure 5.3 shows examples of rate functions created using the method outlined above.

[1] In case of ill-conditioned $\mathbf{K}$ one can use e.g. SVD decomposition to regularize the matrix inversion.

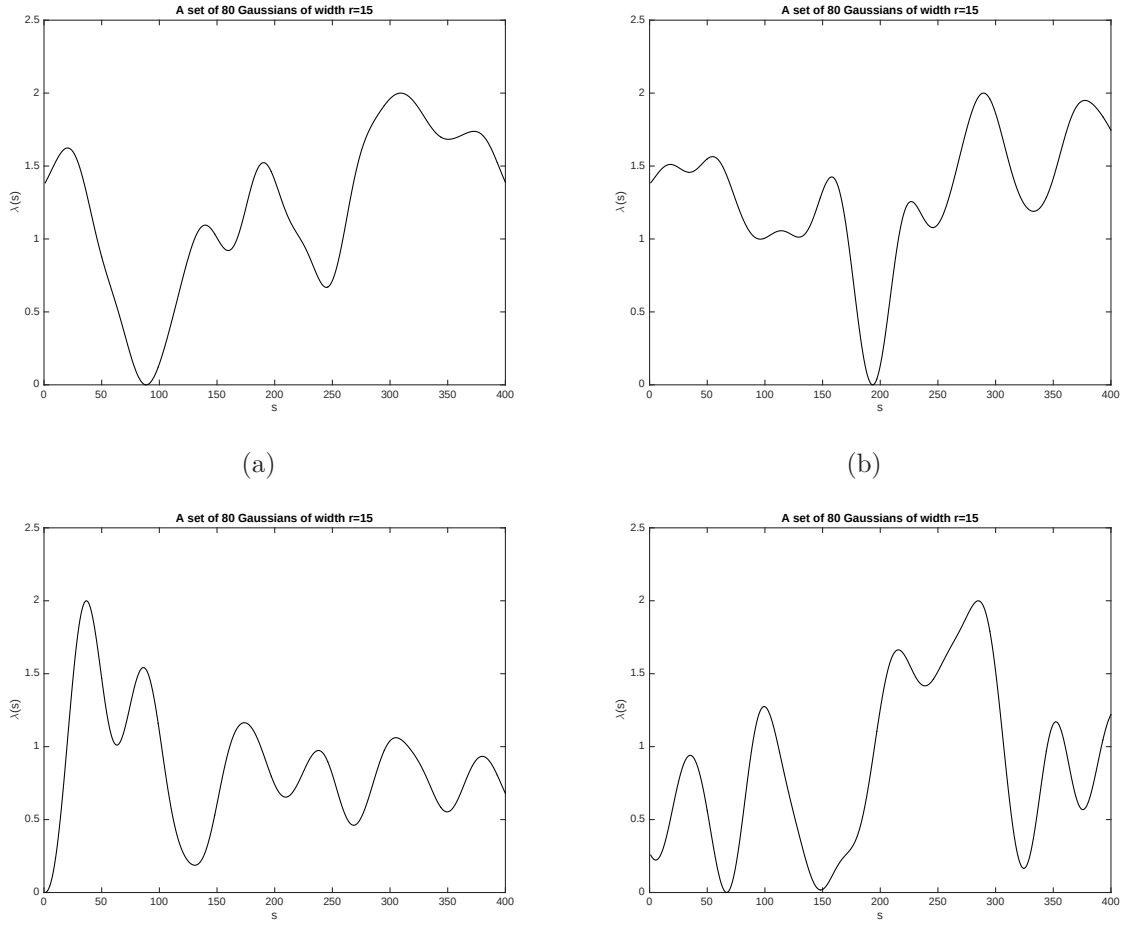


Figure 5.3: Examples of randomly generated rate functions.

Given a rate function $\lambda(s)$, the arrival times were generated using the Thinning technique [10, 34, 68, 112, 121] summarized in Algorithm 4. An example of the resulting stream is shown in Figure 5.4.

Algorithm 4 Thinning technique algorithm. Source [34].

1: Start with $s = 0$ and repeat until the end of period T is reached.

2: Set $\omega = \sup_{t \geq s} \lambda(s)$.

3: Generate a realization d from exponential distribution with mean ω^{-1}.

4: Generate a realization u from uniform distribution over $(0, 1)$.

5: If $u \leq \lambda(s + d)/\omega$, the next arrival time is $s + d$; otherwise $s \leftarrow s + d$ and go to (2).

Using this process, we generate two photon streams from the same rate function: $\{s_i^A\}$

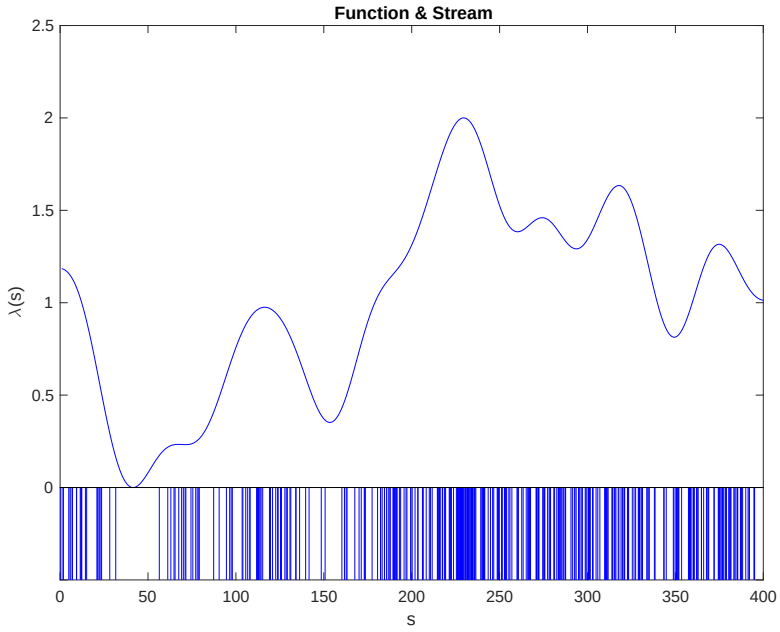


Figure 5.4: An example of a test rate function and the corresponding photon stream.

and $\{s_j^B\}$, $i = 1, 2, \cdots, S^A$ and $j = 1, 2, \cdots, S^B$. To create a pair of time shifted streams, s^B is shifted in time by a delay $\Delta > 0$

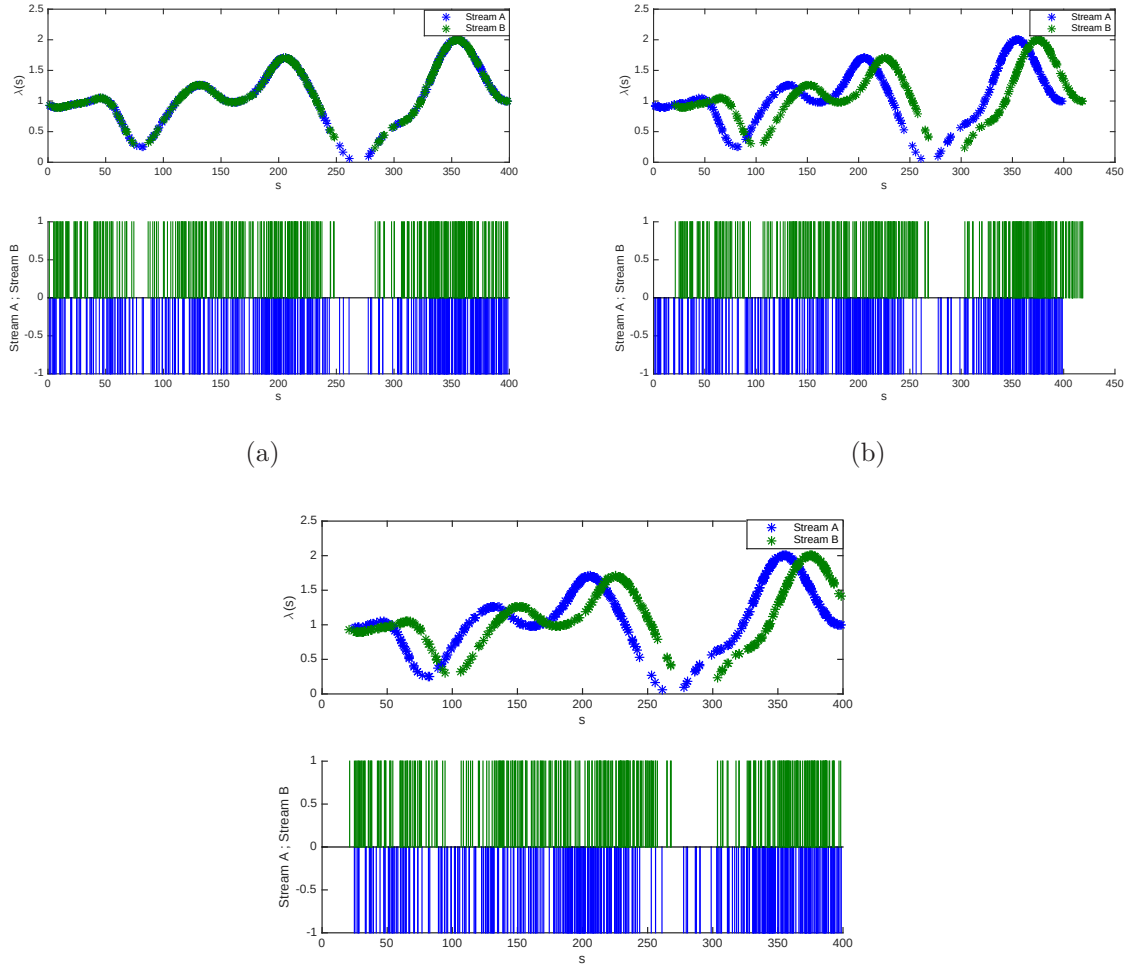
$$s_i^B \leftarrow s_i^B + \Delta, \forall i = 1, 2, \cdots, S^B \tag{5.53}$$

To prepare the streams for experiments, we cut the two streams to ensure they have the same start and end point in time. Figure 5.5 shows an example of the data generation and preparation process.

5.6 Experiments

We performed experiments on synthetic data sets described in Section 5.5. In the experiments we compared our models: PPE and IPEs introduced in Sections 5.2 and 5.3 respectively, with baseline KRE1 and KRE2 (see Section 5.1).

We performed controlled experiments where 100 test rate functions were generated as described in Section 5.5. For each test rate function we imposed four delay values



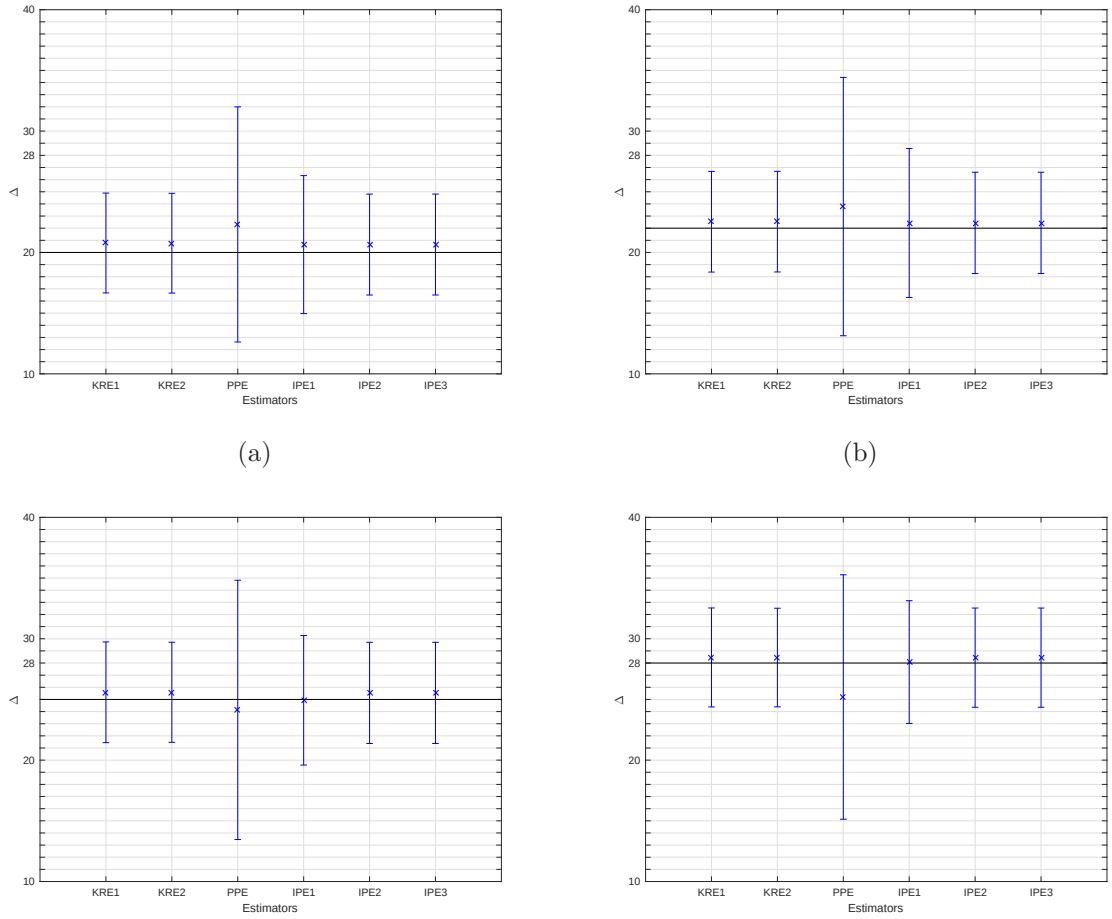
(a) (b)

(c)

Figure 5.5: An example of the data generation and preparation process.

$\Delta \in \{20, 22, 25, 28\}$, resulting in 400 individual experiments. The time delay trial values were taken from the interval $[10, 40]$ with increments of 1.

For each model and each imposed delay $\Delta \in \{20, 22, 25, 28\}$, we report the mean μ and standard deviation σ of the maximum-likelihood delay estimates $\left\{\hat{\Delta}_i\right\}_{i=1}^{100}$ across the set of 100 test rate functions. We also report the mean absolute error (MAE) of the delay estimates and the 95% Credibility Interval (CI). A summary of the results is presented in Figure 5.6 and Tables 5.1, 5.2, 5.3 and 5.4. Furthermore, we produced a global report of the standard deviation, MAE and CI range of the delay estimates across all 400 experiments in Table 5.5.



(a) (b)

(c) (d)

Figure 5.6: Results of experiments across different values of true delay: (a) 20, (b) 22, (c) 25 and (d) 28. $\Delta = [10, 40]$ with increments of 1.

Table 5.1: Statistical analysis of delay estimates. True delay = 20. The results for each method are averaged over 100 test rate functions. The time delay trial values were taken from the interval $[10, 40]$ with increments of 1.

Method	$\mu \pm \sigma$	MAE	CI range	95% CI
KRE1	20.78±4.11	3.08	0.81	[19.97,21.59]
KRE2	20.76±4.11	3.10	0.81	[19.95,21.57]
PPE	22.31±9.69	8.25	1.90	[20.41,24.21]
IPE1	20.65±5.68	4.25	1.11	[19.54,21.76]
IPE2	20.65±4.16	3.12	0.82	[19.83,21.47]
IPE3	20.65±4.16	3.12	0.82	[19.83,21.47]

In order test the performance of all methods in the cases when trial delay values are not rightly specified, we performed controlled experiments on the same test rate functions but this time the delay trial values were taken from the interval $[10, 40]$ with increments

Table 5.2: Statistical analysis of delay estimates. True delay = 22. The results for each method are averaged over 100 test rate functions. The time delay trial values were taken from the interval $[10, 40]$ with increments of 1.

Method	$\mu \pm \sigma$	MAE	CI range	95% CI
KRE1	22.53±4.14	3.19	0.81	[21.72,23.34]
KRE2	22.54±4.15	3.20	0.81	[21.73,23.35]
PPE	23.78±10.64	9.44	2.09	[21.69,25.87]
IPE1	22.43±6.14	4.79	1.20	[21.23,23.63]
IPE2	22.44±4.17	3.21	0.82	[21.62,23.26]
IPE3	22.44±4.17	3.21	0.82	[21.62,23.26]

Table 5.3: Statistical analysis of delay estimates. True delay = 25. The results for each method are averaged over 100 test rate functions. The time delay trial values were taken from the interval $[10, 40]$ with increments of 1.

Method	$\mu \pm \sigma$	MAE	CI range	95% CI
KRE1	25.59±4.15	3.23	0.81	[24.78,26.40]
KRE2	25.59±4.12	3.21	0.81	[24.78,26.40]
PPE	24.14±10.68	9.40	2.09	[22.05,26.23]
IPE1	24.93±5.34	4.17	1.05	[23.88,25.98]
IPE2	25.54±4.17	3.25	0.82	[24.72,26.36]
IPE3	25.54±4.17	3.25	0.82	[24.72,26.36]

Table 5.4: Statistical analysis of delay estimates. True delay = 28. The results for each method are averaged over 100 test rate functions. The time delay trial values were taken from the interval $[10, 40]$ with increments of 1.

Method	$\mu \pm \sigma$	MAE	CI range	95% CI
KRE1	28.46±4.08	3.24	0.80	[27.66,29.26]
KRE2	28.45±4.06	3.22	0.80	[27.65,29.25]
PPE	25.20±10.07	8.88	1.97	[23.23,27.17]
IPE1	28.08±5.06	4.00	0.99	[27.09,29.07]
IPE2	28.44±4.09	3.25	0.80	[27.64,29.24]
IPE3	28.44±4.09	3.25	0.80	[27.64,29.24]

Table 5.5: Overall results across all true delay values where the time delay trial values were taken from the interval $[10, 40]$ with increments of 1.

Method	σ	MAE	CI range
KRE1	5.05	3.19	0.49
KRE2	5.04	3.18	0.49
PPE	10.29	8.99	1.01
IPE1	6.21	4.30	0.61
IPE2	5.09	3.21	0.50
IPE3	5.09	3.21	0.50

of 10. A summary of the results is presented in Figure 5.7 and tables 5.6, 5.7,5.8 and 5.9. A global report of the standard deviation, MAE and CI range of the delay estimates across all 400 experiments in table 5.10.

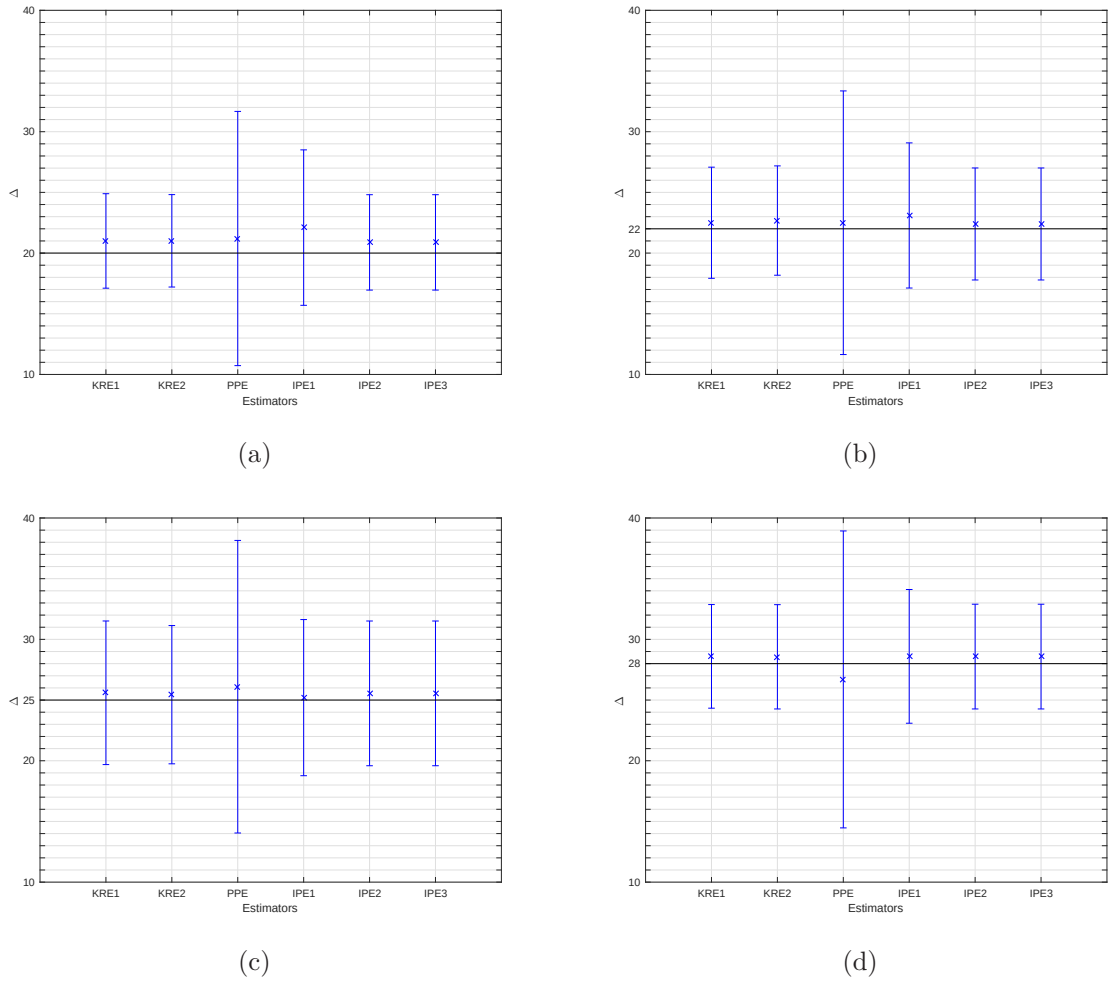


Figure 5.7: Results of experiments across different values of true delay: (a) 20, (b) 22, (c) 25 and (d) 28. The time delay trial values were taken from the interval $[10, 40]$ with increments of 10.

We also performed The Wilcoxon Rank-Sum Test on time delay estimates from selected methods, namely KRE2 and IPE2. The test suggests that the results from KRE2 are more statistically significant than IPE2 only when the time delay trial values were taken from the interval $[10, 40]$ with increments of 10. Examples of reconstructions on test rate functions using KRE1, PPE and IPE1 are shown in Figure 5.8.

Finally, to illustrate robustness of the delay estimators, we plot in Figure 5.9 the values

Table 5.6: Statistical analysis of delay estimates. True delay = 20. The results for each method are averaged over 100 test rate functions. The time delay trial values were taken from the interval $[10, 40]$ with increments of 10.

Method	$\mu \pm \sigma$	MAE	CI range	95% CI
KRE1	21.00±3.89	1.60	0.76	[20.24,21.76]
KRE2	21.20±3.81	1.87	0.75	[20.26,21.76]
PPE	22.10±10.47	8.20	2.05	[19.15,23.25]
IPE1	22.10±6.40	3.30	1.25	[20.85,23.35]
IPE2	20.88±3.93	1.73	0.77	[20.11,21.65]
IPE3	20.88±3.93	1.73	0.77	[20.11,21.65]

Table 5.7: Statistical analysis of delay estimates. True delay = 22; The results for each method are averaged over 100 test rate functions. The time delay trial values were taken from the interval $[10, 40]$ with increments of 10.

Method	$\mu \pm \sigma$	MAE	CI range	95% CI
KRE1	22.50±4.58	3.66	0.90	[21.60,23.40]
KRE2	22.68±4.50	3.49	0.88	[21.80,23.56]
PPE	22.50±10.86	9.18	2.13	[20.37,24.63]
IPE1	23.10±5.98	4.50	1.17	[21.93,24.27]
IPE2	22.40±4.61	3.74	0.90	[21.50,23.30]
IPE3	22.40±4.61	3.74	0.90	[21.50,23.30]

Table 5.8: Statistical analysis of delay estimates. True delay = 25. The results for each method are averaged over 100 test rate functions. The time delay trial values were taken from the interval $[10, 40]$ with increments of 10.

Method	$\mu \pm \sigma$	MAE	CI range	95% CI
KRE1	25.60±5.92	5.50	1.16	[24.44,26.76]
KRE2	25.44±5.70	5.09	1.12	[24.32,26.56]
PPE	26.10±12.05	11.00	2.36	[23.74,28.46]
IPE1	25.20±6.43	5.80	1.26	[23.94,26.46]
IPE2	25.55±5.96	5.54	1.17	[24.38,26.72]
IPE3	25.55±5.96	5.54	1.17	[24.38,26.72]

of E for KRE1, PPE and IPE1 methods, for suggested delays 10, 11, ..., 40, with the true imposed delay set to 20 for one the test rate functions. This picture represents a fairly typical situation - the PPE method suffers from the highest bias, while KRE1 and IPE1 point to a neighborhood of the right delay. Typically, the dip in E around delay of 20 was sharper (more confident estimation) for the IPE1 method than for KRE1. On the other hand, the KRE1 method usually suffers less from local optima. Since the IPE1 method initializes the weights and delay from KRE1, the estimated KRE1 delay typically

Table 5.9: Statistical analysis of delay estimates. True delay = 28. The results for each method are averaged over 100 test rate functions. The time delay trial values were taken from the interval $[10, 40]$ with increments of 10.

Method	$\mu \pm \sigma$	MAE	CI range	95% CI
KRE1	28.60±4.27	3.32	0.84	[27.76,29.44]
KRE2	28.56±4.30	3.27	0.84	[27.72,29.40]
PPE	26.70±12.23	10.94	2.40	[24.30,29.10]
IPE1	28.60±5.51	3.92	1.08	[27.52,29.68]
IPE2	28.58±4.32	3.34	0.85	[27.73,29.43]
IPE3	28.58±4.31	3.34	0.84	[27.74,29.42]

Table 5.10: Overall results across all true delay values where the time delay trial values were taken from the interval $[10, 40]$ with increments of 10.

Method	σ	MAE	CI range
KRE1	5.54	3.52	0.54
KRE2	5.43	3.43	0.53
PPE	11.62	9.83	1.14
IPE1	6.56	4.38	0.64
IPE2	5.60	3.59	0.55
IPE3	5.60	3.59	0.55

positions IPE1 in a local neighborhood of the true delay value.

5.6.1 Sensitivity to baseline intensity and variability of rate function

In this Section we test the sensitivity of the studied methods with respect to two factors related to the test rate functions:

1. **Baseline intensity** - the test rate functions $\lambda(s)$, originally in the range $[0, 2]$, are shifted by a constant $\mathcal{S} \in [0, 2]$, $\lambda(s) \leftarrow \lambda(s) + \mathcal{S}$. Increasing baseline intensity $\mathcal{S}$ can potentially mask the underlying rate function variability (as implicitly represented by the structure of arrival times) and thus destabilize the delay estimations.

2. **Variability** - increasing the number G of kernels when constructing test rate functions will in general increase their variability. For lower number of Gaussians the rate functions will be more "rigid", potentially preventing effective detection of the imposed delay, especially if the delay is small relative to the variability scale of the

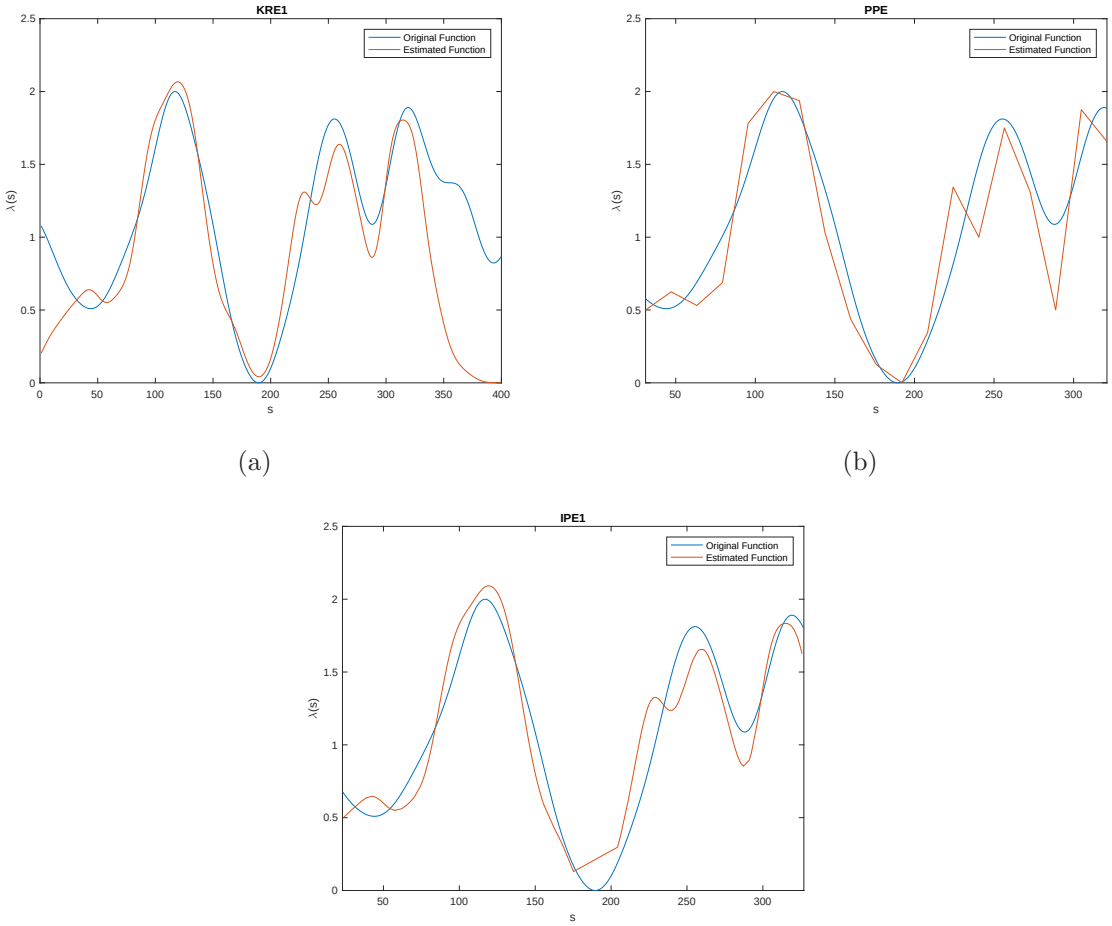
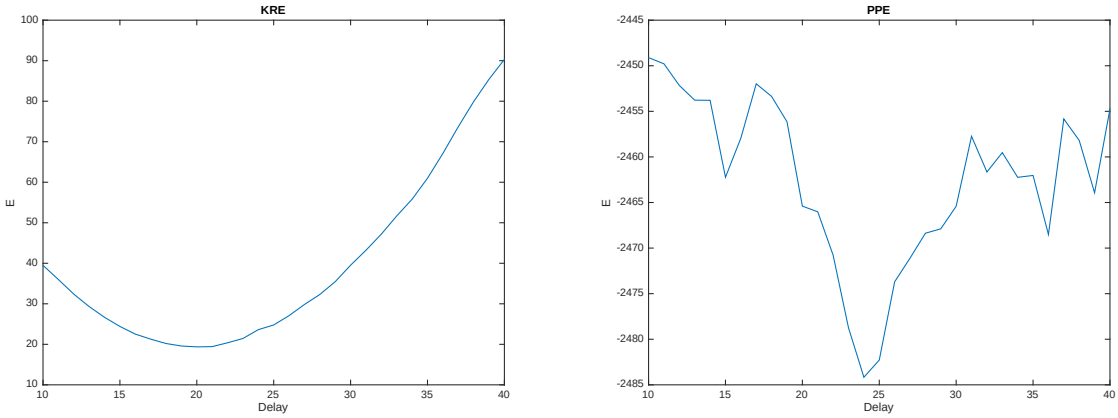


Figure 5.8: Examples of reconstructions on test rate functions.

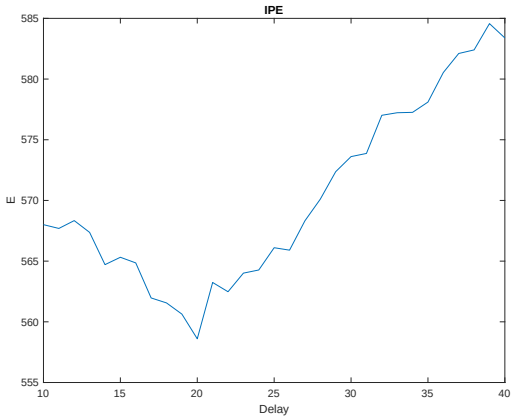
rate function. On the other hand, if the rate functions vary fast, the delay estimation can be hampered by the fact that the highly varying nature of the rate functions will not be adequately reflected in the arrival time structure.

Recall that in the previous experiments, the test rate functions were generated using $G = 80$ Gaussian kernels with a fixed scale (distance between consecutive centers) of 5 units. We will now use $G \in \{40, 100, 400, 800\}$ Gaussian kernels. As before, the kernels are regularly distributed in the time period from $[0, 400]$ with interval $400/G$. We also shift the test rate functions (scaled to $[0, 2]$ by $\mathcal{S} \in \{0, 0.5, 1, 2\}$. The imposed delay was $\Delta = 20$. The results are shown in Figure 5.10. Each plot depicts the average delay estimated over 100 pairs of streams and the error bars represented as 95% confidence



(a)

(b)



(c)

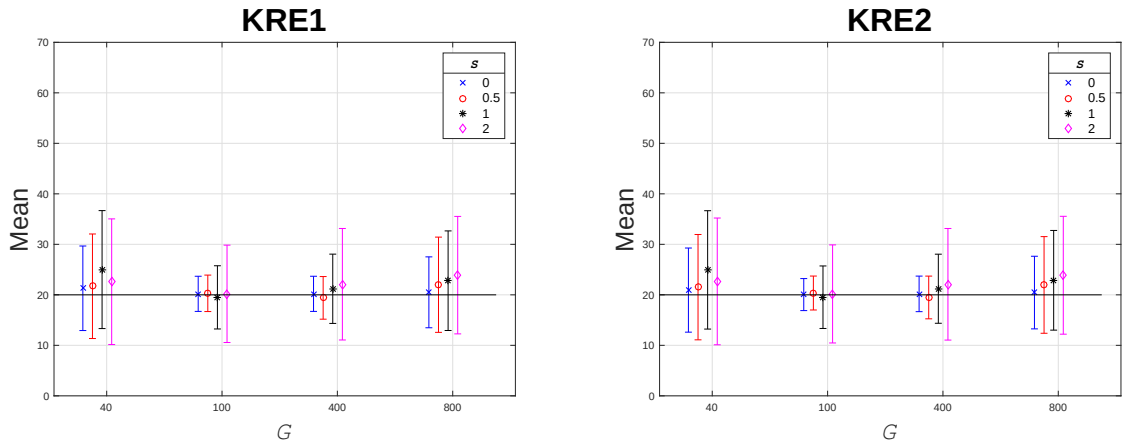
Figure 5.9: E versus Δ for: (a) KRE1, (b) PPE and (c) IPE1.

intervals.

The results indicate that for our setting the optimal rate variability corresponds to 100–400 kernel positions and that the methods, apart from PPE, are reasonably robust with respect to increasing baseline rate intensity. This shows the advantages of using KRE1 in parameters initialization for IPE methods.

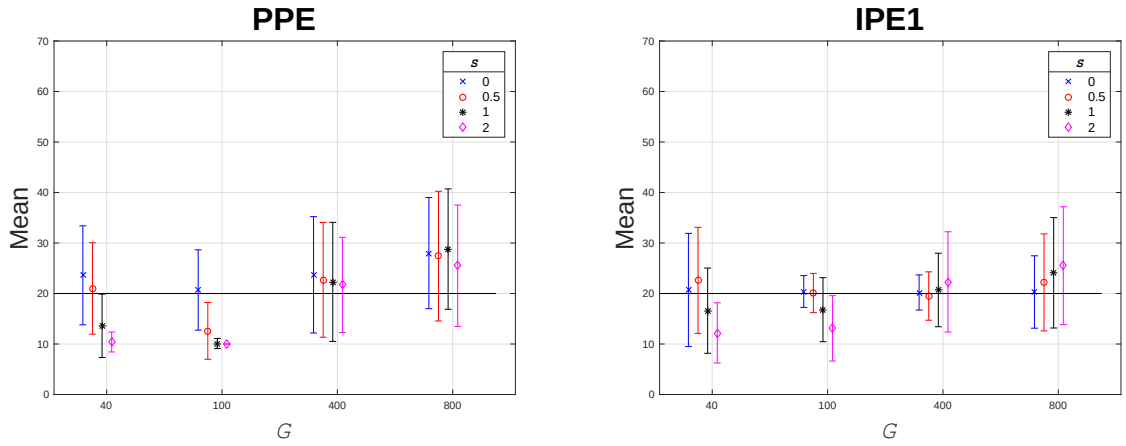
5.7 Summary

We proposed a more principled delay estimation relied on imposing a single latent non-homogeneous Poisson process underlying the lensed photon streams. The rate function



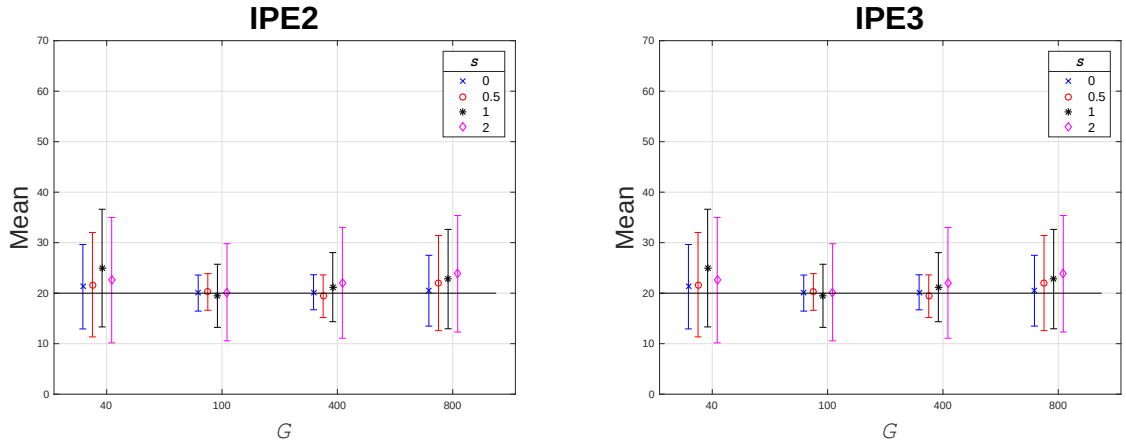
(a) KRE1

(b) KRE2



(c) PPE

(d) IPE1



(e) IPE2

(f) IPE3

Figure 5.10: Results of the experiments of sensitivity to baseline intensity and variability of rate function. Each plot depicts the average delay estimated over 100 pairs of streams and the error bars represented as 95% confidence intervals for: (a) KRE1, (b) KRE2, (c) PPE, (d)IPE1, (e) IPE2 and (f) IPE3.

model was formulated as a linear combination of nonlinear basis functions. We tested this idea in two scenarios - Poisson Process Based Estimation (PPE) and Innovation Process Based Estimation (IPE1, IPE2 and IPE3). In addition, we formulated two baseline methods, KRE1 and KRE2, based on kernel estimation of the rate function of non-homogeneous Poisson process. KRE1 and IPE1 formulation needs a range of suggested trial delays, while KRE2, IPE2 and IPE3 optimize for the delay internally through gradient descent. The IPE3 method optimizes for the kernel width as well using gradient descent while IPE1 and IPE2 use cross-validation for kernel width optimization. We performed controlled experiments on synthetic photon fluxes with known imposed delay in order to compare the baseline with the principled delay estimation methods. We did not perform experiments on real data, since no large real photon streams from known delayed systems with short time delay are available.

CHAPTER 6

CONCLUSIONS

In this chapter we present the general and final conclusions in Sections 6.1 and 6.2. In Section 6.3, we discuss some possible application areas for our methods proposed in Chapter 4 and 5. Finally we briefly introduce some ideas for future research directions in Section 6.4.

6.1 Delay Estimation in Gravitationally Lensed Fluxes

We have introduced a new probabilistic efficient model-based methodology for estimating time delays between two gravitationally lensed images of the same variable point source. The method enables one to use directly the noise levels reported for individual flux measurements. It is more robust to observational gaps than purely 'unmodeled' techniques, since the imposition of an identical smooth model behind multiple lensed fluxes effectively regularizes the overall model fit, and consequently, the time delay estimate itself. Methods such as these will be useful in the automated search for time-delay systems as well as in the accurate measurement of delays in targeted systems in future very large time-domain surveys such as those planned for the LSST (e.g. [52, 69]).

The methods were tested and compared in two experimental settings. In the realistic setting the synthetic data were generated so that multiple aspects of the real data were preserved: noise-to-observed flux ratio, observational gap size distribution and the inter-

gap interval distributions. The core synthetic signals were generated from a GP fitted to the real data. In the larger controlled experimental setting the signals generated from the GP were subject to controlled levels of observational noise and gap sizes. Our method, while being computationally efficient, showed robustness with respect to noise levels and observational gap sizes.

We also applied our method to real observed optical and radio fluxes from quasar Q0957+561 as a combined data set. Of course, with real data one can estimate the variance of the estimator estimations, but never the bias, since the true time delay for Q0957+561 is not known. Our NWE estimator on the combined optical and radio data suggests a delay of approximately 420 days; however, we find that different estimators produce inconsistent results, indicating the presence of statistical or systematic measurement errors in the data in excess of the claimed measurement uncertainty. In particular, the impact of microlensing corrections was not accounted for in this thesis, and needs to be quantified in the future.

6.2 Delay Estimation in Gravitationally Lensed Photon Streams

We tested whether a more principled treatment of delay estimation in lensed photon streams, compared with the standard kernel estimation method, can have benefits of a more accurate (less biased) and/or more stable (less variance) estimation. In particular, we formulated two baseline methods, KRE1 and KRE2, based on kernel estimation of the rate function of non-homogeneous Poisson process. Unlike KRE1, KRE2 does not have to rely on the rightly specified trial delay values. Instead, the delay estimate is refined using gradient descent in the delay parameter on the error functional.

A more principled delay estimation relied on imposing a single latent non-homogeneous Poisson process underlying the lensed photon streams. The rate function model was formulated as a linear combination of nonlinear basis functions, thus making the non-linear

model linear in the mixing parameters. We tested this idea in two scenarios - Poisson Process Based Estimation (PPE) and Innovation Process Based Estimation (IPE1, IPE2 and IPE3). As KRE1, the IPE1 formulation needs a range of suggested trial delays, while IPE2 optimizes for the delay internally through gradient descent. In addition, the IPE3 method optimizes for the kernel width using gradient descent (unlike IPE1 and IPE2 that use cross-validation).

Somewhat surprisingly, the overall emerging picture is that the theoretically more principled methods do not bring much practical benefit in terms of the bias/variance of the delay estimation. This is in contrast to our previous findings on daily flux data [2, 24, 25]. It appears that the fact that underlying latent rate function is represented only implicitly through the streams of arrival times weakens the stabilizing factor of the single unified intensity function that proved so useful in the case of daily flux data [2, 24, 25]. Indeed, in that case, knowing the amount of observational noise and observing noisy flux levels gave much better clues as to what the common source variability could be, thus stabilizing the delay estimation. Nevertheless, we propose that a study of the kind is useful and necessary for future developments of alternative methods for time delay estimation in lensed photon streams.

6.3 Applications

An accurate estimation of the time delay between two gravitationally lensed fluxes can be used to measure the parameters of the universe such as the Hubble constant, and the expansion rate, which used to predict the age and future of the universe. It also indicates the distribution of matter in the universe and, therefore, it is considered to be the most direct way to measure the matter in the universe. The proposed methodology in this thesis (see Chapters 4 and 5) can be successfully applied to real world problems such as the gravitational lensing phenomenon or to any other quasars. This is an active area of research in astrophysics, especially in view of the upcoming surveys such as Large Synoptic

Survey telescope (LSST), which will provide unprecedented data sets with strongly lensed distant quasars.

Problems with irregularly sampled noisy data can be found in all research areas. The proposed methods in Chapter 4 are able to cope with the inevitable noise and gap features of the data. In general these methods can be applied in any other scenarios involving similar time series data that are delayed or corrupted by gaps and noise where the source is represented by a hidden underlying function.

6.4 Future work

Many research directions arise in this area. In this section we will introduce some ideas of possible extensions for the current work.

6.4.1 Delay estimation in gravitationally lensed fluxes

As mentioned in the discussion of Chapter 4, we find that estimates using different frequency estimates on Q0957+561 data appear to be inconsistent even when the same method is used. This suggests that, there may be unmodeled systematics (e.g., micro-lensing) that lead to varied biases for different analysis techniques. Therefore, studying the effect of micro-lensing is a research direction to follow. Another research direction is Supernova modeling and effect since supernova events can cause multiple time delays for the same quasar.

6.4.2 Delay estimation in gravitationally lensed photon streams

We will test our methods on real data when it is made available to us by the astronomers and comparing the results to the ones obtained by other researchers on the same data sets. Prior information from real data can also be incorporated to improve our models.

LIST OF REFERENCES

[1] F. Abdalla, A. Mateus, W. Santos, L. Sodrè, I. Ferreras, and O. Lahav. Predicting spectral features in galaxy spectra from broad-band photometry. *Monthly Notices of the Royal Astronomical Society*, 387(3):945–953, 2008.

[2] S. AL Otaibi, P. Tiňo, J. C. Cuevas-Tello, I. Mandel, and S. Raychaudhury. Kernel regression estimates of time delays between gravitationally lensed fluxes. *MNRAS* , 459(1):573–584, 2016.

[3] S. Bailey, C. Aragon, R. Romano, R. C. Thomas, B. A. Weaver, and D. Wong. How to find more supernovae with less work: Object classification techniques for difference imaging. *The Astrophysical Journal*, 665(2):1246, 2007.

[4] N. M. Ball and R. J. Brunner. Data mining and machine learning in astronomy. *International Journal of Modern Physics D*, 19(07):1049–1106, 2010.

[5] N. M. Ball, R. J. Brunner, A. D. Myers, and D. Tcheng. Robust machine learning applied to astronomical data sets. i. star-galaxy classification of the sloan digital sky survey dr3 using decision trees. *The Astrophysical Journal*, 650(1):497, 2006.

[6] M. Bazarghan and R. Gupta. Automated classification of sloan digital sky survey (sdss) stellar spectra using artificial neural networks. *Astrophysics and Space Science*, 315(1-4):201–210, 2008.

[7] D. P. Bertsekas. *Introduction to Probability*. Athena Scientific, 2002.

[8] C. M. Bishop. *Neural networks for pattern recognition*. Oxford university press, 1995.

[9] R. Blandford and R. Narayan. Cosmological applications of gravitational lensing. *Annual Review of Astronomy and Astrophysics*, 30:311–358, 1992.

[10] P. Bratley, B. Fox, and L. E. Schrage. A guide to simulation, 2nd, 1987.

[11] M. C. Burl, L. Asker, P. Smyth, U. Fayyad, P. Perona, L. Crumpler, and J. Aubele. Learning to recognize volcanoes on venus. *Machine Learning*, 30(2-3):165–194, 1998.

[12] I. Burud, P. Magain, S. Sohy, and J. Hjorth. A novel approach for extracting time-delays from lightcurves of lensed quasar images. *Astronomy & Astrophysics*, 380(02):805–810, 2001.

[13] A. Chipperfield, P. Fleming, and C. Fonseca. Genetic algorithm tools for control systems engineering. In *Proceedings of Adaptive Computing in Engineering Design and Control*, pages 128–133, 1994.

[14] A. Chipperfield, P. Fleming, H. Pohlheim, and C. Fonseca. Genetic algorithm toolbox for use with matlab. 1994.

[15] W. N. Colley, R. E. Schild, C. Abajas, D. Alcalde, Z. Aslan, I. Bikmaev, V. Chavushyan, L. Chinarro, J.-P. Cournoyer, R. Crowe, et al. Around-the-clock observations of the q0957+ 561a, b gravitationally lensed quasar. ii. results for the second observing season. *The Astrophysical Journal*, 587(1):71, 2003.

[16] A. A. Collister and O. Lahav. Annz: estimating photometric redshifts using artificial neural networks. *Publications of the Astronomical Society of the Pacific*, 116(818):345, 2004.

[17] A. Connolly and A. Szalay. A robust classification of galaxy spectra: Dealing with noisy and incomplete data. *The Astronomical Journal*, 117(5):2052, 1999.

[18] A. Connolly, A. Szalay, M. Bershady, A. Kinney, and D. Calzetti. Spectral classification of galaxies: an orthogonal approach. *arXiv preprint astro-ph/9411044*, 1994.

[19] F. Courbin, V. Chantry, Y. Revaz, D. Sluse, C. Faure, M. Tewes, E. Eulaers, M. Koleva, I. Asfandiyarov, S. Dye, P. Magain, H. van Winckel, J. Coles, P. Saha, M. Ibrahimov, and G. Meylan. COSMOGRAIL: the COSmological MOnitoring of GRAvItational Lenses. IX. Time delays, lens dynamics and baryonic fraction in HE 0435-1223. *Astronomy & Astrophysics*, 536, Dec. 2011.

[20] G. Cowan. *Statistical data analysis*. Oxford University Press, 1998.

[21] J. C. Cuevas-Tello. *Estimating Time Delays between Irregularly Sampled Time Series*. PhD thesis, School of Computer Science, University of Birmingham, 2007. http://etheses.bham.ac.uk/88/.

[22] J. C. Cuevas-Tello, R. Gonzalez-Grimaldo, O. Rodriguez-Gonzalez, H. Perez-Gonzalez, and O. Vital-Ochoa. Parallel Approach for Time Series Analysis with General Regression Neural Networks. *Journal of Applied Research and Technology*, 10(2):162–179, 2012.

[23] J. C. Cuevas-Tello and H. Perez-Gonzalez. Multi-Objective Optimisation in Time Series: Time Delay Agreement. *Image*, 17:17–4, 2011.

[24] J. C. Cuevas-Tello, P. Tiňo, and S. Raychaudhury. How Accurate Are the Time Delay Estimates in Gravitational lensing? *Astronomy & Astrophysics*, 454:695–706, Aug. 2006.

[25] J. C. Cuevas-Tello, P. Tiňo, S. Raychaudhury, X. Yao, and M. Harva. Uncovering Delayed Patterns in Noisy and Irregularly Sampled Time Series: An Astronomy Application. *Pattern Recognition*, 43(3):1165–1179, Aug. 2009.

[26] S. Djorgovski, A. Mahabal, R. Brunner, R. Gal, S. Castro, R. De Carvalho, and S. Odewahn. Searches for rare and new types of objects. *arXiv preprint astro-ph/0012453*, 2000.

[27] G. Dobler, C. D. Fassnacht, T. Treu, P. Marshall, K. Liao, A. Hojjati, E. Linder, and N. Rumbaugh. Strong Lens Time Delay Challenge. I. Experimental Design. *Astrophysical Journal* , 799:168, Feb. 2015.

[28] R. Edelson and J. Krolik. The Discrete C orrelation-function - A New Method for Analyzing Unevenly Sampled Variability Data. *The Astrophysical Journal*, 333(2):646–659, 1988.

[29] A. Eigenbrod, F. Courbin, C. Vuissoz, G. Meylan, P. Saha, and S. Dye. COSMOGRAIL: the COSmological MOnitoring of GRAvItational Lenses. I. How to sample the light curves of gravitationally lensed quasars to measure accurate time delays. *Astronomy & Astrophysics*, 436:25–35, June 2005.

[30] E. Eulaers, M. Tewes, P. Magain, F. Courbin, I. Asfandiyarov, S. Ehgamberdiev, S. Rathna Kumar, C. S. Stalin, T. P. Prabhu, G. Meylan, and H. Van Winckel. COSMOGRAIL: the COSmological MOnitoring of GRAvItational Lenses. XII. Time delays of the doubly lensed quasars SDSS J1206+4332 and HS 2209+1914. *Astronomy & Astrophysics*, 553, May 2013.

[31] L. Eyer, A. Jan, P. Dubath, K. Nienartovicz, J. Blomme, J. Debosscher, J. De Ridder, M. Lopez, and L. Sarro. Variability type classification of multi-epoch surveys. *American Institute of Physics Conference Series*, 1082(1):257–262, 2008.

[32] R. Fadely, C. R. Keeton, R. Nakajima, and G. M. Bernstein. Improved Constraints on the Gravitational Lens Q0957+561. II. Strong Lensing. *Astrophysical Journal* , 711:246–267, Mar. 2010.

[33] C. D. Fassnacht, E. Xanthopoulos, L. V. E. Koopmans, and D. Rusin. A Determination of H_0 with the CLASS Gravitational Lens B1608+656. III. A Significant

Improvement in the Precision of the Time Delay Measurements. *Astrophysical Journal* , 581:823–835, Dec. 2002.

[34] B. Fathi-Vajargah and H. Khoshkar-Foshtomi. Simulating nonhomogeneous poisson point process based on multi criteria intensity function and comparison with its simple form. *Journal of Mathematics and Computer Science (JMCS)*, 9(3):133–138, 2014.

[35] J. Fohlmeister, C. S. Kochanek, E. E. Falco, J. Wambsganss, M. Oguri, and X. Dai. A Two-year Time Delay for the Lensed Quasar SDSS J1029+2623. *The Astrophysical Journal*, 764(2):186, Feb. 2013.

[36] A. Ghosh and S. Dehuri. Evolutionary Algorithms for Multi-criterion Optimization: A Survey. *International Journal of Computing & Information Sciences*, 2(1):38–57, 2004.

[37] J. Goebel, J. Stutz, K. Volk, H. Walker, F. Gerbault, M. Self, W. Taylor, and P. Cheeseman. A bayesian classification of the iras lrs atlas. *Astronomy and Astrophysics*, 222:L5–L8, 1989.

[38] D. E. Goldberg et al. *Genetic algorithms in search optimization and machine learning*, volume 412. Addison-wesley Reading Menlo Park, 1989.

[39] G. Golub and W. Kahan. Calculating the singular values and pseudo-inverse of a matrix. *Journal of the Society for Industrial and Applied Mathematics, Series B: Numerical Analysis*, 2(2):205–224, 1965.

[40] G. Golub and C. V. Loan. *Matrix Computations*. The Johns Hopkins University Press, second edition edition, 1989.

[41] R. Gonzalez-Grimaldo and J. C. Cuevas-Tello. Analysis of time series with neural networks. In *Proceedings of 7th Mexican International Conference on Artificial Intelligence (MICAI)*, pages 131–137. IEEE Computer Society, Nov. 2008.

[42] M. Gorenstein, I. Shapiro, and E. Falco. Degeneracies in parameter estimates for models of gravitational lens systems. *The Astrophysical Journal*, 327:693–711, 1988.

[43] Z. S. Greene, S. H. Suyu, T. Treu, S. Hilbert, M. W. Auger, T. E. Collett, P. J. Marshall, C. D. Fassnacht, R. D. Blandford, M. Bradač, and L. V. E. Koopmans. Improving the Precision of Time-delay Cosmography with Observations of Galaxies along the Line of Sight. *The Astrophysical Journal*, 768(1):39, May 2013.

[44] D. Haarsma, J. Hewitt, J. Lehar, and B. Burke. The Radio Wavelength Time Delay of Gravitational Lens 0957+561. *The Astrophysical Journal*, 510(1):64–70, 1999.

[45] D. B. Haarsma, J. N. Hewitt, J. Lehar, and B. F. Burke. The radio wavelength time delay of gravitational lens 0957+ 561. *The Astrophysical Journal*, 510(1):64, 1999.

[46] L. J. Hainline, C. W. Morgan, J. N. Beach, C. S. Kochanek, H. C. Harris, T. Tilleman, R. Fadely, E. E. Falco, and T. X. Le. A New Microlensing Event in the Doubly Imaged Quasar Q 0957+561. *The Astrophysical Journal*, 744(2):104, Jan. 2012.

[47] M. Harva and S. Raychaudhury. A new bayesian look at estimating gravitational lens time delays. In the School of Physics and A. U. of Birmingham, editors, *RAS National Astronomy Meeting: Bayesian techniques in astronomy*. Royal Astronomical Society, Apr. 2005.

[48] T. Hastie, R. Tibshirani, J. Friedman, and J. Franklin. The elements of statistical learning: data mining, inference and prediction. *The Mathematical Intelligencer*, 27(2):83–85, 2005.

[49] S. Haykin. *Neural networks: a comprehensive foundation*. London [etc.]: Prentice Hall, 1999.

[50] A. Hirv, N. Olspert, and J. Pelt. Towards the automatic estimation of gravitational lenses' time delays. *arXiv preprint arXiv:1105.5991*, 2011.

[51] A. Hojjati, A. G. Kim, and E. V. Linder. Robust strong lensing time delay estimation. *Phys. Rev. D* , 87(12):123512, June 2013.

[52] A. Hojjati and E. Linder. . *Physical Review D*, 90:123501, 2015.

[53] J. Holland. *Adaptation in Natural and Artificial Systems*. MIT Press, 1992.

[54] E. P. Hubble. Extragalactic nebulae. *The Astrophysical Journal*, 64, 1926.

[55] Ž. Ivezić, A. Connolly, J. VanderPlas, and A. Gray. *Statistics, Data Mining, and Machine Learning in Astronomy: A Practical Python Guide for the Analysis of Survey Data*. Princeton Series in Modern Observational Astronomy. Princeton University Press, 2014.

[56] Z. Ivezic, J. Tyson, B. Abel, E. Acosta, R. Allsman, Y. AlSayyad, S. Anderson, J. Andrew, R. Angel, G. Angeli, et al. Lsst: from science drivers to reference design and anticipated data products. *arXiv preprint arXiv:0805.2366*, 2008.

[57] I. Jolliffe. *Principal component analysis*. Springer, New York, 2002.

[58] C. Kamath. Sapphire: experiences in scientific data mining. In *Journal of Physics: Conference Series*, volume 125, page 012094. IOP Publishing, 2008.

[59] C. S. Kochanek, N. D. Morgan, E. E. Falco, B. A. McLeod, J. N. Winn, J. Dembicky, and B. Ketzeback. The Time Delays of Gravitational Lens HE 0435-1223: An Early-Type Galaxy with a Rising Rotation Curve. *Astrophysical Journal* , 640:47–61, Mar. 2006.

[60] T. Kohonen. Self-organizing maps, vol. 30 of springer series in information sciences. *ed: Springer Berlin*, 2001.

[61] A. Konak, D. W. Coit, and A. E. Smith. Multi-objective Optimization Using Genetic Algorithms: A Tutorial. *Reliability Engineering & System Safety*, 91(9):992–1007, 2006.

[62] S. Kotz and N. L. Johnson. *Distributions in statistics: discrete distributions*. 1969.

[63] T. Kundic, E. Turner, W. Colley, J. Gott-III, J. Rhoads, Y. Wang, L. bergeron, K. Gloria, D. Long, S. Malhorta, and J. Wambsganss. A Robust Determination of the Time Delay in 0957+561A,B and A Measurement of the Global Value of Hubble's Constant. *The Astrophysical Journal*, 482(1):75–82, 1997.

[64] T. Kundić, E. L. Turner, W. N. Colley, J. R. Gott III, J. E. Rhoads, Y. Wang, L. E. Bergeron, K. A. Gloria, D. C. Long, S. Malhotra, et al. A robust determination of the time delay in 0957+ 561a, b and a measurement of the global value of hubble's constant. *The Astrophysical Journal*, 482(1):75, 1997.

[65] M. Kurtz. Handbook of applied mathematics for engineers and scientists. 1991.

[66] O. Lahav, A. Naim, R. Buta, H. Corwin, G. De Vaucouleurs, A. Dressler, J. Huchra, S. Bergh, S. Raychaudhury, L. Sodre Jr, et al. Galaxies, human eyes and artificial neural networks. *arXiv preprint astro-ph/9412027*, 1994.

[67] J. Lehar, J. Hewitt, D. Roberts, and B. Burke. The Radio Time Delay in the Double Quasar 0957+561. *The Astrophysical Journal*, 384:453–466, 1992.

[68] P. A. Lewis and G. S. Shedler. Simulation of nonhomogeneous poisson processes by thinning. *Naval Research Logistics Quarterly*, 26(3):403–413, 1979.

[69] K. Liao, T. Treu, P. Marshall, C. D. Fassnacht, N. Rumbaugh, G. Dobler, A. Aghamousa, V. Bonvin, F. Courbin, A. Hojjati, N. Jackson, V. Kashyap, S. Rathna Kumar, E. Linder, K. Mandel, X.-L. Meng, G. Meylan, L. A. Moustakas, T. P. Prabhu, A. Romero-Wolf, A. Shafieloo, A. Siemiginowska, C. S. Stalin, H. Tak, M. Tewes, and D. van Dyk. Strong Lens Time Delay Challenge. II. Results of TDC1. *Astrophysical Journal* , 800:11, Feb. 2015.

[70] E. V. Linder. Lensing Time Delays and Cosmological Complementarity. *Physical Review D*, 84(12):123529, Dec. 2011.

[71] H. Lu, H. Zhou, J. Wang, T. Wang, X. Dong, Z. Zhuang, and C. Li. Ensemble learning for independent component analysis of normal galaxy spectra. *The Astronomical Journal*, 131(2):790, 2006.

[72] J. MacQueen et al. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297, 1967.

[73] D. Madgwick, O. Lahav, K. Taylor, et al. Parameterisation of galaxy spectra in the 2df galaxy redshift survey. In *Mining the Sky*, pages 331–336. Springer, 2001.

[74] D. S. Madgwick. Correlating galaxy morphologies and spectra in the 2df galaxy redshift survey. *Monthly Notices of the Royal Astronomical Society*, 338(1):197–207, 2003.

[75] A. Mahabal, S. Djorgovski, M. Turmon, J. Jewell, R. Williams, A. Drake, M. Graham, C. Donalek, E. Glikman, et al. Automated probabilistic classification of transients and variables. *Astronomische Nachrichten*, 329(3):288–291, 2008.

[76] T. McGlynn, A. Suchkov, E. Winter, R. Hanisch, R. White, F. Ochsenbein, S. Derriere, W. Voges, M. Corcoran, S. Drake, et al. Automated classification of rosat sources using heterogeneous multiwavelength source catalogs. *The Astrophysical Journal*, 616(2):1284, 2004.

[77] G. McLachlan and D. Peel. Finite mixture models: Wiley series in probability and mathematical statistics, 2000.

[78] A. Miller and M. Coe. Star/galaxy classification using kohonen self-organizing maps. *Monthly Notices of the Royal Astronomical Society*, 279(1):293–300, 1996.

[79] T. M. Mitchell. Machine learning (international edition). *Computer Science Series. McGraw-Hill, New York*, 1997.

[80] W. W. Morgan and N. Mayall. A spectral classification of galaxies. *Publications of the Astronomical Society of the Pacific*, 69(409):291–303, 1957.

[81] H. Mühlenbein and D. Schlierkamp-Voosen. Predictive models for the breeder genetic algorithm i. continuous parameter optimization. *Evolutionary computation*, 1(1):25–49, 1993.

[82] T. Murata and H. Ishibuchi. Moga: multi-objective genetic algorithms. In *Evolutionary Computation, 1995., IEEE International Conference on*, volume 1, page 289. IEEE, 1995.

[83] T. Murata, H. Ishibuchi, and H. Tanaka. Multi-objective Genetic Algorithm and its Applications to Flowshop Scheduling. *Computers & Industrial Engineering*, 30(4):957–968, 1996.

[84] E. Nadaraya. On estimating regression. *Theory of Probability and its Applications*, 9(1):141–142, 1964.

[85] A. Naim, O. Lahav, R. Buta, H. Corwin, G. De Vaucouleurs, A. Dressler, J. Huchra, S. Van den Bergh, S. Raychaudhury, L. Sodre, et al. A comparative study of morphological classifications of apm galaxies. *Monthly Notices of the Royal Astronomical Society*, 274(4):1107–1125, 1995.

[86] A. Naim, O. Lahav, L. Sodre, and M. Storrie-Lombardi. Automated morphological classification of apm galaxies by supervised artificial neural networks. *Monthly Notices of the Royal Astronomical Society*, 275(3):567–590, 1995.

[87] A. Naim, K. U. Ratnatunga, and R. E. Griffiths. Quantitative morphology of moderate-redshift galaxies: How many peculiar galaxies are there? *The Astrophysical Journal*, 476(2):510, 1997.

[88] M. Nawrot, A. Aertsen, and S. Rotter. Single-trial estimation of neuronal firing

rates: from single-neuron spike trains to population activity. *Journal of neuroscience methods*, 94(1):81–92, 1999.

[89] P. R. Newbury and R. J. Spiteri. Inverting gravitational lenses. *SIAM review*, 44(1):111–130, 2002.

[90] S. Odewahn and M. Nielsen. Star-galaxy separation using neural networks. *Vistas in Astronomy*, 38:281–286, 1994.

[91] S. Odewahn, E. Stockwell, R. Pennington, R. Humphreys, and W. Zumach. Automated star/galaxy discrimination with neural networks. In *Digitised Optical Sky Surveys*, pages 215–224. Springer, 1992.

[92] M. Oguri. Gravitational Lens Time Delays: A Statistical Assessment of Lens Model Dependences and Implications for the Global Hubble Constant. *The Astrophysical Journal*, 660:1–15, May 2007.

[93] A. Oscoz, D. Alcalde, M. Serra-Ricart, E. Mediavilla, C. Abajas, R. Barrena, J. Licandro, V. Motta, and J. A. Munoz. Time delay in qso 0957+561 from 1984-1999 optical data. *The Astrophysical Journal*, 552(1):81, 2001.

[94] A. Oscoz, E. Mediavilla, M. Serra-Ricart, et al. Time delay of qso 0957+ 561 and cosmological implications. *The Astrophysical Journal Letters*, 479(2):L89, 1997.

[95] J. Ovaldsen, J. Teuber, R. Schild, and R. Stabell. New Aperture Photometry of QSO 0957+561; Application to Time Delay and Microlensing. *Astronomy & Astrophysics*, 402(3):891–904, 2003.

[96] J.-E. Ovaldsen, J. Teuber, R. Schild, and R. Stabell. New aperture photometry of QSO 0957+561: Application to time delay and microlensing. *Astronomy & Astrophysics*, 402:891–904, 2003.

[97] B. U. Park and J. S. Marron. Comparison of data-driven bandwidth selectors. *Journal of the American Statistical Association*, 85(409):66–72, 1990.

[98] E. Parzen. On estimation of a probability density function and mode. *The annals of mathematical statistics*, 33(3):1065–1076, 1962.

[99] J. Pelt, R. Kayser, S. Refsdal, and T. Schramm. The Light Curve and the Time Delay of QSO 0957+561. *Astronomy & Astrophysics*, 305(1):97–106, 1996.

[100] J. Pelt, R. Schild, S. Refsdal, and R. Stabell. Microlensing on Different Timescales in the Light Curves of QSO 0957+561 A,B. *Astronomy & Astrophysics*, 336(3):829–839, 1998.

[101] J. Pelt, R. Schild, S. Refsdal, and R. Stabell. Microlensing on different timescales in the lightcurves of qso 0957+ 561 a, b. *Astronomy and Astrophysics*, 336:829–839, 1998.

[102] K. B. Petersen, M. S. Pedersen, et al. The matrix cookbook. *Technical University of Denmark*, 7:15, 2008.

[103] F. P. Pijpers. The determination of time delays as an inverse problem-the case of the double quasar 0957+ 561. *Monthly Notices of the Royal Astronomical Society*, 289(4):933–944, 1997.

[104] T. Press. *Vetterling and Flannery,Numerical Recipes in C++,*. Cambridge University Press, 2002.

[105] D. Pyle. Data preparation for data mining (the morgan kaufmann series in data management systems). 1999.

[106] G. Rasch. The poisson process as a model for a diversity of behavioral phenomena. In *International Congress of Psychology*, volume 2, page 2, 1963.

[107] S. Rathna Kumar, M. Tewes, C. S. Stalin, F. Courbin, I. Asfandiyarov, G. Meylan, E. Eulaers, T. P. Prabhu, P. Magain, H. Van Winckel, and S. Ehgamberdiev. COSMOGRAIL: the COSmological MOnitoring of GRAvItational Lenses XIV. Time

delay of the doubly lensed quasar SDSS J1001+5027. *Astronomy & Astrophysics*, 553, May 2013.

[108] S. Refsdal. On the possibility of determining Hubble's parameter and the masses of galaxies from the gravitational lens effect. *MNRAS* , 128:307, 1964.

[109] S. Refsdal and S. Rosseland. On the possibility of determining the distances and masses of stars from the gravitational lens effect. *Monthly Notices of the Royal Astronomical Society*, 134(3):315–319, 1966.

[110] L. Rokach and O. Maimon. *Data mining with decision trees: theory and applications*. World scientific, 2014.

[111] S. M. Ross. *Simulation, Fourth Edition*. Academic Press, Inc., Orlando, FL, USA, 2006.

[112] S. M. Ross. *Introduction to probability models*. Academic press, 2014.

[113] R. Y. Rubinstein and D. P. Kroese. *Simulation and the Monte Carlo method*, volume 707. John Wiley & Sons, 2011.

[114] P. Saha. Gravitational lensing. *Encyclopedia of Astronomy and Astrophysics*, 2000.

[115] P. L. Schechter. The hubble constant from gravitational lens time delays. *Proceedings of the International Astronomical Union*, 2004(IAUS225):281–296, 2004.

[116] R. Schild and D. Thomson. The Q0957+561 Time Delay from Optical Data. *The Astronomical Journal*, 113(1):130–135, 1997.

[117] G. Shakhnarovich, P. Indyk, and T. Darrell. *Nearest-neighbor methods in learning and vision: theory and practice*. 2006.

[118] G. A. Shields. A brief history of active galactic nuclei. *Publications of the Astronomical Society of the Pacific*, 111(760):661, 1999.

[119] H. Shimazaki and S. Shinomoto. A method for selecting the bin size of a time histogram. *Neural computation*, 19(6):1503–1527, 2007.

[120] H. Shimazaki and S. Shinomoto. Kernel bandwidth optimization in spike rate estimation. *Journal of computational neuroscience*, 29(1-2):171–182, 2010.

[121] K. Sigman. Poisson processes and compound (batch) poisson processes. *Lecture Notes. Columbia University, USA. http://www. columbia. edu/˜ ks20/4703-Sigman/4703-07-Notes-PP-NSPP. pdf*, 2007.

[122] I. Steinwart and A. Christmann. *Support vector machines*. Springer Science & Business Media, 2008.

[123] M. Storrie-Lombardi, M. Irwin, T. von Hippel, and L. Storrie-Lombardi. Spectral classification with principal component analysis and artificial neural networks. *Vistas in Astronomy*, 38:331–340, 1994.

[124] M. Storrie-Lombardi, O. Lahav, L. Sodre, and L. Storrie-Lombardi. Morphological classification of galaxies by artificial neural networks. *Monthly Notices of the Royal Astronomical Society*, 259(1):8P–12P, 1992.

[125] S. H. Suyu, M. W. Auger, S. Hilbert, P. J. Marshall, M. Tewes, T. Treu, C. D. Fassnacht, L. V. E. Koopmans, D. Sluse, R. D. Blandford, F. Courbin, and G. Meylan. Two Accurate Time-delay Distances from Strong Lensing: Implications for Cosmology. *The Astrophysical Journal*, 766(2):70, Apr. 2013.

[126] S. H. Suyu, P. J. Marshall, M. W. Auger, S. Hilbert, R. D. Blandford, L. V. E. Koopmans, C. D. Fassnacht, and T. Treu. Dissecting the Gravitational lens B1608+656. II. Precision Measurements of the Hubble Constant, Spatial Curvature, and the Dark Energy Equation of State. *Astrophysical Journal* , 711:201–221, Mar. 2010.

[127] M. Tewes, F. Courbin, and G. Meylan. Cosmograil: the cosmological monitoring

of gravitational lenses-xi. techniques for time delay measurement in presence of microlensing. *Astronomy & Astrophysics*, 553:A120, 2013.

[128] M. Tewes, F. Courbin, G. Meylan, C. S. Kochanek, E. Eulaers, N. Cantale, A. M. Mosquera, P. Magain, H. Van Winckel, D. Sluse, G. Cataldi, D. Vörös, and S. Dye. COSMOGRAIL: the COSmological MOnitoring of GRAvItational Lenses. XIII. Time delays and 9-yr optical monitoring of the lensed quasar RX J1131-1231. *Astronomy & Astrophysics*, 556:A22, 2013.

[129] T. Treu, P. Marshall, F.-Y. Cyr-Racine, C. Fassnacht, C. Keeton, E. Linder, L. Moustakas, M. Bradac, E. Buckley-Geer, T. Collett, et al. Dark energy with gravitational lens time delays. *arXiv preprint arXiv:1306.1272*, 2013.

[130] S. Van den Bergh. *Galaxy morphology and classification*. Cambridge University Press, 1998.

[131] C. Vuissoz, F. Courbin, D. Sluse, G. Meylan, V. Chantry, E. Eulaers, C. Morgan, M. E. Eyler, C. S. Kochanek, J. Coles, P. Saha, P. Magain, and E. E. Falco. COSMOGRAIL: the COSmological MOnitoring of GRAvItational Lenses. VII. Time delays and the Hubble constant from WFI J2033-4723. *Astronomy & Astrophysics*, 488:481–490, Sept. 2008.

[132] C. Vuissoz, F. Courbin, D. Sluse, G. Meylan, M. Ibrahimov, I. Asfandiyarov, E. Stoops, A. Eigenbrod, L. Le Guillou, H. van Winckel, and P. Magain. COSMOGRAIL: the COSmological MOnitoring of GRAvItational Lenses. V. The time delay in SDSS J1650+4251. *Astronomy & Astrophysics*, 464:845–851, Mar. 2007.

[133] G. Watson. Smooth regression analysis. *Sankhya: The Indian Journal of Statistics, Series A*, 26(4):359–372, 1964.

[134] N. Weir, U. M. Fayyad, and S. Djorgovski. Automated star/galaxy classification for digitized poss-ii. *The Astronomical Journal*, 109:2401, 1995.

[135] N. Weir, U. M. Fayyad, S. Djorgovski, and J. Roden. The skicat system for processing and analyzing digital imaging sky surveys. *Publications of the Astronomical Society of the Pacific*, pages 1243–1254, 1995.

[136] P. Woźniak, S. Williams, W. Vestrand, and V. Gupta. Identifying red variables in the northern sky variability surveybased on observations obtained with the rotse-i robotic telescope operated at los alamos national laboratory. *The Astronomical Journal*, 128(6):2965, 2004.

[137] C.-W. Yip, A. Connolly, A. Szalay, M. SubbaRao, J. Frieman, R. Nichol, A. Hopkins, D. York, S. Okamura, J. Brinkmann, et al. Distributions of galaxy spectral types in the sloan digital sky survey. *The Astronomical Journal*, 128(2):585, 2004.

[138] E. Zitzler, K. Deb, and L. Thiele. Comparison of Multiobjective Evolutionary Algorithms: Empirical Results. *Evolutionary Computation*, 8(2):173–195, 2000.