MASS Summer School “Big Data in Astronomy”
14-18 July, Petnica Science Center (near Valjevo)
===

Monday July 14th
Arrivals – Check-in is available upon arrival

13:30 – 15:00 – *Lunch*

Conference Room
15:30 – Welcome with coffee & tea
16:00 – Milan Stojanović, *Successes of Gaia mission*
17:00 – Dragana Ilić, *Time-domain big-data surveys with Rubin Observatory*
18:00 – Anđelka Kovačević, *An example on meta-learning application on astronomical time-series*

19:30 – 20:30 – *Dinner*

Tuesday July 15th
8:30 – 9:30 – *Breakfast*

Conference Room
10:00 – 11:00 – Presentations of Data Challenges 1 and 2
11:00 – 11:30 – *Coffee/tea break*
11:30 – 12:30 – Presentations of Data Challenges 3 and 4
12:30 – 13:30 – Student work on data challenge

13:30 – 14:30 – *Lunch*

Classrooms 1, 2, 3, 4
15:00 – 16:30 – Student work on data challenge
16:30 – 17:00 – *Coffee/tea break*
17:00 – 19:00 – Student work on data challenge

19:30 – 20:30 – *Dinner*

Classrooms 1, 2, 3, 4
21:00 – 23:00 – Student work on data challenge (*optional*)

Wednesday July 16th
8:30 – 9:30 – Breakfast

Meeting Room
10:00 – 13:00 – Committee meeting (*with coffee/tea at 11h*)

Classrooms 1, 2, 3, 4
10:00 – 13:00 - Student work on data challenge (*with coffee/tea at 11h*)

13:30 – 14:30 – *Lunch*

Classrooms 1, 2, 3, 4
15:00 – 16:30 – Student work on data challenge
16:30 – 17:00 – *Coffee/tea break*
17:00 – 19:00 – Walk to the near-by Petnica cave

19:30 – 20:30 – *Dinner*

Classrooms 1, 2, 3, 4
21:00 – 23:00 – Student work on data challenge (*optional*)

Thrusday July 17th
8:30 – 9:30 – *Breakfast*

Classrooms 1, 2, 3, 4
10:00 – 13:00 – Student work on data challenge (*with coffee/tea at 11h*)

13:30 – 14:30 – *Lunch*

Conference Room
15:00 – 16:30 – Student teams presentation (Vol.1)
(*15min/team: 10min + 3min Q&A*)
16:30 – 17:00 – *Coffee/tea break*
17:00 – 18:00 – Student teams presentation (Vol.2)
(*15min/team: 10min + 3min Q&A*)

18:00 – 19:30 – Presentations of Ed2 Master thesis work (Vol.1)
(*7min: 5min + 2min Q&A*)

19:30 – 20:30 – *Dinner*

Friday July 18th
8:30 – 9:30 – *Breakfast*

Conference Room
10:00 – 11:00 - Presentations of Ed2 Master thesis work (Vol.2)
(*7min: 5min + 2min Q&A*)
11:00 – 11:30 – *Coffee/tea break*
11:30 – 12:30 – Round table: student career progress
12:30 – 13:00 – Closing

13:30 – 14:30 – *Lunch*

Departures

Data Challenges:

1. *Red noise as information source* (mentors Andjelka Kovačević, Dragana Ilić)
2. *Classifying Big Data with AI* (mentors Djordje Savić, Andjelka Kovačević, Dragana Ilić)
3. *Searching for Intruders to the Solar System (Simulating LSST Discoveries of Interstellar Objects)* (mentors Dučan Marčeta, Andjelka Kovačević)
4. *HR Diagram of 2M Gaia Stars* (mentor Milan Stojanović)

***Instructions for Data Challanges*:**
We have created randomly selected 12 teams (4 members, mixed Editions, see below). Each team should select two projects (1st and 2nd wish) with brief justification (1-2 sentence) and send us by July 9th. Then we will allocate one Data Challenge to each team. There will be 3 teams working on each Data Challenge. Please read carefully the instructions and prepare your laptops to work on the data. We will arrange access to the small cluster computer if needed for data analysis.

Teams for the data challenge by surnames:

1. *redistributed due to late min atttendance changes*
2. Alvarez Baena, Banerjee, Ahmad, Atoshi
3. Bergin, Borg, Bajor, Davis
4. Calderon Gomez, Cuceu, Dy, Gesto Herrera, Schickhoff
5. Das, Drake, Hassan, Hoskyns,
6. Glezina, Juthy, Jose, Luis Ramirez,
7. Kaushik, Khan, MAI, Niño Villegas,
8. Marcq, Mtchedlidze, Niu, Antonini
9. Pannikkote, Pisaturo, Prada-Mendez, Purves,
10. Seyffert, Shamsi, Resiga, Stratigou-Psarra,
11. Stoyanova, Suribhatla, Thang, Velásquez Múnera,
12. Shoaib, Villares Guanga, Villegas Montoya, Pranesh

****

INSTRUCTION FOR RED NOISE AS INFORMATION SOURCE

Main authors/team: SER-SAG-S1 in-kind team - Andjelka Kovacevic, Dragana Ilic, Marina Pavlovic, Aman Raju, Iva Cvorovic

It is very important that you install the packages and work through the provided examples. This will not only ensure your environment is set up correctly, but it will also help you understand how the methods work in practice. Please do not skip these steps—they are essential for understanding the upcoming materials.

Main purpose: Nonlinear, probabilistic modeling of quasar light curves to detect multiperiodic signals, reconstruct transfer functions, and infer supermassive black hole (SMBH) parameters such as mass and characteristic variability time scales. The framework is designed as an ensemble of standalone or interoperable packages (QhX, QNPy, and QNPy-Latte), which can be used independently or in combination to complement other methodologies and pipelines in time-domain astrophysics.

Other capabilities: Classification of quasar variability through latent space clustering, statistical validation of periodic detections using Intersection over Union (IoU), ensemble framework for cross-band periodicity detection, and latent representation construction using attentive neural processes and self-organizing maps. Integration with statistical robovetters for confidence estimation.

Link to code if publicly available:
QhX on PyPI: https://pypi.org/project/QhX/
Documents: https://lionandjelka.github.io/QhX1/introduction.html
Github: https://github.com/lionandjelka/QhX1
QNPy-Latte on PyPI: https://pypi.org/project/QNPy-Latte/
QNPy on PyPI (Conditional Neural Process): https://pypi.org/project/QNPy/
GitHub repository (LSST-SER-SAG-S1):https://github.com/LSST-SER-SAG-S1, https://github.com/lionandjelka/QhX1
The demo session recordings are available at [http://ser-sag.pmf.kg.ac.rs:8080/share.cgi?ssid=4fe292f9194142e39f2ec78f68ddd8ca]

Brief code description:
QhX is a Python-based package developed for periodicity detection in red noise time series data, with a focus on the Vera C. Rubin Observatory Legacy Survey of Space and Time [LSST]. Traditional methods, such as those based on the Fourier Transform, often struggle with red noise signals, which are prevalent in real-world datasets.
The core of QhX is a 2D Hybrid method-cross-correlation of wavelet transforms of light curves that operates in a “period-period” phase space, capable of detecting oscillations in one or more light curves. The QhX pipeline first transforms input data into a time-period plane via wavelets and then (auto)correlates the resulting wavelet matrices to obtain a correlation density in the period-period plane.

After integrating the correlation density, the final decision on detected periods is made by a statistical robovetter based on the significance, upper and lower errors of detected periods, and the Intersection over Union (IoU) metric for measuring the proximity and overlap of periods across bands.
Beyond compiling the numerical catalog of vetted periods, QhX offers visualization across photometric bands using QhX.

QNPy employs Self-Organizing Maps (SOMs) and Attentive Latent Neural Processes to construct context-aware latent representations of light curves. In this latent space, Multi-Dimensional Modeling (MDM) is used to infer SMBH-related parameters. QhX uses wavelet-based time-frequency transforms to detect complex periodic signals via a nonlinear correlation density map, validated statistically through IoU metrics. A modular and interoperable framework enables ensemble usage with other methodologies and robust catalog generation of periodic quasar candidates.

Related publications:
From Data to Software to Science with the Rubin Observatory LSST (arXiv:2208.02781)
Raju et al., 2025 “A Meta-Learning Framework for Multitask Reverberation Mapping in Active Galactic Nuclei”, submitted to A&A
Kovacevic et al., 2025 “QhX: A Python package for periodicity detection in red noise”, submitted to JOSS

##

Classifying Big Data with AI

Mentors: Djordje Savić, Andjelka Kovačević, Dragana Ilić

To help you prepare and follow along more effectively, please review the materials and resources outlined below.

We'll be working with astronomical data focused on object classification, specifically distinguishing between stars, galaxies, and quasars. The data we'll use is based on SDSS (Sloan Digital Sky Survey) and can be accessed here:
https://drive.google.com/file/d/1GCFDsZCWBs9yk_sBzp3423lBD_Tk4zRv/view?usp=sharing

You can also find a brief overview of the dataset and how it's used in machine learning contexts in this paper:
https://ui.adsabs.harvard.edu/abs/2023ApJ...953..138S/abstract

For hands-on exploration, please run and review the following notebook until 6 Conclusion:
https://github.com/RichardsGroup/AGN_DataChallenge/blob/main/submissions/SER-SAG/NN_SVM_RF_classification_supervised_Evaluated.ipynb
This notebook demonstrates supervised classification using support vector machines, random forests and neural networks on SDSS data.

For Python requirements, you will need pandas, numpy, scipy, scikit-learn, tensorflow, matplotlib, seaborn, jupyter, yaml. I'd suggest you make a new environment and start a jupyter notebook and start running first cell with subsequent installation of required packages prompted by errors.

Searching for Intruders to the Solar System (Simulating LSST Discoveries of Interstellar Objects)

Dušan Marčeta
dusan.marceta@matf.bg.ac.rs

1 What Will We Do in This Challenge

We will try to estimate how many Interstellar Objects (ISOs) the LSST will discover and whether it will be able to determine their sizes. To do this, we need to simulate millions of orbits and observations.

2 What are Interstellar Objects (ISOs)?

Interstellar Objects are objects that originate beyond our Solar System yet are observed traveling through it. The first two discovered interlopers, 1I/'Oumuamua and 2I/Borisov, showed strikingly different characteristics. While 1I/'Oumuamua appeared as an unresolved, asteroid-like object, 2I/Borisov displayed typical cometary behavior, emitting both gas and dust. Please refer to the thorough review of this population by Jewitt and Seligman [2023].

How ISOs Differ from Solar System Objects in Terms of Observations

ISOs move very fast. Not only can they shift position enough that we might not detect them again when the telescope revisits the same area of the sky, but they are already moving while we are observing them. During a single exposure, they can travel from one pixel to another across the CCD detector.
Hint: Look up trailing loss (Section 5.1.4. of Jones et al. [2018])

How ISOs Are Different from Other Solar System Objects

Unlike most Solar System objects, which tend to move along direct orbits close to the ecliptic, the ISO population is approximately isotropic. This means they can approach from any direction and move in all possible orientations across the sky. As a result, we expect roughly equal numbers of prograde and retrograde objects which pass the Sun in the opposite direction compared to the Solar System objects.

3 How We Simulate the LSST Observational Campaign on These Objects

What Do We Need?

1. The initial positions and orbits of the ISOs as they move through the Solar System.
2. Their photometric properties, such as size, albedo and colors.
3. A database of all telescope pointings throughout the LSST survey. This includes not only where and when the telescope will observe, but also additional parameters such as seeing conditions, the camera orientation at the time of observation, and more.

What Do We Do?

1. For each observation epoch, we propagate the orbits of all ISOs.
2. We calculate their apparent positions and check whether any of them falls within the field of view and is above the detection threshold.

Before we can simulate the observations, we need to define the properties of the ISOs, specifically their sizes and their orbits.

4 Size-Frequency Distribution (SFD)

How can we know how the sizes of ISOs are distributed when we have very limited knowledge about this population, since only two objects have been discovered to date?

The best we can do is to assume that, in this regard, they are similar to Solar System populations. The usual way to describe the size distribution of small bodies, the so-called Size-Frequency Distribution (SFD), is by a power law of the form

$$N(> D) = N_0 \left(\frac{D}{D_0}\right)^q , \tag{1}$$

where N is the number of objects per au^3 (the number density), N_0 is the reference number density for objects of diameter D_0, and q is the slope of the distribution, which indicates how rapidly the number of objects increases as we move toward smaller sizes. The slope q usually ranges from 1 to 4, based on assumptions derived from Solar System populations.

It is assumed that for ISOs, $N_0 = 0.1$ for $D_0 = 100$ m, meaning that we expect 0.1 ISO larger than 100 m in diameter per au^3.

Question: According to Equation 1, how many ISOs larger than 10 meters in diameter are there within a heliocentric sphere of radius 100 au for a moderate slope of $q = 2.5$ and for a very steep SFD with $q = 4$?

Answering this question will show that we need to simulate millions or even billions of ISOs to understand what the Vera Rubin Observatory (VRO) might observe.

5 ISO orbits

What does it mean to know the orbit of a celestial body?

It means knowing three parameters that describe its position and another three that define its velocity vector at a specific moment in time. With this information, we can calculate the body's position at any other time, whether in the future or in the past.

Instead of representing an orbit in this way, we usually use a different set of six parameters that are more practical and intuitive. These are called orbital elements, as illustrated in Fig. 1. We will explain orbital elements in detail during the summer school.

The key question is how we can know the initial positions and corresponding orbits of objects that have not yet been discovered. An even greater challenge is understanding the orbital structure of the entire population when only two such objects have been observed so far, Oumuamua and Borisov. The honest answer is that we cannot.

However, we can make reasonable assumptions about the properties of that population. Based on those assumptions, we are able to construct a synthetic population of interstellar objects that plausibly resembles the real population we have not yet observed.

We do not know what the orbits of interstellar objects look like within the Solar System. Still, since they come from other planetary systems in the nearby stellar environment and from throughout the Galaxy, it is reasonable to assume that they move in a similar way to the stars they originated from.

The basic assumption is that the distribution of heliocentric velocities of interstellar objects is similar to that of the stars from which they come.

There are two ways to generate orbits of interstellar objects in the Solar System based on the assumed velocity distributions. These are the dynamical and the probabilistic approach. Both methods are available at this GitHub repository, and are described in detail in [Marčeta, 2023].

You do not need to study the paper in detail at this stage, since its content will be presented and discussed during the summer school. For now, it is enough to visit the GitHub page, download the code, and try running the examples.

Start by running the script example.py, and generate histograms of the orbital elements. Pay particular attention to the histogram of inclination, as this angle determines whether an object approaches the Sun on a prograde or retrograde trajectory.

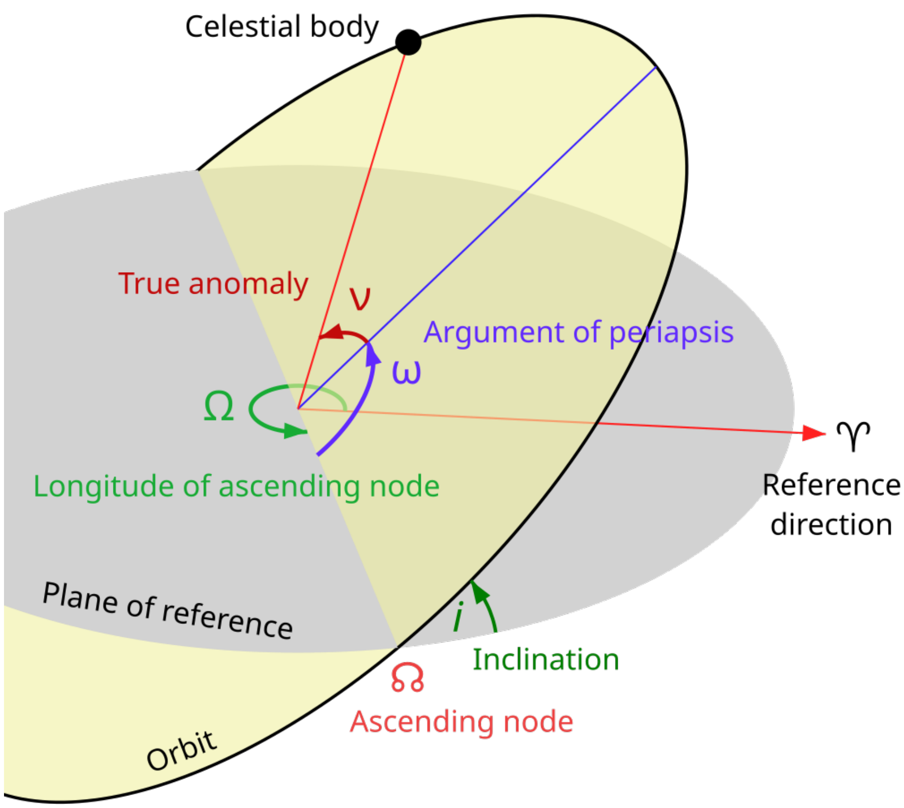

Figure 1: Orbital elements shown in this figure are *inclination,* i, *longitude of ascending node,* Ω, *argument of periapsis,* ω and *true anomaly,* ν. The remaining two not shown here are *semi-major axis,* a and *eccentricity*, e

6 SORCHA

Sorcha is a powerful open-source simulator of the LSST observations, available here.

We will go through it in detail during the summer school. For now, please try to install it and verify that your installation works properly. In any case, we will ensure it works during the summer school.

7 What We Intend to Conclude

Retrograde objects move faster relative to the telescope, which means they experience greater trailing loss, resulting in more detected prograde objects. In other words, we expect an asymmetric distribution of inclinations. Smaller objects, which are more numerous for steeper SFD slopes, have a higher chance of remaining below the detection threshold. Taking this into account, we can expect that steeper SFDs will produce a larger difference in the number of prograde and retrograde objects. Therefore, we can generate synthetic populations with different slopes and, for each, analyze the ratio of prograde to retrograde objects, attempting to relate this ratio to the SFD slope.

Prerequisites

We will be working in Python, so you should be familiar with basic programming, handling files, working with data, and using libraries like NumPy and pandas. If you have any questions, feel free to email me.

References

David Jewitt and Darryl Z. Seligman. The Interstellar Interlopers. , 61:197–236, August 2023. doi: 10.1146/annurev-astro-071221-054221.

R. Lynne Jones, Colin T. Slater, Joachim Moeyens, Lori Allen, Tim Axelrod, Kem Cook, Željko Ivezić, Mario Jurić, Jonathan Myers, and Catherine E. Petry. The Large Synoptic Survey Telescope as a Near-Earth Object discovery machine. , 303:181–202, March 2018. doi: 10.1016/j.icarus.2017.11.033.

Dušan Marčeta. Synthetic population of interstellar objects in the Solar System. *Astronomy and Computing*, 42:100690, January 2023. doi: 10.1016/j.ascom.2023.100690.

Space-based astronomy: Working with Big Databases – Gaia HR diagram

Mentor: Milan Stojanović

Space-based astronomy is important aspect of astronomical research worldwide. There are many different satellites and satellite constellations that were launched in last two decades, that proved space-based telescopes are the future of scientific endeavors in astronomical and astrophysical research.

In this exercise, we will cover one of the best astrometric space-based missions – GAIA! This telescope has just finished working and producing valuable data, and currently catalogues Data Release 3 (DR3) is valid, and all data from this mission is completely open and free for use. The aim of the exercise is to create Hertzsprung-Russell diagram (HRD) as one of the most important tools in stellar studies. The detailed HRD features are important to constrain stellar structure and evolutionary studies as well as stellar atmosphere modelling. Up to publishing Gaia catalogue, a proper understanding of the physical process in the stellar interior and the exact contribution various effects are missing because we lacked large precise and homogeneous samples that cover the full HRD. Working with big databases provide certain challenges that we will overcome using: ADQL, Python and TOPCAT all of them as tools to work with data and present results.

We will be working with Gaia DR3 catalogue which is accessible through Gaia archive website:

https://gea.esac.esa.int/archive/

https://www.cosmos.esa.int/web/gaia-users/archive/extract-data

In order to use archive with full potential, it is needed to have an account created, which is free.

The access to data is done using ADQL, so students should familiarize them with basic ADQL commands.

https://www.ivoa.net/documents/ADQL/20210528/PR-ADQL-2.1-20210528.pdf

Once ADQL query is written, it can be used directly online on the archive website, or using TOPCAT query mode (https://www.star.bristol.ac.uk/mbt/topcat/), or python Astroquery (https://pypi.org/project/astroquery/).

In a paper by Gaia collaboration (https://www.aanda.org/articles/aa/full_html/2018/08/aa32843-18/aa32843-18.html) HRD was created using DR2 and their sample was around 61 million data points, while for DR3 it is around 100 million data points, so it is necessary to have some reasonable computing CPU power and time and network bandwidth in order to complete this task. We will use the same Data filtering as in section 2 of the mentioned paper, and for the purpose of finishing the task in shorter period of time and on every laptop, students should further narrow down the sample to all stars that are within 200 parsecs range. Anyway, CPU with 12 threads (AMD Ryzen 5 or Intel Core i5) or more, and RAM memory of 16GB is recommended.

Creating Gaia HRD will be done using Python, and necessary packages are: pandas (for handling data) and matplotlib for visual presentation. Students should be able to analyze many details of Gaia HRDs like Main sequence, Brown dwarfs, Giant branch, Planetary nebulae and White dwarfs which can finally be marked on the HRD.

On Gaia archive, students can find all the Hipparcos data, which as an extra task, can also be presented as HRD and comparted with Gaia, so students should familiarize with Hipparcos database.

We hope that HRD and data shown here with some historically important astronomical relations, and some other new areas of research may spark the imagination of students to continue working on GAIA data!