Lecciones en Astroinformática Avanzada (Semester 1 2024)

Course Logistics

Nina Hernitschek
Centro de Astronomía CITEVA
Universidad de Antofagasta

May 14, 2024

Course Logistics

course content:

- lecture: Tuesday 15 - 17h (with different professors)
- project: Wednesday 14 - 16h (with Nina Hernitschek)

Course Logistics

course content:

- lecture: Tuesday 15 - 17h (with different professors)
- project: Wednesday 14 - 16h (with Nina Hernitschek)

grading:

- project presentation (45 % of total grade)
- project report (45 % of total grade)
- participation (10 % of total grade)

Project report: hand in during first week of August, can be written together with highlighting who wrote what.

Course Logistics

course content:

- lecture: Tuesday 15 - 17h (with different professors)
- project: Wednesday 14 - 16h (with Nina Hernitschek)

grading:

- project presentation (45 % of total grade)
- project report (45 % of total grade)
- participation (10 % of total grade)

Project report: hand in during first week of August, can be written together with highlighting who wrote what.

contact and course material:

- e-mail: `nina.hernitschek@uantof.cl`
- github: `https://github.com/ninahernitschek/astroinformatica_advanzada_2024_1`

Course Logistics

May 14 Automatic Classification of Variable Stars (I) (Nina Hernitschek),
May 15 Project

May 28 Automatic Classification of Variable Stars (II) (Nina Hernitschek),
May 29 Project

June 4 xxxx,
June 5 Project

June 11 Using supervised machine learning to detect satellite trails (Jeremy Tregloan-Reed, Universidad de Atacama),
June 12 Project

June 18 xxxx,
June 19 Project

June 25 xxxx,
June 26 Project

Course Logistics

July 2 Introduction to TOPCAT and STILTS (I) (Javier Alonso Garcia),
July 3 Project

July 9 Introduction to TOPCAT and STILTS (II) (Javier Alonso Garcia),
July 10 Project

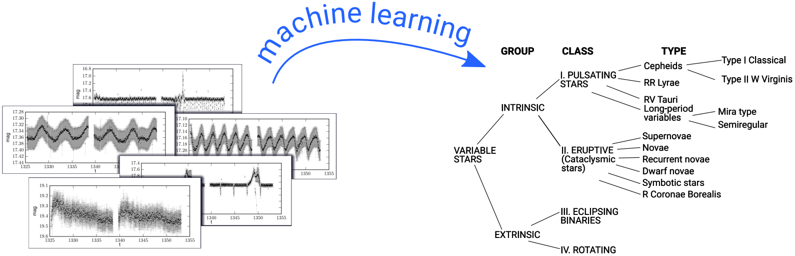
July 23 xxxx,
July 24 Project

July 30 Eclipses and Transits in the Era of Gaia and TESS (Keivan Stassun, Vanderbilt University),
July 31 Project Presentation

The Project: Variability detection and classification in astronomical datasets

What is the problem?

Nowadays large-scale astronomical surveys enable us to see the universe in a much more detail.
Historically, variable stars were identified by manually sorting images or making brightness-over-time (light curve) plots based on small surveys.
Modern all-sky astronomical surveys provide huge amounts of data.
Machine-learning tools are extremely important to evaluate such an amount of data, as well as to allow for a more precise evaluation of the data than done with classical algorithms.



GROUP
CLASS
TYPE
VARIABLE STARS
INTRINSIC
EXTRINSIC
I. PULSATING STARS
II. ERUPTIVE (Cataclysmic stars)
III. ECLIPSING BINARIES
IV. ROTATING
Cepheids
Type I Classical
Type II W Virginis
RR Lyrae
RV Tauri
Long-period variables
Mira type
Semiregular
Supernovae
Novae
Recurrent novae
Dwarf novae
Symbotic stars
R Coronae Borealis

The Project: Variability detection and classification in astronomical datasets

What will you learn in this project?

1) An overview of machine-learning techniques applicable for astronomical data (depending on preknowledge, this can serve either as an intro or as a refresher).

2) You will prepare astronomical data (which comes in the form of tables containing light curves) for processing them with machine learning techniques.

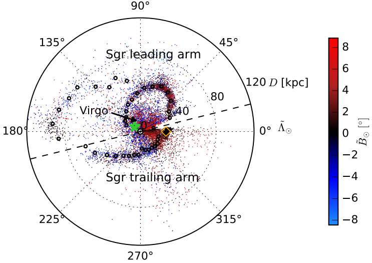
3) You will apply machine-learning techniques to astronomical data.

4) You will understand the benefits, relevance and limitations of machine learning classification on the example of the light-curve data provided for this project.

The Project: Variability detection and classification in astronomical datasets

Why is this scientifically important?

Approximately 1 - 5% of all stars show a significant amount of variability in their intensity. Variable stars are important tracers of the Milky Way's evolution. Stars such as RR Lyrae allow to precisely estimate distances in the old Milky Way components, providing important information to answer questions regarding formation and evolution of galaxies. Other stars such as spotted stars allow to gain insight into magnetic activity of stars. Rare supernova events allow to calculate distances at cosmological scales.



Sgr stream traced with RR Lyrae stars from Pan-STARRS1 3π survey.

The Project: Variability detection and classification in astronomical datasets

How do you get started?

1) Code will be written in Python3.x. Code should be made available on github.
In case you're new to Python, I can recommend taking a look at this course (but you could also take another one):
`https://www.w3schools.com/python/python_intro.asp`
Regarding github: If you haven't done so already, please sign up for github and create a repository.

2) I recommend reading Baron (2019), *Machine Learning in Astronomy: A Practical Overview* `http://research.iac.es/winterschool/2018/media/summaries/ml_summary_dbaron.pdf`

The Project: Variability detection and classification in astronomical datasets

How do you get started?

4) There is a book *Statistics, Data Mining, and Machine Learning in Astronomy: A Practical Python Guide for the Analysis of Survey Data* by Ivezic et al. In case you easily have access to this book, please use it for this project. Otherwise, a lot of example Python code is also available on the book's website `https://www.astroml.org/` for free.

5) Throughout the project, the online documentation of many Python libraries, especialls scikit-learn (sklearn) will be very helpful. `https://scikit-learn.org/stable/tutorial/basic/tutorial.html`

Rules for Coding, Presentations, Report

coding:

- use meaningful variable names
- use comments and documentation
- If you have a question when something doesn't work, summarize what you tried - often this will even lead to the solution.

Rules for Coding, Presentations, Report

coding:

- use meaningful variable names
- use comments and documentation
- If you have a question when something doesn't work, summarize what you tried - often this will even lead to the solution.

project practice reports and project presentation:

- use LaTeX
- figures: all own figures should be in vectorized pdf format
- for all data: data description (incl. citation)
- own work: properly cite what is not your own work; discuss how the previous work is similar to or different from your own work
- implementation: medium-level implementation description with libraries/ software frameworks (incl. citation)
- discussion: reflect your approach (strengths, weaknesses, limitations), lessions learned
- bibliography

Rules for Coding, Presentations, Report

Project Report Structure

Abstract

- Concise summary of your project.
- Do not include citations.

Introduction

- Give the big picture.
- Establish the scope of what you did.

Related Work

- Include both work aimed at similar problems and work that employs similar solutions to yours.
- Although there is no requirement to establish research novelty since it's a course project, you should still discuss how the previous work is similar to or different from your own work (either individually or with respect to an entire group).

Rules for Coding, Presentations, Report

Project Report Structure

Data and Project Tasks

- You should analyze your domain problem according to the project framework.
- Typically data description will need to come first.
- It is very likely that you will need to first have domain-specific descriptions.
- Write a the tasks section by first providing a domain-specific list of tasks, and then present the abstracted version.

Implementation

- Medium-level implementation description. You must include specifics of what you did yourself versus what other implementations you built upon.
- This section is one major divergence from standard research paper format, you need to provide much more detail than would normally be appropriate in a research context.

Rules for Coding, Presentations, Report

Project Report Structure

Milestones

- Include a list of project milestones, where the work is broken down into a series of smaller chunks that are meaningful and useful.
- You should also be thinking about how to break down the work into components that are appropriate for your project in specific.
- Milestones should include four numbers: estimated number of hours to carry out each task, actual number hours it took, estimated date of completion, actual date of completion, contribution from each group member.

Results

- Should include the scenarios explored according to the project framework.
- Walk the reader through how your results are (or not) solving the intended problem.
- Make sure to save all Figures as vector `*.pdf`.

Rules for Coding, Presentations, Report

Project Report Structure

Discussion and Future Work

- Strengths, weaknesses, limitations (reflect on your approach).
- What you have learned from the project (working with the data and algorithms).
- Future work (what would you do if you had more time?).

Conclusions

- Summarize what you've done in a way that's different from the abstract because you can count on the reader having now seen all of the content of the paper in between.

Bibliography

- Check if arXiv papers have been later published in journals.
- Always use the LaTeX `label/ref` mechanism for figure and section cross references.
- Always use the `bibtex/ref` mechanism and a `*.bib` file rather than hardwired citations in the text.

Textbooks

I can recommmend the following textbooks and online resources:

Python cookbook; Beazley, D. & Jones, B.; O'Reilly Media.

Statistics, Data Mining, and Machine Learning in Astronomy: A Practical Python Guide for the Analysis of Survey Data; Ivezic et al.