UNIVERSITÀ
DEGLI STUDI DI MILANO-BICOCCA

SYLLABUS DEL CORSO

Astrostatistics and Machine Learning

2627-1-F5803Q020

Aims

The use of statistics is ubiquitous in astronomy and astrophysics. Modern advances are made possible by the application of increasingly sophisticated tools, often dubbed as "data mining", "machine learning", and "artificial intelligence". This class provides an introduction to (some of) these statistical techniques in a very practical fashion, pairing formal derivations to hands-on computational applications. Although examples will be taken almost exclusively from the realm of astronomy, this class is appropriate to all Physics students interested in machine learning.

Students will develop skills in statistical inference, data analysis, and advanced computing, with hands-on experience applying machine learning techniques to (astro)physical data.

Contents

- Probability and statistics.
- Frequentist and Bayesian inference.
- Machine learning.

Detailed program

1. Introduction

- Data mining and machine learning.
- Supervised and unsupervised learning.
- Python setup.

- Version control with git.

2. Probability and Statistics

- Probability.
- Bayes' theorem.
- Random variables.
- Monte Carlo integration.
- Descriptive statistics.
- Common distributions.
- Central limit theorem.
- Multivariate pdfs.
- Correlation coefficients.
- Sampling from arbitrary pdfs.

3. Frequentist Statistical Inference

- Frequentist vs Bayesian inference.
- Maximum likelihood estimation.
- Omoscedastic Gaussian data, Heteroscedastic Gaussian data, non Gaussian data.
- Maximum likelihood fit.
- Role of outliers.
- Goodness of fit.
- Model comparison.
- Gaussian mixtures.
- Boostrap and jackknife.
- Hypotesis testing.
- Comparing distributions, KS test.

4. Bayesian Statistical Inference

- The Bayesian approach to statistics.
- Prior distributions.
- Credible regions.
- Parameter estimation examples
- Marginalization.
- Model comparison: odds ratio.
- Approximate model comparison.
- Monte Carlo methods.
- Markov chains.
- Burn-in.
- Metropolis-Hastings algorithm.
- MCMC diagnostics: burn-in, autocorrelation lenght, traceplots.
- Samplers in practice: emcee and PyMC3.
- Gibbs sampling.
- Conjugate priors.
- Evidence evaluation.
- Nested sampling.
- Samplers in practice: dynesty.

5. Clustering.

- K-fold cross validation.
- K-Means Clustering.
- Mean-shift Clustering.

- Correlation functions.

6. Dimensional Reduction

- Curse of dimensionality.
- Principal component analysis.
- Non-negative matrix factorization.
- Independent component analysis.
- Non-linear dimensional reduction.
- Locally linear embedding.
- Isometric mapping.
- t-distributed stochastic neighbor embedding.

7. Density estimation

- Histograms
- Kernel density estimation
- Nearest-Neighbor.
- Gaussian Mixtures.

8. Regression

- Linear regression.
- Polynomial regression.
- Basis function regression.
- Kernel regression.
- Over/under fitting.
- Cross validation.
- Learning curves.
- Regularization: Ridge, LASSO.
- Non-linear regression.
- Gaussian process regression.
- Total least squares.

9. Classification

- Generative vs discriminative classification.
- Receiver Operating Characteristic (ROC) curve.
- (Gaussian) Naive Bayes.
- Linear and quadratic discriminant analysis.
- GMM Bayes classification.
- K-nearest neighbor classifier.
- Logistic regression.
- Support vector machines.
- Decision trees.
- Bagging.
- Random forests.
- Boosting.

10. Deep learning

- Loss functions.
- Gradient descent, learning rate.
- Adaptive boosting.
- Neural networks.

- Backpropagation.
- Layers, neurons, activation functions, regularization schemes.
- Neural network in practice: TensorFlow, keras, and pytorch.
- Convolutional neural networks.
- Autoencoders.
- Generative adversarial networks.
- Transformers.
- Tokens.
- Self-attention.
- GPTs.
- Large language models.

Examples of astrophysical datasets we will be using:

- Galaxies and quasars from the Sloan Digital Sky Survey.
- Black-hole binaries from the Laser Interferometer Gravitational-Wave observatory (LIGO).
- Simulated supernova observations
- Gamma Ray bursts
- and more...

Prerequisites

No formal prerequisites. Some previous knowledge of the python programming language and familiarity with the Unix shell are **highly recommended** (see below for some catch-up resources).

Teaching form

Lessons, 6 credits.
Form. About 50% of the class will consist of formal lecture and 50% of interactive demonstrations.

Data mining and machine learning are computational subjects. One does not understand how to treat scientific data by reading equations on the blackboard: you will need to get your hands dirty (and this is the fun part!). Students are required to come to classes with a laptop or any device where you can code on (larger than a smartphone I would say...). Each class will pair theoretical explanations to hands-on exercises and demonstrations. These are the key content of the course, so please engage with them as much a possible.

Textbook and teaching resource

The **main textbook** we will be using is:

"Statistics, Data Mining, and Machine Learning in Astronomy", Željko, Andrew, Jacob, and Gray. Princeton University Press, 2012.

It's a wonderful book that I keep on referring to in my research. The library has a few copies; you can also download a digital version from the Bicocca library website. What I really like about that book is that they provide the code behind each single figure: astroml.org/book_figures. The best way to approach these topics is to study the

introduction on the book, then grab the code and try to play with it. Make sure you get the updated edition of the book (that's the one with a black cover, not orange) because all the examples have been updated to python 3.

There are many **other good resources** in astrostatistics, here is a partial list. Some of them are free.

- "Statistical Data Analysis", Cowan. Oxford Science Publications, 1997.
- "Data Analysis: A Bayesian Tutorial", Sivia and Skilling. Oxford Science Publications, 2006.
- "Bayesian Data Analysis", Gelman, Carlin, Stern, Dunson, Vehtari, and Rubin. Chapman & Hall, 2013. Free!
- "Probability Theory: The Logic of Science. Jaynes. Cambridge University Press, 201
- "Python Data Science Handbook", VanderPlas. O'Reilly Media, 2016. Free!
- "Practical Statistics for Astronomers", Wall and Jenkins. Cambridge University Press, 2003.
- "Bayesian Logical Data Analysis for the Physical Sciences", Gregory. Cambridge University Press, 2005.
- "Modern Statistical Methods For Astronomy" Feigelson and Babu. Cambridge University Press, 2012.
- "Information theory, inference, and learning algorithms" MacKay. Cambridge University Press, 2003. Free!
- Bayesian inference: More than Bayes's theorem Loredo and Wolpert, 2024 [arXiv:2005.14199]
- "Practical Guidance for Bayesian Inference in Astronomy", Eadie et al., 2023 [arXiv:2302.04703]
- "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow", Geron, O'Reilly Media, 2019.
- "Machine Learning for Physics and Astronomy", Acquaviva, Princeton University Press, 2023.
- "A review of Unsupervised Learning in Astronomy", S. Fotopoulou et al., 2024.
- "Statistical Machine Learning for Astronomy, Y.-S. Ting, 2025
- “Data analysis recipes". These free are chapters of a book that is not yet finished by Hogg et al.
    - "Choosing the binning for a histogram" [arXiv:0807.4820]
    - "Fitting a model to data [arXiv:1008.4686]
    - "Probability calculus for inference" [arXiv:1205.4446]
    - "Using Markov Chain Monte Carlo" [arXiv:1710.06068]
    - "Products of multivariate Gaussians in Bayesian inferences" [arXiv:2005.14199]

We will make heavy usage of the python programming language. If you need to refresh your **python skills**, here are some catch-up resources and online tutorials. A strong python programming background is essential in modern astrophysics!

- "Scientific Computing with Python", D. Gerosa. This is a class I teach for the PhD School here at Milano-Bicocca.
- "Lectures on scientific computing with Python", R. Johansson et al.
- Python Programming for Scientists", T. Robitaille et al.
- "Learning Scientific Programming with Python", Hill, Cambridge University Press, 2020. Supporting code: scipython.com.
- "Effective Computation in Physics", A. Scopatz, K. D. Huff, O'Reilly Media, 2015.
- "Python for non pythonians", Rubera, Grossetti, 2019.

Semester

Second semester.

Assessment method

The class will be assessed with an oral exam. A set of computational problems will be assigned during the course. Students will need to complete it in their own time and discuss them during the exam, togheter with broader

questions on the taught material.

The exam will evaluate:

- ability to extract statistical information from the provided datasets;
- knowledge and overall familiarity with the topics treated during lectures;
- creativity and proficiency at tackling conceptual and computational problems related to the techniques covered during lectures.

All classes, excercises, and exams will be in English.

Office hours

Any time, please contact me by email. My office is number 2007 in the U2 building.

Sustainable Development Goals

QUALITY EDUCATION | INDUSTRY, INNOVATION AND INFRASTRUCTURE