Advanced Astroinformatics - Student Project

Machine Learning: Intro to Scikit-Learn

Dr. Nina Hernitschek
June 27, 2022

Data Mining

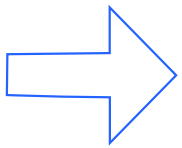
shifting use cases:

As data become more plentiful, finding the *right* data has become more important.

Data Mining

shifting use cases:

As data become more plentiful, finding the *right* data has become more important.

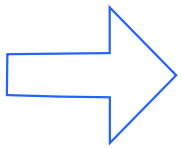


Data Mining

Data Mining

shifting use cases:

As data become more plentiful, finding the *right* data has become more important.



Data Mining

Individual measurements giving way to **statistics, clustering, patterns** in the data.
Data processing needs to be **highly automatized**.
Analysis growing more exploratory rather than pre-defined/ scripted.

Data Mining

Examples:

Finding and classifying variable stars in PS1 3π required processing of 10^9 sparse light curves $\Rightarrow$ 44,000 RRab stars.

Transient science (gravitational wave follow-up, GRBs, unknowns from LSST) requires rapid access to data sets of what is already known, anywhere on sky.

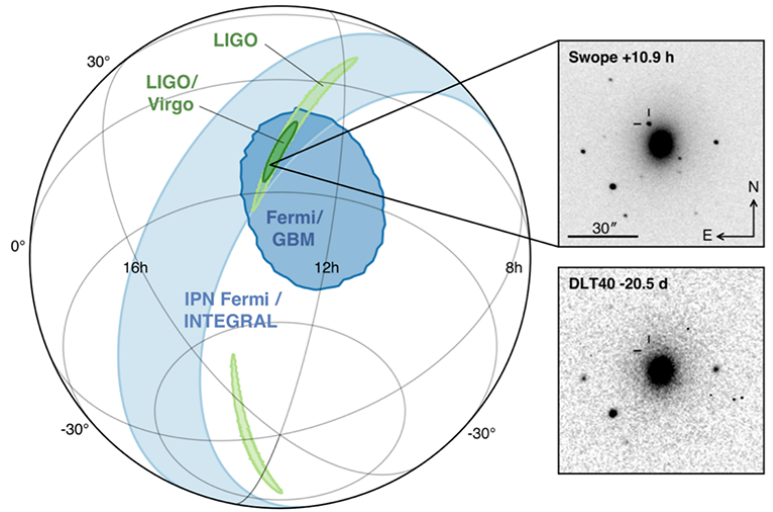


image credit: LIGO

Machine Learning

... is the sub-field of computer science that gives computers the ability to learn without being explicitly programmed
(Arthur Samuel, 1959)

Machine Learning

... is the sub-field of computer science that gives computers the ability to learn without being explicitly programmed (Arthur Samuel, 1959)

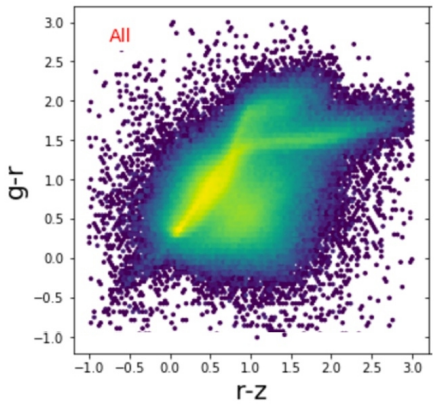
$\Rightarrow$ allows to **uncover hidden correlation patterns** through iterative learning by sample data

Machine Learning

... is the sub-field of computer science that gives computers the ability to learn without being explicitly programmed (Arthur Samuel, 1959)

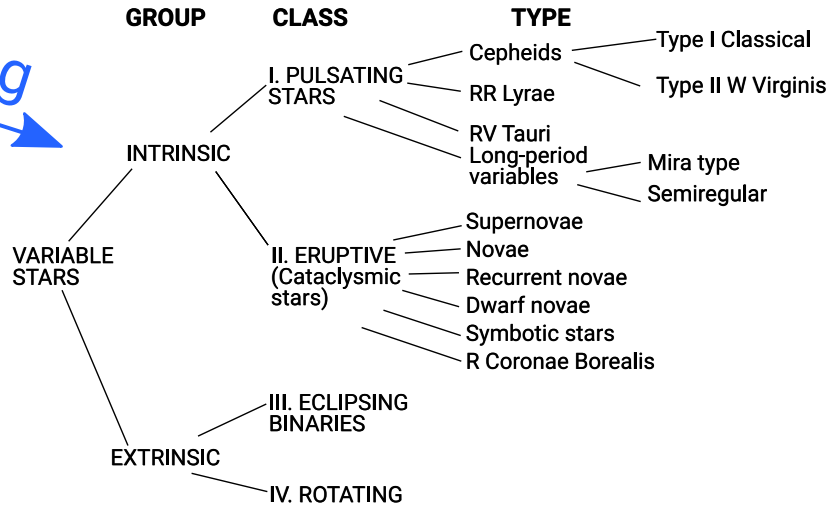
$\Rightarrow$ allows to **uncover hidden correlation patterns** through iterative learning by sample data

parameter space of measurements



machine learning

parameter space of astrophysical objects



Machine Learning

... is the sub-field of computer science that gives computers the ability to learn without being explicitly programmed (Arthur Samuel, 1959)

$\Rightarrow$ allows to **uncover hidden correlation patterns** through iterative learning by sample data

$\Rightarrow$ allows **to model a survey**:

- describing data quality $\rightarrow$ outlier
- describing light curve characteristics $\rightarrow$ "features"
- classifying sources $\rightarrow$ catalogs
- finding substructure $\rightarrow$ clumps, overdensities, ...

Unsupervised vs. Supervised Learning

unsupervised learning or "learning without labels"

Clustering:
Find subtypes or groups that are not defined a priori based on measurements
$\Rightarrow$ members of the same cluster are "close" in some sense

vs.

supervised learning or "learning with labels"

Classification:
Use a priori group labels in analysis to assign new observations to a particular group or class

Regression:
instead of having training data with discrete labels, the "truth" is a continuous property and we are trying to predict the values of that property for the test data

Unsupervised vs. Supervised Learning

supervised learning or "learning without labels"

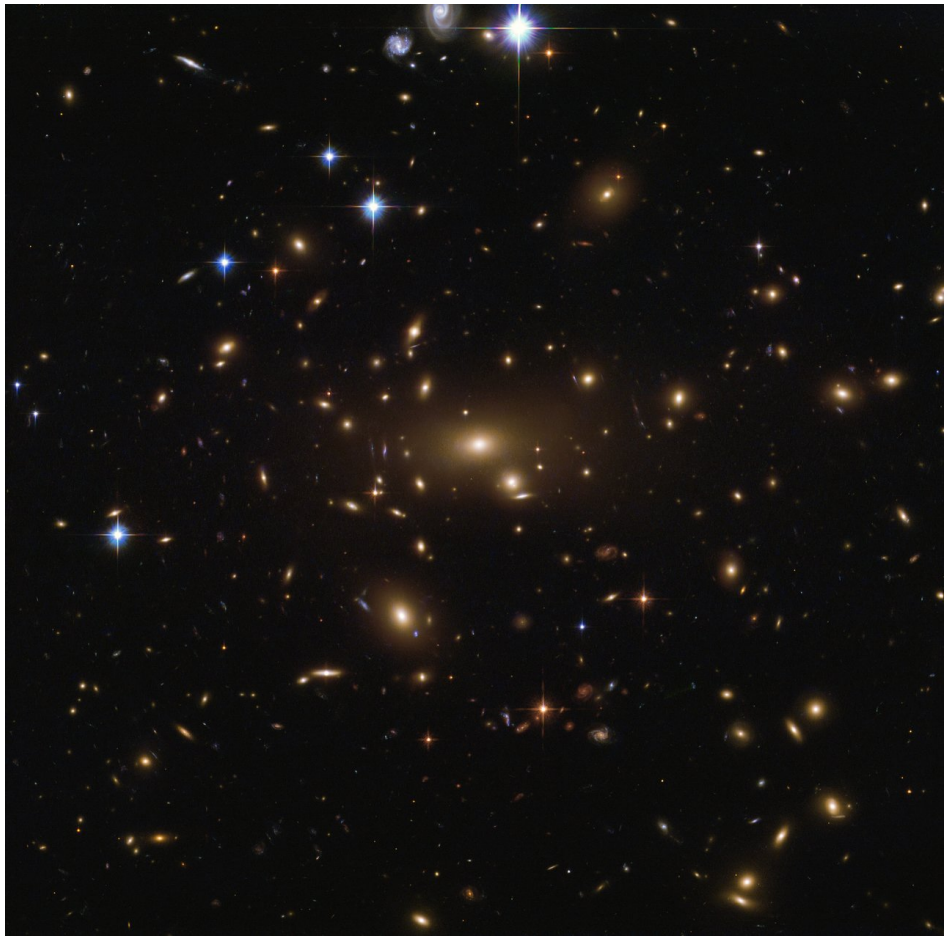
Classification:
Use a priori group labels in analysis to assign new observations to a particular group or class

Regression:
instead of having training data with discrete labels, the "truth" is a continuous property and we are trying to predict the values of that property for the test data

example:
The task of determining whether an object is a star, a galaxy, or a quasar is a classification problem: the label is from three distinct categories. On the other hand, we might wish to estimate the age of an object based on such observations: this would be a regression problem, because the label (age) is a continuous quantity.

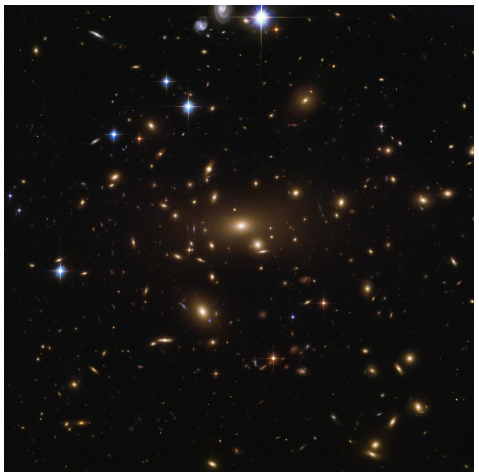
Clustering Methods

- Abell clustering richness class (Abell 1958)



Clustering Methods

- Abell clustering richness class (Abell 1958)



- Gamma Ray Bursts: use properties of GRBs (e.g. location on the sky, arrival time, duration) to find classes of events

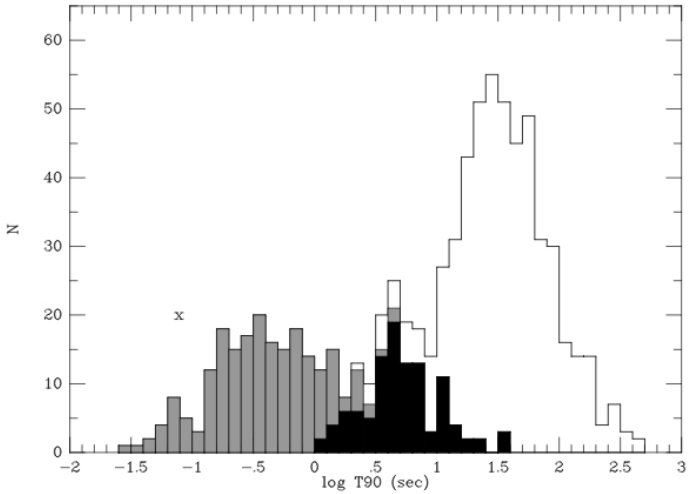


image credit: (Mukherjee+1998)

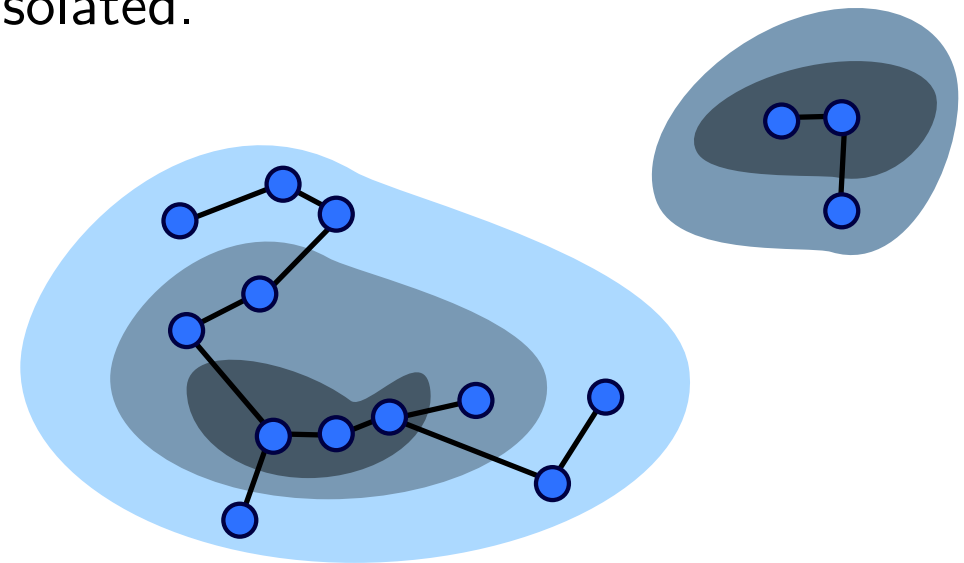
Clustering Methods

Percolation or 'Friends of Friends (FoF)' algorithm

1. Plot data points in a 2-dimensional diagram (or: calculate distances using a metric).

2. Find the closest pair, and call the merged object a *cluster*.

3. Repeat step 2 until some chosen threshold is reached. Some objects will lie in rich clusters, others have one companion, and others are isolated.

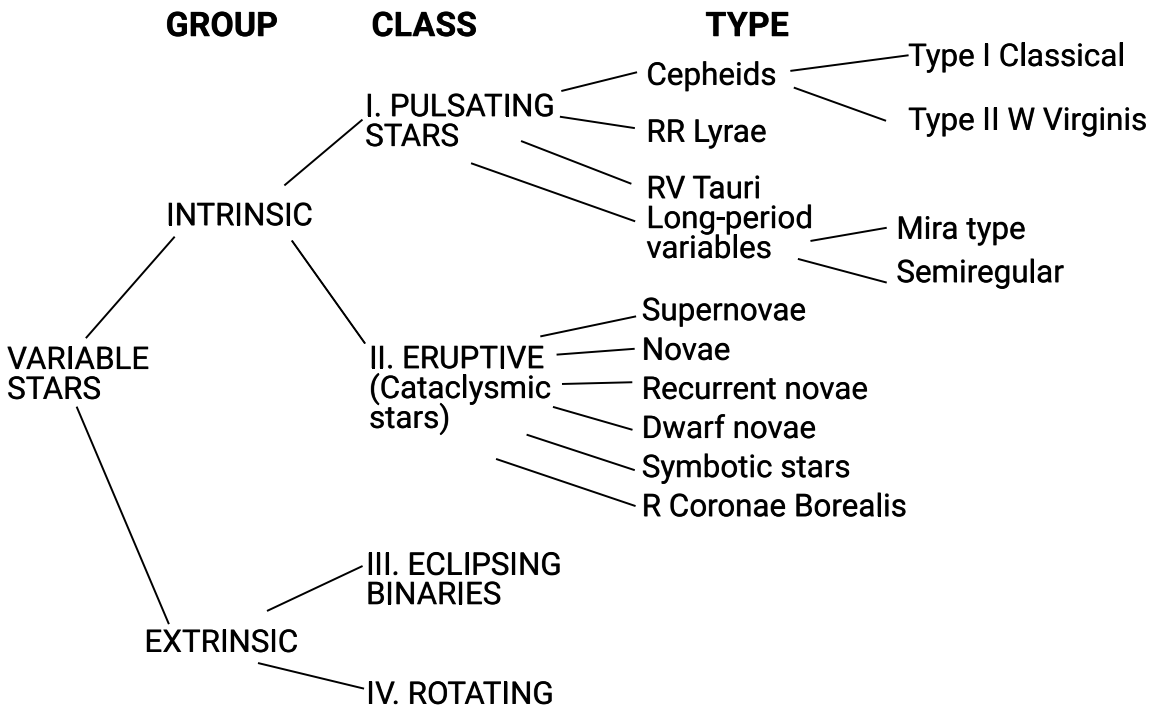


Classification Methods

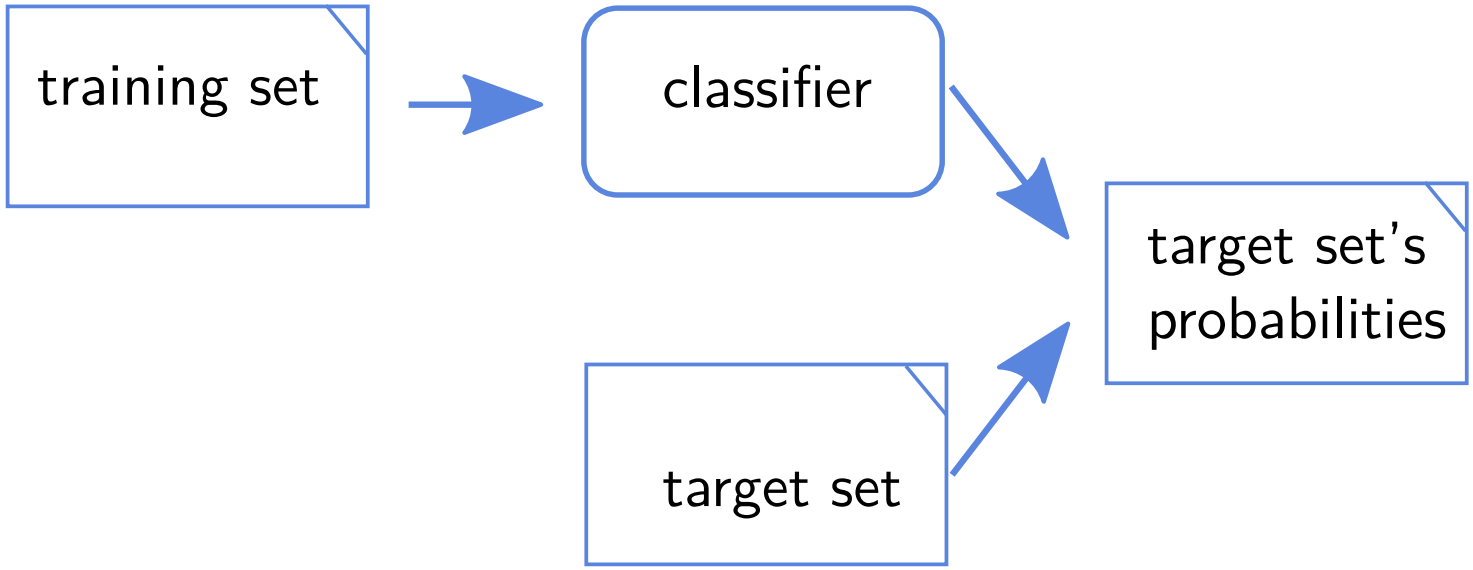
Classification

Use a priori group labels in analysis to assign new observations to a particular known group or class.

$\Rightarrow$ *supervised learning* or "learning with labels".



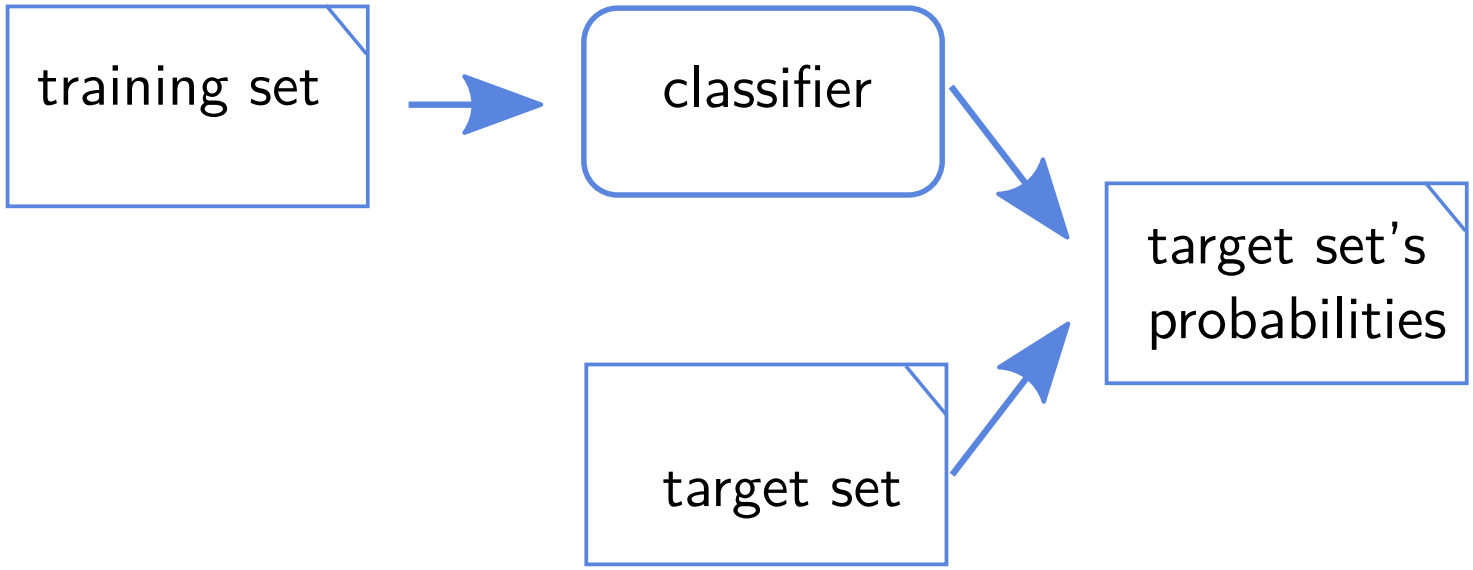
Concepts of Supervised Classification



training set:

- set of sources inside/outside the category we are looking for
- same data quality as found in target set

Concepts of Supervised Classification



training set:

- set of sources inside/outside the category we are looking for
- same data quality as found in target set

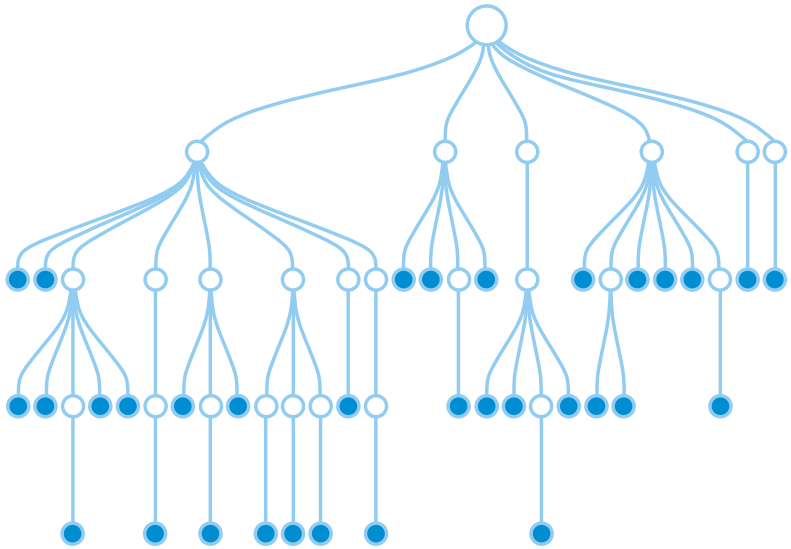
What's happening internally?

Concepts of Supervised Classification

The learning process ("training"):

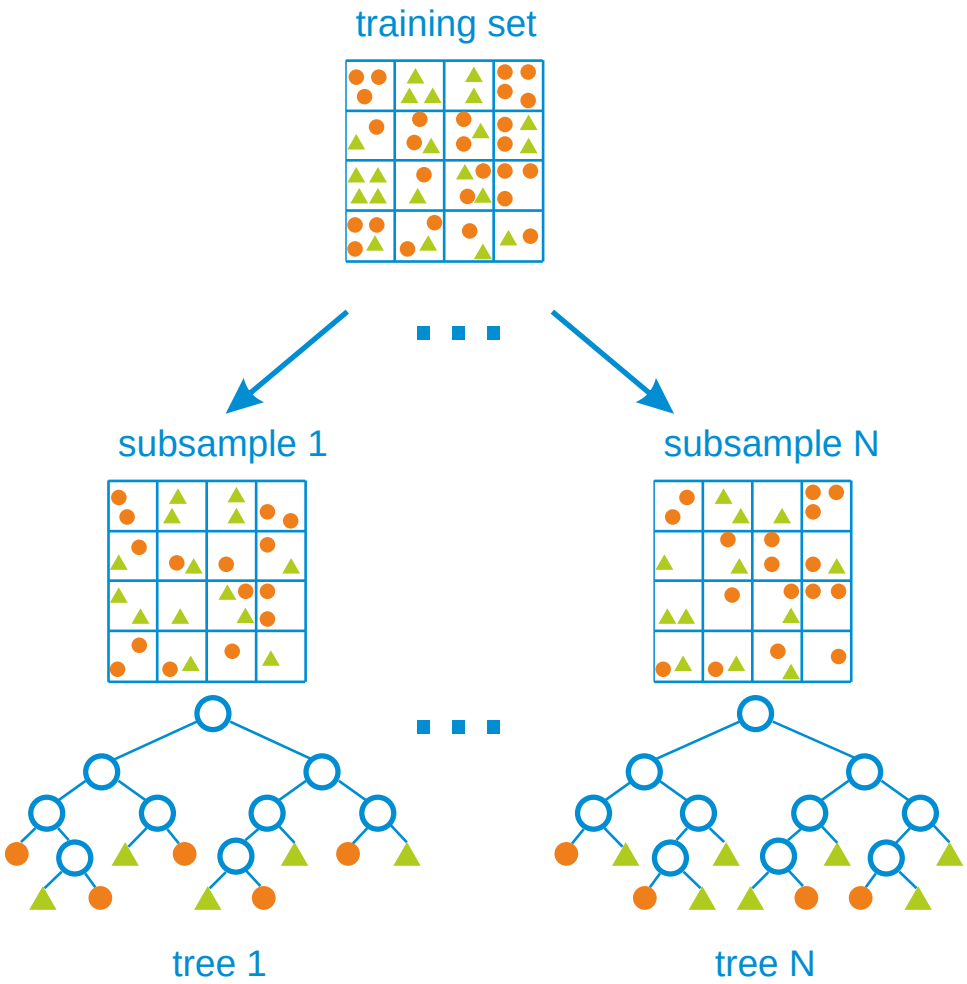
To build a decision tree, the set is divided into smaller and smaller subsets by **splitting** w.r.t. a single **feature** at a time.

Split criteria: select feature and split point to produce the smallest impurity in the two resultant nodes based on the **training set**.



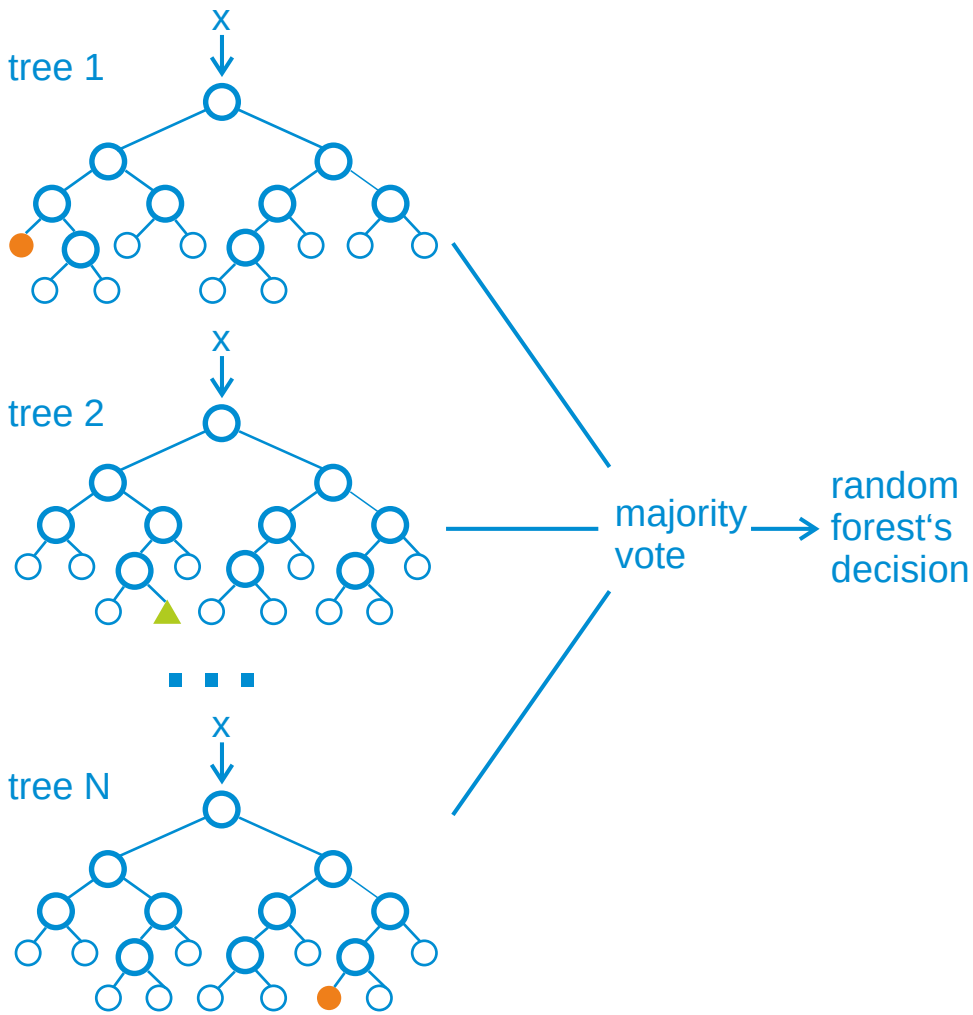
Supervised Classification - Ensemble Methods

Random Forest Classifier as ensemble method: many trees are grown from subsets of the training set

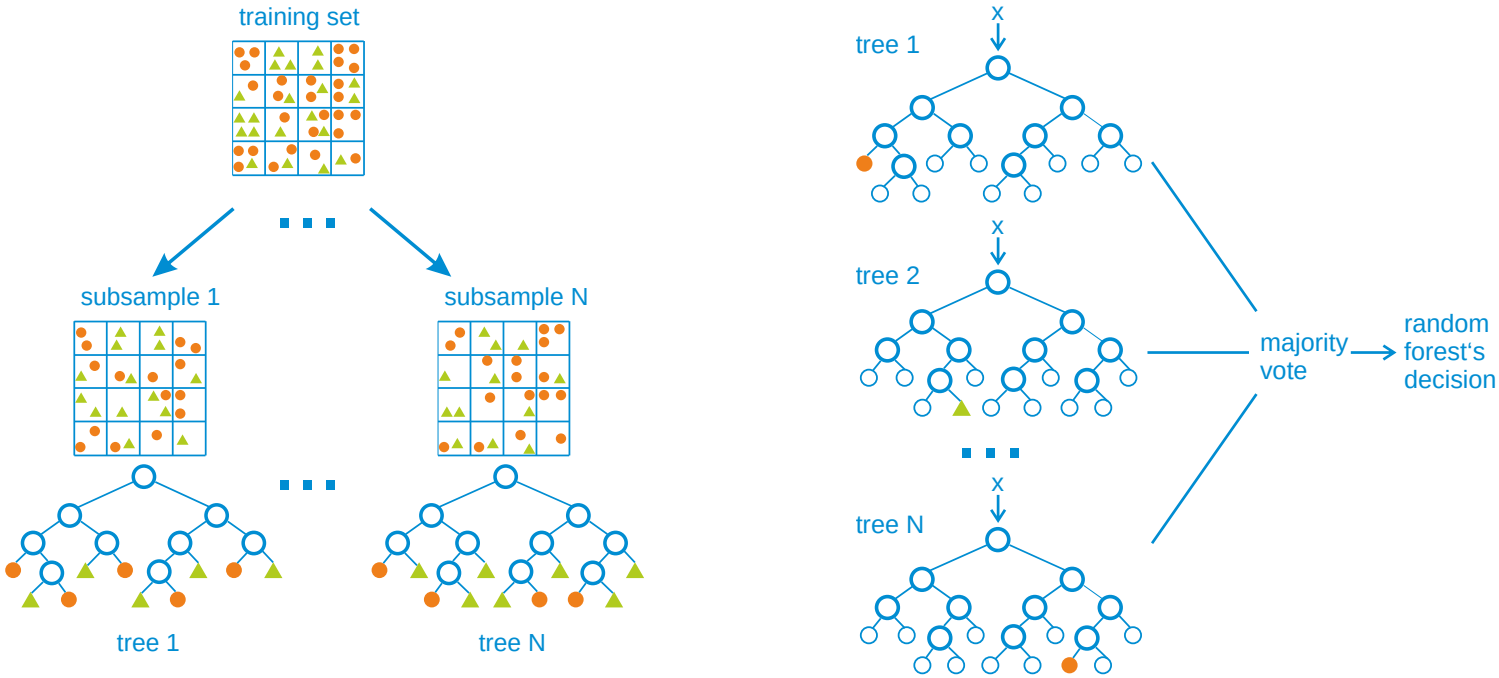


Supervised Classification - Ensemble Methods

Random Forest Classifier as ensemble method: ... and are "voting" for classification



Supervised Classification - Ensemble Methods



divide-and-conquer approach improves classification performance

- less sensitive to training set variances
- robust to outliers
- training and classification can be parallelized

Verification

never apply machine learning as a black box!

various **verification techniques** can be applied by splitting the modeling set into training and validation set

Verification

never apply machine learning as a black box!

various **verification techniques** can be applied by splitting the modeling set into training and validation set

For simplicity, we consider here binary classification where each observation is assigned to either class 1 or 0 (= not 1).
In that case, there are the following outcomes (if you want identify class 1):

- True Positive = correctly identified (class 1 identified as class 1)
- True Negative = correctly rejected (class 0 rejected as class 0)
- False Positive = incorrectly identified (class 0 identified as class 1)
- False Negative = incorrectly rejected (class 1 rejected as class 0)

Verification

Based on these, we define either of the following pairs of terms:

$$\text{completeness} = \frac{\text{true positives}}{\text{true positives} + \text{false negatives}}$$

$$\text{contamination} = \frac{\text{false positives}}{\text{true positives} + \text{false positives}} = \text{false discovery rate}$$

Instead of contamination, often also efficiency (also called purity) is used:
$\text{efficiency} = (1 - \text{contamination})$

or

$$\text{true positive rate} = \frac{\text{true positives}}{\text{true positives} + \text{false negatives}}$$

$$\text{false positive rate} = \frac{\text{false positives}}{\text{true negatives} + \text{false positives}}$$

Similarly

$$\text{efficiency} = 1 - \text{contamination} = \text{precision}.$$

Verification

To illustrate the differences between these measures, let's look at the following **example:**

We have a modeling set containing 100,000 stars and 1000 quasars. If you correctly identify 900 quasars and mistake 1000 stars for quasars, we have:

Verification

To illustrate the differences between these measures, let's look at the following **example:**

We have a modeling set containing 100,000 stars and 1000 quasars. If you correctly identify 900 quasars and mistake 1000 stars for quasars, we have:

- TP = 900 (true positive)
- FN = 100 (false negative)
- TN = 99,000 (true negative)
- FP = 1000 (false positive)

Which gives

$$\text{true positive rate} = \frac{900}{900 + 100} = 0.9 = \text{completeness}$$

$$\text{false positive rate} = \frac{1000}{99000 + 1000} = 0.01$$

Verification

To illustrate the differences between these measures, let's look at the following **example:**

We have a modeling set containing 100,000 stars and 1000 quasars. If you correctly identify 900 quasars and mistake 1000 stars for quasars, we have:

- TP = 900 (true positive)
- FN = 100 (false negative)
- TN = 99,000 (true negative)
- FP = 1000 (false positive)

Which gives

$$\text{true positive rate} = \frac{900}{900 + 100} = 0.9 = \text{completeness}$$

$$\text{false positive rate} = \frac{1000}{99000 + 1000} = 0.01$$

question:
What do you think about these results?

Verification

To illustrate the differences between these measures, let's look at the following **example:**

We have a modeling set containing 100,000 stars and 1000 quasars. If you correctly identify 900 quasars and mistake 1000 stars for quasars, we have:

- TP = 900 (true positive)
- FN = 100 (false negative)
- TN = 99,000 (true negative)
- FP = 1000 (false positive)

Which gives

$$\text{true positive rate} = \frac{900}{900 + 100} = 0.9 = \text{completeness}$$

$$\text{false positive rate} = \frac{1000}{99000 + 1000} = 0.01$$

answer:
Despite the FPR doesn't look bad, there are a lot of stars, so the contamination rate isn't good: $\text{contamination} = \frac{1000}{900+1000} = 0.53$

Verification

To illustrate the differences between these measures, let's look at the following **example:**

We have a modeling set containing 100,000 stars and 1000 quasars. If you correctly identify 900 quasars and mistake 1000 stars for quasars, we have:

- TP = 900 (true positive)
- FN = 100 (false negative)
- TN = 99,000 (true negative)
- FP = 1000 (false positive)

Which gives

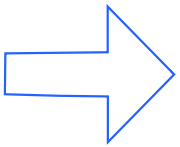
$$\text{true positive rate} = \frac{900}{900 + 100} = 0.9 = \text{completeness}$$

$$\text{false positive rate} = \frac{1000}{99000 + 1000} = 0.01$$

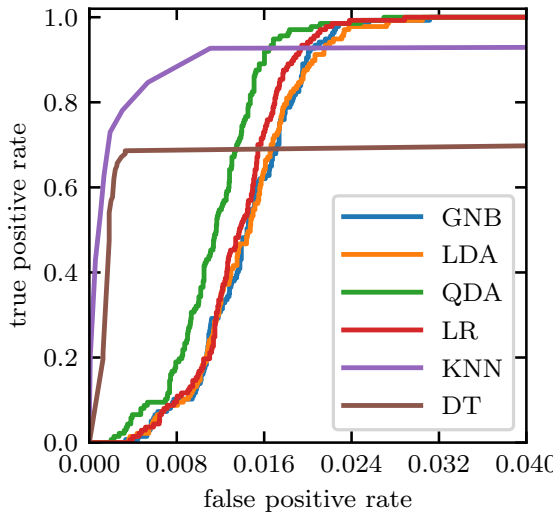
however:
The classifier might be sufficient as one step in a classification pipeline.

Classifier Performance

tradeoff: contamination versus completeness

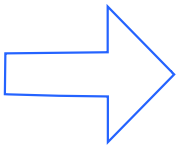


quantify this with a Receiver Operating Characteristic (ROC) curve which plots the true-positive vs. the false-positive rate

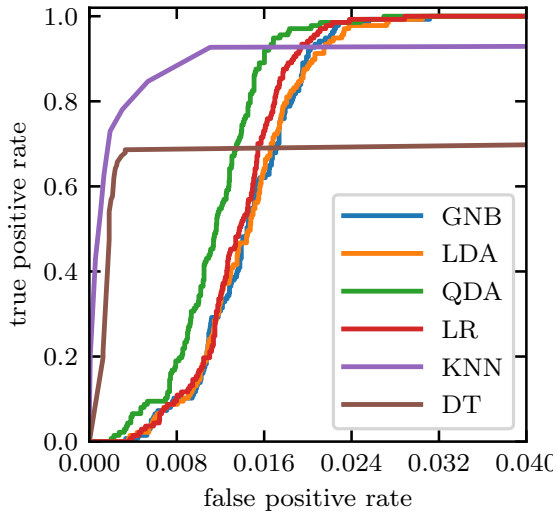


Classifier Performance

tradeoff: contamination versus completeness



quantify this with a Receiver Operating Characteristic (ROC) curve which plots the true-positive vs. the false-positive rate

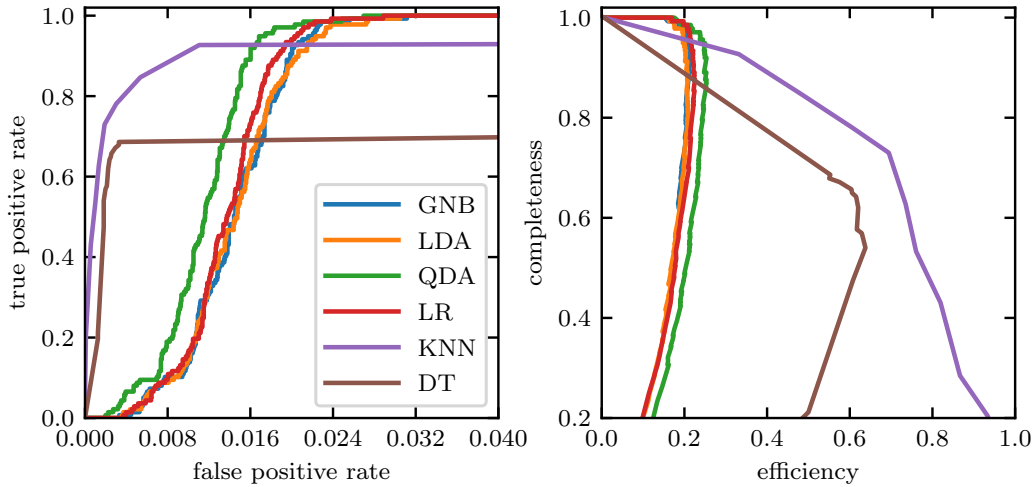


One concern about ROC curves is that they are sensitive to the relative sample sizes: if there are many more background events than source events, small false positive results can dominate a signal.

For these cases we can plot completeness versus efficiency.

Classifier Performance

Here is a comparison of the two types of plots:



Here we see that to get higher completeness, you could actually suffer significantly in terms of efficiency, but your FPR might not go up that much if there are lots of true negatives.

Note that the desired completeness and efficiency is chosen by selecting a decision boundary. The curves show what these possible choices are. Generally, one wants to chose a decision boundary that maximizes the area under the ROC (or completeness versus efficiency) curve.

scikit-learn

scikit-learn is a popular Python package containing a collection of tools for **machine learning**

it includes algorithms used for classification, regression and clustering

it comes with an extensive **online documentation**: http://scikit-learn.org/stable/tutorial/basic/tutorial.html

scikit-learn is built upon Python's NumPy (Numerical Python) and SciPy (Scientific Python) libraries, which enable efficient in-core numerical and scientific computation within Python.

scikit-learn

scikit-learn uses 3 steps for **developing, applying and testing** machine learning algorithms:

- Train the model using an existing data set describing the phenomena you need the model to predict.
- Test the model on another existing data set to ensure it performs well.
- Use the model to predict phenomena as needed for your project.

Break & Questions

afterwards we continue with **`notebook_5.ipynb`** from the `github` repository