Applied Machine Learning
Lecture 2: Ensembles and random forests

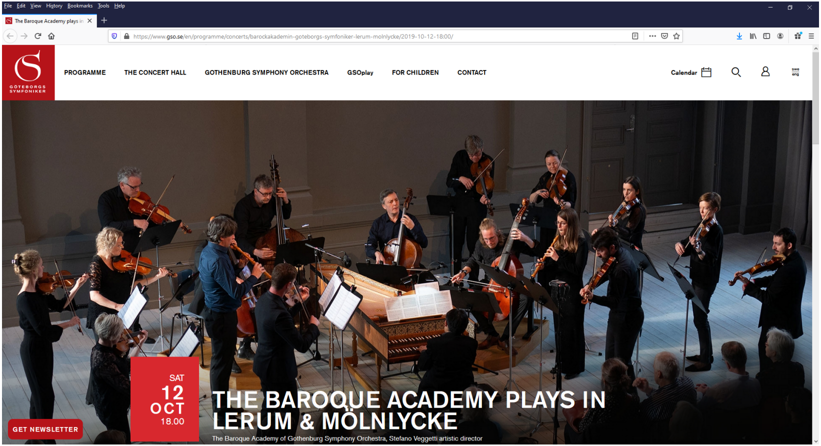
Selpi (selpi@chalmers.se)

The slides are further development of Richard Johansson's slides

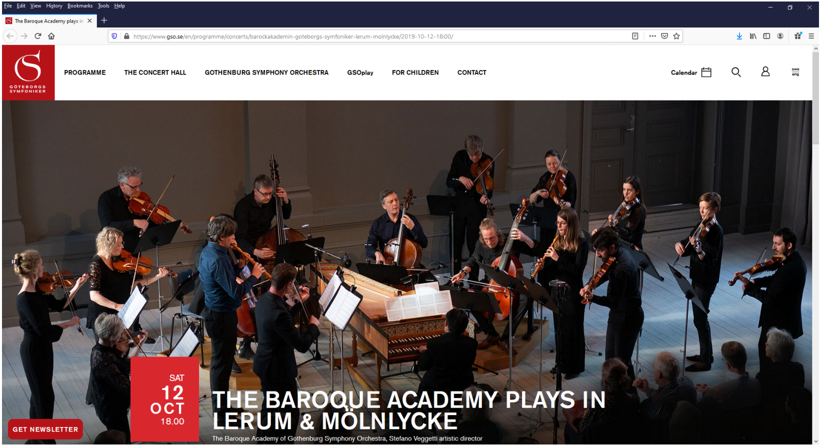
January 24, 2020

Overview

Ensembles

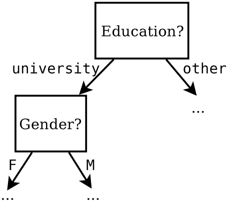

File Edit View History Bookmarks Tools Help
The Baroque Academy plays in
https://www.gso.se/en/programme/concerts/barockakademin-goteborgs-symfoniker-lerum-molnlycke/2019-10-12-18:00/
GÖTEBORGS SYMFONIKER
PROGRAMME
THE CONCERT HALL
GOTHENBURG SYMPHONY ORCHESTRA
GSOplay
FOR CHILDREN
CONTACT
Calendar
SAT
12
OCT
18.00
THE BAROQUE ACADEMY PLAYS IN LERUM & MÖLNLYCKE
The Baroque Academy of Gothenburg Symphony Orchestra, Stefano Veggetti artistic director
GET NEWSLETTER

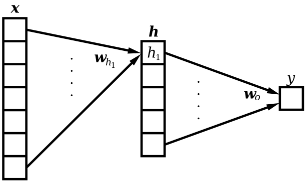
Ensembles

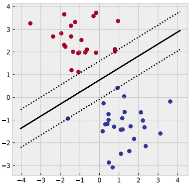


- In machine learning: combination of several models
  - of different types (e.g., decision tree & neural network)
  - of the same type but trained differently (e.g., random forests)

Why use ensembles?



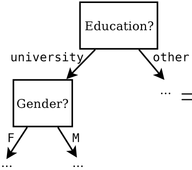




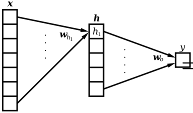
- To create a more powerful model by combining the strengths of individual/base models.
- To reduce overfitting
- Diverse classifiers can complement each other, average out individual mistakes

Overview

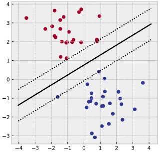
Ensemble technique: Voting



$\Rightarrow$ +1



$\Rightarrow$ -1



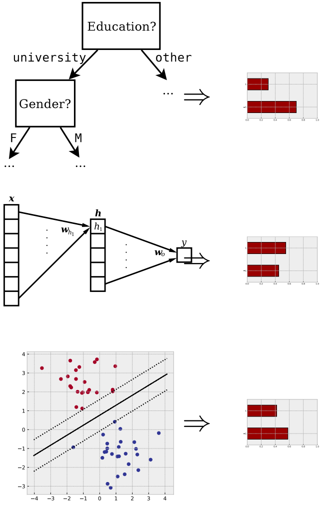
$\Rightarrow$ +1

- Majority
- Weighted

in scikit-learn

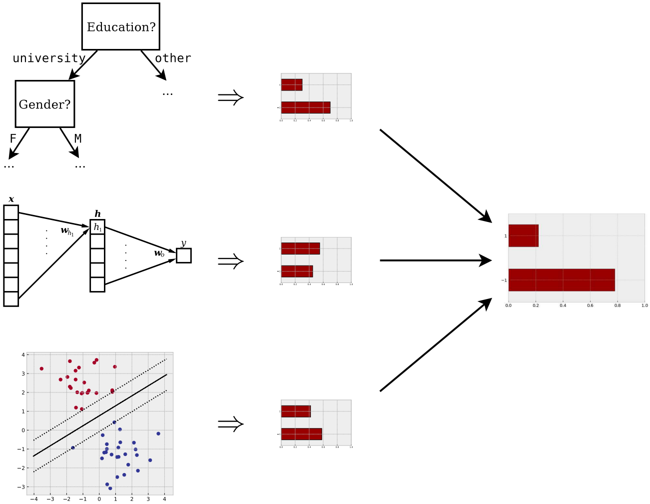
```
ensemble = [
            ('lr', LogisticRegression()),
            ('dt', DecisionTreeClassifier(max_depth=5)),
            ('svc', LinearSVC()),
            ('lr1', LogisticRegression(penalty='l1')),
            ('mlp', MLPClassifier(hidden_layer_sizes=(8),
                                  max_iter=10000))
           ]

pipeline = make_pipeline(
    DictVectorizer(),
    StandardScaler(with_mean=False),
    VotingClassifier(ensemble)
)
```

Ensemble technique: Averaging

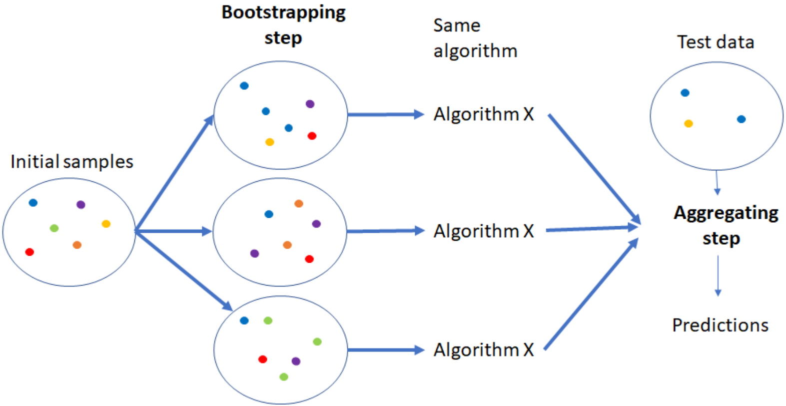


- Simple
- Weighted

Ensemble technique: Stacking



Ensemble technique: Bagging



Implementation of bagging and its variant

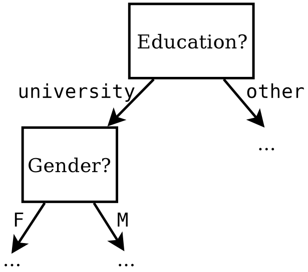
- With random subsets of data
  - without replacement (called **pasting**)
  - with replacement (called **bagging**, short for **bootstrap aggregating**)
  - When to choose pasting vs bagging?
- With random subsets of features (called **feature bagging** or **random subspaces**)

ensemble techniques in scikit-learn

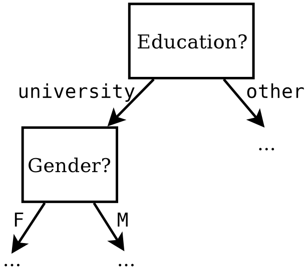
- `sklearn.ensemble.VotingClassifier`
- `sklearn.ensemble.VotingRegressor`
- `sklearn.ensemble.StackingClassifier`
- `sklearn.ensemble.StackingRegressor`
- `sklearn.ensemble.BaggingClassifier`
- `sklearn.ensemble.BaggingRegressor`

Overview

Decision tree recap

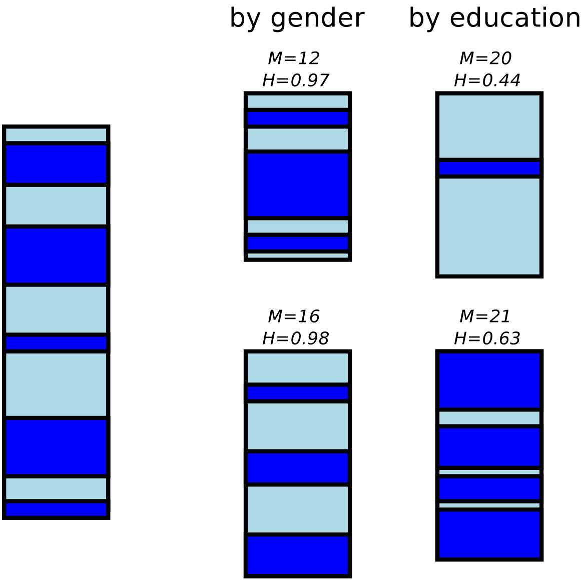


Decision tree recap



- Algorithm (simplified):
  1. Select the feature F that gives the “best” split
  2. Make a tree with F at the top
  3. Split into subsets based on F
  4. Make a tree for each subset

Selecting the feature for the top node



Decision trees aren't fantastic

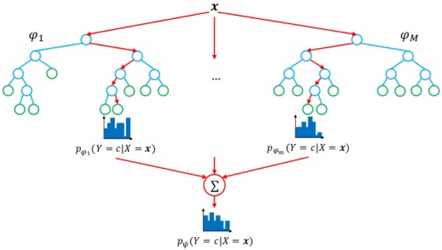
> *"Decision trees are a popular method for various machine learning tasks. Tree learning come[s] closest to meeting the requirements for serving as an off-the-shelf procedure for data mining, because it is invariant under scaling and various other transformations of feature values, is robust to inclusion of irrelevant features, and produces inspectable models. However, they are seldom accurate."*
>
> [Hastie et al., *The Elements of Statistical Learning*]

- Highly dependent on training data, tend to overfit
- All features are considered when splitting a node
- But there are several popular types of ensembles that typically build on decision trees

Overview

Random forest

- Is an ensemble of many decision trees built based on the same raw data and the same tree-algorithm
- Is a nonlinear estimator
- Are applicable for classification and regression
- Can take both numerical and categorical data

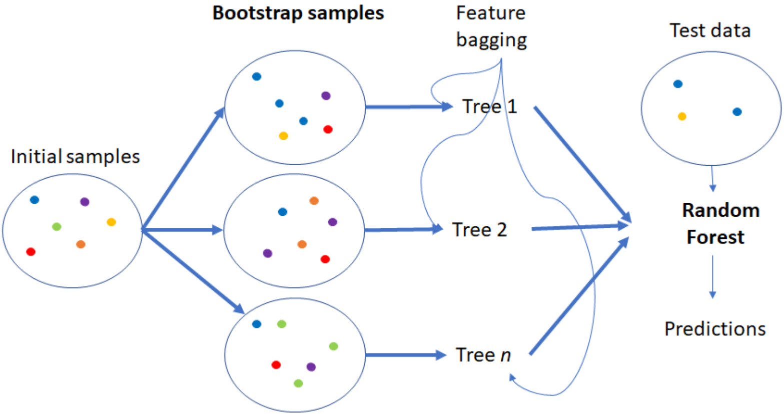


[source]

Training random forests

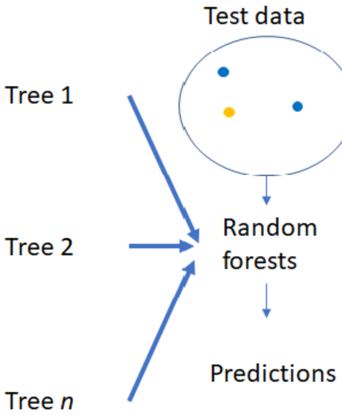
- The main idea: to train each individual tree to predict the target values well, but to build each tree to be as diverse as possible to the other trees in the same random forest.
- Why?
- How?

Making trees as diverse as possible in random forest



Making predictions with random forest

- Make a prediction for every tree in the random forest
- Then aggregate
  - for classification: use a majority vote.
  - for regression: take the mean of all the predictions.



Random forest algorithm

Algorithm 15.1 *Random Forest for Regression or Classification.*

1. For $b = 1$ to B:

   (a) Draw a bootstrap sample $\mathbf{Z}^*$ of size N from the training data.

   (b) Grow a random-forest tree T_b to the bootstrapped data, by recursively repeating the following steps for each terminal node of the tree, until the minimum node size n_{min} is reached.

      i. Select m variables at random from the p variables.

      ii. Pick the best variable/split-point among the m.

      iii. Split the node into two daughter nodes.

2. Output the ensemble of trees $\{T_b\}_1^B$.

To make a prediction at a new point x:

Regression: $\hat{f}_{\text{rf}}^B(x) = \frac{1}{B}\sum_{b=1}^B T_b(x)$.

Classification: Let $\hat{C}_b(x)$ be the class prediction of the bth random-forest tree. Then $\hat{C}_{\text{rf}}^B(x) = \textit{majority vote}\ \{\hat{C}_b(x)\}_1^B$.

[Hastie et al., *The Elements of Statistical Learning*]

in scikit-learn

- `sklearn.ensemble.RandomForestClassifier`
- `sklearn.ensemble.RandomForestRegressor`

- Discuss some parameters: n_estimators, max_features, random_state, n_jobs.

Overview

Review of random forests

- What is the difference between a normal decision tree and a tree within a random forest?
- What are the positive vs negative aspects of random forests?
- When is it good/not good to apply random forests?

Next lecture

- Preprocessing steps and hyperparameter tuning