Supervised Machine Learning: A Review of Classification Techniques

S. B. Kotsiantis
Department of Computer Science and Technology
University of Peloponnese, Greece
End of Karaiskaki, 22100 , Tripolis GR.
Tel: +30 2710 372164
Fax: +30 2710 372160
E-mail: sotos@math.upatras.gr

Overview paper

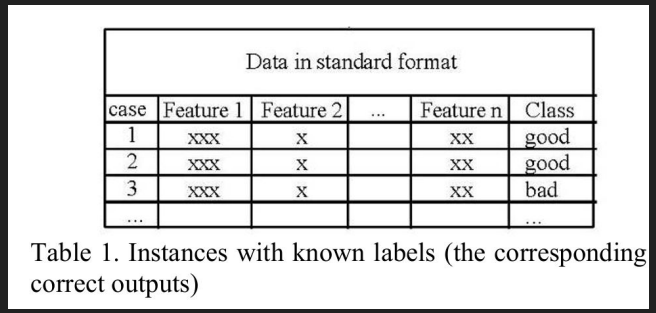
Keywords: classifiers, data mining techniques, intelligent data analysis, learning algorithms

Received: July 16, 2007

Supervised machine learning is the search for algorithms that reason from externally supplied instances to produce general hypotheses, which then make predictions about future instances. In other words, the goal of supervised learning is to build a concise model of the distribution of class labels in terms of predictor features. The resulting classifier is then used to assign class labels to the testing instances where the values of the predictor features are known, but the value of the class label is unknown. This paper describes various supervised machine learning classification techniques. Of course, a single article cannot be a complete review of all supervised machine learning classification algorithms (also known induction classification algorithms), yet we hope that the references cited will cover the major theoretical issues, guiding the researcher in interesting research directions and suggesting possible bias combinations that have yet to be explored.

Introduction

- Data
  - All data points are represented by the same set of features
  - Features may be continuous, categorical or binary
  - Data points may have a label (a.k.a class)
- Supervised learning
  - Some labels are known
  - Goal: predict unknown labels



Data in standard format					
case	Feature 1	Feature 2	...	Feature n	Class
1	xxx	x		xx	good
2	xxx	x		xx	good
3	xxx	x		xx	bad
...					...

Table 1. Instances with known labels (the corresponding correct outputs)

Related but different approaches

- Regression
  - Predict values, not labels
- Unsupervised classification
  - No labels are given
- Semi-supervised classification
  - Both labelled and unlabelled data

General framework

- Data collection
  - Identify features
  - Use expert, common-sense or brute-force
- Irrelevant features often include
  - Noise
  - Missing values
- “Garbage in - Garbage out”

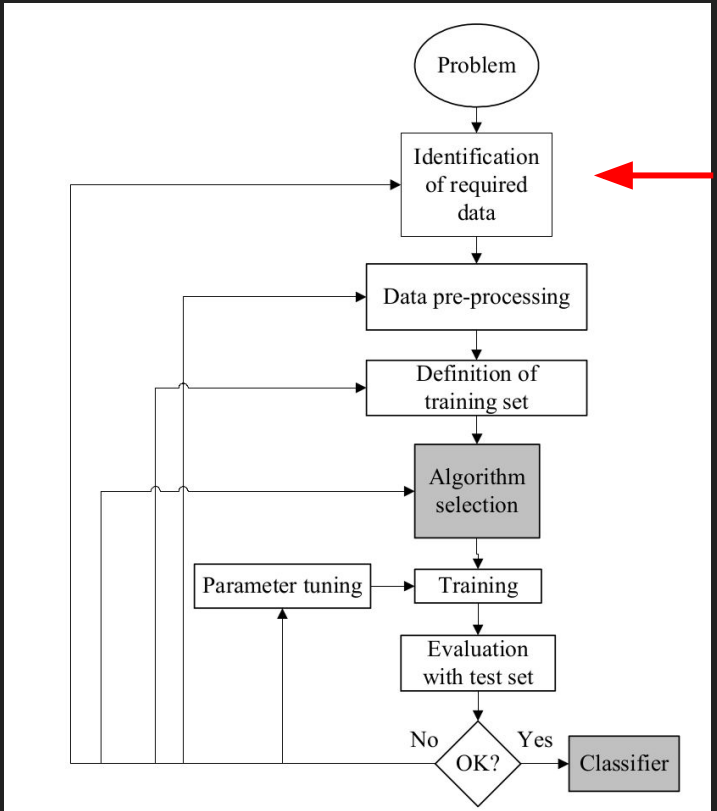


Figure 1. The process of supervised ML

Data pre-processing

- Encode categorical features
- Handle missing features
- Select / combine features
- Select data points

- Convert to library format
- Cast strings to numbers

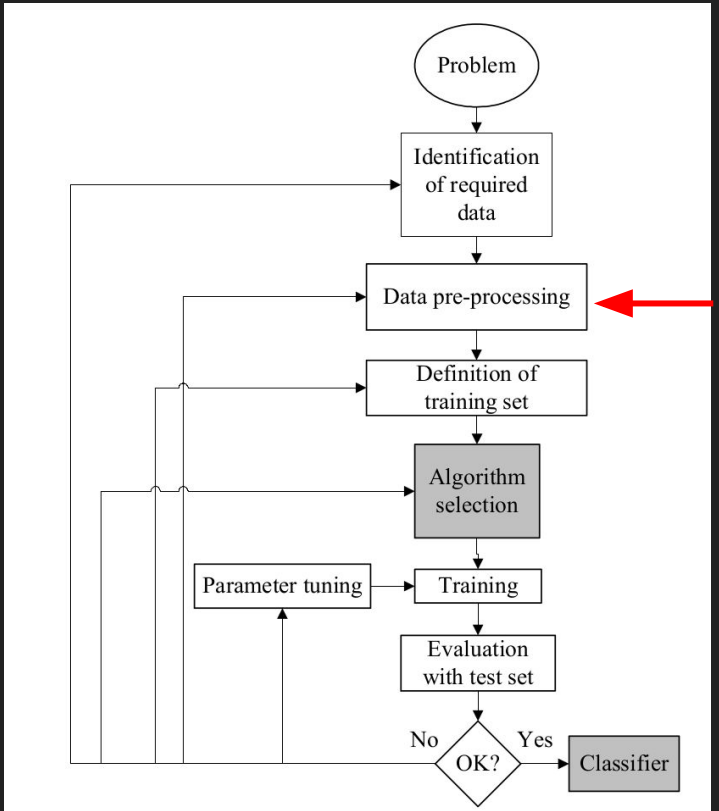


Figure 1. The process of supervised ML

Definition of training set

- Split labelled data in train vs test
- Learn on train; evaluate on test
- Split data randomly
- k-fold cross validation
- Leave-one-out validation

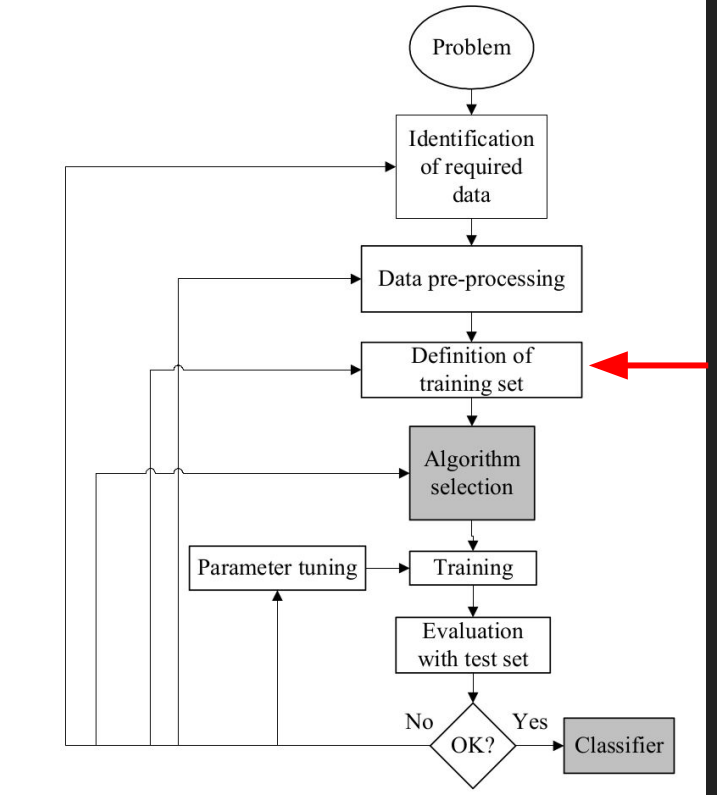
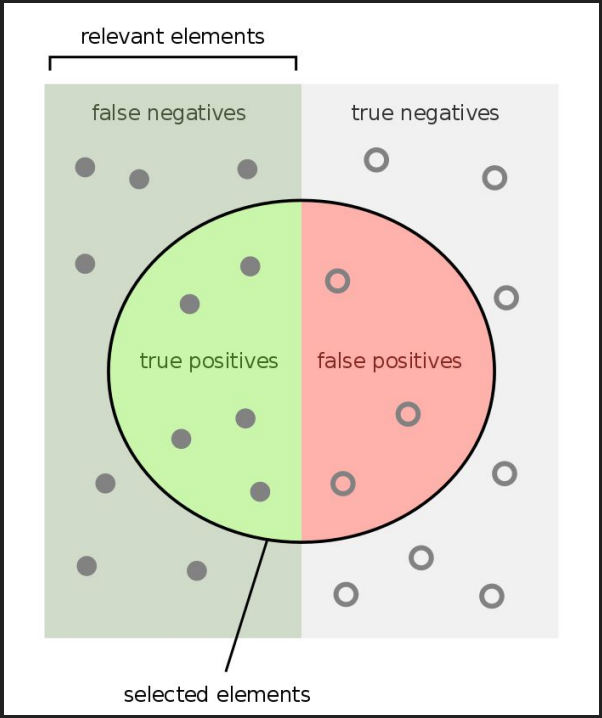


Figure 1. The process of supervised ML

Evaluation with test set



https://en.wikipedia.org/wiki/Sensitivity_and_specificity

Accuracy = (TP + TN) / (TP + TN + FP + FN)

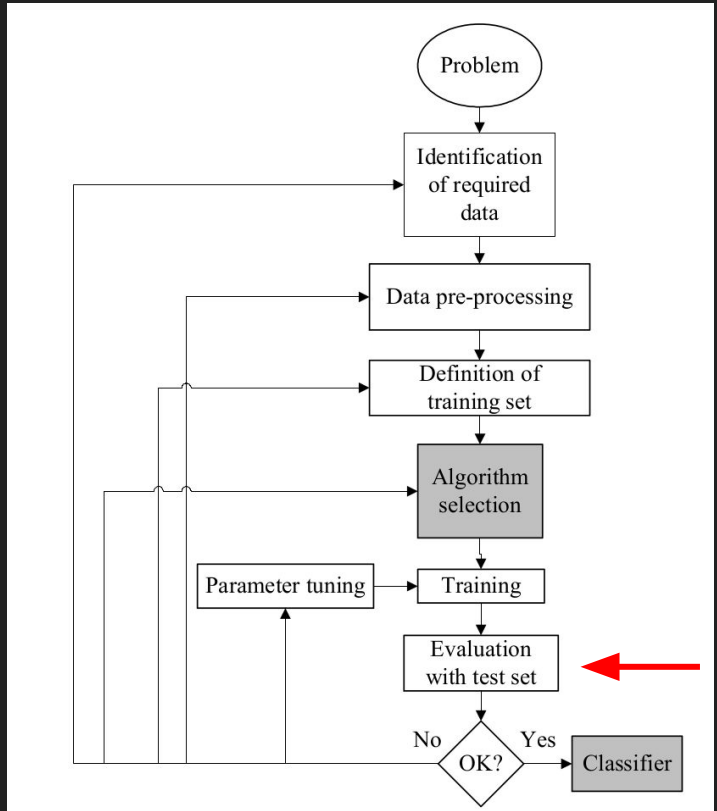


Figure 1. The process of supervised ML

Classification algorithms

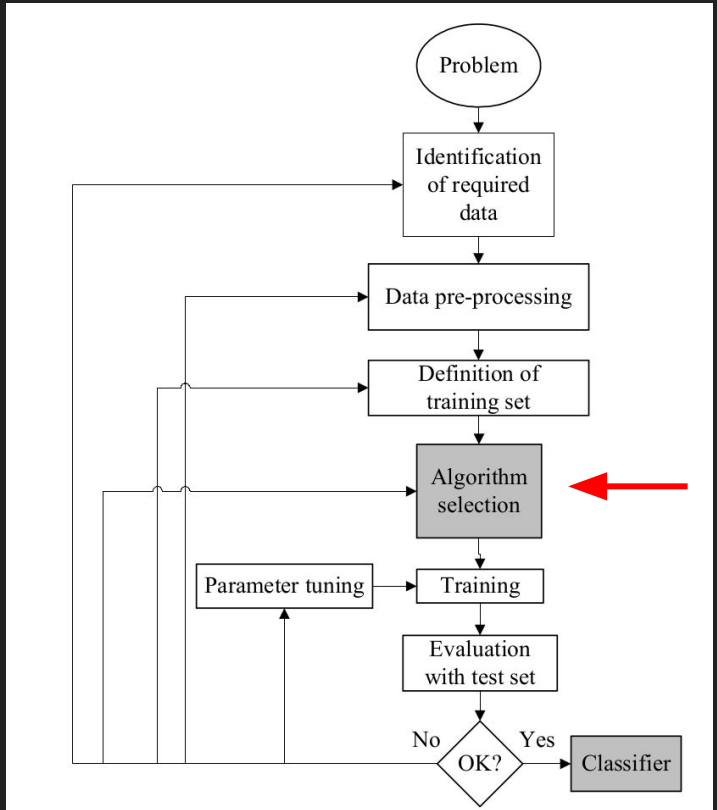


Figure 1. The process of supervised ML

Decision trees

- Sort data points from feature values
- Node: feature
- Branch: value range
- Leaf: label

at1	at2	at3	at4	Class
a1	a2	a3	a4	Yes
a1	a2	a3	b4	Yes
a1	b2	a3	a4	Yes
a1	b2	b3	b4	No
a1	c2	a3	a4	Yes
a1	c2	a3	b4	No
b1	b2	b3	b4	No
c1	b2	b3	b4	No

Table 2. Training Set

=>

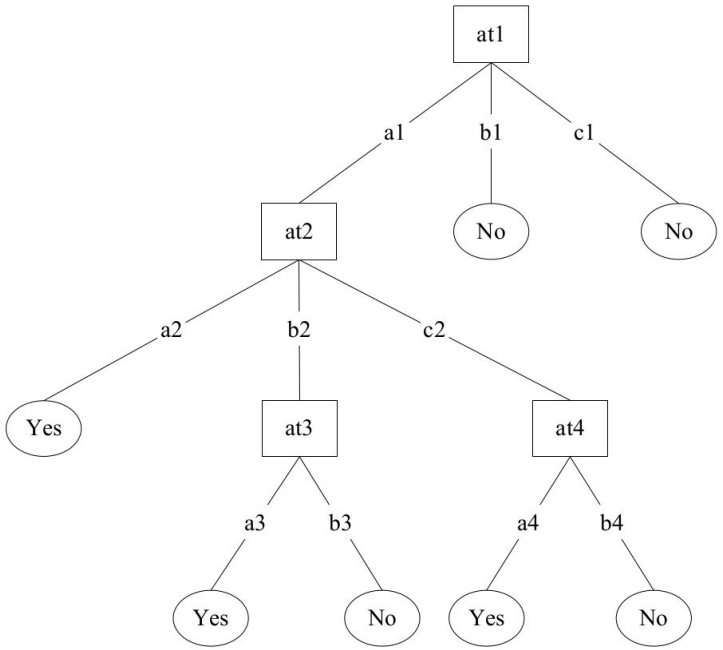


Figure 2. A decision tree

Decision trees: classification

(at1 = a1, at2 = b2, at3 = a3, at4 = b4)

classified as

“Yes”

at1	at2	at3	at4	Class
a1	a2	a3	a4	Yes
a1	a2	a3	b4	Yes
a1	b2	a3	a4	Yes
a1	b2	b3	b4	No
a1	c2	a3	a4	Yes
a1	c2	a3	b4	No
b1	b2	b3	b4	No
c1	b2	b3	b4	No

Table 2. Training Set

=>

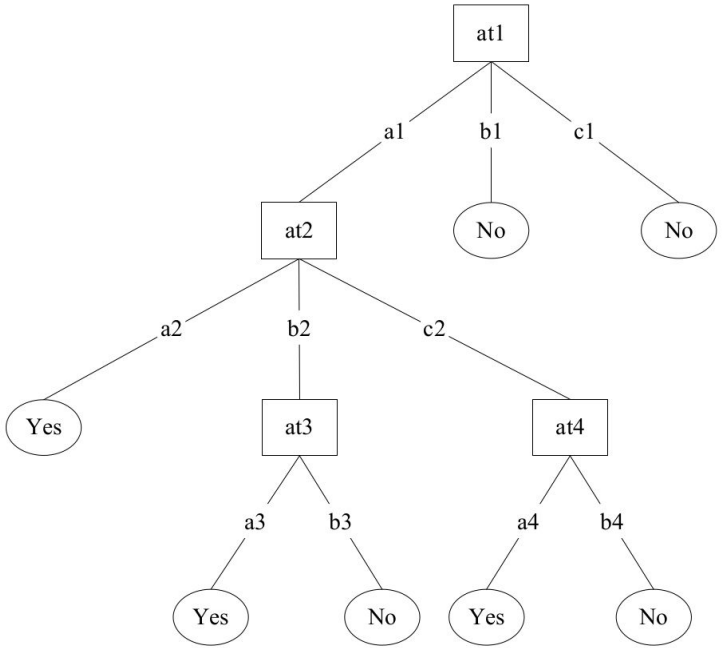


Figure 2. A decision tree

Overfitting

- Classifier ***c*** overfits when ***c'*** exists and:
  - Train: ***c*** >> ***c'***
  - Test: ***c*** << ***c'***

- "Too specific to the training set"
- "Too sensitive to noise"
- "Too focused on details"

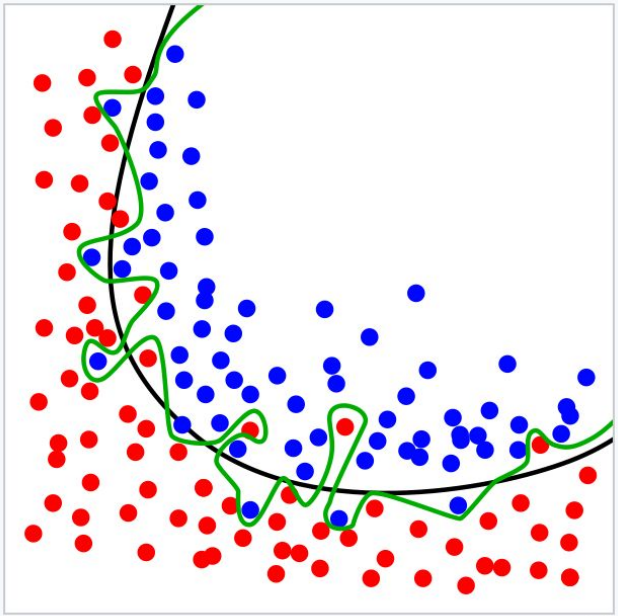


Figure 1. The green line represents an overfitted model and the black line represents a regularized model. While the green line best follows the training data, it is too dependent on that data and it is likely to have a higher error rate on new unseen data, compared to the black line.

https://en.wikipedia.org/wiki/Overfitting

How to avoid overfitting?

- Stop the training algorithm when:
  - Max depth of the tree is reached
  - Information gain (difference in Entropy or Gini) is below threshold
  - Min number of instances per node is reached
- Use more than 1 tree (Ensemble methods)
  - Grow more than 1 tree, using randomness
  - Keep trees that perform well
  - Example: Random Forest

Machine Learning, 45, 5–32, 2001
© 2001 Kluwer Academic Publishers. Manufactured in The Netherlands.

Random Forests

LEO BREIMAN
Statistics Department, University of California, Berkeley, CA 94720

Editor: Robert E. Schapire

Abstract. Random forests are a combination of tree predictors such that each tree depends on the values of a random vector sampled independently and with the same distribution for all trees in the forest. The generalization error for forests converges a.s. to a limit as the number of trees in the forest becomes large. The generalization error of a forest of tree classifiers depends on the strength of the individual trees in the forest and the correlation between them. Using a random selection of features to split each node yields error rates that compare favorably to Adaboost (Y. Freund & R. Schapire, *Machine Learning*: *Proceedings of the Thirteenth International conference*, ∗ ∗ ∗, 148–156), but are more robust with respect to noise. Internal estimates monitor error, strength, and correlation and these are used to show the response to increasing the number of features used in the splitting. Internal estimates are also used to measure variable importance. These ideas are also applicable to regression.

Keywords: classification, regression, ensemble

Random Forests

- Combination of independent decision trees randomly sampled
- Randomness governs
  - The sub-sample of the data from which trees are grown
  - The set of features considered at each node
- Classification: trees vote for labels