Data Mining: Lecture 5

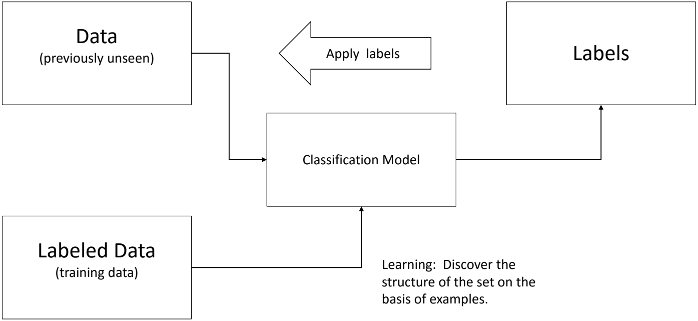
Classification

S. Nõmm

[1]Department of Software Science, Tallinn University of Technology

28.09.2021

Introduction

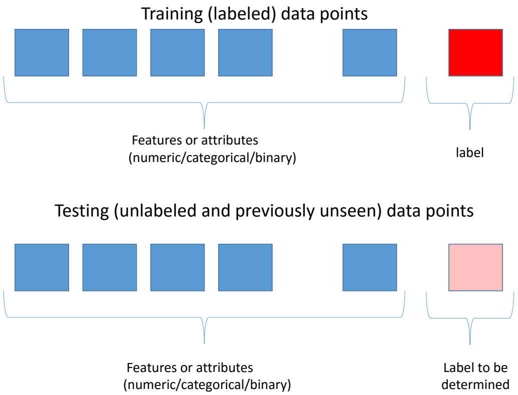


Learning: Discover the structure of the set on the basis of examples.

Introduction

- Classification problem may be seen as learning the structure of a data set of examples, already partitioned into categories or classes (labeled data set).
- Typically, learning leads the model.
- Model is used to estimate labels of the **previously unseen** data.
- Majority of the classification algorithms consist of two phases: Training and Testing
- Usually output of the classification is either Label or Numeric score.

Data



Training (labeled) data points

Features or attributes (numeric/categorical/binary)

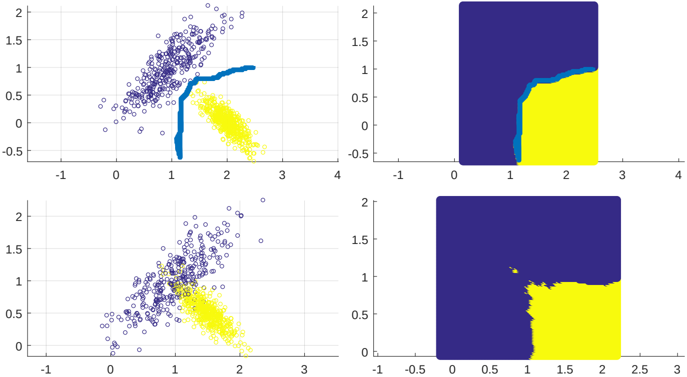
label

Testing (unlabeled and previously unseen) data points

Features or attributes (numeric/categorical/binary)

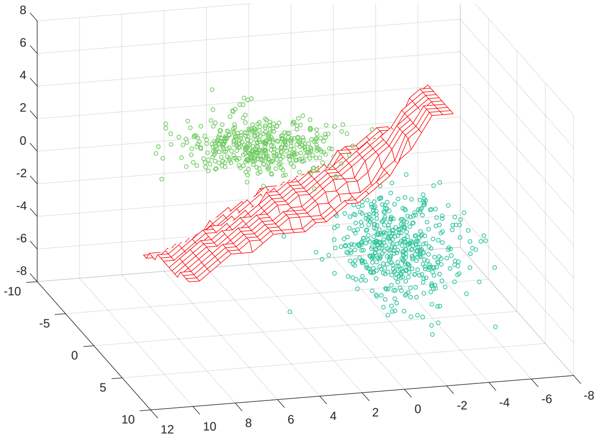
Label to be determined

Decision boundary, geometric interpretation, 2D



- Decision boundary (decision surface) (statistical classification with two classes) is a hypersurface that partitions the data set into two subsets, one for each class.
- Classifier tries to learn (construct) decision boundary that will lead minimal empirical error.

Decision boundary, 3D



Feature selection

- Filter models: a subset of features is evaluated with the use of a class-sensitive discriminative criterion.
  - Gini index.
  - Entropy.
  - Fisher score.
  - Fisher linear discriminant.
- Wrapper models.
- Embedded models.

Gini index

- Measures the discriminative power of a particular feature.
- Typically, it is used for categorical variables, but it can be generalized to numeric attributes by the process of discretization.
- Let $v_1, \dots, v_r$ are the possible values of the particular categorical attribute.
- Let p_j denotes the fraction of the data points containing attribute value v_i belonging to the class $j \in \{1, \dots, k\}$ to the data points containing attribute value v_i then Gini index defined as follows:

$$G(v_i) = 1 - \sum_{j=1}^{k} p_j^2.$$

- The value $1 - 1/k$ indicates that the different classes are distributed evenly for a particular attribute value.
- Lower values of the Gini index imply greater discrimination.

Gini index

The value-specific Gini index may be converted into an attribute wise Gini index.

- $$G(v_i) = 1 - \sum_{j=1}^{k} p_j^2.$$
- Let n_j denote the number of data points that take the value v_i, $\sum_{i=1}^{r} n_i = n$. Overall Gini index is defined as
$$G = \sum_{i=1}^{r} \frac{n_i G(v_i)}{n}.$$
- Lower values of the Gini index imply greater discriminative power.

Entropy

- The class-based entropy measure is related to notions of information gain resulting from fixing a specific attribute value.
- The class- base entropy is defied as follows:

$$E(v_i) = -\sum_{j=1}^{k} p_i \log_2(p_j))$$

  takes its values in $[0, \log_2(k)]$, whereas greater values indicate greater mixing.
- By analogy with Gini index one may define overall Entropy as

$$E = \sum_{i=1}^{r} \frac{n_i E(v_i)}{n}.$$

Fisher score

- The Fisher score is naturally designed for numeric attributes to measure the ratio of the average interclass separation to the average intraclass separation.
- The larger the Fisher score, the greater the discriminatory power of the attribute.
- Let μ_j and σ_j denote the mean and the standard deviation of the of the data points belonging to the class j, for a particular feature. And let p_j be the fraction of the points belonging to the class j. Finally let μ define the mean of the entire data set. The Fisher index is defined as follows:

$$F = \frac{\displaystyle\sum_{j=1}^{k} p_j(\mu_j - \mu)^2}{\displaystyle\sum_{j=1}^{k} p_j\sigma_j^2}$$

- The attributes with the largest value of the Fisher score may be selected for use with the classification algorithm.

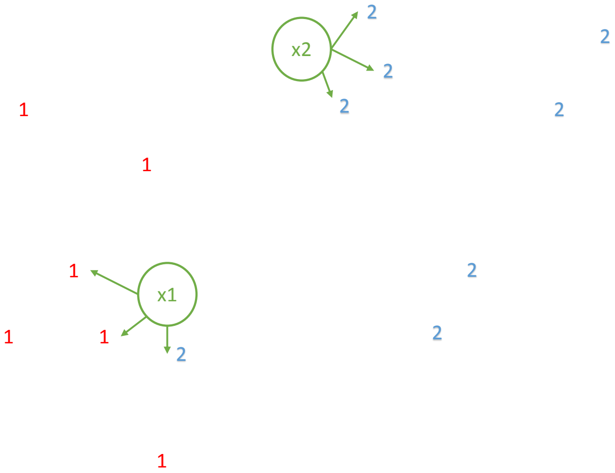
k-nearest neighbour (k-NN) classification

- Let N be a labeled set of points belonging to c different classes such that

$$\sum_{i=1}^{c} N_i = N$$

- Classification of a given point x
    - Find k - nearest points to the point x.
    - Assign x the majority label of neighbouring (k-nearest) points

Example

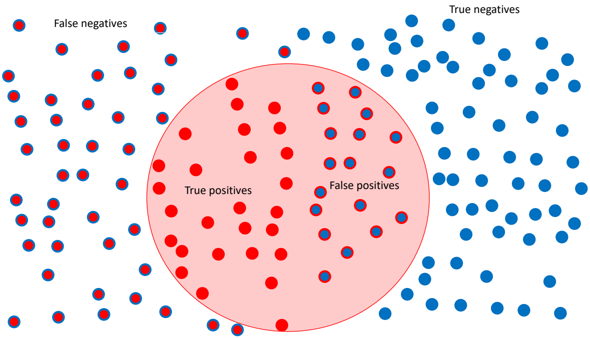


Classification model goodness!

- How good is the model?
- What is the goal of modeling?

Classification outcome

- Consider binary classifier.
- In the data set there are two classes: Positive (P) and negative (N)
- Outcomes of the classification: True positive, true negative, false positive (type I error), false negative(type II error).



Context of information retrieval

NB! Observe notions!

- Relevant elements of the data set. One is interested to find (retrieve elements of the certain class).
- Precision is defined as:

$$\text{precision} = \frac{|\text{relevant} \cap \text{retrieved}|}{|\text{retrieved}|}$$

- Recall is defined as:

$$\text{recall} = \frac{|\text{relevant} \cap \text{retrieved}|}{|\text{relevant}|}$$

Context of classification I

Denote: tp - true positive, tn - true negative, fp - false positive and fn - false negative.

- Precision:

$$\text{Precision} = \frac{tp}{tp + fp}$$

- Recall:

$$\text{Recall} = \frac{tp}{tp + fn}$$

- True negative rate (Specificity):

$$\text{TNR} = \frac{tn}{tn + fp}$$

- Accuracy:

$$\text{Accuracy} = \frac{tp + tn}{tp + tn + fp + fn}$$

- Predicted positive condition rate

$$\text{Predicted} \quad \text{positive} \quad \text{condition} \quad \text{rate} = \frac{tp + fp}{tp + tn + fp + fn}$$

F-measure not to be confused with similarly named values!!!

Frequently referred as F_1-score ... is harmonic average of precision and recall.

- $$F = 2 * \frac{\text{precisionrecall}}{\text{precision} + \text{recall}}$$
- More general definition:

$$F_\beta = (1 + \beta^2) \frac{\text{precisionrecall}}{\beta^2 \text{precision} + \text{recall}}$$

Decision trees

- Non-parametric supervised learning technique.
- Tree-like graph is used to represent the model of decision making and possible consequences of such decisions.
- Internal nodes are conditions (questions). terminal nodes represent labels of classes.
- Questions or conditions play a role of features. Answers to the questions are referred as feature values.
- Training a tree model is referred as tree growing.

Growing a tree 1

Greedy heuristic is the most popular technique. Let F be the possible set of features and S is the subset of data. The idea is to find most useful feature (among remaining) at each node.

$$\begin{aligned} j(S) = \arg\min_{j \in F} \mathrm{cost}(\{x_i, y_i : x_i \in S, x_{i,j} = c_k\}) \\ + \mathrm{cost}(\{x_i, y_i \, x_i \in S, x_{i,j} \neq c_k\}) \end{aligned}$$

Classification cost:

$$\hat{\pi}_c = \frac{1}{|S|} \sum_{x_i \in S} \mathbb{1}\{y_i = c\}$$

Misclassification rate:

$$\frac{1}{|S|} \sum_{x_j} \in S \mathbb{1}(y_i \neq \hat{y}) = 1 - \hat{\pi y}$$

Cost functions

- Entropy:

$$\mathbb{H}(\hat{\pi}) = -\sum_{c=1}^{C} \hat{\pi}_c \log_2 \hat{\pi}_c$$

  Minimizing entropy is equivalent to maximizing information gain which is $\mathbb{H}(Y) - \mathbb{H}(Y|X_j)$.

- Gini index:

$$G = \sum_{c=1}^{C} \hat{\pi}_c(1 - \hat{\pi}_c)$$

Growing a tree 3

- Repeat:
    - For each feature divide data into corresponding subsets. Evaluate accuracy of such split with respect to response variable.
    - "Most accurate" feature wins. It will become condition at a given node.
    - Exclude chosen feature from the feature set.
- Until no more features left.

Example: When to play tennis

Outlook	Temperature	Humidity	Wind	Play
sunny	warm	high	weak	no
sunny	warm	high	strong	no
rain	warm	high	weak	yes
rain	cool	normal	weak	yes
rain	cool	normal	strong	no
sunny	cool	normal	strong	yes
sunny	warm	high	weak	no
sunny	cool	normal	weak	yes
rain	warm	normal	weak	yes
sunny	warm	normal	strong	yes
rain	warm	high	strong	yes
sunny	warm	normal	weak	yes
rain	warm	high	strong	no

Information gain

Definition

Information gain G_I of an action is the decrease of the ambiguity achieved as the result of the action.

- In the context of decision tree growing the action is splitting the node.
- If entropy is chosen as the cost function then information gain is defined as follows:

$$G_I = E - (E_l \cdot p_l + E_r \cdot p_r)$$

where E is the entropy before splitting E_l is the entropy of left child and E_r is the entropy of the right child. Indexes r and l have the same meaning for the proportions p.

Growing the tree: case of continues features

Denote X the matrix where columns correspond to different features and rows correspond to the different observation points.

- If all the data points are of the same class return the leaf node that predicts this class.
- Among all splitting points for each column find the one giving largest information gain.
- Then chose the column with the maximum gain.
- Perform splitting.
- If stopping criteria is satisfied return the tree.
- If stopping criteria is not satisfied apply tree growing procedure to each child.

Pruning

- In order prevent overfitting stop growing the tree when the decrease is not sufficient to justify adding extra subtree.
- Grow a full tree and then prune the branches giving less decrease in error.

Wrapper Models

- Filter models are agnostic to the particular classification algorithm being used. In some cases, it may be useful to leverage the characteristics of the specific classification algorithm to select features.
- Wrapper models can optimize the feature selection process to the classification algorithm at hand.
- Let $\mathcal{A}$ denote a specific classification algorithm. The basic strategy in wrapper models is to iteratively refine a current set of features F by successively adding features to it.
- The algorithm starts with the F set to be empty then two following steps are repeated
    - Create an augmented set of features F by adding one or more features to the current feature set.
    - Use a classification algorithm $\mathcal{A}$ to evaluate the accuracy of the set of features F. Use the accuracy to either accept or reject the augmentation of F.

Rule-Based Classifiers

- Rule-based classifiers use a set of ifthen rules $\mathcal{R} = \{\mathcal{R}_1, \ldots, \mathcal{R}_m\}$ to match antecedents to consequents. A rule is typically expressed in the following form:

$$\text{IF} \quad Condition \quad \text{THEN} \quad Condition$$

- The condition on the left-hand side of the rule, also referred to as the antecedent, may contain a variety of logical operators.
- The right-hand side of the rule is referred to as the consequent, and it contains the class variable.
- The training phase of a rule-based algorithm creates a set of rules. The classification phase for a test instance discovers all rules that are triggered by the test instance.
- In some cases, methods are required to resolve the conflicts in class label prediction.

Bayes theorem

- Let us suppose that there k classes are given.
- The posterior probability of a class C_k for an input x is:

$$p(C_k \mid x) = \frac{p(\boldsymbol{x} \mid C_k)p(C_k)}{p(x)}$$

- $p(\boldsymbol{x} \mid C_k)$ is the likelihood, $p(C_k)$ is the prior probability, $p(x)$ is the marginal data likelihood.
- $p(C_k)$ is the probability of a class $p(C_k)$ a priori, before getting about any knowledge about the data.
- $p(C_k \mid \boldsymbol{x})$ is the class probability a posteriori, after getting knowledge about the data.
- Bayes theorem updates prior distribution into posterior on the basis of empiric information.

Conditional and unconditional independence

- If X and Y are unconditionally independent then their joint distribution is the product of the marginal distributions:

$$X \perp Y \Leftrightarrow p(X,Y) = p(X)p(Y)$$

- If the influence is mediated through a third variable Z, then X and Y are said to be conditionally independent

$$X \perp Y \mid Z \Leftrightarrow p(X,Y \mid Z) = p(X \mid Z)p(Y \mid Z)$$

- Conditional independence does not imply unconditional independence and vice versa:

$$X \perp Y \mid Z \not\Leftrightarrow X \perp Y$$

Naïve Bayes assumption

- Likelihood is computed as:

$$p(\boldsymbol{x} \mid y) = \prod_{j=1}^{n} p(x_j \mid y)$$

- Naïve Bayes assumption: the features are conditionally independent given the class label.
- the word naïve refers to the fact that actually features are not expected to be independent or conditionally independent.
- Model has relatively few parameters and therefore immune to overfilling.

Prediction with naïve Bayes model

- the goal is to find wether a new element is of class 1 or 0 (in the example of spam filtering wether given e-mail message is spam or not).
- According to Bayes theorem.

$$p(y=1 \mid \boldsymbol{x},\boldsymbol{\theta}) \propto p(\boldsymbol{x} \mid y,\boldsymbol{\theta})p(y \mid \boldsymbol{\theta}) = p(y=1 \mid \theta)\prod_{j=1}^{n} p(x_{i,j} \mid y=1,\boldsymbol{\theta})$$

$$p(y=0 \mid \boldsymbol{x},\boldsymbol{\theta}) \propto p(\boldsymbol{x} \mid y,\boldsymbol{\theta})p(y \mid \boldsymbol{\theta}) = p(y=0 \mid \theta)\prod_{j=1}^{n} p(x_{i,j} \mid y=0,\boldsymbol{\theta})$$

- Predict the class with highest posterior probability:

$$y^* = \arg\max_{y\in\{0,1\}} p(y \mid \boldsymbol{x},\boldsymbol{\theta})$$