M. Strohmaier, P. Singer, F. Lemmerich, E. Vasilev
GESIS - Leibniz Institute for the Social Sciences – Computational Social Science Department
Universität Koblenz-Landau

16.11.2016

2. Home Assignment "Machine Learning and Data Mining"

Winter semester 2016 / 2017

You should work on the assignment in groups of 2–3 participants. Send your solution (as .pdf) by email to sent to `vasilev@uni-koblenz.de`, `florian.lemmerich@gesis.org`, and `philipp.singer@gesis.org` until **Tuesday, 22nd of November, 23:59h**. Use `[ML-Assignment]` as the email subject and do not forget to denote the name of all contributing students in the pdf as well as in the email. The deadline is strict!

1 Clustering vs Classification

1. What is the main difference between *clustering* and *classification*?

> For classification, the groups used for categorization are known beforehands. For clustering, the groups are unknown. Clustering methods are *unsupervised*, classification methods are *supervised*.

2. What is easier to evaluate, clustering or classification?

> Classification is easier to evaluate, because the classification result can be (objectively) right or wrong. For clustering, the quality of clustering results have a subjective component.

2 k-Nearest Neighbor

Given are data points $x_1, \ldots, x_{11}$:

$$x_1 = \begin{pmatrix} 3 \\ 2 \end{pmatrix}, x_2 = \begin{pmatrix} 4 \\ 3 \end{pmatrix}, x_3 = \begin{pmatrix} 5 \\ 2 \end{pmatrix}, x_4 = \begin{pmatrix} 3 \\ 6 \end{pmatrix}, x_5 = \begin{pmatrix} 8 \\ 5 \end{pmatrix}, x_6 = \begin{pmatrix} 8 \\ 6 \end{pmatrix},$$

$$x_7 = \begin{pmatrix} 5 \\ 5 \end{pmatrix}, x_8 = \begin{pmatrix} 7 \\ 3 \end{pmatrix}, x_9 = \begin{pmatrix} 4 \\ 6 \end{pmatrix}, x_{10} = \begin{pmatrix} 3 \\ 7 \end{pmatrix}, \text{ and } x_{11} = \begin{pmatrix} 6 \\ 4 \end{pmatrix}$$

The data points $x_1, \ldots, x_6$ belong to class c_1, the data points $x_7, \ldots, x_{10}$ belong to class c_2. x_{11} has not yet been classified.

1. Classify the data point x_{11} with the k-nearest-neighbor approach and $k = 1$. Use the Euclidean distance as the distance metric.

Compute distances:

$d(x_1, x_{11})$ = 3.61
$d(x_2, x_{11})$ = 2.24
$d(x_3, x_{11})$ = 2.24
$d(x_4, x_{11})$ = 3.61
$d(x_5, x_{11})$ = 2.24
$d(x_6, x_{11})$ = 2.83
$d(x_7, x_{11})$ = 1.41
$d(x_8, x_{11})$ = 1.41
$d(x_9, x_{11})$ = 2.83
$d(x_{10}, x_{11})$ = 4.24

$k = 1$ means that only the one closest data point counts. There is a tie for the closest data point between: x_7 and x_8.
Regardless of the random choice between these two points, the x_{11} is classified as belonging to class c_2.

2. Classify the data point x_{11} with the k-nearest-neighbor approach and $k = 5$. Use the Euclidean distance as the distance metric.

The five nearest neighbors are: x_7, x_8, x_2, x_3, x_5
Of these, 3 belong to class c_2 and 1 belong to class c_1. Thus, x_{11} will be classified into class c_1.

3. Classify the data point x_{11} with the k-nearest-neighbor approach and $k = 5$. Use the Euclidean distance as distance metric. As weights, use the inverted distance to the point to be classified, i.e., $\frac{1}{d(x_n, x_c)}$, with x_n is the neighbor, and x_c is the point to be classified.

The five nearest neighbors are: x_7, x_8, x_2, x_3, x_5
The corresponding weights w are:

$w(x_7) = w(x_8) = \frac{1}{1.41} = 0.71$
$w(x_2) = w(x_3) = w(x_5) = \frac{1}{2.24} = 0.45.$

Sum of weights for class c_1 : $0.45 + 0.45 + 0.45 = 1.35$. Sum of weights for class c_2 : $0.71 + 0.71 = 1.42$. Thus, x_{11} gets classified as c_2.

3 Naive Bayes

Given is the following dataset:

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
D_1	Sunny	Hot	High	Weak	No
D_2	Sunny	Hot	High	Strong	No
D_3	Overcast	Hot	High	Weak	Yes
D_4	Rain	Mild	High	Weak	Yes
D_5	Rain	Cool	Normal	Weak	Yes
D_6	Rain	Cool	Normal	Strong	No
D_7	Overcast	Cool	Normal	Strong	Yes
D_8	Sunny	Mild	High	Weak	No
D_9	Sunny	Cool	Normal	Weak	Yes
D_{10}	Rain	Mild	Normal	Weak	Yes
D_{11}	Sunny	Mild	Normal	Strong	Yes
D_{12}	Overcast	Mild	High	Strong	Yes
D_{13}	Overcast	Hot	Normal	Weak	Yes
D_{14}	Rain	Mild	High	Strong	No

1. Train a naive-Bayes classificator with this data and use it to classify the new data point D_{15} = *(O=Overcast, T=Cool, H=High, W=Strong)*!

$$c_{NB} = \underset{c \in \{yes,no\}}{\text{argmax}}\ P(c) \prod_{i=1}^{4} P(o_i|c)$$

$$c_{NB} = \underset{c \in \{yes,no\}}{\text{argmax}}\, P(c) \cdot P(O = Overcast|c) \cdot P(T = Cool|c) \cdot$$
$$P(H = High|c) \cdot P(W = Strong|c)$$

for $c = yes$ we get:

$$\frac{9}{14} \cdot \frac{4}{9} \cdot \frac{3}{9} \cdot \frac{3}{9} \cdot \frac{3}{9} = \frac{2}{189} \approx 0.0106$$

for $c = no$ we get:

$$\frac{5}{14} \cdot \frac{0}{5} \cdot \frac{1}{5} \cdot \frac{4}{5} \cdot \frac{3}{5} = 0$$

Thus, D_{15} gets classified as $c_{NB} = yes$.

2. Use your classifier to classify the data point
   D_{16} = *(T=Cool, H=High, W=Strong)*!
   The value for the attribute *Outlook* was not recorded for this data point.

$$c_{NB} = \underset{c \in \{yes,no\}}{\text{argmax}}\, P(c)P(T = Cool|c)P(H = High|c)P(W = Stro|c)$$

for $c = yes$ we get:

$$\frac{9}{14} \cdot \frac{3}{9} \cdot \frac{3}{9} \cdot \frac{3}{9} = \frac{1}{42} \approx 0.0238$$

for $c = no$ we get:

$$\frac{5}{14} \cdot \frac{1}{5} \cdot \frac{4}{5} \cdot \frac{3}{5} = \frac{6}{175} \approx 0.0343$$

Thus, D_{16} gets classified as $c_{NB} = no$.

3. Compare (*short* explanation!) the results for subtasks 1 and 2!

By ignoring the attribute *Outlook* the classification changes from *yes* to *no*. The probability of $P(O = Overcast|no) = 0$ in subtask 1 dominates the whole term, thus all other features do not influence the result. This is not the case for subtask 2.

4 Decision Trees

A hospital wants to provide support for his physicians. For that purpose they collected data on healthy and ill patients. Decision makers read in a magazine that decision trees allow to create a model that can be used to simulate decision of a doctor.

Patient Nr.	Heart Rate	Blood Pressure	Class
1	irregular	Normal	Ill
2	regular	Normal	Healthy
3	irregular	Abnormal	Ill
4	irregular	Normal	Ill
5	regular	Normal	Healthy
6	regular	Abnormal	Ill
7	regular	Normal	Healthy
8	regular	Normal	Healthy

1. Compute a decision tree for this dataset (using information gain as a split criterion, without using the patient id) and sketch it graphically. If you want to do compute this with pen and paper only (no computer or calculator), you can use the following approximations for the logarithms:

x	$\frac{1}{5}$	$\frac{1}{3}$	$\frac{2}{3}$	$\frac{4}{5}$
$\log_2(x)$	$-\frac{23}{10}$	$-\frac{8}{5}$	$-\frac{3}{5}$	$-\frac{3}{10}$

We use the following formulas:

$$\text{informationsgewinn}(X) = \text{entropy}(T) - \text{entropy}_X(T) \tag{1}$$

$$\text{entropy}(T) = -\sum_{i=1}^{k} \frac{|C_i \cap T|}{|T|} \log_2\left(\frac{|C_i \cap T|}{|T|}\right) \tag{2}$$

$$\text{entropy}_X(T) = \sum_{i=1}^{n} \frac{|T_i|}{|T|} \text{entropy}(T_i) \tag{3}$$

$$\text{entropy}(T_i) = -\sum_{j=1}^{k} \frac{|C_j \cap T_i|}{|T_i|} \log_2\left(\frac{|C_j \cap T_i|}{|T_i|}\right) \tag{4}$$

Here, n is the number of different values of attribute X, T_i is the set of data points that have the value i in attribute X. Furthermore, k denotes the number of different values for the class attribute C, and C_i is the set of data points in the training set that are classified as i. Finally, T is the set of all data points in the dataset.

The entropy for the whole dataset is: $\text{entropy}(T) = -\left(\frac{4}{8} \cdot \log_2\left(\frac{4}{8}\right) + \frac{4}{8} \cdot \log_2\left(\frac{4}{8}\right)\right) = 1$ We now compute the split with the best information gain. According to the formulas that is the split that induces the minimal weighted entropys.

- Evaluate the split with the attribute Heart Rate (HR), i.e., divide T in $T_{regular}$ and $T_{irregular}$: $\text{entropy}(T_{regular}) = -\left(\frac{1}{5} \cdot \log_2\left(\frac{1}{5}\right) + \frac{4}{5} \cdot \log_2\left(\frac{4}{5}\right)\right) \approx \frac{7}{10}$

  $\text{entropy}(T_{irregular}) = -\left(\frac{3}{3} \cdot \log_2\left(\frac{3}{3}\right) + \frac{0}{3} \cdot \log_2\left(\frac{0}{3}\right)\right) = 0$

  Thus: $\text{entropy}_{HR} = \frac{5}{8} \cdot \frac{7}{10} + \frac{3}{8} \cdot 0 = \frac{7}{16}$
- Evaluate the split with the attribute Blood Pressure (BP): $\text{entropy}(T_{normal}) = -\left(\frac{2}{6} \cdot \log_2\left(\frac{2}{6}\right) + \frac{4}{6} \cdot \log_2\left(\frac{4}{6}\right)\right) \approx \frac{14}{15}$

  $\text{entropy}(T_{abnormal}) = -\left(\frac{2}{2} \cdot \log_2\left(\frac{2}{2}\right) + \frac{0}{2} \cdot \log_2\left(\frac{0}{2}\right)\right) = 0$

  Thus: $\text{entropy}_{HR} = \frac{6}{8} \cdot \frac{14}{15} + \frac{2}{8} \cdot 0 = \frac{7}{10}$
- $\frac{7}{16} < \frac{7}{10}$ and thus the first split uses the attribute Heart Rate.
- Since there is only of attribute left this is used for the second split. The second split is not necessary for the branch Heart Rate = irregular, since all data points in this branch have the same class already.

2. What would be the information gain for the split with the attribute *Patient Nr.* in the overall dataset? Compare the result with the information gain of other attributes. What problems occur when a decision tree uses the attribute *Patient Nr.*?

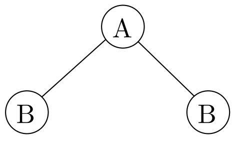
   The information gain for this split is maximal. However, it does generalize very poorly. New patients to be classified will have new patient nr. that do not occur in the data, and thus cannot be classified.

3. State a simple, but valid strategy to parallelize decision tree learning?

   Each branch has a separate list of attributes and instances and thus can be independently processed at a separate computational node.

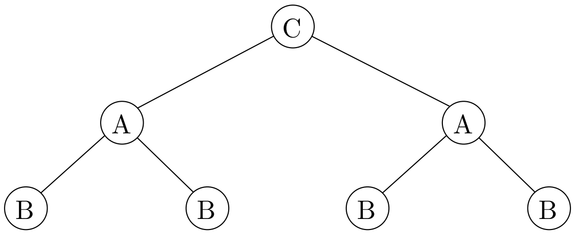
4. Give an example dataset such that the approach of greedily learning decision trees with information gain does *not* lead to an optimum tree with respect to the tree complexity (measured as number of nodes in the tree). Assume that the decision tree should correctly classify all training examples (no pruning).

   We construct a dataset with attributes with A, B and C, such that a correct classifying decision tree T_{opt} with minimum complexity looks as follows:

   That is, the attribute C is not all necessary for the decision process.

However, by constructing the decision tree according to the Information Gain, the attribute C will be selected first, leading the following Tree D_{Greedy}:



This is accomplished by the following dataset:

A	B	C	class
0	0	0	+
0	0	1	+
0	0	1	+
0	1	0	-
0	1	1	-
1	0	0	-
1	0	1	-
1	1	0	+
1	1	1	+
1	1	1	+

The dataset is constructed in a way such that class is positive exactly if the values of A and B are identical. Thus, the above tree T_{opt} correctly classifies the dataset.

We now show that with greedy construction using information gain C is selected first: The class distribution in T is $\frac{3}{5}$ vs. $\frac{2}{5}$. A split using attribute A or a split using attribute B leads to the same class distribution and thus to an Information gain of 0.

By contrast, splitting on C leads to a positiv information gain:

$$\begin{aligned}
inf\_gain(T,C) &= entropy(T) - \sum_{i\in\{0,1\}} \frac{|T_{C=i}|}{|T|} \cdot entropy(T_{C=i}) \\
&= -\left(\frac{3}{5} \cdot ld\left(\frac{3}{5}\right) + \frac{2}{5} \cdot ld\left(\frac{2}{5}\right)\right) \\
&- \frac{4}{10} \cdot \left(\frac{1}{2} \cdot ld\left(\frac{1}{2}\right) + \frac{1}{2} \cdot ld\left(\frac{1}{2}\right)\right) \\
&- \frac{6}{10} \cdot \left(\frac{2}{3} \cdot ld\left(\frac{2}{3}\right) + \frac{1}{3} \cdot ld\left(\frac{1}{3}\right)\right) \\
&> 0
\end{aligned}$$

After splitting on C, in both branches still attributes A and B are both required for a correct classification. Thus, D_{Greedy} has not a minimal complexity.