Machine Learning Worksheet 2

Decision Trees and Nearest Neighbours

1 Learning by doing

You are free to do these completely by hand or use a computer to help speed things up (python, MATLAB, R, Excel, ...). You should, however, show the basic steps of your work and implement your own "helpers" instead of blindly using code. Using a machine learning toolbox and copying the result will not help you understand.

The table below gives you a feature matrix $\boldsymbol{X}$ together with the output z_i for every row i of the feature matrix. This data is also available in Piazza as *homework-02.csv*.

$x_{i,1}$	$x_{i,2}$	$x_{i,3}$	z
5.5	0.5	4.5	2
7.4	1.1	3.6	0
5.9	0.2	3.4	2
9.9	0.1	0.8	0
6.9	-0.1	0.6	2
6.8	-0.3	5.1	2
4.1	0.3	5.1	1
1.3	-0.2	1.8	1
4.5	0.4	2.0	0
0.5	0.0	2.3	1
5.9	-0.1	4.4	0
9.3	-0.2	3.2	0
1.0	0.1	2.8	1
0.4	0.1	4.3	1
2.7	-0.5	4.2	1

Problem 1: Build a decision tree T for your data. Consider all possible feature tests and use the Gini index to build your tree. Build the tree only to a depth of two! Provide at least the value of the final Gini index at each node and the distribution of classes at each leaf.

Problem 2: Use the tree from the previous problem to classify the vectors $\boldsymbol{x}_a = (4.1, -0.1, 2.2)$ and $\boldsymbol{x}_b = (6.1, 0.4, 1.3)$. Provide both your classification $\hat{z}_a$ and $\hat{z}_b$ and their respective probabilities $p(c = \hat{z}_a \mid \boldsymbol{x}_a, T)$ and $p(c = \hat{z}_b \mid \boldsymbol{x}_b, T)$

Problem 3: Classify the two vectors given in Problem 2 with the K-nearest neighbors algorithm. Use $K = 3$ and Euclidean distance.

Problem 4: Now, consider z_i to be real-valued labels rather than classes. Perform 3-NN regression to label the vectors from Problem 2.

Problem 5: Look at the data. Which problem do you see wrt. building a Euclidean-distance-based KNN model on $\boldsymbol{X}$? How can you compensate for this problem? Does this problem also arise when training a decision tree?

2 Probabilistic kNN

Assume that you have two classes. Let N_0 be the number of exemplars in class 0, N_1 the number of exemplars in class 1 (this implies that $p(c = 0) = \frac{N_0}{N_0+N_1}$ and $p(c = 1) = \frac{N_1}{N_0+N_1}$.

Problem 6: Consider the ratio $\frac{p(c=0|\boldsymbol{x}^*)}{p(c=1|\boldsymbol{x}^*)}$. If σ^2 is very small, both numerator and denominator will be dominated by the closest data points (denoted by $\boldsymbol{x}_0$ and $\boldsymbol{x}_1$ respectively). Show that the following approximation holds:

$$\frac{p(c = 0 \mid \boldsymbol{x}*)}{p(c = 1 \mid \boldsymbol{x}*)} \approx \frac{\exp\left((-\|\boldsymbol{x}* - \boldsymbol{x}_0\|^2)/(2\sigma^2)\right)}{\exp\left((-\|\boldsymbol{x}* - \boldsymbol{x}_1\|^2)/(2\sigma^2)\right)} \tag{1}$$

Problem 7: Show that in the limit case ($\sigma \to 0$) this approximation classifies $\boldsymbol{x}^*$ as class 0 if it is closer to $\boldsymbol{x}_0$ than to $\boldsymbol{x}_1$.

Problem 8: How does σ relate to K?

3 Neighbourhood Component Analysis

Problem 9: Compute the gradient of the NCA objective as given in the slides. *This may be a very difficult excercise if you don't have much practice with math → don't spend too much time on it!* The following matrix identity[1] is helpful:

$$\frac{\partial tr(A^T AB)}{\partial A} = A(B + B^T)$$

tr is the *trace* of a matrix. You may also want to use the shorthand $x_{ij} = (x_i - x_j)$.

[1] for more of these things: K. B. Petersen and M. S. Pedersen. 2008. The Matrix Cookbook. Technical Report.