Machine Learning Exercise Sheet 2

k-Nearest Neighbors and Decision Trees

Exercise sheets consist of two parts: In-class exercises and homework. The in-class exercises will be solved and discussed during the tutorial. The homework is for you to solve at home and further engage with the lecture content. There is no grade bonus and you do not have to upload any solutions. Note that the order of some exercises might have changed compared to last year's recordings.

In-class Exercises

kNN Classification

Problem 1: You are given the following dataset, with points of two different classes:

	A	B	C	D	E	F
A	0	1.5	1.5	4.5	7	6
B	1.5	0	3	4	6.5	5.5
C	1.5	3	0	3	5.5	4.5
D	4.5	4	3	0	2.5	3.5
E	7	6.5	5.5	2.5	0	1
F	6	5.5	4.5	3.5	1	0

Name	x_1	x_2	class
A	1.0	1.0	1
B	2.0	0.5	1
C	1.0	2.5	1
D	3.0	3.5	2
E	5.5	3.5	2
F	5.5	2.5	2

	A	B	C	D	E	F
A	0	1.12	1.5	3.2	5.15	4.74
B	1.12	0	2.24	3.16	4.61	4.03
C	1.5	2.24	0	2.24	4.61	4.5
D	3.2	3.16	2.24	0	2.5	2.69
E	5.15	4.61	4.61	2.5	0	1
F	4.74	4.03	4.5	2.69	1	0

We perform 1-NN classification with leave-one-out cross validation on the data in the plot.

a) Compute the distance between each point and its nearest neighbor using L_1-norm as distance measure.

b) Compute the distance between each point and its nearest neighbor using L_2-norm as distance measure. L2的距离值普遍比L1小 Different distance measure → different result

c) What can you say about classification if you compare the two distance measures?

Problem 2: Consider a dataset with 3 classes $\mathcal{C} = \{A, B, C\}$, with the following class distribution $N_A = 16, N_B = 32, N_C = 64$. We use unweighted k-NN classifier, and set k to be equal to the number of data points, i.e. $k = N_A + N_B + N_C =: N$. 分类模糊，模型简单，对于新数据，产生预测错误的可能性增大

a) What can we say about the prediction for a new point x_{new}? ⇒ classified as class C

b) How about if we use the weighted (by distance) version of k-Nearest Neighbors?

The majority class in the neighborhood is thus equal to the majority class in the dataset.

大幅优化分类，模型复杂，学习误差降低，学习准确率提高，预测准确率提高

Problem 3: Assume you use a KNN-classifier on the following training data, that contains at least 100 samples of each class.

too many dimension
class imbalance

Acceleration	max. velocity [km/h]	PS	cylinder capacity [cm^3]	weight [kg]	class
3.6	250	600	3996	2150	car
12.5	178	150	1968	2001	van
3.5	200	113	937	227	motorcycle
...	...	...	...	...	...

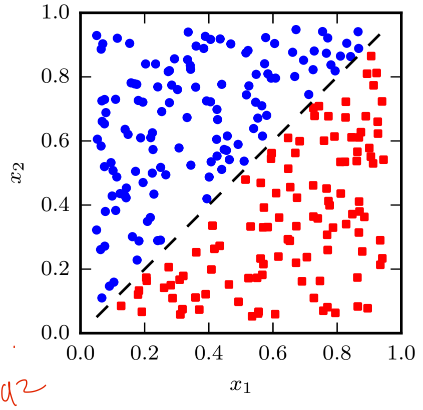
You observe that the obtained model performs poorly on the test set. What might be the problem? Name at least two possible problems and explain how you would solve them. Would a decision tree have the same problems? Justify your answer.

不同数据的量级相差太大 scaling issues，数据标准化Data standardizatio
学习集，验证集没有很好得划分
没有进行交叉验证
数据集太小
k不合适

没有问题，决策树可以处理不相关数据

Decision Trees

Problem 4: The plot below shows data of two classes that can easily be separated by a single (diagonal) line. Does there exist a decision tree of depth 1 that classifies this dataset with 100% accuracy? Justify your answer.



$x_2 > x_1$

blue red.

$x_1 = T$

2w3L 3W2L

$0 = 0.97 - \frac{1}{2} \cdot 0.97 - 0.97$

$= 0$.

$x_2 = M$

2W2L 3W4L.

$0.97 - \frac{2}{5} \cdot 1$

$- \frac{3}{5} \cdot 0.92$

$= 0.018$

$x_3 = S$

3W2L 1W4L.

$0.97 - \frac{1}{2} \cdot 0.97$

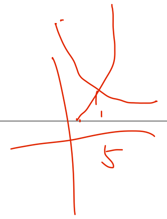
$- \frac{1}{2} \cdot 0.72 = 0.125$

Problem 5: You are developing a model to classify games at which machine learning will beat the world champion within five years. The following table contains the data you have collected.

No.	x_1 (Team or Individual)	x_2 (Mental or Physical)	x_3 (Skill or Chance)	y (Win or Lose)
1	T	M	S	W
2	I	M	S	W
3	T	P	S	W
4	I	P	C	W
5	T	P	C	L
6	I	M	C	L
7	T	M	S	L
8	I	P	S	L
9	T	P	C	L
10	I	P	C	L

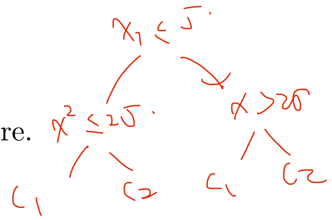
$p(y=w) = \frac{4}{10}$ $\quad p(y=L) = \frac{6}{10}$

$i_H(y) = -\sum_{i=1}^{n} \left(\frac{4}{10} \cdot \log_2 \frac{4}{10} + \frac{6}{10} \cdot \log_2 \cdot \frac{6}{10} \right) = 0.97$



a) Calculate the entropy $i_H(y)$ of the class labels y.

b) Build the optimal decision tree of depth 1 using entropy as the impurity measure.



Problem 6: Assume you have a dataset with two-dimensional points from two different classes C_1 and C_2. The points from class C_1 are given by $A = \{(i, i^2) \mid i \in \{1...100\}\} \subseteq \mathbb{R}^2$, while the points from class C_2 are $B = \{(i, \frac{125}{i}) \mid i \in \{1...100\}\} \subseteq \mathbb{R}^2$.

25 1 point $(5, 25)$

>5

Construct a decision tree of minimal depth that assigns as many data points as possible to the correct class. Provide for each split the feature and corresponding thresholds. How many and which datapoints are missclassified?

假设 $x_2 = y_2$ $x_3 = y_3$. $\Rightarrow d_1 = (x_1 - y_1)^2$.

1 Homework

$= (x_1 - y_1)^2 + |x_1 - y_1| \cdot |x_2 - y_2| + \cdot - + |x_1 - y_1||x_n - y_n| + |x_2 - y_2| \cdot |x_1 - y_1| + (x_2 - y_2)^2$.

√ 这大于式 得证.

Problem 7: You want to perform 1-kNN-classification based on

i) L_1-norm $\quad d_1(x, y)^2 = \left[|x_1 - y_1| + |x_2 - y_2| + |x_3 - y_3| + \cdot \cdot |x_n - y_n| \right]^2$

ii) L_2-norm $\quad d_2(x, y)^2 = (x_1 - y_1)^2 + (x_2 - y_2)^2 + (x_3 - y_3)^2 \; (\cdot \cdot + (x_n - y_n)^2$

Prove or disprove: The L_2-distance $d_2(\boldsymbol{x}, \boldsymbol{y}) = \left(\sum_{i=1}^d (x_i - y_i)^2\right)^{\frac{1}{2}}$ between two points $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^d$ is always smaller or equal than the L_1-distance $d_1(\boldsymbol{x}, \boldsymbol{y}) = \sum_{i=1}^d |x_i - y_i|$.

2×1

Problem 8: Prove or disprove: Consider two arbitrary points $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^2$. If $\boldsymbol{x}$ is the nearest neighbor of $\boldsymbol{y}$ regarding the L_2-norm then $\boldsymbol{x}$ is the nearest neighbor of $\boldsymbol{y}$ regarding the L_1-norm.

$\sqrt{(x-y)^2} \le \sqrt{(y-z)^2} \Rightarrow (x - y \le y - z.) \; d_1$

Programming Task

Problem 9: Load the notebook `exercise_02_notebook.ipynb` from Moodle. Fill in the missing code and run the notebook.

Note: We suggest that you use Anaconda for installing Python and Jupyter, as well as for managing packages. We recommend that you use Python 3.

For more information on Jupyter notebooks, consult the Jupyter documentation.