Announcements

Wednesday, 31 October 2018 16:01

Next week:

- Mon 10:00 – Homework discussion
- Tue 12:15 } Lecture and Practical
- Wed 16:00 }

Starting from HW # 3 (Linear regr.)

→ Deadline on SUNDAY 23:59

Decision trees

Dataset

i	$x_{i,1}$	$x_{i,2}$	$x_{i,3}$	y_i
A	5.5	0.5	4.5	2
B	7.4	1.1	3.6	0
C	5.9	0.2	3.4	2
D	9.9	0.1	0.8	0
E	6.9	-0.1	0.6	2
F	6.8	-0.3	5.1	2
G	4.1	0.3	5.1	1
H	1.3	-0.2	1.8	1
I	4.5	0.4	2.0	0
J	0.5	0.0	2.3	1
K	5.9	-0.1	4.4	0
L	9.3	-0.2	3.2	0
M	1.0	0.1	2.8	1
N	0.4	0.1	4.3	1
O	2.7	-0.5	4.2	1

$x_i \in \mathbb{R}^D$, $D=3$ – Samples

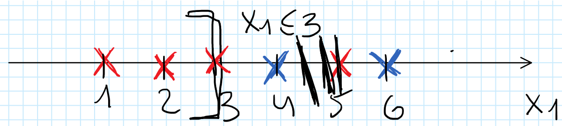
$y_i \in \{0,1,2\}$

$X \in \mathbb{R}^{N \times D}$ $N=15$

$D=3$

$y \in \{0,1,2\}^N$

Problem 1: Build a decision tree T for your data $\{\boldsymbol{X}, \boldsymbol{y}\}$. Consider all possible splits for all features 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.



$x_i \in \mathbb{R}$

$y_i \in \{\text{red}, \text{blue}\}$

$x_1 \le ?$

Split t	$\Delta i(t)$
$x_1 \le 1$	$\Delta i(x_1 \le 1)$
$x_1 \le 2$	$\Delta i(x_1 \le 2)$
$x_1 \le 3$	⋮
⋮	

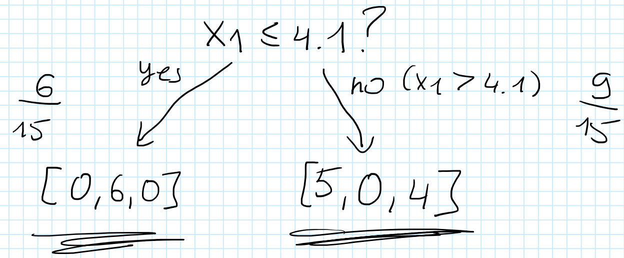
i	$x_{i,1}$	$x_{i,2}$	$x_{i,3}$	y_i
A	5.5	0.5	4.5	2
B	7.4	1.1	3.6	0
C	5.9	0.2	3.4	2
D	9.9	0.1	0.8	0
E	6.9	-0.1	0.6	2
F	6.8	-0.3	5.1	2
G	4.1	0.3	5.1	1
H	1.3	-0.2	1.8	1
I	4.5	0.4	2.0	0
J	0.5	0.0	2.3	1
K	5.9	-0.1	4.4	0
L	9.3	-0.2	3.2	0
M	1.0	0.1	2.8	1
N	0.4	0.1	4.3	1
O	2.7	-0.5	4.2	1

Back to the original problem:

Splits	Δi
$x_1 \leq 5.5$	0.1
$x_1 \leq 7.4$	0.0
$x_1 \leq 4.1$	0.296
$x_2 \leq 0.5$	0.25
$x_2 \leq 1.1$	⋮
⋮	
$x_3 \leq \dots$	

$15 \times 3 = 45$ splits to consider

Computing the information gain



$$\Delta i(x_1 \leq 4.1) = i_G(5,6,4) - \frac{6}{15}\, i_G(0,6,0) - \frac{9}{15}\, i_G(5,0,4)$$

$$= 0.658 - \frac{6}{15}\cdot 0 - \frac{9}{15}\cdot .494 = \boxed{\cancel{0.296}}$$

0.3616

$$i_G(n_0, n_1, n_2) = 1 - \sum_c \pi_c^2$$

$$= 1 - \pi_0^2 - \pi_1^2 - \pi_2^2 = 1 - \left(\frac{n_0}{n}\right)^2 - \left(\frac{n_1}{n}\right)^2 - \left(\frac{n_2}{n}\right)^2$$

Let $n = n_0 + n_1 + n_2$

then $\pi_0 = \frac{n_0}{n}$

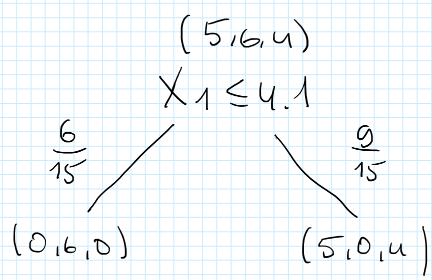
$$i_G(5,6,4) = 1 - \left(\frac{5}{15}\right)^2 - \left(\frac{6}{15}\right)^2 - \left(\frac{4}{15}\right)^2 \approx 0.658$$

$$i_G(0,6,0) = 1 - \left(\frac{0}{6}\right)^2 - \left(\frac{6}{6}\right)^2 - \left(\frac{0}{6}\right)^2 = 0$$

$$i_G(5,0,4) = 0.494$$

Split $x_1 \leq 4.1$ has the highest information gain

$\Rightarrow$ we pick it



$i_G(0,6,0) = 0$

$$\Delta i(\dots) = \underbrace{i_G(\text{root})}_{=0} - P_L \underbrace{i_G(\text{Left})}_{=0} - P_R \underbrace{i_G(\text{right})}_{=0}$$

} Left node is pure! nothing to do

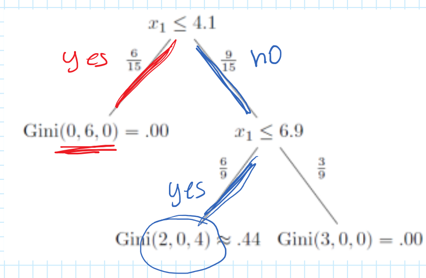
As before, for the right node, consider all possible splits

Split	Δi
$x_1 \leq 5.5$	

for the right node,
consider all possible splits

Split	Δi
$x_1 \le 5.5$	
$\vdots$	
$x_3 \le 4.2$	

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



$p(y_a \mid x_a) = [0, 1, 0]$

$\hat{y}_a = \arg\max_c p(y_a = c \mid x_a) = 1$

wit proba. 1

$p(y_b \mid x_b) = [\frac{2}{6}, 0, \frac{4}{6}]$

$\hat{y}_b = 2$ with proba. $\frac{2}{3}$

KNN (coding task)

Wednesday, 31 October 2018 10:55

Dataset

$X \in \mathbb{R}^{N \times D}$ → X_{train}, X_{test}

$y \in \{0, 1, 2\}^N$ → y_{train}, y_{test}

KNN: how to classify x?

We find k points in X_{train} that are the closest to x.

$x_1, x_2 \in \mathbb{R}^D$

$$d(x_1, x_2) = \sqrt{\sum^{D} (x_{1i} - x_{2i})^2}$$

$$d(x_1, x_2) = \sqrt{\sum_{j=1} (x_{1j} - x_{2j})^2}$$

$$\begin{array}{ccccc} [1, & 5, & 7, & 0.2, & 0.4] \\ 0 & 1 & 2 & 3 & 4 \end{array}$$

$$\begin{array}{ccc} [3 \; 4] & & \\ 0 & 1 & 2 \end{array}$$

$$\begin{bmatrix} \text{---} \; x_1 \; \text{---} \\ \text{---} \; x_2 \; \text{---} \end{bmatrix} = X$$

$$\begin{bmatrix} \text{---} \; x_1 - x \; \text{---} \\ x_2 - x \; \text{---} \end{bmatrix} = \begin{bmatrix} \sum_j (x_{1j} - x_j)^2 \\ \\ \end{bmatrix}$$

$$[y_3, \; y_4, \; y_1] \; ..$$

$$y_{test} \in \{0, 1, 2\}^{N_{test}}$$

$$[0, 1, 1, 0, 1, 2 \ldots]$$

$$\hat{y}_{test} \in \{0, 1, 2\}^{N_{test}}$$

$$[0, 0, 1, 0, 2, 2 \ldots]$$

$$acc(y, \hat{y}) = \frac{1}{N} \sum_{i=1}^{N} \mathbb{1}(y_i = \hat{y}_i)$$

$$\mathbb{1}(x) = \begin{cases} 1 & \text{if } x \text{ is true} \\ 0 & \text{else} \end{cases}$$

KNN

Wednesday, 31 October 2018 10:55

Problem 4: Classify the two vectors x_a and x_b given in Problem 2 with the k-nearest neighbors algorithm. Use $k = 3$ and Euclidean distance.

i	$x_{i,1}$	$x_{i,2}$	$x_{i,3}$	y_i
A	5.5	0.5	4.5	2
B	7.4	1.1	3.6	0
C	5.9	0.2	3.4	2
D	9.9	0.1	0.8	0
E	6.9	-0.1	0.6	2
F	6.8	-0.3	5.1	2
G	4.1	0.3	5.1	1
H	1.3	-0.2	1.8	1
I	4.5	0.4	2.0	0
J	0.5	0.0	2.3	1
K	5.9	-0.1	4.4	0
L	9.3	-0.2	3.2	0
M	1.0	0.1	2.8	1
N	0.4	0.1	4.3	1
O	2.7	-0.5	4.2	1

$x_a = [4.1, -0.1, 2.2]$

$x_b = [6.1, 0.4, 1.3]$

1) For x_{new}, find k points in the training set, who are the closest to x_{new} (in terms of euclidean dist.)

Point	Distance	Label
I	0.671	0
C	2.184	2
O	2.474	1

$$P(y_a = c \mid x_a) = \frac{\#\text{ of nbr that are class } c}{(\#\text{ of nbrs}) =: k}$$

$$P(y_a \mid x_a) = \left[\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right]$$

Because we have a tie, pick 0 (lowest label)

For x_b

Point	Distance	Label
E	1.175	2
I	1.746	0
C	2.119	2

$$P(y_b = c \mid x_b) = \begin{cases} \frac{1}{3} & \text{if } c = 0 \\ 0 & \text{if } c = 1 \\ \frac{2}{3} & \text{if } c = 2 \end{cases}$$

$\hat{y}_b = 2$ (with proba. $\frac{2}{3}$)

Problem 5: Now, consider y_i to be real-valued targets rather than classes. Perform 3-NN regression to label the vectors from Problem 2.

Recall: Regresson means that the targets $y_i \in \mathbb{R}$
(in classification they were discrete categories $y_i \in \{1, 2, \cdots, C\}$)

x_{new} KNN REGRESSION

$$\boxed{\hat{y}_{new} = \frac{1}{Z} \sum_{x_i \in N_k(x_{new})} \frac{1}{d(x_{new}, x_i)} \cdot y_i}$$

Unweighted

$$\hat{y}_{new} = \frac{1}{\cancel{Z}} \sum_{x_i \in N_k(x_{new})} 1\, y_i$$

$$\hat{y}_{new} \quad Z \quad \sum_{x_i \in N_k(x_{new})} d(x_{new}, x_i)$$

$$Z = \left[\sum_{x_j \in N_k(x_{new})} \frac{1}{d(x_{new}, x_i)} \right]^{-1}$$

$$\hat{y}_{new} = \frac{1}{\cancel{Z}\, k} \sum_{x_i \in N_k(x_{new})} 1\, y_i$$

$$Z = \sum_{x_i \in N_k(x_{new})} 1 = k$$

$\hat{y}_a = 0.561$

$\hat{y}_b = 1.396$

Problem 6: Look at the data. Which problem do you see w.r.t. building a Euclidean distance-based k-NN model on $\boldsymbol{X}$? How can you compensate for this problem? Does this problem also arise when training a decision tree?

i	$x_{i,1}$	$x_{i,2}$	$x_{i,3}$	y_i
A	5.5	0.5	4.5	2
B	7.4	1.1	3.6	0
C	5.9	0.2	3.4	2
D	9.9	0.1	0.8	0
E	6.9	-0.1	0.6	2
F	6.8	-0.3	5.1	2
G	4.1	0.3	5.1	1
H	1.3	-0.2	1.8	1
I	4.5	0.4	2.0	0
J	0.5	0.0	2.3	1
K	5.9	-0.1	4.4	0
L	9.3	-0.2	3.2	0
M	1.0	0.1	2.8	1
N	0.4	0.1	4.3	1
O	2.7	-0.5	4.2	1

1) Normalize the columns

- L_2 norm $\overset{!}{=} 1$
- min $= 0$, max $= 1$

2) Use Mahalanobis distance

$$d_M(x_1, x_2 \mid \Sigma) = (x_1 - x_2)^T \Sigma^{-1} (x_1 - x_2)$$

$x_1 \quad [0.4, 9.9]$

$x_2 \quad [-0.5, 1.1]$