



## I. [6v] Bayes and tree learning

Consider the following categorical data

	$y_1$	$y_2$	$z$
$\mathbf{x}_1$	0	1	A
$\mathbf{x}_2$	1	0	B
$\mathbf{x}_3$	0	1	B
$\mathbf{x}_4$	0	1	B
$\mathbf{x}_5$	0	0	C

- 1) [2.5v] Consider a Bayesian classifier with no independence assumption.

Classify observation  $\mathbf{x} = [0,1]^T$  using *ML* and *MAP* estimates. Show all calculus.

$$p(A) = \frac{1}{5}, p(B) = \frac{3}{5}, p(C) = \frac{1}{5}$$

$$p(y_1 = [0,1] | A) = 1, p(y_1 = [0,1] | B) = \frac{2}{3}, p(y_1 = [0,1] | C) = 0$$

Under ML assumption,  $\operatorname{argmax} p(\mathbf{x}|z) = \operatorname{argmax} \{1, \frac{2}{3}, 0\} = A$

Under MAP assumption,  $\operatorname{argmax} p(\mathbf{x}|z)p(z) = \operatorname{argmax} \{\frac{1}{5}, \frac{2}{5}, 0\} = B$

- 2) Considering the learning of a decision tree.

- a) [2v] Draw the learnt tree with no depth limit, showing the information gain calculus.

Root decision:

$$E(z|y_1) = -\frac{4}{5} \times \left( \frac{2}{4} \log \frac{2}{4} + \frac{1}{4} \log \frac{1}{4} + \frac{1}{4} \log \frac{1}{4} \right) - \frac{1}{5} \times 0 = 1.2$$
$$E(z|y_2) = -\frac{2}{5} \times \left( \frac{1}{2} \log \frac{1}{2} + \frac{1}{2} \log \frac{1}{2} \right) - \frac{3}{5} \times \left( \frac{2}{3} \log \frac{2}{3} + \frac{1}{3} \log \frac{1}{3} \right) = 0.95$$
$$IG(y_2) > IG(y_1)$$

Decision tree:

$$y_2 = 0 \Rightarrow \{y_1 = 1 \Rightarrow B; y_1 = 0 \Rightarrow C\}, y_2 = 1 \Rightarrow B$$

- b) [1.5v] Drawing the confusion matrix, compute the training sensitivity of class B.

		<i>True</i>		
		A	B	C
<i>Predicted</i>	A	0	0	0
	B	1	3	0
	C	0	0	1

$$\text{sensitivity}_B = \frac{3}{3} = 1$$

## II. [7v] Clustering and local learning

Consider the following data points

	$y_1$	$y_2$	$z$
$\mathbf{x}_1$	0	3	A
$\mathbf{x}_2$	2	1	C
$\mathbf{x}_3$	3	1	B
$\mathbf{x}_4$	2	3	B

- 3) [2v] Consider a kNN with  $k=3$ , Euclidean distance, non-uniform weights, and the weighted mode estimator. Classify observation  $\mathbf{x}_{new} = \begin{pmatrix} 2 \\ 0 \end{pmatrix}$ . Show all calculus.

$$\|\mathbf{x}_{new} - \mathbf{x}_1\|_2 = \sqrt{13}, \|\mathbf{x}_{new} - \mathbf{x}_2\|_2 = 1, \|\mathbf{x}_{new} - \mathbf{x}_3\|_2 = \sqrt{2}, \|\mathbf{x}_{new} - \mathbf{x}_4\|_2 = 3$$

$$3NN(\mathbf{x}_{new}) = \{\mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4\}$$

$$\hat{z}_{new} = \operatorname{argmax} \left\{ 0, \frac{1}{\sqrt{2}} + \frac{1}{3}, 1 \right\} = B$$

*Nota:  $\mathbf{x}_{new}$  difere em alguns enunciados*

- 4) Consider the unsupervised analysis of input variables ( $y_1$  and  $y_2$ ):

- a) [2.5v] The expectation step of EM clustering with 2 clusters produced the following estimates:

$$\begin{aligned} \pi_1 &= p(c_1) = 0.5, \quad \pi_2 = p(c_2) = 0.5 \\ p(\mathbf{x}_1|c_1) &= 0.03, \quad p(\mathbf{x}_2|c_1) = 0.1, \quad p(\mathbf{x}_3|c_1) = 0.4, \quad p(\mathbf{x}_4|c_1) = 0.03 \\ p(\mathbf{x}_1|c_2) &= 0.01, \quad p(\mathbf{x}_2|c_2) = 0.4, \quad p(\mathbf{x}_3|c_2) = 0.1, \quad p(\mathbf{x}_4|c_2) = 0.01 \end{aligned}$$

Update the *priors* by applying one maximization step of the EM clustering.

$$\gamma_{ik} = p(c_k|\mathbf{x}_i) = \frac{p(\mathbf{x}_i|c_k)p(c_k)}{p(\mathbf{x}_i)}, \quad N_k = \sum_i \gamma_{ik}, \quad \pi_k = \frac{N_k}{N}$$

$$\gamma_{11} = \frac{0.03 \times 0.5}{0.03 + 0.01} = 0.375, \gamma_{21} = 0.1, \gamma_{31} = 0.4, \gamma_{41} = 0.375$$

$$\gamma_{12} = 0.125, \gamma_{22} = 0.4, \gamma_{32} = 0.1, \gamma_{42} = 0.125$$

$$N_1 = 1.25, N_2 = 0.75$$

$$\pi_1 = 0.625, \quad \pi_2 = 0.375$$

- b) [1v] Consider  $z$  as ground truth. Compute the purity of  $(\{\mathbf{x}_1, \mathbf{x}_2\}, \{\mathbf{x}_3, \mathbf{x}_4\})$  clustering solution.

$$\text{purity} = \frac{1}{4}(1 + 2) = \frac{3}{4}$$

- c) [1.5v] Given the covariance matrix  $\Sigma = \begin{pmatrix} 1.58 & -1 \\ -1 & 1.33 \end{pmatrix}$  with two principal components, the first with eigenvector  $\begin{pmatrix} 0.75 \\ -0.66 \end{pmatrix}$ , the second with eigenvalue 0.45. How much data variability is explained by the first component?

$$C\mathbf{u}_1 = \lambda_1\mathbf{u}_1$$

Solving this, yields  $\lambda_1 \approx 2.45$ . The explained variability is thus  $\frac{\lambda_1}{\lambda_1 + \lambda_2} \approx 85\%$

### III. [7v] Perceptron and Neural networks

- 5) [2v] Consider linearly separable training data on the transformed data space  $\Phi(\mathbf{x}) = \begin{pmatrix} \|\mathbf{x}\|_1 \\ \|\mathbf{x}\|_2 \end{pmatrix}$ .

Consider a perceptron in the transformed space with weights as 1 and bias as -1. Compute one update with classic Rosenblatt's rule and  $\eta = 0.1$ , using observation  $\mathbf{x} = \begin{pmatrix} 3 \\ 4 \end{pmatrix}$  with a negative target.

$$\begin{aligned}\Phi(\mathbf{x}) &= \Phi\left(\begin{pmatrix} 3 \\ 4 \end{pmatrix}\right) = \begin{pmatrix} 7 \\ 5 \end{pmatrix} \\ \Delta w_j &= \eta(t - \hat{z}) \cdot \mathbf{x} \\ \mathbf{w} &= \begin{pmatrix} -1 \\ 1 \\ 1 \end{pmatrix} + 0.1 \left( (-1 - \text{sign}(-1 + 7 + 5)) \cdot \begin{pmatrix} 1 \\ 7 \\ 5 \end{pmatrix} \right) = \begin{pmatrix} -1.2 \\ -0.4 \\ 0 \end{pmatrix}\end{aligned}$$

- 6) [1.5v] Consider a perceptron with the following activation function:  $o = \ln((\mathbf{w}^T \mathbf{x})^2 + 1)$

Determine the gradient descent-training rule for squared error,  $E(\mathbf{w}) = \frac{1}{2} \sum_{i=1}^n (t_i - o_i)^2$

$$\begin{aligned}\frac{\partial E}{\partial w_j} &= \sum_{i=1}^n (t_i - o_i) \cdot \frac{\partial}{\partial w_j} \left( t_i - \phi \left( \sum_{j=0}^m w_j \cdot x_{ij} \right) \right) = \frac{\sum_{i=0}^n (t_i - o_i)}{(\sum_{j=0}^m w_j \cdot x_{ij})^2 + 1} \cdot \frac{\partial}{\partial w_j} \left( \left( \sum_{j=0}^m w_j \cdot x_{ij} \right)^2 + 1 \right) \\ &= \frac{\sum_{i=0}^n (t_i - o_i)}{(\sum_{j=0}^m w_j \cdot x_{ij})^2 + 1} \cdot 2 \left( \sum_{j=0}^m w_j \cdot x_{ij} \right) \cdot x_{ij} \\ \Delta w_j &= \eta \cdot 2 \cdot \frac{\sum_{i=0}^n (t_i - o_i)}{(\sum_{j=0}^m w_j \cdot x_{ij})^2 + 1} \cdot \left( \sum_{j=0}^m w_j \cdot x_{ij} \right) \cdot x_{ij} = 2\eta \cdot x_{ij} \cdot \frac{\sum_{i=0}^n (t_i - o_i)}{(net_i)^2 + 1} \cdot net_i\end{aligned}$$

- 7) Given a neural network with weights

$$W^{[1]} = \begin{pmatrix} -1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}, \quad b^{[1]} = \begin{pmatrix} 1 \\ 0 \end{pmatrix},$$

$$W^{[2]} = \begin{pmatrix} 1 & 1 \\ -1 & 1 \\ 1 & 1 \end{pmatrix}, \quad b^{[2]} = \begin{pmatrix} 1 \\ 0 \\ -1 \end{pmatrix},$$

*sigmoid* activation on the hidden layer, *softmax* activation on the output layer, and *cross-entropy* loss

(with natural logarithm). Considering observation  $\mathbf{x} = \begin{pmatrix} 2 \\ -1 \\ 1 \end{pmatrix}$  and corresponding target  $\mathbf{t} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$ :

- a) [1.5v] Is observation  $\mathbf{x}$  correctly classified? Show all your calculus.

$$net^{[1]} = W^{[1]} \mathbf{x} + b^{[1]} = \begin{pmatrix} -1 \\ 2 \end{pmatrix}, \quad \mathbf{x}^{[1]} = \sigma \begin{pmatrix} -1 \\ 2 \end{pmatrix} = \begin{pmatrix} 0.27 \\ 0.88 \end{pmatrix}$$

$$net^{[2]} = W^{[2]} \mathbf{x}^{[1]} + b^{[2]} = \begin{pmatrix} 2.15 \\ 0.61 \\ 0.15 \end{pmatrix}, \quad \mathbf{x}^{[2]} = \text{softmax} \begin{pmatrix} 2.15 \\ 0.61 \\ 0.15 \end{pmatrix} = \begin{pmatrix} 0.74 \\ 0.16 \\ 0.10 \end{pmatrix}$$

No,  $\mathbf{x}$  is not correctly classified.

- b)** [2v] Consider one stochastic gradient descent update (with learning rate of 1) using  $\mathbf{x}$ . Compute the loss and update the biases of the *output/second layer* considering.

$$L(\mathbf{t}, \mathbf{o}) = - \sum_j \mathbf{t}_j \ln(\mathbf{o}_j) = -\ln(0.16) = 0.8$$

$$\delta^{[2]} = \frac{\partial E}{\partial \mathbf{z}^{[2]}} = \frac{\partial}{\partial \mathbf{z}^{[2]}} \left( - \sum_{i=1}^d \mathbf{t}_i \log(\mathbf{x}_i^{[2]}) \right) = \mathbf{x}^{[2]} - \mathbf{t}$$

$$\mathbf{b}^{[2]} = \mathbf{b}^{[2]} - 1\delta^{[2]} = \begin{pmatrix} 1 \\ 0 \\ -1 \end{pmatrix} - \begin{pmatrix} 0.74 \\ -0.84 \\ 0.10 \end{pmatrix} = \begin{pmatrix} 0.26 \\ 0.84 \\ -0.9 \end{pmatrix}$$

**END**