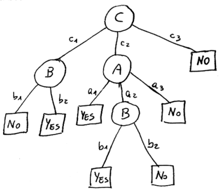
EXERCISE 1

Given a classification problem for the function $f : A \times B \times C \to \{+, -\}$, with $A = \{a_1, a_2, a_3\}, B = \{b_1, b_2\}, C = \{c_1, c_2, c_3\}$ and the following decision tree T that is the result of a learning algorithm on a given data set:



1. Provide a rule based representation of the tree T.
2. Determine if the tree T is consistent with the following set of samples $S \equiv \{s_1 = \langle a_1, b_1, c_1, No\rangle, s_2 = \langle a_2, b_1, c_2, Yes\rangle, s_3 = \langle a_1, b_2, c_3, Yes\rangle, s_4 = \langle a_2, b_2, c_2, Yes\rangle\}$. Show all the passages needed to get to the answer.

The rule based representation is:

$$(B:b_2 \wedge C:c_1) \vee (C:c_2 \wedge (A:a_1 \vee (A:a_2 \wedge B:b_1)))$$

The First sample is consistent, since $A:a_1$ makes the sample "Yes" only if $C:c_2$. δ_2 is consistent $C \xrightarrow{c_2} A \xrightarrow{a_2} B \xrightarrow{b_1} Yes$

S_3 is not consistent since $C:c_3$ gets dassified as "No".

EXERCISE 2

In Bayesian Learning, given a data set D and a hypothesis h, we can express the following relationship between the probability distributions (Bayes theorem):

$$P(h|D) = \frac{P(D|h)P(h)}{P(D)}$$

In this context:

1. define *Maximum a posteriori* (MAP) hypotheses and *Maximum likelihood* (ML) hypotheses.
2. define the concept of *Bayes Optimal Classifier*
3. discuss about practical applcability of the *Bayes Optimal Classifier*

The maximum at posteriori is $h_{MAP} = \arg\max_h \mathbb{P}(h|D) = \arg\max_h \mathbb{P}(D|h)\mathbb{P}(h)$

the max. likelihood assumes that $\mathbb{P}(h)$ is constant.

$h_{ML} = \arg\max_h \mathbb{P}(D|h)$.

The Bayes Optimal Class. uses total probability:

$c^* = \arg\max_c \mathbb{P}(c|x, D) = \arg\max_c \sum_h \mathbb{P}(c|x,h)\mathbb{P}(h|D)$. Is not practical whene we can't enumerate all the possible hypothesis h.

EXERCISE 3

1. Briefly describe the goal of linear regression and define the corresponding model.
2. Given a dataset $\mathcal{D} = \{(\mathbf{x}_1, t_1), \ldots, (\mathbf{x}_N, t_N)\}$ with $\mathbf{x}_n$ the input values and t_n the corresponding target values, explain how the parameters of the model can be estimated either in a batch or in a sequential mode.

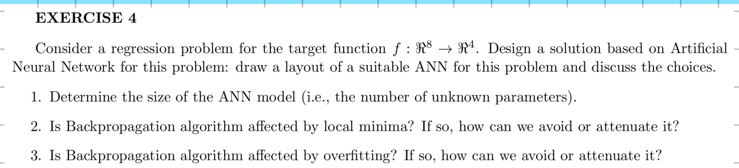
The goal of the linear regression is to approximate a real Function $f : \mathbb{R}^d \to \mathbb{R}^m$ by using a dataset $D = \{(x_i, t_i)\}_{i=1}^n$ with a model $y(x)$ that is linear in his parameter w: $y(x) = w^T \phi(x)$ where $\phi : \mathbb{R}^d \to \mathbb{R}^K$ is a non linear Function. These parameters can be estimated by considering the least-square error function:

$E(w) = \sum_{i=0}^{N} \| t_i - y(x_i) \|^2$. Let Φ to be the matrix s.t. $\Phi_{ij} = \phi_i(x_j)$ and T to be the matrix of the labels: $T_i = t_i \in \mathbb{R}^m$.

We can rewrite $E(w) = \frac{1}{2}(W\Phi - T)^T(W\Phi - T)$. We can minimize this by setting $\nabla E = 0 \Rightarrow$ optimal $W^* = \Phi^{\dagger} \cdot T$. The sequential learning approach apply the gradient descent:

For $i = 1, 2 \dots$ with $\eta \in \mathbb{R}^+$

$$w^{(i+1)} \leftarrow w^{(i)} - \eta \nabla E$$



EXERCISE 4

Consider a regression problem for the target function $f : \Re^8 \to \Re^4$. Design a solution based on Artificial Neural Network for this problem: draw a layout of a suitable ANN for this problem and discuss the choices.

1. Determine the size of the ANN model (i.e., the number of unknown parameters).
2. Is Backpropagation algorithm affected by local minima? If so, how can we avoid or attenuate it?
3. Is Backpropagation algorithm affected by overfitting? If so, how can we avoid or attenuate it?

I consider a NN with 1 hidden layer and ReLU activation Function.

Let $\alpha^{(1)} \in \mathbb{R}^8$ to be the input

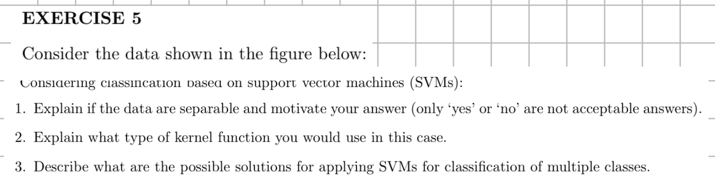
$h^{(2)} = W^{(1)}\alpha^{(1)} + b^{(1)}$ where $W^{(1)} \in Mat(10 \times 8)$ 10, rows, 8 columns and $b^{(1)} \in \mathbb{R}^{10}$

$\alpha^{(2)} = ReLU(h^{(2)})$. $h^{(3)} = W^{(2)}\alpha^{(2)} + b^{(2)}$ where $W^{(2)} \in Mat(4 \times 10)$ and $b^{(2)} \in \mathbb{R}^4$.

the output is $y = \alpha^{(2)}$. The parameters are:

$W^{(1)}, b^{(1)}, W^{(2)}, b^{(2)} \Rightarrow |\theta| = 10 \cdot 8 + 8 + 4 \cdot 10 + 4 = 132$

We clarify that the back propagation IS NOT affected by neither local minima or overfitting since, the back propagation is just an algorithm to compute the gradient of the network. If we consider the gradient descent, this is affected by local minima, and we can avoid that by considering the stochastic version (SGD), the version with momentum, and by executing the algorithm more times with different starting point. The GD is affected by overfitting since the network may be too powerful, if so, we have to reduce the number of parameters (by making smaller the hidden layer), or by adding a regularization term in the error function, that consider the sum of the module of the parameters.

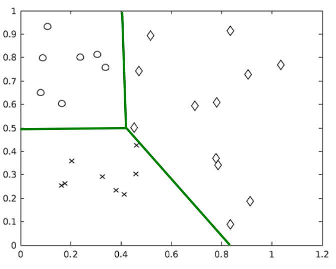


EXERCISE 5

Consider the data shown in the figure below:

Considering classification based on support vector machines (SVMs):

1. Explain if the data are separable and motivate your answer (only ‘yes’ or ‘no’ are not acceptable answers).
2. Explain what type of kernel function you would use in this case.
3. Describe what are the possible solutions for applying SVMs for classification of multiple classes.



the data is linearly separable, we can geometrically observe that there exists three linear discrimant that separates the objects in class, likes the one that i drew. Given that, i would use a linear kernel function.

EXERCISE 6

1. Describe the perceptron model for classification and its training rule.

2. Draw a graphical representation of a 2D data set for binary classification and provide a qualitative graphical example of a possible evolution of perceptron training (4 images showing a possible temporal evolution of the solution of the algorithm on the sketched data set).

A perceptron is a binary classifier $o(x) = \text{sign}(w^T x + w_0)$. The error funct. is $\frac{1}{2}\sum_{i=1}^{N}(t_i - w^T x_i + w_0)^2$ and the gradien descent is used: $w \leftarrow w - \eta \nabla E$ where

$$\frac{\partial E}{\partial w_j} = \sum_{i=1}^{N}(t_i - w^T x_i)(-x_{ij}).$$

η step size in