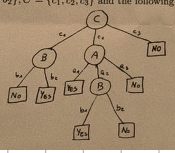
EXERCISE 1

Given a classification problem for the function $f : A \times B \times C \to \{YES, NO\}$, 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 training on some 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
   $S \equiv \{(s_1 = \langle a_2, b_1, c_2 \rangle, YES), (s_2 = \langle a_1, b_1, c_1 \rangle, NO), (s_3 = \langle a_1, b_2, c_3 \rangle, YES), (s_4 = \langle a_3, b_2, c_1 \rangle, YES), (s_5 = \langle a_3, b_1, c_2 \rangle, NO)\}$.
   Motivate your answer.
3. Compute the accuracy of T with respect to S.

The rule based repr. is:

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

The tree is not consistent with S since S_3 have $C = c_3$ and "Yes" as the label, but the tree classifies all the inputs with C_3 as "No".

Except S_3, the other samples are well classified, the accuracy is $\frac{4}{5} = 80\%$

EXERCISE 2

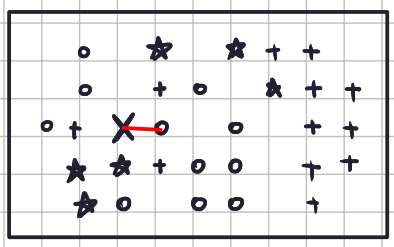
1. Describe the K-nearest neighbors (K-NN) algorithm for classification and discuss its limitations.
2. Draw a 2D classification dataset for three classes $\{circle, star, plus\}$, choose one query point close to points in the dataset belonging to different classes, and determine the answer of K-NN for such a query point for K=1, K=3, and K=5, illustrating a situation in which the answer of K-NN depends on K (i.e., you get different solutions for K=1, K=3, K=5). Motivate your answer, showing (with a graphical drawing) which instances contribute to the solution.

The KNN, given a dataset and an input x, classify x with the most popular class among the K closest point to x in D.

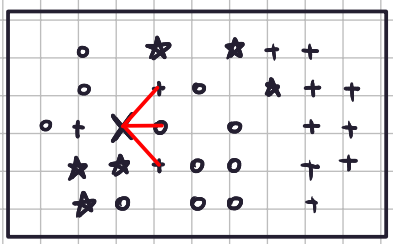
$D_K(x) = \{x_1 \cdots x_K\}$ and $\forall x' \notin D_K(x), \exists x_i \in D_K(x) : \|x - x_i\| < \|x - x'\|$

$$c(x) = \arg\max_C \sum_{x_i \in D_K(x)} \delta(class(x_i) = C).$$

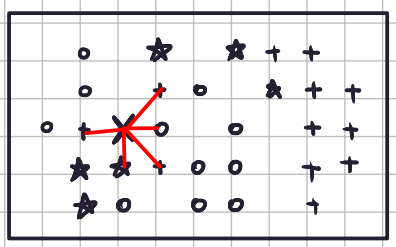
The main limitation is the fact that we must use and consider all the dataset just to make a single prediction, this is expensive in terms of time and space complexity.



K=1 X = O



K = 3 X = +



K=5 X = +

EXERCISE 3

1. Describe the difference between bagging and boosting when combining multiple learners.
2. Consider applying both bagging and boosting to a classification problem and the following models $y_1(x), \ldots, y_M(x)$ for bagging and $z_1(x), \ldots, z_M(x)$ for boosting as results of the training phase. Write down the formal equations to combine the predictions of the different models for each of the two approaches.

In bagging, we train M models with the same template hypothesis on M different dataset $\{D_i\}$ sampled from the available dataset D.

In boosting, we train sequentially M models on the same dataset, but with weighted samples

- if the model c_i failed on some points, the model c_{i+1} will give more importance to that points.

In bagging the Final prediction is:

$$\hat{y}(x) = \text{sign}\left(\frac{1}{M}\sum_{i=1}^{M} y_i(x)\right)$$

In boosting:

$$\hat{y}(x) = \text{sign}\left(\sum_{i=1}^{M} \alpha_i y_i(x)\right)$$

Where $\{\alpha_i\}$ are M coefficients used in the training process:

$J_i = \sum w_j^{(i)}(t_j - y_i(x_j))$ error for y_i

α_j = a measure for the reliability of y_i (depends on J_i)

$w_j^{(i+1)} = w_j^{(i)} \exp(\alpha_j \, \delta(y_i(x_j) \neq t_j))$

EXERCISE 4

1. Describe the principle of maximum margin used by SVM classifiers through its formal mathematical definition.
2. Draw a linearly separable dataset for 2D binary classification. Draw a possible solution obtained by SVM and highights the margin and the support vectors.
3. Discuss why the maximum margin solution is preferred for the classification problem.

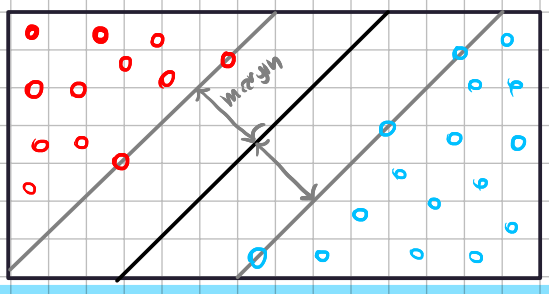
Let $D = \{(x_i, t_i)\}_{i=1}^{N}$ to be the dataset ($t_i \in \{-1, +1\}$). IF is linearly separable, then there exists w, w_0 such that $t_i(w^T x_i + w_0) \geq 0 \;\; \forall i$. We consider the closest sample to the hyperplane defined by w, w_0:

$\min_{x_i} \frac{|y(x_i)|}{\|w\|}$ ← this is the margin. We can set the opt. problem to find the w that maximizes that margin:

$$w^*, w_0^* = \underset{w, w_0}{\text{argmax}} \left(\min_{x_i} \frac{|y(x_i)|}{\|w\|} \right) = \underset{w, w_0}{\text{argmax}} \left(\min_{x_i} \frac{t_i(w^T x_i + w_0)}{\|w\|} \right) \quad \text{subj. to. } t_i y(x_i) \geq 0 \;\; \forall i$$

we can scale the points s.t. the closest to w, w_0 is at distance 1, so the problem becomes:

$$w^*, w_0^* = \underset{w, w_0}{\text{argmax}} \frac{1}{\|w\|} = \underset{w, w_0}{\text{argmin}} \|w\|^2 \quad \text{s.t.} \quad t_i y(x_i) \geq 1 \;\; \forall i$$



the SVM are preferred since it decreases the probability to missclassify new samples (thanks to the margin), respects to other methods (like perceptrons) where the discriminant found as solution is often near to the samples.

EXERCISE 5

Consider a dataset D for the classification problem $f : \mathbb{R}^5 \mapsto \{C_1, C_2, C_3\}$.

1. Describe a probabilistic generative model for such a classification problem, assuming Gaussian distributions.
2. Identify the parameters of the model and determine the size of the model (i.e., the number of independent parameters).

In the probabilistic generative model, we assume that there is a normal distr. on the samples: $\mathbb{P}(x \mid C_i) = \mathcal{N}(x; \mu_i, \Sigma)$, and there are the at priori prob. for each classes p_1, p_2, p_3 with $\sum p_i = 1$.
So we have $\mathbb{P}(x) = \sum_{i=1}^{3} p_i \mathcal{N}(x; \mu_i, \Sigma)$. The goal is to estimate the parameters of the distributions. For each class we have:

a mean μ_i : 5 parameter. p_i = 1 parameter. } ⇒ total size: 43
Then the covariance matrix Σ have 25 par. }

EXERCISE 6

Consider a two-layers ANN which receives in input real-valued vectors $\mathbf{x}$ of dimension 100 and produces in output real-valued vectors $\mathbf{y}$ of dimension 10. The hidden layer of the ANN is composed of 50 units which use the ReLU activation function. The output units use a linear activation function.

1. Compute the number of trainable parameters including the bias terms, motivating the answer (i.e., show how to compute this value)
2. Provide the formula explicitly stating how the values of $\mathbf{y}$ are computed given an input vector $\mathbf{x}$ in terms of the weight matrices and the activation functions (including the bias terms).
3. Provide a suitable loss function for training the network. Motivate your answer, answers with no motivation will not be considered.

We have: (rows, cols)

$W^{(1)} \in Mat(50 \times 100)$ $b^{(1)} \in \mathbb{R}^{50}$, given the input $x \in \mathbb{R}^{100}$ the output is:

$W^{(2)} \in Mat(10 \times 50)$ $b^{(2)} \in \mathbb{R}^{10}$ $\quad \xi(x) = W^{(2)} \, ReLU(W^{(1)} x + b^{(1)}) + b^{(2)}$

So the total number of parameters are: 5560.

The chosen loss function is the sum of square, since we know that,

$$E = \sum_{i=1}^{N} \| t_i - \xi(x_i) \|^2$$

For regression, this is equivalent to find the maximum likelihood solution, assuming zero mean Gaussian noise on the samples.