Machine Learning - 2020-2021
Universitat Pompeu Fabra

Problems 5: DECISION TREE AND ENSEMBLE MODELS

Goal

The goal of this practice is to understand how Decision Tree and Ensemble models are defined and work in supervised classification problems.

Needed Concepts

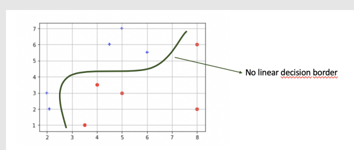
- Gini $= \sum_{l=1}^{K} p_{jl}^2$
- Gini Impurity $= 1 - \sum_{l=1}^{K} p_{jl}^2$
- Entropy $= \sum_{l=1}^{K} * log_2 p_{jl}$
- Information Gain $= 1 - \sum_{l=1}^{K} * log_2 p_{jl}$
- Weighted Gini impurity $= 1 - \sum_{j=1}^{L} Wj * \sum_{l=1}^{K} * log_2 p_{jl}$

Exercise 1

1. **Decision Tree:** Consider a training dataset with two classes: the class $C_1 = \{\mathbf{x}_1 = \binom{2}{3}, \mathbf{x}_2 = \binom{2.1}{2}, \mathbf{x}_3 = \binom{4.5}{6}, \mathbf{x}_6 = \binom{5}{7}, \mathbf{x}_8 = \binom{6}{5.5}\}$ with a label 1 and the class $C_2 = \{\mathbf{x}_4 = \binom{4}{3.5}, \mathbf{x}_5 = \binom{3.5}{1}, \mathbf{x}_7 = \binom{5}{3}, \mathbf{x}_9 = \binom{8}{6}, \mathbf{x}_10 = \binom{8}{2}\}$ with label -1.

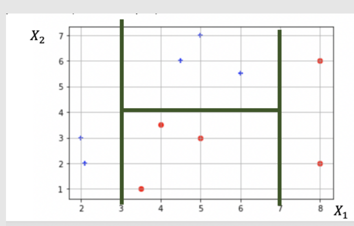
   (a) Plot the points. Is it feasible to find a linear "decision border" to classify both classes?

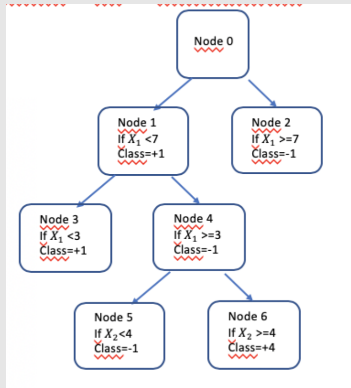
   Solution:

   (b) Draw the node splits decision (or decision stumps) a decision tree in the scatter plot with the following hyperparameters: Minimum samples for a node split= 2 Minimum samples for a terminal node or leaf=2 Maximum depth of tree (vertical depth)=3

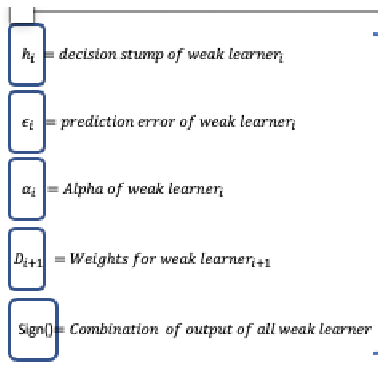
   Solution:

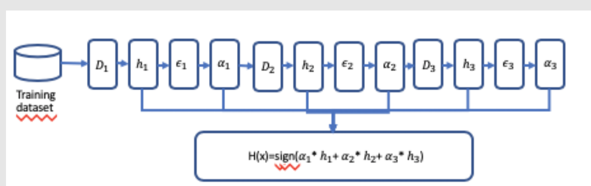


2. **Adaboost:** Using the Decision stump from previous exercise, develop an adaboost ensemble architecture with four week learners.

   (a) Considering the following pseudo-blocks, build an architecture for a 4-estimator (weak learners) adabost ensemble:



Solution:



   (b) First round:

   Consider the decision stump of figure 1, build a table with the actual class, the weight, prediction, loss and weight*loss for every datapoint

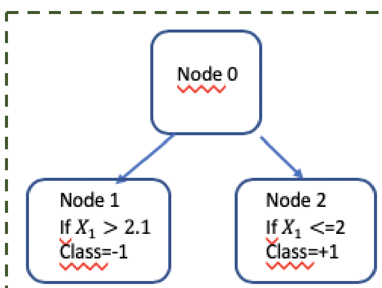


Figure 1

   Calculate the error and alpha

   Update the weights and normalized weights for every datapoint for next round

Solution:

$X_1$0	X_2	y=Actual class	D_1	t=Prediction	Loss	D_1*loss	D_2	Norm_D_2
2	3	1	0.1	1	0	0	0.065	0.071
2.1	2	1	0.1	1	0	0	0.065	0.071
4.5	6	1	0.1	-1	1	0.1	0.153	0.167
4	3.5	-1	0.1	-1	0	0	0.065	0.071
3.5	1	-1	0.1	-1	0	0	0.065	0.071
5	7	1	0.1	-1	1	0.1	0.153	0.167
5	3	-1	0.1	-1	0	0	0.065	0.071
6	5.5	1	0.1	-1	1	0.1	0.153	0.167
8	6	-1	0.1	-1	0	0	0.065	0.071
8	2	-1	0.1	-1	0	0	0.065	0.071

$$\boxed{\epsilon_1 = \sum_{i=1}^{10} D_1(i) * loss(i) = 0.3}$$

$$\boxed{\alpha_1 = \frac{1}{2} ln\left[\frac{(1-\epsilon_1)}{\epsilon_1}\right] = \frac{1}{2} ln\left[\frac{(1-0.3)}{0.3}\right] = 0.42}$$

(c) Second round:

Plot the points which sizes should be aligned with NormD2(i) value

Consider the decision stump of figure 2, build a table with the actual class, the weight, prediction, loss and weight*loss for every datapoint

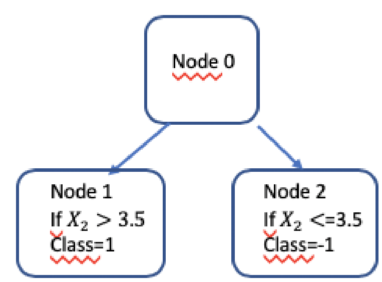
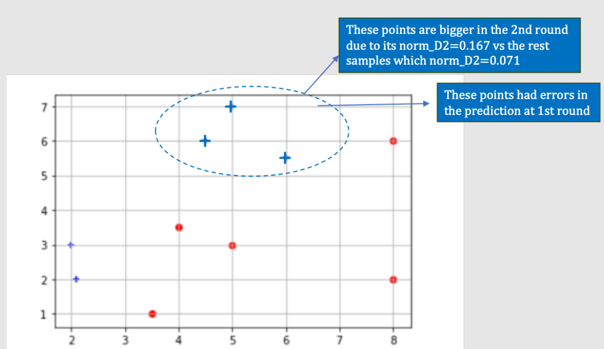


Figure 2

Calculate the error and alpha

Update the weights and normalized weights for every datapoint for next round

Solution:



X_1	X_2	y=Actual class	Norm_D_2	t=Prediction	Loss	Norm_D_2 *loss	D_3	Norm_D_3
2	3	1	0.071	-1	1	0.071	0.136	0.167
2.1	2	1	0.071	-1	1	0.071	0.136	0.167
4.5	6	1	0.167	1	0	0	0.087	0.106
4	3.5	-1	0.071	-1	0	0	0.037	0.045
3.5	1	-1	0.071	-1	0	0	0.037	0.045
5	7	1	0.167	1	0	0	0.087	0.106
5	3	-1	0.071	-1	0	0	0.037	0.045
6	5.5	1	0.167	1	0	0	0.087	0.106
8	6	-1	0.071	1	1	0.071	0.137	0.167
8	2	-1	0.071	-1	0	0	0.037	0.045

$$\epsilon_2 = \sum_{i=1}^{10} norm\_D_2(i) * loss(i) = 0.071 + 0.071 + 0.071 = 0.21$$

$$\alpha_2 = \frac{1}{2} ln\left[\frac{(1-\epsilon_2)}{\epsilon_2}\right] = \frac{1}{2} ln\left[\frac{(1-0.21)}{0.21}\right] = 0.65$$

(d) Third round:

Plot the points which sizes should be aligned with NormD3(i) value

Consider the decision stump of figure 3, build a table with the actual class, the weight, prediction, loss and weight*loss for every datapoint

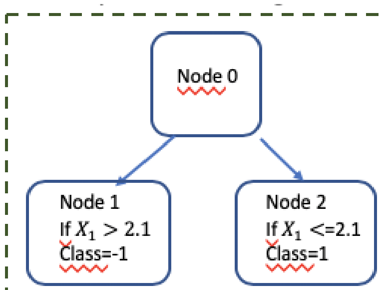
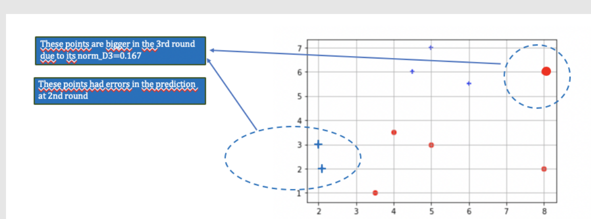


Figure 3

Calculate the error and alpha

Update the weights and normalized weights for every datapoint for next round

Solution:



X_1	X_2	y=Actual class	Norm_D_3	t=Prediction	Loss	Loss*norm_ D_3	D_4	Norm_D_4
2	3	1	0.167	1	0	0	0.136	0.122
2.1	2	1	0.167	1	0	0	0.136	0.122
4.5	6	1	0.106	-1	1	0.106	0.087	0.167
4	3.5	-1	0.045	-1	0	0	0.037	0.033
3.5	1	-1	0.045	-1	0	0	0.037	0.033
5	7	1	0.106	-1	1	0.106	0.087	0.167
5	3	-1	0.045	-1	0	0	0.037	0.033
6	5.5	1	0.106	-1	1	0.106	0.087	0.167
8	6	-1	0.167	-1	0	0	0.137	0.122
8	2	-1	0.045	-1	0	0	0.037	0.033

$$\epsilon_3 = \sum_{i=1}^{10} norm\_D_3(i) * loss(i) = 0.106 + 0.106 + 0.106 = 0.31$$

$$\alpha_3 = \frac{1}{2} ln\left[\frac{(1-\epsilon_3)}{\epsilon_3}\right] = \frac{1}{2} ln\left[\frac{(1-0.31)}{0.31}\right] = 0.38$$

(e) Four round:

Plot the points which sizes should be aligned with NormD4(i) value

Consider the decision stump of figure 4, build a table with the actual class, the weight, prediction, loss and weight*loss for every datapoint

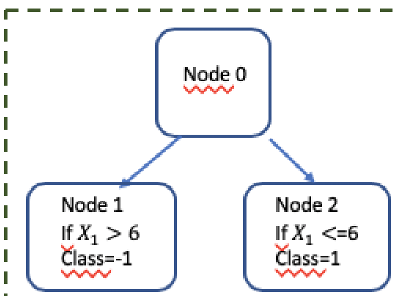
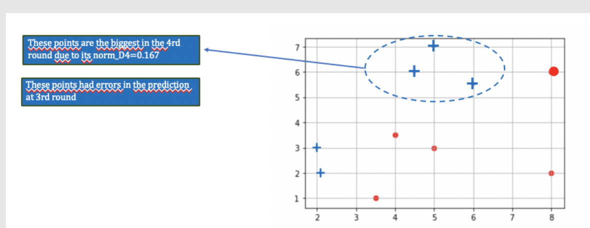


Figure 4

Calculate the error and alpha

Update the weights and normalized weights for every datapoint for next round

Solution:



X_1	X_2	y=Actual class	Norm_D_4	t=Prediction	Loss	Loss*norm_ D_4	D_5	Norm_D_5
2	3	1	0.122	1	0	0	0.041	0.068
2.1	2	1	0.122	1	0	0	0.041	0.068
4.5	6	1	0.167	1	0	0	0.056	0.093
4	3.5	-1	0.033	1	1	0.033	0.100	0.167
3.5	1	-1	0.033	1	1	0.033	0.100	0.167
5	7	1	0.167	1	0	0	0.056	0.093
5	3	-1	0.033	1	1	0.033	0.100	0.167
6	5.5	1	0.167	1	0	0	0.056	0.093
8	6	-1	0.122	-1	0	0	0.041	0.068
8	2	-1	0.033	-1	0	0	0.011	0.019

$$\boxed{\epsilon_4 = \sum_{i=1}^{10} norm\_D_4(i) * loss(i) = 0.033 + 0.033 + 0.033 = 0.10}$$

$$\boxed{\alpha_4 = \frac{1}{2} ln\left[\frac{(1-\epsilon_4)}{\epsilon_4}\right] = \frac{1}{2} ln\left[\frac{(1-0.10)}{0.10}\right] = 1.10}$$

(f) Calculate the prediction for $C_1 = \{\mathbf{x}_1 = \binom{2}{3}\}$:

Solution:

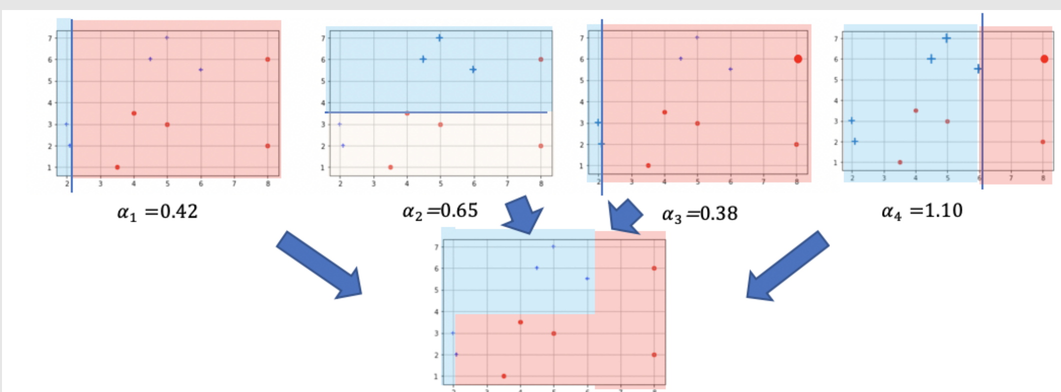
$H(x^{(1)}) = sign(0.42 * h_1(x^{(1)}) + 0.65 * h_2(x^{(1)}) + 0.38 * h_3(x^{(1)}) + 1.10 * h_4(x^{(1)}))$

$H(x^{(1)}) = sign(0.42 * 1 + 0.65 * (-1) + 0.38 * 1 + 1.10 * 1) = sign(1.25) = 1$

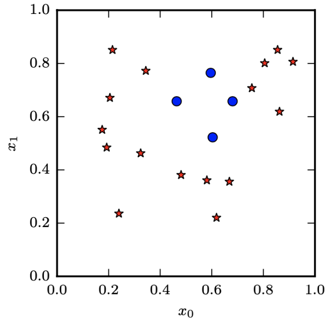
$\boxed{\text{The prediction for } x^{(1)} = \binom{2}{3} \text{ is class 1}}$

(g) Draw the decision areas in the Adaboost classifier:

Solution:



3. **CART:** Consider the following scatter plot of two classes (blue and red blobs) based on two variables x0 and x1.



(a) Calculate the Gini and Gini impurity values of the current distributions

Solution:

$Gnode0 = (\frac{4}{20})^2 + (\frac{16}{20})^2 = 0.68$

$G0impurity = 1 - Gnode0 = 1 - 0.68 = 0.32$

(b) Consider x0=0.4 as split value to create subnodes. Plot the selected split value and calculate the Ginis and Ginis impurity of the new subnodes. Which is the selected class in both nodes? Calculate the Weighted Gini for x0=0.4.

Solution:

$Gnode1 = Gleft = (\frac{7}{7})^2 + (\frac{0}{7})^2 = 1$

$G1impurity = 1 - Gnode1 = 1 - 1 = 0$

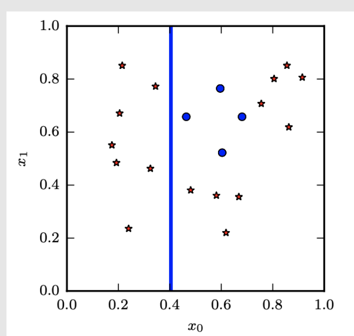
The selected class in node 1 is red class because it is the only class in the subnode

$Gnode2 = Gright = (\frac{4}{13})^2 + (\frac{9}{13})^2 = 0.5732$

$G2impurity = 1 - Gnode2 = 1 - 0.5732 = 0.426$

The selected class in this node 2 is red class because there is 9 red blobs vs 4 blue blobs

$WeightedGinix0_{0.4} = (\frac{7}{20}) * Gnode1 + (\frac{13}{20}) * Gnode2 = 0.72$



(c) Define a new split value in x0. Plot the selected split value and calculate the Ginis and Ginis impurity of the new subnodes. Which is the selected class in both nodes? Calculate the Weighted Gini for this new split.

Solution:

For x0=0.7:

$Gnode3 = Gleft = (\frac{4}{8})^2 + (\frac{4}{8})^2 = 0.5$

$G3impurity = 1 - Gnode3 = 1 - 0.5 = 0.5$

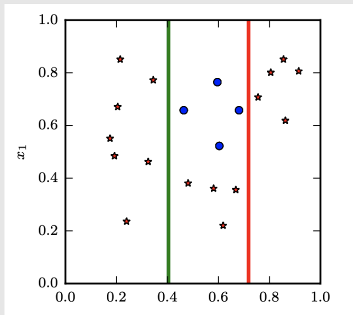
The selected class in node 3 is red or blue class because both classes have the same number of blobs

$Gnode4 = Gright = (\frac{5}{5})^2 + (\frac{0}{5})^2 = 1$

$G4impurity = 1 - Gnode4 = 1 - 1 = 0$

The selected class in node 4 is red because it is the only class

$WeightedGinix0_{0.7} = (\frac{8}{13}) * Gnode3 + (\frac{5}{13}) * Gnode4 = 0.68$



(d) Define a split value in x1. Plot the selected split value and calculate the Ginis and Ginis impurity of the new subnodes. Which is the selected class in both nodes? Calculate the Weighted Gini for this new split.

Solution:

For x1=0.5:

$Gnode5 = Gleft = (\frac{4}{4})^2 + (\frac{0}{4})^2 = 1$

$G5impurity = 1 - Gnode5 = 1 - 1 = 0$

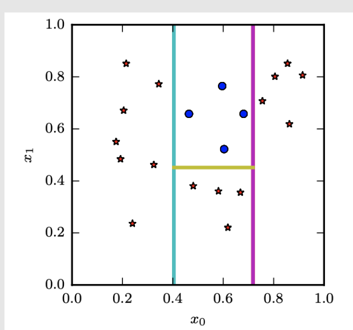
The selected class in node 5 is blue because it is the only class

$Gnode6 = Gright = (\frac{4}{4})^2 + (\frac{0}{4})^2 = 1$

$G6impurity = 1 - Gnode6 = 1 - 1 = 0$

The selected class in node 6 is red because it is the only class

$WeightedGinix1_{0.5} = (\frac{4}{8}) * Gnode5 + (\frac{4}{8}) * Gnode6 = 1$



(e) Calculate the Entropy and Information Gain in Node 0 and Node 4.

Solution:

For Node 0:

$Entropy = -p * log_2(p) - q * log_2(q) = -\frac{4}{20} * log_2(\frac{4}{20}) - \frac{16}{20} * log_2(\frac{16}{20}) = 0.72$

Information Gain node 0=$1 - Entropy = 1 - 0.72 = 0.28$

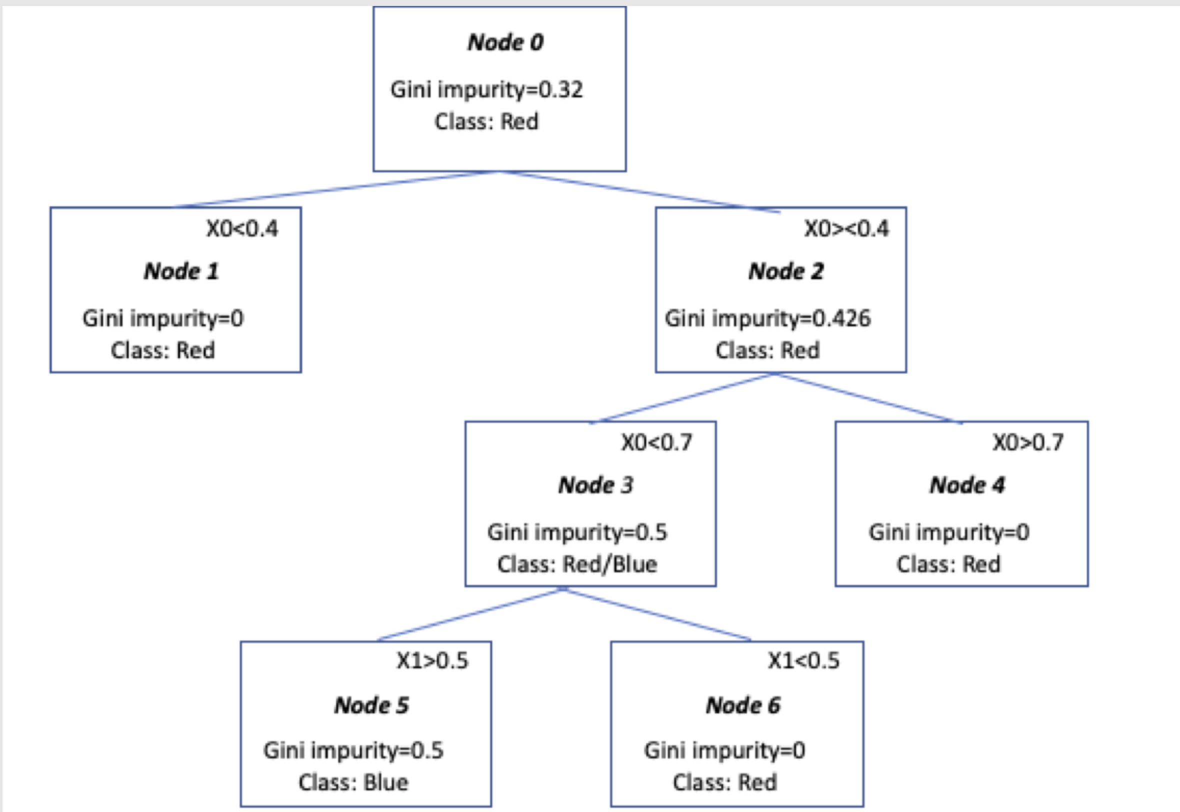
For Node 4:

$Entropy = (-p * log_2(p) - q * log_2(q) = 0$

Information Gain node 4=$1 - Entropy = 1 - 0 = 1$

(f) Plot the diagram of the full decision tree with node and subnodes.

Solution:



(g) Which is the predicted class for a new blob with value $\{\mathbf{x}_{new} = \binom{0.8}{0.2}\}$?

Solution: The predicted class is red because it will be in node 4

4. (To be executed in Jupyter Notebook) In this exercise we will execute a decision tree with the Sklearn library for the Iris multiclassification. We will use the graphviz library to evaluate the node split.

5. (To be executed in Jupyter Notebook) In this exercise we will execute change the decision tree for an adaboost ensemble for the same dataset in the previous exercise.