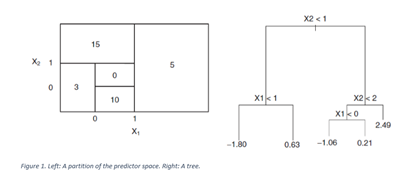
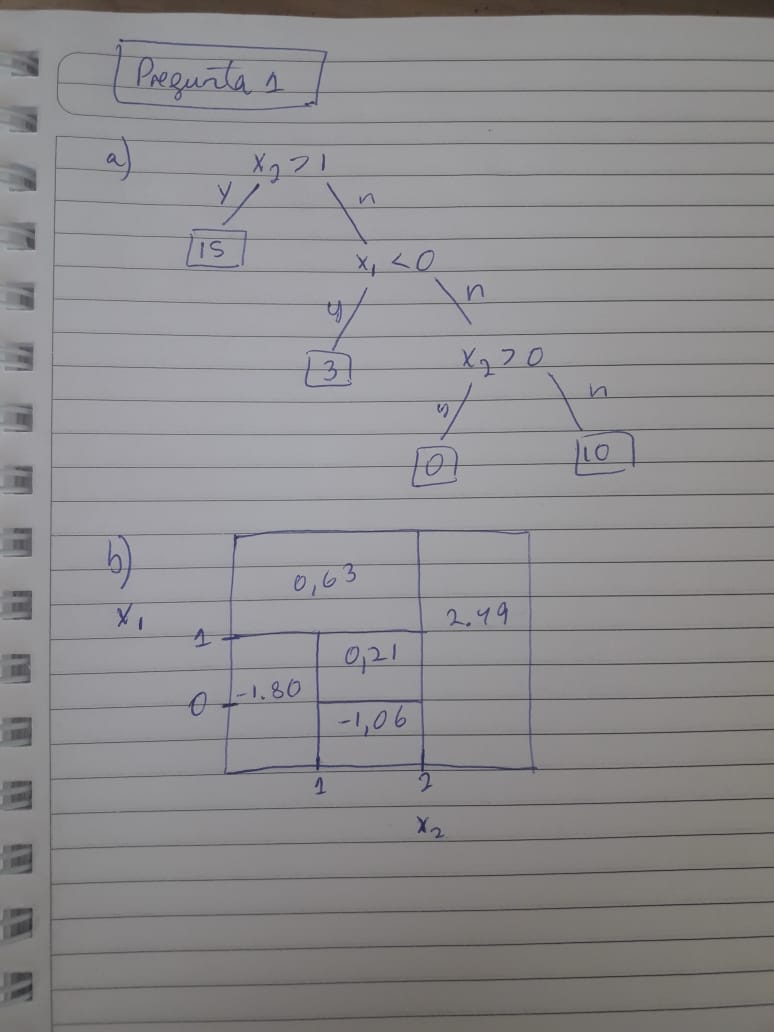
**1. (2 pts) This question relates to the plots in Figure 1.**

**a) Sketch the tree corresponding to the partition of the predictor space illustrated in the left-hand panel of Figure 1. The numbers inside the boxes indicate the mean of 𝑌 within each region.**

**b) Create a diagram similar to the left-hand panel of Figure 1, using the tree illustrated in the right-hand panel of the same figure. You should divide up the predictor space into the correct regions, and indicate the mean for each region.**

****



**2. (2 pts) Consider the Gini index and classification error (a.k.a misclassification) in a simple classification setting with two classes. Create a single plot that displays each of these quantities as a function of 𝑝1. The 𝑥-axis should display 𝑝1, ranging from 0 to 1, and the 𝑦-axis should display the value of the Gini index, classification error. Hint: In a setting with two classes, 𝑝1 = 1− 𝑝2.**

****

**3. (2 pts) Suppose we produce ten bootstrapped samples from a data set containing red and green classes. We then apply a classification tree to each bootstrapped sample and, for a specific value of X, produce 10 estimates of P(Class is Red|X): 0.1, 0.15, 0.2, 0.2, 0.55, 0.6, 0.6, 0.65, 0.7, and 0.75. what is the final classification under a) hard voting, b) soft voting.**

a)valores por encima de 0.50: 6 de 10, por tanto se clasifica como un resultado positivo

b) soft voting: (0.1+ 0.15+0.2+0.2+ 0.55+ 0.6+0.6+0.65+ 0.7+ 0.75)/10=0.45

0.45\*100=45%

**4. What is the most important variable in the decision trees of Figure 1?**

La variable X2

**5. In Ensemble methods, what’s the difference between bagging and boosting?**

Bagging y boosting son similares en que ambas son técnicas de conjunto. Primeramente, para usar Bagging o Boosting, se debe seleccionar un algoritmo base de aprendizaje (ej. Árbol de clasificación), donde ambas técnicas consistirían en un grupo de árboles tan grandes como se quiera.

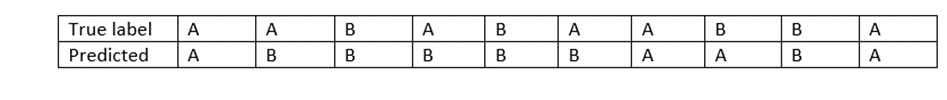
En los algoritmos Boosting, cada clasificador está entrenado en datos, teniendo en cuenta el éxito de los clasificadores anteriores. Después de cada paso de entrenamiento, los pesos se redistribuyen; para aumentar la clasificación de los datos se aumentan sus pesos para enfatizar los casos más difíciles. Mientras que en Bagging, el resultado se obtiene promediando las respuestas de los N datos (o voto mayoritario)

Si el problema es que el modelo único obtiene un rendimiento muy bajo, Boosting podría generar un modelo combinado con errores más bajos, ya que optimiza las ventajas y reduce las trampas del modelo único. Por el contrario, si la dificultad del modelo único es un ajuste excesivo, entonces Bagging es la mejor opción.

**6. Explain the Random Forest algorithm.**

El algoritmo de bosque aleatorio es un algoritmo de clasificación supervisado, este algoritmo crea el bosque con varios árboles (cuantos más arboles haya en el bosque se obtienen resultados de alta precisión). Dado el conjunto de datos de entrenamiento con objetivos y características, el algoritmo del árbol de decisión presentará un conjunto de reglas. Se pueden usar las mismas reglas establecidas para realizar la predicción en el conjunto de datos de prueba; el clasificador de bosque aleatorio manejará los valores faltantes; también se puede modelar el clasificador de bosque aleatorio para valores categóricos.

**7. (3 pts) To evaluate the performance of a classifier, compute the confusion matrix, recall, precision and overall accuracy using the values in the following table, (programa en python)**



**8. What’s the difference between DBSCAN and 𝐾-means**

En resumen, KMeans es una técnica de agrupación basada en la distancia donde, dependiendo de la distancia entre los puntos de datos. En kmeans, inicializa los centros del grupo y luego encuentra la distancia entre cada punto y cada uno del grupo y luego agrupa los puntos a sus centros más cercanos. Aquí el problema de optimización que resolvemos es encontrar el número de clústeres de tal manera que la suma de distancias desde cada punto y su clúster más cercano se minimice. Las desventajas de kmeans son: intenta crear un clúster del mismo tamaño sin importar cómo se dispersen, no funciona bien para estructuras no globulares y no le importa cuán densos estén los datos.

DBSCAN resuelve algunos de los problemas de kmeans trabajando con la densidad de puntos. Este es un método basado en la densidad. La suposición principal de DBSCAN es que dos regiones densas están separadas por una región dispersa. Dado que DBSCAN funciona con densidad, puede modelar fácilmente estructuras no globulares.

La principal diferencia es que funcionan de manera completamente diferente y resuelven diferentes problemas. Kmeans es una optimización de mínimos cuadrados, mientras que DBSCAN encuentra regiones conectadas a la densidad.

**9. Is a node’s Gini impurity generally lower or greater than its parent’s? Is it generally lower/greater, or always lower/greater?**

Generalmente es más baja que la de su padre, esto es debido a la función de costo del algoritmo de entrenamiento, ya que este divide cada nodo, minimizando la suma ponderada de las impurezas de Gini de sus hijos.

**10. What are the parameters of a fully connected neural network architecture?**

El sistema de red neuronal está compuesto por: neurona, capa y red.

Los elementos que constituyen dicha neurona son: entradas, pesos sinápticos, regla de propagación, función de transferencia y la función de salida

**11. If a Decision Tree is underfitting the training set, is it a good idea to try scaling the input features?**

A los árboles de decisiones no les importa si los datos de entrada están escalados o no, por tanto hacerlo no marcaría ninguna diferencia en el resultado .

**12. Compute the Gini index for the following**

**a) {𝐴:3,𝐵:8,𝐶:7}**

**b) {𝐴:2,𝐵:2,𝐶:16}**