Jose Alberto Gonzalez Arteaga A01038061
Jesús Guillermo Falcón Cardona
Machine Learning (Gpo 2)

Decision Tree

Un árbol de decisión es un modelo que nos ayuda a la predicción y clasificación de los datos. La finalidad de esta practica fue hacer un análisis de un arbol de decision, entrenarlo con datos y ver como se comportaba. Así mismo, una de las funciones fue poder ver como eran separados los datos de una manera a detalle más gráfica.

En la primera parte de la practica se realizaron las configuraciones correspondientes, como los imports, ajustar parametros para el guardado de las imagenes creadas, así como las funciones creadas en la antigua practica para hacer la Regresion Logistica con el metodo de Batch Gradient Descent.

1. Iris dataset

Con la finalidad de clasificar en las siguientes categorías: setosa, versicolor y virgina, mediante los parametros: sepal length, sepal width, petal length, petal width. Algunos de los puntos importantes de la implementación con este modelo fueron:

- Se excluyeron los parametros de sepal length y sepal width para generar unos datos binarios y poder hacer el analisis de la toma de decisiones del algoritmo correctamente.
- En la implementación del modelo se utilizó el parametro de max_depth=2 para evitar el overfit de los datos. También tomando en cuenta la distribución de los datos en cada uno de las hojas del arbol, si notaba que solo entraban 2 o 3 valores, no era necesario hacer un nivel más especifico.
- Al calcular la precisión del modelo se acumuló un 0.98 de efectividad el cual es un buen resultado para el modelo.
- El gráfico generado por la función boundaries plot nos ayudó a visualizar como es que dependiendo de los datos de width y length hace la clasificación de los niveles del arbol.
- Al tener 3 clasificaciones diferentes, no se realizó le regresión logistica ni del algoritmo programado ni la generada por SciKit-Learn.

2. Wine dataset

Con la finalidad de clasificar la calidad del vino en las siguientes categorías: quality0, quality1, quality2, mediante los parametros: fixed_acidity, volatile_acidity, critic_acid, residual_sugar, chlorides, free_sulfur_dioxide, total_sulfur_dioxide, density, pH, sulphates and alcohol. Algunos de los puntos importantes de la implementación con este modelo fueron:

- Hacer un split de los datos destinando un .33 a testing y .67 a los datos de entrenamiento.
- En la implementación del modelo se utilizó el parametro de max_depth=2 para evitar el overfit de los datos. También tomando en cuenta la distribución de los datos en cada uno de las hojas del arbol, si notaba que solo entraban 2 o 3 valores, no era necesario hacer un nivel más especifico. Apesar de notar que con un nivel más el score aumentaba, las hojas se empezaron a ser muy especificas y eso perjudica a nuestro modelo para futuras predicciones.
- El score obtenido fue de 0.88, el cual es bueno considerando que se utilizaron una profundidad de 2 para generalizar el modelo.
- Al tener 3 clasificaciones diferentes, no se realizó le regresión logistica ni del algoritmo programado ni la generada por SciKit-Learn.

3. Breast Cancer dataset

Con la finalidad de clasificar si un paciente tiene o no cancer mediante los parametros: 'mean radius', 'mean texture', 'mean perimeter', 'mean area', 'mean smoothness', 'mean compactness', 'mean concavity', 'mean concave points', 'mean symmetry', 'mean fractal dimension', 'radius error', 'texture error', 'perimeter error', 'area error', 'smoothness error', 'compactness error', 'concavity error', 'concave points error', 'symmetry error', 'fractal dimension error', 'worst radius', 'worst texture', 'worst perimeter', 'worst area', 'worst smoothness', 'worst compactness', 'worst concavity','worst concave points', 'worst symmetry', 'worst fractal dimension'. Algunos de los puntos importantes de la implementación con este modelo fueron:

- Hacer un split de los datos destinando un .33 a testing y .67 a los datos de entrenamiento.

- En la implementación del modelo se utilizó el parametro de max_depth=3 para evitar el overfit de los datos. Se optó por utilizar 3 porque con 2 se obtenía un score bajo y al ver que las hojas generadas por 3 sí tenían una distribución considerable, se optó por ser un buen parametro.
- Con dicho modelo se obtuvo un score de 0.92 el cual es muy bueno para el modelo.
- En este dataset sí se pudo hacer el uso del modelo de regresión logistica programada la practica anterior porque solamente se tenían dos clasificaciones. Los parámetros con mejores resultados fueron un b0 de 0.5, un alpha de 0.05, maximo de iteraciones de 1000 y un threshold de 0.0001 generando un score de 0.962 superando el uso del modelo anterior.
- Se realizó la regresión logística con el modelo de SciKit-Learn, se tuvo que poner un maximo de iteraciones (10000) porque sino, no lograba converger. Al final se pudo obtener un score de 0.968. Este modelo fue el más optimo, ganandole ligeramente al de Batch Gradient Decendent, probablemente esto se deba a los parametros ya establecidos por el modelo de SciKit-Learn y su capacidad de hacer las operaciones optimizadas.