Práctica no. 5: Árboles de Decisión

Jair Antonio Bautista Loranca
a01365850@itesm.mx
Tecnológico de Monterrey
Monterrey, N.L., México

Maximiliano Zambada Camacho
a01570146@itesm.mx
Tecnológico de Monterrey
Monterrey, N.L., México

RESUMEN

Así como hemos estado trabajando anteriormente, las técnicas de Machine Learning varían dentro de sus sub divisiones de aprendizaje supervisado y no supervisado, donde en el supervisado es necesario un entrenamiento de las variables a utilizar para poder realizar adecuadamente el modelo de predicción deseado, ya sea de regresión lineal o logística, árboles de decisión, entre otros.

ACM Reference Format:
Jair Antonio Bautista Loranca and Maximiliano Zambada Camacho. 2021. Práctica no. 5: Árboles de Decisión. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1. INTRODUCCIÓN

Continuando con el trabajo que se ha realizado en prácticas anteriores para profundizar en las diferentes técnicas de aprendizaje automático, seguimos con modelos donde es necesario entrenar las variables, como los árboles de decisión. Asimismo, estaremos manejando datasets ya integrados dentro de Scikit Learn, como los de Iris, de Wine, y de Breast Cancer.

2. CONCEPTOS PREVIOS

Los árboles de decisión son un algoritmo de aprendizaje automático que es ampliamente utilizado para poder procesar grandes volúmenes de datos, y así poder llegar a soluciones de problemas.

. En esencia, este algoritmo es estadístico, ya que nos permiten crear modelos predictivos de análisis de datos más que nada para Big Data. Se centran principalmente en su clasificación según algunas características, o bien, en la regresión mediante la relación entre varias variables para poder predecir los valores.

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

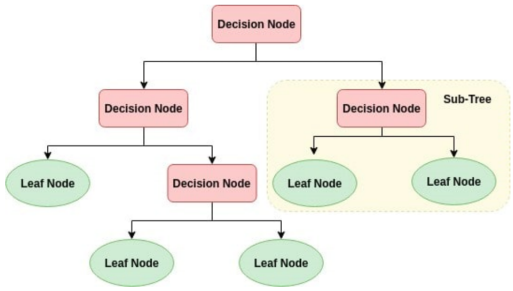

Figura 1: Árbol de Decisión con sus conceptos

. El concepto detrás de los árboles de decisión es seleccionar el mejor atributo usando Medidas de Selección de Atributos para dividir los registros, y así hacer que ese atributo sea un nodo de decisión y divida el conjunto de datos en subconjuntos más pequeños.

. Estas Medidas de Selección son esenciales para poder seleccionar el criterio que divide los datos. Primero que nada, está el cálculo para la Ganancia de Información:

$$Ganancia(S, A) = Entropia(S) - \sum_{Values(A)} \frac{|S_v|}{|S|} * Entropia(S_r)$$

. Esta Ganancia es una propiedad estadística que mide qué tan bien un atributo separa los Training Sets de acuerdo con su clasificación. Se puede ver que dentro de la fórmula estamos usando entropías, las cuales también se calculan con la siguiente fórmula:

$$Entropia(S) = -\frac{p}{N} log_2 \frac{p}{N} - \frac{n}{N} log_2 \frac{n}{N}$$

. En la fórmula, las p son los atributos positivos, mientras que las n son los negativos, y N representa el número total de atributos. La idea es que la entropçia es la impureza de un grupo de datos. Por ende, la ganancia es una disminución de la entropía.

3. METODOLOGÍA

Para el desarrollo de esta práctica se trabajo de manera colaborativa, para el desarrollo de los modelos solicitados. Específicamente trabajamos a la par, uno realizando los procesos de pruebas con el código para saber las fórmulas a utilizar mientras que el otro buscaba soluciones a los problemas encontrados a lo largo del proceso. Estos roles se fueron turnando, consiguiendo así un desarrollo ágil y efectivo de la práctica.

. Al empezar con la práctica, el primer paso fue lo esencial, la creación del archivo, y pasar todo el código ya proporcionado en las instrucciones, tanto para la importación de las librerías necesarias

para la graficación, así como el código para generar los .dot y árbol de decisión.

. Uno de los pasos que pasamos por alto al principio y nos trajo ciertos problemas, fue la instalación de Graphviz, ya que es un paso esencial, ya que no viene por default con python. Después de instalarla, pudimos comenzar a realizar código y pruebas.

. Primeramente, completamos el código faltante con la importación del dataset de Iris, ya incluido dentro de la librería de Scikit Learn. Al ya estar incluido, nos facilitó la tarea de poder dividir los datos en X y y, y después en sus respectivos sets de Entrenamiento y de Testing. Después de eso, simplemente fue entrenarlos con el uso de la función de DecisionTreeClassifier(). Esto nos trajo lo necesario para poder realizar el resto del proceso con graphviz, así como calcular la precisión por medio de la función de score(). Finalmente, al tener generado el archivo de Decision Tree, así como su archivo .dot, se necesitó convertir el archivo dot a png con la Terminal con la línea de:

```
. dot -Tpng decisiontree.dot -o decisiontree.png
```

. Al terminar esto, fue necesario realizar el mismo proceso, ahora con los datasets de Wine, y el de Breast Cancer.

4. RESULTADOS

4.1. Iris Dataset

Después de generar los árboles de decisión, desde la gráfica, hasta el árbol con sus nodos a partir del archivo dot, los resultados quedaron como se muestran debajo.

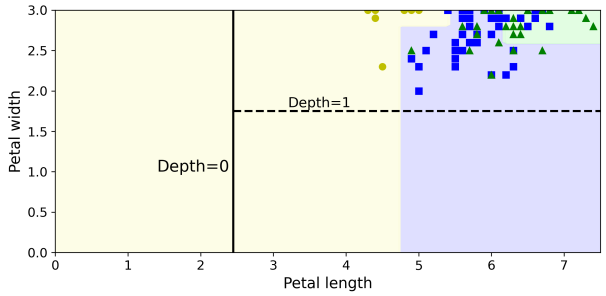

Figura 2: Dataset Iris Boundaries Gráfica

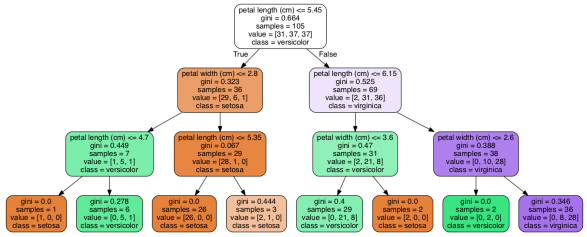

Figura 3: Dataset Iris Árbol de Decisión

4.2. Wine Dataset

Así como se generó todo para el dataset de Iris, se realizó lo mismo para el de Wine.

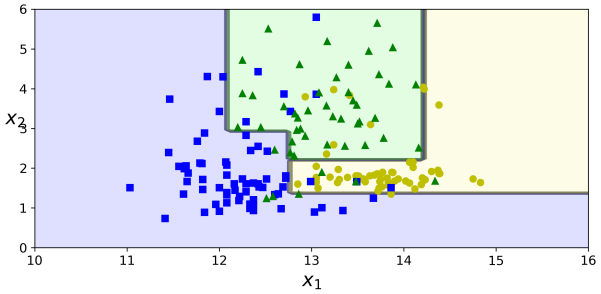

Figura 4: Dataset Wine Boundaries Gráfica

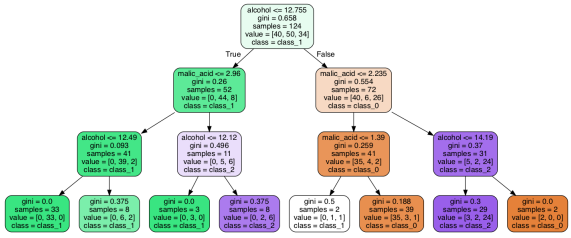

Figura 5: Dataset Wine Árbol de Decisión

4.3. Breast Cancer Dataset

Y finalmente, se realizó el mismo proceso para el de Breast Cancer.

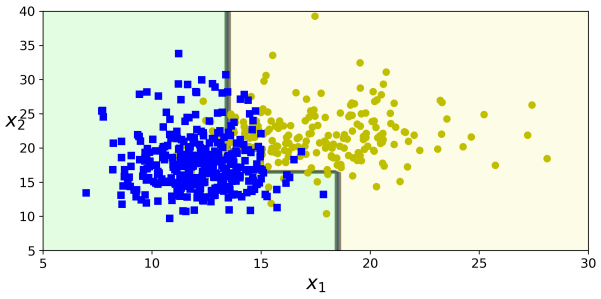

Figura 6: Dataset Breast Cancer Boundaries Gráfica

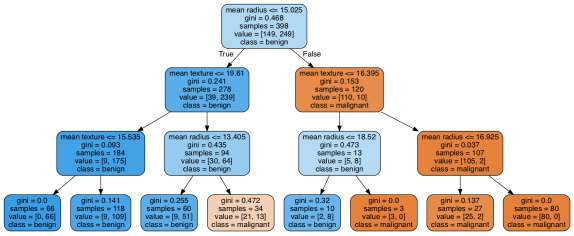

Figura 7: Dataset Breast Cancer `Árbol de Decisión`

4.4. Precisión de los modelos generados

```
python3 practice5.py
IRIS DECISION TREE PRECISION:
75.55555555555556
Saving figure iris_decision_tree_decision_boundaries_plot
WINE DECISION TREE PRECISION:
83.33333333333334
Saving figure wine_decision_tree_decision_boundaries_plot
BREAST CANCER DECISION TREE PRECISION:
90.05847953216374
Saving figure breast_cancer_decision_tree_decision_boundaries_plot
```

Figura 8: Precisión de los modelos

5. CONCLUSIONES Y REFLEXIONES

Como podemos ver los árboles de decisión nos permite generar modelos de clasificación más complicados, ya que la consideración de los atributos por ramas permite ser mucho más específico y poder apegarnos más a la realidad de los datos. Claro que esto es con base a un modelo matemático y puede diferir con respecto a las relaciones que tienen los datos en realidad.

Jair Antonio. A mi me parece bastante interesante la aplicación de teoría de la información (la fórmula de entropía es un resultado de esta área) para generar modelos que nos permiten "ordernar" nuestros datos. Esto me pone a pensar qué otros resultados de otras áreas se pueden utilizar para generar técnicas y métodos para resolver problemas.

Maximiliano Zambada. Todo este tema de árboles de decisión, estar usando la entropía y la ganancia para poder ordenar los datos y tomar decisiones, es muy útil y fácil de implementar con los datasets que nos sean otorgados. Después de indagar con mi equipo, descubrí que el tema de la entropía se deriva de la teoría de información, cosa que me abre puertas a investigar las demás ramas de esta área y los usos que se les puede dar a distintas situaciones.