Universidad Icesi

Andrea Núñez Rodriguez

Danna Garcia Trujillo

Camilo Gutiérrez Cordoba

Camilo Escobar Arteaga

Proyecto Integrador

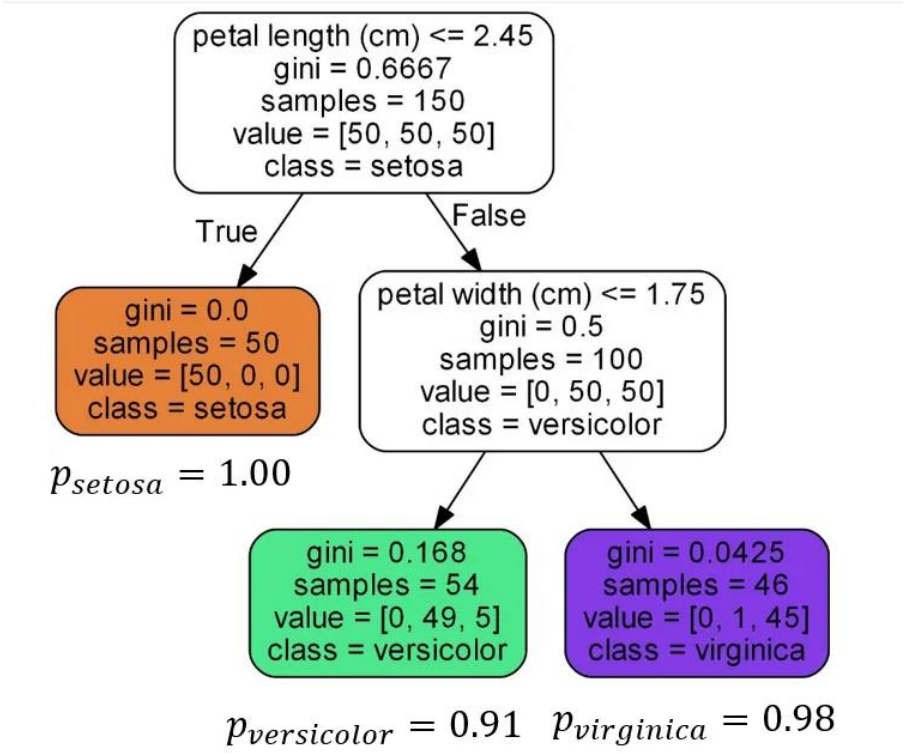
Entrega Final

ETAPAS DEL DISEÑO DE EXPERIMENTOS

1. **Planeación y realización**

A) Entender y delimitar el problema u objeto de estudio

Los árboles de decisión son una técnica de aprendizaje automático supervisado muy utilizada en muchos negocios. Como su nombre indica, esta técnica de machine learning toma una serie de decisiones en forma de árbol. Los nodos intermedios (las ramas) representan soluciones. Los nodos finales (las hojas) nos dan la predicción que vamos buscando. árboles de clasificación cuando la variable dependiente es de tipo cualitativo, La clasificación es un proceso de dos pasos, paso de aprendizaje y paso de predicción. En el paso de aprendizaje, el modelo se desarrolla en base a datos de capacitación dados. En el paso de predicción, el modelo se usa para predecir la respuesta para datos dados.



1. **Árbol de decisión de implementación propia:** Consta de un árbol binario adaptado para contener nodos representativos de valores de verdad correspondientes a un set automatizado de validaciones, dichas validaciones toman la forma de decisiones sobre las variables de entrada evaluando todas las combinaciones posibles y seleccionando la decisión que genere el particionamiento con mayor pureza del conjunto de los datos. Una vez el árbol y sus nodos quedan construidos, el árbol está en la capacidad de recibir una tupla de datos de entrada, verificar dichos datos recorriendo sus nodos y decisiones hasta llegar a un nodo hoja que contiene un tag de clasificación acorde al problema.

2. **Árbol de decisión usando la librería Accord.NET:** Consta de un árbol de decisión disponible en el framework de Accord.NET. Esta librería permite al usuario compilar los árboles de decisiones en código sobre la marcha, aumentando aún más su rendimiento durante la clasificación. La librería brinda la implementación de un árbol de decisión con métodos para calcular la clasificación del árbol dado un vector de entrada; el cual a su vez contiene los nodos de decisión del árbol que pueden o no tener una colección de nodos hijos; brinda también la posibilidad de especificar la variable de decisión según su naturaleza (si la variable es continua o discreta) y cuáles son sus rangos de valores válidos; y finalmente una colección de nodos secundarios junto con información sobre qué atributo de los datos debe compararse con los nodos durante el razonamiento.

El objetivo de este experimento es determinar cuál de las dos implementaciones de los árboles de decisión es más precisa en cuanto al resultado que arroja según el set de entradas que se le introduzca, teniendo en cuenta factores que impacten el resultado de cada una de las respuestas del árbol.

B) Elegir las variables de respuesta que será medida en cada punto del diseño y verificar que se mide de manera confiable.

En función de los resultados que se desean obtener y partiendo del diseño que llevar a cabo, decidimos tomar **la precisión** de los algoritmos como una variable de respuesta, ya que esta nos servirá para verificar el correcto funcionamiento de cada algoritmo. Es por esta razón, que realizando una investigación pertinente encontramos la siguiente información:

En la implementación propia, el valor de precisión se calcula usando el complemento porcentual del valor de Impureza de Gini, el cual está definido matemáticamente por:

$$I_G(f) = \sum_{i=1}^{m} f_i(1 - f_i) = \sum_{i=1}^{m} (f_i - {f_i}^2) = \sum_{i=1}^{m} f_i - \sum_{i=1}^{m} {f_i}^2 = 1 - \sum_{i=1}^{m} {f_i}^2$$

El cual describe la medida de cuán probable es que una tupla de datos ingresada al árbol de decisión sea etiquetada con una clasificación incorrecta.

En la implementación propia, el cálculo de la Impureza de Gini se calcula con base a la probabilidad de un elemento a la vez, como sigue:

```
private double calculateGini(List<Dictionary<string, object>> list)
{
    double gini = 0;

    Dictionary<string, double> counts = new Dictionary<string, double>();

    foreach (string tv in targetValues)
    {
        counts[tv] = 0.0;
    }

    foreach (Dictionary<string, object> item in list)
    {
        counts[(string)item[targetVariableName]]++;
    }

    foreach (string item in targetValues)
    {
        gini += (counts[item] / Convert.ToDouble(list.Count)) * (1 - (counts[item] / Convert.ToDouble(list.Count)));
    }

    return gini;
}
```

En la implementación del árbol de decisión utilizando el framework de Accord.NET se calcula la precisión de la respuesta del árbol mediante el método ZeroOneLoss, el cual realiza el cálculo de la pérdida entre los valores esperados y los valores reales que se han predicho. Para dar la respuesta final sacamos el complemento del cálculo anterior y esa sería como tal la precisión utilizando la implementación del árbol de decisión del framework de Accord.NET.

```
1 referencia
public double Error()
{
    double error = new ZeroOneLoss(outputs).Loss(tree.Decide(inputs));

    return error;
}
```

C) Determinar la unidad experimental:

Teniendo en cuenta que nuestro problema busca encontrar que tan preciso es el resultado que la implementación del árbol de decisión tiene, la unidad experimental es cada una de las simulaciones con diferentes valores de entrada en cada ejecución, ya que con cada una de estas se evalúa el comportamiento del algoritmo del árbol de decisión propio y en el árbol de decisión con la librería de Accord.NET.

D) Determinar cuáles factores deben estudiarse o investigarse, de acuerdo a la supuesta influencia que tienen sobre la respuesta.

Los factores estudiados elegidos para realizar el experimento son los siguientes:

- Entradas del registro el cual se ingresa para evaluar la precisión de cada árbol. Cada registro consta de unas variables de entrada las cuales ingresa el usuario. Estas variables son: el país, el año, generación y el género. A partir de estas entradas el algoritmo de precisión hace el cálculo y arroja una respuesta dependiendo de los datos de entrenamiento que tenga cada árbol de decisión, sin embargo, las variables que determinan como tal la precisión del árbol sería el tipo de implementación que se escoja y el país a consultar.
- Dataset de entrenamiento de cada árbol. Cada implementación de los árboles realiza el entrenamiento de sus datos a partir de un conjunto de registros los cuales son utilizados por el método de entrenar de cada uno. El método de entrenar define internamente

dentro del código que tan precisa va a hacer la respuesta según las entradas del registro que ingresemos y de los datos que estén en el dataset de entrenamiento, por lo tanto, es importante que ambas implementaciones cuenten con un dataset de entrenamiento similar.

Factores no controlables que afectan la variable experimental:

- Funcionamiento interno de la librería con la que se implementa el árbol de decisión la cual es accord.NET, esta cuenta con unos métodos los cuales usa para calcular la respuesta deseada (como por ejemplo ID3Learning(), Learn(), Decide() y ZeroOneLoss()).

E) Seleccionar los niveles de cada factor, así como el diseño experimental adecuado a los factores que se tienen y al objetivo del experimento.

Para las variables de estudio escogidas se escogen los siguientes niveles:

- Entradas del registro: cada entrada se va a componer de unos niveles específicos de pendiendo de la naturaleza de la variable, en este caso se va a escoger todos los países que se encuentran en el dataset de entrenamiento del árbol que en su total son 101, es decir, la variable País tendría 101 niveles. La implementación del árbol tendría dos niveles en este caso, los cuales son:
    - Implementación del árbol de decisión, en este caso se tienen dos implementaciones en el sistema:
        - Implementación propia.
        - Implementación con librería Accord.NET.
- Dataset de entrenamiento, para ambas implementaciones se fija el mismo dataset.

F) Planear y organizar el trabajo experimental.

En este diseño experimental se tomarán en cuenta los siguientes aspectos: Las personas involucradas en el diseño, análisis del experimento, codificación, programación del experimento, toma de resultados e interpretación serán todos los integrantes del equipo; con el fin de que no halla confusiones en la decisión final que se tome sobre el cual es la mejor precisión de las implementaciones de los árboles.

2.

ANÁLISIS

Se lleva a cabo el análisis de varianza (ANOVA) con el objetivo de verificar si las precisiones de los árboles de decisión, tanto el propio como el externo, tienen semejanza.

Lo primero que se hace es sacar la hipótesis nula y la hipótesis alternativa.

$$H0: \sigma_1^2 = \sigma_2^2$$

$$H1: \sigma_1^2 \neq \sigma_2^2$$

Seguidamente se escoge el nivel de significancia del 0,05.

RESUMEN

Grupos	*Cuenta*	*Suma*	*Promedio*	*Varianza*
Implementación Propia	97	4,2097	0,04339897	0,00140287
Implementación Externa	97	4,8443	0,04994124	0,00108609

ANÁLISIS DE VARIANZA

Origen de las variaciones	*na de cuadrados*	*de libertlio de los cua*		*F*	*Probabilidad*	*Valor crítico para F*
Entre grupos	0,00207586	1	0,00207586	1,66805151	0,19807235	3,890347919
Dentro de los grupos	0,23894072	192	0,00124448			
Total	0,24101659	193				

La decisión es no rechazar la hipótesis nula, debido a que el valor F calculado (1,66) es menor que el valor crítico (3.89). Se concluye que no hay una diferencia entre las variaciones de las precisiones de los árboles de decisión.

CONCLUSIONES

1. Al realizar el análisis de varianza Anova, pudimos concluir que ambas implementaciones de los árboles ofrecían el promedio de precisión muy similar, por lo que se concluye que no hay diferencia entre sus variaciones.
2. Una de las cosas que puede haber llevado al caso de que los promedios de las precisiones fueran muy semejantes es que los entrenamientos de ambos árboles se hacían con el mismo dataset de datos, por lo que en ninguno de ambos casos se omitía algún dato en alguno de los árboles.
3. En cuanto a la implementación del árbol propio, se trató de hacer de modo que el árbol quedara muy bien entrenado, y que al compararse con una implementación de librería no hubiera una mayor diferencia.
4. El resultado también se le puede atribuir a que se hicieron varias iteraciones en los casos de pruebas que permitieron una mayor cantidad de datos para analizar.
5. Con respecto a las precisiones de los árboles, se puede concluir que son muy altas, pues el error que ambas tienen es de 0,043 y 0,049, y al hacer el complemento para sacar la precisión, resultan precisiones del más del 90%, lo cual es muy satisfactorio.

Referencias:

https://sitiobigdata.com/2019/12/14/arbol-de-decision-en-machine-learning-parte-1/

https://support.minitab.com/es-mx/minitab/18/help-and-how-to/modeling-statistics/doe/supporting-topics/taguchi-designs/catalogue-of-taguchi-designs/#l16-45