



# **Instituto Tecnológico y de Estudios Superiores de Monterrey Campus Estado de México**

---

## **“Análisis y Reporte sobre el desempeño del modelo”**

Eduardo Acosta Hernández

A01375206

Ingeniería en Tecnologías Computacionales

Inteligencia artificial avanzada para la ciencia de datos

I

Grupo 101

Jorge Adolfo Ramírez Uresti

13 de agosto de 2022

La entrega elegida para este módulo es la entrega 2 que consta de la implementación de un algoritmo con el uso de un framework o una librería, el algoritmo implementado es el Decision Tree Classifier de la librería sklearn.

El algoritmo es capaz de clasificar múltiples clases en un dataset, requiere de un input bidimensional de los parámetros independientes y una variable target la cual contiene las diferentes clases. Este algoritmo subdivide el dataset en arboles de decisión, tomando en cuenta las características proporcionadas. Cabe mencionar que en este tipo de algoritmos es muy importante lograr identificar el número optimo de árboles para generar, debido a que es común que cuando se generan muchos subárboles se presente overfitting.

En cuanto al dataset este cuenta con una variable Species, la cual contiene las clases de pescado, de igual manera cuenta con variables que nos hablan de las dimensiones físicas del pescado como, alto, ancho, longitud, peso, etc. Con el algoritmo se buscará llegar a un modelo que nos permita predecir la especie de un pescado basándonos en sus dimensiones.

Para realizar lo anterior dividimos el dataset en una parte de entrenamiento y otra de pruebas, donde se utilizaron el 80% de los datos para entrenar y 20% para probar. Dicha división del dataset se realizó con la dependencia model selection de la librería sklearn.

```
PS C:\Users\laloa\Documents\0ITC\7.SeptimoSemestre\Reto\entrega
Documents\0ITC\7.SeptimoSemestre\Reto\entregasmodulo2\entrega2f
Species Weight Length1 Length2 Length3 Height Width
0 Bream 242.0 23.2 25.4 30.0 11.5200 4.0200
1 Bream 290.0 24.0 26.3 31.2 12.4800 4.3056
2 Bream 340.0 23.9 26.5 31.1 12.3778 4.6961
3 Bream 363.0 26.3 29.0 33.5 12.7300 4.4555
4 Bream 430.0 26.5 29.0 34.0 12.4440 5.1340
Exactitud del modelo 0.71875
Exactitud del modelo(train) 1.0
```

### **Análisis del ajuste del modelo**

Como se puede observar en el resultado de exactitud de la etapa de entrenamiento se obtuvo un 100%; mientras que en la etapa de pruebas se obtuvo un 77% lo que nos indica que el modelo se sobre ajustó a los datos de entrenamiento. Con sobreajuste nos referimos a que el modelo deja de aprender a generalizar los datos y empieza simplemente a memorizar, lo que ocasiona que tenga un desempeño pobre en la etapa de prueba.

### **Análisis de Bias**

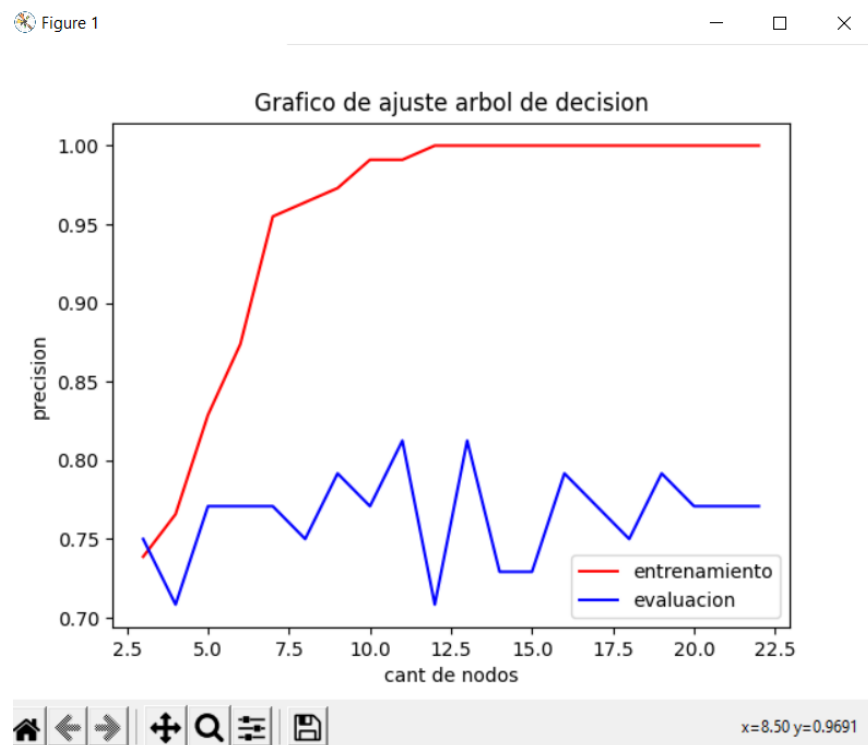
Tomando en cuenta el valor de exactitud de la etapa de entrenamiento podemos observar que no se posee ningún bias o sesgo, pues el modelo puede clasificar de manera perfecta el 100% de las ocasiones.

## Análisis de la Varianza

Si analizamos los valores obtenidos en la etapa de entrenamiento y prueba podemos ver que el modelo se desempeña de manera fenomenal en la parte de entrenamiento, sin embargo, en la parte de prueba lo que sucede es que disminuye su desempeño en un 77%. Eso significa que el modelo en su estado actual posee un nivel de varianza muy elevado, ya que se comporta de manera muy pobre al encontrarse con datos que nunca ha visto antes.

## Ajuste de hiperparámetros

Con el objetivo de eliminar los fallos anteriores lo que se hará es un ajuste de hiperparámetros con la finalidad de cambiar la estructura del modelo. A continuación, se presenta una gráfica donde podemos apreciar la precisión del modelo a medida que se varía la máxima profundidad que este puede tener, a esta gráfica también se le conoce como gráfica de ajuste.



De esta gráfica podemos notar que obtenemos la máxima precisión si ajustamos la máxima profundidad a 11 nodos.

Al realizar el cambio tomando 11 nodos como máximo podemos observar que disminuye la exactitud en la parte de entrenamiento, pues baja hasta un 98%. Sin embargo, lo impresionante se encuentra en la parte de prueba pues podemos observar que el modelo subió su exactitud en esta etapa hasta un 81%, esto nos indica que el ajuste de hiperparámetros funcionó y de seguir ajustando los hiperparámetros del modelo podríamos obtener un mejor desempeño.