

## **Modelo predictivo de conductas externalizantes**

**Presentado Por:**

**Yuri Natalia Bernal Mora**

**Nicolas Felipe Reyes Carillo**

**María Alejandra Maldonado Rojas**

**Modelo de predicción**

**Facultad De Ingeniería**

**Universidad Cooperativa De Colombia**

**Fredys Alberto Simanca Herrera**

**Abril de 2021**

## **Contenido**

Contenido .....	2
1. Introducción .....	4
2. Metodología .....	7
3. Resultados .....	11
4. Bibliografía .....	13

## 1. Introducción

En la actualidad existe una gran variedad de trastornos de conducta, cada uno de ellos con distintas manifestaciones. Los cuales normalmente se caracterizan por una combinación de alteraciones del pensamiento, la percepción, las emociones y la que se ve principalmente afectada es la conducta ya que ayuda a la violación de reglas-normas (grupales y sociales) y a tener conductas inapropiadas para la edad (conducta sexualizada a los 7-8 años), incluyendo desobediencia a padres y profesores y alteraciones en todo tipo de relaciones.

Los trastornos de conducta más conocidos son TDAH, TOP Y TC, pero todos estos trastornos mencionados antes aparecen normalmente desde la edad temprana y los trastornos mentales entre los niños son más comunes de lo que se espera; estos se pueden manifestar como cambios serios y pueden afectar directamente en la forma en que los niños aprenden, se comportan o manejan sus emociones, lo que causa angustia y problemas para pasar el día. Muchos niños ocasionalmente experimentan miedos y preocupaciones o muestran comportamientos perturbadores. Si los síntomas son graves y persistentes e interfieren con las actividades de la escuela, el hogar o juegos.

Los trastornos mentales infantiles afectan a muchos niños y familias. Los niños y niñas de todas las edades y orígenes étnicos / raciales que viven en todas las regiones de los Estados Unidos padecen trastornos mentales. Basado en el informe del National Research Council and Institute of Medicine y en Colombia se ha observado es que el número de personas de 0 a 19 años que consultan por trastornos mentales y del comportamiento es cada día mayor. De 2009 a 2017 en Colombia se atendieron 2.128.573 niños, niñas y adolescentes con diagnósticos con código CIE 10: F00 a F99 (que agrupa los trastornos mentales y del comportamiento), con un promedio de 236.508 de personas atendidas por año, la tendencia es al aumento de casos cada año.

A pesar de que las cifras siguen en aumento en todo el mundo y que es un problema de salud pública. Al momento del diagnóstico aún se usan métodos muy rudimentarios; los cuales retrasan el proceso de diagnóstico o en otros casos ni siquiera se percatan de las dificultades por las que están pasando los niños y jóvenes y algo primordial para

tratar estos trastornos es la detección temprana de estos mismos y claro que existen diferentes tipos de tratamientos para cada uno de estos trastornos por ejemplo en trastornos de la conducta está la Terapia Cognitivo Conductual (TCC), La terapia multisistémica (MST) y Régimen asistencial fuera de casa.

Pero uno de los mayores problemas actualmente en cuanto a los tratamientos es la gran diversidad de síntomas que presenta una persona con estos trastornos y que conlleva a confusiones y mayor tiempo de diagnóstico.

Por esta razón nuestro Modelo de predicción estará basado para la detección de trastornos de conducta en niños y adolescentes de 7 a 12 años. El cual tiene como objetivo general desarrollar un entorno controlado mediante el uso de machine learning y uno de sus algoritmos más conocidos como lo son los árboles de decisión y con esta tecnología poder predecir si los niños tienen o no un trastorno de conducta, esto también facilitará la detección precoz de todos estos trastornos y la reducción los tiempos de diagnóstico y generar un avance e impacto social gracias al uso de esta nueva tecnología en el área de psicología mediante la aplicación de machine learning.

Hoy en día ya existen muchas herramientas tecnológicas que favorecen los procesos terapéuticos ya que estos sistemas se basan en el funcionamiento observado en el sistema cognitivo humano, actualmente en España se ha iniciado un proyecto para la implementación de un entorno virtual que permitirá la evaluación y entrenamiento del Trastorno del Espectro Autista y existen muchas aplicaciones se basan en un chatbot el cual se basa en preguntas hacia los niños de como se sienten normalmente y demás como un acompañante virtual, pero ninguna ayuda a predecir si sufren o no de algún trastorno.

La idea de iniciar con esta investigación es de llegar a poder implementar un ambiente controlado para el manejo de problemas conductuales los cuales han de ser tratados con una machine learning, del que podríamos decir que puede cumplir de mejor manera ciertas funciones de predicción gracias a sus diferentes algoritmos ya que cuentan con funciones analíticas y un aprendizaje constante el cual nos permitiría obtener soluciones a problemas mucho más eficientes "Sin entrar en lo ético, humano y psicológico".

## 2. Metodología

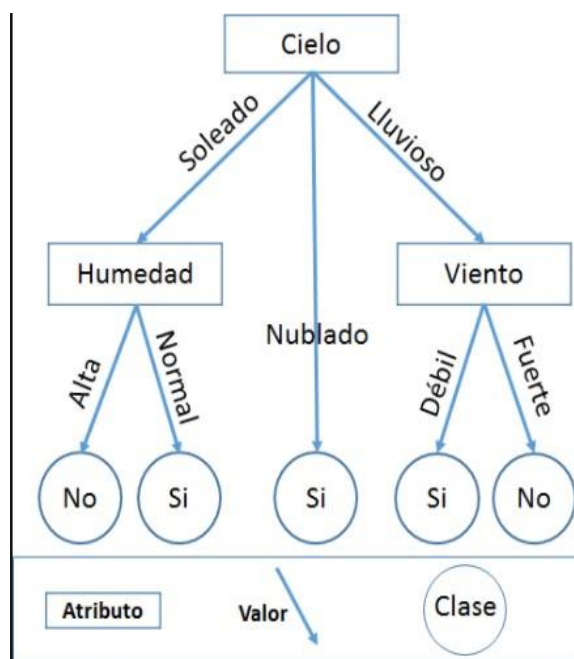
Este modelo predictivo usara el algoritmo de aprendizaje supervisado de Machine Learning, Arboles de decisión, este algoritmo es un modelo clásico de decisiones, este algoritmo tuvo origen en las ciencias sociales con los trabajos de Sonquist y Morgan el año 1964 y Morgan y Messenger el año 1979, ambos realizados en la Universidad de Michigan. El programa para la “Detección de Interacciones Automáticas”, creada el año 1971 por los investigadores Sonquist, Baker y Morgan, fue uno de los primeros métodos de ajuste de los datos basados en árboles de clasificación. En estadística, el año 1980, Kass introdujo un algoritmo recursivo de clasificación no binario, llamado “Detección de Interacciones Automáticas Chi-cuadrado”. Hacia el año 1984, los investigadores Breiman, Friedman, Olshen y Stone, introdujeron un nuevo algoritmo para la construcción de árboles y los aplicaron a problemas de regresión y clasificación. El método es conocido como “Árboles de clasificación y regresión”. Casi al mismo tiempo el proceso de inducción mediante árboles de decisión comenzó a ser usado por la comunidad de “Aprendizaje automático”. como lo indica su nombre este modelo comienza desde un único nodo y se ramifica variables o alternativas. Cada hoja de la ramificación representa o constituye una clasificación o una decisión con el fin de encontrar la mejor opción.

La idea básica detrás de cualquier problema de árbol de decisión es seleccionar el mejor atributo utilizando una medida de selección de atributos o características luego de esto se debe hacer de ese atributo un nodo de decisión y divide el conjunto de datos en subconjuntos más pequeños, para comenzar la construcción del árbol repitiendo este proceso recursivamente para cada atributo hasta que una de las siguientes condiciones coincida: Todas las variables pertenecen al mismo valor de atributo, ya no quedan más atributos o ya no hay más casos.

La medida de selección de atributos es una heurística para seleccionar el criterio de división que divide los datos de la mejor manera posible. También se conoce como reglas de partición porque nos ayuda a determinar puntos de ruptura para conjunto en un nodo dado.

Se puede interpretar también como un mapa de probables resultados de una cadena de pasos relacionados entre sí, “Hay tres tipos diferentes de nodos: nodos de probabilidad, nodos de decisión y nodos terminales. Un nodo de probabilidad, representado con un círculo, muestra las probabilidades de ciertos resultados. Un nodo de decisión,

representado con un cuadrado, muestra una decisión que se tomará, y un nodo terminal muestra el resultado definitivo de una ruta de decisión.” *Qué es un diagrama de árbol de decisión.* (s. f.). Lucidchart. Recuperado 1 de abril de 2021, de <https://www.lucidchart.com/pages/es/que-es-un-diagrama-de-arbol-de-decision>.



En este algoritmo tenemos diferentes medidas de selección de atributos y se definen como una heurística para seleccionar el criterio de división que divide los datos de la mejor manera posible. También se conoce como reglas de división porque nos ayuda a determinar puntos de interrupción para tuplas en un nodo dado.

ASM proporciona un rango para cada característica (o atributo) al explicar el conjunto de datos dado. El atributo de mejor puntuación se seleccionará como un atributo de división.

En el caso de un atributo de valor continuo, los puntos de división para las ramas también deben definirse. Las medidas de selección más populares son Ganancia de información, Proporción de ganancia, Índice de Gini y entropía.

La entropía es una medida que se aplica para cuantificar el desorden de un sistema. Si un nodo es puro su entropía es 0 y solo tiene observaciones de una clase, pero si la entropía es igual a 1, existe la misma frecuencia para cada una de las clases de observaciones.

La entropía tiende a crear nodos balanceados en el número de observaciones. Relacionado con la entropía se define la Ganancia de Información que busca la división con mayor ganancia de información, es decir, con menor entropía ponderada de la

variable. y esta definida por la siguiente ecuación  $H = -\sum_{i=1}^n P_i * \log_2 P_i$  Donde  $P_i$  es la probabilidad de que un ejemplo sea de la clase  $i$ .

Las ventajas de usar este algoritmo es que los árboles de decisión son fáciles de interpretar y visualizar, pueden capturar fácilmente patrones no lineales, requieren menos preprocesamiento de datos por parte del usuario, por ejemplo, no es necesario normalizar las columnas y se puede utilizar para ingeniería de características, como la predicción de valores perdidos, adecuada para la selección de variables ya que el árbol de decisión no tiene suposiciones sobre la distribución debido a la naturaleza no paramétrica del algoritmo.

Para nuestro ejercicio las librerías más importantes que utilizamos son pandas para leer los datos, también utilizamos sklearn que nos permite utilizar varias funcionalidades de esta para limpiar y realizar procesamiento de dato y para hacer separara la información de entrenamiento y test entre otros usos del árbol de decisiones.

Utilizamos adiconal la herramienta de sklearn DecisionTreeClassifier que nos ayuda a el entrenamiento del árbol, donde le pasamos solo los parámetros y el realiza el trabajo de de entrenamiento y permite saber la profundidad que le queramos dar al árbol, esto dependiendo el caso de nuestra data.

### 3. Resultados

## Sistema de Predicción

### Preparación de Datos ¶

```
In [5]: # Libreria arbol decision y procesamiento data
import pandas as pd
from sklearn.tree import DecisionTreeClassifier
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder

#Graficar arbol
from sklearn.tree import export_graphviz
#import graphviz
import matplotlib.pyplot as plt
import numpy as np
```

```
In [6]: # Carga del archivo con configuraciones para que funcionen con utf-8
data = pd.read_csv('data.csv', header=0, encoding='unicode_escape')

In [7]: # Si tuvieran valores nulos
data = data.dropna()

In [8]: # Asignar a variables valores de columnas especificas del archivo
toc = data.iloc[:, 4].values
animo = data.iloc[:, 5].values
obediencia = data.iloc[:, 6].values

# Borramos columnas que vamos a reemplazar
data.drop(['animo', 'obediencia', 'fecha'], axis='columns', inplace=True)

In [9]: # Instancia de LabelEncoder
LabelEncoder_data = LabelEncoder()

# Escalar volver datos string en numeros
data['animo'] = LabelEncoder_data.fit_transform(animo)
data['obediencia'] = LabelEncoder_data.fit_transform(obediencia)

In [10]: #castear algunas columnas con el tipo de dato correspondiente
data = data.astype({'id_niño': int, "actividad": int, "categoria_actividad": int, "edad": int, "tiempo_rsp_seg": int})
```

217 rows x 2 columns

```
In [13]: #Asignamos Los Labels que va a manejar el arbol
predictors_labels = ['animo', 'obediencia']
target_label = ['si', 'no']

In [14]: #Obtenemos solo Los valores para poder entrenar el arbol
x = data.values
y = data['TDAH_TOP_TC']

In [15]: #Creamos el train_test_split que "divide" la informacin una de entrenamiento 80% y otra de test 20%
x_entrena, x_test, y_entrena, y_test = train_test_split(predictors,targets,test_size=0.2, random_state=123)

In [16]: #imprimimos division de data para test y entrenamiento
print("entrena", x_entrena.shape, y_entrena.shape)
print("test", x_test.shape, y_test.shape)

entrena (173, 2) (173,)
test (44, 2) (44,)

In [50]: #Construimos el arbol con una maxima profundida de 4
arbol = DecisionTreeClassifier(max_depth=4)

In [51]: #Entrenamos el arbol con Los datos
arbol.fit(x_entrena, y_entrena)

Out[51]: DecisionTreeClassifier(max_depth=4)
```



```
In [11]: #filtrar por edades
data = data[(data['edad'] > 4) & (data['edad'] <= 10)]
```

## Entrenamiento Modelo

```
In [12]: ##Le asignamos a una nueva variable los datos que vamos a manejar en el modelo
predictors = data[['animo', 'obediencia']]
targets = data['TDAH_TOP_TC']

predictors
```

```
Out[12]:
```

	animo	obediencia
0	5	0
1	5	0
5	3	1
6	3	1
7	5	1
...	...	...
539	2	0
543	4	0
544	0	0

```
In [52]: #probamos la prediccion con datos test
arbolpredicc= arbol.predict(x_test)
arbolpredicc
```

```
Out[52]: array(['si', 'si', 'si', 'si', 'si', 'no', 'no', 'no', 'si', 'no', 'no',
        'no', 'si', 'no', 'no', 'si', 'si', 'no', 'no', 'no', 'si', 'no',
        'no', 'no', 'si', 'si', 'si', 'si', 'no', 'no', 'no', 'no', 'si',
        'si', 'no', 'no', 'si', 'no', 'si', 'si', 'no', 'si', 'no', 'no'],
        dtype=object)
```

## Evaluar Prediccion

```
In [48]: #Porcentaje de Acierto de La prediccion en datos de test
arbol.score(x_test,y_test)
```

```
Out[48]: 0.5227272727272727
```

```
In [49]: #Porcentaje de Acierto de La prediccion en datos de entrenamiento
arbol.score(x_entrena,y_entrena)
```

```
Out[49]: 0.6069364161849711
```

## Visualizar Arbol

```
In [33]: #Tuvimos problemas con la libreria que nos permite exportar a imagen el arbol
from sklearn import tree
```

## Visualizar Arbol

```
In [51]: #Visualizamos el arbol en forma de texto
from sklearn import tree
text_representation = tree.export_text(arbol)
print(text_representation)
```

```
|--- feature_0 <= 0.50
|   |--- feature_1 <= 0.50
|   |   |--- class: no
|   |--- feature_1 > 0.50
|   |   |--- class: no
|--- feature_0 > 0.50
|   |--- feature_1 <= 0.50
|   |   |--- feature_0 <= 3.50
|   |   |   |--- feature_0 <= 1.50
|   |   |   |   |--- class: si
|   |   |   |--- feature_0 > 1.50
|   |   |   |   |--- class: no
|   |   |--- feature_0 > 3.50
|   |   |   |--- feature_0 <= 4.50
|   |   |   |   |--- class: no
|   |   |   |--- feature_0 > 4.50
|   |   |   |   |--- class: no
|   |--- feature_1 > 0.50
|   |   |--- feature_0 <= 4.50
|   |   |   |--- feature_0 <= 2.50
|   |   |   |   |--- class: si
|   |   |   |--- feature_0 > 2.50
|   |   |   |   |--- class: no
|   |   |--- feature_0 > 4.50
|   |   |   |--- class: si
```

```
In [54]: # Gráficar árbol
import matplotlib.pyplot as plt
from sklearn.tree import plot_tree

fig, ax = plt.subplots(figsize=(12, 5))

print(f"Profundidad del árbol: {arbol.get_depth()}")
print(f"Número de nodos terminales: {arbol.get_n_leaves()}")

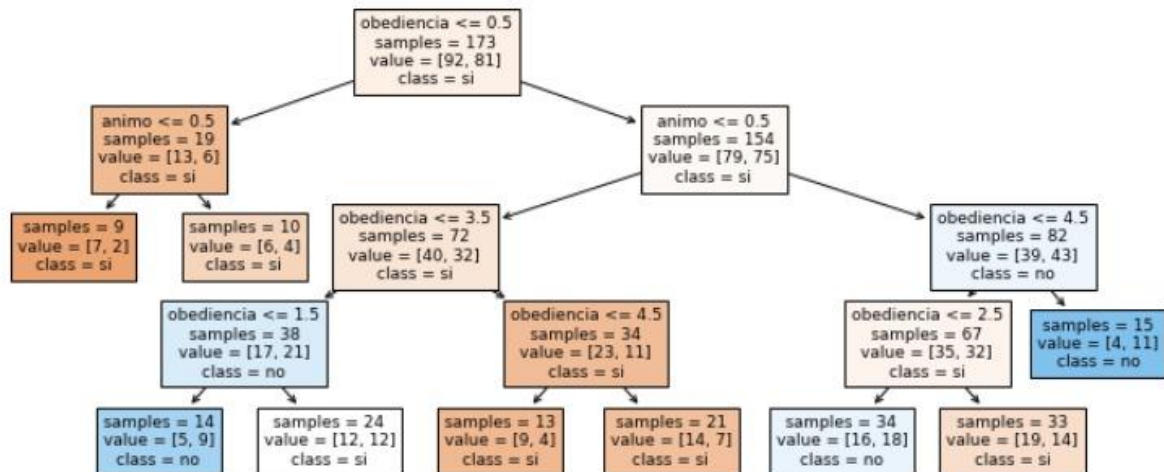
plot = plot_tree(

    decision_tree = arbol,
    feature_names = predictors_labels,
    class_names = target_label,
    filled = True,
    impurity = False,
    fontsize = 9,
    precision = 2,
    ax = ax

)
```

Profundidad del árbol: 4  
Número de nodos terminales: 9

Profundidad del árbol: 4  
Número de nodos terminales: 9



## 4. Bibliografía

- *Data and Statistics on Children's Mental Health* | CDC. (2020, 15 junio). Centers for Disease Control and Prevention.  
<https://www.cdc.gov/childrensmentalhealth/data.html>
- *Children's Mental Health Disorders - A Journey for Parents and Children*. (2014, 5 mayo). [Vídeo]. Youtube.  
<https://youtube.com/watch?v=ewbD2Dw0NLo>
- Bellido, R. M. (2019, 28 junio). *Psicología del preescolar: desarrollo de la conducta humana normal de los 3 a los 7 años de edad*. Repositorio Institucional Universidad Nacional.  
<https://repositorio.unal.edu.co/handle/unal/38944>
- Subdirección de Enfermedades No Trasmisibles Grupo Gestión Integrada para la Salud Mental. (2017, diciembre). *Boletín de salud mental Salud mental en niños, niñas y adolescentes* (N.º 4).  
<https://www.minsalud.gov.co/sites/rid/Lists/BibliotecaDigital/RIDE/VS/PP/ENT/boletin-4-salud-mental-nna-2017.pdf>
- Navarro-Pardo, E. (2012). *Desarrollo infantil y adolescente: trastornos mentales más frecuentes en función de la edad y el género*. Redalyc.org.  
<https://www.redalyc.org/articulo.oa?id=72723439006>
- <https://aprendeia.com/algoritmos-de-clustering-agrupamiento-aprendizaje-no-supervisado/>
- <https://www.clubdetecnologia.net/blog/2018/los-10-modelos-mas-populares-de-inteligencia-artificial/>
- <http://posgrado.lapaz.tecnm.mx/uploads/archivos/TesisHdzCedano.pdf>
- <https://blog.powerdata.es/el-valor-de-la-gestion-de-datos/mineria-de-datos-y-modelos-predictivos-descubriendo-patrones>

- [https://campusvirtual.ucc.edu.co/content/enforced/361842-01BOG\\_FINGE\\_PREG\\_01ISC\\_706164\\_2110\\_7864/7.%20Redes%20Neuronales%20SOM.pdf](https://campusvirtual.ucc.edu.co/content/enforced/361842-01BOG_FINGE_PREG_01ISC_706164_2110_7864/7.%20Redes%20Neuronales%20SOM.pdf)
- <https://agenciab12.com/noticia/que-es-modelo-predictivo-como-aplica-negocio>
- <https://www.lucidchart.com/pages/es/que-es-un-diagrama-de-arbol-de-decision>.
- APUNTES DE CLASE DE PYTHON. (s. f.). APUNTES DE CLASE DE PYTHON. Recuperado 11 de abril de 2021, de [https://campusvirtual.ucc.edu.co/content/enforced/361842-01BOG\\_FINGE\\_PREG\\_01ISC\\_706164\\_2110\\_7864/7.%20Redes%20Neuronales%20SOM.pdf](https://campusvirtual.ucc.edu.co/content/enforced/361842-01BOG_FINGE_PREG_01ISC_706164_2110_7864/7.%20Redes%20Neuronales%20SOM.pdf)
- Aspiazu, G. C. (2021, 20 mayo). Árboles de decisión. Blogspot. <http://menteerrabunda.blogspot.com/2009/05/arboles-de-decision.html>
- Mosquera, R. (s. f.). Máquinas de Soporte Vectorial, Clasificador Naïve Bayes y Algoritmos Genéticos para la Predicción de Riesgos Psicosociales en Docentes de Colegios Públicos Colombianos. Scielo. <https://scielo.conicyt.cl/scielo.php?pid=S071807642018000600153&script>













