INSTITUTO TECNOLÓGICO DE MONTERREY

CAMPUS LEÓN

PREDICTION INVOLVING DECISION TREES AND STUDENT PERFORMANCE DATA

ASIGNATURA: MATEMÁTICAS PARA LA TOMA DE DECISIONES

DOCENTE: MTRO. JORGE DÁVILA ORTIZ

INTEGRANTES DEL EQUIPO 4

Ginno Ángel Aguilar Durán (A00572441)

Frida Carmona Martínez (A00572365)

María Fernanda Guzmán Ayala (A00573432)

Perla Sofia Rico Casanova (A00573436)

Carlos Marcelo López Gómez (A00572357)

SEMESTRE: 2º GRUPO: 1

FECHA DE ENTREGA: LUNES 13 DE JUNIO DEL 2022

PREDICTION INVOLVING DECISION TREES AND STUDENT PERFORMANCE DATA

JUSTIFICACIÓN

Seleccionamos el proyecto anterior debido a que el hecho de que el modelo sea realmente útil en la vida real es algo indispensable, es decir, que esté enfocado en el estudio de data escolar es bastante favorable en distintos aspectos, ya que como se mencionó, éste puede ser aplicable en varios escenarios, resultando no sólo en un beneficio para la sociedad sino también en una oportunidad de negocio.

BASE DE DATOS

Recordando que, una base de datos se encarga no sólo de almacenar datos, sino también de conectarlos entre sí en una unidad lógica. Cabe mencionar que la base de datos proporcionada contiene la información de 649 estudiantes, con 30 atributos por alumno.

A continuación, se presenta el enlace en donde se encuentran los valores y datos del problema:
https://raw.githubusercontent.com/PacktPublishing/Python-Artificial-Intelligence-Projects-for-Beginners/master/Chapter01/dataset/student-por.csv

MARCO TEÓRICO

El proyecto a continuación consiste en la utilización de árboles de decisión para predecir los resultados académicos de los estudiantes utilizando información recopilada previamente; es importante recalcar que los resultados serán obtenidos a partir de una serie de factores que se aseguran ser determinantes en el comportamiento escolar de los estudiantes.

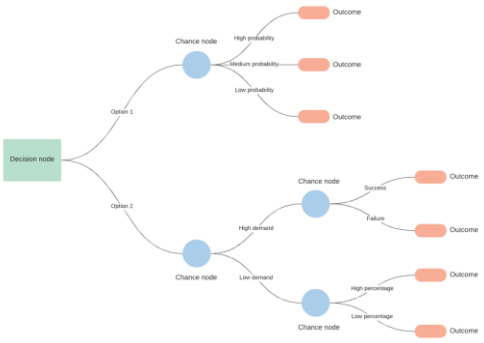
De acuerdo con la información y factores proporcionados, el objetivo del proyecto es lograr predecir con la mayor exactitud posible si el estudiante tendrá calificaciones

satisfactorias para pasar el año, o si éste reprobará.

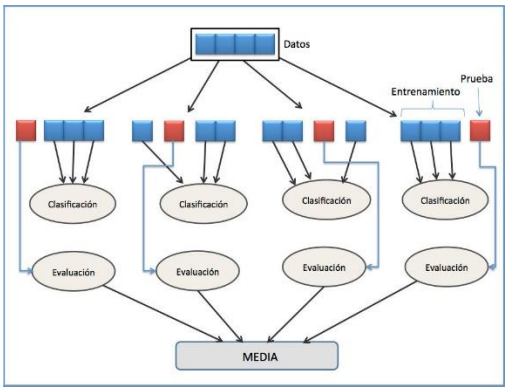
Para poder conocer un poco más del tema, a continuación, se presentan los principales temas involucrados en la solución del problema.

CONCEPTOS

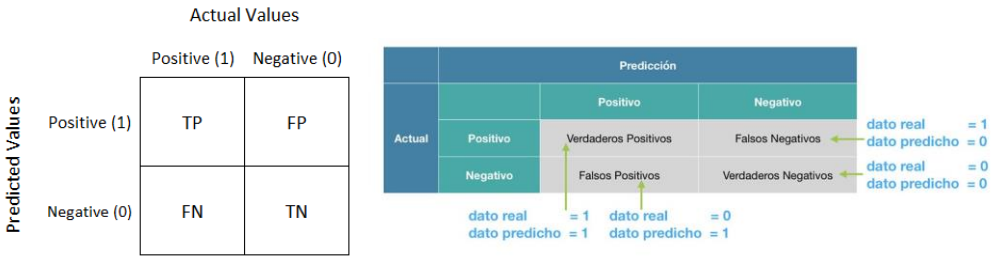
- **Árboles de decisión:** Modelo de predicción utilizado en diversos ámbitos que van desde la Inteligencia Artificial hasta la Economía. Un árbol de decisión es un mapa de los posibles resultados de una serie de decisiones relacionadas, permite que un individuo o una organización compare posibles acciones entre sí según sus costos, probabilidades y beneficios. Se pueden usar para dirigir un intercambio de ideas informal o trazar un algoritmo que anticipe matemáticamente la mejor opción.



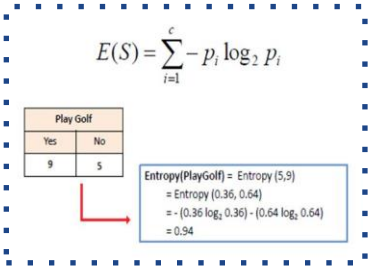
- **Validación cruzada:** Técnica que se usa para evaluar la variabilidad de un conjunto de datos y la confiabilidad de cualquier modelo entrenado con ellos. La herramienta toma como entrada un conjunto de datos con etiquetas, junto con un modelo de clasificación o regresión no entrenado. Después, divide el conjunto de datos en varios subconjuntos, crea un modelo en cada uno y, a continuación, devuelve un conjunto de estadísticas de precisión para cada subconjunto. Al comparar las estadísticas de precisión de todos los subconjuntos, se puede interpretar la calidad del conjunto de datos con el objetivo de saber después si el modelo es susceptible a variaciones en los datos.



- **Matriz de confusión:** Herramienta que permite visualizar el desempeño de un algoritmo de aprendizaje supervisado. Cada columna de la matriz representa el número de predicciones de cada clase, mientras que cada fila representa a las instancias en la clase real. En términos prácticos nos permite ver qué tipos de aciertos y errores está teniendo el modelo a la hora de pasar por el proceso de aprendizaje con los datos.



- **Entropía:** Mide el desorden en la información. En los árboles de decisión, se evalúa la entropía en los nodos, buscando a los que tengan la entropía más pequeña de todas en estos. Para ello, se calcula la homogeneidad de las muestras en los nodos. Con la fórmula, pueden probarse los diferentes atributos para encontrar el de menor entropía, es decir, los atributos más predecibles.



PROCESO DEL PROYECTO

Siguiendo con el desarrollo del proyecto, se presenta la bitácora que engloba todas las actividades que se realizaron día a día (durante las últimas 4 semanas) de acuerdo con la solución del problema presentado.

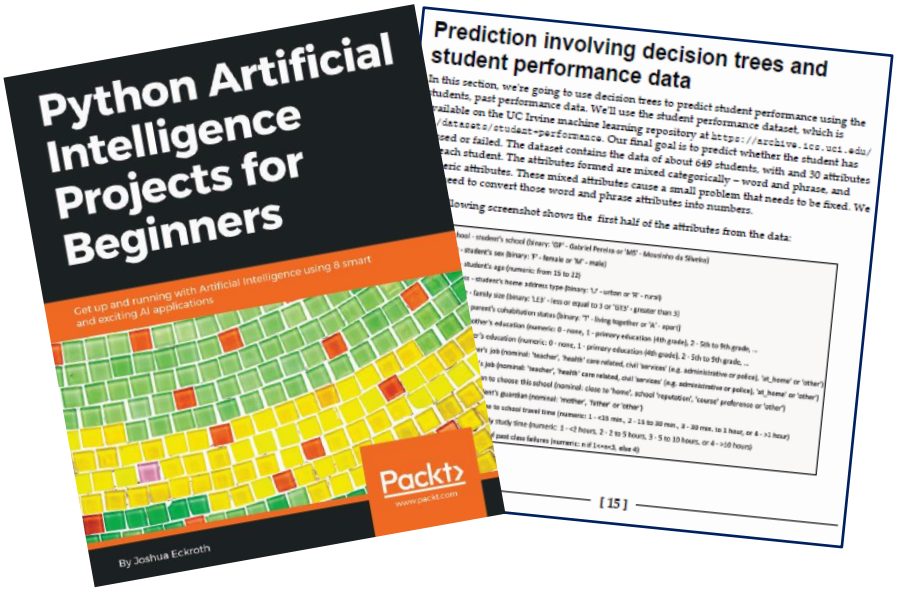
16 de mayo

- Se llevó a cabo la conformación de equipos

  Integrantes del Equipo #4

  - Ginno Ángel Aguilar Durán (A00572441)
  - Frida Carmona Martínez (A00572365)
  - María Fernanda Guzmán Ayala (A00573432)
  - Perla Sofia Rico Casanova (A00573436)
  - Carlos Marcelo López Gómez (A00572357)

- En equipo se seleccionó el proyecto elegido
  - Prediction involving decision trees and student performance data
  - Libro de origen: Python Artificial Intelligence Projects for Beginners
  - Autor: Joshua Eckroth

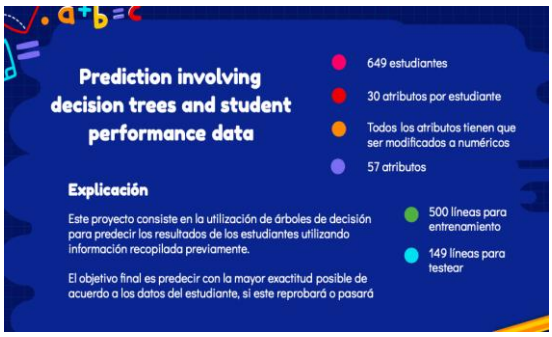


18 de mayo

- Se elaboró la presentación del proyecto que incluyó los siguientes puntos:
  - Proyecto
  - Introducción
  - Razón de elección del proyecto
  - Bitácora inicial
  - Finalidad del proyecto

19 de mayo

- Se llevó a cabo la primera exposición del equipo, en donde los principales temas fueron:
  - Presentación del proyecto
  - Bitácora inicial
  - Establecimiento de compromiso = desarrollo de código



20 de mayo

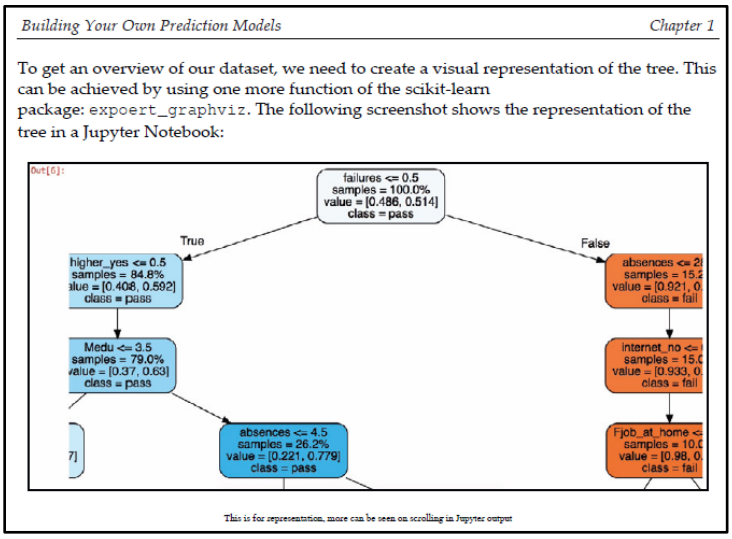
- Para el día 20 de mayo, se llevó a cabo el desarrollo del código en la plataforma de programación replit (plataforma colaborativa).
  - El principal problema que se nos presentó fue que no aparecía el árbol de decisión al correr el código.
  - Asimismo, al no poder desplegarse, esta gráfica tampoco pudo guardarse en ningún formato (jpg, png, o pdf).

23 de mayo

- El lunes 23 de mayo, se realizó la segunda exposición del equipo, en donde se presentó la primera versión del código (código completo con algunos errores).
  - Nuestro principal compromiso fue encontrar cuál había sido el error en el código, en donde no nos aparecía el árbol de decisión.

- Y, de igual manera, nos comprometimos a averiguar cuáles eran las variables más significativas e importantes, es decir, los atributos que influyeron en mayor medida en la predicción del desempeño académico de los estudiantes - de acuerdo con los resultados obtenidos en el árbol de decisión.

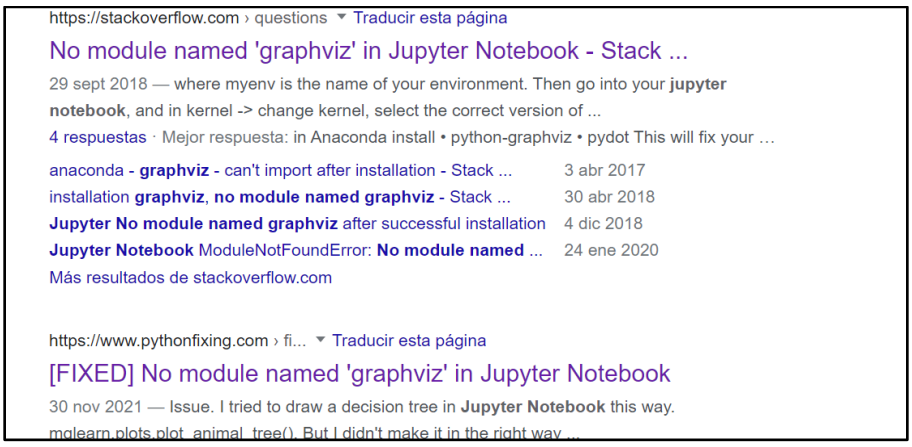
Así mismo, nos dimos cuenta de que el libro no contenía una captura de pantalla del código para la instrucción que permitía crear y visualizar el árbol de decisión, generando una laguna en la información que nos permitió concluir que habría que comparar en otras fuentes.



24 al 27 de mayo

- Durante los días siguientes, se llevó a cabo la comparación del desarrollo del código con compañeros y equipos que ya han cursado la materia, con el fin de poder encontrar las posibles soluciones a los problemas del error en el código de Python. Algunas de las recomendaciones dadas fueron:
  - Buscar ayuda en diferentes foros de programación.
  - Revisar las librerías y las posibles actualizaciones que éstas hayan tenido recientemente.

- Probar el código en las diferentes laptops del equipo, para ver si es que en alguna de estas pueden corren las librerías sin problema, y nos proporcionaba el resultado que buscamos.

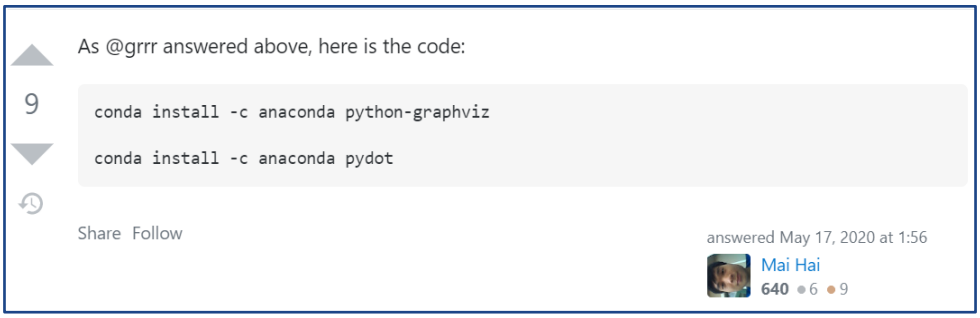


26 de mayo

- Durante la tercera exposición del equipo, se presentó la primera explicación del código y la búsqueda realizada para encontrar la solución al error, con el fin de poder obtener la gráfica final del árbol de decisión.
  - Nuestro compromiso en esta exposición fue el seguir buscando una solución que nos ayudara con el código o con la actualización de las librerías.

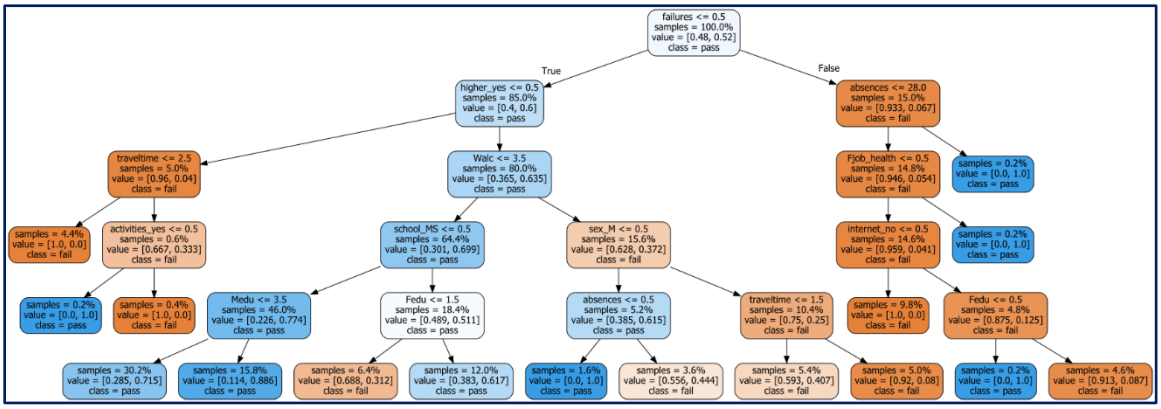
29 al 30 de mayo

- Durante los días del 29 al 30 de mayo se obtuvo el código terminado, incluido su desarrollo y comprobación final. Para ello, consultamos foros de ayuda en programación para comparar si nuestro error en el código había sido un problema previo para otros programadores.
  - La principal ayuda que encontramos fue en el foro “Stackoverflow”, cuyo link es el siguiente: *https://stackoverflow.com/questions/52566756/no-module-named-graphviz-in-jupyter-notebook*
  - De esta manera, nos dimos cuenta que el problema con el árbol de decisión es que necesitábamos instalar dos librerías distintas en Anaconda:

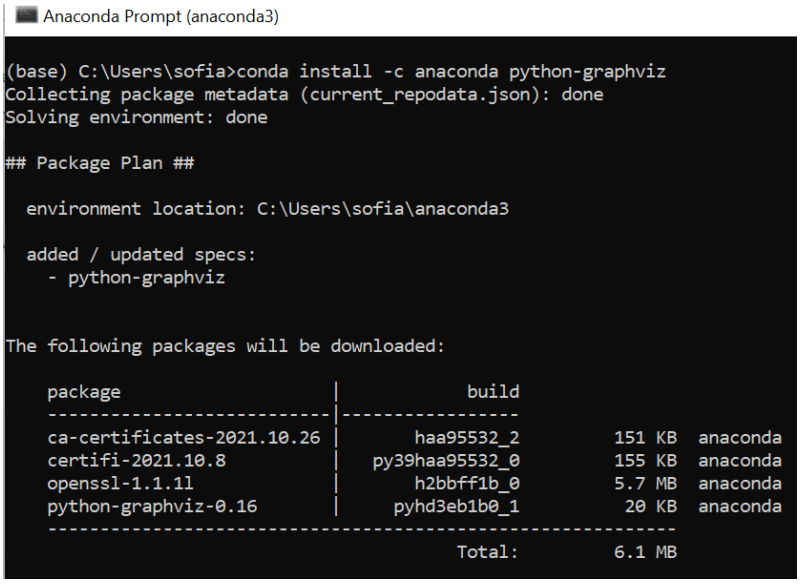


- De igual manera, lo que realizamos fue el cambio de la base de datos por una online proporcionada en otro link de ayuda del libro, con el propósito de facilitar la lectura y proceso de análisis del código.
  https://raw.githubusercontent.com/PacktPublishing/Python-Artificial-Intelligence-Projects-for-Beginners/master/Chapter01/dataset/student-por.csv

- A partir de dichas modificaciones, finalmente pudo ejecutarse el despliegue y guardado del árbol de decisión dentro del código:
  - A continuación, se presenta la imagen de la gráfica del árbol de decisión.



- Por último, podemos mencionar que lo que se hizo para poder resolver el error del código fue descargar las librerías faltantes con ayuda de la respuesta del foro de programación.
  - A continuación, se presenta el screen del proceso que se llevó a cabo:

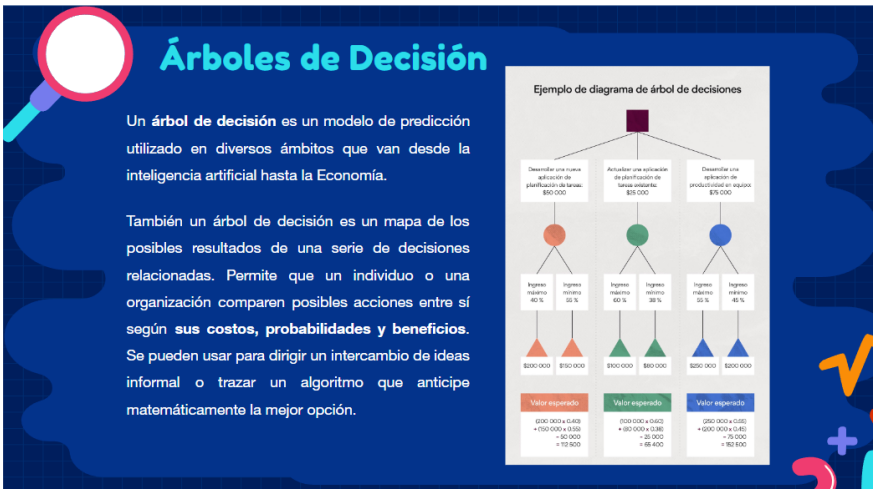


30 de mayo

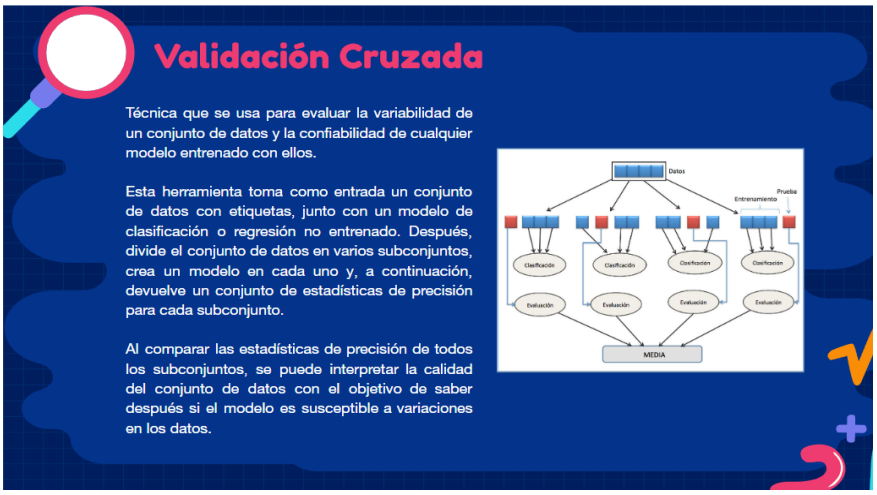
- Se llevó a cabo la cuarta exposición del equipo, compartiendo la solución encontrada y el código terminado.
- Asimismo, nos comprometimos a profundizar en el marco teórico del proyecto, con el objetivo de entender el resultado obtenido en términos de Ciencia de Datos e Inteligencia Artificial, para así poder compartir con el resto del grupo los hallazgos encontrados.

2 de junio

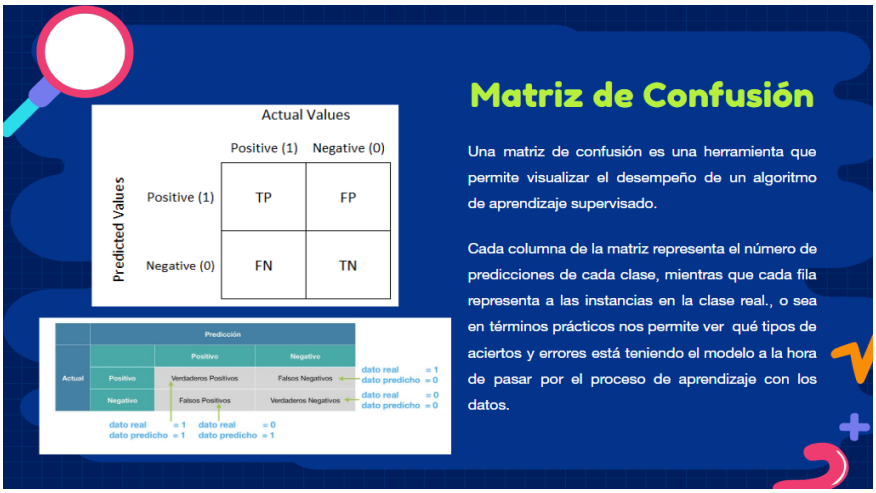
- Tras haber terminado el código y de desarrollarlo un poco, nos dedicamos principalmente a buscar y entender diferentes temas y conceptos relacionados con nuestro proyecto. Los primeros tres temas que presentamos fueron los siguientes (todo lo mostrado será de la presentación):
  - **Árboles de decisión**: Esto lo tuvimos que desarrollar y entender más debido a que es el foco central de nuestro proyecto, para así poder saber más sobre el porqué del código y entender más el árbol de decisión cuando se genere en este mismo.



- **Cross-validation (validación cruzada):** Esto se investigó más a profundidad para poder ayudarnos a entender los subconjuntos de los árboles de decisión, pero también para que se analizen e interpreten los datos con el objetivo de definir cuáles son de mayor calidad.



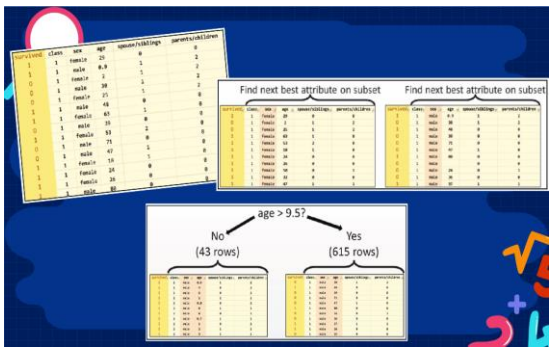
- **Matriz de confusión:** El nombre no está de decoración, pues si es un concepto algo complicado. La razón de por qué necesitaba más profundidad de explicación es porque este se compartía con un equipo que desarrolló un proyecto de Ciencia de Datos distinto, además de que está basado en predicciones y esto nos ayudaría mucho en entender qué predicciones son más certeras.



6 de junio

- Al igual que el día anterior, nos enfocamos en otros conceptos para explicarlos de manera más definida. Los últimos dos conceptos que presentamos fueron solo dos, y son los siguientes:

  - **Entropía**

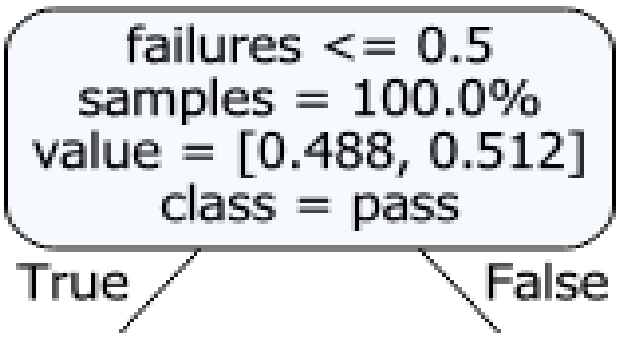


  - **Aplicación del proyecto:**

7-8 de junio

Durante estos días nos dedicamos al entendimiento del árbol de decisión y todos los elementos, entre los cuales se encuentran:

- **Variable.** En el lado izquierdo del primer elemento se encuentra el nombre de la variable que se va a utilizar en ese nodo. En el lado derecho está la condición que tiene que cumplir esta variable.
- **Muestras.** Este elemento muestra el porcentaje de todos los datos que se encuentran dentro de ese nodo en específico y va disminuyendo a medida que el árbol aumenta de profundidad.
- **Valor.** El tercer elemento representa el porcentaje de la población que pasó o falló la condición inicial establecida en la variable.
- **Clase.** En la clase se muestran 2 posibilidades, pass o fail dependiendo del resultado de la mayoría.

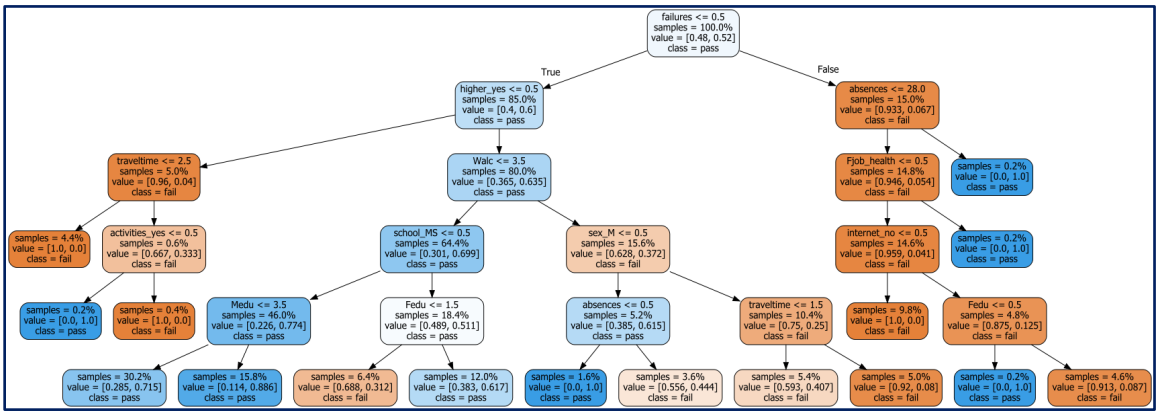


El color del nodo está relacionado directamente con la muestra, ya que dependiendo de la relación entre el pass o fail será la intensidad del color.

9 de junio

- Se realizó la séptima y última presentación del código, donde se interpretó el árbol de decisión y se compartieron las conclusiones finales relacionadas a :
  - Elementos del árbol de decisión (nodos, color, valores, class, etc).
  - Profundidad en los árboles de decisión.

Dentro del árbol de decisión desplegado, se compartieron los atributos más significativos para la predicción del desempeño académico de los estudiantes. Para el caso del árbol de decisión con **max_depth=5**, recomendado por el libro, se obtuvo lo siguiente:



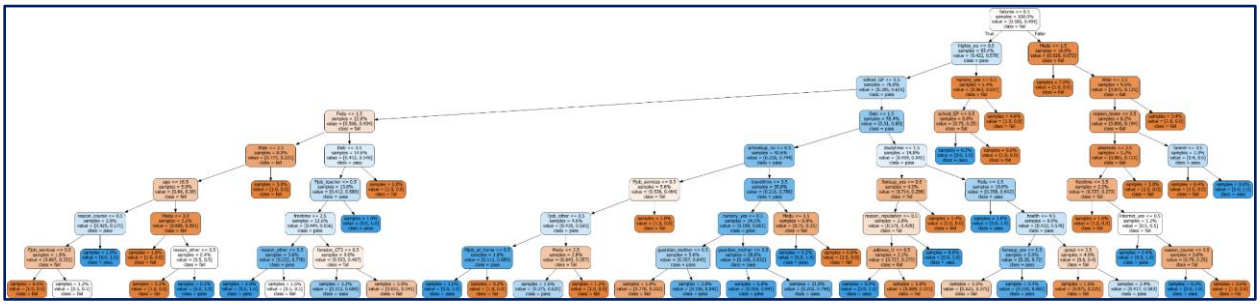
Atributos más significativos:

Atributo Más Relevante: Materias Reprobadas Pasadas

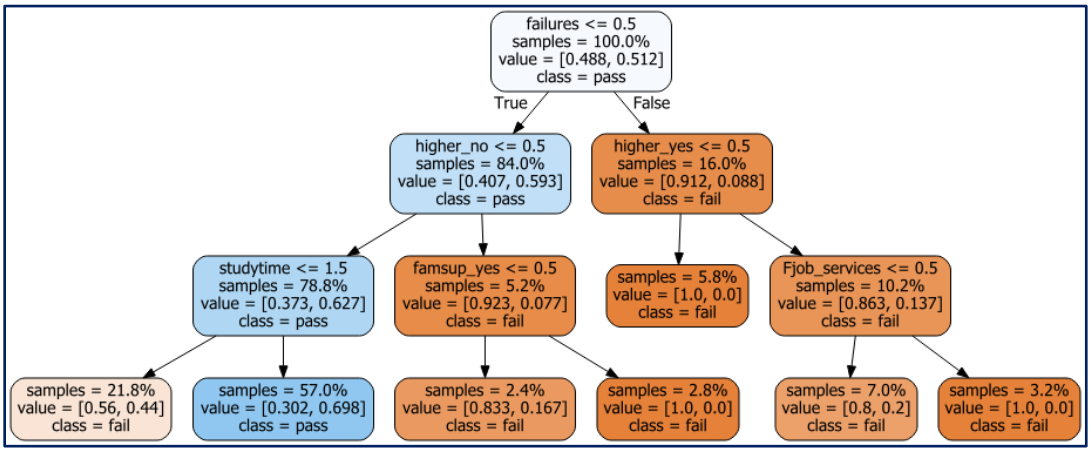
- Deseo de un mayor nivel educativo
- Ausencias pasadas
- Tiempo de traslado escuela-hogar
- Consumo de alcohol en fines de semana
- Trabajo del papá
- Actividades extracurriculares
- Escuela
- Sexo
- Acceso a internet
- Educación de los padres

- Asimismo, se compartieron árboles de decisión con otras profundidades, comparando cómo se visualiza un árbol de decisión y cómo funciona la entropía dependiendo de cuántas ramas se busca tener:

ÁRBOL DE DECISIÓN MAX_DEPTH=9



ÁRBOL DE DECISIÓN MAX_DEPTH=3

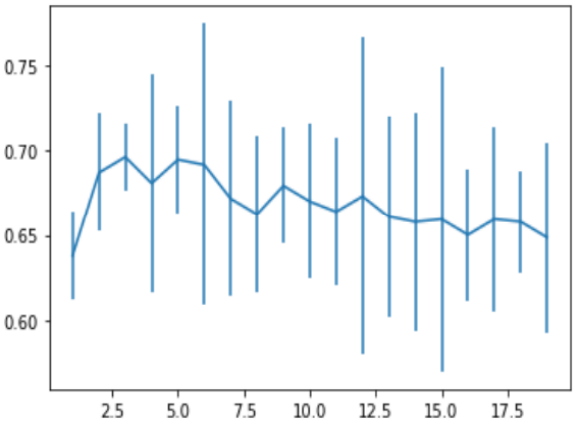


INTERPRETACIÓN Y CONCLUSIONES

Una vez concluido el código, analizando los resultados obtenidos en la precisión de los árboles de decisión, encontramos que el árbol de decisión idóneo era el de **max_depth=3**, pues tiene la mayor precisión de todos los posibles modelos:

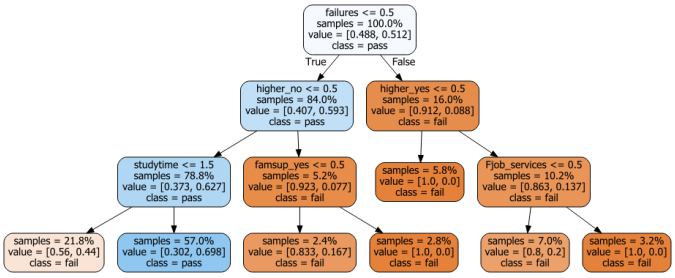
Con ello, corroboramos que mayor profundidad en un árbol de decisión no es sinónimo de mayor precisión, y que a partir de la entropía, la validación cruzada, la matriz de confusión y los elementos del árbol de decisión podemos predecir el desempeño académico de los estudiantes.

```
Max depth: 1, Accuracy: 0.64 (+/- 0.03)
Max depth: 2, Accuracy: 0.69 (+/- 0.04)
Max depth: 3, Accuracy: 0.70 (+/- 0.02)
Max depth: 4, Accuracy: 0.68 (+/- 0.06)
Max depth: 5, Accuracy: 0.69 (+/- 0.02)
Max depth: 6, Accuracy: 0.69 (+/- 0.08)
Max depth: 7, Accuracy: 0.68 (+/- 0.06)
Max depth: 8, Accuracy: 0.67 (+/- 0.06)
Max depth: 9, Accuracy: 0.68 (+/- 0.05)
Max depth: 10, Accuracy: 0.65 (+/- 0.06)
Max depth: 11, Accuracy: 0.69 (+/- 0.05)
Max depth: 12, Accuracy: 0.67 (+/- 0.06)
Max depth: 13, Accuracy: 0.66 (+/- 0.03)
Max depth: 14, Accuracy: 0.65 (+/- 0.08)
Max depth: 15, Accuracy: 0.65 (+/- 0.05)
Max depth: 16, Accuracy: 0.65 (+/- 0.03)
Max depth: 17, Accuracy: 0.66 (+/- 0.03)
Max depth: 18, Accuracy: 0.64 (+/- 0.05)
Max depth: 19, Accuracy: 0.64 (+/- 0.04)
```



Por tanto, como atributos más significativos para la predicción del desempeño de estudiantes tenemos:

1. Materias reprobadas pasadas
2. Deseo de un mayor nivel educativo
3. Tiempo de estudio
4. Apoyo familiar educacional
5. Trabajo del padre → servicios



BITÁCORA REALIZADA DÍA A DÍA

FECHA	PROCESO
16 de mayo	Conformación de equipos y selección del proyecto
18 de mayo	Elaboración de la bitácora inicial y presentación del proyecto • Proyecto • Introducción • Razón • Bitácora • Finalidad
19 de mayo	Primera exposición del equipo 4. Presentación de proyecto + bitácora. • Compromisos: Desarrollo del código
20 de mayo	Desarrollo del código en replit (plataforma colaborativa). Problemas: • No aparece el árbol de decisión • El árbol de decisión no puede guardarse
23 de mayo	Segunda exposición del equipo 4. Primera versión del código. Compromisos: Encontrar el error en el código ¿Cuáles serán las variables más significativas?
24-27 de mayo	Comparación de desarrollo del código con compañeros y equipos que ya han cursado la materia. Recomendaciones: • Buscar en foros de ayuda • Revisar librerías • Correr el código en distintas computadoras
26 de mayo	Tercera exposición del equipo 4. Explicación del código y búsqueda de problemas. Compromisos: Seguir buscando la solución.
29-30 de mayo	Código terminado: • Desarrollo del código • Búsqueda de ayuda en foros ○ https://stackoverflow.com/questions/52566756/no-

	module-named-graphviz-in-jupyter-notebook • Cambio de base de datos (online) ○ https://raw.githubusercontent.com/PacktPublishing/Python-Artificial-Intelligence-Projects-for-Beginners/master/Chapter01/dataset/student-por.csv • Despliegue del árbol de decisión • Guardado del árbol de decisión como imagen • Descarga de librerías (mostrar screenshot en prompt de anaconda)
30 de mayo	Cuarta exposición del equipo 4. Código terminado. Compromisos: Profundizar en el marco teórico del proyecto.
2 de junio	Quinta exposición del equipo 4. Marco teórico del proyecto. 1. Árboles de decisión 2. Cross-validation (validación cruzada) 3. Matriz de confusión Compromisos: Desarrollar la utilidad y áreas de aplicación para el código.
6 de junio	Sexta exposición del equipo 4. Marco teórico del proyecto. 1. Entropía 2. Aplicación del proyecto
7-8 de junio	Desarrollo y exploración del árbol de decisión. • Lectura • Interpretación • Variantes ¿Qué significan los colores? ¿Qué significan los valores por nodo? ¿Qué pasa si cambia la profundidad del árbol? ¿Cuál es la profundidad más certera?
9 de junio	Séptima presentación del equipo 4. Presentación del árbol de decisión y conclusiones finales. Explicación: • Atributos más significativos

	• Elementos del árbol de decisión (nodos, color, valores, class, etc). Conclusiones • La mejor precisión se encontró con max_depth=3 • Análisis de la gráfica de profundidad • Una mayor profundidad no es sinónimo de mayor precisión

CÓDIGO COMPLETO

```
Python 3.9.7 (default, Sep 16 2021, 16:59:28) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 7.29.0 -- An enhanced Interactive Python.

In [1]: #1
   ...: # load dataset
   ...: import pandas as pd
   ...: d = pd.read_csv('https://raw.githubusercontent.com/PacktPublishing/Python-
Artificial-Intelligence-Projects-for-Beginners/master/Chapter01/dataset/student-
por.csv', sep=';')
   ...: len(d)
Out[1]: 649

In [2]: #2
   ...: # generate binary label (pass/fail) based on G1+G2+G3 (test grades, each 0-20
pts); threshold for passing is sum>=30
   ...: d['pass'] = d.apply(lambda row: 1 if (row['G1']+row['G2']+row['G3']) >= 35 else
0, axis=1)
   ...: d = d.drop(['G1', 'G2', 'G3'], axis=1)
   ...: d.head()
Out[2]:
```

	school	sex	age	address	famsize	...	Dalc	Walc	health	absences	pass
0	GP	F	18	U	GT3	...	1	1	3	4	0
1	GP	F	17	U	GT3	...	1	1	3	2	0
2	GP	F	15	U	LE3	...	2	3	3	6	1
3	GP	F	15	U	GT3	...	1	1	5	0	1
4	GP	F	16	U	GT3	...	1	2	5	0	1

```
[5 rows x 31 columns]

In [3]: #3
   ...: # use one-hot encoding on categorical columns
   ...: d = pd.get_dummies(d, columns=['sex', 'school', 'address', 'famsize',
'Pstatus', 'Mjob', 'Fjob',
   ...:                                'reason', 'guardian', 'schoolsup', 'famsup',
'paid', 'activities',
   ...:                                'nursery', 'higher', 'internet', 'romantic'])
   ...: d.head()
Out[3]:
```

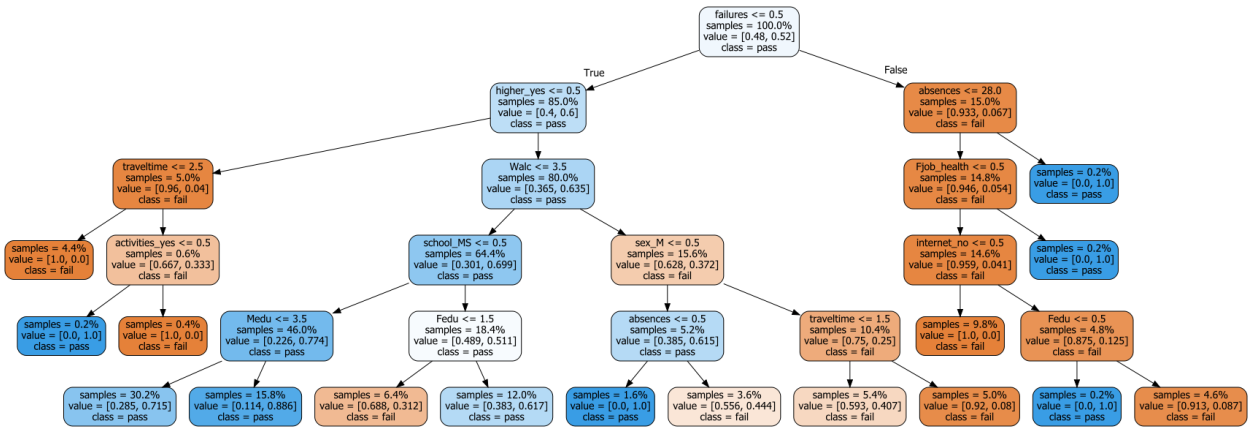
	age	Medu	Fedu	...	internet_yes	romantic_no	romantic_yes
0	18	4	4	...	0	1	0
1	17	1	1	...	1	1	0
2	15	1	1	...	1	1	0
3	15	4	2	...	1	0	1
4	16	3	3	...	0	1	0

```
[5 rows x 57 columns]

In [4]: #4
   ...: # shuffle rows
   ...: d = d.sample(frac=1)
   ...: # split training and testing data
   ...: d_train = d[:500]
   ...: d_test = d[500:]
   ...:
   ...: d_train_att = d_train.drop(['pass'], axis=1)
   ...: d_train_pass = d_train['pass']
   ...:
   ...: d_test_att = d_test.drop(['pass'], axis=1)
   ...: d_test_pass = d_test['pass']
   ...:
   ...: d_att = d.drop(['pass'], axis=1)
   ...: d_pass = d['pass']
   ...:
   ...: # number of passing students in whole dataset:
   ...: import numpy as np
   ...: print("Passing: %d out of %d (%.2f%%)" % (np.sum(d_pass), len(d_pass),
100*float(np.sum(d_pass)) / len(d_pass)))
Passing: 328 out of 649 (50.54%)
```

```
In [5]: #5
   ...: # fit a decision tree
   ...: from sklearn import tree
   ...: t = tree.DecisionTreeClassifier(criterion="entropy", max_depth=5)
   ...: t = t.fit(d_train_att, d_train_pass)

In [6]: #6
   ...: # visualize tree
   ...: import graphviz
   ...: dot_data = tree.export_graphviz(t, out_file=None, label="all", impurity=False,
proportion=True,
   ...:                                 feature_names=list(d_train_att),
class_names=["fail", "pass"],
   ...:                                 filled=True, rounded=True)
   ...: graph = graphviz.Source(dot_data)
   ...: graph
Out[6]:
```



```
In [7]: # save tree
   ...: tree.export_graphviz(t, out_file="student-performance.dot", label="all",
impurity=False, proportion=True,
   ...:                      feature_names=list(d_train_att), class_names=["fail",
"pass"],
   ...:                      filled=True, rounded=True)

In [8]: #8
   ...: t.score(d_test_att, d_test_pass)
Out[8]: 0.6510067114093959

In [9]: #9
   ...: from sklearn.model_selection import cross_val_score
   ...: scores = cross_val_score(t, d_att, d_pass, cv=5)
   ...: # show average score and +/- two standard deviations away (covering 95% of
scores)
   ...: print("Accuracy: %0.2f (+/- %0.2f)" % (scores.mean(), scores.std() * 2))
Accuracy: 0.70 (+/- 0.03)

In [10]: #10
    ...: for max_depth in range(1, 20):
    ...:     t = tree.DecisionTreeClassifier(criterion="entropy", max_depth=max_depth)
    ...:     scores = cross_val_score(t, d_att, d_pass, cv=5)
    ...:     print("Max depth: %d, Accuracy: %0.2f (+/- %0.2f)" % (max_depth,
scores.mean(), scores.std() * 2))
```

```
Max depth: 1, Accuracy: 0.64 (+/- 0.03)
Max depth: 2, Accuracy: 0.69 (+/- 0.04)
Max depth: 3, Accuracy: 0.70 (+/- 0.02)
Max depth: 4, Accuracy: 0.68 (+/- 0.06)
Max depth: 5, Accuracy: 0.69 (+/- 0.02)
Max depth: 6, Accuracy: 0.69 (+/- 0.08)
Max depth: 7, Accuracy: 0.68 (+/- 0.06)
Max depth: 8, Accuracy: 0.67 (+/- 0.06)
Max depth: 9, Accuracy: 0.68 (+/- 0.05)
Max depth: 10, Accuracy: 0.65 (+/- 0.06)
Max depth: 11, Accuracy: 0.69 (+/- 0.05)
Max depth: 12, Accuracy: 0.67 (+/- 0.06)
Max depth: 13, Accuracy: 0.66 (+/- 0.03)
Max depth: 14, Accuracy: 0.65 (+/- 0.08)
Max depth: 15, Accuracy: 0.65 (+/- 0.05)
Max depth: 16, Accuracy: 0.65 (+/- 0.03)
Max depth: 17, Accuracy: 0.66 (+/- 0.03)
Max depth: 18, Accuracy: 0.64 (+/- 0.05)
Max depth: 19, Accuracy: 0.64 (+/- 0.04)
```

```
In [11]: #11
    ...: depth_acc = np.empty((19,3), float)
    ...: i = 0
    ...: for max_depth in range(1, 20):
    ...:     t = tree.DecisionTreeClassifier(criterion="entropy", max_depth=max_depth)
    ...:     scores = cross_val_score(t, d_att, d_pass, cv=5)
    ...:     depth_acc[i,0] = max_depth
    ...:     depth_acc[i,1] = scores.mean()
    ...:     depth_acc[i,2] = scores.std() * 2
    ...:     i += 1
    ...:
    ...: depth_acc
```

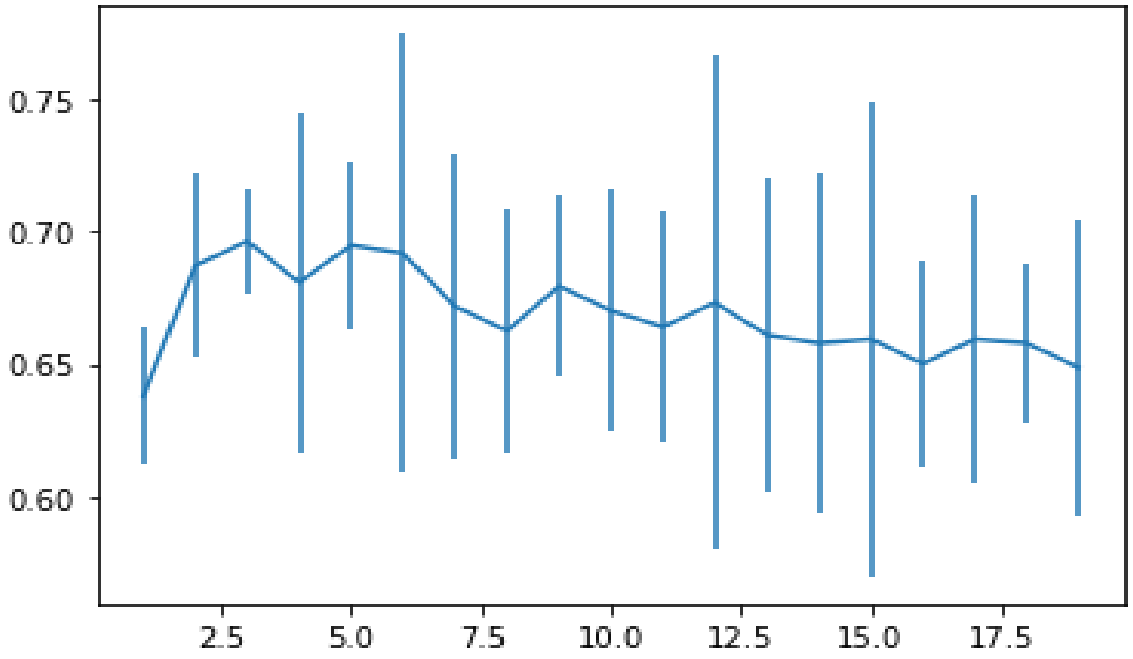
```
Out[11]:
array([[ 1.        ,  0.63790101,  0.02584084],
       [ 2.        ,  0.68720334,  0.03476724],
       [ 3.        ,  0.69643411,  0.01981914],
       [ 4.        ,  0.68100179,  0.06447199],
       [ 5.        ,  0.6949195 ,  0.03146169],
       [ 6.        ,  0.69184258,  0.08298242],
       [ 7.        ,  0.67186643,  0.05758592],
       [ 8.        ,  0.66262373,  0.04638525],
       [ 9.        ,  0.67951103,  0.03433177],
       [10.        ,  0.67024448,  0.04548837],
       [11.        ,  0.66412642,  0.0433115 ],
       [12.        ,  0.67326178,  0.09341119],
       [13.        ,  0.66101371,  0.05922301],
       [14.        ,  0.65791294,  0.06470426],
       [15.        ,  0.65946333,  0.08956317],
       [16.        ,  0.65022063,  0.03875557],
       [17.        ,  0.65947525,  0.05450439],
       [18.        ,  0.65793679,  0.03000608],
       [19.        ,  0.64868217,  0.05571992]])

In [12]: #12
    ...: import matplotlib.pyplot as plt
    ...: fig, ax = plt.subplots()
```

```
    ...: ax.errorbar(depth_acc[:,0], depth_acc[:,1], yerr=depth_acc[:,2])
    ...: plt.show()
```

Warning

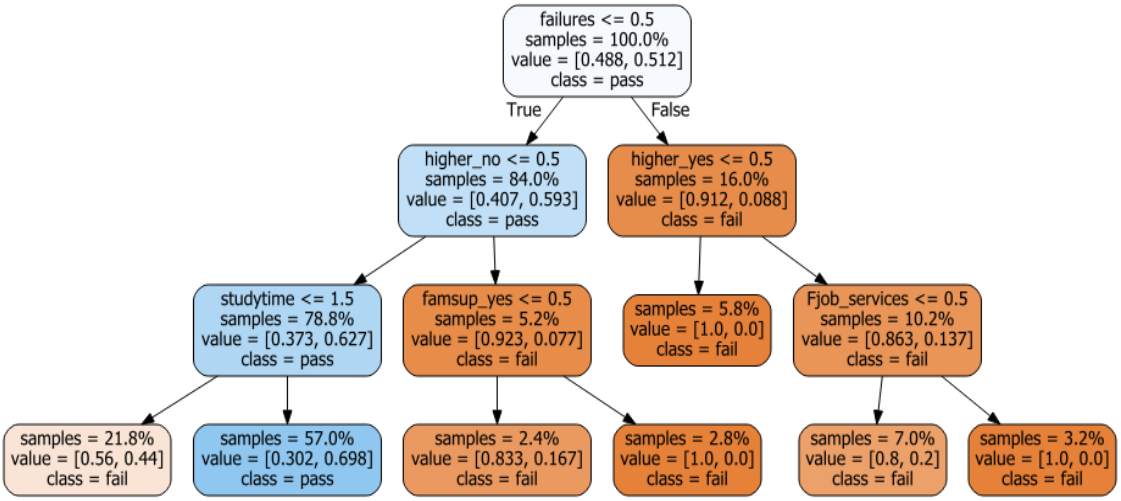
Figures now render in the Plots pane by default. To make them also appear inline in the Console, uncheck "Mute Inline Plotting" under the Plots pane options menu.



```
In [13]: #max_depth=3
    ...: # fit a decision tree
    ...: from sklearn import tree
    ...: t = tree.DecisionTreeClassifier(criterion="entropy", max_depth=3)
    ...: t = t.fit(d_train_att, d_train_pass)

In [14]: # visualize tree
    ...: import graphviz
    ...: dot_data = tree.export_graphviz(t, out_file=None, label="all", impurity=False,
proportion=True,
    ...:                                 feature_names=list(d_train_att),
```

```
    ...:                                 feature_names=list(d_train_att),
class_names=["fail", "pass"],
    ...:                                 filled=True, rounded=True)
    ...: graph = graphviz.Source(dot_data)
    ...: graph
Out[14]:
```



COMENTARIOS FINALES

Fernanda Guzmán

Este proyecto me permitió poner en práctica los conocimientos de Ciencia de Datos que aprendí durante el semestre, y asimismo, conocer otros conceptos importantes dentro de la predicción y el modelaje basado en datos. Me gustó que el resultado fuera útil para mí al revelarme atributos importantes en el desempeño académico. Creo que el nivel de dificultad fue bueno para mis conocimientos actuales, y el libro ayudó mucho al desglosar e interpretar los datos.

Asimismo, me gustó trabajar con mi equipo y que cada uno aportara sus fortalezas hacia el proyecto, encontrando errores de código, entendiendo el funcionamiento de árboles de decisión, entre otros.

Frida Carmona

A través del proyecto elaborado anteriormente, fui capaz de desarrollar distintas habilidades y aprendizajes relacionados con la programación y la ciencia de datos, lo cual considero bastante importante en mi desarrollo profesional y personal, ya que los conocimientos adquiridos pueden ser aplicados en varios escenarios y en la resolución de problemas de la vida real.

Mi equipo de trabajo me gustó bastante porque nos complementamos bastante bien unos con otros, distribuyendo el trabajo de tal forma que cada uno estuvo enfocado en lo que era bueno, es decir, cada quién aportó en su área de fortaleza. Así pues, en conjunto logramos desarrollar óptima y efectivamente un proyecto completo y funcional.

Sofia Rico Casanova

Considero que, este proyecto final me permitió poner en práctica todos los temas y conocimientos adquiridos a lo largo del semestre en relación a la Ciencia de Datos, y no solo eso, sino que también, me permitió poder obtener nuevos conocimientos, como lo fueron los árboles de decisión, la matriz de confusión y el cross-validation, entre otros. La verdad que me agradó poder realizar este proyecto, porque considero que me fue útil para comprobar lo aprendido en clase.

Me gustaría mencionar que, me ha gustado mucho poder trabajar con mi equipo, ya que considero que nos complementamos bien, pues cada uno aportó algo significativo al proyecto de acuerdo con sus habilidades y aptitudes, es por ello que, se pudo obtener algo como lo presentado anteriormente. De igual manera, puedo decir que, la participación de todos fue increíble, ya que, todos estuvimos comprometidos con el desarrollo del proyecto.

Marcelo López

Este proyecto me ayudó a conocer y apreciar un aspecto sobre programación, y fueron los árboles de decisiones y en todos los aspectos flexibles que se pueden aplicar en diferentes áreas de trabajo y de estudio.

Aunque yo no hice un trabajo al nivel de los compañeros, agradezco todo lo que hicieron, aportamos cada quien una perspectiva diferente, y en general siento que todos hicimos un muy buen trabajo.

Ginno Aguilar

Durante el proyecto todos los integrantes del equipo fueron participativos y mostraron interés en el desarrollo del código. La repartición de tareas fue sencilla y equitativa entre los miembros, además de que nos apoyamos entre nosotros.

Yo veo mucha utilidad para este proyecto en específico, sobre todo en aplicaciones en diferentes campos laborales donde podremos facilitar la toma de decisiones al crear este tipo de diagramas para la visualización de información relevante. Agregado a esto es una introducción a un tema aún más amplio con infinitas aplicaciones que pueden hacer mucho más efectivo el trabajo de todas las personas.