Día 1 - 15/11/21 (Lunes)

Introducción

Teórica y Práctica

Lic. Ronaldo Armando Canizales Turcios

UNIVERSIDAD DE GRANADA

Universidad Centroamericana José Simeón Cañas

Agenda Día 1

Bloque A

- Desmitificando el aprendizaje de máquina: conceptos relacionados.

- ¿Qué tipo de problemas de la vida real pueden resolverse con IA?

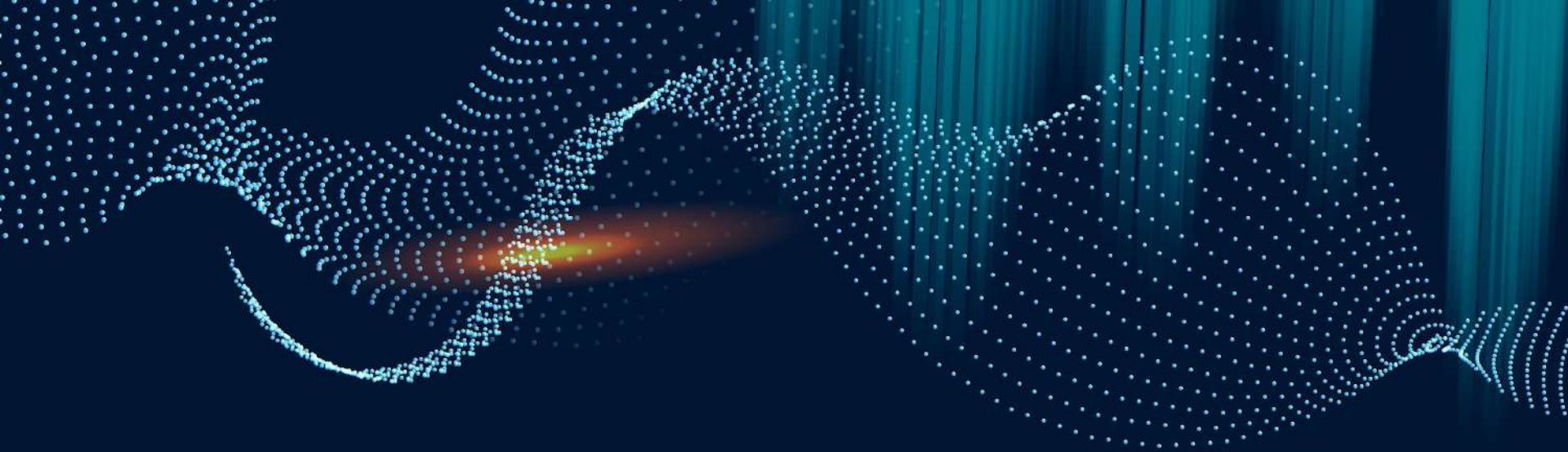
- Etapas de un proyecto de Ciencia de Datos y ML.

Bloque B

- Un vistazo empírico al aprendizaje de máquina.

- Árbol de decisión utilizando datos en tiempo real.

- Mi primer árbol de decisión: clasificar perfiles académicos. **[Práctica]**



01 Desmitificando el ML

Conceptos relacionados

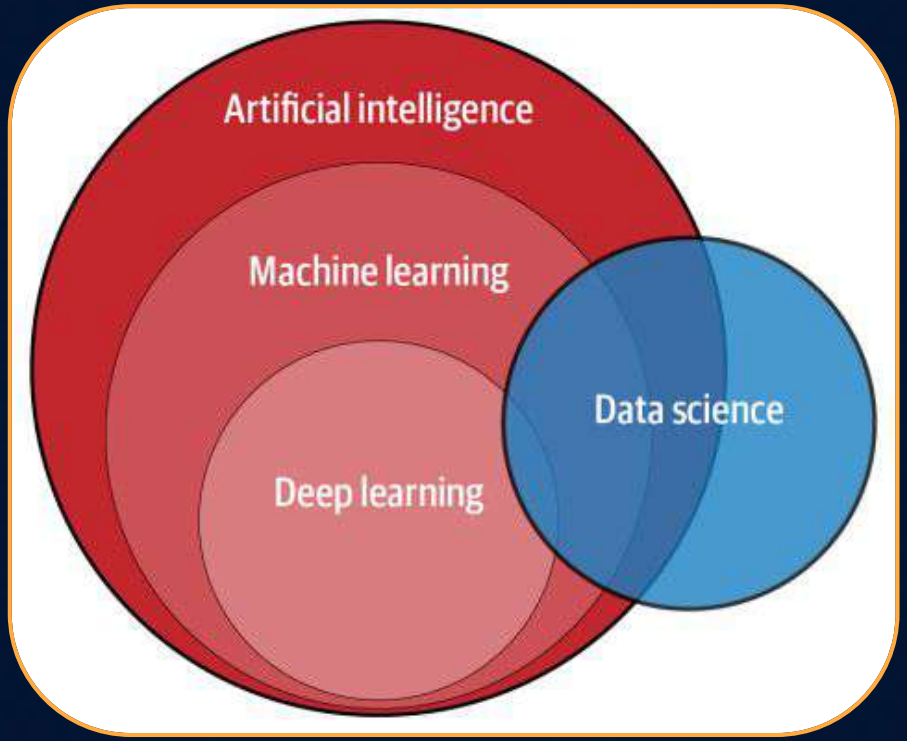
Ciencia de Datos (DS)

Inteligencia Artificial (AI)

Aprendizaje de Máquina (ML)

Aprendizaje Profundo (DL)

DS vs AI vs ML vs DL



La inteligencia artificial es la ciencia que estudia la forma de hacer que una computadora desarrolle la capacidad de realizar con éxito tareas complejas que generalmente requieren inteligencia humana.

- Percepción visual
- Reconocimiento de voz
- Toma de decisiones
- Traducción entre idiomas
- ¡Muchas más!

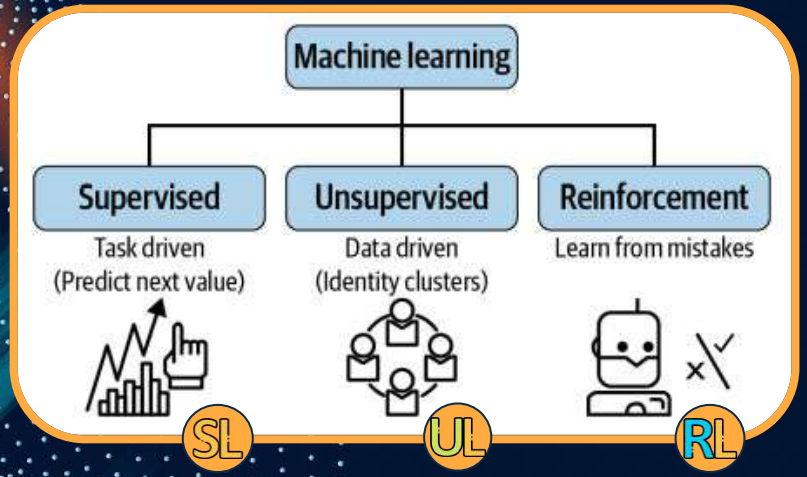
DS vs AI vs ML vs DL

El aprendizaje de máquina (ML) es una aplicación de la IA. Proporciona la capacidad de aprender automáticamente del entorno y aplicar esas lecciones para tomar mejores decisiones.

El aprendizaje profundo (DL) es un subconjunto del ML que implica el estudio de algoritmos relacionados con redes neuronales artificiales. Permite resolver problemas más complejos.

La ciencia de datos (DS) es un campo interdisciplinario similar a la minería de datos. Se encarga de extraer información de los datos en diversas formas, ya sea estructuradas o no.

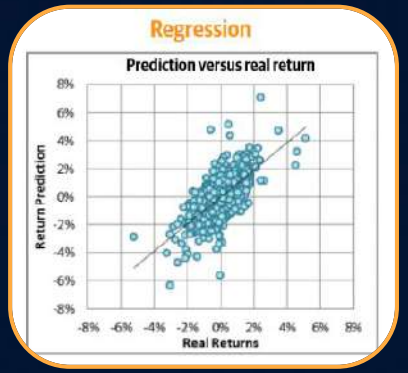
Tipos de Aprendizaje de Máquina



SL

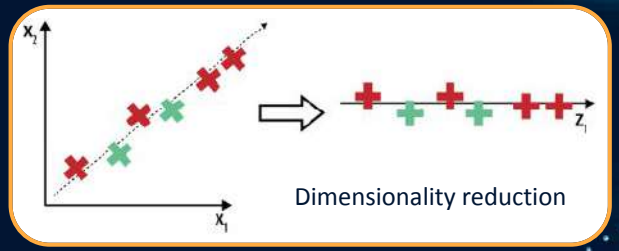
El objetivo principal del aprendizaje supervisado es entrenar un modelo a partir de datos etiquetados que nos permita hacer predicciones sobre datos futuros o no vistos.

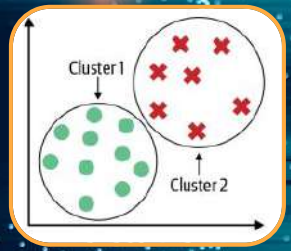
Aquí, el término supervisado se refiere a un conjunto de muestras donde ya se conocen las señales de salida deseadas (etiquetas).



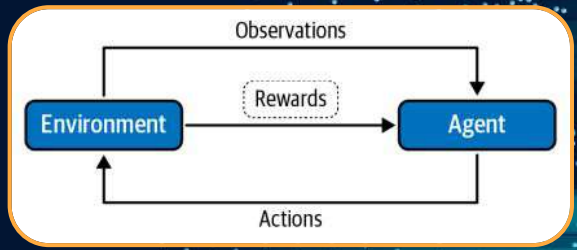
Tipos de Aprendizaje de Máquina

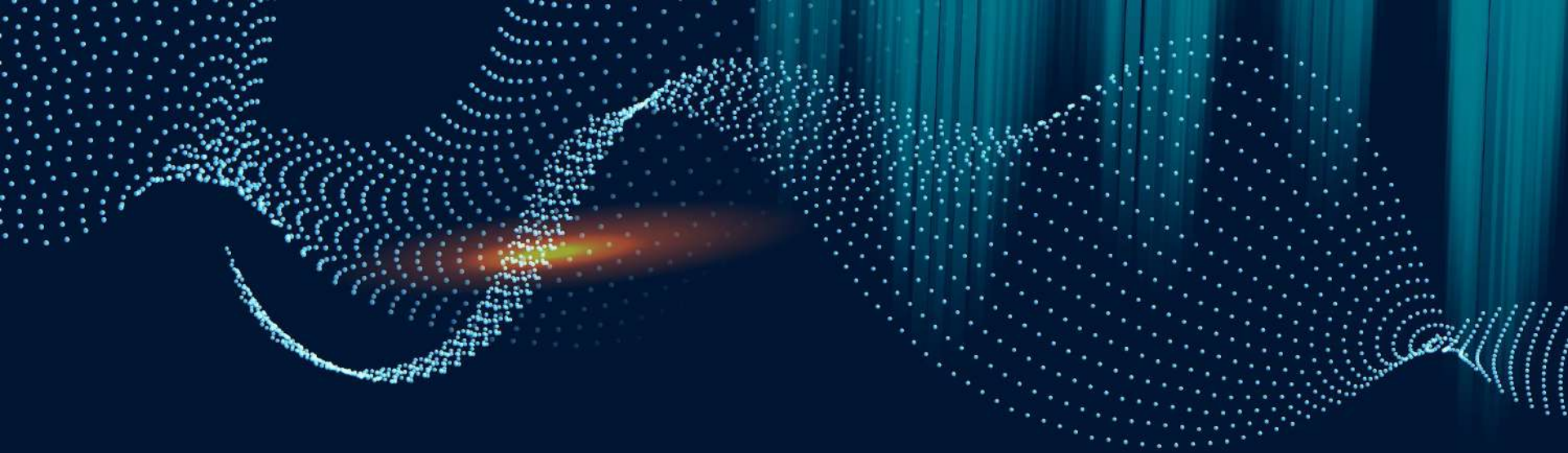
El aprendizaje no supervisado es un tipo de aprendizaje automático que se utiliza para extraer inferencias de conjuntos de datos que consisten en datos de entrada sin respuestas etiquetadas.





Aprender de la experiencia, las recompensas o castigos es el concepto detrás del aprendizaje por refuerzo. Se trata de tomar las acciones adecuadas para maximizar la recompensa en situaciones particulares donde no hay una respuesta explícita.





02 Problemas de la vida real

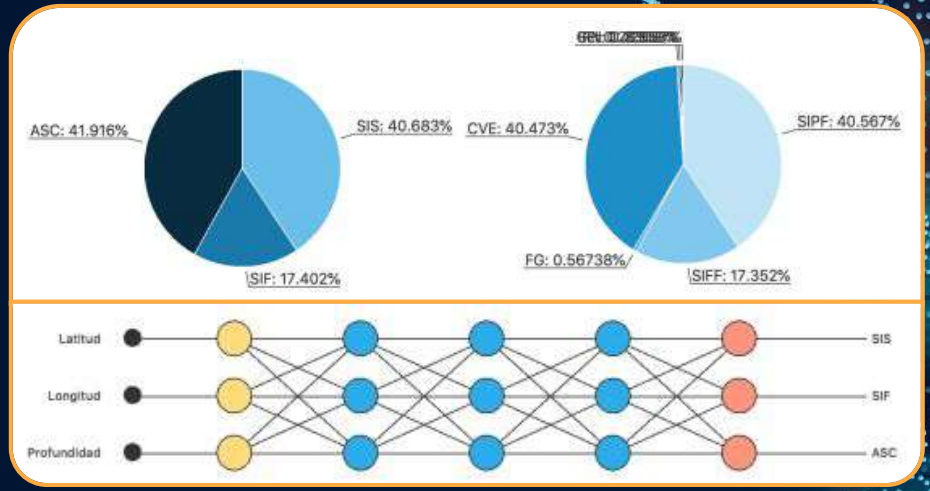
Que pueden resolverse con IA

Caso I: Ambientes tectónicos y fuentes sísmicas

Tipo de problema: clasificación automática para sistematización.

Datos estructurados (fuente: Excel).

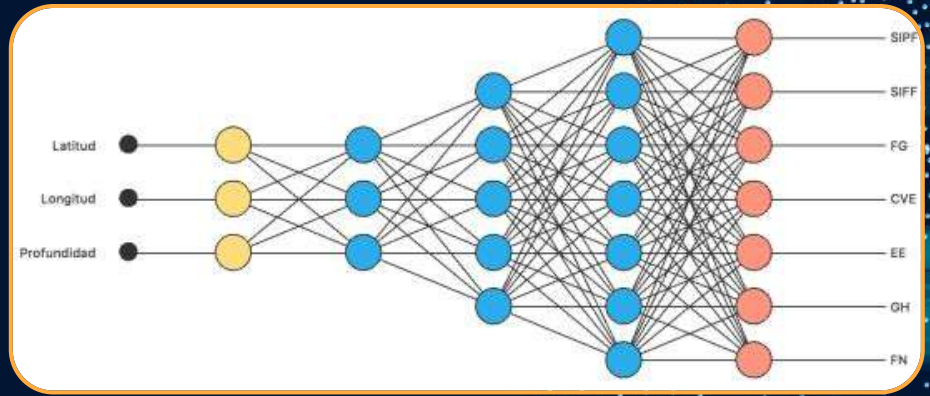
Técnica utilizada: Redes de Neuronas Artificiales (perceptrones multicapa).



97.13% (406/418)

96.20% (402/418)

	Predicho SIS	Predicho SIF	Predicho ASC
Real SIS	**167**	6	0
Real SIF	6	**63**	0
Real ASC	0	0	**176**

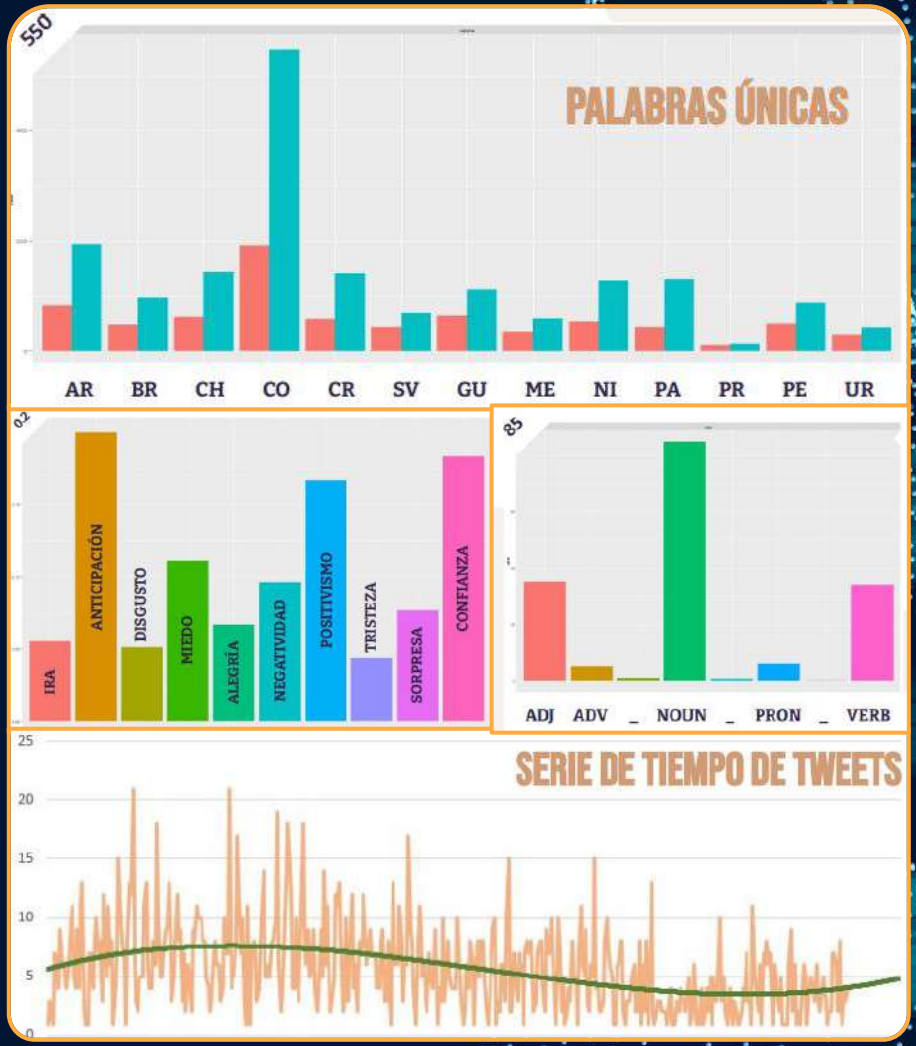


Caso II: Percepción de la cultura digital en redes sociales

Tipo de problema: descubrir patrones de comportamiento.

Datos no estructurados (fuente: Twitter).

Técnica utilizada: Procesamiento de lenguaje natural, análisis de sentimiento y regresión polinomial (extrapolación).



Trabajo de grado realizado por:
Salvador Campos, Jorge Franco, Gerardo Mira y Sara Romero.

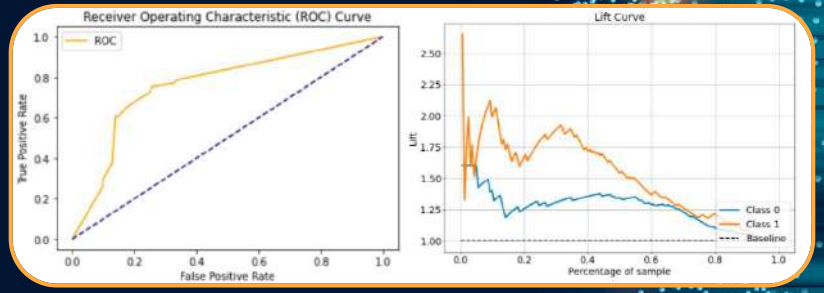
Caso III: Análisis de la efectividad de evaluaciones cualitativas

Tipo de problema: regresión para calificación semiautomática.

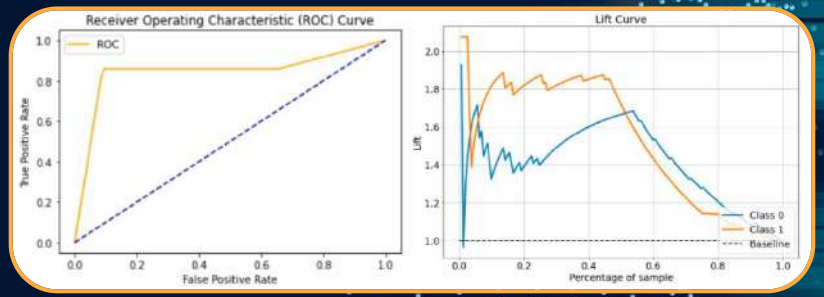
Datos no estructurados (Google Forms).

Técnica utilizada: Árbol de decisión.

Preg. 4



Preg. 7

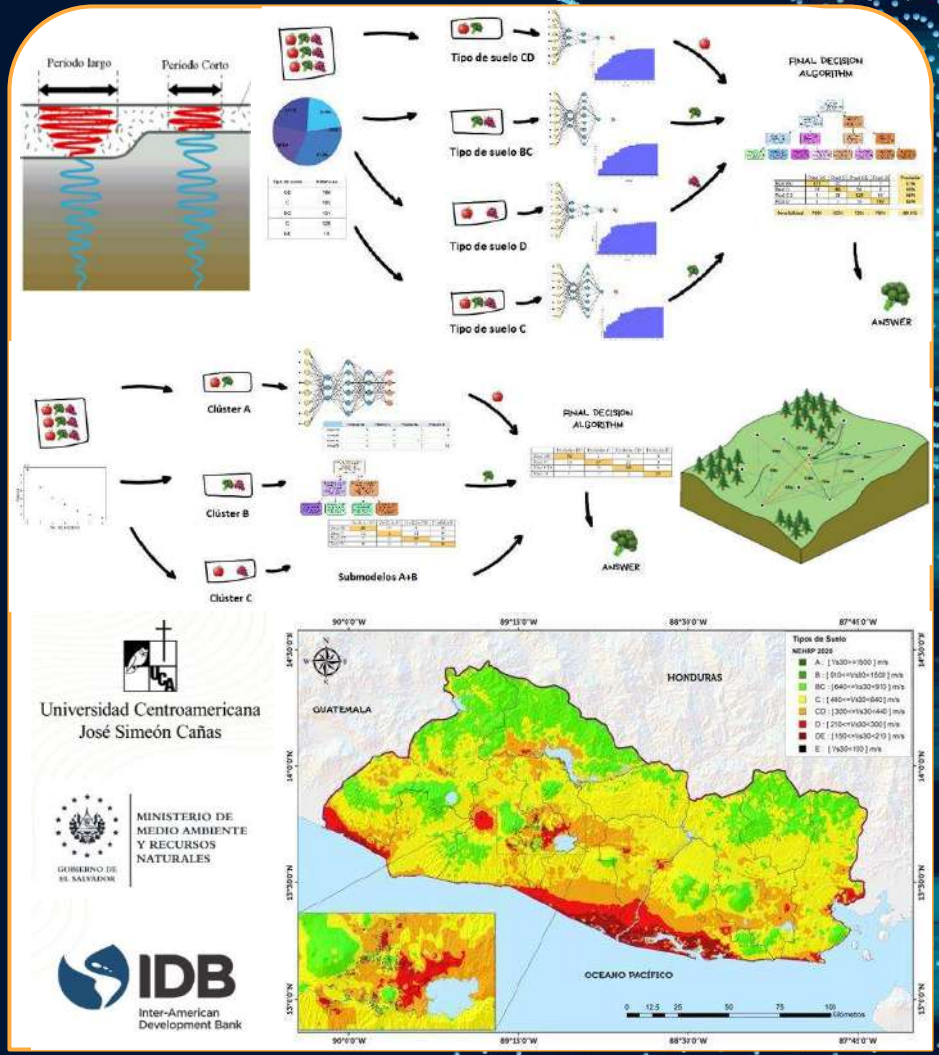


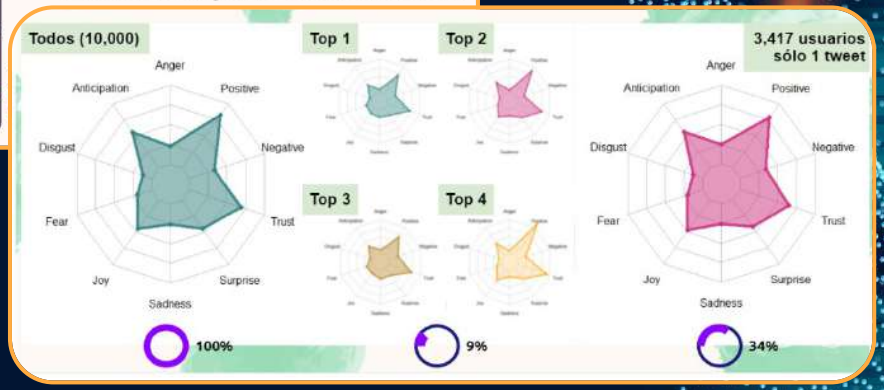
Caso IV: Mapa de respuesta sísmica homogénea de El Salvador

Tipo de problema: regresión y clasificación para generación de mapa más preciso.

Datos estructurados (fuente: diversas).

Técnica utilizada: Métodos de ensamble (mix de stacking, bosques aleatorios, bagging, clusterización y tamiz).



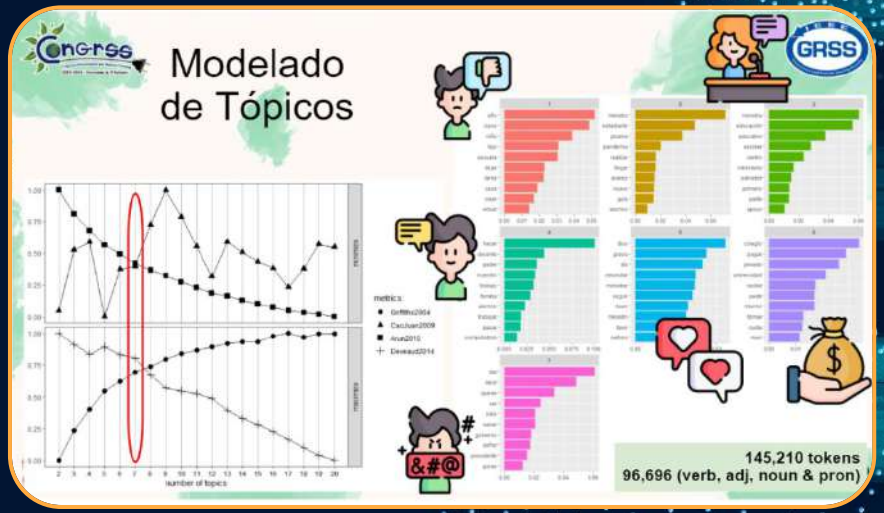


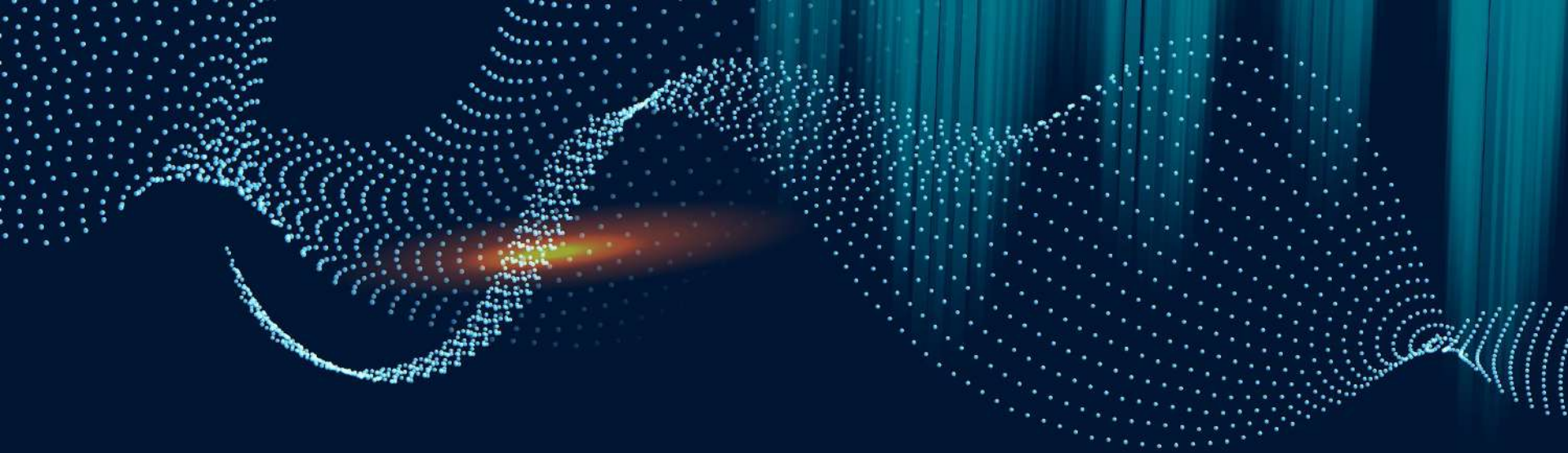
Caso V: Análisis del discurso en redes sociales sobre la relación educativa

Tipo de problema: Descubrir patrones de comportamiento.

Datos no estructurados (Fuente: twitter).

Técnica utilizada: Análisis de sentimiento y modelado de tópicos.

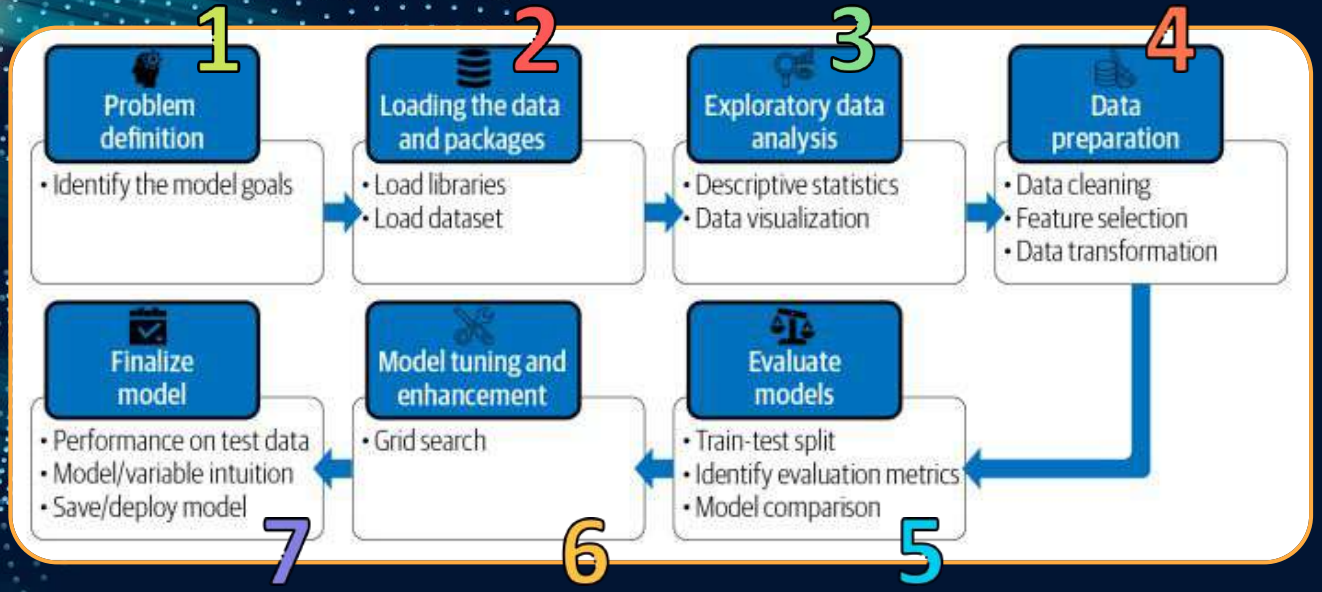




03 Etapas de un proyecto

Ciencia de Datos y Aprendizaje de Máquina

Ciencia de datos

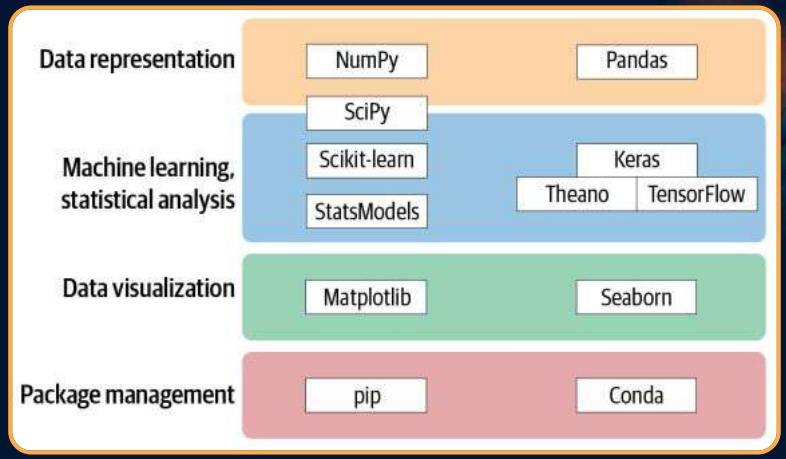


Aprendizaje de Máquina

1 Definición del problema

Se pueden utilizar potentes algoritmos para resolver el problema, pero los resultados no tendrán sentido si el problema incorrecto es resuelto.

- Describa el problema de manera informal y formal.
- Enumere suposiciones y problemas similares.
- Describa cómo se resolvería el problema utilizando “técnicas tradicionales” conocimiento propio de la disciplina.



Carga de datos y librerías 2

¡Manos a la obra!

¡Tarea en pareja para el descanso!

Conseguir un **recipiente** que **usted** considere una **botella**

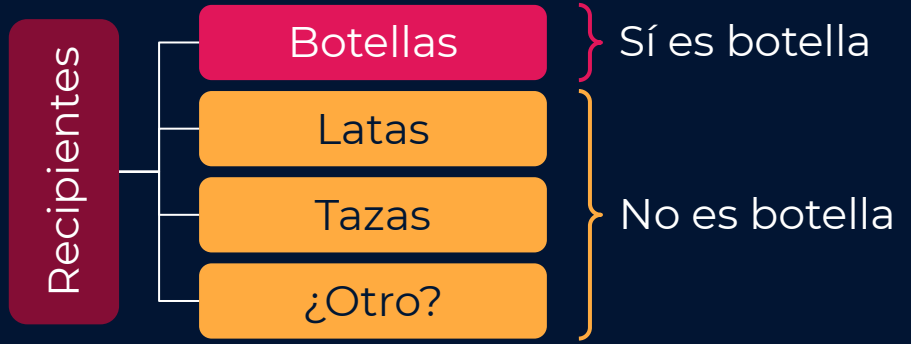
Conseguir un **recipiente** que **usted** considere que **NO es una botella**

04 Un vistazo empírico

Al Aprendizaje de Máquina

¿Que diferencia una botella de “los demás recipientes”?

Para un ser humano es sencillo diferenciarlas, pero no lo es para una máquina ¿o sí?



¡Inventemos reglas de clasificación!

¿Qué criterios ocupamos?

- ¿Tamaño?
- ¿Material?
- ¿Forma?
- ¿Propósito?
- ¿Algún otro?

El conjunto de aprendizaje debe poseer las siguientes características:

Ser significativo

Debe haber un número suficiente de ejemplos.

No podrá generalizar adecuadamente.

Ser representativo

El conjunto debe ser diverso.

Se debe evitar tener muchos más ejemplos de un tipo que del resto.

La finalización del periodo de aprendizaje se puede determinar:

Mediante un número fijo de iteraciones

Cuando el error descienda por debajo de una cantidad preestablecida

Cuando la modificación del modelo sea irrelevante

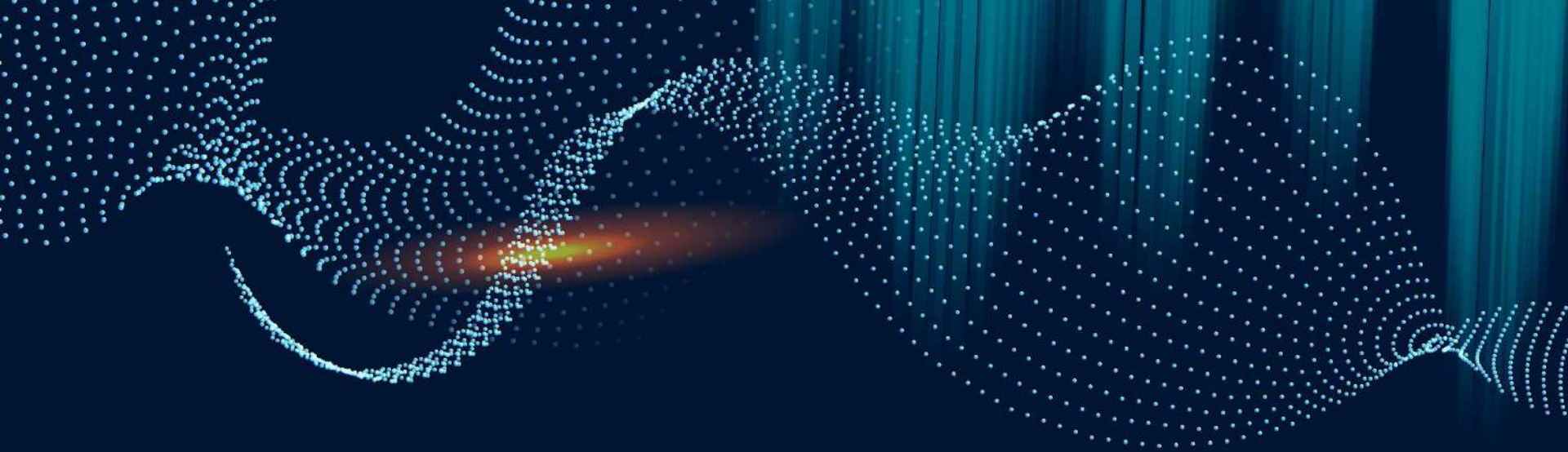
Reglas de oro

Nunca “casarse” con un solo algoritmo, siempre probar varios y elegir el mejor.

No siempre el algoritmo más sofisticado será el adecuado: “No es necesaria una bazuca para matar hormigas” - W. Otoniel.

¿Qué sucedería si...?

¡Cambiar de método!
Elegir centroides



05 Árbol de decisión

Utilizando datos en “tiempo real”

06 Mi primer árbol de decisión

Clasificar perfiles académicos **[Práctica]**

Utilizaremos **árboles de decisión** para predecir el desempeño académico de algunos estudiantes, basados en su desempeño previo.

Nuestro conjunto de datos describe **30 atributos de 649 estudiantes**. Hay datos mixtos, tanto **numéricos** como **categóricos**.

https://archive.ics.uci.edu/ml/datasets/student+performance

1 school - student's school (binary: 'GP' - Gabriel Pereira or 'MS' - Mousinho da Silveira)

2 sex - student's sex (binary: 'F' - female or 'M' - male)

3 age - student's age (numeric: from 15 to 22)

4 address - student's home address type (binary: 'U' - urban or 'R' - rural)

5 famsize - family size (binary: 'LE3' - less or equal to 3 or 'GT3' - greater than 3)

6 Pstatus - parent's cohabitation status (binary: 'T' - living together or 'A' - apart)

7 Medu - mother's education (numeric: 0 - none, 1 - primary education (4th grade), 2 - 5th to 9th grade, ...

8 Fedu - father's education (numeric: 0 - none, 1 - primary education (4th grade), 2 - 5th to 9th grade, ...

9 Mjob - mother's job (nominal: 'teacher', 'health' care related, civil 'services' (e.g. administrative or police), 'at_home' or 'other')

10 Fjob - father's job (nominal: 'teacher', 'health' care related, civil 'services' (e.g. administrative or police), 'at_home' or 'other')

11 reason - reason to choose this school (nominal: close to 'home', school 'reputation', 'course' preference or 'other')

12 guardian - student's guardian (nominal: 'mother', 'father' or 'other')

13 traveltime - home to school travel time (numeric: 1 - <15 min., 2 - 15 to 30 min., 3 - 30 min. to 1 hour, or 4 - >1 hour)

14 studytime - weekly study time (numeric: 1 - <2 hours, 2 - 2 to 5 hours, 3 - 5 to 10 hours, or 4 - >10 hours)

15 failures - number of past class failures (numeric: n if 1<=n<3, else 4)

16 schoolsup - extra educational support (binary: yes or no)

17 famsup - family educational support (binary: yes or no)

18 paid - extra paid classes within the course subject (Math or Portuguese) (binary: yes or no)

19 activities - extra-curricular activities (binary: yes or no)

20 nursery - attended nursery school (binary: yes or no)

21 higher - wants to take higher education (binary: yes or no)

22 internet - Internet access at home (binary: yes or no)

23 romantic - with a romantic relationship (binary: yes or no)

24 famrel - quality of family relationships (numeric: from 1 - very bad to 5 - excellent)

25 freetime - free time after school (numeric: from 1 - very low to 5 - very high)

26 goout - going out with friends (numeric: from 1 - very low to 5 - very high)

27 Dalc - workday alcohol consumption (numeric: from 1 - very low to 5 - very high)

28 Walc - weekend alcohol consumption (numeric: from 1 - very low to 5 - very high)

29 health - current health status (numeric: from 1 - very bad to 5 - very good)

30 absences - number of school absences (numeric: from 0 to 93)

G1 + G2 + G3 = 35

```
from google.colab import drive
drive.mount('/content/gdrive')
```

```
Mounted at /content/gdrive
```

Luego de descargar los datos, es necesario darle permiso al Google Colab de acceder a nuestro Drive

```
import pandas as pd
#URL: https://archive.ics.uci.edu/ml/datasets/student+performance
d = pd.read_csv('/content/gdrive/MyDrive/Granada/Dia1_Cod3_Practica/student-por.csv', sep=';')
len(d)
```

```
649
```

Con la función read_csv se carga el archivo en la memoria, éste contiene 649 registros

```
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()
```

	school	sex	age	address	famsize	Pstatus	Medu	Fedu	Mjob	Fjob	reason	guardian
0	GP	F	18	U	GT3	A	4	4	at_home	teacher	course	mother
1	GP	F	17	U	GT3	T	1	1	at_home	other	course	father
2	GP	F	15	U	LE3	T	1	1	at_home	other	other	mother
3	GP	F	15	U	GT3	T	4	2	health	services	home	mother
4	GP	F	16	U	GT3	T	3	3	other	other	home	father

Ahora crearemos nuestra variable objetivo, que llamaremos pass. Será binaria: 1 si el estudiante aprueba con al menos 35 puntos y 0 caso contrario.

```
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()
```

sex_F	sex_M	school_GP	school_MS
1	0	1	0
1	0	1	0
1	0	1	0
1	0	1	0
1	0	1	0

¿Cómo lidiar con las variables categóricas? Un enfoque es el "one-hot encoding" similar a conexión de cables en un circuito.

```
d = d.sample(frac=1)
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']
```

Tomaremos 500 datos para entrenamiento y luego el resto (149) para prueba.

```
import numpy as np
print("Aprueban %d de %d (%.2f%%)" % (np.sum(d_pass), len(d_pass), 100*float(np.sum(d_pass))/len(d_pass)))
```

```
Aprueban 328 de 649 (50.54%)
```

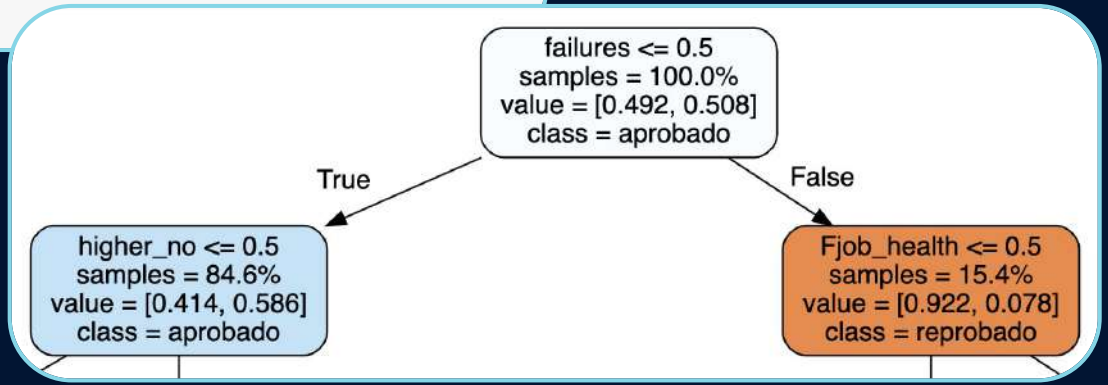
```
from sklearn import tree
t = tree.DecisionTreeClassifier(criterion="entropy", max_depth=5)
t = t.fit(d_train_att, d_train_pass)
```

Realizar el entrenamiento es cosa sencilla (utilizando la librería)

```
import graphviz
data=tree.export_graphviz(t, out_file=None,
  label='all', impurity=False, proportion=True,
  feature_names=list(d_train_att),
  class_names=['reprobado', 'aprobado'],
  filled=True, rounded=True)

graph = graphviz.Source(data, format="png")
graph
```

Para visualizar nuestro árbol, usaremos la librería graphviz

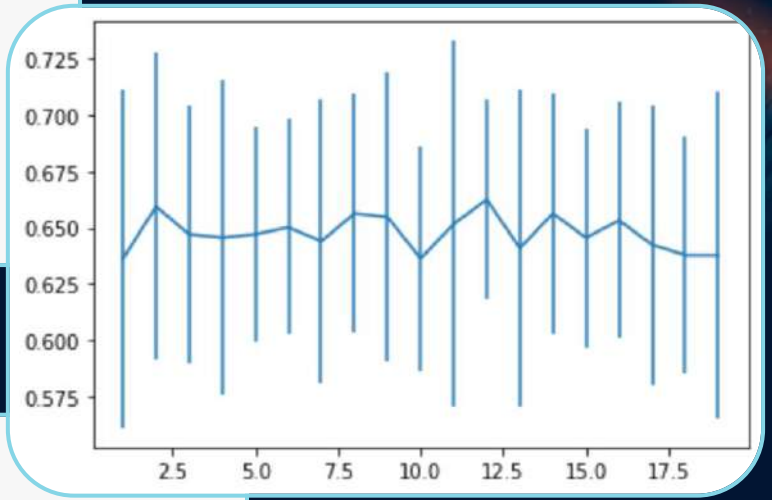



```
t.score(d_test_att, d_test_pass)

0.6577181208053692
```

```python
from sklearn.model_selection import cross_val_score

depth_acc = np.empty((19,3), float)
i = 0
for max_depth in range(1, 20):
  t=tree.DecisionTreeClassifier(criterion='entropy')
  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
```



Ahora, si algo está bien hecho ¿por qué no buscar mejorarlo aún más?

```python
import matplotlib.pyplot as plt
fig, ax = plt.subplots()
ax.errorbar(depth_acc[:,0], depth_acc[:,1], yerr=depth_acc[:,2])
plt.show()
```

¿Consultas,

dudas o comentarios?

Muchas gracias por su asistencia y atención

UNIVERSIDAD DE GRANADA

Universidad Centroamericana José Simeón Cañas