Proyecto Avance 1

Gabriel Vélez 00212365
Gabriel Torres 00137882
Joaquín Orbe 00320820

1. *Limpieza de la base de datos*

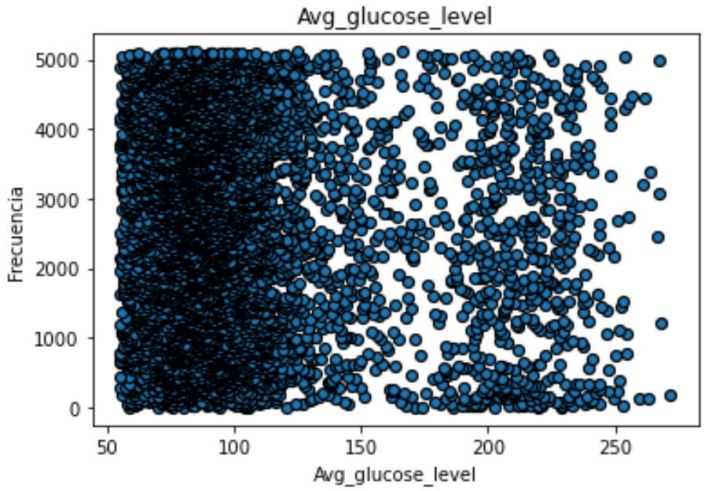
- A primera vista, se puede ver valores nulos, sin sentido y otros que no aportan a la variable
- Es una matriz 5110 filas x 11 columnas. Con tipos de datos enteros, decimales y objetos. Y de los cuales algunos se desconocen la información o está incorrecto
- Para los valores nulos/atípicos, se utilizó diferentes métodos y el K-Nearest Neighbors fue el mejor

Gender	Hypertension	Heart Disease	Married	Work	Residence	Avg_glucose_level	Bmi	Smoking	Age	Stroke
Female	2994	2994	2994	2994	2994	2994	2897	2994	2994	2994
Male	2115	2115	2115	2115	2115	2115	2011	2115	2115	2115
Other	1	1	1	1	1	1	1	1	1	1

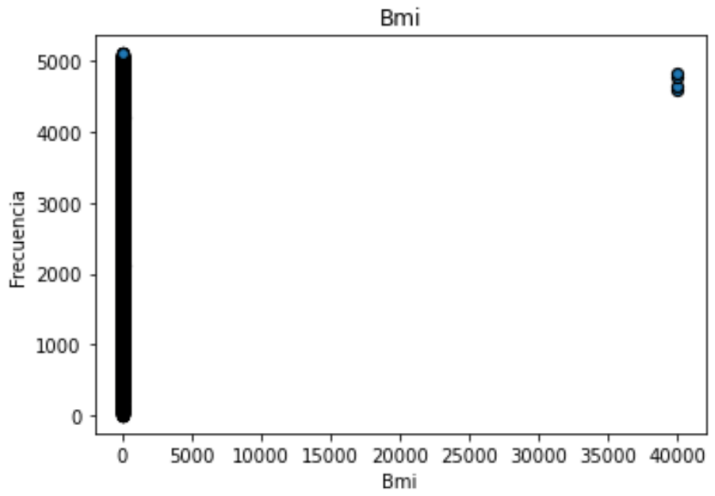
Work	Hypertension	Heart Disease	Married	Residence	Avg_glucose_level	Bmi	Smoking	Gender	Age	Stroke
Govt_job	657	657	657	657	657	630	657	657	657	657
Never_worked	22	22	22	22	22	22	22	22	22	22
Private	2919	2919	2919	2919	2919	2806	2919	2919	2919	2919
Self-employed	818	818	818	818	818	774	818	818	818	818
children	687	687	687	687	687	671	687	687	687	687
dsfsdf	2	2	2	2	2	2	2	2	2	2
k□sdh-k	1	1	1	1	1	0	1	1	1	1
sdfsdf	3	3	3	3	3	3	3	3	3	3
sdsd-i	1	1	1	1	1	1	1	1	1	1

Variables Predictivas

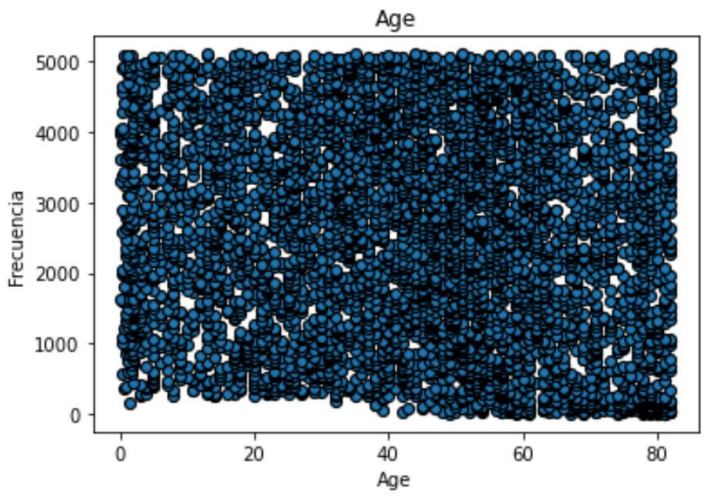
Smoking, Residence, Work, Avg_glucose_level, Married, Hypertension, Heart Disease, Gender



Avg_glucose_level



Bmi



Age

Diferentes modelos para tratamientos de datos

Regresión Lineal con 2 variables numéricas-> R^2: 0,12

Regresión Lineal con todas las variables-> R^2: 0,03

R^2 Lasso: 0,24

R^2 Ridge: 0,25

Regresión con Árbol de Decisión -> R^2: -0,55

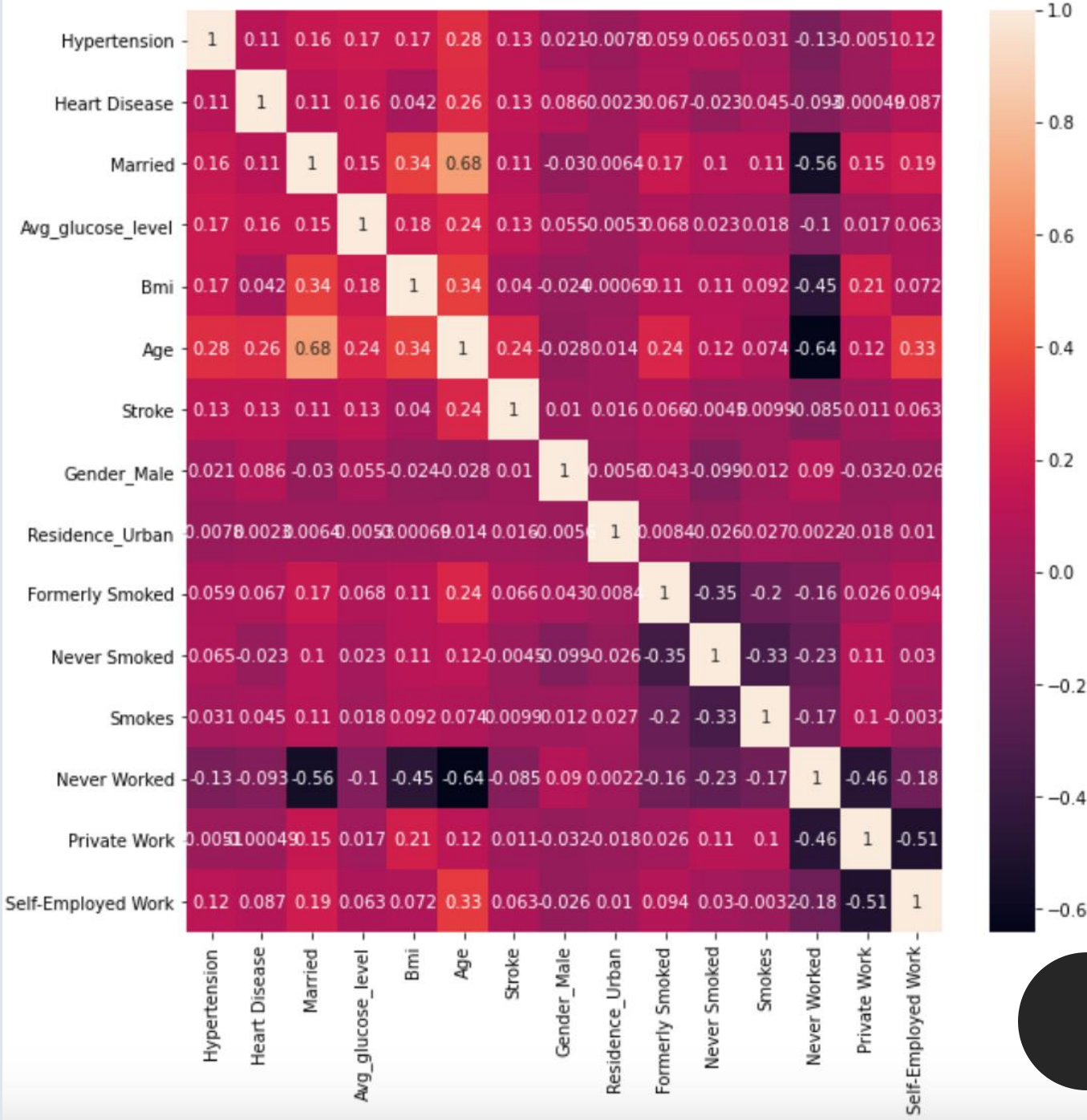
K nearest Neighbors-> R^2: 1

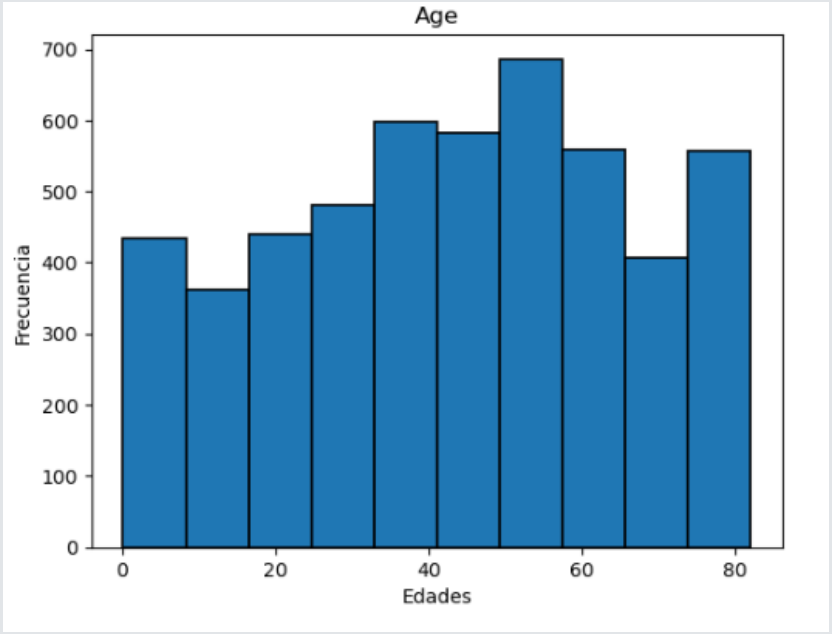
Tabla limpia

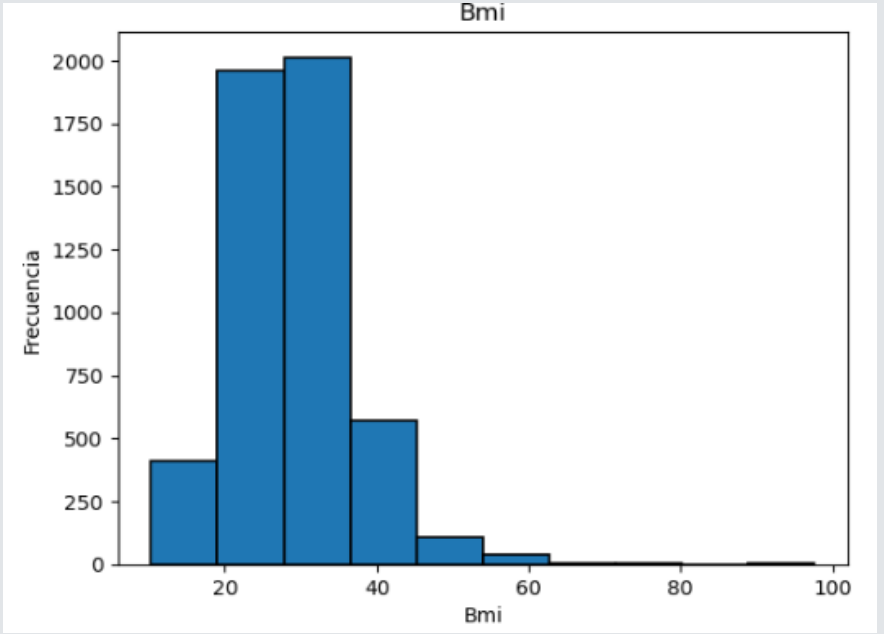
ɔertension	Heart Disease	Married	Avg_glucose_level	Bmi	Age	Stroke	Gender_Male	Residence_Urban	Formerly Smoked	Never Smoked	Smokes	Never Worked	Private Work	Self-Employed Work
0	0	0	55.12	21.8	21.0	0	0	0	0	1	0	0	1	0
0	0	0	55.25	20.4	20.0	0	1	1	0	1	0	0	1	0
0	0	0	55.34	15.3	10.0	0	1	1	0	0	0	1	0	0
0	0	0	55.35	22.7	5.0	0	0	1	0	0	0	1	0	0
0	0	0	55.39	23.2	13.0	0	1	0	0	0	0	1	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
1	1	1	246.53	27.2	59.0	0	1	0	1	0	0	0	1	0
1	1	1	247.51	40.5	68.0	1	0	1	1	0	0	0	1	0
1	1	1	250.89	28.1	81.0	1	1	1	0	0	1	0	1	0
1	1	1	254.63	31.0	67.0	0	1	0	0	1	0	0	1	0
1	1	1	271.74	31.1	68.0	1	1	0	0	0	1	0	1	0

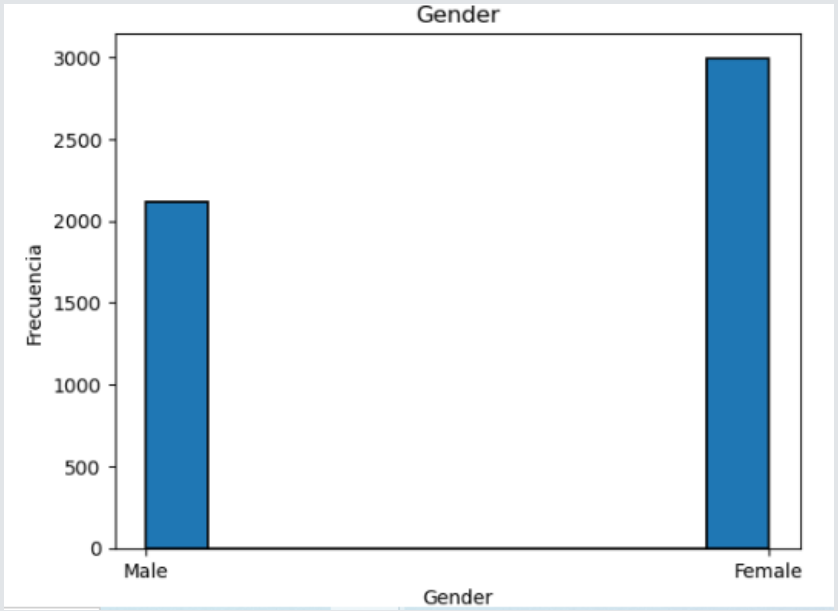
5102 filas x 15 columnas

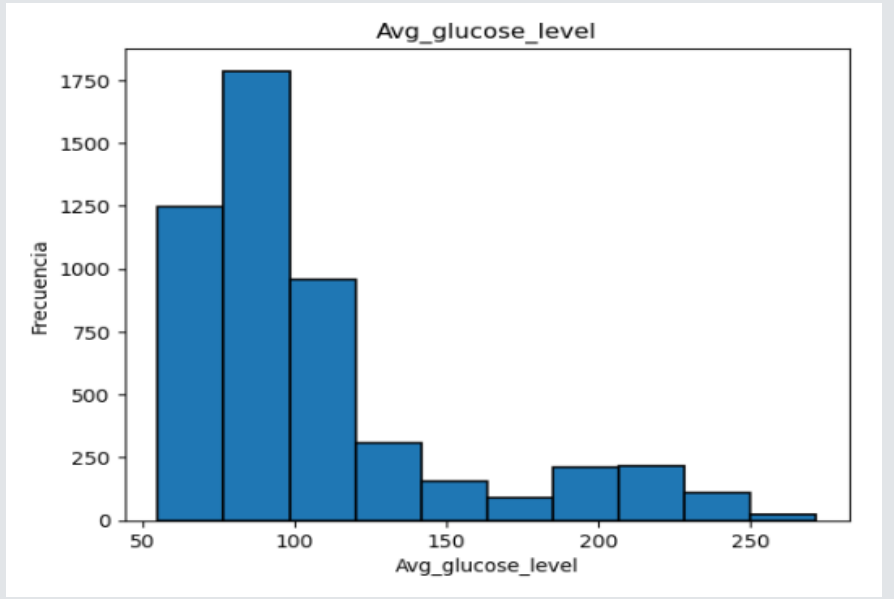
Variables con mayor efecto

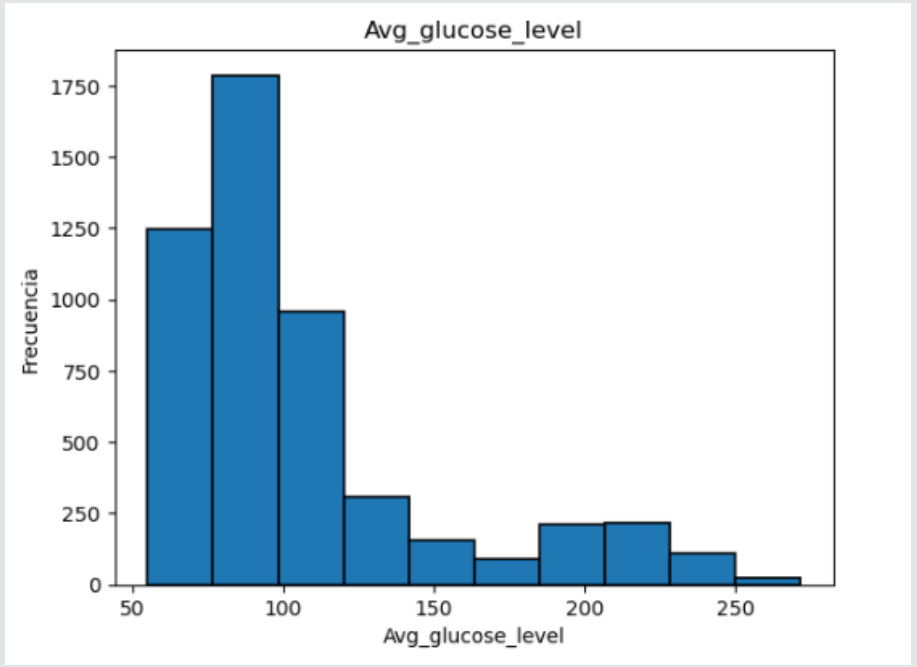


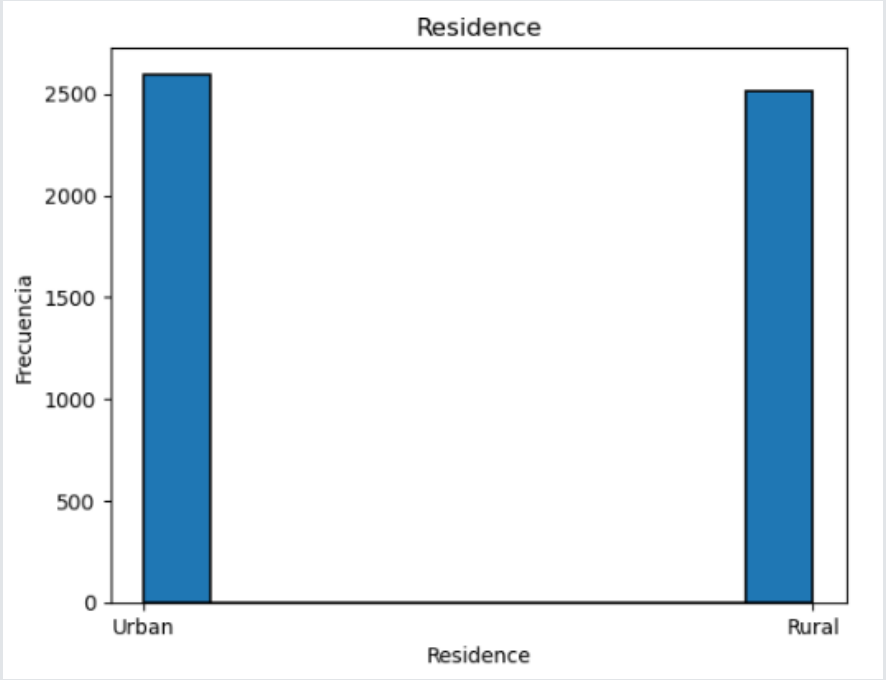

Age
Frecuencia
Edades
0
100
200
300
400
500
600
700
0
20
40
60
80

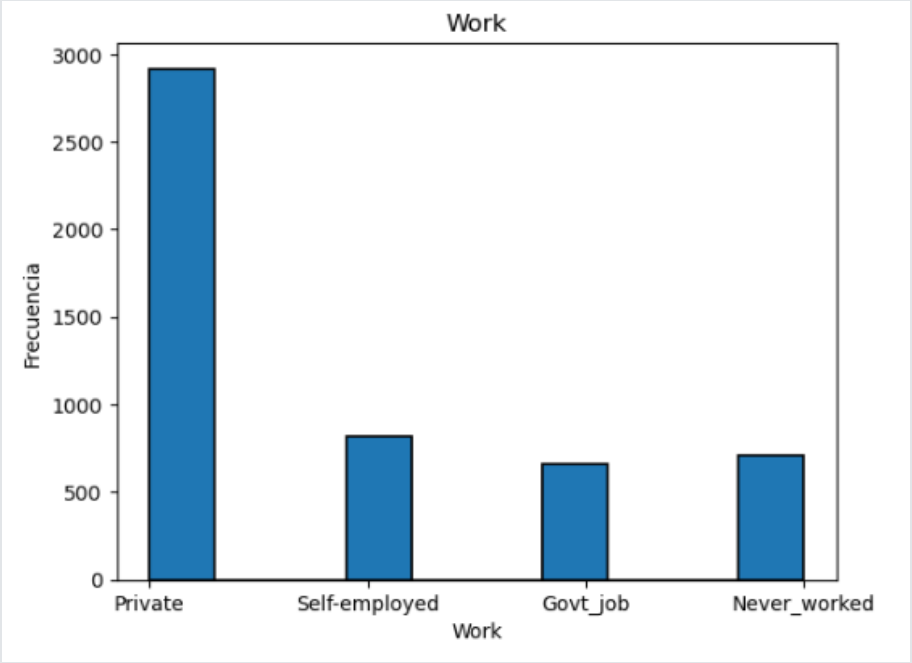

Bmi
Frecuencia
Bmi
0
250
500
750
1000
1250
1500
1750
2000
20
40
60
80
100

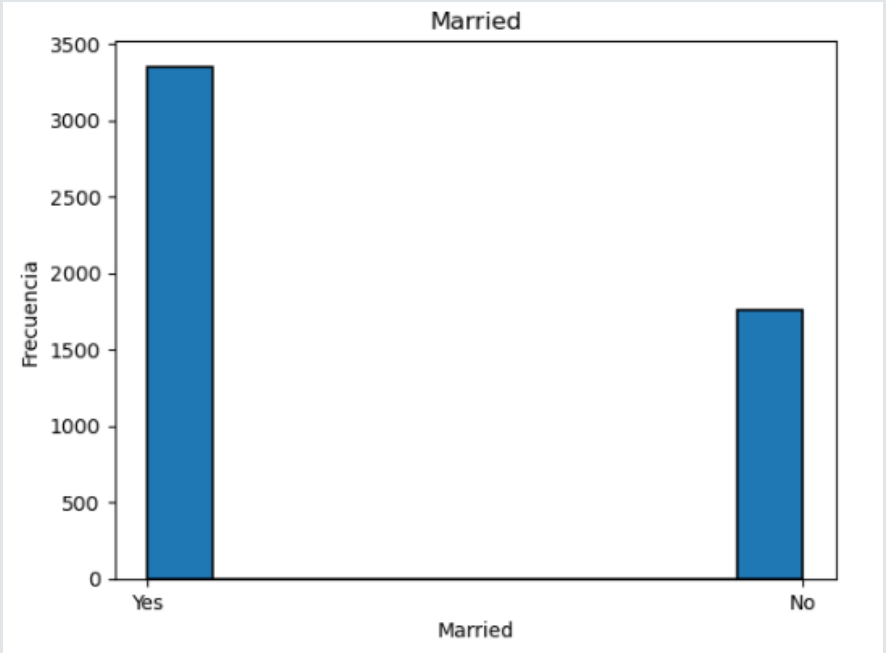

Gender
Frecuencia
Gender
0
500
1000
1500
2000
2500
3000
Male
Female

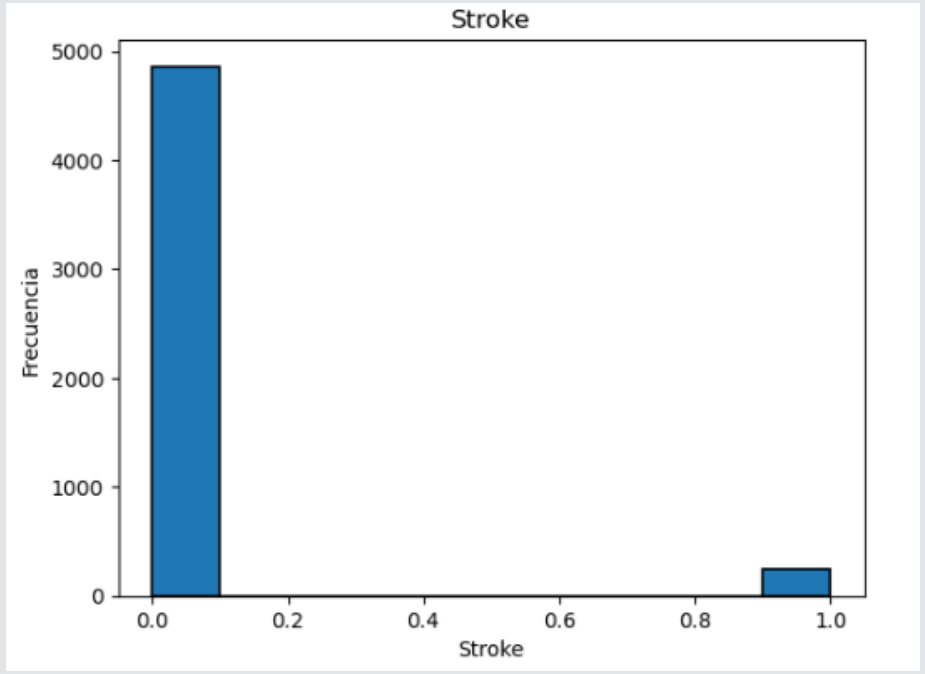

Avg_glucose_level
Frecuencia
Avg_glucose_level
0
250
500
750
1000
1250
1500
1750
50
100
150
200
250

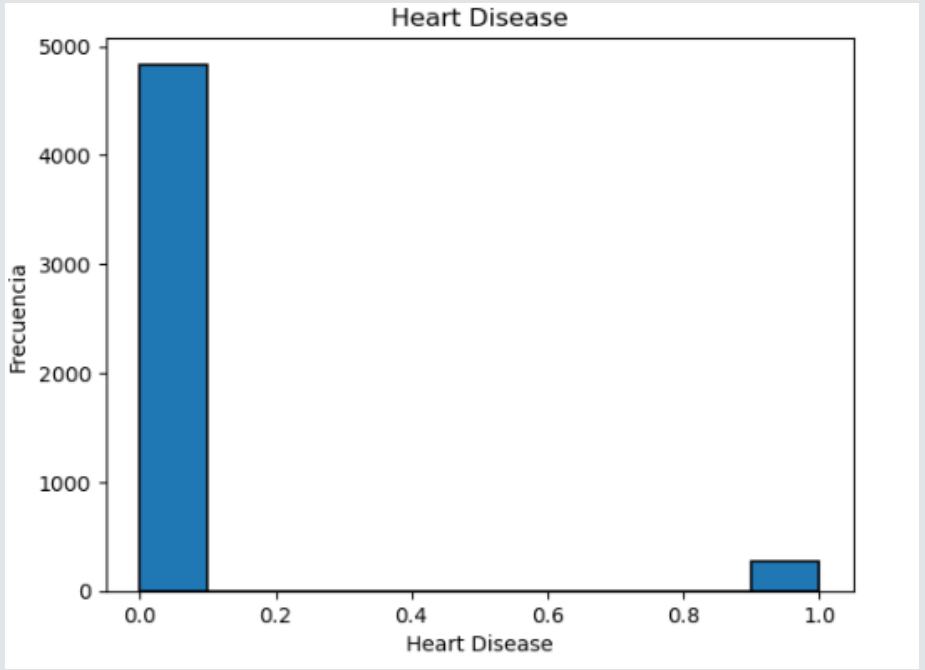

Avg_glucose_level
Frecuencia
0
250
500
750
1000
1250
1500
1750
50
100
150
200
250
Avg_glucose_level

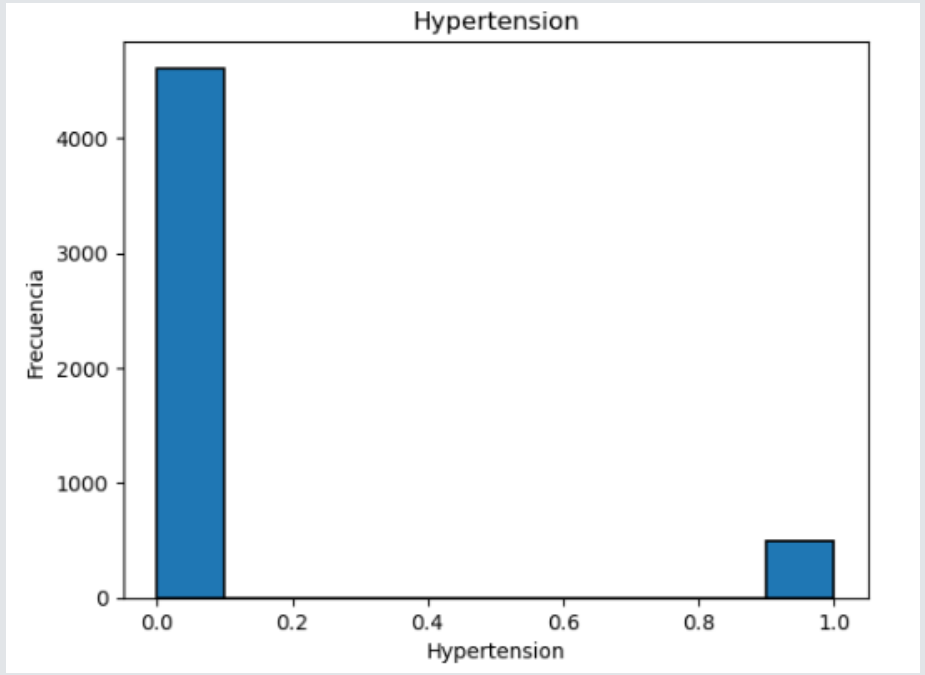

Residence
Frecuencia
0
500
1000
1500
2000
2500
Urban
Rural
Residence

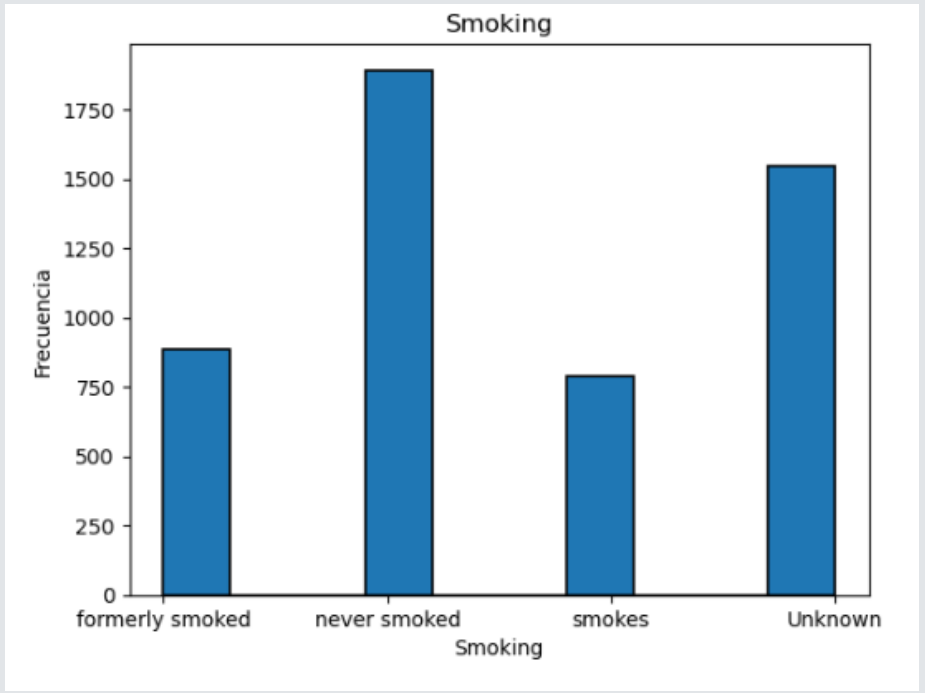

Work
Frecuencia
0
500
1000
1500
2000
2500
3000
Private
Self-employed
Govt_job
Never_worked
Work


Married
Frecuencia
0
500
1000
1500
2000
2500
3000
3500
Yes
No
Married


Stroke
Frecuencia
5000
4000
3000
2000
1000
0
0.0
0.2
0.4
0.6
0.8
1.0
Stroke


Heart Disease
Frecuencia
5000
4000
3000
2000
1000
0
0.0
0.2
0.4
0.6
0.8
1.0
Heart Disease


Hypertension
Frecuencia
4000
3000
2000
1000
0
0.0
0.2
0.4
0.6
0.8
1.0
Hypertension


Smoking
Frecuencia
1750
1500
1250
1000
750
500
250
0
formerly smoked
never smoked
smokes
Unknown
Smoking

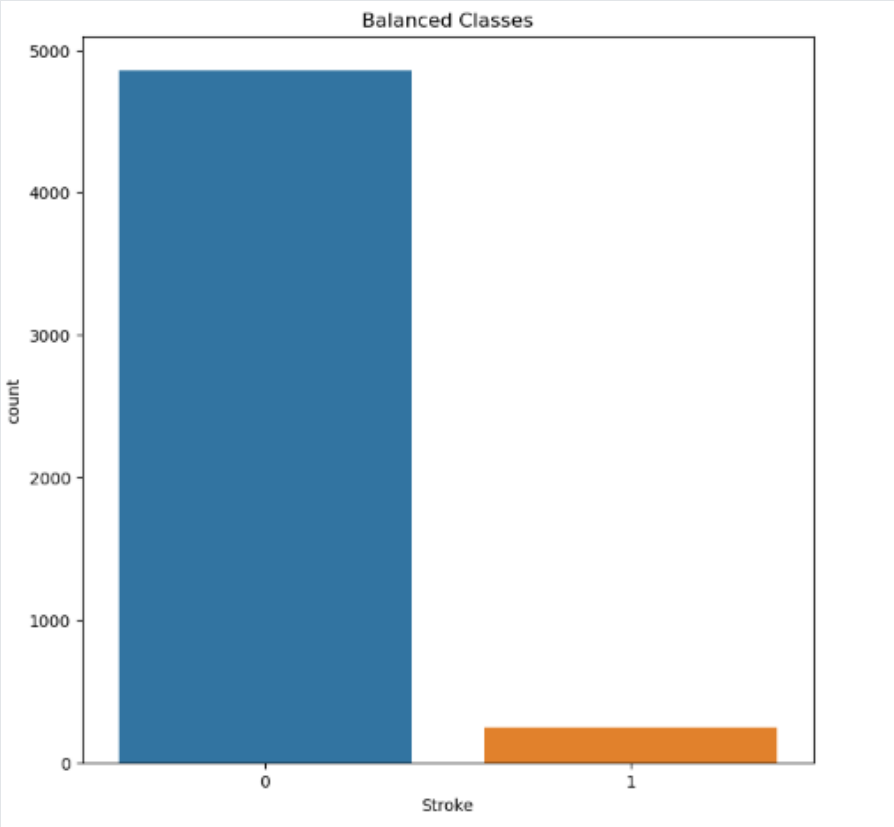
2. Unbalanced Dataset

Una base de datos no balanceada o unbalanced dataset se da cuando, en una base de datos, el número de observaciones de una clase es significativamente mayor que el de otras clases.

Problemas: El algoritmo puede tener un sesgo hacia las clases con mayores observaciones, ignorando a las que tienen un menor número. Además, puede generar overfitting ya que el algoritmo puede sobreajustar los datos de la clase con mayor número de observaciones.

(Badr, 2019)

2. *Unbalanced Dataset*



Soluciones durante el entrenamiento del algoritmo:

Undersampling: lo que quiere decir que se eliminan algunas observaciones de forma aleatoria de las clases mayoritarias para igualar al número de observaciones de las clases minoritarias.

Oversampling: es decir generar datos sintéticos aleatorios para la clase minoritaria. Una de las técnicas usadas para hacer esto es conocida como SMOTE (Synthetic Minority Over-sampling Technique) o K Nearest Neighbors, encontrando así observaciones cercanas para predecir una nueva observación.

(AprendeIA, 2019)

3. Set de entrenamiento, validación y prueba

Set de entrenamiento: 60%

Set de Validación: 20%

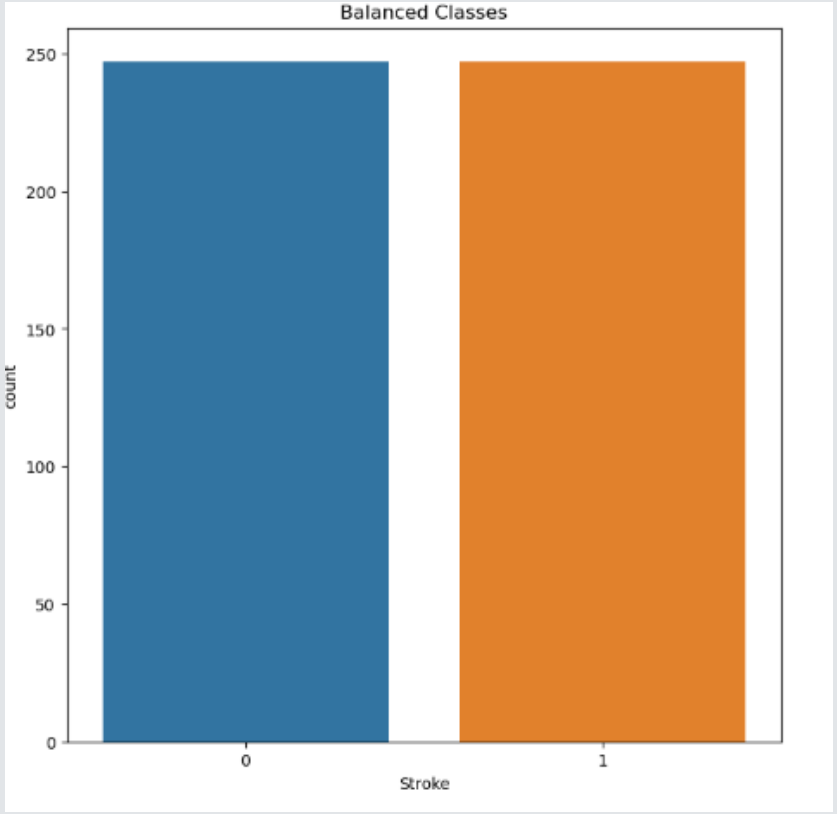
Set de Prueba: 20%

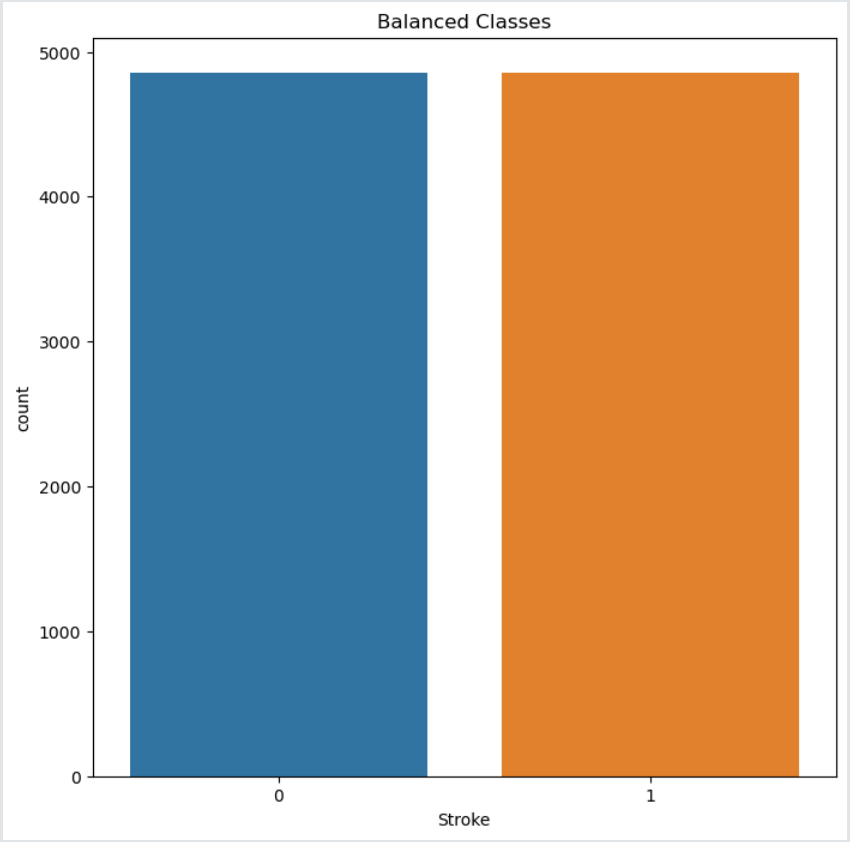
Training (60%) : (2040, 15)
Validation (20%) : (2449, 15)
Test (20%) : (613, 15)

(Cho, 2016)

4. Algoritmos y aplicando técnicas

- Undersampling






```
Mean cross-validation score for OverSampled: 0.62
Mean cross-validation score for UnderSampled: 0.88
ENTONCES SE UTILIZA UNDERSAMPLED
```

- Oversampling

5. Algoritmos y mejor técnica de entrenamiento

- Se utilizaron dos algoritmos:
- Random decision forests: Es un ensamble de varios árboles de decision concatenados
- Support Vector Machines: Este algoritmo no fue visto en clase. Se crea hiperplanos entre todas las clases y se busca maximizar la distancia entre los puntos más cercanos del hiperplano. Estos puntos se llaman vectores de soporte y el algoritmo escoge los vectores que mejor describen al modelo.

Random decision forest

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

```
#Se utiliza el modelo random forest classifier
model = RandomForestClassifier()
```

```
#Se entrena el modelo con nustros sets de entrenamiento
model.fit(X_train, y_train)
```

```
RandomForestClassifier()
```

```
y_pred = model.predict(X_test)
```

```
#Se muestran los puntajes de presición, accuracy, Recall y F1 para el modelo
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score

print("Accuracy:", accuracy_score(y_test, y_pred))
print("Precision:", precision_score(y_test, y_pred))
print("Recall:", recall_score(y_test, y_pred))
print("F1 Score:", f1_score(y_test, y_pred))
```

Accuracy: 0.991

Precision: 0.983

Recall: 1.0

F1 Score: 0.991

Support Vector Machines

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

```
svm = SVC(kernel='linear', C=1, random_state=42)

# Entrenar el modelo SVM
svm.fit(X_train, y_train)

# Se realizan las predicciones
y_pred = svm.predict(X_test)

#Se muestran los puntajes de presición, accuracy, Recall y F1 para el modelo
accuracy = accuracy_score(y_test, y_pred)
precision = precision_score(y_test, y_pred)
recall = recall_score(y_test, y_pred)
f1 = f1_score(y_test, y_pred)
```

```
print("Accuracy: {:.3f}".format(accuracy))
print("Precision: {:.3f}".format(precision))
print("Recall: {:.3f}".format(recall))
print("F1 Score: {:.3f}".format(f1))
```

Accuracy: 0.773

Precision: 0.752

Recall: 0.834

F1 Score: 0.791

Conclusiones

- En conclusión, por más que no se termine el proyecto, ya se tiene una mejor idea con la base de datos correctamente estructurada y analizada
- La limpieza de datos es una parte fundamental en la generación de un modelo adecuado.
- El método K nearest Neighbors es el que realiza una mejor predicción en comparación a la regresión lineal, logística y árbol de decisión para reemplazar valores atípicos.
- Aprendimos técnicas para balancear la base de datos, utilizando undersampling como el métodomas adecuado.
- Dentro de los dos modelos utilizados, el Random decision forest fue el que nos dio el mejor modelo

Referencias

- Badr, W. (2019). Having an Imbalanced Dataset? Here Is How You Can Fix It. Towards Data Science. Having an Imbalanced Dataset? Here Is How You Can Fix It. | by Will Badr | Towards Data Science
- AprendeIA. (2019). Conjunto de datos desbalanceado. AprendeIA. Conjunto de datos desbalanceado - 🤖 Aprende IA
- Gutierrez-Garcia, J. O. (2021). Datos de Entrenamiento, Validación y Prueba: ¿Cómo crearlos y qué objetivos tienen? Machine Learning. YouTube. https://www.youtube.com/watch?v=vdYzm4xC7mc
- Cho,J. Lee,K. Et.al.(2016)How much data is needed to train a medical image deep learning system, to achieve necessary high accuracy