Modelo predictivo para cuantificar el riesgo de mortalidad causada por insuficiencia cardiaca.

Rivera Vergaray, Kevin
Analista TI de Servicios Académicos
UNIA
kevinriveravergaray@gmail.com

Coronel Berrospi, Gustavo
Estudiante de ingeniería de TI
ESAN
gustavocoronel123@gmail.com

Pastor Carreño, Alfonso
Estudiante de ingeniería Mecatrónica
PUCP
alfonso.pastor@pucp.edu.pe

***Abstract*—En el presente artículo se explora la data de DATA Heart Failure Prediction que se encuentra en kaggle, el cual cuenta con 12 características clínicas para predecir eventos de muerte por insuficiencia cardiaca, es así que en este artículo experimentamos con diferentes tipos de modelos predictivos y hacemos una comparación entre la efectividad de los mismos, para esta oportunidad se utilizó el modelo de regresión logística por máxima verosimilitud, árboles de decisión y random forest.**

***Keywords*—Python, Modelo predictivo.**

I. INTRODUCTION (*HEADING 1*)

Las enfermedades cardiovasculares (ECV) son la principal causa de muerte a nivel mundial , cobrando aproximadamente 17,9 millones de vidas cada año , lo que representa el 31% de todas las muertes en todo el mundo .

La insuficiencia cardíaca es un evento común causado por las enfermedades cardiovasculares y este conjunto de datos contiene 12 características que se pueden utilizar para predecir la mortalidad por insuficiencia cardíaca.

La mayoría de las enfermedades cardiovasculares se pueden prevenir abordando los factores de riesgo conductuales, como el consumo de tabaco, la dieta poco saludable y la obesidad, la inactividad física y el consumo nocivo de alcohol mediante estrategias de población.

Las personas con enfermedad cardiovascular o que tienen un alto riesgo cardiovascular (debido a la presencia de uno o más factores de riesgo como hipertensión, diabetes, hiperlipidemia o enfermedad ya establecida) necesitan una herramienta que se anticipe a la enfermedad, en lo que un modelo de aprendizaje automático puede ser de gran ayuda.

Es así que se pretende aplicar conceptos de machine learning y análisis de datos de tal forma que podamos crear un modelo predictivo que nos permita cuantificar el riesgo de mortalidad causada por la insuficiencia cardiaca.

II. TRABAJOS RELACIONADOS

MODELO PREDICTIVO PARA LA SUPERVIVENCIA Y LA MORTALIDAD PERIOPERATORIA EN PACIENTES CON CARCINOMA RENAL Y EXTENSIÓN VENOSA TUMORAL (D. Juan Ignacio Martínez Salamanca - Universidad Autónoma de Madrid).

El tumor renal con extensión venosa es una entidad clínica poco frecuente, que afecta del 4-10% del total de pacientes con carcinoma renal. La presencia de trombo tumoral posee importantes consideraciones pronósticas y condiciona la elección de la técnica quirúrgica, además de constituir, en sí misma un factor pronóstico. Los modelos predictivos disponibles en carcinoma renal se refieren a la estimación de la probabilidad de supervivencia tras la cirugía y no contemplan la extensión venosa, como variable de análisis. En los pacientes con carcinoma renal y compromiso trombótico, se reconoce una elevación del riesgo quirúrgico, pero no es posible conocer su riesgo de mortalidad perioperatoria (primeros 30 días tras la cirugía) en función de variables conocidas y disponibles antes de la cirugía (edad, sexo, tamaño tumoral y nivel de trombo). Asimismo, la eventual predicción de supervivencia posterior a la cirugía resultaría de utilidad para seleccionar pacientes en función del riesgo e indicar tratamientos adyuvantes.

Predicción de muerte súbita en pacientes con insuficiencia cardíaca crónica mediante el estudio de la dinámica periódica de la repolarización(S. Palacios , I. Cygankiewicz , A. Bayés-de-Luna , J.P. Martínez, E. Pueyo– XXXVIII Congreso Anual de la Sociedad Española de Ingeniería Biomédica)

La insuficiencia cardíaca crónica (ICC) es un síndrome clínico con una alta mortalidad debida a la muerte por fallo de bomba (MFB), y al desarrollo de arritmias ventriculares que pueden provocar muerte súbita cardíaca (MSC). Los pacientes con ICC experimentan un desequilibrio en el sistema nervioso autónomo que podría reflejarse en el electrocardiograma (ECG). La Dinámica Periódica de la Repolarización ("Periodic Repolarization Dynamics", PRD) cuantifica las oscilaciones de baja frecuencia en la onda T del ECG y se ha relacionado con la modulación simpática de la repolarización ventricular. Este estudio evalúa la capacidad predictiva de PRD para MFB y MSC en una población con ICC. Se analizaron ECGs de 20 minutos y 3 derivaciones de 569 pacientes con ICC sintomática y ritmo sinusal normal. PRD se midió en segmentos de 5 minutos, con solapamientos de 4 minutos, asignándose a cada registro el mínimo valor entre ellos. Se encontró que el PRD presentaba valores más elevados en víctimas de MSC en comparación al resto de pacientes, aunque las diferencias no fueron estadísticamente significativas. Tomando como umbral la mediana de los valores de PRD en la población global, se definieron grupos de bajo y alto riesgo. Se obtuvo un cociente de riesgo de MSC [Intervalo de confianza al 95%] en el análisis univariado de Cox de 1.808 [1.031-3.169] grados (p=0.039). En conclusión, PRD predice el riesgo de MSC en pacientes con ICC, presentando las víctimas de MSC una mayor magnitud de las oscilaciones de la repolarización ventricular inducidas por la activación simpática.

Aplicación del puntaje de riesgo en Síncope de Boston para la predicción de mortalidad y desenlaces cardiovasculares en pacientes adultos(Mauricio Andrés Quintero Betancur - Universidad Nacional de Colombia)

El síncope lleva a muchas hospitalizaciones innecesarias, al igual que a un costo excesivo e innecesario . En el mundo se han desarrollado modelos predictivos para que las enfermedades como estas ya no generen dichas

características, teniendo como objetivo aplicar en población local la regla de predicción de riesgo en síncope de Boston y determinar sus características operativas en cuanto a su sensibilidad, especificidad, valor predictivo positivo (VPP), valor predictivo negativo (VPN), para predecir desenlaces adversos, en el corto plazo (30 días). Metodología: estudio observacional de seguimiento de una cohorte de pacientes que consultan a urgencias y a la unidad de cuidado intensivo de un hospital de 3er nivel de la ciudad de Bogotá (Colombia) con diagnóstico de síncope, aplicación de un instrumento de recolección de datos que contiene los puntos evaluados por la regla original y evaluar los desenlaces adversos a 30 días mediante llamada telefónica, visita domiciliaria o revisión de historia clínica. Resultados: se reclutaron 98 pacientes entre octubre de 2013 y junio de 2014, la prueba fue positiva en 91 veces y se encontraron 43 desenlaces primarios, lo que aporta Sensibilidad: 100%, Especificidad: 12,73%, Valor predictivo positivo: 47,25% y Valor predictivo negativo: 100% Conclusión: la regla de predicción de resultados adversos en síncope de Boston tiene unas características operativas similares a las reportadas por el estudio original.

III. METODOLOGÍA

La tabla de datos contiene información de la salud de 299 personas, esta información está compuesta por 12 columnas que contienen datos importantes para un correcto análisis de los problemas cardiovasculares .

El objetivo planteado para este trabajo es cuantificar la mortalidad causada por la insuficiencia cardiaca utilizando un modelo predictivo, ya que se conoce que es de mucha importancia poder predecir quien se encuentra en mayor riesgo de descenso. Esto nos lleva a plantearnos nuestra hipótesis, si el modelo predictivo cuantifica el riesgo de mortalidad causada por la insuficiencia cardiaca.

IV. EXPERIMENTO

Se procedió a hacer la lectura de los datos en la plataforma de Google Colab. Luego, con las librerías Pandas y Numpy , se inició una exploración de los datos para determinar la evolución del evento de muerte por tiempo de monitoreo, e ir explorando las principales características que tienen los datos para hacer los ajustes necesarios antes de poder correr los modelos.

TABLA 1
VARIABLES CONSIDERADAS EN EL DATASET

Nº	VARIABLE	DESCRIPCIÓN DE LA VARIABLE
1	age	Edad del paciente.
2	anemia	Si el paciente sufre de anemia.
3	creatinine_phosphokinase	nivel de la encima CPK en la sangre (mcg/L). Cuando el nivel total de CPK es muy alto, a menudo significa que ha habido lesión o estrés en el corazón, el cerebro o el tejido muscular. 10 to 120 micrograms per liter (mcg/L)
4	diabetes	Si el paciente tiene diabetes.
5	ejection_fraction	porcentaje de sangre que sale del corazon en cada contracción.Una fracción de eyección de 55 por ciento o más se considera normal. Una fracción de eyección de 50 por ciento o menos se considera reducida.Una fracción de eyección de entre 50 y 55 por ciento generalmente se considera «limítrofe».
6	high_blood_pressure	Si el paciente tiene hipertensión.
7	platelets	Plaquetas en la sangre (kPlatelets/mL). Valores normales Hombre: 135-317 billones/L (De 135,000 a 317,000/mcL) ,Mujer: 157-371 billones/L (157,000-371,000/mcL)
8	serum_creatinine	nivel de creatinina en la sangre (mg/dL).Niveles normales suelen ser 0.7 a 1.3 mg/dL en hombres 0.5 a 1.2 mg/dL en mujeres
9	serum_sodium	nivel de sodio en la sangre (mEq/L). Un nivel normal de sodio en la sangre oscila entre 135 y 145 miliequivalentes por litro (mEq/L). La hiponatremia se produce cuando el sodio en el cuerpo se encuentra por debajo de 135 mEq/L.
10	sex	sexo.
11	smoking	si el pacientes es fumador.
12	time	tiempo de seguimiento (days).
13	DEATH_EVENT	si el paciente muere en el tiempo de seguimiento.

Además, se realizó la gráfica de los datos con el uso de librerías como Matplotlib.

Se identificaron tres pasos para la exploración de los datos: La carga de datos, exploración de los datos y realización de gráficos estadísticos.

A. Carga de datos: Se obtuvieron los datos de la página de Kaggel, los cuales estaban en formato .csv el cual fue subido al Drive y posteriormente vinculado con el Google Colab.

B. Exploración de datos

Una vez cargado los datos, se utilizó la librería Pandas para obtener información relevante de estos. Es así que al analizar la existencia de data nula se encontró que no había valores nulos.

Luego de analizar los datos se definió las variables categóricas, las variables numéricas y el target quedando los datos de la siguiente manera:

- Variables Categóricas: ['anaemia', 'diabetes', 'high_blood_pressure', 'sex', 'smoking']
- Variables Numéricas: ['age', 'creatinine_phosphokinase', 'ejection_fraction', 'platelets', 'serum_creatinine', 'serum_sodium', 'time']
- target: 'DEATH_EVENT'

Luego de definir nuestras variables se procedió a realizar el análisis univariante y bivariante tanto para las variables numéricas como categóricas.

Análisis Univariante: El objetivo de este análisis es caracterizar a las variables por sus estadísticas resumen principales (media, mediana, moda, etc).

✔ **Variables categóricas**

Para realizar el análisis estadístico de las variables categóricas se realizó el cálculo la frecuencia absoluta y frecuencia relativa de cada una de estas lo cual se detalla a continuación:

1. **anaemia:** Se puede identificar 2 valores 0 si no tiene anemia y 1 si el paciente tiene anemia. A continuación, se muestra la gráfica de distribuciones de esta variable:

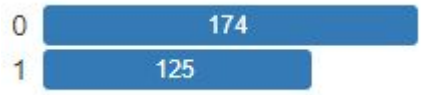
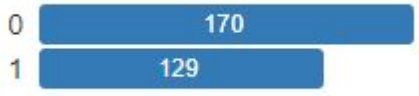


Fig. 1 Análisis de variable “anaemia”

2. **diabetes:** Se puede identificar 2 valores 0 si no tiene diabetes y 1 si el paciente tiene diabetes. A continuación, se muestra la gráfica de distribuciones de esta variable



:

Fig. 2 Análisis de variable “diabetes”

3. **high_blood_pressure:** Se puede identificar 2 valores 0 si no tiene presión alta y 1 si el paciente tiene presión alta. A continuación, se muestra la gráfica de distribuciones de esta variable:

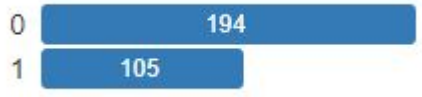


Fig. 3 Análisis de variable “high_blood_pressure”

4. **sex:** Se puede identificar 2 valores 0 si es mujer y 1 si el paciente es varón. A continuación, se muestra la gráfica de distribuciones de esta variable:

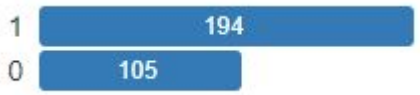


Fig. 4 Análisis de variable “sex”

5. **smoking:** Se puede identificar 2 valores 0 si el paciente no fuma y 1 si el paciente si fuma. A

continuación, se muestra la gráfica de distribuciones de esta variable:

Fig. 5 Análisis de variable “smoking”

✔ **Variables numéricas**

1. **age:** Se obtuvo un mínimo de 40 años , un máximo de 95 años y el promedio de las edades es de 60.83.

2. **creatinine_phosphokinase:** Se obtuvo un mínimo de 23 , un máximo de 7861 y el promedio del nivel de enzima CPK es de 581.83.

3. **ejection_fraction:** Se obtuvo un mínimo de 14, un máximo de 80 y el promedio del porcentaje de sangre de 38.08.

4. **platelets:**Se obtuvo un mínimo de 25100, un máximo de 850000 y el promedio de plaquetas de sangres es 263358.02.

5. **serum_creatinine:** Se obtuvo un mínimo de 0.5, un máximo de 9.4 el promedio de nivel de creatina de 1.3938.

6. **serum_sodium:** Se obtuvo un mínimo de 113, un máximo de 148 el promedio de nivel de sodio de 136.62.

7. **time:** Se obtuvo un mínimo de 4, un máximo de 73 y el promedio de tiempo de monitoreo es de 130.26.

Análisis de percentiles:

TABLA II
ANÁLISIS DE PERCENTILES

	age	creatinine_phosphokinase	ejection_fraction	platelets	serum_creatinine	serum_sodium	time
count	299	299	299	299	299	299	299
mean	60.834	581.839465	38.08361	263358	1.39388	136.625418	130.2609
std	11.895	970.287881	11.83484	97804.24	1.03451	4.412477	77.61421
min	40	23	14	25100	0.5	113	4
25%	51	116.5	30	212500	0.9	134	73
50%	60	250	38	262000	1.1	137	115
75%	70	582	45	303500	1.4	140	203
max	95	7861	80	850000	9.4	148	285

Fig. 5 Gráfica de distribución de datos

Gráfico de Distribución de datos: A continuación, se muestran las gráficas de distribución de datos entre las variables numéricas.

1. **age:**Se puede observar que la mediana se encuentra alrededor de 60 y en resumen los datos en esta variable están bien distribuidos.

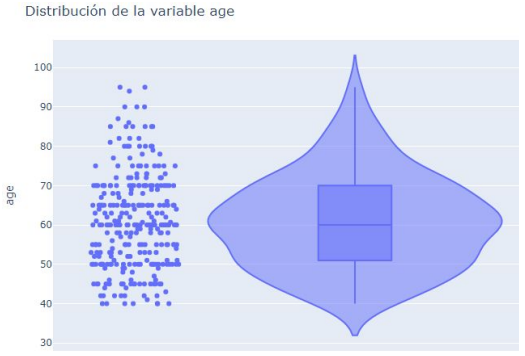


Fig. 6 Análisis de variable “age”

2. **creatinine_phosphokinase:** Se puede observar que la mediana se encuentra alrededor 250 y además podemos ver que en esta variable existen **outliers**.

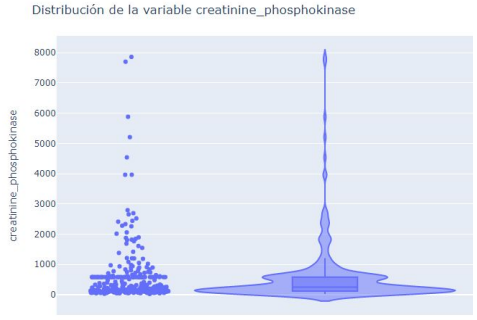


Fig. 7 Análisis de variable “creatinine_phosphokinase”

3. **ejection_fraction:** Se puede observar que la mediana se encuentra alrededor 38 y además podemos ver que en esta variable existen **outliers**.

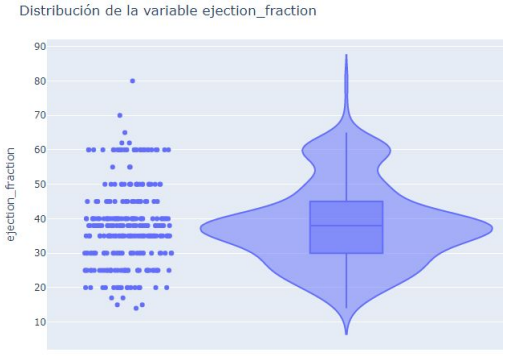


Fig. 8 Análisis de variable “ejection_fraction”

4. **platelets:** Se puede observar que la mediana se encuentra entre 262000, además podemos ver que en esta variable existen **outliers**.

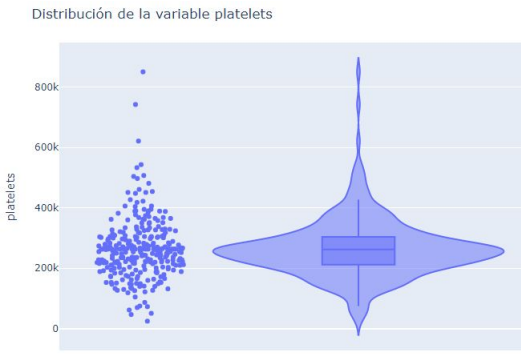


Fig. 9 Análisis de variable “platelets”

5. **serum_creatinine:** Se puede observar que la mediana se encuentra entre 1.1, además podemos ver que en esta variable existen un pequeño grupo de **outliers**.

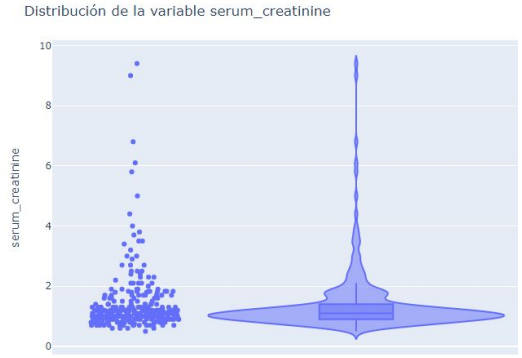


Fig. 10 Análisis de variable “serum_creatinine”

6. **serum_sodium:**Se puede observar que la mediana se encuentra entre 137, además podemos ver que en esta variable existen un pequeño grupo de **outliers**..

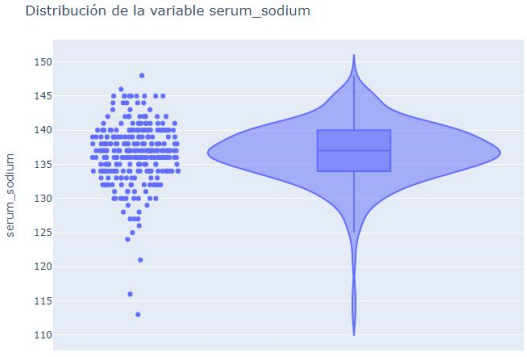


Fig.11 Análisis de variable “serum_sodium”

7. **time:**Se puede observar que la mediana se encuentra alrededor de 115 y en resumen los datos en esta variable están bien distribuidos..

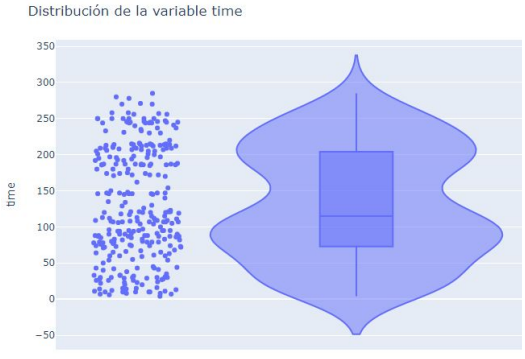


Fig. 12 Análisis de variable “time”

✔ **Target:** Vemos que en nuestra data el **67.9%** son pacientes que no fallecieron de fallo cardiaco y tenemos un **32,1%** que si falleció durante el tiempo que se monitoreo, la cantidad de personas que no fallecen es mayor , sin embargo la cantidad de personas que fallecen si es significativa .

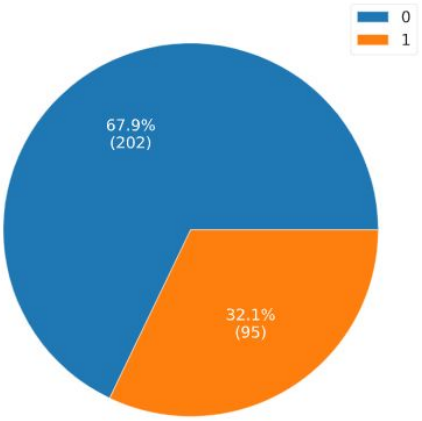


Fig. 13 Análisis de variable Target “”

Análisis bivariante

✔ **Variables categóricas**

1. **anaemia vs DEATH_EVENT:** En esta comparación se puede ver que el porcentaje de las personas con anemia que mueren es mayor que las personas que mueren y no tienen anemia, esto a pesar de que el porcentaje de personas que tienen anemia es mayor que el que no tienen.

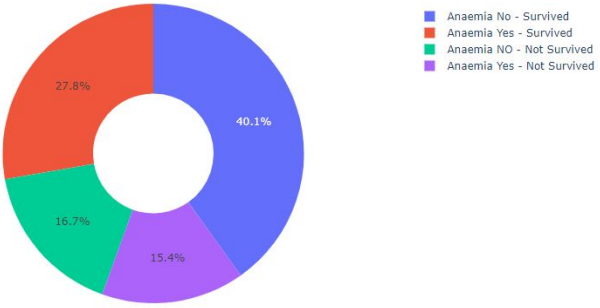


Fig. 14 Análisis de “anaemia vs DEAT H_EVENT ”

2. **high_blood_pressure vs DEATH_EVENT:** En este caso se observa algo no esperado , el porcentaje de personas que fallecen y no sufren de presión alta es mayor que la gente que muere y tiene presión alta. Además, cabe resaltar que es más del doble la cantidad de personas que no sufren de hipertensión.

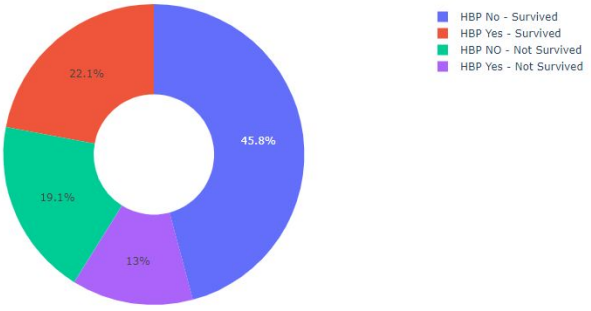


Fig. 16 Análisis de “high_blood_pressure vs DEATH_EVENT”

3. **sex vs DEATH_EVENT:** Se observa que prácticamente la tercera parte del total de hombres y

la tercera parte del total de mujeres son los que fallecen. Por lo cual parece que este indicador no es relevante.

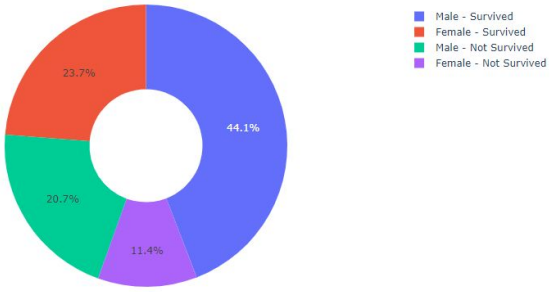


Fig. 17 Análisis de "sex vs DEATH_EVENT"

4. **smoking vs DEATH_EVENT:** En esta variable se observa un patrón igual a la de la variable anterior , por lo cual parece que el hecho de que fumen no es una variable influyente.

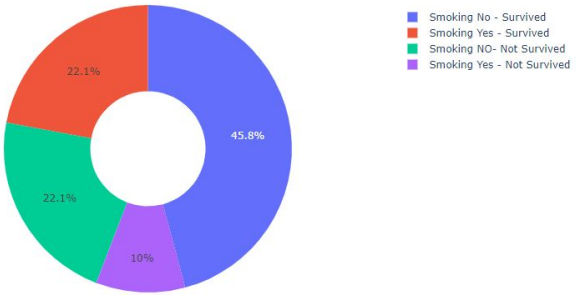


Fig. 18 Análisis de "smoking vs DEATH_EVENT""

5. **diabetes vs DEATH_EVENT:** Se puede visualizar que son los varones los estudiantes que tienen mayor % de deserción.

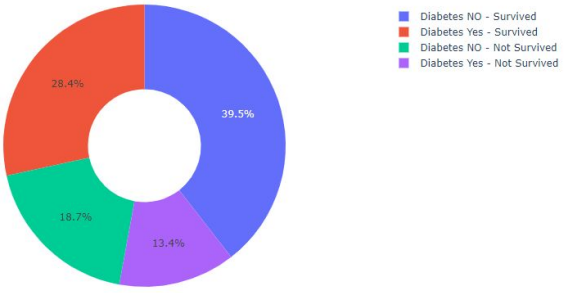


Fig. 19 Análisis de "diabetes vs DEATH_EVENT"

✔ **Variables numéricas**

1. **age vs DEATH_EVENT:** Se puede observar que el mayor porcentaje de deserción tienen hasta 22 años como máximo.

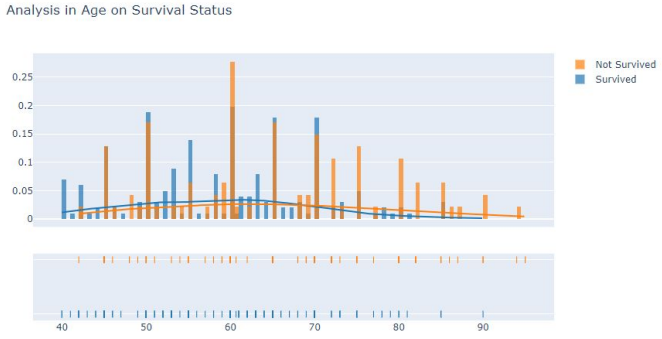


Fig. 20 Análisis de "age vs DEATH_EVENT"

2. **serum_sodium vs DEATH_EVENT:** Se puede observar que para valores bajos la cantidad de personas que fallecen es mayor que la que sobreviven.

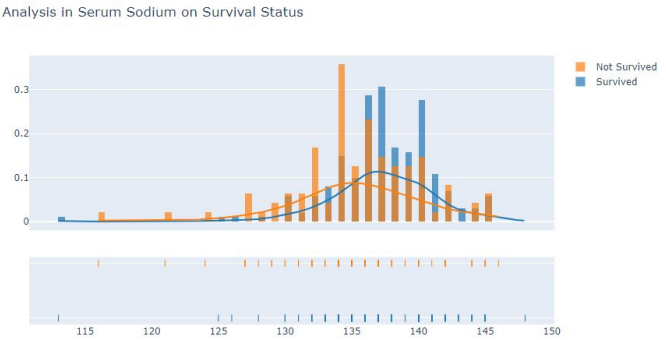


Fig. 21 Análisis de "serum_sodium vs DEATH_EVENT"

3. **serum_creatinine vs DEATH_EVENT:** Se observa que para valores altos la cantidad de personas que fallecen es mayor que la que sobrevive.

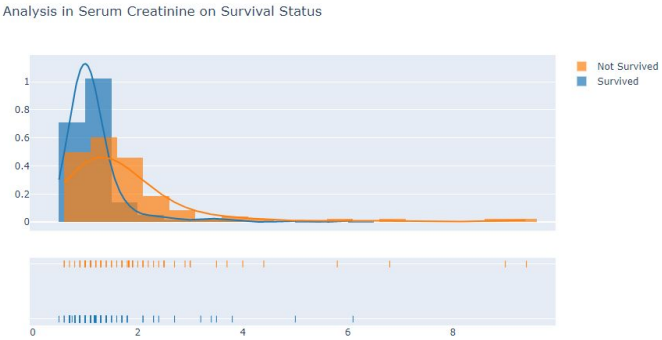


Fig. 22 Análisis de "serum_creatinine vs DEATH_EVENT"

4. **ejection_fraction vs DEATH_EVENT:** Se observa una tendencia que a valores bajos hay mayor cantidad de personas fallecidas.

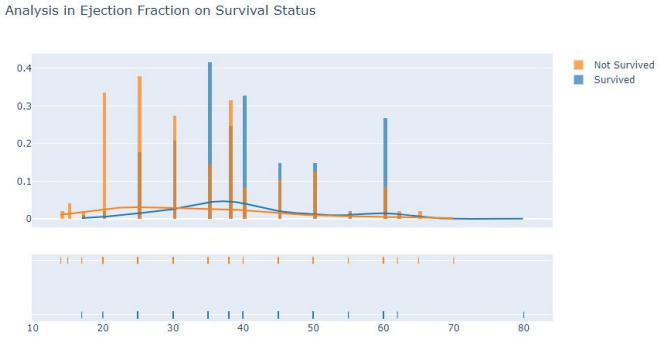


Fig. 23 Análisis de "ejection_fraction vs DEATH_EVENT"

Análisis de correlaciones:

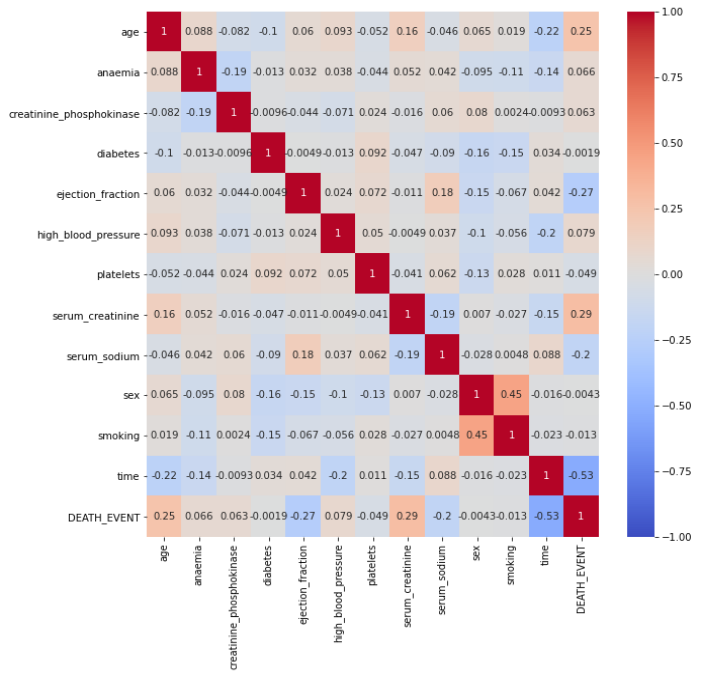


Fig. 24 Análisis de correlación

Se pudo observar que existen correlaciones positivas y negativas entre las variables numéricas analizadas.

Modelado:

1. **Para correr el modelo**

- Elegimos las variables predictoras y el target:
  Variables predictoras: ['anaemia', 'diabetes', 'high_blood_pressure', 'sex', 'smoking','age', 'creatinine_phosphokinase', 'ejection_fraction', 'platelets', 'serum_creatinine', 'serum_sodium', 'time', 'total']
  Definiendo **X** como la variable predictora/independiente y **Y** como la variable respuesta/dependiente.
  Target: 'Desercion'
- Utilizamos la librería **sklearn** para definir el train y el test para **X** y **Y**.
- Verificamos los tamaños de las pruebas de train y test
  - Tamaño del conjunto de datos Inicial: (219, 19)
  - Tamaño del conjunto de características del entrenamiento: (209, 13)
  - Tamaño del conjunto de características de prueba: (90, 13)
  - Tamaño de la variable objetivo del entrenamiento: (209,)
  - Tamaño de la variable objetivo de prueba: (90,)

2. **Utilizando el método de Regresión Logística**

a. **Regresión logística por gradiente del descenso**

Para aplicar este modelo se importó en Python a través de la librería **skelearn,** el modelo de regresión logística, las métricas de los modelos, creamos el modelo de regresión logística y entrenamos el modelo con el **X_train** y **Y_train,** obteniendo los siguientes resultados:

El término de intersección del modelo lineal:
[0.0002405]
Los coeficientes del modelo lineal:
[[6.96530471e-05 3.72623557e-04 -1.38980256e-04 -4.24805528e-0 -1.85607027e-04 4.76778548e-02 3.36268855e-04 -6.70169597e-02 -4.55274939e-06 4.22141453e-03 1.69667725e-02 -2.33038509e-02 -6.26911640e-03]]

Seguidamente predecimos la data de entrenamiento y la data del test(**X_train** y **X_test**), obteniendo como resultado las siguientes matrices de confusión.

Matriz de confusión train:

TABLA IV
MATRIZ DE CONFUSIÓN (Train)

Matriz de confusión (train)		
Predicción/Realidad	**Realmente Falleció**	**Realmente No Falleció**
Predicción de mortalidad	128	16
Predicción de No de no mortalidad	21	14

Matriz de confusión del test

TABLA V
MATRIZ DE CONFUSIÓN (Test)

Matriz de confusión (test)		
Predicción/Realidad	**Realmente Falleció**	**Realmente No Falleció**
Predicción de mortalidad	55	4
Predicción de No de no mortalidad	12	19

Por último, calculamos las principales métricas del modelo:

- **Calculando el Accuracy o Precisión Global del Modelo**
  Accuracy del Train: 0.8229665071770335
  Accuracy del Test: 0.8222222222222222

Nuestro Accuracy es de 82,3% para el train y 82,2% para el test, por lo tanto, nuestro Accuracy al estar por encima del 80% es aceptable.

- **Calculando la Sensibilidad o Recall**
  Sensibilidad del Train: 0.676923076923077
  Sensibilidad del Test: 0.6129032258064516

Nuestra Sensibilidad es de 67,7% para el train y 61,3% para el test, por lo tanto, nuestra Sensibilidad esta por encima del 60%.

- **Calculando la Precisión del Modelo**
  Precisión del Train: 0.73333333333333
  Precisión del Test: 0.8260869565217391

Nuestra precisión del modelo es de 73,3% para el train y 82,61% para el test, por lo tanto, nuestra precisión del modelo es bastante aceptable.

b. **Regresión logística por máxima verosimilitud**

Para poder aplicar este tipo de regresión logística en Python se utilizó la librería **statsmodels.**

"Statsmodels es un paquete de Python que permite a los usuarios explorar datos, estimar modelos estadísticos y realizar pruebas estadísticas" [6]

Luego de importar la librería **statsmodels** se agregó la constante de los datos el **X_train** y el **X_test** utilizando la función **add_constant()** para luego crear el modelo con la función **Logit()** y por último entrenamos el modelo con la con la función **fit().**

Al verificar la **summary** de los resultados del entrenamiento se verificó que dentro de la prueba estadística existían variables excedían el umbral de 0.05, es así que se fue corriendo el modelo varias veces hasta que ninguna variable pueda sobrepasar este umbral, teniendo que eliminar algunas variables. Esto debido a que pasaban del umbral de 0.05 de la prueba estadística y por lo tanto no afectan la salida del target.

Es así que se trabajó con 8 variables predictoras y se obtuvo los siguientes resultados en el **summary.**

Logit Regression Results

Dep. Variable:	DEATH_EVENT	No. Observations:	209
Model:	Logit	Df Residuals:	200
Method:	MLE	Df Model:	8
Date:	Tue, 09 Feb 2021	Pseudo R-squ.:	0.4426
Time:	02:58:53	Log-Likelihood:	-72.221
converged:	True	LL-Null:	-129.56
Covariance Type:	nonrobust	LLR p-value:	4.154e-21

	coef	std err	z	P>|z|	[0.025	0.975]
const	4.8477	1.761	2.753	0.006	1.396	8.299
diabetes	0.6854	0.470	1.460	0.144	-0.235	1.606
sex	-0.8987	0.465	-1.931	0.053	-1.811	0.013
age	0.0529	0.021	2.570	0.010	0.013	0.093
ejection_fraction	-0.1177	0.029	-4.050	0.000	-0.175	-0.061
ejection_fraction_ok	1.9482	1.063	1.832	0.067	-0.136	4.032
serum_sodium_ok	-0.8388	0.527	-1.591	0.112	-1.873	0.195
time	-0.0239	0.004	-5.992	0.000	-0.032	-0.016
total	-0.4147	0.203	-2.042	0.041	-0.813	-0.017

Fig. 25 Summary, logit Regression Results

Se calcularon las métricas del modelo:

- **Calculando el Accuracy o Precisión Global del Modelo**
  Accuracy del Train: 0.861244019138756
  Accuracy del Test: 0.8555555555555555

Nuestro Accuracy es de 86,61% para el train y 85,55% para el test, por lo tanto, nuestro Accuracy al estar por encima del 80% es aceptable.

- **Calculando la Sensibilidad o Recall**
  Sensibilidad del Train: 0.7230769230769231
  Sensibilidad del Test: 0.6774193548387096

Nuestra Sensibilidad es de 83.9% para el train y 84,8% para el test, por lo tanto, nuestra Sensibilidad al estar por encima de 80% es aceptable.

- **Calculando la Precisión del Modelo**
  Precision del Train: 0.8103448275862069
  Precision del Test: 0.875

Nuestra Precisión del Modelo es de 81,03% para el train y 87,05% para el test, por lo tanto, nuestra Precisión del Modelo al estar por encima del 80% es aceptable.

Curva de ROC:

Al calcular la curva ROC se obtiene el área bajo la curva es 0.88 lo cual indica que tenemos un buen modelo.

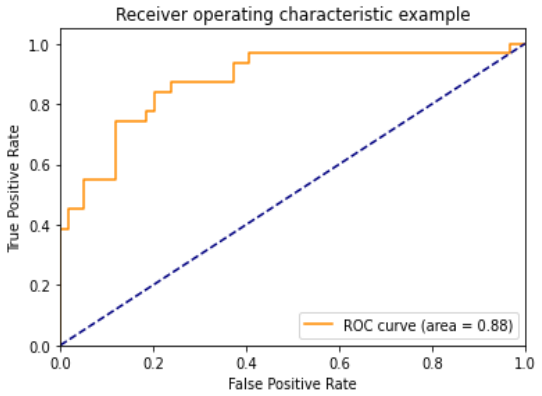


Fig. 26 curva ROC

Obteniendo la Curva ROC, el AUC y el GINI

- **Calculando el AUC del Modelo**
  AUC del Train: 0.8963675213675214
  AUC del Test: 0.8758884636413341

Se verifica que el AUC del train y test de nuestro modelo son bastante aceptables ya que sobrepasan el 87%.

- **Calculando el GINI del Modelo**
  GINI del Train: 0.7927350427350428
  GINI del Test:0.7517769272826682

Se verifica que el GINI del train y test de nuestro modelo son relativamente aceptables ya que sobrepasan el 75%.

Accuracy y Sensibilidad por Umbral de Probabilidad

Se visualiza que el mejor punto de corte es 0.09 ya que Accuracy en este punto es 0.72 y la Sensibilidad es 0.97.

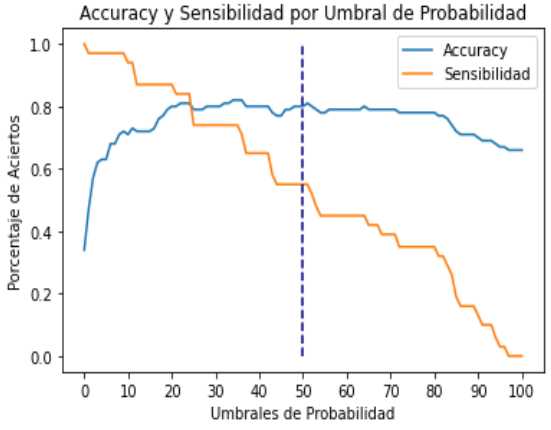


Fig. 27 Accuracy y Sensibilidad por Umbral de Probabilidad (AUC).

3. **Árbol de decisión**

Para aplicar este modelo se importó en Python a través de la librería **skelearn,** el modelo de árbol de decisión, las métricas de los modelos.

Seguidamente predecimos la data de entrenamiento y la data del test(**X_train** y **X_test**), obteniendo como resultado las siguientes matrices de confusión.

Matriz de confusión train:

TABLA VI

MATRIZ DE CONFUSIÓN (Train)

Matriz de confusión (train)		
Predicción/Realidad	**Realmente Falleció**	**Realmente No Falleció**
Predicción de mortalidad	141	9
Predicción de No de no mortalidad	15	44

Matriz de confusión del test

TABLA VII

MATRIZ DE CONFUSIÓN (Test)

Matriz de confusión (test)		
Predicción/Realidad	**Realmente Falleció**	**Realmente No Falleció**
Predicción de mortalidad	51	2
Predicción de No de no mortalidad	16	21

Por último, calculamos las principales métricas del modelo:

- **Calculando el Accuracy o Precisión Global del Modelo**
  Accuracy del Train: 0.8851674641148325
  Accuracy del Test: 0.8

Nuestro Accuracy es de 88,5% para el train y 80,0% para el test, por lo tanto, nuestro Accuracy al ser de 80% es aceptable.

- **Calculando la Sensibilidad o Recall**
  Sensibilidad del Train: 0.7457627118644068
  Sensibilidad del Test: 0.5675675675675675

Nuestra Sensibilidad es de 74,6% para el train y 56,7% para el test, por lo tanto, nuestra Sensibilidad está por encima del 55%.

- **Calculando la Precisión del Modelo**
  Precisión del Train: 0.83018867924528313
  Precisión del Test: 0.9130434782608695

Nuestra precisión del modelo es de 83% para el train y 91,3% para el test, por lo tanto, nuestra precisión del modelo es bastante aceptable.

4. KNN

TABLA VIII

MATRIZ DE CONFUSIÓN (Train)

Matriz de confusión (train)		
Predicción/Realidad	**Realmente Falleció**	**Realmente No Falleció**
Predicción de mortalidad	147	13
Predicción de No de no mortalidad	45	34

Matriz de confusión del test

TABLA IX

MATRIZ DE CONFUSIÓN (Test)

Matriz de confusión (test)		
Predicción/Realidad	**Realmente Falleció**	**Realmente No Falleció**
Predicción de mortalidad	36	7
Predicción de No de no mortalidad	13	4

Por último, calculamos las principales métricas del modelo:

- **Calculando el Accuracy o Precisión Global del Modelo**
  Accuracy del Train: 0.7573221757322176
  Accuracy del Test: 0.6666666666666666

Nuestro Accuracy es de 75,73% para el train y 66,67% para el test, por lo tanto, nuestro Accuracy al estar por encima del 75% es aceptable.

- **Calculando la Sensibilidad o Recall**
  Sensibilidad del Train: 0.43037974683544306
  Sensibilidad del Test: 0.23529411764705882

Nuestra Sensibilidad es de 43,04% para el train y 23,5% para el test, por lo tanto, nuestra Sensibilidad es muy baja con este modelo

- **Calculando la Precisión del Modelo**
  Precisión del Train: 0.723404255319149
  Precisión del Test: 0.36363636363636365

Nuestra precisión del modelo es de 72,34% para el train y 36,36% para el test, por lo tanto, nuestra precisión del modelo es bastante baja.

- **Curva de ROC:**

  Al calcular la curva ROC se obtiene el área bajo la curva es 0.50 lo cual indica que no se cuenta con un buen diagnóstico .

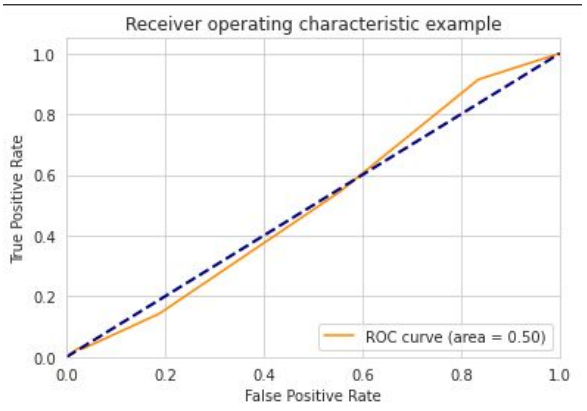


Fig. 28 curva ROC

Obteniendo la Curva ROC, el AUC y el GINI

- **Calculando el AUC del Modelo**
  AUC del Train: 0.7974437343706585
  AUC del Test: 0.5021848739495798

Se verifica que el AUC del train y test de nuestro modelo son poco aceptables ya que no sobrepasan el 80%.

- **Calculando el GINI del Modelo**
  GINI del Train: 0.594887468741317
  GINI del Test: 0.0043697478991595595

Se verifica que el GINI del train y test de nuestro modelo son nada aceptables ya que no sobrepasan el 60%.

Accuracy y Sensibilidad por Umbral de Probabilidad

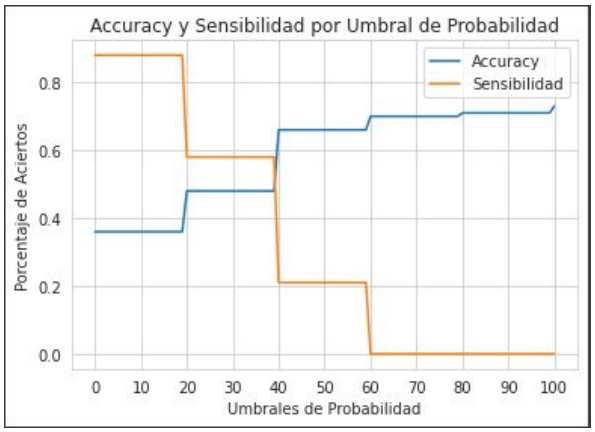


Fig. 29 Accuracy y Sensibilidad por Umbral de Probabilidad (AUC).

V. COMPARACIÓN DE LOS MODELOS UTILIZADOS

Indicador Del Modelo	Modelos utilizados							
	Regresión Logística (GD)		Regresión Logística (MV)		Árboles de Decisión		KNN	
	Train	Test	Train	Test	Train	Test	Train	Test
Accuracy	0.82	0.82	0.86	0.86	0.88	0.8	0.75	0.66
Sensibilidad	0.67	0.61	0.72	0.68	0.74	0.57	0.43	0.23
Precisión	0.73	0.83	0.81	0.88	0.83	0.91	0.72	0.36
AUC	--	--	0.89	0.88	-	-	0.79	0.50
GINI	--	--	0.79	0.75	-	-	0.59	0.00

V. DETALLES TÉCNICOS

Para la implementación de los modelos del presente artículo se utilizó el lenguaje de programación Python para explorar la data, procesar la data y correr los modelos de regresión, lo cual incluyó el uso de las siguientes librerías:

- **Statsmodels** es un paquete de Python que permite a los usuarios explorar datos, estimar modelos estadísticos y realizar pruebas estadísticas. [6]

- **sklearn es** una biblioteca de software de aprendizaje automático para el lenguaje de programación Python.

- **Pandas** es una herramienta de manipulación de datos de alto nivel desarrollada por Wes McKinney. Es construido con el paquete Numpy y su estructura de datos clave es llamada el DataFrame. El DataFrame te permite almacenar y manipular datos tabulados en filas de observaciones y columnas de variables. [7]

- **NumPy** es una biblioteca para el lenguaje de programación Python que da soporte para crear vectores y matrices grandes multidimensionales, junto con una gran colección de funciones matemáticas de alto nivel para operar con ellas. [8]

- **Seaborn** es una biblioteca de visualización de datos de Python basada en matplotlib . Proporciona una interfaz de alto nivel para dibujar gráficos estadísticos atractivos e informativos. [9]

CONCLUSIONES

- Habiendo aplicado el modelo de regresión logística por gradiente del descenso se obtuvo los siguientes resultados: El accuracy salió 82% y 82% para el train y el test respectivamente, la sensibilidad salió 67% y 61% para el traint y test respectivamente y la precisión 73% y 83% para el traint y el test respectivamente, se concluye que se obtuvo un accuray y una precisión bastante aceptable a diferencia de la sensibilidad que es poco aceptable ya que salió con porcentajes bajos.

- Habiendo aplicado el modelo de regresión logística por Máxima Verosimilitud se obtuvo los siguientes resultados: El accuracy salió 86% y 86% para el train y el test respectivamente, la sensibilidad salió 72% y 68% para el traint y test respectivamente, la precisión 81% y 88% para el traint y el test respectivamente, el AUC salió 89% y 88% para el traint y el tes respectivamente, el GINI salió 79% y 75% respectivamente, se concluye que se obtuvo un accuray y una precisión bastante aceptable, una la sensibilidad que es regularmente aceptable ya que salió con porcentajes no muy altos, un AUC y un GINI que superan el 75% lo cual los hace aceptables.

- Habiendo aplicado el modelo de árboles de decisión se obtuvo los siguientes resultados: El accuracy salió 88% y 80% para el train y el test respectivamente, la sensibilidad salió 74% y 57% para el traint y test respectivamente y la precisión 91% y 72% para el traint y el test respectivamente, se concluye que se obtuvo un accuray y una precisión bastante aceptable a diferencia de la sensibilidad que es poco aceptable ya que salió con un bajo porcentaje para el test..

- Habiendo aplicado el modelo de KNN se obtuvo los siguientes resultados: El accuracy salió 75% y 66% para el train y el test respectivamente, la sensibilidad salió 43% y 23% para el traint y test respectivamente, la precisión 72% y 36% para el traint y el test respectivamente, el AUC salió 79% y 50% para el traint y el tes respectivamente, el GINI salió 59% y 75% respectivamente, se concluye que se obtuvo un aceptable, una precisión y una sensibilidad no aceptable ya que salieron con porcentajes muy bajos, un AUC poco aceptable para el test y un GINI muy poco aceptable.

- Habiendo ejecutado 4 modelos predictivos se concluye que el mejor modelo predictivo para cuantificar la mortalidad por

insuficiencia cardiaca es el modelo de regresión logística por máxima verosimilitud.

REFERENCIAS

[1] MODELO PREDICTIVO PARA LA SUPERVIVENCIA Y LA MORTALIDAD PERIOPERATORIA EN PACIENTES CON CARCINOMA RENAL Y EXTENSIÓN VENOSA TUMORAL (D. Juan Ignacio Martínez Salamanca - Universidad Autónoma de Madrid).

[2] Predicción de muerte súbita en pacientes con insuficiencia cardíaca crónica mediante el estudio de la dinámica periódica de la repolarización(S. Palacios, I. Cygankiewicz , A. Bayés-de-Luna , J.P. Martínez, E. Pueyo– XXXVIII Congreso Anual de la Sociedad Española de Ingeniería Biomédica)

[3] DataSet, Predicción de Insuficiencia cardiaca https://www.kaggle.com/andrewmvd/heart-failure-clinical-data

[4] IBERDROLA (2020) QUÉ ES EL 'MACHINE LEARNING' https://www.iberdrola.com/innovacion/machine-learning-aprendizaje-automatico

[5] WIKIPEDIA (2020) STATSMODELS https://en.wikipedia.org/wiki/Statsmodels

[6] LEARNPYTHON (2020) PANDAS BASICS https://www.learnpython.org/es/Pandas%20Basics

[7] WIKIPEDIA (2020) NUMPY https://es.wikipedia.org/wiki/NumPy

[8] SEABORN (2020) SEABORN: VISUALIZACIÓN DE DATOS ESTADÍSTICOS https://seaborn.pydata.org/