Informe del Proyecto I

Inteligencia Artificial

1st Jeffrey Leiva Cascante
Ingeniería en Computación
Instituto Tecnológico de Costa Rica
2021016720
Cartago, Costa Rica
jeffleivajr@estudiantec.cr

2nd Richard Osvaldo León Chinchilla
Ingeniería en Computación
Instituto Tecnológico de Costa Rica
2019003759
Cartago, Costa Rica
r29leonc@estudiantec.cr

***Resumen*—En este proyecto, se aplicaron técnicas de *machine learning* a dos conjuntos de datos distintos utilizando como algoritmos *K-Nearest Neighbors* (KNN) y *Regresión Logística*; el primer conjunto de datos, predice de forma diagnóstica si un paciente tiene o no tiene diabetes, y el segundo predice la decisión de compra de un vehículo por parte de un cliente. Para cada conjunto de datos, se comenzó con la lectura y un análisis exploratorio detallado para comprender las distribuciones y características de los datos. Se utilizaron visualizaciones, tales como gráficos de dispersión e histogramas, para encontrar diferencias, patrones, y características importantes para la comprensión del conjunto de datos. Cada conjunto de datos, se dividió en un $70\,\%$ para entrenamiento, $15\,\%$ para validación, y $15\,\%$ para prueba. El rendimiento de los modelos se evaluaron utilizando métricas de *Accuracy*, *Precision*, *Recall*, *F1-Score* y matriz de confusión. Los resultados mostraron qué, para el primer conjunto de datos la Regresión Logística era una mejor forma de diagnosticar si un paciente tiene diabetes o no y para el segundo conjunto de datos el algoritmo *K-Nearest Neighbors* fue más preciso y eficaz que la *Regresión Logística*.**

***Index Terms*—K-Nearest Neighbors, Machine learning, Matriz de confusión, Métrica de accuracy, Métrica de precision, Métrica de recall, Pandas, Pima indias diabetes dataset, Purchase decision dataset, Regresión logística .**

I. Introducción

El avance de la inteligencia artificial, específicamente las técnicas de machine learning, han revolucionado numerosos campos de investigación, desde la medicina hasta la ingenierías. El machine learning se destaca por su capacidad para analizar grandes cantidades de datos y extraer patrones útiles para la toma de decisiones automatizadas como se explica en [1].

El primer conjunto de datos que se utiliza es *Pima Indians Diabetes Database*. Con base a los datos de [2], este es un conjunto de datos publicado por la University Of California Irvine (UCI). Este consta de 768 observaciones de pacientes con y sin diabetes, considerando diversas medidas diagnósticas incluidas en el conjunto de datos. Para seleccionar las observaciones se aplicaron restricciones distintas restricciones, el sujeto es femenino y debe tener al menos 21 años de edad en el momento que se realizó la evaluación. Sólo se seleccionó un examen por sujeto.

Se tomaron en cuenta varias características para las observaciones:

- Pregnancies: El número de veces que la persona ha estado embarazada.
- Glucose: Concentración de Glucosa en el Plasma a dos horas de realizado un test oral de tolerancia de glucosa (GTT) por sus siglas en inglés.
- Diastolic blood pressure (mm Hg): Presión arterial diastólica.
- Triceps Skin Fold Thickness (mm): Grosor del pliegue cutáneo del tríceps (mm)
- 2-Hour Serum Insulin (μ U/ml) : Insulina sérica de 2 horas (μ U/ml)
- Body Mass Index (Weight in kg / $(Heightinm)^2$)
- Diabetes Pedigree Function: Función del pedigrí de la diabetes.
- Age (years): Edad en años.
- Outcome: Resultado, 0 es negativo, 1 es positivo para diabetes.

Nota: La función (DPF) sirve para proporcionar una síntesis de la historia de la diabetes mellitus en parientes y la relación genética de esos parientes con el sujeto. El DPF utiliza información de padres, abuelos, hermanos y medios hermanos, tíos y tías y medios típs y tías, así cómo de primos. Proporciona una medida de la influencia genétizca esperada de familiares afectados y no afectados sobre la eventual diabetes del sujeto. [3]

El segundo conjunto de datos que se utiliza es *Cars - Purchase decision dataset*. Con base a la información de [4], es un conjunto de datos publicado por el científico de datos Gabriel Santello. Está compuesto por 1000 muestras de clientes, quienes tienen intención de comprar un vehículo, considerando diversos aspectos demográficos y socio-económicos. Se tomaron las muestras considerando cuatro características:

- User ID: Identificación del usuario - número.
- Gender (Male or Female): Género - texto
- Age: Edad en años - número
- AnnualSalary: Salario anual - número.
- Purchased: Comprado, 0 representa que no compró, 1 representa que sí compró.

El presente trabajo tiene como objetivo principal aplicar diversas técnicas de clasificación de datos para los dos conjuntos

de datos seleccionados utilizando los algoritmos de *K-Nearest Neighbors* y *Regresión logística*, analizando sus rendimientos para evaluar su eficacia realizando la clasificación de datos.

II. Metodología

*II-A. **Lectura del conjunto de datos de diabetes***

Al realizar la lectura del archivo .csv correspondiente, se obtiene información sobre los datos presentes en el conjunto de datos de manera general, para entender con que características y valores se va a trabajar.

*II-A1. **Revisión de valores nulos**:* Como se aprecia en el cuadro I, después de cargar el conjunto de datos se revisa si se tienen valores nulos o valores de cero que puedan afectar al modelo que se va a producir. Para este conjunto de datos no se encontraron valores nulos, sin embargo, si se encontraron valores de cero presentes en múltiples características que podrían afectar el modelo. Para la columna de Pregnancies se pueden observar varias pero esto no requiere corrección, puesto que una mujer puede no haber estado embarazada así que son valores válidos.

Característica	Cantidad de ceros
Pregnancies	111
Glucose	5
BloodPressure	35
SkinThickness	227
Insulin	374
BMI	11
DiabetesPedigreeFunction	0
Age	0

Cuadro I
RESUMEN DE CANTIDAD DE CEROS POR COLUMNA ENCONTRADOS

*II-A2. **Tratamiento de características**:* Para tratar el problema de los valores de cero en el conjunto de datos se probaron tres estrategias para ver cual arrojaba mejores resultados en su análisis.

- No realizar ningún tratamiento.
- Reemplazando los ceros por las medias de cada columna, por cada clase.
- Reemplazando los ceros por las medias de cada columna tomando ambas clases.

Después de realizar el análisis, la estrategia que arrojó mejores resultados fue la de reemplazar los ceros por las medias de cada columna, por cada clase, es por esta razón que el entrenamiento de los modelos se realizó después de aplicar este procesamiento.

*II-A3. **Distribución de las clases**:* Como se muestra en el cuadro II, para el número de observaciones de cada clase se obtuvieron los siguientes resultados, donde 0 corresponde a una observación sin diabetes y 1 corresponde a una observación con diabetes.

*II-A4. **Información general del conjunto de datos**:* Al ser muchas las características del conjunto de datos de diabetes, se muestran en dos cuadros distintos: Cuadro III y en el Cuadro IV. El Outcome se omite al ser únicamente valores de 0 o 1.

Outcome	# de observaciones
0	500
1	268

Cuadro II
RESUMEN DE OBSERVACIONES POR CADA CLASE

En el Cuadro III se aprecian las medidas estadísticas para las primeras cuatro variables, destaca el hecho de que la Glucosa. De forma interesante, la Glucosa tiene una media de 121.69 con una desviación estándar de 30.46, lo que indica que esta presenta una mayor variabilidad dependiendo de la observación. En general las características presentan desviaciones estándar con alta variabilidad.

	Pregnancies	Glucose	BloodPressure	SkinThickness
count	768.00	768.00	768.00	768.00
mean	3.84	121.69	72.26	26.635083
std	3.36	30.46	12.11	9.636089
min	0.00	44.00	24.00	7.00
25 %	1.00	99.75	64.00	19.66
50 %	3.00	117.00	72.00	23.00
75 %	6.00	141.00	80.00	32.00
max	17.00	199.00	122.00	99.00

Cuadro III
RESUMEN ESTADÍSTICO #1 DEL CONJUNTO DE DATOS DE DIABETES

En el cuadro IV se presentan las restantes cuatro características presentes en el conjunto de datos, destaca en primer lugar la Insulina con una media de 118.96 y una desviación estándar de 93.55, ya que esto significa que varía en mayor medida dependiendo de la observación que se tenga. Un dato destacable también es que el promedio de edad es de 33 años.

	Insulin	BMI	DiabetesPedigreeFunction	Age
count	768.00	768.00	768.00	768.00
mean	118.96	32.43	0.47	33.24
std	93.55	6.88	0.33	11.76
min	14.00	18.20	0.07	21.00
25 %	68.79	27.50	0.24	24.00
50 %	100.00	32.05	0.37	29.00
75 %	127.25	36.60	0.62	41.00
max	846.00	67.10	2.42	81.00

Cuadro IV
RESUMEN ESTADÍSTICO #2 DEL CONJUNTO DE DATOS DE DIABETES

*II-B. **Lectura del conjunto de datos de decisión de compra***

Al momento de realizar la lectura del archivo .csv, se obtuvo la información general del conjunto de datos para comprender la estructura y características con las que se estará trabajando.

*II-B1. **Revisión de valores nulos**:* Una vez se cargó el conjunto de datos, se revisó si poseía valores nulos o valores con cero que pudieran afectar el modelo. Sin embargo, en este conjunto de datos no se encontró dichos valores.

*II-B2. **Tratamiento de características**:*

- Eliminación de la columna 'User ID': Para efectos del modelo, esta característica es irrelevante, no aporta a la predicción, por lo tanto, se elimina.

- Tratamiento de la columna 'Gender': Debido a que el género está denotado como 'Male' o 'Female', se tratará como 0 para 'Male' (Masculino) y 1 para 'Female' (Femenino).

II-B3. ***Distribución del conjunto de datos:*** Al revisar la columna de 'Purchased' se encontró la siguiente información:

Purchased	Cantidad
0	598
1	402

Cuadro V
DISTRIBUCIÓN DE LA VARIABLE PURCHASED

La cantidad de clientes que sí realizaron la compra del vehículo es de 598 (59,8 %) y 402 (40,2 %) para los que no hicieron la compra. Por lo tanto, se demuestra que este conjunto de datos se encuentra **desbalanceado**.

II-B4. ***Información general del conjunto de datos:*** Se aprecia en el Cuadro VI que el conjunto de datos sobre la decisión de compra de un vehículo, presenta una cantidad 1000 muestras siendo cada una un cliente diferente. Hay una distribución equilibrada entre hombres y mujeres, la edad promedio ronda los 40 años con un rango desde los 18 años hasta los 63 años. El salario anual posee una desviación estándar alta, que indica una amplia variabilidad de ingresos de los clientes.

	Gender	Age	AnnualSalary	Purchased
count	1000.00	1000.00	1000.00	1000.00
mean	0.52	40.11	72689.00	0.40
std	0.50	10.71	34488.34	0.49
min	0.00	18.00	15000.00	0.00
25 %	0.00	32.00	46375.00	0.00
50 %	1.00	40.00	72000.00	0.00
75 %	1.00	48.00	90000.00	1.00
max	1.00	63.00	152500.00	1.00

Cuadro VI
RESUMEN ESTADÍSTICO DEL CONJUNTO DE DATOS DE DECISIÓN DE COMPRA

II-C. Análisis de características del conjunto de datos de diabetes

Para poder analizar las correlaciones y datos relevantes presentes en el conjunto de datos, se utilizaron diversos medios como gráficos de dispersión, histogramas y matrices de correlación. En este caso el análisis se realizó para cada estrategia, a continuación se cubrirá el análisis después de haber elegido la estrategia de preprocesamiento elegida, reemplazando los ceros por las medias de cada columna, por cada clase, el análisis de las estrategias restantes se puede encontrar en el Jupyter Notebook del trabajo.

II-C1. ***Histogramas:*** En la figura 1 se muestran las distribuciones para las siguientes columnas:

- Pregnancies
- Glucose
- BloodPressure
- SkinThickness
- Insulin
- BMI
- Diabetes Pedigree Function
- Age
- Outcome

De igual manera, en la figura 1 se puede observar que muchas características siguen a grandes rasgos distribuciones normales. Sin embargo esto sigue siendo difícil de apreciar para el Insulin y Diabetes Pedigree Function así como para el Age, donde parece que "faltaüna parte en la izquierda del gráfico.

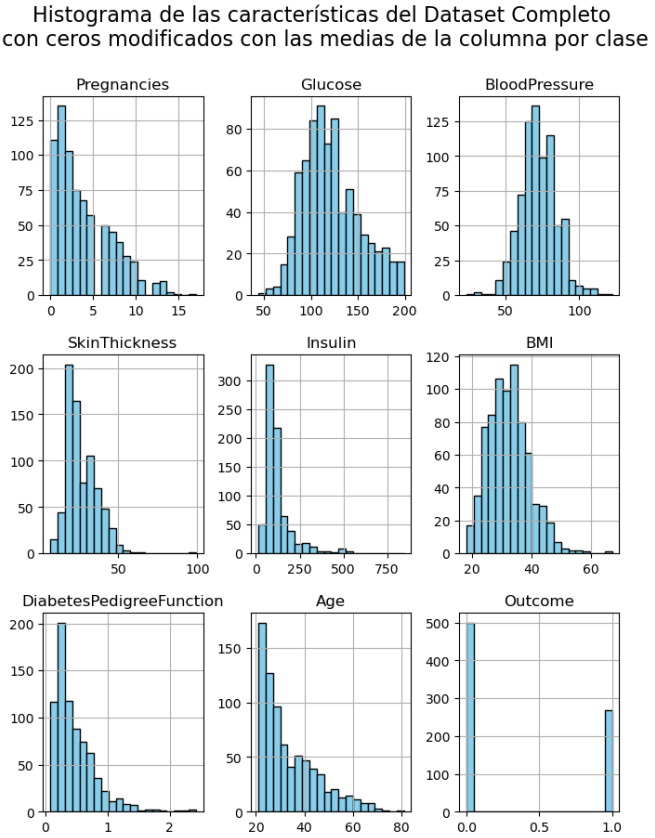

Figura 1. Histogramas Diabetes reemplazando ceros por media de cada clase

II-C2. ***Gráficos de dispersión:*** En la figura 2 se pueden observar las correlaciones existentes entre las variables del conjunto de datos, coloreadas por clase, para ver cuales nos pueden dar una mejor separación de los datos.

En este caso, una combinación de variables que parece dar una mejor separación de los datos por clase es la correlación existente entre las columnas Glucose y BMI, como se puede apreciar en la figura 3

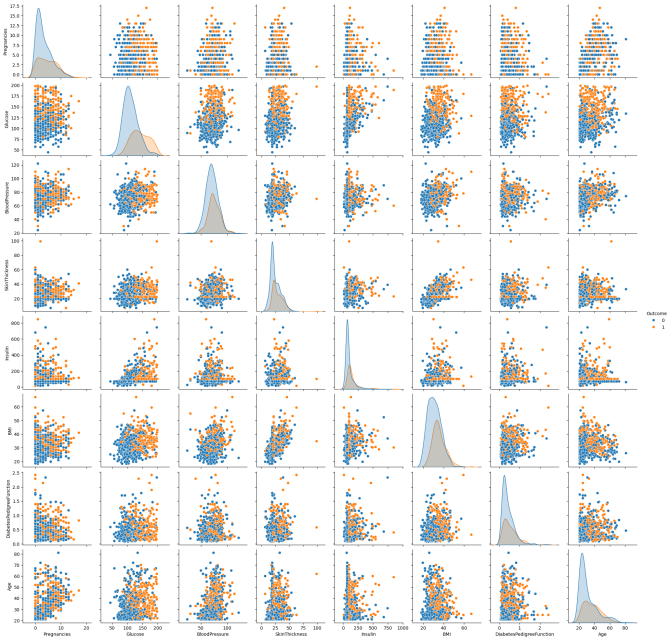

Figura 2. Correlaciones de variables

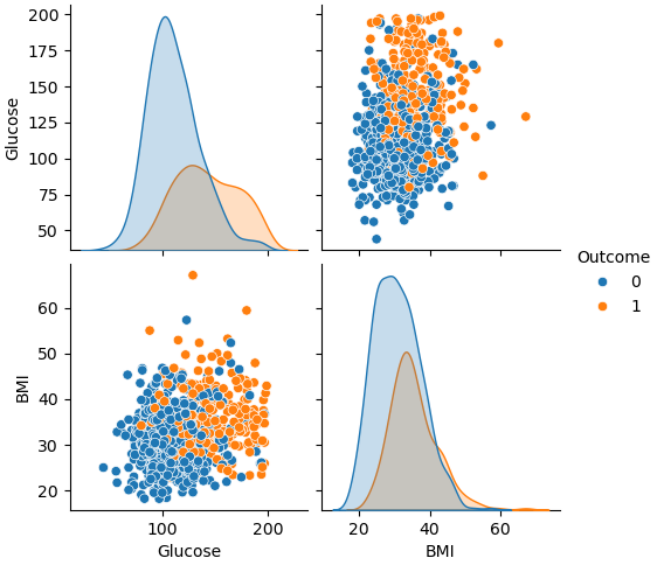

Figura 3. Correlación Glucose-BMI

Esto lo podemos corroborar en la siguiente figura 4, que muestra los valores de correlación para cada par de variable. En la misma se puede notar que la columna Glucose con el Outcome tiene una correlación positiva fuerte de 0.5. Y algo similar sucede con BMI, que tiene una correlación positiva fuerte de 0.32.

Debido a que estas columnas son las que tienen una mayor correlación con el Outcome y parecen separar bien las clases, se entrenará el modelo con los datos de Glucose y BMI, además del Outcome.

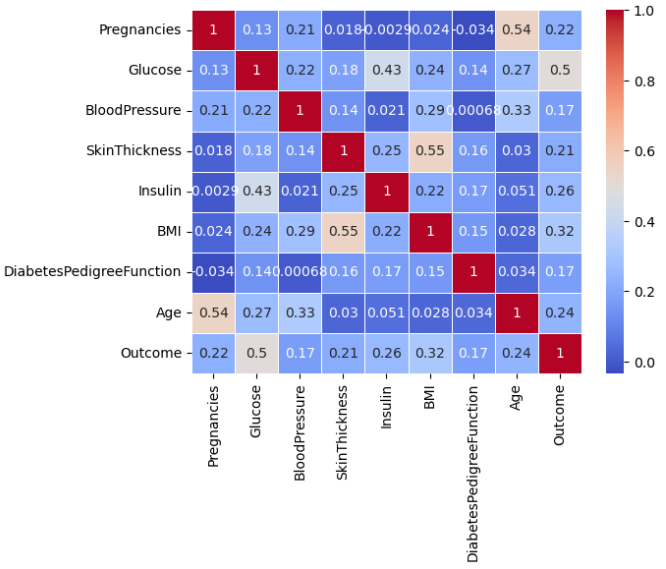

Figura 4. Correlaciones por pares de variables

II-D. Análisis de características del conjunto de datos de decisión de compra

Para visualizar y analizar las diferencias y correlaciones de las características entre los distintos clientes se utilizó gráficos de dispersión e histogramas.

*II-D1. **Histogramas**:* En la Figura 5 se muestran las siguientes distribuciones:

- Gender: Se muestra como la distribución de género tanto para hombres como mujeres se mantiene relativamente equilibrado.
- Age: Respecto a la distribución de la edad, muestra un pico alrededor de los 35 a 40 años, indicando que la mayoría de usuarios rondan dichas edades.
- AnnualSalary: La distribución del salario anual, demuestra que la frecuencia más alta es respecto a los clientes que posee un salario que se sitúa entre los 50 000 a 100 000, dónde se aprecia un pico destacable entre los 80 000 y 90 000.
- Purchased: En la distribución de compras, se aprecia como la mayoría de clientes (rondando los 600) no compraron el vehículo. Por el otro lado, poco más de 400 clientes decidieron sí comprar el vehículo.

En la Figura 6 se aprecia como la mayoría de clientes que no compraron el vehículo se encuentran en el rango de edad entre los 20 a los 45 años aproximadamente, con un pico significativo en los 40 años. Por el otro lado, las personas que sí adquirieron el vehículo se concentran entre los 45 a 60 años.

El comportamiento de compra parece estar influido por la edad; los clientes más jóvenes tienden a no comprar el vehículo, mientras que los mayores tienen una mayor probabilidad de comprar.

En la Figura 7 se aprecia que la mayoría de los clientes que no compraron el vehículo oscilan con salario anual entre 50

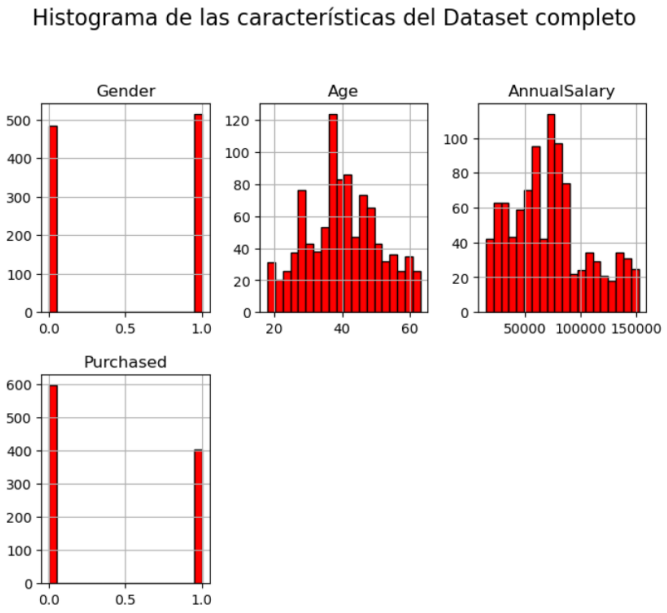

Figura 5. Histogramas Decisión de Compra

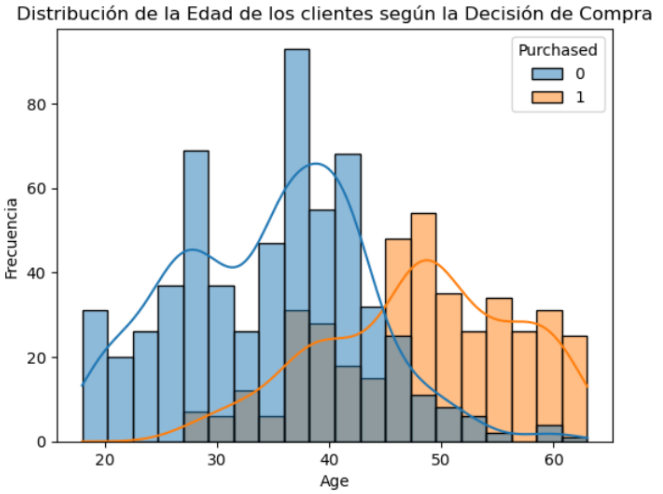

Figura 6. Histograma Edad vs Compra

000 a 80 000 unidades, con un pico destacable en los 60 000. Sin embargo, la probabilidad de compra aumenta con clientes que superan los 100 000.

Hay fuerte correlación entre el salario anual y la decisión de compra. Los clientes que con mayor salario (100 000) son más propensos a realizar la compra. En cambio, los clientes con salarios bajos, parecen no tender a la compra.

*II-D2. **Gráficos de dispersión**:* Como se aprecia la Figura 8 en los gráficos de dispersión se destacan las siguientes distribuciones respecto a la compra de un vehículo:

- Distribución del género: Al revisar las diferentes distribuciones que involucran hombres y mujeres respecto a la compra de un vehículo se mantiene un equilibrio, por lo tanto se confirma que el género no es un factor determinante para la decisión de compra.
- Distribución de Edad vs Salario Anual: Se muestra de forma clara como el salario y la edad son factores determinantes para la decisión de compra de un vehículo, ya que, entre mayor sea el cliente a 40 años y sus ingresos superen los 100 000, la probabilidad de compra es notablemente alta.

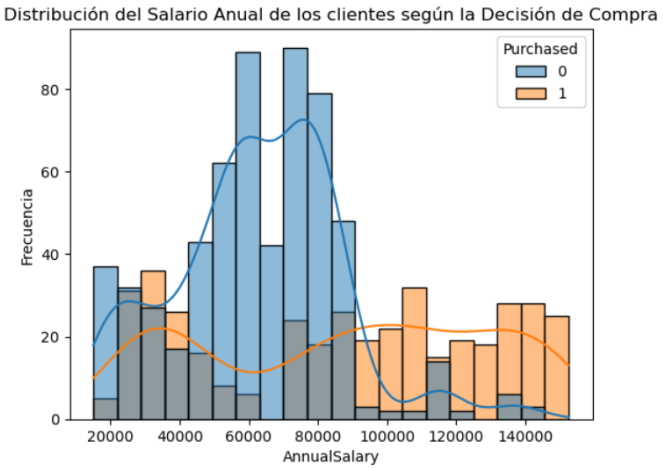

Figura 7. Histograma Salario vs Compra

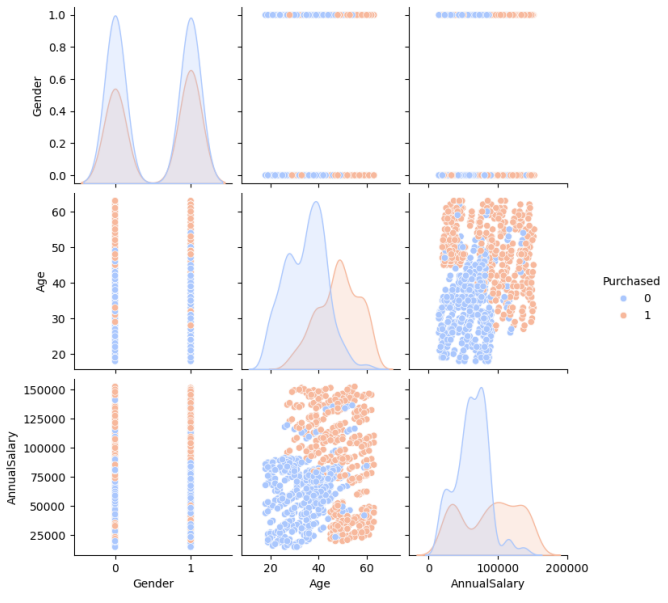

Figura 8. Gráficos de dispersión - Decisión de Compra

II-E. *División y estandarización de los conjuntos de datos*

Para cada conjunto de datos, se dividió en 70 % para entrenamiento, 15 % para validación, y 15 % para pruebas usando *Stratified Sampling* como técnica para subdividir el conjunto de datos, como se muestra a continuación:

```
from sklearn.model_selection import train_test_split
```

```
# 70% de los datos para entrenamiento y el 30% para validacion y pruebas
X_train, X_temp, y_train, y_temp = train_test_split(X,y, test_size=0.3, stratify=y,random_state=10)

# 15% de los datos para validacion y el 15% para pruebas
X_val, X_test, y_val, y_test = train_test_split(X_temp, y_temp, test_size=0.5, stratify=y_temp, random_state=10)
```

Posteriormente, se realizó la estandarización de los datos para mantenerlos en una misma escala y con esto mejorar el rendimiento de los modelos. En el caso del conjunto de datos de diabetes, se aplicó únicamente para la regresión logística, mientras que para el conjunto de datos de decisión de compra se aplicó tanto para KNN como para regresión logística.

```
from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_val = scaler.transform(X_val)
X_test = scaler.transform(X_test)
```

II-F. Entrenamiento del K-Nearest Neighbors para el conjunto de datos de diabetes

Para realizar el entrenamiento de KNN, se entrena con diferentes valores para el hiperparámetro K correspondiente al número de vecinos. Los resultados de cada iteración se validan con el validation set por medio del accuracy y el f1-score que se almacenan en arreglos.

1. Se utiliza la implementación KNN de la biblioteca *Sklearn*, y se crea un rango de 0 a 31, para los valores 'K' a probar, además de los arreglos donde se van a guardar los puntajes para el set de validación tanto para accuracy como para f1-score.

```
n_neighbors_values = range(1, 31)
validation_accuracies = []
validation_f1_scores = []

knn = KNeighborsClassifier()
```

2. Se recorren todos los posibles 'K', se entrena el modelo con ese número de vecinos y se almacenan los resultados del accuracy y f1-score para el conjunto de validación por cada número de vecinos.
   Se escoge el mejor número de vecinos que dio el mejor valor para cada métrica.

```
best_n_neighbors_accuracy = n_neighbors_values[np.argmax (validation_accuracies)]
best_n_neighbors_f1_score = n_neighbors_values[np.argmax (validation_f1_scores)]
```

3. Los mejores resultados los da el parámetro de vecinos encontrado utilizando como guía el parámetro de accuracy, que en este caso corresponde a K = 28. Por lo que el modelo se entrena finalmente con este número de vecinos como hiperparámetro.
4. Entrenamiento vs Validación: Al comparar el accuracy con el mejor K, se obtuvo 0.7574 para el conjunto de entrenamiento, y 0.73275 para el de validación, por lo tanto, se comprueba que no hay *Overfitting* y el modelo generaliza de manera correcta.

II-G. Entrenamiento del K-Nearest Neighbors para el conjunto de datos de decisión de compra

A la hora de realizar el entrenamiento del KNN, se utilizó la técnica *K-Fold Cross Validation* dividiéndolo en 10 partes. Entrenamiento del KNN:

1. Se utiliza la implementación KNN de la biblioteca *Sklearn*, y se crea un rango de 0 a 31, para los valores 'K' a probar.

```

from sklearn.neighbors import KNeighborsClassifier
n_neighbors_values = range(1, 31)
```

2. Se recorren todos los posibles 'K', por cada iteración, se ajusta el modelo y se utiliza la técnica *K-Fold* con F1-Score, luego se guarda el mejor resultado de la iteración. Finalmente se escoge el mejor K de todos los resultados obtenidos.

```

for n in n_neighbors_values:
    knn = KNeighborsClassifier(n_neighbors=n)
    knn.fit(X_train, y_train)

    train_score_cv = cross_val_score(knn, X_train, y_train, cv=10, scoring='f1')
        # Usamos K-Fold Cross Validation con K=10
    train_scores.append(train_score_cv.mean())



best_k = n_neighbors_values[np.argmax(train_scores)]
```

3. Cómo se aprecia en Figura 9, el mejor K corresponde al número 5.
4. Entrenamiento vs Validación: Al comparar el F1-Score con el mejor K, se obtuvo 0.9119 para el conjunto de entrenamiento, y 0.9401 para el de validación, por lo tanto, se comprueba que no hay *Overfitting* y el modelo generaliza de manera correcta.

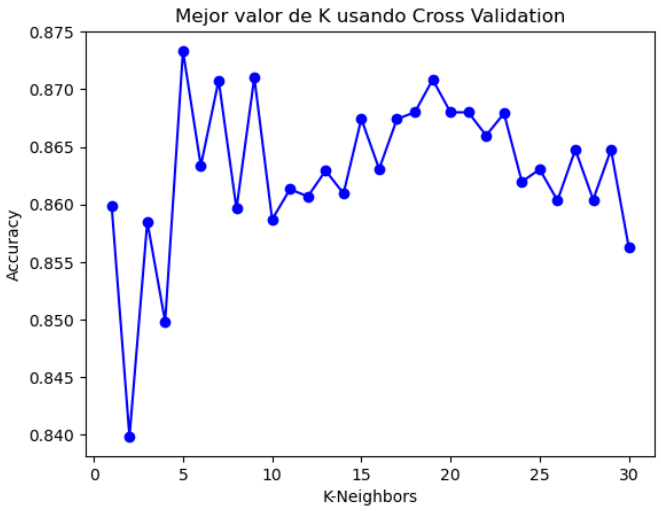

Figura 9. Mejor K - Decisión de compra

II-H. Entrenamiento de Regresión Logística para el conjunto de datos de diabetes

A la hora de realizar el entrenamiento para la regresión logística, se decidió utilizar la técnica de GridSearch para probar múltiples combinaciones de parámetros.

1. Se utiliza la implementación de StandardScaler de la biblioteca **Sklearn** para realizar la estandarización de los datos para la regresión logística.

```
from sklearn.preprocessing import
    StandardScaler

scaler = StandardScaler()

X_train_reg = scaler.fit_transform(X_train)
X_val_reg = scaler.transform(X_val)
X_test_reg = scaler.transform(X_test)
```

2. Además se utiliza la implementación de regresión logística de la biblioteca **Sklearn**

```
from sklearn.linear_model import
    LogisticRegression
...
reglog = LogisticRegression()
```

3. Luego se utiliza la implementación de GridSearchCV para probar diferentes combinaciones de parámetros para el modelo de regresión logística para ver cual combinación arroja mejores resultados, en total se probaron 90 combinaciones de parámetros. Todas las combinaciones de parámetros así cómo su puntaje se pueden encontrar en el Jupyter Notebook.

```
param_grid = {
    'C': [0.01, 0.1, 1, 10, 100],# Inverso de
        la regularizacion
    'penalty': ['l2'], # Tipo de regularizacion
    'solver': ['saga', 'liblinear','lbfgs',
        'sag','newton-cg','newton-cholesky'],
        #solver o algoritmo a utilizar
    'max_iter': [100, 200, 300],
     'class_weight' : ['balanced'],
     'random_state': [10]
}
grid_search = GridSearchCV(reglog, param_grid,
    cv=5, scoring='f1', n_jobs=-1, verbose=1)

grid_search.fit(X_train_reg, y_train)

results = grid_search.cv_results_
```

4. Los mejores y peores resultados de combinaciones son los que se muestran en el Cuadro VII, este indica que los mejores resultados rondan los 0.6389 de media de F1-Score, mientras los peores rondan los 0.6367 de media.

Combination	rank_test_score	mean_test_score
44	1	0.638990
64	1	0.638990
63	1	0.638990
62	1	0.638990
61	1	0.638990
⋮	⋮	⋮
10	73	0.636752
0	73	0.636752
7	88	0.634889
13	88	0.634889
1	88	0.634889

Cuadro VII
RESULTADOS GRIDSEARCH CON VALORES DE RANKING PARA DIABETES.

5. La mejor combinación encontrada es la que se aprecia en el Cuadro XI

Parámetro	Valor
C	0.1
class_weight	balanced
max_iter	100
penalty	l2
random_state	10
solver	saga

Cuadro VIII
MEJORES PARÁMETROS OBTENIDOS DEL GRIDSEARCH - DIABETES.

6. Se evalúan que tan bien califican en el f1-score debido a que el conjunto de datos es desbalanceado, y entonces se escoge la combinación que da el mejor resultado para esta métrica.

```
best_logreg = grid_search.best_estimator_
```

7. La mejor combinación encontrada es la que se aprecia en el Cuadro XI

Parámetro	Valor
C	0.01
class_weight	balanced
max_iter	100
penalty	l2
solver	lbfgs

Cuadro IX
MEJORES PARÁMETROS OBTENIDOS DEL GRIDSEARCH - DECISIÓN DE COMPRA.

8. Al comparar el F1-Score con los mejores parámetros, se obtuvo 0.6386 para el conjunto de entrenamiento, y

0.6117 para el de validación, por lo tanto, se comprueba que no hay *Overfitting* y el modelo generaliza de manera correcta.

II-I. Entrenamiento de Regresión Logística para el conjunto de datos de decisión de compras

Al igual que en el conjunto de datos de diabetes se utilizó la técnica de GridSearch.

1. Implementación de la regresión logística de la biblioteca **SkLearn**

```
from sklearn.linear_model import
    LogisticRegression
from sklearn.model_selection import
    GridSearchCV

log_reg = LogisticRegression(random_state=42)
```

2. Se aplica el entrenamiento del GridSearchCV (utiliza Cross-Validation) para encontrar la mejor combinación de parámetros para el modelo de regresión logística.

```
param_grid = {
    'C': [0.001, 0.01, 0.1, 1, 10, 100], #
        Inversa del parametro de regularizacion
    'penalty': ['l2'], # Tipo de regularizaciun
    'solver': ['liblinear', 'lbfgs',
        'newton-cg', 'sag', 'saga'], #
        Algoritmo a utilizar
    'max_iter': [100, 200, 300], # Numero
        maximo de iteraciones
    'class_weight': ['balanced'] # Peso de las
        clases

}

grid_search = GridSearchCV(log_reg,
    param_grid, cv=10, scoring='f1')
grid_search.fit(X_train, y_train)

results = pd.DataFrame(grid_search.cv_results_)
```

3. Los mejores y peores resultados de combinaciones son los que se muestran en el Cuadro X, este indica que los mejores resultados rondan los 0.7931 de media de F1-Score, mientras los peores rondan los 0.7826 de media.

Combination	rank_test_score	mean_test_score
24	1	0.7931
22	1	0.7931
21	1	0.7931
26	1	0.7931
29	1	0.7931
⋮	⋮	⋮
61	46	0.7826
62	46	0.7826
63	46	0.7826
54	46	0.7826
89	46	0.7826

Cuadro X
RESULTADOS GRIDSEARCH CON VALORES DE RANKING.

4. La mejor combinación encontrada es la que se aprecia en el Cuadro XI
5. Entrenamiento vs Validación: Al comparar el F1-Score, se obtuvo 0.7958 para el conjunto de entrenamiento, y 0.8099 para el conjunto de validación, por lo tanto, se comprueba que no hay *Overfitting* y el modelo generaliza de manera correcta.

Parámetro	Valor
C	0.01
class_weight	balanced
max_iter	100
penalty	l2
solver	lbfgs

Cuadro XI
MEJORES PARÁMETROS OBTENIDOS DEL GRIDSEARCH - DECISIÓN DE COMPRA.

III. RESULTADOS

III-A. Resultados de KNN para el conjunto de diabetes

Los resultados de las predicciones realizadas por el modelo de KNN fueron bastante buenas en el conjunto de training. El accuracy, que fue el parámetro elegido para medir mejor el KNN fue de 0.8362.

Como se muestra en el Cuadro XII, para la clase 0 en cuanto a precision se tiene un valor de 0.85. En cuanto al recall se tiene un valor de 0.91 Además el f1-score se tiene un valor de 0.88, se tiene un buen balance entre precision y recall para la clase sin diabetes (0).

Para la clase 1 en cuanto a precision se tiene un valor de 0.80. En cuanto al recall se tiene un valor de 0.70. Este es uno de los datos más interesantes en este caso, pues se identifica correctamente al 70 % de los casos que tienen diabetes. Para el f1-score se tiene un valor de 0.75 lo que indica que se tiene un buen balance entre precision y recall para la clase con diabetes (1).

	Precision	Recall	F1-Score	Support
0	0.85	0.91	0.88	76
1	0.80	0.70	0.75	40
accuracy			0.84	116
macro avg	0.83	0.80	0.81	116
weighted avg	0.83	0.84	0.83	116

Cuadro XII
MÉTRICAS RESULTADO DE APLICAR PREDICCIÓN AL CONJUNTO DE TESTING UTILIZANDO EL MODELO DE KNN

*III-A1. **Matriz de Confusión**:* En la figura 10 se puede ver una matriz de confusión para la clasificación, en ella podemos observar que para el conjunto de testing, para la clase 0, se han clasificado correctamente 69 ejemplos, y hubo 7 errores. Para la clase 1 (con diabetes) se tuvieron 28 aciertos y 12 errores.

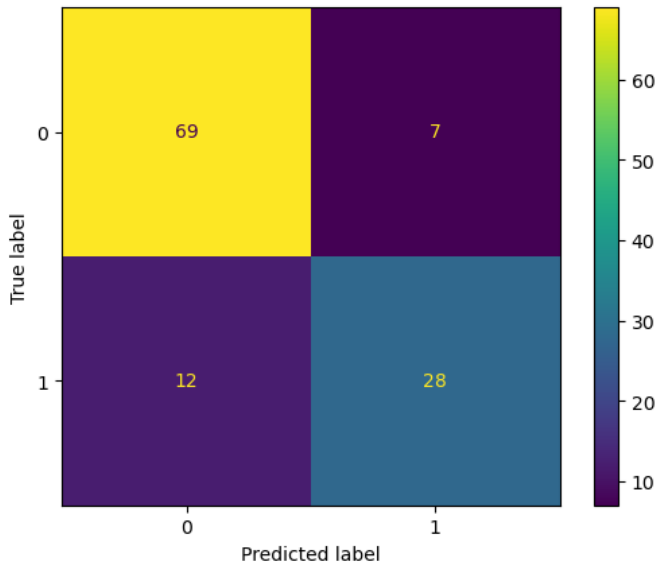

Figura 10. Matriz de Confusión para KNN

*III-B. **Resultados de KNN para el conjunto de decisión de compra***

Usando el mejor K encontrado en la fase de entrenamiento, se procede con la fase de pruebas. Con base al reporte de clasificación se encuentra en el Cuadro XIII se destaca lo siguiente:

- Accuracy: El modelo tiene un accuracy de 0.91, lo que significa que el modelo predijo correctamente el 91 % de todos los casos.
- Precision: El modelo es más confiable al identificar correctamente a los clientes que no compran el vehículo (clase 0) pero es poco menos preciso al identificar a aquellos que sí compran (clase 1)
- Recall: Aunque el modelo es menos preciso al predecir la clase 1, tiene un buen desempeño para detectar correctamente aquellos que compran un vehículo, es importante si se desea que el modelo minimice los falsos negativos (Clientes que compran pero fueron predichos como no compradores)
- F1-Score: Hay un buen balance general entre precision y recall para ambas clases, lo que demuestra que el modelo tiene un buen desempeño general en la clasificación de ambas categorías.

	Precision	**Recall**	**F1-score**	**Support**
0	0.96	0.89	0.92	89
1	0.85	0.95	0.90	61
accuracy			0.91	150
macro avg	0.91	0.92	0.91	150
weighted avg	0.92	0.91	0.91	150

Cuadro XIII
REPORTE DE CLASIFICACIÓN DE KNN - DECISIÓN DE COMPRA

*III-B1. **Matriz de Confusión**:* La matriz de confusión que se muestra en la 11, denota que en la clase 0, el modelo clasificó correctamente 79 de 89 casos, pero cometió 10 errores, prediciendo incorrectamente que esos clientes comprarían el vehículo. Para la clase 1, el modelo clasificó correctamente 58 de 61 casos, con solo 3 errores donde predijo que estos clientes no comprarían el vehículo.

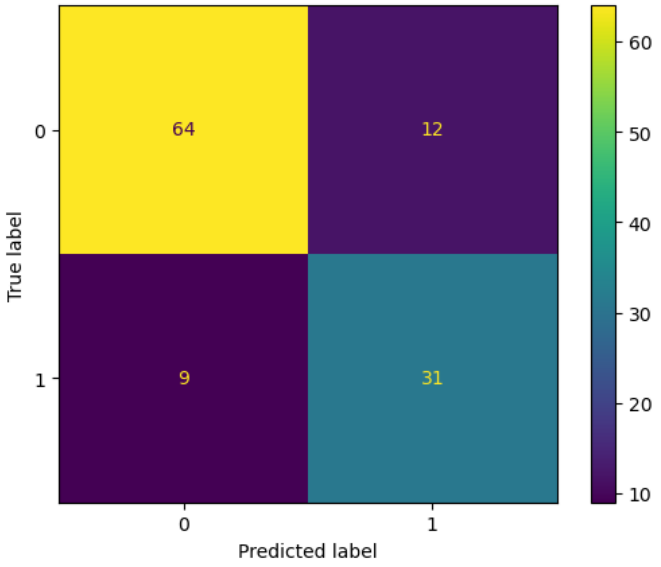

Figura 11. Matriz de Confusión para KNN - Decisión de compra

*III-C. **Resultados de Regresión Logística para el conjunto de diabetes***

Los resultados de predicciones realizadas por el modelo de regresión logística para el conjunto de training fueron buenas.

Como se muestra en el cuadro XIV, para la clase en cuanto a precision se tiene un valor de 0.88. En cuanto al recall se tiene un valor de 0.84. Además, para el f1-score se tiene un valor de 0.86 lo que indica que se tiene un buen balance entre precision y recall para la clase sin diabetes (0)

Para la clase 1, en precision se tiene un valor de 0.72. En cuanto al recall se tiene un valor de 0.78, lo que indica que de todas las instancias que son con diabetes (1), el 78 % se clasificó correctamente como con diabetes (1), esto supone una mejoría con respecto a KNN. Finalmente, para el f1-score se tiene un valor de 0.75 lo que indica que se tiene un buen balance entre precision y recall para la clase con diabetes (1)

	Precision	**Recall**	**F1-Score**	**Support**
0	0.88	0.84	0.86	76
1	0.72	0.78	0.75	40
accuracy			0.82	116
macro avg	0.80	0.81	0.80	116
weighted avg	0.82	0.82	0.82	116

Cuadro XIV
MÉTRICAS RESULTADO DE APLICAR PREDICCIÓN AL CONJUNTO DE TESTING UTILIZANDO EL MODELO DE REGRESIÓN LOGÍSTICA

III-C1. ***Matriz de Confusión****:* A continuación, en la figura 12 se pueden ver una matriz de confusión para la clasificación, en ella podemos observar que para el conjunto de testing, para la clase 0, se han clasificado correctamente 64 ejemplos, y hubo 12 errores. Para la clase 1 (con diabetes) se tuvieron 31 aciertos y 9 errores.

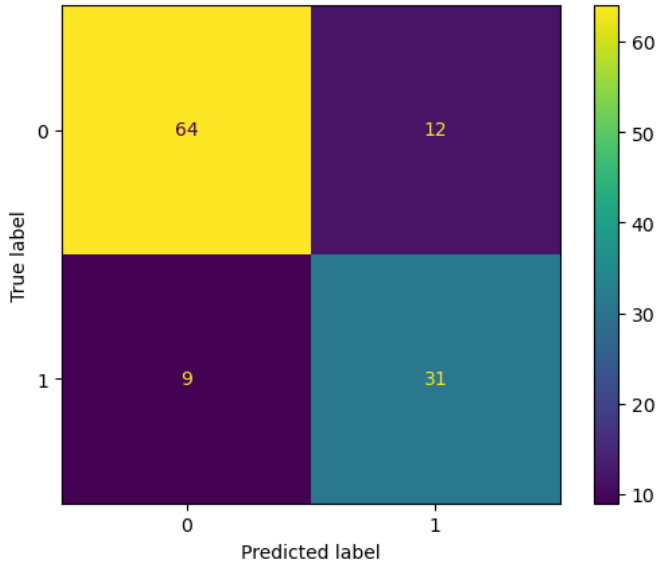

Figura 12. Matriz de Confusión para Regresión Logística

III-D. ***Resultados de Regresión Logística para el conjunto de decisión de compra***

Usando la mejor combinación de parámetros encontrada por el GridSearch, se procede con la fase de pruebas. En el reporte de clasificación mostrado en el Cuadro XV se destaca lo siguiente:

- Accuracy: El accuracy general del modelo es de 0.87, lo que significa que en el 87 % de los casos modelo tiene un buen rendimiento al predecir.
- Precision: El modelo se comporta mucho mejor a la hora de predecir los clientes que no compraron, por el otro lado, para los que sí compraron, baja considerablemente su acierto, dando un promedio global de 86 %
- Recall: El modelo tiene un recall mayor para la clase 1, indicando que el modelo es más preciso a predecir los compradores reales.
- F1-Score: Se muestra que el modelo se desempeña bien para ambas clases, aunque con un tendencia a predecir mejor la clase 0.

	Precision	Recall	F1-score	Support
0	0.92	0.85	0.88	89
1	0.81	0.89	0.84	61
accuracy			0.87	150
macro avg	0.86	0.87	0.86	150
weighted avg	0.87	0.87	0.87	150

Cuadro XV
REPORTE DE CLASIFICACIÓN DE REGRESIÓN LOGÍSTICA - DECISIÓN DE COMPRA

IV. DISCUSIÓN

IV-A. ***Discusión para el conjunto de diabetes***

Basándonos en los Cuadros XII y XIV este caso, en cuanto a accuracy, la mejor puntuación la tiene el modelo de KNN con 0.84 por encima del 0.82 de la Regresión logística, es una diferencia bastante pequeña.

Para la métrica de precision, tanto para el KNN como para la Regresión Logística el precision fue mejor para la clase sin diabetes o 0.

Para el caso de la clase con diabetes o 1. En el KNN la precision es de 0.80, mientras que para la regresión es de 0.72.

Sin embargo, como el problema consiste en clasificar casos de diabetes, nos importa más el recall que la precisión puesto que es de mayor importancia detectar los casos positivos de diabetes. Esto debido a que es más grave no identificar a alguien con la enfermedad (falso negativo) que clasificar a alguien como diabético cuando no lo es.

En este aspecto, el recall para la clase 1 en el KNN es de 0.70 mientras que para la Regresión Logística es de 0.78. Es decir un 8 % más de recall.

Es por esto que se concluye que el modelo de regresión logística es mejor para este caso.

IV-B. ***Discusión para el conjunto de decisión de compra***

A continuación se muestra el análisis comparativo por cada métrica entre el modelo implementado con K-Nearest Neighbors y Regresión Logística:

IV-B1. ***Accuracy****:* El modelo KNN tiene un accuracy de 0.91, mientras que la regresión logística alcanza un accuracy de 0.87. Esto indica que KNN es globalmente más preciso para predecir tanto a compradores como no compradores.

IV-B2. ***Precision****:* El modelo KNN presenta un mayor precision que la regresión logística en ambas clases, especialmente en la clase 0, donde KNN alcanza un 0.96 frente al 0.92 de la regresión logística.

Para la clase 1, KNN también es más preciso, con un precision de 0.85 frente a 0.81 de la regresión logística. Esto demuestra que el modelo KNN es ligeramente superior en identificar correctamente a los compradores y no compradores.

*IV-B3. **Recall**:* Para la clase 0, KNN tiene un recall más alto que la regresión logística, lo que indica que KNN es mejor prediciendo la mayoría de los no compradores.

En la clase 1, KNN destaca significativamente con un recall de 0.95, superando a la regresión logística que tiene un recall de 0.89. Esto demuestra que el KNN es mucho más efectivo para identificar a los compradores reales.

*IV-B4. **F1-Score**:* En el F1-Score, que equilibra precision y recall, el mejor es KNN para ambas clases. Para la clase 0, KNN tiene un 0.92, mientras que la regresión logística tiene 0.88.

Para la clase 1, KNN tiene un F1-Score de 0.90, nuevamente superando a la regresión logística que tiene 0.84. Esto refleja que KNN es un modelo más equilibrado en términos de precision y recall.

*IV-B5. **Mejor métrica**:* A pesar de que el KNN supera en todas las métricas a la regresión logística, en el contexto de decisión de compra de un vehiculo en términos de maximizar las oportunidades de venta, el recall es la métrica más importante debido a que asegura que la mayoría de los compradores sean correctamente identificados, en donde el KNN es ampliamente superior.

V. Conclusiones

*V-A. **Diabetes***

En este estudio se llevó a cabo un análisis comparativo entre dos algoritmos de clasificación, K-Nearest Neighbors (KNN) y Regresión Logística, para predecir la presencia de diabetes a partir de un conjunto de datos con características relacionadas con la salud. Ambos modelos fueron evaluados utilizando diversas métricas. Sin embargo, la selección final del modelo estuvo basada en el recall para la clase de pacientes con diabetes.

- Las características de Glucose y BMI fueron fundamentales para poder identificar y separar los casos de diabetes positivos de los negativos, ya que estos tenían una mayor relación con el diagnóstico que se daba a las distintas observaciones.
- Regresión Logística demostró ser el modelo más adecuado para este problema, ya que obtuvo un mejor recall en la clase de pacientes con diabetes. Esto es crucial en un entorno clínico, donde es más importante minimizar los falsos negativos (es decir, no diagnosticar incorrectamente a un paciente con diabetes) que obtener una alta precisión global.
- El modelo KNN, aunque inicialmente mostró un desempeño competitivo, tuvo un recall inferior para la clase de diabetes, lo que implica un mayor riesgo de no detectar correctamente casos positivos de la enfermedad.

*V-B. **Decisión de compra***

- El modelo K-Nearest Neighbors (KNN) demostró un rendimiento superior en la mayoría de las métricas clave comparado con la regresión logística, lo que lo convierte en una opción más eficaz para predecir la compra de vehículos
- Las características de Salario Anual y Edad fueron fundamentales para identificar patrones de compra entre los clientes. Se observó que a medida que estos valores aumentan, también aumenta la probabilidad de compra.
- El tratar la característica de Gender como 0 o 1, permitió concluir que no existe una correlación significativa entre el género y la decisión de compra, lo que sugiere que el género no es un factor determinante en este contexto.
- Para maximizar las ganancias en un contexto empresarial, la métrica de Recall es crucial, ya que garantiza que la mayoría de los compradores potenciales sean identificados correctamente, minimizando los falsos negativos y aprovechando más oportunidades de venta.

Referencias

[1] AWS, "¿qué es el machine learning?" Disponible: https://aws.amazon.com/es/what-is/machine-learning/, [Accesado: Sept. 08, 2024].

[2] U. M. Learning, "Pima indians diabetes database," Disponible: https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database, [Accesado: Sept. 08, 2024].

[3] E. J. D. W. K. W. . J. R. Smith, J.W., "Using the adap learning algorithm to forecast the onset of diabetes mellitus. in proceedings of the symposium on computer applications and medical care (pp. 261–265)." https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2245318/pdf/procascamc00018-0276.pdf, 1988, [Accesado: Sept. 08, 2024].

[4] G. Santello, "Cars - purchase decision dataset," Disponible: https://aws.amazon.com/es/what-is/machine-learning/, [Accesado: Sept. 08, 2024].

VI. RÚBRICA

Conjunto de datos de Diabetes		
Criterios	**Puntuación máxima**	**Puntuación obtenida**
Análisis del conjunto de datos y features	5	
Análisis de regresión logística	15	
Análisis de KNN	15	
Comparación de modelos	10	
Conjunto de datos seleccionado		
Análisis del conjunto de datos y features	5	
Análisis de regresión logística	15	
Análisis de KNN	15	
Comparación de modelos	10	
Aspectos Generales		
Completitud de entregables	5	
Estructura de artículo científico	5	
Aspectos Extra (SRE)		
Criterio	**Puntuación máxima**	**Puntuación obtenida**
Usar Gitflow como proceso de colaboración y utilizar tags de versionamiento (main branch). Deben participar los 2 integrantes.	5	
Total	**105**	