PROYECTO 2 – REDES NEURONALES

SANTIAGO CAMILO REY BENAVIDES
TOMAS FIGUEROA SIERRA
SANTIAGO VIDES SALCEDO
JESUS DANIEL MOLINA EMILIANI

ANÁLISIS DE CLASIFICACIÓN EMPLEANDO REDES NEURONALES FEED-FORWARD

DOCENTE:
JULIO OMAR PALACIO NIÑO

PONTIFICIA UNIVERSIDAD JAVERIANA
FACULTAD DE INGENIERIA
INGENIERIA DE SISTEMAS
BOGOTA D.C
2023

CONTENIDO

LISTA DE ILUSTRACIONES

RESUMEN

Este proyecto se enfoca en el diseño y evaluación de modelos de redes neuronales para la clasificación precisa de la calidad del vino, utilizando un dataset proporcionado por el repositorio UCI. Se establecieron objetivos claros que abarcan desde el análisis exhaustivo del dataset de vinos blancos hasta la construcción y comparación de diferentes modelos neurales.

El análisis inicial se centró en comprender las características químicas presentes en el dataset, incluyendo la variable objetivo de calidad del vino. Se llevaron a cabo procesos de preprocesamiento detallados, como la unión de datasets, la codificación de variables categóricas y la identificación y tratamiento de datos atípicos.

La construcción del dataset involucró una cuidadosa división en conjuntos de entrenamiento y prueba, con el objetivo de evaluar el rendimiento de los modelos ante diferentes proporciones de datos. Tres arquitecturas neuronales fueron diseñadas y evaluadas: un Perceptrón, una Red Neuronal con una capa oculta y otra con dos capas ocultas. Cada modelo fue preparado, entrenado y evaluado rigurosamente.

Los resultados revelaron métricas de desempeño similares entre los modelos, con precisión en torno al 52-53%. Se observaron ligeras diferencias en precisiones, recalls y F1 Scores, destacando un leve mejor rendimiento en la Red Neuronal con una sola capa oculta.

En resumen, este proyecto ofrece una exploración integral de modelos neurales para la clasificación de calidad del vino, destacando la importancia del preprocesamiento de datos y la evaluación comparativa de diferentes arquitecturas neuronales. Los hallazgos indican un desempeño comparable entre los modelos, resaltando la necesidad de investigaciones adicionales para lograr mejoras significativas en la predicción de la calidad del vino.

PALABRAS CLAVE: Redes Neuronales, Análisis de Clasificación, Dataset UCI

ENLACE REPOSITORIO GOOGLE COLAB:
https://colab.research.google.com/drive/1Kn6tFDdMZ_bD19kTtLdsZ6xLrIlTkjSE?usp=sharing

1. INTRODUCCIÓN

Las Redes Neuronales se han destacado como una herramienta poderosa en este contexto, especialmente en tareas de clasificación de información. Este trabajo se

adentra en el análisis y la comparación de distintas arquitecturas de Redes Neuronales para la clasificación precisa de datos, centrándose en un conjunto específico: la calidad de vinos, utilizando el dataset disponible en el repositorio UCI.

La relevancia de este estudio radica en la necesidad de comprender cómo diferentes estructuras de redes neuronales se comportan en la tarea de clasificación, y específicamente en la determinación de la calidad del vino. El análisis detallado de diversas arquitecturas de redes neuronales permitirá evaluar su efectividad y proporcionar insights valiosos sobre la selección y configuración óptima de modelos para tareas similares en diferentes dominios.

Al explorar y comparar estas arquitecturas, este trabajo busca no solo obtener resultados de clasificación precisos, sino también comprender cómo varían los resultados en función de la complejidad de la red neuronal utilizada, así como la influencia de los datos de entrenamiento y prueba en el rendimiento del modelo.

2. OBJETIVOS

2.1 OBJETIVO GENERAL

Explorar, diseñar y evaluar modelos de redes neuronales para la clasificación precisa de la calidad del vino, utilizando el dataset proporcionado por el repositorio UCI.

2.2 OBJETIVOS ESPECÍFICOS

1. Realizar un análisis exhaustivo del conjunto de datos de Wine Quality, identificando las características relevantes y la variable objetivo para la clasificación.
2. Preprocesar los datos, incluyendo la codificación de variables categóricas, normalización de datos, manejo de valores nulos y detección y tratamiento de datos atípicos.
3. Diseñar y construir múltiples modelos de redes neuronales, incluyendo perceptrón, red neuronal con una capa oculta y red neuronal con dos capas ocultas, con sus respectivas funciones de activación y sesgo.
4. Evaluar y comparar el rendimiento de los modelos utilizando métricas de evaluación estándar (accuracy, precisión, recall, f1-score).

5. Analizar en profundidad los resultados para comprender cómo varían en función de los cambios en la arquitectura de las redes neuronales y la proporción de datos de entrenamiento y prueba.

3. DESARROLLO DEL PROYECTO

3.1 DATASET - WINE QUALITY

Para este proyecto, se escogieron trabajar con los datos del vino blanco para el entrenamiento de los modelos del perceptrón y de las redes neuronales.

3.2 COMPRENSIÓN DEL DATASET

3.2.1 ¿Qué información presenta el dataset?

El conjunto de datos proporcionado contiene atributos numéricos que representan diversas características químicas de muestras de vino. Cada entrada en el dataset está evaluada numéricamente y está compuesta por los siguientes atributos:

1. **Fixed Acidity:** Los ácidos, como el tartárico, málico, cítrico y succínico, son esenciales en el vino, influenciando su sabor. Se dividen en volátiles y no volátiles, siendo los fijos un elemento crucial.
2. **Volatile Acidity:** Este proceso transforma el vino en vinagre. Hay límites legales para la acidez volátil en diferentes tipos de vino en los Estados Unidos.
3. **Citric Acid:** Presente como ácido fijo en los vinos, expresado en g/dm3 en los datos.
4. **Residual Sugar:** Es el azúcar restante después de la fermentación, expresado en g/dm3 en los datos.
5. **Chlorides:** Contribuyen a la salinidad del vino y se expresan en una medida específica.
6. **Sulfur Dioxide:** Su proporción libre y unida es crucial para los enólogos y se mide en mg/dm3.
7. **Total Sulfur Dioxide:** Suma del dióxido de azufre unido y libre, con límites legales para su cantidad en vinos.
8. **Density:** Usada para medir la conversión de azúcar a alcohol, expresada en g/cm3.

9. **Sulphates:** Esenciales en la vinificación, se expresan en g/dm3.
10. **Alcohol:** El porcentaje de alcohol presente en el vino, expresado en % vol.
11. **Quality:** Evaluada por expertos en vino en una escala del 0 al 10, siendo el número la mediana de al menos tres evaluaciones.
12. **pH:** Indica la acidez o basicidad del vino en una escala numérica, donde pH < 7 es ácido, > 7 básico y 7 neutro.

Variable objetivo: Calidad - Quality

3.2.2 Análisis Grafico de distribución y correlaciones:

3.2.2.1 Distribución y descripción de cada atributo:

Aquí una descripción de cada atributo del conjunto de datos de vino blanco:

```
vino_white.describe()
```

Ilustración 1 Descripción atrivutos del vino blanco

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality
count	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000
mean	8.319637	0.527821	0.270976	2.538806	0.087467	15.874922	46.467792	0.996747	3.311113	0.658149	10.422983	5.636023
std	1.741096	0.179060	0.194801	1.409928	0.047065	10.460157	32.895324	0.001887	0.154386	0.169507	1.065668	0.807569
min	4.600000	0.120000	0.000000	0.900000	0.012000	1.000000	6.000000	0.990070	2.740000	0.330000	8.400000	3.000000
25%	7.100000	0.390000	0.090000	1.900000	0.070000	7.000000	22.000000	0.995600	3.210000	0.550000	9.500000	5.000000
50%	7.900000	0.520000	0.260000	2.200000	0.079000	14.000000	38.000000	0.996750	3.310000	0.620000	10.200000	6.000000
75%	9.200000	0.640000	0.420000	2.600000	0.090000	21.000000	62.000000	0.997835	3.400000	0.730000	11.100000	6.000000
max	15.900000	1.580000	1.000000	15.500000	0.611000	72.000000	289.000000	1.003690	4.010000	2.000000	14.900000	8.000000

- **Fixed Acidity:** Esta característica representa la concentración de ácidos no volátiles en el vino. Con una media de 8.32 y una desviación estándar de 1.74, los valores oscilan entre 4.6 y 15.9. Esto indica que la mayoría de los vinos tienen una concentración de ácidos fijos alrededor de 8, aunque algunos pueden ser considerablemente más bajos o altos.
- **Volatile Acidity:** Esta medida indica la presencia de ácidos volátiles en el vino. Con una media de 0.528 y una desviación estándar de 0.179, la mayoría de los vinos tienden a tener valores alrededor de 0.5, aunque hay cierta variación, desde 0.12 hasta 1.58.
- **Citric Acid:** Esta variable indica la cantidad de ácido cítrico presente. Con una media de 0.271 y una desviación estándar de 0.195, la mayoría de los vinos tienen una concentración moderada de ácido cítrico, aunque algunos pueden no tenerlo presente en absoluto.

- **Residual Sugar:** Representa la cantidad de azúcar restante después de la fermentación. Con una media de 2.54 y una desviación estándar de 1.41, la mayoría de los vinos tienen una cantidad moderada de azúcar residual, con valores entre 0.9 y 15.5.
- **Chlorides:** Indica la concentración de cloruros en el vino. Con una media de 0.0875 y una desviación estándar de 0.0471, la mayoría de los vinos tienen una concentración baja de cloruros, aunque hay variabilidad entre 0.012 y 0.611.
- **Free Sulfur Dioxide y Total Sulfur Dioxide:** Estos dos atributos miden la cantidad de dióxido de azufre libre y total. Presentan desviaciones estándar considerables, lo que indica una amplia variabilidad en su concentración en los vinos.
- **Density:** Representa la densidad del vino. La media es 0.997 con poca variación, lo que sugiere que la mayoría de los vinos tienen densidades similares.
- **pH:** Indica la acidez o basicidad del vino. Con una media de 3.31 y una desviación estándar de 0.154, la mayoría de los vinos tienen un pH entre 2.74 y 4.01.
- **Sulphates:** Esta característica representa la cantidad de sulfatos en el vino. Con una media de 0.658 y una desviación estándar de 0.169, los vinos tienden a tener concentraciones moderadas de sulfatos.
- **Alcohol:** Muestra el contenido de alcohol. La media es 10.42, con valores entre 8.4 y 14.9. La mayoría de los vinos tienen un contenido alcohólico alrededor de 10%.
- **Quality:** Es la calificación del vino, con valores entre 3 y 8. La media es 5.64, lo que sugiere que la mayoría de los vinos tienden a tener calificaciones promedio, aunque algunos destacan con puntajes más altos o bajos.

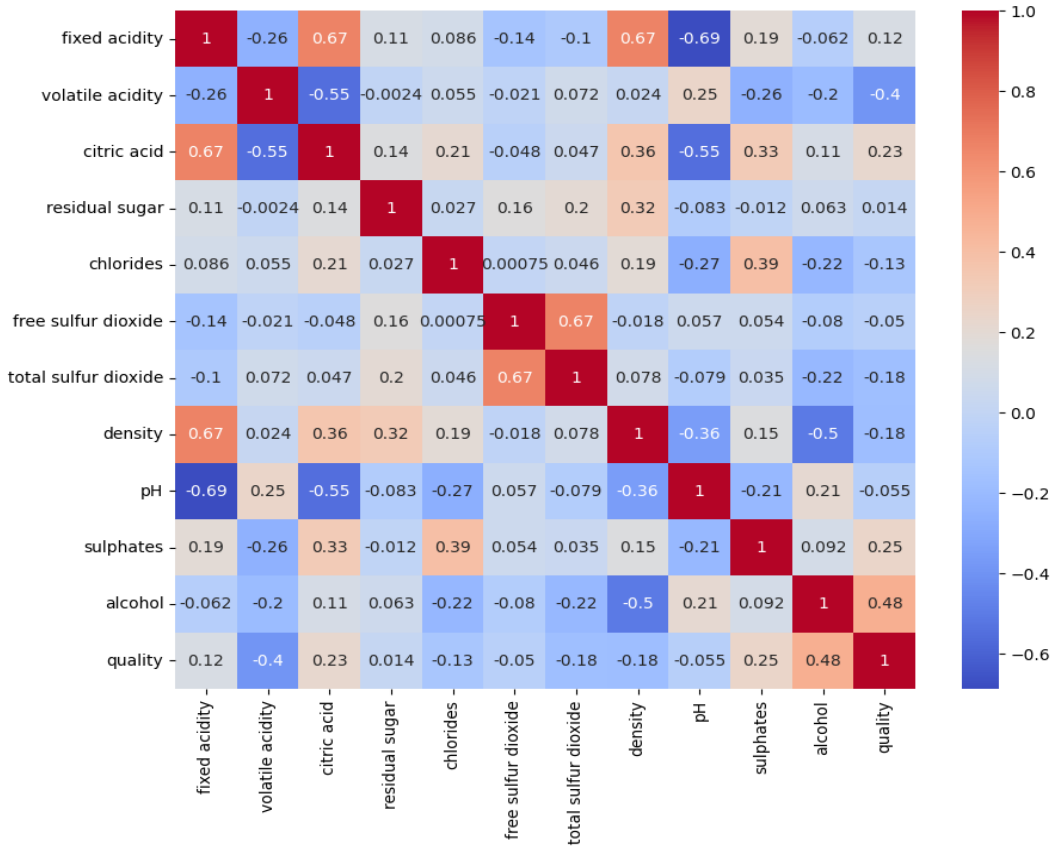
3.2.2.2 Matriz de correlación:

Dado que puede ser algo difícil interpretar gráficos, también es una buena idea trazar una matriz de correlación. Esto dará información más rápidamente sobre qué variables se correlacionan. Como era de esperar, hay algunas variables que se correlacionan, como la densidad y el azúcar residual o como el dióxido de azufre libre y el dióxido de azufre total se correlacionaran.

```
import seaborn as sns
```

```
import matplotlib.pyplot as plt
correlation_matrix = datos_mod_white.corr()
plt.figure(figsize=(10, 8))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm')
plt.show()
```

Ilustración 2 Correlación de las varibales del vino blanco



3.2.2.2 Gráfico de barras Calidad - Alcohol:

Otra buena idea es realizar un gráfico de barras donde el eje “x” representa la calidad del vino, extraída de la columna 'quality' del dataframe 'datos_mod_white', y el eje “y” muestra los grados de alcohol, obtenidos de la columna 'alcohol'.

```
plt.bar( datos_mod_white['quality'],datos_mod_white['alcohol'], color
='linen', width = 0.4)
plt.xlabel("Calificacion calidad")
```

```
plt.ylabel("Grados de alcohol")
plt.title("Calidad y grados de alcohol")
plt.show()
```

Ilustración 3 Gráfico de barras Calidad - Alcohol vino blanco

3.3 PREPROCESAMIENTO DE LA INFORMACIÓN

Para el preprocesamiento de los datos antes empezar con la construcción del dataset, ser realizaron los siguientes tratamientos a los datos del vino blanco, que fue el que escogimos:

- Se eliminaron registros repetidos.
- Se creo un conjunto de datos adicional para el bono, en el que se unieron los vinos blancos y rojos creando una dummy, eliminando datos atípicos y registros repetidos.

Ilustración 4 Datos Arreglados dataset bono Atributos de los vinos

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality	RedWine
0	7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.99780	3.51	0.56	9.4	5	1
2	7.8	0.76	0.04	2.3	0.092	15.0	54.0	0.99700	3.26	0.65	9.8	5	1
4	7.4	0.66	0.00	1.8	0.075	13.0	40.0	0.99780	3.51	0.56	9.4	5	1
5	7.9	0.60	0.06	1.6	0.069	15.0	59.0	0.99640	3.30	0.46	9.4	5	1
6	7.3	0.65	0.00	1.2	0.065	15.0	21.0	0.99460	3.39	0.47	10.0	7	1
...	...	...	...	...	...	...	...	...	...	...	...	...	...
5315	6.2	0.21	0.29	1.6	0.039	24.0	92.0	0.99114	3.27	0.50	11.2	6	0
5316	6.6	0.32	0.36	8.0	0.047	57.0	168.0	0.99490	3.15	0.46	9.6	5	0
5317	6.5	0.24	0.19	1.2	0.041	30.0	111.0	0.99254	2.99	0.46	9.4	6	0
5318	5.5	0.29	0.30	1.1	0.022	20.0	110.0	0.98869	3.34	0.38	12.8	7	0
5319	6.0	0.21	0.38	0.8	0.020	22.0	98.0	0.98941	3.26	0.32	11.8	6	0

4538 rows × 13 columns

- También se empleó una regresión lineal para poder identificar las variables más relevantes a la hora de construir el modelo de las redes neuronales, lo cual también incluye su entrenamiento. En la imagen se puede observar los resultados de la regresión y con ellos su significancia estadística. Para el vino blanco se decide no considerar las variables “chlorides” y “total sulfur dioxide”.

Ilustración 5 Modelo de Regresión lineal para vinos blancos

OLS Regression Results

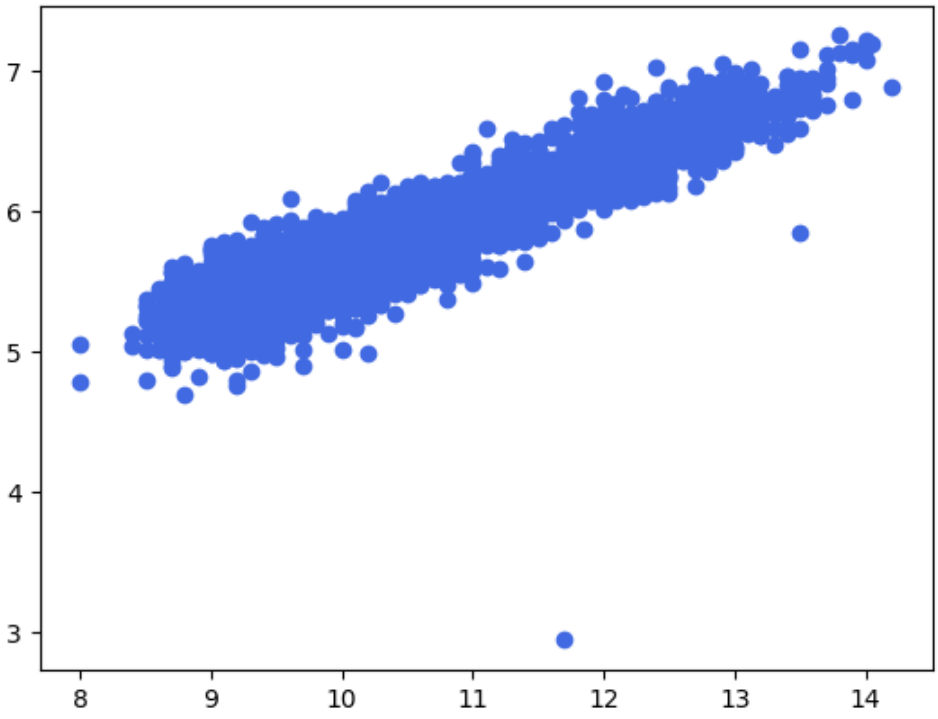
Dep. Variable:	quality	R-squared:	0.253
Model:	OLS	Adj. R-squared:	0.251
Method:	Least Squares	F-statistic:	167.2
Date:	Tue, 21 Nov 2023	Prob (F-statistic):	1.25e-243
Time:	23:11:40	Log-Likelihood:	-4583.9
No. Observations:	3961	AIC:	9186.
Df Residuals:	3952	BIC:	9242.
Df Model:	8		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	195.3255	20.059	9.738	0.000	155.999	234.652
fixed acidity	0.1035	0.023	4.577	0.000	0.059	0.148
residual sugar	0.0906	0.008	11.074	0.000	0.075	0.107
chlorides	-0.9404	0.583	-1.614	0.107	-2.082	0.202
total sulfur dioxide	0.0003	0.000	0.924	0.355	-0.000	0.001
density	-196.9173	20.348	-9.678	0.000	-236.811	-157.024
pH	1.0407	0.117	8.930	0.000	0.812	1.269
sulphates	0.7433	0.114	6.506	0.000	0.519	0.967
alcohol	0.1219	0.026	4.696	0.000	0.071	0.173

Omnibus:	91.811	Durbin-Watson:	1.773
Prob(Omnibus):	0.000	Jarque-Bera (JB):	197.839
Skew:	-0.085	Prob(JB):	1.10e-43
Kurtosis:	4.082	Cond. No.	3.37e+05

Notes:

Ilustración 6 Gráfico Modelo de regresión lineal



- Por último, Se eliminaron los datos atípicos del conjunto de vinos blancos para las variables de interés, utilizando los rangos intercuartílicos de las distribuciones.

3.4 CONSTRUCCIÓN DEL DATASET

Para realizar el análisis de clasificación se sugiere realizar un particionamiento entre dos conjuntos (entrenamiento, pruebas)

```
feature_set = ['fixed acidity','residual
sugar','alcohol','density','pH','sulphates']
features=datos_mod_white[feature_set]
target = datos_mod_white.quality
from sklearn.model_selection import train_test_split
feature_train,feature_test,target_train, target_test =
train_test_split(features,target,test_size=0.3,random_state=1)
```

3.4.1 ¿Qué proporción de conjunto de entrenamiento y de pruebas?

La proporción de conjuntos de entrenamiento y pruebas que se ha utilizado es del 70% para entrenamiento y del 30% para pruebas. Esta proporción se determinó mediante el parámetro **test_size=0.3** en la función **train_test_split**.

3.4.2 ¿Cómo cambia el modelo si existen cambios en las proporciones de los dos conjuntos de trabajo?

1. **Mayor Conjunto de Entrenamiento:**
    - **Positivo:** Más datos de entrenamiento pueden permitir que el modelo capture patrones más complejos y generalice mejor.
    - **Negativo:** Si el conjunto de pruebas se vuelve muy pequeño, la evaluación del modelo puede no ser tan precisa, lo que puede llevar a una estimación engañosa de su rendimiento en datos nuevos.
2. **Mayor Conjunto de Pruebas:**
    - **Positivo:** Proporciona una evaluación más confiable del rendimiento del modelo en datos no vistos.
    - **Negativo:** Menos datos de entrenamiento pueden resultar en un modelo menos preciso, ya que tiene menos información para aprender patrones.
3. **Cambios Extremos en las Proporciones:**
    - Proporciones extremas, como 90-10 o 95-5, podrían llevar a modelos que no generalizan bien debido a la falta de datos de prueba para evaluar el rendimiento.

3.5 ELABORACIÓN DEL MODELO

3.5.1 Perceptrón:

Este código implementa un Perceptrón para regresión utilizando **TensorFlow**. Aquí está el análisis paso a paso del código:

1. **Preparación de datos:**
    - Se divide el conjunto de datos en características (X) y la variable objetivo (y).

- Luego se dividen los datos en conjuntos de entrenamiento y prueba utilizando train_test_split de scikit-learn.
- Los datos se normalizan utilizando StandardScaler de scikit-learn para asegurar que todas las características estén en la misma escala.

2. **Definición de la clase Perceptron:**
   - Se define una clase llamada Perceptron que es una subclase de tf.keras.Model. Esta clase representa el modelo del Perceptrón para regresión.
   - En el método __init__, se inicializa la capa Dense de TensorFlow con un único nodo (units=1), que representa la salida del Perceptrón. La cantidad de entradas se define por input_shape=(num_inputs,), donde num_inputs es la cantidad de características.
   - El método call define cómo se propagarán los datos a través del Perceptrón. En este caso, simplemente pasa los datos a través de la capa densa.
3. **Creación del modelo:**
   - Se instancia el modelo del Perceptrón (Perceptron) con la cantidad correcta de entradas (num_inputs).
4. **Compilación y entrenamiento del modelo:**
   - Se compila el modelo utilizando el optimizador sgd (descenso de gradiente estocástico) y la función de pérdida mse (error cuadrático medio) para minimizar el error en la regresión.
   - Se entrena el modelo (fit) con los datos de entrenamiento durante 100 épocas.
5. **Predicciones y evaluación:**
   - Se realizan predicciones utilizando el modelo entrenado en el conjunto de prueba.
   - Se calcula el Error Cuadrático Medio (MSE) usando mean_squared_error de scikit-learn entre las predicciones y los valores reales del conjunto de prueba.
   - Se calcula el Coeficiente de Determinación (R^2) usando r2_score de scikit-learn entre las predicciones y los valores reales del conjunto de prueba. Este coeficiente indica qué tan bien el modelo se ajusta a los datos reales.

3.5.2 Red Neuronal de 1 capa:

1. **Preparación de datos:**
    - Se divide el conjunto de datos en características (X) y la variable objetivo (y).
    - Se normalizan los datos usando StandardScaler de scikit-learn para asegurar que todas las características estén en la misma escala.
    - Si el problema es de clasificación multiclase, la variable objetivo se codifica utilizando to_categorical de Keras.
2. **Definición del modelo de red neuronal:**
    - Se crea un modelo secuencial (Sequential) de Keras.
    - Se agrega una capa oculta con una función de activación sigmoide (sigmoid) utilizando Dense. La cantidad de neuronas en esta capa es igual a la cantidad de características en los datos de entrada.
    - La capa de salida utiliza la función de activación softmax para problemas de clasificación multiclase.
3. **Compilación del modelo:**
    - Se compila el modelo utilizando el optimizador adam y la función de pérdida categorical_crossentropy, comúnmente utilizada para problemas de clasificación multiclase.
    - Se especifica que se desea medir la precisión (accuracy) durante el entrenamiento.
4. **Entrenamiento del modelo:**
    - Se entrena el modelo con los datos de entrenamiento (X_train e y_train) durante 200 épocas y un tamaño de lote de 10.
5. **Evaluación del modelo:**
    - Se evalúa el rendimiento del modelo en el conjunto de prueba (X_test e y_test) y se muestra la precisión obtenida en la clasificación.

3.5.1 Red Neuronal de 2 capas:

1. **Preparación de datos:**
    - Se separan las características (X) y la variable objetivo (y) del conjunto de datos.

- Los datos se escalan utilizando StandardScaler para normalizar las características.
- Si es un problema de clasificación multiclase, se codifica la variable objetivo (y) utilizando to_categorical de Keras para convertirla en una matriz de variables binarias.

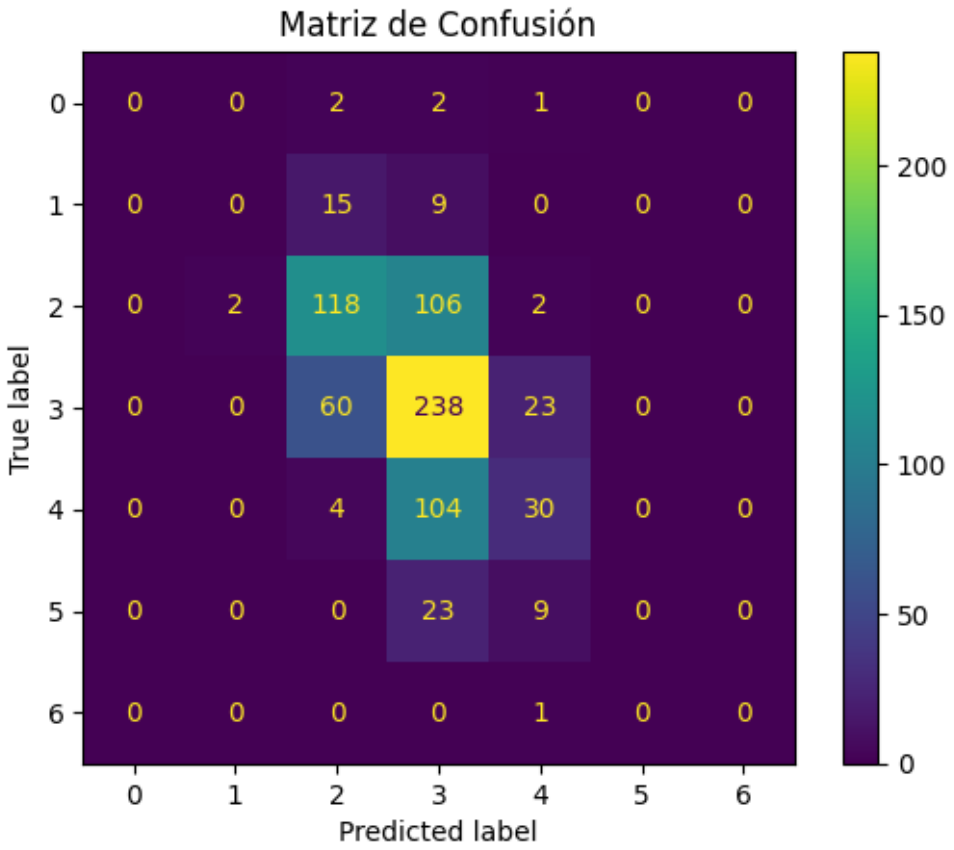
2. **División de datos:**
   - Se dividen los datos en conjuntos de entrenamiento y prueba usando train_test_split de scikit-learn.
3. **Definición del modelo de red neuronal:**
   - Se crea un modelo secuencial (Sequential) de Keras.
   - Se agregan dos capas ocultas, cada una con dos neuronas y función de activación sigmoide, seguidas por una capa de salida con activación softmax para la clasificación multiclase.
4. **Compilación del modelo:**
   - Se compila el modelo utilizando el optimizador adam y la función de pérdida categorical_crossentropy para problemas de clasificación multiclase. Se utilizan métricas de precisión (accuracy) para evaluar el rendimiento durante el entrenamiento.
5. **Entrenamiento y evaluación del modelo:**
   - El modelo se entrena con los datos de entrenamiento durante 200 épocas y un tamaño de lote de 10.
   - Se evalúa el modelo utilizando los datos de prueba para calcular la pérdida y la precisión del modelo.

3.6 ANÁLISIS DE RESULTADOS

3.6.1 **Perceptrón**:

- **Accuracy**: 0.52. Este modelo clasifica correctamente alrededor del 52% de las muestras.
- **Precision**: 0.48. La precisión baja indica que hay una cantidad considerable de falsos positivos.
- **Recall**: 0.52. Un recall de 0.52 sugiere que el modelo es moderadamente capaz de identificar todas las clases relevantes.
- **F1 Score**: 0.48. El F1 score es una medida armónica entre precisión y recall, y un valor más bajo sugiere que el modelo no equilibra bien estas métricas.

Ilustración 7 Matriz de Confusión Perceptrón



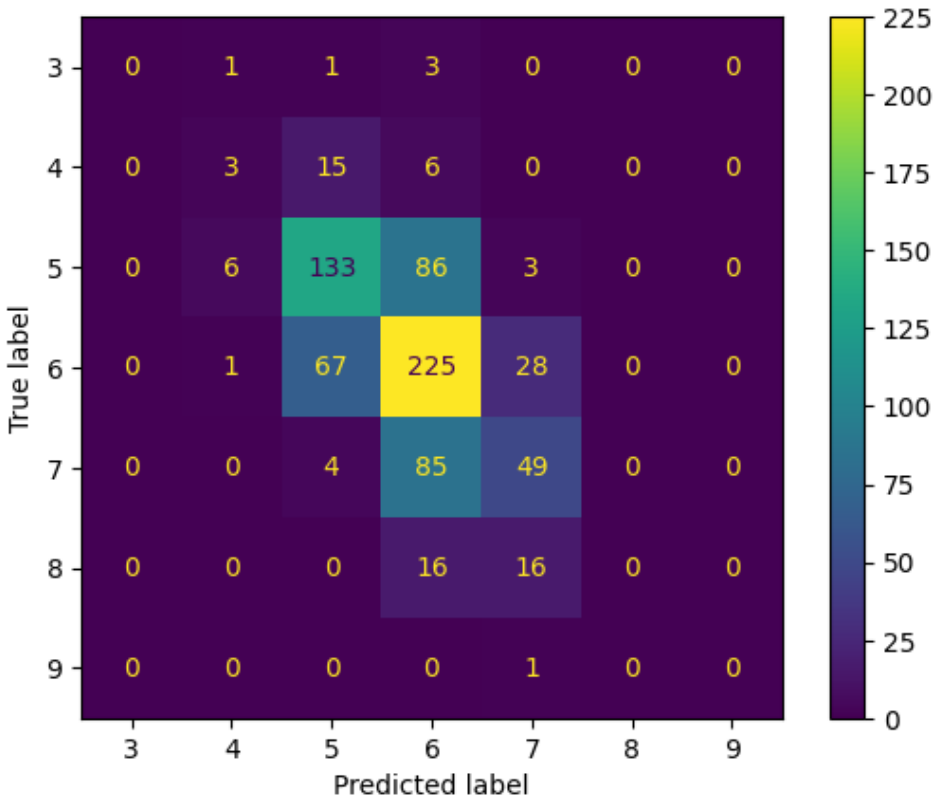
- La matriz de confusión muestra una distribución de predicciones acertadas e incorrectas que no parece enfocarse en una clase de calidad específica. Las clases intermedias son más comunes en los datos de vino, donde las clases de calidad más alta y baja tienen menos representación. Los bloques de color más oscuro, donde las cantidades de verdaderos positivos deberían ser mayores, muestran que el modelo parece tener problemas particulares con ciertas clases.

3.6.2 **Red Neuronal con una Capa Oculta**:

2. **Accuracy**: 0.54. Una leve mejora en la capacidad del modelo para clasificar correctamente las muestras.

3. **Precision**: 0.51. Indica una ligera mejora en la reducción de falsos positivos en comparación con el perceptrón.
4. **Recall**: 0.55. Muestra que el modelo puede identificar correctamente las clases relevantes ligeramente mejor que el perceptrón.
5. **F1 Score**: 0.52. Un F1 score ligeramente mejorado refleja una pequeña mejora en el balance entre precisión y recall.

Ilustración 8 Matriz de Confusión Red Neuronal 1 capa

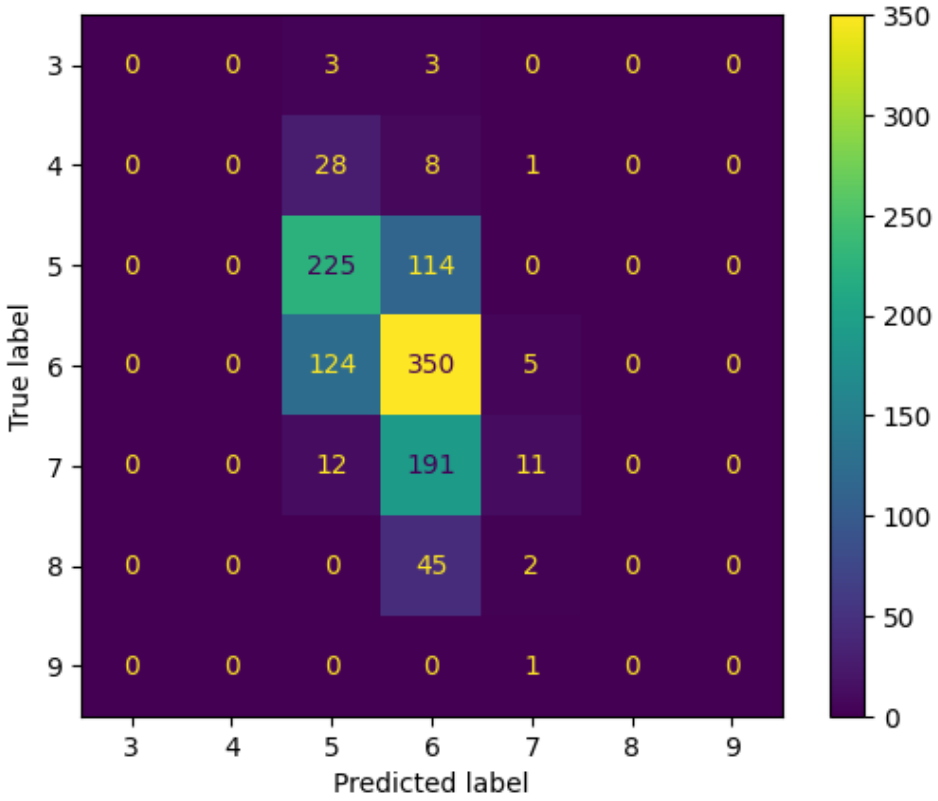


- La presencia de números ligeramente más altos en la diagonal principal de la matriz de confusión muestra que la red neuronal con una sola capa oculta ha mejorado su capacidad para distinguir entre algunas de las clases de calidad. Sin embargo, la confusión entre clases adyacentes sigue existiendo, lo que sugiere que la red puede haber captado alguna relación no lineal pero no es suficiente para una clasificación clara.

3.6.3 **Red Neuronal con Dos Capas Ocultas**:

- **Accuracy**: 0.52. Similar al perceptrón, lo que sugiere que la adición de una capa oculta no ha mejorado la capacidad de clasificación general.
- **Precision**: 0.49. Disminuye en comparación con el modelo de una sola capa oculta, lo que podría indicar un ajuste excesivo o una complejidad innecesaria para el modelo.
- **Recall**: 0.52. Igual que el perceptrón, indica una capacidad moderada de detectar las clases positivas.
- **F1 Score**: 0.45. No hay mejora en el balance entre precisión y recall en comparación con el perceptrón.

Ilustración 9 Matriz de Confusión Red Neuronal 2 capas

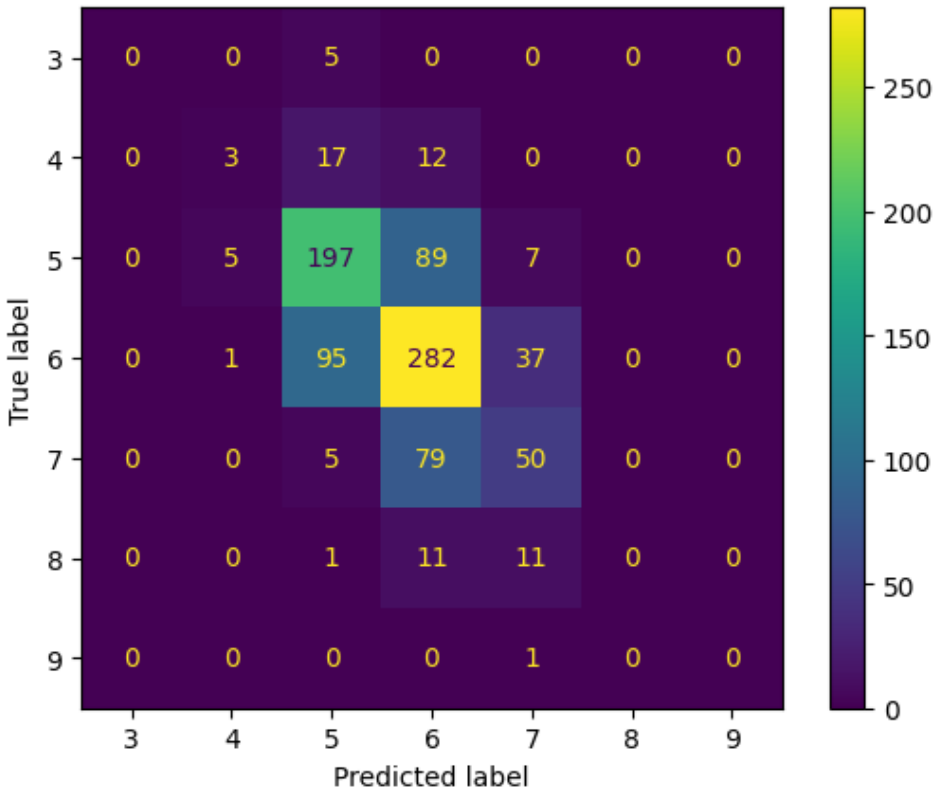


- La matriz de confusión del modelo con dos capas ocultas no muestra una mejora significativa en la clasificación, a pesar de tener una arquitectura más compleja. Esto podría

deberse a que el modelo está sobre ajustado o simplemente porque, dada la naturaleza de los datos, la complejidad adicional no es beneficiosa. Los errores entre clases adyacentes siguen siendo un problema, y las predicciones correctas en la diagonal principal son comparables a las del perceptrón.

3.6.4 **Modelo de dos capas para ambos tipos de vino (Bono)**

- **Accuracy**: 0.59. Superior al perceptrón y las otras redes neuronales. El modelo acierta el 58% de las predicciones.
- **Precision**: 0.56. Aumenta respecto a los demás modelos. Esta métrica no es fundamental para el caso particular de clasificacion de calidad
- **Recall**: 0.59. Mayor a los demás modelos.
- **F1 Score**: 0.57. Hay mejora en el balance entre precisión y recall en comparación con los demás modelos.



- Utilizar ambos tipos de vino mejora la predicción de nivel de calidad respecto a los otros modelos. Puede estar

explicado por una mayor cantidad de datos para el entrenamiento del modelo. También se puede considerar que al utilizar ambos tipos de vino se estiman pesos menos sesgados hacia un tipo de vino y por tanto sean más precisos en general,

3.6.4 **Análisis Comparativo en relación con la variable quality**:

- Los modelos no muestran una diferencia significativa en **accuracy**, lo que sugiere que la capacidad de predecir correctamente la calidad del vino no varía mucho entre un modelo simple y uno con más capas.
- La **precision** más alta se observa en la red neuronal con una sola capa oculta, lo que puede indicar que esta configuración es ligeramente más efectiva para predecir las clases correctas sin incluir tantos falsos positivos.
- El **recall** es similar en los tres modelos, lo que indica que la capacidad de los modelos para encontrar todas las instancias positivas es comparable.
- El **F1 Score** es levemente mejor en la red neuronal con una capa oculta. Este es un indicador importante al tratar con un conjunto de datos donde el equilibrio entre precisión y recall es crítico.

En resumen, la red neuronal con una sola capa oculta parece proporcionar un equilibrio de métricas ligeramente mejor para la clasificación de la calidad del vino, aunque esta mejora es marginal. Ninguno de los modelos se destaca en todos los aspectos, lo que podría indicar que la caracterización de la calidad del vino podría beneficiarse de un enfoque de modelado más sofisticado, como el uso de más datos, características adicionales o una arquitectura de red más compleja. Además, podría ser necesario utilizar un enfoque más matizado para modelar esta variable porque la calidad del vino es subjetiva y puede verse afectada por muchos factores que no se pueden capturar en los datos.

CONCLUSIONES

Conclusiones del proyecto, enfocadas en los objetivos iniciales:

1. **Exploración y Análisis del Dataset:** El análisis detallado de las características químicas del vino y la identificación precisa de la variable objetivo fueron fundamentales para comprender la naturaleza y la

complejidad de los datos. Esta exploración exhaustiva proporcionó una base sólida para la selección adecuada de atributos y la comprensión del dominio del problema.

2. **Preprocesamiento de Datos:** El proceso exhaustivo de preprocesamiento desempeñó un papel crítico en la preparación de datos para el modelado. La unión de datasets, la codificación precisa de variables categóricas y la gestión efectiva de datos atípicos permitieron obtener un conjunto de datos limpio y optimizado para la construcción de modelos.
3. **Diseño y Evaluación de Modelos de Redes Neuronales:** La implementación y comparación de tres arquitecturas de redes neuronales distintas proporcionaron una visión profunda de su rendimiento en la clasificación de la calidad del vino. Aunque se observaron métricas de desempeño similares entre los modelos, se identificó una ligera superioridad en la precisión de la Red Neuronal con una sola capa oculta.
4. **Impacto de las Proporciones de Conjuntos de Entrenamiento y Pruebas:** La experimentación con diferentes proporciones de datos de entrenamiento y prueba permitió comprender cómo estas proporciones influyen en el rendimiento de los modelos. Si bien no se observaron mejoras drásticas, se evidenció la importancia de mantener un equilibrio adecuado en la distribución de datos para evitar sobreajuste o subajuste.

En resumen, este proyecto destaca la importancia del análisis exhaustivo y el procesamiento adecuado de datos para la construcción efectiva de modelos de redes neuronales. Aunque se lograron resultados aceptables en la clasificación de la calidad del vino, se reconoce la necesidad de investigaciones adicionales y enfoques innovadores para lograr mejoras significativas en la precisión y generalización de los modelos.