Proyecto final
Data Science I

CALIDAD DEL VINO

Proyecto Florencia Peluffo
Tutora Abril Noguera
Docente Jorge Ruiz
Coderhouse- Camada 61140

Contenido

Contexto Comercial

WineTech Innovations es una empresa tecnológica especializada en soluciones avanzadas para la industria vitivinícola. Con el auge del mercado global del vino y la creciente demanda de productos de alta calidad, ***WineTech Innovations*** busca implementar herramientas de análisis de datos y machine learning para mejorar la producción y la comercialización de vinos. La empresa está colaborando con diversas bodegas, tanto locales como internacionales, para optimizar los procesos de producción y asegurar que los vinos comercializados cumplan con los estándares de calidad esperados por los consumidores y críticos.

▢ Proyecto: Optimización de la Calidad del Vino

Para este propósito, ***WineTech Innovations*** ha iniciado un proyecto piloto que se centra en el análisis de las propiedades fisicoquímicas del vino y su relación con la calidad final del producto. Utilizando datos obtenidos de diversas bodegas, el proyecto tiene como objetivo desarrollar un modelo predictivo que permita:

1- **Mejorar la Consistencia del Producto:** Garantizar que cada botella de vino producida mantenga un nivel de calidad consistente, cumpliendo con las expectativas del consumidor y reduciendo la variabilidad entre lotes.

2- **Optimización del Proceso de Producción:** Identificar los parámetros clave que influyen en la calidad del vino, permitiendo a los enólogos ajustar los procesos de producción para mejorar la calidad de manera más eficiente.

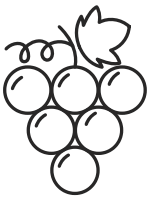
3- **Reducción de Costos:** Minimizar el desperdicio de recursos y materias primas mediante la identificación temprana de lotes que no cumplan con los estándares de calidad.

4- **Desarrollo de Nuevos Productos:** Utilizar los insights obtenidos para experimentar con nuevas combinaciones de propiedades fisicoquímicas, desarrollando vinos con perfiles únicos que puedan captar nuevos segmentos de mercado.

▢ Beneficios Esperados

Al implementar esta solución, **WineTech Innovations** espera no solo mejorar la calidad de los vinos producidos por sus clientes, sino también posicionarse como líder en innovación tecnológica dentro de la industria vitivinícola. Los beneficios clave incluyen:

- **Aumento de la Satisfacción del Cliente**: Producir vinos de alta calidad de manera consistente aumentará la satisfacción del cliente y fortalecerá la lealtad a la marca.
- **Ventaja Competitiva**: Las bodegas que utilicen esta tecnología estarán mejor posicionadas frente a la competencia, destacándose por su capacidad para producir vinos de calidad superior.
- **Sostenibilidad**: La optimización del uso de recursos contribuirá a prácticas de producción más sostenibles, reduciendo el impacto ambiental de la viticultura.



Abstract

Este proyecto se enfoca en analizar cómo las propiedades fisicoquímicas del vino influyen en su calidad. Utilizando un dataset descargado de Kaggle, se examinan las relaciones entre variables como acidez fija, acidez volátil, contenido de azúcar residual, cloruros, entre otras, y la calidad del vino, la cual se clasificará en tres categorías: malo, regular y bueno. La hipótesis principal es que ciertas propiedades fisicoquímicas están significativamente correlacionadas con la calidad del vino. A través de visualizaciones y análisis numéricos, se buscará identificar patrones que puedan ser utilizados para desarrollar un modelo predictivo capaz de estimar la calidad del vino. Para explorar y presentar los hallazgos, se emplearán herramientas de visualización como Seaborn y Matplotlib, facilitando así una interpretación clara y comprensible de los datos.

Hipótesis de Trabajo

La hipótesis central de este proyecto es que existe una relación entre la calidad del vino y ciertas propiedades fisicoquímicas. Específicamente, se investigará si las variables como la acidez fija, la acidez volátil, el contenido de azúcar residual, los cloruros y otros factores influyen en la calidad del vino. Además, se analizará cómo se distribuyen estas propiedades fisicoquímicas en función de la calidad del vino.

- ¿Qué propiedades fisicoquímicas están más fuertemente correlacionadas con la calidad del vino?
- ¿Existe una combinación específica de variables que prediga mejor la calidad del vino?
- ¿Cómo se distribuyen las diferentes calidades del vino en función de sus propiedades fisicoquímicas?

Objetivo

El objetivo principal de este proyecto es comprender en profundidad los factores que influyen en la calidad del vino mediante el análisis de propiedades fisicoquímicas. Además, se busca identificar patrones significativos en estos datos con el fin de desarrollar un modelo predictivo que pueda estimar la calidad del vino en función de estas características.

Considerando las características del dataframe, así como la naturaleza y distribución de sus variables, se ha determinado que la aplicación de un modelo supervisado de clasificación es la metodología de Machine Learning más adecuada.

La variable objetivo en este análisis es la calidad del vino, la cual se clasifica en tres categorías: malo, regular y bueno, basándose en una escala del 1 al 10. Este enfoque permitirá predecir la categoría de calidad de los vinos a partir de los patrones identificados en los datos de entrenamiento, optimizando así la precisión y la eficiencia del análisis predictivo.

Dataset

La base de datos utilizada en el proyecto fue descargada desde **Kaggle**.

El dataset seleccionado posee 12 columnas y 1599 filas y se compone de las siguientes variables (traducidas al español, ya que el idioma original es el inglés) :

Variables de entrada (basadas en pruebas fisicoquímicas):

1 - Acidez fija

2 - Acidez volátil

3 - Ácido cítrico

4 - Azúcar residual

5 - Cloruros

6 - Dióxido de azufre libre

7 - Dióxido de azufre total

8 - Densidad

9 - PH

10 - Sulfatos

11 - Alcohol

Variable de salida (basada en datos sensoriales):

12 - Calidad (puntuación entre 0 y 10)

Análisis Exploratorio de Datos (EDA)

A partir del método .info(), se identificó que no existen valores nulos en los datos, por lo que no fue necesario realizar tareas de limpieza ni de relleno. Además, se observó que todas las columnas, excepto la de "quality" que contiene valores de tipo entero, presentan datos de tipo flotante.

```
[ ] # Detección de valores nulos
    df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1599 entries, 0 to 1598
Data columns (total 12 columns):
 #   Column                Non-Null Count  Dtype  
---  ------                --------------  -----  
 0   fixed acidity         1599 non-null   float64
 1   volatile acidity      1599 non-null   float64
 2   citric acid           1599 non-null   float64
 3   residual sugar        1599 non-null   float64
 4   chlorides             1599 non-null   float64
 5   free sulfur dioxide   1599 non-null   float64
 6   total sulfur dioxide  1599 non-null   float64
 7   density               1599 non-null   float64
 8   pH                    1599 non-null   float64
 9   sulphates             1599 non-null   float64
 10  alcohol               1599 non-null   float64
 11  quality               1599 non-null   int64  
dtypes: float64(11), int64(1)
memory usage: 150.0 KB
```

Variables

El conjunto de datos seleccionado contiene exclusivamente variables cuantitativas. Entre ellas, la variable objetivo o variable de respuesta que es "quality" (calidad). A pesar de que la calidad del vino se mide en una escala ordinal de 1 a 10, para facilitar su análisis en modelos de clasificación supervisada, se considera conveniente transformarla en una variable categórica.

```
[ ] # Convertir la variable de calidad en categórica
    bins = [0, 4, 6, 10]
    labels = ['malo', 'regular', 'bueno']
    df['calidad'] = pd.cut(df['quality'], bins=bins, labels=labels, include_lowest=True)
```

```
[ ] print(df[['quality', 'calidad']].head(10))
```

```
   quality  calidad
0        5  regular
1        5  regular
2        5  regular
3        6  regular
4        5  regular
5        5  regular
6        5  regular
7        7    bueno
8        7    bueno
9        5  regular
```

Medidas Descriptivas

En el marco del análisis realizado, se tomó la decisión de crear un diccionario denominado statistics con el propósito de almacenar las medidas descriptivas calculadas para cada variable del conjunto de datos. La implementación de esta estructura de datos responde a la necesidad de organizar y gestionar de manera eficiente la información estadística obtenida, facilitando su posterior consulta y análisis.

El diccionario statistics presenta una estructura clave-valor, donde cada clave corresponde al nombre de la variable y el valor asociado es un subdiccionario que alberga las medidas de tendencia central calculadas para dicha variable. Específicamente, este subdiccionario contiene las siguientes entradas:

- Media: Representa el valor promedio de todos los datos en la variable correspondiente.
- Mediana: Define el valor que divide al conjunto de datos en dos mitades de igual tamaño, de manera que la mitad de los valores son inferiores a la mediana y la otra mitad superiores.
- Moda: Se refiere al valor que presenta mayor frecuencia de aparición dentro del conjunto de datos.

```python
# Creación de un diccionario para almacenar las estadísticas
statistics = {}

# Calcular media, mediana y moda para cada variable
for column in df.columns:
    mean_value = np.mean(df[column])
    median_value = np.median(df[column])
    mode_value = df[column].mode()[0] if not df[column].mode().empty else np.nan  # Verificar si la moda no está vacía

    statistics[column] = {
        'Media': mean_value,
        'Mediana': median_value,
        'Moda': mode_value
    }

# Imprimir las estadísticas
for column, stats in statistics.items():
    print(f"{column}:")
    print(f"  Media: {stats['Media']}")
    print(f"  Mediana: {stats['Mediana']}")
    print(f"  Moda: {stats['Moda']}\n")
```

Visualizaciones

Se importan las bibliotecas necesarias:

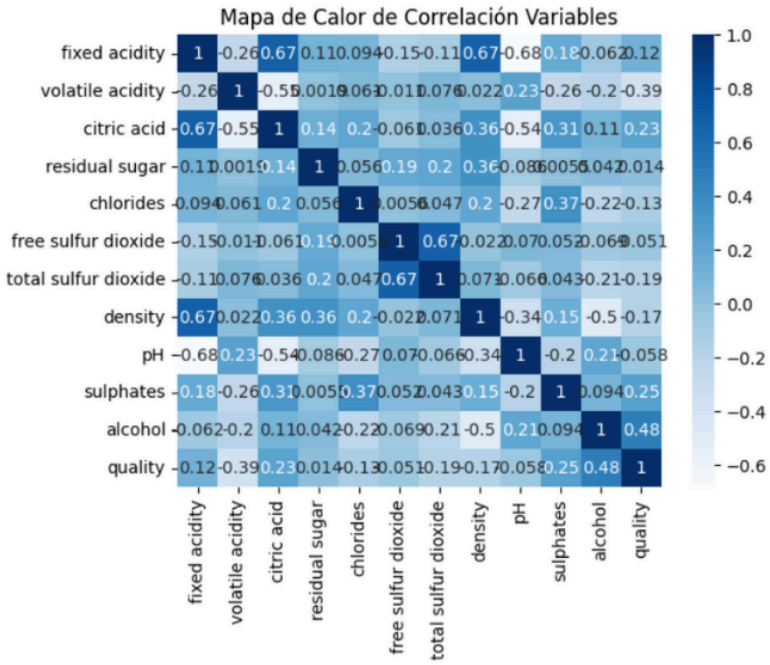
- ‘pandas’ para la manipulación y análisis de datos.
- ‘numpy’ para funciones matemáticas y operaciones con vectores y matrices.
- ‘sklearn’ para algoritmos de aprendizaje automático.
- ‘seaborn’ y ‘matplotlib.pyplot’ para visualizaciones.

```
#Importación de paquetes
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split, GridSearchCV
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report, confusion_matrix
from sklearn.preprocessing import StandardScaler
```

Seaborn

Mapa de Calor de Correlación Variables

```
# Heatmap: Correlación Variables
correlation = df.corr()
sns.heatmap(correlation, annot=True, cmap="Blues")
plt.title("Mapa de Calor de Correlación Variables")
plt.show()
```

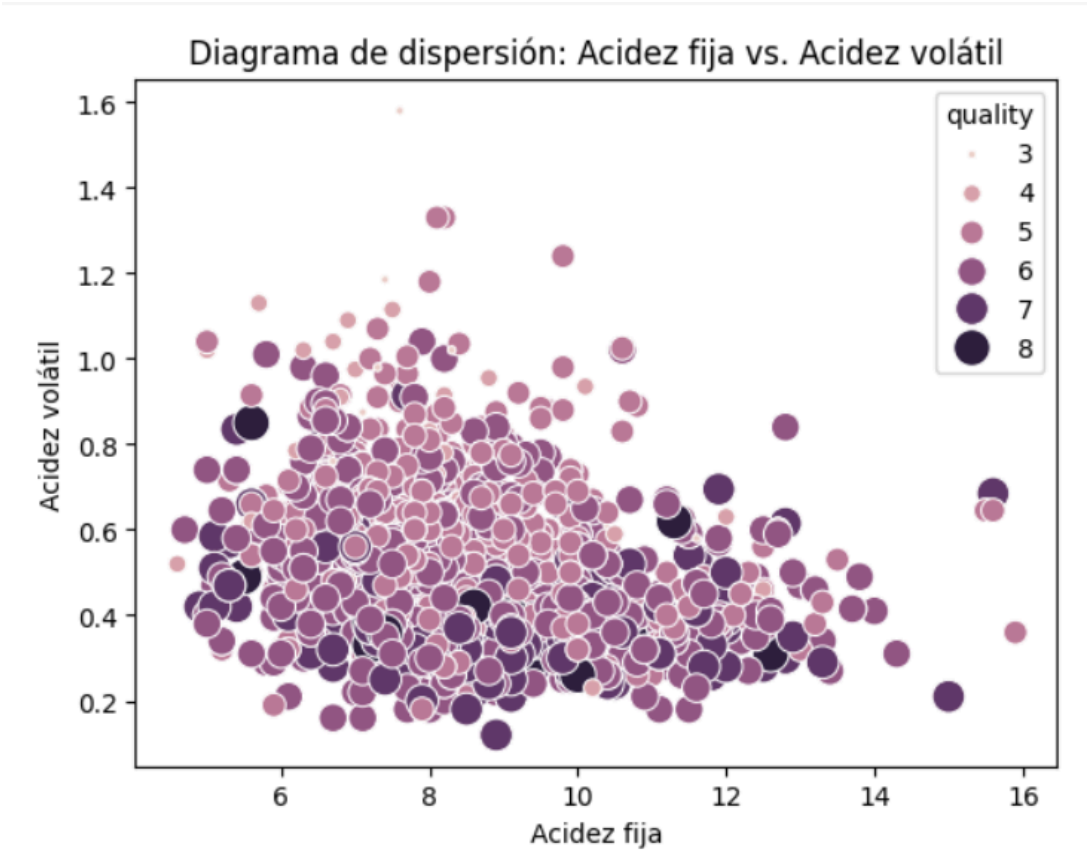


- **Acidez fija (fixed acidity):** Esta variable tiene una correlación positiva con la calidad del vino. A medida que aumenta la acidez fija, es más probable que la calidad del vino también aumente.
- **Acidez volátil** (volatile acidity): La acidez volátil muestra una correlación negativa con la calidad del vino, es decir, tienen una relación inversamente proporcional. Cuanto mayor sea la acidez volátil, es más probable que la calidad del vino disminuya. Esto puede deberse a que niveles elevados de acidez volátil pueden afectar negativamente el sabor y la estabilidad del vino.
- **Ácido cítrico** (citric acid): El ácido cítrico tiene una correlación positiva con la calidad del vino. Un mayor contenido de ácido cítrico tiende a estar asociado con una mejor calidad.
- **Azúcar residual** (residual sugar): No se identifica una correlación fuerte entre el azúcar residual y la calidad del vino.
- **Cloruros** (chlorides): Los cloruros tienen una correlación negativa con la calidad del vino. Cuanto mayor sea la concentración de cloruros, es más probable que la calidad disminuya.
- **Dióxido de azufre libre** (free sulfur dioxide): No muestra una correlación significativa con la calidad del vino.
- **Dióxido de azufre total** (total sulfur dioxide): No hay una correlación fuerte con la calidad.
- **Densidad (density)**: La densidad tiene una correlación negativa con la calidad. Valores más bajos de densidad suelen estar asociados con vinos de mejor calidad.
- **PH:** El PH tampoco muestra una correlación significativa con la calidad del vino.
- **Sulfatos** (sulphates): Los sulfatos tienen una correlación positiva moderada con la calidad. Mayor contenido de sulfatos se relaciona con vinos de mejor calidad.

En síntesis, las variables que se correlacionan positivamente con la calidad del vino son la acidez fija, el ácido cítrico y los sulfatos. Mientras tanto, la acidez volátil, los cloruros y la densidad tienen una relación inversa con la calidad del vino, lo que implica que a medida que aumentan, la calidad tiende a disminuir. En cuanto al azúcar residual, el dióxido de azufre libre, el dióxido de azufre total y el pH, no muestran una correlación significativa con la calidad.

Diagrama de dispersión: Acidez fija vs. Acidez volátil

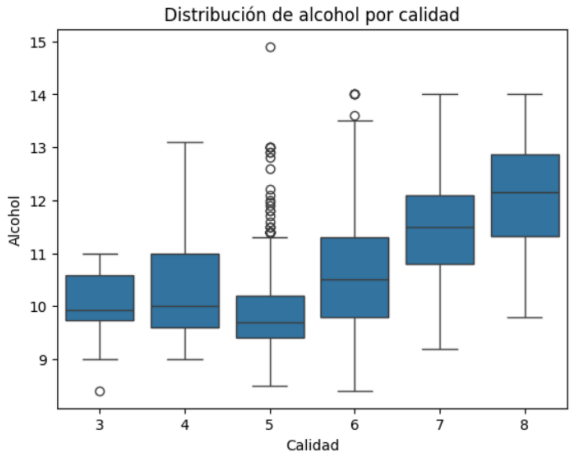
```
# Diagrama de dispersión: Acidez fija vs. Acidez volátil
sns.scatterplot(data=df, x='fixed acidity', y='volatile acidity', hue='quality',size='quality', sizes=(10,200), legend='brief')
plt.xlabel('Acidez fija')
plt.ylabel('Acidez volátil')
plt.title('Diagrama de dispersión: Acidez fija vs. Acidez volátil')
plt.show()
```



El gráfico de dispersión muestra que la acidez fija y la acidez volátil están débilmente correlacionadas positivamente. Los vinos de mayor calidad tienden a tener una acidez fija y volátil más bajas.

Boxplot: Distribución de contenido de alcohol por calidad del vino

```
# Boxplot: Distribución de contenido de alcohol por calidad del vino
sns.boxplot(x='quality', y='alcohol', data=df)
plt.xlabel('Calidad')
plt.ylabel('Alcohol')
plt.title('Distribución de alcohol por calidad')
plt.show()
```



La mediana del alcohol por calidad se encuentra en el centro de la caja, lo que indica que la distribución de los datos es simétrica. El IQR, representado por la altura de la caja, es relativamente pequeño, lo que sugiere que los datos están agrupados alrededor de la mediana.

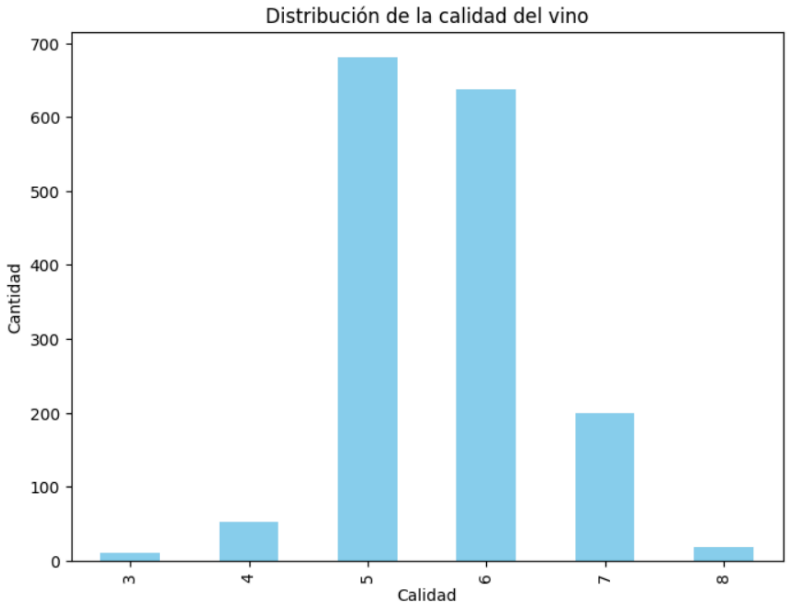
La distribución de los datos parece ser simétrica, ya que la mediana se encuentra en el centro de la caja y los bigotes tienen una longitud similar a ambos lados.

El gráfico de distribución de alcohol por calidad muestra que los datos están distribuidos de manera simétrica y agrupados alrededor de la mediana. No se observan valores atípicos en el conjunto de datos.

Matplotlib

Gráfico de barras: Distribución de la calidad del vino

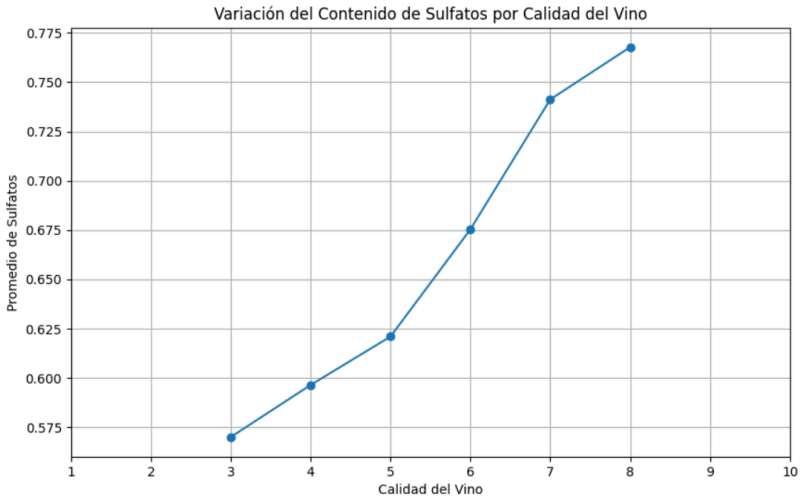
```
# Gráfico de barras: Distribución de la calidad del vino
plt.figure(figsize=(8, 6))
df['quality'].value_counts().sort_index().plot(kind='bar', color='skyblue')
plt.title('Distribución de la calidad del vino')
plt.xlabel('Calidad')
plt.ylabel('Cantidad')
plt.show()
```



El gráfico presenta la distribución de muestras de vino según diferentes puntuaciones de calidad. Se puede observar que la mayoría de las muestras se concentran en las calificaciones de 5 y 6. En contraste, las puntuaciones más bajas (4) y más altas (7 y 8) son menos frecuentes. Esto indica que la calidad del vino tiende a situarse alrededor de las puntuaciones medias.

Gráfico de Líneas: Contenido de Sulfatos por Calidad del Vino

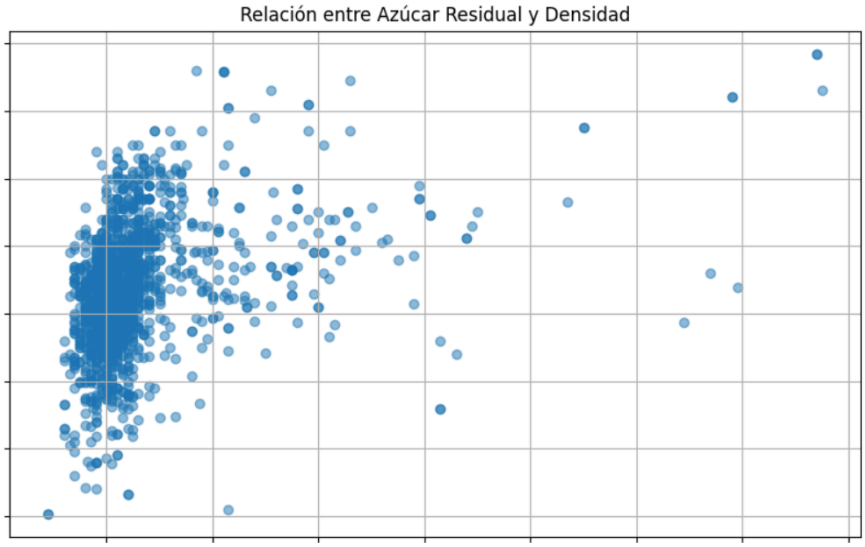
```
#Gráfico de Líneas: Contenido de Sulfatos por Calidad del Vino
plt.figure(figsize=(10, 6))
plt.plot(sulfatos.index, sulfatos.values, marker='o', linestyle='-')
plt.xlabel('Calidad del Vino')
plt.ylabel('Promedio de Sulfatos')
plt.title('Variación del Contenido de Sulfatos por Calidad del Vino')
plt.xticks(range(1, 11))
plt.grid(True)
plt.show()
```



Se identifica una marcada relación positiva entre las variables analizadas. A medida que aumenta la calidad del vino, también aumenta el porcentaje de sulfatos presente en él. Esto sugiere que los vinos de mayor calidad pueden requerir una cantidad específica de sulfatos para lograr su perfil sensorial deseado. Los sulfatos, como componentes químicos, pueden influir en el sabor y la calidad del vino.

Diagrama de Dispersión: Relación entre Azúcar Residual y Densidad

```
#Diagrama de Dispersión: Relación entre Azúcar Residual y Densidad
plt.figure(figsize=(10, 6))
plt.scatter(df['residual sugar'], df['density'], alpha=0.5)
plt.xlabel('Azúcar Residual')
plt.ylabel('Densidad')
plt.title('Relación entre Azúcar Residual y Densidad')
plt.grid(True)
plt.show()
```



El gráfico muestra una concentración densa de puntos alrededor de los valores más bajos de azúcar residual (cerca de 0 en el eje horizontal). La mayoría de estos puntos tienen densidades que oscilan entre aproximadamente 0.992 y 1.002. A medida que aumenta el azúcar residual , los puntos tienden a dispersarse más. Sin embargo, hay una ligera tendencia que sugiere que, a medida que aumenta el azúcar residual, la densidad tiende a disminuir. Esta distribución podría indicar una relación entre el contenido de azúcar residual y la densidad en los vinos, posiblemente influenciada por la cantidad de azúcar no fermentado presente

Machine Learning

Se decidió implementar un modelo supervisado, ya que los mismos son adecuados cuando se dispone de un conjunto de datos etiquetado, es decir, cuando cada instancia del conjunto de datos incluye tanto las características de entrada como la etiqueta o variable objetivo que se desea predecir. En este caso, el conjunto de datos del vino tinto incluye mediciones físico-químicas del vino (características de entrada) y la calidad del vino (que funciona como la variable objetivo).

Los modelos supervisados permiten aprender la relación entre las características de entrada y la variable objetivo, de modo que se pueda predecir la calidad del vino en nuevos datos no vistos. Dado que nuestro objetivo es predecir categorías específicas de calidad (malo, regular, bueno), se considera que un enfoque supervisado es ideal para este problema.

Elección de Algoritmos

En este análisis, se decidió probar dos algoritmos de **clasificación** distintos debido a las características particulares del conjunto de datos. El objetivo era determinar cuál de los algoritmos proporcionaba un mejor rendimiento para la clasificación de la calidad del vino tinto. Los algoritmos seleccionados fueron el **Random Forest** y la **Regresión Logística.**

El conjunto de datos utilizado contiene varias características físico-químicas del vino tinto y una variable de salida que indica la calidad del vino en una escala del 1 al 10. Para facilitar la interpretación y clasificación, se convirtió la variable de calidad en una variable categórica con tres clases:

Malo: Calidad entre 1 y 4.

Regular: Calidad entre 5 y 6.

Bueno: Calidad entre 7 y 10.

Random Forest:

Se eligió este algoritmo debido a su capacidad para manejar conjuntos de datos con muchas características y su robustez frente a datos desbalanceados. El Random Forest es un método de ensamble que crea múltiples árboles de decisión y los combina para obtener una predicción más precisa y generalizable.

Regresión Logística:

La Regresión Logística se seleccionó como un algoritmo de referencia debido a su simplicidad y efectividad en problemas de clasificación binaria y multiclase. Aunque es menos complejo que Random Forest, ofrece un buen punto de comparación para evaluar la necesidad de modelos más sofisticados.

Preparación del modelo

Ambos modelos fueron entrenados y evaluados utilizando el mismo preprocesamiento de datos y conjunto de entrenamiento y prueba, permitiendo una comparación directa de su rendimiento.

1- **Estandarización:** Para asegurar que todas las características contribuyan de manera equitativa al modelo, se utilizó *StandardScaler* para estandarizar las características del conjunto de datos. Este mismo preprocesamiento se aplicó tanto al modelo de Random Forest como al de Regresión Logística.

2- **División del Conjunto de Datos:** El conjunto de datos se dividió en conjuntos de entrenamiento y prueba con una proporción de 80-20 para evaluar el rendimiento del modelo en datos no vistos.

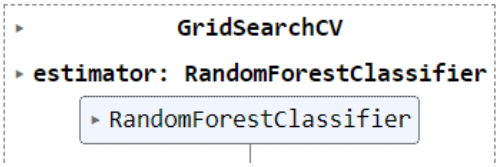
```
#Preparar el modelo. Separar variables predictivas de la variable objetivo
X = df.drop(['quality', 'calidad'], axis=1)
y = df['calidad']
```

```
#Dividir el conjunto de datos para el entrenamiento y testeo
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

```
# Estandarizar los datos
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)
```

Para el modelo de Random Forest, se realizó una búsqueda de hiperparámetros utilizando *GridSearchCV* para optimizar los parámetros del modelo.

```
#Busqueda de hiperparametros
param_grid= {
    'n_estimators': [100, 200, 300],
    'max_depth': [None, 10, 20, 30],
    'min_samples_split':[2,5,10]
}
grid_search= GridSearchCV(rf,param_grid,cv=5, scoring='accuracy')
grid_search.fit(X_train, y_train)
```



Entrenamiento y Evaluación

Entrenamiento: Ambos modelos, el de Random Forest y el de Regresión Logística, fueron entrenados utilizando los datos de entrenamiento preprocesados.

Evaluación: La evaluación del rendimiento de los modelos se llevó a cabo utilizando las métricas de precisión, recall, F1-Score y la matriz de confusión.

Random Forest:

```
#Predecir y evaluar
y_pred= rf.predict(X_test)
print(classification_report(y_test, y_pred))
print(confusion_matrix(y_test, y_pred))
```

```
              precision    recall  f1-score   support

       bueno       0.69      0.57      0.63        47
        malo       0.00      0.00      0.00        11
     regular       0.89      0.95      0.92       262

    accuracy                           0.87       320
   macro avg       0.53      0.51      0.52       320
weighted avg       0.83      0.87      0.85       320
```

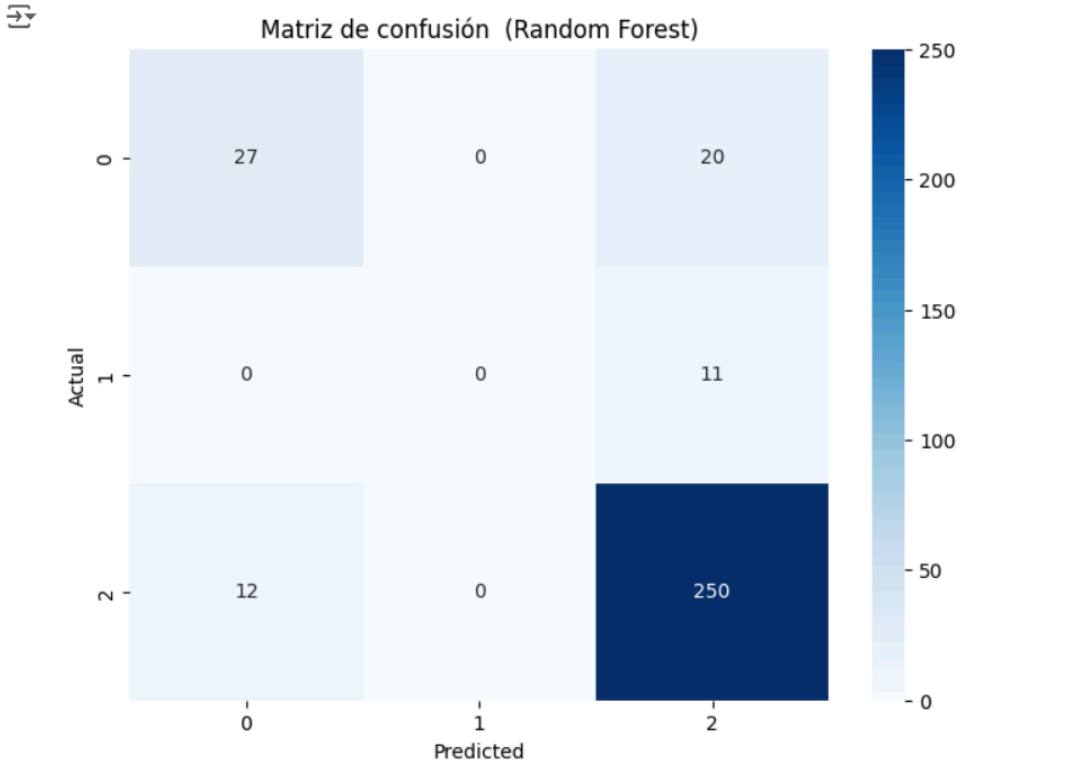
El modelo Random Forest muestra un alto valor de accuracy (0.87), lo que indica que en general, el modelo está prediciendo correctamente en la mayoría de los casos.

La clase "regular" tiene una precisión muy alta (0.89) y un recall (0.95), lo que sugiere que el modelo es muy bueno en identificar y predicir vinos de calidad "regular".

Sin embargo, la clase "malo" tiene un rendimiento muy bajo (0.00 en precisión, recall y F1-Score), lo que indica que el modelo no está capturando bien esta clase.

La clase "bueno" tiene una precisión de 0.69 y un recall de 0.57, lo que sugiere que el modelo tiene un desempeño decente pero no excelente para esta clase.

```python
# Visualización de la matriz de confusión para Random Forest
plt.figure(figsize=(8, 6))
sns.heatmap(confusion_matrix(y_test, y_pred_rf), annot=True, fmt='d', cmap='Blues')
plt.xlabel('Predicted')
plt.ylabel('Actual')
plt.title('Matriz de confusión  (Random Forest)')
plt.show()
```



Regresión Logística:

```python
y_pred_logreg = logreg.predict(X_test)
print("Regresión Logística:")
print(classification_report(y_test, y_pred_logreg))
print(confusion_matrix(y_test, y_pred_logreg))
```

```
Regresión Logística:
              precision    recall  f1-score   support

       bueno       0.57      0.28      0.37        47
        malo       0.00      0.00      0.00        11
     regular       0.85      0.96      0.90       262

    accuracy                           0.83       320
   macro avg       0.47      0.41      0.42       320
weighted avg       0.78      0.83      0.79       320
```

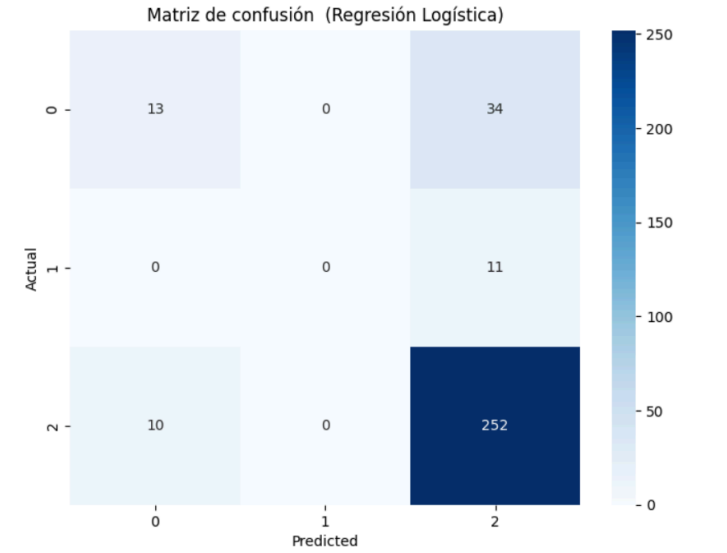
El modelo de Regresión Logística tiene un accuracy de 0.83, ligeramente inferior al modelo Random Forest.

La clase "regular" también tiene un buen rendimiento con precisión (0.85) y recall (0.96), pero es ligeramente inferior al rendimiento de Random Forest para esta clase.

La clase "malo" tampoco tiene un buen rendimiento en el modelo de Regresión Logística (0.00 en precisión, recall y F1-Score).

La clase "bueno" tiene un rendimiento peor que en el modelo Random Forest con precisión de 0.57 y recall de 0.28, indicando que este modelo es menos efectivo para esta clase.

```python
# Visualización de la matriz de confusión
plt.figure(figsize=(8, 6))
sns.heatmap(confusion_matrix(y_test, y_pred_logreg), annot=True, fmt='d', cmap='Blues')
plt.xlabel('Predicted')
plt.ylabel('Actual')
plt.title('Matriz de confusión  (Regresión Logística)')
plt.show()
```



Conclusiones

Este estudio se propuso analizar la relación entre las propiedades fisicoquímicas del vino y su calidad, utilizando modelos de Machine Learning para predecir la calidad del vino en base a estas propiedades. La hipótesis central era que existen correlaciones significativas entre las propiedades del vino y su calidad, y que un modelo predictivo basado en estas propiedades podría ser desarrollado.

Se utilizaron dos conjuntos de datos: uno para entrenamiento y otro para evaluación. Se exploraron las relaciones entre las variables mediante análisis descriptivo y visualizaciones. Se entrenaron dos modelos de Machine Learning, Regresión Logística y Random Forest, para clasificar el vino en tres categorías de calidad: "Malo", "Regular" y "Bueno". Se evaluó el desempeño de los modelos utilizando métricas como precisión, recall, F1-Score y matriz de confusión. Finalmente, se optimizó el modelo Random Forest para mejorar su rendimiento.

Los resultados revelaron que el modelo Random Forest superó al modelo de Regresión Logística en todas las métricas evaluadas, especialmente en la predicción de la clase "Bueno". La optimización del modelo Random Forest condujo a mejoras incrementales en la precisión y el recall. Sin embargo, ambos modelos tuvieron dificultades para predecir la clase "Malo", lo que se atribuye a la baja representación de esta clase en los datos.

Se confirma la hipótesis principal de que existen correlaciones significativas entre las propiedades fisicoquímicas del vino y su calidad. El modelo Random Forest ha demostrado ser efectivo para predecir la calidad del vino en base a estas propiedades.

Se ha alcanzado el objetivo principal del proyecto, que era comprender en profundidad los factores que influyen en la calidad del vino y desarrollar un modelo predictivo para estimar la calidad del vino. El modelo Random Forest optimizado puede ser utilizado como una herramienta valiosa para los productores de vino para mejorar la calidad de sus productos.

Recomendaciones

- Recolección de datos: Se recomienda recolectar más datos, especialmente para la clase "Malo", para mejorar la representatividad de todas las categorías de calidad del vino.
- Variables adicionales: Explorar la incorporación de variables adicionales que podrían ser relevantes para la calidad del vino, como factores ambientales, características del viñedo o métodos de producción.
- Monitoreo del modelo: Implementar un sistema de monitoreo para evaluar el desempeño del modelo en producción y realizar ajustes o actualizaciones periódicas según sea necesario.
- Comunicación efectiva: Comunicar los resultados del análisis y las capacidades del modelo a las bodegas asociadas, destacando los beneficios potenciales para la mejora de la calidad del vino y la optimización de los procesos de producción.

Implicaciones para WineTech Innovations

Los resultados de este estudio proporcionan información valiosa para WineTech Innovations en su proyecto de optimización de la calidad del vino:

Enfocarse en la clase "Malo": Invertir esfuerzos en la recolección de más datos y la exploración de nuevas variables para mejorar la predicción de la clase "Malo".

Optimización continua: Monitorear el desempeño del modelo y realizar ajustes o actualizaciones periódicas para mantener su precisión y efectividad.

Herramienta para la toma de decisiones: Proporcionar a las bodegas asociadas una herramienta valiosa para tomar decisiones informadas sobre la producción de vino, con el objetivo de mejorar la calidad y la consistencia de sus productos.

Consideraciones Finales

Es importante recordar que la calidad del vino es un concepto complejo que involucra factores subjetivos y sensoriales que no pueden ser completamente capturados por modelos de Machine Learning. Sin embargo, este estudio demuestra que los modelos de Machine Learning pueden ser herramientas útiles para identificar patrones y tendencias en los datos fisicoquímicos que pueden ser utilizados para tomar decisiones informadas que mejoren la calidad del producto final. Se espera que la implementación de los modelos y las recomendaciones descritas en este estudio contribuya a la mejora de la calidad del vino producido por las bodegas asociadas a WineTech Innovations, generando un impacto positivo en la satisfacción del cliente, la competitividad y la sostenibilidad de la industria vitivinícola.