



Análisis de calidad de vinos

Data Science – Comisión 19145

Nicolás Boccardo

Facundo Cestari

German Grinberg

Alonzo David Linares Mora



Introducción




Introducción

- En este trabajo de fin de curso, proponemos un enfoque de análisis de datos para predecir la calidad en el sabor del vino humano a partir de 11 características fisicoquímicas.
- Se considera un gran conjunto de datos con muestras de *vino rojo de Portugal* y se realiza un análisis exploratorio de datos.
- Luego se realiza un análisis bivariado y multivariado de los datos para saber la relación entre las distintas variables y nuestro target (la calidad del vino).
- Por ultimo, se realiza un modelo predictivo de árbol de decisión para predecir la calidad del vino de acuerdo a determinadas características fisicoquímicas.



Contexto

- Hoy en día el vino es disfrutado cada vez más por una amplia gama de consumidores.
- Portugal es uno de los diez principales países exportadores de vino, con una cuota de mercado del 3,17 % en 2020.
- Para respaldar su crecimiento, la industria del vino está invirtiendo en nuevas tecnologías tanto para la elaboración como para los procesos de venta.
- La certificación del vino y la evaluación de la calidad son elementos clave en este contexto.
- La evaluación de la calidad suele ser parte del proceso de certificación y se puede utilizar para mejorar la elaboración del vino (mediante la identificación de los factores más influyentes) y para estratificar vinos como marcas premium (útil para fijar precios).
- La certificación del vino se evalúa generalmente mediante pruebas fisicoquímicas y sensoriales.

- 
- Los avances en las tecnologías de la información han hecho posible recopilar, almacenar y procesar conjuntos de datos masivos, a menudo muy complejos.
 - Todos estos datos contienen información valiosa, como tendencias y patrones, que se pueden utilizar para mejorar la toma de decisiones y optimizar las posibilidades de éxito.



Motivación

- La construcción del modelo predictivo es valiosa no solo para las entidades de certificación, sino también para los productores de vino e incluso para los consumidores.
- Se puede utilizar para apoyar las evaluaciones del vino del enólogo, mejorando potencialmente la calidad y la velocidad de sus decisiones.
- Medir el impacto de las pruebas fisicoquímicas en la calidad final del vino es útil para mejorar el proceso de producción.



Análisis exploratorio de los datos

- Se carga el dataset en nuestro compilador y se realizan algunos análisis del mismo.

```
df.describe()
```

✓ 0.1s

	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

- El dataset brinda distintas características fisicoquímicas de los vinos y en la ultima columna 'quality' nos da la calidad del vino. Tiene un total de 1599 datos para cada una de sus 12 columnas.

- 
- Se realiza un análisis del tipo de dato de cada columna del dataset.

```
# Analizamos el tipo de datos de cada columna del dataframe  
df.dtypes
```

```
✓ 0.1s
```

fixed acidity	float64
volatile acidity	float64
citric acid	float64
residual sugar	float64
chlorides	float64
free sulfur dioxide	float64
total sulfur dioxide	float64
density	float64
pH	float64
sulphates	float64
alcohol	float64
quality	int64

- 
- Se realiza un análisis de la cantidad para cada valor de la columna 'quality'

```
df['quality'].value_counts()
✓ 0.3s
```

5	681
6	638
7	199
4	53
8	18
3	10

Name: quality, dtype: int64

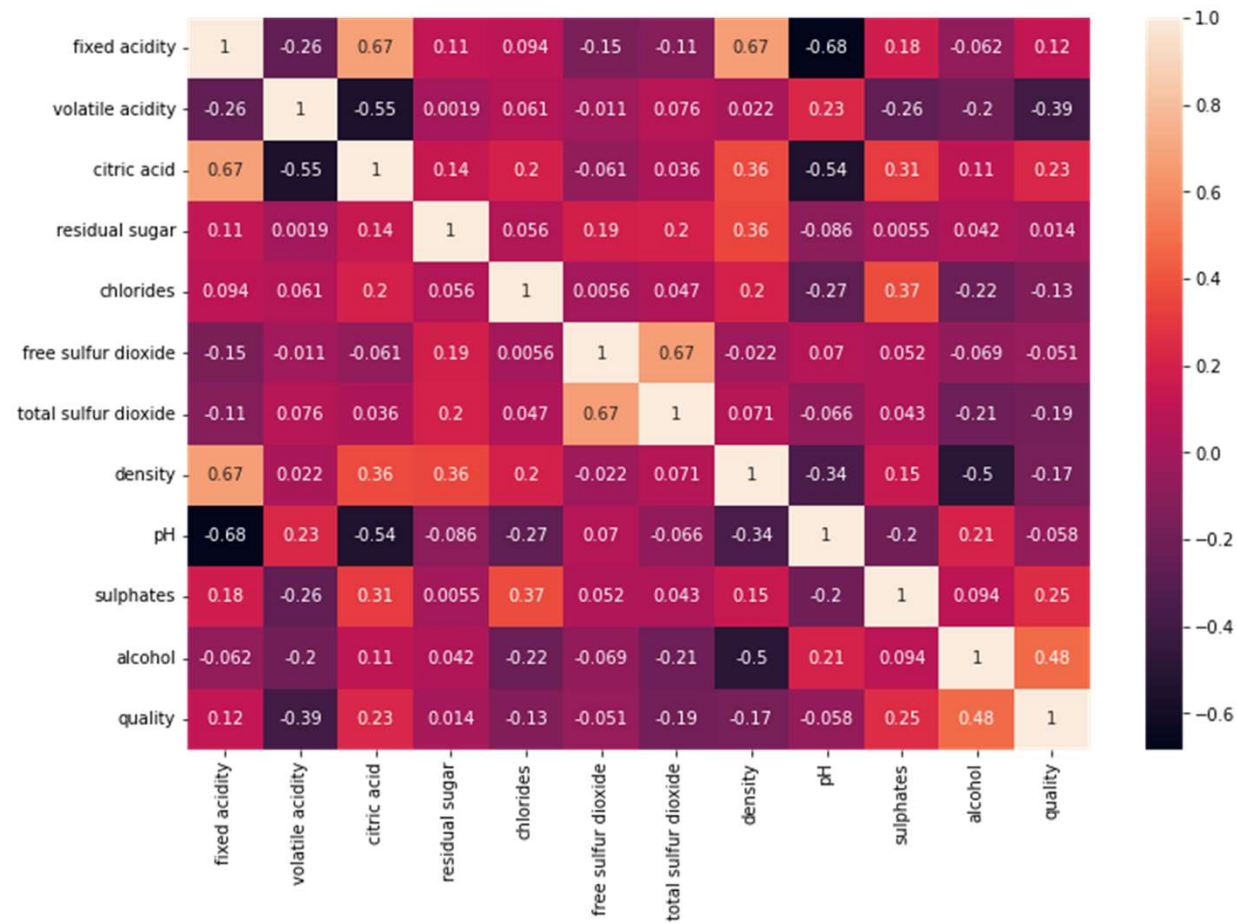
- Se eliminan los valores nulos.


```
# Se eliminan los valores nulos
df.dropna(inplace=True)
✓ 0.2s
```



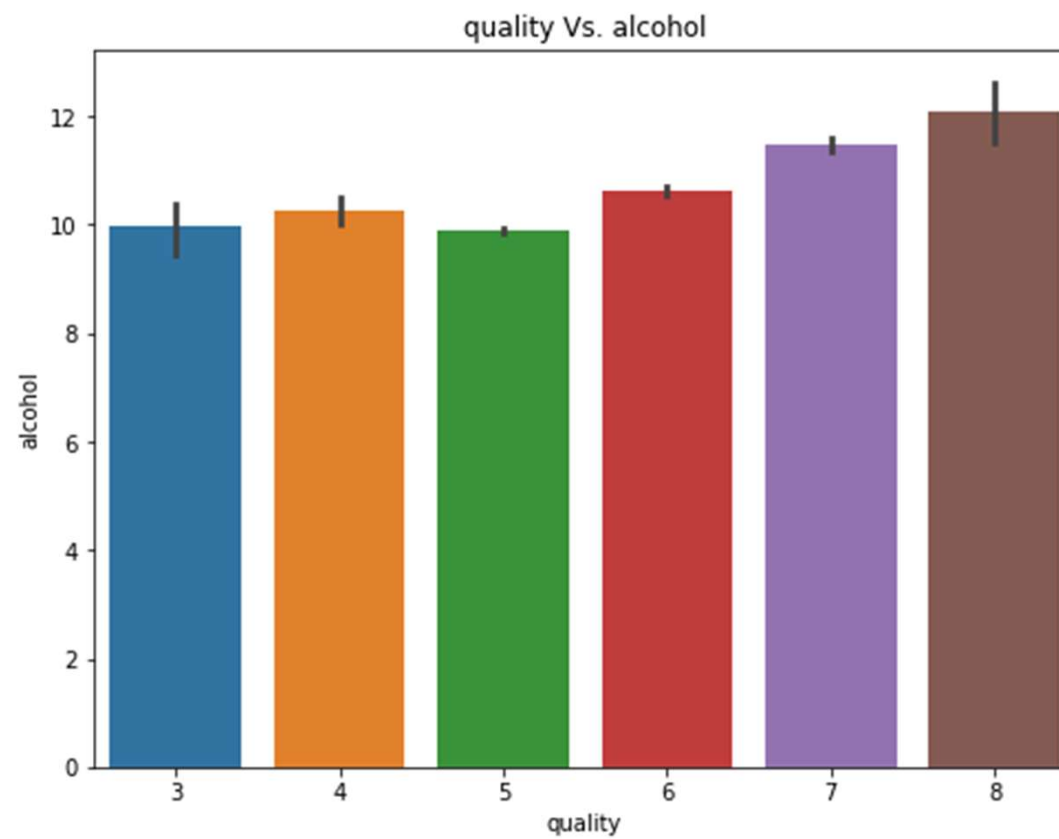
Análisis bivariado/multivariado

- Se realiza una tabla de correlaciones para obtener la correlación entre las distintas variables

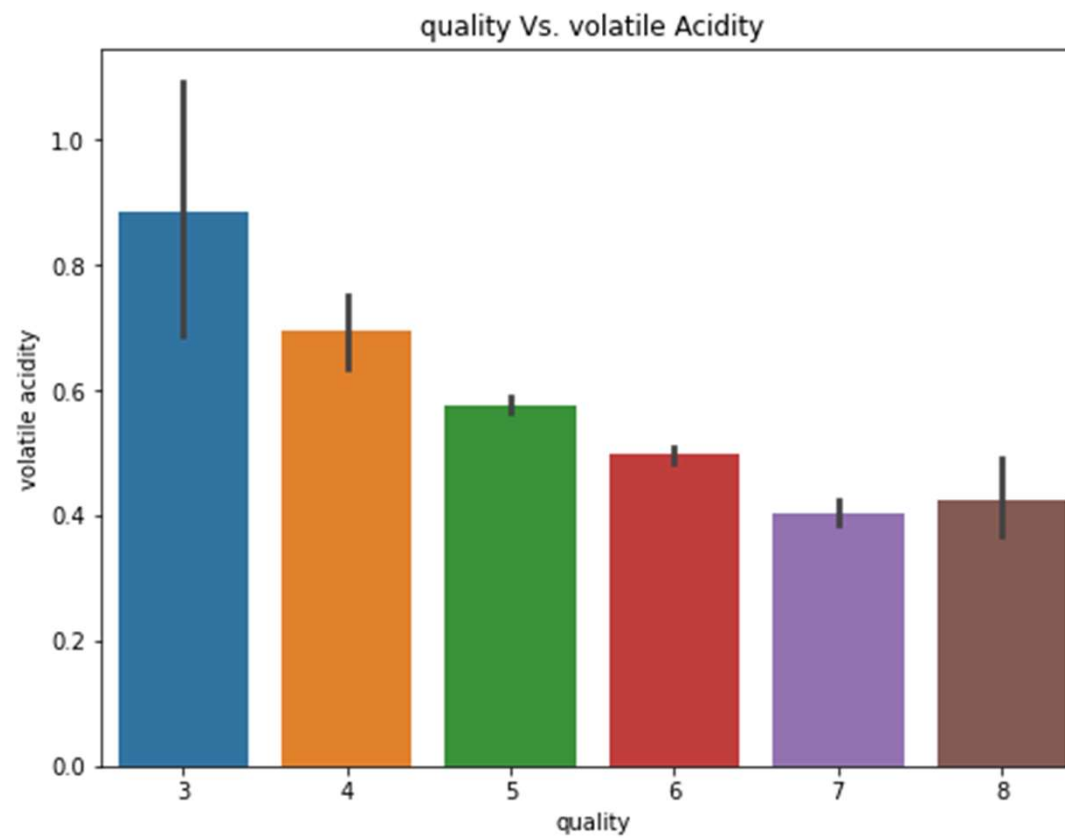


- 
- De la tabla anterior, podemos concluir que nuestro target 'quality' tiene las siguientes relaciones:
 - Relación directamente lineal de 0,48 con la variable 'alcohol'.
 - Relación directamente lineal de 0,25 con la variable 'sulphates'.
 - Relación inversamente lineal de 0,39 con la variable 'volatile acidity'.

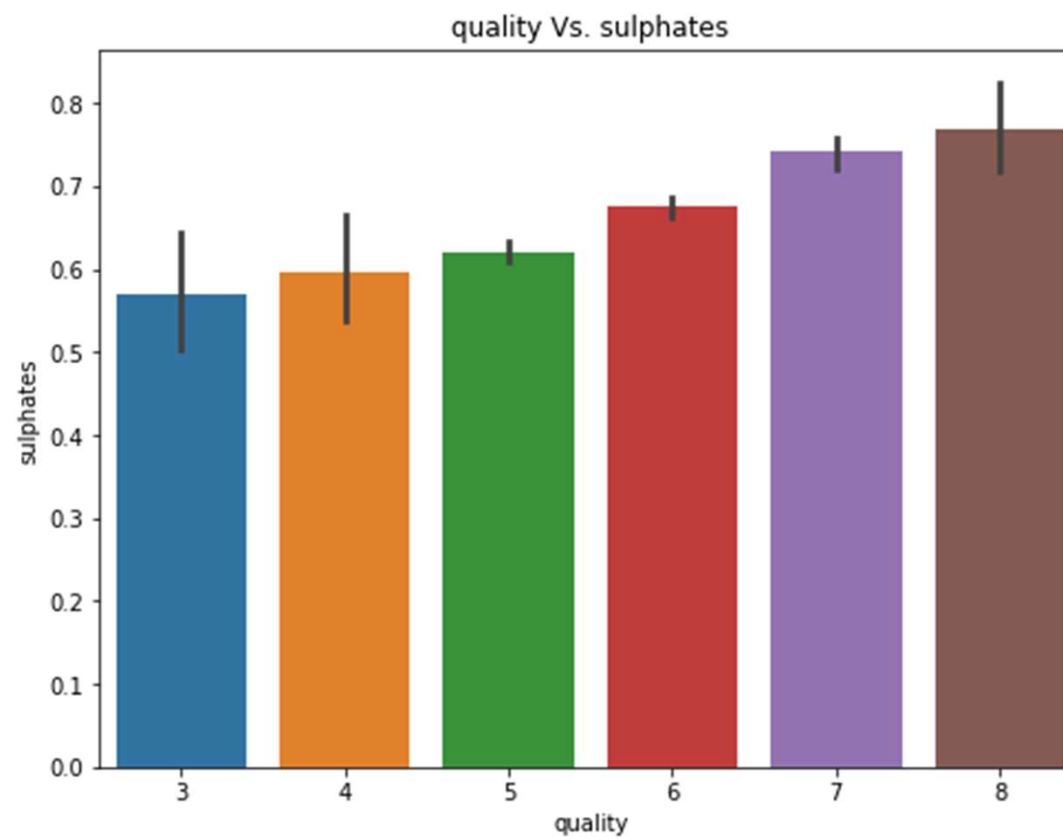
Relación entre 'quality' y 'alcohol'




- Relación entre 'quality' y 'volatile acidity'



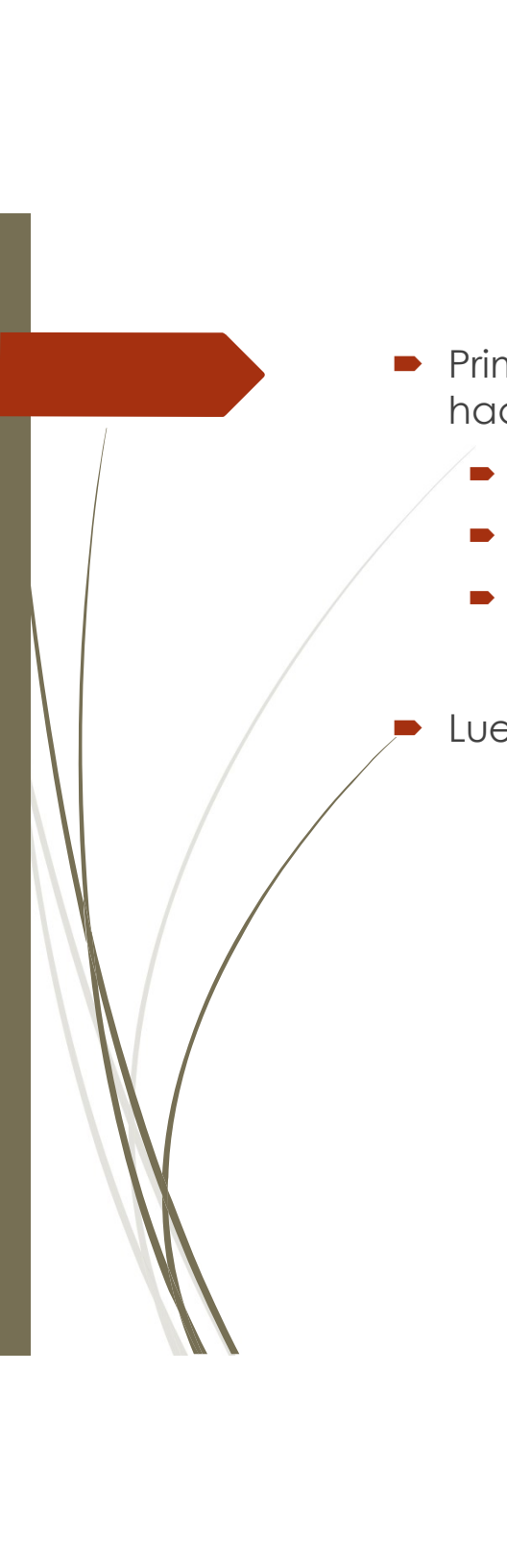
Relación entre 'quality' y 'sulphates'



- 
- En base a este análisis, se toma la decisión de realizar un árbol de decisión para la variable 'quality', con énfasis en las tres variables 'sulphate', 'alcohol' y 'volatile acidity'.



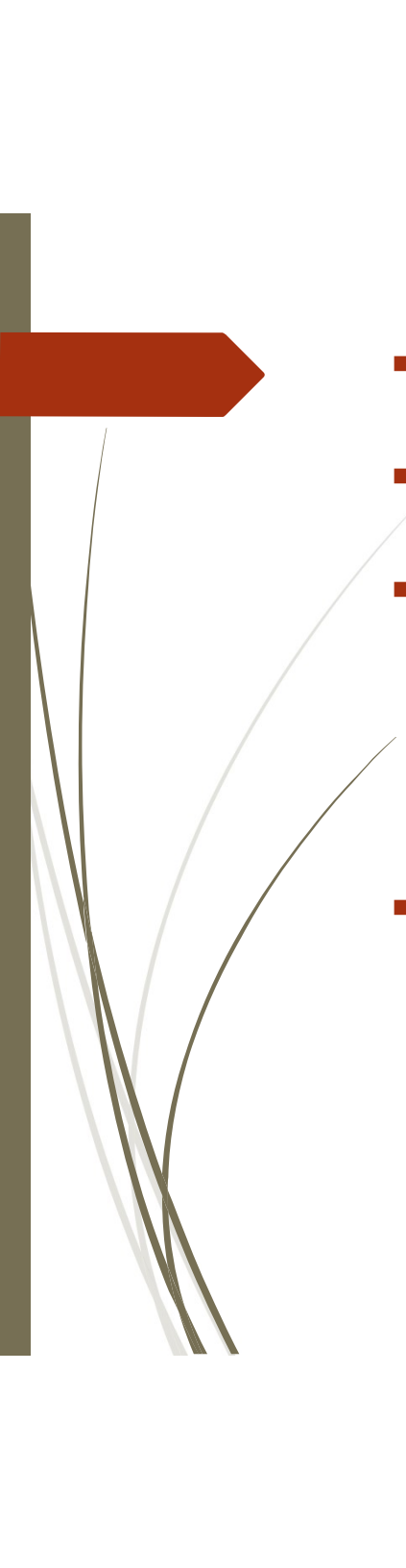
Modelo predictivo

- 
- Primero se realiza un agrupamiento en los valores de nuestro target para hacer mas fácil el análisis y el modelo predictivo, siguiendo la relación:
 - Aquellos vinos con calidad entre 0 y 4 son 'malo'
 - Aquellos vinos con calidad entre 4 y 6 son 'regular'
 - Aquellos vinos con calidad entre 7 y 8 son 'bueno'
 - Luego del agrupamiento, las cantidades resultan:

```
df['quality'].value_counts()
✓ 0.2s
```

regular	1319
bueno	217
malo	63

Name: quality, dtype: int64

- 
- A continuación, se separa el dataset en dos: 'X' e 'y'. 'y' contiene la columna de quality y 'X' contiene el resto del dataset original.
 - Luego definimos que un 70% de nuestro dataset será para entrenar el modelo y el 30% restante será para testearlo
 - Se crea un árbol de decisión con los siguientes parámetros:
 - Max_Depth: 2.
 - random_state: 42.
 - Min_sample_Split: 10.
 - Este modelo arroja los siguientes resultados de acierto:
 - 85.6% de acierto en el dataset de entrenamiento.
 - 82,7% de acierto en el dataset de testeo.