FRACTAL
SOLUTIONS

EVIDENCIA

Machine Learning - Analítica de Datos y Herramientas de Inteligencia Artificial

EQUIPO

Carlos Verdaguer
Project Manager

Fortunato Martínez
Director of Operations

Jorge Oviedo
Director of Logistics

Gerardo Barajas
Data Scientist

Sebastian Neira
Data Analyst

Anibal Angulo
Data Analyst

FRACTAL
SOLUTIONS

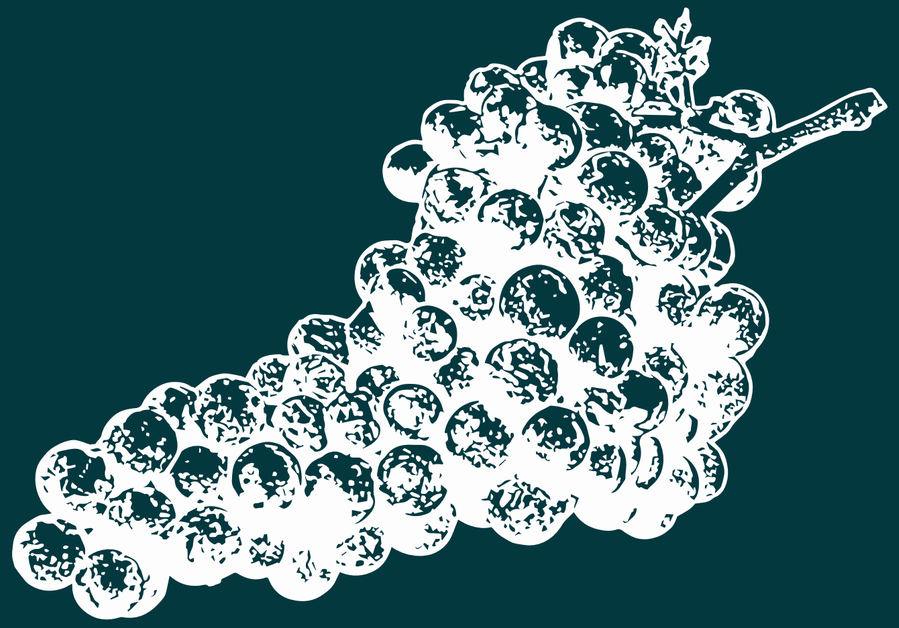
CONTENIDOS

1. Dataset
2. Data Preprocessing
3. Modelos de Regresión
4. Modelos de Clasificación
5. Clustering
6. Principal Component Analysis

Dataset

FRACTAL
SOLUTIONS

Nuestro Objetivo



QUE SE BUSCA OBTENER

Obtener modelos de aprendizaje automático para la clasificación de tipo de vinos y modelos de regresión para predecir la calidad de estos.

Wine Quality

FRACTAL SOLUTIONS

Obtención del dataset kaggle

Variables

FRACTAL SOLUTIONS

Type

Fixed acidity

Volatile acidity

Citric acid

Residual sugar

Sulphates

Chlorides

Free sulfur dioxide

Total sulfur dioxide

Density

pH

Alcohol

Variables

FRACTAL SOLUTIONS

Type

Fixed Acidity

Volatile acidity

Citric acid

Residual sugar

Sulphates

Chlorides

Free sulfur dioxide

Total sulfur dioxide

Density

pH

Alcohol

Variables

FRACTAL SOLUTIONS

Type

Fixed acidity

Volatile acidity

Citric acid

Residual sugar

Sulphates

Chlorides

Free sulfur dioxide

Total sulfur dioxide

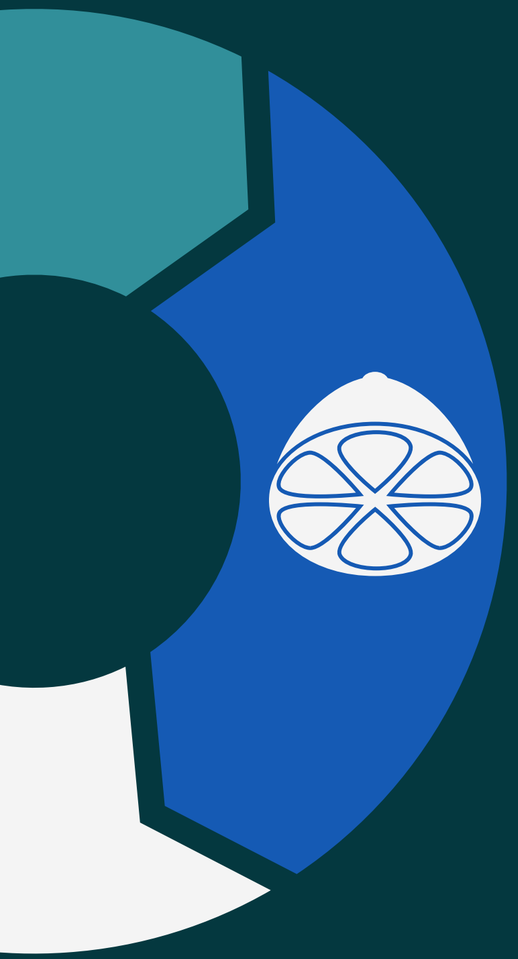
Density

pH

Alcohol

Variables

FRACTAL SOLUTIONS



Type

Fixed acidity

Volatile acidity

Citric acid

Residual sugar

Sulphates

Chlorides

Free sulfur dioxide

Total sulfur dioxide

Density

pH

Alcohol

Variables

FRACTAL SOLUTIONS

Type

Fixed acidity

Volatile acidity

Citric acid

Residual sugar

Sulphates

Chlorides

Free sulfur dioxide

Total sulfur dioxide

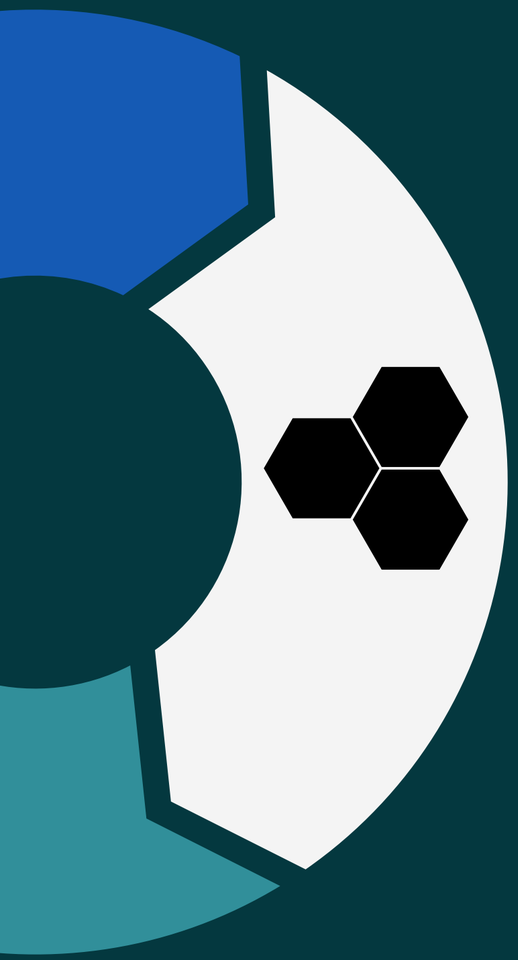
Density

pH

Alcohol

Variables

FRACTAL SOLUTIONS



Type

Fixed acidity

Volatile acidity

Citric acid

Residual sugar

Sulphates

Chlorides

Free sulfur dioxide

Total sulfur dioxide

Density

pH

Alcohol

Variables

FRACTAL SOLUTIONS

Type

Fixed acidity

Volatile acidity

Citric acid

Residual sugar

Sulphates

Chlorides

Free sulfur dioxide

Total sulfur dioxide

Density

pH

Alcohol

Variables

FRACTAL SOLUTIONS

Type

Fixed acidity

Volatile acidity

Citric acid

Residual sugar

Sulphates

Chlorides

Free sulfur dioxide

Total sulfur dioxide

Density

pH

Alcohol

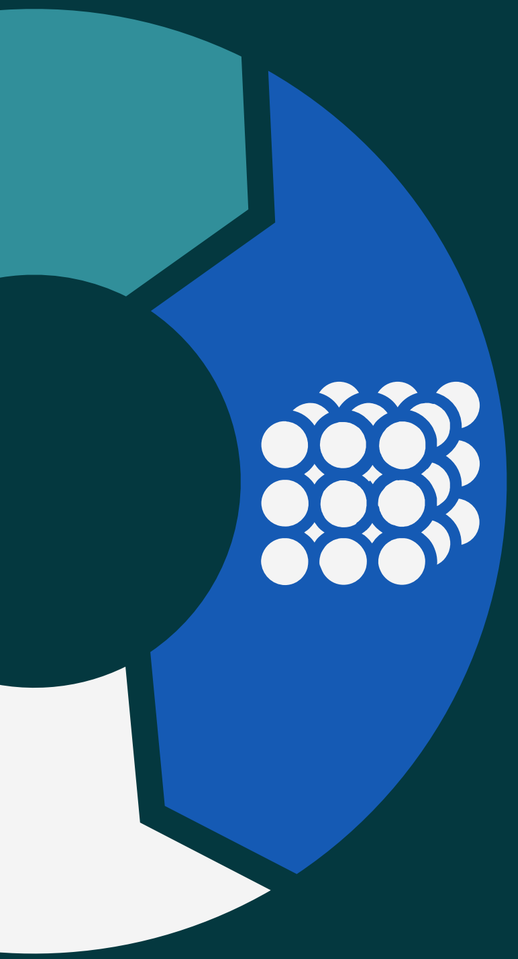
Variables

FRACTAL SOLUTIONS

Type	Chlorides
Fixed acidity	Free sulfur dioxide
Volatile acidity	**Total sulfur dioxide**
Citric acid	Density
Residual sugar	pH
Sulphates	Alcohol

Variables

FRACTAL SOLUTIONS



Type

Fixed acidity

Volatile acidity

Citric acid

Residual sugar

Sulphates

Chlorides

Free sulfur dioxide

Total sulfur dioxide

Density

pH

Alcohol

Variables

Type	Chlorides
Fixed acidity	Free sulfur dioxide
Volatile acidity	Total sulfur dioxide
Citric acid	Density
Residual sugar	**pH**
Sulphates	Alcohol

Variables

FRACTAL SOLUTIONS

Type

Fixed acidity

Volatile acidity

Citric acid

Residual sugar

Sulphates

Chlorides

Free sulfur dioxide

Total sulfur dioxide

Density

pH

Alcohol

Variables

FRACTAL SOLUTIONS

Quality Type

Data Preprocessing

Tratamiento de Datos

FRACTAL SOLUTIONS

Limpieza de datos nulos/faltantes

```
[ ]  dataset.isnull().sum()
```

```
type                    0
fixed acidity          10
volatile acidity        8
citric acid             3
residual sugar          2
chlorides               2
free sulfur dioxide     0
total sulfur dioxide    0
density                 0
pH                      9
sulphates               4
alcohol                 0
quality                 0
dtype: int64
```

```
dataset.dropna(inplace=True)
dataset.reset_index(inplace=True)
dataset
```

```
[ ]  del dataset['index']
```

Relaciones entre Variables

CON Datos Atípicos

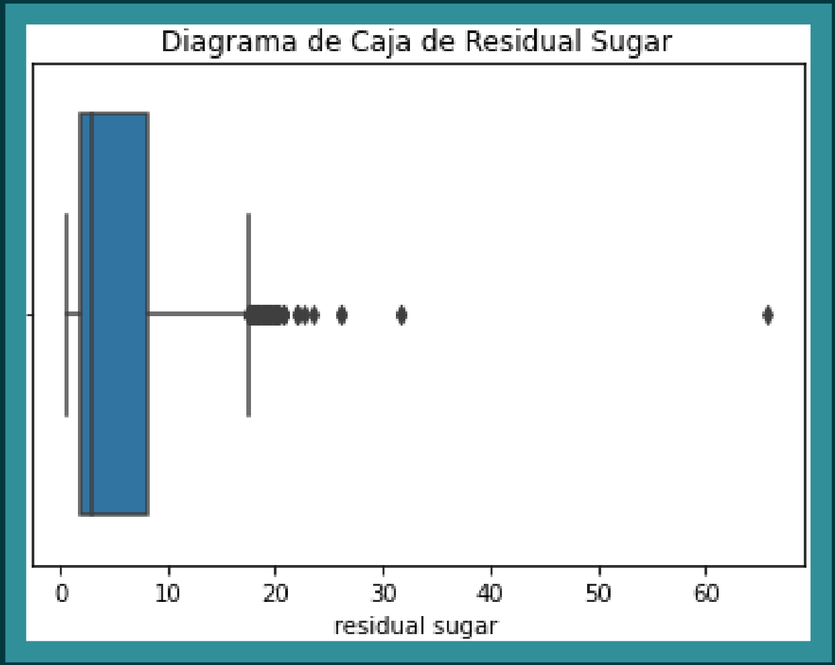
SIN Datos Atípicos

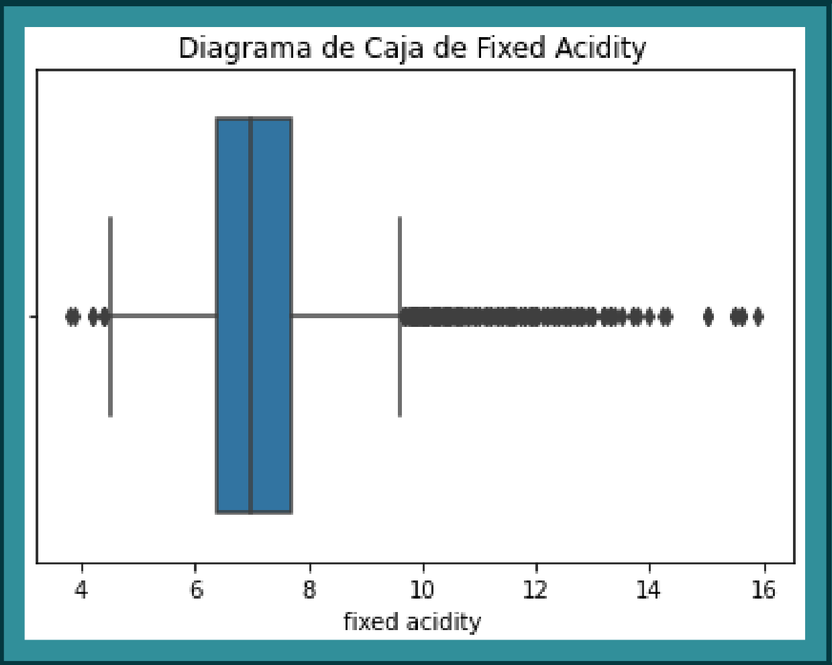
Diagramas de Caja

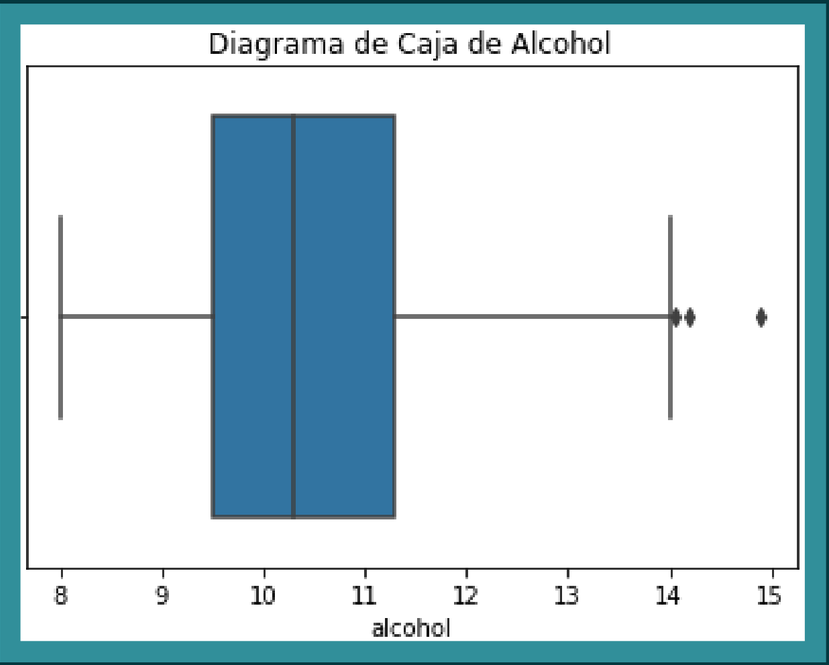
FRACTAL SOLUTIONS

CON Datos Atípicos

En las variables que más describen al vino



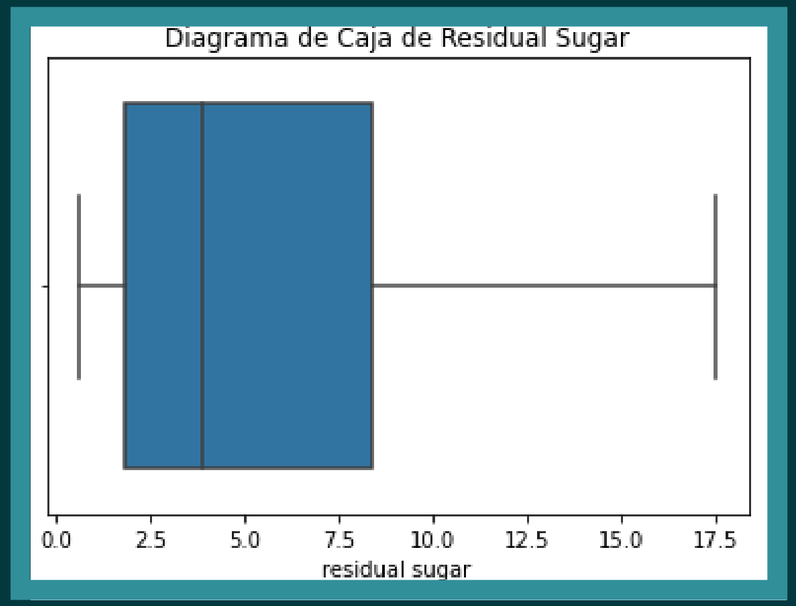


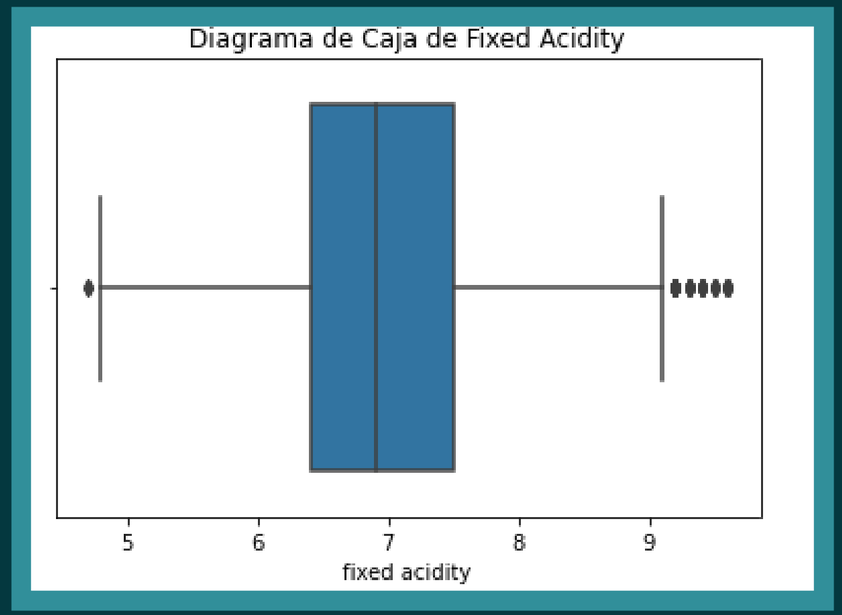


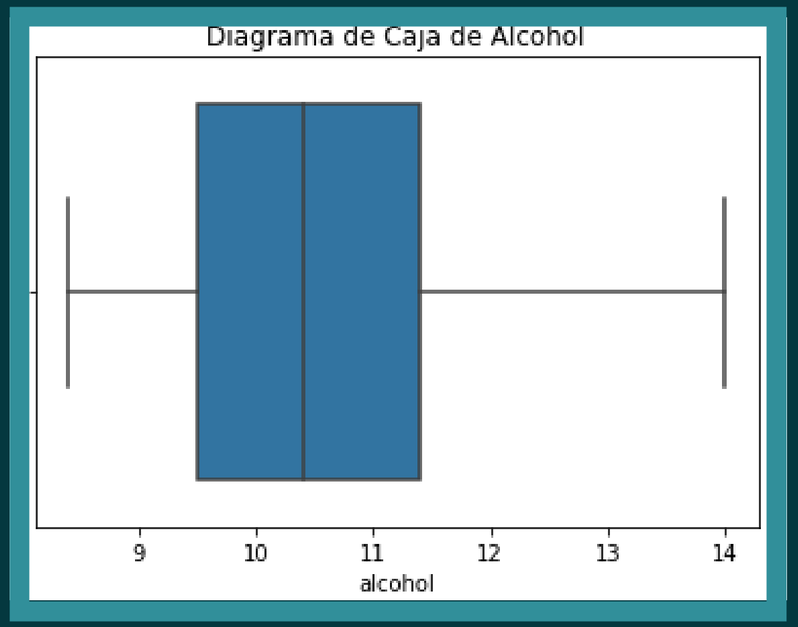
FRACTAL SOLUTIONS

SIN Datos Atípicos

En las variables que más describen al vino







FRACTAL SOLUTIONS

Dimensiones

Número de registros

Con datos atipicos

V BLANCO	V ROJO	TOTAL
4870	1593	6463

Sin datos atipicos

V BLANCO	V ROJO	TOTAL
4311	682	4993

```
print('Dimensión Antes:', dataset.shape)
dataset.type.value_counts()
```

```
dataset.drop(indices_datos_atipicos, inplace=True)
print('Dimensión Después:', dataset.shape)
dataset.type.value_counts()
```

Explicar el proceso para eliminación de los datos atípicos

Como quedaron las variables antes y después, si se puede unos scatterplots de las variables más representativas (como residual sugars, fixed acidity, alcohol,..) antes y después de quitarle los atípicos

No poner muchos boxplots, solo de esas variables.

Poner las dimensiones del dataset antes y después de eliminar los datos atípicos.

Hacer un gráfico de la distribución de los datos para quality y type antes y después de quitar los datos atípicos.

Modelos de Regresión

FRACTAL SOLUTIONS

Intro

Tipos de Modelos

1. Regresion Logistica
2. Random Forest
3. Support Vector Machine

Objetivo

Atraves de los siguientes modelos se busca predecir la calidad del alcohol en base a las variables numericas del vino.

FRACTAL SOLUTIONS

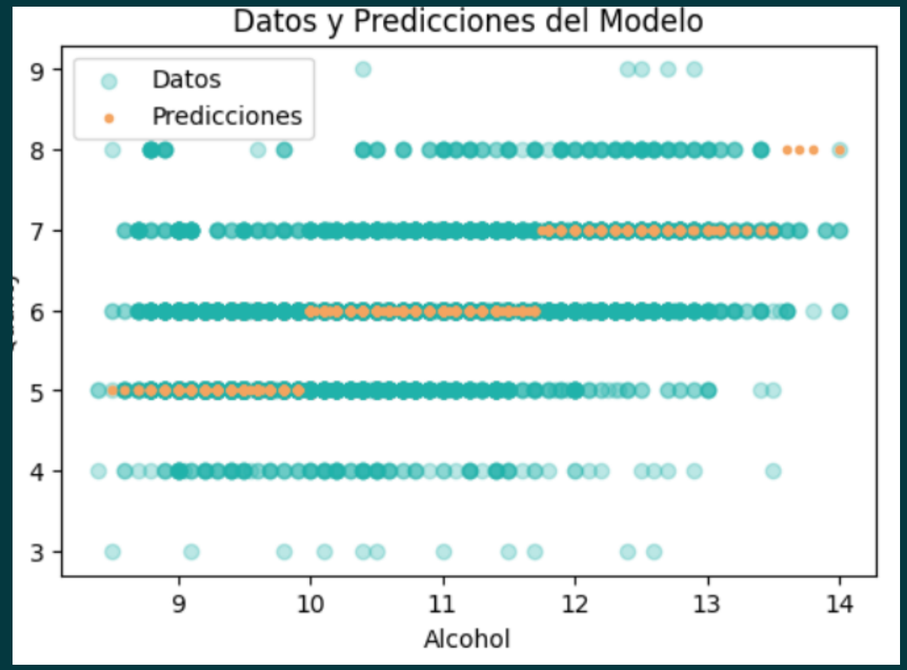
Variables

	type	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality
0	white	6.3	0.300	0.34	1.6	0.049	14.0	132.0	0.99400	3.30	0.49	9.5	6

FRACTAL SOLUTIONS

Modelo 1: Regresión Lineal OLS

En este modelo solo se utilizo la variable x como alcohol para predecir la variable y de quality.



FRACTAL SOLUTIONS

Modelo 2 y 3 : Random Forest y Support Vector Machine

Para estos modelos se utilizaron todas las varisbles:
x: fixed acidity, volatile acidity, citric acid, residual sugar, chlorides, free sulfur, etc...
para calcular la variable y: quality.

```
SVR
SVR(C=1, gamma=0.6)
```

```
[85] from sklearn.ensemble import RandomForestRegressor
     trees = 50
     random_forest = RandomForestRegressor(n_estimators = trees, random_state = 0, criterion='squared_error')
     random_forest.fit(x2_train,y2_train.ravel())

     RandomForestRegressor(n_estimators=50, random_state=0)
```

FRACTAL SOLUTIONS

Comparación de Modelos

Se comparan los modelos con las siguientes métricas de regresión

	Métricas de Regresión			
	Coeficiente R^2	MSE	MAE	Max Error
Regresión Lineal	0.980	0.7107	0.5746	4.0
Random Forest	0.5447	0.4416	0.4853	3.3786
Support Vector Regression	0.5447	0.4416	0.4853	3.3786

Modelos de Clasificacion

FRACTAL SOLUTIONS

Explicar al inicio que se busca predecir, que en este caso para la clasificación es el tipo de vino.

Hacer una diapositiva para cada modelo (son 3), explicando sus parámetros y que cosas se hicieron. Que variables se consideraron, etc. Importante poner que se aplican modelos con pesos determinados dado el desbalance de clases (el parámetro class_weight en cada modelo, donde se penaliza más a las clases con menor cantidad). Si es posible poner una gráfica de cada modelo con sus predicciones. Poner también la curva ROC

Luego mostrar una tabla que resuma las métricas de todos los modelos en una dispoisitva aparte (todo en una); ahi poner que tan bien funciona el modelo en training, testing. Poner ahi tmb el reporte de clasificación (el que tiene el accuracy, precision, f1-score, recall, AUC ; serán imporantes para interpretar el modelo).

FRACTAL SOLUTIONS

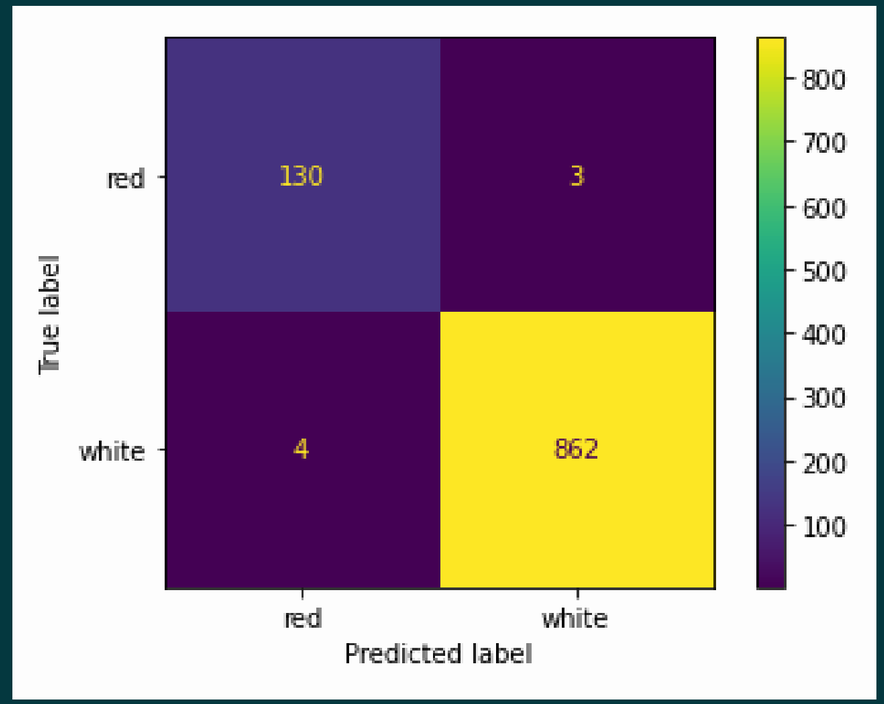
Modelo 1: Regresión Logística

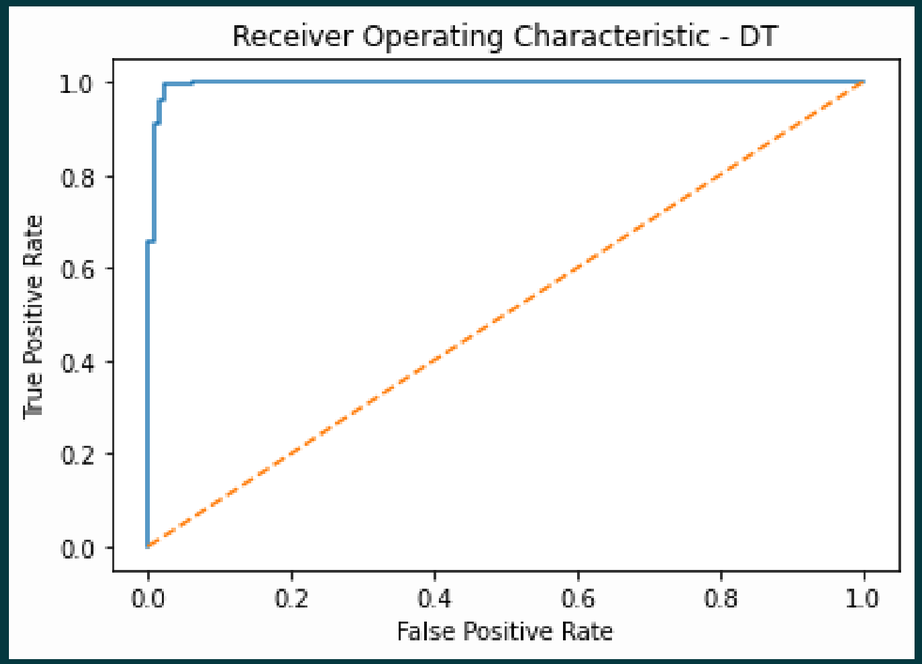
- **Importar datos limpiados**
- **Separar variables**
  - Predictores (Acidez, azúcar, densidad, alcohol, etc.)
  - Respuesta (Tipo de vino)
- **Generar datasets**
  - 80% – Training
  - 20% – Test
- **Establecer pesos**
  - Blancos – 1x
  - Rojos – 3x

Modelo 1: Regresión Logística

- **Score:** 0.993

- **Validación cruzada:**
  - Accuracy: 99.52%
  - Standard Deviation: 0.31%





FRACTAL SOLUTIONS

Modelo 2: Random Forest

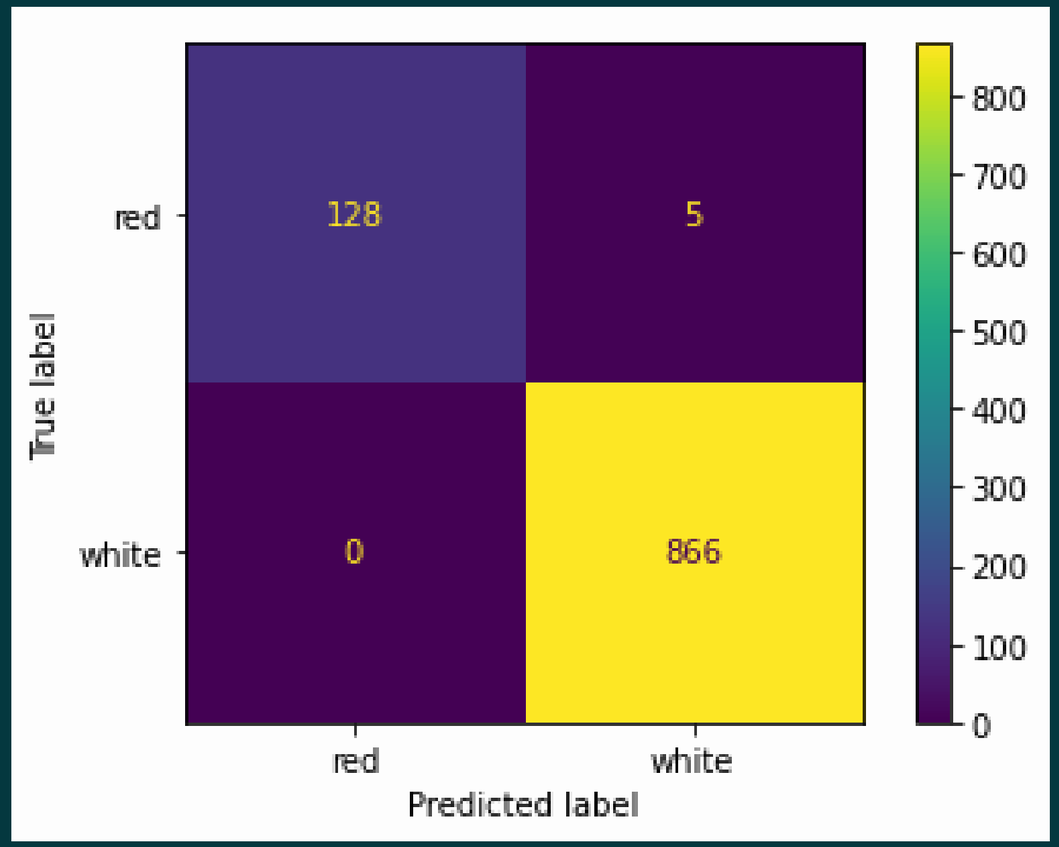
- Repetición de pasos anteriores

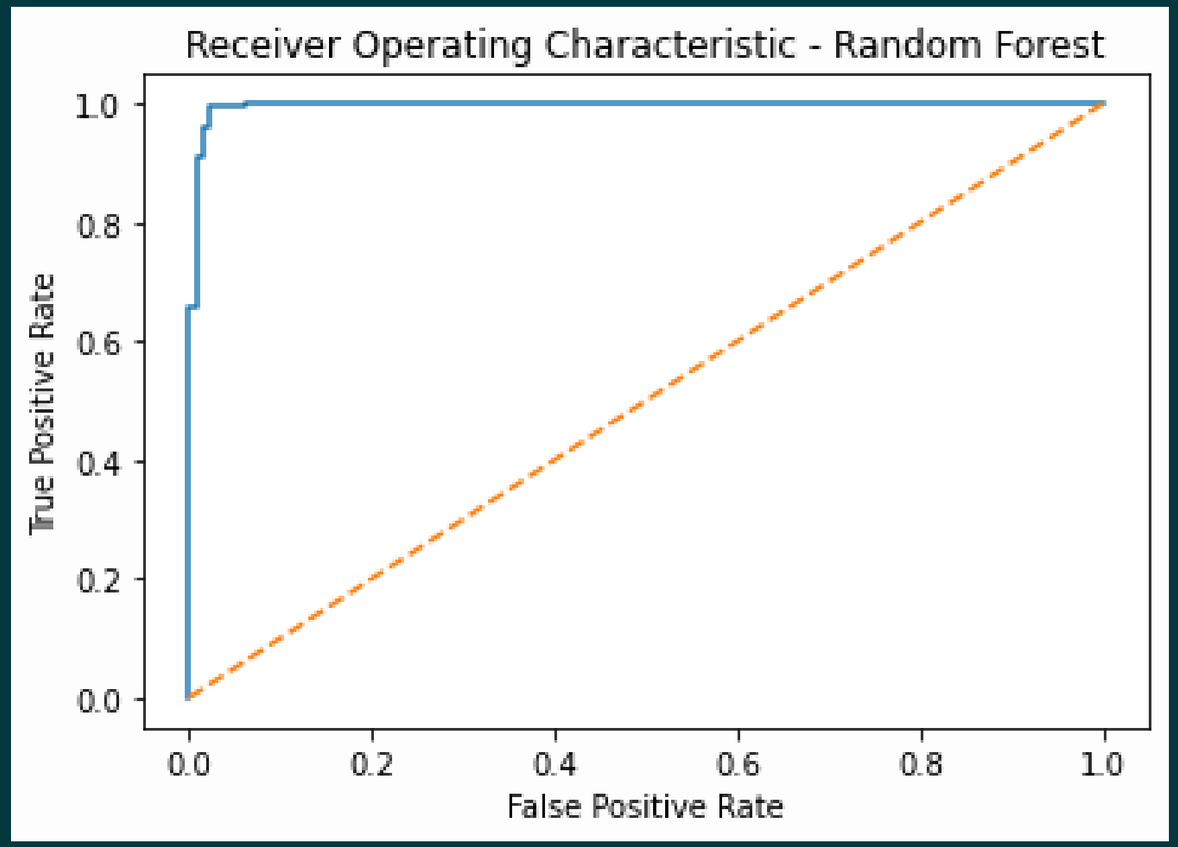
- From **sklearn**
  - **Train Test Split**
  - **Standard Scaler**
  - **Random Forest Classifier**

```
RandomForestClassifier
RandomForestClassifier(class_weight={'red': 3, 'white': 1}, n_estimators=15,
                       random_state=0)
```

Modelo 2: Random Forest

Score: 0.995 **Accuracy:** 99.57% **Standard D:** 0.34%





FRACTAL SOLUTIONS

Comparación de Modelos

Se comparan los modelos con las siguientes métricas

	Métricas de Regresión				
	Accuracy	**Precision**	**F1 - Score**	**Recall**	**AUC**
Regresión Lineal					
Random Forest					

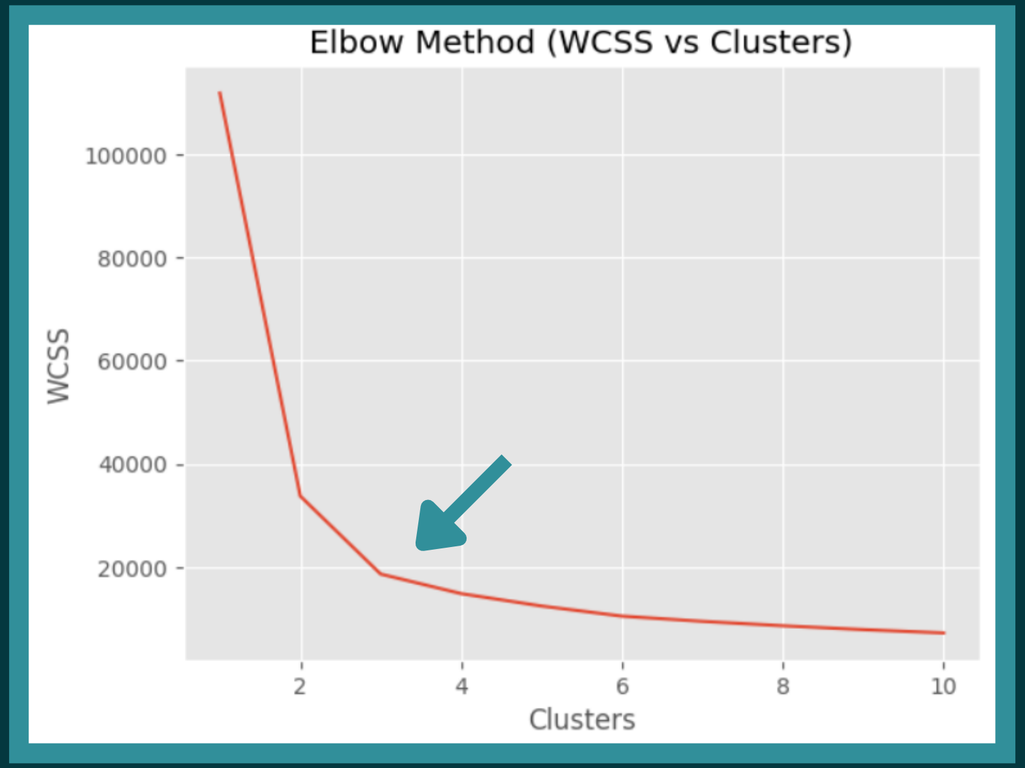
Clustering

FRACTAL SOLUTIONS

K-Means Clustering

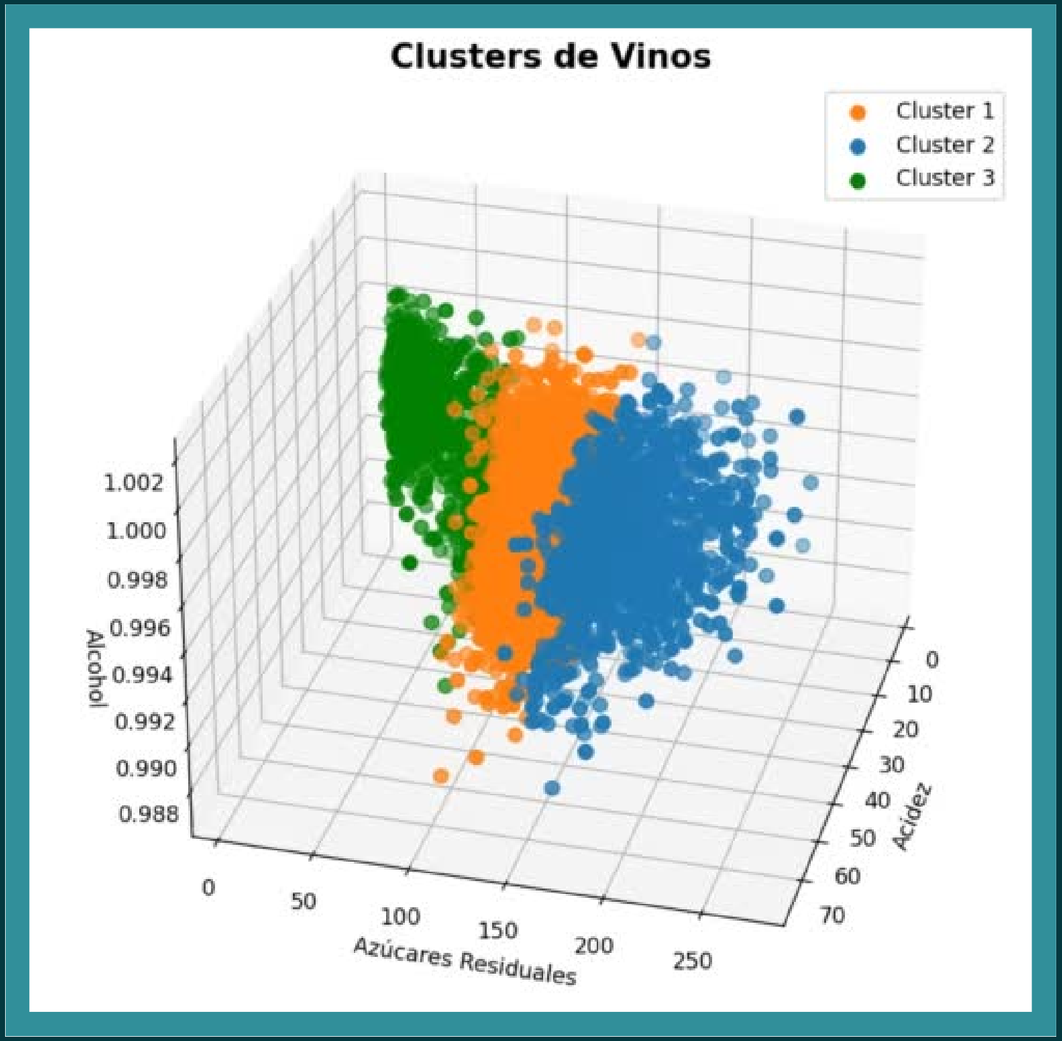
Se hace un clustering de los vinos con base en las variables que más lo describen acorde a fuentes especializadas:

- Son el nivel de alcohol, que tan dulce es (azúcares residuales), acidez, los taninos, etc.
- Se aplica el método de K-Means clustering con K = 3
    - Se utilizó el método del codo para determinar la cantidad de clusters



FRACTAL SOLUTIONS

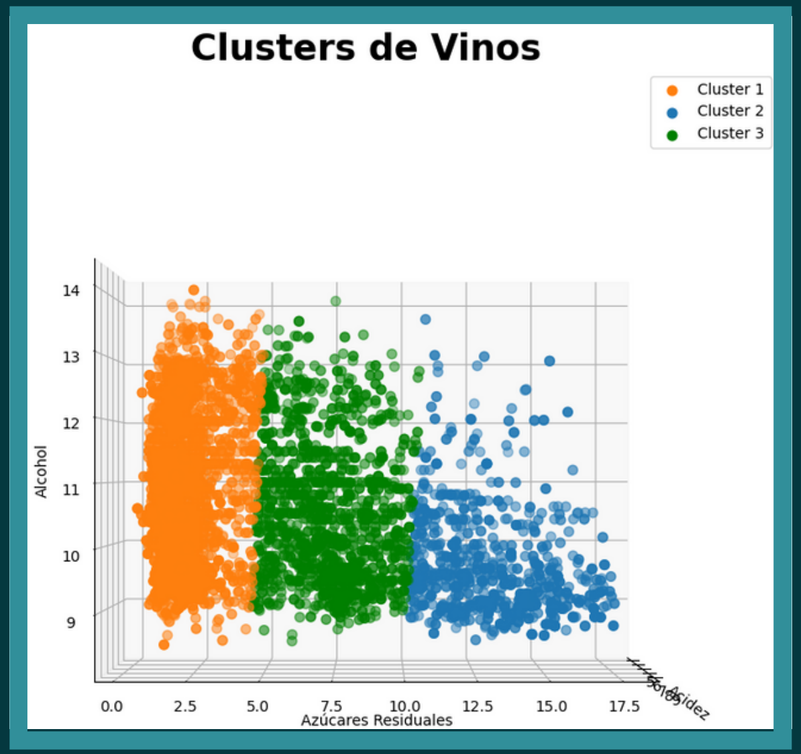
Clustering Resultante



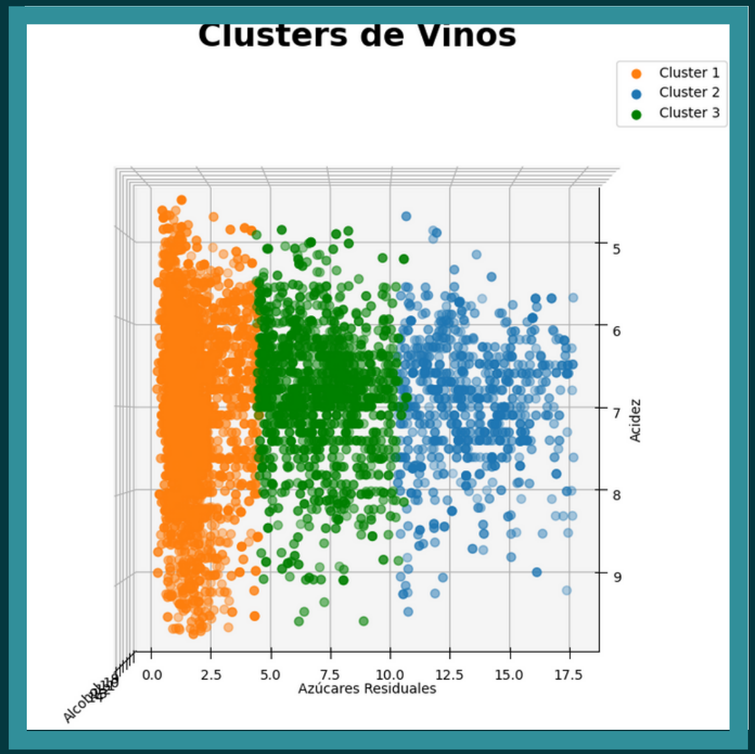
FRACTAL SOLUTIONS

Sub-clusters del Clustering Total

A su vez se obtiene clusters entre las comparaciones relevantes de estas variables mencionadas



¿Entre más azúcares residuales mayor la tendencia a menor acidez?



¿ A mayores azúcares residuales mayor rango de acidez?

PCA

FRACTAL SOLUTIONS

Consideraciones Iniciales

- Se seleccionan todas las variables excepto las de respuesta (type y quality).
- Al seleccionar las variables **se les aplica un feature scaling dada la diferencia de valores** y así aprovechar mejor el PCA
- **¿Qué tipo de escalamiento escoger?** Verificar performance con mismo valor de N
    - Estandarización
    - Normalización
- Con **normalización** hay menor pérdida de información

```python
from sklearn.decomposition import PCA
pca_std = PCA(n_components=3)
principalComponents_std = pca_std.fit_transform(x_std)

#Mostrar porcentaje de variación explicada por los componentes principales
print('Porcentaje de Variacion Explicada:',sum(pca_std.explained_variance_ratio_)*100)
```

```
Porcentaje de Variacion Explicada: 62.173409043872184
```

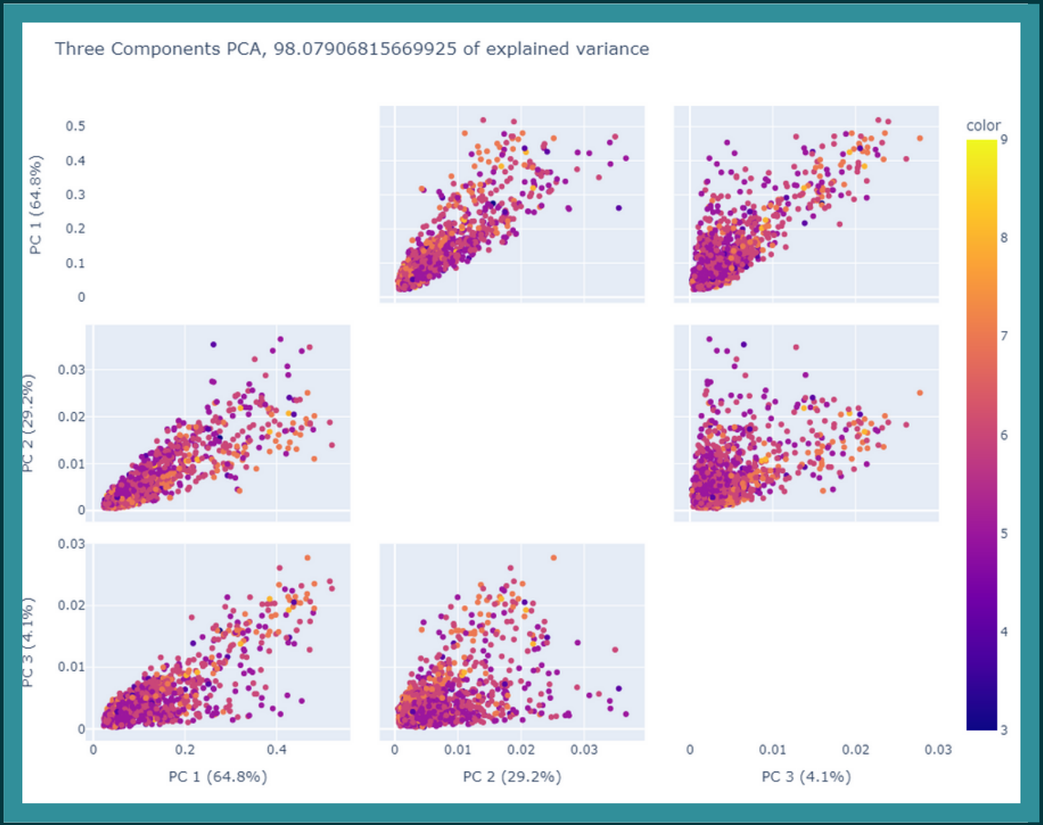
```python
from sklearn.decomposition import PCA
pca_n = PCA(n_components=3)
principalComponents_n = pca_n.fit_transform(x_n)

#Mostrar porcentaje de variación explicada por los componentes principales
print('Porcentaje de Variacion Explicada:',sum(pca_n.explained_variance_ratio_)*100)
```

```
Porcentaje de Variacion Explicada: 98.07906815669925
```

FRACTAL SOLUTIONS

Resultados de PCA

Se tiene una explicación del 94% con dos componentes y 98% con 3 componentes. Se obtiene el porcentaje de la variación explicada por cada componente y se almacenan en un dataframe.

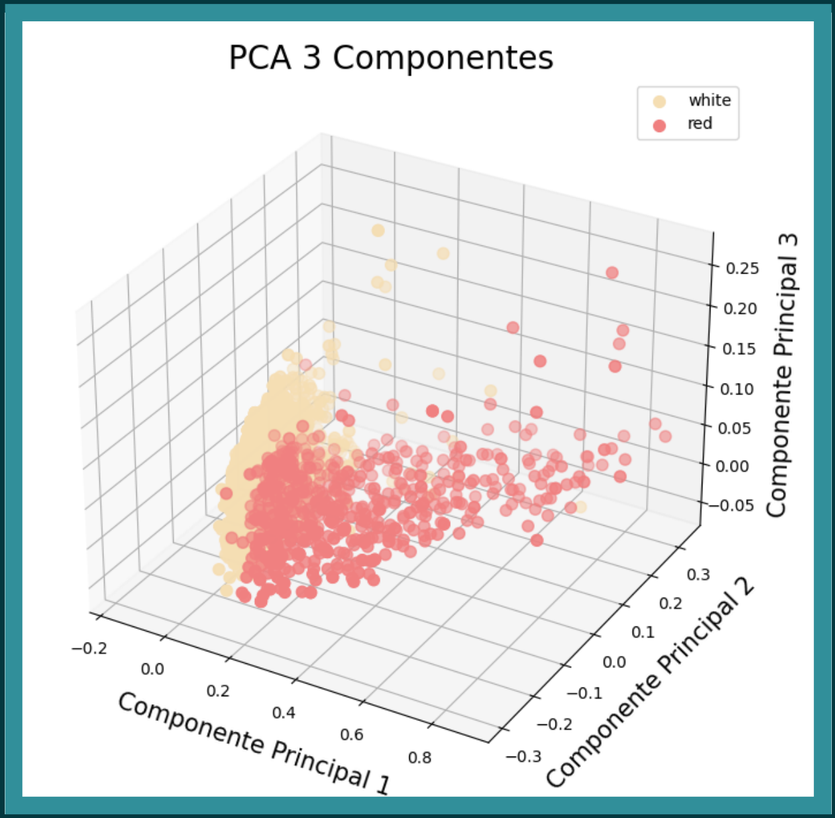


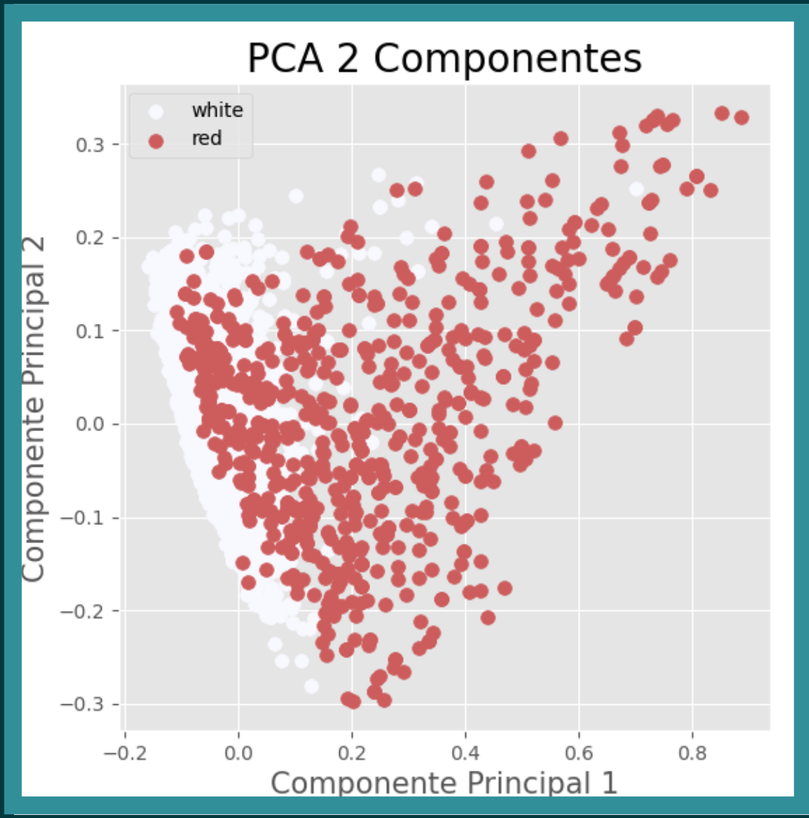
	PC1	PC2	PC3	type	quality
0	-0.107039	0.118904	-0.025498	white	6
1	0.012205	-0.038100	0.021977	white	6
2	-0.068274	-0.018408	0.007810	white	6
3	-0.068274	-0.018408	0.007810	white	6
4	0.012205	-0.038100	0.021977	white	6
...	...	...	...	...	...
4988	0.274732	-0.262046	-0.013748	red	6
4989	0.250447	-0.269738	-0.015200	red	5
4990	0.276485	-0.252977	-0.010144	red	6
4991	0.244999	-0.272874	-0.014438	red	5
4992	0.186330	-0.065984	0.011558	red	6

FRACTAL SOLUTIONS

Visualización de Componentes

Se tiene una explicación del 94% con dos componentes y 98% con 3 componentes. Se obtiene el porcentaje de la variación explicada por cada componente y se almacenan en un dataframe.





CONCLUSIONES