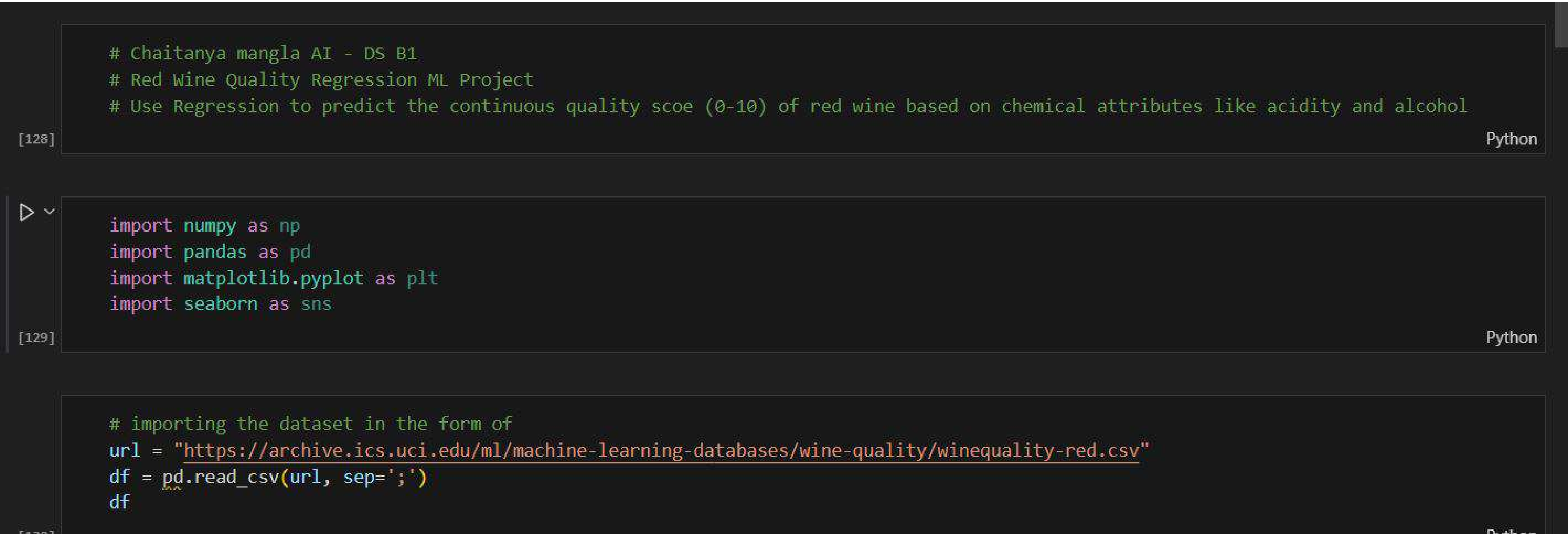
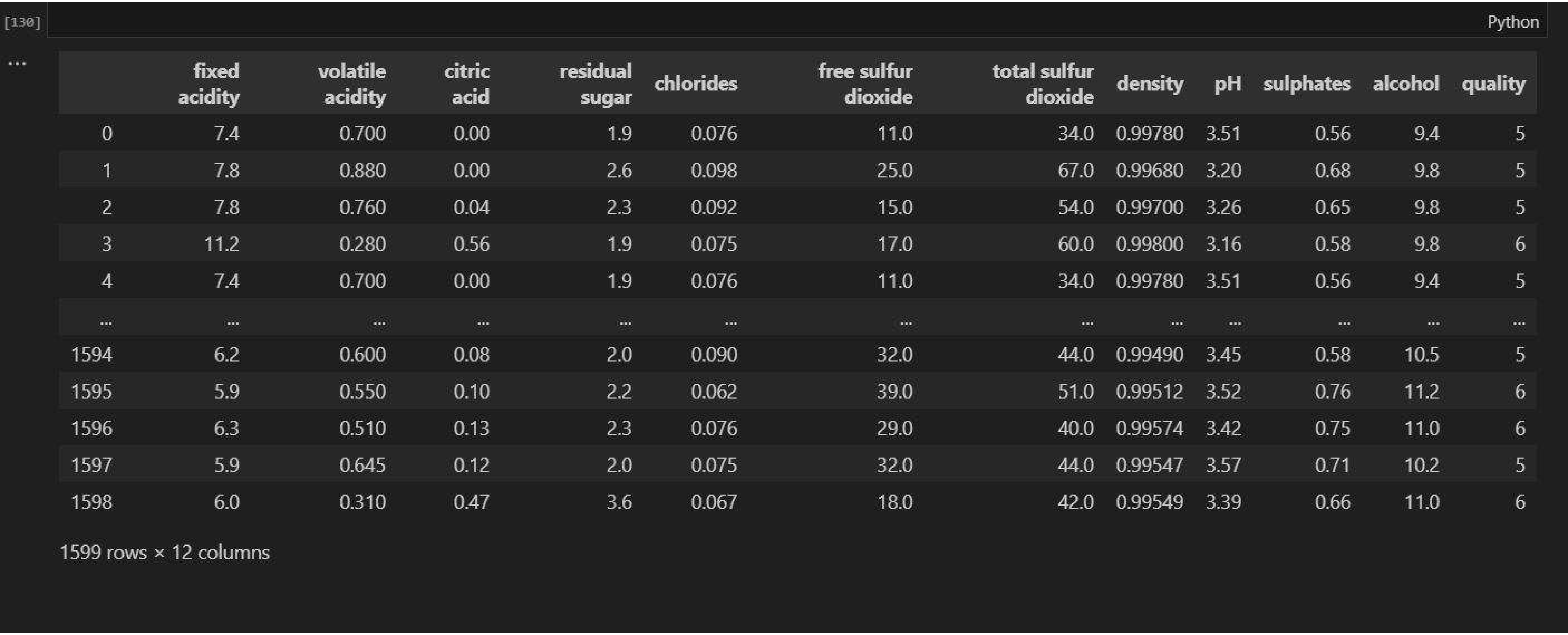


```python
# Chaitanya mangla AI - DS B1
# Red Wine Quality Regression ML Project
# Use Regression to predict the continuous quality scoe (0-10) of red wine based on chemical attributes like acidity and alcohol
```

```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

```python
# importing the dataset in the form of
url = "https://archive.ics.uci.edu/ml/machine-learning-databases/wine-quality/winequality-red.csv"
df = pd.read_csv(url, sep=';')
df
```



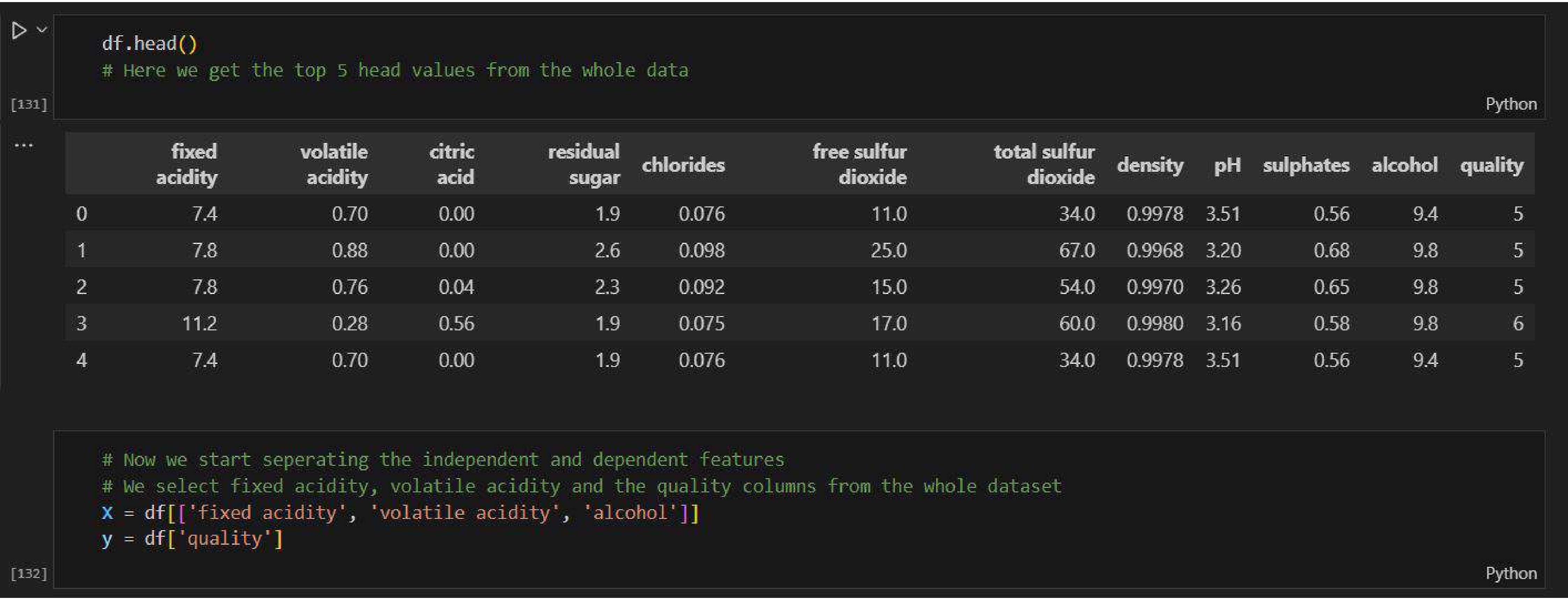
[130]

Python

...

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality
0	7.4	0.700	0.00	1.9	0.076	11.0	34.0	0.99780	3.51	0.56	9.4	5
1	7.8	0.880	0.00	2.6	0.098	25.0	67.0	0.99680	3.20	0.68	9.8	5
2	7.8	0.760	0.04	2.3	0.092	15.0	54.0	0.99700	3.26	0.65	9.8	5
3	11.2	0.280	0.56	1.9	0.075	17.0	60.0	0.99800	3.16	0.58	9.8	6
4	7.4	0.700	0.00	1.9	0.076	11.0	34.0	0.99780	3.51	0.56	9.4	5
...	...	...	...	...	...	...	...	...	...	...	...	...
1594	6.2	0.600	0.08	2.0	0.090	32.0	44.0	0.99490	3.45	0.58	10.5	5
1595	5.9	0.550	0.10	2.2	0.062	39.0	51.0	0.99512	3.52	0.76	11.2	6
1596	6.3	0.510	0.13	2.3	0.076	29.0	40.0	0.99574	3.42	0.75	11.0	6
1597	5.9	0.645	0.12	2.0	0.075	32.0	44.0	0.99547	3.57	0.71	10.2	5
1598	6.0	0.310	0.47	3.6	0.067	18.0	42.0	0.99549	3.39	0.66	11.0	6

1599 rows × 12 columns



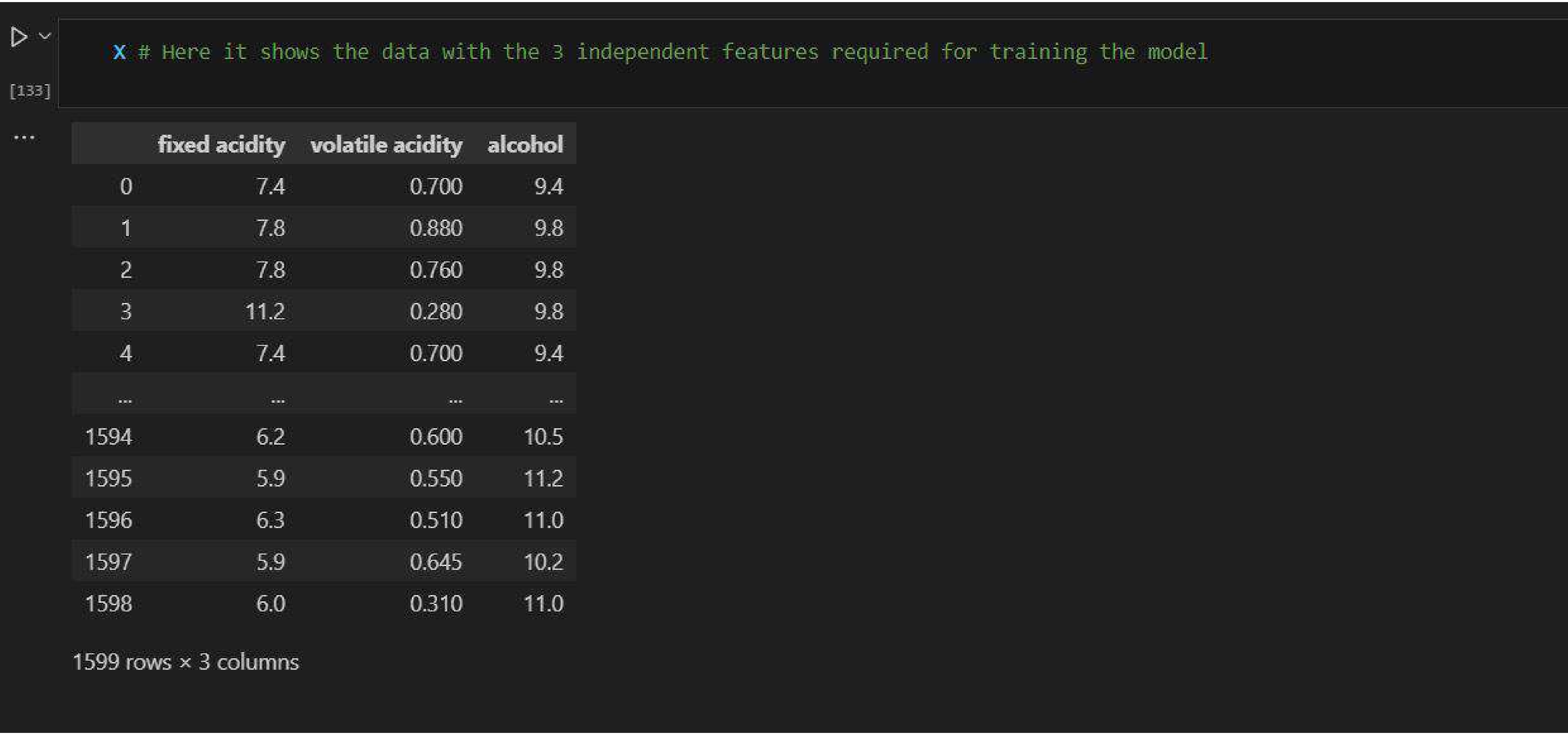
```python
df.head()
# Here we get the top 5 head values from the whole data
```

[131] Python

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality
0	7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	0.56	9.4	5
1	7.8	0.88	0.00	2.6	0.098	25.0	67.0	0.9968	3.20	0.68	9.8	5
2	7.8	0.76	0.04	2.3	0.092	15.0	54.0	0.9970	3.26	0.65	9.8	5
3	11.2	0.28	0.56	1.9	0.075	17.0	60.0	0.9980	3.16	0.58	9.8	6
4	7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	0.56	9.4	5

```python
# Now we start seperating the independent and dependent features
# We select fixed acidity, volatile acidity and the quality columns from the whole dataset
X = df[['fixed acidity', 'volatile acidity', 'alcohol']]
y = df['quality']
```

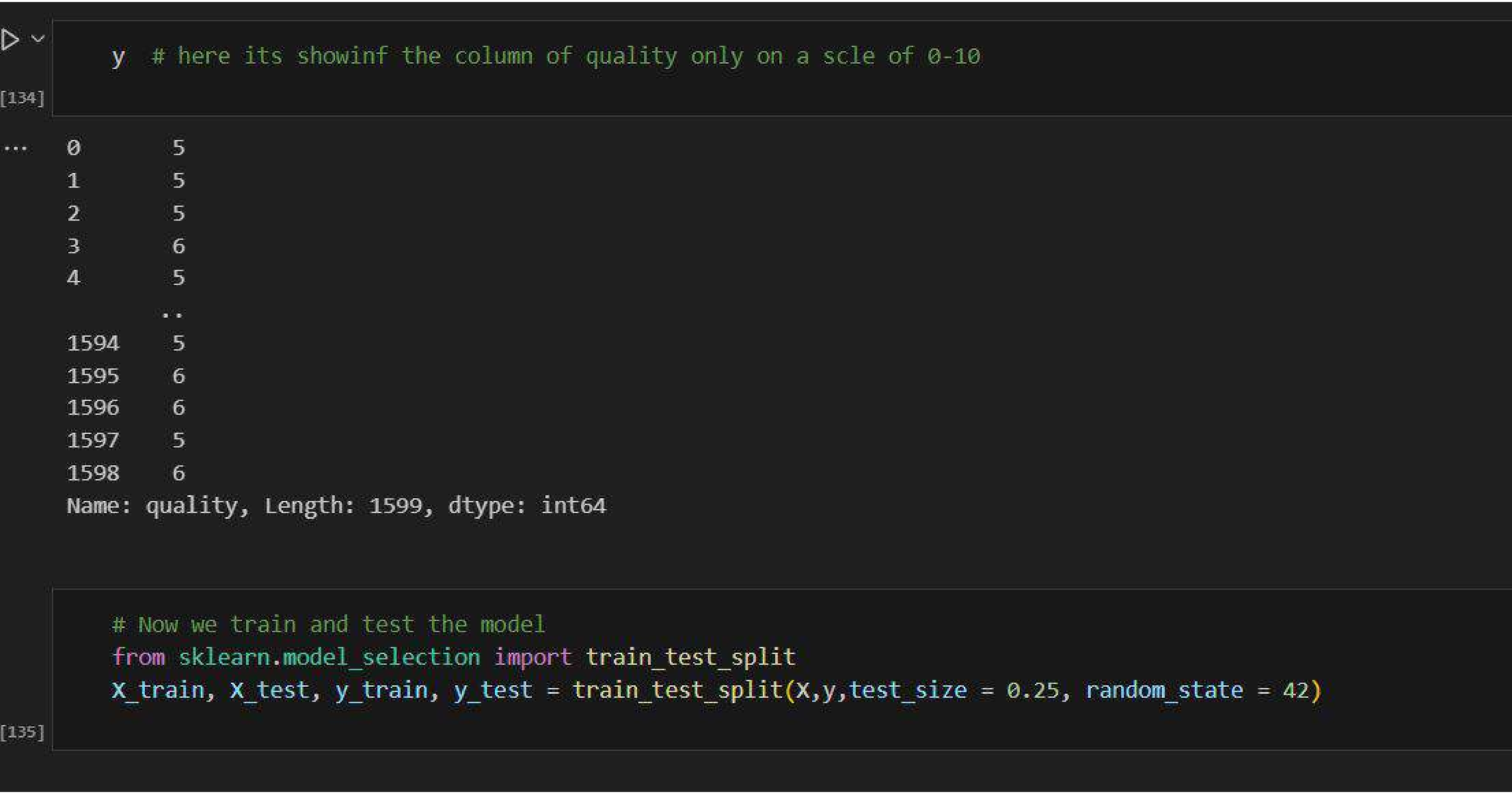
[132] Python



```
X # Here it shows the data with the 3 independent features required for training the model
```

	fixed acidity	volatile acidity	alcohol
0	7.4	0.700	9.4
1	7.8	0.880	9.8
2	7.8	0.760	9.8
3	11.2	0.280	9.8
4	7.4	0.700	9.4
...	...	...	...
1594	6.2	0.600	10.5
1595	5.9	0.550	11.2
1596	6.3	0.510	11.0
1597	5.9	0.645	10.2
1598	6.0	0.310	11.0

1599 rows × 3 columns



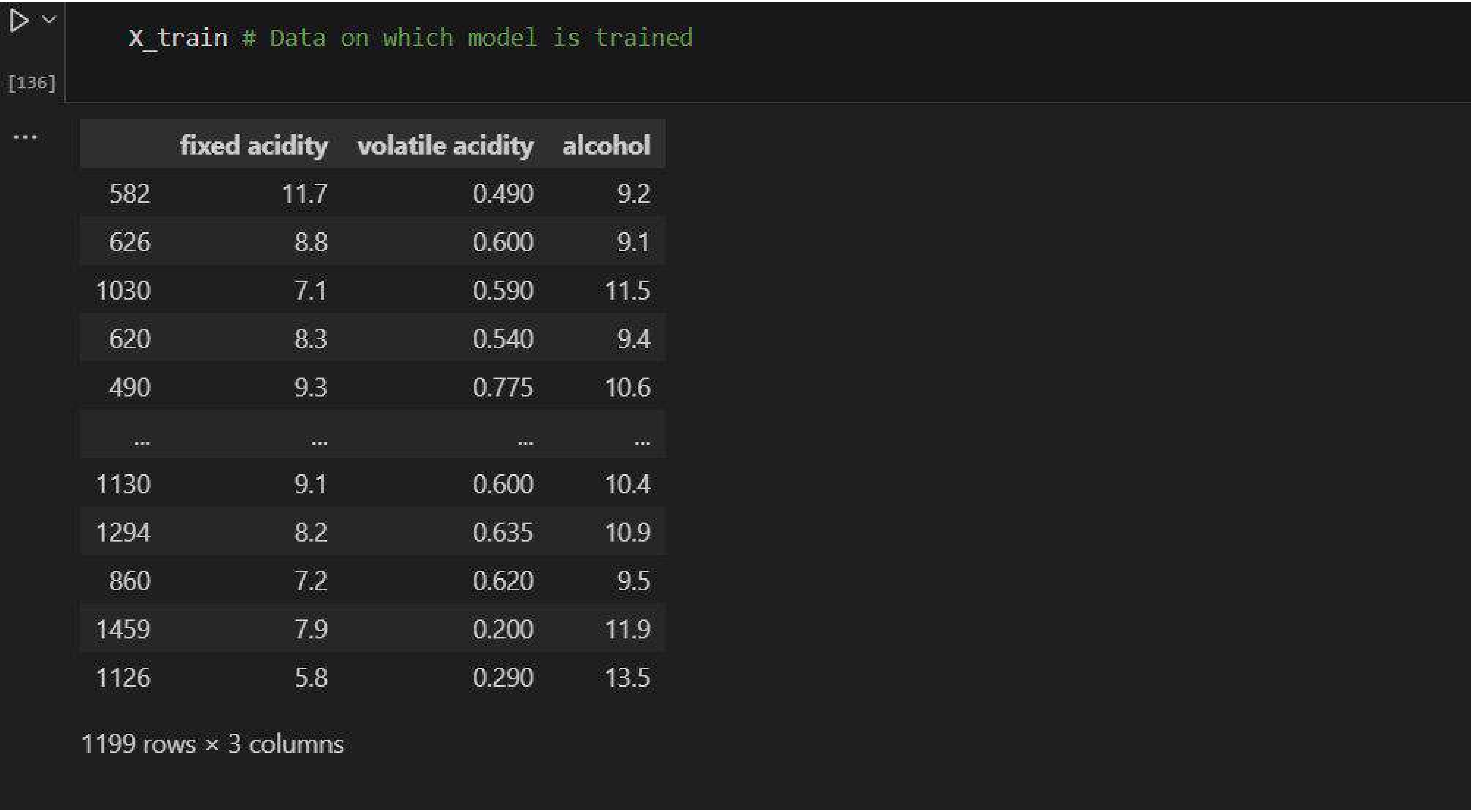
```
y  # here its showinf the column of quality only on a scle of 0-10
```

[134]

```
0       5
1       5
2       5
3       6
4       5
       ..
1594    5
1595    6
1596    6
1597    5
1598    6
Name: quality, Length: 1599, dtype: int64
```

```
# Now we train and test the model
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X,y,test_size = 0.25, random_state = 42)
```

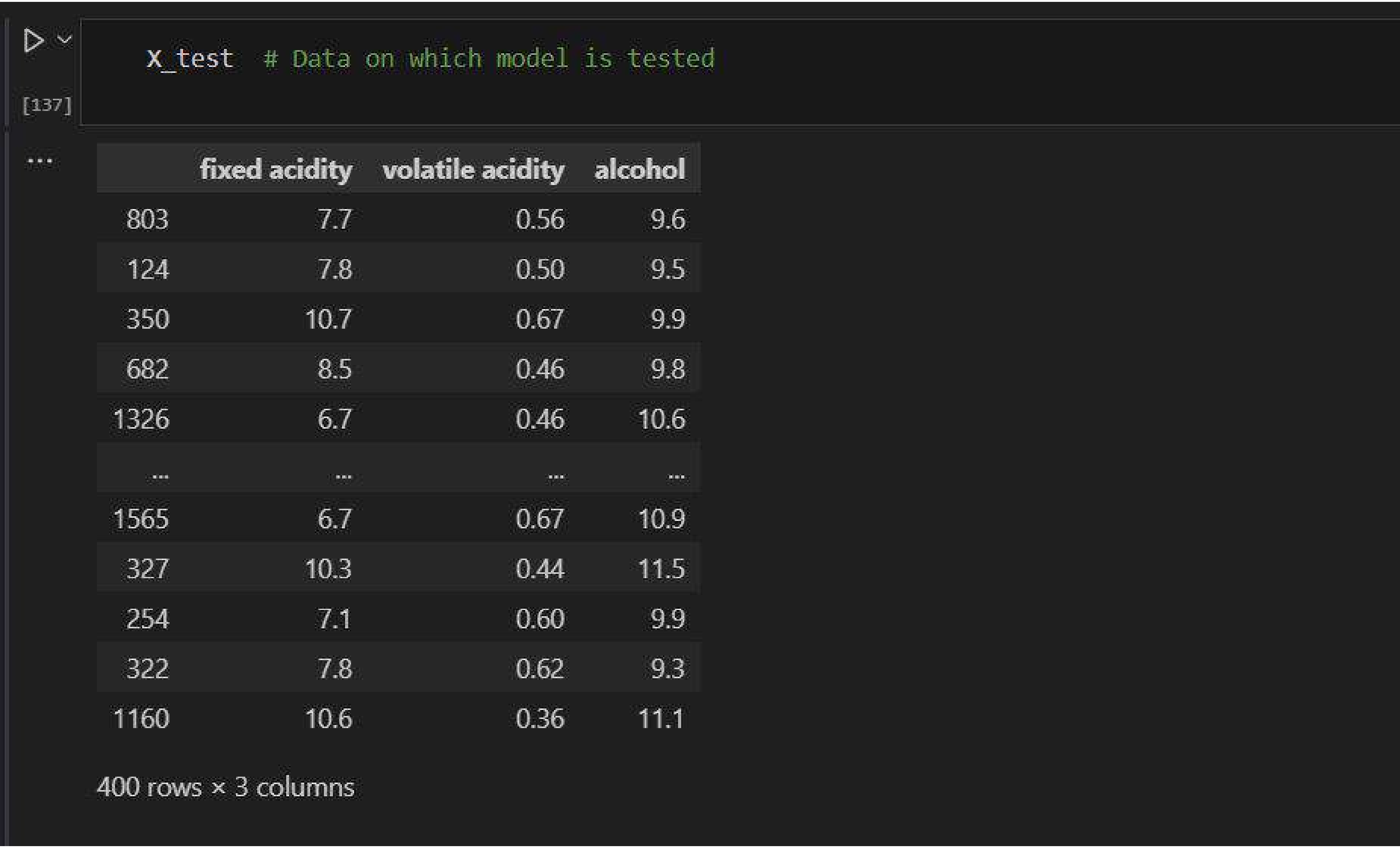
[135]



```
X_train # Data on which model is trained
```

	fixed acidity	volatile acidity	alcohol
582	11.7	0.490	9.2
626	8.8	0.600	9.1
1030	7.1	0.590	11.5
620	8.3	0.540	9.4
490	9.3	0.775	10.6
...	...	...	...
1130	9.1	0.600	10.4
1294	8.2	0.635	10.9
860	7.2	0.620	9.5
1459	7.9	0.200	11.9
1126	5.8	0.290	13.5

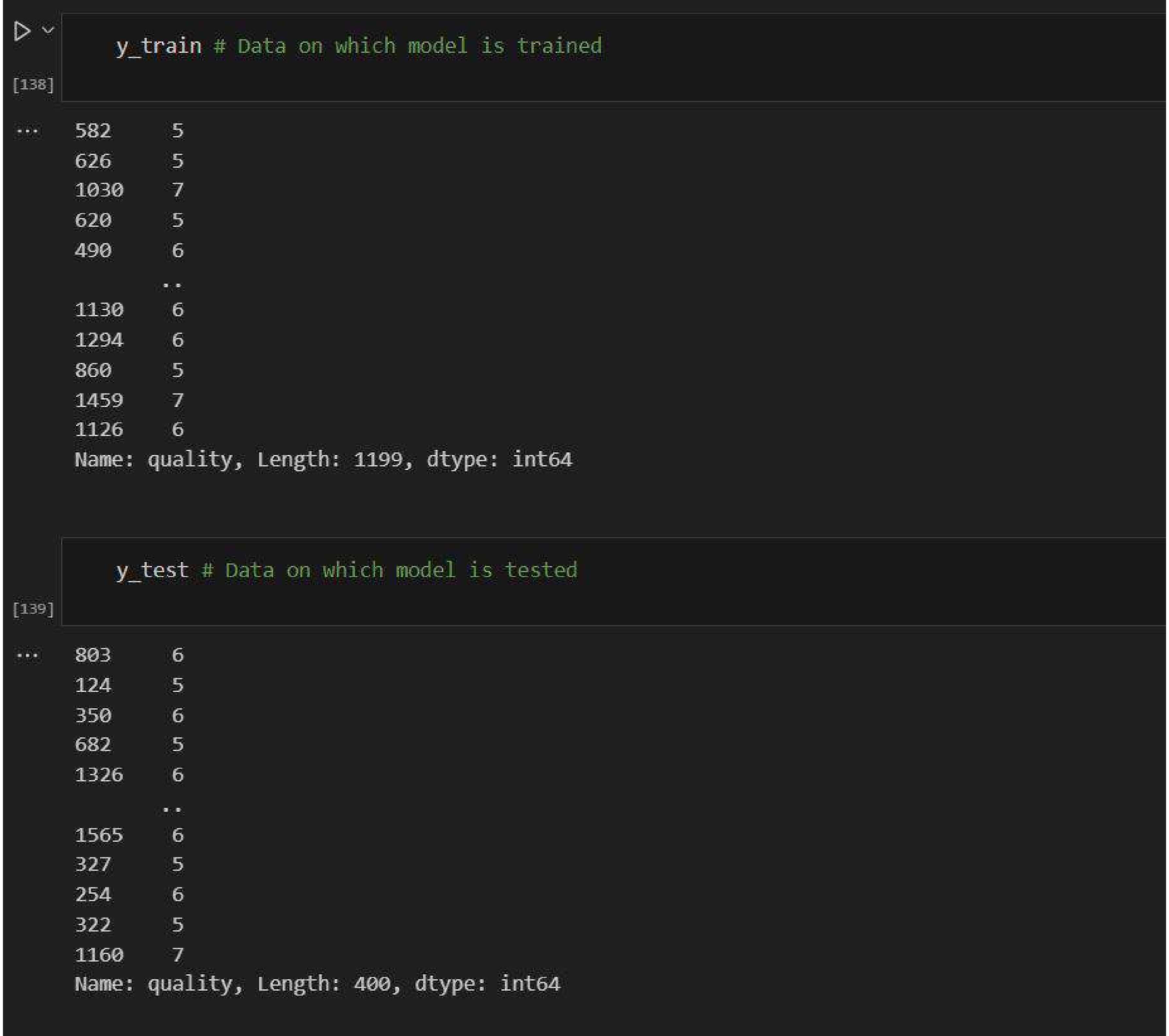
1199 rows × 3 columns



```
X_test  # Data on which model is tested
```

	fixed acidity	volatile acidity	alcohol
803	7.7	0.56	9.6
124	7.8	0.50	9.5
350	10.7	0.67	9.9
682	8.5	0.46	9.8
1326	6.7	0.46	10.6
...	...	...	...
1565	6.7	0.67	10.9
327	10.3	0.44	11.5
254	7.1	0.60	9.9
322	7.8	0.62	9.3
1160	10.6	0.36	11.1

400 rows × 3 columns

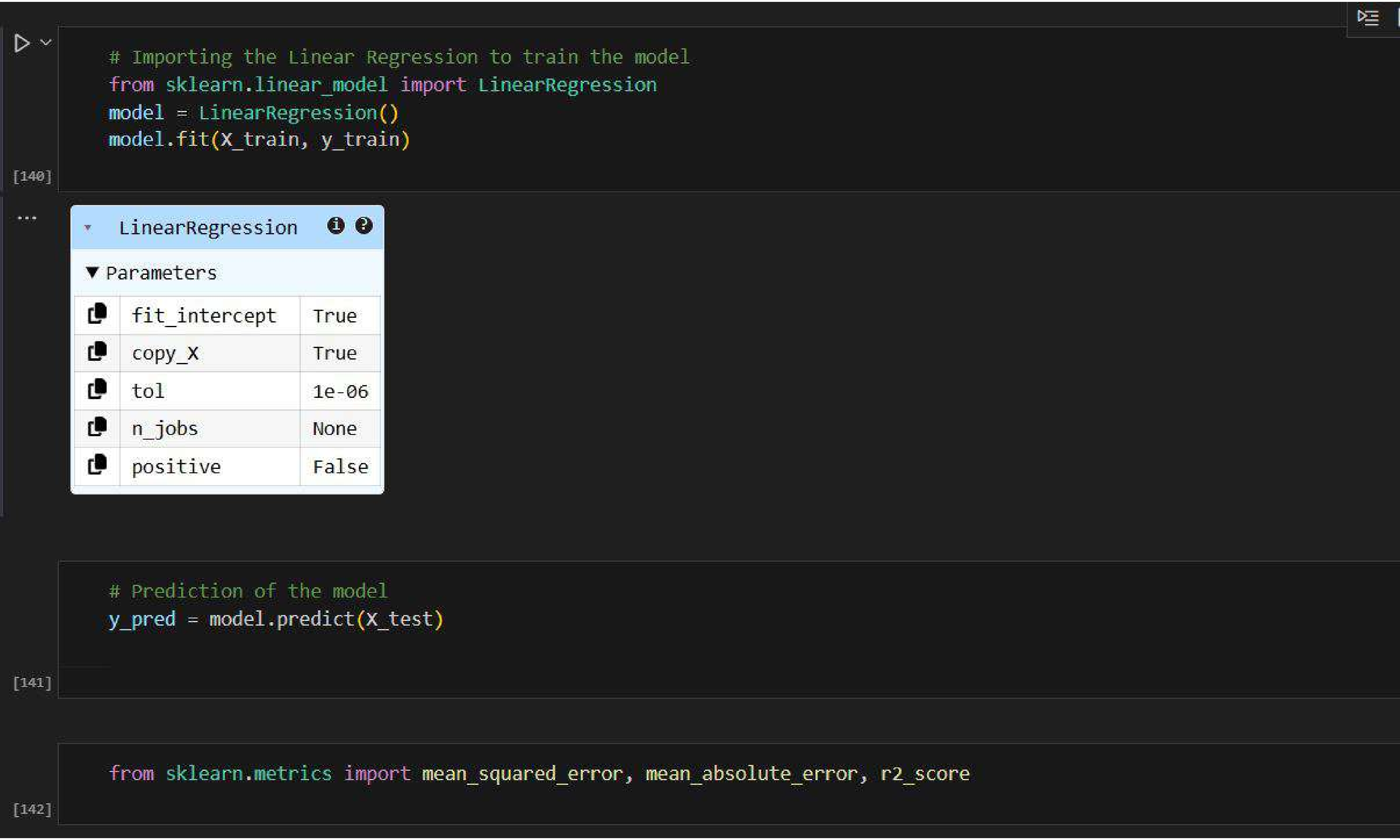


```
y_train # Data on which model is trained
```

```
582     5
626     5
1030    7
620     5
490     6
       ..
1130    6
1294    6
860     5
1459    7
1126    6
Name: quality, Length: 1199, dtype: int64
```

```
y_test # Data on which model is tested
```

```
803     6
124     5
350     6
682     5
1326    6
       ..
1565    6
327     5
254     6
322     5
1160    7
Name: quality, Length: 400, dtype: int64
```



```python
# Importing the Linear Regression to train the model
from sklearn.linear_model import LinearRegression
model = LinearRegression()
model.fit(X_train, y_train)
```

[140]

LinearRegression

▼ Parameters

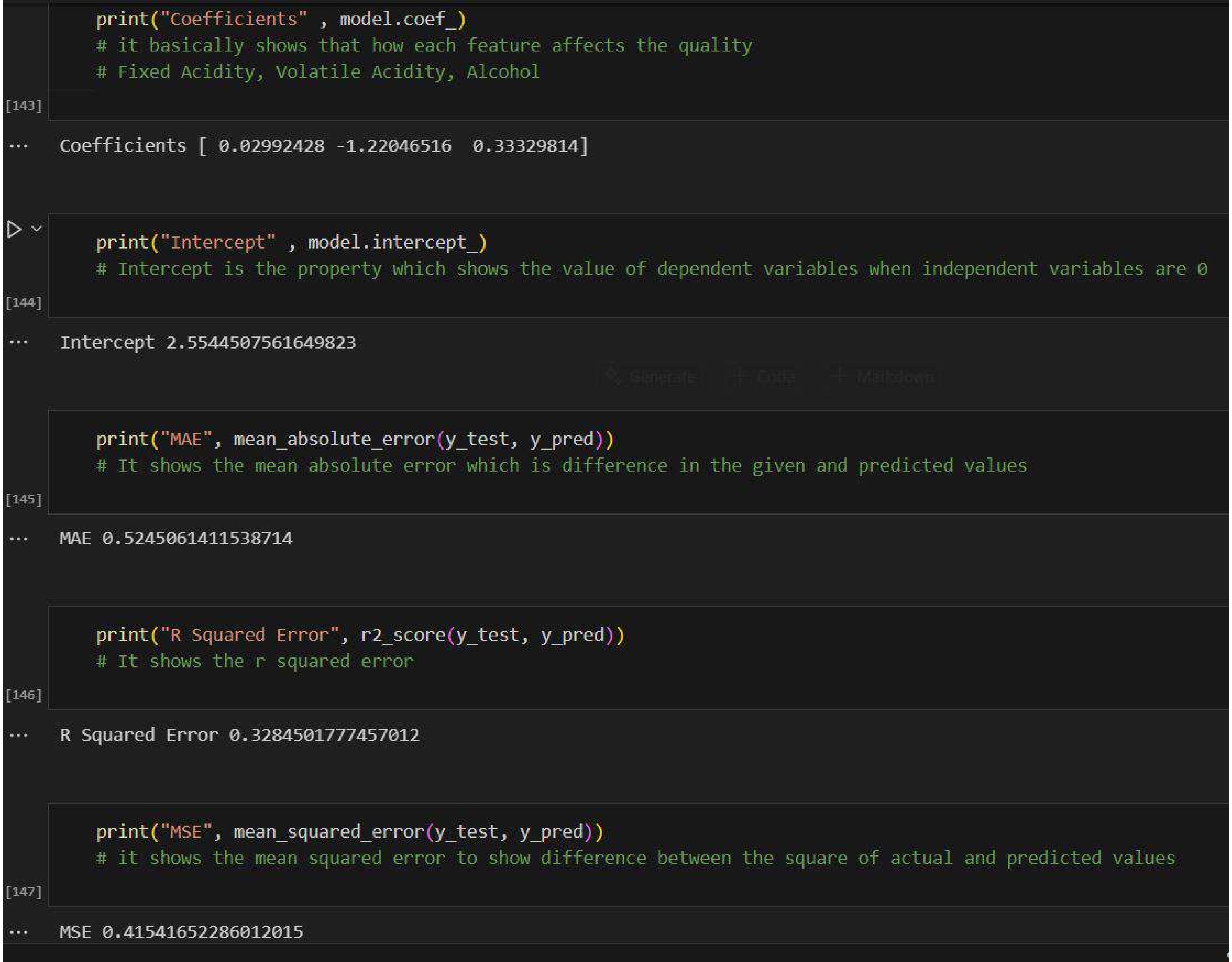
	fit_intercept	True
	copy_X	True
	tol	1e-06
	n_jobs	None
	positive	False

```python
# Prediction of the model
y_pred = model.predict(X_test)
```

[141]

```python
from sklearn.metrics import mean_squared_error, mean_absolute_error, r2_score
```

[142]



```
print("Coefficients" , model.coef_)
# it basically shows that how each feature affects the quality
# Fixed Acidity, Volatile Acidity, Alcohol
```

```
Coefficients [ 0.02992428 -1.22046516  0.33329814]
```

```
print("Intercept" , model.intercept_)
# Intercept is the property which shows the value of dependent variables when independent variables are 0
```

```
Intercept 2.5544507561649823
```

```
print("MAE", mean_absolute_error(y_test, y_pred))
# It shows the mean absolute error which is difference in the given and predicted values
```

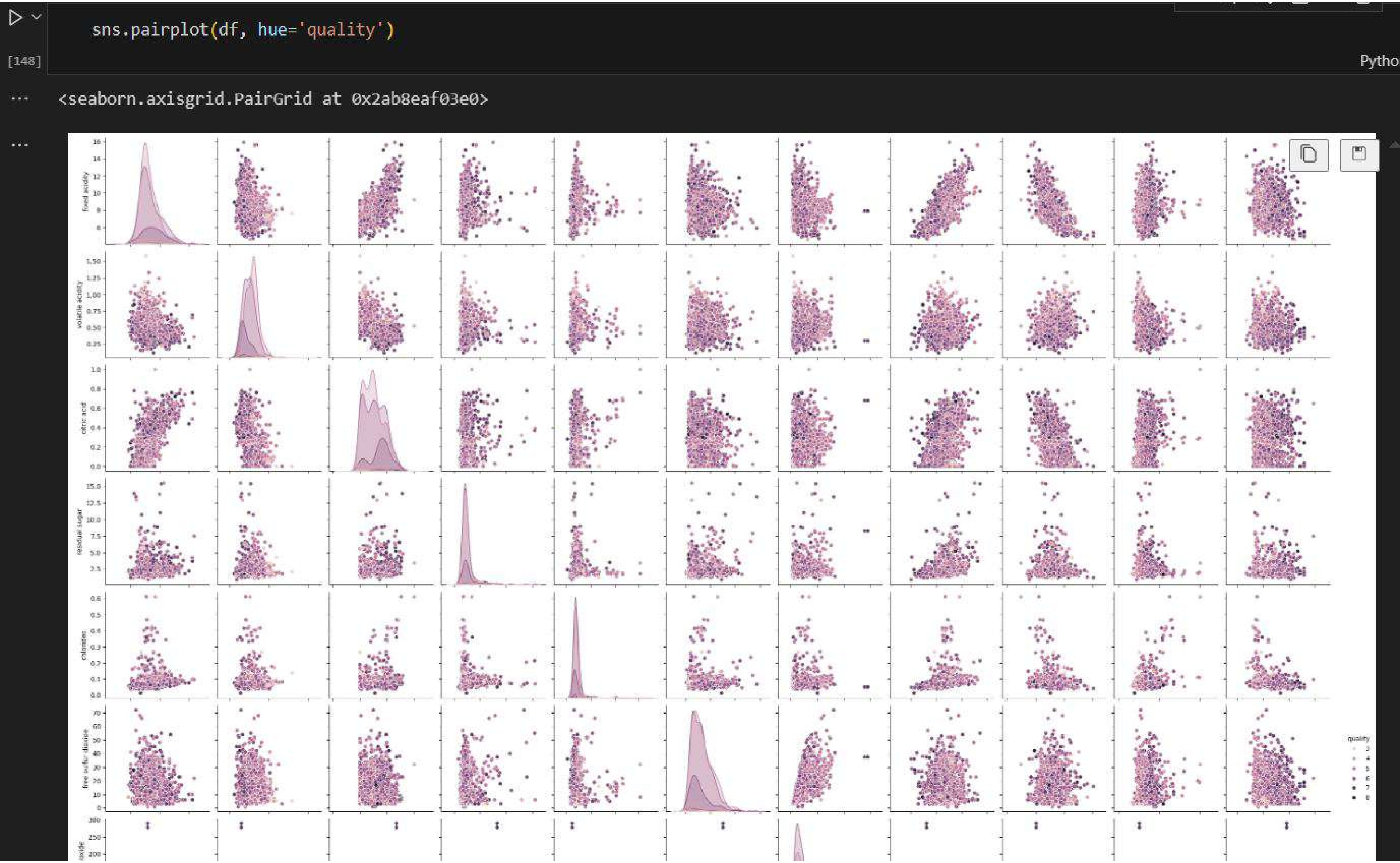
```
MAE 0.5245061411538714
```

```
print("R Squared Error", r2_score(y_test, y_pred))
# It shows the r squared error
```

```
R Squared Error 0.3284501777457012
```

```
print("MSE", mean_squared_error(y_test, y_pred))
# it shows the mean squared error to show difference between the square of actual and predicted values
```

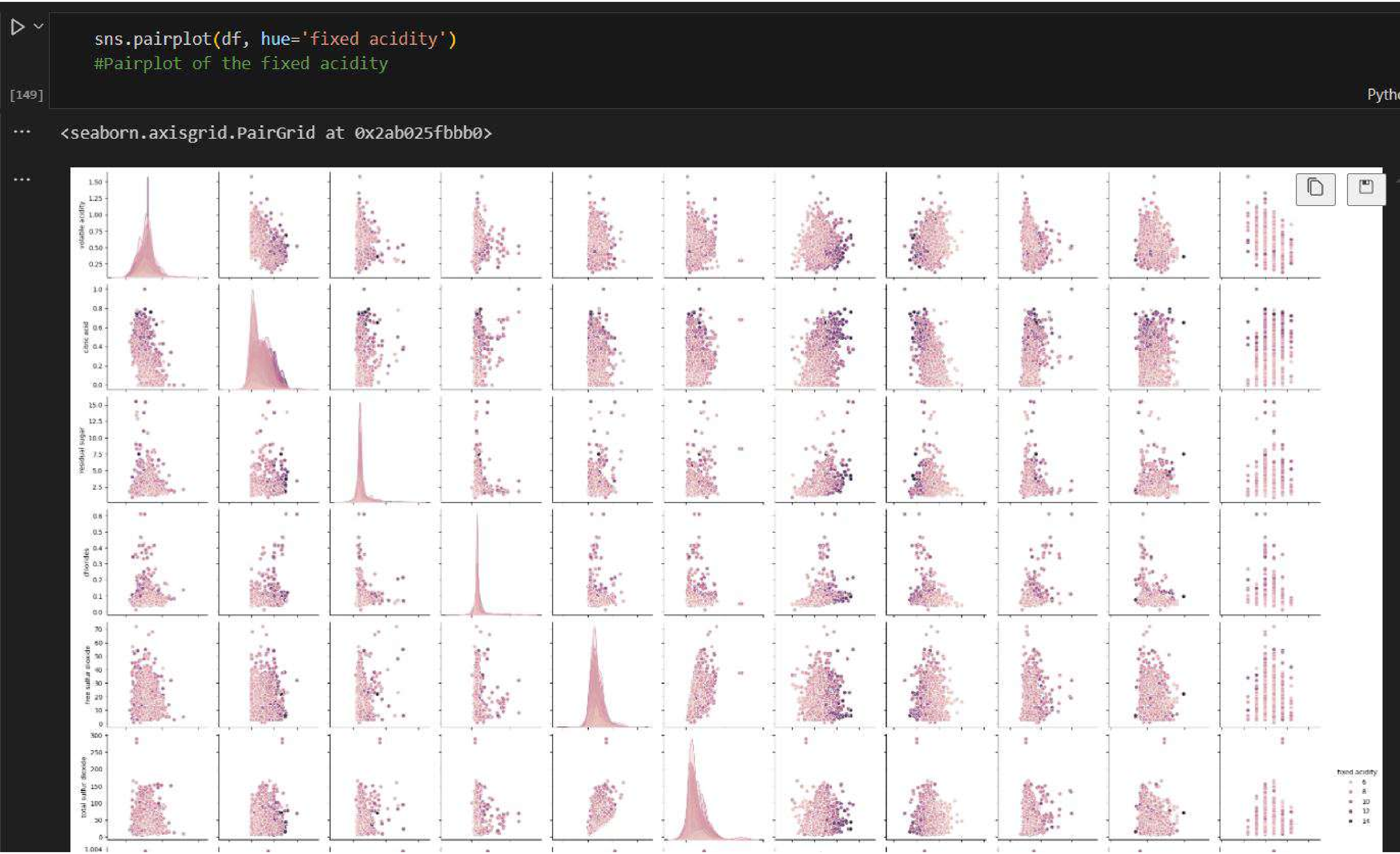
```
MSE 0.41541652286012015
```



[148]

```
sns.pairplot(df, hue='quality')
```

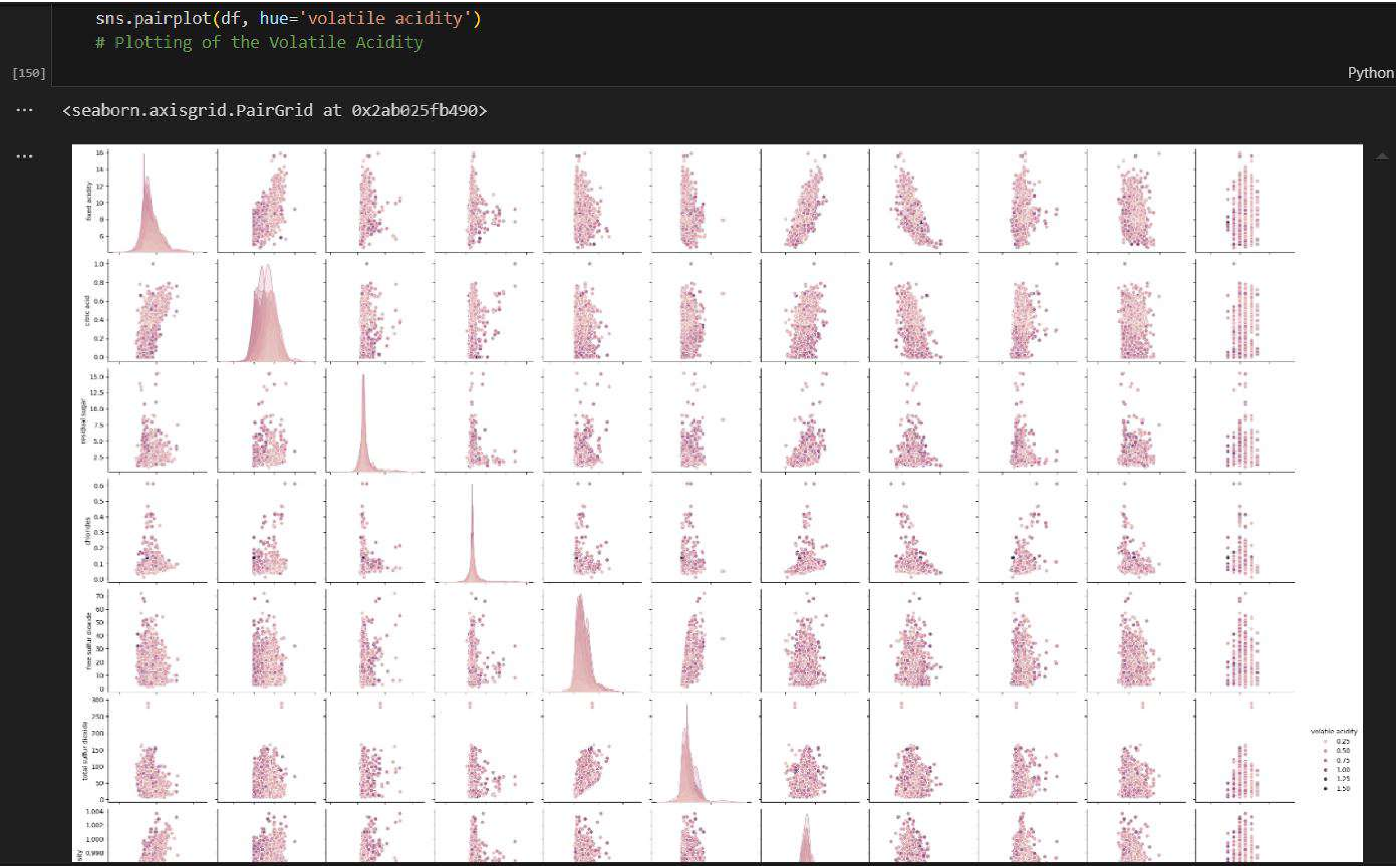
```
<seaborn.axisgrid.PairGrid at 0x2ab8eaf03e0>
```



```
sns.pairplot(df, hue='fixed acidity')
#Pairplot of the fixed acidity
```

[149]

```
<seaborn.axisgrid.PairGrid at 0x2ab025fbbb0>
```

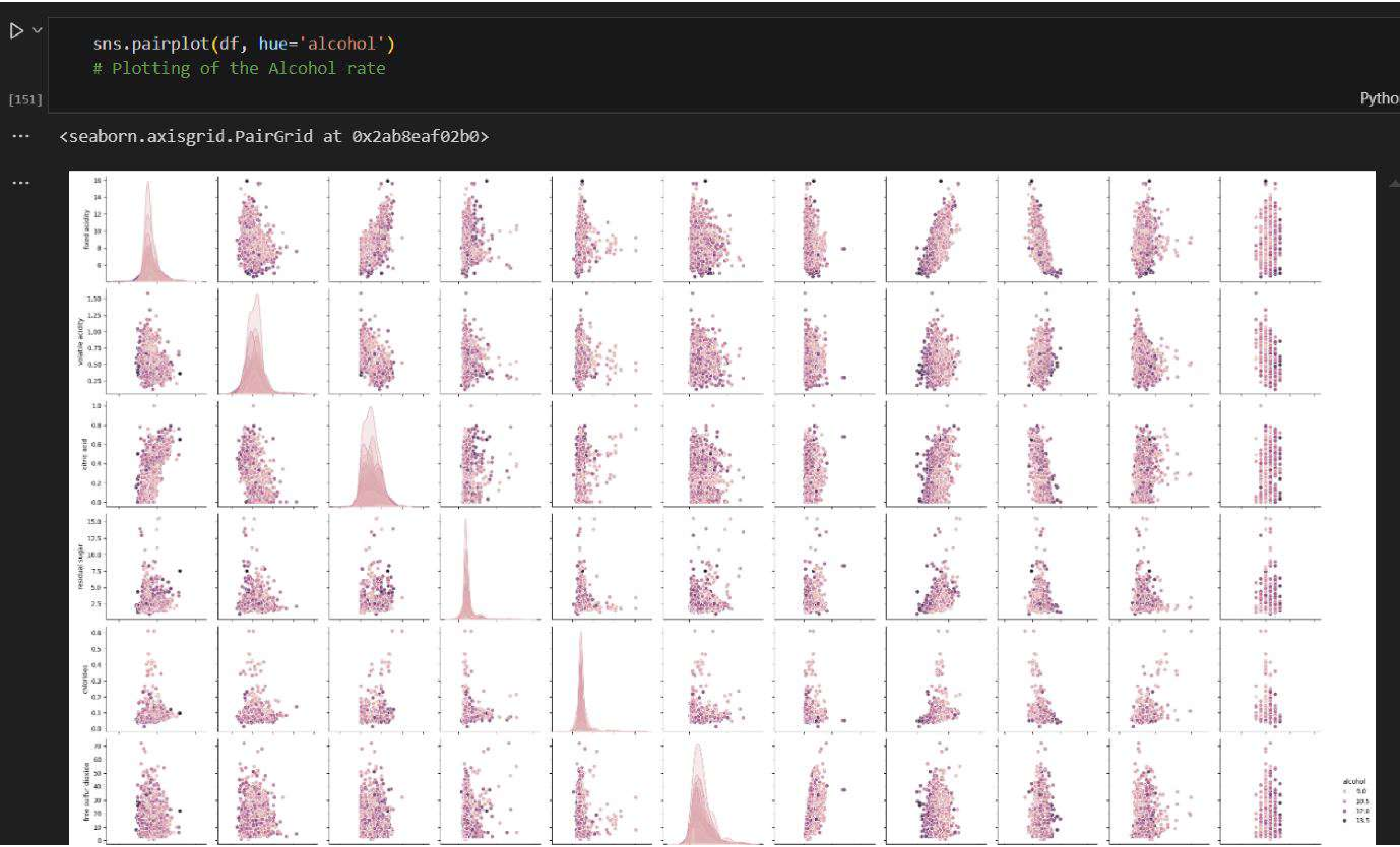


```
sns.pairplot(df, hue='volatile acidity')
# Plotting of the Volatile Acidity
```

[150]

Python

```
<seaborn.axisgrid.PairGrid at 0x2ab025fb490>
```

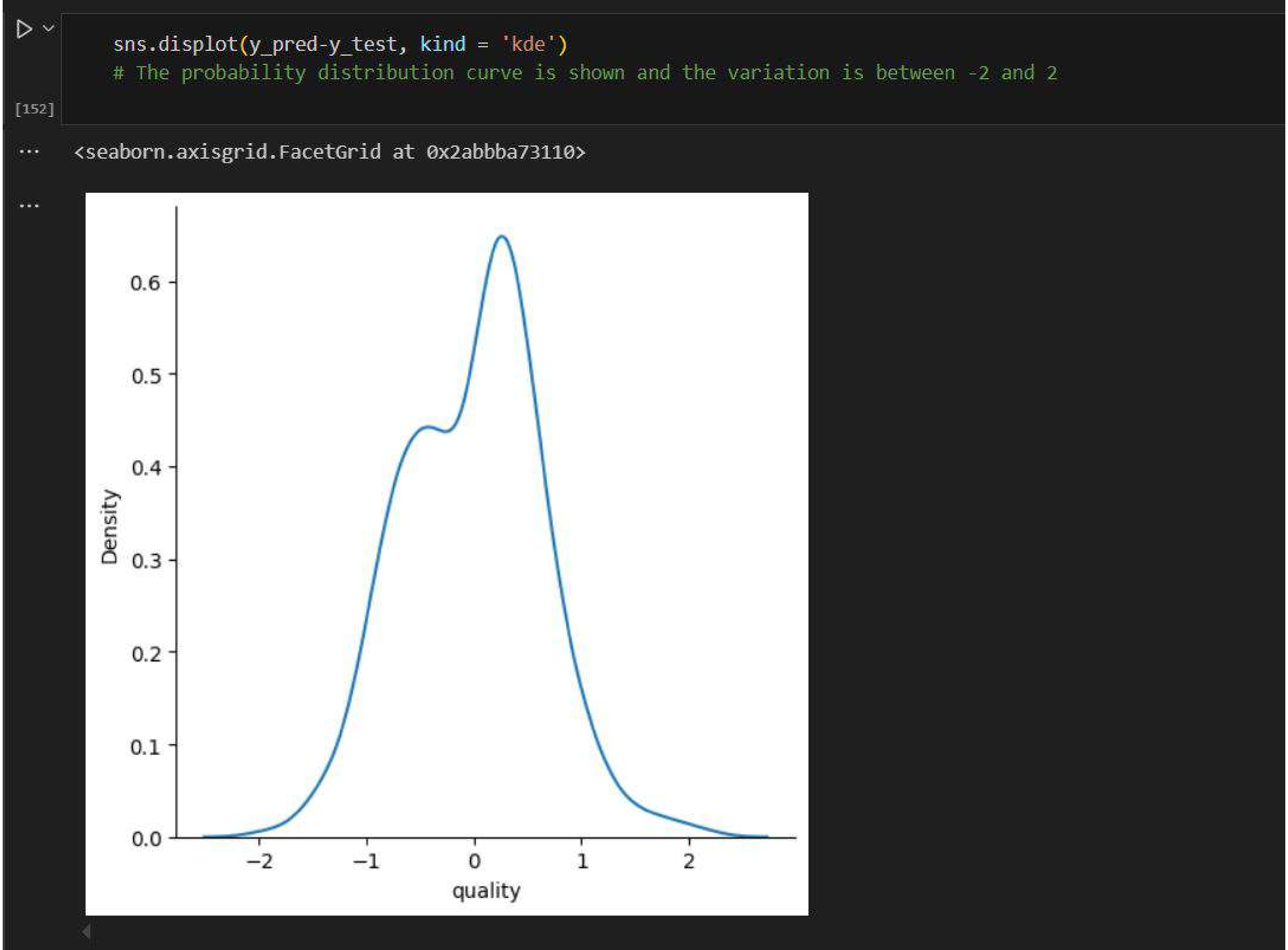


```
sns.pairplot(df, hue='alcohol')
# Plotting of the Alcohol rate
```

[151]

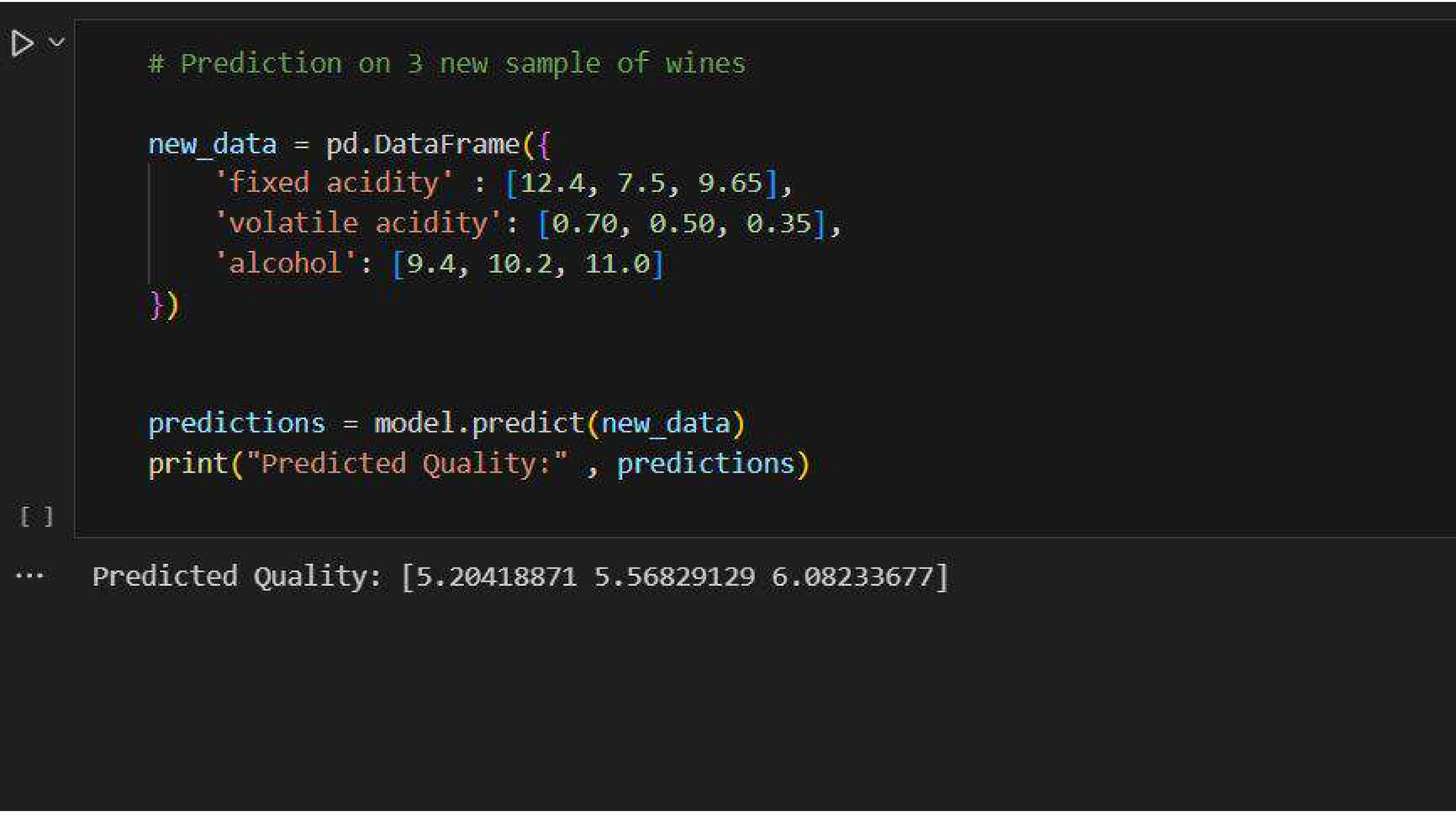
Pytho

```
<seaborn.axisgrid.PairGrid at 0x2ab8eaf02b0>
```



```
[152]
sns.displot(y_pred-y_test, kind = 'kde')
# The probability distribution curve is shown and the variation is between -2 and 2
```

```
<seaborn.axisgrid.FacetGrid at 0x2abbba73110>
```



```
# Prediction on 3 new sample of wines

new_data = pd.DataFrame({
    'fixed acidity' : [12.4, 7.5, 9.65],
    'volatile acidity': [0.70, 0.50, 0.35],
    'alcohol': [9.4, 10.2, 11.0]
})


predictions = model.predict(new_data)
print("Predicted Quality:" , predictions)
```

```
Predicted Quality: [5.20418871 5.56829129 6.08233677]
```