DATA ANALYSIS

Python Project with Pandas v.2

Food

Mirko Rossi

start2impact UNIVERSITY

Introduction

- Wine is a highly appreciated product in Italy and internationally, making it an excellent investment for aspiring entrepreneurs.
- To start an e-commerce, you must have a carefully selected list of wines among the tens of thousands of existing types.
- Concentrating on a specific number and types of wine can ensure a high turnover in the warehouse and short return times on the invested capital.
- This project aims to produce a catalogue of wines from small producers using a dataset of approximately 130,000 wines.

Introduction

The public wine dataset, in CSV format, is available on Kaggle.

I used Jupyter as a development environment and the Pandas, SeaBorn, and Matlplotlib libraries.

The work was divided into three phases:

- Data Cleaning
- Data Analysis and Visualisation
- Creation of the Catalogue

Data Cleaning

Study of the dataset - head method

```
# Dataset visualisation
df.head(5)
```

	Unnamed: 0	country	description	designation	points	price	province	region_1	region_2	taster_name	taster_twitter_handle	title	variety	winery
0	0	Italy	Aromas include tropical fruit, broom, brimston...	Vulkà Bianco	87	NaN	Sicily & Sardinia	Etna	NaN	Kerin O'Keefe	@kerinokeefe	Nicosia 2013 Vulkà Bianco (Etna)	White Blend	Nicosia
1	1	Portugal	This is ripe and fruity, a wine that is smooth...	Avidagos	87	15.0	Douro	NaN	NaN	Roger Voss	@vossroger	Quinta dos Avidagos 2011 Avidagos Red (Douro)	Portuguese Red	Quinta dos Avidagos
2	2	US	Tart and snappy, the flavors of lime flesh and...	NaN	87	14.0	Oregon	Willamette Valley	Willamette Valley	Paul Gregutt	@paulgwine	Rainstorm 2013 Pinot Gris (Willamette Valley)	Pinot Gris	Rainstorm
3	3	US	Pineapple rind, lemon pith and orange blossom ...	Reserve Late Harvest	87	13.0	Michigan	Lake Michigan Shore	NaN	Alexander Peartree	NaN	St. Julian 2013 Reserve Late Harvest Riesling ...	Riesling	St. Julian
4	4	US	Much like the regular bottling from 2012, this...	Vintner's Reserve Wild Child Block	87	65.0	Oregon	Willamette Valley	Willamette Valley	Paul Gregutt	@paulgwine	Sweet Cheeks 2012 Vintner's Reserve Wild Child...	Pinot Noir	Sweet Cheeks

Study of the dataset - head method

Data Cleaning first requires understanding the content of the field under study.

You can observe the presence of redundant columns (*Unnamed: 0, region_2*) or non-essential columns for catalogue purposes *(taster_name, taster_twitter_handle),* along with null values in the price field, which is fundamental for creating the catalogue.

	Unnamed: 0	country	description	designation	points	price	province	region_1	region_2	taster_name	taster_twitter_handle	title	variety	winery
0	0	Italy	Aromas include tropical fruit, broom, brimston...	Vulkà Bianco	87	NaN	Sicily & Sardinia	Etna	NaN	Kerin O'Keefe	@kerinokeefe	Nicosia 2013 Vulkà Bianco (Etna)	White Blend	Nicosia
1	1	Portugal	This is ripe and fruity, a wine that is smooth...	Avidagos	87	15.0	Douro	NaN	NaN	Roger Voss	@vossroger	Quinta dos Avidagos 2011 Avidagos Red (Douro)	Portuguese Red	Quinta dos Avidagos
2	2	US	Tart and snappy, the flavors of lime flesh and...	NaN	87	14.0	Oregon	Willamette Valley	Willamette Valley	Paul Gregutt	@paulgwine	Rainstorm 2013 Pinot Gris (Willamette Valley)	Pinot Gris	Rainstorm
3	3	US	Pineapple rind, lemon pith and orange blossom ...	Reserve Late Harvest	87	13.0	Michigan	Lake Michigan Shore	NaN	Alexander Peartree	NaN	St. Julian 2013 Reserve Late Harvest Riesling ...	Riesling	St. Julian
4	4	US	Much like the regular bottling from 2012, this...	Vintner's Reserve Wild Child Block	87	65.0	Oregon	Willamette Valley	Willamette Valley	Paul Gregutt	@paulgwine	Sweet Cheeks 2012 Vintner's Reserve Wild Child...	Pinot Noir	Sweet Cheeks

Study of the dataset - info method

The info method returns a list of field labels, which integrates with the observation that can be made by the head method.

You can observe that the 129,971 rows have empty fields which percentage can be calculated.

```
# Columns types and non-null count
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 129971 entries, 0 to 129970
Data columns (total 14 columns):
 #   Column                 Non-Null Count   Dtype  
---  ------                 --------------   -----  
 0   Unnamed: 0             129971 non-null  int64  
 1   country                129908 non-null  object 
 2   description            129971 non-null  object 
 3   designation            92506 non-null   object 
 4   points                 129971 non-null  int64  
 5   price                  120975 non-null  float64
 6   province               129908 non-null  object 
 7   region_1               108724 non-null  object 
 8   region_2               50511 non-null   object 
 9   taster_name            103727 non-null  object 
 10  taster_twitter_handle  98758 non-null   object 
 11  title                  129971 non-null  object 
 12  variety                129970 non-null  object 
 13  winery                 129971 non-null  object 
dtypes: float64(1), int64(2), object(11)
memory usage: 13.9+ MB
```

Study of the dataset - percentage of null values

Since the *country* and *price* fields are fundamental for creating a catalogue, all records without these fields must be deleted.

The *designation* field represents the commercial name of the wines, while the *title* field is a combination of *winery* + *year* + *designation* + *region 1* or province.

To avoid redundancy and maintain consistency, I will try to extract the missing *designation* values from the *title* as well as the *year,* which I insert into a new field called *year*.

Finally, I will delete the *title* field.

```
# Calculate the number of null values for each column
null_counts = df.isnull().sum()

# Calculate the total values in each column
total_counts = df.shape[0]

# Calculate the percentage of null values
null_percentage = (null_counts / total_counts * 100).round(2).astype(str) + '%'

# Create a new DataFrame with the results
null_info = pd.DataFrame({'Null Count': null_counts, 'Percentage': null_percentage})

print(null_info)
```

	Null	Percentage
Unnamed: 0	0	0.0%
country	63	0.05%
description	0	[illegible]
designation	37465	28.83%
points	0	[illegible]
price	8996	6.92%
province	63	0.05%
region_1	21247	16.35%
region_2	79460	61.14%
taster_name	26244	20.19%
taster_twitter_handle	31213	[illegible]
title	0	0.0%
variety	1	[illegible]
winery	0	0.0%

Extraction of the year from *title*

As I mentioned earlier, I extract the year using regular expressions and insert it into a new field called *year* .

```python
# Define a function to extract the year from a string using regex
def extract_year_from_string(text):
    year = re.findall(r'\b\d{4}\b', text)  # Find all patterns YYYY in the text
    if year:
        return int(year[0])  # Take the first found year as an integer
    else:
        return None

# Apply the function to the 'title' field to extract the year
df['year'] = df['title'].apply(extract_year_from_string)

# Handle NaN and infinite values by replacing them with 0
df['year'].fillna(0, inplace=True)

# Convert the "year" column into integer values
df['year'] = df['year'].astype(int)
```

Retrieving missing values of *designation*

Ale retrieve missing values of *designation* from *title* can be done using regular expressions.

If the *title* field does not follow the structure " *winery + year + designation + region 1 or province,“ a* match with the commercial name to the variety (e.g., Nebbiolo, Pinot Gris, Chardonnay) can be attempted, without compromising the integrity of the information dramatically.

```python
# Function to copy the value from "title" to "designation" if "designation" is null
def extract_designation_from_title(row):
    if pd.isna(row['designation']):
        # Use a regular expression to extract "designation"
        result = re.search(r'\d{4}\s(.+?)\s\(', row['title'])
        if result:
            row['designation'] = result.group(1)
        else:
            # If extraction from "title" is null, copy the content from "variety"
            row['designation'] = row['variety']
    return row

# Apply the function to each row of the DataFrame
df = df.apply(extract_designation_from_title, axis=1)
```

Deleting and reordering fields

I will delete the non-essential fields and reorder them for easier reading.

```
# Dropping unnecessary columns
df.drop(['Unnamed: 0', 'province', 'region_1', 'region_2',
         'title', 'taster_name', 'taster_twitter_handle'],
        axis=1, inplace=True)
```

```
# Define the desired order of columns
new_column_order = ['country', 'description', 'winery', 'variety',
                    'designation', 'year', 'points', 'price']

# Select the columns in the new order
df = df[new_column_order]
```

Dataset after cleaning

```
# Dataset visualisation
df.head(5)
```

	country	description	designation	points	price	variety	winery
0	Italy	Aromas include tropical fruit, broom, brimston...	Vulkà Bianco	87	NaN	White Blend	Nicosia
1	Portugal	This is ripe and fruity, a wine that is smooth...	Avidagos	87	15.0	Portuguese Red	Quinta dos Avidagos
2	US	Tart and snappy, the flavors of lime flesh and...	Pinot Gris	87	14.0	Pinot Gris	Rainstorm
3	US	Pineapple rind, lemon pith and orange blossom ...	Reserve Late Harvest	87	13.0	Riesling	St. Julian
4	US	Much like the regular bottling from 2012, this...	Vintner's Reserve Wild Child Block	87	65.0	Pinot Noir	Sweet Cheeks

Dataset after cleaning

Checking the *null* fields after the cleaning operations reveals an empty record in the *designation* likely linked to the empty record in *variety*.

	Null Count	Percentage
country	63	0.05%
description	0	0.0%
designation	1	0.0%
points	0	0.0%
price	8996	6.92%
variety	1	0.0%
winery	0	0.0%
year	0	0.0%

Record with empty *variety*

```
filtered_df = df[df['designation'].isnull()]
filtered_df
```

	country	description	winery	variety	designation	year	points	price
86909	Chile	A chalky, dusty mouthfeel nicely balances this...	Carmen	NaN	NaN	1999	88	17.0

Deleting records with *price* and *designation* nulls

As expected the only record with no *designation* value left is one that doesn't even have values in *variety* , making it impossible to characterize.

Finally, I eliminated all the lines that have empty fields in *price* and *designation,* which are two fundamental fields for preparing the catalogue.

```
filtered_df = df[df['designation'].isnull()]
filtered_df
```

	country	description	winery	variety	designation	year	points	price
86909	Chile	A chalky, dusty mouthfeel nicely balances this...	Carmen	NaN	NaN	1999	88	17.0

```
# Delete rows with at least one null value in 'price' and 'designation' columns
df.dropna(subset=['price', 'designation'], inplace=True)
```

Dataset after cleaning is complete

At the end of the Data Cleaning phase, there are no empty values in the fundamental fields.

The *country* field is an important field but eliminating these 59 wines whose price and known commercial name, would result in an unnecessary loss of wines from the catalogue.

	Null Count	Percentage
country	59	0.05%
description	0	0.0%
designation	0	0.0%
points	0	0.0%
price	0	0.0%
variety	0	0.0%
winery	0	0.0%
year	0	0.0%

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 120974 entries, 1 to 129970
Data columns (total 7 columns):
 #   Column       Non-Null Count   Dtype
---  ------       --------------   -----
 0   country      120915 non-null  object
 1   winery       120974 non-null  object
 2   variety      120974 non-null  object
 3   designation  120974 non-null  object
 4   year         120974 non-null  int64
 5   points       120974 non-null  int64
 6   price        120974 non-null  float64
dtypes: float64(1), int64(2), object(4)
memory usage: 7.4+ MB
```

Description of the dataset

FIELD	DESCRIPTION
Country	Winery Nation
Description	Description of the wine
Designation	Commercial name of the wine
Points	Wine score between 80 and 100
Price	Price of wine
Variety	Variety of wine
Winery	Name of the winery
Year	Year of foundation of the winery

Data Analysis and Visualisation

Analysis to be carried out

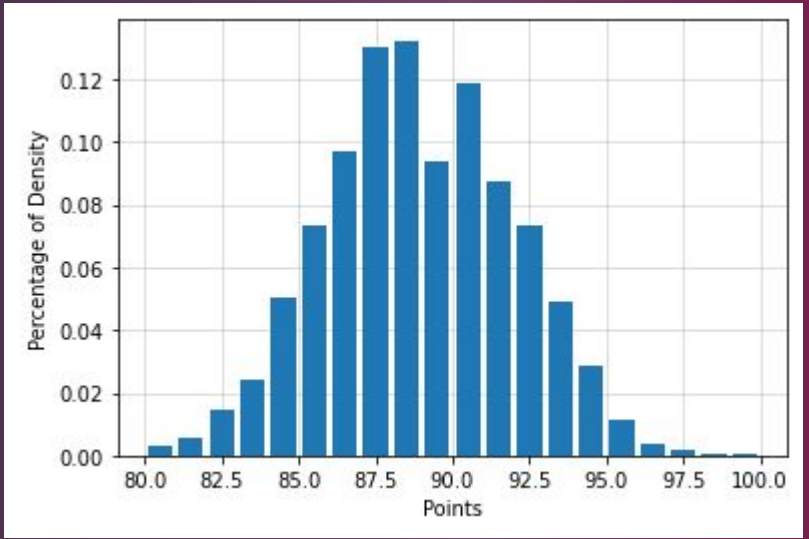
Through graphs and tables I will try to carry out the following analyses.

1. Distribution of scores
2. Price distribution
3. Ranking of wines by country
4. Ranking of wine varieties by country
5. Ranking of wines based on average score
6. Ranking of wines based on average price
7. Price-quality correlation
8. Ranking of the most expensive wines
9. Excellence ranking
10. Ranking of wines by variety
11. Ranking of wine varieties by country

1) Distribution of scores

As a first graph, you can observe how the set of scores is distributed, which falls within the range of 80 and 100.

The obtained distribution curve is close to that of a Gaussian curve with the mean coinciding with the median.



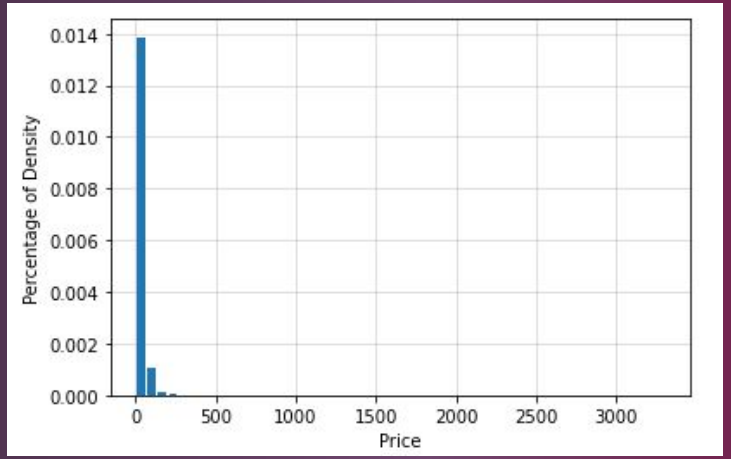

```
df['points'].describe()
```

```
count    120974.000000
mean         88.421884
std           3.044520
min          80.000000
25%          86.000000
50%          88.000000
75%          91.000000
max         100.000000
Name: points, dtype: float64
```

2) Price distribution - 1

Unlike scores, prices are spread over a much wider range. A density peak can be observed, but in this representation, it is not very clear.

The describe method suggests the presence of outliers. In fact, the highest price which is €3300, is far from the average value of €35.



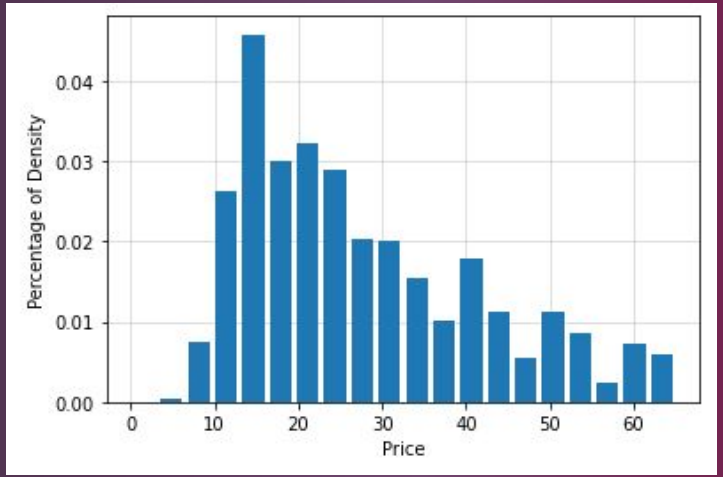

```
df['price'].describe()
```

```
count    120974.000000
mean         35.363541
std          41.022353
min           4.000000
25%          17.000000
50%          25.000000
75%          42.000000
max        3300.000000
Name: price, dtype: float64
```

2) Price distribution - 2

I calculated the position of the 90th percentile, i.e. the *price* within which 90% of the products are included, and redrawn the graph considering this upper limit.

The 90th percentile is €65 and the mean is slightly higher than the median, showing a distribution with slight positive asymmetry.

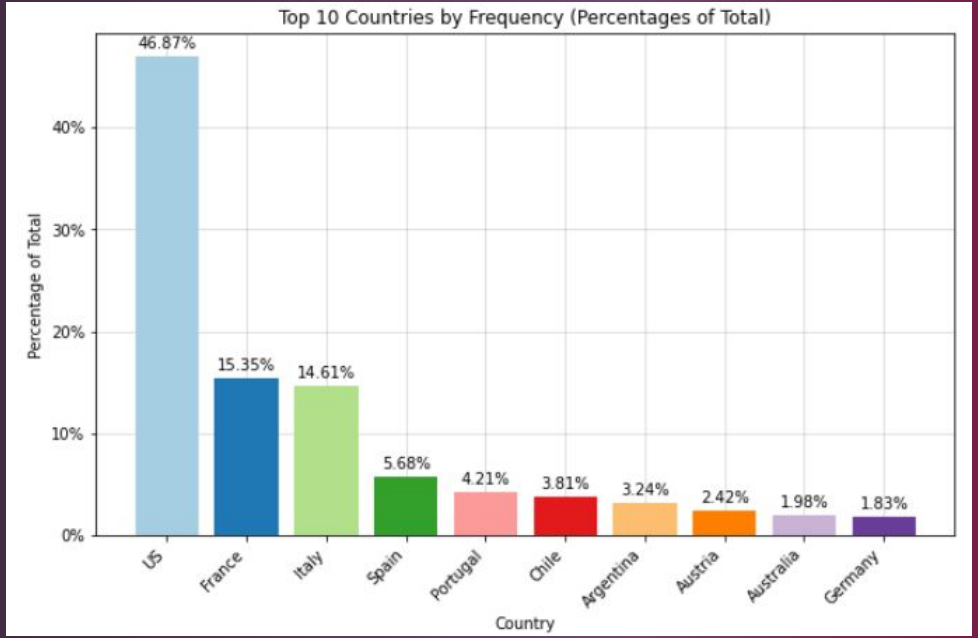



```
count    109946.000000
mean         27.621996
std          14.261669
min           4.000000
25%          16.000000
50%          24.000000
75%          36.000000
max          65.000000
Name: price, dtype: float64
```

3) Ranking of wines by country

By grouping the dataset based on the *country* field, you can observe that the most represented *country* in the dataset is only the United States.

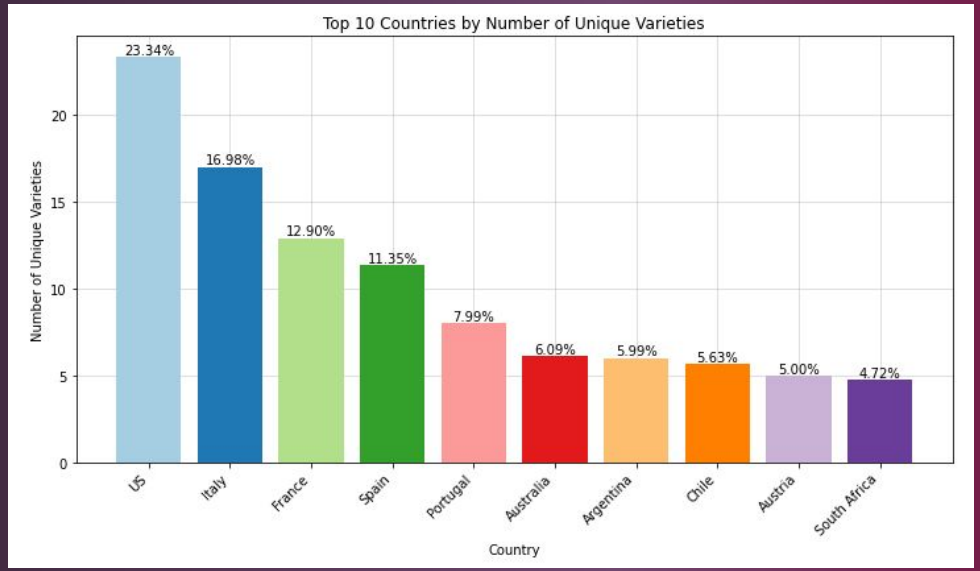
This implies that there are numerous commercial products of US origin available on the market; more than three times as many as French or Italian products.



4) Ranking of wine varieties by country

When grouping the dataset based on the country field, you can observe that the most represented variety in the dataset is still the United States, although, with a smaller gap compared to the countries that follow.

By integrating the information from this graph with the previous one, it becomes evident that the United States sells each variety of wine, on average, with more commercial names than other countries.



5) Ranking of wines based on average scores

Grouping of the wines based on the average *points* reveals that the first country in the ranking is Austria followed by Germany and France.

However, this data does not allow us to significantly discriminate between countries, because the differences between the positions are in the order of tenths.

	Country	Mean Scores
0	Austria	90.19
1	Germany	89.84
2	France	88.73
3	Italy	88.62
4	Australia	88.60
5	US	88.57
6	Portugal	88.32
7	New Zealand	88.31
8	South Africa	87.83
9	Spain	87.29
10	Argentina	86.71
11	Chile	86.50

6) Ranking of wines based on average price

In preparation for a price-score correlation analysis, I enquired how the two rankings present the same countries but in a different order.

There is a more significant gap between the first and tenth position, which turns out to have an average price less than half lower than the first in the ranking.

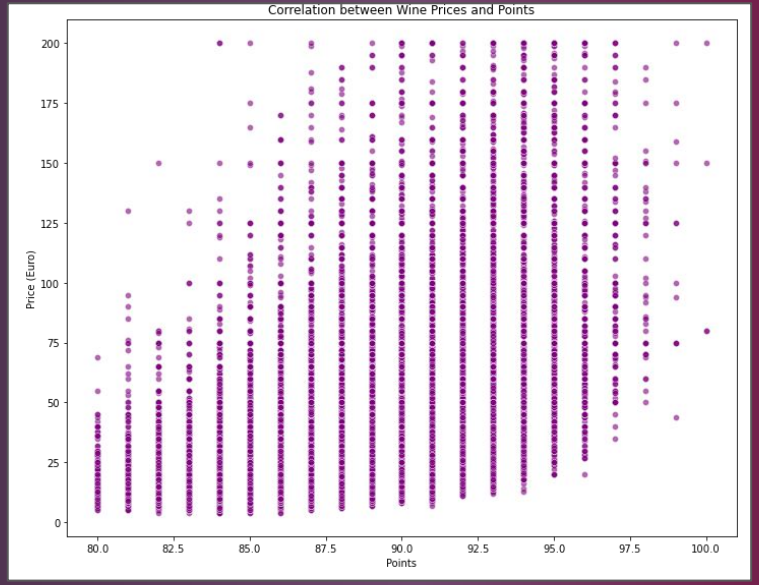
The search was limited to countries with at least 1,000 wines.

	Country	Mean Prices
0	Germany	42.26
1	France	41.14
2	Italy	39.66
3	US	36.57
4	Australia	35.44
5	Austria	30.76
6	Spain	28.22
7	New Zealand	26.93
8	Portugal	26.22
9	South Africa	24.67
10	Argentina	24.51
11	Chile	20.79

7) Price-quality correlation

Using the wine score as an indicator of quality and graphing the price trend as the score increases, you can observe a moderate positive correlation (0.42 on a scale of -1 to 1).

The graph illustrates the scarcity of expensive wines for low qualities and the scarcity of cheap wines for high qualities.



8) Ranking of the most expensive wines

Very interesting for commercial purposes is understanding which are the most expensive wines on sale.

Among the 10 most expensive wines, 9 are French including the most expensive one, €3300, a good €800 more than the second.

Furthermore, among the top 10 wines, 7 are of the Bordeaux variety.

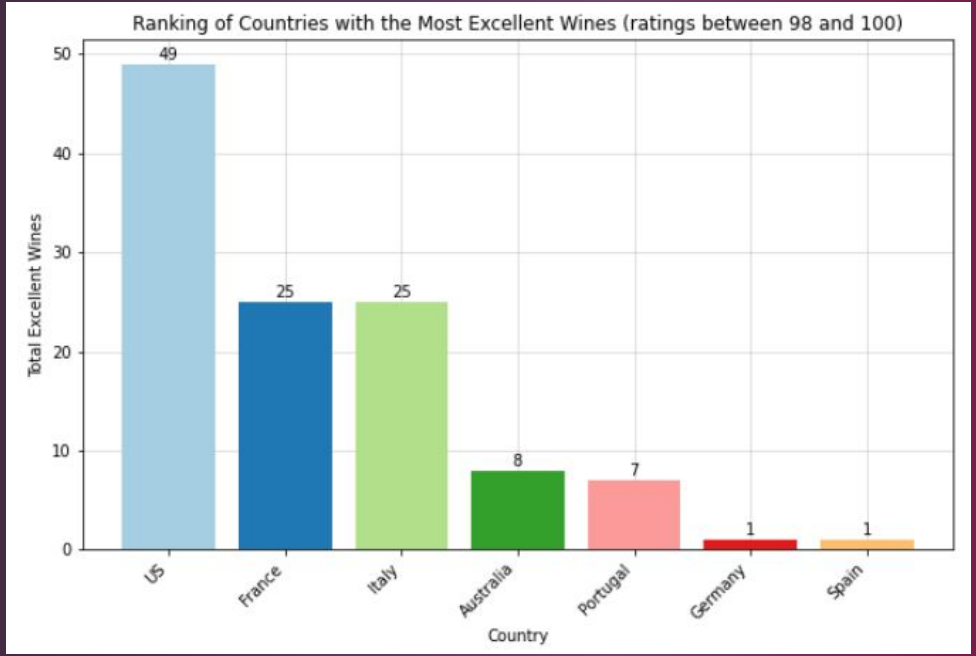
	country	variety	price
0	France	Bordeaux-style Red Blend	3300.0
1	France	Bordeaux-style Red Blend	2500.0
2	France	Pinot Noir	2500.0
3	US	Chardonnay	2013.0
4	France	Pinot Noir	2000.0
5	France	Bordeaux-style Red Blend	2000.0
6	France	Bordeaux-style Red Blend	1900.0
7	France	Bordeaux-style Red Blend	1500.0
8	France	Bordeaux-style Red Blend	1500.0
9	France	Bordeaux-style Red Blend	1300.0

9) Excellence ranking

Much more interesting information can be obtained by searching for excellence.

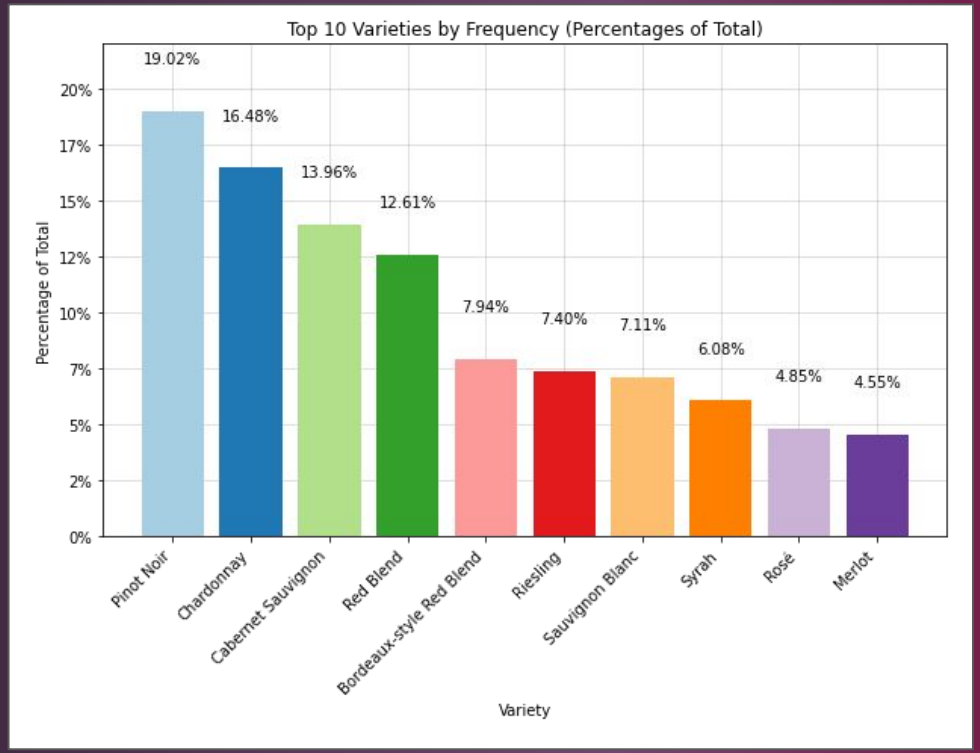
By filtering the data based only on wines with scores between 98 and 100, once again, you can observe the presence of the **United States, France,** and **Italy**.

This ranking provides a better answer to the question of which countries produce the best wines.



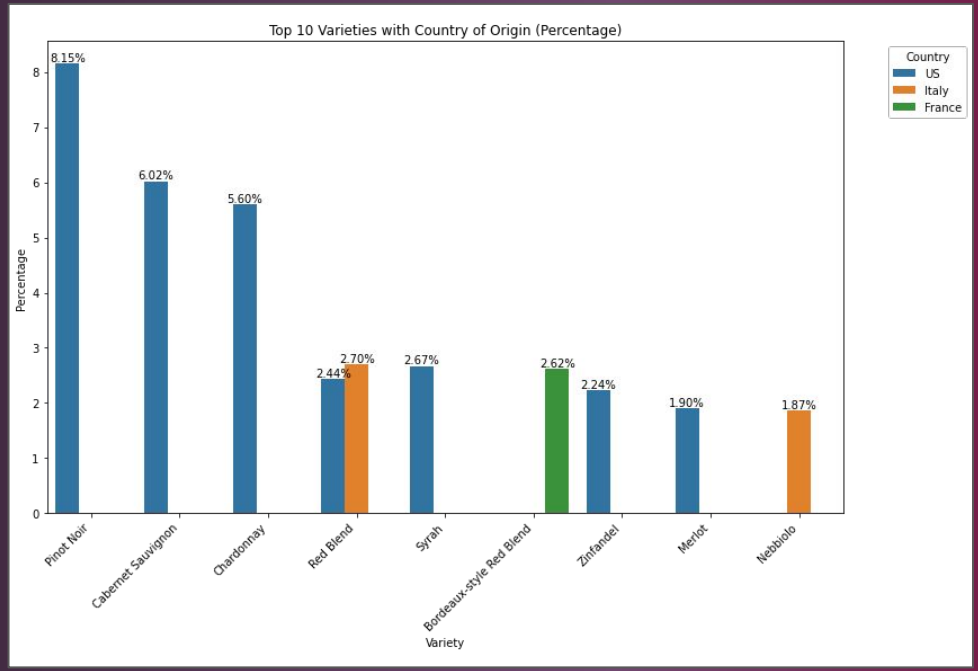
10) Ranking of wines by variety

The most represented and probably best-selling varieties are Pinot Noir, Chardonnay and Cabernet Sauvignon wines.



11) Ranking of wine varieties by country

The most represented and likely best-selling varieties are American, Italian, and French wines, with the United States once again taking the lead.



Key findings

- The United States has, against the common perception, the broadest wine variety and the highest number of excellences.
- Excluding the most expensive wines, the average price of a wine bottle is €27.
- There is only a moderate correlation between wine quality and price.
- The most expensive wines are French.
- The most common wine is Pinot Noir.

Creation of the Catalogue

Catalogue creation

Based on the analysis, I created the catalogue by selecting the top 50 wines with the highest quality priced within €100.

The chosen strategy aims to combine high-quality wines at prices in line with the purchasing power of a large portion of potential customers.

The catalogue can be viewed on my GitHub repository.