DANIELA SURCHICEAN

PROGETTO PYTHON CON PANDAS

INTRODUZIONE

Il progetto si concentra sull'analisi di un vasto dataFrame composto da oltre 130.000 recensioni di vini.
Ogni recensione contiene informazioni cruciali, tra cui la varietà, la provenienza, la vigna di produzione, il prezzo e una descrizione dettagliata.
L'obiettivo principale è quello di utilizzare queste informazioni per creare un marketplace virtuale di vini, mettendo in contatto piccoli produttori locali con acquirenti provenienti da tutto il mondo.

Attraverso l'analisi dei dati disponibili, il nostro obiettivo è identificare le varietà di vino e le vigne che godono di maggiore apprezzamento da parte dei consumatori, e questo ci consentirà di formulare una strategia per la selezione dei vini da includere nel marketplace che stiamo per creare. La selezione di vini di alta qualità e popolarità garantirà un assortimento attraente per gli acquirenti e favorirà il successo del nostro marketplace nel mercato globale del vino.

Il nostro lavoro culminerà con la proposta di una strategia di assortimento, basata sull'analisi dei dati disponibili, che servirà da solida base per il lancio e lo sviluppo del marketplace di vini.

INTRODUZIONE

Il set di dati pubblici sul vino, in formato CSV ed è disponibile su Kaggle.
Per l'analisi dei dati abbiamo utilizzato Jupyter come ambiente di sviluppo e le librerie Pandas, SeaBorn e Matlplotlib.

Il lavoro è stato suddiviso in tre fasi:

- Data Cleaning (ovvero pulizia dei dati)
- Data Analysis and Visualisation (ovvero analisi e visualizzazione dei dati)
- Creazione del catalogo

DATA CLEANING

LIBRERIE

Importo le librerie:

- Pandas: la libreria open source di Python usata per la manipolazione e l'analisi dei dati, utile per lavorare, manipolare e analizzare dati strutturati di grandi quantità.
- NumPy: la libreria Python usata per il calcolo scientifico e l'analisi dei dati.
- Seaborn: la libreria Python, messa a disposizione da Anaconda e costruita su Matplotlb, offre una varietà di pattern per la visualizzazione dei dati.
- Matplotlib: la libreria Python usata per creare una vasta gamma di visualizzazioni statiche, animate e interattive, come grafici di linea, grafici a dispersione, grafici a barre e istogrammi
- Plotly: la libreria open source Python utilizzata per la visualizzazione e la comprensione dei dati in modo semplice e facile; supporta vari tipi di grafici come grafici a linee, grafici a dispersione, istogrammi, ecc.
- Re: la libreria standard - Python che fornisce operazioni di corrispondenza delle espressioni regolari

```python
import pandas as pd
import numpy as np
import re
#data visualization
import seaborn as sns
import plotly.express as px
from matplotlib import pyplot as plt
from scipy.stats import norm
```

STUDIO DEL DATASET

```
dataset = pd.read_csv('winemag-data-130k-v2.csv')
dataset.copy()
```

La pulizia dei dati richiede innanzitutto la comprensione del contenuto come oggetto di studio.
È possibile osservare la presenza di colonne ridondanti, come Unnamed: 0 e region_2S, colonne non essenziali ai fini del catalogo, come: taster_name, taster_twitter_handle e title. Molti valori null nel campo prezzo e nel campo designation fondamentali per la creazione del catalogo.

	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

Inoltre utilizzo il metodo shape per verificare quanto è grande il DataFrame in analisi.
Proprietà che ci restituisce 13 colonne e 129971 righe.

```
dataset.shape
```

```
(129971, 13)
```

STUDIO DEL DATASET

L'istruzione columns restituiamo l'elenco con i nomi delle colonne del DataFrame nominato dataset

```
dataset.info()
```

Il metodo info restituisce un elenco di etichette dei campi, che si integra con l'osservazione che fatta con il metodo head.

Si può osservare che le 129.971 righe hanno campi vuoti la cui percentuale può essere calcolata.

	Unnamed: 0	points	price
count	129971.000000	129971.000000	120975.000000
mean	64985.000000	88.447138	35.363389
std	37519.540256	3.039730	41.022218
min	0.000000	80.000000	4.000000
25%	32492.500000	86.000000	17.000000
50%	64985.000000	88.000000	25.000000
75%	97477.500000	91.000000	42.000000

```
dataset.columns
```

```
Index(['Unnamed: 0', 'country', 'description', 'designation', 'points',
       'price', 'province', 'region_1', 'region_2', 'taster_name',
       'taster_twitter_handle', 'title', 'variety', 'winery'],
      dtype='object')
```

```
 #   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
```

Il metodo describe ci permette di ottenere le statistiche descrittive per ciascuna colonna con valore numerico sia intero che decimale.
Si può osservare come la colonna points ha i dati al completo e il valore medio (tra 80 e 100) è 88,44, invece la colonna price ha circa 1000 valori mancanti e il valore medio è 35,36.

STUDIO DEL DATASET

L'istruzione columns restituiamo l'elenco con i nomi delle colonne del DataFrame nominato dataset

```
dataset.info()
```

Il metodo info restituisce un elenco di etichette dei campi, che si integra con l'osservazione che fatta con il metodo head.

Si può osservare che le 129.971 righe hanno campi vuoti la cui percentuale può essere calcolata.

```
dataset.columns
```

```
Index(['Unnamed: 0', 'country', 'description', 'designation', 'points',
       'price', 'province', 'region_1', 'region_2', 'taster_name',
       'taster_twitter_handle', 'title', 'variety', 'winery'],
      dtype='object')
```

```
 #   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
```

Il metodo describe ci permette di ottenere le statistiche descrittive per ciascuna colonna con valore numerico sia intero che decimale.
Si può osservare come la colonna points ha i dati al completo e il valore medio (tra 80 e 100) è 88,44, invece la colonna price ha circa 1000 valori mancanti e il valore medio è 35,36.

STUDIO DEL DATASET

Il metodo dataset.isnull rileva i dati mancanti a cui si applica il metodo sum così per restituire la somma dei valori mancati di ogni colonna del DataFrame.
Avendo questa somma si calcola la percentuale e di restituisce in output il numero dei valori null e la percentuale di questi.

```
#analizzo e sommo i valori nulli in tabella
total_counts_null = dataset.isnull().sum()
#calcolo il numero totale dei valori in ogni colonna
total_counts = dataset.shape[0]
#calcolo la percetuali dei valori nulli in ogni colonna
total_counts_null_perc=((total_counts_null/total_counts)*100).round(2).astype(str)+'%'
#creo un nuovo dataframe che contiene i risultati dei valori nulli e la percetuale
dataset_null_info = pd.DataFrame({'Number Null' : total_counts_null,'% Null' : total_counts_null_perc})
dataset_null_info
```

	Number Null	% Null
Unnamed: 0	0	0.0%
country	63	0.05%
description	0	0.0%
designation	37465	28.83%
points	0	0.0%
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	24.02%

Si può osservare come la colonna designation, fondamentale per l'analisi, la cui percentuale di valori nulli è pari al 28.83% e la colonna price in cui mancano il 6,02% dei valori complessivi, ovvero per un totale di 8996 valori nulli.

PULIZIA DEL DATASET

Con le 2 funzioni personalizzate andiamo ad uniformare il dataset, riempiere i dati mancanti e creare colonne necessarie all'analisi:

1. La funzione extract_year_from_str attraverso una regex estrae l'anno dalla colonna titolo e infine crea la colonna anno e assegna ad ogni specifica riga il valore corrispodente.
2. La funzione extract_designation_from_str attraverso una regex estrae la designation di ogni riga del dataset dal title e l'assegna alla colonna designation, solo se questa ha un valore nullo.

Dopo aver recuperato i dati necessari per le analisi successive e aver riempito il più possibile i campi nulli visualizziamo il dataset modificato, andato ad eliminare le colonne superflue all'analisi, come province, Unnamed:0, region_1, region_2, taster_name e taster_teitter_handle.

```python
#definisco la funzione che estrare l'anno dal titolo utilizzando l'espressione regolare
def extract_year_from_str (text):
    #regex che descrive 4 numeri con uno spazio bianco prima e uno spazio bianco dopo
    regex_year = r"\s\d\d\d\d\s"
    #utilizzo la funzione findall per restituisce un elenco di tutte le corrispondenze della regex in text
    year = re.findall(regex_year, text)
    if year:
        #restituisco il primo elemento con indice 0 di year
        return int(year[0])
    else:
        return None

#richiamo la funzione con il campo title per estrarre l'anno
dataset['year'] = dataset['title'].apply(extract_year_from_str)
#sostituisco nella colonna year i valori NaN con 0
dataset['year'].fillna(0, inplace=True)
#applico alla colonna year tipo di dato int
dataset['year'] = dataset['year'].astype(int)

#definisco la funzione che estrare l'anno dal titolo utilizzando l'espressione regolare
def extract_designation_from_str (row):
    if pd.isna(row['designation']):
        result=re.search(r"\s\d\d\d\d\s(.+?)\s\(",row['title'])
        if result:
            row['designation']=result.group(1)
        else:
            row['designation']=row['variety']
        #restituisco il primo elemento con indice 0 di year
    return row
dataset=dataset.apply(extract_designation_from_str, axis=1)

#rimuovo le colonne non necessarie dal DataFrame
dataset.drop(["Unnamed: 0", "province", "region_1", "region_2", "taster_name", "taster_twitter_handle"],
             axis='columns', inplace=True)
#visualizzo il DataFrame aggiornato

dataset
```

	country	description	designation	points	price	title	variety	winery	year
0	Italy	Aromas include tropical fruit, broom, brimston...	Vulkà Bianco	87	NaN	Nicosia 2013 Vulkà Bianco (Etna)	White Blend	Nicosia	2013
1	Portugal	This is ripe and fruity, a wine that is smooth...	Avidagos	87	15.0	Quinta dos Avidagos 2011 Avidagos Red (Douro)	Portuguese Red	Quinta dos Avidagos	2011
2	US	Tart and snappy, the flavors of lime flesh and...	Pinot Gris	87	14.0	Rainstorm 2013 Pinot Gris (Willamette Valley)	Pinot Gris	Rainstorm	2013
3	US	Pineapple rind, lemon pith and orange blossom ...	Reserve Late Harvest	87	13.0	St. Julian 2013 Reserve Late Harvest Riesling ...	Riesling	St. Julian	2013
4	US	Much like the regular bottling from 2012, this...	Vintner's Reserve Wild Child Block	87	65.0	Sweet Cheeks 2012 Vintner's Reserve Wild Child...	Pinot Noir	Sweet Cheeks	2012
...	...	...	...	...	...	...	...	...	...
129956	Germany	Notes of honeysuckle and cantaloupe sweeten th...	Brauneberger Juffer-Sonnenuhr Spätlese	90	28.0	Dr. H. Thanisch (Erben Müller-Burggraef) 2013 ...	Riesling	Dr. H. Thanisch (Erben Müller-Burggraef)	2013
129957	US	Citation is given as much as a decade of bottl...	Pinot Noir	90	75.0	Citation 2004 Pinot Noir (Oregon)	Pinot Noir	Citation	2004
129958	France	Well-drained gravel soil gives this wine its c...	Kritt	90	30.0	Domaine Gresser 2013 Kritt Gewurztraminer (Als...	Gewürztraminer	Domaine Gresser	2013
129959	France	A dry style of Pinot Gris, this is crisp with ...	Pinot Gris	90	32.0	Domaine Marcel Deiss 2012 Pinot Gris (Alsace)	Pinot Gris	Domaine Marcel Deiss	2012

STUDIO DEL DATASET

Dopo aver pulito il dataset visualizziamo le colonne e controlliamo la percentuale e il numero dei valori nulli.

```
#DATASET DOPO LA PULIZIA OVVERO IL CLEANING DEI DATI
#analizzo e sommo i valori nulli in tabella
total_counts_null = dataset.isnull().sum()
#calcolo il numero totale dei valori in ogni colonna
total_counts = dataset.shape[0]
#calcolo la percetuali dei valori nulli in ogni colonna
total_counts_null_perc=((total_counts_null/total_counts)*100).round(2).astype(str)+'%'
#creo un nuovo dataframe che contiene i risultati dei valori nulli e la percetuale
dataset_null_info = pd.DataFrame({'Number Null' : total_counts_null,'% Null' : total_counts_null_perc})
dataset_null_info
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 120974 entries, 1 to 129970
Data columns (total 9 columns):
 #   Column       Non-Null Count   Dtype
---  ------       --------------   -----
 0   country      120915 non-null  object
 1   description  120974 non-null  object
 2   designation  120974 non-null  object
 3   points       120974 non-null  int64
 4   price        120974 non-null  float64
 5   title        120974 non-null  object
 6   variety      120974 non-null  object
 7   winery       120974 non-null  object
 8   year         120974 non-null  int64
dtypes: float64(1), int64(2), object(6)
memory usage: 9.2+ MB
```

	Number Null	% Null
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%

DESCRIZIONE DEL DATASET

CAMPO	DESCRIZIONE
Country	Nazione del vino
Description	Descrizione del vino
Designation	Nome del vino
Points	Punti assegnati al vino tra 80 e 100
Variety	Varietà del vino
Winery	Nome dell’azienda vinicola
Year	Anno di fondazione dell’azienda vinicola

DATA ANALYSIS & VISUALIZATION

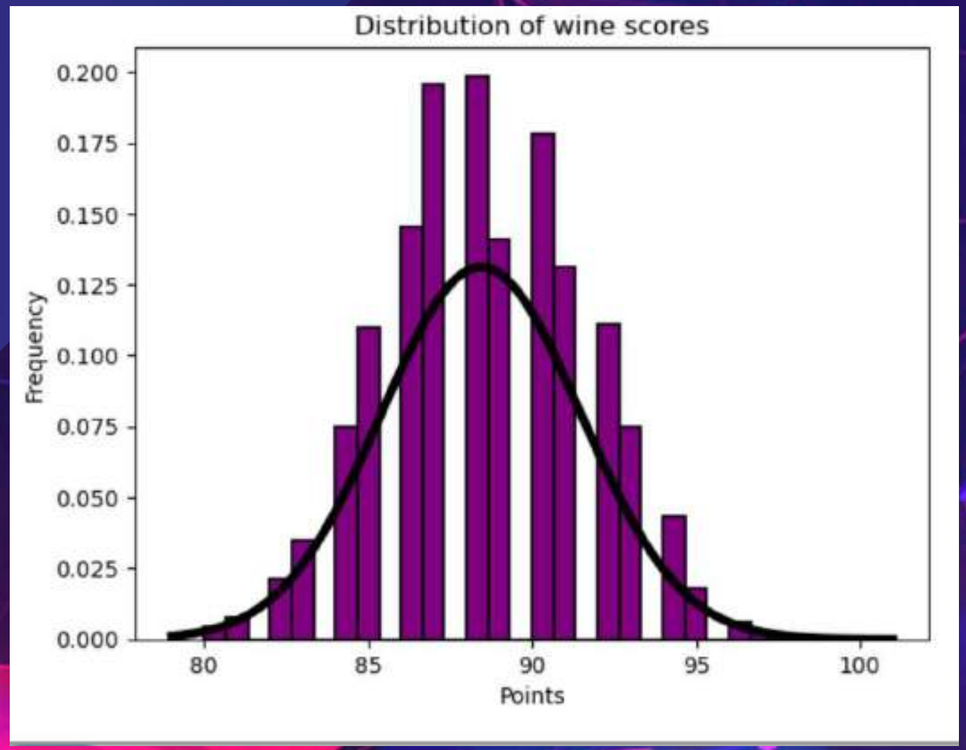
STUDIO DEL DATASET

- ANALISI PUNTEGGI ASSEGNATI AI VINI IN BASE ALLE RECENSIONI
- ANALISI DEGLI OUTLIERS DEI PUNTEGGI ASSEGNATI AI VINI
- ANALISI DELLA DISPERSIONE TRA IL PREZZO DEI VINI E PUNTEGGI ATTRIBUITI
- ANALISI SULLA DISTRIBUZIONE DEI PREZZI DEI VINI
- ANALISI SUL CONTEGGIO DEI PREZZI DEI VINI
- ANALISI SUL CONTEGGIO DELLE VARIETÀ DEI VINI
- ANALISI PUNTEGGI ASSEGNATI AI VINI IN BASE ALLE RECENSIONI
- ANALISI DEGLI OUTLIERS DEI PUNTEGGI ASSEGNATI AI VINI

ANALISI PUNTEGGI ASSEGNATI AI VINI IN BASE ALLE RECENSIONI

Nella prima analisi in cui prendiamo in considerazione i punteggi (ovvero gli score) assegnati ai vini dalle recensioni, si osserva come sono distribuiti i punteggi e che l'intervallo di essi oscilla tra 80.00 e 100.00.
La curva di distribuzione ottenuta si avvicina a quella di una curva gaussiana, con la media che coincide con la mediana. Infine delineiamo la curva della distribuzione normale adattata ai dati.

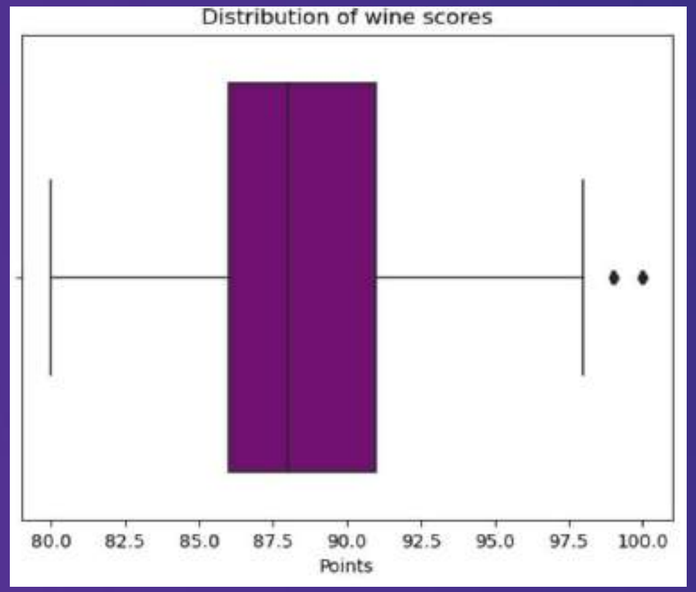
```
#DATA ANALYSIS AND VISUALISATION
#Istogramma Gaussiano che mostra la distribuzione dei punteggi dei vini
data=dataset['points']
mu, std = norm.fit(data)

#Traccio l'istogramma
plt.hist(data, bins=30, density=True, color='purple', edgecolor='black', linewidth=1.2)

xmin, xmax = plt.xlim()
x = np.linspace(xmin, xmax, 100)
p = norm.pdf(x, mu, std)

#Traccio la curva
plt.xlabel('Points')
plt.ylabel('Frequency')
plt.title('Distribution of wine scores')
plt.plot(x, p, 'k', linewidth=4)
plt.show()
```



count	129971.000000
mean	88.447138
std	3.039730
min	80.000000
25%	86.000000
50%	88.000000
75%	91.000000
max	100.000000

ANALISI DEGLI OUTLIERS DEI PUNTEGGI ASSEGNATI AI VINI

```
#Creo. un box plot dei punteggi che rappresenta una panoramica dei punteggi e permette di visualizzare
#eventuali outliers ovvero valori fuori dalla norma
sns.boxplot(x=dataset['points'], linewidth=1.2, color='purple')
plt.xlabel('Points')
plt.title('Distribution of wine scores')
plt.xticks(rotation=90)
plt.show()
```

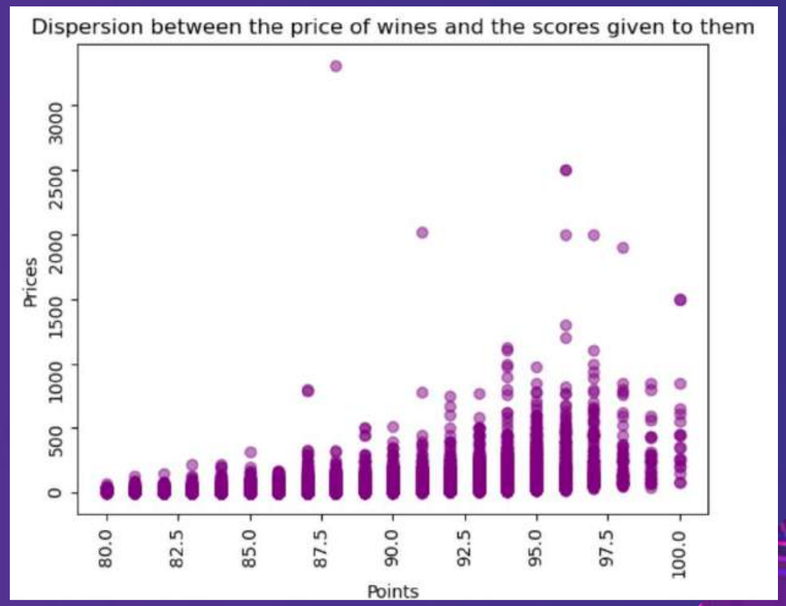


Nell'analisi degli outliers dei punteggi dei vini utilizziamo un box plot per comprendere la centralità della distribuzione, la dispersione, la presenza di valori anomali e la simmetria dei dati.

Gli Outliers, ovvero i valori anomali, cadono al di fuori dei baffi e vengo punti o asterischi, nel caso specifico questi si trovano sull'area tra 90.00 e 100.00 score e si discostano significativamente dalla maggior parte dei dati che si concentrano tra gli 85.00 e i 91.00..

ANALISI DELLA DISPERSIONE TRA IL PREZZO DEI VINI E PUNTEGGI ATTRIBUITI

```
#ANALISI PRICES VINI
#Analizzo e visualizzo la dispersione tra il prezzo dei vini e i punteggi a loro attribuiti
plt.scatter(dataset['points'],dataset['price'], color='purple', alpha=0.5)
plt.xlabel('Points')
plt.ylabel('Prices')
plt.xticks(rotation=90)
plt.title('Dispersion between the price of wines and the scores given to them')
plt.show()
```



Nell'analisi della dispersione tra il prezzo dei vini e punteggi attribuiti utilizziamo un Scatter plot, ovvero grafico a dispersione, rappresenta la correlazione tra punteggi e prezzi dei vini nel dataFrame; sull'asse della ascisse vengono mostrati i punteggi e sull'asse delle ordinate il prezzo dei vini così da mostrare la relazione tra entrambi i valori.
Ogni punto equivale a un vino, quindi una riga del dataFrame quando i punti si concentrano sulla linea verticale significa che esiste una correlazione tra i due valori degli assi.

.

ANALISI SULLA DISTRIBUZIONE DEI PREZZI DEI VINI

```
#Informazioni sulla colonna prices
dataset['price'].describe()
print(f"Curtosi: {dataset['price'].kurt()}")
print(f"Asimmetria: {dataset['price'].skew()}")
```

```
Curtosi: 829.5201815158334
Asimmetria: 18.000957415874364
```

```
price
20.0      6940
15.0      6066
25.0      5805
30.0      4951
18.0      4883
          ...
198.0        1
1125.0       1
470.0        1
268.0        1
848.0        1
Name: count, Length: 390, dtype: int64
```

Nell'analisi sulla distribuzione dei prezzi dei vini utilizziamo il quadro completo delle statistiche descrittive dei prezzi del dataset, inclusi il conteggio, la media, la deviazione standard, i quartili, nonché misure di curtosi e asimmetria per valutare la forma e la distribuzione dei prezzi.

1. Curtosi: è il metodo che calcola e stampa la misura della *"appuntitura"* della distribuzione. Il valore positivo 829.52 dimostra che la distribuzione è molto "*appuntita*".
2. Asimmetria: calcola e stampa l'asimmetria della distribuzione dei prezzi. Il valore 18.00 indica una coda lunga verso destra della distribuzione e una distrubizione asimmetrica.

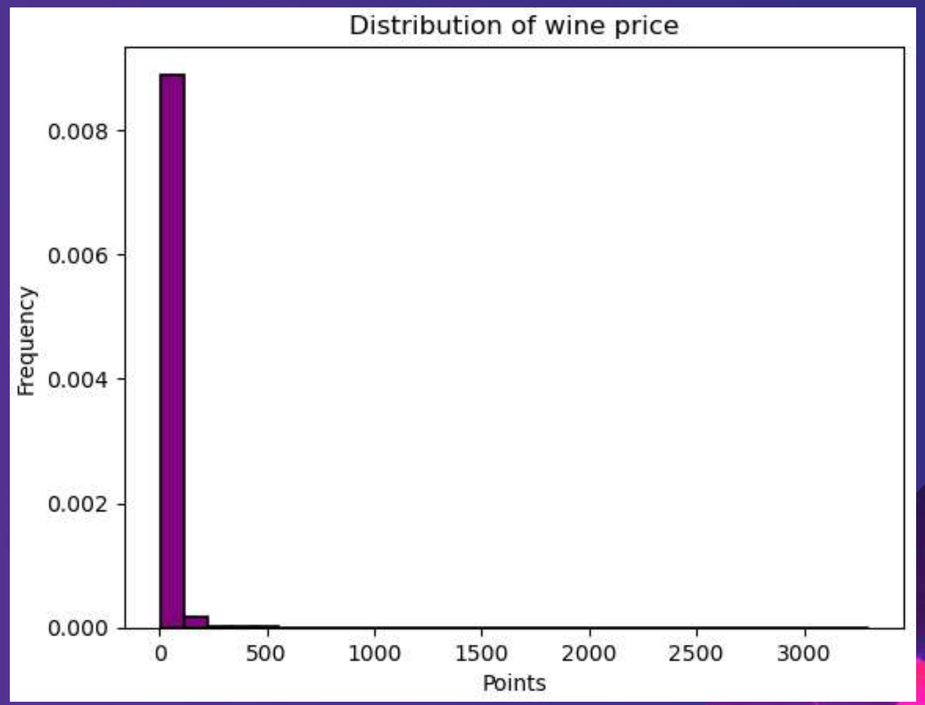
ANALISI SUL CONTEGGIO DEI PREZZI DEI VINI

Nell'analisi sul conteggio dei prezzi dei vini restituiamo una rappresentazione grafica del conteggio dei prezzi del dataFrame utilizzando un grafico a barre. Inoltre la stampa del conteggio dei valori unici della colonna dei prezzi. Questa analisi mostra come il prezzo più comune, ovvero quello assegnato ad un numero di vini maggiore è 20.00€, seguito con 15.00€, 25.00€ e una grande coda in cui i prezzi sono assegnati solo ad un singolo vino come 385.00€, 222.00€, etc. I prezzi unici sono unn totale di 390 record ma molti di essi, come anticipato e mostrato nell'istogramma, hanno assegnato solo un vino.

ANALISI SULLA FREQUENZA DEL PREZZO DEI VINI

Nell'analisi sulla frequenza del prezzo assegnato al singolo vino utlizziamo un istogramma, dimostrando che differenza dei punteggi, i prezzi sono distribuiti

```
#Istogramma che mostra la distribuzione dei prezzi dei vini
plt.hist(dataset['price'], bins=30, density=True, color='purple', edgecolor='black', linewidth=1.2)
plt.xlabel('Points')
plt.ylabel('Frequency')
plt.title('Distribution of wine price')
plt.show()
```



su un intervallo molto più ampio.

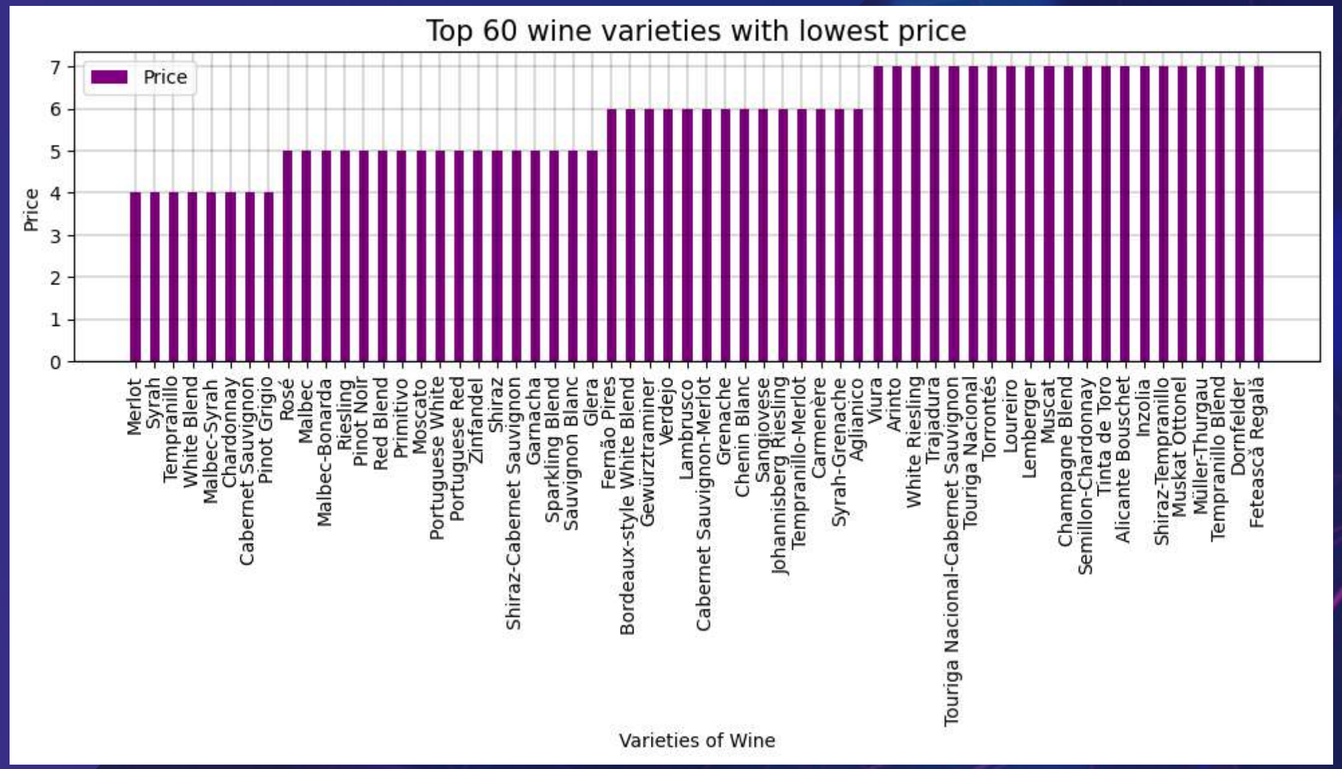
Si può osservare un picco di densità, ma in questa rappresentazione non è molto chiara, come nell metodo descrittivo e nell'analisi degli outlier.
Infatti, il prezzo più alto, pari a 3300.00€, molto lontano dal dal valore medio di 35.00€.

ANALISI SULLE QUALITÀ DI VINO CON PREZZI MENO COSTOSI

Nell'analisi sui prezzi ci concentriamo sulle prime 60 varietà di vino con il prezzo inferiore.

Dimostrando che tutte le 60 tipologie di vino più economiche hanno un prezzo di vendità che parte da 4€ e arriva a 7€.

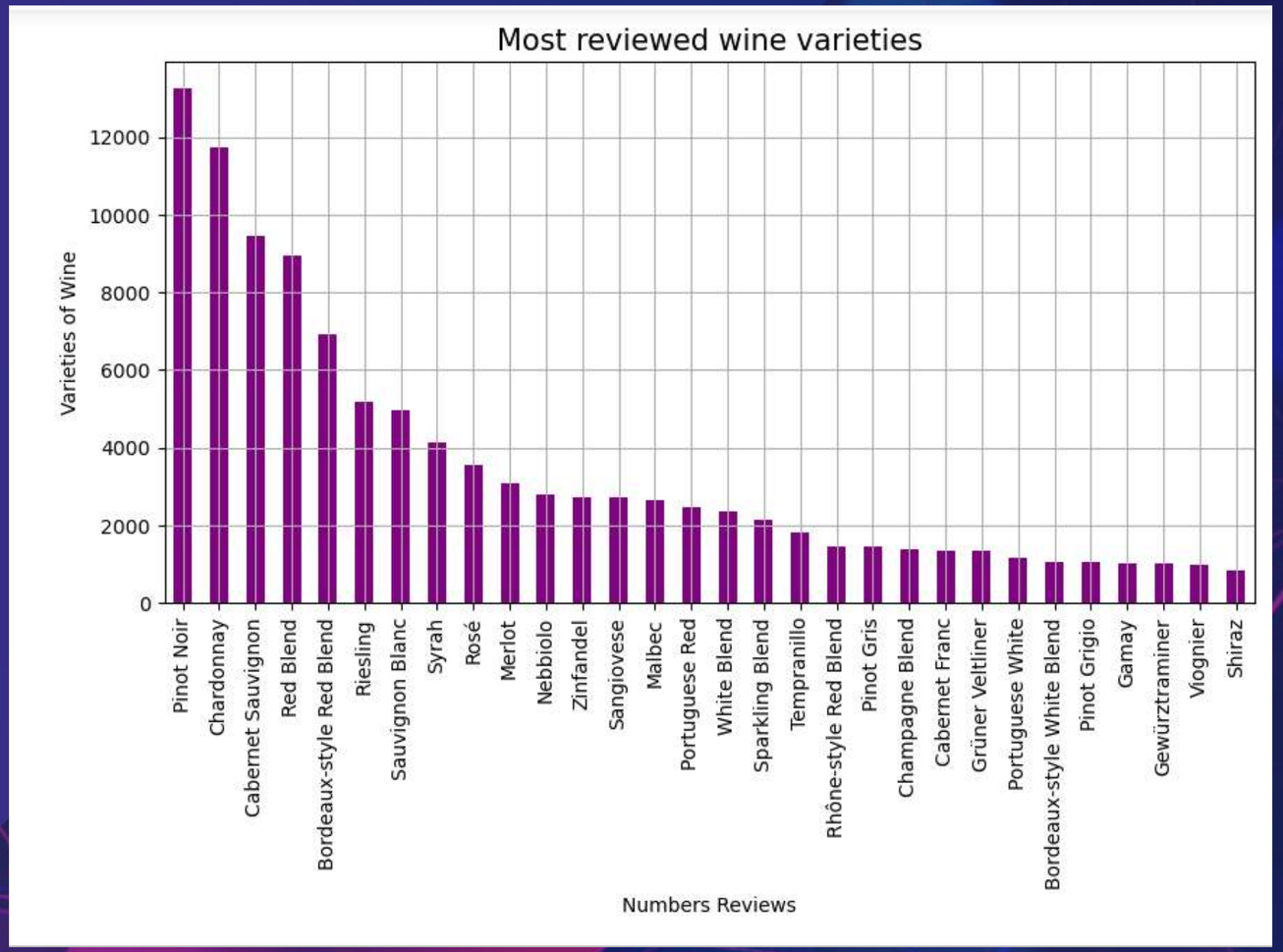
Dunque esistono varietà di vino recensite il cui prezzo è inferiore a 10.00€ e perfino inferiore a 5.00€, questo è il caso di Merlot, Syrag, Tempranillo, White Blend e MalBec-Binarda (il cui prezzo è 4€).



ANALISI SUL CONTEGGIO DELLE VARIETÀ DEI VINI PIÙ RECENSITI

Nell'analisi delle varietà di vini presenti nel dataset e crea un grafico a barre per visualizzare le varietà più recensite. Quindi mettendo in correlazione le qualità di vino e le recensioni attribuite ad esse ed infine ordinandole per numero di recensioni attributi, si mostra nel grafico a barre che la qualità di vino più recensita è Pinot Noir con più di 12000 recensioni, seguita da Chardonnay con poco meno di 12000 recensioni.

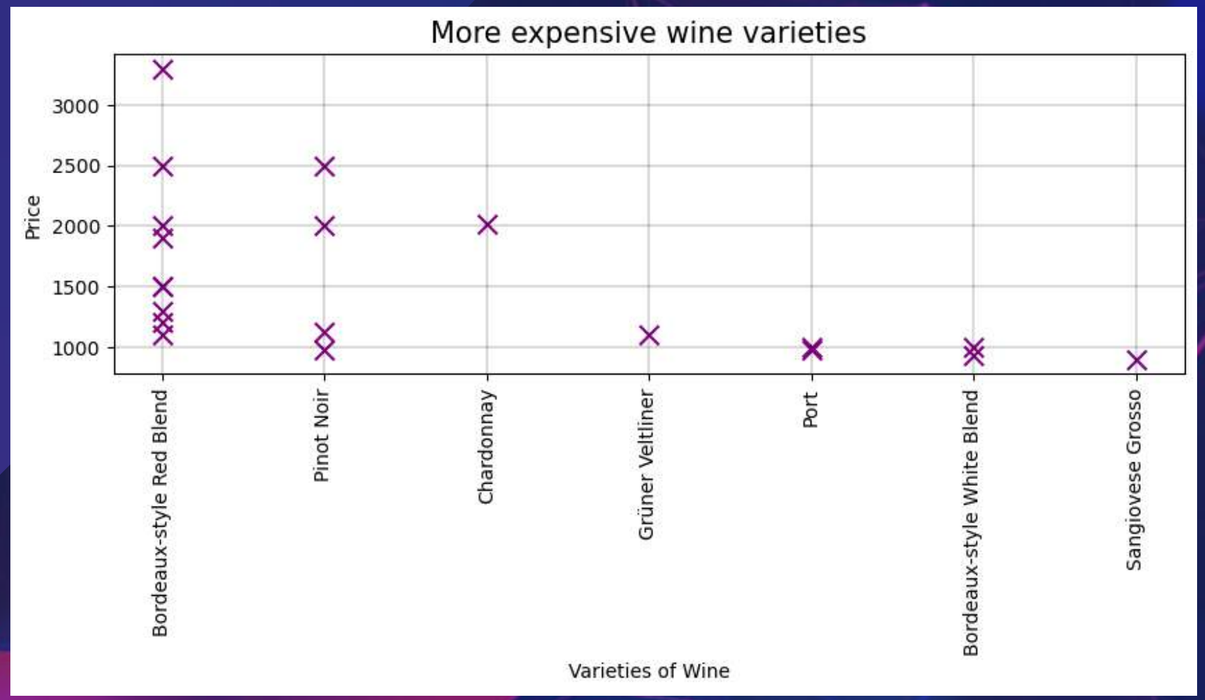
```
#ANALISI VARIETA' VINI
#Conto le varietà di vini e le ordino in modo decrescente
counts_variety=dataset['variety'].value_counts().sort_values(ascending=False)

#Mostro le prime 30 varietà più recensite
varieta_top=counts_variety.head(30)

#Creo l'istogramma che mostra le varietà più recensite
varieta_top.plot(kind="bar", color='purple', figsize = (10, 5))

plt.xlabel('Numbers Reviews',fontsize = 10)
plt.ylabel('Varieties of Wine', fontsize = 10)
plt.title('Most reviewed wine varieties',fontsize = 15)
plt.grid(True)
plt.xticks(rotation=90)
plt.show()
```



ANALISI SULLE QUALITÀ DI VINO PIÙ COSTOSE

Nell'analisi delle varietà di vini presenti nel dataset analizziamo più approfonditamente le qualità di vino più costose.
Dimostrando che il vino più costoso presente nel dataFrame è il Bordeaux-style Red Blend con un prezzo superiore a 3000€, seguito da Pinot Noir, che oltre ad essere il vino più recensito è una delle qualità più costose con il prezzo massimo di 2500€ seguito anche in questa analisi dalla varietà Chardonnay con il prezzo massimo di 2000€; solo dalla quarta varietà più costosa abbiamo un abbassamento notevole dei prezzi massimi che non superano i 1000€.

```
#Le 20 varietà di vino più costose
variety_countries= dataset.nlargest(20,'price')
plt.figure(figsize=(10,3))
#Scatter plot delle 20 varietà di vino più costose
plt.scatter(variety_countries['variety'], variety_countries['price'], marker='x', s=100, color='purple')
plt.xlabel('Varieties of Wine',fontsize = 10)
plt.ylabel('Price', fontsize = 10)
plt.title('More expensive wine varieties',fontsize = 15)
plt.grid(color = 'black', linewidth = 0.2)
plt.xticks(rotation=90)
plt.show()
```



ANALISI SUL PREZZO MEDIO DELLE QUALITÀ DI VINO PIÙ COSTOSE

Nell'analisi sul prezzo medio delle qualità più costose prende in considerazione le prime 20 qualità di vino con il prezzo più elevato e di queste non calcola più il prezzo massimo ma calcoli il prezzo medio più elevato.

Si dimostra che la varietà di vino con il prezzo medio più elevato è Pamisco con il prezzo medio di 500€, seguita da Ternantez con 220€ e Francisa con 170€.

Il prezzo medio si abbassa sotto i 100€ dalla decima posizione in poi con la varietà Cabemet-Shiraz seguita da Mazuelo e Champagne Blend (con prezzo medio 50€).

```
#Media del prezzo per ogni varietà
aver_variety_price=dataset.groupby('variety')['price'].mean()
#prime 20 varietà con prezzo medio più alto
price_high_variety=aver_variety_price.nlargest(20)
#Scatter Plot sul prezzo medio delle 20 qualità più costose
plt.figure(figsize=(10,3))
plt.scatter(price_high_variety.index, price_high_variety.values, marker='x', s=100, color='purple')
plt.xlabel('Varieties of Wine',fontsize = 10)
plt.ylabel('Middle Price', fontsize = 10)
plt.title('Middle price of more expensive wine varieties',fontsize = 15)
plt.grid(color = 'black', linewidth = 0.2)
plt.xticks(rotation=90)
plt.show()
```

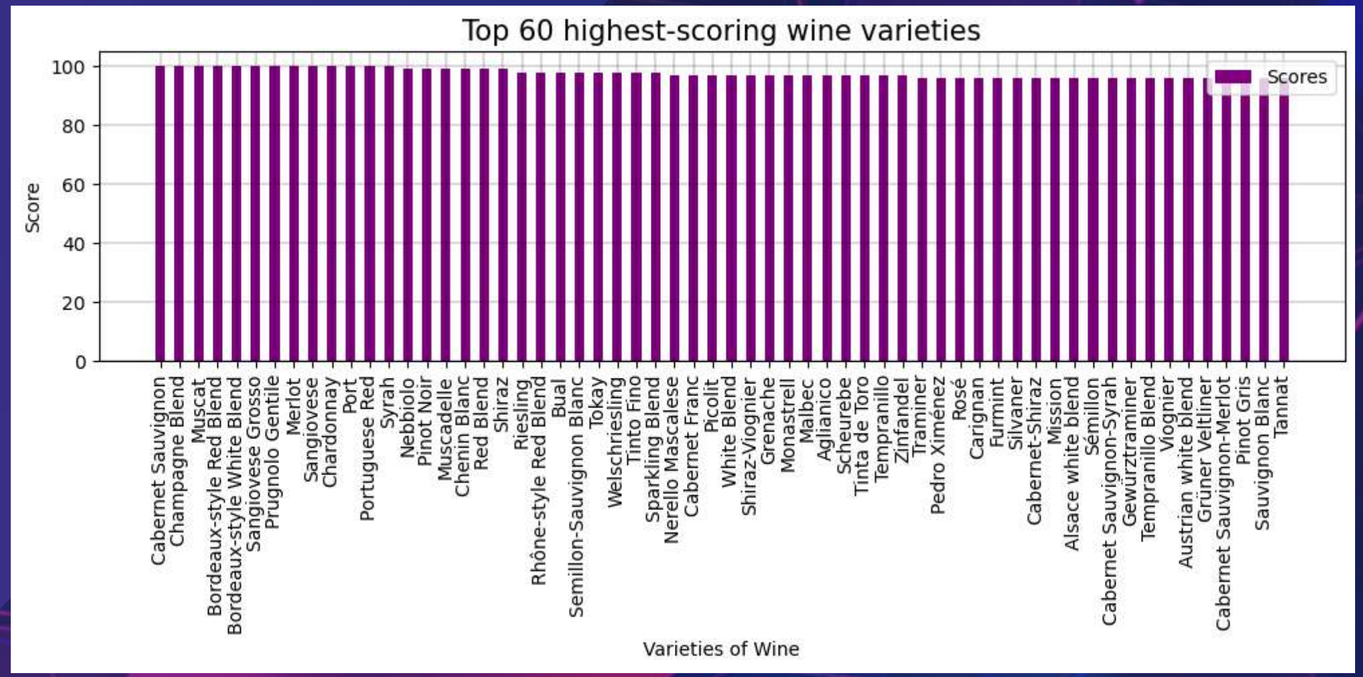
ANALISI DELLE VARIETÀ DI VINI CON SCORE ASSEGNATO PIÙ ALTO

Nell'analisi sugli score assegnati alle qualità di vino restituisce le prime sessanta tipologie di vino che hanno assegnato il punteggio più elevato.

Si dimostra che le varietà di vino più apprezzate e quindi con un punteggio più alto in tutto sono 13 e hanno un punteggio pari a 100 tra di esse ci sono: Cabernet, Sauvignon, Merlot, Sangiovese, Sangiovese Grossom e Muscat.

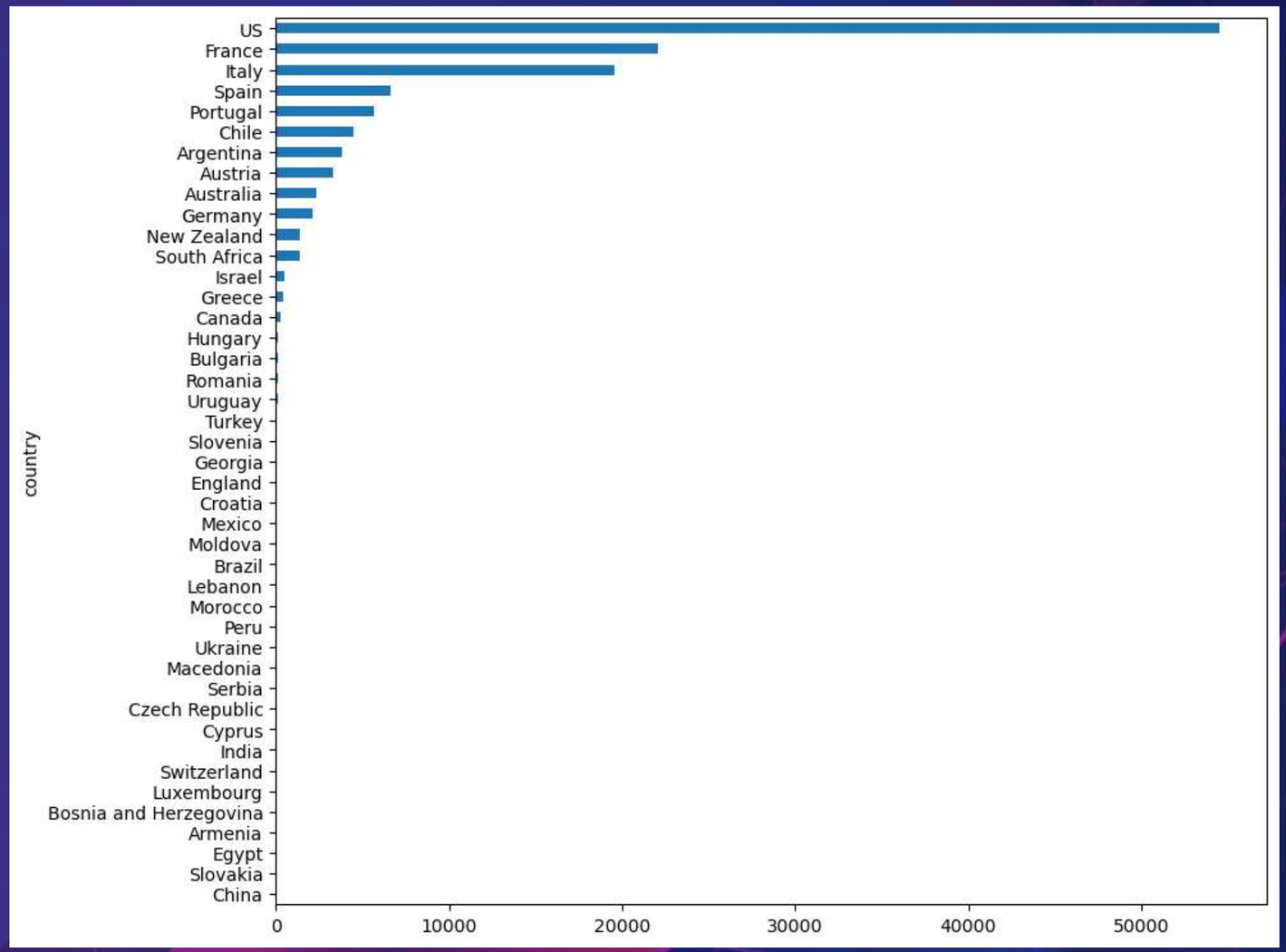
Il punteggio si abbassa sotto i 100 dalla quattordicesima posizione in poi con la varietà Nebbiolo seguita da Pinot Noir, Moscandelle e Chenim Blanc.

```
#ANALISI SULLE PRIME 60 QUALITA DI VINO CON UN PUNTEGGIO MAGGIORE
higher_points=dataset.groupby(['variety'])['points'].max().sort_values(ascending=False).to_frame()[:60]
plt.figure(figsize=(12,3))
plt.bar(higher_points.index, higher_points['points'], width = 0.5, color = 'purple', label='Scores')
plt.legend()
plt.xticks(higher_points.index, rotation=90)
plt.xlabel('Varieties of Wine',fontsize = 10)
plt.ylabel('Score', fontsize = 10)
plt.title('Top 60 highest-scoring wine varieties',fontsize = 15)
plt.grid(color = 'black', linewidth = 0.2)
plt.show()
```



ANALISI DEI VINI PER PAESE DI ORIGINE

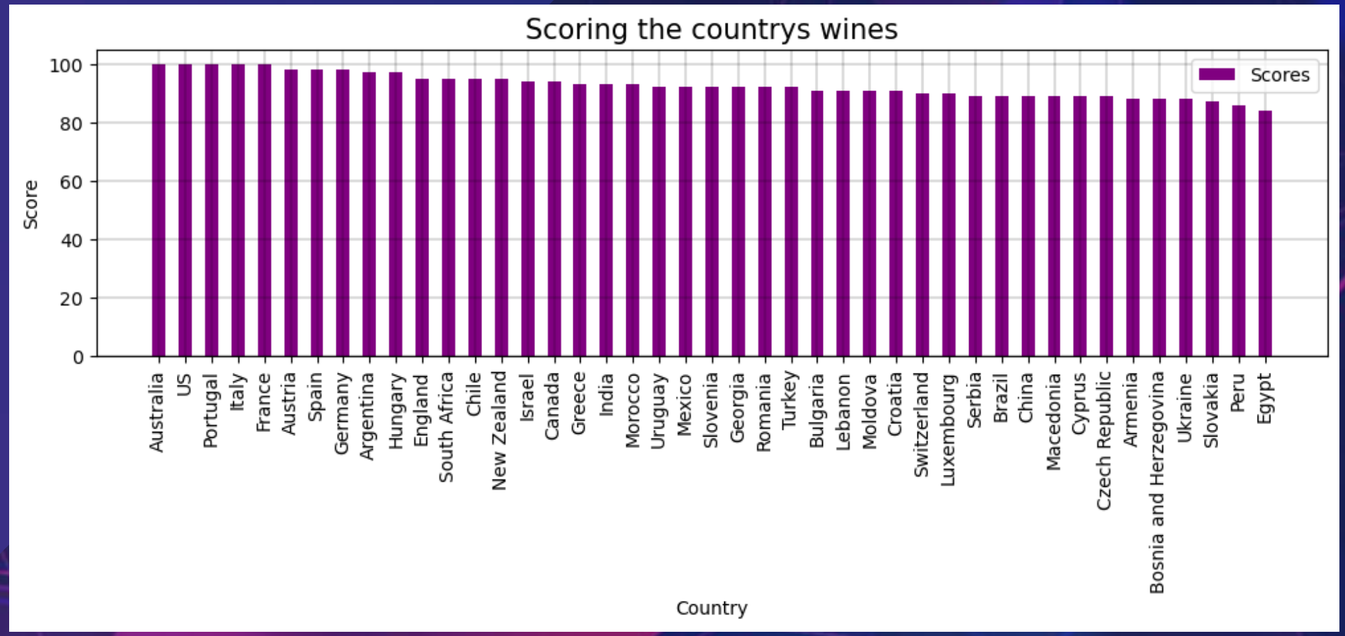
Nell'analisi dei vini per paese di origine, raggruppa i vini per paese e dimostra che il paese con più vini recensiti è US (con 54504 numero di vini), seguito da: Francia - numero vini 22093 e Italia - numero vini 19540. Questi in linea generale corrispondono ai paesi che producono maggiormente vini.
Invece infondo nella coda troviamo: China, Slovakia e Egypt con un solo vino, Armenia e Bosnia and Herzegovina con 2 vini, Luxembourg con 6 vini, Switzerland con 7 vini e India con 9 vini.
Anche nella distribuzione dei vini per paese di produzione si ha un'istogramma con una coda veramente lunga e un divario enorme tra i primi posti e i posti medi.



ANALISI DEI PUNTEGGI DEI VINI PER PAESE DI ORIGINE

Nell'analisi dei punteggi dei vini per paese, dimostra come il paese abbia dei vini di qualità. Similmente dal grafico che mostra il numero di vini per paese in cui US con 54504 vini, i paesi con il punteggio maggiore pari a 100 sono l'Australia che conta 2329 vini, US, Portogallo, Italia e Francia.
Facendo la correlazione tra i 2 grafici possiamo dimostrare che l'Australia, US, Portogallo, Italia e Francia sono i maggiori produttori di vini e i vini che producono sono prodotti di qualità con punteggi pari al 100 in tutte le recensioni su di essi.

```
#ANALISI SUI PAESI CONI I VINI PIU' APPREZZATI
higher_points=dataset.groupby(['country'])['points'].max().sort_values(ascending=False).to_frame()
plt.figure(figsize=(12,3))
plt.bar(higher_points.index, higher_points['points'], width = 0.5, color = 'purple', label='Scores')
plt.legend()
plt.xticks(higher_points.index, rotation=90)
plt.xlabel('Country',fontsize = 10)
plt.ylabel('Score', fontsize = 10)
plt.title('Scoring the countrys wines',fontsize = 15)
plt.grid(color = 'black', linewidth = 0.2)
plt.show()
```



CREAZIONE DEL CATALOGO

CREAZIONE DEL CATALOGO E PROSPETTIVA DI INVESTIMENTO

Dopo aver analizzato approfonditamente i dati all'interno del dataFrame, abbiamo creato il catalogo dei vini consigliati, filtrando i dati attraverso dei parametri specifici per la vendita sul maketplace.
Scegliendo solo vini di qualità e ben apprezzati a cui è assegnato più del 95.00 di score e il prezzo oscilla tra i 20€ e i 400€, un range di prezzi
accessibile per i vini internazionali
di alta qualità.
Dimostro che un marketplace che punta alla vendita di vini internazionali di alta qualità deve acquistare 716 bottiglie di vino con prezzo medio di 126,75€ e investire in totale 90755,00€.

```
#CREAZIONE DEL CATALOGO E PREVISIONI DI INVESTIMENTO
max_price=400
min_price=20
min_score=95
wine_investment=80
prices_investiment=0
#CATALOGO
wines_for_purchase=dataset[(dataset['points']>min_score) & ((dataset['price']>=min_price)
                                                            & (dataset['price']<=max_price))]
#NUMERO VINI IN CATALOGO
num_bottle_wine_purchase=wines_for_purchase.shape[0]
#CALCOLO INVESTIMENTO GLOBALE
for i in wines_for_purchase['price']:
    prices_investiment+=i

print(f'Numero consigliato di bottiglie da acquistare: {num_bottle_wine_purchase}')
print(f'Investimento Totale previsto: {round(prices_investiment,2)}€')
print(f'Prezzo medio per bottiglia di vino: {round(prices_investiment/num_bottle_wine_purchase,2)}€')

wines_for_purchase
```

```
Numero consigliato di bottiglie da acquistare: 716
Investimento Totale previsto: 90755.0€
Prezzo medio per bottiglia di vino: 126.75€
```

CREAZIONE DEL CATALOGO E PROSPETTIVA DI INVESTIMENTO

	country	description	designation	points	price	title	variety	winery	year
345	Australia	This wine contains some material over 100 year...	Rare	100	350.0	Chambers Rosewood Vineyards NV Rare Muscat (Ru...	Muscat	Chambers Rosewood Vineyards	0
346	Australia	This deep brown wine smells like a damp, mossy...	Rare	98	350.0	Chambers Rosewood Vineyards NV Rare Muscadelle...	Muscadelle	Chambers Rosewood Vineyards	0
348	Australia	Deep mahogany. Dried fig and black tea on the ...	Grand	97	100.0	Chambers Rosewood Vineyards NV Grand Muscat (R...	Muscat	Chambers Rosewood Vineyards	0
349	Australia	RunRig is always complex, and the 2012 doesn't...	RunRig	97	225.0	Torbreck 2012 RunRig Shiraz-Viognier (Barossa)	Shiraz-Viognier	Torbreck	2012
350	Italy	After a few minutes in the glass, this stunnin...	Vignolo Riserva	97	150.0	Cavallotto 2010 Vignolo Riserva (Barolo)	Nebbiolo	Cavallotto	2010
...	...	...	...	...	...	...	...	...	...
128267	US	An extraordinary wine. It's the essence of Rus...	Gold Ridge	97	85.0	Semper 2007 Gold Ridge Pinot Noir (Russian Riv...	Pinot Noir	Semper	2007
128268	US	A monumental Cabernet that succeeds on every l...	Estate	97	50.0	Trefethen 2005 Estate Cabernet Sauvignon (Oak ...	Cabernet Sauvignon	Trefethen	2005
128269	US	Massive, dramatic, beautiful, intense, but you...	Cabernet Sauvignon	97	100.0	Hestan 2006 Cabernet Sauvignon (Napa Valley)	Cabernet Sauvignon	Hestan	2006

CONCLUSIONI

RIFLESSIONI FINALI

Tramite l'analisi del dataset abbiamo avuto la possibilità di capire come un vino costoto debba obbligatoriamente avere un punteggio alto ed essere considerato un vino di qualità. Abbiamo analizzato i paesi in correlazione al numero e alla qualità dei suoi vini; le varietà più recensite, quelle più costose e il prezzo medio delle varietà più costose; abbiamo anche analizzato i prezzi andando a dimostrare il divario e gli outlers rispetto al prezzo medio del vino.
Infine abbiamo progetto un potetico investimento per un marketplace che vorrebbe vendere vini di qualità provenienti da tutto il mondo e con un prezzo non eccessivamente sia ridotto che elevatocalcolando la spesa secondo i criteri scelti.
Questa analisi può aiutare il venditore di vino a prendere delle decisioni sui futuri ordini o sulle future scelte commerciali, capendo approfonditamente i gusti più apprezzati, i paesi di origine con una buona qualità di vino e i prezzi medi per del vino di buona qualità.
È stato un esempio di come l'analisi dei dati può essere utilizzata per prendere decisioni nel mondo reale.
In conclusione, questo progetto dimostra il potenziale dell'analisi dei dati nel prendere decisioni informate e personalizzate.