CS513 winery data profiling with cleaned data

1. Set up SQL in notebook

Prerequisite commands to run:

`pip3 install pandas`

`pip3 install ipython-sql`

`pip3 install matplotlib`

In [1]:
```
# Import required libraries
import sqlite3
import pandas as pd
import os
import matplotlib.pyplot as plt
```

In [2]:
```
# Connect to SQLite database
conn = sqlite3.connect(r'cs513_final_project_profiling_clean.db')

# Load CSV data into Pandas DataFrame
winery_ = pd.read_csv('winemag-data-cleaned.csv')

# Write the data to a sqlite table
winery_.to_sql('winery', conn, if_exists='replace', index=False)
```

Out[2]: 118782

In [3]:
```
# Set up ipython-sql
%load_ext sql
# %reload_ext sql
winery_db_url = 'sqlite:///' + os.path.expanduser('cs513_final_project_profi
%sql $winery_db_url
```

1.1 Validate database table is created and schema

In [4]:
```
%%sql
SELECT sql FROM sqlite_schema WHERE name='winery';
```

```
 * sqlite:///cs513_final_project_profiling_clean.db
Done.
```

Out[4]:

sql
CREATE TABLE "winery" ("Unnamed: 0" INTEGER, "country" TEXT, "description" TEXT, "designation" TEXT, "points" INTEGER, "price" REAL, "province" TEXT, "region_1" TEXT, "region_2" TEXT, "taster_name" TEXT, "taster_twitter_handle" TEXT, "title" TEXT, "variety" TEXT, "winery" TEXT, "price_imputeCountry" REAL, "price_imputeProvince" REAL, "price_imputeVariety" REAL)

In [5]:
```
%%sql
SELECT * FROM winery LIMIT 3;
```

```
 * sqlite:///cs513_final_project_profiling_clean.db
Done.
```

Out[5]:

Unnamed: 0	country	description	designation	points	price	province	region_1	re
0	Italy	Aromas include tropical fruit, broom, brimstone and dried herb. The palate isn't overly expressive, offering unripened apple, citrus and dried sage alongside brisk acidity.	Vulkà Bianco	87	35.0	Sicily & Sardinia	Etna	
1	Portugal	This is ripe and fruity, a wine that is smooth while still structured. Firm tannins are filled out with juicy red berry fruits and freshened with acidity. It's already drinkable, although it will certainly be better from 2016.	Avidagos	87	15.0	Douro	None	
2	US	Tart and snappy, the flavors of lime flesh and rind dominate. Some green pineapple pokes through, with crisp acidity underscoring the flavors. The wine was all stainless-	None	87	14.0	Oregon	Willamette Valley	Will

Unnamed: 0	country	description	designation	points	price	province	region_1	re
		steel fermented.						

2. Statistic of the data

In [6]:
```
# Total count of records in the winery table
total_count_sql_result = %sql SELECT COUNT(*) as total_count FROM winery;
total_count = total_count_sql_result[0].total_count

print(f'total count of records in table winery: {total_count}.')
```

```
 * sqlite:///cs513_final_project_profiling_clean.db
Done.
total count of records in table winery: 118782.
```

2.1 Null field count and percentage

In [7]:
```
%%capture
country_null_count_sql_result = %sql SELECT COUNT(*) as country_null_count F

description_null_count_sql_result = %sql SELECT COUNT(*) as description_null

designation_null_count_sql_result = %sql SELECT COUNT(*) as designation_null

points_null_count_sql_result = %sql SELECT COUNT(*) as points_null_count FRO

price_null_count_sql_result = %sql SELECT COUNT(*) as price_null_count FROM

province_null_count_sql_result = %sql SELECT COUNT(*) as province_null_count

region_1_null_count_sql_result = %sql SELECT COUNT(*) as region_1_null_count

region_2_null_count_sql_result = %sql SELECT COUNT(*) as region_2_null_count

taster_name_null_count_sql_result = %sql SELECT COUNT(*) as taster_name_null

taster_twitter_handle_null_count_sql_result = %sql SELECT COUNT(*) as taster

title_null_count_sql_result = %sql SELECT COUNT(*) as title_null_count FROM

variety_null_count_sql_result = %sql SELECT COUNT(*) as variety_null_count F

winery_null_count_sql_result = %sql SELECT COUNT(*) as winery_null_count FRO
;

def print_null_count(sql_result, field_name):
    null_count = sql_result[0][0]
    print(f'count of null value for {field_name} column: {null_count}, perce
```

In [8]:
```
# Stat of null or empty value for each field
print_null_count(country_null_count_sql_result,'country')
print_null_count(description_null_count_sql_result,'description')
print_null_count(designation_null_count_sql_result,'designation')
print_null_count(points_null_count_sql_result,'points')
print_null_count(price_null_count_sql_result,'price')
print_null_count(province_null_count_sql_result,'province')
print_null_count(region_1_null_count_sql_result,'region_1')
print_null_count(region_2_null_count_sql_result,'region_2')
print_null_count(taster_name_null_count_sql_result,'taster_name')
print_null_count(taster_twitter_handle_null_count_sql_result,'taster_twitter
print_null_count(title_null_count_sql_result,'title')
print_null_count(variety_null_count_sql_result,'variety')
print_null_count(winery_null_count_sql_result,'winery')
```

```
count of null value for country column: 0, percentage: 0.0%
count of null value for description column: 0, percentage: 0.0%
count of null value for designation column: 34216, percentage: 28.806%
count of null value for points column: 0, percentage: 0.0%
count of null value for price column: 0, percentage: 0.0%
count of null value for province column: 0, percentage: 0.0%
count of null value for region_1 column: 19379, percentage: 16.315%
count of null value for region_2 column: 72270, percentage: 60.843%
count of null value for taster_name column: 0, percentage: 0.0%
count of null value for taster_twitter_handle column: 29215, percentage: 24.5
95%
count of null value for title column: 0, percentage: 0.0%
count of null value for variety column: 0, percentage: 0.0%
count of null value for winery column: 0, percentage: 0.0%
```

2.2 Statistic of Numeric field

2.2.1 Price

In [9]:
```
price_stat = %sql SELECT MIN(price) as price_min, MAX(price) as price_max, A

print(f'price minimum value: {price_stat[0][0]}, maximum value: {price_stat[
```

```
 * sqlite:///cs513_final_project_profiling_clean.db
Done.
price minimum value: 4.0, maximum value: 155.0, average value: 33.07111346837
0626
```

2.2.2 Points

In [10]:
```
points_stat = %sql SELECT MIN(points) as points_min, MAX(points) as points_m

print(f'points minimum value: {points_stat[0][0]}, maximum value: {points_st
```

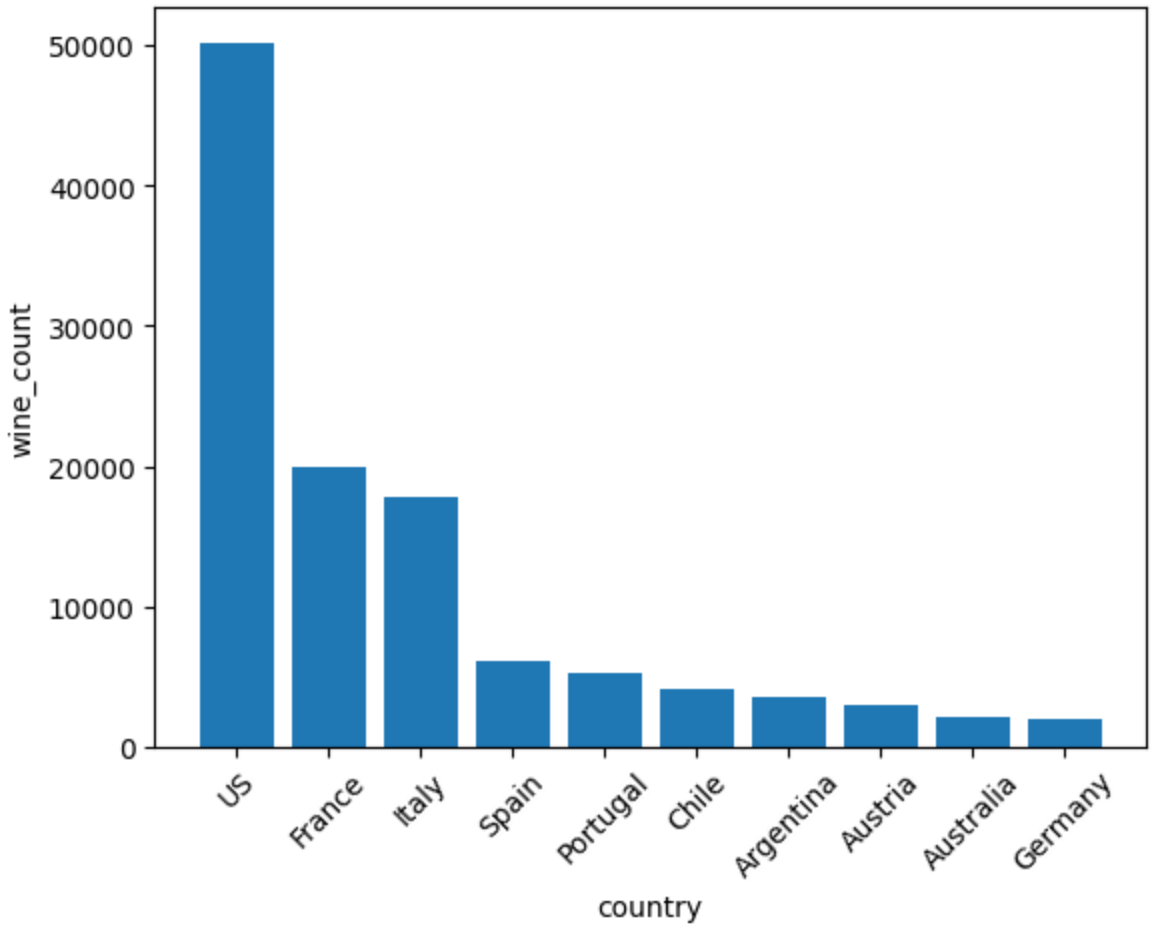
```
 * sqlite:///cs513_final_project_profiling_clean.db
Done.
points minimum value: 80, maximum value: 100, average value: 88.3899328181037
6
```

2.3 Statistic of Non-numeric field

2.3.1. top 10 Country with most wine records

In [11]:
```
country_count_result = %sql SELECT country, count(*) as wine_count FROM wine
pie_plot = country_count_result.bar()
```

```
 * sqlite:///cs513_final_project_profiling_clean.db
Done.
```

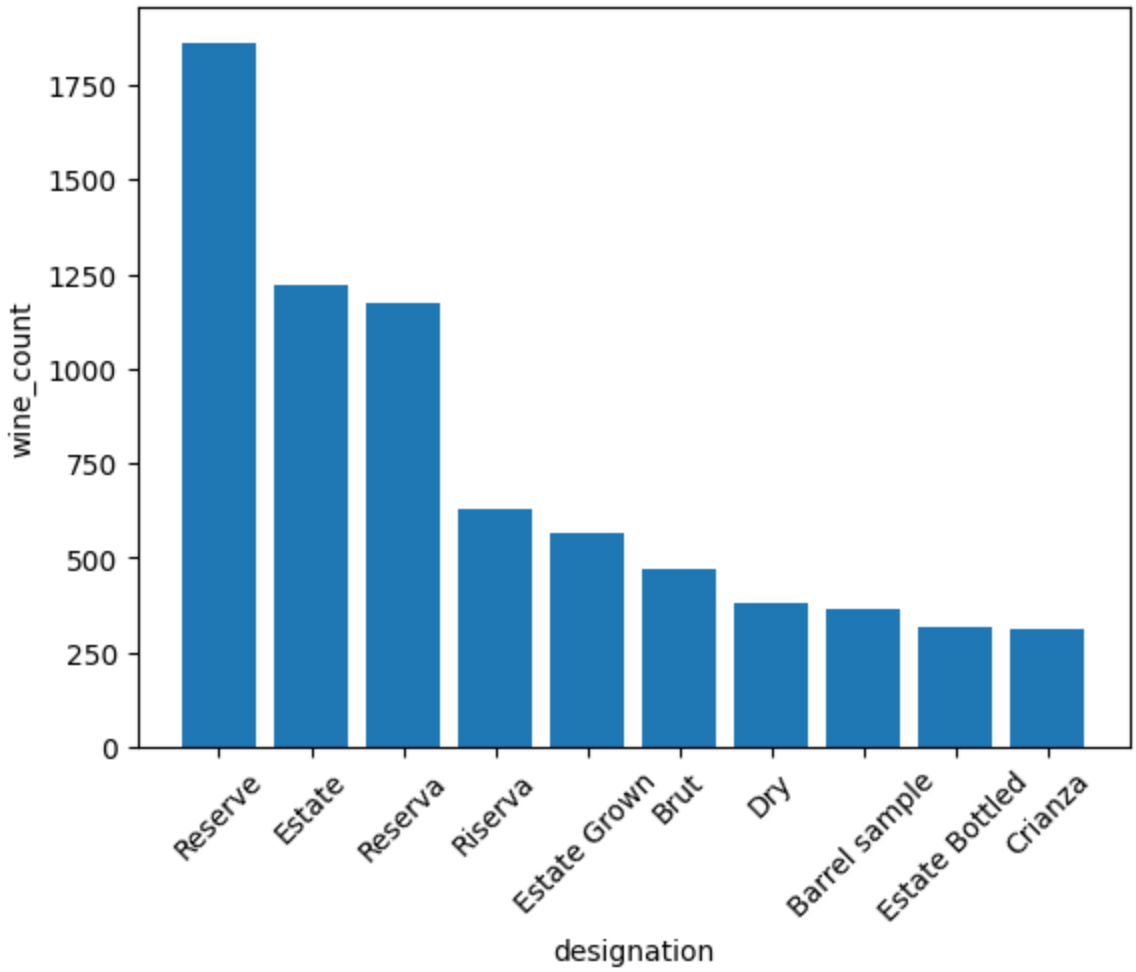


2.3.2. top 10 designation with most wine records¶

In [12]:
```
designation_count_result = %sql SELECT designation, count(*) as wine_count F
designation_count_result.bar()
```

```
 * sqlite:///cs513_final_project_profiling_clean.db
Done.
```

Out[12]:
```
<BarContainer object of 10 artists>
```

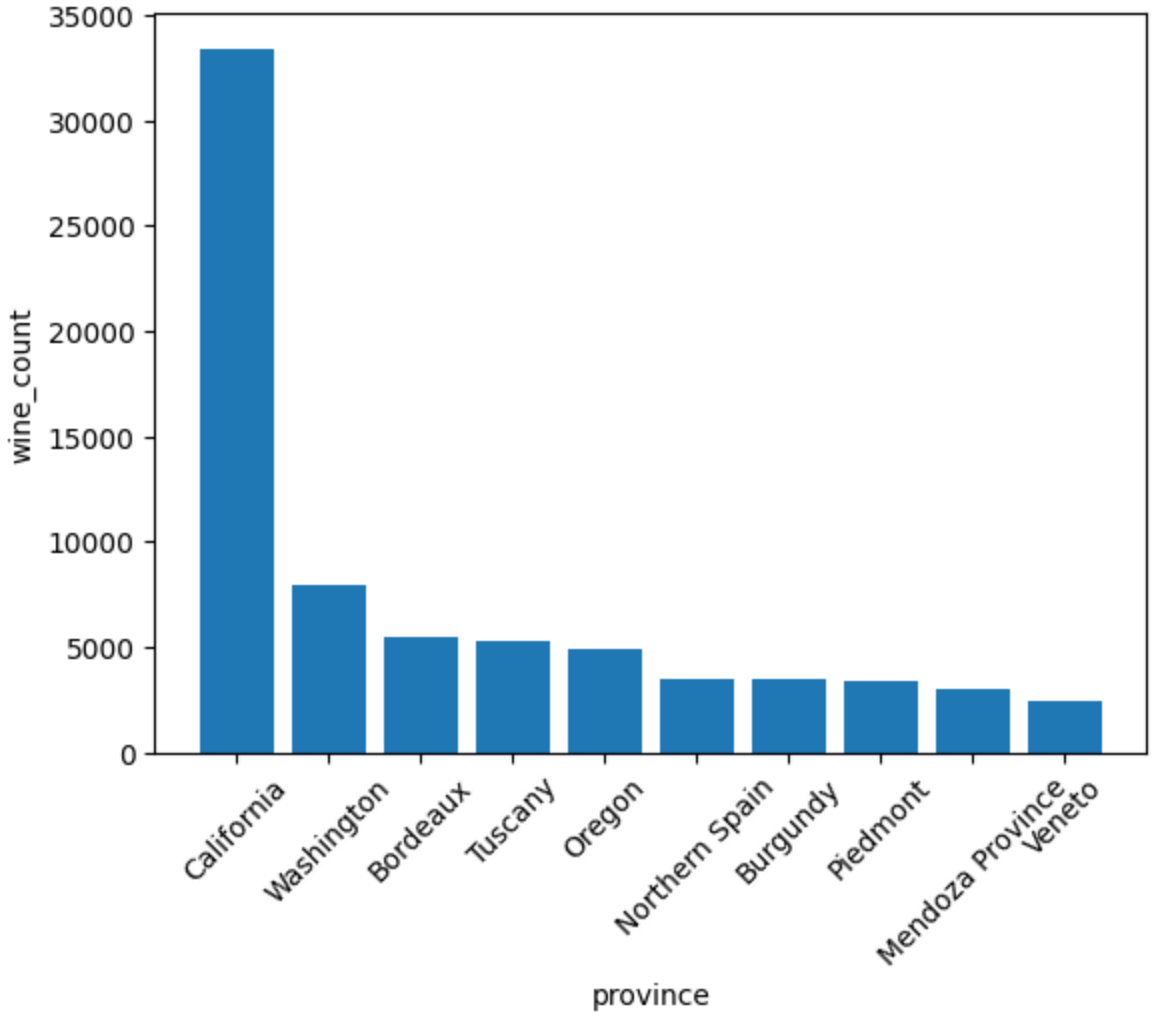


2.3.3. top 10 Province with most wine records

In [13]:
```
province_count_result = %sql SELECT province, count(*) as wine_count FROM wi
province_count_result.bar()
```

```
 * sqlite:///cs513_final_project_profiling_clean.db
Done.
```

Out[13]: <BarContainer object of 10 artists>

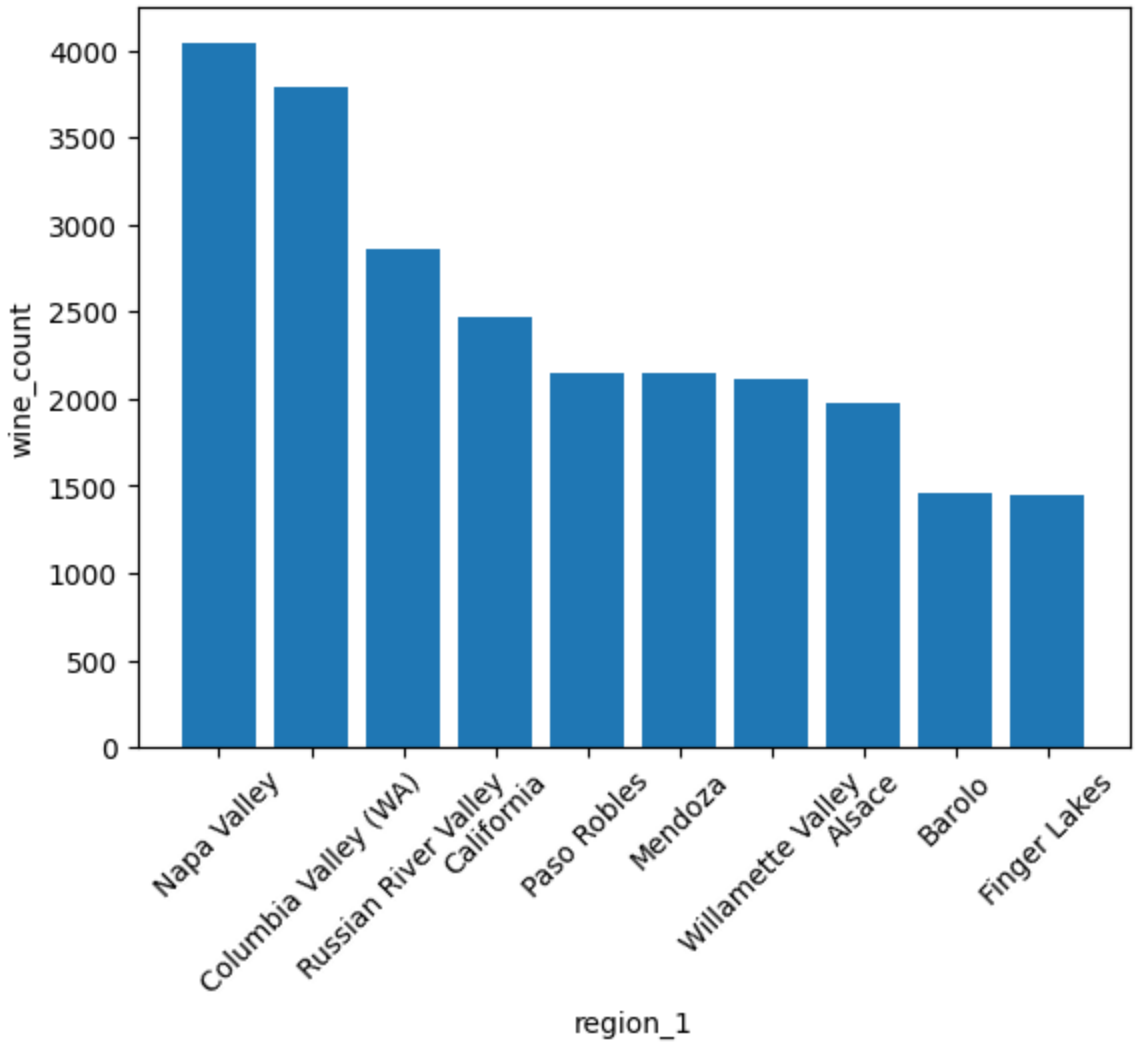


2.3.4. top 10 region_1 with most wine records¶

In [14]:
```
region_1_count_result = %sql SELECT region_1, count(*) as wine_count FROM wi
region_1_count_result.bar()
```

```
 * sqlite:///cs513_final_project_profiling_clean.db
Done.
```

Out[14]: <BarContainer object of 10 artists>

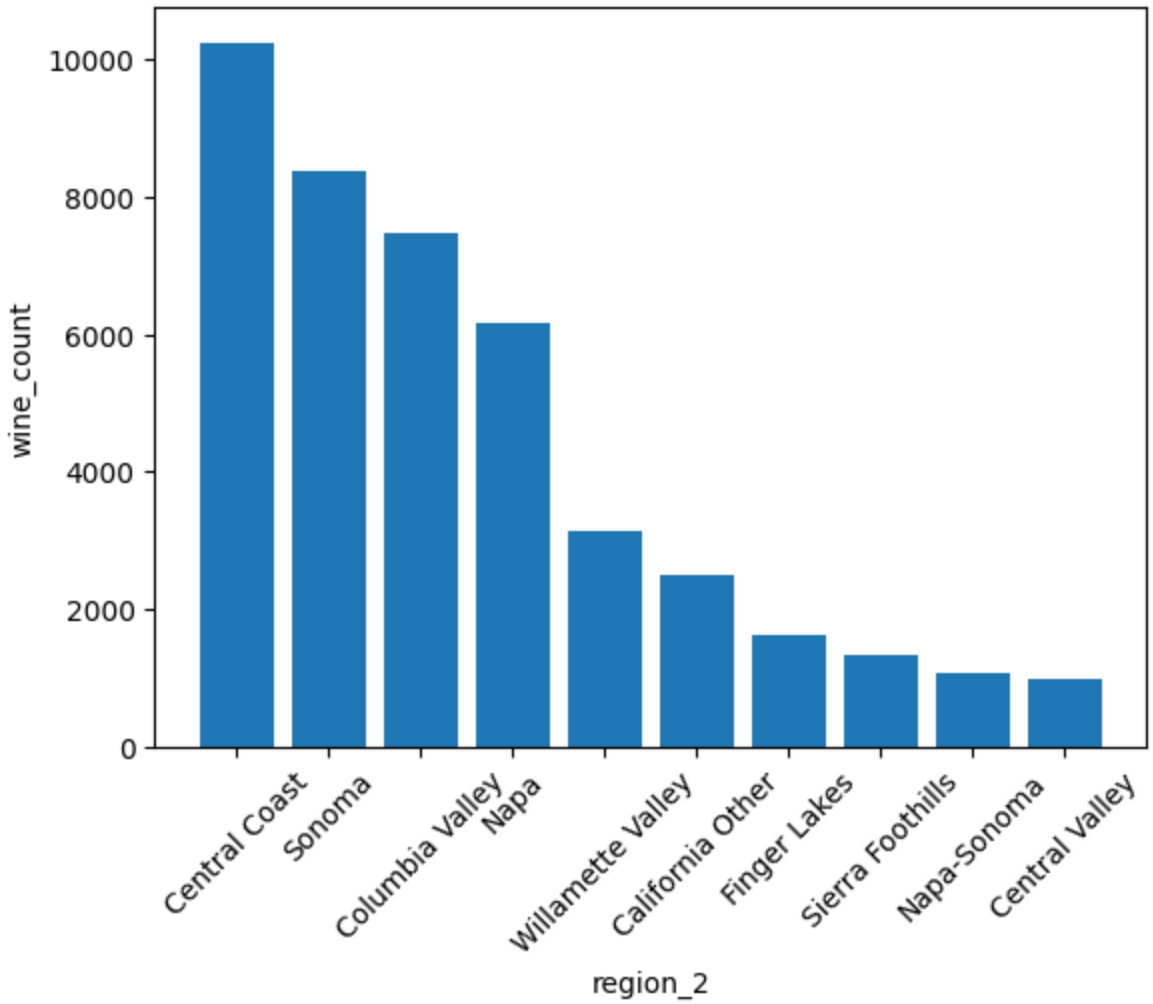


2.3.5. top 10 region_2 with wine records

In [15]:
```
region_2_count_result = %sql SELECT region_2, count(*) as wine_count FROM wi
region_2_count_result.bar()
```

```
 * sqlite:///cs513_final_project_profiling_clean.db
Done.
```

Out[15]: <BarContainer object of 10 artists>

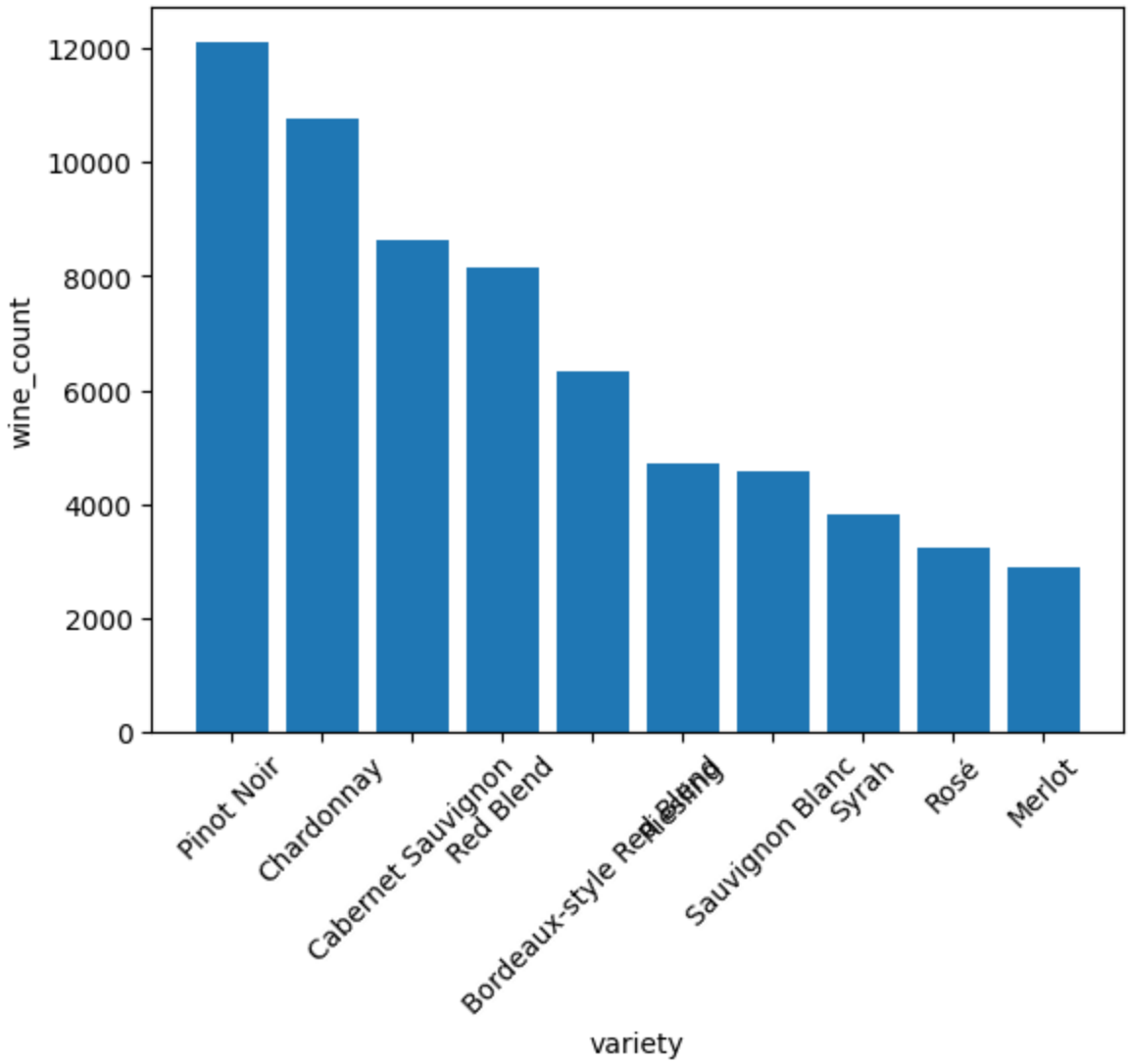


2.3.6. top 10 variety with most wine records¶

In [16]:
```
variety_count_result = %sql SELECT variety, count(*) as wine_count FROM wine
variety_count_result.bar()
```

```
 * sqlite:///cs513_final_project_profiling_clean.db
Done.
```

Out[16]: <BarContainer object of 10 artists>

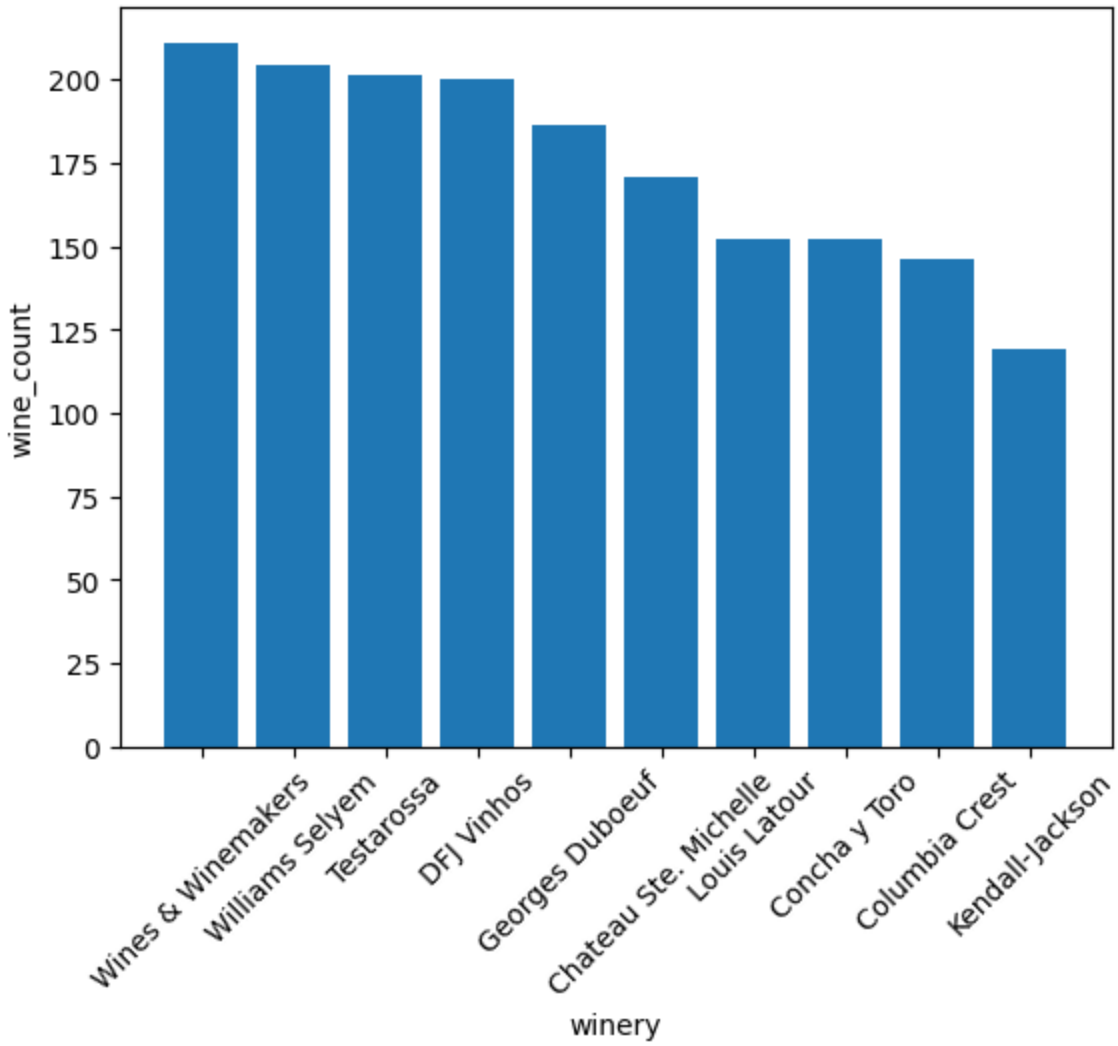


2.3.7. top 10 winery with most wine records¶

In [17]:
```
winery_count_result = %sql SELECT winery, count(*) as wine_count FROM winery
winery_count_result.bar()
```

```
 * sqlite:///cs513_final_project_profiling_clean.db
Done.
```

Out[17]: <BarContainer object of 10 artists>



3. Discovery of integrity constraint violations

In [18]:
```
def constraint_violation_helper(constraint_violation_type, constraint_violat
    print(f'{constraint_violation_type}: {constraint_violation_count}, perce
```

3.1 Key constraint

for given wine title and given reviewer and reviewer's twitter account, there should be no more than 1 review

In [19]:
```
key_constraint_violation_sql_result = %sql select count(*) as key_constrait_
constraint_violation_helper('key_constraint_violation_count', key_constraint
```

```
 * sqlite:///cs513_final_project_profiling_clean.db
Done.
key_constraint_violation_count: 847, percentage: 0.713%
```

3.2 Functional dependency

3.2.1 for given taster twitter_account, there should be no more than 1 taster

In [20]:
```
taster_func_dependency_violation_sql_result = %sql select count(*) from wine
constraint_violation_helper('taster_func_dependency_violation_count', taster

%sql select taster_twitter_handle as twitter_handle_function_dependency_viol
```

```
 * sqlite:///cs513_final_project_profiling_clean.db
Done.
taster_func_dependency_violation_count: 862, percentage: 0.726%
 * sqlite:///cs513_final_project_profiling_clean.db
Done.
```

Out[20]:

twitter_handle_function_dependency_violation
@worldwineguys

3.2.2 for given winery, the combination of province and country should no more than 1

In [21]:
```
winery_func_dependency_violation_sql_result = %sql select count(*) from wine
constraint_violation_helper('winery_func_dependency_violation_count', winery

%sql select winery as winery_function_dependency_violation_top_10 from (sele
```

```
 * sqlite:///cs513_final_project_profiling_clean.db
Done.
winery_func_dependency_violation_count: 23918, percentage: 20.136%
 * sqlite:///cs513_final_project_profiling_clean.db
Done.
```

Out[21]:

winery_function_dependency_violation_top_10
18401 Cellars
1848 Winery
3 Horse Ranch Vineyards
A-Mano
A. Parparoussis
A.A. Badenhorst Family Wines
Abarbanel
Achaia Clauss
Ackerman
Acordeón

3.2.3 for given province, the number of country should be no more than 1

In [22]:
```
province_func_dependency_violation_sql_result = %sql select count(*) from (s
constraint_violation_helper('province_func_dependency_violation_count', prov
```

```
 * sqlite:///cs513_final_project_profiling_clean.db
Done.
province_func_dependency_violation_count: 0, percentage: 0.0%
```

3.2.4 for given title, the variety and designation should be no more than 1

In [23]:
```
title_func_dependency_violation_sql_result = %sql select count(*) from winer
constraint_violation_helper('title_func_dependency_violation_count', title_f
```

```
 * sqlite:///cs513_final_project_profiling_clean.db
Done.
title_func_dependency_violation_count: 488, percentage: 0.411%
```

3.3 Semantic constraint

3.3.1 Points value should be between 0 to 100

In [24]:
```
points_semantic_violation_sql_result = %sql select count(*) from winery wher
constraint_violation_helper('points_semantic_violation_count', points_semant
```

```
 * sqlite:///cs513_final_project_profiling_clean.db
Done.
points_semantic_violation_count: 0, percentage: 0.0%
```

3.3.2 Price value should be greater than 0

In [25]:
```
price_semantic_violation_sql_result = %sql select count(*) from winery where
constraint_violation_helper('price_semantic_violation_count', price_semantic
```

```
 * sqlite:///cs513_final_project_profiling_clean.db
Done.
price_semantic_violation_count: 0, percentage: 0.0%
```

3.4 Inclusion dependency

N/A

4. Clean up

In [26]:
```
# Close connection to SQLite database
conn.close()
```