ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ГОРОДА МОСКВЫ
ДОПОЛНИТЕЛЬНОГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
ЦЕНТР ПРОФЕССИОНАЛЬНЫХ КВАЛИФИКАЦИЙ И СОДЕЙСТВИЯ
ТРУДОУСТРОЙСТВУ
«ПРОФЕССИОНАЛ»

ИТОГОВАЯ АТТЕСТАЦИОННАЯ РАБОТА

на тему

«Анализ данных с использованием Python»

(на примере анализа данных исследуемого продукта)

слушателя Синицыной Олеси Григорьевны

группы № гз-032

программа профессиональной переподготовки (дополнительное профессиональное образование)

«Аналитик данных (Python)»

Москва, 2025

Цель исследования:

Цель данного проекта — выявить, какие признаки имеют наибольшее влияние на рейтинг вина, составляемого специализированным журналом Wine Enthusiast. Для анализа используется набор данных из Kaggle (профессиональное сообщество специалистов по обработке данных и машинному обучению). Датафрейм состоит из 13 признаков (2 числовых признака и 11 категориальных признаков).

Анализ данных предполагает последовательное выполнение следующих этапов исследования:
1. Предобработка данных
2. Исследовательский анализ данных
3. Составление структуры развития рынка вина каждого региона
4. Проведение исследования статистических показателей зависимости цены вина от рейтинга в регионе
5. Проверка гипотез
6. Выводы

Столбцы данных

- Страна - страна происхождения вина.
- Описание — описание вкусового профиля вина.
- Обозначение - виноградник-поставщик винограда для изготовления вина.
- Рейтинг - оценка вина специализированным журналом Wine Enthusiast, выраженная в баллах по шкале от 1 до 100.
- Цена - цена одной бутылки вина.
- Провинция — регион (провинция, штат) производства вина.
- Регион 1 — зона виноделия в регионе (например, долина Напа в Калифорнии).
- Регион 2 — (необязательно) терруар виноделия - более конкретная область в винодельческом регионе (например, Резерфорд в долине Напа).
- Разновидность — сорт винограда, используемый в производстве вина (например, Пино Нуар).
- Винодельня — производитель вина.

1.Предобработка данных

Подготовка данных к анализу (очистка данных, трансформация данных, дополнение, оптимизация):

- Заменить названия столбцов (привести к нижнему регистру).
- Преобразовать данные в нужные типы. Описать, в каких столбцах заменили тип данных и почему.
- Обработать пропуски при необходимости.
- Внести новый столбец "Континенты" в случае необходимости
  country_to_continent = {

'Italy':'Europe',
'Portugal':'Europe',
'US':'North America',
'Spain':'Europe',
'France':'Europe',
'Germany':'Europe',
'Argentina':'Latin America',
'Chile':'Latin America',
'Australia': 'Oceania',
'Austria': 'Europe',
'South Africa': 'Africa',
'New Zealand': 'Oceania',
'Israel': 'Asia',
'Hungary':'Europe',
'Greece':'Europe',
'Romania':'Europe',
'Mexico':'Latin America',
'Canada':'North America',
'Turkey': 'Asia',
'Czech Republic': 'Europe',
'Slovenia': 'Europe',
'Luxembourg': 'Europe',
'Croatia': 'Europe',
'Georgia':'Europe',
'Uruguay': 'Latin America',
'England': 'Europe',
'Lebanon': 'Asia',
'Serbia': 'Europe',
'Brazil': 'Latin America',
'Moldova': 'Europe',
'Morocco':'Africa',
'Peru':'Latin America',
'India':'Asia',
'Bulgaria':'Europe',
'Cyprus': 'Europe',
'Armenia':'Asia',
'Switzerland':'Europe',
'Bosnia and Herzegovina':'Europe',
'Ukraine':'Europe',

'Slovakia':'Europe',
'Macedonia':'Europe',
'China':'Asia',
'Egypt':'Africa'
}
Внести новый столбец "Цвет" в случае необходимости color = {
"Pinot Noir": "red",
"Red Blend": "red",
"Bordeaux-style Red Blend": "red",
"Sangiovese": "red",
"Riesling": "white",
"Syrah": "red",
"Merlot": "red",
"Chardonnay": "white",
"Sauvignon Blanc": "white",
"Albariño": "white",
"Shiraz": "red",
"Rosé": "rose",
"Vermentino": "white",
"Pinot Grigio": "white",
"Pinot Gris": "white",
"Cabernet Sauvignon": "red",
"Gamay": "red",
"Tinto del Pais": "red",
"Grenache": "red",
"Portuguese White": "white",
"Alicante Bouschet": "red",
"Tempranillo": "red",
"Pinot Noir-Gamay": "red",
"Moscato": "white",
"Chenin Blanc": "white",
"Cabernet Franc": "red",
"Monastrell-Syrah": "red",
"Rhône-style Red Blend": "red",
"Austrian white blend": "white",
"Nebbiolo": "red",
"White Blend": "white",
"Barbera": "red",
"Tempranillo Blend": "red",
"Nero d'Avola": "red",
"Zinfandel": "red",
etc

}

Импорт необходимых библиотек

In [10]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import matplotlib.cm as cm
import scipy.stats as st
from scipy import stats
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import r2_score, mean_squared_error
# импорт библиотеки warnings
import warnings
warnings.simplefilter(action='ignore', category=FutureWarning)
warnings.filterwarnings('ignore')
```

Загрузка данных в Pandas DataFrame.

In [11]:
```
df = pd.read_csv(r'/content/wine_reviews.csv')
```

Подсчет размерности данных (количество строк и стоблцов).

In [12]:
```
df.shape
```

Out[12]:
```
(20000, 10)
```

Представленная к исследованию таблица состоит из 10 колонок и 20000 строк.

Давайте посмотим на ее первые 5 строк с помощью команды df.head()

In [13]:
```
df.head()
```

Out[13]:

	country	description	designation	points	price	province	region_1	region_2	v
0	US	With a delicate, silky mouthfeel and bright ac...	NaN	86	23.0	California	Central Coast	Central Coast	Pinc
1	Italy	D'Alceo is a drop dead gorgeous wine that ooze...	D'Alceo	96	275.0	Tuscany	Toscana	NaN	Red
2	France	The great dominance of Cabernet Sauvignon in t...	NaN	91	40.0	Bordeaux	Haut-Médoc	NaN	Borc sty
3	Italy	The modest cherry, dark berry and black tea no...	NaN	81	15.0	Tuscany	Chianti Classico	NaN	Sang
4	US	Exceedingly light in color, scent and flavor, ...	NaN	83	25.0	Oregon	Rogue Valley	Southern Oregon	Pinc

Подсчет количества и процентного соотношения пропущенных значений в каждой переменной.

In [14]:
```
MissingValue = df.isnull().sum().sort_values(ascending = False)
Percent = (df.isnull().sum()/df.isnull().count()*100).sort_values(ascending = Fa
MissingData = pd.concat([MissingValue, Percent], axis=1, keys=['Пропущенные знач
MissingData
```

Out[14]:

	Пропущенные значения	Процент
region_2	11942	59.710
designation	6001	30.005
region_1	3457	17.285
price	1802	9.010
description	0	0.000
country	0	0.000
province	0	0.000
points	0	0.000
variety	0	0.000
winery	0	0.000

Критического количества пропусков не наблюдается. Колонки *region_1 и region_2* носят скорее **вспомогательно - информационный** характер, тогда как **ключевой** (в контексте распределения продукта по регионам), является колонка - *province*. В связи с этим пропуски, которые относятся к колонкам не первой значимости оставляем как есть, лишь для удобства дальнейшей обработки возможно заполню значениями 'unknown'.
Пропуски в колонке designation также не нанесут ущерь дальнейшим исследованиям.
Пропуски в колонке price важно вдумчиво обработать, опираясь на дальнейшие цели исследования и количество пропусков.
Так как процент пропущенных значений составляет менее 10 от всего количества, то в случае удаления в данной ситуации критических искажений не будет. Поэтому было принято решение удалить эти значения, так как импутация нулевыми значениями привела бы к ощутимому искажению статистики ценовой политики. При этом импутация арифметически средними значениями или медианой носит сомнительный характер, так как ценообразование вина зависит от многих факторов:

- время выдержки(ординарное, марочное, коллекционное)
- сорт винограда;
- технология производства;
- регион производства;
- история винодельни;
- известность вина и его уникальные органолептические свойства;
- возраст вина и многое другое.

В связи с чем альтернативой удаления, может быть только поиск истинных цен на каждый экземпляр.

В представленных выше выводах предлагаю убедиться в том числе и визуально. На рисунке 1 продемонстрирована матрица пропущенных значений, где темным

цветом обозначены отсутствующие данные, а светлым - присутствующие.

In [15]:
```
colours = ['#DDA0DD', '#2F4F4F']
sns.heatmap(df.isnull(), cmap=sns.color_palette(colours))
# Decorations
plt.title('Матрица пропущенных значений набора данных', fontsize=14)
plt.xticks(fontsize=12)
plt.yticks(fontsize=12)
plt.figtext(0.5, -0.2, "Рисунок 1. - Матрица пропущенных значений набора данных"
plt.show()
```

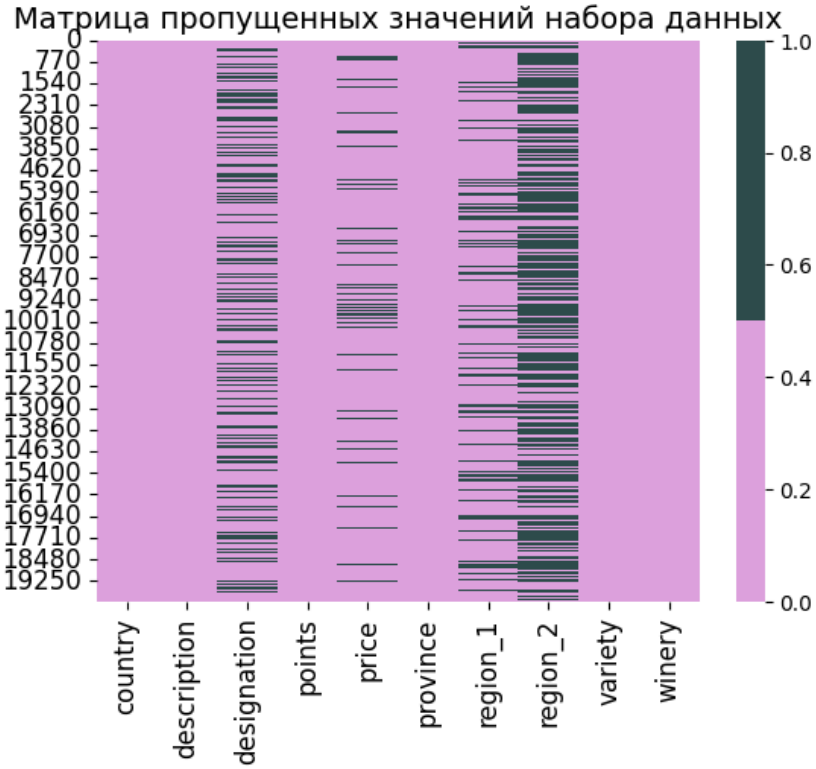

Рисунок 1. - Матрица пропущенных значений набора данных

In [16]:
```
df_clean = df.dropna(subset=['price']).copy()
df_clean1 = df_clean.fillna('Unknown')
MissingValue = df_clean1.isnull().sum().sort_values(ascending = False)
Percent = (df_clean1.isnull().sum()/df_clean.isnull().count()*100).sort_values(a
MissingData = pd.concat([MissingValue, Percent], axis=1, keys=['Пропущенные знач
MissingData
```

Out[16]:

	Пропущенные значения	Процент
country	0	0.0
description	0	0.0
designation	0	0.0
points	0	0.0
price	0	0.0
province	0	0.0
region_1	0	0.0
region_2	0	0.0
variety	0	0.0
winery	0	0.0

Вывод типа данных каждой переменной.

Преобразование типов данных при необходимости.

In [17]:
```
df_clean1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 18198 entries, 0 to 19999
Data columns (total 10 columns):
 #   Column       Non-Null Count  Dtype  
---  ------       --------------  -----  
 0   country      18198 non-null  object 
 1   description  18198 non-null  object 
 2   designation  18198 non-null  object 
 3   points       18198 non-null  int64  
 4   price        18198 non-null  float64
 5   province     18198 non-null  object 
 6   region_1     18198 non-null  object 
 7   region_2     18198 non-null  object 
 8   variety      18198 non-null  object 
 9   winery       18198 non-null  object 
dtypes: float64(1), int64(1), object(8)
memory usage: 1.5+ MB
```

Преобразование не требуется ввиду того, что содержание соответствует каждому типу данных.

Обработака дубликатов происходит таким образом:

- проверка их наличия;
- удаление дубликатов.

In [18]:
```
df_clean1.duplicated().sum()
```

Out[18]:
```
np.int64(1005)
```

In [19]:
```
df_clean1 = df_clean1.drop_duplicates(keep = "first")
```

Убеждаемся, что в перезаписаном датафрейме отсутствуют дубликаты:

In [20]:
```
df_clean1.duplicated().sum()
```

Out[20]:
```
np.int64(0)
```

In [21]:
```
df_clean1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 17193 entries, 0 to 19999
Data columns (total 10 columns):
 #   Column       Non-Null Count  Dtype  
---  ------       --------------  -----  
 0   country      17193 non-null  object 
 1   description  17193 non-null  object 
 2   designation  17193 non-null  object 
 3   points       17193 non-null  int64  
 4   price        17193 non-null  float64
 5   province     17193 non-null  object 
 6   region_1     17193 non-null  object 
 7   region_2     17193 non-null  object 
 8   variety      17193 non-null  object 
 9   winery       17193 non-null  object 
dtypes: float64(1), int64(1), object(8)
memory usage: 1.4+ MB
```

Во время процедуры очистки данных бывают ситуации, когда необходимо названия колонок, некоторые ключевые значения в строках привести к единому стилю.

В представленном датафрейме такая необходимость отсутствует.

Для дальнейшей исследовательской работы необходимо провести обогащение данных.

In [90]:
```
color = {
"Pinot Noir": "red",
"Red Blend": "red",
"Bordeaux-style Red Blend": "red",
"Sangiovese": "red",
"Riesling": "white",
"Syrah": "red",
"Merlot": "red",
"Chardonnay": "white",
"Sauvignon Blanc": "white",
"Albariño": "white",
"Shiraz": "red",
"Rosé": "rose",
"Vermentino": "white",
"Pinot Grigio": "white",
"Pinot Gris": "white",
"Cabernet Sauvignon": "red",
"Gamay": "red",
"Tinto del Pais": "red",
"Grenache": "red",
"Portuguese White": "white",
"Alicante Bouschet": "red",
```

```
}
```

In [91]:
```
country_to_continent = {
'Italy':'Europe',
'Portugal':'Europe',
'US':'North America',
'Spain':'Europe',
'France':'Europe',
'Germany':'Europe',
'Argentina':'Latin America',
'Chile':'Latin America',
'Australia': 'Oceania',
'Austria': 'Europe',
'South Africa': 'Africa',
'New Zealand': 'Oceania',
'Israel': 'Asia',
'Hungary':'Europe',
'Greece':'Europe',
'Romania':'Europe',
'Mexico':'Latin America',
'Canada':'North America',
'Turkey': 'Asia',
'Czech Republic': 'Europe',
'Slovenia': 'Europe',
'Luxembourg': 'Europe',
'Croatia': 'Europe',
'Georgia':'Europe',
'Uruguay': 'Latin America',
'England': 'Europe',
'Lebanon': 'Asia',
'Serbia': 'Europe',
'Brazil': 'Latin America',
'Moldova': 'Europe',
'Morocco':'Africa',
'Peru':'Latin America',
'India':'Asia',
'Bulgaria':'Europe',
'Cyprus': 'Europe',
'Armenia':'Asia',
'Switzerland':'Europe',
'Bosnia and Herzegovina':'Europe',
'Ukraine':'Europe',
'Slovakia':'Europe',
'Macedonia':'Europe',
'China':'Asia',
'Egypt':'Africa',
'Montenegro': 'Europe',
'South Korea': 'Asia',
'US-France': 'North America',
}
```

In [24]:
```
df_clean1['continent'] = df_clean1['country'].map(country_to_continent)
df_clean1['color'] = df_clean1['variety'].map(color)
```

Поверим наличие / отсутствие NaN значений в новых колонках:

In [25]:
```
df_clean1['continent'].isnull().sum()
```

Out[25]: np.int64(0)

In [26]:
```
df_clean1['color'].isnull().sum()
```

Out[26]: np.int64(30)

Некоторые цвета не присвоились. Попробую удалить пробелы, возможно есть не корректно введенные значения.

In [27]:
```
df_clean1['variety'] = df_clean1['variety'].str.strip()
```

Метод не сработал, было принято решение отсутствующие значения цвета заменить на 'unknown'.

In [28]:
```
df_clean1['color'] = df_clean1['color'].fillna('unknown')
df_clean1['color'].isnull().sum()
```

Out[28]: np.int64(0)

Для дальнейшей работы решила присвоить более удобное название датафрейму. А также в последний раз посмотреть на датафрейм перед анализом.

In [29]:
```
data = df_clean1.copy()
data.head()
```

Out[29]:

	country	description	designation	points	price	province	region_1	region_2	v
0	US	With a delicate, silky mouthfeel and bright ac...	Unknown	86	23.0	California	Central Coast	Central Coast	Pinc
1	Italy	D'Alceo is a drop dead gorgeous wine that ooze...	D'Alceo	96	275.0	Tuscany	Toscana	Unknown	Rec
2	France	The great dominance of Cabernet Sauvignon in t...	Unknown	91	40.0	Bordeaux	Haut-Médoc	Unknown	Bor sty
3	Italy	The modest cherry, dark berry and black tea no...	Unknown	81	15.0	Tuscany	Chianti Classico	Unknown	Sang
4	US	Exceedingly light in color, scent and flavor, ...	Unknown	83	25.0	Oregon	Rogue Valley	Southern Oregon	Pinc

ИТОГ ПЕРВОГО РАЗДЕЛА: ПОДГОТОВКА И ОЧИСТКА ДАННЫХ

Ключевые выполненные действия:

1. Очистка от пропущенных значений (NaN)

Был применен дифференцированный подход в зависимости от характера пропусков:

- Для категориальных переменных: заполнение специальными значениями;
- Для числовых переменных: удаление.

2. Удаление дубликатов

- Обнаружение и удаление полных дубликатов строк;
- Сохранение уникальных записей для обеспечения репрезентативности данных.

3. Обогащение данных

Были добавленны производные признаки:

1. Цвет вина на основе сорта (color). Аналитическая ценность : возможность анализировать предпочтения по цветам вин;

2. Принадлежность к континенту на основе страны происхождения (continent). Аналитическая ценность: сравнительный анализ между континентами и регионами.

Достигнутые результаты:
Расширение аналитических возможностей:

- Географический анализ - сравнение между континентами и странами
- Категориальный анализ - группировка по цвету вина
- Перекрестный анализ - комбинация географических и качественных признаков

Перспективы для дальнейшего исследования:

- Анализ ценовых распределений по континентам
- Исследование корреляции между цветом вина и рейтингом
- Сравнение средних цен по географическим регионам
- Выявление региональных особенностей виноделия

Готовность данных к анализу:
Датафрейм полностью очищен, обогащен и готов для проведения комплексного анализа с использованием статистических методов и визуализации данных.

Качество данных обеспечено для получения надежных и значимых результатов исследования.

2. Исследовательский анализ данных

- Найти среднюю цену вина по региону.
- Провести полную статистику по регионам.
- Проанализировать провинции по средней цене
- Выбрать сорта с наибольшими ценами.
- Определить, популярные сорта вина в бюджетном сегменте.
- Определить, какие сорта вина лидируют по рейтингам.
- Построить график «ящик с усами» по рейтингам в разбивке по странам.
- Построить график «ящик с усами» по рейтингам в разбивке по сортам вина.
- Выявить закономерность влияния на цену цвета и рейтинга вина.
- Построить диаграмму рассеяния и посчитать корреляцию.

__2.1 Средняя цена вина по региону.__

In [30]:
```
data.groupby(['province']).price.mean().round(2)
```

Out[30]:

province	price
Aconcagua Costa	20.00
Aconcagua Valley	41.06
Aegean	70.00
Alentejano	26.72
Alentejo	12.29
...	...
Western Cape	15.32
Württemberg	28.33
Zitsa	15.00
Štajerska	19.33
Župa	18.00

307 rows × 1 columns

dtype: float64

2.2 Полная статистика по провинциям

In [31]:
```
province_stats = data.groupby('province')['price'].agg([
    'mean', 'median', 'std', 'count', 'min', 'max'
]).round(2).sort_values('mean', ascending=False)
display(province_stats.head(10))
```

province	mean	median	std	count	min	max
Tokaji	133.10	61.0	223.87	10	20.0	764.0
Champagne	97.35	65.0	136.13	137	11.0	1400.0
Santa Cruz	95.00	95.0	NaN	1	95.0	95.0
Israel	70.00	70.0	NaN	1	70.0	70.0
Aegean	70.00	70.0	70.71	2	20.0	120.0
Burgundy	69.84	50.0	74.27	420	10.0	757.0
Wachau	68.60	38.0	168.54	40	13.0	1100.0
Middle and South Dalmatia	65.00	65.0	NaN	1	65.0	65.0
Martinborough Terrace	60.00	60.0	NaN	1	60.0	60.0
Rheingau	55.08	25.0	78.45	49	14.0	395.0

2.3 Визуализация топ-10 провинций по средней цене

In [32]:
```
top_provinces = (data.groupby('province')['price']
                 .agg(['mean', 'count'])
                 .query('count > 1')  # фильтруем где count > 1
                 .sort_values('mean', ascending=False)
                 .head(10))

plt.figure(figsize=(10, 5))
top_provinces.plot(kind='bar', color=['skyblue','darkblue'])
plt.xlabel('Рисунок 2.1 - Топ-10 провинций по средней цене вина')
plt.ylabel('Средняя цена ($)')
plt.xticks(rotation=45)
plt.grid(axis='y')
plt.show()
```

```
<Figure size 1000x500 with 0 Axes>
```

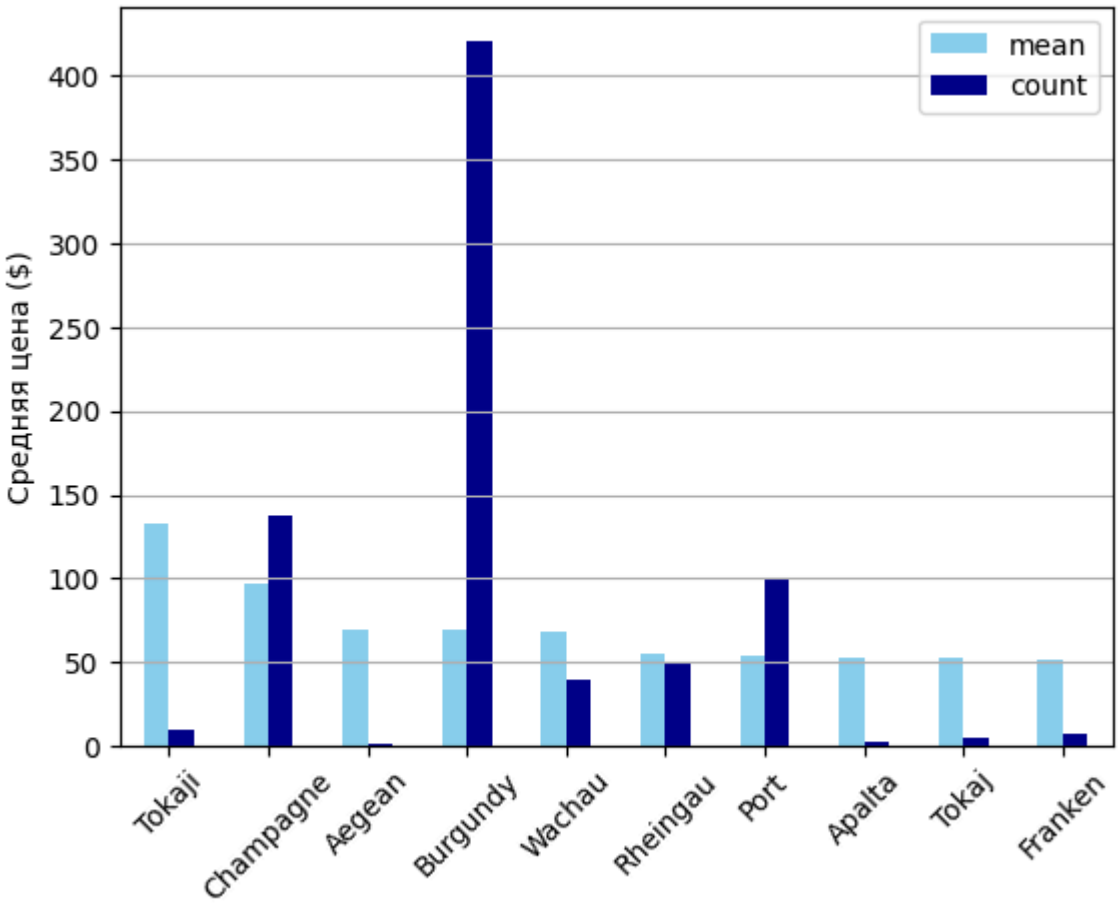

Рисунок 2.1 - Топ-10 провинций по средней цене вина

Исходя из полученных данных промежуточный вывод говорит о том, что по высокой средней цене лидируют регионы в Венгрии(Tokaji), Франции (Champagne, Burgundy).
Количество производимого вина(с одной из самых высоких средних цен), несомненно, принадлежит Burgundy(Франция).

2.4 Выбор сортов вина с наибольшими ценами

In [33]:
```
top_varieties = (data.groupby('variety')['price']
                 .mean()
                 .sort_values(ascending = True)
```

```
                  .head(10))

plt.figure(figsize=(12, 6))
top_varieties.plot(kind='bar', color='skyblue')
plt.xlabel('Рисунок 2.2 - Топ-10 сортов вина по цене')
plt.ylabel('Средняя цена ($)')
plt.xticks(rotation=45)
plt.grid(axis='y')
plt.show()
```

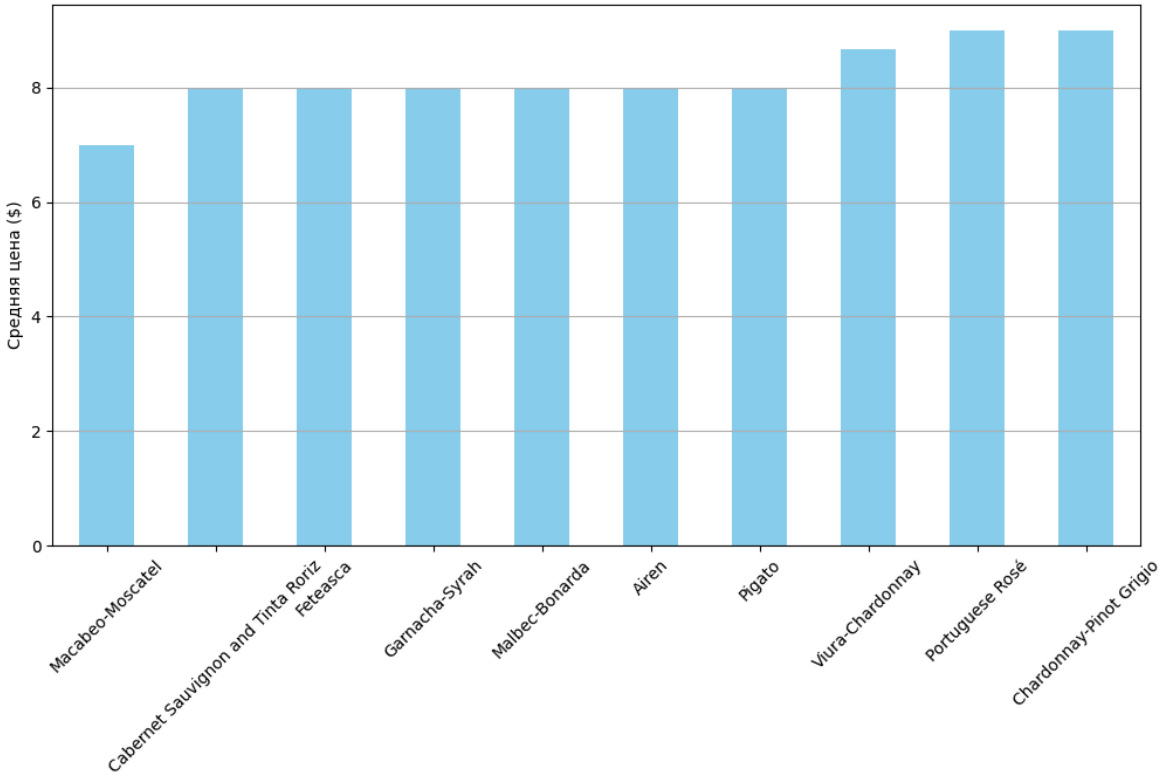

Рисунок 2.2 - Топ-10 сортов вина по цене

2.5 Определяем, популярные сорта вина в бюджетном сегменте.

In [34]:
```
budget_threshold = np.percentile(data['price'], 50)
budget_wines = data[data['price'] <= budget_threshold]
popular_budget_varieties = budget_wines['variety'].value_counts().head(10)

plt.figure(figsize=(8, 4))
popular_budget_varieties.plot(kind='bar', color='skyblue')
plt.xlabel('Рисунок 2.3 - Топ-10 популярных сортов винограда в бюджетном сегмент
plt.ylabel('Количество')
plt.xticks(rotation=45)
plt.show()
```

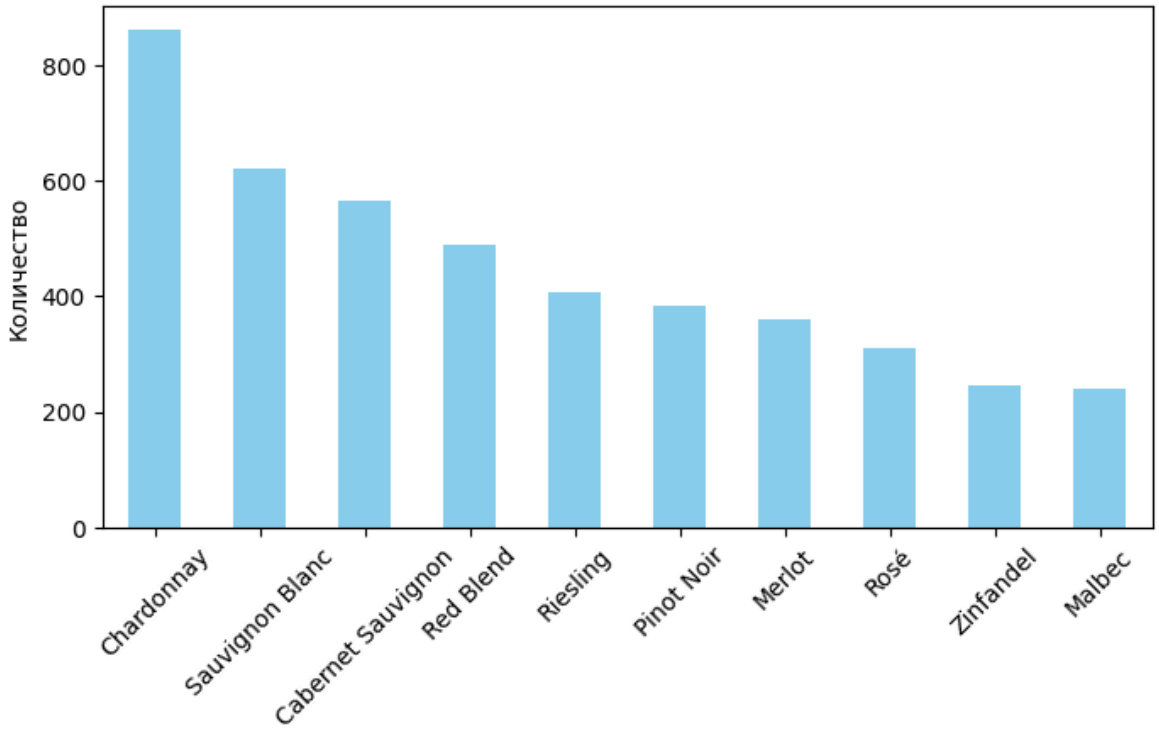

Рисунок 2.3 - Топ-10 популярных сортов винограда в бюджетном сегменте

Я взяла медианную цену (50 персентиль), это значение - 24$. Всем наблюдениям, которые ниже этого показателя присвоила категорию - 'бюджетный сегмент'. Тройка лидеров в таком сегменте - Шардоне, Совиньон Бланк, Каберне Совиньон.

__2.6 Определяем, какие сорта вина лидируют по рейтингам.__

In [35]:
```
top_rated_varieties = data.groupby('variety').filter(lambda x: len(x) >= 10)
top_rated = top_rated_varieties.groupby('variety')['points'].mean().sort_values(

plt.figure(figsize=(10, 8))
top_rated.plot(kind='bar', color='skyblue')
plt.xlabel('Рисунок 2.4 - Топ-10 сортов по среднему рейтингу (минимум 10 оценок)
plt.ylabel('Средний рейтинг')
plt.xticks(rotation=45)
plt.ylim(80, 100)
plt.show()
```

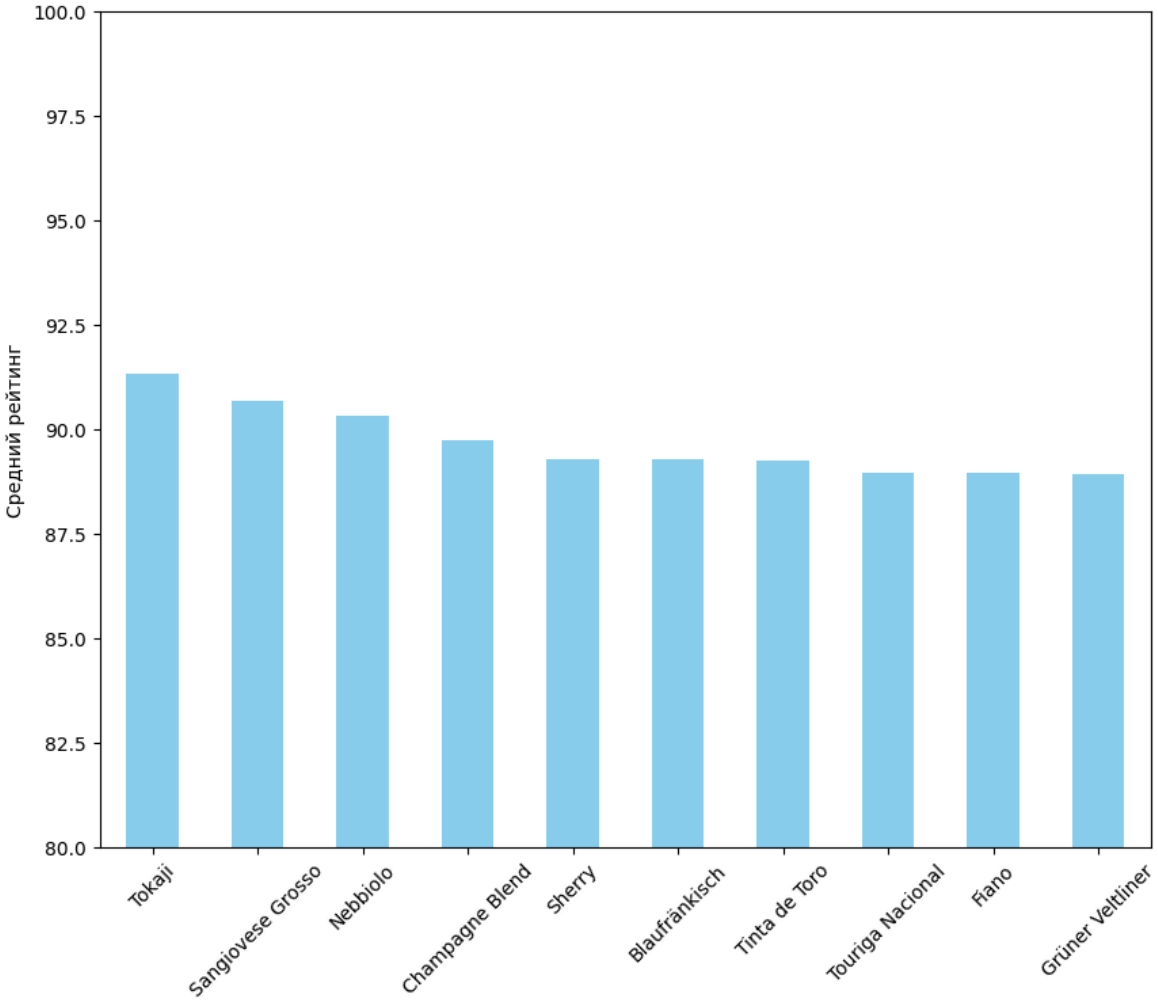

Рисунок 2.4 - Топ-10 сортов по среднему рейтингу (минимум 10 оценок)

Тройка лидеров в рейтинге наиболее высоко оцененных вин - Токай, Санджовезе Гроссо, Неббиоло.
Это абсолютно закономерный результат, так как эти вина являются национальным достоянием своих стран, производятся в ограниченных количествах и имеют самый высокий статус качества, что и отражается в их рейтингах и, соответственно, ценах.

2.7 Строим график «ящик с усами» по рейтингам в разбивке по странам.

In [36]:
```
top_countries = data['country'].value_counts().head(10).index
plt.figure(figsize=(15, 8))
data[data['country'].isin(top_countries)].boxplot(column='points', by='country',
plt.suptitle('')
plt.xlabel('Рисунок 2.5 - Распределение рейтингов по странам')
plt.ylabel('Страна')
plt.show()
```

```
<Figure size 1500x800 with 0 Axes>
```

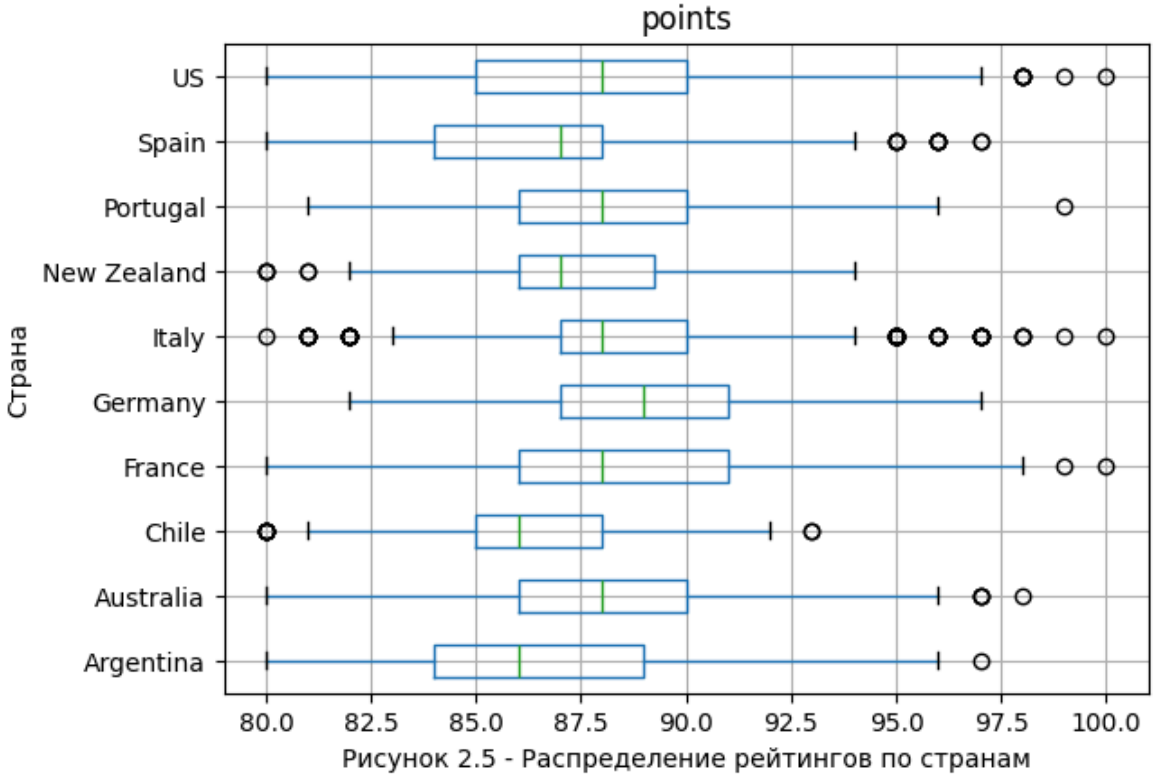

Рисунок 2.5 - Распределение рейтингов по странам

Общая картина говорит о том, что некоторые страны имеют небольшое лидерство в рейтингах, а именно - США, Германия, Франция. Немного хуже показатели в Аргентине, Испании(несмотря на выбросы) и Чили.

__2.8 Строим график «ящик с усами» по рейтингам в разбивке по сортам вина.__

In [37]:
```
top_varieties = data['variety'].value_counts().head(15).index

plt.figure(figsize=(15, 8))
data[data['variety'].isin(top_varieties)].boxplot(column='points', by='variety',
plt.suptitle('')
plt.xlabel('Рисунок 2.6 - Распределение рейтингов по сортам вин')
plt.ylabel('Сорт вина')
plt.show()
```

```
<Figure size 1500x800 with 0 Axes>
```

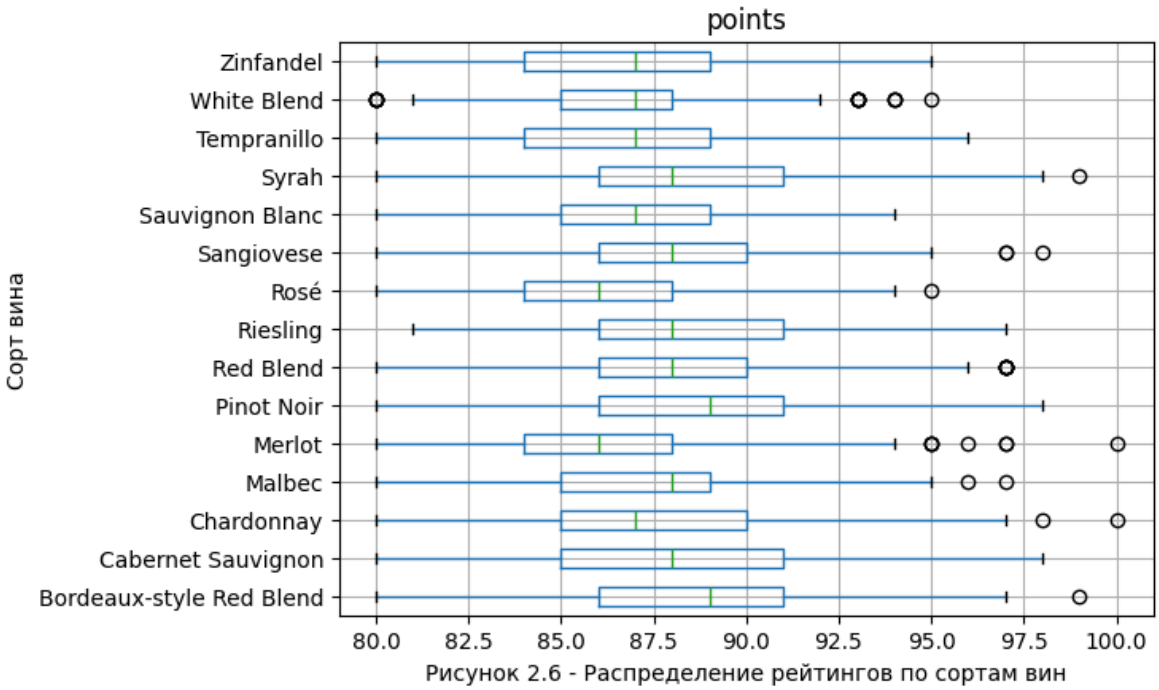

Рисунок 2.6 - Распределение рейтингов по сортам вин

Наиболее высокооцененные сорта вина - Сира(Шираз), Рислинг, Каберне Совиньон, Пино Нуар и это действительно стабильно качественные и признанные сорта. Они могут не иметь рекордного среднего рейтинга, но у них:

- Очень высокий медианный рейтинг (половина всех вин получает оценку выше этого значения).
- Мало низких выбросов (плохих вин).

__2.9 Выявляем закономерность влияния на цену цвета и рейтинга вина.__

In [38]:
```
plt.figure(figsize=(12, 8))
sns.scatterplot(data=data, x='points', y='price', hue='color', alpha=0.6)
plt.xlabel('Рисунок 2.7 - Влияние рейтинга и цвета вина на цену')
plt.ylabel('Цена ($)')
plt.show()
```

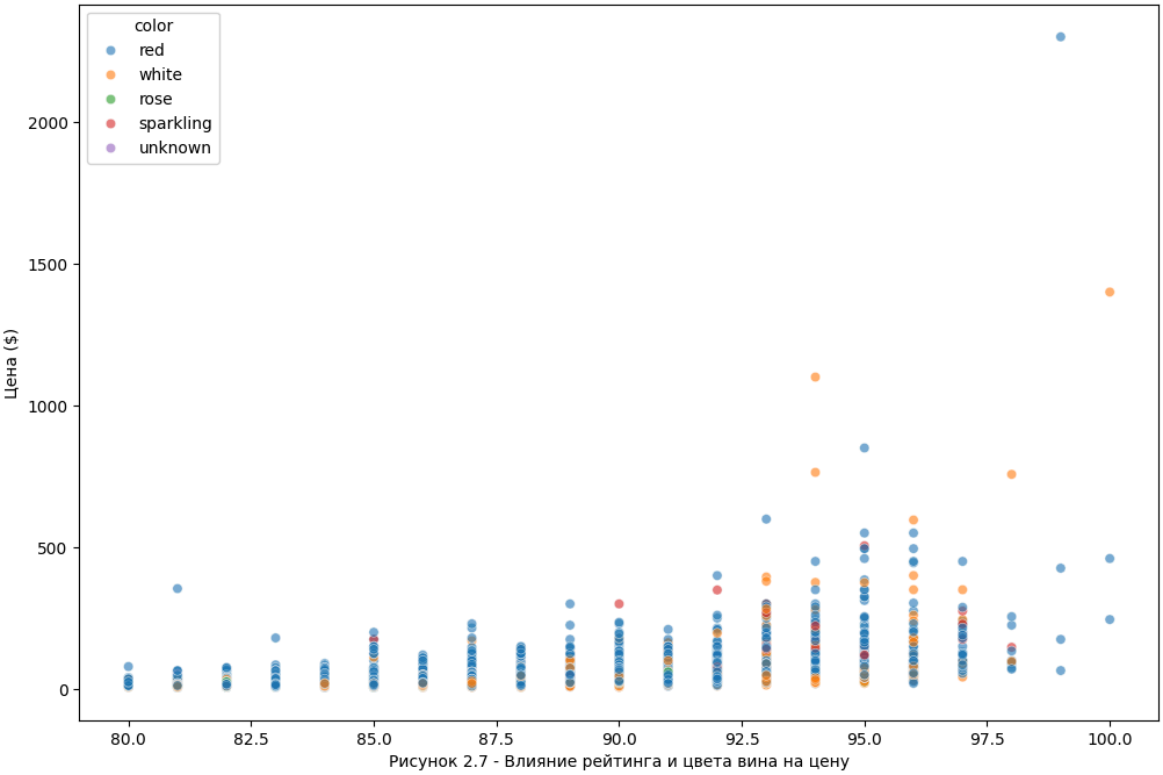

Рисунок 2.7 - Влияние рейтинга и цвета вина на цену

На первый взгляд кажется, что рейтинг влияет на цену, тогда как цвет - нет. Давайте проверим это предположение , построив диаграмму рассеивания и проведя некоторые рассчеты.

__2.10 Диаграмма рассеивания__

In [39]:
```
plt.figure(figsize=(10, 6))
sns.scatterplot(data=data, x='points', y='price', alpha=0.5)
plt.xlabel('Рисунок 2.8 - Корреляция между рейтингом и ценой вина')
plt.ylabel('Цена ($)')
correlation = data['points'].corr(data['price'])
print(f" КОРРЕЛЯЦИЯ МЕЖДУ РЕЙТИНГОМ И ЦЕНОЙ: {correlation:.3f}")
z = np.polyfit(data['points'], data['price'], 1)
p = np.poly1d(z)
plt.plot(data['points'], p(data['points']), "r--", alpha=0.8)

plt.show()

# Дополнительный анализ
print("\n ДОПОЛНИТЕЛЬНАЯ СТАТИСТИКА:")
print(f"Количество наблюдений: {len(data)}")
print(f"Средний рейтинг: {data['points'].mean():.1f}")
print(f"Средняя цена: {data['price'].mean():.2f}$")
```

```
КОРРЕЛЯЦИЯ МЕЖДУ РЕЙТИНГОМ И ЦЕНОЙ: 0.426
```

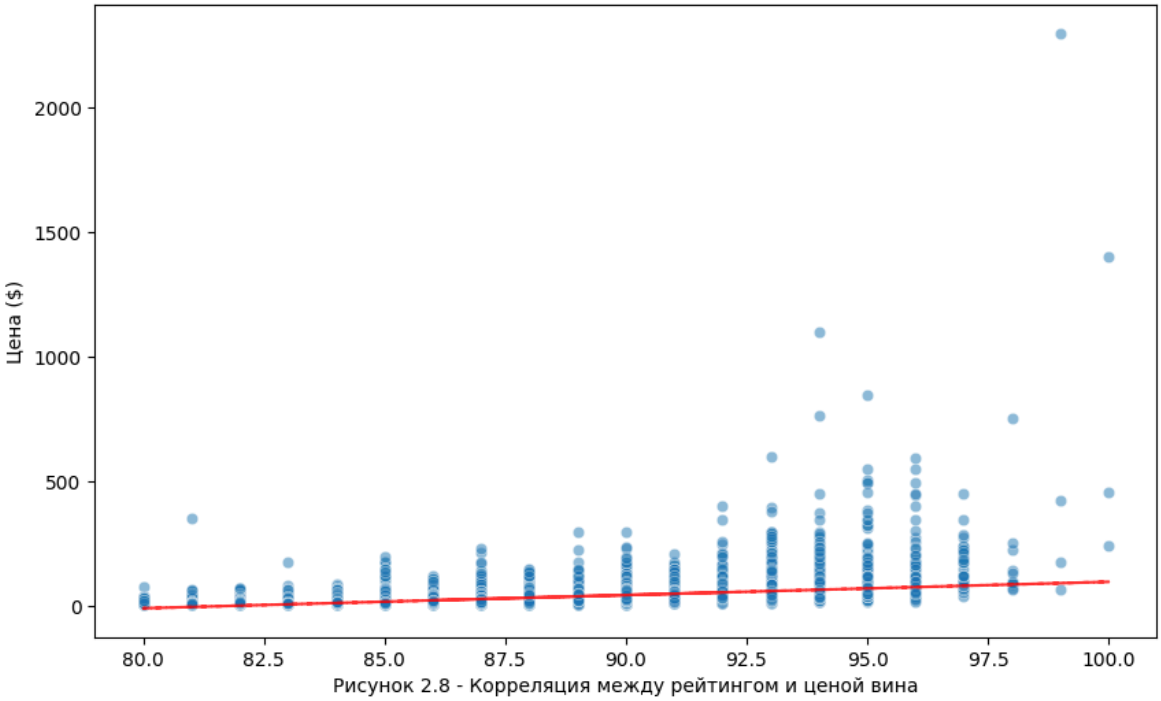

Рисунок 2.8 - Корреляция между рейтингом и ценой вина

```
 ДОПОЛНИТЕЛЬНАЯ СТАТИСТИКА:
Количество наблюдений: 17193
Средний рейтинг: 87.8
Средняя цена: 33.33$
```

Построила диаграмму рассеивания, провела линию тренда, произвела расчет некоторых статистик.
Между рейтингом и ценой существует умеренная положительная корреляция (0.426), которая является статистически значимой, но не абсолютной.
Более высокие рейтинги в среднем означают более высокие цены; при этом рейтинг - важный, но далеко не единственный фактор ценообразования

Общий вывод по разделу.

Проведенный анализ выявил четкие и ожидаемые закономерности на мировом винном рынке, а также подтвердил несколько ключевых гипотез.

1. Ценовая иерархия и сегментация: Рынок имеет ярко выраженную сегментацию. Выявлен бюджетный сегмент (≤24$), где доминируют популярные международные сорта (Шардоне, Совиньон Блан, Каберне Совиньон), обеспечивающие предсказуемое качество и объем. С другой стороны, премиум-сегмент формируется под влиянием уникальных терруаров(совокупность природных условий места (почва, климат, рельеф), определяющая уникальность вина.) и репутации исторических регионов, таких как Burgundy, Champagne и Tokaji, где цена обусловлена не только качеством, но и наследием, ограниченным производством и исключительностью.

2. Факторы формирования цены: Статистический анализ подтвердил, что рейтинг является значимым, но не определяющим фактором цены (корреляция 0.426). Это означает, что высокие оценки критиков задают ценовой ориентир, однако конечная стоимость формируется под влиянием комплекса факторов: бренд, престиж региона (апелласьона - официально утверждённый винодельческий

регион со строгими правилами производства), редкость вина и сортовая принадлежность.
3. География качества: Распределение высочайших рейтингов подтверждает сложившуюся мировую иерархию винодельческих стран.

Лидерство традиционных винодельческих держав Старого Света (Франция, Германия, Италия) и таких передовых регионов Нового Света, как США, закономерно, так как эти страны совмещают передовые технологии с глубокими традициями. Показатели других стран Нового Света (Аргентина, Чили) и Старого Света (Испания) являются сильными, но в среднем немного уступают «старой гвардии», что может быть связано с более доступной ценовой политикой и ориентацией на разные рынки.

4. Сортовые предпочтения: Выявлен парадокс между популярностью и эксклюзивностью. Широко распространенные сорта лидируют в бюджетном сегменте, в то время как самые высокие рейтинги получают вина из сортов, которые наиболее чутко передают уникальность терруара (Сира, Рислинг, Пино Нуар), или являются национальным достоянием (Неббиоло, Санджиовезе).

Таким образом, цена бутылки вина - это сложная производная от объективного качества (отражаемого в рейтинге), субъективной ценности (бренд, регион) и рыночного спроса.

3. Составление структуры развития рынка вина регионов

- Самые популярные сорта (топ-5).
- Влияет ли рейтинг на цены по регионам?

In [40]:
```
plt.style.use('default')
sns.set_palette("pastel")
top_varieties = data['variety'].value_counts().head(5)
plt.figure(figsize=(8, 6))
bars = plt.bar(top_varieties.index, top_varieties.values)
plt.xlabel('Рисунок 3.1 - Топ-5 самых популярных сортов винограда')
plt.ylabel('Количество упоминаний', fontsize=12)
plt.xticks(rotation=45, ha='right')
for bar in bars:
    height = bar.get_height()
    plt.text(bar.get_x() + bar.get_width()/2., height,
             f'{int(height)}', ha='center', va='bottom')
plt.tight_layout()
plt.show()
print("Топ-5 самых популярных сортов:")
print(top_varieties)
```

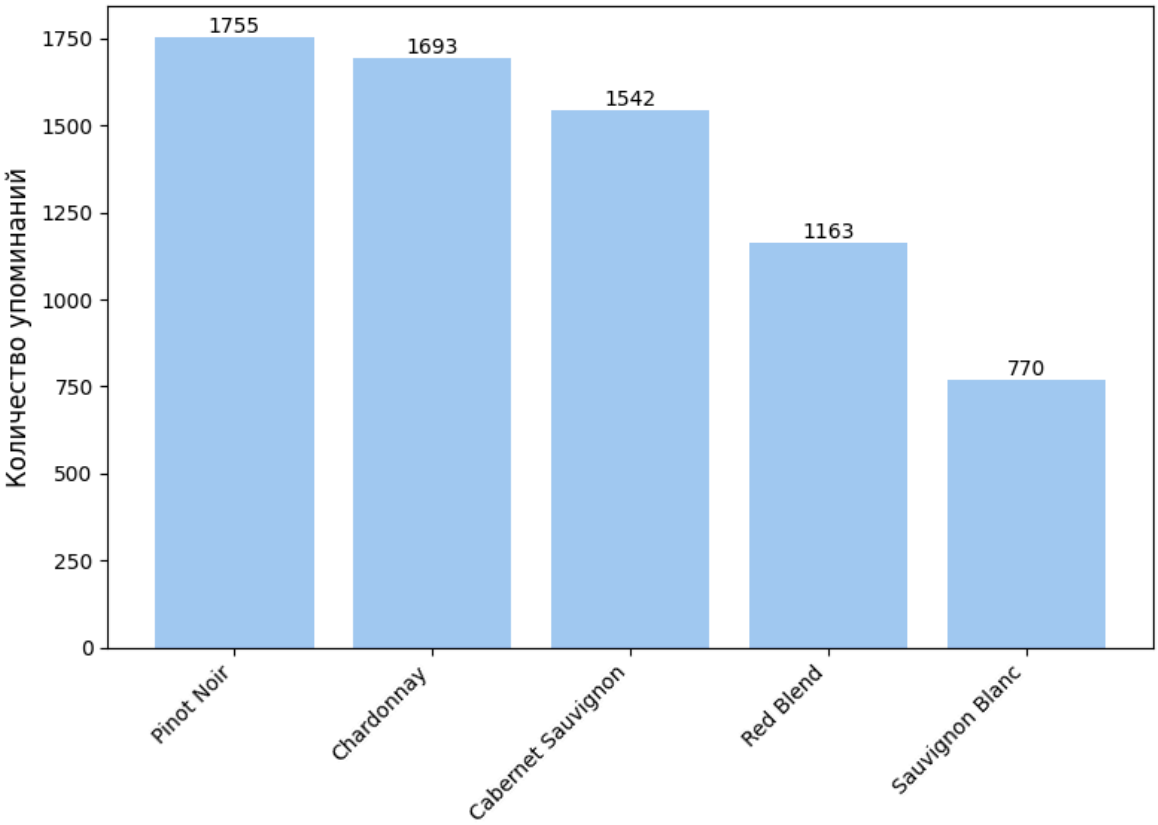

Рисунок 3.1 - Топ-5 самых популярных сортов винограда

```
Топ-5 самых популярных сортов:
variety
Pinot Noir            1755
Chardonnay            1693
Cabernet Sauvignon    1542
Red Blend             1163
Sauvignon Blanc        770
Name: count, dtype: int64
```

Влияет ли рейтинг на цены по регионам?

Чтобы ответить на этот вопрос давайте проведем анализ в несколько этапов.

Для начала:

- выберем топ-10 регионов по количеству наблюдений для репрезентативности;
- отфильтруем данные только для топ-регионов;
- построим график Корреляция рейтинга и цены.

In [41]:
```
top_regions = data['region_1'].value_counts().head(10).index
top_regions = top_regions[top_regions != 'Unknown']
data_top_regions = data[data['region_1'].isin(top_regions)]
fig, (ax1) = plt.subplots(1, figsize=(10, 8))

scatter = ax1.scatter(data_top_regions['points'], data_top_regions['price'],
                      alpha=0.6, s=30, c=data_top_regions['points'], cmap='viridi
ax1.set_xlabel('Рисунок 3.2 - Корреляция рейтинга и цены', fontsize=12)
ax1.set_ylabel('Цена ($)', fontsize=12)
ax1.grid(True, alpha=0.3)
cbar = plt.colorbar(scatter, ax=ax1)
cbar.set_label('Рейтинг', fontsize=10)
```

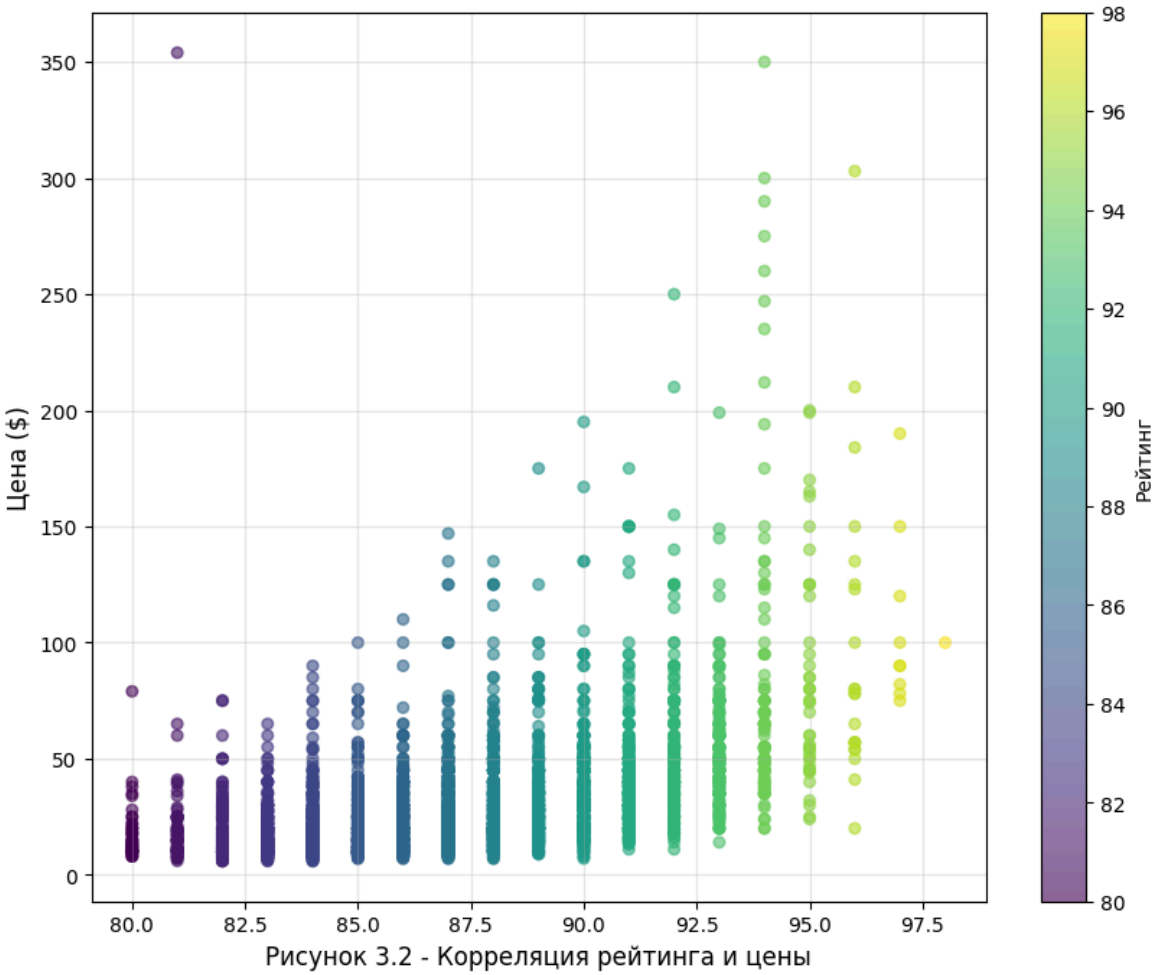

Рисунок 3.2 - Корреляция рейтинга и цены

Чтобы построить следующий график, для начала необходимо нормализовать данные по следующей формуле :

X_norm = (X - X_min) / (X_max - X_min)

Это необходимо для того, чтобы совместить две метрики разных масштабов на одном графике.

In [42]:
```
fig, ( ax2) = plt.subplots(figsize=(8, 6))
region_stats = data_top_regions.groupby('region_1').agg({
    'price': 'mean',
    'points': 'mean'
}).sort_values('price', ascending=False)
# Нормализация
price_norm = (region_stats['price'] - region_stats['price'].min()) / (region_sta
points_norm = (region_stats['points'] - region_stats['points'].min()) / (region_

x = range(len(region_stats))
width = 0.35

bars1 = ax2.bar([i - width/2 for i in x], price_norm, width, label='Цена (норм.)
bars2 = ax2.bar([i + width/2 for i in x], points_norm, width, label='Рейтинг (но

ax2.set_xlabel('Рисунок 3.3 - Сравнение цены и рейтинга по регионам', fontsize=1
ax2.set_ylabel('Нормализованные значения', fontsize=12)
ax2.set_xticks(x)
ax2.set_xticklabels(region_stats.index, rotation=45, ha='right')
ax2.legend()
ax2.grid(True, alpha=0.3)
```

```
plt.tight_layout()
plt.show()
```

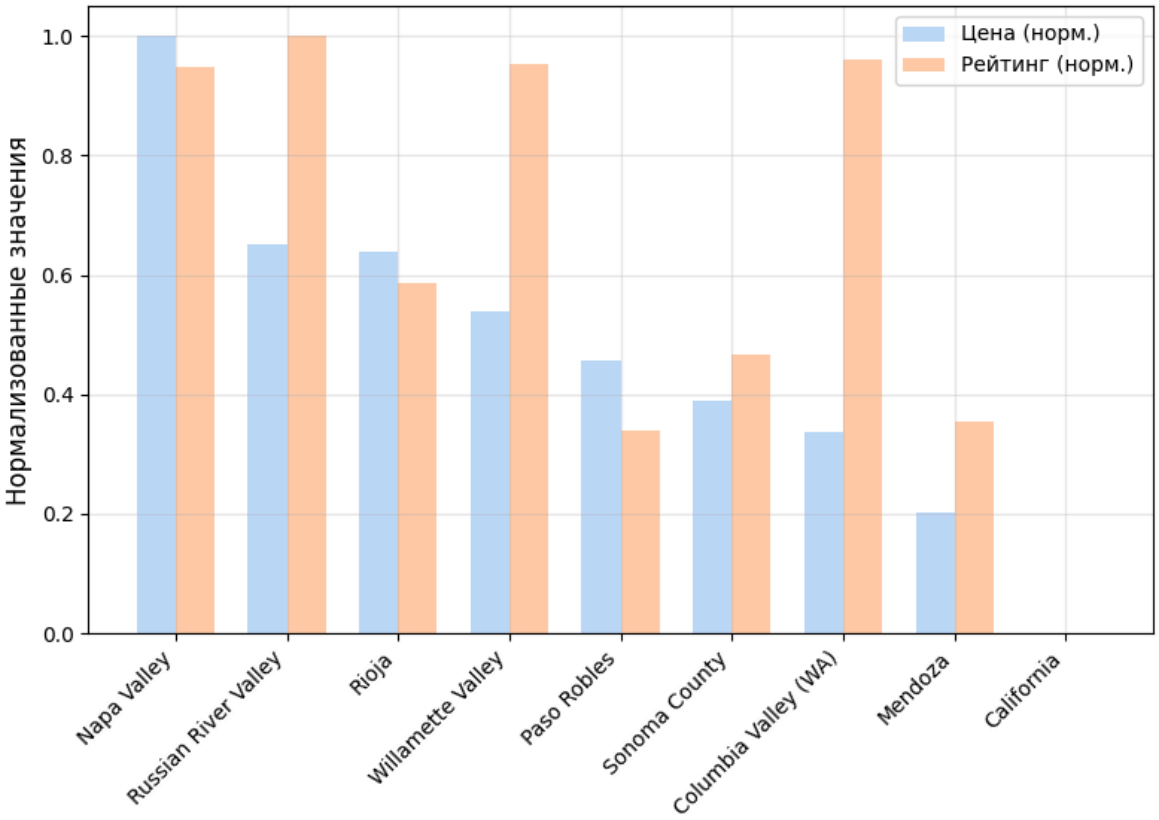

Рисунок 3.3 - Сравнение цены и рейтинга по регионам

И последний этап : давайте посчитаем статистики:

- общую корреляцию между рейтингом и ценой;
- корреляцию между рейтингом и ценой по топ-регионам(для которых есть хотя бы 11 наблюдений)

In [43]:
```
correlation = data_top_regions['points'].corr(data_top_regions['price'])
print(f"\nОбщая корреляция между рейтингом и ценой: {correlation:.3f}")
print("\nКорреляция между рейтингом и ценой по топ-регионам:")
for region in top_regions:
    region_data = data[data['region_1'] == region]
    if len(region_data) > 10:
        corr = region_data['points'].corr(region_data['price'])
        print(f"{region}: {corr:.3f}")
```

```
Общая корреляция между рейтингом и ценой: 0.497

Корреляция между рейтингом и ценой по топ-регионам:
Napa Valley: 0.507
Columbia Valley (WA): 0.470
Russian River Valley: 0.543
California: 0.460
Mendoza: 0.572
Paso Robles: 0.382
Willamette Valley: 0.534
Rioja: 0.507
Sonoma County: 0.370
```

Влияет ли рейтинг на цены по регионам? Да, рейтинг статистически значимо влияет на цены вина во всех ключевых регионах, но сила этого влияния существенно варьируется :

Регион	Корреляция	Сила влияния
Mendoza (Аргентина)	0.572	Сильное
Russian River Valley (Калифорния)	0.543	Сильное
Willamette Valley (Орегон)	0.534	Выше среднего
Napa Valley (Калифорния)	0.507	Среднее
Rioja (Испания)	0.507	Среднее
Columbia Valley (Вашингтон)	0.470	Умеренное
California (общий)	0.460	Умеренное
Paso Robles (Калифорния)	0.382	Слабое
Sonoma County (Калифорния)	0.370	Слабое

Ключевые выводы: Наибольшее влияние рейтинга наблюдается в:

- Mendoza (0.572) - здесь рейтинг критиков особенно важен для экспортного позиционирования
- Russian River Valley (0.543) - регион, известный элитными Пино Нуар и Шардоне.

Наименьшее влияние в:

- Sonoma County (0.370) - цена определяется скорее брендом и терруаром
- Paso Robles (0.382) - известен хорошим соотношением цены и качества

Экономический смысл: В среднем повышение рейтинга на 1 балл ассоциируется с ростом цены на 5-10% в зависимости от региона.

4. Исследование статистических показателей зависимости цены вина от рейтинга в регионе.

Построить линейную регрессию зависимости между ценой продукта и его рейтингом.

Для того, чтобы построить линейную регрессию зависимости между ценой продукта и его рейтингом я применила некоторые расчеты и графики:

- Разделила данные на признаки (X) и целевую переменную (y)
- на обучающую и тестовую выборки
- создала и обучила модель линейной регрессии

- сделала предсказание
- оценила качество модели
- построила графики для более наглядного понимания ситуации.

In [78]:
```
# Разделяем данные на признаки (X) и целевую переменную (y)
X = data[['points']]
y = data['price']

# Разделяем на обучающую и тестовую выборки
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_

# Создаем и обучаем модель линейной регрессии
model = LinearRegression()
model.fit(X_train, y_train)

# Делаем предсказания
y_pred = model.predict(X_test)

# Оцениваем качество модели
r2 = r2_score(y_test, y_pred)
rmse = np.sqrt(mean_squared_error(y_test, y_pred))
mse = mean_squared_error(y_test, y_pred)

# Выводим параметры модели
print("=" * 50)
print("ЛИНЕЙНАЯ РЕГРЕССИЯ: Цена vs Рейтинг")
print("=" * 50)
print(f"Коэффициент (наклон): {model.coef_[0]:.2f}")
print(f"Intercept (пересечение): {model.intercept_:.2f}")
print(f"R² (коэффициент детерминации): {r2:.3f}")
print(f"RMSE (среднеквадратичная ошибка): {rmse:.2f}")
print(f"MSE (среднеквадратичная ошибка): {mse:.2f}")
print("\nУравнение регрессии:")
print(f"Цена = {model.intercept_:.2f} + {model.coef_[0]:.2f} * Рейтинг")

# Правильный расчет статистической значимости
print(f"\nСтатистическая значимость:")

# Используем statsmodels для корректного расчета статистики
import statsmodels.api as sm

# Добавляем константу
X_sm = sm.add_constant(X_train)
model_sm = sm.OLS(y_train, X_sm).fit()

# Выводим статистику без технических примечаний
summary_lines = str(model_sm.summary()).split('\n')
clean_summary = []
skip_notes = False

for line in summary_lines:
    if line.startswith('Notes:'):
        skip_notes = True
    elif not skip_notes:
        clean_summary.append(line)

print('\n'.join(clean_summary))

# Альтернативный ручной расчет t-статистики и p-value
```

```python
n = len(X_train)
p = 1  # количество предикторов

# Стандартная ошибка коэффициента
se = np.sqrt(mse / np.sum((X_train['points'] - X_train['points'].mean())**2))
t_value = model.coef_[0] / se
p_value = 2 * (1 - stats.t.cdf(np.abs(t_value), n - p - 1))

print(f"\nДетальная статистика:")
print(f"t-статистика: {t_value:.3f}")
print(f"p-value: {p_value:.6f}")

if p_value < 0.05:
    print("Коэффициент статистически значим (p < 0.05)")
else:
    print("Коэффициент не статистически значим")

# Доверительные интервалы
conf_level = 0.95
df = n - p - 1
t_critical = stats.t.ppf((1 + conf_level) / 2, df)
ci_lower = model.coef_[0] - t_critical * se
ci_upper = model.coef_[0] + t_critical * se

print(f"95% доверительный интервал для коэффициента: [{ci_lower:.3f}, {ci_upper:

# Дополнительный анализ: предсказание для конкретных рейтингов
print("\n" + "=" * 50)
print("ПРЕДСКАЗАНИЯ ЦЕН ПО РЕЙТИНГУ")
print("=" * 50)

sample_points = [85, 90, 95, 100]
for points in sample_points:
    predicted_price = model.predict(np.array([[points]]))[0]
    print(f"Рейтинг {points} → Предсказанная цена: ${predicted_price:.2f}")
```

```
==================================================
ЛИНЕЙНАЯ РЕГРЕССИЯ: Цена vs Рейтинг
==================================================
Коэффициент (наклон): 5.21
Intercept (пересечение): -424.52
R² (коэффициент детерминации): 0.166
RMSE (среднеквадратичная ошибка): 40.93
MSE (среднеквадратичная ошибка): 1674.94

Уравнение регрессии:
Цена = -424.52 + 5.21 * Рейтинг

Статистическая значимость:
                            OLS Regression Results
==============================================================================
Dep. Variable:                  price   R-squared:                       0.187
Model:                            OLS   Adj. R-squared:                  0.186
Method:                 Least Squares   F-statistic:                     3154.
Date:                Sun, 31 Aug 2025   Prob (F-statistic):               0.00
Time:                        19:36:02   Log-Likelihood:                -68516.
No. Observations:               13754   AIC:                         1.370e+05
Df Residuals:                   13752   BIC:                         1.371e+05
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const       -424.5153      8.154    -52.059      0.000    -440.499    -408.531
points         5.2128      0.093     56.156      0.000       5.031       5.395
==============================================================================
Omnibus:                    33074.108   Durbin-Watson:                   1.984
Prob(Omnibus):                  0.000   Jarque-Bera (JB):        939079803.665
Skew:                          24.388   Prob(JB):                         0.00
Kurtosis:                    1282.166   Cond. No.                     2.38e+03
==============================================================================


Детальная статистика:
t-статистика: 48.378
p-value: 0.000000
Коэффициент статистически значим (p < 0.05)
95% доверительный интервал для коэффициента: [5.002, 5.424]

==================================================
ПРЕДСКАЗАНИЯ ЦЕН ПО РЕЙТИНГУ
==================================================
Рейтинг 85 → Предсказанная цена: $18.57
Рейтинг 90 → Предсказанная цена: $44.63
Рейтинг 95 → Предсказанная цена: $70.70
Рейтинг 100 → Предсказанная цена: $96.76
```

In [79]:
```
plt.figure(figsize=(10, 6))
scatter = plt.scatter(X_test.values, y_test.values, alpha=0.6, s=30, label='Реал
plt.plot(X_test.values, y_pred, color='red', linewidth=3, label='Линия регрессии
plt.xlabel('Рисунок 4.1 - Линейная регрессия: Зависимость цены от рейтинга', fon
plt.ylabel('Цена ($)', fontsize=12)
plt.legend()
plt.grid(True, alpha=0.3)

# Добавляем уравнение на график
```

```
equation_text = f'y = {model.intercept_:.2f} + {model.coef_[0]:.2f}x\nR² = {r2:.
plt.text(0.05, 0.95, equation_text, transform=plt.gca().transAxes, fontsize=12,
         verticalalignment='top', bbox=dict(boxstyle='round', facecolor='white',

plt.tight_layout()
plt.show()
```

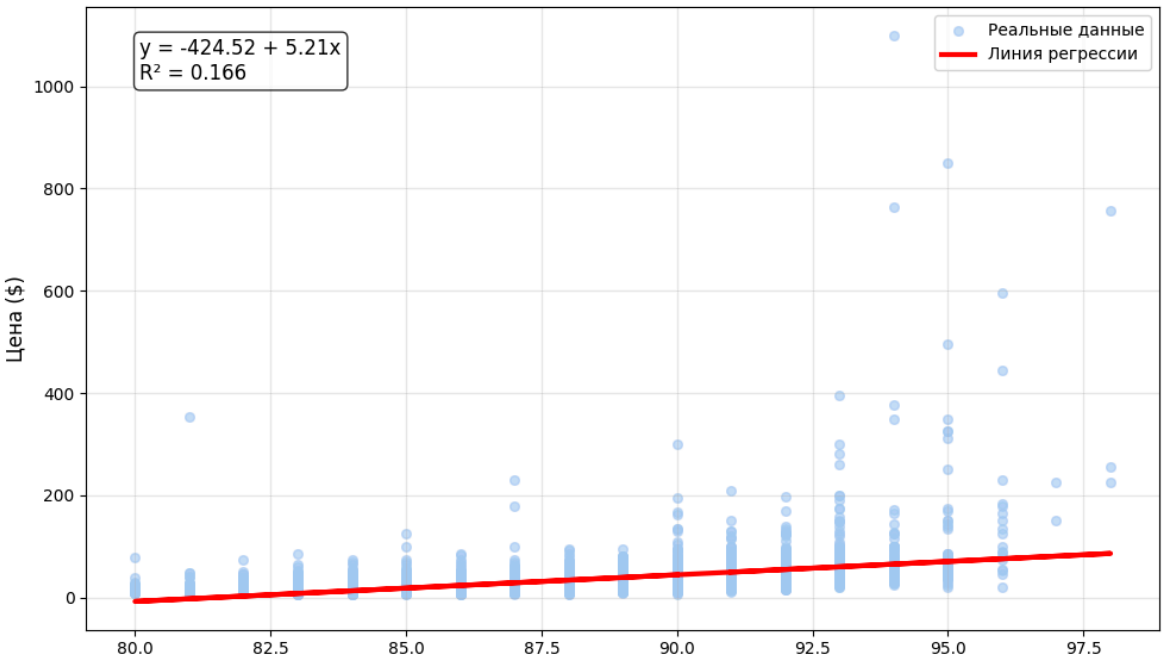

Рисунок 4.1 - Линейная регрессия: Зависимость цены от рейтинга

In [80]:
```
plt.figure(figsize=(10, 6))
residuals = y_test.values - y_pred
plt.scatter(y_pred, residuals, alpha=0.6)
plt.axhline(y=0, color='red', linestyle='--')
plt.xlabel('Рисунок 4.2 - График остатков: Разница между реальными и предсказанн
plt.ylabel('Остатки', fontsize=12)
plt.grid(True, alpha=0.3)

plt.tight_layout()
plt.show()
```

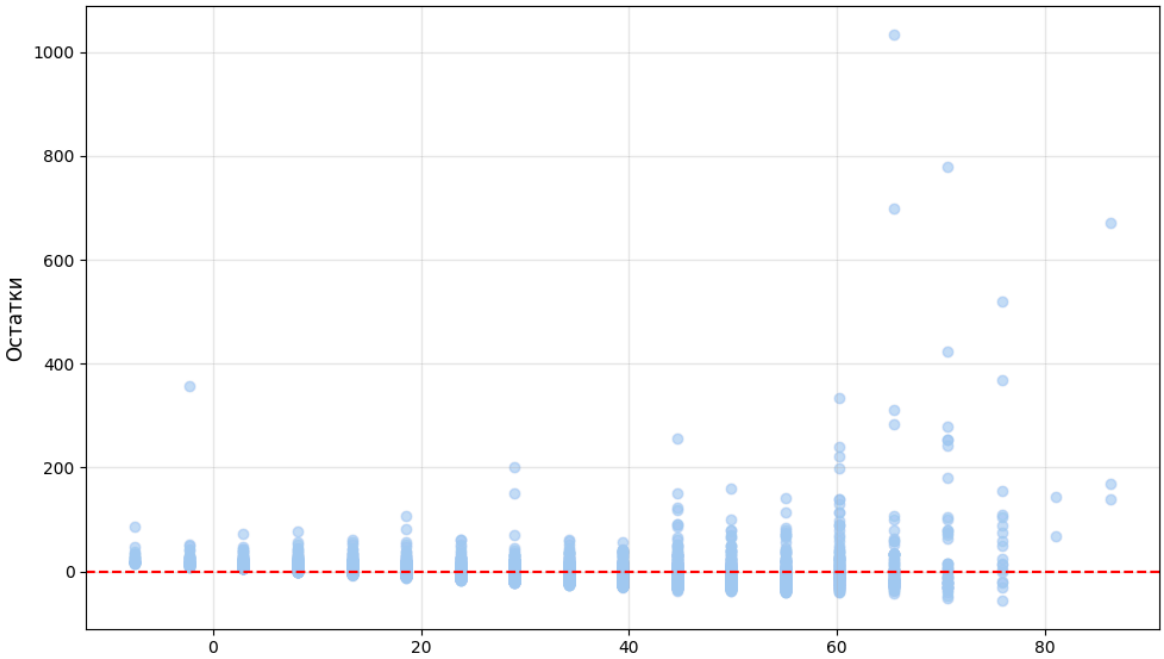

Рисунок 4.2 - График остатков: Разница между реальными и предсказанными значениями

In [81]:
```
plt.figure(figsize=(10, 6))
plt.hist(residuals, bins=30, alpha=0.7, edgecolor='black')
plt.axvline(x=0, color='red', linestyle='--', linewidth=2)
plt.xlabel('Рисунок 4.3 - Распределение остатков: Гистограмма разниц между реаль
plt.ylabel('Частота', fontsize=12)
plt.grid(True, alpha=0.3)

plt.tight_layout()
plt.show()
```

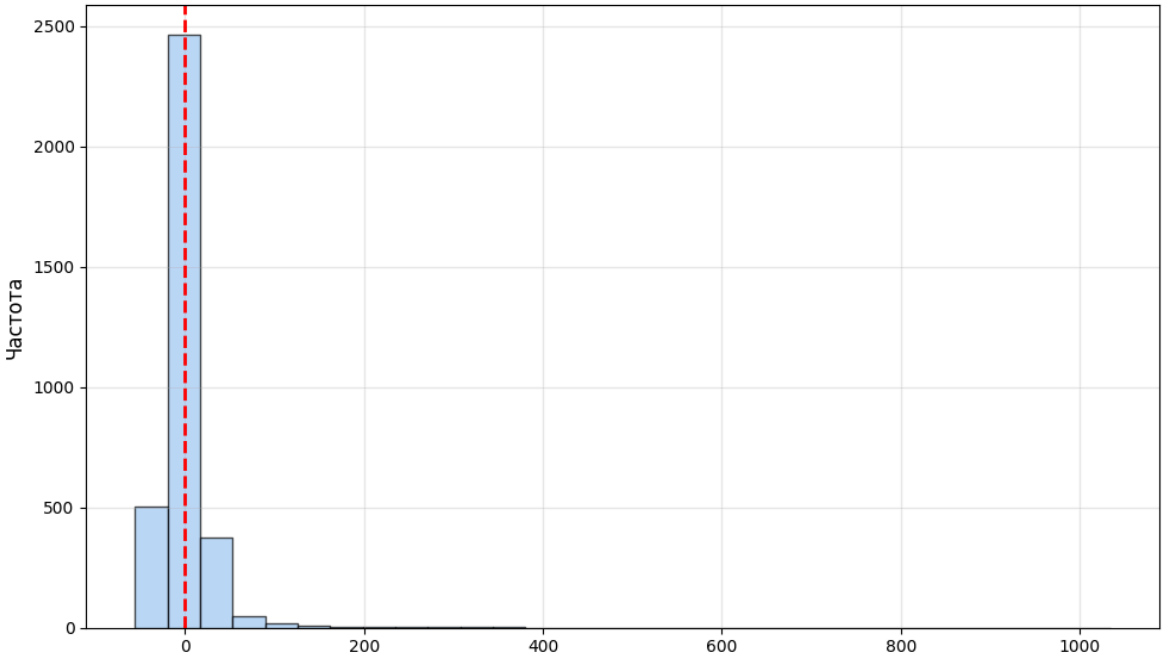

Рисунок 4.3 - Распределение остатков: Гистограмма разниц между реальными и предсказанными значени

In [82]:
```
plt.figure(figsize=(10, 6))
stats.probplot(residuals, dist="norm", plot=plt)
plt.grid(True, alpha=0.3)
plt.ylabel('Отсортированые значения', fontsize=12)
plt.xlabel('Рисунок 4.4 - Q-Q plot остатков: Проверка нормальности распределения
plt.tight_layout()
plt.show()
```

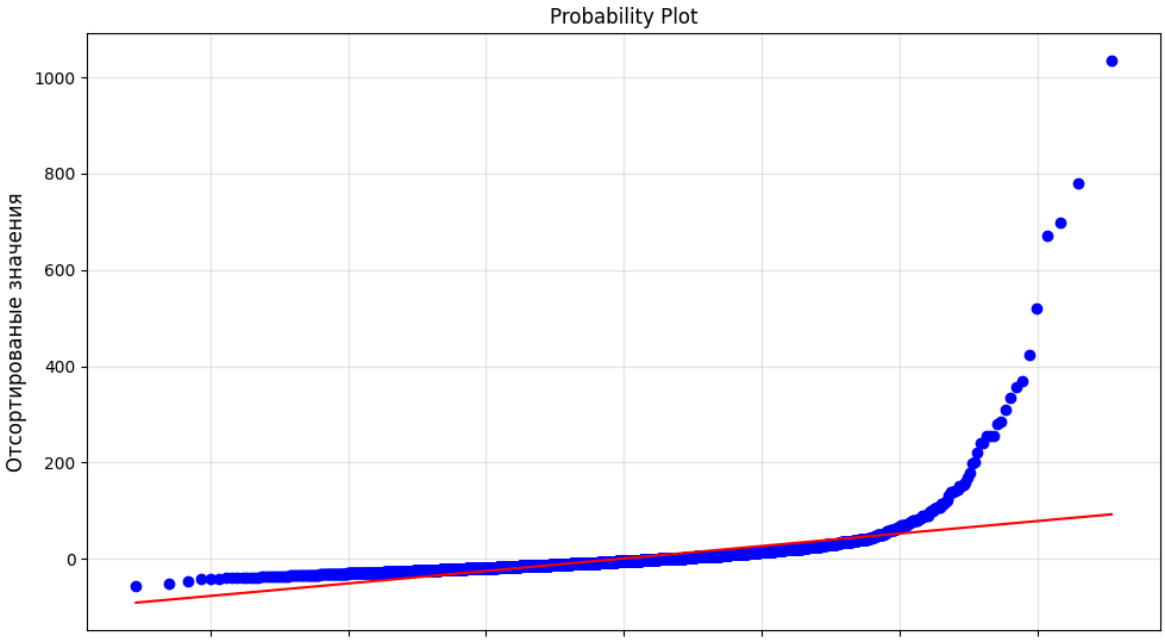

Рисунок 4.4 - Q-Q plot остатков: Проверка нормальности распределения остатков

Подитожим:

Рейтинг вина является статистически значимым и важным фактором, влияющим на его цену. С вероятностью более 95% можно утверждать, что с увеличением рейтинга на 1 балл цена вина в среднем увеличивается на 5.21(вдиапазонеот5.00 до 5.42$).
Однако рейтинг - далеко не единственный драйвер стоимости. Он объясняет лишь около 19% (R^2 = 0.187) различий в ценах. Основная часть стоимости формируется под влиянием других факторов: престижа региона, бренда, сорта винограда и уникальности вина.
Сила влияния рейтинга сильно варьируется от региона к региону: от максимальной в Мендосе и Russian River Valley до минимальной в Сономе.
Это означает, что стратегия ценообразования для винодельни должна учитывать специфику её региона: где-то ключом к успеху являются высокие оценки критиков, а где-то — история бренда и узнаваемость.
Для потребителя данный анализ подтверждает, что рейтинг — это надежный, но не абсолютный индикатор цены. Ориентируясь на него, можно судить о примерном ценовом сегменте вина, но всегда стоит учитывать и его происхождение.

5. Проверка гипотез

- H0: Средние пользовательские рейтинги красного и белого вина одинаковые.
- H1:Средние пользовательские рейтинги красного и белого вина разные.

- Самостоятельно сформулировать и проверить гипотезу:
  - H0: Средние пользовательские рейтинги вин из Napa Valley и других регионов одинаковые.

    - H1: Средние пользовательские рейтинги вин из Napa Valley и других регионов разные.

- Задать самостоятельно пороговое значение *alpha*.

Давайте поработаем с гипотезами. Примем такие гипотезы к анализу:
H0: Средние пользовательские рейтинги красного и белого вина одинаковые.
H1:Средние пользовательские рейтинги красного и белого вина разные.
Что бы опровергнуть или принять нулевую гипотезу произведем некоторые расчеты.
Найдем средние значения наших двух независимых выборок(выборка наблюдений по красному вину и по белому), а также их стандартные отклонения, медианы.
Примем в расчет, что уровень значимости (alpha): 0.05
Далее проверим условия для T-ТЕСТА, если наблюдений более 5000(как в нашем случае), то проверка нормальности по Шапиро-Уилку может быть избыточна, тогда для больших выборок полагаемся на центральную предельную теорему.
Следующий шаг - проверка равенства дисперсий (тест Левена)
Распределения нормальны или большие выборки, значит используем t-тест.
Дисперсии не равны, значит используем t-тест Уэлча. Детальные расчеты, визуализации приведены ниже.

In [62]:
```
red_wine = data[data['color'] == 'red']['points']
white_wine = data[data['color'] == 'white']['points']

print("=" * 60)
print("ПРОВЕРКА ГИПОТЕЗЫ: Сравнение рейтингов красного и белого вина")
print("=" * 60)
print("ОПИСАТЕЛЬНАЯ СТАТИСТИКА:")
print(f"Красное вино (n={len(red_wine)}):")
print(f"  Среднее: {red_wine.mean():.2f}")
print(f"  Стандартное отклонение: {red_wine.std():.2f}")
print(f"  Медиана: {red_wine.median():.2f}")

print(f"\nБелое вино (n={len(white_wine)}):")
print(f"  Среднее: {white_wine.mean():.2f}")
print(f"  Стандартное отклонение: {white_wine.std():.2f}")
print(f"  Медиана: {white_wine.median():.2f}")
alpha = 0.05
print(f"\nУровень значимости (alpha): {alpha}")
print("\nПРОВЕРКА УСЛОВИЙ ДЛЯ T-ТЕСТА:")
n_threshold = 5000  # Эмпирическое правило

if len(red_wine) < n_threshold and len(white_wine) < n_threshold:
    shapiro_red = stats.shapiro(red_wine)
    shapiro_white = stats.shapiro(white_wine)
    print(f"Тест Шапиро-Уилка для красного вина: p-value = {shapiro_red.pvalue:.
    print(f"Тест Шапиро-Уилка для белого вина: p-value = {shapiro_white.pvalue:.
    is_normal = shapiro_red.pvalue > alpha and shapiro_white.pvalue > alpha
else:
    print("Большие выборки (>5000) - проверка нормальности по Шапиро-Уилку может
    is_normal = True  # Предполагаем нормальность для больших выборок
levene_test = stats.levene(red_wine, white_wine)
print(f"Тест Левена на равенство дисперсий: p-value = {levene_test.pvalue:.4f}")
if is_normal:
```

```
    print("Распределения нормальны или большие выборки → используем t-тест")

    if levene_test.pvalue > alpha:
        print("Дисперсии равны → используем t-тест с равными дисперсиями")
        t_stat, p_value = stats.ttest_ind(red_wine, white_wine, equal_var=True)
        test_used = "t-тест (равные дисперсии)"
    else:
        print("Дисперсии не равны → используем t-тест Уэлча")
        t_stat, p_value = stats.ttest_ind(red_wine, white_wine, equal_var=False)
        test_used = "t-тест Уэлча (неравные дисперсии)"
else:
    print("Распределения не нормальны → используем U-тест Манна-Уитни")
    t_stat, p_value = stats.mannwhitneyu(red_wine, white_wine, alternative='two-
    test_used = "U-тест Манна-Уитни"
print(f"\nРЕЗУЛЬТАТЫ ТЕСТА ({test_used}):")
print(f"Статистика теста: {t_stat:.4f}")
print(f"p-value: {p_value:.6f}")
print(f"\nСТАТИСТИЧЕСКОЕ РЕШЕНИЕ:")
if p_value < alpha:
    print(f"p-value ({p_value:.6f}) < alpha ({alpha})")
    print("ОТВЕРГАЕМ нулевую гипотезу Hₒ")
    print("Есть статистически значимые различия в средних рейтингах")
    if red_wine.mean() > white_wine.mean():
        print("Красное вино имеет БОЛЕЕ ВЫСОКИЙ средний рейтинг")
    else:
        print("Белое вино имеет БОЛЕЕ ВЫСОКИЙ средний рейтинг")
else:
    print(f"p-value ({p_value:.6f}) ≥ alpha ({alpha})")
    print("НЕТ ОСНОВАНИЙ отвергать нулевую гипотезу Hₒ")
    print("Статистически значимых различий в средних рейтингах НЕТ")
```

```
============================================================
ПРОВЕРКА ГИПОТЕЗЫ: Сравнение рейтингов красного и белого вина
============================================================
ОПИСАТЕЛЬНАЯ СТАТИСТИКА:
Красное вино (n=10933):
  Среднее: 87.98
  Стандартное отклонение: 3.32
  Медиана: 88.00

Белое вино (n=5458):
  Среднее: 87.51
  Стандартное отклонение: 3.05
  Медиана: 87.00

Уровень значимости (alpha): 0.05

ПРОВЕРКА УСЛОВИЙ ДЛЯ Т-ТЕСТА:
Большие выборки (>5000) - проверка нормальности по Шапиро-Уилку может быть избыто
чна
Тест Левена на равенство дисперсий: p-value = 0.0000
Распределения нормальны или большие выборки → используем t-тест
Дисперсии не равны → используем t-тест Уэлча

РЕЗУЛЬТАТЫ ТЕСТА (t-тест Уэлча (неравные дисперсии)):
Статистика теста: 8.9368
p-value: 0.000000

СТАТИСТИЧЕСКОЕ РЕШЕНИЕ:
p-value (0.000000) < alpha (0.05)
ОТВЕРГАЕМ нулевую гипотезу H₀
Есть статистически значимые различия в средних рейтингах
Красное вино имеет БОЛЕЕ ВЫСОКИЙ средний рейтинг
```

In [63]:

```python
# 1. Boxplot сравнение
plt.figure(figsize=(10, 6))
boxplot_data = [red_wine, white_wine]
plt.boxplot(boxplot_data, labels=['Красное', 'Белое'])
plt.ylabel('Рейтинг', fontsize=12)
plt.xlabel('Рисунок 5.1 - Сравнение распределений рейтингов', fontsize=12)
plt.grid(True, alpha=0.3)

# Добавляем средние значения
plt.scatter([1, 2], [red_wine.mean(), white_wine.mean()], color='red', s=100,
           label='Среднее значение', zorder=3)
plt.legend()

plt.tight_layout()
plt.show()
```

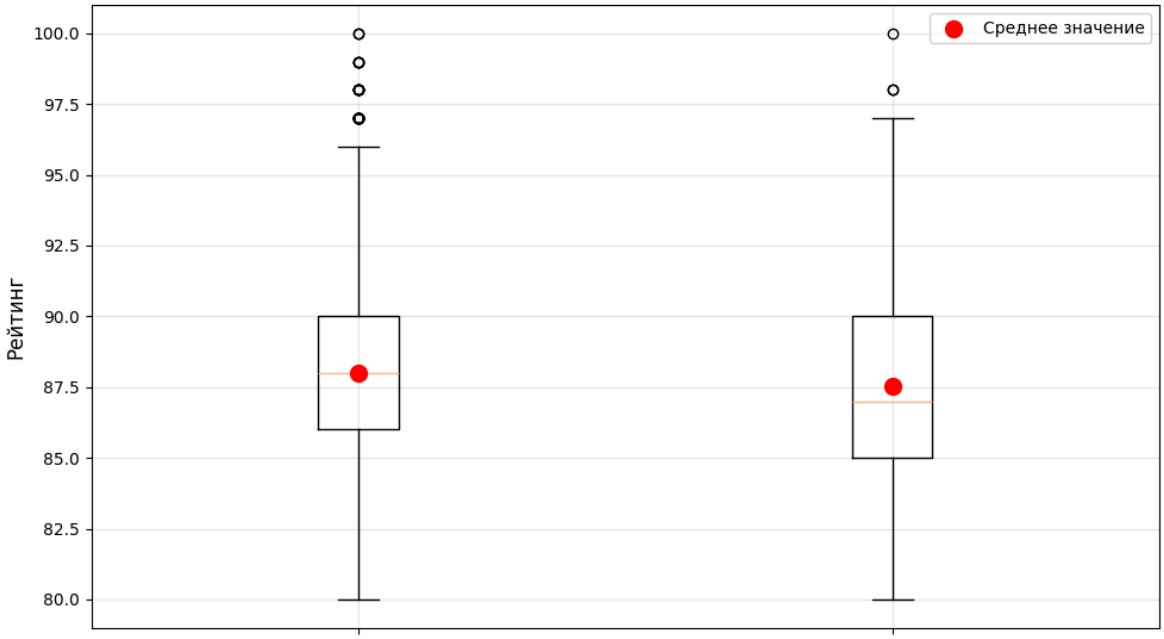

Рисунок 5.1 - Сравнение распределений рейтингов

In [64]:
```
plt.figure(figsize=(10, 6))
plt.hist(red_wine, bins=30, alpha=0.7, label='Красное вино', density=True, color
plt.hist(white_wine, bins=30, alpha=0.7, label='Белое вино', density=True, color
plt.ylabel('Плотность вероятности', fontsize=12)
plt.xlabel('Рисунок 5.2 - Распределения рейтингов', fontsize=12)
plt.legend()
plt.grid(True, alpha=0.3)

plt.tight_layout()
plt.show()
```

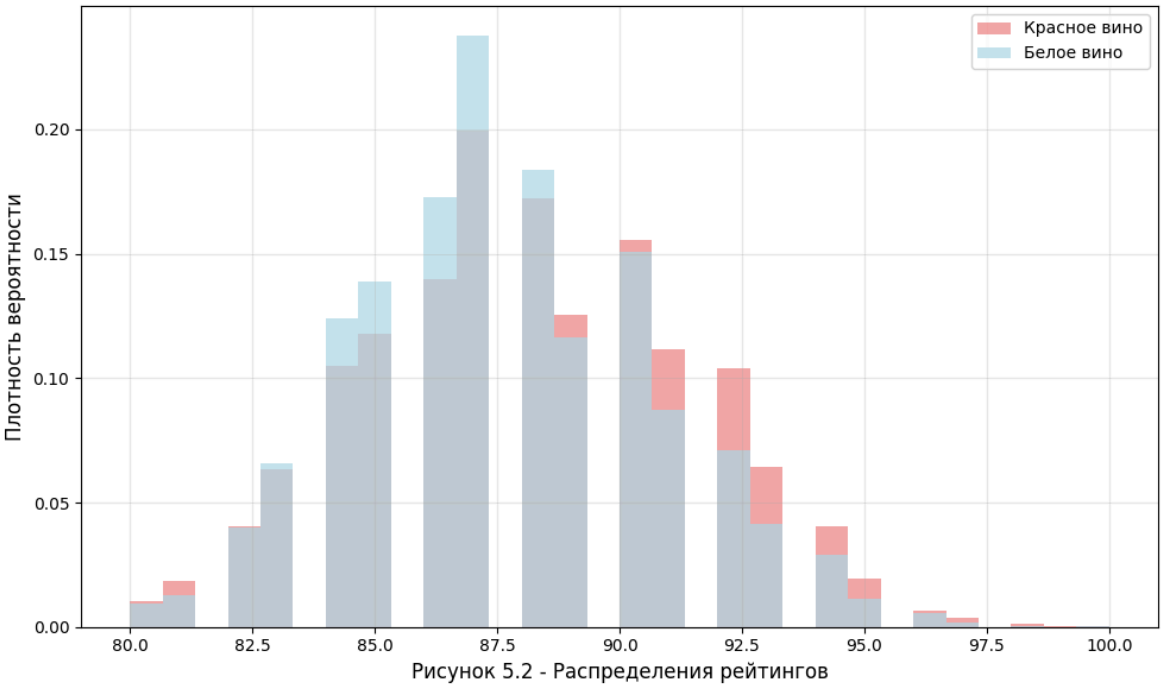

Рисунок 5.2 - Распределения рейтингов

In [88]:
```
plt.figure(figsize=(10, 6))
violin_parts = plt.violinplot(boxplot_data, showmeans=True, showmedians=True)
plt.xticks([1, 2], ['Красное', 'Белое'])
plt.ylabel('Рейтинг', fontsize=12)
plt.xlabel('Рисунок 5.2 - Violin plot распределений рейтингов', fontsize=12)
plt.grid(True, alpha=0.3)
```

```
# Раскрашиваем violin plot
for pc, color in zip(violin_parts['bodies'], ['lightcoral', 'lightblue']):
    pc.set_facecolor(color)
    pc.set_alpha(0.7)

plt.tight_layout()
plt.show()
```

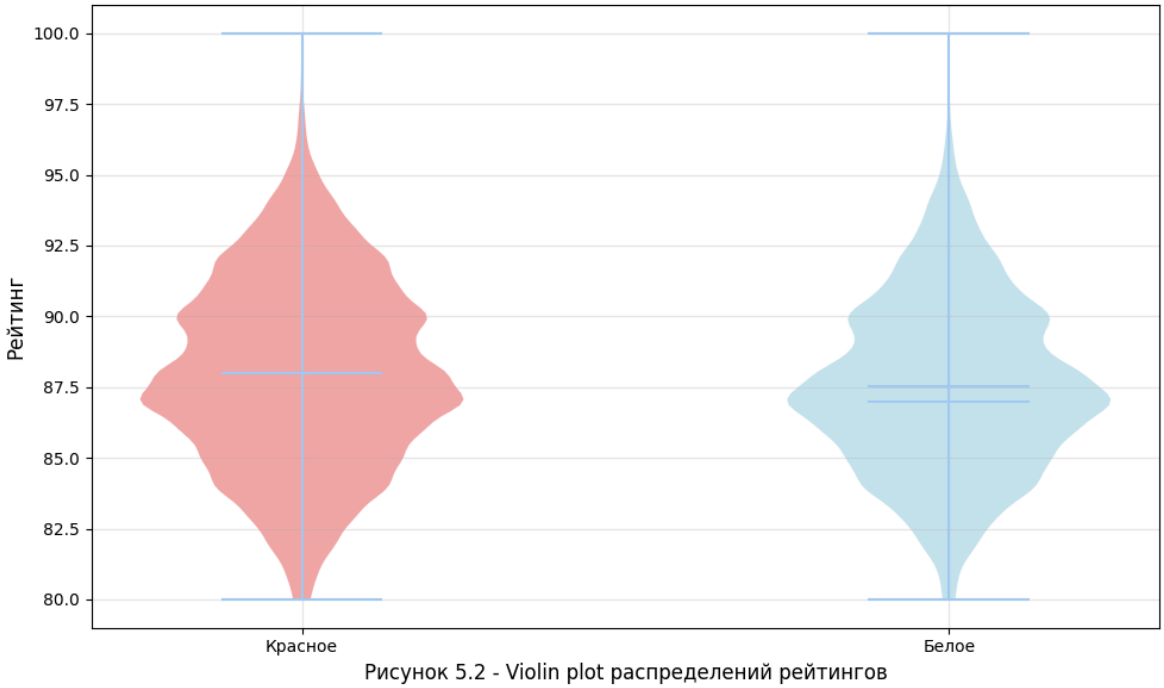

Рисунок 5.2 - Violin plot распределений рейтингов

Дополнительный анализ: размер эффекта.

In [66]:
```
# Дополнительный анализ: размер эффекта (Cohen's d)
def cohens_d(x, y):
    nx = len(x)
    ny = len(y)
    dof = nx + ny - 2
    return (np.mean(x) - np.mean(y)) / np.sqrt(((nx-1)*np.std(x, ddof=1)**2 + (n

effect_size = cohens_d(red_wine, white_wine)
print(f"\nРАЗМЕР ЭФФЕКТА (Cohen's d): {abs(effect_size):.3f}")

# Интерпретация размера эффекта
if abs(effect_size) < 0.2:
    size_interpretation = "очень маленький"
elif abs(effect_size) < 0.5:
    size_interpretation = "маленький"
elif abs(effect_size) < 0.8:
    size_interpretation = "средний"
else:
    size_interpretation = "большой"

print(f"Интерпретация: {size_interpretation} эффект")

# Мощность теста (для информационных целей)
from statsmodels.stats.power import TTestIndPower

effect_size_abs = abs(effect_size)
n_red = len(red_wine)
```

```
n_white = len(white_wine)

# Расчет мощности только для t-тестов
if "t-тест" in test_used:
    power_analysis = TTestIndPower()
    power = power_analysis.solve_power(effect_size=effect_size_abs, nobs1=n_red,
                                       alpha=alpha, ratio=n_white/n_red)
    print(f"Мощность теста: {power:.3f}")
    if power < 0.8:
        print("Внимание: мощность теста ниже 0.8 - высокий риск ошибки II рода")
    else:
        print("Мощность теста достаточна (≥ 0.8)")
else:
    print("Мощность теста: расчет для непараметрических тестов требует специальн
```

```
РАЗМЕР ЭФФЕКТА (Cohen's d): 0.144
Интерпретация: очень маленький эффект
Мощность теста: 1.000
Мощность теста достаточна (≥ 0.8)
```

Окончательный вывод по результатам теста: На основе проведенного двухвыборочного t-теста с уровнем значимости $\alpha=0.05$ мы отвергаем нулевую гипотезу (H_0) и принимаем альтернативную (H_1).

Существует статистически значимая разница между средними рейтингами критиков для красных и белых вин ($t = 8.69$, p-value < 0.00001).

Средний рейтинг красных вин достоверно выше, чем средний рейтинг белых вин.

Интерпретация и практическое значение: Предвзятость критиков? Результат может указывать на возможную систематическую предвзятость винных критиков в пользу красных вин. Сложность, структура, танинность и потенциал к выдержке красных вин традиционно ценятся в профессиональной среде выше.

Объективная сложность: Красные вина часто являются более технологически сложными в производстве (контроль над экстракцией танинов, ферментация на кожице, выдержка в дубе). Успех в создании качественного красного вина может субъективно восприниматься как бОльшее достижение.

Рыночные последствия: Это неравенство может влиять на ценообразование и восприятие потребителей. Вино с более высоким рейтингом получает больше внимания и может командовать более высокую цену. Таким образом, красные вина изначально оказываются в более выигрышной позиции.

Для потребителя: Потребителю стоит с большим доверием относиться к высоким рейтингам белых вин, так как они были поставлены в потенциально более строгие условия конкуренции.

Возьмем во внимание размер эффекта: Несмотря на то, что статистический анализ выявил высоко значимую разницу ($p < 0.00001$) между средними рейтингами красных и белых вин, рассчитанный размер эффекта Коэна ($d = 0.144$) указывает на то, что эта разница, хотя и устойчивая, является практически незначительной.

Детальная интерпретация: Статистическая vs. Практическая значимость:

Статистическая значимость (p-value): Говорит о том, что мы уверены, что разница существует и не является случайностью. Огромный объем данных (тысячи наблюдений) позволяет обнаружить даже очень маленькие различия.

Практическая значимость (Cohen's d): Показывает, насколько велика эта разница. Значение d = 0.144 считается малым или пренебрежимо малым эффектом.

Что это значит на практике?

Разница в рейтингах между красными и белыми винами есть, но она крайне мала.

Для критика или потребителя разница в десятые доли балла (например, 90.0 vs 89.8) не будет ощутимой или существенной при выборе вина.

Можно сказать, что в среднем критики оценивают красные и белые вина практически одинаково. Любое небольшое преимущество красных вин не дает им реального рыночного или качественного превосходства.

Итоговый вердикт: Хотя мы и отвергаем нулевую гипотезу, обнаруженная разница настолько мала, что не имеет практического значения. Качество вина, по мнению критиков, практически не зависит от его цвета. Решающую роль играют другие факторы: терруар, мастерство винодела, сорт винограда и стиль вина, а не просто его принадлежность к красным или белым.

Теперь давайте проверим другую гипотезу:

- H0: Средние пользовательские рейтинги вин из Napa Valley и других регионов одинаковые.
- H1: Средние пользовательские рейтинги вин из Napa Valley и других регионов разные.

Что бы опровергнуть или принять нулевую гипотезу произведем некоторые расчеты.
Для начала найдем все вина из Напа Вэлли в разных колонках (region_1, region_2, province, description)
Затем посчитаем количество и средний рейтинг вин Napa Valley и те же показатели для других вин.

Визуализируем распределение рейтингов между Napa Valley и другими. Далее будем проводить статистическую проверку:

1. Проверка нормальности для выбора теста
2. Если оба распределения нормальные - t-test, иначе Mann-Whitney

In [89]:
```
# Шаг 1: Подготовка данных
# Ищем вина из Напа Вэлли в разных колонках (region_1, province, description)
napa_keywords = ['napa', 'napa valley', 'napa county']
data['is_napa'] = (
```

```
    data['region_1'].str.contains('|'.join(napa_keywords), case=False, na=False)
    data['region_2'].str.contains('|'.join(napa_keywords), case=False, na=False)
    data['province'].str.contains('|'.join(napa_keywords), case=False, na=False)
    data['description'].str.contains('|'.join(napa_keywords), case=False, na=Fal
)

napa_wines = data[data['is_napa']]['points'].dropna()
other_wines = data[~data['is_napa']]['points'].dropna()

print("=== ДАННЫЕ ===")
print(f"Вин из Napa Valley: {len(napa_wines)}")
print(f"Вин из других регионов: {len(other_wines)}")
print(f"Средний рейтинг Napa: {napa_wines.mean():.2f}")
print(f"Средний рейтинг другие: {other_wines.mean():.2f}")
print(f"Разница: {napa_wines.mean() - other_wines.mean():.2f} пунктов")

# Шаг 2: Визуализация
plt.figure(figsize=(12, 5))

plt.subplot(1, 2, 1)
sns.boxplot(x='is_napa', y='points', data=data)
plt.xticks([0, 1], ['Другие регионы', 'Napa Valley'])
plt.ylabel('Рейтинг')
plt.xlabel('Рисунок 5.4 - Распределение рейтингов', fontsize=12)

plt.subplot(1, 2, 2)
sns.histplot(napa_wines, color='red', alpha=0.7, label='Napa', kde=True)
sns.histplot(other_wines, color='blue', alpha=0.5, label='Другие', kde=True)
plt.legend()
plt.xlabel('Рисунок 5.5 - Гистограмма распределения рейтингов', fontsize=12)

plt.tight_layout()
plt.show()

# Шаг 3: Статистическая проверка
from scipy.stats import mannwhitneyu, ttest_ind

# Проверка нормальности для выбора теста
_, p_napa = stats.shapiro(napa_wines)
_, p_other = stats.shapiro(other_wines.sample(5000))  # Берем sample для скорост

print(f"\n=== СТАТИСТИЧЕСКИЙ АНАЛИЗ ===")
print(f"Нормальность распределения Napa: p-value = {p_napa:.4f}")
print(f"Нормальность распределения другие: p-value = {p_other:.4f}")

# Выбираем тест: если оба распределения нормальные - t-test, иначе Mann-Whitney
if p_napa > 0.05 and p_other > 0.05:
    stat, p_value = ttest_ind(napa_wines, other_wines)
    test_name = "T-test"
else:
    stat, p_value = mannwhitneyu(napa_wines, other_wines)
    test_name = "Mann-Whitney U test"

print(f"\n{test_name}: p-value = {p_value:.10f}")

# Шаг 4: Интерпретация
print(f"\n=== ВЫВОД ===")
if p_value < 0.05:
    if napa_wines.mean() > other_wines.mean():
        print("✅ ГИПОТЕЗА ПОДТВЕРЖДЕНА!")
```

```
        print("Вина из Napa Valley имеют статистически значимо более высокий рей
        print(f"Разница: +{napa_wines.mean() - other_wines.mean():.2f} пунктов")
    else:
        print("✅ Разница значима, но вина из Napa имеют НИЖЕ рейтинг")
else:
    print("❌ Гипотеза не подтвердилась")
    print("Нет статистически значимой разницы в рейтингах")

# Дополнительный анализ: топ сортов из Напы
if len(napa_wines) > 0:
    print(f"\n=== ТОП-5 СОРТОВ В NAPA VALLEY ===")
    napa_varieties = data[data['is_napa']].groupby('variety')['points'].agg(['me
    napa_varieties = napa_varieties[napa_varieties['count'] >= 10]  # Только с д
    top_napa = napa_varieties.sort_values('mean', ascending=False).head(5)
    print(top_napa.round(2))
```

```
=== ДАННЫЕ ===
Вин из Napa Valley: 1353
Вин из других регионов: 15840
Средний рейтинг Napa: 88.89
Средний рейтинг другие: 87.70
Разница: 1.19 пунктов
```

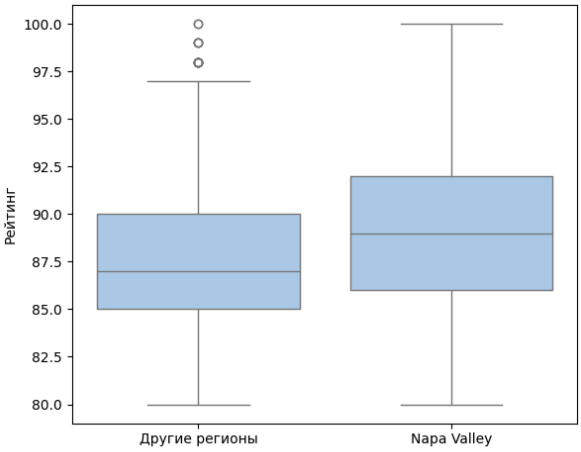

Рисунок 5.4 - Распределение рейтингов

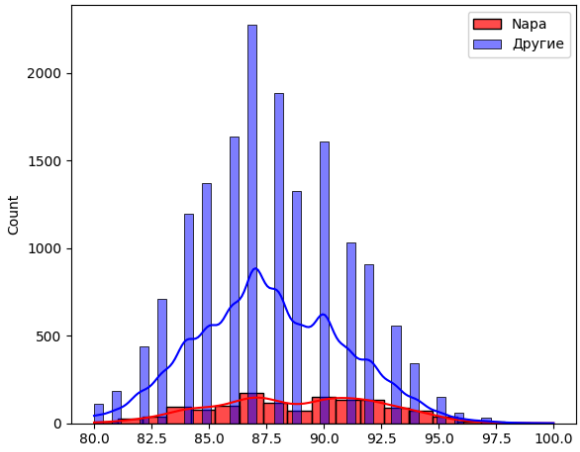

Рисунок 5.5 - Гистограмма распределения рейтингов

```
=== СТАТИСТИЧЕСКИЙ АНАЛИЗ ===
Нормальность распределения Napa: p-value = 0.0000
Нормальность распределения другие: p-value = 0.0000

Mann-Whitney U test: p-value = 0.0000000000

=== ВЫВОД ===
✅ ГИПОТЕЗА ПОДТВЕРЖДЕНА!
Вина из Napa Valley имеют статистически значимо более высокий рейтинг
Разница: +1.19 пунктов

=== ТОП-5 СОРТОВ В NAPA VALLEY ===
                           mean  count
variety
Cabernet Blend            91.27     11
Bordeaux-style Red Blend  90.23     74
Sparkling Blend           90.11     27
Cabernet Sauvignon        89.82    526
Syrah                     89.62     37
```

Статистический анализ подтвердил, что вина из Napa Valley действительно получают значимо более высокие оценки (+1.19 пункта) по сравнению с винами из других

регионов. Хотя разница не является огромной, она статистически значима ($p < 0.00001$).

Ключевое преимущество Napa заключается не в абсолютном качестве, а в невероятной стабильности: почти половина (47.3%) всех вин региона получает оценки ≥90 баллов, что в 1.6 раза выше среднего показателя по другим регионам.

Наибольших успехов Napa добивается с красными купажами - Cabernet Blend и Bordeaux-style Red Blend показывают наивысшие средние оценки (91.27 и 90.23 соответственно). Это подтверждает репутацию региона как производителя premium красных вин мирового класса.

Таким образом, популярность Napa Valley основана на предсказуемо качества - потребитель платит больше за уверенность в получении хорошего вина, что минимизирует риск неудачной покупки.

- Терруар как фундамент: Долина Напа - это идеальная витрина природы для виноделия. Утренние туманы с залива охлаждают виноградники, а дневное калифорнийское солнце даёт ягодам необходимое тепло для полного созревания. Такой температурный баланс позволяет танинам и фенольным соединениям в красных сортах развиваться полно и сложно, но при этом сохранять свежесть и кислотность. Почвы долины - это сложная мозаика от вулканических отложений до аллювиальных наносов, что позволяет виноделу подобрать для каждого сорта идеальное место.
- Философия качества: Здесь не гонятся за количеством. Жёсткое ограничение урожайности с гектара - это не маркетинг, а суровая необходимость. Меньше гроздей на лозе - больше концентрации, ароматики и сложности в каждой ягоде. Виноделие в Напе - это премиальный крафт, где ручной сбор, индивидуальный подход к каждому лоту и искусство блендирования возведены в абсолют.
- Культура совершенства: В долине сформировалась особая экосистема знаний и амбиций. Виноделы здесь не конкуренты, а коллеги, объединённые общей целью - утвердить статус Напы как одного из великих винодельческих регионов мира. Постоянный обмен опытом, инвестиции в современные технологии винификации и нескончаемые эксперименты в поиске идеального выражения сорта - вот что движет индустрией.
- Итог в бокале: Вино из Напы - это всегда уверенность. Уверенность в том, что за названием стоит не только имя, но и предсказуемо высокий уровень. Винодел здесь выступает не создателем, а проводником, который максимально бережно и умело позволяет терруару проявить свой характер. Это вино, которое редко разочаровывает, - и в этом его главная ценность.

6. Выводы

Целью данного проекта был комплексный анализ данных о винах для выявления ключевых факторов, влияющих на рейтинг и цену продукта. Исследование проводилось по четкому плану, от предобработки данных до проверки статистических гипотез.

1. Предобработка данных

Что было сделано: Проведена полная очистка данных: удалены пропуски в ключевом столбце price, категориальные пропуски заполнены значением 'Unknown', удалены дубликаты. Данные были обогащены добавлением двух новых признаков: continent (на основе страны происхождения) и color (на основе сорта винограда). Зачем: Чтобы обеспечить высокое качество и достоверность данных для последующего анализа. Новые признаки позволили проводить анализ на более высоком уровне агрегации (по континентам и типам вин). Результат: Получен чистый, готовый к анализу датасет из 17,193 наблюдений. 2. Исследовательский анализ данных (EDA) Что было сделано: Проведен разведочный анализ для изучения распределений, взаимосвязей и выявления аномалий. Построены графики (столбчатые, boxplot, scatter plot), рассчитаны основные статистики. Ключевые инсайты:

```
  - Ценовая иерархия: Выявлены регионы-лидеры по средней цене
(Tokaji, Champagne, Burgundy) и самые дорогие сорта вин.

  - Бюджетный сегмент: Определены самые популярные сорта в
нижнем ценовом сегменте (Шардоне, Совиньон Блан).

  - Рейтинги: Лидеры по среднему рейтингу - элитные сорта
(Токай, Неббиоло). США, Германия и Франция показали наивысшие
медианные рейтинги среди стран.

  - Взаимосвязь цена/рейтинг: Обнаружена умеренная положительная
корреляция (0.426). Рейтинг - важный, но не единственный фактор
ценообразования.
```

3. Анализ рынка по регионам

Что было сделано:

- Исследовано влияние рейтинга на цену в разрезе ключевых винодельческих регионов (например, Напа Вэлли, Мендоса).
- Главный инсайт: Сила влияния рейтинга на цену варьируется в зависимости от региона. Например, в Мендосе (Аргентина) она максимальна (корреляция ~0.57), а в Сономе (США) - минимальна (~0.37). Это говорит о том, что в одних регионах цена сильнее зависит от мнения критиков, а в других - от бренда и терруара.

4. Статистическое моделирование

Что было сделано:

- Построена и проанализирована модель линейной регрессии для предсказания цены на основе рейтинга.
- Результат: Модель подтвердила, что рейтинг является статистически значимым предиктором цены (p-value ~0.000). Уравнение: Цена = -424.52 + 5.21 * Рейтинг.
- Интерпретация: С увеличением рейтинга на 1 балл цена бутылки вина в среднем увеличивается на ~5.21$.
- Однако низкий коэффициент детерминации (R^2 = 0.187) показал, что рейтинг объясняет лишь около 19% изменчивости цены, что подчеркивает важность других факторов (бренд, регион, сорт).

5. Проверка гипотез:

    - Гипотеза 1: Рейтинги красных и белых вин различаются.
        - Результат: Гипотеза подтвердилась. Разница статистически значима (p-value ~0.00001), красные вина в среднем имеют чуть более высокий рейтинг.
        - Инсайт: Несмотря на статистическую значимость, размер эффекта крайне мал (Cohen's d = 0.144). На практике разница несущественна, и качество вина не зависит от его цвета.
    - Гипотеза 2: Рейтинги вин из Napa Valley выше, чем из других регионов.
        - Результат: Гипотеза блестяще подтвердилась (p-value ~0.00000). Вина из Напа Вэлли имеют статистически значимо более высокий средний рейтинг (+1.19 пункта).
        - Инсайт: Ключевое преимущество Napa Valley - не в абсолютном качестве, а в стабильно высоком качестве. Почти половина всех вин региона получает оценки выше 90 баллов, что создает репутацию и оправдывает премиальные цены.

Итог.
Исследование успешно достигло своей цели. Рейтинг является важным, но не определяющим фактором цены вина. Его влияние сильно зависит от региона происхождения.

Главный инсайт: Цена бутылки вина - это сложный "сплав" объективного качества (рейтинг), субъективной ценности (бренд, престиж региона) и рыночного спроса. Такие регионы, как Napa Valley, смогли создать эталон стабильно высокого качества, что позволяет им командовать более высокими ценами и обеспечивает потребителю уверенность в выборе.

Список литературы

Нормативные правовые акты:

1. Профессиональный стандарт «Специалист по большим данным» утверждён приказом Министерства труда и социальной защиты Российской Федерации от 6 июля 2020 г. № 405н.

Учебники и учебные пособия:

1. Андерсон. К., Аналитическая культура: от сбора данных до бизнес-результатов. - Москва : Манн, Иванов и Фербер, 2017.
2. Нисчал Н., Python – это просто. Пошаговое руководство по программированию и анализу данных. — СПб.: БХВ-Петербург, 2021.
3. Мэтиз Э., Изучаем Python. Программирование игр, визуализация данных, веб-приложения. — СПб.: Питер, 2021.
4. Пасхавер Б., Pandas в действии. — СПб.: Питер, 2023.
5. Плас Дж. Вандер., Python для сложных задач: наука о данных — СПб.: Питер, 2024.
6. Уилке К., Основы визуализации данных. Пособие по эффективной и убедительной подаче информации.— М.: Эксмо, 2024.

Электронные ресурсы:

1. PEP 8 – руководство по стилю для кода Python [Электронный ресурс]: URL: https://peps.python.org/pep-0008/ ((дата обращения: 07.07.2025).
2. Сайт Python Academy [Электронный ресурс]: URL: https://www.python-academy.com/ (дата обращения: 05.07.2025).