

25-09 - WING - HS25 - Fallstudie Apfelsaft - Anlieferung - Mit Lösungsansatz

September 26, 2025

1 Fallstudie: Quantitative Methoden in der Saftproduktion

WING - HS25 - Prof. Adrian Stämpfli

1.1 Kontext

Die Apfelsaft GmbH wurde im Jahr 1995 in einem malerischen Dorf in der Schweiz gegründet. Das Unternehmen begann als kleiner Familienbetrieb, der sich auf die Produktion von hochwertigem Apfelsaft aus regionalen Äpfeln spezialisierte. Mit einer Leidenschaft für Qualität und Nachhaltigkeit hat sich die Apfelsaft GmbH schnell einen Namen gemacht und ist heute einer der führenden Hersteller von Apfelsaft in der Region.

Die Apfelsaft GmbH produziert eine breite Palette von Apfelsaftprodukten, darunter naturtrüben Apfelsaft, klaren Apfelsaft und verschiedene Mischsäfte mit anderen Früchten. Das Unternehmen legt grossen Wert auf die Verwendung von lokal angebauten Äpfeln und setzt auf umweltfreundliche Produktionsmethoden. Die Produktionsstätte ist mit modernster Technologie ausgestattet, um höchste Qualitätsstandards zu gewährleisten.

1.2 Teilaufgabe: Analyse von Anlieferdaten

Du bist Data Scientist (Wirtschaftsingenieur, Qualitätssicherer, Business Developer, Product Manager) bei der Apfelsaft GmbH und hast die Aufgabe die Anlieferung des Obsts besser zu verstehen. Zu diesem Zweck hast du Daten zu Anlieferungen von Äpfeln und Birnen zwischen Juni und November 2024.

Wann wird Obst geliefert? Welche Früchte? Wie viel wird geliefert? Welche Sorten? Welche Qualitäten?

In diesem Jupyter Notebook kannst du mithilfe einer Explorativen Datenanalyse diesen Fragen nachgehen.

1.2.1 Aufbau des Notebooks

Im Notebook werden zuerst fiktive Anlieferdaten generiert. Den Code, welcher die Daten generiert darfst du ignorieren. Deine Aufgabe startet mit dem fertig erstellten Data Frame.

Danach kommt die Explorative Datenanalyse.

Zuerst verschaffst du dir einen Überblick über die Daten. Danach gehst du den einzelnen Fragen nach und vertiefst sie.

1.2.2 Laden der Libraries und Erstellen der fiktiven Anlieferdaten

```
[1]: import random
from datetime import datetime, timedelta
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

random.seed(3)
sns.set_theme(style="whitegrid")

def generate_random_data():
    start_date = datetime(2024, 6, 1)
    end_date = datetime(2024, 11, 30)
    random_date = start_date + timedelta(days=random.randint(0, (end_date -
↪start_date).days))
    random_time = timedelta(hours=random.randint(0, 23), minutes=random.
↪randint(0, 59))
    quantity = round(random.uniform(0.2, 5), 2)
    fruits = ['Äpfel', 'Birnen']
    fruit = random.choices(fruits, weights=[3, 1])[0]

    if fruit == 'Äpfel':
        if random_date.month in [6, 7]:
            varieties = ['Maigold']
        elif random_date.month in [9, 10, 11]:
            varieties = ['Gravensteiner', 'Jonagold']
        else:
            varieties = ['Maigold', 'Gravensteiner', 'Jonagold']
    else:
        if random_date.month in [6, 7, 8]:
            varieties = ['Williams']
        else:
            varieties = ['Conference', 'Williams']

    quality = random.choices([1, 2, 3], weights=[4, 2, 1])[0]
    variety = random.choice(varieties)

    return {
        'Datum und Zeit': random_date + random_time,
        'Menge (Tonnen)': quantity,
        'Frucht': fruit,
        'Qualität': quality,
        'Sorte': variety
    }

dataset = [generate_random_data() for _ in range(200)]
```

```
df = pd.DataFrame(dataset)
df['Monat'] = df['Datum und Zeit'].dt.month
df = df.sort_values(by='Datum und Zeit')
df.head()
```

```
[1]:
```

	Datum und Zeit	Menge (Tonnen)	Frucht	Qualität	Sorte	Monat
120	2024-06-01 02:12:00	2.12	Birnen	1	Williams	6
182	2024-06-04 09:23:00	2.63	Äpfel	1	Maigold	6
20	2024-06-06 12:39:00	3.03	Äpfel	2	Maigold	6
22	2024-06-06 18:03:00	4.78	Äpfel	1	Maigold	6
105	2024-06-09 22:45:00	0.45	Äpfel	1	Maigold	6

1.2.3 Explorative Datenanalyse

Überblick Verschaffen

- Wie viele Datensätze sind enthalten?
- Wie viele Variablen?
- Gibt es fehlende Einträge?
- Welche Typen haben die Variablen?

Hint: Nutze `df.describe()`, `df.info()` sowie die `View data` Funktion.

... Platz für deine Antworten.

```
[2]: # Überblick verschaffen
```

```
df.info()
df.describe()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 200 entries, 120 to 10
Data columns (total 6 columns):
#   Column          Non-Null Count  Dtype
---  -
0   Datum und Zeit  200 non-null   datetime64[ns]
1   Menge (Tonnen)  200 non-null   float64
2   Frucht          200 non-null   object
3   Qualität        200 non-null   int64
4   Sorte           200 non-null   object
5   Monat           200 non-null   int32
dtypes: datetime64[ns](1), float64(1), int32(1), int64(1), object(2)
memory usage: 10.2+ KB
```

```
[2]:
```

	Datum und Zeit	Menge (Tonnen)	Qualität	Monat
count	200	200.000000	200.000000	200.000000
mean	2024-09-05 18:28:21.900000	2.493400	1.510000	8.635000
min	2024-06-01 02:12:00	0.260000	1.000000	6.000000
25%	2024-07-21 06:28:00	1.417500	1.000000	7.000000
50%	2024-09-16 22:51:00	2.435000	1.000000	9.000000

75%	2024-10-19 14:18:15	3.595000	2.000000	10.000000
max	2024-11-30 20:13:00	4.990000	3.000000	11.000000
std	NaN	1.359289	0.694487	1.678051

[]:

Erste Analysen

- Welche Früchte werden angeliefert? Wie häufig? `sns.countplot` ..
- Wie viel wird angeliefert? `sns.histplot` und/oder `sns.boxplot` der Menge
- Welche Sorten werden angeliefert?
- Welche Qualitäten werden angeliefert?
- In welchen Monaten wird angeliefert?

```
[3]: # Anzahl Anlieferungen pro Frucht
plt.figure()
sns.countplot(data=df, x="Frucht")

# Summierte Anliefermenge pro Frucht
plt.figure()
sns.barplot(data=df, x="Frucht", y="Menge (Tonnen)")

# Boxplot Menge pro Frucht
plt.figure()
sns.boxplot(data=df, x="Frucht", y="Menge (Tonnen)")

# Summierte Anliefermenge pro Monat
plt.figure()
sns.barplot(data=df, x="Monat", y="Menge (Tonnen)", estimator=sum)

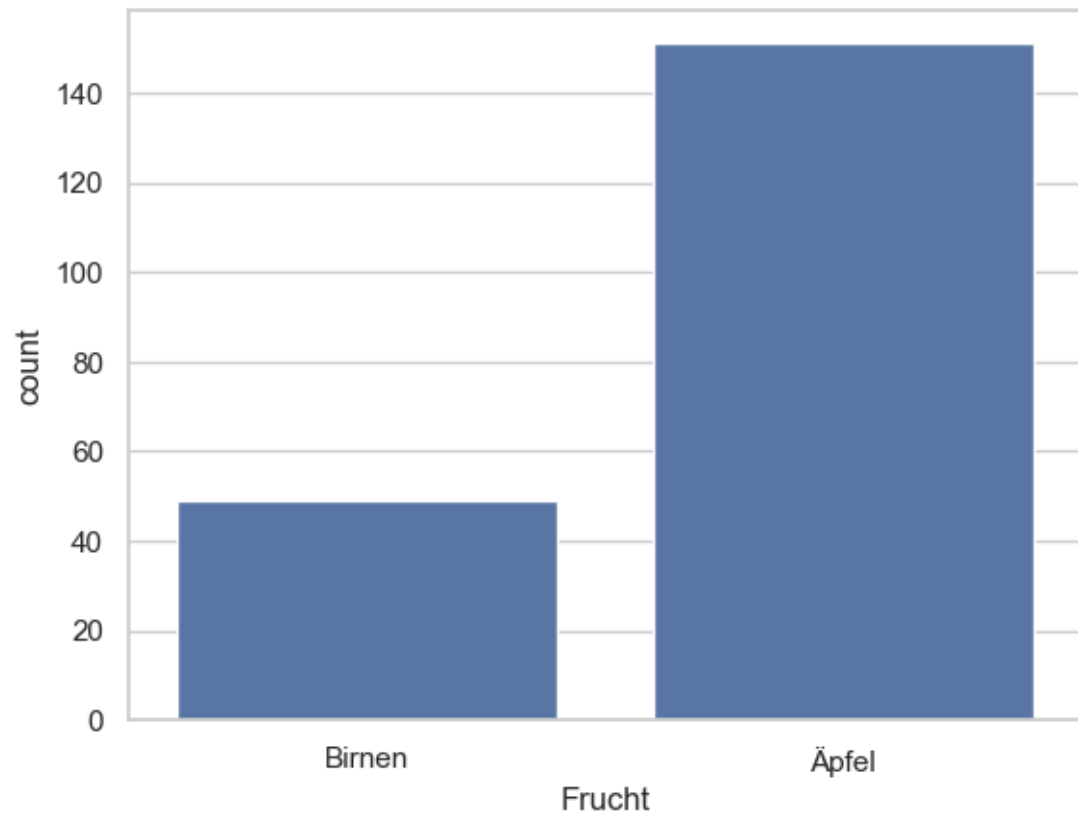
# Durchschnittliche Menge pro Monat
sns.barplot(data=df, x='Monat', y='Menge (Tonnen)')

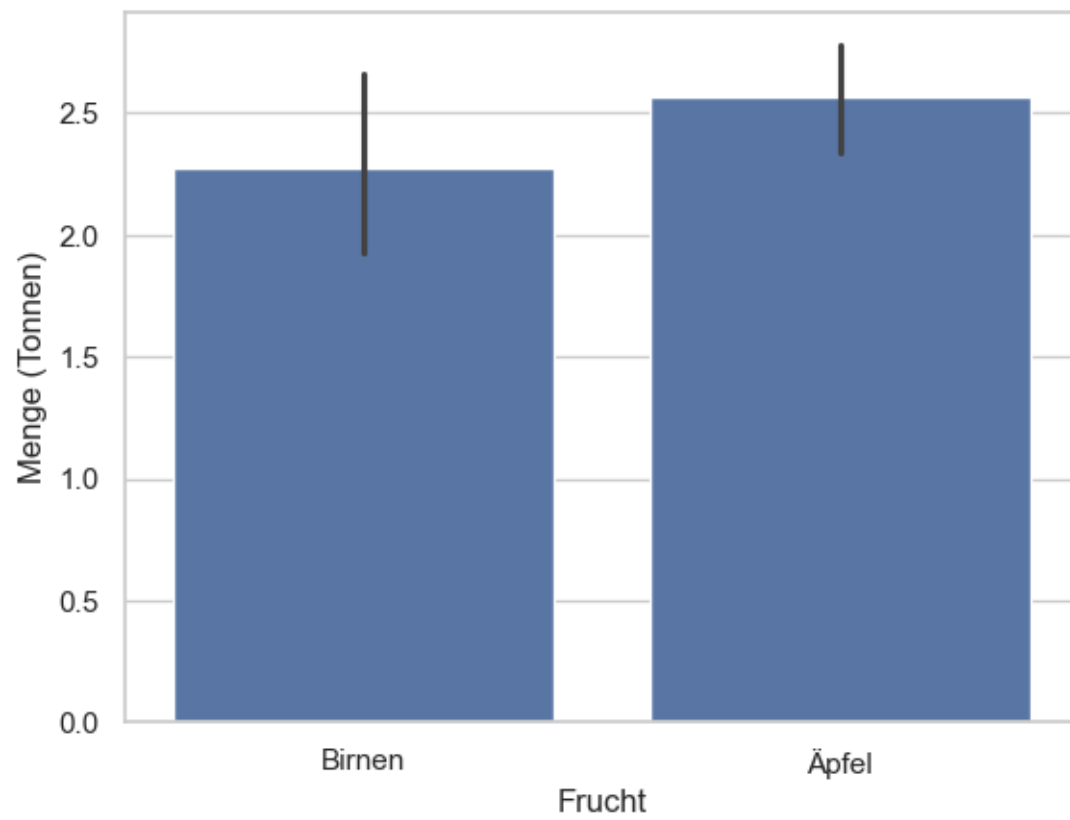
# Saisonale Abhängigkeiten der Sorten (zwei Achsen: Monat & Sorte)
plt.figure()
sns.countplot(data=df, x="Monat", hue="Sorte")

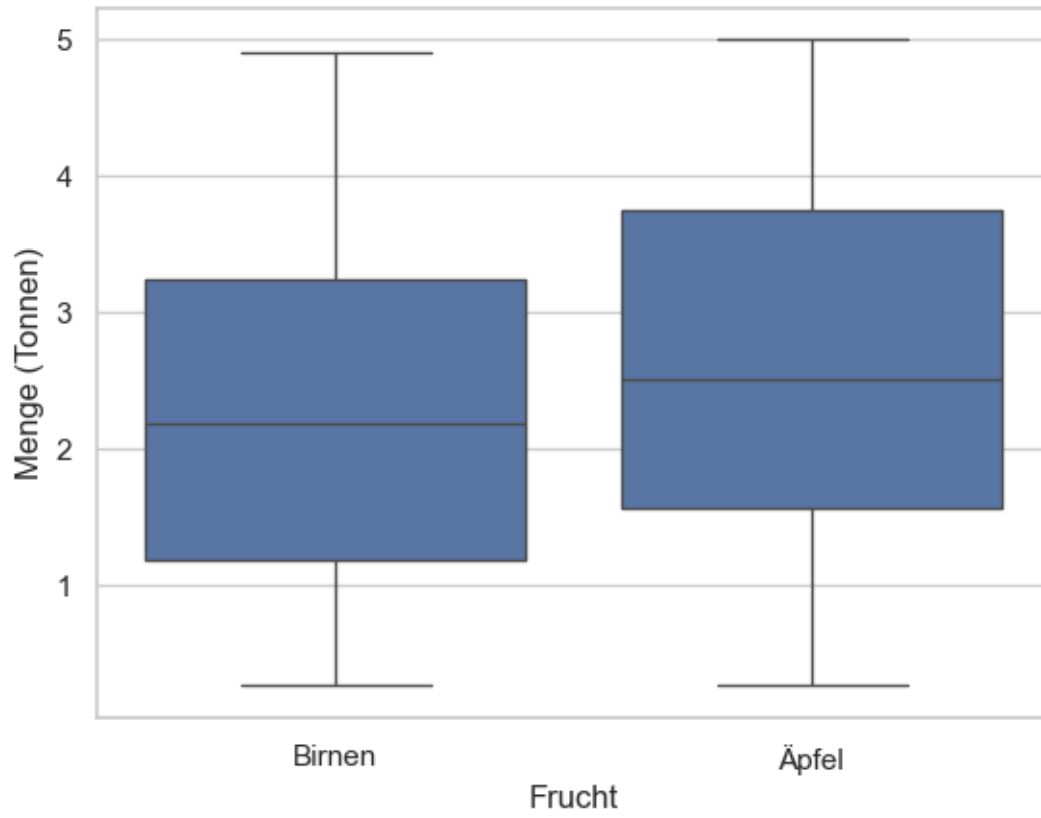
# Boxplot Menge je Sorte
plt.figure()
sns.boxplot(data=df, x="Sorte", y="Menge (Tonnen)")

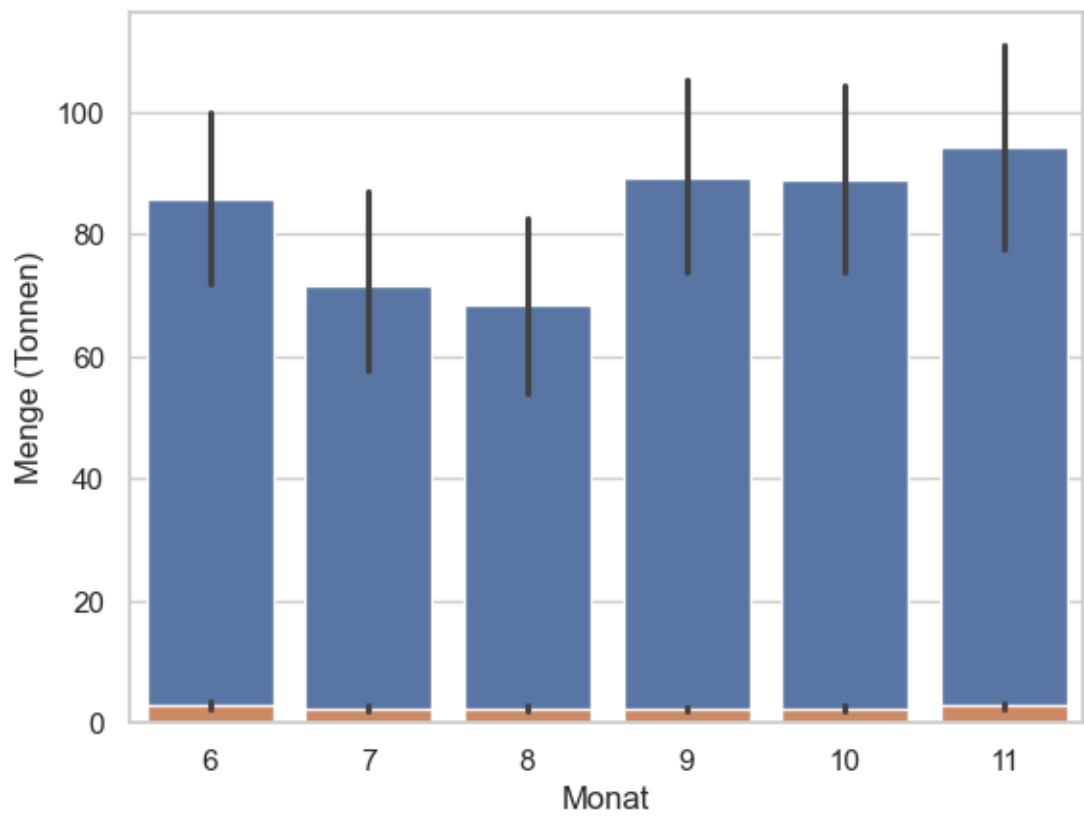
# Summierte Anliefermenge pro Qualitätsstufe
plt.figure()
sns.barplot(data=df, x="Qualität", y="Menge (Tonnen)")

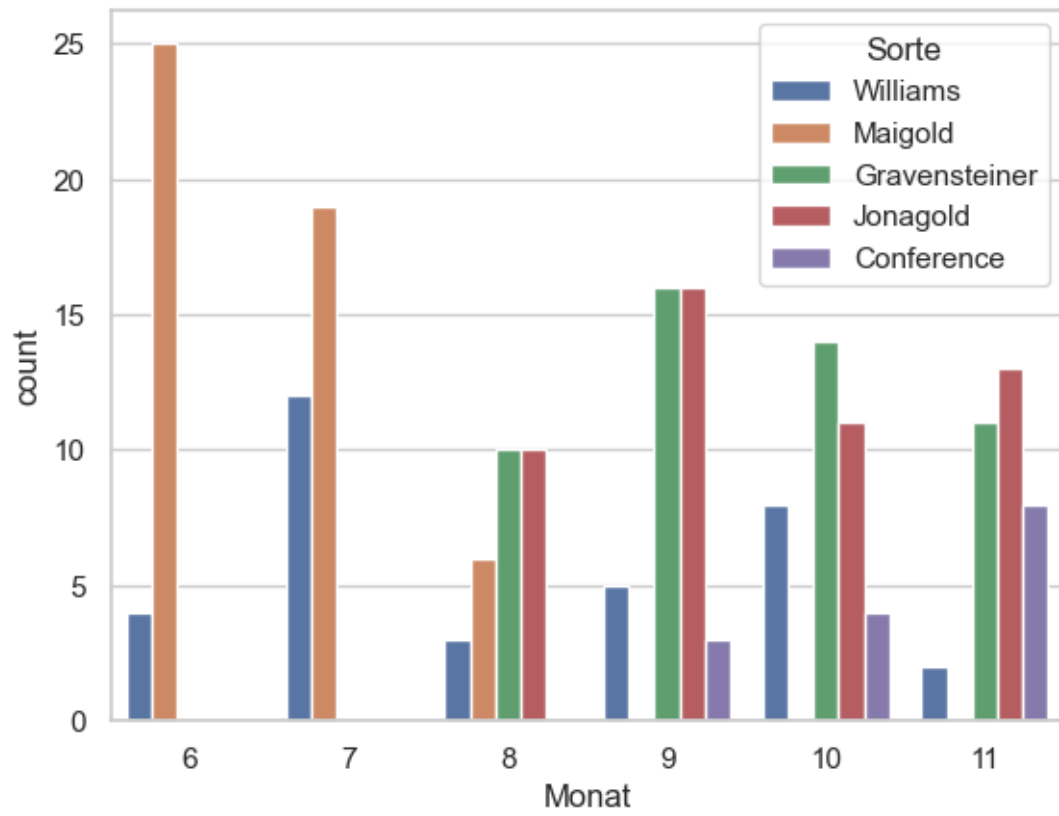
plt.show()
```

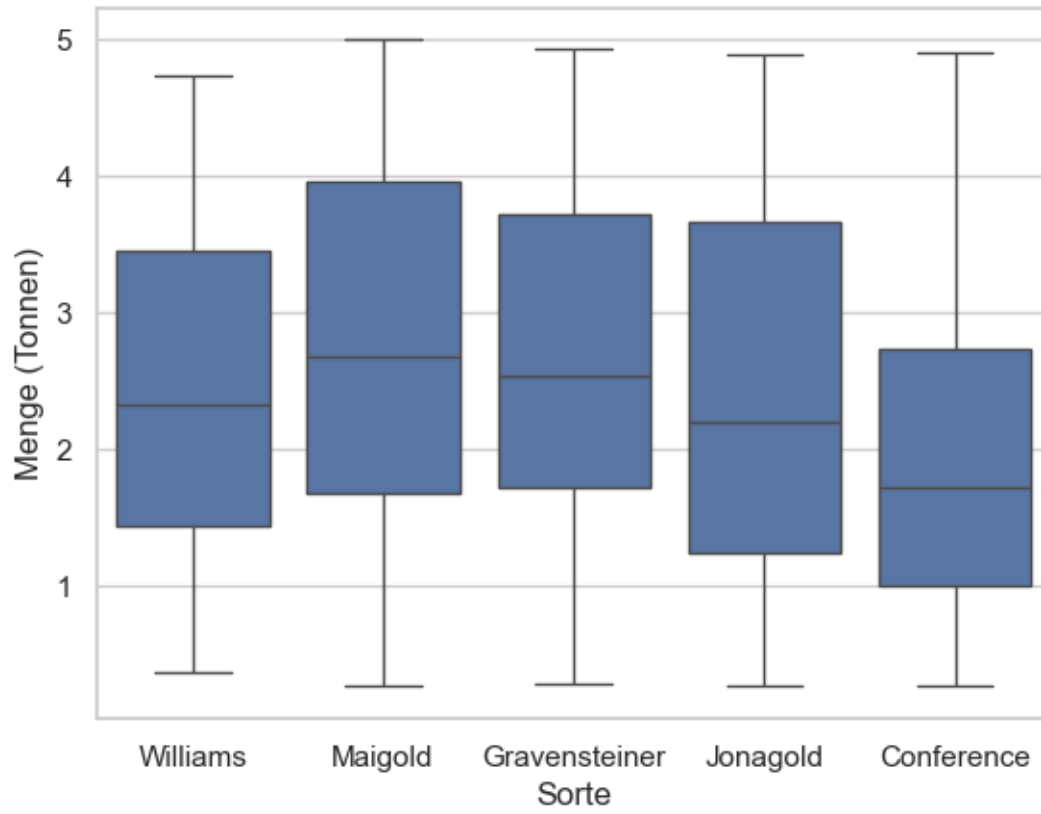


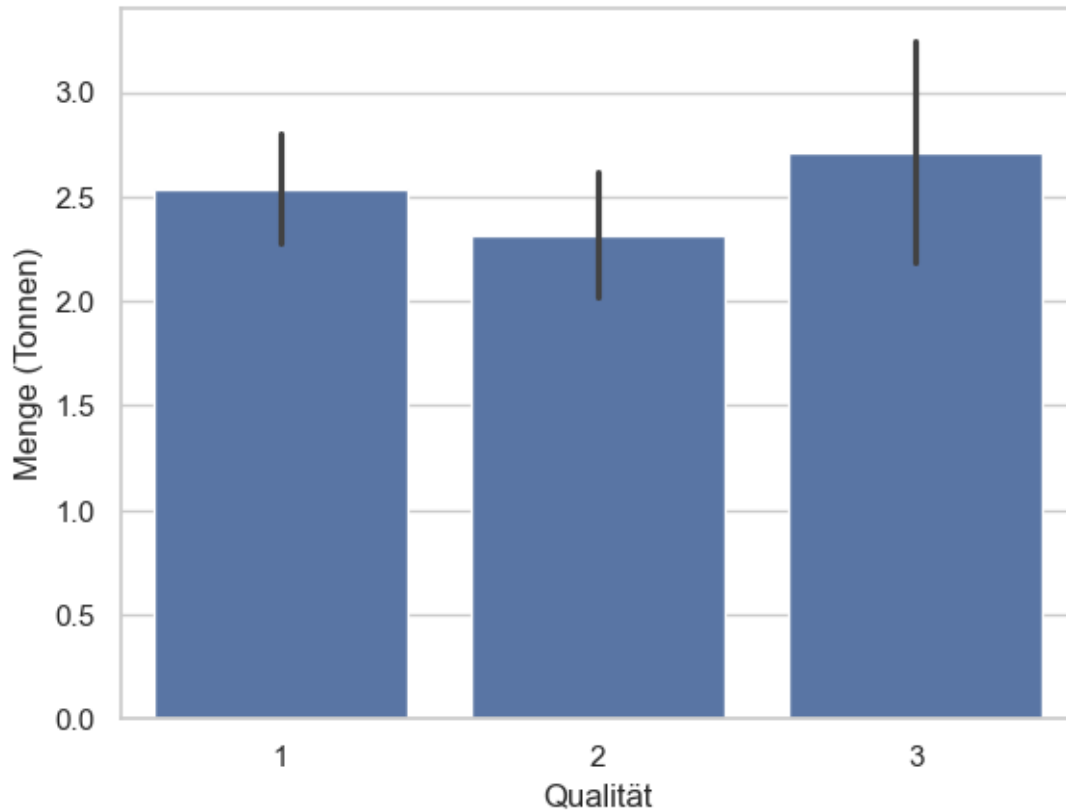












Vertiefende Analysen Hier ist deine Kreativität gefragt. Notiere dir Fragen, welche sich dir stellen und versuche sie mit Plots zu beleuchten.

Einige Beispiele:

- Unterscheidet sich die Qualität je Sorte? je Frucht? je Monat?
- Gibt es Unterschiede in den Anliefermengen je Sorte? je Monat? je Frucht?
- Werden verschiedene Sorten in unterschiedlichen Monaten angeliefert?
- ...

[]: