# Daily Food Nutrition Analysis

**Project Title:**Exploring Nutritional Patterns & Health Insights with Data Analysis

**Overview:**

This project explores daily food consumption patterns using a dataset of 10,000 food entries, analyzing their nutritional content. I conducted data cleaning, exploratory data analysis (EDA), and visualization to uncover trends in macronutrient intake, high-calorie foods, and unhealthy dietary habits.

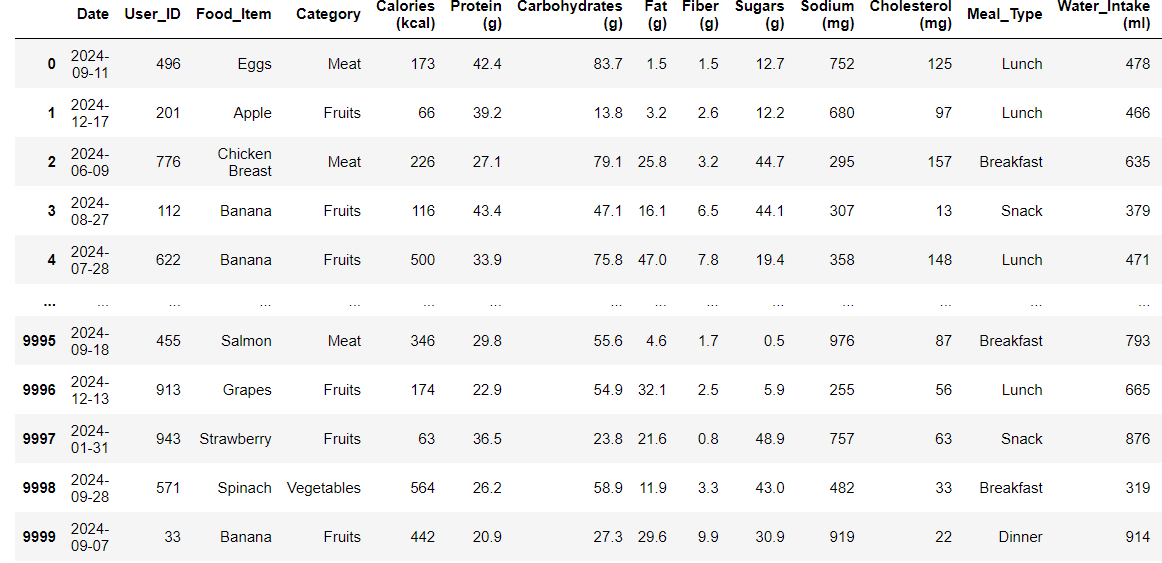
Through Python (Pandas, Matplotlib, Seaborn), I extracted valuable insights into caloric discrepancies, macronutrient distribution, and high-risk food items, enabling better dietary awareness.

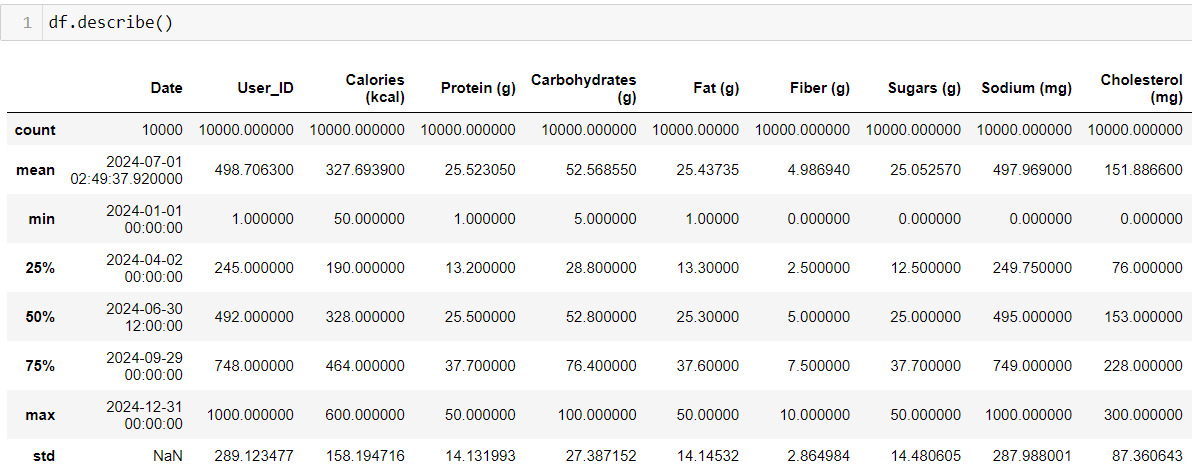
**Key Objectives:**

* **Macronutrient Breakdown**: How protein, carbohydrates, and fats vary across different food categories and meal types.
* **Food Category Insights**: Identifying the food groups contributing the most to daily calorie intake.
* **High-Risk Foods**: Finding foods with excessive sodium, cholesterol, and sugar levels.
* **Trend Analysis Over Time**: Investigating how calorie and nutrient intake fluctuate monthly.
* **Comparison of Provided vs. Calculated Calories**: Assessing data accuracy in reported nutritional values.

**Data Cleaning & Preprocessing**

* Removed **duplicates and missing values**.
* Converted **dates into datetime format** for trend analysis

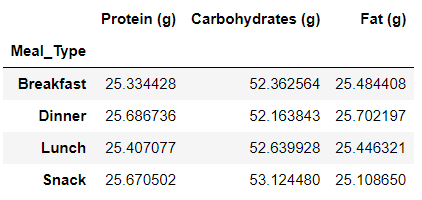




**Key Findings & Visualizations**

**1. Macronutrient Breakdown**

meal\_macronutrient\_ratios = df.groupby("Meal\_Type")[["Protein (g)","Carbohydrates (g)","Fat (g)"]].mean()



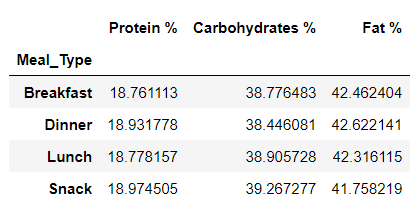
meal\_macronutrient\_ratios["Protein %"] = meal\_macronutrient\_ratios["Protein (g)"] \* 4 / (meal\_macronutrient\_ratios["Protein (g)"] \* 4 + meal\_macronutrient\_ratios["Carbohydrates (g)"] \* 4 + meal\_macronutrient\_ratios["Fat (g)"] \* 9) \* 100

meal\_macronutrient\_ratios["Carbohydrates %"] = meal\_macronutrient\_ratios["Carbohydrates (g)"] \* 4 / (meal\_macronutrient\_ratios["Protein (g)"] \* 4 + meal\_macronutrient\_ratios["Carbohydrates (g)"] \* 4 + meal\_macronutrient\_ratios["Fat (g)"] \* 9) \* 100

meal\_macronutrient\_ratios["Fat %"] = meal\_macronutrient\_ratios["Fat (g)"] \* 9 / (meal\_macronutrient\_ratios["Protein (g)"] \* 4 + meal\_macronutrient\_ratios["Carbohydrates (g)"] \* 4 + meal\_macronutrient\_ratios["Fat (g)"] \* 9) \* 100

meal\_macronutrient\_ratios[["Protein %","Carbohydrates %","Fat %"]]

meal\_macronutrient\_ratios = meal\_macronutrient\_ratios[["Protein %","Carbohydrates %","Fat %"]]



# Stacked bar chart: Macronutrient Breakdown by Meal Type

import matplotlib.pyplot as plt

meal\_macronutrient\_ratios.plot(kind = "bar" , stacked = True, figsize = (8,5) , colormap = "autumn" , alpha = 0.8)

plt.xlabel("Meal Type")

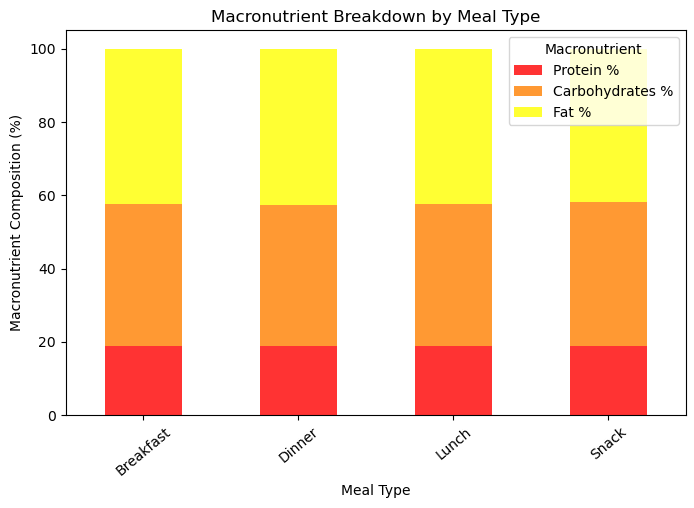
plt.ylabel("Macronutrient Composition (%)")

plt.title("Macronutrient Breakdown by Meal Type")

plt.xticks(rotation = 40)

plt.legend(title = "Macronutrient", loc = "upper right")

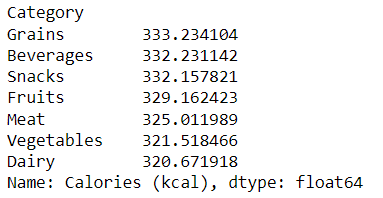
plt.show()



**Insight:** Different meal types and food categories have distinct macronutrient compositions.

**2. Food Category Contributions to Calorie Intake**

food\_category\_calories = df.groupby("Category")["Calories (kcal)"].mean().sort\_values(ascending = False)



# Average calorie intake per food category

plt.figure(figsize=(10,6))

food\_category\_calories.plot(kind = "bar",color = "orange" , alpha = 0.7)

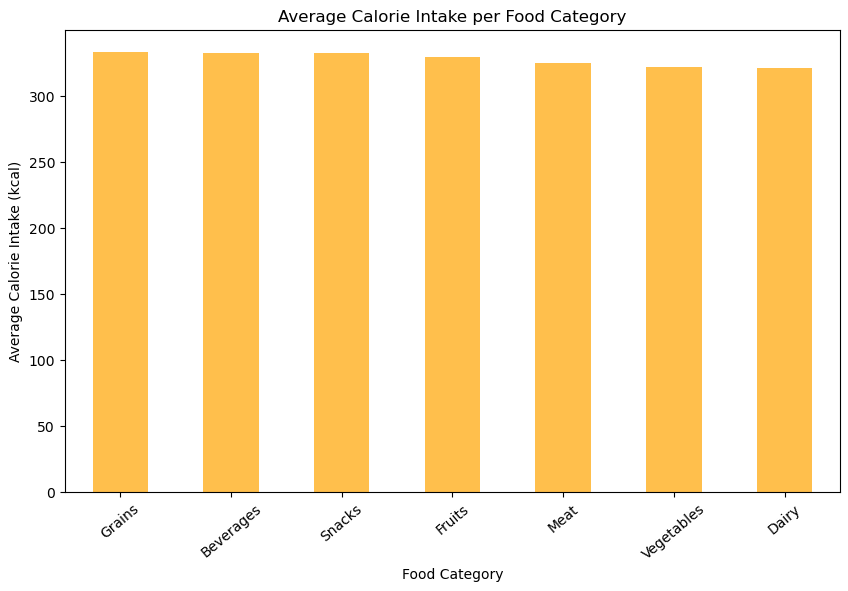
plt.xlabel("Food Category")

plt.ylabel("Average Calorie Intake (kcal)")

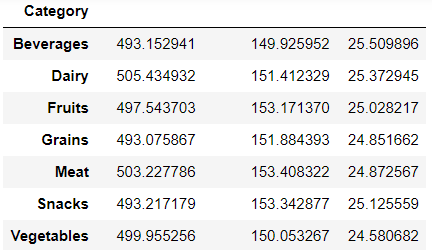
plt.title("Average Calorie Intake per Food Category")

plt.xticks(rotation = 40)

plt.show()



food\_category\_nutrient = df.groupby("Category")[["Sodium (mg)","Cholesterol (mg)","Sugars (g)"]].mean()



# Sodium, Cholesterol, and Sugar content per food category

plt.figure(figsize=(12,6))

food\_category\_nutrient.plot(kind = "bar", colormap = "autumn" , alpha = 0.7)

plt.xlabel("Food Category")

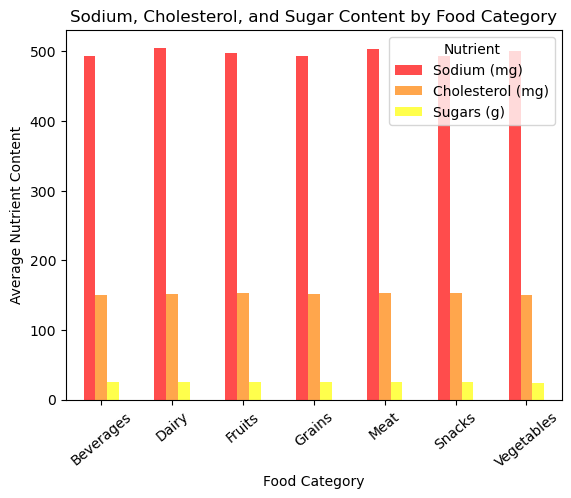
plt.ylabel("Average Nutrient Content")

plt.title("Sodium, Cholesterol, and Sugar Content by Food Category")

plt.xticks(rotation = 40)

plt.legend(title = "Nutrient" , loc = "upper right")

plt.show()



**Insights:**

* Some categories, such as grains and dairy, contribute significantly more to daily calorie intake than vegetables or fruits.
* High sodium content is a concern across all food categories, potentially indicating a reliance on processed or salty foods.
* Cholesterol levels are relatively balanced but higher in animal-based foods, reinforcing the importance of moderation in dairy and meat consumption.
* Sugar intake is lower but still present in some categories, especially Beverages and Fruits, which may indicate added sugars in certain products.

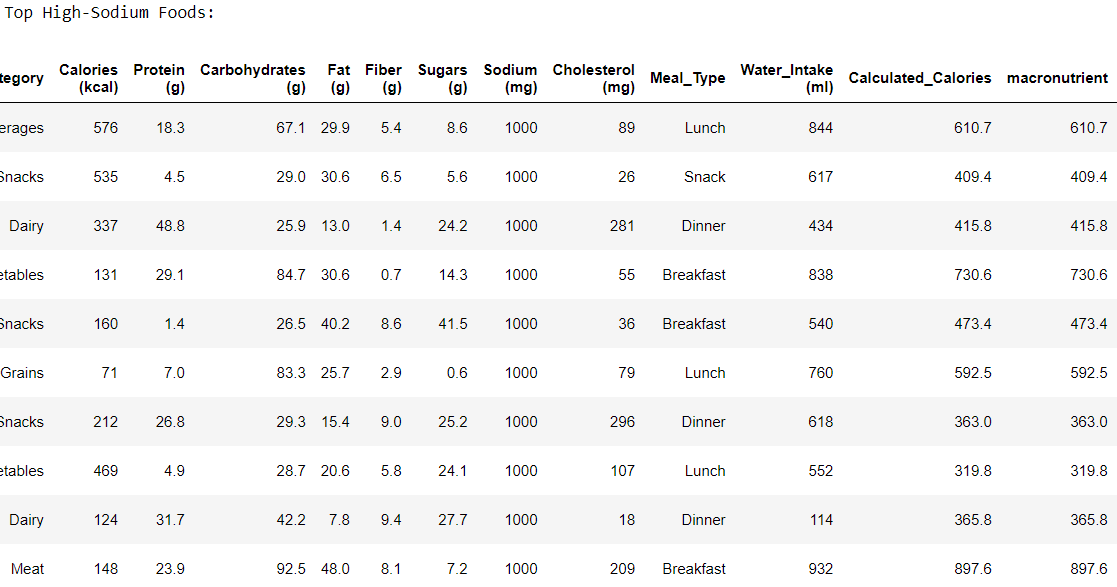
**3. High-Sodium, High-Cholesterol, and High-Sugar Foods**

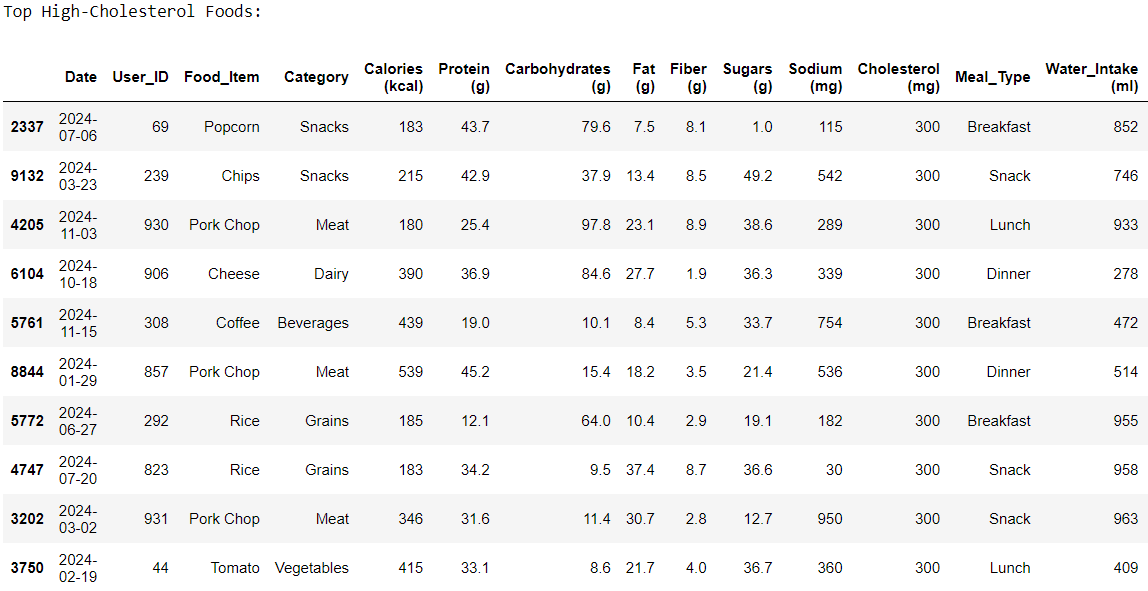
# Identifying Top High-Sodium, High-Cholesterol, and High-Sugar Foods

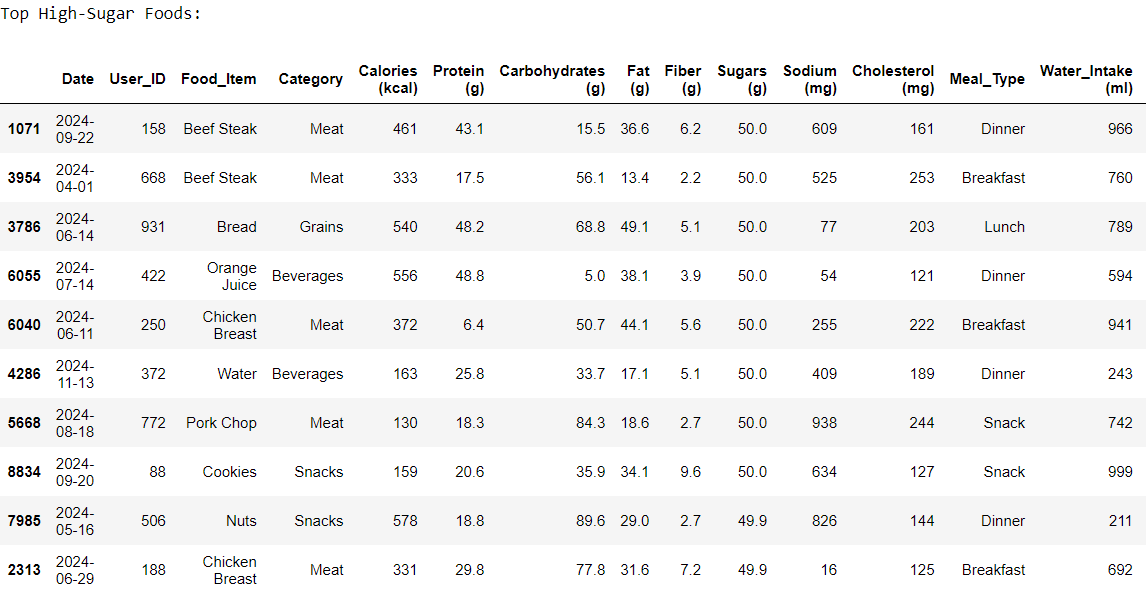
top\_sodium\_foods = df.sort\_values(by = "Sodium (mg)" , ascending= False).head(10)

top\_cholesterol\_foods = df.sort\_values(by = "Cholesterol (mg)" , ascending = False).head(10)

top\_sugar\_foods = df.sort\_values(by="Sugars (g)", ascending=False).head(10)







fig, axes = plt.subplots(1, 3, figsize=(18,6))

# High Sodium Foods

axes[0].barh(top\_sodium\_foods["Food\_Item"], top\_sodium\_foods["Sodium (mg)"], color="red")

axes[0].set\_xlabel("Sodium (mg)")

axes[0].set\_title("Top 10 High-Sodium Foods")

# High Cholesterol Foods

axes[1].barh(top\_cholesterol\_foods["Food\_Item"], top\_cholesterol\_foods["Cholesterol (mg)"], color="blue")

axes[1].set\_xlabel("Cholesterol (mg)")

axes[1].set\_title("Top 10 High-Cholesterol Foods")

# High Sugar Foods

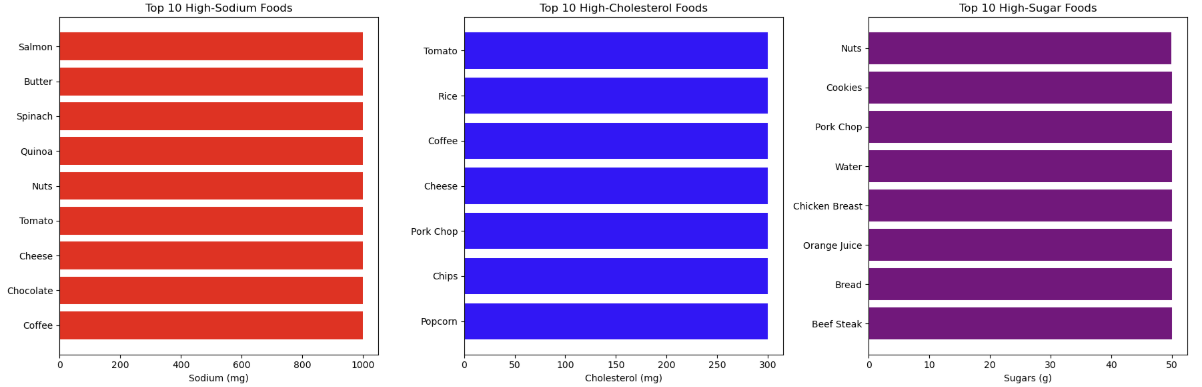
axes[2].barh(top\_sugar\_foods["Food\_Item"], top\_sugar\_foods["Sugars (g)"], color="purple")

axes[2].set\_xlabel("Sugars (g)")

axes[2].set\_title("Top 10 High-Sugar Foods")

plt.tight\_layout()

plt.show()



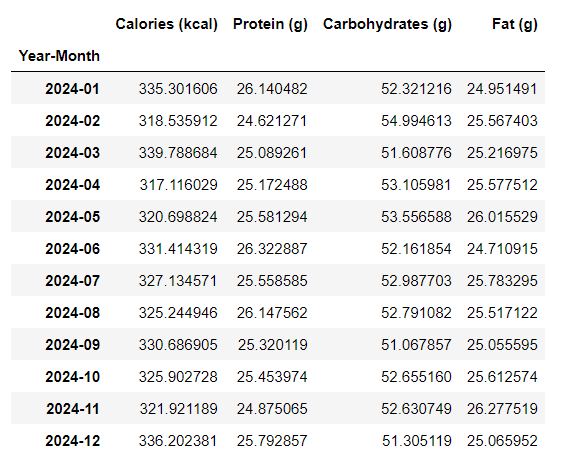
**Insights:**

* Certain natural foods like **salmon, cheese, and spinach** have high sodium content, alongside some processed items like butter and chocolate.
* While dairy (cheese) and meat (pork chop) contain cholesterol, some unexpected items like **tomato and coffee** also appear on the list, possibly due to data inconsistencies.
* Sugary beverages like **orange juice and desserts (cookies)** contain high sugar levels, though some listed items (like pork chop and water) may require further investigation.

**4. Trend Analysis Over Time**

df["Year-Month"] = df["Date"].dt.to\_period("M")

trend\_analysis = df.groupby("Year-Month")[["Calories (kcal)", "Protein (g)", "Carbohydrates (g)", "Fat (g)"]].mean()



# Calorie and nutrient trends over time

plt.figure(figsize=(12,6))

trend\_analysis.plot(marker = 'o' , figsize = (12,6))

plt.xlabel("Time (Year-Month)")

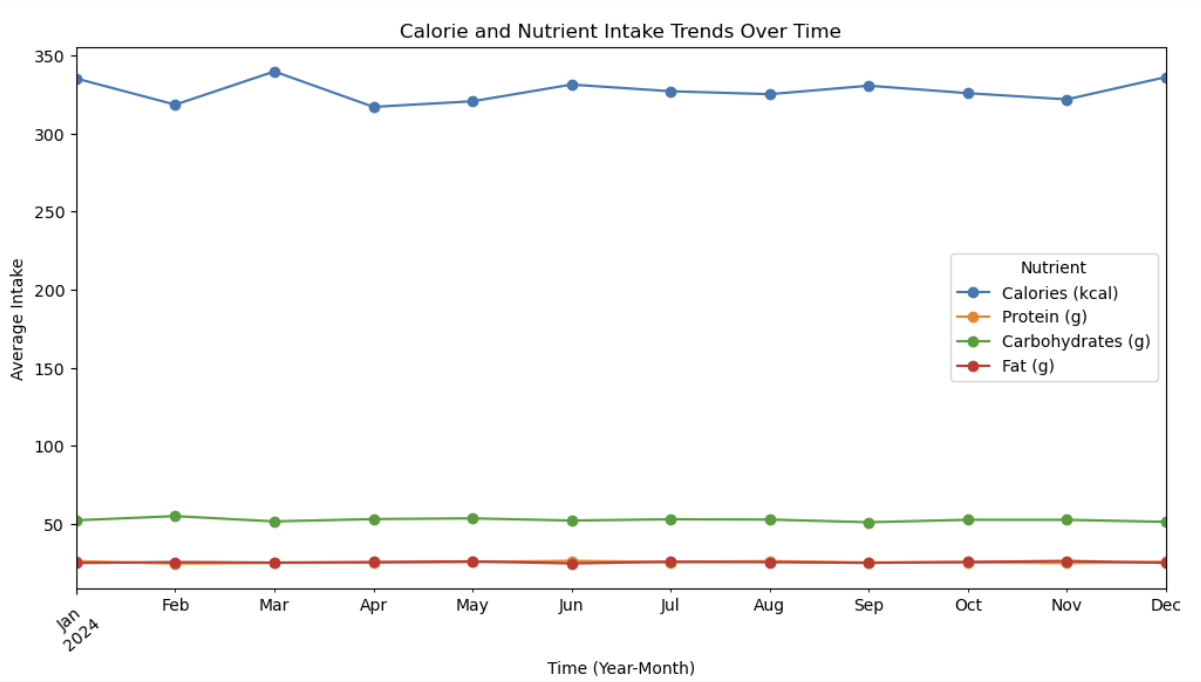
plt.ylabel("Average Intake")

plt.title("Calorie and Nutrient Intake Trends Over Time")

plt.xticks(rotation = 40)

plt.legend(title = "Nutrient")

plt.show()



**Insights:**

* Some months have higher calorie consumption, possibly due to holidays, seasonal food availability, or lifestyle changes.
* The overall diet composition remains stable, implying no major dietary shifts in macronutrient intake.

**5. Comparison of Provided vs. Calculated Calories**

Calories are derived from macronutrients using the following standard values:

* **Protein:** 4 kcal per gram
* **Carbohydrates:** 4 kcal per gram
* **Fat:** 9 kcal per gram

Thus, the calculated calorie formula is:

Calories = (Protein×4) + (Carbohydrates×4) + (Fat×9)

df['Calculated\_Calories'] = df['Protein (g)'] \* 4 + df['Carbohydrates (g)'] \* 4 + df['Fat (g)'] \* 9

df['Macronutrient\_Percentage'] = df['Calculated\_Calories'] / df['Calories (kcal)'] \* 100

import seaborn as sns

plt.figure(figsize=(8,6))

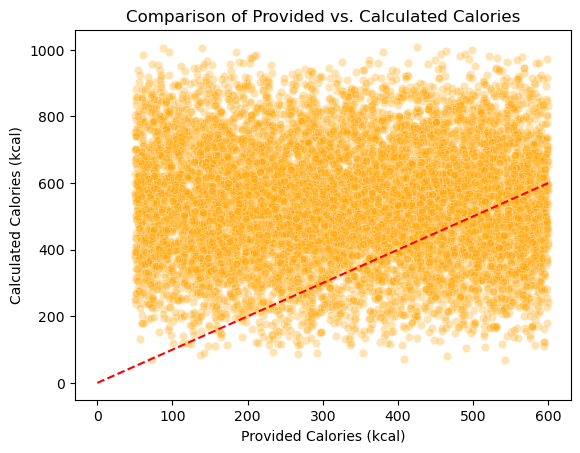
sns.scatterplot(x = df['Calories (kcal)'] , y = df['Calculated\_Calories'] , alpha = 0.3 , color = 'orange')

plt.plot([0, df['Calories (kcal)'].max()], [0, df['Calories (kcal)'].max()], color='red', linestyle='--', label="Ideal Line")

plt.xlabel("Provided Calories (kcal)")

plt.ylabel("Calculated Calories (kcal)")

plt.title("Comparison of Provided vs. Calculated Calories")



**Insights:**

* There is a pattern of underreported calorie values, suggesting potential inaccuracies in the dataset.
* Food items with lower reported calories tend to have a much higher calculated calorie range, which could indicate missing data or incorrect food labeling.
* **Further investigation is needed** to determine if specific food categories have the highest discrepancies and whether certain macronutrients are consistently missing or underestimated.

**Technical Tools and Skills**

* Data Cleaning & Preprocessing
* Data Visualization & Storytelling
* Python (Pandas, Matplotlib, Seaborn)
* Jupyter Notebook

**Next Steps**

* **Further Investigate High Discrepancy Foods:** Identify specific food categories or items with the largest calorie misreporting.
* **Expand Analysis to Dietary Impact:** Investigate how these trends affect health metrics like BMI and nutritional balance.
* **Develop a Predictive Model:** Use machine learning to estimate missing nutritional values.

**Conclusion**

This project highlights the importance of accurate nutritional data and its impact on dietary choices. The analysis uncovered significant discrepancies in reported calorie values, emphasizing the need for better data integrity in food tracking. Additionally, identifying high-risk foods and nutrient trends provides valuable insights for health-conscious decision-making.