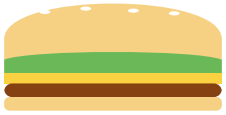

FAST FOOD ANALYSIS

Data Science · Mr. Merrick · December 3, 2025

Overview

Using the **Fast Food Nutrition** dataset (available on Kaggle at https://www.kaggle.com/datasets/ulrikthygepedersen/fastfood-nutrition/), conduct a light exploratory analysis in **R**.

Guiding Questions

Q1. Summarize the dataset.
Report the number of rows, columns, and restaurants. Compute basic summaries for key variables like `calories`, `total_fat`, and `sodium`. Show the distribution of calories (e.g., histogram) and comment on what counts as a "small", "medium", and "large" item in this dataset.

Q2. Visualize nutrition distributions.
Make one or more plots showing the distributions of variables such as `calories`, `total_fat`, `sugar`, and `protein`. You could use separate histograms, density plots, or a facetted plot. Briefly compare which nutrients appear most skewed and which have the widest spread.

Q3. Connect to your daily energy needs (TDEE).
Use an online TDEE calculator (e.g., https://www.calculator.net/tdee-calculator.html) or a value given by your instructor to estimate your personal **TDEE** (Total Daily Energy Expenditure) in kcal. Create a new variable that gives the **percent of your daily calories** each item represents:

$$\texttt{cal\_pct\_day} = \frac{\texttt{calories}}{\texttt{TDEE}} \times 100.$$

Show a table or plot of the items that make up at least 25% of your daily calories and comment on whether they seem "worth it".

Q4. Sodium and daily limits.
Assume a recommended daily sodium limit of 2300 mg (or another value your instructor specifies). Create a new variable for the **percent of daily sodium** each item contains:

$$\texttt{sodium\_pct\_day} = \frac{\texttt{sodium}}{2300} \times 100.$$

Make a plot or table highlighting items that exceed 50% and 100% of this limit, and briefly discuss your findings.

Q5. How do restaurants compare?
Group by `restaurant` and compute summaries such as average calories, average `total_fat`, and average `sodium` per item. Compare at least two restaurants (e.g., Mcdonalds vs Subway) on these metrics. Include a small table and at least one plot (e.g., bar chart of average calories or fat by restaurant) and interpret key differences.

Q6. Can fat predict calories? (Linear regression)
Investigate the relationship between `total_fat` and `calories`. Make a scatter plot with `total_fat` on the x-axis and `calories` on the y-axis, add a linear trend line, and fit the regression model:

$$\texttt{calories} \sim \texttt{total\_fat}.$$

Report the correlation, the slope (how many extra calories per gram of fat), and comment on how well fat alone explains calories.

Dataset Description and Citation

The dataset includes information for menu items from several fast food chains: restaurant name, item name, calories, calories from fat, fat grams, carbohydrates, sugar, protein, sodium, and certain vitamins.

- **Citation:** Ulrik Thyge Pedersen. *Fast Food Nutrition*. Kaggle. Retrieved from https://www.kaggle.com/datasets/ulrikthygepedersen/fastfood-nutrition/data

Group Tutorial / Instructor Key (R only)

```r
# ========================================
# Intro to R: Fast Food Nutrition
# Gentle EDA + Daily Needs + Regression
# Dataset: FastFood.csv (Kaggle: Ulrik Thyge Pedersen)
#
# Questions covered:
# Q1: Summarize dataset + calories distribution
# Q2: Visualize nutrition distributions
# Q3: TDEE + % of daily calories
# Q4: Sodium + % of daily limit
# Q5: Restaurant comparisons
# Q6: Fat vs Calories (linear regression)
# ========================================

# ----- Packages -----
library(readr)
library(dplyr)
library(ggplot2)
library(janitor)
library(tidyr)

# (Optional) nicer tables / models:
# install.packages("broom")
# library(broom)


# ========================================
# Load & Inspect Data
# ========================================
# Make sure FastFood.csv is in your working directory, or set the path explicitly:
# setwd("path/to/your/folder")

fastfood <- read_csv("FastFood.csv", show_col_types = FALSE)


# Look at basic structure
names(fastfood)
dim(fastfood)
head(fastfood, 5)

# Optional: glimpse for a quick overview
glimpse(fastfood)


# ========================================
# Q1. Summarize the dataset
# ========================================

# Basic summaries for key variables
summary_stats <- fastfood %>%
  summarise(
    rows = n(),
    cols = ncol(fastfood),
    restaurants = n_distinct(restaurant),
```

```r
    avg_calories = mean(calories, na.rm = TRUE),
    avg_fat = mean(total_fat, na.rm = TRUE),
    avg_sodium = mean(sodium, na.rm = TRUE)
  )

summary_stats

# Distribution of calories
fastfood %>% ggplot(aes(x = calories)) +
  geom_histogram(bins = 30, color = "white", fill = "steelblue") +
  labs(title = "Distribution of Fast Food Item Calories",
       x = "Calories per item",
       y = "Count")

# Optional: summary of calories (min, median, mean, etc.)
summary(fastfood$calories)


# ========================================
# Q2. Visualize nutrition distributions
# ========================================

# Select a few nutrients of interest and pivot longer for faceting
nutrients <- fastfood %>%
  select(item, restaurant, calories, total_fat, sugar, protein) %>%
  pivot_longer(
    cols = c(calories, total_fat, sugar, protein),
    names_to = "nutrient",
    values_to = "value"
  )

# Facetted histograms for each nutrient
nutrients %>% ggplot(aes(x = value)) +
  geom_histogram(bins = 30, color = "white", fill = "seagreen") +
  facet_wrap(~ nutrient, scales = "free_x") +
  labs(title = "Nutrition Distributions (Selected Nutrients)",
       x = "Value",
       y = "Count")

# Optional: density version (for comparison)
ggplot(nutrients, aes(x = value)) +
  geom_density(fill = "seagreen", alpha = 0.5) +
  facet_wrap(~ nutrient, scales = "free_x") +
  labs(title = "Nutrition Distributions (Density)",
       x = "Value",
       y = "Density")


# ========================================
# Q3. TDEE + percent of daily calories
# ========================================

# Step 1: decide on a TDEE (kcal/day)
# Students can each choose their own value from:
# https://www.calculator.net/tdee-calculator.html
# Here we just pick an example value. In class, they should edit this.
```

```r

tdee <- 2400 # <-- change to your own TDEE (e.g., 2000, 2300, 2600, ...)

# Step 2: create % of daily calories for each item
fastfood <- fastfood %>%
  mutate(
    cal_pct_day = calories / tdee * 100
  )

# Look at items that are at least 25% of daily calories
big_cal_items <- fastfood |>
  filter(cal_pct_day >= 25) |>
  arrange(desc(cal_pct_day)) |>
  select(restaurant, item, calories, cal_pct_day)

big_cal_items

# Histogram of % daily calories
fastfood %>% ggplot(aes(x = cal_pct_day)) +
  geom_histogram(bins = 30, color = "white", fill = "orange") +
  labs(title = "Percent of Your Daily Calories per Item",
       x = "% of daily calories",
       y = "Count")


# ========================================
# Q4. Sodium + percent of daily limit
# ========================================

# Assume daily sodium limit (mg)
daily_sodium_limit <- 2300 # adjust if you want a different guideline

fastfood <- fastfood %>%
  mutate(
    sodium_pct_day = sodium / daily_sodium_limit * 100
  )

# Items that are at least 50% of daily sodium
high_sodium <- fastfood %>%
  filter(sodium_pct_day >= 50) %>%
  arrange(desc(sodium_pct_day)) %>%
  select(restaurant, item, sodium, sodium_pct_day)

high_sodium

# Histogram of % daily sodium
ggplot(fastfood, aes(x = sodium_pct_day)) +
  geom_histogram(bins = 30, color = "white", fill = "tomato") +
  labs(title = "Percent of Daily Sodium per Item",
       x = "% of daily sodium limit",
       y = "Count")

# Optional: zoom in to 0-200% for more readable plot
fastfood %>% filter(sodium_pct_day <= 200) %>%
  ggplot( aes(x = sodium_pct_day)) +
  geom_histogram(bins = 30, color = "white", fill = "tomato") +
```

```r
  labs(title = "Percent of Daily Sodium per Item (0-200%)",
       x = "% of daily sodium limit",
       y = "Count")


# ========================================
# Q5. Restaurant comparisons
# ========================================

# Summaries by restaurant
restaurant_summary <- fastfood %>%
  group_by(restaurant) %>%
  summarise(
    items = n(),
    avg_cal = mean(calories, na.rm = TRUE),
    avg_fat = mean(total_fat, na.rm = TRUE),
    avg_sodium = mean(sodium, na.rm = TRUE),
    avg_sugar = mean(sugar, na.rm = TRUE),
    .groups = "drop"
  ) |>
  arrange(desc(avg_cal))

restaurant_summary

# Bar chart: average calories by restaurant
restaurant_summary %>% ggplot( aes(x = reorder(restaurant, avg_cal), y = avg_cal)) +
  geom_col(fill = "tan3") +
  coord_flip() +
  labs(title = "Average Calories per Item by Restaurant",
       x = "Restaurant",
       y = "Average calories")

# Example pairwise comparison: choose two restaurants
pair_restaurants <- c("Mcdonalds", "Subway") # adjust names to match dataset

pair <- fastfood %>%
  filter(restaurant %in% pair_restaurants)

# Summary table for the pair
pair_summary <- pair %>%
  group_by(restaurant) %>%
  summarise(
    items = n(),
    avg_cal = mean(calories, na.rm = TRUE),
    avg_fat = mean(total_fat, na.rm = TRUE),
    avg_sodium = mean(sodium, na.rm = TRUE),
    avg_sugar = mean(sugar, na.rm = TRUE),
    .groups = "drop"
  )

pair_summary

# Boxplot comparing calories for the two restaurants
pair %>% ggplot(aes(x = restaurant, y = calories)) +
  geom_boxplot(fill = "skyblue") +
  labs(title = "Calories by Restaurant (Example Pair)",
```

```r
      x = "Restaurant",
      y = "Calories per item")


# ========================================
# Q6. Fat vs Calories (linear regression)
# ========================================

# Keep only complete cases for these two variables
fastfood_complete <- fastfood %>%
  filter(!is.na(calories), !is.na(total_fat))

# Scatter plot with regression line
fastfood_complete %>% ggplot(aes(x = total_fat, y = calories)) +
  geom_point(alpha = 0.5) +
  geom_smooth(method = "lm", se = TRUE) +
  labs(title = "Total Fat vs Calories",
      x = "Total fat (g)",
      y = "Calories")

# Correlation between fat and calories
cor_fat_cal <- cor(fastfood_complete$total_fat,
                   fastfood_complete$calories)
cor_fat_cal

# Fit linear model: calories ~ total_fat
model <- lm(calories ~ total_fat, data = fastfood_complete)
summary(model)

# If using broom, you can get a tidy table:
# library(broom)
# tidy(model)
# glance(model)

# You might have students interpret:
# - the intercept (calories when fat = 0)
# - the slope (extra calories per 1g of fat)
# - R-squared (how much of the variability in calories is explained by fat alone)
```