Explore how LEGO set size, price, themes, and release year interact. Your goal: simple summaries, clean plots, and short interpretations.

Overview

Using the **LEGO Sets and Prices Over Time** dataset, conduct a light exploratory analysis in **R**. Include 2–3 sentence descriptions under each figure/table. Cover at least **three** questions (Q2 on missing data counts as one).

Guiding Questions

Q1. Summarize the dataset.
Rows, columns, year range, average pieces and average price. Describe the distribution of set piece-counts.

Q2. Missing data overview.
Which variables (`name`, `theme`, `year`, `pieces`, `usd_msrp`) have missing values? Show a percent-missing bar chart and state how you will handle missingness in later questions.

Q3. What is "value"? (Price per piece)
Define PPP = `usd_msrp/pieces`. List several "best value" sets (e.g., $\geq$ 100 pieces). Show a PPP histogram. *Note: PPP can be very skewed; you may filter clear outliers (e.g., PPP $<$ \$1). Say what you did and comment on why the distribution of PPP is heavily skewed.*

Q4. How do themes differ?
Compare themes by average PPP and/or average pieces. Limit to themes with at least 10 sets. Show a table or bar chart.

Q5. Two-theme comparison (pick any two).
Compare two specific themes (e.g., *Star Wars* vs *Harry Potter*) on price, pieces, and PPP. Include a small summary table and one plot (e.g., two boxplots or a side-by-side bar chart). Briefly interpret the differences.

Q6. Are bigger sets always pricier?
Make a scatter plot of `pieces` vs `usd_msrp` with a trend line and report the correlation.

Q7. How have sets changed over time?
Plot yearly averages for pieces, price, and/or PPP and describe any trends. Also explore the average changes in price/pieces/PPP for Star Wars, Disney, and Harry Potter themed sets.

Dataset Description and Citation

The dataset includes set name/ID, theme, year, piece counts, and prices.

- **Citation:** Alex Racapé. *LEGO Sets and Prices Over Time*. Kaggle. Retrieved from https://www.kaggle.com/datasets/alexracape/lego-sets-and-prices-over-time/data

Group Tutorial (R only)

Setup and Data Import

```r
# ========================================
# Intro to R: LEGO Sets (Prices & Pieces)
# Gentle EDA + Missing Data basics (R only)
# Packages: readr, dplyr, ggplot2, janitor
# File needs: name, theme, year, pieces, usd_msrp
# ========================================
library(readr)
library(dplyr)
library(ggplot2)
library(janitor)

# Read & clean ------------------------------------------------
lego <- read_csv("lego.csv", show_col_types = FALSE) |>
  clean_names()

# If needed (uncomment):
# lego <- lego |>
# rename(pieces = num_parts, usd_msrp = retail_price)

names(lego)
dim(lego)
head(lego, 5)
```

Q1. Summarize

```r
lego |>
  summarise(
    rows = n(),
    cols = ncol(lego),
    avg_pieces = mean(pieces, na.rm = TRUE),
    avg_price = mean(usd_msrp, na.rm = TRUE),
    first_year = min(year, na.rm = TRUE),
    last_year = max(year, na.rm = TRUE)
  )

ggplot(lego, aes(x = pieces)) +
  geom_histogram(bins = 30, color = "white", fill = "steelblue") +
  labs(title = "Distribution of LEGO Set Piece Counts",
       x = "Pieces", y = "Count")
```

Q2. Missing Data Overview

```r
pct <- function(x) round(100 * x, 1)

missing_overview <- lego |>
  summarise(
    n = n(),
    missing_name = sum(is.na(name)),
    missing_theme = sum(is.na(theme)),
```

```
    missing_year = sum(is.na(year)),
    missing_pieces = sum(is.na(pieces)),
    missing_usd = sum(is.na(usd_msrp))
  ) |>
  mutate(
    pct_name = pct(missing_name / n),
    pct_theme = pct(missing_theme / n),
    pct_year = pct(missing_year / n),
    pct_pieces = pct(missing_pieces / n),
    pct_usd = pct(missing_usd / n)
  )

print(missing_overview)

miss_long <- tibble(
  variable = c("name","theme","year","pieces","usd_msrp"),
  pct_missing = c(
    pct(mean(is.na(lego$name))),
    pct(mean(is.na(lego$theme))),
    pct(mean(is.na(lego$year))),
    pct(mean(is.na(lego$pieces))),
    pct(mean(is.na(lego$usd_msrp)))
  )
)

ggplot(miss_long, aes(x = variable, y = pct_missing)) +
  geom_col(fill = "tomato") +
  labs(title = "Percent Missing by Variable",
       x = "Variable", y = "Percent missing") +
  ylim(0, 100)
```

Q3. Value (PPP)

```
# Naive definition (for discussion)
# lego |> mutate(ppp = usd_msrp/pieces)

# Safe PPP with guards for zeros / NAs
lego <- lego |>
  mutate(ppp = ifelse(!is.na(usd_msrp) & !is.na(pieces) & pieces > 0,
                      usd_msrp / pieces, NA_real_))

lego |>
  filter(pieces >= 100, is.finite(ppp)) |>
  arrange(ppp) |>
  select(name, theme, year, pieces, usd_msrp, ppp) |>
  print(n = 50)

# Distribution including outliers
ggplot(lego |> filter(!is.na(ppp)), aes(x = ppp)) +
  geom_histogram(bins = 30, color = "white", fill = "seagreen") +
  labs(title = "Distribution of Price per Piece (PPP)",
       x = "USD per piece", y = "Count")

# Inspect high PPP entries
lego |>
```

```
  arrange(desc(ppp)) |>
  select(name, theme, year, pieces, usd_msrp, ppp) |>
  print(n = 100)

# Optional: trimmed view for a cleaner shape
lego_trim <- lego |> filter(!is.na(ppp), ppp > 0, ppp < 1.5)
ggplot(lego_trim, aes(x = ppp)) +
  geom_histogram(bins = 30, color = "white", fill = "seagreen") +
  labs(title = "PPP (filtered < $1.5)",
       x = "USD per piece", y = "Count")
```

Q4. Themes Compared

```
theme_summary <- lego |>
  filter(!is.na(ppp)) |>
  group_by(theme) |>
  summarise(
    sets = n(),
    avg_ppp = mean(ppp, na.rm = TRUE),
    avg_pieces = mean(pieces, na.rm = TRUE),
    .groups = "drop"
  ) |>
  filter(sets >= 10) |>
  arrange(avg_ppp)

print(theme_summary, n = 100)

top40 <- theme_summary |> slice_min(avg_ppp, n = 40)

ggplot(top40, aes(x = reorder(theme, avg_ppp), y = avg_ppp)) +
  geom_col(fill = "tan3") +
  coord_flip() +
  labs(title = "Top Themes by Value (Lowest Avg PPP)",
       x = "Theme", y = "Average USD per piece)")

# (All-themes boxplot often messy due to many categories)
lego |> filter(!is.na(ppp)) |>
  ggplot(aes(x = theme, y = usd_msrp)) + geom_boxplot()
```

Q5. Two-Theme Comparison

```
themes_to_compare <- c("Star Wars", "Harry Potter") # edit as desired

pair <- lego |> filter(theme %in% themes_to_compare)

pair |> group_by(theme) |>
  summarise(
    sets = n(),
    avg_price = mean(usd_msrp, na.rm = TRUE),
    avg_pieces = mean(pieces, na.rm = TRUE),
    avg_ppp = mean(ppp, na.rm = TRUE),
    .groups = "drop"
  )

```

```
ggplot(pair |> filter(!is.na(usd_msrp)),
       aes(x = theme, y = usd_msrp)) +
  geom_boxplot(fill = "skyblue") +
  labs(title = "Price by Theme (choose two)",
       x = "Theme", y = "Price (USD)")
```

Q6. Pieces vs Price

```
lego_complete_price <- lego |>
  filter(!is.na(pieces), !is.na(usd_msrp))

ggplot(lego_complete_price, aes(x = pieces, y = usd_msrp)) +
  geom_point(alpha = 0.5) +
  geom_smooth(method = "lm", se = TRUE) +
  labs(title = "Pieces vs. Price",
       x = "Pieces", y = "Price (USD)")

cor(lego_complete_price$pieces, lego_complete_price$usd_msrp)
```

Q7. Trends Over Time

```
yearly <- lego |>
  group_by(year) |>
  summarise(
    avg_pieces = mean(pieces, na.rm = TRUE),
    avg_price = mean(usd_msrp, na.rm = TRUE),
    avg_ppp = mean(ppp, na.rm = TRUE),
    sets = n()
  ) |>
  filter(!is.na(year))

ggplot(yearly, aes(x = year, y = avg_pieces)) +
  geom_line() + geom_point() +
  labs(title = "Average Pieces by Year",
       x = "Year", y = "Average pieces")

ggplot(yearly, aes(x = year, y = avg_price)) +
  geom_line() + geom_point() +
  labs(title = "Average Price by Year",
       x = "Year", y = "Average USD")

ggplot(yearly, aes(x = year, y = avg_ppp)) +
  geom_line() + geom_point() +
  labs(title = "Average Price per Piece (PPP) by Year",
       x = "Year", y = "Average USD per piece")
```

Themed Lines Over Time

```
yearly <- lego |>
  filter(theme %in% c("Star Wars", "Harry Potter", "Disney")) |>
  group_by(year, theme) |>
  summarise(
```

```r
    avg_pieces = mean(pieces, na.rm = TRUE),
    avg_price = mean(usd_msrp, na.rm = TRUE),
    avg_ppp = mean(ppp, na.rm = TRUE),
    sets = n()
  ) |>
  filter(!is.na(year))

ggplot(yearly, aes(x = year, y = avg_pieces, colour = theme)) +
  geom_line() + geom_point() +
  labs(title = "Average Pieces by Year",
       x = "Year", y = "Average pieces")

ggplot(yearly, aes(x = year, y = avg_price, colour = theme)) +
  geom_line() + geom_point() +
  labs(title = "Average Price by Year",
       x = "Year", y = "Average USD")

ggplot(yearly, aes(x = year, y = avg_ppp, colour = theme)) +
  geom_line() + geom_point() +
  labs(title = "Average Price per Piece (PPP) by Year",
       x = "Year", y = "Average USD per piece")
```