

Part A - Lego Analysis

Analysis of Lego

```
# Loading libraries
library(tidyverse)
library(tidyuesdayR)
library(ggplot2)
library(scales)
library(kableExtra)
```

Reading in dataset directly from tidyuesday

```
# Read data from tidyuesday
tuesdata = tidyuesdayR::tt_load('2022-09-06')

for (name in names(tuesdata))
{var_name = gsub("\\.csv$", "", name)
  assign(var_name, tuesdata[[name]])}

# Dropping data not relevant for analysis
rm(tuesdata, inventory_sets,
  elements, part_categories,
  part_relationships, minifigs,
  inventory_minifigs, name, var_name)
```

Considering the popularity of Lego, which is the most popular and the least popular themes of Lego and how many parts did they have median

Methodology:

Many sets did not have a number of parts and were either 0 or NA, which affected the data

analysis and data manipulation. What we've done is filtered only the ones that we know of how many parts as if they are not documented then there is no point in analysing it.

```
# Joining sets with themes and filtering out the sets with no parts due to error in data col.
set_themes = sets |>
  filter(!is.na(num_parts) & num_parts > 0) |>
  left_join(themes |>
    select(id, theme_name = name),
    by = c(theme_id = "id"))
```

Surface level analysis by table view

```
# Table with theme name ordered by desc of sets and their median parts
theme_name_summary = set_themes |>
  group_by(theme_name) |>
  summarize(n_sets = n(),
    median_parts = median(num_parts)) |>
  arrange(desc(n_sets)) |>
  filter(n_sets >= 100)

theme_summary = knitr::kable(theme_name_summary, caption = "Theme ordered by sets with their
```

Could even do a correlation analysis to check whether it is true that the more sets produced the lesser parts there are

```
# Creating correlation dataframe
theme_correlation = set_themes |>
  group_by(theme_name) |>
  summarize(n_sets = n(),
    median_parts = median(num_parts, na.rm = TRUE),
    .groups = "drop") |>
  filter(!is.na(median_parts))

# Calculate correlation
correlation_result = cor(theme_correlation$n_sets,
  theme_correlation$median_parts,
  method = "pearson")

correlation_result
```

```
[1] -0.08667486
```

```

# Finding out the most popular themes
most_popular = set_themes |>
  group_by(theme_name) |>
  summarize(total_sets = n(),
            median_parts = round(median(num_parts, na.rm = TRUE)),
            .groups = "drop") |>
  arrange(desc(total_sets)) |>
  head(20)

# Plotting the most popular themes
most_popular_plot =
  ggplot(most_popular,
        aes(x = total_sets,
            y = reorder(theme_name, total_sets))) +
  geom_bar(stat = "identity", aes(fill = median_parts)) +
  geom_text(aes(label = median_parts,
                x = total_sets + 5),
            hjust = 0, size = 3.2) +
  scale_fill_gradient(low = "#00648C", high = "#00BC8C", limits = c(0, 2600)) +
  labs(fill = "Median parts",
       title = "All time most popular lego themes with median parts",
       x = "Total number of sets",
       y = "Theme") +
  theme_minimal()

most_popular_plot

```

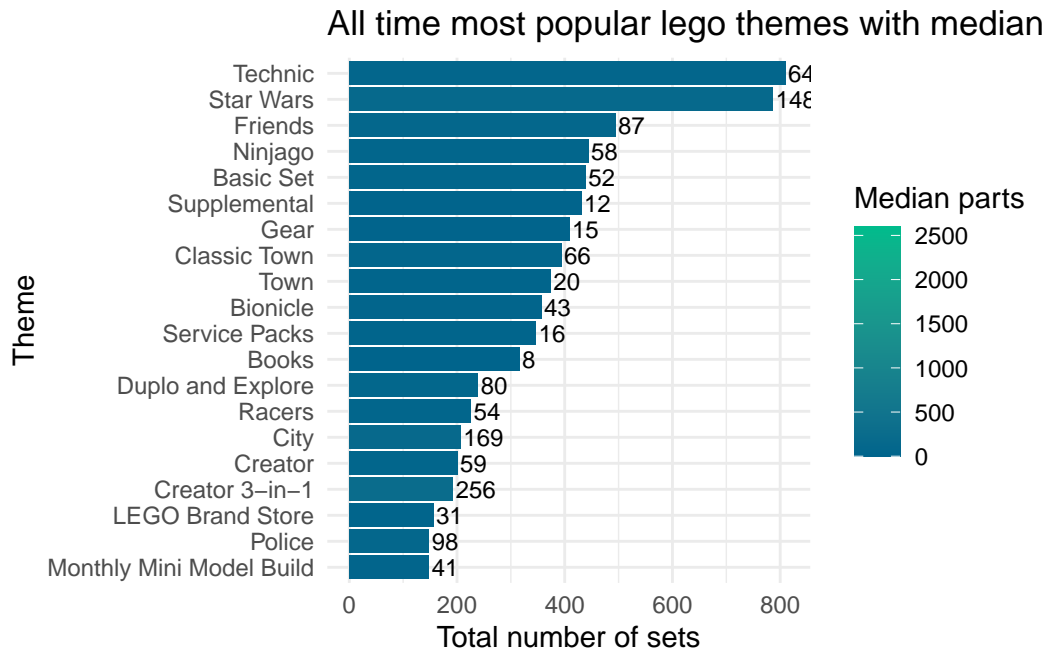


Figure 1: most_popular

```
# Finding out the least popular themes
least_popular = set_themes |>
  group_by(theme_name) |>
  summarize(total_sets = n(),
            median_parts = round(median(num_parts, na.rm = TRUE)),
            .groups = "drop") |>
  arrange(desc(total_sets)) |>
  tail(20)

# Plotting the least popular themes
least_popular_plot =
  ggplot(least_popular,
        aes(x = total_sets,
            y = reorder(theme_name, total_sets))) +
  geom_bar(stat = "identity", aes(fill = median_parts)) +
  geom_text(aes(label = median_parts),
            hjust = 0, size = 3.2) +
  scale_fill_gradient(low = "#00648C", high = "#00BC8C", limits = c(0, 2600)) +
  labs(fill = "Median parts",
       title = "All time least popular lego themes with median parts",
       x = "Total number of sets",
```

```

y = "Theme")+
xlim(0,4)+
theme_minimal()

least_popular_plot

```

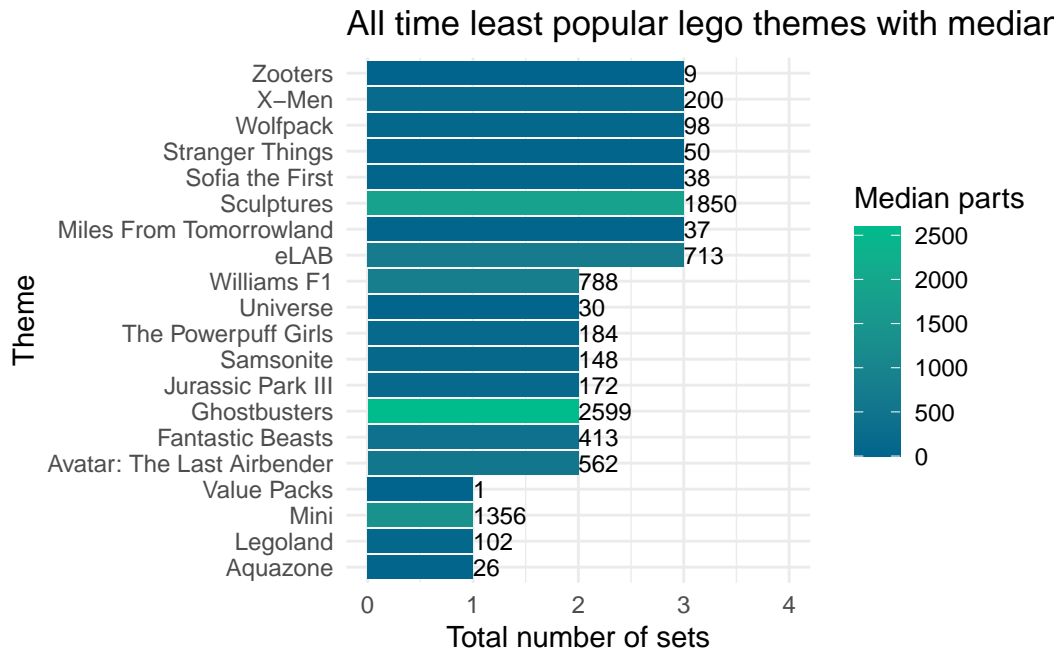


Figure 2: least_popular

Most Popular LEGO Themes

1. Gear is the most popular LEGO theme of all time, with over 2,000 sets released. This category includes accessories like keychains, stationery, and other merchandise, showing strong demand for LEGO-branded items beyond traditional building sets.
2. Star Wars comes in second with over 1,000 sets. Thanks to its powerful movie franchise, this theme appeals to both young fans and adult collectors and has remained a top performer for decades.
3. Technic also has over 1,000 sets and is popular among those who enjoy engineering-style builds. It's known for its complex mechanical features and realistic designs, attracting older children and adult hobbyists.

4. Other top themes include Key Chain, Friends, Books, and Basic Set, each with several hundred sets. This shows that LEGO successfully targets a variety of age groups and interests.

Least Popular LEGO Themes

1. Many themes, such as Zooters, X-Men, Wolfpack, and Stranger Things, have only 1 to 3 sets. These were likely experimental or niche product lines that did not gain enough traction to expand.
2. Themes like Sculptures, eLAB, and Miles From Tomorrowland may have targeted specialized audiences, but limited appeal resulted in minimal set releases.
3. Some licensed themes, such as The Powerpuff Girls, Fantastic Beasts, and Avatar: The Last Airbender, may have faced market or licensing challenges, preventing long-term success.

Conclusion

The most successful LEGO themes share common features: strong brand identity, wide audience appeal, and ongoing product development. The least popular themes were often short-lived, targeted at small audiences, or limited by licensing. Understanding these patterns helps LEGO and similar brands plan better product strategies and maximize market impact.

Seeing the trends of Lego sets over the years, how complex have the Lego sets become.

```
# Creating dataframe for part_year
part_year = set_themes |>
  group_by(year) |>
  summarise(median_parts = median(num_parts, na.rm = TRUE), .groups = "drop")

# Plotting the Line chart of median parts per set over the years
part_year_plot = ggplot(part_year, aes(x = year, y = median_parts)) +
  geom_area(fill = "#00BC8C", alpha = 0.2) +
  geom_line(color = "#00BC8C", linewidth = 1) +
  geom_point(color = "#00648C") +
  geom_smooth(method = "loess", se = FALSE, color = "red", linetype = "dashed") +
  labs(title = "Median number of parts over the years",
       x = "Year",
       y = "Median parts per set")+
  theme_minimal()

part_year_plot
```

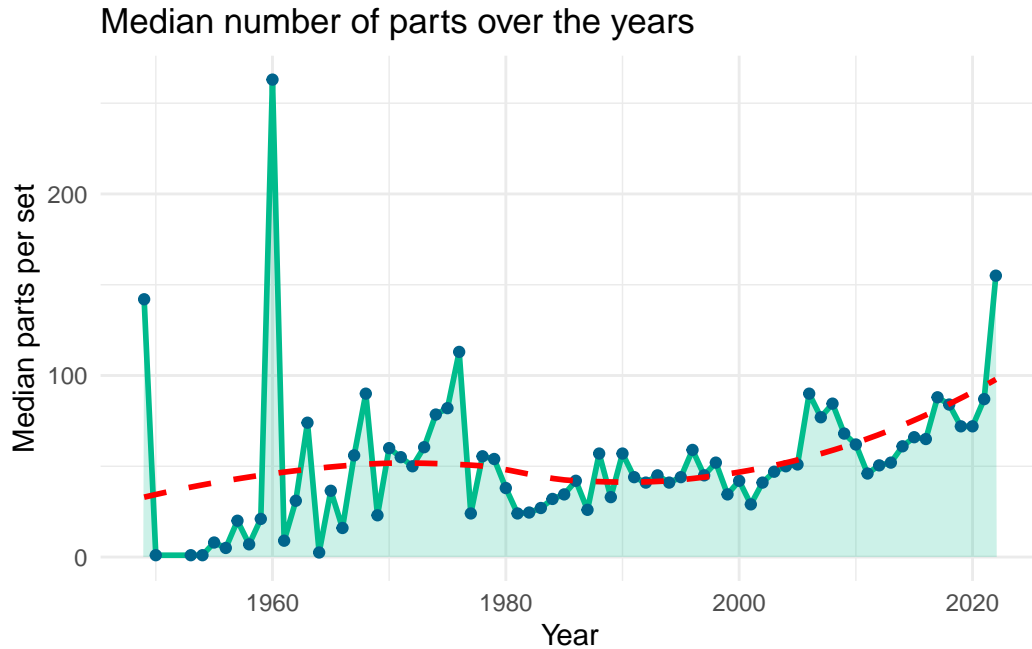


Figure 3: parts_trend

Analysis: Complexity of LEGO Sets Over the Years

1. Early Years (1950s – early 1960s):

The chart shows a large spike in the 1950s, likely due to an outlier set with many parts. Aside from this, most sets had a low median part count, suggesting simple designs.

2. Growth Phase (1960s – late 1970s):

There was a clear upward trend in the median number of parts per set. This indicates that LEGO sets became more complex over time, possibly due to increased design innovation and consumer interest in more detailed builds.

3. Stabilization Period (1980s – 2020s):

After peaking in the late 1970s, the median number of parts stabilized around 40–60 parts per set. Despite some fluctuations, no significant long-term increase is observed.

4. Recent Years (2000s – 2020s):

Complexity remained steady, with occasional dips and rises. This suggests LEGO has balanced its product line with both simple and advanced sets.

Conclusion

LEGO sets became more complex up to the late 1970s. Since then, the median complexity has plateaued, likely due to diversification of set sizes to meet the needs of different age groups and skill levels.

Discuss about the diversity of colors for parts along with complexity

```
# Creating dataframe for yearly_summary
yearly_summary = sets |>
  select(set_num, year, theme_id) |>
  inner_join(inventories, by = "set_num") |>
  inner_join(inventory_parts, by = c("id" = "inventory_id")) |>
  filter(color_id >= 0) |>
  distinct(set_num, year, color_id, theme_id) |>
  group_by(year) |>
  summarize(n_sets = n_distinct(set_num),
            unique_colors = n_distinct(color_id),
            n_themes = n_distinct(theme_id),
            .groups = "drop") |>
  filter(!is.na(year))

# Plot unique colors over years
yearly_summary_plot = ggplot(yearly_summary, aes(x = year, y = unique_colors)) +
  geom_area(fill = "#00BC8C", alpha = 0.2) +
  geom_line(color = "#00BC8C", linewidth = 1) +
  geom_point(color = "#00648C") +
  geom_smooth(method = "loess", se = FALSE, color = "red", linetype = "dashed") +
  labs(title = "Number of unique colors used in lego sets over the years",
       x = "Year",
       y = "Unique Colors") +
  ylim(0, 125) +
  theme_minimal()

yearly_summary_plot
```

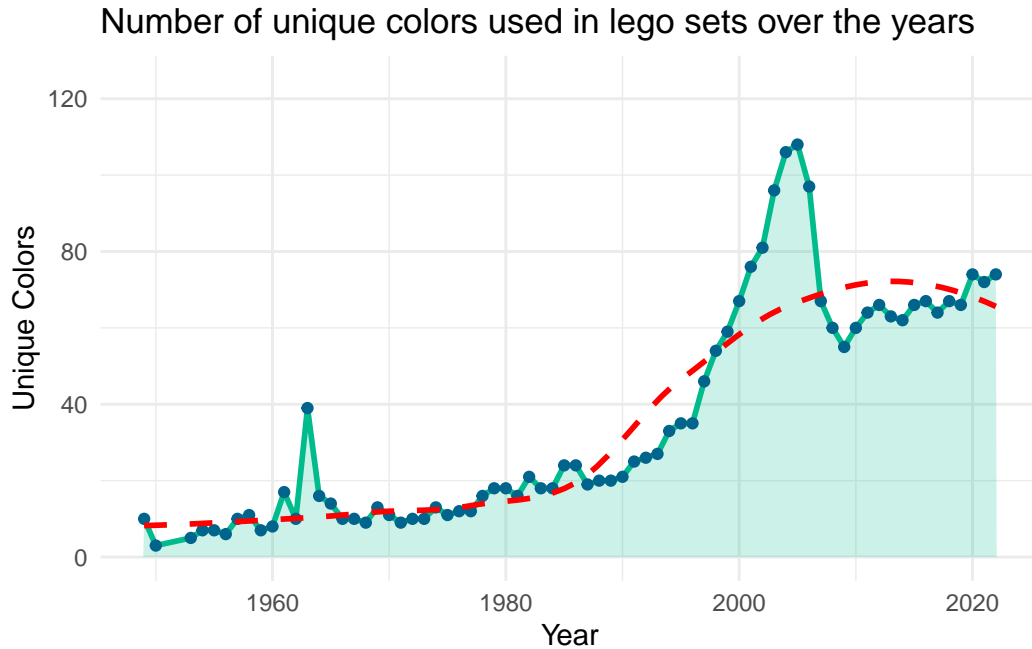



Figure 4: color-diversity

Analysis: Diversity of Colors and Set Complexity in LEGO Over the Years

1. Color Diversity Has Increased Significantly Over Time:

In the early years (1950s–1980s), the number of unique colors remained low and stable, generally below 30. A sharp increase in color variety began around the mid-1990s, peaking around 2005 with over 100 unique colors. This rise reflects LEGO’s expansion into more diverse and creative themes, such as Star Wars, Friends, and Ninjago.

2. Modern LEGO Sets Offer a Wider Visual Palette:

Since 2000, the trend has stabilized but remains at a much higher level of color diversity than earlier decades. The sustained use of 60–80 unique colors in recent years suggests a commitment to more visually rich and customizable sets, appealing to both children and adult collectors.

3. Color Complexity Supports Build Complexity :

The increase in median parts per set (as seen in earlier graphs) correlates with increased color variety. More colors allow for greater detail, theme accuracy, and differentiation of components, which enhances both aesthetic and functional complexity.

4. Design and Brand Evolution

The introduction of licensed sets (e.g., Harry Potter, Marvel) demands specific color palettes, further driving color diversity. Customization and modular building trends also push LEGO to expand its available color options for builders.

Conclusion

The growing number of unique colors used in LEGO sets is a strong indicator of increased design sophistication and build complexity. Color diversity not only enhances the visual experience, but also plays a critical role in enabling LEGO's modern design capabilities and supporting more engaging, realistic, and intricate builds.

Which sets have the most parts and the proportion of colours

```
# Creating a hex column to combine # with the hex rgb values in colors$rgb
colors = colors|>
  mutate(hex = paste0("#", rgb))

# Summarising parts by set and colour
set_parts_colors = inventory_parts |>
  inner_join(inventories, by = c("inventory_id" = "id")) |>
  group_by(set_num, color_id) |>
  summarise(parts_color = sum(quantity, na.rm = TRUE), .groups = "drop")

# Finding the total parts per set
set_total_parts = set_parts_colors |>
  group_by(set_num) |>
  summarise(total_parts = sum(parts_color), .groups = "drop")

# Joining the dataframes together
set_parts_colors = set_parts_colors |>
  inner_join(set_total_parts, by = "set_num") |>
  inner_join(sets |> select(set_num, name), by = "set_num")

# Calculating proportion of color for each set
set_parts_colors = set_parts_colors |>
  mutate(proportion = parts_color/total_parts)

# Retrieving the top 20 sets by total parts
top_20_sets = set_total_parts |>
  arrange(desc(total_parts)) |>
  slice_head(n = 20) |>
```

```

pull(set_num)

# Joining with the top 20 sets with hex color
top_20_data = set_parts_colors |>
  filter(set_num %in% top_20_sets) |>
  left_join(colors |> select(id, hex), by = c("color_id" = "id"))

# Dropping unused datasets to improve performance
rm(top_20_sets)

# Plotting the top 20 sets with the most piece and filled with proportion of colours
top_20_plot = ggplot(top_20_data,
                     aes(x = reorder(name, total_parts),
                         y = parts_color, fill = hex)) +
  geom_bar(stat = "identity",
           color = "black",
           linewidth = 0.1) +
  scale_fill_identity() +
  labs(title = "Top 20 lego sets by # parts and colour proportion",
       x = "Set Name",
       y = "Number of Parts",
       fill = "Color") +
  coord_flip() +
  theme_minimal()

top_20_plot

```

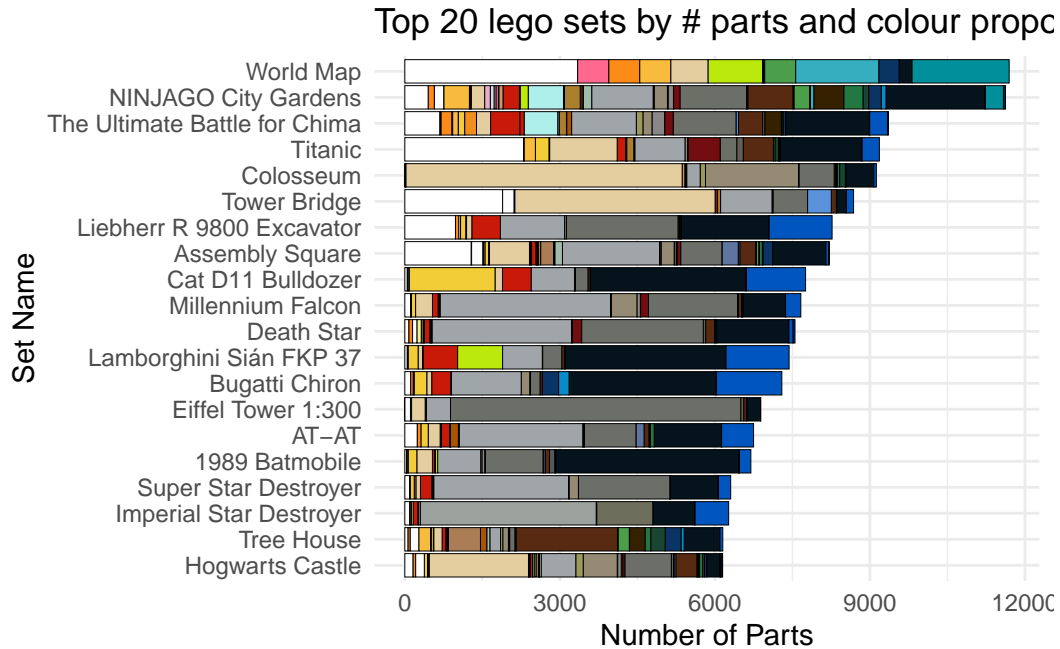


Figure 5: most_pieces

```
# <--Remove # to clean all previous selections
# rm(colors, least_popular, most_popular, part_year, parts, yearly_summary, theme_name_summar
```

Analysis: LEGO Sets with the Most Parts and Color Proportions

1. Sets with the Highest Number of Parts:

World Map is the LEGO set with the most parts, exceeding 11,000 pieces. Its design focuses heavily on tile placement and color detail. NINJAGO City Gardens and The Ultimate Battle for Chima also rank highly, each containing around 5,500 to 6,000 pieces, showcasing advanced structural and decorative elements. Other large sets include Titanic, Colosseum, and Tower Bridge, each with over 9,000 pieces, reflecting highly detailed architectural builds.

2. Color Proportions and Variety:

Sets like World Map and Tree House show high color diversity, using dozens of different shades to depict natural or geographic details. In contrast, Liebherr R 9800 Excavator, Bugatti Chiron, and Lamborghini Sián use a more concentrated color palette, focused on realism and technical detail (e.g., grey, black, yellow, blue). Star Wars sets like the Millennium Falcon, Death Star, and Imperial Star Destroyer are dominated by grey tones, consistent with their on-screen designs.

3. Function vs. Aesthetic Balance:

Some sets prioritize aesthetic variety (e.g., World Map, Tree House) and thus have many colors but simpler build structures. Others focus on mechanical or architectural complexity (e.g., AT-AT, Excavator), using fewer colors but highly technical builds with intricate mechanisms.

Conclusion

The number of parts in a LEGO set often correlates with either extensive design detail (e.g., maps, buildings) or technical function (e.g., vehicles, machinery). Color proportions reveal the thematic purpose of each set—some aim for vibrant storytelling, while others reflect precision and realism. Together, these sets showcase LEGO’s versatility across both form and function.