A Time Mapping of LEGO Bricks: Analysis and Evolution of LEGO Sets Over the Years

A semester project for Introduction to Informatics INFO I501, Spring 2018

Disha Bora, Carl Klutzke, Matthew Brown, Pushkar Joshi, Sai Pramod Yerubandi, Siddhant Bhandari, Revathi Gontumukkala, Utkarsha Bhamare

School of Informatics and Computing,
Indiana University Purdue University Indianapolis,
Indianapolis, IN, USA
{dishbora, cklutzke, mb2, pujoshi, saipyeru, sidbhand, regontu, ubhamare}@iu.edu

Abstract. LEGO bricks are fun and popular with all age groups. Over the years sets have grown more complex with a greater number of pieces, different themes and a variety of colors available for players in the market. An analytical study with help of visualizations of this complex dataset provides players and data analysts worldwide a way to understand the trends in LEGOs. Our aim with this analytical research study is to analyze the sets produced each year, particularly the complexity and variations of colors and themes. Our results show that the number of sets has exponentially increased since the 1950s, with dramatic variations in colors and exponential growth in themes. We predicted the number of sets in coming years using linear regression. Further we plotted colors based on ten different themes and identified the unique colors overall.

Keywords: LEGO · Bricks · Python

1. Introduction

The LEGO company was founded in 1932 by Ole Kirk Kristiansen, and in 1958 the company launched and began producing the ubiquitous LEGO brick that we are familiar with today (Mortensen, 2017).[5] It has become a popular toy brand where the bricks and are often sold in sets designed to build a specific object. Each set contains a number of parts in different shapes, sizes and colors. In this project we explore a dataset that has collected various characteristics—parts, sets, and colors—of the LEGO sets that were produced from 1950 to present. The dataset for our study was obtained from Rebrickable.com. In general this project explored the evolution of the

LEGO sets from 1950 to present, noting trends in set complexity, number of pieces, and color.

This is an educational project to extract and analyze data from a relational database. The LEGO database was selected because it has a rich set of data—with over 11,000 sets dating back to 1950—and because LEGOs are fun! This year the world is celebrating the 60th anniversary of LEGO bricks, hence through this project we are trying to study and explore these amazing fun bricks and analyze the trends over the years. Analyzing the trend in these colors, identifying the popular themes and number of sets is cumbersome with the rich databases available online. Visualizations with LEGO sets have been attempted in other studies to contextualize the available data.[3] Our study's contribution is twofold, first we analyze the trend to prove our hypothesis that LEGO sets have become more complex and second we provide easy to understand visualizations for data analysts and LEGO players to assist them understand the trend in colors and themes. The complexity and evolution of over the years is demonstrated over time in this project.

1.1 Related Work

LEGOs are a fun that are familiar to people around the world, and as such they have inspired several quantitative research projects interested in charting their complexity and the possible combinations. Moltchanova demonstrated how Lego sets have increased in complexity since 1955.[1] Recent research has already shown that Lego sets have become larger and now contain more diverse bricks in both shape and color (Moltchanova, 2018). This is similar to our research. However, Moltchanova's data was obtained without consent by scrubbing Bricklink.com, which led to Bricklink.com blocking Moltchanova computer's IP address. Since Moltchanova's data source was cut off it is unclear if it is complete, or even how accurate it is. Our data was obtained from an open source database from Rebrickable.com, and considered to be more reliable.

Beyond looking at complexity, Durhuus and Eilers explored the number of combinations that can be made with a group of LEGO bricks.[2] With aid of a complex computer algorithm it was shown that the number of combinations that can be made with six 2x4 bricks is 915,103,765 (Eilers, 2005). The computer running the algorithm took a week to compute that number. As you add more bricks the complexity of the calculations grows exponentially. Eilers estimated that it would take 1.31e41 years to compute the number of combinations using 25 bricks. It would be interesting to compare complexity between LEGO sets by calculating the number of possible

combinations that can be made, however due to limitations to computing power it is not feasible to calculate this for all sets in the LEGO database.

In a study on the evolution of LEGO over the past 67 years, Joel Carron analyzed the number of pieces in a set and found out that the number of pieces have been increasing over the years.[4] He also studied the color palette over each decade and found that LEGOs have gotten darker, with white giving way to black and gray.

Lego's color palette has expanded over the decades. Until the 1990s, almost every piece was one of the top ten colors; now only about 80% are. In a similar study Nathanael Aff analyzed LEGO color themes with topic models.[6] He came up with a number of visualizations from the dataset acquired form Rebrickable.com, such as unique LEGO colour appearances by year, relative frequency of brick colours per year, and so on.

In another interesting research project, Bartneck et al. investigate the development of the facial expression for all LEGO minifigures that were released between 1975 and 2010.[8] The results show that the LEGO company started in 1989 to dramatically increase the variety of facial expressions. The two most frequent expressions are happiness and anger, and the proportion of happy faces is decreasing over time. Through a k-cluster analysis they identified six types of facial expression: disdain, confidence, concern, fear, happiness, and anger. In addition they also tested if the perception of the face changes when the face is presented in the context of a complete minifigure. The impression of anger, disgust, sadness and surprise were significantly influenced by the presence of context information.

2. Methodology

Working with a group of people can present many challenges, stemming from communication and varying levels of experience. By establishing a clear list of goals, a corresponding timeline and dividing the responsibilities between our team of eight (Table 1), we were able to avoid miscommunication at different stages of the project. We met periodically over the course of project to discuss updates, issues, brainstorm, and meet the goals of this project. Every team member was encouraged to provide suggestions and seek guidance in the face of doubt to the team. Amongst the members, listening carefully and thanking each other was key to maintaining progress and eliminating any conflict.

Table 1: Team member responsibilities.

Members	Responsibility
Pushkar and Carl	Create MySQL database, import dataset from CSV files, and clean data as required.
Utkarsha, Revathi, Sai, and Matthew	Create visualizations using Python to display data in a manner analyze and provide an answer the research question.
Disha and Siddhant	Compile methods, analysis, and results into a final report.
Matthew	Create a slide deck that will be used to present the methods, analysis, and results of the research to the class.

Members were not limited to these responsibilities, and each contributed to all aspects of the project.

After coming to a common consensus and finalizing our topic and research goal, the first step was to build a relational database of stored data downloaded and extracted from Rebrickable.com's CSV files, retrieved from https://rebrickable.com/downloads/. Our intention was to use a database provided by the school, like those accessible from our student accounts, but accessible to our entire group. As that was not available, we investigated and set up a free MySQL database with AWS EC2. After setting up the database we wrote the SQL queries to retrieve information and answer the research questions and hypothesis. Four team members collaborated to write Python scripts that executed the SQL queries and displayed the results graphically. We used different prediction curves and checked which one best fits our data exponentially to analyze further results.

2.1 Data Storage

In order to make the data for our project available to the whole team instead of under one student's account, we set up a free MySQL database with Amazon Web Services Elastic Cloud 2. We setup SSH with the EC2 instance to access the remote server with private keys. In the EC2 server, we got access to a clean version of Linux OS. On Linux we installed php version 5.6.6, as it was essential going forward for phpMyAdmin. To run php we installed Apache web server on the EC2 instance. The EC2 instance came with a default installation of MySQL server, so we started the

service using the command line. SQL service was not accessible to everyone so we installed phpMyAdmin to access the database with its UI.

We created a database called Lego_db and loaded CSV files of LEGO data from Rebrickable.com into the database, and provided the database credentials to the team so it could be accessed from our Python programs.

2.2 Data Description, Extraction, Analysis, and Modeling

The database schema provided by Rebrickable.com is as follows:

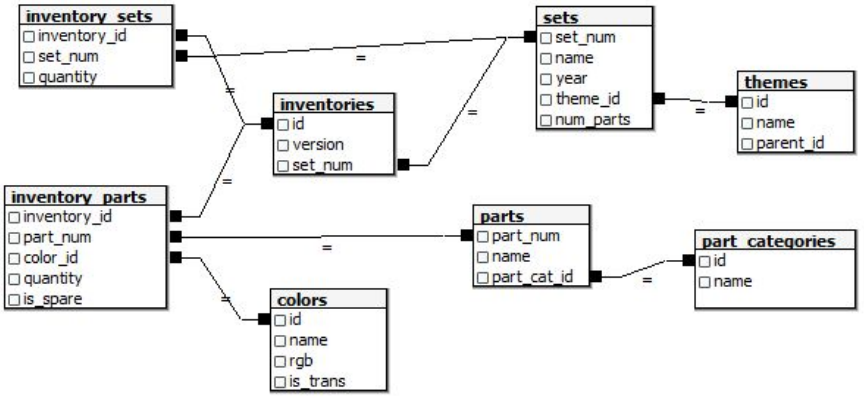

Fig. 1. The tables of the Rebrickable.com database shown in relationship to each other.

Table 2. Rebrickable.com CSV table file descriptions.

Table Name	Description	Size
colors	Colors, including a unique ID for each color, its name, an approximate RGB value, and whether it's transparent.	3.5KB
inventories	Inventories, including a unique ID, its version, and the set number.	179.1KB
inventory_parts	Part inventories, including a unique ID number, the part number, the color of the part, how many are included, and whether it's a spare.	10.8MB

inventory_sets	What inventory is included in which sets, including the inventory ID, the set number, and the quantity of that inventory that are included.	43.0KB
parts	LEGO parts, including a unique ID number, the name of the part, and what part category it's from.	1.8MB
part_categories	The part category (what type of part it is) and a unique ID for that part category.	1.2KB
part_relationships	(Not pictured in schema.) The subparts of which some parts are composed.	294.7KB
sets	LEGO sets, including a unique ID number, the name of the set, the year it was released, its theme, and how many parts it includes.	518.1KB
themes	LEGO themes. Each theme is given a unique ID number, a name, and (if it's part of a bigger theme) which theme it's part of.	11.2KB

We had some concerns about the size of the database, and it didn't all load in one go. So we uploaded our data selectively. We had shared phpMyAdmin access given by the professor, but the upload limit of the database was 8MB per file, and the CSV files that we were uploading were 10MB. What we ended up doing is we divided the CSV files and cut them into different parts of 3 MB. Every file had 4-5 thousand rows and we uploaded all the data to the table by importing one file at a time.

2.3 Program Code

We wrote SQL queries to extract data from the database. The used Python scripts that executed those queries and displayed the results graphically. The scripts were executed in Jupyter Notebook against our MySQL server database. Samples of the queries and scripts are included in the appendices.

Exploratory analysis was performed to understand the database and to gain further insights into the data. After understanding the database and having team consensus on our approach, we used the resulting insights to generate our models and graphs.

Our Python scripts analyzed the LEGO data according to a variety of different measures:

- Number of sets
- Number of parts per set
- Themes
- Colors

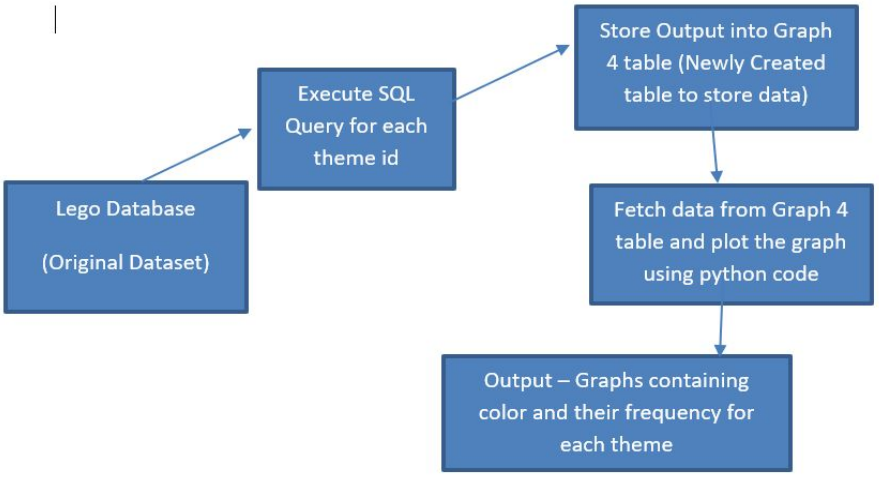

Fig 2: A sample block diagram for generating a graph.

3. Results

We generated a number of graphs to support our hypothesis that LEGO sets have become more complex over time.

3.1 Number of LEGO Sets

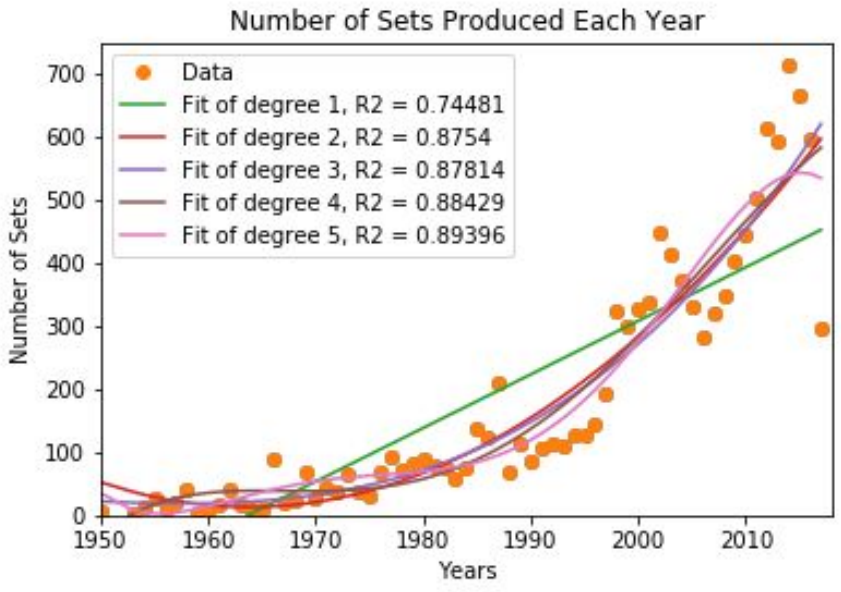

Fig. 3: Linear regression analysis of LEGO sets produced per year.

The visualization in figure 3 gave us insights about the number of LEGO sets that have been produced since 1950. The least number of sets was 3, produced in 1960 when the product line was still young. The greatest number of sets was 713, produced in 2014. It is assumed that LEGO ramped up production that year to coincide with the release of The LEGO Movie in 2014.[7] Linear regression against this data shows that the set production data most closely follows a 4th degree polynomial line (brown).

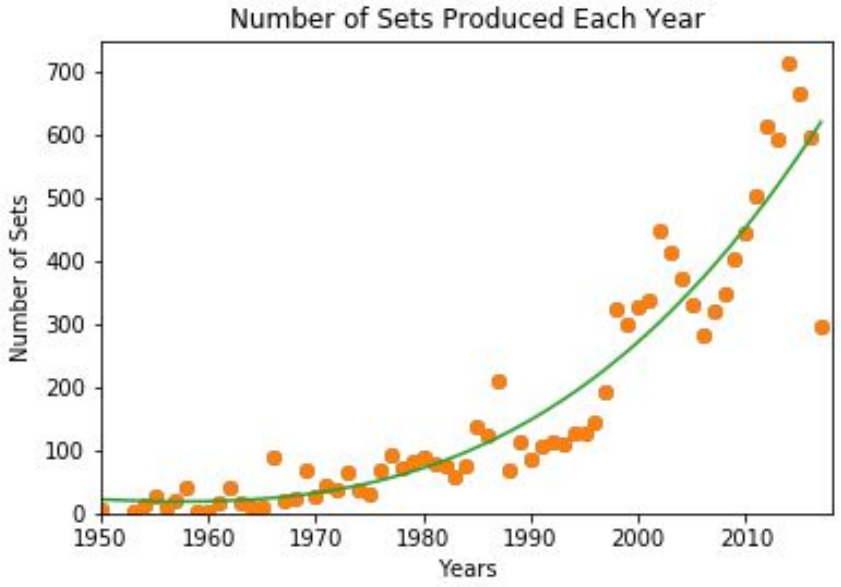

Fig 4: Prediction for number of sets to be produced in 2018.

If LEGO continues producing sets according to our 4th degree polynomial fit line, they will produce 596 sets in 2018. However, this prediction is unlikely to be accurate. Set production declined in 2015 and 2016, and our data for 2017 is incomplete. In addition, corporations are not likely to use polynomial fit lines in this manner to decide their product release strategy.

3.2 Parts per Set

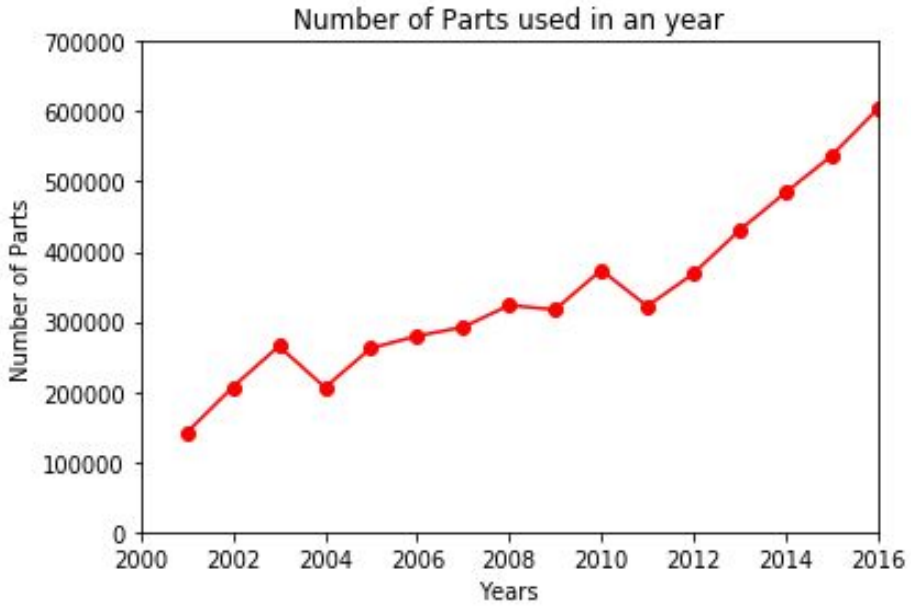

Fig. 5: Total inventory of LEGO parts released in sets since 2000.

The number of parts used in LEGO sets has also been steadily increasing. Since 2000 there has been substantial growth that coincided with the revision of the company mission. A complete inventory of all sets released in 2016 would include almost 600,000 parts.

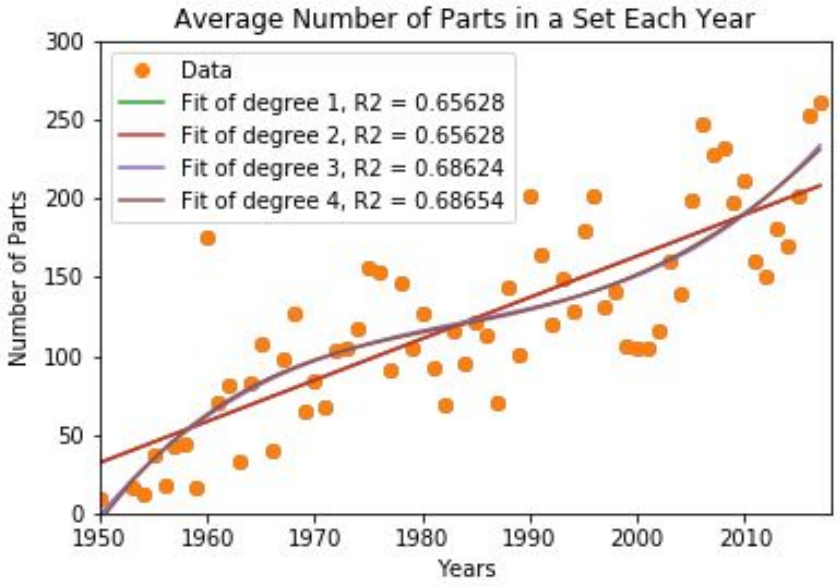

Fig. 6: Linear regression analysis of mean number of LEGO parts per set.

1950, early in the product history, saw an average of only 10 parts per set. The greatest average number of parts per set was established at 260 in 2017. The single largest LEGO set was the Taj Mahal Creator set, released in 2008 with over 5,900 pieces. Linear regression against this data shows that the parts-per-set data most closely follows a 3rd degree polynomial line (violet).

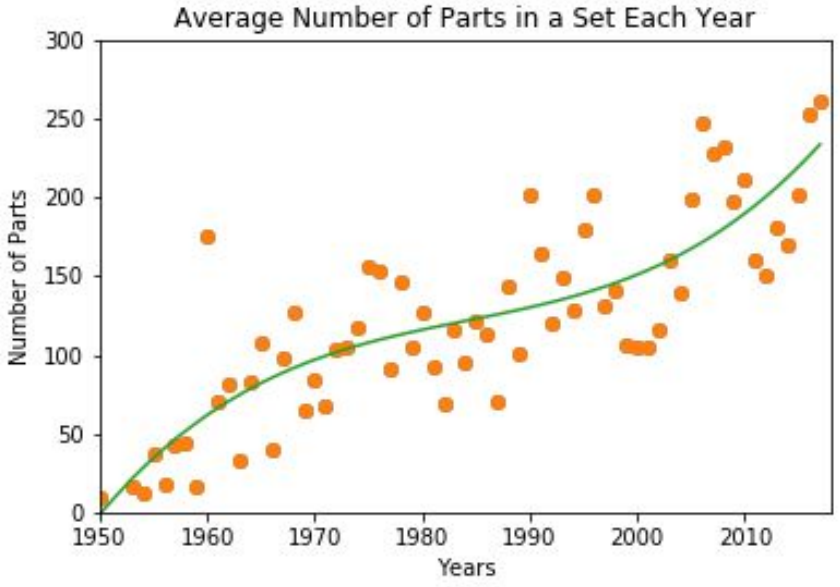

Fig. 7: Prediction for mean parts per set in 2018.

If LEGO continues designing sets according to our 3rd degree polynomial fit line, the sets released in 2018 will on average contain 241 parts.

3.3 Themes

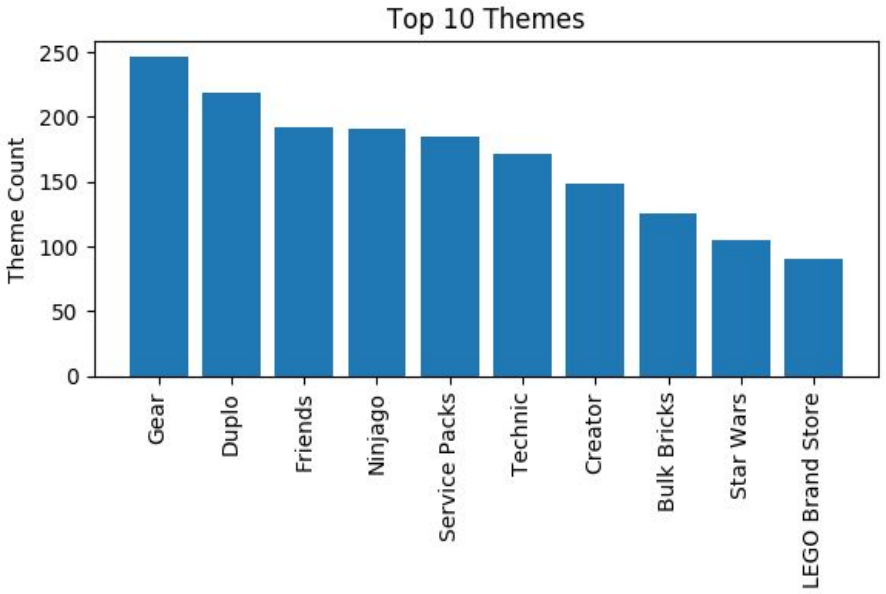

Fig. 8: The number of sets included in the 10 largest LEGO themes.

LEGO sets are grouped according to themes. Each theme shares a common set of parts and colors. Several themes use licensed intellectual properties, such as Star Wars or Harry Potter. LEGO also creates themes for intellectual properties developed internally by LEGO, such as Ninjago. And other themes support specialized LEGO-building interests, such as DUPLO (for small children) and Technic (for advanced builders). Figure 8 shows the themes with the largest number of LEGO sets.

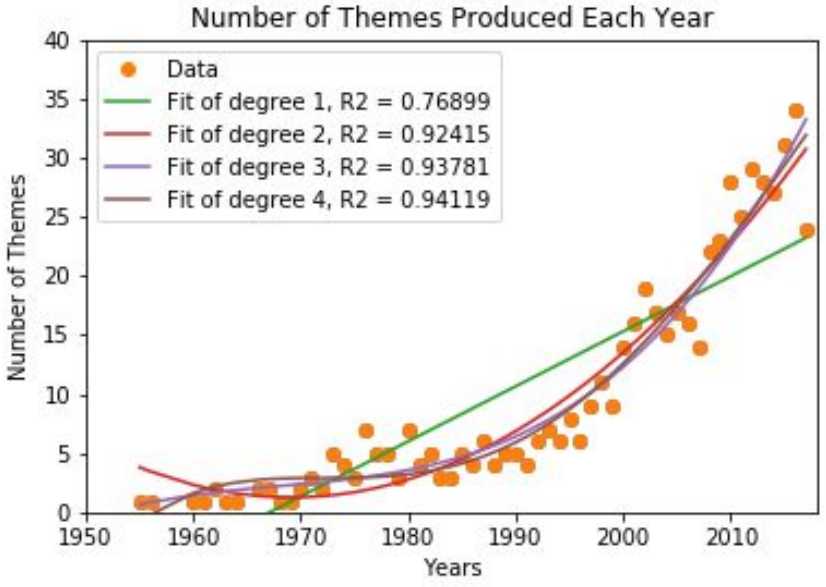

Fig. 9: Linear regression analysis of LEGO themes per year.

Prior to 1955, LEGO did not group their sets into themes, and all sets that year were produced in a single theme. In 2016 LEGO produced sets in 34 themes, the greatest number in its history. Linear regression against this data shows that the themes-per-year data most closely follows a 3rd degree polynomial line (violet).

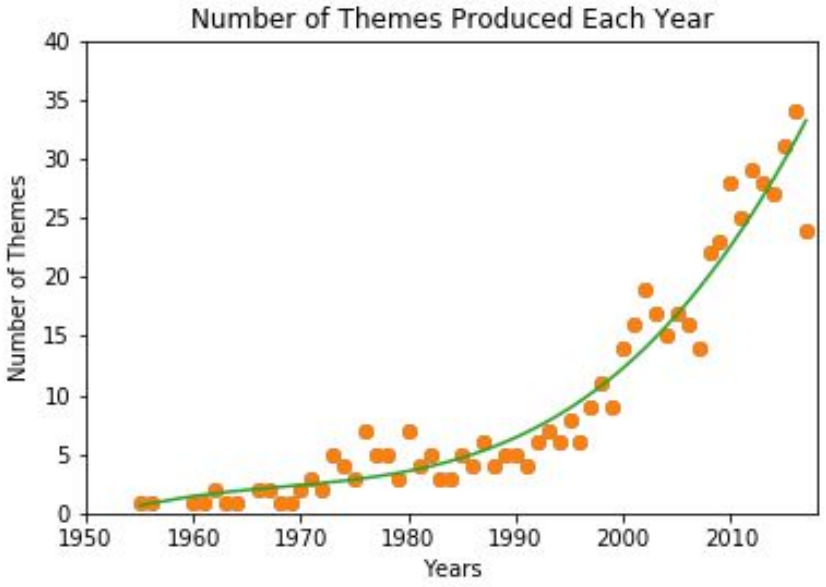

Fig. 10: Prediction for themes produced in 2018.

If LEGO continues producing sets according to our 3rd degree polynomial fit line, the sets released in 2018 will fall into 34 different themes.

3.4 Colors

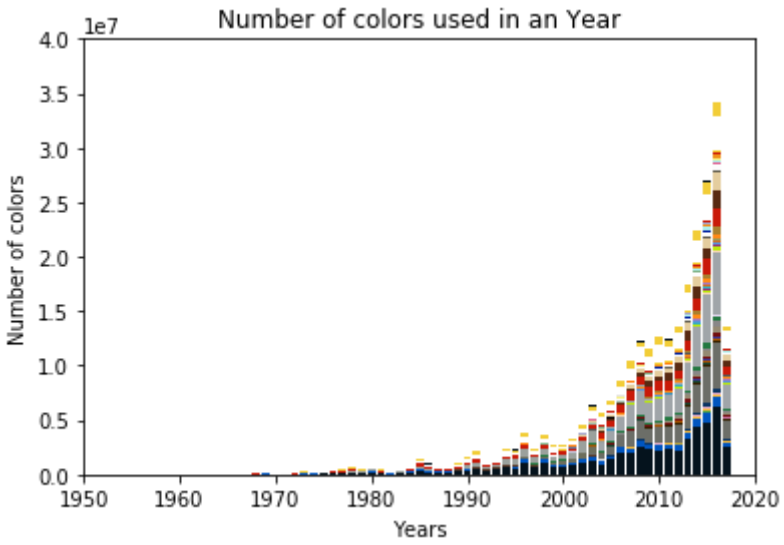

Fig. 11: Stacked bar graph of bricks per year by color.

The variety of colors used in LEGO sets has increased considerably from the six original colors:

- White
- Red
- Yellow
- Green
- Blue
- Clear

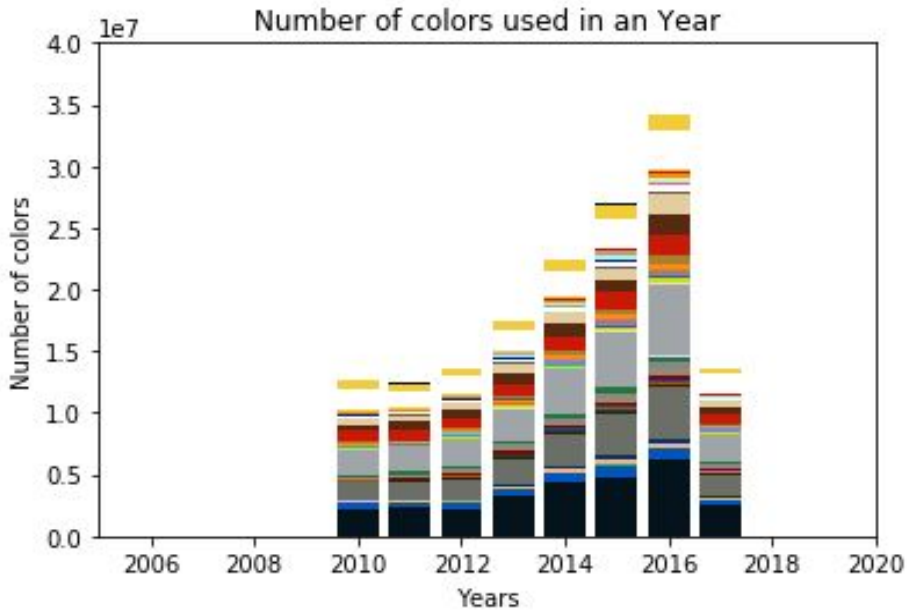

Fig. 12: Detail of stacked bar graph of bricks per year by color from 2010 onward.

Black and gray parts were introduced in the 1950, and the proportion of their use has steadily increased.

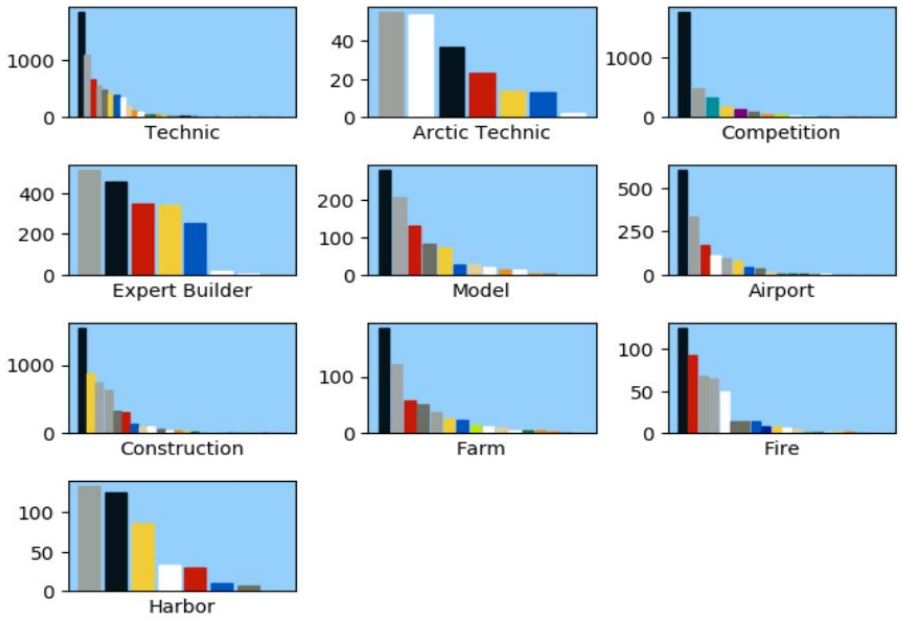

Fig. 13: Distribution of color in selected LEGO themes.

Each theme uses a distribution of colors appropriate to what it portrays. For example, the Arctic Technic theme uses a high proportion of white parts, and the Fire theme employs a high proportion of red parts. Some sets use as few as seven colors, others use as many as 23. Note that black and gray parts are strongly represented in every theme.

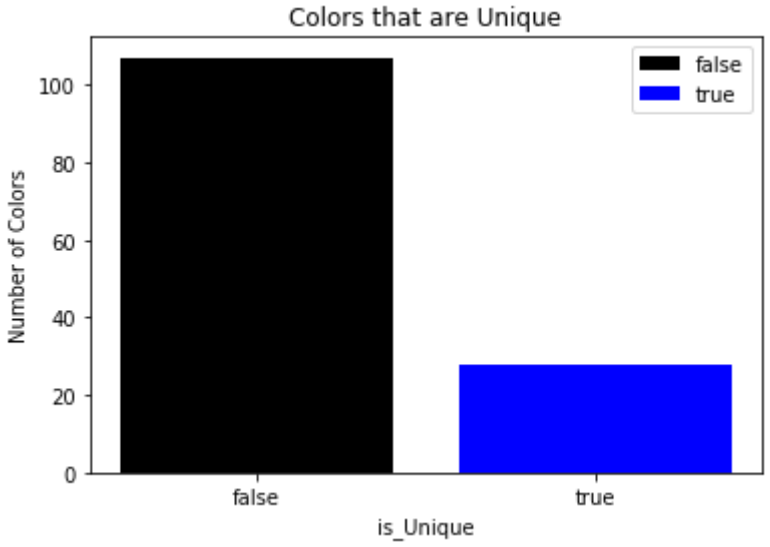

Fig. 14: Count of colors unique to a specific set.

135 distinct colors have been used in LEGO sets. 28 of those colors are unique to a specific LEGO set.

4. Discussion

When it comes to working in a group, it has its own challenges as everyone has their own style and manner of working. But by assigning clear roles and responsibilities according to each members skills and comfort level, at an earlier stage in the project helped us manage things smoothly. Initially while choosing our project we all had our own preferences, therefore to arrive at a conclusion we all proposed our topics and had vote which helped us finalise our topic. Also while deciding the hypothesis we had a lot of discussion about what it should be. Everybody contributed with their opinion which ultimately lead us to decide on a meaningful hypothesis.

By analysing the evolution of LEGO over time one thing that most of our graphs convey is that LEGO's have become far more complex in every possible way than before. The number of sets produced in a year, the number of parts per set, the variety of colours, the number of themes according to our analysis have all increased significantly over the years.

We recognize that there are limitations to our study for example our analysis did not include exploration of the commonality of bricks between sets. Commonality could be another attribute that describes complexity. Being able to demonstrate how the number of parts shared by sets has changed over time could have led to a better

understanding of how complexity changed over time. Additionally, while performing regression analysis the decisions made to use a lower degree polynomial fit line when the coefficient of determination was actually better for a higher degree polynomial was based on intuition rather than theory because there is no theory on what drives the production of LEGOS. We also experienced some limitations with the data. The data for 2017 was incomplete and by including it the results may have been skewed. Additionally the parts inventories table was very large, including approximately 580,000 rows. This proved to be very challenging when we tried to import it into the relational database.

For future work we would like to provide a more in depth analysis of the trend we noticed here by adding financial data and historical records. This would allow us to better understand the driving forces behind the change in LEGO production. This might lead to better models for predicting future LEGO production. We considered analyzing the ability to identify a LEGO theme by random draws. Given all the bricks from a theme in a box, how many draws would you need to take to reliably guess the theme?

5. Conclusion

In conclusion to our study, we performed an analytical study of LEGO sets over the years. Our results indicated that LEGOs have become more complex over time in almost every measurable way.

- The size of sets have grown exponentially
- The number of parts per set has grown exponentially
- The number of themes available has grown exponentially
- The variety of colors has increased dramatically

References

1. Bartneck C, Moltchanova E (2018) LEGO products have become more complex. PLOS ONE 13(1): e0190651. https://doi.org/10.1371/journal.pone.0190651
2. Durhuus, B., Eilers, S. : A LEGO Counting Problem, http://www.math.ku.dk/~eilers/lego.html
3. Tham, Chris. (2016, March 13). LEGO Visualizations - Combining my love for LEGO, Python and Tableau. Retrieved from https://www.linkedin.com/pulse/lego-visualisations-combining-my-love-python-tableau-chris-tham/.

4. Carron, Joel. (2016, July 21). 67 Years of Lego Sets. Retrieved from https://blog.modeanalytics.com/lego-data-analysis/.
5. Mortensen, Tine Froberg (2017, October 17). LEGO Group. LEGO History Timeline. Retrieved from https://www.lego.com/en-us/aboutus/lego-group/the_lego_history.
6. Aff, Nathanael. (2017, August). Finding Lego color themes with topic models. (https://www.kaggle.com/nateaff/finding-lego-color-themes-with-topic-models?scriptVersionId=1514787
7. Roar Rude Trangbæk (2014, September). GLOBAL LEGO® Sales up 15 Percent in First Half of 2014. Retrieved from https://www.lego.com/en-us/aboutus/news-room/2014/september/interim-result-2014
8. Bartneck C, Obaid M, Zawieska K. Agents with faces—What can we learn from LEGO Minifigures. In: 1st International Conference on Human-Agent Interaction; 2013. p. III–2–1. Available from: http://hai-conference.net/ihai2013/proceedings/pdf/III-2-1.pdf.

Appendices

A. Additional Figures

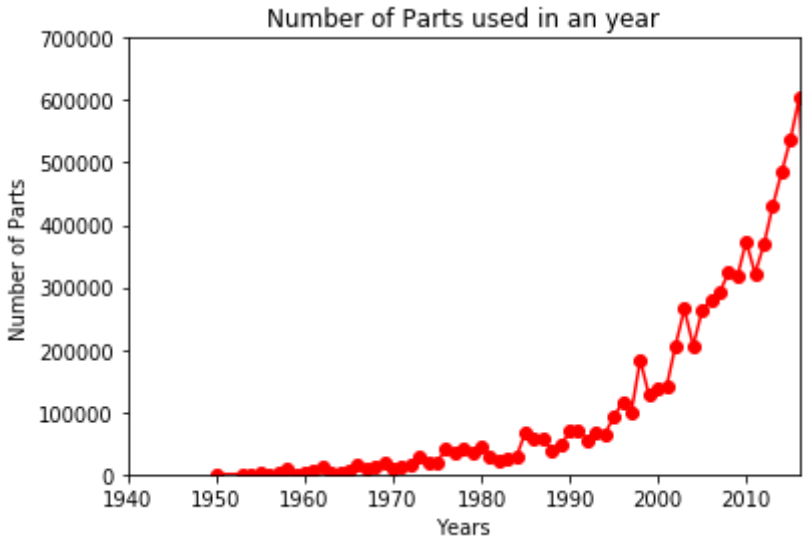

Fig. 15: Total inventory of LEGO parts released in sets.

B. Sample SQL Queries

The following queries were used to generate figure 13, showing distribution of color in selected LEGO themes.

Create a new table named Graph4 to store the extracted data:

```
CREATE TABLE `Graph4` (

  `theme_id` int(11) DEFAULT NULL,

  `theme_name` varchar(250) DEFAULT NULL,

  `color_id` int(10) DEFAULT NULL,

  `color_name` varchar(250) DEFAULT NULL,

  `rgb` varchar(250) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

We executed the following query for each theme id to store data related to that theme its colors into newly created table:

```
INSERT INTO Graph4(theme_id,theme_name,color_id,color_name,rgb)

SELECT

       t.id as theme_id,

       t.name as theme_name,

       c.id as color_id,

       c.name as color_name,

       c.rgb as rgb

FROM colors c

inner join inventory_parts ip
```

C. Sample Python Code

The following Python code was used to generate Figure 12, showing the distribution of color in LEGO sets since 2010.

```
import MySQLdb
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline

conn = MySQLdb.connect(host="localhost", user="regontu",
passwd="********", db="regontu_db")
cursor = conn.cursor()

cursor.execute("select * from Graph where year>2009")

rows =cursor.fetchall()
print(len(rows))

data=sorted(rows, key=lambda num_part: num_part[1])

num_parts_arr = []
years = []
colors = []
rgb_values = []

for element in data:
    num_parts = element[0]
    year = element[1]
    color = element[2]
    rgb_value = str('#'+ element[3])
    num_parts_arr.append(num_parts)
    years.append(year)
    colors.append(color)
    rgb_values.append(rgb_value)

index = [year for year, _ in enumerate(set(years))]

plt.axis([2005,2020,0,40000000])
plt.xscale('linear')
plt.ylabel("Number of colors")
plt.xlabel("Years",labelpad=5)
plt.title("Number of colors used in an Year")

n_rows = len(data)

bar_width = 0.8
columns = len(colors)

num_parts_offset=[]
for index in range(len(set(years))):
    num_parts_offset.append(0)

years_visited = []
```

```
for row in range(n_rows):
    if(years[row] not in years_visited):
        num_parts_offset[index] = 0
        years_visited.append(years[row])
    plt.bar(years[row], num_parts_arr[row], bar_width,
bottom=num_parts_offset[index], color=rgb_values[row])
    num_parts_offset[index] = num_parts_offset[index] +
num_parts_arr[row]

plt.show()
```