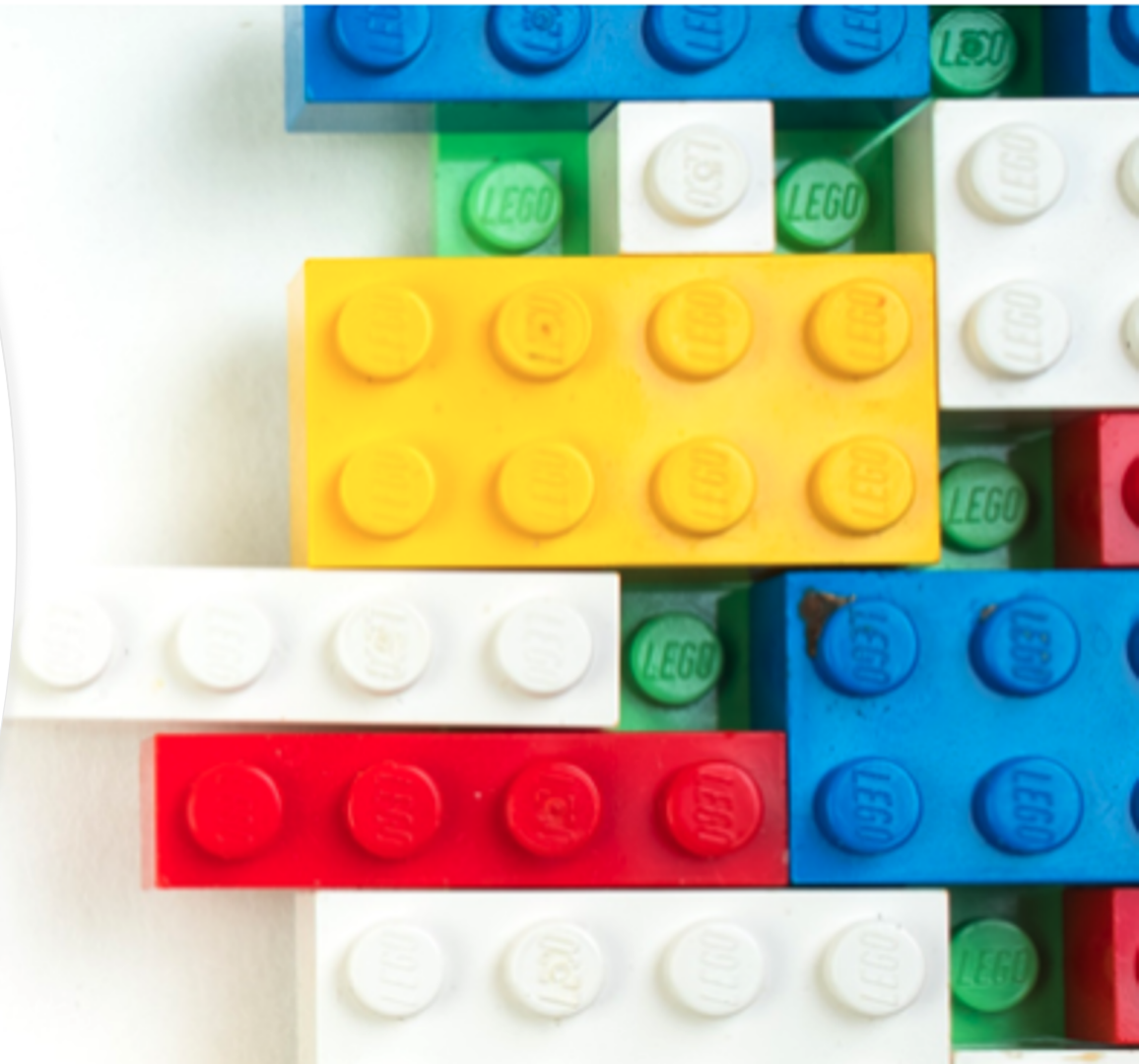
Expand Lego Army

Lego Data Analysis

Final Project for MSCA 31012

By: Lingyi Zhao, Qingyang Mu,
Yixiao Yang, Ke Deng



Meet Our Team

Lingyi (Jannie) Zhao

Yixiao (Shawn) Yang

Ke (Coco) Deng

Qingyang (Ashley) Mu

Content

- Executive Summary
- Business Use Case
- ETL & Database Design
- Visualization & Analysis
- Recommendations

Executive Summary

As one of the most popular and best-selling toys of all time, Lego has become the “world’s most powerful brand”. Over the past 70 years, Lego keeps growing to meet consumer desires and market demands.

The purpose of this project is to help Lego Company track production performance, manage inventory and develop marketing strategies with a global scope.

Business Use Case

For Lego Sellers

- How are the Lego markets like in each country?
- How to do effective inventory management?
- How to predict the prices for each Lego set?
- How to improve ratings for each Lego set?
- What are some recommended marketing strategies?

For Lego Buyers

- What is the average suggested age for lego sets?
- What are the most popular sets among Lego players?
- Which Lego sets are worth collecting?

Dataset Overview

Inventory data

Brickset: Lego data 2020

349 Themes, 184 colors, 5 version, 2632 inventory sets, 18754 child parts, 2114 parent parts, 13530 sets

Sales data

Brickset: Lego sets

31 unique ages, 12261 sales records, 21 countries, customer reviews, ratings

https://brickset.com

Data Quality Metrics

Completeness

Remove NA, unnecessary columns and group age into ranges.

Accuracy

Data from Brickset -- a professional lego data website

Consistency

Use R and openrefine to adjust the set_id inconsistency

Validity

Data value are within the acceptable range

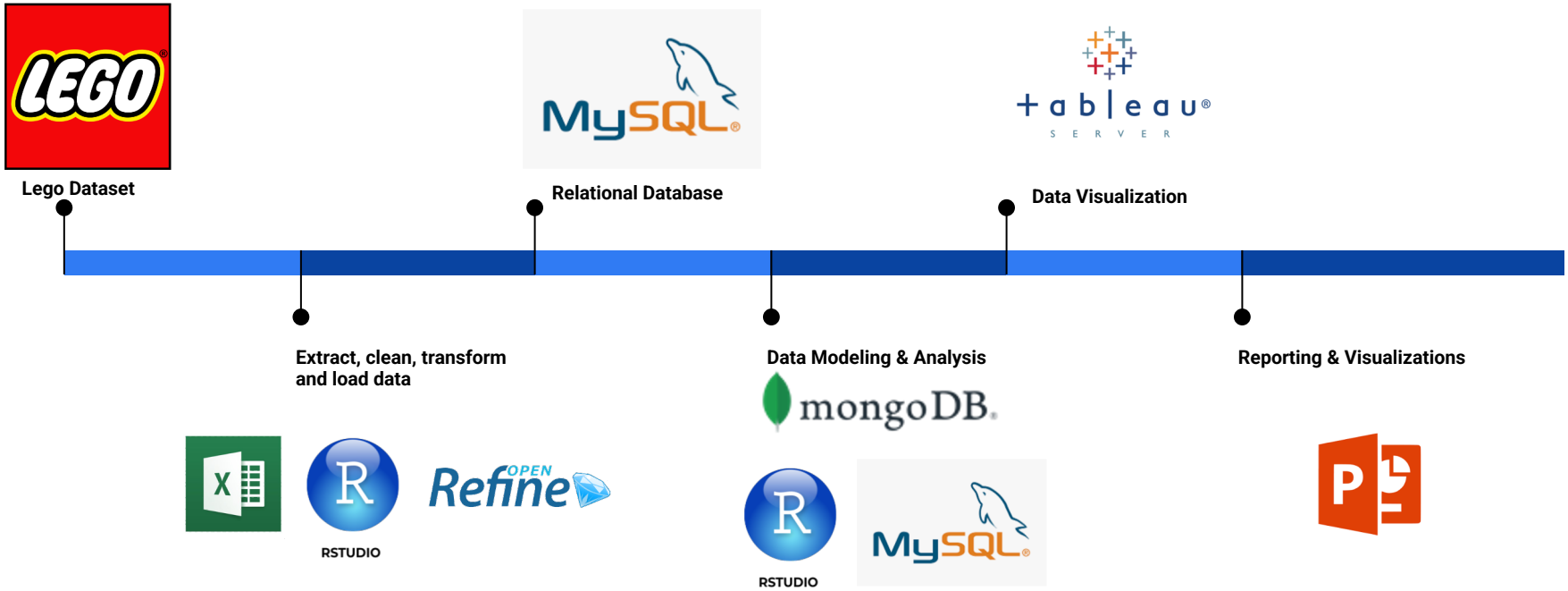
Timeliness

From 1949 to 2020

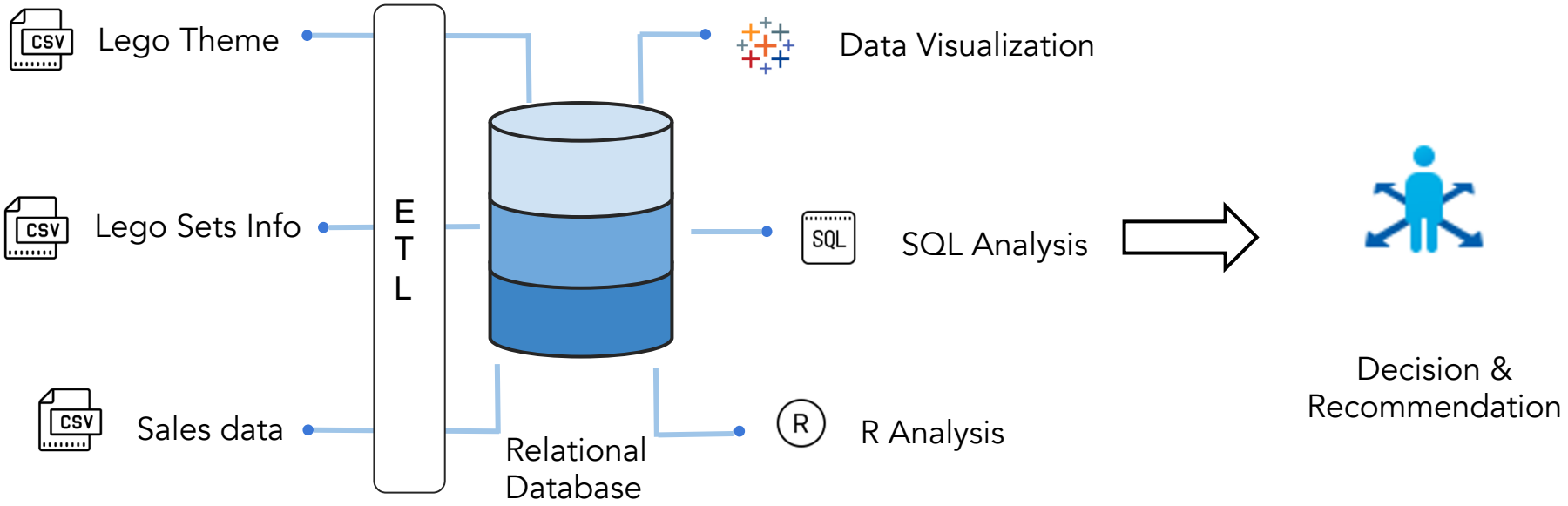
Integrity

Stored data in SQL database

Tools For Database Design



System diagram



ETL Overview

Data files

Data files come from the manufacturer data from 1949 to 2020. Sales data come from a sale record in 21 different countries.

Data cleaning

In our database the id in lego and sales data has some minor difference and we try to update it to uniform.

Processing script

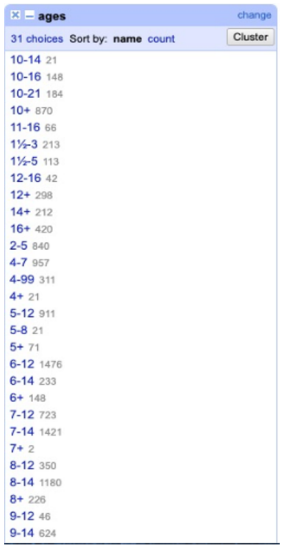
Using SQL scripts to create fact tables and link the data together. Also using SQL queries to normalize our data to 2NF first and then did the 3NF as well. After that we store the data to sql and mongodb and insert them also to R to do analysis.

Production Database

Store the 3NF data into SQL & Mongodb. Result is a dataset of lego fact information included themes, sets, colors, difficulties, and sales information.

Data preparation

- Group ages to avoid duplicated.
- Using openrefine to clean the data and uniform the set id in each of the dataset.
- Delete the column that we are not going to use and extract all columns into our database.



XMASTREE-1	Christmas Tree
wwgp1-1	Wild West Limited Edition Gift Pack
WISHINGWELL-1	Wishing Well
WILLIAM-1	Will.i.am
Wiesbaden-1	LEGO Store Grand Opening Exclusive Set, Wiesbaden, Germany
WHITEHOUSE-1	Micro White House
WEETABIX5-1	Weetabix Promotional Lego Village
WEETABIX4-1	Weetabix Promotional House 1
WEETABIX3-1	Weetabix Promotional House 2
WEETABIX2-1	Weetabix Promotional Windmill

XMASTREE	Christmas Tree
wwgp1	Wild West Limited Edition Gift Pack
WISHINGWELL	Wishing Well
WILLIAM	Will.i.am
Wiesbaden	LEGO Store Grand Opening Exclusive Set, Wiesbaden, Germany
WHITEHOUSE	Micro White House
WEETABIX5	Weetabix Promotional Lego Village

Data process -- DDL

```sql
create database lego_newnew;
use lego_newnew;

CREATE TABLE themes (
theme_id INT NOT NULL,
theme_name VARCHAR(255) NULL,
parent_id VARCHAR(255) NULL,
    PRIMARY KEY (`theme_id`)
);

CREATE TABLE sets (
set_id INT NOT NULL,
name VARCHAR(255) NULL,
year VARCHAR(255) NULL,
theme_id INT NULL,
num_parts VARCHAR(255) NULL,
    PRIMARY KEY (set_id),
    FOREIGN KEY (`theme_id`)
    REFERENCES `lego_newnew`.`themes` (`theme_id`)
);

CREATE TABLE inventory (
inventory_id INT NOT NULL,
version VARCHAR(255) NULL,
set_num INT NULL,
PRIMARY KEY (`inventory_id`),
    FOREIGN KEY (`set_num`)
    REFERENCES `lego_newnew`.`sets` (`set_id`)
);
```

```sql
CREATE TABLE inventory_sets (
inventory_id INT NULL,
set_id INT NULL,
quantity VARCHAR(255) NULL,
    FOREIGN KEY (`inventory_id`)
    REFERENCES `lego_new`.`inventory` (`inventory_id`),
    FOREIGN KEY (`set_id`)
    REFERENCES `lego_newnew`.`sets` (`set_id`)
);

CREATE TABLE part_category (
category_id INT NOT NULL,
category_name VARCHAR(255) NULL,
PRIMARY KEY (`category_id`)
);


CREATE TABLE parts (
part_id VARCHAR(255) NOT NULL,
part_name VARCHAR(255) NULL,
category_id INT NULL,
    PRIMARY KEY (`part_id`),
    FOREIGN KEY (`category_id`)
    REFERENCES `lego_newnew`.`part_category` (`category_id`)
);


CREATE TABLE colors (
color_id INT NOT NULL,
color_name VARCHAR(255) NULL,
rgb VARCHAR(255) NULL,
is_trans VARCHAR(255) NULL,
    PRIMARY KEY (`color_id`)
);
```

```sql
CREATE TABLE inventory_parts (
inventory_id INT NOT NULL,
part_id VARCHAR(255) NULL,
color_id INT NULL,
quantity VARCHAR(255) NULL,
is_spare VARCHAR(255) NULL,
    FOREIGN KEY (`color_id`)
    REFERENCES `lego_newnew`.`colors` (`color_id`),
    FOREIGN KEY (`inventory_id`)
    REFERENCES `lego_newnew`.`inventory` (`inventory_id`),
    FOREIGN KEY (`part_id`)
    REFERENCES `lego_newnew`.`parts` (`part_id`)
);


CREATE TABLE lego_sets (
ages VARCHAR(255) NULL,
list_price VARCHAR(255) NULL,
piece_count INT NULL,
prod_desc VARCHAR(255) NULL,
set_id INT NULL,
review_difficulty VARCHAR(255) NULL,
set_name VARCHAR(255) NULL,
star_rating VARCHAR(255) NULL,
theme_name VARCHAR(255) NULL,
country VARCHAR(255) NULL,
    FOREIGN KEY (`set_id`)
    REFERENCES `lego_newnew`.`sets` (`set_id`)
);
```

Data process -- DML

```
INSERT INTO inventory_sets (inventory_id,set_id,quantity) VALUES
( 35,'75911-1','1' ),
( 35,'75912-1','1' ),
( 39,'75048-1','1' ),
( 39,'75053-1','1' ),
( 50,'4515-1','1' ),
( 50,'4520-1','2' ),
( 50,'4531-1','1' ),
( 71,'7690-1','1' ),
( 71,'7691-1','1' ),
( 71,'7692-1','1' ),
( 71,'7693-1','1' ),
( 71,'7694-1','1' ),
( 71,'7695-1','1' ),
( 71,'7697-1','1' ),
( 81,'8451-1','1' ),
( 81,'8453-1','1' ),
( 87,'10233-1','1' ),
( 87,'88002-1','1' ),
( 87,'8870-1','1' ),
( 87,'8878-1','1' ),
( 87,'8879-1','1' ),
```

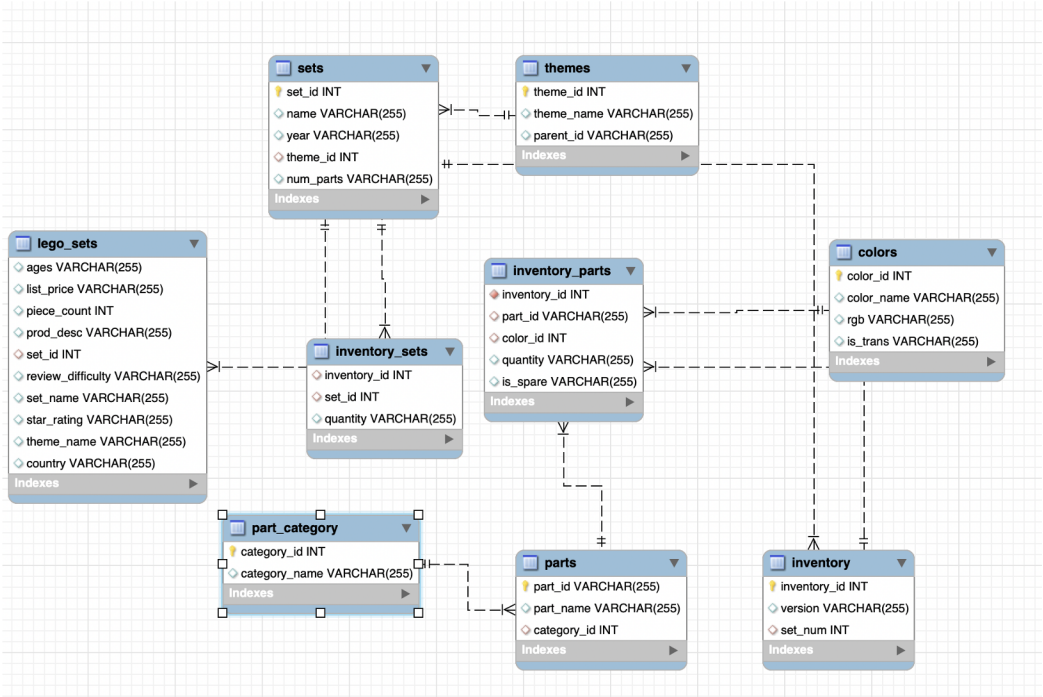
```
INSERT INTO themes (theme_id,theme_name,parent_id) VALUES
( 1,'technic',null ),
( 2,'arctic technic','1' ),
( 3,'competition','1' ),
( 4,'expert builder','1' ),
( 5,'model','1' ),
( 6,'airport','5' ),
( 7,'construction','5' ),
( 8,'farm','5' ),
( 9,'fire','5' ),
( 10,'harbor','5' ),
( 11,'off-road','5' ),
( 12,'race','5' ),
( 13,'riding cycle','5' ),
( 14,'robot','5' ),
( 15,'traffic','5' ),
( 16,'roboriders','1' ),
( 17,'speed slammers','1' ),
( 18,'star wars','1' ),
( 19,'supplemental','1' ),
( 20,'throwbot slizer','1' ),
```

```
INSERT INTO parts (part_id,part_name,category_id) VALUES
( '0687b1','Set 0687 Activity Booklet 1',17 ),
( '0901','Baseplate 16 x 30 with Set 080 Yellow House Print',1 ),
( '0902','Baseplate 16 x 24 with Set 080 Small White House Print',1 ),
( '0903','Baseplate 16 x 24 with Set 080 Red House Print',1 ),
( '0904','Baseplate 16 x 24 with Set 080 Large White House Print',1 ),
( '1','Homemaker Bookcase 2 x 4 x 4',7 ),
( '10','Baseplate 24 x 32',1 ),
( '10016414','Sticker Sheet #1 for 41055-1',17 ),
( '10019stk01','Sticker for Set 10019 - (43274/4170393)',17 ),
( '10026stk01','Sticker for Set 10026 - (44942/4184185)',17 ),
( '10029stk01','Sticker for Set 10029 - (4216816)',17 ),
( '10036stk01','Sticker for Set 10036 - (821407)',17 ),
( '10039','Pullback Motor 8 x 4 x 2/3',44 ),
( '10048','Minifig Hair Tousled',13 ),
( '10049','Minifig Shield Broad with Spiked Bottom and Cutout Corner',27 ),
( '10049pr0001','Minifig Shield Broad with Spiked Bottom and Cutout Corner with Handprint Pri
( '10050','Minifig Sword [Uruk-hai]',27 ),
( '10051','Minifig Helmet Castle with Lateral Comb [Uruk-hai]',27 ),
( '10051pr01','Minifig Helmet Castle with Lateral Comb and Handprint Print',27 ),
( '10052','Minifig Beard, Rounded End [Gandalf]',27 ),
( '10053','Minifig Sword Small',27 ),
```

```
INSERT INTO part_category (category_id,category_name) VALUES
( 1,'Baseplates' ),
( 2,'Bricks Printed' ),
( 3,'Bricks Sloped' ),
( 4,'Duplo, Quatro and Primo' ),
( 5,'Bricks Special' ),
( 6,'Bricks Wedged' ),
( 7,'Containers' ),
( 8,'Technic Bricks' ),
( 9,'Plates Special' ),
( 10,'Tiles Printed' ),
( 11,'Bricks' ),
( 12,'Technic Connectors' ),
( 13,'Minifigs' ),
( 14,'Plates' ),
( 15,'Tiles Special' ),
( 16,'Windows and Doors' ),
( 17,'Non-LEGO' ),
( 18,'Hinges, Arms and Turntables' ),
( 19,'Tiles' ),
```

```
INSERT INTO sets (set_id,name,year,theme_id,num_parts) VALUES
( '001','Gears','1965',1,'43' ),
( '0011','Town Mini-Figures','1978',84,'12' ),
( '0011','Castle 2 for 1 Bonus Offer','1987',199,'0' ),
( '0012','Space Mini-Figures','1979',143,'12' ),
( '0013','Space Mini-Figures','1979',143,'12' ),
( '0014','Space Mini-Figures','1979',143,'12' ),
( '0015','Space Mini-Figures','1979',143,'18' ),
( '0016','Castle Mini Figures','1978',186,'15' ),
( '002','4.5V Samsonite Gears Motor Set','1965',1,'3' ),
( '003','Master Mechanic Set','1966',366,'403' ),
( '005','Basic Building Set in Cardboard','1965',366,'35' ),
( '005','Discovery Set','1967',366,'0' ),
( '00-6','Special Offer','1985',67,'0' ),
( '010','Basic Building Set in Cardboard','1965',366,'57' ),
( '010','Basic Building Set','1968',366,'77' ),
( '011','Basic Building Set','1968',366,'145' ),
( '021','Wheel Set','1966',366,'183' ),
( '022','Basic Building Set','1968',366,'110' ),
( '0241199312','DC Super Heroes: Character Encyclopedia','2016',497,'7' ),
( '0241357594','Star Wars Build Your Own Adventure: Galactic Missions','2019',497,'0' ),
( '0241363500','Amazing Vehicles','2019',497,'26' ),
```

Classic ER Model



Lego_sets: general data of Lego such as reviews, price, year, corresponding ages

Sets: set information of Lego such as number of parts contained, theme of the set, production year

Themes: theme information of Lego

Inventory_parts: information about parts that are stored in each inventory

Colors: color information of each part

Inventory: information of the inventory

Part_category: a normalized table of part

Parts: part information which includes category and part_name

Inventory_sets: table that connects inventory and sets

Normalizing ER Model

```
CREATE TABLE country (
 country_id INT NOT NULL,
 country_name VARCHAR(255) NULL,
 primary key (`country_id`)
);

CREATE TABLE age_info (
 age_range_id INT NOT NULL,
 age_range VARCHAR(255) NULL,
 primary key (`age_range_id`)
);

CREATE TABLE review_difficulty_info (
 review_difficulty_id INT NOT NULL,
 review_difficulty VARCHAR(255) NULL,
 primary key (`review_difficulty_id`)
);
```

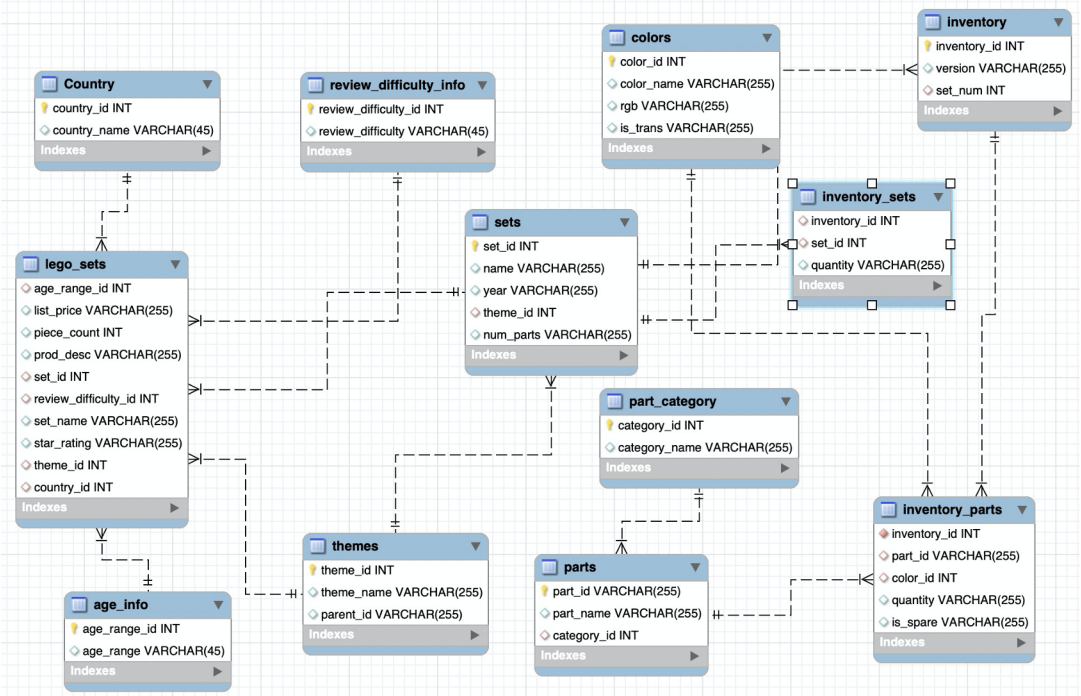
```
CREATE TABLE lego_sets (
age_range_id INT NULL,
list_price VARCHAR(255) NULL,
piece_count INT NULL,
prod_desc VARCHAR(255) NULL,
set_id INT NULL,
review_difficulty_id INT NULL,
set_name VARCHAR(255) NULL,
star_rating VARCHAR(255) NULL,
theme_id INT NULL,
country_id INT NULL,
    FOREIGN KEY (`set_id`)
    REFERENCES `lego_final_normalized`.`sets` (`set_id`),
    FOREIGN KEY (`theme_id`)
    REFERENCEs `lego_final_normalized`.`themes` (`theme_id`),
    FOREIGN KEY (`country_id`)
    REFERENCEs `lego_final_normalized`.`country` (`country_id`),
    FOREIGN KEY (`age_range_id`)
    REFERENCEs `lego_final_normalized`.`age_info` (`age_range_id`),
    FOREIGN KEY (`review_difficulty_id`)
    REFERENCES `lego_final_normalized`.`review_difficulty_info`(`review_difficulty_id`)
);
```

Production Database

Normalized Data

Primary normalized relationships include review_difficulty_id, age_range_id, and country_id in the lego_sets table.

Update anomaly, Insert Anomaly, and Delete Anomaly can be avoided with the normalization.



Applied SQL Queries

```sql
# Calculate the number of each difficulty and their average price.
SELECT
    COUNT(t.theme_name) AS number, review_difficulty AS difficulty, AVG(list_price)
FROM
    themes t,
    sets s,
    lego_sets ls
WHERE
    t.theme_id=s.theme_id AND
    s.set_id=ls.set_id
GROUP BY
    review_difficulty;
```

```sql
# Find the avg quatity of parts in different difficulty.
SELECT
    AVG(num_parts), review_difficulty AS difficulty
FROM
    sets s,
    lego_sets ls
WHERE
    s.set_id=ls.set_id
GROUP BY
    review_difficulty;
```

```sql
# Find the first 3 countries where lego becomes popular.
SELECT
    COUNT(set_id),country
FROM
    lego_sets
GROUP BY
    country
ORDER BY
    COUNT(set_id) DESC LIMIT 3;
```

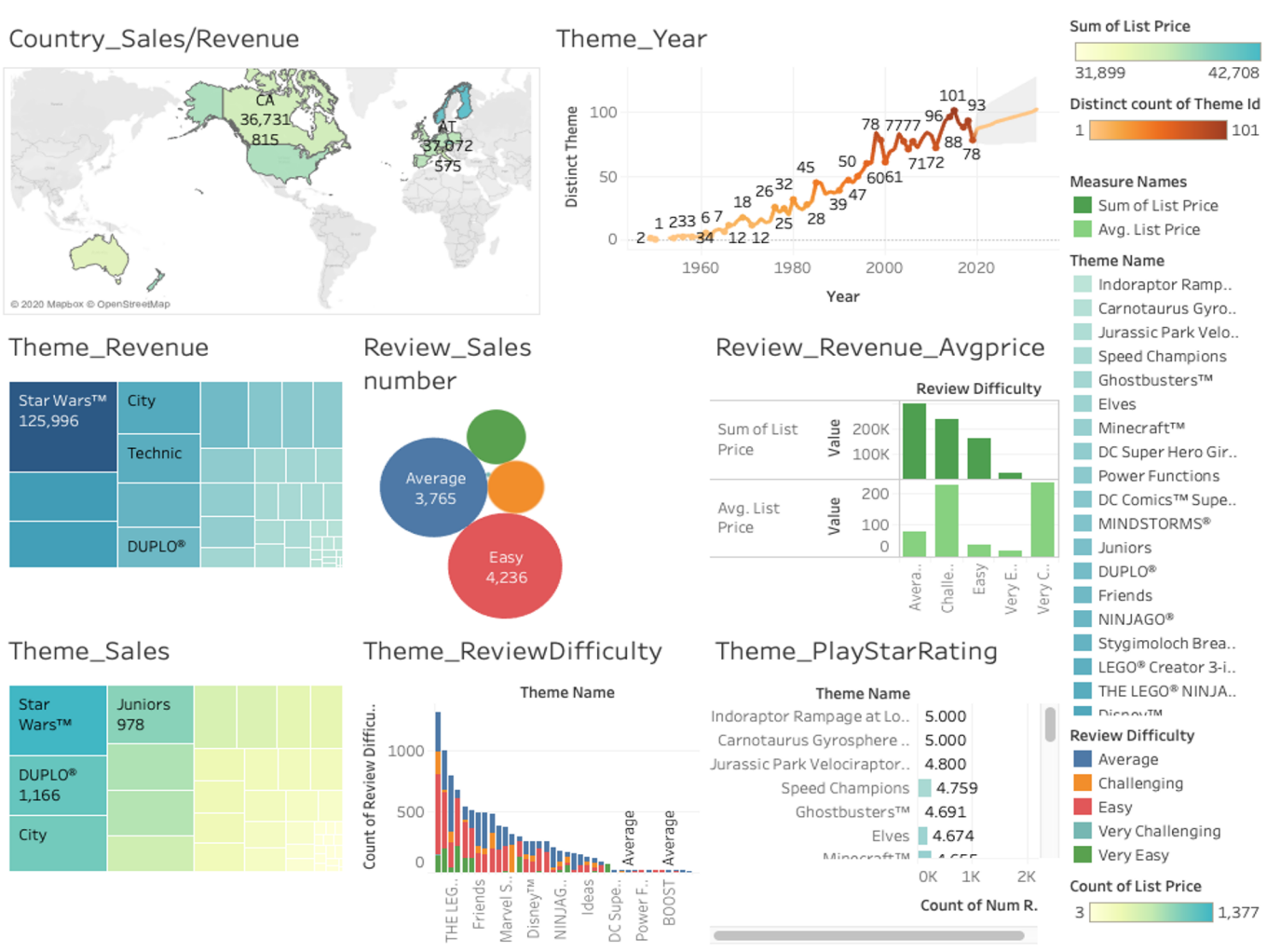
```sql
# Find the most popular colors in each country by each year
SELECT
    country, year, color_id, count(*) as sales
FROM
    lego_sets
    INNER JOIN sets ON lego_sets.set_id = sets.set_id
    INNER JOIN inventory ON inventory.set_num = sets.set_id
    INNER JOIN inventory_parts ON inventory.inventory_id = inventory_parts.inventory_id
GROUP BY country, year, color_id
ORDER BY sales DESC;
```

Applied MongoDB Queries

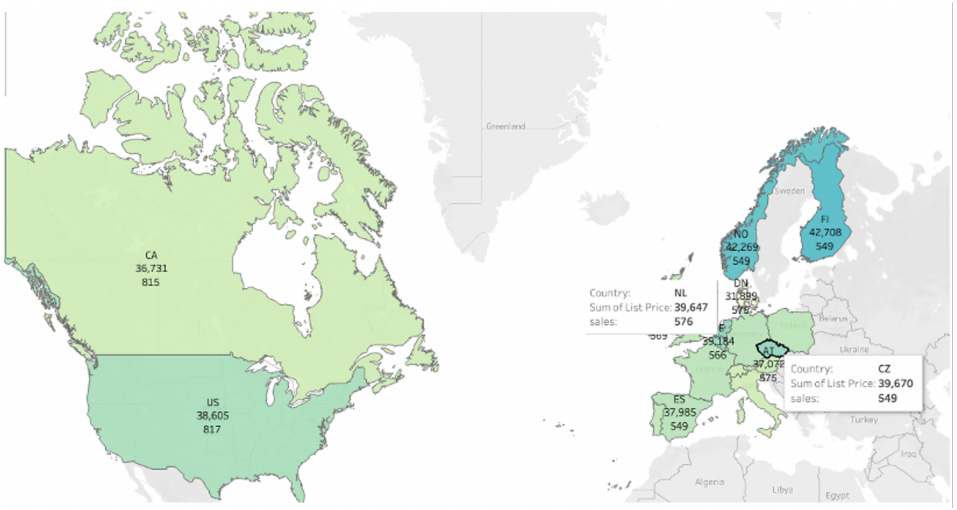

```
/*Find the average and challenging sets that have a low rating and high price*/
db.lego_sets.find({ "review_difficulty": { $in: ["Average", "Challenging"] },
"play_star_rating": { $lt: 2.5 }, "list_price": { $gt: 150 } });

/*Find the sets that have a lot of pieces count, with a good review and a cheap price*/
db.lego_sets.find({ "piece_count": { $gt: 2000 },
"play_star_rating": { $gt: 3.0 }, "list_price": { $lt: 150 } });
```

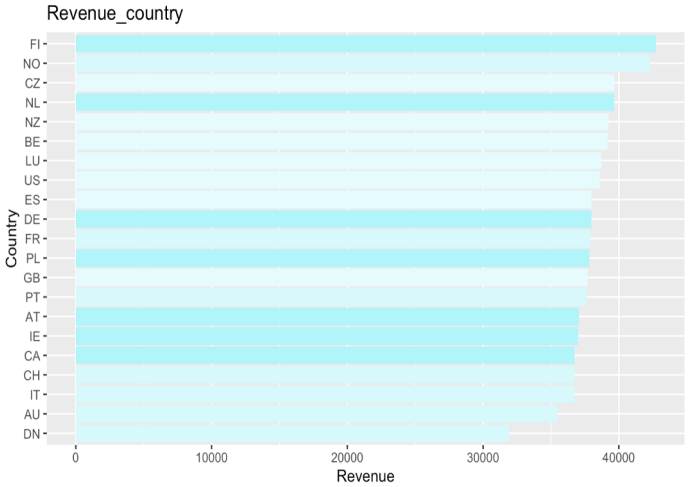
Key	Value	Type	Key	Value	Type
(1) 5fcea2625ee9242271596c07	{15 fields}	Document	(1) 5fcea2625ee92422715966b0	{15 fields}	Document
_id	5fcea2625ee9242271596c07	ObjectId	(2) 5fcea2625ee9242271596fa7	{15 fields}	Document
ages	9-14	String	(3) 5fcea2625ee9242271597090	{15 fields}	Document
list_price	158.5878	Double	(4) 5fcea2625ee9242271597aab	{15 fields}	Document
num_reviews	5	Int32	(5) 5fcea2695ee9242271599788	{15 fields}	Document
piece_count	1,426 (1.4K)	Int32	(6) 5fcea2695ee924227159a07f	{15 fields}	Document
play_star_rating	2	Int32	(7) 5fcea2695ee924227159a168	{15 fields}	Document
prod_desc	Stand your ground and save the realm with the 2-in-1 Knighton Castle!	String	(8) 5fcea2695ee924227159ab83	{15 fields}	Document
prod_id	70357	Int32	(9) 5fcea2695ee924227159b7a0	{15 fields}	Document
prod_long_desc	Explore the awesome 2-in-1 Knighton Castle and defend the king! Unclasp th	String	(10) 5fcea2695ee924227159c097	{15 fields}	Document
review_difficulty	Challenging	String	(11) 5fcea2695ee924227159c180	{15 fields}	Document

Data Visualization



Sales Revenue & Numbers in Countries





Top 3 countries with the most sales

United states(US): 817
Canada(CA): 815
Netherlands(NL): 576

Top 3 countries with the most revenue

Finland(FL): $42708
Norway(NO): $42269
Czech Republic(CZ): $39670

The least revenue

Denmark(DN):$ 31899

Relationship: not positively correlated

Finland, Norway and Czech Republic sell less numbers of lego sets but earn more revenue than other countries.

Denmark: may need promotion strategy to increase sales revenue

Theme(unique) each Years

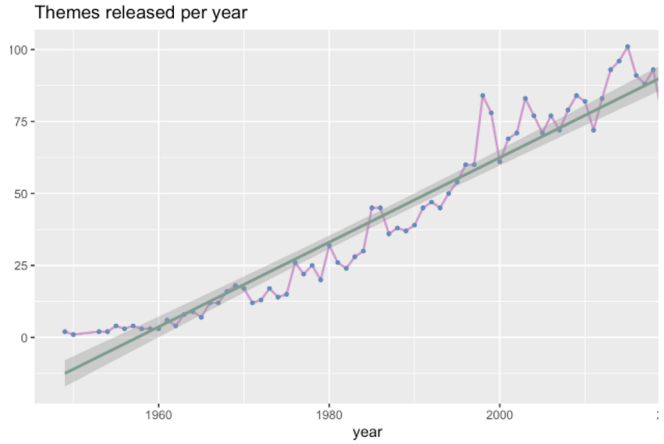
Trend before 2020:

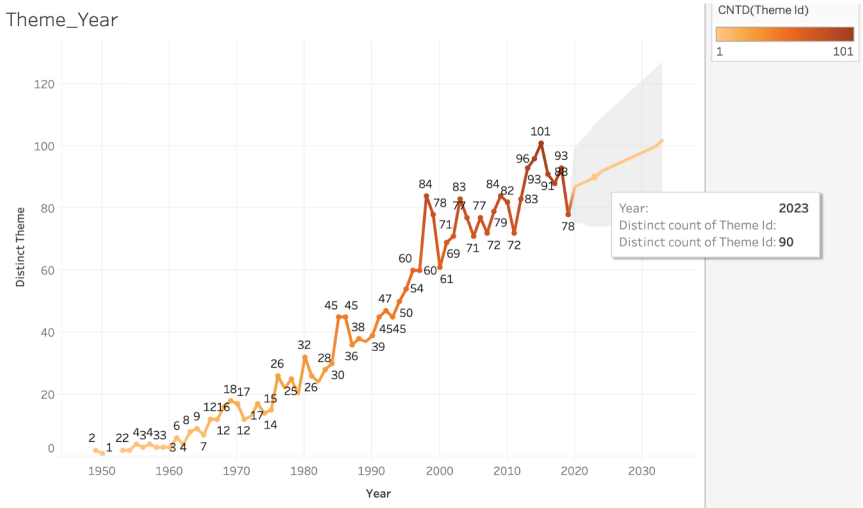
From the line in the first graph (R programming), the number of themes released each year have generally increased.

However, there is a fluctuation from 2004 to 2010. Those years represent difficult period for The Lego Group, when the company teetered on the brink of bankruptcy before picking up the pieces.

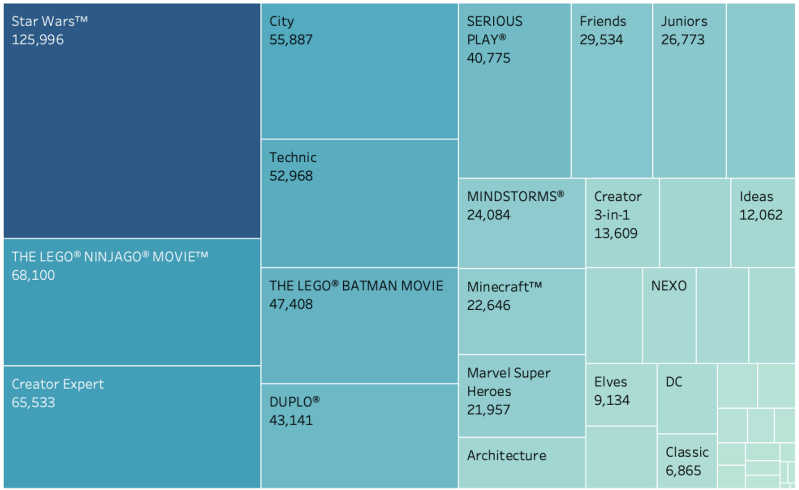
Trend after 2020:

From the forecasting part on the second graph (Tableau), the trend after 2020 is upward. For example, the number of theme which will release in 2023 is 90.





Theme_Revenue

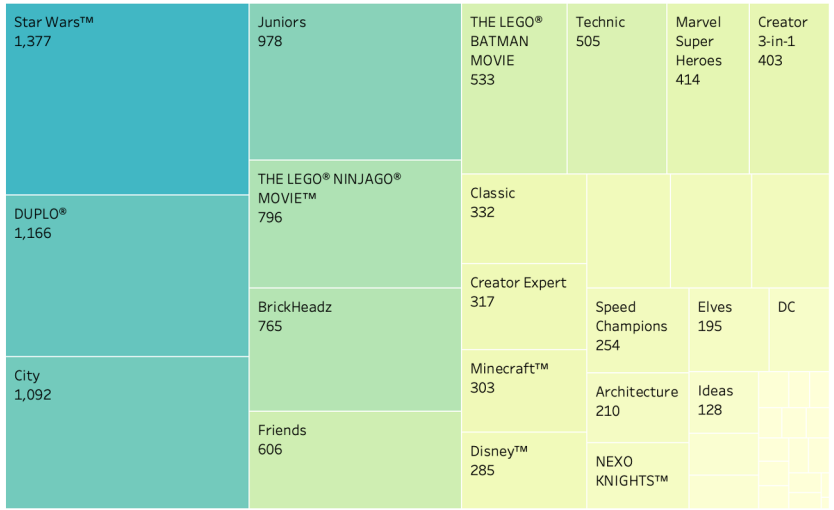


Sales Revenue for Themes

Top 5 Themes with the most revenue:

Star Wars: $125996
The Lego ninjago movie: $68100
Creator Expert: $65533
City: $55887
Technic: $52968

Theme_Sales



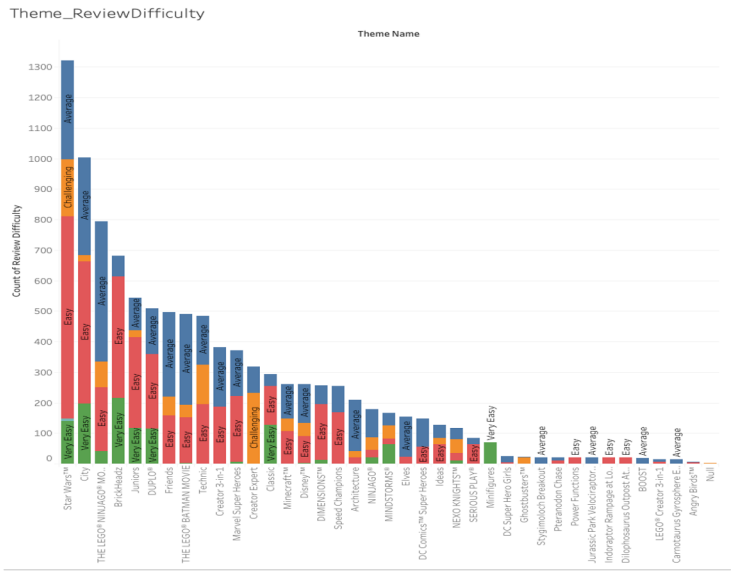
Sales Number for Themes

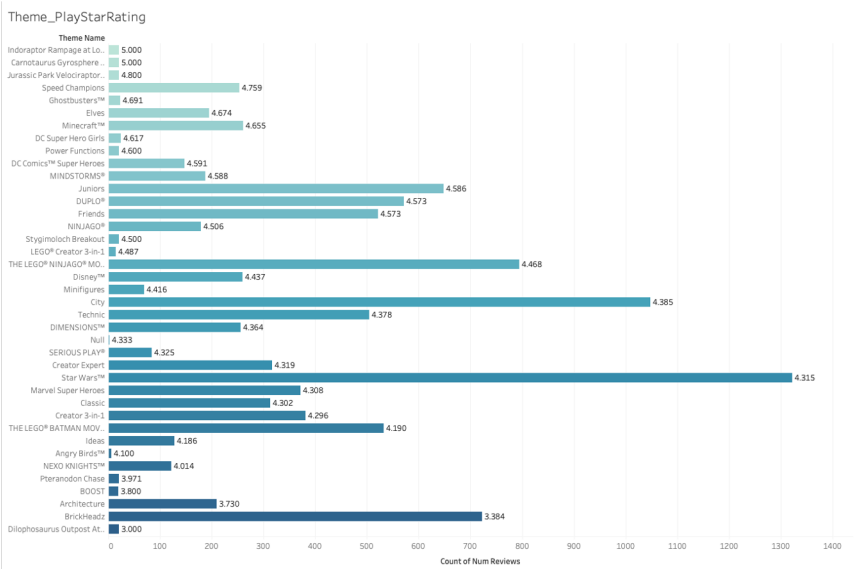
Top 5 Themes with the most sales number:

Star Wars: 1377; Duplo: 1166; City: 1092;
Junior: 978; The Lego ninjago movie: 796

Star Wars:the most popular theme
Duplo: high sales number,lower price (for age 1-5)
The Lego ninjago movie:
Higher price, low sales numbers (for age 8 or older)

Difficulty Level & Play Star Rating for Theme





Top themes (number of each reviewed difficulty level)

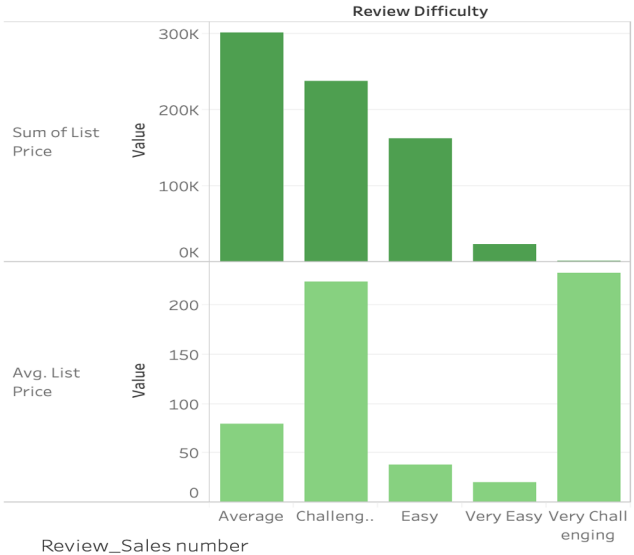
Star wars: easy > average > challenging
City: easy > average > very easy

Top themes (number of review/play star rating)

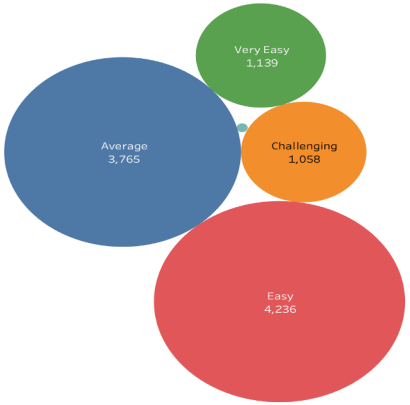
Star wars: high number, comparatively low rating (4.315)
City: high number, comparatively low rating (4.385)

Suggestion: Some customers who buy top themes only because of the popularity and they found it is not worth playing at the end. Increase playability for popular themes and apply marketing strategy to promote other niche themes. For example, Indoraptor Rampage at Lockwood Estate (high rating/low review number).

Review_Revenue_Avgprice



Review_Sales number



Revenue & Avg Price in Difficulty Level

From the 1st bar chart:
Average level and challenging level earn the most revenue

From the 2rd bar chart:
The average price for Challenging level is almost the same as the average price for very challenging level

Sales Numbers in Difficulty Level

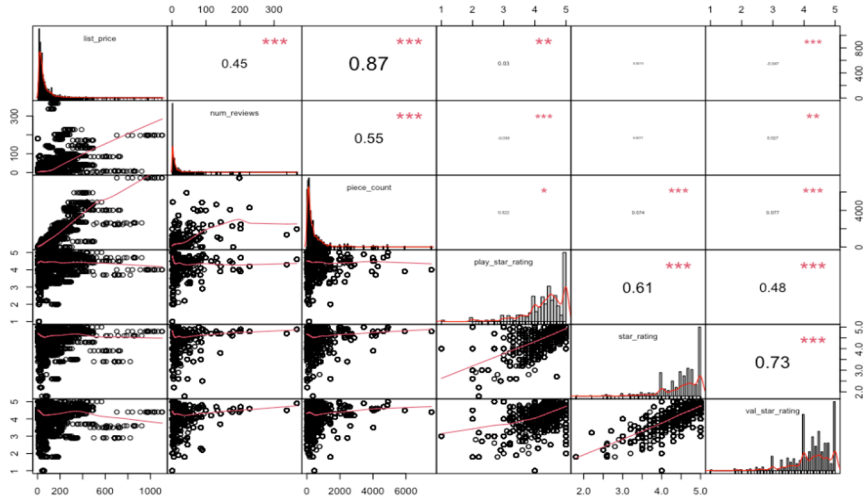
From the review_sales number graph:
The sales number of easy level and average level rank 1st and 2rd while challenging level only sold 1058 sets in total.

Compared with bar charts:
If Lego group has a high profit margin in challenging level sets, one suggestion is that they can increase the production of challenging level sets to earn more profits.

Price Analysis

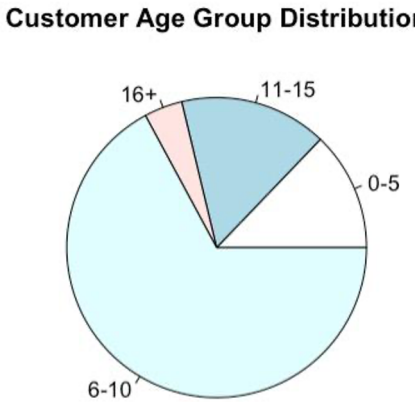
Highly corrected pairs

- list_price vs. piece_count
- play_star_rating vs. star_rating
- star_rating vs. val_star_rating



Customer age distribution

- Majority of the lego sets are designed for customers between age 6 - 10.
- Small proportion for 16 +



Modeling

- Create dummy variables for categorical variables
- Train: 70% Test: 30%
- Backward stepwise selection
- Model performance: adjusted R^2 =0.7928

Next step

Other models

PCA analysis to reduce dimensionality

Get more price related data to better understand price influencers

```
#select numerical features
num_data<-lego_sets3 %>% select_if(is.numeric)

#join with dummy variables
temp <- as.data.frame(cbind(
  num_data,
  'ages'= dummy.code(lego_sets3$ages_grouped),
  'review_difficulty'= dummy.code(lego_sets3$review_difficulty)))
```

```
Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)       -3.289e+01  6.299e+00  -5.221 1.83e-07 ***
num_reviews        5.806e-02  1.964e-02   2.956  0.00313 **
piece_count        1.088e-01  9.524e-04 114.197  < 2e-16 ***
play_star_rating   1.381e+01  1.102e+00  12.528  < 2e-16 ***
review_difficulty  3.381e+00  8.216e-01   4.115 3.92e-05 ***
val_star_rating   -2.487e+01  1.013e+00 -24.558  < 2e-16 ***
`ages.6-10`        7.955e+01  4.289e+00  18.546  < 2e-16 ***
`ages.11-15`       8.814e+01  4.137e+00  21.307  < 2e-16 ***
`ages.0-5`         9.371e+01  4.673e+00  20.053  < 2e-16 ***
---
Signif. codes:
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 46.69 on 7108 degrees of freedom
Multiple R-squared:  0.793,     Adjusted R-squared:  0.7928
F-statistic:  3404 on 8 and 7108 DF,  p-value: < 2.2e-16
```

Recommendation

- Explore Asian markets: fast growing area, include more relevant data
- Promote collaboration: popular IP(intellectual property), customer communities
- Improve popular product's playability
- Adjust production based on profit margin
- Grab the market trend

Thank You!