CSDA 1010 Final Project
Lego’s Future Strategies in the Games Market

By Siddharth Panchal, Alex Liszewski, Laura Visentin, Victoria Tenuta, Monique Holz
January 18, 2020

1.0 Abstract Identifying the influential relationship key characteristics have with the price of a product can have a drastic impact on a business's sales strategy. Throughout our research, we will explore various analytical approaches to predict the list price of lego sets based on various characteristics. In addition, we will develop a sales strategy that Lego can use to move forward in the Games market. The selected dataset contains features on over 12,000 lego sets that are sold in the global market. In this report we will test various algorithms and methodologies in order to determine an all-encompassing, strategic approach that the Lego Company can use to celebrate and promote their 88th year in business.

1.1 Introduction

Lego (hereafter referred to as “the company”) has historically been one of the most popular products on the Games market, with a wide variety of product offerings around the world. Lego is ubiquitous, and its pieces are all part of a unique & universal system. In fact, it is estimated that each year, people around the world spend over 5 billion hours playing with Lego sets (National Geographic - Lego Facts). Although designs over the years vary drastically, each piece remains, compatible with existing pieces. Since inception, the Lego Company has achieved great success. To commemorate their 88th anniversary, the company plans to release a limited edition, 880 piece lego set in each of the 4 major continents in which they currently operate.

1.2 Background

The dataset “Lego Sets”, contains data collected from over 12,000 Lego products, and includes information on consumer age, list price, piece count, difficulty rating, theme and country. Our selected output variable will be list price, with our input variables to be determined through various feature selection methods. Using this data, our plan is to create a sales strategy for Lego regarding their new 880 piece Lego sets in North America, Asia, Europe & Oceania.

1.3 Objective

This report aims to provide the company with a recommendation that is both feasible and well-rounded. Our goal is to provide the company with a highly profitable, unique solution that allows them to individually target each major continent that they currently operate in with a unique sales strategy. Our analysis aims to be all-encompassing, with specific details surrounding list price, target age group, and difficulty level. We will begin by identifying which input variables hold value and which we can disregard. We will then proceed to clean the data by removing any outliers that may skew the data. Our cleaning process will also take care of any missing information in the dataset.

2.0 Analytical Objectives

Throughout our analytical process, there are numerous areas we intend to dive into, and several questions which we would like to answer. Our main objectives are listed below.

1. How does the product mix & range of difficulty levels vary across each continent?
2. What is the average price range for each difficulty level? Is there any relationship between price and difficulty level? Does this vary drastically across the continents?
3. What is the relationship between piece count and price of Lego set?
4. What is the relationship between piece count and difficulty level?
5. How do difficulty level and piece count relate to product price?
6. Which age group has the highest average price in each continent?
7. Which age group has the most Lego sets associated with them?
8. Which sets and/or themes have the most reviews? Which age groups are these sets tied to?

2.1 Business Success Factors

The key success factors that we have identified as essential for this initiative are:

- Provide useful insights regarding the interrelationship of various characteristics of Lego sets and how they vary across continents
- Insights lead to launching a new marketing strategy to drive strong sales growth in 2020, including limited edition set this year.

2.2 Assessing the Situation

In order to assess the current situation of the organization, we must examine the factors which will impact it externally, such as competitors. The major competitors for Lego are Mattel, Bandai Namco & Hasbro. The below table shows how each company's brands are valued internationally.

Company	Brand Ranking	Revenue (US)	Brand Value (US)	Region (Origin)
Lego	1	$5.5B	$6.6B	Europe
Mattel	10	$4.5B	$0.2B	North America
Bandai Namco	2	$6.6B	$1.6B	Asia
Hasbro	6	$4.6B	$0.3B	North America

Data Source(s): Hasbro, Bandai-Namco, Mattel, Lego 2018 Annual Reports

After reviewing the 2018 Annual Reports of each company and “The annual report of the world's most valuable and strongest toy brands” written by Brand Finance, our competitive analysis is as follows:

- Lego remains the most valuable company within the sector with brand value of US$6.8 billion
- Lego's revenue was US$5.5 billion, due to the strength of themes like LEGO Ninjago and LEGO Star Wars.
- Bandai Namco brands continue to grow faster than competitors, with its brand value up 57% to US$1.6 billion
- Mattel is suffering from Disney licence sale, all their toy brands have seen a drop in value
- Hasbro's results suffered from the bankruptcy & subsequent liquidation of Toys R Us in the U.S. and other geographic markets as well as the "rapidly evolving retail landscape" in international markets, particularly Europe.

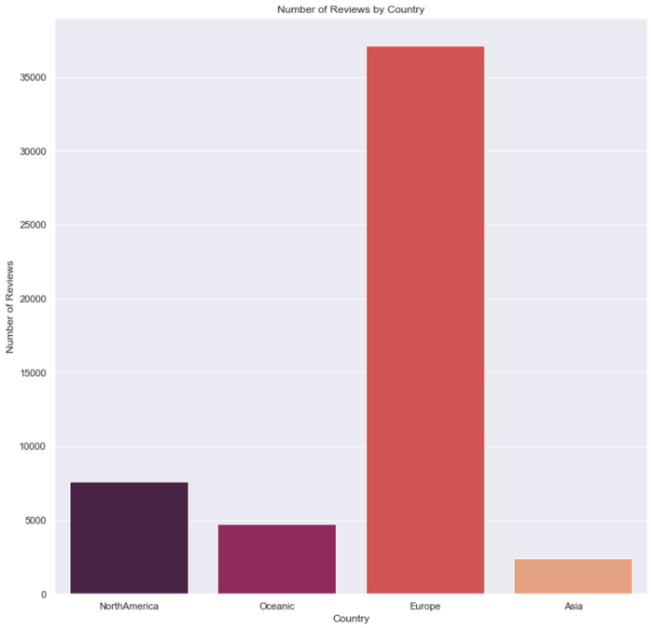
2.3 Application of Machine Learning Ethics

The dataset used in this report was collected via the Lego.com website to improve customer experience. Across several contexts, individuals consult customer ratings to inform their purchase decisions.

There are two types of bias that we need to consider and discuss with Lego with the first being “Social influence” bias. This is seen when a user's opinion is influenced by the opinion of others. If customers saw previous reviews, did they subconsciously become much more likely to develop an opinion in line with the masses? The decision is made to keep all rating levels as Lego does verify reviews with a third-party service provider, as well as rate the reviews helpfulness.

The second bias that we consider is called "Selection" bias which is when a set of users that submit a review is not representative of the entire purchasing population. A component of selection bias is known as under-coverage. This is when only a small subset of customers have submitted reviews, and these reviewers are NOT representative of the overall customer base.

If we look at the graph below, we see an example of under-coverage. One might conclude that Lego is more popular in Europe based on the number of reviews made, but that would be a bias. In conducting an environmental scan for this report, we discovered that Lego has been available in Europe since the 1930's, while Asia only had access to Lego sets three decades ago. Furthermore, Lego sets were higher priced in Asia due to import charges until a manufacturing facility was built in Asia in 2017. This could mean that only affluent consumers could afford to buy Lego sets. These reasons could potentially affect the number of reviews from Asia in our dataset. Therefore, to reduce the amount of human bias, we will leverage both supervised and unsupervised learning models to have more control over bias in the dataset and the data selection.



2.4 Assumptions

There are several assumptions that were made about the data and the project itself:

<u>Data</u>

1. List price is in USD currency.

2. There is no correlation between list price and free shipping of orders over $35.
3. South Korea is not included within the Country attribute.
4. All Lego sets were available in all listed countries.

<u>Project</u>

1. Access to key management resources throughout the project engagement to obtain information, ask clarifying questions and validate findings.
2. Funding is available to complete the project.

3.0 Data Understanding

The dataset "Lego Sets" has 14 attributes and 12,261 instances. The input/independent variables of the dataset are as follows:

- *Ages* (range from 1½ - 99)
- *Number of Reviews*
- *Piece count (ranging from 1 - 7,541 pieces)*
- *Product Description (ex. Celebrate London with this LEGO Architecture Skyline model!)*
- *Product ID* (unique value across product)
- *Long Product Description (ex. Celebrate the architectural diversity of London with this detailed LEGO brick model. The LEGO Architecture Skyline Collection offers models suitable for display in the home and office, and has been developed for all with an interest in travel, architectural culture, history and design…..)*
- *Difficulty rating* (Very easy, Easy, Average, Challenging & Very Challenging)
- *Set name* (741 unique values)
- *Star rating* (between 0-5, considered overall rating)
- *Theme Name* (40 unique values, ex. Star Wars, City, Angry Birds etc.)
- *Play star rating* (between 0-5, related to "Play Experience")
- *Val Star Rating* (between 0-5, related to "Value for Money")
- *Country* (21 countries across 4 continents)

The output/dependant variable of the dataset is:

- *List price* (ranges from $2.27 - $1,104.87)

3.1 Reading the Dataset

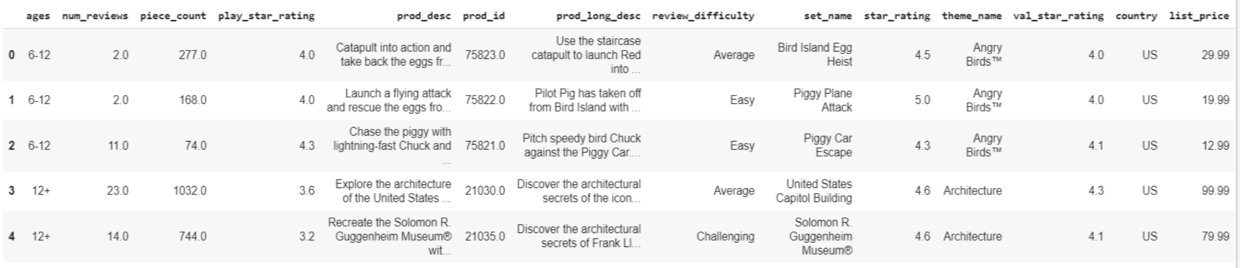
Various Python libraries were used throughout the analytical process. The code to load and initialize the libraries:

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import datetime as datetime
from pylab import *
%matplotlib inline
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split, cross_val_score
from sklearn.metrics import mean_squared_error
```

The dataset was downloaded from the Kaggle site and uploaded on github page to use directly into Jupyter Notebook using the Python statement below.

```
lego = pd.read_csv("https://raw.githubusercontent.com/sidthree6/csv/
master/lego_sets.csv")
```

3.2 Preview of the data



	ages	num_reviews	piece_count	play_star_rating	prod_desc	prod_id	prod_long_desc	review_difficulty	set_name	star_rating	theme_name	val_star_rating	country	list_price
0	6-12	2.0	277.0	4.0	Catapult into action and take back the eggs fr...	75823.0	Use the staircase catapult to launch Red into ...	Average	Bird Island Egg Heist	4.5	Angry Birds™	4.0	US	29.99
1	6-12	2.0	168.0	4.0	Launch a flying attack and rescue the eggs fro...	75822.0	Pilot Pig has taken off from Bird Island with ...	Easy	Piggy Plane Attack	5.0	Angry Birds™	4.0	US	19.99
2	6-12	11.0	74.0	4.3	Chase the piggy with lightning-fast Chuck and ...	75821.0	Pitch speedy bird Chuck against the Piggy Car....	Easy	Piggy Car Escape	4.3	Angry Birds™	4.1	US	12.99
3	12+	23.0	1032.0	3.6	Explore the architecture of the United States ...	21030.0	Discover the architectural secrets of the icon...	Average	United States Capitol Building	4.6	Architecture	4.3	US	99.99
4	12+	14.0	744.0	3.2	Recreate the Solomon R. Guggenheim Museum® wit...	21035.0	Discover the architectural secrets of Frank Ll...	Challenging	Solomon R. Guggenheim Museum®	4.6	Architecture	4.1	US	79.99

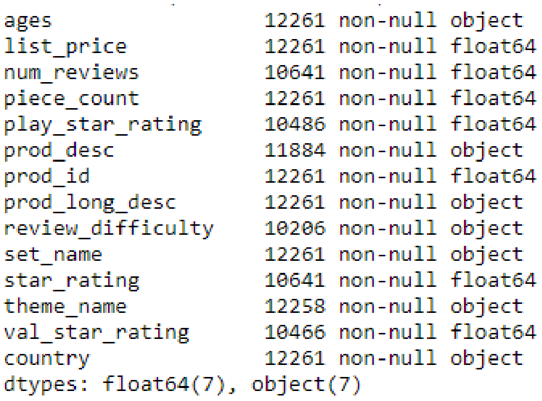
Table 1: Lego Dataset - first 5 rows

To find the data values of each variable, we ran the following python statement.

```
lego.info()
```

Within the dataset there is a mix of:

- Numerical features: piece count, list price, ratings, number of reviews
- Categorical: set name, ages, difficulty, theme name, country
- Character: product descriptions



```
ages                 12261 non-null object
list_price           12261 non-null float64
num_reviews          10641 non-null float64
piece_count          12261 non-null float64
play_star_rating     10486 non-null float64
prod_desc            11884 non-null object
prod_id              12261 non-null float64
prod_long_desc       12261 non-null object
review_difficulty    10206 non-null object
set_name             12261 non-null object
star_rating          10641 non-null float64
theme_name           12258 non-null object
val_star_rating      10466 non-null float64
country              12261 non-null object
dtypes: float64(7), object(7)
```

4.0 Data Preparation

Prior to beginning a deep dive analysis, we must clean the data using various techniques outlined in the next sections.

4.1 Dropping Unused Attributes

There are a number of attributes that are not required in our analysis. We used the following code to drop these attributes from the dataframe.

```
lego = lego.drop(["prod_desc", "prod_id", "prod_long_desc", "set_name",
"play_star_rating", "val_star_rating"], axis=1)
```

The characteristics: “prod_desc”, “prod_id” and “prod_long_desc” were dropped from our data set as we have deemed them to be insignificant when predicting list price. The “prod_desc” and “prod_long_desc” contains important information about the features of each Lego set, however, including them in our model will not add any additional value as the quantity of unique values in each are far too high. Moreover, the attribute “prod_id” is unique to each type of Lego set. It is used for the identification and tracking of the various sets, therefore, we determined that this attribute is insignificant when predicting “list_price”. Our dataset contains 3 different types of ratings - play star, val star and star rating. For simplicity, we will focus solely on star rating in our analysis.

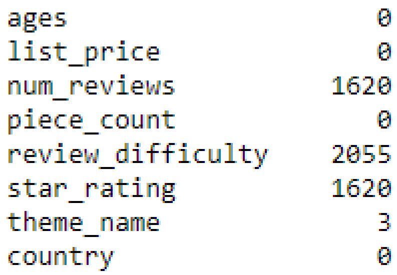
4.2 Working With & Converting Data Types

Ages: The dataset contains 31 different age groupings, ranging from 1½-3, 2-4, 4+, 5-8 to 7+, 9-12, 10-21, 16+ etc. To simplify this category, we have classified the age groupings into: “Toddlers & Preschoolers” ages 1-5, “Grade Schoolers” ages 5-8, “Tweens” ages 9-12 & “Teens+” ages 13+.
Difficulty Level: Within the dataset, difficulty levels are Very Easy, Easy, Average, Challenging & Very Challenging. We converted these to numeric values between 1-5 for simplicity & manipulation purposes.
Country: The dataset contains information across 21 countries. To simplify this category, we have grouped the countries into “North America”, “Asia”, “Oceania” & “Europe”
Theme Name: There are 40 themes across the dataset. We have added a “brand flag” to distinguish between brand name themes (ex. Star Wars™, DUPLO®, THE LEGO® NINJAGO® MOVIE™, etc..) and non-brand name themes (ex. Architecture, Minifigures, BrickHeadz, etc..)

4.3 Handling Missing Values

Firstly, we will identify all instances of missing values.

```
lego.isnull().sum()
```



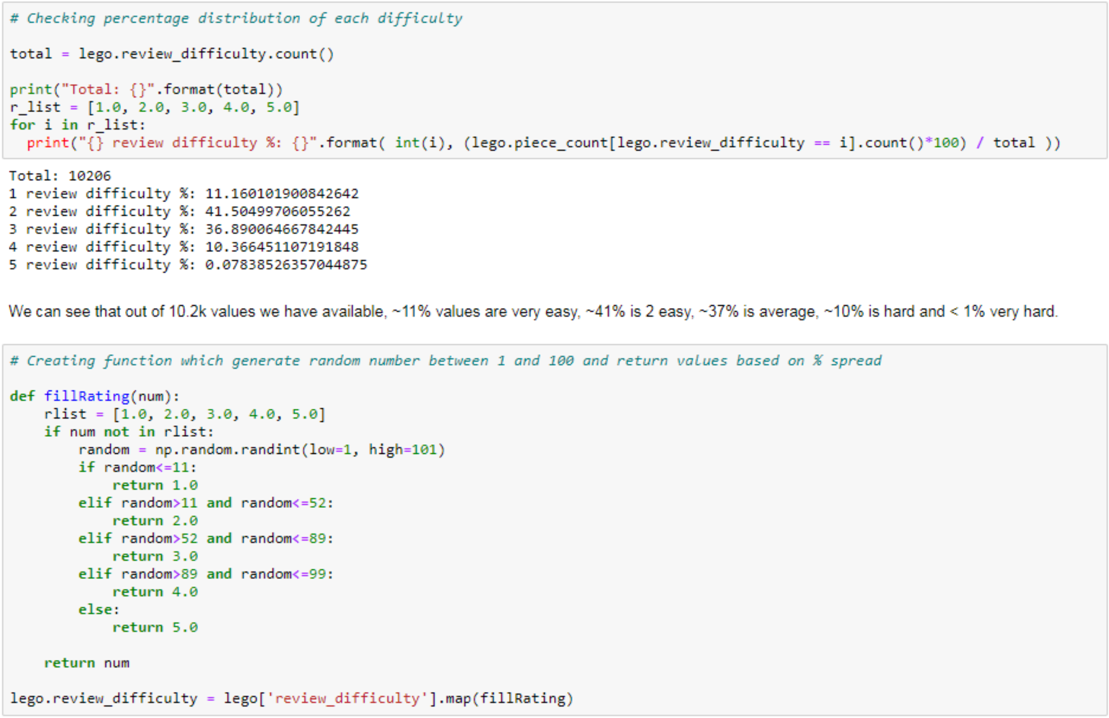
```
ages                    0
list_price              0
num_reviews          1620
piece_count             0
review_difficulty    2055
star_rating          1620
theme_name              3
country                 0
```

The columns, num_reviews, star_rating, review_difficulty & theme_name contain missing records. We have the option of either dropping these values, or using imputation to fill these values.

num_reviews: If an instance has a missing “num_reviews”, we will assume there were no reviews and fill the value with 0.

```
lego["num_reviews"].fillna(0, inplace = True)
```

review_difficulty: Rather than using mean imputation, we can create a function which generates a random number from 1 to 100 and return review difficulty based on %. So for example, if we have 60% of lego sets falling in Average Difficulty, 20% Easy and 20% Very Easy, the function will have a 60% probability to return Average, 20% to return Easy, and so on. Execution is shown below.



```
# Checking percentage distribution of each difficulty

total = lego.review_difficulty.count()

print("Total: {}".format(total))
r_list = [1.0, 2.0, 3.0, 4.0, 5.0]
for i in r_list:
  print("{} review difficulty %: {}".format( int(i), (lego.piece_count[lego.review_difficulty == i].count()*100) / total ))
```

```
Total: 10206
1 review difficulty %: 11.160101900842642
2 review difficulty %: 41.50499706055262
3 review difficulty %: 36.890064667842445
4 review difficulty %: 10.366451107191848
5 review difficulty %: 0.07838526357044875
```

We can see that out of 10.2k values we have available, ~11% values are very easy, ~41% is 2 easy, ~37% is average, ~10% is hard and < 1% very hard.

```
# Creating function which generate random number between 1 and 100 and return values based on % spread

def fillRating(num):
    rlist = [1.0, 2.0, 3.0, 4.0, 5.0]
    if num not in rlist:
        random = np.random.randint(low=1, high=101)
        if random<=11:
            return 1.0
        elif random>11 and random<=52:
            return 2.0
        elif random>52 and random<=89:
            return 3.0
        elif random>89 and random<=99:
            return 4.0
        else:
            return 5.0

    return num

lego.review_difficulty = lego['review_difficulty'].map(fillRating)
```

star_rating : Mean imputation was used to fill the missing values

```
lego["star_rating"].fillna(lego.star_rating.mean(), inplace = True)
```

theme_name : Since there are only 3 missing fields we can fill it with most occurring theme name

```
lego['theme_name'].fillna('Star Wars™', inplace=True)
```

Using these methods, we have maintained our entire dataset.

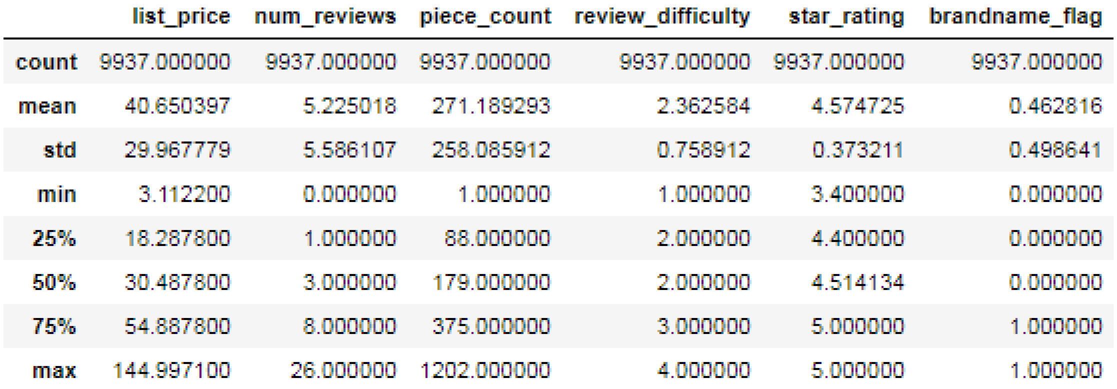
4.4 Investigating & Removing Outliers

The Lego dataset contained a significant amount of outliers that needed to be removed prior to beginning analysis. This was done using the interquartile range or IQR (also referred to as the midspread). This method determines variability within a dataset. It is done by dividing the dataset into quartiles and measuring the statistical dispersion (Taylor 2018). Once the IQR is calculated, the rule of thumb is to multiply that number by 1.5, add it to the third quartile and subtract it from the first quartile. Any number outside of this range is considered an outlier. We were still able to maintain ~10k rows of data once these outliers were removed. The code is shown below.

```
from scipy import stats
Q1 = lego.quantile(0.25)
Q3 = lego.quantile(0.75)
IQR = Q3 - Q1
lego = lego[~((lego < (Q1 - 1.5 * IQR)) |(lego > (Q3 + 1.5 *
IQR))).any(axis=1)]
```

4.5 Data attributes summary

Below is the statistical summary of our dataset after data has been cleaned and features deemed not to be valuable have been removed. These are basic statistical details for each feature in the dataset.



	list_price	num_reviews	piece_count	review_difficulty	star_rating	brandname_flag
count	9937.000000	9937.000000	9937.000000	9937.000000	9937.000000	9937.000000
mean	40.650397	5.225018	271.189293	2.362584	4.574725	0.462816
std	29.967779	5.586107	258.085912	0.758912	0.373211	0.498641
min	3.112200	0.000000	1.000000	1.000000	3.400000	0.000000
25%	18.287800	1.000000	88.000000	2.000000	4.400000	0.000000
50%	30.487800	3.000000	179.000000	2.000000	4.514134	0.000000
75%	54.887800	8.000000	375.000000	3.000000	5.000000	1.000000
max	144.997100	26.000000	1202.000000	4.000000	5.000000	1.000000

Table 2: Lego Dataset Summary

5.0 Data Exploration

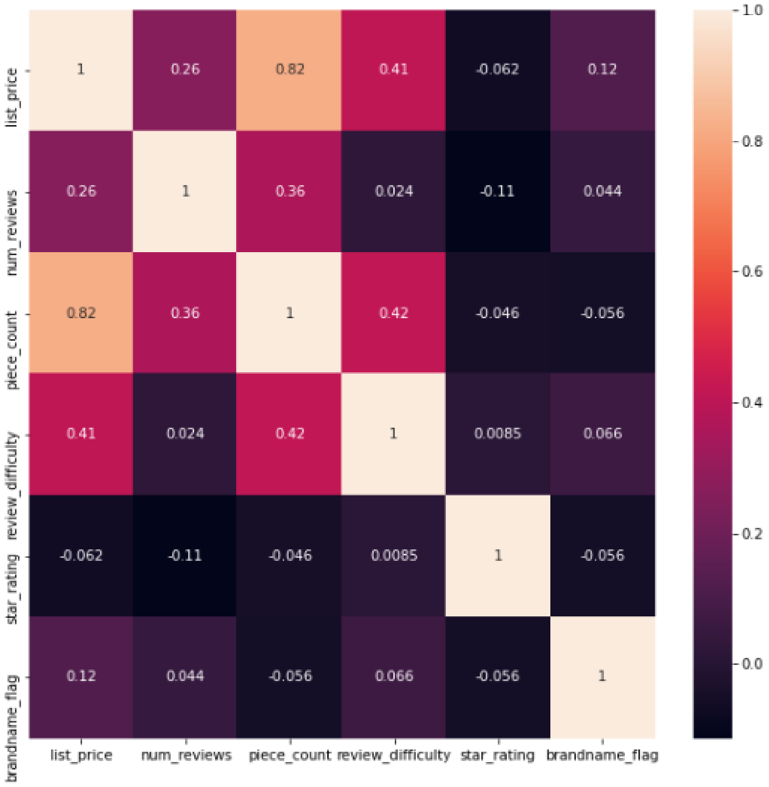
Now that the data has been cleaned, we will begin our data exploration.

5.1 Heatmap - Correlation Between Variables

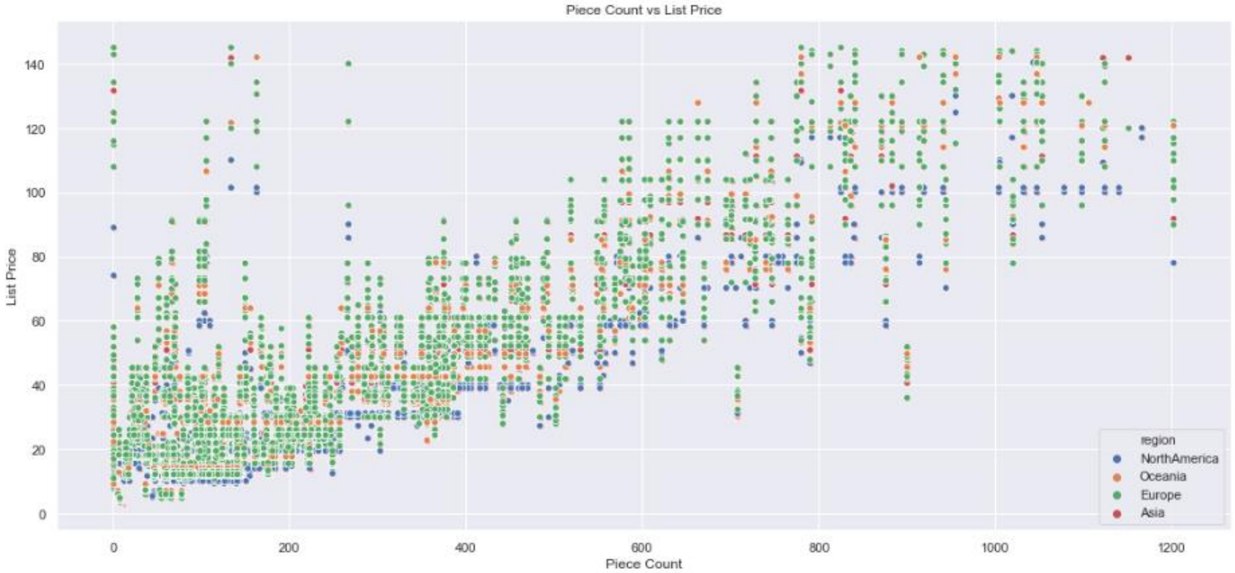
Upon review of the below heatmap, we note the following observations:

- There is a very strong correlation between list price & piece count
- There is a fairly strong correlation between review difficulty & list price
- There is a fairly strong correlation between review difficulty & piece count
- There is an above average correlation between number of reviews & review difficulty
- There is an above average correlation between number of reviews & list price

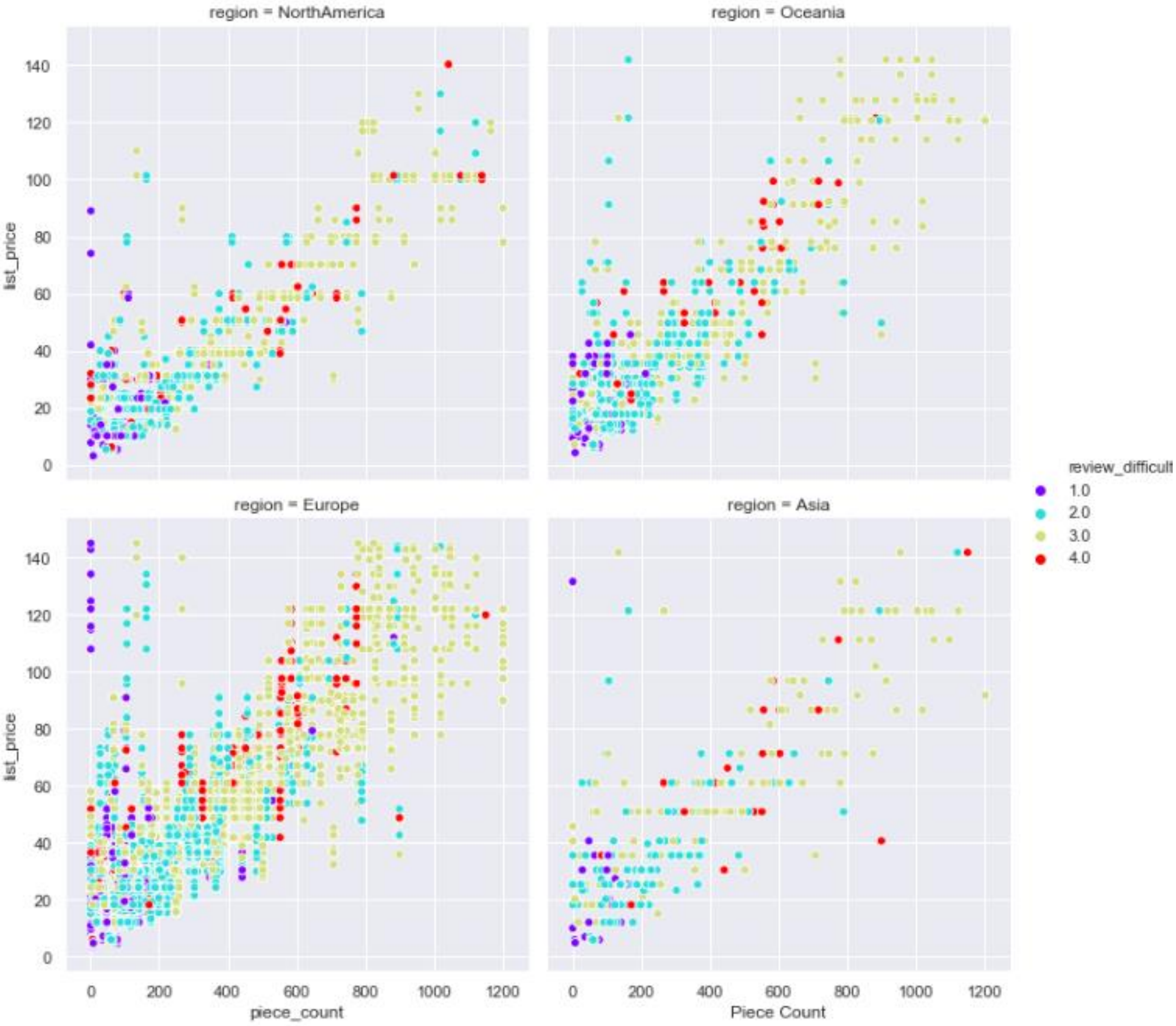
From this we can conclude that piece count is the greatest factor when determining price. The difficulty level also has a fairly strong impact on price, that is, higher difficulty levels (which tend to have higher piece counts) are typically sold for a higher price. These sets also tend to have more reviews. Surprisingly, star rating is not correlated with any other variables.



5.2 Piece Count vs List Price

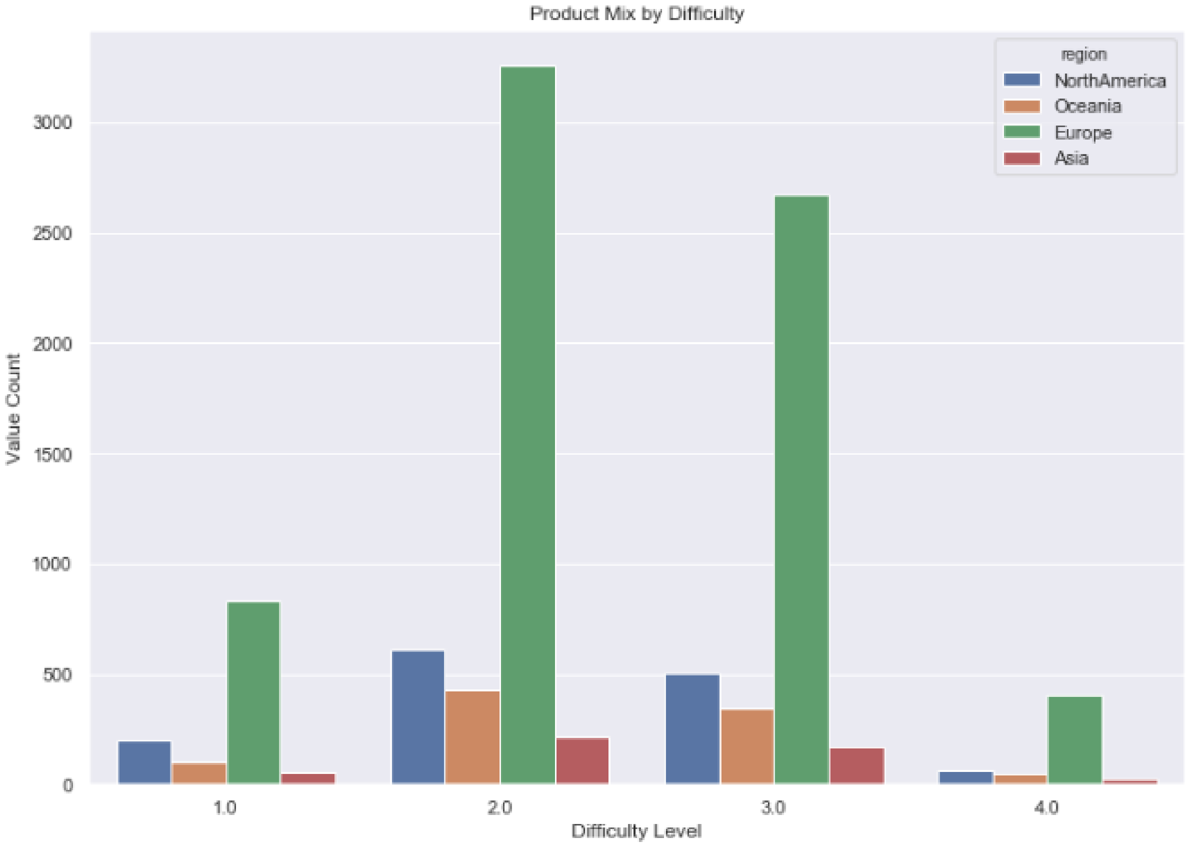


We plotted the relationship between list price and our strongest predictor, piece count. We can see above that there is a strong correlation between the two. As the piece count increases, the list price does the same. As Lego is creating an 880 piece set for their 88th anniversary, although it will vary by continent, we can visually see the price should fall between $40 - $140 USD.

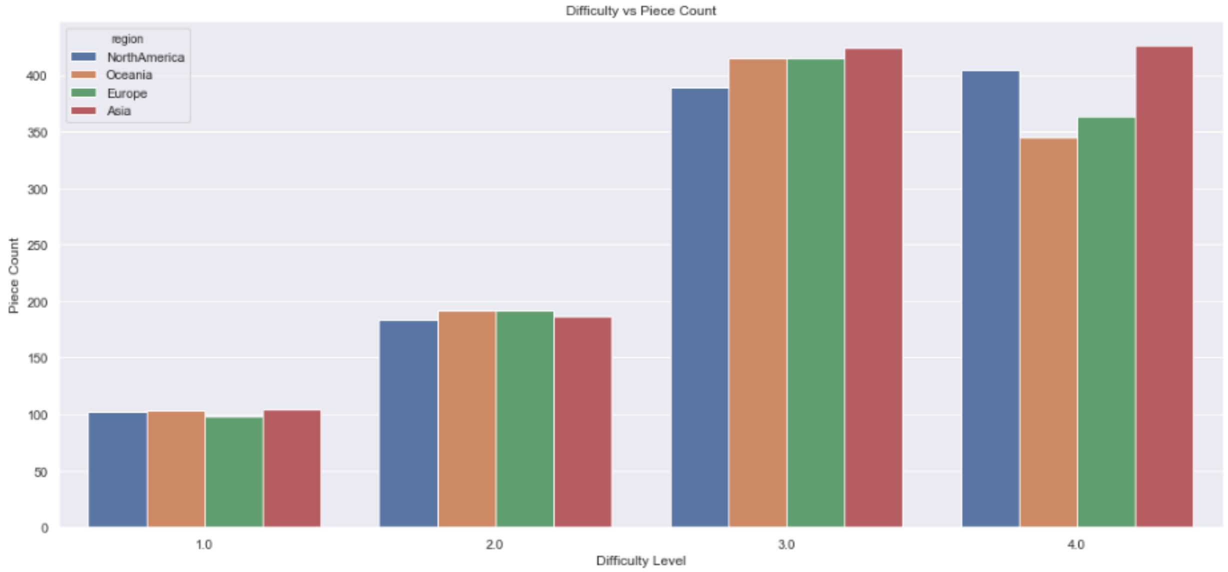


We can see the variation amongst each region in the graph above, which shows list price vs. piece count by difficulty level. If we are focusing on a set with 880 pieces, we can see that this would fall into the “Average” difficulty level in every region.

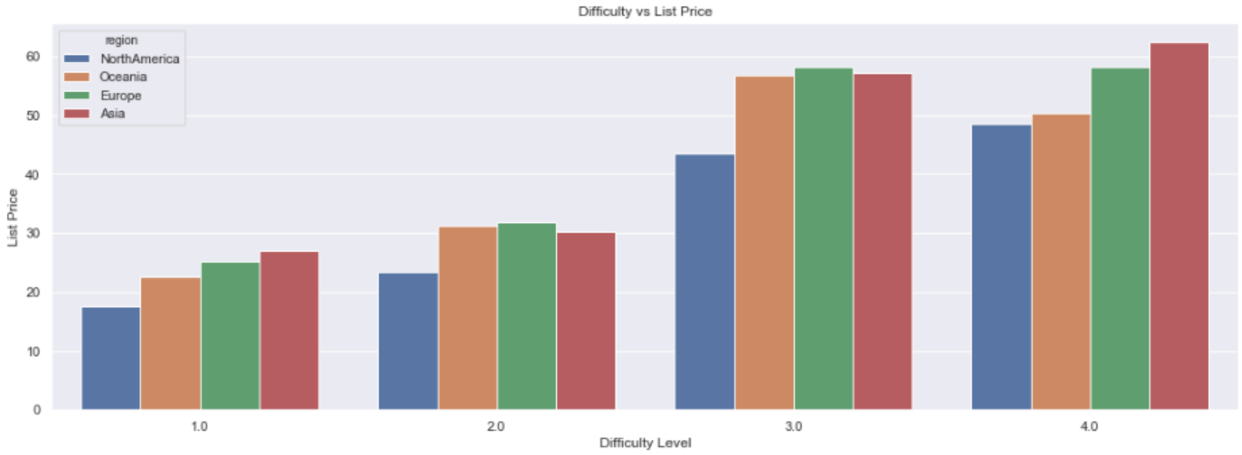
5.3 Exploring Difficulty Levels



The above graph shows the distribution amongst the various difficulty levels available for Lego sets, by each geographic region. We can see that difficulty level 2 (Easy) has the highest value counts, followed by difficulty level 3 (Average), meaning these Lego sets are the most widely available around the globe. Difficulty level 1 (Very Easy) and level 4 (Challenging) have lower value counts, showing they have less of a global presence. *Note: The ‘Very Challenging’ difficulty level is not seen in the graph above. There were only 8 ‘Very Challenging’ Lego sets in our data to work with, and these values were removed in our outlier removal process.*

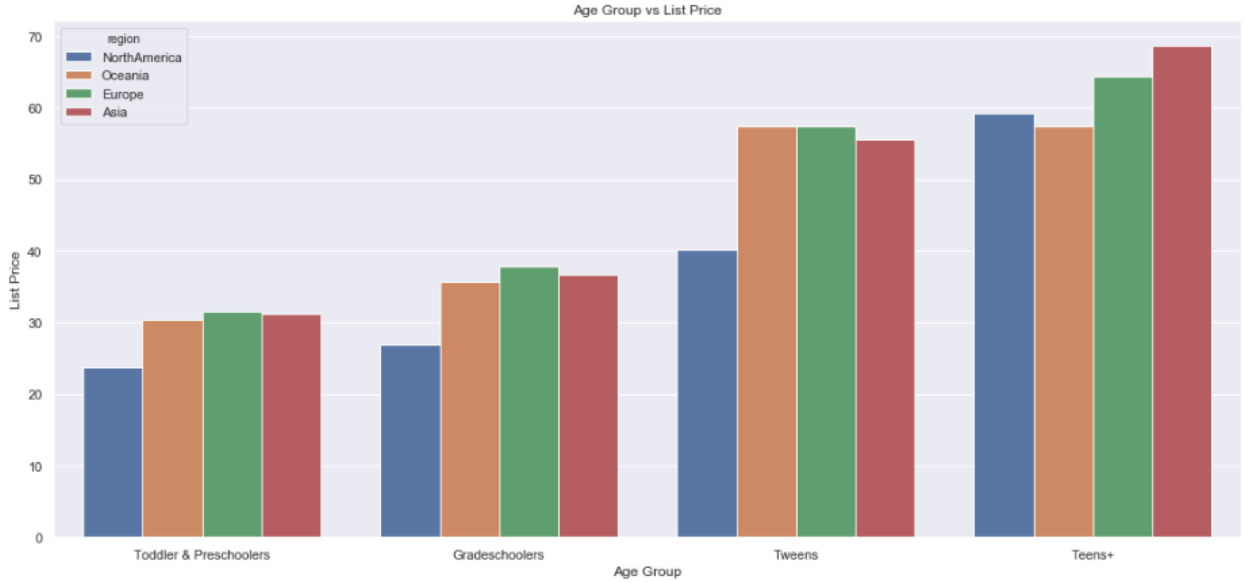


Here we graphed the average piece count for each difficulty level by region. It is evident that as the piece count of Lego sets rise, the difficulty level of the set follows. This pattern is visible across each geographic region. The average piece count for difficulty levels 1-3 (Very Easy to Average) is also fairly similar across each region. This trend continues in North America and Asia for Challenging Lego sets (level 4). However, we were surprised to see a dip in the average piece count for these sets, in Oceania and Europe, and a slightly higher average piece count in Asia.



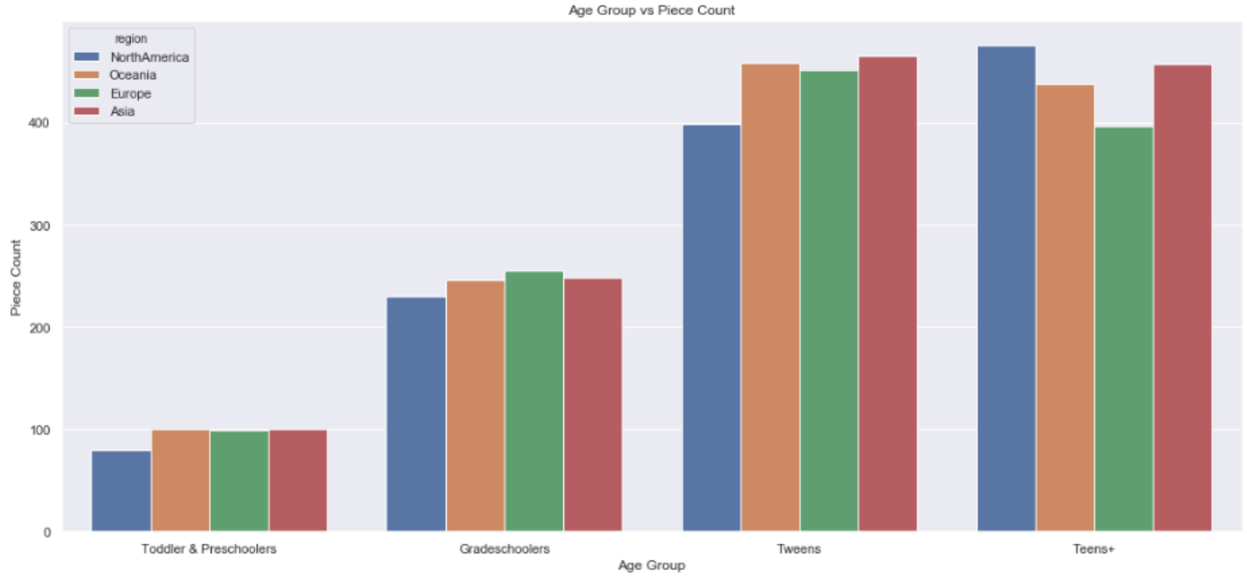
The above graph shows the average list price for each difficulty level by region. Across all difficulty levels, North America tends to have the lowest priced sets for each difficulty level. Across difficulties 1-3 (Very Easy to Average), Europe, Asia and Oceania tend to have fairly similar priced sets. If we look at the Challenging sets on the far right of the graph, sets in the Oceania region surprisingly have a drop in list price as compared to the Average sets shown.

5.4 Exploring Age Groups



Above we can see which age group has the highest average price in each continent. List price increases as age increases, with one exception. For Oceania, we see list price drops slightly from Tweens to Teens+. We were surprised by this finding because Teens+ have higher piece counts than Tweens Lego sets.

The graph provides valuable consumer insights. We can see Oceania has a significant spike in price from Gradeschoolers to Tweens age group. Oceania and Europe absorb the highest list price in the Tweens category. Whereas North America has the lowest list price for Tweens. Asia has the highest average list price for Teens+ sets.

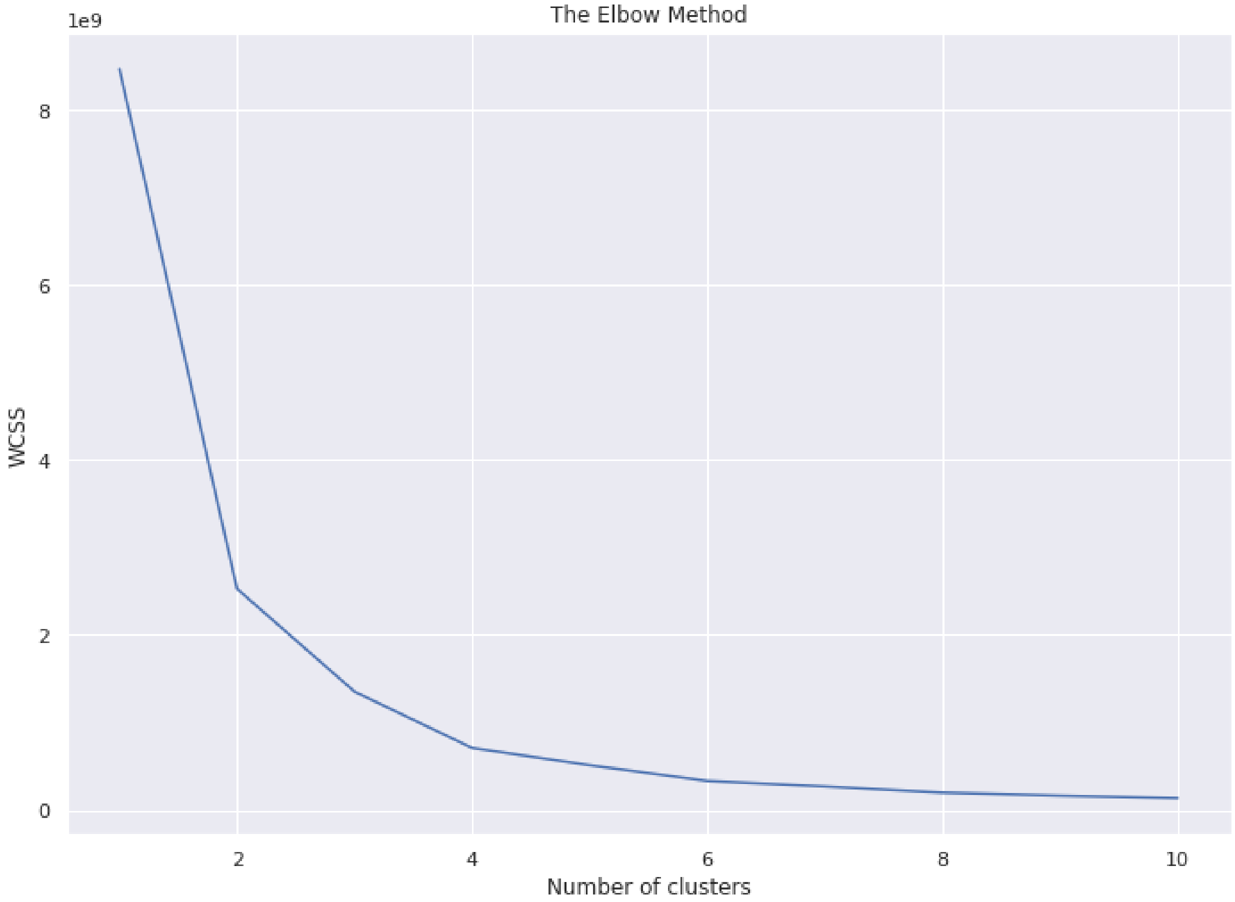


Across all regions, piece count steadily increases from age groups Toddler & Preschoolers to Tweens. The trend continues for Teens+ in North America, however, we see piece count drop in Oceania, Europe and Asia for Teens+.

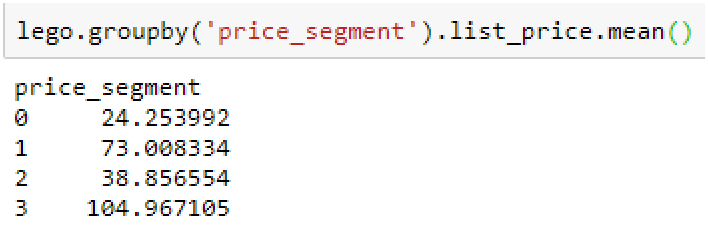
We can see similar correlations amongst each region in the graphs above, which displays list price vs. piece by age group. Since we are focusing on an 880-piece set with a $40 - $140 USD price range, we can see this falls under the Tweens age group for North America, Oceania and Asia. Whereas in Europe, the new Lego set should be marketed for Gradeschoolers and Tweens.

6.0 Modelling

6.1 Clustering Modelling



The first step in our modelling was to find out how many clusters we needed to implement. In order to find this out we employed “The Elbow Method”. Using this we can see there is a sharp drop between 0-2 clusters. This indicates we need more than 2 clusters. Moving to 3 clusters, we can see that there is still a somewhat significant slope between 3 clusters and 4 clusters. Thus, we ended up selecting 4 clusters, as after four clusters the slope seems to not change in a drastic way, and if we were to add additional clusters it would not make the model significantly better.

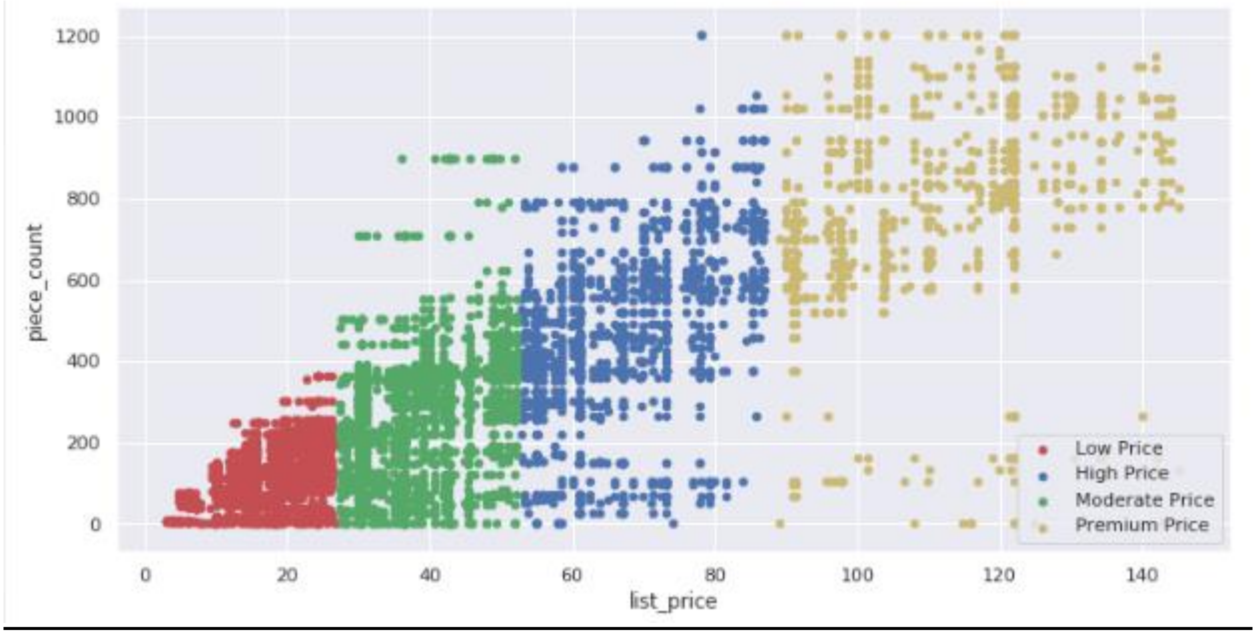


```
lego.groupby('price_segment').list_price.mean()
```

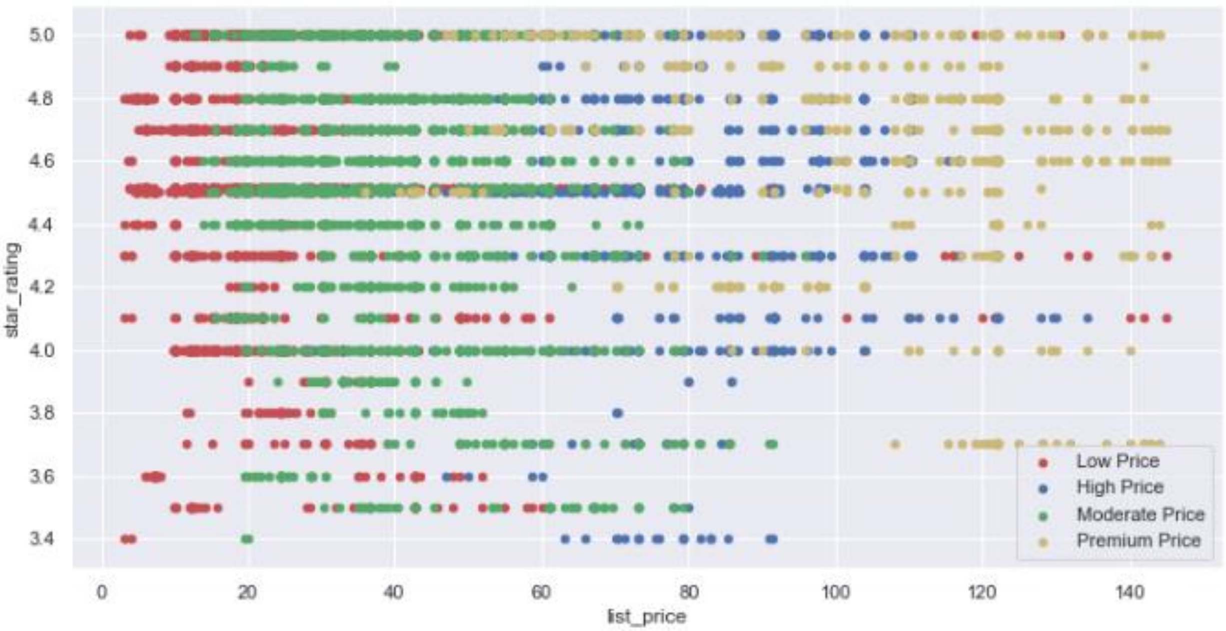
```
price_segment
0     24.253992
1     73.008334
2     38.856554
3    104.967105
```

The next step was to determine what our clusters represented. When we grouped the means of our clusters, we discovered 4 significantly different price ranges in regards to prices. The resulting average prices were different enough that we felt comfortable classifying them as : ‘Low Price’ for cluster 0, ‘High Price’ for cluster 1, ‘Moderate Price’ for cluster 2, and ‘Premium Price’ for cluster 3.

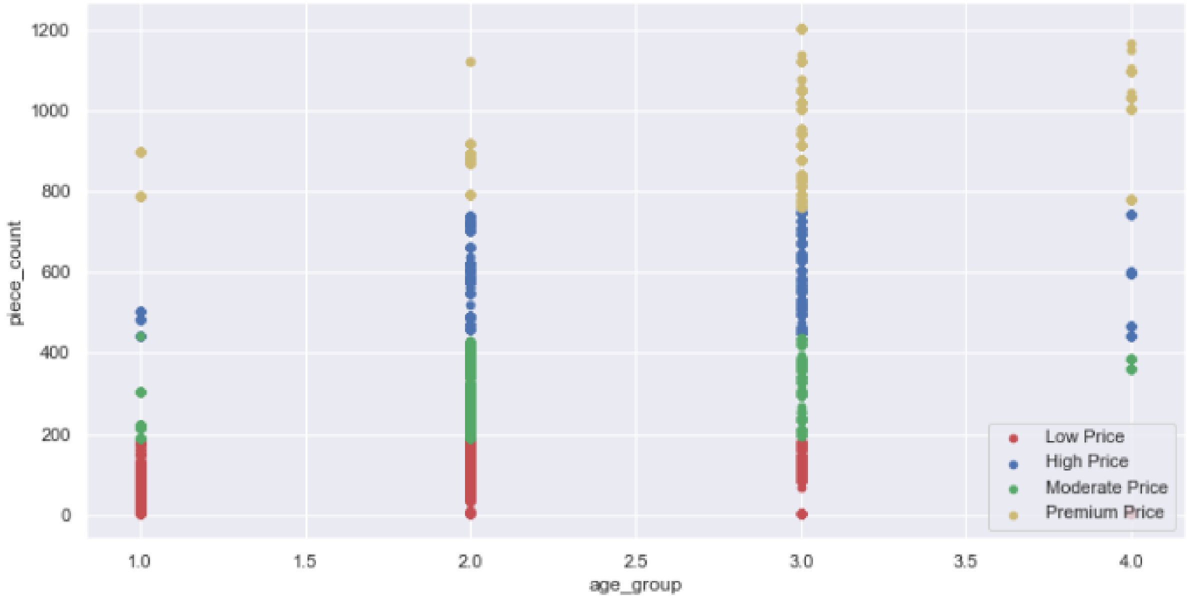
We then ran various clustering models shown below.



We can see in the graph above that the low priced lego sets typically have less than 400 pieces and have an average price of roughly $24. The next segment has an escalation in both the price range and piece count. We can see significantly more sets in the $30-$60 price range sets have up to 900 pieces. Moving to the high priced segment we can see the majority of the sets cost over $60 and have up to 1,200 pieces. The final segment , the ‘Premium’ sets tend to have more than 600 pieces and on average cost over $100. This grouping also has the widest list price spread, from just under $90 to more than $140.



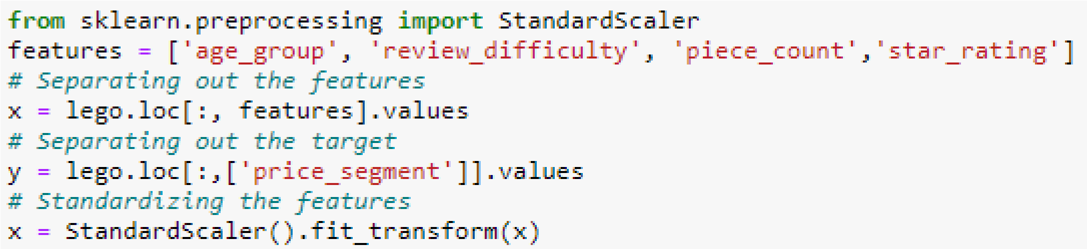
The first thing that is evident is that lego is reviewed well. Across price segments all Lego sets have a significant number of reviews that are over 4 out of 5 stars. Looking closer at the graph though it does reveal some interesting trends, while all lego sets score well, as you go up in price segment there is less dispersion amongst the reviews for the category. For example the premium sets have their lowest scores at 3.7, whereas Low, Moderate & High have 3.4 as their lowest star ratings. So while all lego sets can get great reviews, those in the higher segments have a smaller range of potential scores. This in essence shows that while all lego price segments have the ability or potential to achieve great reviews, if you produce a premium set you have a much greater ability to guarantee a minimum review of 4 out of 5 stars.



After determining Lego should market their new 880-piece anniversary set to Tweens (age_group 3.0) in North America, Oceanic and Asia, and to Gradeschoolers (age_group 2.0) & Tweens in Europe, we are pleased to see that we can charge the new set at a premium price in every continent.

6.2 Principal Component Analysis (PCA)

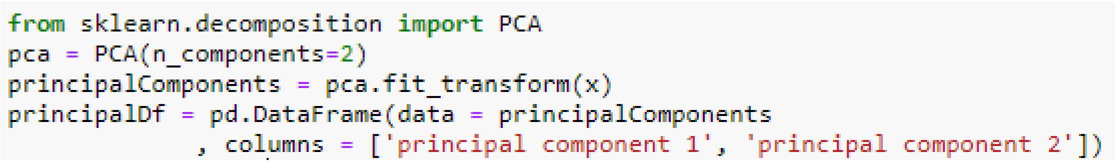
To further understand our data, we used the unsupervised learning, dimensionality reduction algorithm. We started with 4 dimensions; age group (converted to numeric values 1-4), piece count, review difficulty and star rating with a goal of reducing these to 2 dimensions. Since PCA is affected by scale, we scaled the features in the dataset prior to the application of PCA using standard scaler.



```
from sklearn.preprocessing import StandardScaler
features = ['age_group', 'review_difficulty', 'piece_count','star_rating']
# Separating out the features
x = lego.loc[:, features].values
# Separating out the target
y = lego.loc[:,['price_segment']].values
# Standardizing the features
x = StandardScaler().fit_transform(x)
```

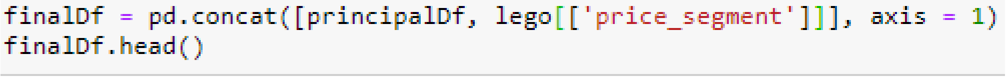
.

We then project the dataset from 4 dimensional to 2 dimensional. PC1 and PC2 are the main dimensions of variation.

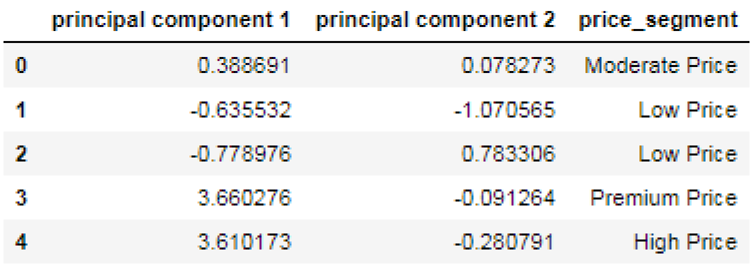


```
from sklearn.decomposition import PCA
pca = PCA(n_components=2)
principalComponents = pca.fit_transform(x)
principalDf = pd.DataFrame(data = principalComponents
             , columns = ['principal component 1', 'principal component 2'])
```

We then combine these 2 principal components with the target (price segment) for our final dataframe.

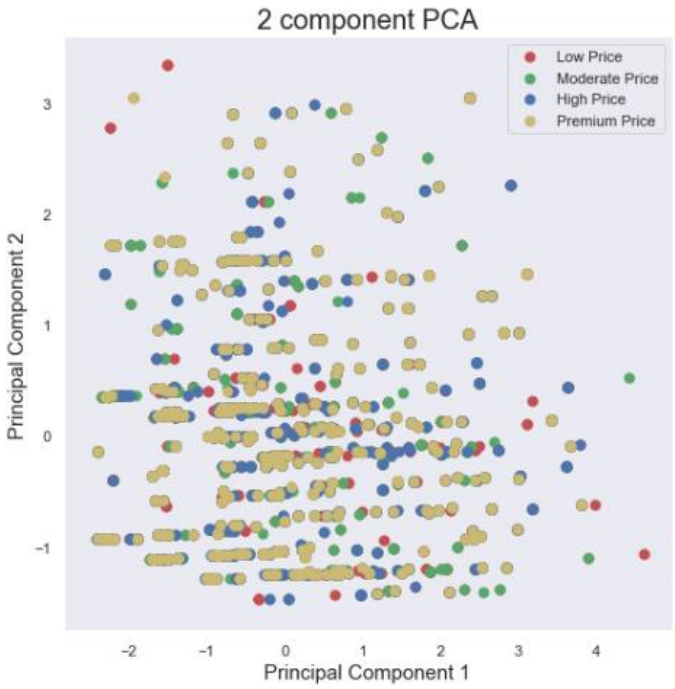


```
finalDf = pd.concat([principalDf, lego[['price_segment']]], axis = 1)
finalDf.head()
```



	principal component 1	principal component 2	price_segment
0	0.388691	0.078273	Moderate Price
1	-0.635532	-1.070565	Low Price
2	-0.778976	0.783306	Low Price
3	3.660276	-0.091264	Premium Price
4	3.610173	-0.280791	High Price

We can see the results below.



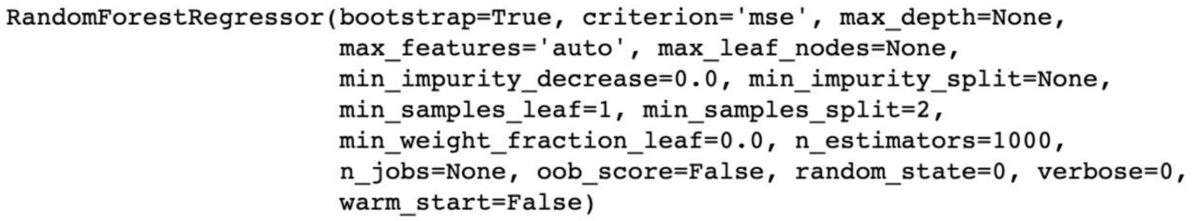
When we moved from 4 dimensional to 2 dimensional data, some information was lost. We can see this using the explained variance ratio. These 2 principal components only take up 68.35% of the variance/information with the first accounting for 43.3%, and the second 25.1%. As a rule of thumb, if the explained variance is below 85% we've lost to much data to move forward with PCA. We will now move onto the regression analysis.

6.3 Random Forest Regression Modelling

Part of creating and marketing our 88th year limited edition set, is to know what Lego should price the set at. First, we split the data set into the 4 regions and ran regression on each one. The code & output shown below is for the Europe region (note all had similar error scores). Since price, our target variable, is numerical we will have to perform a linear regression to determine what the set’s price should be. In order to create our model we need to import the linear regression model class, and train the model. We split 80% of the data to the training set while 20% of the data to test set to run Random Forest Regression on our dataset, we used the following code.

```
X = df.drop(['list_price'], axis=1)
y = df.list_price
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2

rf = RandomForestRegressor(n_estimators = 1000, random_state = 0)
rf.fit(X_train, y_train)
```

Output:

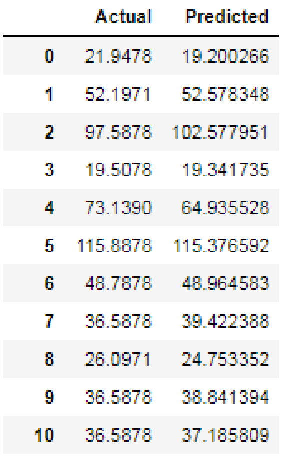


```
RandomForestRegressor(bootstrap=True, criterion='mse', max_depth=None,
                      max_features='auto', max_leaf_nodes=None,
                      min_impurity_decrease=0.0, min_impurity_split=None,
                      min_samples_leaf=1, min_samples_split=2,
                      min_weight_fraction_leaf=0.0, n_estimators=1000,
                      n_jobs=None, oob_score=False, random_state=0, verbose=0,
                      warm_start=False)
```

```
y_pred = rf.predict(X_test)
```

Checking Actuals Vs Predicted

```
df_output = pd.DataFrame({'Actual':y_test.ravel(), 'Predicted':y_pred})
df_output
```

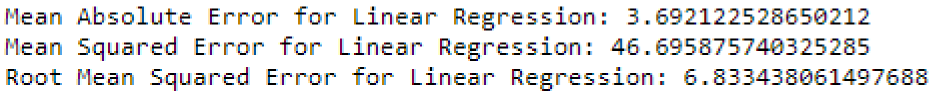


	Actual	Predicted
0	21.9478	19.200266
1	52.1971	52.578348
2	97.5878	102.577951
3	19.5078	19.341735
4	73.1390	64.935528
5	115.8878	115.376592
6	48.7878	48.964583
7	36.5878	39.422388
8	26.0971	24.753352
9	36.5878	38.841394
10	36.5878	37.185809

Mean Absolute Error, Mean Squared Error and Root Mean Squared Error
Once our model was created, it needs to be tested to find the accuracy of the results that it generates. This is done through establishing the R-Score , as well as, determining the model’s Mean Absolute Error, Mean Squared Error and Root Mean Squared Error

```
mae = mean_absolute_error(y_test, y_pred)
mse = mean_squared_error(y_test, y_pred)
rmse = np.sqrt(mean_squared_error(y_test, y_pred))

print("Mean Absolute Error for Linear Regression: {}".format(mae))
print("Mean Squared Error for Linear Regression: {}".format(mse))
print("Root Mean Squared Error for Linear Regression: {}".format(rmse))
```

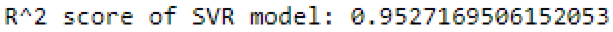
Output:



```
Mean Absolute Error for Linear Regression: 3.692122528650212
Mean Squared Error for Linear Regression: 46.695875740325285
Root Mean Squared Error for Linear Regression: 6.833438061497688
```

```
r_score = r2_score(y_test,y_pred)
print("R^2 score of SVR model: {}".format(r_score))
```

Output:

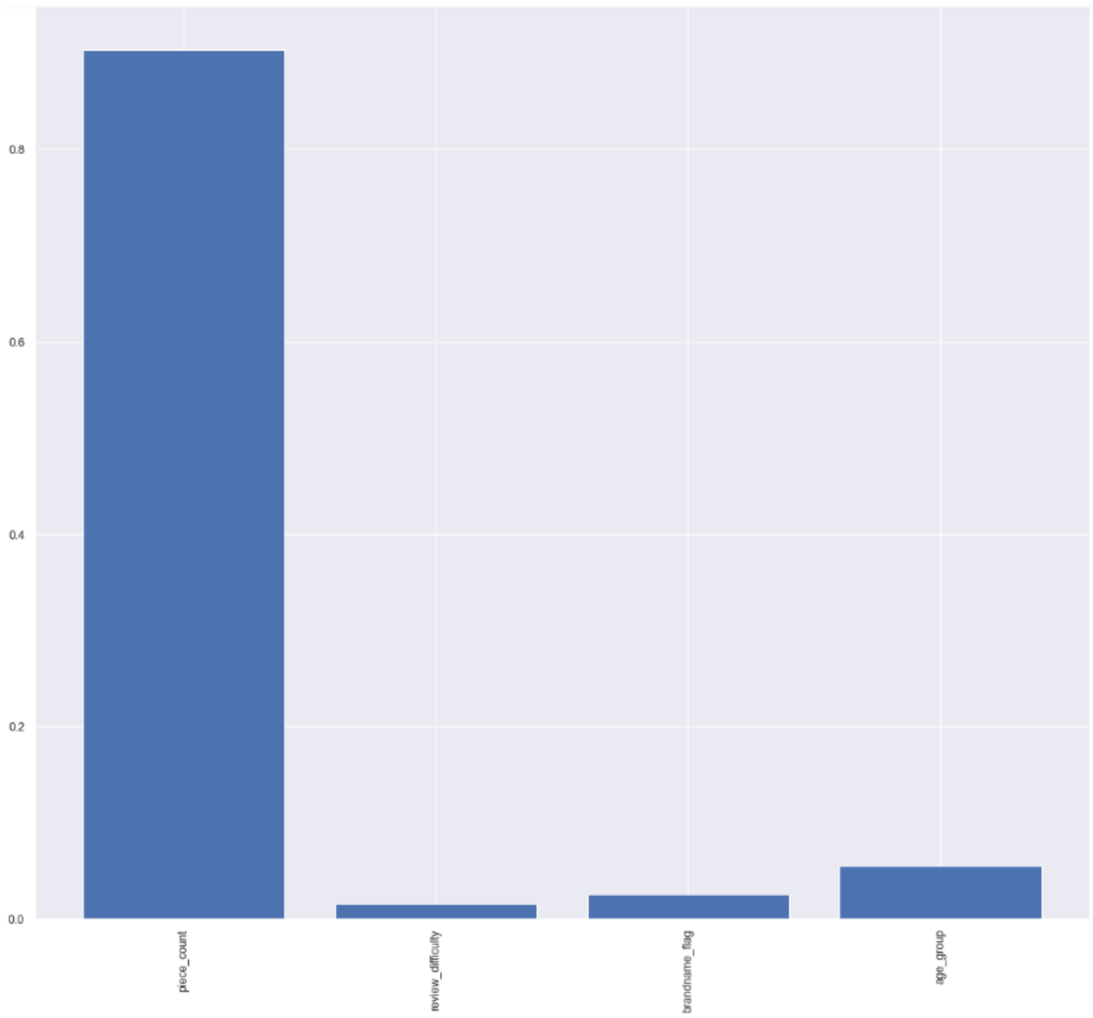


```
R^2 score of SVR model: 0.9527169506152053
```

Analysis of Feature Importance

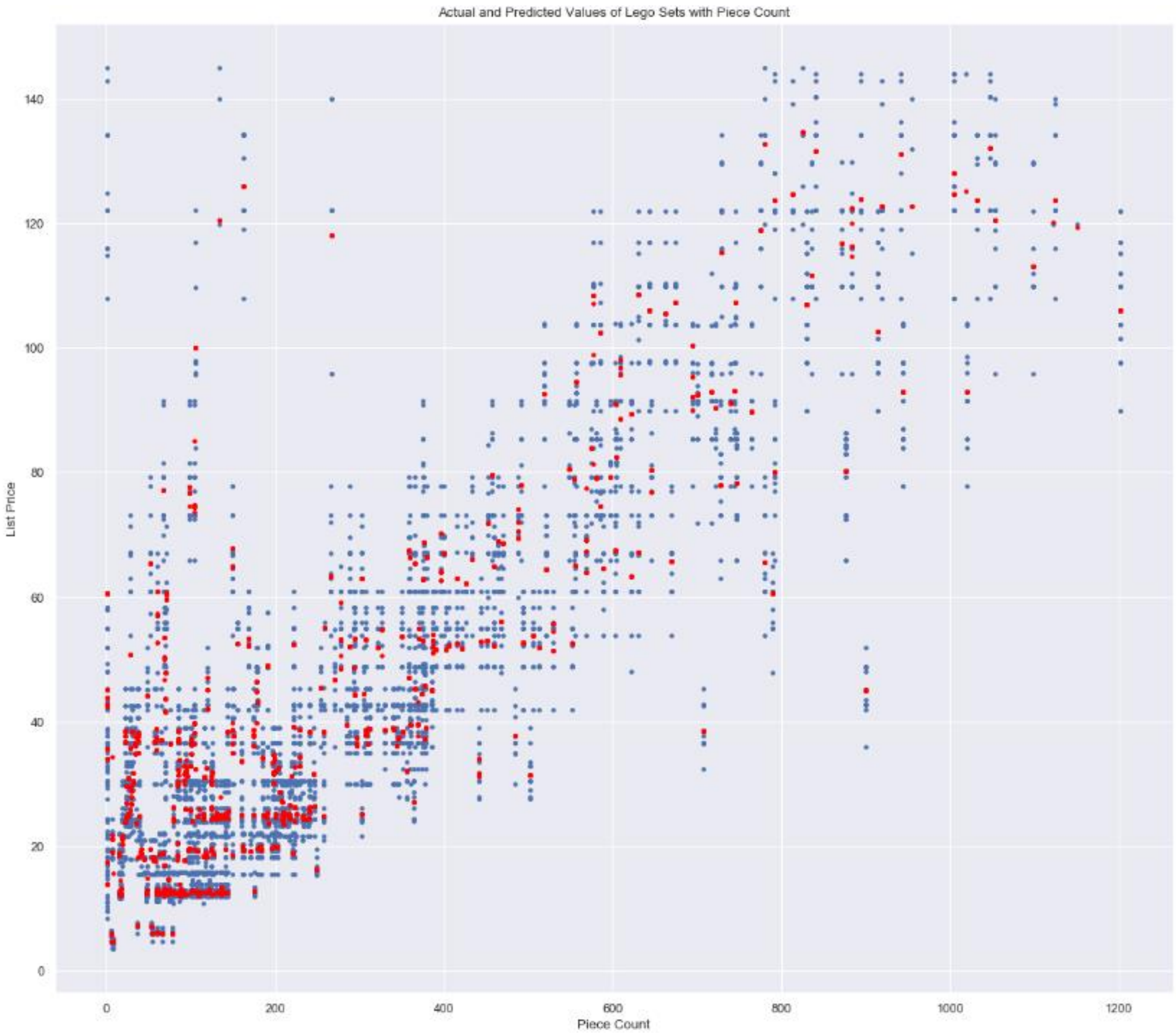
Next, we plan to visualize the importance of the features in our dataset using a plot. To do so, we ran the code below.

```
importances = list(rf.feature_importances_)
cols = X.columns.values

importance_cols = pd.DataFrame({'Column':cols, 'Importance':importances})

plt.figure(figsize=(18, 16), dpi= 80)
plt.bar(importance_cols.Column, importance_cols.Importance, orientation =
'vertical')
plt.xticks(rotation=90)
plt.show()
```



It is very evident that the piece count is driving the price of Lego sets higher or lower. Now let’s visualize the actual values and predicted values of Lego sets with piece count.

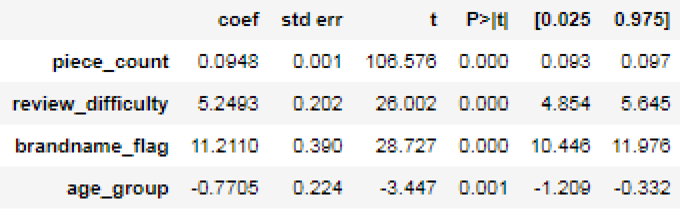
```
y_predict = rf.predict(X)

plt.figure(figsize=(18, 16), dpi= 80)
plt.scatter(X['piece_count'], y, s=10, label = "Actual")
plt.scatter(X['piece_count'], y_predict, c="red", s=10, label = "Actual")
plt.title('Actual and Predicted Values of Lego Sets with Piece Count')
plt.xlabel('Piece Count')
plt.ylabel('List Price')
plt.show()
```



List Price Output (Determined by Regression Modelling)

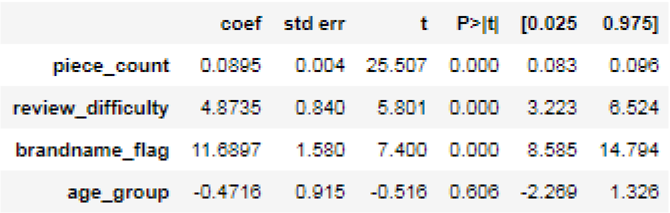
Using our regression model outputs and correlation coefficients, we can create a formula for each region that can be used to determine list price.

Europe: Price = 0.0948xpiece count + 5.2493xdifficulty+11.211xbrand flag-0.7705xage group



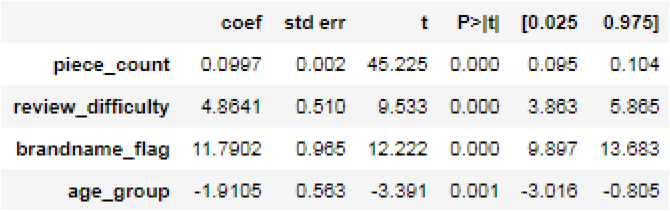
	coef	std err	t	P>|t|	[0.025	0.975]
piece_count	0.0948	0.001	106.576	0.000	0.093	0.097
review_difficulty	5.2493	0.202	26.002	0.000	4.854	5.645
brandname_flag	11.2110	0.390	28.727	0.000	10.446	11.976
age_group	-0.7705	0.224	-3.447	0.001	-1.209	-0.332

Asia: Price = 0.0895xpiece count + 4.8735xdifficulty+11.6897xbrand flag-0.4716xage group



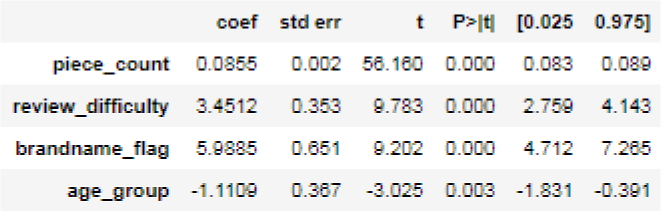
	coef	std err	t	P>|t|	[0.025	0.975]
piece_count	0.0895	0.004	25.507	0.000	0.083	0.096
review_difficulty	4.8735	0.840	5.801	0.000	3.223	6.524
brandname_flag	11.6897	1.580	7.400	0.000	8.585	14.794
age_group	-0.4716	0.915	-0.516	0.606	-2.269	1.326

Oceania: Price = 0.0997xpiece count + 4.8641xdifficulty+11.7902xbrand flag-1.9105xage group



	coef	std err	t	P>|t|	[0.025	0.975]
piece_count	0.0997	0.002	45.225	0.000	0.095	0.104
review_difficulty	4.8641	0.510	9.533	0.000	3.863	5.865
brandname_flag	11.7902	0.965	12.222	0.000	9.897	13.683
age_group	-1.9105	0.563	-3.391	0.001	-3.016	-0.805

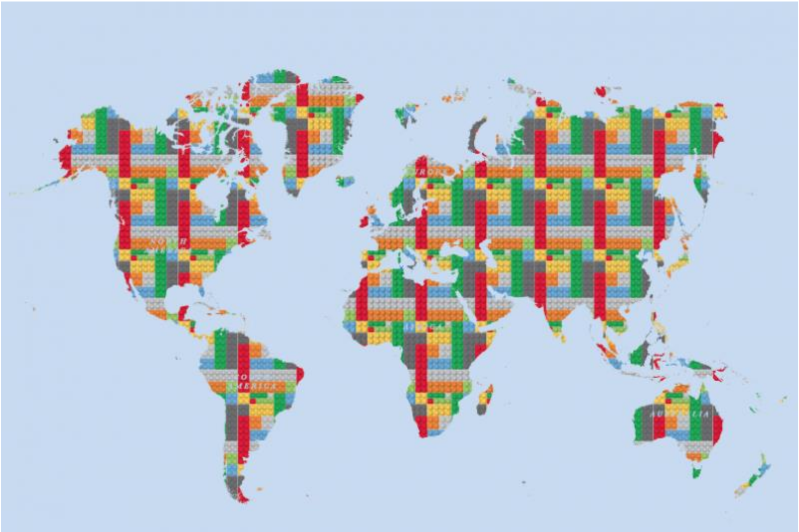
North America: Price = 0.0855xpiece count + 3.4512xdifficulty+5.9885xbrand flag-1.1109xage group



	coef	std err	t	P>|t|	[0.025	0.975]
piece_count	0.0855	0.002	56.160	0.000	0.083	0.089
review_difficulty	3.4512	0.353	9.783	0.000	2.759	4.143
brandname_flag	5.9885	0.651	9.202	0.000	4.712	7.265
age_group	-1.1109	0.367	-3.025	0.003	-1.831	-0.391

Recommendation & Conclusion

After conducting a detailed, in-depth analysis, we have identified a launch strategy for Lego’s 88th anniversary, 880-piece Lego set in each of the continents; North America, Oceania, Asia & Europe. The following is what we have identified as the marketing strategy for Lego to focus on in the specific markets.



Below please find the specific details of our recommendation:

i. Difficulty Level

Lego wanted to create a lego set for their 88th anniversary with 880 pieces. Based on that assumption through our analysis, we have determined that the set should fall in the **Average** difficulty level. The average difficulty will be the complexity level across all regions. As the difficulty of a set is strongly correlated to the piece count we would not be able to recommend a different difficulty level without changing the set’s piece count. As one of Lego’s prerequisites was for the set to contain 880 pieces, the set needs to be of an average difficulty.

ii. Age Group

In terms of ages marketed to, there is a small variation across regions.

North America: *Tweens*
Asia: *Tweens*
Oceania: *Tweens*

In North America, Asia, and Oceania we will be marketing the new Lego set to Tweens. Based on our analysis, an 880-piece Lego set would appeal to this age more than other demographics in these regions.

Europe: *Gradeschoolers & Tweens*

In Europe, we will be marketing to both Grade Schoolers and Tweens. We found additional opportunity to target Gradeschoolers in Europe because this age group has a higher piece count demand compared to the other continents.

In addition, because Lego is celebrating a special anniversary and the company originated in Europe, we recommend expanding the target markets in Europe by selling Lego’s 88th anniversary set to both age groups.

If possible, we would recommend marketing to every age segment, but as we know, marketing budgets are finite, we have identified these as the “best” markets to focus on as they would allow the marketing department to get the value out of their “marketing dollars”.

iii. Brand Name

Across all regions there is a positive correlation with the price and the set being sold with a Brand name. This means that when the Lego partners with another company (ie.

Star Wars™, DUPLO®, NINJAGO®, the price of the set goes up. We have observed that in North America Brand Name sets can be sold for $5.99 higher than non-brand name sets. We see this trend to a greater degree in Europe, Asia & Oceania which have increases of $11.21, $11.70 & 11.79 respectively.

iv. List Price

North America:

After running our regression model, the price we will be recommending in North America will be $88.25. This is based on the set having a Brand name, a difficulty of average, and marketed to the “tweens’ age segment.

Asia:

After running our regression model, the price we will be recommending in Asia will be $103.66.This is based on the set having a Brand name, a difficulty of average, and marketed to the “tweens’ age segment.

Europe:

After running our regression model, the price we will be recommending in Europe will be $106.53. This is based on the set having a Brand name, a difficulty of average, and marketed to both the “tweens’ and ‘gradeschooler’ age segments.

Oceania:

After running our regression model, the price we will be recommending in Oceania will be $108.89. This is based on the set having a Brand name, a difficulty of average, and marketed to the “tweens’ age segment.

Works Cited

Terzolo, Mattie. “Lego Sets.” *Kaggle*, 18 May 2018, https://www.kaggle.com/mterzolo/Lego-sets.

“LEGO facts.” *National Geographic*, 17 August 2011, https://www.nationalgeographic.com.au/history/lego-facts.aspx.

Foster, E. ”LEGO revenue increases 4% in fiscal 2018.” *Kidscreen*, 27 February 2019, http://kidscreen.com/2019/02/27/lego-revenue-increases-4-in-fiscal-2018/.

Bhasin, H. “Top 9 Lego Competitors - Competitor analysis of Lego.” *Marketing91* 14 June 2018, https://www.marketing91.com/lego-competitors/.

“Mattel Announces Full Year And Fourth Quarter 2018 Financial Results Conference Call And 2019 Toy Fair Analyst Meeting.” *MarketWatch*, 24 January 2019, https://www.marketwatch.com/press-release/mattel-announces-full-year-and-fourth-quarter-2018-financial-results-conference-call-and-2019-toy-fair-analyst-meeting-2019-01-24-1818300.

“BANDAI NAMCO Holdings Inc.” *Nikkei Asia Review*, 15 January 2020, https://asia.nikkei.com/Companies/BANDAI-NAMCO-Holdings-Inc.

“Lego's Solid Foundations Stack up as World's Most Valuable Toy Brand.” *Brand Finance*, June 2019, https://brandfinance.com/press-releases/legos-solid-foundations-stack-up-as-worlds-most-valuable-toy-brand/.

McKenna, B. “Hasbro Earnings Once Again Decline on Toys R Us Woes.” *The Motley Fool*, 10 February 2019, https://www.fool.com/investing/2019/02/10/hasbro-earnings-once-again-decline-on-toys-r-us-wo.aspx.

Taylor, Courtney. “Detect the Presence of Outliers With the Interquartile Range Rule.” ThoughtCo, ThoughtCo, 27 Apr. 2018, www.thoughtco.com/what-is-the-interquartile-range-rule-3126244.

“The LEGO Group Annual Report 2018.” *Lego*, https://www.lego.com/cdn/cs/aboutus/assets/blt02144956ae00afa1/Annual_Report_2018_ENG.pdf.

Ghoneim, S. “5 Types of bias & how to eliminate them in your machine learning project.” *Towards Data Science*, 16 April 2019 https://towardsdatascience.com/5-types-of-bias-how-to-eliminate-them-in-your-machine-learning-project-75959af9d3a0.