# Exploratory Data Analytics for Machine Learning – Peer-graded Assignment – Lego sets

Throughout this report, we reference work that was carried out using Python in a Jupyter notebook. This, as well as the data used, can be accessed with the link below and should be reviewed alongside this report:

<https://github.com/warrenoates1/Coursera_data_science_and_machine_learning/blob/master/Exploratory_Data_Analysis_for_MI_-_Peer-graded_Assignment.ipynb>

## The Data set

The data source is the Brickset database, which maintains a record of all released Lego sets and related products (<https://brickset.com/sets/category-Normal>).

The data set we are using contains a list of standard released Lego sets, tagged as ‘normal’ in the database, from the years 2011 to 2020. It does not include books, merchandise, magazines and other Lego branded products.

Data for 4722 sets are included. The attributes include the unique identifiers of set ID, set number, variant ID. The theme and sub-theme of the set are included, alongside the set title and year of release.

Counts for the number of minifigures and number of pieces are included, as well as the currency prices in the UK, US, Canada and Euro-zone.

Data on the number of members on the Bricket set website who have flagged the set as one they own, or one they want is also included.

Finally, some categorisation I have personally carried out previously is included to flag if a set is under media license with another company. These flags include Disney, Marvel, Star Wars, Warner Bros, Universal and Other / unknown. This does not included licenses with non-media companies, such as NASA.

## Data exploration plan

* A Jupyter notebook using Python will be used to explore our data.
* We will identify data types, including any categorical fields and ensure all can be engineered to a useful type in our data set.
* We will investigate if there are any duplicated, or unnecessary data fields and consider removing them.
* We will investigate whether any fields are blank or null, what the reasons for that may be and what we should do about it.
* We will look for outlier values, and explore the mean, median, mode, range and quartiles for our numerical data sets, to better understand them and decide what should be done.
* We will also consider grouping data by categorical attributes to better understand the data for specific categories.
* We will look at correlation of pairs of attributes, to identify if there are highly correlated attributes where we can remove one.
* We will look to see if there are non-linear relationships between attributes or whether there might be skewed data, which might suggest we need to consider transforming our data in future, depending on what we wish to look at.

## Data cleaning and feature engineering

### 3.1 Data types

The data types were reviewed, and it was discovered there were 5 integer fields, 6 float fields and 11 object fields.

The ‘WantedBy’ field was incorrectly identified as ‘object’ due to 45 rows where no one wanted the set being set to ‘No’. The ‘No’ values were changed to 0 and the field type changed to integer.

### 3.2 Unique identifiers

It was identified that SetID and Name were individual unique identifiers, while in combination Number and Variant were also unique identifiers. We removed these fields as they would add no value to our data set.

The ImageURL field was also unique to each set and would add no value, so was also removed.

### 3.3 Null values and unwanted rows

Looking for null values, it was discovered that the Minifigure columns was set to null when there were no minifigures in the set. These null values were set to zero.

A small number of rows (17) has a null entry in the ‘Other or unknown media license’ column. These should have had a value of ‘No’, so this change was made.

When the Pieces column was set to null, it was discovered the sets tended to be non-standard and promotional sets. These rows were dropped as only standard Lego sets are of interest.

Other promotional sets were also discovered under the Themes of Promotional and Miscellaneous – all these rows were also dropped.

The subtheme field was sometimes set to null when the primary Theme did not have subtheme at all, or a set did not belong to a specific subtheme. These null values were changed to ‘None’.

### 3.4 Missing price values

Finally, it was identified that a number of rows had a null value across all four Price columns. Investigation suggested these sets were mostly unusual sets or promotional sets or being standard sets that have not been formally released yet. Again these rows were dropped.

Finally, it was noted that many rows were missing prices in one, two or three of the four regions.

Given that exchange rates can change over time, it was decided to estimate the missing values based on the average exchange rate between a know values within each year.

First, the ratio between prices in two regions was worked out for each set. Then the average ratio for each year was worked out. Finally, the average value was used to work out an estimated missing value.

The approach taken started filling the remaining missing values in the region with the fewest missing first, from the next most complete region until all price values were complete for all sets in all regions.

### 3.4 Binary fields

A number of fields were identified as having binary values of ‘Yes’ and ‘No’. These values where changed with binary encoding, with ‘Yes’ becoming 1 and ‘No’ becoming 0, to give these columns numerical values.

The columns were: *Disney, Star Wars, Marvel, WB, Universal* and *Other or unknown media license.*

3.5 Grouping data and one hot encoding

The Theme and Subtheme fields were counted, grouped and analysed. It was decided to group any themes and subthemes in to ‘Other’ values if the counts were 5 or less. This reduced the number of themes from 81 to 64 and the subthemes from 330 to 191.

One hot encoding was then applied to these two categorical fields to instead make them numerical binary fields for each individual theme and subtheme.

### 3.6 Log transformation

Histogram plots showed significant skew in the data for Pieces, Minifigures, OwnedBy and WantedBy. Log transformations were applied to each in an attempt to remove the skew and make the distribution of values more similar to a normal distribution.

### 3.7 Pairplots

Pairplots were generated for a subset of our features consisting of our remaining original float and integer fields: 'Year', 'Minifigs', 'Pieces', 'UKPrice', 'USPrice', 'CAPrice', 'EUPrice', 'OwnedBy','WantedBy'.

It was discovered that, as expected, the four price fields were all highly correlated with each other.

### Descriptive statistics

The final task was then to produce some descriptive statistics on our resultant transformed data set, including means, modes and ranges.

## Key-data exploration findings

* Key Findings and Insights, which synthesizes the results of Exploratory Data Analysis in an insightful and actionable manner

The original dataset had 4722 rows of data across 22 columns. The data exploration has taken this data set with numerous null values and incorrect values and unique identifier fields and converted it into a usable set with 3982 rows and 268 columns.

We have ensured the dataset contains primarily only standard released Lego sets. We have ensured null values are correctly deal with, either by replacing with the correct value, the best estimated value, or removing the whole row from our data set.

All columns are now numerical values with categorically data removed or transformed. Where necessary, we have also transformed our data, for example, with the log transformations of several numerical data points to remove skews in the data.

We have discovered the four pricing fields are highly correlated, as expected and so could either choose to use just one value in future models, or explore further if the relationship holds true for all types of Lego set. We should remain aware of the estimation missing prices based on the prices from other regions, should we wish to use this data. We may want to consider using a dataset without the estimated values if it makes sense to do so.

We can see from the pairplot that our data is well distributed across the different years between 2011 and 2020, setting us up well to analyse trends across the last decade.

We can see the data set has a large number of different themes and subthemes, yet we have also significantly reduced this number from 81 to 64 for themes and from 330 to 191 for subthemes by grouping the categories with the least data. This will allow us to keep the vast majority of the information held in these fields, while reducing the number of columns and so speeding up any future processing.

From looking at our descriptive stats and distributions of Minifigs and Pieces, we can see a large ranges between the highest and smallest values, with more skew towards the smallest. However, the outliers at the top of the ranges are still valid entries in our data set and contain valuable information and should be left in.

## Hypotheses

**Hypothesis 1**

Null: Lego set prices are cost the same in the UK and the US

Alternative: Lego set prices are less expensive in the UK than the US

**Hypothesis 2**

Null: Lego sets are the same size, on average, in 2020 than they were in 2011

Alternative: Lego sets are, on average, bigger in 2020 than they were in 2011

**Hypothesis 3**

Null: Licensed set contain as many minifigures as non-licensed sets

Alternative: Licensed set contain more minifigures than non-licensed sets

## Significance test

We will conduct a significance test on hypothesis 2:

**Hypothesis 2**

Null: Lego sets are the same size, on average, in 2020 than they were in 2011

Alternative: Lego sets are, on average, bigger in 2020 than they were in 2011

For this test, we will compare the mean number of set pieces in both years using a 2-sample t-test in our Python notebook.

First, we calculated the mean and standard deviation for log-transformed values for the number of pieces in a set in either year.

2011 mean of pieces: 4.342928399832589

2020 mean of pieces: 5.063402950409442

2011 std of pieces: 1.6061282906456205

2020 std of pieces: 1.6000627928740678

The, we applied the t-test on the log-transformed piece data for each year, with a 5% confidence value.

The test gave: p-value = 3.61903535e-09

We can therefore reject the null hypothesis and say that our alternative hypothesis is true, that is that Lego sets are, on average, bigger in 2020 than they were in 2011.

## Next steps

As we only compared the data on set pieces from the years 2011 and 2020 an obvious next step would be to investigate the time series of average set pieces to see if the trends were consistent or if either year already considered was an outlier or went against the general trends

We could also investigate if there were significant changes in other descrivptive sets between the two years – did the range of set sizes change between the two years, has there been any change in to median set size (which could be important in determining if either year’s mean was impacted by more smaller or more larger sets than the other).

Minifigures are more expensive pieces due to the pre-assembly and extra printing required, therefore it will be interesting to find out if the average number in sets changed between 2011 and 2020, like it appears to have done for the number of pieces. If there was a change, was it bigger to smaller than for pieces?

We might also want to consider the impact of themes on this number. Could the change be down to more sets in specific themes in one year compared to the other. Were there bigger or more significant changes in certain themes than others. Can any of the change in set pieces be explained by the retirement of certain themes, or the introduction of specific new themes? Or is the change seen equally across all themes?

Another interesting next step might be to try and build a predictive model using the data to estimate what we might expect in future years. Can we predict the average set size expected in 2021 and beyond?

In addition to this, extensions in to other areas unrelated to just set size could be to consider price predictive machine learning models to estimate the prices of future sets. Usually knowledge about minifigures and set piece counts is leaked well before prices of sets are known. Can we use this early leak data to create accurate estimates of what the final price of a set may actually be before the official release?

## Summary of findings

We have obtained a set of data relating to official Lego set releases between the years 2011 and 2020. We have been able to clean this data set, accounting for missing values, and transform it into a good quality dataset that can be used for testing hypothesis about the data.

We were able to identify a number of hypothesis and carry out a significance test comparing the mean number of pieces in sets in 2011 to 2020 to find, using a 5% confidence interval, that we could reject our null hypothesis and therefore accept our alternative that Lego sets are, on average, bigger in 2020 than they were in 2011.

We then identified a number of opportunities for further research with the data set. Finally, it would be interesting to obtain additional data on the Lego sets for further analysis. Additions for future sets would be welcome as the data becomes available, it would be interesting to seek additional data on sizes of boxes, weights of sets and quantities of each piece to see how these related to the other attributes. Finally, if we could obtain re-sale prices (obtained from websites like Bricklink, eBay and BrickOwl) we could start to investigate models for predicting the future re-sale value of specific Lego sets based on all the other data we have.