# Data Cleaning and Transformation

## Introduction

Data Cleaning is defined as correcting or removing inaccurate values from the dataset. It is one of the crucial steps while performing Data Mining. A Data Scientist spends roughly about 60% of the time cleaning the dataset. Still, the process of Data Cleaning is not perfect. Even after cleaning the data, a dataset can never be assumed to be completely free of inaccurate values. The main aim of Data Cleaning is to normalize all the attributes so that a better and accurate model can be created. Some of the most used Data Cleaning methods are:

* You can completely ignore the tuple if it is missing some values. This method is not effective at all.
* You can input the missing data manually. This method only works if the dataset is small.
* You can calculate and use attribute mean to fill in the missing values in the dataset.
* You can delete the tuple from the dataset.

The best practice is to fill in the missing attribute values either manually or by taking the mean of the entire attribute column. Hence, we have used this data cleaning method in our assignment.

We have two datasets. The game\_detailed\_info.csv and the bgg-15m-reviews.csv. Hence, we will perform all the methods on both of them one by one.

## Reviews Dataset

The reviews dataset contains attributes like user, game ID, rating, comment, game name. Most of the attributes in this dataset contains text data. As Natural language Processing is out of scope for this project, we will be focusing on only the rating attribute from the first dataset. Here is a snapshot of the dataset for reference.

Table

Description automatically generated

As we are focused on the ratings only, we tried finding the null values present in ratings column.

Graphical user interface, text, application

Description automatically generated

We used isnull() to find the null values and sum() to add up all the null values. The output was that there were no null values in the rating attribute. Then, we tried to converted the 0 values into null values but then also we did not find any null values for the rating attribute.

Graphical user interface, text, application, email

Description automatically generated

Next, to calculate the outliers in the dataset, we used IQR method. The IQR came out to be 2. So using the formula, we tried to find the max and min values. All values beyond these are considered as outliers. We got the min as 3 and max was beyond scope of the rating dataset hence it got discarded.

Graphical user interface, text, application

Description automatically generated

We can remove the ratings below 3 as per the guidelines. But as we are trying to create a recommendation system, we will need ratings of 3 and below as well so that we can classify a bad game successfully. Hence, we have decided to keep all the values even after finding the outliers.

For doing the normalization of the rating attribute, we used the z-score. To simply define, a z-score tells us the distance between mean and the datapoint value. But more technically it’s a measure of how many standard deviations below or above the population mean a raw score is.

We normalized the dataset using the pandas inbuilt function scaler.fit\_transform. The rating attribute column was highly unnormalized. We compared the describe functions and histogram plots before and after we did normalization. The results proved that the dataset was not normalized. Please find the screenshots of the histograms and the describe function below. This concludes the data cleaning and data transformation part for the first dataset. Next, we will move to the game\_detailed\_info dataset and explore it further.

A picture containing text

Description automatically generated Table

Description automatically generated

Snapshot of the describe function before and after we applied Data Normalization.

Chart, histogram

Description automatically generatedChart, histogram

Description automatically generated

Snapshot of the histogram plot showing rating values before and after we applied Data Normalization.

## Game Info Dataset

Game info dataset contains a lot of numeric attributes such as yearpublished, minplayers, maxplayers, playingtime, minplaytime, maxplaytime, etc. It also contains categorial data attributes such as descriptions, publisher, game genre, etc. For filling in the null values and for the scope of this particular assignment, we have focused more on the numeric attributes. The text data can be processed using NLP but that is out of scope right now. To find the null values in the dataset, we again used the isnull() and sum() built-in functions for finding the count of the null values in each attribute. The screenshot can be seen below with the initial null values:

Table

Description automatically generated

There can be cases where the value can be inputted 0 instead of null in the dataset. To verify, we used the lambda function to map all the numeric attributes and if it finds a 0 value then replace it by null value. We used the mapping and updated all the numeric data attributes to display null values when we use isnull() and sum() function again.

Text

Description automatically generated

After using the lambda function, we again counted the null values. We can see now a lot of null values as compared to the previous case. Now, we will work on calculating and inputting values in all these null places.

Table

Description automatically generated

In order to input some meaningful data in place of the null values, we decided to use the mean. So, we calculated the individual mean values for all the attributes and inputted them wherever there was a null value. We used fillna() to automatically fill the null values with mean values from the attributes and saved the dataset.

Text

Description automatically generated

Now, if we count for null values in the dataset using isnull() and sum() function, the number of null values should be 0 again.

Table

Description automatically generated

Next, for calculating the outliers, we used IQR again.

Table

Description automatically generated

Table

Description automatically generated

For the binary values created above, we get the values whether they are outliers or not. We can remove all the outlier values, but we have decided to keep them for training the model. We don’t want to lose any data because when we are getting recommendations, we will need good as well as bad game information.

For normalizing the dataset values, we used the z-score. To simply define, a z-score tells us the distance between mean and the datapoint value. But more technically it’s a measure of how many standard deviations below or above the population mean a raw score is. We normalized the dataset using the pandas inbuilt function scaler.fit\_transform. The rating attribute column was highly unnormalized. We compared the describe functions and histogram plots before and after we did normalization. The results proved that the dataset was not normalized. Please find the screenshots of the histograms function below.

After both the datasets where processed with the data cleaning techniques, we went ahead and saved the datasets into new csv files. We will be using these csv files as inputs for the next assignments and ultimately working on the model based on these csv files.

Chart, box and whisker chart

Description automatically generated

Chart, box and whisker chart

Description automatically generated

Snapshot of the histogram before and after normalizing the dataset.