

For office use only

Team Control Number

For office use only

T1 _____

8784

F1 _____

T2 _____

Problem Chosen

F2 _____

T3 _____

A

F3 _____

T4 _____

F4 _____

**2018
HiMCM
Summary Sheet**

Roller coaster riding is top-rated entertainment among the youngsters. However, the ranking systems of roller coasters are largely based on the input of subjective experience and rating instead of quantitative analysis. Therefore, the model we construct aims to provide a reliable method used for the roller coaster-rating based on their properties and objective analysis.

To begin with, we do the data cleaning and interpolation to extract the relevant data. Noisy data are being rectified, and some missing data are interpolated. The process simplifies our model by reducing the independent variables and raising accuracy. We obtained nine properties of the roller coasters for further analysis. Next, we analyze the data mainly by applying Principal Component Analysis to determine an initial ranking of roller coasters and compare it with the ranking online to see if the result can be taken as the training set in the methods that follow.

The model is first constructed with the help of Linear Regression and the KNN algorithm. It can be seen that the two models manifest quantitative analysis towards the issue, while the result is not accurate enough, which then brings about the optimization model. The Principal Component Analysis focuses on the most relevant independent variable with continuous data, while the Bayes Distinction has a strong ability to manipulate the discrete data. We also utilize BP Neural Network to solve the issue, which has the ability to construct an optimized model with proper training sets and possess high accuracy. Then the XG Boosting algorithm is employed to synthesize the three optimized models and produce a more reliable and stable rating. Ultimately, the sensitivity analysis validates the model's stability and precision, thus making its use in real life viable and reliable.

Advantages of the model we construct are shown in various aspects. The optimized model not only provides the quantitative results based on each independent variable but also successfully provides us with an objective rating of every roller coaster, which is far more persuasive than the solely subjective inputs. Furthermore, our model synthesizes the results from various advanced methods with clear logic chain, which guarantees our model's accuracy as well as stability. It is also flexible with the change of data input and enables the self-studying and self-improving through proper additional training sets, so the model is suitable to be applied under various circumstances. Moreover, we demonstrate our concept of the application with the algorithm applied. The application mainly aims to recommend the roller coasters based on the global ranking and individual's preference, as well as constructing a search engine to save the users' time on roller coaster selecting. All of the functions are supported with concrete programming frames, the methods of which include correlation coefficients, Mahalanobis, and BP neuron network. Thus, it can successfully achieve the goal of fulfilling the potential riders' demands.

To conclude, our model proposes a reliable and precise method for the rating of roller coasters based on objective algorithms. It presents high accuracy, reliability, and stability, the features of which make it stands out from other analysis based solely on subjective inputs. The application we conceive can also fulfill the users' various needs and thus possesses high pragmatic value.

Key Words: Roller Coasters, Principle Component Regression, Bayes Distinction, BP Neural Network Fitting, XG Boosting algorithm

News Release: Hop till you Drop !

--- New Techniques of a Modeling Team Shed Light on Unique Roller Coaster Experience

Have you ever been in want of going to an amusement park and ride the roller coasters? Have you ever been troubled by the issue that you do not know which type of roller coaster is the most suitable for you? When you are planning to ride the roller coasters, have you once wondered a scoring system that can have an objective rating system towards all the roller coasters around the globe which is not affected by personal opinions or perceptions? If this is the case, you do not need to be desperate anymore. Fortunately for you, an excellent modeling team has skillfully addressed the concern, who invented a set of algorithms which specially deal with the rating of roller coasters. Don't forget that taking the roller coasters is the obsolescence for almost everyone. With the rating of roller coasters, you will be able to reap the most overwhelming sense when riding them!

Based on 300 roller coasters all over the world, the modeling team utilized several techniques to evaluate each one of them skillfully. They take diverse accounts into consideration, not only the standards that can flash into your mind, such as the speed, height, or the number of inversions, but also some factors that are less concerned but play a significant role in the consideration of people, including the type of coasters, even the material that the coasters are made from. Their algorithms are also well-considered, having a notable performance in different kinds of variables. They employed the cutting-edge BP Neural Network, simulating the principle of human brains and achieve an accuracy of over 99%. They applied XG Boosting algorithm to synthesize the various methods employed for solving the problem, of which the fundamental theory is that a complicated issue can be better estimated when synthesizing the judgment of each expert than that of a sole expert. Hence, their solution is credible and reliable.

We believe that you are eager to know which roller coaster leads the rank, and here it comes. The one which is on the top of the list names T Express locates in Everland Park, Yongin-si, Gyeonggi-do, South Korea, followed by roller coasters in France, the US, China, and Japan. The best roller coaster in the US is Apocalypse Six Flags America opened in 2012, lying in Upper Marlboro Maryland with a 90-foot drop, 100.0 feet high, 55.0 mph speed, 2900.0 feet long. A single loop costs approximately 2 minutes. No matter you want to run after the best roller coasters or have a suburban trip, the new ranking will never let you down.

Want to personalize your recommendation? Don't hesitate to download the newly-designed app given by the modeling team. It features several functions. From the best roller coasters in the world to the best coasters only for your interest, from the enumeration of the entire roller coasters to the search engine of your roller coaster preference, you can find whatever you want in the application. Based on big data of all the users and your personal as well as regional information, it can personalize the range of your choices and dynamically adjust the recommendation just for you. You can even search the parameters of the roller coasters if you want.

No matter whether you are a Spartan or a spare time traveler, no matter whether you are a crazy fancier or a casual visitor, as long as you come and visit, a sea of roller coasters will be unveiled in front you. With the guide from the modeling team, you will definitely be overwhelmed with the bliss on the top, abuzz with the loop, and hop till you drop.

Contents

1. Background

- 1.1 Research Background**
- 1.2 Restatement of the Problem**
- 1.3 Research Method and Train of Thinking**

2. Assumptions, Justifications, and Definitions

- 2.1 Assumptions and Justifications**
- 2.2 Definitions**

3. Data Procurement and Process

- 3.1 Data Cleaning and Interpolation**
- 3.2 Cluster**
- 3.3 Ideal Solution**
- 3.4 Principal Component Analysis**

4. Modeling

- 4.1 Basic Statistics**
- 4.2 AHP Method**
- 4.3 Linear Regression**
- 4.4 KNN Algorithm**

5. Optimization

- 5.1 Principal Component Regression**
- 5.2 Bayes Distinction**
- 5.3 BP Neural Network Fitting**
- 5.4 XG Boosting Algorithm**

6. Analysis of the Model

- 6.1 Sensitivity Analysis**
- 6.2 Strength and Weakness**

7. Comparison of the Top 10 Roller Coasters

8. Concept and Design for a User-friendly App

- 8.1 Initial Recommendation**
- 8.2 Recommendation Base on Preference**
- 8.3 Search Engine for Roller Coasters**
- 8.4 Auxiliary Functions**

9. Conclusion

10. Reference

11. Appendix

- 11.1 MATLAB Code**
- 11.2 PYTHON Code**
- 11.3 Final Result**
- 11.4 Original Data of Rating and Ranking from Websites**

1. Background

1.1 Research Background

Roller coaster is an exciting entertainment fascinated by many of today's youngsters because of its stimulation and pleasure. However, the rating system of roller coasters is relatively scarce and mainly based on people's own experience, lacking quantitative analysis of roller coasters' different traits. For instance, Coaster Buzz posted the current top 100 roller coasters on its website, but the rating process is largely based on its members' track records and subjective experience inputs. Admittedly, one of the method's main advantages proves to be the vast sample size and the ratings' rigorous selection in order to exclude the anomalies. But even with the rating results refreshed weekly, the poll on the internet only reflects the opinion of any one person and the riders who provide their experience and scores may mostly come from the same region, so the ranking is not based on the world's scale and will certainly still lose some great rides. Therefore, most rating methods are highly unstable and unconvincing, making the roller coaster-choosing process inaccurate and the riders being dissatisfied. According to the current needs and lack of quantitative methods of rating, a proper method for ranking the roller-coasters is in dire need.

1.2 Restatement of the Problem

The question is based on the fact that nowadays roller coaster ranking systems are largely dependent on riders' own subjective inputs, with few considering the roller coasters' own properties. Providing us with the basic information of 300 roller coasters around the world, the question asked us to decide the top ten roller coasters using quantitative assessing methods, compare them with other methods currently being used and analysis the strength and weakness. Besides, we are required to develop the concept of a user-friendly APP which aims to help the potential riders finding the proper roller coasters that will satisfy their needs. Finally, we write a News Release to publicize our quantitative methods, the result of top-10 roller coasters based on the data given and the concept of our newly-designed APP.

1.3 Research Method and Train of Thinking

We do the data cleaning first and interpolate the missing data, while extracting useful and relevant data and conducting basic statistics for further research. Next, we come to the data procurement part to examine whether the score of the roller coasters online can be a training set of our model with the help of Principal Component Analysis. Then we apply the results of data procurement for modeling. In the modeling process, we apply results from Principal Component Analysis to the Analytical Hierarchy Process, KNN, and Linear Regression. At this point, we have reached the conclusion of the rank of different independent factors. Furthermore, we conduct optimization for each model. We optimize KNN by Bayes Distinction, optimize Linear Regression by Principal Component Regression, and utilize BP Neural Network Fitting to achieve higher accuracy. Afterward, we employ the XG Boosting algorithm to synthesize the three methods and reach a conclusion over the ranking. Finally, we compare our rating and raking with those online results and design the notion of our desired application. Figure 1 below presents the whole modeling process, and if the method is marked red, it indicates the result of this analysis is not applied to further modeling and optimization. It does not mean that the models in red do not work well for the problems; it merely means that the models in red are technically correct and work if the database is expanded and more data of the roller coasters are given, while they are not perfectly suitable for the current data given.

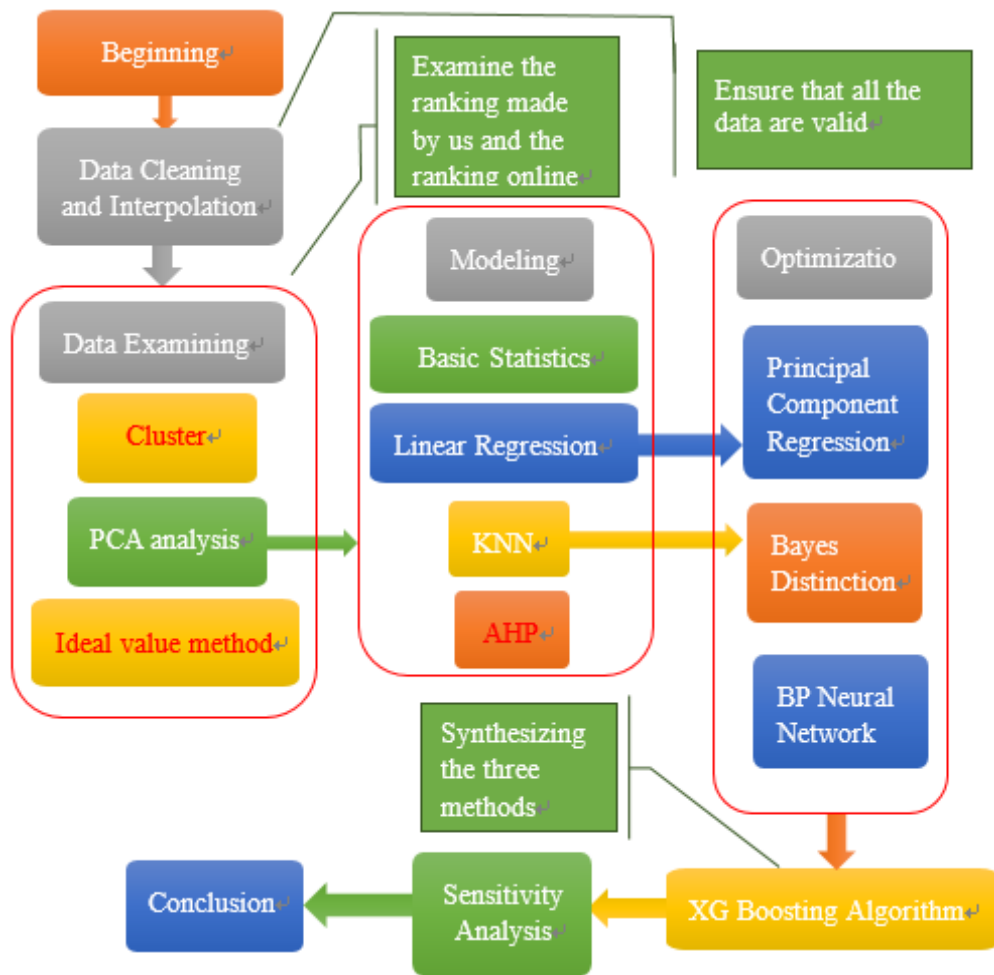


Figure 1: The flow chart of the whole modeling process

2. Assumptions, Justifications, and Definitions

2.1 Assumptions and Justifications

- Assumption 1: All the data which pass the data cleaning part are credible and reliable, meaning that they have no error.

Justification: In light of the fact that the data are given by COMAP, who have provided us the data from reliable sources, the data ought to be without fabrication.

- Assumption 2: The parameters of the roller coasters are constant and do not change at a different time in a day. For instance, the speeds of the roller coasters are always the value given in the data. No matter it is in the morning or evening, and no matter how many people there are on the roller coasters, they will always travel at the speed given.

Justification: It counters our intuitive to run a single roller coaster under different parameters because the operators have no incentives to do so.

- Assumption 3: All the roller coasters are operated under the parameters given in the data and function normally. For instance, the roller coasters will not run at speed lower than designed speed.

Justification: It is reasonable to assume that the amusement parks go their great length to allow the riders to have a thrilling experience, which can be mostly obtained through

operated under the given parameters. It is also rare to see that a roller coaster may break down due to technical issues.

- Assumption 4: Except the parameters given in the data, all other properties of the roller coasters, such as the riding comforts of the seats, are exactly the same, which indicates it has no impact on the final score to the riders.

Justification: We make the assumption so as to simplify the problem, while we are unable to find and evaluate the data of the roller coasters other than the given ones.

- Assumption 5: All the roller coasters are suitable for any riders to hop on, which means all the coasters have no potential safety hazards, and the coasters will not mentally harm the riders.

Justification: In accordance with the policies and the regulations, all the operating roller coasters ought to have passed the mandatory security test given by local authorities, which institute the rules to eradicate safety concerns.

- Assumption 6: The scores online can be used as a set of dependent variables while regarding the properties of the roller coasters as independent variables.

Justification: We will test the assumption in detail in part 3 and further illustrate the viability of applying certain online results for our analysis.

2.2 Definitions

Table 1: the definition of notations

Notation	Definition
A_{ij}	The element in i^{th} Row and j^{th} Column in matrix A
X	The independent variables matrix
x	Row vector of independent variables
y	Row vector of dependent variables
\bar{x}	The algebra average of several data
$d(X, Y)$	The Mahalanobis distance of the data
Σ	The covariance matrix
x_p	The p^{th} original variable
z_q	The q^{th} New variable
m	The number of samples
l	The number of variables in each sample
x_{ij}^*	The standardized data at row i and column j
x_{ij}	The data at row i and column j before standardization
R	The correlation coefficient matrix in principal component analysis
λ_q	The q^{th} characteristic roots or eigenvalues in Weight determination Technique
$a_q(A_q)$	The q^{th} characteristic vectors
a_{pq}	The p^{th} value of the q^{th} characteristic vectors
$(w_1 \quad \dots \quad w_n)$	Weight vector in AHP

n	The number of choices of target layer in AHP
w	The eigenvector in AHP
β	Coefficient matrixes of the original data
β'	Coefficient matrixes of Principal Component Regression
$P\{X\}$	The probability that satisfies condition X
α	Reliability in Regression
θ	Parameters to be estimated of the ensemble in Regression
$\hat{\theta}_1$	The confidence upper limit in Regression
$\hat{\theta}_2$	The confidence lower limit in Regression
$P(B_i A)$	Posteriori probability in Bayes Distinction
$P(A B_i)$	Priori probability in Bayes Distinction
$P(B_i)$	The frequency at which the sample appears in Bayes Distinction
G_i	The ensemble in Bayes Distinction
$f(x)$	Probability density function of G_i in Bayes Distinction
p_i	The priori probability of G_i In Bayes Distinction
k	The number of G_i in Bayes Distinction
$P(j/i)$	The conditional probability of wrongly categorizing the sample of G_i to the ensemble G_j
$c(j/i)$	The loss caused by the wrong categorization
D_k	A division of a set of distinction samples
ECM	The average wrong distinction loss
$L(\theta)$	The overall loss of each classifier
y_i	Classification function
\hat{y}_i	function of each classifier to reduce the loss
S_k	The score of the data to show the accuracy of the prediction

3. Data Procurement and Process

3.1 Data Cleaning and Interpolation

Finally, we obtain 9 variables that we mainly use, which are Geographic Region, Construction, Type, Year Opened, Height, Speed, Length, Duration, and Number of Inversions. The columns that are not mentioned above are regarded as the identification of each roller coaster, which will not be used for modeling. The 293 data after cleaning can be seen in the appendix. The following figure 2 illustrates the process mentioned above.

As we downloaded the data, we first number the 300 roller coasters from 1 to 300 and do the data cleaning as the foundation of the entire model. We remove the drop column, the G Force column and the Vertical Angle column since there are more than a half of the data missing, which renders it void for us to interpolate the missing value. We also remove the status column, since all the roller coasters are operating. Then with the help of XLRD and XLWR module in PYTHON, we convert the expression of the duration cells from both the minutes and seconds into seconds only. We also enumerate the Geographic Region column, the Construction column, and the Type column. For the Geographic Region column, we employ 1 to 8 represent Asia, Europe, North America, Central America, South America, Middle East, Oceania, and Russia respectively. For the Construction column, we use 1 to 2 represent steel and wood respectively. For the Type column, we use 1 to 6 represent sit down, inverted, stand up, suspended, flying, and wing respectively. We also notice that some of the Type cells are filled in steel or wood, which is not a possible choice of Type, which we use 0 to

represent the two choices. We removed the unit in the cells of Height in order that it is able to be dealt with further.

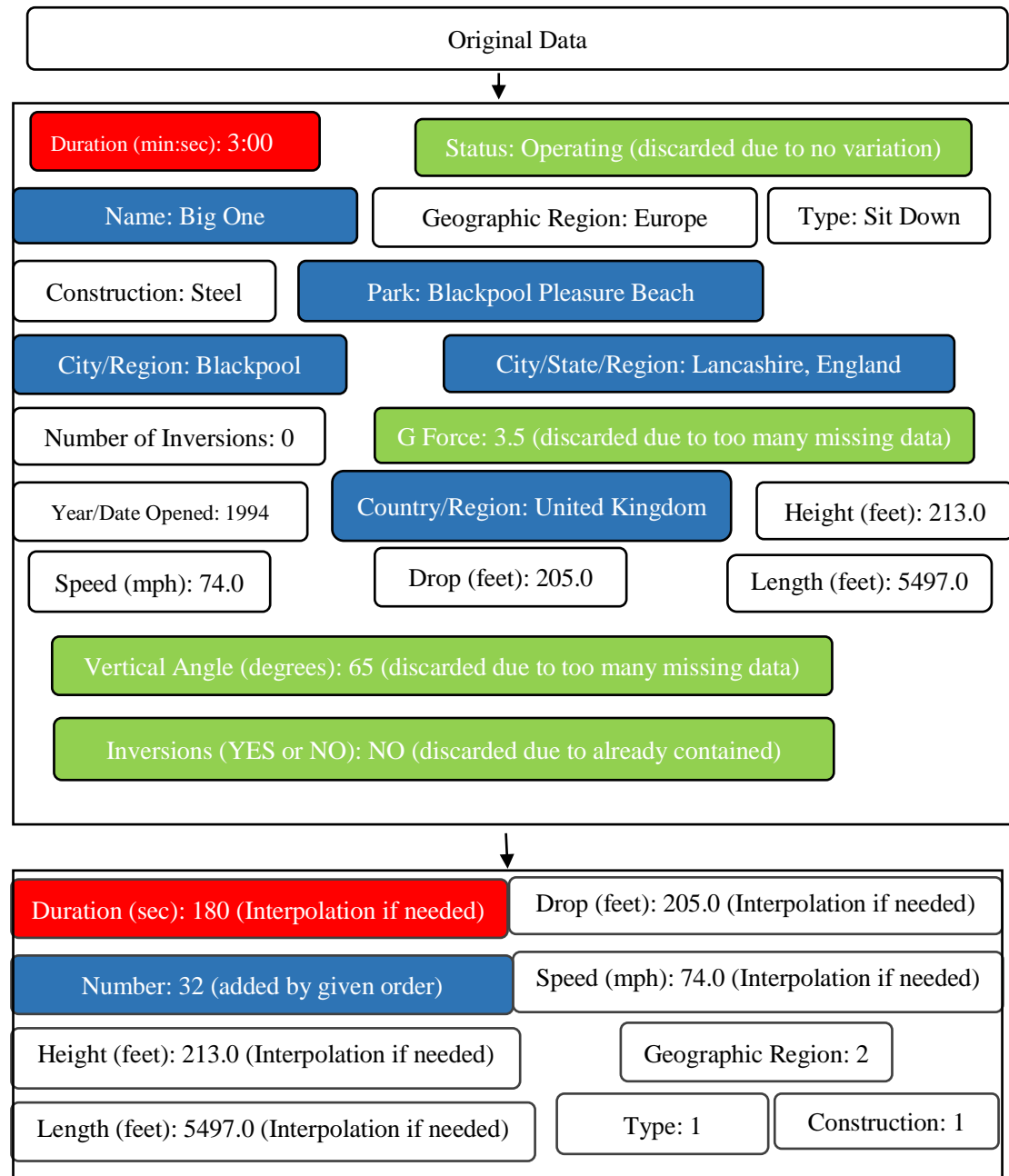


Figure 2: Data Processing diagram

We discover that some of the data in the Height, Speed, Length, and Duration column are missing; thus we consider that we use the interpolation method to fill in the missing number. We examine the correlation coefficients between the columns, and find that the correlation coefficient between Height and Speed is 0.836280084187907, and the one between Length and Duration column is 0.619704366781674, indicating that the two groups of column reveal a strong tendency of correlating, which means we can use the two columns in each group to interpolating the missing data of each other. We sort the interpolating variable and calculate the arithmetic means of the interpolated variable if an interpolating variable refers to more than one interpolated variable in the given data set before we utilize Piecewise Cubic Hermite Interpolation to interpolate our variable. We do the same process for the rest three columns and fill in all the data.

The reason why Piecewise Cubic Hermite Interpolation is suitable for our problem is that it avoids the oscillation between the point series, while we do not pay much attention to the smoothness of the interpolation function. We eliminated some data that miss both interpolating variables and interpolated variable.

Finally, we obtain nine variables that we mainly use, which are Geographic Region, Construction, Type, Year Opened, Height, Speed, Length, Duration, and Number of Inversions. The columns that are not mentioned above are regarded as the identification of each roller coaster, which will not be used for modeling. The 293 data after cleaning can be seen in the appendix. The previous figure 2 illustrates the process mentioned above.

3.2 Cluster

We want to rank the roller coasters at the beginning and compare the ranking produced by our method with the ranking of the scoring system online. If these two are similar, we can regard the online scoring system as a learning set and establish a model to rate all the roller coasters. ^[1]

We utilize cluster to decide which of the roller coasters are similar. It can be predicted that similar roller coasters are more likely to have a similar rating and ranking; therefore we can divide all the roller coasters into several groups. If the roller coasters that are in the same group are more likely to lie in the same online score interval, such as the high score or the low score, we can determine that our ranking system is consistent with the online scoring system, which makes it viable for us to establish a model with the online system.

We use Mahalanobis distance for clustering and draw the dendrogram. The formula is as the following formula 1.

$$d(x, y) = \sqrt{(x - y)\Sigma^{-1}(x - y)^T} \quad (1)$$

Among the formula, x and y denote two row vectors; Σ denotes the covariance matrix; $d(x, y)$ denotes the obtained Mahalanobis distance of the data. The result is shown in figure.

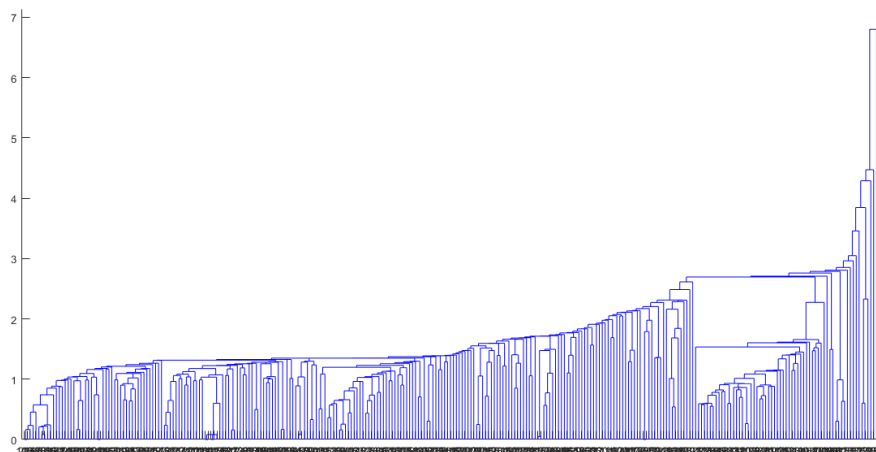


Figure 3: Mahalanobis Clustering Dendrogram, some of the categories contain too less roller coasters

From figure 3 above, we can see that some of the categories contain too fewer roller coasters, which shows that this method is complicated to set the roller coasters apart under the currently given data. Hence, we consider using other methods.

3.3 Ideal Solution

We also come up with a way that enables us to get the maximum value of year opened, height, speed, duration, length, and number of inversions in the data, setting them as an ideal solution. We then calculate the Mahalanobis distance between each data and the ideal solution, taking advantage of the avoidance of the effect of the dimension. The formula 2 is shown following:

$$d(x, y) = \sqrt{(x - y)\Sigma^{-1}(x - y)^T} \quad (2)$$

Among the formula, x and y denote two row vectors; Σ denotes the covariance matrix; $d(x, y)$ denotes the obtained Mahalanobis distance of the data. Part of the results is shown in table 2, the rest of which are in the appendix.

Table 2: Ideal Solution result

Number	Geographic Region	Construction	Type	Year/Date Opened	Height (feet)	Speed (mph)	Length (feet)	Duration (sec)	Number of Inversions	PCA value	Mahalanobis Ideal Solution Distance
240	1	1	1	2000	318.3	95.0	8133.2	240.0	0	1.90100669	4.450616635
28	3	2	1	1979	110.0	64.8	7359.0	250.0	0	1.70406025	2.536349531
101	1	1	1	1996	259.2	80.8	6708.7	216.0	0	1.41536641	4.632855773
279	2	1	1	1991	107.0	50.0	7442.0	250.0	0	1.25479944	7.024809364
59	1	1	1	2016	242.8	84.5	5105.0	252.0	0	1.17235105	3.287317761
226	2	1	1	2002	239.5	78.9	5315.0	240.0	0	1.16312841	1.700167085
152	3	1	1	2012	306.0	92.0	5486.0	208.0	0	1.10930926	4.535407498
164	3	1	1	2000	310.0	93.0	6595.0	140.0	0	1.08838272	2.439859717
100	6	1	1	2010	170.6	149.1	6561.7	143.0	0	1.06852452	4.311175525
288	3	2	1	2006	159.0	67.0	6442.0	165.0	0	1.06850007	1.6937325
271	3	1	1	2001	245.0	85.0	5312.0	210.0	0	1.03797533	1.185300931
104	3	1	1	2015	325.0	95.0	6602.0	140.8	0	1.01938862	6.091589304
105	1	1	1	1998	131.3	60.9	5457.7	236.0	0	0.97599041	1.360278997
143	1	2	1	1992	138.0	57.0	5249.3	154.0	0	0.92729164	3.536106251
132	3	1	1	2010	305.0	90.0	5100.0	180.0	0	0.92263462	2.940183392
32	2	1	1	1994	213.0	74.0	5497.0	180.0	0	0.91634504	3.207843951
131	3	1	1	2010	232.0	75.0	5316.0	213.0	0	0.9016305	1.975295287
257	1	2	1	2008	183.8	64.6	5383.8	138.6	0	0.88054183	4.979570137
29	3	1	1	2008	230.0	77.0	5318.0	201.4	0	0.87603496	3.191376901
242	3	1	1	1997	200.0	75.0	5600.0	180.0	0	0.85200275	1.884368267
167	2	1	1	1998	205.0	75.0	5600.0	180.0	0	0.85165712	2.429884765

However, we find that it is flawed for us to set the highest score as the ideal one with no direct evidence supporting. Therefore, we still need to consider other methods.

3.4 Principal Component Analysis

With the help of PCA, we are able to rank the roller coasters.

We utilize the 9 original variables mentioned in 3.3 as the original data. We still use X to denote independent variables matrixes and Y the dependent variables. The original variables are $x_p (p \in \{p \in N^* | p \leq l\})$; the new variables are $z_q (q \in \{q \in N^* | q \leq p\})$. We use m to denote the number of samples and use l to denote the number of variables in each sample. Thus, the data matrix is as matrix 3^[3]

$$X = \begin{bmatrix} x_{11} & \cdots & x_{1l} \\ \vdots & \ddots & \vdots \\ x_{m1} & \cdots & x_{ml} \end{bmatrix} \quad (3)$$

Since the data vary in dimensions and ranges, we need to standardize the data. We adopt the variance standardization technique to operate the data so that the variance of the standardized data is 1, while we conduct the central translation so that the mean of the data is 0. The formula is shown as formula 4-5

$$\bar{x}_j = \sum_{i=1}^n \frac{x_{ij}}{n}, \sigma_j = \sqrt{\sum_{i=1}^n \frac{(x_{ij} - \bar{x}_j)^2}{n-1}}, x_{ij}^* = \frac{x_{ij} - \bar{x}_j}{\sigma_j} \quad (4-5)$$

x_{ij}^* denotes the standardized data at row i and column j ; x_{ij} denotes the data at row i and column j before standardization. i denotes total column number and j denotes total row number.

Then we establish the correlation coefficient matrix R . The formulas are shown in formula 6-7.

$$r_{ij} = \frac{\sum_{k=1}^n (x_{ki} - \bar{x}_i)(x_{kj} - \bar{x}_j)}{\sqrt{\sum_{k=1}^n (x_{ki} - \bar{x}_i)^2 \sum_{k=1}^n (x_{kj} - \bar{x}_j)^2}} \quad (6)$$

$$R = (r_{ij})_{l \times l} \quad (7)$$

Then we obtain the characteristic vectors $\lambda_q (q \in \{q \in N^* | q \leq l\})$ which satisfy $\lambda_x > \lambda_y$ for $\forall 1 \leq x < y \leq q$ and characteristic vectors $a_q (q \in \{q \in N^* | q \leq l\})$ to determine the load a_{pq} on each new principal component variables z_q of the original variables x_p , which are equal to the q^{th} larger characteristic values of the correlation matrix corresponding to the eigenvectors. a_{pq} is the p^{th} value of the q^{th} characteristic vectors. The formula is as formula 8:

$$RA = \lambda A \quad (8)$$

In the formula, A denotes each characteristic vector, λ denotes each characteristic value. The characteristic roots are shown in table 3. Characteristic vector matrix is in the appendix.

Table 3: Principal Component Analysis Characteristic Value

0.045393	0.232633	0.438638	0.607466	0.853558
1.011198	1.468301	1.707678	2.635135	

The contribution rate formula and the total contribution rate formula is as formula 9-10.

$$\frac{\lambda_i}{\sum_{k=1}^q \lambda_k} \quad (i=1, 2, \dots, p) \quad \text{and} \quad \frac{\sum_{k=1}^i \lambda_k}{\sum_{k=1}^q \lambda_k} \quad (i=1, 2, \dots, p) \quad (9-10)$$

We obtain the total contribution rate until the fifth principal component is 85.29%, which is larger than 85%. Therefore, we take the first fifth eigenvalue as the principal component. Suppose the principal component is formula set 11

$$\begin{aligned} z_1 &= a_{11}x_1 + a_{21}x_2 + a_{31}x_3 + a_{41}x_4 + a_{51}x_5 + \dots + a_{91}x_9 \\ z_2 &= a_{12}x_1 + a_{22}x_2 + a_{32}x_3 + a_{42}x_4 + a_{52}x_5 + \dots + a_{92}x_9 \\ &\dots \\ z_5 &= a_{15}x_1 + a_{25}x_2 + a_{35}x_3 + a_{45}x_4 + a_{55}x_5 + \dots + a_{95}x_9 \end{aligned} \quad (11)$$

In accordance with the first 5 scores of the principal component, we use the contribution rate as the weight and obtained the total score of each of the 293 roller coasters. Ranking the roller coasters, we put the top 5 in table 4, and the rest of roller coasters can be seen in the appendix.

Table 4: First 5 roller coasters of PCA analysis

Number	Geographic Region	Construction	Type	Year/Date Opened	Height (feet)
240	1	1	1	2000	318.3
28	3	2	1	1979	110.0
101	1	1	1	1996	259.2
279	2	1	1	1991	107.0
59	1	1	1	2016	242.8
Number	Speed (mph)	Length (feet)	Duration (sec)	Number of Inversions	PCA value
240	95.0	8133.2	240.0	0	1.901007
28	64.8	7359.0	250.0	0	1.70406
101	80.8	6708.7	216.0	0	1.415366
279	50.0	7442.0	250.0	0	1.254799
59	84.5	5105.0	252.0	0	1.172351

Searching the top roller coasters online in our ranking ^[2], we find that all of the top 10 roller coasters online ranked the top one-third of our ranking. Several top 10 coasters online are in the top 20 coasters provided by us. Thus it shows that the result online can be used as the training set. We download the scores from Costerbuzz ^[2] and use them for further modeling.

4 Modeling

4.1 Basic Statistics

After obtaining the original data, we do the basic statistics process. We download the score from the website, Coaster buzz, and set it as the dependent variables, while the variables given in the chart as independent variables. On the one hand, we make pie charts, as well as line charts, reveal the proportions of the roller coasters with each characteristic over the ensemble, as shown in figure 4-5.

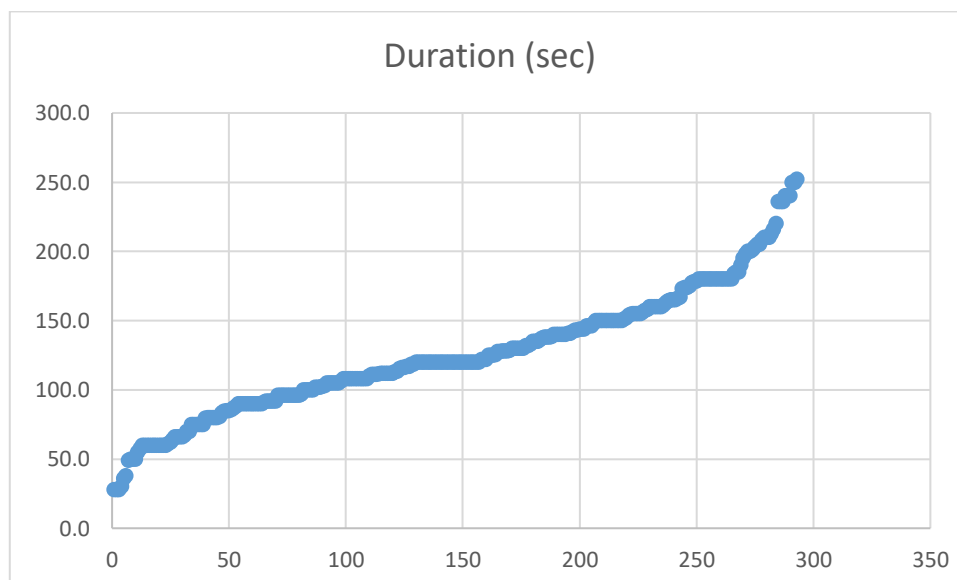


Figure 4: Line Chart of Duration. The Duration focus on 100-150 seconds interval.

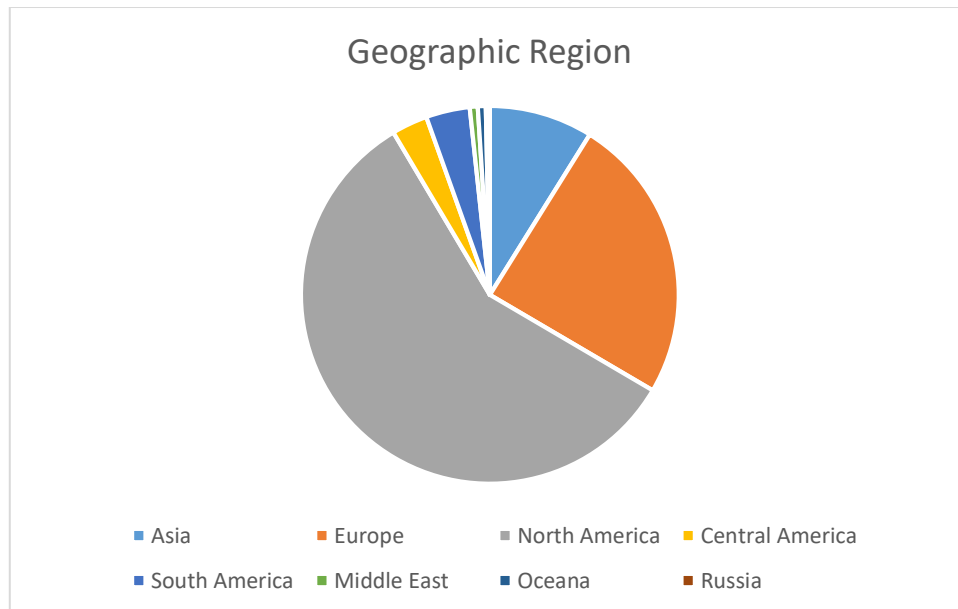


Figure 5: Pie Chart of Geographic Region. The roller coaster from North America takes a major proportion.

The previous charts demonstrate, for instance, that most of the given roller coasters locate in North America. The duration concentrates in 100-200 seconds interval.

4.2 Analytical Hierarchy Process

In order to determine the weight among diverse factors and judge the condition of roller coasters, we utilize the Analytic Hierarchy Process (AHP) to achieve the goal and determine the weight of each option in complicated and uncertain problems. We define each roller coasters as scheme layer, the 9 properties of the roller coasters as the standard layer, and the scores as the target layer to build up the 3-layer AHP model. The structure diagram is shown in the following figure 6. ^[4]

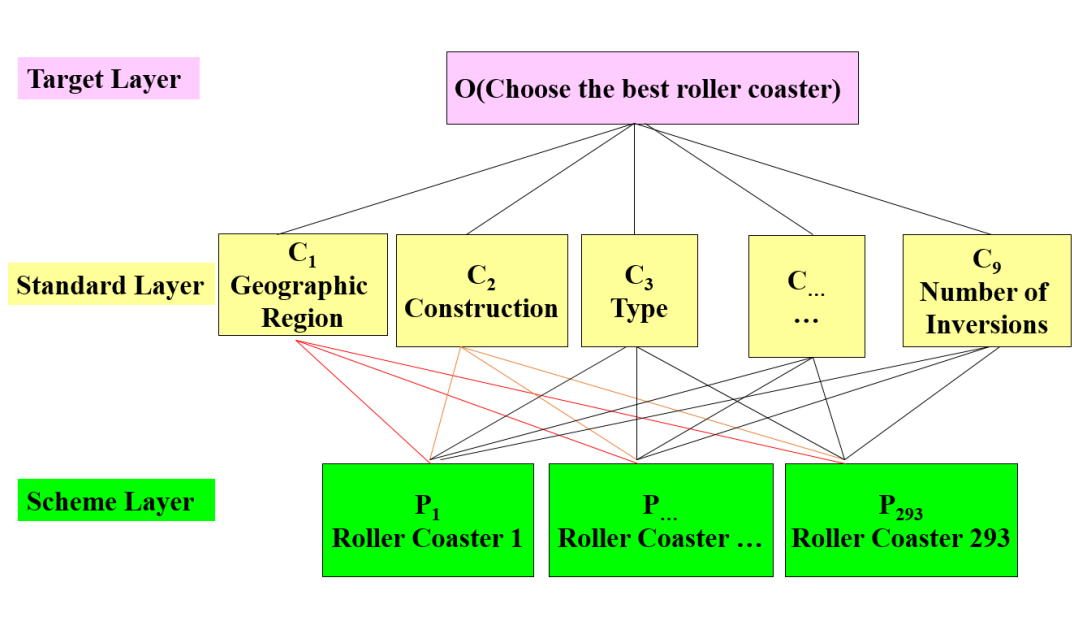


Figure 6: Structure diagram.

First, we define the number of roller coasters that possess certain properties under certain types of properties as w , which refers to the amount of a certain target choice under a certain

scheme layer condition. In accordance with the target choice, we obtain a weight vector $(w_1 \dots w_n)$ (n stands for the number of choices of target layer). We compute the ratio between the number, $w_i (1 \leq i \leq n)$, of each scheme layer choice under a common target layer choice and regard it as the weight of paired comparison matrix. As they are consistent matrixes, we do not need to apply consistency tests to the matrixes, for they are automatically consistent, which means that the eigenvalues are all identical. With the help of the formula of the eigenvalue and eigenvectors shown in formula 12,

$$Aw = \lambda w$$

we can obtain the eigenvectors, w . The following tables 5-6 respectively shows the paired comparing matrix and eigenvector.

Table 5: Paired Comparing matrix from standard layer to target layer

1	2.866024	2.306462	1.349377
0.348915	1	0.80476	0.470819
0.433565	1.242606	1	0.585042
0.741082	2.12396	1.709278	1

Table 6: Paired Comparing matrix from standard layer to object layer

0.396265
0.138263
0.171807
0.293665

Then we repeat the process from standard layer to scheme layer, compose the eigenvalues of each scheme, and obtain a matrix of weight vector from scheme layer to standard layer. Multiplying the two weight matrixes, we obtain the final weight matrix, which is the weight vector from scheme layer to target layer.

To define the paired comparison matrix from the standard layer to the object layer, we calculate the correlation coefficients between the online score and each given standard of data and the cross-ratio between the correlation coefficients. We discover that the possible value of Geographic Region varies too less, which means there are only two values that are different from the rest in the data with online score. The numbers of inversions exist too much zeros. The correlation coefficients of construction, type, and duration are too low for further analysis. Thus we merely take four standards to do further analysis, which are Year, Height, Speed, and Length, discarding the rest variables.

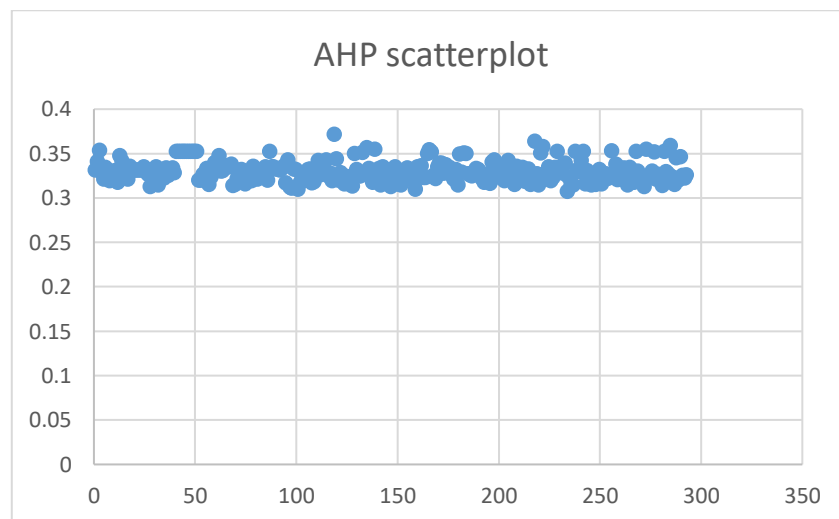


Figure 7: Result analysis. The weight of the result is concentrated in a small interval.

Finally, we draw the statistical chart with each weight vector, such as scatterplot, to clearly express the weight of the result of the roller coasters. The charts are shown in previous figure 7.

We can clearly see that the weight of each roller coaster is concentrated in a small interval, which indicates that the roller coasters are not well-distinguished. We infer that the concentration may result from the low correlation coefficients, we need to consider a better method to solve the problem.

4.3 Linear Regression

The third modeling method we use is Linear Regression. We can regard the properties of roller coasters as independent variables, and the online scores as dependent variables. Based on the samples, each data can be viewed as a mapping from the independent variables, which are the properties, to the dependent variables, which are scores online. As each information is expressed numerical, we can find the function from the independent variables to the dependent variables through linear regression from the data. ^[5]

Let x_1 to x_9 respectively denote the nine properties respectively. Let y denotes online scores. The value of the independent variables and dependent variables is the numbers of each option. We utilize regression formula 13.

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4 + \beta_5 x_5 + \cdots + \beta_9 x_9 \quad (13)$$

Let X denotes the independent variables matrix; Y denote dependent variables matrix; β denotes coefficient matrixes. We apply Least Square Regression Method to the issue, of which the formula is shown in formula 14:

$$\beta' = (X^T X)^{-1} X^T Y = \left(\sum x_i x_i^T \right)^{-1} \left(\sum x_i y_i \right) (i \in \{i \in N^* | i \leq n\}) \quad (14)$$

The formula is set to solve out the value of the coefficient matrixes of point estimation. With MATLAB giving solution, we obtain the coefficient matrixes which are presented in table 7:

Table 7: Linear Regression Coefficient

β_0	-18.2625
β_1	0.089799
β_2	0.08964
β_3	0.017563
β_4	0.010858
β_5	-0.00166
β_6	0.006959
β_7	8.52E-05
β_8	-0.00053
β_9	-0.01581

Point estimation possesses a drawback that it cannot express the accuracy of the data obtained. Thus we utilize interval estimation to reuse the Least Square Regression Method, the formula as in formula 15:

$$P\{\hat{\theta}_1 < \theta < \hat{\theta}_2\} = 1 - \alpha \quad (15)$$

θ denotes the parameters to be estimated of the ensemble; P denotes probability; $\hat{\theta}_1$ denotes Confidence upper limit; $\hat{\theta}_2$ denotes Confidence lower limit; α denotes reliability which satisfies $0 < \alpha < 1$. In this way, we obtain formula 16

$$P\{\hat{\beta}_1 < \beta < \hat{\beta}_2\} = 1 - \alpha \quad (16)$$

With the MATLAB program, we set α as 0.95, under which the regression coefficient bound is shown in table 8.

The residual graph is shown in figure 8. When examining correlation coefficients, we find the correlation coefficients are 0.336705.

Table 8: Linear Regression Coefficient Bound

	Lower Bound	Lower Bound
β_0	-28.9973	-7.52761
β_1	-0.35907	0.538668
β_2	-0.07901	0.258286
β_3	-0.04829	0.083415
β_4	0.005448	0.016268
β_5	-0.00453	0.001203
β_6	-0.00717	0.021093
β_7	-6.05E-07	0.000171
β_8	-0.00273	0.00167
β_9	-0.04853	0.01691

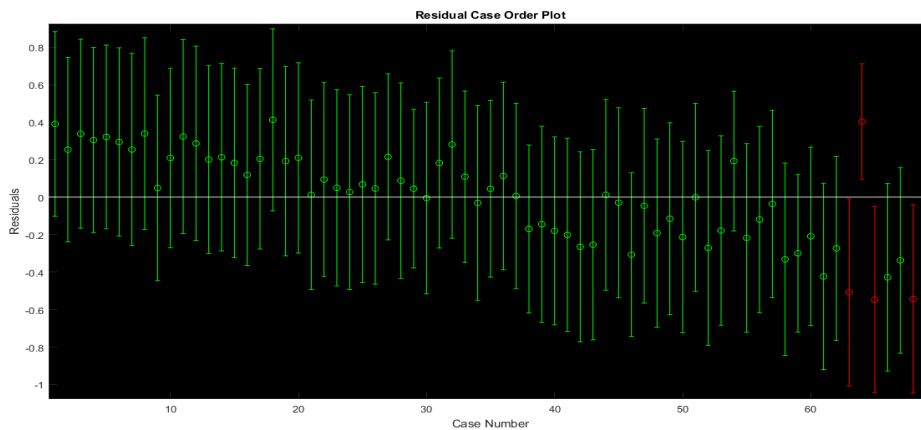


Figure 8: Residual Case Order Plot of Linear Regression

In light of the fact that the accuracy is relatively low, which is insufficient to reveal the features of each variable precisely, we consider taking the advantage of other methods. Principal Component Regression is applied later as an optimized method in part 5.1.

4.4 KNN Algorithm

In accordance with the given data, we try to use the data of which the online scores are matched to conduct the KNN algorithm to highly merge the vast amount of the data and find the shared features and characteristics of each sample to obtain the common properties of the roller coasters under similar condition to determine the relationship. ^[6]

We utilize Mahalanobis distance distinction to operate these data, which is processed after principal component analysis and features eradicating the dimension of each independent variables. The formula is as the following formula 17.

$$d(x, y) = \sqrt{(x - y)\Sigma^{-1}(x - y)^T} \quad (17)$$

Among the formula, x and y denote two row vectors; Σ denotes the covariance matrix; $d(x, y)$ denotes the obtained Mahalanobis distance of the data.

For the accuracy, we correctly categorized 57 samples out of 69, achieving an accuracy of 83%. We made an optimization of this method in 5.2.

5 Optimization

5.1 Principal Component Regression

Principal Component Regression suits explicitly for the problems that have a vast amount of independent data types, not all of which are tightly connected to the dependent data, which means some of the data are loosely related to the data. In view of considering that our problem has 9 independent variables, the method is highly compatible with our research.

We can still do as part 4.3, regarding the properties of roller coasters as dependent variables and the online scores as independent variables. We try to reduce the dimensionality, diminishing the vast amount of the original data and variables into fewer data and variables, while the new variables can retain the information in the original data by and large. ^[7]

We utilize the 9 original variables mentioned in 3.4 as the original data. We still use X to denote independent variables matrixes and Y the dependent variables. The original variables are $x_p (p \in \{p \in N^* | p \leq l\})$; the new variables are $z_q (q \in \{q \in N^* | q \leq p\})$. We use m to denote the number of samples and use l to denote the number of variables in each sample.

Applying Least squares regression, point estimation and interval estimation method which has previously been mentioned, we obtain the principal coefficient matrix β' as shown in table 9 with formula 18.

$$y^* = \beta_1' z_1 + \beta_2' z_2 + \beta_3' z_3 + \cdots + \beta_5' z_5 \quad (18)$$

Table 9: Coefficient Matrix of principal component

Point Estimation	Interval Estimation	
-17.9038	-28.228	-7.5796
0.017885	0.002086	0.033684
-0.01057	-0.01941	-0.00174
0.013596	-0.00226	0.029449
0.078525	0.004955	0.152094
-0.02736	-0.06604	0.01132

The correlation coefficient of this method is 0.322210105118927. Although there is no discernable elevation in the coefficient, the method focuses more on the principal variables.

Ultimately, we conduct the inverse standardization process and obtain the equation interpreted in the original data, which is formula 19, and the final coefficient matrix, as shown in table 10.

Table 10: Final Coefficient Matrix of original variables

Point Estimation	Interval Estimation	
-17.9038	-28.228	-7.5796
0.075562	0.024277	0.126847
0.033707	0.016531	0.050884
0.020766	0.019247	0.022286
0.010701	-0.05472	0.076122
-0.00221	0.007277	-0.0117
0.008843	0.01558	0.002106
0.000105	-0.00306	0.003271
-0.00103	-0.01252	0.010466
-0.00745	-0.00913	-0.00578

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \cdots + \beta_9 x_9 \quad (19)$$

5.2 Bayes Distinction

Bayes Distinction ideally satisfies the requirements of such issue that each individual of the ensemble exists at different frequencies, which indicates that we need to take into consideration that the different possibilities that each individual exists. As for our research, each roller coaster is obviously impossible to appear at identical frequencies, so we apply Bayes Distinction to our study.

In the distance distinction method above, it does not take into account the frequency of each sample as a whole and does not take into account the loss caused by the wrong distinction. The Bayes distinction method modifies on the basis of distance distinction, and the formula is defined as in formula 20: [8]

$$P(B_i | A) = \frac{P(A | B_i)P(B_i)}{\sum P(A | B_i)P(B_i)} \quad (20)$$

Among which $P(B_i | A)$ represents a posteriori probability; $P(A | B_i)$ represents a prior probability; $P(B_i)$ represents the frequency at which the sample appears; Σ represents the total covariance matrixes. The distinction rule is that the posterior probability is the highest and the average wrong distinction loss is the lowest, which brings out the rule is as follows: If the condition meets the following formula 21:

$$P(G_i | x_0) = \frac{p_i f_i(x_0)}{\sum p_j f_j(x_0)} = \max_{1 \leq i \leq k} \frac{p_i f_i(x_0)}{\sum p_j f_j(x_0)} \quad (21)$$

Then we categorize x_0 into G_i , among which G_i is the ensemble, $f(x)$ is the probability density function of G_i , p_i is prior probability of G_i , which is the probability that it belongs a certain category when sample x_0 occurs, and k is the number of G_i . The solution formula for distinction analysis is as the following formulas 22-23:

$$ECM = \sum_{i=1}^k p_i \sum_{j \neq i} C(j/i) P(j/i) \quad (22)$$

$$p(j/i) = P(X \in D_j / G_i) = \int_{D_j} f_i(x) dx \quad i \neq j \quad (23)$$

In this case, $P(j/i)$ represents the conditional probability of wrongly categorizing the sample of G_i to the ensemble G_j . $C(j/i)$ is the loss caused by this categorization. D_k is a division of a set of distinction samples. ECM is the average wrong distinction loss. The solution to a Bayes distinction analysis is to make the smallest set of solutions.

We divide the result of Bayes distinction into 5 categories, which are less than 4, 4 to 4.5, and 4.5 to 5. For the training set, if the online score lies in 4.5 to 5, we define the roller coaster as category 1. Likewise, we define the roller coaster of which the score is from 4 to 4.5 as category 2. We randomly pick out a certain amount of data from ALL the data which has no score online or the score is lower than 4 and define them as category 3. Using the MATLAB program, we still use all the data with the online score to carry out Bayes distinction solution.

The result is shown in the appendix, part of which is as following figure 8-9 and table 20. For instance, the number “36” shows that there are 36 samples with sit down type are judged as Category 1, which is the high score category.

For the accuracy, we correctly categorized 59 samples out of 69, achieving an accuracy of 85%, which is relatively higher than the accuracy obtained from KNN algorithm. The following table 11 is a part of the result.

Table 11: Bayes Distinction Result

Height (feet)	Speed (mph)	Length (feet)	Duration (sec)	Number of Inversions	costerbuzz	input	mahal 5	bayes 9 probability			bayes 9	Name
205.0	74.0	5740.0	150.0	4	4.93785	1	1	0.82568857	0.16064424	0.01366719	1	Steel Vengeance
325.0	95.0	6602.0	140.8	0	4.85632	1	1	0.76659792	0.22189142	0.01151067	1	Fury 325
181.0	70.0	4400.0	102.0	0	4.83099	1	1	0.69684335	0.21206118	0.09109546	1	El Toro
207.0	73.0	3800.0	150.0	0	4.80723	1	1	0.69546	0.21290759	0.09163241	1	Lightning Rod
310.0	93.0	6595.0	140.0	0	4.77173	1	1	0.437385	0.5197592	0.04285581	2	Millennium Force
121.0	57.0	4990.0	220.0	2	4.76087	1	3	0.66375696	0.27406183	0.06218121	1	Twisted Colossus
110.0	60.0	4725.0	150.0	0	4.73288	1	1	0.6757347	0.21247406	0.11179124	1	Boulder Dash
109.0	55.0	3320.0	125.4	3	4.7037	1	3	0.45146701	0.3686575	0.18187549	1	Wicked Cyclone
159.0	67.0	6442.0	165.0	0	4.69832	1	1	0.92323924	0.06599043	0.01077033	1	Voyage
105.0	70.0	4450.0	150.0	2	4.6963	1	1	0.62207853	0.30089748	0.07702399	1	Maverick
179.0	70.0	3266.0	116.3	1	4.68571	1	3	0.31827035	0.46895769	0.21277196	2	Iron Rattler
208.0	77.0	5400.0	155.0	0	4.68571	1	2	0.36009724	0.53051281	0.10938995	2	Superman the Ride
165.0	72.0	3100.0	120.0	2	4.68917	1	1	0.65987442	0.21939201	0.12073357	1	Goliath
306.0	92.0	5486.0	208.0	0	4.66316	1	1	0.41569912	0.52652983	0.05777105	2	Leviathan
80.0	57.0	2900.0	90.0	0	4.65306	1	1	0.61262781	0.18566382	0.20170837	1	Ravine Flyer II
107.0	68.0	2937.0	87.0	3	4.63218	1	1	0.77672073	0.14065304	0.08262623	1	Outlaw Run
200.0	73.0	4760.0	158.0	0	4.61667	1	1	0.32201285	0.58802171	0.08996544	2	Mako
78.0	45.0	3200.0	120.0	0	4.56299	1	3	0.06005057	0.27162472	0.66832471	3	Phoenix
167.0	68.0	4124.0	160.0	7	4.54397	1	1	0.37212078	0.54317252	0.0847067	2	Banshee
100.0	52.0	2744.0	100.0	3	4.54321	1	3	0.37872831	0.3688639	0.2524078	1	Storm Chaser
109.2	53.0	3265.0	120.0	0	4.51948	1	3	0.75029362	0.13796963	0.11173675	1	Mystic Timbers
230.0	77.0	5318.0	201.4	0	4.51163	1	1	0.40486406	0.50442969	0.06070626	2	Behemoth

We also make various charts and tables to exhibit our results, part of which are as the following figures 9-10 and table 12:

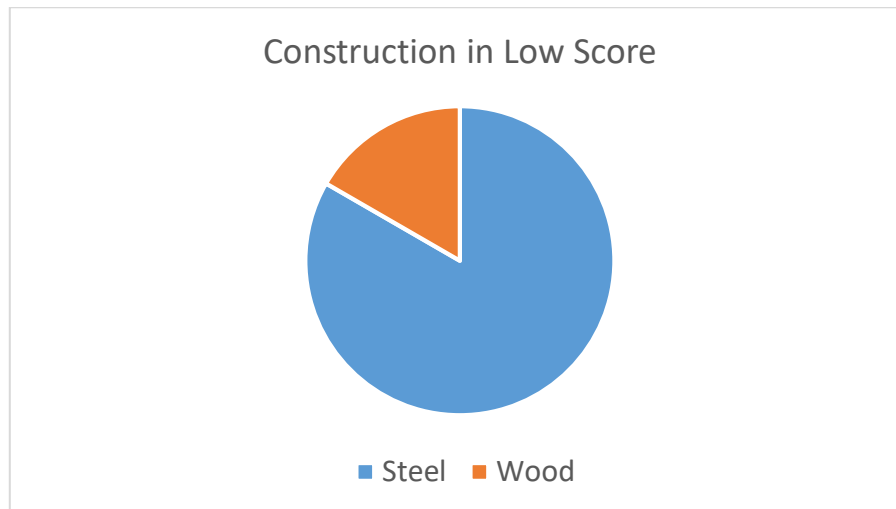


Figure 9: Bayes Result of Construction in Low Score

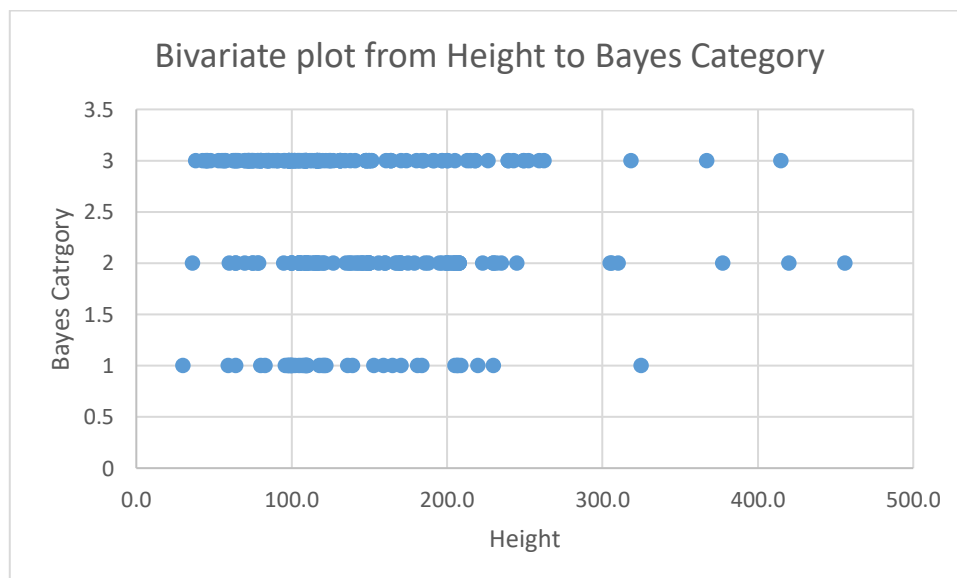


Figure 10: Bivariate plot from Height to Bayes Category

Table 12: Bayes Category to Type

	Sit down	Inverted	Stand up	Suspended	Flying	Wing
Category1	36	0	0	0	0	0
Category2	59	17	2	2	5	3
Category3	132	21	0	4	0	1

From the results given, we can clearly figure out the trend that the roller coasters which are in the place far away from North America tend to have a high score, especially the ones located in Middle East, Oceania, and Russia. The roller coasters that are made from wood are more likely to have a higher score. A newly opened roller coaster is more welcomed. If the roller coaster is relatively higher, it is more possible to achieve a better score. 2 and a half minutes and 60 mph are a proper time for a loop and a satisfactory speed respectively. If the number of inversions is too high, it may conversely do harm to the passion of tourists to ride.

5.3 BP Neural Network Fitting

BP Neural Network is a kind of multilayer feed-forward network, which highly fits for the problem that there are data with a certain scale, the relationship between which is not too complicated to identify. When it comes to our target, we have a middle-sized database, and since the fitting process is not too intricate, the model can be applied to our goal.

We utilize BP neural network fitting as another method to promote the accuracy of the regression. BP neural network aims to encode itself with its high-dimensional features and to carry out dimension reduction processing towards high-dimensional data. It is marked by a feature extraction model with unsupervised learning, which can also combine a few basic features to obtain higher-layer abstract features.^[9]

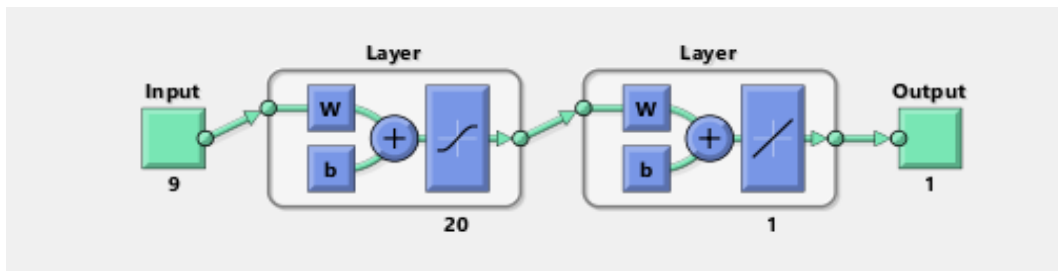


Figure 11: BP Neural Network Structure. The layer number, which is 20, does not consume too much time while the result is satisfactory.

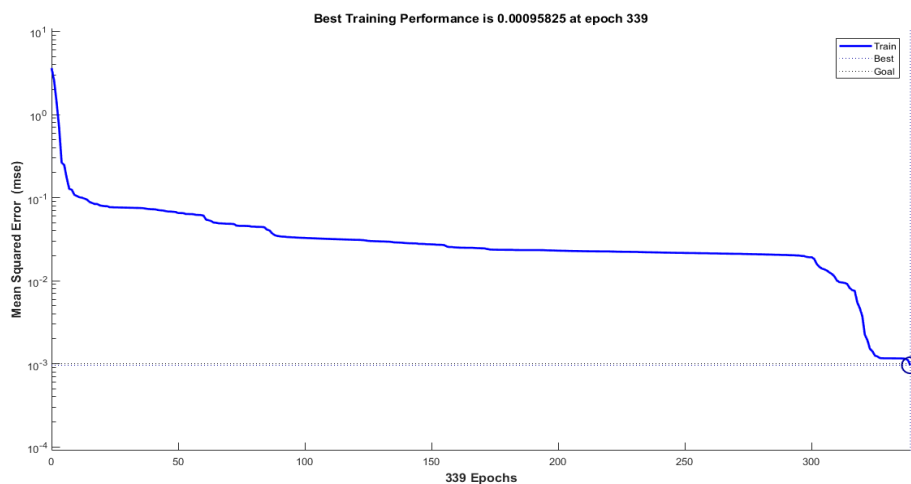


Figure 12: the performance plot of BP Neural Network. The training performance is improving rapidly.

We utilize Tangent Sigmoid function as the transfer function; we use Levenberg Marquardt algorithm (trainlm) as the training algorithm; we use the Gradient descent with momentum weight and bias learning function (learnngdm) as the learning algorithm; we use the mean square error (MSE) method as the learning function. The structure of the network and the performance plot are shown in previous figure 11 and 12.

Applying the MATLAB program, we use all the data with online score to carry out the BP neural network fitting.

We consider dividing the learning samples into three groups, each time using two of the groups to carry out a model and then test it with the test set. In light of the fact that there are mere 69 training data, it is not sufficient enough for us to conduct in this way. Hence, we use all the training data to training the BP Neural Network Algorithm. The result is in the appendix, part of which is as the following table 13.

Table 13: BP Neural Network Result. The error of some numbers is lower than 1%.

Duration (sec)	Number of Inversions	costerbuzz	BP	Score	Name	Park	City/Region	Cit
150.0	4	4.93785	4.93413502	9.99934627	Steel Vengeance	Cedar Point	Sandusky	Ohio
140.8	0	4.85632	4.86625756	9.99822441	Fury 325	Carowinds	Charlotte	North
102.0	0	4.83099	4.781252	9.99101099	El Toro	Six Flags Great Adventure	Jackson	New
150.0	0	4.80723	4.80708852	9.99997444	Lightning Rod	Dollywood	Pigeon Forge	Tenn
140.0	0	4.77173	4.78282655	9.99798246	Millennium Force	Cedar Point	Sandusky	Ohio
220.0	2	4.76087	4.76225957	9.99974652	Twisted Colossus	Six Flags Magic Mountain	Valencia	Califc
150.0	0	4.73288	4.62320178	9.97963471	Boulder Dash	Lake Compounce	Bristol	Conn
125.4	3	4.7037	4.68407238	9.99636796	Wicked Cyclone	Six Flags New England	Agawam	Mass
165.0	0	4.69832	4.71562668	9.99680635	Voyage	Holiday World	Santa Clause	Indiar
150.0	2	4.6963	4.70829667	9.99778402	Maverick	Cedar Point	Sandusky	Ohio
116.3	1	4.68571	4.69334898	9.99858512	Iron Rattler	Six Flags Fiesta Texas	San Antonio	Texas
155.0	0	4.68571	4.67979935	9.99890365	Superman the Ride	Six Flags New England	Agawam	Mass
120.0	2	4.66917	4.67616868	9.99869904	Goliath	Six Flags Great America	Gurnee	Illinois
208.0	0	4.66316	4.62402991	9.99268063	Leviathan	Canada's Wonderland	Vaughan	Ontario
90.0	0	4.65306	4.64953369	9.99934149	Ravine Flyer II	Waldameer	Erie	Penn
87.0	3	4.63218	4.63300263	9.99984576	Outlaw Run	Silver Dollar City	Branson	Missouri
158.0	0	4.61667	4.60768361	9.99830764	Mako	SeaWorld Orlando	Orlando	Florida
120.0	0	4.56299	4.51840222	9.99147074	Phoenix	Knoebels Amusement Park	Elysburg	Penn
160.0	7	4.54397	4.41895206	9.97576767	Banshee	Kings Island	Mason	Ohio
100.0	3	4.54321	4.59451963	9.9902454	Storm Chaser	Kentucky Kingdom	Louisville	Kentucky
120.0	0	4.51948	4.5023516	9.99670188	Mystic Timbers	Kings Island	Mason	Ohio
201.4	0	4.51163	4.50711877	9.99013105	Behemoth	Canada's Wonderland	Vaughan	Ontario

It can be seen that some of the predicted data run an accuracy that is higher than 99%.

5.4 XG Boosting Algorithm

We utilize XG Boosting algorithm to obtain the average value of each method of the samples. The basic formula is as the following formula 24

$$L(\theta) = \sum_{i=1}^n l(y_i, \hat{y}_i) \quad (24)$$

In the formula, $L(\theta)$ denotes the overall loss of each classifier, y_i denotes each classification function, and \hat{y}_i is a function of each classifier to reduce the loss. y_1 denotes the original result of Principal Component Analysis. x_2 denotes the result of Bayes distinction. x_3 denotes the original result of BP neural network fitting. For each category in Bayes distinction, we utilize the mid-value of each interval to numerate each category. We divide the result of Bayes distinction into 5 categories, which are less than 4, 4 to 4.5, and 4.5 to 5. Therefore, we use 3.75, 4.25, and 4.75 to denote the 3 result of the categories.

The main theory of BOOST algorithm is as follows. A complicated issue can be better estimated when synthesizing the judgment of each expert than that of a sole expert. For each step, we generate a model to accumulate each model to a whole model, which enables us to

analyze the problems. Hence, we need to assemble several weak learners into a strong learner by determining the loss functions, $(y_i)^{\wedge}$, to minimize the error and loss of misjudgment.

We input the predicted result of the three learners into the algorithm as the learning set and the real result as the target goal. We regard test set in the Bayes distinction and BP Neural Network as the testing set. With the help of XG Boosting module in PYTHON, we are able to determine the weight of the three learners to generate the final result. ^[10]

We utilize a formula to measure the error of our estimation, reaping an average score same as the original result and receiving almost a full score of 10, which shows that this model can successfully reflect the trend. The formula is as the following formula 25.

$$S_k = \max \left(0, 10 - 10 \times \left| \frac{\log_{10} \left| \frac{x_{predict}}{x_{real}} \right|}{5} \right| \right) \quad (25)$$

In the formula, S_k denotes the score of the data, while $x_{predict}$ and x_{real} respectively denote the predicted value and the real value of the data.

We discover that many roller coasters have the same value of XG Boosting, which may due to the reason that there is too less training set while too much testing set. Since the BP Neural Network reaps a relatively accurate outcome among the 3 optimized models, we decide to use the result of the BP Neural Network Fitting as the final score and ranking if the outputs of XG Boosting are identical.

6 Analysis of the Model

6.1 Sensitivity Analysis

Sensitivity analysis is a method of studying and analyzing the sensitivity of the model to changes in system parameters or surrounding conditions. In the optimization methods of our team, it can detect the stability of our model, especially when the given data is not accurate.

In this part, we will mainly discuss the sensitivity of the application part. If we give the test set of the data an increase or a decrease of 1%, by changing the value of the original data matrix on the program, we discover that the output data of the principal component regression changes precisely 1%; almost all the results in the Bayes Distinction part have no difference in categories; the majority of the output of BP neural network model fluctuates 1% approximately. The output after the change is small enough for us to make a further adjustment. Therefore, it is acceptable in the modeling. This sensitivity analysis also indicates that our model has universality and can be applied to more situations. For instance, if there is some error in the data, our final result does not vary rapidly correspondingly. Therefore, our model is relatively stable. The data of Sensitivity Analysis can be referred to the appendix.

6.2 Strength and Weakness

The strength of the model for the rating of roller coasters and the algorithm being used mainly include the following aspects:

The methods applied in this model includes both qualitative and quantitative analysis, and different conclusions from various methods can be obtained through our modeling process. As for the qualitative analysis, for instance, from the results of Bayes Distinction, we can figure out the trend that the roller coasters which are in the place far away from North America tend to have a high score and that the roller coasters made from wood are more likely to have a higher score. A newly opened roller coasters are more welcomed. If the roller coaster is relatively higher, it is more feasible to achieve a better score. 2 and a half minutes

and 60 mph are a proper time for a loop and a satisfactory speed respectively. All of these are obtained through the qualitative analysis which provides us with valuable information. As for the quantitative analysis, it is evident that nearly all the algorithms being applied need the input of the database and the whole rating process our model depends on needs the analysis and testing of quantitative properties of every roller coasters. Far from the common evaluating methods based on the riders' subjective inputs currently, our model is based on facts, analysis, and training, which will undoubtedly provide more accurate and scientific results on the ranking.

Our methods also take into consideration both continuous and discrete variables, the fact of which makes our model especially suitable for the question's requirement. For instance, in our optimized model, the data of discrete independent variables, like the material used for the construction of the roller coasters, are mainly analyzed by Principal Component Analysis, which is suitable for the processing of discrete data. On the other hand, Bayes Distinction mainly served to analyze the continuous variables including average speed, maximum height, so and so forth. For the XG Boosting algorithm, it is suitable for all independent variables and can successfully synthesize the result of the optimized model. Thus, our model has sufficiently taken into account the property of different types of data, and the result yielded will, in turn, be highly persuasive.

Another outstanding point of the whole modeling process is the variety of methods being used---from the basic statistics and linear regression to the optimized model of BP Network Fitting and XG Boosting algorithm. The final ranking is produced through exploration on various methods, and we have been continuously evaluating the viability of different models and thinking about how to improve our results through more advanced methods further. It is our endeavor for excel that guarantees a more accurate and suitable method as a whole for the quantitative analysis of the rating. The multiplicity of methods not only shows our clear logic chain from the perspective of pragmatic problem solving but also ensures a more stable and precise result.

The model is also propagable as a standard method for the rating of any new roller coaster given. In other words, besides the 300 roller coasters provided in the table, if any new roller coaster with basic properties given are added to the database of our whole modeling process, a rating can also be produced and be added to the original ranking.

Furthermore, our model is highly flexible and can be applied to various situations. For instance, if another new property, no matter continuous or discrete, is added to all the roller coasters, the method and logic behind of our modeling can also be applied, since the ranking is based on a whole series of analysis instead of randomly assigning weight to each property. Thus, a different ranking is likely to be produced. Besides, it is also worth noticing that if more roller coasters with ratings are added to the training set in the XG Boosting algorithm, a more accurate result will be produced. This enhancement in accuracy is because specific optimized methods, like XG Boosting algorithm, yield results based on the self-learning of data input, which indicates more data analysis and more source for learning will boost the accuracy and stability of the method. Therefore, the model we proposed can constantly learn from the additional data input, so it is highly flexible and makes possible the dynamic adjustments, which makes the model suitable for being applied under various circumstances.

Finally, our model yields the precise rating results instead of just the ranking of the top 10. Actually, it is also possible to produce the quantitative result of any roller coaster given. This result will definitely be better than the vague ranking result which is a much weaker conclusion compared. The accurate rating can reflect the difference in a quantitative way between each roller coasters and give the potential riders more compatible information.

The weakness of our model mainly includes the aspects following:

The data of the roller coasters given is sometimes not ample and sufficient enough for the evaluation of one particular independent variable. This sufficiency will, in turn, cause the data input being inaccurate since some information is missing and cannot be applied to the analysis. Though interpolation is done during the data cleaning process, there are still some data left vacancy because of the insufficiency of existed data, for the loss of a large quantity of data will make the interpolation process meaningless. So the loss of data will affect our final rating, even though the methods we applied have considered maintaining the original information as much as possible.

Besides, there are only 300 pieces of information given for the learning and testing of data, the fact of which will unavoidably make the method like XG Boosting not accurate enough. For methods like this, more pieces of data being learned will further boost the accuracy. However, despite some deficiency, it is still a suitable method for the synthesis of the optimized result and provide us with the relatively more stable and precise rating result.

7 Comparison of the Top 10 Roller Coasters

To give out the final ranking of our model, we decide to use the result of XG Boosting as the ranking criterion. As mentioned above, we find a flaw that some scores of roller coasters of XG Boosting are identical, so it is difficult for us to distinguish which roller coasters should be the top ones. In light of the high accuracy of the BP Neural Network Fitting given in 5.3, we decide to use the result of the BP Neural Network to rank the roller coasters if the results of XG Boosting are exactly the same. In other words, the first ranking criterion is the result of XG Boosting, and the second one is the result of BP Neural Network. The top 10 roller coasters and their scores are shown as following table 14.

Table 14: Final Ranking

Number	Name	Park	City/Region	City/State/ Region	Country/ Region	Geographic Region
257	T Express	Everland	Yongin-si	Gyeonggi-do	South Korea	Asia
9	Anaconda	Walygator Parc	Maizieres-les- Metz	Lorraine	France	Europe
66	Crazy Coaster	Loca Joy Holiday Theme Park	Yongchuan	Chongqing	China	Asia
10	Apocalypse	Six Flags America	Upper Marlboro	Maryland	United States	North America
33	Big Thunder Mountain	Disneyland Resort Paris	Marne la Vallee	Ile-de-France	France	Europe
273	Tonnerre de Zeus	Parc Asterix	Plailly	Picardie	France	Europe
143	Jupiter	Kijima Kogen	Beppu	Oita	Japan	Asia
59	Coaster Through the Clouds	Nanchang Wanda Theme Park	Xinjian	Nanchang, Jiangxi	China	Asia
87	Firehawk	Kings Island	Kings Mills	Ohio	United States	North America
226	Silver Star	Europa Park	Rust	Baden Wuerttemberg	Germany	Europe
Number	Inversions	Status	Construction	Type	Drop	Year/Date

	(YES or NO)				(feet)	Opened
257	NO	Operating	Wood	Sit Down	150.9	2008
9	NO	Operating	Wood	Sit Down	40.0	1989
66	NO	Operating	Steel	Sit Down		2013
10	YES	Operating	Steel	Stand Up	90.0	2012
33	NO	Operating	Steel	Sit Down	39.3	1992
273	NO	Operating	Wood	Sit Down		1997
143	NO	Operating	Wood	Sit Down		1992
59	NO	Operating	Steel	Sit Down	255.9	2016
87	YES	Operating	Steel	Flying		2007
226	NO	Operating	Steel	Sit Down	219.8	2002
Number	Height (feet)	Speed (mph)	Length (feet)	Duration (min:sec)	Duration (sec)	Number of Inversions
257	183.8	64.6	5383.8		138.6	0
9	118.1	55.9	3937.0	2:10	130.0	0
66	108.3	52.8	2870.8		178.4	10
10	100.0	55.0	2900.0	2:00	120.0	2
33	72.2	40.4	4921.3	3:56	236.0	0
273	98.0	52.0	4044.0	2:05	125.0	0
143	138.0	57.0	5249.3	2:34	154.0	0
59	242.8	84.5	5105.0	4:12	252.0	0
87	115.0	50.0	3340.0	2:10	130.0	5
226	239.5	78.9	5315.0	4:00	240.0	0
Number	BP	XGBoosting	G Force	Vertical Angle (degrees)		
257	6.706793	4.68571		77		
9	6.004326	4.68571				
66	5.979574	4.68571				
10	5.926418	4.68571				
33	5.87154	4.68571				
273	5.777834	4.68571				
143	5.754164	4.68571		45		
59	5.726858	4.68571				
87	5.54524	4.68571	4.3			
226	5.489252	4.68571	4	68.5		

Apart from the score online we use in the modeling part, we find a second scoring website from the website Coaster Critic ^[11], of which the score can be referred to the appendix.

Comparing the two top 10 roller coasters, the most significant difference we find is that our scores focus on not only the roller coasters in the US but also the roller coasters on a world scale. The location of our top 10 roller coasters includes the US, France, Korea, China, and Japan, which demonstrates that we truly achieve the goal that selecting the roller coasters based on objective and quantitative data rather than personal, subjective opinion. We are able to recommend the roller coasters only from the properties of themselves rather than personal opinions.

There is also some consistency between the two scores. For instance, wooden roller coasters are both highly rated, of which the reason may lie at people prefer the obsolescence of conventional wood roller coasters. The top 10 roller coasters of both are less likely to have inversions comparing with the roller coasters ranked after 10. The opening years both cover a wide range, from the 1980s to two years ago. The roller coasters with average speed, length, duration, or height are both leading the top of the rank.

We also compare our result with other websites, such as the result from MostLuxuriousList^[12] or TheTopTens®^[13], the result of which can be seen in the appendix (some roller coasters in the two websites are missing in the given database of roller coasters). The results of the comparison are also similar to those of the previous website. The top roller coasters online concentrate in the US, while the top coasters our model gives out involves a broader range. The comparison with the ranking from MostLuxuriousList and TheTopTens® also reveal that the parameters of the top coasters lie in an average interval, which manifests that middle-interval coasters are more warmly welcomed.

8 Concept and Design for a User-friendly App

The user-friendly app we construct mainly aims to satisfy the riders' needs on roller coaster riding selection and meet individual demands. The app mainly contains 3 aspects--- recommendation of roller coasters based on all the applied riders' experience on a global scale, the specific recommendation of roller coasters to individuals after the data processing and the analysis on the individual's past preference, and the selection of roller coasters by the filter to meet the users' needs. The roller coasters' own prosperities form the primary database for the selection and recommendation, and the algorithm will help with the analyzing process. Figure 13 shows a flowchart of the principles of our desired application, and figure 14 shows the effect of the app.

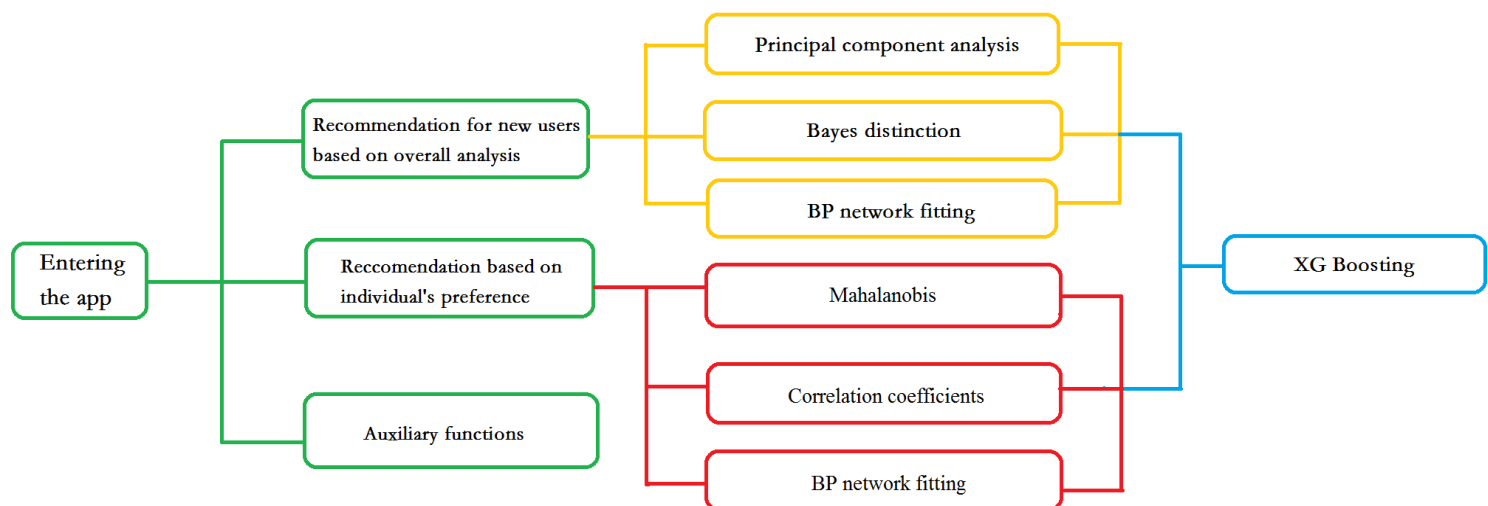


Figure 13: Flow chart of our desired application. We set various functions and models to realize the functions.



Figure 14: Desired Panel of our application. Our panel is attracting!

8.1 Initial Recommendation

First, the app will ask for the individuals' personal information including the region they live in. Then it can first select the roller coasters from that region and make corresponding recommends. The registered riders will be required to rate the roller coasters they have rid after the thrilling experience, and each piece of information they record will be put in the database for the analysis. In order to encourage the registered riders to make a contribution to the database, some rewards may be provided. The question may involve the following aspects: the feeling after the ride, the degree of excitement and stimulation based on the individuals'

experience, the rating of roller coasters as a whole, so and so forth. All of these questions are the users' subjective inputs, based on the rating they have given, and we can use them to refresh the ranking of the roller coasters at every moment. In this way, the personal information can be turned as the input of quantitative analysis, improving the original model's accuracy and stability. At the same time, this ranking would be used to provide the new users with the top roller coasters and encourage them to experience the best ride.

In order to achieve the goal, we can do as what we have done in the previous parts, using the Principal Component Analysis, Bayes Distinction, and BP Neural Network to obtain the score of each roller coaster and use XG Boosting algorithm to synthesize the result of the three methods to achieve the best accuracy of the prediction. Since the results are based on common preference, it will never prove to be fallible by a typical user and tend to recommend the roller coasters that most users want to ride. The method suit well for the problem because when the scored data increases as it continues to be collected from the users, the model will be progressively accurate and achieve a more precise recommendation.

8.2 Recommendation Base on Preference

The information continuously provided by one individual---the track record---can also provide useful information on the individual's own preference. We have two functions, behind which the basic algorithm tries to determine the correlation of the riders' record and the roller coasters in the database, and the correlation of the riders' identity and the ones in the database. We can recommend the roller coasters that manifest a more substantial correlation with the roller coasters that have already been rid, or recommend the roller coasters that the users which have a more significant correlation with the users have rid, eliminating the roller coasters that have been rid by the users.

To define the similarity between the historical data of the users and the data in the database, we can set each data of the user or the data in the database as a row vector and calculate the correlation coefficients between the two. Then we can rank the roller coasters by the correlation coefficients from the largest one to the smallest one, recommending the ones with several largest correlation coefficients; additionally, we can rank the users who show high correlation and recommend the coasters which the similar users have rid. We can also calculate the Mahalanobis distance between the row vectors previously mentioned and rank the roller coasters as above, taking the advantage that the method does not take the dimension of the data into account. Based on the algorithm, the app can thus successfully achieve its second crucial function, and make the recommendation based on the quantitative analysis, spotting the users' need and saving the users' time for searching.

Besides, we can use Neural Network to achieve a personal and private recommendation. We can gather the information of the users, such as gender, region, so and so forth, treating them as independent variables as well as the properties of the roller coasters. In this way, the recommendation of the program can not only take the information of the roller coasters into consideration but also take the properties of the users into account, which gives rise to the exact match between the users and the roller coasters.

8.3 Search Engine for Roller Coasters

The app could also set up a selecting system to meet the riders' special needs. The system will be much like a search engine, but it will be entirely based on the property of the roller coasters. To make the sifting process more user-friendly, the options for the potential riders to choose will not include specific numbers. For instance, if the potential riders want to select a roller coaster with longer duration time, the search engine will not require them to put in specific numbers, but only choose from different levels such as short(30-60sec), medium(60-

120sec), and long(>120sec). Different selecting options will thus minimize the number of roller coasters based on the rider's demand and correspondingly make the proper recommendation.

To algorithmically achieve this goal, we can treat the word typed by the users as a string and find the strings in the database of which the substrings include the string which the users type and print the name of the corresponding roller coasters.

8.4 Auxiliary functions

Besides from the main purposes, auxiliary functions may also be included. First, a community will be set up to let the riders share their own riding experience, which may boost their sense of belonging with others who also like roller-coaster riding. They may even find the app useful as it can allow them to make friends with those who share the same interest with them. Besides, basic information of the roller coaster sites around the globe will be provided, in the form of both pictures and videos to give the potential riders a real sense of spectacularity, and every rider is welcomed to write their own experience and comments. For those especially love the thrilling feeling, they can also keep a journal in this app, and write down anything they want to recall about every one of their stimulating experience. Furthermore, up-to-date news about the roller coasters around the world will be timely reported by converging the information online, capturing the riders interest and promote them to have a try. Some related commercial products like keychains and postcards could also be provided after the cooperation with certain entertainment companies.

In brief, the app we construct uses the algorithm and quantitative analysis to meet the potential riders' needs and help them decide the best option, guaranteeing them a satisfying and enjoyable experience.

9 Conclusion

In the entire model, we first do some research on the background of the issue and have a rough understanding. Then we conduct some assumption to simplify the problem and clarify the notation that we will use in the following part of the essay. Next, we cleaned the data to eliminate the noise and inconsistency of the data. We examined the ranking given by ourselves and the score online to solidify the foundation that we can use the score online to train our model.

We employ several models with optimization to carry out the most comprehensive and matched model to the problem, with each model possessing distinctive features. The Principal Component Analysis is employed for coping with continuous data, while the Bayes Distinction model is specialized in dealing with discrete data. BP Neural Network Fitting can deal with both kinds of data and enhance the model's stability and accuracy, enabling the self-improving and training of the model. To combine the strength of all the models and avoid the drawbacks, we apply XG Boosting to synthesize the three models over this issue. Finally, we include the sensitivity analysis in our essay to ensure our result's stability. We also evaluate the strength and weakness of our model as a whole.

We reap both qualitative and quantitative results of our analysis. We not only find what kind of roller coasters are more likely to be welcomed but also explore the top ten roller coasters in the world and compare the similarities and the differences of the result with the existing results online.

Based on all the work we have done previously, we illustrate the concept for an user-friendly app which can recommend roller coasters for any common users. It combines the work we have done and demands of the users to achieve the model and algorithms' pragmatic value.

10 Reference

- [1] <https://wenku.baidu.com/view/c6e8f183b9d528ea81c779af.html>, Clustering Analysis with Matlab.
- [2] <https://coasterbuzz.com/RollerCoasters/Top100>, Top 100 Roller Coasters: The CoasterBuzz 100.
- [3] http://blog.csdn.net/MATLAB_matlab/article/details/59483185?locationNum=10&fps=1, MATLAB principal component analysis.
- [4] Mingbei C., Gang H., Guoufu Z. Comprehensive evaluation of takeaway website based on AHP method——Eleme website as an example [J]. Modern Business, 2015, 12: 57-58.
- [5] <https://wenku.baidu.com/view/99c8408e6529647d272852cd.html>, Interval estimation and linear regression analysis with MATLAB.
- [6] Jiang W. Comparative Study of Fisher Discriminant and Mahalanobis Distance Discriminant [J]. Journal of Ningbo Polytechnic, 2017, 21(5): 91-94.
- [7] Yimeng F. Analysis of influencing factors of customer purchase behavior in E-commerce [J], Industrial & Science Tribune, 2014, 13(8): 138-139.
- [8] Haiwei W. Yu X., Yalin W. A bivariate hierarchical Bayesian approach to predicting customer purchase behavior [J], Journal of Harbin Engineering University, 2007, 28(8): 949-954.
- [9] Wu P. Application of Cigarette Sales Forecasting Based on Neural Network [J], Computer Simulation, 2012, 29(3): 227-230.
- [10] https://blog.csdn.net/weixin_42029738/article/details/81675234, Hand-in writing XG Boost programs.
- [11] <http://www.coastercritic.com/roller-coaster-reviews/>, Coaster Reviews List – CoasterCritic.
- [12] <https://www.mostluxuriouslist.com/top-10-best-roller-coasters-in-the-world/>, Best Roller Coasters in the World - List of Top Ten Ranking.
- [13] <https://www.thetoptens.com/best-roller-coasters/>, Top Ten Best Roller Coasters - TheTopTens®.

11 Appendix

11.1 MATLAB Code

```
[m,n]=size(X);
x=X(:,1);
y=X(:,2);
temp=x(1,1);
count=0;
sum=0;
row=0;
for i=1:m
    if x(i)==temp;
        sum=sum+y(i);
        count=count+1;
    else
        row=row+1;
        Y(row,1)=temp;
        Y(row,2)=sum/count;
        count=0;
        sum=y(i);
        count=1;
        temp=x(i);
    end
end
Y(row+1,1)=temp;
Y(row+1,2)=sum/count;
x=Y(:,1);
y=Y(:,2);
y1 = interp1(x,y,a,'pchip') ;

d=pdist(A,'Mahal');
z= linkage(d);
H=dendrogram(z,293)
T=cluster(z,30);

stdr=std(x);
[n,m]=size(x);
sddata=x./stdr(ones(n,1),:);
[p,princ,egenvalue]=princomp(sddata);
per=100*egenvalue/sum(egenvalue);

[m,n]=size(a);
for i=1:m
    B{i}=a(i,:);
end
for i=1:m
    C{i}=zeros(n,n);
    for j=1:n
        for k=1:n
            C{i}(j,k)=B{i}(j)/B{i}(k);
```

```

        end
    end
end
eigenvector=[];
for i=1:m
    t=C{i};
    [x,lumda]=eig(t);
    r=abs(sum(lumda));
    n=find(r==max(r));
    max_lumda_A(1,i)=lumda(n,n);
    max_x_A{i}=x(:,n);
    max_x_A{i}=max_x_A{i}./sum(max_x_A{i});
    eigenvector=[eigenvector max_x_A{i}];
end

eigenvector=eigenvector';

%yangben

%g
%group

[m,n]=size(yangben);
for i=1:g
    groupNum(i)=0;
    group(i)=0;
    for j=1:m
        if yangben(j,1)==i
            group(i)=group(i)+1;
        end
    end
    if i==1
        groupNum(i)=group(i);
    else
        groupNum(i)=groupNum(i-1)+group(i);
    end
end
group;
groupNum;

% for j=1:n-1
% TotalMean(j)=0;
% for i=1:m
% TotalMean(j)=TotalMean(j)+yangben(i,j+1);
% end
% TotalMean(j)=TotalMean(j)/m;
% end

```



```
GroupMean=[];
for i=1:g
    if i==1
        low=1;
        up=groupNum(i);
    else
        low=groupNum(i-1)+1;
        up=groupNum(i);
    end
    matrix=yangben(low:up,:);
    MatrixMean=mean(matrix);
    GroupMean=[GroupMean;MatrixMean];

    for u=low:up
        for v=2:n
            C(u,v-1)=yangben(u,v)-MatrixMean(v);
        end
    end

C;
GroupMean;
V=C'*C/(m-g);
V_inv=inv(V);

GroupMean=GroupMean(:,2:n);
Q1=GroupMean*V_inv;

for i=1:g
    lnqi(i)=log(group(i)/m);
    mat=GroupMean(i,:);
    Q2(i)=lnqi(i)-0.5*mat*V_inv*mat';
end
lnqi;
Q2;

[u,v]=size(b);
result=[];
for i=1:u
    x=b(i,:);
    yy=Q1*x'+Q2';
    result=[result yy];
end
res=result';

[rows,cols]=size(result);
for i=1:cols
    iljj=0;
```

```
mlljj=result(:,i);
for j=1:rows
    iljj=iljj+exp(result(j,i)-max(mlljj));
end

for j=1:rows
    houyangailv(j,i)=exp(result(j,i)-max(mlljj))/iljj;
end
end
H=houyangailv';

iii=[];
for a=1:u
    k=max(H(a,:));
    for ii=1:g
        if k==H(a,ii)
            iii=[iii;ii];
        end
    end
end

clear c catagory detection i j k m n
for i=1:7
    c{i}=[];
end

for i=1:293
    catagory=b(i,1);
    for j =1:7
        [m,n]=size(c{j});
        if n~=0
            detection=0;
            for k =1:n

                if c{j}(5,k)==a(i,j)
                    c{j}(catagory,k)=c{j}(catagory,k)+1;
                    detection=1;
                end
            end
            if detection==0
                c{j}(5,(n+1))=a(i,j);
                c{j}(catagory,(n+1))=(c{j}(catagory,(n+1)))+1;
            end
        end
        if n==0
            c{j}(5,1)=a(i,j);
            c{j}(catagory,1)=(c{j}(catagory,1))+1;
        end
    end
end
end
```

```

for i=1:7
    c{i}=c{i}';
end

%
%p=[-1 -1 3 1;-1 1 5 -3];
%t=[-1 -1 1 1];
%BP
net=newff(minmax(p),[20 1],{'tansig','purelin'},'trainlm');
%
net.trainParam.epochs=10000;
net.trainParam.goal=0.001;
net.trainParam.show=50;
net.trainParam.lr=0.05;
net.trainParam.mc=0.9;
net=train(net,p,t);
A=sim(net,traini);

```

11.2 PYTHON Code

```

import xlrd
import xlwt
ExcelFile=xlrd.open_workbook(r'C:\Users\tianzhy\Desktop\COMAP_RollerCoasterData_2018 - Copy.xlsx')
sheet=ExcelFile.sheet_by_name('RollerCoasterData')
workbook = xlwt.Workbook(encoding = 'ascii')
worksheet = workbook.add_sheet('My Worksheet')
for i in range (1,219):
    temp = sheet.cell(i,17).value
    #temp = str.split(temp,":")
    timee=round(float(temp)*1440)
    worksheet.write(i, 0, label = str(timee))
workbook.save('Excel_Workbook.xls')

import pandas as pd
import xgboost as xgb
from sklearn import preprocessing
import numpy as np

train = pd.read_csv(r'D:\XGBoost_learn\click rate\train1.csv', header=0)
tests = pd.read_csv(r'D:\XGBoost_learn\click rate\test_pre.csv', header=0)
# trains=train.iloc[:, 1:].values
# labels=train.iloc[:,1].values
# test = tests.iloc[:, :].values
'''
train['time_stamp'] = pd.to_datetime(pd.Series(train['time_stamp']))
tests['time_stamp'] = pd.to_datetime(pd.Series(tests['time_stamp']))

```

```
train['Year'] = train['time_stamp'].apply(lambda x: x.year)#Year
train['Month'] = train['time_stamp'].apply(lambda x: x.month)#Month
train['weekday'] = train['time_stamp'].dt.dayofweek#weekday
train['time'] = train['time_stamp'].dt.time#time
tests['Year'] = tests['time_stamp'].apply(lambda x: x.year)#Year
tests['Month'] = tests['time_stamp'].apply(lambda x: x.month)#Month
tests['weekday'] = tests['time_stamp'].dt.dayofweek#weekday
tests['time'] = tests['time_stamp'].dt.time#time
train = train.drop('time_stamp', axis=1)
train = train.dropna(axis=0)
tests = tests.drop('time_stamp', axis=1)
tests = tests.fillna(method='pad')
'''

for f in train.columns:
    if train[f].dtype=='object':
        if f != 'shop_id':
            print(f)
            lbl = preprocessing.LabelEncoder()
            lbl.fit(list(train[f].values))
            train[f] = lbl.transform(list(train[f].values))

for f in tests.columns:
    if tests[f].dtype == 'object':
        print(f)
        lbl = preprocessing.LabelEncoder()
        lbl.fit(list(tests[f].values))
        tests[f] = lbl.transform(list(tests[f].values))
print("test")
print(tests.info())
# for f in train.columns:
#     if f != "":
#         train[f] = train[f].astype(float)

print(train.info())
# train = train.astype(float)
# tests = tests.astype(float)
trains = train.iloc[:, 1:].values
labels = train.iloc[:, :1].values
test = tests.iloc[:, 1:].values

feature_columns_to_use = ['wifi_strong1','wifi_strong2','wifi_strong3']

big_X = train[feature_columns_to_use].append(tests[feature_columns_to_use])
train_X = big_X[0:train.shape[0]].as_matrix()
test_X = big_X[train.shape[0]:].as_matrix()
train_y = train['shop_id']
gbm = xgb.XGBClassifier(silent=1, max_depth=10, n_estimators=1000, learning_rate=0.05)
gbm.fit(train_X, train_y)
predictions = gbm.predict(test_X)
```

11.3 Final Result

Number	Name	Park
257	T Express	Cedar Point

9	Anaconda	Carowinds
66	Crazy Coaster	Six Flags Great Adventure
10	Apocalypse	Dollywood
33	Big Thunder Mountain	Cedar Point
273	Tonnerre de Zeus	Six Flags Magic Mountain
143	Jupiter	Lake Compounce
59	Coaster Through the Clouds	Six Flags New England
87	Firehawk	Holiday World
226	Silver Star	Cedar Point
215	Road Runner Express	Six Flags Fiesta Texas
127	Hyper Coaster	Six Flags New England
64	Corkscrew	Six Flags Great America
105	Gao	Canada's Wonderland
181	Nessie Superrollercoaster	Waldameer
75	Do-Dodonpa	Silver Dollar City
222	Shambhala	SeaWorld Orlando
296	Wildfire	Knoebels Amusement Park
283	Velikolukskiy Myasokombinat-2	Kings Island
7	Altair	Kentucky Kingdom
199	Python in Bamboo Forest	Kings Island
142	Jungle Trailblazer	Canada's Wonderland
16	Bandit	Kings Island
57	Coaster Express	Six Flags Over Texas
100	Formula Rossa	Six Flags Great Adventure
18	Bat	Carowinds
218	Saw - The Ride	Cedar Point
128	Hyperion	Six Flags Over Georgia
252	Superman el Último Escape	Six Flags Magic Mountain
101	Fujiyama	Kings Dominion
27	Batwing	SeaWorld Orlando
113	Goliath	Busch Gardens Tampa
219	Schwur des Käran	Kennywood
36	Black Mamba	Busch Gardens Williamsburg
148	Kong	Six Flags Magic Mountain
15	Balder	Busch Gardens Williamsburg
24	Batman The Ride	Hersheypark
243	Steel Vengeance	Kings Island
237	Star Mountain	Hersheypark
166	Mind Eraser	Worlds of Fun
212	Riddler Revenge	Six Flags America
20	Batman The Ride	Valleyfair!
23	Batman The Ride	Darien Lake
22	Batman The Ride	Cedar Point
95	Flight of the Phoenix	Carowinds
165	Mind Eraser	Cedar Point
71	Desert Race	Dorney Park & Wildwater Kingdom
104	Fury 325	Knott's Berry Farm

279	Ultimate	Six Flags Fiesta Texas
83	Eurosat Can Can Coaster	Six Flags Magic Mountain
129	Incredible Hulk	Busch Gardens Williamsburg
153	Lightning Rod	Six Flags Great America
91	Flight Deck	Cedar Point
286	Viper	Walygator Parc
94	Flight of Fear	Holiday World
164	Millennium Force	Six Flags Discovery Kingdom
81	El Toro	Busch Gardens Tampa
130	Incredicoaster	Cedar Point
277	Twisted Colossus	Six Flags Great Adventure
34	Big Thunder Mountain Railroad	Universal Studios Hollywood
72	Desperado	Six Flags Over Texas
68	Demon	SeaWorld Orlando
74	Dinoconda	Holiday World
117	Great White	Santa Cruz Beach Boardwalk
288	Voyage	Mt. Olympus Water & Theme Park
161	Maverick	Six Flags Magic Mountain
298	Wodan Timbur Coaster	Hersheypark
82	El Toro	Dorney Park & Wildwater Kingdom
221	Screamer	Disneyland Resort Paris - Disneyland Park
135	Iron Rattler	Chimelong Paradise
145	Katun	Parque de Atracciones de Madrid
291	Wicked Cyclone	Oaks Amusement Park
260	Taron	Nigloland
254	Superman the Ride	Cinecittà World
103	Furius Baco	Six Flags Great America
65	Crazy Bird	Walygator Parc
258	Takabisha	Six Flags America
152	Leviathan	Six Flags Magic Mountain
115	Goliath	Europa Park
173	Montezum	Kings Island
208	Ravine Flyer II	Liseberg
188	Outlaw Run	Movie Park Germany
53	Boulder Dash	Kings Island
156	Mako	Six Flags New England
183	New Texas Giant	Six Flags Great America
73	Diamondback	Six Flags Great Adventure
178	Mystic Timbers	Six Flags Magic Mountain
163	Medusa Steel Coaster	Six Flags St. Louis
185	Nitro	Six Flags Over Texas
262	Taunusblitz	Six Flags Mexico
69	Demon	Parque Warner Madrid
234	Spatiale Experience	Six Flags America
223	Shock Wave	New York, New York Hotel & Casino
194	Phoenix	Heide-Park Resort

287	Vortex	Blackpool Pleasure Beach
92	Flight Deck	Disneyland Resort Paris
266	Thunder Dolphin	Disneyland
21	Batman The Ride	Six Flags Great Adventure
78	Dragon Mountain	Phantasialand
67	Cyclone	Europa Park
251	Superman / la Atracción de Acero	Six Flags Over Georgia
263	Temple of the Night Hawk	Cedar Point
17	Banshee	Kemah Boardwalk
245	Storm Chaser	Parque de Diversiones
29	Behemoth	Six Flags Mexico
138	Joker	Walibi Rhone-Alpes
131	Intimidator	Fantasilandia
241	Steel Eel	Parque de la Costa
274	Top Thrill Dragster	Elitch Gardens
112	Goliath	Six Flags Fiesta Texas
299	X2	Freizeit-Land Geiselwind
88	Firewhip	Worlds of Fun
206	Raptor	Six Flags St. Louis
132	Intimidator 305	Six Flags Discovery Kingdom
270	Time Traveler	Six Flags St. Louis
8	American Eagle	Lagoon
11	Apocalypse the Ride	Parque Warner Madrid
203	RailBlazer	Puyallup Fair
35	Bizarro	Nanchang Wanda Theme Park
159	Manta	Phantasialand
174	Montu	Thorpe Park
267	Timber Drop	Valleyfair!
158	Mammut	Happy Valley
186	Oblivion	Loca Joy Holiday Theme Park
250	Superman - Ultimate Flight	Lakeside Amusement Park
6	Alpina Blitz	California's Great America
89	Flash	Six Flags Great America
32	Big One	Parque de la Costa
216	Rock 'n' Roller Coaster	Heide-Park Soltau
193	Phantom's Revenge	Buffalo Bill's Resort & Casino
196	Poltergeist	China Dinosaurs Park
31	Big Loop	Fuji-Q Highland
98	Flying Aces	Salitre Magico
12	Apollo's Chariot	PortAventura Park
99	Force One	Marineland Theme Park
261	Tatsu	Dragon Park
119	Griffon	Fuji-Q Highland
246	Storm Runner	Freizeitpark Plohn
28	Beast	Europa Park
229	Skyrush	Holiday Park
197	Prowler	Happy Valley

249	Superman - Ride Of Steel	Kings Island
52	Boss	Beto Carrero World
210	Renegade	Lewa Adventure
214	Ride of Steel	Six Flags New England
205	Raptor	California's Great America
293	Wild One	Canada's Wonderland
4	Afterburn	Kings Dominion
106	GateKeeper	Kings Island
198	Pyrenees	Harborland
180	Nemesis Inferno	Hansa Park
136	iSpeed	Morey's Piers
236	Stampida	Ferrari World Abu Dhabi
259	Talon	Schwaben Park
300	Xcelerator	Ferrari World Abu Dhabi
70	Desafio	Fuji-Q Highland
25	Batman the Ride	PortAventura Park
253	Superman Krypton Coaster	Greenland
102	Full Throttle	Cedar Point
5	Alpengeist	Knott's Berry Farm
202	Raging Bull	Belmont Park
213	Riddler's Revenge	Six Flags Fiesta Texas
155	Magnum XL-200	Six Flags New England
62	Comet	Parc Asterix
204	Ranier Rush	SeaWorld San Antonio
84	Expedition GeForce	Warner Bros. Movie World
232	Soaring Dragon & Dancing Phoenix	California's Great America
233	Soaring with Dragon	Elitch Gardens
37	blue fire Megacoaster	Wild- und Freizeitpark Klotten/Cochem
282	Velikolukskiy Myasokombinat	Liseberg
268	Timber Wolf	Dorney Park & Wildwater Kingdom
207	Raven	Land of Legends Theme Park
85	Extreme Rusher	Energylandia
162	Medusa	Universal Studios Islands of Adventure
150	Kumba	Disney California Adventure Park
281	Valravn	Kings Island
108	GhostRider	Cedar Point
147	Kingda Ka	Mirabilandia
220	Scream!	Six Flags Great America
30	Big Apple Coaster	Six Flags Discovery Kingdom
41	Bocaraca	SeaWorld San Antonio
134	Iron Dragon	Six Flags Over Texas
269	Timberhawk: Ride of Prey	Fantawild Dreamland
168	Monster	Kijima Kogen
211	Revenge of the Mummy the Ride	Hopi Hari
146	Kawazemi	Mirabilandia

271	Titan	Tobu Zoo Park
149	Kraken	Six Flags Discovery Kingdom
171	Montana Rusa	Heide-Park Soltau
200	Quimera	Worlds of Fun
276	Tower of Terror II	Erlebnispark Tripsdrill
272	Titan Cascabel	SeaWorld San Diego
170	Montana Rusa	Six Flags Mexico
144	Katapul	Six Flags America
151	Legend	Elitch Gardens
110	Giant Dipper	Fun Spot America
19	Batman - The Dark Knight	Walyygator Parc
3	Adrenaline Peak	La Feria Chapultpec
190	Pandemonium	Salitre Magico
109	Giant Dipper	VulQano Park
189	Pandemonium	Knott's Berry Farm
230	Smiler	Hopi Hari
121	Hades 360	Movie Park Germany
125	Helix	Six Flags Over Texas
124	HeiBe Fahrt	Six Flags St. Louis
278	Twister II	Alton Towers
111	Goliath	Thorpe Park
40	Boardwalk Bullet	Hansa Park
294	Wild Thing	Six Flags Magic Mountain
86	Fahrenheit	Six Flags Magic Mountain
242	Steel Force	Alton Towers
157	Mamba	Six Flags Fiesta Texas
160	Manta	Six Flags Over Texas
76	Doble Loop	Worlds of Fun
167	Mine Blower	Gyeongju World
118	Green Lantern Coaster	Djurs Sommerland
289	Whirl Wind Looping Coaster	Six Flags Fiesta Texas
290	Whizzer	Parque Espana-Shima Spain Village
201	Racer	Nanchang Wanda Theme Park
116	Goudurix	La Feria Chapultpec
265	Texas Tornado	Kings Island
182	New Revolution	California's Great America
225	Silver Bullet	Puyallup Fair
2	Abismo	Fantasilandia
175	MP-Xpress	Ferrari Land
191	Patriot	Six Flags New England
154	Limit	Six Flags Magic Mountain
238	Star Wars Hyperspace Mountain: Rebel Mission	Six Flags Fiesta Texas
80	Eejanaika	Disneyland Paris - Walt Disney Studios Park
58	Coaster Thrill Ride	Cedar Point
60	Colorado Adventure	Thorpe Park

184	Ninja	Hansa Park
256	Swarm	Six Flags Magic Mountain
96	Fluch von Novgorod	Scandia Amusement Park
217	Rougarou	PortAventura Park
141	Judge Roy Scream	Six Flags Over Texas
97	Fly the Great Nor'Easter	Elitch Gardens
39	Blue Streak	Knott's Berry Farm
55	Cannibal	Europa Park
137	Joker	Holiday Park
126	Hydra the Revenge	Skyline Park
107	Gemini	Alton Towers
122	Half Pipe	Happy Valley
45	Boomerang	Nanchang Wanda Theme Park
44	Boomerang	Hefei Wanda Theme Park
285	Viper	Nigloland
195	Piraten	Fort Fun Abenteuerland
38	Blue Hawk	PortAventura Park
93	Flight of Fear	Beto Carrero World
42	Boomerang	Thorpe Park
77	Dragon Khan	Nagashima Spa Land
79	Dragon's Run	SeaWorld San Antonio
179	Nemesis	Valleyfair!
13	Atlantica SuperSplash	Parque Warner Madrid
209	Red Force	Zoo Safari- und Hollywoodpark Stukenbrock
140	Journey to Atlantis	Six Flags Great America
50	Boomerang	Parque Warner Madrid
49	Boomerang	Six Flags Mexico
90	Flashback	Six Flags Magic Mountain
46	Boomerang	Thorpe Park
47	Boomerang	Everland
51	Boomerang Coast to Coaster	Fuji-Q Highland
292	Wicked Twister	Phantasialand
224	Sidewinder	Taunus Wunderland
255	Superman: Escape from Krypton	Phantasialand
43	Boomerang	Wonderland Amusement Park
26	Batman: Arkham Asylum	Tokyo Dome City
176	Mr. Freeze Reverse Blast	Fraisertuis City
177	Mr. Freeze Reverse Blast	Worlds of Fun
169	Montana Rusa	Wild Waves Theme Park
248	Super Tornado	Silver Dollar City
247	Stunt Fall	Selva Magica
120	Grizzly	Parc Asterix
61	Colossus	Bosque Magico
1	10 Inversion Roller Coaster	Dreamworld
240	Steel Dragon 2000	Elitch Gardens
14	Backlot Stunt Coaster	Lightwater Valley

133	Invertigo	Six Flags Discovery Kingdom
114	Goliath	Wonder Island
280	V2: Vertical Velocity	Wonder Island
284	Vertical Velocity	Six Flags Great America
244	Steel Venom	Six Flags Magic Mountain
172	Montezooma's Revenge	Six Flags Great America
275	Tornado	Kings Island
295	Wild Thing	Wonder Island
192	Phaethon	Six Flags Great America
228	Sky Wheel	Cedar Point
227	Sky Scream	Six Flags America
231	Snow Mountain Flying Dragon	Valleyfair!
48	Boomerang	Wild Waves Theme Park
235	SpeedSnake FREE	Kolmarden
297	Winjas	Phantasialand
239	Stealth	Europa Park

Number	PCA	Bayes	BP	Boosting
257	4.336582	3.833196	6.706793	4.68571
9	4.133928	3.810172	6.004326	4.68571
66	4.039308	3.764449	5.979574	4.68571
10	4.381835	4.209356	5.926418	4.68571
33	4.091386	3.823863	5.87154	4.68571
273	4.245926	3.868304	5.777834	4.68571
143	4.169523	3.767167	5.754164	4.68571
59	4.287984	3.813666	5.726858	4.68571
87	4.306091	4.184203	5.54524	4.68571
226	4.205937	3.961933	5.489252	4.68571
215	4.000315	3.904539	5.484203	4.68571
127	4.379172	4.107312	5.445949	4.68571
64	4.177833	4.017601	5.430006	4.68571
105	4.186951	3.772831	5.397067	4.68571
181	3.850505	3.768114	5.384031	4.68571
75	4.558667	3.825345	5.370121	4.68571
222	4.372537	4.122582	5.306672	4.68571
296	4.425781	4.314859	5.294012	4.68571
283	4.158666	3.914434	5.278471	4.68571
7	4.127752	3.88998	5.277537	4.68571
199	4.454669	3.941942	5.260209	4.68571
142	4.392416	3.825346	5.259359	4.68571
16	4.251919	3.854715	5.255522	4.68571
57	4.291301	3.967848	5.223384	4.68571
100	5.597884	4.748914	5.2217	4.68571
18	4.178079	4.017084	5.211847	4.68571
218	4.189154	3.820671	5.203776	4.68571
128	4.449203	4.243175	5.109731	4.68571

252	4.469208	4.523796	5.098373	4.68571
101	4.210648	3.804209	5.094351	4.68571
27	4.250108	4.160038	5.047515	4.68571
113	4.218848	4.12903	5.030575	4.68571
219	4.335094	4.020408	5.000094	4.68571
36	4.134365	3.817443	4.992807	4.68571
148	4.058081	3.989204	4.984544	4.68571
15	4.240941	3.867368	4.980181	4.68571
24	4.123272	4.053973	4.947348	4.68571
243	4.59315	4.656011	4.934135	4.68571
237	4.133512	4.300915	4.920427	4.68571
166	4.068147	4.006076	4.910059	4.68571
212	4.068147	4.006076	4.910059	4.68571
20	4.059426	4.015089	4.904361	4.68571
23	4.07973	4.028881	4.898765	4.68571
22	4.069765	4.023916	4.894288	4.68571
95	3.9925	3.758289	4.888345	4.68571
165	4.046744	3.995588	4.886327	4.68571
71	4.364627	3.834118	4.879721	4.68571
104	4.611151	4.627544	4.866258	4.68571
279	4.339317	4.107321	4.863529	4.68571
83	3.846005	3.771526	4.828841	4.68571
129	4.316623	4.264738	4.814171	4.68571
153	4.417943	4.551914	4.807089	4.68571
91	4.007598	3.965877	4.798319	4.68571
286	4.236897	4.138059	4.79567	4.68571
94	4.243595	4.075029	4.791162	4.68571
164	4.466224	4.447265	4.782827	4.68571
81	4.454397	4.552874	4.781252	4.68571
130	4.482423	4.526849	4.777476	4.68571
277	4.460649	4.550788	4.76226	4.68571
34	3.668686	3.837163	4.753401	4.68571
72	4.525525	4.525851	4.747403	4.68571
68	3.825933	3.880494	4.74408	4.68571
74	4.181815	3.799945	4.722	4.68571
117	4.080715	4.028194	4.717294	4.68571
288	4.626569	4.706234	4.715627	4.68571
161	4.540567	4.522527	4.708297	4.68571
298	4.276804	3.944711	4.702665	4.68571
82	4.188021	3.814818	4.699727	4.68571
221	4.12376	3.964956	4.695666	4.68571
135	4.358587	4.302749	4.693349	4.68571
145	4.156778	3.897194	4.688874	4.68571
291	4.383662	4.384796	4.684072	4.68571
260	4.577391	4.317432	4.682043	4.68571
254	4.40933	4.375354	4.679799	4.68571
103	4.7527	4.047334	4.039057	4.68571

65	3.907387	3.75296	3.91773	4.68571
258	4.16154	3.769687	2.192034	4.68571
152	4.408085	4.428964	4.62403	4.66316
115	4.422941	4.51957	4.676169	4.63218
173	4.449206	4.605487	4.656585	4.63218
208	4.438899	4.45546	4.649534	4.63218
188	4.514541	4.597047	4.633003	4.63218
53	4.443708	4.531972	4.623202	4.63218
156	4.458773	4.366024	4.607684	4.63218
183	4.484233	4.467905	4.543042	4.63218
73	4.445373	4.456421	4.507688	4.63218
178	4.442774	4.569278	4.502352	4.63218
163	4.408936	4.484899	4.485684	4.63218
185	4.412668	4.380334	4.463805	4.63218
262	3.854419	3.760064	4.671989	4.56299
69	3.822171	3.880266	4.6671	4.56299
234	4.026459	3.778241	4.658624	4.56299
223	4.058895	3.994685	4.628292	4.56299
194	4.091793	3.945863	4.518402	4.56299
287	4.000557	4.056213	4.409307	4.56299
92	4.054969	3.996807	4.289093	4.56299
266	4.070287	3.761889	4.271828	4.56299
21	4.089171	4.023119	4.266922	4.56299
78	4.068502	4.169872	4.251935	4.56299
67	3.652159	3.803237	4.24676	4.56299
251	4.089229	3.866857	4.201314	4.56299
263	3.905063	3.777266	3.620054	4.56299
17	4.399516	4.393707	4.418952	4.54397
245	4.353278	4.31316	4.59452	4.54321
29	4.389928	4.407079	4.507119	4.51163
138	4.391065	4.393842	4.505039	4.51163
131	4.377083	4.411919	4.453373	4.46222
241	4.29859	4.170849	4.523715	4.43777
274	4.207582	4.165751	4.430766	4.43777
112	4.314064	4.287846	4.47452	4.42857
299	4.404807	4.2095	4.4357	4.42231
88	4.336985	4.390703	4.389488	4.39308
206	4.336985	4.390703	4.389488	4.39308
132	4.359395	4.36451	4.381937	4.39308
270	4.264188	4.181815	4.672491	4.38889
8	4.255129	4.163471	4.595619	4.38889
11	4.260442	4.244086	4.589923	4.38889
203	4.255322	4.145286	4.535883	4.38889
35	4.217593	4.215711	4.397806	4.38889
159	4.275273	4.190469	4.370588	4.38889
174	4.138375	4.158807	4.39043	4.3881
267	4.136911	3.805116	4.349815	4.3881

158	4.28284	3.872827	4.661606	4.37838
186	4.191999	3.779611	4.644205	4.37838
250	4.205991	4.116277	4.642145	4.37838
6	4.21349	3.818582	4.638069	4.37838
89	4.282208	3.790153	4.564571	4.37838
32	4.216468	3.931703	4.470904	4.37838
216	4.316529	3.878475	4.463205	4.37838
193	4.297635	4.114411	4.369065	4.37838
196	4.303155	4.138715	4.367114	4.37838
31	3.782222	3.767857	3.969863	4.37838
98	4.647806	4.677904	4.365891	4.3591
12	4.413421	4.346499	4.348601	4.3591
99	4.152823	3.783074	1.990466	4.3591
261	4.334036	4.202162	4.349329	4.34434
119	4.156123	4.130149	4.288196	4.31104
246	4.353862	4.202053	4.327578	4.30056
28	4.435673	4.518218	4.283753	4.28594
229	4.384413	4.329307	4.2732	4.24891
197	4.305581	4.321721	4.198196	4.2069
249	4.418744	4.386033	4.191923	4.19718
52	4.475416	4.548145	4.19102	4.19718
210	4.33063	4.335187	4.155811	4.12821
214	4.397185	4.362836	4.12181	4.12397
205	4.151602	4.131623	4.120059	4.12076
293	4.221353	4.103286	4.107682	4.12016
4	4.182833	4.119704	4.104624	4.12016
106	4.468062	4.259498	4.127708	4.11187
198	4.10485	3.767704	4.346344	4.10753
180	4.092509	3.805997	4.329876	4.10753
136	4.198455	3.864849	4.173473	4.10753
236	4.153096	3.801325	4.146195	4.10753
259	4.154169	4.103004	4.12793	4.10753
300	4.319921	4.170766	4.094117	4.10526
70	4.240673	4.314014	4.317233	4.08911
25	4.175812	4.252745	4.269632	4.08911
253	4.245534	4.233542	4.138941	4.08911
102	4.308088	4.18487	4.069078	4.08824
5	4.092267	4.147016	4.080677	4.0875
202	4.345596	4.300753	4.049583	4.07302
213	4.259642	4.21068	4.146294	4.06499
155	4.259099	4.193219	4.059102	4.06499
62	3.874018	3.767567	4.035552	4.04211
204	4.257423	4.066592	4.348432	4.04032
84	4.333967	3.91021	4.29762	4.04032
232	4.358608	3.825774	4.172103	4.04032
233	4.236322	3.773803	4.143707	4.04032
37	4.254387	3.905401	4.078157	4.04032

282	4.320092	3.978611	4.06196	4.04032
268	4.251634	4.178115	4.04526	4.04032
207	4.209678	4.045957	4.039306	4.04032
85	4.314493	3.767545	3.783168	4.04032
162	4.184367	4.202772	4.037308	4.03614
150	4.106637	4.135475	4.036483	4.03478
281	4.334499	4.333579	4.006873	4.00382
108	4.338755	4.391213	3.706917	4.00382
147	4.25557	4.203876	3.971591	3.96875
220	4.219256	4.240625	3.903299	3.96875
30	4.214282	4.207023	3.870436	3.96875
41	4.141533	4.210094	1.887814	3.96875
134	4.059925	3.985368	4.012044	3.96835
269	4.290664	4.181829	4.002208	3.96835
168	4.253272	3.94861	3.991078	3.96835
211	4.155438	3.947962	3.984413	3.96835
146	4.15526	3.755948	2.930619	3.96835
271	4.343095	4.316721	4.009326	3.94444
149	4.286877	4.279828	3.968349	3.94253
171	4.196848	4.354411	3.426952	3.94253
200	4.284454	4.399524	3.376849	3.94253
276	4.200695	4.279018	2.956356	3.94253
272	4.281005	4.281718	2.694543	3.94253
170	4.249766	4.341786	2.594869	3.94253
144	4.101511	4.270842	2.180477	3.94253
151	4.418208	4.455458	3.975331	3.93923
110	3.504598	3.783217	4.096173	3.93056
19	4.119418	4.060101	3.975496	3.93056
3	4.230278	4.047053	3.957147	3.93056
190	4.050503	3.886493	3.95227	3.93056
109	3.494461	3.782921	3.927384	3.93056
189	4.050866	3.883309	3.924219	3.93056
230	4.280511	4.111281	3.920044	3.93056
121	4.429809	4.565118	4.00343	3.92632
125	4.396566	4.211147	3.985727	3.92632
124	4.024053	3.76978	3.627154	3.92632
278	4.368634	4.406976	3.88738	3.89189
111	4.299955	4.242267	3.869567	3.89189
40	4.311232	4.324565	3.809187	3.89189
294	4.322817	4.288447	3.766436	3.89189
86	4.262683	4.198385	3.856955	3.87296
242	4.372577	4.338466	3.781351	3.82578
157	4.372214	4.344458	3.778304	3.82578
160	4.393478	4.27947	3.704854	3.82578
76	4.505286	4.557111	3.598443	3.82578
167	4.363467	4.418062	3.373922	3.82578
118	4.417314	4.547045	2.430501	3.82578

289	4.422346	4.51884	1.899321	3.82578
290	3.942429	3.888875	3.99899	3.79237
201	4.023904	3.906168	3.994765	3.79237
116	4.014276	3.808206	3.92311	3.79237
265	4.094642	3.941974	3.897264	3.79237
182	3.979992	3.948416	3.876108	3.79237
225	4.136672	4.125002	3.843	3.79237
2	4.08849	3.789315	3.809076	3.79237
175	4.03539	3.80089	3.79424	3.79237
191	4.218268	4.148137	3.78205	3.79237
154	4.013987	3.797644	3.781864	3.79237
238	4.001963	3.794601	3.77484	3.79237
80	4.179584	3.800529	3.75956	3.79237
58	3.609404	3.790094	3.759222	3.79237
60	4.009757	3.795783	3.752768	3.79237
184	4.269517	4.048652	3.701241	3.79237
256	4.30258	3.931803	3.700988	3.79237
96	4.207978	3.817902	3.695781	3.79237
217	4.162868	4.135388	3.674046	3.79237
141	4.028871	3.879188	3.628822	3.79237
97	4.041662	4.002373	3.582229	3.79237
39	3.770743	3.814064	3.570578	3.79237
55	4.202952	4.192102	3.560548	3.79237
137	4.075597	4.053912	3.531198	3.79237
126	4.216859	4.192884	3.522417	3.79237
107	4.067899	3.997678	3.494723	3.79237
122	3.890571	3.832353	3.493968	3.79237
45	4.032545	4.263779	3.468281	3.79237
44	4.011142	4.257449	3.46055	3.79237
285	4.07249	4.123021	3.445235	3.79237
195	4.229793	3.812506	3.409396	3.79237
38	4.053206	4.0009	3.402093	3.79237
93	4.157233	4.038304	3.345004	3.79237
42	3.84997	4.100688	3.31722	3.79237
77	4.168782	3.901192	3.30327	3.79237
79	4.278798	3.792737	3.269814	3.79237
179	4.143943	3.796874	3.19965	3.79237
13	3.90871	3.769135	3.132398	3.79237
209	4.210612	3.933739	3.130881	3.79237
140	3.964957	3.881616	3.124962	3.79237
50	4.041942	3.934939	3.076434	3.79237
49	3.902824	3.875785	3.050967	3.79237
90	3.902824	3.875785	3.050967	3.79237
46	3.892123	3.87252	3.048702	3.79237
47	3.892123	3.87252	3.048702	3.79237
51	3.881421	3.869372	3.046406	3.79237
292	3.874474	3.896008	2.986999	3.79237

224	3.992043	3.843843	2.969019	3.79237
255	3.815023	3.972848	2.95137	3.79237
43	3.698846	3.761168	2.857199	3.79237
26	4.123447	3.819182	2.850922	3.79237
176	3.834442	3.916057	2.819439	3.79237
177	3.834442	3.916057	2.819439	3.79237
169	4.033162	4.181082	2.755625	3.79237
248	3.733576	3.758169	2.730112	3.79237
247	3.913795	3.785363	2.70279	3.79237
120	4.126289	3.98643	2.669744	3.79237
61	4.030293	3.82282	2.481203	3.79237
1	3.997536	3.759199	2.392682	3.79237
240	4.373404	3.949958	2.322234	3.79237
14	4.227624	3.987713	2.290923	3.79237
133	3.912696	3.883496	2.268842	3.79237
114	4.095043	4.017592	2.246689	3.79237
280	4.032131	3.889323	2.174826	3.79237
284	4.019662	3.920615	2.163221	3.79237
244	4.023379	3.922027	2.162694	3.79237
172	3.743166	3.822632	2.028582	3.79237
275	4.098727	4.174378	1.856719	3.79237
295	4.01627	3.897121	1.58447	3.79237
192	4.048518	3.763495	3.785112	0.0253
228	3.974729	3.771337	3.39218	0.0253
227	4.092278	3.782821	3.335717	0.0253
231	4.003201	3.755823	3.139062	0.0253
48	3.827262	3.765946	2.840034	0.0253
235	3.744278	3.758411	2.735072	0.0253
297	4.060993	3.76999	2.517619	0.0253
239	3.989438	3.7875	2.37227	0.0253

11.4 Original Data of Rating and Ranking from Websites

Number	Coaster Critic Score	Number	costerbuzz	Number	Most Luxurious List	The Top Tens®
5	10	243	4.93785	164	1	1
28	10	104	4.85632	274	2	
81	10	81	4.83099	12	3	
150	10	153	4.80723	147	4	7
174	10	164	4.77173	292	6	
183	10	277	4.76087	81		2
208	10	53	4.73288	132		3
53	9.5	291	4.7037	104		4
121	9.5	288	4.69832	161		5
155	9.5	161	4.6963	229		8
4	9	135	4.68571	152		9
112	9	254	4.68571	35		10
131	9	115	4.66917			

132	9	152	4.66316
149	9	208	4.65306
159	9	188	4.63218
164	9	156	4.61667
185	9	194	4.56299
194	9	17	4.54397
261	9	245	4.54321
253	8.5	178	4.51948
119	8.5	29	4.51163
193	8.5	73	4.50813
246	8.5	183	4.49451
299	8.5	185	4.4695
147	8	131	4.46222
202	8	274	4.43777
27	7.5	112	4.42857
93	7.5	299	4.42231
177	7.5	132	4.39308
242	7.5	159	4.38889
19	7	174	4.3881
69	7	193	4.37838
111	7	12	4.3591
249	7	261	4.34434
250	7	119	4.31104
271	7	246	4.30056
290	7	28	4.28594
9	6.5	229	4.24891
120	6	197	4.2069
140	6	249	4.19718
165	5	210	4.12821
223	3.5	214	4.12397
184	2	205	4.12076
		4	4.12016
		106	4.11187
		259	4.10753
		300	4.10526
		253	4.08911
		102	4.08824
		5	4.0875
		202	4.07302
		155	4.06499
		62	4.04211
		207	4.04032
		162	4.03614
		150	4.03478
		281	4.00382
		147	3.96875
		211	3.96835

271	3.94444
149	3.94253
151	3.93923
109	3.93056
121	3.92632
111	3.89189
86	3.87296
242	3.82578
238	3.79237