Application of Machine Learning in Predicting the Transportation Rate on a Spaceship Titanic

San Jose State University

Le Dao (le.h.dao@sjsu.edu)

Dang Minh Nhu Nguyen (dang.m.nguyen@sjsu.edu)

MATH 151: Fundamentals of Data Science

Dr. Cristina Tortora

May 09, 2022

Table of Contents

Abstract
One of the fundamental contrasts between humans and computers is that humans learn from their past experiences, but computers and machines must follow a predetermined procedure. That means that if we want the machine to accomplish something, we must give it comprehensive, step-by-step instructions. As a result, humans created scripts and trained computers to learn on their own. Machine Learning was therefore conceived. Our project centered around a competition that was hosted by Kaggle (i.e. the world's largest community for data scientists and machine learning experts, acquired by Google) [13]. The goal of the challenge is to determine whether a passenger was transported to an alternate dimension during the collision of the Spaceship Titanic with the spacetime anomaly. Along with completing the challenge, our team hopes to build the most accurate model based on machine learning algorithms, which will assist humanity in taking another step forward in their exploration of the enormous cosmos. In this report, we will address some of the challenges we encountered during the data cleaning process, along with the process of building the models. At the end of the report, we will also state the resulting output of each model to give a comparison between each algorithm in addition to interpreting the results, the detailed workflow diagram is in *Figure 1*.

I. Introduction:

Earth - a planet in the solar system, where life has awakened, is the residence of humans and other species in the vast universe. Three-quarters of the planet's surface is covered with water. Constantly changing geological activity helps organisms adapt to survive and promote growth and evolution. Many creatures may perish as a result of Mother Nature's changes, both within and externally, yet many species can adapt, evolve, and interbreed. Life on Earth has a long history that is rapidly accelerating and expanding. With boundless tolerance and patience, Earth has generated and fostered life for millions of years. Our world, on the other hand, is extremely vulnerable to cosmic objects; a huge meteorite is all it takes to wipe out the entire globe. Humans must therefore devise a means of reaching into space to exist.

Modern science is undeniably progressing and attempts to find a habitable planet are proliferating. NASA revealed on March 21, 2022, that after a 30-year search for extraterrestrial worlds, the number of exoplanets has surpassed 5000 [3], and it is unavoidable that humanity will find life somewhere. Since no one can guarantee that meteoric impacts and apocalypse events similar to those that wiped off the dinosaurs millions of years ago will not occur on Earth again; and appreciating the need of investing in the future so that one day, we humans will be able to freely move between planets or migrate to a new one, our team worked on The Spaceship Titanic dataset which was an interstellar passenger liner launched a month ago [17].

Using records recovered from the spaceship's broken computer system, the goal of this study is to obtain the optimal algorithm to forecast which passengers were transported by the anomaly and to assist rescuers in locating missing passengers [17]. Furthermore, we must appreciate the great inventions of human civilization that all began with the development of language, particularly numbers. Humans can explain not just our thoughts and sentiments to others around us, but also clearly and accurately define and describe scientific difficulties.

II. Materials and Methodology:

1. Materials:

Machine learning is the process through which computers learn from previous training data and experience over time. It focuses on developing computer programs that can access data and use it for self-learning[8]. Machine Learning can be classified into three main types: supervised learning, unsupervised learning, and reinforcement learning [14].

As mentioned above, our data is taken from Kaggle Data Competition. The ship set sail on its maiden mission with around 13000 passengers on board, taking emigrants from our solar system (e.g Mars, Earth, and Europa) to three newly habitable exoplanets orbiting nearby stars including TRAPPIST-1e, 55 Cancri e, and PSO J318.5-22. The information of the passengers was split into 2 datasets, train.csv (8693 unique values) and test.csv(4277) [17]. Our target is to use the training dataset to predict whether the passengers were successfully transported to their desired destination. Additionally, understanding the significance of using machine learning to assess and solve the situation at hand. We will use supervised and unsupervised learning to handle the challenge in this project. To anticipate future events, supervised learning (e.g Logistics

Regression, Linear Discriminant Analysis, Quadratic Discriminant Analysis, Random Forest, K Nearest Neighbor, and so on) can apply what has been learned in the past to initial data using labeled examples. The learning method provides an inferred function to make predictions about the output values after evaluating a known training data set. Unsupervised machine learning methods (e.g. K Mean Clustering, Hierarchical clustering), on the other hand, are utilized when the training data is neither classed nor labeled. Unsupervised learning investigates how computers might infer a function from unlabeled data to describe a hidden structure [19].

With the help of computer language, especially R language along with their built-in packages (e.g. caret, mlbench, MASS, class, and so on), we were able to perform our machine learning prediction. Importantly, we used imputation (e.g. MICE - Multivariate Imputation by Chained Equations) - a powerful approach for handling missing data - throughout the data cleaning process, which will be explained in more detail in the next section. This package is used to estimate more realistic regression coefficients that are not impacted by missing values by replacing them with plausible values [6]. The use of numerous plausible values quantifies the uncertainty in guessing what the missing values might be, preventing misleading precision. Thus, it yields precise estimates of quantities or relationships of interest [12].

2. Methodology:

a, Data Cleaning:

The first step to perform our project is loading the training data as well as all necessary packages into RStudio. When loading the train.csv, we need to check whether the data columns' names begin with a character instead of a number, since R does not allow the column names to begin with a number. For any blank in the dataset, it will automatically be filled with NA values. If we did not specify for R to understand, then R will read it as “” for non-numeric columns instead of recognizing them as NA values [11].

The next step is checking if the data has any missing values and investigating them if any. Usually, it is normal to observe the data that contains any types of missing values. There are numerous approaches to dealing with missing data (e.g. Listwise-deletion, Mean/Median substitution, and Multiple Imputation) so that the results are as accurate as possible. Yet, first and foremost, we must determine how significant the missing data is, whose variables it affects, and which data lines are missing[6]. There is approximately 26.73% of our data contain missing values (i.e. NA) in which a lot of well worthy observations were missed (*Figure 2)*. Thus further investigation needs to be performed. In this case, if we choose either Listwise-deletion or Mean/Median it will create biases and affect the accuracy of our result. Rather than omit all the NA values, we decided to perform multiple imputations. This enables us to account for the uncertainty surrounding the true value and generate roughly unbiased estimates.

After determining the amount of NA and also the methods to complete the data, the next step is to perform an imputation. The first step is to check the structure of the whole dataset in order to correct the data type of each column (e.g. converting the variable HomePlanet as a character to factor with 3 distinct levels). Then, we identify the type of missing data (e.g. Missing Completely at Random - MCAR, Missing at Random - MAR, Missing Not at Random - MNAR).

From *Table 1*, not more than 1% correlation rate is observed. Thus, we can conclude that the observations are all independent and NA in this case is MAR. We then started imputation using the package MICE, command “mice” and “complete”. We create several tuples with missing values that are assigned appropriate values when we use multiple imputation. As a result, the given values for each of these datasets will differ slightly. However, in each of these allocated datasets, the non-missing metric will be the same [12]. For different types of data, “mice” will apply different built-in univariate imputation, *Table 2,* (e.g. polyreg - polytomous logistic regression, logreg - logistic regression, predictive mean matching - pmm). After obtaining the complete data from imputation we will then set seed to attain the randomness while subsetting our dataset into train and test with 70% of the observation belonging to the train set. We would need a larger amount of training set rather than the test set to improve the accuracy of our prediction which is also the last step of our data cleaning process.

After having “MICE” imputed all the missing data, the dataset is now completed and can be used to build the model. However, since there are so many columns in this dataset, determining which should be used to predict the model is not easy. Based on how the model is being approached, too many features in a model can either increase model complexity or lead to other issues like multicollinearity and overfitting. Additionally, acquiring a large set of features for future predictions may be more difficult with a very complicated model. As a result, choosing the best characteristics is pivotal [2].

RFE (i.e.Recursive Feature Elimination) is a popular approach for determining which variables are most effective in measuring the response variable of a predictive specific model. To determine the best set of attributes, this type of feature selection uses a backward selection procedure. It starts by creating a model which is based on all columns and calculating the relevance of each one. Then, by using model assessment criteria, it iteratively ranks the variables and drops the feature(s) that are/are the least crucial [2]. In this data, the random forest importance criterion is the method that is utilized to calculate feature importance. This method is repeated until only a small number of characteristics remain in the model [2].

b, Classification:

Machine Learning is incomplete without supervised learning. When the variable to be predicted is categorical, classification techniques are applied [15]. Since the response variable “transported” is binary, such a model like Logistic Regression, Linear Discriminant (LDA), Quadratic Discriminant Analysis (QDA), K-nearest Neighbor (KNN) and Random Forest may be utilized to predict the target feature.

The dataset is first being imputed with **Multiple Logistic Regression**. This method is being chosen because the target column, “transported”, only has two possible outcomes which are “True” or “False” (satisfied the condition of being dichotomous), and there is more than one independent variable being used to predict the data. The model then further uses the sigmoid function to transfer predicted values to probabilities. It converts any real value to a number between 0 and 1. The function has a non-negative derivative and exactly one inflection point at any data point [15]. The formula for this model is written below:

$$p(X) = \frac{e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}}{1 + e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}}.$$

Multiple Logistic function, where X=(X $_1$, ..., X $_p$) are p predictors) [7]

If the data's classes are not well separated, an alternative model called **Linear Discriminant Analysis** comes into place. To identify the classes, LDA assumes that all classes are linearly separable, and numerous linear discrimination functions representing several hyperplanes in the feature space are generated [1]. LDA implemented Bayes Theorem to predict the probabilities. They create predictions based on the likelihood that the input dataset will fall into one of the classes. The output class is the one with the highest probability, after which the LDA provides a prediction[16]. The below equation is the formula for LDA:

$$\delta_k(x) = x^T \Sigma^{-1} \mu_k - \frac{1}{2} \mu_k^T \Sigma^{-1} \mu_k + \log \pi_k$$

Linear Discriminant Analysis function where number of predictor is >1 [7]

Similar to LDA, in order to make predictions, **Quadratic Discriminant Analysis** (QDA) assumes that the observations from each class are selected from a Gaussian Normal distribution and puts the estimation for the parameters into Bayes' theorem. Nevertheless, in contrast to LDA, QDA assumes that each feature has its own covariance matrix and it presupposes that a data point from the kth class has the form X~ Normal(k, k), where k is the kth class's covariance matrix [7]. The equation for this model is specified here:

$$\begin{aligned} \delta_k(x) &= -\tfrac{1}{2}(x-\mu_k)^T \Sigma_k^{-1}(x-\mu_k) + \log \pi_k \\ &= -\tfrac{1}{2} x^T \Sigma_k^{-1} x + x^T \Sigma_k^{-1} \mu_k - \tfrac{1}{2} \mu_k^T \Sigma_k^{-1} \mu_k + \log \pi_k \end{aligned}$$

Quadratic Discriminant Analysis function [7]

An ensemble of decision trees is known as a **Random Forest**. This model combines the results of numerous decision trees to get better accuracy and consistency for the prediction. Random forest increases the model's randomness while generating the trees. At each split, the model looks for the best feature from a random subset of features rather than the most essential feature. As a result, there is a lot of variety, which leads to a better model. [4]

The K-Nearest Neighbor algorithm, shortly written as **K-NN**, is a supervised nonlinear classification algorithm. The K-NN algorithm is a non-parametric algorithm, which means it makes no assumptions about the underlying data or its distribution. It is one of the most basic and extensively used algorithms, and it is based on the k value (Neighbors). The formula for KNN is written here:

$$\hat{f}(x_0) = \frac{1}{K} \sum_{x_i \in N_0} y_i.$$

K-nearest neighbor regression [7]

The KNN model has the following algorithm:

1. Determine the Kth neighbor's number.
2. Using Euclidean distance to calculate the distance between any two observations and get the K Nearest Neighbor of an unknown data point. The formula is written below:
3. Since KNN is calculated based on distance, we must use numeric columns only. For any categorical column, we need to create dummy variables for that column in order to use it for the purpose of prediction. [10]
4. In each category among K-neighbors, calculate the number of data points and then assign the new data point to the category where the most neighbors were counted. [5]

After implementing a different type of method to the dataset, determining which model outperforms the others is necessary. Each model's performance is assessed based on the percentages of accuracy, which means it calculates how precisely the model is trained. In the case of binary classification, this number is measured by a confusion matrix *Table 3*.

True Negative and True Positive is the total number of times that the model's prediction of negative/positive values is equal to the actual negative/positive values in the test set. For instance, the actual observation is a negative value, and what the model predicts is accurate. False Negative/ Positive depicts the total number of times that the model's prediction classified the actual negative/positive values in the test set wrongly. For example, the actual value is positive, but the model returns a negative value. Based on these four values, the formula for the accuracy score is calculated [9]:

$$Accuracy = \frac{True\ Negative + True\ Positive}{True\ Negative + True\ Positive + False\ Postive + False\ Negative}$$

The model that has the highest accuracy score is the best model for this dataset. The results of the accuracy for each model are specified in Table *4*.

III. Results:

Table 5 and Figure 3, generated in R, illustrate the process of Recursive Feature Elimination. Based on the table, 4 variables result in 75.85% of accuracy while increasing the number of features to 8 also increases the accuracy by almost 5%, which is 79.73%. If added to the model, it will lead to the highest accuracy, but when compared with the accuracy of 8 variables, the number only increases by 0.3%, which is not very significant. Based on the graph, the value of 8 is the point where the elbow is shown. Hence, every model should be built based on 8 features in the dataset.

After knowing how many features should be used, the next step is to determine the first 8th most important variables. This can easily be done by using the command "varImp" in R. Figure 4 illustrates the features' importance in descending order.

When we build KNN models, we need to tune the parameters k since different k result in different percentages of accuracy. At k = 7 and k =14, the value of the accuracy is higher than when k = 5, however, it increases very little. Hence, by running a for loop for every k from 1 to 15 and based on the graph below, the value of k that results in the most efficient model is 5, Figure 5.

Similarly in Random Forest, there are many choices for “mtry”. To decide which mtry results in the best model, a cross-validation method with 3 folds is utilized to tune “mtry”. *Table 6* shows the mtry = 2 that results in the highest performance RandomForest model.

A summary of the result of each Machine Learning method is in *Table 4,* which describes the accuracy score of each model in the first completed data that is imputed by “Mice” in R. It can be seen that while other models' accuracy range from 76 to 79, K-Nearest Neighbor outperforms all of them and result in the best score of 85.621%. As mentioned above, three complete imputation sets are created through “mice” in R. K-Nearest Neighbor is the best model that predicts the data most accurately on the first complete imputation set. However, it might not be the best model for the remaining two because R generated the NA in each imputation randomly. Hence, to verify the fact that KNN is robust for the spaceship dataset, the model is built on the other imputation sets and checks for their accuracy. The result shown in *Table 7* proves that the model KNN is robust for all 3 data sets of imputation since the MSE(s) do not change much.

IV. Conclusion:

Machine Learning is an application of artificial intelligence (AI) that gives systems the ability to automatically learn and improve from experience without the need for explicit programming. The learning process begins with observations or data then looks for patterns in data and makes better decisions in the future based on the examples we provide. The main purpose is to allow computers to automatically learn without human intervention or assistance and adjust actions accordingly. We will not only gain a better understanding of missing values and how to deal with them as a result of this project, but we will also have the opportunity to use Machine Learning methods to predict if passengers will travel successfully to the new location. By determining the best model (i.e. KNN) in our dataset, this study gives us a promising future in which one day, we humans could freely explore and travel in the enormous universe.

V. Appendix

	HomePlanet	CryoSleep	Cabin	Destination	Age	VIP	RoomService	FoodCourt	ShoppingMall	Spa	VRDeck	Name
HomePlanet	1	-0.0148043	0.0071583	-0.0011131	-0.011528	-0.00858	0.009728958	-0.0118971	0.010971721	-0.00657	-0.01761	0.012128
CryoSleep	-0.01480425	1	0.0100203	-0.0027975	-0.018008	0.009436	-0.013004246	-0.0080546	0.013547191	0.002218	0.011694	0.024627
Cabin	0.007158337	0.01002027	1	-0.0008936	-0.000529	0.006891	-0.000772732	0.00434403	-0.013898872	-0.01173	-0.00161	-0.01323
Destination	-0.00111314	-0.0027975	-0.0008936	1	-0.00423	-0.00133	-0.004442556	0.01213783	0.003392359	-0.00465	-0.01069	-0.001
Age	-0.01152798	-0.0180081	-0.000529	-0.0042299	1	0.009762	-0.015470696	0.00130795	-0.001499976	-0.01562	0.000717	-0.00604
VIP	-0.00858462	0.00943587	0.0068906	-0.0013306	0.009762	1	-0.001209566	-0.0014509	-0.004272808	-0.01206	-0.01775	-0.01865
RoomService	0.009728958	-0.0130042	-0.0007727	-0.0044426	-0.015471	-0.00121	1	-0.0101596	-0.001744119	-0.00455	-0.0106	-0.017
FoodCourt	-0.0118971	-0.0080546	0.004344	0.0121378	0.001308	-0.00145	-0.010159598	1	-0.012472704	0.006406	0.011251	0.004221
ShoppingMall	0.010971721	0.01354719	-0.0138989	0.0033924	-0.0015	-0.00427	-0.001744119	-0.0124727	1	-0.01247	0.007771	-0.00394
Spa	-0.00656527	0.00221809	-0.0117297	-0.0046531	-0.01562	-0.01206	-0.004547513	0.00640584	-0.012472704	1	-0.00528	0.009566
VRDeck	-0.01761162	0.01169449	-0.0016059	-0.010694	0.0007175	-0.01775	-0.010603136	0.01125097	0.007770875	-0.00528	1	-0.00172
Name	0.012128093	0.02462735	-0.0132279	-0.0010037	-0.006042	-0.01865	-0.017003666	0.0042209	-0.003943577	0.009566	-0.00172	1

Table 1: Correlation of Missing values vs Missing values

Variables	Datatype	Imputation Method
HomePlanet	Factor	polyreg
CryoSleep	Logistic	logreg
Destination	Factor	polyreg
Age	Numeric	pmm
VIP	Logistic	logreg
RoomService	Numeric	pmm
FoodCourt	Numeric	pmm
ShoppingMall	Numeric	pmm
Spa	Numeric	pmm
VRDeck	Numeric	pmm

Table 2: Imputation methods on each variables

	Actual	
Predicted	Negative	Positive
Negative	True Negative	False Negative
Positive	False Positive	True Positive

Table 3. Confusion matrix

Machine Learning Methods	Accuracy Score
Multiple Logistic Regression	0.767638
Linear Discriminant Analysis	0.767638
Quadratic Discriminant Analysis	0.702454
Random Forest (mtry=2)	0.7952454
K-Nearest Neighbor (k=5)	0.8542945

Table 4. Accuracy Table on the First Completed Data

Variables	Accuracy	Kappa	Accuracy SD	Kappa SD	Selected
4	0.7585	0.5171	0.01997	0.03976	
8	0.7973	0.5944	0.01472	0.02945	
10	0.8003	0.6004	0.01754	0.03508	*

Table 5. Feature selections generate by R

mtry	Accuracy	Kappa
2	0.7959256	0.5916462
5	0.7885732	0.5769645
9	0.7820644	0.5641042

Table 6. Result of tuning mtry

Data	1	2	3
K-Nearest Neighbor	85.62117	85.3911	85.16104

Table 7. Accuracy Score on all Completed Data

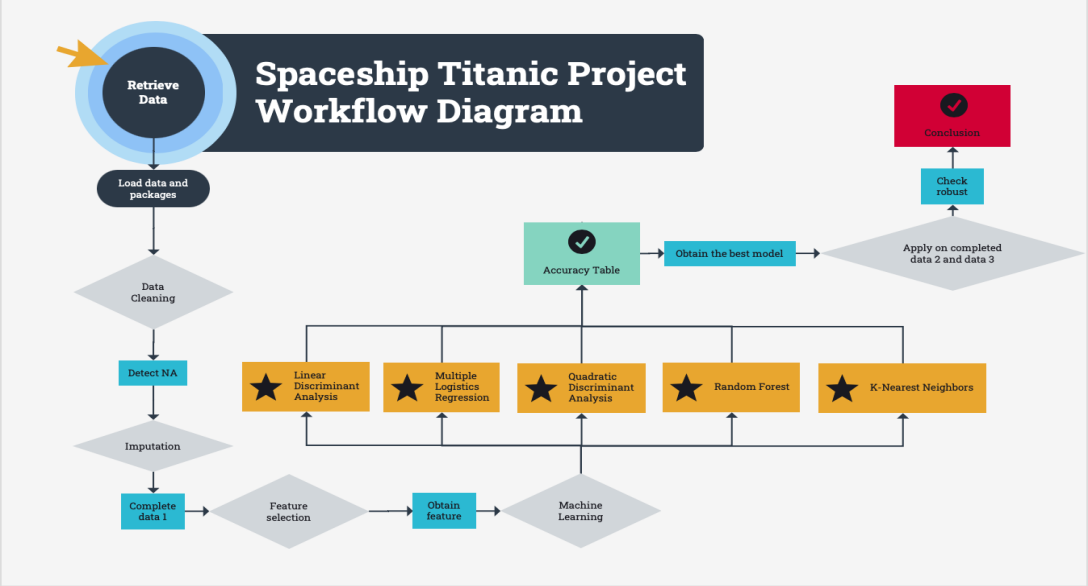

Figure 1. Spaceship Titanic Project Workflow Diagram [18]

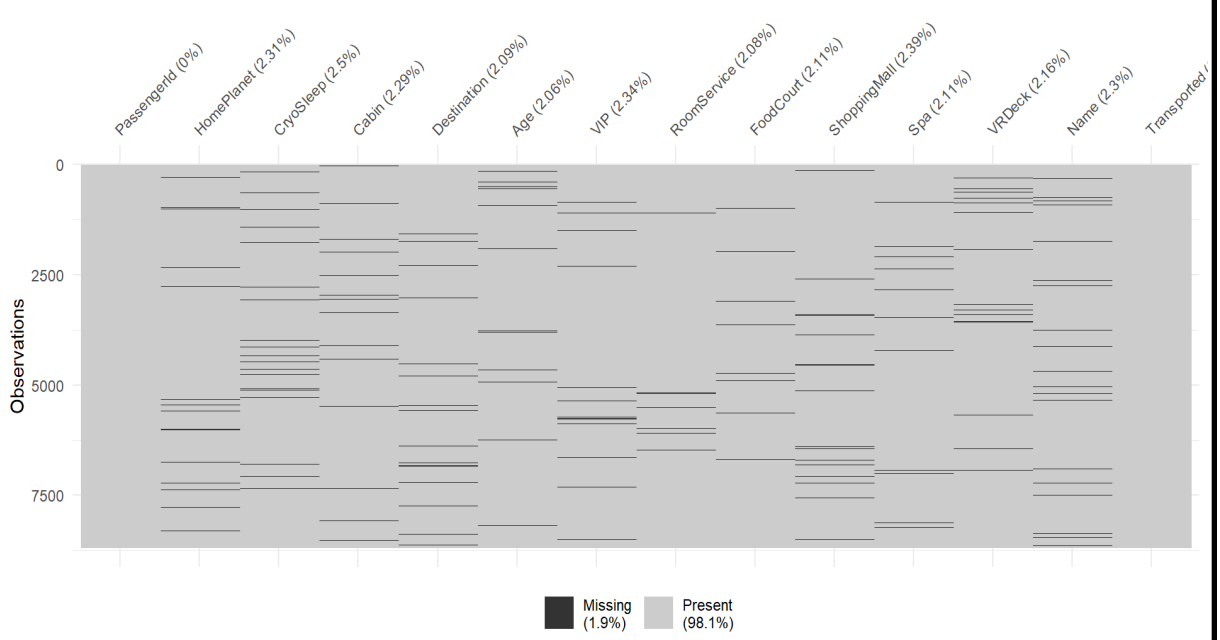

Figure 2: Percentage of Missing Values Per Column

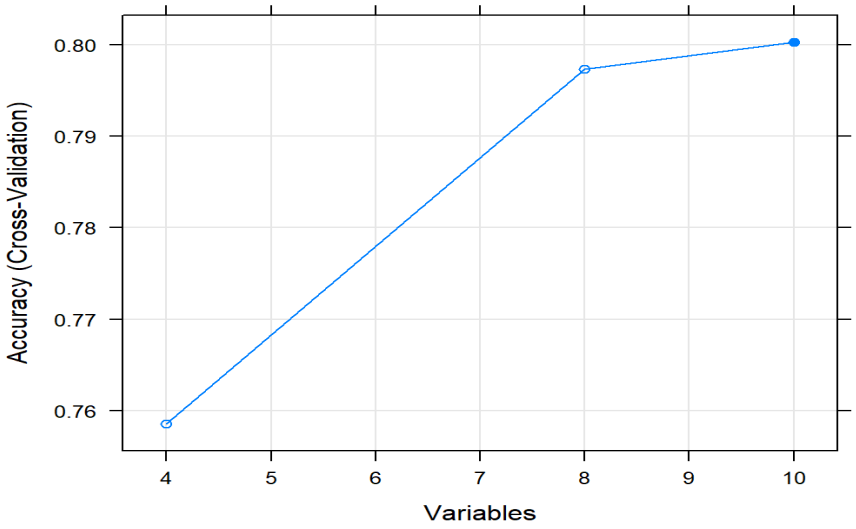

Figure 3. Variables' Accuracy (Cross-Validation)

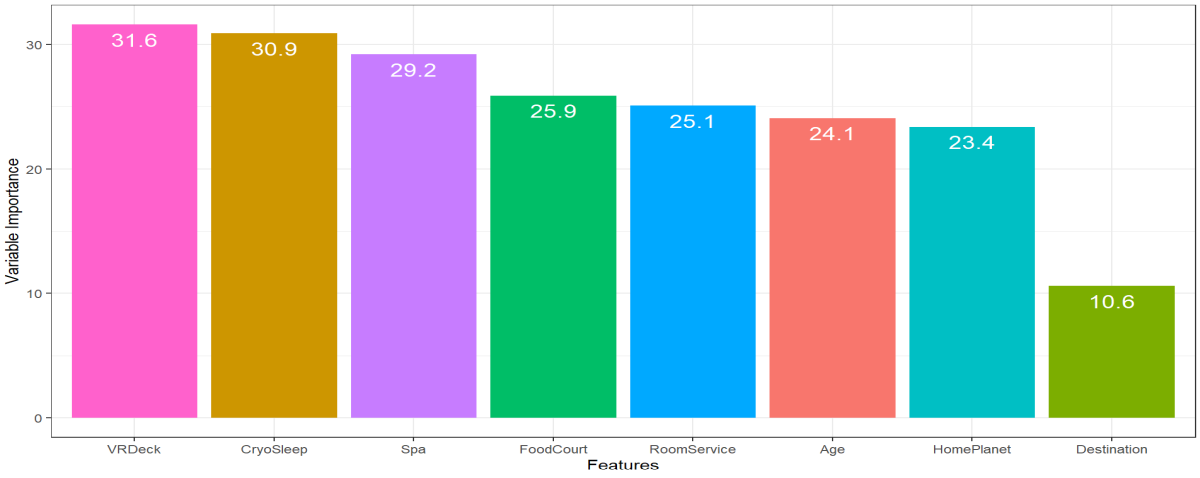

Figure 4. RFE Result Feature

	Df	Sum Sq	Mean Sq	F value	Pr(>F)	
smoker	1	4.361e+10	4.361e+10	564.654	<2e-16	***
children	3	5.793e+08	1.931e+08	2.500	0.0595	.
smoker:children	3	1.404e+08	4.681e+07	0.606	0.6115	
Residuals	320	2.472e+10	7.724e+07			

Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

ANOVA Test Result

VI. References

[1]Balas, V. E., Solanki, V. K., & Kumar, R. (2020). *An Industrial IoT Approach for Pharmaceutical Industry Growth: Volume 2* (1st ed.). Academic Press. https://doi.org/10.1016/B978-0-12-821326-1.00002-4

[2]Bulut, O. (2022, March 30). *Effective Feature Selection: Recursive Feature Elimination Using R*. Medium. https://towardsdatascience.com/effective-feature-selection-recursive-feature-elimination-using-r-148ff998e4f7

[3] *Cosmic Milestone: NASA Confirms 5,000 Exoplanets*. (2022, March 21). NASA Jet Propulsion Laboratory (JPL). https://www.jpl.nasa.gov/news/cosmic-milestone-nasa-confirms-5000-exoplanets

[4] Donges, N. (2022, April 14). *Random Forest Algorithm: A Complete Guide*. Built In. https://builtin.com/data-science/random-forest-algorithm

[5] GeeksforGeeks. (2020, June 22). *K-NN Classifier in R Programming*. https://www.geeksforgeeks.org/k-nn-classifier-in-r-programming/

[6] J. (2020, May 18). *Getting Started with Multiple Imputation in R | University of Virginia Library Research Data Services + Sciences*. University of Virginia Library. https://data.library.virginia.edu/getting-started-with-multiple-imputation-in-r/

[7] James, G., Witten, D., Hastie, T., & Tibshirani, R. (2021). *An Introduction to Statistical Learning: with Applications in R (Springer Texts in Statistics)* (2nd ed. 2021 ed.). Springer.

[8] Jordan, & Mitchell, T. M. (2015). *Machine learning: Trends, perspectives, and prospects*. *Science* (American Association for the Advancement of Science), 349(6245), 255–260. https://doi.org/10.1126/science.aaa8415

[9] Kulkarni, Chong, D., & Batarseh, F. A. (2021). *Foundations of data imbalance and solutions for a data democracy.* https://doi.org/10.1016/B978-0-12-818366-3.00005-8

[10]*k-Nearest Neighbor: An Introductory Example*. (n.d.). Quantdev.Ssri.Psu.Edu. https://quantdev.ssri.psu.edu/sites/qdev/files/kNN_tutorial.html?fbclid=IwAR2JJ9wjt8md-vqnj2Np7mnfK7LQUodfLtZm627oqLsTuNhIT4zTw9p81iI#:%7E:text=Note%3A%20We%20use%20k%2DNN,when%20predicting%20a%20continuous%20outcome

[11]*Loading data into R*. (n.d.). Rstudio-Pubs-Static.https://rstudio-pubs-static.s3.amazonaws.com/13589_1dc68b35cccc4686924ba80c37697b7e.html

[12]Li, Stuart, E. A., & Allison, D. B. (2015). *Multiple Imputation: A Flexible Tool for Handling Missing Data*. JAMA : the Journal of the American Medical Association, 314(18), 1966–1967. https://doi.org/10.1001/jama.2015.15281

[13]Moyer, E. (2017, March 8). *Google buys Kaggle and its gaggle of AI geeks*. CNET. https://www.cnet.com/science/google-buys-kaggle-and-its-gaggle-of-ai-geeks/

[14]Obermeyer, & Emanuel, E. J. (2016). *Predicting the Future — Big Data, Machine Learning, and Clinical Medicine*. The New England Journal of Medicine, 375(13), 1216–1219. https://doi.org/10.1056/NEJMp1606181

[15]Raj, A. (2021, December 16). *Perfect Recipe for Classification Using Logistic Regression*. Medium.https://towardsdatascience.com/the-perfect-recipe-for-classification-using-logistic-regression-f8648e267592#:%7E:text=Advantages%20of%20Logistic%20Regression&text=Logistic%20regression%20is%20easier%20to,as%20indicators%20of%20feature%20importance

[16]Sarkar, P. (n.d.). *What is Linear Discriminant Analysis(LDA)?* upGrad KnowledgeHut. https://www.knowledgehut.com/blog/data-science/linear-discriminant-analysis-for-machine-learning

[17]*Spaceship Titanic | Kaggle*. (2022, April 27). Kaggle. https://www.kaggle.com/competitions/spaceship-titanic/overview

[18]*Venngage | Professional Infographic Maker | 10,000+ Templates*. (n.d.). Venngage. https://venngage.com

[19]Zhou, Pan, S., Wang, J., & Vasilakos, A. V. (2017). *Machine learning on big data: Opportunities and challenges*. *Neurocomputing*, 237, 350–361. https://doi.org/10.1016/j.neucom.2017.01.026