Introduction

In the Spaceship Titanic disaster, this data-driven challenge has been initiated to predict the likelihood of passengers being transported to another dimension. This report outlines the analytical approach taken, including machine learning model selection, data preparation, and evaluation methodologies, to address the prediction task effectively. The focus is on utilizing K-Nearest Neighbors (KNN), Logistic Regression, Random Forest, and Deep Neural Network (DNN) algorithms to analyze the dataset and draw conclusions.

Data Analysis

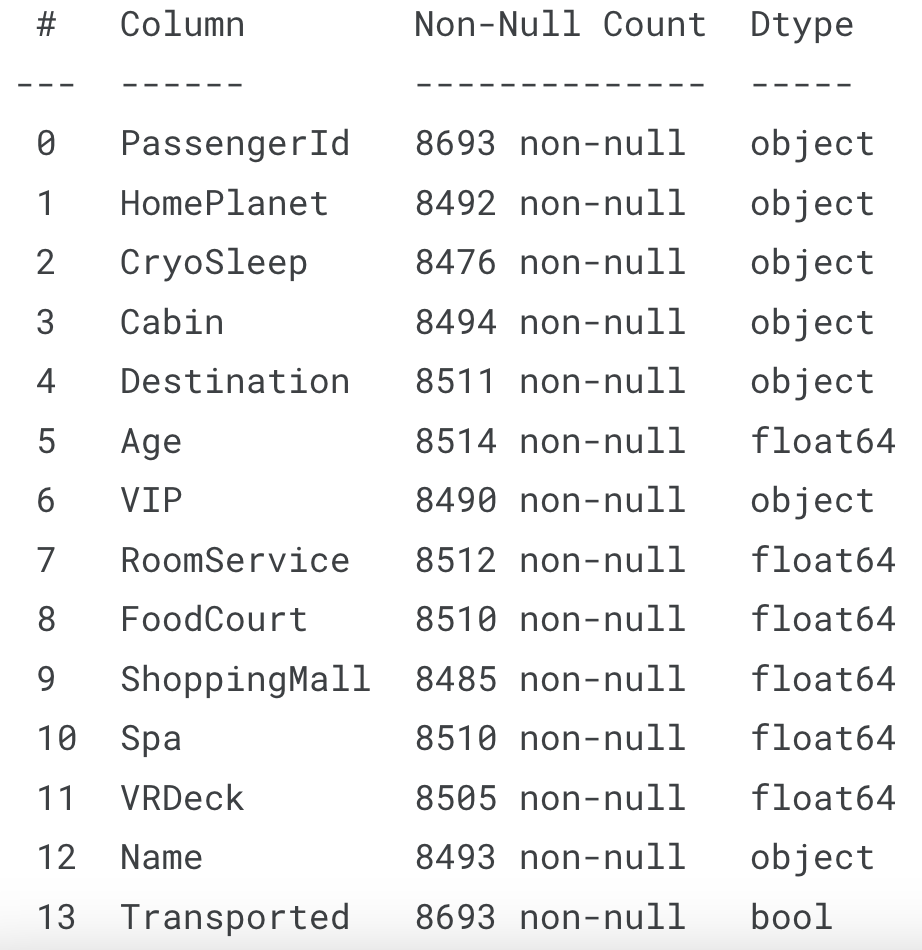
1. Define Objectives:  
In this competition, the task is to predict whether a passenger was transported to an alternate dimension during the Spaceship Titanic’s collision with the spacetime anomaly.

In order to achieve this goal, I will try several machine learning algorithms including K-Nearest Neighbors, Logistic Regression, Random Forest and deep learning. These techniques perform greate results as to classification problems.

2. Data Collection:

The data is divided into two segments: a training set and a test set. The training set is subsequently partitioned into a smaller training subset and a validation subset. The validation subset is utilized for tuning hyperparameters, while the actual test set is employed to assess the model's performance. The accuracy metric for the model is determined based on the results from this real test set.

3. Data Cleaning:



The summary of the dataset indicates that missing values are present in every column except for 'PassengerId' (index column). The data attributes can be classified into two types: categorical and numerical. To address these missing values, the most frequent category is used to fill in the gaps in categorical data. Meanwhile, for numerical data, the missing values are imputed with the feature's average value.

4. Data Preprocessing:

*Data dictionary:*

*PassengerId - A unique Id for each passenger. Each Id takes the form gggg\_pp where gggg indicates a group the passenger is travelling with and pp is their number within the group. People in a group are often family members, but not always.*

*HomePlanet - The planet the passenger departed from, typically their planet of permanent residence.*

*CryoSleep - Indicates whether the passenger elected to be put into suspended animation for the duration of the voyage. Passengers in cryosleep are confined to their cabins.*

*Cabin - The cabin number where the passenger is staying. Takes the form deck/num/side, where side can be either P for Port or S for Starboard.*

*Destination - The planet the passenger will be debarking to.*

*Age - The age of the passenger.*

*VIP - Whether the passenger has paid for special VIP service during the voyage.*

*RoomService, FoodCourt, ShoppingMall, Spa, VRDeck - Amount the passenger has billed at each of the Spaceship Titanic's many luxury amenities.*

*Name - The first and last names of the passenger.*

*Transported - Whether the passenger was transported to another dimension. This is the target, the column you are trying to predict.*

**Feature Engineering**

Upon looking into the dataset, it becomes apparent that the 'Cabin' column includes three distinct features: the deck, the cabin number, and the side of the ship (port or starboard). Thus the next step is extracting these individual elements—'deck,' 'num,' and 'side'—from the 'Cabin' column and imputing any missing values with the most frequently occurring category for each feature.

Moreover, the dataset contains three 'homeplanet' categories and two 'destination' categories. Instead of applying one-hot encoding to these categorical variables, I have assigned numerical representations to each category, as this has yielded a higher accuracy score in this context.

An additional observation is that the 'PassengerId' field is composed of a group name and the number of people in that group, suggesting a potential feature: a count representing the size of each passenger's group. This is predicated on the assumption that passengers are more likely to assist others within their group.

Finally, columns that offer minimal predictive value, such as 'PassengerId,' 'name,' 'Cabin,' and 'Deck,' will be removed from the dataset.

It is also important to note that while I considered categorising 'age' into different groups, this strategy resulted in a decreased accuracy for the models.

**Data Transformation**

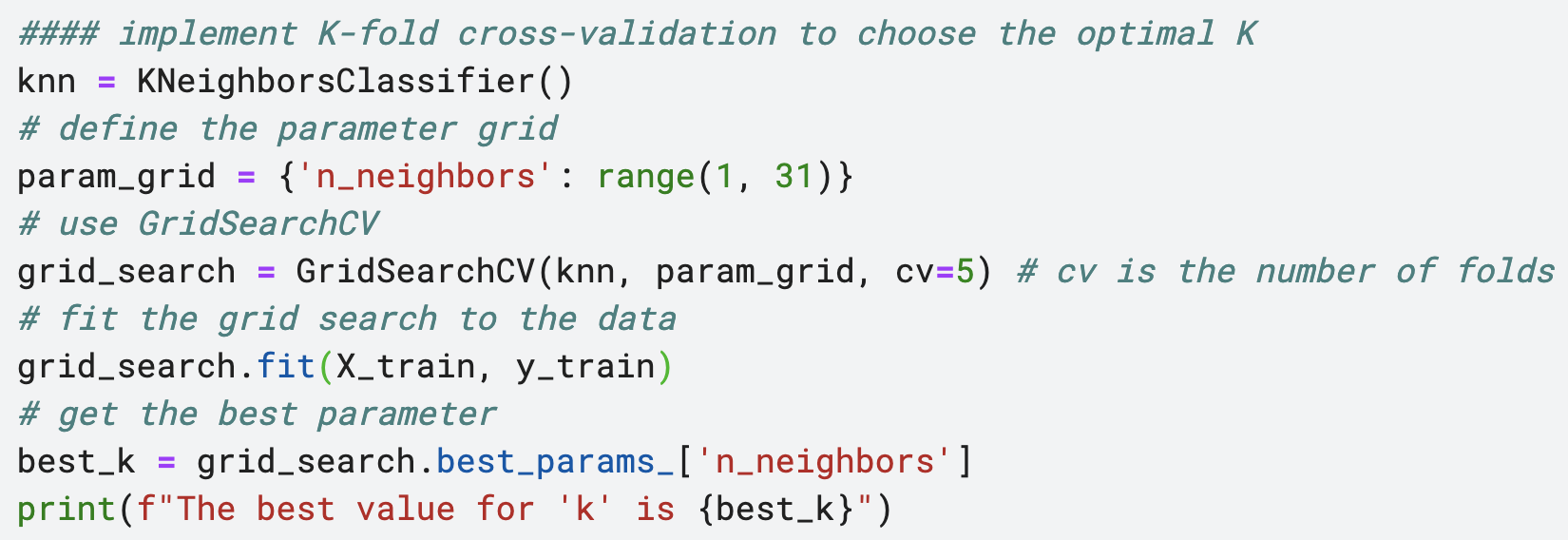
First I separated the features and the target variable. Subsequently, I applied the train\_test\_split() function from the scikit-learn library to partition the data into training and validation sets. Following this division, I applied a Standard Scaler to normalise the numerical features. This step is crucial because numerical features may have varying ranges, and models that are sensitive to the scale of the data could be adversely impacted if the features are not standardised.

5. Data Modeling:

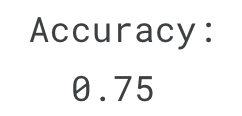
*K Nearest Neighbors (KNN):*

The KNN algorithm is utilized due to its simplicity and effectiveness in classification problems where the relationship between features is not linear. It excels in scenarios where the decision boundary is irregular. The algorithm's reliance on feature similarity allows it to make predictions based on how closely data points resemble each other, which can be particularly useful in systems where proximity is a strong indicator of equivalence.

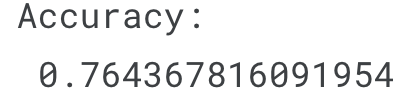
Using K-fold cross-validation to choose optimal K, the final accuracy results in 0.78

.

*Logistic Regression:*

Logistic regression is chosen for its efficiency in binary classification tasks. It provides a probabilistic framework that outputs the probability of the target variable belonging to a specific class. This model is beneficial for its interpretability, as it allows for an understanding of the impact of each feature on the likelihood of outcomes. It is particularly useful when the relationship between the independent variables and the log-odds of the dependent variable is approximately linear. In logistic regression, the coefficients can provide insights into the importance and impact of each feature on the prediction.Features that possess positive coefficients correspond to a higher probability of the predicted result occurring. On the other hand, features with negative coefficients are indicative of a reduced likelihood of the predicted result. The significance of a feature's impact is determined by the size of the coefficient's absolute value: substantial values (as seen with features such as Spa and VRDeck) signify a more pronounced effect, whereas minimal values (such as those associated with Age and Num) suggest a more modest effect. 

*Random Forest:*

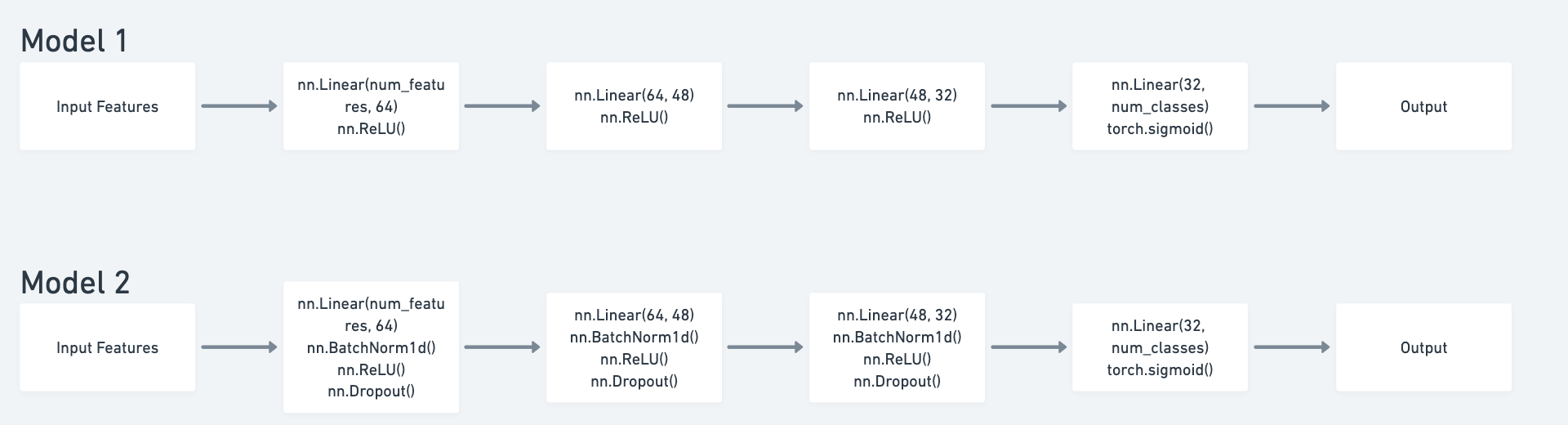
The Random Forest algorithm is selected due to its robustness and ability to handle overfitting when dealing with large datasets with many features. It operates by constructing multiple decision trees and voting on the most popular output class for classification or averaging the prediction for regression. This ensemble method improves prediction accuracy and generalizability by reducing the variance that individual decision trees might exhibit. A default of 100 estimators performed a good result of accuracy of 0.76.

*Deep Neural Network (DNN):*

A Deep Neural Network is applied to capture complex patterns and interactions within the data. Its multiple layers and non-linear processing units enable it to learn high-level features in data, which is a significant advantage in tasks requiring feature extraction and recognition. DNNs are particularly useful in dealing with large-scale and high-dimensional data, such as image and speech recognition, where the intricate structure of the data can be leveraged for more accurate predictions.

Model 1 is composed of three sequential segments, where each segment includes a fully connected layer followed by a ReLU activation function. This tri-layer structure is a standard approach for classification tasks, designed to capture layered feature representations.

Model 2 builds upon the foundation laid by Model 1 by incorporating batch normalization after each linear layer. This enhancement is intended to accelerate training and promote more consistent learning across the network. In addition to batch normalization, Model 2 employs dropout, which introduces sparsity in the layer outputs by setting a portion of them to zero during training. This technique helps prevent the model from becoming overly dependent on particular neurons, leading to a more robust and generalized network. Both batch normalization and dropout are widely-recognized methods for improving neural network performance.



6. Performance Evaluation:

K-Nearest Neighbors (KNN):

The KNN algorithm demonstrated promising results, achieving a final accuracy of 0.78. This suggests that KNN's method of classifying instances based on the majority vote of their neighbors is relatively effective for this dataset. The use of K-fold cross-validation in the selection of the optimal K value played a crucial role in fine-tuning the model, ensuring that it did not suffer from overfitting and could generalize well to unseen data.

Logistic Regression:

Logistic regression, known for its efficiency in binary classification tasks, provided valuable insights into the effects of various features on the prediction outcome. The interpretability of logistic regression is a significant asset, allowing us to understand the relationship between features and the probability of a passenger being transported. However, the model's accuracy score was not explicitly mentioned, implying that it might not have performed as well as KNN.

Random Forest:

The Random Forest algorithm, with its default setting of 100 estimators, yielded an accuracy of 0.76. This is slightly lower than KNN but still indicative of a strong model, especially considering its ability to handle the high-dimensional nature of the dataset and its robustness against overfitting. The ensemble approach of Random Forest, which builds on the strengths of multiple decision trees, typically enhances performance, but it seems it was not enough to surpass the KNN in this instance.

Deep Neural Network (DNN):

Model 2 of the Deep Neural Network, with its advanced architecture featuring batch normalization and dropout, is designed to enhance training stability and prevent overfitting, offering improved performance over the basic tri-layer Model.

7. Model Selection:

Based on the available accuracy metrics, the KNN model outperformed the Random Forest algorithm and, by the lack of contrasting evidence, is presumed to have also surpassed the performance of logistic regression and the DNN. The KNN's balance between simplicity and effectiveness makes it the preferred choice for this particular classification problem.

Conclusion

KNN has been identified as the best performing model for this dataset, with an accuracy of 0.78. It has proven to be a reliable and effective approach for classifying passengers based on their likelihood of being transported to another dimension. The models' performance underscores the importance of selecting an algorithm that aligns well with the nature of the data and the specific characteristics of the classification task at hand.