Semester Project

Ashok Kamath and Daeyeop Kim[1]*

Abstract
In this project, we attempt to classify passengers on a spaceship as transported or not and we want to predict the sale prices of housing.

Keywords
Spaceship — Housing — Prices

[1] *Computer Science, School of Informatics , Computing and Engineering, Indiana University, Bloomington, IN, USA*

Contents

1. Problem and Data Description

1.1 Spaceship Titanic Section

For the spaceship dataset, the problem is predicting the classification of passengers as to whether they were transported to another dimension.

There are 14 attributes and 8693 entries in the training set and there are 13 attributes and 4277 entries in the test set. Therefore, the train-test-split ratio is about 2/3 for training and 1/3 for testing. There is one less attribute for the test set because the classification variable is not there.

There are 6 numerical attributes in the dataset. 5 attributes, Room Service, Food Court, Shopping Mall, Spa and VR Deck, are all monetary measures so they are ratio measurements. These 5 attributes describe how much a passenger spent on the attribute. Age is another ratio variable since it has a true zero point but for this dataset it is discrete. The categorical attributes of the data are the Home Planet, Cryo-Sleep, Cabin, Destination, and VIP features.

The Home Planet is the planet that the passenger departed from while Cryo Sleep is whether the passenger chose to be put to an extended sleep during the space trip. Cabin is the cabin information for the passenger and is in the form of deck/num/side. Side is either P, which means Port, or S, which means Starboard.

1.2 Housing Prices Section

For the housing prices dataset, the problem is predicting the sale prices of the houses and the objective is to use the attributes in the dataset to predict the sale price.

For this dataset, the training set has 1460 rows and 81 columns while the test set has 1459 rows and 80 columns. Therefore, the train-test-split ratio is about 50-50.

The columns describe features of each house such as the Overall Quality, the neighborhood, land slope, year built, roof style, bedroom, kitchen, and gross living area. Since there are so many columns, it could be productive to find the columns that are most correlated with Sale Price and use those in regression.

To be more specific about the meaning of some ambiguous columns, Overall Quality refers to the overall finish and material quality while Basement Exposure refers to whether the basement is walkout or garden level.

2. Data Preprocessing & Exploratory Data Analysis

2.1 Spaceship Titanic Section

2.1.1 Handling Missing Values

For handling the missing values, we first found the number of missing values for each attribute. Most of the columns had missing values. From there, for the ratio attributes, we looked at the distribution and found that for all the monetary attributes (Room Service, Food Court, Shopping Mall, Spa and VR Deck), a majority of the values were 0 since the median was

0. For that reason, we decided to fill in the missing values for these columns with 0. For Age, on the other hand, the median and the mean were similar so we chose the mean to fill in the missing values. For the categorical attributes of Home Planet and Destination, we found the mode and then filled in the missing values with this figure since there was no average or median because it is categorical data.

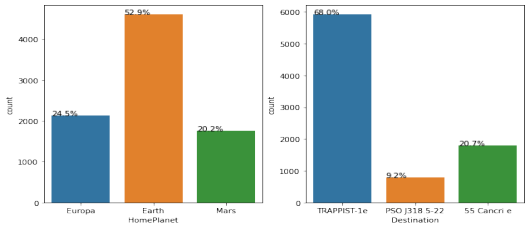

Figure 1. Home Planet and Destination Before covering missing value

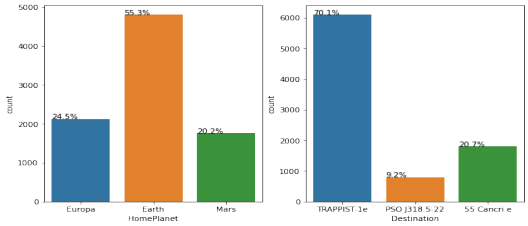

Figure 2. Home Planet and Destination After covering missing value

As this count-plots shows that there were percentages change after the missing value covers. Like Figure1 and 2, there were 1 or less percentage were changed so it does not effect to the result at all.

2.1.2 Exploratory Data Analysis

Before the we check for the classifications and do all the technique for this problem. We checked the target variable is balanced or not.

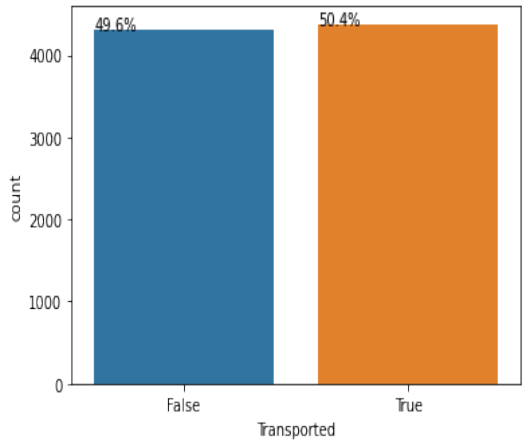

Figure 3. Graph for Target variable(”Transported”)

From the count plot, we can see that the target variable is 49.5% False and 50.4% True. We can see that the target variable is highly balanced so we don’t have to consider technique for under or over sampling.

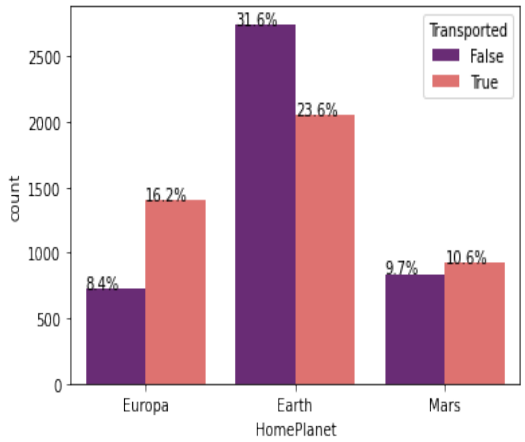

Figure 4. Graph for HomePlanet with Target(”Transported”)

To better understand the data, we started by creating some count plots. Our count plot of Home Planet, which was grouped by whether the passenger was transported, showed that about 2/3 of those from Europa were transported. For the passengers that were from Earth, they had around a 60 % chance of not being transported. For Mars, however, it was evenly distributed.

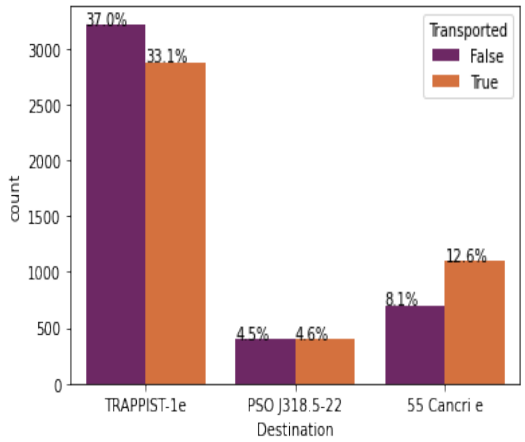

Figure 5. Graph for Destination with Target(”Transported”)

For the Destination attribute, if the passenger was headed for TRAPPIST-1e or PSOJ318.5-22, there was not much of a difference in whether the passenger was transported, but if the passenger was heading towards 55 Cancri-e, then they had about a 60 % chance of being transported.

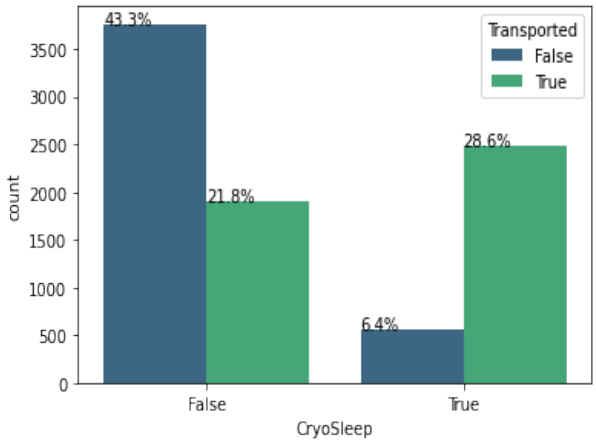

Figure 6. Graph for CyroSleep with Target("Transported")

For the CryoSleep attribute, if it was true that the passenger was in this state, then they had an 82% chance of being transported. If there were not in CryoSleep, then they had a 33% chance of being transported. This indicates that CryoSleep will be an important variable in predicting the classification of passengers in the test set as to whether they were transported.

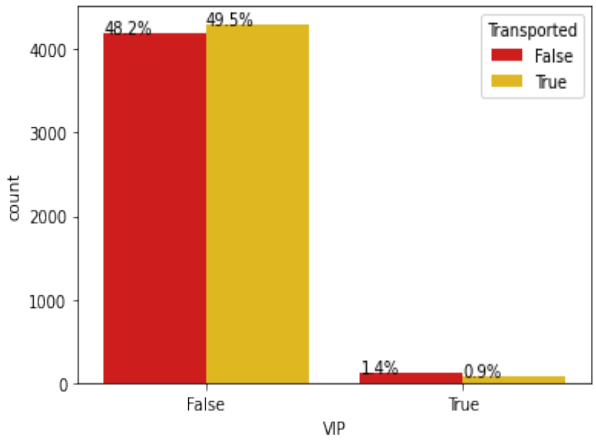

Figure 7. Graph for VIP with Target("Transported")

For the VIP attribute, we found that the distribution for transportation was quite even if the passenger was not VIP. If the passenger was VIP, on the other hand, then there was only a 40% chance of being transported, which means that this variable is helpful in predicting transportation only for the passengers who are VIP.

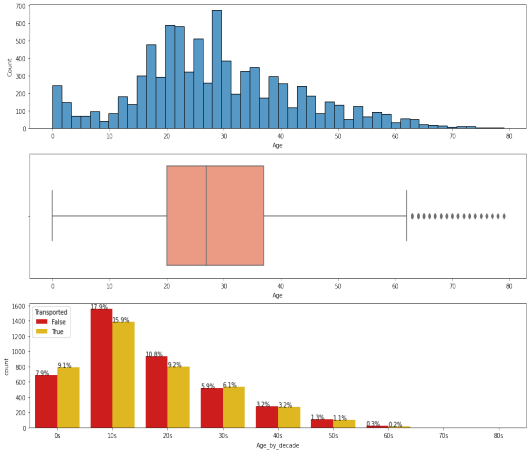

Figure 8. Graph for Age distribution and Age with Target("Transported")

For the Age attribute, distribution shows that 15 30years old were most number of groups in the ship and over 65 years old were least number of groups in the ship And count-plot shows 10s, 20s has higher percentage to be not transported than transported. 0s, 30s has higher percentage to be transported than not transported. Also, over 30s has about equal or 0.1% different percentage transported and not transported.

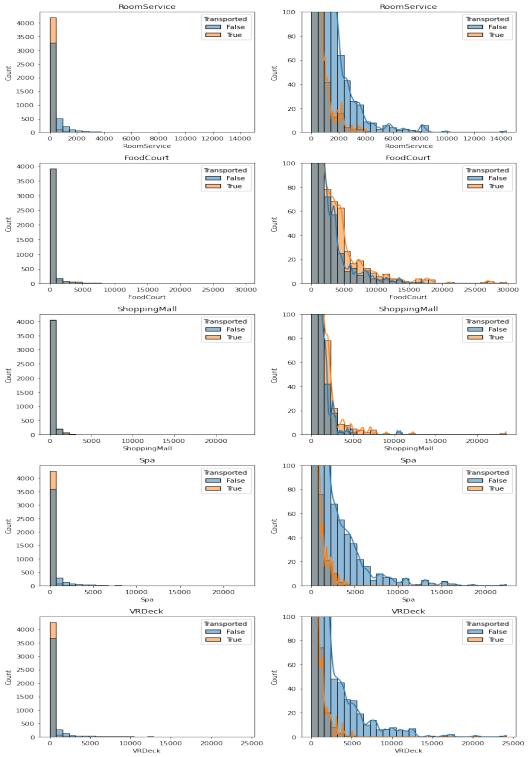

Figure 9. Graph for Bill amounts with Target("Transported")

For the numeric categories(Room service, Food Court, Shopping Mall, Spa, VR Deck) attribute, the graphs are shows that there are small number of outliers and people who were transported tended to spend less.The left-side graph show

mostly which means the most people did not spend any money. And the right-side graphs show on right graphs, the distribution of spending decays exponentially. Since Room Service, Spa, and VR Deck have different distribution to Food court, and Shopping mall, we can might think of think as luxury and essential amenities.

We did PCA since it gives a low dimensional representation of the data while it preserves the local and global structure. We conducted the principal component analysis (the number of principal components is designated as 3). Also, for PCA, we reduced dimension to an array of size (8693, 13). We used the 'extended variance ratio' method to obtain the variance ratio (unique value/total eigenvalue) and it describes variance which represents the information described using specific principal components.

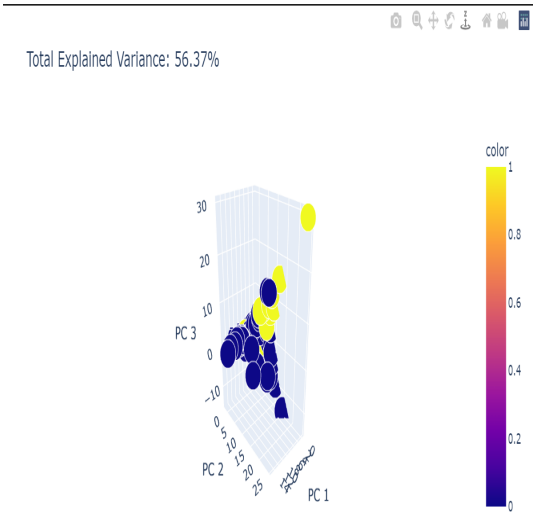

Figure 10. Visualize the distribution of principal component in 3-Dimension

As this 3-Dimension scatter plot show, it explains 56.37% of the principle component from the original data.

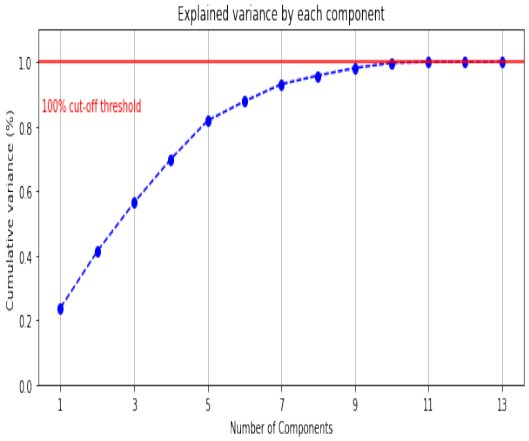

Figure 11. Graph for explained variance by each component)

As this graphs shows, More than 6 principle component describe the most of distribution.

2.2 Housing Prices Section

2.2.1 Handling Missing Values

Since the number of columns was so high, we decided to first find the columns that were the most correlated with sale price and only work with those columns for prediction. For that reason, we only need to handle the missing values in that subset of columns.

After finding the columns with the greatest correlation to Sale Price, we had only two attributes that had missing values, Garage Year Built and Masonry Veneer Area, which had 81 and 8 missing values respectively. From there, we looked at the distribution of both attributes to choose the most reasonable method of filling the missing values. For Garage Year Built, the average and the median are about the same, so either would work and we chose to use the mean. For Masonry Veneer Area, the median was 0 while the mean was 103, indicating a right skewed distribution, so it made sense to fill the missing values with the median, 0.

2.2.2 Exploratory Data Analysis

To further analyze the relationships between Sale Price and the subset of columns that had high correlation with Sale Price, we chose to create scatter plots.

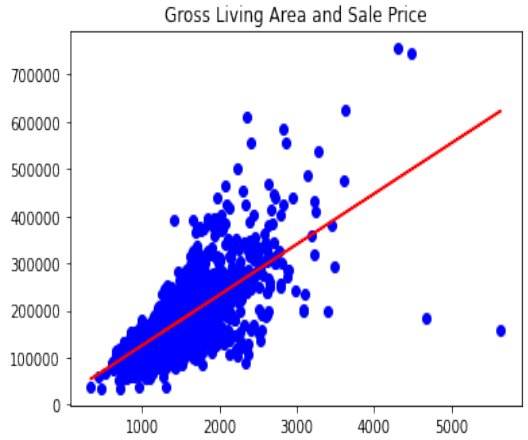

Figure 12. Scatter plot of Gross Living Area with Sale Price

The first scatter plot shows that Gross Living Area and Sale Price have a strong positive relationship, but there are some outliers with high gross living area yet they have a low sale price.

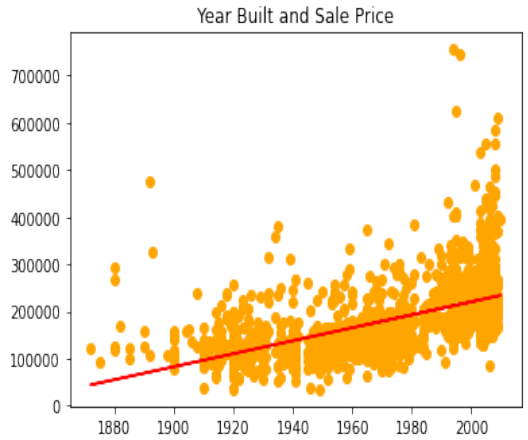

Figure 13. Scatter plot of Gross Living Area with Sale Price

The second scatter plot, which shows the relationship between Year Built and Sale Price, indicates a slight positive relationship between the two variables.

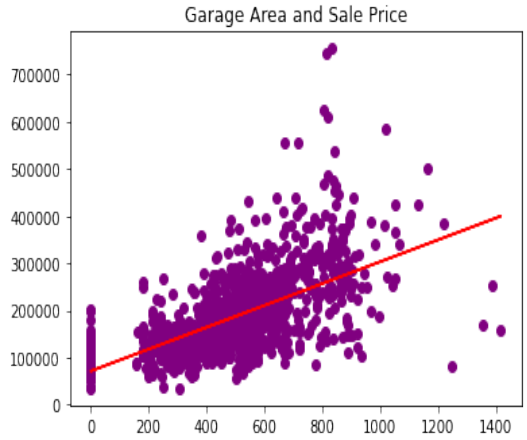

Figure 14. Scatter plot of Garage Area with Sale Price

The third scatter plot shows that Garage Area and Sale Price have a stronger positive relationship than Year Built but not as strong as between Gross Living Area and Sale Price.

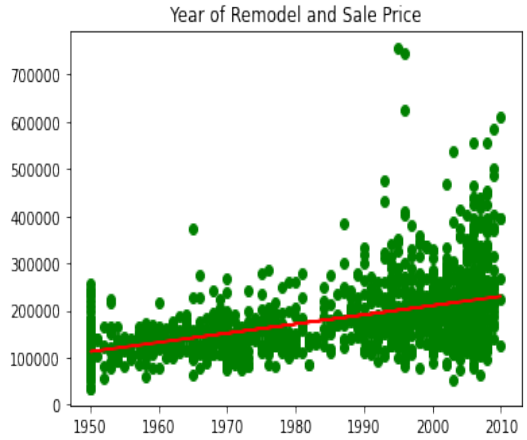

Figure 15. Scatter plot of Year of Remodeling with Sale Price

The Year of Remodeling has the weakest positive relationship with Sale Price of the 4 scatter plots.

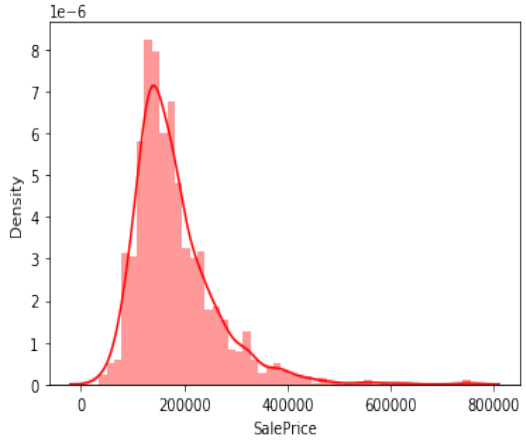

Figure 16. Sales price distribution

We also wanted to see the distribution of some of the attributes that were included in the subset of the data we selected earlier. For Sale Price, we found the data is mostly normally distributed with a slight right skew and gross living area had a similar distribution, which would likely explain the strong correlation between the two variables.

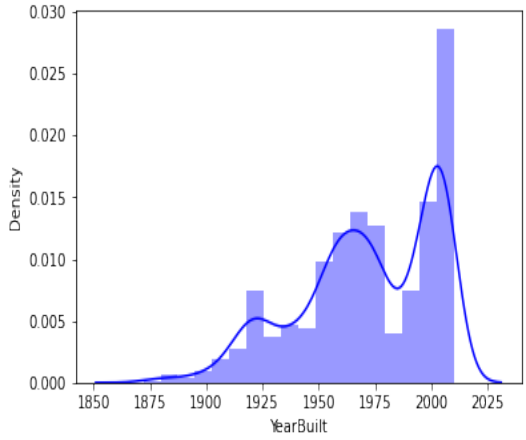

Figure 17. Year Built distribution

Year Built had a distribution that was left skewed. There were very few houses that were built in the 19th century and most were built after 1950. There were many houses that were built after 2000. The latest a house was built was 2010 and the earliest was 1872.

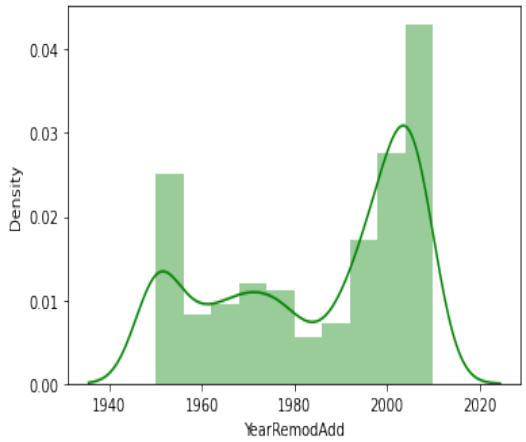

Figure 18. Year Remodeling distribution

The Year of the Remodeling tended to be either quite recent or quite old, as far back as earlier than 1960. The mean year of remodeling was 1985 while the mean was 1994.

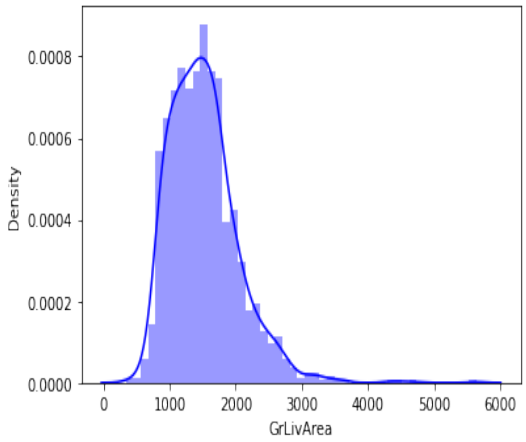

Figure 19. Gross Living Area distribution

The Gross Living Area has a distribution that is right skewed. There were lots of data points from 500 to 3000 .

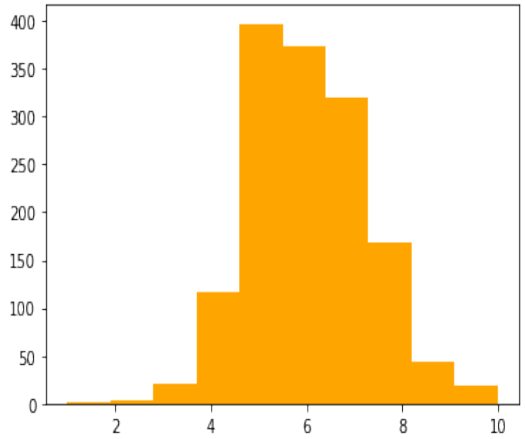

Figure 20. Overall Quality distribution

Overall Quality was normally distributed with the median and mean at 6.

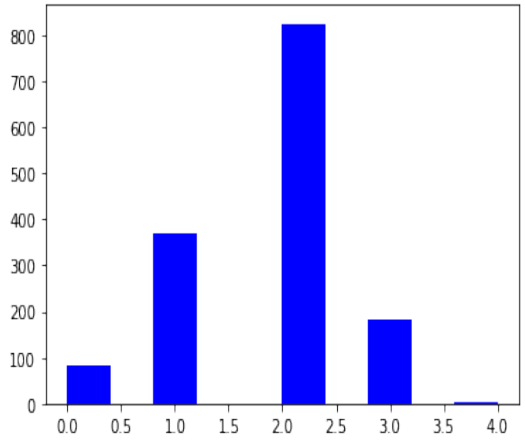

Figure 21. Garage Cars distribution

For Garage Cars, the data was normally distributed but

with a slight left skew since the mean was slightly less than the mean. The maximum number of Garage Cars was 4 while some houses had no Garage Cars.

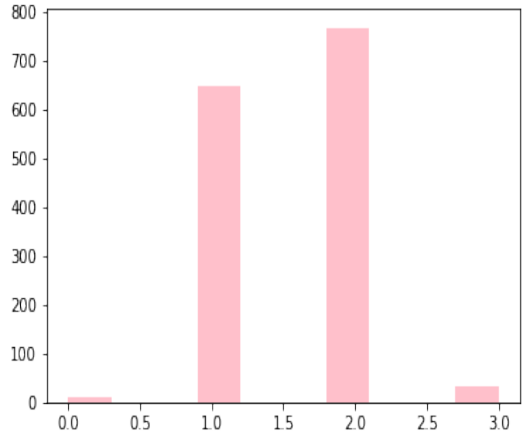

Figure 22. Full Bath distribution

For Full Bath, at least 50 percent of houses had at least 2 while some had none and the maximum number of full baths was 3.

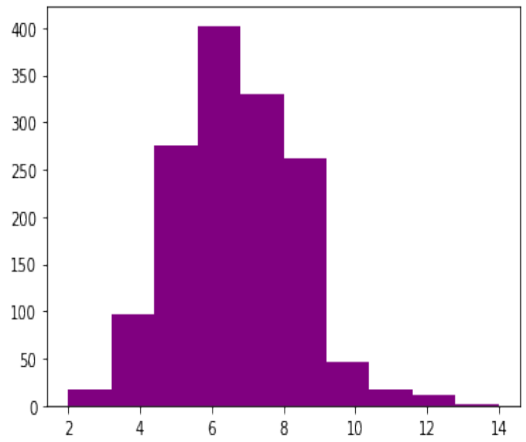

Figure 23. Total Rooms Above Ground distribution

Additionally, Total Rooms Above Ground was normally distributed with the maximum being 14 and the minimum being 2. The data was slightly right skewed since the mean was 6.5 and the media was 6.

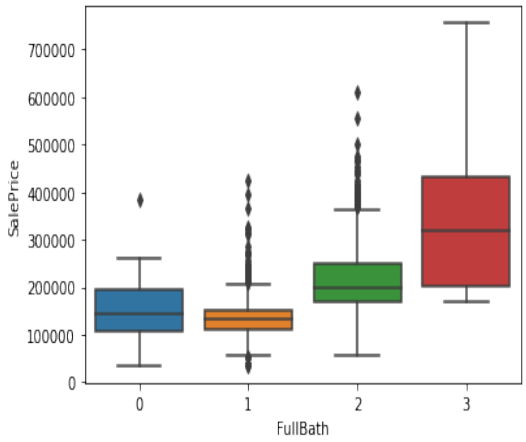

Figure 24. Boxplots of Full Bath with Sales price

Furthermore, our boxplots indicate that with more Full Baths, the price of the house likely increases. However, there is a lot of variance when the house has 3 baths and there are outliers that could affect regression results when there are 0, 1, or 2 baths.

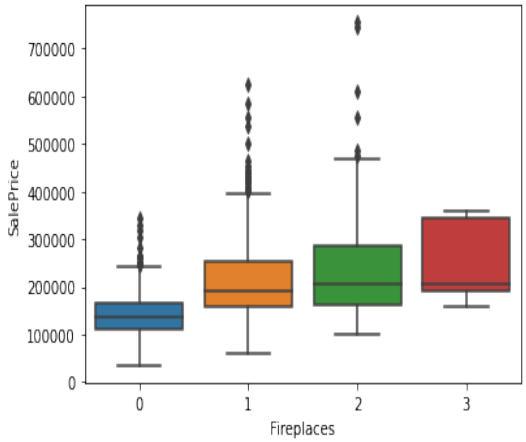

Figure 25. Boxplots of Fireplace with Sales price

For Fireplaces, on the other hand, when there are 3, there is not much variance in the sale price, but there is a strong right skew. For 0, 1 and 2 Fireplaces, the data is more evenly distributed, but there are more outliers.

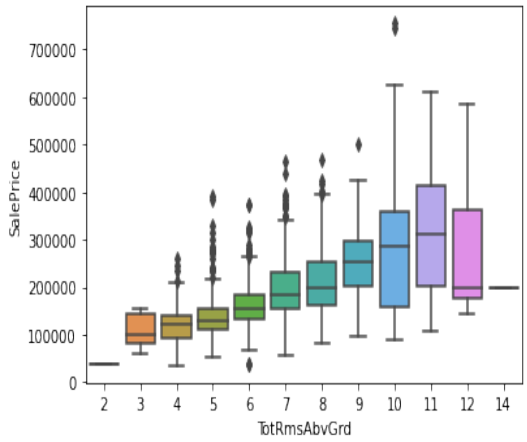

Figure 26. Boxplots of Total Rooms Above Ground with Sales price)

For the Total Rooms Above Ground attribute, there is hardly any variance in Sale Price when there are only 2 or 14 rooms above ground, which is probably due to a lack of entries with this number of rooms above ground. The boxplots indicate that as the total rooms above ground increases, the Sale Price typically will increase. For rooms above ground ranging from 4-10, there are outliers on the positive side.

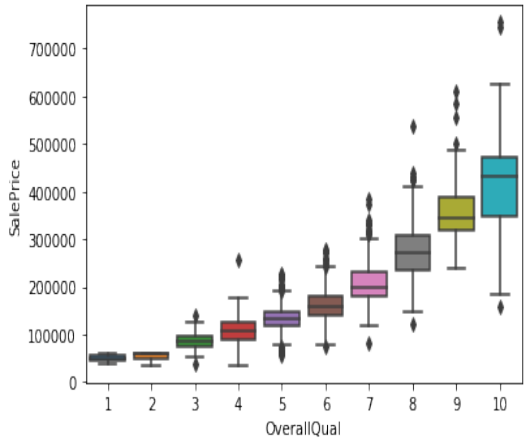

Figure 27. Boxplots of Overall Quality with Sales price)

Finally, Overall Quality has a strong positive relationship with Sale Price. There is little variance in price when the Overall Quality is 1,2, or 3, but the variance in price increases as the Overall Quality increases.

3. Algorithm and Methodology

3.1 Spaceship Titanic Section

For this classification problem, the first method we used was the k-nearest neighbors method. It works by taking an entry that we want to classify, computing the distance between this point and all the points in the training data, and then using the majority class of the nearest neighbors within the training data to classify the testing point. The k-nearest neighbors method is a lazy learner but it can be very powerful when used in the right situations.

Another technique we chose was logistic regression since it can predict a binary outcome, such as transported or not. This supervised machine learning algorithm executes the classification by using the logistic function along with maximum likelihood estimation. Besides k-nearest neighbors and logistic regression, we also used the Naive Bayes classifier, which uses Bayes' Theorem and prior probabilities to arrive at a posterior probability for Transported and Not Transported.

Additionally, we created a decision tree classifier, which cycles through the attributes to see which can best reduce the entropy of the data and then uses that attribute as a decision node in the tree. It can use a variety of measures to see how much the entropy decreases, ranging from information entropy to Gini Impurity. A more advanced model similar to the decision is the random forest model, which is an ensemble method that uses a multitude of decision trees and then takes the majority vote of the decision trees to classify a data point. A random forest model uses bootstrapping to create the decision trees and then randomly selects attributes to use for each decision tree.

Another model we used for this classification problem was extreme gradient boosting. This technique is similar to the random forest model since they are both ensemble methods composed of decision trees. The difference, however, is how the decision trees are constructed. While random forest uses bootstrapping, extreme gradient boosting uses gradient descent on the error residuals of the previously created decision tree to create the next decision tree. We wanted to try this algorithm since it has been very popular over the last few years in the data science community.

We also used the Light Gradient Boosting Machine, which is a distributed, fast decision tree model. Not to mention, we also experimented with the Gradient Boost machine learning algorithm, which is similar to the Extreme Gradient Boosting model and the Light Gradient Boosting Machine. Lastly, we used a neural network multi-layer perceptron as our final model, which uses gradient descent over multiple levels of weighted nodes to arrive at a prediction for the class.

3.2 Housing Prices Section

For predicting the sale prices of housing, we used linear regression to start. This algorithm works by minimizing the sum of squared errors when creating a line of best fit through the training data. The line of best fit for the training data is then used to predict the appropriate values for the entries in the testing data.

In addition to linear regression, we used a random forest regressor, which is similar to the random forest classifier that we used on the spaceship titanic dataset. The random forest regressor, however, averages the results of the decision trees in the forest rather than take the majority vote. We also used a

support vector regressor which attempts to predict new entries by first projecting the data into a higher dimension.

Besides linear regression, random forest and support vectors, we also used the gradient boosting regressor and a simple decision tree regressor. The gradient boosting regressor continually improves a decision tree model by calculating the gradient on the errors of the previous decision tree model. The decision tree regressor, as opposed to classifier, outputs a prediction by using the average of the training data in the leaf node that the decision nodes lead to based on the test entry.

Furthermore, we used Bayesian Ridge Regression, which employs Bayesian methods to make up for insufficient or weakly distributed data. In tandem, we used Lasso Regression, which performs a shrinkage technique that makes the data values shrink towards a central point. Lastly, similar to the classifier models for spaceship titanic, we created an MLP and KNN regressor.

4. Experiments and Results

4.1 Spaceship Titanic Section

First, we checked the correlation matrix to find the relationship between all the features while also checking for redundancy.

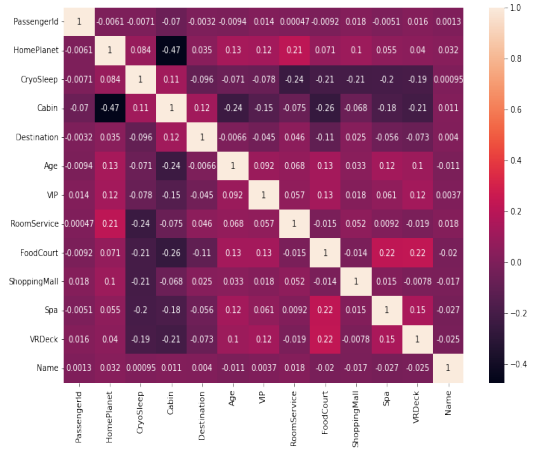

Figure 28. Graph for explained variance by each component)

The correlation matrix shows that Passenger Id, Cabin, Age, Name can be selected for drop.

Accuracy percentage before K-fold	
Model Name	Accuracy percentage
LGBM	78.29%
XGB	78.29%
Random forest	78.75%
Gradient Boost	78.33%
Logistic Regression	78.20%
NNMLP	77.28%
KNN	73.55%
Decision Tree	74.06%
Naive Bayes	68.54%

Accuracy percentage after K-fold	
Model Name	Accuracy percentage
LGBM	79.32%
XGB	78.91%
Random forest	78.89%
Gradient Boost	78.89%
Logistic Regression	78.68%
NNMLP	77.37%
KNN	76.10%
Decision Tree	75.00%
Naive Bayes	69.60%

For k-nearest neighbors, the cross validation accuracy on the training data was about 76.10%. We expected to see a mediocre rate of accuracy for this method since a lazy learner may not work that well on this type of data.

For logistic regression, on the other hand, the cross validation accuracy on the training data was about 78.68%, which was a major improvement from the k-nn accuracy rate. Logistic regression is a powerful tool for binary classification, justifying the result.

In contrast, Naive Bayes performed weakly for cross validation accuracy on the training data, which could be expected since Naive Bayes makes an assumption of independence among all the features, yet it was highly unlikely that features such as RoomService, FoodCourt and ShoppingMall were independent. If someone spends a lot at one location, then they are likely to spend a lot at another location, which implies dependence.

The decision tree performed better for cross validation accuracy on the training data than k-nn and Naive Bayes, but was not as accurate as logistic regression. The decision tree had an accuracy of about 75.00%. We were surprised to see the decision tree perform better than k-nn and Naive Bayes since decision trees can sometimes have trouble with accuracy due to overfitting. With the right parameters, however, such as maximum depth, that problem can be avoided.

From there, our ensemble methods had higher accuracy rates for cross validation on the training data. The random forest classifier had performed at about 79.32% while the extreme gradient boosting model had an accuracy rate of about 78.91%. The Light Gradient Boosting Machine and the Gradient Booster Classifier had accuracies around 78.89% too. The Neural Network multi-layer perceptron performed at an accuracy of about 77.37%, but took a long time to train.

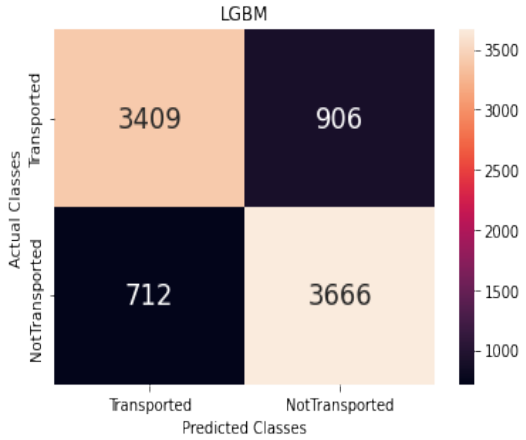

Figure 29. confusion matrix for LGBM (most accurate model)

Since the confusion matrix is a table for comparing predicted and actual values to measure prediction performance through training we can find the accuracy in the diagonal of confusion matrix, precision in the horizontal of confusion matrix, and recall in the vertical part of confusion matrix.

classification report			
rates	Transported	NotTransported	accuracy
precision	0.827226	0.801837	0.813873
recall	0.790035	0.837369	0.813873
f1-score	0.808203	0.819218	0.813873

According to our model, we get a precision accuracy of about 81.40% and a recall accuracy of about 81.40%. For f1-score, we do not need to use this since the target variable is balanced.

Since the Light Gradient Boosting Machine worked the best in cross validation, we used that model on the test data for predictions. This gave us an accuracy rate of 79.32%, which placed us on the leader board at a rank of 1118 out of 1866 teams.

4.2 Housing Prices Section

Accuracy percentage with different model

Threshold percentage with 9 Models	
Model Name	Percentage threshold
Linear Regression	71.8187%
Random Forest Regression	81.9776%
Support Vector Regressor	-5.0832%
Gradient Boosting Regressor	84.2915%
Decision Tree Regressor	67.4467%
Bayesian Ridge Regression	72.1804%
Lasso Regression	71.8188%
MLP Regressor	68.2111%
KNN Regressor	72.1854%

Out of all the models, the Gradient Boosting Regressor model performed the best with a cross validation accuracy rate of about 84.29%. We were impressed with this rate of accuracy since most of the other models were not able to pass the 80% threshold. This demonstrates why the Gradient Boosting algorithm has been so popular within the data science community lately.

The only other model that was able to pass the 80% threshold was the random forest regressor model, which for cross validation on the training data had an accuracy rate of about 82%. This was another model that has its foundations in decision trees, except random forest is not as strategic in improving performance with the creation of each new decision tree, which could explain why it was not as accurate as the Gradient Boosting Regressor.

The KNN Regressor and the Bayesian Ridge Regressor had accuracies slightly above 72% while Lasso and Linear Regression models had accuracies that were slightly below 72%. The MLP Regressor had an accuracy of about 69% for the training data in cross validation while the decision tree model had an accuracy of about 64%. Lastly, the support vector regressor in cross validation failed to report a reasonable accuracy rate.

Since the Gradient Boosting Regressor performed the best in cross validation, we used that model on the testing set and submitted our results to Kaggle, which reported we had an error of .15 while the best of all time submission had an error of .00. We placed on the leaderboard at 2474 out of 4130 spots on the leaderboard.

5. Summary and Conclusions

5.1 Spaceship Titanic Section

For this section, the objective was to classify passengers as transported or not. The training set had more than 8500 entries while the test set had about 4300. There were 13 attributes, excluding the attribute of transported or not. Many of the attributes measured how much a passenger spent at different locations on the spaceship titanic.

When handling missing values, we evaluated whether an attribute's missing values would best be filled by the median, mean or mode of the attribute based on the distribution of the attribute. Through our exploratory data analysis, we found important pieces of information. We discovered that of those from Europa were transported and if a passenger was in CryoSleep, then they had an 82% chance of being transported. If they were not in CryoSleep, then they had only a 33% chance of being transported. We also noticed that the passengers who were VIP had only a 40% chance of being transported while there was not much distinction among those who were not VIP.

When modeling for this problem, we used a variety of algorithms, ranging from k-nearest neighbors, Naive Bayes and Decision Tree to Extreme Gradient Boosting and Random Forest. Ultimately, the Light Gradient Boosting Machine

Classifier was the most accurate for cross validation on the training data so we used that on the test data and for the Kaggle submission, which resulted in an accuracy rate of 78.98%, placing us on the leaderboard at a rank of 1118 out of 1866 teams. The best submission, in rank 1, had a score of about 82%. Our result could be improved by hyper-tuning the model, which could be done through a grid search.

Housing Prices Section For this section, the goal was to predict the sale prices of housing. The dataset had 80 attributes, excluding the target variable of sale price. All of the features described the house that was sold and some examples include Overall Quality, year built, roof style, number of bedrooms and gross living area. For handling the missing values in this dataset, we employed the same reasoning as we did for the Spaceship Titanic dataset.

In our exploratory data analysis, we only worked with columns that had the greatest correlation with sale price since the number of attributes was high, so we wanted to narrow it down. We noticed a strong positive relationship between Gross Living Area and Sale Price via a scatter plot. Additionally, we found Sale Price was normally distributed but with a slight right skew, similar to the distribution for Gross Living Area. Our boxplots showed that a house with more Full Baths will likely have a higher price, but there is more variance as the number of full baths increase. There was a similar relationship between Overall Quality and Sale Price since the variance increased as the Overall Quality increased.

5.2 Housing Prices Section

For this section, the goal was to predict the sale prices of housing. The dataset had 80 attributes, excluding the target variable of sale price. All of the features described the house that was sold and some examples include Overall Quality, year built, roof style, number of bedrooms and gross living area. For handling the missing values in this dataset, we employed the same reasoning as we did for the Spaceship Titanic dataset.

In our exploratory data analysis, we only worked with columns that had the greatest correlation with sale price since the number of attributes was high, so we wanted to narrow it down. We noticed a strong positive relationship between Gross Living Area and Sale Price via a scatter plot. Additionally, we found Sale Price was normally distributed but with a slight right skew, similar to the distribution for Gross Living Area. Our boxplots showed that a house with more Full Baths will likely have a higher price, but there is more variance as the number of full baths increase. There was a similar relationship between Overall Quality and Sale Price since the variance increased as the Overall Quality increased.

To predict the sale prices of housing, we used various models ranging from Bayesian Ridge and Lasso Regression to KNN Regression. Through cross validation on the training data, we found that the Gradient Boosting Regressor performed best so we used that on the test data, giving us an error rate of .15, which had us place at 2474 out of 4130 spots on the leaderboard while the leading submission, in rank 1, had an error rate of .00. Our model could be improved not only by performing a grid search for each model, but also removing outliers, using columns besides those that are highly correlated with sale price, and other techniques.

References

SAMUEL CORTINHAS /Spaceship Titanic: A complete guide