The Titanic Data Analysis

Tie Ma

2024-02-25

In the framework of this analysis, linear regression, logistic regression, and random forest models have been selected to evaluate and contrast their effectiveness on the training dataset for predicting passenger survival within the test dataset. The dataset consists of 11 variables, 8 of which have been used for this study.

Given the constraints of the variable set, the necessity for extensive dimensionality reduction methods such as Principal Component Analysis (PCA) and lasso or double lasso regression is not prioritized, as their performance may not be significantly better than that of simple linear regression and logistic regression. Both models are better suited to situations where there are a larger number of variables with existing multicollinearity. Therefore, I have chosen the linear regression model along with the logistic regression model.

To better grasp complex and nonlinear relationships that linear regression and logistic regression cannot capture, I'm turning to the Random Forest model. This model allows us to dive deeper into the data's patterns, going beyond what simpler models can achieve.

In evaluating the performance of different models on the Titanic dataset, all models had close prediction scales with marginal differences. The Random Forest model had the highest prediction score of 0.77751, followed by logistic regression at 0.77272, and linear regression was the last with a score of 0.77033. The marginal but significant advantage of the Random Forest model compared to logistic regression, with a difference of 0.00479, and 0.00718 compared with linear regression.

```
##
## Call:
## lm(formula = Survived ~ Pclass + Sex_female + Age + SibSp + Embarked,
##     data = TTD)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.06740 -0.21145 -0.08497  0.23357  1.00498
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.874304   0.064553  13.544  < 2e-16 ***
```

```
## Pclass      -0.181151   0.017276 -10.486  < 2e-16 ***
## Sex_female   0.500308   0.027553  18.158  < 2e-16 ***
## Age         -0.005879   0.001076  -5.464 6.05e-08 ***
## SibSp       -0.043021   0.011956  -3.598 0.000338 ***
## EmbarkedQ   -0.002029   0.055018  -0.037 0.970586
## EmbarkedS   -0.071286   0.033864  -2.105 0.035569 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3794 on 882 degrees of freedom
## Multiple R-squared:  0.3954, Adjusted R-squared:  0.3913
## F-statistic: 96.14 on 6 and 882 DF,  p-value: < 2.2e-16
```

```
##
## Call:
## glm(formula = Survived ~ Pclass + Sex_female + Age + SibSp +
##     Embarked, family = binomial, data = TTD)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  2.757830   0.450767   6.118 9.47e-10 ***
## Pclass      -1.158903   0.125587  -9.228  < 2e-16 ***
## Sex_female   2.693571   0.195318  13.791  < 2e-16 ***
## Age         -0.039909   0.007822  -5.102 3.36e-07 ***
## SibSp       -0.333284   0.103738  -3.213  0.00131 **
## EmbarkedQ   -0.028613   0.378611  -0.076  0.93976
## EmbarkedS   -0.454434   0.231608  -1.962  0.04975 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1182.82  on 888  degrees of freedom
## Residual deviance:  785.27  on 882  degrees of freedom
## AIC: 799.27
##
## Number of Fisher Scoring iterations: 5
```

demonstrates the Random Forest model's capability in handling complex and non-linear relationships between variables where the linear models fall short. Because linear and logistic regression models are both constructed based on the assumption of linearity between the dependent and independent variables, an assumption may not hold true in real-world situations.

Compared with linear regression, logistic regression's prediction accuracy increases by 0.00239. This is because logistic regression is designed to model probabilities directly, thereby aptly

handling binary dependent variables with outputs between 0 and 1. Compared to linear regression, which can only generate predictions within the range of 0 and 1, making it less suitable for binary classification tasks.

The first part encompasses data cleaning, inspection, and variable selection through best subset and Ridge regression, comparing the performance and Kaggle scores across two distinct variable sets with logistic regression and linear regression. The results are closely matched, with only a slight, marginally better performance difference favoring logistic regression over linear regression. Nevertheless, the best-performing model is logistic regression, which includes the variables: Pclass, Sex_female, Age, SibSp, and Embarked..

The second part of this assignment involves using a model. I first used 10-fold cross-validation to determine the optimal settings for mtry, and then generated the random forest model.

Part one: The data evaluation process.

The Taitanic training data set including following 11 variable:

- `PassengerId`: The unique Identification number for given to each passage.
- `Survived`: Indicates if a passenger survived(1) or not (0).
- `Pclass` passenger class (1 for first-class, 2 for second-class, 3 for third-class).
- `Sex`: The gender of the passenger (male or female).
- `Age`: The age of a passenger.
- `SibSp`: The number of siblings or spouses aboard.
- `Parch`: The number of parents or children aboard.
- `Fare`: The ticket fare.
- `Embarked`: port of embarkation (C for Cherbourg, Q for Queenstown, S for Southampton).
- `Cabin`: The cabin number.
- `Ticket`: The ticket number.

To better train the model, I removed the names, ticket numbers, and cabin numbers during the data import process because they are not crucial for training the dataset, and having only a few cabin numbers does not contribute to the effectiveness of the model training. First, I checked for any NA (missing) values in the training dataset.

```
## PassengerId    Survived      Pclass         Sex         Age       SibSp
##           0           0           0           0         177           0
##       Parch        Fare    Embarked
##           0           0           2
```

The training data is missing 177 Age values and 2 Embarked values. Considering the size of the training dataset is only 881 rows, removing all the missing Age variables would further reduce the available data for the model. Therefore, I decided to use the average age to fill the missing 177 values. However, I do need to check the average age between the people who survived and those who did not to avoid introducing bias into the training dataset.

The average age by Survived as following.

```
##   Survived      Age
## 1        0 30.62618
## 2        1 28.34369
```

The average age of the entire data set

```
## [1] 29.69912
```

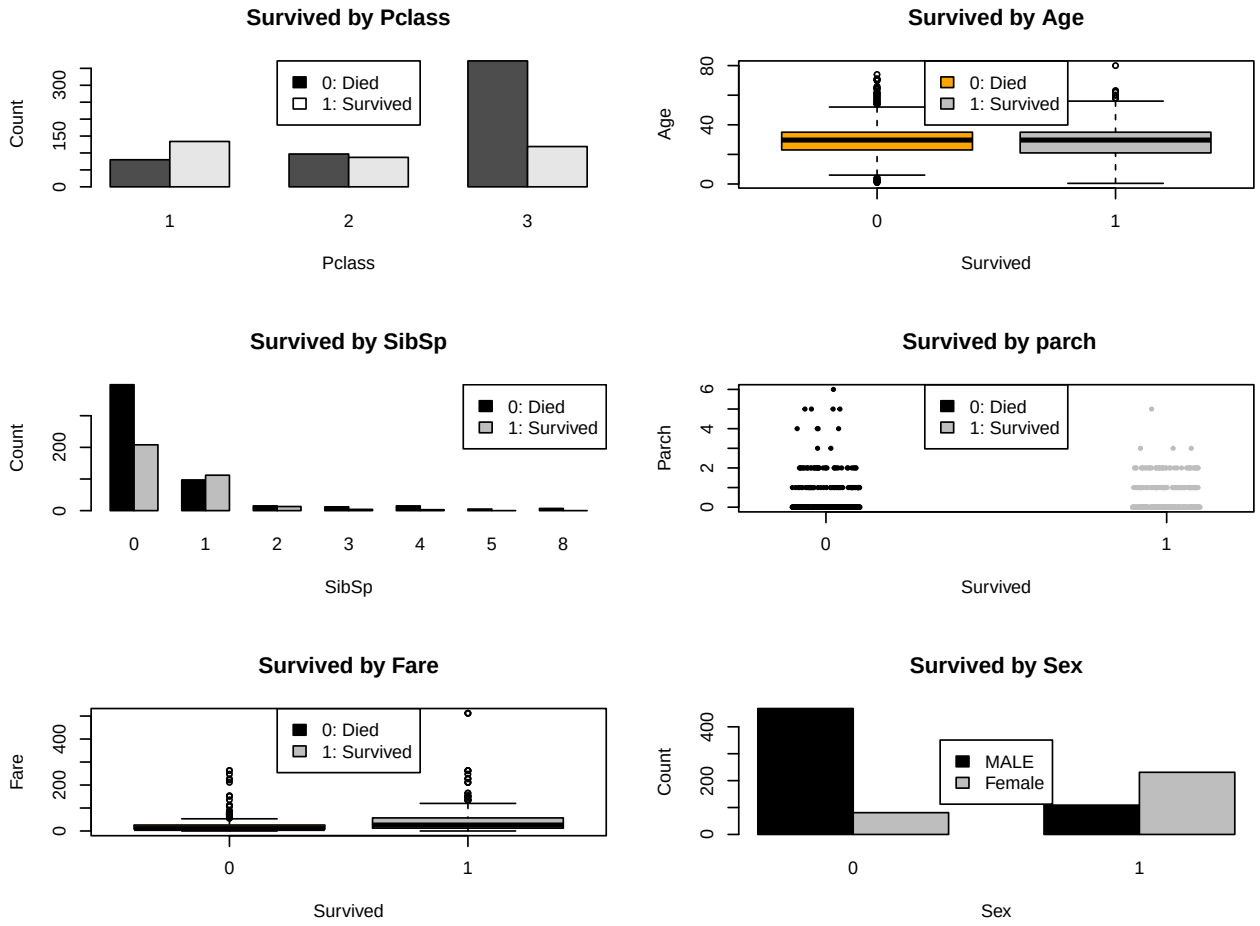
Using the average age to impute missing values into the dataset, the average age within the training dataset is 29.69, which lies between the average ages of those who survived (28.343) and those who died (30.62). Therefore, using the average age to replace the missing 177 values will avoid the creation of artificial peaks in the data distribution, mitigating the potential for bias. Furthermore, it is consistent with the linear relationship assumed between independent variables and the outcome variable in linear regression, and between the independent variable and the log odds of the outcome in logistic regression models.

Furthermore, after filling in all the missing age values, I removed the 2 missing rows of the Embarked value from the training dataset and transformed the 'Sex' column into a dummy variable. This transformation was necessary because machine learning models require numerical input, and converting 'Sex' into a dummy variable. .

Step two: Check The Data.

For a better understanding of the possible patterns in the data and to determine potential variables for the linear regression and logit models, I examined graphs comparing the impact of different variables on passenger survival. From these graphs, we can observe that Passenger Class (Pclass), SibSp, and gender significantly influence the survival of passengers. Furthermore, the fare, representing the ticket price, shows that passengers who paid higher fares were more likely to survive. This observation aligns with the conclusion drawn from the Passenger Class graphs.

Furthermore, the rest of the variables do not provide a visible impact on survival according to the analysis.



To improve the performance of the linear regression, I checked for heteroscedasticity and multicollinearity in the training dataset.

For The heteroscedasticity test, I am using BP test.

```
##
##  studentized Breusch-Pagan test
##
## data:  data_test_1
## BP = 9.8992, df = 7, p-value = 0.1944
```

Since the p-value is 0.19, which is greater than 0.05, we fail to reject the null hypothesis. We conclude that there is not enough statistically significant evidence to suggest heteroskedasticity exists in the dataset

For the multicollinearity, I am using the VIF test.

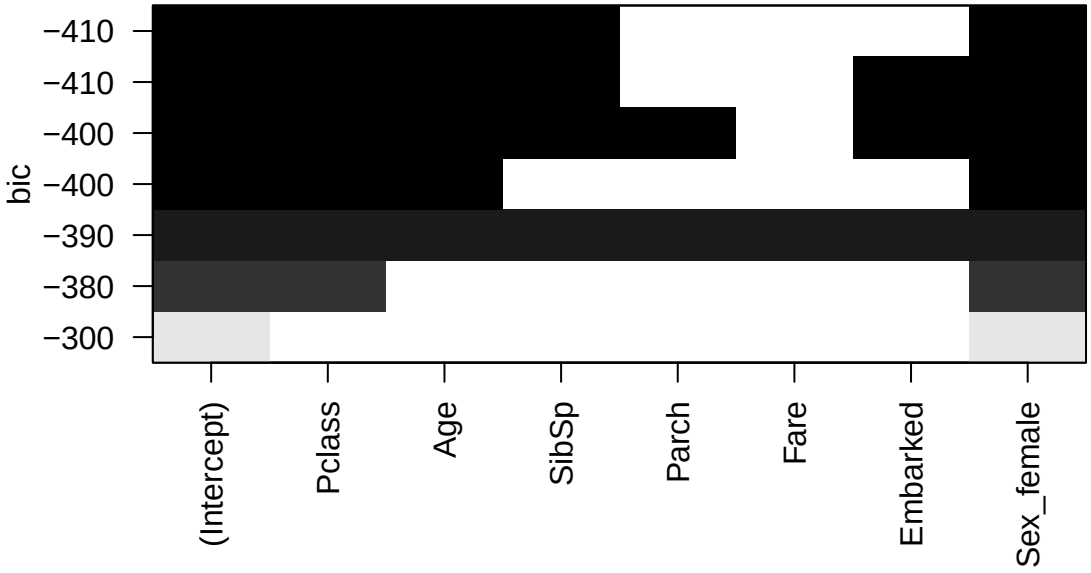
```
##     Pclass        Age      SibSp      Parch       Fare   Embarked Sex_female
##   1.664130   1.203003   1.281804   1.323066   1.645957   1.081159   1.109890
```

All the VIF values for the variable are between 1 and 1.6. This indicates that the training data show moderate correlation which will not impact the model outcome.

Step Three: Choose The Variable

In order to select the optimal variable for constructing the linear regression model without risking over fitting, I will use the best subset selection and Ridge regression to explore all possible linear combinations of variables

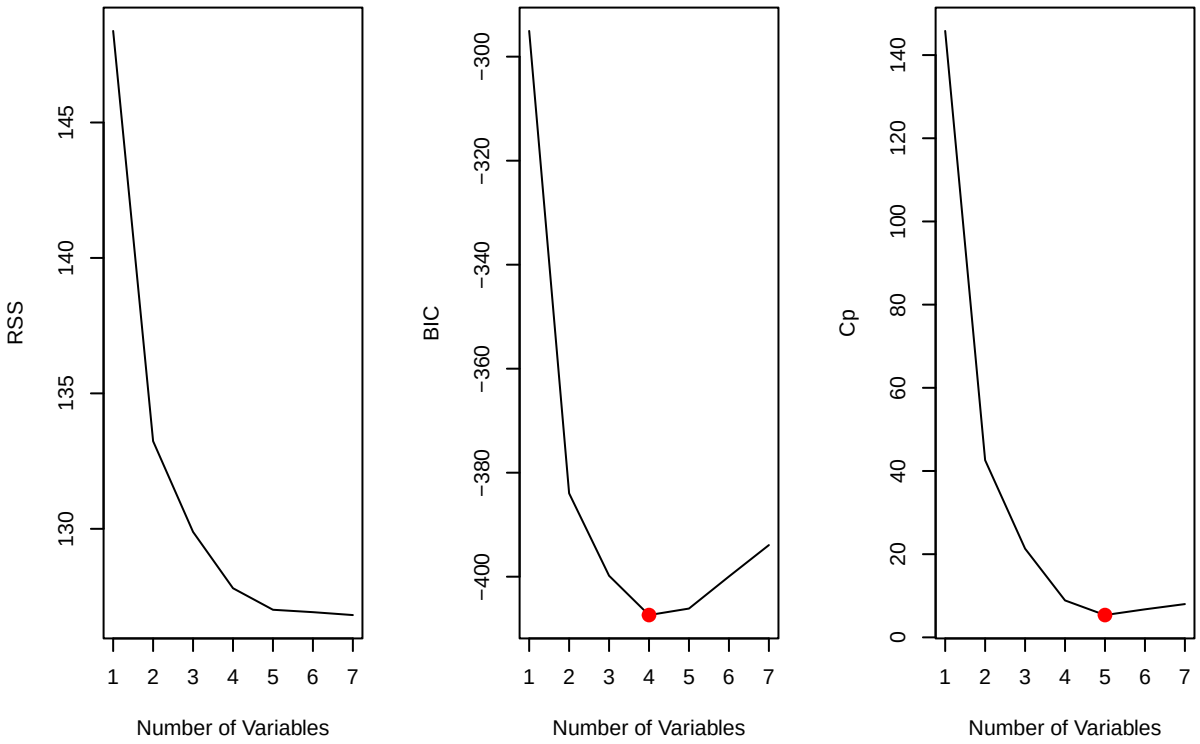
First, I will use the best subset selection to search all possible combinations and compare their BIC values.



The Bayesian Information Criterion (BIC) evaluates a model's performance by considering how well the model explains the data and penalizing models for their number of parameters, which helps prevent the problem of overfitting. A lower BIC value suggests that the model is more effective at explaining the data while using fewer parameters, striking a balance between a good fit and avoiding overfitting.

From the graph, we can see that the combination of Pclass, Age, SibSp, and sex_female has the lowest BIC, indicating that this model is likely the most efficient among the tested combinations in terms of balancing model complexity and goodness of fit.

Next, we determine the number of variables.



Mallows' Cp (Cp) assesses a function similar to that of the BIC; it measures a model's performance by evaluating the trade-off between the model's complexity and its ability to fit the data closely. Similar to BIC, a lower Cp value suggests that the model best balances fitting the data and complexity. The Cp graph suggests that the optimal number of variables is 5.

According to the best subset selection, I should choose the two group with the following variables.

- The first group includes:
  - Pclass, Age, SibSp, Sex_female
- The second group includes:
  - Pclass, Age, SibSp, Embarked, Sex_female"

Now, let's proform the Ridge regression.

```
##                Variable           OLS         Ridge
## Pclass           Pclass -0.1688667659 -0.1564689820
## Age                 Age -0.0058566464 -0.0053474253
## SibSp             SibSp -0.0411820740 -0.0388136461
## Parch             Parch -0.0168729693 -0.0138132743
## Fare               Fare  0.0002829035  0.0003878972
## Embarked       Embarked -0.0353433928 -0.0358988675
## Sex_female Sex_female  0.5059113514  0.4799392450
```

Ridge regression applies a penalty to the size of coefficients to shrink them toward zero, particularly targeting less important predictors. notably, the coefficients for the variables

"Sex_female" and "Pclass" remain relatively high in both OLS and Ridge regression, suggesting a strong relationship with the dependent variable. The coefficients for SibSp, Embarked, and Parch follow this trend. However, Ridge regression only shrinks the coefficients of two variables, "Age" and "Fare," towards zero. Therefore, it does not provide any new model suggestions.

Step Four: Model Comparison and Kaggle Scale

In this step, I divided the training dataset into two parts: 90% of the data will be used for training, and the remaining 10% will serve as the test set. This division will enable us to calculate the Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and Mean Absolute Error (MAE) to evaluate the performance of two sets of linear regression models and two sets of logistic regression models. MSE and RMSE impose larger penalties for higher prediction errors, making the cost of large errors more significant. In contrast, MAE, by focusing more on the general performance of the model, provides an average error without specifically penalizing large errors. By comparing these three measurements, we can better evaluate the model's performance in prediction. Furthermore, I have integrated the Kaggle prediction scale with the table.

```
##           Test Model_1_lm Model_1_glm Model_2_lm Model_2_glm
## 1          MSE  0.1789076   0.1774529  0.1786623   0.1748403
## 2         RMSE  0.4229747   0.4212516  0.4226846   0.4181391
## 3          MAE  0.3271691   0.3216490  0.3259556   0.3169184
## 4 Kaggle Scale  0.7679400   0.7538500  0.7703300   0.7727200
```

The logistic regression model with variable set two (Pclass + Sex_female + Age + SibSp + Embarked) has the highest kaggle prediction scale with the longest MSE, RMSE and MAE.

Part 2: random forest model

To construct the best Random Forest model, the optimal values of `ntree()` and `mtry()` are needed. `ntree()` refers to the number of trees in the forest, which determines how many individual decision trees are built to make the final prediction. `mtry()`, on the other hand, involves choosing how many variables to consider at each decision point when building these trees. Therefore, `mtry()` is more important than the variable `ntree()`; it helps in balancing between model bias and variance, and further constructs a more robust model. As for `ntree()`, all we need to do is set it to a high enough number to stable the error rate of the model.

First, I need to find the optimal value of `mtry()`, which is typically suggested to be set as one-third of the total number of features. The processed data have 10 variables, therefore, I should set the value of `mtry()` between 3 or 4. I am using 10-fold cross-validation to determine the best `mtry()`value from 1 to 10.

This method divides the entire dataset into 10 folds, using 9 of them to train the model and one to test it. Repeat this process 10 times, with every fold of data being used both for training and testing. It examines the model's performance across the entire dataset. This

method will help ensure that the `mtry()` value I choose is truly optimal for our model's performance.

```
## Random Forest
##
## 889 samples
##   9 predictor
##   2 classes: '0', '1'
##
## No pre-processing
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 800, 800, 801, 800, 800, 800, ...
## Resampling results across tuning parameters:
##
##   mtry  Accuracy   Kappa
##    1    0.8087717  0.5741640
##    2    0.8245020  0.6157622
##    3    0.8267748  0.6228800
##    4    0.8324055  0.6353924
##    5    0.8256512  0.6236363
##    6    0.8200332  0.6114304
##    7    0.8189224  0.6100784
##    8    0.8177988  0.6072135
##    9    0.8188968  0.6106127
##   10    0.8188968  0.6101722
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was mtry = 4.
```

According to the output, the `mtry()` = 4 has the highest accuracy with the highest Kappa scale.

Then I constructed my Random Forest with 10,000 trees `ntree()`, set `mtry()` equal to 3, and `nodesize()` to 1 for classifying the output.

```
##
## Call:
##  randomForest(formula = Survived ~ ., data = TTD, ntree = 10000,      mtry = 4, nodes
##                Type of random forest: classification
##                      Number of trees: 10000
## No. of variables tried at each split: 4
##
##         OOB estimate of  error rate: 17.66%
## Confusion matrix:
##     0   1 class.error
## 0 492  57   0.1038251
## 1 100 240   0.2941176
```

Out of Bag error rate (OBB) is measurement of average performance of the forest, at 17.55% The Random Forest method constructs each individual tree by randomly selecting subsets from the entire dataset, a process called 'in-bag' data selection. To evaluate the performance of each individual tree, the algorithm uses the 'out-of-bag' data, which consists of all the data points that were not selected to construct each tree, to evaluate the accuracy of each tree. Consequently, the OOB error rate offers an estimate of the average performance of the entire forest.

The confusion matrix shows that the Random Forest correctly predicted 492 passengers who didn't survive and 241 who survived. However, it made errors on 57 individuals who the model predicted would survive but didn't, and 100 individuals who the model guessed wouldn't survive but did. The error rates are 10.4% for those who did not survive and 29.4% for those who survived. This indicates that this Random Forest model is better at identifying who won't survive than who wil For the random forest model, the prediction score is 0.77751 which is the higher than logistic model and linear regression.

Conclstion:

In this assignment, I used three different models—linear regression, logistic regression, and random forest models—to train on the Titanic dataset and predict passenger survival in the test set. The results suggest that the random forest model delivers the best performance with a prediction score of 0.77751, enhancing accuracy by 0.00718 compared to the best linear regression and logistic regression, which both have a variable set at 0.77033. Comparing logistic regression to linear regression, logistic regression performs slightly better in terms of MAE, RMSE, MSE and prediction scare with 0.00239.This research shows that Random Forest model are more suitable on predict the survive on the titanic data set due to its ability to capture non-linear and complex relationships between variables.