Madison White
Charles Wallace
CSE 198
Homework 8

Executive Report

Our research analyzes the comparative strength of logistic regression and K Nearest Neighbors in classifying survival aboard the Titanic. The goal of our analysis is to standardize a model able to apply links between early twentieth century social factors and shipwreck survival to determine the probability of survival for any given passenger.

Given the initial data set, we first determined which of the fourteen characteristics actually provided insight into whether or not a passenger would survive. After running information gain analysis and deducing which of the variables were applicable, we decided to build the model on five features: pclass(denoting ticket class), embarked(indicating where they boarded the ship), fare(denoting the amount paid to travel aboard the Titanic), age, and sex (describing gender). We then cleaned the data set to make fare a number with two decimal places, which is consistent with how the pound is presented, and to make age a whole number instead of a decimal number.

These features were then used to create both the logistic regression and the knn models. The performance of the models was then determined by seeing what they would predict for the likelihood of survival of a passenger, given certain information about them, and comparing that result with the passenger's true value of survival.

After calculating various indicators of performance, we determined that the knn model outperformed the logistic regression model in predicting the likelihood of survival. Thus, we recommend the use of this model over the use of the logistic regression model.

Technical Report

Introduction

The sinking of the Titanic is a significant event in history. Many passengers aboard died in the crash. The object of this project was to determine the best model for predicting the likelihood of survival for a passenger aboard the titanic. The effectiveness of logistic regression and K-nearest neighbors models in predicting survival served as the primary subjects.

Data

The Titanic manifest Data used to train and test the KNN and logistic Regression models was obtained from the follwing source:

http://biostat.mc.vanderbilt.edu/wiki/pub/Main/DataSets/Ctitanic3.html

This data was divided into two separate datasets to be used for training and testing. The raw data set contains 14 columns: name, sex, age, sibsp, parcnamesh, ticket, fare, cabin, embarked, boat, body, and home.dest, with a total of 1307 rows describing individual passengers. The training set contains data on 1046 passengers, and the test set contains data on 261 passengers allowing for 80% of the raw data set to be used for training and 20% to be held out for testing. Five of the fourteen features in the raw dataset were chosen to train the model: pclass, fare, sex, age and embarked. A subset of the original data set consisting of these five features and their survival target attributes were used for training and testing.

Data Cleansing

Before creating and testing the models, the features with the most influence on the survival rate needed to be identified. Therefore, information gain analysis was run on all 14 features to determine which ones had the strongest relationship with survival rate. The results are recorded in the table below.

	attr_importance
pclass	0.54147890
name	1.60811076
sex	0.15722569
age	0.08304861
sibsp	0.02345611
parch	0.00000000
ticket	1.49451468
fare	0.44410669
cabin	0.36149511
embarked	0.05819713
boat	0.63925162
body	0.04485976
home.dest	0.86439676

The features of name and ticket were not chosen because they are specific to the observation and not a general feature that could be tested.

Additionally, boat was disregarded as the value of the life-boat a person was on biases training and testing, informing the model that they survived without generalizing to cases where the lifeboat data is not known. The body attribute was exluded for similar reasons. A body number is assigned to passengers who died, providing no useful information for those passengers without a body number. Once these were removed, the features with the most information gain were pclass, sex, age, fare, and embarked. Cabin was not used because a person's cabin would inform one of their pclass, and pclass had a higher information gain than cabin. Including it would have been redundant.

Once the features of interest had been identified, the data then needed to be cleaned. First, the features that were decided to be irrelevant for the survival prediction were removed from the data set. Any entries with missing values were also removed from the data set. The original entries for the age of the passengers had values after the decimal place. Since age is not typically expressed as a float number, this value was floored to an integer. The original format of fare also contained multiple decimal place values. However, fare should be recorded in a manner that is consistent with the British pound; therefore, fare was rounded to two decimal places.

Modeling

KNN:

The kknn library was used to train a knn classifier. The kknn classifier takes two parameters: "k" indicating the number of neighbors to be used for classification, and "distance" indicating the integer valued distance parameter used to calculate the minkowski distance between a sample point and its neighbors.

A custom function "kknnResults" was used to determine the best possible values for k and distance. The kknnResults function takes six parameters - the cleaned training data set, cleaned test data set, start value for the distance parameter, end value for the distance parameter, start value for the k parameter, and end value for the k parameter.

The runKKNN function is run within a double for loop iterating through integer values between the start and end values provided for k and distance to produce models testing all possible combinations within the provided range. Upon each iteration a row consisting of performance measures calculated after tabulating and passing a confusion matrix to the "retMetrics" function is stored within a dataframe.

The runKKNN function returns a data frame consisting of the scores for precision, recall, fmeasure, accuracy, k value, and distance value. In turn, the dataframe generated by the kknnResults function can be passed to the "bestResults" function returning the highest scores for precision, recall, fmeasure, and accuracy, along with the parameters used to obtain the respective maximums. Using this process, it was decided that the values distance = 9 and k = 13 produced the most accurate results, yielding the highest accuracy, second best f-measure, second best precision, and second best recall of 1500 models with distances between 1 and 50, and k values between 1 and 30.

Once the parameters for the model were chosen, the model was built again with the runKNN function, then passed to the custom function "retConfusion" generating a confusion matrix. The resulting Confusion matrix was then passed to a custom function "retMetrics" re-calculating precision, recall, f-measure, and accuracy. Once measures for performance were obtained, a ROC graph was plotted and the area under the curve was calculated.

Logistic Regression:

A custom function, "runGLM" was made to create the logistic regression model. A logistic regression model was desired because the value the model should predict is categorical with binary outcomes. The function was the test and training data set. It then called the glm function, which was passed survived as the desired variable to predict, the training data set, and a command that told R to create a logistic, not a linear, regression. The function then tested the logistic regression model on the testing portion of the data set. It predicted the value of survival for the data set based on the five features identified. The function then returned the value of the model's predictions. Then a function, "retConfusion", was called which compared its prediction for survival with the actual value of survival from the test data, and returned a confusion matrix of the comparison results.

The resulting confusion matrix was passed to the "retMetrics" function, which calculated the precision, recall, f-measure, and accuracy of the model. Next, an ROC curve of the model's performance was created and plotted. Finally, the area under the curve(AUC) was calculated for the model. The output from the testing can be found in the Results section.

Results

KNN:

The confusion matrix for the KNN model:

	FALSE	TRUE
0	142	19
1	38	62

The choice of distance = 9 and k = 13 resulted in the following scores for precision, recall, and accuracy (taking survival to be the positive class, and death to be the negative class):

Precision	Recall	F-Measure	Accuracy
0.620000	**0.765432**	**0.6850829**	**0.7816092**

After generating a ROC curve from the KNN classifier's performance on the test set, the area under the curve was computed to be 0.828229813664596.

Logistic Regression:

The confusion matrix for the logistic regression model:

	FALSE	TRUE
0	124	37
1	35	65

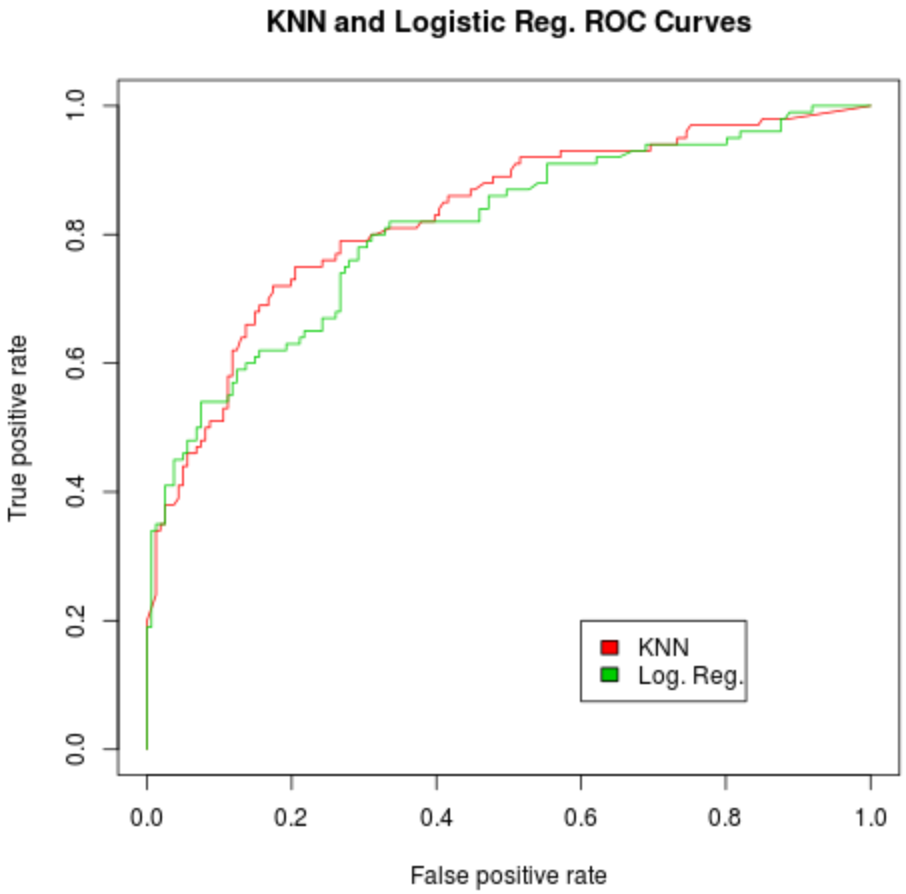
The scores for precision, recall, and accuracy resulting from application of the logistic regression model to the test set:

Precision	Recall	F-Measure	Accuracy
0.650000	**0.6372549**	**0.6435644**	**0.7241379**

After generating a ROC curve from the Logistic Regression classifier's performance on the test set, the area under the curve was computed to be 0.808757763975155

Conclusion

The following ROC plot illustrates a comparison of the relative performance of the two classifiers:



The tuned k-nearest neighbor algorithm outperforms logistic regression in recall, f-measure, accuracy, and auc score. The one category in which Logistic Regression outperforms K-nearest neighbors is the score for precision. Logistic regression's precision is 65% while K-nearest neighbors is only 62% demonstrating slightly better performance over k-nearest neighbors in predicting survivors out of the sum of true positives and false positives. On the other hand, K-nearest neighbors demonstrates 76.5% recall while Logistic Regression falls behind at 63.73%, suggesting that K-nearest neighbors correctly classifies survivors out of all possible correct classifications for survival more frequently. The relatively small difference in precision and wide gap in recall justifies K-nearest neighbor's greater f-measure of 68.5% over Logistic Regression is 64.33 % f-measure. K-nearest neighbors also predicts survival with a greater degree of accuracy, having 78% accuracy over logistic regression's 72% accuracy.

The final measure used as a basis for comparing the models was the area under the curve obtained from each model's ROC plot. The k-nearest neighbor model has an area of .8282

while the logistic regression model only has an area of .8088. This indicates better overall performance from the k-nearest neighbors model independent of class proportions.

It is clear that the tuned k-nearest neighbors algorithm demonstrates superior performance to logistic regression, making it the model of choice for the task of classifying survivors of titanic ship wreck.