Survival prediction with Titanic data set

Course: Big data analytics

Team members:

Shubham Mehta
Alvaro Taix
Sinem Ozeroglu
Vianney Lacroix
Astrid Junco Sommer

Cleaning

Among the available options for cleaning the dataset like Tableau, Excel and python, we used Excel for cleaning the training set and test set with no labels. To begin off, we created a consolidated list of training set and test set with no labels given that passenger id sorted at the end in ascending order will help us to separate the two lists again. The rationale for combining these two sets is that they are travelling in the same boat and should contribute to the overall calculation of average or median but are merely divided into batches for our analysis. Moving on, we analysed the values in all rows for different columns to check for the empty values, duplicate values and check outliers via maximum and minimum values obtained from these columns. To add to this, we would like to mention one important useful tool we obtained in bigml for analysis purpose. This is that you upload the consolidated dataset to bigml, create one-click dataset and when you click to view the dataset from this source, the entire analysis with regard to proportion and different categories within the dataset is presented by bigml. We realised that this is a balanced dataset with 38.3% people who survived and women passengers forming 35.5% of the cohort on board. So with this analysis, we obtained two missing values in embarkment, one missing value in fare; 263, 352, 1014 missing values out of 1309 in age, ticket and cabin respectively.

Once ascertained, we went ahead to fill in those empty values which can be reasonably filled with mean or median values like age and fare, given that they remain continuous variables. Further, we eliminated those columns/parameters which could not be reasonably filled as they were discrete variables like ticket and cabins and also those which would show no correlation with survival prediction like passenger id, name and ticket number. We created a new parameter out of name calling it title with entries like Mr., Mrs., Miss, Master and others. Then we went to look for the duplicate values and found none in names and a few under other parameters like ticket number, cabin number and fares but they were justified and this is self explanatory. Further we looked for any outliers in the dataset but had to rest with no findings in this category.

We further filled in the missing values in age, first by using mean and in the second attempt by using median. Also the male and female cohort were grouped separately for filling in the missing age values, as per the mean/median of their cohort. We made third and last attempt by further sub categorising males into people with titles as Mr. and Master because of clear distinction in this category. We couldn’t do the same for females as there was no clear distinction in usage for Mrs. and Miss and were used interchangeably across the entire age range.

Results of Machine Learning Models

So we made three different sets of models. One with mean values in age, second with median values and third with sub-categorisation of males between Mr. and Master. So working with first set, we picked only six parameters including pclass, sex, age, sibsp, parch and embarkment for our analysis based on the reasons described above, across three models of decision tree, random forest and Deep neural network(DNN). DNN showed highest accuracy at 77.9%, random forest at 75.5% and decision tree at 75.1%.

We decided to add one more parameter-prices, to see if rich people had higher probability of survival. So, with new results, the accuracy of DNN decreased by 0.15%, decision trees were also reduced by 1.9% but random forest showed an increase in accuracy by 0.3%. Now in the third and final iteration with titles like Mr., Mrs. et al. also included, DNN retained the same accuracy from the previous iteration at 77.75%, but random forest and decision tree both showed reduction in accuracy to the lowest levels amongst the three iterations done so far. So we obtained that DNN with initial six parameters provides the best accuracy at 77.9%. Its precision and recall with that data set is 75% and 62.6%.

Now we chose median values to fill the missing cells for ages and we picked the most accurate model of DNN to see if there is any change in accuracy. So the model contained six initial parameters along with titles but not fares (which had led to reduction of accuracy of DNN model). We found out that accuracy further improved for the DNN model by 0.5% to reach 78.4%. The precision and recall for the decision tree model is 61.9% and 68.9% respectively. So we saw increase in accuracy and recall but decrease in precision. The precision and recall for random forest and decision tree subsequently can be found in Annex1.

Now in the final iteration, we sub-categorised males into Mr. and Master as there were clear distinctions in the median and mean values of these two sets and filled in the missing values accordingly as per the median of the respective cohort. The accuracy, precision and recall for this model turned out to be as follows: For decision trees 73.2%, 63% and 70.2% respectively, for random forest 75.1%, 67.3% and 66.4% respectively and finally for the DNN model- 76.5%, 69.2% and 68.3% respectively. WIth greater sub-categorization within the model, we saw an increase in the accuracy of the random forest model.

Conclusion

The significant portion of our time was actually consumed in cleaning the data as it involved making decisions on eliminating, filling empty values and finding the correlations between parameters and survival of the passengers. For some parameters like fares' values, it consisted of accumulative fare for all the family members or accompanying acquaintances. This made the training of the dataset difficult. Also the training remains opaque since it is based on underlying algorithms in bigml and hence it was not known if the algorithm was able to make sense of categorical parameters like title, embarkment etc.

Nonetheless, DNN turned out to be the best model with highest accuracy at 78.4%. Since our training and test set had a mix of categorical and numerical values, so DNN was well positioned to solve such a complex problem of making batch predictions. Especially the back-propagation in Deep neural networks works to ensure that appropriate weights are assigned to each parameter. It works perfectly with high-dimensional data and high cardinality outcomes where there are a number of discrete categories available in the dataset. The only shortcoming is that we are not able to determine which parameter has been levied higher weightage or lower weightage. The best value is obtained when we use median values because outlier values don't affect the central tendency and filled in values present the right estimate.

Since our data was balanced, random forest is not the best model in such scenarios as it is best optimised for unbalanced data. But one good aspect of this is that it doesn't overfit. One can see in the Annex1 attached that when we made further sub-categorization with male category into Mr. and Master., the accuracy of the model improved. Besides, the longer the tree, the better is its accuracy. But the best accuracy with this model was 75.8%, way lower than one found with DNN.

In case of decision trees, the accuracy kept on decreasing as we added more parameters to our model in training and test sets. This is because the decision tree doesn't work well if lots of parameters are involved. Decision trees are typically not competitive amongst available supervised learning approaches in terms of prediction accuracy.

For our various models, precision and recall can be analysed. Precision and accuracy showed no direct - positive or negative correlation. So in order to maintain balance between precision and accuracy, F1 parameter can be used and its value was found to be the highest in the same model with highest accuracy value.

Annex1

	Decision Tree				Random Forest				Deep Neural Network			
	Acc	Pre	Rec	F1	Acc	Pre	Rec	F1	Acc	Pre	Rec	F1
1. Average with Six parameters	0.751	0.673	0.664	0.668	0.755	0.679	0.670	0.674	0.779	0.75	0.626	0.682
2. Six parameters+Fares	0.732	0.649	0.632	0.64	0.758	0.699	0.632	0.663	0.777	0.748	0.62	0.678
3. Six parameters+Fares+ Titles	0.717	0.626	0.626	0.626	0.746	0.675	0.632	0.652	0.777	0.712	0.689	0.7
4. Median +Titles(no Fares)	0.722	0.619	0.689	0.652	0.727	0.642	0.626	0.633	0.784	0.723	0.696	0.709
5. Median +Titles(no Fares) with male sub-categorization	0.732	0.63	0.702	0.664	0.751	0.673	0.664	0.668	0.765	0.692	0.683	0.687

- Acc-Accuracy; Pre-Precision; Rec-Recall