

Titanic dataset: Data cleaning and validation

Jesús Ros Solé

30 de diciembre, 2018

Contents

1. Dataset description	1
2. Data integration and selection	2
3. Data cleaning	3
3.1. Empty values	3
3.2. Extreme scores	4
3.3. Exploratory analysis	5
4. Data analysis	6
4.1. Objective	6
4.2. Normality and homogeneity of variance	7
4.3. Association with target variable	7
4.4. Model building	9
5. Conclusion	13
6. Resources	13

1. Dataset description

This dataset has been obtained from the Kaggle competition Titanic: Machine Learning from Disaster and contains a set of attributes for each of the passengers boarding the Titanic in the day of its accident.

It contains two datasets: a training dataset to build models with and a test dataset to perform predictions on. The training dataset has information on 891 passengers (rows) and 12 attributes (columns) while the test dataset has 418 passengers and 11 attributes (minus the target attribute **Survived**).

The attribute descriptions are:

- *PassengerID*: an integer identifier for each passenger.
- *Survived*: target class encoded as 1 (survived) or 0 (not survived). Missing in test set.
- *Pclass*: boarding class encoded as 1 (first class), 2 (second class) or 3 (third class).
- *Name*: name of the passenger as a string.
- *Sex*: sex of the passenger encoded as **male** or **female**.
- *Age*: age of the passenger.
- *SibSp*: number of siblings/spouse aboard.
- *Parch*: number of parent/child aboard.
- *Ticket*: identifier of the boarding pass.
- *Fare*: ticket fare amount.

- *Cabin*: identifier of the passenger cabin.
- *Embarked*: identifier for the port in which the passenger embarked encoded as C (Cherbourg), S (Southampton) and Q (Queenstown).

The objective of this work is study the groups of people that are more likely to survive according to the given attributes and to predict the survival chances of the people in the test set.

2. Data integration and selection

First we will load the training and test set using `read.csv` indicating the corresponding data types. We then extract the target attribute from the test set and merge the training and test datasets for the next steps.

```
train <- read.csv("../data/raw/train.csv",
  colClasses=c("integer", "factor", "factor", "character", "factor",
    "numeric", "integer", "integer", "character", "numeric",
    "character", "factor"),
  na.strings = c("NA", ""))
test <- read.csv("../data/raw/test.csv",
  colClasses=c("integer", "factor", "character", "factor", "numeric",
    "integer", "integer", "character", "numeric",
    "character", "factor"),
  na.strings = c("NA", ""))

y_train <- train["Survived"]
train$Survived <- NULL
test_id <- test["PassengerId"]
all <- rbind(train, test)
```

Now that we have our data properly loaded, we will select the attributes that are going to be useful for the posterior analysis. First note that the `PassengerID` and `Ticket` attributes will not yield much information to our analysis since they are mostly unique identifiers and can thus be ignored. The `Name` attribute by itself will not be very informative either, but we can extract the passenger title from it.

```
all$Title <- gsub("^.*, (.*)\\..*$", "\\1", all$Name)
kable(table(all$Title), format="pandoc", col.names=c("Title", "Freq"), digits=4)
```

Title	Freq
Capt	1
Col	4
Don	1
Dona	1
Dr	8
Jonkheer	1
Lady	1
Major	2
Master	61
Miss	260
Mlle	2
Mme	1
Mr	757
Mrs	197
Ms	2
Rev	8
Sir	1
the Countess	1

Title	Freq
-------	------

By looking at the obtained titles we can see that **Master**, **Miss**, **Mr** and **Mrs** are the most common while the others are quite rare. We will try to aggregate some of them to the most common ones (for instance **Ms** is a different spelling from **Miss**) and create an additional class for the rest.

```
all$Title[all$Title %in% c("Mlle", "Ms")] <- "Miss"
all$Title[!(all$Title %in% c('Master', 'Miss', 'Mr', 'Mrs'))] <- "Other"
kable(table(all$Title), format="pandoc", col.names=c("Title", "Freq"), digits=4)
```

Title	Freq
Master	61
Miss	264
Mr	757
Mrs	197
Other	30

```
all$Title <- as.factor(all$Title)
```

Additionally, we can create a new variable indicating the family size aboard the Titanic from the **SibSp** and **Parch** attributes.

```
all$FamilySize <- all$SibSp + all$Parch + 1 #includes self
```

Now we can drop the variables that yield no information **PassengerId**, **Name** and **Ticket**.

```
drop <- c("PassengerId", "Name", "Ticket")
all <- all[,!(names(all) %in% drop)]
```

3. Data cleaning

3.1. Empty values

At this point, we have selected the attributes that will be useful for our posterior analysis and have created a couple new derived attributes from our dataset. Now we will inspect the remaining attributes for empty values that need to be taken care of.

```
count_empty <- function(attr){sum(is.na(attr))}
kable(sapply(all, count_empty), format="pandoc", col.names=c("Empty"), digits=4)
```

	Empty
Pclass	0
Sex	0
Age	263
SibSp	0
Parch	0
Fare	1
Cabin	1014
Embarked	2
Title	0
FamilySize	0

We observed that the **Cabin** attribute contains mostly empty elements, in fact, about 77% of the rows contain missing data. Therefore, this attribute can be ignored since it will not yield much information. The other attributes that contain missing data are **Age**, **Fare** and **Embarked**. We will use kNN imputation with default settings after dropping the **Cabin** column.

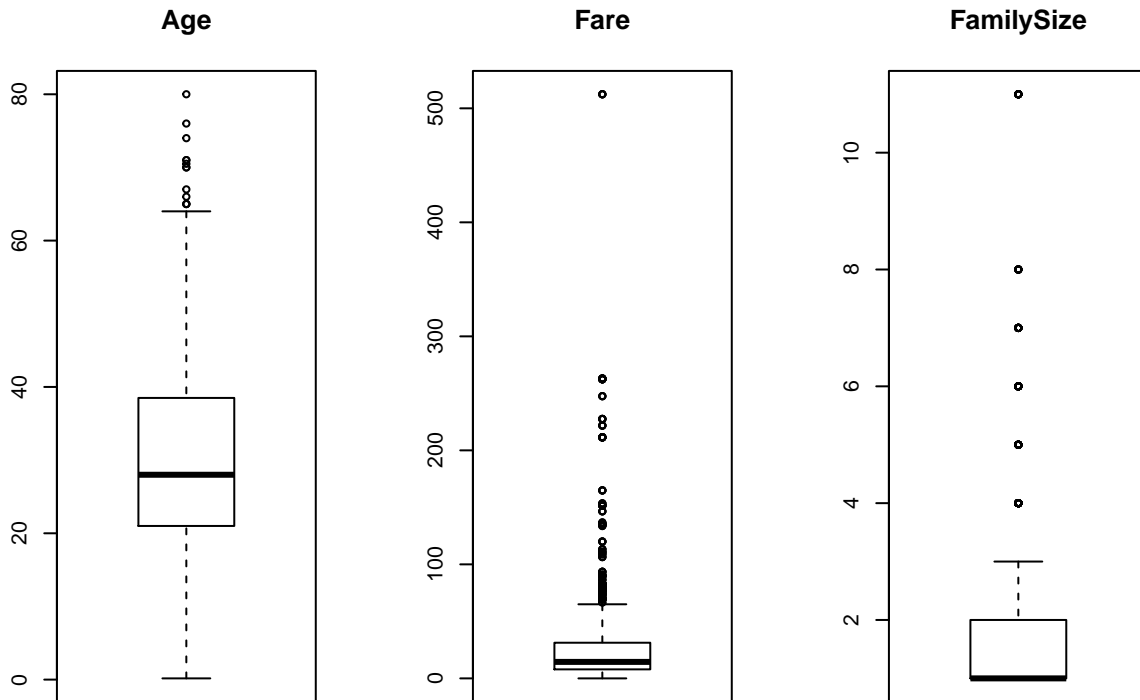
```
all$Cabin <- NULL
all_clean <- suppressWarnings(kNN(all, imp_var=FALSE))
kable(sapply(all_clean, count_empty), format="pandoc",
       col.names=c("Empty"), digits=4)
```

	Empty
Pclass	0
Sex	0
Age	0
SibSp	0
Parch	0
Fare	0
Embarked	0
Title	0
FamilySize	0

3.2. Extreme scores

Next we will investigate the extreme values in the dataset for numeric attributes. Numeric attributes are **Age**, **SibSp**, **Parch**, **Fare** and **FamilySize**. Since **FamilySize** is derived from **SibSp** and **Parch** we will only investigate **Age**, **Fare** and **FamilySize** using a boxplot.

```
par(mfrow=c(1, 3))
boxplot(all_clean$Age, main="Age")
boxplot(all_clean$Fare, main="Fare")
boxplot(all_clean$FamilySize, main="FamilySize")
```



We observe some outliers in all 3 variables but their values seem to be realistic. We have ages up to 80 years, fares up to 500 dollars and families of up to 11 members. Therefore we decide to leave the outliers as is.

Finally, we have a clean dataset for further analysis. We will export the clean dataset to a datafile, respecting the initial training and test partition and adding the target attribute to the training file.

```
train <- cbind(all_clean[1:nrow(train),], y_train)
test <- all_clean[(nrow(train)+1):nrow(all_clean),]
write.csv(train, "../data/clean/train.csv", row.names=F)
write.csv(test, "../data/clean/test.csv", row.names=F)
```

3.3. Exploratory analysis

Finally, we will look at some descriptive statistics of the variables in our dataset.

For the numerical variables, we will look at both central tendency (Mean, Median, 95% Trimmed mean and 95% Winsorized Mean) and dispersion measures (Standard deviation, Interquartile Range and Median Absolute Deviation).

```
c_summary <- function(x, na.rm=TRUE, trim=0.05){
  c(Mean=mean(x,na.rm=na.rm),
    Median=median(x,na.rm=na.rm),
    Trimmed_Mean=mean(x,trim=trim,na.rm=na.rm),
    Winsorized_Mean=winsor.mean(x,trim=trim,na.rm=na.rm),
    Std=sd(x,na.rm=na.rm),
    IQR=IQR(x,na.rm=na.rm),
    MAD=mad(x,na.rm=na.rm))
}
```

```
}

supply(all_clean[c("Age", "SibSp", "Parch", "Fare", "FamilySize")], c_summary)
```

```
##           Age      SibSp      Parch      Fare FamilySize
## Mean      29.86606 0.4988541 0.3850267 33.29473   1.883881
## Median    28.00000 0.0000000 0.0000000 14.45420   1.000000
## Trimmed_Mean 29.61747 0.3282443 0.2663274 24.73775   1.636132
## Winsorized_Mean 29.70512 0.3949580 0.3391902 29.27630   1.771581
## Std       13.70605 1.0416584 0.8655603 51.73889   1.583639
## IQR       17.50000 1.0000000 0.0000000 23.37920   1.000000
## MAD       11.86080 0.0000000 0.0000000 10.23617   0.000000
```

- *Age*: the average age is 29.87 with a standard deviation of 13.71. We observed that the trimmed and winsorized means are close to the mean, meaning there are not many outliers.
- *SibSp*, *Parch* and *FamilySize*: we observe that most of the passengers travel alone since the medians are 0, 0 and 1, respectively.
- *Fare*: the average fare is 33.29 dollars with a standard deviation of 51.74. This variable seems to be largely dispersed and with some values deviating significantly from the average, as noted by the differences between the mean and the trimmed and winsorized means.

For the categorical variables, we will show their distribution along the categories of each variable.

```
summary(all_clean[c("Pclass", "Sex", "Embarked", "Title")])
```

```
## Pclass      Sex      Embarked      Title
## 1:323   female:466   C:270   Master: 61
## 2:277   male :843   Q:123   Miss  :264
## 3:709                      S:916   Mr    :757
##                               Mrs   :197
##                               Other : 30
```

- *Pclass*: around 25% of the passengers are from first class, 21% of second class and the majority (54%) are of third class.
- *Sex*: 36% of the passengers are females while 64% are males.
- *Embarked*: 21% of the passengers embarked from Cherbourg, 9% from Queenstown and the majority (70%) from Southampton.
- *Title*: 5% of the passengers have the title Master, 20% Miss, 58% Mr, 15% Mrs and the rest (2%) have another title.

4. Data analysis

4.1. Objective

The objective of the analysis is two-fold: first we will investigate associations between the variables with the target attribute **Survived**. This analysis will help us decide which models to build afterwards.

Since we have both numerical and categorical attributes and the target attribute is categorical, we will use the corresponding p-value from a One-Way ANOVA test to test significance between continuous variables and the target attribute and the p-value from a chi-squared test to test significance between categorical variables and the target attribute.

The null hypothesis for the One-Way ANOVA test is that all group produce the same variation on the response, on average. The null hypothesis for the chi-squared test is that the two variables are independent.

4.2. Normality and homogeneity of variance

First of all, we will test the numerical attributes “Age” and “Fare” for normality using the Shapiro-Wilk test and the homogeneity of the variance using Fligner-Killeen Test.

```
numeric_attr <- c("Age", "Fare")
#Shapiro-Wilk test
shapiro.wilk <- sapply(train[numeric_attr], function(x){shapiro.test(x)$p.value})
#Fligner-Killeen Test for homogeneity of variance
homogeneity_variance <- function(attr_name, dataset, target_name){
  fligner.test(as.formula(paste(attr_name, target_name, sep="~")),
               data=dataset)$p.value
}
fligner <- sapply(numeric_attr, homogeneity_variance, dataset=train,
                 target_name="Survived")
kable(rbind(shapiro.wilk, fligner), format="pandoc", digits=4)
```

	Age	Fare
shapiro.wilk	0.0000	0
fligner	0.1914	0

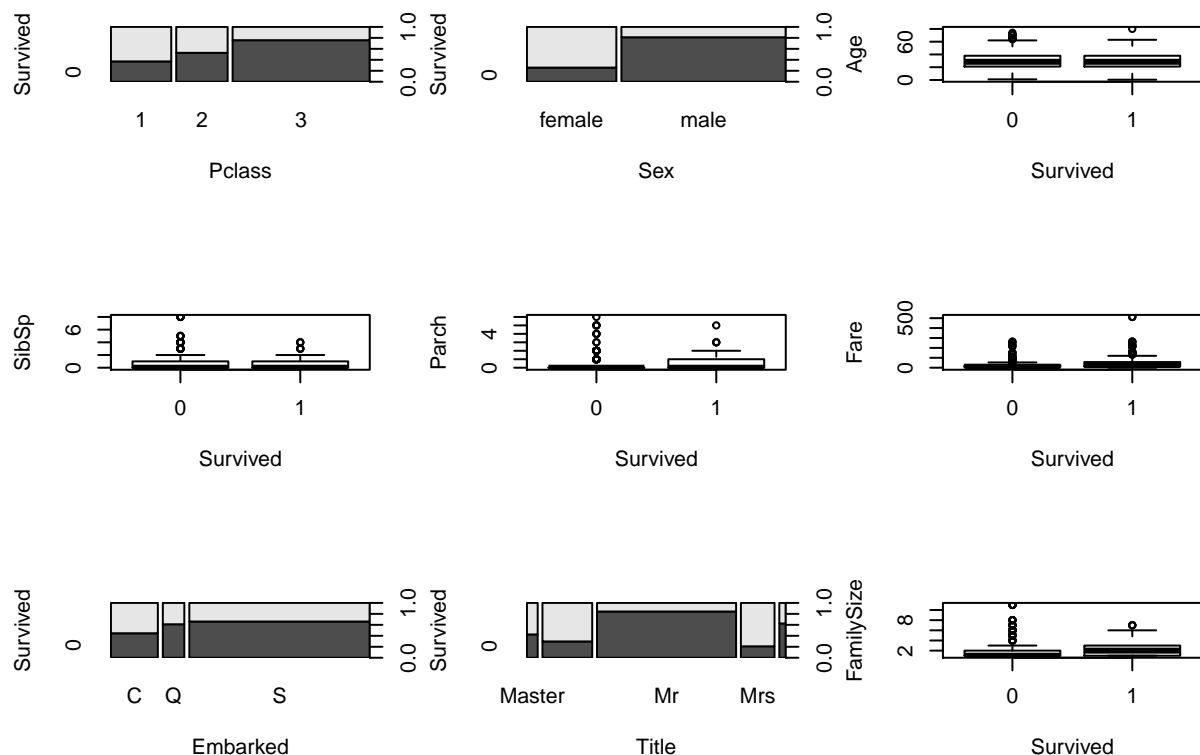
We can see that none of the variables appear to be normally distributed and that the attribute **Fare** does not have equal variances when grouped by the target value.

4.3. Association with target variable

Now we can finally explore the association between variables, specifically which variables are associated with the target attribute **Survived**. This will hint on which variables can be more informative during model building.

First we will grafically display the association between the variables and the target variable.

```
par(mfrow=c(3,3))
plot(train$Pclass, train$Survived, xlab="Pclass", ylab="Survived")
plot(train$Sex, train$Survived, xlab="Sex", ylab="Survived")
boxplot(Age~Survived, data=train, xlab="Survived", ylab="Age")
boxplot(SibSp~Survived, data=train, xlab="Survived", ylab="SibSp")
boxplot(Parch~Survived, data=train, xlab="Survived", ylab="Parch")
boxplot(Fare~Survived, data=train, xlab="Survived", ylab="Fare")
plot(train$Embarked, train$Survived, xlab="Embarked", ylab="Survived")
plot(train$Title, train$Survived, xlab="Title", ylab="Survived")
boxplot(FamilySize~Survived, data=train, xlab="Survived", ylab="FamilySize")
```



Some of the insights that we can get from these plots are:

- The higher the class the better the changes of survival.
- Women survived more than men.
- Medium size families seem to have better survival chances than passengers who travel alone. The largest families seem to have a lower chance of survival.
- People who paid higher fares have more chances of survival (probably correlated with higher class).
- Passengers who embarked from Cherbourg seem to have better chances of survival.
- Passengers with a title of Master have a higher chance of survival than men with a title of Mr.

Since some of the groups have unequal variances we will be using the Welch ANOVA test that does not assume equal variances between groups. Additionally, we can use this parametric test because we have large sample sizes, even though we have seen they are not normally distributed. An alternative would be to use a non-parametric test such as the Kruskal-Wallis test.

To account for the effect sizes of the associations we are going to use the Intraclass correlation (continuous variables) and Cramer's V (categorical variables). The first ranges between -1 (perfect anticorrelation) to 1 (perfect correlation) having absence of correlation a value of 0, while the latter ranges from 0 (no correlation) to 1 (perfect correlation).

```
cramer.v <- function(x, y){
  as.numeric(sqrt(chisq.test(x, y)$statistic /
    (length(x) * (min(length(unique(x)), length(unique(y))) - 1))))
}

compute_corr <- function(attr_name, dataset, target_name){
  if(is.numeric(dataset[[attr_name]])){
    pvalue <- oneway.test(as.formula(paste(attr_name, target_name,
```



```

                                sep=~"))),
                                data=dataset)$p.value
    effect_size <- ICCest(target_name, attr_name, data=dataset)$ICC
  } else { #it is factor
    pvalue <- chisq.test(dataset[[attr_name]], dataset[[target_name]])$p.value
    effect_size <- cramer.v(dataset[[attr_name]], dataset[[target_name]])
  }
  list(pvalue=round(pvalue,digits=4), effect_size=round(effect_size,digits=4))
}

kable(t(sapply(names(train[-ncol(train)]), compute_corr, dataset=train,
              target_name="Survived")), format="pandoc")

```

	pvalue	effect_size
Pclass	0	0.3398
Sex	0	0.5409
Age	0.1191	0.0036
SibSp	0.2327	3e-04
Parch	0.0134	0.0116
Fare	0	0.1283
Embarked	0	0.1707
Title	0	0.5671
FamilySize	0.5853	-0.0018

We can see that for all variables except `Age`, `SibSp` and `FamilySize` the results are significant at a 95% confidence level, that is those variables are associated with the target attribute `Survived` although the strength of such association is not very strong, as shown in the `effect_size` column, where the highest values are slightly higher than 0.5. Therefore, there is not a very strong association between those variables and the target variable.

4.4. Model building

Now that we understand the association between the attributes and the response variable, we will propose several models to predict the target class. Since we are interested in models that output the probability of survival, we will use logistic regression models with different attributes as models.

First we will inspect a model with all variables that turned out significant in the previous analysis. That is, a model that related `Survived` with `Pclass`, `Sex`, `Parch`, `Fare`, `Embarked` and `Title`.

```

m1 <- glm(Survived ~ Pclass + Sex + Parch + Fare + Embarked + Title,
          family=binomial(link='logit'), data=train)
summary(m1)

```

```

##
## Call:
## glm(formula = Survived ~ Pclass + Sex + Parch + Fare + Embarked +
##      Title, family = binomial(link = "logit"), data = train)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.4623  -0.6341  -0.3660   0.6170   2.3451
##
## Coefficients:

```

```
##           Estimate Std. Error z value Pr(>|z|)
## (Intercept) 18.257723 437.522408  0.042 0.966714
## Pclass2     -0.835196  0.297943 -2.803 0.005060 **
## Pclass3     -1.993777  0.286341 -6.963 3.33e-12 ***
## Sexmale     -15.257644 437.522134 -0.035 0.972181
## Parch       -0.461983  0.129339 -3.572 0.000354 ***
## Fare         0.001793  0.002334  0.768 0.442382
## EmbarkedQ    -0.269478  0.384326 -0.701 0.483197
## EmbarkedS    -0.603872  0.241278 -2.503 0.012321 *
## TitleMiss   -15.327244 437.522307 -0.035 0.972054
## TitleMr      -3.086177  0.405979 -7.602 2.92e-14 ***
## TitleMrs    -14.844836 437.522339 -0.034 0.972934
## TitleOther   -3.460938  0.679286 -5.095 3.49e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 1186.66 on 890 degrees of freedom
## Residual deviance: 755.39 on 879 degrees of freedom
## AIC: 779.39
##
## Number of Fisher Scoring iterations: 13
```

From the coefficients of the obtained model, we see that the coefficients for the attributes **Fare**, **Sex** and some of the categories in **Title** and **Embarked** are not significant at a 95% confidence level. We will compare this model with another without these variables.

```
m2 <- glm(Survived ~ Pclass + Parch + Embarked + Title,
          family=binomial(link='logit'), data=train)
summary(m2)
```

```
##
## Call:
## glm(formula = Survived ~ Pclass + Parch + Embarked + Title, family = binomial(link = "logit"),
##      data = train)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.5125  -0.6270  -0.3639   0.6235   2.3442
##
## Coefficients:
##           Estimate Std. Error z value Pr(>|z|)
## (Intercept)  3.18197    0.47073   6.760 1.38e-11 ***
## Pclass2     -0.96862    0.27106  -3.573 0.000352 ***
## Pclass3     -2.12311    0.24648  -8.614 < 2e-16 ***
## Parch       -0.44173    0.12477  -3.540 0.000399 ***
## EmbarkedQ    -0.31693    0.38215  -0.829 0.406920
## EmbarkedS    -0.63296    0.23661  -2.675 0.007469 **
## TitleMiss   -0.06907    0.39210  -0.176 0.860174
## TitleMr      -3.10724    0.40614  -7.651 2.00e-14 ***
## TitleMrs     0.40068    0.42385   0.945 0.344483
## TitleOther   -2.97072    0.60704  -4.894 9.89e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 1186.66  on 890  degrees of freedom
## Residual deviance:  763.22  on 881  degrees of freedom
## AIC: 783.22
##
## Number of Fisher Scoring iterations: 5
```

Comparing both models we can see that the second model has most of its attributes significant, except the same categories in `Title` and `Embarked` attributes. Comparing the AIC values for both models suggest that the first model is slightly better since it has a lower value of AIC.

AIC stands for Akaike Information Criterion and is a measure of the quality of a given model. This estimator balances the goodness of fit of the model with its complexity. More complex models tend to have a higher capacity of better explaining the data at the cost of overfitting it. Therefore, this estimator introduces a penalisation for higher complexity.

Since we have obtained a lower AIC value for our more complex model (it introduces more regressors), it means that this additional complexity is compensated by a better goodness of fit.

Finally, we will try a third model from the first one just dropping the `Fare` attribute. This corresponds to the idea that in a disaster such as the Titanic, the rescue teams would prioritize women and children and, therefore, `Age` might be important even though its coefficient turned out not significant.

```
m3 <- glm(Survived ~ Pclass + Sex + Parch + Embarked + Title,
          family=binomial(link='logit'), data=train)
summary(m3)
```

```
##
## Call:
## glm(formula = Survived ~ Pclass + Sex + Parch + Embarked + Title,
##      family = binomial(link = "logit"), data = train)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.5013  -0.6340  -0.3640   0.6155   2.3441
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  18.4133    437.2343   0.042 0.966409
## Pclass2      -0.9270     0.2730  -3.396 0.000685 ***
## Pclass3     -2.1058     0.2468  -8.533 < 2e-16 ***
## Sexmale     -15.2634    437.2341  -0.035 0.972152
## Parch       -0.4368     0.1246  -3.505 0.000457 ***
## EmbarkedQ   -0.2901     0.3836  -0.756 0.449450
## EmbarkedS   -0.6313     0.2385  -2.647 0.008115 **
## TitleMiss   -15.3298    437.2342  -0.035 0.972031
## TitleMr     -3.0938     0.4061  -7.619 2.55e-14 ***
## TitleMrs    -14.8616    437.2343  -0.034 0.972885
## TitleOther  -3.5014     0.6786  -5.160 2.47e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 1186.7  on 890  degrees of freedom
```

```
## Residual deviance: 756.0 on 880 degrees of freedom
## AIC: 778
##
## Number of Fisher Scoring iterations: 13
```

Indeed it turns out to be a slightly better model according to the AIC values obtained. We will therefore select the third model to make the final prediction on the test set. Before that, we will inspect the significant coefficients of the model to gain some insight on how they affect the response variable.

- Being of a lower class reduces the chances of survival so passengers of class 1 are expected to survive more than passengers of class 2 and even more than passengers of class 3.
- Having more parent/children reduces the chances of survival. Since this attribute will be at most 2 for children (they can at most have their 2 parent aboard) while parents of large families will have larger values of this attribute, it is expected that parents of large families will have less chances of survival.
- Passengers who embarked from Southampton also have lower chances of survival. It could be explored if those passengers have different demographics as others.
- Having a title of Mr. or Other. reduces chances of survival. This is in line with the fact the women survived more than man. As for the Other class, there were a few passengers with this class and it might be interesting to explore if they also had different demographics as other passengers.

To get an estimate of the performance of this model on unseen data we will use a 5-fold cross validation strategy on the training dataset. We will output the resulting accuracy score obtained as an average of the 5-fold scores along with its standard deviation.

```
train_control <- trainControl(method = "cv", number = 5)
performance <- train(Survived ~ Pclass + Sex + Parch + Embarked + Title,
  data = train,
  trControl = train_control,
  method = "glm",
  family=binomial(link='logit'))
sprintf("5-fold accuracy score: %.4f +/- %.4f",
  performance$results$Accuracy, performance$results$AccuracySD)
```

```
## [1] "5-fold accuracy score: 0.8103 +/- 0.0233"
```

Finally, we will use the third model to make predictions on the test set.

```
y_pred <- predict(m3, test, type="response")
kable(head(cbind(test, y_pred)), format="pandoc", digits=4)
```

	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked	Title	FamilySize	y_pred
892	3	male	34.5	0	0	7.8292	Q	Mr	1	0.0879
893	3	female	47.0	1	0	7.0000	S	Mrs	2	0.6931
894	2	male	62.0	0	0	9.6875	Q	Mr	1	0.2385
895	3	male	27.0	0	0	8.6625	S	Mr	1	0.0641
896	3	female	22.0	1	1	12.2875	S	Mrs	3	0.5933
897	3	male	14.0	0	0	9.2250	S	Mr	1	0.0641

From the predictions we can see that passenger 892 has a predicted probability of survival of less than 9% while passenger 893 has almost 70% chances of survival. To determine the predicted class we will use a probability of 50% as a threshold to determine which passengers we think survived. We can finally make a prediction of the test set classes.

```
y_test <- as.factor(ifelse(y_pred > 0.5, 1, 0))
submission <- data.frame(PassengerId = test_id, Survived=y_test)
write.csv(submission, "../data/submission/submission.csv", row.names=F, quote=F)
```

5. Conclusion

In this report we have explored the association of the different attributes with the survival chance of passengers of the Titanic using the dataset from the Kaggle competition Titanic: Machine Learning from Disaster which has helped in building logistic regression models to predict the class for unseen data.

Our analysis determine that all variables except **Age**, **SibSp** and **FamilySize** are significantly associated with **Survived** at a 95% confidence level although the strength of such association is not very strong. Furthermore, the model that uses the attributes **Pclass**, **Sex**, **Parch**, **Embarked** and **Title** as regressors has been the chosen model to predict with according to resulting AIC values amongst the tested models.

This model allows us to explore the relationships between regressors and the target attribute as well as predict probability of survival for passengers. In particular we have determined that lower classes, large families, passengers with a title of *Mr* or *Other* and passengers who embarked at Southampton have a significantly lower chance of survival, according to the coefficients from the regression model that turned out to be significant.

Finally, we have computed an estimate of the performance of this model using 5-fold cross validation on the training dataset and we have used this model to label the instances from the test dataset with their predicted class values. Our predicted accuracy from the 5-fold CV is around 80%.

6. Resources

1. Peter Dalgaard (2008). Introductory statistics with R. Springer Science & Business Media.
2. Jiawei Han, Micheline Kamber, Jian Pei (2012). Data mining: concepts and techniques. Morgan Kaufmann. Chapter 3.
3. Jason W. Osborne (2010). Data Cleaning Basics: Best Practices in Dealing with Extreme Scores. Newborn and Infant Nursing Reviews; 10 (1): pp. 1527-3369.
4. R Documentation. Shapiro-Wilk Normality Test. <http://finzi.psych.upenn.edu/R/library/stats/html/shapiro.test.html>.
5. R Documentation. Fligner-Killeen Test of Homogeneity of Variances. <http://finzi.psych.upenn.edu/R/library/stats/html/fligner.test.html>
6. R Documentation. Test for Equal Means in a One-Way Layout. <http://finzi.psych.upenn.edu/R/library/stats/html/oneway.test.html>
7. R Documentation. Pearson's Chi-squared Test for Count Data. <http://finzi.psych.upenn.edu/R/library/stats/html/chisq.test.html>
8. R Documentation. ICC: Facilitating Estimation of the Intraclass Correlation Coefficient. <https://cran.r-project.org/web/packages/ICC/index.html>