Logistic Regression Model to Predict Probability of House Sparrow Survival

STA 207 Final Project, Winter 2017

Pamela Patterson

Zamirbek Akimbekov

Contents

Data and Project Description

After a severe winter storm in Providence, RI in 1898 left 136 house sparrows on the ground, an ecologist, Hermon Bumpus, recorded various characteristics of the sparrows with the goal of investigating the probability of survival associated with the physical characteristics. The data that was collected is Survival (STATUS) (survived or perished), Age (AG) (adult=1, juvenile=2), Total length (TL), Alar extent (AE), Weight (WT), Length of beak and head (BH), Length of humerus (HL), Length of femur (FL), Length of tibio-tarsus (TT), Width of skull (SK) and Length of keel of sternum (KL).

Data exploration

All of the variables are numeric, except for the "status" variable, which is a "character". Since it is binary, we can leave it as a character, and tell R to treat it as a factor. We also tell R to treat age as a factor. After finding that there are no missing values, we look at the variables. We can see in Figure 1 that there are no obviously skewed distributions among the variables, so we don't need to pursue any transformations of the variables. Looking at the correlation matrix of the non-factor variables (Figure 2), we can see that all of the variables are positively correlated, and that the smallest correlation coefficient is 0.31, so there is indication of multicollinearity. The pairwise scatterplot matrix (Figure 2) confirms multicollinearity. Looking at the pie charts of the two factor variables (Figure 3), we can see that the majority of the birds were adults and the majority survived. However, the ratio of survived to perished is almost ½, so there is a significant number of birds that perished. We now want to see if there are any interaction terms we should include. First, we notice that there are three interaction terms that appear to have the strongest linear relationship on the pairwise scatterplot matrix. These are HL:FL, HL:TT, and FL:TT. Let's do a log likelihood test comparing the full model that includes all terms plus these three interaction terms with the reduced model that drops the three interaction terms. We get G^2 =6.42592 (df=11). Performing a test at level $\alpha = 0.05$, we have:

Full: $\pi = \beta_0 + \beta_1 \mathrm{AG} + \beta_2 \mathrm{TL} + \beta_3 \mathrm{AE} + \beta_4 \mathrm{WT} + \ldots + \beta_{10} \mathrm{KL} + \beta_{11} \mathrm{HL} * \mathrm{FL} + \beta_{12} \mathrm{HL} * \mathrm{TT} + \beta_{13} \mathrm{FL} * \mathrm{TT}$

Reduced: $\pi = \beta_0 + \beta_1 \mathrm{AG} + \beta_2 \mathrm{TL} + \beta_3 \mathrm{AE} + \beta_4 \mathrm{WT} + \ldots + \beta_{10} \mathrm{KL}$

$$\mathbf{H_0}: \beta_{11} = \beta_{12} = \beta_{13} = 0$$

$$\mathbf{H_a}: \text{not all } \beta_k = 0, \qquad k = 11, 12, 13$$

$$\chi^2(0.95;\ 3) = 7.81$$

Since $G^2 \leq 7.81$, we conclude H_0 and therefore don't include these interaction terms in the model. Since these interaction terms appeared to have the strongest linear relationship, we won't include any interaction terms in the model. There doesn't appear to be any nonlinearity in any of the terms, so we won't include nonlinear terms in the model. After splitting the data into training and testing sets, we can check that the two sets have similar distributions for each variable. Looking at Figure 4, we can see that the training and validation sets are similarly distributed.

Now we are ready for model selection. We will start with a full model that includes all terms and no interaction terms, and perform forward and backward selection, AIC, and SBC to determine the best model. We then do diagnostics using the testing data to evaluate the effectiveness of our chosen model.

Model Selection

The summary of statistics for the logistic regression coefficients of the full model, only TL is statistically significant at $\alpha = 0.05$ out of 11 features. However, due to high correlations among predictor variables, it should be expected. Hence, we performed four different model selection methods one by one to identify the potential final model.

First, we run backward elimination to decide which predictor variables can be dropped from the full logistic regression model controlling the risk using F-test (Table 2). At the end, this method keeps five, TL, WT, FL, SK, and KL, variables out of initial 11. Here is the selected model (model 2 from now on):

$$\pi \text{ (Survival)} = 59.2388 - 0.9341 * \mathrm{TL} - 0.5493 * \mathrm{WT} + 64.4373 * \mathrm{FL} + 52.6669 * \mathrm{SK} + 32.7469 * \mathrm{KL}$$

Having determined backward selection model, we conducted forward selection procedure using F – test at each step during the decision of keeping variables or not

(Table 3). Fortunately, the forward selection resulted in picking the model 2 as its champion. This gave us more confidence in the appropriateness of the model 2.

In the next step, we found the best model according to AIC criterion (Table 4). Again, this best model selection method selected model 2. Finally, we used SBC criterion to identify the best model (Table 5). However, the SBC criterion chose model (model 3 from now on):

$$\pi \text{ (Survival)} = 76.0955 - 0.8195 * \text{TL} + 78.3934 * \text{FL}$$

The fact that SBC criterion chose a model with less number of predictors could be expected since SBC criterion penalizes model more heavily. Hence, it is not surprising that model 3 has only two features whereas model 2 has five predictor variables (Table 6 and 7). To decide whether the other 3 features could be dropped from model 2 and match model 3, we performed Log-Likelihood ratio test at $\alpha = 0.05$:

$$\textbf{Full: } \pi = \beta_0 + \beta_1 \text{TL} + \beta_2 \text{WT} + \beta_3 \text{FL} + \beta_4 \text{SK} + \beta_5 \text{KL}$$

$$\textbf{Reduced: } \pi = \beta_0 + \beta_1 \text{TL} + \beta_2 \text{FL}$$

$$\mathbf{H_0}: \ \beta_2 = \beta_4 = \beta_5 = 0$$

$$\mathbf{H_a}: \ \text{not all } \beta_k = 0, \qquad k = 2, 4, 5$$

$$\chi^2(0.95;\ 3) = 7.81$$

$G^2(.95, 3) = 9.88$ and it is greater than $\chi^2(0.95;\ 3)$. Hence, we reject H_0 in favor of H_a at $\alpha = 0.05$ (Table 8). This means that WT, SK, and KL variables should be dropped. Therefore, we chose the model 3 as our final model, Table 7. The WT, SK, and KL are highly correlated with TL and FL, and therefore probably carry the same information.

Model Diagnostics

Since our goal is to build a predictive model for the probability of house sparrow survival, we started out diagnosis by analyzing the predictive ability of the model 3.

The difficulty in making predictions of a binary outcome is in determining the cutoff point, below which the outcome 0 (Perished) is predicted and above the outcome 1 (Survived) is predicted. Using the general 0.5 as the cutoff gave us about 72 %

accuracy on the validation set. To improve the model 3 performance on labeling the outcomes correctly as “Perished” or “Survived”, we calculated accuracies for threshold values ranging from 0 to 1 by 0.05 increment. On Figure 5, one can see that the highest accuracy for the prediction of outcomes of validation set occurs between 0.65-0.70. Hence, with this approach, the prediction rule is:

If π (Survival) exceeds 0.67, predict “Survived”; otherwise predict “Perished”.

With this prediction rule, the accuracy of our model increased to 77.3 % which is quite decent with the amount of data we have to train the logistic regression model 3.

We also calculated the true positive rate (a.k.a sensitivity, $P(\hat{\pi}\ (\text{Survival}) = 1 \mid \pi\ (\text{Survival}) = 1)$) and false negative rate (a.k.a specificity, $1 - P(\hat{\pi}\ (\text{Survival}) = 0 \mid \pi\ (\text{Survival}) = 0)$). True positive rate is 0.6667 whereas false negative rate is 0.1

Moreover, we plotted the receiver operating characteristic (ROC) curve, Figure 6. The area under the ROC curve is 0.7863 which again shows that our model has quite strong predictive power.

Once we were confident that the model 3 has satisfying predictive ability, we performed diagnostics to see how much the model 3 deviates from the assumptions. Figure 7 – 9, shows that assumption regarding normality, linearity, and constant variance are not highly violated and therefore we may accept model 3 to be well designed.

Discussion

Based on previous sections, we chose model 3:

$$\pi\ (\text{Survival}) = 76.0955 - 0.8195 * TL + 78.3934 * FL$$

Does including these house sparrow features to predict a sparrow’s survival make sense? First, numerous research articles indicate that a house sparrow’s survival is significantly related to its size. Thus, Pugesek et al. who studied the Bumpus house sparrow data using structural equations reported that survival increased significantly with increasing the general size and was unrelated to leg size and head size.[1-3] As most of the variables from the initial pool are measurements of head and leg size, their exclusion in our final model is warranted. Hence, model we selected includes TL (total length, a unit to measure the size of a sparrow) feature with p-value as 0.000257.

Second, the length of femur, a bone that makes a sparrow walk or jump, could provide strengths to the general body and might be considered as important for a house sparrow survival during a storm. Therefore, it is also added to the final model with p-value as 0.001074. Also, these two variables might be considered as generalization of the total size of sparrow and bone effect on the survival of house sparrow during storm. Finally, checking for the multi-collinearity effects shows that the final model 3 does not suffer from it with all VIF values being less than 10, Table 9. Hence, once again, the final model is adequate, Table 7 and 10.

Conclusion

In this project, we developed a classification model for survival of house sparrow after winter storm using the Hermun Bumpus data set and logistic regression. Our final selected model is:

$$\pi \text{ (Survival)} = 76.0955 - 0.8195 * TL + 78.3934 * FL$$

It includes the total length (TL) and femur length of house sparrows as significant variables in their survival during the winter storm. All three coefficients (intercept, TL, and FL) are statistically significant at $\alpha = 0.05$ with p-values being considerably smaller than 0.005. For classification, the model uses 0.67 as a cutoff above which outcome is predicted as “Survived”, else otherwise. The classification power of the model is strong with 77.3 % accuracy, 0.6667 true positive (specificity) and 0.1 false negative rates, and the area under the ROC curve is 0.7863. All in all, the model could be used to predict whether a house sparrow would survive the winter storm or not with just two variables, total length of the sparrow body and femur length. However, the accuracy of the model could be improved using larger sample sizes to train the logistic regression model.

References

1. Pugesek, B. H. et al. *Evolutionary Ecology* **1996**, 10, 387 – 404
2. Buttemer, W. A. *The Condor* **1992**, 94, 944 – 954
3. Holand, H. et al. *Journal of Avian Biology* **2014**, 45, 001 - 009

Appendices

A1. Figures

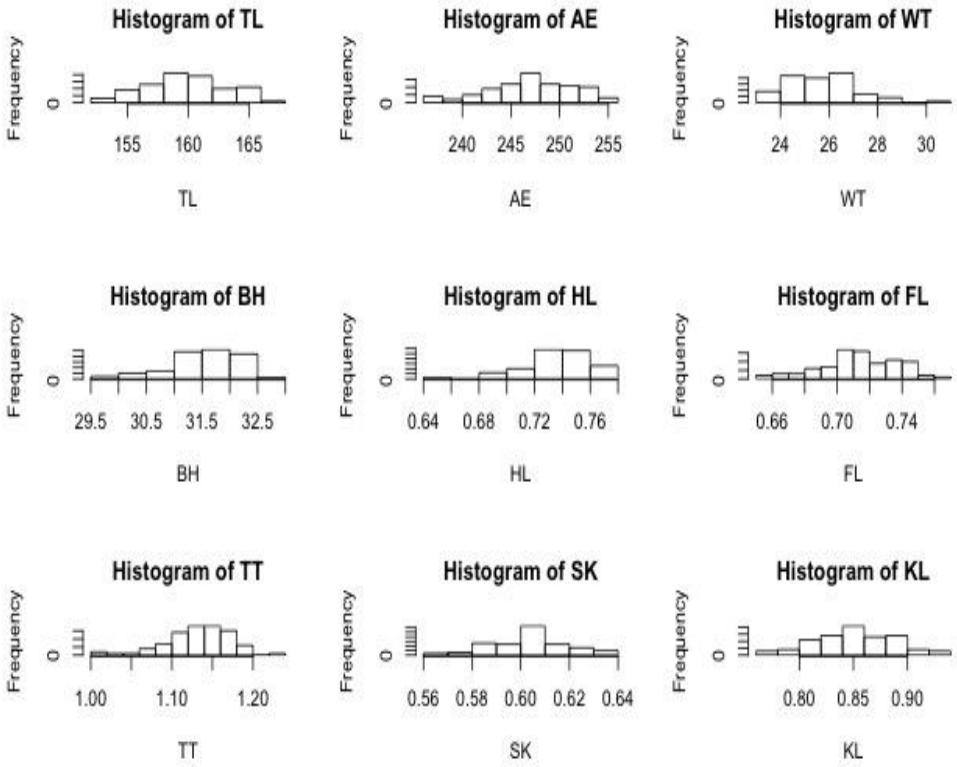
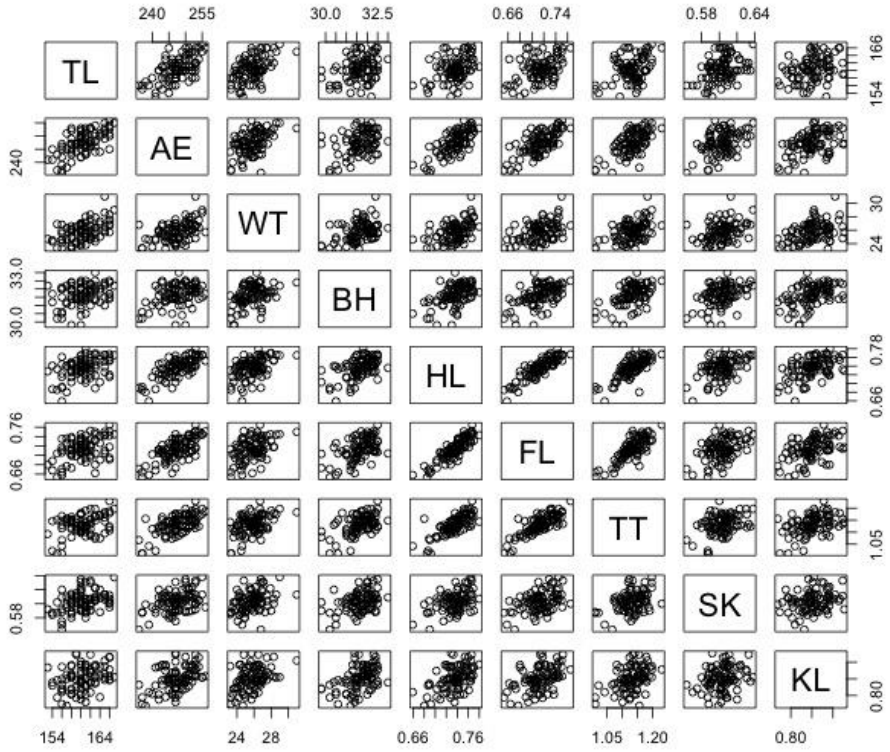


Figure 1. Histogram for the distribution of the quantitative variables.



	TL	AE	WT	BH	HL	FL	TT	SK	KL
TL	1	0.62	0.53	0.30	0.40	0.44	0.38	0.39	0.31
AE	0.62	1	0.5	0.37	0.74	0.71	0.64	0.43	0.46
WT	0.53	0.5	1	0.42	0.48	0.44	0.47	0.36	0.40
BH	0.30	0.37	0.42	1	0.52	0.53	0.53	0.40	0.45
HL	0.40	0.74	0.48	0.52	1	0.88	0.77	0.47	0.49
FL	0.44	0.71	0.44	0.53	0.88	1	0.81	0.45	0.47
TT	0.38	0.64	0.47	0.53	0.77	0.81	1	0.39	0.43
SK	0.39	0.43	0.36	0.40	0.47	0.45	0.39	1	0.26
KL	0.31	0.46	0.40	0.45	0.49	0.47	0.43	0.26	1

Figure 2. Correlation matrix and correlation values for the quantitative variables.

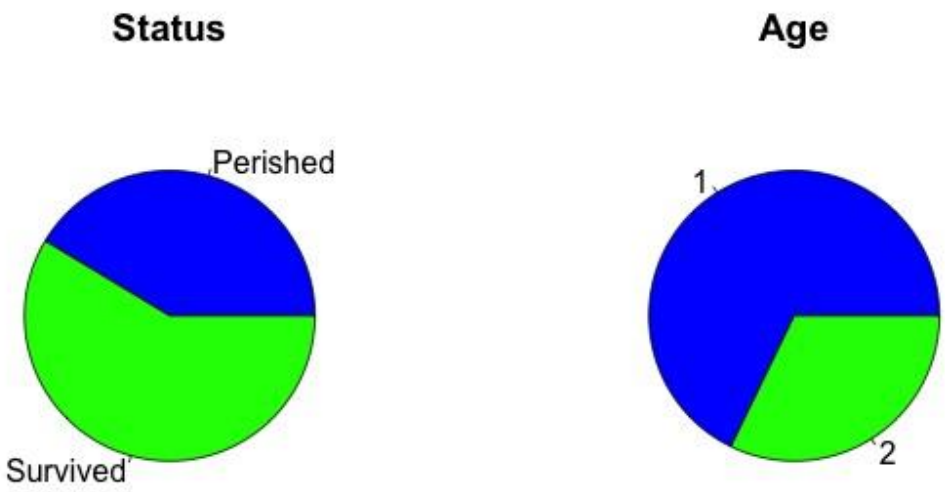


Figure 3. Proportion of different status and age groups in the data set.

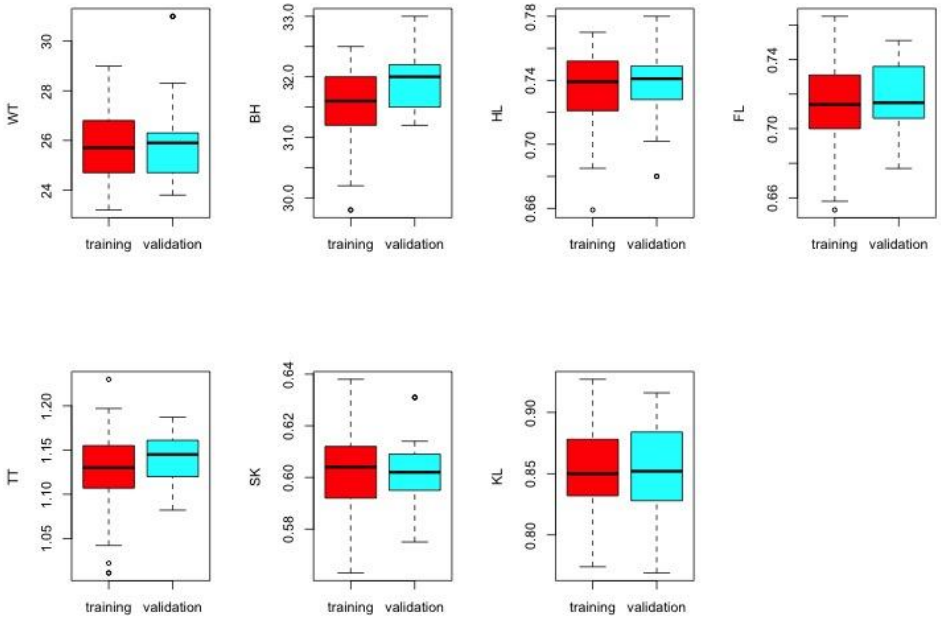


Figure 4. Distribution of training and validation data

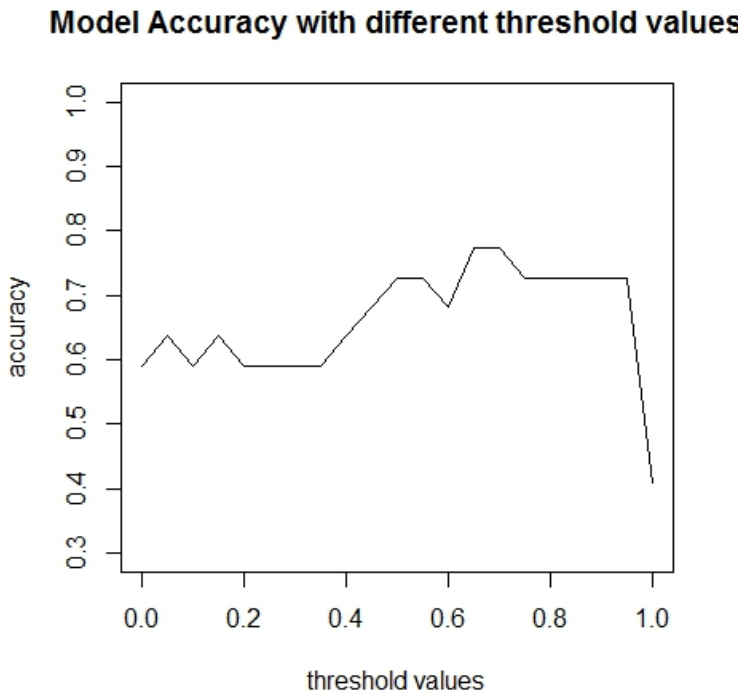


Figure 5. Model accuracy with different threshold values

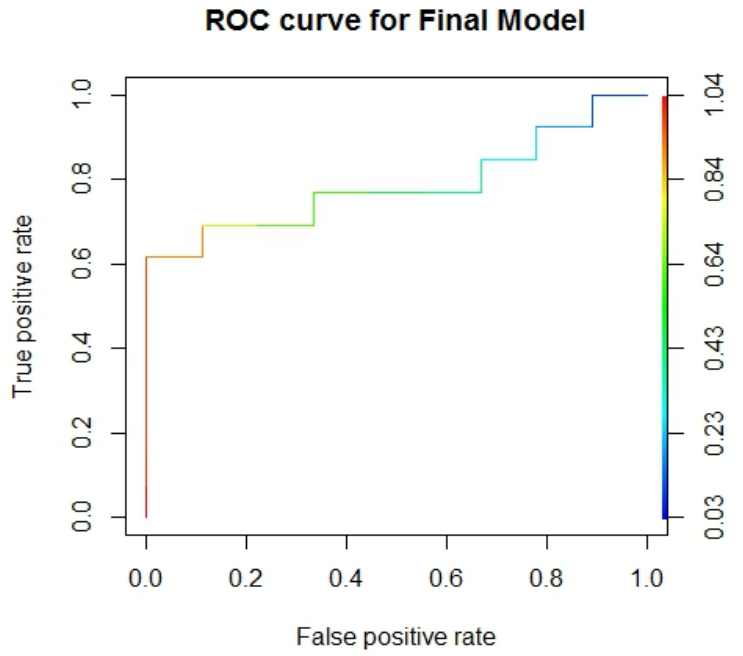


Figure 6. ROC curve for the selected model.

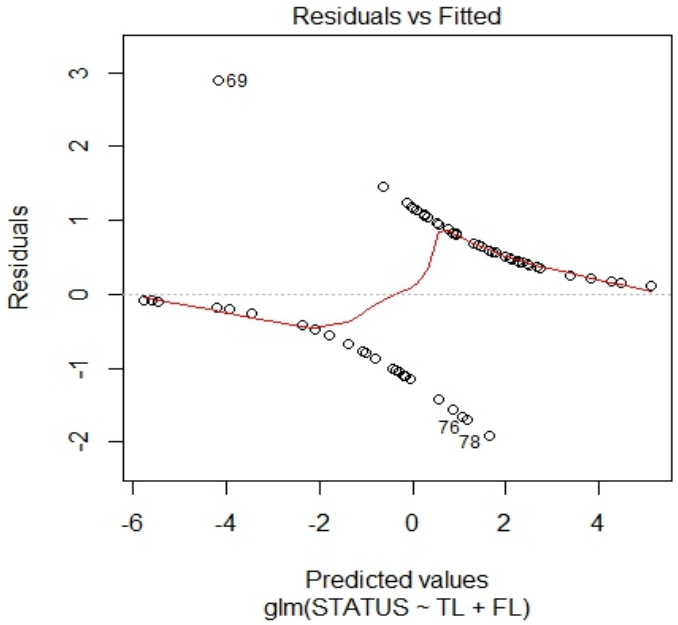


Figure 7. Predicted values versus Residuals.

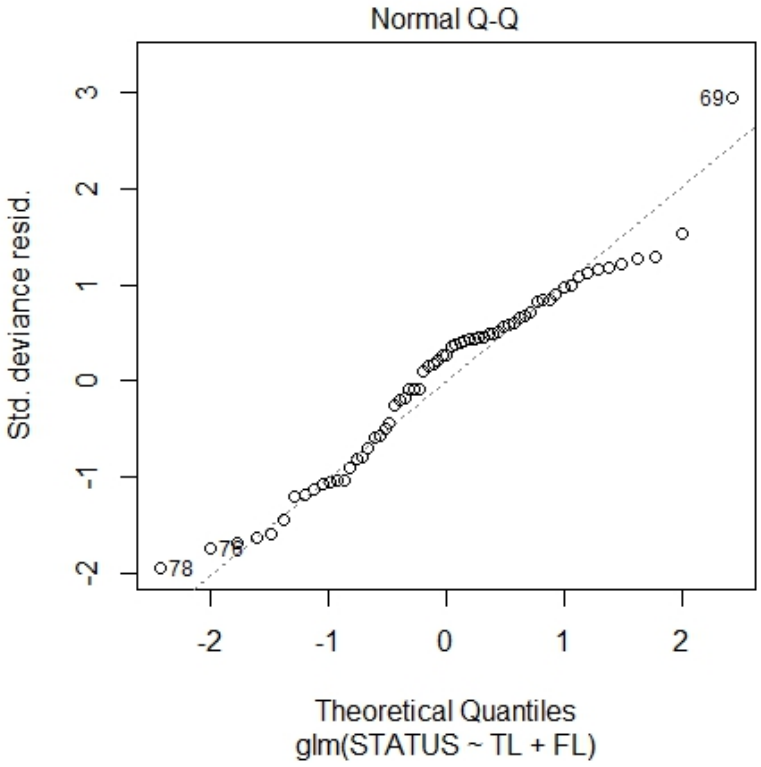


Figure 8. Q-Q plot for the standardized deviance residual.

A2. Tables

Table1. Summary for the initial full model

```
#Fit the model
fullmodel = glm(STATUS ~., family = 'binomial', data = train)
summary(fullmodel)

##
## Call:
## glm(formula = STATUS ~ ., family = "binomial", data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.7329  -0.5811   0.1445   0.4736   1.9265
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  48.0295    32.7171   1.468 0.142098
## AG2           0.2976     0.9122   0.326 0.744276
## TL           -1.0244     0.2983  -3.434 0.000595 ***
## AE            0.1677     0.1705   0.984 0.325301
## WT           -0.5728     0.3660  -1.565 0.117593
## BH           -0.1353     0.8609  -0.157 0.875119
## HL          -15.0541    42.3730  -0.355 0.722383
## FL           66.9238    49.6682   1.347 0.177846
## TT            0.3178    19.1087   0.017 0.986729
## SK           54.3921    34.7304   1.566 0.117319
## KL           29.1097    17.8781   1.628 0.103476
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 88.239  on 64  degrees of freedom
## Residual deviance: 45.044  on 54  degrees of freedom
## AIC: 67.044
##
## Number of Fisher Scoring iterations: 6
```

Table 2. Backward selection method and the summary for the final model this method selects.

```
# Backward selection method
step(model.full, scope = list(lower = model.null),
     direction = 'backward', test = 'F')
##
## Call:  glm(formula = STATUS ~ TL + WT + FL + SK + KL, family = "binomial",
##     data = train)
##
## Coefficients:
## (Intercept)           TL           WT           FL           SK
##     59.2388      -0.9341      -0.5493      64.4373      52.6669
##          KL
##     32.7469
##
## Degrees of Freedom: 64 Total (i.e. Null);  59 Residual
## Null Deviance:       88.24
## Residual Deviance: 46.52     AIC: 58.52
```

Table 3. Forward selection method and the summary for the final model this method selects

```
# Forward seleciton method
step(model.null, scope = list(upper = model.full),
     direction = 'forward', test = 'F')
##
## Call:  glm(formula = STATUS ~ TL + FL + SK + KL + WT, family = "binomial",
##     data = train)
##
## Coefficients:
## (Intercept)           TL           FL           SK           KL
##     59.2388      -0.9341      64.4373      52.6669      32.7469
##          WT
##     -0.5493
##
## Degrees of Freedom: 64 Total (i.e. Null);  59 Residual
## Null Deviance:       88.24
## Residual Deviance: 46.52     AIC: 58.52
```

Table 4. Stepwise method based on AIC and the summary for its best method.

```
##
## Call:  glm(formula = STATUS ~ TL + WT + FL + SK + KL, family = "binomial",
##     data = train)
##
## Coefficients:
## (Intercept)           TL           WT           FL           SK
##     59.2388      -0.9341      -0.5493      64.4373      52.6669
##          KL
##     32.7469
##
## Degrees of Freedom: 64 Total (i.e. Null);  59 Residual
## Null Deviance:       88.24
## Residual Deviance: 46.52     AIC: 58.52
```

Table 5. Stepwise method based on SBC and the summary for its best method.

```
##
## Call:  glm(formula = STATUS ~ TL + FL, family = "binomial", data = train)
##
## Coefficients:
## (Intercept)           TL           FL
##     76.0955      -0.8195      78.3934
##
## Degrees of Freedom: 64 Total (i.e. Null);  62 Residual
## Null Deviance:       88.24
## Residual Deviance: 56.41     AIC: 62.41
```

Table 6. Summary for the model 2.

```
summary(model2)

##
## Call:
## glm(formula = STATUS ~ TL + WT + FL + SK + KL, family = "binomial",
##     data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.4551  -0.5020   0.1410   0.5565   1.7159
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  59.2388    26.9240   2.200 0.027791 *
## TL           -0.9341     0.2697  -3.464 0.000533 ***
## WT           -0.5493     0.3479  -1.579 0.114298
## FL           64.4373    25.4327   2.534 0.011288 *
## SK           52.6669    30.5797   1.722 0.085018 .
## KL           32.7469    15.8474   2.066 0.038791 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 88.239  on 64  degrees of freedom
## Residual deviance: 46.519  on 59  degrees of freedom
## AIC: 58.519
##
## Number of Fisher Scoring iterations: 6
```

Table 7. Summary for the model 3.

```
summary(model3)

##
## Call:
## glm(formula = STATUS ~ TL + FL, family = "binomial", data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.9095  -0.6674   0.2598   0.6604   2.8950
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  76.0955    23.4990   3.238 0.001203 **
## TL           -0.8195     0.2242  -3.655 0.000257 ***
## FL           78.3934    23.9697   3.271 0.001074 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 88.239  on 64  degrees of freedom
## Residual deviance: 56.406  on 62  degrees of freedom
## AIC: 62.406
##
## Number of Fisher Scoring iterations: 5
```

Table 8. Likelihood ratio test for model 2 and model 3.

```
## Likelihood ratio test
##
## Model 1: STATUS ~ TL + FL
## Model 2: STATUS ~ TL + WT + FL + SK + KL
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   3 -28.203
## 2   6 -23.259  3 9.8867    0.01955 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Table 9. VIF values

```
##Multicollinearity effects
vif(model3)

##       TL       FL
## 3.436771 3.436771
```

Table 10. Summary and ANOVA table for the final selected model 3.

```
summary(model3)

##
## Call:
## glm(formula = STATUS ~ TL + FL, family = "binomial", data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.9095  -0.6674   0.2598   0.6604   2.8950
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  76.0955    23.4990   3.238 0.001203 **
## TL           -0.8195     0.2242  -3.655 0.000257 ***
## FL           78.3934    23.9697   3.271 0.001074 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 88.239  on 64  degrees of freedom
## Residual deviance: 56.406  on 62  degrees of freedom
## AIC: 62.406
##
## Number of Fisher Scoring iterations: 5

anova(model3)

## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: STATUS
##
## Terms added sequentially (first to last)
##
##
##      Df Deviance Resid. Df Resid. Dev
## NULL                    64     88.239
## TL    1   14.101        63     74.138
## FL    1   17.732        62     56.406
```

A3. Codes

```
library(readxl)
sparrows = read_excel("survival_sparrow.xls")

###Data Exploration###
sapply(sparrows, class) #get variable type

##      STATUS          AG          TL          AE          WT          BH
## "character"   "numeric"   "numeric"   "numeric"   "numeric"   "numeric"
##          HL          FL          TT          SK          KL
##   "numeric"   "numeric"   "numeric"   "numeric"   "numeric"

sapply(sparrows, function(x) sum(is.na(x))) #check missing values

## STATUS     AG     TL     AE     WT     BH     HL     FL     TT     SK
##      0      0      0      0      0      0      0      0      0      0
##     KL
##      0

summary(sparrows) #summary statistics

##     STATUS                AG              TL              AE
##  Length:87          Min.   :1.000   Min.   :153.0   Min.   :236.0
##  Class :character   1st Qu.:1.000   1st Qu.:158.0   1st Qu.:245.0
##  Mode  :character   Median :1.000   Median :160.0   Median :247.0
##                     Mean   :1.322   Mean   :160.4   Mean   :247.5
##                     3rd Qu.:2.000   3rd Qu.:162.5   3rd Qu.:251.0
##                     Max.   :2.000   Max.   :167.0   Max.   :256.0
##        WT              BH              HL              FL
##  Min.   :23.2   Min.   :29.80   Min.   :0.6590   Min.   :0.6530
##  1st Qu.:24.7   1st Qu.:31.40   1st Qu.:0.7245   1st Qu.:0.7025
##  Median :25.8   Median :31.70   Median :0.7390   Median :0.7140
##  Mean   :25.8   Mean   :31.64   Mean   :0.7354   Mean   :0.7135
##  3rd Qu.:26.7   3rd Qu.:32.10   3rd Qu.:0.7520   3rd Qu.:0.7315
##  Max.   :31.0   Max.   :33.00   Max.   :0.7800   Max.   :0.7650
##        TT              SK              KL
##  Min.   :1.011   Min.   :0.5630   Min.   :0.769
##  1st Qu.:1.111   1st Qu.:0.5920   1st Qu.:0.829
##  Median :1.135   Median :0.6030   Median :0.850
##  Mean   :1.132   Mean   :0.6032   Mean   :0.851
##  3rd Qu.:1.159   3rd Qu.:0.6110   3rd Qu.:0.878
##  Max.   :1.230   Max.   :0.6380   Max.   :0.927

sparrows$STATUS = as.factor(sparrows$STATUS)
sparrows$AG = as.factor(sparrows$AG)

par(mfrow = c(3,3))
for(i in 3:11) {
    hist(sparrows[,i], xlab= names(sparrows)[i], main = paste("Histogram of",
names(sparrows)[i]))
} #histograms of variables
```

##Removed plot

```
#pairwise scatter plot of quantitative variables
par(mfrow = c(1,1))
pairs(~TL + AE + WT + BH + HL + FL + TT + SK + KL, data = sparrows) #pairwise scatter plots
```

##Removed plot

```r
#pairwise correlation matrix of quantitative variables
round(cor(sparrows[,3:11]),2)
```

```
##      TL   AE   WT   BH   HL   FL   TT   SK   KL
## TL 1.00 0.62 0.53 0.30 0.40 0.44 0.38 0.39 0.31
## AE 0.62 1.00 0.50 0.37 0.74 0.71 0.64 0.43 0.46
## WT 0.53 0.50 1.00 0.42 0.48 0.44 0.47 0.36 0.40
## BH 0.30 0.37 0.42 1.00 0.52 0.53 0.53 0.40 0.45
## HL 0.40 0.74 0.48 0.52 1.00 0.88 0.77 0.47 0.49
## FL 0.44 0.71 0.44 0.53 0.88 1.00 0.81 0.45 0.47
## TT 0.38 0.64 0.47 0.53 0.77 0.81 1.00 0.39 0.43
## SK 0.39 0.43 0.36 0.40 0.47 0.45 0.39 1.00 0.26
## KL 0.31 0.46 0.40 0.45 0.49 0.47 0.43 0.26 1.00
```

```r
#pie chart of qualitative variable
par(mfrow = c(1,2))
pie(table(sparrows$STATUS), col = c('blue','green'), main = 'Status')
pie(table(sparrows$AG), col = c('blue','green'), main = 'Age')
```

##Removed plot

```r
#split the data
library(caTools)
set.seed(88)
split = sample.split(sparrows$STATUS, SplitRatio = 0.75)
train = subset(sparrows, split == TRUE)
test = subset(sparrows, split == FALSE)

#check the distribution of the split data
#with side by side box plots of training and validation data
vars = c("TL", "AE", "WT", "BH", "HL", "FL", "TT", "SK", "KL")
train.box = train[,(names(train)%in%vars)]
test.box = test[,(names(test)%in%vars)]

par(mfrow = c(3,3))

for(i in 1:9){
    boxplot(train.box[,i],
            names="training",
            ylab=names(train.box[i]),col=rainbow(2))
}
```

##Removed plot

```r
for(i in 1:9){
    boxplot(test.box[,i],
            names="validation",
            ylab=names(test.box[i]),col=rainbow(2))
}

# Likelihood ratio test for interaction terms
#fit full model
full = glm(STATUS ~ AG + TL + AE + WT + BH + HL + FL + TT + SK + KL + HL:FL + HL:TT + FL:TT,
family = binomial, data = sparrows)

# fit reduced model
reduced = glm(STATUS ~ AG + TL + AE + WT + BH + HL + FL + TT + SK + KL, family= binomial, data
= sparrows)

#difference in loglikelihood,  (14.60)
-2*(logLik(reduced)-logLik(full))
```

```
## 'log Lik.' 6.42592 (df=11)

library(xlsx)

library(ROCR)

library(lme4)

library(lmtest)

library(MASS)
library(leaps)


sur.sparrows = read.xlsx('survival_sparrow.xls', sheetName = 'EX2016')
head(sur.sparrows)

##     STATUS AG  TL  AE   WT   BH    HL    FL    TT    SK    KL
## 1 Survived  1 154 241 24.5 31.2 0.687 0.668 1.022 0.587 0.830
## 2 Survived  1 160 252 26.9 30.8 0.736 0.709 1.180 0.602 0.841
## 3 Survived  1 155 243 26.9 30.6 0.733 0.704 1.151 0.602 0.846
## 4 Survived  1 154 245 24.3 31.7 0.741 0.688 1.146 0.584 0.839
## 5 Survived  1 156 247 24.1 31.5 0.715 0.706 1.129 0.575 0.821
## 6 Survived  1 161 253 26.5 31.8 0.780 0.743 1.144 0.607 0.893

str(sur.sparrows)

## 'data.frame':    87 obs. of  11 variables:
##  $ STATUS: Factor w/ 2 levels "Perished","Survived": 2 2 2 2 2 2 2 2 2 ...
##  $ AG    : num  1 1 1 1 1 1 1 1 1 ...
##  $ TL    : num  154 160 155 154 156 161 157 159 158 158 ...
##  $ AE    : num  241 252 243 245 247 253 251 247 247 252 ...
##  $ WT    : num  24.5 26.9 26.9 24.3 24.1 ...
##  $ BH    : num  31.2 30.8 30.6 31.7 31.5 ...
##  $ HL    : num  0.687 0.736 0.733 0.741 0.715 ...
##  $ FL    : num  0.668 0.709 0.704 0.688 0.706 ...
##  $ TT    : num  1.02 1.18 1.15 1.15 1.13 ...
##  $ SK    : num  0.587 0.602 0.602 0.584 0.575 ...
##  $ KL    : num  0.83 0.841 0.846 0.839 0.821 ...

sur.sparrows$AG = as.factor(sur.sparrows$AG)
sur.sparrows$STATUS = as.factor(sur.sparrows$STATUS)

###########split first into training and validation parts
library(caTools)
set.seed(88)
split = sample.split(sur.sparrows$STATUS, SplitRatio = .75)
train = subset(sur.sparrows, split == TRUE)
test = subset(sur.sparrows, split == FALSE)


#Fit the model
fullmodel = glm(STATUS ~., family = 'binomial', data = train)
summary(fullmodel)

##
## Call:
## glm(formula = STATUS ~ ., family = "binomial", data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.7329  -0.5811   0.1445   0.4736   1.9265
```

```
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  48.0295    32.7171   1.468 0.142098
## AG2           0.2976     0.9122   0.326 0.744276
## TL           -1.0244     0.2983  -3.434 0.000595 ***
## AE            0.1677     0.1705   0.984 0.325301
## WT           -0.5728     0.3660  -1.565 0.117593
## BH           -0.1353     0.8609  -0.157 0.875119
## HL          -15.0541    42.3730  -0.355 0.722383
## FL           66.9238    49.6682   1.347 0.177846
## TT            0.3178    19.1087   0.017 0.986729
## SK           54.3921    34.7304   1.566 0.117319
## KL           29.1097    17.8781   1.628 0.103476
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 88.239  on 64  degrees of freedom
## Residual deviance: 45.044  on 54  degrees of freedom
## AIC: 67.044
##
## Number of Fisher Scoring iterations: 6

anova(fullmodel, test = 'Chisq')

## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: STATUS
##
## Terms added sequentially (first to last)
##
##
##      Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                    64     88.239
## AG    1   0.0222        63     88.216 0.8816059
## TL    1  14.1768        62     74.040 0.0001664 ***
## AE    1  12.1850        61     61.855 0.0004818 ***
## WT    1   1.2776        60     60.577 0.2583446
## BH    1   5.8248        59     54.752 0.0158015 *
## HL    1   2.1430        58     52.609 0.1432172
## FL    1   2.6240        57     49.985 0.1052569
## TT    1   0.0052        56     49.980 0.9423541
## SK    1   1.8474        55     48.132 0.1740804
## KL    1   3.0881        54     45.044 0.0788681 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

#It seems that only 1 of the 11 initial features are statistically significant.

#Let's see how the model performs on validation set


### Let's analyze the features and see whether some of them could be considered as
### unimportant.

model.full = glm(STATUS ~ ., family = 'binomial', data = train)
model.null = glm(STATUS ~ 1, data = train, family = 'binomial')
```

```r
# Backward selection method
step(model.full, scope = list(lower = model.null),
     direction = 'backward', test = 'F')
```

```
## Start:  AIC=67.04
## STATUS ~ AG + TL + AE + WT + BH + HL + FL + TT + SK + KL

## Warning in drop1.glm(fit, scope$drop, scale = scale, trace = trace, k =
## k, : F test assumes 'quasibinomial' family

##        Df Deviance    AIC F value    Pr(>F)
## - TT    1   45.045 65.045  0.0003   0.98553
## - BH    1   45.069 65.069  0.0296   0.86401
## - AG    1   45.151 65.151  0.1283   0.72160
## - HL    1   45.171 65.171  0.1517   0.69847
## - AE    1   46.099 66.099  1.2641   0.26585
## - FL    1   47.033 67.033  2.3841   0.12841
## <none>      45.044 67.044
## - SK    1   47.719 67.719  3.2061   0.07897 .
## - WT    1   47.794 67.794  3.2963   0.07499 .
## - KL    1   48.132 68.132  3.7020   0.05962 .
## - TL    1   68.741 88.741 28.4082 1.971e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Step:  AIC=65.04
## STATUS ~ AG + TL + AE + WT + BH + HL + FL + SK + KL

## Warning in drop1.glm(fit, scope$drop, scale = scale, trace = trace, k =
## k, : F test assumes 'quasibinomial' family

##        Df Deviance    AIC F value    Pr(>F)
## - BH    1   45.069 63.069  0.0299   0.86333
## - AG    1   45.153 63.153  0.1322   0.71755
## - HL    1   45.171 63.171  0.1542   0.69609
## - AE    1   46.149 64.149  1.3488   0.25051
## <none>      45.045 65.045
## - SK    1   47.719 65.719  3.2652   0.07624 .
## - WT    1   47.806 65.806  3.3714   0.07175 .
## - FL    1   47.815 65.815  3.3826   0.07129 .
## - KL    1   48.135 66.135  3.7739   0.05718 .
## - TL    1   69.137 87.137 29.4166 1.345e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Step:  AIC=63.07
## STATUS ~ AG + TL + AE + WT + HL + FL + SK + KL

## Warning in drop1.glm(fit, scope$drop, scale = scale, trace = trace, k =
## k, : F test assumes 'quasibinomial' family

##        Df Deviance    AIC F value    Pr(>F)
## - HL    1   45.193 61.193  0.1537   0.69649
## - AG    1   45.212 61.212  0.1781   0.67462
## - AE    1   46.458 62.458  1.7252   0.19438
## <none>      45.069 63.069
## - FL    1   47.915 63.915  3.5355   0.06527 .
## - WT    1   47.967 63.967  3.6003   0.06293 .
## - SK    1   48.011 64.011  3.6555   0.06101 .
## - KL    1   48.635 64.635  4.4304   0.03980 *
```

```
## - TL    1   69.487 85.487 30.3403 9.449e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Step:  AIC=61.19
## STATUS ~ AG + TL + AE + WT + FL + SK + KL

## Warning in drop1.glm(fit, scope$drop, scale = scale, trace = trace, k =
## k, : F test assumes 'quasibinomial' family

##        Df Deviance    AIC F value    Pr(>F)
## - AG    1   45.281 59.281  0.1108   0.74050
## - AE    1   46.519 60.519  1.6721   0.20119
## <none>      45.193 61.193
## - SK    1   48.014 62.014  3.5577   0.06437 .
## - WT    1   48.379 62.379  4.0182   0.04977 *
## - KL    1   48.789 62.789  4.5359   0.03752 *
## - FL    1   50.061 64.061  6.1402   0.01620 *
## - TL    1   72.748 86.748 34.7548 2.133e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Step:  AIC=59.28
## STATUS ~ TL + AE + WT + FL + SK + KL

## Warning in drop1.glm(fit, scope$drop, scale = scale, trace = trace, k =
## k, : F test assumes 'quasibinomial' family

##        Df Deviance    AIC F value   Pr(>F)
## - AE    1   46.519 58.519  1.5859  0.21296
## <none>      45.281 59.281
## - SK    1   48.186 60.186  3.7220  0.05860 .
## - WT    1   48.524 60.524  4.1542  0.04610 *
## - KL    1   48.958 60.958  4.7100  0.03410 *
## - FL    1   50.189 62.189  6.2875  0.01498 *
## - TL    1   72.858 84.858 35.3232 1.69e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Step:  AIC=58.52
## STATUS ~ TL + WT + FL + SK + KL

## Warning in drop1.glm(fit, scope$drop, scale = scale, trace = trace, k =
## k, : F test assumes 'quasibinomial' family

##        Df Deviance    AIC F value    Pr(>F)
## <none>      46.519 58.519
## - WT    1   49.268 59.268  3.4866  0.066837 .
## - SK    1   49.877 59.877  4.2592  0.043445 *
## - KL    1   51.734 61.734  6.6147  0.012653 *
## - FL    1   55.001 65.001 10.7584  0.001745 **
## - TL    1   73.283 83.283 33.9453 2.507e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##
## Call:  glm(formula = STATUS ~ TL + WT + FL + SK + KL, family = "binomial",
##     data = train)
##
## Coefficients:
## (Intercept)           TL           WT           FL           SK
```

```
##     59.2388      -0.9341      -0.5493      64.4373      52.6669
##          KL
##     32.7469
##
## Degrees of Freedom: 64 Total (i.e. Null);  59 Residual
## Null Deviance:       88.24
## Residual Deviance: 46.52     AIC: 58.52

# Forward seleciton method
step(model.null, scope = list(upper = model.full),
     direction = 'forward', test = 'F')

## Start:  AIC=90.24
## STATUS ~ 1

## Warning in add1.glm(fit, scope$add, scale = scale, trace = trace, k = k, :
## F test assumes quasibinomial family

##        Df Deviance    AIC F value    Pr(>F)
## + TL    1   74.138 78.138 11.9827 0.0009699 ***
## + WT    1   84.368 88.368  2.8904 0.0940428 .
## + HL    1   85.761 89.761  1.8197 0.1821754
## + KL    1   85.928 89.928  1.6938 0.1978465
## <none>      88.239 90.239
## + FL    1   86.270 90.270  1.4376 0.2350230
## + TT    1   86.416 90.416  1.3287 0.2533859
## + SK    1   86.964 90.964  0.9236 0.3401997
## + BH    1   87.841 91.841  0.2850 0.5953370
## + AE    1   88.212 92.212  0.0189 0.8909532
## + AG    1   88.216 92.216  0.0158 0.9002433
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Step:  AIC=78.14
## STATUS ~ TL

## Warning in add1.glm(fit, scope$add, scale = scale, trace = trace, k = k, :
## F test assumes quasibinomial family

##        Df Deviance    AIC F value    Pr(>F)
## + FL    1   56.406 62.406 19.4908 4.121e-05 ***
## + HL    1   59.601 65.601 15.1214 0.0002483 ***
## + TT    1   61.274 67.274 13.0160 0.0006177 ***
## + AE    1   62.151 68.151 11.9578 0.0009891 ***
## + KL    1   64.091 70.091  9.7190 0.0027642 **
## + SK    1   64.560 70.560  9.1979 0.0035352 **
## + BH    1   68.317 74.317  5.2823 0.0249288 *
## <none>      74.138 78.138
## + AG    1   74.040 80.040  0.0820 0.7755314
## + WT    1   74.137 80.137  0.0001 0.9929604
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Step:  AIC=62.41
## STATUS ~ TL + FL

## Warning in add1.glm(fit, scope$add, scale = scale, trace = trace, k = k, :
## F test assumes quasibinomial family

##        Df Deviance    AIC F value  Pr(>F)
## + SK    1   52.892 60.892  4.0517 0.04855 *
```

```
## + KL    1   52.983 60.983  3.9408 0.05163 .
## + AE    1   53.895 61.895  2.8414 0.09698 .
## <none>      56.406 62.406
## + WT    1   55.165 63.165  1.3723 0.24598
## + BH    1   56.148 64.148  0.2796 0.59891
## + HL    1   56.164 64.164  0.2622 0.61048
## + TT    1   56.180 64.180  0.2452 0.62229
## + AG    1   56.371 64.371  0.0371 0.84780
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Step:  AIC=60.89
## STATUS ~ TL + FL + SK

## Warning in add1.glm(fit, scope$add, scale = scale, trace = trace, k = k, :
## F test assumes quasibinomial family

##        Df Deviance    AIC F value  Pr(>F)
## + KL    1   49.268 59.268  4.4140 0.03985 *
## <none>      52.892 60.892
## + AE    1   51.085 61.085  2.1230 0.15032
## + WT    1   51.734 61.734  1.3432 0.25106
## + TT    1   52.759 62.759  0.1518 0.69823
## + HL    1   52.836 62.836  0.0636 0.80181
## + AG    1   52.876 62.876  0.0185 0.89231
## + BH    1   52.892 62.892  0.0005 0.98231
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Step:  AIC=59.27
## STATUS ~ TL + FL + SK + KL

## Warning in add1.glm(fit, scope$add, scale = scale, trace = trace, k = k, :
## F test assumes quasibinomial family

##        Df Deviance    AIC F value  Pr(>F)
## + WT    1   46.519 58.519  3.4866 0.06684 .
## <none>      49.268 59.268
## + AE    1   48.524 60.524  0.9046 0.34542
## + BH    1   48.697 60.697  0.6911 0.40915
## + AG    1   49.238 61.238  0.0352 0.85178
## + HL    1   49.266 61.266  0.0025 0.96066
## + TT    1   49.268 61.268  0.0002 0.98985
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Step:  AIC=58.52
## STATUS ~ TL + FL + SK + KL + WT

## Warning in add1.glm(fit, scope$add, scale = scale, trace = trace, k = k, :
## F test assumes quasibinomial family

##        Df Deviance    AIC F value Pr(>F)
## <none>      46.519 58.519
## + AE    1   45.281 59.281  1.5859 0.2130
## + BH    1   46.169 60.169  0.4400 0.5098
## + HL    1   46.459 60.459  0.0751 0.7851
## + TT    1   46.465 60.465  0.0669 0.7969
## + AG    1   46.519 60.519  0.0002 0.9881
```

```
##
## Call:  glm(formula = STATUS ~ TL + FL + SK + KL + WT, family = "binomial",
##     data = train)
##
## Coefficients:
## (Intercept)           TL           FL           SK           KL
##     59.2388      -0.9341      64.4373      52.6669      32.7469
##          WT
##     -0.5493
##
## Degrees of Freedom: 64 Total (i.e. Null);  59 Residual
## Null Deviance:       88.24
## Residual Deviance: 46.52     AIC: 58.52

# step-wise selection based on AIC criterion
step(model.full) #TL + WT + FL + SK + KL

## Start:  AIC=67.04
## STATUS ~ AG + TL + AE + WT + BH + HL + FL + TT + SK + KL
##
##        Df Deviance    AIC
## - TT    1   45.045 65.045
## - BH    1   45.069 65.069
## - AG    1   45.151 65.151
## - HL    1   45.171 65.171
## - AE    1   46.099 66.099
## - FL    1   47.033 67.033
## <none>      45.044 67.044
## - SK    1   47.719 67.719
## - WT    1   47.794 67.794
## - KL    1   48.132 68.132
## - TL    1   68.741 88.741
##
## Step:  AIC=65.04
## STATUS ~ AG + TL + AE + WT + BH + HL + FL + SK + KL
##
##        Df Deviance    AIC
## - BH    1   45.069 63.069
## - AG    1   45.153 63.153
## - HL    1   45.171 63.171
## - AE    1   46.149 64.149
## <none>      45.045 65.045
## - SK    1   47.719 65.719
## - WT    1   47.806 65.806
## - FL    1   47.815 65.815
## - KL    1   48.135 66.135
## - TL    1   69.137 87.137
##
## Step:  AIC=63.07
## STATUS ~ AG + TL + AE + WT + HL + FL + SK + KL
##
##        Df Deviance    AIC
## - HL    1   45.193 61.193
## - AG    1   45.212 61.212
## - AE    1   46.458 62.458
## <none>      45.069 63.069
## - FL    1   47.915 63.915
## - WT    1   47.967 63.967
## - SK    1   48.011 64.011
## - KL    1   48.635 64.635
## - TL    1   69.487 85.487
```

```
##
## Step:  AIC=61.19
## STATUS ~ AG + TL + AE + WT + FL + SK + KL
##
##        Df Deviance    AIC
## - AG    1   45.281 59.281
## - AE    1   46.519 60.519
## <none>      45.193 61.193
## - SK    1   48.014 62.014
## - WT    1   48.379 62.379
## - KL    1   48.789 62.789
## - FL    1   50.061 64.061
## - TL    1   72.748 86.748
##
## Step:  AIC=59.28
## STATUS ~ TL + AE + WT + FL + SK + KL
##
##        Df Deviance    AIC
## - AE    1   46.519 58.519
## <none>      45.281 59.281
## - SK    1   48.186 60.186
## - WT    1   48.524 60.524
## - KL    1   48.958 60.958
## - FL    1   50.189 62.189
## - TL    1   72.858 84.858
##
## Step:  AIC=58.52
## STATUS ~ TL + WT + FL + SK + KL
##
##        Df Deviance    AIC
## <none>      46.519 58.519
## - WT    1   49.268 59.268
## - SK    1   49.877 59.877
## - KL    1   51.734 61.734
## - FL    1   55.001 65.001
## - TL    1   73.283 83.283

##
## Call:  glm(formula = STATUS ~ TL + WT + FL + SK + KL, family = "binomial",
##     data = train)
##
## Coefficients:
## (Intercept)           TL           WT           FL           SK
##     59.2388      -0.9341      -0.5493      64.4373      52.6669
##          KL
##     32.7469
##
## Degrees of Freedom: 64 Total (i.e. Null);  59 Residual
## Null Deviance:       88.24
## Residual Deviance: 46.52     AIC: 58.52

# step-wise selection based on BIC criterion
step(model.full, k = log(nrow(train))) #TL, FL

## Start:  AIC=90.96
## STATUS ~ AG + TL + AE + WT + BH + HL + FL + TT + SK + KL
##
##        Df Deviance     AIC
## - TT    1   45.045  86.789
## - BH    1   45.069  86.813
## - AG    1   45.151  86.895
```

```
## - HL    1   45.171  86.915
## - AE    1   46.099  87.843
## - FL    1   47.033  88.777
## - SK    1   47.719  89.463
## - WT    1   47.794  89.538
## - KL    1   48.132  89.876
## <none>      45.044  90.963
## - TL    1   68.741 110.485
##
## Step:  AIC=86.79
## STATUS ~ AG + TL + AE + WT + BH + HL + FL + SK + KL
##
##        Df Deviance     AIC
## - BH    1   45.069  82.639
## - AG    1   45.153  82.722
## - HL    1   45.171  82.740
## - AE    1   46.149  83.719
## - SK    1   47.719  85.288
## - WT    1   47.806  85.375
## - FL    1   47.815  85.384
## - KL    1   48.135  85.705
## <none>      45.045  86.789
## - TL    1   69.137 106.706
##
## Step:  AIC=82.64
## STATUS ~ AG + TL + AE + WT + HL + FL + SK + KL
##
##        Df Deviance     AIC
## - HL    1   45.193  78.588
## - AG    1   45.212  78.608
## - AE    1   46.458  79.853
## - FL    1   47.915  81.310
## - WT    1   47.967  81.362
## - SK    1   48.011  81.406
## - KL    1   48.635  82.030
## <none>      45.069  82.639
## - TL    1   69.487 102.882
##
## Step:  AIC=78.59
## STATUS ~ AG + TL + AE + WT + FL + SK + KL
##
##        Df Deviance     AIC
## - AG    1   45.281  74.501
## - AE    1   46.519  75.739
## - SK    1   48.014  77.234
## - WT    1   48.379  77.599
## - KL    1   48.789  78.010
## <none>      45.193  78.588
## - FL    1   50.061  79.282
## - TL    1   72.748 101.969
##
## Step:  AIC=74.5
## STATUS ~ TL + AE + WT + FL + SK + KL
##
##        Df Deviance    AIC
## - AE    1   46.519 71.565
## - SK    1   48.186 73.233
## - WT    1   48.524 73.570
## - KL    1   48.958 74.004
## <none>      45.281 74.501
## - FL    1   50.189 75.236
```

```
## - TL    1   72.858 97.904
##
## Step:  AIC=71.57
## STATUS ~ TL + WT + FL + SK + KL
##
##        Df Deviance    AIC
## - WT    1   49.268 70.140
## - SK    1   49.877 70.749
## <none>      46.519 71.565
## - KL    1   51.734 72.606
## - FL    1   55.001 75.873
## - TL    1   73.283 94.155
##
## Step:  AIC=70.14
## STATUS ~ TL + FL + SK + KL
##
##        Df Deviance     AIC
## - KL    1   52.892  69.590
## - SK    1   52.983  69.680
## <none>      49.268  70.140
## - FL    1   56.078  72.775
## - TL    1   85.129 101.826
##
## Step:  AIC=69.59
## STATUS ~ TL + FL + SK
##
##        Df Deviance    AIC
## - SK    1   56.406 68.929
## <none>      52.892 69.590
## - FL    1   64.560 77.083
## - TL    1   86.009 98.532
##
## Step:  AIC=68.93
## STATUS ~ TL + FL
##
##        Df Deviance    AIC
## <none>      56.406 68.929
## - FL    1   74.138 82.486
## - TL    1   86.270 94.619

##
## Call:  glm(formula = STATUS ~ TL + FL, family = "binomial", data = train)
##
## Coefficients:
## (Intercept)           TL           FL
##     76.0955      -0.8195      78.3934
##
## Degrees of Freedom: 64 Total (i.e. Null);  62 Residual
## Null Deviance:       88.24
## Residual Deviance: 56.41     AIC: 62.41

##Model 2 is based on Backward, Forward and AIC step-wise selection methods.
model2 = glm(formula = STATUS ~ TL + WT + FL + SK + KL,
             family = "binomial", data = train)

# Model 3 is based on the BIC step-wise selection method.
model3 = glm(formula = STATUS ~ TL + FL, family = "binomial",
             data = train)

# Summaries for model2 and model3
summary(model2)
```

```
##
## Call:
## glm(formula = STATUS ~ TL + WT + FL + SK + KL, family = "binomial",
##     data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.4551  -0.5020   0.1410   0.5565   1.7159
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  59.2388    26.9240   2.200 0.027791 *
## TL           -0.9341     0.2697  -3.464 0.000533 ***
## WT           -0.5493     0.3479  -1.579 0.114298
## FL           64.4373    25.4327   2.534 0.011288 *
## SK           52.6669    30.5797   1.722 0.085018 .
## KL           32.7469    15.8474   2.066 0.038791 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 88.239  on 64  degrees of freedom
## Residual deviance: 46.519  on 59  degrees of freedom
## AIC: 58.519
##
## Number of Fisher Scoring iterations: 6

summary(model3)

##
## Call:
## glm(formula = STATUS ~ TL + FL, family = "binomial", data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.9095  -0.6674   0.2598   0.6604   2.8950
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  76.0955    23.4990   3.238 0.001203 **
## TL           -0.8195     0.2242  -3.655 0.000257 ***
## FL           78.3934    23.9697   3.271 0.001074 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 88.239  on 64  degrees of freedom
## Residual deviance: 56.406  on 62  degrees of freedom
## AIC: 62.406
##
## Number of Fisher Scoring iterations: 5

##################
#Lets do Likelihood ratio test, whether WT and SK could be dropped, alpha = 0.05
#Hnull: model3 is true
#Halt:  model2 is true

lrtest(model3, model2)
```

```
## Likelihood ratio test
##
## Model 1: STATUS ~ TL + FL
## Model 2: STATUS ~ TL + WT + FL + SK + KL
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   3 -28.203
## 2   6 -23.259  3 9.8867    0.01955 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
chi_sq_statistic = lrtest(model2, model3)$Chisq[2]
chi_sq_statistic
```

```
## [1] 9.886719
```

```
chi_sq_star = qchisq(1 - 0.05, 3)
chi_sq_star
```

```
## [1] 7.814728
```

```
chi_sq_statistic > chi_sq_star
```

```
## [1] TRUE
```

```
#based on the Likelihood ratio test, we reject Hnull at alpha = 0.05
#hence, choose model 3 as the right one.

#Now that we selected model 3, let's see how it performs on validation set
#and obtain model's accuracy and ROC measures.

#1. Find a suitable threshold value to classify the predictions as
#   Survived and Perished.

calculate_accuracy = function(amodel, threshold){
    fitted.results = predict(amodel, newdata=test, type='response')
    fitted.results = ifelse(fitted.results > threshold, 'Survived', 'Perished')
    misClasificError = mean(fitted.results != test$STATUS)

    1 - misClasificError
}

accuracies = sapply(seq(0, 1, 0.05), function(x) calculate_accuracy(model3, x))

plot(seq(0, 1, 0.05), accuracies, type = 'l',
     main = 'Model Accuracy with different threshold values',
     xlab = 'threshold values', ylab = 'accuracy', cex = 2,
     ylim = c(0.3, 1))
which(accuracies == max(accuracies))
```

```
## [1] 14 15
```

```
#Hence, choose threshold value as 0.67

calculate_accuracy(model3, 0.67)
```

```
## [1] 0.7727273
```

```
#Confusion matrix
p = predict(model3, newdata= test, type="response")
table(test$STATUS, p >= 0.67)
```

```
##
##            FALSE TRUE
##   Perished     8    1
##   Survived     4    9

true_positive = 8 / 12
true_positive

## [1] 0.6666667

false_positive = 1 / 10
false_positive

## [1] 0.1

#ROC plot
plot_roc = function(amodel){
    p = predict(amodel, newdata= test, type="response")
    pr = prediction(p, test$STATUS)
    prf = performance(pr, measure = "tpr", x.measure = "fpr")
    auc = performance(pr, measure = "auc")
    auc = auc@y.values[[1]]
    print (paste('AUC is ', auc))
    return (plot(prf, colorize = TRUE, text.adj = c(-0.2,1.7),
                 main = 'ROC curve for Final Model'))
}


plot_roc(model3)

## [1] "AUC is  0.786324786324786"

plot(model3)
```

##Removed plot