**Introduction**

Haberman's Survival dataset contains cases from a study that was conducted between 1958 and 1970 at the University of Chicago's Billings Hospital on the survival of patients who had undergone surgery for breast cancer. The dataset contains four variables: patient’s age at the time of surgery, year of the surgery, the number of positive axillary nodes detected, and survival status. The survival status is 1 if the patient survived 5 years or longer and 2 if the patient died within 5 years. Here, we analyse how the first three variables impact the survival of a patient. While the patient’s age and year of surgery are intuitive, positive axillary node is a lymph node in the armpit where the breast cancer has spread. There are 306 observations in the dataset.

**Data Exploration**

As stated above, there are 4 variables and 306 observations. There are no missing values for any of the variables. While patient’s age, year of surgery and positive axillary nodes are predictor (independent) variables, survival status is a response (dependent) variable. All the predictor variables are continuous and the response variable is discrete or categorical. Since the response variable is categorical, logical regression technique can be used for analysis.

There are no outliers for any of the predictor variables. However, for positive axillary nodes, values above 10 can be considered extreme.

**Data Preparation**

The response variable has values 1 and 2. As per logistic regression model, this needs to be converted to 0 and 1. Patient who survived 5 years or more can be coded as 0 and patient who died within 5 years as 1. In R, we get the probability of the outcome that is coded as 1. Before building the model, split the dataset into training and validation dataset in 7:3 ratio.

**Build the Model**

Logistic regression technique is used to build the model. In R language, *glm()* function is used to build the model. The first step is to build the model on the training dataset. Then, apply the model on the validation dataset to predict the probability of survival that is coded as 1(In this case, survival within 5 years). To obtain model performance, confusion matrix, ROC curve and AUC are used.

In the first iteration, all the predictor variables are used. Fit a logistic regression model on the training data set using the following code.

*hb\_model <- glm(survival\_rate~patient\_age + op\_year + pos\_axillary\_nodes, data = train, family = "binomial")*

**Interpret the results**

hb\_model now contains the coefficient associated with the logistic model. *Summary(hb\_model)* gives summary of the model. The output table of this command shows the estimated coefficients and p-values associated with the test of statistical significance of coefficients of each variable.

*Call:*

*glm(formula = Survival ~ Patient\_Age + Op\_Year + Pos\_Axillary\_Nodes,*

*family = "binomial", data = train)*

*Deviance Residuals:*

*Min 1Q Median 3Q Max*

*-2.3763 -0.7510 -0.6865 1.0395 1.8885*

*Coefficients:*

*Estimate Std. Error z value Pr(>|z|)*

*(Intercept) -2.231106 3.159862 -0.706 0.480140*

*Patient\_Age 0.015103 0.014671 1.029 0.303291*

*Op\_Year 0.002006 0.050054 0.040 0.968036*

*Pos\_Axillary\_Nodes 0.087901 0.022670 3.877 0.000106 \*\*\**

*---*

*Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1*

*(Dispersion parameter for binomial family taken to be 1)*

*Null deviance: 257.61 on 213 degrees of freedom*

*Residual deviance: 237.56 on 210 degrees of freedom*

*AIC: 245.56*

*Number of Fisher Scoring iterations: 4*

While *pos\_axillary\_nodes* has statistically significant coefficients, *op\_year* and *patient\_age* do not. Since the variables are not multi-collinear and Null deviance (deviance when the model is developed without predictor variables), Residual deviance (deviance when the model is developed with predictor variables) and AIC (captures the trade off between explanability and complexity) do not change much by dropping them from the fit, they are retained in the model.

The coefficients in the logistic regression output are expressed in terms of log of odds. So, the coefficient 0.087901implies that a one unit change in Pos\_Axillary\_Nodes results in a 0.087901 unit change in the log of the odds of survival less than 5 years. Similarly for Patient\_Age and op\_Year. The odds ratio can be computed by raising **e** to the power of the logistic coefficient.

e0.087901 = 1.091988

e0.015103 = 1.015218

e0.002006 =1.002002

For pos\_axillary\_nodes, a unit change results in 1.09 times increase in survival less than 5 years, for patient age, it increases by 1.015 times and for op\_year, it increases by 1.002 times.

**Analyse model performance**

Apply the above model on validation sample to predict the probability of survival. Prediction in R requires use of *predict(object, data, type)* function. Predict the probability of each datum in the validation data set taking the value 2 using the following command.

*predictTest <- predict(hb\_model, test, type="response")*

Finally, performance of the model needs to be measured. Confusion matrix, ROC curve and AUC are some of the measures used.

A **confusion matrix** for a binary classifier is a 2 X 2 matrix in which each row represents actual values and each column, predicted values. To classify the predicted probability of each datum as good(1) or bad(0), a cutoff probability needs to be specified. A value above the cutoff probability indicates a good record and below that indicates a bad record. In this example, 0.5 is used as a cutoff value.

*predictTestbkt <- ifelse(predictTest>0.5, "G", "B")*

*table(predictTestbkt, test$Survival)*

The above commands yield the following confusion matrix.

*predictTestbkt 1 2*

*B 71 16*

*G 2 3*

Using confusion matrix, *accuracy* of the classifier is calculated as (TP + TN) /total

Where TP (True Positive) is when both actual and predicted values are 1 and TN (True Negative) is when both actual and predicted values are 0

*Misclassification Rate* is calculated as (FP + FN)/ total. This is equivalent to 1 – accuracy.

Where FP (False Positive) is when actual is 0 and predicted is 1 and FN (False Negative) is when actual is 1 and predicted is 0.

Confusion matrix for this model fit has yielded an accuracy of (71+3)/(71+16+2+3) = 0.8 which is fairly high.

With different cut off values, the predicted value varies. **ROC curve** helps in finding the optimal cut off value. It plots TPR (True Positive Rate) against FPR (False Positive Rate) at various cut off settings.

Where TPR is calculated as TP/total positives and FPR as FP/total negatives.

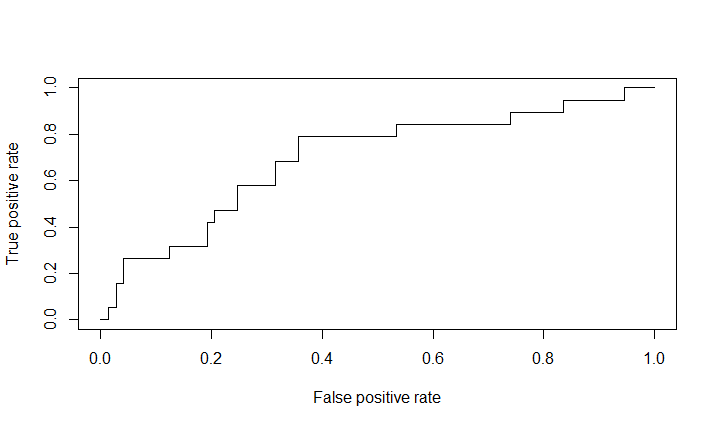
The cut off value that gives the highest TPR for a lowest FPR is chosen.

*pred <- prediction(predicted, train$Survival)*

*perf<- performance(pred, "tpr", "fpr")*

*plot(perf)*

The resulting ROC curve is as follows.

**

**AUC (Area under the curve)** metric is used to find the most optimal ROC curve. It calculates the area under ROC curve. The code to find AUC is given below.

*auc <- performance(pred,"auc")*

*auc<- unlist(slot(auc,"y.values"))*

This model gives an AUC score of 0.70, which is considered good for a model.