Example: An Implementation of the Logistic Regression Analysis with R

```
> library(rpart)
>data(kyphosis, package="rpart")
n<- length(kyphosis[,1])
>kyphosis.class<-rep(0, n)
>for(i in 1:n) {if(kyphosis[i,1]==”present”) kyphosis.class[i]<-1}
# Class: “present” -> 1, and “absent” -> 0

> kyphosis.all<-cbind(kyphosis[,2:4],kyphosis.class)
>kyphosis.train<-kyphosis.all[1:50,]
>kyphosis.test<-kyphosis.all[51:n,]
# Create the objects for the train and test data.

>result.LR1<-glm(kyphosis.class ~ ., data=kyphosis.train, binomial)
>summary(result.LR1)
# Obtain the estimate of the Logistic Regression model.

>result.LR2<-predict(result.LR1,newdata=kyphosis.test, type="response")
>table(kyphosis.test[,4],round(result.LR2))
# Create the confusion matrix for the test data (the threshed =0.5)

> library(ROCR)
>fit.LR.pred<- prediction(result.LR2, kyphosis.test[,4])
>fit.LR.perf <- performance(fit.LRpred,"tpr","fpr")
> plot(fit.LR.perf,lwd=2,col="blue", main="ROC: LR on Kyphosis")
>abline(a=0,b=1)
# Plot the ROC Curve by using the package ROCR

>auc.LR.tmp <- performance(fit.LR.pred, "auc")
> auc.Tree <- as.numeric(auc.LR.tmp@y.values)
>auc.LR
# Obtain the AUC by using the package ROCR
```

See the web site: http://www.statmethods.net/advstats/glm.html and below:

Generalized Linear Models

Generalized linear models are fit using the **glm( )** function. The form of the **glm** function is **glm(***formula*, **family=***familytype*(**link=***linkfunction*), **data=)**

Family	Default Link Function
binomial	(link = "logit")
gaussian	(link = "identity")
Gamma	(link = "inverse")
inverse.gaussian	(link = "1/mu^2")
poisson	(link = "log")
quasi	(link = "identity", variance = "constant")
quasibinomial	(link = "logit")
quasipoisson	(link = "log")

See **help(glm)** for other modeling options. See **help(family)** for other allowable link functions for each family. Three subtypes of generalized linear models will be covered here: logistic regression, poisson regression, and survival analysis.

Logistic Regression

Logistic regression is useful when you are predicting a binary outcome from a set of continuous predictor variables. It is frequently preferred over discriminant function analysis because of its less restrictive assumptions.

```
# Logistic Regression
# where F is a binary factor and
# x1-x3 are continuous predictors
fit <- glm(F~x1+x2+x3,data=mydata,family=binomial())
summary(fit) # display results
confint(fit) # 95% CI for the coefficients
exp(coef(fit)) # exponentiated coefficients
exp(confint(fit)) # 95% CI for exponentiated coefficients
predict(fit, type="response") # predicted values
residuals(fit, type="deviance") # residuals
```