```
#analysis of arsenic in wells data

library(arm)
library(pROC)

arsenic = read.csv("arsenic.csv", header = T)
dim(arsenic)
summary(arsenic)

#let's do some exploratory data analysis
boxplot(arsenic~switch, data = arsenic, xlab = "Switch", ylab = "Arsenic")
boxplot(dist~switch, data = arsenic, xlab = "Switch", ylab = "Distance")
boxplot(educ~switch, data = arsenic, xlab = "Switch", ylab = "Education")

# if you want to see data in tabular format
table(arsenic$assoc, arsenic$switch)
table(arsenic$assoc, arsenic$switch)/3020

#another way to see relationships of categorical variables and response -- tapply command
tapply(arsenic$switch, arsenic$assoc, mean)

#could do with education, too, since it is an integer variable
tapply(arsenic$switch, arsenic$educ, mean)
table(arsenic$educ)

#remember that there are few observations at some of these values of the predictors, so the
percentages
# need to be considered in the context of large uncertainties.  but, this does suggest a
change at 7 years of education.
#we might consider a dummy variable for 7 or higher rather than a linear term... something to
try later.

#let's look at binnedplots of continuous predictors versus switch
#ignore the SD lines in these plots -- they are only relevant when plotting binned residuals
versus the predicted probabilities

binnedplot(arsenic$arsenic, y=arsenic$switch, xlab = "Arsenic", ylab = "Switch cases", main =
"Binned Arsenic and Switch cases")

#note the quickly increasing trend followed by flattening. Probability does not start to
decrease, though, so
#unlikely we'd want a quadratic term.  We would expect some flattening with a linear trend.

binnedplot(arsenic$dist, y=arsenic$switch, xlab = "Distance", ylab = "Switch cases", main =
"Binned Distance and Switch cases")
#no obvious transformation suggested.

#let's try a logistic regression that has a main effect for every variable and linear
predictors
#begin by centering the continuous predictors (we'll leave educ alone since we might use a
dummy variable later)

arsenic$arsenic.c = arsenic$arsenic - mean(arsenic$arsenic)
arsenic$dist.c = arsenic$dist - mean(arsenic$dist)

arsreg1 = glm(switch ~ arsenic.c + dist.c + assoc + educ, data = arsenic, family = binomial)
summary(arsreg1)



####model diagnostics

##binned residual plots

# compute raw residuals
```

```
rawresid1 = arsenic$switch - fitted(arsreg1)

binnedplot(x=arsenic$arsenic.c, y = rawresid1, xlab = "Arsenic centered", ylab = "Residuals",
main = "Binned residuals versus arsenic")
#note the up-then-down nature of the binned plot! we might try a transformation, say log.

binnedplot(x=arsenic$dist.c, y = rawresid1, xlab = "Distance centered", ylab = "Residuals",
main = "Binned residuals versus distance")
#not as much of a trend, really.

#let's look at average residuals by education using the tapply command

tapply(rawresid1, arsenic$educ, mean)
#unremarkable -- although, many average residuals for educ < 7 are negative and for educ > 7
are positive (at least where the sample size for the education level is non-negligible
#we could try the dummy variable split....

tapply(rawresid1, arsenic$assoc, mean)
#nothing helpful here...

#let's do the confusion matrix with .5 threshold and with .58 threshold (marginal percentage
in data)

threshold = 0.5
table(arsenic$switch, arsreg1$fitted > threshold)

threshold = 0.58
table(arsenic$switch, arsreg1$fitted > threshold)

#huge difference!  seems a lot of predicted probabilities are in the .5 yo .58  range, so
cutoff matters.
#either way, we have large off-diagonal numbers.  let's see if we can't improve the model.

#look at ROC curve
roc(arsenic$switch, fitted(arsreg1), plot=T, legacy.axes=T)

#pretty tight to the line -- not a strongly predictive logistic regression

#based on binned residual plot, let's try a log of arsenic to start with, and see what
happens.

arsenic$logarsenic = log(arsenic$arsenic)
arsenic$logarsenic.c = arsenic$logarsenic - mean(arsenic$logarsenic)

arsreg2 = glm(switch ~ logarsenic.c + dist.c + assoc + educ, data = arsenic, family =
binomial)
summary(arsreg2)

#back to diagnostics

rawresid2 = arsenic$switch - fitted(arsreg2)

binnedplot(x=arsenic$logarsenic.c, y = rawresid2, xlab = "Log(Arsenic) centered", ylab =
"Residuals", main = "Binned residuals versus log(arsenic)")
#seems to have helped some!

binnedplot(x=arsenic$dist, y = rawresid2, xlab = "Distance centered", ylab = "Residuals",
main = "Binned residuals versus distance")
#still not as much of a trend.

tapply(rawresid2, arsenic$educ, mean)
#similar pattern as before for education.

#let's do the confusion matrix

threshold = 0.5
```

```
table(arsenic$switch, arsreg2$fitted > threshold)

threshold = 0.58
table(arsenic$switch, arsreg2$fitted > threshold)

#we seem to have improved at .58 threshold (although not by much, really).

#look at ROC curve
roc(arsenic$switch, fitted(arsreg2), plot=T, legacy.axes=T)

#not much difference from last curve really, although a little more prediction accuracy


#let's see what happens if we make education a binary split at 7.
arsenic$educg6 = rep(0, 3020)
arsenic$educg6[arsenic$educ > 6] = 1

arsreg3 = glm(switch ~ logarsenic.c + dist.c + assoc + educg6, data = arsenic, family =
binomial)
summary(arsreg3)

#this seems to have helped the significance of education, and it is scientifically plausible.
let's keep it!
#should go through binned residuals again to make sure this did not worsen model fit.  It did
not.

#look at new roc curve to see if it is any better
roc(arsenic$switch, fitted(arsreg3), plot=T, legacy.axes=T)

# it is a little better, with a higher area under the curve (AUC = .6632).  let's keep this
model.

### interactions in logistic regression

#scientifically, it is plausible to think that there might be interactions among all the
variables and arsenic, or among education and distance.
#let's add the interactions to see if any stand out.

#first, here is a way to explore the data for interactions using the binnedplot command

#lets set up the graphics device to show two plots side by side
par(mfcol=c(2,1))

#first plot for educg6 = 0
binnedplot(arsenic$logarsenic.c[arsenic$educg6==0], y=arsenic$switch[arsenic$educg6==0], xlab
= "Log Arsenic", ylab = "Switch cases", main = "Binned Arsenic and Switch cases (Educ <7)")

#next the plot for educg6 = 1
binnedplot(arsenic$logarsenic.c[arsenic$educg6==1], y=arsenic$switch[arsenic$educg6==1], xlab
= "Log Arsenic", ylab = "Switch cases", main = "Binned Arsenic and Switch cases (Educ > 6)")

#we are looking for differences in the trend.  not strong ones, except possibly at low levels
of arsenic.
#I will include an interaction based on scientific arguments in favor of an interaction
effect, but I am not expecting a very strong interaction effect based on this plot.

#let's try for association and arsenic
#first plot for assoc = 0
binnedplot(arsenic$logarsenic.c[arsenic$assoc==0], y=arsenic$switch[arsenic$assoc==0], xlab =
"Log Arsenic", ylab = "Switch cases", main = "Binned Arsenic and Switch cases (Assoc = 0)")

#next the plot for assoc = 1
binnedplot(arsenic$logarsenic.c[arsenic$assoc==1], y=arsenic$switch[arsenic$assoc==1], xlab =
"Log Arsenic", ylab = "Switch cases", main = "Binned Arsenic and Switch cases (Assoc = 1)")

#even less reason to suspect an interaction effect from this plot.
```

```
#how about distance and education?
#first plot for educg6 = 0
binnedplot(arsenic$dist.c[arsenic$educg6==0], y=arsenic$switch[arsenic$educg6==0], xlab =
"Distance", ylab = "Switch cases", main = "Binned Distance and Switch cases (Educ <7)")

#next the plot for educg6 = 1
binnedplot(arsenic$dist.c[arsenic$educg6==1], y=arsenic$switch[arsenic$educg6==1], xlab =
"Distance", ylab = "Switch cases", main = "Binned Distance and Switch cases (Educ > 6)")

#this is a little more interesting -- we see one plot flatten and the other decrease.  here
an interaction might be useful.


#let's first try the model with all the interactions
arsreg4 = glm(switch ~ dist.c*educg6  + logarsenic.c * (assoc + educg6), data = arsenic,
family = binomial)
summary(arsreg4)

#these collectively look sort of useful, especially the education ones!

#change in deviance tests to see if the full set of interactions are useful.

anova(arsreg4, arsreg3, test= "Chisq")

#the whole group of interactions is significant.  let's just test if the distance interaction
is useful, given the other two in the model.

arsreg4a= glm(switch ~ logarsenic.c * (assoc + educg6), data = arsenic, family = binomial)
summary(arsreg4a)

anova(arsreg4a, arsreg4, test= "Chisq")

#looks like the interaction with distance and education is useful.

#let's make our final model (arsreg5) be the one with all the interaction effects.

arsreg5 = glm(switch ~ dist.c*educg6  + logarsenic.c * (assoc + educg6), data = arsenic,
family = binomial)
summary(arsreg5)

#let's do the binned residual plots with this perhaps final model one more time!

rawresid5 = arsenic$switch - fitted(arsreg5)

binnedplot(x=arsenic$logarsenic, y = rawresid5, xlab = "Log(Arsenic) centered", ylab =
"Residuals", main = "Binned residuals versus log(arsenic)")
binnedplot(x=arsenic$dist, y = rawresid5, xlab = "Distance centered", ylab = "Residuals",
main = "Binned residuals versus distance")
tapply(rawresid5, arsenic$educ, mean)
#a little more diversity with education, so that seems to have helped.
#still a little trouble fitting small log arsenic, but not too much more we can do....  go
with this model!

#let's do the confusion matrix

threshold = 0.5
table(arsenic$switch, arsreg5$fitted > threshold)

threshold = 0.58
table(arsenic$switch, arsreg5$fitted > threshold)

#still not moving much.... the model can predict only so well

#ROC curve...
roc(arsenic$switch, fitted(arsreg5), plot=T, legacy.axes=T)
```

```
#a little better still... but we really aren't gaining a whole lot.  this is about as
#good as we are going to get with only these variables, apparently.

###model interpretations

confint.default(arsreg5)   #on log odds scale
exp(confint.default(arsreg5))   #on odds scale

##model is quite complicated to interpret due to interactions.  arsenic is of most interest.
#let's make plots to display relationships.

#plot of predicted probabilities as arsenic increases for different groups.
#set distance = to average distance (centering means we don't need to worry about it when
making predictions at the average distance)

#create some arsenic values in line with those in the data, going from logs centered to raw
scale.
samplelogarsenic.c = seq(from = -1, to = 2, by = .1)
samplelogarsenic = samplelogarsenic.c + mean(arsenic$logarsenic)
samplearsenic = exp(samplelogarsenic)

#set association = educg6 = 0.
logitpredvalue = .239731 + .94455*samplelogarsenic.c
predprobbaseline = exp(logitpredvalue) / (1 + exp(logitpredvalue))

plot(y=predprobbaseline, x= samplearsenic, pch= 1, xlab = "Arsenic", ylab = "Predicted
probability", main = "Arsenic vs. Predicted Probability")

#set association =1, educg6 = 0
logitpredvalue = .239731 + .94455*samplelogarsenic.c -.14144 - .241134*samplelogarsenic.c
predprobassoc = exp(logitpredvalue) / (1 + exp(logitpredvalue))

plot(y=predprobassoc, x= samplearsenic, pch= 2, xlab = "Arsenic", ylab = "Predicted
probability", main = "Arsenic vs. Predicted Probability")

#set association =0 , educg6 = 1
logitpredvalue = .239731 + .94455*samplelogarsenic.c + .526891 + .24461*samplelogarsenic.c
predprobeducg6 = exp(logitpredvalue) / (1 + exp(logitpredvalue))

plot(y=predprobeducg6, x= samplearsenic, pch= 3, xlab = "Arsenic", ylab = "Predicted
probability", main = "Arsenic vs. Predicted Probability")

#set association = 1, educg6 = 1
logitpredvalue = .239731 + .94455*samplelogarsenic.c + .526891 + .24461*samplelogarsenic -
.14144 - .241134*samplelogarsenic.c
predprobeducg6assoc = exp(logitpredvalue) / (1 + exp(logitpredvalue))

plot(y=predprobeducg6assoc, x= samplearsenic, pch= 4, xlab = "Arsenic", ylab = "Predicted
probability", main = "Arsenic vs. Predicted Probability")

#plot them all on one plot with different symbols
#make the outlines of the plot without any data.
#to get the y axis to stretch from zero to one, make up a variable with 31 values

madeupy = c(0, 1, rep(.5, 29))
plot(y = madeupy, x = samplearsenic, type = "n", ylab = "Predicted probability", xlab =
"Arsenic", main = "Predicted Probability vs. Arsenic for Different Groups")

#now add the points for each category to the graph
points(y=predprobbaseline, x= samplearsenic, pch= 1)
points(y=predprobassoc, x= samplearsenic, pch= 2)
points(y=predprobeducg6, x= samplearsenic, pch= 3)
points(y=predprobeducg6assoc, x= samplearsenic, pch= 4)
```