Determining Predictors of a Mosquito-borne Disease Using Logistic Regression

Table of Contents

Abstract

This survey data set, from Table 14.3 in the textbook, is about the outbreak of a disease carried by mosquitoes. Individuals were randomly sampled from two different sectors in a particular city. The variables are:

Response: Person contracted the disease (coded as 1) or not (coded as 0)

Predictor variables: Age, socioeconomic status (Lower, Middle, Upper), sector of city (2 levels coded as 0 and 1)

The goal of the analysis is to find an optimal logistic regression model using these three predictor variables. The model could then potentially be used to help limit future outbreaks by showing where the likelihood of disease is the highest.

```r
disout.data <- read.table(file="C:/Users/iverph01/Documents/Stat
327/KutnerData/Chapter 14 Data Sets/CH14TA03.txt",header=FALSE, col.names =
c('obsnum', 'age', 'col3', 'col4', 'sector', 'disease'))

attach(disout.data)

# Print the contingency table of disease status vs. sector
ftable(disease, sector)
```

```
##          sector  0  1
## disease
## 0               49 18
## 1               10 21
```

Data Characteristics

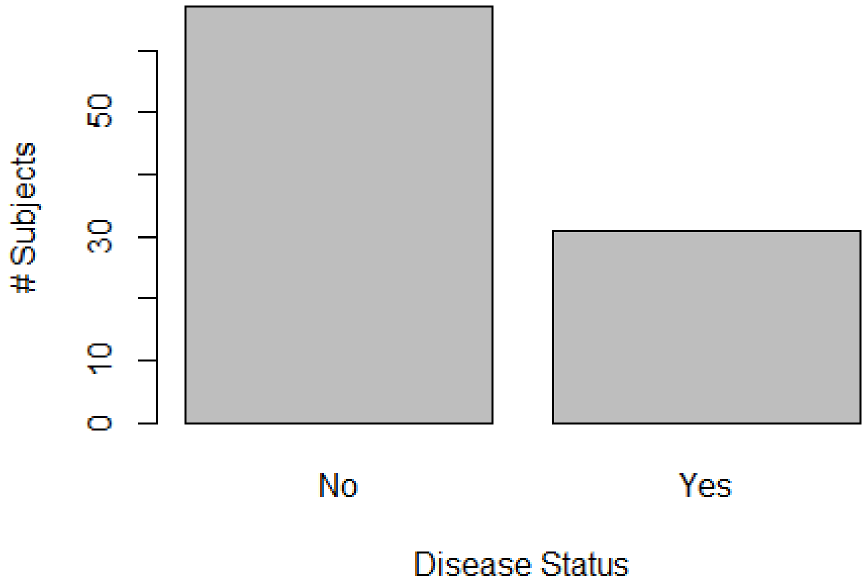
The frequency table above shows that 59 people in the sample live in sector 0, while 39 live in sector 1. 31 respondents had the disease, while 67 did not. Disease outbreak appears to be higher in sector 1 (21/39 in sector 1 vs. 10/59 in sector 0).

Here's another way to make the frequency table:

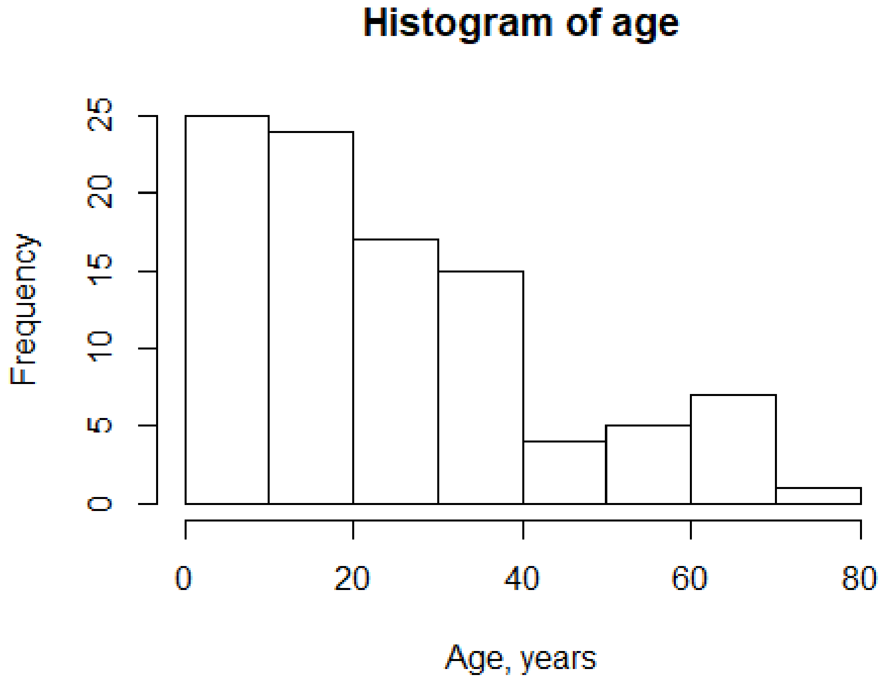
```
# One-time: install package, vcd.  Should already have MASS.

library(vcd)

## Loading required package: grid

library(MASS)
table1= xtabs(~disease+sector)
table1

##         sector
## disease  0  1
##       0 49 18
##       1 10 21

# Note: Version 1 of this file had Mosaic plots here.  They have been
removed.
# the vcd package may only be needed for mosaic plots.
# mosaicplot (table1)
```

Plots of individual variables:

```
barplot (table (ifelse (disease==1, "Yes", "No")), xlab="Disease Status",
ylab="# Subjects")
```



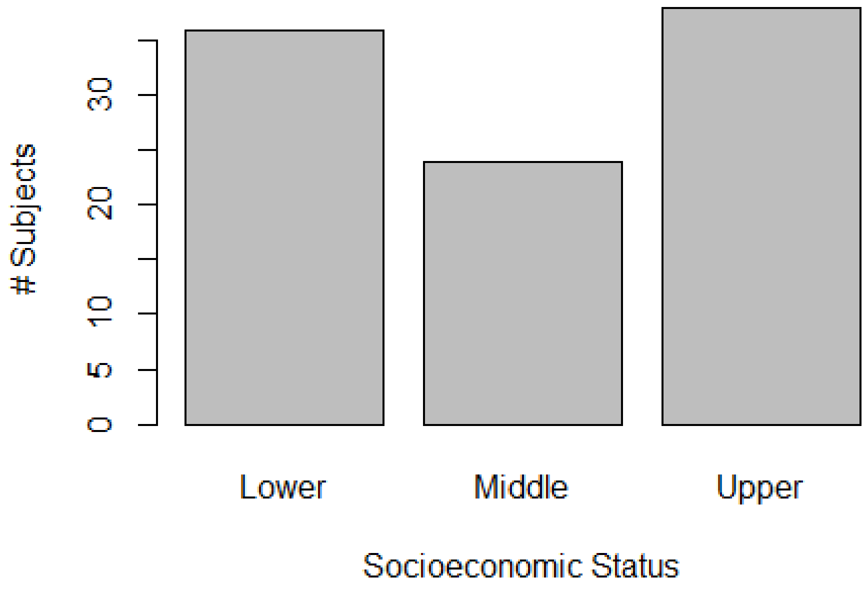
```
hist (age, xlab="Age, years")
```



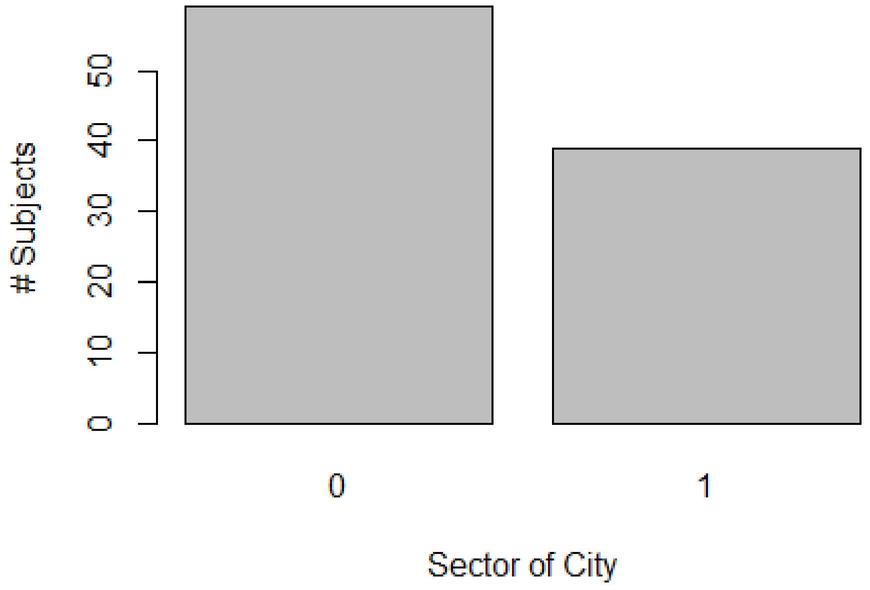
```
disout.data$socio = ifelse (col3==0, ifelse (col4==0, "Upper", "Lower"),
"Middle")
attach (disout.data)
```

```
## The following objects are masked from disout.data (pos = 6):
##
##     age, col3, col4, disease, obsnum, sector
```

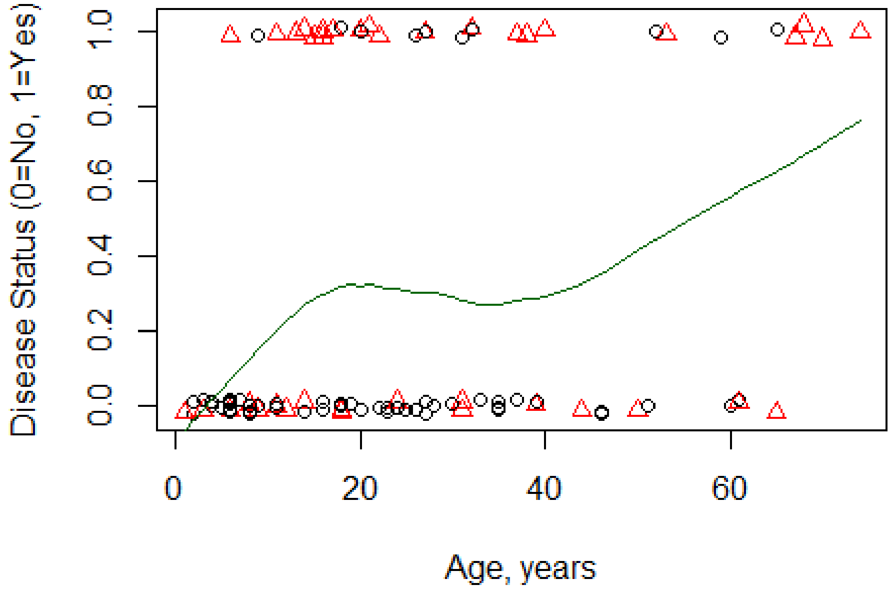
```
barplot (table (socio), xlab="Socioeconomic Status", ylab="# Subjects")
```



```
barplot (table (sector), xlab="Sector of City", ylab="# Subjects")
```



```
plot (jitter (disease, 0.1) ~ age, col=ifelse (sector==0, 1, 2), pch=ifelse
(sector==0, 1, 2), xlab="Age, years", ylab="Disease Status (0=No, 1=Yes)")
lines (lowess (age, disease), col='darkgreen')
```



None of the predictor variables require a transformation.

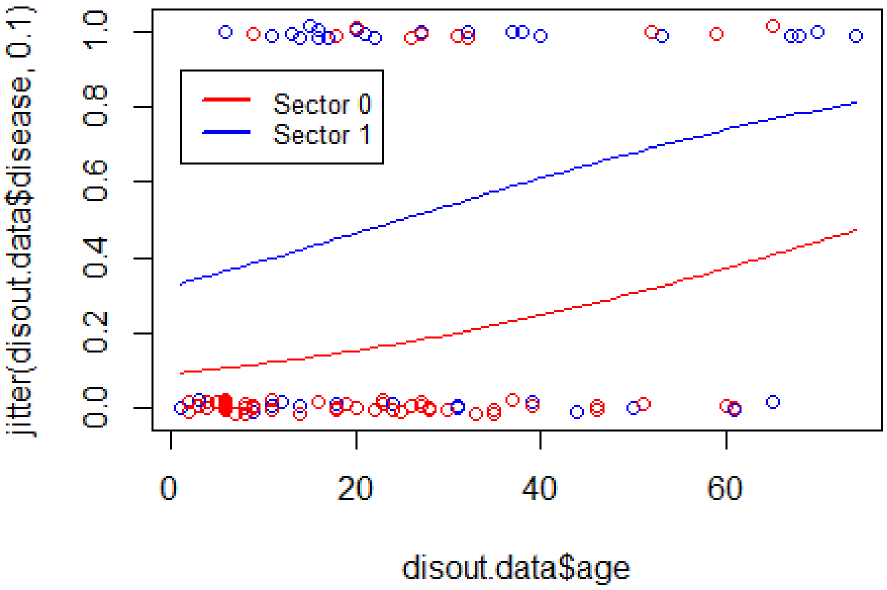
First-order logistic regression model.

A first-order model was fit using all three predictors.

```
# Fitting the regression model:
disease.logit <- glm(disease ~ age + as.factor (socio) + sector,
family=binomial)
summary(disease.logit)

##
## Call:
## glm(formula = disease ~ age + as.factor(socio) + sector, family =
binomial)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6552  -0.7529  -0.4788   0.8558   2.0977
##
## Coefficients:
##                           Estimate Std. Error z value Pr(>|z|)
## (Intercept)               -2.61819    0.61314  -4.270 1.95e-05 ***
## age                        0.02975    0.01350   2.203  0.02758 *
## as.factor(socio)Middle     0.71404    0.65371   1.092  0.27470
## as.factor(socio)Upper      0.30525    0.60413   0.505  0.61336
## sector                     1.57475    0.50162   3.139  0.00169 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 122.32  on 97  degrees of freedom
## Residual deviance: 101.05  on 93  degrees of freedom
## AIC: 111.05
##
## Number of Fisher Scoring iterations: 4

exp (disease.logit$coefficients)

##            (Intercept)                    age as.factor(socio)Middle
##              0.0729348              1.0301970              2.0422350
##  as.factor(socio)Upper                 sector
##              1.3569704              4.8295304

exp (confint (disease.logit))

## Waiting for profiling to be done...
##                             2.5 %     97.5 %
## (Intercept)            0.01934928  0.2200004
## age                    1.00400227  1.0591703
## as.factor(socio)Middle 0.56615077  7.5604424
## as.factor(socio)Upper  0.41366026  4.5463223
## sector                 1.84597726 13.3851059
```

Age and sector are significant predictors. Socioeconomic status is not. The probability of a subject having the disease is higher in older subjects and if a subject is from sector 1 (vs. 0). In particular, the odds of disease is 3% higher for each additional year of age, and 4.8 times higher in sector 1 vs. sector 0. The 95% confidence intervals for the age and sector odds ratios are, respectively, 0.4 to 5.9% higher odds of disease per year of age and 1.8 to 13.4 times higher odds of disease in sector 1 vs sector 0.

```
# plot the model.  Note: We can use the predict function to get predicted
probabilities.

plot(disout.data $age, jitter(disout.data$disease, .1), col=ifelse
(sector==0, "red", "blue"))

ageseq = seq (min(age), max(age), by=1)
fitprob0 = predict (disease.logit, data.frame (age=ageseq, socio="Upper",
sector=0), type='response')
lines (ageseq, fitprob0, col='red')

fitprob1 = predict (disease.logit, data.frame (age=ageseq, socio="Upper",
sector=1),
                    type='response')
lines (ageseq, fitprob1, col='blue')

legend(1,0.9,c('Sector 0','Sector 1'),lty=c(1,1),lwd=c(2.5,2.5), cex=0.8,
col=c("red","blue"))
```



Model selection

Stepwise regression on the first-order model

Next we apply step-wise regression to the first-order model

```
m2 = step (disease.logit, direction='both')

## Start:  AIC=111.05
## disease ~ age + as.factor(socio) + sector
##
##                     Df Deviance    AIC
## - as.factor(socio)  2   102.26 108.26
## <none>                  101.05 111.05
## - age               1   106.20 114.20
## - sector            1   111.50 119.50
##
## Step:  AIC=108.26
## disease ~ age + sector
##
##                     Df Deviance    AIC
## <none>                  102.26 108.26
## + as.factor(socio)  2   101.05 111.05
## - age               1   107.53 111.53
## - sector            1   114.91 118.91
```

```
summary (m2)
```

```
##
## Call:
## glm(formula = disease ~ age + sector, family = binomial)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.7296  -0.7048  -0.4940   0.9870   2.0929
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -2.33515    0.51113  -4.569 4.91e-06 ***
## age          0.02929    0.01317   2.224 0.026153 *
## sector       1.67345    0.48734   3.434 0.000595 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 122.32  on 97  degrees of freedom
## Residual deviance: 102.26  on 95  degrees of freedom
## AIC: 108.26
##
## Number of Fisher Scoring iterations: 4
```

```
beta2 = coefficients(m2)
exp (beta2)
```

```
## (Intercept)         age      sector
##  0.09679588  1.02972313  5.33053503
```

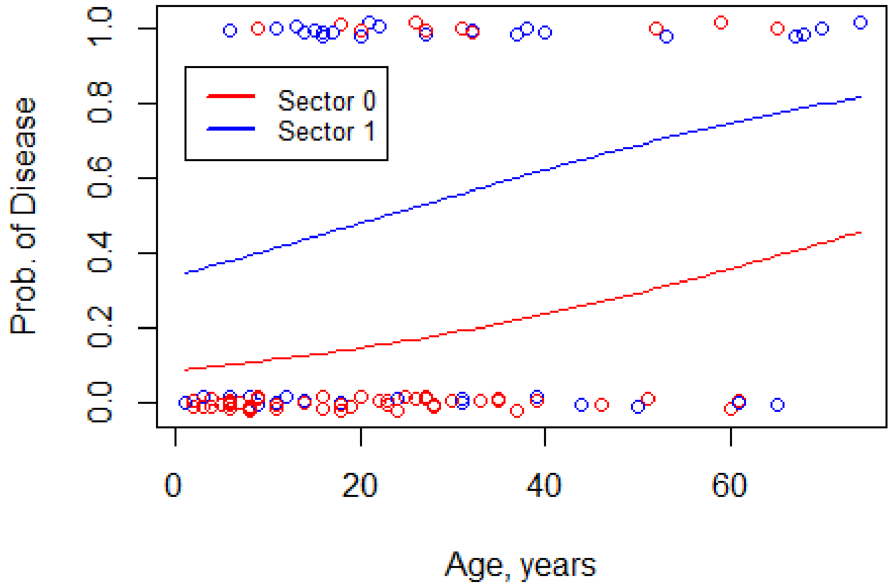
```
exp (confint (m2))
```

```
## Waiting for profiling to be done...
```

```
##                  2.5 %     97.5 %
## (Intercept) 0.03249728  0.2457814
## age         1.00422908  1.0579988
## sector      2.09905769 14.3733246
```

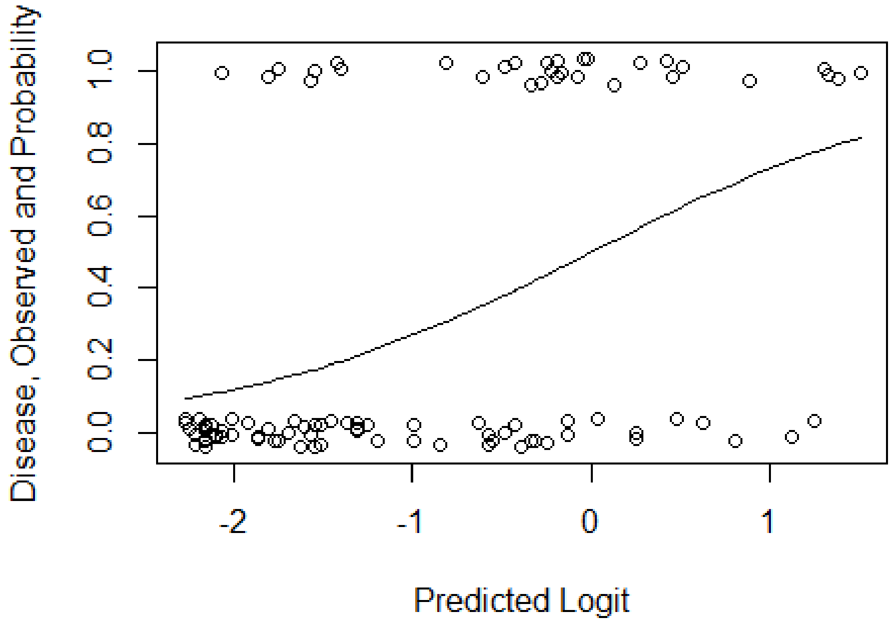
The socio-economic status predictor was removed. The reduced model has a lower AIC, so it’s a better model. The effects of age and sector are similar to the full model. The odds of disease is 0.42 to 5.8% higher per year of age, and 2.1 to 14.4 times higher in sector 1 vs sector 0.

```
ageseq = seq (min(age), max(age), by=1)
prob0 = predict (m2, data.frame (age=ageseq, sector=0), type='response')
prob1 = predict (m2, data.frame (age=ageseq, sector=1), type='response')

plot (age, jitter (disease, 0.1), col=ifelse (sector==0, 'red', 'blue'),
xlab="Age, years", ylab="Prob. of Disease")
lines (ageseq, prob0, col='red')
lines (ageseq, prob1, col='blue')

legend(1,0.9,c('Sector 0','Sector 1'),cex=0.8, lty=c(1,1),lwd=c(2.5,2.5),
col=c("red","blue"))
```



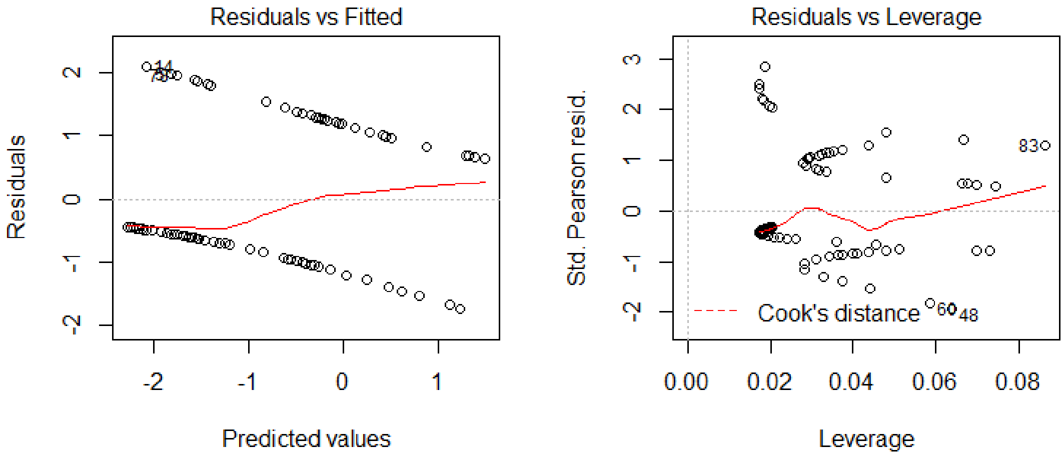
To illustrate a general plot of response vs predicted:

```
predpr = predict (m2, type='response')
predlogit = predict (m2)
plot (jitter (disease, 0.2) ~ predlogit, xlab="Predicted Logit",
      ylab="Disease, Observed and Probability")
pred.ord = order (predlogit)
lines (predlogit[pred.ord], predpr[pred.ord])
```



In the plot above, we see that the probability of disease increases with age in both sectors of the city, and the probability is higher in sector 1 than in sector 0. Since this model does not have an interaction effect between age and sector, the two fitted probability curves are parallel on the logit (i.e., the log odds) scale (not shown).

```
par (mfrow=c(1,2))
plot (m2, which=c(1,5))
```



The residual plots above for model, m2, look reasonable. In the first plot, the fitted lowess line is relatively flat and close to zero. In the second plot, there are no obvious patterns nor obvious outliers in either dimension (standardized residuals on the Y-axis or Leverage on the X-axis). The Cook's distance contours do not appear on the plot, which means there are no unusually large values for that influence measure.

Fit a model with interactions:

Next, we fit a full model with all two-way interactions.

```
disout.data$age.c = age - mean(age)
m3 = glm (disease ~ (age.c + as.factor(socio) + sector)^2, family=binomial,
data=disout.data)
summary (m3)

##
## Call:
## glm(formula = disease ~ (age.c + as.factor(socio) + sector)^2,
##     family = binomial, data = disout.data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.4692  -0.7379  -0.4194   1.0128   2.1734
##
## Coefficients:
##                              Estimate Std. Error z value Pr(>|z|)
## (Intercept)                  -1.86185    0.60005  -3.103  0.00192 **
## age.c                         0.04439    0.02417   1.837  0.06624 .
## as.factor(socio)Middle        0.65223    1.00794   0.647  0.51757
## as.factor(socio)Upper         0.09207    0.88130   0.104  0.91680
## sector                        1.37780    0.95433   1.444  0.14881
## age.c:as.factor(socio)Middle  0.09142    0.05741   1.592  0.11129
## age.c:as.factor(socio)Upper  -0.01397    0.03161  -0.442  0.65861
## age.c:sector                 -0.03421    0.03093  -1.106  0.26876
## as.factor(socio)Middle:sector 0.43046    1.52746   0.282  0.77809
## as.factor(socio)Upper:sector  0.73963    1.24892   0.592  0.55371
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 122.318  on 97  degrees of freedom
## Residual deviance:  93.996  on 88  degrees of freedom
## AIC: 114
##
## Number of Fisher Scoring iterations: 5

exp (m3$coefficients)
```

```
##                   (Intercept)                         age.c
##                     0.1553856                     1.0453896
##        as.factor(socio)Middle         as.factor(socio)Upper
##                     1.9198104                     1.0964428
##                        sector  age.c:as.factor(socio)Middle
##                     3.9661576                     1.0957245
##   age.c:as.factor(socio)Upper                  age.c:sector
##                     0.9861297                     0.9663704
## as.factor(socio)Middle:sector  as.factor(socio)Upper:sector
##                     1.5379623                     2.0951552
```

```
exp (confint (m3))
```

```
## Waiting for profiling to be done...
```

```
##                                    2.5 %     97.5 %
## (Intercept)                   0.03821056  0.4386551
## age.c                         0.99756029  1.1001085
## as.factor(socio)Middle        0.23348667 14.0624912
## as.factor(socio)Upper         0.17617882  6.4090195
## sector                        0.58930274 28.3186206
## age.c:as.factor(socio)Middle  0.99408741  1.2586818
## age.c:as.factor(socio)Upper   0.92555556  1.0511542
## age.c:sector                  0.90748080  1.0271886
## as.factor(socio)Middle:sector 0.08574751 42.7489106
## as.factor(socio)Upper:sector  0.18033977 26.4625311
```

None of the predictors are significant at $p < 0.05$ in the full model with interactions. This is probably due to collinearity.

Stepwise regression

Next, we run a step-wise analysis on the full interaction model.

```
# Stepwise with AIC criterion
m.both = step (m3, directon='both')
```

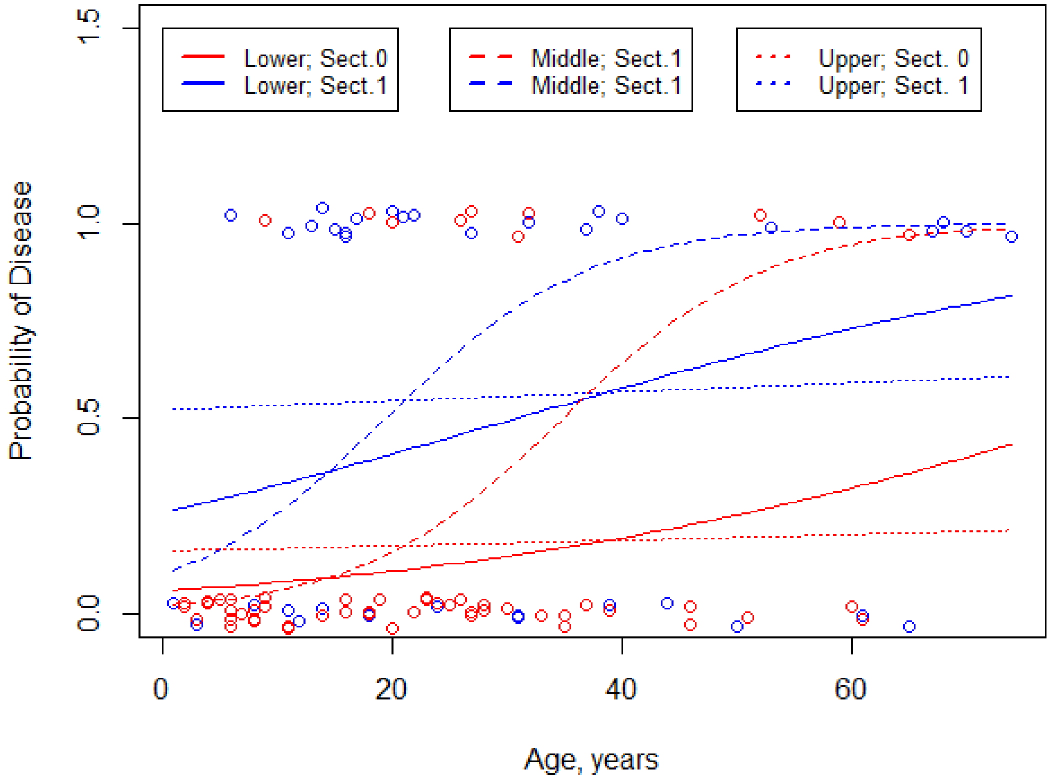
```
## Start:  AIC=114
## disease ~ (age.c + as.factor(socio) + sector)^2
## 
##                            Df Deviance    AIC
## - as.factor(socio):sector  2   94.349 110.35
## - age.c:sector             1   95.223 113.22
## <none>                         93.996 114.00
## - age.c:as.factor(socio)   2   99.351 115.35
## 
## Step:  AIC=110.35
## disease ~ age.c + as.factor(socio) + sector + age.c:as.factor(socio) + 
##     age.c:sector
## 
##                           Df Deviance    AIC
## - age.c:sector            1   95.469 109.47
## <none>                        94.349 110.35
## - age.c:as.factor(socio)  2   99.834 111.83
##
## Step:  AIC=109.47
## disease ~ age.c + as.factor(socio) + sector + age.c:as.factor(socio)
##
##                          Df Deviance    AIC
## <none>                        95.469 109.47
## - age.c:as.factor(socio)  2  101.054 111.05
## - sector                  1  107.121 119.12
```

The step-wise procedure retained the predictors, age, socioeconomic status, sector and the age by socioeconomic status interaction effect.

```
# Final model summary
summary (m.both)

##
## Call:
## glm(formula = disease ~ age.c + as.factor(socio) + sector +
age.c:as.factor(socio),
##     family = binomial, data = disout.data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6960  -0.6664  -0.4469   1.0046   2.1410
##
## Coefficients:
##                               Estimate Std. Error z value Pr(>|z|)
## (Intercept)                   -1.93879    0.50813  -3.816 0.000136 ***
## age.c                          0.03422    0.02204   1.552 0.120589
## as.factor(socio)Middle         0.84038    0.70976   1.184 0.236400
## as.factor(socio)Upper          0.39859    0.61172   0.652 0.514668
## sector                         1.74436    0.53614   3.254 0.001140 **
## age.c:as.factor(socio)Middle  0.07901    0.05297   1.491 0.135831
## age.c:as.factor(socio)Upper  -0.02948    0.02929  -1.006 0.314192
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 122.318  on 97  degrees of freedom
## Residual deviance:  95.469  on 91  degrees of freedom
## AIC: 109.47
##
## Number of Fisher Scoring iterations: 5
```

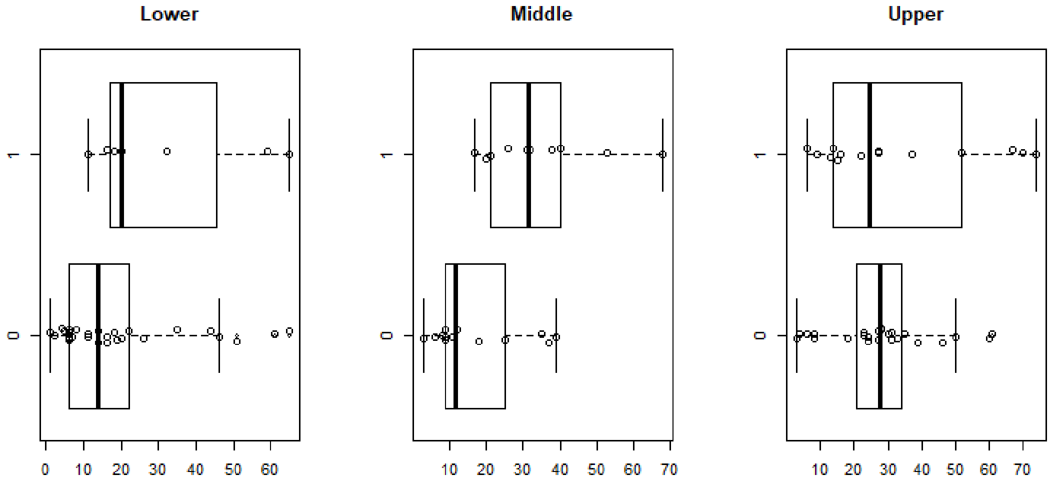
It is interesting that only sector is statistically significant in the final model. It is also worth noting that the final step-wise model has an AIC value of 109.57, and the earlier model with just age and sector has an AIC of 108.26. Thus, the earlier model with just age and sector is actually better based on AIC. However, for purposes of illustration in this example report, I will keep the step-wise model with the age-by-socioeconomic status interaction effect.

```
par (mfrow=c(1,1))
plot(disout.data$age, jitter(disout.data$disease, .2), col=ifelse (sector==0,
"red", "blue"), ylim=c(0,1.5), xlab="Age, years", ylab="Probability of
Disease")

ageseq = seq (min(age), max(age), by=1)
fit.lower.0 = predict (m.both, data.frame (age.c=ageseq-mean(age),
socio="Lower", sector=0),
                       type='response')
lines (ageseq, fit.lower.0, col='red', lty=1)

fit.lower.1 = predict (m.both, data.frame (age.c=ageseq-mean(age),
socio="Lower", sector=1),
                       type='response')
lines (ageseq, fit.lower.1, col='blue', lty=1)

fit.middle.0 = predict (m.both, data.frame (age.c=ageseq-mean(age),
socio="Middle", sector=0),
                       type='response')
lines (ageseq, fit.middle.0, col='red', lty=2)

fit.middle.1 = predict (m.both, data.frame (age.c=ageseq-mean(age),
socio="Middle", sector=1),
                        type='response')
lines (ageseq, fit.middle.1, col='blue', lty=2)

fit.upper.0 = predict (m.both, data.frame (age.c=ageseq-mean(age),
socio="Upper", sector=0),
                       type='response')
lines (ageseq, fit.upper.0, col='red', lty=3)

fit.upper.1 = predict (m.both, data.frame (age.c=ageseq-mean(age),
socio="Upper", sector=1),
                        type='response')
lines (ageseq, fit.upper.1, col='blue', lty=3)

legend(0, 1.5,c('Lower; Sect.0','Lower; Sect.1'), lty=c(1,1), lwd=c(2.5,2.5),
cex=0.8,
       col=c("red","blue"))

legend(25,1.5,c('Middle; Sect.1','Middle; Sect.1'), lty=c(2,2),
lwd=c(2.5,2.5), cex=0.8,
       col=c("red","blue"))

legend(50,1.5,c('Upper; Sect. 0','Upper; Sect. 1'), lty=c(3,3),
lwd=c(2.5,2.5), cex=0.8,
       col=c("red","blue"))
```



We can see the age-by-socioeconomic status interaction effect in the plot above. In the Lower status group, the probability of disease rises gradually with age. In the Middle status group, the probability of disease rises quickly and reaches maximum probability around age 45 in sector 1 and age 65 in sector 0. In the Upper status group, the probability of disease does not depend on age.

```
par (mfrow=c(1,3))
boxplot (age [socio=="Lower"] ~ disease [socio=="Lower"], horizontal=T,
main="Lower")
points (age [socio=="Lower"], jitter (disease [socio=="Lower"], 0.2) + 1)

boxplot (age [socio=="Middle"] ~ disease [socio=="Middle"], horizontal=T,
main="Middle")
points (age [socio=="Middle"], jitter (disease [socio=="Middle"], 0.2) + 1)

boxplot (age [socio=="Upper"] ~ disease [socio=="Upper"], horizontal=T,
main="Upper")
points (age [socio=="Upper"], jitter (disease [socio=="Upper"], 0.2) + 1)
```

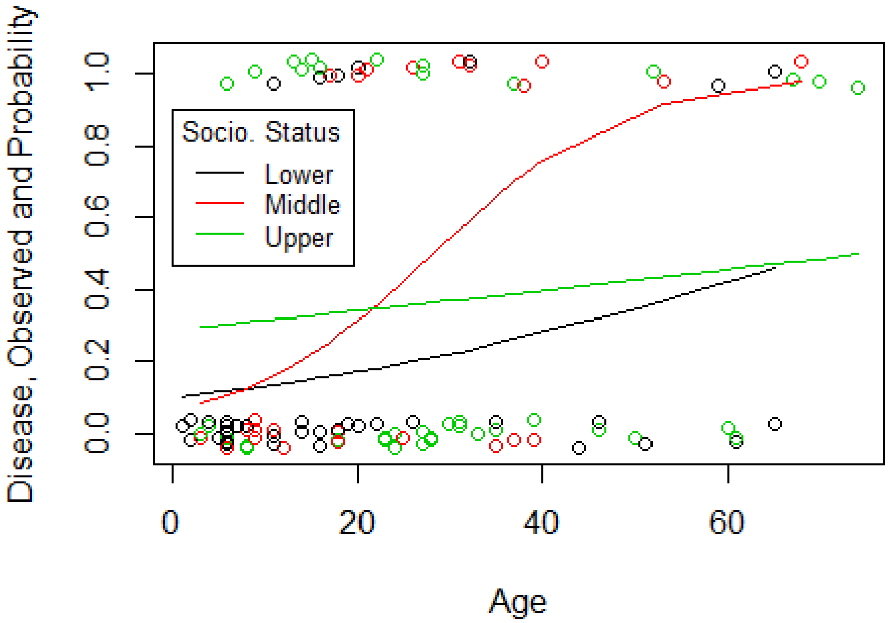


The plots above may help explain why the predicted probability curve for the Middle socioeconomic status rises the fastest with increasing age. In the Middle category, the age distribution among non-diseased individuals is shifted more to the left, with no individuals above age 40 that do not have disease. Thus, the age distribution among the diseased individuals in the Middle group is shifted more clearly shifted to the right relative the non-disease group.

Illustrate an interaction plot for 2 predictors, regardless of the total number of predictors. In this case, we are plotting the interaction effect between age and socioeconomic status.

```
par (mfrow=c(1,1))
plot (jitter (disease, 0.2) ~ age, col=as.factor(socio), xlab="Age",
      ylab="Disease, Observed and Probability")

# Generate separate logistic fits of cnt vs age for each socio status

fit.lower = glm (disease[socio=="Lower"] ~ age[socio=="Lower"],
family=binomial)
fit.middle = glm (disease[socio=="Middle"] ~ age[socio=="Middle"],
family=binomial)
fit.upper = glm (disease[socio=="Upper"] ~ age[socio=="Upper"],
family=binomial)

# Save a list of indices for each socio status that will put the age values
# in order from smallest to largest

age.ord.L = order (age [socio=="Lower"])
age.ord.M = order (age [socio=="Middle"])
age.ord.U = order (age [socio=="Upper"])

# Add lines plotting the logistic fit vs age for each socio status
```

```
lines (age[socio=="Lower"][age.ord.L],  predict (fit.lower,
type='response')[age.ord.L], col=1)
lines (age[socio=="Middle"][age.ord.M], predict (fit.middle,
type='response')[age.ord.M], col=2)
lines (age[socio=="Upper"][age.ord.U],  predict (fit.upper,
type='response')[age.ord.U], col=3)

legend (0, 0.9, list("Lower", "Middle", "Upper"), lty=rep(1, 3), col=1:3,
        title="Socio. Status", cex=0.8)
```



The next table shows the odds ratios for the parameter estimates and their standard errors.

```
cbind.data.frame (exp.beta = exp (m.both$coefficients),
                  exp (as.data.frame (confint(m.both))))

## Waiting for profiling to be done...

##                               exp.beta      2.5 %    97.5 %
## (Intercept)                  0.1438781 0.04765466  0.359653
## age.c                        1.0348107 0.99089001  1.082703
## as.factor(socio)Middle       2.3172520 0.57712633  9.716790
## as.factor(socio)Upper        1.4897205 0.44859704  5.115541
## sector                       5.7222114 2.07173422 17.306720
## age.c:as.factor(socio)Middle 1.0822103 0.98497835  1.216219
## age.c:as.factor(socio)Upper  0.9709504 0.91556489  1.028624
```

We cannot directly interpret the age or socioeconomic effects, because they are involved in an interaction. Regarding sector, a person in sector 1 has a 5.7 times higher odds of disease compared to sector 0, with 95% confidence between 2.1 and 17.3 times higher.

Model Diagnostics

```
#------------------------Deviance test of lack of fit--------------------

# First model
pchisq(deviance(disease.logit), df.residual(disease.logit), lower=F)

## [1] 0.2666791

# Final model
pchisq(deviance(m.both), df.residual(m.both), lower=F)

## [1] 0.3537353
```

There is no significant lack of fit in either the first model or the final model (p > 0.05).

```
#Likelihood Ratio (LR) test statistic and P-value in R (multiple logistic
regression):

# First model

1 - pchisq(disease.logit$null.deviance - disease.logit$deviance,
           disease.logit$df.null - disease.logit$df.residual)

## [1] 0.0002807651

# Final model

1 - pchisq(m.both$null.deviance - m.both$deviance,
           m.both$df.null - m.both$df.residual)

## [1] 0.0001545938
```

Both the first and final models have significant effects on disease status (p < 0.05).

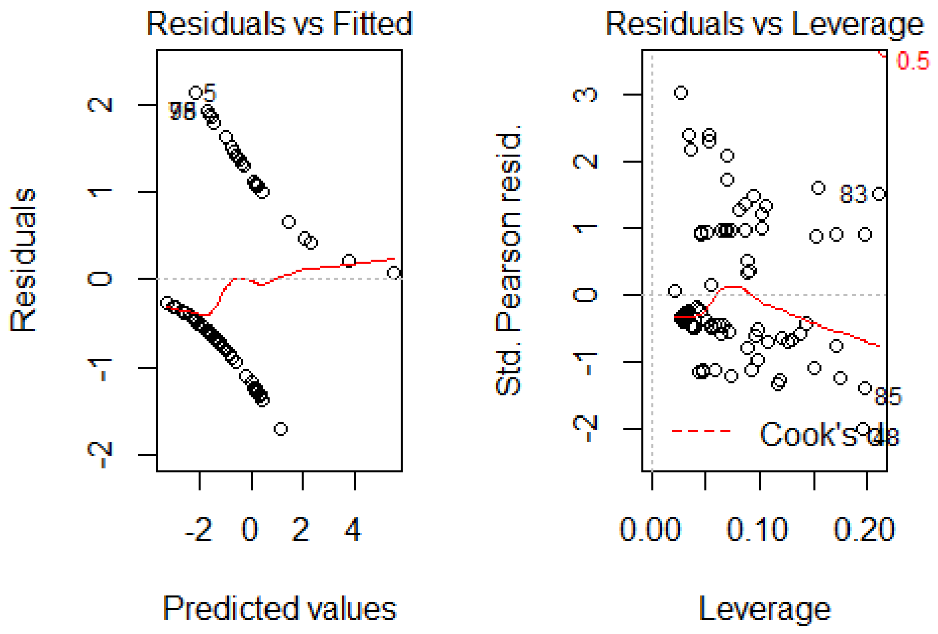
```
# Compare the final model with the reduced model (age + sector)

1 - pchisq(m2$deviance - m.both$deviance,
           m2$df.residual - m.both$df.residual)

## [1] 0.1474076
```

The reduced model, m2, is not significantly worse the the best model with interaction.

Residual plots.

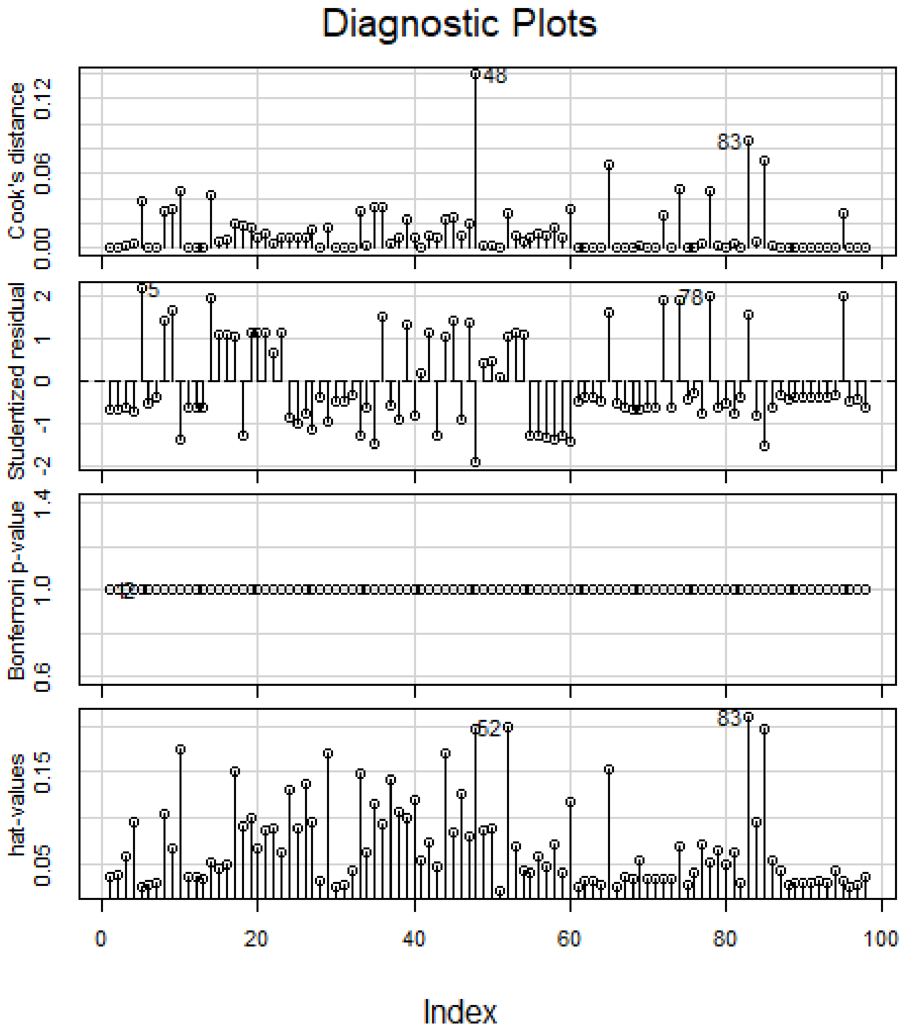
Following are residual plots for the final model.

```
# Residual plot
par (mfrow=c(1,2))

# The plot function applied to the fitted model object, selecting plots 1 and
5,
# provides the necessary plots.
plot (m.both, which = c(1,5))
```



The Residuals vs Fitted plot provides evidence of a reasonable fit, since the red Lowess line is relatively flat and close to zero. The residuals vs leverage plot also indicates a reasonable fit. There are no unusually high leverage values (on the X-axis). There are no large standardized residuals (outside the range of +/- 4 on the Y-axis). And there are no points outside of the 0.5 Cook's distance contour lines.

```
# Influence diagnostics

library(car)

## Loading required package: carData

influenceIndexPlot(m.both)
```



Rows 48 and 83 have relatively high Cook's distance, but they are not above the 0.5 cutoff. Rows 52 and 83 have the highest leverage values, but they are not obvious outliers with respect to the rest of the leverage values, and they are plot the $2\sqrt{p/n}$ cutoff, which is 0.49.

Conclusion

ROC Curve

```
# ROC curve - install package ROCR
par (mfrow=c(1,1))
library(ROCR)

## Loading required package: gplots
```

```
##
## Attaching package: 'gplots'
```

```
## The following object is masked from 'package:stats':
##
##     lowess
```

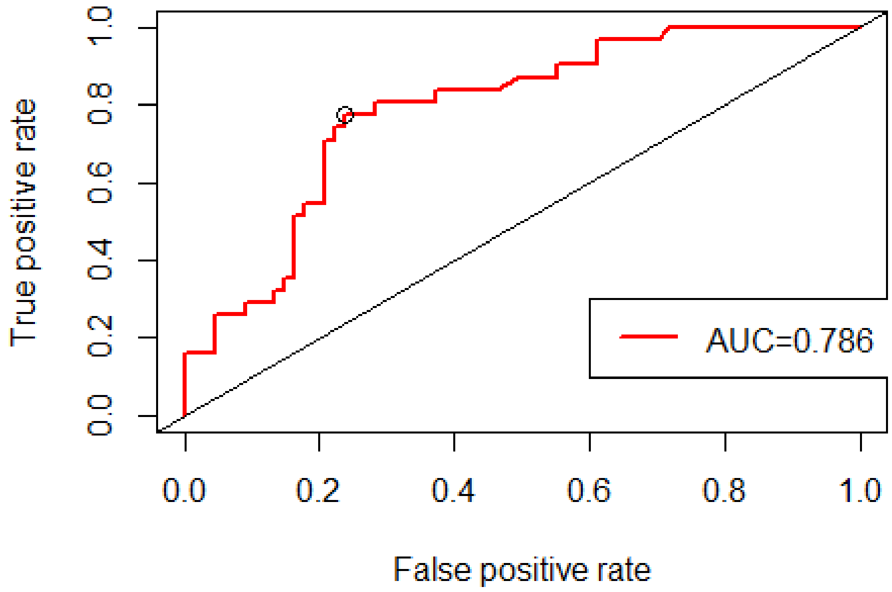
```
pred1 <- prediction(m.both$fitted.values, m.both$y)
perf1 <- performance(pred1,"tpr","fpr")
auc1 <- performance(pred1,"auc")@y.values[[1]]
auc1
```

```
## [1] 0.7859894
```

```
plot(perf1, lwd=2, col=2)
abline(0,1)
legend(0.6, 0.3, c(paste ("AUC=", round (auc1, 4), sep="")),   lwd=2, col=2)

# Extract the X and Y values from the ROC plot, as well as the probability
cutoffs
roc.x = slot (perf1, "x.values") [[1]]
roc.y = slot (perf1, "y.values") [[1]]
cutoffs = slot (perf1, "alpha.values") [[1]]

auc.table = cbind.data.frame(cutoff=pred1@cutoffs,
                             tp=pred1@tp, fp=pred1@fp, tn=pred1@tn,
fn=pred1@fn)
names (auc.table) = c("Cutoff", "TP", "FP", "TN", "FN")
auc.table$sensitivity = auc.table$TP / (auc.table$TP + auc.table$FN)
auc.table$specificity = auc.table$TN / (auc.table$TN + auc.table$FP)
auc.table$FalsePosRate = 1 - auc.table$specificity
auc.table$sens_spec = auc.table$sensitivity + auc.table$specificity

# Find the row(s) in the AUC table where sensitivity + specificity is
maximized

auc.best = auc.table [auc.table$sens_spec == max (auc.table$sens_spec),]
auc.best
```

```
##       Cutoff TP FP TN FN sensitivity specificity FalsePosRate sens_spec
## 65 0.3138152 24 16 51  7   0.7741935    0.761194     0.238806  1.535388
```

```
# Plot the maximum point(s) on the ROC plot

points (auc.best$FalsePosRate, auc.best$sensitivity, cex=1.3)
```



The ROC curve suggests the predictive ability of this model is better than random guessing, since the AUC (0.786) is larger than 0.5. The optimal cutoff for classification is a fitted probability of 0.3138, which has a false positive rate (1 - specificity) of 0.24, and a true positive rate (sensitivity) of 0.77. That point is shown as a black circle on the ROC curve.

Here are some example predictions, looking at ages in the 20s or 50-60s in each of the two sectors and three socioeconomic classes:

```
# With logistic regression, the predict function does not provide confidence
limits, even
# with the interval= option.  Instead, we request the se.fit=T option and
calculate our own
# limits on the logist scale, and then back-transform to the probability
scale.

preds = predict (m.both, se.fit = T)
pred.df = cbind.data.frame (disout.data, as.data.frame (preds))

pred.df$lwr = pred.df$fit - 1.96 * pred.df$se.fit
pred.df$upr = pred.df$fit + 1.96 * pred.df$se.fit

pred.df$fit.pr = round (exp (pred.df$fit) / (1 + exp (pred.df$fit)), 3)
pred.df$lwr.pr = round (exp (pred.df$lwr) / (1 + exp (pred.df$lwr)), 3)
pred.df$upr.pr = round (exp (pred.df$upr) / (1 + exp (pred.df$upr)), 3)
```

```
# Selected subjects in their 20's
pred.df [c(30,47,78,21,16,11), c(2,5:7,14:16)]
```

```
##    age sector disease  socio fit.pr lwr.pr upr.pr
## 30  20      0       0  Lower  0.107  0.040  0.256
## 47  20      1       1  Lower  0.408  0.180  0.683
## 78  20      0       1 Middle  0.156  0.051  0.391
## 21  21      1       1 Middle  0.542  0.271  0.791
## 16  22      1       1  Upper  0.547  0.332  0.746
## 11  23      0       0  Upper  0.175  0.073  0.364
```

```
# Selected subjects in their 50's - 60's
pred.df [c(84, 48, 41, 51, 74, 58), c(2,5:7,14:16)]
```

```
##    age sector disease  socio fit.pr lwr.pr upr.pr
## 84  51      0       0  Lower  0.258  0.079  0.584
## 48  65      1       0  Lower  0.763  0.294  0.961
## 41  53      1       1 Middle  0.978  0.657  0.999
## 51  68      1       1 Middle  0.996  0.728  1.000
## 74  52      0       1  Upper  0.196  0.062  0.474
## 58  50      1       0  Upper  0.580  0.321  0.801
```

The tables above show some example predicted probabilities and their confidence intervals for subjects in their 20s or 50-60s for age.

The following table summarizes the observed and predicted classifications of disease:

```
pred.df$pred.dis = ifelse (pred.df$fit.pr >= auc.best$Cutoff[1], "Pred.Yes",
"Pred.No")
table (pred.df$disease, pred.df$pred.dis)
```

```
##
##     Pred.No Pred.Yes
##   0      51       16
##   1       7       24
```

We have shown that the probability of mosquito-borne disease can be predicted from a person's age, socioeconomic status and whether they live in sector 1 or 0 in the city of _____. These results could potentially be used to help control the incidence of disease in the future by targeting those groups that are at higher risk of disease.

Some follow-up questions to this analysis could include:

1. Why did the Middle socioeconomic group have the fastest-rising predicted probability of disease with age? Was it because there were so few older subjects in the Middle group?

2. Are there other factors that could be measured, such as level of education or type of job, that could improve the predictability of the model?

3. Could an education program be created that would target older subjects and teach how to protect themselves against mosquito-borne disease? Then a follow-up study could be conducted to test the effectiveness of that program to reduce the probability of future disease outbreak.