5.1.1 - R: Logistic Regression

Stat 5100: Dr. Bean

Example: (Text Table 14.3) Individuals were randomly sampled within two sectors of a city, and checked for presence of disease (here, spread by mosquitoes). Subjects' age (in years), socioeconomic status (low, medium, high), and city sector are to be used to predict the probability of contracting the disease.

In R, we can create logistic regression models with the glm function. "GLM" stands for generalized linear model, and can be used to fit a variety of linear models. To specify logistic regression, we set an option in side the glm function that specifies a binomial (two classes) response.

Fit a logistic regression model

```
# Input the data
library(stat5100)
data(outbreak)

# Some of the things we do below in the document work better when these
# variables are treated numerically.
# Best practice for converting factor to numeric is to convert to a
# character variable first.
outbreak$SES_mid <- as.numeric(as.character(outbreak$SES_mid))
outbreak$SES_low <- as.numeric(as.character(outbreak$SES_low))
outbreak$sector <- as.numeric(as.character(outbreak$sector))

# To do logistic regression, we use the glm function.
outbreak_logreg <- glm(disease ~ age + SES_mid + SES_low + sector,
                       data = outbreak, family = "binomial")
summary(outbreak_logreg)
```

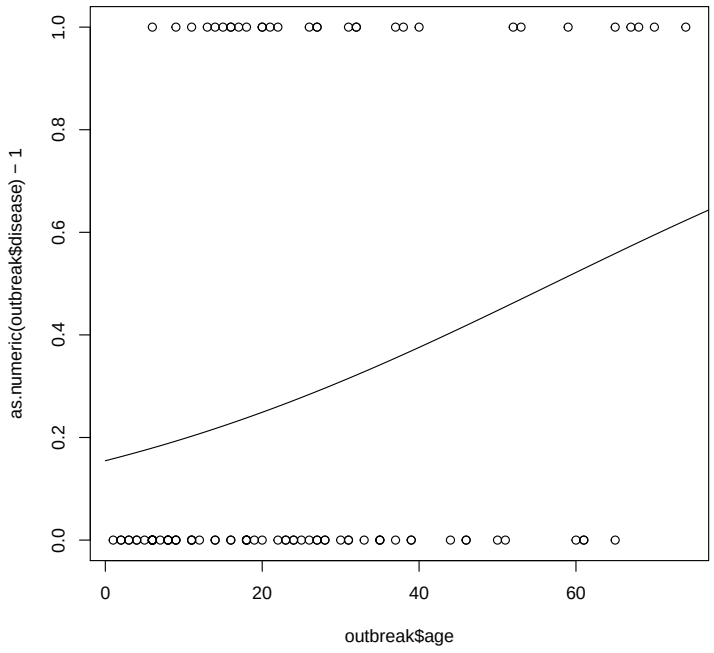
```
##
## Call:
## glm(formula = disease ~ age + SES_mid + SES_low + sector, family = "binomial",
##     data = outbreak)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6552  -0.7529  -0.4788   0.8558   2.0977
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -2.31293    0.64259  -3.599 0.000319 ***
## age          0.02975    0.01350   2.203 0.027577 *
## SES_mid      0.40879    0.59900   0.682 0.494954
## SES_low     -0.30525    0.60413  -0.505 0.613362
## sector       1.57475    0.50162   3.139 0.001693 **
## ---
## 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
```

Plot a graph of observed values and predicted probabilities

Because we have multiple predictor variables, we have to choose one variable to put on the x-axis, and the rest of the variables will have to be fixed at some value. We will use Age for our x-axis variable. We will set SES_mid = 0.245, SES_low = 0.367, and Sector = 0.398.

```
# Get a range of ages, and then predict the probability with the predict()
# function to get the shape of the predicted probability curve.
age_range <- seq(0, 80, length = 500)
npred <- length(age_range)

pred_data <- data.frame(age = age_range, SES_mid = rep(0.245, npred),
                        SES_low = rep(0.367, npred),
                        sector = rep(0.398, npred))
pred_disease <- predict(outbreak_logreg, newdata = pred_data, type = "response")

plot(outbreak$age, as.numeric(outbreak$disease) - 1)
lines(age_range, pred_disease)
```



Goodness of fit tests

Note that the Hosmer-Lemeshow test statistic and p-value are partially dependent on the number of groups in which the observations are organized. The default number of groups is 10 but different software programs

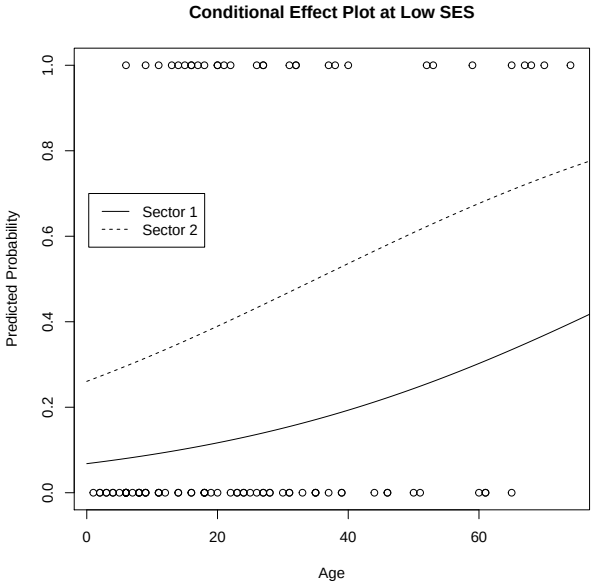
have different rules for creating the groups. For this reason, it is difficult if not impossible to get the results from this test to exactly match a similar implementation in SAS.

```
# Conduct a Hosmer and Lemeshow Goodness-of-Fit Test
# To do the test, we first need a vector of the predicted probabilities
# from the logistic regression model.
outbreak_pred <- fitted(outbreak_logreg)

# Now actually perform the test
ResourceSelection::hoslem.test(x = outbreak_logreg$y,
                               y = outbreak_logreg$fitted.values)

##
##  Hosmer and Lemeshow goodness of fit (GOF) test
##
## data:  outbreak_logreg$y, outbreak_logreg$fitted.values
## X-squared = 5.3267, df = 8, p-value = 0.7222
```

Create a conditional effects plot

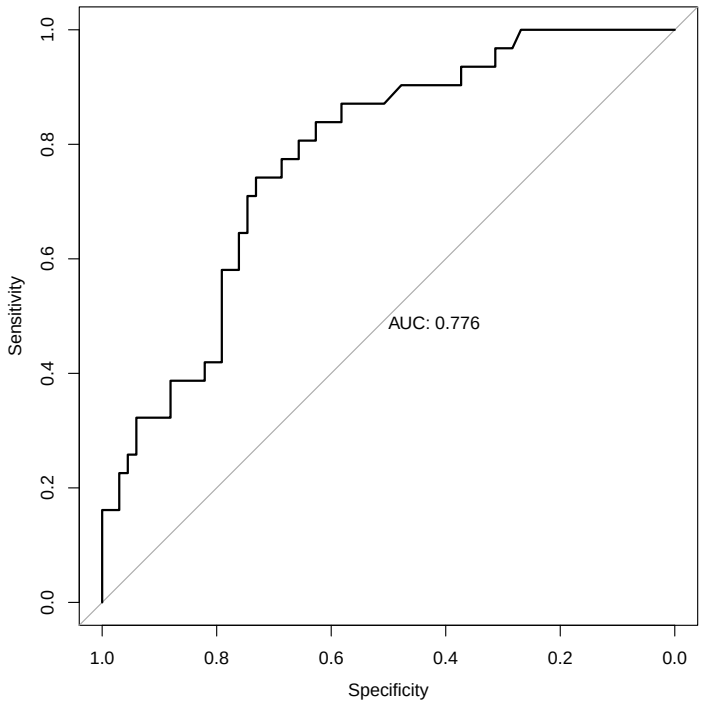
Compare the predicted disease probabilities for sector 1 (displayed as sector=0) vs sector 2 (displayed as sector=1) at low socioeconomic status (SES_mid=0, SES_low=1) as a function of age. Note that the numbers plugged in below come from the summary of the fitted logistic regression model.

```
age_range <- seq(0, 80, length = 500)

prob_1 <- 1/(1 + exp(-(-2.3127 + 0.0297*age_range +
                         0.4088*0 - 0.3051*1 + 1.5746*0)))
prob_2 <- 1/(1 + exp(-(-2.3127 + 0.0297*age_range +
                         0.4088*0 - 0.3051*1 + 1.5746*1)))

plot(outbreak$age, as.numeric(outbreak$disease) - 1,
     xlab = "Age", ylab = "Predicted Probability",
     main = "Conditional Effect Plot at Low SES")
lines(age_range, prob_1, lty = 1)
lines(age_range, prob_2, lty = 2)
legend(xy.coords(0.3, 0.7), legend = c("Sector 1", "Sector 2"),
       lty = 1:2, cex = 1.0)
```


Conditional Effect Plot at Low SES
Predicted Probability
0.0
0.2
0.4
0.6
0.8
1.0
Sector 1
Sector 2
0
20
40
60
Age

Plot an ROC curve

```
# ROC Curve
prob <- fitted(outbreak_logreg)
pROC::roc(outbreak$disease ~ prob, plot = TRUE, print.auc = TRUE)

## Setting levels:  control = 0, case = 1
## Setting direction:  controls < cases
```




```
##
## Call:
## roc.formula(formula = outbreak$disease ~ prob, plot = TRUE, print.auc = TRUE)
##
## Data: prob in 67 controls (outbreak$disease 0) < 31 cases (outbreak$disease 1).
## Area under the curve: 0.7764
```

Check for multicollinearity

We can get the variance inflation factors for the logistic model with the vif() function inside the car package:

```
car::vif(outbreak_logreg)

##      age  SES_mid  SES_low   sector
## 1.023343 1.221073 1.255794 1.047822
```

Variable selection

In R, we can perform variable selection for logistic regression by using the stepAIC() function available in the MASS package.

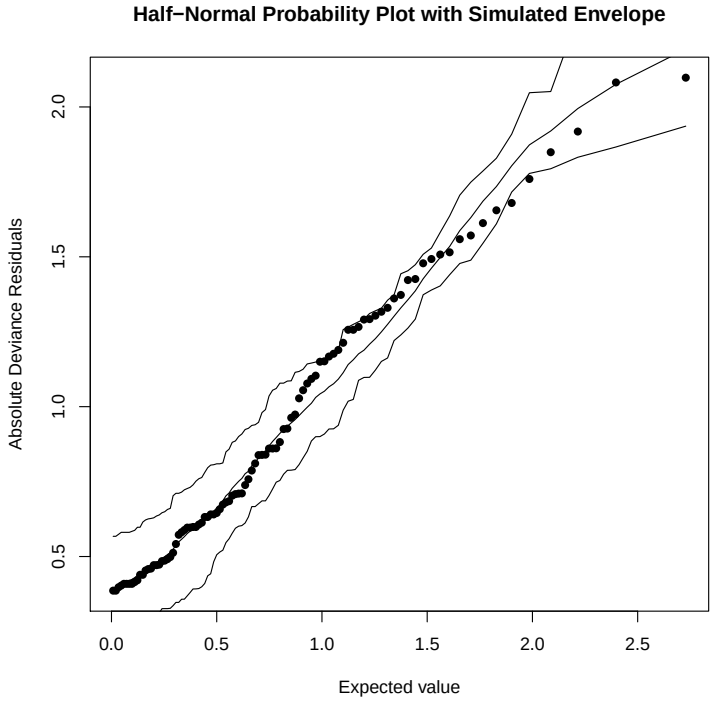
```
MASS::stepAIC(outbreak_logreg, trace = FALSE)

##
## Call:  glm(formula = disease ~ age + sector, family = "binomial", data = outbreak)
##
## Coefficients:
## (Intercept)          age       sector
##    -2.33515      0.02929      1.67345
##
## Degrees of Freedom: 97 Total (i.e. Null);  95 Residual
## Null Deviance:     122.3
## Residual Deviance: 102.3  AIC: 108.3
```

Based upon the above, the so-called "best model" would be the model that includes age and sector as predictor variables.

Simulated envelope

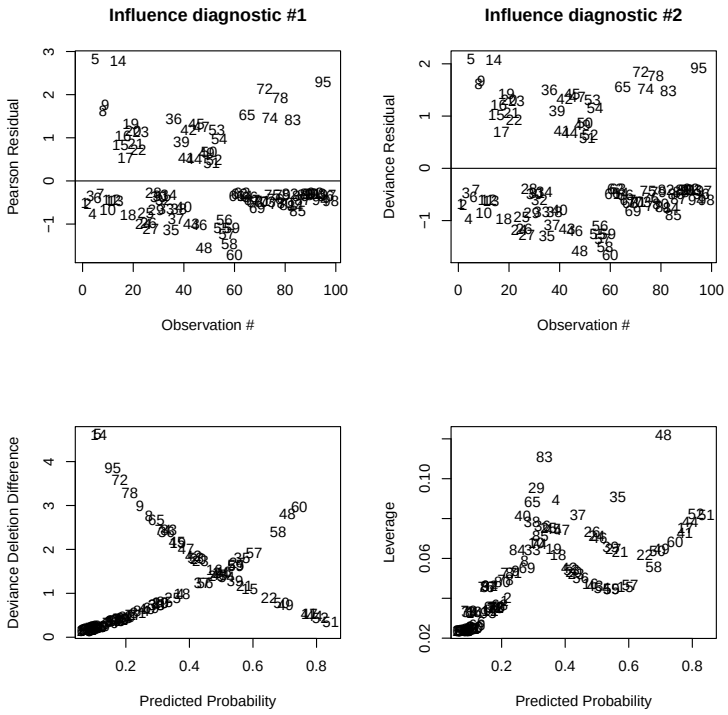
Check for outliers using the half-normal probability plot with simulated envelope.

```
# Set a seed first because this does use some randomization
set.seed(1741)

stat5100::simulated_envelope_logreg(outbreak_logreg)
```



Check for influential observations

```
stat5100::logistic_influence_diagnostics(outbreak_logreg)
```



Look at a suspect observation

```
outbreak[outbreak$case == 48, ]

##    case age SES_mid SES_low sector disease
## 48   48  65       0       1      1       0
```