Homework 4

Austin Thrash

2023-12-03

Exercise 1

```
setwd('~/Downloads')
liver <- read.csv(
  'liver-1 (1).csv',
  header = TRUE
)
sleep <- read.csv(
  'sleep-1 (1).csv',
  header = TRUE
)
```

(a)

For only females in the data set, find and specify the best set of predictors via stepwise selection with AIC criteria for a logistic regression model predicting whether a female is a liver patient.

```
liver_female <- liver %>%
  filter(Gender == 'Female')

full.model <- glm(LiverPatient ~ Age + TB + Alkphos + Alamine + Aspartate + TP + ALB, data = liver_fema

step.model <- step(full.model, direction = "both",
                      trace = FALSE)
summary(step.model)
```

```
##
## Call:
## glm(formula = LiverPatient ~ TB + Aspartate, family = "binomial",
##     data = liver_female)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.493351   0.374587  -1.317   0.1878
## TB           0.466267   0.300636   1.551   0.1209
## Aspartate    0.011356   0.006125   1.854   0.0637 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 175.72  on 134  degrees of freedom
## Residual deviance: 155.02  on 132  degrees of freedom
```

```
## AIC: 161.02
##
## Number of Fisher Scoring iterations: 7
```

We can see that from our results Aspartate is the best predictor for our liver patient model.

(b)

Comment on the significance of parameter estimates under significance level alpha=0.1, what Hosmer-Lemeshow's test tells us about goodness of fit and point out any issues with diagnostics by checking residual plots and cook's distance plot (with cut-off 0.25).

```
cook.d = cooks.distance(step.model)
round(cook.d, 2)
```

```
##    1    2    3    4    5    6    7    8    9   10   11   12   13   14   15   16
## 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.06 0.06 0.00 0.01 0.00 0.00 0.00 0.01 0.00
##   17   18   19   20   21   22   23   24   25   26   27   28   29   30   31   32
## 0.00 0.00 0.00 0.01 0.01 0.00 0.00 0.00 0.01 0.00 0.00 0.00 0.00 0.00 0.01 0.00
##   33   34   35   36   37   38   39   40   41   42   43   44   45   46   47   48
## 0.01 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.01 0.00 0.00 0.00 0.00 0.00
##   49   50   51   52   53   54   55   56   57   58   59   60   61   62   63   64
## 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.01
##   65   66   67   68   69   70   71   72   73   74   75   76   77   78   79   80
## 0.00 0.00 0.00 0.00 0.00 0.00 0.03 0.00 0.01 0.00 0.00 0.00 0.00 0.01 0.00 0.00
##   81   82   83   84   85   86   87   88   89   90   91   92   93   94   95   96
## 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.01 0.00 0.00 0.01 0.00 0.04 0.00 0.00
##   97   98   99  100  101  102  103  104  105  106  107  108  109  110  111  112
## 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
##  113  114  115  116  117  118  119  120  121  122  123  124  125  126  127  128
## 0.00 0.01 0.01 0.00 0.05 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.01 0.00 0.01
##  129  130  131  132  133  134  135
## 0.00 0.01 0.00 0.00 0.00 0.00 0.00
```

```
hoslem.test(step.model$y, fitted(step.model), g=10)
```

```
##
##  Hosmer and Lemeshow goodness of fit (GOF) test
##
## data:  step.model$y, fitted(step.model)
## X-squared = 10.525, df = 8, p-value = 0.2301
```

```
resid.d<-residuals(step.model, type = "deviance")
resid.p<-residuals(step.model, type = "pearson")
std.res.d<-residuals(step.model, type = "deviance")/sqrt(1 - hatvalues(step.model)) # standardized devi
std.res.p <-residuals(step.model, type = "pearson")/sqrt(1 - hatvalues(step.model)) # standardized pear

dev.new(width = 1000, height = 1000, unit = "px")
par(mfrow=c(1,2))
plot(std.res.d[step.model$model$LiverPatient==0], col = "red",
     ylim = c(-3.5,3.5), ylab = "std. deviance residuals", xlab = "ID")
points(std.res.d[step.model$model$LiverPatient==1], col = "blue")

plot(std.res.p[step.model$model$LiverPatient==0], col = "red",
     ylim = c(-3.5,3.5), ylab = "std. Pearson residuals", xlab = "ID")
points(std.res.p[step.model$model$LiverPatient==1], col = "blue")
```

```
plot(cook.d,col="pink", pch=19, cex=1)
text(cooks.distance(step.model))
```

The residuals do not look very random, we can see that they tend to follow a pattern, we can also see that we have quiet a few points that are greater than our cooks-d cuttoff. We can also see from the hosmer test that our model is inadequate

(c)

Interpret relationships between predictors in the final model and the odds of an adult female being a liver patient. (based on estimated Odds Ratio).

```
inf.id = which(cooks.distance(step.model)>0.015)
final.model=glm(LiverPatient ~ Aspartate + TP, data=liver_female[-inf.id, ], family = "binomial")
summary(final.model)
```

```
##
## Call:
## glm(formula = LiverPatient ~ Aspartate + TP, family = "binomial",
##     data = liver_female[-inf.id, ])
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.86801    1.34930  -1.384  0.16623
## Aspartate    0.03060    0.01091   2.804  0.00505 **
## TP           0.21575    0.18563   1.162  0.24514
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 165.03  on 129  degrees of freedom
## Residual deviance: 144.87  on 127  degrees of freedom
## AIC: 150.87
##
## Number of Fisher Scoring iterations: 7
```

```
OR=exp(final.model$coefficients)
round(OR, 3)
```

```
## (Intercept)   Aspartate          TP
##       0.154       1.031       1.241
```

We can see from our odd ratios, for every one unit increase in Aspartate has a 3.1% increase chance of being liver patient. While for every one unit increase in TP there is a 24.1% increase chance of being a liver patient.

Exercise 2

(a)

For only males in the data set, find and specify the best set of predictors via stepwise selection with AIC criteria for a logistic regression model predicting whether a female is a liver patient.

```
liver_male <- liver %>%
  filter(Gender == 'Male')

full.model <- glm(LiverPatient ~ Age + TB + Alkphos + Alamine + Aspartate + TP + ALB, data = liver_male
```

```r
step.model <- step(full.model, direction = "both",
                      trace = FALSE)
summary(step.model)
```

```
##
## Call:
## glm(formula = LiverPatient ~ Age + TB + Alamine + TP + ALB, family = "binomial",
##     data = liver_male)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.480288   0.957804  -1.546 0.122225
## Age          0.019422   0.008366   2.321 0.020261 *
## TB           0.216593   0.088517   2.447 0.014409 *
## Alamine      0.018996   0.005259   3.612 0.000304 ***
## TP           0.437547   0.203497   2.150 0.031544 *
## ALB         -0.772168   0.289323  -2.669 0.007610 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 476.28  on 422  degrees of freedom
## Residual deviance: 393.61  on 417  degrees of freedom
## AIC: 405.61
##
## Number of Fisher Scoring iterations: 7
```

We can see that every predictor but aspartate is significant in the model.

(b)

Comment on the significance of parameter estimates under significance level alpha=0.1, what Hosmer-Lemeshow's test tells us about goodness of fit and point out any issues with diagnostics by checking residual plots and cook's distance plot (with cut-off 0.25).

```r
cook.d = cooks.distance(step.model)
round(cook.d, 2)
```

```
##    1    2    3    4    5    6    7    8    9   10   11   12   13   14   15   16
## 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.01 0.00 0.02 0.00 0.00 0.00 0.00 0.00
##   17   18   19   20   21   22   23   24   25   26   27   28   29   30   31   32
## 0.00 0.01 0.00 0.00 0.00 0.01 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.01 0.04 0.00
##   33   34   35   36   37   38   39   40   41   42   43   44   45   46   47   48
## 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
##   49   50   51   52   53   54   55   56   57   58   59   60   61   62   63   64
## 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.01 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
##   65   66   67   68   69   70   71   72   73   74   75   76   77   78   79   80
## 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.01 0.00 0.00 0.02 0.02 0.01 0.01
##   81   82   83   84   85   86   87   88   89   90   91   92   93   94   95   96
## 0.00 0.00 0.00 0.00 0.11 0.05 0.00 0.00 0.00 0.00 0.00 0.00 0.01 0.00 0.01 0.00
##   97   98   99  100  101  102  103  104  105  106  107  108  109  110  111  112
## 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.01 0.00 0.00
##  113  114  115  116  117  118  119  120  121  122  123  124  125  126  127  128
## 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
```

```
##  129  130  131  132  133  134  135  136  137  138  139  140  141  142  143  144
## 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.01 0.00 0.00 0.00
##  145  146  147  148  149  150  151  152  153  154  155  156  157  158  159  160
## 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
##  161  162  163  164  165  166  167  168  169  170  171  172  173  174  175  176
## 0.00 0.00 0.01 0.00 0.01 0.00 0.01 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
##  177  178  179  180  181  182  183  184  185  186  187  188  189  190  191  192
## 0.00 0.00 0.01 0.00 0.00 0.00 0.00 0.01 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
##  193  194  195  196  197  198  199  200  201  202  203  204  205  206  207  208
## 0.00 0.00 0.00 0.01 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.02
##  209  210  211  212  213  214  215  216  217  218  219  220  221  222  223  224
## 0.00 0.00 0.01 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.01 0.00 0.00 0.00 0.00
##  225  226  227  228  229  230  231  232  233  234  235  236  237  238  239  240
## 0.00 0.00 0.00 0.00 0.01 0.01 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
##  241  242  243  244  245  246  247  248  249  250  251  252  253  254  255  256
## 0.00 0.00 0.01 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.01
##  257  258  259  260  261  262  263  264  265  266  267  268  269  270  271  272
## 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
##  273  274  275  276  277  278  279  280  281  282  283  284  285  286  287  288
## 0.01 0.00 0.00 0.01 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
##  289  290  291  292  293  294  295  296  297  298  299  300  301  302  303  304
## 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.01 0.00 0.00 0.00 0.00 0.00 0.00 0.01 0.00
##  305  306  307  308  309  310  311  312  313  314  315  316  317  318  319  320
## 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
##  321  322  323  324  325  326  327  328  329  330  331  332  333  334  335  336
## 0.00 0.00 0.00 0.00 0.01 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.01 0.00 0.00
##  337  338  339  340  341  342  343  344  345  346  347  348  349  350  351  352
## 0.03 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.01 0.00 0.00
##  353  354  355  356  357  358  359  360  361  362  363  364  365  366  367  368
## 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
##  369  370  371  372  373  374  375  376  377  378  379  380  381  382  383  384
## 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
##  385  386  387  388  389  390  391  392  393  394  395  396  397  398  399  400
## 0.02 0.01 0.00 0.00 0.00 0.00 0.00 0.01 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
##  401  402  403  404  405  406  407  408  409  410  411  412  413  414  415  416
## 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
##  417  418  419  420  421  422  423
## 0.00 0.00 0.02 0.00 0.00 0.00 0.00
```

```
hoslem.test(step.model$y, fitted(step.model), g=10)
```

```
##
##  Hosmer and Lemeshow goodness of fit (GOF) test
##
## data:  step.model$y, fitted(step.model)
## X-squared = 3.4433, df = 8, p-value = 0.9035
```

```
resid.d<-residuals(step.model, type = "deviance")
resid.p<-residuals(step.model, type = "pearson")
std.res.d<-residuals(step.model, type = "deviance")/sqrt(1 - hatvalues(step.model)) # standardized devi
std.res.p <-residuals(step.model, type = "pearson")/sqrt(1 - hatvalues(step.model)) # standardized pear

dev.new(width = 1000, height = 1000, unit = "px")
par(mfrow=c(1,2))
plot(std.res.d[step.model$model$LiverPatient==0], col = "red",
```

```
     ylim = c(-3.5,3.5), ylab = "std. deviance residuals", xlab = "ID")
points(std.res.d[step.model$model$LiverPatient==1], col = "blue")

plot(std.res.p[step.model$model$LiverPatient==0], col = "red",
     ylim = c(-3.5,3.5), ylab = "std. Pearson residuals", xlab = "ID")
points(std.res.p[step.model$model$LiverPatient==1], col = "blue")

plot(cook.d,col="pink", pch=19, cex=1)
text(cooks.distance(step.model))
```

We can notice that there is less of a pattern in the residuals for this model, seems a bit more random than the previous models residuals. We can also see that there are 5 points that are greater than the cook's d cut off. From the hosmer test, we get a p-value greater than 0.5 meaning our model is inadequate

(c)

Interpret relationships between predictors in the final model and the odds of an adult female being a liver patient. (based on estimated Odds Ratio).

```
inf.id = which(cooks.distance(step.model)>0.015)
final.model=glm(LiverPatient ~ Age + TB + Alamine + TP + ALB, data=liver_male[-inf.id, ], family = "bin
summary(final.model)
```

```
##
## Call:
## glm(formula = LiverPatient ~ Age + TB + Alamine + TP + ALB, family = "binomial",
##     data = liver_male[-inf.id, ])
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -2.383056   1.108221  -2.150 0.031528 *
## Age          0.021151   0.009076   2.331 0.019777 *
## TB           0.540119   0.192376   2.808 0.004991 **
## Alamine      0.027752   0.007206   3.851 0.000117 ***
## TP           0.609537   0.235602   2.587 0.009677 **
## ALB         -1.040910   0.329889  -3.155 0.001603 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 450.24  on 412  degrees of freedom
## Residual deviance: 340.98  on 407  degrees of freedom
## AIC: 352.98
##
## Number of Fisher Scoring iterations: 8
```

```
OR=exp(final.model$coefficients)
round(OR, 3)
```

```
## (Intercept)         Age          TB     Alamine          TP         ALB
##       0.092       1.021       1.716       1.028       1.840       0.353
```

We can see that for males: - Every one unit increase in Age is associated with a 2.1% increase chance of being a liver patient - Every one unit increase in TB is associated with a 71.6% increase chance of being a liver patient - Every one unit increase in Alamine is associated with a 2.8% increase chance of being a liver

patient - Every one unit increase in TP is associated with a 84% increase chance of being a liver patient - Every one unit increase in ALB is associated with a 35.3% decrease chance of being a liver patient

This is model is much different from the model made for females. We can see that almost every predictor, execpt for Aspartate, has an effect on males chances of being liver patient. We can also see that are AIC is much higher for the model made for just males, we can see a 200 unit increase in AIC.

Exercise 3

(a)

First find and specify the best set of predictors via stepwise selection with AIC criteria.

```
full.model <- glm(maxlife10 ~ bodyweight + brainweight + totalsleep + gestationtime + factor(predationi

null.model <- glm(maxlife10 ~ 1, data = sleep, family = "binomial")

step.model <- step(null.model, direction = "both", scope = list(upper=full.model),
                     trace = FALSE, alpha = 0.1)
summary(step.model)
```

```
##
## Call:
## glm(formula = maxlife10 ~ brainweight + totalsleep + factor(sleepexposureindex) +
##     factor(predationindex), family = "binomial", data = sleep)
##
## Coefficients:
##                               Estimate Std. Error z value Pr(>|z|)
## (Intercept)                 -6.602e+00  4.864e+00  -1.357   0.1747
## brainweight                  5.101e-02  5.084e-02   1.003   0.3157
## totalsleep                   4.230e-01  2.647e-01   1.598   0.1100
## factor(sleepexposureindex)2  4.998e+00  2.559e+00   1.953   0.0508 .
## factor(sleepexposureindex)3  3.636e+01  9.624e+03   0.004   0.9970
## factor(sleepexposureindex)4  3.370e+01  1.037e+04   0.003   0.9974
## factor(sleepexposureindex)5  7.341e+01  1.262e+04   0.006   0.9954
## factor(predationindex)2     -2.535e+00  1.960e+00  -1.293   0.1960
## factor(predationindex)3     -2.512e+01  1.253e+04  -0.002   0.9984
## factor(predationindex)4     -1.826e+01  6.795e+03  -0.003   0.9979
## factor(predationindex)5     -5.264e+01  1.143e+04  -0.005   0.9963
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 68.31  on 50  degrees of freedom
## Residual deviance: 15.88  on 40  degrees of freedom
## AIC: 37.88
##
## Number of Fisher Scoring iterations: 20
```

From our results we can see that only sleepexposureindex of 3 has an significant effect on maxlife10

(b)

What does Hosmer-Lemeshow's test tells us about goodness of fit? And point out any issues with diagnostics by checking residual plots and cook's distance plot. Do not remove influential points but just make comments on suspicious observations.

```
cook.d = cooks.distance(step.model)
round(cook.d, 2)
```

```
##    1    2    3    4    5    6    7    8    9   10   11   12   13   14   15   16
## 0.00 0.00 0.03 0.00 0.00 0.02 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.02 0.06 0.02
##   17   18   19   20   21   22   23   24   25   26   27   28   29   30   31   32
## 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.02 0.00 0.00 0.00 0.04 0.01
##   33   34   35   36   37   38   39   40   41   42   43   44   45   46   47   48
## 0.00 0.00 0.27 0.00 0.00 0.00 0.00 0.28 0.00 0.04 0.00 0.00 0.00 0.01 0.00 0.00
##   49   50   51
## 0.00 0.00 0.03
```

```
hoslem.test(step.model$y, fitted(step.model), g=10)
```

```
## Warning in hoslem.test(step.model$y, fitted(step.model), g = 10): The data did
## not allow for the requested number of bins.
```

```
##
##  Hosmer and Lemeshow goodness of fit (GOF) test
##
## data:  step.model$y, fitted(step.model)
## X-squared = 7.0397, df = 7, p-value = 0.4248
```

```
resid.d<-residuals(step.model, type = "deviance")
resid.p<-residuals(step.model, type = "pearson")
std.res.d<-residuals(step.model, type = "deviance")/sqrt(1 - hatvalues(step.model)) # standardized devi
std.res.p <-residuals(step.model, type = "pearson")/sqrt(1 - hatvalues(step.model)) # standardized pear

dev.new(width = 1000, height = 1000, unit = "px")
par(mfrow=c(1,2))
plot(std.res.d[step.model$model$maxlife10==0], col = "red",
     ylim = c(-3.5,3.5), ylab = "std. deviance residuals", xlab = "ID")
points(std.res.d[step.model$model$maxlife10==1], col = "blue")

plot(std.res.p[step.model$model$maxlife10==0], col = "red",
     ylim = c(-3.5,3.5), ylab = "std. Pearson residuals", xlab = "ID")
points(std.res.p[step.model$model$maxlife10==1], col = "blue")

plot(cook.d,col="pink", pch=19, cex=1)
text(cooks.distance(step.model))
```

We can see from our hosmer test, that the model is adequate as the p-value is less than 0.5. We can also see that we have a few points above the cut off in our cook's d plot. These points may be causing some issues, however we are told to not remove them. ### (c) Interpret what the model tells us about relationships between the predictors and the odds of a species' maximum lifespan being at least 10 years.

```
summary(step.model)
```

```
##
## Call:
## glm(formula = maxlife10 ~ brainweight + totalsleep + factor(sleepexposureindex) +
##     factor(predationindex), family = "binomial", data = sleep)
##
## Coefficients:
##                               Estimate Std. Error z value Pr(>|z|)
## (Intercept)                 -6.602e+00  4.864e+00  -1.357   0.1747
```

```
## brainweight                  5.101e-02  5.084e-02   1.003   0.3157
## totalsleep                   4.230e-01  2.647e-01   1.598   0.1100
## factor(sleepexposureindex)2  4.998e+00  2.559e+00   1.953   0.0508 .
## factor(sleepexposureindex)3  3.636e+01  9.624e+03   0.004   0.9970
## factor(sleepexposureindex)4  3.370e+01  1.037e+04   0.003   0.9974
## factor(sleepexposureindex)5  7.341e+01  1.262e+04   0.006   0.9954
## factor(predationindex)2     -2.535e+00  1.960e+00  -1.293   0.1960
## factor(predationindex)3     -2.512e+01  1.253e+04  -0.002   0.9984
## factor(predationindex)4     -1.826e+01  6.795e+03  -0.003   0.9979
## factor(predationindex)5     -5.264e+01  1.143e+04  -0.005   0.9963
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 68.31  on 50  degrees of freedom
## Residual deviance: 15.88  on 40  degrees of freedom
## AIC: 37.88
##
## Number of Fisher Scoring iterations: 20
```

```
round(exp(step.model$coefficients), 3)
```

```
##                  (Intercept)                  brainweight
##                1.000000e-03                1.052000e+00
##                   totalsleep factor(sleepexposureindex)2
##                1.527000e+00                1.480500e+02
## factor(sleepexposureindex)3 factor(sleepexposureindex)4
##                6.173141e+15                4.332708e+14
## factor(sleepexposureindex)5     factor(predationindex)2
##                7.603846e+31                7.900000e-02
##     factor(predationindex)3     factor(predationindex)4
##                0.000000e+00                0.000000e+00
##     factor(predationindex)5
##                0.000000e+00
```

We can see that for animals: - Every one unit increase in brainweight is associated with a 5.2% increase chance of max life greater than 10. - Every one unit increase in totalsleep is associated with a 52.7% increase chance of max life greater than 10.

I am not sure if I may have messed something up or done something incorrectly, but my odd ratios for sleepexposureindex and predationindex seem to be incorrect.

Exercise 4

(a)

First find and specify the best set of predictors via stepwise selection with AIC criteria.

```
full.model <- glm(maxlife10 ~ bodyweight + brainweight + totalsleep + gestationtime + predationindex + s

null.model <- glm(maxlife10 ~ 1, data = sleep, family = "binomial")

step.model <- step(null.model, direction = "both", scope = list(upper=full.model),
                   trace = FALSE, alpha = 0.1)
summary(step.model)
```

```
##
## Call:
## glm(formula = maxlife10 ~ brainweight + totalsleep + sleepexposureindex +
##     predationindex, family = "binomial", data = sleep)
##
## Coefficients:
##                    Estimate Std. Error z value Pr(>|z|)
## (Intercept)        -6.16387    3.59301  -1.716   0.0863 .
## brainweight         0.06018    0.03544   1.698   0.0895 .
## totalsleep          0.35985    0.20995   1.714   0.0865 .
## sleepexposureindex  4.42111    1.97540   2.238   0.0252 *
## predationindex     -3.36917    1.51823  -2.219   0.0265 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 68.310  on 50  degrees of freedom
## Residual deviance: 19.212  on 46  degrees of freedom
## AIC: 29.212
##
## Number of Fisher Scoring iterations: 11
```

We can see that we get a similar model when the indexes are seen as continous instead of categorical. However this time, all predictors are significant.

(b)

What does Hosmer-Lemeshow's test tells us about goodness of fit? And point out any issues with diagnostics by checking residual plots and cook's distance plot. Do not remove influential points but just make comments on suspicious observations.

```
cook.d = cooks.distance(step.model)
round(cook.d, 2)
```

```
##    1    2    3    4    5    6    7    8    9   10   11   12   13   14   15   16
## 0.00 0.00 0.01 0.00 0.00 0.01 0.00 0.00 0.00 0.39 0.00 0.00 0.01 0.03 0.06 0.05
##   17   18   19   20   21   22   23   24   25   26   27   28   29   30   31   32
## 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.01 0.00 0.00 0.00 0.03 0.01
##   33   34   35   36   37   38   39   40   41   42   43   44   45   46   47   48
## 0.00 0.00 0.43 0.00 0.01 0.00 0.00 0.63 0.00 0.04 0.00 0.00 0.00 0.02 0.00 0.00
##   49   50   51
## 0.01 0.33 0.02
```

```
hoslem.test(step.model$y, fitted(step.model), g=10)
```

```
##
##  Hosmer and Lemeshow goodness of fit (GOF) test
##
## data:  step.model$y, fitted(step.model)
## X-squared = 1.4406, df = 8, p-value = 0.9937
```

```
resid.d<-residuals(step.model, type = "deviance")
resid.p<-residuals(step.model, type = "pearson")
std.res.d<-residuals(step.model, type = "deviance")/sqrt(1 - hatvalues(step.model)) # standardized devi
std.res.p <-residuals(step.model, type = "pearson")/sqrt(1 - hatvalues(step.model)) # standardized pear
```

```
dev.new(width = 1000, height = 1000, unit = "px")
par(mfrow=c(1,2))
plot(std.res.d[step.model$model$maxlife10==0], col = "red",
     ylim = c(-3.5,3.5), ylab = "std. deviance residuals", xlab = "ID")
points(std.res.d[step.model$model$maxlife10==1], col = "blue")

plot(std.res.p[step.model$model$maxlife10==0], col = "red",
     ylim = c(-3.5,3.5), ylab = "std. Pearson residuals", xlab = "ID")
points(std.res.p[step.model$model$maxlife10==1], col = "blue")

plot(cook.d,col="pink", pch=19, cex=1)
text(cooks.distance(step.model))
```

We can see now that our model is inadequate based on the hosmer test as our p-value is greater than 0.5. We can also notice an increase in the number of points greater than the cut off in the cook's d plot.

(c)

Interpret what the model tells us about relationships between the predictors and the odds of a species' maximum lifespan being at least 10 years.

```
summary(step.model)
```

```
##
## Call:
## glm(formula = maxlife10 ~ brainweight + totalsleep + sleepexposureindex +
##     predationindex, family = "binomial", data = sleep)
##
## Coefficients:
##                    Estimate Std. Error z value Pr(>|z|)
## (Intercept)        -6.16387    3.59301  -1.716   0.0863 .
## brainweight         0.06018    0.03544   1.698   0.0895 .
## totalsleep          0.35985    0.20995   1.714   0.0865 .
## sleepexposureindex  4.42111    1.97540   2.238   0.0252 *
## predationindex     -3.36917    1.51823  -2.219   0.0265 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 68.310  on 50  degrees of freedom
## Residual deviance: 19.212  on 46  degrees of freedom
## AIC: 29.212
##
## Number of Fisher Scoring iterations: 11
```

```
round(exp(step.model$coefficients), 3)
```

```
##        (Intercept)        brainweight         totalsleep sleepexposureindex
##              0.002              1.062              1.433             83.188
##     predationindex
##              0.034
```

- Every one unit increase in brainweight is associated with a 6.2% increase chance of max life greater than 10.

- Every one unit increase in totalsleep is associated with a 43.3% increase chance of max life greater than 10.

again, I am unsure what I have done incorrectly as the sleepexporesureindex predictor is reporting a 8000% increase, which I do not think is correct, while predationindex is reporting a 97.6% decrease based on our odd ratios.