Course EPIB-621 - Data Analysis for the Health Sciences

Assignment 4 - Solutions

1. There is a data set (adapted from Hosmer and Lemeshow) on the course web site called drugfree.txt. The variables contained in this data set are described in the table below:

Description	Code	Variable Name
Age at Enrollment	Years	age
Beck Depression Score at Admission	0.000-54.000	beck
IV Drug Use History at Admission	1 = Never, 2 = Previous 3 = Recent	ivhx
Number of Prior Drug Treatments	0-40	ndrugtx
Subject's Race	0 = White 1 = Other	race
Treatment Randomization Assignment	0 = Short 1 = Long	treat
Remained Drug Free for 12 Months	1 = Remained Drug Free 0 = Otherwise	drug.free

(a) The main outcome is the drugfree variable. For all continuous variables (age, beck, ndrugtx), present descriptive statistics within the two subgrouops defined by drug.free = 1 versus drug.free=0.

```
> drugfree.dat <- read.table(file="g:\\assignments\\drugfree.txt", header=T)
> attach(drugfree.dat)
> summary(age[drug.free==0])
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   20.0    27.0    32.0    32.2    37.0    53.0
> summary(age[drug.free==1])
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
```

```
   22.0    27.0    33.0    32.9    36.0    56.0

#  No apparent effect of age

> summary(beck[drug.free==0])
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   0.00   10.75   17.00   17.54   23.00   43.00
> summary(beck[drug.free==1])
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   0.00    9.00   16.00   16.83   24.00   54.00

#  No apparent effect of beck

> summary(ndrugtx[drug.free==0])
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   0.00    1.00    3.00    4.96    6.00   40.00
> summary(ndrugtx[drug.free==1])
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  0.000   1.000   2.000   3.327   4.000  34.000

#  Looks like there may be an effect of previous drug treatments
```

(b) Similarly, for all dichotomous (race, treat) or trichotomous (ivhx) variables, present proportions within each category within the two subgroups defined by drug.free = 1 versus drug.free=0.

```
#  Convert ivhx to a factor variable as it is trichotomous

> ivhx.f <- as.factor(ivhx)

#  Create tables for categorical variables

> table(ivhx.f, drug.free)
      drug.free
ivhx.f   0   1
     1 148  75
     2  83  26
     3 197  46

#  Looks like a strong effect for ivhx.f
```

```
> table(race, drug.free)
    drug.free
race   0   1
   0 330 100
   1  98  47

#  Looks like a strong effect for race

> table(treat, drug.free)
     drug.free
treat   0   1
    0 227  62
    1 201  85

#  Effect here too, maybe not as strong as above two effects
```

(c) Run a univariate logistic regression for each of the six independent variables. In each analysis, report the odds ratio with confidence interval. Remember to declare your factor variables.

```
#  Factor variable already declared above

#  For brevity, just report OR's + CI's here:
============================
age

> output <- glm(drug.free ~ age, family=binomial)
> logistic.regression.or.ci(output)

$OR
     age
1.018338

$OR.ci
[1] 0.9881691 1.0494288

#  No strong effect, but overall inconclusive, as cannot rule out a 5%
#  effect per year.
============================
beck
```

```
> output <- glm(drug.free ~ beck, family=binomial)
> logistic.regression.or.ci(output)

$OR
     beck
0.9918396

$OR.ci
[1] 0.9719252 1.0121619

#  No effect

============================
ivhx.f

> output <- glm(drug.free ~ ivhx.f, family=binomial)
$OR
  ivhx.f2   ivhx.f3
0.6181526 0.4607783

$OR.ci
          [,1]      [,2]
[1,] 0.3672199 1.0405553
[2,] 0.3013990 0.7044371

#  Strong effect from third category, inconclusive (but highly
#  suggestive) from second category, each in comparison to first
#  category
============================
ndrugtx

> output <- glm(drug.free ~ ndrugtx, family=binomial)

$OR
  ndrugtx
0.9277822

$OR.ci
[1] 0.8839724 0.9737633

#  Strong effect here, recall that this is per past treatment
```

```
============================
race

> output <- glm(drug.free ~ race, family=binomial)

$OR
    race
1.582653

$OR.ci
[1] 1.046652 2.393145

#  Evidence for an effect (but it may well just be weak,
#  or maybe strong, CI is wide)

============================
treat

> output <- glm(drug.free ~ treat, family=binomial)

$OR
   treat
1.548307

$OR.ci
[1] 1.060522 2.260447

#  Evidence for an effect (but it may well just be weak,
#  or maybe strong, CI is wide)

============================
```

2. Continuing the same example as above, run a multivariate logistic regression including all six variables.

```
> output <- glm(drug.free ~ age + ivhx.f + race + treat + beck +
           ndrugtx, family = binomial)

$OR
      age   ivhx.f2   ivhx.f3      race     treat      beck   ndrugtx
1.0522532 0.5528548 0.4679227 1.2314547 1.5511261 0.9998355 0.9387690
```

```
$OR.ci
          [,1]      [,2]
[1,] 1.0169791 1.0887508
[2,] 0.3151432 0.9698717
[3,] 0.2855551 0.7667581
[4,] 0.7977219 1.9010143
[5,] 1.0498638 2.2917183
[6,] 0.9789697 1.0211460
[7,] 0.8927132 0.9872010
```

3. Using your results from the first two questions, Create a table comparing all odds ratios and their confidence intervals between the univariate and multivariate models. Do you see any evidence for confounding? For each set of possibly confounded variables, check the correlation (if two continuous variables are involved) and create a table (if two categorical or a categorical and a continuous variable are involved). In this way, check that the preconditions for confounding are present (i.e., confounded variables are related to each other, and both are related to the outcome of interest).

The table is below:

Variable	*Multivariate OR + CI*	*Univariate OR + CI*
age	*1.05 (1.02, 1.09)*	*1.02 (0.99, 1.05)*
ivhx.f2	*0.55 (0.32, 0.97)*	*0.62 (0.37, 1.04)*
ivhx.f3	*0.47 (0.29, 0.77)*	*0.46 (0.30, 0.70)*
race	*1.23 (0.80, 1.90)*	*1.58 (1.05, 2.39)*
treat	*1.55 (1.05, 2.29)*	*1.55 (1.06, 2.26)*
beck	*1.00 (0.98, 1.02)*	*0.99 (0.97, 1.01)*
ndrugtx	*0.94 (0.89, 0.99)*	*0.93 (0.88, 0.97)*

Note that coefficients for treatment, ndrugtx, and beck were extremely stable from univariate to multivariate analyses, while race, and possibly ivhx.f and age changed a bit, and so were possibly confounded. Let's check how these variables correlate with each other.

```
> summary(age[race==0])
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  20.00   27.00   32.00   32.33   37.00   53.00

> summary(age[race==1])
```

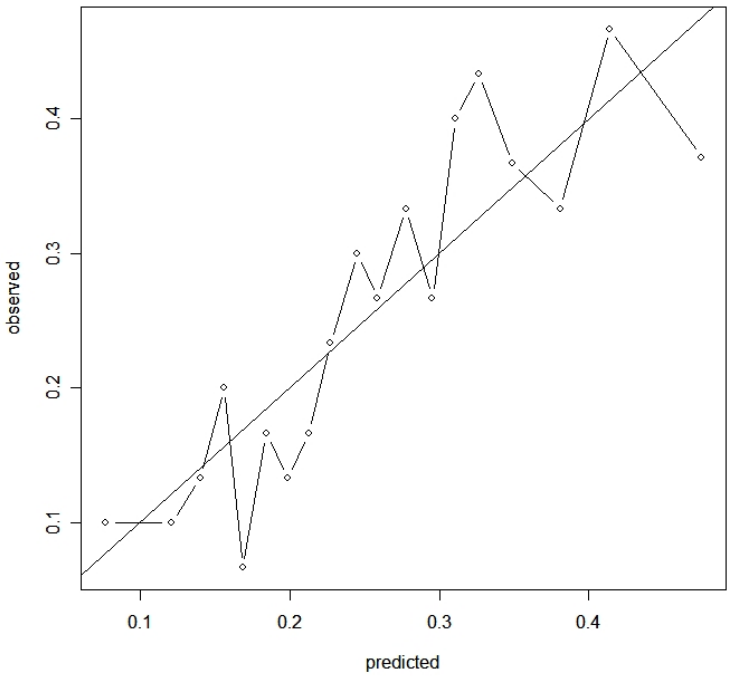
```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  22.00   28.00   32.00   32.53   37.00   56.00

#  Not much correlation here.

> table(race, ivhx.f)
    ivhx.f
race   1   2   3
   0 143  83 204
   1  80  26  39

#  Get percentages of ivhx.f in each race category

> c(143, 83, 204)/sum(c(143, 83, 204))
[1] 0.3325581 0.1930233 0.4744186

> c(80,26,39)/sum(c(80,26, 39))
[1] 0.5517241 0.1793103 0.2689655

#  Whites were twice as likely to have recent IV drug use, which
#  probably accounts for at least some of the confounding here.

#  We can check this by running a model with race and ivhx.f alone:

> output <- glm(drug.free ~ race + ivhx.f, family=binomial)
> logistic.regression.or.ci(output)

$OR
     race   ivhx.f2   ivhx.f3
1.3701673 0.6413400 0.4901924

$OR.ci
          [,1]     [,2]
[1,] 0.8952476 2.097027
[2,] 0.3797474 1.083133
[3,] 0.3178530 0.755974
```

4. Following the example in the class notes concerning creating a (Hosmer-Lemeshow) graph of predicted versus observed results, divide your fitted values from the full multivariate model into about 20 categories, from lowest predicted probabilities to highest. Create a plot (see plot in class notes for example) of average predicted probabilities versus observed event rates within these 20 subgroups of subjects.

Comment on how well the model fits overall, based on this graph.

```
#  Run model and create a summary of the fitted values

> output <- glm(drug.free ~ age + ivhx.f+ race + treat + beck + ndrugtx, family=binomial
> summary(output$fitted)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
0.01897 0.17360 0.24610 0.25570 0.32450 0.57380

#  For convenience, save fitted values to a vector called fit

#  Index these values from smallest to greatest

> index <- sort.list(fit)

#  Create a matrix of this index and the outcome, drug.free

> hosmer <- matrix(c(drug.free[index], fit[index]), byrow=F, ncol=2)

#  Let's take groups of 30 each, as 19*30 = 570.  We will have left over
#  which we will move into the last category.  So, we will have 18 categories of
#  30 subjects, and one last category with 35 subjects.

#  Create a blank vector to store results

>  observed <- rep(NA, 19)

#  Fill in first 18 entries of the observed vector

>  for (i in 1:18) {observed[i] <- sum(hosmer[(30*(i-1)+1):(30*i),1])/30}

#  Now fill in last entry of observed

> observed[19] <- sum(hosmer[(18*30+1):(18*30+35),1])/35

#  Look at results

> observed
 [1] 0.10000000 0.10000000 0.13333333 0.20000000 0.06666667 0.16666667
     0.13333333 0.16666667 0.23333333
[10] 0.30000000 0.26666667 0.33333333 0.26666667 0.40000000 0.43333333
     0.36666667 0.33333333 0.46666667
```

```
[19] 0.37142857

# Do same for predicted rates

# Create a blank vector to store results

> predicted <- rep(NA, 19)

> for (i in 1:18) {predicted[i] <- sum(hosmer[(30*(i-1)+1):(30*i),2])/30}

#  Now fill in last entry of predicted

> predicted[19] <- sum(hosmer[(18*30+1):(18*30+35),2])/35

#  Look at results

> predicted
 [1] 0.07686795 0.12094999 0.14024729 0.15658428 0.16880979 0.18429774
     0.19909437 0.21283650 0.22725254
[10] 0.24537235 0.25886692 0.27792091 0.29544333 0.31080266 0.32670269
     0.34878431 0.38062735 0.41399894
[19] 0.47532007

#  And create the plot

> plot(predicted, observed, type="b")
> abline(a=0, b=1)

#  Graph is far from perfect, but main trends are there.
```



5. In this question we will repeat number 2, but from a Bayesian viewpoint using WinBUGS. Run a multivariate logistic regression of the full model (i.e., using all six independent variables). In your WinBUGS program, add lines that calculate all odds ratios, and create a probability prediction for the outcome (drugfree) for someone who is aged 30, has a beck depression score of 10, has a recent history of IV drug use, is white, no prior drug treatment, and had a short treatment scheme.

Note: To create the WinBUGS data set, you need to add [ ]'s after each variable name. You also need to split the trichotomous variable ivhx into two separate dummy variables. This is done for you in the data set called drugfree.bugs.txt. The 1's have all been recoded as 0's, while ivhx2 has all 2's coded as 1, and inhx3 has all 3's coded as 1's.

```
#  WinBUGS program and results are below

model
 {
    for (i in 1:575)
 {                          #   Logistic model

  logit(p[i]) <-  alpha + b.age*age[i] + b.race*race[i] + b.beck*beck[i]
```

```
                + b.ivhx2 *ivhx2[i]  + b.ivhx3 *ivhx3[i]
                + b.ndrugtx * ndrugtx[i] + b.treat*treat[i]
                          #  Likelihood function for each data point
    drug.free[i]   ~ dbern(p[i])
    }
    alpha     ~ dnorm(0.0,1.0E-2)   #  Prior for intercept
    b.age     ~ dnorm(0.0,1.0E-2)   #  Priors for slopes
    b.race     ~ dnorm(0.0,1.0E-2)
    b.ivhx2     ~ dnorm(0.0,1.0E-2)
    b.ivhx3     ~ dnorm(0.0,1.0E-2)
    b.beck     ~ dnorm(0.0,1.0E-2)
    b.ndrugtx    ~ dnorm(0.0,1.0E-2)
    b.treat     ~ dnorm(0.0,1.0E-2)

#  Now to calculate the odds ratios

or.age     <-     exp(b.age)
or.race    <-     exp(b.race)
or.ivhx2   <-     exp(b.ivhx2)
or..ivhx3  <-     exp(b.ivhx3)
or.beck    <-     exp(b.beck)
or.ndrugtx <-     exp(b.ndrugtx)
or.treat   <-     exp(b.treat)

#   Predict for:   age =30, beck  = 10, ivhx.2 = 0, ivhx3 = 1,
#  race = 0, ndrugtx = 0, treat = 0

pred <- exp(alpha + b.age*30 + b.beck*10 + b.ivhx3 )/(1+ exp(alpha +
                    b.age*30 + b.beck*10 + b.ivhx3))
}

#  Inits

list(alpha=0, b.age =0, b.race =0, b.ivhx2 =0, b.ivhx3 =0, b.beck =0, b.ndrugtx =0,
                       b.treat =0)

#  Data

age[]    beck[]     ivhx2[] ivhx3[]      ndrugtx[]   race[]     treat[]     drug.free[]
39          9          0       1             1          0          1            0
33         34          1       0             8          0          1            0
33         10          0       1             3          0          1            0
```

```
32          20          0       1             1          0          0             0
24           5          0       0             5          1          1             1
30          32          0       1             1          0          1             0
.....................................etc.............................................
28          10          1       0             3          0          1             0
35          17          0       1             2          0          0             1
46        31.5          0       1            15          1          1             1
END

#  Results

node         mean      sd       MC error    2.5%      median     97.5%    start    sample
alpha       -2.427     0.5508   0.03539    -3.564    -2.422     -1.37     1001     10000
b.age        0.0533    0.01625  0.001054    0.0225    0.0531     0.08742  1001     10000
b.beck      -3.64E-4   0.01074  2.82E-4    -0.02171  -2.453E-4  0.02046  1001     10000
b.ivhx2     -0.6089    0.2913   0.006027   -1.199    -0.6067    -0.05303  1001     10000
b.ivhx3     -0.7721    0.2558   0.006286   -1.273    -0.7723    -0.2741   1001     10000
b.ndrugtx   -0.06637   0.02588  4.535E-4   -0.1204   -0.06494   -0.01838  1001     10000
b.race       0.2055    0.2223   0.003186   -0.2436    0.2086     0.6378   1001     10000
b.treat      0.4423    0.2005   0.004015    0.04342   0.4444     0.8274   1001     10000
or.ivhx3     0.4774    0.1237   0.002976    0.2799    0.4619     0.7602   1001     10000
or.age       1.055     0.01716  0.001114    1.023     1.055      1.091    1001     10000
or.beck      0.9997    0.01073  2.818E-4    0.9785    0.9998     1.021    1001     10000
or.ivhx2     0.5673    0.1669   0.00337     0.3014    0.5451     0.9484   1001     10000
or.ndrugtx   0.9361    0.02416  4.237E-4    0.8865    0.9371     0.9818   1001     10000
or.race      1.259     0.283    0.00404     0.7838    1.232      1.892    1001     10000
or.treat     1.588     0.3208   0.006424    1.044     1.56       2.287    1001     10000
pred         0.1707    0.03736  0.001122    0.1053    0.1681     0.2525   1001     10000

#  Very similar to frequentist results of previous questions.
```