Applied Logistic Regression – Assignment 3

Juho Ruohonen

April 23, 2017

Assignment 3

1. Computing Missing Statisics

```
(e1<-matrix(ncol=4,
  c(2.5634,0.7579,0.5033,NA,0.2071,0.3137,NA,0.2873,-12.380,
    NA,1.800,1.179,2*pnorm(-12.380),0.0157,0.0719,NA),
  dimnames=list(c("(Intercept)","C","D","G"),c("Estimate","Std.Err","z-value","Pr(>|z|)"))))
```

```
##             Estimate Std.Err z-value     Pr(>|z|)
## (Intercept)   2.5634  0.2071 -12.380 3.353428e-35
## C             0.7579  0.3137      NA 1.570000e-02
## D             0.5033      NA   1.800 7.190000e-02
## G                 NA  0.2873   1.179           NA
```

Our first unknown to solve is the z-score of predictor C. The z-score is a standardized measure of the "extremeness" of a given value, computed by using the standard error/deviation as the unit of measurement. Therefore, the z-score of C is obtained by dividing the estimated coefficient by the standard error:

```
(e1["C","z-value"] <- e1["C","Estimate"] / e1["C","Std.Err"])
```

```
## [1] 2.416003
```

```
e1
```

```
##             Estimate Std.Err    z-value     Pr(>|z|)
## (Intercept)   2.5634  0.2071 -12.380000 3.353428e-35
## C             0.7579  0.3137   2.416003 1.570000e-02
## D             0.5033      NA   1.800000 7.190000e-02
## G                 NA  0.2873   1.179000           NA
```

Our next unknown is the standard error of predictor D. The z-score tells us that the estimated coefficient 0.5033 is 1.8 times the standard error, so the SE is the estimated coefficient divided by z:

```
(e1["D","Std.Err"] <- e1["D","Estimate"] / e1["D","z-value"])
```

```
## [1] 0.2796111
```

```
e1
```

```
##             Estimate   Std.Err    z-value     Pr(>|z|)
## (Intercept)   2.5634 0.2071000 -12.380000 3.353428e-35
## C             0.7579 0.3137000   2.416003 1.570000e-02
## D             0.5033 0.2796111   1.800000 7.190000e-02
## G                 NA 0.2873000   1.179000           NA
```

Now we must solve the estimated coefficient of predictor G. We know that its magnitude is exactly 1.179 standard errors, thus:

```
(e1["G","Estimate"] <- e1["G","z-value"] * e1["G","Std.Err"])
```

```
## [1] 0.3387267
```

```
e1
```

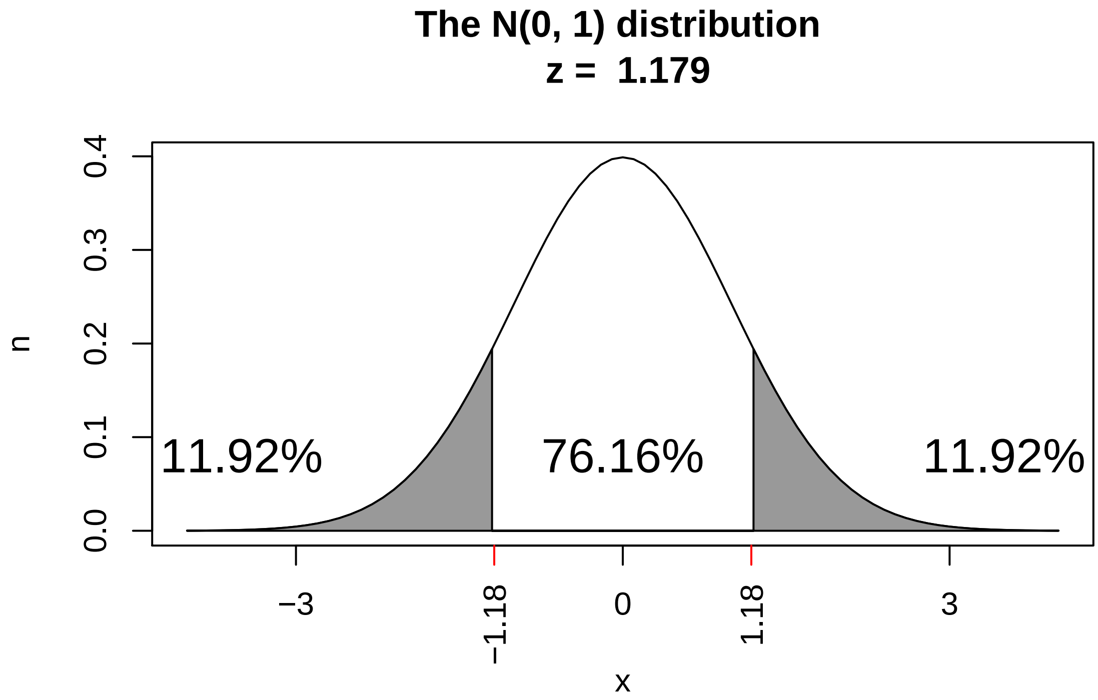
```
##               Estimate   Std.Err    z-value     Pr(>|z|)
## (Intercept) 2.5634000 0.2071000 -12.380000 3.353428e-35
## C           0.7579000 0.3137000   2.416003 1.570000e-02
## D           0.5033000 0.2796111   1.800000 7.190000e-02
## G           0.3387267 0.2873000   1.179000           NA
```

Lastly, we must calculate a p-value for the effect of predictor G. If we were paying attention during the introductory stats course, we'll remember that the p-value is usually just 2x the cumulative tail probability of a given quantile. In the case at hand, that quantile is the z-value, so p can be derived directly from z. We just have to remember to take both tails into account:

```
(e1["G","Pr(>|z|)"] <- 2*(1-pnorm(abs(e1["G","z-value"]))))
```

```
## [1] 0.2383982
```

```
zplot(e1["G","z-value"])
```



2a: Evans County Heart Study – High Catecholamine as the Sole Predictor

Now we'll analyze potential predictors of coronary heart disease. We'll start with just one predictor, namely, high catecholamine levels.

```
addmargins(mtable<-matrix(ncol = 2,c(27,44,95,443),dimnames =
  list(c("c1","c0"),c("ill","healthy"))))
```

```
##     ill healthy Sum
## c1   27      95 122
## c0   44     443 487
## Sum  71     538 609
```

```
CHD <- c(rep(1,times=71),rep(0,times=538))
cat.high <- c(rep(1,times=27),rep(0,times=44),rep(1,times=95),rep(0,times=443))
catecholamine<-glm(CHD ~ cat.high, family = binomial(link="logit"))
summary(catecholamine)
```

```
##
## Call:
## glm(formula = CHD ~ cat.high, family = binomial(link = "logit"))
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.7073  -0.4352  -0.4352  -0.4352   2.1928
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -2.3094     0.1581 -14.610  < 2e-16 ***
## cat.high      1.0513     0.2693   3.903 9.49e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 438.56  on 608  degrees of freedom
## Residual deviance: 424.43  on 607  degrees of freedom
## AIC: 428.43
##
## Number of Fisher Scoring iterations: 5
```

This output suggest that high levels of catecholamine are a very statistically significant predictor of CHD. Though the p-value is what tends to be reported more often, I personally find the z-value more informative – unlike the p-value, the z-score gives us both the direction and the statistical significance of the effect. Its sign indicates whether the effect is positive or negative, while the absolute value reports the magnitude, with 1.96 being equivalent to $p = .05$.

However, it is always risky to jump to conclusions based on models that have only one independent variable. Such a model takes no account of potential confounding variables. In fact, I can't imagine a real-world situation where I'd want to do a logistic regression with just one predictor – if there's only predictor, then I'd be inclined to use Pearson's chi-squared or Fisher's exact test instead. They are simpler to use and interpret and they measure the same thing with much less of a hassle. As far as I can see, the very purpose of regression analysis (logistic or otherwise) is to analyze the individual effects of predictors in multivariate models where all the other variables have been adjusted for.

2b: Evans County Heart Study – a 3-Predictor Model

We'll now add 2 more dichotomous predictors – a=1 indicates age 55 or higher, while e=1 indicates an abnormal result in an ECG scan. We'll first display the partial tables.

```
(a0e0<-matrix(ncol = 2, c(1,17,7,257),
  dimnames=list(c("c1","c0"),a0e0=c("ill","healthy"))))
```

```
##     a0e0
##      ill healthy
##   c1   1       7
##   c0  17     257
```

```
(a0e1<-matrix(ncol = 2, c(3,7,14,52),
  dimnames=list(c("c1","c0"),a0e1=c("ill","healthy"))))
```

```
##     a0e1
##      ill healthy
##   c1   3      14
##   c0   7      52
```

```
(a1e0<-matrix(ncol = 2, c(9,15,30,107),
  dimnames=list(c("c1","c0"),a1e0=c("ill","healthy"))))
```

```
##     a1e0
##      ill healthy
##   c1   9      30
##   c0  15     107
```

```
(a1e1<-matrix(ncol = 2, c(14,5,44,27),
  dimnames=list(c("c1","c0"),a1e1=c("ill","healthy"))))
```

```
##     a1e1
##      ill healthy
##   c1  14      44
##   c0   5      27
```

Then we'll bind this data together into a data frame that **glm()** can work with. In the data frame format, each individual is represented by one row. Converting contingency tables into this raw format is the opposite procedure of what I usually do, so I don't know an elegant way to accomplish it. **Glm()** doesn't seem to understand arrays either (like the mantel-haenzel test does), so the following is the only way I can think of to make the data usable.

```
type1<-data.frame(ill=1,high.cat=1,over54=0,ECG.bad=0) #a0e0c1 -- ill
type2<-data.frame(ill=rep(1,times=17),high.cat=0,over54=0,ECG.bad=0) #a0e0c1 -- ill
type3<-data.frame(ill=rep(0,times=7),high.cat=1,over54=0,ECG.bad=0) #a0e0c0 -- healthy
type4<-data.frame(ill=rep(0,times=257),high.cat=0,over54=0,ECG.bad=0) #a0e0c1 -- healthy
type5<-data.frame(ill=rep(1,times=3),high.cat=1,over54=0,ECG.bad=1) #a0e1c1 -- ill
type6<-data.frame(ill=rep(1,times=7),high.cat=0,over54=0,ECG.bad=1) #a0e1c0 -- ill
type7<-data.frame(ill=rep(0,times=14),high.cat=1,over54=0,ECG.bad=1) #a0e1c1 -- healthy
type8<-data.frame(ill=rep(0,times=52),high.cat=0,over54=0,ECG.bad=1) #a1e1c0 -- healthy
type9<-data.frame(ill=rep(1,times=9),high.cat=1,over54=1,ECG.bad=0) #a1e0c1 -- ill
type10<-data.frame(ill=rep(1,times=15),high.cat=0,over54=1,ECG.bad=0) #a1e0c0 -- ill
type11<-data.frame(ill=rep(0,times=30),high.cat=1,over54=1,ECG.bad=0) #a1e0c1 -- healthy
type12<-data.frame(ill=rep(0,times=107),high.cat=0,over54=1,ECG.bad=0) #a1e0c1 -- healthy
type13<-data.frame(ill=rep(1,times=14),high.cat=1,over54=1,ECG.bad=1) #a1e1c1 -- ill
type14<-data.frame(ill=rep(1,times=5),high.cat=0,over54=1,ECG.bad=1) #a1e1c0 -- ill
type15<-data.frame(ill=rep(0,times=44),high.cat=1,over54=1,ECG.bad=1) #a1e1c1 -- healthy
type16<-data.frame(ill=rep(0,times=27),high.cat=0,over54=1,ECG.bad=1) #a1e1c0 -- healthy
EvansData<-rbind(type1,type2,type3,type4,type5,type6,type7,type8,
  type9,type10,type11,type12,type13,type14,type15,type16) #Concatenate
EvansData[sample(nrow(EvansData),size=10),] #View 10 randomly sampled rows
```

```
##     ill high.cat over54 ECG.bad
## 549   0        1      1       1
## 83    0        0      0       0
## 500   0        0      1       0
## 479   0        0      1       0
## 242   0        0      0       0
## 227   0        0      0       0
## 258   0        0      0       0
## 232   0        0      0       0
## 560   0        1      1       1
## 437   0        0      1       0
```

Now we can finally use **glm()**:

```
EvansModel<-glm(ill ~ high.cat+over54+ECG.bad, data = EvansData, family=binomial(link = "logit"))
summary(EvansModel)
```

```
##
## Call:
## glm(formula = ill ~ high.cat + over54 + ECG.bad, family = binomial(link = "logit"),
##     data = EvansData)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.7860  -0.5037  -0.3756  -0.3756   2.3181
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -2.6163     0.2123 -12.322   <2e-16 ***
## high.cat      0.6223     0.3194   1.948   0.0514 .
## over54        0.6157     0.2838   2.169   0.0301 *
## ECG.bad       0.3620     0.2904   1.247   0.2126
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 438.56  on 608  degrees of freedom
## Residual deviance: 418.18  on 605  degrees of freedom
## AIC: 426.18
##
## Number of Fisher Scoring iterations: 5
```

The output of the 3-predictor model shows that the significance of high catecholamine levels was being greatly exaggerated by the confounding effect of age earlier. Its p-value, which was below .001 in the single-variable model, now rises above the .05 threshold, in principle rendering the predictor non-significant. Much like our first assignment (the one dealing with the Cochran-Maentel-Haenzel test), this is another pedagogical example of confounding. In the three-predictor model, age is the most significant risk factor of CHD. High cathecolamine levels seem associated too, despite the slightly-too-large p-value, while abnormal ECG results fall clearly short of the limit.

3: Evans County Heart Study – Adding an Interaction Term

Now we'll add one more predictor – a variable representing the interaction of age over 54 and abnormal ECG resuilts:

```
Evans.4p <- glm(formula = ill ~ high.cat + over54 + ECG.bad + over54*ECG.bad,
                data=EvansData, family=binomial(link="logit"))
summary(Evans.4p)
```

```
##
## Call:
## glm(formula = ill ~ high.cat + over54 + ECG.bad + over54 * ECG.bad,
##     family = binomial(link = "logit"), data = EvansData)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.7526  -0.5218  -0.3590  -0.3590   2.3554
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)     -2.7096     0.2442 -11.095   <2e-16 ***
## high.cat         0.6437     0.3166   2.033   0.0420 *
## over54           0.7842     0.3419   2.294   0.0218 *
## ECG.bad          0.6509     0.4275   1.523   0.1278
## over54:ECG.bad  -0.4859     0.5430  -0.895   0.3708
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 438.56  on 608  degrees of freedom
## Residual deviance: 417.39  on 604  degrees of freedom
## AIC: 427.39
##
## Number of Fisher Scoring iterations: 5
```

The interaction of high age and abnormal ECGs turns out to be the first and only disfavoring effect in the model. For this reason, the ill effects of the three other variables become more pronounced. In particular, high catecholamine levels become a statistically significant risk factor again.

Why is the combination of abnormal ECG results and age over 54 associated with a lesser risk of CHD? Here's a hypothesis: abnormal ECG results are relatively common and "typical" in old people, whereas in younger people they are not to be expected, and are more often a warning sign of heart disease. As an analogy, balding is normal and "to be expected" in older males, while in male children it is invariably an indicator that something is wrong.

4: Birth Control Use among Fijian Women

The next dataset is about use vs. non-use of birth control among married Fizian women of fertile age. 1,607 women were interviewed for the study. The independent variables are:

1. Whether she wants more children (dichotomous)
2. Education level (dichotomous: low or high)
3. Age (ordinal with 4 categories)

The youngest age group is reported as <25 years, which is unnecessarily vague on the part of the authors. IMHO they should have reported the lower bound. Googling suggests that the minimum marriageable age for women is 16 in Fiji. Should we rename this category for additional clarity? My suggestion is no: though we know the legal limit, we don't know that the actual age range of the youngest group in the study actually runs from 16 to 25. So we'll leave the naming as it is.

First we must wrangle the data into analyzable form. I still don't know a way to do this without either A) writing lots of code, or B) doing a lot of manual editing in Excel/Calc and then importing the result to R. I'll use the former strategy again, to make everything transparent:

```
subsample1<-data.frame(Age=rep("<25",53),Education="Low",WantsMore="yes",bc.use=0)
subsample2<-data.frame(Age=rep("<25",6),Education="Low",WantsMore="yes",bc.use=1)
subsample3<-data.frame(Age=rep("<25",10),Education="Low",WantsMore="no",bc.use=0)
subsample4<-data.frame(Age=rep("<25",4),Education="Low",WantsMore="no",bc.use=1)
subsample5<-data.frame(Age=rep("<25",212),Education="High",WantsMore="yes",bc.use=0)
subsample6<-data.frame(Age=rep("<25",52),Education="High",WantsMore="yes",bc.use=1)
subsample7<-data.frame(Age=rep("<25",50),Education="High",WantsMore="no",bc.use=0)
subsample8<-data.frame(Age=rep("<25",10),Education="High",WantsMore="no",bc.use=1)
subsample9<-data.frame(Age=rep("25-29",60),Education="Low",WantsMore="yes",bc.use=0)
subsample10<-data.frame(Age=rep("25-29",14),Education="Low",WantsMore="yes",bc.use=1)
subsample11<-data.frame(Age=rep("25-29",19),Education="Low",WantsMore="no",bc.use=0)
subsample12<-data.frame(Age=rep("25-29",10),Education="Low",WantsMore="no",bc.use=1)
subsample13<-data.frame(Age=rep("25-29",155),Education="High",WantsMore="yes",bc.use=0)
subsample14<-data.frame(Age=rep("25-29",54),Education="High",WantsMore="yes",bc.use=1)
subsample15<-data.frame(Age=rep("25-29",65),Education="High",WantsMore="no",bc.use=0)
subsample16<-data.frame(Age=rep("25-29",27),Education="High",WantsMore="no",bc.use=1)
subsample17<-data.frame(Age=rep("30-39",112),Education="Low",WantsMore="yes",bc.use=0)
subsample18<-data.frame(Age=rep("30-39",33),Education="Low",WantsMore="yes",bc.use=1)
subsample19<-data.frame(Age=rep("30-39",77),Education="Low",WantsMore="no",bc.use=0)
subsample20<-data.frame(Age=rep("30-39",80),Education="Low",WantsMore="no",bc.use=1)
subsample21<-data.frame(Age=rep("30-39",118),Education="High",WantsMore="yes",bc.use=0)
subsample22<-data.frame(Age=rep("30-39",46),Education="High",WantsMore="yes",bc.use=1)
subsample23<-data.frame(Age=rep("30-39",68),Education="High",WantsMore="no",bc.use=0)
subsample24<-data.frame(Age=rep("30-39",78),Education="High",WantsMore="no",bc.use=1)
subsample25<-data.frame(Age=rep("40-49",35),Education="Low",WantsMore="yes",bc.use=0)
subsample26<-data.frame(Age=rep("40-49",6),Education="Low",WantsMore="yes",bc.use=1)
subsample27<-data.frame(Age=rep("40-49",46),Education="Low",WantsMore="no",bc.use=0)
subsample28<-data.frame(Age=rep("40-49",48),Education="Low",WantsMore="no",bc.use=1)
subsample29<-data.frame(Age=rep("40-49",8),Education="High",WantsMore="yes",bc.use=0)
subsample30<-data.frame(Age=rep("40-49",8),Education="High",WantsMore="yes",bc.use=1)
subsample31<-data.frame(Age=rep("40-49",12),Education="High",WantsMore="no",bc.use=0)
subsample32<-data.frame(Age=rep("40-49",31),Education="High",WantsMore="no",bc.use=1)
bc<-rbind(subsample1,subsample2,subsample3,subsample4,subsample5,subsample6,
  subsample7,subsample8,subsample9,subsample10,subsample11,subsample12,subsample13,
  subsample14,subsample15,subsample16,subsample17,subsample18,subsample19,subsample20,
  subsample21,subsample22,subsample23,subsample24,subsample25,subsample26,subsample27,
  subsample28,subsample29,subsample30,subsample31,subsample32)
#Make WantsMore a binary variable where 0=no and 1=yes:
bc$WantsMore<-as.numeric(bc$WantsMore); bc$WantsMore[bc$WantsMore==2]<-0
```

Whew, that was a lot of work! Now we can finally begin the analysis.

```
model1<-glm(bc.use ~ Age+Education+WantsMore, family=binomial(link="logit"), data=bc)
summary(model1) #Model summary
```

```
##
## Call:
## glm(formula = bc.use ~ Age + Education + WantsMore, family = binomial(link = "logit"),
##     data = bc)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.3429  -0.8819  -0.6129   1.1351   2.0480
##
```

```
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)    
## (Intercept)     -1.1332     0.1871  -6.057 1.39e-09 ***
## Age25-29         0.3894     0.1759   2.214  0.02681 *  
## Age30-39         0.9086     0.1646   5.519 3.40e-08 ***
## Age40-49         1.1892     0.2144   5.546 2.92e-08 ***
## EducationHigh    0.3250     0.1240   2.620  0.00879 ** 
## WantsMore       -0.8330     0.1175  -7.091 1.33e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2003.7  on 1606  degrees of freedom
## Residual deviance: 1867.8  on 1601  degrees of freedom
## AIC: 1879.8
##
## Number of Fisher Scoring iterations: 4
```

```
data.frame(exp(coef(model1))) #Odds ratios
```

```
##                exp.coef.model1..
## (Intercept)            0.3219965
## Age25-29               1.4760678
## Age30-39               2.4808804
## Age40-49               3.2845805
## EducationHigh          1.3840232
## WantsMore              0.4347628
```

There is nothing unexpected in these results. For obvious reasons, the desire to have more children is the most statistically significant predictor, with the odds ratio of birth control use .43 between those who want more children and those who do not. Age is the second-most significant predictor, with older age groups consistently more likely to use birth control. The oldest group's odds of using birth control are on average about 3.3 times those of the youngest. This is to be expected. The young have low impulse control and high libido compared to the old, both of which are factors that favor reckless or otherwise short-sighted sexual behavior. Older women also have more reason to avoid accidental pregnancies, since the health hazards involved in pregnancy are much greater at 40 than at 20.

The binary distinction between low and high level of education is the third-most important factor. Those with a high education have about 38% higher odds of using birth control than their less-educated counterparts. This might also be tied to impulse control. Educational success requires discipline and planning, at least one of which is also necessary component of successful birth control.

This is not good enough, however. We saw in the previous example how the interpretation changed after adding an interaction term. Moreover, with only 3 independent variables involved, there is really no excuse for not testing for every possible two-way interaction – the number of variables will still remain manageable, and valuable new information might be obtained. In comparing two models, it can be useful to pay attention to the Residual Deviance and AIC statistics, both of which are statistics where a lower value indicates a better fit. In the simple 3-variable model, these statistics are 1867.8 and 1879.8, respectively. Now we'll fit a model with interaction terms:

```
model2<-glm(bc.use ~ Age*Education+Age*WantsMore+Education*WantsMore,
  family=binomial(link="logit"), data=bc)
summary(model2) #Model summary
```

```
##
## Call:
```

```
## glm(formula = bc.use ~ Age * Education + Age * WantsMore + Education *
##     WantsMore, family = binomial(link = "logit"), data = bc)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6306  -0.8372  -0.6525   1.1346   2.0387
##
## Coefficients:
##                         Estimate Std. Error z value Pr(>|z|)
## (Intercept)             -1.47100    0.45337  -3.245 0.001176 **
## Age25-29                 0.75577    0.50936   1.484 0.137868
## Age30-39                 1.57249    0.45913   3.425 0.000615 ***
## Age40-49                 1.48716    0.48904   3.041 0.002358 **
## EducationHigh            0.01935    0.42172   0.046 0.963400
## WantsMore               -0.47338    0.39674  -1.193 0.232802
## Age25-29:EducationHigh  -0.15790    0.45838  -0.344 0.730495
## Age30-39:EducationHigh  -0.05178    0.41840  -0.124 0.901500
## Age40-49:EducationHigh   0.98645    0.52114   1.893 0.058378 .
## Age25-29:WantsMore      -0.22536    0.41024  -0.549 0.582772
## Age30-39:WantsMore      -0.95014    0.38199  -2.487 0.012870 *
## Age40-49:WantsMore      -1.19012    0.51278  -2.321 0.020291 *
## EducationHigh:WantsMore  0.48617    0.26522   1.833 0.066785 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2003.7  on 1606  degrees of freedom
## Residual deviance: 1840.4  on 1594  degrees of freedom
## AIC: 1866.4
##
## Number of Fisher Scoring iterations: 4
```

```
data.frame(exp(coef(model2))) #Odds ratios
```

```
##                         exp.coef.model2..
## (Intercept)                     0.2296967
## Age25-29                        2.1292508
## Age30-39                        4.8186302
## Age40-49                        4.4244969
## EducationHigh                   1.0195400
## WantsMore                       0.6228945
## Age25-29:EducationHigh          0.8539370
## Age30-39:EducationHigh          0.9495344
## Age40-49:EducationHigh          2.6816880
## Age25-29:WantsMore              0.7982269
## Age30-39:WantsMore              0.3866855
## Age40-49:WantsMore              0.3041840
## EducationHigh:WantsMore         1.6260829
```

Both Residual Deviance and AIC went down. In other words, the more complex model fits better even after a penalty is imposed for its increased complexity. This suggests that it's the better model. There are many interesting observations that can be made on this output. To mention just one, advanced age is still a significantly predictor of birth control control use UNLESS the woman wants more children, in which case it's a significant predictor of non-use.

However, we pay a price for this increased specificity of information. It is now harder to determine the significance of individual factors because their respective effects have been fractionated across several items. For instance, WantsMore, which is was the most potent predictor in the simpler model, now loses its independent statistical significance (although it appears quite significant when coinciding with the older age groups). Something similar happens to Education, which loses all statistical significance as an independent factor. It is now harder to provide concise answers to questions about these variables' respective effects. This is doubtless an acceptable price to pay if the model predicts outcomes significantly better than a simpler alternative. But does it? In order to assess whether this increased complexity is truly worth the trouble, we will next compare this model to the simpler one using leave-one-out cross-validation.

Cross-validation is a technique where a dataset is divided into complementary training and testing subsets. The model coefficients are calculated by fitting the model on the training data, and then the model's predictive performance is measured on the testing data. This provides an estimate of how useful a model is with unseen data, thus helping detect overfitting. Overfitting is a phenomenon where a model performs well on the dataset that it was trained on but is of limited use predicting outcomes in new data. An overfitted model relies excessively on the idiosyncrasies of the training dataset, whereby it fits that particular dataset well but is of limited use with unseen data.

Leave-one-out cross-validation performs the training-testing cycle n times – the model is trained on a sub-dataaset consisting of all but one of the observations, then the outcome of that sole remaining observation is predicted. This is repeated until every single unit of observation has once been the one that was left out and predicted on, and the mean error rate across all cycles is calculated. We will now perform this procedure on both models and compare their respective prediction error rates on unseen data. Note that this entails performing two cycles of 1,607 logistic regressions and predictions, which is not a simple task even for modern computers.

```r
library(boot)
ErrorRate<-function(Y,prob){mean(abs(Y-prob)>0.5)} #Define Error Rate
data.frame(Model1.TestingError = cv.glm(data=bc, glmfit=model1, cost=ErrorRate)$delta[1],
  Model2.TestingError = cv.glm(data=bc, glmfit=model2, cost=ErrorRate)$delta[1])
```

```
##   Model1.TestingError Model2.TestingError
## 1           0.2962041           0.3242066
```

It appears that even though the more complex model seems good on paper, with favorable AIC and residual deviance ratings, its real-world value in predicting outcomes is less than that of the simpler model. This suggests that the model with the interactions was indeed overfitted, placing undue importance on idiosyncratic characteristics of the dataset that have little predictive power outside that particular data. We shall therefore declare the simple model containing only Age, WantsMore and Education the winner. Increased age and high education correlate with increased use of birth control, while the desire to have more children correlates with it negatively.

5. Attitudes to Racial Integration in Public Housing

This dataset investigates the effects of interracial contact, proximity to public housing projects, and overall attitudes to racial integration on white people's sentiments towards racial integration in public housing. As usual, we must first wrangle the data into analyzable form. For the sake of conciseness and clarity, I'm renaming the dependent variable PositiveSentiment, whose presence = 1 and absence = 0. I'm also converting Proximity from a character factor into a binary variable for which 1 = proximity and 0 = no proximity.

```r
subset1<-data.frame(Proximity=rep(1,times=77),Contact=1,Norms="Favorable",PositiveSentiment=1)
subset2<-data.frame(Proximity=rep(1,times=32),Contact=1,Norms="Favorable",PositiveSentiment=0)
subset3<-data.frame(Proximity=rep(1,times=30),Contact=1,Norms="Unfavorable",PositiveSentiment=1)
subset4<-data.frame(Proximity=rep(1,times=36),Contact=1,Norms="Unfavorable",PositiveSentiment=0)
subset5<-data.frame(Proximity=rep(1,times=14),Contact=0,Norms="Favorable",PositiveSentiment=1)
subset6<-data.frame(Proximity=rep(1,times=19),Contact=0,Norms="Favorable",PositiveSentiment=0)
```

```r
subset7<-data.frame(Proximity=rep(1,times=15),Contact=0,Norms="Unfavorable",PositiveSentiment=1)
subset8<-data.frame(Proximity=rep(1,times=27),Contact=0,Norms="Unfavorable",PositiveSentiment=0)
subset9<-data.frame(Proximity=rep(0,times=43),Contact=1,Norms="Favorable",PositiveSentiment=1)
subset10<-data.frame(Proximity=rep(0,times=20),Contact=1,Norms="Favorable",PositiveSentiment=0)
subset11<-data.frame(Proximity=rep(0,times=36),Contact=1,Norms="Unfavorable",PositiveSentiment=1)
subset12<-data.frame(Proximity=rep(0,times=37),Contact=1,Norms="Unfavorable",PositiveSentiment=0)
subset13<-data.frame(Proximity=rep(0,times=27),Contact=0,Norms="Favorable",PositiveSentiment=1)
subset14<-data.frame(Proximity=rep(0,times=36),Contact=0,Norms="Favorable",PositiveSentiment=0)
subset15<-data.frame(Proximity=rep(0,times=41),Contact=0,Norms="Unfavorable",PositiveSentiment=1)
subset16<-data.frame(Proximity=rep(0,times=118),Contact=0,Norms="Unfavorable",PositiveSentiment=0)
contact<-rbind(subset1,subset2,subset3,subset4,subset5,subset6,subset7,subset8,subset9,subset10,
  subset11,subset12,subset13,subset14,subset15,subset16)
```

A priori, I don't feel confident enough to make predictions on the effect of proximity and contact. In some cases, both factors probably reinforce the existing attitudes towards racial integration, while in others it probably dispels those attitudes, whatever they are. Favorable Norms should obviously correlate with Positive Sentiment and vice versa. Since only three independent variables were recorded, we'll start with the obvious approach of including all of them in the model:

```r
con.model1<-glm(PositiveSentiment ~ Proximity+Contact+Norms,
  family=binomial(link = "logit"), data=contact)
summary(con.model1)
```

```
##
## Call:
## glm(formula = PositiveSentiment ~ Proximity + Contact + Norms,
##     family = binomial(link = "logit"), data = contact)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.5356  -1.0804  -0.7819   1.1709   1.6335
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)      -0.23258    0.17876  -1.301    0.193
## Proximity         0.09816    0.18441   0.532    0.595
## Contact           0.94585    0.18113   5.222 1.77e-07 ***
## NormsUnfavorable -0.79599    0.17665  -4.506 6.61e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 839.96  on 607  degrees of freedom
## Residual deviance: 772.37  on 604  degrees of freedom
## AIC: 780.37
##
## Number of Fisher Scoring iterations: 4
```

Residual Deviance is 772.37; AIC is 780.37. Here are the confidence intervals of the odds ratios:

```r
exp(confint(con.model1))
```

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

```
##                      2.5 %    97.5 %
## (Intercept)      0.5572623 1.1242437
## Proximity        0.7669561 1.5812799
## Contact          1.8085643 3.6809564
## NormsUnfavorable 0.3186408 0.6371853
```

This analysis lends strong support to the hypothesis that contact with racial minorities decreases prejudice towards them. It also confirms the assumption that general norms regarding racial integration very strongly influence attitudes towards mixed-race public housing. On the other hand, mere geographic proximity, independently of contact, does not appear to have a significant effect. This suggests that our model could be made more parsimonious by dropping this variable altogether. Before we can do so, however, it is a good idea to check whether it interacts meaningfully with the other two independent variables – in fact, let's again check all possible two-factor interactions, since we have so few variables that it's easy to do:

```r
con.model2<-glm(PositiveSentiment ~ Proximity*Contact+Proximity*Norms+Contact*Norms,
  family=binomial(link="logit"),data=contact)
summary(con.model2)
```

```
##
## Call:
## glm(formula = PositiveSentiment ~ Proximity * Contact + Proximity *
##     Norms + Contact * Norms, family = binomial(link = "logit"),
##     data = contact)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -1.560  -1.023  -0.787   1.220   1.627
##
## Coefficients:
##                            Estimate Std. Error z value Pr(>|z|)
## (Intercept)                -0.37551    0.24100  -1.558 0.119205
## Proximity                   0.23536    0.35371   0.665 0.505795
## Contact                     1.24144    0.32583   3.810 0.000139 ***
## NormsUnfavorable           -0.63784    0.28446  -2.242 0.024940 *
## Proximity:Contact          -0.28205    0.37376  -0.755 0.450469
## Proximity:NormsUnfavorable  0.04757    0.37010   0.129 0.897733
## Contact:NormsUnfavorable   -0.32932    0.36867  -0.893 0.371716
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 839.96  on 607  degrees of freedom
## Residual deviance: 771.15  on 601  degrees of freedom
## AIC: 785.15
##
## Number of Fisher Scoring iterations: 4
```

There are no meaningful interactions, since each interaction term has a high p-value. The negative effect of Unfavorable Norms is strongly moderated by Contact, though it stays negative. As for Proximity, it looks safe to drop it from the model. We will now run a Likelihood Ratio Test to analyze whether dropping Proximity from the model significantly worsens the fit:

```r
con.model3<-glm(PositiveSentiment ~ Contact+Norms, family=binomial(link="logit"),data=contact)
anova(con.model1,con.model3,test = "LRT")
```

```
## Analysis of Deviance Table
```

```
##
## Model 1: PositiveSentiment ~ Proximity + Contact + Norms
## Model 2: PositiveSentiment ~ Contact + Norms
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1       604     772.37
## 2       605     772.66 -1 -0.28257    0.595
```

The Residual Deviance (unexplained variance) barely changes after the removal of Proximity, and the p-value associated with the change is nearly .6, so it appears safe to stick with the new, simpler model. Here are its summary and the confidence intervals of the odds ratios of the remaining predictors:

```
summary(con.model3)
```

```
##
## Call:
## glm(formula = PositiveSentiment ~ Contact + Norms, family = binomial(link = "logit"),
##     data = contact)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.5190  -1.0946  -0.7888   1.1925   1.6243
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)        -0.1980     0.1664  -1.189    0.234
## Contact             0.9726     0.1742   5.583 2.37e-08 ***
## NormsUnfavorable   -0.8102     0.1747  -4.638 3.52e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 839.96  on 607  degrees of freedom
## Residual deviance: 772.66  on 605  degrees of freedom
## AIC: 778.66
##
## Number of Fisher Scoring iterations: 4
```

```
exp(confint(con.model3))
```

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

```
##                      2.5 %    97.5 %
## (Intercept)      0.5910296 1.1360853
## Contact          1.8827283 3.7289352
## NormsUnfavorable 0.3153705 0.6257843
```

Similarly to the birth control exercise, we'll again test our conclusions through leave-one-out crossvalidation:

```
data.frame(ConModel1.TestingError = cv.glm(data=contact, glmfit=con.model1, cost=ErrorRate)$delta[1],
  ConModel2.TestingError = cv.glm(data=contact, glmfit=con.model2, cost=ErrorRate)$delta[1],
  ConModel3.TestingError = cv.glm(data=contact, glmfit=con.model3, cost=ErrorRate)$delta[1])
```

```
##   ConModel1.TestingError ConModel2.TestingError ConModel3.TestingError
## 1              0.4128289              0.3536184              0.3536184
```

The procedure confirms that any extra variables in the model besides Contact and Norms yield no benefit whatsoever. The simplest model with just Contact and Norms is the clear winner.

This data appears to robustly support both the contact hypothesis and the fairly self-evident assumption that the overall norms of the host society influence the individual's attitudes. The data dates back to 1955 – an era when racial discrimination was the norm rather than the anomaly. Policies of desegregation in housing had not yet come into effect, nor were they being actively promoted at the time. We might therefore assume that the effect seen in the data is one of culturally imposed, largely untested negative assumptions being challenged and moderated by first-hand contact with racial minorities. Mere proximity without contact, on the other hand, appears to have no effect to speak of.

Do the results mean that contact with racial minorities always has a positive effect on the attitudes towards those minorities? Or is it instead the case that culturally imposed attitudes to racial minorities tend to be stereotyped exaggerations which, whether positive or negative, are usually revised and neutralized as a result of first-hand contact? In other words, if the culturally imposed preconceptions about an ethnic minority are *positive* instead of negative, do they become more positive upon contact, or do they become less positive, i.e., are revised and neutralized following contact? Answering this question would require a dataset from a very different time and place.