Stat 5421 (Geyer) Spring 2016
Homework Assignment 4
Due Friday, April 29, 2016

Our problem 4.1. Agresti problem 8.28 except just analyze as a contingency table. Do not treat any variable as the "response." The data can be read into R as follows

```
> foo <-
+     read.table(url("http://www.stat.umn.edu/geyer/5421/data/exercise-8.28.txt"),
+         header = TRUE)
> sapply(foo, class)

satisfaction      contact    influence         type       counts
    "factor"     "factor"     "factor"     "factor"    "integer"
```

Our problem 4.2. Agresti problem 8.8 except rather than use the analysis in Agresti do your own analysis using the R function `polr` in the R package `MASS`. The data can be read into R as follows as follows

```
> foo <-
+     read.table(url("http://www.stat.umn.edu/geyer/5421/data/table-8.18.txt"),
+         header = TRUE)
> sapply(foo, class)

   status seat.belt  location    gender    counts
 "factor"  "factor"  "factor"  "factor" "integer"
```

Note that `status` is not an ordered factor (`read.table` doesn't automagically do that). You have to handle that yourself.

The thing I said in class about having to redefine the data so there is one row of the data for each individual turns out to be unnecessary. One of the examples on the help for the `polr` function shows how to use the `weights` argument to avoid that.

Find the best model you can. The `anova.polr` function produces P-values for tests of model comparison.

I'm not sure exactly what Agresti meant for (b). Just make a confidence interval for the regression coefficient for gender.

For part (c) interpret all of the regression coefficients. What does it mean that they have the signs they do (if anything). In interpreting an interaction term (like the location by seat belt interaction Agresti asks about), figure out what the size of the effects are for all four classes (for this interaction,

rural without seat belt, rural with seat belt, urban without seat belt, urban with seat belt).

It may help to read the details section of the on-line help for the `polr` function. It may also help to look at the probabilities produced by the `predict.polr` function. It is weird that this function has no on-line help but as arguments unlike `predict.glm`. Say

```
predict(out, type = "probs")
```

to get predicted class probabilities. The way I figured this out was to look at the source, which is found with

```
getAnywhere(predict.polr)
```

(shouldn't have to do that). Maybe Venables and Ripley expect you to look in their book.

Our problem 4.3. Agresti problem 4.13. The data can be read into R as follows

```
> foo <-
+     read.table(url("http://www.stat.umn.edu/geyer/5421/data/table-4.11.txt"),
+         header = TRUE)
> sapply(foo, class)

     made attempted
"integer" "integer"
```