36-350: Data Mining

Lab 11
Date: November 7, 2003 **Due: end of lab**

Interspersed throughout this lab are questions that you will have to answer at check-off.

1. Download the files for this lab from the course web page to the desktop:

   `http://www.stat.cmu.edu/~minka/courses/36-350/lab/`

2. Open a Word or Notepad document to record your work.

Start R

3. `Start -> All Programs -> Class software -> R 1.7.0`

4. Load the special functions for this lab:

   `File -> Source R code...`

   Browse to the desktop and pick `lab11.r` (it may have been renamed to `lab11.r.txt` when you downloaded it). Another window will immediately pop up for you to pick the `mining.zip` file you downloaded.

The dataset

5. The dataset (in `x`) is the result of crash tests on 274 cars, ranging over 1987–1992 model years. For each test, an instrumented dummy was seated in the car and the car was crashed into a barrier at 35mph.

Head	Head injury to the dummy
D.P	Dummy in the `Driver` or `Passenger` seat
Protection	Kind of protection:
	Driver and passenger airbags (`d&p airbags`)
	Driver-side airbag (`d airbag`)
	Motorized belts, Passive belts, Manual belts
Doors	Number of doors on the car (2 or 4)
Size	Car weight/size category:
	small, light, compact, midsize, heavy, SUV

   The data is contained in variable `x`. All of the variables except `Head` are categorical. `Head` has already been transformed with a logarithm to make its distribution symmetric.

   Historical context: Around this time period, automatic protection was required by law, but auto makers strongly preferred automatic belts over airbags, despite mounting evidence that automatic belts were ineffective and sometimes worse than manual belts. Congress took up the issue, and eventually ruled in 1991 that all new cars starting in 1998 must have driver and passenger airbags.

Linear model

6. Make a linear model to predict `Head` from all other variables. (Because `Head` is not the last column, you can't use `formula(x)` this time.) Notice how `lm` codes the categories.

7. Make a plot which shows the importance of `Protection`. *Which protections make a difference to head injury (i.e. their effect is nonzero)? Which protections are significantly different from each other?*

Interactions

8. Find one significant interaction between `D.P` and the other variables. Add it to the model.

9. Repeat adding interactions with `D.P` until the residual plot is flat. To verify that your interaction terms are useful, make a partial residual plot.

10. Now find and eliminate all interactions between `Doors` and the other variables, working one interaction at a time. Make a new partial residual plot.

11. Remove any irrelevant variables in the model. Make a `summary` of the final model for the homework.

12. You can now get checked off.

Plotting residuals To plot residuals or partial residuals using line charts:

```
predict.plot(fit)
predict.plot(fit,partial=T,las=3)
```

The option `las=3` rotates the axis labels.

Slice plots `interact.plot` (from lab 10) will make line charts when the predictors are categorical:

```
interact.plot(fit)
interact.plot(fit,ypred="D.P",las=3)
```

The second line makes only the plots with `D.P` as the slicing variable. Places where the curves fail to be flat are interactions.

Adding interaction terms An interaction term between categorical variables is simply an indicator for a combination of categories. For example, suppose the head injury is unusually high for two-door SUVs. The interaction term for this would be `Yes` when `Doors=2` and `Size=SUV`, and `No` otherwise. This term could be added to `x` as follows:

```
x[,"door2.suv"] = factor.logical((x[,"Doors"]=="2") & (x[,"Size"]=="SUV"))
```

Then re-fit the model including `door2.suv`. To later remove it from `x`:

```
x = not(x,"door2.suv")
```