36-315: Statistical Graphics and Visualization

Lab 15

Date: April 29, 2003 **Due: end of lab**

In this lab, you will use parallel-coordinate plots to find predictors relevant to your response, and use contour plots to rank them by strength.

In this lab, you are given the commands to type in, but with question marks (`??`) denoting things you must provide, based on the dataset you are using.

1. Download the files for this lab from the course web page.
2. Open a Word document to record your work (plots and commands).

Start R

3. `Start -> Programs -> Class software -> R 1.5.1`
4. Set the working directory to `My Documents`:

   `File -> Change dir...`

5. Load the special functions for this lab:

   ```
   source("lab15.r")
   ```

   The file `variables.r` contains a suggested set of variables and transformations, in a formula called `goodvars`. It has been changed a bit from the previous lab.

   ```
   source("variables.r")
   ```

Load your data

6. From the class web page, download the census data files for your state. Unzip them into `My Documents`. Then they can be loaded via

   ```
   a = read.csv("tracta.csv")
   b = read.csv("tractb.csv")
   frame = clean.census.data(a,b)
   ```

Parallel-coordinate plots

7. A parallel-coordinate plot can help find new variables relevant to the response. First you split the tracts by quantiles of the response (`cut.quantile`), and compute the median in each group (`prototypes`):

```
y = frame[,??]
f = cut.quantile(y,4)
x = model.frame(goodvars,frame)
x = na.omit(x)
vars = age.vars
xp = prototypes(x[,vars],f)
parallel.plot(xp,??)
```

8. Change `vars` above to `income.vars`, `race.vars`, and then `family.vars`, giving four parallel-coordinate plots. *Which variables follow the trend of the response?*

Interaction plots

9. Of all the variables which correlate with the response, you want to separate the true causes. This can be done by finding predictors that make others obsolete. Take a group of variables where you suspect one of them is a true cause and make an `interact.plot`. *Rank the variables by strength and make a causal graph, as shown in class.*

10. Show us your graphs.

parallel.plot If `xp` is a data frame with a small number of rows, a basic parallel-coordinate plot can be made via

```
parallel.plot(xp)
```

The axis labels can be rotated and made smaller via

```
parallel.plot(xp,las=2,cex.axis=0.8)
```

The colors can be made sequential via

```
parallel.plot(xp,color.pal=YlGnBu.colors,bg=grey(0.5))
```

The entire plot can be rotated via `horizontal=T`. The variable scales and ordering are automatically chosen by the linear profiles method. (For some other visualizations, try `star.plot(xp)` and `data.image(xp)`.)

interact.plot Instead of making contour plots one by one, the function `interact.plot` will make plots for all pairs of predictors in a formula. Since this can take awhile, we suggest testing with a subsampled frame first:

```
i = sample(1:nrow(frame),500)
x = model.frame(? ~ ? + ? + ...,frame)
interact.plot(x[i,],bg=grey(0.5),nlev=8,cex=0)
```

Then try without `i`.