36-315: Statistical Graphics and Visualization

Lab 4
Date: February 4, 2002 **Due: start of class February 11, 2002**

1 Introduction

In this lab, it is up to you to design scatterplots that describe the relationships between various Census variables. The lab will walk you through one example and then let you apply what you've learned.

This lab will use variables from both census files. Load `tracta.csv` and `tractb.csv` into separate variables, say `a` and `b`, then concatenate them horizontally via

```
frame <- cbind(a,b)
```

2 Example

In class it was shown that, in Kentucky at least, there is a relationship between population density and the percent of female residents. To see if this is true in your state, you can make a scatterplot:

```
i <- frame$TOTPOP>100 & frame$POPPSQMI>0
frame <- frame[i,]
x <- frame$PCTFEMAL
y <- frame$POPPSQMI
plot(x,y,xlab="PCTFEMAL",ylab="POPPSQMI")
```

2.1 Transform for symmetry

To make this easier to read, transform the population density variable to even its distribution:

```
y <- log(frame$POPPSQMI)
plot(x,y,xlab="PCTFEMAL",ylab="log(POPPSQMI)")
```

The logarithm function is a transformation in the power family, $\frac{x^p-1}{p}+1$, corresponding to $p=0$. You may want to try other powers to see if they improve matters.

2.2 Choose limits

Next is to choose the right limits. Don't insist on including zero in the plot—instead, you should zoom in on where the bulk of the data lies. This will reveal the shape of the relationship best. For example, you might limit `PCTFEMAL` to 40–60, by specifying an `xlim` vector:

```
plot(x,y,xlim=c(40,60),xlab="PCTFEMAL",ylab="log(POPPSQMI)")
```

Now the relationship should be prominently displayed. Some other stylistic choices, like the plotting symbol, tick marks, grid lines, margins, and such have already been made by S-PLUS, and usually they are good ones.

To suppress warnings about points being out of bounds, click the box for

```
Options -> Graph Options... -> Editable Graphics
```

Ironically, the checkbox called "Suppress warnings" doesn't seem to do anything.

2.3 Smoothing

You can add a smooth `lowess` curve to any scatterplot by typing

```
lines(lowess(x,y,f=1/2),col=2,lwd=3)
```

The `f` argument controls the percentage of data included in each local regression window, `col` sets the color of the curve, and `lwd` sets the line width. You should generally be skeptical of the default value of `f`, just like you should be skeptical of the default number of bins in a histogram. The rule is similar to histograms: gradually increase the `f` argument until it starts to show too much wiggle, then move back. (In this case, there are no error bars to help you.)

The resulting curve probably won't look right, because it is trying to predict `POPPSQMI` from `PCTFEMAL`, when what we want is `PCTFEMAL` as a function of `POPPSQMI`. A clue that lowess will fail is that the data is tall and skinny, rather than short and wide. To fix this, swap x and y and repeat the above commands.

Many trends that appear linear at first are actually curved. To make sure that you don't miss this, choose a reasonable aspect ratio for the figure. For S-PLUS 2000, turn on Editable Graphics. Click on a corner of the axis rectangle and six green squares will appear. Dragging one of these squares will resize the plot. You can also move the plot around on the page, represented by a white rectangle. The white rectangle will have a fixed aspect ratio, regardless of how you size the window. It represents a piece of paper, and shows how your graph will appear when you print it.

2.4 Transform for linearity

If the trend is a simple upward or downward curve, you can often straighten it out by transforming the variables appropriately. The reason for doing this is that people are better at judging vertical distances from a line than from a curve. Consequently, you get a better picture of how the data deviates from the trend, and can more easily spot outliers. In this case, the curve for `PCTFEMAL` versus `log(POPPSQMI)` should be nearly straight already, so no transformation is needed.

With bivariate data, there are two different kinds of outliers. A `univariate outlier` is one that could be spotted by looking at histograms of individual variables, i.e. it is an extreme point on one or both variables. A (strictly) `bivariate outlier` shows up only when you plot the two variables against each other. It lives outside the main distribution of data and is unusually far away from the trend curve, yet not extreme on any one variable. Often these are the most interesting kinds of outliers, since they 'buck the trend'.

> **Question:** Are there any strictly bivariate outliers on your final graph of `PCTFEMAL` vs. `log(POPPSQMI)`? Circle them.

3 Making your own plots

Now it's time to make your own plots. For each pair of variables below, make a scatterplot which best shows their relationship.

1. MEDHHINC vs. POPPSQMI
2. MEDYRBLT vs. MEDHHINC
3. PCTVACNT vs. MEDRENT
4. MEDHHINC vs. PCTELEM

It is not specified which variable should be `x` and which should be `y`—that is up to your judgement. Follow the four steps in the example: transform for symmetry, choose limits, smooth, and transform for linearity. Submit your code and your plots. Only one plot for each variable pair will be accepted. On each plot, identify points which deviate from the main trend (strictly bivariate outliers). If there are outliers outside the plotting limits, draw arrows to indicate their direction.