36-315: Statistical Graphics and Visualization

Lab 3
Date: January 28, 2002 **Due: start of class February 4, 2002**

---

1 Introduction

In this lab, you will get a chance to compare some of the graphical methods shown in class and see how they compare on your data. Specifically, you will make graphics that compare the values of multiple Census variables. As before, start by reading your state's `tracta.csv` file into S-PLUS and cleaning up the names.
The functions `cut.quantile` and `dotplot.superpose` that will be used in the lab are available at `www.stat.cmu.edu/~minka/courses/36-315/code/compare.s`. Download it into your work directory and read it via

```
source("compare.s")
```

(Ignore the warnings about functions being redefined.)
There are four questions with associated plots to turn in. All plots should be labeled correctly using `xlab` and `ylab`. The first few examples show you how to do this.

2 Comparing distributions

The variable `PCT18.24` gives the percentage of residents aged 18–24 with each census tract. If you make a histogram of this variable, you will probably find that it is skewed. What causes this? To investigate, you can look at how the distribution changes with respect to population density.
First extract the `PCT18.24` variable and remove undefined values:

```
x <- frame$PCT18.24
i <- !is.na(x)
x <- x[i]
```

Now extract the `POPPSQMI` variable of the same census tracts:

```
y <- frame$POPPSQMI[i]
```

You can abstract this variable into high/medium/low via the function `cut.quantile`. It uses the 33% and 66% quantiles as the breakpoints.

```
a <- cut.quantile(y)
```

The `x` values can now be split among the groups in `a` via `split(x,a)`.
For example, this command makes a set of strip plots:

```
stripplot(a~x,jitter=T,xlab="PCT18.24",ylab="POPPSQMI")
```

These give a bare-bones look at how the distribution changes.
To compare the distributions more easily, it helps to simplify the representation. A stack of histograms is one possibility, but these are difficult to read. In this situation the `boxplot` is ideal. You can make a simple boxplot via

```
boxplot(split(x,a))
```

Some more parameters will make it pretty:

```
boxplot(split(x,a),outline=F,outpch=1,ylab="PCT18.24",xlab="POPPSQMI")
```

Turn in both strip and box plots.

> **Question:** Describe how the distribution of `PCT18.24` changes as the population density changes. Why do you think this happens?

3 Comparing means

An even simpler comparison is to compare the means only. The group means can be calculated with

```
m <- tapply(x,a,mean)
```

A barplot represents the means using bars that extend from zero. This has the effect of making differences look small.

```
barplot(m,horiz=T,names=names(m))
```

A dotplot represents the means using dots, with limits that only span the range of the data. It emphasizes differences.

```
dotplot(m)
```

Turn in both plots.

> **Question:** Based on what you found in the previous section, is it misleading to compare only the group means?

4 Selecting rows from a data frame

Columns 52 to 64 give the number of residents in various age groups from 0 to 85+. You can select all of these variables out of the data frame via

```
x <- frame[52:64]
```

(With some versions of S-PLUS or R, it is columns 53 to 65 instead. Go figure.) Like last time, we want to remove rows that have missing values. Except now we have a frame with multiple columns, and we want to remove a row if *any* column value is missing. The command `sapply(x,is.na)` returns a matrix telling you which elements of the data frame are undefined and which are not. This matrix can be collapsed across columns via `apply`, like so:

```
i <- !apply(sapply(x,is.na),1,any)
```

The vector `i` now indicates which rows are fully defined. To keep only those rows, say

```
x <- x[i,]
```

The comma indicates that you are selecting rows instead of columns.

5 Graphing the full age distribution

The values in `x` are totals for each tract. To get percentages, sum across all tracts and divide by the total, as follows:

```
m <- sapply(x,sum)/sum(x)
```

To graph these percentages, you can use a pie chart:

```
piechart(m)
```

a bar plot:

```
barplot(m,horiz=T,names=names(m))
```

or a dot plot:

```
dotplot(m)
```

The graph has some unusually sharp steps, especially evident in the bar plot. Why is this? If you look at the age ranges, you will see that they are not equal. This causes some values to seem depressed or augmented compared to their neighbors. The right thing to do is plot density, which means dividing each variable by its age interval:

```
w <- c(5,5,4,4,7,10,10,10,5,5,10,10,10)
m <- m/w
```

Plots of the divided values look a lot more reasonable. Turn in these plots.

> **Question:** Point out two age ranges which are difficult to compare on the pie chart but easy to compare on the bar/dot plots.

6 Superposed dotplots

Now you can see how the full age distribution changes with population density. Redefine `y` to use the same rows as `x`, and re-cut it:

```
y <- frame$POPPSQMI[i]
a <- cut.quantile(y)
```

Define a function to compute percentages, like it was done above:

```
pct <- function(x) sapply(x,sum)/sum(x)
```

Then apply it to each group:

```
m <- sapply(split(x,a),pct)
```

This returns a matrix giving the percent in a certain age range for each population density group. For the plots to look right, the age ranges need to be normalized. Change the `pct` function to read:

```
pct <- function(x) sapply(x,sum)/sum(x)/w
```

Then recompute `m`.
To make age the column variable, you should transpose the matrix via the function called `t`:

```
m <- t(m)
```

If you run `barplot` on this matrix, you will get a stacked bar plot. Stacked bars represent values using non-aligned lengths, which are difficult to decode perceptually. Note that bars cannot just be superposed because they would obscure each other. However dot plots can be superposed, giving a complete and uncluttered picture:

```
dotplot.superpose(m)
```

Turn in this plot.

> **Question:** Using the superposed dotplot, describe how the age distribution changes from low density areas to high density areas. Speculate what might cause these changes.