Statistics 787 **Assignment 1** **Due: August 20**

1. Create an R function which draws dotcharts. You should try to make the function be as flexible as possible and should be able to handle the following:

   (a) multiple label strips to the left of the plot (like the graph area dotchart),

   (b) multiple range intervals (like the animal speeds plot),

   (c) multiple points with different plotting symbols for each row of the dotchart.

   (d) a key describing what each plotting symbol represents (when there are multiple columns to x).

   (e) The ability to independently specify an axis and axis label both above and below the plot (so that you could have a normal axis below and a log axis above, for example).

   You may find it useful to structure the arguments in the following way:

   `x`: a numeric matrix containing the values to be plotted. Each row of the matrix should give the coordinates for several points to be plotted on each row of the dotchart. The `pch` argument, described below, should be able to be used to specify a different plotting symbol for each column.

   `labels`: a character matrix with the same number of rows as `x`. Each column of `labels` should give the elements of a labelling panel to the left of the dotchart. The `adj` argument, described below, should give the justification for each column.

   `pch`: the plotting symbol to be used for each column. (There should be a reasonable set of defaults.)

   `adj`: numeric vector giving the string justification to be used in each labelling panel. (the default should be left-justification for all columns.)

   `group`: an optional grouping vector which enables the plot to be partitioned into groups (as with the graph areas plot).

   `...`: other arguments for specifying axes and the plotting symbol key.

Hints:

It is not possible to use `split` to partition a column into groups *but* it is possible to partition a data frame. This suggests that you do something like the following:

```
groups = split(data.frame(x = x, labels = labels),
               group)

##  organise the layout here

for(i in 1:length(groups)) {
    labels = groups[[i]]$labels
    x = groups[[i]]$x

    for(j in 1:ncol(labels))
        ## draw label panel j for group i

    ## draw the points

    ## draw axes and labels if it is the first or
    ## last group
 }
```

Axes are a bit tricky, but I would approach it in the following way. The arguments would be something like this:

```
at = NULL, labels = TRUE, tick = TRUE, smallticks = NULL,
at1 = at, labels1 = labels1, tick1 = tick,
   smallticks1 = smallticks,
at3 = at, labels3 = FALSE, tick3 = tick,
   smallticks3 = smallticks,
```

When it comes time to draw the bottom axes the calls would be:

```
axis(1, at = at1, labels = labels1, tick = tick1)
if (!is.null(smallticks1)
    axis(1, at = smallticks1, labels = FALSE,
         tcl = -0.25)
```

By default, this would produce the default style of axis. Alternative tick positions and labels could be produced by specifying either `at` and `labels` or `at1` and `labels1`. If `smallticks` or `smallticks1` was specified, smaller ticks with no labels would be drawn at these positions.

When it comes time to draw the top axes, the calls would be:

```
axis(1, at = at3, labels = labels3, tick = tick3)
if (!is.null(smallticks3)
    axis(1, at = smallticks3, labels = FALSE,
         tcl = -0.25)
```

This would produce an axis with tick positions and small tick positions determined by `at3` and `smallticks3` (defaulting to the same as the axes below the plot) but no labels. It is possible to specify arbitrary tick positions and labels using `at3` and `labels3`.

The size of margins above and below would be determined by whether or not `labels1` and `labels3` were non-null.

2. Produce a function which draws a "dotchart matrix" like that on page 151 of the pages from Cleveland's book. This can be either just as shown, or using Trellis style labelling strips, rather than a label above the plots.