Mixed models in R using the lme4 package
Part 2: Lattice graphics

Douglas Bates

University of Wisconsin - Madison
and R Development Core Team
<Douglas.Bates@R-project.org>

University of Lausanne
July 1, 2009

Outline

Presenting data

Scatter plots

Histograms and density plots

Box-and-whisker plots and dotplots

Outline

Presenting data

Scatter plots

Histograms and density plots

Box-and-whisker plots and dotplots

Outline

Presenting data

Scatter plots

Histograms and density plots

Box-and-whisker plots and dotplots

Outline

Presenting data

Scatter plots

Histograms and density plots

Box-and-whisker plots and dotplots

Outline

Presenting data

Scatter plots

Histograms and density plots

Box-and-whisker plots and dotplots

Exploring and presenting data

- When possible, use graphical presentations of data. Time spend creating informative graphical displays is well invested.
- Ron Snee, a friend who spent his career as a statistical consultant for DuPont, once said, *"Whenever I am writing a report, the most important conclusion I want to communicate is always presented as a graphic and shown early in the report. On the other hand, if there is a conclusion I feel obligated to include but would prefer people not notice, I include it as a table."*
- One of the strengths of R is its graphics capabilities.
- There are several styles of graphics in R. The style in Deepayan Sarkar's *lattice* package is well-suited to the type of data we will be discussing.
- Deepayan's book, *Lattice: Multivariate Data Visualization with R* (Springer, 2008) provides in-depth documentation and explanations of lattice graphics.

The formula/data method of specifying graphics

- The first two arguments to lattice graphics functions are usually `formula` and `data`.

- This specification is used in model-fitting functions (`lm`, `aov`, `lmer`, ...) and in other functions such as `xtabs`.

- The formula incorporates a tilde, ($\sim$), character. A one-sided formula specifies the value on the x-axis. A two-sided formula specifies the x and y axes.

- The second argument, `data`, is usually the name of a data frame.

- Many optional arguments are available. Ones that we will use frequently allow for labeling axes (`xlab`, `ylab`), and controlling the type of information, `type`.

- The `lattice` package is not attached by default. You must enter `library(lattice)` before you can use lattice functions.
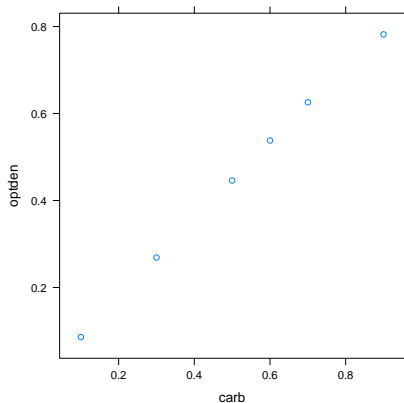
Outline

A simple scatterplot in lattice

```
> xyplot(optden ~ carb, Formaldehyde)
```

Scatterplots in lattice

- A scatter plot is the most versatile plot in applied statistics. It is simply a plot of a numeric response, `y`, versus a numeric covariate, `x`.

- The lattice function `xyplot` produces scatter plots. I typically specify `type = c("g","p")` requesting a background grid in addition to the plotted points.

- The `type` argument takes a selection from

  - "p" points
  - "g" background grid
  - "l" lines
  - "b" both points and lines
  - "r" reference (or "regression") straight line
  - "smooth" scatter-plot smoother lines

- In evaluating a scatterplot the aspect ratio (ratio of vertical size to horizontal size) can be important. In particular, differences in slopes are most apparent near 45°.

General principles of lattice graphics

- The formula is of the form $\sim x$ or $y \sim x$ or $y \sim x \mid f$ where x is the variable on the x axis (usually continuous), y is the variable on the y axis and f is a factor that determines the panels.

- Titles can be added with xlab, ylab, main and sub. Titles can be character strings or, more generally, expressions that allow for special characters, subscripts, superscripts, etc. See help(plotmath) for details.

- The groups argument, if used, specifies different point styles and different line styles for each level of the group. If lines are calculated, each group has separate lines.

- If groups is used, we usually also use auto.key to add a key relating the line or point styles to the groups.

- The layout specifies the number of columns and rows of panels.

An enhanced scatterplot of the Formaldehyde data
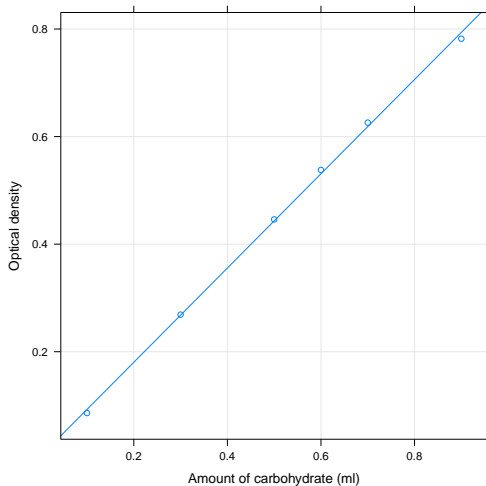
Saving plots

- I recommend using the facilities in the R application to save plots and transcripts.

- To save a plot, ensure that the graphics window is active and use the menu item `File→Save To Clipboard→Windows Metafile`. (On a Mac, save as PDF.) Then switch to a word processor and paste the figure.

- Adjust the aspect ratio of the graphics window to suit the pasted version of the plot before you copy the graphic.

- Those who want more control (and less cutting and pasting) could consider the Sweave system or the odfWeave package.

Outline

Histograms and density plots

- A histogram is a type of bar plot created from dividing numeric data into adjacent bins (typically having equal width).

- The purpose of a histogram is to show the distribution or density of the observations. It is almost never a good way of doing this.

- A `densityplot` is a better way of showing the density or, even better, comparing the densities of observations associated with different groups. Also, densityplots for different groups can be overlaid.

- If you have only a few observations you may want to use a comparative box-and-whisker plot (`bwplot`) or a comparative `dotplot` instead. Density plots based on a small number of observations tend to be rather "lumpy".

- If the data are bounded, perhaps because the data must be positive, a density plot can blur the boundary. However, this may indicate that the data are more meaningfully represented on another scale.
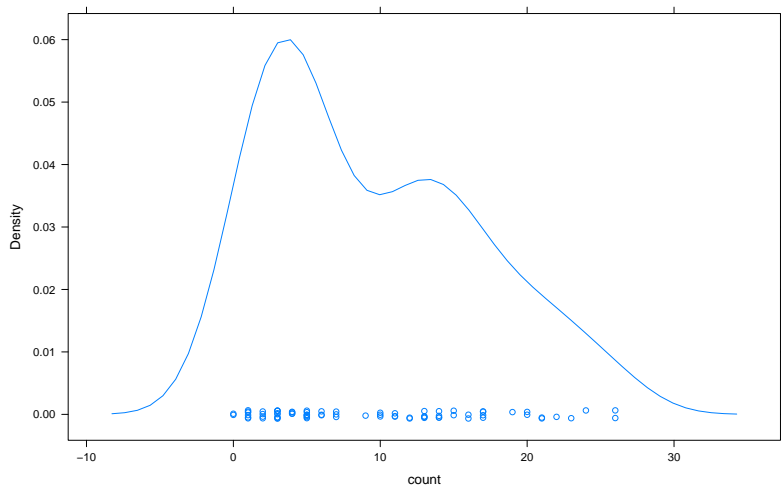
Histogram of the InsectSprays data
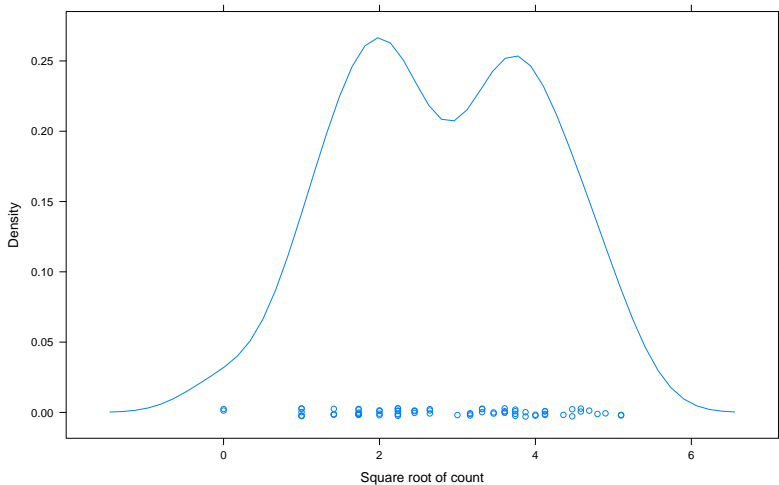
```
> histogram(~count, InsectSprays)
```
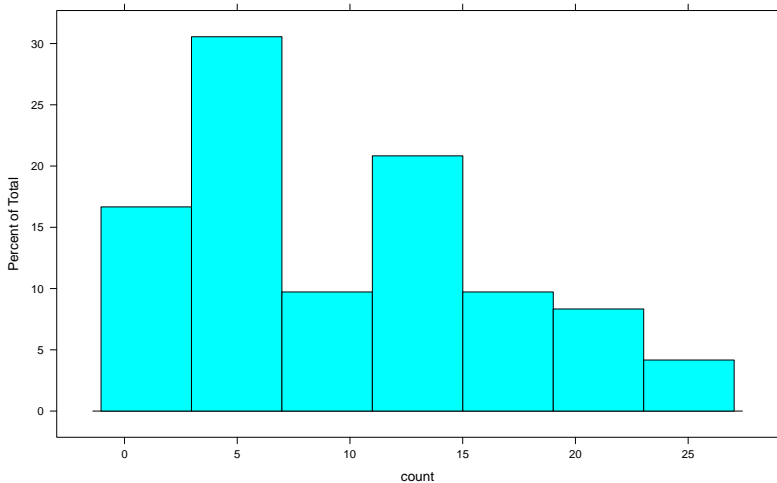
Density plot of the InsectSprays data
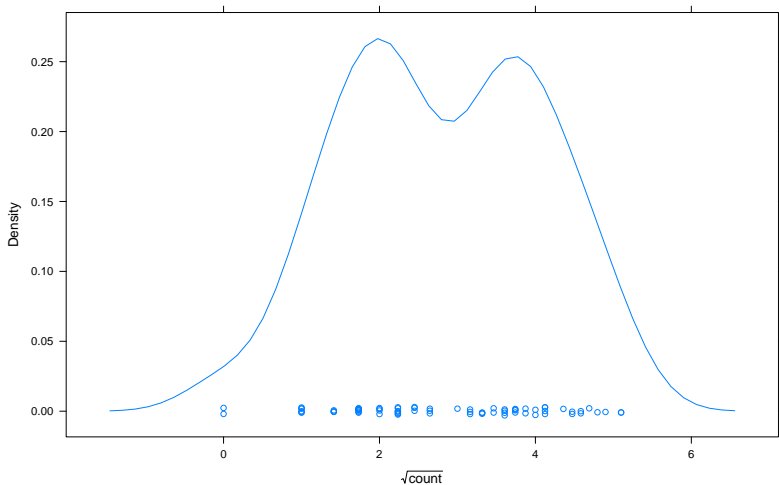
```
> densityplot(~count, InsectSprays)
```

Density plot of the square root of the count

```
> densityplot(~sqrt(count), InsectSprays, xlab = "Square root of
```
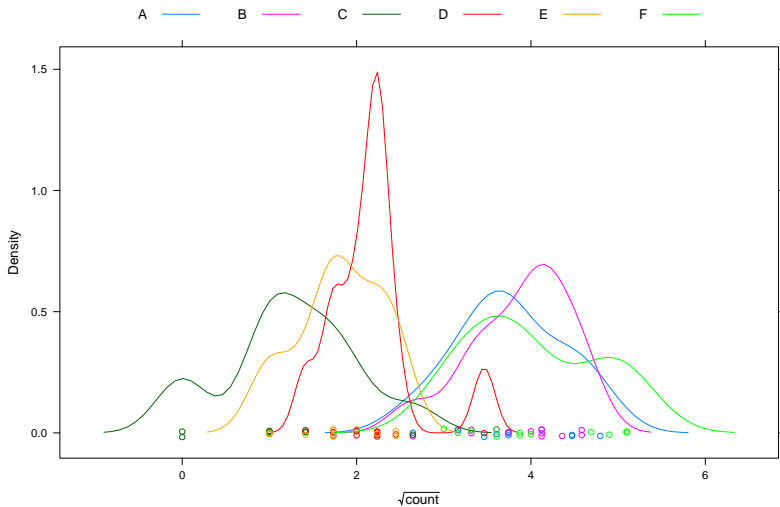
Density plot of the square root with fancy label

```
> densityplot(~sqrt(count), InsectSprays, xlab = expression(sqrt
```
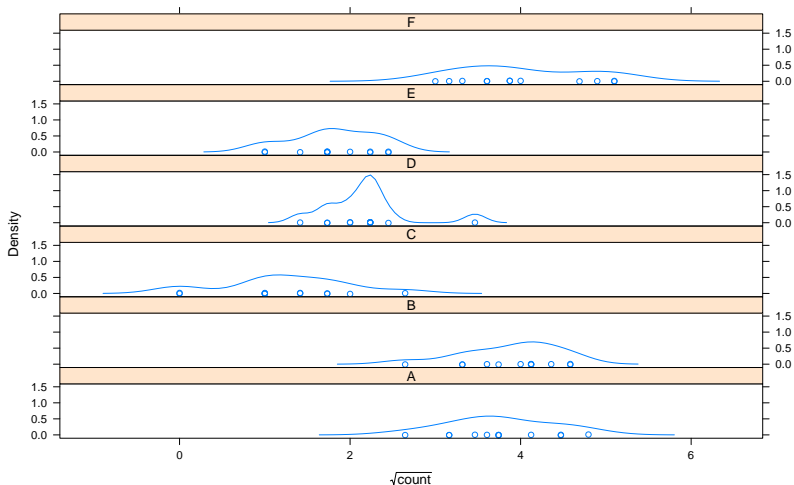
Comparative density plot of square root

```
> densityplot(~sqrt(count), InsectSprays, groups = spray,
+     auto.key = list(columns = 6))
```
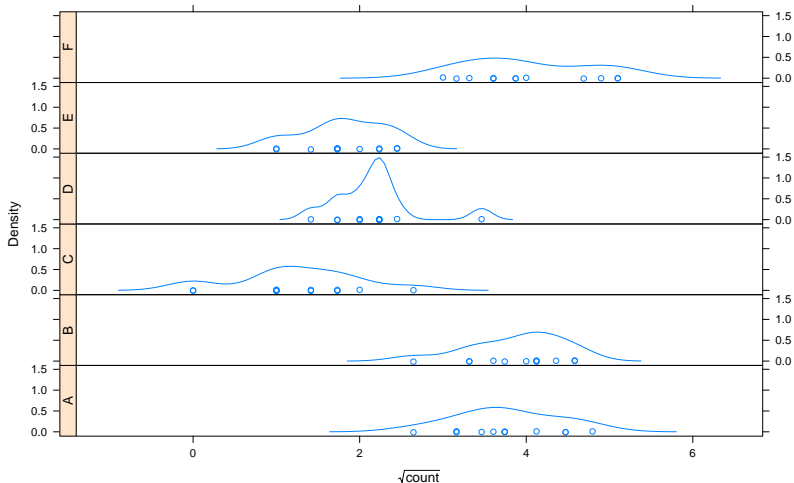
Comparative density plot, separate panels

```
> densityplot(~sqrt(count) | spray, InsectSprays, layout = c(1,
+       6))
```
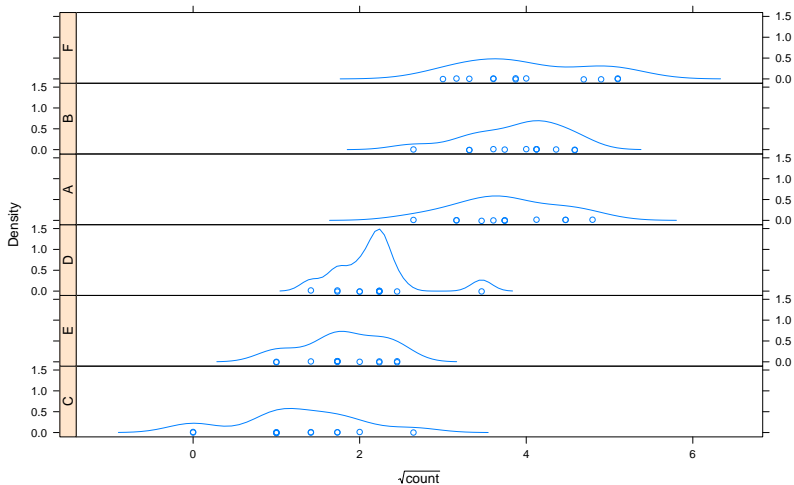
Comparative density plot, separate panels, strip at left

```
> densityplot(~sqrt(count) | spray, InsectSprays, layout = c(1,
+       6), strip = FALSE, strip.left = TRUE)
```

Comparative density plot, separate panels, reordered

```
> densityplot(~sqrt(count) | reorder(spray, count), InsectSprays
```

Outline

Presenting data

Scatter plots

Histograms and density plots

Box-and-whisker plots and dotplots

Box-and-whisker plot and dotplot

- A box-and-whisker plot gives a rough summary (based on the five-number summary - min, 1st quartile, median, 3rd quartile, max) of the distribution.
- A dotplot consists of points on a number line. For a large number of data values we jitter the y values to avoid overplotting. By default a density plot also shows a dotplot.
- Box-and-whisker plots or dotplots are often used for comparison of groups.
- It is widely believed that a comparative boxplot should have the response on the vertical axis. Most of the time it is more effective to put the response on the horizontal axis.
- If the default ordering of the groups is arbitrary reorder them according to the level of the response (mean response, by default).
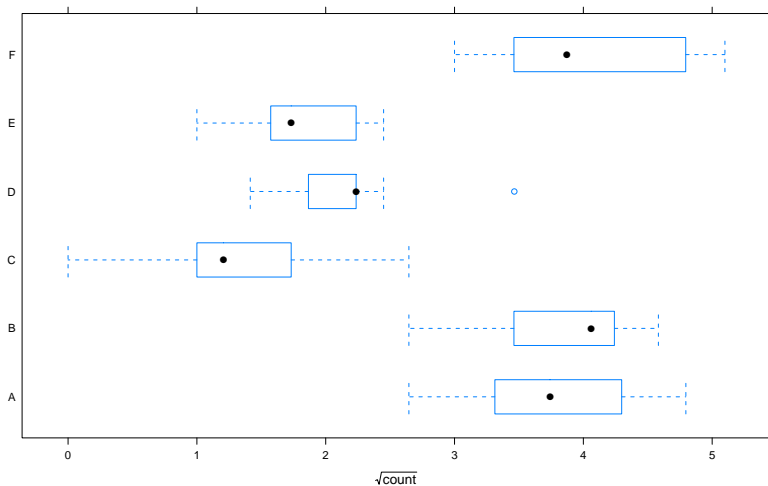- Reordering makes it easier to see if the variability increases with the level of the response.

Vertical comparative box-and-whisker plot

```
> bwplot(sqrt(count) ~ spray, InsectSprays)
```
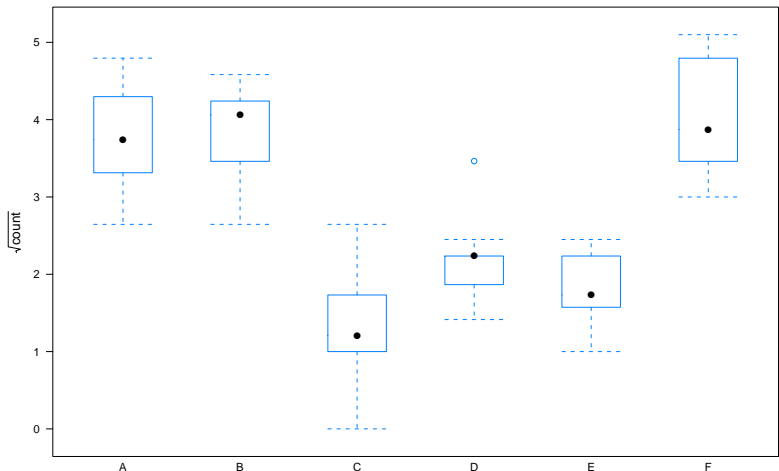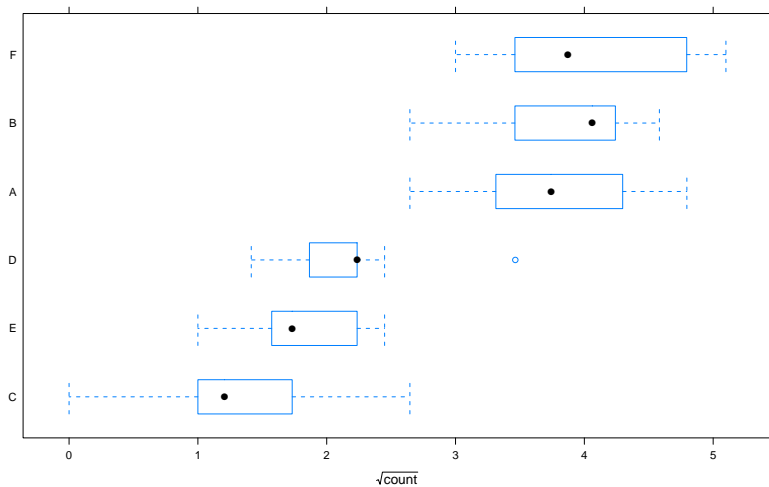
Horizontal comparative box-and-whisker plot

```
> bwplot(spray ~ sqrt(count), InsectSprays)
```
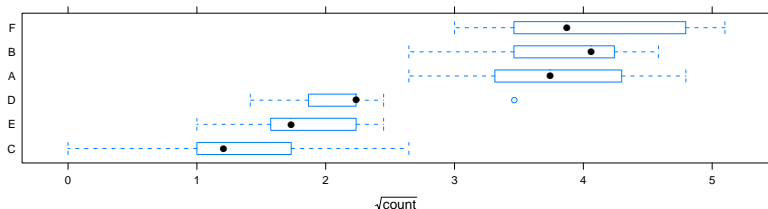
Reordered horizontal comparative box-and-whisker plot

```
> bwplot(reorder(spray, count) ~ sqrt(count), InsectSprays)
```

Compressed horizontal comparative box-and-whisker plot

```
> bwplot(reorder(spray, count) ~ sqrt(count), InsectSprays,
+     aspect = 0.2)
```


$\sqrt{count}$

- You can extract much more information from this, compressed plot than from the original vertical box-and-whisker plot.
- In Edward Tufte's phrase, this plot has a greater "information/ink ratio".

Comparative dotplots

- When the number of observations per group is small, a box-and-whisker plot can obscure the structure of the data, rather than illuminating it.

- By default, the density plot provides a dotplot on the "rug".

- A comparative dotplot displays all of the data. The principles described for a comparative boxplot (factor on vertical axis, reorder levels if no natural order, choose an appropriate scale) apply here too.

- By default, the character in the dotplot is filled. I often use optional arguments `pch = 21` and `jitter.y = TRUE` to avoid overplotting.

- Setting `type = c("p","a")` provides a line joining the group averages.

- Interaction plots can be produced as a comparative dotplot with `groups`

Comparative dotplot of InsectSprays

```
> dotplot(reorder(spray, count) ~ sqrt(count), InsectSprays,
+         type = c("p", "a"), pch = 21, jitter.y = TRUE)
```
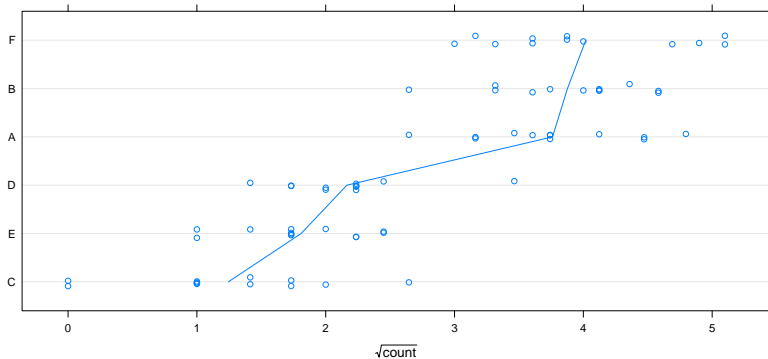
Summary

- In order of importance the graphic displays I consider are scatter plots, density plots, box-and-whisker plots, dot plots and histograms.

- Pay careful attention to layout and axis labels. Include units in the axis labels, if known.

- For mixed models we always have at least one unordered categorical covariate and often have a numeric response. Comparative dot plots and box-and-whisker plots will be important data presentation techniques for us.

- Plots of a continuous response by levels of a categorical variable work best with the category on the vertical axis. Consider reordering the levels of the category if they do not have a natural order.