Data visualization, focusing on ggplot and multilevel data

Ben Bolker

15:48 20 April 2018

Goals/contexts of data visualization

Exploration

- want *nonparametric/robust* approaches: impose as few assumptions as possible
- boxplots instead of mean/standard deviation (generally base locations on medians rather than means)
- loess/GAM instead of linear/polynomial regression
- need *speed*: quick and dirty
- canned routines for standard tasks, flexibility for non-standard tasks
- manipulation in the context of visualization: need to summarize on the fly

Diagnostics

- attempt to determine fitting problems graphically: looking for absence of patterns in residuals
- e.g. scale-location plot, Q-Q plot; `plot.lm`
- plot methods: generic (e.g. residuals vs fitted) vs specific (e.g. residuals vs predictors)
- plotting predictions (intuitive) vs plotting residuals (amplifies/zooms in on discrepancies)
- plotting unmodeled characteristics (e.g. spatial, temporal autocorrelation): much easier to draw a picture than fit a model
- code contrasts for visual simplicity (e.g. deviations from linearity: Q-Q plots, signed square-root profiles)

Presentation

- how closely should one match analyses with graphs? “Let the data speak for themselves” vs “Tell a story”
- display data (e.g. boxplots, standard deviations) or inferences from data (confidence intervals)
- superimposing model fits (`geom_smooth`)
- avoid excessive cleverness/data density
- coefficient plots vs parameter tables (Gelman, Pasarica, and Dodhia 2002)
- tradeoff between visual design (tweaking) and reproducibility: learning to futz with label positioning etc. may pay off in the long run (a few tools exist for automatic placement)
- order factors in a sensible order (i.e *not* alphabetical or numerical unless (1) the labels have some intrinsic meaning or (2) you expect that readers will be interested in looking up particular levels in the plot). This is sometimes called the “what’s so special about Alabama?” problem, although the Canadian version would substitute “Alberta”.

Basic criteria for data presentation

Visual perception of quantitative information: Cleveland hierarchy (W. S. Cleveland and McGill 1984,W. S. Cleveland and McGill (1987),W. Cleveland (1993))

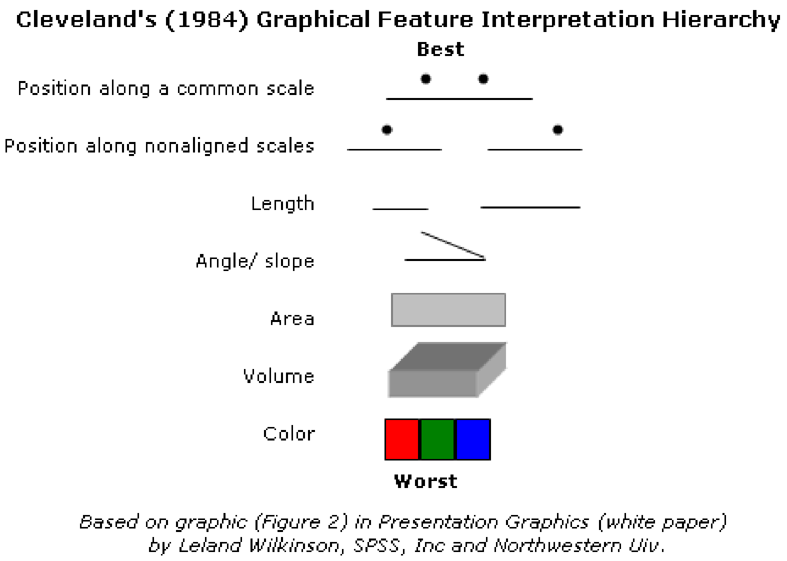

Cleveland's (1984) Graphical Feature Interpretation Hierarchy
Best
Position along a common scale
Position along nonaligned scales
Length
Angle/ slope
Area
Volume
Color
Worst
Based on graphic (Figure 2) in Presentation Graphics (white paper)
by Leland Wilkinson, SPSS, Inc and Northwestern Uiv.

Figure 1: cleveland

Data presentation scales with data size

- **small** show all points, possibly dodged/jittered, with some summary statistics: dotplot, beeswarm. Simple trends (linear/GLM)
- **medium** boxplots, loess, histograms, GAM (or linear regression)
- **large** modern nonparametrics: violin plots, hexbin plots, kernel densities: computational burden, and display overlapping problems, relevant
- combinations or overlays where appropriate (beanplot)

Rules of thumb

- (Continuous) response on the y-axis, most salient (continuous) predictor on the x-axis
- Put most salient comparisons within the same subplot (distinguished by color/shape), and nearby within the subplot when grouping bars/points
- Facet rows > facet columns
- Use transparency to include important but potentially distracting detail
- Do category levels need to be *identified* or just *distinguished*?
- Order categorical variables meaningfully
- Display *population variation* (standard deviations, boxplots) vs. *estimate variation* (standard errors, mean ± 2 SE, boxplot notches)
- Try to match graphics to statistical analysis, but not at all costs
- Choose colors carefully (`RColorBrewer`/ColorBrewer, IWantHue: respect dichromats and B&W printouts

Techniques for multilevel data

- faceting (= trellis plots = small multiples) vs grouping ("spaghetti plots")
- join data within a group by lines (perhaps thin/transparent)
- can colour lines by group (get legend), but more useful for explanatory than presentation graphics

```
data("cbpp", package = "lme4")
## make period *numeric* so lines will be
## connected/grouping won't happen
cbpp2 <- transform(cbpp, period = as.numeric(as.character(period)))
g0 <- ggplot(cbpp2, aes(period, incidence/size))
## spaghetti plot
g1 <- g0 + geom_line(aes(colour = herd)) + geom_point(aes(size = size,
    colour = herd))
g2 <- ggplot(cbpp2, aes(period, incidence/size,
    colour = herd))
(g3 <- g2 + geom_line() + geom_point(aes(size = size)))
```

```
## facet instead
(g4 <- g1 + facet_wrap(~herd))
## order by average prop. incidence
g1 %+% transform(cbpp2, herd = reorder(herd, incidence/size))
g4 %+% transform(cbpp2, herd = reorder(herd, incidence/size))
## also consider colouring by incidence/order
## ...
```

Makes it fairly easy to do a simple *two-stage* analysis on the fly:

```
g0 + geom_point(aes(size = size, colour = herd)) +
    geom_smooth(aes(colour = herd, weight = size),
        method = "glm", method.args = list(family = binomial),
        se = FALSE)
```

(ignore `glm.fit` warnings if you try this)

You can do similar stuff with the `lattice` package:

```
library(lattice)
xyplot(incidence/size ~ period, group = herd,
    data = cbpp, type = "l", auto.key = TRUE)  ## need to mess around to get key right
xyplot(incidence/size ~ period | herd, data = cbpp,
    type = "l")
```

Challenges

high-dimensional data (esp continuous)

Possible solutions:

- use color, shape for discrete predictor variables (up to ~10 categories); text plots
- small multiples, conditioning plots (shingles/facets); i.e. discretize continuous plots
- contour plots (worse)
- perspective plots (worst?)

large data sets

- problems with computation, file size, presentation
- file size: raster (PNG) instead of vector (PS/PDF), `pch="."`
- overplotting (alpha), kernel density estimation, hexagonal binning
- summarize (quantiles, kernel densities, etc.)

discrete data

- lots of point overlap; jittering OK for exploratory analysis but ugly. Need to summarize/bin appropriately (`stat_sum`); beeswarm plots

spatial data

- the best parts of the Cleveland hierarchy (x and y axes) are already taken, usually have to resort to color/size/pie charts. Representing uncertainty is a big challenge, usually must be done separately (transparency/saturation?)

compositional data

- would like to display "sum to 1.0" constraint but also allow accurate comparison of magnitudes: stacked bars vs grouped bars (or dotplots)?
- harder if also need to represent uncertainty (would like to show correlations among components)
- *ternary diagrams*: nice but don't generalize past 3 elements

multilevel data

- often hard (or messy) to represent all levels of variation

next generation tools

- dynamic/exploratory graphics: ggobi, Mondrian, [latticist] (`http://code.google.com/p/latticist`), JMP, Shiny, Gapminder, googleVis, rCharts, ggviz (next-generation/interactive `ggplot2`)
- GUI frameworks: JMP, R Commander, Deducer, Rattle, web interface to ggplot2, Shiny
- presentation technologies: JCGS editorial with supplementary materials
- computational frameworks: `lattice`, `ggplot`, http://d3js.org/, `ggviz`

Graphics culture

- the gods: Cleveland (hierarchy), Tufte (E. Tufte 2001; E. R. Tufte 1995; E. R. Tufte 1997; E. R. Tufte 2006) ("chartjunk", minimizing non-data-ink, sparklines)
- the demi-gods: Wilkinson (Grammar of Graphics) (Wilkinson 1999), Wickham (ggplot *et al*) (Wickham 2009), Stephen Few, Kaiser Fung (Junk Charts)
- dionysians (infovis) vs. apollonians (statistical graphics) (Gelman and Unwin 2013)
- graphics pedants: dynamite plots, pie charts (esp. 3D), dual-axis figures . . .

Data visualization in R

Base graphics

- simple 'canvas' approach
- straightforward, easy to customize
- most plot methods written in base graphics

Lattice

- newer
- documented in a book (Sarkar 2008)
- based on `grid` graphics package
- faceting, conditioning plots
- much more automatic, better graphical defaults
- implements banking, other aspect ratio control
- more 'magic', harder to customize
- some plot methods (nlme package)
- `latticeExtra`, `directlabels` packages may be handy

ggplot

- newest
- based on Wilkinson's ''Grammar of Graphics''
- documented in a book (see below) and on a web site, as well as an active mailing list
- explicit mapping from variables to ''aesthetics'': x, y, colour, size, shape
- implements faceting (not quite as flexibly as lattice: no aspect ratio control)
- some data summaries etc. built in
- easier to overlay multiple data sets, data summaries, model predictions etc.
- no 3D plots
- rendering can be slow
- `gridExtra`, `ggExtra`, `cowplot`, `directlabels` packages may be handy
- ggplot gallery

ggplot intro

mappings + geoms

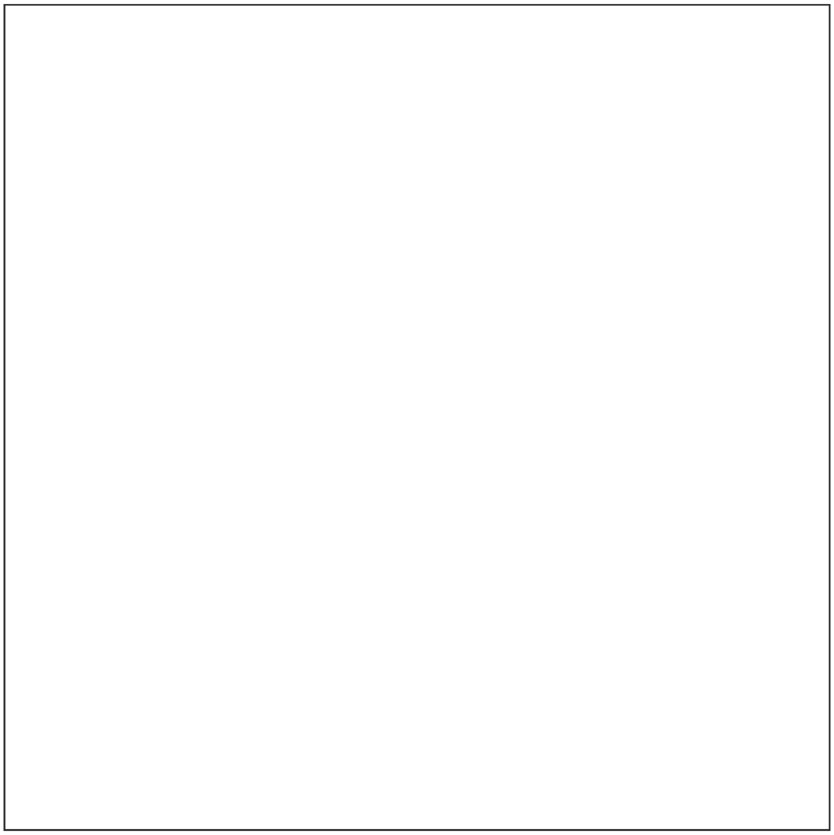
Data

Specified explicitly as part of a `ggplot` call:

```
library(mlmRev)
head(Oxboys)
```

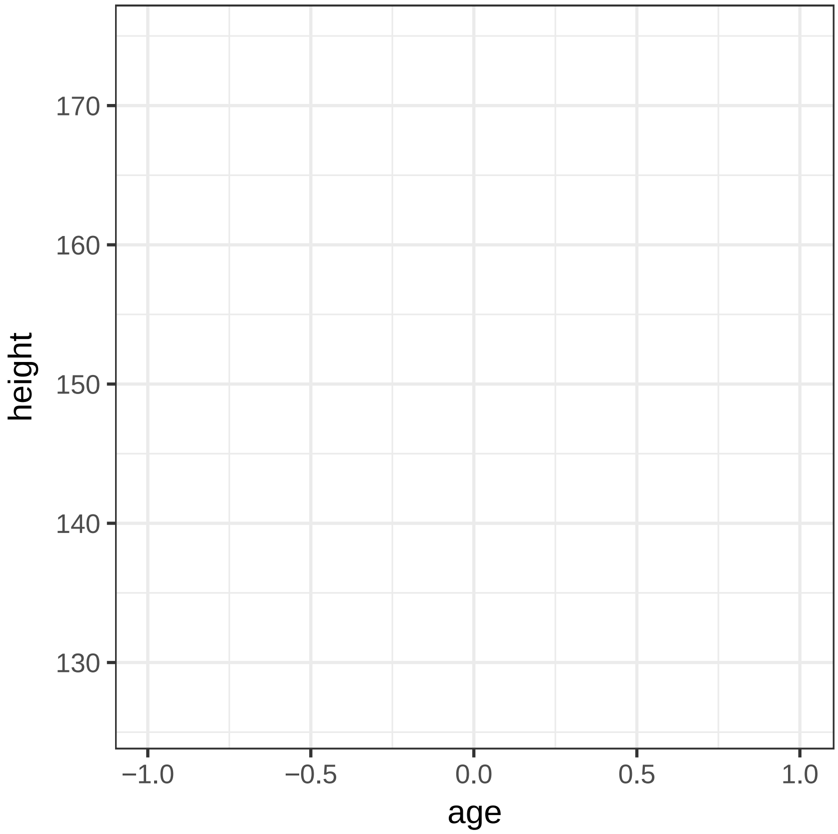
```
##   Subject     age height Occasion
## 1       1 -1.0000  140.5        1
## 2       1 -0.7479  143.4        2
## 3       1 -0.4630  144.8        3
## 4       1 -0.1643  147.1        4
## 5       1 -0.0027  147.7        5
## 6       1  0.2466  150.2        6
```

```
library(ggplot2)
ggplot(Oxboys)
```



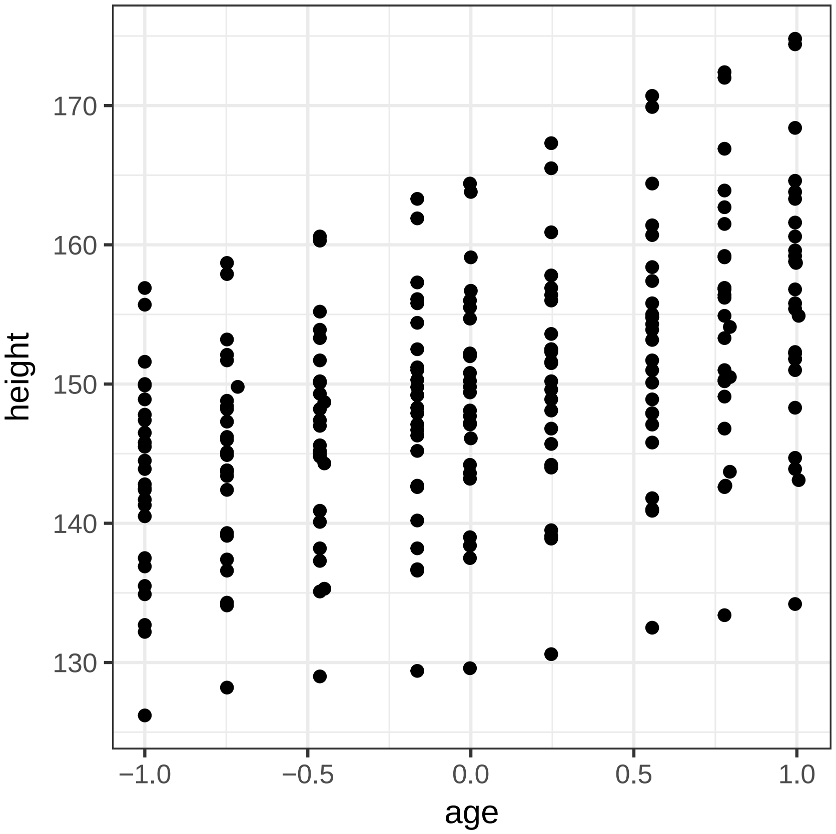
But that isn't quite enough: we need to specify a *mapping* between variables (columns in the data set) and *aesthetics* (elements of the graphical display: x-location, y-location, colour, size, shape . . .)

```
ggplot(Oxboys, aes(x = age, y = height))
```



but (as you can see) that's still not quite enough. We need to specify some geometric objects (called `geoms`) such as points, lines, etc., that will embody these aesthetics. The weirdest thing about `ggplot` syntax is that these `geoms` get *added* to the existing `ggplot` object that specifies the data and aesthetics; unless you explicitly specify other aesthetics, they are inherited from the initial `ggplot` call.

```
ggplot(Oxboys, aes(x = age, y = height)) + geom_point()
```



- many more geoms (lines, bars, etc.)
- summarizers: smooth lines and summaries (`geom_smooth`, `stat_sum`)
- control of scales (e.g. log transforms, colors, etc.)
- faceting (grid and wrap)

See Karthik Ram's ggplot intro or my intro for disease ecologists, among many others.

```
sessionInfo()
```

```
## R Under development (unstable) (2018-04-16 r74611)
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: Ubuntu 16.04.4 LTS
##
## Matrix products: default
## BLAS: /usr/local/lib/R/lib/libRblas.so
## LAPACK: /usr/local/lib/R/lib/libRlapack.so
##
## locale:
##  [1] LC_CTYPE=en_CA.UTF8
##  [2] LC_NUMERIC=C
##  [3] LC_TIME=en_CA.UTF8
##  [4] LC_COLLATE=en_CA.UTF8
##  [5] LC_MONETARY=en_CA.UTF8
##  [6] LC_MESSAGES=en_CA.UTF8
##  [7] LC_PAPER=en_CA.UTF8
##  [8] LC_NAME=C
##  [9] LC_ADDRESS=C
## [10] LC_TELEPHONE=C
## [11] LC_MEASUREMENT=en_CA.UTF8
## [12] LC_IDENTIFICATION=C
##
## attached base packages:
## [1] stats     graphics  grDevices utils
## [5] datasets  methods   base
##
## other attached packages:
## [1] mlmRev_1.0-6       lme4_1.1-17
## [3] Matrix_1.2-14      lattice_0.20-35
## [5] ggplot2_2.2.1.9000 knitr_1.20
##
## loaded via a namespace (and not attached):
##  [1] Rcpp_0.12.16      magrittr_1.5
##  [3] MASS_7.3-49       splines_3.6.0
##  [5] munsell_0.4.3     colorspace_1.3-2
##  [7] rlang_0.2.0.9001  minqa_1.2.4
##  [9] stringr_1.3.0     plyr_1.8.4
## [11] tools_3.6.0       grid_3.6.0
## [13] nlme_3.1-137      gtable_0.2.0
## [15] withr_2.1.2       htmltools_0.3.6
## [17] yaml_2.1.18       lazyeval_0.2.1
## [19] rprojroot_1.3-2   digest_0.6.15
## [21] tibble_1.4.2      nloptr_1.0.4
## [23] formatR_1.5       evaluate_0.10.1
## [25] rmarkdown_1.9     labeling_0.3
## [27] stringi_1.1.7     compiler_3.6.0
## [29] pillar_1.2.1      scales_0.5.0.9000
## [31] backports_1.1.2   tufte_0.3
```

References

Cleveland, William. 1993. *Visualizing Data*. Summit, NJ: Hobart Press.

Cleveland, William S., and Robert McGill. 1984. "Graphical Perception: Theory, Experimentation, and Application to the Development of Graphical Methods." *Journal of the American Statistical*

Association 79 (387): 531–54. doi:10.2307/2288400.

———. 1987. "Graphical Perception: The Visual Decoding of Quantitative Information on Graphical Displays of Data." *Journal of the Royal Statistical Society. Series A (General)* 150 (3): 192–229. doi:10.2307/2981473.

Gelman, Andrew, and Antony Unwin. 2013. "Infovis and Statistical Graphics: Different Goals, Different Looks." *Journal of Computational and Graphical Statistics* 22 (1): 2–28. doi:10.1080/10618600.2012.761137.

Gelman, Andrew, Cristian Pasarica, and Rahul Dodhia. 2002. "Let's Practice What We Preach: Turning Tables into Graphs." *The American Statistician* 56 (2): 121–30. `http://www.jstor.org/stable/3087382`.

Sarkar, Deepayan. 2008. *Lattice: Multivariate Data Visualization with R*. 1st ed. Springer.

Tufte, Edward. 2001. *The Visual Display of Quantitative Information*. 2d ed. Graphics Press.

Tufte, Edward R. 1995. *Envisioning Information*. Cheshire, Conn.: Graphics Press.

———. 1997. *Visual Explanations: Images and Quantities, Evidence and Narrative*. Cheshire, Conn.: Graphics Press.

———. 2006. *Beautiful Evidence*. Cheshire, Conn.: Graphics Press.

Wickham, Hadley. 2009. *Ggplot2: Elegant Graphics for Data Analysis*. 2nd Printing. Springer.

Wilkinson, L. 1999. *The Grammar of Graphics*. New York: Springer.