Today: Hierarchical Clustering, Base R Graphics

Sam Ventura

36-315

Today: Lab Exam Notes, Hierarchical Clustering, Base R Graphics

Department of Statistics
Carnegie Mellon University

October 26, 2016

Hierarchical Clustering

As we will see on the upcoming lab and HW, we can project the data into lower-dim space and visualize the results. Why might this be a bad idea?

Another option: Use **hierarchical linkage clustering**.

Single linkage: the distance between two groups is the shortest possible distance between two points, one from each group

Complete linkage: the distance between two groups is the largest possible distance between two points, one from each group

Reminder: Why We Use ggplot()

There are many reasons:

1. "Grammar of graphics"
2. Default graphs / colors / etc are already nice
3. Can easily change geometry of plot without changing the code
4. Can save plots – including portions of plots – as objects
5. Easy to perform multivariate exploration (coloring, faceting, etc)
6. Legends are generated and updated automatically when you change your graphic
7. Can build plots in layers (e.g. add points, then add 2D density estimate, then add regression lines, error bars, etc)
8. Documentation is detailed; easy to find help/tutorials online
9. "All the cools kids are doing it"

Base R Graphics

Base R graphics are the standard way to create plots in R

Generally, these use the plot() function to plot specific pieces of data

Big difference from ggplot(): plot() requires vectors for each argument, while ggplot() uses data.frames

plot(x = data$variable1, y = data$variable2, col = data$color_variable, pch = data$point_type_variable)

Naming conventions in base R graphics are often un-intuitive

Some more advanced statistical methods have specific types of plots implemented in base-R graphics; ggplot() is relatively new, so some of these are not yet implemented in ggplot()