Graphics in R

• There are three plotting systems in R
  – base
    • Convenient, but hard to adjust after the plot is created
  – lattice
    • Good for creating conditioning plot
  – ggplot2
    • Powerful and flexible, many tunable feature, may require some time to master

• We will focus on the ggplot2 package today
Basic Concepts of `ggplot2`

- An implementation of Grammar of Graphics by Leland Wilkinson, which represents abstraction of graphic objects and ideas
- Two ways of plotting
  - The `qplot` function: similar to the base plotting system with more features
  - The `ggplot` function: the engine of `qplot` - powerful, flexible and full featured
- Make the plots in layers
  - Plot the data
  - Overlay a summary
  - Metadata and annotation
ggplot2 Components

- **Data frame**
- **Aesthetic mappings**: how data are mapped to color, size
- **Geoms**: geometric objects like points, lines, shapes.
- **Facets**: for conditional plots.
- **Stats**: statistical transformations like binning and smoothing.
- **Scales**: what scale an aesthetic map uses (example: male = red, female = blue).
- **Coordinate system**
Example Data Frame

- Benchmark data on the SuperMIC cluster
  - Four columns: nt (number of threads), time, device ("mic" or "cpu"), node (number of nodes)

```r
> head(bmData, 3)
   nt  time device node
 1  1 33358.562    mic    1
 2  5  6894.032    mic    1
 3 10  3566.916    mic    1

> tail(bmData, 3)
   nt  time device node
182 18 10.237771    cpu    16
183 19  9.936815    cpu    16
184 20  9.452416    cpu    16
```
qplot Function

> qplot(nt,bmData[1,"time"]/time,
  data=subset(bmData,node==1 & device=="mic"))
Using ggplot2

```r
> g <- ggplot(subset(bmData, device == "mic"),
             aes(nt, speedup, color = factor(node)))
> summary(g)
data: nt, time, device, speedup, node [104x5]
mapping: x = nt, y = speedup, colour = factor(node)
faceting: facet_null()
> g # No plot here yet
Error: No layers in plot
> g + geom_line() + geom_point() # Add geometric objects
```
The same plot with decorations
g <- ggplot(subset(bmData, device=="mic"), aes(nt,speedup,color=factor(node)))  
# ggplot call

g + geom_line(size=1) +  # plot lines
geom_point(size=5) +  # plot points
# Title and axis labels
 labs(x="Number of OpenMP Threads", y="Speedup", title="SuperMIC Benchmark") +
# Format the title, legends and axes
 theme(plot.title = element_text(colour="black", size=22, face="bold", vjust=2.5),
       legend.title = element_text(colour="black", size=16, face="bold"),
       legend.position=c(0.2,0.8),
       legend.text=element_text(face="italic", size=16),
       axis.title=element_text(size=16)) +

scale_color_discrete(name="Offload to Intel Xeon Phi", # Legend title and keys
                     labels=c("1 node", "4 nodes", "9 nodes", "16 nodes")) +

scale_x_continuous(limits=c(1, max(bmData$nt)), # Limits and ticks of x axis
                   breaks=c(1,seq(20,max(bmData$nt),20))) +

scale_y_continuous(limits=c(1, max(bmData$speedup)), # Limits and ticks of y axis
                   breaks=c(1,seq(20,max(bmData$speedup),20))) +
annotate("text", x=200, y=30, # Annotation text
          label="MIC KMP AFFINITY=\n"granularity=fine,balanced\"", size=8)

Here's the code
SuperMIC Benchmark

16 MPI Tasks on 16 Nodes

- Intel Xeon Phi
- Intel Xeon

Plot with inset
(The code can be found in the script)
Plot with a 2x2 facet
(The code can be found in the script)
Plot Functions

• Plot a math function by using the `stat_function()` function

```r
p <- ggplot(data.frame(x=c(-3,3)), aes(x=x))
p + stat_function(fun = dnorm)
```
Control The Output Format

• Graphic Devices
  – File devices
    • Bitmap: png, jpeg, etc.
    • Vector: pdf, svg, etc.
  – Screen devices (quartz, X11, etc.)

• Steps
  – Launch a graphic device
    • Functions: `png()`, `pdf()`, `svg()`, etc.
  – Make the plot
  – Close the device with `dev.off()` function
R Scripts

• We can put all the commands in a script
• Run R scripts with the `Rscript` command

```bash
[lyan1@qb4 Gnuplot-training]$ ls
benchmark.R map3d.eps map3d.pdf multiplot.gpl time-01-node-smic.dat time-09-node-smic.dat
input3d.dat map3d.gpl multiplot.eps multiplot.pdf time-04-node-smic.dat time-16-node-smic.dat
[lyan1@qb4 Gnuplot-training]$ Rscript benchmark.R
Warning messages:
1: Removed 22 rows containing missing values (geom_path).
2: Removed 22 rows containing missing values (geom_point).
[lyan1@qb4 Gnuplot-training]$ ls -tr
input3d.dat map3d.pdf time-04-node-smic.dat multiplot.eps
map3d.gpl multiplot.gpl time-09-node-smic.dat multiplot.pdf
map3d.eps time-01-node-smic.dat time-16-node-smic.dat benchmark.R
```

```bash
smic_benchmark_1.pdf
smic_benchmark_2.svg
smic_benchmark_3.png
```