Data Visualization in R

Vanilla R vs. ggplot2

- Vanilla R or base R – a blank canvas you build on
- ggplot2 – set of tools to use with constraints

Vanilla R vs. ggplot2

Making Figures with Vanilla R

Open and closing an output PDF file

```
> pdf.options(family="Helvetica")
> pdf("output_file.pdf", pointsize=10)
…
> dev.off()
```

NOTE: > to indicates an R command

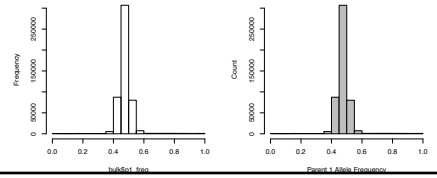
* It can be easy to forget dev.off(). If you are working in RStudio and you can't produce chart it may be because you forgot the dev.off() command after using pdf().

Making a scatter plot between two variables

```
> plot(var$col_1, var$col_2)
> plot(var$col_1, var$col_2, xlab="Column 1", ylab="Column 2")
```

Making a histogram plot

```
> hist(var$col_1)
> hist(var$col_1, xlab="P1 Allele Frequency", ylab="Count", col="gray")
```



Other Features That Can Be Helpful

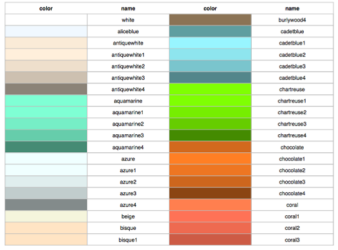
Adding lines to a plot

```
> lines(c(2,7),c(0, 500), col="red", lty=1, lwd=2)
```

Adding boxes to a plot

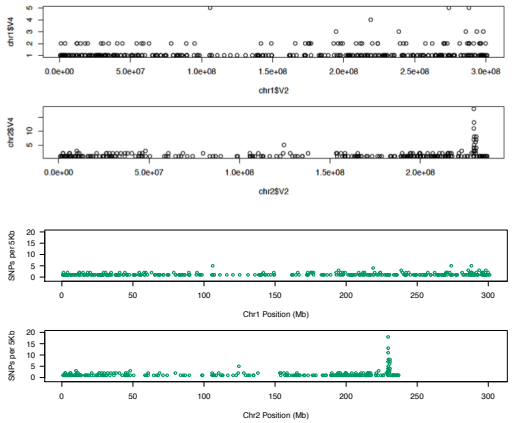
```
> rect(455861, -0.1, 13854546, 1.1, col="#99330025", border=NA)
```

R_lines.pdf



R_colors.pdf or hex colors

Putting It All Together



Putting It All Together

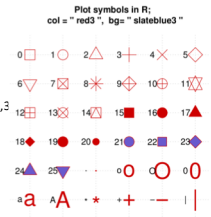
```
> snp_window<-read.delim(file="snps_per_5kb.txt", sep="\t", head=F)
> chr1<-subset(snp_window, V1=="1")
> chr2<-subset(snp_window, V1=="2")
…

> pos<-c(0,50000000,100000000,150000000,200000000,250000000,300000000,3
> lab<-c(0,50,100,150,200,250,300,350)
> mark<-c(0,5,10,15,20)

> pdf.options(family="Helvetica")
> pdf(file="distribution_5kb_window.pdf")
> par(mfrow=c(5,1), mar=c(5,4,0.2,0.2))

> plot(chr1$V2, chr1$V4, ylim=c(-1,20), xlim=c(1,301354135), xlab="Chr1 Position (Mb)", ylab="SNPs per
5Kb", xaxt="n", yaxt="n", pch=21, col="#009966", bg="#00996625", cex=0.75)
> axis(1,at=pos,labels=lab)
> axis(2,at=mark,labels=mark, las=1)

> plot(chr2$V2, chr2$V4, ylim=c(-1,20), xlim=c(1,301354135), xlab="Chr2 Position (Mb)", ylab="SNPs per
5Kb", xaxt="n", yaxt="n", pch=21, col="#009966", bg="#00996625", cex=0.75)
> axis(1,at=pos,labels=lab)
> axis(2,at=mark,labels=mark, las=1)
…
> dev.off()
```



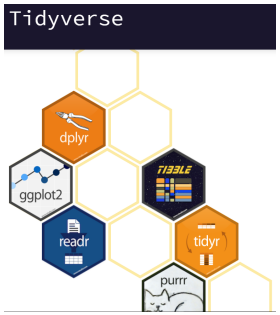
Let's see some of this live…

ggplot2 introduction

"ggplot2 is a plotting system for R, based on the grammar of graphics, which tries to take the good parts of base graphics and none of the bad parts. It takes care of many of the fiddly details that make plotting a hassle as well as providing a powerful model of graphics that makes it easy to produce complex multi-layered graphics."

-Hadley Wickham

ggplot2 introduction



- The "Tidyverse" is a collection of R packages designed using the Wickham philosophy
- **ggplot2** is the package in the Tidyverse collection used for plotting

```
# Install and load all at once
> install.packages("tidyverse")
> library("tidyverse")

# Install single package
> install.packages("ggplot2")
> library("ggplot2")
```

Wide vs. Long format

ggplot and other plotting packages like data in long format

```
> install.packages("reshape2")
> library("reshape2")
> my_data_long <- melt(my_data)
```

Wide-format

	Genotype	Tissue	Mapped	Unmapped
1	B73	shoot	0.82	0.06
2	Mo17	shoot	0.69	0.05
3	Mo17xB73	shoot	0.77	0.06
4	B73	root	0.85	0.05
5	Mo17	root	0.67	0.05
6	Mo17XB73	root	0.80	0.05
7	B73	internode	0.87	0.06
8	Mo17	internode	0.72	0.05
9	Mo17xB73	internode	0.80	0.05
10	B73	leaf	0.85	0.07
11	Mo17	leaf	0.69	0.06
12	Mo17xB73	leaf	0.75	0.06

Long-format

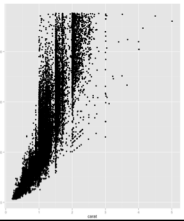
	Genotype	Tissue	variable	value
1	B73	shoot	Mapped	0.82
2	Mo17	shoot	Mapped	0.69
3	Mo17xB73	shoot	Mapped	0.77
4	B73	root	Mapped	0.85
5	Mo17	root	Mapped	0.67
6	Mo17XB73	root	Mapped	0.80
7	B73	internode	Mapped	0.87
8	Mo17	internode	Mapped	0.72
9	Mo17xB73	internode	Mapped	0.80
10	B73	leaf	Mapped	0.85
11	Mo17	leaf	Mapped	0.69
12	Mo17xB73	leaf	Mapped	0.75
13	B73	shoot	Unmapped	0.06
14	Mo17	shoot	Unmapped	0.05
15	Mo17xB73	shoot	Unmapped	0.06
16	B73	root	Unmapped	0.05
17	Mo17	root	Unmapped	0.05
18	Mo17XB73	root	Unmapped	0.05
19	B73	internode	Unmapped	0.06
20	Mo17	internode	Unmapped	0.05
21	Mo17xB73	internode	Unmapped	0.05
22	B73	leaf	Unmapped	0.07
23	Mo17	leaf	Unmapped	0.06
24	Mo17xB73	leaf	Unmapped	0.06

Quick ggplot

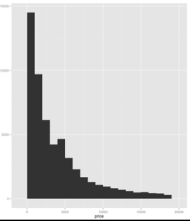
qplot is a quick plotting function within ggplot2 that is similar to the plot() function in base R

Best for getting a quick glance at your data

```
> qplot(diamonds$carat,  diamonds$price)
```



```
> qplot(price, data = diamonds, geom=
"histogram", binwidth=1000)
```



ggplot philosophy

All ggplot2 plots start with ggplot()

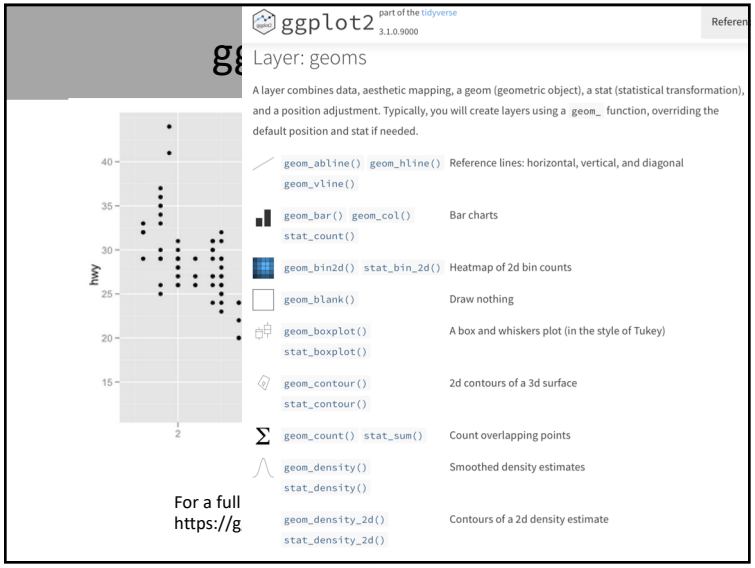
- data – *data frame* containing data to be plotted
- aes() – *aesthetic* mappings to pass on the plot elements

```
> ggplot(mpg, aes(displ, hwy))
Error: No layers in plot
```

ggplot needs "layers" in the plot

- The geometric object (the type of plot you want) is a required layer

```
> ggplot(mpg, aes(displ, hwy)) + geom_point()
```

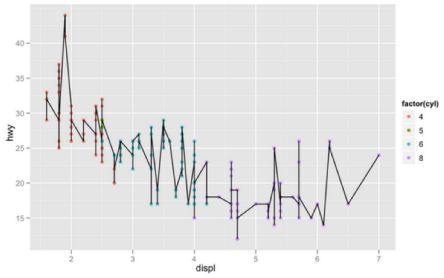


ggplot – intro to layers

Within a "geom" you can add new aesthetics

- geom_line() – add line to a point graph
- aes(color = factor(cyl)) – color the points by cylinder type

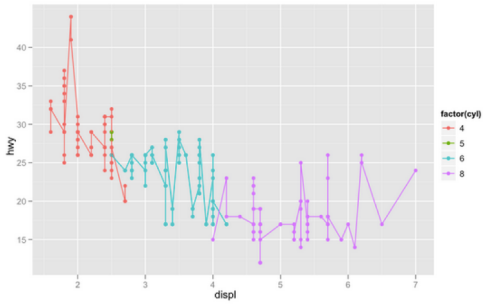
```
> ggplot(mpg, aes(displ, hwy)) + geom_point(aes(color = factor(cyl))) + geom_line()
```



ggplot – adding layers

We can also use the aesthetic mapping within the ggplot function to color both points and lines

```
> ggplot(mpg, aes(displ, hwy, color=factor(cyl))) + geom_point() + geom_line()
```



ggplot – additional customization

Many other "layers" can be added to customize a plot

Change the overall "look of plot"

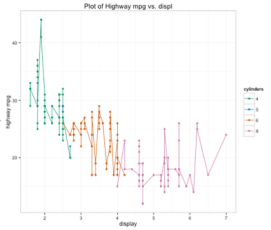
```
p <- ggplot(mpg, aes(displ, hwy, color=factor(cyl))) + geom_point() + geom_line() + theme_bw()
p + scale_colour_manual(name="cylinders", values=c("#009E73", "#0072B2", "#D55E00", "#CC79A7")) +
  labs(title="Plot of Highway mpg vs. displ", x="displ", y = "highway mpg", fill="my_legend")
```

Add custom line colors

Customize legend title

Add labels to the axis and title

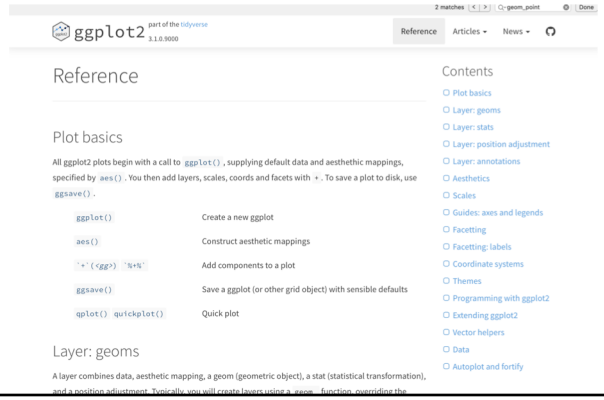
```
#plot intersection count per Mb of chromosome
ggplot(tempdata, aes(x=factor(chrom),y=pos1,z=value))+stat_summary_2d(bins=301, fun="mean")+
  scale_x_discrete(name="Chromosome",limits=c("chr1","chr2","chr3","chr4","chr5","chr6","chr7","chr8","chr9","chr10"))+
  labs(title = "Frequency Variants")+
  scale_y_continuous(name="Physical Position (Mb)",breaks=c(0,50,100,150,200,250,300))+
  labs(title = "Frequency of Variants by Chromosome and Position for Trimmed Sequences", key = "Density")+
  theme(plot.title = element_text(size = 25,margin=margin(20,20,20,20)),
        legend.key.size =unit(1,"cm"),
        legend.text = element_text(size = 20),
        legend.title = element_text(size = 20,margin=margin(20,20,20,20)),
        axis.title.x = element_text(size = 20),
        axis.text.x = element_text(size = 20, angle = 45),
        axis.title.y = element_text(size = 20,margin=margin(20,20,20,20)),
        axis.text.y = element_text(size = 20),
        panel.background = element_rect(fill = "white"))
ggsave("alignment_intersections.jpg",height=8,width=10)
```



Let's see some of this live…

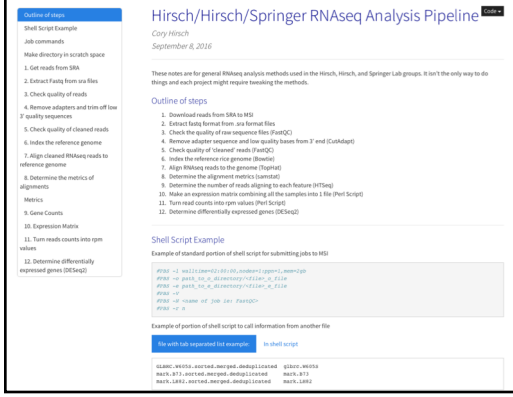
Helpful ggplot Reference Page

https://ggplot2.tidyverse.org/reference/



R Markdown in R Studio

- Tutorial at http://rmarkdown.rstudio.com/lesson-1.html



- Easy to keep organized
- Images contained within notes (if generated in R!)
- Flat file that is rendered into a .html file to view