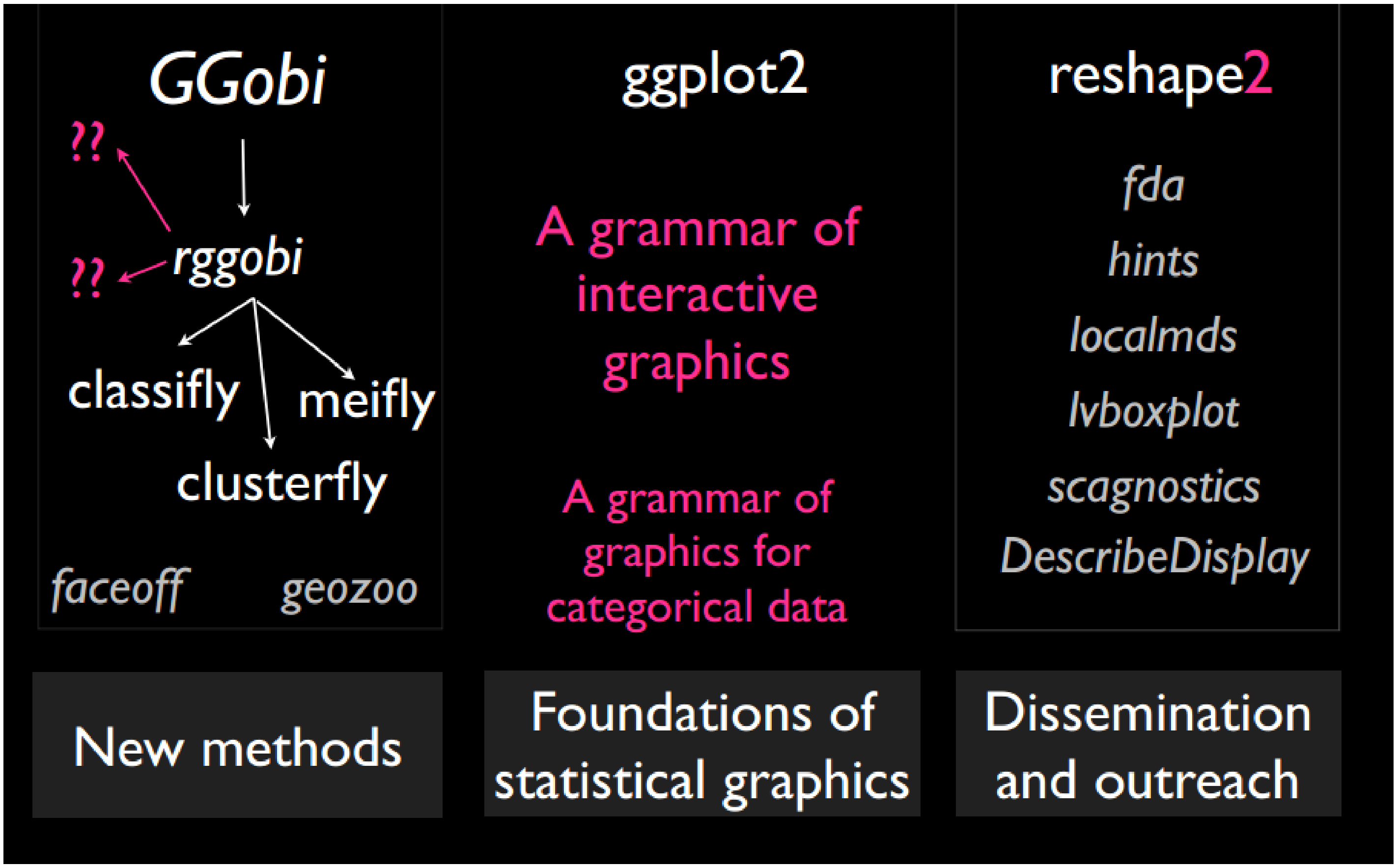
Lab#3: Tutorial of ggplot2

In this tutorial, we continue to use the USOpen.csv file as in our previous R basic tutorial. We will go through all the basic arguments and graphical types in the ggplot2 package with the qplot() function. The qplot function is also called quick plot function, the grammar of which is similar as the basic package of graphics in R.

“ggplot2” provides tool to create colorful and multi-layered visualizations by adding different layers. It was initially developed by Leland Wilkinson in 1990s. The philosophy of the creation of ggplot2 is to make graphics easier and to facilitate research into new types of display. The below figure shows the current graphical packages in R(Screenshot from ggplot2 official slide show on its website).



GGobi

??

??

rggobi

classifly

meifly

clusterfly

faceoff *geozoo*

ggplot2

A grammar of interactive graphics

A grammar of graphics for categorical data

reshape2

fda

hints

localmds

lvboxplot

scagnostics

DescribeDisplay

New methods

Foundations of statistical graphics

Dissemination and outreach

```
#Set the working directory and load in the data USOpen.csv.

>setwd("") #fill in where you store your data and change all slashes to back slashes

# for example, mine is setwd("C:/Users/Tony Tong/Desktop/R_file")

>tennis<-read.csv("USOpen.csv")

>install.packages("ggplot2")

> library("ggplot2")
```

1. Bar Plots

```
# Choose five countries, and see players from which of those countries won most games every year.

>t<-subset(tennis,country1=="USA"|country1=="ITA"|country1=="RUS"|country1=="GER"|country1=="BEL")

> qplot(year, data=t, fill=country1, geom="bar")

>ggplot(t, aes(year, fill=country1)) + geom_bar() # ggplot function can do similar things as qplot function

# Change the plot to horizontal view by adding “coord_flip()”.

> qplot(year, data=t, fill=country1, geom="bar")+ coord_flip()

#Further specify the position argument with “fill” or “dodge”. The default position is “stack”

> qplot(year, data=t, fill=country1, geom="bar", position="fill")+coord_flip()

#Adjust the bind width with “binwidth”.

> f1<-qplot(year, data=t, fill=country1, geom="bar", position="dodge",binwidth=0.5)

# Adjust the scale with “scale_x_continuous”.

>f2=f1+scale_x_continuous(breaks = seq(2003, 2011, 1))

# Change the color you want with “scale_fill_manual”. See color code:
http://www.computerhope.com/htmcolor.htm

> f3<-f2+scale_fill_manual(values=c("#7fc6bc","#083642","#b1df01","#cdef9c","#466b5d"))

#Find you favorite color combination with “scale_fill_brewer”

> qplot(year, data=t, fill=country1, geom="bar", position="dodge",binwidth=0.5)+
scale_fill_brewer(palette="Paired")

#Where can I find all available brewers in R?

>RColorBrewer::display.brewer.all()# Try  palette="Blues"
```

2. Scatter Plots

# Let’s re-do the visualization between winners and errors:

#Use size argument to specify the size of dots

```
>qplot(c(winner1,winner2),c(error1,error2),data=tennis,size=c(winner1,winner2)/c(error1,error2))
```

#Use color argument to add discrete colors to dots

```
>qplot(c(winner1,winner2),c(error1,error2),data=tennis,size=c(winner1,winner2)/c(error1,error2),color=factor(+rep(1:0,each=1000)))# The factor() is used to convert numeric objects to non-numeric.
```

#Use color statement and “scale_color_gradient” to add continuous colors to dots

```
>qplot(c(winner1,winner2),c(error1,error2),data=tennis,size=c(winner1,winner2)/c(error1,error2),color=c(ace1,+ace2))+scale_colour_gradient(low="black", high="red")
```

#Use shape statement and to distinguish shapes of different dots

```
>qplot(c(winner1,winner2),c(error1,error2),data=tennis,size=c(winner1,winner2)/c(error1,error2),color=c(ace1,+ace2), shape= factor(rep(0:1,each=1000)))+scale_colour_gradient(low="blue", high="red")
```

#Specify the names of axis and legends by adding related arguments step by step.

```
>plot1<-qplot(c(winner1,winner2),c(error1,error2),data=tennis,size=c(winner1,winner2)/c(error1,error2),color=c(ace1,+ace2), shape= factor(rep(0:1,each=1000)))+scale_colour_gradient(low="blue", high="red",name="Number of Aces")
```

```
>plot2<-plot1+ xlab("Number of Winners")+ ylab("Number of Errors")
```

```
>plot3<-plot2+scale_colour_gradient(low="blue", high="red", name="Number of Aces")+scale_shape_discrete(name="Result",labels=c("Winner","Loser"))+scale_size_continuous(name="Ratio of Winners to Errors")
```

#Add trend lines and shade by using “geom_smooth()” argument with “gam”,”loess” and “lm” methods.

```
> plot4<-qplot(c(winner1,winner2),c(error1,error2),data=tennis,color=factor(rep(0:1,each=1000)))
```

```
> plot4+geom_smooth() # default method=”gam” when n>1000, otherwise, method=”loess”.
```

```
> plot4+geom_smooth(method="loess", level=0.99) # Use se=F to delete the shade.
```

#Add labels to your lines

```
> plot4+geom_smooth(method="lm", level=0.99) +geom_text(aes(40, 60, label="Trend Line for losers",color="1"))+ geom_text(aes(50, 30, label="Trend Line for Winners",color="0"))
```

3. Density Plots and Histgrams

#Compare the distributions of the average speed of first serve and the average speed of second serve.

```
> qplot(c(firstPointWon1,firstPointWon2,secPointWon1,secPointWon2),geom="density",data=tennis,
fill=factor(rep(1:4,each=1000)), alpha=0.2) #Alpha is transparency #+facet_grid(year~.) multiple layers
```

#Adjust the color and the level of smooth.

```
> qplot(c(firstPointWon1,firstPointWon2,secPointWon1,secPointWon2),geom="density",adjust=3, data=tennis,
fill=factor(rep(1:4,each=1000)), alpha=0.2)+scale_fill_manual(values=c("red","pink","blue","lightblue"))
```

#Create a stacked histogram and get a sense of overall distribution of proportion of serve points won.

```
> qplot(c(firstPointWon1,firstPointWon2,secPointWon1,secPointWon2),geom="histogram",data=tennis,
fill=factor(rep(1:4,each=1000)), binwidth=0.02,alpha=0.2)
```

#Stack density of each years' number of aces.

```
>qplot(c(ace1,ace2),geom="density",adjust=3,data=tennis,fill=factor(rep(year,2)),alpha=0.5,position="stack")
```

#Sometimes, we only need density lines.

```
> qplot(c(ace1,ace2),geom="density",adjust=3,data=tennis,color=factor(rep(year,2)),alpha=0.5,position="stack")
```

4. Pie charts and box plots

# Generate nationality pie charts by rotating bar charts

```
>qplot(country1, data=t, geom="bar", fill=country1,width=1)+ coord_polar()# x as direction
```

```
>qplot(country1, data=t, geom="bar", fill=country1,width=1)+ coord_polar(theta="y")# y as direction
```

```
>qplot(factor(1), data=t, geom="bar", fill=country1)+coord_polar(theta="y")# Ordinary pie chart
```

#Use box plot to compare the different distributions of first and second point won.

```
> w1<- qplot(factor(rep(1:4,each=1000)),c(firstPointWon1,firstPointWon2,secPointWon1,secPointWon2),
data=tennis, geom="boxplot", fill=factor(rep(1:4,each=1000)))
```

# Mark distributions of dots and outliers with green color(A good choice for small data sets)

```
>w1+ geom_boxplot(outlier.colour = "green", outlier.size = 2)
```

```
>w2<- qplot(factor(rep(1:4,each=1000)),c(firstPointWon1,firstPointWon2,secPointWon1,secPointWon2),
data=tennis, geom=c("jitter","boxplot"), fill=factor(rep(1:4,each=1000)))
```

# Violin Plot

```
> w3<- qplot(factor(rep(1:4,each=1000)),c(firstPointWon1,firstPointWon2,secPointWon1,secPointWon2),
data=tennis, geom=c("jitter","violin"), fill=factor(rep(1:4,each=1000)))
```

5. Lab #3 Practice:

Use a scatter plot to show whether there is a correlation between one's average speed of first serve and her opponent's return points won. Please specify the size, color and shape of dots according to variables that you are interested in.

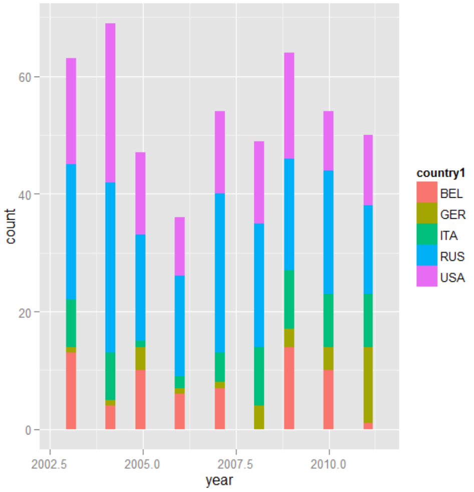
How about one's average speed of second serve v.s. her opponent's return points won?

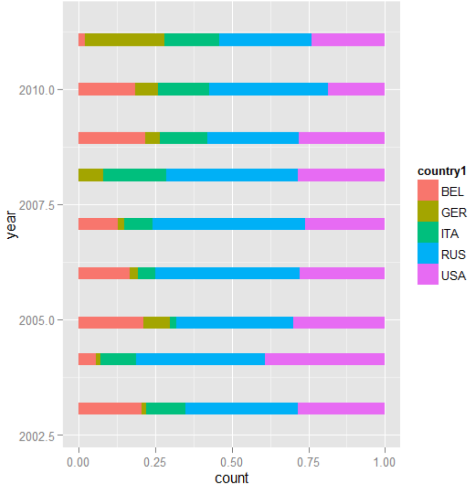
Add any comment or use other types of plots if you think you can achieve the goal.

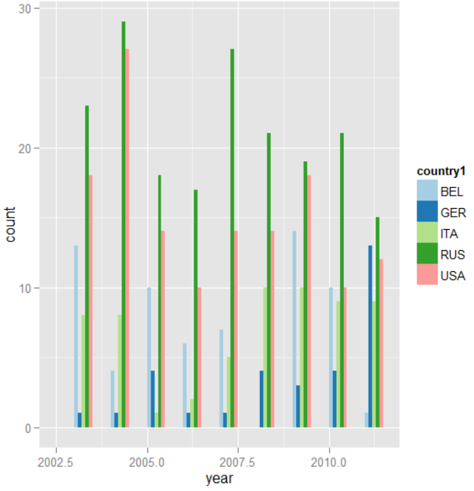
6. Useful resources for ggplot2:

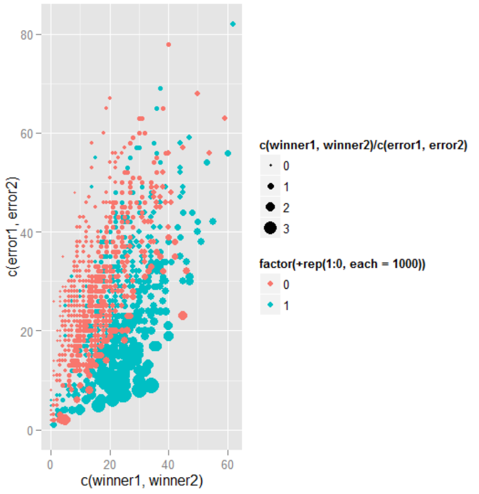
a. Find examples for the usage of all basic arguments: http://docs.ggplot2.org/current/
b. Subscribe ggplot2 and join the official discussion group: http://ggplot2.org/
c. Learning to code in ggplot() style: http://ggplot2.org/resources/2007-vanderbilt.pdf
d. Advanced Learners: http://cran.r-project.org/web/packages/ggplot2/ggplot2.pdf

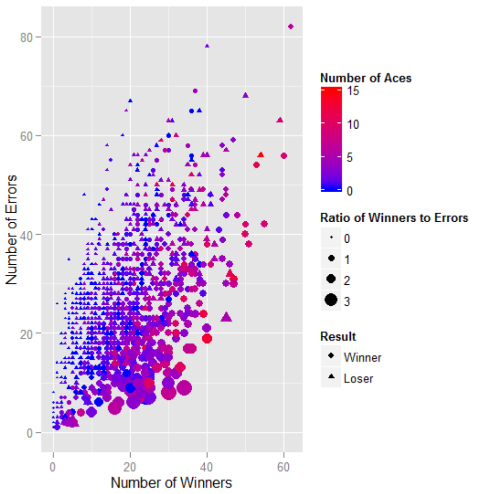
7. Check if you get all those graphs correct after class.

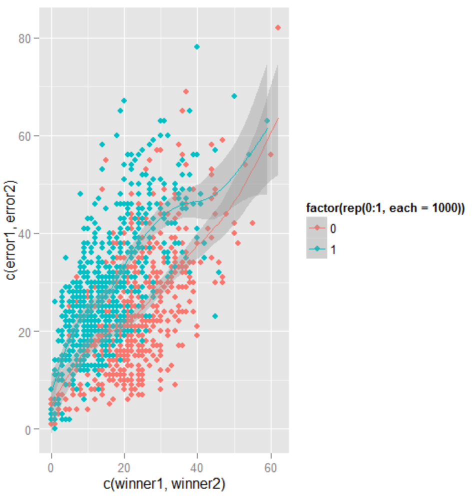




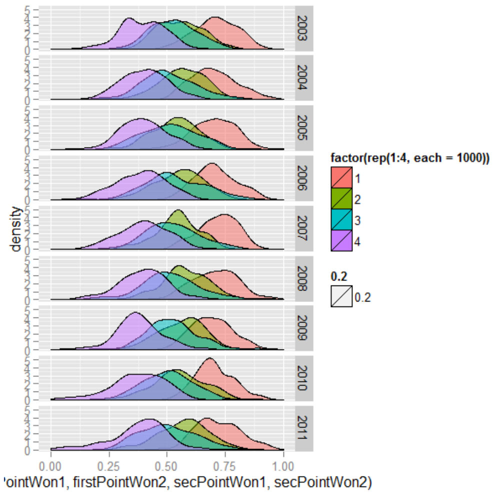


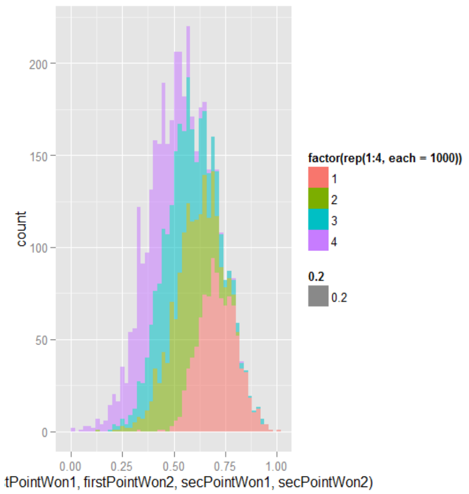


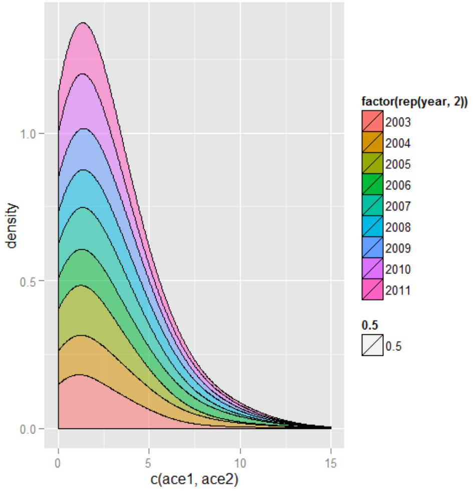


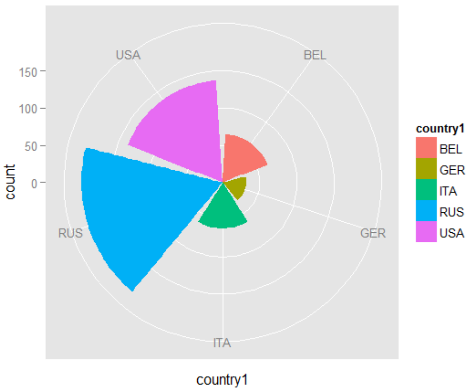


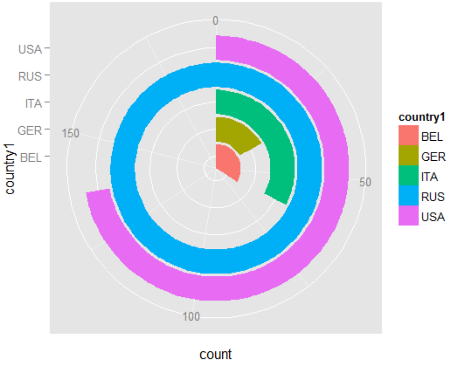
Tony Guangyu Tong gt2298@columbia.edu QMSS 4063 Data Visualization(13F)

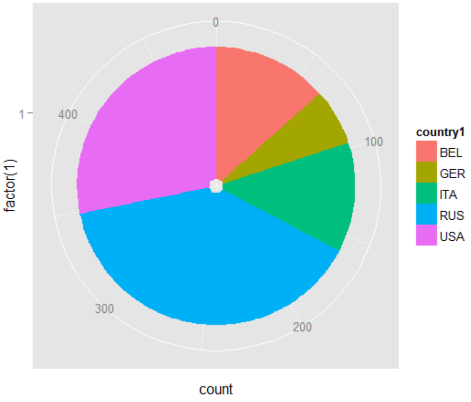

density
'ointWon1, firstPointWon2, secPointWon1, secPointWon2)
factor(rep(1:4, each = 1000))

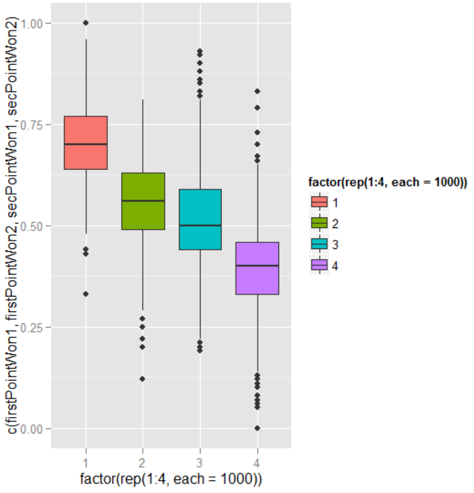

count
tPointWon1, firstPointWon2, secPointWon1, secPointWon2)
factor(rep(1:4, each = 1000))

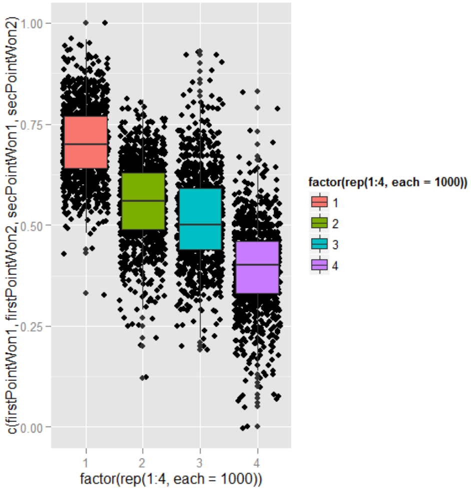

density
c(ace1, ace2)
factor(rep(year, 2))

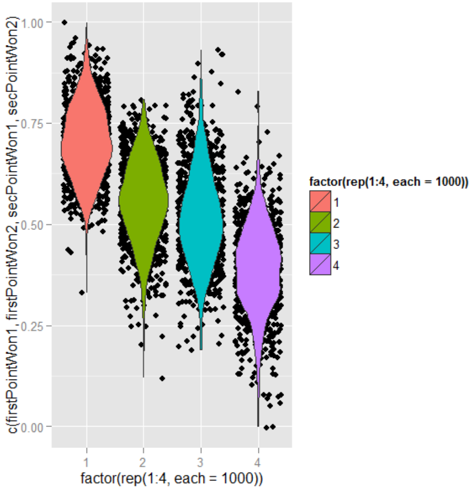

count
USA
BEL
RUS
GER
ITA
country1
country1
BEL
GER
ITA
RUS
USA


country1
USA
RUS
ITA
GER
BEL
count
country1
BEL
GER
ITA
RUS
USA


factor(1)
count
country1
BEL
GER
ITA
RUS
USA


c(firstPointWon1, firstPointWon2, secPointWon1, secPointWon2)
factor(rep(1:4, each = 1000))
factor(rep(1:4, each = 1000))


c(firstPointWon1, firstPointWon2, secPointWon1, secPointWon2)
factor(rep(1:4, each = 1000))
factor(rep(1:4, each = 1000))


c(firstPointWon1, firstPointWon2, secPointWon1, secPointWon2)
factor(rep(1:4, each = 1000))
factor(rep(1:4, each = 1000))