Data Visualization and Graphics in R

Xin Luo

December 6th, 2016

Outline

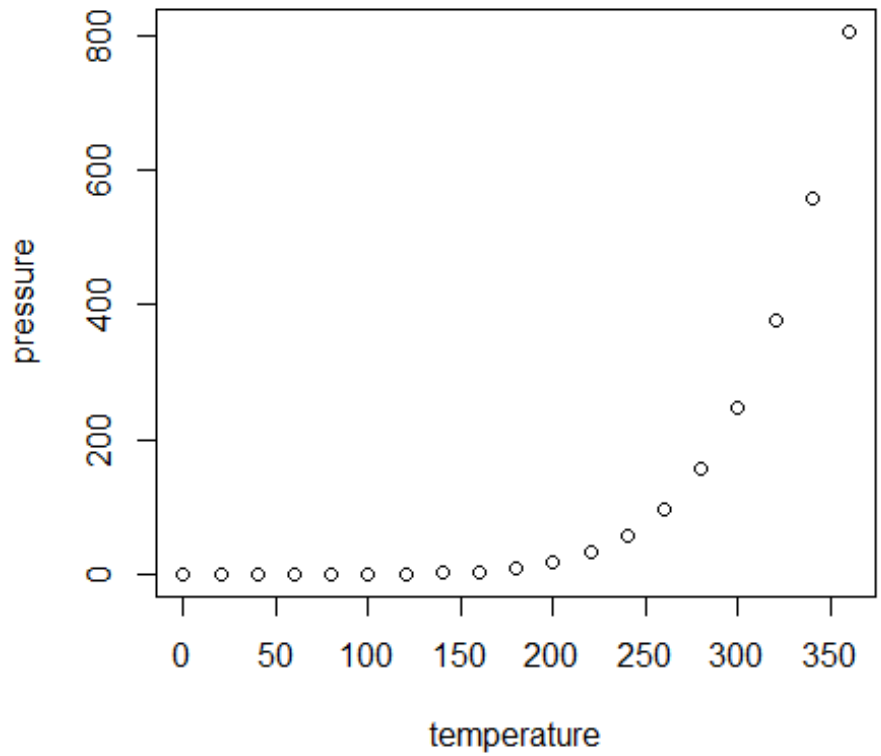
- Plotting using build in graphics tools in R
- Plotting with graphic packages in R (ggplot2)
- Visualizing data by different types of graphs in R (scatter plot, line graph, bar graph, histogram, boxplot, pie chart, heat map, Venn diagram, correlation plot)
- Generate and output polished graphs for publication and presentation

Why using R for plotting

1. When the large sample size exceed the capacity for excel, prism or other graphic tools
2. Fast and simple
3. Super easy with any modification
4. Reproduce the figures and keep exact same format for new figures on new data
5. Bundle with complicated statistical analysis
6. Many add-on packages for various needs beyond build in graphics tools

Basic Plot Example

```
data ()
str (pressure)
plot (pressure)
```



Labels and Axes

Default: R uses the variable names for axes labels and computes range for axes.

Manual change by:

- axes labels: xlab, ylab
- size of labels: cex.lab
- axes range: xlim, ylim

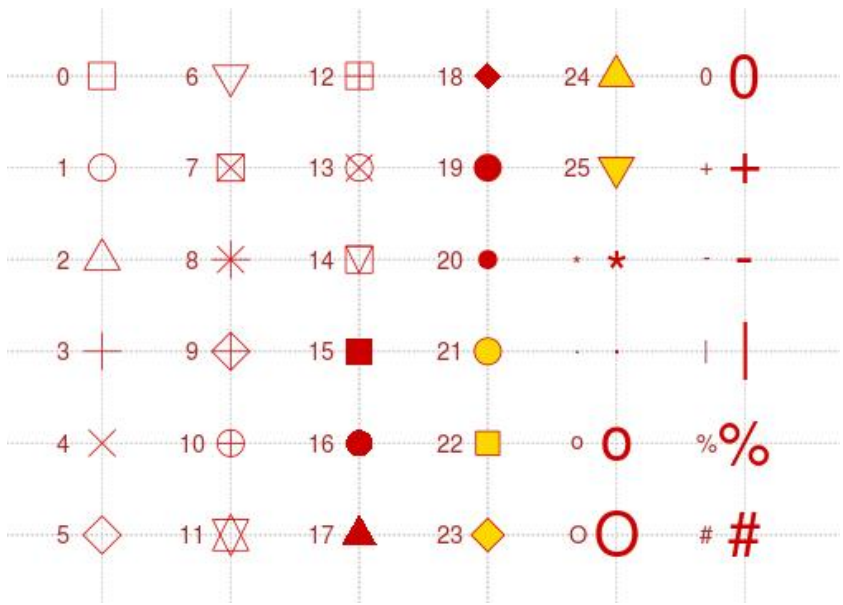
Titles

- main: sets plot title (above plot)
- sub: sets subtitle (beneath plot)

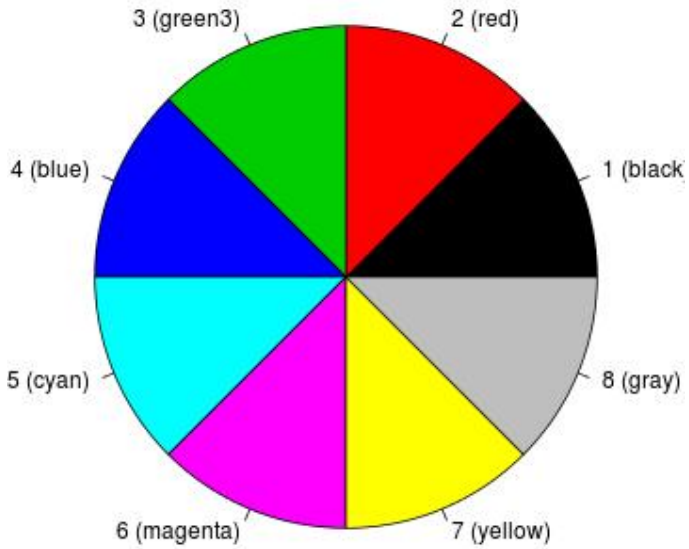
Symbols, colors and lines

- type: "p" for points, "l" for lines, "b" for both, "h" for histogram-like
- pch: point symbol
- col: color
- cex: size factor
- lty: line type
- lwd: line width

Plot symbols



The default color palette in R:



RcolorBrewer Package Palette

R Color Palette other options

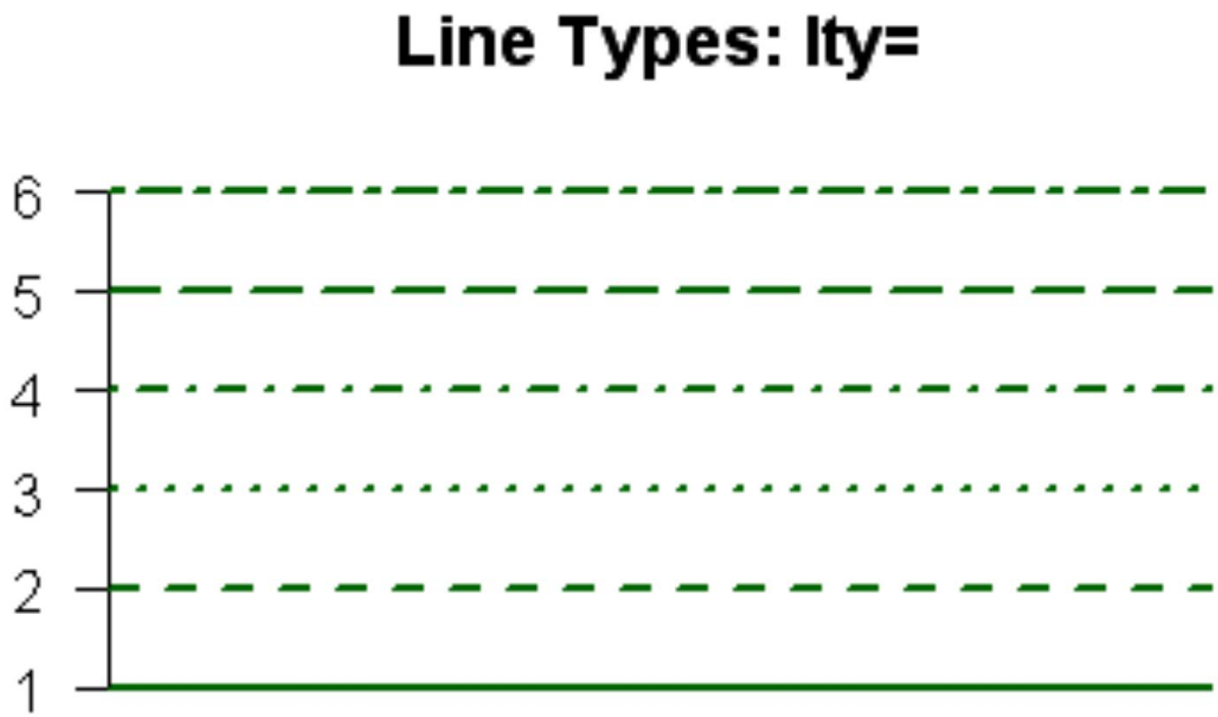
- rainbow(n)
- heat.colors(n)
- terrain.colors(n)
- topo.colors(n)
- cm.colors(n)

R color cheat sheet

1	white	#FFFFFF	255	255	255
2	aliceblue	#F0F8FF	240	248	255
3	antiquewhite	#FAEBD7	250	235	215
4	antiquewhite1	#FFEFDB	255	239	219
5	antiquewhite2	#EEDFCC	238	223	204
6	antiquewhite3	#CDC0B0	205	192	176
7	antiquewhite4	#8B8378	139	131	120
8	aquamarine	#7FFFD4	127	255	212
9	aquamarine1	#7FFFD4	127	255	212
10	aquamarine2	#76EEC6	118	238	198
11	aquamarine3	#66CDAA	102	205	170
12	aquamarine4	#458B74	69	139	116
13	azure	#F0FFFF	240	255	255
14	azure1	#F0FFFF	240	255	255
15	azure2	#E0EEEE	224	238	238
16	azure3	#C1CDCD	193	205	205
17	azure4	#838B8B	131	139	139
18	beige	#F5F5DC	245	245	220
19	bisque	#FFE4C4	255	228	196
20	bisque1	#FFE4C4	255	228	196
21	bisque2	#EED5B7	238	213	183
22	bisque3	#CDB79E	205	183	158
23	bisque4	#8B7D6B	139	125	107
24	black	#000000	0	0	0
25	blanchedalmond	#FFEBCD	255	235	205
26	blue	#0000FF	0	0	255
27	blue1	#0000FF	0	0	255
28	blue2	#0000EE	0	0	238
29	blue3	#0000CD	0	0	205
30	blue4	#00008B	0	0	139
31	blueviolet	#8A2BE2	138	43	226
32	brown	#A52A2A	165	42	42
33	brown1	#FF4040	255	64	64
34	brown2	#EE3B3B	238	59	59
35	brown3	#CD3333	205	51	51
36	brown4	#8B2323	139	35	35
37	burlywood	#DEB887	222	184	135
38	burlywood1	#FFD39B	255	211	155
39	burlywood2	#EEC591	238	197	145
40	burlywood3	#CDAA7D	205	170	125
41	burlywood4	#8B7355	139	115	85
42	cadetblue	#5F9EA0	95	158	160
43	cadetblue1	#98F5FF	152	245	255
44	cadetblue2	#8EE5EE	142	229	238
45	cadetblue3	#7AC5CD	122	197	205
46	cadetblue4	#53868B	83	134	139
47	chartreuse	#7FFF00	127	255	0
48	chartreuse1	#7FFF00	127	255	0
49	chartreuse2	#76EE00	118	238	0
50	chartreuse3	#66CD00	102	205	0
51	chartreuse4	#458B00	69	139	0
52	chocolate	#D2691E	210	105	30
53	chocolate1	#FF7F24	255	127	36
54	chocolate2	#EE7621	238	118	33
55	chocolate3	#CD661D	205	102	29
56	chocolate4	#8B4513	139	69	19
57	coral	#FF7F50	255	127	80
58	coral1	#FF7256	255	114	86
59	coral2	#EE6A50	238	106	80
60	coral3	#CD5B45	205	91	69
61	coral4	#8B3E2F	139	62	47
62	cornflowerblue	#6495ED	100	149	237
63	cornsilk	#FFF8DC	255	248	220
64	cornsilk1	#FFF8DC	255	248	220
65	cornsilk2	#EEE8CD	238	232	205
66	cornsilk3	#CDC8B1	205	200	177
67	cornsilk4	#8B8878	139	136	120
68	cyan	#00FFFF	0	255	255
69	cyan1	#00FFFF	0	255	255
70	cyan2	#00EEEE	0	238	238
71	cyan3	#00CDCD	0	205	205
72	cyan4	#008B8B	0	139	139
73	darkblue	#00008B	0	0	139
74	darkcyan	#008B8B	0	139	139
75	darkgoldenrod	#B8860B	184	134	11
76	darkgoldenrod1	#FFB90F	255	185	15
77	darkgoldenrod2	#EEAD0E	238	173	14
78	darkgoldenrod3	#CD950C	205	149	12
79	darkgoldenrod4	#8B6508	139	101	8
80	darkgray	#A9A9A9	169	169	169
81	darkgreen	#006400	0	100	0
82	darkgrey	#A9A9A9	169	169	169
83	darkkhaki	#BDB76B	189	183	107
84	darkmagenta	#8B008B	139	0	139
85	darkolivegreen	#556B2F	85	107	47
86	darkolivegreen1	#CAFF70	202	255	112
87	darkolivegreen2	#BCEE68	188	238	104
88	darkolivegreen3	#A2CD5A	162	205	90
89	darkolivegreen4	#6E8B3D	110	139	61
90	darkorange	#FF8C00	255	140	0
91	darkorange1	#FF7F00	255	127	0
92	darkorange2	#EE7600	238	118	0
93	darkorange3	#CD6600	205	102	0
94	darkorange4	#8B4500	139	69	0
95	darkorchid	#9932CC	153	50	204
96	darkorchid1	#BF3EFF	191	62	255
97	darkorchid2	#B23AEE	178	58	238
98	darkorchid3	#9A32CD	154	50	205
99	darkorchid4	#68228B	104	34	139
100	darkred	#8B0000	139	0	0

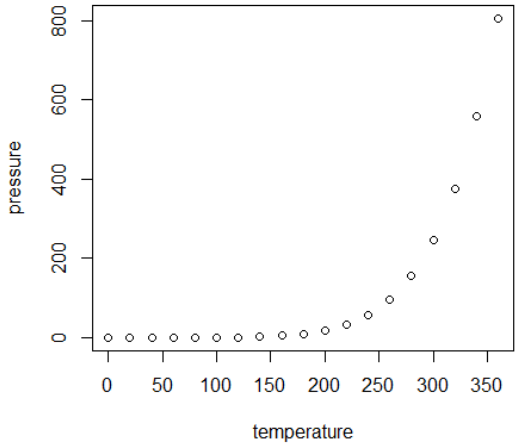
```
plot(pressure, col="#0000FF")
plot(pressure, col="blue")
```

Line Types

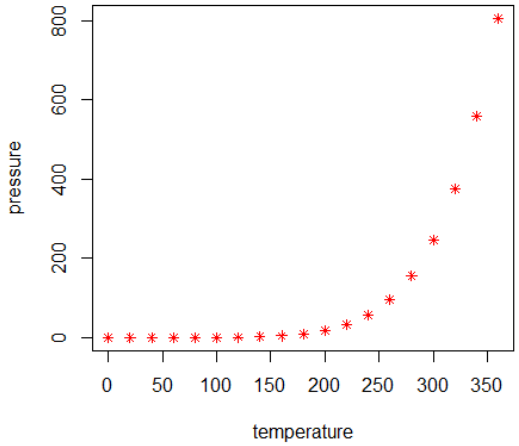


Plot example 1 plot points with formatting

```
plot (pressure, type="p")
```

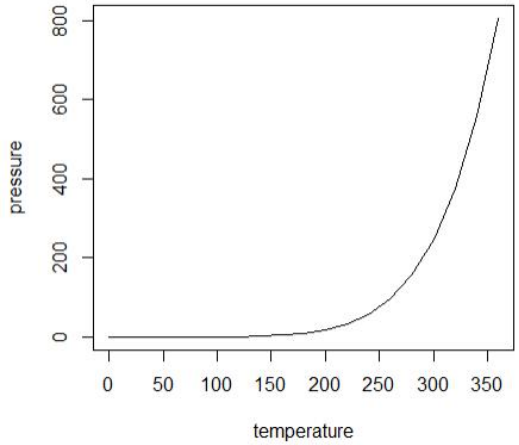



```
plot (pressure, type="p", pch = 8, cex =0.8, col="red")
```

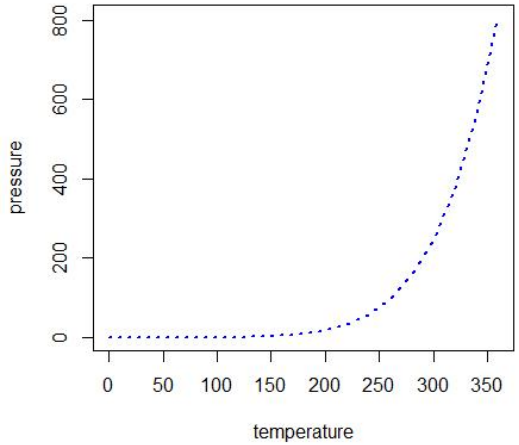


Plot example 2 line graph with formatting

```
plot (pressure, type="l")
```

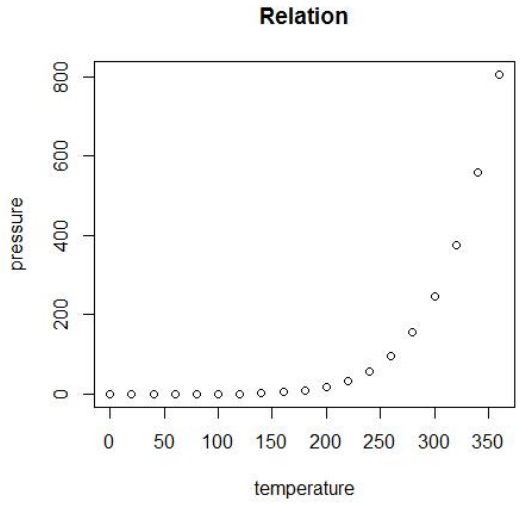



```
plot (pressure, type="l", lty = 3, lwd =2, col="blue")
```

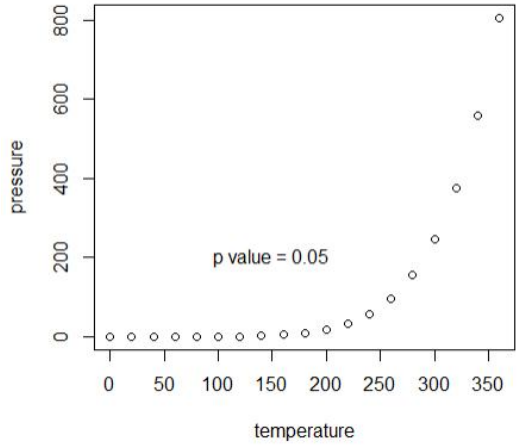


Plot example 3 add title and text

```
plot (pressure, main="Relation" )
```

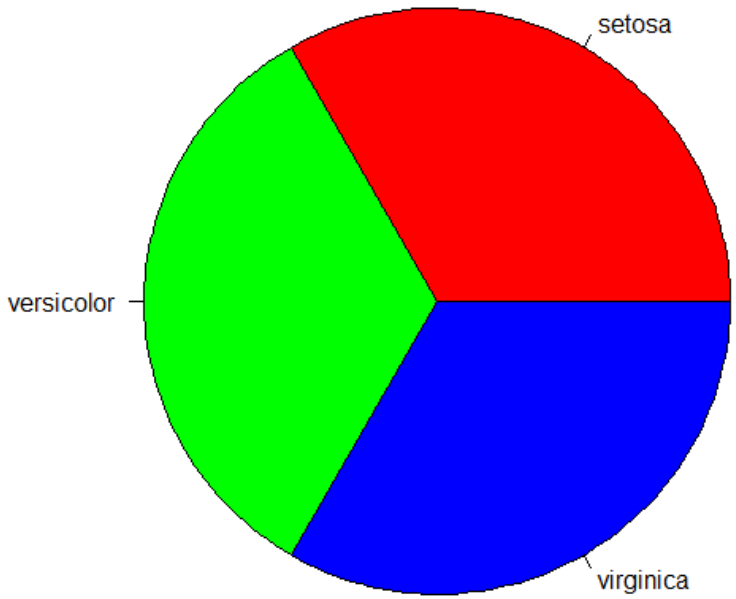



```
plot ( pressure )
text (150 ,200 , label =" p value = 0.05 ")
```



Pie chart

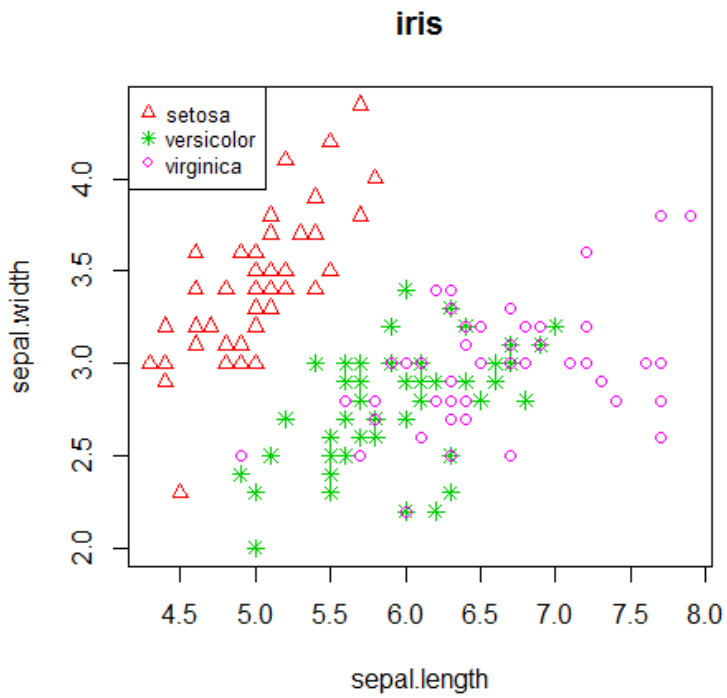
Pie chart is used to visualize the composition of the data groups




```
pie(table(iris$Species), col=rainbow(3))
```

Plot for multiple group

```
data(iris) # load iris data
pch.vec <- c(2 ,8 ,21)[iris$Species]
col.vec <- c(2 ,3 ,6)[iris$Species]
plot(iris$Sepal.Length, iris$Sepal.Width, col = col.vec, pch=pch.vec, xlab="sepal.length", ylab="sepal.width",main="iris")
legend ("topleft", pch=c(2 ,8 ,21) ,col=c(2 ,3 ,6) ,legend = unique(iris$Species), cex=0.8)
```



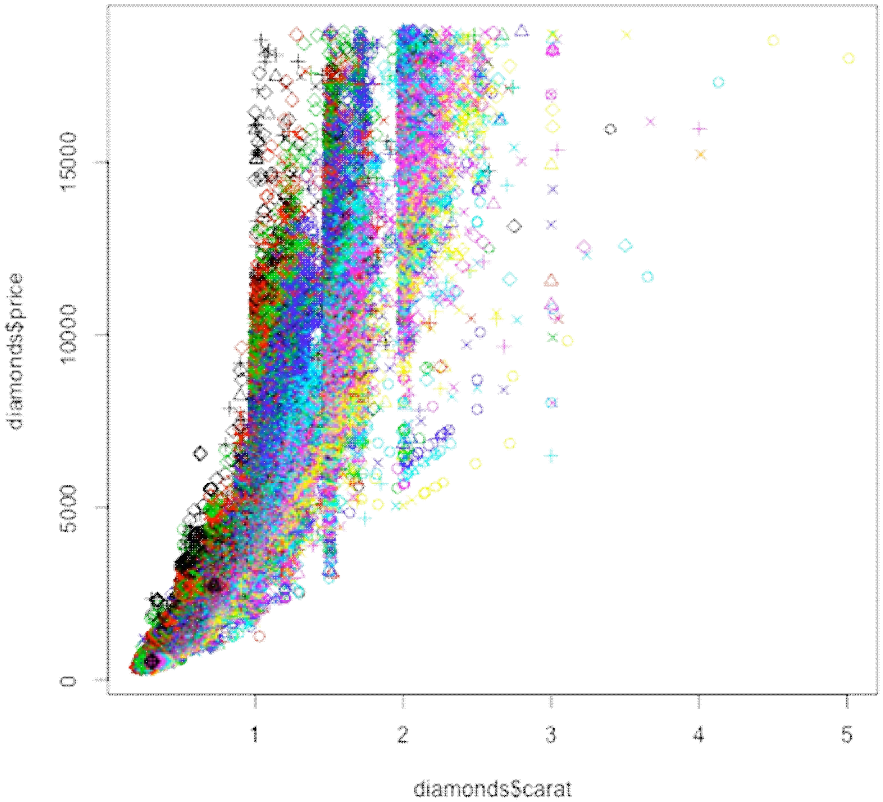
Beyond simple graphs: ggplot2

- Hadley Wickham's ggplot2 package provides a unified interface and simple set of options.
- Once you learn how ggplot2 works for one type of plot, you can easily apply the knowledge for any other types of plots
- It provides beautiful, publication ready results.
- Easy to plot for data with multiple groups and build legend automatically

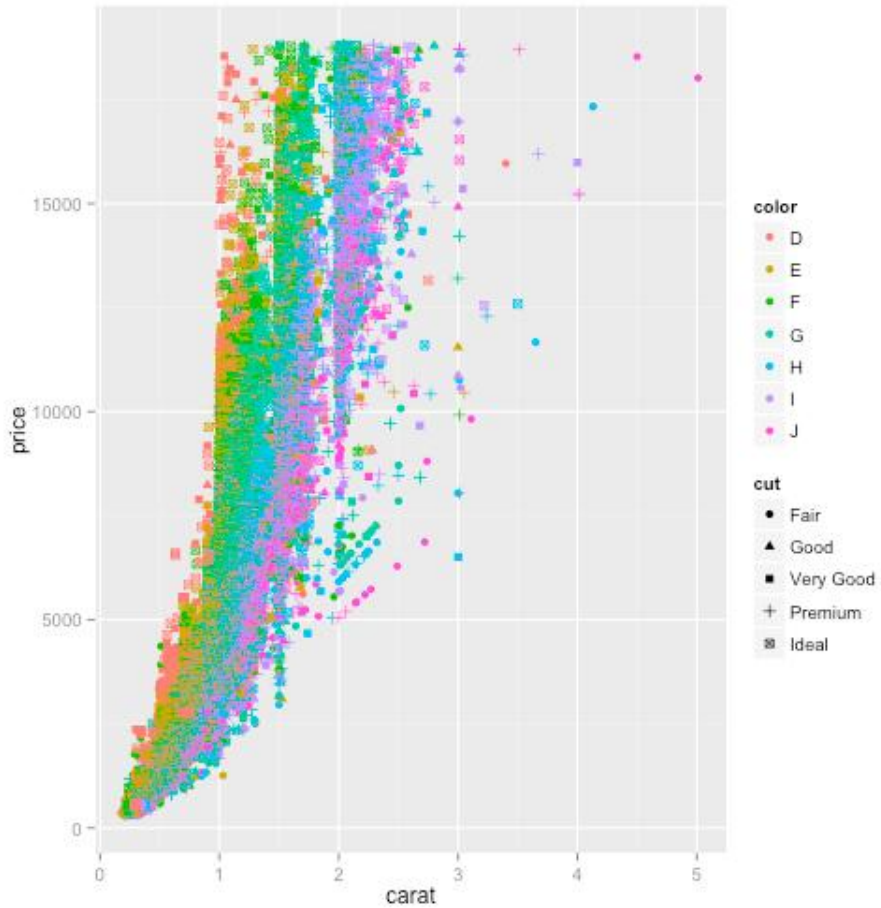
"*R Graphics Cookbook* by Winston Chang (O'Reilly). Copyright 2013 Winston Chang, 978-1-449-31695-2."

http://www.cookbook-r.com/Graphs/

Build-in R Plotting VS ggplot2



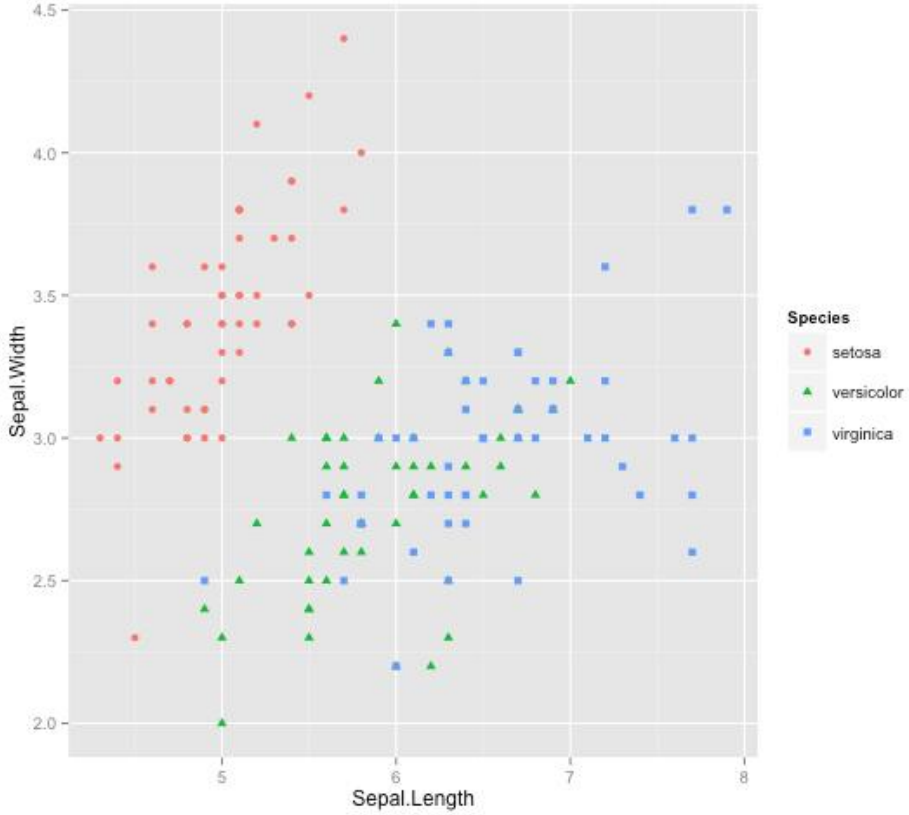

```
plot(diamonds$carat, diamonds$price, col = diamonds$color,
   pch = as.numeric(diamonds$cut))
```




```
ggplot(diamonds, aes(carat, price, col = color, shape = cut)) +
geom_point()
```

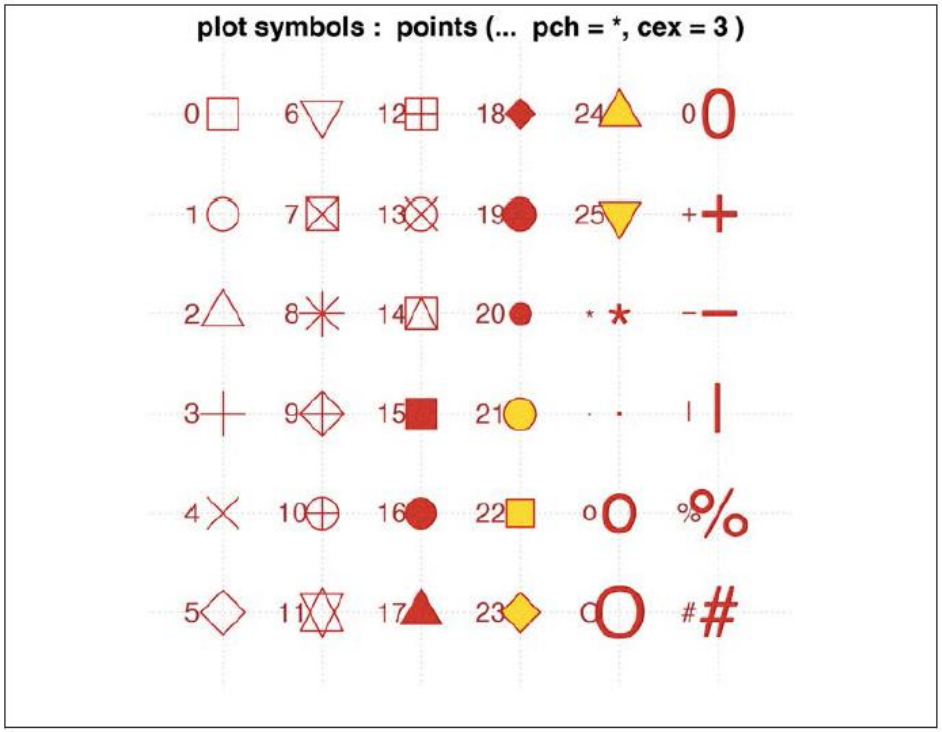
Scatter plot

display the relationship between two continuous variables




```
ggplot(iris, aes(x=Sepal.Length, y=Sepal.Width, shape=Species, colour=Species)) +
  geom_point()
```

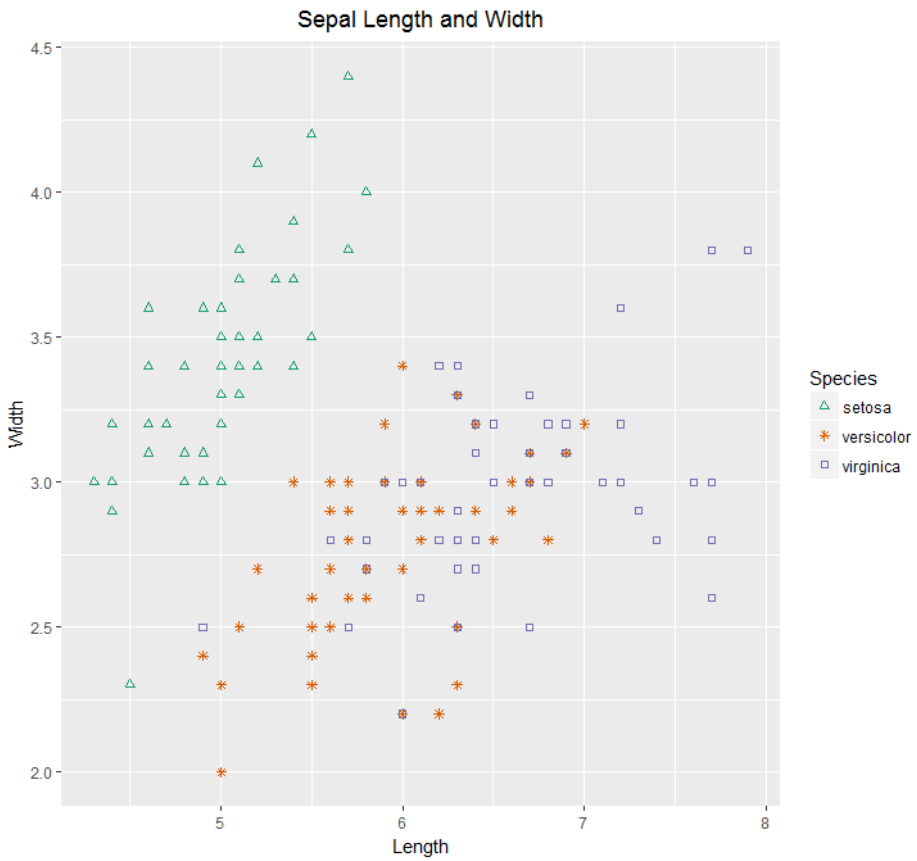
scale_shape_manual



scale_colour_brewer

Scatter plot

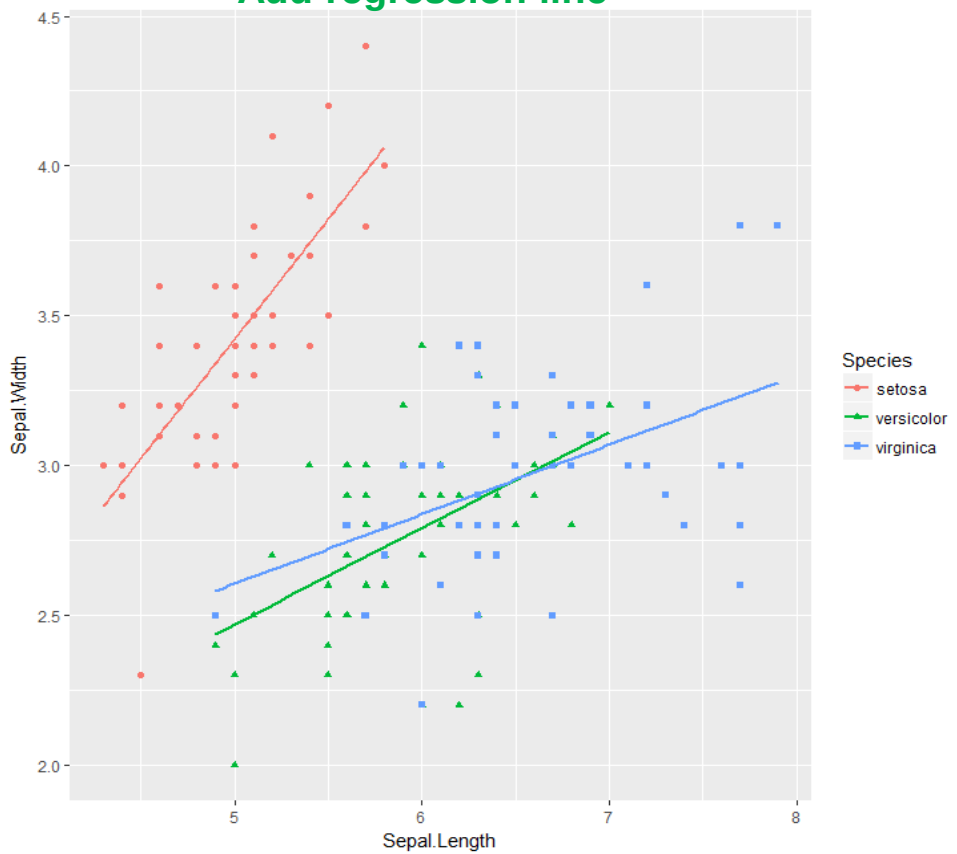
Change the points shape and color




```
ggplot(iris, aes(x=Sepal.Length, y=Sepal.Width, shape=Species, colour=Species)) +  geom_point() +
  scale_colour_brewer(palette="Dark2")+
  scale_shape_manual(values=c(2,8,0))+
  labs(x="Length",y="Width",title="Sepal Length and Width")+
  theme(plot.title = element_text(hjust = 0.5))
```

Scatter plot

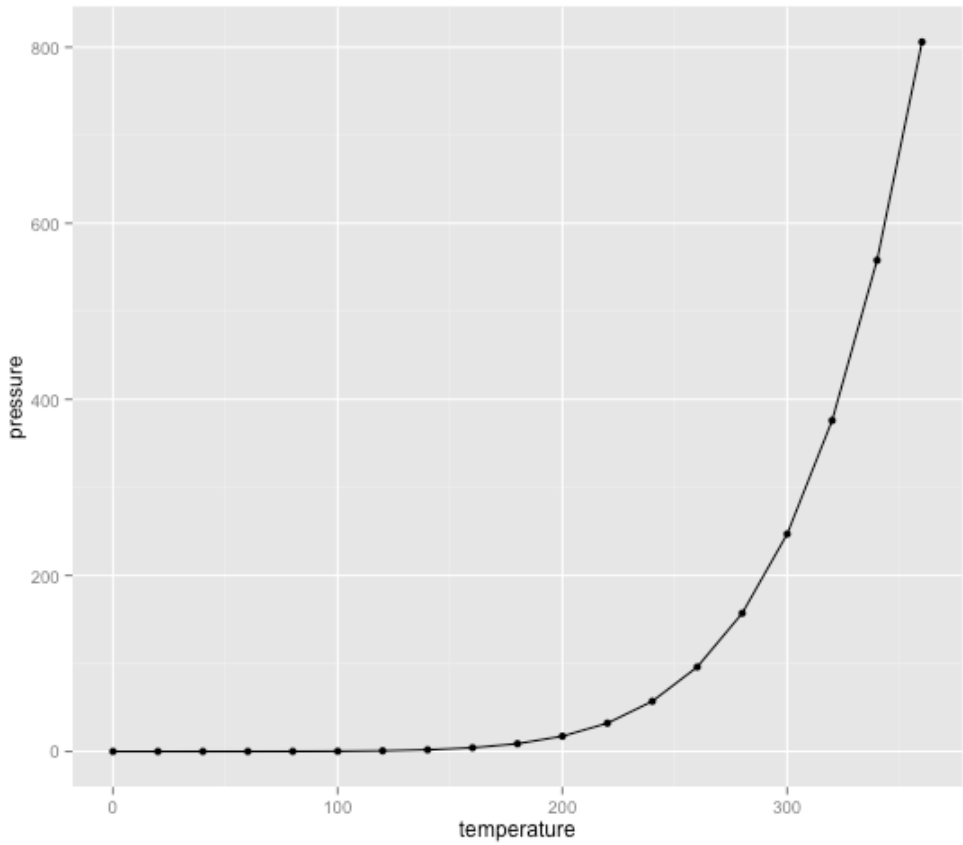
Add regression line




```
ggplot(iris, aes(x=Sepal.Length, y=Sepal.Width, shape=Species, colour=Species)) +
  geom_point() +
  geom_smooth(method=lm, se=FALSE)
```

Line Graph

the trend over time or other continuous variables

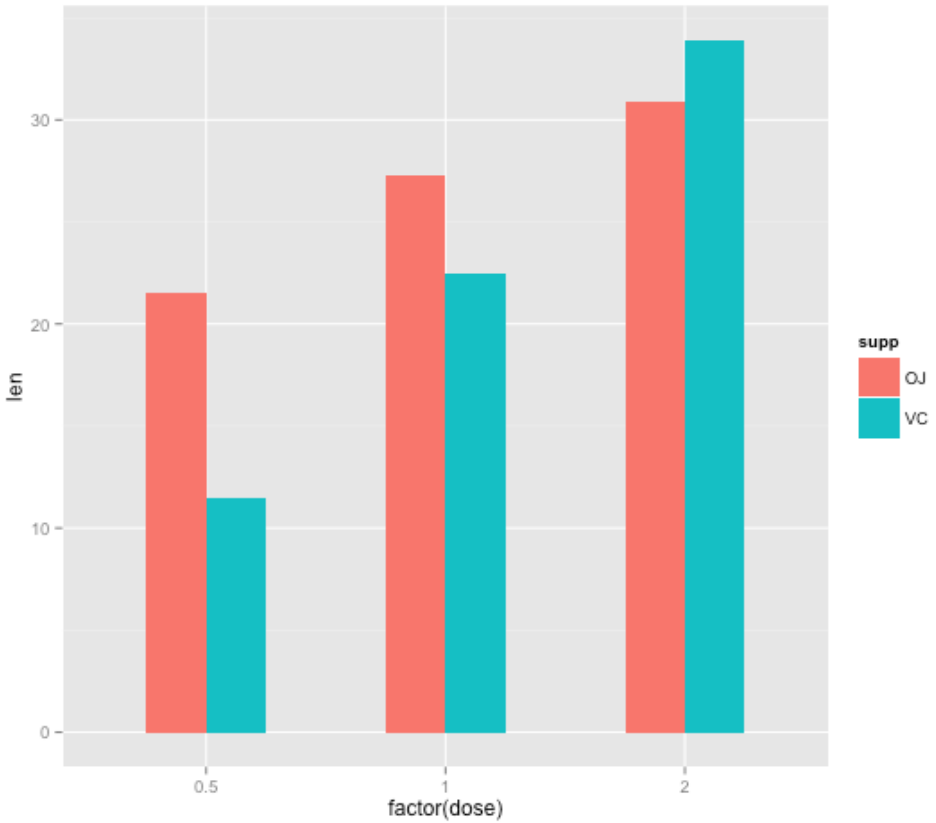



```
ggplot(pressure, aes(x=temperature, y=pressure)) + geom_line() + geom_point()
```

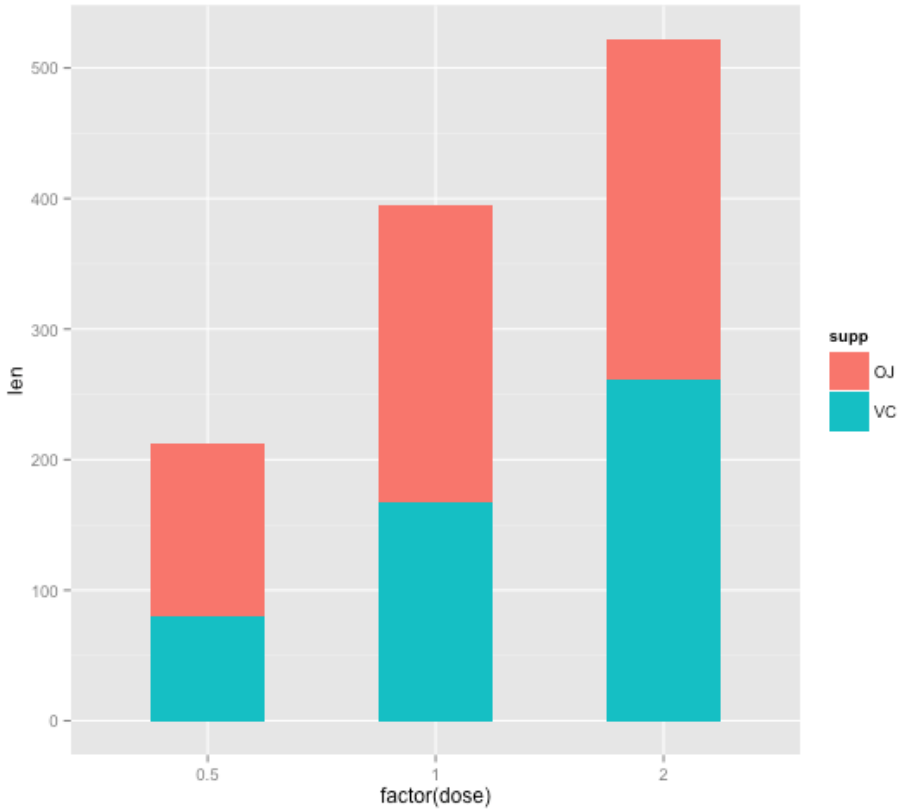
Bar Graph

display numeric values (y-axis) for different categories (x-axis)

1. Bar graph for exact value for y



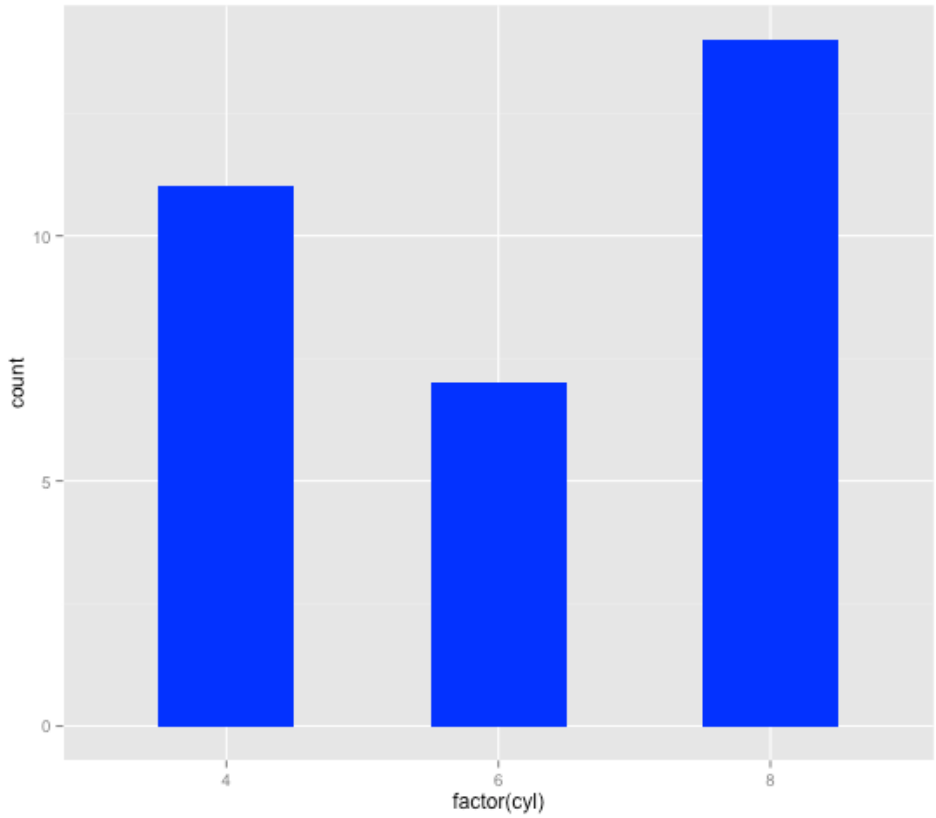

```
ggplot(ToothGrowth,aes(x=factor(dose),y=len, fill=supp))+
  geom_bar(stat="identity", position="dodge", width=0.5)
```




```
ggplot(ToothGrowth,aes(x=factor(dose),y=len, fill=supp))+
  geom_bar(stat="identity", width=0.5)
```

Bar Graph

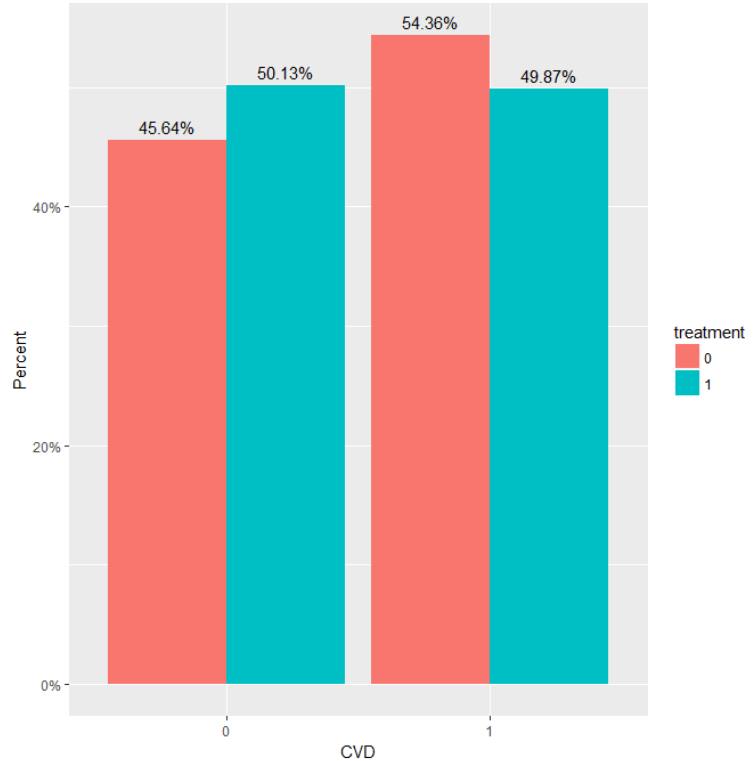
2. bar graph for counts of a categorical variable




```
ggplot(mtcars, aes(x=factor(cyl))) +
geom_bar(fill="blue",width=0.5)
```

Bar Graph

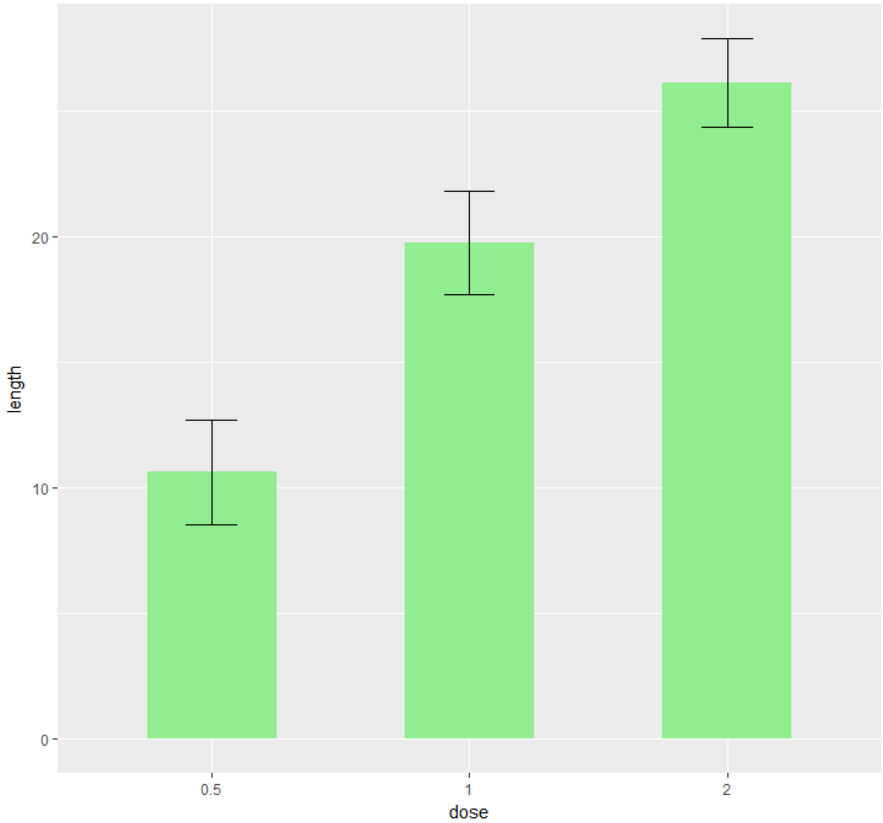
3. bar graph for percentage of a categorical variable




```
ggplot(dig, aes(x= factor(CVD),  group=factor(TRTMT))) +
  geom_bar(aes(y = ..prop.., fill = factor(TRTMT)), stat="count", position="dodge")+
  geom_text(aes(label = scales::percent(..prop..), y= ..prop.. ),
            stat= "count",position=position_dodge(0.9), vjust = -.5) +
  labs(x="CVD",y = "Percent", fill="treatment") +
  scale_y_continuous(labels=scales::percent)
```

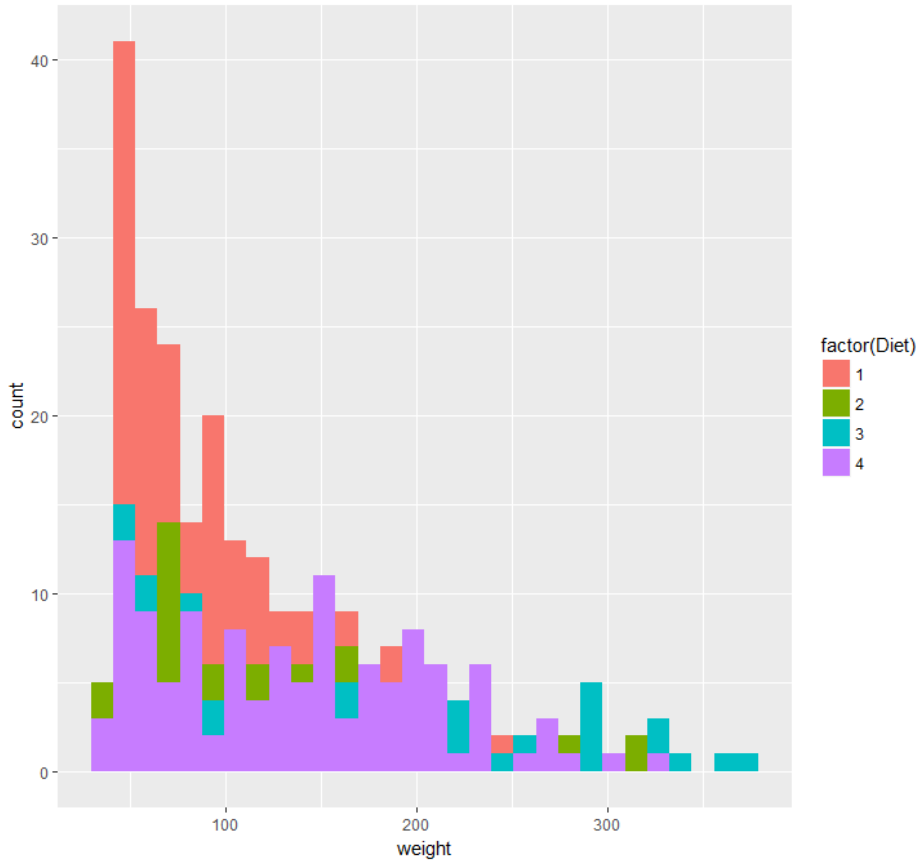
Bar Graph

4. Plot mean and error bars

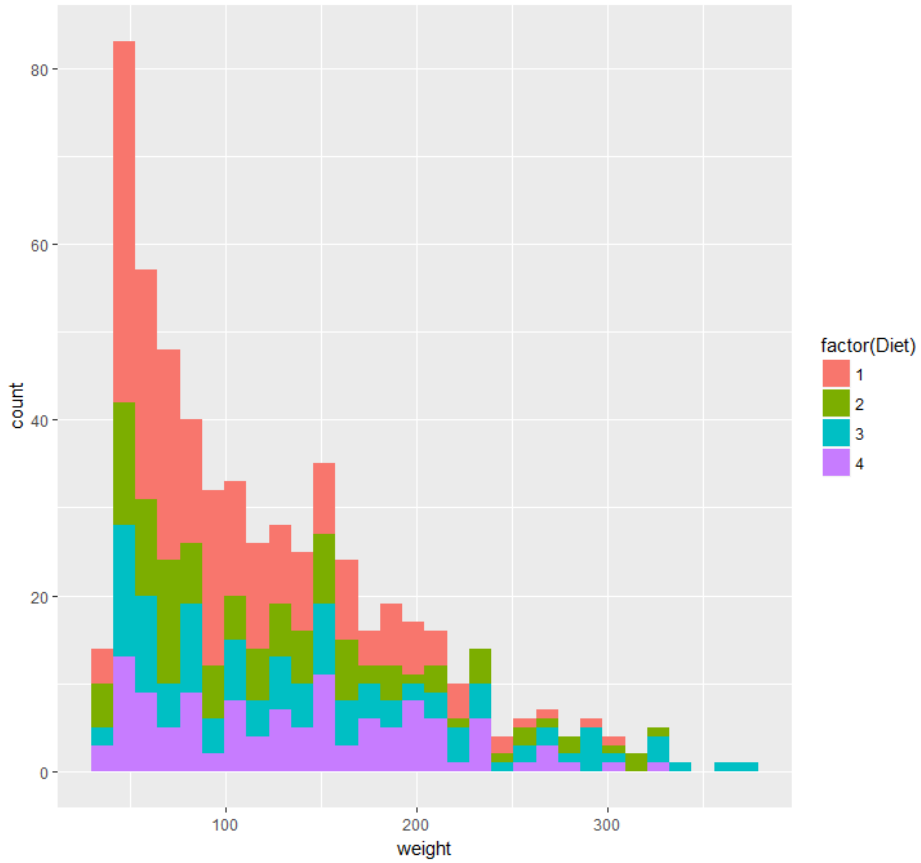



```
ggplot(ToothGrowth, aes(factor(dose), len )) +
  stat_summary(fun.y = mean, geom = "bar", width=0.5, fill="lightgreen") +
  stat_summary(fun.data = mean_cl_normal, geom = "errorbar", width=0.2)+
  labs(x="dose", y="length")
```

Visualize the distribution of data: I. histogram

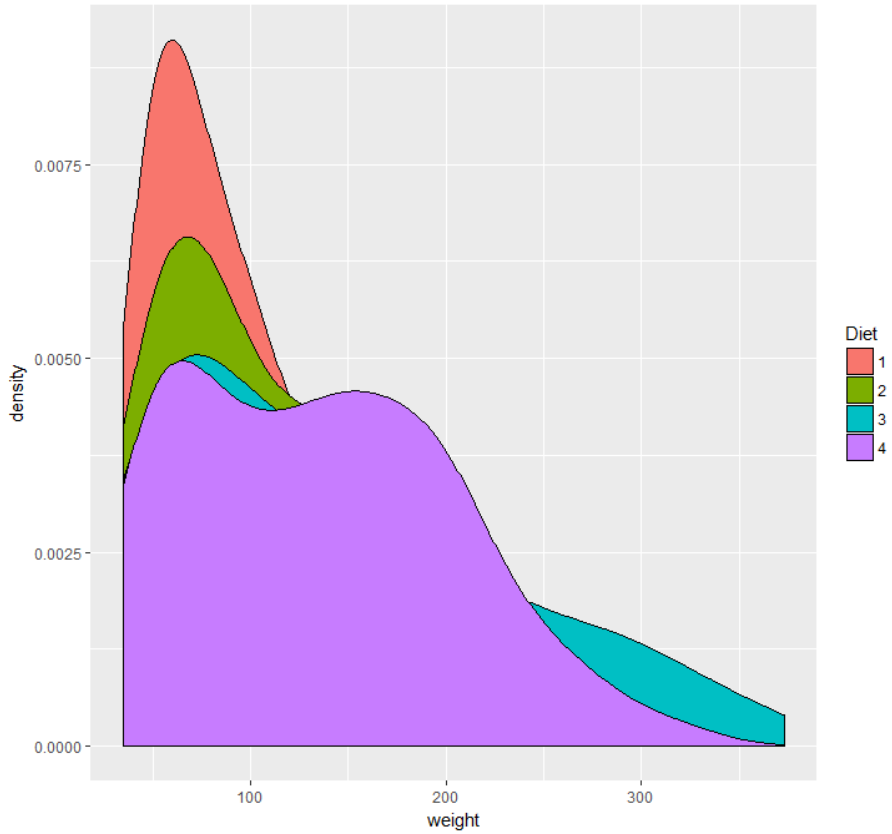



```
ggplot(ChickWeight, aes(x=weight, fill=factor(Diet))) +
  geom_histogram(position="identity")
```




```
ggplot(ChickWeight, aes(x=weight, fill=factor(Diet))) +
  geom_histogram()
```

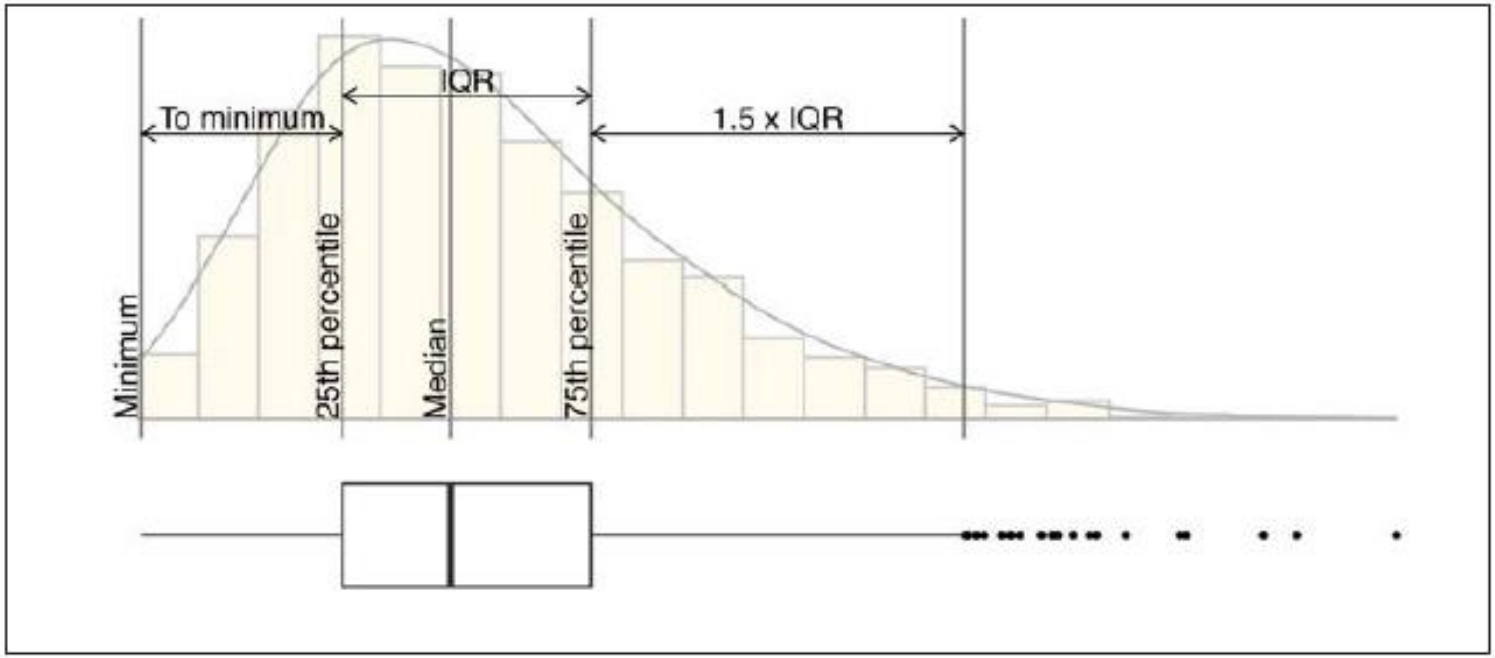
Visualize the distribution of data: II. Density Curve




```
ggplot(ChickWeight, aes(x=weight, fill=Diet)) +
geom_density()
```

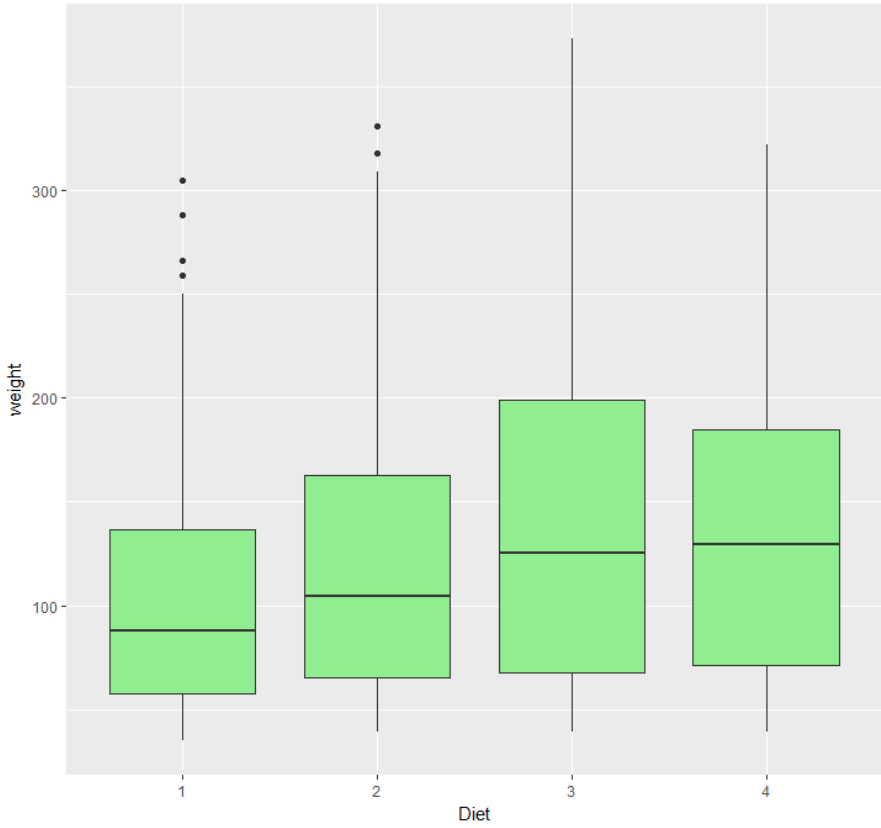
Visualize the distribution of data: III. Boxplot

Transformation among histogram, density curve and boxplot



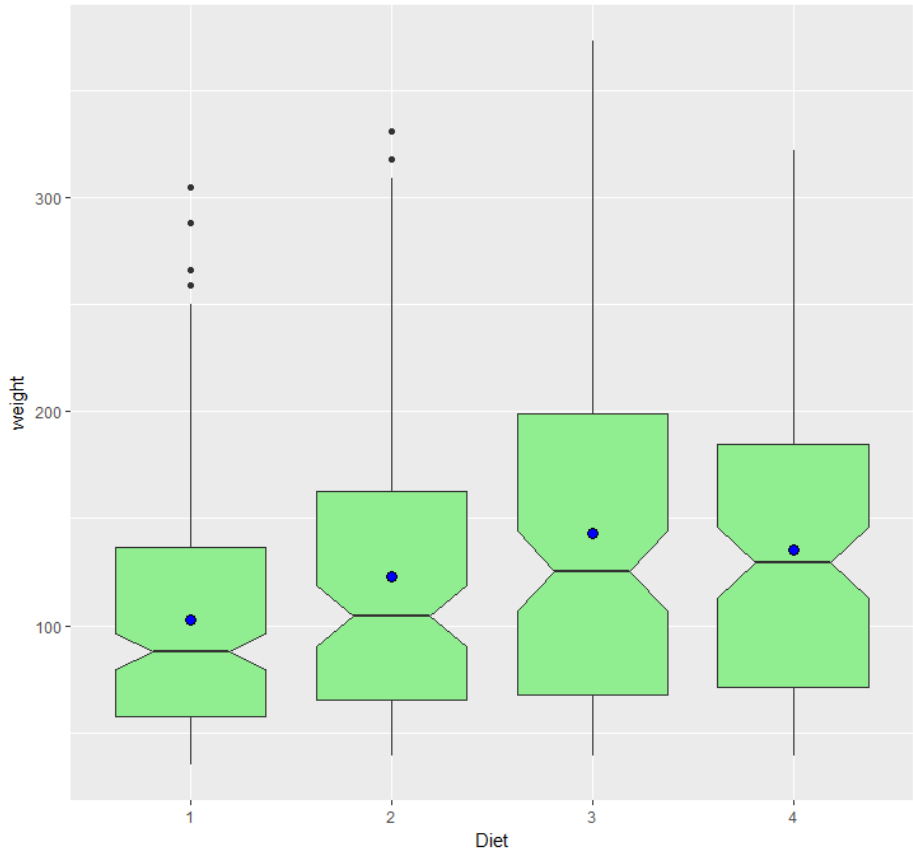
"*R Graphics Cookbook* by Winston Chang (O'Reilly). Copyright 2013 Winston Chang, 978-1-449-31695-2."

Visualize the distribution of data: III. Boxplot



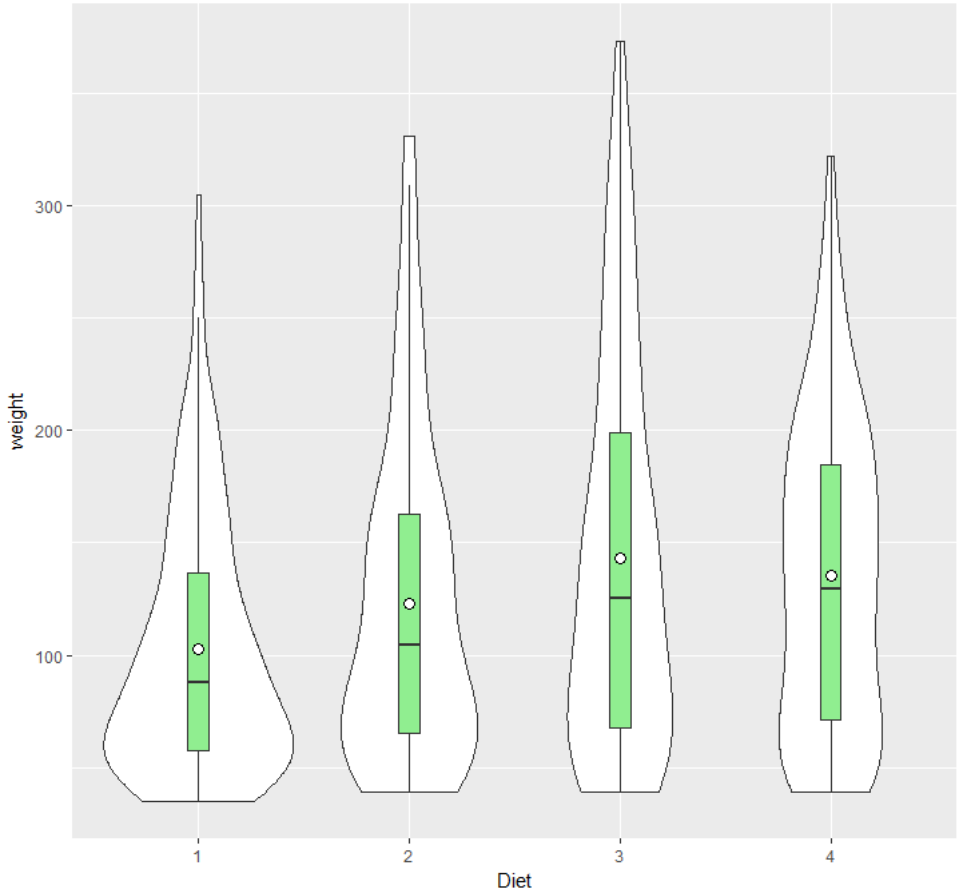

```
ggplot(ChickWeight, aes(x=Diet, y=weight)) +
geom_boxplot(fill="lightgreen")
```

Visualize the distribution of data: III. Boxplot



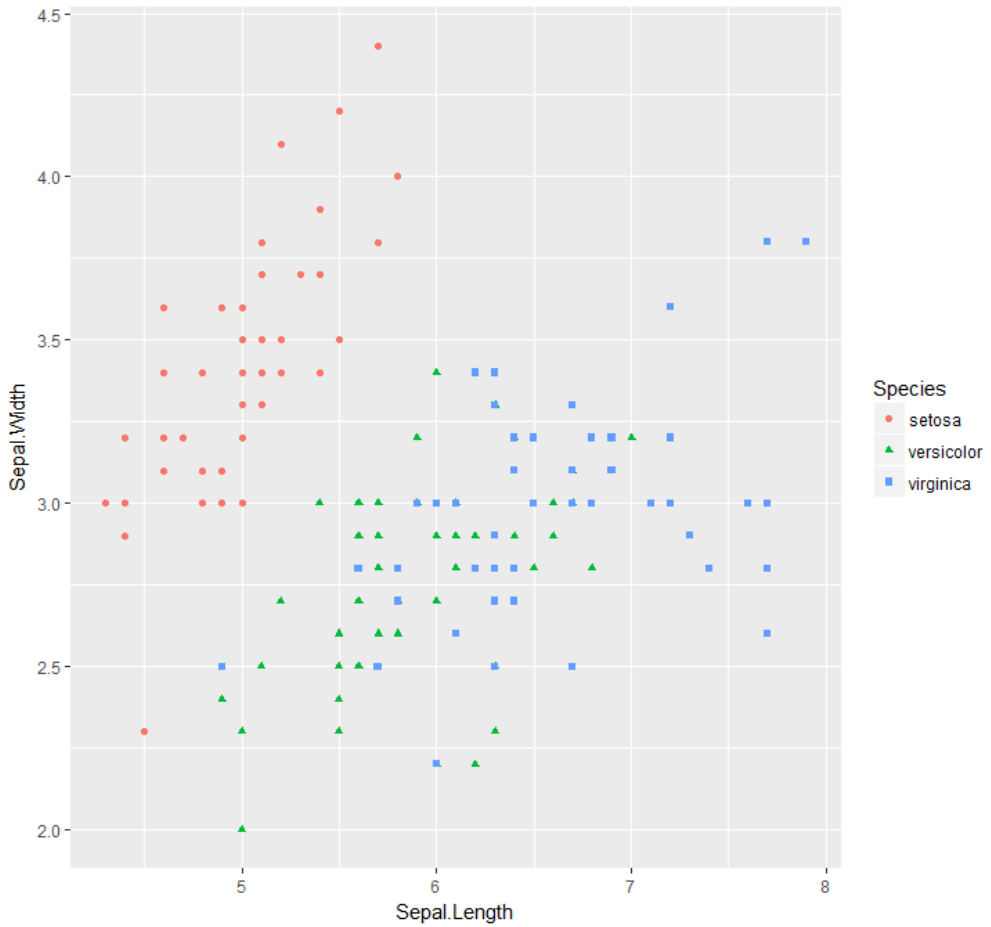

```
ggplot(ChickWeight, aes(x=Diet, y=weight)) +
geom_boxplot(fill="lightgreen",notch=TRUE)+
stat_summary(fun.y="mean",geom="point",  fill="blue", shape=21, size=3)
```

Visualize the distribution of data: III. Boxplot



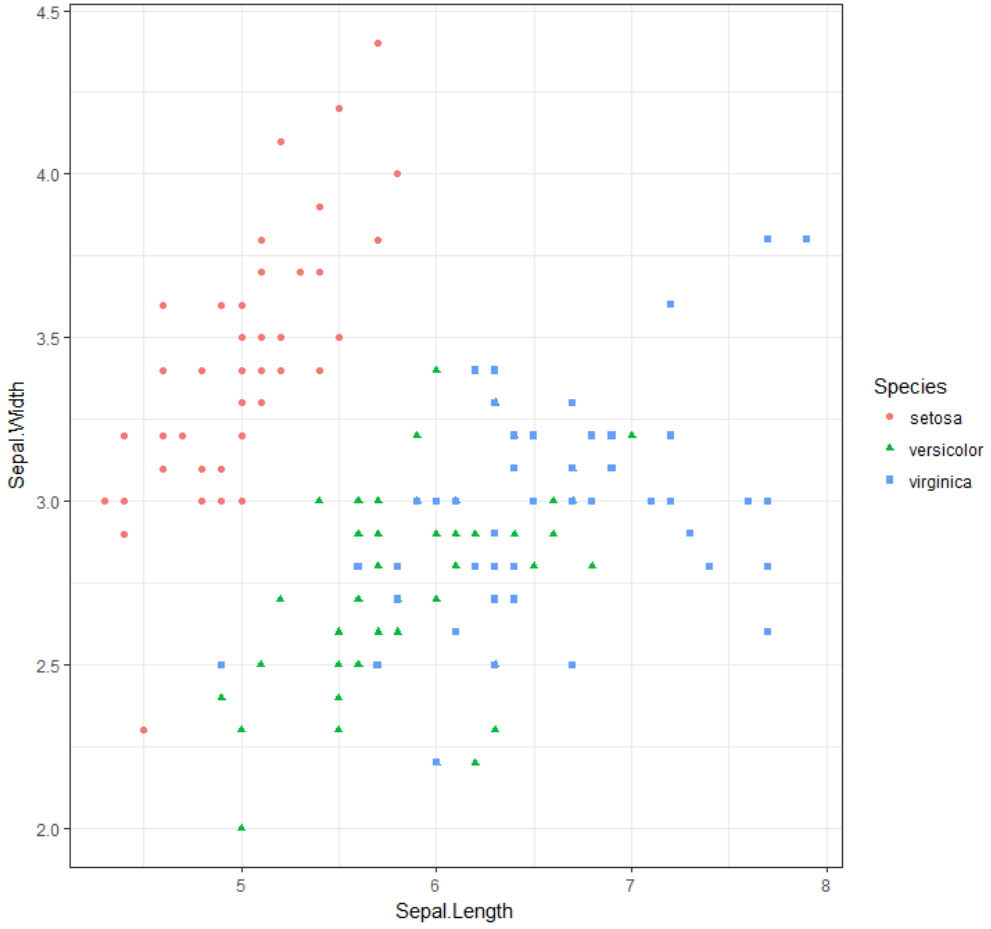

```
ggplot(ChickWeight, aes(x=Diet, y=weight)) +
geom_boxplot(notch=TRUE)+
stat_summary(fun.y="mean",geom="point",  fill="blue", shape=21, size=3)
```

Change the overall appearance of the graph



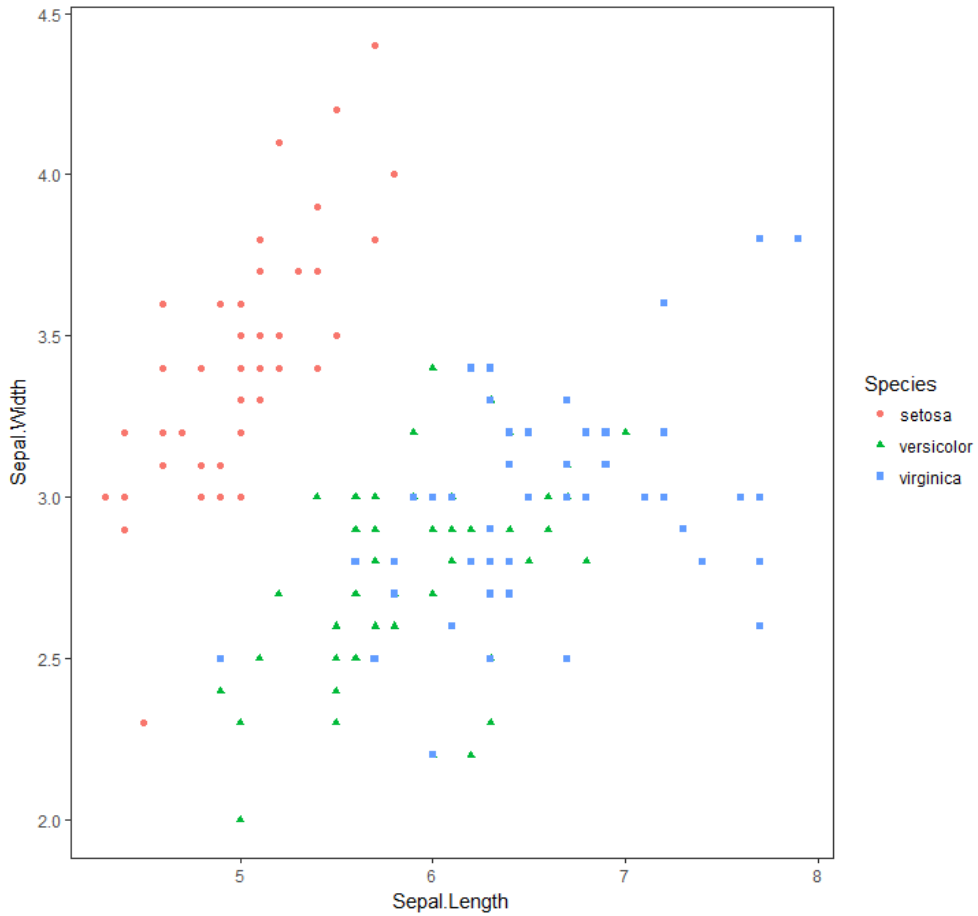

```
bp <- ggplot(iris, aes(x=Sepal.Length, y=Sepal.Width, color=Species, shape=Species))+
        geom_point()
bp
```

Set background to be black and white



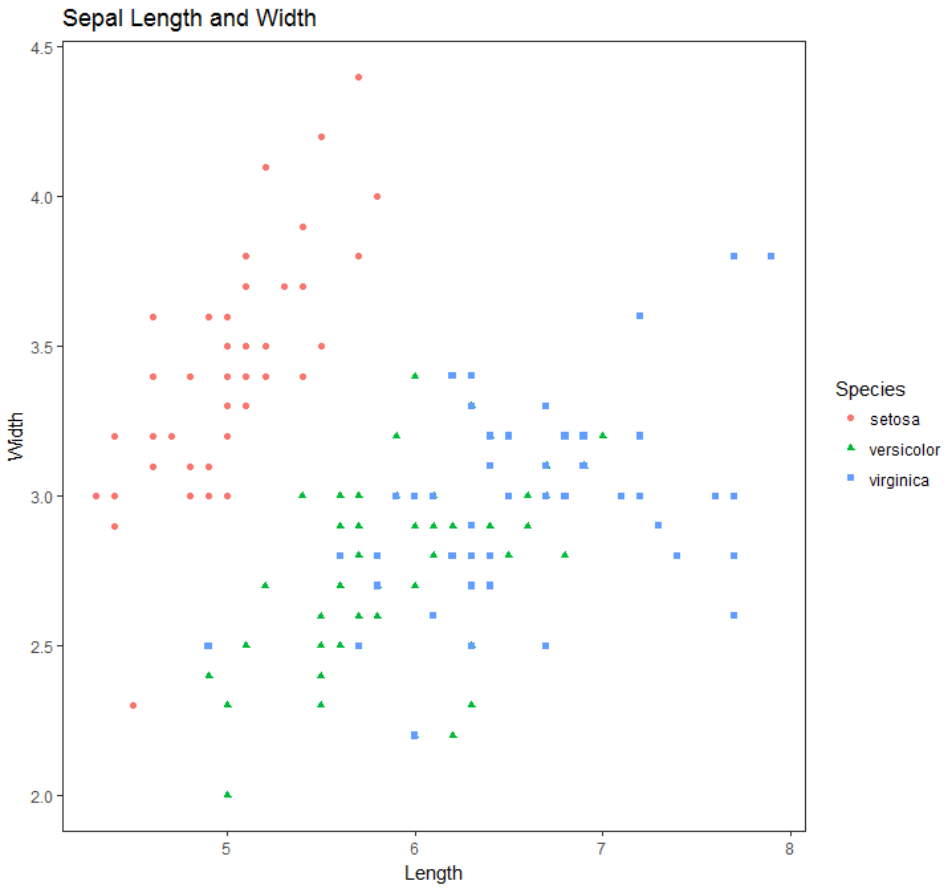

```
bp1<-bp + theme_bw()
bp1
```

Remove grid lines



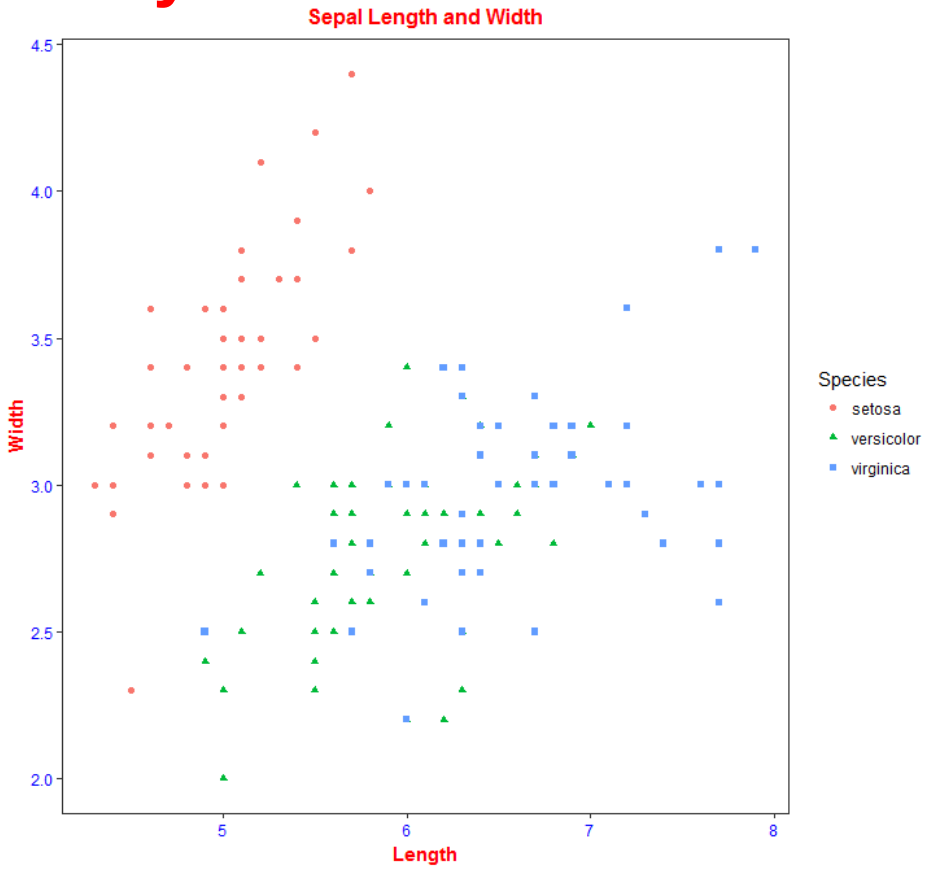

```
bp2<-bp1 + theme(panel.grid.major = element_blank(),panel.grid.minor = element_blank())
bp2
```

Add Labels



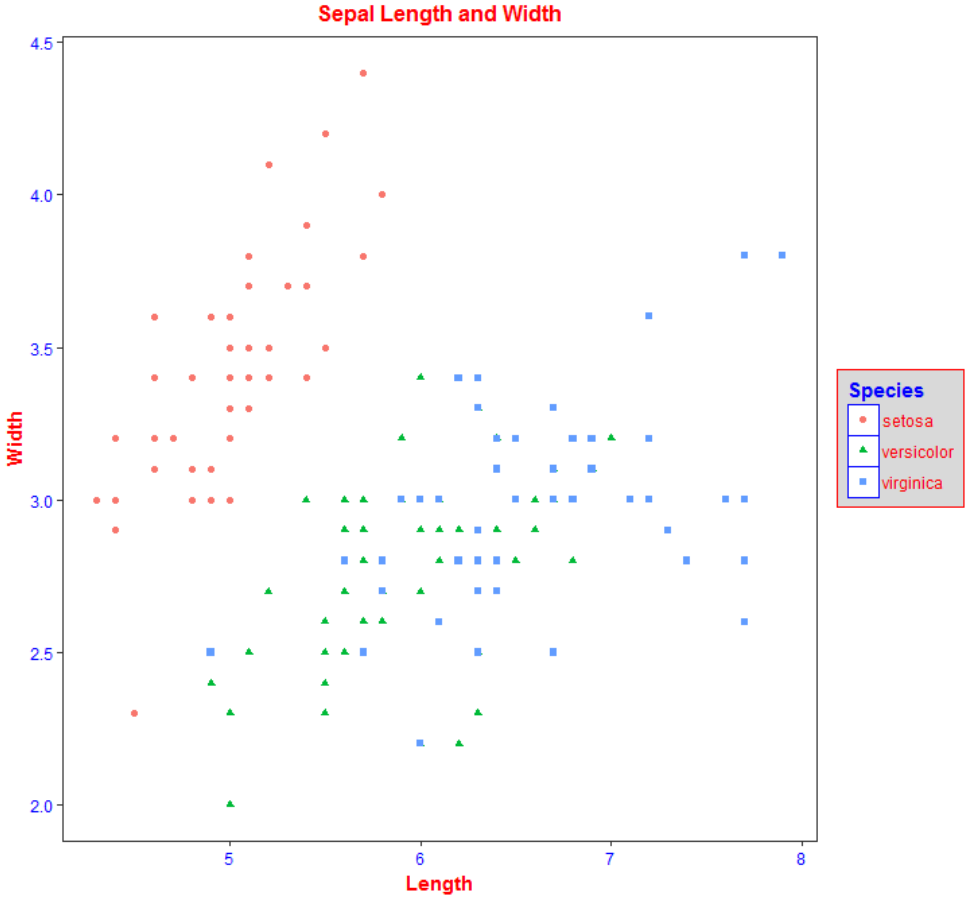

```
bp3<- bp2+labs(x="Length",y="Width",title="Sepal Length and Width")
bp3
```

Modify title and axis labels




```
bp4<- bp3 +
  theme(axis.title.x = element_text(colour="red", size=11,face="bold"),
        axis.text.x = element_text(colour="blue"),
        axis.title.y = element_text(colour="red", size=11,face="bold", angle = 90),
        axis.text.y = element_text(colour="blue"),
        plot.title = element_text(colour="red", size=12, face="bold", hjust=0.5))
bp4
```

Modify legend




```
bp4 +
  theme(legend.background = element_rect(fill="grey85", colour="red", size=0.2),
        legend.title = element_text(colour="blue", face="bold", size=11),
        legend.text = element_text(colour="red"),
        legend.key = element_rect(colour="blue", size=0.2))
```

Simplified code (plotting for single group)

Default tools in R		ggplot
	Scatter plot	
plot (x, y, type="p")		ggplot(data, aes(x= , y=,)) + geom_point()
	Line graph	
plot (x, y, type="l")		ggplot(data, aes(x= , y=,)) + geom_line()
	Bar graph	
barplot(x=table(x))		ggplot(data, aes(x=factor())) + geom_bar()
	Histogram	
hist(x)		ggplot(data, aes(x=)) + geom_histogram()
	Boxplot	
boxplot(data=data, y~x)		ggplot(data, aes(x=factor(),y=)) + geom_boxplot()

Simplified code (plotting for multiple groups)

Scatter plot

```
ggplot(data, aes(x= , y=, shape=factor(group), colour=factor(group))) +
  geom_point()
```

Bar graph

```
ggplot(data, aes(x=factor(), fill=factor(group))) +
  geom_bar(postion="dodge")
```

Histogram

```
ggplot(data, aes(x=,fill=factor(group))) +
  geom_histogram(position ="identity")
```

Density Curve

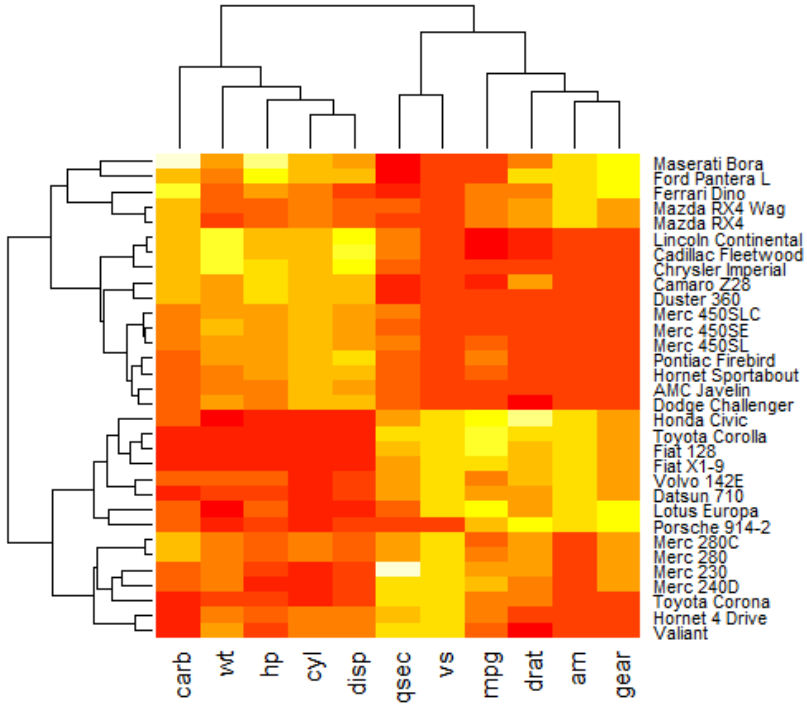
```
ggplot(data, aes(x=,fill=factor(group))) +
  geom_density()
```

Boxplot

```
ggplot(data, aes(x=factor(),y=, fill=factor(group))) +
  geom_boxplot()
```

Heatmap

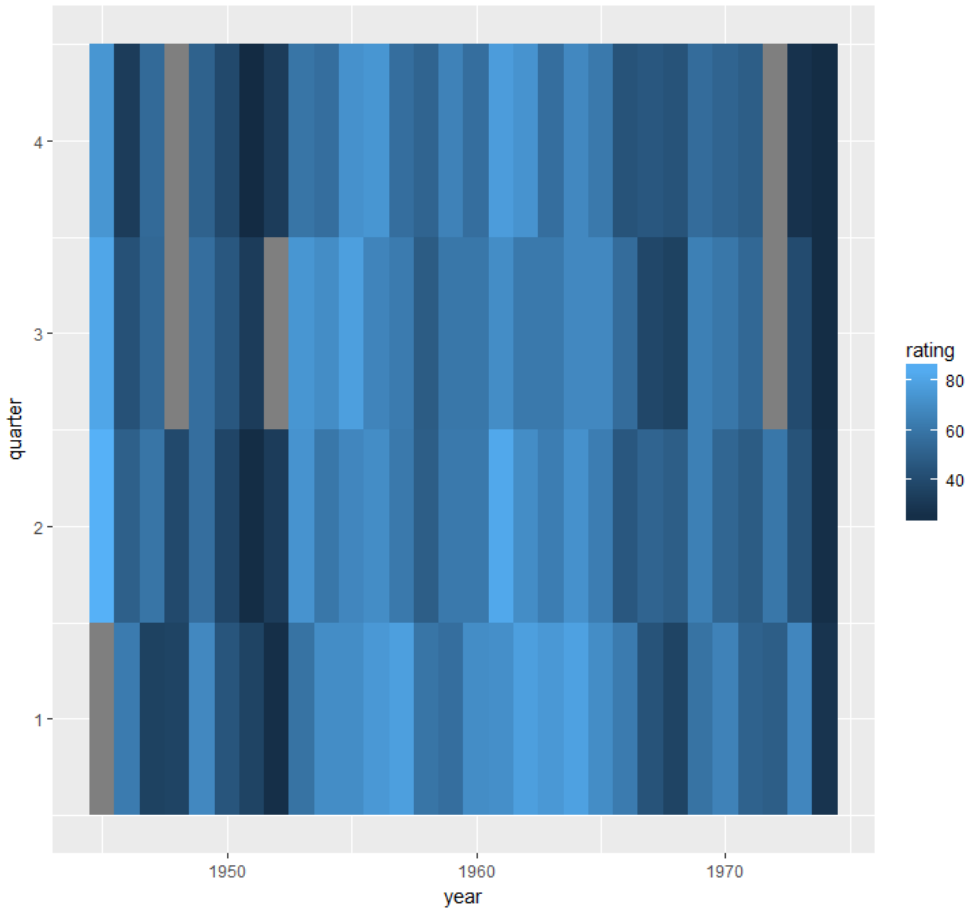
Heat maps are used to visualize the level of signal of one variable across different time point or other groups




```
# scale data to mean=0, sd=1 and convert to matrix
mtscaled <- as.matrix(scale(mtcars))
# create heatmap and don't reorder columns
 heatmap(mtscaled, Colv=F, scale='none')
```

Heatmap

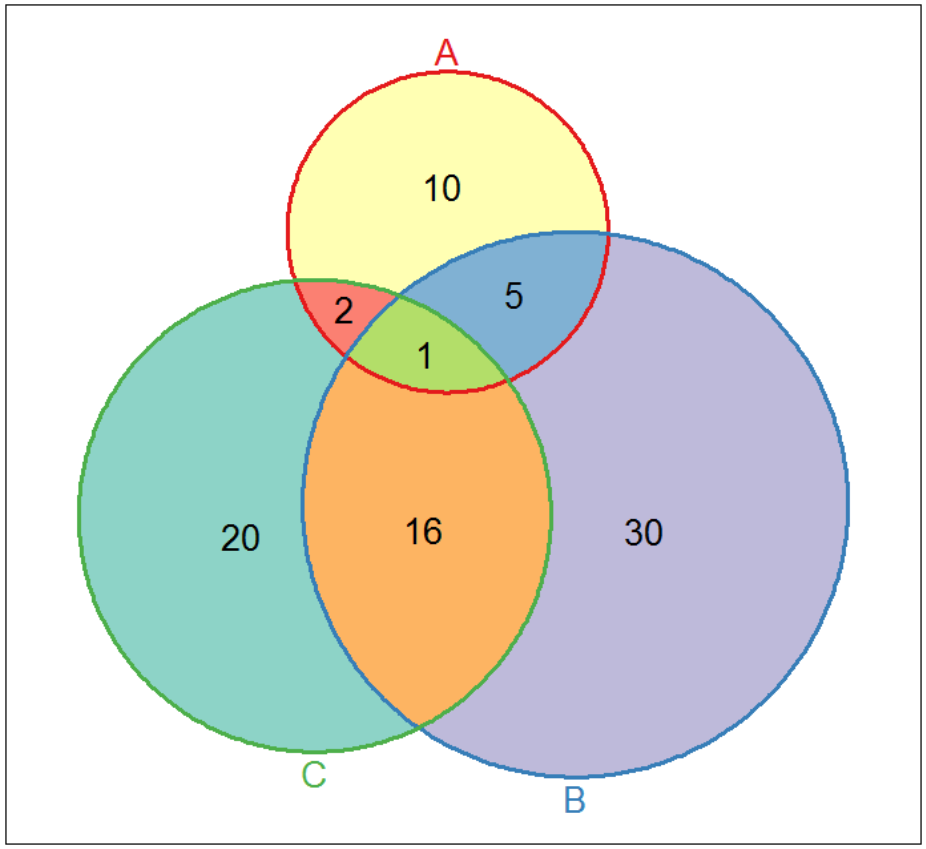
Using ggplot2 to draw heat map on time series data




```
ggplot(pres_rating, aes(x=year, y=quarter, fill=rating))+geom_tile()
```

Venn Diagram

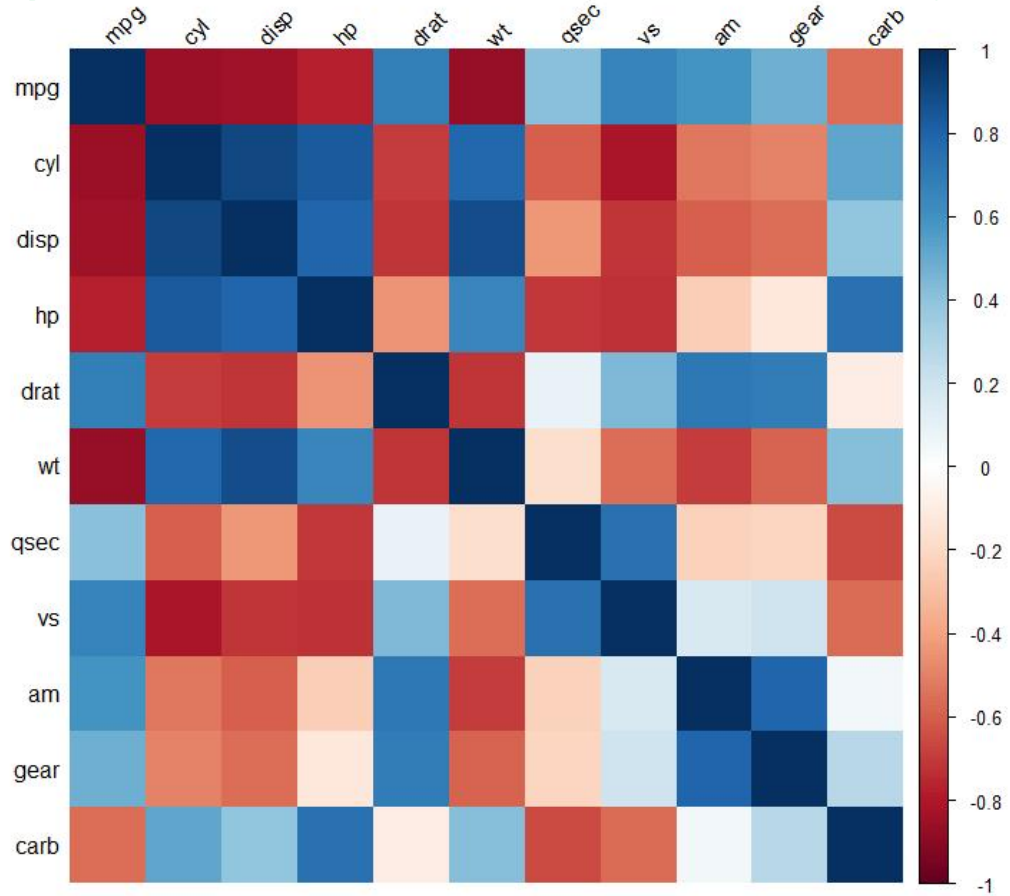
Venn diagrams are used to check the overlap among different groups




```
library(Vennerable)
V <- Venn(SetNames=c('A','B','C'),Weight=c(0,10,30,5,20,2,16,1))
plot(V, doWeights=TRUE,type='circles')
```

Correlation Plot

Correlation plot is used to check the association or similarity among variables




```
mcor<-cor(mtcars)
library(corrplot)
corrplot(mcor, method="shade", shade.col=NA, tl.col="black", tl.srt=45)
```

Output for publication or presentation

Output to PNG:

```
png("myplot-%d.png", width=400, height=400)
plot(mtcars$wt, mtcars$mpg)
ggplot(mtcars, aes(x=wt, y=mpg)) + geom_point()
dev.off()
```

Output to TIFF:

```
tiff("myplot-%d.tiff", width=400, height=400)
plot(mtcars$wt, mtcars$mpg)
ggplot(mtcars, aes(x=wt, y=mpg)) + geom_point()
dev.off()
```

Output to PDF:

```
pdf("myplot.pdf", width=4, height=4)
plot(mtcars$wt, mtcars$mpg)
ggplot(mtcars, aes(x=wt, y=mpg)) + geom_point()
dev.off()
```

Output to postscript:

```
postscript("myplot.eps", width=4, height=4)
plot(mtcars$wt, mtcars$mpg)
ggplot(mtcars, aes(x=wt, y=mpg)) + geom_point()
dev.off()
```

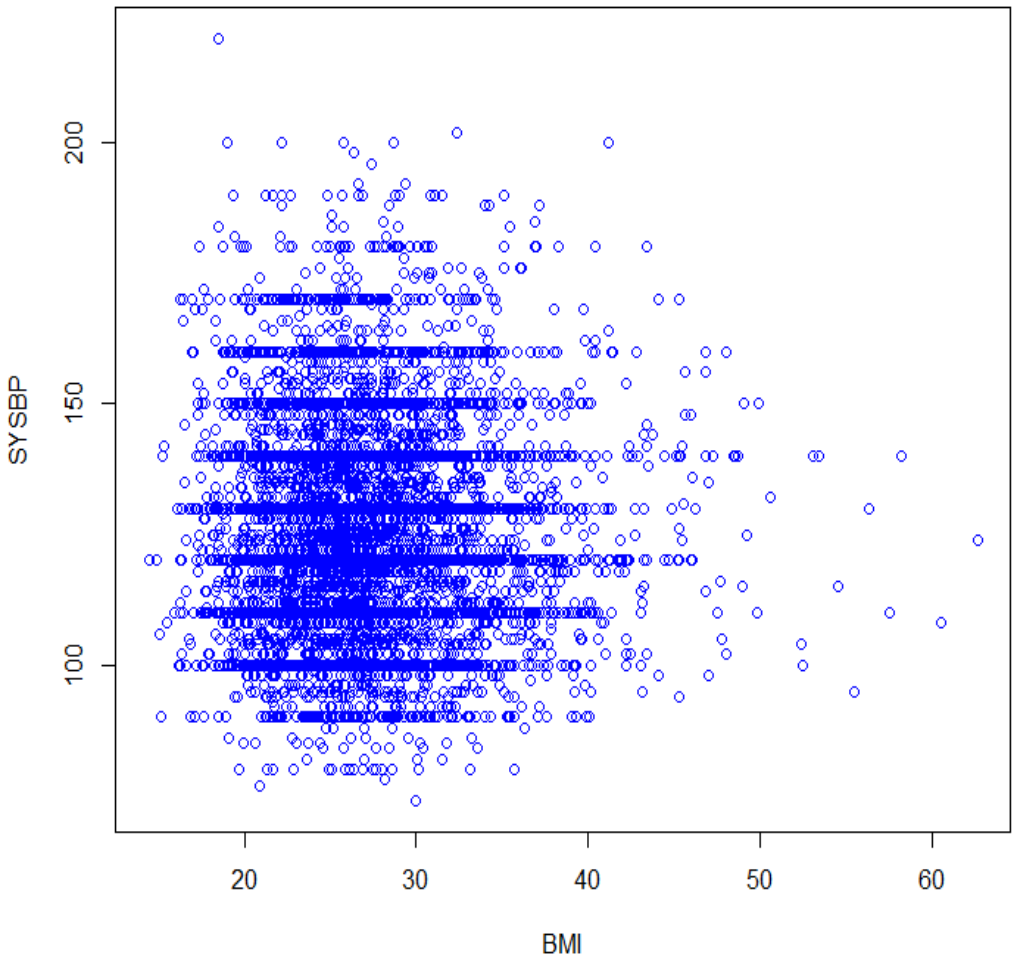
Note: dev .off () is to let R know you're finished with plotting commands and it can output the file.

Practice Session

1. An R script that includes all the code covered in this presentation is provided with detailed step-wise annotation and you will go over all the plots during the first half of the practice session.

2. You will try to answer a list of five questions and draw plots to visualize the practicing clinical trial DIG NHLBT Teaching data set

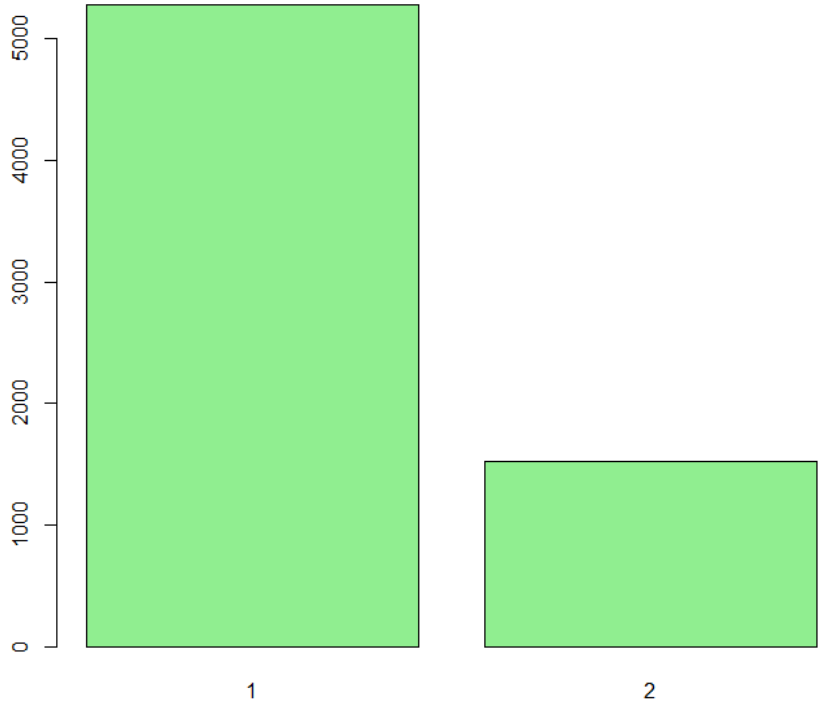
Q1: Check the relationship between BMI and Systolic BP (SYSBP)

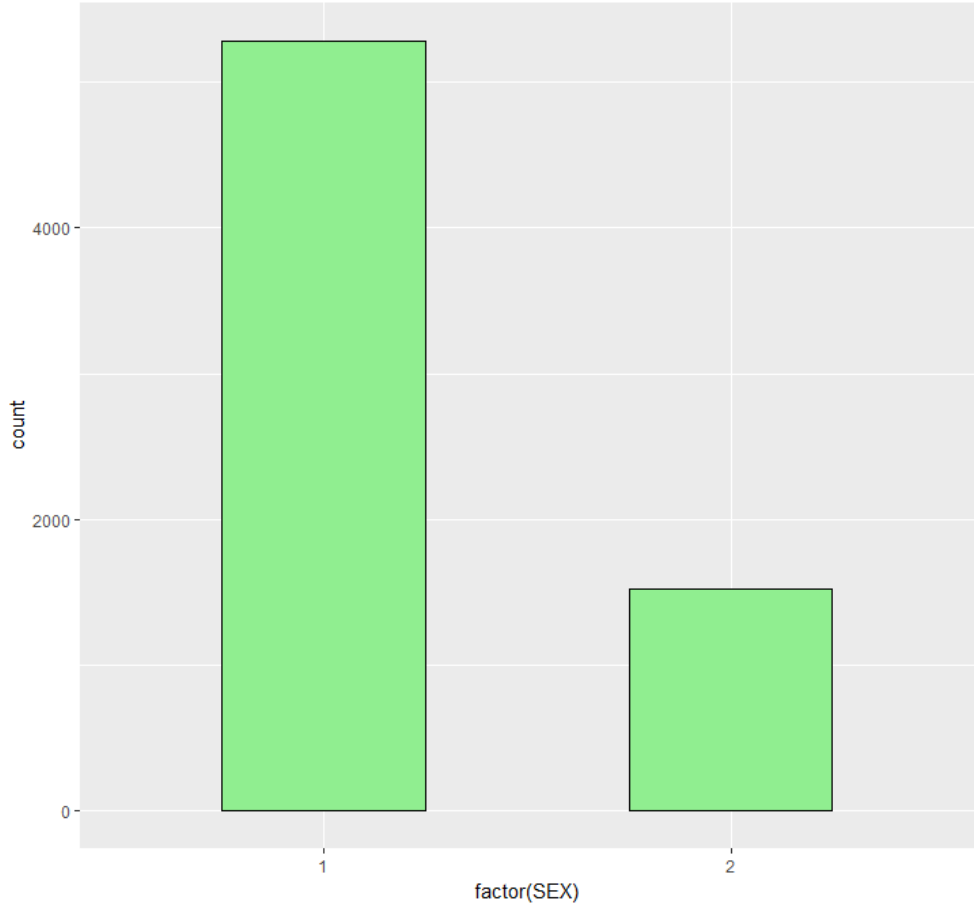



```
plot(dig$BMI,dig$SYSBP, xlab="BMI", ylab="SYSBP", col="blue")
```


```
ggplot(dig, aes(x=BMI, y=SYSBP)) +
 geom_point(colour="blue", shape=21)
```

Q2: Plot the number of patients for different SEX groups

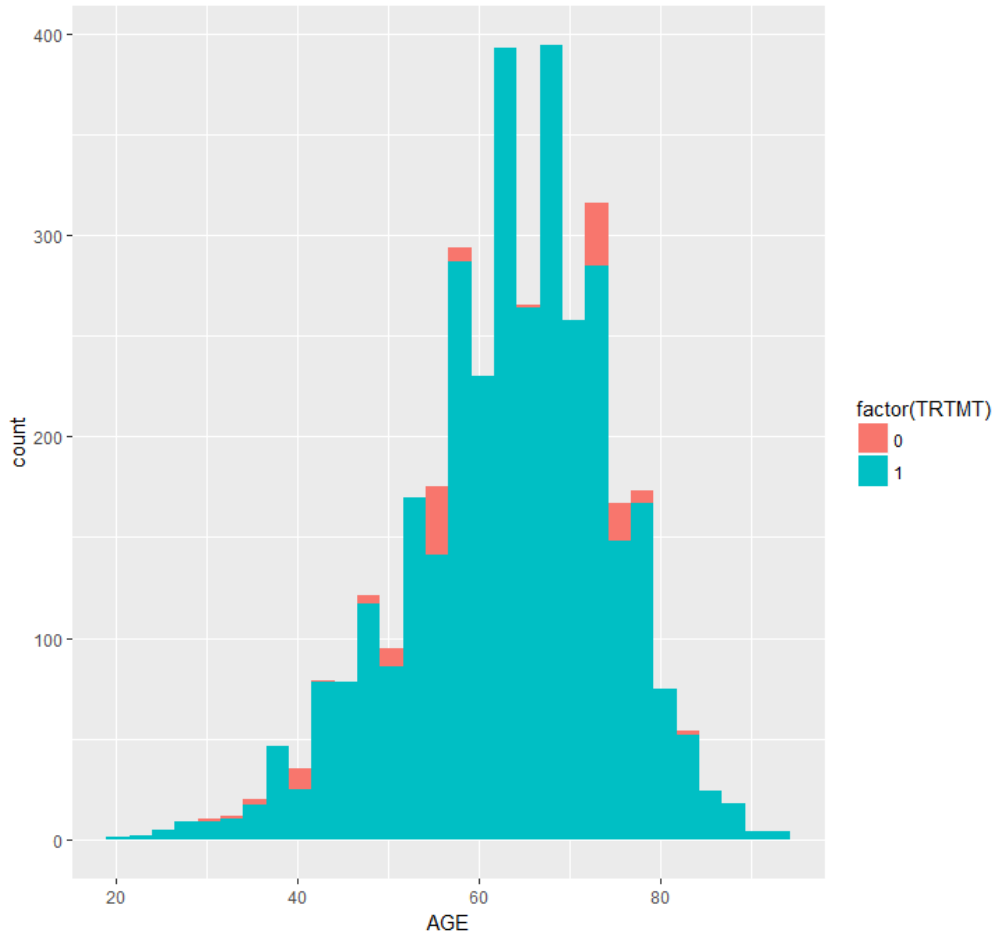





```
barplot(table(dig$SEX), col="lightgreen")
```

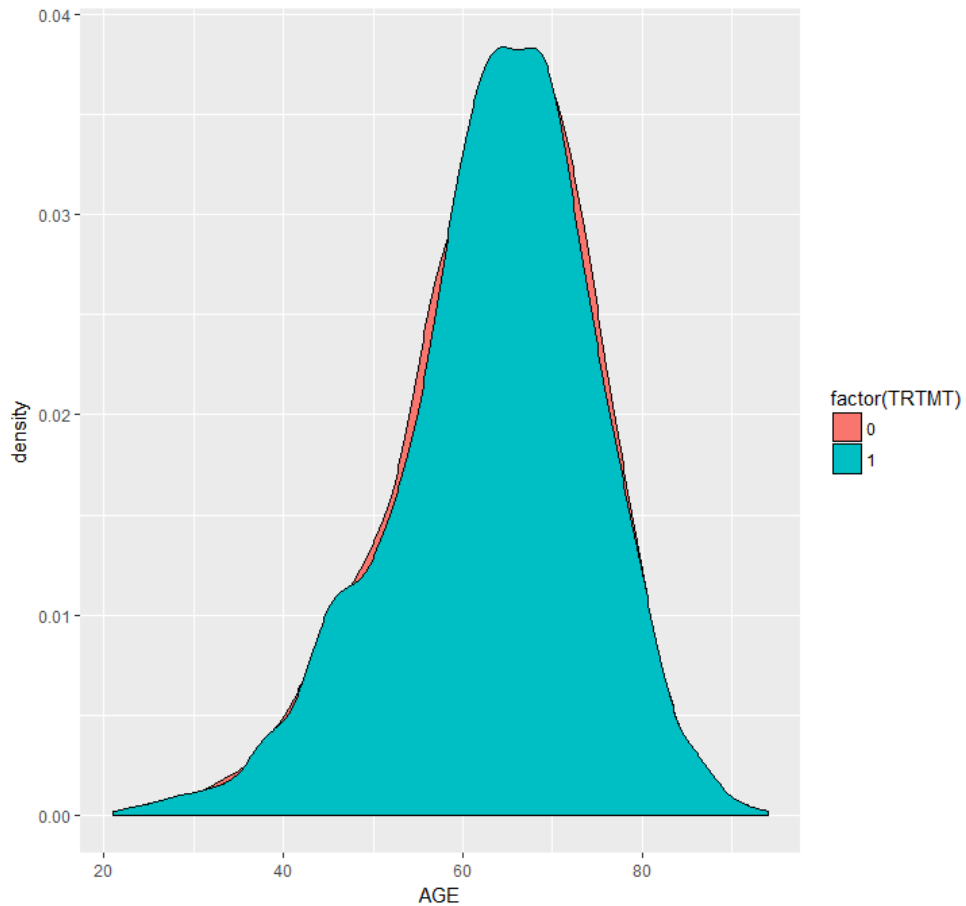
```
ggplot(dig, aes(x=factor(SEX))) +
geom_bar( colour="black", fill="lightgreen", width=0.5)
```

Q3: Use ggplot to check the distribution of AGE in different treatment groups (TRTMT) using three different types of plots - histogram



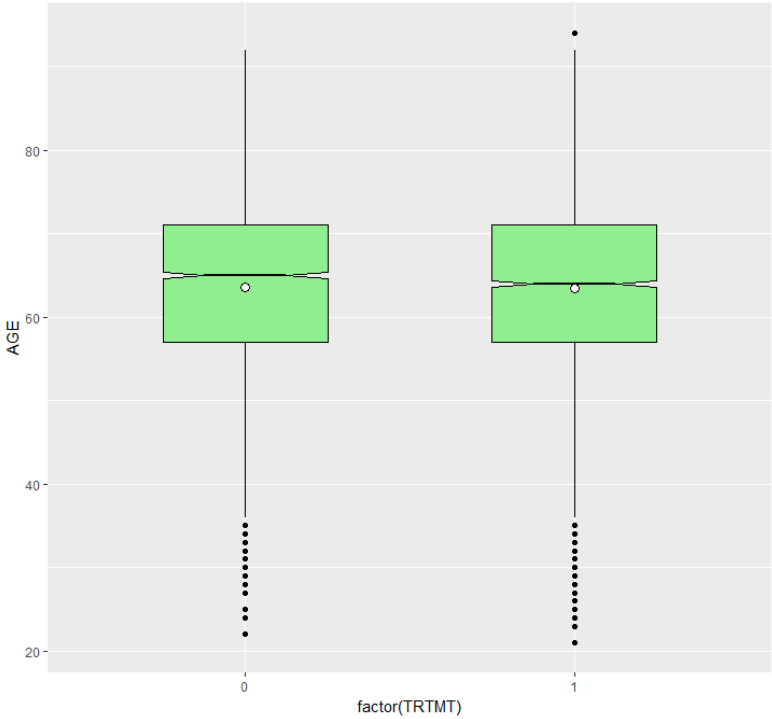

```
ggplot(dig, aes(x=AGE, fill=factor(TRTMT))) +
geom_histogram(position="identity")
```

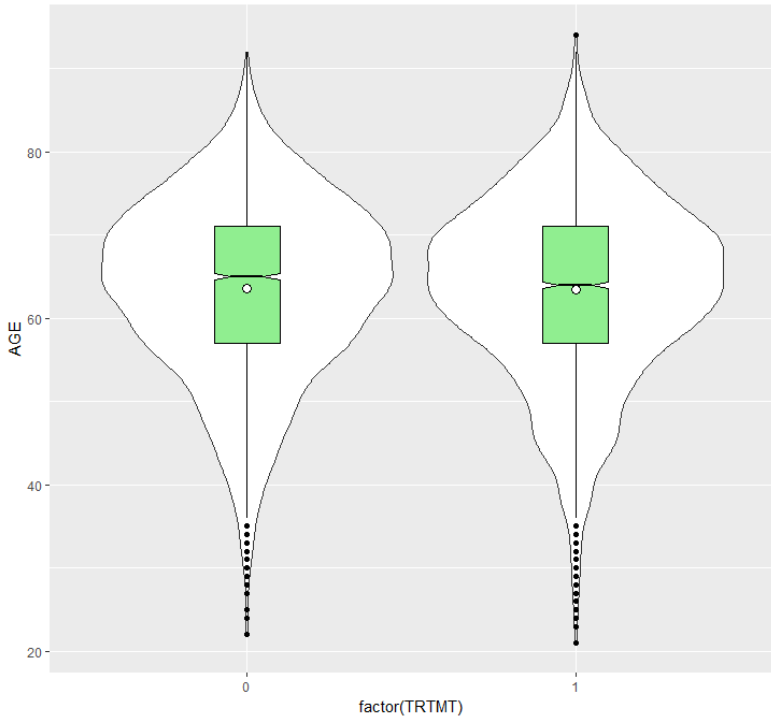
Q3: use ggplot to check the distribution of AGE in different treatment groups (TRTMT) using three different types of plots – density curve




```
ggplot(dig, aes(x=AGE, fill=factor(TRTMT))) +
geom_density()
```

Q3: use ggplot to check the distribution of AGE in different treatment groups (TRTMT) using three different types of plots – boxplot

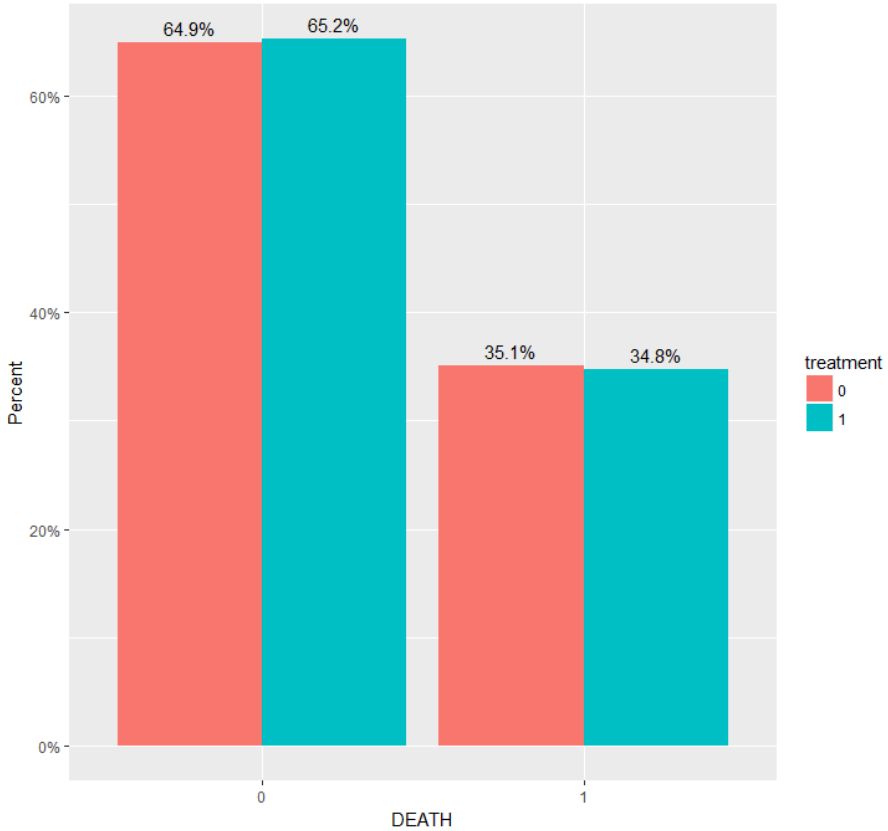





```
ggplot(dig, aes(x=factor(TRTMT), y=AGE)) +
geom_boxplot(notch=TRUE, width=0.5, colour="black", fill="lightgreen") +
  stat_summary(fun.y="mean",geom="point",  fill="white", shape=21, size=3)
```

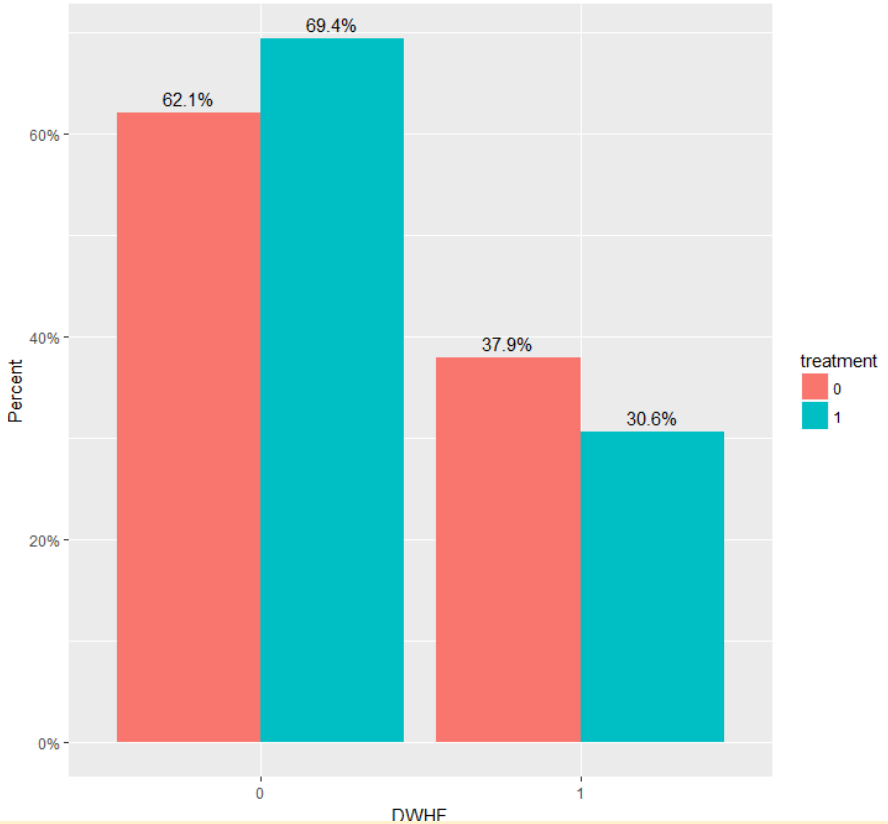
```
ggplot(dig, aes(x=factor(TRTMT), y=AGE))  + geom_violin() +
geom_boxplot(notch=TRUE, width=0.2, colour="black", fill="lightgreen")+
  stat_summary(fun.y="mean",geom="point",  fill="white", shape=21, size=3)
```

Q4. A. plot the percentage of DEATH in different treatment groups (TRTMT)




```
ggplot(dig, aes(x= factor(DEATH),  group=factor(TRTMT))) +
  geom_bar(aes(y = ..prop.., fill = factor(TRTMT)), stat="count", position="dodge")+
  geom_text(aes( label = scales::percent(..prop..), y= ..prop.. ), stat= "count",position=position_dodge(0.9),
vjust = -.5) +
  labs(x="DEATH",y = "Percent", fill="treatment") +
  scale_y_continuous(labels=scales::percent)
```

Q4. B. plot the percentage of death attributed to worsening heart failure (DWHF) in different treatment groups (TRTMT)




```
ggplot(dig, aes(x= factor(DWHF),  group=factor(TRTMT))) +
  geom_bar(aes(y = ..prop.., fill = factor(TRTMT)), stat="count", position="dodge")+
  geom_text(aes( label = scales::percent(..prop..), y= ..prop.. ), stat= "count",position=position_dodge(0.9),
vjust = -.5) +
  labs(x="DWHF",y = "Percent", fill="treatment") +
  scale_y_continuous(labels=scales::percent)
```

Q5 take plot from Q4b, try to polish the graph

```
Q4B<-ggplot(dig, aes(x= factor(DWHF),  group=factor(TRTMT))) +
  geom_bar(aes(y = ..prop.., fill = factor(TRTMT)), stat="count", position="dodge")+
  geom_text(aes( label = scales::percent(..prop..), y= ..prop.. ), stat=
"count",position=position_dodge(0.9), vjust = -.5) +
  labs(x="DWHF",y = "Percent", fill="treatment") +
  scale_y_continuous(labels=scales::percent)
Q4B+
  theme_bw()+
  theme(panel.grid.major = element_blank(),panel.grid.minor = element_blank())+
  labs(x="DWHF", y="Percentage", title="Percentage of DWHF in Different Treatment
Group") +
  theme(axis.title.x = element_text(colour="red", size=11,face="bold"),
        axis.text.x = element_text(colour="blue"),
        axis.title.y = element_text(colour="red", size=11,face="bold", angle = 90),
        axis.text.y = element_text(colour="blue"),
        plot.title = element_text(colour="red", size=12, face="bold", hjust=0.5)) +
  theme(legend.background = element_rect(fill="grey85", colour="red", size=0.2),
        legend.title = element_text(colour="blue", face="bold", size=11),
        legend.text = element_text(colour="red"))
```