library(ggplot2)  
library(reshape2)  
library(ggthemes)  
require(gridExtra)

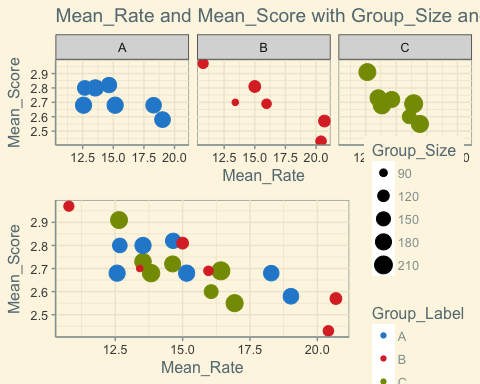
## Loading required package: gridExtra

#set new label  
new\_label <- c("Date", "Mean\_Rate", "Mean\_Score", "Group\_Size", "Group\_Label")  
data = read.csv('STK\_Data\_Gates\_2017.csv')  
names(data) <- new\_label  
#Date  
new\_Date\_format<-as.Date((data$Date),"%m/%d/%Y")  
data$Date <-strftime(new\_Date\_format,"%m/%d")  
#stats 5 summary  
print(summary(data))

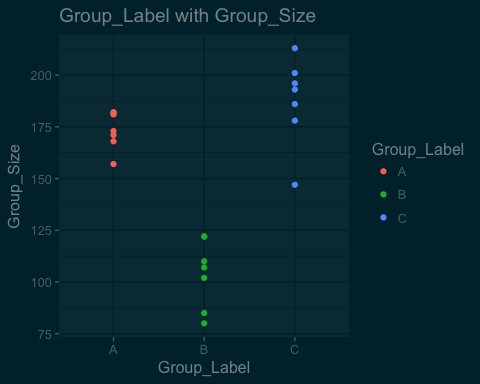
## Date Mean\_Rate Mean\_Score Group\_Size   
## Length:21 Min. :10.78 Min. :2.430 Min. : 80   
## Class :character 1st Qu.:13.53 1st Qu.:2.600 1st Qu.:122   
## Mode :character Median :15.00 Median :2.690 Median :171   
## Mean :15.56 Mean :2.691 Mean :155   
## 3rd Qu.:16.93 3rd Qu.:2.800 3rd Qu.:182   
## Max. :20.69 Max. :2.970 Max. :213   
## Group\_Label  
## A:7   
## B:7   
## C:7   
##   
##   
##

11

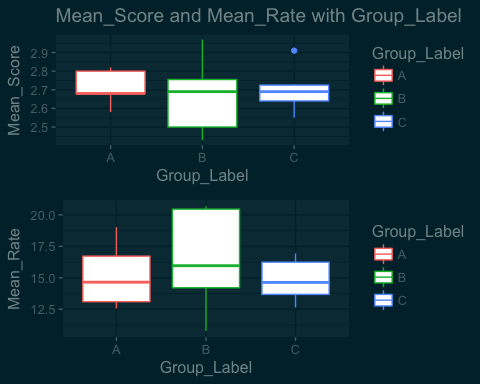
#different groups  
  
g1<-ggplot(data, aes(Mean\_Rate,Mean\_Score,color = Group\_Label,size = Group\_Size)) + geom\_point() + theme\_solarized() +  
scale\_color\_solarized() + ggtitle("Mean\_Rate and Mean\_Score with Group\_Size and Group\_Label") + facet\_grid(. ~ Group\_Label) + theme(legend.position="none")  
g2<-ggplot(data, aes(Mean\_Rate,Mean\_Score,color = Group\_Label,size = Group\_Size)) + geom\_point() + theme\_solarized() +  
scale\_color\_solarized()   
grid.arrange(g1,g2,nrow=2,ncol=1)



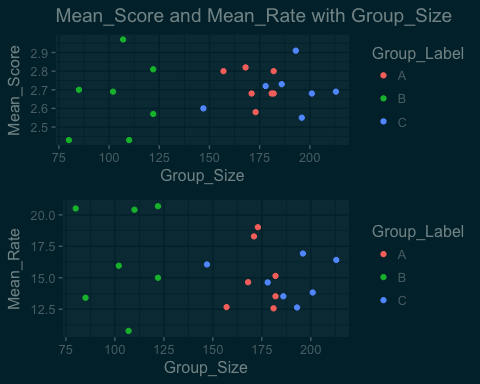
ggplot(data, aes(Group\_Label,Group\_Size,color=Group\_Label)) + geom\_point() + theme\_solarized\_2(light = FALSE) + ggtitle("Group\_Label with Group\_Size")

 22

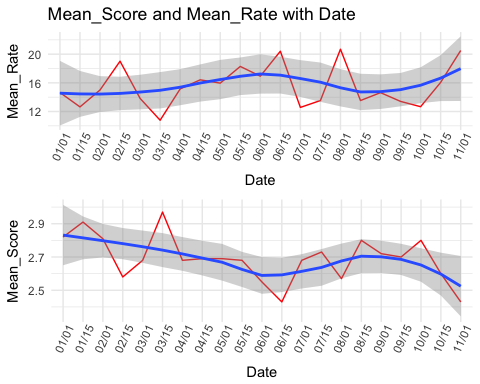
#subplots  
  
#boxplot  
g1<-ggplot(data, aes(Group\_Label,Mean\_Score,color = Group\_Label)) + geom\_boxplot() + theme\_solarized\_2(light = FALSE) + ggtitle("Mean\_Score and Mean\_Rate with Group\_Label")  
g2<-ggplot(data, aes(Group\_Label,Mean\_Rate,color = Group\_Label)) + geom\_boxplot() + theme\_solarized\_2(light = FALSE)   
grid.arrange(g1,g2,nrow=2,ncol=1)



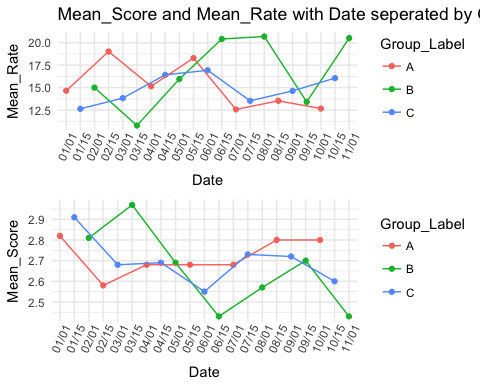
g1<-ggplot(data, aes(Group\_Size,Mean\_Score,color = Group\_Label)) + geom\_point() + theme\_solarized\_2(light = FALSE) + ggtitle("Mean\_Score and Mean\_Rate with Group\_Size")  
g2<-ggplot(data, aes(Group\_Size,Mean\_Rate,color = Group\_Label)) + geom\_point() + theme\_solarized\_2(light = FALSE)   
grid.arrange(g1,g2,nrow=2,ncol=1)



#subplot & timeline  
g1<-ggplot(data, aes(Date,Mean\_Rate,group=1)) + geom\_line(color = 'red') + geom\_smooth(method = 'loess') + scale\_fill\_ptol() +  
 theme\_minimal() + theme(axis.text.x= element\_text(angle = 65, vjust = 0.7)) + ggtitle("Mean\_Score and Mean\_Rate with Date")  
g2<-ggplot(data, aes(Date,Mean\_Score,group=1)) + geom\_line(color = 'red') + geom\_smooth(method = 'loess') + scale\_fill\_ptol() +  
 theme\_minimal() + theme(axis.text.x= element\_text(angle = 65, vjust = 0.7))  
grid.arrange(g1,g2,nrow=2,ncol=1)



#timeline   
g1 <- ggplot(data, aes(Date,Mean\_Rate,group = Group\_Label,color = Group\_Label)) + geom\_point() + geom\_line() + scale\_x\_discrete(limits= data$Date) + scale\_fill\_ptol() + theme\_minimal() + theme(axis.text.x= element\_text(angle = 65, vjust = 0.7)) + ggtitle("Mean\_Score and Mean\_Rate with Date seperated by Group")  
g2 <- ggplot(data, aes(Date,Mean\_Score,group = Group\_Label,color = Group\_Label)) + geom\_point() + geom\_line() + scale\_x\_discrete(limits= data$Date) + scale\_fill\_ptol() + theme\_minimal() + theme(axis.text.x= element\_text(angle = 65, vjust = 0.7))   
grid.arrange(g1,g2,nrow=2,ncol=1)



g1 <- ggplot(data, aes(Date,Group\_Label,color = Group\_Label,group =1)) + geom\_point() + geom\_step() + scale\_x\_discrete(limits= data$Date) + scale\_fill\_ptol() + theme\_minimal() + theme(axis.text.x= element\_text(angle = 65, vjust = 0.7)) + ggtitle("Group\_Label and Group\_Size with Date seperated by Group")  
g2 <- ggplot(data, aes(Date,Group\_Size,group = Group\_Label,color = Group\_Label)) + geom\_point() + geom\_line() + scale\_x\_discrete(limits= data$Date) + scale\_fill\_ptol() + theme\_minimal() + theme(axis.text.x= element\_text(angle = 65, vjust = 0.7))   
grid.arrange(g1,g2,nrow=2,ncol=1)

