Game Analysis

Bowen Cheng

November 2, 2018

## R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

library(ggplot2)  
library(dplyr)

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

library(corrplot)

## corrplot 0.84 loaded

library(corrgram)  
library(gridExtra)

##   
## Attaching package: 'gridExtra'

## The following object is masked from 'package:dplyr':  
##   
## combine

Games <- read.csv("D:/Ryerson Course/CKME136/Video\_Games\_Sales\_as\_at\_22\_Dec\_2016.csv",stringsAsFactors = FALSE)  
str(Games)

## 'data.frame': 16719 obs. of 16 variables:  
## $ Name : chr "Wii Sports" "Super Mario Bros." "Mario Kart Wii" "Wii Sports Resort" ...  
## $ Platform : chr "Wii" "NES" "Wii" "Wii" ...  
## $ Year\_of\_Release: chr "2006" "1985" "2008" "2009" ...  
## $ Genre : chr "Sports" "Platform" "Racing" "Sports" ...  
## $ Publisher : chr "Nintendo" "Nintendo" "Nintendo" "Nintendo" ...  
## $ NA\_Sales : num 41.4 29.1 15.7 15.6 11.3 ...  
## $ EU\_Sales : num 28.96 3.58 12.76 10.93 8.89 ...  
## $ JP\_Sales : num 3.77 6.81 3.79 3.28 10.22 ...  
## $ Other\_Sales : num 8.45 0.77 3.29 2.95 1 0.58 2.88 2.84 2.24 0.47 ...  
## $ Global\_Sales : num 82.5 40.2 35.5 32.8 31.4 ...  
## $ Critic\_Score : int 76 NA 82 80 NA NA 89 58 87 NA ...  
## $ Critic\_Count : int 51 NA 73 73 NA NA 65 41 80 NA ...  
## $ User\_Score : chr "8" "" "8.3" "8" ...  
## $ User\_Count : int 322 NA 709 192 NA NA 431 129 594 NA ...  
## $ Developer : chr "Nintendo" "" "Nintendo" "Nintendo" ...  
## $ Rating : chr "E" "" "E" "E" ...

###Data Preparation###  
#Remove attributes not part of the analysis  
Games$Name <- NULL  
Games$Platform <- NULL  
Games$Critic\_Count <- NULL  
Games$User\_Count <- NULL  
Games$Developer <- NULL  
Games$Critic\_Score <- as.numeric(Games$Critic\_Score)  
  
#Remove all rows with NA and missing values  
GamesCleaned <- na.omit(Games)  
GamesCleaned2 <- filter(GamesCleaned, Rating!='')  
GamesCleaned2$User\_Score <- as.numeric(GamesCleaned2$User\_Score)

## Warning: NAs introduced by coercion

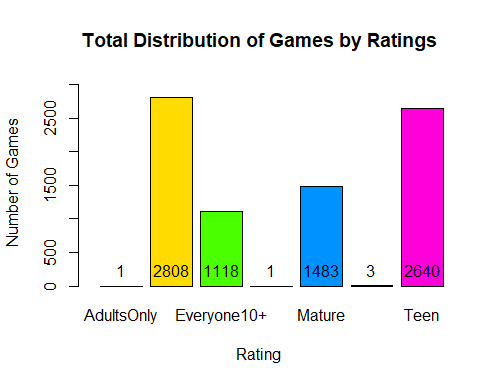
#Renaming the abbreviated ratings to their full description  
GamesCleaned2$Rating[GamesCleaned2$Rating=="RP"]="RatingPending"  
GamesCleaned2$Rating[GamesCleaned2$Rating=="EC"]="EarlyChildhood"  
GamesCleaned2$Rating[GamesCleaned2$Rating=="E"]="Everyone"  
GamesCleaned2$Rating[GamesCleaned2$Rating=="E10+"]="Everyone10+"  
GamesCleaned2$Rating[GamesCleaned2$Rating=="T"]="Teen"  
GamesCleaned2$Rating[GamesCleaned2$Rating=="M"]="Mature"  
GamesCleaned2$Rating[GamesCleaned2$Rating=="AO"]="AdultsOnly"  
GamesCleaned2$Rating[GamesCleaned2$Rating=="K-A"]="KidsToAdults"  
  
str(GamesCleaned2)

## 'data.frame': 8054 obs. of 11 variables:  
## $ Year\_of\_Release: chr "2006" "2008" "2009" "2006" ...  
## $ Genre : chr "Sports" "Racing" "Sports" "Platform" ...  
## $ Publisher : chr "Nintendo" "Nintendo" "Nintendo" "Nintendo" ...  
## $ NA\_Sales : num 41.4 15.7 15.6 11.3 14 ...  
## $ EU\_Sales : num 28.96 12.76 10.93 9.14 9.18 ...  
## $ JP\_Sales : num 3.77 3.79 3.28 6.5 2.93 4.7 4.13 3.6 0.24 2.53 ...  
## $ Other\_Sales : num 8.45 3.29 2.95 2.88 2.84 2.24 1.9 2.15 1.69 1.77 ...  
## $ Global\_Sales : num 82.5 35.5 32.8 29.8 28.9 ...  
## $ Critic\_Score : num 76 82 80 89 58 87 91 80 61 80 ...  
## $ User\_Score : num 8 8.3 8 8.5 6.6 8.4 8.6 7.7 6.3 7.4 ...  
## $ Rating : chr "Everyone" "Everyone" "Everyone" "Everyone" ...  
## - attr(\*, "na.action")= 'omit' Named int 2 5 6 10 11 13 19 21 22 23 ...  
## ..- attr(\*, "names")= chr "2" "5" "6" "10" ...

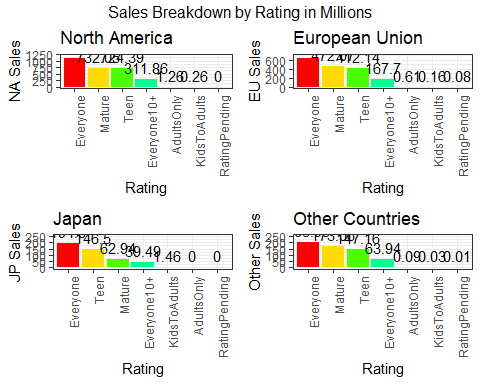
summary(GamesCleaned2)

## Year\_of\_Release Genre Publisher   
## Length:8054 Length:8054 Length:8054   
## Class :character Class :character Class :character   
## Mode :character Mode :character Mode :character   
##   
##   
##   
##   
## NA\_Sales EU\_Sales JP\_Sales Other\_Sales   
## Min. : 0.0000 Min. : 0.0000 Min. :0.00000 Min. : 0.00000   
## 1st Qu.: 0.0500 1st Qu.: 0.0100 1st Qu.:0.00000 1st Qu.: 0.01000   
## Median : 0.1300 Median : 0.0500 Median :0.00000 Median : 0.02000   
## Mean : 0.3562 Mean : 0.2097 Mean :0.05528 Mean : 0.07304   
## 3rd Qu.: 0.3500 3rd Qu.: 0.1800 3rd Qu.:0.01000 3rd Qu.: 0.06000   
## Max. :41.3600 Max. :28.9600 Max. :6.50000 Max. :10.57000   
##   
## Global\_Sales Critic\_Score User\_Score Rating   
## Min. : 0.0100 Min. :13.00 Min. :0.500 Length:8054   
## 1st Qu.: 0.1000 1st Qu.:60.00 1st Qu.:6.500 Class :character   
## Median : 0.2500 Median :71.00 Median :7.500 Mode :character   
## Mean : 0.6945 Mean :68.97 Mean :7.183   
## 3rd Qu.: 0.6600 3rd Qu.:79.00 3rd Qu.:8.200   
## Max. :82.5300 Max. :98.00 Max. :9.600   
## NA's :1107

GameCounts <- table(GamesCleaned2$Rating)  
xlim <- c(0, 8)  
ylim <- c(0, 1.1\*max(GameCounts))  
bp <- barplot(GameCounts, main="Total Distribution of Games by Ratings", xlab="Rating", ylab ="Number of Games", xlim=xlim , ylim=ylim, col=rainbow(7))  
text(bp, 0, round(GameCounts, 1),pos=3)



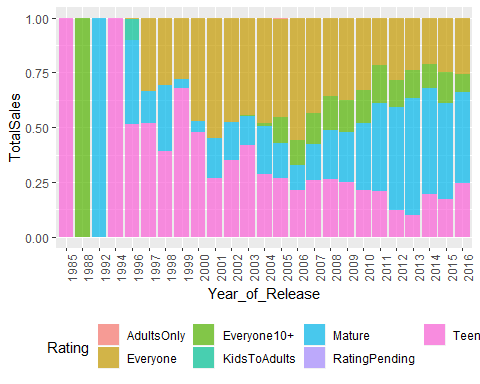
g1 <- GamesCleaned2 %>% group\_by(Rating) %>%  
 summarise(Sales = sum(NA\_Sales)) %>%  
 arrange(desc(Sales)) %>%  
 ggplot(aes(x = reorder(Rating, -Sales), y = Sales)) +   
 geom\_bar(stat = 'identity', fill = rainbow(n=7)) +  
 ylim(0, 1200) +  
 geom\_text(aes(label=Sales), position=position\_dodge(width=0.9), vjust=-0.5) +  
 xlab('Rating') +   
 ylab('NA Sales') +   
 ggtitle('North America') +   
 theme\_bw() +   
 theme(axis.text.x = element\_text(angle = 90, hjust = 1))  
g2 <- GamesCleaned2 %>% group\_by(Rating) %>%  
 summarise(Sales = sum(EU\_Sales)) %>%  
 arrange(desc(Sales)) %>%  
 ggplot(aes(x = reorder(Rating, -Sales), y = Sales)) +   
 geom\_bar(stat = 'identity', fill = rainbow(n=7)) +  
 ylim(0, 700) +  
 geom\_text(aes(label=Sales), position=position\_dodge(width=0.9), vjust=-0.5) +  
 xlab('Rating') +   
 ylab('EU Sales') +   
 ggtitle('European Union') +   
 theme\_bw() +   
 theme(axis.text.x = element\_text(angle = 90, hjust = 1))  
g3 <- GamesCleaned2 %>% group\_by(Rating) %>%  
 summarise(Sales = sum(JP\_Sales)) %>%  
 arrange(desc(Sales)) %>%  
 ggplot(aes(x = reorder(Rating, -Sales), y = Sales)) +   
 geom\_bar(stat = 'identity', fill = rainbow(n=7)) +  
 ylim(0, 250) +  
 geom\_text(aes(label=Sales), position=position\_dodge(width=0.9), vjust=-0.5) +  
 xlab('Rating') +   
 ylab('JP Sales') +   
 ggtitle('Japan') +   
 theme\_bw() +   
 theme(axis.text.x = element\_text(angle = 90, hjust = 1))  
g4 <- GamesCleaned2 %>% group\_by(Rating) %>%  
 summarise(Sales = sum(Other\_Sales)) %>%  
 arrange(desc(Sales)) %>%  
 ggplot(aes(x = reorder(Rating, -Sales), y = Sales)) +   
 geom\_bar(stat = 'identity', fill = rainbow(n=7)) +  
 ylim(0, 250) +  
 geom\_text(aes(label=Sales), position=position\_dodge(width=0.9), vjust=-0.5) +  
 xlab('Rating') +   
 ylab('Other Sales') +   
 ggtitle('Other Countries') +   
 theme\_bw() +   
 theme(axis.text.x = element\_text(angle = 90, hjust = 1))  
grid.arrange(g1,g2,g3,g4,nrow=2,ncol=2,top="Sales Breakdown by Rating in Millions")



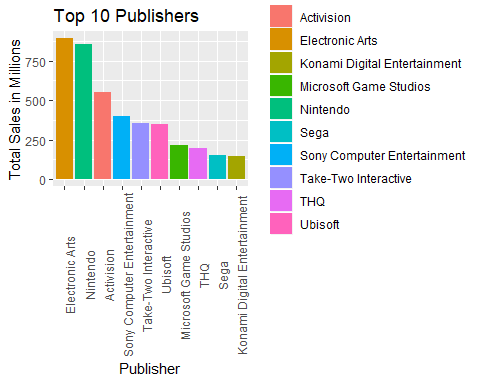
SumRatingSale <- aggregate(cbind(Global\_Sales,EU\_Sales,NA\_Sales,JP\_Sales,Other\_Sales)~Rating, data=GamesCleaned2, FUN=sum)  
SumRatingSale <- SumRatingSale[order(-SumRatingSale$Global\_Sales),]  
SumRatingSale

## Rating Global\_Sales EU\_Sales NA\_Sales JP\_Sales Other\_Sales  
## 2 Everyone 2134.62 636.19 1098.97 194.80 203.95  
## 5 Mature 1440.24 472.07 732.05 62.94 173.06  
## 7 Teen 1431.22 412.14 724.39 146.50 147.16  
## 3 Everyone10+ 583.37 167.70 311.86 39.49 63.94  
## 1 AdultsOnly 1.95 0.61 1.26 0.00 0.09  
## 4 KidsToAdults 1.92 0.16 0.26 1.46 0.03  
## 6 RatingPending 0.08 0.08 0.00 0.00 0.01

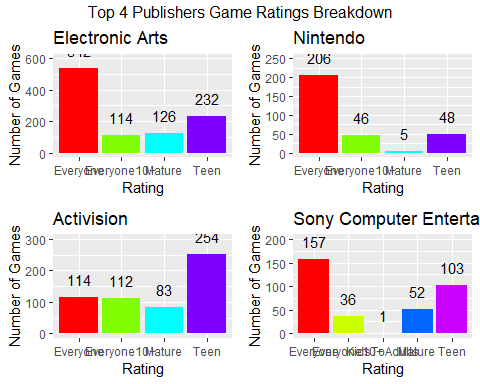
GamesCleaned3 <- filter(GamesCleaned2, Year\_of\_Release!='N/A')  
g5 <- GamesCleaned3 %>% select(Rating,Year\_of\_Release,Global\_Sales) %>%   
 group\_by(Rating,Year\_of\_Release) %>%  
 summarise(TotalSales=sum(Global\_Sales)) %>%  
 arrange(desc(TotalSales)) %>%  
 ggplot(aes(x=Year\_of\_Release,y=TotalSales,group=Rating,fill=Rating)) +   
 geom\_bar(stat="identity",position = "fill",alpha=0.7) +  
 theme(legend.position = "bottom",axis.text.x = element\_text(angle=90))  
  
g6 <- GamesCleaned3 %>% select(Publisher,Global\_Sales) %>%  
 group\_by(Publisher) %>%  
 summarise(Total\_Sales=sum(Global\_Sales)) %>%   
 arrange(desc(Total\_Sales)) %>%   
 head(10) %>%  
 ggplot(aes(x=factor(Publisher,level=Publisher),y=Total\_Sales,fill=Publisher))+geom\_bar(stat="identity") +  
 theme(axis.text.x=element\_text(angle=90)) +  
 labs(x="Publisher",y="Total Sales in Millions",title="Top 10 Publishers")  
g5



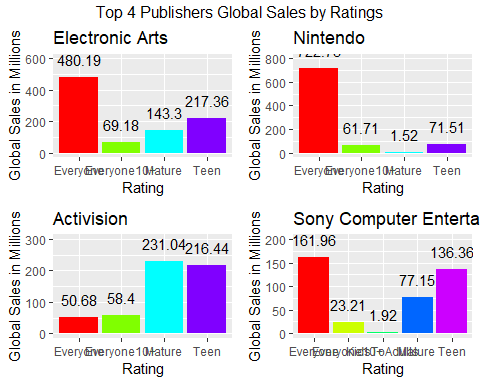
g6



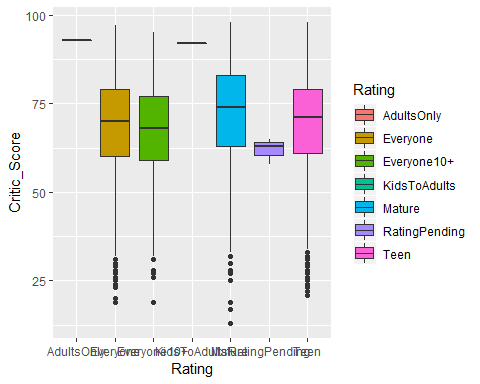
p1 <- GamesCleaned3 %>% subset(Publisher=='Electronic Arts') %>% group\_by(Rating) %>%  
 ggplot(aes(x = Rating)) +   
 geom\_bar(stat = 'count', fill = rainbow(n=4)) +  
 geom\_text(stat = 'count', aes(label=..count..), vjust=-1) +  
 ylim(0, 600) +  
 xlab('Rating') +   
 ylab('Number of Games') +   
 ggtitle('Electronic Arts')  
p2 <- GamesCleaned3 %>% subset(Publisher=='Nintendo') %>% group\_by(Rating) %>%  
 ggplot(aes(x = Rating)) +   
 geom\_bar(stat = 'count', fill = rainbow(n=4)) +  
 geom\_text(stat = 'count', aes(label=..count..), vjust=-1) +  
 ylim(0, 250) +  
 xlab('Rating') +   
 ylab('Number of Games') +   
 ggtitle('Nintendo')  
p3 <- GamesCleaned3 %>% subset(Publisher=='Activision') %>% group\_by(Rating) %>%  
 ggplot(aes(x = Rating)) +   
 geom\_bar(stat = 'count', fill = rainbow(n=4)) +  
 geom\_text(stat = 'count', aes(label=..count..), vjust=-1) +  
 ylim(0, 300) +  
 xlab('Rating') +   
 ylab('Number of Games') +   
 ggtitle('Activision')  
p4 <- GamesCleaned3 %>% subset(Publisher=='Sony Computer Entertainment') %>% group\_by(Rating) %>%  
 ggplot(aes(x = Rating)) +   
 geom\_bar(stat = 'count', fill = rainbow(n=5)) +  
 geom\_text(stat = 'count', aes(label=..count..), vjust=-1) +  
 ylim(0, 200) +  
 xlab('Rating') +   
 ylab('Number of Games') +   
 ggtitle('Sony Computer Entertainment')  
grid.arrange(p1,p2,p3,p4,nrow=2,ncol=2,top="Top 4 Publishers Game Ratings Breakdown")



p5 <- GamesCleaned2 %>% subset(Publisher=='Electronic Arts') %>% group\_by(Rating) %>%  
 summarise(Sales = sum(Global\_Sales)) %>%  
 ggplot(aes(x = Rating, y = Sales)) +   
 geom\_bar(stat = 'identity', fill = rainbow(n=4)) +  
 geom\_text(stat = 'identity', aes(label=Sales), vjust=-1) +  
 ylim(0, 600) +  
 xlab('Rating') +   
 ylab('Global Sales in Millions') +   
 ggtitle('Electronic Arts')  
p6 <- GamesCleaned2 %>% subset(Publisher=='Nintendo') %>% group\_by(Rating) %>%  
 summarise(Sales = sum(Global\_Sales)) %>%  
 ggplot(aes(x = Rating, y = Sales)) +   
 geom\_bar(stat = 'identity', fill = rainbow(n=4)) +  
 geom\_text(stat = 'identity', aes(label=Sales), vjust=-1) +  
 ylim(0, 800) +  
 xlab('Rating') +   
 ylab('Global Sales in Millions') +   
 ggtitle('Nintendo')  
p7 <- GamesCleaned2 %>% subset(Publisher=='Activision') %>% group\_by(Rating) %>%  
 summarise(Sales = sum(Global\_Sales)) %>%  
 ggplot(aes(x = Rating, y = Sales)) +   
 geom\_bar(stat = 'identity', fill = rainbow(n=4)) +  
 geom\_text(stat = 'identity', aes(label=Sales), vjust=-1) +  
 ylim(0, 300) +  
 xlab('Rating') +   
 ylab('Global Sales in Millions') +   
 ggtitle('Activision')  
p8 <- GamesCleaned2 %>% subset(Publisher=='Sony Computer Entertainment') %>% group\_by(Rating) %>%  
 summarise(Sales = sum(Global\_Sales)) %>%  
 ggplot(aes(x = Rating, y = Sales)) +   
 geom\_bar(stat = 'identity', fill = rainbow(n=5)) +  
 geom\_text(stat = 'identity', aes(label=Sales), vjust=-1) +  
 ylim(0, 200) +  
 xlab('Rating') +   
 ylab('Global Sales in Millions') +   
 ggtitle('Sony Computer Entertainment')  
grid.arrange(p5,p6,p7,p8,nrow=2,ncol=2,top="Top 4 Publishers Global Sales by Ratings")



g6 <- ggplot(GamesCleaned2, aes(x=Rating, y=Critic\_Score, fill=Rating)) +geom\_boxplot()  
g6



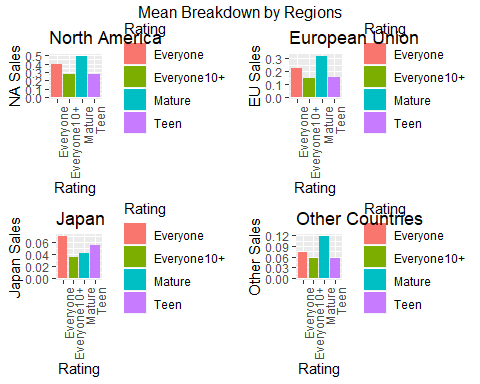
GamesCleaned4 <- filter(GamesCleaned2, Rating!='AdultsOnly' & Rating!='KidsToAdults' & Rating!='RatingPending')  
SumGenreSale <- aggregate(cbind(Global\_Sales,EU\_Sales,NA\_Sales,JP\_Sales,Other\_Sales)~Genre, data=GamesCleaned4, FUN=sum)  
SumGenreSale <- SumGenreSale[order(-SumGenreSale$Global\_Sales),]  
MeanRatingSale <- aggregate(cbind(Global\_Sales,EU\_Sales,NA\_Sales,JP\_Sales,Other\_Sales)~Rating, data=GamesCleaned4, FUN=mean)  
MeanRatingSale <- MeanRatingSale[order(-MeanRatingSale$Global\_Sales),]  
SumGenreSale

## Genre Global\_Sales EU\_Sales NA\_Sales JP\_Sales Other\_Sales  
## 1 Action 1269.34 406.72 631.55 77.80 152.57  
## 11 Sports 891.02 262.59 493.73 35.20 99.05  
## 9 Shooter 828.19 264.74 455.67 18.63 88.88  
## 7 Racing 505.47 173.20 243.68 28.05 60.56  
## 8 Role-Playing 504.60 119.78 221.08 123.28 40.61  
## 4 Misc 458.00 131.32 248.39 31.96 46.05  
## 5 Platform 395.51 113.36 205.36 43.31 33.33  
## 3 Fighting 255.07 61.97 139.61 27.88 25.51  
## 10 Simulation 214.45 69.82 98.44 28.10 18.13  
## 6 Puzzle 100.17 29.87 46.48 15.89 7.74  
## 2 Adventure 95.51 28.93 48.40 9.16 8.92  
## 12 Strategy 72.12 25.80 34.88 4.47 6.76

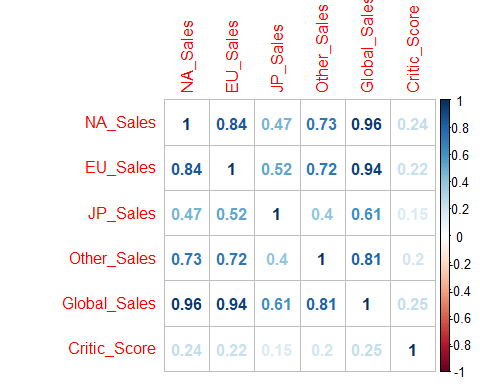
MeanRatingSale

## Rating Global\_Sales EU\_Sales NA\_Sales JP\_Sales Other\_Sales  
## 3 Mature 0.9711666 0.3183210 0.4936278 0.04244100 0.11669589  
## 1 Everyone 0.7601923 0.2265634 0.3913711 0.06937322 0.07263177  
## 4 Teen 0.5421288 0.1561136 0.2743902 0.05549242 0.05574242  
## 2 Everyone10+ 0.5217979 0.1500000 0.2789445 0.03532200 0.05719141

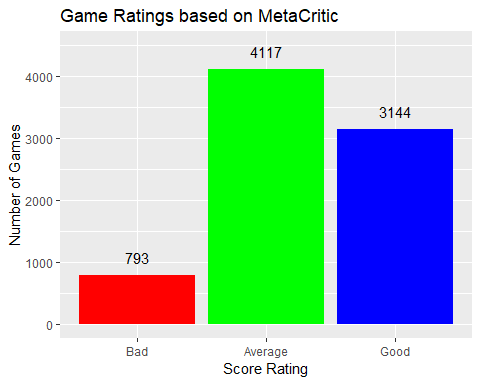
g7 <- MeanRatingSale %>% select(Rating,NA\_Sales) %>% group\_by(Rating) %>%  
 ggplot(aes(x=Rating, y=NA\_Sales, fill=Rating)) +   
 geom\_bar(stat="identity") +  
 xlab('Rating') +   
 ylab('NA Sales') +   
 ggtitle('North America') +   
 theme(axis.text.x = element\_text(angle = 90, hjust = 1))  
g8 <- MeanRatingSale %>% select(Rating,EU\_Sales) %>% group\_by(Rating) %>%  
 ggplot(aes(x=Rating, y=EU\_Sales, fill=Rating)) +   
 geom\_bar(stat="identity") +  
 xlab('Rating') +   
 ylab('EU Sales') +   
 ggtitle('European Union') +   
 theme(axis.text.x = element\_text(angle = 90, hjust = 1))  
g9 <- MeanRatingSale %>% select(Rating,JP\_Sales) %>% group\_by(Rating) %>%  
 ggplot(aes(x=Rating, y=JP\_Sales, fill=Rating)) +   
 geom\_bar(stat="identity") +  
 xlab('Rating') +   
 ylab('Japan Sales') +   
 ggtitle('Japan') +   
 theme(axis.text.x = element\_text(angle = 90, hjust = 1))  
g10 <- MeanRatingSale %>% select(Rating,Other\_Sales) %>% group\_by(Rating) %>%  
 ggplot(aes(x=Rating, y=Other\_Sales, fill=Rating)) +   
 geom\_bar(stat="identity") +  
 xlab('Rating') +   
 ylab('Other Sales') +  
 ggtitle('Other Countries') +   
 theme(axis.text.x = element\_text(angle = 90, hjust = 1))  
grid.arrange(g7,g8,g9,g10,nrow=2,ncol=2,top="Mean Breakdown by Regions")



CorrData <- cor(GamesCleaned2[, 4:9])  
corrplot(CorrData, method ="number")

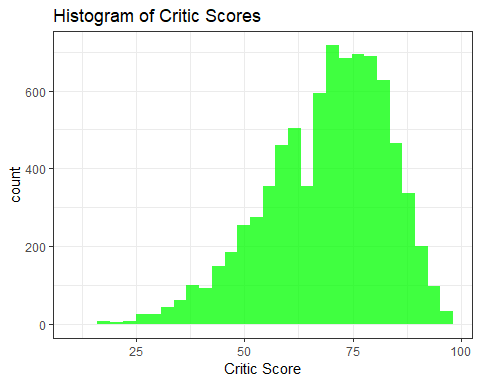


GamesCleaned2$ScoreRating <- ifelse(GamesCleaned2$Critic\_Score >= 75, 'Good',  
 ifelse(GamesCleaned2$Critic\_Score >=50 & GamesCleaned2$Critic\_Score < 75, 'Average',  
 ifelse(GamesCleaned2$Critic\_Score < 50, 'Bad', 'N/A')))  
g10 <- GamesCleaned2 %>% group\_by(ScoreRating) %>%  
 ggplot(aes(factor(ScoreRating, levels = c('Bad', 'Average', 'Good')))) +   
 geom\_bar(stat = 'count', fill = rainbow(n=3)) +  
 geom\_text(stat = 'count', aes(label=..count..), vjust=-1) +  
 ylim(0, 4500) +  
 xlab('Score Rating') +   
 ylab('Number of Games') +   
 ggtitle('Game Ratings based on MetaCritic')  
g10

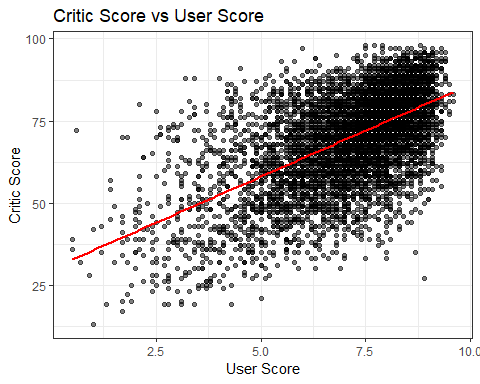


GamesCleaned2a <- filter(GamesCleaned2, Rating!='AdultsOnly' & Rating!='KidsToAdults' & Rating!='RatingPending')  
g10 <- GamesCleaned2 %>%   
 ggplot() +   
 geom\_histogram(aes(x = Critic\_Score), fill = 'green', alpha = 0.75, show.legend = T) +   
 ggtitle('Histogram of Critic Scores') + xlab('Critic Score') +  
 theme\_bw()  
g11 <- GamesCleaned2 %>% subset(Critic\_Score & User\_Score) %>%  
 ggplot(aes(x = as.numeric(User\_Score), y = as.numeric(Critic\_Score))) +   
 geom\_point(alpha = 0.5) +   
 stat\_smooth(method = 'lm', col = 'red', se = FALSE) +   
 xlab('User Score') + ylab('Critic Score') + ggtitle('Critic Score vs User Score') +   
 theme\_bw()  
g12 <- GamesCleaned2a %>%   
 ggplot(aes(x=User\_Score,y=Critic\_Score)) +  
 geom\_point(aes(color=Rating),size=2,alpha=.5) +   
 ggtitle('Breakdown of Ratings') + xlab('User Score') +  
 ylim(0,100) +  
 geom\_smooth(method = "lm", size=.5,color="black", formula = y ~ x) + facet\_wrap(~Rating)  
g10

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



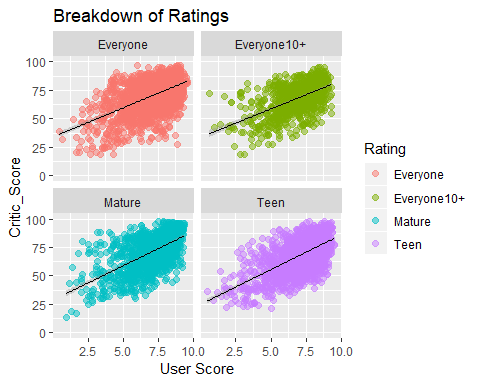
g11



g12

## Warning: Removed 1106 rows containing non-finite values (stat\_smooth).

## Warning: Removed 1106 rows containing missing values (geom\_point).



GamesCleaned2b <- filter(GamesCleaned2, !is.na(User\_Score))  
  
cor.test(GamesCleaned2$Critic\_Score, GamesCleaned2$Global\_Sales)

##   
## Pearson's product-moment correlation  
##   
## data: GamesCleaned2$Critic\_Score and GamesCleaned2$Global\_Sales  
## t = 22.78, df = 8052, p-value < 2.2e-16  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.2254342 0.2664708  
## sample estimates:  
## cor   
## 0.2460628

cor.test(GamesCleaned2b$User\_Score, GamesCleaned2b$Global\_Sales)

##   
## Pearson's product-moment correlation  
##   
## data: GamesCleaned2b$User\_Score and GamesCleaned2b$Global\_Sales  
## t = 7.41, df = 6945, p-value = 1.412e-13  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.06518734 0.11185058  
## sample estimates:  
## cor   
## 0.08856755

CriticScore\_Sales <- lm(data = GamesCleaned2, Global\_Sales ~ Critic\_Score)  
summary(CriticScore\_Sales)

##   
## Call:  
## lm(formula = Global\_Sales ~ Critic\_Score, data = GamesCleaned2)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.460 -0.618 -0.286 0.156 81.609   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -1.525499 0.099425 -15.34 <2e-16 \*\*\*  
## Critic\_Score 0.032187 0.001413 22.78 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 1.769 on 8052 degrees of freedom  
## Multiple R-squared: 0.06055, Adjusted R-squared: 0.06043   
## F-statistic: 518.9 on 1 and 8052 DF, p-value: < 2.2e-16

UserScore\_Sales <- lm(data = GamesCleaned2b, Global\_Sales ~ User\_Score)  
summary(UserScore\_Sales)

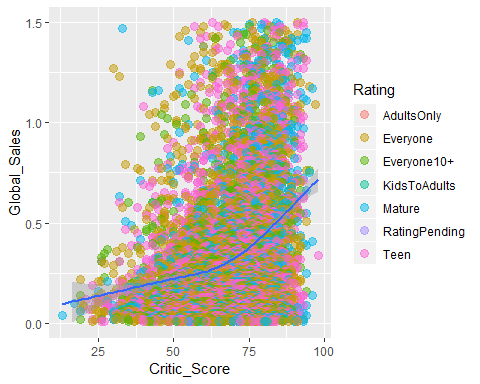
##   
## Call:  
## lm(formula = Global\_Sales ~ User\_Score, data = GamesCleaned2b)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.040 -0.657 -0.415 -0.005 81.659   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -0.08675 0.11835 -0.733 0.464   
## User\_Score 0.11970 0.01615 7.410 1.41e-13 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 1.941 on 6945 degrees of freedom  
## Multiple R-squared: 0.007844, Adjusted R-squared: 0.007701   
## F-statistic: 54.91 on 1 and 6945 DF, p-value: 1.412e-13

#Removing Outliers  
remove\_outliers <- function(x, na.rm = TRUE, ...) {  
 qnt <- quantile(x, probs=c(.25, .75), na.rm = na.rm, ...)  
 H <- 1.5 \* IQR(x, na.rm = na.rm)  
 y <- x  
 y[x < (qnt[1] - H)] <- NA  
 y[x > (qnt[2] + H)] <- NA  
 y  
}  
  
GamesClean\_Outliers <- GamesCleaned2 %>%  
 mutate(Global\_Sales = remove\_outliers(Global\_Sales))  
  
g13 <- GamesClean\_Outliers %>%  
 ggplot(aes(x=Critic\_Score,y=Global\_Sales)) +   
 geom\_point(aes(color=Rating),size=3,alpha=.5) +   
 ylim(0,1.5) +   
 geom\_smooth()  
g14 <- GamesClean\_Outliers %>%  
 ggplot(aes(x=User\_Score,y=Global\_Sales)) +   
 geom\_point(aes(color=Rating),size=3,alpha=.5) +   
 ylim(0,1.5) +   
 geom\_smooth()  
g13

## `geom\_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'

## Warning: Removed 831 rows containing non-finite values (stat\_smooth).

## Warning: Removed 831 rows containing missing values (geom\_point).



g14

## `geom\_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'

## Warning: Removed 1926 rows containing non-finite values (stat\_smooth).

## Warning: Removed 1926 rows containing missing values (geom\_point).

