

Homework 10

Group 8

November 19, 2019

R Markdown

#Ignore this all for now, it's just a bunch of operations

```
# data=read.csv('PlayersGroupedByMonth.csv')
# par(mfrow=c(2,2))
# plot(data$CollegeGames,data$Minutes)
# plot(data$CollegeGames,data$Pts36,ylim=c(0,30))
# plot(data$CollegeGames,data$Ast36)
# plot(data$CollegeGames,data$TO36)
#
# bin=rep(0,length(data$br_url))
# #for (i in 1:length(bin)){
#   if (data$College_Games[i] <= 35 && data$College_Games[i] >= 1) {
#     bin[i]=1
#   } else if (data$College_Games[i] >= 36 && data$College_Games[i] <= 70) {
#     bin[i]=2
#   } else if (data$College_Games[i] >= 71 && data$College_Games[i] <= 105) {
#     bin[i]=3
#   } else if (data$College_Games[i] >= 106 && data$College_Games[i] <= 152) {
#     bin[i]=4
#   } else if (data$Intl_Minutes[i] <= 669 && data$Intl_Minutes[i] >= 2) {
#     bin[i]=5
#   } else if (data$Intl_Minutes[i] >= 670 && data$Intl_Minutes[i] <= 1221) {
#     bin[i]=6
#   } else if (data$Intl_Minutes[i] >= 1222 && data$Intl_Minutes[i] <= 3328) {
#     bin[i]=7
#   } else if (data$Intl_Minutes[i] >= 3329 && data$Intl_Minutes[i] <= 13659) {
#     bin[i]=8
#   }
# }
# data$bin=bin
# data$minsgroup<-NULL
# write.csv(data, 'PlayersGroupedByMonth2.csv')
#
#
# par(mfrow=c(2,2))
# plot(data$Month,data$Minutes,col=data$MinBin)
# plot(data$Month,data$Pts36,col=data$MinBin)
# plot(data$Month,data$Ast36,col=data$MinBin)
# plot(data$Month,data$TO36,col=data$MinBin)
#
# data$month<-factor(data$Month,levels=c(10,11,12,1,2,3,4,5,6))
```

Here is the good stuff!

```

library(ggplot2)
library(ggpubr)

## Loading required package: magrittr

library(grid)
library(gridBase)
data=read.csv('PlayersGroupedByMonthAndBin.csv')
cols <- c("0"="black", '1'='greenyellow', '2'='forestgreen', '3'='dodgerblue',
          '4'='darkorchid1', '5'='yellow', '6'='orange', '7'='darksalmon', '8'='firebrick3')
data$Month<-factor(data$Month,levels=c(10,11,12,1,2,3,4,5,6))
Game_Score36=data$Game_Score/data$Minutes*36
data$Game_Score36=Game_Score36

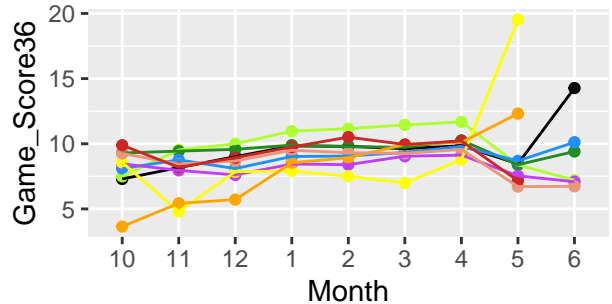
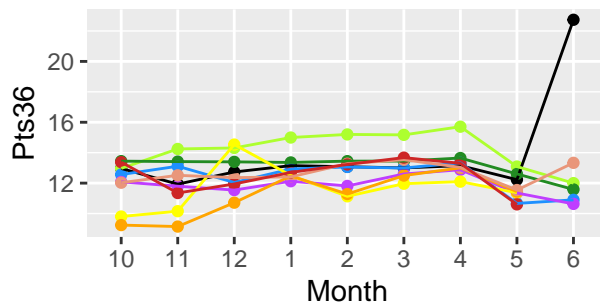
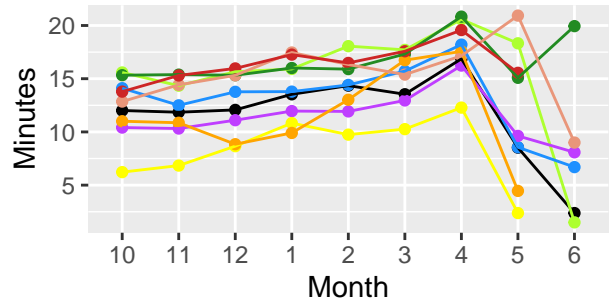
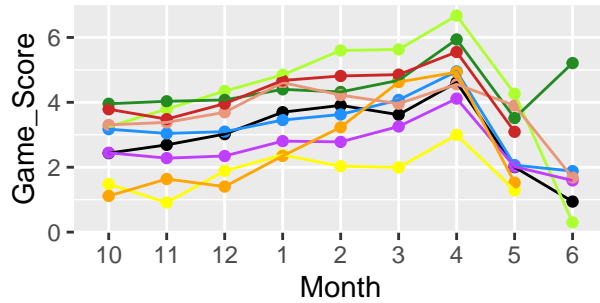
Game_Score_Plot=ggplot(data, aes(x = Month, y = Game_Score, group = Bin, color = 'Bin')) +
  geom_point(aes(color = factor(Bin))) +
  scale_color_manual(values = cols, labels=c('HS/Unknown Intl', 'Col 1-35 Gms', 'Col 36-70 Gms',
                                             'Col 71-105 Gms', 'Col 106+ Gms', 'Intl 1-669 Mins',
                                             'Intl 670-1221 Mins', 'Intl 1222-3328 Mins',
                                             'Intl 3329+ Mins')) +
  geom_line(aes(color = factor(Bin)))
Minutes_Plot=ggplot(data, aes(x = Month, y = Minutes, group = Bin, color = 'Bin')) +
  geom_point(aes(color = factor(Bin))) +
  scale_color_manual(values = cols, labels=c('HS/Unknown Intl', 'Col 1-35 Gms', 'Col 36-70 Gms',
                                             'Col 71-105 Gms', 'Col 106+ Gms', 'Intl 1-669 Mins',
                                             'Intl 670-1221 Mins', 'Intl 1222-3328 Mins',
                                             'Intl 3329+ Mins')) +
  geom_line(aes(color = factor(Bin)))
Pts36_Plot=ggplot(data, aes(x = Month, y = Pts36, group = Bin, color = 'Bin')) +
  geom_point(aes(color = factor(Bin))) +
  scale_color_manual(values = cols, labels=c('HS/Unknown Intl', 'Col 1-35 Gms', 'Col 36-70 Gms',
                                             'Col 71-105 Gms', 'Col 106+ Gms', 'Intl 1-669 Mins',
                                             'Intl 670-1221 Mins', 'Intl 1222-3328 Mins',
                                             'Intl 3329+ Mins')) +
  geom_line(aes(color = factor(Bin)))
Game_Score_36_Plot=ggplot(data, aes(x = Month, y = Game_Score36, group = Bin, color = 'Bin')) +
  geom_point(aes(color = factor(Bin))) +
  scale_color_manual(values = cols, labels=c('HS/Unknown Intl', 'Col 1-35 Gms', 'Col 36-70 Gms',
                                             'Col 71-105 Gms', 'Col 106+ Gms', 'Intl 1-669 Mins',
                                             'Intl 670-1221 Mins', 'Intl 1222-3328 Mins',
                                             'Intl 3329+ Mins')) +
  geom_line(aes(color = factor(Bin)))
plot=ggarrange(Game_Score_Plot, Minutes_Plot, Pts36_Plot, Game_Score_36_Plot,
               ncol=2, nrow=2, common.legend=TRUE)
annotate_figure(plot, top=text_grob('Game Score, Minutes, Pts Per 36 Mins,
                                   and Game Score Per 36 Mins Plots by Month'))

```

Game Score, Minutes, Pts Per 36 Mins, and Game Score Per 36 Mins Plots by Month

ur

● HS/Unknown Intl	● Col 36–70 Gms	● Col 106+ Gms	● Intl 670–1221 Mins	● Intl 3329+
● Col 1–35 Gms	● Col 71–105 Gms	● Intl 1–669 Mins	● Intl 1222–3328 Mins	



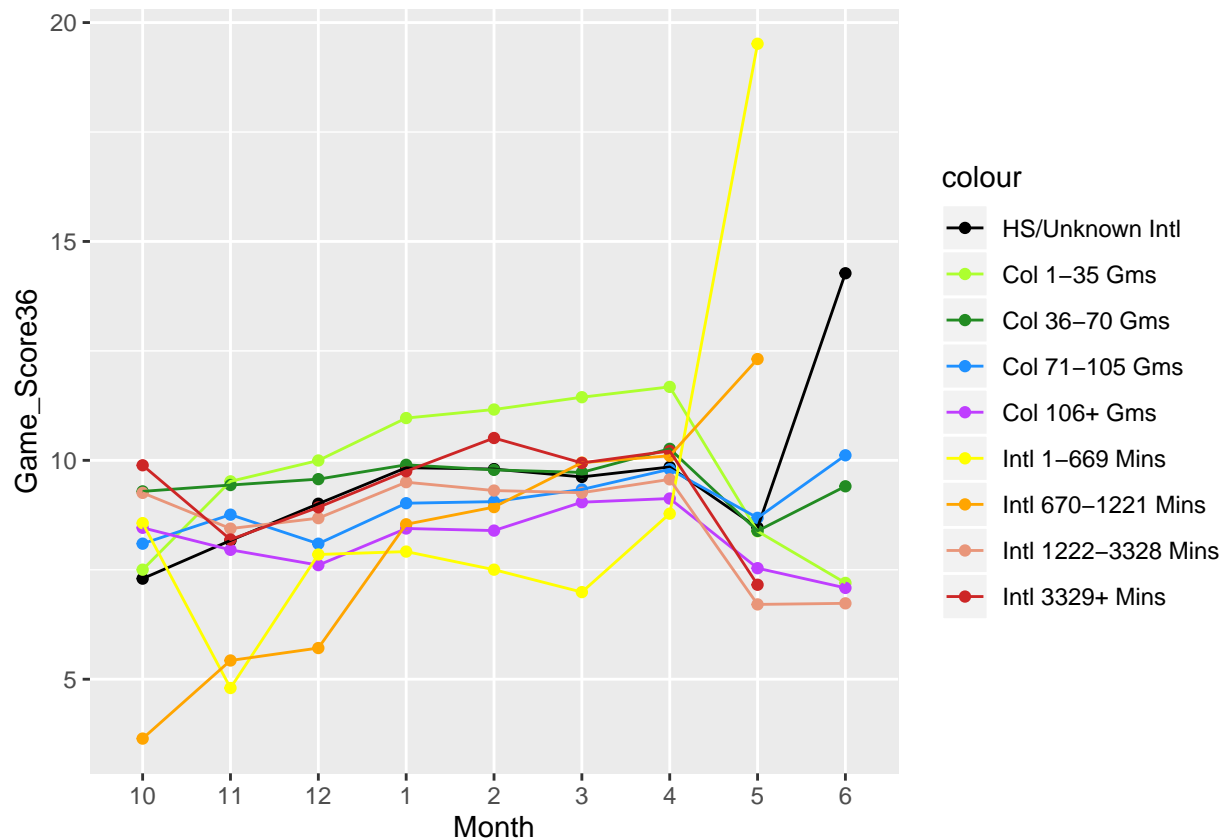
```
#vp1=viewport(x=0,y=0,width=0.5,height=0.5,just=c('left','bottom'))
#vp2=viewport(x=0,y=0.5,width=0.5,height=0.5,just=c('left','bottom'))
#vp3=viewport(x=0.5,y=0,width=0.5,height=0.5,just=c('left','bottom'))
#vp4=viewport(x=0,y=0.5,width=0.5,height=0.5,just=c('left','bottom'))
#print(Game_Score_Plot,vp=vp1)
#print(Minutes_Plot,vp=vp2)
#print(Pts36_Plot,vp=vp3)
#print(Game_Score_36_Plot,vp=vp4)
```

```
# Function
plot_func <- function(plot_name) {
  if (plot_name == "Game_Score") {
    y = data$Game_Score
  } else if (plot_name == "Minutes") {
    y = data$Minutes
  } else if (plot_name == "Pts36") {
    y = data$Pts36
  } else if (plot_name == "Game_Score36") {
    y = data$Game_Score36
  } else {
    print("Plot name not found")
    return()
  }
  ggplot(data, aes(x = Month, y = y, group = Bin, color = 'Bin')) +
    geom_point(aes(color = factor(Bin))) +
    scale_color_manual(values = cols, labels=c('HS/Unknown Intl','Col 1-35 Gms','Col 36-70 Gms',
      'Col 71-105 Gms','Col 106+ Gms','Intl 1-669 Mins',
      'Intl 670-1221 Mins','Intl 1222-3328 Mins','Intl 3329+'))
}
```

```

    'Intl 670-1221 Mins', 'Intl 1222-3328 Mins',
    'Intl 3329+ Mins')) +
  geom_line(aes(color = factor(Bin))) +
  labs(y = plot_name)
}
plot_func("Game_Score36")

```



```

# Multiple plot function
#
# ggplot objects can be passed in ..., or to plotlist (as a list of ggplot objects)
# - cols:   Number of columns in layout
# - layout: A matrix specifying the layout. If present, 'cols' is ignored.
#
# If the layout is something like matrix(c(1,2,3,3), nrow=2, byrow=TRUE),
# then plot 1 will go in the upper left, 2 will go in the upper right, and
# 3 will go all the way across the bottom.
#
multiplot <- function(..., plotlist=NULL, file, cols=1, layout=NULL) {
  library(grid)

  # Make a list from the ... arguments and plotlist
  plots <- c(list(...), plotlist)

  numPlots = length(plots)

  # If layout is NULL, then use 'cols' to determine layout
  if (is.null(layout)) {

```

```

# Make the panel
# ncol: Number of columns of plots
# nrow: Number of rows needed, calculated from # of cols
layout <- matrix(seq(1, cols * ceiling(numPlots/cols)),
                  ncol = cols, nrow = ceiling(numPlots/cols))
}

if (numPlots==1) {
  print(plots[[1]])
} else {
  # Set up the page
  grid.newpage()
  pushViewport(viewport(layout = grid.layout(nrow(layout), ncol(layout))))

  # Make each plot, in the correct location
  for (i in 1:numPlots) {
    # Get the i,j matrix positions of the regions that contain this subplot
    matchidx <- as.data.frame(which(layout == i, arr.ind = TRUE))

    print(plots[[i]], vp = viewport(layout.pos.row = matchidx$row,
                                     layout.pos.col = matchidx$col))
  }
}
}

```

Month	Oct.	Nov.	Dec.	Jan.	Feb.	Mar.	Apr.	May	June
HS Draft pos. 1-14 15-30 31-44 45-60									
College < 35 Gm									
Coll. 35 < x < 70 Gm									
Coll. 70 < x < 105									
Coll. 105 < x < 140									
wt. < 670 m.									
wt. 670-1221									
wt. 1222-3328									
wt. 3329+									
Week	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4					

Game Score

10

0