Preliminary Thoughts

```
#import csvs
season18_19 <- read.csv("hackathon-business-track/data/hackathon_2018_19_attendance.csv")
season21_22 <- read.csv("hackathon-business-track/data/hackathon_2021_22_attendance.csv")
#combine seasons
seasonboth <- rbind(season18_19, season21_22)
```

Some Summarization

```
# group by tier
seasonByTier <- seasonboth %>% group_by(Game.Tier)

# take away rows with promo
seasonByTierWithoutPromotion <- subset(seasonByTier, seasonByTier$Promo == "")

# take the summary
summaryOfNoPromo <- seasonByTierWithoutPromotion %>% summarise(
  tip = mean(Attendance.at.tip),
  total = mean(Total.Attendance)
)

# take only rows with promo
seasonByTierWithPromotion <- subset(seasonByTier, seasonByTier$Promo != "")

# take the summary
summaryOfWithPromo <- seasonByTierWithPromotion %>% summarise(
  tip = mean(Attendance.at.tip),
  total = mean(Total.Attendance)
)

# merge total summary
totalSummary <- merge(summaryOfNoPromo, summaryOfWithPromo, by="Game.Tier")
totalSummary <- rename(totalSummary, "tipWithout" = "tip.x", "totalWithout" = "total.x",
                       "tipWith" = "tip.y", "totalWith" = "total.y")

# create a final summary with percent changes
finalSummary <- totalSummary %>%
  mutate(PercTip = (tipWith/tipWithout - 1)*100) %>%
  mutate(PercTotal = (totalWith/totalWithout - 1)*100)

# see results
names(finalSummary) <- c('Game Tier', 'At Tip/No Promo', 'Total/No Promo',
                         'At Tip/Promo', 'Total/Promo', '% Diff at Tip',
                         '% Diff of Total')
finalSummary #%>%
```

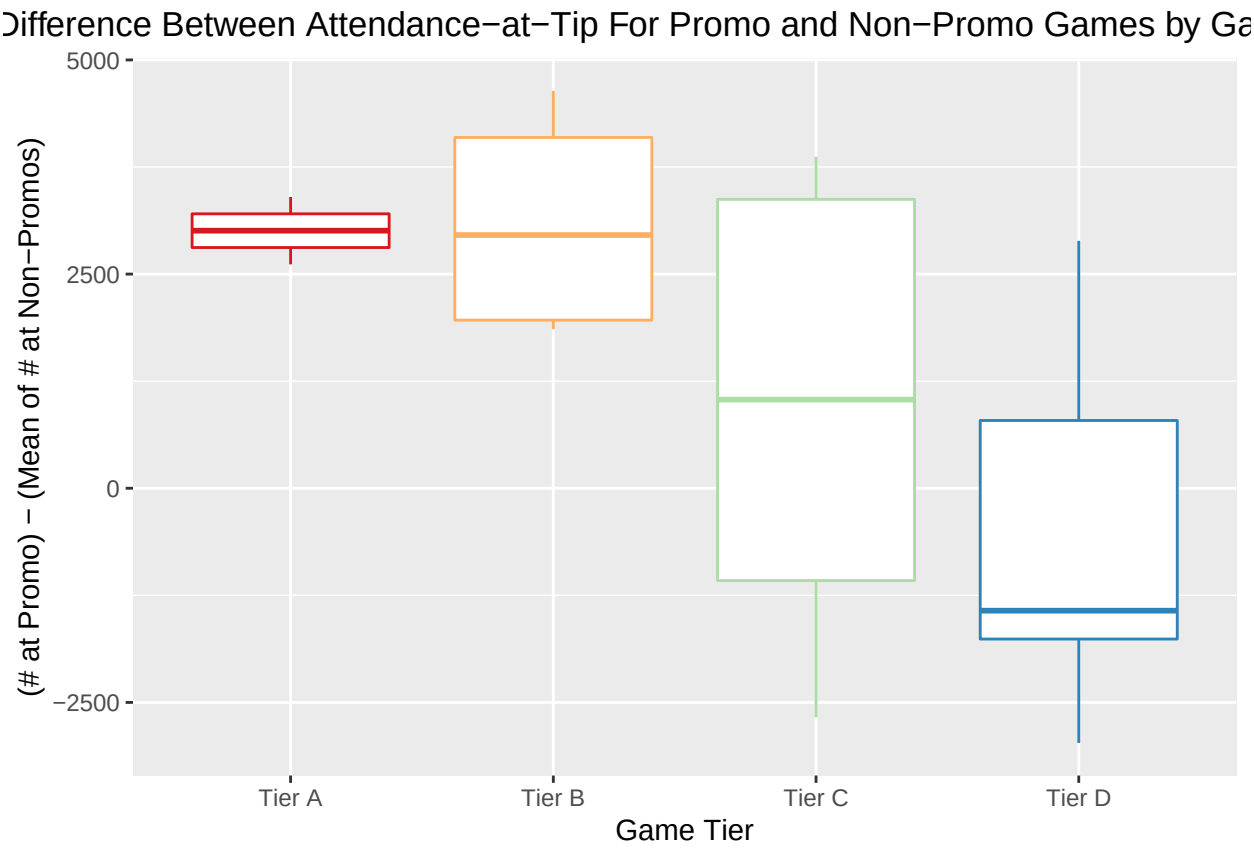
```
##   Game Tier At Tip/No Promo Total/No Promo At Tip/Promo Total/Promo
## 1    Tier A        11520.60       14508.40     14527.50    14972.00
## 2    Tier B        12060.67       15408.33     15162.75    16466.50
## 3    Tier C        12335.13       14371.26     13309.43    15435.71
## 4    Tier D        11517.04       14326.17     10828.56    14026.11
##   % Diff at Tip % Diff of Total
## 1     26.100203        3.195390
## 2     25.720662        6.867496
## 3      7.898564        7.406820
## 4     -5.977992       -2.094508
```

```
  #gt() %>%
  #gt_theme_nytimes() %>%
  #tab_header(title = "Summary Statistics of Games With or Without Promotions")
#commented out to get the table aesthetics, but table on slides has new format
```

We can see that generally, for higher tier games, there is a much higher amount of people at tip-off (26% and 25% increase for tier A and B respectively). However, there are marginal increases in total attendances for those two tiers (3% and 7% increase), while for tier C, there is an increase of around 7-8% for both tip and total attendance. For Tier D games, we actually saw a decrease in attendance at tip and total, of around -6 and -2 percent respectively. So perhaps we can argue that promos should not be applied towards games that are already decided to be Tier D.

```
#get means by tier of games of attendance at tip off
getMeanOfNoPromo <- seasonByTier %>%
  mutate(promoGame = ifelse(Promo == "", 0, 1)) %>%
  filter(promoGame == 0) %>%
  group_by(Game.Tier) %>%
  mutate(meanOfTier = mean(Attendance.at.tip))

#delete extra columns
getMeanOfNoPromo <- getMeanOfNoPromo[,c(4,10)]

# group by tier and add the mean of the no-promo games
seasonByTierNew <- seasonByTier %>%
  mutate(promoGame = ifelse(Promo == "", 0, 1)) %>%
  left_join(getMeanOfNoPromo, by = "Game.Tier") %>%
  distinct()

# variance in promos
seasonByTierNew <- seasonByTierNew %>%
  mutate(diffPromo = Attendance.at.tip - meanOfTier) %>%
  mutate(variance = var(diffPromo))

# get only promo games for our boxplot
seasonByTierBoxplot <- seasonByTierNew[,-c(7:8)] %>%
  filter(promoGame == 1)

# plot our boxplot and set our aesthetics
p <- ggplot(seasonByTierBoxplot, aes(x=Game.Tier, y=diffPromo, color = Game.Tier)) +
  geom_boxplot()
p + scale_color_brewer(palette="Spectral") +
  theme(legend.position="none") +
```

```
  labs(title = "Difference Between Attendance-at-Tip For Promo and Non-Promo Games by Game Tier", y = "
  theme(plot.title = element_text(hjust = 0.5)) #+
```




```
  #theme(text=element_text(family="Times New Roman"))
# would not knit without commenting out font above, but image has times new roman

# save the boxplot as an image
# ggsave("attendanceComparison.png", width = 7.25, height = 3.5)
```

Boxplot is cool and reinforces what I had thought from the summary table above.