Lineup Analysis Formation

Max Sleek

2023-01-15

Packages and Stuff

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.4.0      v purrr   1.0.0
## v tibble  3.1.8      v dplyr   1.0.10
## v tidyr   1.2.1      v stringr 1.5.0
## v readr   2.1.3      v forcats 0.5.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(rvest)
```

```
##
## Attaching package: 'rvest'
##
## The following object is masked from 'package:readr':
##
##     guess_encoding
```

Scrape Regular Season Lineups From 2004-2022 From Cleaning The Glass

The easiest way to scrape the data from Cleaning The Glass was to download it as individual csv files. A csv was available for all offensive and defensive statistics for each lineup. The code below is a sample of how I created/merged these data frames, I did not include the other 17 years in the knit due to file length. Note: I had to rename some of the variables to denote offense or defense; sorry for any clutter caused by that. Note 2: 'Rank' actually means percentile.

```
#2022
# Four Factors
fourfactors = as.data.frame(read_csv("lineups_four_factors_22.csv", show_col_types = FALSE)) %>%
  mutate("Year" = 2022)
# Offensive Shot Frequency
```

```
shotfreq = as.data.frame(read_csv("lineups_offense_shooting_frequency_22.csv", show_col_types = FALSE))
  select(10:23) %>%
  rename("OFFENSE: Rim Frequency Rank" = 1, "OFFENSE: Rim Frequency" = 2,"OFFENSE: Short Mid Frequency
# Offensive Shot Accuracy
shotacc = as.data.frame(read_csv("lineups_offense_shooting_accuracy_22.csv", show_col_types = FALSE)) %
  select(10:23) %>%
  rename("OFFENSE: Rim FG% Rank" = 1, "OFFENSE: Rim FG%" = 2,"OFFENSE: Short Mid FG Rank" = 3, "OFFENSE
# Offensive Halfcourt
half = as.data.frame(read_csv("lineups_offense_halfcourt_and_putbacks_22.csv", show_col_types = FALSE))
  select(10:21)
# Offensive Transition
trans = as.data.frame(read_csv("lineups_offense_transition_22.csv", show_col_types = FALSE)) %>%
  select(10:27) %>%
  rename("OFFENSE TRANSITION: Pts+/Poss Rank" = 1, "OFFENSE TRANSITION: Pts+/Poss" = 2, "OFFENSE TRANSI
# Defensive Shot Frequency
defshotfreq = as.data.frame(read_csv("lineups_defense_shooting_frequency_22.csv", show_col_types = FALS
  select(10:23) %>%
  rename("DEFENSE: Rim Frequency Rank" = 1, "DEFENSE: Rim Frequency" = 2,"DEFENSE: Short Mid Frequency
# Defensive Shot Accuracy
defshotacc = as.data.frame(read_csv("lineups_defense_shooting_accuracy_22.csv", show_col_types = FALSE)
  select(10:23) %>%
  rename("DEFENSE: Rim FG% Rank" = 1, "DEFENSE: Rim FG%" = 2,"DEFENSE: Short Mid FG Rank" = 3, "DEFENSE
# Defensive Halfcourt
defhalf = as.data.frame(read_csv("lineups_defense_halfcourt_and_putbacks_22.csv", show_col_types = FALS
  select(10:21) %>%
  rename("DEFENSE HALFCOURT: Pts/Play Rank" = 1,"DEFENSE HALFCOURT: Pts/Play" =2, "DEFENSE HALFCOURT: O
# Defensive Transition
deftrans = as.data.frame(read_csv("lineups_defense_transition_22.csv", show_col_types = FALSE)) %>%
  select(10:27) %>%
  rename("DEFENSE TRANSITION: Pts+/Poss Rank" = 1, "DEFENSE TRANSITION: Pts+/Poss" = 2, "DEFENSE TRANSI

# Create Data Frame
reg_season_lineups22 = cbind.data.frame(fourfactors, shotfreq, shotacc, half, trans, defshotfreq, defsh
# Clear Environment
rm(fourfactors, shotfreq, shotacc, half, trans, defshotfreq, defshotacc, defhalf, deftrans)
```

Combine Regular Season Lineup Data Frames, Clear Environment

```
all_reg_lineups <- rbind.data.frame(reg_season_lineups04, reg_season_lineups05, reg_season_lineups06, r

vec1 <- c("Regular Season")
vec1 <- rep(vec1, nrow(all_reg_lineups))
all_reg_lineups <- mutate(all_reg_lineups, "Regular Season or Playoffs" = vec1)
all_reg_lineups <- relocate(all_reg_lineups, "Regular Season or Playoffs")

rm(reg_season_lineups04, reg_season_lineups05, reg_season_lineups06, reg_season_lineups07, reg_season_l
```

Scrape Playoff Lineups From 2004-2022 From Cleaning The Glass

Same as above, only including the first year for reference.

```
#2022
fourfactors = as.data.frame(read_csv("lineups_four_factors_playoffs_22.csv", show_col_types = FALSE)) %
  mutate("Year" = 2022)
shotfreq = as.data.frame(read_csv("lineups_offense_shooting_frequency_playoffs_22.csv", show_col_types =
  select(10:23) %>%
  rename("OFFENSE: Rim Frequency Rank" = 1, "OFFENSE: Rim Frequency" = 2,"OFFENSE: Short Mid Frequency
shotacc = as.data.frame(read_csv("lineups_offense_shooting_accuracy_playoffs_22.csv", show_col_types =
  select(10:23) %>%
  rename("OFFENSE: Rim FG% Rank" = 1, "OFFENSE: Rim FG%" = 2,"OFFENSE: Short Mid FG Rank" = 3, "OFFENSE
half = as.data.frame(read_csv("lineups_offense_halfcourt_and_putbacks_playoffs_22.csv", show_col_types =
  select(10:21)
defshotfreq = as.data.frame(read_csv("lineups_defense_shooting_frequency_playoffs_22.csv", show_col_type
  select(10:23) %>%
  rename("DEFENSE: Rim Frequency Rank" = 1, "DEFENSE: Rim Frequency" = 2,"DEFENSE: Short Mid Frequency
defshotacc = as.data.frame(read_csv("lineups_defense_shooting_accuracy_playoffs_22.csv", show_col_types
  select(10:23) %>%
  rename("DEFENSE: Rim FG% Rank" = 1, "DEFENSE: Rim FG%" = 2,"DEFENSE: Short Mid FG Rank" = 3, "DEFENSE
defhalf = as.data.frame(read_csv("lineups_defense_halfcourt_and_putbacks_playoffs_22.csv", show_col_type
  select(10:21) %>%
  rename("DEFENSE HALFCOURT: Pts/Play Rank" = 1,"DEFENSE HALFCOURT: Pts/Play" =2, "DEFENSE HALFCOURT: O
trans = as.data.frame(read_csv("lineups_offense_transition_playoffs_22.csv", show_col_types = FALSE)) %
  select(10:27) %>%
  rename("OFFENSE TRANSITION: Pts+/Poss Rank" = 1, "OFFENSE TRANSITION: Pts+/Poss" = 2, "OFFENSE TRANSI
deftrans = as.data.frame(read_csv("lineups_defense_transition_playoffs_22.csv", show_col_types = FALSE)
  select(10:27) %>%
  rename("DEFENSE TRANSITION: Pts+/Poss Rank" = 1, "DEFENSE TRANSITION: Pts+/Poss" = 2, "DEFENSE TRANSI

playoff_lineups22 = cbind.data.frame(fourfactors, shotfreq, shotacc, half, trans, defshotfreq, defshota
rm(fourfactors, shotfreq, shotacc, half, trans, defshotfreq, defshotacc, defhalf, deftrans)
```

Combine Playoff Lineup Data Frames, Clear Environment

```
all_playoff_lineups <- rbind.data.frame(playoff_lineups04, playoff_lineups05, playoff_lineups06, playof

vec1 <- c("Playoffs")
vec1 <- rep(vec1, nrow(all_playoff_lineups))
all_playoff_lineups <- mutate(all_playoff_lineups, "Regular Season or Playoffs" = vec1)
all_playoff_lineups <- relocate(all_playoff_lineups, "Regular Season or Playoffs")

rm(playoff_lineups04, playoff_lineups05, playoff_lineups06, playoff_lineups07, playoff_lineups08, playo
```

Merge Regular Season and Playoff Lineup Data

```r
all_lineups <- rbind.data.frame(all_reg_lineups, all_playoff_lineups)
```

Scrape Players From 2004-2022 From RealGM

I wanted to include biographical information for the lineups, which Cleaning The Glass lacks. So, I used RealGM's player database. Once again, I'm only including the most recent year for clutter's sake.

```r
playersurl22 <- "https://basketball.realgm.com/nba/players/2022"

players22 = playersurl22 %>%
  read_html() %>%
  html_table(fill = TRUE) %>%
  .[[13]] %>%
  mutate("Year" = 2022)
```

Combine Player Data Frames, Change Height to Inches, Change Draft Status to Numbers, Write CSV, Clear Environment

```r
#Combine
all_players_with_dups <- rbind.data.frame(players04, players05, players06, players07, players08, players

all_players_with_dups <- relocate(all_players_with_dups, Year)
all_players_with_dups <- arrange(all_players_with_dups, desc(Year))

#Change height, reorder
all_players_with_dups[c("Feet", "Inches")] <- str_split_fixed(all_players_with_dups$HT, '-', 2)
all_players_with_dups$Feet <- as.numeric(all_players_with_dups$Feet)
all_players_with_dups$Inches <- as.numeric(all_players_with_dups$Inches)
all_players_with_dups <- mutate(all_players_with_dups, "Height" = 12*Feet + Inches) %>%
  subset(select = -c(HT, Feet, Inches)) %>%
  relocate(Height, .before = WT)

#Change Draft Status to Numbers
all_players_with_dups[c("Draft Year", "Draft Pick")] <- str_split_fixed(all_players_with_dups$'Draft Sta
all_players_with_dups[c("Draft Round", "Draft Pick Num")] <- str_split_fixed(all_players_with_dups$'Drai
all_players_with_dups$`Draft Round` <- as.numeric(all_players_with_dups$`Draft Round`)
all_players_with_dups$`Draft Pick Num` <- as.numeric(all_players_with_dups$`Draft Pick Num`)
all_players_with_dups$`Draft Pick` <- all_players_with_dups$`Draft Round` * all_players_with_dups$`Draf
all_players_with_dups$`Draft Pick` <- as.numeric(all_players_with_dups$`Draft Pick`)
all_players_with_dups <- subset(all_players_with_dups, select = -c(11, 15, 16))

#Write CSV
all_players <- all_players_with_dups[!duplicated(all_players_with_dups$Player),]
write_csv(all_players_with_dups, "EachYearPlayersFrom04To22.csv")
write_csv(all_players, "TotalPlayersFrom04To22.csv")

#Remove
rm(players04, players05, players06, players07, players08, players09, players10, players11, players12, pl
```

Fix Names

Since I used 2 different data sets, there were bound to be some discrepancies in names. The below code is simply me fixing names. I had to do this for every position but did not include all of it.

```
# Manually input proper names for missing PG values
all_reg_lineups$PG[all_reg_lineups$PG=="Ronald Murray"]<-"Flip Murray"
all_reg_lineups$PG[all_reg_lineups$PG=="Marcus Williams"]<-"Marcus E. Williams"
all_reg_lineups$PG[all_reg_lineups$PG=="Roger Mason Jr."]<-"Roger Mason"
all_reg_lineups$PG[all_reg_lineups$PG=="CJ Watson"]<-"C.J. Watson"
all_reg_lineups$PG[all_reg_lineups$PG=="John Lucas III"]<-"John Lucas"
all_reg_lineups$PG[all_reg_lineups$PG=="Matthew Dellavedova"]<-"Matt Dellavedova"
all_reg_lineups$PG[all_reg_lineups$PG=="Raul Neto"]<-"Raulzinho Neto"
all_reg_lineups$PG[all_reg_lineups$PG=="CJ McCollum"]<-"C.J. McCollum"
all_reg_lineups$PG[all_reg_lineups$PG=="TJ McConnell"]<-"T.J. McConnell"
all_reg_lineups$PG[all_reg_lineups$PG=="Dennis Smith Jr."]<-"Dennis Smith"
all_reg_lineups$PG[all_reg_lineups$PG=="Ronald Murray"]<-"Flip Murray"
all_reg_lineups$PG[all_reg_lineups$PG=="Bruce Brown"]<-"Bruce Brown, Jr."
all_reg_lineups$PG[all_reg_lineups$PG=="Kevin Porter"]<-"Kevin Porter, Jr."
all_reg_lineups$PG[all_reg_lineups$PG=="Kira Lewis Jr."]<-"Kira Lewis, Jr."
all_reg_lineups$PG[all_reg_lineups$PG=="Nah'Shon Hyland"]<-"Bones Hyland"
all_reg_lineups$PG[all_reg_lineups$PG=="Gary Trent Jr."]<-"Gary Trent, Jr."
```

Transfer Height, Draft Status, and Nationality to Lineups Data

I then ran a match function to combine the 2 data sets and transfer the information I wanted. Note: One could also use a join function to complete this task.

```
#Regular Season
all_reg_lineups <- mutate(all_reg_lineups, "PG Height" = NA) %>%
  relocate("PG Height", .after = PG) %>%
  mutate(all_reg_lineups, "SG Height" = NA) %>%
  relocate("SG Height", .after = SG) %>%
  mutate(all_reg_lineups, "SF Height" = NA) %>%
  relocate("SF Height", .after = SF) %>%
  mutate(all_reg_lineups, "PF Height" = NA) %>%
  relocate("PF Height", .after = PF) %>%
  mutate(all_reg_lineups, "C Height" = NA) %>%
  relocate("C Height", .after = C)
all_reg_lineups <- mutate(all_reg_lineups, "PG Nationality" = NA) %>%
  relocate("PG Nationality", .after = 'PG Height') %>%
  mutate(all_reg_lineups, "SG Nationality" = NA) %>%
  relocate("SG Nationality", .after = 'SG Height') %>%
  mutate(all_reg_lineups, "SF Nationality" = NA) %>%
  relocate("SF Nationality", .after = 'SF Height') %>%
  mutate(all_reg_lineups, "PF Nationality" = NA) %>%
  relocate("PF Nationality", .after = 'PF Height') %>%
  mutate(all_reg_lineups, "C Nationality" = NA) %>%
  relocate("C Nationality", .after = 'C Height')
all_reg_lineups <- mutate(all_reg_lineups, "PG Draft Year" = NA) %>%
  relocate("PG Draft Year", .after = 'PG Height') %>%
  mutate(all_reg_lineups, "SG Draft Year" = NA) %>%
```

```
  relocate("SG Draft Year", .after = 'SG Height') %>%
  mutate(all_reg_lineups, "SF Draft Year" = NA) %>%
  relocate("SF Draft Year", .after = 'SF Height') %>%
  mutate(all_reg_lineups, "PF Draft Year" = NA) %>%
  relocate("PF Draft Year", .after = 'PF Height') %>%
  mutate(all_reg_lineups, "C Draft Year" = NA) %>%
  relocate("C Draft Year", .after = 'C Height')
all_reg_lineups <- mutate(all_reg_lineups, "PG Draft Pick" = NA) %>%
  relocate("PG Draft Pick", .after = 'PG Draft Year') %>%
  mutate(all_reg_lineups, "SG Draft Pick" = NA) %>%
  relocate("SG Draft Pick", .after = 'SG Draft Year') %>%
  mutate(all_reg_lineups, "SF Draft Pick" = NA) %>%
  relocate("SF Draft Pick", .after = 'SF Draft Year') %>%
  mutate(all_reg_lineups, "PF Draft Pick" = NA) %>%
  relocate("PF Draft Pick", .after = 'PF Draft Year') %>%
  mutate(all_reg_lineups, "C Draft Pick" = NA) %>%
  relocate("C Draft Pick", .after = 'C Draft Year')

# PG
pgs <- all_reg_lineups$PG
value <- c()
for (i in pgs){
    value <- c(value, match(c(i), all_players$Player))
}
all_reg_lineups$`PG Height` <- all_players$Height[value]
all_reg_lineups$`PG Nationality` <- all_players$Nationality[value]
all_reg_lineups$`PG Draft Year` <- all_players$`Draft Year`[value]
all_reg_lineups$`PG Draft Pick` <- all_players$`Draft Pick`[value]

# SG
sgs <- all_reg_lineups$SG
value <- c()
for (i in sgs){
    value <- c(value, match(c(i), all_players$Player))
}
all_reg_lineups$`SG Height` <- all_players$Height[value]
all_reg_lineups$`SG Nationality` <- all_players$Nationality[value]
all_reg_lineups$`SG Draft Year` <- all_players$`Draft Year`[value]
all_reg_lineups$`SG Draft Pick` <- all_players$`Draft Pick`[value]

# SF
sfs <- all_reg_lineups$SF
value <- c()
for (i in sfs){
    value <- c(value, match(c(i), all_players$Player))
}
all_reg_lineups$`SF Height` <- all_players$Height[value]
all_reg_lineups$`SF Nationality` <- all_players$Nationality[value]
all_reg_lineups$`SF Draft Year` <- all_players$`Draft Year`[value]
all_reg_lineups$`SF Draft Pick` <- all_players$`Draft Pick`[value]

# PF
pfs <- all_reg_lineups$PF
```

```
value <- c()
for (i in pfs){
    value <- c(value, match(c(i), all_players$Player))
}
all_reg_lineups$`PF Height` <- all_players$Height[value]
all_reg_lineups$`PF Nationality` <- all_players$Nationality[value]
all_reg_lineups$`PF Draft Year` <- all_players$`Draft Year`[value]
all_reg_lineups$`PF Draft Pick` <- all_players$`Draft Pick`[value]

# C
cs <- all_reg_lineups$C
value <- c()
for (i in cs){
    value <- c(value, match(c(i), all_players$Player))
}
all_reg_lineups$`C Height` <- all_players$Height[value]
all_reg_lineups$`C Nationality` <- all_players$Nationality[value]
all_reg_lineups$`C Draft Year` <- all_players$`Draft Year`[value]
all_reg_lineups$`C Draft Pick` <- all_players$`Draft Pick`[value]

# Add Average Height Column
all_reg_lineups <- mutate(all_reg_lineups, "Average Height" = (all_reg_lineups$`PG Height`+all_reg_lineu
all_reg_lineups <- relocate(all_reg_lineups, "Average Height", .before = "Poss")

# Playoffs
all_playoff_lineups <- mutate(all_playoff_lineups, "PG Height" = NA) %>%
  relocate("PG Height", .after = PG) %>%
  mutate(all_playoff_lineups, "SG Height" = NA) %>%
  relocate("SG Height", .after = SG) %>%
  mutate(all_playoff_lineups, "SF Height" = NA) %>%
  relocate("SF Height", .after = SF) %>%
  mutate(all_playoff_lineups, "PF Height" = NA) %>%
  relocate("PF Height", .after = PF) %>%
  mutate(all_playoff_lineups, "C Height" = NA) %>%
  relocate("C Height", .after = C)
all_playoff_lineups <- mutate(all_playoff_lineups, "PG Nationality" = NA) %>%
  relocate("PG Nationality", .after = 'PG Height') %>%
  mutate(all_playoff_lineups, "SG Nationality" = NA) %>%
  relocate("SG Nationality", .after = 'SG Height') %>%
  mutate(all_playoff_lineups, "SF Nationality" = NA) %>%
  relocate("SF Nationality", .after = 'SF Height') %>%
  mutate(all_playoff_lineups, "PF Nationality" = NA) %>%
  relocate("PF Nationality", .after = 'PF Height') %>%
  mutate(all_playoff_lineups, "C Nationality" = NA) %>%
  relocate("C Nationality", .after = 'C Height')
all_playoff_lineups <- mutate(all_playoff_lineups, "PG Draft Year" = NA) %>%
  relocate("PG Draft Year", .after = 'PG Height') %>%
  mutate(all_playoff_lineups, "SG Draft Year" = NA) %>%
  relocate("SG Draft Year", .after = 'SG Height') %>%
  mutate(all_playoff_lineups, "SF Draft Year" = NA) %>%
  relocate("SF Draft Year", .after = 'SF Height') %>%
  mutate(all_playoff_lineups, "PF Draft Year" = NA) %>%
  relocate("PF Draft Year", .after = 'PF Height') %>%
```

```
  mutate(all_playoff_lineups, "C Draft Year" = NA) %>%
  relocate("C Draft Year", .after = 'C Height')
all_playoff_lineups <- mutate(all_playoff_lineups, "PG Draft Pick" = NA) %>%
  relocate("PG Draft Pick", .after = 'PG Draft Year') %>%
  mutate(all_playoff_lineups, "SG Draft Pick" = NA) %>%
  relocate("SG Draft Pick", .after = 'SG Draft Year') %>%
  mutate(all_playoff_lineups, "SF Draft Pick" = NA) %>%
  relocate("SF Draft Pick", .after = 'SF Draft Year') %>%
  mutate(all_playoff_lineups, "PF Draft Pick" = NA) %>%
  relocate("PF Draft Pick", .after = 'PF Draft Year') %>%
  mutate(all_playoff_lineups, "C Draft Pick" = NA) %>%
  relocate("C Draft Pick", .after = 'C Draft Year')

# PG
pgs <- all_playoff_lineups$PG
value <- c()
for (i in pgs){
    value <- c(value, match(c(i), all_players$Player))
}
all_playoff_lineups$`PG Height` <- all_players$Height[value]
all_playoff_lineups$`PG Nationality` <- all_players$Nationality[value]
all_playoff_lineups$`PG Draft Year` <- all_players$`Draft Year`[value]
all_playoff_lineups$`PG Draft Pick` <- all_players$`Draft Pick`[value]

# SG
sgs <- all_playoff_lineups$SG
value <- c()
for (i in sgs){
    value <- c(value, match(c(i), all_players$Player))
}
all_playoff_lineups$`SG Height` <- all_players$Height[value]
all_playoff_lineups$`SG Nationality` <- all_players$Nationality[value]
all_playoff_lineups$`SG Draft Year` <- all_players$`Draft Year`[value]
all_playoff_lineups$`SG Draft Pick` <- all_players$`Draft Pick`[value]

# SF
sfs <- all_playoff_lineups$SF
value <- c()
for (i in sfs){
    value <- c(value, match(c(i), all_players$Player))
}
all_playoff_lineups$`SF Height` <- all_players$Height[value]
all_playoff_lineups$`SF Nationality` <- all_players$Nationality[value]
all_playoff_lineups$`SF Draft Year` <- all_players$`Draft Year`[value]
all_playoff_lineups$`SF Draft Pick` <- all_players$`Draft Pick`[value]

# PF
pfs <- all_playoff_lineups$PF
value <- c()
for (i in pfs){
    value <- c(value, match(c(i), all_players$Player))
}
all_playoff_lineups$`PF Height` <- all_players$Height[value]
```

```
all_playoff_lineups$`PF Nationality` <- all_players$Nationality[value]
all_playoff_lineups$`PF Draft Year` <- all_players$`Draft Year`[value]
all_playoff_lineups$`PF Draft Pick` <- all_players$`Draft Pick`[value]

# C
cs <- all_playoff_lineups$C
value <- c()
for (i in cs){
    value <- c(value, match(c(i), all_players$Player))
}
all_playoff_lineups$`C Height` <- all_players$Height[value]
all_playoff_lineups$`C Nationality` <- all_players$Nationality[value]
all_playoff_lineups$`C Draft Year` <- all_players$`Draft Year`[value]
all_playoff_lineups$`C Draft Pick` <- all_players$`Draft Pick`[value]

#Add Average Height Column
all_playoff_lineups <- mutate(all_playoff_lineups, "Average Height" = (all_playoff_lineups$`PG Height`+a
all_playoff_lineups <- relocate(all_playoff_lineups, "Average Height", .before = "Poss")

# All
all_lineups <- mutate(all_lineups, "PG Height" = NA) %>%
  relocate("PG Height", .after = PG) %>%
  mutate(all_lineups, "SG Height" = NA) %>%
  relocate("SG Height", .after = SG) %>%
  mutate(all_lineups, "SF Height" = NA) %>%
  relocate("SF Height", .after = SF) %>%
  mutate(all_lineups, "PF Height" = NA) %>%
  relocate("PF Height", .after = PF) %>%
  mutate(all_lineups, "C Height" = NA) %>%
  relocate("C Height", .after = C)
all_lineups <- mutate(all_lineups, "PG Nationality" = NA) %>%
  relocate("PG Nationality", .after = 'PG Height') %>%
  mutate(all_lineups, "SG Nationality" = NA) %>%
  relocate("SG Nationality", .after = 'SG Height') %>%
  mutate(all_lineups, "SF Nationality" = NA) %>%
  relocate("SF Nationality", .after = 'SF Height') %>%
  mutate(all_lineups, "PF Nationality" = NA) %>%
  relocate("PF Nationality", .after = 'PF Height') %>%
  mutate(all_lineups, "C Nationality" = NA) %>%
  relocate("C Nationality", .after = 'C Height')
all_lineups <- mutate(all_lineups, "PG Draft Year" = NA) %>%
  relocate("PG Draft Year", .after = 'PG Height') %>%
  mutate(all_lineups, "SG Draft Year" = NA) %>%
  relocate("SG Draft Year", .after = 'SG Height') %>%
  mutate(all_lineups, "SF Draft Year" = NA) %>%
  relocate("SF Draft Year", .after = 'SF Height') %>%
  mutate(all_lineups, "PF Draft Year" = NA) %>%
  relocate("PF Draft Year", .after = 'PF Height') %>%
  mutate(all_lineups, "C Draft Year" = NA) %>%
  relocate("C Draft Year", .after = 'C Height')
all_lineups <- mutate(all_lineups, "PG Draft Pick" = NA) %>%
  relocate("PG Draft Pick", .after = 'PG Draft Year') %>%
  mutate(all_lineups, "SG Draft Pick" = NA) %>%
```

```
  relocate("SG Draft Pick", .after = 'SG Draft Year') %>%
  mutate(all_lineups, "SF Draft Pick" = NA) %>%
  relocate("SF Draft Pick", .after = 'SF Draft Year') %>%
  mutate(all_lineups, "PF Draft Pick" = NA) %>%
  relocate("PF Draft Pick", .after = 'PF Draft Year') %>%
  mutate(all_lineups, "C Draft Pick" = NA) %>%
  relocate("C Draft Pick", .after = 'C Draft Year')

# PG
pgs <- all_lineups$PG
value <- c()
for (i in pgs){
    value <- c(value, match(c(i), all_players$Player))
}
all_lineups$`PG Height` <- all_players$Height[value]
all_lineups$`PG Nationality` <- all_players$Nationality[value]
all_lineups$`PG Draft Year` <- all_players$`Draft Year`[value]
all_lineups$`PG Draft Pick` <- all_players$`Draft Pick`[value]

# SG
sgs <- all_lineups$SG
value <- c()
for (i in sgs){
    value <- c(value, match(c(i), all_players$Player))
}
all_lineups$`SG Height` <- all_players$Height[value]
all_lineups$`SG Nationality` <- all_players$Nationality[value]
all_lineups$`SG Draft Year` <- all_players$`Draft Year`[value]
all_lineups$`SG Draft Pick` <- all_players$`Draft Pick`[value]

# SF
sfs <- all_lineups$SF
value <- c()
for (i in sfs){
    value <- c(value, match(c(i), all_players$Player))
}
all_lineups$`SF Height` <- all_players$Height[value]
all_lineups$`SF Nationality` <- all_players$Nationality[value]
all_lineups$`SF Draft Year` <- all_players$`Draft Year`[value]
all_lineups$`SF Draft Pick` <- all_players$`Draft Pick`[value]

# PF
pfs <- all_lineups$PF
value <- c()
for (i in pfs){
    value <- c(value, match(c(i), all_players$Player))
}
all_lineups$`PF Height` <- all_players$Height[value]
all_lineups$`PF Nationality` <- all_players$Nationality[value]
all_lineups$`PF Draft Year` <- all_players$`Draft Year`[value]
all_lineups$`PF Draft Pick` <- all_players$`Draft Pick`[value]

# C
```

```
cs <- all_lineups$C
value <- c()
for (i in cs){
    value <- c(value, match(c(i), all_players$Player))
}
all_lineups$`C Height` <- all_players$Height[value]
all_lineups$`C Nationality` <- all_players$Nationality[value]
all_lineups$`C Draft Year` <- all_players$`Draft Year`[value]
all_lineups$`C Draft Pick` <- all_players$`Draft Pick`[value]

# Add Average Height Column
all_lineups <- mutate(all_lineups, "Average Height" = (all_lineups$`PG Height`+all_lineups$`SG Height`+a
all_lineups <- relocate(all_lineups, "Average Height", .before = "Poss")
```

Check For Missing Height Values for Each Position

```
## PG
missing_pgs <- all_lineups$PG[(is.na(all_lineups$`PG Height`))]
missing_pgs <- missing_pgs[complete.cases(missing_pgs)]
missing_pgs <- unique(missing_pgs)
if (length(missing_pgs) == 0){
  rm(missing_pgs)
}else{
  print("Watch out! There are missing values for Point Guards. Go back and check if the names match.")
}

## SG
missing_sgs <- all_lineups$SG[(is.na(all_lineups$`SG Height`))]
missing_sgs <- missing_sgs[complete.cases(missing_sgs)]
missing_sgs <- unique(missing_sgs)
if (length(missing_sgs) == 0){
  rm(missing_sgs)
}else{
  print("Watch out! There are missing values for Shooting Guards. Go back and check if the names match.
}

## SF
missing_sfs <- all_lineups$SF[(is.na(all_lineups$`SF Height`))]
missing_sfs <- missing_sfs[complete.cases(missing_sfs)]
missing_sfs <- unique(missing_sfs)
if (length(missing_sfs) == 0){
  rm(missing_sfs)
}else{
  print("Watch out! There are missing values for Small Forwards. Go back and check if the names match."
}

## PF
missing_pfs <- all_lineups$PF[(is.na(all_lineups$`PF Height`))]
missing_pfs <- missing_pfs[complete.cases(missing_pfs)]
missing_pfs <- unique(missing_pfs)
if (length(missing_pfs) == 0){
```

```
  rm(missing_pfs)
}else{
  print("Watch out! There are missing values for Power Forwards. Go back and check if the names match.")
}

## C
missing_cs <- all_lineups$C[(is.na(all_lineups$`C Height`))]
missing_cs <- missing_cs[complete.cases(missing_cs)]
missing_cs <- unique(missing_cs)
if (length(missing_cs) == 0){
  rm(missing_cs)
}else{
  print("Watch out! There are missing values for Centers. Go back and check if the names match.")
}

rm(i,value,pgs,sgs,sfs,pfs,cs)
```

Transfer Age, Experience, and Games Played to Lineups Data

```
# Regular Season
all_reg_lineups <- mutate(all_reg_lineups, "PG Age" = NA) %>%
  relocate("PG Age", .after = 'PG Height') %>%
  mutate(all_reg_lineups, "SG Age" = NA) %>%
  relocate("SG Age", .after = 'SG Height') %>%
  mutate(all_reg_lineups, "SF Age" = NA) %>%
  relocate("SF Age", .after = 'SF Height') %>%
  mutate(all_reg_lineups, "PF Age" = NA) %>%
  relocate("PF Age", .after = 'PF Height') %>%
  mutate(all_reg_lineups, "C Age" = NA) %>%
  relocate("C Age", .after = 'C Height')
all_reg_lineups <- mutate(all_reg_lineups, "PG Experience" = NA) %>%
  relocate("PG Experience", .after = 'PG Age') %>%
  mutate(all_reg_lineups, "SG Experience" = NA) %>%
  relocate("SG Experience", .after = 'SG Age') %>%
  mutate(all_reg_lineups, "SF Experience" = NA) %>%
  relocate("SF Experience", .after = 'SF Age') %>%
  mutate(all_reg_lineups, "PF Experience" = NA) %>%
  relocate("PF Experience", .after = 'PF Age') %>%
  mutate(all_reg_lineups, "C Experience" = NA) %>%
  relocate("C Experience", .after = 'C Age')
all_reg_lineups <- mutate(all_reg_lineups, "PG GP" = NA) %>%
  relocate("PG GP", .after = 'PG Experience') %>%
  mutate(all_reg_lineups, "SG GP" = NA) %>%
  relocate("SG GP", .after = 'SG Experience') %>%
  mutate(all_reg_lineups, "SF GP" = NA) %>%
  relocate("SF GP", .after = 'SF Experience') %>%
  mutate(all_reg_lineups, "PF GP" = NA) %>%
  relocate("PF GP", .after = 'PF Experience') %>%
  mutate(all_reg_lineups, "C GP" = NA) %>%
  relocate("C GP", .after = 'C Experience')
```

```
# PG
pgs <- all_reg_lineups$PG
for (i in pgs){
  value <- c()
  career <- subset(all_players_with_dups, all_players_with_dups$Player == i)
  lineups_in <- subset(all_reg_lineups, pgs == i)
    for (f in lineups_in$Year){
      value <- c(value, match(c(f), career$Year))
    }
  lineups_in$`PG Age` <- career$Age[value]
  all_reg_lineups$`PG Age`[which(all_reg_lineups$PG == i)] <- lineups_in$`PG Age`
  lineups_in$`PG Experience` <- career$YOS[value]
  all_reg_lineups$`PG Experience`[which(all_reg_lineups$PG == i)] <- lineups_in$`PG Experience`
  lineups_in$`PG GP` <- career$GP[value]
  all_reg_lineups$`PG GP`[which(all_reg_lineups$PG == i)] <- lineups_in$`PG GP`
}

# SG
sgs <- all_reg_lineups$SG
for (i in sgs){
  value <- c()
  career <- subset(all_players_with_dups, all_players_with_dups$Player == i)
  lineups_in <- subset(all_reg_lineups, sgs == i)
    for (f in lineups_in$Year){
      value <- c(value, match(c(f), career$Year))
    }
  lineups_in$`SG Age` <- career$Age[value]
  all_reg_lineups$`SG Age`[which(all_reg_lineups$SG == i)] <- lineups_in$`SG Age`
  lineups_in$`SG Experience` <- career$YOS[value]
  all_reg_lineups$`SG Experience`[which(all_reg_lineups$SG == i)] <- lineups_in$`SG Experience`
  lineups_in$`SG GP` <- career$GP[value]
  all_reg_lineups$`SG GP`[which(all_reg_lineups$SG == i)] <- lineups_in$`SG GP`
}

# SF
sfs <- all_reg_lineups$SF
for (i in sfs){
  value <- c()
  career <- subset(all_players_with_dups, all_players_with_dups$Player == i)
  lineups_in <- subset(all_reg_lineups, sfs == i)
    for (f in lineups_in$Year){
      value <- c(value, match(c(f), career$Year))
    }
  lineups_in$`SF Age` <- career$Age[value]
  all_reg_lineups$`SF Age`[which(all_reg_lineups$SF == i)] <- lineups_in$`SF Age`
  lineups_in$`SF Experience` <- career$YOS[value]
  all_reg_lineups$`SF Experience`[which(all_reg_lineups$SF == i)] <- lineups_in$`SF Experience`
  lineups_in$`SF GP` <- career$GP[value]
  all_reg_lineups$`SF GP`[which(all_reg_lineups$SF == i)] <- lineups_in$`SF GP`
}

# PF
pfs <- all_reg_lineups$PF
```

```
for (i in pfs){
  value <- c()
  career <- subset(all_players_with_dups, all_players_with_dups$Player == i)
  lineups_in <- subset(all_reg_lineups, pfs == i)
    for (f in lineups_in$Year){
      value <- c(value, match(c(f), career$Year))
    }
  lineups_in$`PF Age` <- career$Age[value]
  all_reg_lineups$`PF Age`[which(all_reg_lineups$PF == i)] <- lineups_in$`PF Age`
  lineups_in$`PF Experience` <- career$YOS[value]
  all_reg_lineups$`PF Experience`[which(all_reg_lineups$PF == i)] <- lineups_in$`PF Experience`
  lineups_in$`PF GP` <- career$GP[value]
  all_reg_lineups$`PF GP`[which(all_reg_lineups$PF == i)] <- lineups_in$`PF GP`
}

# C
cs <- all_reg_lineups$C
for (i in cs){
  value <- c()
  career <- subset(all_players_with_dups, all_players_with_dups$Player == i)
  lineups_in <- subset(all_reg_lineups, cs == i)
    for (f in lineups_in$Year){
      value <- c(value, match(c(f), career$Year))
    }
  lineups_in$`C Age` <- career$Age[value]
  all_reg_lineups$`C Age`[which(all_reg_lineups$C == i)] <- lineups_in$`C Age`
  lineups_in$`C Experience` <- career$YOS[value]
  all_reg_lineups$`C Experience`[which(all_reg_lineups$C == i)] <- lineups_in$`C Experience`
  lineups_in$`C GP` <- career$GP[value]
  all_reg_lineups$`C GP`[which(all_reg_lineups$C == i)] <- lineups_in$`C GP`
}

# Playoffs
all_playoff_lineups <- mutate(all_playoff_lineups, "PG Age" = NA) %>%
  relocate("PG Age", .after = 'PG Height') %>%
  mutate(all_playoff_lineups, "SG Age" = NA) %>%
  relocate("SG Age", .after = 'SG Height') %>%
  mutate(all_playoff_lineups, "SF Age" = NA) %>%
  relocate("SF Age", .after = 'SF Height') %>%
  mutate(all_playoff_lineups, "PF Age" = NA) %>%
  relocate("PF Age", .after = 'PF Height') %>%
  mutate(all_playoff_lineups, "C Age" = NA) %>%
  relocate("C Age", .after = 'C Height')
all_playoff_lineups <- mutate(all_playoff_lineups, "PG Experience" = NA) %>%
  relocate("PG Experience", .after = 'PG Age') %>%
  mutate(all_playoff_lineups, "SG Experience" = NA) %>%
  relocate("SG Experience", .after = 'SG Age') %>%
  mutate(all_playoff_lineups, "SF Experience" = NA) %>%
  relocate("SF Experience", .after = 'SF Age') %>%
  mutate(all_playoff_lineups, "PF Experience" = NA) %>%
  relocate("PF Experience", .after = 'PF Age') %>%
  mutate(all_playoff_lineups, "C Experience" = NA) %>%
  relocate("C Experience", .after = 'C Age')
```

```
all_playoff_lineups <- mutate(all_playoff_lineups, "PG GP" = NA) %>%
  relocate("PG GP", .after = 'PG Experience') %>%
  mutate(all_playoff_lineups, "SG GP" = NA) %>%
  relocate("SG GP", .after = 'SG Experience') %>%
  mutate(all_playoff_lineups, "SF GP" = NA) %>%
  relocate("SF GP", .after = 'SF Experience') %>%
  mutate(all_playoff_lineups, "PF GP" = NA) %>%
  relocate("PF GP", .after = 'PF Experience') %>%
  mutate(all_playoff_lineups, "C GP" = NA) %>%
  relocate("C GP", .after = 'C Experience')

# PG
pgs <- all_playoff_lineups$PG
for (i in pgs){
  value <- c()
  career <- subset(all_players_with_dups, all_players_with_dups$Player == i)
  lineups_in <- subset(all_playoff_lineups, pgs == i)
    for (f in lineups_in$Year){
      value <- c(value, match(c(f), career$Year))
    }
  lineups_in$`PG Age` <- career$Age[value]
  all_playoff_lineups$`PG Age`[which(all_playoff_lineups$PG == i)] <- lineups_in$`PG Age`
  lineups_in$`PG Experience` <- career$YOS[value]
  all_playoff_lineups$`PG Experience`[which(all_playoff_lineups$PG == i)] <- lineups_in$`PG Experience`
  lineups_in$`PG GP` <- career$GP[value]
  all_playoff_lineups$`PG GP`[which(all_playoff_lineups$PG == i)] <- lineups_in$`PG GP`
}

# SG
sgs <- all_playoff_lineups$SG
for (i in sgs){
  value <- c()
  career <- subset(all_players_with_dups, all_players_with_dups$Player == i)
  lineups_in <- subset(all_playoff_lineups, sgs == i)
    for (f in lineups_in$Year){
      value <- c(value, match(c(f), career$Year))
    }
  lineups_in$`SG Age` <- career$Age[value]
  all_playoff_lineups$`SG Age`[which(all_playoff_lineups$SG == i)] <- lineups_in$`SG Age`
  lineups_in$`SG Experience` <- career$YOS[value]
  all_playoff_lineups$`SG Experience`[which(all_playoff_lineups$SG == i)] <- lineups_in$`SG Experience`
  lineups_in$`SG GP` <- career$GP[value]
  all_playoff_lineups$`SG GP`[which(all_playoff_lineups$SG == i)] <- lineups_in$`SG GP`
}

# SF
sfs <- all_playoff_lineups$SF
for (i in sfs){
  value <- c()
  career <- subset(all_players_with_dups, all_players_with_dups$Player == i)
  lineups_in <- subset(all_playoff_lineups, sfs == i)
    for (f in lineups_in$Year){
      value <- c(value, match(c(f), career$Year))
```

```
    }
  lineups_in$`SF Age` <- career$Age[value]
  all_playoff_lineups$`SF Age`[which(all_playoff_lineups$SF == i)] <- lineups_in$`SF Age`
  lineups_in$`SF Experience` <- career$YOS[value]
  all_playoff_lineups$`SF Experience`[which(all_playoff_lineups$SF == i)] <- lineups_in$`SF Experience`
  lineups_in$`SF GP` <- career$GP[value]
  all_playoff_lineups$`SF GP`[which(all_playoff_lineups$SF == i)] <- lineups_in$`SF GP`
}

# PF
pfs <- all_playoff_lineups$PF
for (i in pfs){
  value <- c()
  career <- subset(all_players_with_dups, all_players_with_dups$Player == i)
  lineups_in <- subset(all_playoff_lineups, pfs == i)
    for (f in lineups_in$Year){
      value <- c(value, match(c(f), career$Year))
    }
  lineups_in$`PF Age` <- career$Age[value]
  all_playoff_lineups$`PF Age`[which(all_playoff_lineups$PF == i)] <- lineups_in$`PF Age`
  lineups_in$`PF Experience` <- career$YOS[value]
  all_playoff_lineups$`PF Experience`[which(all_playoff_lineups$PF == i)] <- lineups_in$`PF Experience`
  lineups_in$`PF GP` <- career$GP[value]
  all_playoff_lineups$`PF GP`[which(all_playoff_lineups$PF == i)] <- lineups_in$`PF GP`
}

# C
cs <- all_playoff_lineups$C
for (i in cs){
  value <- c()
  career <- subset(all_players_with_dups, all_players_with_dups$Player == i)
  lineups_in <- subset(all_playoff_lineups, cs == i)
    for (f in lineups_in$Year){
      value <- c(value, match(c(f), career$Year))
    }
  lineups_in$`C Age` <- career$Age[value]
  all_playoff_lineups$`C Age`[which(all_playoff_lineups$C == i)] <- lineups_in$`C Age`
  lineups_in$`C Experience` <- career$YOS[value]
  all_playoff_lineups$`C Experience`[which(all_playoff_lineups$C == i)] <- lineups_in$`C Experience`
  lineups_in$`C GP` <- career$GP[value]
  all_playoff_lineups$`C GP`[which(all_playoff_lineups$C == i)] <- lineups_in$`C GP`
}

# All
all_lineups <- mutate(all_lineups, "PG Age" = NA) %>%
  relocate("PG Age", .after = 'PG Height') %>%
  mutate(all_lineups, "SG Age" = NA) %>%
  relocate("SG Age", .after = 'SG Height') %>%
  mutate(all_lineups, "SF Age" = NA) %>%
  relocate("SF Age", .after = 'SF Height') %>%
  mutate(all_lineups, "PF Age" = NA) %>%
  relocate("PF Age", .after = 'PF Height') %>%
  mutate(all_lineups, "C Age" = NA) %>%
```

```
  relocate("C Age", .after = 'C Height')
all_lineups <- mutate(all_lineups, "PG Experience" = NA) %>%
  relocate("PG Experience", .after = 'PG Age') %>%
  mutate(all_lineups, "SG Experience" = NA) %>%
  relocate("SG Experience", .after = 'SG Age') %>%
  mutate(all_lineups, "SF Experience" = NA) %>%
  relocate("SF Experience", .after = 'SF Age') %>%
  mutate(all_lineups, "PF Experience" = NA) %>%
  relocate("PF Experience", .after = 'PF Age') %>%
  mutate(all_lineups, "C Experience" = NA) %>%
  relocate("C Experience", .after = 'C Age')
all_lineups <- mutate(all_lineups, "PG GP" = NA) %>%
  relocate("PG GP", .after = 'PG Experience') %>%
  mutate(all_lineups, "SG GP" = NA) %>%
  relocate("SG GP", .after = 'SG Experience') %>%
  mutate(all_lineups, "SF GP" = NA) %>%
  relocate("SF GP", .after = 'SF Experience') %>%
  mutate(all_lineups, "PF GP" = NA) %>%
  relocate("PF GP", .after = 'PF Experience') %>%
  mutate(all_lineups, "C GP" = NA) %>%
  relocate("C GP", .after = 'C Experience')

# PG
pgs <- all_lineups$PG
for (i in pgs){
  value <- c()
  career <- subset(all_players_with_dups, all_players_with_dups$Player == i)
  lineups_in <- subset(all_lineups, pgs == i)
    for (f in lineups_in$Year){
      value <- c(value, match(c(f), career$Year))
    }
  lineups_in$`PG Age` <- career$Age[value]
  all_lineups$`PG Age`[which(all_lineups$PG == i)] <- lineups_in$`PG Age`
  lineups_in$`PG Experience` <- career$YOS[value]
  all_lineups$`PG Experience`[which(all_lineups$PG == i)] <- lineups_in$`PG Experience`
  lineups_in$`PG GP` <- career$GP[value]
  all_lineups$`PG GP`[which(all_lineups$PG == i)] <- lineups_in$`PG GP`
}

# SG
sgs <- all_lineups$SG
for (i in sgs){
  value <- c()
  career <- subset(all_players_with_dups, all_players_with_dups$Player == i)
  lineups_in <- subset(all_lineups, sgs == i)
    for (f in lineups_in$Year){
      value <- c(value, match(c(f), career$Year))
    }
  lineups_in$`SG Age` <- career$Age[value]
  all_lineups$`SG Age`[which(all_lineups$SG == i)] <- lineups_in$`SG Age`
  lineups_in$`SG Experience` <- career$YOS[value]
  all_lineups$`SG Experience`[which(all_lineups$SG == i)] <- lineups_in$`SG Experience`
  lineups_in$`SG GP` <- career$GP[value]
```

```
  all_lineups$`SG GP`[which(all_lineups$SG == i)] <- lineups_in$`SG GP`
}

# SF
sfs <- all_lineups$SF
for (i in sfs){
  value <- c()
  career <- subset(all_players_with_dups, all_players_with_dups$Player == i)
  lineups_in <- subset(all_lineups, sfs == i)
    for (f in lineups_in$Year){
      value <- c(value, match(c(f), career$Year))
    }
  lineups_in$`SF Age` <- career$Age[value]
  all_lineups$`SF Age`[which(all_lineups$SF == i)] <- lineups_in$`SF Age`
  lineups_in$`SF Experience` <- career$YOS[value]
  all_lineups$`SF Experience`[which(all_lineups$SF == i)] <- lineups_in$`SF Experience`
  lineups_in$`SF GP` <- career$GP[value]
  all_lineups$`SF GP`[which(all_lineups$SF == i)] <- lineups_in$`SF GP`
}

# PF
pfs <- all_lineups$PF
for (i in pfs){
  value <- c()
  career <- subset(all_players_with_dups, all_players_with_dups$Player == i)
  lineups_in <- subset(all_lineups, pfs == i)
    for (f in lineups_in$Year){
      value <- c(value, match(c(f), career$Year))
    }
  lineups_in$`PF Age` <- career$Age[value]
  all_lineups$`PF Age`[which(all_lineups$PF == i)] <- lineups_in$`PF Age`
  lineups_in$`PF Experience` <- career$YOS[value]
  all_lineups$`PF Experience`[which(all_lineups$PF == i)] <- lineups_in$`PF Experience`
  lineups_in$`PF GP` <- career$GP[value]
  all_lineups$`PF GP`[which(all_lineups$PF == i)] <- lineups_in$`PF GP`
}

# C
cs <- all_lineups$C
for (i in cs){
  value <- c()
  career <- subset(all_players_with_dups, all_players_with_dups$Player == i)
  lineups_in <- subset(all_lineups, cs == i)
    for (f in lineups_in$Year){
      value <- c(value, match(c(f), career$Year))
    }
  lineups_in$`C Age` <- career$Age[value]
  all_lineups$`C Age`[which(all_lineups$C == i)] <- lineups_in$`C Age`
  lineups_in$`C Experience` <- career$YOS[value]
  all_lineups$`C Experience`[which(all_lineups$C == i)] <- lineups_in$`C Experience`
  lineups_in$`C GP` <- career$GP[value]
  all_lineups$`C GP`[which(all_lineups$C == i)] <- lineups_in$`C GP`
}
```

```
rm(i, f, value, pgs, sgs, sfs, pfs, cs, career, lineups_in)
```

Remove % Signs, Make Data Numeric

```
all_lineups <- lapply(all_lineups, gsub, pattern='%', replacement='') %>%
  as_tibble() %>%
  type.convert(as.is = TRUE)
all_lineups$Year <- as.character(all_lineups$Year)
all_playoff_lineups <- as.data.frame(lapply(all_playoff_lineups, gsub, pattern='%', replacement='')) %>%
  as_tibble() %>%
  type.convert(as.is = TRUE)
all_playoff_lineups$Year <- as.character(all_playoff_lineups$Year)
all_reg_lineups <- as.data.frame(lapply(all_reg_lineups, gsub, pattern='%', replacement='')) %>%
  as_tibble() %>%
  type.convert(as.is = TRUE)
all_reg_lineups$Year <- as.character(all_reg_lineups$Year)
```

Write Lineups Data CSV

```
write_csv(all_lineups, "AllLineupsFrom04To22.csv")
write_csv(all_reg_lineups, "AllRegSeasonLineupsFrom04To22.csv")
write_csv(all_playoff_lineups, "AllPlayoffLineupsFrom04To22.csv")
all_reg_lineups_minus_averages = filter(all_reg_lineups, Team != "League Averages")
all_playoff_lineups_minus_averages = filter(all_playoff_lineups, Team != "League Averages")
write_csv(all_reg_lineups_minus_averages, "AllRegSeasonLineupsMinusAverages.csv")
write_csv(all_playoff_lineups_minus_averages, "AllPlayoffLineupsMinusAverages.csv")
```