Spatial

Derek Corcoran

September 1, 2016

2015

2016

2014

```
BostonOff <- filter(shotDataTotal2016, TEAM_NAME == "Boston Celtics")

SAOff <- filter(shotDataTotal2016, TEAM_NAME == "San Antonio Spurs")

BostonDef<- shotDatafDef2016[[2]]

SADef <- shotDatafDef2016[[23]]

## find the bounds for the complete data
xbnds <- range(c(shotDataTotal2016$LOC_X, BostonDef$LOC_X))
ybnds <- range(c(shotDataTotal2016$LOC_Y, BostonDef$LOC_Y))
nbins <- 40

#  function to make a data.frame for geom_hex that can be used with stat_identity
makeHexData <- function(df) {
  h <- hexbin(df$LOC_X, df$LOC_Y, nbins, xbnds = xbnds, ybnds = ybnds, IDs = TRUE)
  data.frame(hcell2xy(h),
             PPS = tapply(as.numeric(as.character(df$SHOT_MADE_FLAG))*ifelse(tolower(df$SHOT_TYPE) == "
             ST = tapply(df$SHOT_MADE_FLAG, h@cID, FUN = function(z) length(z)),
             cid = h@cell)
}
#make dataframes of the hex with percentages
Totalhex <- makeHexData(shotDataTotal2016)
BostonDefhex <- makeHexData(BostonDef)
SAOffhex <- makeHexData(SAOff)

##  not all cells are present in each binning, we need to merge by cellID
BostonDeffbyCell <- merge(Totalhex, BostonDefhex, by = "cid", all = T)
SAOffByCell <- merge(Totalhex, SAOffhex, by = "cid", all = T)


##  when calculating the difference empty cells should count as 0
BostonDeffbyCell$PPS.x[is.na(BostonDeffbyCell$PPS.x)] <- 0
BostonDeffbyCell$PPS.y[is.na(BostonDeffbyCell$PPS.y)] <- 0

SAOffByCell$PPS.x[is.na(SAOffByCell$PPS.x)] <- 0
SAOffByCell$PPS.y[is.na(SAOffByCell$PPS.y)] <- 0
```

```
SAOffByCell$ST.y[is.na(SAOffByCell$ST.y)] <- 0


##  make a "difference" data.frame
BostonDiffDeff <- data.frame(x = ifelse(is.na(BostonDeffbyCell$x.x), BostonDeffbyCell$x.y, BostonDeffby
                   y = ifelse(is.na(BostonDeffbyCell$y.x), BostonDeffbyCell$y.y, BostonDeffbyCell$y.x),
                   PPS= BostonDeffbyCell$PPS.y - BostonDeffbyCell$PPS.x,
                   cid= BostonDeffbyCell$cid)

SADiffOff <- data.frame(x = ifelse(is.na(SAOffByCell$x.x), SAOffByCell$x.y, SAOffByCell$x.x),
                            y = ifelse(is.na(SAOffByCell$y.x), SAOffByCell$y.y, SAOffByCell$y.x),
                            PPS= SAOffByCell$PPS.y - SAOffByCell$PPS.x,
                            ST = SAOffByCell$ST.x,
                            cid = SAOffByCell$cid)

Comparison <- merge(SADiffOff, BostonDiffDeff, by = "cid", all = T)
Comparison <- Comparison[,-c(6:7)]
Comparison$Diff <- c(Comparison$PPS.x + Comparison$PPS.y)


weighted.mean((Comparison$PPS.x + Comparison$PPS.y), Comparison$ST)
```

```
## [1] 0.03873406
```

```
### Plot Difference data.frame hex

SAOFF <- ggplot(SADiffOff)  +
  annotation_custom(court, -250, 250, -52, 418) +
    geom_hex(aes(x = x, y = y, fill = PPS),
             stat = "identity", alpha = 0.8) +
    scale_fill_gradientn (colours = c("blue","red")) +
    guides(alpha = FALSE, size = FALSE) +

  coord_fixed()  +theme(line = element_blank(),
        axis.title.x = element_blank(),
        axis.title.y = element_blank(),
        axis.text.x = element_blank(),
        axis.text.y = element_blank(),
        legend.title = element_blank(),
        plot.title = element_text(size = 17, lineheight = 1.2, face = "bold")) + ggtitle("SA Offensive\
```

```
## Scale for 'fill' is already present. Adding another scale for 'fill',
## which will replace the existing scale.
```

```
BOSDEF <- ggplot(BostonDiffDeff)  +
  annotation_custom(court, -250, 250, -52, 418) +
    geom_hex(aes(x = x, y = y, fill = PPS),
             stat = "identity", alpha = 0.8) +
    scale_fill_gradientn (colours = c("blue","red")) +
    guides(alpha = FALSE, size = FALSE) +

  coord_fixed()  +theme(line = element_blank(),
```

```
        axis.title.x = element_blank(),
        axis.title.y = element_blank(),
        axis.text.x = element_blank(),
        axis.text.y = element_blank(),
        legend.title = element_blank(),
        plot.title = element_text(size = 17, lineheight = 1.2, face = "bold")) + ggtitle("Bos Deffensiv
```

```
## Scale for 'fill' is already present. Adding another scale for 'fill',
## which will replace the existing scale.
```

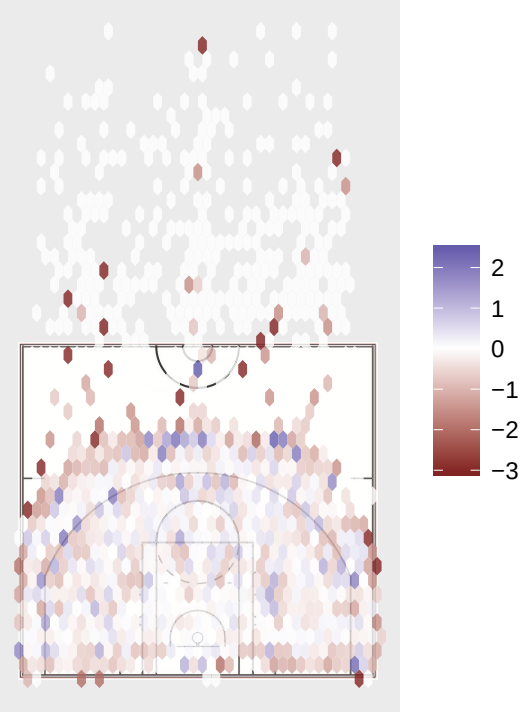
```
COMP <- ggplot(Comparison)  +
  annotation_custom(court, -250, 250, -52, 418) +
    geom_hex(aes(x = x.x, y = y.x, fill = Diff),
             stat = "identity", alpha = 0.8) +
    scale_fill_gradientn (colours = c("blue","red")) +
    guides(alpha = FALSE, size = FALSE) +

  coord_fixed()  +theme(line = element_blank(),
        axis.title.x = element_blank(),
        axis.title.y = element_blank(),
        axis.text.x = element_blank(),
        axis.text.y = element_blank(),
        legend.title = element_blank(),
        plot.title = element_text(size = 17, lineheight = 1.2, face = "bold")) + ggtitle("Comparison\n
```

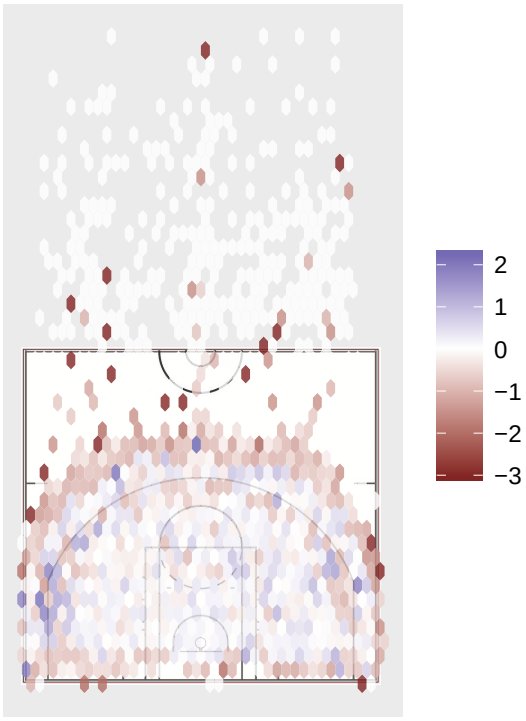
```
## Scale for 'fill' is already present. Adding another scale for 'fill',
## which will replace the existing scale.
```

```
BOSDEF
```

Bos Deffensive Shot Chart

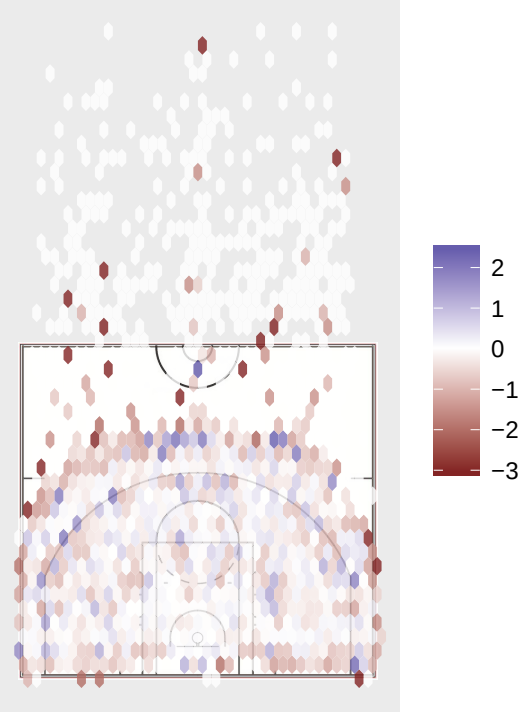



```
SAOFF
```




```
grid.arrange(BOSDEF, SAOFF, ncol=2)
```

Bos Deffensive Shot Chart



SA Offensive Shot Chart