# Washington Spirit - xG Analysis

Karl Mbouombouo

2023-11-03

## Packages

```
library(rvest)
library(tidyverse)
library(dplyr)
library(zoo)
library(magrittr)
library(reshape2)
library(ggplot2)
library(ggh4x)
```

## Data

```
# URL of the website
url <- "https://fbref.com/en/squads/e442aad0/Washington-Spirit-Stats"

# Read the HTML code of the page
html_code <- read_html(url)

# Use the html_nodes function to extract the table
table_html <- html_code %>% html_nodes("table") %>% .[[2]]

# Use the html_table function to convert the table
# HTML code into a data frame
df <- table_html %>% html_table()

# Inspect the first few rows of the data frame
head(df)
```

```
## # A tibble: 6 x 19
##   Date     Time  Comp  Round Day   Venue Result    GF    GA Opponent    xG   xGA
##   <chr>    <chr> <chr> <chr> <chr> <chr> <chr>  <int> <int> <chr>    <dbl> <dbl>
## 1 ""       ""    NWSL  Regu~ ""    Away  ""        NA    NA Bay FC    NA    NA
## 2 ""       ""    NWSL  Regu~ ""    Away  ""        NA    NA Angel C~  NA    NA
## 3 "2024-0~ "15:~ NWSL  Regu~ "Sun" Away  "L"        0     1 Reign      0     1.6
## 4 "2024-0~ "19:~ NWSL  Regu~ "Sat" Home  "W"        2     1 Bay FC     2.1   0.8
## 5 "2024-0~ "13:~ NWSL  Regu~ "Sun" Home  "W"        2     1 Royals     3.1   1.2
## 6 "2024-0~ "19:~ NWSL  Regu~ "Fri" Away  "W"        3     1 Dash       2.1   0.6
## # i 7 more variables: Poss <int>, Attendance <chr>, Captain <chr>,
## #   Formation <chr>, Referee <chr>, `Match Report` <chr>, Notes <lgl>
```

## Data Manipulation

```
# Remove column
df = select(df, -c('Time', 'Day', 'Result', 'Poss', 'Attendance', 'Captain',
                   'Formation', 'Referee', 'Match Report', 'Notes', 'GA', 'GF'))
head(df)
```

```
## # A tibble: 6 x 7
##   Date         Comp  Round          Venue Opponent      xG   xGA
##   <chr>        <chr> <chr>          <chr> <chr>      <dbl> <dbl>
## 1 ""           NWSL  Regular Season Away  Bay FC      NA    NA
## 2 ""           NWSL  Regular Season Away  Angel City  NA    NA
## 3 "2024-03-17" NWSL  Regular Season Away  Reign        0     1.6
## 4 "2024-03-23" NWSL  Regular Season Home  Bay FC       2.1   0.8
## 5 "2024-03-31" NWSL  Regular Season Home  Royals       3.1   1.2
## 6 "2024-04-12" NWSL  Regular Season Away  Dash         2.1   0.6
```

```
# Add column home team
df$Home <- "Washington Spirit"

# for loop to update "Home" values
for (i in 1:nrow(df)) {
  if (df$Venue[i] == "Away") {
    df$Home[i] <- df$Opponent[i]
  }
}

# for loop to update "Opponent" values
for (i in 1:nrow(df)) {
  if (df$Venue[i] == "Away") {
    df$Opponent[i] = "Washington Spirit"
  }
}

head(df)
```

```
## # A tibble: 6 x 8
##   Date         Comp  Round          Venue Opponent            xG   xGA Home
##   <chr>        <chr> <chr>          <chr> <chr>            <dbl> <dbl> <chr>
## 1 ""           NWSL  Regular Season Away  Washington Spirit  NA    NA   Bay FC
## 2 ""           NWSL  Regular Season Away  Washington Spirit  NA    NA   Angel C~
## 3 "2024-03-17" NWSL  Regular Season Away  Washington Spirit   0     1.6 Reign
## 4 "2024-03-23" NWSL  Regular Season Home  Bay FC              2.1   0.8 Washing~
## 5 "2024-03-31" NWSL  Regular Season Home  Royals              3.1   1.2 Washing~
## 6 "2024-04-12" NWSL  Regular Season Away  Washington Spirit   2.1   0.6 Dash
```

```
# Data from regular season & Challenge Cup
df_season <- df[df$Comp == 'NWSL', ]
df_cup <- df[df$Comp == 'NWSL Challenge Cup', ]
df_cup = select(df_cup, -('Round'))

head(df_season)
```

```
## # A tibble: 6 x 8
##   Date         Comp  Round          Venue Opponent            xG   xGA Home
##   <chr>        <chr> <chr>          <chr> <chr>            <dbl> <dbl> <chr>
```

```
## 1 ""           NWSL  Regular Season Away  Washington Spirit  NA    NA   Bay FC
## 2 ""           NWSL  Regular Season Away  Washington Spirit  NA    NA   Angel C~
## 3 "2024-03-17" NWSL  Regular Season Away  Washington Spirit   0     1.6 Reign
## 4 "2024-03-23" NWSL  Regular Season Home  Bay FC              2.1   0.8 Washing~
## 5 "2024-03-31" NWSL  Regular Season Home  Royals              3.1   1.2 Washing~
## 6 "2024-04-12" NWSL  Regular Season Away  Washington Spirit   2.1   0.6 Dash
```

```
head(df_cup)
```

```
## # A tibble: 0 x 7
## # i 7 variables: Date <chr>, Comp <chr>, Venue <chr>, Opponent <chr>, xG <dbl>,
## #   xGA <dbl>, Home <chr>
```

### Split Home and Away games

```
# Create a copy of the original data frame
home_df <- df_season

# Melt the data frame
home_df <- melt(home_df, id.vars = c("Date", "Home", "Opponent"))

home_df <- home_df[home_df$Home == 'Washington Spirit', ]

colnames(home_df)[4] <- "Variable"

home_df <- home_df %>%
  filter(grepl("xG|xGA", Variable, ignore.case = TRUE))

head(home_df)
```

```
##         Date              Home   Opponent Variable value
## 1 2024-03-23 Washington Spirit     Bay FC       xG   2.1
## 2 2024-03-31 Washington Spirit     Royals       xG   3.1
## 3 2024-04-20 Washington Spirit  Gotham FC       xG  <NA>
## 4 2024-04-26 Washington Spirit      Pride       xG  <NA>
## 5 2024-05-18 Washington Spirit Angel City       xG  <NA>
## 6 2024-05-24 Washington Spirit      Reign       xG  <NA>
```

```
# Create a copy of the original data frame
away_df <- df_season

# Melt the data frame
away_df <- melt(away_df, id.vars = c("Date", "Home", "Opponent"))

away_df <- away_df[away_df$Home != 'Washington Spirit', ]

colnames(away_df)[4] <- "Variable"

away_df <- away_df %>%
  filter(grepl("xG|xGA", Variable, ignore.case = TRUE))

head(away_df)
```

```
##         Date       Home          Opponent Variable value
## 1                Bay FC Washington Spirit       xG  <NA>
## 2           Angel City Washington Spirit       xG  <NA>
```

```
## 3 2024-03-17      Reign Washington Spirit       xG     0
## 4 2024-04-12       Dash Washington Spirit       xG   2.1
## 5 2024-05-01  Red Stars Washington Spirit       xG  <NA>
## 6 2024-05-04     Thorns Washington Spirit       xG  <NA>
```

```r
# Join home_df and away_df
df2 <- rbind(home_df, away_df)
rownames(df2) <- NULL
```

### Washington Spirit (WS)*

```r
df2 <- df2 %>% arrange(Date)
df2$value <- as.numeric(df2$value)

# xG conceded and xG created
Y_for <- df2[df2$Variable == "xG", , drop = FALSE]
Y_ag <- df2[df2$Variable == "xGA", , drop = FALSE]
X <- seq_along(Y_for$value)

# Compute the rolling average (min_periods is used for the partial average)
# Here we're using a 10 game rolling average
Y_for_avg <- rollmean(Y_for$value, k = 10, fill = NA, align = "right")

Y_ag_avg <- rollmean(Y_ag$value, k = 10, fill = NA, align = "right")

str(Y_for_avg)
```

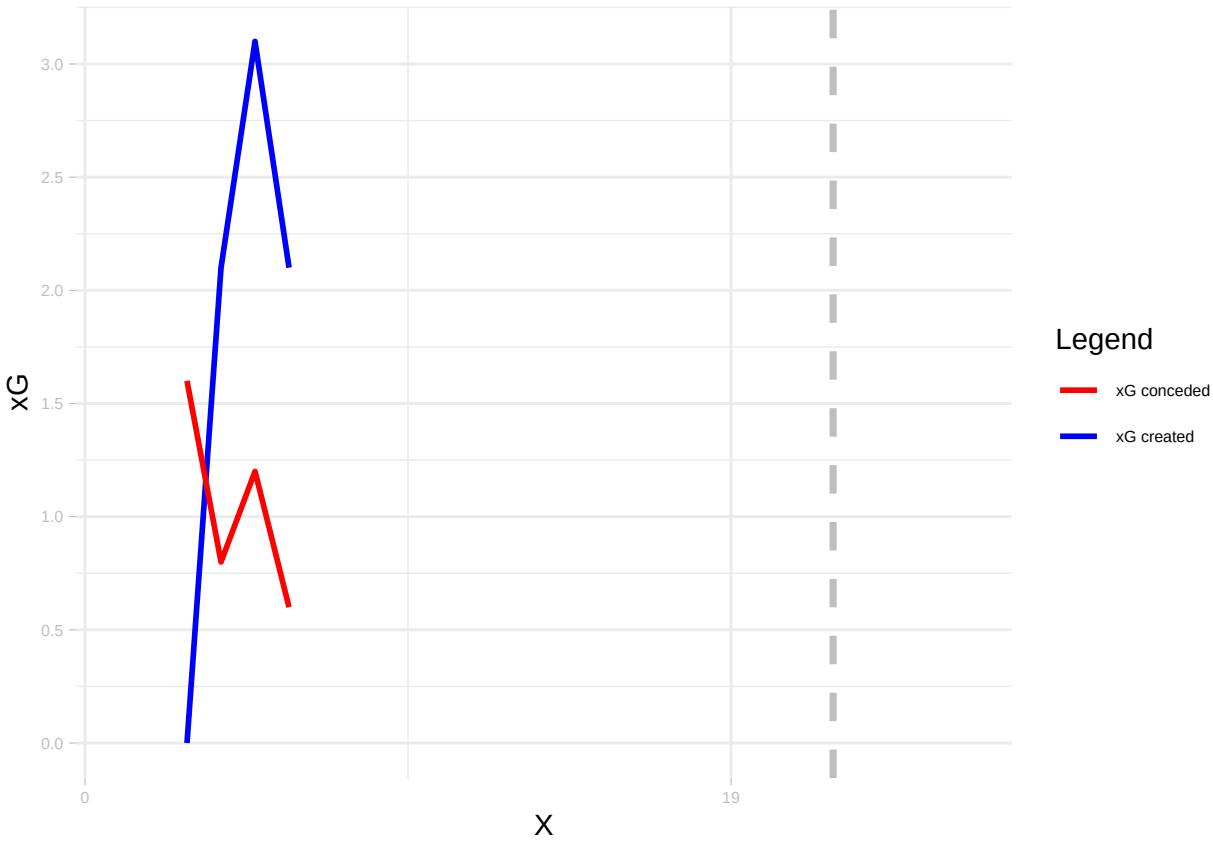
```
##  num [1:26] NA NA NA NA NA NA NA NA NA NA ...
```

```r
# Calculate the mean of all numeric values in the "value" column
mean_Y_for <- mean(Y_for$value, na.rm = TRUE)
mean_Y_ag <- mean(Y_ag$value, na.rm = TRUE)

# Replace "NA" values with the mean
Y_for_avg[is.na(Y_for_avg)] <- mean_Y_for
Y_ag_avg[is.na(Y_ag_avg)] <- mean_Y_ag
```

### Charts

```r
data <- data.frame(X = X, Y_for = Y_for, Y_ag = Y_ag)

plot <- ggplot(data, aes(x = X)) +
  geom_line(aes(y = Y_for.value, color = "xG created"), size = 1) +
  geom_line(aes(y = Y_ag.value, color = "xG conceded"), size = 1) +
  labs(color = "Legend") +
  theme_minimal() +
  theme(legend.position = "top") +
  scale_color_manual(values = c("xG created" = "blue", "xG conceded" = "red"))

plot <- plot + ylab("xG")

# Remove top & right spines and change the color
plot <- plot + theme_minimal() +
  theme(
```

```r
    axis.line = element_line(color = "grey"),  # Change axis lines' color
    panel.grid.minor = element_line(color = "lightgrey", linetype = "dotted"),  # Set minor grid lines
    panel.grid.major = element_line(color = "lightgrey", linetype = "dotted")  # Set major grid lines
  )

# Modify tick parameters
plot <- plot + theme_minimal() +
  theme(
    axis.text = element_text(color = "grey", size = 6),  # Set label color and size
    axis.ticks = element_line(color = "grey", linewidth = 0.2, lineend = "butt")  # Set tick color and
  )

# Define the x-coordinate for the division between seasons (assuming it's at game 22)
division_nwsl <- 22

# Add a vertical line to mark the division between seasons
plot <- plot + geom_vline(xintercept = division_nwsl, linetype = "dashed", size = 1.25, color = "grey")

# Set x-axis major tick positions to 19 game multiples
plot <- plot + scale_x_continuous(breaks = seq(0, max(X), by = 19))

# Set y-axis major tick positions to 0.5 xG multiples
Y_for$value[is.na(Y_for$value)] <- mean_Y_for
plot <- plot + scale_y_continuous(breaks = seq(0, max(Y_for$value), by = 0.5))

# Set y-axis lower limit to 0
plot <- plot + coord_cartesian(ylim = c(0, max(Y_for$value)))

# Adjust the legend's font size
plot <- plot + theme(legend.text = element_text(size = 6))

plot
```


3.0
2.5
2.0
1.5
1.0
0.5
0.0
xG
0
19
X
Legend
xG conceded
xG created