Rates Série A 2014-2019

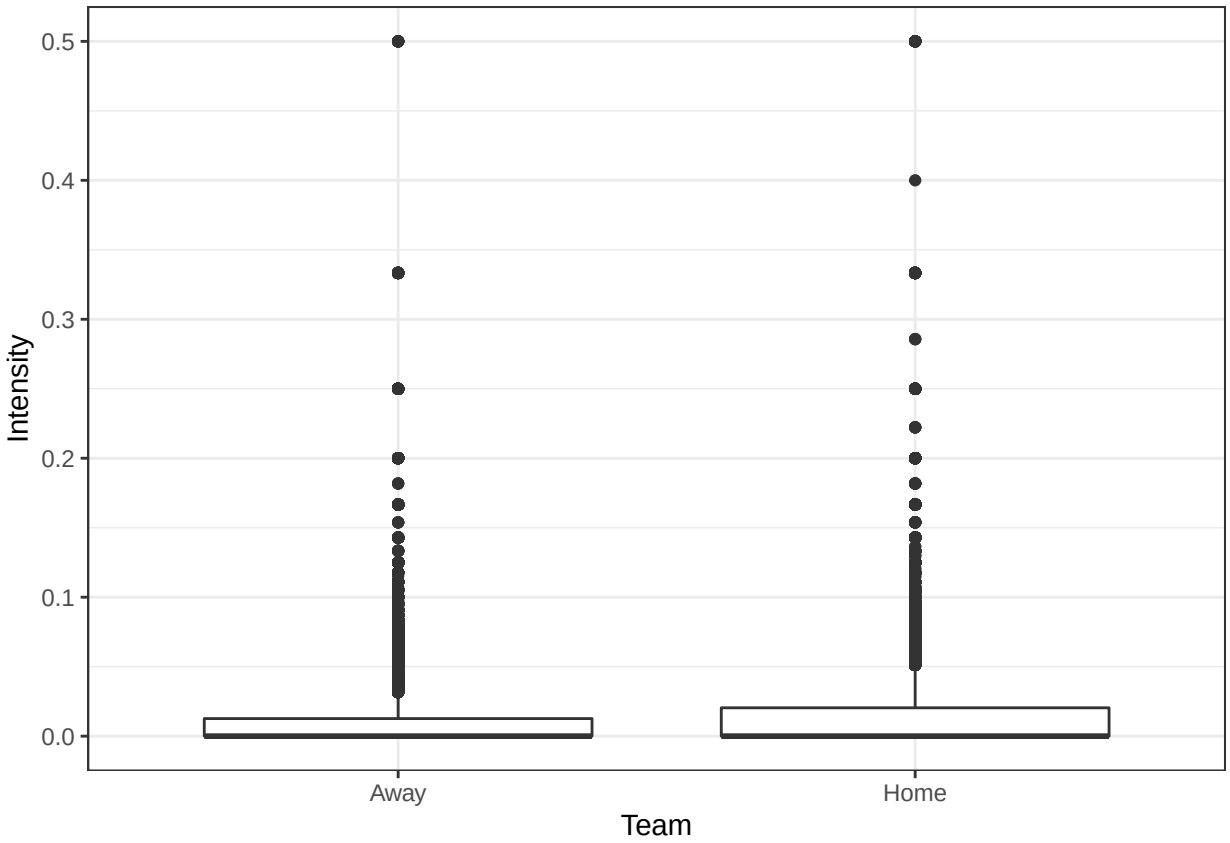
```
library(dplyr)
library(knitr)
library(reshape2)
library(ggplot2)

load("input_2014_2019.RData")

x = list(); y = list()
for(k in 1:N) {
  x[[k]] = c(x1[[k]], x2[[k]])
  y[[k]] = c(y1[[k]], y2[[k]])
}

intensity_x = list(); intensity_y = list()
for(k in 1:N) {
  intensity_x[[k]] = x[[k]]/(1:length(x[[k]]))
  intensity_y[[k]] = y[[k]]/(1:length(y[[k]]))
}

intensity_x %>%
  unlist() %>%
  summary()
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.00000 0.00000 0.00000 0.01205 0.02041 0.50000
```

```
intensity_y %>%
  unlist() %>%
  summary()
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## 0.000000 0.000000 0.000000 0.007869 0.012658 0.500000
```

```
tibble(Intensity = c(unlist(intensity_x), y = unlist(intensity_y)),
       Team = c(rep("Home", length(unlist(intensity_x))),
                rep("Away", length(unlist(intensity_y))))) %>%
  ggplot(aes(Team, Intensity)) +
  geom_boxplot() +
  theme_bw()
```




```
            #  1      2     3      4
intervals = c(0, 0.015, 0.03, 0.045, Inf)

interval_x = list(); interval_y = list()
for(k in 1:N) {
  tmp_interval_x = rep(NA, length(intensity_x[[k]]))
  tmp_interval_y = rep(NA, length(intensity_y[[k]]))
  for(i in 1:(length(intervals)-1)) {
    tmp_interval_x[which(intensity_x[[k]] >=
                           intervals[i] & intensity_x[[k]] < intervals[i+1])] = i
    tmp_interval_y[which(intensity_y[[k]] >=
                           intervals[i] & intensity_y[[k]] < intervals[i+1])] = i
  }
  interval_x[[k]] = tmp_interval_x
  interval_y[[k]] = tmp_interval_y
}

interval_x %>%
  unlist() %>%
  table()
```

```
## .
##      1      2      3      4
## 152997  41171  19075  11067
```

```
interval_y %>%
  unlist() %>%
  table()
```

```
## .
##      1      2      3      4
## 177786  30111  10307   6106
```

```
xy = list()
for(k in 1:N) {
  xy[[k]] = paste(interval_x[[k]], interval_y[[k]], sep = "-")
}

tables = lapply(xy, table)

scores = NULL
c = 0
for(i in 1:4) {
  for(j in 1:4) {
    c = c + 1
    scores[c] = paste(i, j, sep = "-")
  }
}

diff_x = list(); diff_y = list()
for(k in 1:N) {
  diff_x[[k]] = c(diff(x[[k]]))
  diff_y[[k]] = c(diff(y[[k]]))
}

t_x = list(); t_y = list()
for(k in 1:N) {
  t_x[[k]] = which(diff_x[[k]] == 1)
  t_y[[k]] = which(diff_y[[k]] == 1)
}

delta_home = list(); delta_away = list(); t = list()
for(i in 1:length(scores)) {
  tmp_delta_home = NULL; tmp_delta_away = NULL; tmp_t = NULL
  for(k in 1:N) {
    if(scores[i] %in% names(tables[[k]])) {
      tmp_t[k] = tables[[k]][scores[i]]
      minutes = which(xy[[k]] == scores[i])
      tmp_delta_home[k] = sum(t_x[[k]] %in% minutes)
      tmp_delta_away[k] = sum(t_y[[k]] %in% minutes)
    } else {
      tmp_delta_home[k] = 0
      tmp_delta_away[k] = 0
      tmp_t[k] = 0
    }
  }
  delta_home[[i]] = tmp_delta_home
  delta_away[[i]] = tmp_delta_away
```

```
  t[[i]] = tmp_t
}

rates = NULL; rates_home = NULL; rates_away = NULL
for(i in 1:length(scores)) {
  rates[i] = (sum(delta_home[[i]]) + sum(delta_away[[i]])) / sum(t[[i]])
  rates_home[i] = sum(delta_home[[i]]) / sum(t[[i]])
  rates_away[i] = sum(delta_away[[i]]) / sum(t[[i]])
}

sd_home = NULL; sd_away = NULL; sd = NULL
for(i in 1:length(scores)) {
  sd[i] = rates[i] / sqrt(sum(delta_home[[i]]) + sum(delta_away[[i]]))
  sd_home[i] = rates_home[i] / sqrt(sum(delta_home[[i]]))
  sd_away[i] = rates_away[i] / sqrt(sum(delta_away[[i]]))
}

tib = tibble(Rate = paste0("$\\nu_{", stringr::str_replace(scores, "-", ""), "}$"),
             `Est.(both)` = rates, `Est.(home)` = rates_home, `Est.(away)` = rates_away,
             `S.e.(both)` = sd, `S.e.(home)` = sd_home, `S.e.(away)` = sd_away)
```

```
kable(tib, digits = 4, caption = "Estimates and standard errors of the rate of the time
      to the next goal")
```

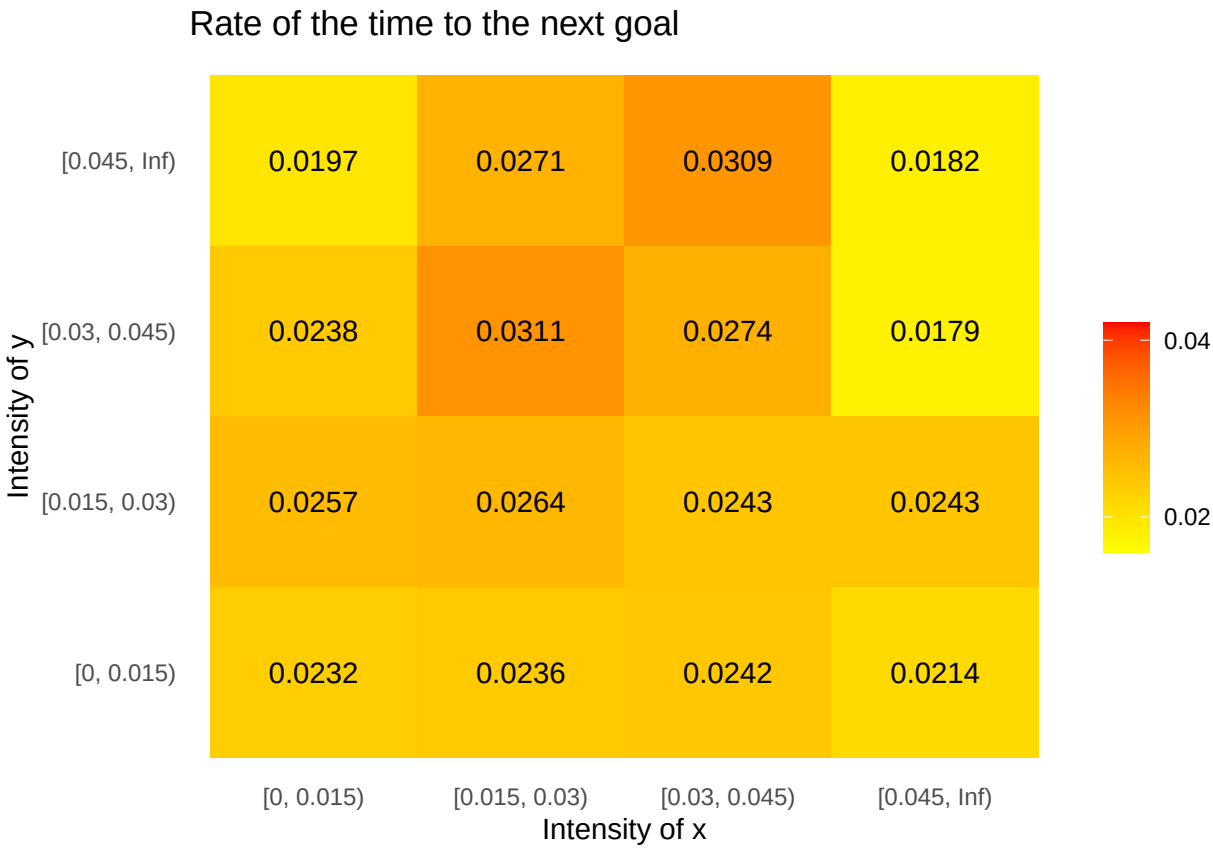
Table 1: Estimates and standard errors of the rate of the time to the next goal

Rate	Est.(both)	Est.(home)	Est.(away)	S.e.(both)	S.e.(home)	S.e.(away)
ν_{11}	0.0232	0.0140	0.0092	0.0004	0.0003	0.0003
ν_{12}	0.0257	0.0166	0.0092	0.0012	0.0010	0.0007
ν_{13}	0.0238	0.0140	0.0098	0.0019	0.0015	0.0012
ν_{14}	0.0197	0.0140	0.0057	0.0020	0.0017	0.0011
ν_{21}	0.0236	0.0142	0.0094	0.0009	0.0007	0.0006
ν_{22}	0.0264	0.0175	0.0089	0.0017	0.0014	0.0010
ν_{23}	0.0311	0.0206	0.0106	0.0043	0.0035	0.0025
ν_{24}	0.0271	0.0181	0.0090	0.0078	0.0064	0.0045
ν_{31}	0.0242	0.0142	0.0100	0.0013	0.0010	0.0008
ν_{32}	0.0243	0.0146	0.0097	0.0028	0.0022	0.0018
ν_{33}	0.0274	0.0157	0.0117	0.0037	0.0028	0.0024
ν_{34}	0.0309	0.0206	0.0103	0.0126	0.0103	0.0073
ν_{41}	0.0214	0.0131	0.0083	0.0015	0.0012	0.0010
ν_{42}	0.0243	0.0156	0.0087	0.0046	0.0037	0.0027
ν_{43}	0.0179	0.0108	0.0072	0.0080	0.0062	0.0051
ν_{44}	0.0182	0.0127	0.0055	0.0057	0.0048	0.0031

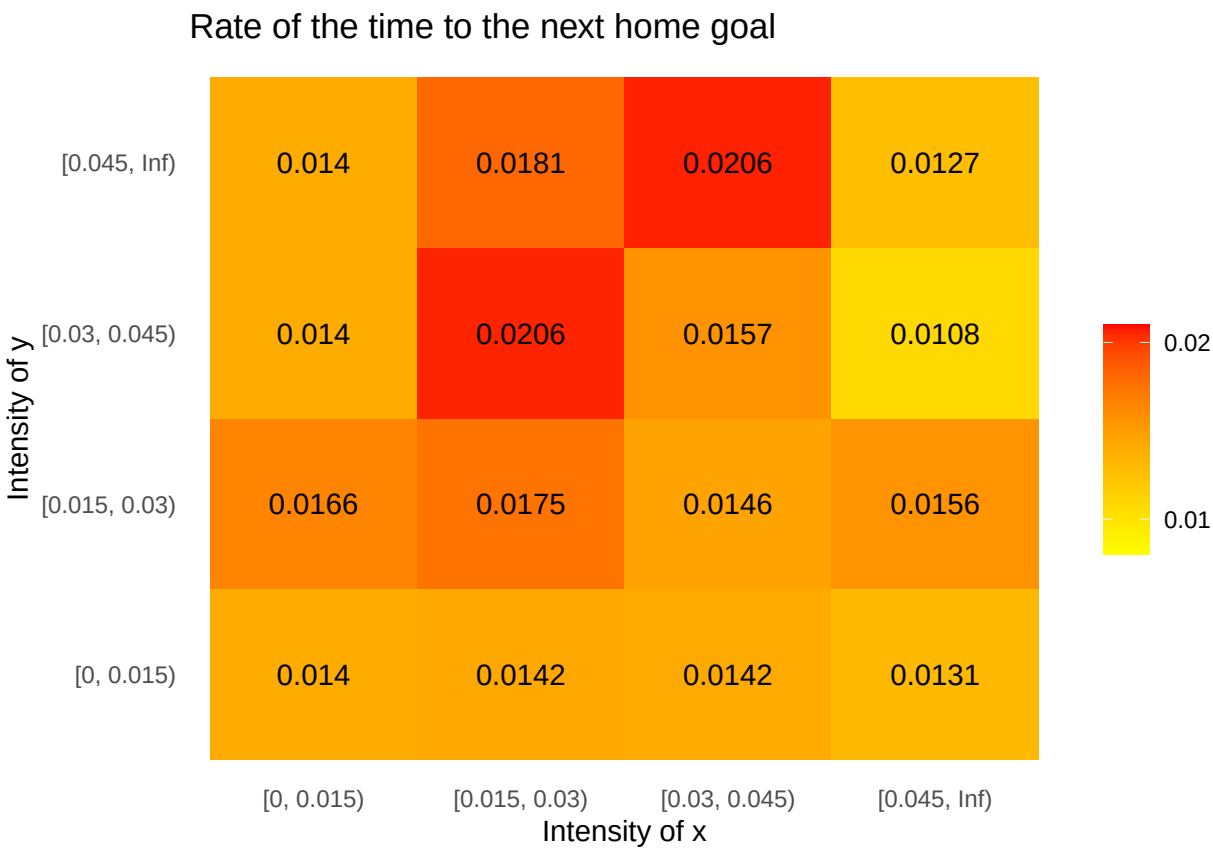
```
names(rates) = scores
names(rates_home) = scores
names(rates_away) = scores

mat = matrix(NA, nrow = 4, ncol = 4)
rownames(mat) = paste0(1:4)
```

```
colnames(mat) = paste0(1:4)
mat_home = mat
mat_away = mat
for(i in 1:4) {
  for(j in 1:4) {
    mat[i,j] = rates[paste(i, j, sep = "-")]
    mat_home[i,j] = rates_home[paste(i, j, sep = "-")]
    mat_away[i,j] = rates_away[paste(i, j, sep = "-")]
  }
}

melted_mat = melt(mat) %>%
  rename(x = Var1, y = Var2)
melted_mat_home = melt(mat_home) %>%
  rename(x = Var1, y = Var2)
melted_mat_away = melt(mat_away) %>%
  rename(x = Var1, y = Var2)

for(i in 1:(length(intervals)-1)) {
  melted_mat$x[which(melted_mat$x == i)] =
    paste0("[", intervals[i], ", ", intervals[i+1], ")")
  melted_mat_home$x[which(melted_mat_home$x == i)] =
    paste0("[", intervals[i], ", ", intervals[i+1], ")")
  melted_mat_away$x[which(melted_mat_away$x == i)] =
    paste0("[", intervals[i], ", ", intervals[i+1], ")")
  melted_mat$y[which(melted_mat$y == i)] =
    paste0("[", intervals[i], ", ", intervals[i+1], ")")
  melted_mat_home$y[which(melted_mat_home$y == i)] =
    paste0("[", intervals[i], ", ", intervals[i+1], ")")
  melted_mat_away$y[which(melted_mat_away$y == i)] =
    paste0("[", intervals[i], ", ", intervals[i+1], ")")
}
```

```
ggplot(data = melted_mat, aes(x, y, fill = value)) +
  geom_tile() +
  scale_fill_gradient2(low = "yellow", mid = "orange", high = "red",
                       midpoint = 0.029, limit = c(0.016, 0.042),
                       name = "",
                       breaks = c(0.02, 0.04)) +
  geom_text(aes(x, y, label = round(value, 4)), color = "black") +
  theme(panel.grid.major = element_blank(),
        panel.border = element_blank(),
        panel.background = element_blank(),
        axis.ticks = element_blank()) +
  ggtitle("Rate of the time to the next goal") +
  xlab("Intensity of x") +
  ylab("Intensity of y")
```


Rate of the time to the next goal
[0.045, Inf)
[0.03, 0.045)
[0.015, 0.03)
[0, 0.015)
Intensity of y
0.0197
0.0271
0.0309
0.0182
0.0238
0.0311
0.0274
0.0179
0.0257
0.0264
0.0243
0.0243
0.0232
0.0236
0.0242
0.0214
[0, 0.015)
[0.015, 0.03)
[0.03, 0.045)
[0.045, Inf)
Intensity of x
0.04
0.02

```
ggplot(data = melted_mat_home, aes(x, y, fill = value)) +
  geom_tile() +
  scale_fill_gradient2(low = "yellow", mid = "orange", high = "red",
                       midpoint = 0.0145, limit = c(0.008, 0.021),
                       name = "",
                       breaks = c(0.01, 0.02)) +
  geom_text(aes(x, y, label = round(value, 4)), color = "black") +
  theme(panel.grid.major = element_blank(),
        panel.border = element_blank(),
        panel.background = element_blank(),
        axis.ticks = element_blank()) +
  ggtitle("Rate of the time to the next home goal") +
  xlab("Intensity of x") +
  ylab("Intensity of y")
```



```
ggplot(data = melted_mat_away, aes(x, y, fill = value)) +
  geom_tile() +
  scale_fill_gradient2(low = "yellow", mid = "orange", high = "red",
                       midpoint = 0.0145, limit = c(0.008, 0.021),
                       name = "",
                       breaks = c(0.01, 0.02)) +
  geom_text(aes(x, y, label = round(value, 4)), color = "black") +
  theme(panel.grid.major = element_blank(),
        panel.border = element_blank(),
        panel.background = element_blank(),
        axis.ticks = element_blank()) +
  ggtitle("Rate of the time to the next away goal") +
  xlab("Intensity of x") +
  ylab("Intensity of y")
```