$$f(t) = (t+1)^3$$

```
options(scipen = 999)

library(dplyr)
library(ggplot2)
library(CVXR)

load("scrape/data/reds.RData")
load("2015-2020/data/input.RData")
```

$$\lambda_k^*(t) = A_\lambda (t+1)^3$$
$$\mu_k^*(t) = A_\mu (t+1)^3$$

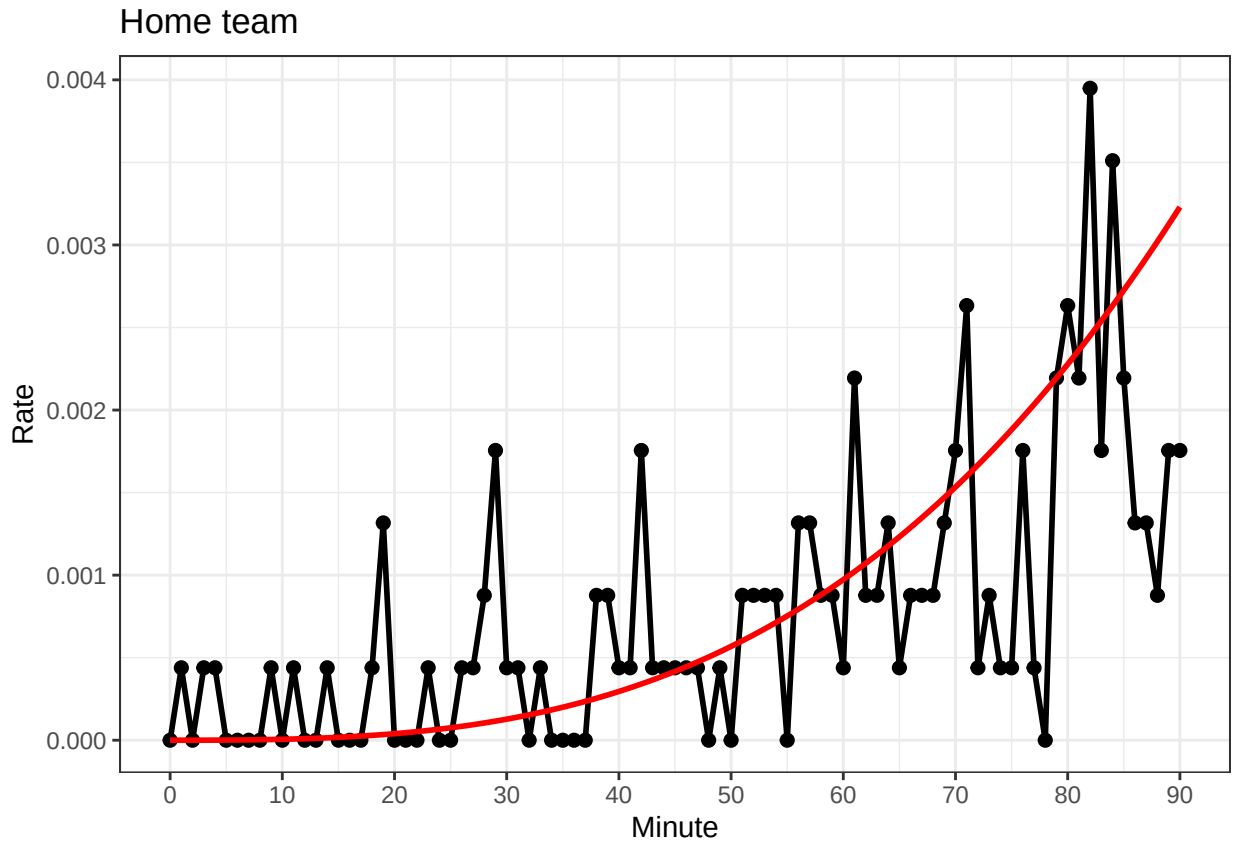
```
reds$Stoppage_Time[which(is.na(reds$Stoppage_Time))] = 0

reds = reds %>%
  mutate(Minute = Minute + Stoppage_Time) %>%
  filter(Minute <= 45)

reds$Minute[which(reds$Half == 2)] = reds$Minute[which(reds$Half == 2)] + 45

tib_zeros = tibble(Minute = 0:90, n = 0L)
complete_zeros <- function(tib_count) {
  tib_count %>%
    full_join(tib_zeros, by = c("Minute", "n")) %>%
    group_by(Minute) %>%
    summarise(n = sum(n))
}

home = reds %>%
  filter(Team == 1) %>%
  count(Minute) %>%
  complete_zeros() %>%
  mutate(Rate = n/N)

both = reds %>%
  count(Minute) %>%
  complete_zeros() %>%
  mutate(Rate = n/N)

away = reds %>%
  filter(Team == 2) %>%
  count(Minute) %>%
  complete_zeros() %>%
  mutate(Rate = n/N)
```

```
for(k in 1:N) {
  I2s[[k]] = I2s[[k]] + 45
}

int_reds_1 = list(); int_reds_2 = list();
for(k in 1:N) {
  tmp_int_reds_1 = NULL
  for(l in 1:(length(I1s[[k]])-1)) {
    t1 = I1s[[k]][l]
    t2 = I1s[[k]][l+1]
    tmp_int_reds_1[l] = 1/4*((t2+1)^4 - (t1+1)^4) # <---
  }
  int_reds_1[[k]] = tmp_int_reds_1
}
for(k in 1:N) {
  tmp_int_reds_2 = NULL
  for(l in 1:(length(I2s[[k]])-1)) {
    t1 = I2s[[k]][l]
    t2 = I2s[[k]][l+1]
    tmp_int_reds_2[l] = 1/4*((t2+1)^4 - (t1+1)^4) # <---
  }
  int_reds_2[[k]] = tmp_int_reds_2
}
int_reds_1 = unlist(int_reds_1)
int_reds_2 = unlist(int_reds_2)

a = Variable(2)

loglambda1s = log(int_reds_1) + a[1]
logmu1s = log(int_reds_1) + a[2]
loglambda2s = log(int_reds_2) + a[1]
logmu2s = log(int_reds_2) + a[2]

log_lik_reds = sum_entries(
  - exp(loglambda1s) - exp(logmu1s) +
    H1s*loglambda1s + A1s*logmu1s) +
  sum_entries(
    - exp(loglambda2s) - exp(logmu2s) +
      + H2s*loglambda2s + A2s*logmu2s)

objective = Maximize(log_lik_reds)
problem = Problem(objective)
solution = solve(problem, solver = "MOSEK")
A = exp(as.vector(solution$getValue(a)))
names(A) = c("A_lambda", "A_mu")
A
```

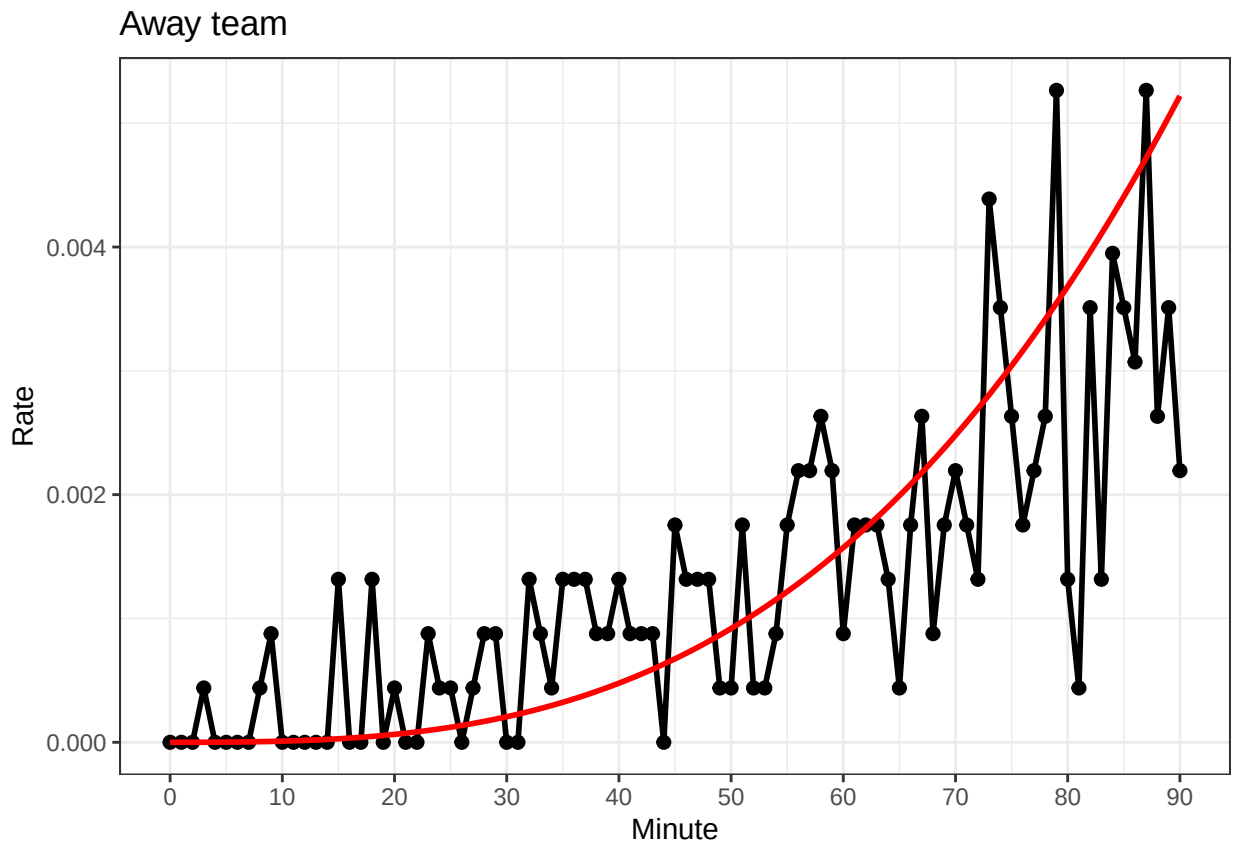
```
##          A_lambda              A_mu
## 0.000000004285642 0.000000006927765
```

```
t = seq(0, 90, by = 0.01)
rate_home = A[1] * (t+1)^3 # <---
rate_away = A[2] * (t+1)^3 # <---
```

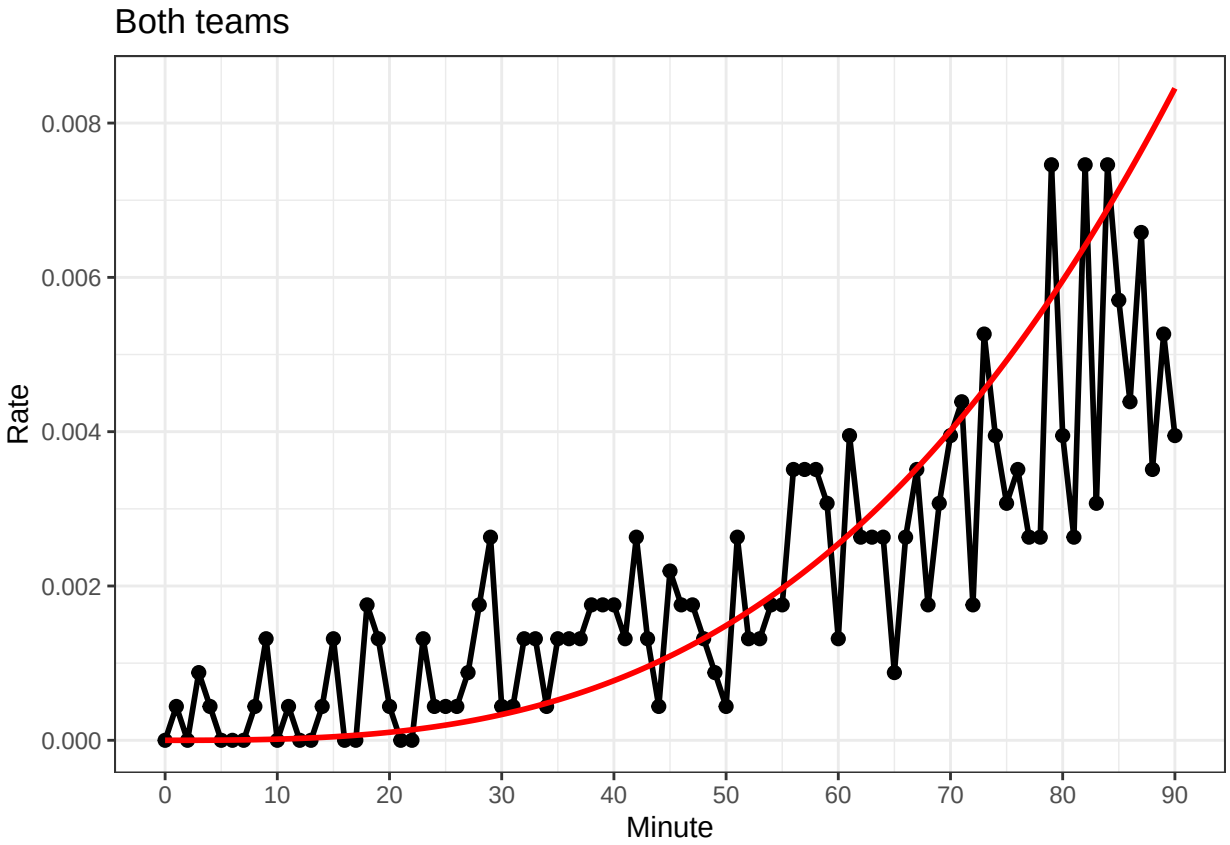
```
rate_both = rate_home + rate_away

home %>%
  ggplot(aes(x = Minute, y = Rate)) +
  geom_line(size = 1) +
  geom_point(size = 2) +
  theme_bw() +
  scale_x_continuous(breaks = seq(from = 0, to = 90, by = 10),
                     labels = c(seq(from = 0, to = 90, by = 10))) +
  geom_line(data = tibble(t, rate_both), aes(x = t, y = rate_home),
            col = "red", size = 1) +
  ggtitle("Home team")
```




```
away %>%
  ggplot(aes(x = Minute, y = Rate)) +
  geom_line(size = 1) +
  geom_point(size = 2) +
  theme_bw() +
  scale_x_continuous(breaks = seq(from = 0, to = 90, by = 10),
                     labels = c(seq(from = 0, to = 90, by = 10))) +
  geom_line(data = tibble(t, rate_away), aes(x = t, y = rate_away),
            col = "red", size = 1) +
  ggtitle("Away team")
```




```
both %>%
  ggplot(aes(x = Minute, y = Rate)) +
  geom_line(size = 1) +
  geom_point(size = 2) +
  theme_bw() +
  scale_x_continuous(breaks = seq(from = 0, to = 90, by = 10),
                     labels = c(seq(from = 0, to = 90, by = 10))) +
  geom_line(data = tibble(t, rate_both), aes(x = t, y = rate_both),
            col = "red", size = 1) +
  ggtitle("Both teams")
```




```
solution$value
```

```
## [1] -2670.37
```