The Least-Squares Method

Download Data

```
df <- read.csv(file="https://www.football-data.co.uk/mmz4281/2324/E0.csv", header = T)

FTHG = df$FTHG
```

Define lambdas

```
lambda_str = seq(0.1, 10, by=0.01)
```

Original MLE method

```
log_likelihood_pois = function(theta, data) {
  l = sum(dpois(data, lambda=theta, log=T))
  return(l)
}

value = optimize(log_likelihood_pois, c(0, 10), maximum=TRUE, data=FTHG)
print(value)
```

```
## $maximum
## [1] 1.800004
##
## $objective
## [1] -631.4621
```

The Least Squares Method

```
## The method I was suggesting is called least-square method. Implement it.

rf = table(FTHG) / length(FTHG)  # Relative frequencies
rf
```

```
## FTHG
##           0           1           2           3           4           5
## 0.178947368 0.286842105 0.255263158 0.165789474 0.068421053 0.036842105
##           6
## 0.007894737
```

```
cols = c(0, 1, 2, 3, 4, 5, 6)

# Define the least-squares objective function
least_square = function(theta) {
  predicted = dpois(cols, lambda = theta)
  l = sum((predicted - rf)^2)
  return(l)
}
```

```r
# Optimize
result = optimize(least_square, interval = c(0, 10), maximum = FALSE)
result
```

```
## $minimum
## [1] 1.791971
##
## $objective
## [1] 0.0006115876
```

As can be seen, the lambda is converging to 1.8 just like in the other method.