Kobe record

Mark Chen

2020/2/9

Kober Bryant NBA record (R)

To explore the NBA career of Kobe Bryant, the best player in our generation !

Import packages

```
library(tidyverse) # Data wrangle
library(lubridate) # Date
```

Import data

```
working_dir <- "C:/Users/Public/medium_kobe/"
file_name <- "kobe.csv"
file_dir <- paste0(working_dir, file_name)
data <- read.csv(file_dir, stringsAsFactors = F)
```

Data Preprocessing

```
data <- mutate(data, shot_point = as.numeric(substring(shot_type, 1, 1)),
               shot_gain = shot_point * shot_made_flag,
               total_seconds_remaining = minutes_remaining * 60 + seconds_remaining)
data[, "game_date"] <- as.Date(data[, "game_date"] ,"%Y/%m/%d")
```

(1)

```
data_1 <-
  data %>%
  select(opponent, shot_type, shot_made_flag) %>%
  filter(opponent == "HOU") %>%
  group_by(shot_type) %>%
  summarise(FGp = sum(shot_made_flag)/length(shot_made_flag),
            shooting_number = n())
data_1
```

shot_type	FGp	shooting_number
2PT Field Goal	0.3881481	1350
3PT Field Goal	0.2658228	316

(2)

```
data_2 <-
  data %>%
  select(game_id, opponent, shot_gain) %>%
  group_by(game_id, opponent) %>%
  summarise(game_score = sum(shot_gain)) %>%
  group_by(opponent) %>%
  summarise(mean_score = mean(game_score)) %>%
  arrange(mean_score)
head(data_2,5)
```

opponent	mean_score
VAN	11.16667
BKN	12.66667
IND	14.02564
UTA	14.14286
NJN	14.17857

(3)

```
data_3 <-
  data %>%
  select(game_id, opponent, playoffs, period, total_seconds_remaining, shot_gain) %>%
  filter(playoffs == 1, period == 4, total_seconds_remaining <= 3*60) %>%
  group_by(game_id, opponent) %>%
  summarise(total_scores = sum(shot_gain)) %>%
  arrange(desc(total_scores))
head(data_3,5)
```

game_id	opponent	total_scores
49900074	POR	9
40100234	SAS	8
40700402	BOS	7
49900024	SAC	7
40100402	NJN	6

(4)

```
data_4 <-
  data %>%
  select(season, opponent, playoffs, period, total_seconds_remaining,
         action_type, shot_made_flag) %>%
  filter(playoffs == 1, period == 4, total_seconds_remaining <= 1*60,
         action_type == "Jump Shot") %>%
  group_by(season) %>%
  summarise(FGp = sum(shot_made_flag)/ length(shot_made_flag))
data_4
```

season	FGp
1996-97	0.4000000
1997-98	0.0000000
1998-99	0.1428571
1999-00	0.1000000
Apr-03	0.1666667
Aug-07	0.0000000
Dec-11	0.0909091
Feb-01	0.5000000
Jan-00	0.0000000
Jul-06	0.5000000
Jun-05	0.0000000
Mar-02	0.3333333
Nov-10	0.0000000
Oct-09	0.3333333
Sep-08	0.1428571

(5)

```
continuous_detect <- function (x) {
  #check
  if (class(x)[1] == "tbl_df"){
    x <- pull(x)
  }
  ini <- 1; con_df <- c(); end <- length(x)
  while (ini != end) {
    if ( x[ini] == 0) {
      ini <- ini + 1
    }else{
      start <- ini
      while (x[ini] == 1 ) {
        ini <- ini + 1
      }
      df <- data.frame(start_index = start, end_index = ini - 1, interval = ini - start )
      con_df <- rbind(con_df, df)
    }
  }
  con_df
}
data_5_intme <-
  data %>%
  select(game_id, game_date, shot_made_flag) %>%
  group_by(game_id, game_date) %>%
  summarise(FGp = sum(shot_made_flag)/ n()) %>%
  arrange(game_date) %>%
  mutate(meet_goal = if_else(FGp < 0.33, 0, 1) )
dt <- continuous_detect(data_5_intme[,4])
date <- dt %>% arrange(desc(interval)) %>% head(3)
calender <- lapply(1:nrow(date), function(i)
  as.character(c(data_5_intme$game_date[date[i,1]], data_5_intme$game_date[date[i,2]])))
data_5 <- as.data.frame(cbind(do.call(rbind,calender), date$interval))
colnames(data_5) <- c("starting_date", "ending_date", "interval")
data_5
```

starting_date	ending_date	interval
2000-02-01	2000-03-12	19
2001-06-15	2001-12-07	19
2002-01-14	2002-02-17	16

(6)

```
FsurpassS_id <-
  data %>%
  select(game_date, period, shot_gain) %>%
  mutate(half = ifelse(period <= 2, "first", "second")) %>%
  group_by(game_date, half) %>%
  summarise(half_scores = sum(shot_gain)) %>%
  group_by(game_date) %>%
  summarise(gains_diff =  half_scores[1] - half_scores[2]) %>%
  filter(gains_diff > 0)
data_6 <-
  data %>%
  select(game_date, period, opponent, shot_gain, shot_made_flag) %>%
  filter(game_date %in% FsurpassS_id$game_date, period %in% 1:4) %>%
  group_by(game_date, opponent) %>%
  summarise(FGp = sum(shot_made_flag)/n(),
            points = sum(shot_gain)) %>%
  inner_join(FsurpassS_id) %>%
  arrange(FGp)
head(data_6,3)
```

game_date	opponent	FGp	points	gains_diff
2015-11-24	GSW	0.0714286	3	3
2003-12-21	PHX	0.0833333	2	2
2016-04-05	LAC	0.0833333	2	2

(7)

```
continuous_time <- 3
continuous_max_intvl <- function (x) {
  ini <- 1; end <- length(x); interval <- 0;interval_list <- c()
  if ( end < 2) {
    if ( x[1] == 0){
      interval <- interval + 1
    }
    interval
  }else{
    interval_list <-  c(interval_list, interval)
    while (ini != end + 1 ) {
      if ( x[ini] == 1) {
        ini <- ini + 1
      }else{
        interval <- 0
        while ((x[ini] == 0) & (ini != end + 1)){
          interval <- interval + 1
          ini <- ini + 1
        }
        interval_list <- c(interval_list, interval)
        }
    }
    max(interval_list)
  }
}
data_7_ini <-
  data %>%
  select(game_date, period, minutes_remaining, shot_made_flag) %>%
  arrange(game_date, period) %>%
  group_split(game_date)
total_intvl <-unlist(lapply(1:length(data_7_ini), function(i)
  continuous_max_intvl(data_7_ini[[i]]$shot_made_flag) ) )
total_date <- unlist(lapply(1:length(data_7_ini), function(i)
  as.character(data_7_ini[[i]]$game_date[1])))
total_date <- as.Date(total_date ,"%Y-%m-%d")
date_intvl <- data.frame(game_date = total_date, interval = total_intvl)
index <- date_intvl %>% arrange(desc(interval)) %>% head(continuous_time)
data_7 <-
  data %>%
  select(game_date, opponent, shot_gain) %>%
  filter(game_date %in% index$game_date) %>%
  group_by(game_date, opponent) %>%
  summarise(points = sum(shot_gain)) %>%
  inner_join(index) %>%
  arrange(desc(interval))
data_7
```

game_date	opponent	points	interval
2010-04-02	UTA	8	16
2015-12-01	PHI	10	16
2003-03-12	DET	12	15

Kober Bryant NBA record (Python)

```
import pandas as pd
import numpy as np
import time
from itertools import accumulate
import os
os.chdir("C:/Users/Public/medium_kobe/")
kobe_data = pd.read_csv("kobe.csv")
data = kobe_data.copy() #copy, to construct scores dataset
data["shot_type_value"] = data["shot_type"].apply(lambda x: 2 if x=="2PT Field Goal" else 3) #map
data["points"] = data["shot_type_value"] * data["shot_made_flag"] #transform
data['game_date'] =pd.to_datetime(data['game_date']) #data time
data.sort_values("game_date",inplace=True)
data.index =list(range(len(data)))
```

(1)

```
data_1 = kobe_data.copy()
data_1.index = data_1.opponent
data_1_final = data_1.loc[["HOU"],["shot_type","shot_made_flag"]].groupby("shot_type").mean()
data_1_final.rename(columns ={"shot_made_flag":"Average_field_goal_rate"},inplace = True) #Rename colum
data_1_final.head()
```

```
##                 Average_field_goal_rate
## shot_type
## 2PT Field Goal                 0.388148
## 3PT Field Goal                 0.265823
```