

gameday

Zichun Ye

2014-11-20

You know then problem: You're in your office writing R code and suddenly have the urge to check whether your NHL team has a game or what the scores of NHL games are today. Before you know it you just wasted 15 minutes browsing the latest news on your favorite hockey webpage. Suffer no more! You can now ask R directly, without tempting yourself by firing up your web browser.

Function

1. gday

- parameter: team and date;
- return: Logical. TRUE if team has a NHL game on date, FALSE otherwise.

2. scores

- parameter: date;
- return: a data frame with game place, home team, away team, home score, away score and whether the game is completed.

3. info_team

- parameter: team and date;
- return: a data frame with information about arena, team, lat and long

Dataset

1. arena_team: information about arena, team, lat and long

More Examples

The following is some examples of these two functions.

- E.g. 1: calculate the winning percentage of home team in the past month.

```
library(plyr)
library(ggplot2)
library(ggthemes)
library(gameday)
date<-data.frame(date = as.Date(-30:0,Sys.time()))

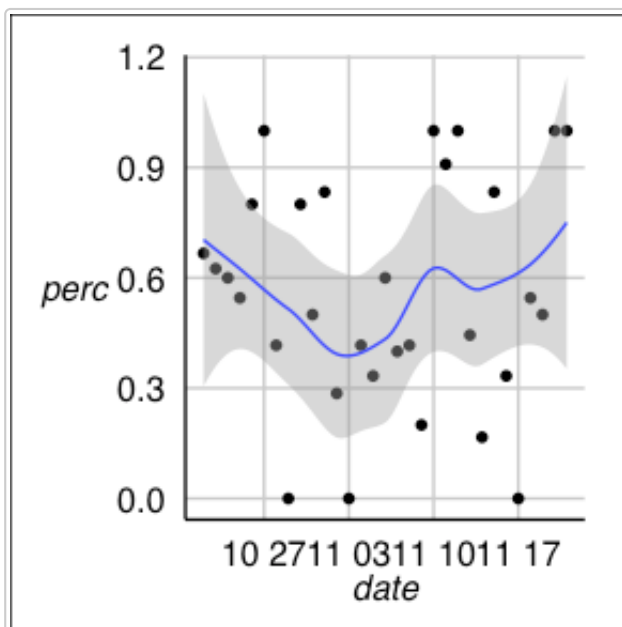
perc<-ddply(date, ~date, function(x){
  score<-scores(x$date)
  return(data.frame(perc=nrow(score[score$home_score>score$away_score,])/nrow(score)))
})
```

```
})
```

```
## Warning in Ops.factor(score$home_score, score$away_score): > not
## meaningful for factors
```

```
## Warning in Ops.factor(score$home_score, score$away_score): > not
## meaningful for factors
```

```
ggplot(perc,aes(x = date, y = perc))+
  geom_point()+
  geom_smooth(method = "loess")+
  theme_gdocs()
```



- E.g. 2: compare games and back to back games of Canucks in the last 6 weeks.

Here back to back game is defined to be have games in two contiguous days.

```
date<-data.frame(date = as.Date(-41:0,Sys.time()))
```

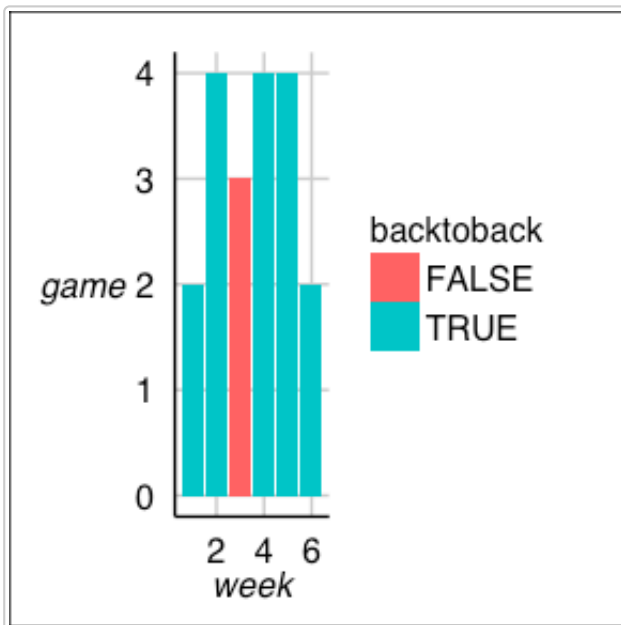
```
games<-ddply(date, ~date, function(x){
  if(gday(date = x$date)) return(TRUE)
  return(FALSE)
})
```

```
backtoback<-ddply(date, ~date, function(x){
  if(gday(date = x$date)&gday(date = x$date+1)) return(TRUE)
  return(FALSE)
})
```

```
data<-data.frame(week = 1:6)
```

```
for(i in 1:6){
  data$game[i] = sum(games[(7*i-6):(7*i),2])
  data$backtoback[i] = (sum(backtoback[(7*i-6):(7*i),2])!=0)
}
```

```
ggplot(data, aes(week, game, fill=backtoback))+  
  geom_bar(stat="identity")+  
  theme_gdocs()
```



The red bar represents the week without back to back game while the green one represent the week with back to back games.