Week 6.1 - Understanding and Cleaning the NBA Shot Log Data

We will use the 2016-2017 basketball shot log data to demonstrate how to test the hot hand.

```
library(tidyverse)
```

Import useful libraries and the shot log data

```
## -- Attaching packages --------------------------------------------------- tidyverse 1.3.0 --
## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.3     v dplyr   1.0.0
## v tidyr   1.1.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0
## -- Conflicts ------------------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
Shotlog = read.csv("~/Google Drive/Sports Analytics Moocs/MOOC 1 - Foundations of sports analytics/Week
head(Shotlog)
```

```
##   team_previous_shot player_position home_game location_x
## 1                                 SF       Yes         97
## 2             MISSED               C       Yes         52
## 3             SCORED              SG       Yes        239
## 4             SCORED              PG       Yes        102
## 5             SCORED              PF       Yes        128
## 6             MISSED              PG       Yes        210
##   opponent_previous_shot home_team            shot_type points away_team
## 1                 SCORED       ATL     Pullup Jump Shot      2       WAS
## 2                 SCORED       ATL        Tip Dunk Shot      2       WAS
## 3                 MISSED       ATL            Jump Shot      2       WAS
## 4                 SCORED       ATL     Pullup Jump Shot      2       WAS
## 5                 MISSED       ATL Turnaround Jump Shot      2       WAS
## 6                 SCORED       ATL     Pullup Jump Shot      2       WAS
##   location_y time       date    shoot_player time_from_last_shot quarter
## 1        405 1:09 10/27/2016   Kent Bazemore                  NA       1
## 2        250 1:11 10/27/2016   Dwight Howard                   2       1
## 3        223 1:41 10/27/2016     Kyle Korver                  30       1
## 4        385 2:16 10/27/2016 Dennis Schroder                  35       1
## 5        265 2:40 10/27/2016    Paul Millsap                  24       1
## 6        267 3:07 10/27/2016 Dennis Schroder                  27       1
```

```
##   current_shot_outcome
## 1               MISSED
## 2               SCORED
## 3               SCORED
## 4               SCORED
## 5               MISSED
## 6               MISSED
```

```
dim(Shotlog)
```

```
## [1] 210072     16
```

Data Preparation

Missing Value

```
sapply(Shotlog, function(x) sum(is.na(x)))
```

```
##     team_previous_shot        player_position              home_game
##                      0                      0                      0
##             location_x opponent_previous_shot              home_team
##                    397                      0                      0
##              shot_type                 points              away_team
##                      0                      0                      0
##             location_y                   time                   date
##                    397                      0                      0
##           shoot_player    time_from_last_shot                quarter
##                      0                  10000                      0
##   current_shot_outcome
##                      0
```

Let's create some useful variables.

- Create dummy variables to indicate hit or miss of current shot and previous shot.

```
Shotlog$current_shot_hit = ifelse(Shotlog$current_shot_outcome =="SCORED", 1, 0)
head(Shotlog)
```

```
##   team_previous_shot player_position home_game location_x
## 1                                SF       Yes         97
## 2            MISSED               C       Yes         52
## 3            SCORED              SG       Yes        239
## 4            SCORED              PG       Yes        102
## 5            SCORED              PF       Yes        128
## 6            MISSED              PG       Yes        210
##   opponent_previous_shot home_team            shot_type points away_team
## 1                 SCORED       ATL     Pullup Jump Shot      2       WAS
## 2                 SCORED       ATL        Tip Dunk Shot      2       WAS
## 3                 MISSED       ATL            Jump Shot      2       WAS
## 4                 SCORED       ATL     Pullup Jump Shot      2       WAS
## 5                 MISSED       ATL Turnaround Jump Shot      2       WAS
## 6                 SCORED       ATL     Pullup Jump Shot      2       WAS
##   location_y time       date    shoot_player time_from_last_shot quarter
## 1        405 1:09 10/27/2016   Kent Bazemore                  NA       1
## 2        250 1:11 10/27/2016   Dwight Howard                   2       1
## 3        223 1:41 10/27/2016     Kyle Korver                  30       1
```

```
## 4        385 2:16 10/27/2016 Dennis Schroder                  35       1
## 5        265 2:40 10/27/2016    Paul Millsap                  24       1
## 6        267 3:07 10/27/2016 Dennis Schroder                  27       1
##   current_shot_outcome current_shot_hit
## 1               MISSED                0
## 2               SCORED                1
## 3               SCORED                1
## 4               SCORED                1
## 5               MISSED                0
## 6               MISSED                0
```

- Make sure the variable "date" is stored as a date type variable.

```
Shotlog$date = mdy(Shotlog$date)
```

- Convert the variable "time" to be datetime type variable We will use "hm" from the lubridate package to work with variable with only time information.

```
Shotlog$time = hm(Shotlog$time)
```

- Create lagged variable to indicate the result of the previous shot by the same player in the same game.

1. We will first sort the shot outcome by the time in the game;
2. We will group the data by player and game (date) and use the "lag" command to create a lag variable.

```
Shotlog = Shotlog %>% group_by(shoot_player, date) %>%
  mutate(lag_shot_hit = lag(current_shot_hit, order_by = time)) %>%
  ungroup()
head(Shotlog)
```

```
## # A tibble: 6 x 18
##   team_previous_s~ player_position home_game location_x opponent_previo~
##   <chr>            <chr>           <chr>          <int> <chr>
## 1 ""               SF              Yes               97 SCORED
## 2 "MISSED"         C               Yes               52 SCORED
## 3 "SCORED"         SG              Yes              239 MISSED
## 4 "SCORED"         PG              Yes              102 SCORED
## 5 "SCORED"         PF              Yes              128 MISSED
## 6 "MISSED"         PG              Yes              210 SCORED
## # ... with 13 more variables: home_team <chr>, shot_type <chr>, points <int>,
## #   away_team <chr>, location_y <int>, time <Period>, date <date>,
## #   shoot_player <chr>, time_from_last_shot <int>, quarter <int>,
## #   current_shot_outcome <chr>, current_shot_hit <dbl>, lag_shot_hit <dbl>
```

```
Shotlog = Shotlog %>% arrange(shoot_player, date, time)
```

We can sort the shot log data by player, game(date), and time of the shot. *Notice that for the first shots of the game by the given players, the lagged outcome variable will have missing value.*

Let's create a dataframe for average success rate of players over the season. Since the "current_shot_hit" variable is a dummy variable (=1 if hit, =0 if miss), the average of this variable would indicate the success rate of the player over the season.

```
Player_Stats = Shotlog %>% group_by(shoot_player) %>%
  summarise(mean(current_shot_hit)) %>% ungroup()
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
head(Player_Stats)
```

```
## # A tibble: 6 x 2
##   shoot_player   `mean(current_shot_hit)`
##   <chr>                             <dbl>
## 1 A.J. Hammons                      0.405
## 2 Aaron Brooks                      0.403
## 3 Aaron Gordon                      0.455
## 4 Aaron Harrison                    0
## 5 Adreian Payne                     0.426
## 6 Al Horford                        0.473
```

- Let's rename the "current_shot_hit" variable in the newly created date frame as "average_hit".

```
Player_Stats = rename(Player_Stats, average_hit = 'mean(current_shot_hit)')
```

```
Shotlog = left_join(Shotlog, Player_Stats, by = 'shoot_player')
head(Shotlog)
```

We will use the player statistics to analyze the hot hand. So we will merge average player statistics dataframe back to the shot log dataframe.

```
## # A tibble: 6 x 19
##   team_previous_s~ player_position home_game location_x opponent_previo~
##   <chr>            <chr>           <chr>          <int> <chr>
## 1 MISSED           C               No               210 SCORED
## 2 SCORED           C               No               308 SCORED
## 3 MISSED           C               No               167 SCORED
## 4 SCORED           C               No               131 MISSED
## 5 MISSED           C               No                72 MISSED
## 6 SCORED           C               Yes              882 SCORED
## # ... with 14 more variables: home_team <chr>, shot_type <chr>, points <int>,
## #   away_team <chr>, location_y <int>, time <Period>, date <date>,
## #   shoot_player <chr>, time_from_last_shot <int>, quarter <int>,
## #   current_shot_outcome <chr>, current_shot_hit <dbl>, lag_shot_hit <dbl>,
## #   average_hit <dbl>
```

- Create a variable to indicate the total number of shots recorded in the dataset for each player.

```
Player_Shots = Shotlog %>% count(shoot_player) %>% rename(shot_count = n) %>%
  arrange(desc(shot_count))
head(Player_Shots)
```

```
## # A tibble: 6 x 2
##   shoot_player      shot_count
##   <chr>                  <int>
## 1 Russell Westbrook       1940
## 2 Andrew Wiggins          1568
## 3 DeMar DeRozan           1545
## 4 James Harden            1532
## 5 Anthony Davis           1525
## 6 Damian Lillard          1489
```

We should also note that players have different number of shots in each individual game. We will need to treat the data differently for a player who had only two shots in a game compared to those who had attempted 30 in a game.

- Create a variable to indicate the number of shots in each game for by each player.

```
Player_Game = Shotlog %>% count(shoot_player, date) %>% rename(shot_per_game = n)
head(Player_Game)
```

```
## # A tibble: 6 x 3
##   shoot_player date       shot_per_game
##   <chr>        <date>             <int>
## 1 A.J. Hammons 2016-11-09             5
## 2 A.J. Hammons 2016-11-23             1
## 3 A.J. Hammons 2016-11-25             1
## 4 A.J. Hammons 2016-12-03             2
## 5 A.J. Hammons 2016-12-07             2
## 6 A.J. Hammons 2016-12-12             2
```

```
Shotlog = left_join(Shotlog, Player_Shots, by = 'shoot_player')

Shotlog = left_join(Shotlog, Player_Game,
                    by = c('shoot_player', 'date'))
head(Shotlog)
```

We will merge the shot count data frames back to the shot log dataframe.

```
## # A tibble: 6 x 21
##   team_previous_s~ player_position home_game location_x opponent_previo~
##   <chr>            <chr>           <chr>          <int> <chr>
## 1 MISSED           C               No               210 SCORED
## 2 SCORED           C               No               308 SCORED
## 3 MISSED           C               No               167 SCORED
## 4 SCORED           C               No               131 MISSED
## 5 MISSED           C               No                72 MISSED
## 6 SCORED           C               Yes              882 SCORED
## # ... with 16 more variables: home_team <chr>, shot_type <chr>, points <int>,
## #   away_team <chr>, location_y <int>, time <Period>, date <date>,
## #   shoot_player <chr>, time_from_last_shot <int>, quarter <int>,
## #   current_shot_outcome <chr>, current_shot_hit <dbl>, lag_shot_hit <dbl>,
## #   average_hit <dbl>, shot_count <int>, shot_per_game <int>
```

```
tail(Shotlog)
```

```
## # A tibble: 6 x 21
##   team_previous_s~ player_position home_game location_x opponent_previo~
##   <chr>            <chr>           <chr>          <int> <chr>
## 1 MISSED           C               Yes              118 MISSED
## 2 BLOCKED          C               Yes              866 SCORED
## 3 SCORED           C               Yes               58 MISSED
## 4 SCORED           C               Yes              239 SCORED
## 5 MISSED           C               Yes               52 SCORED
## 6 MISSED           C               Yes              241 MISSED
## # ... with 16 more variables: home_team <chr>, shot_type <chr>, points <int>,
## #   away_team <chr>, location_y <int>, time <Period>, date <date>,
## #   shoot_player <chr>, time_from_last_shot <int>, quarter <int>,
## #   current_shot_outcome <chr>, current_shot_hit <dbl>, lag_shot_hit <dbl>,
## #   average_hit <dbl>, shot_count <int>, shot_per_game <int>
```

```
Shotlog = Shotlog %>% arrange(shoot_player, date, time)
head(Shotlog)
```

We will sort the data again after merging.

```
## # A tibble: 6 x 21
##   team_previous_s~ player_position home_game location_x opponent_previo~
##   <chr>            <chr>           <chr>          <int> <chr>
## 1 MISSED           C               No               210 SCORED
## 2 SCORED           C               No               308 SCORED
## 3 MISSED           C               No               167 SCORED
## 4 SCORED           C               No               131 MISSED
## 5 MISSED           C               No                72 MISSED
## 6 SCORED           C               Yes              882 SCORED
## # ... with 16 more variables: home_team <chr>, shot_type <chr>, points <int>,
## #   away_team <chr>, location_y <int>, time <Period>, date <date>,
## #   shoot_player <chr>, time_from_last_shot <int>, quarter <int>,
## #   current_shot_outcome <chr>, current_shot_hit <dbl>, lag_shot_hit <dbl>,
## #   average_hit <dbl>, shot_count <int>, shot_per_game <int>
```

```
Shotlog$points = as.factor(Shotlog$points)
Shotlog$quarter = as.factor(Shotlog$quarter)
```

We will treat the "points" and "quarter" variables as factors.

Missing values

- Drop observations with missing value in lagged variable.

```
Shotlog = Shotlog %>% filter(!is.na(lag_shot_hit))
```

```
dim(Shotlog)
```

Let's take a quick look at the number of variables and the number of observations in our clean dataframe.

```
## [1] 185052     21
```

Save our updated data

```
write.csv(Shotlog, 'Shotlog1.csv', row.names=FALSE)
write.csv(Player_Stats, 'Player_Stats.csv', row.names=FALSE)
write.csv(Player_Shots, 'Player_Shots.csv', row.names=FALSE)
write.csv(Player_Game, 'Player_Game.csv', row.names=FALSE)
```