THE EFFECT OF CONGESTED FIXTURE PERIOD ON BASEBALL PLAYER'S PERFORMANCE

EXECUTIVE SUMMARY

The objective of this report is to investigate the correlation between/influence of congested match schedules and the performances of Baseball players. The variation in performances (using different performance metrics for pitchers and batters) of 20 pitchers and batters each (with 5 top and bottom players in the list and 10 being in the middle of the list) were examined for different time gaps between consecutive matches from Major League Baseball for the period 2017-2019. Play-by-play data and the box score data was used from Retrosheet. Metrics used to measure performance differs for pitchers and batters. For pitchers, we us ER and SO and for batters, we use OPS and BA. The statistical tests for the top and middle order batters showed that there is no effect of congestion on batter performance, however the test for the bottom order batters showed that there is an effect of congestion on their performance. This shows that the higher order batters are used to the congested nature of baseball tournament schedules and are unaffected by it.

INTRODUCTION

In sports analytics, statistical methods are applied to the game data to estimate and gain answers to the many existing questions. In team sports, accurate assessment of player performance is an invaluable part of understanding his contribution to the team and the team's success. The insights gained from these assessments allows better decision making and helps to make modifications accordingly. In Baseball, these evaluations can be conducted using performance standards which are easily quantifiable and are capable of being compared. The evaluation methods rely on statistical measures of individual and team performance.

Statistics plays an integral role in sports, from rating players to deciding who plays, with which teammates, against which team, to measuring their performances and forecasting results. Hence, use of statistics in baseball is not a new concept. It helps to measure performance and productivity. According to Estes, it is "imperative for evaluators" to understand the factors responsible for increase or decrease in player performance .

The purpose of this research study is to investigate the effect of intense match schedules on baseball player performance. The performance metrics for measuring hitting performance are bats, hits, home runs, runs batted in (RBIs), at bats, plate appearances, batting average (BA), on-base percentage (OBP), slugging percentage (SLG) and on-base plus slugging (OPS). Measures of pitching performance include win/loss ratio, earned runs, saves, innings pitches, strikeouts, earned runs average and others. In this analysis, we have used BA, OPS and ER and SO for batters and pitchers respectively.

The ideal measure of overall player performance is each player's contribution to the number of games that his team wins. However, this is a bit complex as it is difficult to assign the total number of wins to each player as players have various roles and performance. The research work will help address this unexplored area. Its aim is to identify if there is some correlation between heavy match schedules and player performance.

The game of Baseball is arranged with a high amount of games within a short period of time. Such hectic schedules

Several investigations have analysed the pitcher and batter performance over seasons { }

However, not much has been done to understand the effects of congested match schedules on batter and pitcher performance.

The investigation is done on 40 Major League Players, out of which 20 are pitcher and the rest are batters.

The major findings revealed that the

However, these findings are not conclusive due to specific limitations:

1. Number of years the player has been playing is not taken into consideration which is a factor that affects the players ability to deal with congested schedules while maintaining his performance.

Further research is necessary to explore the effects of short period of time between matches on player performance.

LITERATURE REVIEW

Baseball has become one of the most analysed sport [1] with an enormous amount of data collected every game. Use of data analysis in Baseball has evolved over years. Sabermetrics is one of the categories mainly used in the analysis of player performance [2].

According to research, participation in sports can result in reduction in the anaerobic performance of players. There is some evidence that the highly intensive competition periods and/or training exposure can have a negative impact on match performance and can lead to increased injury risk [2]. Therefore, one can expect a drop in measures of physical performance when there is an insufficient resting time between matches.

In addition to that, recent study says that Baseball players these days are throwing with higher velocity and hitting with more power than ever before. [3] Using more energy along with congested match schedules can affect player performance.

Much of the academic literature involves evaluating player performance post overuse injuries [16], evaluating pre and post-operative player performance [14][17], predicting player performance, analysis of batting [15] and hitting performance of players (yearly or seasonally) and teams, among all the other explored areas. However, no work has taken place to study the player performance as a result of player overuse; whether back-to-back matches bring out enhanced, deteriorated or normal player performance. There is a need to study the statistical difference in athlete performance across successive matches played by a baseball player in a short time versus matches played after an interval.

According to a study in the game of football, fixture congestion is considered as a threat to both match performance and players' health. To maintain match performance during dense schedules, player rotation strategies have been suggested [8]. However, investigation conducted by … on a professional soccer team concluded that athlete performance and injury rates were generally unaffected and they were able to "cope with a congested playing calendar" [9]. A similar research has been conducted for a professional soccer team. They investigated the "influence of playing multiple games with a short recovery time between matches on physical activity, technical performance and injury rates" [10]. This area needs to be explored in Baseball as it will be an edition to the existing knowledge.

Too many matches can lead to deterioration of concentration which can affect coordination leading to underperformance and greater risk of injury [11]. It can also lead to lack of motivation and mental burn out – that is, players can no longer gear themselves up for the matches and training sessions [12].

The findings from the study will be noteworthy in a way that it will be the first attempt to highlight the impact of successive matches on player performance. The performance metrics that will be used are proven criteria of performance measurements [13]. The data that will be used for the research will be from trustworthy repository – Retrosheet. A quantitative research method will be used to answer the research question – How is intense match schedule associated to player performance? How is the player performance during congested fixture period different from that during non-congested fixture period?

The scope of this analysis is limited to pitchers and hitters and not fielders. To explain briefly, a pitcher in Baseball is the player who throws the ball towards the catcher to begin each play, who in turn is faced by a batter or hitter. The pitcher aims to retire the batter while the latter aims to either hit the ball or draw a walk. Hitting in Baseball is horizontal, unlike other sports in which there is vertical hitting.

APPROACH

To analyse the effect of congested fixture schedule on player performance, I examined different performance metrics for both pitchers and batters for both congested and non-congested periods. All the matches played by the players for the period 2017-2019 were considered. As Baseball matches are generally closely scheduled, a congested period is the one which has one or less than one day gap

between two consecutive matches. Consequently, a non-congested is a period with more than one day gap between matches.

ABOUT THE DATA

Retrosheet, a volunteer organization with a strong interest in the history of Baseball, was founded in 1989 by Professor David Smith. Its website contains play-by-play data available at https://www.retrosheet.org/game.htm, accounting for every "event' happening in a game and game-by-game data which accounts for the data of every Baseball game played since 1871 till date available at https://www.retrosheet.org/gamelogs/index.html# . [4]

Performance metrics data is produced by Retrosheet.

We are ignoring the data for fielder and for the purpose of this analysis are focusing only on hitters and pitchers.

The images below give a quick view of the data.

B_SH	B_SF	B_SB	B_CS	B_XI	B_G_DH	B_G_PH	B_G_PR	P_G	P_GS	P_CG	P_SHO	P_GF	P_W	P_L	P_SV	P_OUT	P_TBF	P_AB	P_R
0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	3	3	3	0
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	2	2	0
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	3	4	4	1
0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	4	7	5	1
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	1	0
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	1	1	0
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	3	3	3	0

game.key	game.source	game.date	game.number	appear.date	site.key	season.phase	team.alignment	team.key	opponent.key	person.key	slot
ARI201908150	evt	2019-08-15	0	2019-08-15	PHO01	R	0	SFN	ARI	abadf001	2
ARI201908160	evt	2019-08-16	0	2019-08-16	PHO01	R	0	SFN	ARI	abadf001	9
ARI201908180	evt	2019-08-18	0	2019-08-18	PHO01	R	0	SFN	ARI	abadf001	9
CHN201908200	evt	2019-08-20	0	2019-08-20	CHI11	R	0	SFN	CHN	abadf001	9
OAK201908240	evt	2019-08-24	0	2019-08-24	OAK01	R	0	SFN	OAK	abadf001	0
SFN201908260	evt	2019-08-26	0	2019-08-26	SFO03	R	1	SFN	ARI	abadf001	9
SFN201908270	evt	2019-08-27	0	2019-08-27	SFO03	R	1	SFN	ARI	abadf001	9
SFN201908290	evt	2019-08-29	0	2019-08-29	SFO03	R	1	SFN	SDN	abadf001	9

PARTICIPANTS

The players included in the study were based on their ranking in the LeaderBoards for multiple seasons from 2017 to 2019, irrespective of the teams they played for. This data was collected from a popular sports website [1] which provides statistics for every Major League Baseball player. From the 115 pitchers for the full season, a sample of 20 was taken – 5 top pitchers, 10 middle level pitchers and 5 bottom level pitchers. Similarly, 20 batters were taken from a total of 234. [1]

The following table shows the Batters and the Pitchers included in this analysis, along with their Retrosheet IDs –

PITCHERS	BATTERS

	Name	RetrosheetID		Name	RetrosheetID
TOP 5	Max Scherzer	schem001	TOP 5	Mike Trout	troum001
	Jacob deGrom	degrj001		Mookie Betts	bettm001
	Chris Sale	salec001		Christian Yelich	yelic001
	Justin Verlander	verlj001		Anthony Rendon	renda001
	Gerrit Cole	coleg001		Alex Bregman	brega001
MIDDLE 10	Joe Musgrove	musgj001	MIDDLE 10	Jonathan Villar	villj001
	Cole Hamels	hamec001		Justin Smoak	smoaj001
	Mike Foltynewicz	foltm001		Austin Hedges	hedga001
	Dallas Keuchel	keucd001		Josh Reddick	reddj001
	Kyle Freeland	freek001		Kevin Pillar	pillk001
	Dylan Bundy	bundd001		Jonathan Schoop	schoj001
	Kyle Gibson	gibsk002		Starlin Castro	casts001
	Madison Bumgarner	bumgm001		Mark Canha	canhm001
	David Price	pricd001		David Freese	freed001
	Rick Porcello	porcr001		Miguel Sano	sanom001
BOTTOM 5	James Shields	shiej002	BOTTOM 5	Yangervis Solarte	solay001
	Dan Straily	strad003		Kendrys Morales	morak001
	Felix Hernandez	hernf002		Ian Desmond	desmi001
	Matt Harvey	harvm001		Albert Pujols	pujoa001
	Derek Holland	holld003		Chris Davis	davic003

ANALYSIS FOR BATTERS

The metrics available in the data are very specific and singular; such as games played, plate appearances, at bats, runs scored, hits allowed, runs batted - which makes it hard to evaluate and compare overall performance of batters. In order to better understand the metrics and to enable us to compare performance, we derive more complex statistics that are a combination of one or more variables.

At a glance, we can derive a very useful metric known as the Batting Average (BA) which is the ratio of a player's hits and his at-bats essentially measuring how frequently the batter can hit the ball. The higher the batting average, the more frequent a player can hit the ball and hence the better his performance.

As per Jim Albert, a batters ability is measured best when he can create runs, which can be achieved by – "getting on base and advancing runners that are already on base". [2] Considering this, the two metrics that we can use to evaluate batter performance are On-Base Percentage (OBP) and Slugging Percentage (SLG) which are calculated from the data as follows:

- OBP – OBP is the ratio of number of times on-base and number of plate appearances. It is a good measure of the ability to get on base.

$$OBP = \frac{H + BB + HBP}{AB + BB + HBP + SF}$$

- SLG – SLG is the ratio of total bases and number of at-bats. It helps measure batters ability to advance runners.

$$SLG = \frac{Total\ Bases\ (TB)}{At\ Bats\ (AB)}$$

However, there exists a combination of the above-mentioned statistics which is the "On Base plus Slugging" as defined below:

- OPS – It is the sum of OBP and SLG, indicating how often a batter reaches base and how many extra-base hits he accounts for. Similarly, to the batting average, the greater the OPS, the better the performance of the batter.

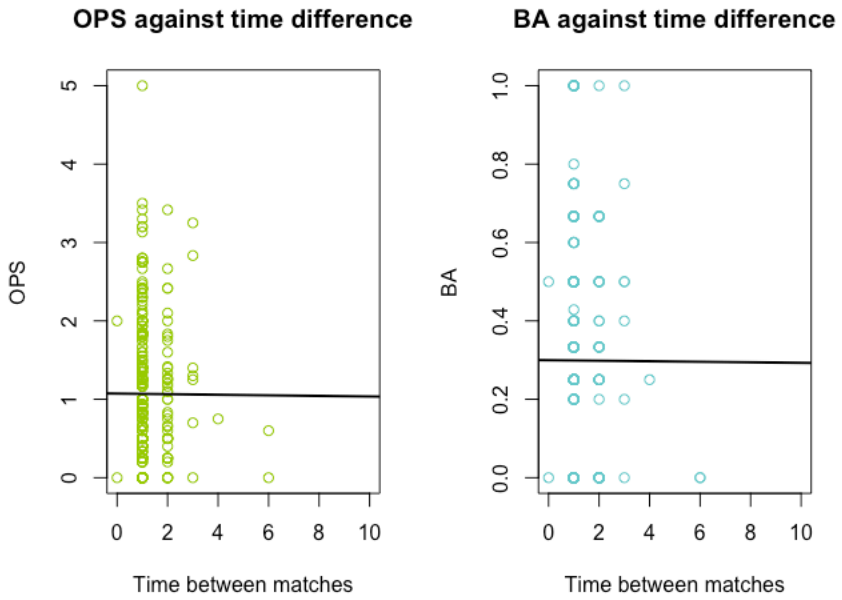
Exploratory Analysis

As outlined earlier, a congested period is defined as a gap of one day or less between games for players. We compute the differences between games for each of the player groups outlined above and plot it against the 2 performance metrics to visualise the trends.

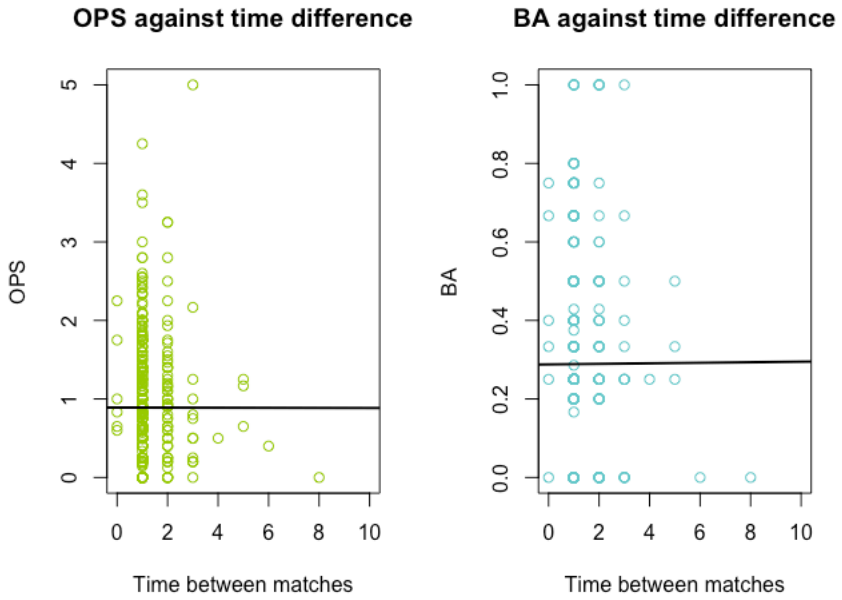
Top 5 Batters

Mike Trout

The graph below shows the trend of OPS and BA of Mike Trout against time between matches. A trendline through the points shows a very slight downward slope that can be negligible and hence seen as relatively flat. It can be inferred that Mike Trout's performance is not significantly affected by time between matches.



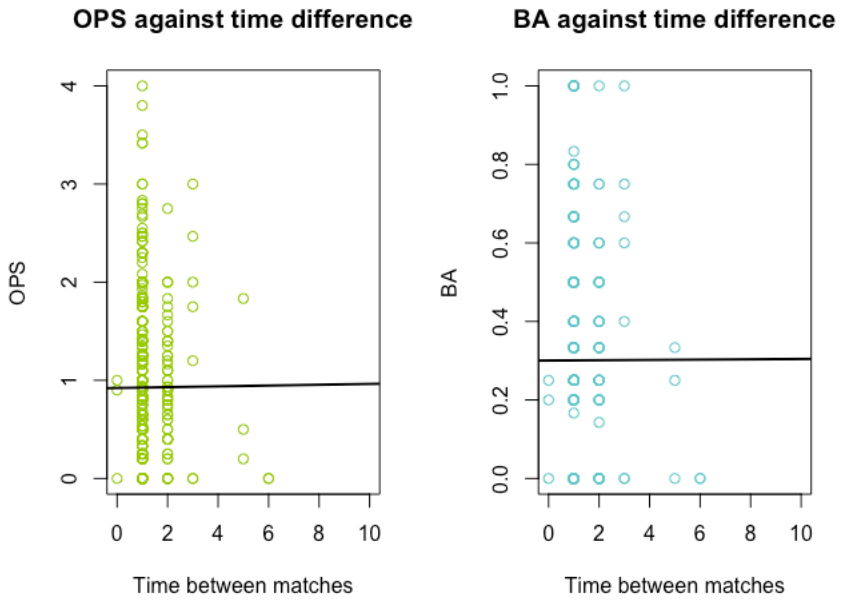
Mookie Betts



The graph above shows the trend of OPS and BA of Mookie Betts against time between matches. A trendline through the points is quite flat and very slightly upward sloping. It can be inferred that Mookie Betts performance is not significantly affected by time between matches.

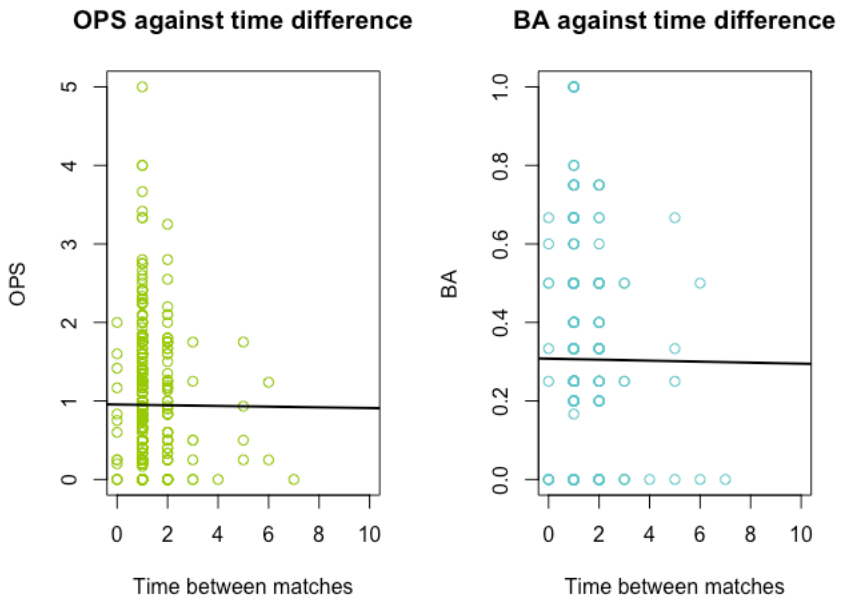
Christian Yelich

The graph below shows the trend of OPS and BA of Christian Yelich against time between matches. At trendline through the points shows a very slight upward slope that can be negligible and hence seen as relatively flat. It can be inferred that Christian Yelich's performance is not significantly affected by time between matches.



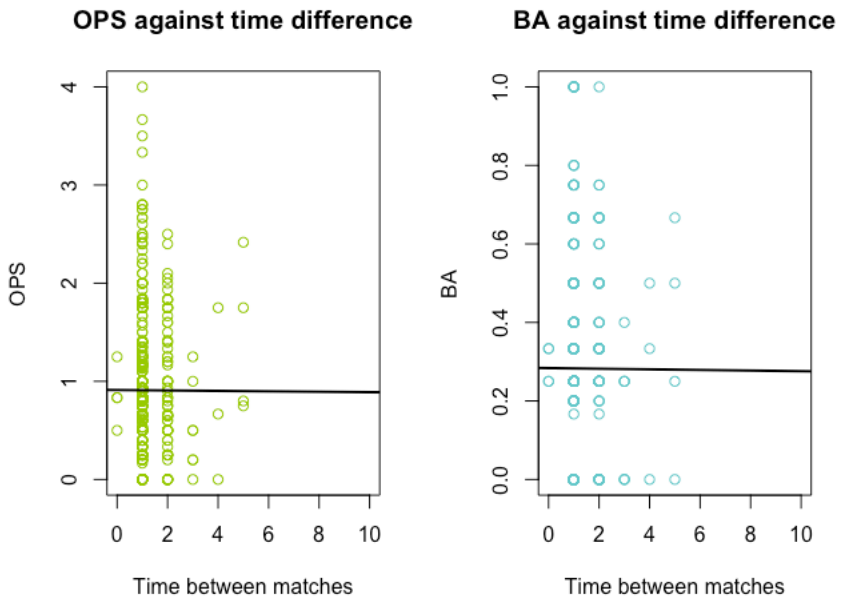
Anthony Rendon

The graph below shows the trend of OPS and BA of Anthony Rendon against time between matches. At trendline through the points shows a very slight downward slope that can be negligible and hence seen as relatively flat. It can be inferred that Anthony Rendon's performance is not significantly affected by time between matches.



Alex Bregman

The graph below shows the trend of OPS and BA of Alex Bregman against time between matches. At trendline through the points shows a very slight downward slope that can be negligible and hence seen as relatively flat. It can be inferred that Alex Bregman's performance is not significantly affected by time between matches.



All of the above graphs give us a better idea of the trend but it is almost impossible to draw any conclusions from the graph. Furthermore, a comparison between performance means in different schedules does not help either.

Comparison of means between congested and non-congested games

	BA_hitter_top	congestedBA	non_congestedsBA
1	Mike Trout	0.3004455	0.2870968
2	Mookie Betts	0.2876171	0.2968801
3	Christian Yelich	0.3006775	0.3037323
4	Anthony Rendonr	0.3129213	0.2748188
5	Alex Bregman	0.2826877	0.2809764

ANALYSIS FOR HITTERS

The metrics available in the data are games played, games started, complete game, shutouts, wins, losses, saves, at bats, earned runs allowed, total bases, walks, strikeouts, sacrifice hits, sacrifice flies, number of pitches and strikes. After proper consideration, from the statistics we have, the best performance metric for pitchers are earned runs (ER) and strikeouts (SO).

ER – ER is the number of runs allowed by the pitcher who is personally accountable for them. It is the runs scored by the batter against the pitcher. Lower the value of ER, the better.

SO – A stikeout is said to occur when a pitcher throws three swinging or looking strikes to the hitter. On batters side, the batter racks up three strikes during a time at bat. It usually means that the batter is out. A strikeout suggests that the pitcher dominated the batter. A higher value of SO for a pitcher is better.

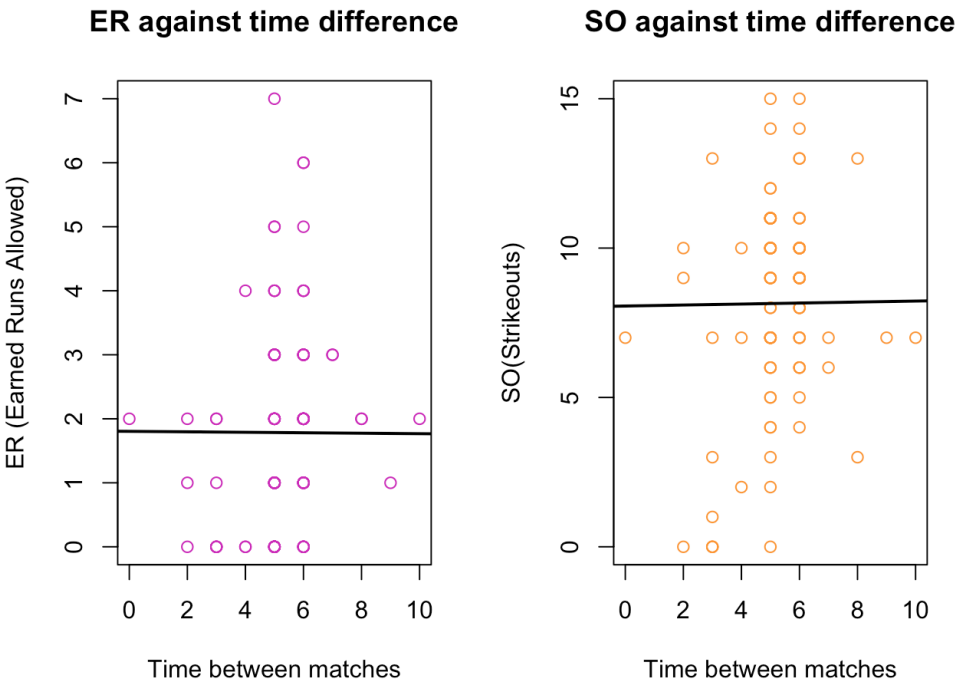
Exploratory Analysis

As outlined earlier, a congested period is considered to eb the one when there is only one or no day between consecutive matches. We calculate the time gap for each of the players outlined above and plot it against the two performance metrics to visualise the trends.

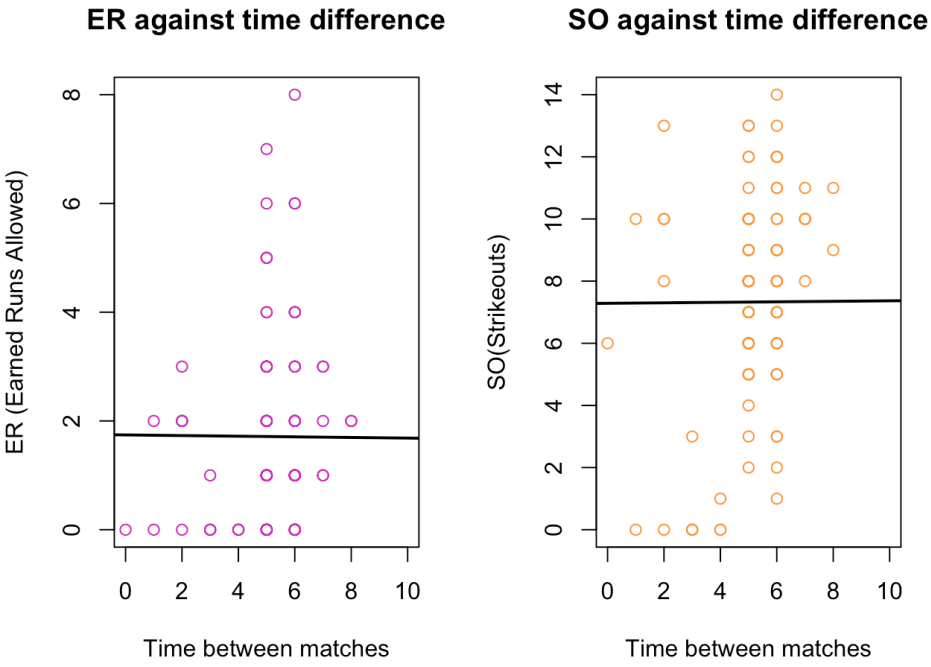
Two of the top 5 Pitchers

Max Scherzer

The graph below shows the trend of ER and So of the pitcher Max Scherzer against time between matches. A trendline through the points show a very slight downward slope that can be negligible and hence seen as relatively flat. It can be inferred that Max Scherzer's performance is not significantly affected by the time between matches.



Mookie Betts

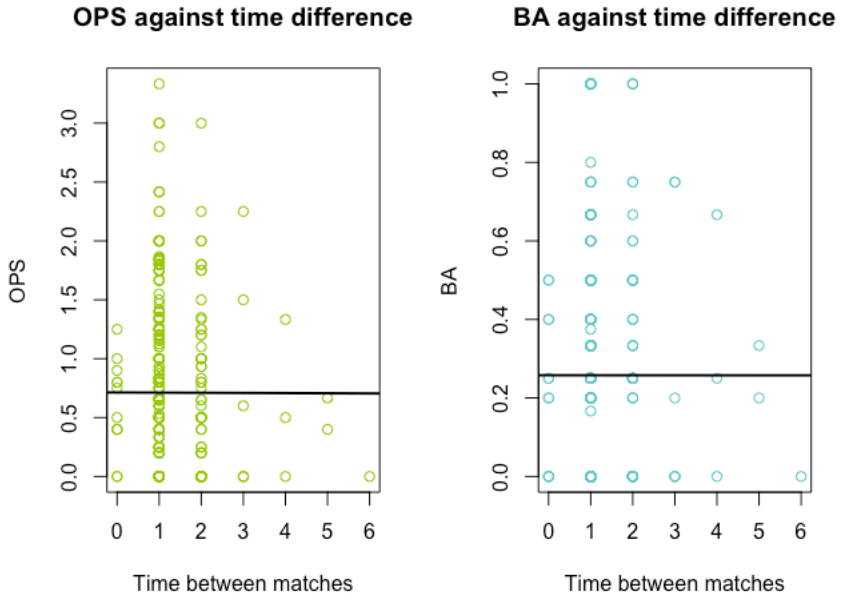


The above graph shows the trend for Mookie Betts against time between matches. The trendline through the points show a very slight downward slope that can be neglected and hence seen as relatively flat. Therefore, it can be inferred that Mookie Betts's performance is not significantly affected by the time between matches.

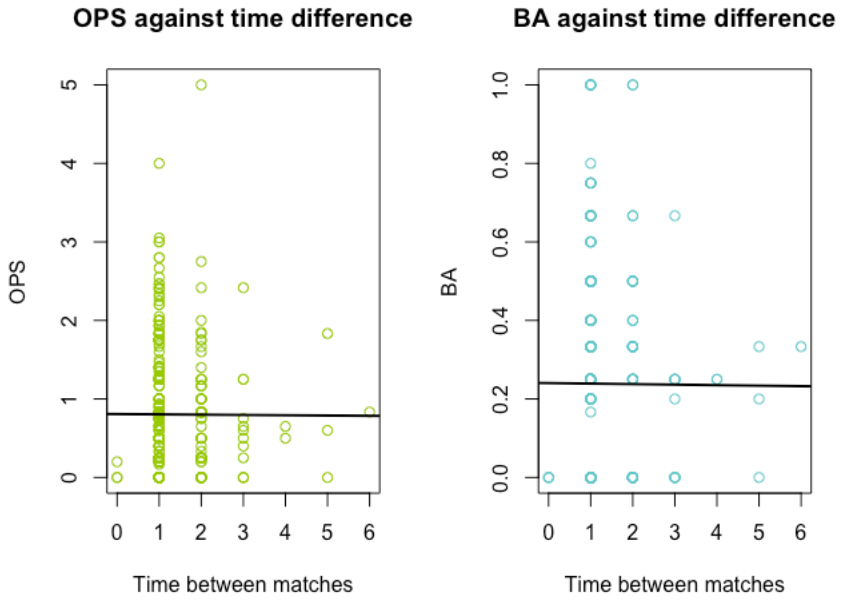
```
      Pitcher_topER congestedER non_congestedER
1      Max Scherzer   2.0000000        1.769231
2      Jacob deGrom   0.6666667        1.722772
3        Chris Sale   0.0000000        2.108696
4 Justin Verlander   2.0000000        2.008621
5       Gerrit Cole   5.0000000        2.152381
> 
```

Middle 10

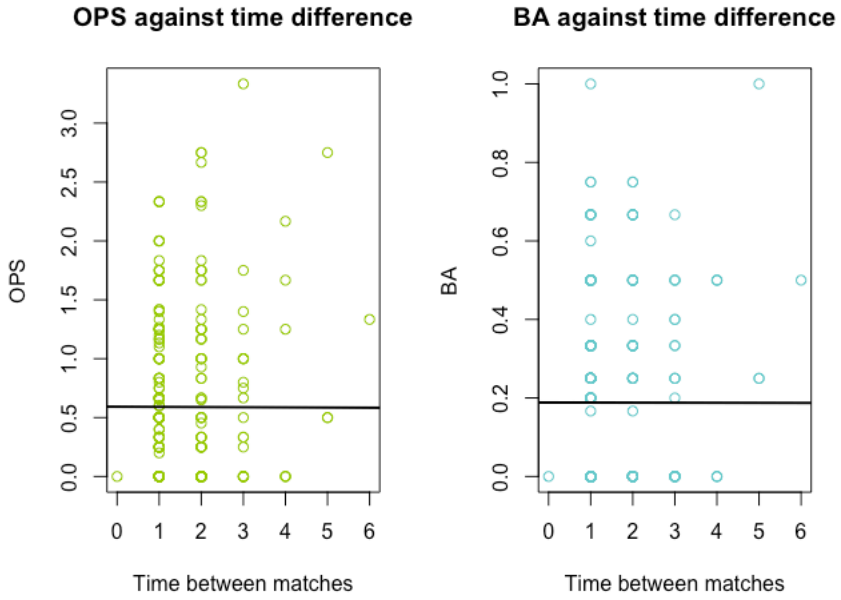
Jonathan Villar



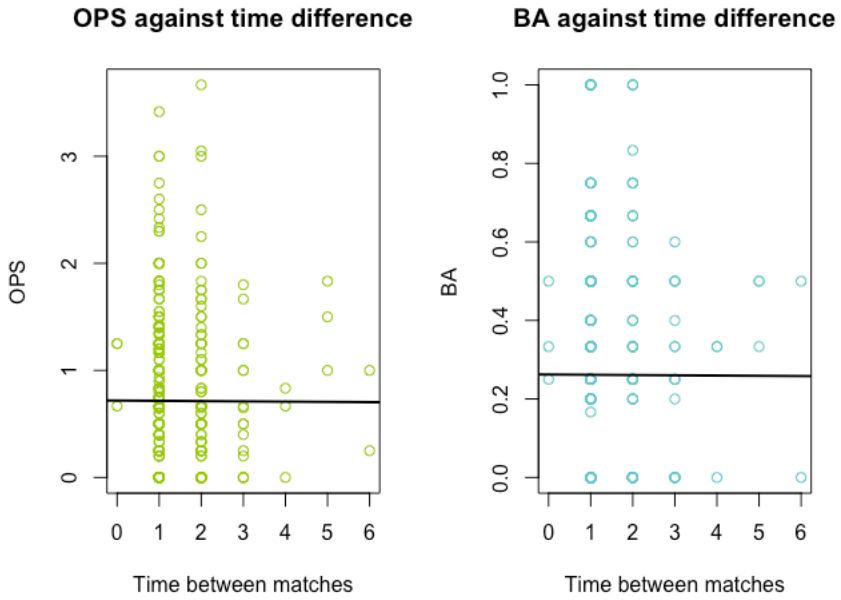
Justin Smoak



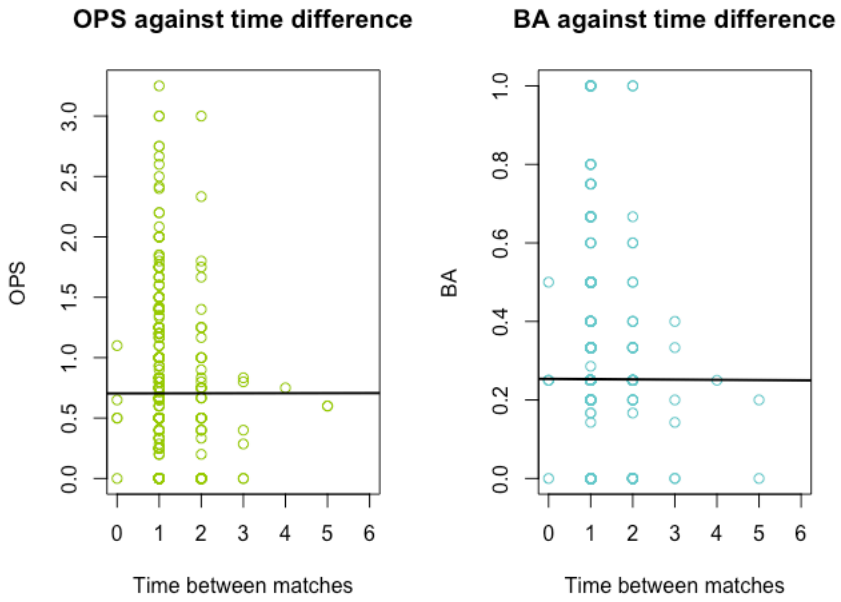
Austin Hedges



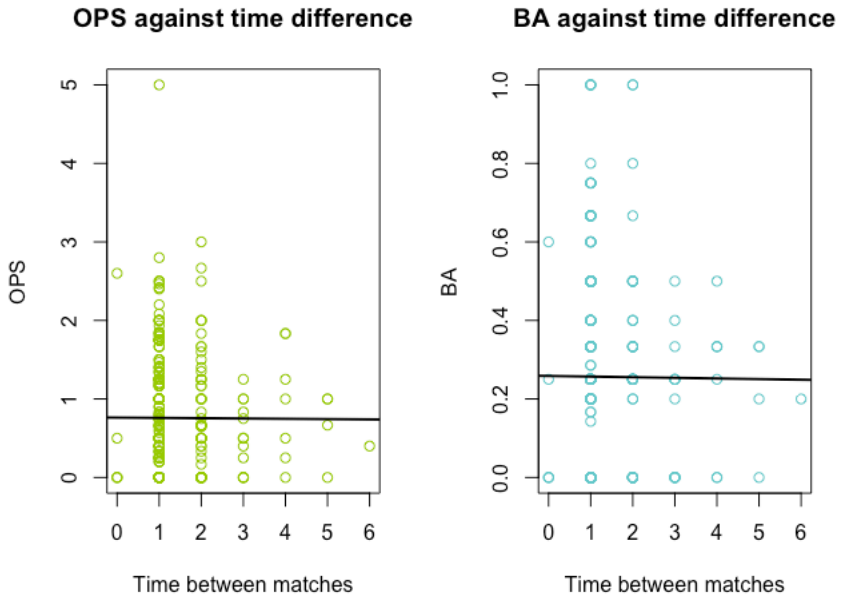
Josh Reddick



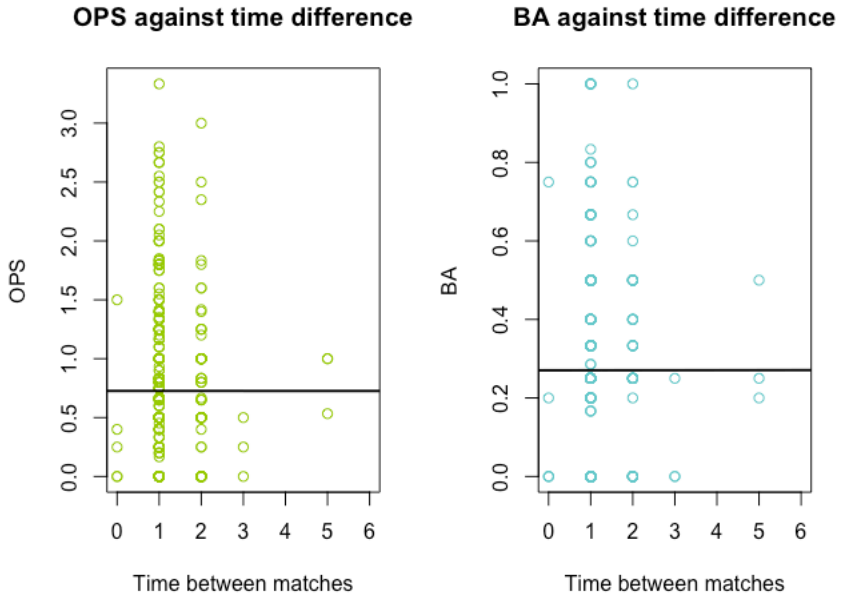
Kevin Pillar



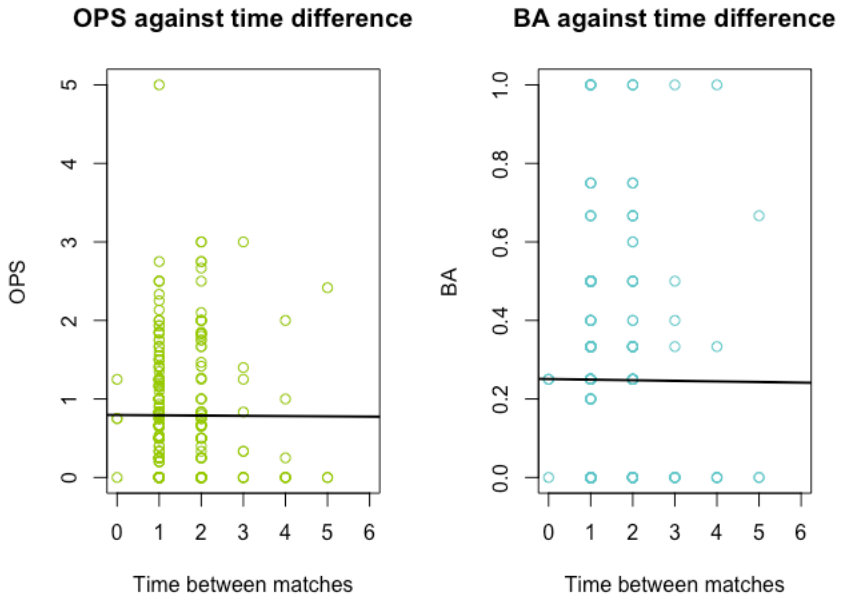
Jonathan Schoop



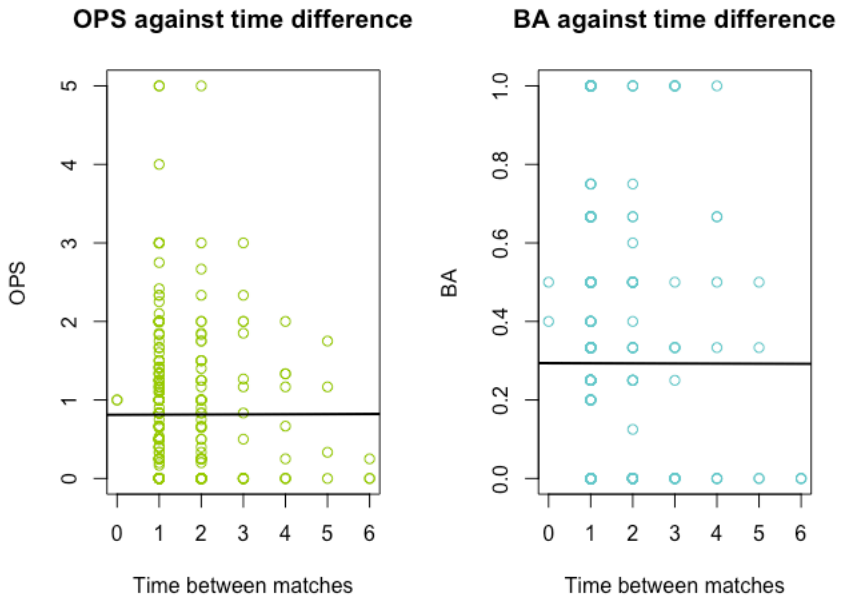
Starlin Castro



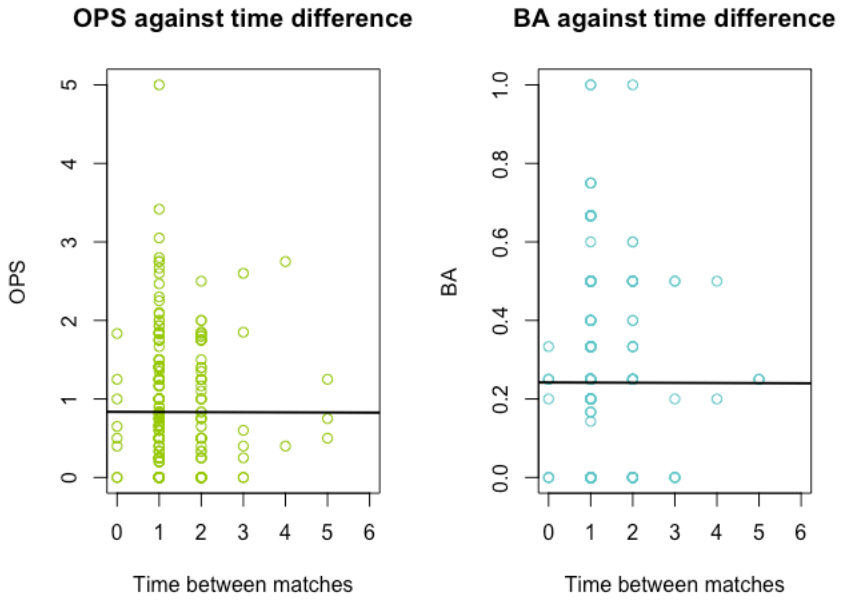
Mark Canha



David Freese



Miguel Sano



Again, all of the trend lines are very similar and hence suggested that the performance of the batters is not significantly affected by time between matches.

STATISTICAL ANALYSES

Top 5 batters

In order to get a conclusive result, we carry out a two tailed t-test with the following hypothesis and 5% level of significance:

Null Hypothesis: There is no effect of congestion on batter performance

Alternative Hypothesis: There is an effect of congestion on batter performance

Null Hypothesis: mean_difference = 0

Alternative Hypothesis: mean_difference is not equal to 0

```
> t.test(congestedBA, non_congestedsBA, paired = TRUE, alternative = "two.sided")

        Paired t-test

data:  congestedBA and non_congestedsBA
t = 0.97833, df = 4, p-value = 0.3833
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.01501394  0.03135180
sample estimates:
mean of the differences
            0.008168933
```

The p-value of the test is 0.3833 which is greater than the 5% level of significance hence we do not reject the null hypothesis.

There is no effect of congestion on batter performance.

A similar test for OPS also yields the same results

```
> t.test(congestedOPS, non_congestedsOPS, paired = TRUE, alternative = "two.sided")

        Paired t-test

data:  congestedOPS and non_congestedsOPS
t = 0.69638, df = 4, p-value = 0.5245
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.05420968  0.09050690
sample estimates:
mean of the differences
            0.01814861
```

Middle 10 batters

In order to get a conclusive result, we carry out a two tailed t-test with the following hypothesis

Null: There is no effect of congestion on batter performance

Alternative: There is an effect of congestion on batter performance

Null: mean_difference = 0

Alternative: mean_difference is not equal to 0

```
        Paired t-test

data:  congestedsBA and non_congestedsBA
t = -0.074913, df = 9, p-value = 0.9419
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.01530505  0.01432386
sample estimates:
mean of the differences
          -0.0004905939
```

```
        Paired t-test

data:  congestedsOPS and non_congestedsOPS
t = -0.070217, df = 9, p-value = 0.9456
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.06389208  0.06004510
sample estimates:
mean of the differences
           -0.001923491
```

The p-value of the tests are 0.9419 and 0.9456 which is greater than the 5% level of significance hence we do not reject the null hypothesis.

There is no effect of congestion on batter OPS and BA.

Bottom 5 batters

```
        Paired t-test

data:  congestedsBA and non_congestedsBA
t = -5.6032, df = 4, p-value = 0.004982
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.028609711 -0.009651189
sample estimates:
mean of the differences
            -0.01913045
```

```
        Paired t-test

data:  congestedsOPS and non_congestedsOPS
t = -6.0674, df = 4, p-value = 0.003727
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.12731017 -0.04737418
sample estimates:
mean of the differences
            -0.08734218
```

A Similar T-test for the bottom players results in p-values less than 5% level of significance hence we reject null hypothesis.

There is an effect of congestion on batter performance.

	Batting Average (BA)		On-base plus Slugging (OPS)	
	p-value	Inference	p-value	Inference
Top 5 batters	0.3833	As p-value is greater than 5% level of significance, we do not reject the null hypothesis. There is no effect of congestion on batter performance.	0.5245	As p-value is greater than 5% level of significance, we do not reject the null hypothesis. There is no effect of congestion on batter performance.

Middle 10 batters	0.9419	As p-value is greater than 5% level of significance, we do not reject the null hypothesis. There is no effect of congestion on batter performance.	0.9456	As p-value is greater than 5% level of significance, we do not reject the null hypothesis. There is no effect of congestion on batter performance.
Bottom 5 batters	0.0037	As p-value is less than 0.05, we reject null hypothesis. Hence, there is effect of congestion on batter performance.	0.004	As p-value is less than 0.05, we reject null hypothesis. Hence, there is effect of congestion on batter performance.

	Earned Runs (ER)		Strikouts (SO)	
	p-value	Inference	p-value	Inference
Top 5 hitters	0.9828	As p-value is greater than 5% level of significance, we do not reject the null hypothesis. There is no effect of congestion on pitcher performance.	0.2605	As p-value is greater than 5% level of significance, we do not reject the null hypothesis. There is no effect of congestion on batter performance.
Middle 10 hitters	0.0069	As p-value is less than 0.05, we reject null hypothesis. Hence, there is effect of congestion on pitcher performance.	0.2042	As p-value is greater than 5% level of significance, we do not reject the null hypothesis. There is no effect of congestion on batter performance.
Bottom 5 hitters	0.0181	As p-value is less than 0.05, we reject null hypothesis. Hence, there is effect of congestion on pitcher performance.	0.8162	As p-value is greater than 5% level of significance, we do not reject the null hypothesis.

				There is no effect of congestion on batter performance.

Limitations

- Player performance can be affected by numerous other factors such as the strength of the opposition team, weather, team morale etc. It is impossible to isolate and understand the effect of one single factor on player performance. Hence for the purposes for this study, we assume that all other factors remain constant.

Conclusion

The statistical tests for the top and middle order batters showed that there is no effect of congestion on batter performance, however the test for the bottom order batters showed that there is an effect of congestion on their performance. This shows that the higher order batters are used to the congested nature of baseball tournament schedules and are unaffected by it.

The statistical tests for the top and middle order pitchers showed that there is no effect of congestion on pitcher performance, however the test for the bottom order pitchers showed that there is an effect of congestion on their performance. This shows that the higher order pitchers are used to the congested nature of baseball tournament schedules and are unaffected by it.

REFLECTION

This analysis or study is the first of its kind investigating the effect of congested match schedules on player performance.

This study would help the organisers in deciding whether they need to schedule the match calendar in a different way or not to avoid back to back matches for players.

Future studies should analyse how performance varies in congested period after considering the number of years the player has been an active Baseball player, that is it should consider the career length of the participant player. Moreover, future studies can also analyse/determine the performance after considering the amount of time the player was on field and group accordingly and then analyse the effect of consecutive matches.

References

["FanGraphs," [Online]. Available: https://www.fangraphs.com/leaders.aspx?pos=all&stats=bat&lg=all&qual=y&type=8&season=2019&month=0&season1=2017&ind=0&team=0&rost=0&age=0&filter=&players=0&startdate=&enddate=.

[J. Albert, *Using Play-by-Play Baseball Data to Develop a Better Measure of Batting Performance,* 2001.

[V. J. Sabesan, B. Prey, R. Smith, D. Lombardo, W. Borroto and J. Whaley, "Concussion rates and effects on player performance in Major League Baseball players," *Open Access J Sports Med,* vol. 13, 2018.

["Retrosheet Organizational Description," 25 07 2009. [Online]. Available: https://www.retrosheet.org/about.htm.

[K. Kaan and M. Stephan, "Machine Learning Applications in Baseball: A Systematic Literature Review," *Original Articles,* pp. 745-763, 26 Feb 2018.

[[Online]. Available: https://sabr.org/.

["Risk factor analysis for injuries in football players. Possibilities for a prevention program," *Am J Sports Med,* 2000.

[M. W. M. H. J Ekstrand, *A congested football calendar and the wellbeing of players: correlation between match exposure of European footballers before the World Cup 2002 and their injuries and performances during that World Cup,* Br J Sports Med, 2004.

[L. G. D. carling, *Are physical performance and injury risk in a professional soccer team in match-play affected over a prolonged period of fixture congestion?,* 2012.

[T. J. C. E. S. S. W B Kibler, *Musculoskeletal adaptations and injuries due to overtraining,* 1992.

[C. Alexandra, *The effects of a congested fixture period on physical performance, technical activity and injury rate during matches in a professional soccer team.*

[Jacob and Charles, *Wednesday, September 26, 2018 7:35 AM–9:00 AM ePosters: P168. Professional baseball player performance outcomes and return to play following back injury.*

[Salvotare, *Career Length and Performance Among Professional Baseball Players Returning to Play After Hip Arthroscopy.*

Appendix 1: Supplementary Information

```
data2017 <- read.csv("/Users/yashttm/Downloads/playing-2017.csv")

data2018 <- read.csv("/Users/yashttm/Downloads/playing-2018.csv")

data2019 <- read.csv("/Users/yashttm/Downloads/playing-2019.csv")

data20172018 <- rbind(data2017, data2018)

data <- rbind(data20172018, data2019)


head(data)



## BATTERS >>>>>>>>>>>>>>>>>>>>

## batting average (BA, or AVG above) = H/AB

data$BA<-data$B_H/data$B_AB

summary(data$BA)


## on-base percentage (OBP) = (H + BB + HBP)/ (AB + BB + HBP + SF)

data$OBP<-(data$B_H+data$B_BB+data$B_HP)/(data$B_AB+data$B_BB+data$B_HP+data$P_SF)


## slugging (SLG) = (TB/AB)
```

```
data$SLG<-data$B_TB/data$B_AB

##On-base plus slugging (OPS) = OBP+SLG
data$OPS<-data$OBP+data$SLG

#Create separate batting analysis data
colnames(data)
bat<-data[-c(2, 5:8, 10, 12:13 , 23:175)]
colnames(bat)

###################### ANALYSIS FOR TOP 5 BATTERS ##########################
# Mike Trout - Batter T1
troum001_df <- subset(bat, person.key=="troum001")
troum001_df$game.date<-as.Date(troum001_df$game.date)
x<-diff(troum001_df$game.date)
x<-append(x,0,after = 0)
troum001_df$timediff<-x

par(mfrow=c(1,2))
miketroutOPS<-lm(OPS ~ timediff,data = troum001_df)
plot(troum001_df$timediff, troum001_df$OPS,xlab = "Time between matches",
    ylab = "OPS",main = "OPS against time difference",col=  "#99CC00", xlim=c(0,10)
)
abline(a = coef(miketroutOPS)[1] , b = coef(miketroutOPS)[2] , lwd = 2)

miketroutBA<-lm(BA ~ timediff,data = troum001_df)
plot(troum001_df$timediff, troum001_df$BA,xlab = "Time between matches",
```

```
    ylab = "BA",main = "BA against time difference",col = "#66CCCC", xlim=c(0,10)
)
abline(a = coef(miketroutBA)[1] , b = coef(miketroutBA)[2] , lwd = 2)



# Mookie Betts -  Batter T2
bettm001_df <- subset(bat, person.key=="bettm001")
bettm001_df$game.date<-as.Date(bettm001_df$game.date)
x<-diff(bettm001_df$game.date)
x<-append(x,0,after = 0)
bettm001_df$timediff<-x

par(mfrow=c(1,2))
mookiebettsOPS<-lm(OPS ~ timediff,data = bettm001_df)
plot(bettm001_df$timediff, bettm001_df$OPS,xlab = "Time between matches",
    ylab = "OPS",main = "OPS against time difference", col = "#99CC00", xlim=c(0,10)
)
abline(a = coef(mookiebettsOPS)[1] , b = coef(mookiebettsOPS)[2] , lwd = 2)

mookiebettsBA<-lm(BA ~ timediff,data = bettm001_df)
plot(bettm001_df$timediff, bettm001_df$BA,xlab = "Time between matches",
    ylab = "BA",main = "BA against time difference",col = "#66CCCC", xlim=c(0,10)
)
abline(a = coef(mookiebettsBA)[1] , b = coef(mookiebettsBA)[2] , lwd = 2)
```

```
# Christian Yelich - Batter T3
yelic001_df <- subset(bat, person.key=="yelic001")
yelic001_df$game.date<-as.Date(yelic001_df$game.date)
x<-diff(yelic001_df$game.date)
x<-append(x,0,after = 0)
yelic001_df$timediff<-x

par(mfrow=c(1,2))
christianyelichOPS<-lm(OPS ~ timediff,data = yelic001_df)
plot(yelic001_df$timediff, yelic001_df$OPS,xlab = "Time between matches",
    ylab = "OPS",main = "OPS against time difference",col = "#99CC00", xlim=c(0,10)
)
abline(a = coef(christianyelichOPS)[1] , b = coef(christianyelichOPS)[2] , lwd = 2)

christianyelichBA<-lm(BA ~ timediff,data = yelic001_df)
plot(yelic001_df$timediff, yelic001_df$BA,xlab = "Time between matches",
    ylab = "BA",main = "BA against time difference",col = "#66CCCC", xlim=c(0,10)
)
abline(a = coef(christianyelichBA)[1] , b = coef(christianyelichBA)[2] , lwd = 2)


# Anthony Rendon - Batter T4
renda001_df <- subset(bat, person.key=="renda001")
renda001_df$game.date<-as.Date(renda001_df$game.date)
x<-diff(renda001_df$game.date)
x<-append(x,0,after = 0)
```

```
renda001_df$timediff<-x


par(mfrow=c(1,2))
anthonyrendonOPS<-lm(OPS ~ timediff,data = renda001_df)
plot(renda001_df$timediff, renda001_df$OPS,
    xlab = "Time between matches",
    ylab = "OPS",
    main = "OPS against time difference",
    col = "#99CC00", xlim=c(0,10)
)
abline(a = coef(anthonyrendonOPS)[1] , b = coef(anthonyrendonOPS)[2] , lwd = 2)


anthonyrendonBA<-lm(BA ~ timediff,data = renda001_df)
plot(renda001_df$timediff, renda001_df$BA,
    xlab = "Time between matches",
    ylab = "BA",
    main = "BA against time difference",
    col = "#66CCCC", xlim=c(0,10)
)
abline(a = coef(anthonyrendonBA)[1] , b = coef(anthonyrendonBA)[2] , lwd = 2)



# Alex Bregman - Batter T5
brega001_df <- subset(bat, person.key=="brega001")
str(brega001_df)
brega001_df$game.date<-as.Date(brega001_df$game.date)
x<-diff(brega001_df$game.date)
```

```
x<-append(x,0,after = 0)
brega001_df$timediff<-x


par(mfrow=c(1,2))
alexbregmanOPS<-lm(OPS ~ timediff,data = brega001_df)
plot(brega001_df$timediff, brega001_df$OPS,
    xlab = "Time between matches",
    ylab = "OPS",
    main = "OPS against time difference",
    col = "#99CC00", xlim=c(0,10)
)
abline(a = coef(alexbregmanOPS)[1] , b = coef(alexbregmanOPS)[2] , lwd = 2)


alexbregmanBA<-lm(BA ~ timediff,data = brega001_df)
plot(brega001_df$timediff, brega001_df$BA,
    xlab = "Time between matches",
    ylab = "BA",
    main = "BA against time difference",
    col=  "#66CCCC", xlim=c(0,10)
)
abline(a = coef(alexbregmanBA)[1] , b = coef(alexbregmanBA)[2] , lwd = 2)




# Comparison of top 5 batters
#BA
congestedBA<-c(
```

```
  mean(troum001_df$BA[troum001_df$timediff<=1],na.rm = TRUE),
  mean(bettm001_df$BA[bettm001_df$timediff<=1],na.rm = TRUE),
  mean(yelic001_df$BA[yelic001_df$timediff<=1],na.rm = TRUE),
  mean(renda001_df$BA[renda001_df$timediff<=1],na.rm = TRUE),
  mean(brega001_df$BA[brega001_df$timediff<=1],na.rm = TRUE)
)
non_congestedsBA<-c(
  mean(troum001_df$BA[troum001_df$timediff>=2],na.rm = TRUE),
  mean(bettm001_df$BA[bettm001_df$timediff>=2],na.rm = TRUE),
  mean(yelic001_df$BA[yelic001_df$timediff>=2],na.rm = TRUE),
  mean(renda001_df$BA[renda001_df$timediff>=2],na.rm = TRUE),
  mean(brega001_df$BA[brega001_df$timediff>=2],na.rm = TRUE)
)

BA_hitter_top<-c(
  "Mike Trout","Mookie Betts","Christian Yelich","Anthony Rendonr","Alex Bregman")

final_hittertopBA<-data.frame(BA_hitter_top, congestedBA,non_congestedsBA)

## Hypothesis Testing for BA
print(final_hittertopBA)


t.test(congestedBA, non_congestedsBA, paired = TRUE, alternative = "two.sided")

#proof
final_hittertopBA$difference<-final_hittertopBA$non_congestedsBA-
final_hittertopBA$congestedBA
```

```
print(final_hittertopBA)



#OPS
congestedOPS<-c(
  mean(troum001_df$OPS[troum001_df$timediff<=1],na.rm = TRUE),
  mean(bettm001_df$OPS[bettm001_df$timediff<=1],na.rm = TRUE),
  mean(yelic001_df$OPS[yelic001_df$timediff<=1],na.rm = TRUE),
  mean(renda001_df$OPS[renda001_df$timediff<=1],na.rm = TRUE),
  mean(brega001_df$OPS[brega001_df$timediff<=1],na.rm = TRUE)
)
non_congestedsOPS<-c(
  mean(troum001_df$OPS[troum001_df$timediff>=2],na.rm = TRUE),
  mean(bettm001_df$OPS[bettm001_df$timediff>=2],na.rm = TRUE),
  mean(yelic001_df$OPS[yelic001_df$timediff>=2],na.rm = TRUE),
  mean(renda001_df$OPS[renda001_df$timediff>=2],na.rm = TRUE),
  mean(brega001_df$OPS[brega001_df$timediff>=2],na.rm = TRUE)
)

OPS_Hitter_top<-c("Mike Trout","Mookie Betts","Christian Yelich","Anthony Rendonr","Alex Bregman")

OPS_final_hittertop<-data.frame(OPS_Hitter_top,congestedOPS,non_congestedsOPS)
print(OPS_final_hittertop)

## Hypothesis Testing

#Null: There is an effect of congestion on batter performance
```

```
#Alternative: There is no effect of congestion on batter performance

#Null: mean_difference = 0

#Alternative: mean_difference is not equal to 0


t.test(congestedOPS, non_congestedsOPS, paired = TRUE, alternative = "two.sided")


#proof

OPS_final_hittertop$OPS_difference<- OPS_final_hittertop$non_congestedsOPS-

  OPS_final_hittertop$congestedsOPS

print(OPS_final_hittertop)









###################### ANALYSIS FOR MIDDLE 10 BATTERS ##########################


#Batter M1 - Jonathan Villar

villj001_df <- subset(bat, person.key =="villj001")   #create dataframe

str(villj001_df)

villj001_df$game.date<-as.Date(villj001_df$game.date)

x<-diff(villj001_df$game.date)

x<-append(x,0,after = 0)
```

```
villj001_df$timediff<-x


par(mfrow = c(1, 2))
villarOPS<-lm(OPS ~ timediff,data = villj001_df)
plot(villj001_df$timediff, villj001_df$OPS,
    xlab = "Time between matches",
    ylab = "OPS",
    main = "OPS against time difference",
    col ="#99CC00", xlim=c(0,6)
)
abline(a = coef(villarOPS)[1] , b = coef(villarOPS)[2] , lwd = 2)



villarBA<-lm(BA ~ timediff,data = villj001_df)
plot(villj001_df$timediff, villj001_df$BA,
    xlab = "Time between matches",
    ylab = "BA",
    main = "BA against time difference",
    col ="#66CCCC", xlim=c(0,6)
)
abline(a = coef(villarBA)[1] , b = coef(villarBA)[2] , lwd = 2)



# Batter M2 - Justin Smoak
smoaj001_df <- filter(bat, person.key=="smoaj001")   #create dataframe
str(smoaj001_df)
smoaj001_df$game.date<-as.Date(smoaj001_df$game.date)
```

```
x<-diff(smoaj001_df$game.date)

x<-append(x,0,after = 0)

smoaj001_df$timediff<-x


par(mfrow = c(1, 2))

justinOPS<-lm(OPS ~ timediff,data = smoaj001_df)

plot(smoaj001_df$timediff, smoaj001_df$OPS,

    xlab = "Time between matches",

    ylab = "OPS",

    main = "OPS against time difference",

    col ="#99CC00", xlim=c(0,6)

)

abline(a = coef(justinOPS)[1] , b = coef(justinOPS)[2] , lwd = 2)



justinBA<-lm(BA ~ timediff,data = smoaj001_df)

plot(smoaj001_df$timediff, smoaj001_df$BA,

    xlab = "Time between matches",

    ylab = "BA",

    main = "BA against time difference",

    col ="#66CCCC", xlim=c(0,6)

)

abline(a = coef(justinBA)[1] , b = coef(justinBA)[2] , lwd = 2)



# Batter M3 - Austin Hedges
```

```
hedga001_df <- filter(bat, person.key=="hedga001")   #create dataframe
str(hedga001_df)
hedga001_df$game.date<-as.Date(hedga001_df$game.date)
x<-diff(hedga001_df$game.date)
x<-append(x,0,after = 0)
hedga001_df$timediff<-x


par(mfrow = c(1, 2))
hedgesOPS<-lm(OPS ~ timediff,data = hedga001_df)
plot(hedga001_df$timediff, hedga001_df$OPS,
    xlab = "Time between matches",
    ylab = "OPS",
    main = "OPS against time difference",
    col ="#99CC00", xlim=c(0,6)
)
abline(a = coef(hedgesOPS)[1] , b = coef(hedgesOPS)[2] , lwd = 2)



hedgesBA<-lm(BA ~ timediff,data = hedga001_df)
plot(hedga001_df$timediff, hedga001_df$BA,
    xlab = "Time between matches",
    ylab = "BA",
    main = "BA against time difference",
    col ="#66CCCC", xlim=c(0,6)
)
abline(a = coef(hedgesBA)[1] , b = coef(hedgesBA)[2] , lwd = 2)
```

```
# Batter M4 - Josh Reddick
reddj001_df <- filter(bat, person.key=="reddj001")   #create dataframe
str(reddj001_df)
reddj001_df$game.date<-as.Date(reddj001_df$game.date)
x<-diff(reddj001_df$game.date)
x<-append(x,0,after = 0)
reddj001_df$timediff<-x

par(mfrow = c(1, 2))
reddickOPS<-lm(OPS ~ timediff,data = reddj001_df)
plot(reddj001_df$timediff, reddj001_df$OPS,
    xlab = "Time between matches",
    ylab = "OPS",
    main = "OPS against time difference",
    col ="#99CC00", xlim=c(0,6)
)
abline(a = coef(reddickOPS)[1] , b = coef(reddickOPS)[2] , lwd = 2)


reddickBA<-lm(BA ~ timediff,data = reddj001_df)
plot(reddj001_df$timediff, reddj001_df$BA,
    xlab = "Time between matches",
    ylab = "BA",
    main = "BA against time difference",
    col ="#66CCCC", xlim=c(0,6)
)
```

```
abline(a = coef(reddickBA)[1] , b = coef(reddickBA)[2] , lwd = 2)



# Batter M5 - Kevin Pillar
pillk001_df <- filter(bat, person.key=="pillk001")   #create dataframe
str(pillk001_df)
pillk001_df$game.date<-as.Date(pillk001_df$game.date)
x<-diff(pillk001_df$game.date)
x<-append(x,0,after = 0)
pillk001_df$timediff<-x

par(mfrow = c(1, 2))
kevinpillarOPS<-lm(OPS ~ timediff,data = pillk001_df)
plot(pillk001_df$timediff, pillk001_df$OPS,
    xlab = "Time between matches",
    ylab = "OPS",
    main = "OPS against time difference",
    col ="#99CC00", xlim=c(0,6)
)
abline(a = coef(kevinpillarOPS)[1] , b = coef(kevinpillarOPS)[2] , lwd = 2)



kevinpillarBA<-lm(BA ~ timediff,data = pillk001_df)
plot(pillk001_df$timediff, pillk001_df$BA,
    xlab = "Time between matches",
    ylab = "BA",
```

```
    main = "BA against time difference",
    col ="#66CCCC", xlim=c(0,6)
)
abline(a = coef(kevinpillarBA)[1] , b = coef(kevinpillarBA)[2] , lwd = 2)



#Batter M6 - Jonathan Schoop
schoj001_df <- filter(bat, person.key=="schoj001")   #create dataframe
str(schoj001_df)
schoj001_df$game.date<-as.Date(schoj001_df$game.date)
x<-diff(schoj001_df$game.date)
x<-append(x,0,after = 0)
schoj001_df$timediff<-x

par(mfrow = c(1, 2))
schoopOPS<-lm(OPS ~ timediff,data = schoj001_df)
plot(schoj001_df$timediff, schoj001_df$OPS,
    xlab = "Time between matches",
    ylab = "OPS",
    main = "OPS against time difference",
    col ="#99CC00", xlim=c(0,6)
)
abline(a = coef(schoopOPS)[1] , b = coef(schoopOPS)[2] , lwd = 2)



schoopBA<-lm(BA ~ timediff,data = schoj001_df)
plot(schoj001_df$timediff, schoj001_df$BA,
```

```
    xlab = "Time between matches",
    ylab = "BA",
    main = "BA against time difference",
    col ="#66CCCC", xlim=c(0,6)
)
abline(a = coef(schoopBA)[1] , b = coef(schoopBA)[2] , lwd = 2)



#Batter M7- Starlin Castro


casts001_df <- filter(bat, person.key=="casts001")   #create dataframe
str(casts001_df)
casts001_df$game.date<-as.Date(casts001_df$game.date)
x<-diff(casts001_df$game.date)
x<-append(x,0,after = 0)
casts001_df$timediff<-x


par(mfrow = c(1, 2))
castroOPS<-lm(OPS ~ timediff,data = casts001_df)
plot(casts001_df$timediff, casts001_df$OPS,
    xlab = "Time between matches",
    ylab = "OPS",
    main = "OPS against time difference",
    col ="#99CC00", xlim=c(0,6)
)
abline(a = coef(castroOPS)[1] , b = coef(castroOPS)[2] , lwd = 2)
```

```
castroBA<-lm(BA ~ timediff,data = casts001_df)
plot(casts001_df$timediff, casts001_df$BA,
    xlab = "Time between matches",
    ylab = "BA",
    main = "BA against time difference",
    col ="#66CCCC", xlim=c(0,6)
)
abline(a = coef(castroBA)[1] , b = coef(castroBA)[2] , lwd = 2)



#Batter M8 - Mark Canha
canhm001_df <- filter(bat, person.key=="canhm001")   #create dataframe
str(canhm001_df)
canhm001_df$game.date<-as.Date(canhm001_df$game.date)
x<-diff(canhm001_df$game.date)
x<-append(x,0,after = 0)
canhm001_df$timediff<-x

par(mfrow = c(1, 2))
canhamarkOPS<-lm(OPS ~ timediff,data = canhm001_df)
plot(canhm001_df$timediff, canhm001_df$OPS,
    xlab = "Time between matches",
    ylab = "OPS",
    main = "OPS against time difference",
```

```
    col ="#99CC00", xlim=c(0,6)
)
abline(a = coef(canhamarkOPS)[1] , b = coef(canhamarkOPS)[2] , lwd = 2)



canhamarkBA<-lm(BA ~ timediff,data = canhm001_df)
plot(canhm001_df$timediff, canhm001_df$BA,
    xlab = "Time between matches",
    ylab = "BA",
    main = "BA against time difference",
    col ="#66CCCC", xlim=c(0,6)
)
abline(a = coef(canhamarkBA)[1] , b = coef(canhamarkBA)[2] , lwd = 2)



#Batter M9 - David Freese
freed001_df <- filter(bat, person.key=="freed001")   #create dataframe
str(freed001_df)
freed001_df$game.date<-as.Date(freed001_df$game.date)
x<-diff(freed001_df$game.date)
x<-append(x,0,after = 0)
freed001_df$timediff<-x

par(mfrow = c(1, 2))
freesedavidOPS<-lm(OPS ~ timediff,data = freed001_df)
plot(freed001_df$timediff, freed001_df$OPS,
```

```
    xlab = "Time between matches",
    ylab = "OPS",
    main = "OPS against time difference",
    col ="#99CC00", xlim=c(0,6)
)
abline(a = coef(freesedavidOPS)[1] , b = coef(freesedavidOPS)[2] , lwd = 2)



freesedavidBA<-lm(BA ~ timediff,data = freed001_df)
plot(freed001_df$timediff, freed001_df$BA,
    xlab = "Time between matches",
    ylab = "BA",
    main = "BA against time difference",
    col ="#66CCCC", xlim=c(0,6)
)
abline(a = coef(freesedavidBA)[1] , b = coef(freesedavidBA)[2] , lwd = 2)



#Batter M10 - Miguel Sano
sanom001_df <- filter(bat, person.key=="sanom001")   #create dataframe
str(sanom001_df)
sanom001_df$game.date<-as.Date(sanom001_df$game.date)
x<-diff(sanom001_df$game.date)
x<-append(x,0,after = 0)
sanom001_df$timediff<-x
```

```
par(mfrow = c(1, 2))
sanomiguelOPS<-lm(OPS ~ timediff,data = sanom001_df)
plot(sanom001_df$timediff, sanom001_df$OPS,
    xlab = "Time between matches",
    ylab = "OPS",
    main = "OPS against time difference",
    col ="#99CC00", xlim=c(0,6)
)
abline(a = coef(sanomiguelOPS)[1] , b = coef(sanomiguelOPS)[2] , lwd = 2)



sanomiguelBA<-lm(BA ~ timediff,data = sanom001_df)
plot(sanom001_df$timediff, sanom001_df$BA,
    xlab = "Time between matches",
    ylab = "BA",
    main = "BA against time difference",
    col ="#66CCCC", xlim=c(0,6)
)
abline(a = coef(sanomiguelBA)[1] , b = coef(sanomiguelBA)[2] , lwd = 2)



## Comparison for middle 10 batters
#BA

congestedsBA<-c(
  mean(villj001_df$BA[villj001_df$timediff<=1],na.rm = TRUE),
  mean(smoaj001_df$BA[smoaj001_df$timediff<=1],na.rm = TRUE),
```

```
  mean(hedga001_df$BA[hedga001_df$timediff<=1],na.rm = TRUE),
  mean(reddj001_df$BA[reddj001_df$timediff<=1],na.rm = TRUE),
  mean(pillk001_df$BA[pillk001_df$timediff<=1],na.rm = TRUE),
  mean(schoj001_df$BA[schoj001_df$timediff<=1],na.rm = TRUE),
  mean(casts001_df$BA[casts001_df$timediff<=1],na.rm = TRUE),
  mean(canhm001_df$BA[canhm001_df$timediff<=1],na.rm = TRUE),
  mean(freed001_df$BA[freed001_df$timediff<=1],na.rm = TRUE),
  mean(sanom001_df$BA[sanom001_df$timediff<=1],na.rm = TRUE)
)
non_congestedsBA<-c(
  mean(villj001_df$BA[villj001_df$timediff>=2],na.rm = TRUE),
  mean(smoaj001_df$BA[smoaj001_df$timediff>=2],na.rm = TRUE),
  mean(hedga001_df$BA[hedga001_df$timediff>=2],na.rm = TRUE),
  mean(reddj001_df$BA[reddj001_df$timediff>=2],na.rm = TRUE),
  mean(pillk001_df$BA[pillk001_df$timediff>=2],na.rm = TRUE),
  mean(schoj001_df$BA[schoj001_df$timediff<=2],na.rm = TRUE),
  mean(casts001_df$BA[casts001_df$timediff<=2],na.rm = TRUE),
  mean(canhm001_df$BA[canhm001_df$timediff<=2],na.rm = TRUE),
  mean(freed001_df$BA[freed001_df$timediff<=2],na.rm = TRUE),
  mean(sanom001_df$BA[sanom001_df$timediff<=2],na.rm = TRUE)
)

Hitter_middleBA<-c("Jonathan Villar", "Justin Smoak", "Austin Hedges", "Josh Reddick",
                   "Kevin Pillar", "Jonathan Schoop", "Starlin Castro", "Mark Canha",
                   "David Freese", "Miguel Sano")
final_hittermiddle<-data.frame(Hitter_middleBA, congestedsBA,non_congestedsBA)
```

```
## Hypothesis Testing for BA

print(final_hittermiddle)`


#Null: There is an effect of congestion on batter performance

#Alternative: There is no effect of congestion on batter performance

#Null: mean_difference = 0

#Alternative: mean_difference is not equal to 0

t.test(congestedsBA, non_congestedsBA, paired = TRUE, alternative = "two.sided")


#proof

final_hittermiddle$difference<-final_hittermiddle$break_btwn_games-
final_hittermiddle$consecutive_games

print(final_hittermiddle)


#OPS

congestedsOPS<-c(

  mean(villj001_df$OPS[villj001_df$timediff<=1],na.rm = TRUE),

  mean(smoaj001_df$OPS[smoaj001_df$timediff<=1],na.rm = TRUE),

  mean(hedga001_df$OPS[hedga001_df$timediff<=1],na.rm = TRUE),

  mean(reddj001_df$OPS[reddj001_df$timediff<=1],na.rm = TRUE),

  mean(pillk001_df$OPS[pillk001_df$timediff<=1],na.rm = TRUE),

  mean(schoj001_df$OPS[schoj001_df$timediff<=1],na.rm = TRUE),

  mean(casts001_df$OPS[casts001_df$timediff<=1],na.rm = TRUE),

  mean(canhm001_df$OPS[canhm001_df$timediff<=1],na.rm = TRUE),

  mean(freed001_df$OPS[freed001_df$timediff<=1],na.rm = TRUE),

  mean(sanom001_df$OPS[sanom001_df$timediff<=1],na.rm = TRUE)

)

non_congestedsOPS<-c(
```

```
  mean(villj001_df$OPS[villj001_df$timediff>=2],na.rm = TRUE),
  mean(smoaj001_df$OPS[smoaj001_df$timediff>=2],na.rm = TRUE),
  mean(hedga001_df$OPS[hedga001_df$timediff>=2],na.rm = TRUE),
  mean(reddj001_df$OPS[reddj001_df$timediff>=2],na.rm = TRUE),
  mean(pillk001_df$OPS[pillk001_df$timediff>=2],na.rm = TRUE),
  mean(schoj001_df$OPS[schoj001_df$timediff>=2],na.rm = TRUE),
  mean(casts001_df$OPS[casts001_df$timediff>=2],na.rm = TRUE),
  mean(canhm001_df$OPS[canhm001_df$timediff>=2],na.rm = TRUE),
  mean(freed001_df$OPS[freed001_df$timediff>=2],na.rm = TRUE),
  mean(sanom001_df$OPS[sanom001_df$timediff>=2],na.rm = TRUE)
)

OPS_Hittermiddle<-c("Jonathan Villar", "Justin Smoak", "Austin Hedges", "Josh Reddick",
                    "Kevin Pillar", "Jonathan Schoop", "Starlin Castro", "Mark Canha",
                    "David Freese", "Miguel Sano")
OPS_finalmiddle<-data.frame(OPS_Hittermiddle,congestedsOPS,non_congestedsOPS)
print(OPS_finalmiddle)

## Hypothesis Testing

#Null: There is an effect of congestion on batter performance
#Alternative: There is no effect of congestion on batter performance

#Null: mean_difference = 0
#Alternative: mean_difference is not equal to 0

t.test(congestedsOPS, non_congestedsOPS, paired = TRUE, alternative = "two.sided")
```

```
#proof

OPS_finalmiddle$OPS_difference<-                    OPS_finalmiddle$non_congestedsOPS-
OPS_finalmiddle$congestedsOPS

print(OPS_finalmiddle)
```

```
###################### ANALYSIS FOR BOTTOM 5 BATTERS ##########################

# Batter B1 - Yangervis Solarte

solay001_df <- filter(bat, person.key=="solay001")   #create dataframe

str(solay001_df)

solay001_df$game.date<-as.Date(solay001_df$game.date)

x<-diff(solay001_df$game.date)

x<-append(x,0,after = 0)

solay001_df$timediff<-x


par(mfrow = c(1, 2))

solarteOPS<-lm(OPS ~ timediff,data = solay001_df)

plot(solay001_df$timediff, solay001_df$OPS,

    xlab = "Time between matches",

    ylab = "OPS",

    main = "OPS against time difference",

    col ="#99CC00" , xlim=c(0,6)
```

```
)
abline(a = coef(solarteOPS)[1] , b = coef(solarteOPS)[2] , lwd = 2)



solarteBA<-lm(BA ~ timediff,data = solay001_df)
plot(solay001_df$timediff, solay001_df$BA,
    xlab = "Time between matches",
    ylab = "BA",
    main = "BA against time difference",
    col ="#66CCCC", xlim=c(0,6)
)
abline(a = coef(solarteBA)[1] , b = coef(solarteBA)[2] , lwd = 2)



# Batter B2 - Kendrys Morales
morak001_df <- filter(bat, person.key=="morak001")   #create dataframe
str(morak001_df)
morak001_df$game.date<-as.Date(morak001_df$game.date)
x<-diff(morak001_df$game.date)
x<-append(x,0,after = 0)
morak001_df$timediff<-x

par(mfrow = c(1, 2))
moralesOPS<-lm(OPS ~ timediff,data = morak001_df)
plot(morak001_df$timediff, morak001_df$OPS,
    xlab = "Time between matches",
    ylab = "OPS",
```

```
    main = "OPS against time difference",
    col ="#99CC00", xlim=c(0,6)
)
abline(a = coef(moralesOPS)[1] , b = coef(moralesOPS)[2] , lwd = 2)



moralesBA<-lm(BA ~ timediff,data = morak001_df)
plot(morak001_df$timediff, morak001_df$BA,
    xlab = "Time between matches",
    ylab = "BA",
    main = "BA against time difference",
    col ="#66CCCC", xlim=c(0,6)
)
abline(a = coef(moralesBA)[1] , b = coef(moralesBA)[2] , lwd = 2)



# Batter B3 - Ian Desmond
desmi001_df <- filter(bat, person.key=="desmi001")   #create dataframe
str(desmi001_df)
desmi001_df$game.date<-as.Date(desmi001_df$game.date)
x<-diff(desmi001_df$game.date)
x<-append(x,0,after = 0)
desmi001_df$timediff<-x

par(mfrow = c(1, 2))
ianOPS<-lm(OPS ~ timediff,data = desmi001_df)
plot(desmi001_df$timediff, desmi001_df$OPS,
```

```
    xlab = "Time between matches",
    ylab = "OPS",
    main = "OPS against time difference",
    col ="#99CC00", xlim=c(0,6)
)
abline(a = coef(ianOPS)[1] , b = coef(ianOPS)[2] , lwd = 2)



ianBA<-lm(BA ~ timediff,data = desmi001_df)
plot(desmi001_df$timediff, desmi001_df$BA,
    xlab = "Time between matches",
    ylab = "BA",
    main = "BA against time difference",
    col ="#66CCCC", xlim=c(0,6)
)
abline(a = coef(ianBA)[1] , b = coef(ianBA)[2] , lwd = 2)



# Albert Pujols - Batter B4


pujoa001_df <- filter(bat, person.key=="pujoa001")   #create dataframe
str(pujoa001_df)
pujoa001_df$game.date<-as.Date(pujoa001_df$game.date)
x<-diff(pujoa001_df$game.date)
x<-append(x,0,after = 0)
pujoa001_df$timediff<-x
```

```
par(mfrow = c(1, 2))
albertpujolsOPS<-lm(OPS ~ timediff,data = pujoa001_df)
plot(pujoa001_df$timediff, pujoa001_df$OPS,
    xlab = "Time between matches",
    ylab = "OPS",
    main = "OPS against time difference",
    col ="#99CC00", xlim=c(0,6)
)
abline(a = coef(albertpujolsOPS)[1] , b = coef(albertpujolsOPS)[2] , lwd = 2)



albertpujolsBA<-lm(BA ~ timediff,data = pujoa001_df)
plot(pujoa001_df$timediff, pujoa001_df$BA,
    xlab = "Time between matches",
    ylab = "BA",
    main = "BA against time difference",
    col ="#66CCCC", xlim=c(0,6)
)
abline(a = coef(albertpujolsBA)[1] , b = coef(albertpujolsBA)[2] , lwd = 2)



# Batter B5 - Chris Davis
davic003_df <- filter(bat, person.key=="davic003")   #create dataframe
str(davic003_df)
davic003_df$game.date<-as.Date(davic003_df$game.date)
x<-diff(davic003_df$game.date)
x<-append(x,0,after = 0)
```

```
davic003_df$timediff<-x


par(mfrow = c(1, 2))
chrisOPS<-lm(OPS ~ timediff,data = davic003_df)
plot(davic003_df$timediff, davic003_df$OPS,
    xlab = "Time between matches",
    ylab = "OPS",
    main = "OPS against time difference",
    col ="#99CC00", xlim=c(0,6)
)
abline(a = coef(chrisOPS)[1] , b = coef(chrisOPS)[2] , lwd = 2)



chrisBA<-lm(BA ~ timediff,data = davic003_df)
plot(davic003_df$timediff, davic003_df$BA,
    xlab = "Time between matches",
    ylab = "BA",
    main = "BA against time difference",
    col ="#66CCCC", xlim=c(0,6)
)
abline(a = coef(chrisBA)[1] , b = coef(chrisBA)[2] , lwd = 2)



## Comparison for bottom 5 batters


#BA
```

```
congestedsBA<-c(
  mean(solay001_df$BA[solay001_df$timediff<=1],na.rm = TRUE),
  mean(morak001_df$BA[morak001_df$timediff<=1],na.rm = TRUE),
  mean(desmi001_df$BA[desmi001_df$timediff<=1],na.rm = TRUE),
  mean(pujoa001_df$BA[pujoa001_df$timediff<=1],na.rm = TRUE),
  mean(davic003_df$BA[davic003_df$timediff<=1],na.rm = TRUE)
)
non_congestedsBA<-c(
  mean(solay001_df$BA[solay001_df$timediff>=2],na.rm = TRUE),
  mean(morak001_df$BA[morak001_df$timediff>=2],na.rm = TRUE),
  mean(desmi001_df$BA[desmi001_df$timediff>=2],na.rm = TRUE),
  mean(pujoa001_df$BA[pujoa001_df$timediff>=2],na.rm = TRUE),
  mean(davic003_df$BA[davic003_df$timediff>=2],na.rm = TRUE)
)

#names to be changed
Hitter_bottomBA<-c("Yangervis Solarte","Kendrys Morales","Ian Desmond" ,"Albert Pujols", "Chris Davis")

final_hitterbottom<-data.frame(Hitter_bottomBA, congestedsBA,non_congestedsBA)

## Hypothesis Testing for BA
print(final_hitterbottom)

#Null: There is an effect of congestion on batter performance
#Alternative: There is no effect of congestion on batter performance
#Null: mean_difference = 0
#Alternative: mean_difference is not equal to 0
```

```
t.test(congestedsBA, non_congestedsBA, paired = TRUE, alternative = "two.sided")


#proof

final_hitterbottom$difference<-final_hitterbottom$break_btwn_games-
final_hitterbottom$consecutive_games

print(final_hitterbottom)




#OPS

congestedsOPS<-c(

  mean(solay001_df$OPS[solay001_df$timediff<=1],na.rm = TRUE),

  mean(morak001_df$OPS[morak001_df$timediff<=1],na.rm = TRUE),

  mean(desmi001_df$OPS[desmi001_df$timediff<=1],na.rm = TRUE),

  mean(pujoa001_df$OPS[pujoa001_df$timediff<=1],na.rm = TRUE),

  mean(davic003_df$OPS[davic003_df$timediff<=1],na.rm = TRUE)

)

non_congestedsOPS<-c(

  mean(solay001_df$OPS[solay001_df$timediff>=2],na.rm = TRUE),

  mean(morak001_df$OPS[morak001_df$timediff>=2],na.rm = TRUE),

  mean(desmi001_df$OPS[desmi001_df$timediff>=2],na.rm = TRUE),

  mean(pujoa001_df$OPS[pujoa001_df$timediff>=2],na.rm = TRUE),

  mean(davic003_df$OPS[davic003_df$timediff>=2],na.rm = TRUE)

)


OPS_Hitterbottom<-c("Yangervis Solarte","Kendrys Morales","Ian Desmond" ,"Albert Pujols", "Chris
Davis")
```

```
OPS_finalbottom<-data.frame(OPS_Hitterbottom,congestedsOPS,non_congestedsOPS)

print(OPS_finalbottom)
```

Hypothesis Testing

#Null: There is an effect of congestion on batter performance

#Alternative: There is no effect of congestion on batter performance

#Null: mean_difference = 0

#Alternative: mean_difference is not equal to 0

```
t.test(congestedsOPS, non_congestedsOPS, paired = TRUE, alternative = "two.sided")

#proof

OPS_finalbottom$OPS_difference<- OPS_finalbottom$non_congestedsOPS-OPS_finalbottom$congestedsOPS

print(OPS_finalbottom)
```

###################### PITCHERS

```
# Lower ER is better

# High SO is good


#Create separate batting analysis data

colnames(data)

pitch<-data[-c(1:2, 4:8, 10, 12:38, 74:179)]

colnames(pitch)



############################## ANALYSIS FOR TOP 5 PITCHERS ###################


# Max Scherzer - Pitcher 1 - 35obs

schem001_df <- filter(pitch, person.key=="schem001")

str(schem001_df)

schem001_df$game.date<-as.Date(schem001_df$game.date)

x<-diff(schem001_df$game.date)

x<-append(x,0,after = 0)

schem001_df$timediff<-x


par(mfrow = c(1, 2))

maxscherzerER<-lm(P_ER ~ timediff,data = schem001_df)

plot(schem001_df$timediff, schem001_df$P_ER,

    xlab = "Time between matches",

    ylab = "ER (Earned Runs Allowed)",

    main = "ER against time difference",

    col ="333366", xlim=c(0,10)

)
```

```
abline(a = coef(maxscherzerER)[1] , b = coef(maxscherzerER)[2] , lwd = 2)



maxscherzerSO<-lm(P_SO ~ timediff,data = schem001_df)
plot(schem001_df$timediff, schem001_df$P_SO,
    xlab = "Time between matches",
    ylab = "SO(Strikeouts)",
    main = "SO against time difference",
    col ="#FF9933", xlim=c(0,10)
)
abline(a = coef(maxscherzerSO)[1] , b = coef(maxscherzerSO)[2] , lwd = 2)



# - Jacob deGram - Pitcher 2 - 34 obs

degrj001_df <- filter(pitch, person.key=="degrj001")
str(degrj001_df)
degrj001_df$game.date<-as.Date(degrj001_df$game.date)
x<-diff(degrj001_df$game.date)
x<-append(x,0,after = 0)
degrj001_df$timediff<-x

par(mfrow = c(1, 2))
jacobdegramER<-lm(P_ER ~ timediff,data = degrj001_df)
plot(degrj001_df$timediff, degrj001_df$P_ER,
    xlab = "Time between matches",
    ylab = "ER (Earned Runs Allowed)",
```

```
    main = "ER against time difference",
    col ="333366", xlim=c(0,10)
)
abline(a = coef(jacobdegramER)[1] , b = coef(jacobdegramER)[2] , lwd = 2)



jacobdegramSO<-lm(P_SO ~ timediff,data = degrj001_df)
plot(degrj001_df$timediff, degrj001_df$P_SO,
    xlab = "Time between matches",
    ylab = "SO(Strikeouts)",
    main = "SO against time difference",
    col ="#FF9933", xlim=c(0,10)
)
abline(a = coef(jacobdegramSO)[1] , b = coef(jacobdegramSO)[2] , lwd = 2)



# Chris Sale - Pitcher 4 - 25 obs
salec001_df <- filter(pitch, person.key=="salec001")
str(salec001_df)
salec001_df$game.date<-as.Date(salec001_df$game.date)
x<-diff(salec001_df$game.date)
x<-append(x,0,after = 0)
salec001_df$timediff<-x

par(mfrow = c(1, 2))
chrissalesER<-lm(P_ER ~ timediff,data = salec001_df)
```

```
plot(salec001_df$timediff, salec001_df$P_ER,
    xlab = "Time between matches",
    ylab = "ER (Earned Runs Allowed)",
    main = "ER against time difference",
    col ="333366", xlim=c(0,10)
)
abline(a = coef(chrissalesER)[1] , b = coef(chrissalesER)[2] , lwd = 2)



chrissalesSO<-lm(P_SO ~ timediff,data = salec001_df)
plot(salec001_df$timediff, salec001_df$P_SO,
    xlab = "Time between matches",
    ylab = "SO(Strikeouts)",
    main = "SO against time difference",
    col ="#FF9933", xlim=c(0,10)
)
abline(a = coef(chrissalesSO)[1] , b = coef(chrissalesSO)[2] , lwd = 2)

# Justin Verlander - Pitcher 3
verlj001_df <- filter(pitch, person.key=="verlj001")
str(verlj001_df)
verlj001_df$game.date<-as.Date(verlj001_df$game.date)
x<-diff(verlj001_df$game.date)
x<-append(x,0,after = 0)
verlj001_df$timediff<-x

par(mfrow = c(1, 2))
```

```
justinverlanderER<-lm(P_ER ~ timediff,data = verlj001_df)
plot(verlj001_df$timediff, verlj001_df$P_ER,
    xlab = "Time between matches",
    ylab = "ER (Earned Runs Allowed)",
    main = "ER against time difference",
    col ="333366", xlim=c(0,10)
)
abline(a = coef(justinverlanderER)[1] , b = coef(justinverlanderER)[2] , lwd = 2)



justinverlanderSO<-lm(P_SO ~ timediff,data = verlj001_df)
plot(verlj001_df$timediff, verlj001_df$P_SO,
    xlab = "Time between matches",
    ylab = "SO(Strikeouts)",
    main = "SO against time difference",
    col ="#FF9933", xlim=c(0,10)
)
abline(a = coef(justinverlanderSO)[1] , b = coef(justinverlanderSO)[2] , lwd = 2)



# Gerrit Cole - Pitcher 5
coleg001_df <- filter(pitch, person.key=="coleg001")
str(coleg001_df)
coleg001_df$game.date<-as.Date(coleg001_df$game.date)
x<-diff(coleg001_df$game.date)
x<-append(x,0,after = 0)
```

```
coleg001_df$timediff<-x


par(mfrow = c(1, 2))
gerritcoleER<-lm(P_ER ~ timediff,data = coleg001_df)
plot(coleg001_df$timediff, coleg001_df$P_ER,
    xlab = "Time between matches",
    ylab = "ER (Earned Runs Allowed)",
    main = "ER against time difference",
    col ="333366", xlim=c(0,10)
)
abline(a = coef(gerritcoleER)[1] , b = coef(gerritcoleER)[2] , lwd = 2)



gerritcoleSO<-lm(P_SO ~ timediff,data = coleg001_df)
plot(coleg001_df$timediff, coleg001_df$P_SO,
    xlab = "Time between matches",
    ylab = "SO(Strikeouts)",
    main = "SO against time difference",
    col ="#FF9933", xlim=c(0,10)
)
abline(a = coef(gerritcoleSO)[1] , b = coef(gerritcoleSO)[2] , lwd = 2)



## Comparison for top 5 pitchers


#ER
congestedER<-c(
```

```
  mean(schem001_df$P_ER[schem001_df$timediff<=1],na.rm = TRUE),
  mean(degrj001_df$P_ER[degrj001_df$timediff<=1],na.rm = TRUE),
  mean(salec001_df$P_ER[salec001_df$timediff<=1],na.rm = TRUE),
  mean(verlj001_df$P_ER[verlj001_df$timediff<=1],na.rm = TRUE),
  mean(coleg001_df$P_ER[coleg001_df$timediff<=1],na.rm = TRUE)

)
non_congestedER<-c(
  mean(schem001_df$P_ER[schem001_df$timediff>=2],na.rm = TRUE),
  mean(degrj001_df$P_ER[degrj001_df$timediff>=2],na.rm = TRUE),
  mean(salec001_df$P_ER[salec001_df$timediff>=2],na.rm = TRUE),
  mean(verlj001_df$P_ER[verlj001_df$timediff>=2],na.rm = TRUE),
  mean(coleg001_df$P_ER[coleg001_df$timediff>=2],na.rm = TRUE)
)

#names to be changed
Pitcher_topER<-c("Max Scherzer","Jacob deGrom","Chris Sale","Justin Verlander","Gerrit Cole")

final_pitchertop<-data.frame(Pitcher_topER, congestedER,non_congestedER)

## Hypothesis Testing for BA
print(final_pitchertop)

#Null: There is an effect of congestion on batter performance
#Alternative: There is no effect of congestion on batter performance
#Null: mean_difference = 0
#Alternative: mean_difference is not equal to 0
```

```
t.test(congestedER, non_congestedER, paired = TRUE, alternative = "two.sided")

#proof
final_pitchertop$difference<-final_pitchertop$non_congestedER-final_pitchertop$congestedER
print(final_pitchertop)

#SO
congestedSO<-c(
  mean(schem001_df$P_SO[schem001_df$timediff<=1],na.rm = TRUE),
  mean(degrj001_df$P_SO[degrj001_df$timediff<=1],na.rm = TRUE),
  mean(salec001_df$P_SO[salec001_df$timediff<=1],na.rm = TRUE),
  mean(verlj001_df$P_SO[verlj001_df$timediff<=1],na.rm = TRUE),
  mean(coleg001_df$P_SO[coleg001_df$timediff<=1],na.rm = TRUE)
)
non_congestedSO<-c(
  mean(schem001_df$P_SO[schem001_df$timediff>=2],na.rm = TRUE),
  mean(degrj001_df$P_SO[degrj001_df$timediff>=2],na.rm = TRUE),
  mean(salec001_df$P_SO[salec001_df$timediff>=2],na.rm = TRUE),
  mean(verlj001_df$P_SO[verlj001_df$timediff>=2],na.rm = TRUE),
  mean(coleg001_df$P_SO[coleg001_df$timediff>=2],na.rm = TRUE)
)

SO_Pitchertop<-c("Max Scherzer","Jacob deGrom","Chris Sale","Justin Verlander","Gerrit Cole")

SO_finalpitchertop<-data.frame(SO_Pitchertop,congestedSO,non_congestedSO)
```

```
print(SO_finalpitchertop)
```

Hypothesis Testing

```
#Null: There is an effect of congestion on batter performance
#Alternative: There is no effect of congestion on batter performance

#Null: mean_difference = 0
#Alternative: mean_difference is not equal to 0

t.test(congestedSO, non_congestedSO, paired = TRUE, alternative = "two.sided")

#proof
SO_finalpitchertop$OPS_difference<- SO_finalpitchertop$non_congestedSO-
  SO_finalpitchertop$congestedSO
print(SO_finalpitchertop)
```

```
################################## ANALYSIS FOR MIDDLE 10 PITCHERS
```

```
# PITCHER M1 - Joe Musgrove

musgj001_df <- filter(pitch, person.key=="musgj001")

str(musgj001_df)

musgj001_df$game.date<-as.Date(musgj001_df$game.date)

x<-diff(musgj001_df$game.date)

x<-append(x,0,after = 0)

musgj001_df$timediff<-x


par(mfrow = c(1, 2))

musgroveER<-lm(P_ER ~ timediff,data = musgj001_df)

plot(musgj001_df$timediff, musgj001_df$P_ER,

    xlab = "Time between matches",

    ylab = "ER (Earned Runs Allowed)",

    main = "ER against time difference",

    col ="333366", xlim=c(0,10)

)

abline(a = coef(musgroveER)[1] , b = coef(musgroveER)[2] , lwd = 2)



musgroveSO<-lm(P_SO ~ timediff,data = musgj001_df)

plot(musgj001_df$timediff, musgj001_df$P_SO,

    xlab = "Time between matches",

    ylab = "SO(Strikeouts)",

    main = "SO against time difference",

    col ="#FF9933", xlim=c(0,10)

)

abline(a = coef(musgroveSO)[1] , b = coef(musgroveSO)[2] , lwd = 2)
```

```
# PITCHER M2 - Cole Hamels

hamec001_df <- filter(pitch, person.key=="hamec001")

str(hamec001_df)

hamec001_df$game.date<-as.Date(hamec001_df$game.date)

x<-diff(hamec001_df$game.date)

x<-append(x,0,after = 0)

hamec001_df$timediff<-x


par(mfrow = c(1, 2))

hamelscoleER<-lm(P_ER ~ timediff,data = hamec001_df)

plot(hamec001_df$timediff, hamec001_df$P_ER,

    xlab = "Time between matches",

    ylab = "ER (Earned Runs Allowed)",

    main = "ER against time difference",

    col ="333366", xlim=c(0,10)

)

abline(a = coef(hamelscoleER)[1] , b = coef(hamelscoleER)[2] , lwd = 2)



hamelscoleSO<-lm(P_SO ~ timediff,data = hamec001_df)

plot(hamec001_df$timediff, hamec001_df$P_SO,

    xlab = "Time between matches",

    ylab = "SO(Strikeouts)",

    main = "SO against time difference",

    col ="#FF9933", xlim=c(0,10)
```

```
)
abline(a = coef(hamelscoleSO)[1] , b = coef(hamelscoleSO)[2] , lwd = 2)



# PITCHER M3 - Mike Foltynewicz
foltm001_df <- filter(pitch, person.key=="foltm001")
str(foltm001_df)
foltm001_df$game.date<-as.Date(foltm001_df$game.date)
x<-diff(foltm001_df$game.date)
x<-append(x,0,after = 0)
foltm001_df$timediff<-x

par(mfrow = c(1, 2))
fmikeER<-lm(P_ER ~ timediff,data = foltm001_df)
plot(foltm001_df$timediff, foltm001_df$P_ER,
    xlab = "Time between matches",
    ylab = "ER (Earned Runs Allowed)",
    main = "ER against time difference",
    col ="333366", xlim=c(0,10)
)
abline(a = coef(fmikeER)[1] , b = coef(fmikeER)[2] , lwd = 2)



fmikeSO<-lm(P_SO ~ timediff,data = foltm001_df)
plot(foltm001_df$timediff, foltm001_df$P_SO,
    xlab = "Time between matches",
    ylab = "SO(Strikeouts)",
```

```
    main = "SO against time difference",
    col ="#FF9933", xlim=c(0,10)
)
abline(a = coef(fmikeSO)[1] , b = coef(fmikeSO)[2] , lwd = 2)



# PITCHER M4 - Dallas Keuchel
keucd001_df <- filter(pitch, person.key=="keucd001")
str(keucd001_df)
keucd001_df$game.date<-as.Date(keucd001_df$game.date)
x<-diff(keucd001_df$game.date)
x<-append(x,0,after = 0)
keucd001_df$timediff<-x

par(mfrow = c(1, 2))
dallasER<-lm(P_ER ~ timediff,data = keucd001_df)
plot(keucd001_df$timediff, keucd001_df$P_ER,
    xlab = "Time between matches",
    ylab = "ER (Earned Runs Allowed)",
    main = "ER against time difference",
    col ="333366", xlim=c(0,10)
)
abline(a = coef(dallasER)[1] , b = coef(dallasER)[2] , lwd = 2)



dallasSO<-lm(P_SO ~ timediff,data = keucd001_df)
plot(keucd001_df$timediff, keucd001_df$P_SO,
```

```
    xlab = "Time between matches",
    ylab = "SO(Strikeouts)",
    main = "SO against time difference",
    col ="#FF9933", xlim=c(0,10)
)
abline(a = coef(dallasSO)[1] , b = coef(dallasSO)[2] , lwd = 2)



# PITCHER M5 - Kyle Freeland
freek001_df <- filter(pitch, person.key=="freek001")
str(freek001_df)
freek001_df$game.date<-as.Date(freek001_df$game.date)
x<-diff(freek001_df$game.date)
x<-append(x,0,after = 0)
freek001_df$timediff<-x

par(mfrow = c(1, 2))
freelandkER<-lm(P_ER ~ timediff,data = freek001_df)
plot(freek001_df$timediff, freek001_df$P_ER,
    xlab = "Time between matches",
    ylab = "ER (Earned Runs Allowed)",
    main = "ER against time difference",
    col ="333366", xlim=c(0,10)
)
abline(a = coef(freelandkER)[1] , b = coef(freelandkER)[2] , lwd = 2)
```

```
freelandkSO<-lm(P_SO ~ timediff,data = freek001_df)
plot(freek001_df$timediff, freek001_df$P_SO,
    xlab = "Time between matches",
    ylab = "SO(Strikeouts)",
    main = "SO against time difference",
    col ="#FF9933", xlim=c(0,10)
)
abline(a = coef(freelandkSO)[1] , b = coef(freelandkSO)[2] , lwd = 2)



# Dylan Bundy - Pitcher M6
bundd001_df <- filter(pitch, person.key=="bundd001")
str(bundd001_df)
bundd001_df$game.date<-as.Date(bundd001_df$game.date)
x<-diff(bundd001_df$game.date)
x<-append(x,0,after = 0)
bundd001_df$timediff<-x


par(mfrow = c(1, 2))
dylanbundyER<-lm(P_ER ~ timediff,data = bundd001_df)
plot(bundd001_df$timediff, bundd001_df$P_ER,
    xlab = "Time between matches",
    ylab = "ER (Earned Runs Allowed)",
    main = "ER against time difference",
    col ="333366", xlim=c(0,10)
)
abline(a = coef(dylanbundyER)[1] , b = coef(dylanbundyER)[2] , lwd = 2)
```

```
dylanbundySO<-lm(P_SO ~ timediff,data = bundd001_df)
plot(bundd001_df$timediff, bundd001_df$P_SO,
    xlab = "Time between matches",
    ylab = "SO(Strikeouts)",
    main = "SO against time difference",
    col ="#FF9933", xlim=c(0,10)
)
abline(a = coef(dylanbundySO)[1] , b = coef(dylanbundySO)[2] , lwd = 2)

# Kyle Gibson - Pitcher M7
gibsk002_df <- filter(pitch, person.key=="gibsk002")
str(gibsk002_df)
gibsk002_df$game.date<-as.Date(gibsk002_df$game.date)
x<-diff(gibsk002_df$game.date)
x<-append(x,0,after = 0)
gibsk002_df$timediff<-x

par(mfrow = c(1, 2))
kylegibsonER<-lm(P_ER ~ timediff,data = gibsk002_df)
plot(gibsk002_df$timediff, gibsk002_df$P_ER,
    xlab = "Time between matches",
    ylab = "ER (Earned Runs Allowed)",
    main = "ER against time difference",
    col ="333366", xlim=c(0,10)
)
```

```
abline(a = coef(kylegibsonER)[1] , b = coef(kylegibsonER)[2] , lwd = 2)



kylegibsonSO<-lm(P_SO ~ timediff,data = gibsk002_df)
plot(gibsk002_df$timediff, gibsk002_df$P_SO,
    xlab = "Time between matches",
    ylab = "SO(Strikeouts)",
    main = "SO against time difference",
    col ="#FF9933", xlim=c(0,10)
)
abline(a = coef(kylegibsonSO)[1] , b = coef(kylegibsonSO)[2] , lwd = 2)



# PITCHER M8 - Madison Bumgarner
bumgm001_df <- filter(pitch, person.key=="bumgm001")
str(bumgm001_df)
bumgm001_df$game.date<-as.Date(bumgm001_df$game.date)
x<-diff(bumgm001_df$game.date)
x<-append(x,0,after = 0)
bumgm001_df$timediff<-x

par(mfrow = c(1, 2))
madisonER<-lm(P_ER ~ timediff,data = bumgm001_df)
plot(bumgm001_df$timediff, bumgm001_df$P_ER,
    xlab = "Time between matches",
    ylab = "ER (Earned Runs Allowed)",
    main = "ER against time difference",
```

```
    col ="333366", xlim=c(0,10)
)
abline(a = coef(madisonER)[1] , b = coef(madisonER)[2] , lwd = 2)



madisonSO<-lm(P_SO ~ timediff,data = bumgm001_df)
plot(bumgm001_df$timediff, bumgm001_df$P_SO,
    xlab = "Time between matches",
    ylab = "SO(Strikeouts)",
    main = "SO against time difference",
    col ="#FF9933", xlim=c(0,10)
)
abline(a = coef(madisonSO)[1] , b = coef(madisonSO)[2] , lwd = 2)



# PITCHER M9 - David Price
pricd001_df <- filter(pitch, person.key=="pricd001")
str(pricd001_df)
pricd001_df$game.date<-as.Date(pricd001_df$game.date)
x<-diff(pricd001_df$game.date)
x<-append(x,0,after = 0)
pricd001_df$timediff<-x

par(mfrow = c(1, 2))
pricedavidER<-lm(P_ER ~ timediff,data = pricd001_df)
plot(pricd001_df$timediff, pricd001_df$P_ER,
    xlab = "Time between matches",
```

```
    ylab = "ER (Earned Runs Allowed)",
    main = "ER against time difference",
    col ="333366", xlim=c(0,10)
)
abline(a = coef(pricedavidER)[1] , b = coef(pricedavidER)[2] , lwd = 2)



pricedavidSO<-lm(P_SO ~ timediff,data = pricd001_df)
plot(pricd001_df$timediff, pricd001_df$P_SO,
    xlab = "Time between matches",
    ylab = "SO(Strikeouts)",
    main = "SO against time difference",
    col ="#FF9933", xlim=c(0,10)
)
abline(a = coef(pricedavidSO)[1] , b = coef(pricedavidSO)[2] , lwd = 2)

# Rick Porcello - Pitcher M10
porcr001_df <- filter(pitch, person.key=="porcr001")
str(porcr001_df)
porcr001_df$game.date<-as.Date(porcr001_df$game.date)
x<-diff(porcr001_df$game.date)
x<-append(x,0,after = 0)
porcr001_df$timediff<-x

par(mfrow = c(1, 2))
rickporcelloER<-lm(P_ER ~ timediff,data = porcr001_df)
plot(porcr001_df$timediff, porcr001_df$P_ER,
```

```
    xlab = "Time between matches",
    ylab = "ER (Earned Runs Allowed)",
    main = "ER against time difference",
    col ="333366", xlim=c(0,10)
)
abline(a = coef(rickporcelloER)[1] , b = coef(rickporcelloER)[2] , lwd = 2)



rickporcelloSO<-lm(P_SO ~ timediff,data = porcr001_df)
plot(porcr001_df$timediff, porcr001_df$P_SO,
    xlab = "Time between matches",
    ylab = "SO(Strikeouts)",
    main = "SO against time difference",
    col ="#FF9933", xlim=c(0,10)
)
abline(a = coef(rickporcelloSO)[1] , b = coef(rickporcelloSO)[2] , lwd = 2)

## Comparison for bottom 10 pitchers

#ER
congestedER<-c(
  mean(musgj001_df$P_ER[musgj001_df$timediff<=1],na.rm = TRUE),
  mean(hamec001_df$P_ER[hamec001_df$timediff<=1],na.rm = TRUE),
  mean(foltm001_df$P_ER[foltm001_df$timediff<=1],na.rm = TRUE),
  mean(keucd001_df$P_ER[keucd001_df$timediff<=1],na.rm = TRUE),
  mean(freek001_df$P_ER[freek001_df$timediff<=1],na.rm = TRUE),
  mean(bundd001_df$P_ER[bundd001_df$timediff<=1],na.rm = TRUE),
```

```
  mean(gibsk002_df$P_ER[gibsk002_df$timediff<=1],na.rm = TRUE),
  mean(bumgm001_df$P_ER[bumgm001_df$timediff<=1],na.rm = TRUE),
  mean(pricd001_df$P_ER[pricd001_df$timediff<=1],na.rm = TRUE),
  mean(porcr001_df$P_ER[porcr001_df$timediff<=1],na.rm = TRUE)
)
non_congestedER<-c(
  mean(musgj001_df$P_ER[musgj001_df$timediff>=2],na.rm = TRUE),
  mean(hamec001_df$P_ER[hamec001_df$timediff>=2],na.rm = TRUE),
  mean(foltm001_df$P_ER[foltm001_df$timediff>=2],na.rm = TRUE),
  mean(keucd001_df$P_ER[keucd001_df$timediff>=2],na.rm = TRUE),
  mean(freek001_df$P_ER[freek001_df$timediff>=2],na.rm = TRUE),
  mean(bundd001_df$P_ER[bundd001_df$timediff>=2],na.rm = TRUE),
  mean(gibsk002_df$P_ER[gibsk002_df$timediff>=2],na.rm = TRUE),
  mean(bumgm001_df$P_ER[bumgm001_df$timediff>=2],na.rm = TRUE),
  mean(pricd001_df$P_ER[pricd001_df$timediff>=2],na.rm = TRUE),
  mean(porcr001_df$P_ER[porcr001_df$timediff>=2],na.rm = TRUE)
)

#names to be changed
Pitcher_middleER<-c("Joe Musgrove","Cole Hamels","Mike Foltynewicz","Dallas Keuchel",
                    "Kyle Freeland","Dylan Bundy","Kyle Gibson","Madison Bumgarner",
                    "David Price","Rick Porcello")

final_pitchermiddle<-data.frame(Pitcher_middleER, congestedER,non_congestedER)
## Hypothesis Testing for BA
print(final_pitchermiddle)
```

```
#Null: There is an effect of congestion on batter performance

#Alternative: There is no effect of congestion on batter performance

#Null: mean_difference = 0

#Alternative: mean_difference is not equal to 0

t.test(congestedER, non_congestedER, paired = TRUE, alternative = "two.sided")


#proof

final_pitchermiddle$ERdifference<-final_pitchermiddle$non_congestedER-
final_pitchermiddle$congestedER

print(final_pitchermiddle)


#OPS

congestedSO<-c(

  mean(musgj001_df$P_SO[musgj001_df$timediff<=1],na.rm = TRUE),

  mean(hamec001_df$P_SO[hamec001_df$timediff<=1],na.rm = TRUE),

  mean(foltm001_df$P_SO[foltm001_df$timediff<=1],na.rm = TRUE),

  mean(keucd001_df$P_SO[keucd001_df$timediff<=1],na.rm = TRUE),

  mean(freek001_df$P_SO[freek001_df$timediff<=1],na.rm = TRUE),

  mean(bundd001_df$P_SO[bundd001_df$timediff<=1],na.rm = TRUE),

  mean(gibsk002_df$P_SO[gibsk002_df$timediff<=1],na.rm = TRUE),

  mean(bumgm001_df$P_SO[bumgm001_df$timediff<=1],na.rm = TRUE),

  mean(pricd001_df$P_SO[pricd001_df$timediff<=1],na.rm = TRUE),

  mean(porcr001_df$P_SO[porcr001_df$timediff<=1],na.rm = TRUE)

)

non_congestedSO<-c(

  mean(musgj001_df$P_SO[musgj001_df$timediff>=2],na.rm = TRUE),

  mean(hamec001_df$P_SO[hamec001_df$timediff>=2],na.rm = TRUE),

  mean(foltm001_df$P_SO[foltm001_df$timediff>=2],na.rm = TRUE),
```

```
 mean(keucd001_df$P_SO[keucd001_df$timediff>=2],na.rm = TRUE),
 mean(freek001_df$P_SO[freek001_df$timediff>=2],na.rm = TRUE),
 mean(bundd001_df$P_SO[bundd001_df$timediff>=2],na.rm = TRUE),
 mean(gibsk002_df$P_SO[gibsk002_df$timediff>=2],na.rm = TRUE),
 mean(bumgm001_df$P_SO[bumgm001_df$timediff>=2],na.rm = TRUE),
 mean(pricd001_df$P_SO[pricd001_df$timediff>=2],na.rm = TRUE),
 mean(porcr001_df$P_SO[porcr001_df$timediff>=2],na.rm = TRUE)
)

SO_Pitchermiddle<-c("Joe Musgrove","Cole Hamels","Mike Foltynewicz","Dallas Keuchel",
                    "Kyle Freeland","Dylan Bundy","Kyle Gibson","Madison Bumgarner",
                    "David Price","Rick Porcello")

SO_finalpitchermiddle<-data.frame(SO_Pitchermiddle,congestedSO,non_congestedSO)

print(SO_finalpitchermiddle)

## Hypothesis Testing

#Null: There is an effect of congestion on batter performance
#Alternative: There is no effect of congestion on batter performance

#Null: mean_difference = 0
#Alternative: mean_difference is not equal to 0

t.test(congestedSO, non_congestedSO, paired = TRUE, alternative = "two.sided")
```

```
#proof
SO_finalpitchermiddle$SO_difference<- SO_finalpitchermiddle$non_congestedSO-
  SO_finalpitchermiddle$congestedSO
print(SO_finalpitchermiddle)
```

```
################################## ANALYSIS FOR BOTTOM 5 PITCHERS

# PITCHER B1 - James Shields
shiej002_df <- filter(pitch, person.key=="shiej002")
str(shiej002_df)
shiej002_df$game.date<-as.Date(shiej002_df$game.date)
x<-diff(shiej002_df$game.date)
x<-append(x,0,after = 0)
shiej002_df$timediff<-x

par(mfrow = c(1, 2))
shieldsjamesER<-lm(P_ER ~ timediff,data = shiej002_df)
```

```
plot(shiej002_df$timediff, shiej002_df$P_ER,
    xlab = "Time between matches",
    ylab = "ER (Earned Runs Allowed)",
    main = "ER against time difference",
    col ="333366", xlim=c(0,10)
)
abline(a = coef(shieldsjamesER)[1] , b = coef(shieldsjamesER)[2] , lwd = 2)



shieldsjamesSO<-lm(P_SO ~ timediff,data = shiej002_df)
plot(shiej002_df$timediff, shiej002_df$P_SO,
    xlab = "Time between matches",
    ylab = "SO(Strikeouts)",
    main = "SO against time difference",
    col ="#FF9933", xlim=c(0,10)
)
abline(a = coef(shieldsjamesSO)[1] , b = coef(shieldsjamesSO)[2] , lwd = 2)



# PITCHER B2 - Dan Straily
strad003_df <- filter(pitch, person.key=="strad003")
str(strad003_df)
strad003_df$game.date<-as.Date(strad003_df$game.date)
x<-diff(strad003_df$game.date)
x<-append(x,0,after = 0)
strad003_df$timediff<-x
```

```
par(mfrow = c(1, 2))
danER<-lm(P_ER ~ timediff,data = strad003_df)
plot(strad003_df$timediff, strad003_df$P_ER,
    xlab = "Time between matches",
    ylab = "ER (Earned Runs Allowed)",
    main = "ER against time difference",
    col ="333366", xlim=c(0,10)
)
abline(a = coef(danER)[1] , b = coef(danER)[2] , lwd = 2)



danSO<-lm(P_SO ~ timediff,data = strad003_df)
plot(strad003_df$timediff, strad003_df$P_SO,
    xlab = "Time between matches",
    ylab = "SO(Strikeouts)",
    main = "SO against time difference",
    col ="#FF9933", xlim=c(0,10)
)
abline(a = coef(danSO)[1] , b = coef(danSO)[2] , lwd = 2)



# PITCHER B3 - Felix Hernandez
hernf002_df <- filter(pitch, person.key=="hernf002")
str(hernf002_df)
hernf002_df$game.date<-as.Date(hernf002_df$game.date)
x<-diff(hernf002_df$game.date)
x<-append(x,0,after = 0)
```

```
hernf002_df$timediff<-x


par(mfrow = c(1, 2))
felixER<-lm(P_ER ~ timediff,data = hernf002_df)
plot(hernf002_df$timediff, hernf002_df$P_ER,
    xlab = "Time between matches",
    ylab = "ER (Earned Runs Allowed)",
    main = "ER against time difference",
    col ="333366", xlim=c(0,10)
)
abline(a = coef(felixER)[1] , b = coef(felixER)[2] , lwd = 2)



felixSO<-lm(P_SO ~ timediff,data = hernf002_df)
plot(hernf002_df$timediff, hernf002_df$P_SO,
    xlab = "Time between matches",
    ylab = "SO(Strikeouts)",
    main = "SO against time difference",
    col ="#FF9933", xlim=c(0,10)
)
abline(a = coef(felixSO)[1] , b = coef(felixSO)[2] , lwd = 2)



# PITCHER B4 - Matt Harvey
harvm001_df <- filter(pitch, person.key=="harvm001")
str(harvm001_df)
harvm001_df$game.date<-as.Date(harvm001_df$game.date)
```

```
x<-diff(harvm001_df$game.date)

x<-append(x,0,after = 0)

harvm001_df$timediff<-x


par(mfrow = c(1, 2))

harveyER<-lm(P_ER ~ timediff,data = harvm001_df)

plot(harvm001_df$timediff, harvm001_df$P_ER,

    xlab = "Time between matches",

    ylab = "ER (Earned Runs Allowed)",

    main = "ER against time difference",

    col ="333366", xlim=c(0,10)

)

abline(a = coef(harveyER)[1] , b = coef(harveyER)[2] , lwd = 2)



harveySO<-lm(P_SO ~ timediff,data = harvm001_df)

plot(harvm001_df$timediff, harvm001_df$P_SO,

    xlab = "Time between matches",

    ylab = "SO(Strikeouts)",

    main = "SO against time difference",

    col ="#FF9933", xlim=c(0,10)

)

abline(a = coef(harveySO)[1] , b = coef(harveySO)[2] , lwd = 2)



# PITCHER B5 - Derek Holland

holld003_df <- filter(pitch, person.key=="holld003")
```

```
str(holld003_df)

holld003_df$game.date<-as.Date(holld003_df$game.date)

x<-diff(holld003_df$game.date)

x<-append(x,0,after = 0)

holld003_df$timediff<-x


par(mfrow = c(1, 2))

hollandderekER<-lm(P_ER ~ timediff,data = holld003_df)

plot(holld003_df$timediff, holld003_df$P_ER,

    xlab = "Time between matches",

    ylab = "ER (Earned Runs Allowed)",

    main = "ER against time difference",

    col ="333366", xlim=c(0,10)

)

abline(a = coef(hollandderekER)[1] , b = coef(hollandderekER)[2] , lwd = 2)



hollandderekSO<-lm(P_SO ~ timediff,data = holld003_df)

plot(holld003_df$timediff, holld003_df$P_SO,

    xlab = "Time between matches",

    ylab = "SO(Strikeouts)",

    main = "SO against time difference",

    col ="#FF9933", xlim=c(0,10)

)

abline(a = coef(hollandderekSO)[1] , b = coef(hollandderekSO)[2] , lwd = 2)
```

```
## Comparison for bottom 10 pitchers

#ER

congestedER<-c(
  mean(shiej002_df$P_ER[shiej002_df$timediff<=1],na.rm = TRUE),
  mean(strad003_df$P_ER[strad003_df$timediff<=1],na.rm = TRUE),
  mean(hernf002_df$P_ER[hernf002_df$timediff<=1],na.rm = TRUE),
  mean(harvm001_df$P_ER[harvm001_df$timediff<=1],na.rm = TRUE),
  mean(holld003_df$P_ER[holld003_df$timediff<=1],na.rm = TRUE)
)
non_congestedER<-c(
  mean(shiej002_df$P_ER[shiej002_df$timediff>=2],na.rm = TRUE),
  mean(strad003_df$P_ER[strad003_df$timediff>=2],na.rm = TRUE),
  mean(hernf002_df$P_ER[hernf002_df$timediff>=2],na.rm = TRUE),
  mean(harvm001_df$P_ER[harvm001_df$timediff>=2],na.rm = TRUE),
  mean(holld003_df$P_ER[holld003_df$timediff>=2],na.rm = TRUE)
)

#names to be changed

Pitcher_bottomER<-c("James Shields", "Dan Straily", "Felix Hernandez", "Matt Harvey", "Derek Holland")

final_pitcherbottom<-data.frame(Pitcher_bottomER, congestedER,non_congestedER)

## Hypothesis Testing for BA

print(final_pitcherbottom)
```

```
#Null: There is an effect of congestion on batter performance

#Alternative: There is no effect of congestion on batter performance

#Null: mean_difference = 0

#Alternative: mean_difference is not equal to 0

t.test(congestedER, non_congestedER, paired = TRUE, alternative = "two.sided")


#proof

final_pitcherbottom$ERdifference<-final_pitcherbottom$non_congestedER-
final_pitcherbottom$congestedER

print(final_pitcherbottom)


#OPS


congestedSO<-c(
  mean(shiej002_df$P_SO[shiej002_df$timediff<=1],na.rm = TRUE),
  mean(strad003_df$P_SO[strad003_df$timediff<=1],na.rm = TRUE),
  mean(hernf002_df$P_SO[hernf002_df$timediff<=1],na.rm = TRUE),
  mean(harvm001_df$P_SO[harvm001_df$timediff<=1],na.rm = TRUE),
  mean(holld003_df$P_SO[holld003_df$timediff<=1],na.rm = TRUE)
)
non_congestedSO<-c(
  mean(shiej002_df$P_SO[shiej002_df$timediff>=2],na.rm = TRUE),
  mean(strad003_df$P_SO[strad003_df$timediff>=2],na.rm = TRUE),
  mean(hernf002_df$P_SO[hernf002_df$timediff>=2],na.rm = TRUE),
  mean(harvm001_df$P_SO[harvm001_df$timediff>=2],na.rm = TRUE),
  mean(holld003_df$P_SO[holld003_df$timediff>=2],na.rm = TRUE)
)
```

```
SO_Pitcherbottom<-c("James Shields", "Dan Straily", "Felix Hernandez", "Matt Harvey", "Derek Holland")


SO_finalpitcherbottom<-data.frame(SO_Pitcherbottom,congestedSO,non_congestedSO)

print(SO_finalpitcherbottom)


## Hypothesis Testing

#Null: There is an effect of congestion on batter performance

#Alternative: There is no effect of congestion on batter performance

#Null: mean_difference = 0

#Alternative: mean_difference is not equal to 0


t.test(congestedSO, non_congestedSO, paired = TRUE, alternative = "two.sided")


#proof

SO_finalpitcherbottom$SO_difference<- SO_finalpitcherbottom$non_congestedSO-
  SO_finalpitcherbottom$congestedSO

print(SO_finalpitcherbottom)
```