Can he hit? Measuring and prognosis of offensive performance by baseball teams and individual team members

Paul Berbée (B.Sc. VWL), Max Lobeck (B.Sc. VWL), David Schulze (B.Sc. VWL)

January 10, 2015

Abstract

We propose a probability model for simulating baseball games that takes into account measurable statistics of team performance. We assume team performance can be approximated as a function of individual players abilities. Using an 11 seasons player dataset and an aggregate team dataset we find the best performance proxy for number of runs in every game. An empirical ability can be constructed through aggregating individual player data. Finally this is used to simulate multiple possible season outcomes. Our proxy, Total Team Average, depends heavily on individual player's statistics. Therefore we propose a theoretical model that estimates each player's contribution from his statistics. This yields a relative ability measure for every player. This should be a major factor in determining the price offer made to the future player. Using our simulation model, this makes it possible to calculate what-if scenarios for changes to the team roster. In order to gain insight in future performance we use R programming to discover trends in player performance over the course of their career.

Contents

1 List of abbreviations

AB	At Bats	Plate appearances, not including bases on balls, being hit by pitch, sacrifices, interference, or obstruction.
AVG	Batting Average	Hits divided by at bats (H/AB)
BB	Bases on Balls	Hitter not swinging at four pitches called out of the strike zone and awarded first base
CS	Caught Stealing	Times tagged out while attempting to steal a base
D	Doubles	Hits on which the batter reaches second base safely without the contribution of a fielding error.
G	Games Played	
GDP	Ground into Double Plays	Number of ground balls hit that became double plays
H	Hits	Times reached base because of a batted, fair ball without error by the defense
HBP	Hit by Pitch	Times touched by a pitch and awarded first base as a result
HR	Home Runs	Hits on which the batter successfully touched all four bases, without the contribution of a fielding error
OBP	On-base Percentage	Times reached base (H+BB+HBP) divided by at bats plus walks plus hit by pitch plus sacrifice flies (AB+BB+HBP+SF)
R	Runs Scored	The score made by an offensive player who advances from batter to runner and touches first, second, third and home bases in that order.
PCP	Player Contribuiton Percentage	Player total average divided by team total average (PTA/TTA)
PTA	Player Total Average	Reweighted TTA for an individual player
RBI	Runs Batted In	Number of runners who score due to a batter's action, except when batter grounded into double play or reached on an error
S	Single	Hits on which the batter reaches first base safely without the contribution of a fielding error.
SAC	Sacrifice Bunts	A batter's act of deliberately bunting the ball, before there are two outs, in a manner that allows a runner on base to advance to another base
SB	Stolen Bases	Number of bases advanced by the runner while the ball is in the possession of the defense.
SLG	Slugging Average	Total bases achieved on hits divided by at-bats (TB/AB)
SF	Sacrifice Flies	Fly balls hit to the outfield which although caught for an out, allow a baserunner to advance
SO	Strikeouts	Number of times that a third strike is taken or swung at and missed, or bunted foul. Catcher must catch the third strike or batter may attempt to run to first base
T	Triples	Hits on which the batter reaches third base safely without the contribution of a fielding error.
TA	Total Average	Total bases, plus walks, plus hit by pitch, plus steals, minus caught stealing divided by at bats, minus hits, plus caught stealing, plus grounded into double plays [(TB+BB+HBP+SB-CS)/(AB-H+CS+GIDP)]
TB	Total Bases	One for each single, two for each double, three for each triple, and four for each home run [H+D+(2*T)+(3*HR)] or [S+(2*D)+(3*T)+(4*HR)]
TTA	Team Total Average	Weighted total average of one team

2 Introduction

2.1 Project Outline (all)

Why analyse baseball data? The game offers a discrete and limited set of rules and factors, most of them observable and recorded in official statistics. One of our data sets for example lists 20 individual variables for 74 players over the course of 11 years and about 1800 games of one team. These are 75 time series if you count the team as aggregate. We hope to discover a metric that captures individual player's ability in comparison to the rest of the team. This could serve as a foundation for determining a fair player compensation and motivation for business oriented reasearch, as players earn up to millions of dollars in professional baseball.

2.2 Methods Outline (Max and David

We follow a probability model proposed by Albert and Bennett (2003) for teams and extend it to the player level. Namely we argue that if the team as an aggregate plays at 100% of it's ability count, each player can reside at, below or above this percentage. We hope to end up with a function that uses the official team statistics available to estimate a teams ability count and produce a player specific team contribution percentage. In a hypothetical scenario a new player could then be estimated in value relative to his new team mates using the ability count function.

2.3 Data Sources (all)

The Major League Baseball website publishes all official statistics for all games and players in the Major League since 1871. We downloaded and processed an examplary set of the eleven 2000-2011 seasons's Boston Red Sox player's offensive statistics. The programs and models produced should work with all other data sets one would wish to download, only minor steps were performed manually. In addition we used a team aggregate data set with seasons 2000-2013.

3 Data Processing

3.1 Aggregate Team Data (Max)

The team aggregate dataset includes seasonal offensive data for all teams playing in the MLB. As it covers 14 seasons (from 2000 to 2013) and 30 teams playing every season in the league it consists of 420 observations. It has been downloaded from baseball-reference and apart from merging the different seasons in one data frame no major transformations in the data have been undertaken.

3.2 Individual Player Data (all)

The data for the player specific data is reported on the MLB website (MLB.com). Fortunately for us, the datasets were already checked. With an average of 6.35 million viewers per game (Business Week, 0ctober 21,2013) there is enough confidence to rule out measurement errors. If we wanted to compare over a whole team however, we still needed to erase games not played for the particular team, in our case the Boston Red Sox. The second task was arranging the data by player to facilitate player specific evaluation. We didn't know the reshape-package by Hadley Wickham (link) by the time we prepared the dataset, so we wrote some functions of our own that delivered the desired result, you can review them in the appendix: A. For similar reasons we didn't use the time series format provide by R. Concerning the data preparation process we would like to report a few of our experiences:

3.2.1 Original Datasets (all)

Secondly we use datasets for every individual player of the Boston Red Sox during the seasons 2000-2010. Again, only offensive performance is measured, but in addition we considered only players that have played more than 40 games in a season. It contains the player's statistics for every single game and informations about the opponent and wether the game has been a home game. Merging these data frames we downloaded from mlb.com to one dataset gave us a total number of 17,793 observations.

If we wanted to analyze explicitly the team however, we still needed to erase games not played for the Boston Red Sox by players who changed to other teams during the season. The second task was arranging the data by player to facilitate player specific evaluation. We didn't know the reshape-package by Hadley Wickham (link) by the time we prepared the dataset, so we wrote some functions of our own that delivered the desired result, you can review them in the appendix: A For similar reasons we didn't use the time series format provide by R. Concerning the data preparation process, we would like to report a few of our experiences:

3.2.2 Data Transportation and Formatting (all)

Although R offers probably more functionality than could ever be needed, we used some auxiliary software to help us with handling the data. Some software, like Notepad++ was very useful, since it offered extended search and replace functions, as well as simple macro programming. Some software like Microsoft Excel seemed to be very handy at first, because of our previous experience and the usability. The severe drawbacks however include auto-format and other formatting that changed data points like date and float numerals in a way that was difficult to reverse. As a result, we can recommend only Notepad++ and the csv-format as a means of transportation and formatting of data.

3.2.3 Data Preparation (David)

As we didn't know any packages that fitted our purpose at that moment, we decided to use the programming function of R to make some data preparation functions. To name a few: "remove.games", "datastr" and "arrdat". The portability to other data sets should be very limited, because we lack the experience to write universal packages. For documentary purposes I will briefly discuss the way they work:

1. **"remove.games"** creates a removal criterion "1"-vector of the dimension of all dates in the data set. For every player, every game is checked at points "date" and "opponent". The two data points can uniquely identify the team for which the player was playing. If the check is positive, the "1" is replaced with a "0". At the end, remaining "1"'s must be games played for teams other than the Red Sox and are subsequently deleted by the function and the cleaned data set is returned.

2. **"datastr"** seems to be an ineffective function, because it requires a lot of processing time for a simple task: Restructuring the date variable as a game number. This task was the first step to create a unified dataset with all exported values being arranged to a single date vector. Previously all data was listed as vectors with missing values at the bottom, resulting in every player having his own date vector. This made comparison across players difficult. The function literally walks through every line of every column of the dataset and replaces every date with the correct game number. This pedantic check makes sure that even if there are dates not in the right order, the game number will be correct. At first the function ran forever, because for every data entry the whole list was consulted. The new version made use of the assumption that all dates are in the right order. Now dates are checked against each next entry. The result shows that all dates were proven to be in the right order and received their appropriate game number. You can see that we added a "time.taken" in our function. We used it to compare different codes performance. At first the code was expected to take over an hour for completion, now the running time is a few seconds to a few minutes, depending on the calculation power of the machine used.

3. **"arrdat"** is the last step to create the unified dataset we were looking for: Each column represents one player, one vector shows a universal date for each row. Each player-date combination is represented with the exported statistic. This could be hits per game, or batting average. How did we approach the problem of unifying the date structure? We couldn't arrange the date in the existing data frame for fear of overwriting data. For this reason we created a transport vector of lenght of the maximum game numbers. In turn we filled this vector with each players data vector, placing every value at the place that corresponds to the new unified date vector. Again a pedantic check of every line was necessary, because players missed single games every once in a while. After a player's column is completed, the full transport vector is inserted in the old data frame and emptied again to be filled with the next player's column. Because we needed some help vectors to

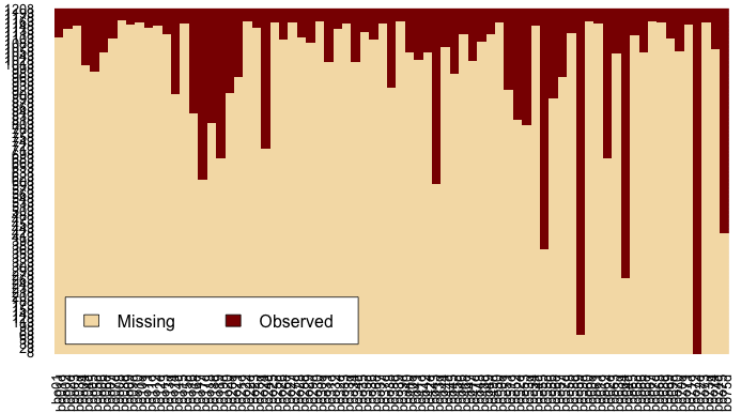


Figure 1: Result of "datastr": missingness map (x: players; y: games)

perform this operation, we have to add in some cleaning at the end. The last part is deleting the old individual player's date vectors. They are not needed now that all values are aligned to one date vector.

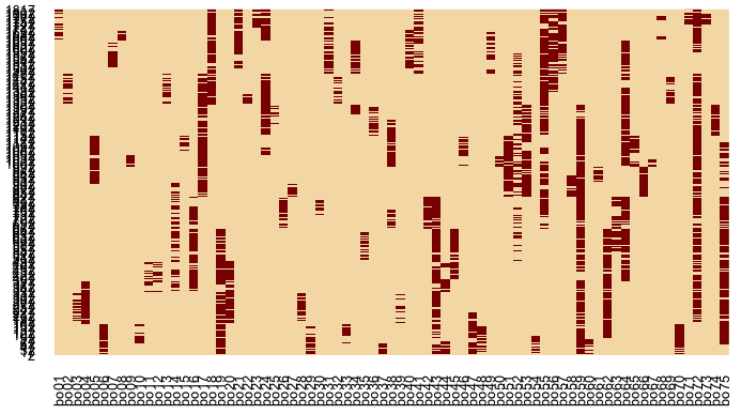


Figure 2: Result of "arrdat": missingness map (x: players; y: games)

The result is 19 data frames with players on columns and games on rows. Each dataframe includes one statistical variable relevant to player game data: At Bases, Runs, Hits, Total Bases, Second Bases, Third Bases, Home Runs, Runs batted in, Base on Balls, Intentional Base on Balls, Strike Outs, Stolen Bases, Caught Stealing Base, Batting Average,

On base percentage, Slugging Average, Home Base Percentage, Sacrificed Bunts, Sacrificed Flies.

This dataset allows elaborate comparison and plotting of different time periods, players, statistics and descriptive values. As an example we show here a stacked line plot of all players Batting Average in 80 games. A record of 4 can be interpreted as $\frac{4}{9}$ times at bat, players succesfully hit a thrown ball. The width of a coloured area indicates the contribution share of the player.

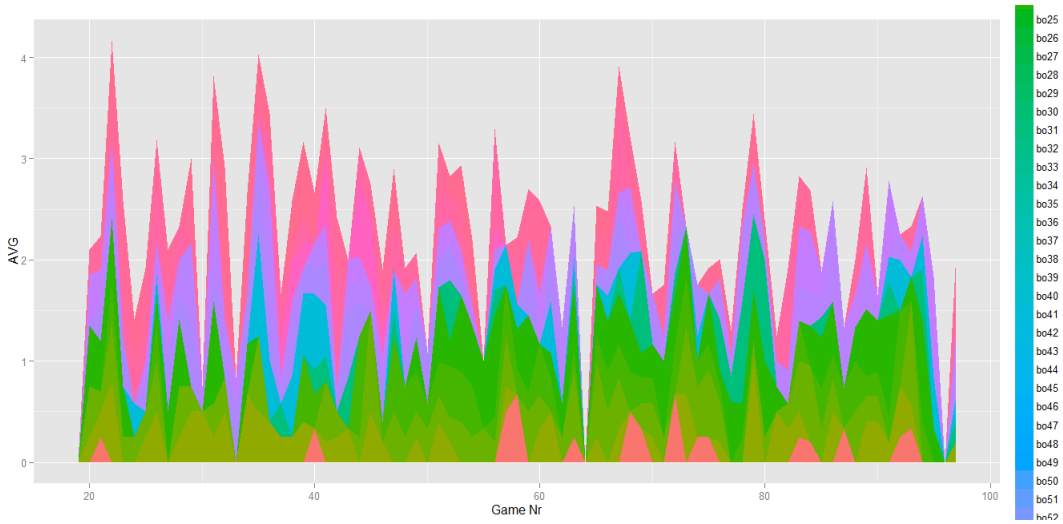


Figure 3: Boston Red Sox, Batting Average, Games 19-99

4 Data Exploration (Paul)

4.1 Aggregate Team Data

As already mentioned, the team dataset contains aggregated offensive statistics for MLB teams over several seasons. Therefore it offers information about the structure of the league, the performance of different teams and trends across several seasons.

We are especially interested in runs scored ("R") because it is the essential measure in order to win games: The team that scores most, wins.

Runs scored per game and per team in the MLB

Min.	1. Quart.	Median	Mean	3. Quart.	Max.
3.167	4.285	4.593	4.624	4.957	6.037

In figure 4 the average runs per game of all MLB teams are plotted by season. We notice surprisingly large differences between runs scored in the league in different seasons: The MLB's median differs from about 5.14 in 2000 to 4.17 in 2013. That's an

decrease in runs of about 19% even in the aggregated data. Altogether there seems to be a downward trend in runs per game during the 13 years.

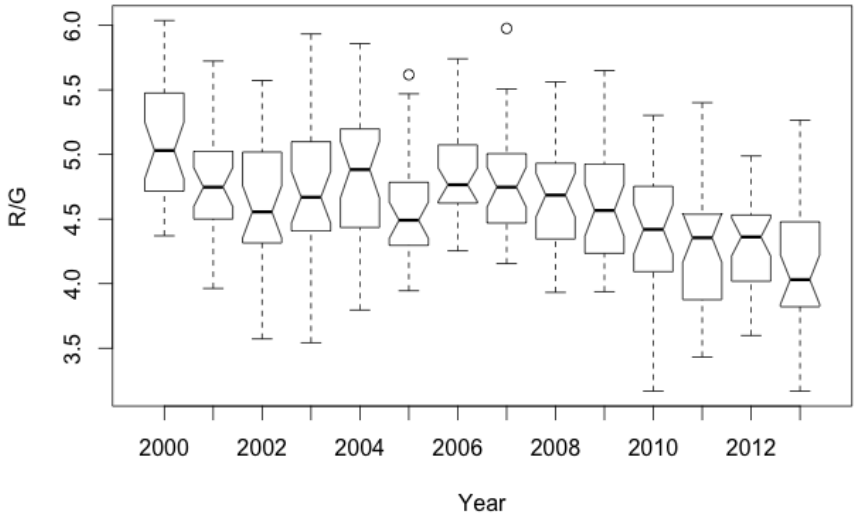


Figure 4: Average runs per game of MLB-teams: 2000-2013

It would be also interesting to get to know if the teams in the league differ systematically in means of runs scored. Therefor we plot the total runs per season of the five Teams of the American League East which are the New York Yankees, the Boston Red Sox, the Toronto Blue Jays, the Tampa Bay Rays (before 2008: Devil Rays), and the Baltimore Orioles (figure 5).

Apparently the Yankees and the Red Sox overall score more runs than the other three teams whereas the Bay Rays and the Orioles often come out last. We are talking about differences between the best and the worst team of roughly 220 runs per game which is about 30% of the median of the MLB's median of 744.

We suppose that there are systematic differences in performance of the different teams. Nevertheless also the variability of a single team's runs from one season to another is quite large: Several times there are changes in about 150 runs from one season to another.

The fact that the MLB's teams differ in their abilities can also be shown using other measures such as slugging averagen (SLG) in figure 6. The three outliers on the top end are the Yankees, Texas Rangers and the Boston Red Sox. The least performing team is Maimi Marlings.

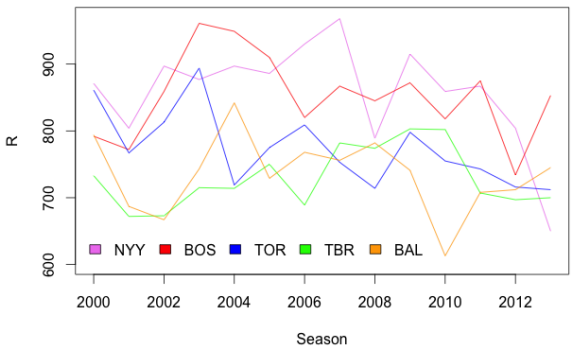


Figure 5: Runs scored per season by teams of the AL East

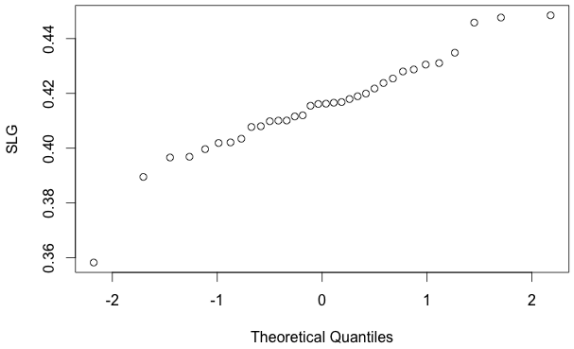


Figure 6: qq-plot: Slugging Average of all MLB-teams

4.2 Individual Player Data

The individual player data consist of statistics of 75 different players playing in 1817 games over 11 seasons for the Boston Red Sox. Hence it is well suited to analyze the performance of different players and more detailed time series.

Since a single player has a limited number of plate appearances in one game most data consist of discrete values and quite often take the value 0. An example is hits per game figure (figure 7): The majority of all players have none or one successful hit per game. It happens very rarely that a player scores more than 3 hits in one game. The plot also shows to us a small home field advantage on hits because the players hit slightly more successfully during home games.

An other interesting question is how to use this data in order to compare the performance of different players. That is particularly difficult since baseball is a team sport and different players contribute in different ways to the teams success. An example of

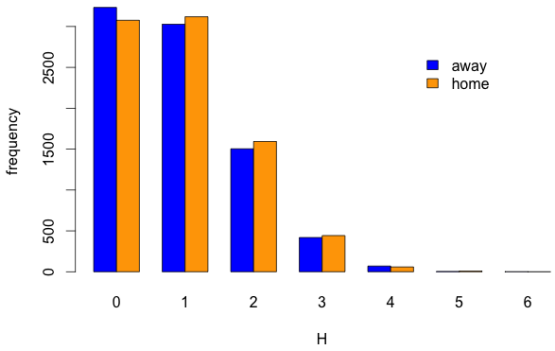


Figure 7: Hits per game for away and home games

in our statistics different player types can be illustrated by the fixed batting order of a team.

In order to score runs a team first needs players to get on base by scoring hits, getting bases on balls or to be hit by a pitch. Players who are good at doing these things will be among the first to be at bat. Once players are on base they have to be advanced by hits, walks, errors or -most effectively- by home runs. These are the tasks of an other type of players who is able to hit balls strongly. It is difficult to split the players in different groups of player types because there are also some that are very good at both getting on base and hitting and some that perform less in both aspects.

That's why we would like to show this point by an example with two players: Jacoby Ellsbury who is first in the Red Sox's batting order and J.D. Drew who usually starts somewhere in the middle of the row.

Figure 8 plots at bats per game for both of them. It is obvious that Ellsbury gets considerably more at bats in the course of one game since he is the first one at bat. That means he gets more opportunities to hit and to get into the game than Drew. For that reason various offensive measurements such as batting average, slugging average and on-base-percentage take at bats into account. Looking at these different measures for both players we notice that indeed the two players contribute in different ways: Ellsbury has a higher batting average which means that he gets more often on base after hitting a ball than Drew. On the other side Drew's slugging average is higher because he hits balls stronger and runs more bases per at bat.

	Ellsbury	Drew
AVG (=H/AB)	0.2912	0.2704
SLG (=TB/AB)	0.4049	0.4762

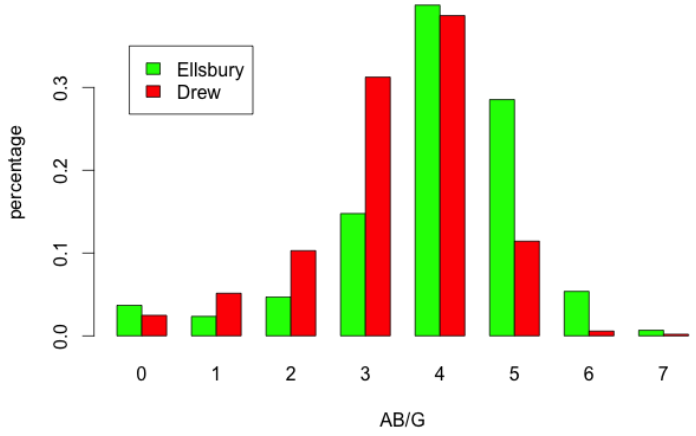


Figure 8: At bats per game for Jacoby Ellsbury and J.D. Drew

5 Theoretical Framework (all)

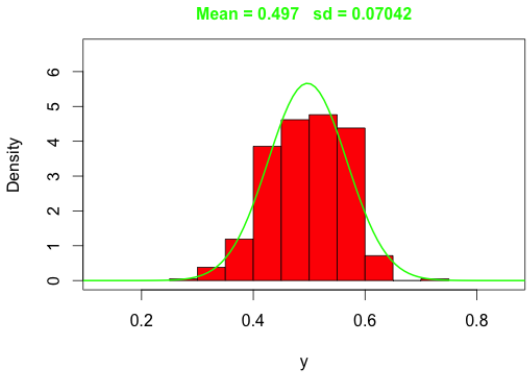
In the following, we will propose a model to simulate a season. The model proposes a framework, where every team is assigned an ability, which is a relative value, normally distributed accross the league. The abilities will create winning propabilities, which are the basis for a game simulation. Like in most games of sports, chance will ultimately decide who wins the game.

In order to estimate player's contribution to a team's ability, we use our set of statistics. We found an appropriate proxy for team performance, TTA, so we construct a regression model that estimates player performance parameters, PTA, from our dataset, using this proxy. The resulting PTAs are on average equal to TTA, so we interpret individual deviation as a difference in player contribution. This difference can be used for estimating the effect of changes to the team roster.

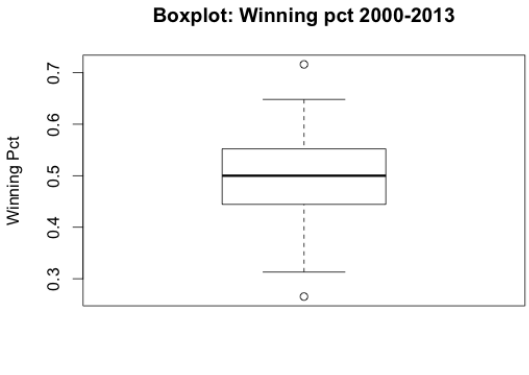
5.1 Theoretical ability as winning propability (Max)

In order to simulate a season, we use a model proposed by Albert and Bennett (2003). The model describes the team's performance as an ability.

We can observe in figure 9 that the median in the distribution of the winning percentages is at 0.5 and the mean is 0.497, showing that an average team will win half its games. The distribution is drawn out of a data set, which contains all the winning percentages from the seasons 2000-2013 with 420 observations. The intervall of the first and third quartile is [.444,.552]. Detroit had the worst winning percentage, in 2003 with .265 and Seattle had the best with .716, in 2001.



(a) Distribution of winning Percentage with fitted Normal Distribution



(b) Boxplot of winning Percentages

Figure 9

But what does this tell us about the performance of a baseball team? Most of the teams will be situatuated within an intervall of the first and third quantile, so they will perform around the average value of $\approx .5$. Teams, which perform above this value, can be considered as good teams, the others can be considered as bad teams. Finally, we can see in figure 9 that performance in the MLB is following a normal distribution. We have good and bad teams, but the average team will perform around the .5 mark.

If we say the performance of a team follows a normal distribition, we can create a general model to simulate a league. Ability is a parameter that identifies the talent a team has. This is a relative value, normally distributed with $\mu = 0$ and $\sigma = .19$ on an interval of $[-1, 1]$. This is an implication of the observation made above. An average team has the ability of $\mu = 0$. Teams having an above average ability will be greater than 0 and should win more than half their games; teams, which have an below average ability, will have an ability smaller than 0 and should lose more than half their games. The standard deviation of the distribution is taken from an approximation made by Albert and Benett (2003).

Figure 10 shows how the abilities from 30 teams are distributed randomly. 30% of the teams have an average ability, while 25% are good and 10% of teams have an excellent ability. On the other side. 25% of the teams have a poor ability and 10% of the teams have an ability considered as bad.

The Simulation (Max)

The simulation of a season is done by calculating winning propabilities and simulate a game, by tossing a weighted coin, using a simple binomial distribution. The propabilities

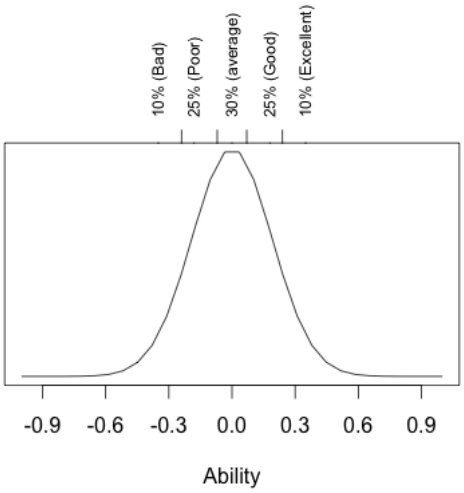


Figure 10: Normal distribution of team abilities

characterize the nature of the game. Of course winning a game depends on the talent or ability of the team to play baseball. But just as in most other sports, there is a role of chance of every team to win a game, even the team with the least talent can beat the most talented team, thus winning can be approximated as a biased game of chance, which can be characterized by a weighted coin toss. In order to obtain the propability to win a game, we have to transform the ability parameter into a strength parameter:

$$s_i = e^{a_i} \tag{1}$$

The strength of a team is nothing else than the exponent of a team's ability, which allows our abilities to be positive numbers. These strength parameters are then used to calculate the propability to win a game:

$$Prob(\text{A defeates B}) = \frac{s_a}{s_a + s_b} \tag{2}$$

Lets assume that team A is a poor team with an ability of -.15 ($s_A = 0.861$) and team B is an excellent team with an ability of .26 ($s_B = 1.297$). The propability that team A will defeat team B is therefore 0.399. If both teams play in the same division, they will have 19 games in a season. The random coin toss for all meetings in 10 seasons can be simulated with R using the following command:

```
rbinom(n=10, size=19, prob=.399)
```

Table 2 shows that the role of chance ist still playing a big role in our model. In some seasons team A won far more games then they were expected to. Their simulated winning percentage is therefore .22 higher than their winning propability. In other words

Season	Wins
1	10
2	9
3	5
4	8
5	6
6	13
7	7
8	9
9	7
6	10
Win pct:	.421

Table 2: Simulation results for 10 seasons.

team A was more lucky than team B.

After we obtained an empirical ability for the Boston Red Sox, we will simulate a number of seasons for the Boston Red Sox using the same model proposed above.

5.2 The Player Contribution Percentage (David)

It is widely held, and we share that belief, that even though baseball as a game focuses on individual performance, in the end the runs, the game results, depend on a team. Because the game results and the individual results are what is available to us, we propose a way to use them to calculate a measure for individual player's contribution.

We assume that all players contribute differently to each game, but their average contribution is equal to team results. We find Team Total Average (TTA) to be the best proxy for team results, so we expect Total Player Average to be the best proxy for player results. Instead of runs, we regress the statistical player values on the calculated TTA, because runs are too much directly dependent on player performance (for example, all players Runs Batted In mostly sums up all runs). The individual player's relative deviation from the team average could now be an indicator for their contribution at each game (the Player Contribution Percentage (PCP), where 100% equals the TTA). The individual players parameters could indicate a profile of their performance, but we expect heavy multicollinearity to obscure this effect, all the while leaving the player total average (PTA) intact. To make the fitted values a real average, we have to account for the number of players per game in the regression function. Here is a formal description:

$$X\beta = p\mathbb{1}y \tag{3}$$

$$X = (X_1 \vdots X_2 \vdots ... \vdots X_{75}) \qquad \beta = \begin{pmatrix} \beta_1 \\ \beta_2 \\ ... \\ \beta_{75} \end{pmatrix} \qquad y = \begin{pmatrix} TTA_1 \\ TTA_2 \\ ... \\ TTA_{1817} \end{pmatrix}$$

$$(1817 \times (75 * K)) \qquad ((75 * K) \times 1) \qquad (1817 \times 1)$$

X_i = statistics needed for total average of player i in 75, dimension 1817 ×K
β_i = unknown weights K for total average of player i in 75, dimension K ×1
y = empirical total team averages for games 1-1817 (calculation see **??** Probability Model)
p = player count, each element is the number of players involved in game j, dimension 1817 ×1
$\mathbb{1}$ = identity matrix, dimension 1817 ×1817

$$x_{j,i}\beta_1 = PTA_{j,i} \tag{4}$$

$$\frac{PTA_{j,i}}{TTA_j} = PCP_{j,i} \tag{5}$$

$$\frac{1}{p_j} \sum_{i=1}^{p_j} \frac{PTA_{j,i}}{TTA_j} = \frac{TTA_j}{TTA_j} = 1 \tag{6}$$

j = game index
i = player index
p_j = number of players in game j
$PTA_{j,i}$ = player i total average in game j
TTA_j = team total average in game j

We propose that the fitted values of regression (1) will show the player specific contribution, relative to the team total average (TTA) in game j. The formal criterion for a relative contribution value is shown in (4), all player's contribution percentages (PCP) average to 100% of the TTA. Further research would analyse the resulting PTA values to determine the best game period for regression. A very long period might give too much weight to the past, especially if the position of a player has changed. The latest PTA should be the most relevant for todays decisions, but we could add older values with a discount factor. Similarily, should we predict future values and add their discounted values? The last task would be composing a present value function that includes both past values and future estimates.

5.3 Hiring the New Guy (Paul)

The player contribution analysis can also be used in order to measure a player's possible contribution before joining a new team. The results might be valuable for a team

manager's decision whether to sign a contract with a new player and how to determine a reasonable salary.

Therefore we recall that in the player contribution analysis we calculated individual regression weights for each player. In doing so we considered that different players contribute in different ways to the team results. That is why we can assume that the PTA as a proxy for player's offensive performance is independent from the performance of his team mates and won't change if the player is transferred to another team. Although this assumption obviously is a simplification, it seems reasonable when we suppose that a player is always replaced by another one with a similar profile and in the same position.

When we are thinking about replacing our old player q by a new player p, we can calculate the new guy's PTA using past data from his old team.

PTA_p = PTA of player p who will join our team
TTA_t = TTA of our team t before the transfer
TTA'_t = TTA of team t after the transfer (with player p and without player q)
n = Players playing for team t before the transfer

$$TTA_t = \sum_{i=1}^{n} PTA_i \tag{7}$$

$$TTA'_t = \sum_{i=1}^{n-1} PTA_i + PTA_p = TTA_t + PTA_p - PTA_q \tag{8}$$

$$\Delta(TTA_t) = TTA_t - TTA'_t = PTA_p - PTA_q \tag{9}$$

$\Delta(TTA_t)$, the difference between the leaving player's PTA_q and the arriving player's PTA_p, indicates wether the TTA_t of our team will increase or decrease after the transfer. Using the future PCP of the new man in our team and knowing our budget we can define a reasonable salary we would be willing to pay for him. Using TTA'_t we can also calculate the effects of the new player on the winning chance of our team. In chapter 7 we simulate complete season outcomes. It is possible to use this simulation model and estimate the effects of the transfer on the team's performance.

6 Implementing the Models (Max)

In order to find the best proxy to show runs per game, we will use aggregated team data per season from mlb.com. The reason we use team data is trivial; a run is always the sum (excluding the case of a one run home run) of individual performance. E.g. a batter hits a double, the only way he will score is through another player scoring a hit in the inning, so he can advance to the home plate and score a run.

The data is from all major league teams, from the seasons 2000-2013. Since there are 30 teams we have 420 observations for every official parameter in the statistic. We took this sample, because it is the current performance of the teams, therefore excluding significant changes in the way the game is played. On the other side, we have enough observation to conduct a thorough analysis in the correlations of the parameters and in the correlation of the weighting in the TA. All calculations used in this section are done by R and can be traced in the R file "Finding_best_proxy.R".

6.1 A Proxy to measure Offensive Performance

One of the main questions in Baseball literature is, how to obtain a good proxy to explain runs scored per game. In order to find this proxy, we will look at correlations of different offensive parameters to runs scored per game. We will take a look at the official measurements, constructed by the Major League Baseball:

- Batting Average: $avg = \frac{H}{AB}$
- Slugging Percentage: $slg = \frac{1 \cdot S + 2 \cdot D + 3 \cdot T + 4 \cdot HR + 1 \cdot}{AB}$
- On Base Percentage $obp = \frac{H + BB + HBP}{PA}$

It will be shown in 6.2 that slg and obp correlate far better with runs per game than avg.

Thus, it might be smart to find a parameter, which combines the ability of getting on base and the ability of hitting the ball hard, depicted in slg. Therefore sportswriter Thomas Boswell created the parameter Total Average (TA) in 1981, which is nothing else than total bases divided by the number of outs:

$$TA = \frac{1 \cdot S + 2 \cdot D + 3 \cdot T + 4 \cdot HR + 1 \cdot BB + 1 \cdot HBP + 1 \cdot SB}{AB - H + CS + GDP} \tag{10}$$

As shown in the equation above, TA puts a weight on each of the equation's elements, saying that HR contributes 4 times as much to the TA than a hit and therefor to the runs scored. It also implies that a walk or hit has an equal contribution than a hit. This might simplify the model, but it is not very convincing, as a hit is qualitively not equal with a base on balls. The underlying argument is that a hit will always advance runners on base at least one base no matter, where these runners stand, a base on balls will only advance other baserunners, if they are in a force situation, i.e. they have to move on one base, in order to clear the base they are on right now. E.g. we have a runner on first, the batter draws a walk or gets hit by a pitch. The runner on first has to move on to second, to clear first base for the batter. Would the runner be on second base before, he would have stayed there, if the runner would have hit a base-hit, the runner would have been able to advance to third base. So the contribution of scoring a run on a walk is far less than with a base-hit, though in both cases the batter runner gets to first base.

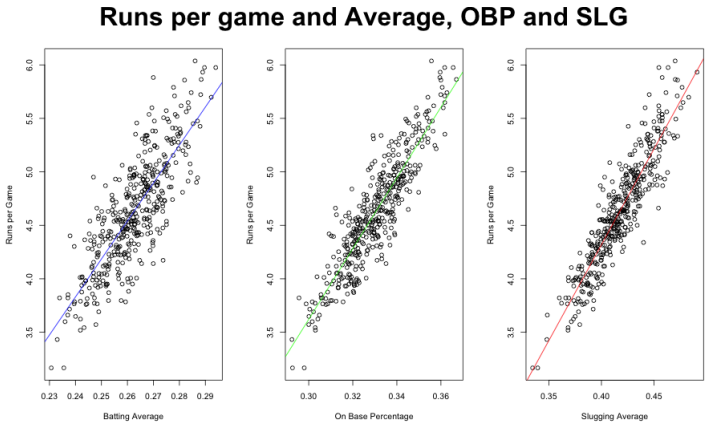


Figure 11: Scatterplot of the 3 official mlb parameters

Therefore, we will reweight the elements of the TA by regressing it with the runs scored per games:

$$TA = \frac{w_s S + w_d D + wtT + w_{hr} HR + w_{bb}(BB + HBP) + w_{sb} SB}{G} \tag{11}$$

We assume, that the number of outs in a season are the same across the league, so we can use games instead of total outs in the denominator. We combine base on balls and hit by pitch, as the effect is the same. By regressing the elements of TA to runs scored per games, we are able to get a better fit, and therefor a better proxy to estimate runs. The regression results are shown in 6.4.

6.2 The Proxy for offensive performance

As one can see in figure 11, the parameter slg seems to have the best fit to the runs scored per game, closely followed by obp. This can also be seen in the correlations: Avg has a correlation of .811, OBP of .899 and SLG of .915. As one can see the power parameter slg and the getting on base parameter obp have a similar correlation with runs scored per game. So the fit might be increased if we find a parameter that combines both. As described in the section above, the measurement of TA is one of these parameters. The correlation of the TA with runs scored per game is .95, and therefor substantially higher than the other proxies.

6.3 Set-up of a fitted TA

As shown in 6.1, the fit of the TA could be improved, if we reweight the weights of the TA's elements. The basis of our regression model is the reweighted equation (11). The weights are the independent variables, which are used as regressors. The dependent variable in our regression will be runs per game, as we try to find the best fit of the

weights, to explain runs per game.
Our regression model will have the following the form:

$$RpG = \frac{w_s S}{G} + \frac{w_d D}{G} + \frac{wtT}{G} + \frac{w_{hr} HR}{G} + \frac{w_{bb}(BB + HBP)}{G} + \epsilon \tag{12}$$

The regression should be without an intercept, as the runs scored per game should be 0, when the TA is 0.

We expect that all the independent variables, have a significant impact on the dependent variable, as all the variables contribute to a run being scored. Beyond that, the weights should increase with the total bases, and a single should have a higher weight on runs scored per game than the bb, hbp and the stolen base parameter. The weights should then be used to calculate a reweighted TA, which has a better fit to runs per game.

6.4 Output of the fitted TA

Regressing for the weights of the TA from (12), R produced the following output:

	Estimate	Std. Error	t value	Pr(> |t|)
S	0.2385	0.0200	11.90	0.0000***
D	0.4624	0.0698	6.62	0.0000***
T	1.0237	0.2101	4.87	0.0000***
HR	1.3188	0.0615	21.45	0.0000***
BB	0.2251	0.0267	8.44	0.0000***
SB	-0.0024	0.0599	-0.04	0.9676
Residual standard error: 0.2229 on 414 degrees of freedom				
Multiple R-squared: 0.9977, Adjusted R-squared: 0.9977				
F-statistic: 3.043e+04 on 6 and 414 DF, p-value: < 2.2e-16				

Table 3: Regression results for weights without intercept

As one can see in table 3, all the regressors with the exception of stolen base are highly significant. The fact that stolen base is not a significant variable, makes the model problematic. A stolen base brings a runner into scoring-position and increases the chance of scoring a run. In order to get an output, which shows a significant stolen base regressor, we will modify the model (13) and add an intercept.
Regressing for the weights of the TA with an intercept, R produces the following output:

The output in table 4 produces significant values for all independent variables.

Comparing the two outputs

In the following we will use the regressed weights and see, which of the reweighted TTAs has a better fit on runs scored per game.

	Estimate	Std. Error	t value	Pr(> |t|)
(Intercept)	-3.2198	0.1320	-24.40	0.0000***
S	0.5665	0.0186	30.47	0.0000***
D	0.7626	0.0464	16.43	0.0000***
T	1.2432	0.1349	9.22	0.0000***
HR	1.4827	0.0400	37.09	0.0000***
BB	0.3398	0.0177	19.17	0.0000***
SB	0.1473	0.0389	3.79	0.0002***
Residual standard error: 0.1429 on 413 degrees of freedom				
Multiple R-squared: 0.9246, Adjusted R-squared: 0.9235				
844.4 on 6 and 413 DF, p-value: $<$ 2.2e-16				

Table 4: Regression results for weights with intercept

When comparing the fit of the reweighted regression results on the runs scored per game, we can see that the weights produced by the regression without an intercept explain runs scored per game far worse, than the weights received from the regression with an intercept. This results from the inclusion of stolen bases into the TA. The correlation for the weights from the regression without an intercept is .903, with an intercept it is .961 and without reweighting it is .95. Therefore the values of the regressors from table 3 are the worst to explain runs scored per game and the values from table 4 are the best.

This observation does also hold when taking a look at the mean squared errors of the

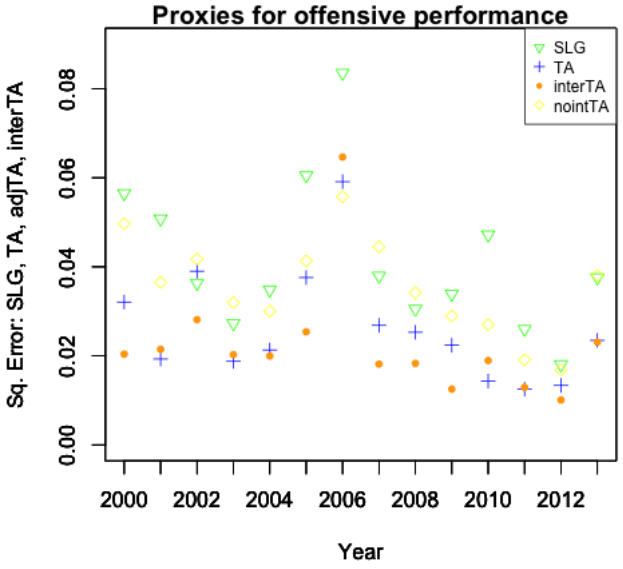


Figure 12: Comparison of the residuals to the fit of TA, SLG, reweighted TA with and without intercept

proxies for every season, depicted in figure 12. In 9 of the 14 seasons, our regressed

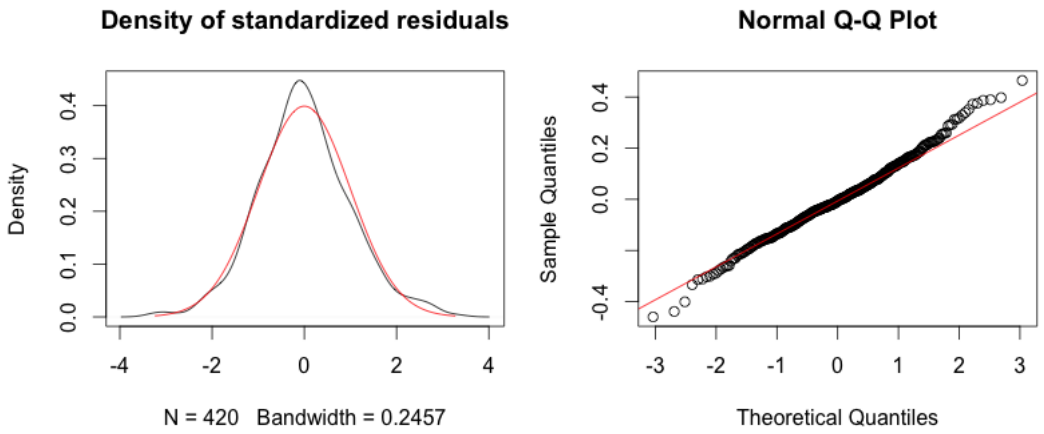
values from the regression with an intercept explained runs scored per game the best. In 3 seasons the unmodified TA was the best proxy and only in one season the weights from regression without an intercept, where the best to explain runs scored per games. The official statistic of slugging was the worst in all the seasons.

From that we can conclude that the best proxy to explain runs per games is the reweighted TA. The weights should this be acquired by using the following linear model:

$$RpG = w_0 + \frac{w_s S}{G} + \frac{w_d D}{G} + \frac{wtT}{G} + \frac{w_{hr} HR}{G} + \frac{w_{bb}(BB + HBP)}{G} + \epsilon \tag{13}$$

6.5 Robustness Checks

In the following, we will show that the regression is robust and holds the assumption of normally distribution of the errors.



(a) Distribution of residuals in the regression for weights with an intercept and fitted normal distribution

(b) QQPlot of residuals and the fitted QQline

Figure 13

Figure 13 suggest that there does not seem to be any significant trend in the plotted residuals and that an abnormal distribution of the rediduals seems to be unlikely. That our residuals follow a normal distribution is shown here in a density plot of the empirical residual distribution and the QQ-Plot.

As the residuals follow a normal distribition, we can use the weighted total average parameter (using the weights from table 4) as our best proxy to measure offensive performance. Of course this parameter can also be used on an individual basis to measure individual offensive performance. This means, we can use individual TA data and ag-

gregate this. This will yield to a TTA, which can be used to forecast team performance, as it will be shown in the following section. As baseball is a highly individualistic sport, one can also include TA data of players, who played for a different team and aggregate it into the TTA, therefore prognosing the effect of a trade.

7 Simulating seasons of the Boston Red Sox (Max)

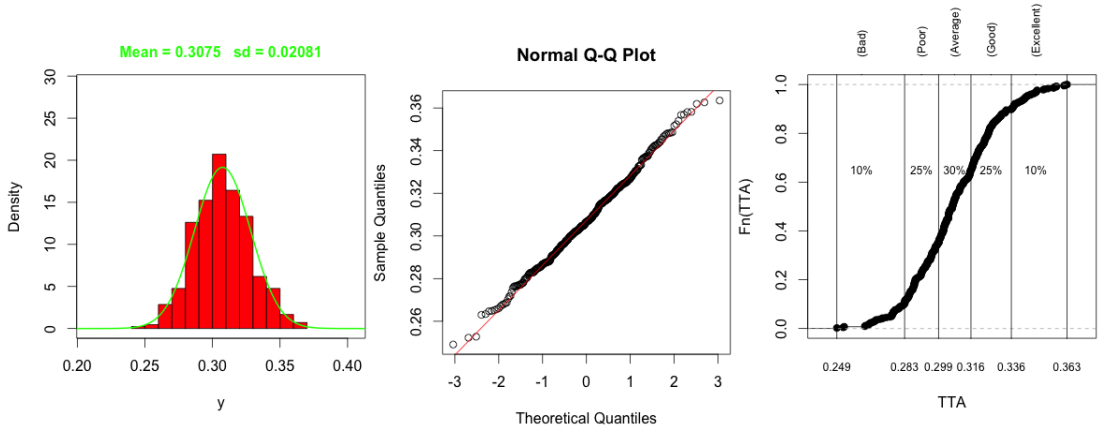
As we have shown in 5.1, winning percentages can be translated to ability, which allows us to calculate winning propabilities and simulate games by throwing a weighted coin. In the following we will create an empirical ability for the Boston Red Sox. The basis of this ability will be Team Total Average (TTA), which is our best proxy for offensive performance, aggregated from the individual player dataset. We simplify the model, by leaving defensive performance outside of our analysis. When simulating the seasons, we will create a random, normally distributed vector of abilities for the other teams. Nonetheless, one can also create an empirical measurement of the ability through the TTA for all the other teams with the data from mlb.com. Goal is to show that aggregated TTA data is suitable to be transformed into an ability value proposed in 5.1. The calculations that aggregated the Boston Red Sox 2007 TTA can be traced in the R file "Aggregation_of_TTA.R". The simulation can be traced in the R file "Season_Simulation.R".

Translating TTA into ability

Before simulating the season, it should however be clarified how TTA is translated into abilities. We will use a slightly different TTA than the one which is shown in 6.1. Here we will divide the weighted total bases through the total number of outs, leaving GIDP out of the calculation (AB-H+CS), as this will enable us to use single player data set and we can therefore aggregate a TTA from every game. Though the values are different, there wont be any difference in the relative distribution of the TTA across the league, as GIDP should be more or less equal for all teams in one season.

To receive an empirical distribution of the TTA, we will use the Team Aggregate Dataset. Here 420 TTAs with total numbers of outs in the denominators are regarded. The sample has a mean $\mu = .3075$ and a standard deviation of $\sigma = .02081$. Figure 14 shows the density plot, QQplot and the distribution of the TTA. It shows clearly that the observations follow a normal distribution, just as the abilities are supposed to.

As the abilities are not distributed between .25 and .36, but on the intervall [-1,1], one has to find a function, which is able to translate the TTA into an ability. As the mean of the abilities is 0, we standardized the TTA values. The resulting vector was distributed normally on the interval [-2,2]. To create a distribution on the intervall [-1,1], we simply divided all the values by 2. Though this method seems crude, it is highly effective. The calculated data has a mean $\mu = -.0007$ and a standard deviation $\sigma = .3702$ and follows the normal distribution.



(a) Empirical and Fitted Densitiy of TTA

(b) QQ-Plot of TTA

(c) Empirical Distribution Function of TTA

Figure 14

We can therefore say that the approximated ability function dependendent from the TTA can be formalized in the following way:

$$ability(TTA) = (\frac{TTA - 0.3075}{0.02081}) * 0.5 \tag{14}$$

One might notice that the standard deviation of the ability distribution transformed from the empirical TTA values is much higher (.37) than the one proposed by Albert and Bennet (.19). The reason for that might lie in the fact that we only consider offensive data. Offensive performance is more widely spread than winning percentages, as teams, who have weak offensive production, compensate this flaw by focusing and investing in a strong pitching staff. As we leave the defensive side of the game out of our analysis, we use the higher standard deviation derived from the TTA. Analyzing the defensive side of the game will be matter of further research.

7.1 The Simulation

In our simulation, we will show how the ability calculated from the aggregated TTA of the Boston Red Sox 2007 faired in a simulated league playing a hundret seasons. The schedule they played is equal to the one from 2008. The other teams were assigned random ability following a normal distribution with the parameter shown in figure 15. The Boston Red Sox of 2007 had the highest ability with .5878, which is not unlikely, as they won the championship that year. On average the team won 61,59 % games per season. The winning percentages were spread on an interval of [51.23%, 69.75%], so not even in the worst case scenario a team with such a high ability would have won less than half their games. For a detailed sample of hour simulation result look at (5) in the appendix or the R-code. As the mean in the winning propabilities is lying at 0.657, one can say that the Red Sox underachieved in the 100 seasons, as they won less games

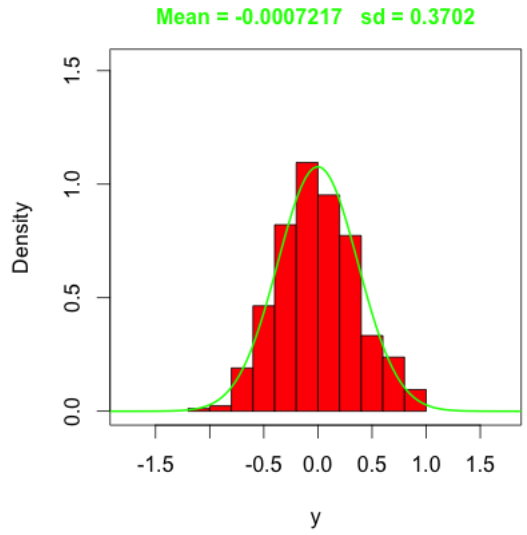


Figure 15: Empirical distribution of abilities translated from TTA

than their ability would forecast. This shows how big the role of chance is, even if you simulate a larger number of seasons.

8 Prognosis for the Player's Performance (David)

In order to make valid proposals for a valuation parameter it is impossible to ignore the future. We could just leave our metric as a snapshot valuation, but since we worked hard to produce time series data, we wanted to at least get an idea of the career development of players offensive performances.

8.1 Career Highs and Lows

Looking on time series plots like this (total hits per game), we first assumed all players to exhibit one career peak, with high performance in the middle, and medium performance towards the beginning and end of a career.
While this is true in general, there are indicators like seasonal averages (8 game batting moving average) that show ups and downs inside the middle, creating something like a career range with performance peaks and valleys. How to know where a player is standing at the moment?

In order to better understand performance relative to career maxima we decided to compare all players arranged on said maxima. The resulting plots offer interesting insights. Here are three samples: These plots show the distribution of players performance

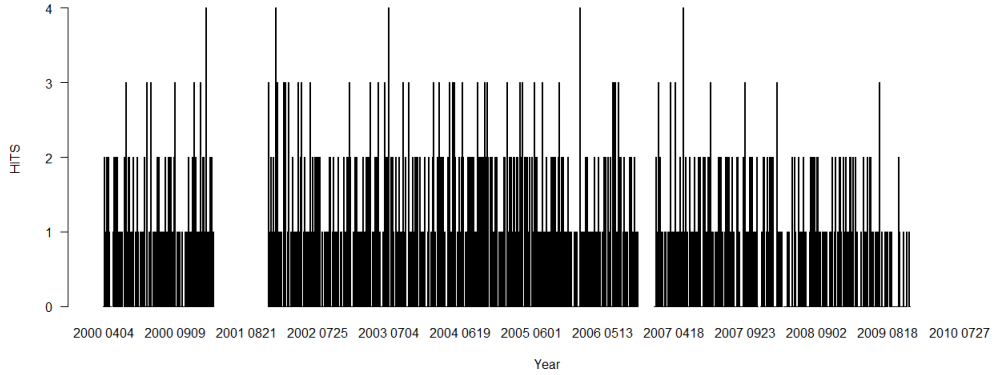


Figure 16: Jason Varitek, Hits per Game, total dataset

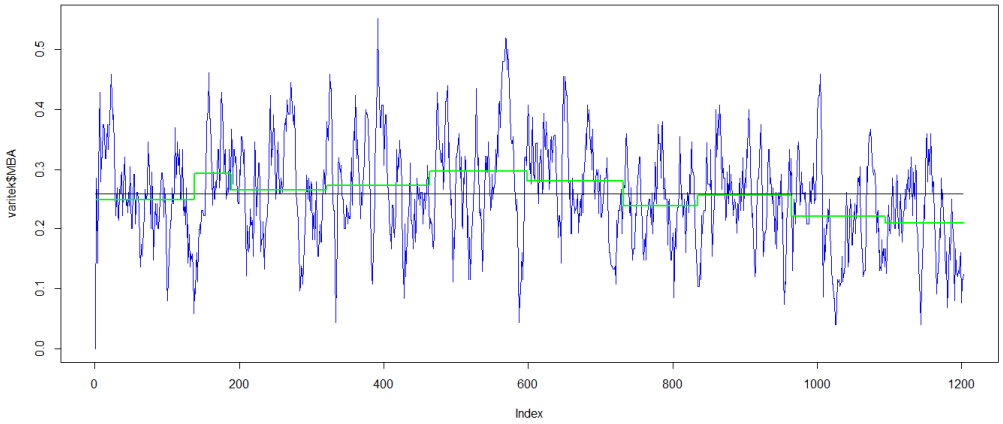


Figure 17: Jason Varitek, 8 game batting average, without missing values

statistics compared to their respective first maximum. The above plot (hits) leads us to believe that performance is generally right skewed, meaning players perform closer to their maximum after they have reached it in an early stage of their career. This however is not so obvious in the middle plot (slugging average). It rather indicates a small performance hill about a hundred games before the maximum, or a late career maximum! The last plot shows absolute Runs Batted In, arranged on the first maximum and again seems to indicate an earlier career maximum. In conclusion, comparative career performance analysis can help to find typical career "mountain range" shapes, typical performance development and in time allow for making predictions about the future player performance. Particular weight should be given to choice of variables. Runs Batted In are far more valuable than mere hits, home runs are a great measure for hitting and running ability, but only occur very rarely. We would like to note that overplotting is not a problem in these plots, as we only care about maxima. The plots serve as an

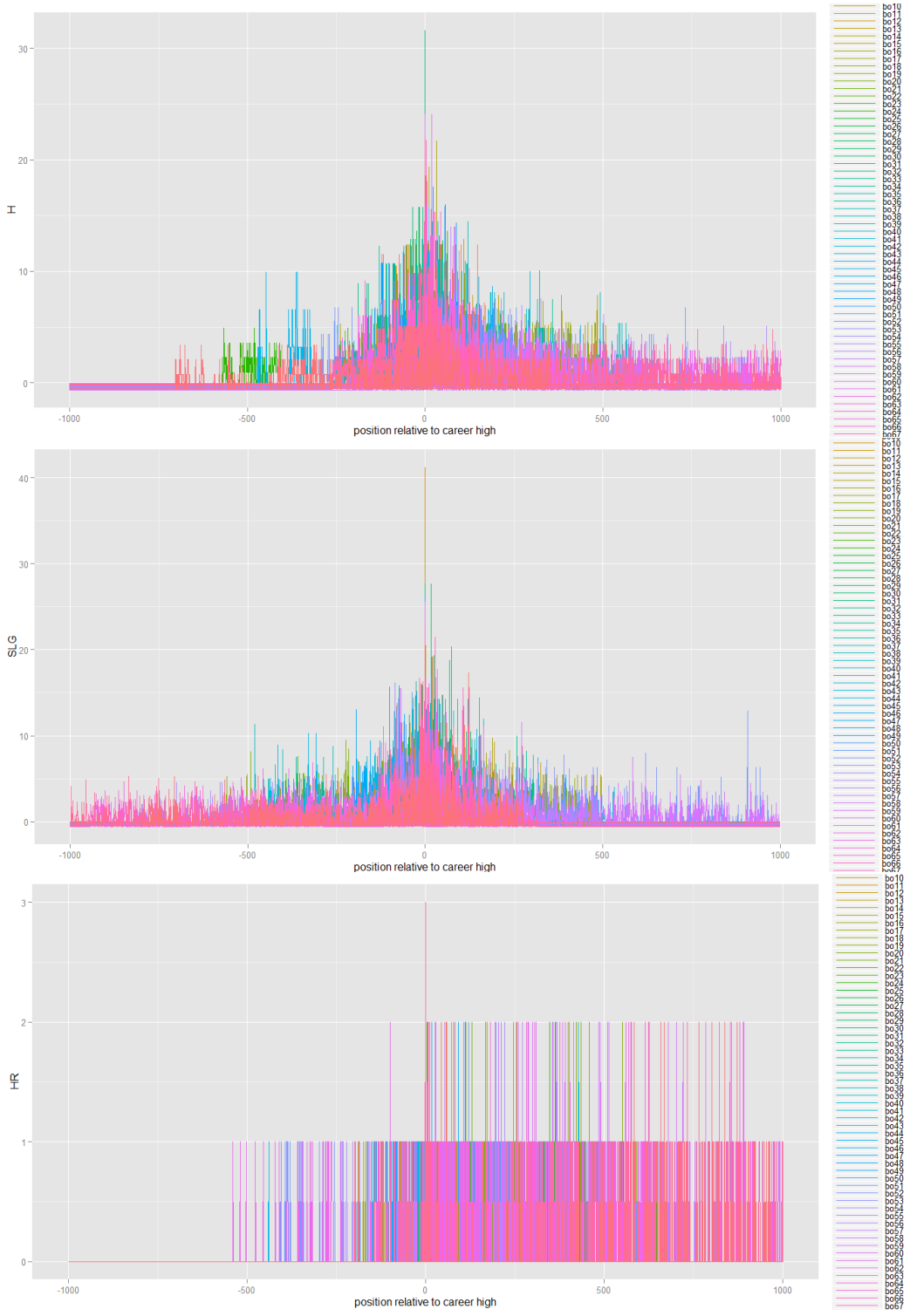


Figure 18: above: all players, hits, normalized, arranged on first career maximum middle: slugging average (bases reached per at bat), same specifications below: all players, home runs, arranged on first career maximum

orientation, significant inference will have to rely on numerical tests.

9 Results (Paul)

9.1 Regression results

The best proxy for team offensive performance we find is team total average ("TTA"). For that purpose we reweighted total average with regression results for seasonal aggregated team data. The resulting TTA explains runs per game better than any other measurement available to us. Robustness checks do not report any problems.

We follow the theoretical model of Albert and Bennett of team ability and simulating seasons but use our own empirical TTA as a base to identify ability for the Boston Red Sox in the season 2007. By transforming TA measurements into abilities, one can aggregate individual information into a TTA and team ability. Although our calculations overestimate the empirical spread in winning percentages, due to a higher standard deviation (supposedly because we have to ignore defensive performance) the results indicate relative placement in the league quite well.

Using individual player data we suggest a measure for player ability based on contribution to the team performance approximated by TTA. Since the resulting measure, the player total average ("PTA") is individually weighted for every player we suppose it to be independent from the performance of the team mates.

Taking as well player as team total average into account we can estimate a new player's contribution and effects on the team success before he enters the team.

9.2 Prognosis results

The arranged time series datasets permit to plot different individual offensive measurements for every game of the Red Sox. This can give an idea about typical player careers and is useful in order to compare past performance development of different players.

An important aspect of the time series prognosis is the indication of highs and lows in a player's career which is relevant when interpreting performance variabilities in the recent past.

9.3 Explanation Power of our Model

The regression models and the results of our work are consistent with the data at our disposal. Though the dataset we collected only covers 11 seasons of the Boston Red Sox, it has great potential for further analysis.

The data available to us prove TTA clearly to be a good proxy for offensive team performance and the models used for winning abilities and simulations are in line with current literature. They enable us to model team offensive performance and the role of a single player in a teams offensive pretty well.

When it comes to simulations of complete seasons and changing players we reach the limit of what can be clearly proven with our data and rely rather on intuition while

working in a more theoretical way. In order to simulate the season of the complete league with the empirical measurements we suggested, detailed and complete informations about all players in all teams would be be necessary. Another handicap is that we only dispose of offensive statistics and that presumably a teams ability depends as well on its defensive power. That is also the reason why it is not possible to empirically prove our findings on hiring new players: We lack player data from other teams and cannot explain impact of defensive performance on player prices.

The main achievement concerning the time series prognosis is the data transformation. Still it offers a very useful tool to get an overview and a first impression of the performance of single players in the Red Sox team across several games and seasons.

9.4 Outlook

The data necessary to resolve the drawbacks mentioned in this chapter, is easily accessible on mlb.com and we already provide the methodology necessary to deepen the empirical analysis. Hence we are convinced that -especially after some further research- our results will be valuable for baseball teams and sport fans.

The next step would be to implement the model of the player contribution percentage, proposed in 5.2. The empirical implementation of the model was not done in this paper, as it would go beyond the scope of this paper, however it should be implememented soon, as it yields a way to value players and can serve as a method to estimate a fair player salary.

A R Code

The following are R-functions used in data preparation. For the full code and do-files to replicate our data we will readily provide you with all the files necessary.

A.1 Cleaning

```
###### function to remove "illegal" games
remove.games <- function(x,key) {
 removex <- rep(1,dim(x)[1]) #create removal criterion vector
 for(i in 1:dim(x)[1]) {  #check game in line i with keyfile
  for(j in 1:dim(key)[1]) {
    if(as.vector(x$DATE[i])==as.vector(key$DATE[j]) && as.vector(x$OPP[i])==as.vector(key
    removex[i] <- 0
    }
  }
 for(i in length(removex):1) {
  if(removex[i]==1)     # use vector to remove "illegal" games
    x <- x[-i,]
  }
return(x)
}
```

A.2 Restructuring

```
###### Restructure the DATE variable as GAMENR, by D.Schulze
datstr <- function(x,key){
  start.time <- Sys.time()
  x <- data.frame(x, stringsAsFactors=FALSE)
  i <- 1
  j <- 1
  while(i < 16559 && j < 1818){
   if(x[i,26]==key[j,2]){
    x[i,26] <- key[j,1]
    i <- i+1
    }
   else {
    j <- j+1
    }
   }
  end.time <- Sys.time()
     time.taken <- end.time - start.time
     print(time.taken)
  return(x)
```

```
}
```

A.3 Aligning

```
###### Arrange the player vectors on one DATE and GAMENR vector
arrdat <- function(x,key){  # arrange old data to date
 c <- data.frame(matrix(nrow=609,ncol=148))
 x <- rbindlist(list(x,c))  # add empty rows to make space for sorting
 x <- data.frame(x,key$X, stringsAsFactors=F) # add "date" column
 r <- cbind(c(rep("NA",1817)),c(rep("NA",1817))) # create transport vector
 rc<- r
 i <- 1  # column-pairs (players) in old data
 j <- 1 # date in old data
 k <- 1 # row in old data
 for(i in 1:74){  # for each player of the old data
  lng <- (1817-dim(x[is.na(x[,(i*2)])==T,])[1]) # how many rows (games) with data
  for(j in as.integer(x[1:lng,(i*2)])){  # for each row (game)
   r[as.integer(j),2] <- j  # put the date in the transport vector at the correct point
   r[as.integer(j),1] <- x[k,((i*2)-1)] # same for the statistic (hits e.g.)
   k <- k+1  # next row in the old data
   }
  x[,(((i*2)-1):(i*2))] <- r # insert the transport vector
  k <- 1 # back to row one for the next player
  r <- rc # empty the transporter
 }
 DATES <- key[,2] # clean up the data
 GAMENR <- x[,149]
 x <- data.frame(DATES,GAMENR,x[,-c(seq(2,148,2),149)])
 return(x)
}
```

A.4 Stacking

```
###### Prepare the stacked plots for ggplot2::geom_area
stc <- function(stackfile,name) {
 stackA <- rep(stackfile[,1],74)
 stackB <- stackfile[,-(1:2)]
 stackB <- stack(stackB)
 stackfile<- data.frame(stackA,stackB)
 names(stackfile) <- c("DATE",as.character(name),"PLAYER")
 stackfile[,1] <- as.numeric(stackfile[,1])
 stackfile[,2] <- as.numeric(stackfile[,2])
 return(stackfile) }
```

	90	91	92	93	94	95	96	97	98	99	100
BOSANA	4.00	4.00	6.00	6.00	5.00	2.00	3.00	6.00	4.00	5.00	3.00
BOSARI	3.00	2.00	2.00	3.00	2.00	2.00	2.00	3.00	2.00	3.00	3.00
BOSBAL	12.00	11.00	13.00	15.00	16.00	12.00	15.00	10.00	13.00	10.00	16.00
BOSCIN	3.00	2.00	2.00	2.00	2.00	3.00	2.00	1.00	3.00	3.00	3.00
BOSCLE	4.00	5.00	3.00	4.00	4.00	6.00	4.00	3.00	2.00	6.00	4.00
BOSCWS	6.00	5.00	7.00	4.00	4.00	3.00	5.00	4.00	6.00	4.00	6.00
BOSDET	6.00	5.00	5.00	6.00	6.00	5.00	6.00	5.00	6.00	5.00	6.00
BOSHOU	3.00	1.00	2.00	3.00	1.00	1.00	3.00	2.00	2.00	2.00	2.00
BOSMIL	2.00	2.00	2.00	3.00	3.00	3.00	3.00	3.00	3.00	1.00	3.00
BOSMIN	3.00	4.00	2.00	6.00	6.00	3.00	5.00	5.00	6.00	6.00	4.00
BOSNYY	8.00	11.00	10.00	13.00	10.00	9.00	11.00	11.00	12.00	14.00	9.00
BOSOAK	6.00	4.00	3.00	4.00	5.00	4.00	5.00	4.00	6.00	4.00	6.00
BOSPHI	1.00	2.00	1.00	1.00	0.00	1.00	2.00	1.00	2.00	1.00	2.00
BOSSEA	7.00	5.00	8.00	7.00	7.00	7.00	7.00	7.00	8.00	8.00	5.00
BOSSTL	2.00	3.00	3.00	2.00	3.00	3.00	1.00	3.00	2.00	3.00	2.00
BOSTBD	12.00	11.00	10.00	11.00	10.00	12.00	12.00	10.00	14.00	12.00	12.00
BOSTEX	6.00	6.00	4.00	7.00	4.00	10.00	6.00	2.00	5.00	8.00	5.00
BOSTOR	10.00	11.00	8.00	11.00	7.00	9.00	9.00	14.00	6.00	12.00	9.00
sum	98.00	94.00	91.00	108.00	95.00	95.00	101.00	94.00	102.00	107.00	100.00
winnpct	0.60	0.58	0.56	0.67	0.59	0.59	0.62	0.58	0.63	0.66	0.62

Table 5: Sample results (Season 90-100) from simulation of 100 seasons

References

[1] Jim Albert, Jay Bennett, *Curve Ball: Baseball, Statistics, and the Role of Chance in the Game*. Springer Verlag New York Inc., New York, 2003 (2001).