The value of a basketball player

J.A.Hartigan Yale University
September 2013

1 Money

In 2012-2013, the top salary in the National Baseball Association was 30 million dollars for Kobe Bryant, Los Angeles Lakers; the top ten player each receive more than 18 million dollars. The starting salary for a rookie player was 0.5 million dollars.

The EWA( Estimated Wins Added) is a statistic developed by John Hollinger that puts the price of a win at 1.6 million dollars, and estimates for each player how many wins he adds to a team. His salary is regarded as good for the team if the number of wins added multiplied by 1.6 million exceeds his salary. See

http://www.sonicscentral.com/statsite.html

as a source for many different theoretical approaches to basketball. Many of them are based on "play by play" statistics, the basketball events that occur during each "matchup" between a particular 5 players on one team with a particular 5 players on the other team. The play by play data is available up to the end of the 2012 season at

http://basketballvalue.com/downloads.php

We develop a value measure predicting each player’s contribution to the point difference between his team’s score and the opponent’s team score, after adjusting for home court advantage and possession advantage. This statistic is routinely produced for evaluating players, for example, at http://www.82games.com/1112/1112NYK1.HTM.
We copy and paste the following partial table for 6 March 2012. The production scores are for a 48 minute game. For example, Lin played 32% of the time in the games up till 6 March 2012. His Net48 is 7.8 more points for his team than for the opponents team per 48 minutes played.

<table>
<thead>
<tr>
<th>Player</th>
<th>Minutes</th>
<th>Net48</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anthony</td>
<td>51%</td>
<td>+0.7</td>
</tr>
<tr>
<td>Novak</td>
<td>22%</td>
<td>+14.5</td>
</tr>
<tr>
<td>Jeffries</td>
<td>32%</td>
<td>+1.1</td>
</tr>
<tr>
<td>Lin</td>
<td>31%</td>
<td>+7.8</td>
</tr>
<tr>
<td>Shumpert</td>
<td>50%</td>
<td>+0.5</td>
</tr>
<tr>
<td>Fields</td>
<td>64%</td>
<td>+3.7</td>
</tr>
</tbody>
</table>
2 Estimating player value

We assume that each player on the floor makes an additive contribution to the difference in team scores in each minute of play. There are five players from each team, so the away team – away team score is estimated by the sum of 5 players values from the away team less the sum of 5 players values from the home team. In addition, a term is added to the minute score difference for the away team in possession of the ball, and for the home court advantage of the home team.

There are many substitutions, so to determine these player values, we need to know the player on the floor at any time. The data will come with one row of data for each interval of time between substitutions in which the same sets of 5 players are on the floor from both teams. The estimation problem is then a simple (but voluminous) regression calculation in which the score difference is predicted by a sum of terms of player values, possession values and home court values; there are 525 players, but only 10 of them appear in any given time segment.

The matchup data is assembled in section 7, and saved in the file “data/201213matchups.csv”

Start Data analysis from here.

m <- read.csv(file="data/201213matchups.csv", as.is=T)
head(m, 3)

<table>
<thead>
<tr>
<th>gameId</th>
<th>times</th>
<th>ahscore</th>
<th>aposs</th>
<th>a1</th>
<th>a2</th>
<th>a3</th>
<th>a4</th>
<th>a5</th>
<th>hposs</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>372</td>
<td>2</td>
<td></td>
<td>.off Emeka Okafor Trevor Ariza</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>137</td>
<td>-4</td>
<td></td>
<td>.def Emeka Okafor Trevor Ariza</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>60</td>
<td>-3</td>
<td></td>
<td>.off Jan Vesely Trevor Ariza</td>
<td>Bradley Beal</td>
<td>Trevor Booker</td>
<td>A.J. Price</td>
<td>.def</td>
<td></td>
</tr>
</tbody>
</table>

h1           h2           h3           h4           h5           ateam         hteam
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Anderson Varejao</td>
<td>Alonzo Gee</td>
<td>Kyrie Irving</td>
<td>Dion Waiters</td>
<td>C.J. Miles</td>
<td>Alonzo Gee</td>
<td>Kyrie Irving</td>
</tr>
<tr>
<td>2</td>
<td>C.J. Miles</td>
<td>Daniel Gibson</td>
<td>Kyrie Irving</td>
<td>Tyler Zeller</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>C.J. Miles</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Tristan Thompson</td>
<td>Washington Wizards</td>
<td>Cleveland Cavaliers</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>-----------------</td>
<td>--------------------</td>
<td>--------------------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

\[ \text{dim}(m) \]

[1] 20235 17
3 Regression to estimate player value

Define iterative regression, to be repeated in various ways. The algorithm does the regression one team at a time. The best fit of the first team to is carried out with least squares on the per second score in each time interval, weighted by the total number of seconds in the time interval. The variables used are the 5 team players in that time interval, the beginning possession, and the home court advantage. The residuals from the fit are used in the next least squares fit for the second team. The iterative calculation is repeated 10 times for the 30 teams.

\[ \text{iR} <- \text{iterateRegress(m)} \]

```
\begin{verbatim}
\text{it: 1 sqres: 5963035.646}
\text{it: 2 sqres: 5944757.88}
\text{it: 3 sqres: 5943853.923}
\text{it: 4 sqres: 5943713.473}
\text{it: 5 sqres: 5943682.52}
\text{it: 6 sqres: 5943675.036}
\text{it: 7 sqres: 5943673.096}
\text{it: 8 sqres: 5943672.49}
\text{it: 9 sqres: 5943672.21}
\text{it: 10 sqres: 5943672.01}
\text{it: 11 sqres: 5943671.829}
\text{it: 12 sqres: 5943671.653}
\text{it: 13 sqres: 5943671.478}
\text{it: 14 sqres: 5943671.303}
\text{it: 15 sqres: 5943671.128}
\text{it: 16 sqres: 5943670.954}
\text{it: 17 sqres: 5943670.779}
\text{it: 18 sqres: 5943670.605}
\text{it: 19 sqres: 5943670.43}
\text{it: 20 sqres: 5943670.256}
\end{verbatim}
```

\[ \text{s1} <- \text{ShowTeam(iR$est,16)} \]

We will compare this later with a smoothed estimate.
4 Prior adjustment for infrequent players

Do analysis by teams with a prior for each player effect corresponding to zero score difference in 5-g games, whenever the player played in g < 5 games. This prior compresses the scores of those infrequent players towards zero.

```r
mfake <- m
est <- initializeEst(m)

We need to know the estimated times to do the adjustment:

```r
est$time <- iR$est$time
for (t in 1:length(est$team))
    est$est[[t]] <- (rep(0, length(est$teams[[t]])))

# add fake data for each player, 5-g games with 0 scores
i <- dim(m)[1]
for (t in 1:length(est$team)){
    for ( p in 2:length(est$teams[[t]])) {
        if( est$time[[t]][p] < 5){
            i <- i+1
            mfake[i, ] <- mfake[i-1, ]
            mfake$ateam[i] <- est$team[t]
            mfake$hteam[i] <- ""
        # a game has 60*48 = 2880 seconds
        mfake$times[i] <- (5-est$time[[t]][p]) * 2880
        mfake$ahscore[i] <- 0
        mfake[i, 4:15] <- ""
        mfake[i, 9] <- est$teams[[t]][p]
        }
    }
}
iR <- iterateRegress(mfake)

it: 1 sqres: 5997591.415
it: 2 sqres: 5982294.941
it: 3 sqres: 5981700.565
it: 4 sqres: 5981593.254
it: 5 sqres: 5981540.117
it: 6 sqres: 5981496.34
it: 7 sqres: 5981454.762
it: 8 sqres: 5981413.998
it: 9 sqres: 5981373.766
it: 10 sqres: 5981334.004
it: 11 sqres: 5981294.695
it: 12 sqres: 5981255.833
it: 13 sqres: 5981217.41
it: 14 sqres: 5981179.424
it: 15 sqres: 5981141.868
it: 16 sqres: 5981104.738
it: 17 sqres: 5981068.028
it: 18 sqres: 5981031.735
it: 19 sqres: 5980995.854
it: 20 sqres: 5980960.379

s2 <- ShowTeam(iR$est,16)
names(s2) <- c("pdiff48", "pse", "pgames")
print(iR$est$team[16])

[1] "Miami Heat"

The difference between estimates with and without priors:

print(cbind(s1,s2))

diff48  se  games  pdiff48  pse  pgames
..home court  -1  5   26   -1  5   26
..off  3  5   26    2  5   26
Chris Andersen  6 10    7    8 10    7
Chris Bosh  1  8   31    2  7   31
Dwyane Wade  4  6   29    3  6   29
James Jones  9 15    2    3  8    5
Jarvis Varnado  -8 28    1    1  8    5
Joel Anthony  -1 10    6    1 10    6
Josh Harrellson  -78 48    0    0  8    5
Juwan Howard  18 27    0    4  8    5
LeBron James  14  7   37    12  7   37
Mario Chalmers  19  9   27    16  9   27
Mike Miller  2  7   11    3  6   11
Norris Cole  7  9   18    4  8   18
Patrick Patterson  -37  35  0   1  8  5  
Rashard Lewis       -1  7   9   -1  7   9  
Ray Allen           1  6  23   0  6  23  
Shane Battier       -3  6  21   -1  6  21  
Terrel Harris       59  44   0   3  8  5  
Udonis Haslem       -8  7  19   -6  7  19  

Note that the estimates for frequent players are affected only a little, and the wildly varying estimates for the infrequent players, (less than 5 games), are set close to zero. It tends to reduce the positive effects of the star players like LeBron James, but we don't know these effects very accurately based just on one season.
5 Player estimates displayed

tiff("pictures/diff48.dif", w=900, h=900)

Grid(c(seq(-25, 10, 5), 12), c(0, 30), at=c(0, -22),
     ylab=" diff48 for all players/Teams", cex=2.5)

se <- mean(unlist(iR$est$se))
rect(-2*se, 31, 2*se, 32, col="light grey", xpd=T)
lines(c(0,0), c(0,32), lwd=2)

text(10, 31.5, "+2se", cex=1.7, xpd=T)
text(-10, 31.5, "-2se", cex=1.7, xpd=T)

# plot points for each team
# allow text outside plot region
par(xpd=T)
for(i in 1:30){
  points(iR$est$est[[i]][-c(1,2)], rep(i,length(iR$est$est[[i]])-2), pch="|")
  points(iR$est$est[[i]][1], i, pch="H")
  points(iR$est$est[[i]][2], i, pch="O")
  text( -25, i, iR$est$team[i], cex=1.8, pos=4)
}

dev.off()
mean(unlist(iR$est$se))

[1] 7.339993953

There are relatively few values greater than 0, many less than 0. If you were testing for statistical significance, none of the players would be judged to have a statistically significant effect on the score. The possession effects (o), and home court effects(H) are nearly all positive.
6 Detailed estimates

We show only estimates that exceed $\sqrt{2}$ the standard error, which corresponds to the Akaike criterion for model fitting:

```r
teams <- iR$est$team
for(t in 1:30){
cat(" \
")
cat(teams[t], "\n")
}
```

```r
print(ShowTeam(iR$est,t, threshold = sqrt(2)))
```

Atlanta Hawks

<table>
<thead>
<tr>
<th>diff48</th>
<th>se</th>
<th>games</th>
</tr>
</thead>
<tbody>
<tr>
<td>.off</td>
<td>8</td>
<td>5</td>
</tr>
</tbody>
</table>

Kyle Korver 9 6 30

Boston Celtics

<table>
<thead>
<tr>
<th>diff48</th>
<th>se</th>
<th>games</th>
</tr>
</thead>
<tbody>
<tr>
<td>.home court</td>
<td>9</td>
<td>5</td>
</tr>
<tr>
<td>.off</td>
<td>10</td>
<td>5</td>
</tr>
<tr>
<td>Jeff Green</td>
<td>9</td>
<td>6</td>
</tr>
<tr>
<td>Paul Pierce</td>
<td>18</td>
<td>6</td>
</tr>
</tbody>
</table>

Brooklyn Nets

<table>
<thead>
<tr>
<th>diff48</th>
<th>se</th>
<th>games</th>
</tr>
</thead>
<tbody>
<tr>
<td>Andray Blatche</td>
<td>13</td>
<td>8</td>
</tr>
<tr>
<td>Gerald Wallace</td>
<td>11</td>
<td>6</td>
</tr>
</tbody>
</table>

Charlotte Bobcats

<table>
<thead>
<tr>
<th>diff48</th>
<th>se</th>
<th>games</th>
</tr>
</thead>
<tbody>
<tr>
<td>.home court</td>
<td>9</td>
<td>5</td>
</tr>
<tr>
<td>.off</td>
<td>10</td>
<td>5</td>
</tr>
<tr>
<td>Ben Gordon</td>
<td>-8</td>
<td>5</td>
</tr>
<tr>
<td>Byron Mullens</td>
<td>-16</td>
<td>6</td>
</tr>
<tr>
<td>Hakim Warrick</td>
<td>-18</td>
<td>8</td>
</tr>
</tbody>
</table>

Chicago Bulls

<table>
<thead>
<tr>
<th>diff48</th>
<th>se</th>
<th>games</th>
</tr>
</thead>
<tbody>
<tr>
<td>Team</td>
<td>Player</td>
<td>Off</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-----------------</td>
<td>-----</td>
</tr>
<tr>
<td>Cleveland Cavaliers</td>
<td>Taj Gibson</td>
<td>9</td>
</tr>
<tr>
<td></td>
<td>Kyrie Irving</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>Shaun Livingston</td>
<td>10</td>
</tr>
<tr>
<td>Dallas Mavericks</td>
<td>Dirk Nowitzki</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td>Vince Carter</td>
<td>16</td>
</tr>
<tr>
<td>Denver Nuggets</td>
<td>Andre Iguodala</td>
<td>14</td>
</tr>
<tr>
<td></td>
<td>Andre Miller</td>
<td>9</td>
</tr>
<tr>
<td></td>
<td>JaVale McGee</td>
<td>-14</td>
</tr>
<tr>
<td></td>
<td>Timofey Mozgov</td>
<td>-16</td>
</tr>
<tr>
<td></td>
<td>Ty Lawson</td>
<td>9</td>
</tr>
<tr>
<td>Detroit Pistons</td>
<td>Austin Daye</td>
<td>13</td>
</tr>
<tr>
<td></td>
<td>Khris Middleton</td>
<td>15</td>
</tr>
<tr>
<td>Golden State Warriors</td>
<td>Andrew Bogut</td>
<td>-14</td>
</tr>
<tr>
<td></td>
<td>Carl Landry</td>
<td>-13</td>
</tr>
<tr>
<td></td>
<td>Festus Ezeli</td>
<td>-10</td>
</tr>
<tr>
<td></td>
<td>Klay Thompson</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>Richard Jefferson</td>
<td>16</td>
</tr>
<tr>
<td></td>
<td>Stephen Curry</td>
<td>11</td>
</tr>
<tr>
<td>Houston Rockets</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Team</td>
<td>diff48 se games</td>
<td></td>
</tr>
<tr>
<td>----------------------</td>
<td>------------------------------------------------------</td>
<td>---</td>
</tr>
<tr>
<td></td>
<td>.home court</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>.off</td>
<td>9</td>
</tr>
<tr>
<td></td>
<td>Omer Asik</td>
<td>16</td>
</tr>
<tr>
<td>Indiana Pacers</td>
<td>diff48 se games</td>
<td></td>
</tr>
<tr>
<td></td>
<td>.off</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>David West</td>
<td>28</td>
</tr>
<tr>
<td></td>
<td>Ian Mahinmi</td>
<td>-21</td>
</tr>
<tr>
<td></td>
<td>Tyler Hansbrough</td>
<td>19</td>
</tr>
<tr>
<td>Los Angeles Clippers</td>
<td>diff48 se games</td>
<td></td>
</tr>
<tr>
<td></td>
<td>.off</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>Chris Paul</td>
<td>14</td>
</tr>
<tr>
<td></td>
<td>Ronny Turiaf</td>
<td>-18</td>
</tr>
<tr>
<td>Los Angeles Lakers</td>
<td>diff48 se games</td>
<td></td>
</tr>
<tr>
<td></td>
<td>.home court</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>.off</td>
<td>16</td>
</tr>
<tr>
<td>Memphis Grizzlies</td>
<td>diff48 se games</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Chris Johnson</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>Darrell Arthur</td>
<td>-21</td>
</tr>
<tr>
<td></td>
<td>Jerryd Bayless</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td>Marreese Speights</td>
<td>-22</td>
</tr>
<tr>
<td></td>
<td>Mike Conley</td>
<td>16</td>
</tr>
<tr>
<td></td>
<td>Quincy Pondexter</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>Tayshaun Prince</td>
<td>13</td>
</tr>
<tr>
<td></td>
<td>Wayne Ellington</td>
<td>17</td>
</tr>
<tr>
<td></td>
<td>Zach Randolph</td>
<td>-11</td>
</tr>
<tr>
<td>Miami Heat</td>
<td>diff48 se games</td>
<td></td>
</tr>
<tr>
<td></td>
<td>LeBron James</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>Mario Chalmers</td>
<td>16</td>
</tr>
<tr>
<td>Milwaukee Bucks</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Team</td>
<td>diff48 se games</td>
<td>.off</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-----------------</td>
<td>------</td>
</tr>
<tr>
<td>J.J. Redick</td>
<td></td>
<td>12</td>
</tr>
</tbody>
</table>

Minnesota Timberwolves

<table>
<thead>
<tr>
<th>diff48 se games</th>
<th>.off</th>
<th>13</th>
<th>5</th>
<th>24</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;0 rows&gt; (or 0-length row.names)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

New Orleans Hornets

<table>
<thead>
<tr>
<th>diff48 se games</th>
<th>.off</th>
<th>12</th>
<th>5</th>
<th>22</th>
</tr>
</thead>
<tbody>
<tr>
<td>Greivis Vasquez</td>
<td>14</td>
<td>9</td>
<td>34</td>
<td></td>
</tr>
<tr>
<td>Xavier Henry</td>
<td>-13</td>
<td>7</td>
<td>7</td>
<td></td>
</tr>
</tbody>
</table>

New York Knicks

<table>
<thead>
<tr>
<th>diff48 se games</th>
<th>.off</th>
<th>11</th>
<th>5</th>
<th>24</th>
</tr>
</thead>
<tbody>
<tr>
<td>James White</td>
<td>-13</td>
<td>9</td>
<td>6</td>
<td></td>
</tr>
</tbody>
</table>

Oklahoma City Thunder

<table>
<thead>
<tr>
<th>diff48 se games</th>
<th>Hasheem Thabeet</th>
<th>12</th>
<th>8</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kendrick Perkins</td>
<td>12</td>
<td>8</td>
<td>26</td>
<td></td>
</tr>
<tr>
<td>Kevin Durant</td>
<td>12</td>
<td>7</td>
<td>39</td>
<td></td>
</tr>
<tr>
<td>Nick Collison</td>
<td>18</td>
<td>8</td>
<td>20</td>
<td></td>
</tr>
</tbody>
</table>

Orlando Magic

<table>
<thead>
<tr>
<th>diff48 se games</th>
<th>.off</th>
<th>16</th>
<th>5</th>
<th>24</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jameer Nelson</td>
<td>10</td>
<td>6</td>
<td>26</td>
<td></td>
</tr>
<tr>
<td>Moe Harkless</td>
<td>-14</td>
<td>6</td>
<td>21</td>
<td></td>
</tr>
</tbody>
</table>

Philadelphia 76ers

<table>
<thead>
<tr>
<th>diff48 se games</th>
<th>Royal Ivey</th>
<th>-13</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spencer Hawes</td>
<td>11</td>
<td>6</td>
<td>28</td>
<td></td>
</tr>
<tr>
<td>Thaddeus Young</td>
<td>11</td>
<td>6</td>
<td>35</td>
<td></td>
</tr>
</tbody>
</table>

Phoenix Suns

<table>
<thead>
<tr>
<th>diff48 se games</th>
<th>.off</th>
<th>9</th>
<th>5</th>
<th>22</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Points</td>
<td>Assists</td>
<td>Rebounds</td>
<td></td>
</tr>
<tr>
<td>---------------------</td>
<td>--------</td>
<td>---------</td>
<td>----------</td>
<td></td>
</tr>
<tr>
<td>Goran Dragic</td>
<td>20</td>
<td>9</td>
<td>31</td>
<td></td>
</tr>
<tr>
<td>Marcin Gortat</td>
<td>-11</td>
<td>7</td>
<td>24</td>
<td></td>
</tr>
<tr>
<td>Markieff Morris</td>
<td>-11</td>
<td>7</td>
<td>21</td>
<td></td>
</tr>
</tbody>
</table>

Portland Trail Blazers

<table>
<thead>
<tr>
<th>Player</th>
<th>Points</th>
<th>Assists</th>
<th>Rebounds</th>
</tr>
</thead>
<tbody>
<tr>
<td>Damian Lillard</td>
<td>25</td>
<td>8</td>
<td>41</td>
</tr>
<tr>
<td>J.J. Hickson</td>
<td>-25</td>
<td>8</td>
<td>31</td>
</tr>
<tr>
<td>Joel Freeland</td>
<td>-16</td>
<td>10</td>
<td>5</td>
</tr>
<tr>
<td>Luke Babbitt</td>
<td>-14</td>
<td>8</td>
<td>7</td>
</tr>
<tr>
<td>Meyers Leonard</td>
<td>-18</td>
<td>8</td>
<td>14</td>
</tr>
<tr>
<td>Ronnie Price</td>
<td>19</td>
<td>10</td>
<td>6</td>
</tr>
</tbody>
</table>

Sacramento Kings

<table>
<thead>
<tr>
<th>Player</th>
<th>Points</th>
<th>Assists</th>
<th>Rebounds</th>
</tr>
</thead>
<tbody>
<tr>
<td>James Johnson</td>
<td>-10</td>
<td>6</td>
<td>1</td>
</tr>
<tr>
<td>John Salmons</td>
<td>9</td>
<td>6</td>
<td>31</td>
</tr>
<tr>
<td>Thomas Robinson</td>
<td>-13</td>
<td>8</td>
<td>10</td>
</tr>
</tbody>
</table>

San Antonio Spurs

<table>
<thead>
<tr>
<th>Player</th>
<th>Points</th>
<th>Assists</th>
<th>Rebounds</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tiago Splitter</td>
<td>12</td>
<td>7</td>
<td>25</td>
</tr>
<tr>
<td>Tim Duncan</td>
<td>12</td>
<td>7</td>
<td>26</td>
</tr>
</tbody>
</table>

Toronto Raptors

<table>
<thead>
<tr>
<th>Player</th>
<th>Points</th>
<th>Assists</th>
<th>Rebounds</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amir Johnson</td>
<td>12</td>
<td>7</td>
<td>28</td>
</tr>
<tr>
<td>Quincy Acy</td>
<td>12</td>
<td>8</td>
<td>5</td>
</tr>
</tbody>
</table>

Utah Jazz

<table>
<thead>
<tr>
<th>Player</th>
<th>Points</th>
<th>Assists</th>
<th>Rebounds</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alec Burks</td>
<td>10</td>
<td>7</td>
<td>13</td>
</tr>
<tr>
<td>Gordon Hayward</td>
<td>10</td>
<td>6</td>
<td>27</td>
</tr>
</tbody>
</table>

Washington Wizards

<table>
<thead>
<tr>
<th>Player</th>
<th>Points</th>
<th>Assists</th>
<th>Rebounds</th>
</tr>
</thead>
</table>
I found these estimates surprising. LeBron James of the Miami Dolphins widely regarded as the best player in the league, but these results suggest many other players are better, including Mario Chalmers on his own team. Still, none of the diffs are very accurately known.

There are substantial effects in most teams for home court advantage and offense. For example, The Washington Wizards play 14 points better per game at home than away, and play offense better than defense by 19 points per game.
7 Assembling matchup data

7.1 Find the team names for 2012-2013 season:

The team names are located at:

http://espn.go.com/nba/teams

```r
teams <- scan("http://espn.go.com/nba/teams", what="", sep="\n")

Pick the team names out of the source file to match an expression beginning with team/-/name and ending in characters -,/, letters, or 7 or 6:

teamnames <- regmatches(teams, regexpr("team/_/name/[/-a-z76]*", teams))
teamnames <- substring(teamnames, 13, nchar(teamnames))

The last name is a repeat:

teamnames <- teamnames[-31]
```
7.2 Find the recap ID's for all games

Find the URLs for the different teams' schedules:

```r
snames <- teamnames

Capitalize the first letter:

uctn <- toupper(substr(teamnames,1,1))
head(uctn)

[1] "B" "B" "N" "P" "T" "G"

substr(snames,1,1) <- uctn
head(snames)

[1] "Bos/boston-celtics" "Bkn/brooklyn-nets"
[3] "Ny/new-york-knicks" "Phi/philadelphia-76ers"
[5] "Tor/toronto-raptors" "Gs/golden-state-warriors"

Add seasonType/2/ to the names:

ssnames <- sub("/", "/seasonType/2/", snames)
head(ssnames)

[1] "Bos/seasonType/2/boston-celtics"
[2] "Bkn/seasonType/2/brooklyn-nets"
[3] "Ny/seasonType/2/new-york-knicks"
[4] "Phi/seasonType/2/philadelphia-76ers"
[5] "Tor/seasonType/2/toronto-raptors"
[6] "Gs/seasonType/2/golden-state-warriors"

Paste the espn header to the expanded team name:

```r
gssnames <- paste("http://espn.go.com/nba/team/schedule/_/name/",
ssnames, sep="")
head(gssnames, 2)

[1] "http://espn.go.com/nba/team/schedule/_/name/Bos/seasonType"
Extract gameIds for all games:

```r
gameId <- NULL
for( team in 1:length(sssnames)){
sched <- scan(sssnames[team], what="", sep="\n", quiet=T)

# Locate the single line in the source page containing the gameId:
recap <- grep("recap", sched)

# Find the positions of all those gameId's:
pos <- unlist(gregexpr("[0-9]{9}", sched[recap]))
if( team==1) head(pos)

# Loop through the positions to get the ID's:
u <- rep("", length(pos))
for(i in 1:length(pos)) u[i] <- substr(sched[recap], pos[i], pos[i]+8)
if(team==1) head(u)
gameId <- c(gameId, u)
}
gameId <- sort(unique(gameId))
length(gameId)
```

[1] 0

We sort the unique gameId (each gameId appears twice because each scheduled game appears in two teams list of games). The gameIds go from 400277721 to 400278950 in order, corresponding to order of dates of the games, with two additional mysterious gameIds at 400431053 and 400440940, which indeed are regular season games.
You may view the play by plays for the first game at:
http://espn.go.com/nba/playbyplay?gameId=400277721
7.3 Collecting Play by Play records

A single paragraph is extracted from the source file of the play by play for each game. The collected lines are saved for further processing. It takes about 30 minutes, so you don't want to do this more than once.

```r
if (FALSE) {

root <- "http://espn.go.com/nba/playbyplay?gameId="

for (id in gameId) {
  x <- scan(paste(root, id, "&period=0", sep=""),
          what="", sep="\n", quiet=T)

  # possible play by play not available
  if (length(grep("Play-By-Play not available", x)) == 0) {
    x <- scan(paste(root, id, "&period=0", sep=""),
              what="", sep="\n", quiet=T)
    start <- grep("Complete Play-By-Play", x)
    end <- grep("End of Game | End of the 4th Quarter | End of Fourth Quarter", x)

    if (length(end) > 0) {
      end <- max(end)
      use <- start:end
      pbp <- c(pbp, x[use])
    }
  }
}

teams <- Matches("[.A-Za-z67 ]+ - Play By Play - ",
                  x[5])
teams <- gsub(" - Play By Play - ", ",", teams)
teams <- paste(id, teams, sep="/"")
teams <- gsub(" vs. ", "/", teams)
gameteams <- c(gameteams, teams)
}
gameteams <- unlist(strsplit(gameteams, "/"))
gameteams <- data.frame(t(matrix(gameteams, nrow=3)))
```
names(gameteams) <- c("gameId", "ateam", "hteam")
cat(pbp, file="data/201213play-by-play", sep="\n")
head(pbp, 3)

write.csv(gameteams,
         file="data/201213gameteams.csv",row.names=F)
head(gameteams)
}
### 7.4 Express play by plays as data frame

```r
pbp <-
scan("data/201213play-by-play",what="q",sep="\n")

length(pbp)

[1] 531797

head(pbp, 3)


```

```html

<table border="1" width="100%"
class="mod-data">
<thead><tr bgcolor="#555555">
<th width="5%" style="text-align:center; padding:0;">TIME</th>
<th width="40%" style="text-align:left;">WASHINGTON</th>
<th width="5%" style="text-align:center;">SCORE</th>
<th width="40%" style="text-align:left;">CLEVELAND</th>
</tr></thead>
<tbody>
<tr class="even"><td valign=top width=50 style="text-align:center;">12:00</td>
<td valign=top>&nbsp;</td>
<td valign=top style="text-align:top">Emeka Okafor vs. Tristan Thompson (Trevor Booker gains possession)</td>
</tr>
<tr class="odd"><td valign=top width=50 style="text-align:center;">11:42</td>
<td valign=top>&nbsp;</td>
<td valign=top style="text-align:top">Trevor Ariza makes three point jumper (A.J. Price assists)</td>
</tr>
</tbody>
</table>
```

```r
```

```html

8:59 0-0</td>
</tr>
```
Find GameId:
```r
gameId <- Matches("[0-9]{9}\", pbp)
length(gameId)
```
```
[1] 1224
```

Find scores:
Eliminate entries without scores:
```r
length(scores)
```
```
[1] 511040
```

```r
length(pbp)
```
```
[1] 511040
```

Find times:
```r
times <- Matches("[0-9]+:[0-9]{2}\", pbp)
length(times)
```
```
[1] 511040
```

Make times numerical in seconds:
```r
t <- as.numeric(unlist(strsplit(times, ":")))
times <- 60 * t[seq(1, length(t), 2)] + t[seq(2, length(t), 2)]
```
Find Possessions:
Specify possess and gameId (NOWRAP) appears after action for away actions; the expression describes an action with at least two words:

```r
eexpr <- "\>[ ]?[ ]?[\-\.:,"\(\(\)a-zA-Z0-9]+\-[\-\.:,"\(\(\)a-zA-Z0-9]+<"
possess <- rep(NA, length(pbp))
away <- grepl(paste(expr,".*NOWRAP", sep=""), pbp)
home <- !away & grepl(expr, pbp)
possess[away] <- "AWAY"
possess[home] <- "HOME"
possess[grepl("Complete Play-By-Play", pbp)] <- gameId
head(possess)
```

```
[1] "400277721" "AWAY" "HOME" "AWAY"
[5] "AWAY" "HOME"
```

Put gameId into every entry:
Find where the gameId's are located:

```r
startgame <- grep("Complete Play-By-Play", pbp)
length(startgame)
```

```
[1] 1224
```

```r
startgame <- c(startgame, length(times)+1)
Game <- rep("", length(times))
```

Loop through each startgame and copy gameId into full game:

```r
for(i in 1:(length(startgame)-1))
Game[startgame[i]: (startgame[i+1]-1)] <- possess[startgame[i]]
possess[startgame[i]] <- "NA"
head(Game,3)
```

```
[1] "400277721" "400277721" "400277721"
```
tail(Game, 3)

[1] "400440940" "400440940" "400440940"

Find actions after fixing awkward phrases:

```r
actions <- rep("", length(pbp))
pbp <- gsub("Complete Play|1st|2nd|3rd|4th|Quarter|Summary", "", pbp)
pbp <- gsub("Double technical|off foul|Jr. |Al-", "", pbp)
pbp <- gsub("shot clock|game violation|delay of", "", pbp)
pbp <- gsub("Ed Davis makes free throw 2 of 2. ", "", pbp)
```

Get rid of team names:

```r
pbp <- gsub ("[A-Z]{3,}", "", pbp)
```

Handle a circumflex and player with more than two names:

```r
pbp <- gsub("Jose Juan Barea", "Jose Barea", pbp)
pbp <- gsub("Nando de Colo", "Nando Colo", pbp)
pbp <- gsub("Nenê", "Nene Nene", pbp)
pbp <- gsub("Metta World Peace", "Metta World", pbp)
pbp <- gsub("Luc Richard Mbah a Moute", "Luc Moute", pbp)
```

Lots of funny things can happen in the actions, discovered by seeing when a proposed regular expression for the action fails:

```r
wactions <- grepl(expr, pbp)
```

These bad actions will be deleted from the list later:

```r
actions[!wactions] <- ""
actions[wactions] <- Matches(expr, pbp[wactions])
actions <- gsub("<|>", "", actions)
length(actions)
```
head(\texttt{actions})

\begin{verbatim}
[1] "Emeka Okafor vs. Tristan Thompson (Trevor Booker gains possession)"
[3] "Emeka Okafor blocks Alonzo Gee's layup"
[4] "Trevor Ariza defensive rebound"
[5] "Emeka Okafor makes layup (Trevor Ariza assists )"
[6] "Kyrie Irving makes three point jumper (Tristan Thompson assists )"
\end{verbatim}

Make data frame with the 5 variables:

\begin{verbatim}
pbpExtract <- data.frame(gameId = Game, times=times, scores=scores, possess=possess, actions=actions, stringsAsFactors=FALSE)
head(pbpExtract)
\end{verbatim}

\begin{verbatim}
    gameId times scores possess
1  400277721   720    0-0 400277721
2  400277721   702    3-0    AWAY
3  400277721   678    3-0     HOME
4  400277721   675    3-0    AWAY
5  400277721   668    5-0    AWAY
6  400277721   648    5-3     HOME
\end{verbatim}

\begin{verbatim}
actions
1 Emeka Okafor vs. Tristan Thompson (Trevor Booker gains possession)
2 Trevor Ariza makes three point jumper (A.J. Price assists )
3 Emeka Okafor blocks Alonzo Gee's layup
4 Trevor Ariza defensive rebound
5 Emeka Okafor makes layup (Trevor Ariza assists )
\end{verbatim}
6 Kyrie Irving makes three point jumper (Tristan Thompson assists)

dim(pbpExtract)

[1] 511040

pbpExtract <- pbpExtract[actions != "",]
pbpExtract <- na.omit(pbpExtract)

Modify actions to consist of player names involved in action:
First handle team actions and blocks and various non player actions:

```r
pbpExtract <- pbpExtract[!grepl("team|vs.", pbpExtract$actions),]
actions <- pbpExtract$actions
actions <- gsub("[-'.a-zA-Z]+ [-'.a-zA-Z]+ blocks ", ",
"", actions)
actions[grep("technical foul", actions)] <- ""
actions <- gsub("'s", ", actions)
dim(pbpExtract)
```

[1] 486542

length(actions)

[1] 486542

```r
actions[grep("Sec Inbound", actions)] <- ""
# change some particular enter errors
actions[pbpExtract$times== 363 & pbpExtract$gameId=="400277723"] <- "Metta World enters the game for Antawn Jamison"
length(actions)
```

[1] 486542

Handle non-enter actions:

```r
enters <- grepl(" enters the game for ", actions)
pexpr <- "[A-Z][-'.a-zA-Z]+ [A-Z][-'.a-zA-Z]+"
```
bad <- !grepl(pexpr, actions)
actions[bad] <- ""
actions[!(enters|bad)] <-
    Matches(pexpr, actions[!(enters|bad)])
length(actions)

[1] 486542

Treat enter the game actions differently:
actions[enters] <-
gsub("[ ]+enters the game for[ ]+", "/", actions[enters])
badenter <- !grepl(eexpr, actions)
actions[enters & badenter] <- ""
be <- enters &!badenter
actions[be] <- Matches(eexpr, actions[be])
pbpExtract$actions <- actions
length(actions)

[1] 486542

dim(pbpExtract)

[1] 486542  5

Remove missing and "" values:
pbpExtract <- na.omit(pbpExtract)
pbpExtract <- pbpExtract[actions !="", ]
pbpExtract[bpbExtract$gameId=="400277723", ][140:145, ]

<table>
<thead>
<tr>
<th>gameId</th>
<th>times</th>
<th>scores</th>
<th>possess</th>
</tr>
</thead>
<tbody>
<tr>
<td>1005</td>
<td>400277723</td>
<td>384</td>
<td>37-37</td>
</tr>
<tr>
<td>1006</td>
<td>400277723</td>
<td>363</td>
<td>37-37</td>
</tr>
<tr>
<td>1007</td>
<td>400277723</td>
<td>356</td>
<td>39-37</td>
</tr>
<tr>
<td>1008</td>
<td>400277723</td>
<td>326</td>
<td>39-39</td>
</tr>
<tr>
<td>1009</td>
<td>400277723</td>
<td>313</td>
<td>39-39</td>
</tr>
</tbody>
</table>
write.csv(pbpExtract, "data/1213pbpExtract.csv", row.names=F)
7.5 Parse pbp actions to get matchups

Extract <- read.csv("data/1213pbpExtract.csv", as.is=T)
Extract[Extract$gameId=="400277723", ][140:145, ]

<table>
<thead>
<tr>
<th>gameId</th>
<th>times</th>
<th>scores</th>
<th>possess</th>
</tr>
</thead>
<tbody>
<tr>
<td>958</td>
<td>384</td>
<td>37-37</td>
<td>HOME</td>
</tr>
<tr>
<td>959</td>
<td>363</td>
<td>37-37</td>
<td>HOME</td>
</tr>
<tr>
<td>960</td>
<td>356</td>
<td>39-37</td>
<td>AWAY</td>
</tr>
<tr>
<td>961</td>
<td>326</td>
<td>39-39</td>
<td>HOME</td>
</tr>
<tr>
<td>962</td>
<td>313</td>
<td>39-39</td>
<td>AWAY</td>
</tr>
<tr>
<td>963</td>
<td>311</td>
<td>39-39</td>
<td>HOME</td>
</tr>
</tbody>
</table>

actions
958        Pau Gasol
959        Metta World/Antawn Jamison
960        Rodrigue Beaubois
961        Kobe Bryant
962        Jae Crowder
963        Metta World

Find matchups in each game applying the function gamematchup. The starters are found for each quarter, and then a new quintet of players is discovered after each enter action. Many times the enter actions are contradictory, and we do not know the quintet. The times and score information is used to get the change in score and the change in time for each quintet-quintet matchup.

gid <- unique(Extract$gameId)
length(gid)
[1] 1224

system.time({

for( g in gid) {
    gm <- gamematchup(Extract[Extract$gameId==g,])
    if(g == gid[1]) m <- gm
    if(g != gid[1]) m <- rbind(m, gm)
user  system elapsed
151.01    0.34  151.74

print(dim(m))

[1] 34646    15

head(m[,1:6])

    gameId times ahscore aposs      a1
1  400277721   372       2 .off  Emeka Okafor
2  400277721   137     -4 .def  Emeka Okafor
3  400277721    60     -3 .off   Jan Vesely
4  400277721   151     -2 .off   Jan Vesely
5  400277721   147     -2 .off   Jan Vesely
6  400277721    72      1 .def   Jan Vesely

    a2
1      Trevor Ariza
3      Trevor Ariza
5      Trevor Ariza
8  Martell Webster
10     Jannero Pargo
11     Jannero Pargo

Add team names to matchups.

gt <- read.csv("data/201213gameteams.csv", as.is=T)
head(gt)

    gameId    ateam               hteam
1  400277721 Washington Wizards Cleveland Cavaliers
2  400277722      Boston Celtics       Miami Heat
3  400277723  Dallas Mavericks   Los Angeles Lakers
4  400277724    Indiana Pacers       Toronto Raptors
5  400277725 Denver Nuggets     Philadelphia 76ers
6  400277726     Houston Rockets      Detroit Pistons
mrow <- dim(m)[1]
m$ateam <- rep("", mrow)
m$hteam <- rep("", mrow)

Run over all games and insert teams corresponding to the gameId:

gid <- unique(m$gameId)
for (g in gid){
m$ateam[m$gameId == g] <- gt$ateam[gt$gameId==g]
m$hteam[m$gameId == g] <- gt$hteam[gt$gameId==g]
}
head(m)

<table>
<thead>
<tr>
<th>gameId</th>
<th>times</th>
<th>ahscore</th>
<th>aposs</th>
<th>a1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 400277721</td>
<td>372</td>
<td>2</td>
<td>.off</td>
<td>Emeka Okafor</td>
</tr>
<tr>
<td>3 400277721</td>
<td>137</td>
<td>-4</td>
<td>.def</td>
<td>Emeka Okafor</td>
</tr>
<tr>
<td>5 400277721</td>
<td>60</td>
<td>-3</td>
<td>.off</td>
<td>Jan Vesely</td>
</tr>
<tr>
<td>8 400277721</td>
<td>151</td>
<td>-2</td>
<td>.off</td>
<td>Jan Vesely</td>
</tr>
<tr>
<td>10 400277721</td>
<td>147</td>
<td>-2</td>
<td>.off</td>
<td>Jan Vesely</td>
</tr>
<tr>
<td>11 400277721</td>
<td>72</td>
<td>1</td>
<td>.def</td>
<td>Jan Vesely</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>a2</th>
<th>a3</th>
<th>a4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Trevor Ariza</td>
<td>Bradley Beal</td>
<td>Trevor Booker</td>
</tr>
<tr>
<td>Trevor Ariza</td>
<td>Bradley Beal</td>
<td>Trevor Booker</td>
</tr>
<tr>
<td>Trevor Ariza</td>
<td>Bradley Beal</td>
<td>Trevor Booker</td>
</tr>
<tr>
<td>Martell Webster</td>
<td>Jordan Crawford</td>
<td>Trevor Booker</td>
</tr>
<tr>
<td>Jannero Pargo</td>
<td>Martell Webster</td>
<td>Chris Singleton</td>
</tr>
<tr>
<td>Jannero Pargo</td>
<td>Martell Webster</td>
<td>Chris Singleton</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>a5</th>
<th>hposs</th>
<th>h1</th>
<th>h2</th>
</tr>
</thead>
<tbody>
<tr>
<td>A.J. Price</td>
<td>.def</td>
<td>Anderson Varejao</td>
<td>Alonzo Gee</td>
</tr>
<tr>
<td>A.J. Price</td>
<td>.off</td>
<td>C.J. Miles</td>
<td>Alonzo Gee</td>
</tr>
<tr>
<td>A.J. Price</td>
<td>.def</td>
<td>C.J. Miles</td>
<td>Daniel Gibson</td>
</tr>
<tr>
<td>A.J. Price</td>
<td>.def</td>
<td>C.J. Miles</td>
<td>Daniel Gibson</td>
</tr>
<tr>
<td>Jordan Crawford</td>
<td>.def</td>
<td>Tyler Zeller</td>
<td>Daniel Gibson</td>
</tr>
<tr>
<td>Jordan Crawford</td>
<td>.off</td>
<td>Anderson Varejao</td>
<td>Daniel Gibson</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>h3</th>
<th>h4</th>
<th>h5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kyrie Irving</td>
<td>Dion Waiters</td>
<td>Tristan Thompson</td>
</tr>
<tr>
<td>Kyrie Irving</td>
<td>Tyler Zeller</td>
<td>Tristan Thompson</td>
</tr>
<tr>
<td>Kyrie Irving</td>
<td>Tyler Zeller</td>
<td>Tristan Thompson</td>
</tr>
<tr>
<td>Kyrie Irving</td>
<td>Tyler Zeller</td>
<td>Luke Walton</td>
</tr>
<tr>
<td>C.J. Miles</td>
<td>Donald Sloan</td>
<td>Luke Walton</td>
</tr>
</tbody>
</table>
Compute Seconds of time covered by valid and invalid matchups:

```r
sum(m$times[is.na(m[,5])])
```

[1] 1383068

```r
sum(m$times[!is.na(m[,5])])
```

[1] 2142052

So we lose nearly 40% of the matchups due to lack of accurate reporting of substitutions.

```r
mm <- na.omit(m)
dim(mm)
```

[1] 20235 17

```r
write.csv(mm, file="data/201213matchups.csv", row.names=F)
```
11 Functions

Grid <- function(xticks, yticks, ylab="", at=(min(xticks)+ mean(xticks))/2, cex=2.5){
# background for plot using grid of light grey lines

# initialise plot space
par(mar=c(3,3,6,2))
plot(1, 1, xlim=range(xticks),
     ylim = range(yticks),
     xlab="", ylab="", axes=F, pch="")

# use only interior values of tick ranges in plots
usey <- rep( T, length(yticks) )
usey[ c( 1, length(yticks) ) ] <- F
usex <- rep( T, length(xticks) )
usex[ c( 1, length(xticks) ) ] <- F

# grey lines in both directions
for ( row in yticks[ usey ] )
    lines(range(xticks), c(row, row), col="light grey")
for ( col in xticks[ usex ] )
    lines(c(col, col), range(yticks), col="light grey")

# put ylab on left top, using / to split long
expressions
ylabs <- unlist(strsplit(ylab,"/"))

# identify tick marks on both axes
if ( length(yticks) > 2 )
    text(pos=2, rep(min(xticks), length(yticks)-2 ),
         yticks[usey], yticks[usey], cex=2, xpd=T)
if ( length(xticks)>2)
    text(pos=1, xticks[usex], rep(min(yticks),
         length(xticks)-2), xticks[usex], cex=2, xpd=T)

lylabs <- min(5, length(ylabs))
if (lylabs > 0)
    mtext(ylabs, side=3, line = (5/lylabs)*(lylabs-1):0,
          at = at, cex=cex)
except <- function(x,y){
  if(length(y)==0) return(x)
  return (x[ ! (x %in% y)])
}

starters <- function(x){
  # find starters given history of plays and substitutions
  # initialise values
  enters <- grep("/", x)
  le <- length(enters)
  bench <- rep("", 0)
  start <- rep("",0)
  mm <- rep(NA, 5)
  if (length(enters)==0)  return(unique(x))
  # consider changes at substitution
  for(ent in 1:length(enters)){
    # identify two player in substitution
    player <- unlist(strsplit(x[ enters[ent] ], "/" ) )
    # handle substitution on the first play
    if (enters[ent] == 1) {
      start <- player[2]
      bench <- player[1]
    }
  }
}
if (enters[ent] > 1) {
    # handle the first substitution
    if (ent==1) {
        if (player[1] %in% start) return(mm)
        start <- unique(c(player[2], x[1:(enters[ent]-1)]))
        bench <- player[1]
    }
    # handle later substitutions
    if (ent > 1) {
        # add a player to bench unless he already starts
        bench <- unique(c(bench, except(player[1], start)))
        # add a player to start unless already bench
        start <- unique(c(start, except(player[2], bench)))
        # add intervening player to start unless bench
        if (enters[ent-1] < enters[ent]-1) {
            start <- unique(c(start, except(x[(enters[ent-1]+1) : (enters[ent]-1)], bench)))
        }
    }
    if (length(start) > 5) return(mm)
}

# add player to start if any after last enter
if (length(x) > max(enters)) {
    start <- unique(c(start, except(x[(max(enters)+1) : length(x)], bench)))
}

# give up if not 5 starters
if (length(start) != 5) return(mm)
return(start)
matchups <- function(x, allstarters){

# find matchups given history of plays and subs
enters <- grep("/", x)

# initialize array
ze <- length(enters)
options(stringsAsFactors = FALSE)
m <- data.frame( matrix(NA, nrow= ze +2, ncol=10) )
names(m) <-c( paste("a", 1:5, sep=""),
             paste("h", 1:5, sep=""))

# return missing for bad starters
if (sum( is.na(allstarters) ) > 0) return(m)
if(length(allstarters) != 10) return(m)
# now do matchups
m[1, ] <- allstarters
for(ent in 1:length(enters)){
  player <- unlist(strsplit(x[ enters[ent] ], "/" ) )

  # the player going out must be in previous 10
  # the player coming in must not be in previous 10
  if(!(player[2] %in% unlist(m[ent,]))|
    player[1] %in% unlist(m[ent,]) ) return(m)

  m[ent+1, ] <- m[ent, ]
  m[ent+1, which(unlist(m[ent,]) == player[2])] <-
    player[1]
}
m[ent+2,] <- m[ent+1, ]
return(m)
}

matchup <- function(Extract){
  # combine matchups with other game data
  am <-
    starters(Extract$actions[Extract$possess=="AWAY"])
  hm <-
    starters(Extract$actions[Extract$possess=="HOME"])
  m <- matchups(Extract$actions, c(am, hm))

  # get variables at enter times
  enters <- grep("/", Extract$actions)
  le <- length(enters)
mgameId <- rep(Extract$gameId[1], le+2)
la <- length(Extract$actions)
if(le ==0) {
  mtimes <- c(720, 0)
  mscores <- c("0-0", Extract$scores[la])
  mpossess <- c(Extract$possess[1],
                Extract$possess[la])
}

# take some care with initial and final quintets
if(le > 0){
  mtimes <- c(720, Extract$times[enters], 0)
  mscores <- c("0-0",
               Extract$scores[enters], Extract$scores[la])
  penters <- enters+1
  if(penters[le] > la) penters[le] <- la
  mpossess <- c(Extract$possess[1],
                Extract$possess[penters], Extract$possess[la])
}

# define data frame with additional variables
ma <- data.frame(matrix("", le+2, 15))
names(ma) <- c("gameId", "times", "score", "aposs",
              names(m)[1:5], "hposs", names(m)[6:10])

# specify variable values
ma[, 1] <- mgameId
ma[, 2] <- mtimes
ma[, 3] <- mscores
ma[, 4] <- ".off"
ma[mpossess=="HOME", 4] <- ".def"
ma[, 10] <- ".off"
ma[mpossess=="AWAY", 10] <- ".def"
ma[, c(5:9, 11:15)] <- m
return(ma)
gamematchup <- function(Extract){

# do matchups for all quarters
ends <- which (diff(Extract$times) > 0)
ends <- c(0, ends, dim(Extract)[1])

# handle overtime
if(length(ends) > 5) ends <- ends[1:5]

# bind together the 4 quarters
for( i in 1:(length(ends)-1) ){
  mi <- matchup(Extract[(ends[i]+1): ends[i+1],])
  if(i == 1) gm <- mi
  if(i > 1) gm <- rbind(gm, mi)
}

# getting starting scores from previous quarter
st <- which(gm$times==720)
gm$score[st[-1]] <- gm$score[st[-1]-1]

# diff times and scores
gm$times <- c(-diff(gm$times), 0)
scores <- unlist(strsplit(gm$score, "-"))
scores <- as.numeric(scores)
score1 <- scores[seq(1, length(scores), 2)]
score2 <- scores[seq(2, length(scores), 2)]

# away-home score, differenced
gm$score <- c(diff(score1-score2), 0)
names(gm) <- gsub("score", "ahscore", names(gm))
gm <- gm[gm$times > 0,]

return(gm)
initializeEst <- function(m) {

# Initialize estimates, se, times played for each player
est <- list()
team <- sort(unique(m$ateam))
est$team <- team

# tricky first two names are possession and home court
for (t in 1:length(team)) {
    # Define list of players for each team
    # avoid first .def player to avoid singularity in reg
    est$teams[[t]] <- sort(unique(unlist(m[m$ateam==team[t],][,4:9])))
est$teams[[t]] <- est$teams[[t]][est$teams[[t]] != "" &
est$teams[[t]] != ".def"]

    # add in term for home court advantage
    est$teams[[t]] <- c("..home court",est$teams[[t]])
est$est[[t]] <- (rep(0, length(est$teams[[t]])))
est$time[[t]] <- est$est[[t]]
est$se[[t]] <- est$est[[t]]
}

return(est)
}
iterateRegress <- function(m, iter=20){

# iterative regression, one team at a time, to estimate player value
est <- initializeEst(m)
team <- est$team

# update estimates and differences
sqres <- rep(0,iter)
for (it in 1:iter){
  for ( t in 1:length(team) ){

    # find the home and away games for this team
    lp <- length(est$teams[[t]])
    if(lp >0){
      htid <- m$hteam == team[t]
atid <- m$ateam == team[t]
lht <- sum(htid)
lat <- sum(atid)
ht <- m[htid,]
at <- m[atid,]
pdiff <- c(at$ahscore, -ht$ahscore)
ptime <- c(at$times, ht$times)/(48*60)
playmat <- matrix(0, lht + lat, lp)
playmat[(lat+1):(lat + lht), 1] <- 1
est$time[[t]][1] <- sum(ht$times) /(48*60)

    # go through the players to fix prediction matrix
    for (p in 2:lp){

  }
}
}
# set the predictor vector 1 for segments with player
# pl < est$teams[[t]][p]
pl <- est$teams[[t]][p]


ause <- pl==at[,5]|pl==at[,6]|pl==at[,7]|
      pl==at[,8]|pl==at[,9]|pl==at[,4]

playmat[c(ause, huse), p] <- 1

est$time[[t]][p] <- sum(ptime[c(ause, huse)])

# use no constant term in the regression for each player
# contribution
blm <- lm(pdiff/ptime ~ 0 + playmat, weight=ptime)

# update estimate for player and for ahscore res
res <- blm$res * ptime

m$ahscore[atid] <- res[1:lat]

m$ahscore[htid] <- -res[lat + (1:lht)]

est$est[[t]] <- est$est[[t]] + blm$coef
est$se[[t]] <- summary(blm)$coef[,2]

sqres[it] <- sum(m$ahscore^2/(m$times/(48*60)))
cat("it:", it, " sqres:", sqres[it], "\n")

# make estimates time average to zero, to cancel
# indeterminacy

tsum <- 0
ttime <- 0
for ( t in 1:length(est$team)) {
  if(length(est$est[[t]]) > 0){
    tsum <- sum(est$est[[t]][-c(1,2)] *
               est$time[[t]][-c(1,2)])
    ttime <- sum(est$time[[t]][-c(1,2)])
  }
}
tave <- sum(tsum)/sum(ttime)
for (t in 1:length(est$team)) {
    if(length(est$est[[t]]) > 0)
        est$est[[t]][-c(1,2)] <- est$est[[t]][-c(1,2)] - tave
}
}
return(list(est=est, sqres=sqres))

ShowTeam <- function(est, t, threshold = 0) {
    # show the estimates for a team
    nt <- length(est$est[[t]])
    show <- data.frame(matrix(NA, nrow=nt, ncol=3))
    colnames(show) <- c("diff48","se","games")

    # handle empty teams
    if( nt==0) {
        show <- data.frame(matrix(NA, nrow=1, ncol=3))
        colnames(show) <- c("diff48","se","games")
        rownames(show) <- est$teams[[t]]
        return(show)
    }

    teamname <- rep(est$team[t], nt)
    rownames(show) <- sort(unlist(est$teams[[t]]))
    show[, 1] <- round(unlist(est$est[[t]]))
    show[, 2] <- round(unlist(est$se[[t]]))
    show[, 3] <- round(unlist(est$time[[t]]))
    show <- show[abs(show$diff48) >= threshold * show$se,]
    return(show)