# Basketball II

Produced by Dr. Mario

UNC STOR 390

# +/- Player Ratings

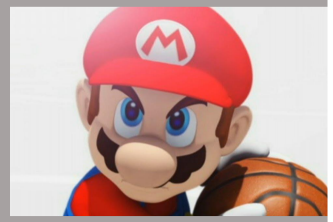
- Recall the Box Score

**Toronto Raptors**

| PLAYERS | MIN | FG | 3FG | FT | +/- | OFF | DEF | REB | AST | STL | BLK | TO | PF | PTS |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Siakam | 46 | 10-17 | 3-6 | 3-4 | 2 | 2 | 8 | 10 | 3 | 1 | 1 | 2 | 2 | 26 |
| Lowry | 42 | 9-16 | 4-7 | 4-6 | 16 | 2 | 5 | 7 | 10 | 3 | 0 | 3 | 5 | 26 |
| Leonard | 41 | 7-16 | 1-5 | 7-8 | -2 | 1 | 5 | 6 | 3 | 2 | 1 | 2 | 4 | 22 |
| Gasol | 27 | 0-5 | 0-2 | 3-4 | -7 | 3 | 6 | 9 | 4 | 0 | 0 | 1 | 4 | 3 |
| Green | 18 | 0-0 | 0-0 | 0-0 | 7 | 0 | 1 | 1 | 3 | 1 | 0 | 1 | 1 | 0 |

Notice the Additional Metric

- Traditional Statistics Do Not Measure Player's Ability to … "Make the Team Better"




# +/- Player Ratings

- Pure +/- Ratings
  - Historically from Hockey
    - Number of Goals a Player's Team Outscores Opponent When a Specific Player is Playing on Ice
    - Highest: Bobby Orr, 1970-1971, +124
    - "Worst Statistic in Hockey" by Hockey-Graphs.com
    - Counting Statistic of Rare Event (Subject to Outliers)
    - Time on Ice Not Reflected
    - Players Who Play the Most and Least Have +/- Closer to 0
    - Weakest Players Not Given Time to Accumulate Negative +/-
    - Same Values are Not Equal i.e. +5 Can Result from Many Scenarios
  - Application to Basketball
    - Pure +/- Statistic Based on Points and Scaled to 48 Minutes
    - Depends on Quality of Players When Player is on Court
    - Players on Bad Teams (Below .500 Record) Get Penalized

# +/- Player Ratings

- Pure +/- Ratings
  - Found on www.82games.com
    - Cavs Championship Season
    - Seasonal Player Stats
    - Lebron James on the Court

$$Net\ Points\ Per\ Min \times 48\ Min = \frac{617}{2709} \times 48$$
$$= 0.22776 \times 48 = 10.9$$

    - Lebron James on Bench

$$Net\ Points\ Per\ Min \times 48\ Min = \frac{-125}{1261} \times 48$$
$$= -0.09913 \times 48 = -4.8$$

    - Pure +/- Per 48 Minutes

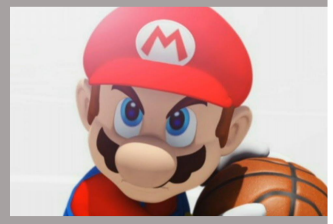
$$Court\ -\ Bench = 10.9 - (-4.8) = 15.7$$

| | | Production | | | On Court/Off Court | | | Simple |
|---|---|---|---|---|---|---|---|---|
| Player | Min | Own | Opp | Net | On | Off | Net | Rating |
| James | 68% | 30.3 | 10.7 | +19.6 | +10.9 | -4.8 | +15.7 | +18.3 |
| Love | 61% | 21.2 | 15.3 | +5.9 | +8.4 | +2.2 | +6.2 | +6.0 |
| Irving | 42% | 21.4 | 18.5 | +2.9 | +5.9 | +6.0 | -0.1 | +1.9 |
| Thompson | 57% | 17.9 | 17.3 | +0.5 | +7.6 | +3.7 | +3.9 | +1.6 |
| McRae | 3% | 15.2 | 16.8 | -1.6 | +13.1 | +5.7 | +7.4 | +1.4 |
| Varejao | 8% | 12.7 | 12.7 | -0.0 | +8.1 | +5.8 | +2.3 | +0.8 |
| Dellavedova | 47% | 12.2 | 15.4 | -3.2 | +9.8 | +2.6 | +7.2 | +0.2 |
| Frye | 11% | 16.3 | 16.7 | -0.3 | +5.5 | +6.0 | -0.5 | -0.4 |
| JR.Smith | 59% | 13.4 | 14.4 | -1.0 | +6.3 | +5.5 | +0.8 | -0.4 |
| Shumpert | 33% | 9.4 | 14.9 | -5.5 | +6.8 | +5.5 | +1.3 | -3.2 |
| Jones | 12% | 12.7 | 13.8 | -1.1 | -2.2 | +7.0 | -9.2 | -3.8 |
| Mozgov | 33% | 16.2 | 19.8 | -3.6 | +0.1 | +8.9 | -8.8 | -5.3 |
| Kaun | 2% | 13.4 | 18.5 | -5.1 | -2.0 | +6.1 | -8.1 | -6.1 |
| Jefferson | 33% | 10.6 | 15.6 | -5.0 | -1.6 | +9.7 | -11.4 | -7.1 |
| Mo.Williams | 19% | 13.4 | 18.5 | -5.2 | -3.2 | +8.1 | -11.3 | -7.2 |
| D.Jones | 1% | 11.1 | 21.6 | -10.6 | -1.1 | +6.0 | -7.2 | -9.4 |
| Cunningham | 9% | 6.4 | 18.3 | -11.9 | -1.9 | +6.7 | -8.6 | -10.8 |
| Harris | 0% | 3.8 | 34.1 | -30.3 | -66.0 | +6.2 | -72.3 | -30.2 |

| Stat | ON Court | OFF Court | Net |
|---|---|---|---|
| Minutes | 2709 | 1261 | 68% |
| Offense: Pts per 100 Poss. | 116.6 | 103.0 | +13.6 |
| Defense: Pts per 100 Poss. | 105.1 | 107.8 | -2.7 |
| Net Points per 100 Possessions | +11.5 | -4.8 | +16.3 |
| Points Scored | 6089 | 2466 | +3623 |
| Points Allowed | 5472 | 2591 | +2881 |
| Net Points | +617 | -125 | +742 |

# +/- Player Ratings

- Adjusted +/- Rating
  - Adjustment for Teammates Played With
  - Adjustment for Opponents Played Against
  - Adjustments Based on Play-by-Play Data Over Whole Season
  - Average +/- Rating = 0
- Simulated Game Data
  - Players 1-9 Compete Against Players 10-18 in 20 Games
  - Assume Starters Play the Entire Game
  - Results of Game Shown Below




```
head(GameData)
```

A tibble: 6 x 12

| Game | Result | P1 | P2 | P3 | P4 | P5 | P6 | P7 | P8 | P9 | P10 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| <int> | <int> | <int> | <int> | <int> | <int> | <int> | <int> | <int> | <int> | <int> | <int> |
| 1 | -13 | 4 | 1 | 7 | 5 | 2 | 15 | 16 | 10 | 17 | 14 |
| 2 | 19 | 1 | 6 | 2 | 5 | 4 | 11 | 17 | 14 | 15 | 18 |
| 3 | -4 | 1 | 9 | 2 | 8 | 4 | 15 | 14 | 10 | 17 | 13 |
| 4 | 29 | 1 | 6 | 5 | 3 | 2 | 16 | 17 | 18 | 14 | 11 |
| 5 | -3 | 9 | 7 | 1 | 5 | 6 | 17 | 15 | 12 | 18 | 10 |
| 6 | 12 | 7 | 2 | 5 | 1 | 4 | 17 | 11 | 15 | 16 | 18 |

# +/- Player Ratings

- Modified Game Data into Matrix (*A*)
  - Each Row is a Different Game (Except Last Row)
  - Each Column is A Different Player
    - 1 = Played on Team 1
    - 0 = Did Not Play
    - -1 = Played on Team 2
  - Notice Last Row of All 1's

```
print(A)
 [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12] [,13] [,14] [,15] [,16] [,17] [,18]
    1    1    0    1    1    0    1    0    0   -1    0    0    0   -1   -1   -1   -1    0
    1    1    0    1    1    1    0    0    0    0   -1    0    0   -1   -1    0   -1   -1
    1    1    0    1    0    0    0    1    1   -1    0    0   -1   -1   -1    0   -1    0
    1    1    1    0    1    1    0    0    0    0   -1    0    0   -1    0   -1   -1   -1
    1    0    0    0    1    1    1    0    1   -1    0   -1    0    0   -1    0   -1   -1
    1    1    0    1    1    0    1    0    0    0   -1    0    0    0   -1   -1   -1   -1
    1    0    0    0    1    1    0    1    1   -1    0   -1   -1    0   -1   -1    0    0
    0    1    1    1    1    0    0    0    1   -1    0   -1    0    0   -1    0   -1   -1
    1    0    1    0    0    0    1    1    1    0   -1    0    0   -1   -1   -1   -1    0
    1    1    0    1    0    1    0    0    1   -1   -1    0   -1    0    0   -1    0   -1
    0    1    1    0    1    1    1    0    0    0    0   -1   -1   -1   -1    0   -1    0
    0    0    1    1    0    1    1    1    0    0   -1   -1    0    0   -1    0   -1   -1
    0    1    0    1    1    1    0    0    1    0   -1    0   -1   -1    0    0   -1   -1
    1    1    0    1    0    0    1    1    0    0    0   -1   -1   -1    0    0   -1   -1
    1    0    0    0    0    1    1    1    1   -1    0   -1    0   -1   -1    0   -1    0
    1    1    1    0    0    1    0    1    0   -1    0    0    0   -1    0   -1   -1   -1
    0    1    1    0    1    1    1    0    0   -1   -1    0   -1   -1    0   -1    0    0
    1    0    1    1    0    1    1    0    0    0   -1    0   -1   -1   -1    0    0   -1
    1    1    0    0    1    1    0    1    0    0    0   -1   -1   -1   -1    0    0   -1
    0    1    1    1    1    0    0    1    0   -1   -1    0    0    0    0   -1   -1   -1
    1    1    1    1    1    1    1    1    1    1    1    1    1    1    1    1    1    1
```

# +/- Player Ratings

- Game Results into Vector (*y*)
  - Each Element is a Different Game (Except Last One)
  - Notice 0 in Last Element

- Code Used to Create Matrix *A* and Vector *y*

```
#Modifed Data
GameData2 = cbind(GameData[,1:2],matrix(NA,20,18))
names(GameData2)[3:20]=paste("Player",1:18,sep="")

for(j in 1:20){
  for(k in 1:18)
  GameData2[j,k+2]= as.numeric(k %in% GameData[j,3:12])
}

GameData2[,12:20]=-GameData2[,12:20]
Games.Played=colSums(GameData2[,3:20])


#Added Constraint to Data (Sum of Effects = 0)
GameData2[21,]=c(NA,0,rep(1,18))

#Create Matrix A
A=as.matrix(GameData2[,3:20])

#Create Vector y
y=as.matrix(GameData2[,2])
```

```
print(y)
      [,1]
[1,]   -13
[2,]    19
[3,]    -4
[4,]    29
[5,]    -3
[6,]    12
[7,]    -5
[8,]   -32
[9,]    18
10,]    17
11,]   -11
12,]   -14
13,]    29
14,]    17
15,]    -4
16,]    -7
17,]     9
18,]    24
19,]    18
20,]   -24
21,]     0
```

# +/- Player Ratings

- **Goal: Estimate Adjusted +/- for All 18 Players**
  - Expressed into Vector (*b*)

    $\vec{b} = [b_1, b_2, \cdots, b_{18}]'$
  - Constraint: We Want The Sum of Adjusted +/- to Equal 0
  - We Invoke Constraint With Last Row of A and Element of y
  - Solve the Linear Equation Using Least Squares Regression

$$\vec{y} = \mathbf{A}\vec{b} + \epsilon \longrightarrow \vec{b} \approx (A'A)^{-1}A'\vec{y}$$

  - Code for Solving System of Linear Equations

```
#Solve Linear Equations
b=solve(t(A)%*%A)%*%t(A)%*%y
```

  - Adjusted +/- For Each Player →

```
> print(b)
                [,1]
Player1    13.441601
Player2    -1.306216
Player3    -7.755180
Player4    -7.446202
Player5     1.759840
Player6    -1.925423
Player7     2.055698
Player8    -3.983937
Player9     7.633862
Player10   14.295170
Player11  -14.596050
Player12    1.285212
Player13   -4.753639
Player14  -11.077068
Player15    9.742491
Player16    2.115617
Player17    6.092244
Player18   -5.578021
```

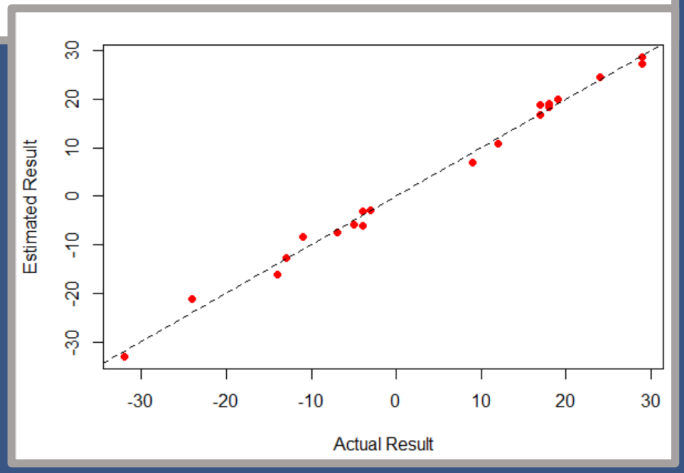
# +/- Player Ratings

- Can Be Used to Approximate Game Result
  - Code to Calculate Predicted Scores Using Adjusted +/-

```
Approx.Score=rep(NA,20)
for(k in 1:20){
  Team1Total=sum(as.numeric(b)[as.numeric(GameData[k,3:7])])
  Team2Total=sum(as.numeric(b)[as.numeric(GameData[k,8:12])])
  Approx.Score[k]=Team1Total-Team2Total
}
```

  - Code and Graphic Comparing Predicted Versus Actual

```
plot(x=y[-21],y=Approx.Score,col="red",pch=16,
     xlab="Actual Result",ylab="Estimated Result")
abline(a=0,b=1,lty=2)
```



# Basketball Statistics

- **Classic Measures of Field Goal Percentage**

$$FG\% = \frac{FGM}{FGA} \qquad 3FG\% = \frac{3FGM}{3FGA}$$

FGM = Field Goal Made
FGA = Field Goal Attempt
3FGM = 3-Pointer Made
3FGA = 3-Pointer Attempt

- **Effective Field Goal Percentage (EFG)**
  - **Problem with Previous Metrics**
    - **Knicks: 15/20 Field Goals = 30 Points**
    - **Lakers: 15/20  3-Pt Field Goals = 45 Points**
    - **Same Field Goal Percentage (75%)**
  - **New Metric**

$$EFG\% = \frac{FGM + 0.5 \times 3FGM}{FGA}$$

  - **Adjusted EFG%**
    - **Knicks: 75%**
    - **Lakers: 1125%**

# Final Inspiration

Nine out of 10 schools are cheating.
The other one is in last place.

- Jerry Tarkanian