Regression Fit

Just going to use the data to see what different methods of forecasting look like.

```
install.packages("readr")
```

```
## Installing package into '/home/rstudio-user/R/x86_64-pc-linux-gnu-library/3.4'
## (as 'lib' is unspecified)
```

```
library("readr", lib.loc="~/R/x86_64-pc-linux-gnu-library/3.4")
All_Season_Stats=read_csv("All_Season_Stats.csv")
```

```
## Parsed with column specification:
## cols(
##   .default = col_integer(),
##   Player = col_character(),
##   Pos = col_character(),
##   Tm = col_character(),
##   `3P_16` = col_double(),
##   `2P_16` = col_double(),
##   FT_16 = col_double(),
##   FTA_16 = col_double(),
##   ORB_16 = col_double(),
##   DRB_16 = col_double()
## )
```

```
## See spec(...) for full column specifications.
```

```
tail(All_Season_Stats,25)
```

```
## # A tibble: 25 x 104
##    Player         Pos     Age Tm     G_18 GS_18 MP_18 FG_18 FGA_18 `3P_18`
##    <chr>          <chr> <int> <chr> <int> <int> <int> <int>  <int>   <int>
##  1 Vander Blue    SG       25 LAL       5     0    45     1      5       0
##  2 Victor Oladipo SG       25 IND      75    75  2552   640   1342     161
##  3 Vince Carter   SF       41 SAC      58     5  1026   114    283      57
##  4 Vince Hunter   PF       23 MEM       4     0     7     3      5       0
##  5 Wade Baldwin   PG       21 POR       7     0    80    14     21       4
##  6 Walt Lemon Jr. PG       25 NOP       5     0    35     7     16       1
##  7 Wayne Ellingt~ SG       30 MIA      77     2  2041   285    700     227
##  8 Wayne Selden   SG       23 MEM      35     9   692   116    269      49
##  9 Wesley Iwundu  SF       23 ORL      62    12  1020    93    218       9
## 10 Wesley Johnson SF       30 LAC      74    40  1486   150    368      60
## # ... with 15 more rows, and 94 more variables: `3PA_18` <int>,
## #   `2P_18` <int>, `2PA_18` <int>, FT_18 <int>, FTA_18 <int>,
## #   ORB_18 <int>, DRB_18 <int>, TRB_18 <int>, AST_18 <int>, STL_18 <int>,
## #   BLK_18 <int>, TOV_18 <int>, PF_18 <int>, `PS/G_18` <int>, G_17 <int>,
## #   GS_17 <int>, MP_17 <int>, FG_17 <int>, FGA_17 <int>, `3P_17` <int>,
## #   `3PA_17` <int>, `2P_17` <int>, `2PA_17` <int>, FT_17 <int>,
## #   FTA_17 <int>, ORB_17 <int>, DRB_17 <int>, TRB_17 <int>, AST_17 <int>,
## #   STL_17 <int>, BLK_17 <int>, TOV_17 <int>, PF_17 <int>,
## #   `PS/G_17` <int>, G_16 <int>, GS_16 <int>, MP_16 <int>, FG_16 <int>,
## #   FGA_16 <int>, `3P_16` <dbl>, `3PA_16` <int>, `2P_16` <dbl>,
## #   `2PA_16` <int>, FT_16 <dbl>, FTA_16 <dbl>, ORB_16 <dbl>, DRB_16 <dbl>,
## #   TRB_16 <int>, AST_16 <int>, STL_16 <int>, BLK_16 <int>, TOV_16 <int>,
## #   PF_16 <int>, `PS/G_16` <int>, G_15 <int>, GS_15 <int>, MP_15 <int>,
## #   FG_15 <int>, FGA_15 <int>, `3P_15` <int>, `3PA_15` <int>,
## #   `2P_15` <int>, `2PA_15` <int>, FT_15 <int>, FTA_15 <int>,
## #   ORB_15 <int>, DRB_15 <int>, TRB_15 <int>, AST_15 <int>, STL_15 <int>,
## #   BLK_15 <int>, TOV_15 <int>, PF_15 <int>, `PS/G_15` <int>, G_14 <int>,
## #   GS_14 <int>, MP_14 <int>, FG_14 <int>, FGA_14 <int>, `3P_14` <int>,
## #   `3PA_14` <int>, `2P_14` <int>, `2PA_14` <int>, FT_14 <int>,
## #   FTA_14 <int>, ORB_14 <int>, DRB_14 <int>, TRB_14 <int>, AST_14 <int>,
## #   STL_14 <int>, BLK_14 <int>, TOV_14 <int>, PF_14 <int>, `PS/G_14` <int>
```

```
##The data as it stands could be reorganized, I am going to reorgainzie it by season with an emphasys or
```

I'll Start by Making the data Set using the data from online

```
Name=rep(NA,6*length(All_Season_Stats$Player))#Making the data set full with NA's first, this will have

i=1


while(floor(i/6)+1<=length(All_Season_Stats$Player)){###the floor(i/5)+1 always rounds to the first name

Name[i]=All_Season_Stats$Player[floor(i/6)+1]
Name[i+1]=All_Season_Stats$Player[floor(i/6)+1]
Name[i+2]=All_Season_Stats$Player[floor(i/6)+1]
Name[i+3]=All_Season_Stats$Player[floor(i/6)+1]
Name[i+4]=All_Season_Stats$Player[floor(i/6)+1]
Name[i+5]=All_Season_Stats$Player[floor(i/6)+1]
i=i+6
}
Name=as.factor(Name)
tail(Name,25)##Now names are factors that repeat 6 times in the data set for each name
```

```
##  [1] Zach Collins  Zach LaVine   Zach LaVine   Zach LaVine   Zach LaVine
##  [6] Zach LaVine   Zach LaVine   Zach Randolph Zach Randolph Zach Randolph
## [11] Zach Randolph Zach Randolph Zach Randolph Zaza Pachulia Zaza Pachulia
## [16] Zaza Pachulia Zaza Pachulia Zaza Pachulia Zaza Pachulia Zhou Qi
## [21] Zhou Qi       Zhou Qi       Zhou Qi       Zhou Qi       Zhou Qi
## 540 Levels: Aaron Brooks Aaron Gordon Aaron Harrison ... Zhou Qi
```

```
Pos2=rep(NA,6*length(All_Season_Stats$Player))##This will repeat the position of the player listed in th



##Since my goal for this is the same as with name, to repeat the position 6 times, i will re use my old

j=1


while(floor(j/6)+1<=length(All_Season_Stats$Pos)){###the floor(j/5)+1 always rounds to the first positio
```

```r
Pos2[j]=All_Season_Stats$Pos[floor(j/6)+1]
Pos2[j+1]=All_Season_Stats$Pos[floor(j/6)+1]
Pos2[j+2]=All_Season_Stats$Pos[floor(j/6)+1]
Pos2[j+3]=All_Season_Stats$Pos[floor(j/6)+1]
Pos2[j+4]=All_Season_Stats$Pos[floor(j/6)+1]
Pos2[j+5]=All_Season_Stats$Pos[floor(j/6)+1]

j=j+6
}
Pos2=as.factor(Pos2)
tail(Pos2,25)##Now Pos2 is a vector of positions as factors
```

```
##  [1] C  SG SG SG SG SG SG PF PF PF PF PF PF C  C  C  C  C  C  C  C  C  C
## [24] C  C
## Levels: C PF PG PG-SG SF SF-SG SG
```

```r
Age=rep(NA,6*length(All_Season_Stats$Player))##This will end up being a list of ages, starting at curre






k=1


while(floor(k/6)+1<=length(All_Season_Stats$Pos)){###the floor(j/6)+1 always rounds to the first name f





Age[k]=All_Season_Stats$Age[floor(k/6)+1]+1##Predicted age is one more then age in 2018
Age[k+1]=All_Season_Stats$Age[floor(k/6)+1]###Using the same old loop which repeats entries I edited it
Age[k+2]=All_Season_Stats$Age[floor(k/6)+1]-1
Age[k+3]=All_Season_Stats$Age[floor(k/6)+1]-2
Age[k+4]=All_Season_Stats$Age[floor(k/6)+1]-3
Age[k+5]=All_Season_Stats$Age[floor(k/6)+1]-4


k=k+6
}
tail(Age,25)
```

```
##  [1] 16 23 22 21 20 19 18 37 36 35 34 33 32 34 33 32 31 30 29 23 22 21 20
## [24] 19 18
```

```r
##Tm I am not going to use for anaylsys

#I won't be using games played, games started, or minutes played in the analysis

Year=rep(NA,6*length(All_Season_Stats$Player))##This will contain years 2019 through 2014

y=c(2019,2018,2017,2016,2015,2014)
```

```r
Year=rep(y,length(All_Season_Stats$Player))

tail(Year,25)
```

```
##  [1] 2014 2019 2018 2017 2016 2015 2014 2019 2018 2017 2016 2015 2014 2019
## [15] 2018 2017 2016 2015 2014 2019 2018 2017 2016 2015 2014
```

```r
FG=rep(NA,6*length(All_Season_Stats$Player))##This will contain a record of fieldgoals made on total in

ii=2###Start at 2 because first entry for each player is year 2019 nobody has made any FG
while(ii<=(6*length(All_Season_Stats$Player))){

##First entry stays NA each time
FG[ii]=All_Season_Stats$FG_18[floor(ii/6)+1]
FG[ii+1]=All_Season_Stats$FG_17[floor(ii/6)+1]###Every 5th entry is the field goals for 18, and so on
FG[ii+2]=All_Season_Stats$FG_16[floor(ii/6)+1]
FG[ii+3]=All_Season_Stats$FG_15[floor(ii/6)+1]
FG[ii+4]=All_Season_Stats$FG_14[floor(ii/6)+1]

ii=ii+6
}
tail(FG,25)
```

```
##  [1]  NA  NA 136 326 433 286  NA  NA 361 433 431 454 560  NA 149 164 219
## [18] 240 149  NA   6  NA  NA  NA  NA
```

```r
#I won't be using field goals attempted

Bhind_Arc=rep(NA,6*length(All_Season_Stats$Player))##This will contain the 3 point field goals made per

jj=2
while(jj<=(6*length(All_Season_Stats$Player))){
Bhind_Arc[jj]=All_Season_Stats$`3P_18`[floor(jj/6)+1]
Bhind_Arc[jj+1]=All_Season_Stats$`3P_17`[floor(jj/6)+1]###Every 5th entry is the 3 points made for 18, a
Bhind_Arc[jj+2]=All_Season_Stats$`3P_16`[floor(jj/6)+1]
Bhind_Arc[jj+3]=All_Season_Stats$`3P_15`[floor(jj/6)+1]
Bhind_Arc[jj+4]=All_Season_Stats$`3P_14`[floor(jj/6)+1]

jj=jj+6
}
tail(Bhind_Arc,25)
```

```
##  [1]      NA      NA  42.000 120.000   0.452  57.000      NA      NA
##  [9]  51.000  21.000   0.475   7.000   2.000      NA   0.000   0.000
## [17]   0.466   0.000   0.000      NA   2.000      NA      NA      NA
## [25]      NA
```

```r
In_Arc=rep(NA,length(All_Season_Stats$Player))#This will contain 2 point field goals made total through



l=2
while(l<=(6*length(All_Season_Stats$Player))){
In_Arc[l]=All_Season_Stats$`2P_18`[floor(l/6)+1]
In_Arc[l+1]=All_Season_Stats$`2P_17`[floor(l/6)+1]###Every 5th entry is the 2 points made for 18, and s
In_Arc[l+2]=All_Season_Stats$`2P_16`[floor(l/6)+1]
In_Arc[l+3]=All_Season_Stats$`2P_15`[floor(l/6)+1]
In_Arc[l+4]=All_Season_Stats$`2P_14`[floor(l/6)+1]

l=l+6
}
tail(In_Arc,25)
```

```
##  [1]      NA      NA  94.000 206.000   0.389 229.000      NA      NA
##  [9] 310.000 412.000   0.231 447.000 558.000      NA 149.000 164.000
## [17]   0.000 240.000 149.000      NA   4.000      NA      NA      NA
## [25]      NA
```

```r
FreeT=rep(NA,length(All_Season_Stats$Player))##This will contain free throws made total throughout the

ll=2
while(ll<=(6*length(All_Season_Stats$Player))){
FreeT[ll]=All_Season_Stats$FT_18[floor(ll/6)+1]
FreeT[ll+1]=All_Season_Stats$FT_17[floor(ll/6)+1]###Every 5th entry is the free throw made for 18, and
FreeT[ll+2]=All_Season_Stats$FT_16[floor(ll/6)+1]
FreeT[ll+3]=All_Season_Stats$FT_15[floor(ll/6)+1]
FreeT[ll+4]=All_Season_Stats$FT_14[floor(ll/6)+1]

ll=ll+6
}
tail(FreeT,25)
```

```
##  [1]      NA      NA  87.000 117.000   0.482 149.000      NA      NA
##  [9]  84.000 141.000   0.482 228.000 250.000      NA  75.000  98.000
## [17]   0.469 126.000 110.000      NA   8.000      NA      NA      NA
## [25]      NA
```

```r
OfReb=rep(NA,length(All_Season_Stats$Player))##This will contain offensive rebound made total thruoghout



w=2
while(w<=(6*length(All_Season_Stats$Player))){
OfReb[w]=All_Season_Stats$ORB_18[floor(w/6)+1]
OfReb[w+1]=All_Season_Stats$ORB_17[floor(w/6)+1]###Every 5th entry is the offreb for 18, and so on
OfReb[w+2]=All_Season_Stats$ORB_16[floor(w/6)+1]
OfReb[w+3]=All_Season_Stats$ORB_15[floor(w/6)+1]
OfReb[w+4]=All_Season_Stats$ORB_14[floor(w/6)+1]

w=w+6

}
tail(OfReb,25)
```

```
##  [1]  NA  NA   9  19 203  27  NA  NA  97 182 216 225 265  NA  89 140 276
## [18] 197 141  NA   6  NA  NA  NA  NA
```

```r
DefReb=rep(NA,length(All_Season_Stats$Player))##This will be like OfReb but for defensive rebounds



ww=2
```

```r
while(ww<=(6*length(All_Season_Stats$Player))){
DefReb[ww]=All_Season_Stats$DRB_18[floor(ww/6)+1]
DefReb[ww+1]=All_Season_Stats$DRB_17[floor(ww/6)+1]###Every 5th entry is the defreb for 18, and so on
DefReb[ww+2]=All_Season_Stats$DRB_16[floor(ww/6)+1]
DefReb[ww+3]=All_Season_Stats$DRB_15[floor(ww/6)+1]
DefReb[ww+4]=All_Season_Stats$DRB_14[floor(ww/6)+1]

ww=ww+6

}
tail(DefReb,25)
```

```
##  [1]      NA      NA  85.000 141.000   0.793 187.000      NA      NA
##  [9] 300.000 416.000   0.796 522.000 530.000      NA 232.000 270.000
## [17]   0.768 303.000 192.000      NA  16.000      NA      NA      NA
## [25]      NA
```

```r
##Not going to use total rebounds, as in the shiny app a user can just make OfReb and DefReb the same f
```

```r
Assist=rep(NA,length(All_Season_Stats$Player))##This will be the total  assists in each season



m=2
while(m<=(6*length(All_Season_Stats$Player))){
Assist[m]=All_Season_Stats$AST_18[floor(m/6)+1]
Assist[m+1]=All_Season_Stats$AST_17[floor(m/6)+1]###Every 5th entry is the assist for 18, and so on
Assist[m+2]=All_Season_Stats$AST_16[floor(m/6)+1]
Assist[m+3]=All_Season_Stats$AST_15[floor(m/6)+1]
Assist[m+4]=All_Season_Stats$AST_14[floor(m/6)+1]

m=m+6

}

tail(Assist,25)
```

```
##  [1]  NA  NA  72 139 201 276  NA  NA 127 122 350 153 200  NA 109 132 469
## [18] 178 136  NA   2  NA  NA  NA  NA
```

```r
Steal=rep(NA,length(All_Season_Stats$Player))##This will be the total steals in each season

mm=2
while(mm<=(6*length(All_Season_Stats$Player))){
Steal[mm]=All_Season_Stats$STL_18[floor(mm/6)+1]
Steal[mm+1]=All_Season_Stats$STL_17[floor(mm/6)+1]###Every 5th entry is the steal for 18, and so on
Steal[mm+2]=All_Season_Stats$STL_16[floor(mm/6)+1]
Steal[mm+3]=All_Season_Stats$STL_15[floor(mm/6)+1]
Steal[mm+4]=All_Season_Stats$STL_14[floor(mm/6)+1]

mm=mm+6

}
tail(Steal,25)
```

```
##  [1]  NA  NA  24  41 228  54  NA  NA  42  38 529  69  54  NA  38  59 718
## [18]  80  45  NA   2  NA  NA  NA  NA
```

```
Block=rep(NA,length(All_Season_Stats$Player))##This will be the total blocks in each season


q=2
while(q<=(6*length(All_Season_Stats$Player))){
Block[q]=All_Season_Stats$BLK_18[floor(q/6)+1]
Block[q+1]=All_Season_Stats$BLK_17[floor(q/6)+1]###Every 5th entry is the block for 18, and so on
Block[q+2]=All_Season_Stats$BLK_16[floor(q/6)+1]
Block[q+3]=All_Season_Stats$BLK_15[floor(q/6)+1]
Block[q+4]=All_Season_Stats$BLK_14[floor(q/6)+1]

q=q+6

}
tail(Block,25)
```

```
##  [1]  NA  NA   4  10 251  10  NA  NA  10  10 142  14  23  NA  17  33 128
## [18]  21  14  NA  14  NA  NA  NA  NA
```

```
TOV=rep(NA,length(All_Season_Stats$Player))##This will be the total turnovers per game in each season


qq=2
while(qq<=(6*length(All_Season_Stats$Player))){
TOV[qq]=All_Season_Stats$TOV_18[floor(qq/6)+1]
TOV[qq+1]=All_Season_Stats$TOV_17[floor(qq/6)+1]###Every 5th entry is the turnover for 18, and so on
TOV[qq+2]=All_Season_Stats$TOV_16[floor(qq/6)+1]
TOV[qq+3]=All_Season_Stats$TOV_15[floor(qq/6)+1]
TOV[qq+4]=All_Season_Stats$TOV_14[floor(qq/6)+1]

qq=qq+6

}


tail(TOV,25)
```

```
##  [1]  NA  NA  43  85  69 193  NA  NA 116  99  43 156 183  NA  72  87  64
## [18] 133  92  NA  10  NA  NA  NA  NA
```

```
Fouls=rep(NA,length(All_Season_Stats$Player))##This will be the average fouls per game in each season


s=2
while(s<=(6*length(All_Season_Stats$Player))){
Fouls[s]=All_Season_Stats$PF_18[floor(s/6)+1]
Fouls[s+1]=All_Season_Stats$PF_17[floor(s/6)+1]###Every 5th entry is the turnover for 18, and so on
Fouls[s+2]=All_Season_Stats$PF_16[floor(s/6)+1]
Fouls[s+3]=All_Season_Stats$PF_15[floor(s/6)+1]
Fouls[s+4]=All_Season_Stats$PF_14[floor(s/6)+1]

s=s+6

}
```

```r
tail(Fouls,25)
```

```
##  [1]  NA  NA  55 104  17 158  NA  NA 119 136  13 175 210  NA 122 166  22
## [18] 170 124  NA  14  NA  NA  NA  NA
```

```r
##I won't be using points as we can just do 3 * 3 pointers made 2 * field goals made and 1 * free throws
```

```r
DatSet=data.frame(Name,Pos2,Age,Year,FG,Bhind_Arc,In_Arc,FreeT,OfReb,DefReb,Assist,Steal,Block,TOV,Fouls
tail(DatSet,25)
```

```
##               Name Pos2 Age Year  FG Bhind_Arc  In_Arc   FreeT OfReb
## 3216  Zach Collins    C  16 2014  NA        NA      NA      NA    NA
## 3217   Zach LaVine   SG  23 2019  NA        NA      NA      NA    NA
## 3218   Zach LaVine   SG  22 2018 136    42.000  94.000  87.000     9
## 3219   Zach LaVine   SG  21 2017 326   120.000 206.000 117.000    19
## 3220   Zach LaVine   SG  20 2016 433     0.452   0.389   0.482   203
## 3221   Zach LaVine   SG  19 2015 286    57.000 229.000 149.000    27
## 3222   Zach LaVine   SG  18 2014  NA        NA      NA      NA    NA
## 3223 Zach Randolph   PF  37 2019  NA        NA      NA      NA    NA
## 3224 Zach Randolph   PF  36 2018 361    51.000 310.000  84.000    97
## 3225 Zach Randolph   PF  35 2017 433    21.000 412.000 141.000   182
## 3226 Zach Randolph   PF  34 2016 431     0.475   0.231   0.482   216
## 3227 Zach Randolph   PF  33 2015 454     7.000 447.000 228.000   225
## 3228 Zach Randolph   PF  32 2014 560     2.000 558.000 250.000   265
## 3229 Zaza Pachulia    C  34 2019  NA        NA      NA      NA    NA
## 3230 Zaza Pachulia    C  33 2018 149     0.000 149.000  75.000    89
## 3231 Zaza Pachulia    C  32 2017 164     0.000 164.000  98.000   140
## 3232 Zaza Pachulia    C  31 2016 219     0.466   0.000   0.469   276
## 3233 Zaza Pachulia    C  30 2015 240     0.000 240.000 126.000   197
## 3234 Zaza Pachulia    C  29 2014 149     0.000 149.000 110.000   141
## 3235       Zhou Qi    C  23 2019  NA        NA      NA      NA    NA
## 3236       Zhou Qi    C  22 2018   6     2.000   4.000   8.000     6
## 3237       Zhou Qi    C  21 2017  NA        NA      NA      NA    NA
## 3238       Zhou Qi    C  20 2016  NA        NA      NA      NA    NA
## 3239       Zhou Qi    C  19 2015  NA        NA      NA      NA    NA
## 3240       Zhou Qi    C  18 2014  NA        NA      NA      NA    NA
##       DefReb Assist Steal Block TOV Fouls
## 3216      NA     NA    NA    NA  NA    NA
## 3217      NA     NA    NA    NA  NA    NA
## 3218  85.000     72    24     4  43    55
## 3219 141.000    139    41    10  85   104
## 3220   0.793    201   228   251  69    17
## 3221 187.000    276    54    10 193   158
## 3222      NA     NA    NA    NA  NA    NA
## 3223      NA     NA    NA    NA  NA    NA
## 3224 300.000    127    42    10 116   119
## 3225 416.000    122    38    10  99   136
## 3226   0.796    350   529   142  43    13
## 3227 522.000    153    69    14 156   175
## 3228 530.000    200    54    23 183   210
## 3229      NA     NA    NA    NA  NA    NA
## 3230 232.000    109    38    17  72   122
## 3231 270.000    132    59    33  87   166
## 3232   0.768    469   718   128  64    22
## 3233 303.000    178    80    21 133   170
## 3234 192.000    136    45    14  92   124
## 3235      NA     NA    NA    NA  NA    NA
## 3236  16.000      2     2    14  10    14
## 3237      NA     NA    NA    NA  NA    NA
## 3238      NA     NA    NA    NA  NA    NA
## 3239      NA     NA    NA    NA  NA    NA
## 3240      NA     NA    NA    NA  NA    NA
```

Now I will test a method of modeling with field goals

```
fit1=glm(FG~I(Age^2)+Age+Name)
head(fit1$coefficients,5)
```

```
##        (Intercept)           I(Age^2)                Age
##        -1899.96065           -3.06895         162.52126
##   NameAaron Gordon NameAaron Harrison
##          145.58705         -180.08953
```

```
pred_FG01=predict(fit1,DatSet)
DatSet01=data.frame(DatSet,pred_FG01)
tail(DatSet01,5)
```

```
##         Name Pos2 Age Year FG Bhind_Arc In_Arc FreeT OfReb DefReb Assist
## 3236 Zhou Qi    C  22 2018  6         2      4     8     6     16      2
## 3237 Zhou Qi    C  21 2017 NA        NA     NA    NA    NA     NA     NA
## 3238 Zhou Qi    C  20 2016 NA        NA     NA    NA    NA     NA     NA
## 3239 Zhou Qi    C  19 2015 NA        NA     NA    NA    NA     NA     NA
## 3240 Zhou Qi    C  18 2014 NA        NA     NA    NA    NA     NA     NA
##      Steal Block TOV Fouls  pred_FG01
## 3236     2    14  10    14    6.00000
## 3237    NA    NA  NA    NA  -24.55643
## 3238    NA    NA  NA    NA  -61.25076
## 3239    NA    NA  NA    NA -104.08299
## 3240    NA    NA  NA    NA -153.05311
```

```
install.packages("ggplot2")
```

```
## Installing package into '/home/rstudio-user/R/x86_64-pc-linux-gnu-library/3.4'
## (as 'lib' is unspecified)
```

```
library("ggplot2", lib.loc="~/R/x86_64-pc-linux-gnu-library/3.4")
```

```
GraphFG_Name=function(Name){
 ii=(which(All_Season_Stats$Player==Name))
ggplot(DatSet01[(6*ii-5):(6*ii),],mapping=aes(x=Age, y=FG,color="Actual"))+geom_line(mapping=aes(x=Age,
labs(title = paste0("Predicted Field Goal's of ",Name),
 subtitle = "Based on data from 2014 to 2018",
 x = "Age",
 y ="Field Goals",
 color = "Actual or Predicted")
}
GraphFG_Num=function(ii){

ggplot(DatSet01[(6*ii-5):(6*ii),],mapping=aes(x=Age, y=FG,color="Actual"))+geom_line(mapping=aes(x=Age,
labs(title = paste0("Predicted Field Goal's of ",All_Season_Stats$Player[ii]),
 subtitle = "Based on data from 2014 to 2018",
 x = "Age",
 y ="Field Goals",
 color = "Actual or Predicted")
}
```

```
install.packages("dplyr")
```

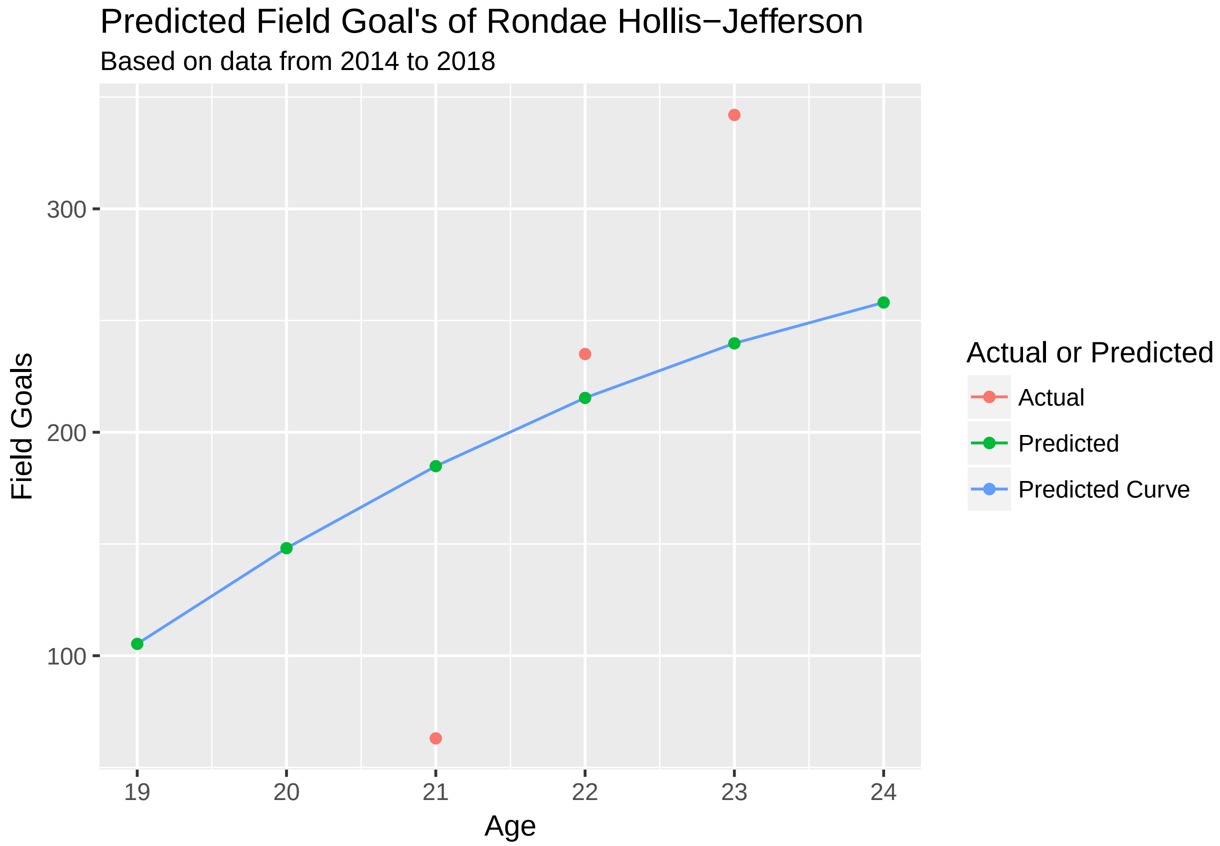
```
## Installing package into '/home/rstudio-user/R/x86_64-pc-linux-gnu-library/3.4'
## (as 'lib' is unspecified)
```

```
library("dplyr", lib.loc="~/R/x86_64-pc-linux-gnu-library/3.4")
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

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

```
GraphFG_Num(442)
```

```
## Warning: Removed 3 rows containing missing values (geom_point).
```



Since that seemed to work so well I will now model all of the attributes with functions like I did field goals

```
fit_Bhind_Arc=glm(Bhind_Arc~I(Age^2)+Age+Name)##A fit for 3 points made total in a season
head(fit_Bhind_Arc$coefficients,5)
```

```
##        (Intercept)           I(Age^2)                Age
##       -466.4827664         -0.5861519         35.4628088
##   NameAaron Gordon NameAaron Harrison
##         48.3082233        -25.6107265
```

```
fit_In_Arc=glm(In_Arc~I(Age^2)+Age+Name)##A fit for 2 points made total in a season
head(fit_In_Arc$coefficients,5)
```

```
##        (Intercept)           I(Age^2)                Age
##       -1420.449217          -2.443577         125.419793
##   NameAaron Gordon NameAaron Harrison
##         101.311044        -150.466050
```

```
fit_FreeT=glm(FreeT~I(Age^2)+Age+Name)##A fit for free throws made total in a season
head(fit_FreeT$coefficients,5)
```

```
##        (Intercept)           I(Age^2)                Age
##        -834.890948          -1.414575          72.920221
##   NameAaron Gordon NameAaron Harrison
##          57.769857         -77.423142
```

```
fit_OfReb=glm(OfReb~I(Age^2)+Age+Name)##A fit for offensive rebounds made total in a season
head(fit_OfReb$coefficients,5)
```

```
##        (Intercept)           I(Age^2)                Age
##       -316.2896783         -0.6142792         30.0658236
##   NameAaron Gordon NameAaron Harrison
##         59.6089701        -44.7711303
```

```
fit_DefReb=glm(DefReb~I(Age^2)+Age+Name)##A fit for defensive rebounds made total in a season
head(fit_DefReb$coefficients,5)
```

```
##        (Intercept)           I(Age^2)                Age
##       -1356.420863          -2.006733         108.493717
##   NameAaron Gordon NameAaron Harrison
##         261.887839         -46.510737
```

```
fit_Assist=glm(Assist~I(Age^2)+Age+Name)##A fit for assists made total in a season
head(fit_Assist$coefficients,5)
```

```
##        (Intercept)           I(Age^2)                Age
##         -705.06856           -1.37939           70.87809
##   NameAaron Gordon NameAaron Harrison
##          -54.76954         -180.37189
```

```
fit_Steal=glm(Steal~I(Age^2)+Age+Name)##A fit for Steals made total in a season
head(fit_Steal$coefficients,5)
```

```
##        (Intercept)           I(Age^2)                Age
##       -287.4471563         -0.5701596         28.0616418
##   NameAaron Gordon NameAaron Harrison
##          3.5057904        -48.5715879
```

```
fit_Block=glm(Block~I(Age^2)+Age+Name)##A fit for Blocks made total in a season
head(fit_Block$coefficients,5)
```

```
##        (Intercept)           I(Age^2)                Age
##       -156.7116524         -0.2841078         14.1841133
##   NameAaron Gordon NameAaron Harrison
##         26.1887888        -16.9745686
```

```
fit_TOV=glm(TOV~I(Age^2)+Age+Name)##A fit for turnovers made total in a season
head(fit_TOV$coefficients,5)
```

```
##        (Intercept)           I(Age^2)                Age
##       -363.5117770         -0.7565463         38.0214787
##   NameAaron Gordon NameAaron Harrison
##        -22.0442815       -103.9546598
```

```
fit_Fouls=glm(Fouls~I(Age^2)+Age+Name)##A fit for Fouls made total in a season
head(fit_Fouls$coefficients,5)
```

```
##        (Intercept)           I(Age^2)                Age
##       -363.1294542         -0.8323177         41.3629497
##   NameAaron Gordon NameAaron Harrison
##         -4.2390862       -130.4587733
```

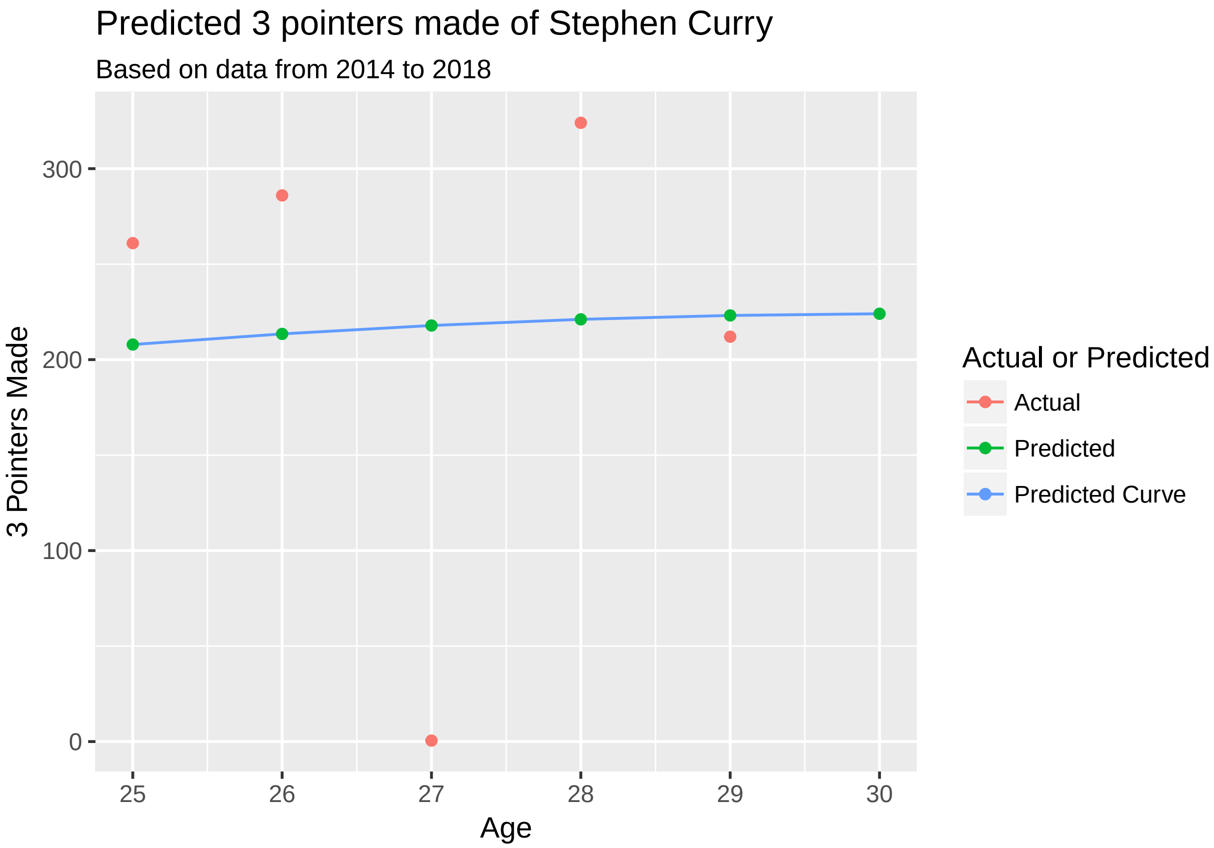
```
pred_Bhind_Arc=predict(fit_Bhind_Arc,DatSet)#Predictions for players 3 point totals in a season at 6 di

pred_In_Arc=predict(fit_In_Arc,DatSet)#Predictions for players 2 point totals in a season at 6 differen

pred_FreeT=predict(fit_FreeT,DatSet)#Predictions for players free throw totals in a season at 6 differe

pred_OfReb=predict(fit_OfReb,DatSet)#Predictions for players offensive rebound totals in a season at 6

pred_DefReb=predict(fit_DefReb,DatSet)#Predictions for players defencsive rebounds totals in a season a

pred_Assist=predict(fit_Assist,DatSet)#Predictions for players assists totals in a season at 6 differen

pred_Steal=predict(fit_Steal,DatSet)#Predictions for players assists totals in a season at 6 different

pred_Block=predict(fit_Block,DatSet)#Predictions for players blocks totals in a season at 6 different a

pred_TOV=predict(fit_TOV,DatSet)#Predictions for players turnovers totals in a season at 6 different ag

pred_Fouls=predict(fit_Fouls,DatSet)#Predictions for players Fouls totals in a season at 6 different ag


DatSetPrime=data.frame(DatSet,pred_Bhind_Arc,pred_In_Arc,pred_FreeT,pred_OfReb,pred_DefReb,pred_Assist,
tail(DatSetPrime,5)##The data set with all the predictions
```

```
##         Name Pos2 Age Year FG Bhind_Arc In_Arc FreeT OfReb DefReb Assist
## 3236 Zhou Qi    C  22 2018  6         2      4     8     6     16      2
## 3237 Zhou Qi    C  21 2017 NA        NA     NA    NA    NA     NA     NA
## 3238 Zhou Qi    C  20 2016 NA        NA     NA    NA    NA     NA     NA
## 3239 Zhou Qi    C  19 2015 NA        NA     NA    NA    NA     NA     NA
## 3240 Zhou Qi    C  18 2014 NA        NA     NA    NA    NA     NA     NA
##      Steal Block TOV Fouls pred_Bhind_Arc pred_In_Arc pred_FreeT
## 3236     2    14  10    14       2.000000     4.00000   8.000000
## 3237    NA    NA  NA    NA      -8.258276   -16.34599  -4.093488
## 3238    NA    NA  NA    NA     -19.688856   -41.57914 -19.016126
## 3239    NA    NA  NA    NA     -32.291739   -71.69944 -36.767915
## 3240    NA    NA  NA    NA     -46.066927  -106.70690 -57.348854
##      pred_OfReb pred_DefReb pred_Assist pred_Steal pred_Block   pred_TOV
## 3236   6.000000   16.000000    2.000000   2.000000  14.000000  10.000000
## 3237   2.348181   -6.204177   -9.564336  -1.544777  12.032524   4.510011
## 3238  -2.532196  -32.421820  -23.887451  -6.229873   9.496833  -2.493070
## 3239  -8.641131  -62.652930  -40.969345 -12.055289   6.392926 -11.009244
## 3240 -15.978625  -96.897508  -60.810018 -19.021024   2.720803 -21.038511
##      pred_Fouls
## 3236  14.000000
## 3237   8.426713
## 3238   1.188791
## 3239  -7.713767
## 3240 -18.280960
```

```
Graph_Bhind_Arc_Name=function(Name){###Graphing behind the arc using functions with either name strings
 ii=(which(All_Season_Stats$Player==Name))
ggplot(DatSetPrime[(6*ii-5):(6*ii),],mapping=aes(x=Age, y=Bhind_Arc,color="Actual"))+geom_line(mapping=a
labs(title = paste0("Predicted 3 pointers made of ",Name),
 subtitle = "Based on data from 2014 to 2018",
 x = "Age",
 y ="3 Pointers Made",
 color = "Actual or Predicted")
}




Graph_Bhind_Arc_Num=function(ii){


  ggplot(DatSetPrime[(6*ii-5):(6*ii),],mapping=aes(x=Age, y=Bhind_Arc,color="Actual"))+geom_line(mapping
labs(title = paste0("Predicted 3 pointers made of ",All_Season_Stats$Player[ii]),
 subtitle = "Based on data from 2014 to 2018",
 x = "Age",
 y ="3 Pointers Made",
 color = "Actual or Predicted")











}
```

```
Graph_Bhind_Arc_Num(466)
```

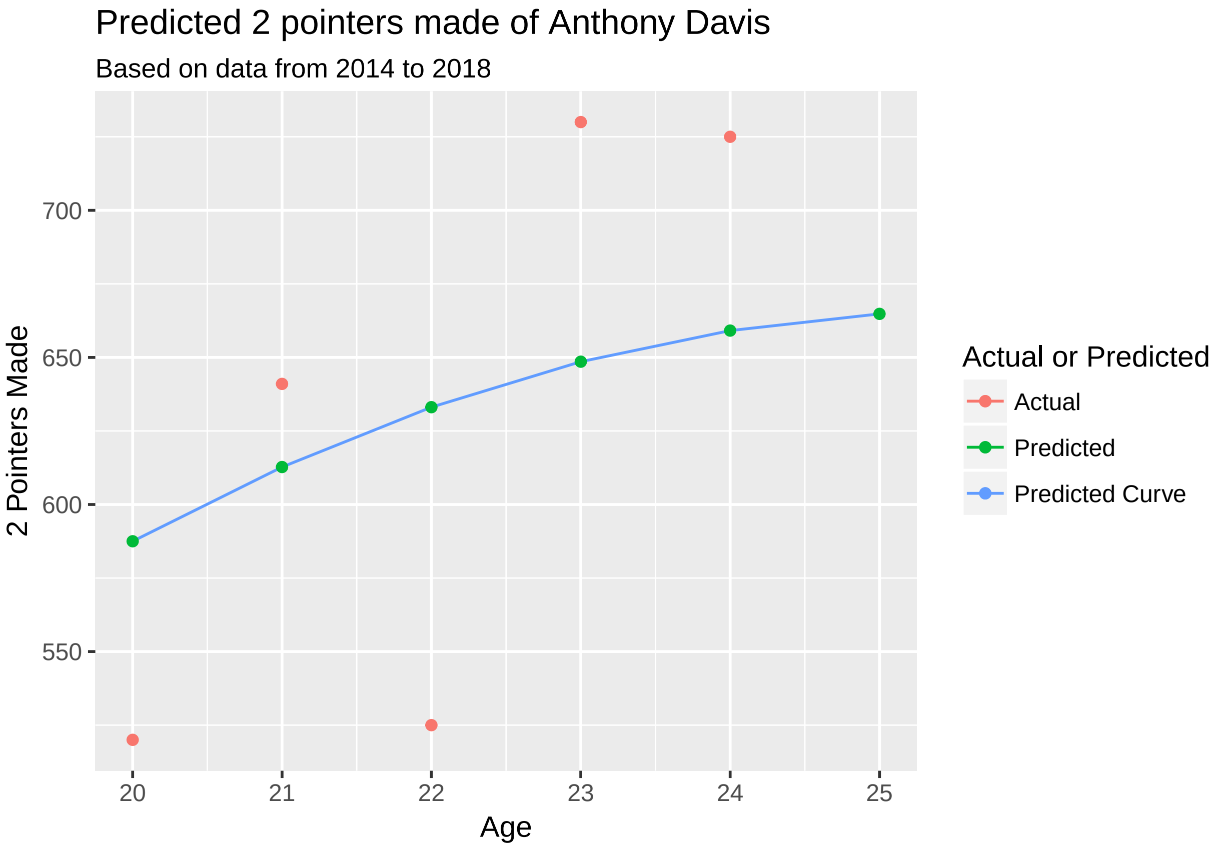
```
## Warning: Removed 1 rows containing missing values (geom_point).
```



```
Graph_In_Arc_Name=function(Name){###Graphing in the arc using functions with either name strings or num
 ii=(which(All_Season_Stats$Player==Name))
ggplot(DatSetPrime[(6*ii-5):(6*ii),],mapping=aes(x=Age, y=In_Arc,color="Actual"))+geom_line(mapping=aes
labs(title = paste0("Predicted 2 pointers made of ",Name),
 subtitle = "Based on data from 2014 to 2018",
 x = "Age",
 y ="2 Pointers Made",
 color = "Actual or Predicted")
}




Graph_In_Arc_Num=function(ii){





  ggplot(DatSetPrime[(6*ii-5):(6*ii),],mapping=aes(x=Age, y=In_Arc,color="Actual"))+geom_line(mapping=ae
labs(title = paste0("Predicted 2 pointers made of ",All_Season_Stats$Player[ii]),
 subtitle = "Based on data from 2014 to 2018",
 x = "Age",
 y ="2 Pointers Made",
 color = "Actual or Predicted")
}
```

```
Graph_In_Arc_Name("Anthony Davis")
```

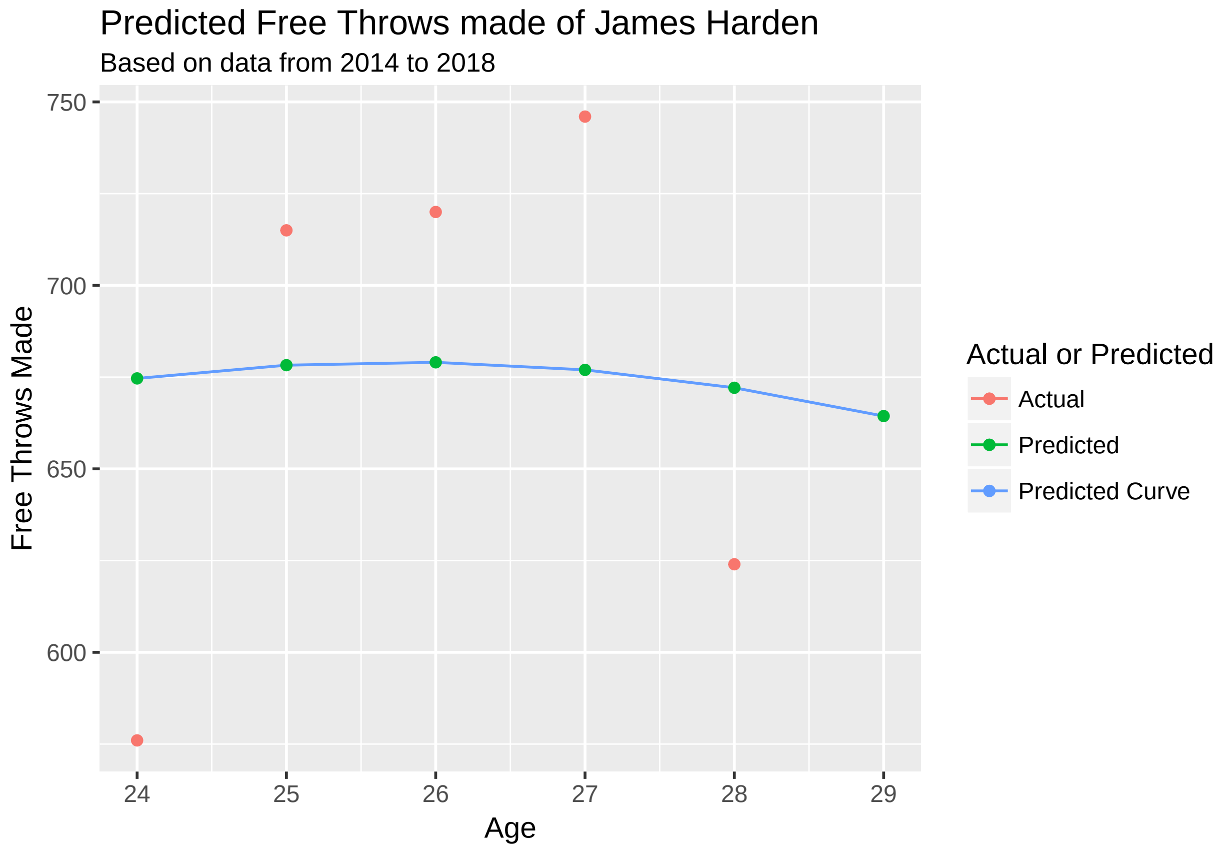
```
## Warning: Removed 1 rows containing missing values (geom_point).
```



```
Graph_FreeT_Name=function(Name){###Graphing free throws using functions with either name strings or num
 ii=(which(All_Season_Stats$Player==Name))
ggplot(DatSetPrime[(6*ii-5):(6*ii),],mapping=aes(x=Age, y=FreeT,color="Actual"))+geom_line(mapping=aes(
labs(title = paste0("Predicted Free Throws made of ",Name),
 subtitle = "Based on data from 2014 to 2018",
 x = "Age",
 y ="Free Throws Made",
 color = "Actual or Predicted")
}




Graph_FreeT_Num=function(ii){



  ggplot(DatSetPrime[(6*ii-5):(6*ii),],mapping=aes(x=Age, y=FreeT,color="Actual"))+geom_line(mapping=ae
labs(title = paste0("Predicted Free Throws made of ",All_Season_Stats$Player[ii]),
 subtitle = "Based on data from 2014 to 2018",
 x = "Age",
 y ="Free Throws Made",
 color = "Actual or Predicted")
}
```

```
Graph_FreeT_Name("James Harden")
```

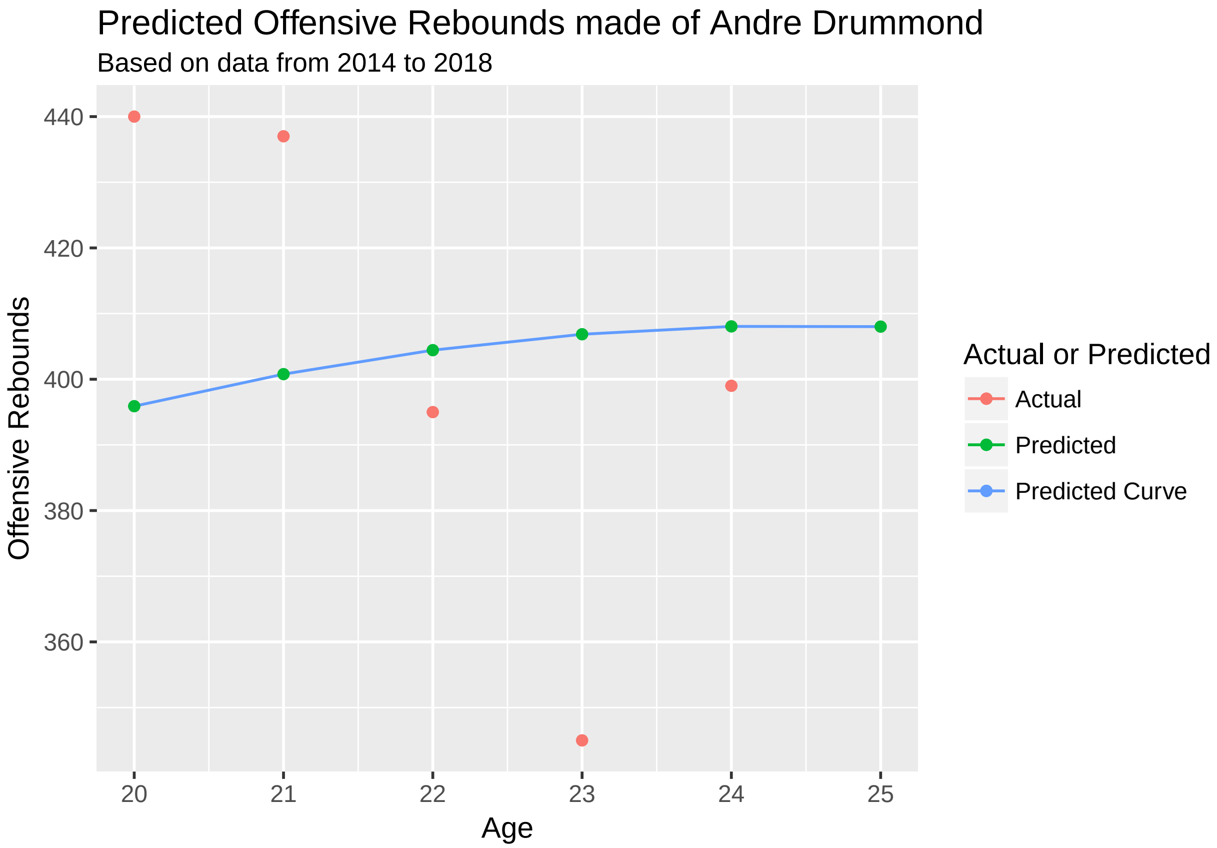
```
## Warning: Removed 1 rows containing missing values (geom_point).
```



```
Graph_OfReb_Name=function(Name){###Graphing Offensive Rebounds using functions with either name strings
 ii=(which(All_Season_Stats$Player==Name))
ggplot(DatSetPrime[(6*ii-5):(6*ii),],mapping=aes(x=Age, y=OfReb,color="Actual"))+geom_line(mapping=aes(
labs(title = paste0("Predicted Offensive Rebounds made of ",Name),
 subtitle = "Based on data from 2014 to 2018",
 x = "Age",
 y ="Offensive Rebounds",
 color = "Actual or Predicted")
}




Graph_OfReb_Num=function(ii){

  ggplot(DatSetPrime[(6*ii-5):(6*ii),],mapping=aes(x=Age, y=OfReb,color="Actual"))+geom_line(mapping=ae
labs(title = paste0("Predicted Offensive Rebounds made of ",All_Season_Stats$Player[ii]),
 subtitle = "Based on data from 2014 to 2018",
 x = "Age",
 y ="Offensive Rebounds",
 color = "Actual or Predicted")
}
```

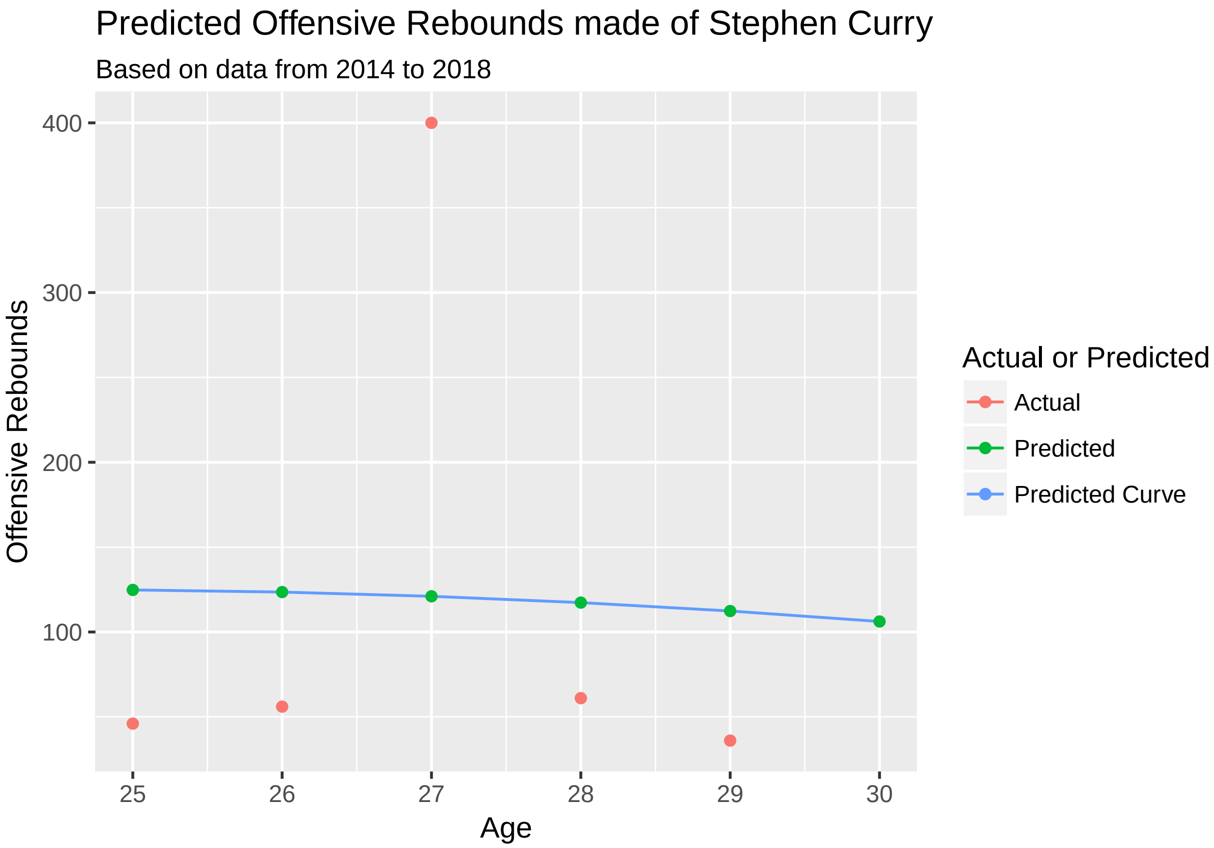
```
Graph_OfReb_Name("Andre Drummond")
```

```
## Warning: Removed 1 rows containing missing values (geom_point).
```



```
Graph_OfReb_Num(466)
```

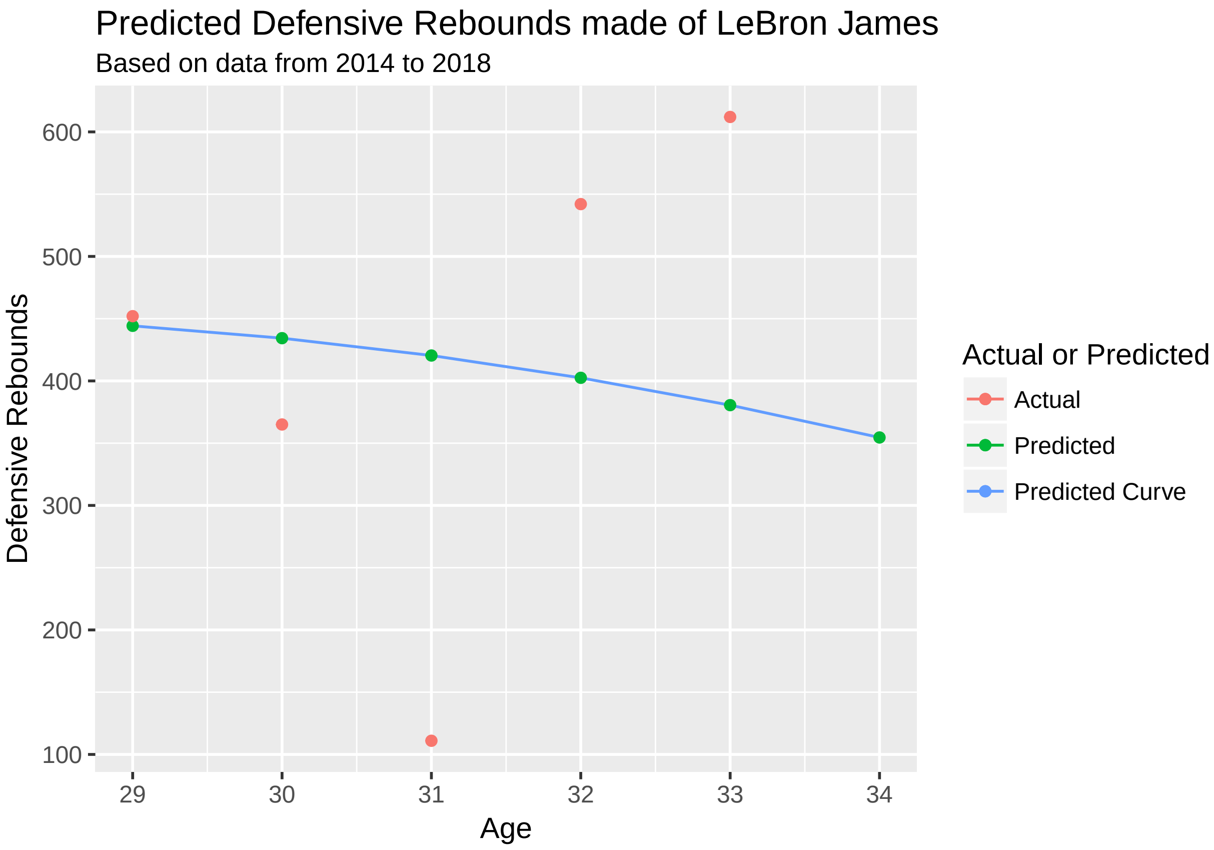
```
## Warning: Removed 1 rows containing missing values (geom_point).
```



```
Graph_DefReb_Name=function(Name){###Graphing defensive Rebounds using functions with either name string
 ii=(which(All_Season_Stats$Player==Name))
ggplot(DatSetPrime[(6*ii-5):(6*ii),],mapping=aes(x=Age, y=DefReb,color="Actual"))+geom_line(mapping=aes
labs(title = paste0("Predicted Defensive Rebounds made of ",Name),
 subtitle = "Based on data from 2014 to 2018",
 x = "Age",
 y ="Defensive Rebounds",
 color = "Actual or Predicted")
}




Graph_DefReb_Num=function(ii){

  ggplot(DatSetPrime[(6*ii-5):(6*ii),],mapping=aes(x=Age, y=DefReb,color="Actual"))+geom_line(mapping=ae
labs(title = paste0("Predicted Defensive Rebounds made of ",All_Season_Stats$Player[ii]),
 subtitle = "Based on data from 2014 to 2018",
 x = "Age",
 y ="Defensive Rebounds",
 color = "Actual or Predicted")
}
```

```
Graph_DefReb_Name("LeBron James")
```

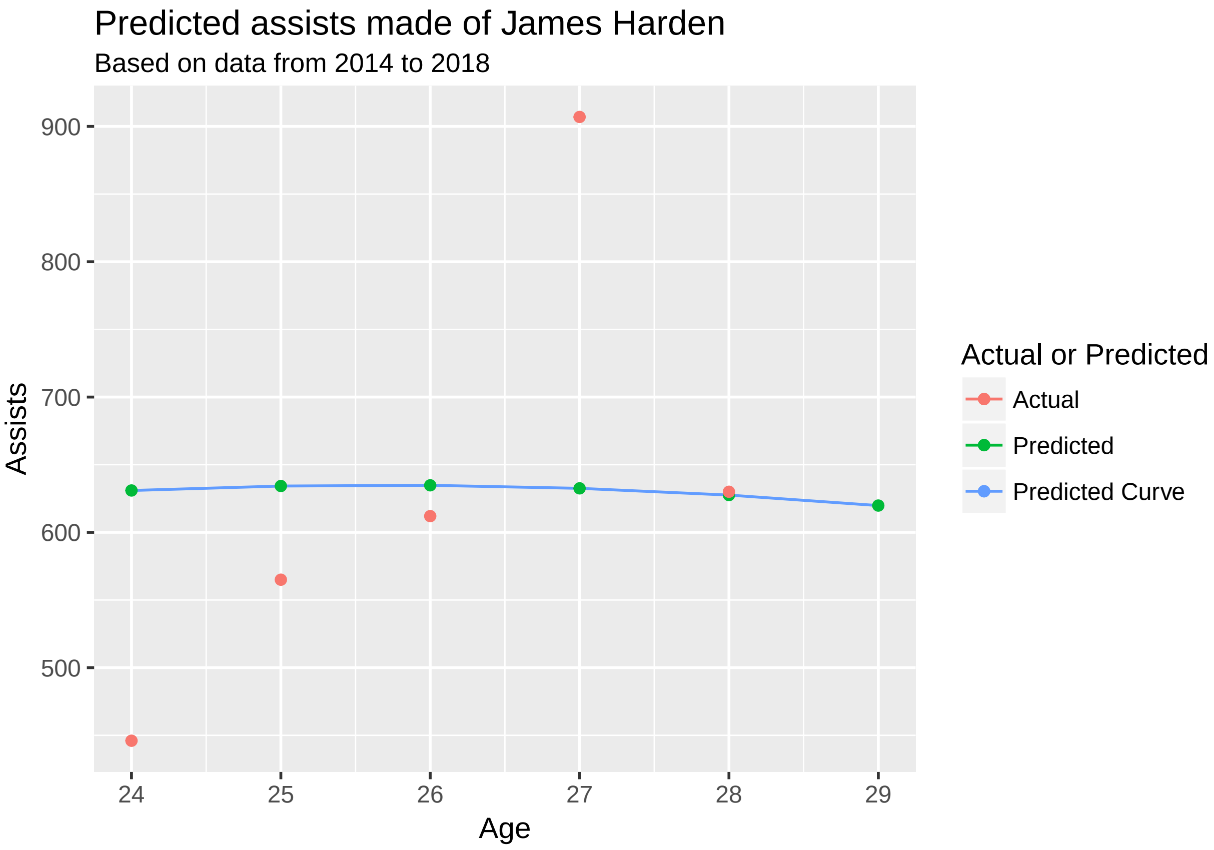
```
## Warning: Removed 1 rows containing missing values (geom_point).
```



```
Graph_Assist_Name=function(Name){###Graphing defensive Rebounds using functions with either name string
 ii=(which(All_Season_Stats$Player==Name))
ggplot(DatSetPrime[(6*ii-5):(6*ii),],mapping=aes(x=Age, y=Assist,color="Actual"))+geom_line(mapping=aes
labs(title = paste0("Predicted assists made of ",Name),
 subtitle = "Based on data from 2014 to 2018",
 x = "Age",
 y ="Assists",
 color = "Actual or Predicted")
}




Graph_Assist_Num=function(ii){


  ggplot(DatSetPrime[(6*ii-5):(6*ii),],mapping=aes(x=Age, y=Assist,color="Actual"))+geom_line(mapping=ae
labs(title = paste0("Predicted assists made of ",All_Season_Stats$Player[ii]),
 subtitle = "Based on data from 2014 to 2018",
 x = "Age",
 y ="Assists",
 color = "Actual or Predicted")
}
```

```
Graph_Assist_Name("James Harden")
```

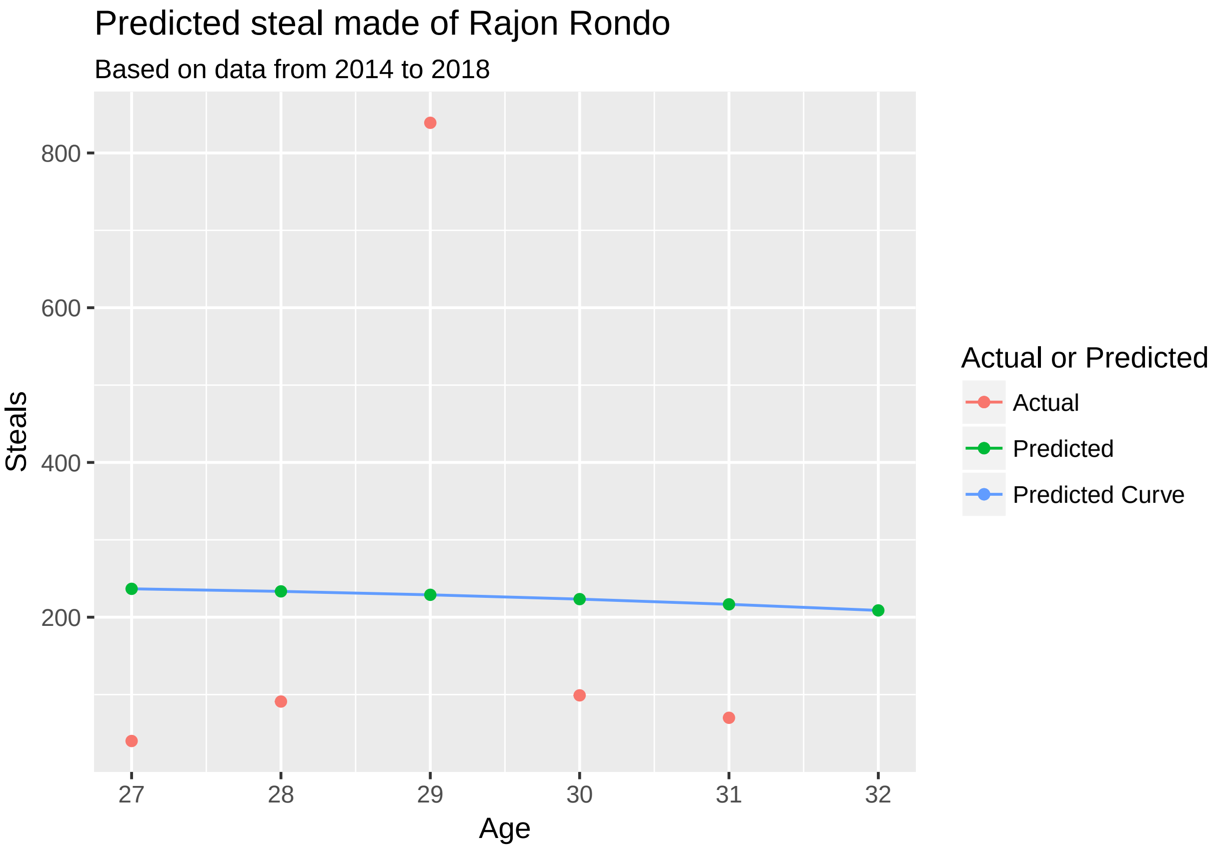
```
## Warning: Removed 1 rows containing missing values (geom_point).
```



```
Graph_Steal_Name=function(Name){###Graphing defensive Rebounds using functions with either name strings
 ii=(which(All_Season_Stats$Player==Name))
ggplot(DatSetPrime[(6*ii-5):(6*ii),],mapping=aes(x=Age, y=Steal,color="Actual"))+geom_line(mapping=aes(
labs(title = paste0("Predicted steal made of ",Name),
 subtitle = "Based on data from 2014 to 2018",
 x = "Age",
 y ="Steals",
 color = "Actual or Predicted")
}




Graph_Steal_Num=function(ii){


  ggplot(DatSetPrime[(6*ii-5):(6*ii),],mapping=aes(x=Age, y=Steal,color="Actual"))+geom_line(mapping=ae
labs(title = paste0("Predicted steal made of ",All_Season_Stats$Player[ii]),
 subtitle = "Based on data from 2014 to 2018",
 x = "Age",
 y ="Steals",
 color = "Actual or Predicted")
}
```

```
Graph_Steal_Name("Rajon Rondo")
```

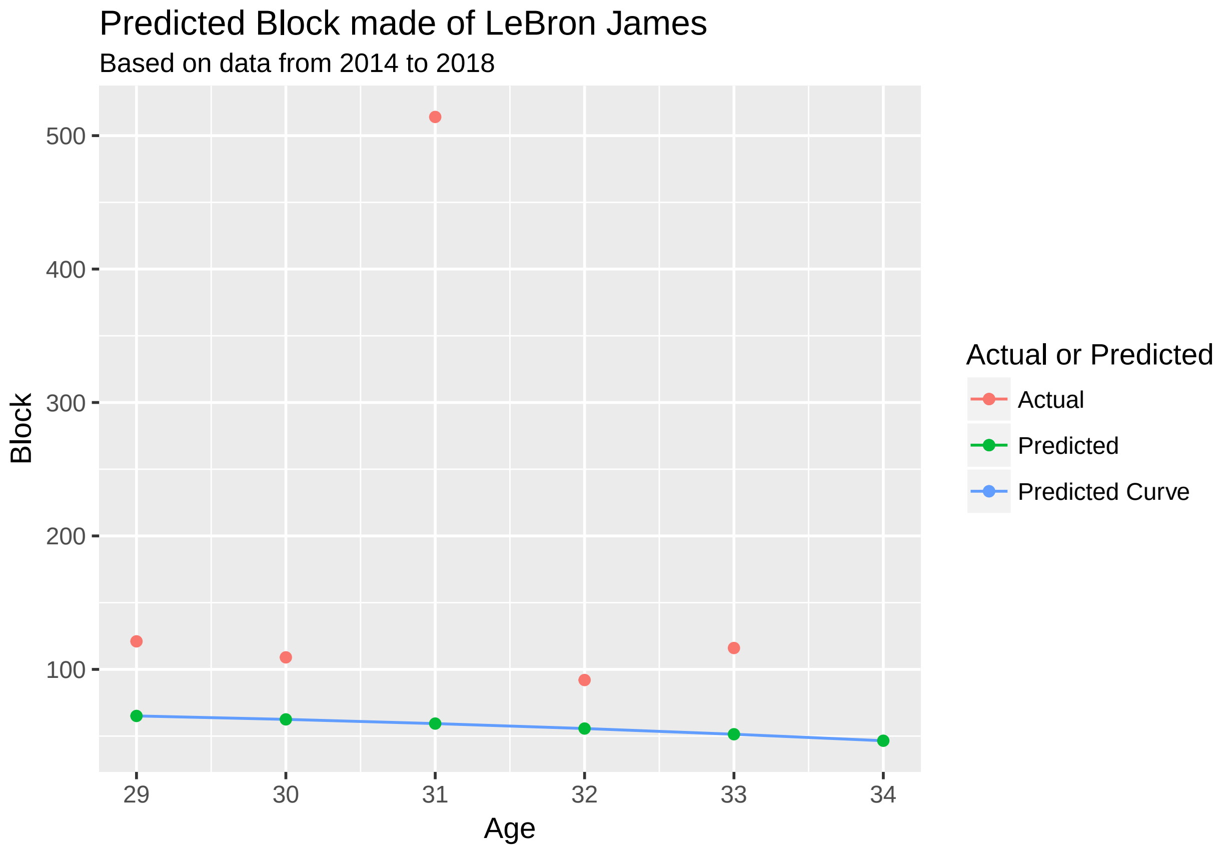
```
## Warning: Removed 1 rows containing missing values (geom_point).
```



```
Graph_Block_Name=function(Name){###Graphing defensive Rebounds using functions with either name strings
 ii=(which(All_Season_Stats$Player==Name))
ggplot(DatSetPrime[(6*ii-5):(6*ii),],mapping=aes(x=Age, y=Steal,color="Actual"))+geom_line(mapping=aes(
labs(title = paste0("Predicted Block made of ",Name),
 subtitle = "Based on data from 2014 to 2018",
 x = "Age",
 y ="Block",
 color = "Actual or Predicted")
}




Graph_Block_Num=function(ii){


 ggplot(DatSetPrime[(6*ii-5):(6*ii),],mapping=aes(x=Age, y=Steal,color="Actual"))+geom_line(mapping=aes
labs(title = paste0("Predicted Block made of ",All_Season_Stats$Player[ii]),
 subtitle = "Based on data from 2014 to 2018",
 x = "Age",
 y ="Block",
 color = "Actual or Predicted")
}
```

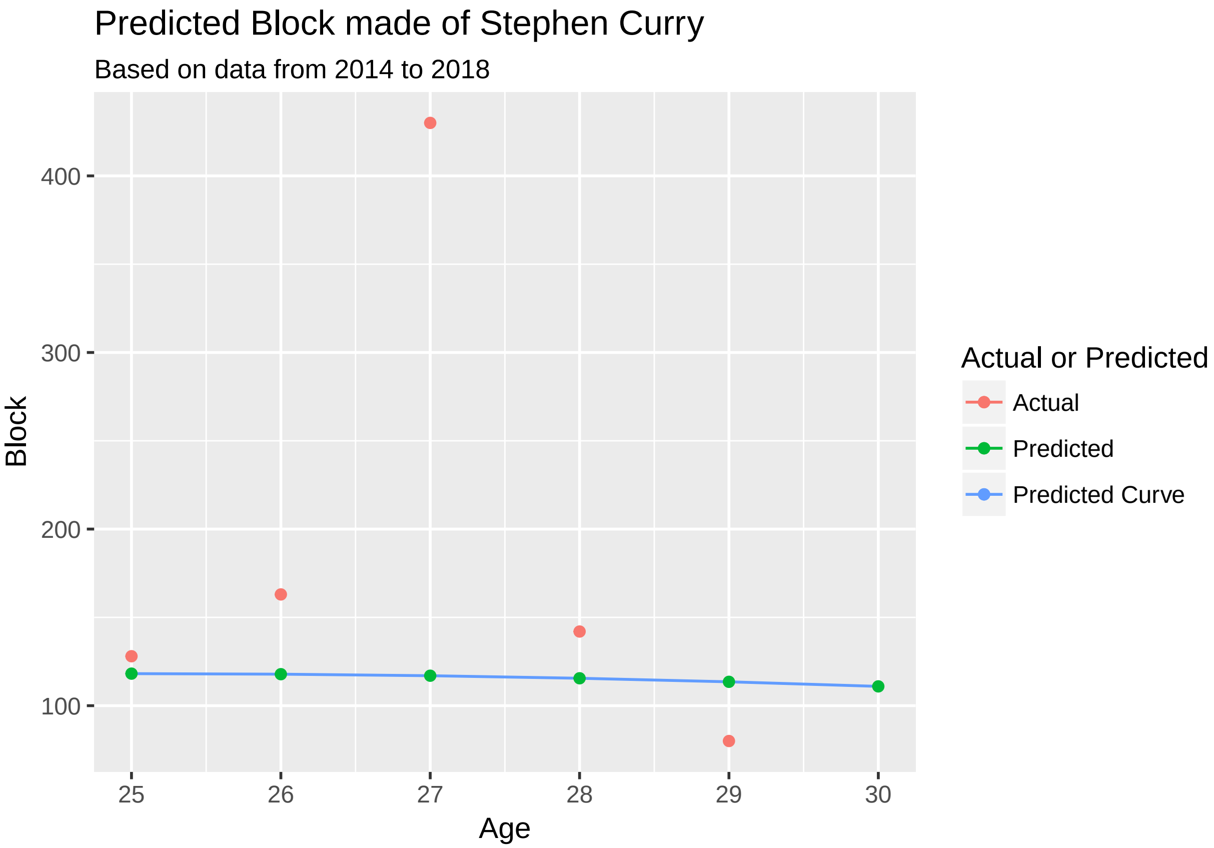
```
Graph_Block_Name("LeBron James")
```

```
## Warning: Removed 1 rows containing missing values (geom_point).
```



```
Graph_Block_Num(466)
```

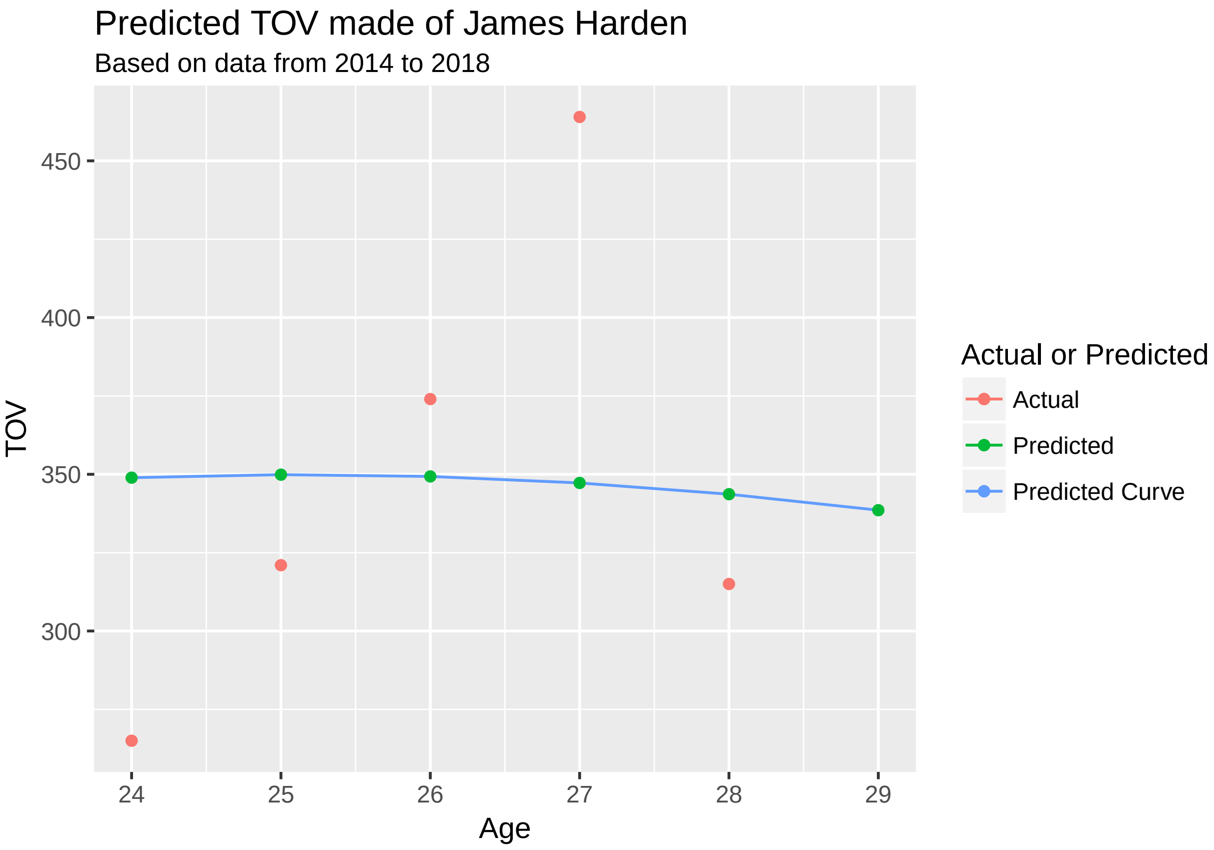
```
## Warning: Removed 1 rows containing missing values (geom_point).
```



```
Graph_TOV_Name=function(Name){###Graphing defensive Rebounds using functions with either name strings o
 ii=(which(All_Season_Stats$Player==Name))
ggplot(DatSetPrime[(6*ii-5):(6*ii),],mapping=aes(x=Age, y=TOV,color="Actual"))+geom_line(mapping=aes(x=
labs(title = paste0("Predicted TOV made of ",Name),
 subtitle = "Based on data from 2014 to 2018",
 x = "Age",
 y ="TOV",
 color = "Actual or Predicted")
}




Graph_TOV_Num=function(ii){



  ggplot(DatSetPrime[(6*ii-5):(6*ii),],mapping=aes(x=Age, y=TOV,color="Actual"))+geom_line(mapping=aes(
labs(title = paste0("Predicted TOV made of ",All_Season_Stats$Player[ii]),
 subtitle = "Based on data from 2014 to 2018",
 x = "Age",
 y ="TOV",
 color = "Actual or Predicted")
}
```

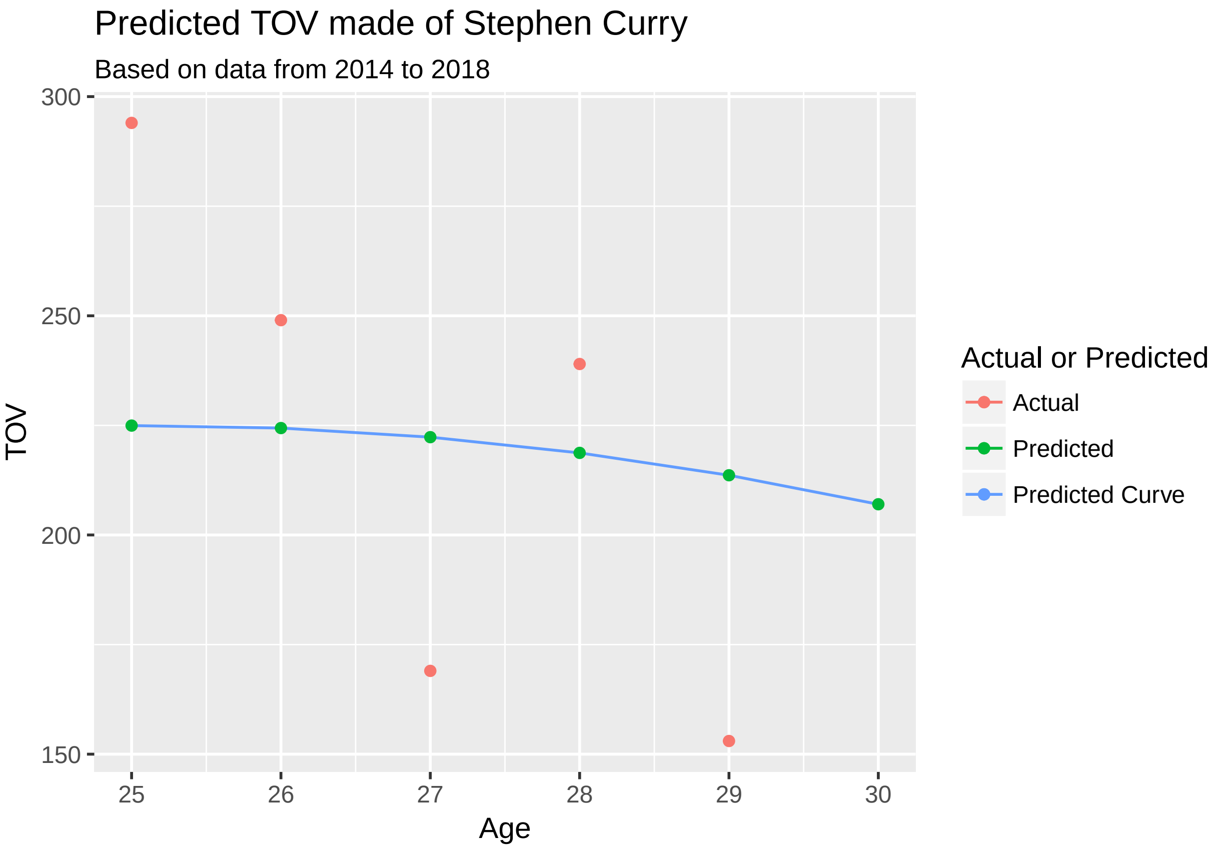
```
Graph_TOV_Name("James Harden")
```

```
## Warning: Removed 1 rows containing missing values (geom_point).
```



```
Graph_TOV_Num(466)
```

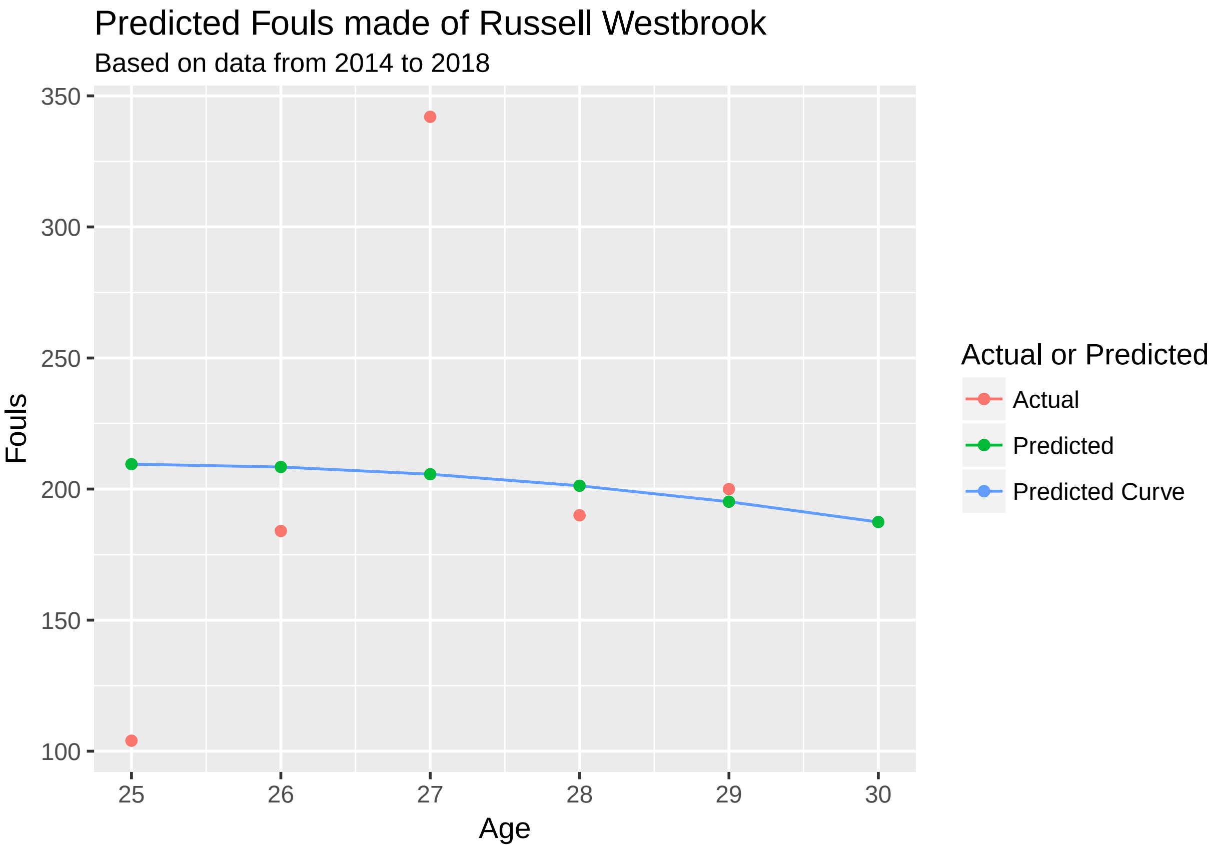
```
## Warning: Removed 1 rows containing missing values (geom_point).
```



```
Graph_Fouls_Name=function(Name){###Graphing defensive Rebounds using functions with either name strings
 ii=(which(All_Season_Stats$Player==Name))
ggplot(DatSetPrime[(6*ii-5):(6*ii),],mapping=aes(x=Age, y=Fouls,color="Actual"))+geom_line(mapping=aes(
labs(title = paste0("Predicted Fouls made of ",Name),
 subtitle = "Based on data from 2014 to 2018",
 x = "Age",
 y ="Fouls",
 color = "Actual or Predicted")
}




Graph_Fouls_Num=function(ii){


  ggplot(DatSetPrime[(6*ii-5):(6*ii),],mapping=aes(x=Age, y=Fouls,color="Actual"))+geom_line(mapping=ae
labs(title = paste0("Predicted Fouls made of ",All_Season_Stats$Player[ii]),
 subtitle = "Based on data from 2014 to 2018",
 x = "Age",
 y ="Fouls",
 color = "Actual or Predicted")
}
```

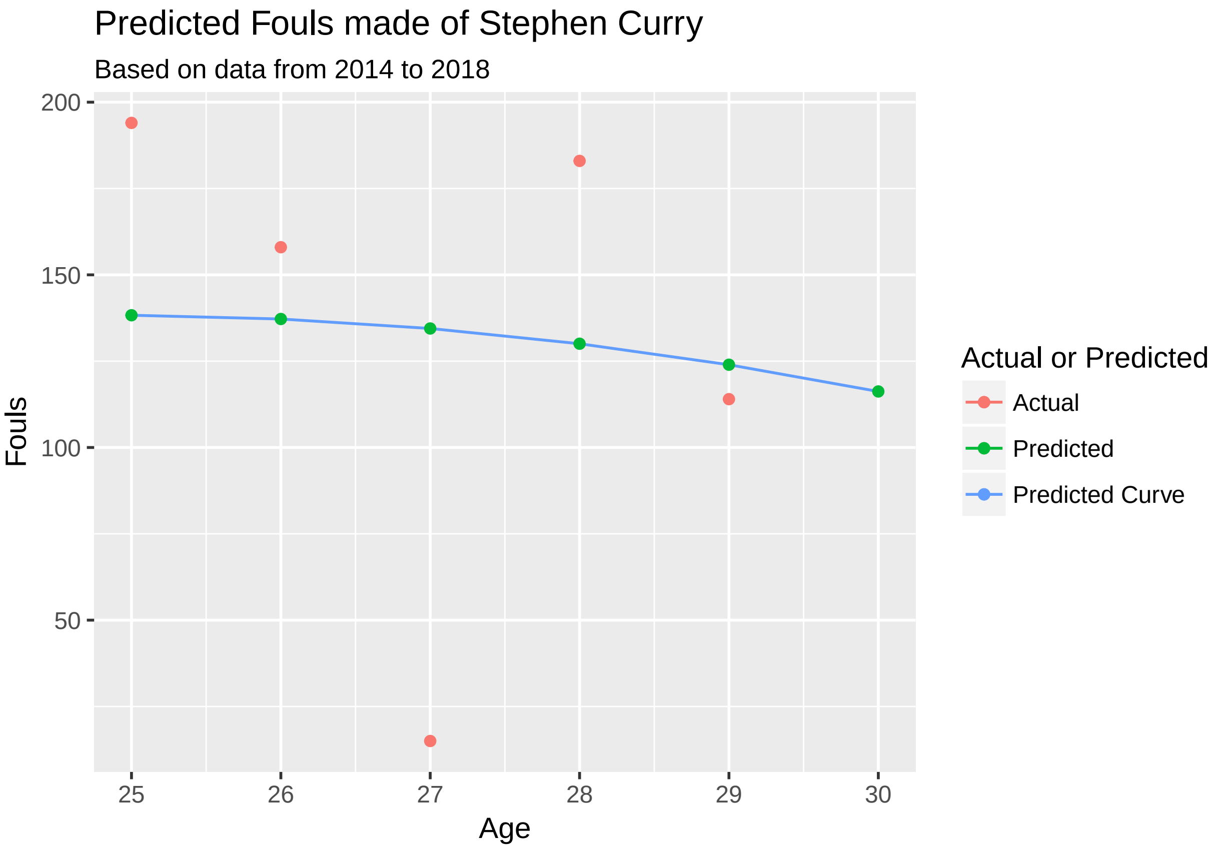
```
Graph_Fouls_Name("Russell Westbrook")
```

```
## Warning: Removed 1 rows containing missing values (geom_point).
```



```
Graph_Fouls_Num(466)
```

```
## Warning: Removed 1 rows containing missing values (geom_point).
```

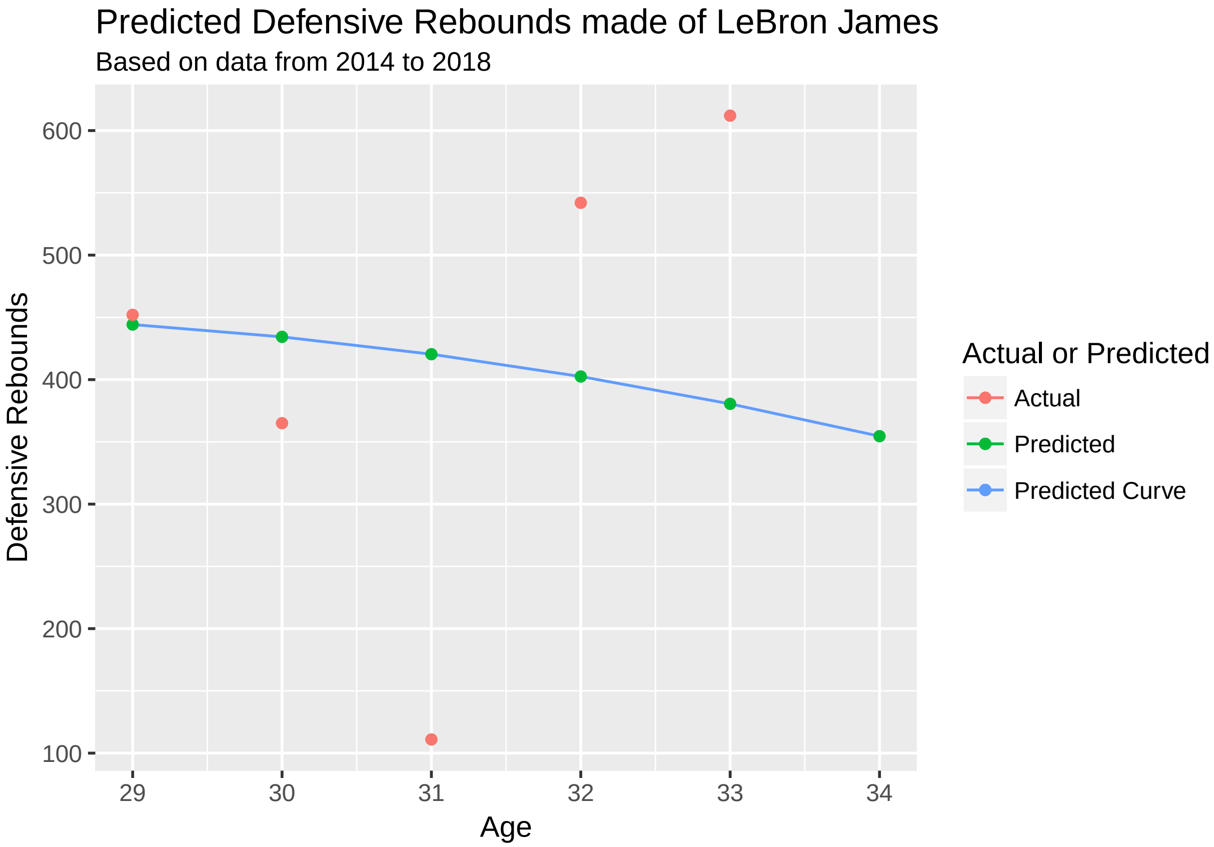


#Now That I have predictions functions for all of them I can make a function of functions

```
MegFun=function(Player_Name="",iii=0,Bhind_Arc=F,In_Arc=F,FreeT=F,OfReb=F,DefReb=F,Assist=F,Steal=F,Blo
  if(Name==""&ii==0){
    return("Enter A Name or Number")##The function is writen so that if you enter a Name then the numbe
  }
  else if(Player_Name==""){
    Player_Name=All_Season_Stats$Player[iii]##but if u dont enter a Name then the number finds the name
  }
  if(Bhind_Arc==T){
    return(Graph_Bhind_Arc_Name(Player_Name))##Do to how return works it only outputs on graph at once
  }
  else if(In_Arc==T){
    return(Graph_In_Arc_Name(Player_Name))##and that graph is the leftmost true value
  }
  else if(FreeT==T){
    return(Graph_FreeT_Name(Player_Name))##in the future we can update it so it can return more then on
  }
  if(OfReb==T){
    return(Graph_OfReb_Name(Player_Name))
  }
    if(DefReb==T){
    return(Graph_DefReb_Name(Player_Name))
  }
  if(Assist==T){
    return(Graph_Assist_Name(Player_Name))
  }
  if(Steal==T){
    return(Graph_Steal_Name(Player_Name))
  }
  if(Block==T){
    return(Graph_Block_Name(Player_Name))
  }
  if(TOV==T){
    return(Graph_TOV_Name(Player_Name))
  }
   if(Fouls==T){
    return(Graph_Fouls_Name(Player_Name))
  }





}
```

```
MegFun(Player_Name="LeBron James",DefReb=T)
```

```
## Warning in if (Name == "" & ii == 0) {: the condition has length > 1 and
## only the first element will be used

## Warning: Removed 1 rows containing missing values (geom_point).
```


Predicted Defensive Rebounds made of LeBron James
Based on data from 2014 to 2018
Defensive Rebounds
Age
Actual or Predicted
Actual
Predicted
Predicted Curve