03 - intro to ddply

R Workshop

- Data Formatting and Reshaping -

Outline

- conditionals & subsets
- `for` loops
- avoiding `for` loops with `ddply`

Baseball Data

- package `plyr` contains data set `baseball`
- seasonal batting statistics of all major league players (until 2007)
- `library(plyr)`
  `help(baseball)`
  `head(baseball)`

```
          id year stint team lg  g  ab  r  h X2b X3b hr rbi sb cs bb so ibb hbp sh sf gidp
4   ansonca01 1871     1  RC1    25 120 29 39  11   3  0  16  6  2  2  1  NA  NA NA NA   NA
44  forceda01 1871     1  WS3    32 162 45 45   9   4  0  29  8  0  4  0  NA  NA NA NA   NA
68  mathebo01 1871     1  FW1    19  89 15 24   3   1  0  10  2  1  2  0  NA  NA NA NA   NA
99  startjo01 1871     1  NY2    33 161 35 58   5   1  1  34  4  2  3  0  NA  NA NA NA   NA
102 suttoez01 1871     1  CL1    29 128 35 45   3   7  3  23  3  1  1  0  NA  NA NA NA   NA
106 whitede01 1871     1  CL1    29 146 40 47   6   5  1  21  2  2  4  1  NA  NA NA NA   NA
```

Baseball Data

- We would like to create career summary statistics for each player
- Plan: subset on a player, and compute statistics

```
ss <- subset(baseball, id=="sosasa01")
head(ss)
            id year stint team lg   g  ab  r   h X2b X3b hr rbi sb cs bb  so ibb hbp sh sf gidp
66822 sosasa01 1989     1  TEX AL  25  84  8  20   3   0  1   3  0  2  0  20   0   0  4  0    3
66823 sosasa01 1989     2  CHA AL  33  99 19  27   5   0  3  10  7  3 11  27   2   2  1  2    3
67907 sosasa01 1990     1  CHA AL 153 532 72 124  26  10 15  70 32 16 33 150   4   6  2  6   10
69018 sosasa01 1991     1  CHA AL 116 316 39  64  10   1 10  33 13  6 14  98   2   2  5  1    5
70599 sosasa01 1992     1  CHN NL  67 262 41  68   7   2  8  25 15  7 19  63   1   4  4  2    4
71757 sosasa01 1993     1  CHN NL 159 598 92 156  25   5 33  93 36 11 38 135   6   4  0  1   14

mean(ss$h/ss$ab)
[1] 0.2681506
```

Baseball Data

- We would like to create career summary statistics for each player
- Plan: subset on a player, and compute statistics

```
ss <- subset(baseball, id=="sosasa01")
head(ss)
            id year stint team lg   g  ab  r   h X2b X3b hr rbi sb cs bb  so ibb hbp sh sf gidp
66822 sosasa01 1989     1  TEX AL  25  84  8  20   3   0  1   3  0  2  0  20   0   0  4  0    3
66823 sosasa01 1989     2  CHA AL  33  99 19  27   5   0  3  10  7  3 11  27   2   2  1  2    3
67907 sosasa01 1990     1  CHA AL 153 532 72 124  26  10 15  70 32 16 33 150   4   6  2  6   10
69018 sosasa01 1991     1  CHA AL 116 316 39  64  10   1 10  33 13  6 14  98   2   2  5  1    5
70599 sosasa01 1992     1  CHN NL  67 262 41  68   7   2  8  25 15  7 19  63   1   4  4  2    4
71757 sosasa01 1993     1  CHN NL 159 598 92 156  25   5 33  93 36 11 38 135   6   4  0  1   14

mean(ss$h/ss$ab)
[1] 0.2681506
```

Need an automatic way of calculating this

`for` loops

- Idea of for loops
  repeat the same (set of) statement(s) for each element of an index set

- Household chores:
  - Introduce counter variable (often times `i`)
  - Reserve space for results

- Generic Setup

```
result <- rep(NA, length(indexset))
for (i in indexset) {
 ... some statements ...
  result[i] <-  ...
}
```

`for` loops Baseball

- Idea of for loops
  repeat the same (set of) statement(s) for each element of an index set

- Household chores:
  - Introduce counter variable (often times `i`)
  - Reserve space for results

- Generic Setup

```
n <- length(indexset)
result <- rep(NA, n)
for (i in 1:n) {
 ... some statements ...
  result[i] <-  ...
}
```

All baseball players' careers

```
players <- unique(baseball$id)
n <- length(players)
ba <- rep(NA, n)

for (i in 1:n) {
 career <- subset(baseball, id == players[i])
 ba[i] <- with(career, mean(h/ab, na.rm=T))
}

summary(ba)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
 0.0000  0.1831  0.2459  0.2231  0.2699  0.5000       6
```

```r
mba <- rep(NA, n)

for (i in 1:n) {
  career <- subset(baseball,
                   id == players[i])


  mba[i] <- with(career, mean(h/ab, na.rm=T))
}
```

```r
mba <- rep(NA, n)

for (i in 1:n) {
  career <- subset(baseball,
                   id == players[i])


  mba[i] <- with(career, mean(h/ab, na.rm=T))
}
```

```
mba <- rep(NA, n)

for (i in 1:n) {
  career <- subset(baseball,
                   id == players[i])


  mba[i] <- with(career, mean(h/ab, na.rm=T))
}
```

mba

NA
NA
NA
NA
NA
NA
NA
NA
NA
NA
NA

```
mba <- rep(NA, n)

for (i in 1:n) {
  career <- subset(baseball,
                   id == players[i])


  mba[i] <- with(career, mean(h/ab, na.rm=T))
}
```

mba

NA
NA
NA
NA
NA
NA
NA
NA
NA
NA
NA

```
mba <- rep(NA, n)

for (i in 1:n) {
  career <- subset(baseball,
                   id == players[i])


  mba[i] <- with(career, mean(h/ab, na.rm=T))
}
```

mba

NA
NA
NA
NA
NA
NA
NA
NA
NA
NA
NA

```
mba <- rep(NA, n)

for (i in 1:n) {
  career <- subset(baseball,
                   id == players[i])


  mba[i] <- with(career, mean(h/ab, na.rm=T))
}
```

i = 1

mba

NA
NA
NA
NA
NA
NA
NA
NA
NA
NA
NA

```
mba <- rep(NA, n)

for (i in 1:n) {
  career <- subset(baseball,
                   id == players[i])


  mba[i] <- with(career, mean(h/ab, na.rm=T))
}
```

i = 1

mba

NA
NA
NA
NA
NA
NA
NA
NA
NA
NA
NA

```
mba <- rep(NA, n)

for (i in 1:n) {
  career <- subset(baseball,
                   id == players[i])


  mba[i] <- with(career, mean(h/ab, na.rm=T))
}
```

i = 1

mba

NA
NA
NA
NA
NA
NA
NA
NA
NA
NA
NA

```
mba <- rep(NA, n)

for (i in 1:n) {
  career <- subset(baseball,
                   id == players[i])

  mba[i] <- with(career, mean(h/ab, na.rm=T))
}
```

i = 1

mba

NA
NA
NA
NA
NA
NA
NA
NA
NA
NA
NA

```
mba <- rep(NA, n)

for (i in 1:n) {
  career <- subset(baseball,
                   id == players[i])


  mba[i] <- with(career, mean(h/ab, na.rm=T))
}
```

i = 1

mba

NA
NA
NA
NA
NA
NA
NA
NA
NA
NA
NA

```
mba <- rep(NA, n)

for (i in 1:n) {
  career <- subset(baseball,
                   id == players[i])


  mba[i] <- with(career, mean(h/ab, na.rm=T))
}
```

i = 1

mba

0.301
NA
NA
NA
NA
NA
NA
NA
NA
NA
NA

```
mba <- rep(NA, n)

for (i in 1:n) {
  career <- subset(baseball,
                   id == players[i])


  mba[i] <- with(career, mean(h/ab, na.rm=T))
}
```

i = 1

mba

0.301
NA
NA
NA
NA
NA
NA
NA
NA
NA
NA

```
mba <- rep(NA, n)

for (i in 1:n) {
  career <- subset(baseball,
                   id == players[i])


  mba[i] <- with(career, mean(h/ab, na.rm=T))
}
```

i = 1

mba

0.301
NA
NA
NA
NA
NA
NA
NA
NA
NA
NA

```
mba <- rep(NA, n)

for (i in 1:n) {
  career <- subset(baseball,
                   id == players[i])


  mba[i] <- with(career, mean(h/ab, na.rm=T))
}
```

i = 2

mba

0.301
NA
NA
NA
NA
NA
NA
NA
NA
NA
NA

```
mba <- rep(NA, n)

for (i in 1:n) {
  career <- subset(baseball,
                   id == players[i])


  mba[i] <- with(career, mean(h/ab, na.rm=T))
}
```

i = 2

mba

0.301
NA
NA
NA
NA
NA
NA
NA
NA
NA
NA

```
mba <- rep(NA, n)

for (i in 1:n) {
  career <- subset(baseball,
                   id == players[i])


  mba[i] <- with(career, mean(h/ab, na.rm=T))
}
```

i = 2

mba

0.301
NA
NA
NA
NA
NA
NA
NA
NA
NA
NA

```
mba <- rep(NA, n)

for (i in 1:n) {
  career <- subset(baseball,
                   id == players[i])


  mba[i] <- with(career, mean(h/ab, na.rm=T))
}
```

i = 2

mba

0.301
NA
NA
NA
NA
NA
NA
NA
NA
NA
NA

```
mba <- rep(NA, n)

for (i in 1:n) {
  career <- subset(baseball,
                   id == players[i])


  mba[i] <- with(career, mean(h/ab, na.rm=T))
}
```

i = 2

mba

0.301
NA
NA
NA
NA
NA
NA
NA
NA
NA
NA

```
mba <- rep(NA, n)

for (i in 1:n) {
  career <- subset(baseball,
                   id == players[i])


  mba[i] <- with(career, mean(h/ab, na.rm=T))
}
```

i = 2

mba

0.301
0.182
NA
NA
NA
NA
NA
NA
NA
NA
NA

```
mba <- rep(NA, n)

for (i in 1:n) {
  career <- subset(baseball,
                   id == players[i])


  mba[i] <- with(career, mean(h/ab, na.rm=T))
}
```

i = 2

... and so on ...

mba

0.301
0.182
NA
NA
NA
NA
NA
NA
NA
NA
NA

```
mba <- rep(NA, n)

for (i in 1:n) {
  career <- subset(baseball,
                   id == players[i])


  mba[i] <- with(career, mean(h/ab, na.rm=T))
}
```

i = 2

... and so on ...

mba

0.301
0.182
0.236
0.210
0.238
0.275
0.089
0.152
0.112
0.249
0.158

Your Turn

- MLB rules for the greatest all-time hitters are that players have to have played at least 1000 games with at least as many at bats, in order to be considered.
- Extend the for loop above to collect the additional information,
  i.e. introduce and collect data for two new variables `games` and `atbats`

How did the Your Turn go?

- What was difficult?

How did the Your Turn go?

- What was difficult?

  - household chores distract from ‘real work’
  - indices are error-prone
  - loops often times result in slow code, because we don’t make use of R’s optimized vector approach

plyr package

- Routines from the plyr package help us to avoid loops
- usage:
  `ddply(.data, .variables, .fun = NULL, ...)`

- Split-apply-combine approach
  i.e. *split* data into subsets on each element of an index set
  apply the same statements for each element
  combine results

Example

```
allstats <- ddply(baseball, .(id), mean)
```

- Separates baseball data into one subset for each player
- Computes the mean for all columns of the subset

Not all columns are numeric

```
head(allstats)
  id     year    stint team lg         g        ab          r          h        X2b       X3b         hr
1 NA 1965.000 1.000000   NA NA 143.39130 537.56522 94.5217391 163.956522 27.1304348 4.2608696 32.8260870
2 NA 1965.235 1.235294   NA NA  40.05882  10.64706  0.7058824   1.470588  0.1764706 0.0000000  0.0000000
3 NA 1964.333 1.133333   NA NA  77.66667 267.93333 25.2000000  68.133333 10.8666667 1.2666667  3.8000000
4 NA 1916.368 1.000000   NA NA  25.36842  53.63158  4.1578947  11.368421  1.6315789 0.7894737  0.1578947
5 NA 1952.667 1.066667   NA NA  85.40000 267.93333 39.4000000  72.133333 12.5333333 3.2666667  2.4666667
6 NA 1958.000 1.000000   NA NA 115.23529 388.58824 48.4117647 107.764706 17.3529412 2.0588235 19.7647059
         rbi          sb       cs         bb        so      ibb        hbp        sh      sf       gidp
1 99.8695652 10.43478261 3.173913 60.9565217 60.130435       NA 1.39130435 0.9130435 5.26087 14.2608696
2  0.5294118  0.00000000 0.000000  0.3529412  4.352941 0.000000 0.11764706 0.8823529 0.00000  0.2352941
3 24.4000000  1.93333333 1.933333 13.8666667 33.266667 2.066667 1.13333333 2.7333333 2.00000  9.9333333
4  3.9473684  0.05263158       NA  2.7894737        NA       NA 0.05263158 1.8421053      NA         NA
5 20.2000000  4.46666667       NA 27.6000000 29.800000       NA 1.13333333 5.2000000      NA  4.1333333
6 66.0000000  1.17647059       NA 34.9411765 62.294118       NA 1.00000000 3.0588235      NA 13.1176471
     season
1 12.000000
2 11.235294
3  7.333333
4 11.368421
5  7.666667
6  9.000000
```

We need to look at the function a bit ...

summarize

- A special function: `summarise` (or `summarize`)

```
summarize(baseball, ab = mean(h/ab, na.rm=T))

summarize(baseball,
  ab = mean(h/ab, na.rm=T),
  games = sum(g, na.rm=T),
  hr=sum(hr, na.rm=T),
  ab = sum(ab, na.rm=T))

summarize(subset(baseball, id=="sosasa01"),
  ab = mean(h/ab, na.rm=T),
  games = sum(g, na.rm=T),
  hr=sum(hr, na.rm=T),
  ab = sum(ab, na.rm=T))
```

ddply + summarise

- Powerful combination to create summary statistics

```
careers <- ddply(baseball, .(id), summarise,
  ba = mean(h/ab, na.rm=T),
  games = sum(g, na.rm=T),
  atbats = sum(ab, na.rm=T)
)
```

```
head(careers)
         id        ba games atbats
1 aaronha01 0.3010752  3298  12364
2 abernte02 0.1824394   681    181
3 adairje01 0.2363071  1165   4019
4 adamsba01 0.2096513   482   1019
5 adamsbo03 0.2378073  1281   4019
6 adcocjo01 0.2751690  1959   6606
```

Your Turn

- Find some summary statistics for each of the teams (variable `team`):
  - how many different (`unique`) players has the team had
  - what was the team's first/last season?
- Challenge:
  find the number of players on each team over time - does the number of players change over time?