had some much needed downtime this weekend, after an exhausting week, along with some self-care – Saturday I had a one-hour deep tissue massage, which left me a little bruised but much more relaxed, and Sunday I spent a few hours in the salon chair having my color touched up, which left me much blonder. Which is why I’m a little late with my Statistics Sunday post, but today, I’m introducing another recently discovered r package: [rpart](https://cran.r-project.org/web/packages/rpart/index.html). Short for “recursive partitioning,” this package creates decision trees for classification, regression, and survival analyses. Today, I’m going to demonstrate using the rpart package for visualizing regression.

To demonstrate this technique, I’m using my 2017 reading dataset. A reader requested I make this dataset available, which I have done – you can download it [here](https://www.dropbox.com/s/rt5q7t5rk5z9nr2/2017_books.csv?dl=1). [This post](http://www.deeplytrivial.com/2018/01/data-analysis-of-my-2017-reading.html) describes the data in more detail, but the short description is that this dataset contains the 53 books I read last year, with information on book genre, page length, how long it took me to read it, and two ratings: my own rating and the average rating on Goodreads. Look for another dataset soon, containing my 2018 reading data; I made a goal of 60 books and I’ve already read 50, meaning lots more data than last year. I’m thinking of going for broke and bumping my reading goal up to 100, because apparently 60 books is not enough of a challenge for me now that I spend so much downtime reading.

First, I’ll load my dataset, then conduct the basic linear model I demonstrated in the post linked above.

setwd("~/R")  
library(tidyverse)

## Warning: Duplicated column names deduplicated: 'Author' => 'Author\_1' [13]

## Parsed with column specification:  
## cols(  
## .default = col\_integer(),  
## Title = col\_character(),  
## Author = col\_character(),  
## G\_Rating = col\_double(),  
## Started = col\_character(),  
## Finished = col\_character()  
## )

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

colnames(books)[13] <- "Author\_Gender"  
myrating<-lm(My\_Rating ~ Pages + Read\_Time + Author\_Gender + Fiction + Fantasy + Math\_Stats + YA\_Fic, data=books)  
summary(myrating)

##   
## Call:  
## lm(formula = My\_Rating ~ Pages + Read\_Time + Author\_Gender +   
## Fiction + Fantasy + Math\_Stats + YA\_Fic, data = books)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.73120 -0.34382 -0.00461 0.24665 1.49932   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 3.5861211 0.2683464 13.364 <2e-16 \*\*\*  
## Pages 0.0019578 0.0007435 2.633 0.0116 \*   
## Read\_Time -0.0244168 0.0204186 -1.196 0.2380   
## Author\_Gender -0.1285178 0.1666207 -0.771 0.4445   
## Fiction 0.1052319 0.2202581 0.478 0.6351   
## Fantasy 0.5234710 0.2097386 2.496 0.0163 \*   
## Math\_Stats -0.2558926 0.2122238 -1.206 0.2342   
## YA\_Fic -0.7330553 0.2684623 -2.731 0.0090 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.4952 on 45 degrees of freedom  
## Multiple R-squared: 0.4624, Adjusted R-squared: 0.3788   
## F-statistic: 5.53 on 7 and 45 DF, p-value: 0.0001233

These analyses show that I give higher ratings for books that are longer and Fantasy genre, and lower ratings to books that are Young Adult Fiction. Now let’s see what happens if I run this same linear model through rpart. Note that this is a slightly different technique, looking for cuts that differentiate outcomes, so it will find slightly different results.

library(rpart)  
tree1 <- rpart(My\_Rating ~ Pages + Read\_Time + Author\_Gender + Fiction + Fantasy + Math\_Stats + YA\_Fic, method = "anova", data=books)  
printcp(tree1)

##   
## Regression tree:  
## rpart(formula = My\_Rating ~ Pages + Read\_Time + Author\_Gender +   
## Fiction + Fantasy + Math\_Stats + YA\_Fic, data = books, method = "anova")  
##   
## Variables actually used in tree construction:  
## [1] Fantasy Math\_Stats Pages   
##   
## Root node error: 20.528/53 = 0.38733  
##   
## n= 53   
##   
## CP nsplit rel error xerror xstd  
## 1 0.305836 0 1.00000 1.03609 0.17531  
## 2 0.092743 1 0.69416 0.76907 0.12258  
## 3 0.022539 2 0.60142 0.71698 0.11053  
## 4 0.010000 3 0.57888 0.74908 0.11644

These results different somewhat. Pages is still a significant variable, as is Fantasy. But now Math\_Stats (indicating books that are about mathematics or statistics, one of my top genres from last year) also is. These are the variables used by the analysis to construct my regression tree. If we look at the full summary –

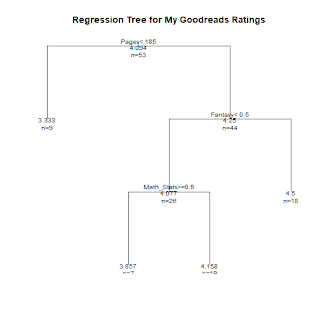
summary(tree1)

## Call:  
## rpart(formula = My\_Rating ~ Pages + Read\_Time + Author\_Gender +   
## Fiction + Fantasy + Math\_Stats + YA\_Fic, data = books, method = "anova")  
## n= 53   
##   
## CP nsplit rel error xerror xstd  
## 1 0.30583640 0 1.0000000 1.0360856 0.1753070  
## 2 0.09274251 1 0.6941636 0.7690729 0.1225752  
## 3 0.02253938 2 0.6014211 0.7169813 0.1105294  
## 4 0.01000000 3 0.5788817 0.7490758 0.1164386  
##   
## Variable importance  
## Pages Fantasy Fiction YA\_Fic Math\_Stats Read\_Time   
## 62 18 8 6 4 3   
##   
## Node number 1: 53 observations, complexity param=0.3058364  
## mean=4.09434, MSE=0.3873265   
## left son=2 (9 obs) right son=3 (44 obs)  
## Primary splits:  
## Pages < 185 to the left, improve=0.30583640, (0 missing)  
## Fiction < 0.5 to the left, improve=0.24974560, (0 missing)  
## Fantasy < 0.5 to the left, improve=0.20761810, (0 missing)  
## Math\_Stats < 0.5 to the right, improve=0.20371790, (0 missing)  
## Author\_Gender < 0.5 to the right, improve=0.02705187, (0 missing)  
##   
## Node number 2: 9 observations  
## mean=3.333333, MSE=0.2222222   
##   
## Node number 3: 44 observations, complexity param=0.09274251  
## mean=4.25, MSE=0.2784091   
## left son=6 (26 obs) right son=7 (18 obs)  
## Primary splits:  
## Fantasy < 0.5 to the left, improve=0.15541600, (0 missing)  
## Fiction < 0.5 to the left, improve=0.12827990, (0 missing)  
## Math\_Stats < 0.5 to the right, improve=0.10487750, (0 missing)  
## Pages < 391 to the left, improve=0.05344995, (0 missing)  
## Read\_Time < 7.5 to the right, improve=0.04512078, (0 missing)  
## Surrogate splits:  
## Fiction < 0.5 to the left, agree=0.773, adj=0.444, (0 split)  
## YA\_Fic < 0.5 to the left, agree=0.727, adj=0.333, (0 split)  
## Pages < 370 to the left, agree=0.682, adj=0.222, (0 split)  
## Read\_Time < 3.5 to the right, agree=0.659, adj=0.167, (0 split)  
##   
## Node number 6: 26 observations, complexity param=0.02253938  
## mean=4.076923, MSE=0.2248521   
## left son=12 (7 obs) right son=13 (19 obs)  
## Primary splits:  
## Math\_Stats < 0.5 to the right, improve=0.079145230, (0 missing)  
## Pages < 364 to the left, improve=0.042105260, (0 missing)  
## Fiction < 0.5 to the left, improve=0.042105260, (0 missing)  
## Read\_Time < 5.5 to the left, improve=0.016447370, (0 missing)  
## Author\_Gender < 0.5 to the right, improve=0.001480263, (0 missing)  
##   
## Node number 7: 18 observations  
## mean=4.5, MSE=0.25   
##   
## Node number 12: 7 observations  
## mean=3.857143, MSE=0.4081633   
##   
## Node number 13: 19 observations  
## mean=4.157895, MSE=0.132964

we see that Fiction, YA\_Fic, and Read\_Time were also significant variables. The problem is that there is multicollinearity between Fiction, Fantasy, Math\_Stats, and YA\_Fic. All Fantasy and YA\_Fic books are Fiction, while all Math\_Stats books are not Fiction. And all YA\_Fic books I read were Fantasy. This is probably why the tree didn’t use Fiction or YA\_Fic. I’m not completely clear on why Read\_Time didn’t end up in the regression tree, but it may be because my read time was pretty constant among the different splits and didn’t add any new information to the tree. If I were presenting these results somewhere other than my blog, I’d probably want to do some follow-up analyses to confirm this fact.

Now the fun part: let’s plot our regression tree:

plot(tree1, uniform = TRUE, main = "Regression Tree for My Goodreads Ratings")  
text(tree1, use.n = TRUE, all = TRUE, cex = 0.8)

[](https://i2.wp.com/1.bp.blogspot.com/-0yqOWxHEGPQ/W4QedVlw5NI/AAAAAAAANro/Pp-oSMVe5gcbi3u2-bMyI1AwZZ1J1XzkACLcBGAs/s1600/unnamed-chunk-4-1.png?ssl=1)

This tree shows that, before taking into account anything, my average book rating was 4.09. If a book is shorter than 185 pages (9 books in my dataset), it’s average rating was 3.33. For longer books (44), the average rating was 4.25. But there’s more to it than that. For non-Fantasy books (26), the average was 4.08, while the Fantasy (18 books) average was 4.5. If the book was Math\_Stats (7), I gave it an average of 3.86, and if it was not Math\_Stats (19), the average was 4.16. (Unfortunately, R cuts off the bottom part of the plot.)

While this plot is great for a quick visualization, I can make a nicer looking plot (which doesn’t cut off the bottom text) as a PostScript file.

post(tree1, file = "mytree.ps",  
 title = "Regression Tree for Rating")