Untitled

library(ggplot2)

## Warning: package 'ggplot2' was built under R version 3.6.2

library(dplyr)

##   
## 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

library(car)

## Loading required package: carData

##   
## Attaching package: 'car'

## The following object is masked from 'package:dplyr':  
##   
## recode

library(leaps)  
library(rms)

## Loading required package: Hmisc

## Loading required package: lattice

## Loading required package: survival

## Loading required package: Formula

##   
## Attaching package: 'Hmisc'

## The following objects are masked from 'package:dplyr':  
##   
## src, summarize

## The following objects are masked from 'package:base':  
##   
## format.pval, units

## Loading required package: SparseM

##   
## Attaching package: 'SparseM'

## The following object is masked from 'package:base':  
##   
## backsolve

##   
## Attaching package: 'rms'

## The following objects are masked from 'package:car':  
##   
## Predict, vif

library(ggfortify)

## Warning: package 'ggfortify' was built under R version 3.6.2

library(GGally)

## Warning: package 'GGally' was built under R version 3.6.2

## Registered S3 method overwritten by 'GGally':  
## method from   
## +.gg ggplot2

##   
## Attaching package: 'GGally'

## The following object is masked from 'package:dplyr':  
##   
## nasa

library(ggcorrplot)  
library(sandwich)

## Warning: package 'sandwich' was built under R version 3.6.2

library(lmtest)

## Warning: package 'lmtest' was built under R version 3.6.2

## Loading required package: zoo

## Warning: package 'zoo' was built under R version 3.6.2

##   
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':  
##   
## as.Date, as.Date.numeric

##   
## Attaching package: 'lmtest'

## The following object is masked from 'package:rms':  
##   
## lrtest

setwd("~/Desktop/Denison/DA/DA 220/Final Project")  
trails <- read.csv("AllTrails data - nationalpark.csv")  
t1 <- read.csv("All National Parks Visitation 1904-2016.csv")

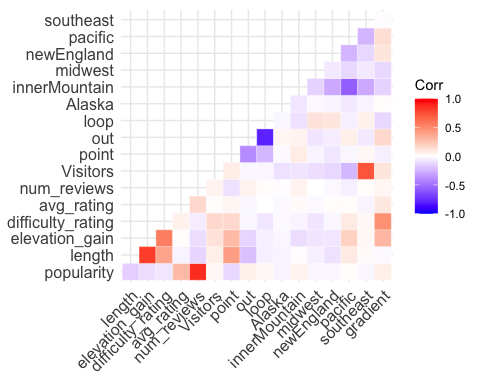
The goal of this dataset is to show what aspects of different trail lead to them having higher popularity among its users. The reason I’m exploring this is because the outdoors are more important than even in a globabl pandemic. Being outside have been linked to higher vitamin D levels, more exercise which improves mood, greater happiness, better concentration, and most importantly, a boosted immune system. If people are able to find what makes an outdoors spot popular, they may be able to better take advantage of these benefits linked to being outside

t1 <- t1 %>% filter(YearRaw == 2016)  
t1 <- t1 %>% rename("area\_name" = Unit.Name)  
t2 <- merge(trails, t1, by = "area\_name")  
trails1 <- t2 %>%  
 select(area\_name, popularity:num\_reviews, Region, Visitors)  
  
trails2 <- trails1 %>%  
 mutate(point = ifelse(route\_type == "point to point", 1, 0),   
 out = ifelse(route\_type == "out and back", 1, 0),   
 loop = ifelse(route\_type == "loop", 1, 0),   
 Alaska = ifelse(Region == "AK", 1, 0),   
 innerMountain = ifelse(Region == "IM", 1, 0),   
 midwest = ifelse(Region == "MW", 1, 0),   
 newEngland = ifelse(Region == "NE", 1, 0),   
 pacific = ifelse(Region == "PW", 1, 0),   
 southeast = ifelse(Region == "SE", 1, 0),   
 gradient = atan2(elevation\_gain, length))  
  
  
train<-sample\_frac(trails2, 0.6)  
sid<-as.numeric(rownames(train))  
test<-trails2[-sid,]  
  
str(train)

## 'data.frame': 1943 obs. of 21 variables:  
## $ area\_name : Factor w/ 60 levels "Acadia National Park",..: 42 40 58 32 1 37 18 58 53 16 ...  
## $ popularity : num 3.87 2.47 10.92 33.31 6.41 ...  
## $ length : num 9656 35084 10300 5472 6437 ...  
## $ elevation\_gain : num 118 1722 242 250 290 ...  
## $ difficulty\_rating: int 1 5 1 3 3 3 5 3 5 3 ...  
## $ route\_type : Factor w/ 3 levels "loop","out and back",..: 2 2 2 2 1 2 3 2 2 1 ...  
## $ visitor\_usage : int NA 3 3 4 2 NA 2 1 2 3 ...  
## $ avg\_rating : num 3.5 4.5 4.5 5 4.5 0 5 4 4 4.5 ...  
## $ num\_reviews : int 8 2 42 697 20 0 5 10 20 158 ...  
## $ Region : Factor w/ 8 levels "AK","IM","MW",..: 8 7 2 7 5 7 1 2 5 3 ...  
## $ Visitors : int 586514 607479 4257177 1263558 3303393 2505286 587412 4257177 1437341 2423390 ...  
## $ point : num 0 0 0 0 0 0 1 0 0 0 ...  
## $ out : num 1 1 1 1 0 1 0 1 1 0 ...  
## $ loop : num 0 0 0 0 1 0 0 0 0 1 ...  
## $ Alaska : num 0 0 0 0 0 0 1 0 0 0 ...  
## $ innerMountain : num 0 0 1 0 0 0 0 1 0 0 ...  
## $ midwest : num 0 0 0 0 0 0 0 0 0 1 ...  
## $ newEngland : num 0 0 0 0 1 0 0 0 1 0 ...  
## $ pacific : num 0 1 0 1 0 1 0 0 0 0 ...  
## $ southeast : num 1 0 0 0 0 0 0 0 0 0 ...  
## $ gradient : num 0.0122 0.049 0.0235 0.0456 0.045 ...

We first need to do some data cleaning. We first need to rename the Unit.name variable in the second dataset to be able to merge with the first dataset. After this, we’re able to select out the important numeric and factor variables that we need for our linear regression. We also need to change our factor variables into 0 1 variables to be run through the best subsets command. We also want to add a gradient variable which describes how much elevation gain that a trail has for its length. Finally, we can break up our data into training and testing sets using the sample\_frac() command.

train1 <- train %>%  
 select(-area\_name, -route\_type, -Region, -visitor\_usage)  
  
ggcorrplot(cor(train1), method = NULL, type = "lower", outline.color = "white")

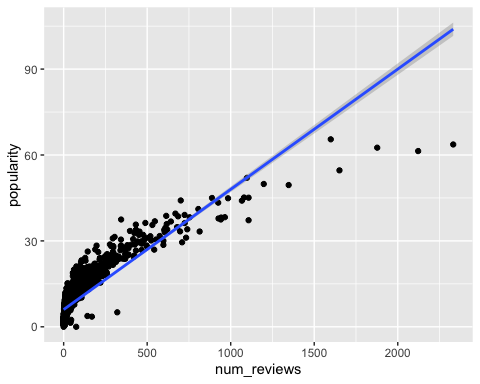


train1 <- train1 %>% select(-length, -southeast, -loop)

This correlation matrix plot indicates that length and elevation gain, Visitors and southeast, and out and loop are highly correlated. We want to remove one of each of these pairs as to not lose significance when it comes to building our model out of these variables.

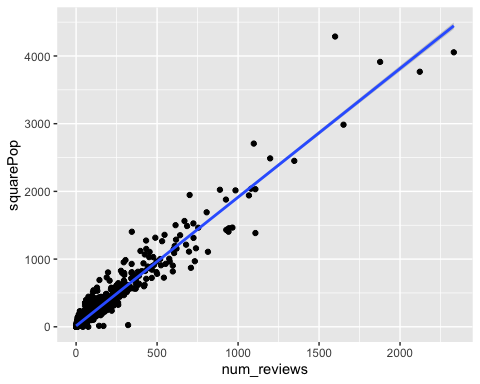
train3 <- train1 %>%  
 mutate(squarePop = popularity^2) %>%  
 select(-popularity)  
  
train1 %>%  
 ggplot(aes(num\_reviews, popularity)) +  
 geom\_point() +   
 stat\_smooth(method = "lm")

## `geom\_smooth()` using formula 'y ~ x'



train3 %>%  
 ggplot(aes(num\_reviews, squarePop)) +  
 geom\_point() +   
 stat\_smooth(method = "lm")

## `geom\_smooth()` using formula 'y ~ x'



This initial plot indicated that the data is roughly a sqrt(x) kind of plot. In order to linearize this data, we must take the square root of the y axis. This can be seen by our second plot with popularity squared which looks much more linear that the first plot.

best.subset <- regsubsets(popularity~.,train1,nvmax = 15)  
sum <- summary(best.subset)  
sum$outmat

## elevation\_gain difficulty\_rating avg\_rating num\_reviews Visitors  
## 1 ( 1 ) " " " " " " "\*" " "   
## 2 ( 1 ) " " " " "\*" "\*" " "   
## 3 ( 1 ) " " " " "\*" "\*" " "   
## 4 ( 1 ) " " " " "\*" "\*" " "   
## 5 ( 1 ) " " " " "\*" "\*" " "   
## 6 ( 1 ) " " "\*" "\*" "\*" " "   
## 7 ( 1 ) " " "\*" "\*" "\*" " "   
## 8 ( 1 ) " " "\*" "\*" "\*" " "   
## 9 ( 1 ) " " "\*" "\*" "\*" "\*"   
## 10 ( 1 ) " " "\*" "\*" "\*" "\*"   
## 11 ( 1 ) " " "\*" "\*" "\*" "\*"   
## 12 ( 1 ) " " "\*" "\*" "\*" "\*"   
## 13 ( 1 ) "\*" "\*" "\*" "\*" "\*"   
## point out Alaska innerMountain midwest newEngland pacific  
## 1 ( 1 ) " " " " " " " " " " " " " "   
## 2 ( 1 ) " " " " " " " " " " " " " "   
## 3 ( 1 ) "\*" " " " " " " " " " " " "   
## 4 ( 1 ) "\*" " " " " " " " " " " "\*"   
## 5 ( 1 ) "\*" " " " " "\*" " " " " "\*"   
## 6 ( 1 ) "\*" " " " " "\*" " " " " "\*"   
## 7 ( 1 ) "\*" " " " " "\*" " " " " "\*"   
## 8 ( 1 ) "\*" "\*" " " "\*" " " " " "\*"   
## 9 ( 1 ) "\*" "\*" " " "\*" " " " " "\*"   
## 10 ( 1 ) "\*" "\*" " " "\*" " " "\*" "\*"   
## 11 ( 1 ) "\*" "\*" "\*" "\*" " " "\*" "\*"   
## 12 ( 1 ) "\*" "\*" "\*" "\*" "\*" "\*" "\*"   
## 13 ( 1 ) "\*" "\*" "\*" "\*" "\*" "\*" "\*"   
## gradient  
## 1 ( 1 ) " "   
## 2 ( 1 ) " "   
## 3 ( 1 ) " "   
## 4 ( 1 ) " "   
## 5 ( 1 ) " "   
## 6 ( 1 ) " "   
## 7 ( 1 ) "\*"   
## 8 ( 1 ) "\*"   
## 9 ( 1 ) "\*"   
## 10 ( 1 ) "\*"   
## 11 ( 1 ) "\*"   
## 12 ( 1 ) "\*"   
## 13 ( 1 ) "\*"

reg <- lm(popularity~num\_reviews+avg\_rating+difficulty\_rating+gradient+point+innerMountain+pacific, data = train1)  
summary(reg)

##   
## Call:  
## lm(formula = popularity ~ num\_reviews + avg\_rating + difficulty\_rating +   
## gradient + point + innerMountain + pacific, data = train1)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -37.987 -2.265 -0.310 1.971 15.316   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.0546793 0.4109600 0.133 0.89417   
## num\_reviews 0.0403733 0.0004961 81.385 < 2e-16 \*\*\*  
## avg\_rating 1.3752212 0.0878392 15.656 < 2e-16 \*\*\*  
## difficulty\_rating -0.2432505 0.0554638 -4.386 1.22e-05 \*\*\*  
## gradient 13.1433862 4.1009865 3.205 0.00137 \*\*   
## point -1.2834939 0.2680791 -4.788 1.81e-06 \*\*\*  
## innerMountain 0.8177618 0.2023119 4.042 5.51e-05 \*\*\*  
## pacific 1.2851543 0.2095103 6.134 1.04e-09 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 3.544 on 1935 degrees of freedom  
## Multiple R-squared: 0.8021, Adjusted R-squared: 0.8014   
## F-statistic: 1120 on 7 and 1935 DF, p-value: < 2.2e-16

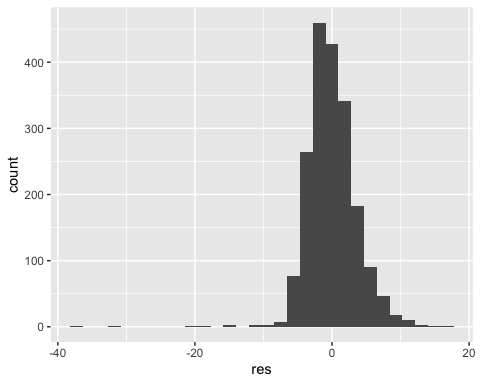
reg1 <- lm(popularity~num\_reviews+avg\_rating+difficulty\_rating+point+innerMountain+pacific, data = train1)  
summary(reg1)

##   
## Call:  
## lm(formula = popularity ~ num\_reviews + avg\_rating + difficulty\_rating +   
## point + innerMountain + pacific, data = train1)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -38.188 -2.210 -0.315 2.012 15.910   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.1932507 0.4096566 0.472 0.637167   
## num\_reviews 0.0404802 0.0004961 81.591 < 2e-16 \*\*\*  
## avg\_rating 1.3972559 0.0877791 15.918 < 2e-16 \*\*\*  
## difficulty\_rating -0.1559011 0.0484210 -3.220 0.001305 \*\*   
## point -1.3968527 0.2663710 -5.244 1.74e-07 \*\*\*  
## innerMountain 0.7508073 0.2017116 3.722 0.000203 \*\*\*  
## pacific 1.3066378 0.2099039 6.225 5.89e-10 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 3.553 on 1936 degrees of freedom  
## Multiple R-squared: 0.801, Adjusted R-squared: 0.8004   
## F-statistic: 1299 on 6 and 1936 DF, p-value: < 2.2e-16

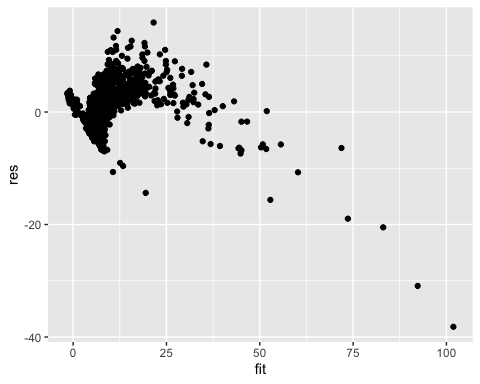
As we saw from the previous scatterplots, the square of popularity better linearizes the data. However, we will still run a regression with just normal popularity. Best subsets indicates which model is the best to run and we can see that it produces a pretty good model. All but one of the variables are very statistically significant. We also have an R-squared of 0.7932 which tells us that 79.32% of the variation in popularity can be accounted for by the x variables that we chose. We then run a robust standard error output which we can see doesn’t really change too much. This even with the fact that the standard errors have increased.

train5 <- train1 %>% mutate(res = resid(reg1)) %>% mutate(fit = fitted(reg1))  
  
train5 %>%  
 ggplot(aes(res))+  
 geom\_histogram()

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



train5 %>%  
 ggplot(aes(fit,res))+  
 geom\_point()



Our residual histogram looks good and roughly normally distributed. However, the residual scatterplot looks very heteroscedastic and quadratic, again indicating that we may want to square our y-variable, popularity.

best.subset <- regsubsets(squarePop~.,train3,nvmax = 13)  
sum <- summary(best.subset)  
sum$outmat

## elevation\_gain difficulty\_rating avg\_rating num\_reviews Visitors  
## 1 ( 1 ) " " " " " " "\*" " "   
## 2 ( 1 ) "\*" " " " " "\*" " "   
## 3 ( 1 ) " " " " " " "\*" " "   
## 4 ( 1 ) "\*" " " " " "\*" " "   
## 5 ( 1 ) "\*" " " "\*" "\*" " "   
## 6 ( 1 ) "\*" " " "\*" "\*" " "   
## 7 ( 1 ) "\*" " " "\*" "\*" " "   
## 8 ( 1 ) "\*" " " "\*" "\*" "\*"   
## 9 ( 1 ) "\*" " " "\*" "\*" "\*"   
## 10 ( 1 ) "\*" " " "\*" "\*" "\*"   
## 11 ( 1 ) "\*" " " "\*" "\*" "\*"   
## 12 ( 1 ) "\*" "\*" "\*" "\*" "\*"   
## 13 ( 1 ) "\*" "\*" "\*" "\*" "\*"   
## point out Alaska innerMountain midwest newEngland pacific  
## 1 ( 1 ) " " " " " " " " " " " " " "   
## 2 ( 1 ) " " " " " " " " " " " " " "   
## 3 ( 1 ) " " " " " " "\*" " " " " "\*"   
## 4 ( 1 ) " " " " " " "\*" " " " " "\*"   
## 5 ( 1 ) " " " " " " "\*" " " " " "\*"   
## 6 ( 1 ) " " " " " " "\*" "\*" " " "\*"   
## 7 ( 1 ) "\*" " " " " "\*" "\*" " " "\*"   
## 8 ( 1 ) "\*" " " " " "\*" " " "\*" "\*"   
## 9 ( 1 ) "\*" " " " " "\*" "\*" "\*" "\*"   
## 10 ( 1 ) "\*" " " "\*" "\*" "\*" "\*" "\*"   
## 11 ( 1 ) "\*" " " "\*" "\*" "\*" "\*" "\*"   
## 12 ( 1 ) "\*" " " "\*" "\*" "\*" "\*" "\*"   
## 13 ( 1 ) "\*" "\*" "\*" "\*" "\*" "\*" "\*"   
## gradient  
## 1 ( 1 ) " "   
## 2 ( 1 ) " "   
## 3 ( 1 ) " "   
## 4 ( 1 ) " "   
## 5 ( 1 ) " "   
## 6 ( 1 ) " "   
## 7 ( 1 ) " "   
## 8 ( 1 ) " "   
## 9 ( 1 ) " "   
## 10 ( 1 ) " "   
## 11 ( 1 ) "\*"   
## 12 ( 1 ) "\*"   
## 13 ( 1 ) "\*"

null <- lm(squarePop~1, train3)  
full <- lm(squarePop~., train3)  
step(null, scope=list(lower=null, upper=full),direction="forward")

## Start: AIC=22480.23  
## squarePop ~ 1  
##   
## Df Sum of Sq RSS AIC  
## + num\_reviews 1 193247716 12223221 16999  
## + avg\_rating 1 6369623 199101314 22421  
## + point 1 1741721 203729216 22466  
## + elevation\_gain 1 1179731 204291206 22471  
## + innerMountain 1 782166 204688771 22475  
## + out 1 708581 204762356 22476  
## + gradient 1 695879 204775058 22476  
## + Visitors 1 470073 205000864 22478  
## + difficulty\_rating 1 287064 205183873 22480  
## + newEngland 1 283506 205187431 22480  
## <none> 205470937 22480  
## + midwest 1 179544 205291393 22480  
## + Alaska 1 76008 205394929 22482  
## + pacific 1 52656 205418281 22482  
##   
## Step: AIC=16999.15  
## squarePop ~ num\_reviews  
##   
## Df Sum of Sq RSS AIC  
## + elevation\_gain 1 215690 12007531 16967  
## + pacific 1 212124 12011098 16967  
## + midwest 1 208469 12014753 16968  
## + avg\_rating 1 133314 12089908 16980  
## + difficulty\_rating 1 88180 12135041 16987  
## + newEngland 1 65686 12157535 16991  
## + gradient 1 51421 12171801 16993  
## + Alaska 1 32917 12190304 16996  
## + innerMountain 1 18466 12204755 16998  
## <none> 12223221 16999  
## + out 1 6011 12217210 17000  
## + Visitors 1 2849 12220373 17001  
## + point 1 1112 12222110 17001  
##   
## Step: AIC=16966.56  
## squarePop ~ num\_reviews + elevation\_gain  
##   
## Df Sum of Sq RSS AIC  
## + midwest 1 172989 11834542 16940  
## + pacific 1 144915 11862616 16945  
## + avg\_rating 1 123381 11884150 16948  
## + newEngland 1 48375 11959156 16961  
## + innerMountain 1 34281 11973250 16963  
## + Alaska 1 27954 11979577 16964  
## + out 1 19480 11988051 16965  
## + Visitors 1 12519 11995012 16966  
## <none> 12007531 16967  
## + point 1 10892 11996639 16967  
## + gradient 1 6820 12000711 16968  
## + difficulty\_rating 1 1990 12005541 16968  
##   
## Step: AIC=16940.36  
## squarePop ~ num\_reviews + elevation\_gain + midwest  
##   
## Df Sum of Sq RSS AIC  
## + avg\_rating 1 118707 11715835 16923  
## + pacific 1 111383 11723159 16924  
## + newEngland 1 63423 11771119 16932  
## + Alaska 1 31139 11803403 16937  
## + Visitors 1 24455 11810086 16938  
## + innerMountain 1 13892 11820650 16940  
## <none> 11834542 16940  
## + point 1 11995 11822547 16940  
## + out 1 9720 11824822 16941  
## + gradient 1 1396 11833145 16942  
## + difficulty\_rating 1 633 11833909 16942  
##   
## Step: AIC=16922.77  
## squarePop ~ num\_reviews + elevation\_gain + midwest + avg\_rating  
##   
## Df Sum of Sq RSS AIC  
## + pacific 1 109321 11606514 16907  
## + newEngland 1 66629 11649206 16914  
## + Alaska 1 29284 11686551 16920  
## + Visitors 1 24086 11691749 16921  
## + point 1 15897 11699938 16922  
## + innerMountain 1 12965 11702870 16923  
## + out 1 12150 11703685 16923  
## <none> 11715835 16923  
## + gradient 1 41 11715794 16925  
## + difficulty\_rating 1 5 11715830 16925  
##   
## Step: AIC=16906.56  
## squarePop ~ num\_reviews + elevation\_gain + midwest + avg\_rating +   
## pacific  
##   
## Df Sum of Sq RSS AIC  
## + innerMountain 1 151603 11454911 16883  
## + newEngland 1 32516 11573998 16903  
## + Alaska 1 21946 11584568 16905  
## <none> 11606514 16907  
## + point 1 9087 11597427 16907  
## + out 1 6759 11599755 16907  
## + Visitors 1 3892 11602622 16908  
## + gradient 1 345 11606169 16908  
## + difficulty\_rating 1 166 11606349 16908  
##   
## Step: AIC=16883.01  
## squarePop ~ num\_reviews + elevation\_gain + midwest + avg\_rating +   
## pacific + innerMountain  
##   
## Df Sum of Sq RSS AIC  
## + point 1 15802.9 11439108 16882  
## <none> 11454911 16883  
## + Alaska 1 7074.8 11447836 16884  
## + Visitors 1 6367.8 11448543 16884  
## + newEngland 1 4936.2 11449975 16884  
## + out 1 2544.8 11452366 16885  
## + gradient 1 1149.4 11453761 16885  
## + difficulty\_rating 1 205.5 11454705 16885  
##   
## Step: AIC=16882.33  
## squarePop ~ num\_reviews + elevation\_gain + midwest + avg\_rating +   
## pacific + innerMountain + point  
##   
## Df Sum of Sq RSS AIC  
## <none> 11439108 16882  
## + Visitors 1 7672.8 11431435 16883  
## + Alaska 1 7174.8 11431933 16883  
## + newEngland 1 4095.7 11435012 16884  
## + gradient 1 247.1 11438861 16884  
## + difficulty\_rating 1 149.1 11438959 16884  
## + out 1 6.6 11439101 16884

##   
## Call:  
## lm(formula = squarePop ~ num\_reviews + elevation\_gain + midwest +   
## avg\_rating + pacific + innerMountain + point, data = train3)  
##   
## Coefficients:  
## (Intercept) num\_reviews elevation\_gain midwest   
## -45.44201 1.89841 0.01039 -31.78801   
## avg\_rating pacific innerMountain point   
## 8.35285 30.53430 23.56231 -9.75942

reg <- lm(formula = squarePop ~ num\_reviews + pacific + innerMountain + avg\_rating + elevation\_gain + Visitors + newEngland, data = train3)  
summary(reg)

##   
## Call:  
## lm(formula = squarePop ~ num\_reviews + pacific + innerMountain +   
## avg\_rating + elevation\_gain + Visitors + newEngland, data = train3)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -752.86 -21.73 -6.52 13.09 1224.91   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -6.805e+01 9.999e+00 -6.805 1.34e-11 \*\*\*  
## num\_reviews 1.898e+00 1.075e-02 176.594 < 2e-16 \*\*\*  
## pacific 5.045e+01 6.449e+00 7.822 8.46e-15 \*\*\*  
## innerMountain 4.100e+01 5.959e+00 6.879 8.09e-12 \*\*\*  
## avg\_rating 7.927e+00 1.897e+00 4.179 3.06e-05 \*\*\*  
## elevation\_gain 8.825e-03 2.053e-03 4.298 1.81e-05 \*\*\*  
## Visitors 2.085e-06 7.098e-07 2.937 0.00336 \*\*   
## newEngland 2.293e+01 7.630e+00 3.005 0.00269 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 76.9 on 1935 degrees of freedom  
## Multiple R-squared: 0.9443, Adjusted R-squared: 0.9441   
## F-statistic: 4688 on 7 and 1935 DF, p-value: < 2.2e-16

reg1 <- lm(formula = squarePop ~ num\_reviews + pacific + innerMountain + elevation\_gain + avg\_rating + midwest, data = train3)  
summary(reg1)

##   
## Call:  
## lm(formula = squarePop ~ num\_reviews + pacific + innerMountain +   
## elevation\_gain + avg\_rating + midwest, data = train3)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -748.76 -21.44 -6.01 12.27 1225.52   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -45.145836 8.621259 -5.237 1.81e-07 \*\*\*  
## num\_reviews 1.899819 0.010726 177.118 < 2e-16 \*\*\*  
## pacific 30.765810 4.752492 6.474 1.21e-10 \*\*\*  
## innerMountain 22.988532 4.541507 5.062 4.54e-07 \*\*\*  
## elevation\_gain 0.009361 0.002021 4.633 3.85e-06 \*\*\*  
## avg\_rating 8.206515 1.894713 4.331 1.56e-05 \*\*\*  
## midwest -31.788944 10.089431 -3.151 0.00165 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 76.92 on 1936 degrees of freedom  
## Multiple R-squared: 0.9443, Adjusted R-squared: 0.9441   
## F-statistic: 5465 on 6 and 1936 DF, p-value: < 2.2e-16

reg2 <- lm(squarePop ~ elevation\_gain + num\_reviews + pacific + avg\_rating + midwest + Visitors:newEngland, data=train3)  
summary(reg2)

##   
## Call:  
## lm(formula = squarePop ~ elevation\_gain + num\_reviews + pacific +   
## avg\_rating + midwest + Visitors:newEngland, data = train3)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -749.67 -20.36 -6.21 11.33 1230.87   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -2.981e+01 8.259e+00 -3.609 0.000315 \*\*\*  
## elevation\_gain 8.837e-03 2.035e-03 4.342 1.49e-05 \*\*\*  
## num\_reviews 1.900e+00 1.079e-02 176.180 < 2e-16 \*\*\*  
## pacific 1.412e+01 3.930e+00 3.594 0.000334 \*\*\*  
## avg\_rating 8.626e+00 1.906e+00 4.527 6.36e-06 \*\*\*  
## midwest -4.874e+01 9.770e+00 -4.989 6.62e-07 \*\*\*  
## Visitors:newEngland -5.614e-06 2.255e-06 -2.489 0.012892 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 77.3 on 1936 degrees of freedom  
## Multiple R-squared: 0.9437, Adjusted R-squared: 0.9435   
## F-statistic: 5408 on 6 and 1936 DF, p-value: < 2.2e-16

coeftest(reg, vcov = vcovHC(reg, type = "HC1"))

##   
## t test of coefficients:  
##   
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -6.8047e+01 6.5574e+00 -10.3771 < 2.2e-16 \*\*\*  
## num\_reviews 1.8984e+00 5.2286e-02 36.3082 < 2.2e-16 \*\*\*  
## pacific 5.0449e+01 5.8287e+00 8.6554 < 2.2e-16 \*\*\*  
## innerMountain 4.0996e+01 5.2718e+00 7.7766 1.202e-14 \*\*\*  
## avg\_rating 7.9267e+00 1.0499e+00 7.5499 6.671e-14 \*\*\*  
## elevation\_gain 8.8254e-03 2.5668e-03 3.4383 0.0005977 \*\*\*  
## Visitors 2.0845e-06 5.9687e-07 3.4924 0.0004894 \*\*\*  
## newEngland 2.2929e+01 5.4592e+00 4.2001 2.790e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

coeftest(reg1,vcov = vcovHC(reg1, type = "HC1"))

##   
## t test of coefficients:  
##   
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -45.1458358 4.6037042 -9.8064 < 2.2e-16 \*\*\*  
## num\_reviews 1.8998186 0.0523217 36.3103 < 2.2e-16 \*\*\*  
## pacific 30.7658104 3.8915966 7.9057 4.435e-15 \*\*\*  
## innerMountain 22.9885323 3.6693455 6.2650 4.581e-10 \*\*\*  
## elevation\_gain 0.0093614 0.0024650 3.7978 0.0001505 \*\*\*  
## avg\_rating 8.2065154 1.0490603 7.8227 8.430e-15 \*\*\*  
## midwest -31.7889444 9.0825630 -3.5000 0.0004758 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

coeftest(reg2,vcov = vcovHC(reg2, type = "HC1"))

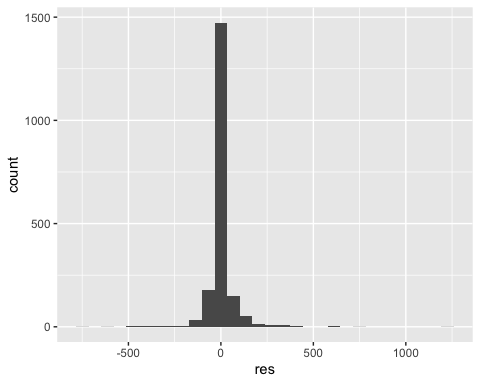
##   
## t test of coefficients:  
##   
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -2.9811e+01 4.0050e+00 -7.4436 1.466e-13 \*\*\*  
## elevation\_gain 8.8373e-03 2.5039e-03 3.5294 0.0004263 \*\*\*  
## num\_reviews 1.9001e+00 5.2678e-02 36.0708 < 2.2e-16 \*\*\*  
## pacific 1.4123e+01 3.9003e+00 3.6211 0.0003009 \*\*\*  
## avg\_rating 8.6262e+00 1.0520e+00 8.1998 4.332e-16 \*\*\*  
## midwest -4.8743e+01 9.1366e+00 -5.3349 1.068e-07 \*\*\*  
## Visitors:newEngland -5.6137e-06 1.5349e-06 -3.6574 0.0002616 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Now that we’ve squared popularity, we decided to run a best subsets and stepwise to see which variables to include in our regression. The final stepwise model told us to use num\_reviews, pacific, innerMountain, avg\_rating, difficulty\_rating, and midwest while the best subsets model told us to run a regression with num\_reviews, pacific, innerMountain, avg\_rating, elevation\_gain, Visitors, and newEngland. These produced almost similar R-squared values, but the stepwise regression gave us more significant variables. Again, this significance is confirmed by using our robust standard errors also being statistically significant.

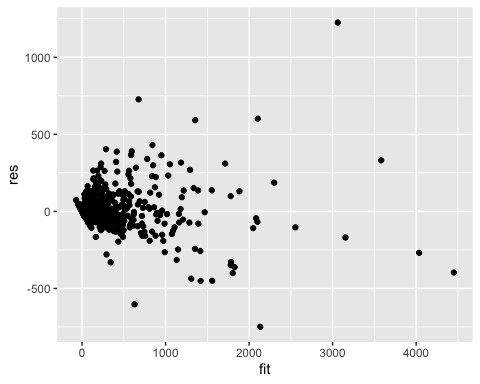
We also hypothesized that there may be some sort of interaction between the number of visitors a trail has and its region. That is, more populous regions or regions where the population density is greater, may see more visitors. This hypothesis was more or less confirmed with such a statistically significant p-value followed by an even more significant p-value using robust standard errors.

train4 <- train3 %>% mutate(res = resid(reg1)) %>% mutate(fit = fitted(reg1))  
  
train4 %>%  
 ggplot(aes(res))+  
 geom\_histogram()

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



train4 %>%  
 ggplot(aes(fit,res))+  
 geom\_point()



We have similar situation with our residual plots that we did last time. Our residual histogram looks relatively normally distributed but our residual histogram looks a bit heteroscedastic. This violates our model assumptions so we will have to be very selective in choosing which variables to include in our final model.

train8 <- train3 %>%  
 mutate(logRating = log(avg\_rating+1)) %>%  
 select(-avg\_rating)  
  
best.subset <- regsubsets(squarePop~.,train8,nvmax = 13)  
sum <- summary(best.subset)  
sum$outmat

## elevation\_gain difficulty\_rating num\_reviews Visitors point out  
## 1 ( 1 ) " " " " "\*" " " " " " "  
## 2 ( 1 ) "\*" " " "\*" " " " " " "  
## 3 ( 1 ) " " " " "\*" " " " " " "  
## 4 ( 1 ) "\*" " " "\*" " " " " " "  
## 5 ( 1 ) "\*" " " "\*" " " " " " "  
## 6 ( 1 ) "\*" " " "\*" " " " " " "  
## 7 ( 1 ) "\*" " " "\*" "\*" " " " "  
## 8 ( 1 ) "\*" " " "\*" "\*" "\*" " "  
## 9 ( 1 ) "\*" " " "\*" "\*" "\*" " "  
## 10 ( 1 ) "\*" " " "\*" "\*" "\*" " "  
## 11 ( 1 ) "\*" " " "\*" "\*" "\*" "\*"  
## 12 ( 1 ) "\*" "\*" "\*" "\*" "\*" "\*"  
## 13 ( 1 ) "\*" "\*" "\*" "\*" "\*" "\*"  
## Alaska innerMountain midwest newEngland pacific gradient  
## 1 ( 1 ) " " " " " " " " " " " "   
## 2 ( 1 ) " " " " " " " " " " " "   
## 3 ( 1 ) " " "\*" " " " " "\*" " "   
## 4 ( 1 ) " " "\*" " " " " "\*" " "   
## 5 ( 1 ) " " "\*" " " " " "\*" " "   
## 6 ( 1 ) " " "\*" "\*" " " "\*" " "   
## 7 ( 1 ) " " "\*" " " "\*" "\*" " "   
## 8 ( 1 ) " " "\*" " " "\*" "\*" " "   
## 9 ( 1 ) " " "\*" "\*" "\*" "\*" " "   
## 10 ( 1 ) "\*" "\*" "\*" "\*" "\*" " "   
## 11 ( 1 ) "\*" "\*" "\*" "\*" "\*" " "   
## 12 ( 1 ) "\*" "\*" "\*" "\*" "\*" " "   
## 13 ( 1 ) "\*" "\*" "\*" "\*" "\*" "\*"   
## logRating  
## 1 ( 1 ) " "   
## 2 ( 1 ) " "   
## 3 ( 1 ) " "   
## 4 ( 1 ) " "   
## 5 ( 1 ) "\*"   
## 6 ( 1 ) "\*"   
## 7 ( 1 ) "\*"   
## 8 ( 1 ) "\*"   
## 9 ( 1 ) "\*"   
## 10 ( 1 ) "\*"   
## 11 ( 1 ) "\*"   
## 12 ( 1 ) "\*"   
## 13 ( 1 ) "\*"

null <- lm(squarePop~1, train8)  
full <- lm(squarePop~., train8)  
step(null, scope=list(lower=null, upper=full),direction="forward")

## Start: AIC=22480.23  
## squarePop ~ 1  
##   
## Df Sum of Sq RSS AIC  
## + num\_reviews 1 193247716 12223221 16999  
## + logRating 1 3979560 201491377 22444  
## + point 1 1741721 203729216 22466  
## + elevation\_gain 1 1179731 204291206 22471  
## + innerMountain 1 782166 204688771 22475  
## + out 1 708581 204762356 22476  
## + gradient 1 695879 204775058 22476  
## + Visitors 1 470073 205000864 22478  
## + difficulty\_rating 1 287064 205183873 22480  
## + newEngland 1 283506 205187431 22480  
## <none> 205470937 22480  
## + midwest 1 179544 205291393 22480  
## + Alaska 1 76008 205394929 22482  
## + pacific 1 52656 205418281 22482  
##   
## Step: AIC=16999.15  
## squarePop ~ num\_reviews  
##   
## Df Sum of Sq RSS AIC  
## + elevation\_gain 1 215690 12007531 16967  
## + pacific 1 212124 12011098 16967  
## + midwest 1 208469 12014753 16968  
## + difficulty\_rating 1 88180 12135041 16987  
## + newEngland 1 65686 12157535 16991  
## + logRating 1 60762 12162459 16992  
## + gradient 1 51421 12171801 16993  
## + Alaska 1 32917 12190304 16996  
## + innerMountain 1 18466 12204755 16998  
## <none> 12223221 16999  
## + out 1 6011 12217210 17000  
## + Visitors 1 2849 12220373 17001  
## + point 1 1112 12222110 17001  
##   
## Step: AIC=16966.56  
## squarePop ~ num\_reviews + elevation\_gain  
##   
## Df Sum of Sq RSS AIC  
## + midwest 1 172989 11834542 16940  
## + pacific 1 144915 11862616 16945  
## + logRating 1 71951 11935580 16957  
## + newEngland 1 48375 11959156 16961  
## + innerMountain 1 34281 11973250 16963  
## + Alaska 1 27954 11979577 16964  
## + out 1 19480 11988051 16965  
## + Visitors 1 12519 11995012 16966  
## <none> 12007531 16967  
## + point 1 10892 11996639 16967  
## + gradient 1 6820 12000711 16968  
## + difficulty\_rating 1 1990 12005541 16968  
##   
## Step: AIC=16940.36  
## squarePop ~ num\_reviews + elevation\_gain + midwest  
##   
## Df Sum of Sq RSS AIC  
## + pacific 1 111383 11723159 16924  
## + logRating 1 70947 11763595 16931  
## + newEngland 1 63423 11771119 16932  
## + Alaska 1 31139 11803403 16937  
## + Visitors 1 24455 11810086 16938  
## + innerMountain 1 13892 11820650 16940  
## <none> 11834542 16940  
## + point 1 11995 11822547 16940  
## + out 1 9720 11824822 16941  
## + gradient 1 1396 11833145 16942  
## + difficulty\_rating 1 633 11833909 16942  
##   
## Step: AIC=16923.99  
## squarePop ~ num\_reviews + elevation\_gain + midwest + pacific  
##   
## Df Sum of Sq RSS AIC  
## + innerMountain 1 157249 11565909 16900  
## + logRating 1 71110 11652048 16914  
## + newEngland 1 29955 11693203 16921  
## + Alaska 1 23479 11699680 16922  
## <none> 11723159 16924  
## + point 1 6174 11716985 16925  
## + out 1 4954 11718205 16925  
## + Visitors 1 3929 11719229 16925  
## + difficulty\_rating 1 1275 11721884 16926  
## + gradient 1 146 11723013 16926  
##   
## Step: AIC=16899.75  
## squarePop ~ num\_reviews + elevation\_gain + midwest + pacific +   
## innerMountain  
##   
## Df Sum of Sq RSS AIC  
## + logRating 1 70518 11495391 16890  
## + point 1 12065 11553844 16900  
## <none> 11565909 16900  
## + Alaska 1 7735 11558174 16900  
## + newEngland 1 7129 11558780 16900  
## + Visitors 1 6750 11559160 16901  
## + gradient 1 4197 11561712 16901  
## + out 1 1467 11564442 16902  
## + difficulty\_rating 1 1342 11564567 16902  
##   
## Step: AIC=16889.86  
## squarePop ~ num\_reviews + elevation\_gain + midwest + pacific +   
## innerMountain + logRating  
##   
## Df Sum of Sq RSS AIC  
## + point 1 14214.2 11481177 16890  
## <none> 11495391 16890  
## + Alaska 1 7049.3 11488342 16891  
## + Visitors 1 6845.7 11488546 16891  
## + newEngland 1 5951.1 11489440 16891  
## + gradient 1 2109.2 11493282 16892  
## + out 1 1878.0 11493513 16892  
## + difficulty\_rating 1 843.5 11494548 16892  
##   
## Step: AIC=16889.46  
## squarePop ~ num\_reviews + elevation\_gain + midwest + pacific +   
## innerMountain + logRating + point  
##   
## Df Sum of Sq RSS AIC  
## <none> 11481177 16890  
## + Visitors 1 8134.9 11473042 16890  
## + Alaska 1 7144.3 11474033 16890  
## + newEngland 1 5090.4 11476087 16891  
## + gradient 1 846.1 11480331 16891  
## + difficulty\_rating 1 745.0 11480432 16891  
## + out 1 6.4 11481171 16892

##   
## Call:  
## lm(formula = squarePop ~ num\_reviews + elevation\_gain + midwest +   
## pacific + innerMountain + logRating + point, data = train8)  
##   
## Coefficients:  
## (Intercept) num\_reviews elevation\_gain midwest   
## -42.7192 1.9014 0.0109 -32.0334   
## pacific innerMountain logRating point   
## 30.9492 23.9136 19.4480 -9.2512

reg <- lm(formula = squarePop ~ num\_reviews + pacific + innerMountain + elevation\_gain + logRating + pacific:Visitors, data = train8)  
summary(reg)

##   
## Call:  
## lm(formula = squarePop ~ num\_reviews + pacific + innerMountain +   
## elevation\_gain + logRating + pacific:Visitors, data = train8)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -749.42 -21.43 -6.77 12.84 1224.29   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -4.703e+01 9.652e+00 -4.872 1.19e-06 \*\*\*  
## num\_reviews 1.903e+00 1.072e-02 177.491 < 2e-16 \*\*\*  
## pacific 3.413e+01 6.203e+00 5.502 4.25e-08 \*\*\*  
## innerMountain 2.747e+01 4.372e+00 6.283 4.10e-10 \*\*\*  
## elevation\_gain 1.032e-02 2.035e-03 5.071 4.34e-07 \*\*\*  
## logRating 1.926e+01 5.575e+00 3.455 0.000561 \*\*\*  
## pacific:Visitors 4.173e-07 1.780e-06 0.235 0.814616   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 77.26 on 1936 degrees of freedom  
## Multiple R-squared: 0.9438, Adjusted R-squared: 0.9436   
## F-statistic: 5415 on 6 and 1936 DF, p-value: < 2.2e-16

coeftest(reg, vcov = vcovHC(reg, type = "HC1"))

##   
## t test of coefficients:  
##   
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -4.7026e+01 4.5683e+00 -10.2940 < 2.2e-16 \*\*\*  
## num\_reviews 1.9027e+00 5.2182e-02 36.4631 < 2.2e-16 \*\*\*  
## pacific 3.4130e+01 5.3233e+00 6.4114 1.807e-10 \*\*\*  
## innerMountain 2.7466e+01 3.7181e+00 7.3871 2.220e-13 \*\*\*  
## elevation\_gain 1.0320e-02 2.4175e-03 4.2688 2.060e-05 \*\*\*  
## logRating 1.9263e+01 2.6229e+00 7.3443 3.033e-13 \*\*\*  
## pacific:Visitors 4.1735e-07 2.0389e-06 0.2047 0.8378   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Finally, we want to see what will happen when we decide to log our avg\_rating variable and add another interaction term. This didn’t seem to work too well. Although we still have a good R-squared, we lose some significance in the avg\_rating variable by logging it. This will not be a variable that we include in the final model.

test1 <- test %>% mutate(squarePop = popularity^2) %>% select(-popularity)  
  
finalModel <- lm(squarePop ~ num\_reviews + pacific + avg\_rating + innerMountain, data=test1)  
summary(finalModel)

##   
## Call:  
## lm(formula = squarePop ~ num\_reviews + pacific + avg\_rating +   
## innerMountain, data = test1)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1962.61 -23.94 -5.14 15.37 1095.72   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -45.35079 14.92861 -3.038 0.00243 \*\*   
## num\_reviews 1.98556 0.01437 138.199 < 2e-16 \*\*\*  
## pacific 45.62209 9.26700 4.923 9.62e-07 \*\*\*  
## avg\_rating 8.21688 3.16665 2.595 0.00957 \*\*   
## innerMountain 20.87209 8.91738 2.341 0.01940 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 114.5 on 1291 degrees of freedom  
## Multiple R-squared: 0.9383, Adjusted R-squared: 0.9381   
## F-statistic: 4905 on 4 and 1291 DF, p-value: < 2.2e-16

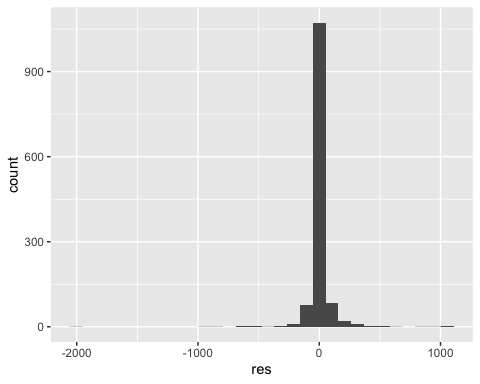
coeftest(finalModel,vcov = vcovHC(finalModel,type = "HC1"))

##   
## t test of coefficients:  
##   
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -45.350785 7.515202 -6.0345 2.080e-09 \*\*\*  
## num\_reviews 1.985560 0.083855 23.6785 < 2.2e-16 \*\*\*  
## pacific 45.622086 6.749615 6.7592 2.095e-11 \*\*\*  
## avg\_rating 8.216882 1.680719 4.8889 1.141e-06 \*\*\*  
## innerMountain 20.872088 7.381105 2.8278 0.00476 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

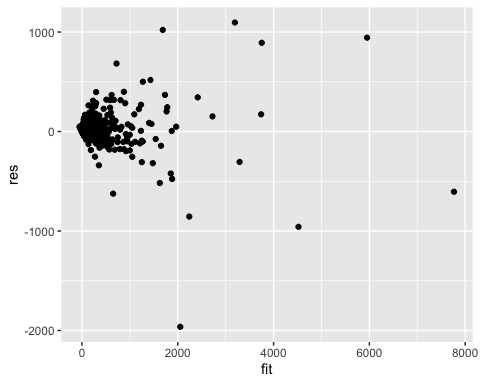
This is our final model based on which variables will best predict popularity squared. Bear in mind, when choosing variables for the final model, we had to be very picky about which variables to include and only choose those which the highest p-value. Doing this, we get an R-squared of 0.9383 which tells us that 93.83% of the variation in popularity squared is explained by num\_reviews, pacific, avg\_rating, and innerMountain. We also can see that for any 1 unit increase in num\_reviews, pacific, avg\_rating, and innerMountin, we would expect popularity squared to increase by 1.985, 45.622, 8.216, and 20.872 respectively. Again, this model is validataed by using our robust standard errors.

test2 <- test1 %>% mutate(res = resid(finalModel)) %>% mutate(fit = fitted(finalModel))  
  
test2 %>%  
 ggplot(aes(res))+  
 geom\_histogram()

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



test2 %>%  
 ggplot(aes(fit,res))+  
 geom\_point()



Testing our model assumptions, we see a similar result. We have a good residual histogram and a kind of conical residual scatterplot.

confint(finalModel)

## 2.5 % 97.5 %  
## (Intercept) -74.637784 -16.063786  
## num\_reviews 1.957374 2.013746  
## pacific 27.442055 63.802118  
## avg\_rating 2.004536 14.429228  
## innerMountain 3.377944 38.366232

newdata = data.frame(num\_reviews = 100, pacific = 1, avg\_rating = 4.5, innerMountain = 0)  
predict(finalModel,newdata,interval = "predict")

## fit lwr upr  
## 1 235.8033 10.829 460.7776

sqrt(10.829)

## [1] 3.290745

sqrt(235.8033)

## [1] 15.35589

sqrt(460.7776)

## [1] 21.46573

Our confidence interval of the final model reveals what the p-values in the regression told us. We can see that 0 is not contained or even close to being contained in any of the intervals. Note that pacific and innerMountain have such high coefficients because they are 0 1 variables and not continuous like avg\_rating and num\_reviews.

Finally, running a prediction for a trail with 100 reviews, in the pacific region, and with an average rating of 4.5, we can see that we are predicted a popularity squared of anywhere from 10.829 to 460.778 but the fitted value is 235.803. Taking the square root of these to find the popularity, we find that the popularity ranges any where from 3.29 to 21.47 with a fitted value of 15.36.

Overall, we were able to build a very effective model which accounted for more than 90% of the variation in our dependent variable. What comes next? Well, I’d like to test and see if being outdoors really has the benefits that are associated with it. Most importantly, I’d like to see if being outside is an effective treatment for COVID-19, or rather, any viral disease. We’d likely have to perform an experiment rather than a study to imply causation for the benefits of being outside.