STOR 455 - Class 21 - R Cross Validation

library(readr)  
StateSAT <- read\_csv("https://raw.githubusercontent.com/JA-McLean/STOR455/master/data/StateSAT.csv")  
Pulse <- read\_csv("https://raw.githubusercontent.com/JA-McLean/STOR455/master/data/Pulse.csv")

**Example: State SAT Scores** Source: Statistical Sleuth, Case 12.1 pg. 339  
Response Variable:  
SAT =Average combined SAT Score Potential Predictors:  
Takers = % taking the exam Income = median family income ($100’s) Years = avg. years of study (SS, NS, HU) Public = % public school Expend = spend per student ($100’s) Rank = median class rank of takers

**Second Order Models** - Definition: A second order model for two quantitative predictors would be Y = Bo + First Order + First Order + Quadratic + Quadratic + Interaction, where Quadratic + Quadratic \_ Interaction = Second ORder - Y = B0 + B1X1 + B2X2 + B3X1^2 + B4X2^2 + B5X1\*X2 + Error - Example: Try a full second order model for Y=SAT using X1=Takers and X2=Expend

*Previously on* - We did with taker’s vsariable - We looked at quadratic, transformation alone wasnt good enoughb/c we can’t change vertex - Cubic didn’t give us much there - we could keep going, but we’re probabbly going to overfit too much if we kept going *Polynomial regression in other dimensions* - It can extend to other dimensions - When we have 2 dimentions, it’s a plane, but as we add more, then it’s not technically a line, but we call it a line - A curved plane in 3D, we are building this polynomial with 2 extra predictors - We are looking at the terms themselves and also looking at the squared terms themsevles adnthe interaction between them (this is the second order model we are looking at)

# Full SEcond ORder Model   
modSAT2ndorder=lm(SAT~Takers+Expend+I(Takers^2)+I(Expend^2)+I(Takers\*Expend),data=StateSAT)  
summary(modSAT2ndorder)

##   
## Call:  
## lm(formula = SAT ~ Takers + Expend + I(Takers^2) + I(Expend^2) +   
## I(Takers \* Expend), data = StateSAT)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -50.472 -13.535 1.023 8.866 60.870   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 893.66283 36.14094 24.727 < 2e-16 \*\*\*  
## Takers -7.05561 0.83740 -8.426 9.96e-11 \*\*\*  
## Expend 10.33333 2.49600 4.140 0.000155 \*\*\*  
## I(Takers^2) 0.07725 0.01328 5.816 6.28e-07 \*\*\*  
## I(Expend^2) -0.11775 0.04426 -2.660 0.010851 \*   
## I(Takers \* Expend) -0.03344 0.03716 -0.900 0.373087   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 23.68 on 44 degrees of freedom  
## Multiple R-squared: 0.8997, Adjusted R-squared: 0.8883   
## F-statistic: 78.96 on 5 and 44 DF, p-value: < 2.2e-16

# Summary at bottom tells us the overall anova test   
# Ho: coef of all the coef are zero vs the Ha: at least one is non zero <- for the ANOVA test   
# Low pvalue - There is a relationship with at least one of these predictors   
# Looking at the individual predictors and there are many that are very low   
# The itneraction term is now a sig pvalue   
# FOr a model like this for polynomial regression or the compelte second order model, when we are looking at adding categorical variables, we don't want ot pick and choose different parts of this model and keep them   
# We want either a full polynomial model with all the degrees of the predictor in it or some other model   
# Even though the interaction term isn't sig, we don't want to get rid of it   
  
# Residual analysis   
# Linearity looks good   
# Normal looks pretty good, some devation but pretty good   
  
# overall: Seems like a good model   
  
# Adj R squared; almost 90% of the varability is predicted by this model   
# One warning error: There are some weird things with the plot, but don't worry about that; AK is just giving us a leverage warning   
  
# Do we need the interaction term?   
# YEs, if we have the other htings   
  
# Do we need the second order terms?  
# Are any useful? Takers and linear parts of it, and the squared value   
# Do the extra terms give us much improvement vs a model without it? We can't tell that from teh summary table, we have to do some drop in deviance test to tell that or a nested test   
  
# Do we need the terms with Expend?  
# A model with just tackers was a good model, and that is a subset of this one, so do we need the extra things?

**NEsted test to see if we don’t need the additional terms**

modSAT2ndorder\_Reduced=lm(SAT~Takers+Expend+I(Takers^2)+I(Expend^2),data=StateSAT)  
anova(modSAT2ndorder\_Reduced, modSAT2ndorder) # named mod2 in the lecutre, may refer to as mod2

## Analysis of Variance Table  
##   
## Model 1: SAT ~ Takers + Expend + I(Takers^2) + I(Expend^2)  
## Model 2: SAT ~ Takers + Expend + I(Takers^2) + I(Expend^2) + I(Takers \*   
## Expend)  
## Res.Df RSS Df Sum of Sq F Pr(>F)  
## 1 45 25123   
## 2 44 24669 1 454 0.8098 0.3731

# If wanted to see the additional terms int eh second order, we can do an anova test with teh mod2 and the full model from the previoud chunck   
# THis is a hypo test askign if any of the additional second order model terms are useful for us   
# Null: Is the coef for takers squared, explanatory square and the itnearction all equal to zero   
# Alternative: Do we have evidence that they are not equal to zero? (Meaning that there is some sig)  
# Low pvalue - at least one is non zero   
# The full second order mdoel looks like an improvemnet instead of jsut using a linear model   
  
# What if we build a model with jsut the taker's terms in it?   
modSAT2ndorder\_Reduced2=lm(SAT~Takers+Expend, data=StateSAT)  
anova(modSAT2ndorder\_Reduced2,modSAT2ndorder)

## Analysis of Variance Table  
##   
## Model 1: SAT ~ Takers + Expend  
## Model 2: SAT ~ Takers + Expend + I(Takers^2) + I(Expend^2) + I(Takers \*   
## Expend)  
## Res.Df RSS Df Sum of Sq F Pr(>F)   
## 1 47 56278   
## 2 44 24669 3 31609 18.793 5.404e-08 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

# The above anova test checks that   
# Ho: Is the coef for expend, expend^2, and interaction = 0   
# HA: Is coef for at least one != 0   
# Small pvalue, at least one is nonzero   
# This might be the more ideal model   
# WE shoudl also check the lienar model conditions  
  
# Linear model conditions are good   
modSAT2ndorder\_Reduced3 = lm(SAT~Takers+I(Takers^2),data=StateSAT)  
anova(modSAT2ndorder\_Reduced3, modSAT2ndorder)

## Analysis of Variance Table  
##   
## Model 1: SAT ~ Takers + I(Takers^2)  
## Model 2: SAT ~ Takers + Expend + I(Takers^2) + I(Expend^2) + I(Takers \*   
## Expend)  
## Res.Df RSS Df Sum of Sq F Pr(>F)   
## 1 47 42101   
## 2 44 24669 3 17432 10.364 2.787e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

# not going to look at very much, but this really is so you cna use it if you don't want to or transofmration just aren't doing it

**Would our model do a good job at predicting future if we got a different set of data?**

**Cross Validation** - Concern: A model may reflect the structure of a particular sample, but not generalize well to the population. - Cross validation checks for overfitting

**To see if this is a problem:** Split the original sample into two parts (a) A “training” sample to build a model (b) A “holdout” sample to test the model

1. Build model on a training set,

* subset it to build the model
* keep a holdout testing sampel to see how well the model does with teh new data
* sometimes you will have to break it up yourself

**Example: Pulse Rates** Response Variable:  
Active pulse Predictors:  
Resting pulse Hgt Wgt Sex (0=M, 1=F) Smoke (0=No, 1=Yes) Exercise (1=Slight, 2=Moderate, 3=Lots)

**Example: Active Pulse Rates** Training sample – first 300 cases (PulseTrain) Holdout sample – #301-375 (PulseHoldout)

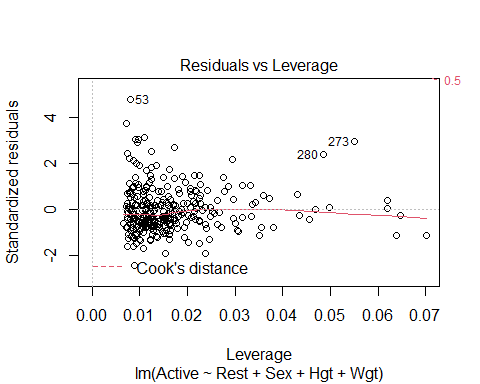
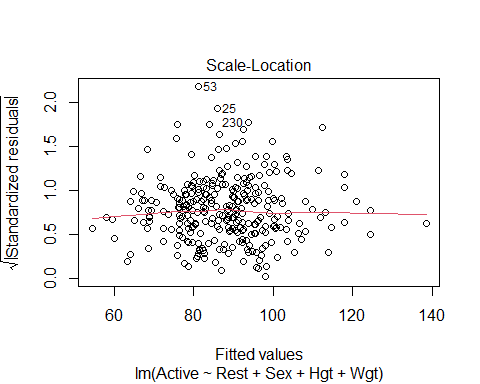
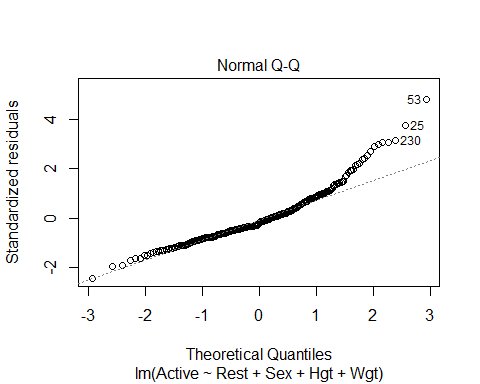
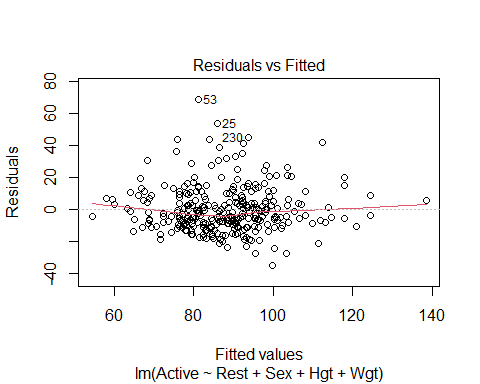
set.seed(12345) # Only set the seed to get the same results at the end, if you want actual random, then don't set seed  
# Want to do it randomly because we dont want to have a certain connection with the different rows  
rows <- sample(nrow(Pulse)) # Counts rows in pulse and takes 374 values without repleacement in a random order  
Pulse\_shuffled = Pulse[rows,] # reassign pulse in a different order here   
  
PulseTrain=Pulse\_shuffled[1:300,] # Make the training data, took about 75% of the data into the training   
PulseHoldout=Pulse\_shuffled[301:375,] # Holdout

What is the best model to predict Active pulse?

#"best" model  
# Ran a model selection sequence, we don't really care how we got this for the thing for this example  
PulseTrainMod=lm(Active~Rest+Sex+Hgt+Wgt,data=PulseTrain)  
  
summary(PulseTrainMod)

##   
## Call:  
## lm(formula = Active ~ Rest + Sex + Hgt + Wgt, data = PulseTrain)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -34.869 -8.916 -2.794 6.515 68.782   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 33.70967 23.49083 1.435 0.15234   
## Rest 1.19626 0.08650 13.830 < 2e-16 \*\*\*  
## Sex 4.30152 2.51881 1.708 0.08873 .   
## Hgt -0.69392 0.35542 -1.952 0.05184 .   
## Wgt 0.11892 0.04128 2.881 0.00426 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 14.44 on 295 degrees of freedom  
## Multiple R-squared: 0.4216, Adjusted R-squared: 0.4137   
## F-statistic: 53.75 on 4 and 295 DF, p-value: < 2.2e-16

plot(PulseTrainMod) # WE dont want to touch the holdout sample, not until the end



# Looking at the model   
# SUmmary   
# ANOVA test gives us evident that some of the things are nonzero   
# Height isn't good on its own - probably multicollinearity going on here   
# Things correlated with height and all that   
  
# Model conditions   
# Some curve that could be an ainssue, we could flatten it out with transformations, but we wont mess with that now   
# WE could have issues with constant variance, but we wont mess with either   
# Normal is an issue, skew on the right side

* We want to use the model made to predict the data in teh test data **Fit for Training Model** Another way to think of R2: 𝑅2 = square of correlation between 𝑌 and 𝑌̂
* If we use this model to predict all the values in the new data set, we can find the cross validation correlation
* how well correlated the actual values are predicted vs how re predict they are with the model constructed from the old data
* WE’re coming the R2 for the other models with the new model

**How does the training model work for the holdout sample?**  - Compute predicted values for the holdout sample using the fitted prediction equation from the training. – fitActive=predict(mod,newdata=PulseHoldout) - Compute the residuals for the holdout sample. – holdoutresid=PulseHoldout$Active - fitActive

We want to see how far off we are from teh testing data to the holdout sample

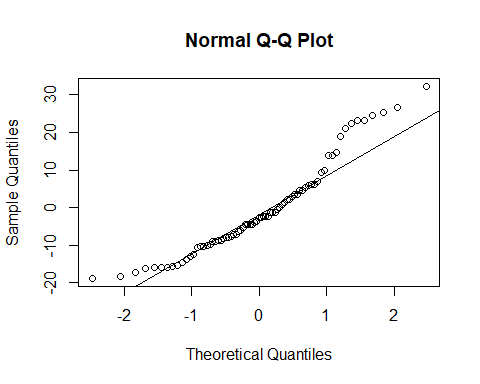
#predict active heart rates for data in the holdout sample with the model made from the training data  
fitActive=predict(PulseTrainMod,newdata=PulseHoldout)  
# Just all teh predctions   
  
#Actual active heart rates in holdout sample minus their predicted values  
# how far off are we?   
holdoutresid=PulseHoldout$Active - fitActive  
# Looks at the residuals; so the actual - the predicted   
  
# WE saw the linear was not prefect in teh past; does this model predict data in a simialr way   
# Is there as imilar center, and spread and the Residual SE the same?   
# IS it distributed nmuch the same way?   
# IS there the same type of skew in this or is it different?   
#So welook at the center, spread and shape below   
  
#Center, spread, and shape for the distribution of holdouts  
mean(holdoutresid)# Want value to be close to zero; because if its above or below it, there are bias in teh predicitons

## [1] -0.826801

# Big or little is subjective to teh data; this is saying we are off 1 beat per minute, if we are predicting GPA, then 1 point would be really bad and off, but this isn't too bad   
sd(holdoutresid)

## [1] 12.21473

# See how spread out htings are   
# THis says 12.21 doesn't say much on it's own   
# Is the similar spread as the orignal dat?   
# So look at teh orinigal training mod summary, and that is the REsd SE, and we want to see if the REsd SE is similar to teh spread of the SE of the holdout residusla   
# One is 12 and one is 14, so the dat appears more compact, we odnt knwo why, but its okay   
# IT doesn't llook drastically different   
# THis is saything that htere is some difference, but it looks okay   
  
# Distribution plots   
qqnorm(holdoutresid)  
qqline(holdoutresid)



# Above we see that the line looks okay,   
# It's not great, there are tail issues with skew, but its liek the orignial data, so there doesnt look to be mcuh difference in teh shape   
# This tells you should probably fix it in the oringial model   
# Check it with residuals to make sure no drastic differences between teh other predicitons in teh orinal model   
#Want th shape to be similar to the orinigal model

cor(PulseHoldout$Active,fitActive)

## [1] 0.6474953

# Correlation bt holdout and the predictions for the values   
# This tells you there is a 0.64 correlation bt the two values   
# THis means that   
# Compare teh value from a previous model   
# THis is the cross validation ocorreation   
#Look below

#Correlation between predicted and actual active heart rates  
crosscorr=cor(PulseHoldout$Active,fitActive)  
# In previous model the multi r squared did the same thing; it was jsut the square of teh predicted adn actual values   
crosscorr^2

## [1] 0.4192502

# Looking at the difference bt crosscorr^2 and what the oringial model says tells you the shrinkage from the orinigal and the test   
# Want to see how different the predicitons with teh model correlatioe with teh actual and predicted values   
# How well teh actual and predicted values correalte even though this model wasn't use to build the model   
# WE want these to be similar and very close values   
  
#Change in r^2 from the training to the holdout  
shrinkage = summary(PulseTrainMod)$r.squared-crosscorr^2  
shrinkage

## [1] 0.002325805

# Very clsoe to zero, less than 1 percentage  
# Gives a some measure how not as well the oringal model predcits teh new data   
# IF teh thing is big, then you're not predicitng very well for the new data vs the old data   
# If it is big you're probably overfitting the original model   
# You are really only get a value from 0 - 1, under 0.15 is pretty okay, but above 0.2 can be kinda bad   
# YOu could get a negative, which would tell you that you are predicting your stuff better than thte oringial model