Class 26 R Logistic Regression

library(titanic)  
data("titanic\_train")  
head(titanic\_train)

## PassengerId Survived Pclass  
## 1 1 0 3  
## 2 2 1 1  
## 3 3 1 3  
## 4 4 1 1  
## 5 5 0 3  
## 6 6 0 3  
## Name Sex Age SibSp Parch  
## 1 Braund, Mr. Owen Harris male 22 1 0  
## 2 Cumings, Mrs. John Bradley (Florence Briggs Thayer) female 38 1 0  
## 3 Heikkinen, Miss. Laina female 26 0 0  
## 4 Futrelle, Mrs. Jacques Heath (Lily May Peel) female 35 1 0  
## 5 Allen, Mr. William Henry male 35 0 0  
## 6 Moran, Mr. James male NA 0 0  
## Ticket Fare Cabin Embarked  
## 1 A/5 21171 7.2500 S  
## 2 PC 17599 71.2833 C85 C  
## 3 STON/O2. 3101282 7.9250 S  
## 4 113803 53.1000 C123 S  
## 5 373450 8.0500 S  
## 6 330877 8.4583 Q

**Logistic Regression** In all of our regression models (so far) the response variable, Y, has been quantitative. What if we want to model a categorical response?

**Categorical Response Variables** - Ways you can think about categorical response variables - WE will only focus on binary responses

* Binary Response: Whether or not a person smokes and Success of a medical treatment, where Y is divided into NOn-smoker vs smoker and X is divided into Durvies vs Dies
* Ordinal Response: Opinion Poll responses: Where Y = Agree, Netural, and Disagree
* Nominal Response: Political preference; where y = Democrat, Republican, independent

**Binary Logistic Regression** - Response variable (Y) is categorical with just two categories (yes/no or success/failure or 0/1 …). - One approach: Code the response Y as a (0,1) dummy (indicator) variable. - Assume we have a single quantitative predictor X.

**Titanic Survival** Y = Survived (0 = no; 1 = yes) X = Fare (ticket cost in dollars) - Want to predict if the people on the titance survived based on how much they paid

# amkes a table that tells you how many people survived out of the titanic overall   
# Just how many people survived total vs died  
table(titanic\_train$Survived)

##   
## 0 1   
## 549 342

# Survival related to ticket   
# SO a table on if survived based on what type of ticket class they bought   
# Shows a rough relatioship between the pclass and the others   
table(titanic\_train$Survived, titanic\_train$Pclass)

##   
## 1 2 3  
## 0 80 97 372  
## 1 136 87 119

# The below caompres the survided to teh class,   
  
# the first calss,   
# third class a lot didn't survived   
# Wouldnt an underlying variable be that there are more people buying lower class tickets than higher class tickets? I think people should look at the proportion   
# WEll, based on teh propprtions, it still looks like the upper class surived more, hmm wonder why

Below: low pvale; we ave strong ev to say that it si nonzero and there is some linear relationship

* if we look at teh model we can plot survied by fare;

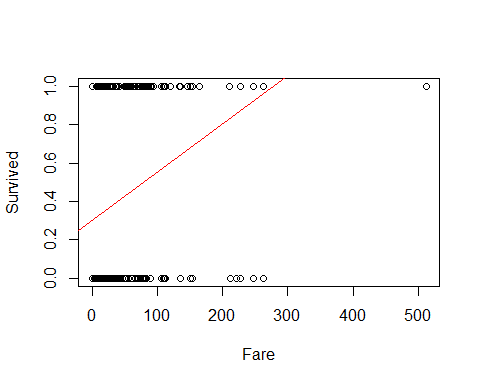
Titanic\_mod=lm(Survived ~ Fare, data=titanic\_train)  
summary(Titanic\_mod)

##   
## Call:  
## lm(formula = Survived ~ Fare, data = titanic\_train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.9653 -0.3391 -0.3222 0.6044 0.6973   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.3026994 0.0187849 16.114 < 2e-16 \*\*\*  
## Fare 0.0025195 0.0003174 7.939 6.12e-15 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.4705 on 889 degrees of freedom  
## Multiple R-squared: 0.06621, Adjusted R-squared: 0.06516   
## F-statistic: 63.03 on 1 and 889 DF, p-value: 6.12e-15

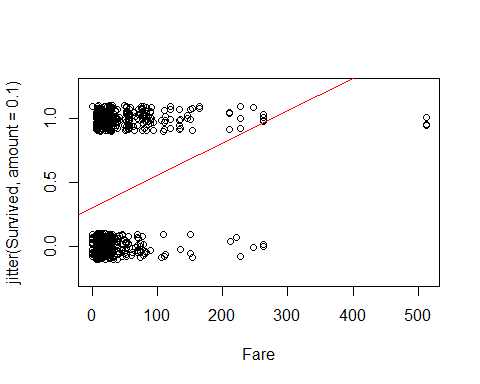
plotting surivied by fare; hard to see how dense it is; there are a lot of calues on top of the m on the bottom;

if we jitter teh data, it moves it up and down a random amount so we can see the visual difference; it doens’t chnage the data, it just changes the visual of it.

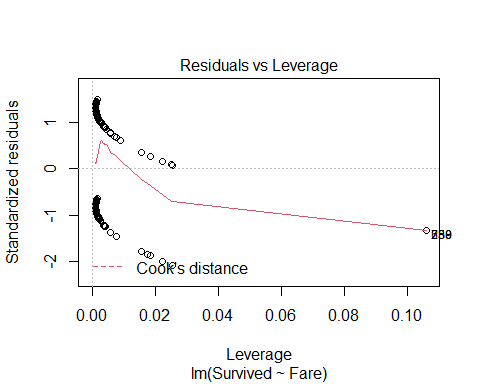
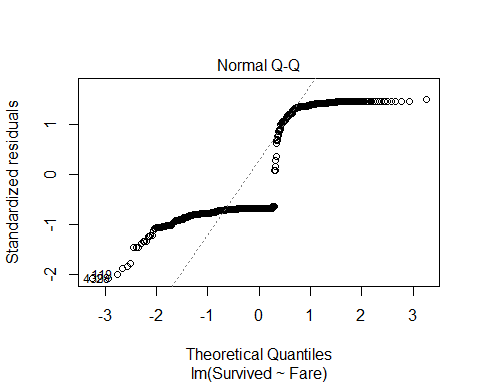
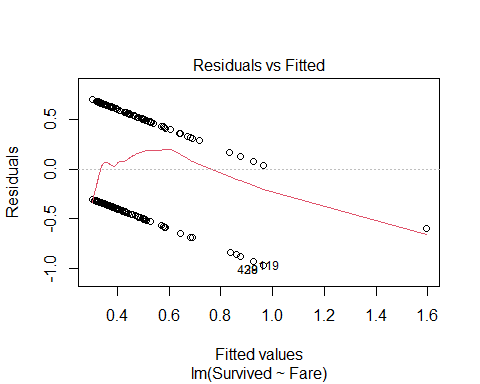
plot(Survived ~ Fare, data=titanic\_train)  
abline(Titanic\_mod, col="red")

 Jitter shows that there is a trend; there appeares to be if you paied more for your ticket, you appear to survied more - residual analysis helps see the difference in plottin gof the things

plot(  
 jitter(Survived, amount=0.1) ~ Fare,   
 ylim = c(-0.25,1.25),   
 data=titanic\_train  
 )  
  
 abline(Titanic\_mod, col="red")



plot(Titanic\_mod, c(1, 2, 5))



The aove shows that teh resultial by fitted has a path residuals are NOT normally distributed and the cook’s distance doesn’t hae any poitns of influence;

but bottom line the model doesn’t work very well

**Binary Logistic Regression Model** Y = Binary response X = Quantitative predictor π = proportion of 1’s (yes, success,…) at any x Probability form 𝜋=𝑒^(𝛽\_𝑜+𝛽\_1 𝑥)/(1+𝑒^(𝛽\_𝑜+𝛽\_1 𝑥) ) - curve(exp(B0+B1*x)/(1+exp(B0+B1*x)),add=TRUE, col=“red”)

below is a model of it by fare ; yo umaek teh family binomial What does it mean when you make the family binomial? Does that make it binary? - No it’s not binary, it means that it is a squared plot

Titanic\_logitmod = glm(Survived ~ Fare, family = binomial, data=titanic\_train)  
# Darws a curve that has a curve in teh middle with a similar likelihood of surviving or not suriving   
# We are looking at the model and predicting teh pi outcome   
# The probabiltiy of that outcome   
# Predict prob of 0 - 1someone who fits this fare would survive based ont eh model we have created   
# WE have to use teh glm function to tdo that   
  
summary(Titanic\_logitmod)

##   
## Call:  
## glm(formula = Survived ~ Fare, family = binomial, data = titanic\_train)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.4906 -0.8878 -0.8531 1.3429 1.5942   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -0.941330 0.095129 -9.895 < 2e-16 \*\*\*  
## Fare 0.015197 0.002232 6.810 9.79e-12 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 1186.7 on 890 degrees of freedom  
## Residual deviance: 1117.6 on 889 degrees of freedom  
## AIC: 1121.6  
##   
## Number of Fisher Scoring iterations: 4

# this can be aline if we want it to be   
# But we have to look at it differently

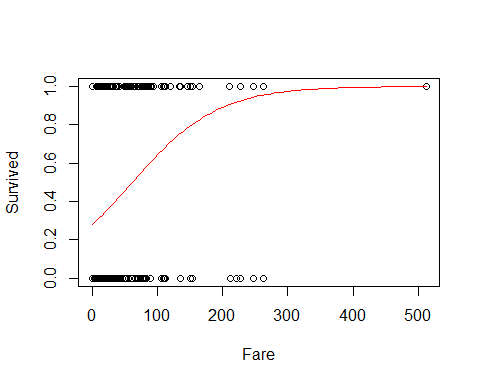
ABove, we are going to claim that the above is a linear model; we;re going to replot the data with teh jitter data; then see if the curve function equation looks nicer

we will learn later where teh curve function is coming from

we want to plot the curve = (exp(B0+B1*x))/(1+exp(B0+B1*x)) (Can see this formula filled in below)

**Predicting Proportion of “Success”** In regression the model predicts the mean Y for any combination of predictors. - What’s the “mean” of a 0/1 indicator variable? 𝑦̄=(∑𝑦\_𝑖 )/𝑛=(#” of ” 1′𝑠)/(#” of trials” )=“Proportion of "success"” - Goal for this model: Predict the “true” proportion of success, π, at any value of the predictor.

plot(Survived ~ Fare, data=titanic\_train)  
  
B0 = summary(Titanic\_logitmod)$coef[1]  
B1 = summary(Titanic\_logitmod)$coef[2]  
  
curve(exp(B0+B1\*x)/(1+exp(B0+B1\*x)),add=TRUE, col="red")



# predicitng the changes of dying based ont eh ticket you bought   
# WE say that there is about a 20% chance of dying if you payed a ceratin amount. THat's what the red line says; at what price of your ticket would oyu have X precentage of curiviing or dying

set.seed(10012020)  
passenger = titanic\_train[sample(nrow(titanic\_train),1),]  
passenger

## PassengerId Survived Pclass Name Sex Age SibSp  
## 622 622 1 1 Kimball, Mr. Edwin Nelson Jr male 42 1  
## Parch Ticket Fare Cabin Embarked  
## 622 0 11753 52.5542 D19 S

# We are randomly selecting one person so that we can check the residuals for a random value

predict(Titanic\_logitmod, passenger, type="response")

## 622   
## 0.4643927

# This is telling us, what do we predict a person who bought a certain amount's chance of surviving?   
# This looks at how much they paid for their ticket and tells us where on teh red curve we would expect this dude to fall   
# So thsi tells us that the dude has a 46% chance of surviving if he paid X amount for his ticket

**Binary Logistic Regression Model** Y = Binary response X = Quantitative predictor π = proportion of 1’s (yes, success,…) at any x Probability form: 𝜋=𝑒^(𝛽\_𝑜+𝛽\_1 𝑥)/(1+𝑒^(𝛽\_𝑜+𝛽\_1 𝑥) ) Logit form: log⁡(𝜋/(1−𝜋))=𝛽\_0+𝛽\_1 𝑥 **NOTE** The logit form can be solved to be in linear form, which is why we can use linear regression rules with it.

**Binary Logistic Regression Model** **Probability Form:** P(X) = ((e(Bo+B1X))/(e(B0+B1X)+1)) **Logit Form:** ln(p/(1-p)) = B0 + B1X

**Example: Golf Putts** Build a model to predict the proportion of putts made (success) based on length (in feet). Data are in Putts1 of Stat2Data.

library(Stat2Data)  
  
data("Putts1")  
head(Putts1)

## Length Made  
## 1 3 1  
## 2 3 1  
## 3 3 1  
## 4 3 1  
## 5 3 1  
## 6 3 1

**Logistic Regression for Putting**

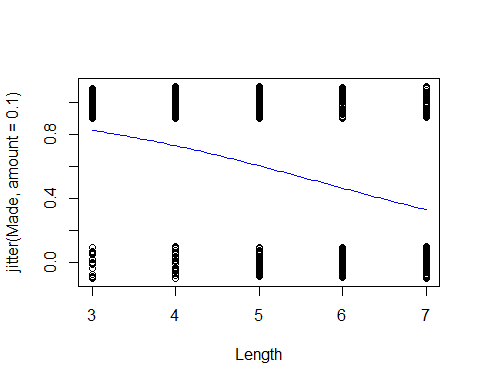
modPutt=glm(Made~Length,family=binomial,data=Putts1)  
summary(modPutt)

##   
## Call:  
## glm(formula = Made ~ Length, family = binomial, data = Putts1)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.8705 -1.1186 0.6181 1.0026 1.4882   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 3.25684 0.36893 8.828 <2e-16 \*\*\*  
## Length -0.56614 0.06747 -8.391 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 800.21 on 586 degrees of freedom  
## Residual deviance: 719.89 on 585 degrees of freedom  
## AIC: 723.89  
##   
## Number of Fisher Scoring iterations: 4

# pvalues are small   
# So we like these realtionships, but we should plot it to see what it actually loks like and that is done below

logit = function(B0, B1, x)  
 {  
 exp(B0+B1\*x)/(1+exp(B0+B1\*x))  
}  
# THis function will make the curve that we need on the curve above   
#SO , this is the same as above

B0 = summary(modPutt)$coef[1]  
B1 = summary(modPutt)$coef[2]  
  
plot(jitter(Made,amount=0.1)~Length,data=Putts1)  
# These lines overall plot what the data looks like   
curve(exp(B0+B1\*x)/(1+exp(B0+B1\*x)),add=TRUE, col="red")  
# This line is the line that we would use to predict someone will make a put based on the distance from the hole.   
  
# Can also use the logit function   
curve(logit(B0, B1, x), add = TRUE, col = "blue")



# THis will make the same line as above

logit = function(B0, B1, x)  
 {  
 exp(B0+B1\*x)/(1+exp(B0+B1\*x))  
}  
# THis function will make the curve that we need on the curve above   
#SO , this is the same as above

**Golf Putts Probabilities** 𝜋̂=𝑒(3.257−0.5661𝐿𝑒𝑛𝑔𝑡ℎ)/(1+𝑒(3.257−0.5661𝐿𝑒𝑛𝑔𝑡ℎ) ) Where phat = 𝑝̂=(# 𝑚𝑎𝑑𝑒)/(# 𝑡𝑟𝑖𝑎𝑙𝑠)

**Golf Putts Probabilities** Length: 3,4,5,6,7 phat: 0.835, 0.739, 0.565, 0.488, 0.328 pihat: 0.826, 0.730, 0.605, 0.465, 0.330

Making a table of those that are made vs failed at different lengths

Putts.table = table(Putts1$Made, Putts1$Length)  
Putts.table

##   
## 3 4 5 6 7  
## 0 17 31 47 64 90  
## 1 84 88 61 61 44

p.hat = as.vector(Putts.table[2,]/colSums(Putts.table))  
# Make it a vector because we want to be able to use it with dataframes   
p.hat

## [1] 0.8316832 0.7394958 0.5648148 0.4880000 0.3283582

pi.hat=0  
  
# Compare the predictions, so 3 - 7 feet; so from 3 ft to 4 ft to 5 ft to 6 ft etc.   
# Will make the pihat for each of these   
# Pihat = the probability of sucess at a certain feet distance   
# Pi = success/trials   
for(i in 3:7)  
 {  
 pi.hat[i-2] = logit(B0, B1, i)  
 }  
  
pi.hat

## [1] 0.8261256 0.7295364 0.6049492 0.4650541 0.3304493

# Makea a dataframe that tells you the pihat values and the p hat values   
# We dont know the difference btween pi and p hat   
Putts = data.frame(  
 "Length" = c(3:7),   
 "p.hat" = p.hat,   
 "pi.hat" = pi.hat)  
  
head(Putts)

## Length p.hat pi.hat  
## 1 3 0.8316832 0.8261256  
## 2 4 0.7394958 0.7295364  
## 3 5 0.5648148 0.6049492  
## 4 6 0.4880000 0.4650541  
## 5 7 0.3283582 0.3304493

* Probability form of puttin gmodel\*
* etended from 0 - 12; the points on teh graph are the actual proprtions that were made; the p hat values;

the line shows the pi hat values; teh line on the 4 = what we predict; teh dot = the actual value - these are close adnthis is how we test teh linear model - we are going to put it back to the logit form so that we cna put this on a line and we want to see if teh data fits teh lie - if it doens’t then we will have to do transformations - it gets a lot math-y-er we agoig to spend next class talking more about the math how to lok at hypothesis testing and anova doesn’t make sence here anymore because teh residual doesn’t work each point represntes differen combination of data points.

**Probability Form of Putting Model**

plot(p.hat~Length,ylim=c(0,1), xlim=c(0,12), data=Putts)  
curve(logit(B0, B1, x),add=TRUE, col="red")

