statistical-modeling.R

tdham

2025-03-31

library(dplyr)

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

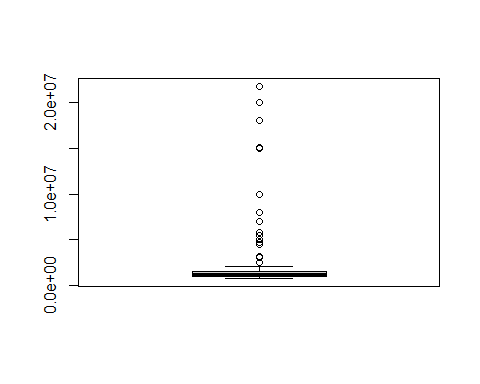
## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

library(readr)  
  
data <- read\_csv("../final\_clean\_nfl.csv")

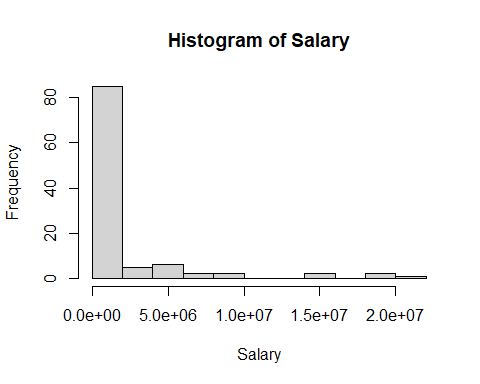
## New names:  
## • `` -> `...1`

## Rows: 105 Columns: 15  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## chr (1): Player  
## dbl (14): ...1, Rec, Yds, TD, X20., X40., Rec.1st, X1st., Rec.FUM, Rec.YAC.R...  
##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

data <- arrange(data, data$Salary)  
attach(data)  
  
  
# MODEL 1: Salary ~ catchRate  
# plot histograms (or boxplots) of each salary  
boxplot(Salary)



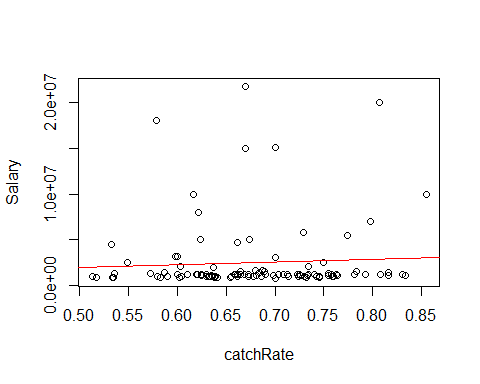
hist(Salary)



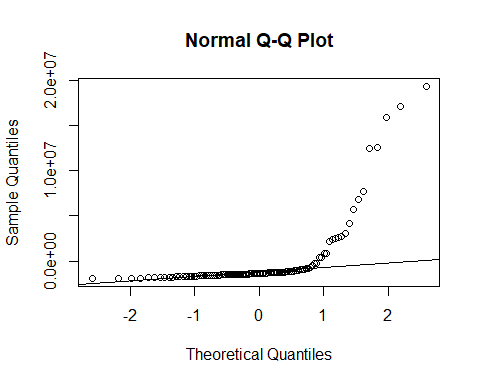
model1 <- lm(Salary~catchRate)  
summary(model1)

##   
## Call:  
## lm(formula = Salary ~ catchRate)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1915514 -1553677 -1357043 -876916 19331292   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) 487768 3451913 0.141 0.888  
## catchRate 2981732 5057791 0.590 0.557  
##   
## Residual standard error: 3951000 on 103 degrees of freedom  
## Multiple R-squared: 0.003363, Adjusted R-squared: -0.006313   
## F-statistic: 0.3475 on 1 and 103 DF, p-value: 0.5568

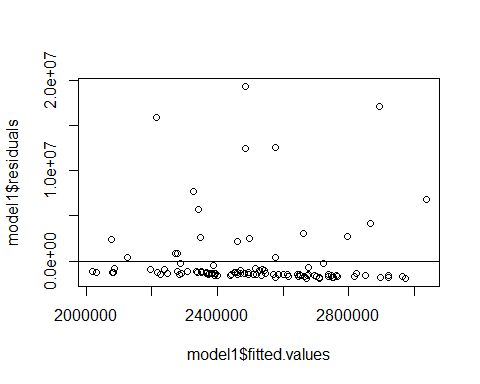
plot(catchRate, Salary)  
abline(model1, col="red")



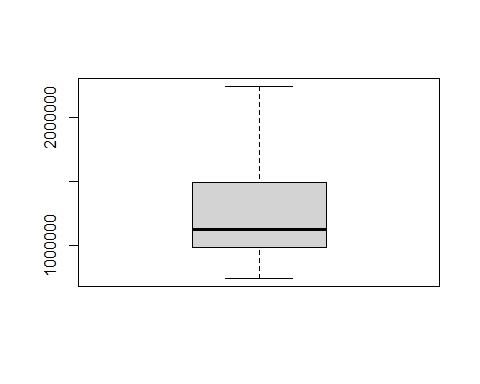
qqnorm(model1$residuals)  
qqline(model1$residuals)



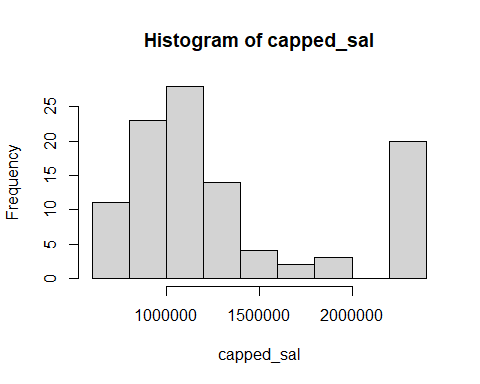
plot(model1$fitted.values, model1$residuals)  
abline(h=0)



# MODEL 1 - CAPPED OUTLIERS  
q1 <- quantile(Salary, 0.25)  
q3 <- quantile(Salary, 0.75)  
iqr <- q3 - q1  
lower <- q1 - 1.5\*iqr  
upper <- q3 + 1.5\*iqr  
  
capped\_sal <- ifelse(Salary < lower, lower,  
 ifelse(Salary > upper, upper, Salary))  
  
boxplot(capped\_sal)



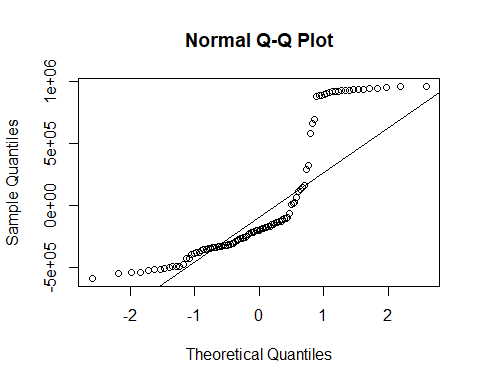
hist(capped\_sal)



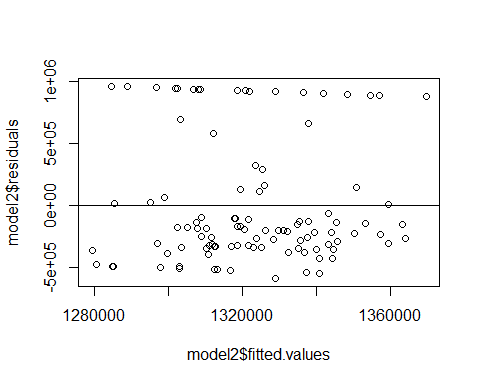
model2 <- lm(capped\_sal~catchRate)  
summary(model2)

##   
## Call:  
## lm(formula = capped\_sal ~ catchRate)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -589328 -338005 -201322 149275 962740   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1143921 443209 2.581 0.0113 \*  
## catchRate 264073 649397 0.407 0.6851   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 507200 on 103 degrees of freedom  
## Multiple R-squared: 0.001603, Adjusted R-squared: -0.00809   
## F-statistic: 0.1654 on 1 and 103 DF, p-value: 0.6851

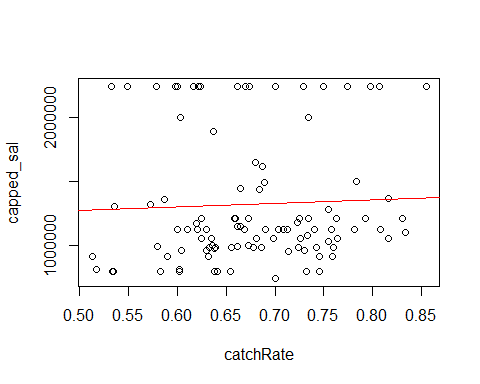
qqnorm(model2$residuals)  
qqline(model2$residuals)



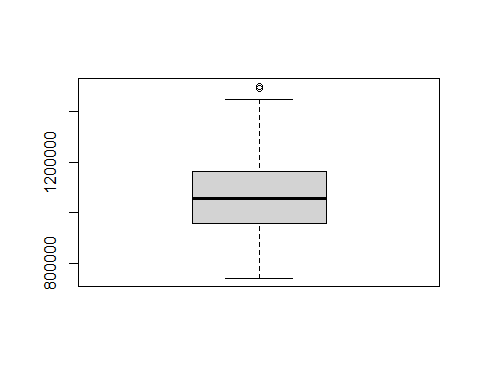
plot(model2$fitted.values, model2$residuals)  
abline(h=0)



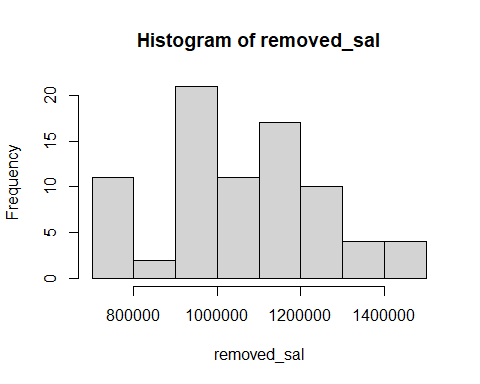
plot(catchRate, capped\_sal)  
abline(model2, col="red")



# MODEL 1 - REMOVED OUTLIERS  
removed\_sal <- Salary[1:80]  
  
boxplot(removed\_sal)



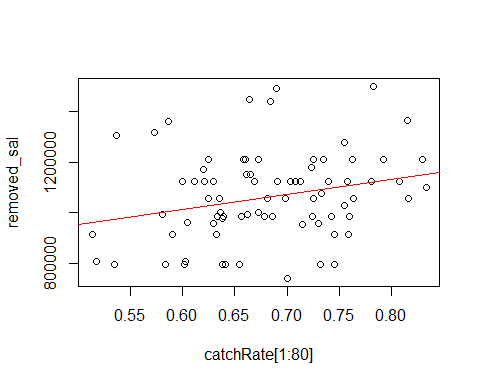
hist(removed\_sal)



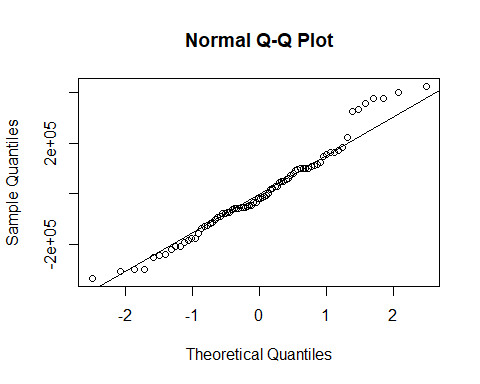
model.rem <- lm(Salary~catchRate, data=data.frame(Salary=Salary[1:80],   
 catchRate=catchRate[1:80]))  
summary(model.rem)

##   
## Call:  
## lm(formula = Salary ~ catchRate, data = data.frame(Salary = Salary[1:80],   
## catchRate = catchRate[1:80]))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -333030 -104864 -16469 99858 423768   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 650055 175966 3.694 0.000408 \*\*\*  
## catchRate 603456 257398 2.344 0.021601 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 172500 on 78 degrees of freedom  
## Multiple R-squared: 0.06583, Adjusted R-squared: 0.05385   
## F-statistic: 5.496 on 1 and 78 DF, p-value: 0.0216

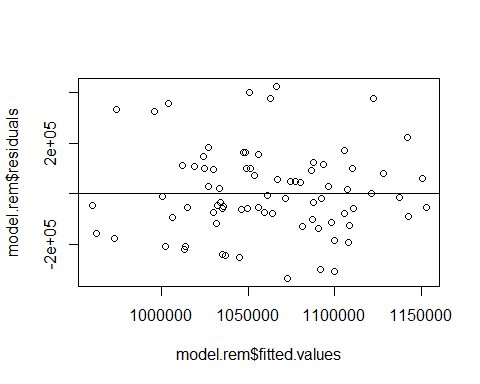
plot(catchRate[1:80], removed\_sal)  
abline(model.rem, col="red")



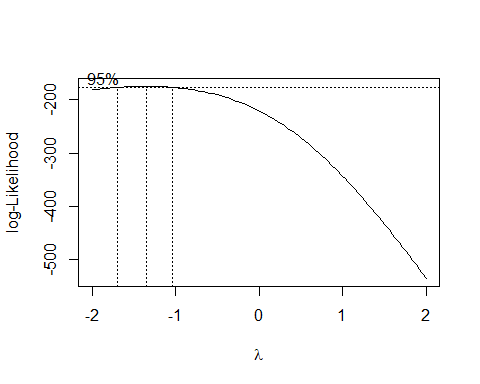
# check residuals  
qqnorm(model.rem$residuals)  
qqline(model.rem$residuals)



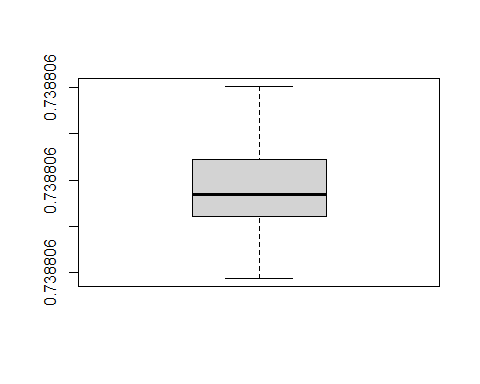
plot(model.rem$fitted.values, model.rem$residuals)  
abline(h=0)



# box-cox transformation  
# https://www.r-bloggers.com/2022/10/box-cox-transformation-in-r/  
MASS::boxcox(model1) # CI in plot does not contain 1, so continue with transform



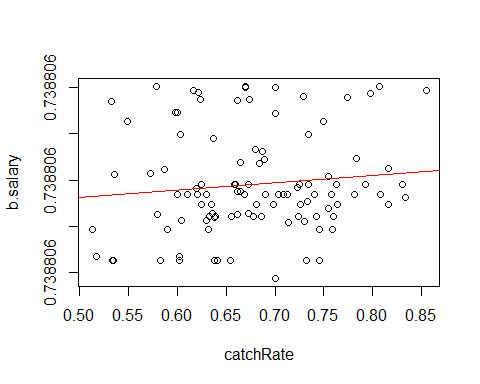
# default = 6  
options(digits=10)  
  
b <- MASS::boxcox(model1)  
lambda <- b$x[which.max(b$y)]  
b.salary <- (Salary^lambda - 1) / lambda  
  
boxplot(b.salary)



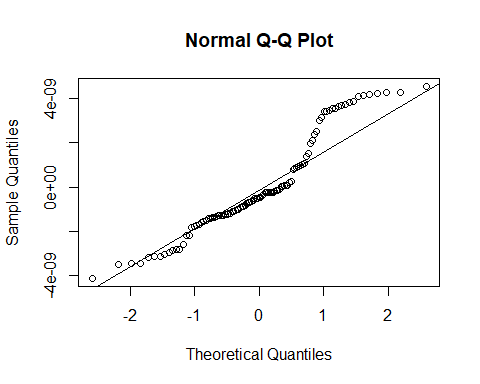
plot(catchRate, b.salary)  
  
b.model <- lm(b.salary~catchRate)  
summary(b.model)

##   
## Call:  
## lm(formula = b.salary ~ catchRate)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -4.127536e-09 -1.320518e-09 -4.538924e-10 1.012803e-09 4.557751e-09   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 7.388060e-01 1.933650e-09 3.820785e+08 < 2e-16 \*\*\*  
## catchRate 3.184746e-09 2.833211e-09 1.124080e+00 0.26359   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 2.212956e-09 on 103 degrees of freedom  
## Multiple R-squared: 0.01211879, Adjusted R-squared: 0.002527708   
## F-statistic: 1.263548 on 1 and 103 DF, p-value: 0.2635935

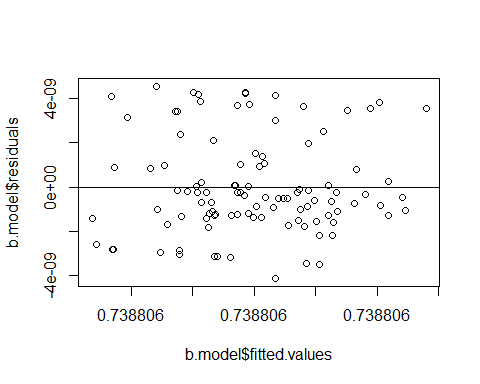
plot(catchRate, b.salary)  
abline(b.model, col="red")



qqnorm(b.model$residuals)  
qqline(b.model$residuals)



plot(b.model$fitted.values, b.model$residuals)  
abline(h=0)



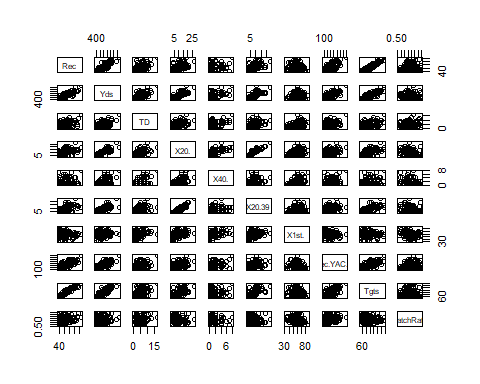
# AIC to compare models  
AIC(model.rem)

## [1] 2160.353852

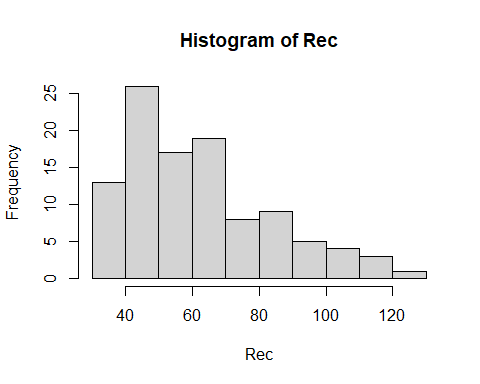
AIC(model1, model2, b.model)

## df AIC  
## model1 3 3491.723178  
## model2 3 3060.668469  
## b.model 3 -3883.118888

# box-cox has lowest AIC, but removed outliers has lowest adjusted R^2  
# (can't compare two models with different # of observations)  
  
  
# OTHER MODELS  
# Explore possible relationships  
# remove Rec.1st, Rec.FUM, Salary (collinear, outliers)  
new\_data <- data %>% select(Rec, Yds, TD, X20., X40., X20.39, X1st.,  
 Rec.YAC.R, Tgts, catchRate)  
pairs(new\_data)



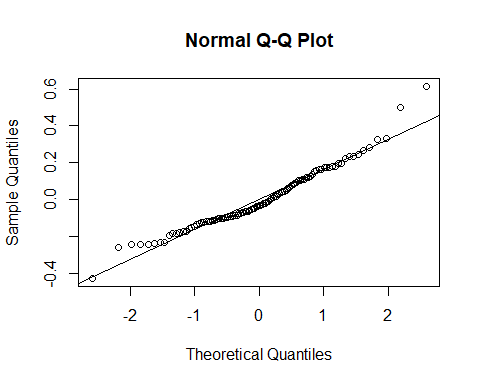
# POISSON MODELS  
# Rec ~ Yds  
hist(Rec)



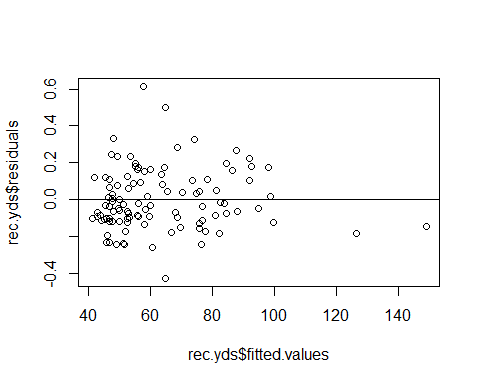
rec.yds <- glm(Rec ~ Yds, family='poisson')  
summary(rec.yds)

##   
## Call:  
## glm(formula = Rec ~ Yds, family = "poisson")  
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 3.395568e+00 3.567148e-02 95.19000 < 2.22e-16 \*\*\*  
## Yds 9.409508e-04 4.001124e-05 23.51716 < 2.22e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for poisson family taken to be 1)  
##   
## Null deviance: 699.18881 on 104 degrees of freedom  
## Residual deviance: 178.31672 on 103 degrees of freedom  
## AIC: 805.25958  
##   
## Number of Fisher Scoring iterations: 4

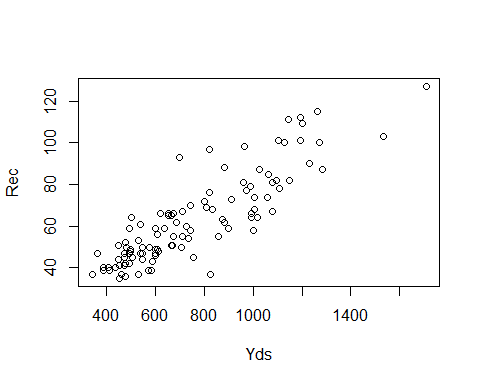
qqnorm(rec.yds$residuals)  
qqline(rec.yds$residuals)



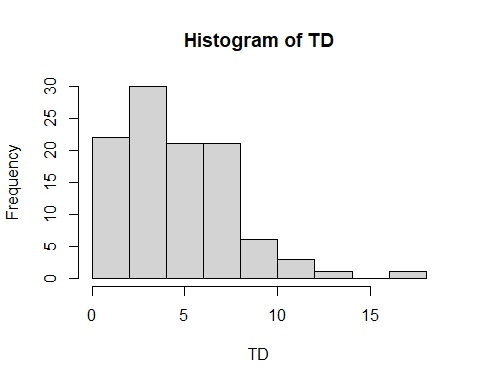
plot(rec.yds$fitted.values, rec.yds$residuals)  
abline(h=0)



plot(Rec~Yds)  
lines(x=Yds, y=predict(rec.yds, data, type = "response")\*10, col="red")



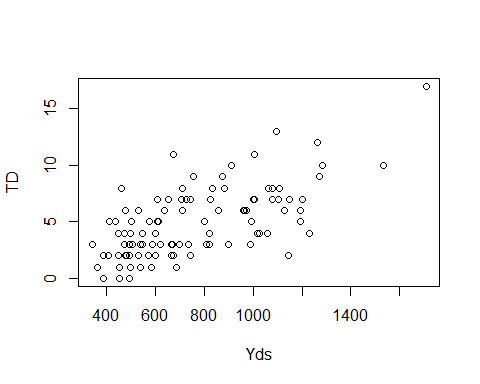
# TD ~ Yds  
hist(TD)



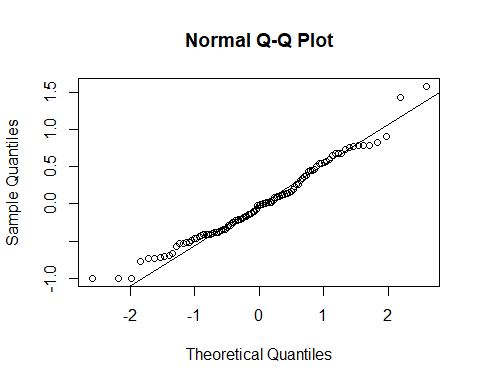
td.yds <- glm(TD ~ Yds, family='poisson')  
summary(td.yds)

##   
## Call:  
## glm(formula = TD ~ Yds, family = "poisson")  
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 0.6359335513 0.1271641190 5.00089 5.7067e-07 \*\*\*  
## Yds 0.0012105638 0.0001384364 8.74455 < 2.22e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for poisson family taken to be 1)  
##   
## Null deviance: 194.67008 on 104 degrees of freedom  
## Residual deviance: 123.60796 on 103 degrees of freedom  
## AIC: 469.51054  
##   
## Number of Fisher Scoring iterations: 4

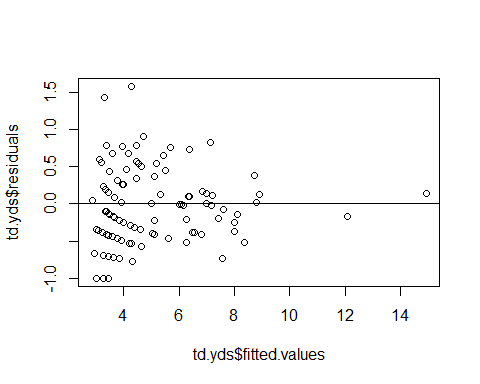
plot(TD~Yds)



qqnorm(td.yds$residuals)  
qqline(td.yds$residuals)



plot(td.yds$fitted.values, td.yds$residuals)  
abline(h=0)



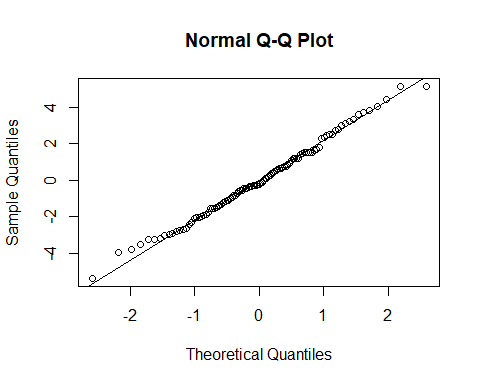
# MULTIPLE REGRESSION  
# step-wise selection on TD  
td\_model\_data <- data[,c(-1, -2, -5, -13)]  
  
full\_td\_model <- lm(TD~., data=td\_model\_data)  
td\_step <- step(full\_td\_model, direction="backward")

## Start: AIC=179.41  
## TD ~ Rec + Yds + X20. + X40. + Rec.1st + X1st. + Rec.FUM + Rec.YAC.R +   
## Tgts + catchRate + X20.39  
##   
##   
## Step: AIC=179.41  
## TD ~ Rec + Yds + X20. + X40. + Rec.1st + X1st. + Rec.FUM + Rec.YAC.R +   
## Tgts + catchRate  
##   
## Df Sum of Sq RSS AIC  
## - X20. 1 0.0608139 470.22290 177.42088  
## - catchRate 1 0.1064450 470.26853 177.43107  
## - Rec 1 0.1394212 470.30150 177.43843  
## - Yds 1 0.1633955 470.32548 177.44378  
## - Rec.FUM 1 0.4744918 470.63657 177.51321  
## - Tgts 1 0.6150827 470.77716 177.54458  
## - X1st. 1 1.2613809 471.42346 177.68862  
## - Rec.YAC.R 1 6.0132150 476.17530 178.74170  
## <none> 470.16208 179.40730  
## - Rec.1st 1 10.1860450 480.34813 179.65783  
## - X40. 1 26.3211704 496.48325 183.12689  
##   
## Step: AIC=177.42  
## TD ~ Rec + Yds + X40. + Rec.1st + X1st. + Rec.FUM + Rec.YAC.R +   
## Tgts + catchRate  
##   
## Df Sum of Sq RSS AIC  
## - catchRate 1 0.0858782 470.30877 175.44006  
## - Rec 1 0.1120674 470.33496 175.44590  
## - Yds 1 0.1140024 470.33690 175.44633  
## - Rec.FUM 1 0.4762984 470.69919 175.52718  
## - Tgts 1 0.6815418 470.90444 175.57296  
## - X1st. 1 1.3912333 471.61413 175.73108  
## - Rec.YAC.R 1 5.9528677 476.17576 176.74180  
## <none> 470.22290 177.42088  
## - Rec.1st 1 11.9277306 482.15063 178.05110  
## - X40. 1 29.7162941 499.93919 181.85524  
##   
## Step: AIC=175.44  
## TD ~ Rec + Yds + X40. + Rec.1st + X1st. + Rec.FUM + Rec.YAC.R +   
## Tgts  
##   
## Df Sum of Sq RSS AIC  
## - Rec 1 0.0328092 470.34158 173.44738  
## - Yds 1 0.1275730 470.43635 173.46853  
## - Rec.FUM 1 0.4404448 470.74922 173.53834  
## - X1st. 1 1.7842743 472.09305 173.83765  
## - Rec.YAC.R 1 5.9268073 476.23558 174.75499  
## <none> 470.30877 175.44006  
## - Tgts 1 11.1455023 481.45428 175.89935  
## - Rec.1st 1 13.1980606 483.50683 176.34604  
## - X40. 1 29.6424832 499.95126 179.85778  
##   
## Step: AIC=173.45  
## TD ~ Yds + X40. + Rec.1st + X1st. + Rec.FUM + Rec.YAC.R + Tgts  
##   
## Df Sum of Sq RSS AIC  
## - Yds 1 0.148313 470.48990 171.48048  
## - Rec.FUM 1 0.458282 470.79987 171.54964  
## - X1st. 1 4.201102 474.54269 172.38108  
## - Rec.YAC.R 1 7.063255 477.40484 173.01247  
## <none> 470.34158 173.44738  
## - Tgts 1 16.534971 486.87655 175.07528  
## - X40. 1 29.864842 500.20642 177.91135  
## - Rec.1st 1 39.323716 509.66530 179.87836  
##   
## Step: AIC=171.48  
## TD ~ X40. + Rec.1st + X1st. + Rec.FUM + Rec.YAC.R + Tgts  
##   
## Df Sum of Sq RSS AIC  
## - Rec.FUM 1 0.420039 470.90993 169.57418  
## - X1st. 1 4.385010 474.87491 170.45456  
## - Rec.YAC.R 1 6.930953 477.42085 171.01599  
## <none> 470.48990 171.48048  
## - Tgts 1 17.471637 487.96153 173.30900  
## - X40. 1 66.442584 536.93248 183.35076  
## - Rec.1st 1 72.364125 542.85402 184.50241  
##   
## Step: AIC=169.57  
## TD ~ X40. + Rec.1st + X1st. + Rec.YAC.R + Tgts  
##   
## Df Sum of Sq RSS AIC  
## - X1st. 1 3.965143 474.87508 168.45460  
## - Rec.YAC.R 1 6.771023 477.68096 169.07318  
## <none> 470.90993 169.57418  
## - Tgts 1 17.097874 488.00781 171.31896  
## - X40. 1 66.731955 537.64189 181.48940  
## - Rec.1st 1 72.198868 543.10880 182.55168  
##   
## Step: AIC=168.45  
## TD ~ X40. + Rec.1st + Rec.YAC.R + Tgts  
##   
## Df Sum of Sq RSS AIC  
## - Rec.YAC.R 1 3.036170 477.91125 167.12379  
## <none> 474.87508 168.45460  
## - Tgts 1 14.534509 489.40959 169.62014  
## - X40. 1 66.840656 541.71573 180.28201  
## - Rec.1st 1 138.995047 613.87012 193.41142  
##   
## Step: AIC=167.12  
## TD ~ X40. + Rec.1st + Tgts  
##   
## Df Sum of Sq RSS AIC  
## <none> 477.91125 167.12379  
## - Tgts 1 21.257984 499.16923 169.69341  
## - X40. 1 64.048302 541.95955 178.32926  
## - Rec.1st 1 136.799092 614.71034 191.55504

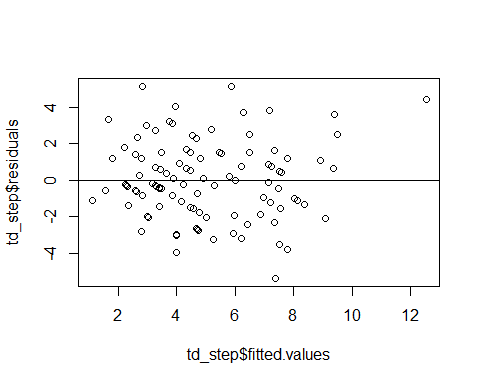
summary(td\_step)

##   
## Call:  
## lm(formula = TD ~ X40. + Rec.1st + Tgts, data = td\_model\_data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -5.3942112 -1.4860737 -0.2230947 1.4788605 5.1601430   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.42349491 0.68588763 0.61744 0.53833370   
## X40. 0.47635633 0.12947654 3.67909 0.00037751 \*\*\*  
## Rec.1st 0.17983999 0.03344703 5.37686 4.9035e-07 \*\*\*  
## Tgts -0.02965444 0.01399077 -2.11957 0.03649323 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 2.175269 on 101 degrees of freedom  
## Multiple R-squared: 0.5000729, Adjusted R-squared: 0.4852236   
## F-statistic: 33.67649 on 3 and 101 DF, p-value: 3.59554e-15

qqnorm(td\_step$residuals)  
qqline(td\_step$residuals)



plot(td\_step$fitted.values, td\_step$residuals)  
abline(h=0)



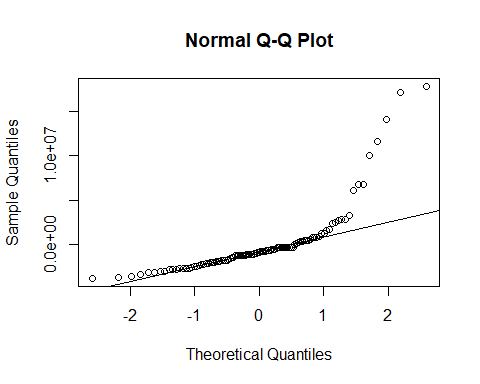
# step-wise selection on Salary  
sal\_model\_data <- data[,c(-1, -2, -13, -14)]  
full\_sal\_model <- lm(Salary~., data=sal\_model\_data)  
  
sal\_step <- step(full\_sal\_model, direction="backward")

## Start: AIC=3199.14  
## Salary ~ Rec + Yds + TD + X20. + X40. + Rec.1st + X1st. + Rec.FUM +   
## Rec.YAC.R + Tgts + X20.39  
##   
##   
## Step: AIC=3199.14  
## Salary ~ Rec + Yds + TD + X20. + X40. + Rec.1st + X1st. + Rec.FUM +   
## Rec.YAC.R + Tgts  
##   
## Df Sum of Sq RSS AIC  
## - Tgts 1 1.1087662e+10 1.4531221e+15 3197.1457  
## - X1st. 1 4.9542905e+11 1.4536064e+15 3197.1807  
## - Rec.1st 1 1.1577690e+12 1.4542688e+15 3197.2285  
## - Rec.FUM 1 2.6792040e+12 1.4557902e+15 3197.3383  
## - X20. 1 6.7623233e+12 1.4598733e+15 3197.6324  
## - Rec 1 8.7908551e+12 1.4619019e+15 3197.7782  
## - X40. 1 2.1082835e+13 1.4741938e+15 3198.6574  
## - Rec.YAC.R 1 2.3267006e+13 1.4763780e+15 3198.8128  
## <none> 1.4531110e+15 3199.1449  
## - Yds 1 3.3562804e+13 1.4866738e+15 3199.5425  
## - TD 1 5.8836878e+13 1.5119479e+15 3201.3126  
##   
## Step: AIC=3197.15  
## Salary ~ Rec + Yds + TD + X20. + X40. + Rec.1st + X1st. + Rec.FUM +   
## Rec.YAC.R  
##   
## Df Sum of Sq RSS AIC  
## - X1st. 1 5.2256847e+11 1.4536447e+15 3195.1835  
## - Rec.1st 1 1.1610853e+12 1.4542832e+15 3195.2296  
## - Rec.FUM 1 2.6849019e+12 1.4558070e+15 3195.3395  
## - X20. 1 6.7568429e+12 1.4598789e+15 3195.6328  
## - Rec 1 1.2559622e+13 1.4656817e+15 3196.0493  
## - X40. 1 2.1074185e+13 1.4741963e+15 3196.6575  
## <none> 1.4531221e+15 3197.1457  
## - Rec.YAC.R 1 2.8305050e+13 1.4814271e+15 3197.1713  
## - Yds 1 3.5154580e+13 1.4882767e+15 3197.6557  
## - TD 1 5.9981424e+13 1.5131035e+15 3199.3928  
##   
## Step: AIC=3195.18  
## Salary ~ Rec + Yds + TD + X20. + X40. + Rec.1st + Rec.FUM + Rec.YAC.R  
##   
## Df Sum of Sq RSS AIC  
## - Rec.FUM 1 2.4102134e+12 1.4560549e+15 3193.3574  
## - X20. 1 7.3355249e+12 1.4609802e+15 3193.7120  
## - Rec.1st 1 1.3254514e+13 1.4668992e+15 3194.1365  
## - X40. 1 2.3397386e+13 1.4770420e+15 3194.8600  
## <none> 1.4536447e+15 3195.1835  
## - Rec.YAC.R 1 2.9314690e+13 1.4829593e+15 3195.2798  
## - Rec 1 3.2129815e+13 1.4857745e+15 3195.4790  
## - Yds 1 3.7794907e+13 1.4914396e+15 3195.8786  
## - TD 1 5.9468621e+13 1.5131133e+15 3197.3935  
##   
## Step: AIC=3193.36  
## Salary ~ Rec + Yds + TD + X20. + X40. + Rec.1st + Rec.YAC.R  
##   
## Df Sum of Sq RSS AIC  
## - X20. 1 7.3770447e+12 1.4634319e+15 3191.8880  
## - Rec.1st 1 1.1381725e+13 1.4674366e+15 3192.1750  
## - X40. 1 2.2342441e+13 1.4783973e+15 3192.9563  
## <none> 1.4560549e+15 3193.3574  
## - Rec.YAC.R 1 2.9492624e+13 1.4855475e+15 3193.4629  
## - Yds 1 3.7210841e+13 1.4932657e+15 3194.0071  
## - Rec 1 3.7855831e+13 1.4939107e+15 3194.0524  
## - TD 1 5.9528373e+13 1.5155832e+15 3195.5647  
##   
## Step: AIC=3191.89  
## Salary ~ Rec + Yds + TD + X40. + Rec.1st + Rec.YAC.R  
##   
## Df Sum of Sq RSS AIC  
## - Rec.1st 1 6.1433963e+12 1.4695753e+15 3190.3279  
## - X40. 1 1.7082901e+13 1.4805148e+15 3191.1066  
## <none> 1.4634319e+15 3191.8880  
## - Rec 1 3.0522633e+13 1.4939545e+15 3192.0555  
## - Rec.YAC.R 1 3.3768282e+13 1.4972002e+15 3192.2834  
## - Yds 1 4.1872069e+13 1.5053040e+15 3192.8501  
## - TD 1 5.8810526e+13 1.5222424e+15 3194.0251  
##   
## Step: AIC=3190.33  
## Salary ~ Rec + Yds + TD + X40. + Rec.YAC.R  
##   
## Df Sum of Sq RSS AIC  
## - X40. 1 1.2250986e+13 1.4818263e+15 3189.1996  
## <none> 1.4695753e+15 3190.3279  
## - Rec 1 4.0719050e+13 1.5102944e+15 3191.1977  
## - Rec.YAC.R 1 4.2304639e+13 1.5118800e+15 3191.3078  
## - Yds 1 4.3079142e+13 1.5126545e+15 3191.3616  
## - TD 1 7.7817089e+13 1.5473924e+15 3193.7457  
##   
## Step: AIC=3189.2  
## Salary ~ Rec + Yds + TD + Rec.YAC.R  
##   
## Df Sum of Sq RSS AIC  
## <none> 1.4818263e+15 3189.1996  
## - Rec 1 2.9951612e+13 1.5117779e+15 3189.3008  
## - Rec.YAC.R 1 3.0873291e+13 1.5126996e+15 3189.3648  
## - Yds 1 3.2863292e+13 1.5146896e+15 3189.5028  
## - TD 1 8.9582982e+13 1.5714093e+15 3193.3628

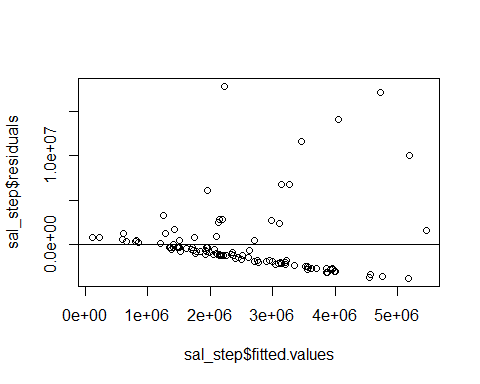
summary(sal\_step)

##   
## Call:  
## lm(formula = Salary ~ Rec + Yds + TD + Rec.YAC.R, data = sal\_model\_data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -3800775.6 -1924664.2 -862109.2 317483.8 17775045.7   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 2155947.586 1194406.671 1.80504 0.074078 .  
## Rec 57096.555 40160.439 1.42171 0.158221   
## Yds -4707.511 3161.073 -1.48921 0.139578   
## TD 405242.122 164816.419 2.45875 0.015660 \*  
## Rec.YAC.R -6001.444 4157.794 -1.44342 0.152028   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 3849450 on 100 degrees of freedom  
## Multiple R-squared: 0.0812707, Adjusted R-squared: 0.04452153   
## F-statistic: 2.211497 on 4 and 100 DF, p-value: 0.07308567

qqnorm(sal\_step$residuals)  
qqline(sal\_step$residuals)



plot(sal\_step$fitted.values, sal\_step$residuals)  
abline(h=0)



# DASHBOARDS  
# Source: https://rstudio.github.io/DT/options.html  
library(DT)

## Warning: package 'DT' was built under R version 4.4.3

# predict salaries from catchRate (using best model 1)  
pred\_salary <- data.frame(Player=Player[1:80],  
 catchRate=round(catchRate[1:80], 3),  
 predicted=round(predict(model.rem, newdata=data.frame(  
 catchRate=catchRate[1:80]))),  
 actual=removed\_sal)  
  
# predict touchdowns (using step-wise model)  
pred\_td <- data.frame(Player=Player, Rec.over.40.yds=X40., Rec.1st=Rec.1st,   
 Tgts=Tgts, predicted=round(predict(td\_step,   
 newdata=data.frame(X40.=X40., Rec.1st=Rec.1st,  
 Tgts=Tgts),type='response')),  
 actual=TD)  
  
# export prediction data  
write.csv(pred\_salary, "./salary\_predictions.csv")  
write.csv(pred\_td, "./td\_predictions.csv")