library(car)

library(alr4)

# 1

golf <- read.csv("/home/thomas/git/datascience/MSDataSci/MATH550/data/pgatour2006.csv",header=TRUE)

# Removing unneeded columns

golf <- subset(golf, select=-c(Name,TigerWoods,AveDrivingDistance,BounceBack))

attach(golf)

golf

pairs(PrizeMoney~., data = golf)

model\_golf <- lm(PrizeMoney ~., data=golf)

summary(model\_golf)

# Running a full summary of a linear model now does not give us the best R-squared value at 0.3843.

# Let's try a transform...

par(mfrow = c(1,1))

invResults\_golf <- invResPlot(model\_golf)

# The Inverse response plot picked out a visually appropriate lambda-hat of about 0.12

# to use as an exponent in the power transform for PrizeMoney

# assigning lambda-hat to a variable

lambdaHat\_golf <- invResults\_golf$lambda[1]

p1 <- powerTransform(PrizeMoney ~.,data=golf)

# Power transforming PrizeMoney

golf\_power <- golf

golf\_power$PrizeMoney <- golf\_power$PrizeMoney^lambdaHat\_golf

golf\_power

attach(golf\_power)

pairs(PrizeMoney~., data = golf\_power)

# The power transformed pairs plot looks much better. Now, a rough linear relationship can be seen

# between PrizeMoney and the predictors with some multicollinearity.

model\_golf\_power <- lm(PrizeMoney ~., data=golf\_power)

summary(model\_golf\_power)

# The adjusted R-squared value for the new power-transformed model is a lot better at 0.5354.

# Now, let's try a Log Transformation on Prize Money

# Log transforming PrizeMoney

golf\_log <- golf

golf\_log$PrizeMoney <- log(golf\_log$PrizeMoney)

golf\_log

attach(golf\_log)

pairs(PrizeMoney~., data = golf\_log)

# The log transformed pairs plot looks much better. Now, a rough linear relationship can be seen

# between PrizeMoney and the predictors.

model\_golf\_log <- lm(PrizeMoney ~., data=golf\_log)

summary(model\_golf\_log)

# The adjusted R-squared value for the new log-transformed model is 0.5412.

# Although the log transformation is slightly better than the power transformation given the

# data provided, The inverse response plot visually looks like the data would be much better

# explained by a power transformation. I will continue with the power transformation

# In the new power transformation pairs plot, there looks to be a lot of multicollinearity.

# Because of this, I will perform backwards step AIC and BIC to reduce the

# number of predictors and potential multicollinearity

n <- length(model\_golf\_power$residuals)

backAIC <- step(model\_golf\_power,direction="backward", data=golf\_power)

backBIC <- step(model\_golf\_power,direction="backward", data=golf\_power, k=log(n))

# The step model results driven by AIC give us back SandSaves, Scrambling, GIR, and BirdieConversion

# as the best predictors.

# For BIC, I am given Scrambling, GIR, and BirdieConversion

# Since there are already so few terms, the BIC approach might not be completely necessary

# It might be better to evaluate the models for each before making a decision on the best

# AIC

model\_golf\_power <- lm(PrizeMoney~SandSaves+Scrambling+GIR+BirdieConversion, data=golf\_power)

summary(model\_golf\_power)

# BIC

model\_golf\_power <- lm(PrizeMoney~Scrambling+GIR+BirdieConversion, data=golf\_power)

summary(model\_golf\_power)

pairs(PrizeMoney~SandSaves+Scrambling+GIR+BirdieConversion, data = golf\_power)

# While the AIC has an additional term, the R-squared is slightly higher than using BIC, yet negligible -

# a difference between 0.5378 (AIC) and 0.5315 (BIC)

# Since I do not expect the span of the addition of another variable from the AIC method over the BIC method

# To be much larger, I will default to the predictors picked by the back step BIC method.

# I do this because, since the influence of an additional predictor is very small, I do not want to run the

# risk of overfitting the data on that extra variable now and causing the model to falter with future data.

par(mfrow = c(2,2))

mmp(model\_golf\_power)

mmp(model\_golf\_power,Scrambling)

mmp(model\_golf\_power,GIR)

mmp(model\_golf\_power,BirdieConversion)

# The Marginal Model plots for the model and the individual predictors are looking good!

# The model is tracking well wit the data.

StanRes <- rstandard(model\_golf\_power)

par(mfrow = c(2,2))

plot(Scrambling,StanRes, ylab="Standardized Residuals",pch=19,col=10)

abline(h=0,lwd=2)

plot(GIR,StanRes, ylab="Standardized Residuals",pch=19,col=10)

abline(h=0,lwd=2)

plot(BirdieConversion,StanRes, ylab="Standardized Residuals",pch=19,col=10)

abline(h=0,lwd=2)

# The standard residuals also are looking nice and flat.

# If I had to investigate the model further, I would look into Outlier Detection.

# There are a few points that look like they might be providing slightly more

# leverage than others and may need to be addressed.

# 2

dwaste <- read.csv("/home/thomas/git/datascience/MSDataSci/MATH550/data/dwaste.csv", header=TRUE)

# removing Day

dwaste <- subset(dwaste, select=-c(Day))

attach(dwaste)

pairs(O2UP~., data = dwaste)

# Already, I see that a transformation of at least the O2UP variable might be needed.

# This assupmtion is just based onvisual inspectiona and will be investigated later.

model <- lm(O2UP ~., data=dwaste)

summary(model)

# Running a full summary of a linear model now does not give us the best R-squared value at 0.2609.

# Let's try a transform...

par(mfrow = c(1,1))

invResults\_dwaste <- invResPlot(model)

# The Inverse response plot picked out a visually appropriate lambda-hat of about 0.04

# to use as an exponent in the power transform for O2UP

# assigning lambda-hat to a variable

lambdaHat\_dwaste <- invResults\_dwaste$lambda[1]

p1 <- powerTransform(O2UP ~.,data=dwaste)

# Power transforming O2UP

dwaste\_power <- dwaste

dwaste\_power$O2UP <- dwaste\_power$O2UP^lambdaHat\_dwaste

dwaste\_power

attach(dwaste\_power)

pairs(O2UP~., data = dwaste\_power)

# The new pairs plot looks much better. Now, a rough linear relationship can be seen

# between O2UP and the predictors.

model\_dwaste\_power <- lm(O2UP ~., data=dwaste\_power)

summary(model\_dwaste\_power)

# The adjusted R-squared value for the new power-transformed model is a lot better at 0.715.

# Log transforming O2UP

dwaste\_log <- dwaste

dwaste\_log$O2UP <- log(dwaste\_log$O2UP)

dwaste\_log

attach(dwaste\_log)

pairs(O2UP~., data = dwaste\_log)

# The new pairs plot looks much better. Now, a rough linear relationship can be seen

# between O2UP and the predictors.

model\_dwaste\_log <- lm(O2UP ~., data=dwaste\_log)

summary(model\_dwaste\_log)

# The adjusted R-squared value for the new power-transformed model is a lot better at 0.7413.

# the log-transform performs slightly better than the power transform.

# I will be continuing with the log transform model.

# In the problem summary (and as can be seen in the new pairs plot),

# it alluded that there were several predictors that coud be correlated.

# because of this, Iwill perform backwards step AIC and BIC to reduce the

# number of predictors and potential multicollinearity

n <- length(model\_dwaste\_log$residuals)

backAIC <- step(model\_dwaste\_log,direction="backward", data=dwaste\_log)

backBIC <- step(model\_dwaste\_log,direction="backward", data=dwaste\_log, k=log(n))

# For both the AIC and BIC bact step methods, I am given TS and COD

# This simplifies the decision making for which method to choose because they both agree.

# AIC & BIC

model\_dwaste\_log <- lm(O2UP~TS+COD, data=dwaste\_log)

summary(model\_dwaste\_log)

pairs(O2UP~TS+COD, data = dwaste\_log)

# When I go back to look at the problem statement, it does not specifically mention

# any previously understood relationship between TS, and COD.

# This suggests that perhaps the span of these two predictors will still be appropriate

# for a model. So, I will be using the predictors given to us by the AIC and BIC step results.

par(mfrow = c(2,2))

mmp(model\_dwaste\_log)

mmp(model\_dwaste\_log,TS)

mmp(model\_dwaste\_log,COD)

# the Marginal Model plots for the model and the individual predictors

# look good. The data tracks well with the proposed model

StanRes <- rstandard(model\_dwaste\_log)

par(mfrow = c(2,2))

plot(TS,StanRes, ylab="Standardized Residuals",pch=19,col=10)

abline(h=0,lwd=2)

plot(COD,StanRes, ylab="Standardized Residuals",pch=19,col=10)

abline(h=0,lwd=2)

# The standard residuals also are looking nice and flat.

# If I had to investigate the model further, I would look into Outlier Detection.

# There are a few points that look like they might beproviding much more leverage

# than others and may need to be addressed.