######### APPENDIX ##################

######### ASIGNMENT 2 R CODE ##########

load("/Users/davidkirui/Desktop/PhD work/Year 3/Spring 2017/STAT 974 - Modern Regression/Assignments/Assignment 2 (First Real)/Homeless.rdata")

library("ggplot2")

library("dplyr")

library("car")

install.packages("stargazer")

library(stargazer)

library(DAAG)

library(glmnet)

library(mgcv) #make sure to NOT load this with gam

str(Homeless)

class(Homeless)

##CLEANING DATA:

NA\_PropVacant <- which(is.na(Homeless$PropVacant))

Homeless <- Homeless[-NA\_PropVacant,] #getting rid of NAs, don't do this twice

#summary(Homeless$PropVacant)

#summary(Homeless)

#CONVERTING PROPORTIONS TO PERCENTAGES FOR PROP. MINORITY AND PROP VACANT

PctVacant <- c()

Homeless$PctVacant <- (Homeless$PropVacant)\*100

PctMinority <- c()

Homeless$PctMinority <- (Homeless$PropMinority)\*100

#head(Homeless)

#summary(Homeless) #problem statement: consider the correlates of homelessness in census tracts in los angeles country, end goal is an estimate of homeless population in each census tract

#in data description, you must tell your reader what the variables are, must be specific.

# looking at univaraite statistics; these data are from los angeles county

#stargazer(Homeless, type = "text")

#summary(Homeless$StreetTotal) # Notes from review: notice that the range is extraordinary, these numbers are probably sensible. First quartile: homelessness is not common in large numbers; we know that the distribution is skewed to the right

# problem in LA with homelessness is concentrated in a few census tracts

#ggplot(Homeless, aes(x="",y=StreetTotal)) + geom\_boxplot()

#ggplot(Homeless, aes(x=StreetTotal)) + geom\_histogram()

#summary(Homeless$MedianIncome) #median income of zero is non sensical - but could be completely industrial (we need to learn more about this zero value and if it's true)

#ggplot(Homeless, aes(x=MedianIncome)) + geom\_histogram()

#ggplot(Homeless, aes(x="",y=MedianIncome)) + geom\_boxplot()

#qqnorm(Homeless$MedianIncome)

#qqline(Homeless$MedianIncome) #assessing normality

Homeless <- Homeless %>%

mutate(LogMedianIncome = log(MedianIncome)) #logging income

summary(Homeless)

#ggplot(Homeless, aes(x=LogMedianIncome)) + geom\_histogram() #logged income

#ggplot(Homeless, aes(x="",y=LogMedianIncome)) + geom\_boxplot() #logged income

#summary(Homeless$PropVacant)

#ggplot(Homeless, aes(x=PctVacant)) + geom\_histogram()

#ggplot(Homeless, aes(x="",y=PctVacant)) + geom\_boxplot()

#summary(Homeless$PctMinority) #defined as underrepresented minorities

#ggplot(Homeless, aes(x=PropMinority)) + geom\_histogram()

#ggplot(Homeless, aes(x=PctMinority)) + geom\_histogram() #bimodal distribution - closer to uniform than normal

#ggplot(Homeless, aes(x="",y=PctMinority)) + geom\_boxplot()

#qqnorm(Homeless$PropMinority)

#qqline(Homeless$PropMinority)

#summary(Homeless$PctCommercial) #land use data in most major cities and you can get the proportion of the land area that is zoned commercial.

#ggplot(Homeless, aes(x=PctCommercial)) + geom\_histogram()

#ggplot(Homeless, aes(x="",y=PctCommercial)) + geom\_boxplot()

#boxplot(Homeless$PctCommercial)

#summary(Homeless$PctIndustrial)

#ggplot(Homeless, aes(x=PctIndustrial)) + geom\_histogram()

#ggplot(Homeless, aes(x="",y=PctIndustrial)) + geom\_boxplot()

####DESCRIPTIVE STATISTICS####

#colnames(Homeless)

#HomelessShort <- Homeless[,-c(2:4)]

HomelessShort <- Homeless[,-c(3:4,9)]

#colnames(HomelessShort)

stargazer(HomelessShort, type = "latex", median = TRUE) #

#CHANGE THE DATA FRMAE USED HEREstargazer(HomelessShort, type = "latex", title = "Descriptive Statistics", median = TRUE) #PUT TABLE IN PAPER

#DESIGNATING THE MULTI-PLOT FUNCTION

multiplot <- function(..., plotlist=NULL, file, cols=1, layout=NULL) {

library(grid)

# Make a list from the ... arguments and plotlist

plots <- c(list(...), plotlist)

numPlots = length(plots)

# If layout is NULL, then use 'cols' to determine layout

if (is.null(layout)) {

# Make the panel

# ncol: Number of columns of plots

# nrow: Number of rows needed, calculated from # of cols

layout <- matrix(seq(1, cols \* ceiling(numPlots/cols)),

ncol = cols, nrow = ceiling(numPlots/cols))

}

if (numPlots==1) {

print(plots[[1]])

} else {

# Set up the page

grid.newpage()

pushViewport(viewport(layout = grid.layout(nrow(layout), ncol(layout))))

# Make each plot, in the correct location

for (i in 1:numPlots) {

# Get the i,j matrix positions of the regions that contain this subplot

matchidx <- as.data.frame(which(layout == i, arr.ind = TRUE))

print(plots[[i]], vp = viewport(layout.pos.row = matchidx$row,

layout.pos.col = matchidx$col))

}

}

}

#multi-plot for univariate statistics #USE HOMELESS NOT HOMELESSSHORT BECAUSE NO MEDIAN INCOME IN HOMELESS

One.p1 <- ggplot(Homeless, aes(x=StreetTotal)) + geom\_histogram() +

ggtitle("Distribution of Homeless Persons") + theme(plot.title=element\_text(hjust=0.5))

One.p2 <- ggplot(Homeless, aes(x=MedianIncome)) + geom\_histogram() +

ggtitle("Distribution of Median Income") + theme(plot.title=element\_text(hjust=0.5))

One.p3 <- ggplot(Homeless, aes(x=PctVacant)) + geom\_histogram() +

ggtitle("Distribution of Percent Vacant") + theme(plot.title=element\_text(hjust=0.5))

One.p4 <- ggplot(Homeless, aes(x=PctMinority)) + geom\_histogram() +

ggtitle("Distribution of Percent Minority") + theme(plot.title=element\_text(hjust=0.5))

One.p5 <- ggplot(Homeless, aes(x=PctCommercial)) + geom\_histogram() +

ggtitle("Distribution of Percent Commercial") + theme(plot.title=element\_text(hjust=0.5))

One.p6 <- ggplot(Homeless, aes(x=PctIndustrial)) + geom\_histogram() +

ggtitle("Disribution of Percent Industrial") + theme(plot.title=element\_text(hjust=0.5))

multiplot(One.p1, One.p2, One.p3, One.p4, One.p5, One.p6, cols=2) #univariate plots

#summary(HomelessShort$MedianIncome)

#tail(sort(HomelessShort$MedianIncome)) #looking at the last three observations, last three are 200,001 but this could be an issue with the way in which the data were collected.

#### BIVARIATE RELATIONSHIPS####

cor(HomelessShort, y = NULL, use = "complete.obs", method = "pearson") #CORRELATION TABLE

# Multiple plot function

#

# ggplot objects can be passed in ..., or to plotlist (as a list of ggplot objects)

# - cols: Number of columns in layout

# - layout: A matrix specifying the layout. If present, 'cols' is ignored.

#

# If the layout is something like matrix(c(1,2,3,3), nrow=2, byrow=TRUE),

# then plot 1 will go in the upper left, 2 will go in the upper right, and

# 3 will go all the way across the bottom.

#

# Homeless X Income:

#ggplot(HomelessShort, aes(x=MedianIncome, y=StreetTotal)) + geom\_point() + geom\_smooth(method = 'lm', se = FALSE)

p1 <- ggplot(Homeless, aes(x=MedianIncome, y=StreetTotal)) + geom\_point() +

geom\_smooth(method = 'loess', se = TRUE) + ggtitle("Homelessness by Median Income") +

theme(plot.title=element\_text(hjust=0.5))

#Homeless X Log Income

#ggplot(HomelessShort, aes(x=LogMedianIncome, y=StreetTotal)) + geom\_point() + geom\_smooth(method = 'lm', se = TRUE)

p2 <- ggplot(Homeless, aes(x=LogMedianIncome, y=StreetTotal)) + geom\_point() +

geom\_smooth(method = 'loess', se = TRUE) + ggtitle("Homelessness by Log(Median Income) ") +

theme(plot.title=element\_text(hjust=0.5))

#Homeless X PctVacant

#ggplot(HomelessShort, aes(x=PropVacant, y=StreetTotal)) + geom\_point() + geom\_smooth(method = "lm", se = TRUE)

p3 <- ggplot(Homeless, aes(x=PctVacant, y=StreetTotal)) + geom\_point() +

geom\_smooth(method = "loess", se = TRUE) + ggtitle("Homelessness by Percent Vacant") +

theme(plot.title=element\_text(hjust=0.5))

#Homeless X PctMinority

#ggplot(HomelessShort, aes(x=PropMinority, y=StreetTotal)) + geom\_point() + geom\_smooth(method = "lm", se = TRUE)

p4 <- ggplot(Homeless, aes(x=PctMinority, y=StreetTotal)) + geom\_point() +

geom\_smooth(method = "loess", se = FALSE) + ggtitle("Homelessness by Percent Minority") +

theme(plot.title=element\_text(hjust=0.5))

#Homeless X PctCommercial

#ggplot(HomelessShort, aes(x=PctCommercial, y=StreetTotal)) + geom\_point() + geom\_smooth(method = "lm", se = TRUE)

p5 <- ggplot(Homeless, aes(x=PctCommercial, y=StreetTotal)) + geom\_point() +

geom\_smooth(method = "loess", se = TRUE) + ggtitle("Homelessness by Percent Commercial") +

theme(plot.title=element\_text(hjust=0.5))

#Homeless X PctIndustrial

#ggplot(HomelessShort, aes(x=PctIndustrial, y=StreetTotal)) + geom\_point() + geom\_smooth(method = "lm", se = TRUE)

p6 <- ggplot(Homeless, aes(x=PctIndustrial, y=StreetTotal)) + geom\_point() +

geom\_smooth(method = "loess", se = TRUE) + ggtitle("Homelessness by Percent Industrial") +

theme(plot.title=element\_text(hjust=0.5))

multiplot(p1, p2, p3, p4, p5, p6, cols=2)

#colnames(Homeless)

pairs(HomelessShort,panel = panel.smooth) #scatterplot matrix - street total is horizontal

#when you graph two variables, the axes must include the full range of values

#if the red lines represent what's typical. if we're trying to study homelessness and homelessness is an outlier the smoothers don't capture the homelessness that's clustered at the low end of the distribution

#for subject matter reasons, it looks like we have a problem here.

#head(HomelessShort)

#head(HomelessShort)

summary(Homeless)

##### MULTIVARIATE ANALYSES ####

set.seed(25) #set random number seed

#creating Training, Evaluation, and Test data:

TrainIndex<-sample(1:505,168) # random row indices

Train<-Homeless[TrainIndex,] # select those rows

temp<-Homeless[-TrainIndex,] # not those rows, hence the minus command.

EvalIndex<-sample(1:337,168) # other row indices

Eval<-temp[EvalIndex,] # select those rows

Test<-temp[-EvalIndex,] # not those rows

summary(Train) #summary statistics for all three datasets:

#nrow(Train)

summary(Eval) #summary statistics for all three datasets:

#nrow(Eval)

summary(Test) #summary statistics for all three datasets:

#nrow(Test)

#OLS model

OLSmod <- lm(StreetTotal ~ LogMedianIncome + PctVacant + PctMinority + PctCommercial + PctIndustrial, data = Train)

summary(OLSmod)

#plotting residuals

plot(OLSmod$fitted.values, OLSmod$residuals, pch=16)

abline(h = 0, lwd = 5, col="red")

identify(OLSmod$fitted.values, OLSmod$residuals, labels=Train$StreetTotal, pos=3)

which(Train$StreetTotal == 666) # some huge residual outliers, biggest is observation 129 with 666 Homeless ppl

Train[164,]

plot(OLSmod)

qqnorm(OLSmod$residuals) # a reasonably well fitted line implies normal residuals

qqline(OLSmod$residuals)

summary(OLSmod)

#Ridge Regression (incomplete code - pg. 76)

#X.mat <- model.matrix(StreetTotal~., data = Train)[, -1] #code for GLMNET

#Y <- Train$StreetTotal

#ridge <- glmnet(X.mat,Y,family="gaussian", alpha = 0)

#plot(log(ridge$lambda), ridge$beta[2,],ylim=c(-2,.2), type="1",

# col="blue", lwd=3 ,xlab="Log of Lambda", ylab="Regression Coefficient",

# main="Path of Regression Coefficients by Log of Lambda")

#lines(log(ridge$lambda), ridge$beta)

#INITIAL MODEL IN TRAINING WITH ALL COVIARIATES SMOOTHED

out1 <- gam(StreetTotal~s(LogMedianIncome)+s(PctVacant)+s(PctMinority)+s(PctCommercial)+s(PctIndustrial), family=gaussian, data = Train) # sp chosen empirically, we need to go back and specify the sp parameter

summary(out1)

#Family: gaussian

#Link function: identity

#Formula:

# StreetTotal ~ s(LogMedianIncome) + s(PctVacant) + s(PctMinority) +

# s(PctCommercial) + s(PctIndustrial)

#Parametric coefficients:

#Estimate Std. Error t value Pr(>|t|)

#(Intercept) 37.613 3.182 11.82 <2e-16 \*\*\*

#---

#Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

#Approximate significance of smooth terms:

#edf Ref.df F p-value

#s(LogMedianIncome) 1.000 1.000 4.010 0.0471 \*

#s(PctVacant) 7.371 8.320 6.578 1.49e-07 \*\*\*

#s(PctMinority) 1.976 2.468 1.744 0.1292

#s(PctCommercial) 8.978 8.999 16.685 < 2e-16 \*\*\*

#s(PctIndustrial) 8.850 8.988 22.814 < 2e-16 \*\*\*

#---

#Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

#R-sq.(adj) = 0.751 Deviance explained = 79.3%

#GCV = 2058.9 Scale est. = 1701.3 n = 168

plot(out1)

out1$sp #INITIAL SPs BEFORE TREATING LOGINCOME AND PCTMINORITY AS LINEAR

#s(LogMedianIncome) s(PctVacant) s(PctMinority) s(PctCommercial) s(PctIndustrial)

# 1.767365e+10 5.340752e-04 2.044080e+00 3.508346e-05 2.863822e-04

help(gamObject)

hist(fitted.values(out1),breaks=20)

#SWITCHING TO EVALUATION DATA

preds<-predict(out1, newdata = Eval) # do this with evaluation data

hist(preds, breaks = 50) # what does the distribution look like? LARGE OUTLYING NEGATIVE RESIDUALS

resids<-Eval$StreetTotal-preds # get residuals

var(resids) # MSE for the residuals as a measure of fit

#[1] 21259.26

sd(resids) #average deviation around the fitted values

#[1] 145.8056

#SECOND MODEL IN TRAINING DATA WITH TRAINING DATA

out2 <- gam(StreetTotal~LogMedianIncome + s(PctVacant) + s(PctMinority) +

s(PctCommercial) + s(PctIndustrial), family=gaussian, data = Train) ## new model with PctCommercial and PctMinority smoothed

# (because edfs were 1 in the previous model, indicating linearity)

summary(out2) #SAME RESULTS AS ABOVE FOR THE SMOOTHED COVARIATES

#Family: gaussian

#Link function: identity

#Formula:

# StreetTotal ~ LogMedianIncome + s(PctVacant) + s(PctMinority) +

# s(PctCommercial) + s(PctIndustrial)

#Parametric coefficients:

# Estimate Std. Error t value Pr(>|t|)

#(Intercept) 301.21 131.67 2.288 0.0237 \*

#LogMedianIncome -25.10 12.53 -2.003 0.0472 \*

#---

#Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

#Approximate significance of smooth terms:

# edf Ref.df F p-value

#s(PctVacant) 7.371 8.320 6.578 1.49e-07 \*\*\*

#s(PctMinority) 1.976 2.468 1.744 0.129

#s(PctCommercial) 8.978 8.999 16.685 < 2e-16 \*\*\*

#s(PctIndustrial) 8.850 8.988 22.814 < 2e-16 \*\*\*

#---

#Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

#R-sq.(adj) = 0.751 Deviance explained = 79.3%

#GCV = 2058.9 Scale est. = 1701.3 n = 168

plot(out2)

out2$sp #SP VALUES STAY THE SAME FOR SMOOTHED PREDICTORS AS IN THE FIRST MODEL

#s(PctVacant) s(PctMinority) s(PctCommercial) s(PctIndustrial)

#5.340745e-04 2.044080e+00 3.508956e-05 2.863823e-04

hist(fitted.values(out2),breaks=20)

#SWITCHING TO EVALUATION DATA

preds<-predict(out2, newdata = Eval) # do this with evaluation data

hist(preds, breaks = 50) # what does the distribution look like? LARGE OUTLYING NEGATIVE PREDICTED VALUES

resids<-Eval$StreetTotal-preds # get residuals

var(resids) # MSE for the residuals as a measure of fit

#[1] 21259.26

sd(resids) #average deviation around the fitted values

#[1] 145.8056

out3 <- gam(StreetTotal~LogMedianIncome + s(PctVacant) + PctMinority +

s(PctCommercial) + s(PctIndustrial), family=gaussian, data = Train)

summary(out3) #OUTPUT CHANGES A LITTLE BIT WHEN TREATING PCTMINORITY AS LINEAR

#Family: gaussian

#Link function: identity

#Formula:

# StreetTotal ~ LogMedianIncome + s(PctVacant) + PctMinority +

# s(PctCommercial) + s(PctIndustrial)

#Parametric coefficients:

# Estimate Std. Error t value Pr(>|t|)

#(Intercept) 278.3215 136.7863 2.035 0.0438 \*

#LogMedianIncome -21.4823 12.4705 -1.723 0.0872 .

#PctMinority -0.2916 0.2064 -1.413 0.1600

#---

#Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

#Approximate significance of smooth terms:

# edf Ref.df F p-value

#s(PctVacant) 7.451 8.378 7.129 2.83e-08 \*\*\*

#s(PctCommercial) 8.984 9.000 16.952 < 2e-16 \*\*\*

#s(PctIndustrial) 8.834 8.987 22.139 < 2e-16 \*\*\*

#---

#Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

#R-sq.(adj) = 0.746 Deviance explained = 78.7%

#GCV = 2090.9 Scale est. = 1739.1 n = 168

plot(out3)

out3$sp

#s(PctVacant) s(PctCommercial) s(PctIndustrial)

#4.932779e-04 2.520589e-05 3.184400e-04

hist(fitted.values(out3),breaks=20)

#SWITCHING TO EVALUATION DATA

preds<-predict(out3, newdata = Eval) # do this with evaluation data

hist(preds, breaks = 50) # what does the distribution look like? LARGE OUTLYING NEGATIVE PREDICTED VALUES

resids<-Eval$StreetTotal-preds # get residuals

var(resids) # MSE for the residuals as a measure of fit

#[1] 22110.78

sd(resids) #average deviation around the fitted values

#[1] 148.6969

#########ASSIGNING SP VALUES #########

#USING OUTPUT FROM MODEL 2 (MODEL WITH INCOME TREATED AS LINEAR) - USE FACTOR OF 20 FOR THE FINAL MODEL

sp.vac <- (out2$sp[1])\*10

sp.min <- (out2$sp[2])\*10

sp.com <- (out2$sp[3])\*10

sp.ind <- (out2$sp[4])\*10

#USING OUTPUT FROM MODEL 3 (MODEL WITH INCOME AND PCTMINORITY TREATED AS LINEAR)

#sp.vac.alt <- (out3$sp[1])\*20

#sp.com.alt <- (out3$sp[2])\*20

#sp.ind.alt <- (out3$sp[3])\*20

out4 <- gam(StreetTotal~LogMedianIncome + s(PctVacant, sp = sp.vac) + PctMinority +

s(PctCommercial, sp = sp.com) + s(PctIndustrial, sp = sp.ind), family=gaussian, data = Train) # OPTIMAL MODEL - OUT4, RUN WITH THE TEST DATA BELOW

summary(out4)

plot(out4)

hist(fitted.values(out4),breaks=20)

#SWITCHING TO EVALUATION DATA

preds<-predict(out4, newdata = Eval) # do this with evaluation data

hist(preds, breaks = 50) # what does the distribution look like? LARGE OUTLYING NEGATIVE PREDICTED VALUES

resids<- Eval$StreetTotal-preds # get residuals

var(resids) # MSE for the residuals as a measure of fit

#[1] 15127.37

sd(resids) #average deviation around the fitted values

#[1] 122.9934

#### ITERATION PROCESS ####

sp.vac <- (out2$sp[1])\*20

sp.min <- (out2$sp[2])\*20

sp.com <- (out2$sp[3])\*20

sp.ind <- (out2$sp[4])\*20

out7 <- gam(StreetTotal~LogMedianIncome + s(PctVacant, sp = sp.vac) + PctMinority +

s(PctCommercial, sp = sp.com) + s(PctIndustrial, sp = sp.ind), family=gaussian, data = Train) # OPTIMAL MODEL - OUT4, RUN WITH THE TEST DATA BELOW

summary(out7)

hist(fitted.values(out7),breaks=20)

#SWITCHING TO EVALUATION DATA

preds<-predict(out7, newdata = Eval) # do this with evaluation data

hist(preds, breaks = 50) # what does the distribution look like? LARGE OUTLYING NEGATIVE PREDICTED VALUES

resids<- Eval$StreetTotal-preds # get residuals

var(resids) # MSE for the residuals as a measure of fit

sd(resids) #average deviation around the fitted values

#########ASSIGNING SP VALUES #########

#USING OUTPUT FROM MODEL 2 (MODEL WITH INCOME TREATED AS LINEAR) - USE FACTOR OF 20 FOR THE FINAL MODEL

sp.vac <- (out2$sp[1])\*30

sp.min <- (out2$sp[2])\*30

sp.com <- (out2$sp[3])\*30

sp.ind <- (out2$sp[4])\*30

out4 <- gam(StreetTotal~LogMedianIncome + s(PctVacant, sp = sp.vac) + PctMinority +

s(PctCommercial, sp = sp.com) + s(PctIndustrial, sp = sp.ind), family=gaussian, data = Train)

summary(out4)

plot(out4)

hist(fitted.values(out4),breaks=20)

#SWITCHING TO EVALUATION DATA

preds<-predict(out4, newdata = Eval) # do this with evaluation data

hist(preds, breaks = 50) # what does the distribution look like? LARGE OUTLYING NEGATIVE PREDICTED VALUES

resids<- Eval$StreetTotal-preds # get residuals

var(resids) # MSE for the residuals as a measure of fit

sd(resids) #average deviation around the fitted values

#########ASSIGNING SP VALUES #########

#USING OUTPUT FROM MODEL 2 (MODEL WITH INCOME TREATED AS LINEAR) - USE FACTOR OF 20 FOR THE FINAL MODEL

sp.vac <- (out2$sp[1])\*40

sp.min <- (out2$sp[2])\*40

sp.com <- (out2$sp[3])\*40

sp.ind <- (out2$sp[4])\*40

out4 <- gam(StreetTotal~LogMedianIncome + s(PctVacant, sp = sp.vac) + PctMinority +

s(PctCommercial, sp = sp.com) + s(PctIndustrial, sp = sp.ind), family=gaussian, data = Train)

summary(out4)

plot(out4)

hist(fitted.values(out4),breaks=20)

#SWITCHING TO EVALUATION DATA

preds<-predict(out4, newdata = Eval) # do this with evaluation data

hist(preds, breaks = 50) # what does the distribution look like? LARGE OUTLYING NEGATIVE PREDICTED VALUES

resids<- Eval$StreetTotal-preds # get residuals

var(resids) # MSE for the residuals as a measure of fit

sd(resids) #average deviation around the fitted values

#########ASSIGNING SP VALUES #########

#USING OUTPUT FROM MODEL 2 (MODEL WITH INCOME TREATED AS LINEAR) - USE FACTOR OF 20 FOR THE FINAL MODEL

sp.vac <- (out2$sp[1])\*50

sp.min <- (out2$sp[2])\*50

sp.com <- (out2$sp[3])\*50

sp.ind <- (out2$sp[4])\*50

out4 <- gam(StreetTotal~LogMedianIncome + s(PctVacant, sp = sp.vac) + PctMinority +

s(PctCommercial, sp = sp.com) + s(PctIndustrial, sp = sp.ind), family=gaussian, data = Train)

summary(out4)

plot(out4)

hist(fitted.values(out4),breaks=20)

#SWITCHING TO EVALUATION DATA

preds<-predict(out4, newdata = Eval) # do this with evaluation data

hist(preds, breaks = 50) # what does the distribution look like? LARGE OUTLYING NEGATIVE PREDICTED VALUES

resids<- Eval$StreetTotal-preds # get residuals

var(resids) # MSE for the residuals as a measure of fit

sd(resids) #average deviation around the fitted values

#########ASSIGNING SP VALUES #########

#USING OUTPUT FROM MODEL 2 (MODEL WITH INCOME TREATED AS LINEAR) - USE FACTOR OF 20 FOR THE FINAL MODEL

sp.vac <- (out2$sp[1])\*100

sp.min <- (out2$sp[2])\*100

sp.com <- (out2$sp[3])\*100

sp.ind <- (out2$sp[4])\*100

out4 <- gam(StreetTotal~LogMedianIncome + s(PctVacant, sp = sp.vac) + PctMinority +

s(PctCommercial, sp = sp.com) + s(PctIndustrial, sp = sp.ind), family=gaussian, data = Train)

summary(out4)

plot(out4)

hist(fitted.values(out4),breaks=20)

#SWITCHING TO EVALUATION DATA

preds<-predict(out4, newdata = Eval) # do this with evaluation data

hist(preds, breaks = 50) # what does the distribution look like? LARGE OUTLYING NEGATIVE PREDICTED VALUES

resids<- Eval$StreetTotal-preds # get residuals

var(resids) # MSE for the residuals as a measure of fit

sd(resids) #average deviation around the fitted values

#########ASSIGNING SP VALUES #########

#USING OUTPUT FROM MODEL 2 (MODEL WITH INCOME TREATED AS LINEAR) - USE FACTOR OF 20 FOR THE FINAL MODEL

sp.vac <- (out2$sp[1])\*500

sp.min <- (out2$sp[2])\*500

sp.com <- (out2$sp[3])\*500

sp.ind <- (out2$sp[4])\*500

out4 <- gam(StreetTotal~LogMedianIncome + s(PctVacant, sp = sp.vac) + PctMinority +

s(PctCommercial, sp = sp.com) + s(PctIndustrial, sp = sp.ind), family=gaussian, data = Train)

summary(out4)

plot(out4)

hist(fitted.values(out4),breaks=20)

#SWITCHING TO EVALUATION DATA

preds<-predict(out4, newdata = Eval) # do this with evaluation data

hist(preds, breaks = 50) # what does the distribution look like? LARGE OUTLYING NEGATIVE PREDICTED VALUES

resids<- Eval$StreetTotal-preds # get residuals

var(resids) # MSE for the residuals as a measure of fit

sd(resids) #average deviation around the fitted values

#########ASSIGNING SP VALUES #########

#USING OUTPUT FROM MODEL 2 (MODEL WITH INCOME TREATED AS LINEAR) - USE FACTOR OF 20 FOR THE FINAL MODEL

sp.vac <- (out2$sp[1])\*1000

sp.min <- (out2$sp[2])\*1000

sp.com <- (out2$sp[3])\*1000

sp.ind <- (out2$sp[4])\*1000

out4 <- gam(StreetTotal~LogMedianIncome + s(PctVacant, sp = sp.vac) + PctMinority +

s(PctCommercial, sp = sp.com) + s(PctIndustrial, sp = sp.ind), family=gaussian, data = Train)

summary(out4)

plot(out4)

hist(fitted.values(out4),breaks=20)

#SWITCHING TO EVALUATION DATA

preds<-predict(out4, newdata = Eval) # do this with evaluation data

hist(preds, breaks = 50) # what does the distribution look like? LARGE OUTLYING NEGATIVE PREDICTED VALUES

resids<- Eval$StreetTotal-preds # get residuals

var(resids) # MSE for the residuals as a measure of fit

sd(resids) #average deviation around the fitted values

#########ASSIGNING SP VALUES #########

#USING OUTPUT FROM MODEL 2 (MODEL WITH INCOME TREATED AS LINEAR) - USE FACTOR OF 20 FOR THE FINAL MODEL

sp.vac <- (out2$sp[1])\*5000

sp.min <- (out2$sp[2])\*5000

sp.com <- (out2$sp[3])\*5000

sp.ind <- (out2$sp[4])\*5000

out4 <- gam(StreetTotal~LogMedianIncome + s(PctVacant, sp = sp.vac) + PctMinority +

s(PctCommercial, sp = sp.com) + s(PctIndustrial, sp = sp.ind), family=gaussian, data = Train)

summary(out4)

plot(out4)

hist(fitted.values(out4),breaks=20)

#SWITCHING TO EVALUATION DATA

preds<-predict(out4, newdata = Eval) # do this with evaluation data

hist(preds, breaks = 50) # what does the distribution look like? LARGE OUTLYING NEGATIVE PREDICTED VALUES

resids<- Eval$StreetTotal-preds # get residuals

var(resids) # MSE for the residuals as a measure of fit

sd(resids) #average deviation around the fitted values

#########ASSIGNING SP VALUES #########

#USING OUTPUT FROM MODEL 2 (MODEL WITH INCOME TREATED AS LINEAR) - USE FACTOR OF 20 FOR THE FINAL MODEL

sp.vac <- (out2$sp[1])\*10000

sp.min <- (out2$sp[2])\*10000

sp.com <- (out2$sp[3])\*10000

sp.ind <- (out2$sp[4])\*10000

out4 <- gam(StreetTotal~LogMedianIncome + s(PctVacant, sp = sp.vac) + PctMinority +

s(PctCommercial, sp = sp.com) + s(PctIndustrial, sp = sp.ind), family=gaussian, data = Train)

summary(out4)

plot(out4)

hist(fitted.values(out4),breaks=20)

#SWITCHING TO EVALUATION DATA

preds<-predict(out4, newdata = Eval) # do this with evaluation data

hist(preds, breaks = 50) # what does the distribution look like? LARGE OUTLYING NEGATIVE PREDICTED VALUES

resids<- Eval$StreetTotal-preds # get residuals

var(resids) # MSE for the residuals as a measure of fit

sd(resids) #average deviation around the fitted values

#### RESETTING SP TO A FACTOR OF 20 ####

sp.vac <- (out2$sp[1])\*20

sp.min <- (out2$sp[2])\*20

sp.com <- (out2$sp[3])\*20

sp.ind <- (out2$sp[4])\*20

#### FINAL MODEL WITH TEST DATA ####

final.model <- gam(StreetTotal~LogMedianIncome + s(PctVacant, sp = sp.vac) + PctMinority +

s(PctCommercial, sp = sp.com) + s(PctIndustrial, sp = sp.ind), family=gaussian, data = Test) #FINAL MODEL WITH SP = FACTOR OF 20 (FOR PAPER)

summary(final.model)

dim(Test)

par(mai=rep(0.5, 4))

layout(matrix(c(1,1, 2,2, 0, 3,3, 0), ncol = 4, byrow = TRUE))

plot(final.model,residual=T,cex=1,pch=19,shade=T,

shade.col="light blue",col="blue")

#out5 <- gam(StreetTotal~LogMedianIncome + s(PctVacant, sp = sp.vac) + PctMinority +

# s(PctCommercial, sp = sp.com) + s(PctIndustrial, sp = sp.ind), family=gaussian, data = Train)

#summary(out4)

#help(gamObject)

#hist(fitted.values(out3),breaks=50)

#preds<-predict(out4, newdata = Eval) # do this with evaluation data

#hist(preds, breaks = 50) # what does the distribution look like?

#resids<-Eval$StreetTotal-preds # get residuals

#var(resids) # MSE for the residuals as a measure of fit

#[1] 10272.85

#sd(resids) #average deviation around the fitted values

#[1] 101.3551

### WITH SP AS A FACTOR OF 10

#out4 <- gam(StreetTotal~s(LogMedianIncome, sp = 3.895469) + s(PctVacant, sp = 0.006358943) + PctMinority +

# PctCommercial +s(PctIndustrial, sp = 0.002648138), family=gaussian, data = Train) # new model with PctCommercial and PctMinority not smoothed

# (because edfs were 1 in the previous model, indicating linearity)

#summary(out4)

#Family: gaussian

#Link function: identity

#Formula:

# StreetTotal ~ s(LogMedianIncome, sp = 3.895469) + s(PctVacant,

# sp = 0.006358943) + PctMinority + PctCommercial + s(PctIndustrial,

# sp = 0.002648138)

#Parametric coefficients:

# Estimate Std. Error t value Pr(>|t|)

#(Intercept) 37.35774 13.58779 2.749 0.0067 \*\*

#PctMinority -0.06247 0.19661 -0.318 0.7511

#PctCommercial 0.28750 0.32985 0.872 0.3848

#---

#Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

#Approximate significance of smooth terms:

# edf Ref.df F p-value

#s(LogMedianIncome) 1.972 2.562 0.238 0.782

#s(PctVacant) 5.082 6.086 6.767 1.60e-06 \*\*\*

#s(PctIndustrial) 7.302 8.295 13.406 3.81e-15 \*\*\*

#---

#Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

#R-sq.(adj) = 0.559 Deviance explained = 60.2%

#GCV = 2540.5 Scale est. = 2278 n = 168

#plot(out4)

#out4$sp

#hist(fitted.values(out4),breaks=50)

#preds<-predict(out4, newdata = Eval) # do this with evaluation data

#hist(preds, breaks = 50) # what does the distribution look like?

#resids<-Eval$StreetTotal-preds # get residuals

#var(resids) # MSE for the residuals as a measure of fit

#[1] 9842.96

#sd(resids) #average deviation around the fitted values

#[1] 99.21169

#summary(resids) #some HUGE residuals

# Min. 1st Qu. Median Mean 3rd Qu. Max.

#-346.800 -21.080 -5.590 7.293 11.200 842.500

### WITH SP AS A FACTOR OF 50

#sp.inc <- (out3$sp[1])\*50

#sp.vac <- (out3$sp[2])\*50

#sp.ind <- (out3$sp[3])\*50

#out5 <- gam(StreetTotal~s(LogMedianIncome, sp = sp.inc) + s(PctVacant, sp = sp.vac) + PctMinority +

# PctCommercial +s(PctIndustrial, sp = sp.ind), family=gaussian, data = Train)

#summary(out5)

#Family: gaussian

#Link function: identity

#Formula:

# StreetTotal ~ s(LogMedianIncome, sp = sp.inc) + s(PctVacant,

# sp = sp.vac) + PctMinority + PctCommercial + s(PctIndustrial,

# sp = sp.ind)

#Parametric coefficients:

# Estimate Std. Error t value Pr(>|t|)

#(Intercept) 35.05253 15.14129 2.315 0.0219 \*

#PctMinority -0.02105 0.21805 -0.097 0.9232

#PctCommercial 0.28889 0.35806 0.807 0.4210

#---

#Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

#Approximate significance of smooth terms:

# edf Ref.df F p-value

#s(LogMedianIncome) 1.299 1.551 0.095 0.817170

#s(PctVacant) 3.638 4.433 5.490 0.000492 \*\*\*

#s(PctIndustrial) 5.535 6.636 8.305 1.26e-08 \*\*\*

#---

#Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

#R-sq.(adj) = 0.438 Deviance explained = 48%

#GCV = 3153.3 Scale est. = 2900.4 n = 168

### WITH SP DIVIDED BY 10

#sp.inc <- (out3$sp[1])/10

#sp.vac <- (out3$sp[2])/10

#sp.ind <- (out3$sp[3])/10

#out6 <- gam(StreetTotal~s(LogMedianIncome, sp = sp.inc) + s(PctVacant, sp = sp.vac) + PctMinority +

# PctCommercial +s(PctIndustrial, sp = sp.ind), family=gaussian, data = Train)

#summary(out6)

#Family: gaussian

#Link function: identity

#Formula:

# StreetTotal ~ s(LogMedianIncome, sp = sp.inc) + s(PctVacant,

# sp = sp.vac) + PctMinority + PctCommercial + s(PctIndustrial,

# sp = sp.ind)

#Parametric coefficients:

# Estimate Std. Error t value Pr(>|t|)

#(Intercept) 43.6790 13.2581 3.295 0.00125 \*\*

#PctMinority -0.1515 0.1945 -0.779 0.43746

#PctCommercial 0.2070 0.3265 0.634 0.52709

#---

#Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

#Approximate significance of smooth terms:

# edf Ref.df F p-value

#s(LogMedianIncome) 6.124 7.197 1.299 0.268

#s(PctVacant) 8.716 8.967 6.024 3.29e-07 \*\*\*

#s(PctIndustrial) 8.968 9.000 15.876 < 2e-16 \*\*\*

#---

#Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

#R-sq.(adj) = 0.611 Deviance explained = 67.1%

#GCV = 2389.5 Scale est. = 2008.2 n = 168

#plot(out6)

#out6$sp

#hist(fitted.values(out6),breaks=50)

#preds<-predict(out6, newdata = Eval) # do this with evaluation data

#hist(preds, breaks = 50) # what does the distribution look like?

#resids<-Eval$StreetTotal-preds # get residuals

#var(resids) # MSE for the residuals as a measure of fit

#[1] 10470.12

#sd(resids) #average deviation around the fitted values

#[1] 102.3236

#summary(resids) #some HUGE residuals

# Min. 1st Qu. Median Mean 3rd Qu. Max.

#-409.900 -23.650 -7.055 5.459 15.760 818.800

### WITH SP DIVIDED BY 10

#sp.inc <- (out3$sp[1])\*200

#sp.vac <- (out3$sp[2])\*200

#sp.ind <- (out3$sp[3])\*200

#out7 <- gam(StreetTotal~s(LogMedianIncome, sp = sp.inc) + s(PctVacant, sp = sp.vac) + PctMinority +

# PctCommercial +s(PctIndustrial, sp = sp.ind), family=gaussian, data = Train)

#summary(out7)

#Family: gaussian

#Link function: identity

#Formula:

# StreetTotal ~ s(LogMedianIncome, sp = sp.inc) + s(PctVacant,

# sp = sp.vac) + PctMinority + PctCommercial + s(PctIndustrial,

# sp = sp.ind)

#Parametric coefficients:

# Estimate Std. Error t value Pr(>|t|)

#(Intercept) 43.6790 13.2581 3.295 0.00125 \*\*

#PctMinority -0.1515 0.1945 -0.779 0.43746

#PctCommercial 0.2070 0.3265 0.634 0.52709

#---

#Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

#Approximate significance of smooth terms:

# edf Ref.df F p-value

#s(LogMedianIncome) 6.124 7.197 1.299 0.268

#s(PctVacant) 8.716 8.967 6.024 3.29e-07 \*\*\*

#s(PctIndustrial) 8.968 9.000 15.876 < 2e-16 \*\*\*

#---

#Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

#R-sq.(adj) = 0.611 Deviance explained = 67.1%

#GCV = 2389.5 Scale est. = 2008.2 n = 168

#plot(out7)

#out7$sp

#hist(fitted.values(out7),breaks=50)

#preds<-predict(out7, newdata = Eval) # do this with evaluation data

#hist(preds, breaks = 50) # what does the distribution look like?

#resids<-Eval$StreetTotal-preds # get residuals

#var(resids) # MSE for the residuals as a measure of fit

#[1] 10470.12

#sd(resids) #average deviation around the fitted values

#[1] 102.3236

#summary(resids) #some HUGE residuals

# Min. 1st Qu. Median Mean 3rd Qu. Max.

#-409.900 -23.650 -7.055 5.459 15.760 818.800