R Codes for Final Project - Home Sweet Home, Inc

**Strategizing Code-**

dataset <- read.csv("house.csv")

if(typeof(dataset$LOT.SQFT) == "character") {dataset$LOT.SQFT <- factor(dataset$LOT.SQFT)}

if(typeof(dataset$YR.BUILT) == "character") {dataset$YR.BUILT <- factor(dataset$YR.BUILT)}

if(typeof(dataset$GROSS.AREA) == "character") {dataset$GROSS.AREA <- factor(dataset$GROSS.AREA)}

if(typeof(dataset$LIVING.AREA) == "character") {dataset$LIVING.AREA <- factor(dataset$LIVING.AREA)}

if(typeof(dataset$FLOORS) == "character") {dataset$FLOORS <- factor(dataset$FLOORS)}

if(typeof(dataset$ROOMS) == "character") {dataset$ROOMS <- factor(dataset$ROOMS)}

if(typeof(dataset$BEDROOMS) == "character") {dataset$BEDROOMS <- factor(dataset$BEDROOMS)}

if(typeof(dataset$FULL.BATH) == "character") {dataset$FULL.BATH <- factor(dataset$FULL.BATH)}

if(typeof(dataset$HALF.BATH) == "character") {dataset$HALF.BATH <- factor(dataset$HALF.BATH)}

if(typeof(dataset$KITCHEN) == "character") {dataset$KITCHEN <- factor(dataset$KITCHEN)}

if(typeof(dataset$FIREPLACE) == "character") {dataset$FIREPLACE <- factor(dataset$FIREPLACE)}

if(typeof(dataset$REMODEL) == "character") {dataset$REMODEL <- factor(dataset$REMODEL)}

training <- tail(dataset,5220-20)

testing <- head(dataset,nrow(dataset)-(5220-20))

#install.packages("randomForest")

library(randomForest)

model <- randomForest(TOTAL.VALUE~LOT.SQFT+YR.BUILT+GROSS.AREA+LIVING.AREA+FLOORS+ROOMS+BEDROOMS+FULL.BATH+HALF.BATH+KITCHEN+FIREPLACE+REMODEL,data=training)

testingout <- subset(testing, FALSE)

REMODEL <- sort(unique(training$REMODEL))

d1 <- expand.grid(REMODEL = REMODEL)

for (row in 1:nrow(testing))

{

testingbkup <- testing[row,]

testingtemp <- testing[row,]

for (i in 2:nrow(d1)) {testingtemp <- rbind(testingtemp, testingbkup)}

testingtemp$REMODEL <- d1$REMODEL

testingtemp$predicted <- predict(model,testingtemp,type="class")

testingout <- rbind(testingout, testingtemp[which.max(testingtemp$predicted),])

}

write.csv(testingout,"house\_Strategy.csv",row.names=FALSE)

training$predicted <- predict(model,training,type="class")

accuracy <- 100 \* cor(training$predicted,training$TOTAL.VALUE) ^ 2

paste(round(accuracy,2),"%",sep="")

testingout

**Strategizing Code Result-**

[1] "96.59%"

> testingout

TOTAL.VALUE LOT.SQFT YR.BUILT GROSS.AREA LIVING.AREA FLOORS

2 NA 9965 1880 2436 1352 2.0

21 NA 6590 1945 3108 1976 2.0

31 NA 7500 1890 2294 1371 2.0

41 NA 13773 1957 5032 2608 1.0

51 NA 5000 1910 2370 1438 2.0

61 NA 5000 1954 3220 1916 2.0

71 NA 10000 1950 2208 1200 1.0

81 NA 6835 1958 2582 1092 1.0

91 NA 5093 1900 4818 2992 2.0

101 NA 5000 1960 2624 1485 1.5

111 NA 6768 1958 2844 1460 1.5

121 NA 5000 1889 2196 1290 2.0

131 NA 12288 2004 4616 2378 2.0

141 NA 5000 1954 2536 1272 1.5

151 NA 5000 1940 2129 864 1.0

161 NA 10000 1950 2099 1445 1.0

171 NA 5000 1910 2408 1470 2.0

181 NA 9001 1875 2840 1632 2.0

191 NA 4450 1920 1400 1232 2.0

201 NA 5000 1889 2560 1302 1.5

ROOMS BEDROOMS FULL.BATH HALF.BATH KITCHEN FIREPLACE REMODEL

2 6 3 1 1 1 0 Recent

21 10 4 2 1 1 0 Recent

31 8 4 1 1 1 0 Recent

41 9 5 1 1 1 1 Recent

51 7 3 2 0 1 0 Recent

61 7 3 1 1 1 0 Recent

71 6 3 1 0 1 0 Recent

81 5 3 1 0 1 1 Recent

91 8 4 2 0 1 0 Recent

101 6 3 2 0 1 1 Recent

111 6 3 2 0 1 1 Recent

121 6 3 1 0 1 0 Recent

131 9 4 2 1 1 1 Recent

141 6 3 1 1 1 1 Recent

151 7 3 2 0 1 0 Recent

161 7 3 1 1 1 1 Recent

171 7 3 1 0 1 0 Recent

181 7 3 1 0 1 0 Recent

191 7 3 1 0 1 0 Recent

201 6 2 1 0 1 0 Recent

predicted

2 392.1610

21 446.8704

31 388.0497

41 568.1644

51 377.8822

61 420.9613

71 346.4595

81 344.1785

91 535.0377

101 355.5193

111 372.8347

121 358.6280

131 610.8098

141 346.3999

151 316.6811

161 367.2933

171 368.1482

181 398.4047

191 307.7673

201 333.8956

**Genetic Algorithm Code-**

dataset <- read.csv("house\_Strategy (1).csv")

dataset <- subset(dataset,)

#install.packages("genalg")

library(genalg)

limit <- 20000

iter <- 100

evalFunc <- function(x) {

current\_solution\_predicted <- x %\*% dataset$predicted

current\_solution\_GROSS.AREA <- x %\*% dataset$GROSS.AREA

if (current\_solution\_GROSS.AREA > limit)

return(0) else return(-current\_solution\_predicted)

}

monitor <- function(obj) {

print(paste("GENERATION:", obj$iter))

print(obj$population[which.min(obj$evaluations), ])

}

GAmodel <- rbga.bin(size = nrow(dataset), popSize = nrow(dataset)^2, iters = iter, mutationChance = 0.01, elitism = T, evalFunc = evalFunc, monitorFunc = monitor)

cat(summary(GAmodel))

solution <- GAmodel$population[which.min(GAmodel$evaluations),]

dataset[solution == 1, ]

write.csv(dataset[solution == 1, ],"house\_Strategy (1)\_GeneticAlgorithm.csv",row.names=FALSE)

-GAmodel$best[iter]

**Genetic Algorithm Code Result-**

> cat(summary(GAmodel))

GA Settings

Type = binary chromosome

Population size = 400

Number of Generations = 100

Elitism = TRUE

Mutation Chance = 0.01

Search Domain

Var 1 = [,]

Var 0 = [,]

GA Results

Best Solution : 1 0 1 0 0 0 1 0 0 0 0 1 0 0 1 1 1 0 1 1

> solution <- GAmodel$population[which.min(GAmodel$evaluations),]

> dataset[solution == 1, ]

TOTAL.VALUE LOT.SQFT YR.BUILT GROSS.AREA LIVING.AREA FLOORS

1 NA 9965 1880 2436 1352 2.0

3 NA 7500 1890 2294 1371 2.0

7 NA 10000 1950 2208 1200 1.0

12 NA 5000 1889 2196 1290 2.0

15 NA 5000 1940 2129 864 1.0

16 NA 10000 1950 2099 1445 1.0

17 NA 5000 1910 2408 1470 2.0

19 NA 4450 1920 1400 1232 2.0

20 NA 5000 1889 2560 1302 1.5

ROOMS BEDROOMS FULL.BATH HALF.BATH KITCHEN FIREPLACE REMODEL

1 6 3 1 1 1 0 Recent

3 8 4 1 1 1 0 Recent

7 6 3 1 0 1 0 Recent

12 6 3 1 0 1 0 Recent

15 7 3 2 0 1 0 Recent

16 7 3 1 1 1 1 Recent

17 7 3 1 0 1 0 Recent

19 7 3 1 0 1 0 Recent

20 6 2 1 0 1 0 Recent

predicted

1 392.1610

3 388.0497

7 346.4595

12 358.6280

15 316.6811

16 367.2933

17 368.1482

19 307.7673

20 333.8956

> write.csv(dataset[solution == 1, ],"house\_Strategy (1)\_GeneticAlgorithm.csv",row.names=FALSE)

> -GAmodel$best[iter]

[1] 3179.084