HW1

Anish Mohan

April 23, 2016

1. Q1

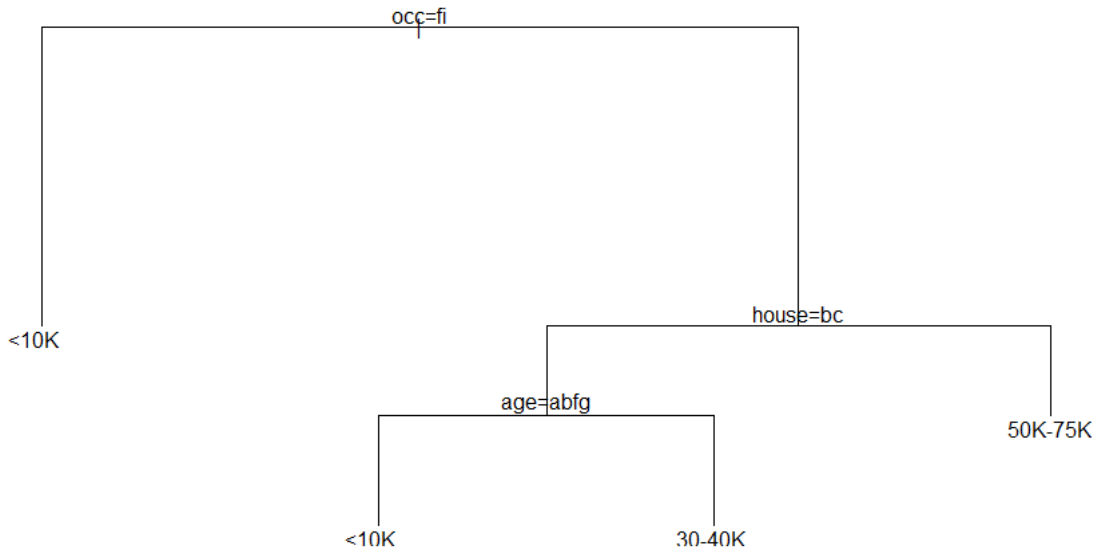
```
library(rpart)
    Income=read.csv("Income_Data.txt")
    ModIncome=data.frame(Inc=Income$X9,sex=Income$X2,marital=Income$X1,age=In
come$X5,edu=Income$X4,occ=Income$X5.1,dwelltime=Income$X5.2,dual=Income$X3,hh
=Income$X3.1,hh18=Income$X0,house=Income$X1.1,hometype=Income$X1.2,Ethnic=Inc
ome$X7,lang=Income$NA.)

    Inc=factor(ModIncome$Inc, levels=1:9, labels=c("<10K","10-15K","15-20K","
20-25K","25-30K","30-40K","40-50K","50K-75K",">75K"))
    sex=factor(ModIncome$sex, levels=1:2, labels=c("Male","Female"))
    marital=factor(ModIncome$marital, levels=1:5,labels=c("Married","live-in"
,"Divorced","Seperated","Single"))
    age=factor(ModIncome$age,levels=1:7,labels=c("14-17","18-24","25-34","35-
44","45-54","55-64","over 65"))
    edu=factor(ModIncome$edu,levels=1:6,labels=c("less grade 8","grade 9-11",
"grad high","1-3 college","College grad","Grad"))
    occ=factor(ModIncome$occ,levels=1:9,labels=c("Professional","Sales","labo
rer","Clerk","Home","Student","Military","Retired","Unemployed"))
    dwelltime=factor(ModIncome$dwelltime,levels=1:5,labels=c("<1year","1-3 ye
ars","4-6 years","7-10 years",">10 years"))
    dual=factor(ModIncome$dual, levels=1:3, labels=c("Not Married","Yes","No"
))
    hh=factor(ModIncome$hh, levels=1:9, labels=c("1","2","3","4","5","6","7",
"8",">9"))
    hh18=factor(ModIncome$hh18, levels=1:9, labels=c("1","2","3","4","5","6",
"7","8",">9"))
    house=factor(ModIncome$house, levels=1:3, labels=c("Own","Rent","Live wit
h family"))
    hometype=factor(ModIncome$house,levels=1:5, labels =c("House","Condo","Ap
a","Mobile","Other"))
    ethnic=factor(ModIncome$Ethnic, levels=1:8, labels=c("American Ind","Asia
n","Black","East indian","Hispanic","Pacific Island","White","Other"))
    lang=factor(ModIncome$lang,levels=1:3, labels=c("English","Spanish","Othe
r"))

    FinalInc=data.frame(Inc=Inc,sex=sex,marital=marital,age=age,edu=edu,occ=o
cc, dwelltime=dwelltime, dual=dual, hh=hh, hh18=hh18,house=house, hometype=ho
metype,ethnic=ethnic, lang=lang)

    incfit=rpart(Inc~.-Inc,FinalInc)
```

```
    plot(incfit)
    text(incfit)
```




```
    summary(incfit)
```

```
## Call:
## rpart(formula = Inc ~ . - Inc, data = FinalInc)
##   n= 8992
##
##           CP nsplit rel error    xerror        xstd
## 1 0.08500069      0 1.0000000 1.0000000 0.005174763
## 2 0.02842556      1 0.9149993 0.9115496 0.005777209
## 3 0.01000000      3 0.8581482 0.8598041 0.006035673
##
## Variable importance
##      occ      age    house      edu hometype  marital     dual
##       38       20       19        8        7        5        4
##
## Node number 1: 8992 observations,    complexity param=0.08500069
##   predicted class=<10K     expected loss=0.8059386  P(node) =1
##     class counts:  1745   775   667   813   722  1110   969  1308   883
##    probabilities: 0.194 0.086 0.074 0.090 0.080 0.123 0.108 0.145 0.098
##   left son=2 (1843 obs) right son=3 (7149 obs)
##   Primary splits:
##       occ      splits as  RRRRRLRRL, improve=495.4924, (136 missing)
##       age      splits as  LRRRRRR,   improve=495.4387, (0 missing)
##       edu      splits as  LLRRRR,    improve=400.2344, (86 missing)
```

```
##       house    splits as  RRL,       improve=396.2046, (240 missing)
##       hometype splits as  RRL--,     improve=396.2046, (240 missing)
##   Surrogate splits:
##       age      splits as  LRRRRRR, agree=0.859, adj=0.314, (136 split)
##       edu      splits as  LLRRRR,  agree=0.832, adj=0.185, (0 split)
##       house    splits as  RRL,     agree=0.832, adj=0.185, (0 split)
##       hometype splits as  RRL--,   agree=0.832, adj=0.185, (0 split)
##
## Node number 2: 1843 observations
##   predicted class=<10K     expected loss=0.371134  P(node) =0.20496
##     class counts:  1159   142    77    61    51    75    78   106    94
##    probabilities: 0.629 0.077 0.042 0.033 0.028 0.041 0.042 0.058 0.051
##
## Node number 3: 7149 observations,    complexity param=0.02842556
##   predicted class=50K-75K  expected loss=0.8318646  P(node) =0.79504
##     class counts:   586   633   590   752   671  1035   891  1202   789
##    probabilities: 0.082 0.089 0.083 0.105 0.094 0.145 0.125 0.168 0.110
##   left son=6 (3950 obs) right son=7 (3199 obs)
##   Primary splits:
##       house    splits as  RLL,     improve=159.5983, (210 missing)
##       hometype splits as  RLL--,   improve=159.5983, (210 missing)
##       marital  splits as  RLLLL,   improve=136.6468, (109 missing)
##       dual     splits as  LRR,     improve=125.6204, (0 missing)
##       age      splits as  LLRRRRR, improve=108.3470, (0 missing)
##   Surrogate splits:
##       age     splits as  LLLRRRR,   agree=0.733, adj=0.405, (210 split)
##       marital splits as  RLLRL,     agree=0.721, adj=0.377, (0 split)
##       dual    splits as  LRR,       agree=0.703, adj=0.337, (0 split)
##       occ     splits as  RLLLR-LR-, agree=0.621, adj=0.155, (0 split)
##       edu     splits as  LLLLLR,    agree=0.583, adj=0.070, (0 split)
##
## Node number 6: 3950 observations,    complexity param=0.02842556
##   predicted class=30-40K   expected loss=0.8592405  P(node) =0.4392794
##     class counts:   502   518   470   530   463   556   394   371   146
##    probabilities: 0.127 0.131 0.119 0.134 0.117 0.141 0.100 0.094 0.037
##   left son=12 (1569 obs) right son=13 (2381 obs)
##   Primary splits:
##       age      splits as  LLRRRLL,   improve=58.65649, (0 missing)
##       occ      splits as  RLLLL-LL-, improve=49.69127, (78 missing)
##       edu      splits as  LLLRRR,    improve=36.17191, (36 missing)
##       house    splits as  -RL,       improve=32.30111, (122 missing)
##       hometype splits as  -RL--,     improve=32.30111, (122 missing)
##   Surrogate splits:
##       house    splits as  -RL,       agree=0.674, adj=0.180, (0 split)
##       hometype splits as  -RL--,     agree=0.674, adj=0.180, (0 split)
##       occ      splits as  RLRRR-LL-, agree=0.658, adj=0.140, (0 split)
##       edu      splits as  RLLRRR,    agree=0.648, adj=0.113, (0 split)
##       marital  splits as  RRRLL,     agree=0.637, adj=0.087, (0 split)
##
## Node number 7: 3199 observations
```

```
##   predicted class=50K-75K  expected loss=0.7402313  P(node) =0.3557607
##     class counts:    84   115   120   222   208   479   497   831   643
##    probabilities: 0.026 0.036 0.038 0.069 0.065 0.150 0.155 0.260 0.201
##
## Node number 12: 1569 observations
##   predicted class=<10K     expected loss=0.7743786  P(node) =0.1744884
##     class counts:   354   308   229   187   134   127    86    99    45
##    probabilities: 0.226 0.196 0.146 0.119 0.085 0.081 0.055 0.063 0.029
##
## Node number 13: 2381 observations
##   predicted class=30-40K   expected loss=0.8198236  P(node) =0.2647909
##     class counts:   148   210   241   343   329   429   308   272   101
##    probabilities: 0.062 0.088 0.101 0.144 0.138 0.180 0.129 0.114 0.042
```

- **1.1 Short Summary on the results:**
  - Occupation seems to be one of the key factors that influence the Annual Income. If the occupation is "Unemployed" or "Student" the predicted annual income is less than 10K. Next good predictor is if the family rents v.s owns the house. If the family owns the house, it more likely that their annual income is >=$50K. If the household rent or lives with a family, then age is a next important predictor of the annual income. For people in the age group 14-24 and people above 55 the predicted household income is below 10K
- **1.a** Yes, surrogate splits were used in the construction of optimal tree.
  - A surrogate split is used when a data point is missing the variable value which is used for decision about which branch the data point should be sent to. A surrogate is the value of another dimension/variable for the same data point; The surrogate value is used for making the splitting decision instead of the missing variable value.
  - In my decision tree, Occupation is the first variable used for splitting. However for 136 data points, occupation is not listed. In this case, age is used as a surrogate for occupation. Age and occupation give the split for ~86% of given datapoints
  -

```
#OPTIMAL TREE
incfittemp=rpart(Inc~.-Inc,FinalInc, cp=0.001)
printcp(incfittemp)

##
## Classification tree:
## rpart(formula = Inc ~ . - Inc, data = FinalInc, cp = 0.001)
##
## Variables actually used in tree construction:
## [1] age     edu     ethnic  hh      house   marital occ     sex
##
```

```
## Root node error: 7247/8992 = 0.80594
##
## n= 8992
##
##           CP nsplit rel error  xerror      xstd
## 1  0.0850007      0   1.00000 1.00000 0.0051748
## 2  0.0284256      1   0.91500 0.90838 0.0057949
## 3  0.0057265      3   0.85815 0.86036 0.0060332
## 4  0.0051056      5   0.84670 0.85249 0.0060674
## 5  0.0046916      6   0.84159 0.84587 0.0060951
## 6  0.0041856      8   0.83221 0.83938 0.0061213
## 7  0.0040017     11   0.81965 0.83041 0.0061562
## 8  0.0031047     13   0.81165 0.81661 0.0062066
## 9  0.0030357     15   0.80544 0.81565 0.0062100
## 10 0.0020698     17   0.79937 0.80778 0.0062369
## 11 0.0019318     18   0.79730 0.80751 0.0062378
## 12 0.0017938     20   0.79343 0.80778 0.0062369
## 13 0.0012419     21   0.79164 0.80378 0.0062501
## 14 0.0011729     23   0.78915 0.80364 0.0062505
## 15 0.0011039     25   0.78681 0.80392 0.0062496
## 16 0.0010349     26   0.78570 0.80557 0.0062442
## 17 0.0010000     28   0.78363 0.80557 0.0062442

incfit2=rpart(Inc~.-Inc,FinalInc, method='class', cp=0.0019)
```

- **The Optimal tree(This cannot be plotted in HW because of two many nodes)**
- **Lowest cross-validated error is for 23 split tree==0.80281. Using 1SE rule, find the simplest tree that is 1 SE away= 0.80281+0.0062532 ie. 18 split tree).**

- **1b.**

```
mydata=data.frame(Inc=" ",sex="Male",marital="Married",age="35-44",edu="G
rad",occ="Professional",dwelltime="1-3 years",dual="No",hh="3",hh18="3",h
ouse="Own",hometype="House",ethnic="Asian",lang="Other")
tree.pred=predict(incfit2,mydata, type="class")
tree.pred

##    1
## >75K
## Levels: <10K 10-15K 15-20K 20-25K 25-30K 30-40K 40-50K 50K-75K >75K
```

- **Based on the data, household is predicted as >75K.**

2. Q2

```
HouseType=read.csv("Housetype_Data.txt")
ModHouse=data.frame(Hometype=HouseType$X1,sex=HouseType$X2, marital=House
Type$X4,age=HouseType$X7,edu=HouseType$X4.1,occ=HouseType$X5,Inc=HouseTyp
e$NA.,dwelltime=HouseType$X5.1,dual=HouseType$X1.1,hh=HouseType$X1.2,hh18
=HouseType$X0,housestatus=HouseType$X1.3,Ethnic=HouseType$X7.1,lang=House
Type$X1.4)

Hometype=factor(ModHouse$Hometype,levels=1:5, labels =c("House","Condo","
Apa","Mobile","Other"))
sex=factor(ModHouse$sex, levels=1:2, labels=c("Male","Female"))
marital=factor(ModHouse$marital, levels=1:5,labels=c("Married","live-in",
"Divorced","Seperated","Single"))
age=factor(ModHouse$age,levels=1:7,labels=c("14-17","18-24","25-34","35-4
4","45-54","55-64","over 65"))
edu=factor(ModHouse$edu,levels=1:6,labels=c("less grade 8","grade 9-11","
grad high","1-3 college","College grad","Grad"))
occ=factor(ModHouse$occ,levels=1:9,labels=c("Professional","Sales","labor
er","Clerk","Home","Student","Military","Retired","Unemployed"))
Inc=factor(ModHouse$Inc, levels=1:9, labels=c("<10K","10-15K","15-20K","2
0-25K","25-30K","30-40K","40-50K","50K-75K",">75K"))
dwelltime=factor(ModHouse$dwelltime,levels=1:5,labels=c("<1year","1-3 yea
rs","4-6 years","7-10 years",">10 years"))
dual=factor(ModHouse$dual, levels=1:3, labels=c("Not Married","Yes","No")
)
hh=factor(ModHouse$hh, levels=1:9, labels=c("1","2","3","4","5","6","7","
8",">9"))
hh18=factor(ModHouse$hh18, levels=1:9, labels=c("1","2","3","4","5","6","
7","8",">9"))
housestatus=factor(ModHouse$house, levels=1:3, labels=c("Own","Rent","Liv
e with family"))
ethnic=factor(ModHouse$Ethnic, levels=1:8, labels=c("American Ind","Asian
","Black","East indian","Hispanic","Pacific Island","White","Other"))
lang=factor(ModHouse$lang,levels=1:3, labels=c("English","Spanish","Other
"))

Finalhouse=data.frame(Hometype=Hometype,sex=sex,marital=marital,age=age,e
du=edu,occ=occ,Inc=Inc, dwelltime=dwelltime, dual=dual, hh=hh, hh18=hh18,
housestatus=housestatus, ethnic=ethnic, lang=lang)

homefit=rpart(Hometype~.-Hometype,data=Finalhouse, method='class', cp=0.0
01)
printcp(homefit)

##
## Classification tree:
## rpart(formula = Hometype ~ . - Hometype, data = Finalhouse, method = "
class",
##     cp = 0.001)
```

```
##
## Variables actually used in tree construction:
## [1] age         dwelltime   ethnic      hh          housestatus Inc
## [7] marital     occ
##
## Root node error: 3694/9012 = 0.4099
##
## n= 9012
##
##           CP nsplit rel error  xerror     xstd
## 1  0.3256632      0   1.00000 1.00000 0.012639
## 2  0.0173254      1   0.67434 0.67380 0.011490
## 3  0.0083920      3   0.63969 0.64077 0.011309
## 4  0.0040606      4   0.63129 0.63508 0.011277
## 5  0.0035192      5   0.62723 0.63454 0.011274
## 6  0.0032485      7   0.62019 0.63319 0.011266
## 7  0.0013535      8   0.61695 0.62669 0.011228
## 8  0.0012182      9   0.61559 0.62859 0.011239
## 9  0.0010828     11   0.61316 0.62994 0.011247
## 10 0.0010000     18   0.60531 0.63048 0.011250
```

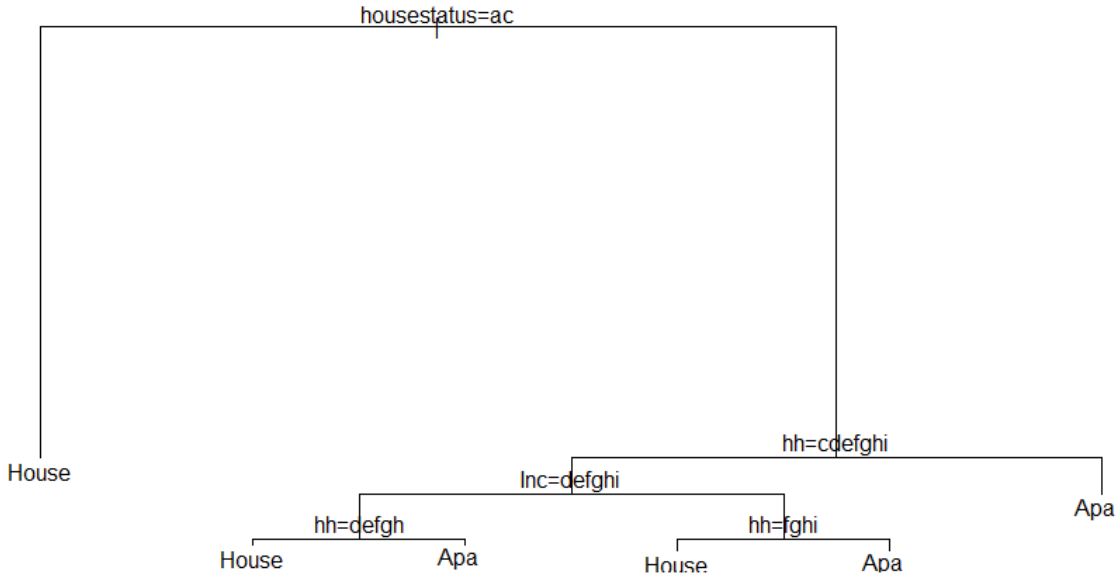
- **Based on the printcp, model with lower cross-valdiated error is 8 split model. Using 1-SE rule, we can select the simplest model with 1SE away i,e 0.62940+0.011244= 0.640644 cross-validated error ie, we choose the 4 split model and plot it**

```
homefit2=rpart(Hometype~.-Hometype,data=Finalhouse, method='class', cp=0.
004)
printcp(homefit2)

##
## Classification tree:
## rpart(formula = Hometype ~ . - Hometype, data = Finalhouse, method = "
class",
##     cp = 0.004)
##
## Variables actually used in tree construction:
## [1] hh          housestatus Inc
##
## Root node error: 3694/9012 = 0.4099
##
## n= 9012
##
##          CP nsplit rel error  xerror     xstd
## 1 0.3256632      0   1.00000 1.00000 0.012639
## 2 0.0173254      1   0.67434 0.67434 0.011493
## 3 0.0083920      3   0.63969 0.64645 0.011341
```

```
## 4 0.0040606      4   0.63129 0.64402 0.011328
## 5 0.0040000      5   0.62723 0.64131 0.011312

plot(homefit2)
text(homefit2)
```



- **Misclassifcations: Number of miss-classification is 3694*0.63129= 2332 nodes**
- **Summary:**
  - Householder status seems to be the most important predictor for the Type of Home. If the householder owns the home,it is more likely that the property is a house. It looks like in this dataset, people who lived in a house owned them rather renting them.
  - If the Householder is renting or living with a family then the next important factor is the number of people in the household.
  - If there are 4-8 people in a household and their income is high enough i.e greater than 20K, then there is a high chance they live in a house. This makes sense as the family makes sufficient income and there are many people in the household, for comfort reasons they probably live in a house.
  - However, it looks like if the family is small 3 or less or very large 9 or more, then the family lives in an apartment. It is likely that apartment suffices for a small family of 3 hence that is the preferred choice. If a family has more than 9 members in in the household, then most of the income probably goes for people expenses leaving little money to rent, hence they they choose the apartment.

- If the household income is low i.e upto 20k, then if the household has 1-5 members it is more likely they live in in an apartment. If there are more than 5 members in the household, they live in a house. This is similar to the trend earlier, that smaller families probably choose to live in an apartment because an apartment supports small family easily.
- Larger families even on low income require the space of a house hence they live in a house.

3. Q3

- A Target function gives the true mapping from the Input variable space to the output variable space f: X−> Y If the true Target function is known, we can predict the value of Output/Response given an input variable.
- Theoretically, A Target function should give an accurate function for prediction; However, even when a true target function is known, we might not be able to accurately predict values because of Noise.
- Our goal is to get the best estimate of the target function $\hat{Y} \approx Y = F(X)$. And for estimating this, we have data point pairs $< x_i, y_i >$

4. Q4

- Empirical risk evaluated on training data is a reasonable surrogate for the actual(population) prediction risk if certain assumptions are true:
  - Training data comes from the same distribution as the population data.
  - A learning method is not overfit to reduce the empirical risk only for the training data.
  - Generally, it is a good idea to keep a part of training set (test set) separate only to evaluate the empirical risk of a learning method. This test set is not used to tune the parameters of learning method but to act as a surrogate to estimate the empirical risk for the actual population.

5. Q5

- We try to approximate the target function using by assuming certain model. Note that we do not have a complete joint distribution of all Input variables $= and out put variable Y. We only have some datapoint from the joint distribution.
- 1st restriction is that any model we choose for approximating the target function only represents a subset of the entire target function space. This is a restriction of the model (e.g linear model). Hence, by choosing a particular model we restriction the set of functions that can be selected.
- 2nd restriction is because we only have some sample data points from the joint distribution. We define a loss function and try to find the best function that fits to the data points. However, there are many functions that might fit the particular data set which will minmize the loss function. And if the data set is tweaked a new set of functions will emerge. Hence choosing a model restricts the function class. If we choose from a class of all possible functions, we will find many functions that fit the training data well, but gives poor results on test data.

6. Q6

- Bias is the error introduced by approximating a complex target function with a simplet model which does not capture the true underlying form of the target function. Example approximating a quadratic function with least squares. Because we chose a simpler model, we will never be able to accurately estimate the target function beyond an error level. This is called the Bias of the model.
- Variance is the amount by which a predicted function $\hat{f}$ would change if we were using a different training data set. Different training sets give different value of the function and the variations in the function value is called the Variance.
- Bias-Variance tradeoff: For finding the best function that approximates the target function we use different models or classes of function. Bias is introduced by choosing a particular class of functions to approximate the target function. Typically, more flexible a function class is, lower will be the bias associated with that function class. However, by increasing flexibility, variance of the output $\hat{f}$ is high when we use different training set with this flexible function class. Bias-Variance tradeoff refers to this trade-off where following things happen:
  - compromise on increasing in Variance by choosing a more flexible class of functions to represent the target function
  - compromise on increase in Bias by choosing a simpler model which reduces the variance.
- We manage the Bias-Variance trade-off to get a lower expected test error with the training data or test data.

8. Q8:

- The variable to use for primary split is selected after reviewing all the variables and considering all possible splits to get the best reduction in prediction error. Surrogate variable does not contain as much information as the primary split variable. Surrogate variables are useful in the scenario where the primary variable is a missing in particular data point. Hence we look for best substitute variable or the surrogate variable for the primary variable.

9. Q9

Q9)

Regression tree model

$$F(x) = \sum_{m=1}^{M} c_m I(x \in R_m)$$

The squared risk criterion is

$$R = \sum_{i=1}^{N} [y_i - F(x_i)]^2$$

Let us consider the criteria for region = 1

hence the risk criteria is

$$R_{R_1} = \sum_{i=1}^{N} [y_i I(x_i \in R_1) - F(x_i)]^2$$

but $F(x_i)$ for Region-1 $= c_1 I(x_i \in R_1)$

$$\Rightarrow R_{R_1} = \sum_{i=1}^{N} [y_i I(x_i \in R_1) - c_1 I(x_i \in R_1)]^2$$

differentiating w.r.t x & setting the derivative to zero

$$\frac{\partial}{\partial x} R_{R_1} = 0 \Rightarrow \frac{\partial}{\partial x} \left[\sum_{i=1}^{N} [y_i I(x_i \in R_1) - c_1 I(x_i \in R_1)]^2 = 0 \right.$$

$$\Rightarrow 2 \left[\sum_{i=1}^{N} [y_i I(x_i \in R_1) - c_1 I(x_i \in R_1)] \frac{\partial}{\partial x} [\sum_{i=1}^{N} [y_i I(x_i \in R_1) - c_1 I(x_i \in R_1)] = 0 \right.$$

$$\Rightarrow \sum_{i=1}^{N} y_i \, I(x_i \in R_1) - \sum_{i=1}^{N} C_1 \, I(x_i \in R_1) = 0$$

$$\Rightarrow \boxed{C_1 = \frac{\sum_{i=1}^{N} y_i I(x_i \in R_1)}{\sum_{i=1}^{N} I(x_i \in R_1)}}$$

This was proved for Region R_1 & factor C_1
Can be proved for all Region 1-m
and the general expression is

$$\hat{C}_m = \frac{\sum_{i=1}^{N} y_i \, I(x_i \in R_m)}{\sum_{i=1}^{N} I(x_i \in R_m)}$$

10. Q10

Q10)

Improvement in the squared risk error was

$$m^* = \underset{1 \le m \le M}{\operatorname{argmax}} \sum_{x_i \in R_m} \left\{ [y_i - \bar{y}_m^{\ell} I(x_i \in R_m^{\ell}) - \bar{y}_m^{r} I(x_i \in R_m^{r})]^2 - [y_i - \bar{y}_m]^2 \right\} \quad \text{(A)}$$

The expanded form can be simplified as follows

Say points $1:K$ belong to R^{ℓ} for a particular region under consideration

$\Rightarrow$ Risk for that region $= (y_1 - \bar{y}^{\ell})^2 + (y_2 - \bar{y}^{\ell})^2 + \cdots (y_K - \bar{y}^{\ell})^2$

$$= y_1^2 + (\bar{y}^{\ell})^2 - 2y_1\bar{y}^{\ell} + y_2^2 + (\bar{y}^{\ell})^2 - 2y_2\bar{y}^{\ell} \cdots$$

$$= (y_1^2 + y_2^2 + \cdots y_K^2) + \bar{y}^{\ell})^2 + \bar{y}^{\ell})^2 + \cdots \text{ to } -2\bar{y}^{\ell}(y_1 + y_2 + \cdots y_K)$$

$$= (y_1^2 + y_2^2 + \cdots y_K^2) + N\bar{y}^{\ell})^2 - 2\bar{y}^{\ell}(N\bar{y}^{\ell})$$

$$= (y_1^2 + y_2^2 + \cdots y_K^2) + N\bar{y}^{\ell 2} - 2N\bar{y}^{\ell 2}$$

$$= (y_1^2 + y_2^2 + y_3^2 \cdots) - N\bar{y}^{\ell 2} \quad \text{(1)}$$

Similarly if points $y_{k+1} \cdots y_n$ belong to region R^2 then we get

$$\sum (y_i - \bar{y}_m^3 I(x_i \in R_m^3))^2 = y_{k+1}^2 + y_{k+2}^2 + \cdots y_n^2 - N\bar{y}^2 \quad - (2)$$

Now looking at

$$\sum_{i=1}^{N} (y_i - \bar{y}_m)^2 = (y_1^2 + y_2^2 + \cdots y_n^2) + N\bar{y}_m^2 - 2\bar{y}_m(y_1 + y_2 + \cdots y_n)$$

$$= (y_1^2 + y_2^2 + \cdots y_n^2) + N\bar{y}_m^2 - 2N\bar{y}_m^2$$

$$= (y_1^2 + y_2^2 + y_n^2) - N\bar{y}_m^2 \quad \text{---} \quad (3)$$

Substituting (1), (2) & (3) in (A) we get the risk for a region m as

$$\text{Risk} = (y_1^2 + y_2^2 + \cdots y_k^2) - Ny_k^2 + (y_{k+1}^2 + y_{k+2}^2 + \cdots y_n^2) - Ny^2$$

$$+ (y_1^2 + y_2^2 + \cdots y_n^2) + N\bar{y}_m^2$$

$$\boxed{\text{Risk} = -(N\bar{y}_m^2 - N\bar{y}_k^2 - N\bar{y}_n^2)}$$

Now $\bar{y}_m = \frac{y_m}{N}$ ← sum of all y in Region M
N ← # of points in M

$$N(\bar{y}_m)^2 = N\left(\frac{y_m}{N}\right)^2$$

$y_m = y_l + y_r$ ← sum of left + Right region

$N = N_l + N_r$ ← sum of left & Right points

$$= \frac{N}{N^2}(y_l + y_r)^2$$

$$= \frac{1}{N}\left(N_l \bar{y}_l + N_r \bar{y}_r\right)^2$$

$\bar{y}_l$ = avg of left region

$\bar{y}_r$ = avg of right region

$$N\bar{y}_m^2 = \frac{N_l^2 \bar{y}_l^2 + N_r^2 \bar{y}_r^2 + 2N_l N_r \bar{y}_l \bar{y}_r}{N}$$

$$= \left(N\bar{y}_m^2 - N_r \bar{y}_r^2 - N_l \bar{y}_l^2\right)$$

$$= -\left(\frac{N_l^2 \bar{y}_l + N^2 \bar{y}_r^2 + 2N_l N_r \bar{y}_l \bar{y}_r - (N_r + N_l)(N_l \bar{y}_l^2 + N_r \bar{y}_r^2)}{(N_l + N_r)}\right)$$

$$= -\left(\frac{N_l^2 \bar{y}_l + N_r^2 \bar{y}_r^2 + 2N_l N_r \bar{y}_l \bar{y}_r - N_r^2 \bar{y}_r^2 + N_l N_r \bar{y}_r^2 - N_l N_r \bar{y}_l^2 - N_r^2 \bar{y}_l^2}{N_l + N_r}\right)$$

$$= \frac{N_l N_r (\bar{y}_l^2 + y_r^2 - 2y_l y_r)}{N}$$

$$= \frac{N_l N_r (\bar{y}_l - \bar{y}_r)^2}{N}$$

Risk for a region

$$= \frac{N_R N_C (\bar{y}_c - \bar{y}_n)^2}{N}$$

11. Q11

(Q11)

Change in squared risk, when a point K changes group

i.e proved

$$\text{Risk} = \frac{N_L N_R (\bar{y}_L - \bar{y}_R)^2}{N}$$

→ Say point 'K' moved from left to right region then Risk is

$$\text{Risk}' = \frac{N_L' N_R' (\bar{y}_L' - \bar{y}_R')^2}{N}$$

where

→ N_L' = # of points in new left region
= $N_L - 1$

→ N_R' = # of points in new right region
= $N_R + 1$

→ $\bar{y}_L'$ = New mean of left region

$$= \frac{(\bar{y}_L N_L - K)}{(N_L - 1)}$$

→ $\bar{y}_R$ = New mean for right region

$$= \frac{\bar{y}_R N_R + K}{(N_R + 1)}$$

$$\Rightarrow (RSSK') = \frac{(N_1-1)(N_2+1)}{N}\left[\frac{\bar{y}_1 N_1 - K}{(N_1-1)} - \frac{\bar{y}_2 N_2 + K}{N_2+1}\right]^2$$

$$= \frac{(N_1-1)(N_2+1)}{N}\left[\frac{\bar{y}_1 N_1}{N_1-1} - \frac{K}{N_1-1} - \frac{\bar{y}_2 N_2}{N_2+1} - \frac{K}{N_2+1}\right]^2$$

$$= \frac{(N_1-1)(N_2+1)}{N}\left[\frac{\bar{y}_1 N_1 (N_2+1) + \bar{y}_2 N_2 (N_1-1)}{(N_1-1)(N_2+1)} - \frac{K(N_2+1) + K(N_1-1)}{(N_1-1)(N_2+1)}\right]^2$$

$$= \frac{(N_1-1)(N_2+1)}{N\,(N_1-1)^2(N_2+1)^2}\left[\bar{y}_1 N_1 N_2 + \bar{y}_1 N_2 - \bar{y}_2 N_2 N_1 + \bar{y}_2 N_2 - \left(KN_2 + K + KN_1 - K\right)\right]^2$$

$$= \frac{1}{N(N_1-1)(N_2+1)}\left[N_2 N_1 (\bar{y}_1 - \bar{y}_2) + \bar{y}_1 N_1 + \bar{y}_2 N_2 - KN\right]^2$$

$$= \frac{1}{N(N_1-1)(N_2+1)}\left[N_2 N_1 (\bar{y}_1 - \bar{y}_2) + N\bar{y} - KN\right]^2$$

$$= \frac{1}{(N)(N_2+1)(N_1-1)}\left(N_2 N_1 (\bar{y}_1 - \bar{y}_2) + N(\bar{y} - K)\right)^2$$

so the updating formula for risk when a point moves from left to right region is

$$= \frac{1}{N(N_L-1)(N_R+1)} \left[N_L N_L(\bar{y}_L-\bar{y}_L) - N(\bar{y}-K)\right]^2$$

where $\bar{y}$ = mean of entire region

By symmetry, when a point moves from right to left the updation formula is

$$= \frac{1}{N(N_R-1)(N_L+1)} \left[(N_L N_L(\bar{y}_L-y_R) - N(\bar{y}-K)\right]^2$$

12. Q12

- Predicting observation with missing value of splitting variable by treating this as a terminal node:
  - This is a fastest approach and is O(1) time operation. This will be much faster than surrogate variables since that requires finding the alternative variables that similar split to primary variables.
  - Other dimensions of this observation, will not be considered for prediction which means that there is higher probability of the predicition being incorrect.
- Send observation to daughter node that contains majority of the training data.
  - This is a faster approach compared to surrogate variables.
  - After this split, other dimensions will be considered for further split. This is similar to Surrogate split but different from the earlier approach of treating this as terminal node.
  - Since the observation is classified with majority of training data, the chances of error are less than $\sim 50\%$. However, with surrogate variables, if a surrogate variable has same split as the primary variable, it is possible to classify the observation to the correct node with error being very low. However, if the surrogate variable does not match with primary split for atleast 50% of observations, then the approach of sending the observation with majority might work better.

13. Q13:

- Relative Advantage/Disadvantage of treating "missing as a class"
  - Advantage: This approach is faster than surrogate split as we just treat the missing as a category. Surrogate splits require calculating another variable that does a similar split to primary split variable, hence slower.
  - Advantage: By treating missing as a category, it becomes easier to predict with test data, when it was some dimensions missing.
  - Disadvantage: By treating observations with 'missing' values as one class, we incorrectly club them together in one category as "missingness" is not a true feature. Surrogate split does better to find the another variable that has close match to the primary split variable.
  - Disadvantage: The three-way split for orderable variables, collects observation with missing entries together, when there is no orderable relationship between the observations.
  - Disadvantage: The three-way split and categorizing missing information as a category -vely impacts the "interpretability" of the tree.
- If certain datapoints have missing variables at random, this strategy might not encourage correlated sets to substitute each other. However if there is a pattern to missing variables, e.g high net worth individuals do not share their taxable revenue, or women of certain age group leave the age entry blank etc, then this approach might allow correlated sets to substitute. For e.g, house value instead of a net worth of an individual, husband's age in case of women etc. However, if the missingness is randome, then this method just categorizes all observations with missing variables as one class, where there may not be any similarty between the datapoints.
- The strategy is specifically to categorize "missing" as a class of the categorical; So in general, if there are no missing values in an observation then this method cannot be directly used.
- For predicting with missing values in future data, we can use methods like following:
  - Assume that node as the terminal node and provide the estimate for the prediction
  - Use the known surrogate or correlated variable to walk down the tree.