Information Gain Simulation

MA Mingyu 14110562D

4/4/2017

COMP4433 Assignment 2 Question 1a

derek.ma@connect.polyu.hk derek.ma

Initialization of Training Data

```
#init demo data to a data frame
training_set <- data.frame(
  sex=c("F","F","M","M","F","F","M","F","M","F","F","F","M","M","F","F","M","F","M","F"),
  age=c("Y","M","S","M","Y","S","Y","M","Y","M","M","M","Y","S","S","Y","S","Y","M","S"),
  married=c("Y","N","N","Y","Y","Y","N","Y","N","Y","Y","Y","Y","N","Y","N","N","Y","N","Y"),
  income=c("H","H","M","M","M","L","H","L","M","M","H","H","L","M","M","L","H","H","M","L"),
  plan=c("A","C","B","B","C","B","C","C","A","C","C","A","B","A","A","C","C","B","B","A"),
  renew=c("Y","Y","N","Y","Y","N","N","Y","N","Y","Y","Y","N","N","N","Y","N","Y","Y","N")
  )
```

Functions to Calculate Entropy and Informtion Gain

```
entropy <- function(dataset,targetFeature){
  #input: the target feature column after selected
  #out: the entropy under this condition
  target <- dataset[,targetFeature]
  allValues <- unique(target)
  n <- length(target)
  entropyValue <- 0
  for (value in allValues){
    p <- (length(subset(target, target == value))/n)
    entropyValue = entropyValue - p*log2(p)
  }
  entropyValue
}
```

```
infoGain <- function(dataset,feature,targetFeature){
  infoGain_value <- entropy(dataset,targetFeature)
  currentColumn <- dataset[,feature]
  allValues <- unique(currentColumn)
  n <- length(currentColumn)
  for (value in allValues){
    p <- (length(subset(currentColumn, currentColumn == value))/n)
    entropyTemp <- entropy(subset(dataset, dataset[,feature] == value), targetFeature)
    infoGain_value <- infoGain_value - p*entropyTemp
  }
  infoGain_value
}
```

Best Split Selection - First Round

Calculate information gain for each attribute to select the root split.

```
infoGain_1 <- data.frame(
  sex = infoGain(training_set,"sex","renew"),
  age = infoGain(training_set,"age","renew"),
  married = infoGain(training_set,"married","renew"),
  income = infoGain(training_set,"income","renew"),
  plan = infoGain(training_set,"plan","renew")
  )
infoGain_1
```

```
##         sex       age    married     income       plan
## 1 0.1814963 0.6479446 0.06002335 0.04794461 0.09277445
```

According to the calculation result for infomation gain, initial split is on age, because it has the highest information gain.

Split - First Round

```
training_set_ageY <- subset(training_set, training_set$age == "Y")
training_set_ageM <- subset(training_set, training_set$age == "M")
training_set_ageS <- subset(training_set, training_set$age == "S")
training_set_ageY
```

```
##    sex age married income plan renew
## 1    F   Y       Y      H    A     Y
## 5    F   Y       Y      M    C     Y
## 7    M   Y       N      H    C     N
## 9    M   Y       N      M    A     N
## 13   M   Y       Y      L    B     N
## 16   F   Y       N      L    C     Y
## 18   F   Y       Y      H    B     Y
```

```
training_set_ageM
```

```
##    sex age married income plan renew
## 2    F   M       N      H    C     Y
## 4    M   M       Y      M    B     Y
## 8    F   M       Y      L    C     Y
## 10   F   M       Y      M    C     Y
## 11   F   M       Y      H    C     Y
## 12   F   M       Y      H    A     Y
## 19   M   M       N      M    B     Y
```

```
training_set_ageS
```

```
##    sex age married income plan renew
## 3    M   S       N      M    B     N
## 6    F   S       Y      L    B     N
## 14   M   S       N      M    A     N
## 15   F   S       Y      M    A     N
## 17   M   S       N      H    C     N
## 20   F   S       Y      L    A     N
```

Check Stopping Criteria

We found that for records of age “Middle”, all of the customers will renew and for records of age “Senior”, all of the customer will not renew. These two branches match the stopping criteria. We do not need to split these two branches any more. They are pure already.

Best Split Selection - Second Round

Now we are going to find the best split feature for the branch with age “Young”.

```r
infoGain_2 <- data.frame(
  sex = infoGain(training_set_ageY,"sex","renew"),
  married = infoGain(training_set_ageY,"married","renew"),
  income = infoGain(training_set_ageY,"income","renew"),
  plan = infoGain(training_set_ageY,"plan","renew")
  )
infoGain_2
```

```
##         sex   married     income       plan
## 1 0.9852281 0.1280853 0.02024421 0.02024421
```

We can find that “sex” feature has highest information gain. Then “sex” should be selected as next level split node.

Split - Second Round

```r
training_set_ageY_sexM <- subset(training_set_ageY, training_set_ageY$sex == "M")
training_set_ageY_sexF <- subset(training_set_ageY, training_set_ageY$sex == "F")
training_set_ageY_sexM
```

```
##    sex age married income plan renew
## 7    M   Y       N      H    C     N
## 9    M   Y       N      M    A     N
## 13   M   Y       Y      L    B     N
```

```r
training_set_ageY_sexF
```

```
##    sex age married income plan renew
## 1    F   Y       Y      H    A     Y
## 5    F   Y       Y      M    C     Y
## 16   F   Y       N      L    C     Y
## 18   F   Y       Y      H    B     Y
```

Check Stopping Criteria

All samples for the given nodes belong to the same class. Thus the split action terminates and the decision has been formed.