Homework 3

Kevin Wilson (syx009)

March 2, 2018

1

Explain how the basic decision tree algorithm can be extended to incorporate the ranges (for Age and Salary) and the counts (in the count column) into the calculation of the impurity measures. Then, use the extended algorithm by hand to find the best split of the given data using the following impurity measures. You only need to show how to find the best split at the root node of the decision tree. You need to show the details of the calculations for at least one attribute, and show the results for the rest of attributes. You may want to write a program to perform the calculations. If you do so, also hand in your program source code.

The basic algorithm can be extended by treating the continuous-valued attributes for Age and Salary as discrete values instead, using the ranges provided in the table as a single discrete value, which works for information gain and gain ratio. For Gini index, the attributes' best splitting subset must be found by selecting the subset such that the selected subset gives the minimum Gini index overall between all possible splitting subsets. The counts may be incorporated into the algorithm using the AVC-set (simply the counts given in the table and their respective tuple values) of each of the attributes and their different values at the current node of the decision tree.

(a) information gain (calculated by hand)

$Info(D) = -\frac{113}{165}\log_2\frac{113}{165} - \frac{52}{165}\log_2\frac{52}{165} \approx 0.8990$

$$
\begin{aligned}
Info_{department}(D) =& \frac{110}{165}(-\frac{80}{110}\log_2\frac{80}{110} - \frac{30}{110}\log_2\frac{30}{110}) + \\
& \frac{31}{165}(-\frac{23}{31}\log_2\frac{23}{31} - \frac{8}{31}\log_2\frac{8}{31}) + \\
& \frac{14}{165}(-\frac{4}{14}\log_2\frac{4}{14} - \frac{10}{14}\log_2\frac{10}{14}) + \\
& \frac{10}{165}(-\frac{6}{10}\log_2\frac{6}{10} - \frac{4}{10}\log_2\frac{4}{10}) + \\
& \approx 0.8504 \\
Gain(department) =& Info(D) - Info_{department}(D) \\
\approx& 0.8990 - 0.8504 \approx 0.0486
\end{aligned}
$$

$Info_{age}(D) \approx 0.4743$
$Gain(age) \approx 0.8990 - 0.4743 \approx 0.4247$

$Info_{salary}(D) \approx 0.3615$
$Gain(age) \approx 0.8990 - 0.3615 \approx 0.5375$

$Max = Gain(salary) \approx 0.5375$ so will split root node based on salary.

(b) gain ratio (calculated by hand)

$$
\begin{aligned}
Splitinfo_{department}(D) =& -\frac{110}{165}\log_2\frac{110}{165} - \frac{31}{165}\log_2\frac{31}{165} - \frac{14}{165}\log_2\frac{14}{165} - \frac{10}{165}\log_2\frac{10}{165} \\
\approx& 1.3903 \\
GainRatio(department) =& \frac{Gain(department)}{Splitinfo_{department}(D)} \\
=& \frac{0.0486}{1.3903} \approx 0.0350
\end{aligned}
$$

$Splitinfo_{age}(D) \approx 1.8782$
$GainRatio(age) \approx \frac{0.4247}{1.8782} \approx 0.2261$

$Splitinfo_{salary}(D) \approx 2.0116$
$GainRatio(salary) \approx \frac{0.5375}{2.0116} \approx 0.2672$

$Max = GainRatio(salary) \approx 0.2672$ so will split root node based on salary.

(c) gini index (calculated using calc_gini.py)

$Gini(D) = 1 - \left(\frac{113}{165}\right)^2 - \left(\frac{52}{165}\right)^2 \approx 0.4317$

$$
\begin{aligned}
Gini_{department \in \{sales\}}(D) &= \frac{110}{165}\left(1 - \left(\frac{80}{110}\right)^2 - \left(\frac{30}{110}\right)^2\right) + \frac{55}{165}\left(1 - \left(\frac{33}{55}\right)^2 - \left(\frac{22}{55}\right)^2\right) \\
&\approx 0.4245 \\
Gini_{department \in \{systems\}}(D) &= \frac{31}{165}\left(1 - \left(\frac{23}{31}\right)^2 - \left(\frac{8}{31}\right)^2\right) + \frac{134}{165}\left(1 - \left(\frac{90}{134}\right)^2 - \left(\frac{44}{134}\right)^2\right) \\
&\approx 0.4302 \\
Gini_{department \in \{marketing\}}(D) &= \frac{14}{165}\left(1 - \left(\frac{4}{14}\right)^2 - \left(\frac{10}{14}\right)^2\right) + \frac{151}{165}\left(1 - \left(\frac{109}{151}\right)^2 - \left(\frac{42}{151}\right)^2\right) \\
&\approx 0.4021 \\
Gini_{department \in \{secretary\}}(D) &= \frac{10}{165}\left(1 - \left(\frac{6}{10}\right)^2 - \left(\frac{4}{10}\right)^2\right) + \frac{155}{165}\left(1 - \left(\frac{107}{155}\right)^2 - \left(\frac{48}{155}\right)^2\right) \\
&\approx 0.4307 \\
Gini_{department \in \{sales,systems\}}(D) &= \frac{141}{165}\left(1 - \left(\frac{103}{141}\right)^2 - \left(\frac{38}{141}\right)^2\right) + \frac{24}{165}\left(1 - \left(\frac{10}{24}\right)^2 - \left(\frac{14}{24}\right)^2\right) \\
&\approx 0.4072 \\
Gini_{department \in \{sales,marketing\}}(D) &= \frac{124}{165}\left(1 - \left(\frac{84}{124}\right)^2 - \left(\frac{40}{124}\right)^2\right) + \frac{41}{165}\left(1 - \left(\frac{29}{41}\right)^2 - \left(\frac{12}{41}\right)^2\right) \\
&\approx 0.4313 \\
Gini_{department \in \{sales,secretary\}}(D) &= \frac{120}{165}\left(1 - \left(\frac{86}{120}\right)^2 - \left(\frac{34}{120}\right)^2\right) + \frac{45}{165}\left(1 - \left(\frac{27}{45}\right)^2 - \left(\frac{18}{45}\right)^2\right) \\
&\approx 0.4263
\end{aligned}
$$

$Gini_{age \in \{21..25\}}(D) \approx 0.4043$
$Gini_{age \in \{26..30\}}(D) \approx 0.3478$
$Gini_{age \in \{31..35\}}(D) \approx 0.4016$
$Gini_{age \in \{36..40\}}(D) \approx 0.3711$
$Gini_{age \in \{41..45\}}(D) \approx 0.4143$
$Gini_{age \in \{46..50\}}(D) \approx 0.4084$
$Gini_{age \in \{21..25,26..30\}}(D) \approx 0.2889$
$Gini_{age \in \{21..25,31..35\}}(D) \approx 0.4272$
$Gini_{age \in \{21..25,36..40\}}(D) \approx 0.4315$
$Gini_{age \in \{21..25,41..45\}}(D) \approx 0.4206$
$Gini_{age \in \{21..25,46..50\}}(D) \approx 0.4242$
$Gini_{age \in \{26..30,31..35\}}(D) \approx 0.4196$

$Gini_{age \in \{26..30,36..40\}}(D) \approx 0.4080$
$Gini_{age \in \{26..30,41..45\}}(D) \approx 0.3707$
$Gini_{age \in \{26..30,46..50\}}(D) \approx 0.3773$
$Gini_{age \in \{31..35,36..40\}}(D) \approx 0.3467$
$Gini_{age \in \{31..35,41..45\}}(D) \approx 0.3882$
$Gini_{age \in \{31..35,46..50\}}(D) \approx 0.3832$
$Gini_{age \in \{36..40,41..45\}}(D) \approx 0.3514$
$Gini_{age \in \{36..40,41..45\}}(D) \approx 0.3447$
$Gini_{age \in \{41..45,46..50\}}(D) \approx 0.3901$
$Gini_{age \in \{21..25,26..30,31..35\}}(D) \approx 0.3239$
$Gini_{age \in \{21..25,26..30,36..40\}}(D) \approx 0.3663$
$Gini_{age \in \{21..25,26..30,41..45\}}(D) \approx 0.3159$
$Gini_{age \in \{21..25,26..30,46..50\}}(D) \approx 0.3241$
$Gini_{age \in \{21..25,31..35,36..40\}}(D) \approx 0.3945$
$Gini_{age \in \{21..25,31..35,41..45\}}(D) \approx 0.4210$
$Gini_{age \in \{21..25,31..35,46..50\}}(D) \approx 0.4183$
$Gini_{age \in \{21..25,36..40,41..45\}}(D) \approx 0.4286$
$Gini_{age \in \{21..25,36..40,46..50\}}(D) \approx 0.4268$
$Gini_{age \in \{21..25,41..45,46..50\}}(D) \approx 0.4304$

$Gini_{salary \in \{26k..30k\}}(D) \approx 0.3549$
$Gini_{salary \in \{31k..35k\}}(D) \approx 0.3681$
$Gini_{salary \in \{36k..40k\}}(D) \approx 0.4084$
$Gini_{salary \in \{41k..45k\}}(D) \approx 0.4267$
$Gini_{salary \in \{46k..50k\}}(D) \approx 0.3054$
$Gini_{salary \in \{66k..70k\}}(D) \approx 0.3839$
$Gini_{salary \in \{26k..30k,31k..35k\}}(D) \approx 0.2154$
$Gini_{salary \in \{26k..30k,36k..40k\}}(D) \approx 0.3836$
$Gini_{salary \in \{26k..30k,41k..45k\}}(D) \approx 0.3453$
$Gini_{salary \in \{26k..30k,46k..50k\}}(D) \approx 0.4212$
$Gini_{salary \in \{26k..30k,66k..70k\}}(D) \approx 0.4045$
$Gini_{salary \in \{31k..35k,36k..40k\}}(D) \approx 0.3951$
$Gini_{salary \in \{31k..35k,41k..45k\}}(D) \approx 0.3594$
$Gini_{salary \in \{31k..35k,46k..50k\}}(D) \approx 0.4139$
$Gini_{salary \in \{31k..35k,66k..70k\}}(D) \approx 0.4136$
$Gini_{salary \in \{36k..40k,41k..45k\}}(D) \approx 0.4282$
$Gini_{salary \in \{36k..40k,46k..50k\}}(D) \approx 0.2721$
$Gini_{salary \in \{36k..40k,66k..70k\}}(D) \approx 0.3581$
$Gini_{salary \in \{41k..45k,46k..50k\}}(D) \approx 0.3230$
$Gini_{salary \in \{41k..45k,46k..50k\}}(D) \approx 0.4123$
$Gini_{salary \in \{46k..50k,66k..70k\}}(D) \approx 0.2349$
$Gini_{salary \in \{26k..30k,31k..35k,36k..40k\}}(D) \approx 0.2558$
$Gini_{salary \in \{26k..30k,31k..35k,41k..45k\}}(D) \approx 0.1933$
$Gini_{salary \in \{26k..30k,31k..35k,46k..50k\}}(D) \approx 0.3911$

$Gini_{salary \in \{26k..30k, 31k..35k, 66k..70k\}}(D) \approx 0.2915$
$Gini_{salary \in \{26k..30k, 36k..40k, 41k..45k\}}(D) \approx 0.3751$
$Gini_{salary \in \{26k..30k, 36k..40k, 46k..50k\}}(D) \approx 0.4077$
$Gini_{salary \in \{26k..30k, 36k..40k, 66k..70k\}}(D) \approx 0.4190$
$Gini_{salary \in \{26k..30k, 41k..45k, 46k..50k\}}(D) \approx 0.4251$
$Gini_{salary \in \{26k..30k, 41k..45k, 66k..70k\}}(D) \approx 0.3976$
$Gini_{salary \in \{26k..30k, 46k..50k, 66k..70k\}}(D) \approx 0.3876$

Minimum Gini overall: $Gini_{salary \in \{26k..30k, 31k..35k, 41k..45k\}}(D) \approx 0.1933$
Maximum reduction of impurity: $Gini(D) - Gini_{salary \in \{26k..30k, 31k..35k, 41k..45k\}}(D) \approx 0.4317 - 0.1933 \approx 0.2384$
So will split root node based on whether tuple has a salary in 26k..30k, 31k..35k, 41k..45k or not.

2

Extend the Naive Bayes classifier algorithm so that it can also incorporate the ranges and counts in calculation of the probabilities.

(a) Show how the extended algorithm would calculate the prior probabilities and the conditional probabilities $P(A_k|C)$ using the data table as the training data

$P(x_k|C_i) = \frac{|x_{k,C_i}|}{|C_{i,D}|}$ where $|x_{k,C_i}|$ is the number of tuples of class C_i having value x_k for A_k and $|C_{i,D}|$ is the number of tuples of class C_i in D.

$P(department = sales|status = junior) = \frac{80}{113} \approx 0.7080$
$P(department = sales|status = senior) = \frac{30}{52} \approx 0.5769$
$P(department = systems|status = junior) = \frac{23}{113} \approx 0.2035$
$P(department = systems|status = senior) = \frac{8}{52} \approx 0.1538$
$P(department = marketing|status = junior) = \frac{4}{113} \approx 0.0354$
$P(department = marketing|status = senior) = \frac{10}{52} \approx 0.1923$
$P(department = secretary|status = junior) = \frac{6}{113} \approx 0.0531$
$P(department = secretary|status = senior) = \frac{4}{52} \approx 0.0769$

$P(age = 21..25|status = junior) = \frac{20}{113} \approx 0.1770$
$P(age = 21..25|status = senior) = \frac{0}{52} = 0$
$P(age = 26..30|status = junior) = \frac{49}{113} \approx 0.4336$
$P(age = 26..30|status = senior) = \frac{0}{52} = 0$
$P(age = 31..35|status = junior) = \frac{44}{113} \approx 0.3894$
$P(age = 31..35|status = senior) = \frac{35}{52} \approx 0.6731$
$P(age = 36..40|status = junior) = \frac{0}{113} = 0$
$P(age = 36..40|status = senior) = \frac{10}{52} \approx 0.1923$
$P(age = 41..45|status = junior) = \frac{0}{113} = 0$
$P(age = 41..45|status = senior) = \frac{3}{52} \approx 0.0577$
$P(age = 46..50|status = junior) = \frac{0}{113} = 0$
$P(age = 46..50|status = senior) = \frac{4}{52} \approx 0.0769$

$P(salary = 26k..30k|status = junior) = \frac{46}{113} \approx 0.4071$
$P(salary = 26k..30k|status = senior) = \frac{0}{52} = 0$
$P(salary = 31k..35k|status = junior) = \frac{40}{113} \approx 0.3540$
$P(salary = 31k..35k|status = senior) = \frac{0}{52} = 0$
$P(salary = 36k..40k|status = junior) = \frac{0}{113} = 0$
$P(salary = 36k..40k|status = senior) = \frac{4}{52} \approx 0.0769$
$P(salary = 41k..45k|status = junior) = \frac{4}{113} \approx 0.0354$
$P(salary = 41k..45k|status = senior) = \frac{0}{52} = 0$
$P(salary = 46k..50k|status = junior) = \frac{23}{113} \approx 0.2035$
$P(salary = 46k..50k|status = senior) = \frac{40}{52} \approx 0.7692$
$P(salary = 66k..70k|status = junior) = \frac{0}{113} = 0$
$P(salary = 66k..70k|status = senior) = \frac{8}{52} \approx 0.1538$

For the zero probabilities, I am assuming a laplacian correction will be applied once a tuple is given to the algorithm to classify.

(b) Show how the extended algorithm would determine the status of the following data tuple

$t =< department : systems, status :?, age : 28, salary : 50k >$

Again, you need to show the details of the calculation for some of the probabilities, and for tuple t

Must apply laplacian correction to age in calculation of $P(age = 26..30|status = junior)$ and $P(age = 26..30|status = senior)$ since $P(age = 26..30|senior) = 0$
$Laplacian(P(age = 26..30|status = junior)) = \frac{49+1}{113+6} \approx 0.4202$

$$\begin{aligned} P(status = junior|t) &= P(systems|junior)P(26..30|junior)P(46k..50k|junior)P(junior) \\ &= 0.2035 \cdot 0.4202 \cdot 0.2035 \cdot 0.6848 \\ &\approx 0.0119 \end{aligned}$$

$Laplacian(P(age = 26..30|status = senior)) = \frac{0+1}{52+6} \approx 0.0172$

$$\begin{aligned} P(status = senior|t) &= P(systems|senior)P(26..30|senior)P(46k..50k|senior)P(senior) \\ &= 0.1538 \cdot 0.0172 \cdot 0.7692 \cdot 0.3152 \\ &\approx 0.0006 \end{aligned}$$

Since $P(status = junior|t) > P(status = senior|t)$ the algorithm would determine the status of tuple t to be junior.

3

Use this dataset to create a suitable new data file, either hwk03.arff or hwk03.csv, by replicating each row with the number of copies as indicated in the count column. For example, you should make the first row in the given table appear 30 times in the new table. Then, remove the count column.

Write a program that trains a decision tree using the new data file as the training data and use the decision tree to predict the status of a user provided unseen data, for example,

$t =< department : systems, status :?, age : 28, salary : 50k >$

Specifically, you either write a Java program that uses Wekas J48 or a Python Jupyter notebook that uses SciKit-Learns DecisionTreeClassifier to learn the decision tree. Notice that SciKit-Learn requires to encode categorical attributes as integer attributes.

You may have to convert the actual age and salary into the corresponding ranges for the decision tree to work on the unseen data.

Completed in decision_tree_classifier.py.

```
Input:
Enter tuple for Decision Tree in this format: DEPARTMENT AGE SALARY
>> systems 28 50k

Output:
Decision Tree Prediction of {'department': 'systems', 'age': '26..30',
'salary': '46k..50k'}:
junior
```

4

Make another new dataset (named hwk03-02.arff or hwk03-02.csv) from the data file obtained in the previous exercise by converting the values in the age and salary columns to random values drawn from the specific range for each row. For example, suppose the age of a row is 31..35, replace it by a random integer between 31 and 35 inclusively.

Write a program that uses either Weka or SciKit-Learn to learn a Naive Bayes classifier and use it to find the status of a user provided unseen data, for example,

$t =< department : systems, status :?, age : 28, salary : 50k >$

Completed in naive_bayes_classifier.py.

```
Input:
Enter tuple for Naive Bayes in this format: DEPARTMENT AGE SALARY
>> systems 28 50k

Output:
Naive Bayes Prediction of {'department': 'systems', 'age': '28', 'salary': '50k'}:
junior
```