Middle Tennessee State University
College of Basic and Applied Sciences
Spring 2014

CSCI 7350: Data Mining
Professor: Dr. Cen Li

Homework 2
By: Zane Colgin

January 30, 2014
modified: February 4, 2014

NOTE: All calculations carried through to the end at double floating point precision. Numbers rounded only for presentation.

1. **Download C4.5 program and compile the program.**

   Complete. I followed the steps on the C4.5 tutorial page from the link from the class website, and from the professor's email.

2. **Apply C4.5 to learn a complete decision tree from this data. Draw (or copy and paste) the resulting tree in your answer.**

   Results from the `c4.5` executable on the given dataset:

```
Decision Tree:

Odor = almond: edible (8.0)
Odor = spicy: edible (4.0/2.0)
Odor = foul: poisonous (2.0)


Tree saved


Evaluation on training data (14 items):

 Before Pruning           After Pruning
----------------   ---------------------------
Size      Errors   Size      Errors   Estimate

 4    2(14.3%)      4    2(14.3%)    (38.3%)   <<
```

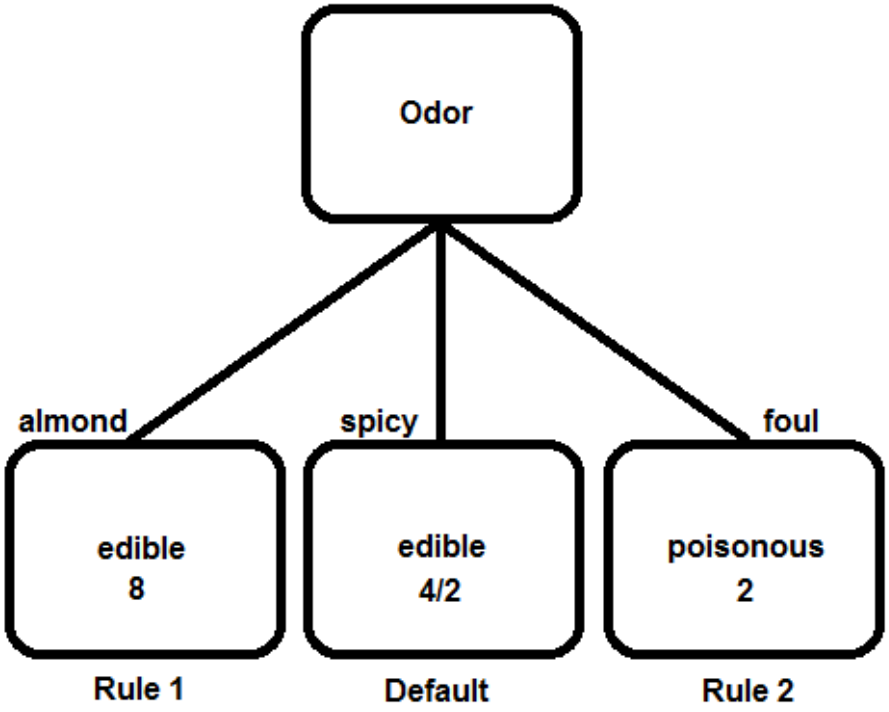

Figure 1: Decision tree constructed by the method used in the C4.5 tutorial webpage. Includes rules as determined by the `c4.5rules` executable on the given dataset. The number in the box indicates the number of data objects in the training set which are categorized by each rule. Note that the default rule for "spicy" falsely identifies two of the training set objects as edible.

3. **Classify the following three mushrooms using the decision tree learned from the previous step:**

Data Object	**Cap Shape**	**Cap Color**	**Odor**	**class**
q1	bell	grey	almond	edible
q2	convex	grey	spicy	edible
q3	flat	brown	foul	poisonous

4. **Perform χ^2 test to determine if the first level split performed in question 1 is statistically significant? ($\rho = 0.05$) Should it be pruned away? Show all the computations involved.**

   Assume: The samples are a good random sample of the population it represents. Is "Odor" a good way to predict if a mushroom is edible or poisonous? Null hypothesis: "Odor has no relation to the edibility of a mushroom."

	edible	poisonous	Total
almond	8	0	8
spicy	2	2	4
foul	0	2	2
Total	10	4	14

- almond/edible: $(8 \cdot 10)/14 \approx 5.71$
- almond/poisonous: $(8 \cdot 4)/14 \approx 2.29$
- spicy/edible: $(4 \cdot 10)/14 \approx 2.86$
- spicy/poisonous: $(4 \cdot 4)/14 \approx 1.14$
- foul/edible: $(2 \cdot 10)/14 \approx 1.43$
- foul/poisonous: $(2 \cdot 4)/14 \approx 0.57$

	edible	poisonous	Total
almond (observed)	8	0	8
almond (expected)	5.71	2.29	
spicy (observed)	2	2	4
spicy (expected)	2.86	1.14	
foul (observed)	0	2	2
foul (expected)	1.43	0.57	
Total	10	4	14

$$\sum_{i=1}^{rowsize}\sum_{j=1}^{colsize}\frac{(O_{ij}-E_{ij})^2}{E_{ij}}=$$

$$\sum_{i=1}^{3}\sum_{j=1}^{2}\frac{(O_{ij}-E_{ij})^2}{E_{ij}}\approx$$

$$\frac{(8-5.71)^2}{5.71}+\frac{(2-2.86)^2}{2.86}+\frac{(0-1.43)^2}{1.43}+$$
$$\frac{(0-2.29)^2}{2.29}+\frac{(2-1.14)^2}{1.14}+\frac{(2-0.57)^2}{0.57}\approx$$

$$0.9143+2.2857+0.2571+0.6429+1.4286+3.5714\approx 9.1$$

Degree of Freedom of this problem:

$r=(\text{number of rows}-1)(\text{number of columns}-1)=(3-1)(2-1)=2$

	$P(X \le x)$							
	0.010	0.025	0.050	0.100	0.900	0.950	0.975	0.990
r	$\chi^2_{0.99}(r)$	$\chi^2_{0.975}(r)$	$\chi^2_{0.95}(r)$	$\chi^2_{0.90}(r)$	$\chi^2_{0.10}(r)$	$\chi^2_{0.05}(r)$	$\chi^2_{0.025}(r)$	$\chi^2_{0.01}(r)$
1	0.000	0.001	0.004	0.016	2.706	3.841	5.024	6.635
2	0.020	0.051	0.103	0.211	4.605	5.991	7.378	9.210
3	0.115	0.216	0.352	0.584	6.251	7.815	9.348	11.34
4	0.297	0.484	0.711	1.064	7.779	9.488	11.14	13.28
5	0.554	0.831	1.145	1.610	9.236	11.07	12.83	15.09
6	0.872	1.237	1.635	2.204	10.64	12.59	14.45	16.81
7	1.239	1.690	2.167	2.833	12.02	14.07	16.01	18.48
8	1.646	2.180	2.733	3.490	13.36	15.51	17.54	20.09
9	2.088	2.700	3.325	4.168	14.68	16.92	19.02	21.67
10	2.558	3.247	3.940	4.865	15.99	18.31	20.48	23.21

Figure 2: χ^2 Table – Penn State University webpage: https:// onlinecourses.science.psu.edu/ stat414/ sites/ onlinecourses.science.psu.edu.stat414/ files/ lesson16/ ChiSquareTableR10.gif

$$\chi^2_{0.05}(2)=5.991$$

Since $9.1 > 5.991$ we reject the null hypothesis and conclude: Odor has a relationship with the edibility of a mushroom.

5. **Describe how to compute the classification accuracy of the DecisionTree program using "leave one out" on the data set provided.**

   Using leave-one-out, we remove a data object from the training set, generate a decision tree, and then compute the error of the tree using the removed data object as test data. This process is repeated for each data object in the original data set. The error is then averaged for all these tests. If the original training set has n data objects then there should be n tests using n decision trees each constructed from $n-1$ data objects. In the class notes this is called N-fold cross validation. Ahmad Taherkhani describes this as *leave-one-out*.[1]

```
% hw2.m
% CSCI 7350 - Homework 2
% Professor: Cen Li
%
% AUTHOR:
%       Zane Colgin
%       Middle Tennessee State University
%       January 2014
%
% NOTE:
%       "!!!" in comments indicates programming note
%       i.e. look here for debugging notes, optimizations

clear all; close all; clc; format compact
% ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
% ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
%% ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ INITIALIZE / DATA

% Data:
% Object    Odor        class

% X1        almond      edible
% X2        almond      edible
% X3        spicy       poisonous
% X4        almond      edible
% X5        almond      edible
% X6        spicy       edible
% X7        almond      edible
% X8        almond      edible
% X9        foul        poisonous
% X10       spicy       edible
% X11       almond      edible
% X12       spicy       poisonous
% X13       almond      edible
% X14       foul        poisonous

X(14)=struct('Odor','foul','Class','poisonous');
X(1)=struct('Odor','almond','Class','edible');
X(2)=struct('Odor','almond','Class','edible');
X(3)=struct('Odor','spicy','Class','poisonous');
X(4)=struct('Odor','almond','Class','edible');
X(5)=struct('Odor','almond','Class','edible');
X(6)=struct('Odor','spicy','Class','edible');
X(7)=struct('Odor','almond','Class','edible');
X(8)=struct('Odor','almond','Class','edible');
X(9)=struct('Odor','foul','Class','poisonous');
```

```
X(10)=struct('Odor','spicy','Class','edible');
X(11)=struct('Odor','almond','Class','edible');
X(12)=struct('Odor','spicy','Class','poisonous');
X(13)=struct('Odor','almond','Class','edible');

C = {'edible','poisonous'};
Odors = { 'almond'; 'spicy'; 'foul' };
structFieldName = 'Odor';
className = 'Class';

m=length(C);
n=length(Odors);


% ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
% ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
%% ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ TABLE 1
% extra row and column for totals
% !!! probably shouldn't do it like this
% !!! try storing totals in separate vector
obsTable = zeros(n+1,m+1);

for k=1:length(X)       % number of data objects
    for i=1:n           % attribute type (which Odor)

        if (strcmp(X(k).(structFieldName),Odors{i}))
            for j=1:m % number of classes
                if (strcmp(X(k).(className),C{j}))
                    obsTable(i,j) = obsTable(i,j) + 1;
                end
            end
        end

    end
end

% Totals
obsTable(1:n,m+1) = sum(obsTable(1:n,1:m),2);
obsTable(n+1,1:m) = sum(obsTable(1:n,1:m),1);
obsTable(n+1,m+1) = sum(obsTable(1:n,m+1));
obsTable


% ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
% ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
%% ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ TABLE 2
expTable = zeros(n,m);
for i=1:n
    for j=1:m
```

```
        expTable(i,j) = ...
            obsTable(i,m+1)*obsTable(n+1,j)/obsTable(n+1,m+1);
    end
end
expTable


% ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
a = (  obsTable(1:n,1:m) - expTable  ).^2 ./expTable
b = sum(sum(a))
```

output

```
obsTable =
     8     0     8
     2     2     4
     0     2     2
    10     4    14
expTable =
    5.7143    2.2857
    2.8571    1.1429
    1.4286    0.5714
a =
    0.9143    2.2857
    0.2571    0.6429
    1.4286    3.5714
b =
    9.1000
```

References

[1] Taherkhani, A. *Using Decision Tree Classifiers in Source Code Analysis to Recognize Algorithms: An Experiment with Sorting Algorithms*, The Computer Journal, 54 (2011).