[

**Fall 2016 Data Mining Homework 2 RUID:174004719**

1. [30 points: 5 for each part] Consider the training examples shown in Table 1 for a binary classification problem.

Table 1: Data set for Exercise1.

|  |  |  |  |
| --- | --- | --- | --- |
| Movie ID | Format | Movie Category | Class |
| 1 | DVD | Entertainment | C0 |
| 2 | DVD | Comedy | C0 |
| 3 | DVD | Documentaries | C0 |
| 4 | DVD | Comedy | C0 |
| 5 | DVD | Comedy | C0 |
| 6 | DVD | Comedy | C0 |
| 7 | Online | Comedy | C0 |
| 8 | Online | Comedy | C0 |
| 9 | Online | Comedy | C0 |
| 10 | Online | Documentaries | C0 |
| 11 | DVD | Comedy | C1 |
| 12 | DVD | Entertainment | C1 |
| 13 | Online | Entertainment | C1 |
| 14 | Online | Documentaries | C1 |
| 15 | Online | Documentaries | C1 |
| 16 | Online | Documentaries | C1 |
| 17 | Online | Documentaries | C1 |
| 18 | Online | Entertainment | C1 |
| 19 | Online | Documentaries | C1 |
| 20 | Online | Documentaries | C1 |

* 1. Compute the Entropy for the overall collection of training examples.

Entropy:

entropy

P(C0) = 10/20 and P(C1)=10/20

Entropy : – Σ p (C0)\*log(C0) – p(C1)\*log(C1)

= – (10/20) \* log(10/20) – (10/20) \* log (10/20)

= – (0.5) \* 0.5 – (0.5) \* 0.5

= 1

* 1. Compute the Entropy for the Movie ID attribute.

Ans:

entropy

H(Movie Id) = -Σ 20\*log 20

However, Movie ID attributes are not similar to calculate the enrtropy for the same.

Thus, Entropy for Movie ID = 0.

* 1. Compute the Entropy for the Format attribute.

Ans:

Node Count Total

DVD 8 20

Online 12 20

However for the Training examples it is:

For DVD:

– Σ (6/8)\*log (6/8) + (2/8) \* log( 2/8)

= 0.311 +0.5

= 0.811

For Online:

– Σ (4/12) \* log (4/12) + (8/12) \* log (8/12)

= 0.528 + 0.390

= 0.92

Entropy for the Format is:

= [(8/20)\*entropy for DVD] + [(12/20)\*entropy for Online]

= 0.4 \* (0.811) + 0.6 \* 0.92

= 0.8752

* 1. Compute the Entropy for the Movie Category attribute using multiway split.

Nodes Count Total

Entertainment 4 20

Comedy 8 20

Documentation 8 20

Entertainment -> C0 C1

1. 3

Comedy -> C0 C1

7 1

Documentation-> C0 C1

1. 6

Entropy for Entertainment:

= – ( (1/4) \* log(1/4) + (3/4) \* log(3/4))

= - 0.25\*(-2) + 0.75 \* (-0.415)

= 0.5 + 0.31125

= 0.81125

Entropy for Comedy:

= – ( (7/8) \* log(7/8) + (1/8) \*log (1/8))

= -0.875 \* (-0.1926) + 0.125 \* (-3)

= 0.1685 + 0.375

= 0.5435

Entropy for Documentation:

– ( (2/8) \* log(2/8) + (6/8) \* log(6/8))

-0.25 \* (-2) + 0.75 \* (-0.415)

0.5 + 0.31125

= 0.81125

Σ(4/20)\*Entropy for Entertainment + (8/20)\*Entropy for Comedy + (8/20)\*Entropy for Documentation:

0.2\* 0.81125 + 0.4\*0.5435+ 0.4\* 0.81125

0.16225 + 0.2174 + 0.3245

0.70415

* 1. Which of the three attributes has the lowest Entropy?

Ans: The Attribute with the lowest Entropy is Movie Id (which is 0).

* 1. Which of the three attributes will you use for splitting at the root node? Briefly explain your choice.

I will use the attribute –Movie category to split the root node. Movie category has most discrete variables which will help in splitting the variables of other categories. The entropy of Movie Category tells the variability of the class better and therefore it is a better candidate for being the root node .Movie id is not a good choice even though it has the lowest entropy because record numbers is very less for each partition to make any choice. .

.

1. [19 points: 6+5+8] Consider the decision tree shown in Figure 1, and the corresponding training and test sets in Tables 2 and 3 respectively.
   1. Estimate the generalization error rate of the tree using both the optimistic approach and the pessimistic approach. While computing the error with pessimistic approach, to account for model complexity, use a penalty value of 2 to each leaf node.

Solution:

Penalty for Each leaf node = 2

Thus Optimistic Error Calculation = 0

Pessimistic Error = [0 + (6\*2)]/15 = 0.8

* 1. Compute the error rate of the tree on the test set shown in Table 3.

Error = 4+3+0+0+0+0/15

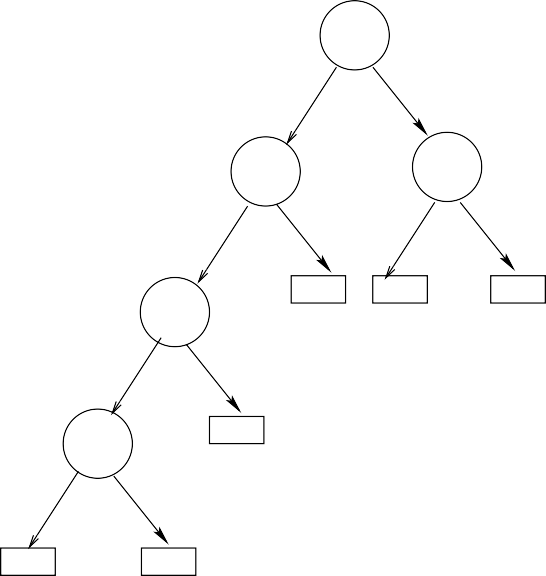
=0.47

* 1. Comment on the behavior of training and test set errors with respect to model complexity. Comment on the utility of incorporating model complexity in building a predictive model.

Solution

* When the tree is pruned more we see an increase of optimistic generalization error on the training set.
* Error on the test decreases to 0 when we prune level 3 of the data set.
* This shows overfitting problem.
* As we prune the data more it results in increase or errors for both test and training data.
* Error p = [2 + (4\*2)]/15 = 2/3
* In this case the original tree suffers from over fitting , which can be easily seen from increase in generalization error on the next data set.
* Small tress would prune data.
* Its better to comporate model complexity through pessimistic error rate ,thus allowing simple tree to be selected.

# level 1



A

0

1

B

B

0

1

−

0

+

1

−

C

0

1

D

−

0

1

+

−

level 2

level 3

level 4

level 5

Figure 1: Decision tree for Exercise 2.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| A | B | C | D | Number of + instances | Number of - instances |
| 0 | 0 | 0 | 0 | 4 | 0 |
| 0 | 0 | 0 | 1 | 0 | 1 |
| 0 | 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 0 | 3 | 0 |
| 1 | 1 | 0 | 1 | 0 | 5 |

Table 2: Training set for Problem 2

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| A | B | C | D | Number of + instances | Number of *−* instances |
| 0 | 0 | 0 | 1 | 4 | 0 |
| 0 | 0 | 1 | 1 | 3 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 |
| 0 | 1 | 1 | 0 | 0 | 2 |
| 1 | 0 | 0 | 0 | 2 | 0 |
| 1 | 0 | 0 | 1 | 3 | 0 |

Table 3: Test set for Problem 2

Attributes

Class B

Class A

Class B

Records

Discriminating

Attributes

Noise Attributes

Class A

Class B

Attribute Y

Attribute X

* 1. Synthetic data set 1. (b) Synthetic data set 2.

Figure 2: Data sets for Question 3

1. [20 points: 10 for each part] Given the data sets shown in Figure 2, explain how the decision tree and k-nearest neighbor (k-NN) classifiers would perform on these data sets.

Ans: (a) For the first case- Decision tree will perform well on this data set because there are attributes which can be distinguished from each other .When calculation entropy gain they have better discriminating power than k-nearest neighbor.

However, the k-nearest neighbor algorithm will not perform as well due to comparatively a large number of noise attributes. To identify similarities in K-nearest will be difficult as it cannot handle noise and there are no similarities between attributes which can be used to create separate class.

(b) For the second case, K-nearest neighbour will work the best due to the proximity of the attributes of the same class to each other. Decision tree will have to be large in order to capture the circular decision boundaries, and thus is not the ideal solution.

1. [16 points: 4 for each part] Consider the problem of predicting if a given person is a defaulted borrower (DB) based on the attribute values:

* Home Owner = Yes, No
* Marital Status = Single, Married, Divorced
* Annual Income = Low, Medium, High
* Currently Employed = Yes, No

Suppose a rule-based classifier produces the following rules:

* Home Owner = Yes *→* DB = Yes
* Marital Status = Single *→* DB = Yes
* Annual Income = Low *→* DB = Yes
* Annual Income = High, Currently Employed = No *→* DB = Yes
* Annual Income = Medium, Currently Employed = Yes *→* DB = No
* Home Owner = No, Marital Status = Married *→* DB = No
* Home Owner = No, Marital Status = Single *→* DB = Yes

Answer the following questions. Make sure to provide a brief explanation or an example to illustrate the answer.

1. Are the rules mutually exclusive?
2. Is the rule set exhaustive?
3. Is ordering needed for this set of rules?
4. Do you need a default class for the rule set?

Ans:

* 1. No the rules are not mutually exclusive.

The reason being there are conflict between instances and they are not independent.

The instance {Home Owner = Yes, Marital Status = Single} will trigger the ﬁrst two rules and thus the rules are not mutually exclusive.

* 1. No the rule set is not exhaustive since the rule set has more than one attribute for one category.

Also, for the instance {Marital Status = Divorced, Home Owner = No, Annual Income = High, Currently Employed = Yes} is not covered by any of the rules which makes them non-exhaustive.

* 1. Ordering is needed as the rules in this data set are not mutually exclusive. Ordering the rule set will a make sure that the set would be mutually exclusive. The record can match two or more rules that give conﬂicting predictions about the class. For example, the instance Home Owner=Yes, Marital Status=Single will trigger rule 1 which is the prediction: Default Borrower=Yes and rule 5 (prediction: Default Borrower=No).

If you do not tell the system to prefer one rule to another (i.e., order them), the system will not know how to classify the instance.

(d) Yes we will need a default class as the rules are not mutually exclusive

1. [15 points] Consider the problem of predicting whether a movie is popular given the following attributes: Format (DVD/Online), Movie Category (Comedy/Documentaries), Release Year, Number of world-class stars, Director, Language, Expense of Production and Length. If you had to choose between RIPPER and a *k*-nearest neighbor classifier, which would you prefer and why? Briefly explain why the other one may not work so well?

Ans: RIPPER is preferred over the K-nearest algorithm in this problem, since the variables are diverse in nature and thus are expected to be of varying relevance for differentiating between the“popular”and“not popular” classes. The K-nearest performance can be adversely impacted by irrelevant attributes, as they can unduly inﬂuence the similarity function. RIPPER, like most other rule-based models, as well as decision tree classiﬁers, performs variable selection using measures such as the Gini index that, although not perfect, allow irrelevant variables to be discarded. Decision tree fails to perform when there too many attributes which are unrelated to each other and contains values which affect the decision making process. In the above attributes each attribute can be degenerated into many categories which RIPPER algorithm can easily distinguish but decision tree will fail.