S228/401C

DUBLIN INSTITUTE OF TECHNOLOGY
KEVIN STREET, DUBLIN 8

BSc. (Hons) in Computer Science

Stage 4

SEMESTER 2 EXAMINATIONS 2011/2012

*** *SOLUTIONS* ***

ARTIFICIAL INTELLIGENCE II

Dr. John Kelleher
Dr. Deirdre. Lillis
Mr. D. Tracey

Monday
14^{th} May 2012
4:00 p.m to 6:00 p.m

Question 1 is **compulsory**

Answer Question 1 (40 marks) **and**

any 2 Other Questions (30 marks each).

****** SOLUTIONS ******

****** SOLUTIONS ******

1. (a) Explain what is meant by **inductive learning**.

(5 marks)

Inductive Learning involves the process of learning by example where a system tries to induce a general rule from a set of observed instances

(b) For some data sets it is possible to devise multiple hypotheses that are consistent with the data. Describe a heuristic for choosing among multiple consistent hypotheses and explain why your heuristic is reasonable.

(5 marks)

One answer is to use Occams razor (sometimes called Ockhams razor) : prefer the hypothesis that maximizes a combination of simplicity and consistency with the data. This makes sense, because hypotheses that are no simpler than the data themselves are failing to extract any pattern from the data. Defining simplicity is not easy but it seems reasonable to say that a degree-1 polynomial is simpler than a degree-12 polynomial.

(c) Describe the differences between **lazy learners** and **eager learners**, giving examples of each.

(10 marks)

Definitions:

Lazy learners do not try to build a model from the training data, but simply use it at classification time

Eager learners build a mode from the training data during training, and use only this model at classification time, ignoring the original data.

Key differences:

- Lazy methods may consider query instance when deciding how to generalise beyond the training data D; eager methods cannot since they have already chosen global approximation when seeing the query.
- **Efficiency** lazy learners require less training times but more time at prediction; eager learners require more training times by less time for prediction
- **Accuracy** lazy learners effectively uses a richer hypothesis space since it uses many local linear functions to form its implicit global approximation to the target function; eager learners must commit to a single hypothesis that covers the entire instance space.
- It is easier for lazy learners to handle **concept drift**

Examples:

Lazy learning example : case based reasoning

Eager learning example : Decision=tree, neural networks, support vector machines

(d) Let us say we have three classification algorithms. How can we order these three from best to worst?

(20 marks)

This is a discursive question so giving a precise answer is not appropriate. However, key points that the student should touch on include:

- Predictive accuracy
- Speed and scalability
  - Time to construct the model
  - Time to use the model
- Robustness (handling noise and missing values)
- Scalability
- Interpretability (understanding and insight provided by the model)

It should be noted also, that these evaluation criteria are application dependent.

Table 1: Class Exam Results: a 1 indicates the student possesses the feature listed in the column and 0 indicates that they do not. The final column lists whether or not the student was awarded a 1^{st} this year

Student	1^{st} last year	Works hard?	Blonde	1^{st} this year
X	1	1	0	1
Y	0	1	1	0
Z	0	1	0	0

Table 2: The attributes of the student whose script was not marked. A 1 indicates the student possesses the feature listed in the column and 0 indicates that they do not. The column on the right contains a ? because they have not been graded yet.

Student	1^{st} last year	Works hard?	Blond	1^{st} this year
U	1	0	0	?

2. (a) Discuss the advantages and disadvantages of ***k*-Nearest Neighbour** classification.

(5 marks)

Strengths

(i) No training involved lazy learning

(ii) New data can be added on the fly

(iii) Some explanation capabilities

(iv) Robust to noisy data by averaging k-nearest neighbors

Weaknesses

(i) Not the most powerful classification (generally its accuracy will be lower than an ANN or SVM model)

(ii) Slow classification

(iii) Curse of dimensionality (as you increase the number of features you need more and more examples to cover the problem space - kNN are particularly susceptible to this issue as they do not do any feature selection).

(b) Just before an exam board a lecturer finds an exam script by student U that the lecturer had forgotten to mark. The lecturer does not have time to correct the script before the exam board, so they decide to use a nearest-neighbour approach to decide whether or not to award student U a 1^{st}. The case base of results the lecturer used is listed in Table 1 and the attributes of student U are listed in Table 2 .

(i) Assuming the lecturer uses Euclidean distance

$$d(x_1, x_2) = \sqrt{\sum_{r=1}^{n}(a_r(x_1) - a_r(x_2))^2}$$

as their distance metric, compute the distance between the student U and each of the students in the case base.

(5 marks)

Student	1^{st} last year	Works hard?	Blond	1^{st} this year	Distance
X	1	1	0	1	1.00
Y	0	1	1	0	1.732
Z	0	1	0	0	1.414

(ii) Given that the lecturer used ***1*-NN** classification was student U awarded a 1^{st}?

(5 marks)

Using Euclidean distance as a measure of distance student X is the closest instance in the case base to student U. Consequently, using $1-NN$ classification student U would be awarded the same result as student X. So student U would be awarded a 1^{st}

(iii) If the lecturer used ***3*-NN** classification would student U be awarded a 1^{st}?

(5 marks)

If the lecturer used $3-NN$ all the instance in the database would be considered and the student would be awarded the most frequent occurring classification in the case base. In this instance, the student would not be awarded a 1^{st} because two out of the three students in the case base were not awarded 1^{st}s.

(c) Table 3, on the next page lists a classification dataset. Each instance in the dataset has two explanatory attributes (attribute 1 and attribute 2) and is classified as either a positive (+) or a negative(-) example.

(i) Calculate the classification **entropy** for this dataset.

(5 marks)

Entropy is $-\frac{3}{5}log_2\frac{3}{5} - \frac{2}{5}log_2\frac{2}{5} = 0.971$

(ii) Calculate the **information gain** for attribute 1 and attribute 2.

(5 marks)

Entropy for attribute 1 = T $-\frac{3}{4}log_2\frac{3}{4} - \frac{1}{4}log_2\frac{1}{4} = 0.811$
Entropy for attribute 1 = F $0 - \frac{1}{1}log_2\frac{1}{1} = 0$
Gain for attribute 1 $0.971 - (\frac{4}{5} \times 0.811 + \frac{1}{5} \times 0) = 0.322$
Entropy for attribute 2 = T $-\frac{2}{3}log_2\frac{2}{3} - \frac{1}{3}log_2\frac{1}{3} = 0.918$
Entropy for attribute 2 = F $-\frac{1}{2}log_2\frac{1}{2} - \frac{1}{2}log_2\frac{1}{2} = 1.0$
Gain for attribute 2 $0.971 - (\frac{3}{5} \times 0.918 + \frac{2}{5} \times 1) = 0.02$

Attribute 1	Attribute 2	Classification
T	T	+
T	F	-
T	F	+
T	T	+
F	T	-

Table 3: Classification Dataset

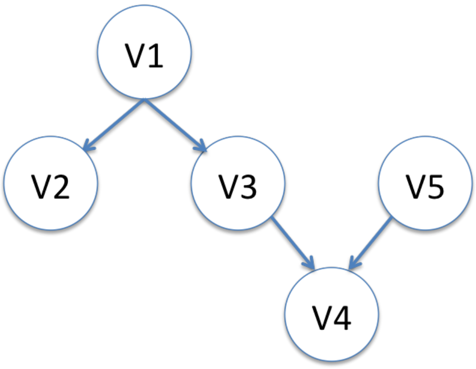

Figure 1: An example Bayesian network.

Table 4: Full joint distribution for a dentist visit

	$toothache$		$\neg toothache$	
	$catch$	$\neg catch$	$catch$	$\neg catch$
$cavity$	0.108	0.012	0.072	0.008
$\neg cavity$	0.016	0.064	0.144	0.576

3. (a) Given the full joint distribution shown in Table 4, calculate $\mathbf{P}(Toothache|cavity)$.

(5 marks)

This asks for the vector of probability values for $Toothache$, given that $Cavity$ is true.
$P(Toothache|cavity) = \langle \frac{0.108+.012}{0.2}, \frac{0.072+0.008}{0.2} \rangle = \langle 0.6, 0.4 \rangle$

(b) Express the joint probability distribution for the Bayesian network shown in Figure 1 using the chain rule.

(10 marks)

The chain rule is:

$$P(x_1, \ldots, x_n) = \prod_{i=1}^{n} P(x_i | x_{i-1}, \ldots, x_1)$$

In a properly constructed bayesian network:

$$parents(X_i) \subseteq \{x_{i-1}, \ldots, x_1\}$$

Therefore, in a properly constructed Bayesian network:

$$P(x_1, \ldots, x_n) = \prod_{i=1}^{n} P(x_i | x_{i-1}, \ldots, x_1) = \prod_{i=1}^{n} P(x_i | parents(X_i))$$

So the joint probability distribution for the given network is:

$$P(V4|V3, V5)P(V3|V1)P(V2|V1)P(V5)P(V1)$$

(c) Consider the following time keeping patterns of the lecturers in your college:

- 25% of lecturers start 75% of their lectures on time and 25% late.
- 50% of lecturers start 50% of their lectures on time and 50% late.
- 25% of lecturers start 25% of their lectures on time and 75% late.

(i) Given that both the 1^{st} and 2^{nd} Artificial Intelligence lectures of the year started on time, compute the posterior probability that your Artificial Intelligence lecturer follows each of the three time keeping patterns.

(10 marks)

To begin we will define some notation. Let:

- h_1 denote the hypothesis that your AI lecturer starts 75% of their lectures on time $P(h_1) = 0.25$.
- h_2 denote the hypothesis that your AI lecturer starts 50% of their lectures on time $P(h_2) = 0.50$.
- h_3 denote the hypothesis that your AI lecturer starts 25% of their lectures on time $P(h_3) = 0.25$.

Also, if we use the notation $ontime_x$ to represent the observation that a lecture x started on time, then the probability of any given AI lecture starting on time given a particular hypothesis h is:

- $P(ontime_x|h_1) = 0.75$.
- $P(ontime_x|h_2) = 0.50$.
- $P(ontime_x|h_3) = 0.25$.

Then:

- By Bayes' rule, we can compute the posterior probability of a hypothesis given the data so far using:

$$P(h_i|\mathbf{d}) = \alpha P(\mathbf{d}|h_i)P(h_i)$$

- And, the likelihood of the data given a hypothesis is calculated using:

$$P(\mathbf{d}|h_i) = \prod_j P(d_j|h_i)$$

So:

- $P(h_1|ontime_1, ontime_2) = \alpha(\prod_{j=1}^{2} P(ontime_j|h_1))P(h_1) = \alpha 0.75^2 \times 0.25 = \alpha 0.140625 = \frac{0.140625}{0.28125} \approx 0.5$
- $P(h_2|ontime_1, ontime_2) = \alpha(\prod_{j=1}^{2} P(ontime_j|h_2))P(h_1) = \alpha 0.50^2 \times 0.50 = \alpha 0.125 = \frac{0.125}{0.28125} \approx 0.4444$
- $P(h_3|ontime_1, ontime_2) = \alpha(\prod_{j=1}^{2} P(ontime_j|h_3))P(h_1) = \alpha 0.25^2 \times 0.25 = \alpha 0.015625 = \frac{0.015625}{0.28125} \approx 0.0556$

(ii) Given that both the 1^{st} and 2^{nd} Artificial Intelligence lectures of the year started on time, what is the Bayesian Prediction that the 3^{rd} Artificial Intelligence lecture will start on time?

(5 marks)

Bayesian predictions use a likelihood-weighted sum over the hypotheses:

$$P(X|\mathbf{d}) = \sum_i P(X|h_i)P(h_i|\mathbf{d})$$

In this instance we get:

$$\begin{aligned} P(ontime_3|\mathbf{d}) &= \sum_i P(ontime_3|h_i)P(h_i|\mathbf{d}) \\ &= (0.75 * 0.5) + (0.5 * 0.4444) + (0.25 * 0.0556) \\ &= 0.375 + 0.2222 + 0.0139 \\ &\approx 0.6111 \end{aligned}$$

x	2	4	6	8
y	2	5	5	8

Table 5: Example Dataset for Linear Regression Question

4. (a) Assuming a domain with one explanatory variable x and one dependent variable y linear regression uses the following formula to model the relationship between the explanatory and dependent variable:

$$f(x) = w_1 x + w0$$

where $w1$ and $w0$ are computed using the following formulae (where M is number of data points in the dataset):

$$w_1 = \frac{(M \sum_{i=1}^{M} x_i y_i) - (\sum_{i=1}^{M} x_i \sum_{i=1}^{M} y_i)}{(M \sum_{i=1}^{M} x_i^2) - (\sum_{i=1}^{M} x_i)^2}$$

$$w_0 = (\frac{1}{M} \sum_{i=1}^{M} y_i) - (\frac{w_1}{M} \sum_{i=1}^{M} x_i)$$

Using the data in Table 5 compute the values of w_0 and w_1 that provide the best linear fit to the data.

(10 marks)

First we need to compute the values of the equation components:

- M = 4
- $\sum_{i=1}^{M} x_i y_i = 4 + 20 + 30 + 64 = 118$
- $\sum_{i=1}^{M} x_i = 2 + 4 + 6 + 8 = 20$
- $\sum_{i=1}^{M} y_i = 2 + 5 + 5 + 8 = 20$
- $\sum_{i=1}^{M} x_i^2 = 4 + 16 + 36 + 64 = 120$
- $(\sum_{i=1}^{M} x_i)^2 = 20^2 = 400$

Given these values, w_1:

$$w_1 = \frac{(4*118) - (20*20)}{(4*120) - 400} = \frac{72}{80} = 0.9$$

And, w_0:

$$w_0 = (\frac{1}{4} * 20) - (\frac{0.9}{4} * 20) = 5 - 4.5 = 0.5$$

(b) Figure 2 shows a backprogation network that is currently processing the training vector $[1.0, 0.9, 0.9]$ that has an associated target vector $[0.1, 0.9, 0.1]$. Given that the output from unit B is 0.6 and from C is 0.8, and assuming that the activation function used at all nodes in the network is the logistic function (i.e., $f(x) = \frac{1}{1+\exp^{-x}}$):

(i) Calculate the actual output vector (to 3 decimal places).

(5 marks)

Output of unit $i = f(\sum_{j=1}^{n} W_{j,i} \times activation_j)$
First output unit input = -0.3 x 0.6 + 0.9 x 0.8 = 0.54 → f(0.54) = 0.632
Second output unit input = -0.6 x 0.6 + -0.1 x 0.8 = -0.44 → f(-0.44) = 0.392
Third output unit input = 0.4 x 0.6 + 1.2 x 0.8 = 1.2 → f(1.2)= 0.769

(ii) Calculate the error for each output unit.

(5 marks)

Error = target - output
First output unit = (0.1 - 0.632) = - 0.532
Second output unit = (0.9 - 0.392) = 0.508
Third output unit = (0.1 - 0.769) = - 0.696

(iii) Calculate the error for each hidden unit B and C.

(10 marks)

Each hidden node j is responsible for some fraction of the error Err_i of each of the output units i to which it connects. Thus the Err_i values are divided according to the strengths of the connection between the hidden node and the output nodes and are propagated back to the hidden nodes. Where a hidden node feeds-forward into more than 1 output node the errors propagated back to it are summed: $Err_j = \sum_{i=1}^{n} W_{ji} \times Err_i$:
$Err_B = (-0.3 \times -0.532) + (-0.6 \times 0.508) + (0.4 \times -0.696) = 0.1596 + -0.3048 + -0.2784 = -0.4236$
$Err_C = (0.9 \times -0.532) + (-0.1 \times 0.508) + (1.2 \times -0.696) = -0.4788 + -0.0508 + -0.8352 = -1.3648$

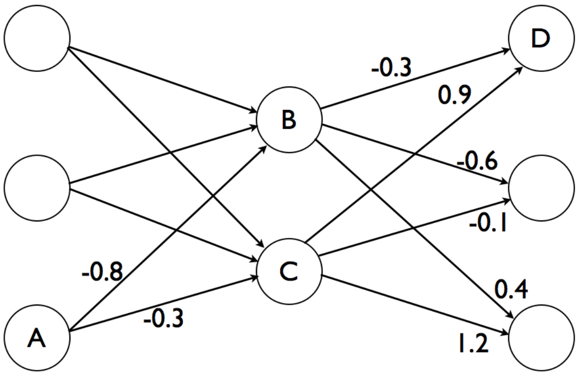

Figure 2: Example Neural Net