A30406

Calculators may be used in this examination provided they are not capable of being used to store alphabetical information other than hexadecimal numbers.

UNIVERSITY OF
BIRMINGHAM

School of Computer Science

Third Year - BSc Artificial Intelligence and Computer Science
Third Year – BEng Computer Systems Engineering
Third Year – BSc Computer Science
Third Year – MSci Computer Science
Third Year – MEng Computer Science/Software Engineering
Third Year – BSc Mathematics and Computer Science
Third Year – MSci Mathematics and Computer Science
Third Year – BSc Computer Science with Study Abroad
Third Year – BSc Computer Science with Business Management
Third Year – BSc Computer Science with Industrial Year
Third Year – MEng Computer Science/Software Engineering with Industrial Year
Third Year – BSc Computer Science with Business Management with Industrial Year
Third Year – MSci Computer Science with Industrial Year

06 26428

Machine Learning

Summer May/June Examinations 2016

Time allowed: 1 hour 30 minutes

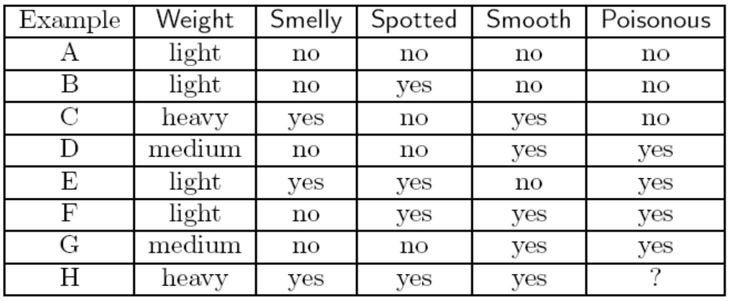
[Answer ALL Questions]

Non Alpha only

1. (a) Suppose that in answering a question on a true/false test, an examinee either knows the answer with probability p or s/he guesses with probability 1-p. Assume that if the examinee knows the answer to a question, the probability that s/he gives the correct answer is 1, and if s/he guesses then s/he only gives the correct answer with probability 0.5.

Use Bayes rule to compute the probability that an examinee knew the answer to a question given that s/he has correctly answered it. [5%]

(b) You are stranded on a deserted island. Mushrooms of various types grow wildly all over the island, but no other food is anywhere to be found. Some of the mushrooms have been determined as poisonous and others as not. You are the only one remaining on the island. You have the following data to consider:



Example	Weight	Smelly	Spotted	Smooth	Poisonous
A	light	no	no	no	no
B	light	no	yes	no	no
C	heavy	yes	no	yes	no
D	medium	no	no	yes	yes
E	light	yes	yes	no	yes
F	light	no	yes	yes	yes
G	medium	no	no	yes	yes
H	heavy	yes	yes	yes	?

(i) Using this data and the Naive Bayes classifier, what is the prediction on mushroom H? Show all your working. [10%]

(ii) Suppose you have no chance to collect any more training examples, would you recommend dropping the Naive Bayes assumption to achieve better accuracy, or would you rather keep the Naive Bayes assumption? Explain your reasons. [5%]

(iii) Explain, how could you construct a Naive Bayes classifier to solve the same problem as in (i), if some of the attributes (for example, 'Weight', and/or 'Smelliness') would be given as continuous-valued measurements? [5%]

(c) Answer the following questions about the Gaussian classifier:

(i) Is it possible for a Gaussian classifier to implement a non-linear decision boundary? If so, draw an example. If not, explain why not. [5%]

(ii) How about a Gaussian Naive Bayes classifier? Justify your answer. [5%]

2. (a) The XOR problem is to learn the function from the following input points to their class labels:
Class -1: (1,1), (-1,-1)
Class +1: (1,-1), (-1,1).
We know that support vector machine (SVM) with a kernel can solve this problem by mapping the points in a higher dimensional space. But higher dimensional spaces are difficult to visualise, and we would like to construct a support vector machine that classifies these points correctly in a 2-dimensional input space. Is this possible? If so, how? If not, explain why not. [10%]
Hint: Try to come up with a feature-transformation that stays in 2D (i.e. maps R^2 into R^2) and makes the classes linearly separable.

(b) Can you construct a nearest-neighbour classifier that has zero leave-one-out error on the XOR problem? If so, show precisely how? If not, explain why not. [10%]
Hint: You are free to define the similarity (or distance).

(c) Suppose you do Adaboost on a 2-dimensional data set, where your weak learners can only have horizontal or vertical separation lines. Explain how the ensemble can produce a nonlinear boundary. You can draw an example to help you explain. [5%]

3. Consider a finite set of functions, *H*, that map an input set *X* into the set of labels *{0,1}*. Let *L* be an algorithm that for any function c from *H*, and any training set *S* of *N* training points, drawn independently from some unknown distribution *D* over *X*, returns a hypothesis, $h_{S,}$ that is consistent (it has zero training error). Under these conditions it is known (as proved in the class) that for any choice of $\varepsilon > 0$ the generalisation error of h_S is upper bounded by ε with probability of at least $1-|H|exp(-N\varepsilon)$. Here, |*H*| denotes the cardinality of a set (that is the number of elements in the set *H)*.

   (a) What is the random variable in the above probability statement? [5%]

   (b) Does the above mentioned theorem mean any of the following?

      (i) If the classifier learned on a given training set S has zero error on the set S it was trained on, then it is likely to have true generalisation error less than ε outside that particular training set S

      (ii) Given a large enough training set, it is likely that the learner with either return a classifier that generalises well, or is unable to find a classifier with zero training error.

      (iii) None of the above. Justify your answer

[5%]

4. (a) Suppose you have collected a dataset of already-classified instances and you have built a classifier. Maybe your classifier is a probabilistic one using the joint probability distribution; or maybe it's a Naive Bayes classifier; or maybe it uses kNN; or maybe it works in some other way. How will you know how good your classifier is? Describe 2 methods that you can use in practice to answer this question. [5%]

(b) The curse of dimensionality generically refers to problems associated with data that is high dimensional, i.e. data that has a large number of attributes. Suggest an approach to mitigate at least one of these problems. Say which problem can be mitigated by your suggested approach. [5%]

(c) Your friend suggests that machine learning is taught in a bad way because instead of trying to cover many classification methods you should be taught only the method that works best. Can you come up with an argument or an example to illustrate and explain your friend that no method can be best on all tasks? For example you might like to construct a data set of up to 10 points in 2 classes, on which one method would fail and another method would succeed, and then construct another data set on which the latter method fails and the former method succeeds. [10%]

5. a) Point out one similarity and one difference between the EM algorithm for Gaussian mixtures, and the k-means clustering algorithm. [5%]

b) Suppose you have run k-means clustering on a data set and later you get more data points into the same data set. How would you cluster the new points without re-running the algorithm? [5%]