COL 774 Quiz

Apr 19, 2015

Notes:

- **Time: 4:00 pm to 4:40 pm. Maximum Points: 12**
- **This exam is closed books/notes.**
- **There is a total of three short answer problems (worth 4 points each).**
- **Start every question on a new page on your answer sheet.**
- **Clearly write the question number on top of the page.**
- **Justify all your answers. Answers without justification may not get any points.**

1. Derive the computational complexity of k-means algorithm in terms of m, n, k and T, where m is the number of data points, n is the number of dimensions, k is the number of clusters and T is the number of iterations for which algorithm is run. You should justify the steps of your derivation to get full points.

2. Suppose we are given a set of points $\{x^{(1)}, \ldots, x^{(m)}\}$ in $\mathcal{R}^n$. Let us assume that we have pre-processed the data to have zero-mean. Consider applying PCA on this data. In an alternate formulation of PCA, to find the first principal component, we look for the direction which minimizes the mean squared distance between the projected and the original points. Show that this is equivalent to the original PCA interpretation in which we find the direction maximizing the variance of the projected data. You should not assume any expression for the variance as done in class and derive it from first principles.

3. Consider the Bayesian network shown in Figure 1. Suppose you have a training set composed of the examples given in Table 1, with "?" indicating a missing value. Show the first iteration of the EM algorithm (initial parameters, E-step, M-step), assuming the parameters are initialized ignoring missing values.

Figure 1: A Bayesian Network

A	B
0	1
0	1
0	1
?	0
?	1
1	1
1	0
1	0

Table 1: Data with Missing Values

COL 774 Minor 2

Mar 21, 2015

Notes:

- **Time: 1:00 pm to 2:00 pm. Maximum Points: 24**
- **This exam is closed books/notes.**
- **There is a total of six short answer problems (worth 4 points each).**
- **Start every question on a new page on your answer sheet.**
- **Clearly write the question number on top of the page.**
- **Justify all your answers. Answers without justification may not get any points.**

1. Consider a machine learning problem with input feature vectors $x \in R^n$ where each x lies on a unit sphere in R^n (i.e., $||x|| = 1$). Given two feature vectors $x, z \in \mathcal{R}^n$ (and satisfying the above property), consider the function $K(x,z)$ defined as $K(x,z) = ||x+z||^2$. Using Mercer's theorem, show that $K(x,z)$ is a valid kernel. Note: In order to get any credit, your solution is required to make use of the Mercer's theorem.

2. One way to avoid overfitting in decision trees is to prune the tree using a separate validation set. Typically, a full-blown tree is learnt on the training set first. This is followed by iterative pruning of the learned tree until further pruning does not lead to decrease in error on the validation set. An alternative approach is to keep checking error on the validation set while the tree is being constructed. The tree construction is stopped when the error (on the validation set) does not decrease any further. Which of these approaches do you think would work better in general. Why?

3. Consider the Naïve bayes model with Boolean valued features and binary class labels. Show that $P(y=1|x)$ takes the form of a logistic function i.e. $P(y=1|x) = \frac{1}{1+e^{-\theta^T x}}$. Clearly express θ in terms of the parameters of the Naïve bayes model. Do not forget to include the intercept term in θ.

4. Let X and Y be two discrete valued random variables. Let H be the entropy function as defined in class. Let $H(X)$ and $H(Y)$ denote the entropy of the random variables X and Y, respectively. Let $H(Y|X)$ denote the conditional entropy of Y given X. Prove that $H(Y|X) = H(X|Y) + H(Y) - H(X)$. You should prove it from first principles and not use any existing facts about entropy. Note: This is the entropy analagoue of the Bayes rule for probabilities.

5. Draw a neural network to represent the Boolean function $f(x_1, x_2, x_3) = (x_1 \vee x_2 \vee x_3) \wedge (\neg x_1 \vee x_2 \vee \neg x_3)$ defined over 3 input variables. Note that $x_1, x_2, x_3 \in \{0,1\}$. Here, $\wedge$ denotes the *and* operator. $\vee$ denotes the *or* operator and $\neg$ denotes the *negation*, as in the standard Boolean algebra. Your network should have at most one hidden layer and use at most 3 network units. Use the threshold function $g(x) = \mathbb{1}\{x \geq 0\}$ ($\mathbb{1}$ represents the indicator function) to process the output of each of the units. Clearly specify the interconnections and the associated weights. Also argue briefly why your construction is correct.

6. Assume you have a biased coin with probability of heads given by the parameter ϕ. Consider m independent tosses of this coin resulting in a sequence with p heads and n tails ($n+p=m$). Note that the observed data D here corresponds to the sequence of heads and tails as described above. Let the prior distribution over ϕ be given by $\phi \sim Beta(2,2)$ [1]. Calculate the expected value $E[\phi]$ of the parameter ϕ under the posterior distribution $P(\phi|D)$ in terms of n and p. You can use the fact that for k_1, k_2 positive integers, $\int_0^1 \phi^{k_1}(1-\phi)^{k_2} d\phi = \frac{k_1! k_2!}{(k_1+k_2+1)!}$

[1] Recall $\phi \sim Beta(\alpha, \beta)$ means that $P(\phi) \propto \phi^{\alpha-1}(1-\phi)^{\beta-1}$