Qualifying Exam in Machine Learning

October 20, 2009

Instructions: Answer two out of the three questions in Part 1. In addition, answer two out of three questions in two additional parts (choose two parts out of models and algorithms, decision processes, and learning theory). Please write clearly and support all your arguments as much as possible.

Part 1: Core

1. (a) “Machine learning methods always result in optimization problems.” True or false? Discuss why or why not. Name and discuss specific methods to support your argument.

(b) Name at least two examples of common machine learning methods whose objective functions are convex. In practical terms, what are the advantages and disadvantages of convex optimizations in machine learning?

(c) Commonly, in many machine learning methods, most of the parameters are chosen by an optimization routine, while a small number are chosen using a ‘model selection’ approach (most commonly, k-fold cross-validation). Give at least two examples of this.

(d) Can model selection for k-fold cross-validation be considered a kind of optimization?

(e) In each of the two examples, could/should all of the parameters be chosen by a single optimization routine, in principle? In each of the two examples, could/should all of the parameters be chosen by cross-validation, in principle? Discuss the theoretical and practical limits of what is possible.

2. Consider the situation where you observe a sequence of continuous random variables $X^{(1)}, \ldots, X^{(n)} \sim p_\theta$ in order to estimate $\theta$ using an estimator $\hat{\theta}_n(X^{(1)}, \ldots, X^{(n)})$. A common way to measure the prediction quality is using $L_r = E(|\theta - \hat{\theta}_n|^r)$.

(a) Write the expectation in $L_r$ explicitly.

(b) Name one reason to prefer $L_r$ with $r = 2$ over $r = 1$ and one reason to prefer $L_r$ with $r = 1$ over $r = 2$.

(c) Show mathematically that $L_2$ decomposes into a squared bias term and a variance term. Explain intuitively what roles these two quantities play.

(d) Show that in the case of $\theta = EX$ and $\hat{\theta}_n = \frac{1}{n} \sum_{i=1}^{n} X^{(i)}$, the bias and the variance go to 0 as $n \to \infty$. 

1
3. (a) What is the bias-variance tradeoff in machine learning?
(b) What is overfitting? What are the contributing factors to it?
(c) What is the relationship between the bias-variance tradeoff and overfitting? Describe a well known method for resolving the tradeoff and avoiding overfitting in each of the following models: decision trees, logistic regression, and neural networks.

Part 2: Models and Algorithms

1. (a) Name at least three widely-used methods for classification and at least three widely-used methods for density estimation.
(b) Many would argue that for high-dimensional data, we have good methods for classification, but not for density estimation.
(c) Do you agree? Why or why not? Name and discuss specific methods to support your argument. Is the issue more to do with the inherent hardness of each type of task, or more to do with the methods we currently have available?
(d) How could one settle this question rigorously, mathematically? How could one settle this question rigorously, empirically?

2. Consider the following exponential family model

\[ p_{\theta}(X) = \exp \left( \sum_{i=1}^{m} \theta_i f_i(X) - \log Z(\theta) \right) \]

where \( Z \) is the normalization term, and a dataset \( D = (X^{(1)}, \ldots, X^{(n)}) \) sampled iid from \( p_{\theta_0} \). Please try to be as rigorous as possible in your answers below.

(a) Express the gradient vector of \( \log p_{\theta}(X) \) using expectations, and the Hessian matrix of \( \log p_{\theta}(X) \) using covariances (in both cases the derivatives are with respect to \( \theta \)).
(b) Prove that the loglikelihood is concave.
(c) Assuming that the maximum likelihood estimator (mle) exists, prove that it can be obtained by solving for \( \theta \) in the following \( m \) equations

\[ E_{p_{\theta}}\{f_i(X)\} = \frac{1}{n} \sum_{j=1}^{n} f_i(X^{(j)}) \quad i = 1, \ldots, m. \]

(d) Assuming the model above, in some cases the mle does not exist. Explain what are these cases. Draw by hand a loglikelihood function for which this might happen. Explain why this does not contradict (b) above.
(e) One way to address the situation in (c) is to maximize instead the regularized likelihood

\[ \frac{1}{n} \sum_{i=1}^{n} \log p_{\theta}(X^{(i)}) - \alpha \sum_{j=1}^{m} \theta_j^2 \]

for some \( \alpha > 0 \). Repeat (a)-(c) for this case. Prove that in this case the situation described in (d) will never happen.
3. Consider the problem of using labeled data \((X^{(1)}, Y^{(1)}), \ldots, (X^{(l)}, Y^{(l)})\) to learn a linear classifier \(f: X \rightarrow Y, f(x) = \text{sign}(\sum_j \theta_j x_j)\) such as boosting, logistic regression, and linear SVM.

(a) Write down the exponential loss that boosting minimizes, the logistic loss that logistic regression minimizes (the negative log-likelihood) and the hinge-loss that linear SVM minimizes.

(b) It is sometimes argued that logistic regression and SVM are more resistant to outliers than boosting. Do you agree with this claim? Explain why it is correct or incorrect.

(c) Show mathematically using convex duality that minimizing the loglikelihood is equivalent to a maximum entropy problem. Write precisely the maximum entropy objective function and its constraints.

Part 3: Decision Processes

1. Machine learning algorithms have traditionally had difficulty scaling to large problems. In classification and traditional supervised learning this problem arises with data that exist in very high dimensional spaces or when there are many data points for computing, for example, estimates of conditional densities. In reinforcement learning this is also the case, arising when, for example, there are many, many states or when actions are at a very low level of abstraction.

   - Typical approaches to addressing such problems in RL include function approximation and problem decomposition. Compare and contrast these two approaches. What problems of scale do these approaches address? What are their strengths and weaknesses? Are they orthogonal approaches? Can they work well together?

   - What are the differences between hierarchical and modular reinforcement learning? Explain both the theoretical and practical limits of these approaches.

2. You have been hired as a consultant to solve a problem for a large corporation that has no machine learning researchers. The problem involves a virtual agent with a set of virtual actuators and virtual but real-valued (continuous) sensors. Your client doesn’t know how to describe what he wants the agent to do, but he definitely knows when it has done well or poorly.

   - Formally describe this as a reinforcement learning problem. Argue for this formulation, explaining why it isn’t well-modeled as a supervised learning problem, for example. Also detail your actual methodology: how will you set up the problem? How will you train? How will you know that you are doing well? What assumptions will you make? How will you choose the various free parameters of your algorithm (so, yes, please choose an algorithm)?

   - Your client has flipped through a book on unsupervised learning and so demands that you also use unsupervised learning. You are ethical, so you want to use some algorithm in a reasonable and useful way. Given the details of this particular problem, how might you actually use unsupervised learning here to some good effect? Be specific: commit to an algorithm (or two or three) and explain what you would hope to get out of its use.
3. Although modeling decision processes as Markov Decision Processes has proven useful, critics have noted that MDPs only model single agents and thus either avoid several multi-agent problems or hide serious complexity inside the transition model, making it difficult for an agent to act optimally under a variety of common conditions.

- Formally define the extension of MDPs and reinforcement learning to the multi-agent case. What are the requirements of a policy in this case?
- What are the issues that arise with such an extension? In particular, discuss the implications of extending the Bellman equation to the multi-agent case. What are the theoretical and computational problems that arise? Is this even a particular reasonable way to proceed?
- Briefly propose some solutions to the problems you identify. Briefly compare and critique these solutions.

Part 4: Learning Theory

1. PAC learning of simple classes
   
   (a) Suppose that $C$ is a finite set of functions from $X$ to $\{0,1\}$. Prove that for any distribution $D$ over $X$, any target function, and any $\epsilon, \delta > 0$, if we draw a sample $S$ from $D$ of size
   $$|S| \geq \frac{1}{\epsilon} \left[ \ln(|C|) + \ln \left(\frac{1}{\delta}\right) \right],$$
   then with probability $1 - \delta$, all $h \in C$ with $err_D(h) \geq \epsilon$ have $err_S(h) > 0$. Here, $err_D(h)$ is the true error of $h$ under $D$ and $err_S(h)$ is the empirical error of $h$ on the sample $S$.
   
   (b) Present an algorithm for PAC learning conjunctions over $\{0,1\}^n$.

2. Expressivity of LTFs.

   Assume each example $x$ is given by $n$ boolean features (variables). A decision list is a function of the form: “if $\ell_1$ then $b_1$, else if $\ell_2$ then $b_2$, else if $\ell_3$ then $b_3$, ..., else $b_m$,” where each $\ell_i$ is a literal (either a variable or its negation) and each $b_i \in \{0,1\}$. For instance, one possible decision list is the rule: “if $\overline{x}_1$ then positive, else if $x_5$ then negative, else positive.” Decision lists are a natural representation language in many settings and have also been shown to have a collection of useful theoretical properties.

   (a) Show that conjunctions (like $x_1 \land \overline{x}_2 \land x_3$) and disjunctions (like $x_1 \lor \overline{x}_2 \lor x_3$) are special cases of decisions lists.

   (b) Show that decisions lists are a special case of linear threshold functions. That is, any function that can be expressed as a decision list can also be expressed as a linear threshold function “$f(x) = +$ iff $w_1 x_1 + \ldots + w_n x_n \geq w_0$”, for some values $w_0, w_1, \ldots, w_n$.

3. VC-dimension of linear separators: In the problems below you will prove that the VC-dimension of the class $H_n$ of halfspaces (another term for linear threshold functions) in $n$ dimensions is $n + 1$. ($H_n$ is the set of functions $w_1 x_1 + \ldots + w_n x_n \geq a_0$, where $w_0, \ldots, w_n$ are...
real-valued.) We will use the following definition: The convex hull of a set of points \( S \) is the set of all convex combinations of points in \( S \); this is the set of all points that can be written as \( \sum_{x_i \in S} \lambda_i x_i \), where each \( \lambda_i \geq 0 \), and \( \sum \lambda_i = 1 \). It is not hard to see that if a halfspace has all points from a set \( S \) on one side, then the entire convex hull of \( S \) must be on that side as well.

(a) [definition] Define the VC dimension and describe the importance and usefulness of VC dimension in machine learning.

(b) [lower bound] Prove that VC-dim(\( H_n \)) \( \geq n + 1 \) by presenting a set of \( n + 1 \) points in \( n \)-dimensional space such that one can partition that set with halfspaces in all possible ways. (And, show how one can partition the set in any desired way.)

(c) [upper bound part 1] The following is “Radon’s Theorem,” from the 1920’s.

**Theorem.** Let \( S \) be a set of \( n + 2 \) points in \( n \) dimensions. Then \( S \) can be partitioned into two (disjoint) subsets \( S_1 \) and \( S_2 \) whose convex hulls intersect.

Show that Radon’s Theorem implies that the VC-dimension of halfspaces is at most \( n + 1 \). Conclude that VC-dim(\( H_n \)) = \( n + 1 \).

(d) [upper bound part 2] Now we prove Radon’s Theorem. We will need the following standard fact from linear algebra. If \( x_1, \ldots, x_{n+1} \) are \( n + 1 \) points in \( n \)-dimensional space, then they are linearly dependent. That is, there exist real values \( \lambda_1, \ldots, \lambda_{n+1} \) not all zero such that \( \lambda_1 x_1 + \ldots + \lambda_{n+1} x_{n+1} = 0 \).

You may now prove Radon’s Theorem however you wish. However, as a suggested first step, prove the following. For any set of \( n+2 \) points \( x_1, \ldots, x_{n+2} \) in \( n \)-dimensional space, there exist \( \lambda_1, \ldots, \lambda_{n+2} \) not all zero such that \( \sum \lambda_i x_i = 0 \) and \( \sum \lambda_i = 0 \). (This is called affine dependence.) Now, think about the lambdas...