

CS 229, Autumn 2009

Practice Midterm Solutions

Notes:

1. The midterm will have about 5-6 long questions, and about 8-10 short questions. Space will be provided on the actual midterm for you to write your answers.
2. The midterm is meant to be educational, and as such some questions could be quite challenging. Use your time wisely to answer as much as you can!
3. For additional practice, please see CS 229 extra problem sets available at

<http://see.stanford.edu/see/materials/aimlcs229/assignments.aspx>

1. [13 points] Generalized Linear Models

Recall that generalized linear models assume that the response variable y (conditioned on x) is distributed according to a member of the exponential family:

$$p(y; \eta) = b(y) \exp(\eta T(y) - a(\eta)),$$

where $\eta = \theta^T x$. For this problem, we will assume $\eta \in \mathbb{R}$.

- (a) [10 points] Given a training set $\{(x^{(i)}, y^{(i)})\}_{i=1}^m$, the loglikelihood is given by

$$\ell(\theta) = \sum_{i=1}^m \log p(y^{(i)} | x^{(i)}; \theta).$$

Give a set of conditions on $b(y)$, $T(y)$, and $a(\eta)$ which ensure that the loglikelihood is a concave function of θ (and thus has a unique maximum). Your conditions must be reasonable, and should be as weak as possible. (E.g., the answer “any $b(y)$, $T(y)$, and $a(\eta)$ so that $\ell(\theta)$ is concave” is not reasonable. Similarly, overly narrow conditions, including ones that apply only to specific GLMs, are also not reasonable.)

Answer: The log-likelihood is given by

$$\ell(\theta) = \sum_{k=1}^M \log(b(y)) + \eta^{(k)} T(y) - a(\eta^{(k)})$$

where $\eta^{(k)} = \theta^T x^{(k)}$. Find the Hessian by taking the partials with respect to θ_i and θ_j ,

$$\frac{\partial}{\partial \theta_i} \ell(\theta) = \sum_{k=1}^M T(y) x_i^{(k)} - \frac{\partial}{\partial \eta} a(\eta^{(k)}) x_i^{(k)}$$

$$\frac{\partial^2}{\partial \theta_i \partial \theta_j} \ell(\theta) = \sum_{k=1}^M - \frac{\partial^2}{\partial \eta^2} a(\eta^{(k)}) x_i^{(k)} x_j^{(k)}$$

$$\begin{aligned}
 &= H_{i,j} \\
 H &= - \sum_{k=1}^M \frac{\partial^2}{\partial \eta^2} a(\eta^{(k)}) x^{(k)} x^{(k)T} \\
 z^T H z &= - \sum_{k=1}^M \frac{\partial^2}{\partial \eta^2} a(\eta^{(k)}) (z^T x^{(k)})^2
 \end{aligned}$$

If $\frac{\partial^2}{\partial \eta^2} a(\eta) \geq 0$ for all η , then $z^T H z \leq 0$. If H is negative semi-definite, then the original optimization problem is concave.

- (b) **[3 points]** When the response variable is distributed according to a Normal distribution (with unit variance), we have $b(y) = \frac{1}{\sqrt{2\pi}} e^{-\frac{y^2}{2}}$, $T(y) = y$, and $a(\eta) = \frac{\eta^2}{2}$. Verify that the condition(s) you gave in part (a) hold for this setting.

Answer:

$$\frac{\partial^2}{\partial \eta^2} a(\eta) = 1 \geq 0.$$

2. [15 points] Bayesian linear regression

Consider Bayesian linear regression using a Gaussian prior on the parameters $\theta \in \mathbb{R}^{n+1}$. Thus, in our prior, $\theta \sim \mathcal{N}(\vec{0}, \tau^2 I_{n+1})$, where $\tau^2 \in \mathbb{R}$, and I_{n+1} is the $n+1$ -by- $n+1$ identity matrix. Also let the conditional distribution of $y^{(i)}$ given $x^{(i)}$ and θ be $\mathcal{N}(\theta^T x^{(i)}, \sigma^2)$, as in our usual linear least-squares model.¹ Let a set of m IID training examples be given (with $x^{(i)} \in \mathbb{R}^{n+1}$). Recall that the MAP estimate of the parameters θ is given by:

$$\theta_{MAP} = \arg \max_{\theta} \left(\prod_{i=1}^m p(y^{(i)} | x^{(i)}, \theta) \right) p(\theta)$$

Find, in closed form, the MAP estimate of the parameters θ . For this problem, you should treat τ^2 and σ^2 as fixed, known, constants. [Hint: Your solution should involve deriving something that looks a bit like the Normal equations.]

Answer:

$$\begin{aligned}
 \theta_{MAP} &= \arg \max_{\theta} \left(\prod_{i=1}^m p(y^{(i)} | x^{(i)}, \theta) \right) p(\theta) \\
 &= \arg \max_{\theta} \log \left[\left(\prod_{i=1}^m p(y^{(i)} | x^{(i)}, \theta) \right) p(\theta) \right] \\
 &= \arg \min_{\theta} \left(-\log p(\theta) - \sum_{i=1}^m \log p(y^{(i)} | x^{(i)}, \theta) \right)
 \end{aligned} \tag{1}$$

Substituting expressions for $p(\theta)$ and $p(y^{(i)} | x^{(i)}, \theta)$, and dropping terms that don't affect the optimization, we get:

$$\theta_{MAP} = \arg \min_{\theta} \left(\frac{\sigma^2}{\tau^2} \theta^T \theta + (Y - X\theta)^T (Y - X\theta) \right) \tag{2}$$

¹Equivalently, $y^{(i)} = \theta^T x^{(i)} + \varepsilon^{(i)}$, where the $\varepsilon^{(i)}$'s are distributed IID $\mathcal{N}(0, \sigma^2)$.

In the above expression, Y is an m -vector containing the training labels $y^{(i)}$, X is an m -by- n matrix with the data $x^{(i)}$, and θ is our vector of parameters. Taking derivatives with respect to θ , equating to zero, and solving, we get:

$$\theta_{MAP} = (X^T X + \frac{\sigma^2}{\tau^2} I_n)^{-1} X^T Y \quad (3)$$

Observe the similarity between this expression, and the least squares solution derived in the notes.

3. [18 points] Kernels

In this problem, you will prove that certain functions K give valid kernels. Be careful to justify every step in your proofs. Specifically, if you use a result proved either in the lecture notes or homeworks, be careful to state exactly which result you're using.

- (a) [8 points] Let $K(x, z)$ be a valid (Mercer) kernel over $\mathbb{R}^n \times \mathbb{R}^n$. Consider the function given by

$$K_e(x, z) = \exp(K(x, z)).$$

Show that K_e is a valid kernel. [Hint: There are many ways of proving this result, but you might find the following two facts useful: (i) The Taylor expansion of e^x is given by $e^x = \sum_{j=0}^{\infty} \frac{1}{j!} x^j$ (ii) If a sequence of non-negative numbers $a_i \geq 0$ has a limit $a = \lim_{i \rightarrow \infty} a_i$, then $a \geq 0$.]

Answer: Let $K_i(x, z) = \sum_{j=0}^i \frac{1}{j!} K(x, z)^j$. K_i is a polynomial in $K(x, z)$ with positive coefficients. As proved in the homework, $K_i(x, z)$ is also a kernel, so $z^T K_i z \geq 0$. Thus,

$$\lim_{i \rightarrow \infty} z^T K_i z \geq 0$$

$$z^T (\lim_{i \rightarrow \infty} K_i) z \geq 0$$

Since $\lim_{i \rightarrow \infty} K_i = K_e$, K_e is positive semi-definite, and thus a valid kernel.

- (b) [8 points] The Gaussian kernel is given by the function

$$K(x, z) = e^{-\frac{\|x-z\|^2}{\sigma^2}},$$

where $\sigma^2 > 0$ is some fixed, positive constant. We said in class that this is a valid kernel, but did not prove it. Prove that the Gaussian kernel is indeed a valid kernel. [Hint: The following fact may be useful. $\|x - z\|^2 = \|x\|^2 - 2x^T z + \|z\|^2$.]

Answer:

We can rewrite the Gaussian kernel as

$$K(x, z) = e^{-\frac{\|x\|^2}{\sigma^2}} e^{-\frac{\|z\|^2}{\sigma^2}} e^{\frac{2}{\sigma^2} x^T z}$$

The first two terms together form a kernel by the fact proved in the homework that $K(x, z) = f(x)f(z)$ is a valid kernel. The third term is $e^{K(x, z)}$, which we've already shown to be a valid kernel. By the result proved in the homework, the product of two kernels is also a kernel.

4. [18 points] One-class SVM

Given an unlabeled set of examples $\{x^{(1)}, \dots, x^{(m)}\}$ the *one-class SVM algorithm* tries to find a direction w that maximally separates the data from the origin. **Answer:**²

More precisely, it solves the (primal) optimization problem:

$$\begin{aligned} \min_w \quad & \frac{1}{2} w^\top w \\ \text{s.t.} \quad & w^\top x^{(i)} \geq 1 \quad \text{for all } i = 1, \dots, m \end{aligned}$$

A new test example x is labeled 1 if $w^\top x \geq 1$, and 0 otherwise.

- (a) [9 points] The primal optimization problem for the one-class SVM was given above. Write down the corresponding dual optimization problem. Simplify your answer as much as possible. In particular, w should not appear in your answer. **Answer:** The Lagrangian is given by

$$L(w, \alpha) = \frac{1}{2} w^\top w + \sum_{i=1}^m \alpha_i (1 - w^\top x^{(i)}). \quad (4)$$

Setting the gradient of the Lagrangian with respect to w to zero, we obtain $w = \sum_{i=1}^m \alpha_i x^{(i)}$. It follows that

$$\max_{\alpha \geq 0} \min_w \left(\frac{1}{2} w^\top w + \sum_{i=1}^m \alpha_i (1 - w^\top x^{(i)}) \right) \quad (5)$$

$$= \max_{\alpha \geq 0} \frac{1}{2} \left(\sum_{i=1}^m \alpha_i x^{(i)} \right)^\top \left(\sum_{i=1}^m \alpha_i x^{(i)} \right) + \sum_{i=1}^m \alpha_i \left(1 - \left(\sum_{i=1}^m \alpha_i x^{(i)} \right)^\top x^{(i)} \right) \quad (6)$$

$$= \max_{\alpha \geq 0} \left(\sum_{i=1}^m \alpha_i - \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m \alpha_i \alpha_j x^{(i)\top} x^{(j)} \right) \quad (7)$$

The first equality follows from setting the gradient w.r.t. w equal to zero, and solving for w , which gives $w = \sum_{i=1}^m \alpha_i x^{(i)}$ and substituting this expression for w . The second equality follows from simplifying the expression.

- (b) [4 points] Can the one-class SVM be kernelized (both in training and testing)? Justify your answer.

Answer: Yes. For training we can use the dual formulation, in which only inner products of the data appear. For testing at a point z we just need to evaluate $w^\top z = \left(\sum_{i=1}^m \alpha_i x^{(i)} \right)^\top z = \sum_{i=1}^m \alpha_i x^{(i)\top} z$ in which the training data and the test point z only appear in inner products.

²This turns out to be useful for anomaly detection. In anomaly detection you are given a set of data points that are all considered to be 'normal'. Given these 'normal' data points, the task is to decide for a new data point whether it is also 'normal' or not. Adding slack variables allows for training data that are not necessarily all 'normal'. The most common formulation with slack variables is not the most direct adaptation of the soft margin SVM formulation seen in class. Instead the ν -SVM formulation is often used. This formulation allows you to specify the fraction of outliers (instead of the constant C which is harder to interpret). See the literature for details.

- (c) [5 points] Give an SMO-like algorithm to optimize the dual. I.e., give an algorithm that in every optimization step optimizes over the smallest possible subset of variables. Also give in closed-form the update equation for this subset of variables. You should also justify why it is sufficient to consider this many variables at a time in each step.

Answer: Since we have convex optimization problem with only independent coordinate wise constraints ($\alpha_i \geq 0$), we can optimize iteratively over 1 variable at a time. Optimizing w.r.t. α_i is done by setting

$$\alpha_i = \max \left\{ 0, \frac{1}{K_{i,i}} \left(1 - \sum_{j \neq i} \alpha_j K_{i,j} \right) \right\}$$

(Set the derivative w.r.t. α_i equal to zero and solve for α_i . And take into account the constraint. Here, we defined $K_{i,j} = x^{(i)\top} x^{(j)}$.)

5. [18 points] Uniform Convergence

In this problem, we consider trying to estimate the mean of a biased coin toss. We will repeatedly toss the coin and keep a running estimate of the mean. We would like to prove that (with high probability), after some initial set of N tosses, the running estimate from that point on will *always* be accurate and never deviate too much from the true value.

More formally, let $X_i \sim \text{Bernoulli}(\phi)$ be IID random variables. Let $\hat{\phi}_n$ be our estimate for ϕ after n observations:

$$\hat{\phi}_n = \frac{1}{n} \sum_{i=1}^n X_i.$$

We'd like to show that after a certain number of coin flips, our estimates will stay close to the true value of ϕ . More formally, we'd like to show that for all $\gamma, \delta \in (0, 1/2]$, there exists a value N such that

$$P \left(\max_{n \geq N} |\phi - \hat{\phi}_n| > \gamma \right) \leq \delta.$$

Show that in order to make the guarantee above, it suffices to have $N = O\left(\frac{1}{\gamma^2} \log\left(\frac{1}{\delta\gamma}\right)\right)$. You may need to use the fact that for $\gamma \in (0, 1/2]$, $\log\left(\frac{1}{1 - \exp(-2\gamma^2)}\right) = O\left(\log\left(\frac{1}{\gamma}\right)\right)$.

[Hint: Let A_n be the event that $|\phi - \hat{\phi}_n| > \gamma$ and consider taking a union bound over the set of events $A_n, A_{n+1}, A_{n+2}, \dots$]

Answer:

$$\begin{aligned} \Pr\left(\max_{n \geq N} |\theta - \hat{\theta}_n| > \gamma\right) &= \Pr\left(\bigcup_{n \geq N} \{|\theta - \hat{\theta}_n| > \gamma\}\right) \\ &\leq \sum_{n \geq N} \Pr(|\theta - \hat{\theta}_n| > \gamma) \\ &\leq \sum_{n \geq N} 2e^{-2\gamma^2 n} \\ &= \frac{2(e^{-2\gamma^2})^N}{1 - e^{-2\gamma^2}} \end{aligned}$$

Hence, in order to guarantee that $\Pr(\max_{n \geq N} |\theta - \hat{\theta}_n| > \gamma) \leq \delta$, we only need to choose N such that

$$\begin{aligned} \frac{2(e^{-2\gamma^2})^N}{1 - e^{-2\gamma^2}} &\leq \delta \\ (e^{-2\gamma^2})^N &\leq \delta(1 - e^{-2\gamma^2})/2 \\ \log((e^{-2\gamma^2})^N) &\leq \log(\delta(1 - e^{-2\gamma^2})/2) \\ (-2\gamma^2)N &\leq \log \delta + \log(1 - e^{-2\gamma^2}) - \log 2 \\ N &\geq \frac{1}{2\gamma^2}(-\log \delta - \log(1 - e^{-2\gamma^2}) + \log 2) \\ N &\geq \frac{1}{2\gamma^2}(\log \frac{1}{\delta} + \log(\frac{1}{1 - e^{-2\gamma^2}}) + \log 2) \end{aligned}$$

Thus, it is sufficient to have

$$\begin{aligned} N &= O\left(\frac{1}{\gamma^2}(\log \frac{1}{\delta} + \log(\frac{1}{1 - e^{-2\gamma^2}}))\right) \\ &= O\left(\frac{1}{\gamma^2}(\log \frac{1}{\delta} + \log \frac{1}{\gamma})\right) \\ &= O\left(\frac{1}{\gamma^2} \log \frac{1}{\delta\gamma}\right) \end{aligned}$$

6. [40 points] Short Answers

The following questions require a true/false accompanied by one sentence of explanation, or a reasonably short answer (usually at most 1-2 sentences or a figure).

To discourage random guessing, one point will be deducted for a wrong answer on multiple choice questions! Also, no credit will be given for answers without a correct explanation.

- (a) [5 points] Let there be a binary classification problem with continuous-valued features. In Problem Set #1, you showed if we apply Gaussian discriminant analysis using the same covariance matrix Σ for both classes, then the resulting decision boundary will be linear. What will the decision boundary look like if we modeled the two classes using separate covariance matrices Σ_0 and Σ_1 ? (I.e., $x^{(i)}|y^{(i)} = b \sim \mathcal{N}(\mu_b, \Sigma_b)$, for $b = 0$ or 1 .)

Answer: The decision boundary is given by the following equation (then using Bayes rule and replacing terms independent of x by constants C_1, C_2 on the next line)

$$\begin{aligned} \log p(y = 0|x) &= \log p(y = 1|x) \\ \log p(x|y = 0) + \log p(y = 0) - \log p(x) &= \log p(x|y = 1) + \log p(y = 1) - \log p(x) \\ C_1 - \frac{1}{2}(x - \mu_0)^\top \Sigma_0^{-1}(x - \mu_0) &= C_2 - \frac{1}{2}(x - \mu_1)^\top \Sigma_1^{-1}(x - \mu_1) \end{aligned}$$

This is a quadratic decision boundary in x . (In particular, if we consolidate all terms on the left side, the quadratic term in x does not cancel.)

- (b) [5 points] Consider a sequence of examples $(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \dots, (x^{(m)}, y^{(m)})$. Assume that for all i we have $\|x^{(i)}\| \leq D$ and that the data are linearly separated with a margin γ . Suppose that the perceptron algorithm makes exactly $(D/\gamma)^2$ mistakes on this sequence of examples. Now, suppose we use a feature mapping $\phi(\cdot)$ to a higher dimensional space and use the corresponding kernel perceptron algorithm on the same sequence of data (now in the higher-dimensional feature space). Then the kernel perceptron (implicitly operating in this higher dimensional feature space) will make a number of mistakes that is
- strictly less than $(D/\gamma)^2$.
 - equal to $(D/\gamma)^2$.
 - strictly more than $(D/\gamma)^2$.
 - impossible to say from the given information.

Answer: Impossible to say from the given information, since the number of mistakes depends on the configuration of the points in the higher dimensional feature space, about which no information is given.

- (c) [5 points] Let any $x^{(1)}, x^{(2)}, x^{(3)} \in \mathbb{R}^p$ be given ($x^{(1)} \neq x^{(2)}, x^{(1)} \neq x^{(3)}, x^{(2)} \neq x^{(3)}$). Also let any $z^{(1)}, z^{(2)}, z^{(3)} \in \mathbb{R}^q$ be fixed. Then there exists a valid Mercer kernel $K : \mathbb{R}^p \times \mathbb{R}^p \rightarrow \mathbb{R}$ such that for all $i, j \in \{1, 2, 3\}$ we have $K(x^{(i)}, x^{(j)}) = (z^{(i)})^\top z^{(j)}$. True or False?

Answer: True. Consider any feature mapping that satisfies $\phi(x^{(i)}) = z^{(i)}$ for $i \in \{1, 2, 3\}$. [E.g. extend $\phi(\cdot)$ to be identically $\vec{0}$ for any argument different from $x^{(1)}, x^{(2)}, x^{(3)}$.] The kernel $K(\cdot, \cdot)$ that satisfies for all u, v $K(u, v) = \phi(u)^\top \phi(v)$ is a Mercer kernel and satisfies the desired properties.

- (d) [5 points] Let $f : \mathbb{R}^n \mapsto \mathbb{R}$ be defined according to $f(x) = \frac{1}{2}x^\top Ax + b^\top x + c$, where A is symmetric positive definite. Suppose we use Newton's method to minimize f . Show that Newton's method will find the optimum in exactly one iteration. You may assume that Newton's method is initialized with $\vec{0}$.

Answer: Using the formulas from the lecture notes, we have that

$$\nabla_x f(x) = Ax + b \quad (8)$$

$$\nabla_x^2 f(x) = A \quad (9)$$

Setting the gradient to zero, we find that the optimum of the function is given by

$$x = -A^{-1}b$$

Newton's method will perform the update

$$x = x - \nabla_x^2 f(x)^{-1} \nabla_x f(x) \quad (10)$$

$$= 0 - A^{-1}(A0 + b) \quad (11)$$

$$= -A^{-1}b \quad (12)$$

So it will move to the optimum on the first iteration.

- (e) [5 points] Consider binary classification, and let the input domain be $\mathcal{X} = \{0, 1\}^n$, i.e., the space of all n -dimensional bit vectors. Thus, each sample x has n binary-valued features. Let \mathcal{H}_n be the class of all boolean functions over the input space. What is $|\mathcal{H}_n|$ and $VC(\mathcal{H}_n)$?

Answer: $|\mathcal{H}_n| = 2^{2^n}$. There are exactly 2^n distinct points in \mathcal{X} , and \mathcal{H} can realize all 2^{2^n} labellings on those 2^n points; thus $VC(\mathcal{H}_n) \geq 2^n$. Since there can be no more than 2^n distinct points in the input space, $VC(\mathcal{H}_n) \leq 2^n$, and thus $VC(\mathcal{H}_n) = 2^n$.

- (f) [5 points] Suppose an ℓ_1 -regularized SVM (with regularization parameter $C > 0$) is trained on a dataset that is linearly separable. Because the data is linearly separable, to minimize the primal objective, the SVM algorithm will set all the slack variables to zero. Thus, the weight vector w obtained will be the same no matter what regularization parameter C is used (so long as it is strictly bigger than zero). True or false?

Answer: No - outliers can still affect the choice of separating line. We may choose to misclassify a point if it makes the margin larger. The value of C will affect how we choose to make this tradeoff.

- (g) [5 points] Consider using hold-out cross validation (using 70% of the data for training, 30% for hold-out CV) to select the bandwidth parameter τ for locally weighted linear regression. As the number of training examples m increases, would you expect the value of τ selected by the algorithm to generally become larger, smaller, or neither of the above? For this problem, assume that (the expected value of) y is a non-linear function of x .

Answer: Smaller. For any fixed τ , as the amount of training data increases, the prediction of locally weighted regression for a given x will converge to the prediction of the line that best fits (the expected value of) y in the τ -sized region around x . Since (the expected value of) y is a non-linear function of x , the smaller τ , the better the performance for sufficiently large training data set.

- (h) [5 points] Consider a feature selection problem in which the mutual information $MI(x_i, y) = 0$ for all features x_i . Also for every subset of features $S_i = \{x_{i_1}, \dots, x_{i_k}\}$ of size $< n/2$ we have $MI(S_i, y) = 0$.³ However there is a subset S^* of size exactly $n/2$ such that $MI(S^*, y) = 1$. I.e. this subset of features allows us to predict y correctly. Of the three feature selection algorithms listed below, which one do you expect to work best on this dataset?
- i. Forward Search.
 - ii. Backward Search.
 - iii. Filtering using mutual information $MI(x_i, y)$.
 - iv. All three are expected to perform reasonably well.

Answer: Backward Search, since it will be able to maintain a subset of features that predicts y correctly down to only $n/2$ features. Forward Search has no way to distinguish between any subset of $< n/2$ features and will thus end up with arbitrary subsets of $< n/2$ features, which will be good subsets with very low probability; resulting in also having inferior subsets of features of size $\geq n/2$. Filtering using mutual information cannot distinguish between any of the features, and will thus pick random subsets of the features.

³ $MI(S_i, y) = \sum_{S_i} \sum_y P(S_i, y) \log(P(S_i, y)/P(S_i)P(y))$, where the first summation is over all possible values of the features in S_i .