

## Mock Midterm Exam - Solutions

### 1 Subgradient Descent [20pts]

Derive the (sub)gradient descent update rule for a one-parameter linear model using the Mean Absolute Error,

$$\mathcal{L}_{\text{MAE}}(\mathbf{X}, \mathbf{y}, w) = \frac{1}{N} \sum_{n=1}^N |wx_n - y_n|.$$

Hint: The function  $f(x) = |ax|$  is a composition of two simpler function. Use the chain rule!

*Solution:* Our cost function is  $\mathcal{L}(\mathbf{X}, \mathbf{y}, w) = \frac{1}{N} \sum \mathcal{L}_n(x_n, y_n, w)$  where  $\mathcal{L}_n(x_n, y_n, w) = |wx_n - y_n|$ . We have that

$$\frac{\partial \mathcal{L}(\mathbf{X}, \mathbf{y}, w)}{\partial w} = \frac{1}{N} \sum_{n=1}^N \frac{\partial \mathcal{L}_n(x_n, y_n, w)}{\partial w}.$$

Let  $a, e$  be two functions such that  $a(x) = |x|$  and  $e(x, y, w) = wx - y$ . We can rewrite  $\mathcal{L}_n$  as  $a \circ e$ . Let us find the derivative of  $\mathcal{L}_n$  using the chain rule.

$$\begin{aligned} \frac{\partial \mathcal{L}_n(x_n, y_n, w)}{\partial w} &= \frac{\partial a(e(x_n, y_n, w))}{\partial w} \\ &= \frac{\partial a(e(x_n, y_n, w))}{\partial e} \frac{\partial e(x_n, y_n, w)}{\partial w} \end{aligned}$$

We have that  $\frac{\partial e}{\partial w}(x_n, y_n, w) = x_n$ , but  $|x|$  is not differentiable at  $x = 0$ . We will have to use subgradients.

Remember that a vector  $\mathbf{g}$  is a subgradient of the function  $f$  in  $\mathbf{x}$  if  $f(\mathbf{y}) \geq f(\mathbf{x}) + \mathbf{g}^\top(\mathbf{y} - \mathbf{x})$ , for all  $\mathbf{y}$ . Note that in our one dimensional case, for a subgradient of  $|x|$  in  $x = 0$ , we need to find  $g$  such that

$$|y| \geq gy, \forall y.$$

Any  $g : |g| \leq 1$  will do, but since we want the error to go to 0 and stay here, we will use  $g = 0$ . We therefore have that

$$\frac{\partial a(x)}{\partial x} = \begin{cases} 1 & \text{if } x > 0 \\ 0 & \text{if } x = 0 \\ -1 & \text{if } x < 0 \end{cases} \text{ and } \frac{\partial e(x, y, w)}{\partial w} = x$$

Which gives us the following expression for the gradient

$$\frac{\partial \mathcal{L}(x, y, w)}{\partial w} = \frac{1}{N} \sum_{n=1}^N \begin{cases} x_n & \text{if } wx_n - y_n > 0 \\ 0 & \text{if } wx_n - y_n = 0 \\ -x_n & \text{if } wx_n - y_n < 0 \end{cases}$$

Therefore, one step of gradient descent with step size  $\gamma$  is given by  $w^{(i+1)} = w^{(i)} - \frac{\gamma}{N} \begin{cases} x & \text{if } wx - y > 0 \\ 0 & \text{if } wx - y = 0 \\ -x & \text{if } wx - y < 0 \end{cases}$

## 2 Multiple-Output Regression [20pts]

Let  $S = \{(\mathbf{y}_n, \mathbf{x}_n)\}_{n=1}^N$  be our training set for a regression problem with  $\mathbf{x}_n \in \mathbb{R}^D$  as usual. But now  $\mathbf{y}_n \in \mathbb{R}^K$ , i.e., we have  $K$  outputs for each input. We want to fit a linear model for each of the  $K$  outputs, i.e., we now have  $K$  regressors  $f_k(\cdot)$  of the form

$$f_k(\mathbf{x}) = \mathbf{x}^\top \mathbf{w}_k,$$

where each  $\mathbf{w}_k^\top = (w_{k1}, \dots, w_{kD})$  is the weight vector corresponding to the  $k$ -th regressor. Let  $\mathbf{W}$  be the  $D \times K$  matrix whose columns are the vectors  $\mathbf{w}_k$ .

Our goal is to minimize the following cost function  $\mathcal{L}$ :

$$\mathcal{L}(\mathbf{W}) = \sum_{k=1}^K \sum_{n=1}^N \frac{1}{2\sigma_k^2} (y_{nk} - \mathbf{x}_n^\top \mathbf{w}_k)^2 + \frac{1}{2} \sum_{k=1}^K \|\mathbf{w}_k\|_2^2,$$

where the  $\sigma_k$  are known real-valued scalars. Let  $\boldsymbol{\sigma} = (\sigma_1, \dots, \sigma_K)$ .

For the solution, let  $\mathbf{X}$  be the  $N \times D$  matrix whose rows are the feature vectors  $\mathbf{x}_n$ .

1. (4pts) Write down the normal equations for  $\mathbf{W}^*$ , the minimizer of the cost function. I.e., what is the first-order condition that  $\mathbf{W}^*$  has to fulfill in order to minimize  $\mathcal{L}(\mathbf{W})$ .

*Solution:* Note that the cost function  $\mathcal{L}(\mathbf{W})$  is the sum of  $K$  cost functions,  $\mathcal{L}(\mathbf{w}_k)$ , each of which only depends on its own parameter  $\mathbf{w}_k$ . So if we compute the gradient with respect to  $\mathbf{w}_k$  then this only involves the term  $\mathcal{L}(\mathbf{w}_k)$  and we get

$$\frac{1}{\sigma_k^2} \mathbf{X}^\top (\mathbf{X} \mathbf{w}_k - \mathbf{y}_k) + \mathbf{w}_k = 0.$$

This is essentially the expression we had for ridge regression.

2. (8pts) Is the minimum  $\mathbf{W}^*$  unique? Assuming it is, write down an expression for this unique solution.

*Solution:* We show that the problem is strictly convex and therefore has a unique minimizer. This follows from the fact that the first double sum term (the cost function) is convex in  $\mathbf{W}$ , and that the second term, the squared norm regularizer, is strictly convex in  $\mathbf{W}$ . Therefore the sum of both is strictly convex. We have the solution

$$\mathbf{w}_k^* = \left( \frac{1}{\sigma_k^2} \mathbf{X}^\top \mathbf{X} + \mathbf{I}_D \right)^{-1} \frac{1}{\sigma_k^2} \mathbf{X}^\top \mathbf{y}_k.$$

3. (8pts) Write down a probabilistic model, so that the MAP solution for this model coincides with minimizing the above cost function. Note that this will involve specifying the the likelihoods as well as a suitable prior (which will give you the regression term).

*Solution:* You are asked to derive a probabilistic model under which is the maximum a posteriori estimate. However, since "Posterior probability  $\propto$  Likelihood  $\times$  Prior probability", the question asks specifically for the prior and the likelihood only. Knowing that the maximization over a Gaussian is equivalent to minimizing the mean square error, one can check that  $\mathbf{w}_{\text{MAP}}^* = \arg \max_{\mathbf{w}} p(\mathbf{y} | \mathbf{X}, \mathbf{w}) p(\mathbf{w})$  is equivalent to the above cost minimization  $\mathbf{w}_{\text{normal}}^* = \arg \min_{\mathbf{w}} \mathcal{L}(\mathbf{w})$  if:

$$p(\mathbf{y} | \mathbf{X}, \mathbf{w}) = \prod_{n=1}^N \prod_{k=1}^K \mathcal{N}(y_{nk} | \mathbf{w}_k^\top \mathbf{x}_n, \sigma_k^2)$$

and

$$p(\mathbf{w}) = \prod_{k=1}^K \mathcal{N}(\mathbf{w}_k | \mathbf{0}, \mathbf{I}_D)$$

### 3 Proportional Hazard Model [20pts]

Let  $S = \{(y_n, \mathbf{x}_n)\}_{n=1}^N$  be our training set for a regression problem with  $\mathbf{x}_n \in \mathbb{R}^D$  as usual. We assume that the output  $y_n$  is *ordered*, i.e., takes values in the set  $\{1, 2, \dots, K\}$  where we think of these numbers as *ordered* by the natural ordering. We wish to fit a linear model.

In the *proportional hazard* model we use the following probability distribution,

$$p(y_n = k \mid \mathbf{x}_n, \mathbf{w}, \Theta) = \frac{e^{\eta_{nk}}}{\sum_{j=1}^K e^{\eta_{nj}}},$$

where  $\eta_{nk} = \Theta_k + \mathbf{x}_n^\top \mathbf{w}$ . The scalars  $\Theta_k$  are assumed to be ordered, i.e.,  $\Theta_1 > \Theta_2 > \dots > \Theta_K$ . Let  $\Theta = (\Theta_1, \dots, \Theta_K)$ .

1. (4pts) Show that  $p(y_n \mid \mathbf{x}_n, \mathbf{w}, \Theta)$  (and therefore also  $p(\mathbf{y} \mid \mathbf{X}, \mathbf{w}, \Theta)$ ) is a valid distribution.

Hint: What are the *two* conditions that you need to verify?

*Solution:* We need to verify that the expression is non-negative and sums up (as a function of  $k$ ) to 1. The first property is trivially true (the exponential function is always non-negative for real-valued arguments). The second one is true by construction (see denominator).

2. (8pts) Derive the log-likelihood for this model.

*Solution:* We proceed in our standard fashion. Let  $\tilde{\mathbf{y}}_n$  be a vector that is equal to the all-zero vector of length  $K$  except that  $\tilde{y}_{nk} = 1$  if  $y_n = k$ . Recall that all samples are assumed to be independent so that the joint distribution is equal to the product of the individual distributions. We get

$$\begin{aligned} \ln \prod_{n=1}^N \prod_{k=1}^K p(y_n = k \mid \mathbf{x}_n, \mathbf{w}, \Theta)^{\tilde{y}_{nk}} &= \ln \prod_{n=1}^N \prod_{k=1}^K \frac{e^{\tilde{y}_{nk} \eta_{nk}}}{(\sum_{j=1}^K e^{\eta_{nj}})^{\tilde{y}_{nk}}} \\ &= \sum_{n=1}^N \sum_{k=1}^K \tilde{y}_{nk} \eta_{nk} - \sum_{n=1}^N \ln \left[ \sum_{j=1}^K e^{\eta_{nj}} \right]. \end{aligned}$$

3. (8pts) Show that the negative of the log-likelihood is convex with respect to  $\Theta$  and  $\mathbf{w}$ .

*Solution:* We know that the sum of convex functions is convex. Therefore it suffices to show that each of the  $N$  terms

$$\sum_{k=1}^K \sum_{n=1}^N (-\tilde{y}_{nk} \eta_{nk}) + \sum_{n=1}^N \ln \left[ \sum_{j=1}^K e^{\eta_{nj}} \right].$$

is convex.

The term

$$-\sum_{k=1}^K \tilde{y}_{nk} \eta_{nk} = -\sum_{k=1}^K \tilde{y}_{nk} (\Theta_k + \mathbf{x}_n^\top \mathbf{w}_k)$$

is linear in the parameters and hence convex.

The second term is the composition of a linear function with the function  $\ln(e^{t_1} + \dots + e^{t_K})$  which we can assume to be convex. Hence the composed function is convex as well.

## 4 Multiple Choice Questions and Simple Problems [40pts]

Mark the correct **answer(s)**. More than one answer can be correct!

*Solution:* Correct solutions are marked in bold face.

- In regression, “complex” models tend to
  1. (1 pt) **overfit**
  2. (1 pt) have large bias
  3. (1 pt) **have large variance**
- In regression, “simple” models tend to
  1. (1 pt) overfit
  2. (1 pt) **have large bias**
  3. (1 pt) have large variance
- We add a regularization term because
  1. (1 pt) **this sometimes renders the minimization problem of the cost function into a strictly convex/concave problem**
  2. (1 pt) **this tends to avoid overfitting**
  3. (1 pt) this converts a regression problem into a classification problem
- The  $k$ -nearest neighbor classifier
  1. (1 pt) typically works the better the larger the dimension of the feature space
  2. (1 pt) can classify up to  $k$  classes
  3. (1 pt) **typically works the worse the larger the dimension of the feature space**
  4. (1 pt) can only be applied if the data can be linearly separated
  5. (1 pt) **has a misclassification rate of at most two times the one of the Bayes classifier if we have lots of data**
  6. (1 pt) has a misclassification rate that is two times better than the one of the Bayes classifier
- A real-valued scalar Gaussian distribution
  1. (1 pt) is a member of the exponential family with one scalar parameter
  2. (1 pt) **is a member of the exponential family with two scalar parameters**
  3. (1 pt) is not a member of the exponential family
- Which of the following statements is correct, where we assume that all the stated minima and maxima are in fact taken on in the domain of relevance. **All correct!**
  1. (1 pt)  $\max\{0, x\} = \max_{\alpha \in [0,1]} \alpha x$
  2. (1 pt)  $\min\{0, x\} = \min_{\alpha \in [0,1]} \alpha x$
  3. (1 pt) Let  $g(x) := \min_y f(x, y)$ . Then  $g(x) \leq f(x, y)$
  4. (1 pt)  $\max_x g(x) \leq \max_x f(x, y)$
  5. (1 pt)  $\max_x \min_y f(x, y) \leq \min_y \max_x f(x, y)$
- Which of the following statements are correct?
  1. **(1 pt)** The training error is typically smaller than the test error.
  2. **(1 pt)** The SVM (support vector machine) formulation we discussed can be optimized using SGD.
  3. (1 pt) One iteration of SGD for ridge regression costs roughly  $\Theta(ND)$ , where  $N$  is the number of samples and  $D$  is the dimension.
  4. **(1 pt)** Logistic regression as formulated in class can be optimized using SGD.

- The following functions are convex:

1. **(1 pt)**  $f(x) = x^2, x \in \mathbb{R}$
2. **(1 pt)**  $f(x) = x^3, x \in [-1, 1]$
3. **(1 pt)**  $f(x) = -x^3, x \in [-1, 0]$
4. **(1 pt)**  $f(x) = e^{-x}, x \in \mathbb{R}$
5. **(1 pt)**  $f(x) = e^{-x^2/2}, x \in \mathbb{R}$
6. **(1 pt)**  $f(x) = \ln(1/x), x \in (0, \infty)$
7. **(1 pt)**  $f(x) = g(h(x)), x \in \mathbb{R}$ , where  $g, h$  are convex and increasing over  $\mathbb{R}$

- **(5 pts)** Let  $f : \mathbb{R}^D \rightarrow \mathbb{R}$  be the function  $f(\mathbf{w}) := \exp(\mathbf{x}^\top \mathbf{w})$ , where  $\mathbf{x} \in \mathbb{R}^D$ . What is  $\nabla_{\mathbf{w}} f$ ?

$$\nabla_{\mathbf{w}} f(\mathbf{w}) = f(\mathbf{w}) \mathbf{x} \in \mathbb{R}^D$$