CS 274A Homework 4

Probabilistic Learning: Theory and Algorithms, CS 274A, Winter 2020

Due Date: Noon, Monday Feb 24th, submit via Gradescope

Instructions and Guidelines for Homeworks

- Please answer all of the questions and submit a scanned copy of your written solutions to Gradescope (either hand-written or typed are fine as long as the writing is legible).
- All problems are worth 10 points unless otherwise stated. All homeworks will get equal weight in computation of the final grade for the class.
- The homeworks are intended to help you work through the concepts we discuss in class in more detail. It is important that you try to solve the problems yourself. The homework problems are important to help you better learn and reinforce the material from class. If you don't do the homeworks you will likely have difficulty in the exams later in the quarter.
- If you can't solve a problem, you can discuss it *verbally* with another student. However, please note that before you submit your homework solutions you are not allowed to view (or show to any other student) any *written material* directly related to the homeworks, including other students' solutions or drafts of solutions, solutions from previous versions of this class, and so forth. The work you hand in should be your own original work.
- You are allowed to use reference materials in your solutions, such as class notes, textbooks, other reference material (e.g., from the Web), or solutions to other problems in the homework. It is strongly recommended that you first try to solve the problem yourself, without resorting to looking up solutions elsewhere. If you base your solution on material that we did not discuss in class, or is not in the class notes, then you need to clearly provide a reference, e.g., "based on material in Section 2.2 in ....."
- In problems that ask for a proof you should submit a complete mathematical proof (i.e., each line must follow logically from the preceding one, without "hand-waving"). Be as clear as possible in explaining your notation and in stating your reasoning as you go from line to line.
- If you wish to use LaTeX to write up your solutions you may find it useful to use the .tex file for this homework that is posted on the Web page. And please feel free to submit the .tex (as well as the .pdf) file for your solutions —it may be helpful to us when we send out solutions later on.

Suggested reading for Homework 4:

- Section 6.6 in Mathematics for Machine Learning about Conjugacy and the Exponential Family (for Problem 1)
- Chapter 9 (Sections 9.1, 9.2, and 9.3) in Mathematics for Machine Learning for the regression problems

Problem 1: Exponential Family

Assume you have a Poisson distribution for a random variable with parameter λ taking values $x \in \{1, 2, 3, \ldots, \}$. You also have an IID dataset $\{x_1, \ldots, x_N\}$ of samples. Use the results about the functional form of the exponential family to

1. Write the Poisson distribution in exponential family form and identify the sufficient statistic for the Poisson model.
2. Use this exponential family form to determine what the conjugate prior is for the Poisson model.

Problem 2: Maximum Likelihood Estimation for Least-Squares Regression

In class we showed that least squares regression could be derived from a conditional Gaussian likelihood. Assume we have training data in the form $D = \{(x_i, y_i)\}, i = 1, \ldots, N$, where x_i and y_i are both one-dimensional and real-valued. Say we assume that y given x is a conditional Gaussian density with mean $E[y|x] = ax + b$ and with variance σ^2. Assume that a, b, and σ^2 are unknown. Derive equations for the maximum likelihood estimates for each of a, b, and σ^2.

Problem 3: Normal Equations for Least Squares Estimation

Assume we have training data in the form $D = \{(\underline{x}_i, y_i)\}, i = 1, \ldots, N$, where the $\underline{x}$ are d-dimensional real-valued vectors (with one component set to the constant 1 to allow for an intercept term) and where y is a real-valued scalar. Assume we wish to fit a linear model of the form $\theta^T \underline{x}$ where θ is a d-dimensional parameter vector, where by "fit" we mean here that we want to minimize $MSE(\theta) = \frac{1}{N} \sum_{i=1}^{N} \left(y_i - \theta^T \underline{x}_i\right)^2$.

1. Prove that the solution to this problem can be written as the solution of a system of d linear equations (often referred to as the "normal equations") that can be written in the form $\mathbf{A}\theta = \mathbf{b}$ where θ has dimension $d \times 1$, $\mathbf{A}$ is a $d \times d$ matrix, and $\mathbf{b}$ is a $d \times 1$ vector. Starting from the definition of $MSE(\theta)$ above, carefully write out all steps in your proof, and clearly show how $\mathbf{A}$ and $\mathbf{b}$ are defined. If you need to assume as part of your solution that a particular matrix is full rank then assume so and state that you have assumed this.
2. Define the time complexity of minimizing $MSE(\theta)$ via the normal equations given a dataset $D = \{(\underline{x}_i, y_i)\}, i = 1, \ldots, N$.

Problem 4: Gradients for MAP Gaussian Regression

Consider a regression problem with data $D = \{(\underline{x}_i, y_i)\}, i = 1, \ldots, N$, where $\underline{x}$ is a d-dimensional real-valued vector (where one component is set to the constant 1 to allow for an intercept term). Consider a linear model in the form $f(\underline{x}; \theta) = \theta^T \underline{x}$ where θ is a d-dimensional parameter vector with one weight for each component of $\underline{x}$. Consider a Gaussian regression model of the form $y|\underline{x} \sim N(\theta^T \underline{x}, \sigma^2)$ where σ^2 is assumed known. Assume that we have independent priors on each weight of the form $\theta_j \sim N(\theta; 0, s^2)$ with prior mean 0 and where the prior variance s^2 is assumed known.

1. Define $\log P(\theta|D)$ for this problem
2. Derive the gradient ∇_θ with respect to the parameters θ for this problem.
3. Define the time complexity of computing one full gradient vector (on all N data points). Assuming the gradient descent algorithm takes M steps (i.e., M evaluations of the full gradient) compare the complexity of the gradient algorithm with the complexity of the Normal Equation method of the previous problem.

Problem 5: L1 or Lasso Regression

Consider a squared error loss function $MSE(\theta) = \sum_{i=1}^{N} (y_i - f(\underline{x}_i; \theta)^2$ with training data $D = \{(\underline{x}_i, y_i)\}, i = 1, \ldots, N$ and where f is some prediction model with unknown parameters $\theta = (\theta_1, \ldots, \theta_p)$. A popular regularization method takes the form $r(\theta) = \sum_{j=1}^{p} |\theta_j|$, resulting in an optimization problem where we minimize $MSE(\theta) + \lambda r(\theta)$, where λ is the relative weight of the regularization term (this is known as L1 or Lasso regularization).

Clearly show how we can interpret L1 regularization in terms of a prior on θ (by viewing this optimization problem from a Bayesian MAP perspective). Be sure to state clearly what distributional form this prior is, i.e., what name it has.

The next two problems are examples of generalized linear models, where we extend the concepts of regression to other types of noise and other types of y variables.

Problem 6: Poisson Regression

Consider a problem where we have a data set $D = \{(\underline{x}_i, y_i)\}, i = 1, \ldots, N$ where $\underline{x}_i$ are real-valued d-dimensional vectors and $y_i \in \{0, 1, 2, \ldots, \}$, i.e., the y_i's are non-negative integers, e.g., a count of the number of purchases an individual i makes on a Website given that they visit the site. In a Poisson regression model we build a model where the conditional distribution of y, $P(y|\underline{x}; \theta)$, is assumed to be a Poisson distribution with mean $E[y|\underline{x}] = \lambda(\underline{x}) = f(\underline{x}; \theta)$ where the mean varies as a function of $\underline{x}$, for some fixed value of parameters θ, rather than being having a fixed mean value λ. To ensure that $\lambda(\underline{x}) > 0$, a common parametrization is $\lambda(\underline{x}) = exp(\theta^T \underline{x})$, which is what we will use in this problem.

1. Derive the log-likelihood for this problem
2. Derive the gradient of the log-likelihood with respect to θ for this problem

Problem 7: Convexity for Logistic Classifiers

Consider a classification problem where we have training data $D = \{(\underline{x}_i, y_i)\}, 1 \ldots, i, \ldots, N$ where $\underline{x}_i$ are real-valued d-dimensional vectors and $y_i \in \{0, 1\}$ are binary class labels. We will assume for convenience that the first component of $\underline{x}$ always takes value 1, allowing us to have an intercept (or bias) term in our model. Let $f(\underline{x}; \theta)$ be a logistic regression model where

$$f(\underline{x}; \theta) \;=\; \frac{1}{1 + exp(-\theta^T \underline{x})}.$$

Let the empirical loss be the cross-entropy loss, defined as

$$CE(\theta) \;=\; -\frac{1}{N} \sum_{i=1}^{N} y_i \log f(x_i; \theta) \;+\; (1 - y_i) \log\big(1 - f(x_i; \theta)\big).$$

1. Derive the equation for the gradient for θ for this problem
2. Prove that $CE(\theta)$ is a convex function of θ and thus, that it has a single global minimum and no local maxima.