CAP 6610 Machine Learning, Spring 2022

Homework 1

Due 1/28/2022 11:59PM

Helpful reading: *Murphy* Chap. 2, 3.5, 4.1–4.3; *Bishop* Chap. 2, 3.1, 4.1–4.2.

1. (10 points) *Numerical check of the least squares solution.* Use your favorite language to generate a random 40×10 matrix $\boldsymbol{\Phi}$ and a random 40-vector $\boldsymbol{\psi}$. Compute the least squares solution $\boldsymbol{\theta}^\star = (\boldsymbol{\Phi}^\top \boldsymbol{\Phi})^{-1} \boldsymbol{\Phi}^\top \boldsymbol{\psi}$ and the associated loss $\|\boldsymbol{\Phi}\boldsymbol{\theta}^\star - \boldsymbol{\psi}\|^2$. (There may be several ways to do this, depending on the software package you use. In MATLAB or Julia, the command is simply `Phi\psi`.) Generate a random 10-vectors $\boldsymbol{\delta}$ and verify that $\|\boldsymbol{\Phi}(\boldsymbol{\theta}^\star + \boldsymbol{\delta}) - \boldsymbol{\psi}\|^2 > \|\boldsymbol{\Phi}\boldsymbol{\theta}^\star - \boldsymbol{\psi}\|^2$ holds. Repeat several times with different values of $\boldsymbol{\delta}$.

   Submit your code, including the code that checks whether the expected inequality that involves $\boldsymbol{\delta}$ holds.

2. (10 points) *Smokers.* According to the Center for Disease Control (CDC), "Compared to nonsmokers, men who smoke are about 23 times more likely to develop lung cancer and women who smoke are about 13 times more likely." The CDC also states that roughly 15% of all women are smokers, 18% of all adults are smokers. Assume that half the adults are women.

   (a) If you learn that a woman has been diagnosed with lung cancer, and you know nothing else about her, what is the probability she is a smoker?

   (b) What fraction of adult smokers in the USA are women?

   Hint. Let A be the event that "a woman smokes," and B be the event that "a woman gets lung cancer."

3. (10 points) Recall that the PMF of a Poisson random variable is

$$p(x; \lambda) = \frac{\lambda^x e^{-\lambda}}{x!},$$

   with one parameter λ. Given i.i.d. samples $x_1, ..., x_n \sim \text{Pois}(\lambda)$, derive the maximum likelihood estimate (MLE) for λ.

4. (10 points) The function `randn(d,1)` generates a multivariate normal variable $\boldsymbol{x} \in \mathbb{R}^d$ with zero mean and covariance $\boldsymbol{I}$. Describe how to generate a random variable from $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. *Hint.* If $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, then $\boldsymbol{A}\boldsymbol{x} + \boldsymbol{b} \sim \mathcal{N}(\boldsymbol{A}\boldsymbol{\mu} + \boldsymbol{b}, \boldsymbol{A}\boldsymbol{\Sigma}\boldsymbol{A}^\top)$.

5. (10 points) Consider a data set in which each data sample i is associated with a weighting factor $r_i > 0$, and we instead try to minimize the weighted MSE function

$$\underset{\boldsymbol{\theta}}{\text{minimize}} \quad \frac{1}{n} \sum_{i=1}^{n} r_i (y_i - \boldsymbol{\phi}_i^\top \boldsymbol{\theta})^2.$$

   Find an expression for the solution $\boldsymbol{\theta}^\star$ that minimizes this loss function.

6. (50 points) The 20 Newsgroups data set is a collection of approximately 20,000 newsgroup documents, partitioned (nearly) evenly across 20 different newsgroups. The data set can be downloaded here: `<http://qwone.com/~jason/20Newsgroups/>`. For simplicity, we will just focus on the "bag-of-words" representation of the documents given in the Matlab/Octave section. In this case, the input $\boldsymbol{x}_i$ is a vector of word histogram of doc i, and the output y_i is the news group that it belongs to.

   The data has been divided into a training set and test set. You will build a model from the training set, and evaluate its performance on the test set. The vocabulary size is more than 60,000, which is bigger than the number of documents. (Also, you will find that the `test_data` matrix has more rows than the `train_data` matrix, which means some words in the test data never appear in the training data.) To simplify computation, you will prune the data first by keeping only the words that have appeared more than 1,000 times in the training data (which is approximately 300) and keep only those rows in both the `train_data` matrix and the `test_data` matrix.

   (a) One effective classifer by Tom Mitchell is an instance of the naive Bayes model. First, the actual word count is ignored in the input data; we only consider whether a word j appears in doc i or not. Then each feature in $\boldsymbol{x}$ can be viewed as a Bernoulli random variable. Furthermore, the naive Bayes assumption states that each of these Bernoulli random variables are conditionally independent given the label y, i.e., $p(\boldsymbol{x}|y) = \prod p(x_j|y)$. Each of the $p(x_j|y)$ can be easily estimated using the training data.

   (b) To incorporate the word count, we can impose a rather different probabilistic model. Assume each doc is a huge multinomial random variable, with cardinality equal to the vocabulary size and the total number of draws is the length of that doc, given the label. In other words, $p(\boldsymbol{x}|y)$ is multinomial.

   (c) We can also assume each $p(\boldsymbol{x}|y)$ follows a multivariate normal distribution. Here we assume their covariance matrices are the same, which means the classifer is equivalent to the linear discriminant analysis. Of course, most people don't believe that bag-of-words actually follows a normal distribution, so some pre-processing is used. Do a Google search of TF-IDF and apply that to the data set before training your LDA model.

   (d) Build a least squares classifier using the TF-IDF representation of the data plus a constant 1 as the features. For more than 2 classes the least squares classifier will not be exactly the same as the multi-class LDA, but highly related.

   For each case, derive the mathematical expressions for the corresponding classifiers. Be specific about how to calculate each and every model parameter from data. Pick a language that you like and program these four classifiers using the training set, and report their prediction accuracy on the test set.

   Remark. When calculating the likelihoods for Naive Bayes-Bernoulli and multinomials, the expression $\prod_i p_i^{x_i}$ may give extremely small numbers that could cause underflow. We can overcome this issue by instead calculating $\sum_i x_i \log(p_i)$, which does not change which one is the maximum. However, beware of not taking $\log(0)$.

   Submit your code and make sure they are executable. We may or may not check your code.