CAP 6610 Machine Learning, Spring 2020

Homework 1

Due 1/27/2020 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) The uniform distribution for a continuous variable x is

$$p(x; a, b) = \frac{1}{b-a}, \qquad a \leq x \leq b.$$

   Verify that this distribution is normalized (integrats to one), and find expressions for its mean and variance.

2. (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 MLE for λ.

3. (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 $\boldsymbol{A}\boldsymbol{x} + \boldsymbol{b} \sim \mathcal{N}(\boldsymbol{A}\boldsymbol{\mu} + \boldsymbol{b}, \boldsymbol{A}\boldsymbol{\Sigma}\boldsymbol{A}^\top)$.

4. (20 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 $\widehat{\boldsymbol{\theta}}$ that minimizes this loss function.

5. (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.

   (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.

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 three classifiers using the training set, and report their prediction accuracy on the test set.

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