COMS 4771 Machine Learning *Fall 2023*

Homework 1

Due September 22, 2023 at 11:59 PM

Instructions for preparing and submitting your homework write-up are available here: `https://www.cs.columbia.edu/~djhsu/coms4771-f23/policies-homework.html`

Nearest neighbors

The MNIST dataset is available in the file `mnist.pkl`, and can be loaded as follows.

```
import pickle
mnist = pickle.load(open('mnist.pkl', 'rb'))
```

This loads the dataset as a Python dictionary with four keys: `'data'`, `'labels'`, `'testdata'`, and `'testlabels'`. The feature vectors for the training data are stored in `mnist['data']`, and the corresponding labels are stored in `mnist['labels']`. Test data are in `mnist['testdata']` and `mnist['testlabels']`. Numerical operations on the feature vectors are best performed after converting the feature values from unsigned 8-bit integers to floating point numbers: `mnist['data'].astype(float)` and `mnist['testdata'].astype(float)`.

You can view the images in the dataset using the following code.

```
import matplotlib.pyplot as plt
plt.imshow(mnist['data'][0].reshape(28,28), cmap='gray_r') # show first image
plt.show()
```

Problem 1. Recall that 28×28 pixel images from the MNIST dataset are treated as 784-dimensional vectors, where every component is an integer between 0 and 255.

(a) Suppose you find two images: one with all pixels at minimum intensity (0), and the other with all pixels at maximum intensity (255). What is the Euclidean distance between these two images?

(b) For every image x from the MNIST test dataset, let $\mathrm{NN}(x; \mathcal{S}_{\mathrm{MNIST}})$ denote its nearest neighbor (in Euclidean distance) in the MNIST training dataset $\mathcal{S}_{\mathrm{MNIST}}$. What is the average value of $\|x - \mathrm{NN}(x; \mathcal{S}_{\mathrm{MNIST}})\|^2$ over the test dataset? What is its square root?

(c) For each image x in both the MNIST training and test datasets, augment the image with 280 random pixels by sampling each pixel independently and uniformly at random from $\{0, 1, \ldots, 255\}$. So now each image can be regarded as a 28×38 pixel image (or 38×28 pixel image). What is the test error rate of the nearest neighbor classifier with these "noisy" images?

(d) What property of the training dataset determines the training error rate of the k-nearest neighbor classifier for $k = n$, where n is the number of training examples? The

"property" should be simple to describe and compute. Compute it on the MNIST training data and determine the training error rate.

For (b)–(d), please explain how you obtained your answers by giving (detailed) pseudocode or by displaying actual Python code alongside your answers.

Problem 2. Recall that functions $f_1 \colon \mathcal{X} \to \mathcal{Y}$ and $f_2 \colon \mathcal{X} \to \mathcal{Y}$ are equal if $f_1(x) = f_2(x)$ for all $x \in \mathcal{X}$. Let $\mathcal{S} = ((x^{(i)}, y^{(i)}))_{i=1}^n$ be a dataset from $\mathbb{R}^d \times \{0, 1\}$. In each of the following cases, determine whether the following pairs of functions (f_1, f_2) are equal.

(a) Function $f_1 \colon \mathbb{R}^d \to \{0, 1\}$ is the nearest neighbor classifier for $\mathcal{S}$ using distance function $D(x, z) = \|x - z\|_1$. Function $f_2 \colon \mathbb{R}^d \to \{0, 1\}$ is the nearest neighbor classifier for $\mathcal{S}$ using distance function $D(x, z) = \|x - z\|_4$.

(b) Function $f_1 \colon \mathbb{R}^d \to \{0, 1\}$ is the nearest neighbor classifier for $\mathcal{S}$ using distance function $D(x, z) = \|x - z\|$. Function $f_2 \colon \mathbb{R}^d \to \{0, 1\}$ is the nearest neighbor classifier for $\mathcal{S}$ using distance function $D(x, z) = \|x - z\|^2$.

(Assume ties are broken using a fixed rule, e.g., to favor the example with smaller index.)

(c) Suppose two vectors x and z from $\mathbb{R}^d$ satisfy $\|x - z\|_1 = 1$. What can you conclude about $\|x - z\|_3$? Some options: at least 1, at most 1, exactly equal to 1.

For each part, please briefly justify your answer.

Randomized response

Suppose you would like to estimate the proportion of people in a population with a positive COVID19 test result, i.e., the *positivity rate*. (For the purpose of this problem, assume everyone has been tested.) You randomly sample n individuals from the population. When sampling from a finite population, we typically use sampling without replacement. But to simplify the problem, assume that sampling *with* replacement is used. You could just ask each person whether or not they have tested positive. However, they may be uncomfortable sharing that information with you outright.

The *randomized response* method was developed to address this privacy concern. In the randomized response method, surveyed individuals are not asked to directly tell you whether or not they have tested positive. Instead, you ask each surveyed individual to do the following:

- Toss a fair coin four times (without letting you see the outcomes).
- If exactly two or three tosses come up heads:
  - Respond truthfully (i.e., say 1 if the individual has tested positive; 0 if not).
- Else:
  - Give the opposite response (i.e., say 0 if tested positive; 1 if not).

Because you do not observe the outcomes of the coin tosses, you never learn, with certainty, whether the surveyed individual has tested positive or not. This is a very strong privacy guarantee for the surveyed individuals called *differential privacy*. Moreover, the collected information can be used to obtain a good estimate of the proportion of individuals in the population with a positive test.

Problem 3. Regard the responses collected using the randomized response protocol described above as i.i.d. Bernoulli(θ) random variables $Y_1, \ldots, Y_n$, where the parameter θ is the positivity rate.

(a) What is the probability that $Y_1 = 1$? Give your answer in terms of the parameter θ.

(b) What is the log-likelihood of θ given data $y_1, \ldots, y_n \in \{0, 1\}$? Write your answer in terms of θ and $y_1, \ldots, y_n$.

(c) Suppose $n = 100$, and the number of y_i that are equal to 1 is 40 (i.e., $\sum_{i=1}^{n} y_i = 40$). Plot the log-likelihood as a function of $\theta \in [0, 1]$, and include this plot in your write-up. What appears to be the θ with highest likelihood?

No need to submit code for this problem.

Generative models

When we fit the normal generative model to the iris dataset using only on the "sepal length" feature, we obtained the following parameter estimates via MLE:

k	setosa (1)	versicolor (2)	virginica (3)
$\hat{\pi}_k$	1/3	1/3	1/3
$\hat{\mu}_k$	4.99	5.93	6.61
$\hat{\sigma}_k$	0.31	0.47	0.68

The classifier based on the distribution with these parameters is defined by

$$\hat{f}(x) = \operatorname*{arg\,max}_{k \in \{1,2,3\}} \hat{\pi}_k \cdot \frac{1}{\sqrt{2\pi\hat{\sigma}_k^2}} \exp\left(-\frac{(x - \hat{\mu}_k)^2}{2\hat{\sigma}_k^2}\right).$$

Problem 4. Give an explicit description of the set of x's for which $\hat{f}(x) = 3$. (For example, if it is an interval of the form $[a, b]$, write down this fact and give the numbers a and b.) Explain how you arrived at your answer.

Problem 5. Suppose that your intended application, distinguishing versicolor and virginica is not important, but it is worse to mistake either of them for setosa than it is to mistake setosa for versicolor or virginica. So, you come up with the following loss function:

	$y = \text{setosa}$	$y = \text{versicolor}$	$y = \text{virginica}$
$\hat{y} = \text{setosa}$	0	2	2
$\hat{y} = \text{versicolor}$	1	0	0
$\hat{y} = \text{virginica}$	1	0	0

Suppose $\hat{f} \colon \mathbb{R} \to \{\text{setosa}, \text{versicolor}, \text{virginica}\}$ is a classifier that minimizes the expected loss, assuming the distribution of labeled data is given by the normal generative model with the parameters from above. Give a description of the region where $\hat{f}(x) = \text{setosa}$, and explain how you arrived at your answer. (Write your description in terms of the parameters $(\hat{\pi}_k, \hat{\mu}_k, \hat{\sigma}_k)$; however, it need not be as simple/simplified as your answer to Problem 4.)

Training data

Problem 6. Consider the following potential applications of machine learning. In each case, does the training data seem suitable for the application? Base your answer on whether the data may be regarded as a representative sample from the population of interest. Write a brief but careful argument to justify your answer.

(a) A newspaper editor wants to build a predictor that predicts the number of "views" an article will get on the newspaper's website. The training data is a collection of articles that the newspaper has published over the past year, labeled by the number of views they received.

(b) A college administrator wants to build a classifier that predicts whether or not an incoming student for Fall 2023 will graduate within four years or not. The training data are the college applications of the college's incoming students from Fall 2018, labeled by whether or not they graduated within four years.

(c) A bank wants to build a classifier that predicts whether or not a loan applicant will repay a loan if granted one. The training data is a collection of loan applications that the bank has accepted over the past 20 years, labeled according whether or not the loan was repaid.