CSE 475: Statistical Methods in AI **Monsoon 2019**

Assignment 1: Understanding Data and Classifiers from First Principles

Lecturer: C. V. Jawahar *Date: DATE*

1.1 Objectives

The objective of this assignment is to enhance the understanding of the basic mathematical tools required for the machine learning.

- **Out:** Aug 9
- **In:** Aug 24 5pm

1.2 Experimental Setting

Dataset We use a subset of the extremely popular simple data set MNIST for this Assignment. This data set has many images of handwritten digits 0 to 9. i.e., 10 classes. Each image is of size 28×28. In our subset there are 7000 images in total from 10 classes, out of which 6000 are for training and 1000 are for testing. The train and test splits are provided along with the jupyter notebook template.

Since each sample is an image of size 28×28, we consider the sample as a vector of dimension $28^2 = 784$ or in R^{784}.

1.3 Questions

1.3.1 Representation

1. Plot the eigen value spectrum of the covariance matrix of
   (a) Samples corresponding to last digit of your roll number (say your-digit).
   (b) Samples corresponding to last digit of your friend's roll number (say another-digit).
   (c) All the training data
   (d) a randomly selected 50% of the training data.

   Normalize the eigen value spectrum for ease in plotting and really show only the first few (say 100) eigen values. What are the "approximate ranks" of these three covariance matrices.?

   Discuss: Is (a) and (b) different significantly? Why? Is (b) and (c) Different significantly? Why?

2. We know that each sample is in R^{784} with elements in $\{0, 1\}$. How many possible images could be there (pure combinatorics)? What percentage of these possibilities is accessible to us as MNIST data? If we had access to all these possible 28×28 binary images, how should have the eigen value spectrum of the covariance of the full data look like?

1.3.2 Linear Transformation

1. How does the eigen spectrum change if the original data was multiplied by an orthonormal matrix? Answer analytically and then also validate experimentally.
2. If samples were multiplied by 784×784 matrix of rank 1 or 2, (rank deficient matrices), how will eigen spectrum look like?
3. Project the data into the first and second eigen vectors and plot in 2D.

1.3.3 Probabilistic View

Let us fit a multivariate gaussian for each of the 10 classes in MNIST. Use

$$\mu = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i$$

$$\Sigma = \frac{1}{N}\sum_{i=1}^{N}[\mathbf{x}_i - \mu][\mathbf{x}_i - \mu]^T.$$

1. Classify each of the samples as

   $$\underset{i}{\operatorname{argmax}}\, P(x|\omega_i)$$

   using MAP and MLE

2. Consider two classes i and j and the corresponding covariances Σ_i. Let us simplify the setting by assuming that the covariances are same for both the classes

   $$\Sigma = \frac{\Sigma_1 + \Sigma_2}{2}$$

   Given that the covariances are equal, we can compute a linear decision boundary that separates class

i and class j. And classify each test sample as either class i and class j.

For MNIST, we need to do this for $10C_2$ times (for all pairs). A test sample is assigned to a class where it gets maximum votes. (majority voting).

3. Consider a simple linear classifier as the perpendicular bisector of line joining means for class i and j. Repeat the majority voting based classification as in the above question.

4. Now, we have 4 different ways of classifying (i) MAP (ii) MLE (iii) Bayesian pairwise (iv) Simple perpendicular bisectors.

   How do you compare the performances and what are your salient observations?

1.3.4 Nearest Neighbour based Tasks and Design

1. **NN Classification with various K** Implement a KNN classifier with K=1, 3, 7. Are the accuracies same? Why? Why not? How do we identify the best K? Suggest a computational procedure, with a logical explanation.

2. **Reverse NN based outlier detection:** A sample can be thought of as an outlier, if it is NOT in the in the nearest neighbour set of anybody else. Expand this idea into a computational algorithm. Now add some examples of images of the same size with english characters into this set. Automatically detect these outliers with your algorithm.

3. **NN for regression** Can we use NN based schemes for designing a solution to regression? Assume we had a “neatness score” for all the MNIST examples (instead of classID), how do we predict the neatness of a new sample.? (Assume neatness is a real number score between 0 and 1? Write the algorithm. Validate with toy samples on paper.

1.4 Submission and Instructions

Jupyter notebooks will be used for this assignment.

1. A Jupyter notebook with detailed instructions and function stubs will be shared.

2. You must write your code in the indicated code cells and analysis in the indicated text cells.

3. Do NOT attempt to modify any other cells.

4. Ensure that the notebook runs without errors if the code cells are run in sequence.

5. Rename the completed notebook to

   `<rollnumber>.ipynb`

   and upload only the notebook file to moodle.