Problem Set 4

CS 4375

Due: 11/23/2025 by 11:59pm

Note: all answers should be accompanied by explanations for full credit. Late homeworks will not be accepted.

Problem 1: PCA and Feature Selection (50pts)

In this problem, we will explore ways that we can use PCA for the problem of generating or selecting "good" features.

SVMs and PCA (25pts)

Consider the Gisette data set (attached to this problem set), which corresponds to a binary classification task (the data elements are the rows and the first column is the class label).

- Perform PCA on the training data to reduce the dimensionality of the data set (ignoring the class labels for the moment). What are the top six eigenvalues of the data covariance matrix?
- Build a set of values $K = \{k_{99}, k_{95}, k_{90}, k_{80}, k_{75}\}$, where k_z is equal to the smallest number of top k eigenvalues whose inclusion explains $z\%$ of the variance.
- For each $k \in K$, project the training data into the best k dimensional subspace (with respect to the Frobenius norm) and use the SVM with slack formulation under a Gaussian kernel to build a classifier. You should pick values of the slack penalty and variance in the Gaussian kernel using the validation data.
- What is the error of the best model the test data (report the k, c, and σ)? How does it compare to the best classifier without feature selection? Explain your observations.
- If you had to pick a value of k before evaluating the performance on the validation set (e.g., if this was not a supervised learning problem), how might you pick it?

PCA for Feature Selection (25pts)

If we performed PCA directly on the training data as we did in the first part of this question, we would generate new features that are linear combinations of our original features. If instead, we wanted to find a subset of our current features that were good for classification, we could still use PCA, but we would need to be more clever about it. The primary idea in this approach is to select features from the data that are good at explaining as much of the variance as possible. To do this, we can use the results of PCA as a guide. Implement the following algorithm for a given k and s:

1. Compute the top k eigenvalues and eigenvectors of the covariance matrix corresponding to the data matrix omitting the labels (recall that the columns of the data matrix are the input data points). Denote the top k eigenvectors as $v^{(1)}, \ldots, v^{(k)}$.

2. Define $\pi_j = \frac{1}{k} \sum_{i=1}^{k} {v_j^{(i)}}^2$.

3. Treating π as a probability distribution over the features, e.g., π_1 is the probability of picking feature 1, sample s features independently from π.

- Why does π define a probability distribution?

- Again, using the Gisette data set, for each $k \in \{k_{99}, k_{95}, k_{90}, k_{80}, k_{75}\}$ and each $s \in \{10, 20, \ldots, 100\}$, report the average test error of the SVM with slack classifier over 10 experiments. For each experiment use only the s selected features (note that there may be some duplicates, so only include each feature once).

- Does this provide a reasonable alternative to the SVM with slack formulation without feature selection on this data set? What are the pros and cons of this approach?

Problem 2: Exponential Maximum Likelihood Estimation (20pts)

Consider a nonnegative, real-valued random variable X that is distributed according to an exponential distribution $X \sim \lambda e^{-\lambda x}$ for some real-valued parameter $\lambda > 0$ and $x \geq 0$.

1. Given data samples $x^{(1)}, \ldots, x^{(m)}$, what is the maximum likelihood estimate for λ?

2. Give an example of a prior such that the maximum likelihood estimate of λ does not converge to the true value as the number of data observations goes to infinity.

3. Suppose we use the Gamma distribution as a prior probability distribution for λ: $\lambda \sim \frac{\beta^\alpha}{\Gamma(\alpha)} \lambda^{\alpha-1} e^{-\beta\lambda}$, where $\alpha, \beta > 0$ are parameters of the prior and $\Gamma(\alpha)$ is the normalizing constant. What is the MAP estimate under this choice of prior? Does the MAP estimator converge to the MLE as the number of data observations tends towards infinity?

Problem 3: k-Means++ (30pts)

For this problem, you will use the `leaf.data` file provided with this problem set. This data set was generated from the UCI Leaf Data Set (follow the link for information about the format of the data). The class labels are still in the data set and should be used for evaluation only (i.e., don't use them in the clustering procedure), but the specimen number has been removed. You should preprocess the data so that the non-label attributes have mean zero and variance one.

As discussed in class, random initializations of the k-means algorithm can easily get stuck in suboptimal clusterings. An improvement of the k-means algorithm, known as k-means++, instead chooses an initialization as follows:

- Choose a data point uniformly at random to be the first center.

- Repeat the following until k centers have been selected:

- For each data point x compute the distance between x and the nearest cluster center in the current set of centers. Denote this distance as d_x.
- Sample a training data point at random from the distribution p such that $p(x) \propto d_x^2$. Add the sampled point to the current set of centers.

1. For each value of $k \in \{10, 20, 30, 36, 40\}$, do the following.
   (a) Implement the k-means algorithm using k data points selected uniformly at random with replacement from the data. Run your method 100 times on the given data set and report the mean and standard deviation of the k-means objective for these runs.
   (b) Implement the k-means++ algorithm as described above. Run your k-means++ method 100 times on the given data set and report the mean and standard deviation of the k-means objective for these runs.
2. Using the labels from the data set, which approach do you think performed better on this data set?