MNIST Classification with Multivariate Gaussian Model

Submit a PDF of your answers to Canvas.

1. In a previous assignment, you computed the optimal decision boundary for the problem below: Consider a random vector $\boldsymbol{x}$ that comes from one of two classes:

$$\begin{aligned}\boldsymbol{x}|Y=0 &\sim \mathcal{N}\left(\begin{bmatrix}1\\0\end{bmatrix}, \begin{bmatrix}8 & 3\\3 & 2\end{bmatrix}\right)\\ \boldsymbol{x}|Y=1 &\sim \mathcal{N}\left(\begin{bmatrix}0\\2\end{bmatrix}, \begin{bmatrix}1 & 0.1\\0.1 & 1\end{bmatrix}\right)\end{aligned}$$

Previously we assumed a uniform prior: $p(y=0) = p(y=1) = 1/2$. In this problem, we will relax that assumption.

a) Last time we found a decision boundary for the MAP classifier had the form $\boldsymbol{x}^T\boldsymbol{B}\boldsymbol{x} + \boldsymbol{x}^T\boldsymbol{w} > c$. Is this form valid if the prior is non-uniform?

b) Find $\boldsymbol{B}$, $\boldsymbol{w}$ and c as expressions of the prior $p(y=0)$ and $p(y=1)$. *Hint: review previous solutions, as some terms may not depend on the prior.*

c) Create a receiver operating characteristic (ROC) curve by drawing 10,000 instances from each class, and classifying them using the decision boundary for a range of prior probabilities. The axis of the plot should be the estimates of $\mathbb{P}(\widehat{y}=1|y=0)$ and $\mathbb{P}(\widehat{y}=0|y=1)$. Note that changing the assumed prior creates the curve on the ROC curve.

SOLUTION:

a) For the binary case, the MAP classifier has the form

$$\frac{p(\boldsymbol{x}|y=0)p(y=0)}{p(\boldsymbol{x}|y=1)p(y=1)} \gtrless 1 \tag{1}$$

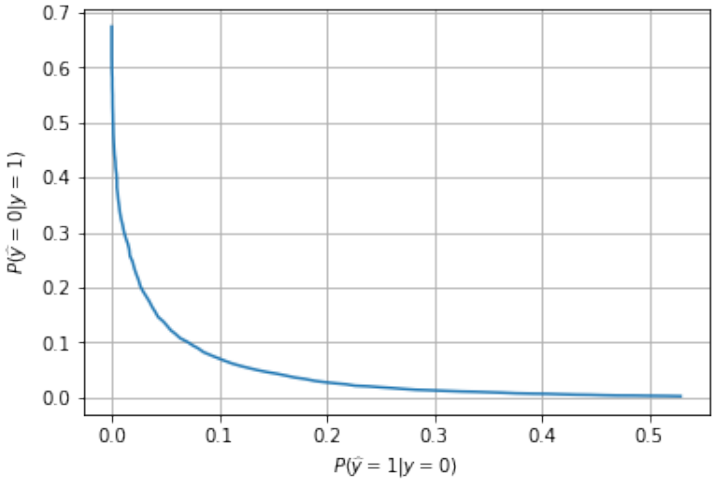
The quadratic form is still valid when $p(\boldsymbol{x}|y=0)$ and $p(\boldsymbol{x}|y=1)$ are Gaussian. We can see this by taking the log of both sides above.

b) Plugging in the expression for a multivariate normal in the likelihood ratio, and taking the log, we conclude that

$$\begin{aligned}c &= \log\left(|\boldsymbol{\Sigma}_0|\right) - \log\left(|\boldsymbol{\Sigma}_1|\right) + \boldsymbol{\mu}_0^T\boldsymbol{\Sigma}_0^{-1}\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1^T\boldsymbol{\Sigma}_1^{-1}\boldsymbol{\mu}_1 + \log p(y=0) - \log(p(y=1))\\ \boldsymbol{w} &= 2(\boldsymbol{\Sigma}_0^{-1}\boldsymbol{\mu}_0 - \boldsymbol{\Sigma}_1^{-1}\boldsymbol{\mu}_1)\\ \boldsymbol{B} &= \boldsymbol{\Sigma}_1^{-1} - \boldsymbol{\Sigma}_0^{-1}.\end{aligned}$$

Hence, $\boldsymbol{w}$ and $\boldsymbol{B}$ remain unchanged. Only the constant that defines the threshold changes.

c) The ROC curve is shown below:



Code is included in the associated notebook.

2. *Modeling MNIST as a Multivariate Gaussian.* Recall the maximum likelihood (ML) classifier:

$$\widehat{y} = \arg\min_{y} p(\boldsymbol{x}|y) \tag{2}$$

In previous exercises, we used Naïve Bayes and a simple conditional independence structure to help to learn $p(\boldsymbol{x}|y)$ from data. Recall that the class $y \in \{0, 1, \ldots, 9\}$ represents the true digit. In previous activities, we converted the images to black and white, i.e, $\boldsymbol{x} \in \{0, 1\}^{784}$ corresponded to the 28×28 black and white image. In this problem, we will imagine that the images are real valued vectors generated by a class conditional Gaussian distribution. More specifically, we will assume that

$$p(\boldsymbol{x}|y) \sim \mathcal{N}(\boldsymbol{\mu}_y, \boldsymbol{\Sigma}_y) \text{ for } y = 0, 1, 2, \ldots, 9 \tag{3}$$

For this exercise, do not convert the images to black/white, but instead leave them as gray scale (0-255).

a) What are the dimensions of $\boldsymbol{\mu}_y$ and $\boldsymbol{\Sigma}_y$?

b) Estimate the empirical mean and covariance matrices $\widehat{\boldsymbol{\mu}}_y, \widehat{\boldsymbol{\Sigma}}_y$ for $y = 0, 1, \ldots, 9$ from the training dataset.

c) Display the mean as a 28×28 image for each of the 10 classes. Does the mean look as expected?

d) Computing the Gaussian pdf (or the log-likelihood ratio) involves inverting the sample covariance. If training data is limited, this will fail as the sample covariance has rank less than or equal to the number of samples of that class. There are a number of ways to deal with this, and we will address it using the simplest approach. Define the regularized covariance estimate as

$$\widehat{\boldsymbol{\Sigma}}_\lambda = \widehat{\boldsymbol{\Sigma}} + \lambda \boldsymbol{I}$$

Argue that $\widehat{\boldsymbol{\Sigma}}_\lambda$ is always invertible if $\lambda > 0$.

e) Write a function that computes log likelihood of a new image for each class $y = 0, 1, \ldots, 9$. The inputs to the function should be the vector under test, a mean vector, and the inverse of the regularized covariance matrix.

f) Set $\lambda = 1$. Classify each of the test images by selecting the maximum likelihood estimate of the class. What is the classification error of your classifier on the $10k$ test images? *Hint: your code will run faster if you compute the inverse of the regularized covariance once for each class, instead of computing it for each test image.*

g) Loop over a handful of values of λ ranging from 10^5 to 10^{-3} . Create a plot of the accuracy as a function of λ.

h) What is the best classification error on the $10k$ test images?

i) What value of λ is best?

j) **Optional.** Find a better approach to regularizing the covariance matrix, and test the MNIST dataset using the other approach (one promising approach is called the graphical LASSO).

SOLUTION:

a) Since the images are 28×28, and we vectorize them, we have $\boldsymbol{x} \in \mathbb{R}^{784}$. Hence, $\boldsymbol{\mu}$ is a vector of size 784 and $\boldsymbol{\Sigma}$ is a matrix of size 784 by 784.

b) See associated notebook.

c) Yes - the digit is recognizable from the mean, and looks like a smoothed or averaged version of the handwritten digit.

d) First note that $\widehat{\boldsymbol{\Sigma}}$ is positive semi-definite since it is a sum of outer products of $n \times 1$ vectors. That is, $\boldsymbol{x}^T \widehat{\boldsymbol{\Sigma}} \boldsymbol{x} \geq 0$ for any $\boldsymbol{x} \neq 0$. Next, note that the identity is a positive definite matrix, since $\boldsymbol{x}^T \boldsymbol{x} > 0$ for any $\boldsymbol{x} \neq 0$. Next, $\boldsymbol{\Sigma} + \lambda \boldsymbol{I}$ must be positive definite, since $\boldsymbol{x}^T(\boldsymbol{\Sigma} + \lambda \boldsymbol{I})\boldsymbol{x} = \boldsymbol{x}^T \boldsymbol{\Sigma} \boldsymbol{x} + \lambda \boldsymbol{x}^T \boldsymbol{x} > 0$ for any $\boldsymbol{x} \neq 0$. Since any PD matrix is invertible, this gives the results.

e) See associated notebook.

f) 15.88%.

g) See associated notebook.

h) 4.89%. See the associated notebook.

i) This depends in if the images were scaled or not. If not, $\lambda = 10^3$ gives an accuracy of 4.89%.