CS189: Introduction to Machine Learning

Homework 3

Due: March 1st, 2015, 11:59 pm

Problem 1: Visualizing Eigenvectors of Gaussian Covariance Matrix

We have two one dimensional random variables $X_1 \sim \mathcal{N}(3, 9)$ and $X_2 \sim \frac{1}{2}X_1 + \mathcal{N}(4, 4)$, where $\mathcal{N}(\mu, \sigma^2)$ is a Gaussian distribution with mean μ and variance σ^2. In software, draw $N = 100$ random samples of X_1 and of X_2.

(a) Compute the mean of the sampled data.

(b) Compute the covariance matrix of the sampled data.

(c) Compute the eigenvectors and eigenvalues of this covariance matrix.

(d) On a two dimensional grid with a horizontal axis for X_1 ranging from $[-15, 15]$ and a vertical axis for X_2 ranging from $[-15, 15]$, plot the following:

  i) All $N = 100$ data points

  ii) Arrows representing both covariance eigenvectors. The eigenvector arrows should originate from the mean and have magnitude equal to their corresponding eigenvalues.

(e) By placing the eigenvectors of the covariance matrix into the columns of a matrix $U = [v_1\ v_2]$, where v_1 is the eigenvector corresponding to the largest eigenvalue, we can use U' as a rotation matrix to rotate each of our sampled points from our original (X_1, X_2) coordinate system to a coordinate system aligned with the eigenvectors (without the transpose, U can rotate back to the original axes). Center your data points by subtracting the mean and then rotate each point by U', specifically $x_{rotated} = U'(x - \mu)$. Plot these rotated points on a new two dimensional grid with both axes ranging from [-15,15].

Problem 2: Covariance Matrixes and Decompositions

As described in lecture, a covariance matrix $\Sigma \in \mathbb{R}^{N,N}$ for a random variable $X \in \mathbb{R}^N$ with the following values, where $cov(X_i, X_j) = E[(X_i - \mu_i)(X_j - \mu_j)]$ is the covariance between the ith and jth elements of the random vector X:

$$\Sigma = \begin{bmatrix} cov(X_1, X_1) & ... & cov(X_1, X_n) \\ ... & & ... \\ cov(X_n, X_1) & ... & cov(X_n, X_n) \end{bmatrix} \tag{1}$$

For now, we are going to leave the formal definition of covariance matrices aside and focus instead on some transformations and properties. The motivating example we will use is the N dimensional Multivariate Gaussian Distribution defined as follows:

$$f(x) = \frac{1}{\sqrt{(2\pi)^N|\Sigma|}} e^{-\frac{1}{2}((x-\mu)^\top \Sigma^{-1}(x-\mu)} \tag{2}$$

(a) We usually assume that Σ^{-1} exists, but in many cases it will not. Describe the conditions for which Σ_X^{-1} corresponding to random variable X will not exist. Explain how to convert the random variable X into a new random variable X' without loss of information where $\Sigma_{X'}^{-1}$ does exist.

(b) Consider a data point x drawn from a zero mean Multivariate Gaussian Random Variable $X \in \mathbb{R}^N$ like shown above. Prove that there exists matrix $A \in R^{N,N}$ such that $x^\top \Sigma^{-1} x = \|Ax\|_2^2$ for all vectors x. What is the matrix A?

(c) In the context of Multivariate Gaussians from the previous problem, what is the intuitive meaning of $x^\top \Sigma^{-1} x$ when we transform it into $\|Ax\|_2^2$?

(d) Lets constrain $\|x\|_2 = 1$. In other words, the L2 norm (or magnitude) of vector x is 1. In this case, what is the maximum and minimum value of $\|Ax\|_2^2$? If we have $X_i \perp\!\!\!\perp X_j \ \forall i, j$, then what is the intuitive meaning for the maximum and minimum value of $\|Ax\|_2^2$? To maximize the probability of $f(x)$, which x should we choose?

Problem 3: Isocontours of Normal Distributions

Let $f(\mu, \Sigma)$ denote the density function of a Gaussian random variable. Plot isocontours of the following functions:

a) $f(\mu, \Sigma)$, where $\mu = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$ and $\Sigma = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}$

b) $f(\mu, \Sigma)$, where $\mu = \begin{bmatrix} -1 \\ 2 \end{bmatrix}$ and $\Sigma = \begin{bmatrix} 3 & 1 \\ 1 & 2 \end{bmatrix}$

c) $f(\mu_1, \Sigma_1) - f(\mu_2, \Sigma_2)$, where $\mu_1 = \begin{bmatrix} 0 \\ 2 \end{bmatrix}$, $\mu_2 = \begin{bmatrix} 2 \\ 0 \end{bmatrix}$ and $\Sigma_1 = \Sigma_2 = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix}$

d) $f(\mu_1, \Sigma_1) - f(\mu_2, \Sigma_2)$, where $\mu_1 = \begin{bmatrix} 0 \\ 2 \end{bmatrix}$, $\mu_2 = \begin{bmatrix} 2 \\ 0 \end{bmatrix}$, $\Sigma_1 = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix}$ and $\Sigma_2 = \begin{bmatrix} 3 & 1 \\ 1 & 2 \end{bmatrix}$

e) $f(\mu_1, \Sigma_1) - f(\mu_2, \Sigma_2)$, where $\mu_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$, $\mu_2 = \begin{bmatrix} -1 \\ -1 \end{bmatrix}$, $\Sigma_1 = \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}$ and $\Sigma_2 = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$

Problem 4: Gaussian Classifiers for Digits

In this problem we will build Gaussian classifiers for digits in MNIST. More specifically, we will model each digit class as a Gaussian distribution and make our decisions on the basis of posterior probabilities. This is a generative method for classifying images where we are modelling the class conditional probabilities as normal distributions. The steps mentioned below should be done for each training set in `train.mat` and you need to plot a curve of error rate vs no. of training examples upon evaluating on the test set in `test.mat`. Submit your trained labels for the `kaggle.mat` dataset on the Kaggle competition website. Please use do not use the datasets that we provided in the HW1.zip folder, and only use the datasets provided in the current HW3.zip folder. We have randomized the MNIST test and training sets.

a) Taking raw pixel values as features, fit a Gaussian distribution to each digit class using maximum likelihood estimation. This involves finding the means and covariance matrices for each digit class. Say we have i.i.d observations $X_1 ... X_n$, what are the maximum likelihood estimates for the mean and covariance matrix of a Gaussian distribution? Are these estimators unbiased? (No need to prove for covariance)
Tip: It is a good idea to contrast normalize images before using the raw pixel values. One way of normalization is to divide the pixel values of an image by the l_2 norm of its pixel values.

b) How would you model the prior distribution for each class? Compute prior probabilities for all classes.

c) Visualize the covariance matrix for a particular class. Do you see any kind of structure in the matrix? What does this symbolize?

d) We will now classify digits in the test set on the basis of posterior probabilities using two different approaches:

   i) Define $\Sigma_{overall}$ to be the average of the covariance matrices of all the classes. We will use this matrix as an estimate of the covariance of all the classes. This amounts to modelling class conditionals as Gaussians ($\sim \mathcal{N}(\mu_i, \Sigma_{overall})$) with different means and the same covariance matrix. Using this form of class conditional probabilities, classify the images in the test set into one of the 10 classes assuming 0-1 loss and compute the error rate and plot it over the following number of randomly chosen training data points [100, 200, 500, 1000, 2000, 5000, 10000, 30000, 60000]. Expect some variance in your error rate for low training data scenarios. What is the form of the decision boundary in this case? Why?

ii) We can also model class conditionals as $\mathcal{N}(\mu_i, \Sigma_i)$, where Σ_i is the estimated covariance matrix for the i^{th} class. Classify images in the test set using this form of the conditional probability (assuming 0-1 loss) and compute the error rate and plot it over the following number of randomly chosen training data points [100, 200, 500, 1000, 2000, 5000, 10000, 30000, 60000]. What is the form of the decision boundary in this case?

iii) Compare your results in parts a and b. What do you think is the source of difference in the performance?

iv) Train your best classifier using `train.mat` and classify the images in `kaggle.mat`. Submit your labels to the online Kaggle competition and record your optimum prediction rate. If you used an additional featurizer, please describe your implementation. Please only use any extra "image featurizer" code on this portion of the assignment.

Note: In your submission, you need to include learning curves (error-rate vs no. of training examples) and actual error-rate values for the above two cases and short explanations for the all the questions. Also, the covariance matrices you compute using MLE might be singular (and thus non-invertible). In order to make them non-singular and positive definite, you can add a small weight to their diagonals by setting $\Sigma_i = \Sigma_i + \alpha I$, where α is the weight you want to add to the diagonals. You may want to use k-fold cross validation to see what the optimum "small weight" is.

e) Now that you have developed Gaussian classification for digits, lets apply this to spam. Use the training and testing data located in `spam_dataset.mat` to generate a set of test labels that you will submit to the online Kaggle competition and record your optimum prediction rate. If you used an additional featurizer, please describe your implementation.

Optional: If you use the default feature set, you may obtain relatively low classification rates. The TA's suggest using a bag of words model. You may download 3rd party packages if you wish. Also, normalizing your vectors like before may help

f) *Extra for Experts:* Using the `training_data` and `training_labels` in `spam_dataset.mat`, identify 10 words in your features set corresponding to the maximum and minimum variances. Use k-fold cross validation to train your classifier only using 10 variance maximum words and record your average classification rate. Do the same with the 10 minimum variance words. What do you notice? Can you tie this in with what you proved in part 1.d)? Will the assumption of independence between words hold here? For more information: **PCA, Courtesy of Professor Laurent El Ghaoui**

Problem 5: Centering and Ridge Regression
You are given a training set $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$. Let $\mathbf{X}$ be the design matrix (i.e. the matrix whose i^{th} row is $\mathbf{x}_i$), and let $\mathbf{y}$ be the column vector whose i^{th} entry is y_i. Let $\mathbf{1}$ be a $n \times 1$ column vector of ones.

Define $\bar{\mathbf{x}} = \frac{1}{n}\sum_i \mathbf{x}_i$ and $\bar{y} = \frac{1}{n}\sum_i y_i$. Assume that the input data has been centered, so that $\bar{\mathbf{x}} = 0$. Show that the optimizer of the following loss function $J(\mathbf{w}, w_0)$

$$J(\mathbf{w}, w_0) = (\mathbf{y} - \mathbf{X}\mathbf{w} - w_0\mathbf{1})^\top(\mathbf{y} - \mathbf{X}\mathbf{w} - w_0\mathbf{1}) + \lambda\mathbf{w}^\top\mathbf{w}$$

is given by

$$\hat{w_0} = \bar{y} \text{ , } \hat{\mathbf{w}} = (\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^\top\mathbf{y}.$$

Problem 6: MLE For Simple Linear Regression
Simple linear regression refers to the case of linear regression in which the input is a scalar quantity.

Let the data set be $\{(x_i, y_i)\}_{i=1}^n$, where each sample is drawn independently from a joint distribution over input and output: $(x_i, y_i) \sim (X, Y)$. Assume the Gaussian noise setting:

$$y_i|x_i \sim \mathcal{N}(w_0 + w_1 x_i, \sigma^2)$$

Show that the MLE in this simple linear regression model is given by the following equations, which may be familiar from basic statistics classes:

$$\begin{aligned} w_1 &= \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sum_i (x_i - \bar{x})^2} = \frac{\sum_i x_i y_i - n\bar{x}\bar{y}}{\sum_i x_i^2 - n\bar{x}^2} \approx \frac{\text{cov}(X,Y)}{\text{var}(X)} \\ w_0 &= \bar{y} - w_1\bar{x} \approx \text{E}[Y] - w_1\text{E}[X] \end{aligned}$$

Problem 7: Independence vs. Correlation

(a) Consider the random variables X and Y $\in \mathbb{R}$ with the following conditions.

   (i) X and Y can take values [-1,0,1].

   (ii) When X is 0, Y takes values 1 and -1 with equal probability ($\frac{1}{2}$). When Y is 0, X takes values 1 and -1 with equal probability ($\frac{1}{2}$).

   (iii) X and Y are 0 with equal probability ($\frac{1}{2}$).

   Are X and Y uncorrelated? Are X and Y independent? Prove your assertions. *Hint:* Graph these points onto the Cartesian Plane. What's each point's joint probability?

(b) Consider three Bernoulli random variables B_1, B_2, B_3 which take values $\{0, 1\}$ with equal probability. Lets construct the following random variables X, Y, Z: $X = B_1 \oplus B_2$, $Y = B_2 \oplus B_3$, $Z = B_1 \oplus B_3$, where $\oplus$ indicates the XOR operator. Are X, Y, and Z pairwise independent? Mutually independent? Prove it.

Problem 8: Real World Spam Classification Teasers (Extra for Experts!)
Motivation: After taking CS 189, students should understand the math and design of machine learning algorithms and be able to wrestle with ”real world” data and problems. These issues might be deeply technical and require theoretical background, or might demand a large amount of domain knowledge. Here are some examples that the TA's have encountered during their work.

For each response, please write a short paragraph (around 5-10 sentences) explaining your thought process. This question is open ended and can be many correct answers, and you should explain and support your ideas effectively. Feel free to write more and do your own research if you want to dive deeper, we will look highly on those who really dig deep and provide references!

(a) Daniel recently interned as an anti-spam product manager for a large email service provider. His company uses a linear SVM to predict whether an incoming spam message is spam or ham. He notices that the number of spam messages received tend to spike massively upwards a couple minutes before and after midnight. Eager to obtain a return offer, he adds the time-stamp of the received message, stored as number of milliseconds since the previous midnight, to each feature vector for the SVM to train on, in hopes that the ML model will be able to identify the abnormal spike in spam volume at night. To his dismay, after A/B testing with his newly added feature, Daniel discovers that the linear SVM's success rate barely improves. He wants to try to use a kernel to improve his model, but unfortunately he is limited to a degree two quadratic kernel.

Why can't the linear SVM utilize the new feature well, and what can Daniel do to improve his results? (Note: This was an actual interview question Daniel received for a Machine Learning Engineer position!)

(b) While interning as an anti-spam product manager, Daniel finds out that his company doesn't use a content-based spam filter. GASP! All of his knowledge learned in school... All those hours spent climbing the leaderboards on Kaggle... WASTED?!! Unacceptable, Daniel storms into his bosses room to find out why. Why are content-based spam filters not solely used in spam classification? What problems can you identify with the content-based model, and how can they be mitigated? What other models could one use in conjunction?

Hint: One can approach this problem from many perspectives, not limited to: model/feature selection, computational complexity, economic and risk implications

Submission Instructions

In your submission, you need to include a write up with answers to all the questions and the plots. You also need to include your code and a README with instructions as to how we can run your code. All solutions should be submitted via bCourses. Please submit your competition scores early since you will only get 2 submissions each day!

Please indicate in the README and in your bCourses submission your name and student ID.

Good luck!