Assignment 3: CS 215

Due: 1st October September before 11:55 pm

All members of the group should work on all parts of the assignment. Copying across groups or from other sources is not allowed. We will adopt a zero-tolerance policy against any violation.

Submission instructions:

1. You should type out a report containing all the answers to the written problems in Word (with the equation editor) or using Latex, or write it neatly on paper and scan it. In either case, prepare a single pdf file.
2. The report should contain names and roll numbers of all group members on the first page as a header.
3. Put the pdf file and the code for the programming parts all in one zip file. The pdf should contain the names and ID numbers of all students in the group within the header. The pdf file should also contain instructions for running your code. Name the zip file as follows: A2-IdNumberOfFirstStudent-IdNumberOfSecondStudent-IdNumberofThirdStudent.zip. (If you are doing the assignment alone, the name of the zip file is A2-IdNumber.zip, if there are two students it should be A2-IdNumberOfFirstStudent-IdNumberOfSecondStudent.zip).
4. Upload the file on moodle BEFORE 11:55 pm on the due date. We will nevertheless allow and not penalize any submission until 10:00 am on the following day (i.e. 2nd October). No assignments will be accepted thereafter.
5. Note that only one student per group should upload their work on moodle, though all group members will receive grades.
6. Please preserve a copy of all your work until the end of the semester.

Questions:

1. Images acquired by a camera suffer from random errors which are called 'noise'. People in image processing have always tried to determine the distribution of the noise affecting image acquisition. In this exercise, we will study at one (surprisingly simple) way to do this. Let us restrict ourselves to grayscale images for simplicity, i.e. images $I(x, y)$ which contain single intensity values at each pixel (x, y). Suppose I mount a camera on a stable tripod stand and acquire $N = 10000$ pictures of a checkerboard pattern consisting of multiple well-separated small squares each with a different constant intensity level. Taking so many pictures is feasible if the camera is configured in 'burst mode'. Given so many pictures of the same static scene, explain how the noise distribution can be obtained. Define mathematical symbols precisely and use them in your explanation. In your answer, assume that the noisy image is given by $J(x, y) = I(x, y) + W(x, y)$ where $E(W(x, y)) = 0$ and the noise samples in all pixels are independent. [15 points]
2. (a) A student is trying to design a procedure to generate a sample from a distribution function F, where F is invertible. For this, (s)he generates a sample u_i from a $[0, 1]$ uniform distribution using the 'rand' function of MATLAB, and computes $v_i = F^{-1}(u_i)$. This is repeated n times for $i = 1...n$. Prove that the values $\{v_i\}_{i=1}^n$ follow the distribution F. [6 points]

   (b) Let $Y_1, Y_2, ..., Y_n$ represent data from a continuous distribution F. The empirical distribution function F_e of these data is defined as $F_e(x) = \frac{\sum_{i=1}^n \mathbf{1}(Y_i \leq x)}{n}$ where $\mathbf{1}(z) = 1$ if the predicate z is true and 0 otherwise. Now define $D = \max_x |F_e(x) - F(x)|$. Also define $E = \max_{0 \leq y \leq 1} \left| \frac{\sum_{i=1}^n \mathbf{1}(U_i \leq y)}{n} - y \right|$ where $U_1, U_2, ..., U_n$ represent data from a $[0, 1]$ uniform distribution. Now prove that $P(E \geq d) = P(D \geq d)$.

   (c) Explain what you think is the practical significance of this result in statistics. [3+6+6=15 points]

3. (a) In this exercise, we will consider maximum likelihood based plane fitting. Let the equation of the plane be $z = ax + by + c$. Let us suppose we have access to accurate X and Y coordinates of some N points lying on the plane. We also have access to the Z coordinates of these points, but those have been corrupted independently by noise from $\mathcal{N}(0, \sigma^2)$. Write down the log-likelihood function $\mathcal{L}$ to be maximized in order to determine a, b, c. Write down three linear equations corresponding to setting partial derivatives of $\mathcal{L}$ w.r.t. a, b, c (respectively) to 0. Express these equations in matrix and vector form. [3+4=7 points]

   (b) Repeat the previous part if z had the form $z = a_1x^2 + a_2y^2 + a_3xy + a_4x + a_5y + a_6$. Again, let us suppose we have access to accurate X and Y coordinates of some N points lying on the plane. We also have access to the Z coordinates of these points, but those have been corrupted independently by noise from $\mathcal{N}(0, \sigma^2)$. Write down the log-likelihood function $\mathcal{L}$ to be maximized in order to determine $a_1, a_2, ..., a_6$. Write down linear equations corresponding to setting partial derivatives of $\mathcal{L}$ w.r.t. $a_1, a_2, ..., a_6$ (respectively) to 0. Express these equations in matrix and vector form. [4+4=8 points]

   (c) Is knowledge of the noise variance required for the model fitting in the previous two questions? How will you estimate the noise variance? [5 points]

   (d) Now write MATLAB code to solve this linear system for data consisting of XYZ coordinates of $N = 2000$ points, stored in the file 'XYZ.txt' in the homework folder. Read the data using the MATLAB function `dlmwread`. The data consist of N rows, each containing the X,Y,Z coordinates of a point (in that order). What is the predicted equation of the plane? What is the predicted noise variance? State these in your report, and print them out via your code. [10 points]

   (e) Now suppose that due to some data processing error, some or all coordinates of a small number of the N entries were exchanged with those of other entries. Devise an algorithm to estimate the plane parameters in this case. You do not need to implement it, but just describe your algorithm and explain why it will work. [15 points]

4. We have extensively seen parametric PDF estimation in class via maximum likelihood. In many situations, the family of the PDF is however unknown. Estimation under such a scenario is called nonparametric density estimation. We have studied one such technique in class, namely histogramming, and we also analyzed its rate of convergence. There is another popular technique for nonparametric density estimation. It is called KDE or Kernel density esitmation, the formula for which is given as $\hat{p}_n(x;\sigma) = \frac{\sum_{i=1}^{n} \exp\left(-(x-x_i)^2/(2\sigma^2)\right)}{n\sigma\sqrt{2\pi}}$. Here $\hat{p_n}(x)$ is an estimate of the underlying probability density at value x, $\{x_i\}_{i=1}^n$ are the n samples values, from which the unknown PDF is being estimated, and σ is a bandwidth parameter (similar to a histogram bin-width parameter). The choice of the appropriate σ is not very straightforward. We will implement one possible procedure to choose σ - called cross-validation. For this, do as follows:

   (a) Use MATLAB to draw $n = 1000$ independent samples from $\mathcal{N}(0, 16)$. We will use a random subset of 750 samples (set T) for building the PDF, and the remaining 250 as the validation set V. Note that T and V must be disjoint sets.

   (b) In your report, write down an expression for the joint likelihood of the samples in V, based on the estimate of the PDF built from T with bandwidth parameter σ. [3 points]

   (c) For different values of σ from the set $\{0.001, 0.1, 0.2, 0.9, 1, 2, 3, 5, 10, 20, 100\}$, write MATLAB code to evaluate the log of the joint likelihood LL of the samples in V, based on the estimate of the PDF built from T. Plot of a graph of LL versus $\log\sigma$ and include it in your report. In the report, state which value of σ yielded the best LL value, and print it via your code as well. This procedure is called cross-validation. For this best sigma, plot a graph of $\hat{p}_n(x;\sigma)$ for $x \in [-8 : 0.1 : 8]$ and overlay the graph of the true density on it, for the same values of x. Include this plot in your report. [7 points]

   (d) In this experiment, we know the ground truth pdf which we shall denote as $p(x)$. So we can peek into it, in order to choose the best σ. This is impractical in actual experiments, but for now it will serve as a method of comparison. For each σ, write MATLAB code to evaluate $D = \sum_{x_i \in V}(p(x_i) - \hat{p}_n(x_i;\sigma))^2$. Plot of a graph of D versus $\log\sigma$ and include it in the report. In the report, state which value of σ yielded the best D value, and also what was the D value for the σ parameter which yielded the best LL. For this best sigma, plot a graph of $\hat{p}_n(x;\sigma)$ for $x \in [-8 : 0.1 : 8]$ and overlay the graph of the true density on it, for the same values of x. Include this plot in your report. [7 points]

(e) Now, suppose the set T and V were equal to each other. What happens to the cross-validation procedure, and why? Explain in the report. [4+4=8 points]