# Assignment 1: CS 215

Due: 10th August before 11:55 pm, 100 points

**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. 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: A1-IdNumberOfFirstStudent-IdNumberOfSecondStudent.zip. (If you are doing the assignment alone, the name of the zip file is A1-IdNumber.zip).
3. Upload the file on moodle BEFORE 11:55 pm on the due date (i.e. 10th August). We will nevertheless allow and not penalize any submission until 6:00 am on the following day (i.e. 11th August). No assignments will be accepted thereafter.
4. Note that only one student per group should upload their work on moodle.
5. Please preserve a copy of all your work until the end of the semester.

**Questions:**

1. Given $n$ distinct values $\{x_i\}_{i=1}^n$ with mean $\mu$ and standard deviation $\sigma$, prove that for all $i$, we have $|x_i - \mu| \leq \sigma\sqrt{n-1}$. [15 points]
2. Given $n$ values $\{x_i\}_{i=1}^n$ having mean $\mu$, median $\tau$ and standard deviation $\sigma$, prove that $|\mu - \tau| \leq \sigma$. Assume $n$ is even. Is this result stronger or weaker than the two-sided Chebyshev's inequality? Justify. [15 points]
3. A contestant is on a game show and is allowed to choose between three doors. Behind one of them lies a car, behind the other two there lies a stone. The contestant will get whatever is behind the door that (s)he picked, and quite naturally (s)he wants the car. Suppose (s)he chooses the first door, and the host of the show who knows what is behind every door, opens (say) the third door, behind which there lies a stone (without opening the first door). The host now asks the contestant whether (s)he wishes to choose the second door instead of the first one. Your task is to determine whether switching the contestant's choice is going to increase his/her chance of winning the car. Remember that the host is intelligent: (s)he is always going to open a door not chosen by the contestant, and is also going to open a door behind which there is a stone. You should approach this problem only from the point of view of conditional probability as follows. To this end, let $C_1, C_2, C_3$ be events that the car is behind doors 1,2,3 respectively. Assume $P(C_i) = 1/3, i \in \{1, 2, 3\}$.
   (a) Let $Z_1$ be the event that the contestant chose door 1. Write down the value of $P(C_i|Z_1)$ for all $i \in \{1, 2, 3\}$.
   (b) Let $H_3$ be the event that the host opened door 3. Write down the value of $P(H_3|C_i, Z_1)$ for all $i \in \{1, 2, 3\}$.

(c) Clearly the conditional probability of winning by switching is $P(C_2|H3, Z1)$. This is equal to $\frac{P(H_3|C_2, Z_1)P(C2, Z1)}{P(H_3, Z_1)}$. Evaluate this probability. Note that $P(A_1, A_2)$ denotes the joint probability of events $A_1, A_2$.

(d) Likewise evaluate $P(C_1|H3, Z1)$.

(e) Conclude whether switching is indeed beneficial. The answer is quite surprising :-). [2+2+5+5+1=15 points]

*In the following problems, you can use the mean, median and standard deviation functions from MATLAB.*

4. Generate a sine wave in MATLAB of the form $y = 5\cos(2.2x + \pi/3)$ where $x$ ranges from -3 to 3 in steps of 0.02. Now randomly select a fraction $f = 30\%$ of the values in the array $y$ (using MATLAB function 'randperm') and corrupt them by adding random values from 100 to 120 using the MATLAB function 'rand'. This will generate a corrupted sine wave which we will denote as $z$. Now your job is to filter $z$ using the following steps.

   - Create a new array $y_{median}$ to store the filtered sine wave.
   - For a value at index $i$ in $z$, consider a neighborhood $N(i)$ consisting of $z(i)$, 8 values to its right and 8 values to its left. For indices near the left or right end of the array, you may not have 8 neighbors in one of the directions. In such a case, the neighborhood will contain fewer values.
   - Set $y_{median}(i)$ to the median of all the values in $N(i)$. Repeat this for every $i$.

   This process is called as 'moving median filtering', and will produce a filtered signal in the end. Repeat the entire procedure described here using the arithmetic mean instead of the median. This is called as 'moving average filtering'. Repeat the entire procedure described here using the first quaritle (25 percentile) instead of the median. This is called as 'moving quartile filtering'. Plot the original (i.e. clean) sine wave $y$, the corrupted sine wave $z$ and the filtered sine wave using each of the three methods on the same figure in different colors. Introduce a legend on the plot (find out how to do this in MATLAB). Include an image of the plot in your report. Now compute and print the relative mean squared error between each result and the original clean sine wave. The relative mean squared error between $y$ and its estimate $\hat{y}$ (i.e. the filtered signal - by any one of the different methods) is defined as $\frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i y_i^2}$.
   Now repeat all the steps above using $f = 60\%$, and include the plot of the sine waves in your report, and write down the relative mean square error values.
   Which of these methods (median/quartile/arithmetic mean) produced better relative mean squared error? Why? Explain in your report. [6+5+3+3+3=20 points]

5. Suppose that you have computed the mean, median and standard deviation of a set of $n$ numbers stored in array $A$ where $n$ is very large. Now, you decide to add another number to $A$. Write a MATLAB function to update the previously computed mean, another MATLAB function to update the previously computed median, and yet another MATLAB function to update the previously computed standard deviation. Note that you are not allowed to simply recompute the mean, median or standard deviation by looping through all the data. You may need to derive formulae for this. Include the formulae and their derivation in your report. Note that your MATLAB functions should be of the following form

```
function newMean = UpdateMean (OldMean, NewDataValue, n),
function newMedian = UpdateMedian (oldMedian, NewDataValue, A, n),
function newStd = UpdateStd (OldMean, OldStd, NewMean, NewDataValue, n).
```

   Also explain, how would you update the histogram of $A$, if you received a new value to be added to $A$? (Only explain, no need to write code.) **Note:** For updating the median, you may assume that the array $A$ is sorted in ascending order, that the numbers are all unique. For sorted arrays with a even number of

elements, MATLAB returns the answer as $(A(N/2) + A(N/2+1))/2$. You may use MATLAB's convention though it is not strictly required. [5+5+5+5 = 20 points]

6. Determine using a mathematical formula and a computer algorithm the smallest number $n$ of people such that the probability that at least two of them share their birthday is at least $p$ where $p \in \{5, 10, 15, 20, 30, 40, 50, 60, 70, 80, 90, 95, 99, 99.99, 99.9999, 100\}\%$. Plot a graph of $n$ on Y axis versus $p$ on X axis. The algorithm is to be implemented in MATLAB. [15 points]