

Assignment 7: Matlab Functions

Date Due: October 28, 2013

Instructor: Trani

Problem 1

Task 1

A civil engineer is designing a rainstorm water management system for a large parking lot of a new Soccer stadium. The system consists of a ponding area designed to collect water during heavy thunderstorms and channel the water to a water storm pipe system. The system is illustrated in Figure 1.

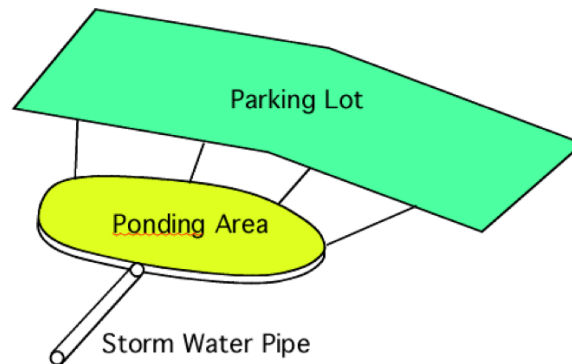


Figure 1. Storm-water management system with ponding area.

The handling capacity of the pipe system is 3 cubic meters per second. During a severe thunderstorm (100-year storm event), the rainfall rate at the stadium parking lot generates water runoff given by the following function:

$$q(t) = 15 * \sin(t / 1500) * e^{(-t/2000)}$$

Where q is the runoff volume (cubic meters per second) generated by the parking lot going into the ponding area and t is the time (in seconds) after the thunderstorm starts.

Task 1

Create a Matlab function to estimate the runoff volume of water (q) for a given value of time (t). Your function should work with array values for time (t).

Task 2

Create a Matlab script to: a) call the function created in Task 1 and b) to plot the resulting values of q vs time.

Task 3

For the 100-year storm event, find the time when the water starts accumulating in the ponding area. Also find the time when the volume in the ponding area reaches a maximum level.

Task 4

Find the total volume of water produced by the 100-year storm event (from $t=0$ to $t=4500$ seconds). Also, find the volume of water accumulated in the ponding area considering the handling capacity of the system. The water in the ponding area is the difference between the water runoff produced minus the handling capacity of the pipes. Use two methods in the computation: a) Trapezoid rule and b) Newton-Cotes (Quad) functions in Matlab.

Problem 2

A civil engineer wants to estimate the reliability of the performance of a batch of high-strength concrete samples. This is important in the design of applications such as the construction of beams and columns in buildings. The engineer collects 100 random samples and finds that the value of the Young's Modulus of the material is normally distributed with a mean of 31 GPa and a standard deviation of 1.7 GPa. The equation of the Probability Density Function (PDF) - called $f(x)$ - for a Normal distribution is given in any statistics textbook and repeated here for completeness (http://en.wikipedia.org/wiki/Normal_distribution).

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\left[\frac{(x-\mu)^2}{2\sigma^2}\right]}$$

where: x is the random variable in question (modules of elasticity in units GPa) and μ and σ are the mean and standard deviation of the random phenomena modeled.

Task 1

Create a first Matlab script to estimate $f(x)$ using the values of μ and σ observed. Plot the Probability Density Function (PDF) as a function of the random variable (x) which represents the Young's Modulus. Label accordingly. Verify that the PDF plot is the well-known bell-shape curve of the Normal distribution.

Task 2

Create a Matlab function to calculate the value of the Normal distribution PDF ($f(x)$) given three arguments: x , μ and σ . The output of the function is the value of $f(x)$.

Task 3

Create a second Matlab script that calls the function created in Task 2 and estimate the probably that concrete specimens fall below the minimum 28 GPa required in the construction project. The area under the curve $f(x)$ below a value ($x=a$) represents the probability that a specimen Young's Modulus falls below a threshold value (a) (see Figure 2). Use the Matlab Quad function in this analysis.

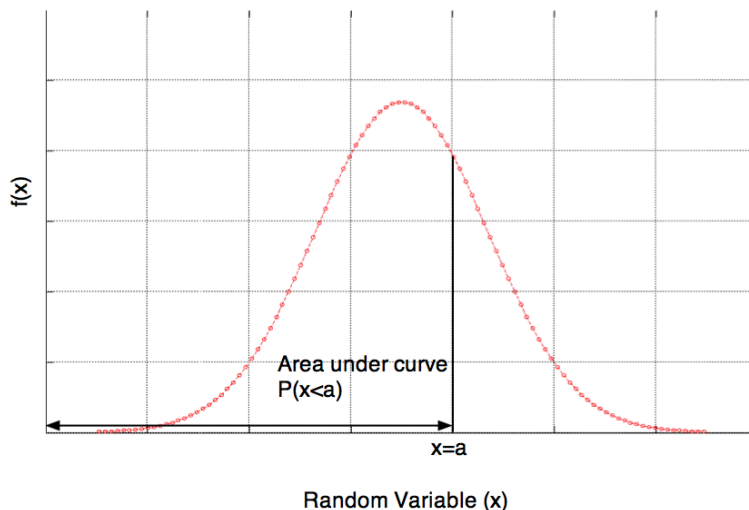


Figure 2. Explanation of area under the curve in Probability and Statistics.

Task 4

Using your Matlab function and script, verify that the area under the $f(x)$ curve is equal to one.

Task 5

Estimate the probably that concrete specimens have Young's Modulus values between 30 and 33 GPa.

Problem 3

Use the car data file (in week 1 of the syllabus) to do this problem.

Task 1

Create a Matlab script to estimate read the values of the car data file. Use any of the input-output functions learned in class. Rename the numeric variables (such as vehicle weight, turning circle, etc) using individual variables.

Task 2

Add to your script created in Task 1 instructions to plot the turning circle (x axis) vs the vehicle weight (y-axis).

Using the Matlab `polyfit` function find the coefficients of the best polynomial of order 1 to estimate the vehicle "weight" (dependent variable) given the vehicle turning circle (independent variable). Plot your regression line using 100 equally spaced values in the turning circle axis. The limits of the equally space values in x should be the minimum and the maximum values of turning circle found in the data (all this is done in code).

Find the coefficient of correlation (R-square) using the Matlab function:

```
R=corrcoef(x,y)
```

where: x and y are the vectors containing the data studied. Comment on how good is the fit to the data.

Verify that your regression coefficients match those obtained if you use the "Basic Fitting" under the Tools pull down menu of the figure containing the plot turning circle vs. vehicle weight.

Task 3

Add to your script created in Task 2 instructions to plot the vehicle weight (x axis) vs the vehicle Horsepower (y-axis).

Using the Matlab `polyfit` function find the coefficients of the best polynomial of order 2 to estimate the vehicle "Horsepower" (independent variable) given the vehicle weight (dependent variable). Plot your regression line using 100 equally spaced values in the vehicle weight axis. The limits of the equally space values in x should be the minimum and the maximum values of vehicle weight found in the data (all this is done in Matlab code).

Find the coefficient of correlation (R-square) using the Matlab function:

```
R=corrcoef(x,y)
```

where: x and y are the vectors containing the data studied. Comment on how good is the fit to the data.