**Francis Lee**

**20.440 Biological Networks**

**Pset1**

https://github.com/francisglee/Problem\_Sets/blob/master/20.440\_Biological\_Networks/Pset1/

**Problem 1 – Probability distributions**

1. The genome of an organism you are studying has a G/C content of 0.4. You are analyzing sequences obtained from this genome and wish to determine probabilities associated with seeing certain sequence characteristics by chance.

I’ve described below a statistical analysis of a random DNA sequence of length, L using a binomial distribution of discrete random variable, X, with a success probability, 𝞱, of 0.6 (GC content of 0.4), predicated on a Bernoulli variable, Y. However a nucleotide sequence in reality, is not a Bernoulli variable as there are hidden interactions that govern nucleotide sequence identity. I don’t think we’ve covered the proper statistical analysis in class, so I’m hoping the analysis described below satisfies the scope of this pset.

* 1. Write down the probability of finding 10 A/T bases in a window of 30 bases. They can be non-consecutive. Please write this in terms of a well-described probability distribution.

**Poisson Distribution**

Probability Mass function:

Where:

* 1. Compute the actual number by hand and check with Matlab. (HINT: Matlab contains a number of probability density function calculators with the general name xpdf, where x is some short-hand for the distribution. For example, for a Normal PDF, you can use normpdf with appropriate parameters for this distribution type as arguments).

I’m guessing it would be incorrect to use a PDF here as this function is not continuous? And if it were, you wouldn’t be able to obtain a non-zero number for P without a range for X.

**Probability of finding 10 A/T bases using a Poisson mass function:**

Where:

k = 10

n = 30

p = 0.6

* 1. What is the probability of less than or equal to ten A or T bases in the same window? Write an equation for solving this probability and solve for the number in Matlab.

**Probability of finding 10 A/T bases using a poisson cumulative distribution function:**

* 1. What is the probability of between 10 and 15 A or T bases in the same window? Write an equation for solving this probability and solve for the number in Matlab.

**Probability of finding between 10 and 15 A/T bases using a binomial continuous distribution function:**

* 1. Plot the full probability distribution for observing 0:30 A or T bases in the 30 base window. To refresh your memory, an example of basic plotting in Matlab for this problem would be:

x = 0:30 % The range of values over which you will plot

y = ; % Use appropriate code/function to produce probabilities

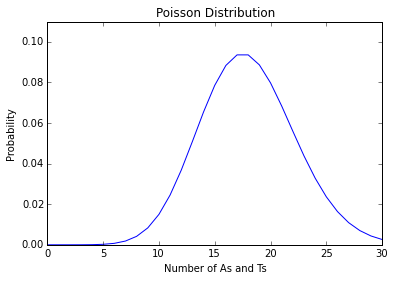
figure();% Create a figure window

plot(x,y,’b-o’);% Plot as blue line with markers at x values used

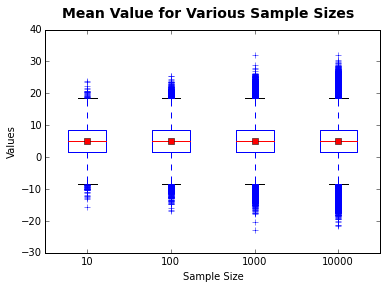
set(gca,’FontSize’,14); % Increase plot font size to 14

xlabel(‘Number of A or T bases’); % provide x-label

ylabel(‘Probability’); % Provide y-label



1. In this question you will explore how much sampling you need to do to get an accurate estimate of the parameters of a probability distribution. You will use the randn function in Matlab to randomly sample from a Normal distribution with a true mean of 5 and standard deviation of 5. For each sample size N = [10, 100, 1000, 10000], randomly generate N samples from the distribution and use those values to estimate the mean. Repeat this process 1000 times, storing the means for each of the 4\*1,000 samples. Plot a box-plot of the mean values for each value of N.



1. A bag contains 10 marbles, 4 of which are red. You randomly draw 4 marbles from the bag at the same time and notice that 3 of them are red.
   1. Manually approximate the probability of drawing 3 **or more** red marbles using the binomial distribution.

I think that the binomial distribution is not the right model for this problem, as it doesn’t take into account N total number of samples..

𝒴 = {0, 1}

**Binomial Distribution**

Probability Mass function:

* 1. Manually calculate exactly the probability of drawing 3 **or more** red marbles using the hypergeometric distribution.

**Hypergeometric Distribution**

Probability Mass function:

where:

* 1. Demonstrate the convergence of the hypergeometric distribution to the binomial distribution as background size increases by calculating with the hygecdf function in Matlab the probability of drawing 3 **or more** red marbles in 4 tries from a bag containing:
     1. 10 marbles, of which 4 are red
     2. 50 marbles, of which 20 are red
     3. 100 marbles, of which 40 are red
     4. 1000 marbles, of which 400 are red
     5. 5000 marbles, of which 2000 are red
     6. 10000 marbles, of which 4000 are red

First, I’ll calculate the binomial distribution for each scenario, which should be unchanged, as it is independent of sample size. From 1Ca:

**Binomial Distribution**

Probability Mass function:

Second, I’ll calculate the hypergeometric probability, defined in 1Cb, for each scenario described in 1Cc:

**Hypergeometric Distribution**

Probability Mass function

Probability Mass function for :

I’ll assign random variables to each scenario as described below:

A hypergeometric probability of picking 3 or more red balls from a bag with 10 marbles and 4 red balls

B hypergeometric probability of picking 3 or more red balls from a bag with 50 marbles and 20 red balls

C hypergeometric probability of picking 3 or more red balls from a bag with 100 marbles and 40 red balls

D hypergeometric probability of picking 3 or more red balls from a bag with 1000 marbles and 400 red balls

E hypergeometric probability of picking 3 or more red balls from a bag with 5000 marbles and 2000 red balls

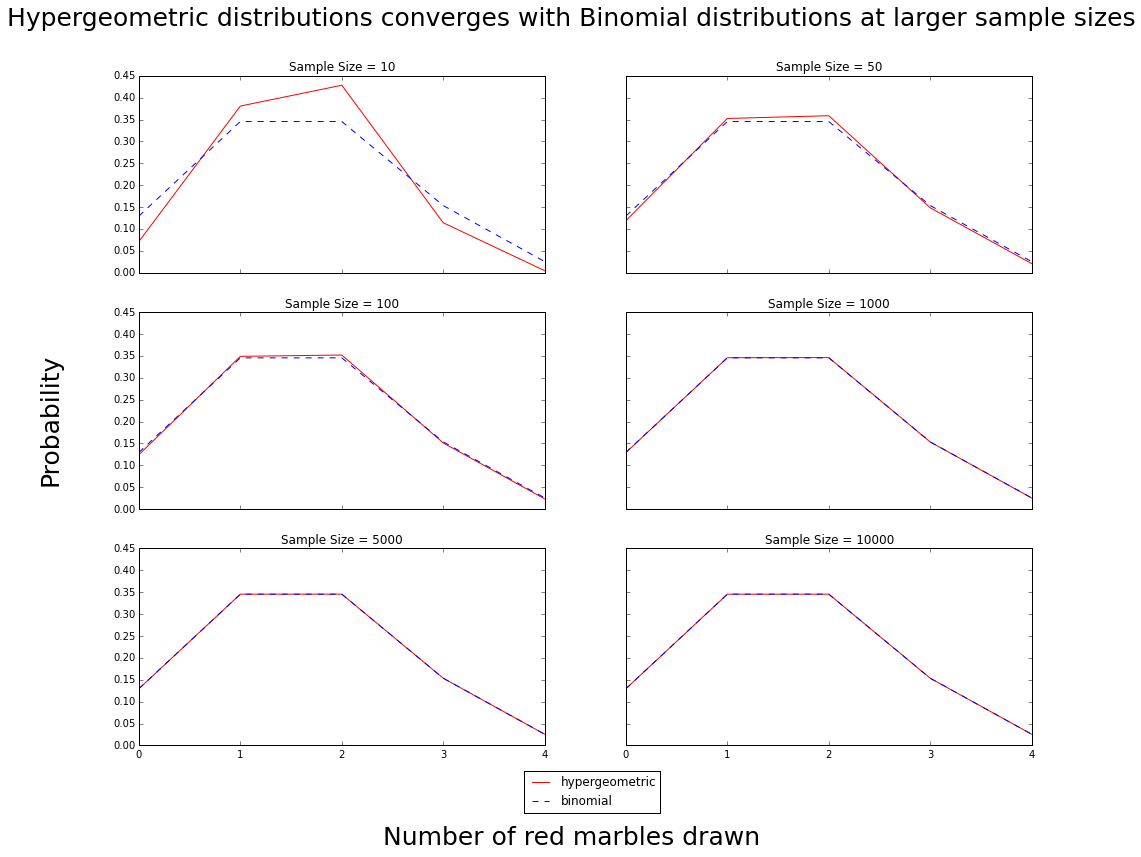
F hypergeometric probability of picking 3 or more red balls from a bag with 10000 marbles and 4000 red balls

And solve for the hypergeometric probability between :

where:

With a binomial probability of 0.1792, the hypergeometric probability appears to converge with the binomial probability as the sample size increase.

* 1. For each of the background sizes in (c) above, also **plot** the probability densities over the range [0 4] (for drawing between 0 and 4 red marbles) for both the binomial and hypergeometric distributions. (NOTE: the subplot function in Matlab is useful for creating multiple plots in one figure window. To create a figure with a 2x3 grid of plots: 1) initialize the figure with figure(); 2) prior to calling the plotting command for each, use subplot(2,3,x) to set up each subplot – x here is an integer value on [1,2,3,4,5,6] ).



**Problem 2 – Hypothesis testing**

1. The Q-Q plot is a simple graphical tool to help decide if data come from a theoretical distribution. The Q stands for “quantile” and defines the fraction of the data that lies below some value. For example, the 0.5 quantile is the same as the median – the value below which half the data lie. If one plots the quantiles in the sample vs. the theoretical quantiles for the correct distribution, the points should lie on a straight line. Since Normal distributions are ubiquitous (due to the center limit theorem) many software packages provide tools to produce Q-Q plots against the Normal distribution: In Matlab, the function is normplot. Plot a Q-Q plot comparing each of the cases below to the Normal distribution:
   1. 1000 points sampled from a **Normal distribution** with mean equal to 0 and standard deviation equal to 1.
   2. 10 points sampled from a **Normal distribution** with mean equal to 0 and standard deviation equal to 1.
   3. 100 points sampled from a **Binomial distribution** with p = 0.5.
   4. 1000 points sampled from a **Binomial distribution** with p = 0.5.
   5. 1000 points sampled from a **uniform distribution** between -1 and 1.
   6. 1000 points where the first 500 are drawn from a **Normal distribution** with mean equal to 5 and standard deviation equal to 1 and where the second 500 are drawn from a **Normal distribution** with mean equal to 10 and standard deviation equal to 2.

I am going to make an assumption of the parameter n (number of trials) for the Binomial distribution because we cannot perform a Binomial sampling with the information provided above.

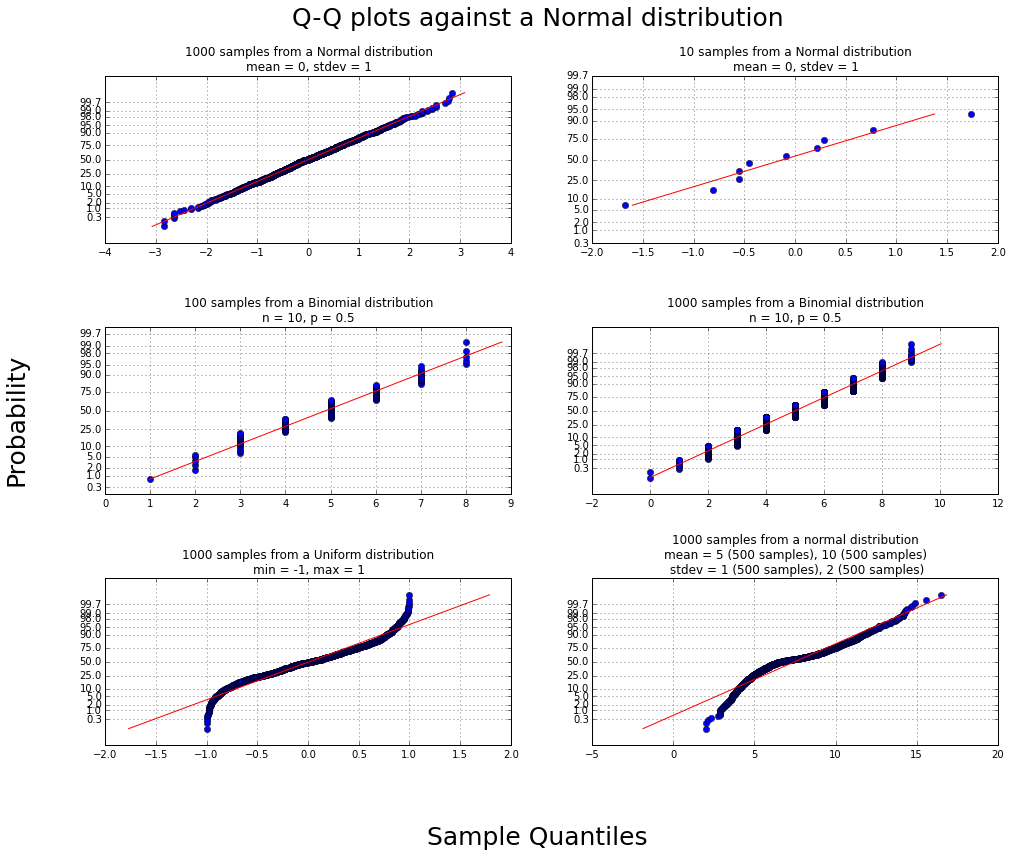
Both the python numpy random binomial sampling function:

where:

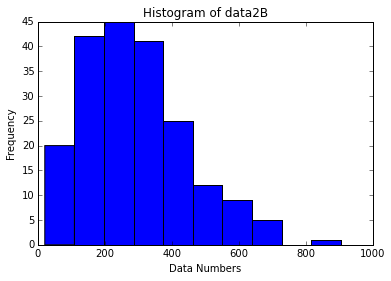
and the MATLAB random binomial sampling function:

where:

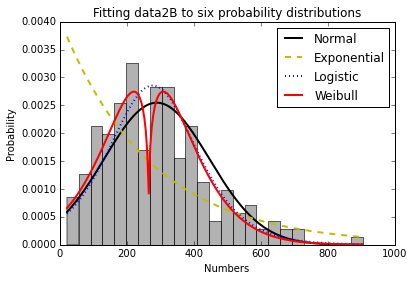
We are given p = 0.5, and N = 100, 1000. I will assume n = 10 for both Binomial distributions because I’m guessing the hope of this question is to evaluate the fitness of the probability distribution (list of n-samples) based on the the sampling size, N.



1. A fellow researcher comes to you with some data that follows a standard statistical distribution, though the correct one is unknown. The researcher believes there are potentially six different distributions to which the data likely fits. Being familiar with methods to assess probability distributions, you agree to help your friend. To help, you perform the following steps (NOTE: the data used here is synthetic data drawn from one of the six probability distributions you will test):
   1. Load the data in the file data2B.mat and visualize the raw data. To load the data, use load(‘data2B.mat’) which will load the data as a vector titled ‘data2B’. The hist function will also be useful for visualizing the data.



* 1. Use the fitdist function to fit the data to each of these six probability distributions: Exponential, Poisson, Normal, Negative Binomial, Logistic, and Weibull.



* 1. Use the qqplot function to plot the data against the fit distribution. Comment on how well/poorly the data fits to each of the distributions.
  2. You decide that the data fits well to more than one of the tested distributions, so you decide to have a closer look at the raw data. You notice that the numbers are all integer values. With this new information, provide a best guess as to the likely distribution that best models the data.

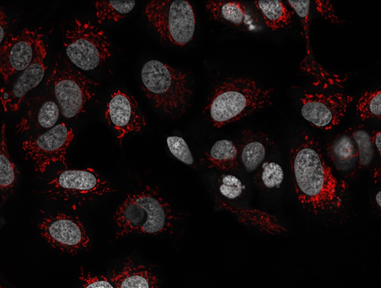
1. Now you will look at two different sets of synthetic data, assess whether or not those data are drawn from Normal distributions, and apply appropriate statistical tests to assess differences between the pairs of measurements. The first set of data in data2C1.mat contains responses drawn from two cell populations that both follow Normal distributions. For this dataset:
   1. Load the data in Matlab using the load function – there are two samples stored as vectors labeled ‘s1’ and ‘s2’ that you should extract (i.e. S1 = data2C1.s1; S2 = data2C1.s2;).
   2. Show with Q-Q plots that these two response sets follow Normal distributions.
   3. Compute the means and variances for both responses.
   4. Choose the most appropriate form of the t-test to assess a difference in the response between these two populations. Consider the sample sizes (equal vs. unequal), the equality of the variances, and whether or not a one-tailed or two-tailed test is most appropriate. Write below the equations most appropriate for calculating the t-statistic and degrees of freedom for the t-distribution.
   5. In Matlab, calculate the t-statistic and degrees of freedom, then use the tcdf function to compute the p-value for this data. According to this value, can you say (and with what confidence) that there is a difference between the means of the two distributions? Describe what the mathematical p-value you obtained tells you in terms of the data.
   6. Now, compare the p-value in (d) to one obtained from a non-parametric test. Use the ranksum function in Matlab to compute the Mann-Whitney U-test p-value. What conclusions can you draw regarding using this non-parametric test on the data versus using the parametric t-test?

Now, investigate the next pair of cell responses contained in data2C2.mat. Load the data as you did for the first pair of responses.

* 1. Comment on whether or not the data appears to follow Normal distributions. Based on your conclusions, propose the most appropriate test for the data.
  2. Report the p-value for the test proposed in (a). Also, compute the p-value from a non-parametric test (e.g. Mann-Whitney) if the choice in (a) involves using a parametric test; do the opposite if you choose a non-parametric test for (a) (e.g. t-test).
  3. Comment on the p-values obtained from the different statistical tests you performed in (b).
  4. What conclusions can be drawn regarding this data given the results of your hypothesis test?

**PROBLEM 3 – Hypothesis testing on real high-content imaging data\***

1. You are a breast cancer researched studying the effects of various compounds on cell stress and/or death. You choose the breast cancer cell line MCF-7 as your model system, which is commonly used in the field. You plan to perform high-content imaging of many individual cells to assess physical disruptions induced by such compounds. You select 16 compounds and treat your cells with several concentrations of these (totaling 61 treatments). You then stain the cells with DRAQ5 (to visualize nuclei and cytoplasm) and TMRE, a read-out of mitochondrial membrane potential, to assist in quantifying various features of the cells following each compound treatment. A sample image looks like this (DRAQ5 in gray, TMRE in red):



After imaging, you use some very sophisticated software to quantify various features of single cells from the images. You then decide to test which compounds at what concentrations have an effect on overall cellular TMRE staining and nuclear compaction. To do this, you must perform hypothesis testing on the cellular populations to decide which treatments significantly affect these parameters. Your task is to write a Matlab script that will perform all your hypothesis tests.

The data is contained in the file “MCF7\_data\_all.txt”, with the column headings:

1. *Cells* – all entries are “MCF7”
2. *ID* – all entries are “ctrl” here
3. *Treatment\_Sum* – an underscore-delimited string of the form {ID}\_{treatmentcompound}\_{concentration}, e.g. ctrl\_ActD\_100nM or ctrl\_Null\_0
4. *Treatment01* – same entries as ID
5. *Treatment02* – the treatment compound
6. *Treatment03* – the concentration of the compound
7. *Ch1\_MOR\_Nucleus\_compact* – the nuclear compaction measurements for each cell
8. *Ch2\_INT\_Cytoplasm\_intensity* – the intensity values for TMRE staining for each cell

The data contained in columns 3, 7, and 8 will be most useful/needed for completing this problem. We have provided a skeleton file (Problem\_3A\_analyze\_MCF7\_data.m) with some code already written to assist you.

1. Read in the data file ‘MCF7\_data\_all.txt’– there are read-outs for 19,703 individual cells quantified from the 62 conditions (61 treatments plus 1 control). We already provide skeleton code using the textscan function to do this, extracting as variables: 1) treatments, which contains the string labels for the different conditions (column 3 above), 2) NC, which contains the nuclear compaction measurements, and 3) TMRE, which contains the TMRE staining intensity measurements.
2. Create normal probability plots for: 1) the nuclear compaction measurements for the conditions ABT737 at 4µM and ActD at 1 nM and 2) TMRE measurements for the conditions ABT737 at 40µM and ActD at 10 nM. Comment on how well these measurements conform to a standard Normal distribution. (NOTE: the strcmp function and logical indexing will be helpful for identifying which rows in the extracted data vectors match to the individual conditions).
3. Write code that will perform 61 hypothesis tests, comparing all the treatment conditions to the control (‘ctrl\_Null\_0’), using the non-parametric Mann-Whitney test (ranksum function in Matlab). The comparisons of TMRE and nuclear compaction can be done separately. Your code will need to store the 61 p-values calculated for each test against the control.
4. Write routines to perform multiple hypothesis corrections on the p-values calculated in (3) above. You should write functions for both the Bonferroni and Benjamini-Hochberg (B-H) FDR methods discussed in class. You will report these corrected p-values along with the raw values. Note that corrected p-values should still not exceed 1 and that the B-H procedure does not necessarily produce a monotonically increasing set of values. A simple, valid fix for this is to set *q(i)* to the smallest calculated *q(k)* for *k*  ≥ *i*. You can check your function against Matlab’s B-H estimates by using mafdr(your\_pvalues,’BHFDR’,’true’).
5. Write your results for the two sets of hypothesis tests with corrections to output files. Your output should include the treatments tested, raw p-values, Bonferroni-corrected values, and B-H FDRs in ascending order (i.e. most significant to least).
6. omment on which compounds had the most significant effects and how the different multiple hypothesis correction procedures altered the results obtained.

\*Data obtained from<http://lincs.hms.harvard.edu/db/datasets/20244/>