PSYCH-GA.2211/NEURL-GA.2201 – Fall 2018
Mathematical Tools for Neural and Cognitive Science

Homework 4

Due: 13 Nov 2018
(late homeworks penalized 10% per day)

See the course web site for submission details. For each problem, show your work - if you only provide the answer, and it is wrong, then there is no way to assign partial credit! And, please don't procrastinate until the day before the due date... *start now*!

1. **Bayes' rule and eye color.** A male and female chimpanzee have blue and brown eyes, respectively. Assume a simple genetic model in which the gene for brown eyes is always dominant (so that the trait of blue eyes can only arise from two blue-eyed genes, but the trait of brown eyes can arise from two brown-eyed genes, or one of each). You can also assume that the apriori probability that each of the four gene configurations is equally probable. For each question, provide the math, and explain your reasoning.

   (a) Suppose you observe that they have a single child with brown eyes. What is the probability that the female chimp has a blue-eyed gene?

   (b) Suppose you observe that they have a second child with brown eyes. Now what is the probability?

   (c) Generalizing, suppose they have N children with brown eyes... express the probability, as a function of N.

2. **Poisson neurons.** The Poisson distribution is commonly used to model neural spike counts:

$$p(k) = \frac{\mu^k e^{-\mu}}{k!},$$

   where k is the spike count (over some specified time interval), and μ is the mean rate over that interval.

   (a) Create a vector `p` of length 21, whose elements contain the probabilities of Poisson spike counts for $k = [0...20]$. Set the mean rate to $\mu = 5$ spikes/interval. Since we're clipping the range at a maximum value of 20, you'll need to normalize the vector so it sums to one (the distribution given above is normalized over the range from 0 to infinity). Write a function `samples = randp(p, num)` that generates `num` samples from the PDF specified by `p`. [Hint: use the `rand` function, which generates real values over the interval $[0...1]$, and partition this interval into portions proportional in size to the probabilities in `p`]. Test your function by drawing 1,000 samples, plotting a histogram of how many times each value is sampled, and comparing this to the frequencies predicted by `p`. Verify qualitatively that the answer gets closer (converges) as you increase the number of samples (try 10 raised to powers $[2, 3, 4, 5]$).

   (b) Imagine you're recording with an electrode from two neurons simultaneously, whose spikes have very similar waveforms (and thus can't be distinguished by the spike sorting software). Create a probability vector, `q`, for the second neuron, assuming a mean rate of 2 spikes/interval. The observed spike counts will be the sum of spike counts from the two neurons (since their spikes cannot be distinguished). What is the PDF of the sum of a sample drawn from `p` and a sample drawn from `q`? [Hint: the output vector should have length $m + n - 1$ when m and n are the lengths of the two input PDFs.]

Verify your answer by comparing it to the histogram of 1,000 samples generated by summing two calls to `randp`. As before, verify that the histogram gets closer as you increase the number of samples.

(c) Now imagine you are recording from a neuron with mean rate 7 spikes/interval (the sum of the rates from the neurons above). Compare the distribution of spike counts for this neuron (computed using the formula above) to the distribution of the sum of the previous two neurons. Based on the results of these two experiments, if we record a new spike train, can you tell whether the spikes you have recorded came from one or two neurons just by looking at their distribution of spike counts?

3. **The Central Limit theorem.** The Central Limit theorem states that the distribution of the average of a set of samples (drawn independently, from any distribution with finite mean and variance) gets closer and closer to a Normal (Gaussian) distribution as the size of the sample increases. Specifically, if the mean and variance of the original distribtion are μ and σ, the distribution of the average converges to $\mathcal{N}(\mu, \sigma/\sqrt{n})$ as n increases.

   (a) Generate 1,000 samples of two values each from a uniform distribution (use `rand`). Compute the average of each sample (pair of values), and plot a histogram of these. What shape is it, approximately? What shape should it have in the limit, as you gather more and more samples (try with 100,000 samples)? Why?

   (b) Now try this again with samples containing 3 values. How has the histogram changed? Try sample sizes of 4 and 5 as well. When do you judge that the histogram starts looking Normal?

   (c) Test the Normality of the distribution a bit more carefully, using a "Q-Q" (quantile-quantile) plot (plot the quantiles of one distribution against another). If the two distributions match, the values should lie on a unit-slope line. For this problem, you can use the matlab function `normplot`, which plots the quantiles of a sample of data against those of a Normal distribution of the same mean and variance. First, try this on a sample of 1,000 values from a normal distribution (use `randn`). The points should fall (close to) a straight line, indicating that the sample is close to normal, as expected. Try this a few times to see how the plot varies (you might want to put them on the same graph, using matlab's `hold on` command). Now call `normplot` on a sample of 1,000 values from a uniform distribution. Explain qualitatively why it has the shape it does (hint: think about the quantiles of the uniform and Normal distributions). Do this for averages of uniform samples of different size (2, 3, 4, ...). Keep increasing sample size until you cant tell the resulting QQ plot from the QQ plots for samples from the Normal distribution. Roughly how big does the sample have to be?

4. **Multi-dimensional Gaussians.**

   (a) Write a function `samples = ndRandn(mean, cov, num)` that generates a set of samples drawn from an N-dimensional Gaussian distribution with the specified `mean` (an N-vector) and `cov`ariance (an NxN matrix). The parameter `num` should be optional (defaulting to 1) and should specify the number of samples to return. The returned value should be a matrix with num columns each containing a sample of N elements. (Hint: use the MATLAB function randn to generate samples from an N-dimensional Gaussian with zero mean and identity covariance matrix, and then transform these to achieve the desired mean/cov. Recall that the covariance of $Y = MX$ is $E(YY^T) = MC_X M^T$ where C_X is the covariance of X).

(b) Now consider the marginal distribution of your 2-D Gaussian in which samples are projected onto a unit vector $\hat{u}$ to obtain a 1-D distribution. Write a mathematical expression for the mean and variance of this marginal distribution as a function of $\hat{u}$ and check it for a set of 48 unit vectors spaced evenly around the unit circle. For each of these, compare the mean and variance predicted from your mathematical expression to the sample mean and variance estimated by projecting your 1,000 samples onto $\hat{u}$. Plot the mathematically computed mean and the sample mean (on the same plot), and also plot the mathematical variance and the sample variance.

(c) Now scatterplot 1,000 samples of a 2-dimensional Gaussian (choose an arbitrary nonzero mean and nonzero covariance). [If you're having trouble getting it working, you might want to first try a zero-mean, unit variance example]. Measure the sample mean and covariance of your data points, comparing to the values that you requested when calling the function. Plot an ellipse on top of the scatterplot, by generating unit vectors equi-spaced around the circle, and rescalling each one to have length proportional to two standard deviations of the data projected onto the corresponding direction specified by that unit vector (as computed using the covariance matrix), and adding the mean. Try this on several random data sets. Does this ellipse capture the shape of the data?

(d) How would you, mathematically, compute the direction (unit vector) that maximizes the variance of the marginal distribution? Compute this direction and verify that it is consistent with your plot.