PSYCH-GA.2211/NEURL-GA.2201 – Fall 2022
Mathematical Tools for Neural and Cognitive Science

Homework 4

Due: 8 Nov 2022
(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. **Bayesian inference of eye color.** A male and female chimpanzee have blue and brown eyes, respectively. The brown-eyed allele can be denoted as a capital B, whereas the blue-eyed allele can be represented as a lowercase b. 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). Children get one allele from each parent. You can also assume: i) the probability of the mother being BB is 50% and the probability of her being Bb is 50%; and ii) the *a priori* probability that a child gets any one of the four gene configurations from the parents is 25%. 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. **Sums of random variables.** Consider a discrete distribution specified by a vector `p` of length `n` whose elements provide the frequency of occurrence of each of the integers $1 \ldots n$. (The values in `p` should be non-negative and sum to 1).

   (a) Write a function `samples = randp(p, num)` that generates `num` samples from the PDF specified by `p`. Test your function by choosing some arbitrary `p` of length 10, 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 10^2, 10^4, 10^6, ...).

   (b) Next, write a function `psum(p, q)` that, for two discrete PDFs `p` and `q`, returns a vector encoding the PDF for the sum of a sample drawn from `p` and a sample drawn from q. Hint: the size of the output vector must cover the full range of possible values when summing the two variables.

   Test your function on `p = [1:6] / sum([1:6])` (a weighted die), and using repeated calls to `psum` to compute the PDF predicted for a sum of *four* rolls of the die. Now compare this PDF to a histogram of samples generated by summing four samples drawn using `randp`. Again, verify that the histogram gets closer to the true distribution as you increase the number of samples.

3. **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` rows 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 X, and then transform these to achieve the desired `mean` and `cov`ariance. Recall that the covariance of $Y = MX$ is $E(YY^T) = MC_XM^T$ where C_X is the covariance of X.) For this, use mean $\mu = [4, 5]$ with $C_Y = [10, -4; -4, 5]$ to sample and scatterplot 1,000 points to verify your function worked as intended.

   (b) Now consider the marginal distribution of a generalized 2-D Gaussian with mean μ and covariance C 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 from part (a) onto $\hat{u}$. Stem plot the mathematically computed mean and the sample mean (on the same plot), and also plot the mathematical variance and the sample variance, both plotted as a function of the angle of $\hat{u}$ with the x-axis.

   (c) Now scatterplot 1,000 new samples of a 2-dimensional Gaussian using the same μ and C_Y from part (a). Measure the sample mean and covariance of your data points, comparing to the values that you requested when calling the function. For each of the unit vectors from (b), find the endpoint of that unit vector scaled by the *standard deviation* of the data projected onto that unit vector. Plot a closed contour that connects all those endpoints. Plot a second closed contour using the theoretical values using the results in (b). Try this on three additional random data sets with different means and covariance matrices. Does this contour 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.

4. **Analyzing and simulating experimental data.** An international coffee conglomerate recruits you to characterize the neuropsychology underlying their customers' adoration of pumpkin spice. You devise a blood-oxygen level dependent (BOLD) fMRI pilot experiment in which you present one of two classes of odorants to an individual while monitoring the activity of three key voxels located in the amygdala, a structure known to be associated with emotional responses. The file `experimentData.mat` contains: a $(N \times 3)$ matrix `data`, where each row is the BOLD response of the three voxels on a given trial relative to some baseline; and a $(N \times 1)$ vector `trialConds` indicating the experimental condition of each trial. Condition 1 includes trials in which you present an odorant selected randomly from a library of possible control odorants, and condition 2 includes trials in which the trade-secret pumpkin-spice odorant is presented.

   (a) Before doing anything quantitative with your data, it is always good practice to visualize it. First, determine how many trials of each condition were completed. Display this information as a 2-bin histogram with each bin representing each of the two possible

conditions, and their heights representing their respective trial counts. Next, plot a 3D scatter plot of the recorded responses, with each point color-coded according to its associated condition (use the function `scatter3` in Matlab (or see footnote[1] for Python) and be sure to label your axes). Describe your data qualitatively using this figure. Is there a noticeable difference between the two conditions? What geometric shape are these 'response clouds', and what distribution would you use to model them?

(b) Quantify the response statistics of each individual condition. Calculate the means of each response cloud, as well as their respective covariance matrices. Compute the covariance matrices of each response cloud using matrix multiplication (remember to center the data first). Verify that your calculation is correct by comparing with the output given by the `cov` function. How do the covariance matrices compare between condition 1 and condition 2 (are they similar at all or wildly different)?

(c) Next, compute the SVD of each covariance matrix. Plot the three singular vectors originating from the center of each response cloud and scale their amplitude by the square root of the singular values. Relative to how similar the covariance matrices were before computing their SVD, how do each condition's respective set of singular values compare? Describe what this tells us about the relationship between the two conditions and, more fundamentally, the relationship between the three voxels across conditions.

(d) A powerful method to validate a model is by *generating* (i.e., simulating) new data matching your quantitative description of the real data, and then comparing them with real data. Create a function

```
simResponses = odorExperiment(numTrials1,numTrials2)
```

where `numTrials1` and `numTrials2` are the number of trials in a simulated experiment for condition 1 and 2, respectively. `simResponses` is a $(N \times 3)$ matrix containing simulated responses of each of your 3 voxels during $N =$ `numTrials1` + `numTrials2` trials. [Hint: use `ndRandn` from the previous problem.] Plot the simulated and real responses in the same figure (use subplots if you wish) to compare the two. Is your simulated response data a good characterization of the real amygdala voxel responses?

[1] Make sure you run `from mpl_toolkits.mplot3d import Axes3D` and `%matplotlib notebook` at some point. Then run `fig = plt.figure(); ax = fig.add_subplot(111, projection('3d')); ax.plot(''whatever you want'')`. Note that this does *not* work in Colab, and you need to have Jupyter notebook on your own computer for interactive 3D plots.