Mathematical Tools for Neuroscience (NEU 314)
Princeton University, Spring 2016
Jonathan Pillow

Homework 6: Probability Basics II:
Gaussians, Expectations, & Central Limit Theorem

Due: Tuesday, April 12, 9:59am

Gaussians, Ellipses, and PCA

1. Draw 200 samples from a 2D (or "bivariate") Gaussian with mean $\mu = [10; 10]$ and covariance $C = [25, -14.5; -14.5, 9]$. Make a scatter plot of the data, using a black dot (or circle) for every data point and a red asterisk for the mean. Plot black lines showing the x and y axes over the range from $[-20, 20]$. (You can use the command "`hold on;`" to ensure that subsequent plot commands don't over-write the output of earlier plot commands. Use "`hold off;`" to turn off this state when you're finished adding lines to the plot, if desired). Set the axes to have equal scaling using "`axis equal;`". Lastly, add ellipses to the plot showing the 1-standard deviation and 2-standard deviation contours of the distribution. (4 points).

   Write the ellipse plotting code yourself using the SVD of C (i.e., don't use any advanced matlab commands beyond `svd`). You can exploit the fact that you previously wrote code to plot a unit circle. To obtain an ellipse, you can transform the points on the 2D circle by first scaling them by the square-root of the eigenvalues (which correspond to the standard deviation of the Gaussian along each singular vector; the raw eigenvalues are variances). Then, multiply them by a matrix containing the singular vectors along each column. In other words, if $C = USU^\top$ is the SVD of C, then you can take points on the unit circle and left-multiply them by the matrix $US^{\frac{1}{2}}$, where $S^{\frac{1}{2}}$ is a diagonal matrix with the square root of the eigenvalues along the diagonal. This will give you an ellipse at 1 standard deviation of the distribution. (Finally, of course, add μ to shift these points to have the desired location in the x-y plane.)

2. Compute the first principal component of the data. How does it relate to the eigenvectors of C? What fraction of the total variance of the data is accounted for by the first eigenvector? (2 points).

Expectations

3. Compute the $\mathbb{E}[X^4]$, the 4th moment (or expectation of the 4th power) of a standard Gaussian random variable, that is $X \sim \mathcal{N}(0, 1)$ using two different methods. (2 points each).

(a) Compute the expectation numerically by gridding the normal distribution over some range (e.g., $x = -6 : dx : 6$, with $dx = .01$).

(b) Compute the expectation using Monte Carlo integration. Let N be the number of Monte Carlo samples you draw. Report an estimate with 1SD error bars by repeating the experiment m times and computing the mean of these m values as your estimate, and standard deviation as 1SD error bars.

(Do the two estimates agree?)

4. Use the same two methods to compute the variance of $X =$ the number of heads obtained when flipping a fair coin 20 times. Recall that variance is given by the expectation $\mathbb{E}[(X - \mathbb{E}[X])^2]$. Note that (for the "gridding" method) your grid over X should include only the possible real outcomes (i.e., integers between 0 and 20). Do not use `binopdf` or `binornd`. Compute the binomial distribution pmf yourself using the formula $P(X = k|p, n) = \binom{n}{k} p^k (1-p)^{n-k}$, where $p = 0.5$ and $n = 20$. (Feel free to use `nchoosek`, or simply ignore the coefficient in front and normalize the probability vector by dividing by its sum after you've generated it using $p^k(1-p)^{n-k}$ for each value of k).

   For the Monte Carlo estimate, generate the 20 coin flips for each experment yourself using `rand`. Feel to use the fact that we already "know" the expectation $\mathbb{E}[X] = 10$ here when evaluating $f(X) = (X - \mathbb{E}[X])^2$ on each sample. Once again, report 1SD error bars for your estimate.

The Central Limit Theorem (CLT)

5. The mean of a sample of uniform random variables is also a random variable with a pdf of its own. The Central Limit Theorem says that the pdf of the mean converges to the Normal (Gaussian) pdf as the size of the sample increases. Lets see how it works. (1 point each).

   (a) Generate 100,000 samples of two values each from a uniform distribution (use 'rand'). Compute the mean of each sample (pair of values), and plot a histogram of these. What shape is it? If you're unsure, try it again with more samples.

   (b) Now try it 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) Above, the histogram started to look Normal for remarkably small sample sizes. If we look more closely, though, we can detect deviations from normality: the distribution is converging, but it still has a way to go. To see this, we'll redo parts (a) and (b), making quantile-quantile plots, which plot the quantiles of one distribution against the quantiles of another instead of histograms. (These are known commonly as "QQ plots"). We'll use the function `normplot` to make QQ plots showing quantiles of our samples against quantiles of a normal distribution.

   (d) To warm up, try it on a sample of 10,000 values from a normal distribution. The points fall nearly on a straight line telling us that the sample is close to normal, as expected. Try this a few times to see how normplot behaves with a sample from a normal random variable.

Now try it on a sample of 10,000 values from a uniform distribution. Notice it isn't a straight line. Explain qualitatively why it has the shape it does. Hint: Work out the quantiles of the uniform distribution and the normal distribution.

(e) Now redo the computation of (b) for the means over 10,000 samples sized 2, 3 and 4, and make QQ plots of the means for each. Keep increasing the sample size (the 4, not the 10,000) until you cant tell the resulting QQ plot from the normal QQ plots in (c). How big does it have to be?