Tutorial problems for Lectures 1, 2, and 3. Due Wednesday September 19th.

Problem 1: Show that a Poisson distribution converges to a Gaussian in the limit of large λ. Hints - use Stirling's approximation plus use the first two terms in a log expansion.

Answer

Poisson is $\exp(-\lambda)\lambda^n/n!$. Let's let $n = \lambda + \delta$. Furthermore, it's a bit easier to work in log space. So, we have:

$$\log(\text{PDF}) = -\lambda + n * \log(\lambda) - log(n!)$$

Stirling's approximation is $n! \sim \sqrt{2\pi n}(n/e)^n$ which gives

$$-\lambda + n\log(\lambda) - n\log(n) + n - \frac{1}{2}\log(2\pi n)$$

replace n with $\lambda + \delta$ and group the two log terms to get

$$-\lambda - (\lambda + \delta)\log(1 + \delta/\lambda) + \lambda + \delta - \frac{1}{2}\log(2\pi n)$$

As mentioned, we'll need to expand the log to second order, $log(1 + \epsilon) \sim \epsilon - \epsilon^2/2$. Cancelling the +/-$\lambda$ that have appeared, we now have

$$-(\lambda + \delta)(\delta/\lambda - \delta^2/2\lambda^2) + \delta - \frac{1}{2}\log(2\pi n)$$

Multiplying out the terms gives us

$$-\delta - \delta^2/\lambda + \delta^2/2\lambda + \delta^3/2\lambda^2 + \delta - \frac{1}{2}\log(2\pi n)$$

The δ-only terms cancel, and we can drop the δ^3 term since it is subdominant in the limit of large λ. This leaves

$$-\frac{\delta^2}{2\lambda} - \frac{1}{2}\log(2\pi n)$$

As λ gets large, n converges to λ to leading order, and so we can let the square root go to $\frac{1}{2}\log(2\pi\lambda)$. Now, we had taken the log, so to get back to the PDF, we can take the exponential of this:

$$\text{PDF} \sim \exp(-\frac{\delta^2}{2\lambda})/\sqrt{2\pi\lambda}$$

which is just a Gaussian with variance λ and mean λ, since we set $\delta = n - \lambda$.

Problem 2: The gold standard for a believable result is usually 5σ. Let's define the Gaussian approximation as "good enough" if it agrees with the Poisson to within a factor of 2. How large does n need to be for the Gaussian to be good enough at 5σ? How about at 3σ?

Answer See code for how to solve. 3σ seems to be good at a couple of dozen points, but 5σ takes ∼500. The exact values you get depend on how you have defined "good to within a factor of 2", so don't be concerned if your values were a bit different.

Problem 3: Let's say we have n Gaussian-distributed data points with identical standard deviations σ, and identical but unknown mean. What is the error on the maximum-likelihood estimate of the mean? Now let's say we got the errors on half the data wrong by a factor of $\sqrt{(2)}$ (so the variance is off by a factor of 2). What is the true error on the new non-optimal mean, and how does it compare to the maximum-likelihood you could have gotten had you gotten the noises right? How about if you underweight 1% of the data by a factor of ∼100? And if you overweight 1% of the data by a factor of 100? What type of mistake in weighting your data should you be most concerned about?

Answer For the standard error on the mean, we need to find the variance of

$$\frac{\sum(x_i)}{n}$$

Recalling that the variance of ax for constant a is just $a^2\mathrm{Var}(x)$, and that the variance of the sum is the sum of the variances if the data points are uncorrelated, we know this variance must be

$$\frac{\sum \sigma^2}{n^2} = \frac{n\sigma^2}{n^2} = \frac{\sigma^2}{n}$$

where σ^2 is the variance of the individual data points. So, the standard error is just the square root of this, or $\sigma/\sqrt{n}$.

As shown in class, the variance of $(2x_1 + x_2)/3$ is what we get if we misweight half our data by a factor of 2, assuming the data points all have the same true variance. We can see that the variance of this is just

$$4\sigma^2/9 + \sigma^2/9 = 5\sigma^2/9$$

where σ^2 is the variance of our individual data points. If we had done things properly, the minimum variance would have ben $(x_1 + x_2)/2$ which have variance

$$\sigma^2/4 + \sigma^2/4 = \sigma^2/2$$

The ratio is just (5/9)/(1/2)=(10/9), so our estimator has had its variance increased by 1/9 relative to the best we could have done, or about a 5.4% increase in the error bars.

If we underweight 1% of our data by a factor of 100, we're very close to just throwing it away. If we had 100 data points, the variance of the mean is then going to be something like $n/100$. If we throw away a point, we have 99 left, and the variance becomes $n/99$, for a ratio of 100/99. Taking the square root, we find we have increased the error bar by 0.5%. If we overweight 1 by a factor of 100 and have another 100 points that are correctly weighted, then we have

$$\frac{100x_1 + \sum_2^{101} x_i}{200}$$

which has variance $\frac{10^4\sigma^2+100\sigma^2}{4*10^4}$, or about $\sigma^2/4$. So, by getting the error bar too small by a factor of 10 on one point out of 100, we got the same error bar as we would have with 4 properly weighted

points. So, we have effectively thrown away 96% of our data. This should make you very nervous about letting bad data in/artificially small errors. While it's always sad to throw away good data, losing a bit of good data is far safer than letting in bad data/overweighting data you have by a lot.

Problem 4: In linear least-squares, our estimate for fit parameters m is unbiased if $< m >= m_{true}$. If our model is correct, $< d >= \mathrm{A}m$, then show that the least-squares solution is unbiased. Show that this result does not depend on our noise matrix N actually being the noise in the data.

Answer If m_{true} is the true model and $\hat{m}$ is our estimated model parameters, we know that

$$< d >= \mathrm{A}m_{true}$$

and

$$\mathrm{A}^T\mathrm{N}^{-1}\mathrm{A}\hat{m} = \mathrm{A}^T\mathrm{N}^{-1}d$$

We can take the expectation, to get

$$\left\langle \mathrm{A}^T\mathrm{N}^{-1}\mathrm{A}\hat{m}\right\rangle = \left\langle \mathrm{A}^T\mathrm{N}^{-1}d\right\rangle$$

but we know what the expectation of the data is, so:

$$\left\langle \mathrm{A}^T\mathrm{N}^{-1}\mathrm{A}\hat{m}\right\rangle = \left\langle \mathrm{A}^T\mathrm{N}^{-1}\mathrm{A}m_{true}\right\rangle$$

if $\mathrm{A}^T\mathrm{N}^{-1}\mathrm{A}$ is invertible, we can multiply on the left by its inverse, and noting that the expectation of m_{true} is just m_{true}, to get

$$\langle \hat{m}\rangle = m_{true}$$

We have not relied on anything about the noise matrix, other than the fact that $\left\langle \mathrm{A}^T\mathrm{N}^{-1}d\right\rangle = \mathrm{A}^T\mathrm{N}^{-1}\mathrm{A}m_{true}$. In slightly more detail, if $d = \mathrm{A}m_{true} + n$ with n the vector of actual noises in our measurement, we have relied on the fact that

$$\left\langle \mathrm{A}^T\mathrm{N}^{-1}n\right\rangle = 0$$

. This is generically true as long as the noise matrix doesn't have any correlation with the specific realization of n. That's obviously true if the noise matrix is correct, or if we came up with it without looking at the data. It can, however, break down if the noise matrix we use is derived from the data, which we will see in the next problem.

Problem 5: The preceding statement comes with an important caveat, namely that our noise estimate is not correlated with any residual signal in the data. Write a computer program that generates random Gaussian noise (numpy.random.randn may come in handy here), and adds a template (possibly a Gaussian as would be typical for a source seen by a telescope with a finite-resolution beam, but the details aren't important) to it. Estimate the noise by assuming it's constant and equal to the scatter in the observed data, which has the template added to it. Show that the least-squares estimate for any individual chunk is unbiased, but that the least-squares estimate for many data chunks analyzed jointly is biased low. Basically, your program should fit

an amplitude and error to each individual chunk, then use that to get an overall amplitude/error. How might you go about mitigating this bias? Note that this is an extremely common situation when you say observed the same field/source several times and want to make your "best" estimate of what you have seen.

Answer See code example for how to calculate this. Basic problem is that if noise anti-correlates with the signal then we get a low answer, but also a low estimate of the noise. So, when we do the weighted average of a bunch of measurements, the total average is biased low. There are a bunch of ways of addressing this, depending on what your data look like. One would be to subtract off the best-fit signal, then re-estimate the noise. Repeat this a few times until the mean quits changing. If you think the noises ought to be similar, you could treat all your measurements the same, which we have seen gives an unbiased result. You might also try using the median instead of the standard deviation for the noise estimate. That will be quite a bit more robust to these problems as well.

Problem 6 - bonus: In class it was asserted that adding orthogonal matrices into the expression for χ^2 let us work with correlated data. In particular, show that

$$\chi^2 = \delta^T V^T V N^{-1} V^T V \delta$$

, where $\delta_i = d_i - < d_i >$, is equivalent to

$$\chi^2 = \tilde{\delta}^T \tilde{N}^{-1} \tilde{\delta}$$

and that

$$\tilde{N}_{ij} = < \tilde{\delta}_i \tilde{\delta}_j >$$

Answer The first part is straightforward. Obeying the rules of matrix transposes, $\delta^T V^T = (V\delta)^T$, so if we define $\tilde{\delta} \equiv V\delta$ and $\tilde{N} = VNV^T$, we have

$$\chi^2 = \tilde{\delta}^T \tilde{N}^{-1} \tilde{\delta}$$

You can see my (ugly!) answer to the second part below, but one of you had a more elegant answer than I did, which I'll reproduce here first.

Now that we have an expression for the rotated data, what is its covariance? Being careful with transposes, the covariance of the rotated data is

$$\left\langle \tilde{\delta} \tilde{\delta}^T \right\rangle$$

But, we know that $\tilde{\delta} = V\delta$ so we have

$$cov = \left\langle V\delta (V\delta)^T \right\rangle = \left\langle V\delta\delta^T V^T \right\rangle$$

But, the central part $\delta\delta^T$ is just the covariance of the uncorrelated data, which is just a diagonal matrix with σ_i^2 along the diagonal, which is our original noise matrix. So, we have

$$\left\langle \tilde{\delta}\tilde{\delta}^T \right\rangle = VNV^T = \tilde{N}$$

And now for my uglier, index-based proof...

We now want to work out the expectation of $\tilde{\delta_i}\tilde{\delta_j}$. We know that

$$\tilde{\delta_i} = \sum V_{ik}\delta_k$$

and so that

$$\left\langle \tilde{\delta_i}\tilde{\delta_j} \right\rangle = \left\langle \sum V_{ik}\delta_k \sum V_{jk'}\delta'_k \right\rangle$$

Now, since the original δ are uncorrelated, the expectation of all the terms on the right is zero unless $k = k'$. So we can replace the double sum by a single sum:

$$\left\langle \tilde{\delta_i}\tilde{\delta_j} \right\rangle = \left\langle \sum V_{ik}\delta_k V_{jk}\delta_k \right\rangle$$

But the expectation of δ_k^2 is just the variance of δ_k, N_{kk}. So, we have

$$\left\langle \tilde{\delta_i}\tilde{\delta_j} \right\rangle = \sum V_{ik}V_{jk}N_{kk}$$

where we dropped the expectation on the right because we have now averaged over data realizations.

We now need to compare this to $\tilde{N} = VNV^T$. Recalling that for matrix multiplication $C = AB$ that

$$C_{ij} = \sum_k A_{ik}B_{kj}$$

we can first see that NV^T has elements (where we have swapped the indices on V to reflect the fact that it's a transpose)

$$NV^T_{ij} = \sum N_{ik}V_{jk}$$

but N_{ik} is zero unless $i = k$. So,

$$NV^T_{ij} = N_{ii}V_{ji}$$

We now want to work out VNV^T_{ij}, which will be

$$\sum V_{ik}(NV^T)_{kj}$$

After swapping the naming of indices on NV^T this gives us

$$(VNV^T)_{ij} = \sum_k V_{ik}(NV)_{jk} = \sum_k V_{ik}V_{jk}N_{kk}$$

which is exactly what we have for the expectation of $\tilde{\delta_i}\tilde{\delta_k}$. So, if we can directly calculate the rototated covariance $\left\langle \tilde{\delta_i}\tilde{\delta_j} \right\rangle$ then we can work with correlated noise directly without ever being forced to work in a space in which they are uncorrelated.