Problem 1

$$P(k) = \frac{\lambda^k e^{-\lambda}}{k!} \quad , \quad \lambda : \text{mean or expected rate}$$

Stirling's approximation: $k! \approx \sqrt{2\pi k}\left(\frac{k}{e}\right)^k$ as $k \to \infty$.

we substitute and get $P(k) = \frac{\lambda^k e^{-\lambda}}{\sqrt{2\pi k}}\left(\frac{e}{k}\right)^k = \frac{1}{\sqrt{2\pi k}} e^{k-\lambda}\left(\frac{\lambda}{k}\right)^k$

we look at the standard poisson variable $z = \frac{k-\lambda}{\sqrt{\lambda}}$, since mean = variance $= \lambda$

rearranging, $z\sqrt{\lambda} = k - \lambda$

$$\frac{z}{\sqrt{\lambda}} = \frac{k}{\lambda} - 1 \quad \Rightarrow \quad \frac{\lambda}{k} = \left(\frac{z}{\sqrt{\lambda}} + 1\right)^{-1}$$

so $P(z) = \left(\frac{1}{\sqrt{2\pi(z\sqrt{\lambda}+\lambda)}}\right) e^{z\sqrt{\lambda}}\left(\frac{z}{\sqrt{\lambda}} + 1\right)^{-z\sqrt{\lambda}-\lambda}$

We look at $P' = e^{z\sqrt{\lambda}}\left(\frac{z}{\sqrt{\lambda}}+1\right)^{-z\sqrt{\lambda}-\lambda}$ and take $\ln$ of both sides.

$$\ln(P') = \ln\left(e^{z\sqrt{\lambda}}\right) + \ln\left(\frac{z}{\sqrt{\lambda}}+1\right)^{-z\sqrt{\lambda}-\lambda}$$

$$= z\sqrt{\lambda} - (z\sqrt{\lambda}+\lambda)\ln\left(\frac{z}{\sqrt{\lambda}}+1\right)$$

Now, $\ln(x+1) \sim x - \frac{x^2}{2} + \frac{x^3}{3} - \frac{x^4}{4} + \dots$, so we can approximate using the first 2 terms of this expansion.

$$\ln(P') \approx z\sqrt{\lambda} - (z\sqrt{\lambda}+\lambda)\left[\frac{z}{\sqrt{\lambda}} - \frac{z^2}{2\lambda}\right] = \cancel{z\sqrt{\lambda}} - \left[z^2 - \frac{z^3}{2\sqrt{\lambda}} + \cancel{z\sqrt{\lambda}} - \frac{z^2}{2}\right]$$

$$\ln(P') = -z^2 + \frac{z^3}{2\sqrt{\lambda}} + \frac{z^2}{2} = -\frac{z^2}{2} + O(\lambda^{-1/2})$$

Then taking the limit as $\lambda \to \infty$,

$$\lim_{\lambda\to\infty} \ln(P') = \lim_{\lambda\to\infty}\left[-\frac{z^2}{2} + O(\lambda^{-1/2})\right] = -\frac{z^2}{2}$$

$$\Rightarrow \lim_{\lambda\to\infty} P' = e^{-z^2/2} = e^{-\frac{1}{2}\left(\frac{k-\lambda}{\sqrt{\lambda}}\right)^2}$$

$\therefore$ In the limit of large λ, we have $P(z) = \left(\frac{1}{\sqrt{2\pi(z\sqrt{\lambda}+\lambda)}}\right) \cdot e^{-z^2/2}$

or substituting back with $z = \frac{k-\lambda}{\sqrt{\lambda}}$, $\boxed{P(k) = \frac{1}{\sqrt{2\pi k}}\, e^{-\frac{1}{2}\left(\frac{k-\lambda}{\sqrt{\lambda}}\right)^2}}$

which is a Gaussian with mean λ and std $\sqrt{\lambda}$.

Problem 2

$$P(n) = \frac{\lambda^n e^{-\lambda}}{n!} \quad , \quad G(n) = \frac{1}{\sqrt{2\pi\lambda}}\, e^{-\frac{1}{2}\left(\frac{n-\lambda}{\sqrt{\lambda}}\right)^2}$$

are the equations giving the poisson distribution and its gaussian approximation, as derived in Problem 1.

In order to avoid computational barriers of taking factorials of large values of n, I will compute $\log(P)$ on code and raise e to the output to form my distribution.

The functions and process are shown in the jupyter notebook in this repo.

I cycled through a range of λ values and for each picked an $n = \lambda + 3\sqrt{\lambda}$ (or $5\sqrt{\lambda}$ for $5\sqrt{\lambda}$ calculation). Then, I found $P(n)$ and $G(n)$ at such a value of n. If the ratio between both was less than a factor of 2, I kept the n. Otherwise, I increased λ (thus increased n) and tried again.

For the 5σ case, I had to go to $\boxed{n \cong 900}$ to get a good enough approximation.

For the 3σ case, I only had to go to $\boxed{n \cong 41}$ for the same requirement.

Problem 3

Say our sample is made up of n Gaussian-distributed points with mean μ and variance σ^2.

Then they each follow with probability given by $G(x_i) = \frac{1}{\sqrt{2\pi\sigma_0^2}} \cdot \exp\left(-\frac{1}{2}\frac{(x_i-\mu_0)^2}{\sigma_0^2}\right)$

Then the likelihood function is

$$L(\mu, \sigma^2, x_1, \dots, x_n) = \prod_{i=1}^{n} G(x_i) = (2\pi\sigma^2)^{-n/2} \exp\left(-\frac{1}{2\sigma^2} \sum_{i=1}^{n} (x_i-\mu)^2\right)$$

And the log-likelihood function is

$$\ell(\mu, \sigma^2, x_1, \dots, x_n) = -\frac{n}{2} \ln(2\pi\sigma^2) - \frac{1}{2\sigma^2} \sum_{i=1}^{n} (x_i-\mu)^2$$

Now to find the maximum likelihood estimator for μ, we must find the values of μ that maximize $\ell(\mu, \sigma^2, x_1, \dots, x_n)$, i.e. $\frac{\partial \ell}{\partial \mu} = 0$

$$\frac{\partial \ell}{\partial \mu_0} = \frac{-2(-1)}{2\sigma^2} \sum_{i=1}^{n} (x_i-\mu) = \frac{1}{\sigma} \sum_{i=1}^{n} (x_i-\mu) = \frac{1}{\sigma^2}\left(-n\mu + \sum_{i=1}^{n} x_i\right)$$

$$\text{So } \frac{\partial \ell}{\partial \mu} = 0 \Rightarrow n\mu = \sum_{i=1}^{n} x_i \Rightarrow \boxed{\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} x_i}$$

Then the error on that estimate is

$$\begin{aligned} Var(\hat{\mu}) &= Var\left(\frac{1}{n}\sum_{i=1}^{n} x_i\right) \\ &= \frac{1}{n^2} Var\left(\sum_{i=1}^{n} x_i\right) \\ &= \frac{1}{n^2}\left[Var(x_1) + Var(x_2) + \dots + Var(x_n) + \text{covariance terms}\right] \end{aligned}$$

Since x_n variables are taken randomly from sample, they should be independent, so covariance terms go to 0.

$$\Rightarrow Var(\hat{\mu}) = \frac{1}{n^2}\sum_{i=1}^{n} Var(x_i) = \frac{1}{n^2}\sum_{i=1}^{n} \sigma_n^2 = \frac{1}{n^2}(n\sigma_n^2)$$

$$\Rightarrow Var(\hat{\mu}) = \frac{\sigma_n^2}{n} \Rightarrow \boxed{\text{error on } \hat{\mu} = \frac{\sigma_n}{\sqrt{n}}}$$

Now, say we mistook the errors on half of our data points by a factor of $\sqrt{2}$. So for $\frac{n}{2}$ of our points, we have $var(x_i) = \frac{\sigma^2}{2}$ instead of $var(x_i) = \sigma^2$.

Then we have $Var(\hat{\mu}_2) = \frac{1}{n^2}\left[\frac{n}{2}\left(\frac{\sigma^2}{2}\right) + \frac{n}{2}(\sigma_0^2)\right] = \frac{3}{4}\frac{\sigma^2}{n}$

Therefor the true error on new non-optimal mean is given by $\boxed{err(\hat{\mu}) = \sqrt{\frac{4}{3}} \cdot \frac{\sigma}{\sqrt{n}}}$

This new error is larger than what we would have had if we had estimated the errors properly.

If we underweight/overweight 1% of points,

$$Var(\hat{\mu}_{-100}) = \frac{1}{n^2}\left[\frac{n}{100}\left(\frac{\sigma_0^2}{100}\right) + \frac{99n}{100}(\sigma_0^2)\right]$$

$$= \frac{1}{n}\left[\frac{9901\,\sigma_0^2}{10000}\right] \Rightarrow Var(\hat{\mu}_{-100}) = 0.9901\left(\frac{\sigma_0^2}{n}\right)$$

$$\Rightarrow \text{error on } \hat{\mu} = \sqrt{\frac{1}{0.9901n}}\;\sigma_0 = \boxed{1.005\frac{\sigma_0}{\sqrt{n}}}$$

$$Var(\hat{\mu}_{+100}) = \frac{1}{n^2}\left[\frac{n}{100}(100\sigma_0^2) + \frac{99n}{100}(\sigma_0^2)\right]$$

$$= \frac{1}{n}\left[\frac{199\sigma_0^2}{100}\right] \Rightarrow Var(\hat{\mu}_{+100}) = 1.99\left(\frac{\sigma_0^2}{n}\right)$$

$$\Rightarrow \text{error on } \hat{\mu} = \sqrt{\frac{1}{1.99n}}\;\sigma_0 = \boxed{0.71\frac{\sigma_0}{\sqrt{n}}}$$

We see that when our actual variance on part of the data points is smaller than we predicted, we get an error on the maximum-likelihood estimate of the mean that is slightly larger than if we got it correct.

However, if we instead predicted a larger error, the resulting error on $\hat{\mu}$ will be a little smaller.

Therefor, we should be most concerned about predicting errors that are too large, as our estimates on the mean will have a higher error if we end up having smaller errors on part of the data.

Problem 4

For this problem, we want to show that the maximum likelihood estimator is unbiased, so $\langle m \rangle = m_{true}$.

To do that, I wrote a program seen in the Jupyter notebook in this github repo.

I created a noisy signal, with the signal coming from a Gaussian with $\mu = 0$, $\sigma = 1$ and amplitude $A_0 = 1$. Then I added random gaussian noise on top of it.

The objective was to use least squares fitting to estimate the amplitude of the signal. Then, comparing that to the true amplitude $A_0 = 1$, we see if the method is biased or not.

To construct my A matrix, I linearized $ae^{-x^2/2} \approx a - \frac{3ax^2}{2} + \frac{ax^4}{2} + \dots$ around $x=0$. Then I used a polynomial of order 4, where the first parameter (first coefficient), would be my amplitude estimation.

I repeated the process of finding $m = (A^T N^{-1} A)^{-1} A^T N^{-1} d$ for 10,000 chunks of data. For each of them, $N = \sigma^2 \mathbb{1}$, where σ^2 was estimated to be the spread of the data for that given realization of random noise + signal.

To calculate the error of this fit, I used $\mathrm{var} = (A^T N^{-1} A)^{-1}$ and took the first element on the diagonal to represent the variance on the amplitude estimate.

Now, using all of this, the variance weighted average for the fitted amplitude was given by

$\boxed{\mathrm{Amp} = 0.966 \pm 0.014}$

If we find the difference to A_0, we see this is biased low and the significance of the bias is given by 0.03, considering the error in the data.

In order to mitigate the bias, we should make sure the noise estimation is done after model subtraction, so it is less affected by the signal. In general, the model will depend on the noise estimates, so this must be done iteratively. It might not be realistic to repeat as many times as needed to remove all bias, but some mitigation could be helpful.

Problem 5

rotates space so χ^2 has a non-diagonal noise matrix.

show in new rotated space $\tilde{N}_{ij} = \langle \tilde{n}_i \tilde{n}_j \rangle$

As seen in class, we use the orthogonal matrix V to rotate the space, so $VV^T = V^TV = \mathbb{1}$.

Our original χ^2 expression was $\chi^2 = (d - A(m))^T N^{-1} (d - A(m))$

now we manipulate this by including VV^T $(= \mathbb{1})$ terms,

$$\chi^2 = (d - A(m))^T V^T V \; N^{-1} V^T V \; (d - A(m)) = r^T V^T V \; N^{-1} V^T V \; r$$

for $r = d - A(m)$.

$$\Rightarrow \chi^2 = (Vr)^T (VNV^T)^{-1} (Vr)$$

Then, we define new variables, $\tilde{N} = VNV^T$, $\tilde{r} = Vr$

so $\chi^2 = \tilde{r}^T \tilde{N}^{-1} \tilde{r}$ in the new rotated space

Now if we think about $\tilde{N}$, we have by definition $\tilde{N} = VNV^T$, for $\tilde{N}$ no longer diagonal.

So thinking about the elements of this matrix, we have

$$\tilde{N}_{ij} = \begin{bmatrix} V_{11} & V_{12} & \cdots & V_{1j} \\ V_{21} & \ddots & & \\ V_{31} & & \ddots & \\ \vdots & & & V_{ij} \end{bmatrix} \begin{bmatrix} n_{11} & 0 & 0 & \cdots \\ 0 & n_{22} & & \\ \vdots & & \ddots & \\ & & & n_{ij} \end{bmatrix} \begin{bmatrix} V_{11} & V_{21} & V_{31} & \cdots \\ V_{12} & \ddots & & \\ \vdots & & \ddots & \\ V_{1n} & & & V_{nn} \end{bmatrix}$$

$$= \begin{bmatrix} V_{11}n_{11} & V_{12}n_{22} & V_{13}n_{33} \cdots & V_{1j}n_{jj} \\ V_{21}n_{11} & V_{22}n_{22} & \vdots \; \ddots & \\ V_{31}n_{11} & V_{32}n_{22} & \vdots & \\ \vdots & \vdots & & V_{ij}n_{jj} \end{bmatrix} \begin{bmatrix} V_{11} & V_{21} & \cdots \\ V_{12} & \ddots & \\ \vdots & & \\ V_{1n} & & V_{nn} \end{bmatrix}$$

$$= \begin{bmatrix} V_{11}n_{11}V_{11} + V_{12}n_{22}V_{12} + V_{13}n_{33}V_{13} + \cdots & V_{11}n_{11}V_{21} + V_{12}n_{22}V_{22} + \cdots & \cdots \\ V_{21}n_{11}V_{11} + V_{2} & \ddots & \vdots \\ \cdots & \cdots & \ddots \end{bmatrix}$$

Then $\boxed{\tilde{N}_{ij} = V_i N V_j^T}$ by inspection of the terms.

Also, we know $\tilde{r}_i = V_i^T r$, so $\langle \tilde{r}_i \tilde{r}_j \rangle = \langle V_i^T r r^T V_j \rangle$

but $\langle r r^T \rangle = N$, our original diagonal noise matrix..

$\Rightarrow \boxed{\langle \tilde{r}_i \tilde{r}_j \rangle = V_i N V_j^T}$

Finally, we compare both results to conclude that $\boxed{\tilde{N}_{ij} = \langle \tilde{r}_i \tilde{r}_j \rangle}$.