MATH 3375 - Assignment 5

Population Coding

Consider a population of N neurons, responsible for estimating a continuous stimulus parameter $\theta \in [0, 2\pi)$. Let the mean firing rate of neuron i be:

$$f_i(\theta) = r_{max} e^{\frac{2N}{\pi}(cos(\theta - \phi_i) - 1)}.$$

Further, let the population be evenly tiled in stimulus space with $\phi_i = (i-1)\Delta\phi + \frac{\Delta\phi}{2}$, where $\Delta\phi = \frac{2\pi}{N}$. Let the population response $\mathbf{r} = [r_1, \ldots, r_N]$ to a single trial presentation of stimulus θ obey the following density function. (The MATLAB function *random* with a 'poiss' specifier will be useful. In XPP, it is *poisson*.)

$$P(\mathbf{r}|\theta) = \prod_{i=1}^{N} \frac{(f_i(\theta)T)^{r_i T}}{(r_i T)!} e^{-f_i(\theta)T},$$

where T is the duration of the trial. Unless otherwise mentioned set $r_{max} = 50$ Hz and $T = 0.25s$.

1. Discretize stimulus θ into $\{0, \Delta\theta, 2\Delta\theta, \ldots, M\Delta\theta\}$ with $M = 100$ and $\Delta\theta = 2\Pi/100$. For a fixed θ and a single stimulus trial let the stimulus estimate be

$$\begin{aligned} C &= \sum_{i=1}^{N} r_i \cos\phi_i \\ S &= \sum_{i=1}^{N} r_i \sin\phi_i \\ \theta_{est} &= \text{atan2}(S, C) \end{aligned}$$

where atan2(y, x) is the value of θ such $x = \cos\theta$ and $y = \sin\theta$ (it is available in XPP and MatLab)

For each fixed θ perform 10^4 trials and numerically compute $\langle\theta_{est}\rangle$ and $\sigma_{est}^2 = \langle(\theta - \theta_{est})^2\rangle$, where $\langle\cdot\rangle$ is an expectation over trials. Plot $\langle\theta_{est}\rangle$ and σ_{est}^2 as a function of θ for populations with $N = 4$ and $N = 100$ (Beware that your θ_{est} for each trial might be different form θ by $\pm 2\pi$, so that when you compute θ_{est} ar each trial, you might have to add $\pm 2\pi$ to bring it as close as possible to θ.)

2. For the remainder of the exercises, just do the cricket, $N = 4$ neurons. For the results of question (1) compute the bias of the estimate, $b_{est}(\theta) = \langle\theta_{est}\rangle - \theta$, and fit a simple periodic function (i.e cos or sin) to give a smooth approximation to $b_{est}(\theta)$. Plot $b_{est}(\theta)$ and your approximation. (You can eyeball this - it will be proportaional to $K\cos 8\pi\theta$ I think.)

3. Compute the Cramer-Rao lower bound for $\sigma^2_{est}(\theta)$:
$$CR(\theta) = \frac{(1 + b'_{est}(\theta))^2}{I_F(\theta)},$$
where I_F is the Fisher information and $'$ denotes differentiation with respect to θ. Use the formulas 3.41 and 3.45 from your book using the approximate equation for $b_{est}(\theta)$.

4. Repeat question (3) with $T = \{0.1, 0.5, 1.0, 2.0\}$. Comment on the relation (if any) of the minima and maxima of σ^2_{est} and CR.In particular, does $b_{est}(\theta)$ depend on T? If not, then it should be clear as how CR depends on T.

5. **Noise driven neural models.** Compute the FI curve for a noisy quadrattic integrate-and-fire mode and for a noisy integrate-and-fire model as follows. Let $f(V, I)$ be the right-hand sides of the equations, $f(V, I) = -g(V - V_L) + I$ for the LIF and $f(V, I) = g(V - V_L)(V - V_T)/(V_T - V_L) + I$ for the LIF. In both cases, $g = .05$ and $V_L = -65$. You can use $C = 1$ for the capacitance. For the QIF, let $V_T = -50$. For both neurons, let $V_{reset} = -70$ and for the LIF, $V_{spike} = -50$; for the QIF, $V_{spike} = 20$. Solve the equation:
$$V(t + h) = V(t) + hf(V(t), I) + \sqrt{h}\sigma N(0, 1)$$
which is just Eulers method for white noise. Here N is a normally distributed random variable. Now, pick $h = .05$ for example and solve this equation as you vary I for $T = 20000$ (or more, note this would be 400000 iterates) and count the number of spikes. The frequency is then the count divided by T and then multiplied by 1000 to get it to Hz. Do this for the LIF ranging $-1 \le I < 2$ in 150 steps for $\sigma = 2, 1, 0.5, 0$. For the $\sigma = 2$ case, see if you can fit the FI curve to the function:
$$g_{LIF}(I) = A(I - I_0)/(1 - \exp(-B(I - I_0)))$$
Thus, you want to choose A, I_0, B so that your simulated FI curve is reasonably well fit by the above function. Note you do not have to use a sophisticated fitting algorithm. I did it in about 2 minutes using gnuplot.

Repeat this with the QIF but try $\sigma = 4, 2, 1, 0$ and range I between -1 and 3 in 200 steps. The QIF is better fit with a square-root nonlinearity,

$$g_{QIF}(I) = \sqrt{g_{LIF}(I)}$$

for large I, but the exponential linear, g_{LIF} is a good fit when I is small.