Optimal Design for estimation in stochastic LIF models - take 2

Alexandre Iolov, Susanne Ditlevsen, André Longtin
<aiolo040 at uottawa dot ca>, alongtin at uottawa dot ca

April 25, 2014

Abstract

Given a leaky, noisy integrate-and-fire neuronal model - we discuss optimal design-type questions on what is the best external perturbation in order to facilitate parameter estimation using discrete-time, trajectory observations

Contents

1 Problem Formulation

The basic goal of 'Optimal Design' is to perturb a dynamical system in an 'optimal' way such as to 'best' estimate its structural parameters.

Consider the system

$$dX = \underbrace{(\alpha + \beta(\mu - X)}_{U(x,\alpha)} \, \mathrm{dt} + \underbrace{\sigma}_{\sqrt{2D}} dW \tag{1}$$

That can be solved as:

$$\begin{aligned}
dX =& (\alpha + \beta(\mu - X)dt + \sigma dW \\
(e^{\beta t}dX + e^{\beta t}\beta X_t)dt =& e^{\beta t}(\alpha + \beta\mu)dt + \sigma e^{\beta t} dW \\
Xe^{\beta t} - X_0 =& \int e^{\beta t}(\alpha + \beta\mu)dt + \int \sigma e^{\beta t} dW \\
\implies X_t =& e^{-\beta t}X_0 + \frac{(\alpha + \beta\mu)}{\beta} \cdot (1 - e^{-\beta t}) + \sigma \cdot \sqrt{\frac{1 - e^{-2\beta t}}{2\beta}} \cdot \xi
\end{aligned}$$

where ξ is a standard normal RV.

i.e. if X_0 is constant

$$X_t \sim N\left(e^{-\beta t}x_0 + \frac{(\alpha + \beta\mu)}{\beta} \cdot (1 - e^{-\beta t}), \quad \sigma \cdot \sqrt{\frac{1 - e^{-2\beta t}}{2\beta}}\right)$$

Consider discrete observations X_n, t_n obtained at uniform t_n. Then the transition probabilities $p_n(X_n|X_{n-1})$ are given by:

$$\begin{aligned}
p_n(X_n|X_{n-1}; \mu, \beta, \sigma; \Delta_n) \propto& \frac{\beta}{\sigma\sqrt{1 - e^{-2\beta\Delta_n}}} \\
& \cdot \exp\left(\frac{\left(X_n - (\frac{\alpha}{\beta} + \mu) - (X_{n-1} - \frac{\alpha}{\beta} - \mu) \cdot e^{-\beta\Delta_n}\right)^2 \cdot \beta}{\sigma^2(1 - e^{-2\beta\Delta_n})}\right)
\end{aligned}$$

The likelihood is simply the product:

$$L(\{X_n\}|; \mu, \beta, \sigma; \Delta_n; \alpha) = \prod_n p_n(X_n|X_{n-1}; \mu, \beta, \sigma; \Delta_n) \tag{2}$$

And the log-likelihood of X_n, t_n is

$$\begin{aligned}
l(\beta, \mu, \sigma|) =& \sum \log p_n(X_n|X_{n-1}) \\
=& \frac{N}{2} \log \frac{\beta}{\sigma^2(1 - e^{-2\beta\Delta_n})} \\
& - \sum_n \left(X_n - (\frac{\alpha}{\beta} + \mu) - (X_{n-1} - \frac{\alpha}{\beta} - \mu) \cdot e^{-\beta\Delta_n}\right)^2 \cdot \frac{\beta}{\sigma^2(1 - e^{-2\beta\Delta_n})}
\end{aligned}$$

ML estimators for μ, β, σ are obtained via setting $\partial_p l$ to zero for each parameter p. (ignore for now that Δ_n is not the same throughout). However, it turns out that it is easier to first compute the ML estimate for σ

and then plug it back into the likelihood, l, to simplify things:

$$
\begin{aligned}
\partial_\sigma l() =& -\frac{N}{\sigma} + 2\sum_n \frac{\left(X_n - e^{-\beta\Delta_n}X_{n-1} - (\frac{\alpha}{\beta}+\mu)\cdot(1-e^{-\beta\Delta_n})\right)^2\beta}{\sigma^3\cdot(1-e^{-2\beta\Delta_n})} \\
\implies \hat{\sigma}^2 =& 2\sum_n \frac{\left(X_n - e^{-\hat{\beta}\Delta_n}X_{n-1} - (\frac{\alpha}{\hat{\beta}}+\mu)\cdot(1-e^{-\hat{\beta}\Delta_n})\right)^2\hat{\beta}}{N\cdot(1-e^{-2\hat{\beta}\Delta_n})} \qquad (3)
\end{aligned}
$$

With that the likelihood becomes:

$$
l(\beta,\mu|\, X_n) = -\,N\log\left(\frac{\sum_n\left(X_n - e^{-\beta\Delta_n}X_{n-1} - (\frac{\alpha}{\beta}+\mu)\cdot(1-e^{-\beta\Delta_n})\right)^2}{N}\right) - 1
$$

Now maximizing the negative of a log plus a const is the same as minimizing the argument of the log. So we need to *minimize*

$$
l(\beta,\mu|\, X_n) \equiv \sum_n\left(X_n - e^{-\beta\Delta_n}X_{n-1} - (\frac{\alpha}{\beta}+\mu)\cdot(1-e^{-\beta\Delta_n})\right)^2
$$

Let's differentiate

$$
\partial_\mu l() = 2\sum_n\left(X_n - e^{-\beta\Delta_n}X_{n-1} - (\frac{\alpha}{\beta}+\mu)\cdot(1-e^{-\beta\Delta_n})\right)\cdot(1-e^{-\beta\Delta_n}) \qquad (4a)
$$

$$
\begin{aligned}
\partial_\beta l() =& 2\sum_n\left(X_n - e^{-\beta\Delta_n}X_{n-1} - (\frac{\alpha}{\beta}+\mu)\cdot(1-e^{-\beta\Delta_n})\right) \qquad (4b) \\
&\times\left(\Delta_n e^{-\beta\Delta_n}X_{n-1} + \frac{\alpha}{\beta^2}(1-e^{-\beta\Delta_n}) - (\frac{\alpha}{\beta}+\mu)(\Delta_n e^{-\beta\Delta_n})\right)
\end{aligned}
$$

The μ equation is straight-forward to solve

$$
\hat{\mu} = \frac{\sum_n\left(X_n - e^{-\beta\Delta_n}X_{n-1} - \frac{\alpha}{\beta}(1-e^{-\beta\Delta_n})\right)}{N(1-e^{-\beta\Delta_n})} \qquad (5)
$$

This means that we only need to solve one equation in one unknown

$$
\begin{aligned}
0 =& \sum_n\left(X_n - e^{-\beta\Delta_n}X_{n-1} - (\frac{\alpha}{\beta}+\mu)\cdot(1-e^{-\beta\Delta_n})\right) \qquad (6) \\
&\times\left(\Delta_n e^{-\beta\Delta_n}X_{n-1} + \frac{\alpha}{\beta^2}(1-e^{-\beta\Delta_n}) - (\frac{\alpha}{\beta}+\mu)(\Delta_n e^{-\beta\Delta_n})\right)
\end{aligned}
$$

for β and then plug into eqs. (3) and (5)

(I have checked that in the case of $\alpha = 0$ this reduces to well-known expressions!)

Note: we've been quite cavalier about the constancy of Δ_n, mostly treating it as a constant, in order to focus on the impact of α. Later we can go back and be a little more rigorous, treating Δ_n as a function of n.

On the contrary, while not being explicit in the notation, we have never assumed α is constant and everything above can be rewritten in terms of α_n.

2 Optimal Design

The ultimate question is how to choose the controls $\alpha(t) = \alpha(t_{n-1})$, assumed piecewise constant over Δ_n, such as to facilitate the estimation of the parameters μ, β, σ?

Two references suggest that one should use the mutual information [3, 1] as the criterion:

Let us follow [3]: Call the prior of the parameters, $\theta = \{\mu, \beta, \sigma\}$:

$$\rho(\theta)$$

and the posterior

$$p(\theta|X;\alpha) = \frac{L(x|\theta;\alpha) \cdot \rho(\theta)}{\int_\Theta L(x|\theta;\alpha) \cdot \rho(\theta)\,\mathrm{d}\theta} \tag{7}$$

Where, L is the likelihood of X - $L(X|\theta;\alpha)$, that is given in eq. (2). X could represent only one observation, X_n or a set of observations $\{X_k\}_n^{n+K}$.

Then one wants to optimize the mutual information:

$$I(\alpha) = \int_\Theta \int_X \log\left[\frac{p(\theta|x;\alpha)}{\rho(\theta)}\right] \cdot L(x|\theta;\alpha) \cdot \rho(\theta)\,\mathrm{d}\theta\,\mathrm{d}x \tag{8}$$

This is straight from equations 6,8,9 of [3] (That paper is on Mendeley)

Replacing the posterior in eq. (8), with the bayesian formula from eq. (7), we get:

$$I(\alpha) = \int_\Theta \int_X \log\left[\frac{L(x|\theta;\alpha)}{\int_\Theta L(x|\theta;\alpha) \cdot \rho(\theta)\,\mathrm{d}\theta}\right] \cdot L(x|\theta;\alpha) \cdot \rho(\theta)\,\mathrm{d}\theta\,\mathrm{d}x \tag{9}$$

We want to choose α as to make I biggest.

But now we need to consider how are we going to represent the parameter distributions $\rho(\theta) = \rho(\mu, \beta, \sigma)$??? And how are we going to update it...?

One very simplistic way to proceed is take a Gaussian approximations for the prior centred at the current ML estimates. Then to compute the argmax over α in eq. (9). But then to update the posterior using the Gaussian approx'n / ML estimates again, instead of computing the true Bayesian Update...

This makes for a very nice soup - we have entropy (as the objective), Bayesian updates (to form the entropy), ML likelihoods to center the estimates and Fisher Information computation (evaluated at the estimates) to compute the Gaussian Approximation (for the prior)... :-)

So

$$\rho(\theta) \propto \exp\Big((\theta - \hat{\theta}) \cdot \hat{\Xi}^{-1} \cdot (\theta - \hat{\theta})\Big)) \tag{10}$$

Two questions:

1. How are we going to deal with the normalizing constant: $\int_\Theta L(x|\theta;\alpha) \cdot \rho(\theta)$?
2. How are we going to compute the Fisher Information?

There are two ways to compute the Fisher Info, Φ:

1. As the covariance of the score $\Phi = \mathbb{E}[(\nabla_\theta l) : (\nabla_\theta l)]$, this is the definition of Φ
2. As the expected Hessian of the log-likelihood: $\Phi = \mathbb{E}[\nabla\nabla : l(\ldots)]$. This is only true, however, for the true parameters.

Hm! Maybe using Fisher Information for the Gaussian Approximation of prior/posterior of the parameters is a BAD idea. . . ?

More issues: in eq. (8) we have integration wrt. the RV x. If, this is only one observation of the process, then 'x' is just X_n, but if we take this to be K observations: then 'x' is $\{X_k\}_n^{n+K}$ and we have a K dimensional integral. . . However it does factor as we can integrate, first wrt X_n then x_{n+1} and so on all the way to X_{n+K}.

One thing to do is for fixed θ to sample a few X paths. This is akin to particle filtering. . . then the objective will look like:

$$I(\alpha) = \int_\Theta \sum_{X_i|\theta} \log\left[\frac{L(X_i|\theta;\alpha)}{\int_\Theta L(X_i|\theta;\alpha)\cdot\rho(\theta)\,\mathrm{d}\theta}\right]\cdot\rho(\theta)\,\mathrm{d}\theta \tag{11}$$

Now suppose that θ is chosen using some kind of a Gauss-Hermite quadrature scheme in 3-d. Say that requires 125 points (that is only five points in each direction, so quite conservative, but also should be quite efficient!) That means that to evaluate J we need to sample $I \times 125$ paths. And then do the summation. But wait, there's more. For each x, θ pair we also need to do the integral for the normalizing constant. . . so now we have $I \times 125^2$ calculations. Just to evaluate J.

Uf!

Still that is what we have computers for. . . But there should be something simpler to do. . .

2.1 Simple illustration

Let us make a simple proof-of-concept.

We will take for the true parameters

$$\beta = .05; \mu = -60; \sigma = .1;$$

Now since the values of μ, σ are determined by β, let us reduce the prior to a one-dimensional Gaussian $\rho(\beta) \propto \exp(-(\beta - \hat{\beta})^2/{\sigma_\beta}^2)$ and then for a given β we will obtain μ, σ from the formulas in eqs. (3) and (5).

We will only focus on the next observation, so the likelihood is also a gaussian distribution (conditional on μ, β, σ) in 1-d.

Let us write it out using x_0 as the current value and x as the future value and Δ as the time interval until the next observation: then the terms L, ρ in the mutual information

$$I(\alpha) = \int_\Theta \int_X \log\left[\frac{L(x|\theta;\alpha)}{\int_\Theta L(x|\theta;\alpha)\cdot\rho(\theta)\,\mathrm{d}\theta}\right]\cdot L(x|\theta;\alpha)\cdot\rho(\theta)\,\mathrm{d}\theta\,\mathrm{d}x$$

are given by:

$$
\begin{aligned}
L(x|\theta;\alpha) =& \frac{\beta}{\sigma\sqrt{2\pi(1-e^{-2\beta\Delta})}} \cdot \exp\left(\frac{\left(x - (\frac{\alpha}{\beta} + \mu) - (x_0 - \frac{\alpha}{\beta} - \mu) \cdot e^{-\beta\Delta}\right)^2 \cdot \beta}{\sigma^2(1-e^{-2\beta\Delta})} \right) \\
\rho(\theta) =& \frac{1}{\sigma_\beta\sqrt{2\pi}} \exp(-\frac{(\beta-\hat{\beta})^2}{\sigma_\beta^2})
\end{aligned}
$$

Thus for each x we need to form a Gaussian integral for the normalizing constant. And on top of that we need to make a two dimensional independent gaussian integral for the x, β.

Let us illustrate this. For the calculations we will use 1. Gauss-Hermite integration and 2. simple Curtis-Clenshaw integration as a sanity check.

Let us start with a sample of 50 (ms) observations sampled at .1 ms. Then if we take a coarser sampling of 1s to create our boot-strap of estimates for β we get:

$$
\begin{gathered}
\hat{\beta} \quad \sigma_{\hat{\beta}} \\
0.0881, 0.0182 \\
0.0974, 0.0514 \\
\boxed{0.0902, 0.0562} \text{//used in subsequent simulations} \\
0.3060, 0.2968 \\
0.0696, 0.0386 \\
0.1947, 0.0790 \\
0.1702, 0.0624 \\
0.1706, 0.0481 \text{//way off including too small std} \\
0.1332, 0.0879
\end{gathered}
$$

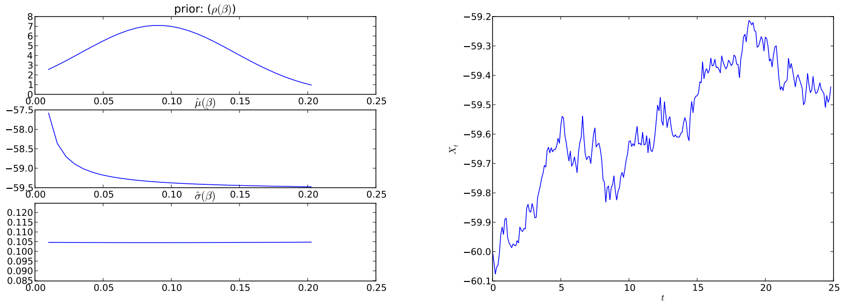
That is actually ok. In 1 case, we couldn't actually estimate β (the data implies there is no solution to eq. (6)). In the other 9 cases, 8 times the true value (.05) is within 2 standard deviations of the mean estimate and in 5 cases it is within 1 standard deviation of the mean.

Note that calculating σ_β as

$$\sigma_\beta^2 = \sum(\hat{\hat{\beta}} - \hat{\beta})^2$$

where $\hat{\hat{\beta}}$ are bootstrap estimates from a reduced data set helps increase the value of σ_β^2. If instead of $\hat{\beta}$, we use the mean of the bootstrap values themselves to calculate σ_β, we will get smaller values for σ_β, which will usually put the true value more than two standard deviations from the short-time estimate.

Let's use the values of $\hat{\beta}, \sigma_{\hat{\beta}} = 0.0902, 0.0562$ and visualize the resulting distributions for β and the functions $\mu(\beta), \sigma(\beta, \mu)$, see fig. 1. Essentially what we see is that μ depends on β inversely while σ does not really depend on β given the data and a μ estimate. That is perfectly natural. We know that the long term equilibrium of X_t is $\mu\beta$, so given a sample, $\hat{\mu}$ will be roughly equal to the sample mean divided by $\hat{\beta}$. On the other hand $\hat{\sigma}$ seems affected only by the product $\hat{\mu}\hat{\beta}$, so that for a given β, once μ is estimated the two variations cancel each other and we get the flat line on the bottom panel in fig. 1a. At this point we might start to think



(a) prior

(b) underlying X path

Figure 1: An example of a path, a prior over β built based on the path and the resulting functional relations between β, μ and σ, which are from eqs. (3) and (5). The prior for β and the resulting μ, σ are obtained using the path in (b)

a normal prior for a positive random variable is a bad idea, especially if we are near zero and the std. dev. is non-negligible. Perhaps, we should use a log-normal prior or a gamma distribution . . .

Now let's calculate the integral of the likelihood wrt. the prior for a fixed x, ie. the marginal forward distribution of x.

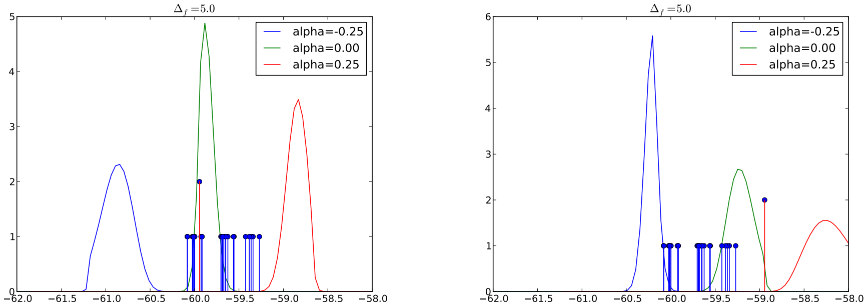
$$p(x) = \int_{\Theta} L(x|\theta; \alpha) \cdot \rho(\theta)\, \mathrm{d}\theta$$

This is also the normalizing constant in bayes rule. We will used a forward horizon of $\Delta_f = 5$ (ms) (For reference, the data has $\Delta = 0.1$ and has length $T = 50$ ms). $p(x)$ si shown in fig. 2a. Essentially, fig. 2a is telling us that the current value of x is unlikely, all things considered. Our current estimates for β, μ, σ are letting us believe that it should be significantly closer to 59.5 if we continue with the current value of $\alpha = 0$. The two opposing proposed values for $\alpha = \pm 0.25$ result in much more spread distributions for $X_{t+\Delta_f}$. In particular, one might argue that $\alpha = -0.25$ is most informative, since the forward distribution has the most spread. (if we intuitively equate spread with information). This is consistent with the notion that the most informative experiments are the ones that lead X furthest from its current equilibrium (given $\alpha = 0$). In this case the negative is better than positive, since the current value of X_t is already negative in relation to the current equilibrium. Basically this is consistent with the following selection mechanism of α: If $X_t > \mu\beta$ choose $\alpha_{\max}$ otherwise if $X_t < \mu\beta$ chose $\alpha_{\min}$. This is illustrated in fig. 2b, where we artificially move the value of X_t to the right of the (current) equili-

α	$I(\alpha)$
-0.25	0.688
0.00	0.100
0.25	0.419

Table 1: values of MI

birum, which makes the forward distribution arising from $\alpha = 0.25$ more informative (higher spread), then the one with $\alpha = -0.25$



(a) Left of equilibrium

(b) Right of equilibrium

Figure 2: Marginal of x (normalizing constant), the tall red stem is the current value of X_t. This is the last observed value based on which we compute transition densities. The blue and red curves are forward densities given different applied forward values for α. The green curve is the forward density given the hitherto used value of $\alpha = 0$. It mostly coincides with the so-far observed data which is the blue stems. In float (b) we artificially move the starting point to the right. Note that the green curves in a), b) are not the same, since they depend on the value of x_0 as well as the observed data (which is the same in both panels)

Let us verify this intuition formally, by calculating $I(\alpha)$, for $\alpha = [-0.25, 0, 0.25]$.

An aside: Unfortunately, naively calculating the double integral (in Python using 'quad' or 'romberg') is not numerically efficient. Calculating $I(\alpha)$ for a single α takes on the order of 10 secs, once you relax the quadrature tolerances without incurring any significant error… On the other hand the process is supposed to be taking on the order of milliseconds. We need to be NOT naive! Since we are dealing with Gaussian-type integrals, a Gauss-Hermite Integration scheme might be very effective… But let's leave that for now.

Now, crushing through with brute force we get the result in table 1. We have used the same starting (current) value of X_t to form the forward likelihoods as in fig. 2a and indeed we get the result.

$$I(-0.25) > I(.25) > I(.0)$$

which is consistent with our expectations after calculating the corresponding $p(x)|\alpha$ in fig. 2a. This basically says that it should be most informative to stimulate down ($\alpha < 0$), and it should be least informative to do nothing $\alpha = 0$.

2.2 Experiment

Let's now see what that means in practice. We will pretend that we have done the analysis in a blink and then see what happens when we further simulate the process from X_t for Δ_f time and see if there is any advantage to using the mutually most informative α, i.e. $\alpha = \arg\max I(\alpha)$, or not. First we generate 10 forward trajectories of duration $2\Delta_f = 10$ (ms). They are shown in fig. 3. Now what we would like to see is that the estimates corresponding to $\alpha = -.25$ are 'better' than the ones corresponding to $\alpha = .25$ and that they are much better than the ones corresponding to $\alpha = .0$.

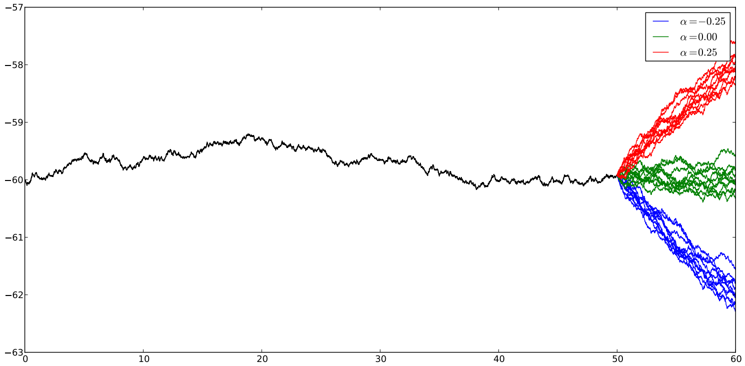


Figure 3: Different Trajectories perturbed by different values of α after the MI calculation

Let's see: The resulting estimates for the 10 trajectories are shown in fig. 4. Well, what do you know, visually, it is clear that indeed $\alpha = -.25$ is 'better' than $\alpha = .25$, which in turn is better than $\alpha = .0$.

2.3 Bang-Bang?

We expect that it is actually best to apply maximum inhibition or maximum excitation. That is we expect that $I(\alpha_2) > I(\alpha_1)$ for $|\alpha_2| > |\alpha_1|$. We verify this in table 2, where we see the general tendency that bigger is more informative than smaller and negative is more informative than

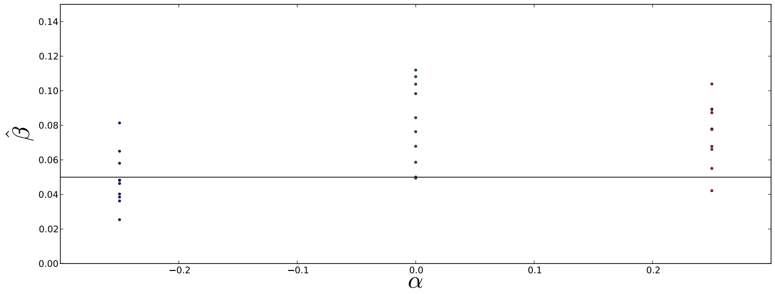


Figure 4: The estimates for β given the three perturbed trajectories (one is actually un-perturbed ($\alpha = .0$)). The solid black line indicates the true value of β

α	$I(\alpha)$
-2	2.31
-1.00	1.714
-0.50	1.182
0.50	0.885
1.00	1.563
2	2.20

Table 2: values of MI

positive. Table 2 and our intuition suggest then that the choice for which is always between the two extremes st. $\alpha_{\text{most informative}} \in \{\alpha_{\min}, \alpha_{\max}\}$.

Now we wonder what happens to the calculated value of I as we move Δ_f, see fig. 5. Now this is interesting. As Δ_f increases, we have a raise in the mutual information. Intuitively this is obvious, since bigger Δ_f means more data. However, recall that we are only considering the information contained in the final value (at $t = \Delta_f$). That is also why I levels off eventually, if we were considering the full path and not just the final value, then it should continue increasing monotonically, although perhaps with a decreasing slope. However! It is also clear that while in the current context and for the current example at this time, inhibition is best $I(-) > I(+)$. At some point in the future excitation will be better! That is as the X variable settles into its new, lower, equilibrium, $I(+) > I(-)$.

The question becomes how to formulate this problem as to decide on when to switch.

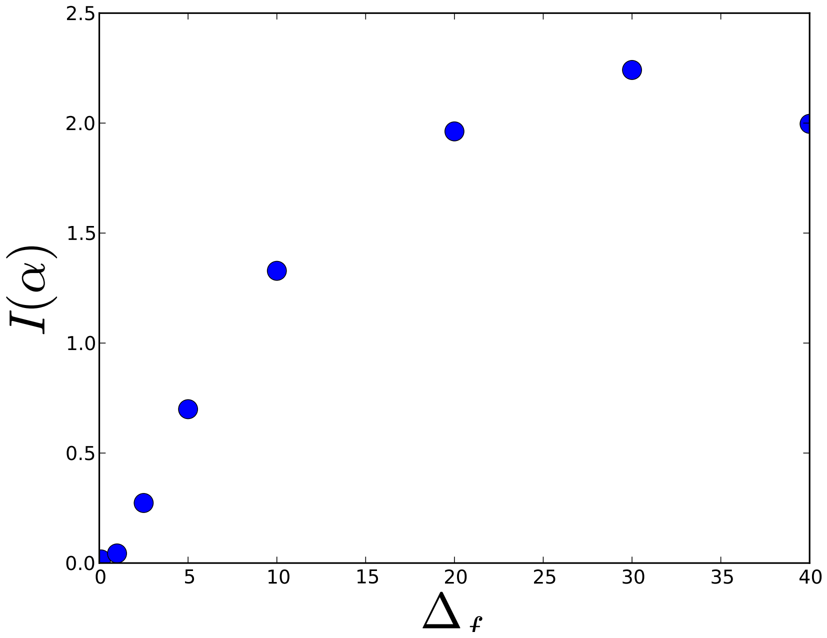


Figure 5: values of MI while varying Δ_f. Note that the right-most value is a little iffy, as the integration routines warn about possible problems with the variuos integrals

Let me explain why, potentially, this is an interesting problem, what is a possible solution and why that solution is possibly very difficult to enact:

2.3.1 Optimal Switching for Optimal Design

Let us recap where (we think) we are: There is an observed OU process X_t. We estimate it on the fly and have estimates for β, μ, σ or more precisely we have a distribution for β and a one-to-one relation between β and μ, σ.

We have a control of α with which we can stimulate X. We want to use α to improve the observations of β, μ, σ. We use the Mutual Information criterion, $I(\alpha)$ to select α. From the structure of the OU process, it is conjectured and empirically observed that the bigger in magnitude α the more informative it will be. Thus, in the absence of an energy cost, we are left only to select between the two extreme values of α, $[\alpha_{\min}, \alpha_{\max}]$, which we can assume to be just $[\pm\alpha_{\max}]$ for some $\alpha_{\max} > 0$. Thus at any time, t, we can calculate $I(\alpha_{\min}), I(\alpha_{\max})$ and choose the α associated with the larger I.

However, in a sense this is greedy and thus not necessarily optimal.

Here is a simple way to think about it:

Imagine that $\alpha_{\min}, \alpha_{\max}$ correspond to two equilibria x_-, x_+. Our intuition is that there is a mid-point x_{mid} between x_-, x_+ st. if $X_t < x_{mid}$, we should apply $\alpha_{\max}$ and conversely. It is now clear that this can easily result in chattering - being below x_{mid} we stimulate which sends us above x_{mid} and then we inhibit, since above x_{mid} it is most informative to inhibit and so on.

When you add the noise, it is hard to see if that is even a better idea than doing nothing.

Of course, things are not so simple as the distribution on the parameters, ρ may make x_{mid} itself move as ρ shifts and shrinks, but let's ignore that for now.

At this point it becomes clear that we need a way to choose the switching time to switch between $\alpha_{\max}, \alpha_{\min}$ using something more sophisticated than just the instantaneous value of $I(\alpha)$. This is related to the problem of how to choose the observation time Δ_f in the formulation of $I(\alpha)$.

Basically we need to select what we are going to do now in part based on what we can do later. And that is Dynamic Optimization!

The main difference here from standard Dynamic Optimization is that our state is not so much the value of X_t but the value of $\hat{\beta}_t, \hat{\mu}_t, \hat{\sigma}_t$, the estimates at time t. Once we realize this, we also realize our main challenge - the updates for β, μ, σ using the ML formulas are non-Markovian! That is we look back on all the old data when taking the new data into account to form $\hat{\beta}_t, \hat{\mu}_t, \hat{\sigma}_t$.

So we need to do one of two things

1. come up with a heuristic way of choosing Δ_f before which to consider switching (if Δ_f is large enough, we will always switch (I think))
2. Come up with an incremental form for updating β

Since route 2 leads to Dynamic Programing type equations, we follow it.

2.4 Framing the Problem as a Dynamic Program

FROM HERE ON IT IS ALL SPECULATIVE AND MIGHT AS WELL OMIT READING

Let us assume that we have figured out how to update β, equivalently $\rho(\beta)$ truly based on Bayesian updates, i.e. Markov-like. A very crude method to this end is to take the posterior and simply project it onto a Gaussian...

Here comes the second challenge - roughly speaking we have an infinite dimensional state space - the belief distribution of β. Now, as soon as we parametrize the belief distribution to a normal with mean variance $\bar{\beta}, \sigma_b^2$ then we are back to finite dimensions and we are good to go.

We will consider framing the problem as a discounted infinite-horizon dynamic program:

$$v(\rho_t(\beta), X_t) = \max_{\alpha_k} \sum_{k \geq 0} \gamma^k I(\alpha_k, \Delta_k; X_k, \rho_k(\beta)) \tag{12}$$

where $\gamma \in (0,1)$ is an arbitrary discount factor and $\Delta_k = t_{k+1} - t_k$ is the forward increment which is assumed fixed. A simplification thus is that we assume the observation times are a priori known and fixed.

So as long as we figure a smart way to update the parameter belief distribution(s) online, we can then try to solve the Bellman equation for infinite horizon, discrete-step problems:

$$v(x_0, \rho_0) = \max_{\alpha \in \{\alpha_{\min}, \alpha_{\max}\}} \Big[I(\alpha, \Delta_0; x_0, \rho_0(\beta)) + \gamma \, \mathbb{E}_\alpha[v(x_1, \rho_1)] \Big] \tag{13}$$

and then choose the optimal stimulus simply as:

$$\alpha = \arg\max_\alpha \Big[I(\alpha, \Delta_0, x_0, \rho_0(\beta)) + \gamma \, \mathbb{E}_\alpha[v(x_1, \rho_1)] \Big]$$

On one hand this looks deceptively simple. On the other it is strictly more complicated than the greedy approach where α is selected based on:

$$\alpha_{\text{greedy}} = \arg\max_\alpha \Big[I(\alpha, \Delta_0; x_0, \rho_0(\beta)) \Big]$$

If the calculation of the mutual information I is expensive (as so far it has been), we can rule out small Δ_f i.e. small Δ_0, at least in the calculation of the switch times. We can still observe at a higher frequency.

A few more words on the value function, v and its arguments x_0, ρ. If we take the Gaussian approximation for $\rho(\beta) \sim N(\bar{\beta}, \sigma_\beta)$ then $v(x_0, \rho) = v(x_0, \bar{\beta}, \sigma_\beta)$, which is just a 3-dimensional function. If we want to have a full prior, $\rho(\mu, \beta, \sigma)$ then we will have a 1+3+6-dimensional function. Already at $d = 3$, but certainly at $d = 10$ we will need to apply some kind of function approximation to v.

$$v(\ldots) = \sum_j r_j \phi_j(\ldots)$$

where ϕ_j are appropriately chosen basis functions.

Let us dissect a little bit eq. (13). In particular what is $\mathbb{E}_\alpha[v(x_1, \rho_1)]$? Well the expectation is taken with respect to what is random, i.e. the next realization of X, that is the value of $X_{t+\Delta_0}$. But $X_{t+\Delta_0}$ must further be conditioned on the current value of the belief distribution. So

$$\mathbb{E}_\alpha[v(x_1, \rho_1)] = \int_{\{X,\theta\}} v(x, \rho_1(|x))L(x|\theta)\rho_0(\theta)\,\mathrm{d}\theta\,\mathrm{d}x$$

We write $\rho_1(|x))$ as a reminder that the posterior depends on the realized value x. Basically, for every possible θ, x in the joint forward density, we need to consider what will be the updated ρ_1. And thus we need an efficient way to update ρ_1.

We now explore in turn the two main challenges:

1. How to update the prior → posterior efficiently
2. How to solve the bellman equation, eq. (13)

2.4.1 Updating the Prior on the fly

Recall that the great disadvantage of the ML formulas for β, μ, σ is that they are not recursive. Moreover the bootstrap approach to form priors is even more non-recursive:-! This robs us of a Markovian structure, and in turn we cannot use Dynamic PRograming techniques / the bellman equation.

We need to be more shrewd.

Note that the parameter belief distributions is only used to guide the online OptDesign process, i.e. the on-the-fly seclection of α. There is nothing stopping the analyst in using the ML formulas once all the data has been collected.

Thus, we can hope that we can use a less efficient estimator than ML as long as it is roughly correct. In particular having the wrong value of σ_β shouldn't be a big deal… In fact, from messing around with the routines, it seems the only thing that is really of any importance is that σ_β is not zero. If it were zero then $I(\alpha) = 0\,\forall\alpha$, which is reasonable since X carries no additional information on the already fully specified parameter set θ. Other than that, wildy different values for $\hat{\beta}, \sigma_\beta$ seem to give the same ordering between $I(\alpha_{\min}), I(\alpha_{\max})$… This is a little strange, since it looks like the prior doesn't really matter… but let us move on for now.

Ok, how do we efficiently update the belief distributions given a new observations, and approximately if necessary.

Let us make it hard for ourselves at first. Assume a joint prior $\rho(\mu, \beta, \sigma)$.

Then the posterior is

$$p(\mu, \beta, \sigma|x) = \frac{L(x|\theta)\cdot\rho(\theta)}{\int_\Theta L(x|\theta)\rho(\theta)\,\mathrm{d}\theta}$$

where the likelihood is given by

$$L(x|\theta) = \frac{\beta}{\sigma\sqrt{(1-e^{-2\beta\Delta_0})}}\cdot\exp\left(\frac{\left(x - (\frac{\alpha}{\beta}+\mu) - (x_0 - \frac{\alpha}{\beta} - \mu)\cdot e^{-\beta\Delta_0}\right)^2\cdot\beta}{\sigma^2(1-e^{-2\beta\Delta_0})}\right)$$

So what is a good way to update $\rho\ldots$. Well, in a sense what we have here is a (simple) filtering problem. β, μ, σ can be thought as the unknown (hidden) process which is actually static, while X is the observation process. Assuming that we cannot close-form the updates for the posterior, which is the same as stating that we cannot find in analytical form a conjugate prior, we need to open the toolbox of filtering theory, such as one of

1. Extended Kalman Filters
2. Particle Filters
3. Sigma-Point (Unscented Kalman) Filters
4. ...

2.4.2 The Value function

Here comes the tricky part. Any of the above will allow us to form an approximate posterior, p. But will any of them be usable in the DP formulation above? The simplest case is if v only depends on 2-nd order statistics of p. Then we can just ask the approximate posterior for its 2-nd order stats and then plug them into v

Thus all we need to do is calculate the moments:

$$\bar{\theta}(x) = \int_\Theta \theta \cdot p(\theta|x)\,\mathrm{d}\theta = \int_\Theta \theta \cdot \frac{L(x|\theta)\cdot\rho(\theta)}{\int_\Theta L(x|\theta)\rho(\theta)\,\mathrm{d}\theta}\,\mathrm{d}\theta$$

and

$$\Xi_\theta(x) = \int_\Theta (\theta - \bar{\theta}) : (\theta - \bar{\theta}) \cdot p(\theta|x)\,\mathrm{d}\theta = \int_\Theta (\theta - \bar{\theta}) : (\theta - \bar{\theta}) \cdot \frac{L(x|\theta)\cdot\rho(\theta)}{\int_\Theta L(x|\theta)\rho(\theta)\,\mathrm{d}\theta}\,\mathrm{d}\theta$$

where we use : to write the outer product to form the variance (we only need to calculate the upper triangle of the matrix).

and then have $v(x, \rho) = v(x, \bar{\theta}, \Xi_\theta)$. This makes v a function of 1+3+6 parameters, if we have 3 unknown params, μ, β, σ

Let's take a step back - to calculate the value function, we now need to calculate:

$$\mathbb{E}_\alpha[v(x_1, \rho_1)] = \int_{\{X,\theta\}} v(x, \rho_1(\theta|x)) L(x|\theta, \alpha)\rho_0(\theta)\,\mathrm{d}\theta\,\mathrm{d}x$$

and then for each value of x inside the integral we need to further calculate what the moments of the posterior would be using the moment updating equations above (or others). That is just to evaluate the expectation of the value function.

In the best case, when we need to choose over only two values of α, we will have to evaluate I, v twice, once for each alternative α.

What we need to hope for is that a rough approximation for v will be good enough...

2.4.3 Cts-time observations

Sometimes things are easier if they are stated in continuous time. Let $f(x_t, t|x_0, 0; \theta)$ be the forward density of x given θ. f is a soln to the Forward Kolmogorov/ Fokker-Planck equation. Then the mutual information is the limit of:

$$I(X,\theta) = \int_\theta \int_{X_{t_1}} \cdots \int_{X_{t_n}} \log\left(\frac{\prod_n (f(X_{t_{n+1}}, t_{n+1}|X_{t_n}, t_n; \theta)}{\int_\theta \prod_n (f(X_{t_{n+1}}, t_{n+1}|X_{t_n}, t_n; \theta)\rho(\theta)\,\mathrm{d}\theta} \right) \cdot \prod_n (f(X_{t_{n+1}}, t_{n+1}|X_{t_n}, t_n; \theta) \cdot \rho(\theta)\,\mathrm{d}x_1 \cdots \mathrm{d}x_n\,\mathrm{d}\theta$$

as X_{t_n} refines to the full cts-time trajectory X_t. It is clearly unfeasible to evaluate I as we must do an infinite number of joined integrals. The only approach is to do a Monte Carlo integration using X_t trajectory samples.

However one may try to pose the problem in the more simplistic average of each time:

$$\tilde{I}(X,\theta) = \int_\theta \int_0^T \int_X \log\left(\frac{f(x,t|\theta)}{\int_\theta f(x,t|\theta)\rho(\theta)} \right) f(x,t|\theta) \cdot \rho(\theta)\,\mathrm{d}x\,\mathrm{d}t\,\mathrm{d}\theta$$

Once again, $\tilde{I}$ is not the mutual information, I is. But the equation to maximize $\tilde{I}$ looks like a dynamic programing problem. It can also be written as:

$$\tilde{I}(X,\theta) = \mathbb{E}_\theta\,\mathbb{E}_{X_t}\left[\int_0^T \log\left(\frac{f(x,t|\theta)}{\int_\theta f(x,t|\theta)\rho(\theta)} \right) \mathrm{d}t \right]$$

which is the form necessary to apply the dynamic programing equations, except that we have an extra expectaton (wrt. to the parameter prior):

2.4.4 the value function in cts. time

NOTE: THIS IS A MESS

Instead we might find that working in cts. time makes things easier for us.

$$v(\rho(\theta), x) = \max_{\alpha_k} \mathbb{E}\left[\int_{t\geq 0} e^{-\gamma t} i(\alpha_k, dt; X_t, \rho_t(\theta))\,\mathrm{d}t \right] \tag{14}$$

where i is the infinitesimal mutual information for a small interval dt

$$i(\alpha_k, dt; X_t, \rho_t(\theta) =$$

Now we know that the continuous time-likelihood L in general for a stochastic process

$$dX = f dt + \sigma dW$$

over the interval $[0,t]$ is

$$L(X) = \int_0^t \frac{-1}{2}\frac{f^2}{s^2} dt + \frac{f}{s} dW$$

where the W in the integral corresponds to the one that generated the data X_t.

Now we are trying to maximize the running reward

$$\mathbb{E}[\int_0^\infty e^{-rt} i(X_t, \rho_t(\theta); \alpha) \,\mathrm{d}t]$$

where i is the infinitesimal mutual information between $X_t + dX$ and the posterior $p(\theta) \sim \rho_t(\theta) + d\rho$. The value function is written as:

$$v(x_0, \rho_0(\theta)) = \sup_\alpha \mathbb{E}[\int_0^\infty e^{-rt} i(X_t, \rho_t(\theta); \alpha) \,\mathrm{d}t$$

Now for θ fixed. This leads to the Bellman equation:

$$0 = -rv(x, \rho) + \sup_\alpha [i(x, \rho; \alpha) + \frac{\mathbb{E}[v(x + dx, \rho + d\rho)]] - v(x, \rho)}{dt} \qquad \forall x_0, \rho_0$$

The interesting term in this expression is, of course,

$$\frac{\mathbb{E}[v(x + dx, \rho + d\rho)]] - v(x, \rho)}{dt}$$

If v were a function of x only then this would be just

$$\frac{\mathbb{E}[v(x + dx)] - v(x)}{dt} = \int_\theta {\mathcal{L}^*}_{\theta,\alpha}[v] \cdot \rho(\theta) \,\mathrm{d}\theta$$

where $\mathcal{L}^*$ is the generator for X_t's diffusion given fixed, θ, α i.e.

$$\mathcal{L}^*[\cdot] = f(x; \theta, \alpha) \cdot \partial_x[\cdot] + \frac{\sigma^T \sigma}{2} : \partial_x^2[\cdot]$$

HOWEVER! if v is a function of both the observation and the current belief distribution X_t, ρ_t, things become more complicated since the posterior depends on X i.e.

$$v(x + dx, \rho + d\rho) = v(x + dx, \rho(x + dx))$$

This means that when we do things like take x-differentials, $\partial_x v$, we must take total differentials

$$\partial_x v = \partial_1 v(x, \rho(x)) + \partial_2 v(x, \rho(x)) \cdot \partial_x(\rho(x))$$

where we write ∂_1, ∂_2 to indicate we are differentiating with respect to the first or second argument of v Now what is ∂_ρ? Well, for fixed θ, $\rho(\theta|x)$ is just a number. it is the probability or probability density of θ given x. So for fixed θ ∂_ρ is a single variable derivative.

So we are left to wonder what is the shape of the derivative of the posterior wrt x evaluated at x_0 (the last observation).

This is not obvious. Let's start from the posterior-prior definition (i.e. Bayes rule):

$$p(x) = \frac{L(x|\theta)\rho(\theta)}{\int_\theta L(x|\theta')\rho(\theta') \,\mathrm{d}\theta'}$$

For small dt, the likelihood $L(x)$ is a highly peaked normal, centred at x_0 with variance dt.

$$p(x) = \frac{\exp(-\frac{(x-(x_0+f(x_0,\theta)dt))^2}{2\sigma^2 dt} \cdot \rho(\theta))}{\int_\theta \exp(-\frac{(x-(x_0+f(x_0,\theta)dt))^2}{2\sigma^2 dt} \cdot \rho(\theta) \,\mathrm{d}\theta}$$

Now $\partial_x p = \partial_x \log(p) \cdot p$, so we will try to calculate the derivative of the log instead

$$\log(p(x)) = -\frac{(x-(x_0+f(x_0,\theta)dt))^2}{2\sigma^2 dt} + \log(\rho) - \log(\int_\theta \exp(-\frac{(x-(x_0+f(x_0,\theta)dt))^2}{2\sigma^2 dt} \cdot \rho(\theta)\,\mathrm{d}\theta)$$

Taking its derivative

$$\begin{aligned}\partial_x \log(p) = &-\frac{(x-(x_0+f(x_0,\theta)dt))}{\sigma^2 dt} \\ &-\frac{\int_\theta -\frac{(x-(x_0+f(x_0,\theta)dt))}{\sigma^2 dt} \cdot \exp(-\frac{(x-(x_0+f(x_0,\theta)dt))^2}{2\sigma^2 dt} \cdot \rho(\theta)\,\mathrm{d}\theta}{\int_\theta \exp(-\frac{(x-(x_0+f(x_0,\theta)dt))^2}{2\sigma^2 dt} \cdot \rho(\theta)\,\mathrm{d}\theta}\end{aligned}$$

The crudest thing to say is that $\partial_x(\rho(x)) = 0$! B/c if it is not, the complications become untenable.

Then the dynamic programing equation reduces to:

$$0 = \sup_\alpha \left\{ i(\alpha, x, \rho) + \int_\theta \mathcal{L}^*_\theta[v(x, \rho(\theta))] \cdot \rho(\theta)\,\mathrm{d}\theta - \gamma v(x, \rho(\theta)) \right\} \tag{15}$$

for all possible state-points x, and priors ρ. As this is an infinite-dimensional state-space problem, we need to have a way to truncate ρ to a finite dimension.

How we solve eq. (15) is discussed next.

2.4.5 Value function approximation

Value function approximation refers to approximting v with a pre-determined basis:

$$v(x, \rho) = \sum_k r_k \phi_k(x, \rho)$$

The art is choosing the functions ϕ_k so as to fit the problem. A good choice of ϕ will mould to the actual (but unknown) shape of v.

I can't right now come up with obvious examples of ϕ that will be appropriate. A common generic choice is tensor-products of Chebyshev polynoms up to order 3.

3 Gauss-Hermite Integration

NOTE: THIS DOES NOT WORK FOR NOW, SO THIS SECTION IS OPTIONAL READING:

Gauss-Hermite integration is a well-known technique for Gaussian quadrature on the Real axis with a Gaussian weight: i.e. how to integrate:

$$I(f) = \int_{\mathbb{R}} f(x)e^{-x^2}\,\mathrm{d}x \tag{16}$$

The general theory is in sec 4.5 of [4], here we just cite the results:

$$I(f) = \sum_{j}^{d} w_j f(x_j) \tag{17}$$

where the weights and nodes w_j, x_j are related to the roots of the Hermite polynomials of order d. (We use a function in numpy, hermite.hermgauss, to obtain them for a given d).

This suffices to integrate against an arbitrary Gaussian weight. How?

$$\begin{aligned} I(f) =& \frac{1}{\sigma_y\sqrt{2\pi}} \int_{\mathbb{R}} f(y)e^{-\frac{(y-\bar{y})^2}{2\sigma_y^2}}\,\mathrm{d}y \\ =& \frac{1}{\sqrt{\pi}} \int_{\mathbb{R}} f(\sqrt{2}\sigma_y x + \bar{y})e^{-x^2}\,\mathrm{d}x \\ =& \frac{1}{\sqrt{\pi}} \sum_j w_j f(\sqrt{2}\sigma_y x_j + \bar{y}) \end{aligned}$$

To relate this explicitly to our case above, integrals of against the β prior can be calculated as:

$$I(f) = \int_{\mathbb{R}} f(\beta)\rho(\beta)\,\mathrm{d}\beta = \frac{1}{\sqrt{\pi}} \sum_j w_j f(\sqrt{2}\sigma_\beta x_j + \bar{\beta})$$

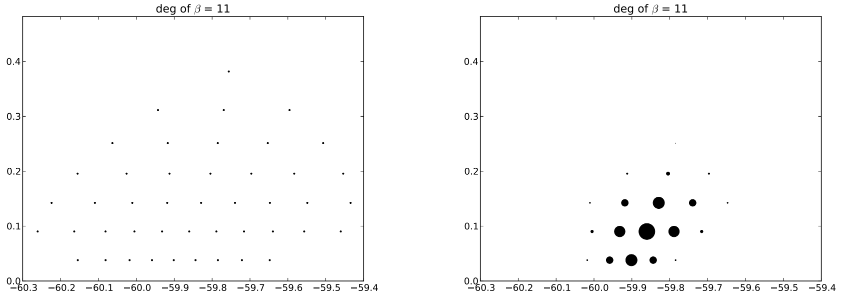
I.e. we just shift and scale the generic variable $x \rightarrow \sqrt{2}\sigma_\beta x + \bar{\beta}$.

Our problem above consists of a double integration, where the x variable is normal conditional on β, but the two are certainly not jointly normal! So we will proceed as follows:

We lay out a gauss-hermite grid of nodes for the β variable and then for each beta node, we will lay out a gauss-hermite grid for x.

At this point it again worth pointing out that a pure normal distribution for β is a bad idea! And cutting off the nodes for $\beta < 0$, works (roughly), but spoils the spectral convergence of the quad-rule. For now we continue with this bad prior, just so we can compare with our previous results.

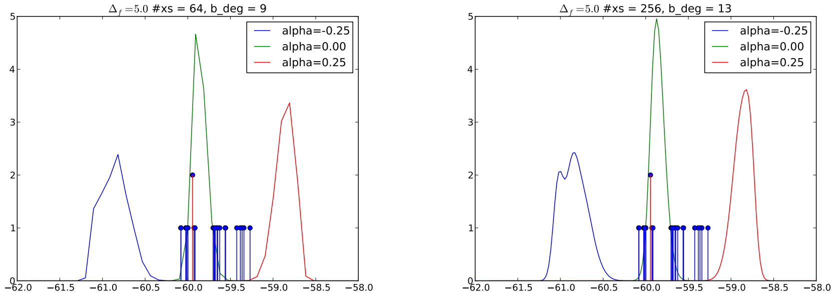
We visualize the node-spread in fig. 6. It looks like we only need about 9 nodes in the β direction. However when we attempt to duplicate the marginal figures from fig. 2a, we run into complications. See fig. 7. In particular even at high nodes, we run into this blip for the forward density of X for $\alpha == .25$. In general this is probably not there. We think it appears because the likelihood of X which is being integrated against the prior of β is not sufficiently smooth. IN particular, Gauss-type quadrature works best for functions that are low-order polynomials or at more generally that are analytic with a power-series representation that drops off sufficiently fast. Now, for fixed forward X the likelihood has a term of the form α/β. This may be the cause of the blip on the right. (Note that there is no blip for $\alpha = 0$ in fig. 7). Still, the fact that the overall shape is maintained, leads us to move on.



(a) node distribution

(b) node distribution

Figure 6: Spread of the nodes in the $x - \beta$ domain. All the nodes are illustrated on the left and the nodes are proportional to their weight $w_{j,x} \cdot w_{i,\beta}$. Basically more than 9 nodes in the β direction seem not necessary, but see fig. 7



(a) Coarse Nodes

(b) Fine Nodes

Figure 7: The same figure as in fig. 2a, but computed with Gauss Hermite quadrature of different order

We proceed as follows. For every b_i, x_j node we calculate,

$$f(x_j, \beta_i) = \log \left[\frac{L(x|\theta;\alpha)}{\int_\Theta L(x|\theta;\alpha) \cdot \rho(\theta)\, \mathrm{d}\theta} \right]$$

deg(β)	$I(\alpha = [-.25, .0, .25])$
7	['0.793', '0.132', '0.420']
9	['0.752', '0.110', '0.380']
13	['0.684', '0.087', '0.331']
21	['0.824', '0.136', '0.421']
43	['0.824', '0.135', '0.418']
87	['0.777', '0.114', '0.384']

Table 3: GH convergence of $I(\alpha)$ with increasing β order

deg(β)	deg(x)	$I(\alpha = [-.25, .0, .25])$
9	1	['0.7528', '0.1105', '0.3806']
9	3	['0.7513', '0.1101', '0.3802']
9	7	['0.7514', '0.1100', '0.3801']
9	11	['0.7516', '0.1100', '0.3802']

Table 4: GH convergence of $I(\alpha)$ with increasing x order, we see that using 1 for the base degree is roughly enough and 7 more than sufficient. Note that if the deg(β) was any larger, than the convergence in x will be even faster (there will be no difference to 4 sig. digits between 1 and higher).

and then we simply sum this

$$I(\alpha) = \frac{1}{\pi} \sum_{i,j} w_j w_i f(x_j, \beta_i).$$

We need to be careful and keep track of the normalizing constant, which in two dimensions is $(1/\sqrt{\pi})^2$.

The main thing to do is to try to recalculate the results in table 1, but with the GH quadrature scheme. Hm... I can't quite do it. In particular there is no convergence as we raise the β order see table 3. It is a little frustrating b/c for $d_\beta = 13$ for example, the $I(-.25)$ is right. but then it goes completely wrong. It is also frustrating, because the general ordering is consistently maintained:

$$I(-.25) > I(.25) > I(0)$$

On the other hand, for a fixed β degree, varying the x degree is almost useless, see table 4, convergence happens almost instantaneously!

So we only need to think about what are we doing wrong in terms of β. Because for deg(β) fixed convergence is achieved.

So... either this just doesn't work or we have made a mistake while GH integrating against β. Let's play 'find the bug... '. Well, one thing to try is compute the normalizing constant by conventional methods ('quad' in numpy) instead of through the GH method. Nope that has no effect (only 3rd sig. digit effect on $I(\alpha)$). Also integrating a purely β expression like β or $(\beta - \hat{\beta})^2$ gives the expected results (mean, variance)...

At this point I am going to suspend further efforts in this direction ... although as we have seen the efficiency gains can be pretty good (compute time on the order of .1 s to compute the $I(a)$ which once ported to C brings us into real-time)

A Mutual Info calculation

Here we show why eq. (8) for the Mutual Information agrees with the usual definition of the Mutual Information, which is

$$I = \int_\Theta \int_X p(x,\theta) \cdot \log \left(\frac{p(x,\theta)}{p(x)p(\theta)} \right) \mathrm{d}x \, \mathrm{d}\theta$$

First of all, $p(\theta) = \rho(\theta)$ is just the prior of θ The joint distribution $p(x,y) = L(x|\theta)\rho(\theta)$, while the x marginal, $p(x) = \int_\Theta L(x|\theta)\rho(\theta)\,\mathrm{d}\theta$. Pluging the three expressions gives:

$$I = \int_\Theta \int_X L(x|\theta)\rho(\theta) \cdot \log \left(\frac{L(x|\theta)\rho(\theta)}{\int_\Theta L(x|\theta)\rho(\theta)\,\mathrm{d}\theta \cdot \rho(\theta)} \right) \mathrm{d}x \, \mathrm{d}\theta$$

after canceling $\rho(\theta)$ inside the log, we get eq. (8).

B Relation between Mutual Information and Fisher Information

We'd like to explore the relation between Mutual Information and the Fisher Information in cts. time. In order to relate to the work of Hooker et al. [2].

The system is

$$dX_t = f(x, \theta, \alpha)dt + \sigma dW$$

st.σ does not depend on θ, α, but may depend on X_t

The corresponding cts-time log-likelihood is written as:

$$l(\theta|X_0^T) = \int_0^T \frac{f^2}{2\sigma^2}dt + \frac{f(x;\theta)}{\sigma^2}dX = \int_0^T \frac{-f^2}{2\sigma^2}dt + \frac{f(x;\theta)}{\sigma}dW$$

Then the Fisher Information is evaluated as:

$$\Phi(\theta) = \mathbb{E}\left[\int_0^T \frac{(\nabla_\theta f)^2}{\sigma^2}dt\right]$$

In general the fisher information is defined to be

$$\Phi(\theta) = \mathbb{E}_X\left[(\nabla_\theta l(\theta|X))^2\right]$$

And I don't exactly see how to go from the definition to the expression $\mathbb{E}\left[\int_0^T \frac{(\nabla_\theta f)^2}{\sigma^2}dt\right]$... Hooker et al. quote Rao 1999's book, which is a pretty serious book, so I am sure they are right. I should check it out.

Now we would like to relate the Fisher Information to the Mutual Information given some prior of the parameters. Perhaps the Fisher Info is a limit of the Mutual Info as the prior shrinks to a point... I don't know... what is their relation if any?

One thing is clear, however:

In the case of many parameters, the Fisher Information is a matrix, so one has to reduce it to a single metric by taking some norm of the matrix, like its determinant or its max eigen value or the sum of its eigen values and the choice of norm is subjective.

On the other hand, the Mutual Information is always a scalar, no matter the dimension of the unknown parameter. A priori, this gives a certain elegance advantage to methods based on maximizing the Mutual Info vs. the Fisher Info.

C Hooker et al. [2] take on the prior

Of course, the Fisher Information depends on the parameter that we are trying to estimate:

In

References

[1] Jeremy Lewi, Robert Butera, and Liam Paninski. Sequential optimal design of neurophysiology experiments. Neural computation, 21(3):619–87, March 2009.

[2] Kevin K Lin, Giles Hooker, and Bruce Rogers. Control Theory and Experimental Design in Diffusion Processes. pages 1–23.

[3] Jay I Myung, Daniel R Cavagnaro, and Mark a Pitt. A Tutorial on Adaptive Design Optimization. Journal of mathematical psychology, 57(3-4):53–67, January 2013.

[4] William H. Press, Brian P. Flannery, Saul A. Teukolsky, and William T. Vetterling. Numerical Recipes in C: The Art of Scientific Computing, Second Edition. Cambridge University Press, 1992.