3 Preliminary Work

Todorova and Ventura (to be cited) showed that searching the minimum prediction risk model over a large class of models can double the efficientcy of a common Kalman filter decoder in offline reconstruction of arm reaches of a Rhesus macaque monkey.

Details of their work will be in the Lit sec. Waiting for the work to be published

Based on the same idea we want to provide an efficient model selection procedures that allow us to include non-linear transformation of the spike counts and multiple observation equations per neuron.

3.1 Generalized Weighed Least Squares (GWLS)

Let us recall that the generative model can be written as

$$\mathbf{z} = H\mathbf{v} + \epsilon \tag{1}$$

with covariance matrix $\Sigma \in R^{N \times N}$, where we removed subscript t for the sake of notation. In the first part of this section we will "forget" about the temporal prior specification and think to estimate the velocity only from equation (1).

The GLSE(Generalized Least Square Estimator) for $\mathbf{v}$ can be computed as

$$\hat{\mathbf{v}} = (H^T \Sigma^{-1} H)^{-1} H^T \Sigma^{-1} \mathbf{z}, \tag{2}$$

where $\hat{\mathbf{v}}$ is the BLUE(Best Linear Unbiased Estimator) when Σ is known.

Based on Todorova and Ventura (2016) results, we do not believe that model (2) is correct for all neurons at the sime time, instead we want to derive a new estimator that weight the different observation equations based on "how good they are". This problem is apparently common in survey analysis where units could be sampled with unequal probability and it is necessary to give them unequal weight, Lumely [1].

If we let W be the weight matrix $W = \text{diag}(w_1, \ldots, w_n)$, where n represents the total number of neurons, we suggest a new estimator

$$\hat{\mathbf{v}}_W = (H^T \Lambda^{-1} H)^{-1} H^T \Lambda^{-1} \mathbf{z}, \tag{3}$$

where $\Lambda = W^{-1/2} \Sigma W^{-1/2}$ is a scaled version of the covariance matrix. In practice, the new estimator downweights the "worst" neurons.

Letting $n = 2$, and $\theta_i = 1/w_i$ for all $i = 1, \ldots n$, Λ can be visualized as

$$\Lambda = \begin{pmatrix} \sqrt{1/w_1} & 0 \\ 0 & \sqrt{1/w_2} \end{pmatrix} \begin{pmatrix} \sigma_{11} & \sigma_{12} \\ \sigma_{21} & \sigma_{22} \end{pmatrix} \begin{pmatrix} \sqrt{1/w_1} & 0 \\ 0 & \sqrt{1/w_2} \end{pmatrix} = \begin{pmatrix} \theta_1 \sigma_{11} & \sqrt{\theta_1 \theta_2} \sigma_{12} \\ \sqrt{\theta_1 \theta_2} \sigma_{21} & \theta_2 \sigma_{22} \end{pmatrix}. \tag{4}$$

H and Σ are estimated in the ecoding step, as usual in the Kalman Filter procedure, while W is estimated by minimizing the mean square error between the true velocity $\mathbf{v}$ and the estimated velocity $\hat{\mathbf{v}}_W$, which can be written as

$$\widehat{W} = \arg\min_{\{w_1, .., w_n : \sum w_i = 1\}} MSE(\mathbf{v}, \mathbf{v_W}) = \arg\min_{\{w_1, .., w_n : \sum w_i = 1\}} (\hat{\mathbf{v}}_W - \mathbf{v})^T (\hat{\mathbf{v}}_W - \mathbf{v}). \tag{5}$$

We can possibly apply other convex penalties but at the moment we will focus on differentiate the expression in equation (5) and possibly prove its convexity with respect with the weights. Notice that adding convex penalties would preserve the convexity of the Penalized MSE.

If we let $\Gamma = \Lambda^{-1}$, miniming equation (5) corresponds to solve the following system of equations

$$\begin{cases} H(H^T\Gamma H)^{-1}H^T\Gamma = \mathbf{I}_{(n,n)} \\ H^T H(H^T\Gamma H)^{-1}H^T\Gamma = H^T \\ H^T\Gamma = H^T(H^T\Gamma H)(H^T H)^{-1}. \end{cases} \tag{6}$$

3.2 Boosting Approach

3.3 Leave One Neuron Out Approach

3.4 Forward Approach: minimizing the Penalized MSE through direct regression

A Minimization of the MSE

Let's recap that the new estimator for $\mathbf{v}$ is

$$\hat{\mathbf{v}}_W = ((H^T\Lambda^{-1}H)^{-1}H^T\Lambda^{-1}\mathbf{z})_{(p,1)} = S_{(p,n)}\mathbf{z}_{(n,1)}, \tag{7}$$

where $\Lambda = W^{-1/2}\Sigma W^{-1/2}$. This estimator is unbiased, in fact

$$\mathbb{E}(\hat{\mathbf{v}}_W) = \mathbb{E}((H^T\Lambda^{-1}H)^{-1}H^T\Lambda^{-1}H\mathbf{v}) = \mathbf{v}_W.$$

Let's now specify the dimensions for the matrix involved: $B_{(n,p)}$, $B^T_{(p,n)}$, $(BB^T)_{(n,n)}$, $(B^TB)_{(p,p)}$, and let's define $\Lambda^{-1} = \Gamma$.

$$\begin{aligned} \text{MSE}(\hat{\mathbf{v}}_W, \mathbf{v}) &= Tr(\mathbb{V}(\hat{\mathbf{v}}_W)) = Tr(\mathbb{V}(S\mathbf{z}) = Tr(S\Sigma S^T) \\ &= Tr\left[(H^T\Gamma_1 H)^{-1}H^T\Gamma_2\Sigma\Gamma_3 H(H^T\Gamma_4 H)^{-1}\right]. \end{aligned} \tag{8}$$

We basically need to differentiate equation (8) wrt Γ, where Γ appears 4 times. The subscript is to keep track of which one we are differentiating wrt.

$$\begin{aligned} 2. &= \frac{\partial(tr\left[(H^T\Gamma_1H)^{-1}H^T\Gamma_2\Sigma\Gamma_3H(H^T\Gamma_4H)^{-1}\right])}{\partial\Gamma_2} = \frac{\partial(tr\left[\Gamma_2\Sigma\Gamma_3H(H^T\Gamma_4H)^{-2}H^T\right])}{\partial\Gamma_2} = \Sigma\Gamma_3H(H^T\Gamma_4H)^{-2}H^T \\ 3. &= \frac{\partial(tr\left[(H^T\Gamma_1H)^{-1}H^T\Gamma_2\Sigma\Gamma_3H(H^T\Gamma_4H)^{-1}\right])}{\partial\Gamma_3} = \frac{\partial(tr\left[\Gamma_3H(H^T\Gamma_4H)^{-2}H^T\Gamma_2\Sigma\right])}{\partial\Gamma_3} = H(H^T\Gamma_4H)^{-2}H^T\Gamma_2\Sigma \end{aligned} \tag{9}$$

For 1. and 4. things are more complicated so let's do it separately. Let's define

$$C = H^T\Gamma\Sigma\Gamma H(H^T\Gamma H)^{-1}$$

and

$$D = (H^T\Gamma H)^{-1}H^T\Gamma\Sigma\Gamma H.$$

Then we can rewrite the above as $Tr((H^T\Gamma_1H)^{-1}C)$, when we want to differentiate with respect to Γ_1 and as $Tr((H^T\Gamma_4H)^{-1}D)$, when we want to differentiate with respect to Γ_4. We can now differentiate using the same strategy for 1. and 4., that is

$$\begin{aligned} 1. &= \frac{\partial tr((H^T\Gamma_1H)^{-1}C)}{\partial\Gamma_1} = -H(H^T\Gamma H)^{-1}C(H^T\Gamma H)^{-1}H^T \\ 4. &= \frac{\partial tr((H^T\Gamma_4H)^{-1}D)}{\partial\Gamma_4} = -H(H^T\Gamma H)^{-1}D(H^T\Gamma H)^{-1}H^T. \end{aligned} \tag{10}$$

Combining all of them we get these 4 parts, where each term is in fact a $n \times n$ matrix

$$\begin{cases} 1. = & -H(H^T\Gamma H)^{-1}H^T\Gamma\Sigma\Gamma H(H^T\Gamma H)^{-2}H^T = -P(\Gamma) \\ 2. = & +\Sigma\Gamma H(H^T\Gamma H)^{-2}H^T = Q(\Gamma) \\ 3. = & +H(H^T\Gamma H)^{-2}H^T\Gamma\Sigma = Q(\Gamma)^T \\ 4. = & -H(H^T\Gamma H)^{-2}H^T\Gamma\Sigma\Gamma H(H^T\Gamma H)^{-1}H^T = -P(\Gamma)^T. \end{cases} \tag{11}$$

For minimizing our problem, we want to find Γ such that

$$\begin{aligned}\mathbf{0}_{(nxn)} &= -P(\Gamma) - P(\Gamma)^T + Q(\Gamma) + Q(\Gamma)^T \\ &= -H(H^T\Gamma H)^{-1}H^T\Gamma\Sigma\Gamma H(H^T\Gamma H)^{-2}H^T + \Sigma\Gamma H(H^T\Gamma H)^{-2}H^T \\ &+ H(H^T\Gamma H)^{-2}H^T\Gamma\Sigma - H(H^T\Gamma H)^{-2}H^T\Gamma\Sigma\Gamma H(H^T\Gamma H)^{-1}H^T\end{aligned} \tag{12}$$

We could start by solving $P(\Gamma) = Q(\Gamma)$. In fact if $P(\Gamma) = Q(\Gamma)$ then $P(\Gamma)^T = Q(\Gamma)^T$, which implies (12) satisfied.

Let's define $U_\Gamma = H(H^T\Gamma H)^{-1}H^T$ and $V_\Gamma = H(H^T\Gamma H)^{-2}H^T$, then

$$\begin{aligned}Q(\Gamma) &= \Sigma\Gamma V_\Gamma \\ P(\Gamma) &= U_\Gamma\Gamma\Sigma\Gamma V_\Gamma = U_\Gamma\Gamma Q(\Gamma).\end{aligned} \tag{13}$$

Therefore we need to solve

$$Q(\Gamma) = U_\Gamma\Gamma Q(\Gamma). \tag{14}$$

$Q(\Gamma)_{(n,n)}$ is not full rank, but if we find Γ that satisfies $U_\Gamma\Gamma = \mathbf{I}_{(n,n)}$, then we have equation (14) satisfied as well. It seems that we can just solve the following identity

$$\begin{aligned}&U_\Gamma\Gamma = \mathbf{I}_{(n,n)} \\ &H(H^T\Gamma H)^{-1}H^T\Gamma = \mathbf{I}_{(n,n)}.\end{aligned} \tag{15}$$

From the equation above we get that the sytstem of equations

$$\begin{cases} H(H^T\Gamma H)^{-1}H^T\Gamma = \mathbf{I}_{(n,n)} \\ H^T H(H^T\Gamma H)^{-1}H^T\Gamma = H^T \\ H^T\Gamma = H^T(H^T\Gamma H)(H^T H)^{-1}. \end{cases} \tag{16}$$

References

[1] Lumely, T. (2004). *Analysis of complex survey samples.* Journal of Statistical Software.