Description of methods for numerical result on concatenated decoders

Dhruva Raman

March 25, 2020

0.1 Long version

A linear decoder can regress the mouse's position against measured neural activity with reasonable accuracy, for data gathered on a single given day. However the expected vector of neural activity, given a known trajectory of mouse position, changes over days with a characteristic velocity. What about the direction of this change? Is it particularly conducive to downstream decoding of position by a slowly adapting decoder? We seek to answer this question by constructing a null model relating neural representations to synthetically generated 'mouse positions'. We construct the null model such that the accuracy of a linear decoder on inferring mouse position from neural representation matches the data. On each 'day', we change the null model. The velocity of change is such that the degradation of accuracy for a linear decoder trained on day d, when tested on day $d+1$, matches the data. We then test the performance of a static 'concatenated' decoder, trained on data from our shifting null model over multiple days, in predicting mouse position. As with the true data, as the number of days increases, the performance of the concatenated decoder decreases. We compare this decay in performance to that of a concatenated decoder trained on the true neural data, over multiple days.

For each mouse, we collected the mean-centred neural activity recorded on day one. This takes the form of a single matrix $X \in \mathbb{R}^{n \times d}$, where n represents the number of recorded neurons and d represents the number of datapoints. On day r, we generate synthetic 'observed data' $Y_r = m_r X + \epsilon_r$, where $m_r, \epsilon_r \in \mathbb{R}^{1 \times n}$. The vector ϵ_r is generated as scaled i.i.d. Gaussian noise (the scaling will be detailed subsequently). On day 1, m_1 is generated as a vector of uniform random variables on $[0, 1]$. It provides the 'ground-truth' relationship between neural activity X and the 'observed data' Y_1. This ground-truth model is updated each day.

Suppose we regressed Y_r against neural activity X. We would obtain regression coefficients

$\hat{m}_r$ satisfying

$$\hat{m}_r = \min_{v} \|Y_r - vX\|_2^2.$$

This corresponds to the r^{th} day best linear decoder for the null model. Now let

- $\mathcal{R}_w^2$ be the expected coefficient of multiple correlation of the predictor $\hat{m}_r X$ against the data Y_r. We will call this 'within-day $\mathcal{R}^2$'
- $\mathcal{R}_b^2$ be the expected coefficient of multiple correlation of the predictor $\hat{m}_r X$ against the data Y_{r+1}. We will call this 'between-day $\mathcal{R}^2$'.

We extracted the within/between-day $\mathcal{R}^2$ values of linear decoders trained on the true data. We averaged each of these over days, to give desired average within/between day $\mathcal{R}^2$ values for our null model to conform to.

We now present a method of stochastically updating m_r and generating ϵ_r, on each day, such that the $\mathcal{R}_w^2$ and $\mathcal{R}_b^2$ closely correspond to pre-defined values.

First note that $\|\hat{m}_r - m_r\|_2^2$ is very small in practice, as the number of datapoints d per day matches that of the true data, and is in the thousands for each mouse. We will assume that $\hat{m}_r = m_r$. In this case, we get

$$\mathcal{R}_w^2 = \frac{var(\epsilon_r)}{var(Y_r)} \tag{1a}$$

$$\mathcal{R}_b^2 = \frac{var([\Delta m_r]X + \epsilon)}{var(Y_r)} \tag{1b}$$

$$= \frac{var(\Delta m_r X) + var(\epsilon)}{var(Y_r)}, \tag{1c}$$

where $\Delta m_r := m_{r+1} - m_r$. By enforcing $\|m_{r+1}\|_2 = \|m_r\|_2$, we ensure that $var(Y_r)$ is constant over days, in expectation. This corresponds to the following constraint:

$$\Delta m_r^T m_r = \frac{-\|\Delta m_r\|_2^2}{2}.$$

This means that, for provided values of $\mathcal{R}_w^2$ and $\mathcal{R}_b^2$, we can solve equations (1) to get the corresponding desired values of $var(\Delta m_r X)$ and $var(\epsilon_r)$. We synthesise ϵ_r by generating a vector of i.i.d Gaussian random variables, and scaling them to obtain the desired variance.

To synthesise an appropriate Δm_r, note that

$$\begin{aligned} var(\Delta m_r X) &= \mathbb{E}[\Delta m_r X X^T \Delta m_r^T], \\ &= \|\Delta m_r\|_2^2 \frac{Tr(XX^T)}{nd}, \end{aligned}$$

as Δm_r is uncorrelated with the eigenvalues of X. This specifies the desired value $\|\Delta m_r\|_2^2$.

We now have a constraint on the norm of Δm_r, and a constraint on its' correlation with M. We generate an explicit value of Δm_r in two stages. We first generate a vector ν of i.i.d Gaussian random variables. We then decompose ν into components parallel and orthogonal to m_r. So we get

$$\nu = a + b$$
$$a = (\nu^T m_r) m_r.$$

We then take

$$\Delta m_r = c_1 a + c_2 b,$$

for some scalar constants c_1 and c_2. Finding the values of c_1 and c_2 that ensure the constraints on Δm_r is a well-posed problem, which we solve to obtain Δm_r.

1 Terse version (new)

For each animal we take the matrix $X \in \mathbb{R}^{n \times d}$ of mean-centered neural activity on day one. We relate this matrix to pseudo-observations of mouse position Y via a null model of the form $Y_r = m_r X + \epsilon_r$, where $m_r, \epsilon_r \in \mathbb{R}^{1 \times n}$. Note that r indexes days. The vector ϵ_r is generated as scaled i.i.d. Gaussian noise. We choose the scaling of ϵ_r so that the accuracy of a linear decoder trained on the data (X, Y_r) matches the average (over days) accuracy of a single day decoder trained on the true data. Next, we consider the choice of m_r. On day one, m_1 is generated as a vector of uniform random variables on $[0, 1]$. Given m_r, we desire an m_{r+1} that satisfies

- $\|m_{r+1}\|_2 = \|m_r\|_2$.
- The expected coefficient of multiple correlation of $Y_{r+1} = m_{r+1} X$ against the predictive model $m_r X$ (between day R^2) matches the average (over days) of the equivalent statistic generated from the true data.

To do this, we first generate a candidate $\Delta m'_r \in \mathbb{R}^{1 \times n}$ as a vector of i.i.d. white noise. The components of $\Delta m'_r$ orthogonal and parallel to m_r are then scaled so that $m_{r+1} = m_r + \Delta m_r$ satisfies the constraints above.