<u>**Supplemental Material**</u>

Ridge Shrinkage and the Effective Degrees of Freedom

Here we give a more thorough description of how ridge regression shrinks the model

coefficients and recount a rough derivation of the effective degrees of freedom, N_{df}, based

largely on a discussion originally presented in *(Hastie et al., 2001)*.

The singular value decomposition (SVD) of the centered neural input matrix is

given by

$$R = USV^T,$$ (S1)

where U and V are orthogonal matrices containing the singular vectors and S is a diagonal

matrix containing the nonzero singular values of R, where $s_1 \geq s_2 \geq \ldots \geq s_n \geq 0$. Using

Equation S1, the least squares fitted estimate, $\hat{X}^{LS}$, can also be expressed as a projection

of X onto the orthonormal basis U, that is

$$\begin{aligned} \hat{X}^{LS} = R\hat{B}^{LS} &= R(R^T R)^{-1} R^T X \\ &= UU^T X \end{aligned}$$ (S2)

Similarly the ridge fitted estimate, $\hat{X}^{ridge}$, can be re-expressed:

$$\begin{aligned} \hat{X}^{ridge} = R\hat{B}^{ridge} &= R(R^T R + \lambda I)^{-1} R^T X \\ &= US(S^2 + \lambda I)^{-1} SU^T X \\ &= \sum_{j=1}^{N} \frac{d_j^2}{d_j^2 + \lambda} u_j^T X \end{aligned}$$ (S3)

where u_j are the columns of U. Therefore, in addition to performing the orthogonal

projection as the least squares estimate does, ridge shrinks each coordinate of the

orthogonal projection of X by the factor $\dfrac{d_j^2}{d_j^2 + \lambda}$. Thus, the basis vectors that have the

smallest d_j^2 receive the largest amount of shrinkage, which notably correspond to

principal component directions that have the smallest sample variance. Hastie and

colleagues then define the effective degrees of freedom, N_{df}, as the trace of the projection

matrix, that is

$$
\begin{aligned}
N_{df} &= tr[R(R^T R + \lambda I)^{-1} R^T] \\
&= tr[US(S^2 + \lambda I)^{-1} SU^T] \\
&= \sum_{j=1}^{N} \frac{d_j^2}{d_j^2 + \lambda}
\end{aligned}
\tag{S4}
$$

This measure provided us with a convenient way to quantify the effective

complexity of the ridge model for each neural ensemble. In general, it provides a simple,

but mathematically sound method for determining a relevant (in a linear sense) set of

neural inputs to be used for training a variety of decoding algorithms for a neural

prosthetic application.

Discrete G-Kalman Filter Two-step Estimation

Estimation using the Kalman filter follows a well-known two-step recursive process,

consisting of an *a priori* time prediction followed by an *a posterior* measurement update.

This iterative prediction (Equation S5) and update process (Equation S6) is summarized

below:

$$
\begin{aligned}
\hat{x}_k^- &= A\hat{x}_{k-1} && \text{(\textit{a priori} estimate)} \\
P_k^- &= AP_{k-1}A^T + W && \text{(\textit{a priori} error covariance)}
\end{aligned}
\tag{S5}
$$

$$
\begin{aligned}
K_k &= P_k^- H^T \left(HP_k^- H^T + Q \right)^{-1} && \text{(Kalman gain update)} \\
\hat{x}_k &= \hat{x}_k^- + K_k \left(R_k - H\hat{x}_k^- \right) && \text{(\textit{a posterior} estimate)} \\
P_k &= (I - K_k H) P_k^- && \text{(\textit{a posterior} error covariance),}
\end{aligned}
\tag{S6}
$$

where W and Q are covariance matrices for the zero-mean Gaussian noise processes belonging to the process and observation models (i.e. Equations 6 and 7), respectively. P_k^- and P_k are covariance matrices for the *a priori* and *a posterior* estimate errors, e_k^- and e_k, and are defined as:

$$
\begin{aligned}
e_k^- &= x_k - \hat{x}_k^- & P_k^- &= E[e_k^- e_k^{-T}] \\
e_k &= x_k - \hat{x}_k & P_k &= E[e_k e_k^{T}]
\end{aligned}
$$
(S7)

The Kalman gain matrix of Equation S6, K_k, is optimal in the sense that it minimizes the *a posterior* error covariance P_k. A derivation of the Kalman gain is not provided here but can be obtained by minimizing the trace of P_k (which is equivalent to the MSE of the *a posterior* estimate) (Maybeck, 1979). Intuitively, the magnitude of the Kalman gain depends proportionally on the *a priori* error covariance P_k^- (i.e. the uncertainty in the *a priori* estimate) and inversely proportionally on the measurement noise Q (Gelb, 1974). Two limiting cases give insight into how the Kalman gain is adjusted at each time step to optimally combine the contributions of the process and observation equations. When uncertainty in the *a priori* estimate is very low, P_k^- and consequently K_k will approach zero, and therefore the *a posterior* estimate will rely entirely on the *a priori* process estimate, ignoring any measurement innovation ($R_k - H\hat{x}_k^-$) altogether. Conversely, when the measurement error, Q, is very small, K_k approaches H^{-1}, and as a result the *a posterior* estimate relies more heavily on the measurement innovation (Welch and Bishop, 2006).

Discrete G-Kalman Filter Stability

Figure S1*A-B* illustrates that both the Kalman gain, K_k, and the covariance matrix, P_k, quickly converge (via an exponential decrease) toward a stable asymptote by changing progressively less from one time step to the next (*k* to *k+1*) over the course of a trial. In Figure S1*C*, we plotted all of the coefficients in the Kalman gain matrix associated with position or velocity as function of time in the trial, again illustrating how K_k stabilizes quickly to steady-state values, in less than 1 second. A similar plot for the acceleration and target gain coefficients is illustrated in Figure S1*D*.

Note that during the early phases of a trajectory, it is probable that the G-Kalman filter does not optimally balance the contributions of the process and observation models, potentially resulting in somewhat unstable estimates. However, based on our cross-validated reconstruction results, we did not observe any substantial decrease in performance during these periods in the trajectory, and instead found these early estimates to be comparably reliable to those in later periods. In future experiments, we expect that a continuous pursuit task (in which multiple trajectories are executed in series to a sequence of randomly presented targets) will result in longer periods of continuous movement, (Wu et al., 2002; Pistohl et al., 2008) and undoubtedly enable the Kalman gain and covariance to operate at their steady-state values for a larger percentage of the time.
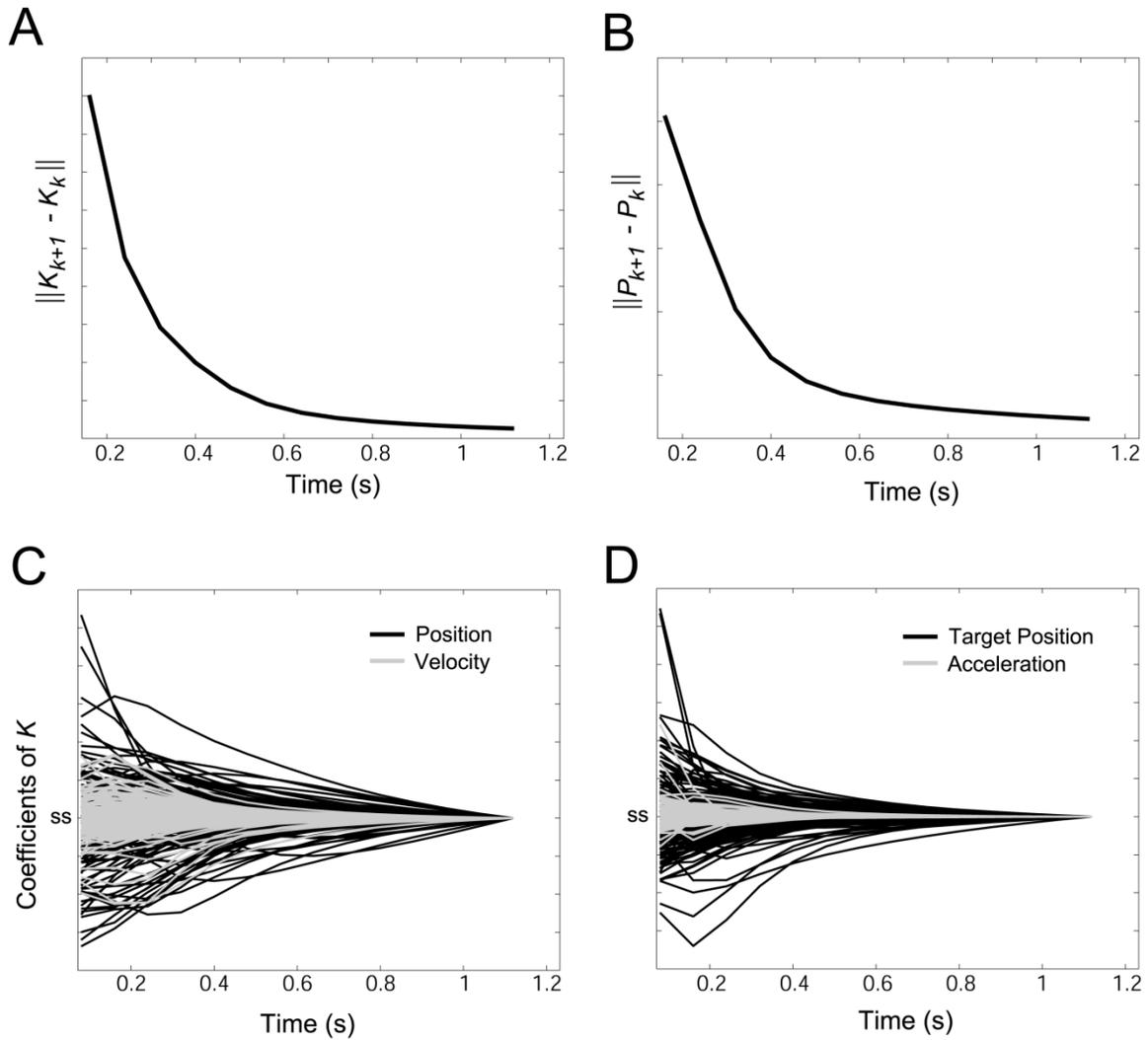
Figure S1. Stability analysis for G-Kalman filter. *A*, Plot of Frobenius norm of difference between consecutive Kalman gain matrices, illustrating that the Kalman gain changes exponentially less with elapsed time in the trajectory. *B*, Similarly, the covariance matrix also changes exponentially less with time (same format as *A*). *C-D*, Temporal evolution of Kalman gain coefficients for position and velocity (*C*) and target position and acceleration (*D*), showing that these coefficients rapidly toward their steady-state values (denoted as 'ss').

Supplemental References

Gelb A (1974) Applied Optimal Estimation. Cambridge, MA: MIT Press.

Hastie T, Tibshirani R, Friedman J (2001) The Elements of Statistical Learning. New York, NY: Springer-Verlag New York.

Maybeck PS (1979) Stochastic models, estimation, and control. New York, NY: Academic Press, Inc.

Pistohl T, Ball T, Schulze-Bonhage A, Aertsen A, Mehring C (2008) Prediction of arm movement trajectories from ECoG-recordings in humans. Journal of Neuroscience Methods 167:105-114.

Welch G, Bishop G (2006) An introduction to the Kalman filter. In: Tech. Rep. 95-041.

Wu W, Black MJ, Gao Y, Bienenstock E, Serruya M, Shaikhouni A, Donoghue JP (2002) Neural decoding of cursor motion using a kalman filter. Neural Information Processing Systems.