Dimensionality reduction techniques such as PCA are a cornerstone of analyzing high dimensional data. PCA involves an eigenvalue decomposition of the data covariance matrix, which produces ranked orthogonal dimensions that can be used to linearly project the data to lower dimension. The covariance matrix is but one choice of summary matrix that can be used for linear projections. When the data are time series, the same decomposition can be used on a linear description of the dynamics (instead of the data covariance) to obtain a different projection. This 'dynamical PCA' produces projections representing the largest eigenvalues of the linear dynamical system. Linear dynamical systems capture scaling and rotational aspects of the data (and combining scalings/rotations yields familiar features such as shear, projection, reflection, etc.). Dynamical PCA makes no distinction between scaling and rotation, seeking only directions of largest and most consistent dynamical activity. However, in some settings, one may hypothesize the existence of certain types of dynamics, and thus an algorithm is desired that can verify those dynamics. For example, it is common for neural circuits to generate oscillations (or rotations), and we seek projections that best capture these fundamental aspects of the neural response. Here we introduce jPCA, which specifically extracts projections of largest *rotational* dynamics. Using skew-symmetric matrices, the jPCA algorithm is an extension of fitting a linear dynamical system. It has a unique, closed-form solution that can be quickly computed. Importantly, like PCA and dynamical PCA, jPCA produces simple projection vectors - orthogonal linear directions - and so the interpretation of jPCA is identical to PCA or any other linear projection of the data. We motivate and present details of the method, and we discuss its importance in extracting rotational structure from electrophysiological data recorded from primate motor cortex.

Additional Details of the jPCA algorithm

We consider high dimensional time series data $\mathbf{x}(t) = [x_1(t), ..., x_n(t)]$, which we write as a matrix $X \in I\!R^{T \times n}$ (T time points, n data dimensions). Fitting a time-invariant linear dynamical system (defined as $\dot{\mathbf{x}}(t) = \mathbf{x}(t)M$, where $M \in I\!R^{n \times n}$) is simple least squares: we write $\dot{X} = XM$ and solve $M^* = \operatorname{argmin}_{M \in I\!R^{n \times n}} ||\dot{X} - XM||_F$ (often written in closed form as $M^* = (X^T X)^{-1} X^T \dot{X}$, or in MATLAB notation, $M^* = X \backslash \dot{X}$).

Importantly, this dynamics matrix M^* is a valid summary matrix describing the data, just like the data covariance $\Sigma = X^T X$ (and decomposition of Σ results in PCA). Just as with Σ, we can decompose M^* to produce a valid set of linear projections: we call this approach 'dynamical PCA.' While PCA ignores temporal information and extracts only those directions in n dimensional space that have the most variance, dynamical PCA extracts directions in the data that have consistent and strong linear dynamical properties.

$M^* \in I\!R^{n \times n}$ is the best general linear dynamical system describing X, without regard to scalings vs. rotations. Here, instead, we are interested only in the *rotational* aspects of the data. Every linear transformation M is some mixture of a symmetric matrix and a skew-symmetric matrix. The skew-symmetric matrix (also called anti-symmetric, written symbolically as $\mathbb{S}^{n \times n}$) is $M_{skew} = \frac{1}{2}(M - M^T)$. Such matrices satisfy $M_{skew} = -M_{skew}^T$, have purely imaginary eigenvalues (in complex conjugate pairs), and describe only rotations in the data.

Thus, whereas previously we solved $M^* = \operatorname{argmin}_{M \in I\!R^{n \times n}} ||\dot{X} - XM||_F$, for rotational systems we solve $M_{skew}^* = \operatorname{argmin}_{M \in \mathbb{S}^{n \times n}} ||\dot{X} - XM||_F$. This problem has a unique optimum that can be quickly solved in closed form using a novel matrix-vector reparameterization that we introduce. We call the algorithm solving this constrained problem (and then decomposing and projecting) jPCA. Crucially, just like regular PCA and dynamical PCA, jPCA produces a ranked set of orthonormal vectors that we can use to project our data. Again, the only difference is that, whereas in regular PCA we are interested in directions of highest variance, jPCA allows us to solve for highest frequency and consistency in *rotational* linear dynamical systems.

In working towards jPCA, we have developed three useful methods: dynamical PCA, jPCA, and symmetric PCA (the symmetric complement to jPCA). These enable exploration of a variety of structured activity, not simply the highest variance (traditional PCA). Finally, jPCA successfully projects the responses of tens to hundreds of neurons onto a plane that captures significant underlying rotational structure. These results and their neuroscientific implications are discussed in Churchland**, Cunningham**, et al, COSYNE 2011.