MATLAB CODE FOR PCA

by Jonathon Shlens (Salk Institute for Biological Sciences)

http://www.snl.salk.edu/~shlens/pub/notes/pca.pdf

This first version follows Section 5 (of above tutorial) by examining the covariance of the data set.

```
function [signals,PC,V] = pca1(data)
% PCA1: Perform PCA using covariance.
% data - MxN matrix of input data
% (M dimensions, N trials)
% signals - MxN matrix of projected data
% PC - each column is a PC
% V - Mx1 matrix of variances
[M,N] = size(data);
% subtract off the mean for each dimension
mn = mean(data,2);
data = data - repmat(mn,1,N);
% calculate the covariance matrix
covariance = 1 / (N-1) * data * data';
% find the eigenvectors and eigenvalues
[PC, V] = eig(covariance);
% extract diagonal of matrix as vector
V = diag(V);
% sort the variances in decreasing order
[junk, rindices] = sort(-1*V);
V = V(rindices);
```

```
PC = PC(:,rindices);
% project the original data set
signals = PC' * data;
```

This second version follows Section 6 computing PCA through SVD.

```
function [signals,PC,V] = pca2(data)
% PCA2: Perform PCA using SVD.
% data - MxN matrix of input data
% (M dimensions, N trials)
% signals - MxN matrix of projected data
% PC - each column is a PC
% V - Mx1 matrix of variances
[M,N] = size(data);
% subtract off the mean for each dimension
mn = mean(data,2);
data = data - repmat(mn,1,N);
% construct the matrix Y
Y = data' / sqrt(N-1);
% SVD does it all
[u,S,PC] = svd(Y);
% calculate the variances
S = diag(S);
V = S .* S;
% project the original data
signals = PC' * data;
```