
Generalizable Singular Value Decomposition for Ill-posed Datasets

Ulrik Kjems **Lars K. Hansen**
Department of Mathematical Modelling
Technical University of Denmark
DK-2800 Kgs. Lyngby, Denmark
uk,lkhansen@imm.dtu.dk

Stephen C. Strother
PET Imaging Service
VA medical center
Minneapolis
steve@pet.med.va.gov

Abstract

We demonstrate that statistical analysis of ill-posed data sets is subject to a bias, which can be observed when projecting independent test set examples onto a basis defined by the training examples. Because the training examples in an ill-posed data set do not fully span the signal space the observed training set variances in each basis vector will be too high compared to the average variance of the test set projections onto the same basis vectors. On basis of this understanding we introduce the Generalizable Singular Value Decomposition (GenSVD) as a means to reduce this bias by re-estimation of the singular values obtained in a conventional Singular Value Decomposition, allowing for a generalization performance increase of a subsequent statistical model. We demonstrate that the algorithm successfully corrects bias in a data set from a functional PET activation study of the human brain.

1 Ill-posed Data Sets

An ill-posed data set has more dimensions in each example than there are examples. Such data sets occur in many fields of research typically in connection with image measurements. The associated statistical problem is that of extracting structure from the observed high-dimensional vectors in the presence of noise. The statistical analysis can be done either supervised (i.e. modelling with target values: classification, regression) or unsupervised (modelling with no target values: clustering, PCA, ICA). In both types of analysis the ill-posedness may lead to immediate problems if one tries to apply conventional statistical methods of analysis, for example the empirical covariance matrix is prohibitively large and will be rank-deficient.

A common approach is to use Singular Value Decomposition (SVD) or the analogue Principal Component Analysis (PCA) to reduce the dimensionality of the data. Let the N observed i -dimensional samples \mathbf{x}_j , $j = 1 \dots N$, collected in the data matrix $\mathbf{X} = [\mathbf{x}_1 \dots \mathbf{x}_N]$ of size $I \times N$, $I > N$. The SVD-theorem states that such a matrix can be decomposed as

$$\mathbf{X} = \mathbf{U}\mathbf{\Lambda}\mathbf{V}^\top, \tag{1}$$

where \mathbf{U} is a matrix of the same size as \mathbf{X} with orthogonal basis vectors spanning the space of \mathbf{X} , so that $\mathbf{U}^\top \mathbf{U} = \mathbf{I}_{N \times N}$. The square matrix $\mathbf{\Lambda}$ contains the singular values in the diagonal, $\mathbf{\Lambda} = \text{diag}(\lambda_1, \dots, \lambda_N)$, which are ordered and positive $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_N \geq 0$, and \mathbf{V} is $N \times N$ and orthogonal $\mathbf{V}^\top \mathbf{V} = \mathbf{I}_N$. If there is a mean value significantly different from zero it may at times be advantageous to perform the above analysis on mean-subtracted data, i.e. $\mathbf{X} - \bar{\mathbf{X}} = \mathbf{U} \mathbf{\Lambda} \mathbf{V}^\top$ where columns of $\bar{\mathbf{X}}$ all contain the mean vector $\bar{\mathbf{x}} = \sum_j \mathbf{x}_j / N$.

Each observation \mathbf{x}_j can be expressed in coordinates in the basis defined by the vectors of \mathbf{U} with no loss of information [Lautrup et al., 1995]. A change of basis is obtained by $\mathbf{q}_j = \mathbf{U}^\top \mathbf{x}_j$ as the orthogonal basis rotation

$$\mathbf{Q} = [\mathbf{q}_1 \dots \mathbf{q}_N] = \mathbf{U}^\top \mathbf{X} = \mathbf{U}^\top \mathbf{U} \mathbf{\Lambda} \mathbf{V}^\top = \mathbf{\Lambda} \mathbf{V}^\top. \quad (2)$$

Since \mathbf{Q} is only $N \times N$ and $N \ll I$, \mathbf{Q} is a compact representation of the data. Having now N examples of N dimension we have reduced the problem to a marginally ill-posed one. To further reduce the dimensionality, it is common to retain only a subset of the coordinates, e.g. the top P coordinates ($P < N$) and the supervised or unsupervised model can be formed in this smaller but now well-posed space.

So far we have considered the procedure for modelling from a training set. Our hope is that the statistical description generalizes well to new examples proving that it is a good description of the generating process. The model should, in other words, be able to perform well on a new example, \mathbf{x}^* , and in the above framework this would mean the predictions based on $\mathbf{q}^* = \mathbf{U}^\top \mathbf{x}^*$ should generalize well. We will show in the following, that in general, *the distribution of the test set projection \mathbf{q}^* is quite different from the statistics of the projections of the training examples \mathbf{q}_j* . It has been noted in previous work [Hansen and Larsen, 1996, Roweis, 1998, Hansen et al., 1999] that PCA/SVD of ill-posed data does not by itself represent a probabilistic model where we can assign a likelihood to a new test data point, and procedures have been proposed which make this possible. In [Bishop, 1999] PCA has been considered in a Bayesian framework, but does not address the significant bias of the variance in training set projections in ill-posed data sets. In [Jackson, 1991] an asymptotic expression is given for the bias of eigen-values in a sample covariance matrix, but this expression is valid only in the well-posed case and is not applicable for ill-posed data.

1.1 Example

Let the signal source be I -dimensional multivariate Gaussian distribution $\mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$ with a covariance matrix where the first K eigen-values equal σ^2 and the last $I - K$ are zero, so that the covariance matrix has the decomposition

$$\mathbf{\Sigma} = \sigma^2 \mathbf{Y} \mathbf{D} \mathbf{Y}^\top, \quad \mathbf{D} = \text{diag}(1, \dots, 1, 0, \dots, 0), \quad \mathbf{Y}^\top \mathbf{Y} = \mathbf{I} \quad (3)$$

Our N samples of the distribution are collected in the matrix $\mathbf{X} = [\mathbf{x}_{ij}]$ with the SVD

$$\mathbf{X} = \mathbf{U} \mathbf{\Lambda} \mathbf{V}^\top, \quad \mathbf{\Lambda} = \text{diag}(\lambda_1, \dots, \lambda_N) \quad (4)$$

and the representation of the N examples in the N basis vector coordinates defined by \mathbf{U} is $\mathbf{Q} = [\mathbf{q}_{ij}] = \mathbf{U}^\top \mathbf{X} = \mathbf{\Lambda} \mathbf{V}^\top$. The total variance per training example is

$$\begin{aligned} \frac{1}{N} \sum_{i,j} x_{ij}^2 &= \frac{1}{N} \text{Tr}(\mathbf{X}^\top \mathbf{X}) = \frac{1}{N} \text{Tr}(\mathbf{V} \mathbf{\Lambda} \mathbf{U}^\top \mathbf{U} \mathbf{\Lambda} \mathbf{V}^\top) = \frac{1}{N} \text{Tr}(\mathbf{V} \mathbf{\Lambda}^2 \mathbf{V}^\top) \\ &= \frac{1}{N} \text{Tr}(\mathbf{V} \mathbf{V}^\top \mathbf{\Lambda}^2) = \frac{1}{N} \text{Tr}(\mathbf{\Lambda}^2) = \frac{1}{N} \sum_i \lambda_i^2 \end{aligned} \quad (5)$$

Note that this variance is the same in the \mathbf{U} -basis coordinates:

$$\frac{1}{N} \sum_{i,j} q_{ij}^2 = \frac{1}{N} \text{Tr}(\mathbf{Q}^\top \mathbf{Q}) = \frac{1}{N} \text{Tr}(\mathbf{V} \mathbf{\Lambda}^2 \mathbf{V}^\top) = \frac{1}{N} \sum_i \lambda_i^2 \quad (6)$$

We can derive the expected value of this variance:

$$\left\langle \frac{1}{N} \sum_{i,j} x_{j,i}^2 \right\rangle = \left\langle \sum_i x_{1,i}^2 \right\rangle = \langle \mathbf{x}_1^\top \mathbf{x}_1 \rangle = \text{Tr} \mathbf{\Sigma} = \sigma^2 K \quad (7)$$

Now, consider a test example $\mathbf{x}^* \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$ with the projection $\mathbf{q}^* = \mathbf{U}^\top \mathbf{x}^*$ which will have the average total variance

$$\begin{aligned} \sum_i q_i^{*2} &= \left\langle \text{Tr}[(\mathbf{U}^\top \mathbf{x}^*)^\top (\mathbf{U}^\top \mathbf{x}^*)] \right\rangle = \text{Tr}[\langle \mathbf{x}^* \mathbf{x}^{*\top} \rangle \mathbf{U} \mathbf{U}^\top] \\ &= \text{Tr}[\mathbf{\Sigma} \mathbf{U} \mathbf{U}^\top] = \text{Tr}[\mathbf{D} \mathbf{U} \mathbf{U}^\top] = \sigma^2 \min(N, K) \end{aligned} \quad (8)$$

In summary, this means that the orthogonal basis \mathbf{U} computed from the training set spans all the variance in the training set but fails to do so on the test examples when $N < K$, i.e. for ill-posed data. The training set variance is $K/N\sigma^2$ on average per coordinate, compared to σ^2 for the test examples. So which of the two variances is ‘‘correct’’? From a modelling point of view, the variance from the test example tells us the true story, so the training set variance should be regarded as biased. This suggests that the training set singular values should be corrected for this bias, in the above example by re-estimating the training set projections using $\hat{\mathbf{Q}} = \sqrt{N/K} \mathbf{Q}$.

In the more general case we do not know K , and the true covariance may have an arbitrary eigen-spectrum. The GenSVD algorithm below is a more general algorithm for correcting for the training set bias.

2 The GenSVD Algorithm

The data matrix consists of N statistically independent samples $\mathbf{X} = [\mathbf{x}_1 \dots \mathbf{x}_N]$ so \mathbf{X} is size $I \times N$, and each column of \mathbf{X} is assumed multivariate Gaussian, $\mathbf{x}_j \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$ and is ill-posed with $\text{rank} \mathbf{\Sigma} > N$.

With the SVD $\mathbf{X} = \mathbf{U}_\theta \mathbf{\Lambda}_\theta \mathbf{V}_\theta^\top$, we now make the approximation that \mathbf{U}_θ contains an actual subset of the true eigen-vectors of $\mathbf{\Sigma}$

$$\mathbf{\Sigma} = \begin{bmatrix} \mathbf{U}_\theta & \mathbf{U}_\perp \end{bmatrix} \begin{bmatrix} \mathbf{\Lambda}^2 & \mathbf{0} \\ \mathbf{0} & \mathbf{\Lambda}_\perp^2 \end{bmatrix} \begin{bmatrix} \mathbf{U}_\theta & \mathbf{U}_\perp \end{bmatrix} \quad (9)$$

where we have collected the remaining (unspanned by \mathbf{X}) eigen-vectors and values in \mathbf{U}_\perp and $\mathbf{\Lambda}_\perp^2$, satisfying $\mathbf{U}_\perp^\top \mathbf{U}_\perp = \mathbf{I}$ and $\mathbf{U}_\theta^\top \mathbf{U}_\perp = \mathbf{0}$. The unknown ‘true’ eigen-values corresponding to the observed eigen-vectors are collected in $\mathbf{\Lambda} = \text{diag}(\lambda_1, \dots, \lambda_N)$, which are the values we try to estimate in the following.

It should be noted that a direct estimation of $\mathbf{\Sigma}$ using $\hat{\mathbf{\Sigma}} = \frac{1}{N} \mathbf{X} \mathbf{X}^\top$ yields $\hat{\mathbf{\Sigma}} = \frac{1}{N} \mathbf{U}_\theta \mathbf{\Lambda}_\theta \mathbf{V}_\theta^\top \mathbf{V}_\theta \mathbf{\Lambda}_\theta \mathbf{U}_\theta^\top = \frac{1}{N} \mathbf{U}_\theta \mathbf{\Lambda}_\theta^2 \mathbf{U}_\theta^\top$, i.e., the nonzero eigen-vectors and values of $\hat{\mathbf{\Sigma}}$ is \mathbf{U}_θ and $\mathbf{\Lambda}_\theta$.

The distribution of test samples \mathbf{x}^* inside the space spanned by \mathbf{U}_θ is

$$\mathbf{U}_\theta^\top \mathbf{x} \sim \mathcal{N}(\mathbf{0}, \mathbf{U}_\theta^\top \mathbf{\Sigma} \mathbf{U}_\theta) = \mathcal{N}(\mathbf{0}, \mathbf{\Lambda}^2) \quad (10)$$

The problem is that U_0 and the examples \mathbf{x}_j are not independent, so $U_0^\top \mathbf{x}_j$ is biased, e.g. the SVD estimate $\frac{1}{N} \Lambda_0^2$ of Λ^2 assigns all variance to lie within U_0 .

The GenSVD algorithm bypasses this problem by, for each example, computing a basis on all other examples, estimating the variances in Λ^2 in a leave-one-out manner. Consider

$$\mathbf{z}_j = U_0^\top \mathbf{B}_{-j} \mathbf{B}_{-j}^\top \mathbf{x}_j \quad (11)$$

where we introduce the notation \mathbf{X}_{-j} for the matrix of all examples except the j 'th, and this matrix is decomposed as $\mathbf{X}_{-j} = \mathbf{B}_{-j} \Lambda_{-j} \mathbf{V}_{-j}^\top$. The operation $\mathbf{B}_{-j} \mathbf{B}_{-j}^\top \mathbf{x}_j$ projects the example onto the basis defined by the remaining examples, and back again, so it 'strips' off the part of signal space which is special for \mathbf{x}_j which could be signal which does not generalize across examples.

Since \mathbf{B}_{-j} and \mathbf{x}_j are independent $\mathbf{B}_{-j}^\top \mathbf{x}_j$ has the same distribution as the projection of a test example \mathbf{x}^* , $\mathbf{B}_{-j}^\top \mathbf{x}^*$. Thus, $\mathbf{B}_{-j} \mathbf{B}_{-j}^\top \mathbf{x}_j$ and $\mathbf{B}_{-j} \mathbf{B}_{-j}^\top \mathbf{x}^*$ have the same distribution as well. Now, since $\text{span } \mathbf{B}_{-j} = \text{span } \mathbf{X}_{-j}$ and $\text{span } U_0 = \text{span} [\mathbf{X}_{-j} \mathbf{x}_j]$ we have that $\text{span } \mathbf{B}_{-j} \subseteq \text{span } U_0$ so we see that \mathbf{z}_j and $U_0^\top \mathbf{B}_{-j} \mathbf{B}_{-j}^\top \mathbf{x}^*$ are identically distributed. This means that \mathbf{z}_j has the covariance $U_0^\top \mathbf{B}_{-j} \mathbf{B}_{-j}^\top \Sigma \mathbf{B}_{-j} \mathbf{B}_{-j}^\top U_0$ and using Eq. (9) and that $U_\perp^\top \mathbf{B}_{-j} = \mathbf{0}$ (since $U_\perp^\top U_0 = \mathbf{0}$) we get

$$\mathbf{z}_j \sim \mathcal{N}(\mathbf{0}, U_0^\top \mathbf{B}_{-j} \mathbf{B}_{-j}^\top U_0 \Lambda U_0^\top \mathbf{B}_{-j} \mathbf{B}_{-j}^\top U_0) \quad (12)$$

We note that this distribution is degenerate because the covariance is of rank $N-1$. For a sample \mathbf{z}_j from the above distribution we have that

$$U_0^\top \mathbf{B}_{-j} \mathbf{B}_{-j}^\top U_0 \mathbf{z}_j = U_0^\top \mathbf{B}_{-j} \mathbf{B}_{-j}^\top U_0 U_0^\top \mathbf{B}_{-j} \mathbf{B}_{-j}^\top \mathbf{x}_j = U_0^\top \mathbf{B}_{-j} \mathbf{B}_{-j}^\top \mathbf{x}_j = \mathbf{z}_j \quad (13)$$

As a second approximation, assume that the observed \mathbf{z}_j are independent so that we can write the likelihood of Λ

$$\begin{aligned} -\log L(\Lambda^2) &\simeq \sum_j \log \left[(2\pi)^{N/2} \left| (U_0^\top \mathbf{B}_{-j}) (\mathbf{B}_{-j}^\top U_0) \Lambda^2 (U_0^\top \mathbf{B}_{-j}) (\mathbf{B}_{-j}^\top U_0) \right|^{1/2} \right] \\ &\quad + \frac{1}{2} \sum_j \mathbf{z}_j^\top (U_0^\top \mathbf{B}_{-j}) (\mathbf{B}_{-j}^\top U_0) \Lambda^{-2} (U_0^\top \mathbf{B}_{-j}) (\mathbf{B}_{-j}^\top U_0) \mathbf{z}_j \\ &\simeq c + \frac{N}{2} \sum_j \log \lambda_j^2 + \frac{1}{2} \sum_j \mathbf{z}_j^\top \Lambda^{-2} \mathbf{z}_j \end{aligned} \quad (14)$$

where we have used Eq. (13) and that the determinant¹ is approximated by $|\Lambda^2|$. This above expression is maximized when

$$\hat{\lambda}_i^2 = \frac{1}{N} \sum_j z_{ij}^2. \quad (15)$$

The GenSVD of \mathbf{X} is then $\mathbf{X} = U_0 \hat{\Lambda} \mathbf{V}^\top$, $\hat{\Lambda} = \text{diag}(\hat{\lambda}_1, \dots, \hat{\lambda}_N)$.

In practice, using Eq. (11) directly to compute an SVD of the matrix \mathbf{X}_{-j} for each example is computationally demanding. It is possible to compute \mathbf{z}_j in a more efficient two-level procedure with the following algorithm:

$$\text{Compute } U_0 \Lambda_0 \mathbf{V}_0^\top = \text{svd}(X) \text{ and } \mathbf{Q}_0 = [\mathbf{q}_j] = \Lambda_0 \mathbf{V}_0^\top$$

¹Since \mathbf{z}_j is degenerate, we define the likelihood over the space where \mathbf{z}_j occur, i.e. the determinant in Eq. 14 should be read as 'the product of non-zero eigenvalues'.

```

foreach  $j = 1 \dots N$ 
  Compute  $\mathbf{B}_{-j} \mathbf{\Lambda}_{-j} \mathbf{V}_{-j}^\top = \text{svd}(\mathbf{Q}_{-j})$ 
   $\mathbf{z}_j = \mathbf{B}_{-j} \mathbf{B}_{-j}^\top \mathbf{q}_j$ 
 $\hat{\lambda}_i^2 = \frac{1}{N} \sum_j z_{ij}^2$ 

```

If the data has a mean value that we wish to remove prior to the SVD it is important that this is done within the GenSVD algorithm. Consider a centered matrix $\mathbf{X}_c = \mathbf{X} - \bar{\mathbf{X}}$ where $\bar{\mathbf{X}}$ contains the mean $\bar{\mathbf{x}}$ replicated in all N columns. The signal space in \mathbf{X}_c is now corrupted because each centered example will contain a component of all examples, which means the 'stripping' of signal components not spanned by other examples no longer works: $\mathbf{B}_{-j}^\top \mathbf{x}_j$ is no longer distributed like $\mathbf{B}_{-j}^\top \mathbf{x}^*$. This suggests the alternative algorithm for data with removal of mean component:

```

Compute  $\mathbf{U}_0 \mathbf{\Lambda}_0 \mathbf{V}_0^\top = \text{svd}(X)$  and  $\mathbf{Q}_0 = [\mathbf{q}_j] = \mathbf{\Lambda}_0 \mathbf{V}_0^\top$ 
foreach  $j = 1 \dots N$ 
   $\bar{\mathbf{q}}_{-j} = \frac{1}{N-1} \sum_{j' \neq j} \mathbf{q}_{j'}$ 
  Compute  $\mathbf{B}_{-j} \mathbf{\Lambda}_{-j} \mathbf{V}_{-j}^\top = \text{svd}(\mathbf{Q}_{-j} - \bar{\mathbf{Q}}_{-j})$ 
   $\mathbf{z}_j = \mathbf{B}_{-j} \mathbf{B}_{-j}^\top (\mathbf{q}_j - \bar{\mathbf{q}}_{-j})$ 
 $\hat{\lambda}_i^2 = \frac{1}{N-1} \sum_j z_{ij}^2$ 

```

Finally, note that it is possible to leave out more than one example at a time if the data is independent only in block, i.e. let \mathbf{Q}_{-k} would be \mathbf{Q}_0 with the k 'th block left out.

Example With PET Scans

We compared the performance of GenSVD to conventional SVD on a functional [^{15}O] water PET activation study of the human brain. The study consisted of 18 subjects, who were scanned four times while tracing a star-shaped maze with a joy-stick with visual feedback, in total 72 scans of dimension ~ 25000 spatial voxels. After the second scan, the visual feedback was mirrored, and the subject accommodated to and learned the new control environment during the last two scans. Scans were normalized by 1) dividing each scan by the average voxel value measured inside a brain mask and 2) for each scan subtracting the average scan for that subject thereby removing subject effects and 3) intra and inter-subject normalization and transformation using rigid body reorientation and affine linear transformations respectively. Voxels inside aforementioned brain mask were arranged in the data matrix with one scan per column.

Figure 1 shows the results of an SVD decomposition compared to GenSVD. Each marker represents one scan and the glyphs indicate scan number out of the four (circle-square-star-triangle). The ellipses indicate the mean and covariances of the projections in each scan number. The 32 scans from eight subjects were used as a training set and 40 scans from the remaining 10 subjects for testing. The training set projections are filled markers, test-set projections onto the basis defined by the training set are open markers (i.e. we plot the first two columns of $\mathbf{U}_0 \mathbf{\Lambda}_0$ for SVD and of $\mathbf{U}_0 \hat{\mathbf{\Lambda}}$ for GenSVD). We see that there is a clear difference in variance in the train- and test-examples, which is corrected quite well by GenSVD. The lower plot in Figure 1 shows the singular values for the PET data set. We see that GenSVD estimates are much closer to the actual test projection standard deviations than the SVD singular values.

3 Conclusion

We have demonstrated that projection of ill-posed data sets onto a basis defined by the same examples introduces a significant bias on the observed variance when comparing to projections of test examples onto the same basis. The GenSVD algorithm has been presented as a tool for correcting for this bias using a leave-one-out re-estimation scheme, and a computationally efficient implementation has been proposed.

We have demonstrated that the method works well on an ill-posed real-world data set, where the distribution of the GenSVD-corrected training test set projections matched the distribution of the observed test set projections far better than the uncorrected training examples. This allows a generalization performance increase of a subsequent statistical model, in the case of both supervised and unsupervised models.

Acknowledgments

This work was supported partly by the Human Brain Project grant P20 MH57180, the Danish Research councils for the Natural and Technical Sciences through the Danish Computational Neural Network Center (CONNECT) and the Technology Center Through Highly Oriented Research (THOR).

References

- [Bishop, 1999] Bishop, C. (1999). Bayesian pca. In Kearns, M. S., Solla, S. A., and Cohn, D. A., editors, *Advances in Neural Information Processing Systems*, volume 11. The MIT Press.
- [Hansen et al., 1999] Hansen, L., Larsen, J., Nielsen, F., Strother, S., Rostrup, E., Savoy, R., Lange, N., Sidtis, J., Svarer, C., and Paulson, O. (1999). Generalizable patterns in neuroimaging: How many principal components? *NeuroImage*, 9:534–544.
- [Hansen and Larsen, 1996] Hansen, L. K. and Larsen, J. (1996). Unsupervised learning and generalization. In *Proceedings of IEEE International Conference on Neural Networks*, pages 25–30.
- [Jackson, 1991] Jackson, J. E. (1991). *A User's Guide to Principal Components*. Wiley Series on Probability and Statistics, John Wiley and Sons.
- [Lautrup et al., 1995] Lautrup, B., Hansen, L. K., Law, I., Mørch, N., Svarer, C., and Strother, S. (1995). Massive weight sharing: A cure for extremely ill-posed problems. In Hermann, H. J., Wolf, D. E., and Pöppel, E. P., editors, *Proceedings of Workshop on Supercomputing in Brain Research: From Tomography to Neural Networks: From tomography to neural networks*, HLRZ, KFA Jülich, Germany, pages 137–148. World Scientific.
- [Roweis, 1998] Roweis, S. (1998). Em algorithms for pca and spca. In Jordan, M. I., Kearns, M. J., and Solla, S. A., editors, *Advances in Neural Information Processing Systems*, volume 10. The MIT Press.

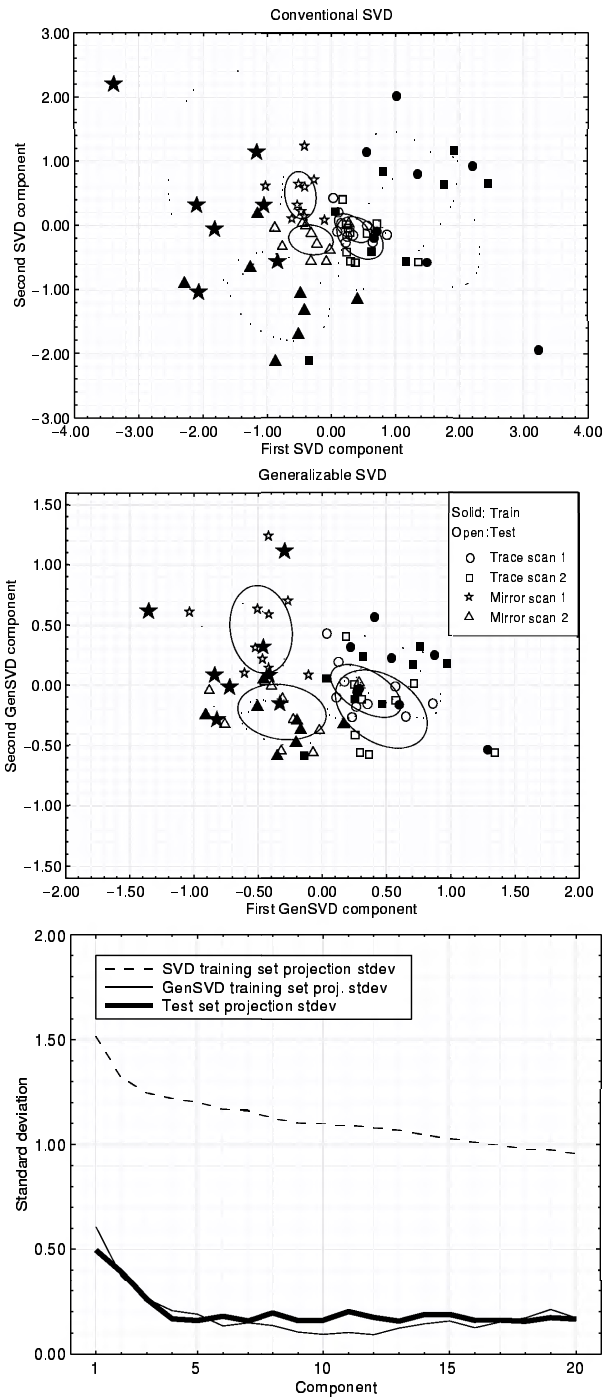


Figure 1: Projections of PET data in SVD and GenSVD. Each subject's four scans are indicated by: circle, square, star, triangle. Training set scans are marked with filled glyphs and test set with open glyphs. Solid and dotted Ellipses indicate test/train covariance per scan number. The third plot shows the standard deviations for the training and test set for SVD and GenSVD projections.