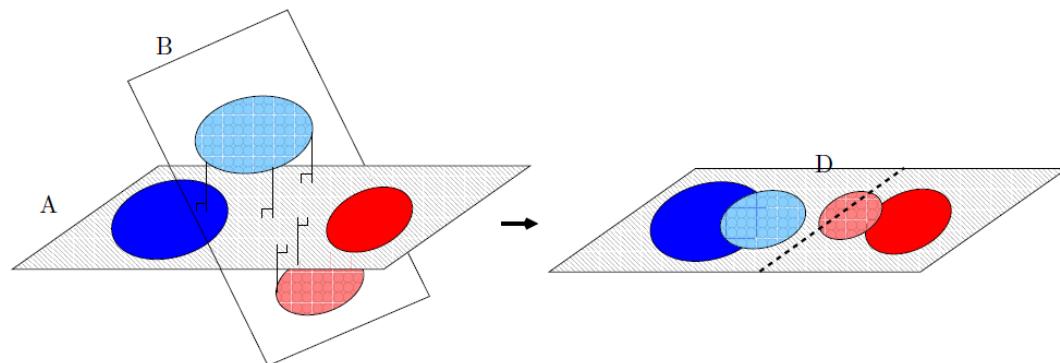


Variance inflation

Lars Kai Hansen

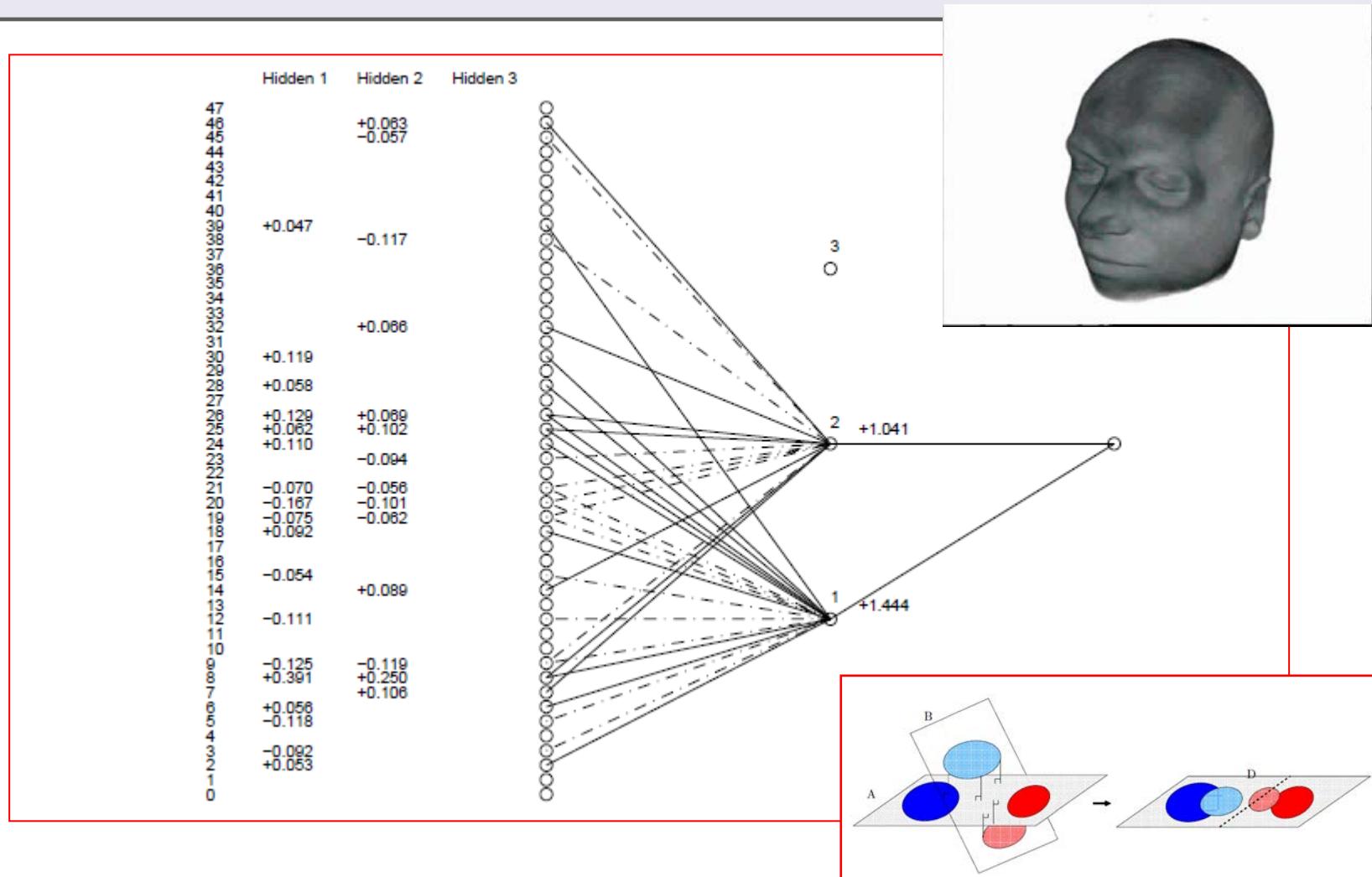
DTU Compute
Technical University of Denmark



Co-workers:
Trine Abrahamsen, Ulrik Kjems, Stephen Strother

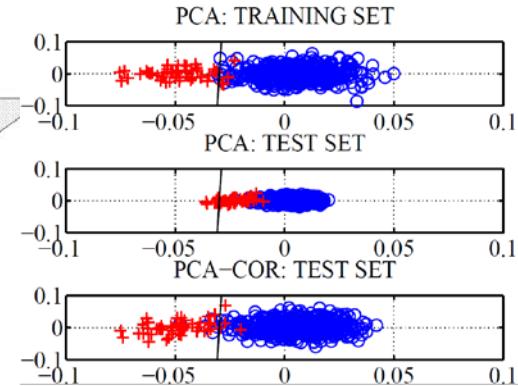
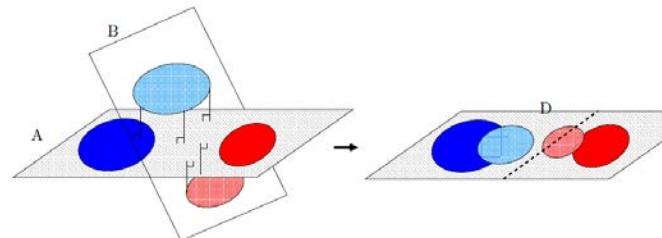
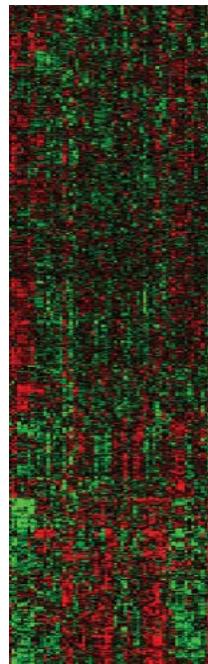


"Deep" network for decoding PET brain scans (1994)



Lautrup, B., Hansen, L. K., Law, I., Mørch, N., Svarer, C. L. A. U. S., & Strother, S. C. (1994). Massive weight sharing: a cure for extremely ill-posed problems. In *Workshop on supercomputing in brain research: From tomography to neural networks* (pp. 137-144).

OUTLINE



Variance inflation in PCA kPCA Linear regression and SVMs

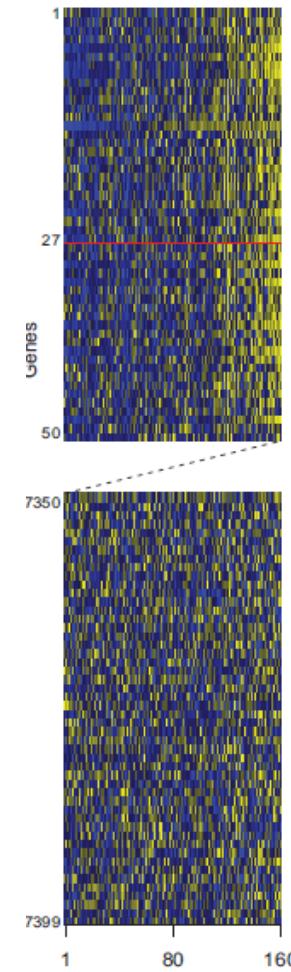
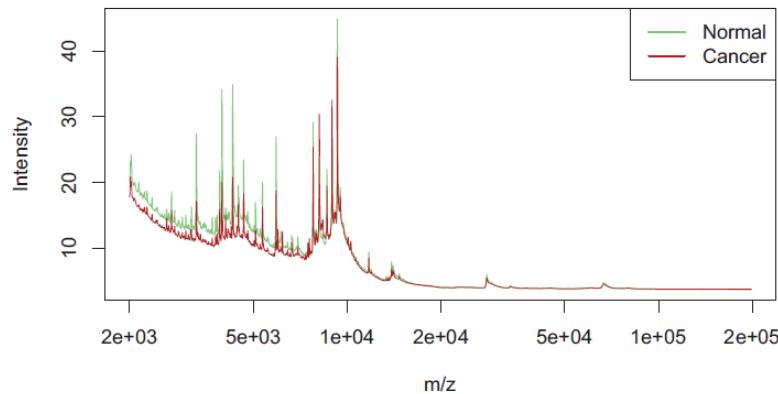
D

D sensors
N samples

$D \gg N$

N

High dimensions – small samples ($D \gg N$)



"HDLSS" high dimension, low sample size (Hall 2005, Ahn et al, 2007)

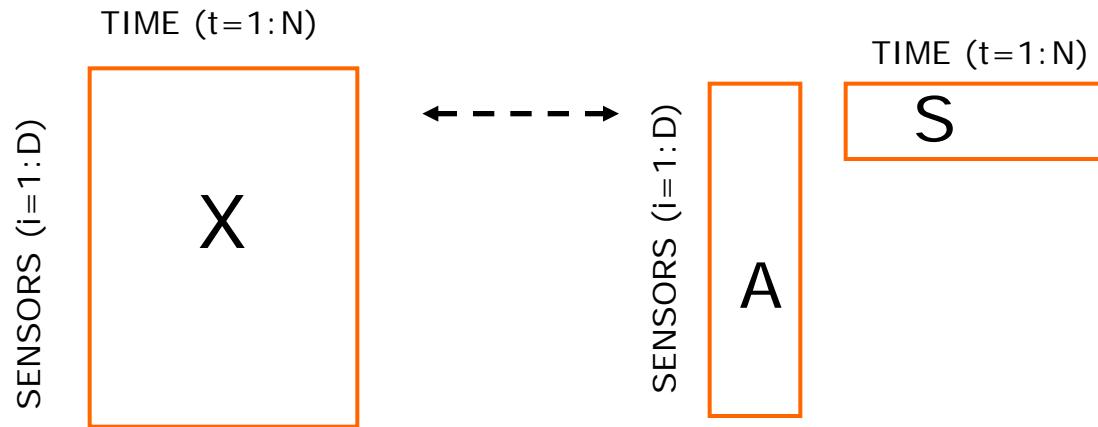
"Large p, small n" (West, 2003), "Curse of dimensionality" (Occam, 1350)

"Large underdetermined systems" (Donoho, 2001)

"Ill-posed data sets" (Kjems, Strother, LKH, 2001)

Factor models

Represent a datamatrix by a low-dimensional approximation



$$X(i,t) \approx \sum_{k=1}^K A(i,k)S(k,t)$$

Unsupervised learning: Factor analysis generative model

$$\mathbf{x} = \mathbf{As} + \boldsymbol{\varepsilon}, \quad \boldsymbol{\varepsilon} \sim N(\mathbf{0}, \Sigma)$$

$$p(\mathbf{x} | \mathbf{A}, \boldsymbol{\theta}) = \int p(\mathbf{x} | \mathbf{A}, \mathbf{s}, \Sigma) p(\mathbf{s} | \boldsymbol{\theta}) d\mathbf{s}$$

$$p(\mathbf{x} | \mathbf{A}, \mathbf{s}, \Sigma) = [2\pi\Sigma]^{-1/2} e^{-\frac{1}{2}(\mathbf{x}-\mathbf{As})^T \Sigma^{-1} (\mathbf{x}-\mathbf{As})}$$

S known:	GLM
(1-A) ⁻¹ sparse:	SEM
S,A positive:	NMF

Source distribution:
PCA: ... normal
ICA: ... other
IFA: ... Gauss. Mixt.
kMeans: .. binary

$$\text{PCA: } \Sigma = \sigma^2 \cdot \mathbf{1},$$

$$\text{FA: } \Sigma = \mathbf{D}$$

Højen-Sørensen, Winther, Hansen,
Neural Computation (2002),
Neurocomputing (2002)

Matrix factorization: SVD/PCA, NMF, Clustering

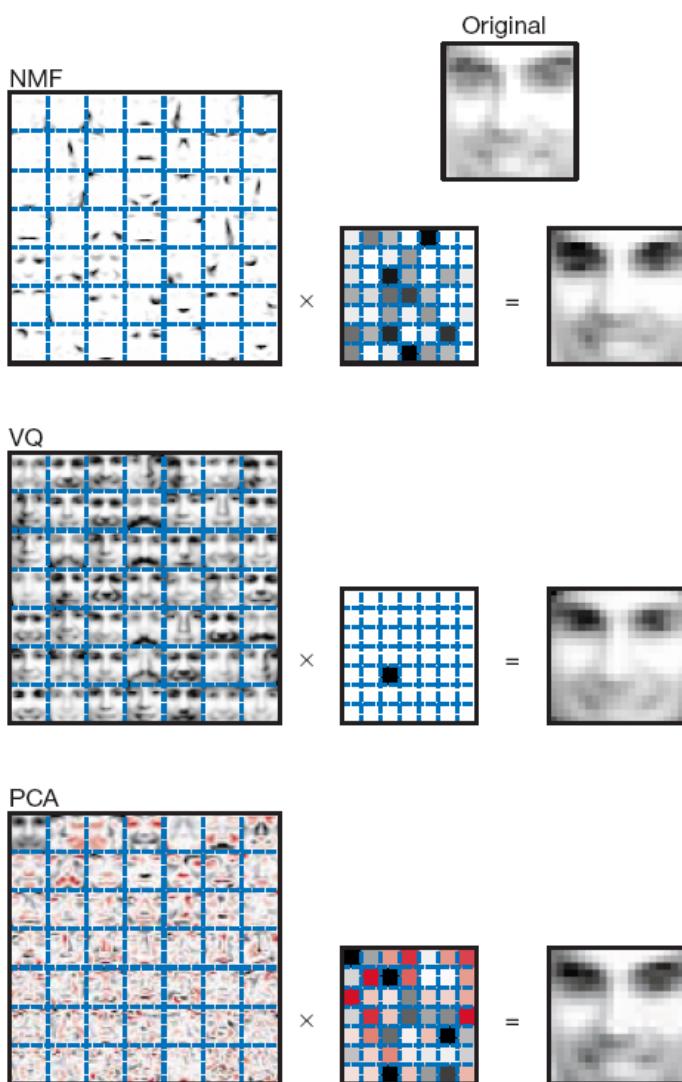


Figure 1 Non-negative matrix factorization (NMF) learns a parts-based representation of faces, whereas vector quantization (VQ) and principal components analysis (PCA) learn holistic representations. The three learning methods were applied to a database of $m = 2,429$ facial images, each consisting of $n = 19 \times 19$ pixels, and constituting an $n \times m$ matrix V . All three find approximate factorizations of the form $V \approx WH$, but with three different types of constraints on W and H , as described more fully in the main text and methods. As shown in the 7×7 montages, each method has learned a set of $r = 49$ basis images. Positive values are illustrated with black pixels and negative values with red pixels. A particular instance of a face, shown at top right, is approximately represented by a linear superposition of basis images. The coefficients of the linear superposition are shown next to each montage, in a 7×7 grid, and the resulting superpositions are shown on the other side of the equality sign. Unlike VQ and PCA, NMF learns to represent faces with a set of basis images resembling parts of faces.

Learning the parts of objects by non-negative matrix factorization

Daniel D. Lee* & H. Sebastian Seung*†

* Bell Laboratories, Lucent Technologies, Murray Hill, New Jersey 07974, USA

† Department of Brain and Cognitive Sciences, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA

NATURE | VOL 401 | 21 OCTOBER 1999 | www.nature.com

Generalizability

Do not multiply causes!



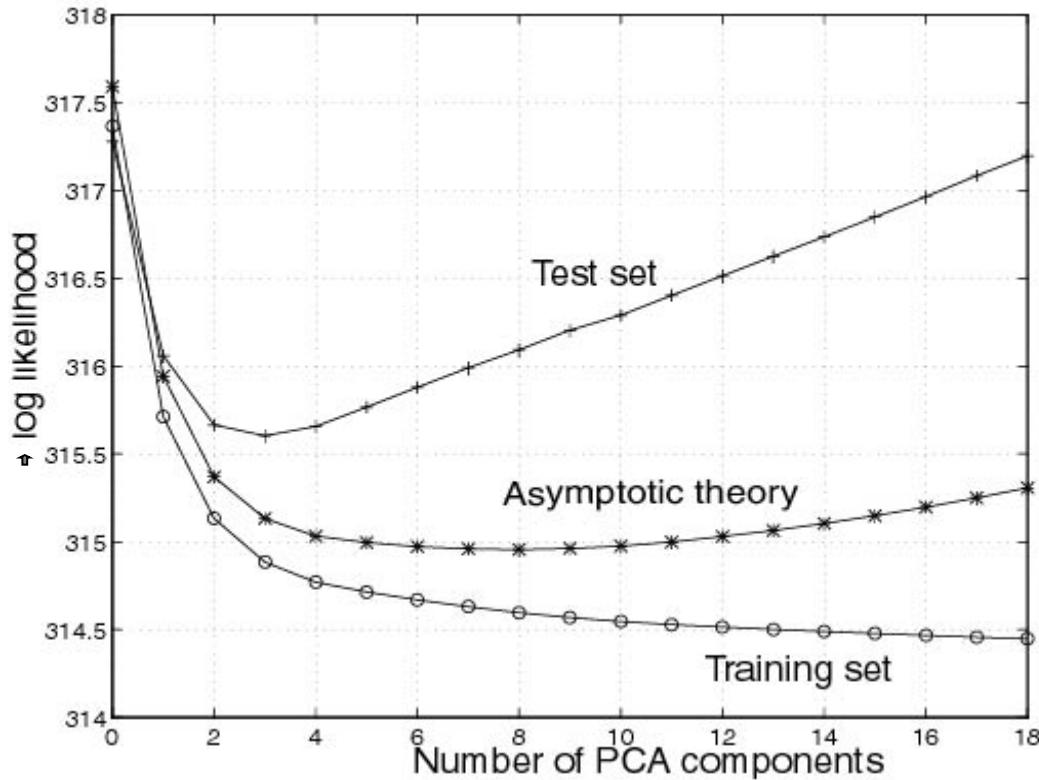
- Generalizability is defined as *the expected performance on a random new sample*
 - A models mean performance on a "fresh" data set is an unbiased estimate of generalization
- Typical loss functions:

$$\begin{aligned} & \langle -\log p(\mathbf{s} | \mathbf{x}, D) \rangle, & \langle -\log p(\mathbf{x} | D) \rangle, \\ & \left\langle (\mathbf{s} - \hat{\mathbf{s}}(D))^2 \right\rangle, & \left\langle \log \frac{p(\mathbf{s}, \mathbf{x} | D)}{p(\mathbf{s} | D)p(\mathbf{x} | D)} \right\rangle \end{aligned}$$

- Results can be presented as "bias-variance trade-off curves" or "learning curves"

L.K. Hansen and J. Larsen: *Unsupervised Learning and Generalization*
Proc. of IEEE International Conference on Neural Networks,
Washington DC, pp. 25-30, June 1996

Bias-variance trade-off as function of PCA dimension in fMRI data



Hansen et al. *NeuroImage* (1999)

Variance inflation in PCA

Journal of Machine Learning Research 12 (2011) 2027-2044

Submitted 1/11; Published 6/11

A Cure for Variance Inflation in High Dimensional Kernel Principal Component Analysis

Trine Julie Abrahamsen

TJAB@IMM.DTU.DK

Lars Kai Hansen

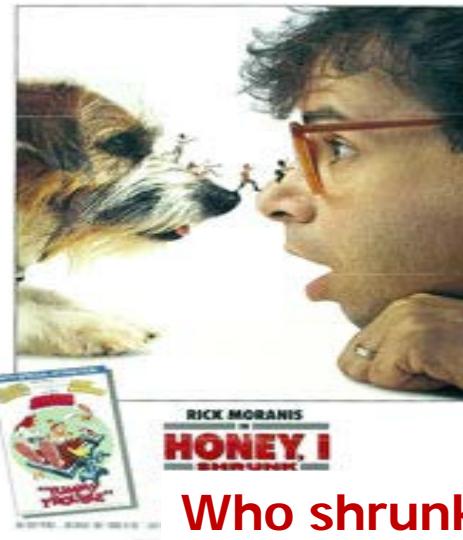
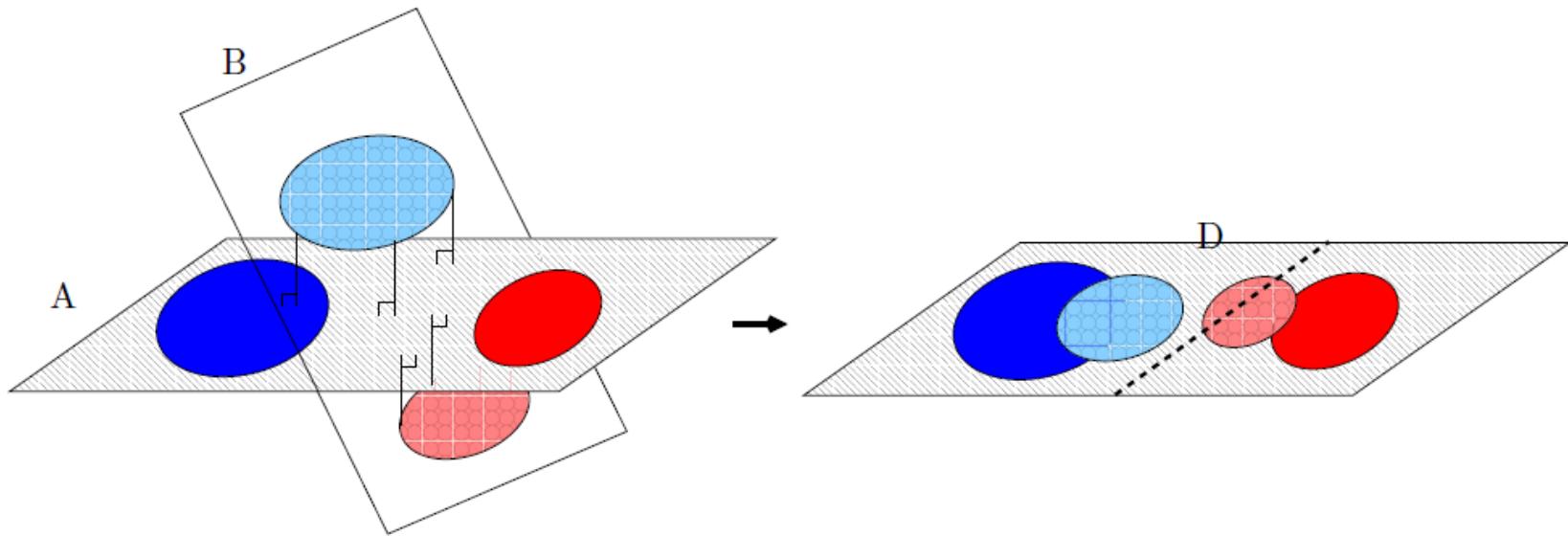
LKH@IMM.DTU.DK

DTU Informatics

Technical University of Denmark

Richard Petersens Plads, 2800 Lyngby, Denmark

Variance inflation in PCA



Who shrunk the test set?

Modeling the generalizability of SVD

- Rich physics literature on "retarded" learning

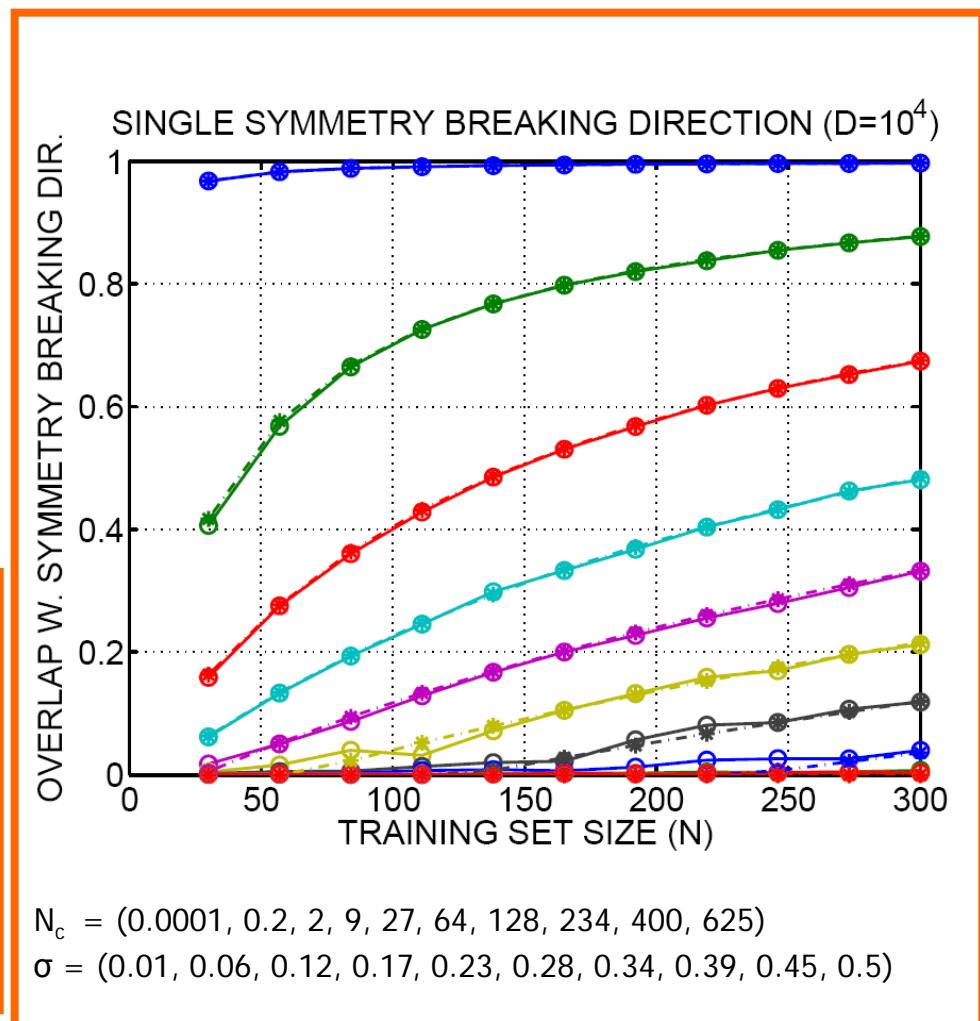
- Universality**

- Generalization for a "single symmetry breaking direction" is a function of ratio of N/D and signal to noise S
- For subspace models-- a bit more complicated -- depends on the component SNR's and eigenvalue separation
- For a single direction, the mean squared overlap $R^2 = \langle (u_1^T u_0)^2 \rangle$ is computed for $N, D \rightarrow \infty$

$$R^2 = \begin{cases} (\alpha S^2 - 1) / S(1 + \alpha S) & \alpha > 1/S^2 \\ 0 & \alpha \leq 1/S^2 \end{cases}$$

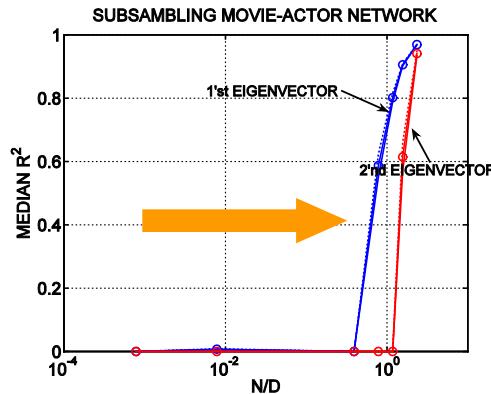
$$\alpha = N/D \quad S = 1/\sigma^2 \quad N_c = D/S^2$$

Hoyle, Rattray: Phys Rev E 75 016101 (2007)

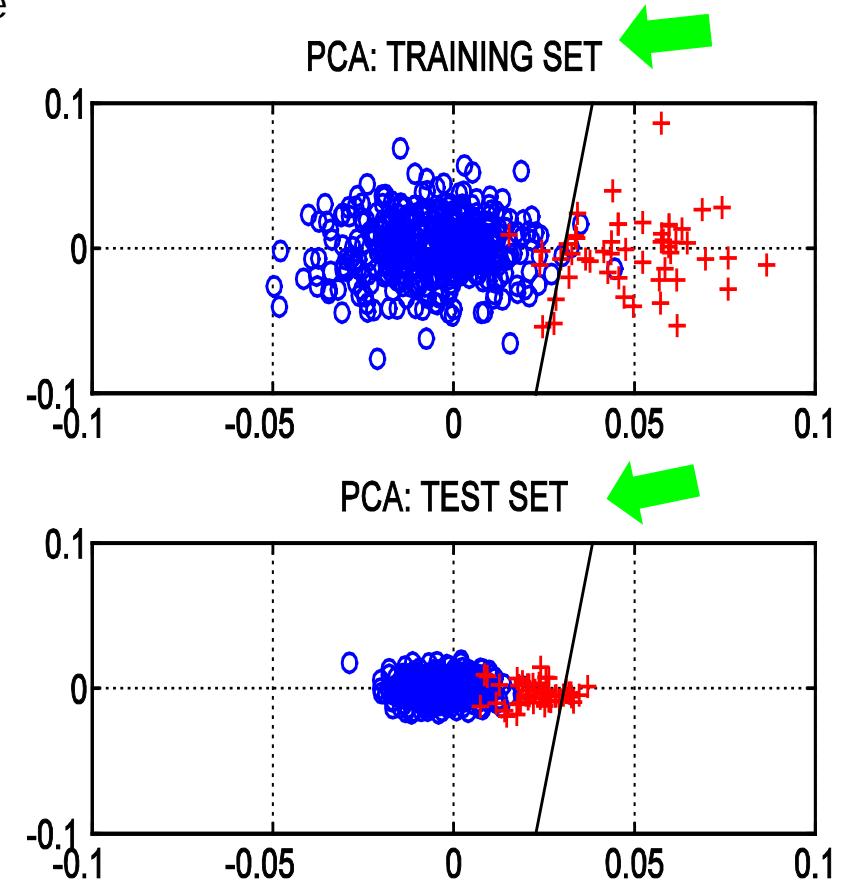


Restoring the generalizability of SVD

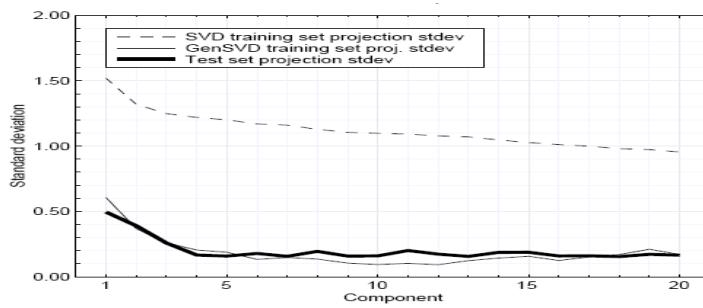
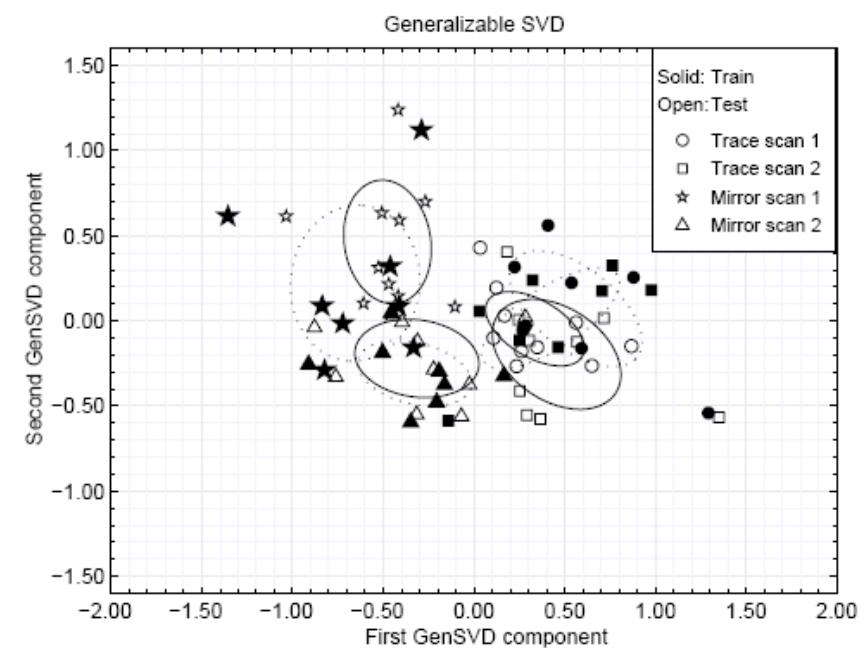
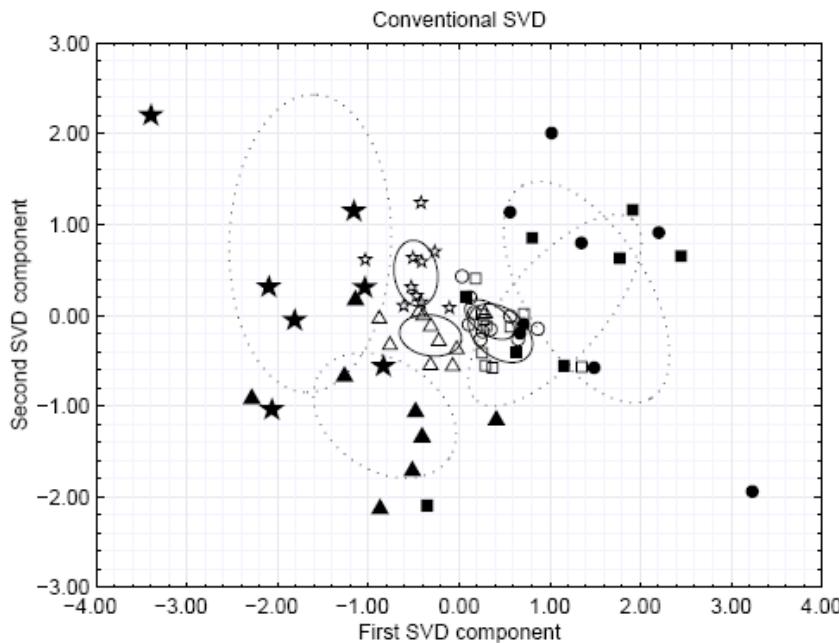
- Now what happens if you are on the slope of generalization, i.e., N/D is just beyond the transition to retarded learning ?



- The estimated projection is offset, hence, future projections will be too small!
- ...problem if discriminant is optimized for unbalanced classes in the training data!



Heuristic: Leave-one-out re-scaling of SVD test projections

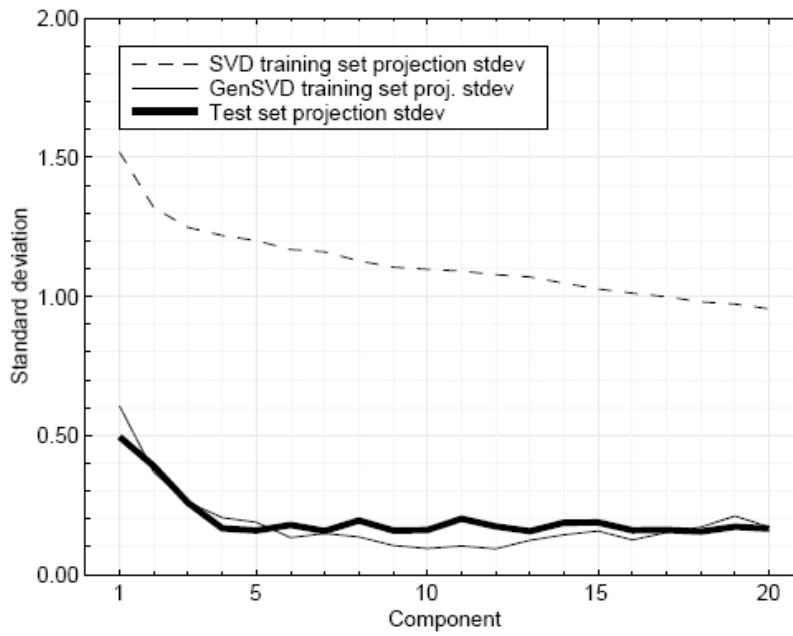


$N=72, D=2.5 \cdot 10^4$

Kjems, Hansen, Strother: "Generalizable SVD for Ill-posed data sets" NIPS (2001)

Re-scaling the component variances by leave one out

Possible to compute the new scales by leave-one-out
doing N SVD's of size $N \ll D$



Kjems, Hansen, Strother: NIPS (2001)

Approximating LOO (leave-one-out: "N")

Let $\{x_1, \dots, x_N\}$ be N training data points in a D dimensional input space

$$x_N = x_N^\perp + x_N^{\parallel}, \quad u_{N-1,k}^T \cdot x_N = u_{N-1,k}^T \cdot x_N^{\parallel},$$

$$u_{N-1,k}^T \cdot x_N = u_{N-1,k}^T \cdot x_N^{\parallel} \approx u_{N,k}^T \cdot x_N^{\parallel}$$

T.J. Abrahamsen, L.K. Hansen. A Cure for Variance Inflation in High Dimensional Kernel Principal Component Analysis. Journal of Machine Learning Research 12:2027-2044 (2011).

Two approximations

Adjusting for the mean overlap

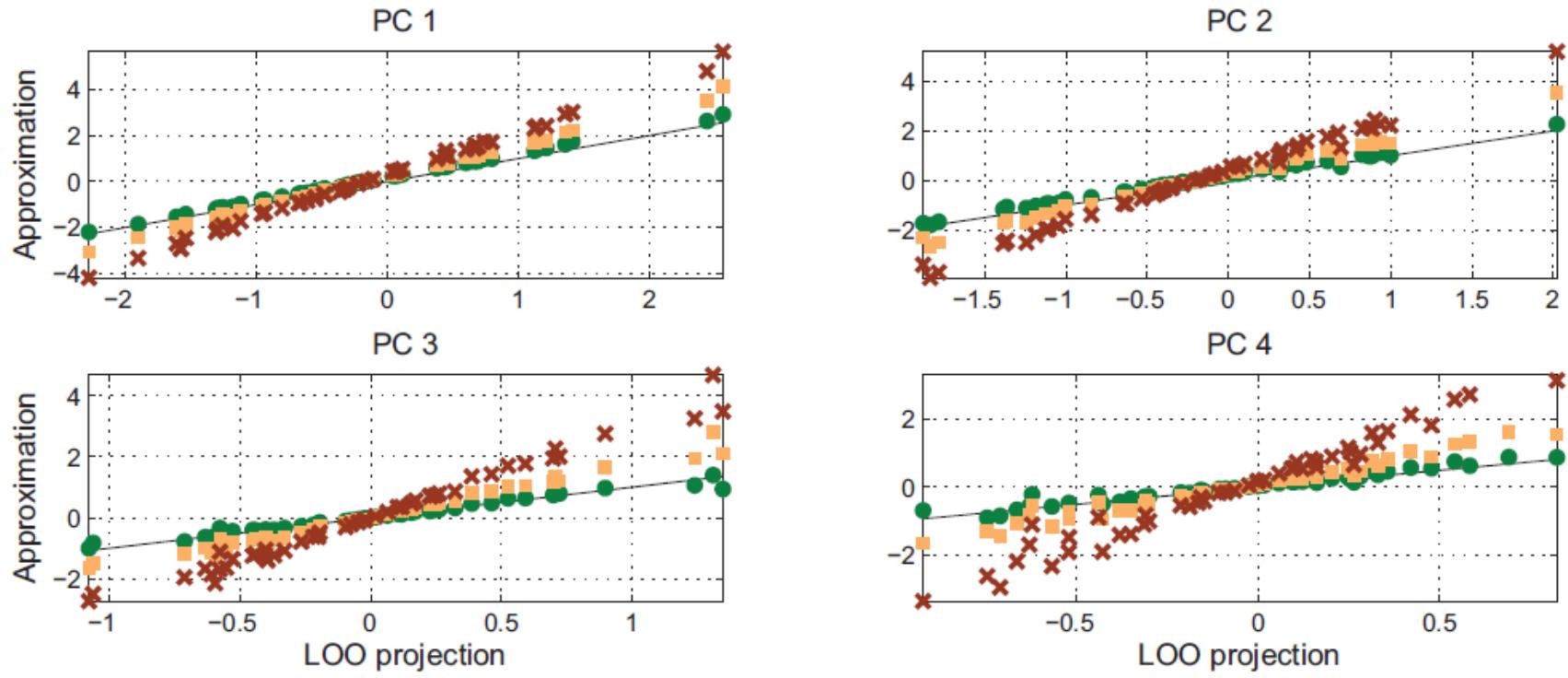
$$R^2 = \begin{cases} (\alpha S^2 - 1) / S(1 + \alpha S) & \alpha > 1/S^2 \\ 0 & \alpha \leq 1/S^2 \end{cases}$$

$$\alpha = N/D \quad S = 1/\sigma^2 \quad N_c = D/S^2$$

Hoyle, Rattray: Phys Rev E **75** 016101 (2007)

Adjusting for lost projection

$$\mathbf{u}_{N-1,k}^T \cdot \mathbf{x}_N = \mathbf{u}_{N-1,k}^T \cdot \mathbf{x}_N^\parallel \approx \mathbf{u}_{N,k}^T \cdot \mathbf{x}_N^\parallel$$

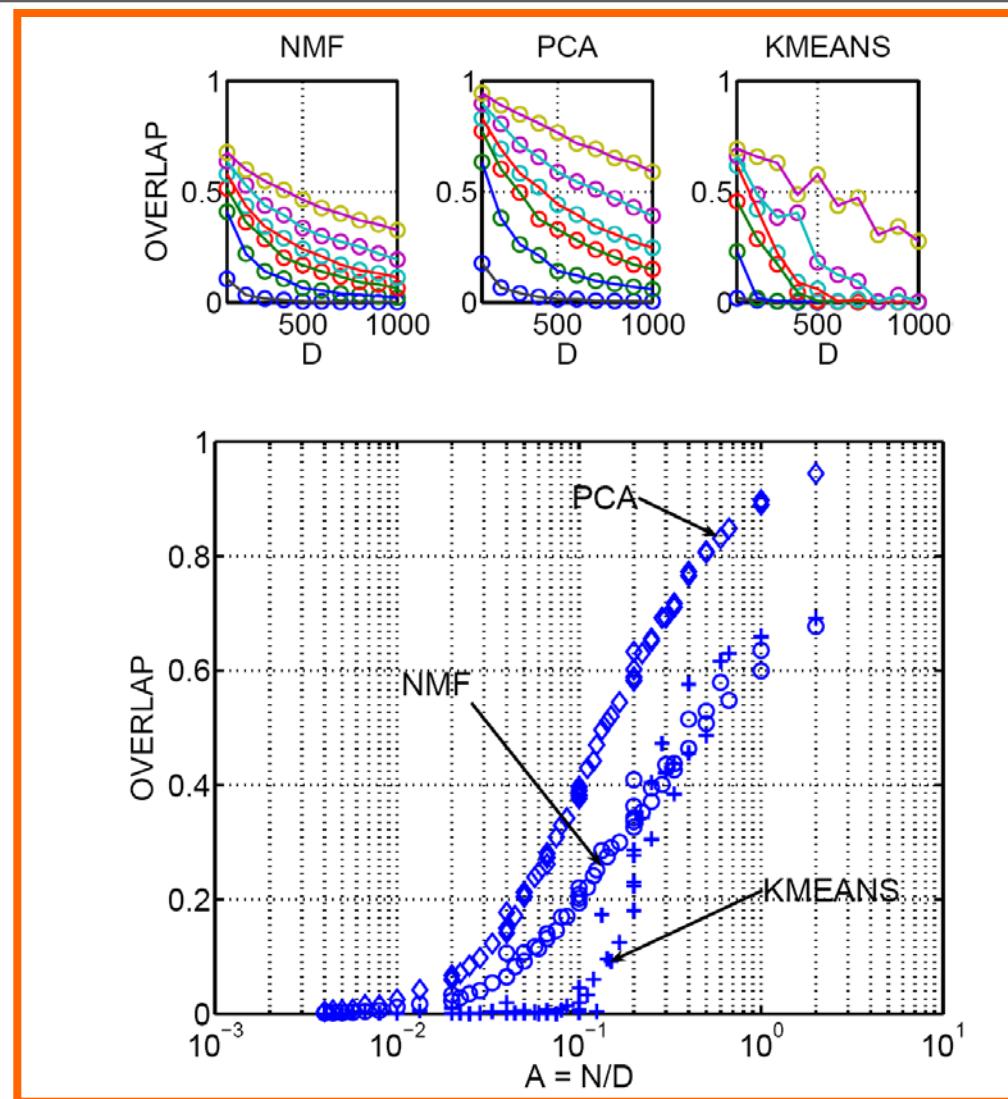


Approximating the leave-one-out (LOO) procedure. Here we simulate data with four normal independent signal components, $x = \sum_{k=1}^4 \eta_k u_k + \epsilon$ of strengths (1.4, 1.2, 1.0, 0.8, embedded in i.i.d. normal noise $\epsilon \sim N(0, \sigma^2 \mathbf{1})$, with $\sigma = 0.2$. The dimension was $D = 2000$ and the sample size was $N = 50$. In the four panels we show the training set projections (red crosses), the projections corrected for the theoretical mean overlap (Hoyle and Rattray, 2007) (yellow squares) and the geometric approximation in Equation (1) (green dots) versus the exact LOO projections (black line).

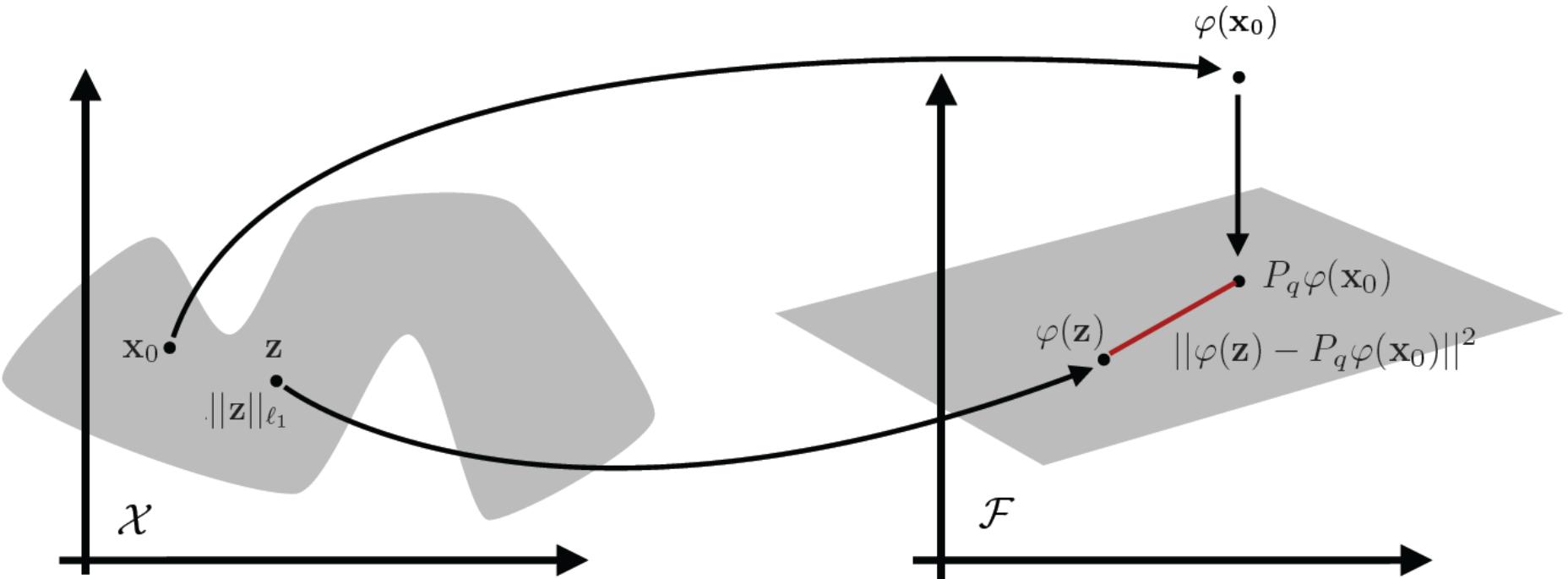
Universality in PCA, NMF, Kmeans

- Looking for universality by simulation
 - learning two clusters in white noise.
- Train K=2 component factor models.
- Measure overlap between line of sight and plane spanned by the two factors.

Experiment
Variable: N, D
Fixed: SNR



Beyond the linear model: Non-linear denoising and manifold representations



TJ Abrahamsen, LKH. Sparse non-linear denoising: Generalization performance
and pattern reproducibility in functional MRI . Pattern Recognition Letters 32(15) 2080-2085 2011

Beyond the linear model:

Kernel PCA is based on non-linear mapping of N data points

$$\mathbf{x}_n \rightarrow \varphi(\mathbf{x}_n) \equiv \varphi_n$$

Our aim is to let the local geometry of mapped points represent the local geometry of the input space, hence we connect the spaces by letting inner products be defined so that close points in input space are represented by high values of their inner product

$$K_{n,n'} = \varphi_n^T \varphi_{n'} = \exp\left(-\frac{||\mathbf{x}_n - \mathbf{x}_{n'}||^2}{c}\right)$$

Note $c \rightarrow \infty$ defaults to linear PCA

Schölkopf et al. : Nonlinear Component Analysis as a Kernel Eigenvalue Problem. Neural Comp (1998)

Beyond the linear model:

- Kernel PCA is based on non-linear mapping of data to

$$\mathbf{x}_n \rightarrow \varphi(\mathbf{x}_n) \equiv \varphi_n, \quad n = 1, \dots, N$$

Aim is to locate maximum variance directions in the feature space, i.e.

$$\mathbf{l}_1 \equiv \arg \max_{\|\mathbf{l}\|=1} \left\langle (\mathbf{l}^T \cdot \varphi)^2 \right\rangle, \quad \varphi(\mathbf{x}_n) = \sum_k \mathbf{l}_k s_{k,n}$$

The principal direction is in the span of data:

$$\mathbf{l}_1 = \sum_{n=1}^N a_{1,n} \varphi_n$$

$$\mathbf{a}_1 = \arg \max_{\|\mathbf{a}\|=1} \left\langle \mathbf{a}^T \cdot \mathbf{K} \cdot \mathbf{a} \right\rangle, \quad \mathbf{K}_{n,n'} = \varphi_n^T \cdot \varphi_{n'} = \exp \left(-\frac{\|x_n - x_{n'}\|^2}{2c} \right)$$

TJ Abrahamsen and LK Hansen. "Input Space Regularization Stabilizes Pre-image for Kernel PCA De-noising". Proc. of Int. Workshop on Machine Learning for Signal Processing, Grenoble, France (2009).

Approximating the LOO cure for kPCA

Let $\{x_1, \dots, x_N\}$ be N training data points

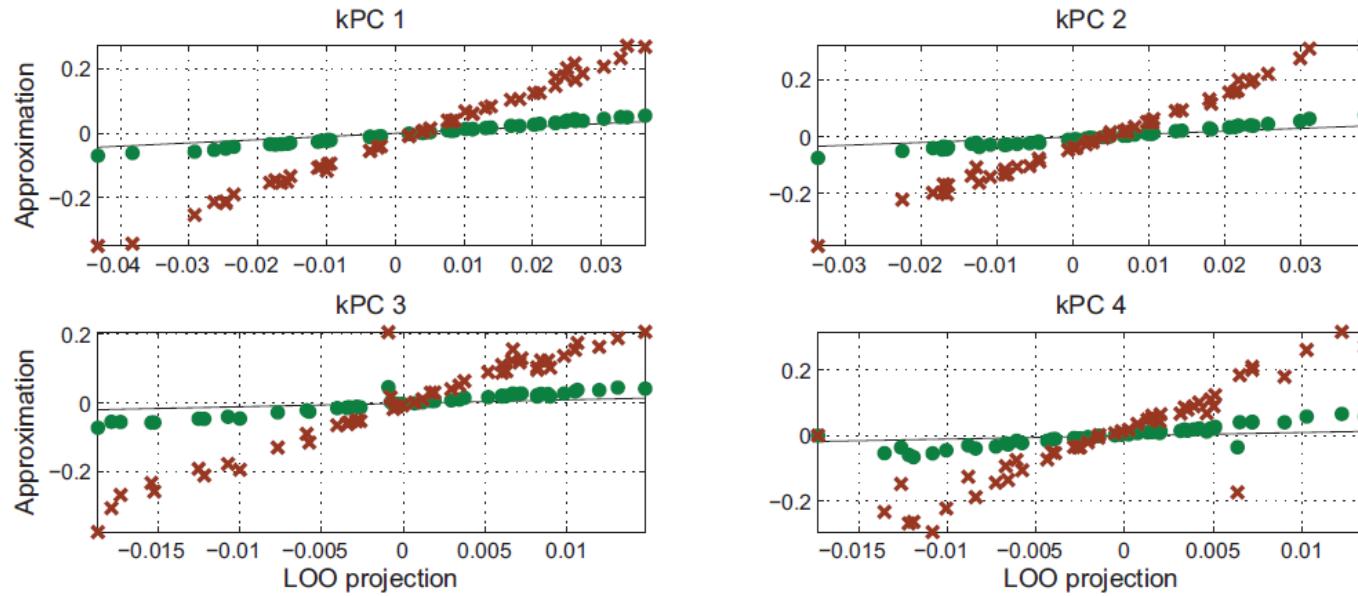
$$\tilde{\phi}(x) = \phi(x) - \bar{\phi}.$$

$$\widetilde{K} = K - \frac{1}{N} \mathbf{1}_{NN} K - \frac{1}{N} K \mathbf{1}_{NN} + \frac{1}{N^2} \mathbf{1}_{NN} K \mathbf{1}_{NN}$$

$$\widetilde{K} \alpha_i = \lambda_i \alpha_i$$

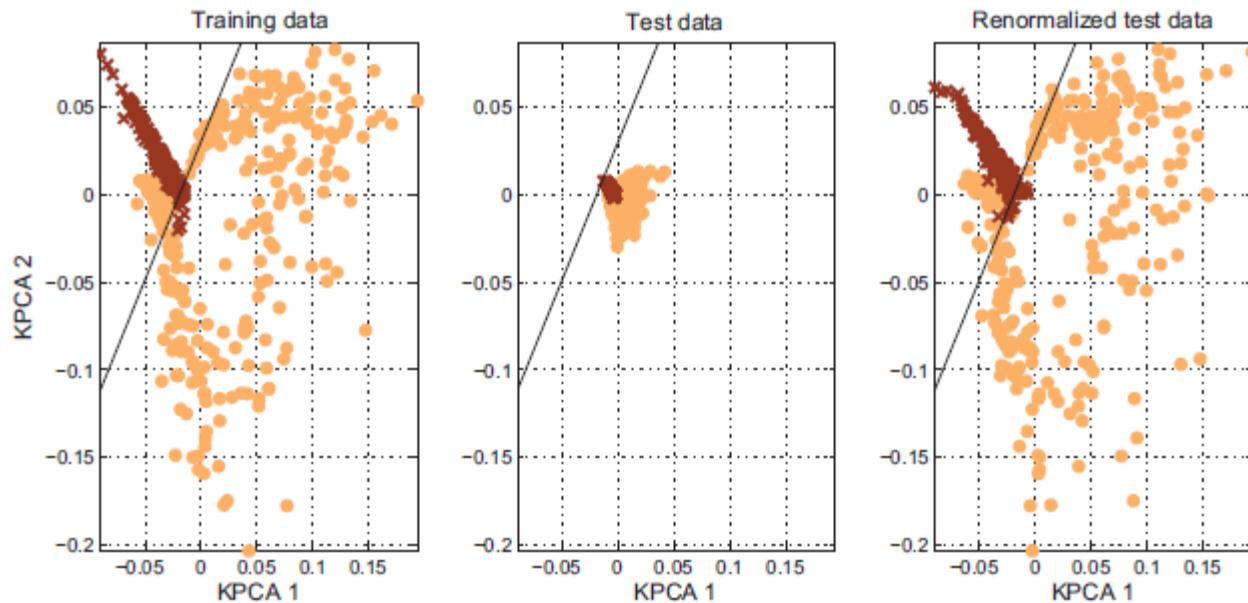
$$\beta_i = \tilde{\phi}(x)^T v_i = \sum_{n=1}^N \alpha_{in} \tilde{\phi}(x)^T \tilde{\phi}(x_n) = \sum_{n=1}^N \alpha_{in} \tilde{k}(x, x_n)$$

$$\|x_n - x_N\|^2 = \|x_n - x_N^{\parallel}\|^2 + \|x_N^{\perp}\|^2$$

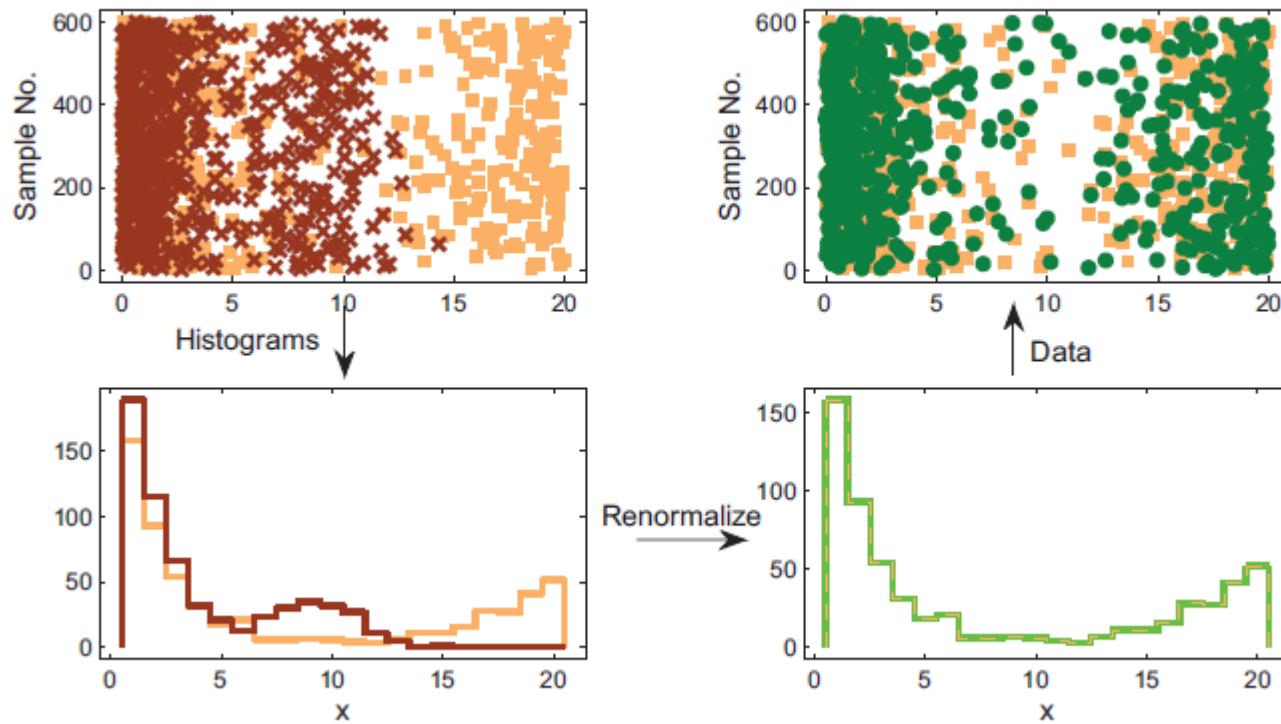


$$\beta_i(x_N) = \sum_{n=1}^{N-1} \alpha_{in} \tilde{k}(x_N, x_n) = \exp \left(-\frac{1}{c} \|x_N^{\perp}\|^2 \right) \sum_{n=1}^{N-1} \alpha_{in} \tilde{k}(x_N^{\parallel}, x_n)$$

Application to classification of high-dimensional data on manifolds



Non-parametric histogram equalization



```
>> [as,ia]=sort(a);  
>> [bs,ib]=sort(b);  
>> b(ib)=as;
```

Non-parametric histogram equalization

Algorithm 1 Approximate renormalization in kernel PCA

Require: X_{tr} and X_{te} to be $N_{tr} \times D$ and $N_{te} \times D$ respectively

Compute \tilde{K}_{tr} using Equation (2) and find the eigenvectors, $\alpha_1, \dots, \alpha_q$

for $i = 1$ to N_{tr} **do**

$f_{tr}^{i,:} \leftarrow P_q(x_{tr}^{i,:}) = \tilde{k}_{x_i}^T \alpha^{i,q}$ {see Equation (3)}

end for

for $j = 1$ to N_{te} **do**

$f_{te}^{j,:} \leftarrow P_q(x_{te}^{j,:}) = \tilde{k}_{x_j}^T \alpha^{i,q}$ {see Equation (3)}

end for

for $d = 1$ to q **do**

$[f_{sort}, \cdot] \leftarrow \text{sort}(f_{tr}^{:,d})$ {ascending order}

$[\cdot, I] \leftarrow \text{sort}(f_{te}^{:,d})$ {ascending order}

if $N_{tr} = N_{te}$ **then**

$h \leftarrow f_{sort}$

else { $N_{tr} \neq N_{te}$ }

$h \leftarrow \text{spline}([1 : N_{tr}], f_{sort}, \text{linspace}(1, N_{tr}, N_{te}))$ {interpolate to create N_{te} values of f_{sort} in the interval $[1 : N_{tr}]$ }

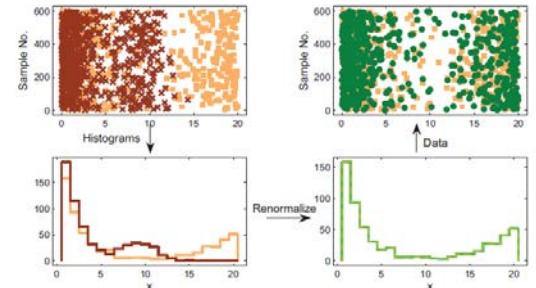
end if

for $n = 1$ to N_{te} **do**

$\tilde{g}_{te}^{I(n),d} \leftarrow h^{n,d}$ {renormalized test data in the principal subspace, see Equation (4)}

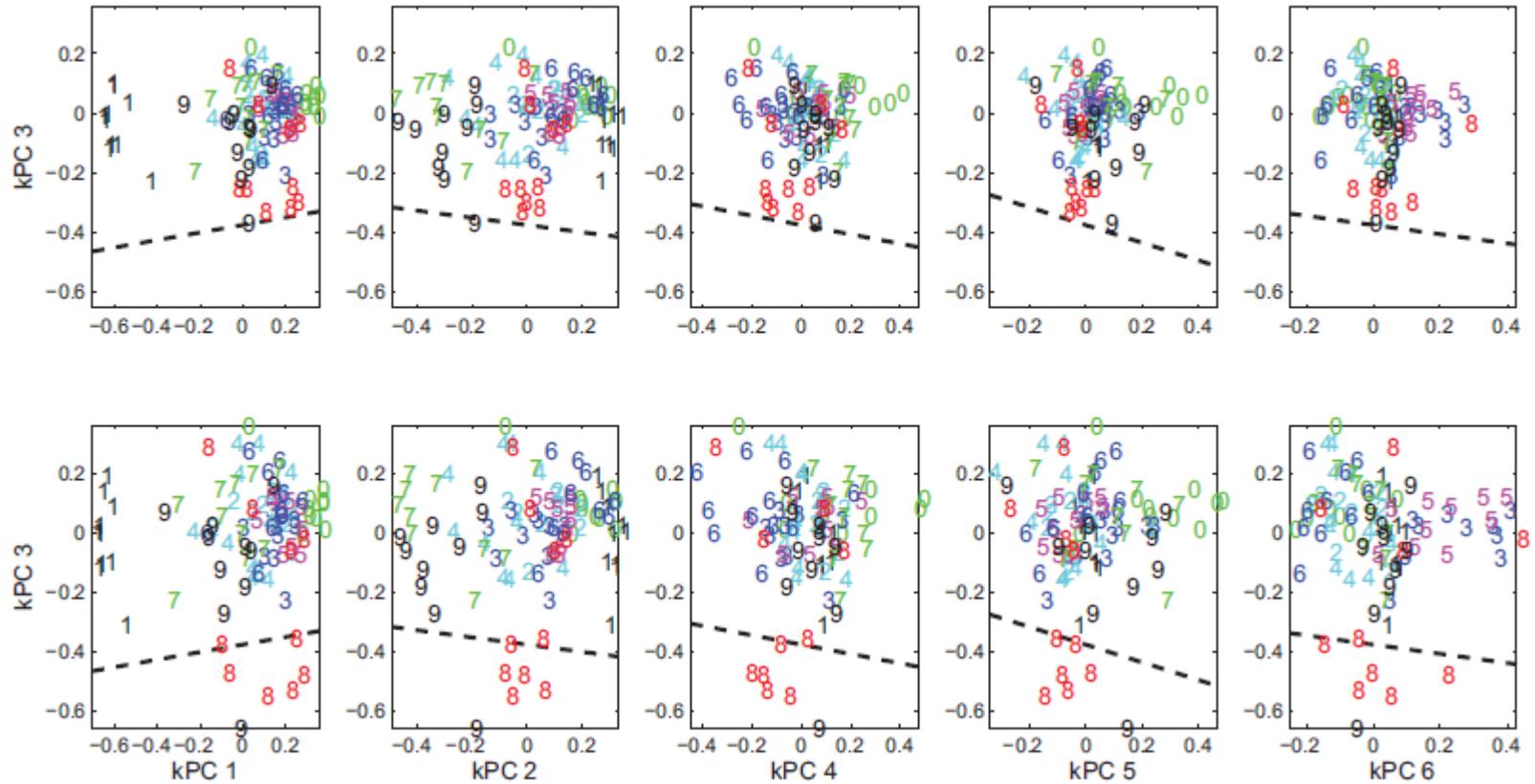
end for

end for



Application to classification of high-dimensional data on manifolds

Test prior to scaling



Test post scaling

Supervised learning from small samples in high dimensional spaces

EEG imaging

Linear ill-posed
inverse problem

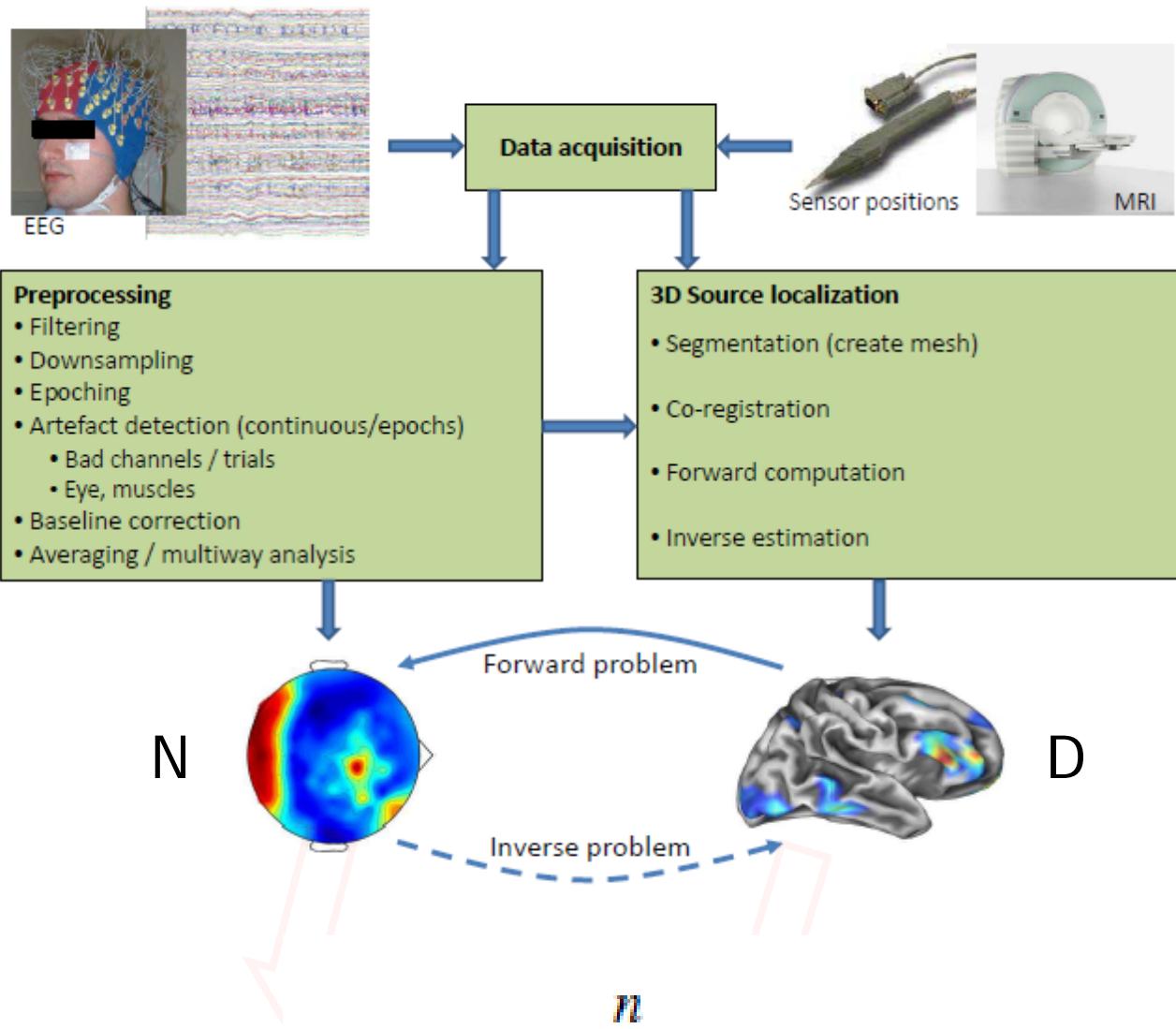
$$Y: 1 \times N$$

$$W: 1 \times D$$

$$X: D \times N$$

$$D \gg N$$

Need priors to
solve!



$$y_{\mu} = \sum_{i=1}^n w_i X_{i\mu} + \xi_{\mu},$$

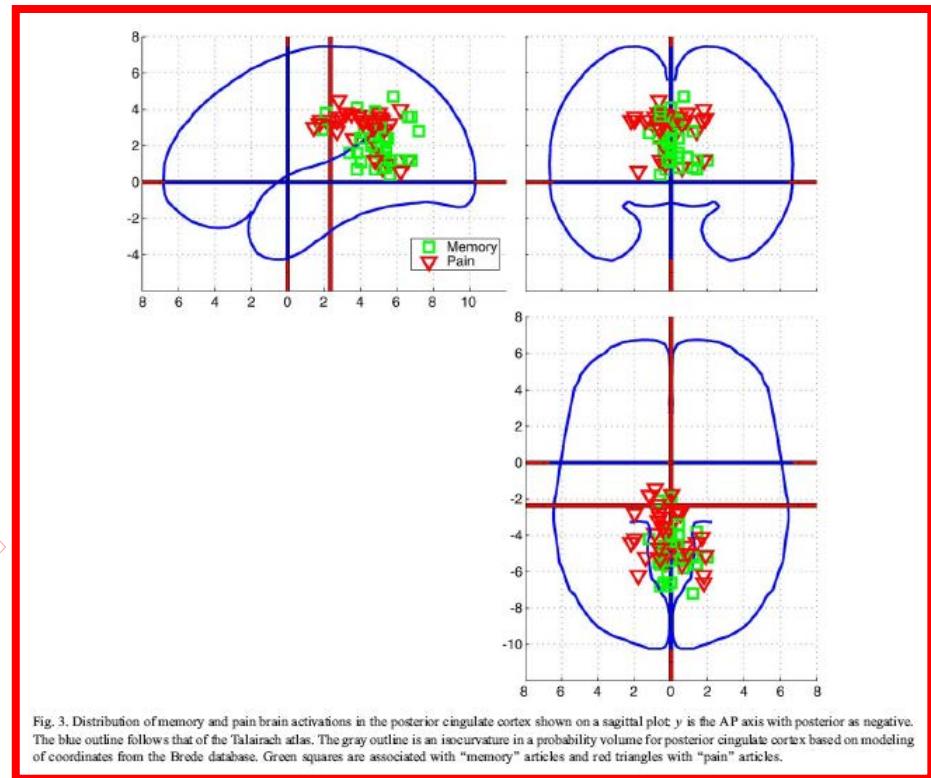
Why 3D real-time imaging?

Enable on-line visual quality control

Neuro-feedback applications can be based on activity in specific brain structures /networks

Context priors may relate to 3D location (from meta analysis)

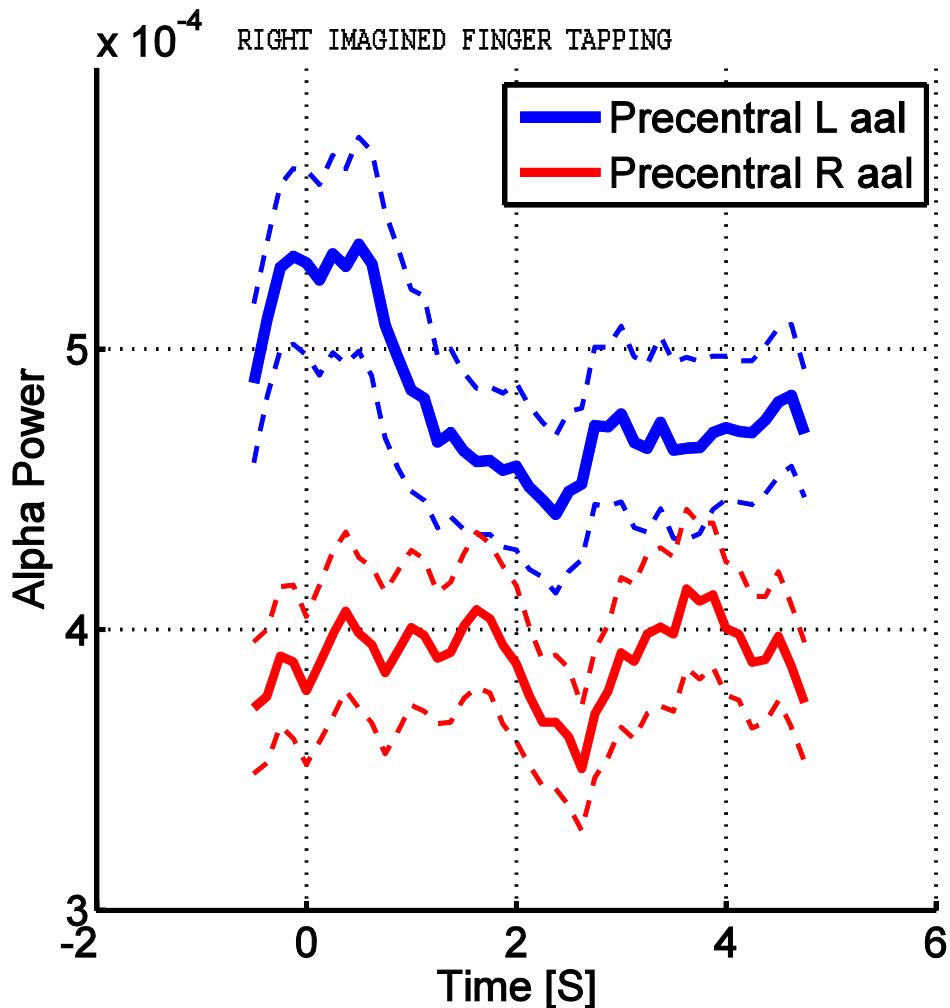
Evidence that BCI /decoding can be improved by 3D representation



Finn Årup Nielsen, Daniela Balslev, Lars Kai Hansen, "Mining the Posterior Cingulate: Segregation between memory and pain components". *NeuroImage*, 27(3):520-532, (2005)

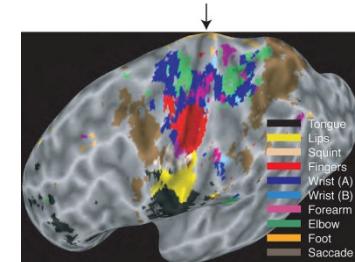
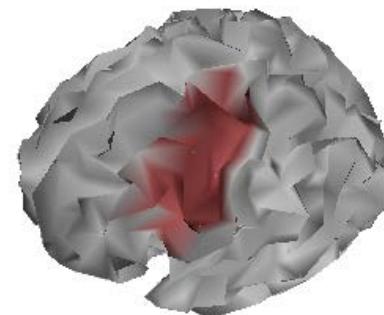
Trujillo-Barreto, Nelson J., Eduardo Aubert-Vázquez, and Pedro A. Valdés-Sosa. "Bayesian model averaging in EEG/MEG imaging." *NeuroImage* 21, no. 4 (2004): 1300-1319.

Do we get meaningful 3D reconstructions?



Imagined finger tapping
Left or right cued (at t=0)

Signal collected from an
AAL region



Variance inflation in linear regression

$$y = \mathbf{w}^\top \mathbf{x} + \epsilon = \sum_{d=1}^D w_d x_d + \epsilon, \quad \mathbf{x} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}).$$

$$\hat{\mathbf{w}} = \operatorname{argmin}_{\mathbf{w}} \sum_{n=1}^N (y_n - \mathbf{w}^\top \mathbf{x}_n)^2$$

$$G(N) = E_{y,\mathbf{x}} \left\{ E_N \left\{ (y - \mathbf{w}_N^\top \mathbf{x})^2 \right\} \right\}$$

Analytic learning curve 5

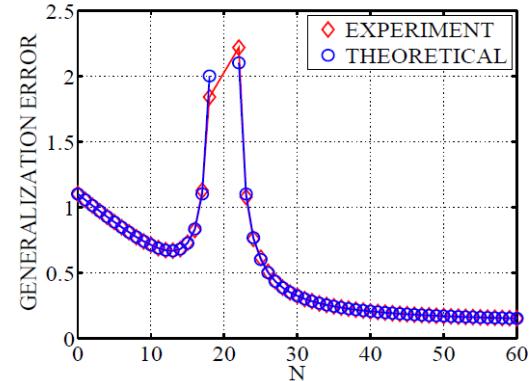


Fig. 1. Experimental and theoretical learning curves for the case $D = 20$ with $\sigma^2 = 0.1$, $\|\mathbf{w}_0\|^2 = 1$. The theoretical result for $N > D + 1$ is given in Hansen (1993). The sample size for the minimal error (for $N < D - 1$) is located at $N_{\min} = [D - 1 - \sqrt{D(D-1)}] \sqrt{\frac{\sigma^2}{\|\mathbf{w}_0\|^2}} = 13$. The results are based on 10000 simulated data sets.

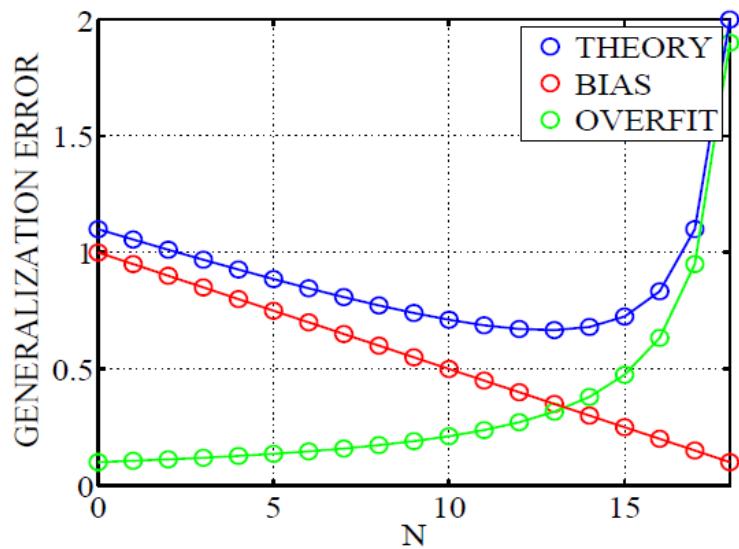
$$G(N) = \begin{cases} \left(1 - \frac{N}{D}\right) \|\mathbf{w}_0\|^2 + \frac{D-1}{D-N-1} \sigma^2 & N < D - 1, \\ \infty & D - 1 \leq N \leq D + 1 \\ \frac{N-1}{N-D-1} \sigma^2 & N > D + 1. \end{cases}$$

Hansen, L. K. Stochastic linear learning: Exact test and training error averages. *Neural Networks* 6(3): 393–396 (1993)

Barber, D., D. Saad, and P. Sollich. Test error fluctuations in finite linear perceptrons. *Neural computation* 7(4): 809-821 (1995)

Variance inflation in linear regression

$$G(N) = \begin{cases} \left(1 - \frac{N}{D}\right) \|\mathbf{w}_0\|^2 + \frac{D-1}{D-N-1} \sigma^2 & N < D-1, \\ \infty & D-1 \leq N \leq D+1 \\ \frac{N-1}{N-D-1} \sigma^2 & N > D+1. \end{cases}$$



$$\|\mathbf{w}_0\|^2 - E_N \{\|\hat{\mathbf{w}}\|^2\} = \left(1 - \frac{N}{D}\right) \|\mathbf{w}_0\|^2$$

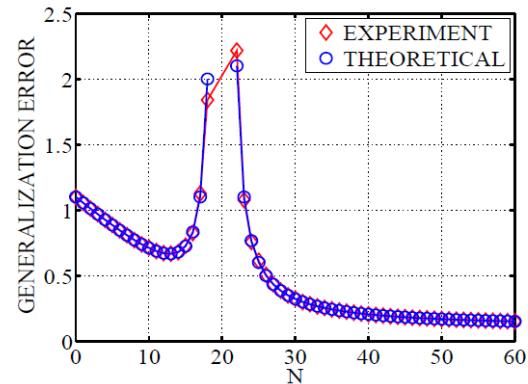


Fig. 1. Experimental and theoretical learning curves for the case $D = 20$ with $\sigma^2 = 0.1$, $\|\mathbf{w}_0\|^2 = 1$. The theoretical result for $N > D+1$ is given in Hansen (1993). The sample size for the minimal error (for $N < D-1$) is located at $N_{\min} = [D-1 - \sqrt{D(D-1)} \sqrt{\frac{\sigma^2}{\|\mathbf{w}_0\|^2}}] = 13$. The results are based on 10000 simulated data sets.

Variance inflation in linear regression

$$\mathbf{w} = \sum_{n=1}^N \beta_n \mathbf{x}_n \quad K_{m,n} = \mathbf{x}_m^\top \mathbf{x}_n$$

$$\hat{\mathbf{w}} = \sum_{m,n=1}^N \mathbf{x}_n (K^{-1})_{n,m} y_m$$

$$\sigma^2 (\hat{\mathbf{w}}^\top \mathbf{x}_n) = 1/N \sum_{n=1}^N y_n^2$$

Training set
variance of
predictions

$$E_N \left\{ 1/N \sum_{n=1}^N y_n^2 \right\} = \|\mathbf{w}_0\|^2 + \sigma^2$$

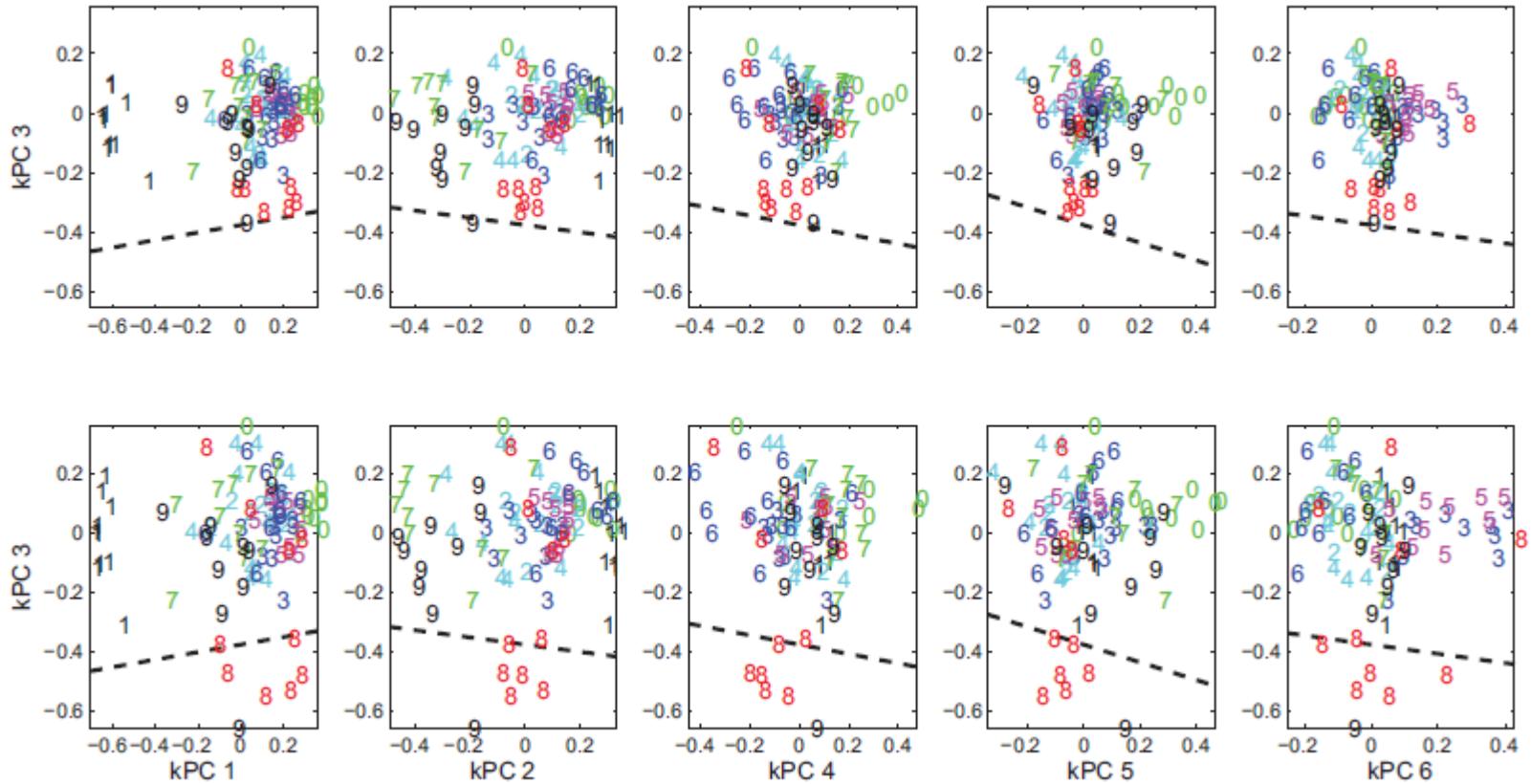
Test set variance
of predictions

$$E_{\mathbf{x}} \left\{ E_N \left\{ \hat{\mathbf{w}}^\top \mathbf{x} \right\} \right\} = E_N \left\{ \|\hat{\mathbf{w}}\|^2 \right\} = \frac{N}{D} \|\mathbf{w}_0\|^2$$

Application to classification of high-dimensional data on manifolds

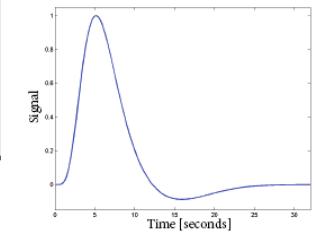
USPS data classification: Digit "8" vs rest

Test prior to scaling

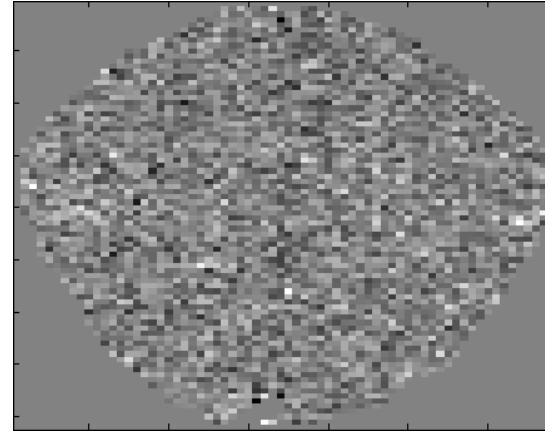


Test post scaling

Functional MRI



- Indirect measure of neural activity - hemodynamics
- A cloudy window to the human brain
- Challenges:
 - Signals are multi-dimensional mixtures
 - No simple relation between measures and brain state - "what is signal and what is noise"?



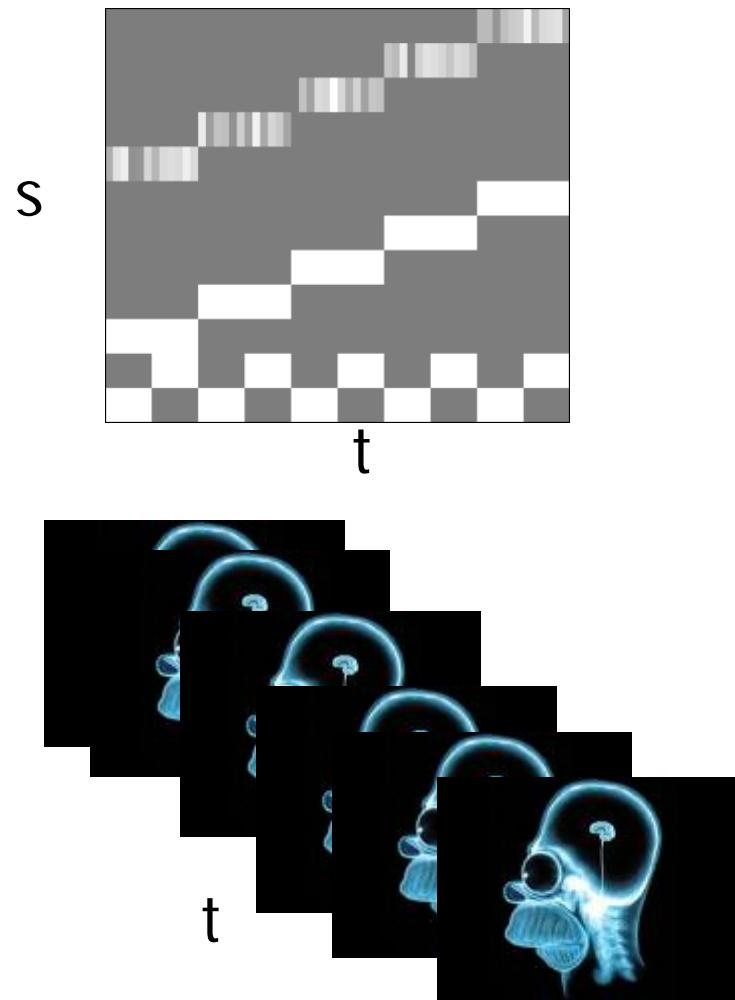
TR = 333 ms

Multivariate neuroimaging models

Neuroimaging aims at extracting the mutual information between stimulus and response.

- Stimulus: Macroscopic variables, "design matrix" ... $s(t)$
- Response: Micro/meso-scopic variables, the neuroimage ... $x(t)$
- Mutual information is stored in the joint distribution ... $p(x,s).$

Often $s(t)$ is assumed known....unsupervised methods consider $s(t)$ or parts of $s(t)$ "hidden".....



Application to classification of high-dimensional data on manifolds (fMRI , exceptional good SNR in raw data)

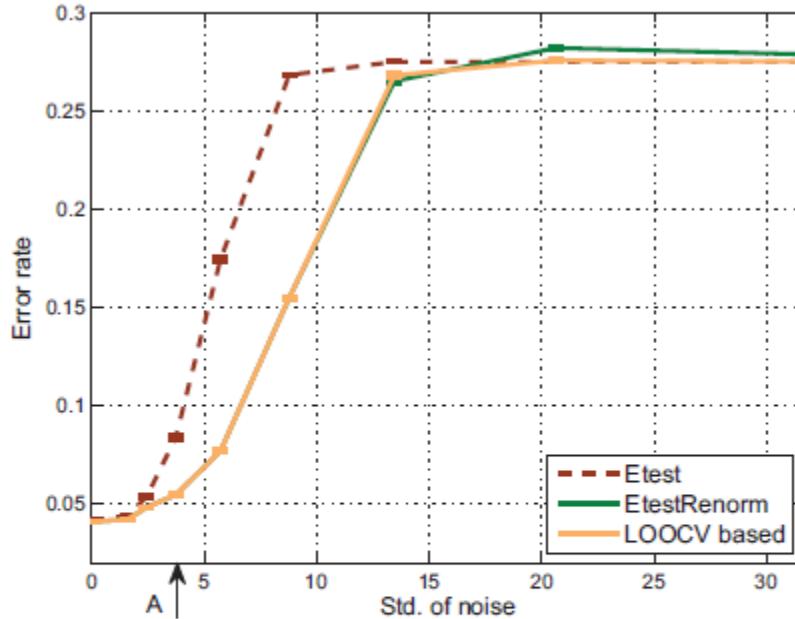


Figure 11: Mean error rates ± 1 standard deviation as a function of the noise level for fMRI data ($D = 16,384, N = 605$) . The test error based on conventional kernel PCA projections, renormalized projections, and a LOOCV scheme is shown. Renormalization is seen to clearly improve the performance. Arrow 'A' indicates the noise level used in Figure 12

Application to classification of high-dimensional data on manifolds (fMRI, exceptional good SNR in raw data)

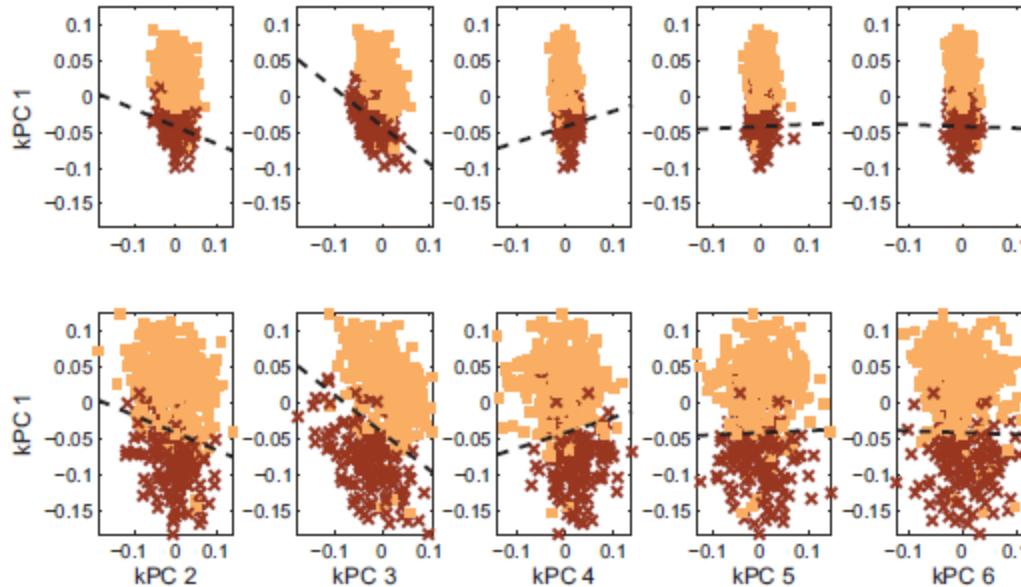


Figure 12: Test set projections of the fMRI data with Gaussian noise added as marked on Figure 11 ($\epsilon_i = \mathcal{N}(0, 3.8^2)$). The top row shows the conventional projections, while the bottom row shows the projections after renormalization. The ‘red class’ indicates activation, while the blue observations are acquired during rest. The dashed line marks the linear discriminant. The scale is chosen as the 5th percentile of the mutual distances.

Implications for the SVM?

Distribution of the decision function

$$\text{sign}(y(\mathbf{x})) = \text{sign} \left(\sum_{i \in S} y_i \alpha_i k(\mathbf{x}_i, \mathbf{x}) + b \right),$$

$$\beta_i(x_N) = \sum_{n=1}^{N-1} \alpha_{in} \tilde{k}(x_N, x_n) = \exp \left(-\frac{1}{c} \|x_N^\perp\|^2 \right) \sum_{n=1}^{N-1} \alpha_{in} \tilde{k}(x_N^\parallel, x_n)$$

'....unlike other machine learning methods, SVMs generalization error is related not to the input dimensionality of the problem, but to the margin with which it separates the data...'

J. Kwok IEEE TNN (1999)

Variance inflation in SVM decision function

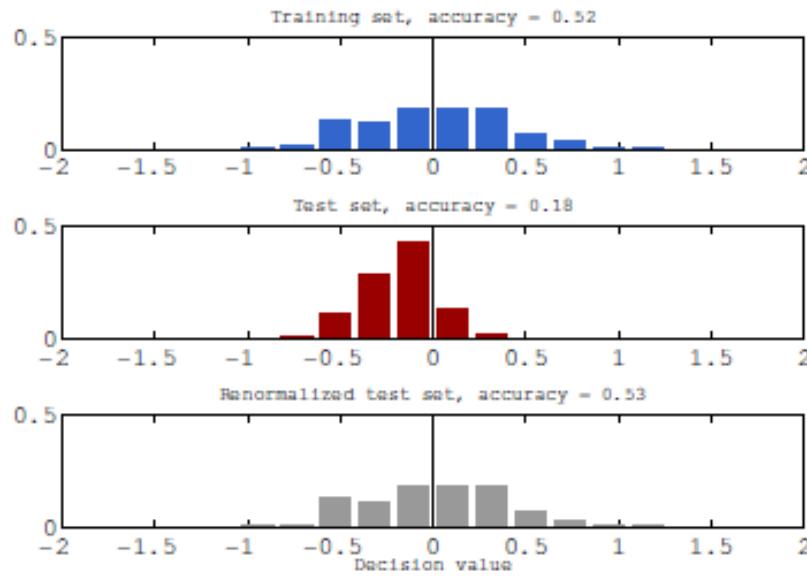


Figure 1: Illustration of the variance inflation phenomena in simulated data. The plots show the distribution of the decision values, f , of a FLD. The top panel is the training data, the middle panel is the test data, and the lower panel shows the result after applying the non-parametric scheme for restoring the variation as described in following section. The inflated variance of the training data compared to the test data is evident.

Decision function mis-match in the SVM (USPS)

$$G\text{-mean} = \sqrt{\text{sensitivity} \cdot \text{specificity}}$$

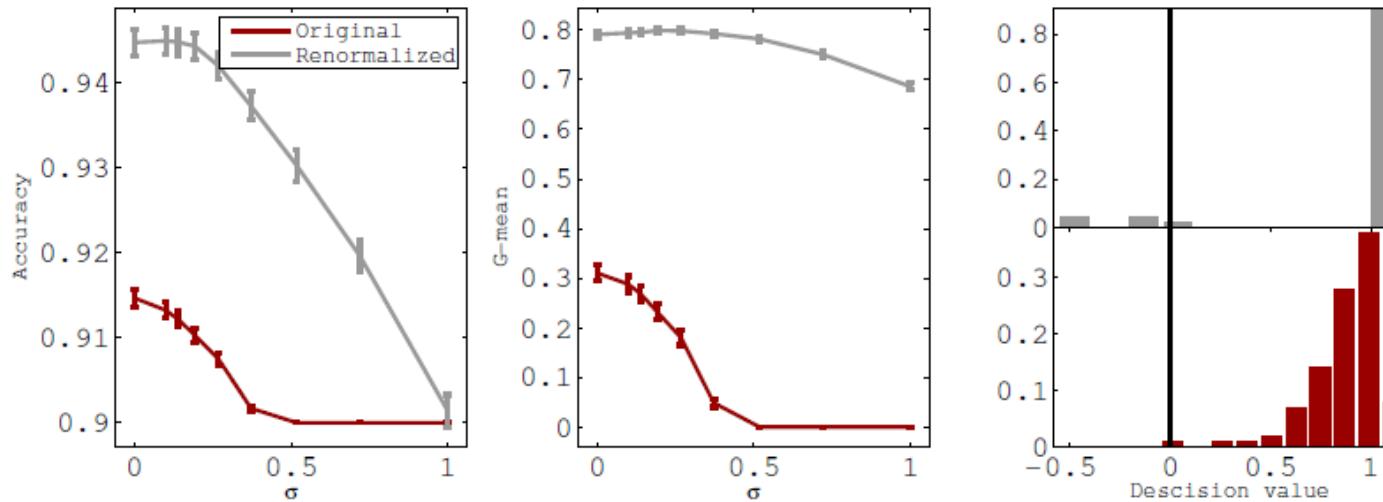


Fig. 1. Mean performance measures ± 1 std as a function of the noise level for the USPS data. The left and middle panels show the accuracy and the G-mean respectively. The test accuracy is shown in red while the renormalized test accuracy is shown in gray. The right panel shows an example of the histogram before and after renormalization (for a noise level of $\sigma = 0.27$).

T.J. Abrahamsen, LKH: Restoring the Generalizability of SVM based Decoding in High Dimensional Neuroimage Data
NIPS Workshop: Machine Learning and Interpretation in Neuroimaging (MLINI-2011)

Decision function mis-match in the SVM (fMRI)

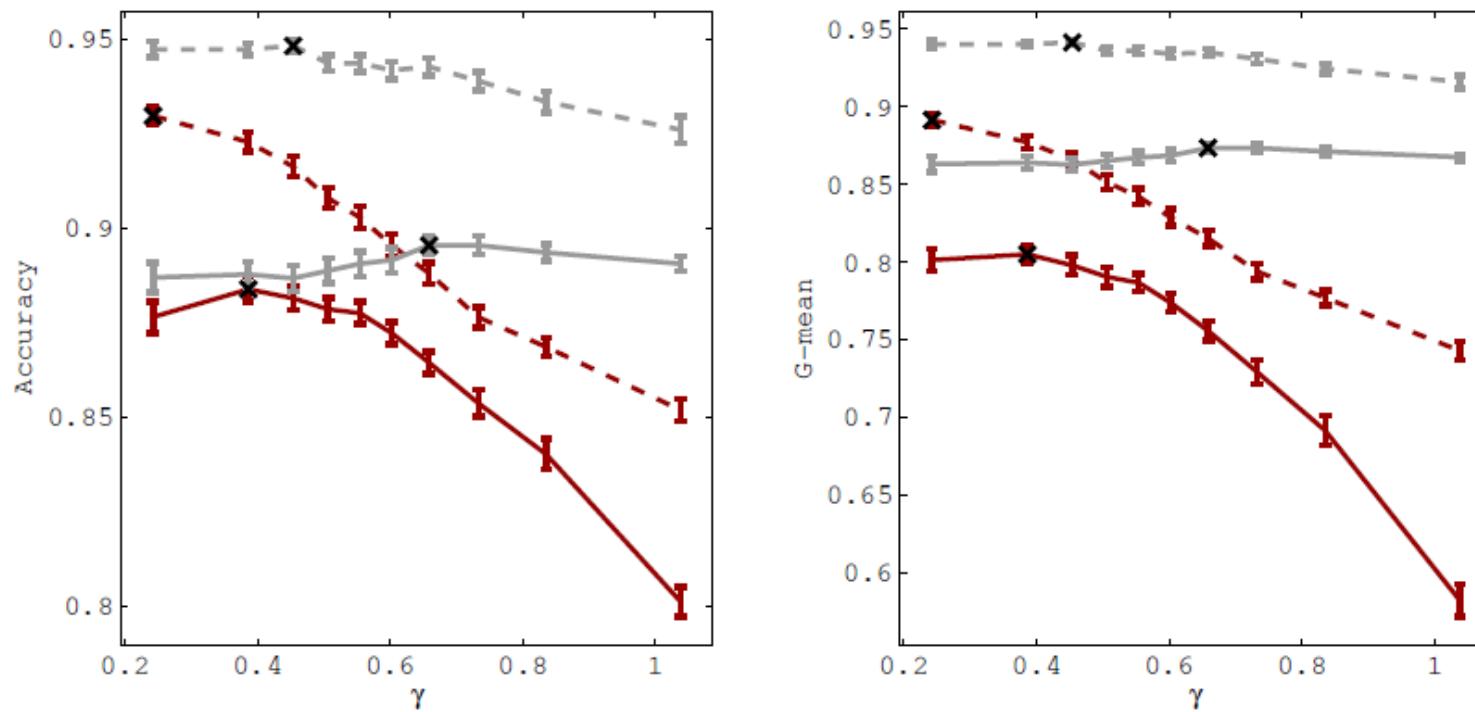
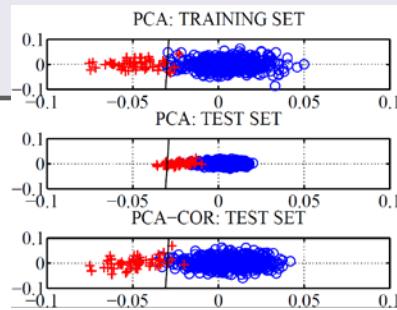


Fig. 2. Mean performance measures ± 1 std as a function of kernel hyperparameter for the fMRI data. Higher values of γ lead to more non-linear kernel embeddings. The left and right panel shows the accuracy and the G-mean respectively. The dashed lines correspond to the scheme where data with no stimuli are omitted, while the full lines show the performance on the subsampled data. The test accuracy is shown in red while the renormalized test accuracy is shown in gray. The black crosses indicate the optimal kernel hyperparameter. Renormalization is seen to improve performance and notably it leads to more non-linear optimal kernels as the optimal scale parameters chosen by cross-validation are increased.

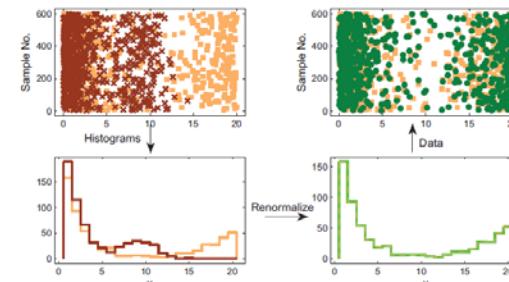
$$\gamma = 1/c$$

Conclusion

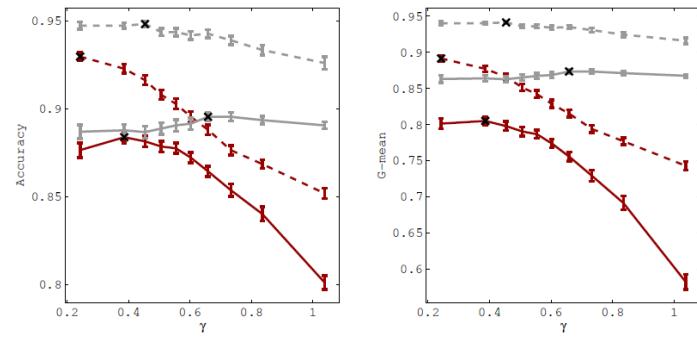
- Variance inflation in PCA
Cure: Rescale std's



- Variance inflation in kPCA
Cure: Non-parametric
renormalization of components



- Support Vector Machines:
In-line renormalization seems to enable
more non-linear classifiers in $D \gg N$



Acknowledgments

Lundbeck Foundation (www.cimbi.org)
NIH Human Brain Project grant (P20 MH57180)
Danish Research Councils

