An Introduction to Geometric Data Analysis and its Possible Applications

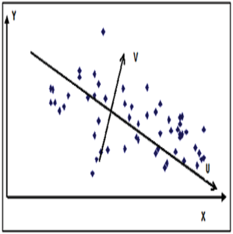
JEN-MEI CHANG

Department of Mathematics and Statistics
California State University, Long Beach
jchang9@csulb.edu

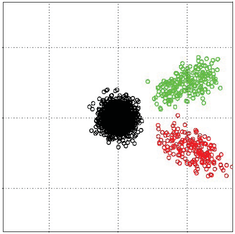
Claremont Colleges Mathematics Colloquia

Outline

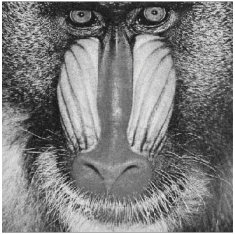
Why analysis?



Representation

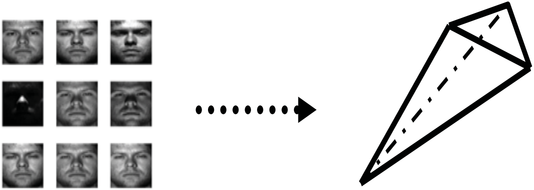


Visualization

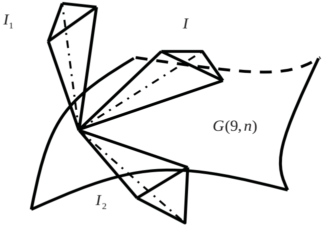


Applications

Why synthesis?



Model building



Prediction and classification

Full SVD

Definition

(**Full SVD**) Any $m \times n$ real matrix A, with $m \geq n$, can be factorized into

$$A = U \begin{pmatrix} \Sigma \\ 0 \end{pmatrix} V^T,$$

where $U \in \mathbb{R}^{m \times m}$ and $V \in \mathbb{R}^{n \times n}$ are orthogonal, and $\Sigma \in \mathbb{R}^{n \times n}$ is diagonal with

$$\Sigma = \mathrm{diag}(\sigma_1, \sigma_2, \ldots, \sigma_n), \; \sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0.$$

Thin SVD

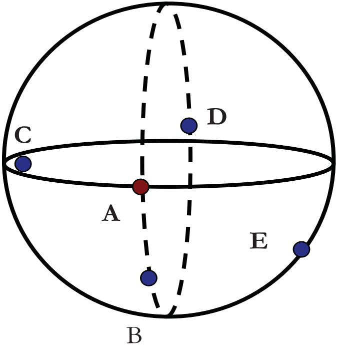
Definition

(**Thin SVD**) With the partitioning $U = (U_1, U_2)$, where $U_1 \in \mathbb{R}^{m\times n}$, we get the *thin SVD*
$$A = U_1 \Sigma V^T,$$

Structural Illustration:

$$A = U_1 \Sigma V^T = (u_1\ u_2\ \cdots\ u_n) \begin{pmatrix} \sigma_1 & & & \\ & \sigma_2 & & \\ & & \ddots & \\ & & & \sigma_n \end{pmatrix} \begin{pmatrix} v_1^T \\ v_2^T \\ \vdots \\ v_n^T \end{pmatrix} = \sum_{i=1}^{n} \sigma_i u_i v_i^T.$$

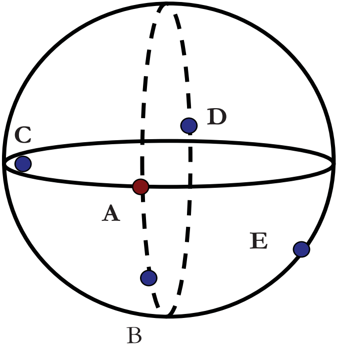
Distance



What is A closest to?

- No geometry: D
- With geometry: B

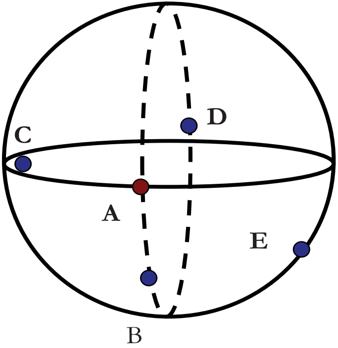
Distance



What is A closest to?

- No geometry: D
- With geometry: B

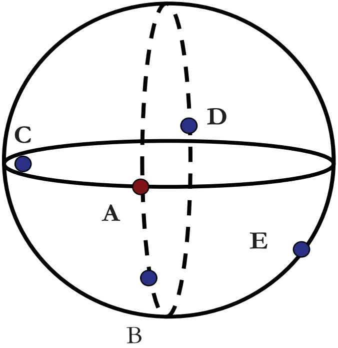
Distance



What is A closest to?

- No geometry: D
- With geometry: B

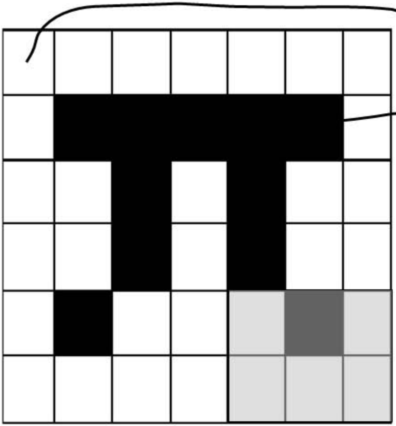
Distance



What is A closest to?

- No geometry: D
- With geometry: B

Data matrix



$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

Data vector

$$
\begin{bmatrix}
1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 0 & 0 & 0 & 0 & 0 & 1 \\
1 & 1 & 0 & 1 & 0 & 1 & 1 \\
1 & 1 & 0 & 1 & 0 & 1 & 1 \\
1 & 0 & 1 & 1 & 1 & 0 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 & 1
\end{bmatrix}
\longrightarrow
\begin{bmatrix}
1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ \hline
1 \\ 0 \\ 1 \\ 1 \\ 0 \\ 1 \\ \hline
\vdots \\ \hline
1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1
\end{bmatrix}
$$

$\boxed{\text{IMAGE} \longrightarrow \text{MATRIX} \longrightarrow \text{VECTOR}}$

Approximation theorem

If we know the correct rank of A, e.g., by inspecting the singular values, then we can **remove the noise and compress the data** by approximating A by a matrix of the correct rank. One way to do this is to truncate the singular value expansion:

Theorem

If

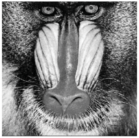
$$A_k = \sum_{i=1}^{k} \sigma_i u_i v_i^T \quad (1 \leq k \leq r)$$

then

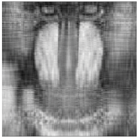
$$A_k = \min_{\mathrm{rank}(B)=k} \|A - B\|_2 \quad \textit{and} \quad A_k = \min_{\mathrm{rank}(B)=k} \|A - B\|_F \,.$$

An example

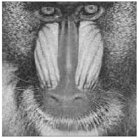
The error term of rank k approximation is given by the $(k+1)^{\text{th}}$ singular value σ_{k+1}.



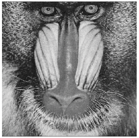
(a) full rank (rank 480)



(b) rank 10, rel. err. = 0.0551



(c) rank 50, rel. err. = 0.0305



(d) rank 170, rel. err. = 0.0126

General classification paradigm

1.5 **Preprocessing**
Geometric normalization,
Feature extraction, etc

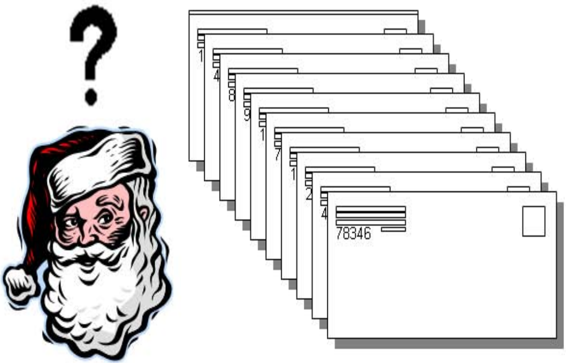
1. **Data collection**
Database creation
a.k.a. **Gallery**

2. **Present novel data**
a.k.a. **Probe**

3. **Classification**
Assign label to probe
and assess accuracy

Problem definition - globally

Santa thought to himself, “only if these mails can go to the right place according to their zip code”.



Handwritten digit classification

Problem. (Human) handwritten digits are sometimes very hard to recognize, even by human operators.

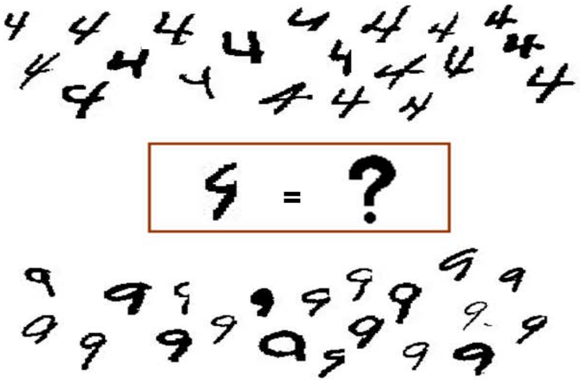
Importance. Accurate identification of the digits ensures a reliable delivery system.

Beneficiaries. Postal services (mail sorting), seaports (cargo registration), etc.

Even Santa Clause can benefit from an efficient digit classification algorithm.

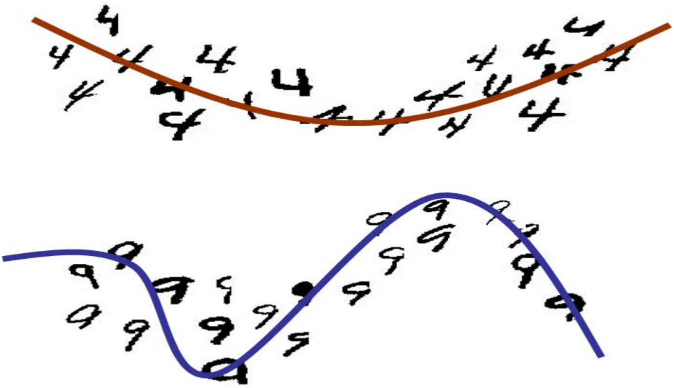
Problem definition - locally

How do we tell whether a new digit is a 4 or a 9?



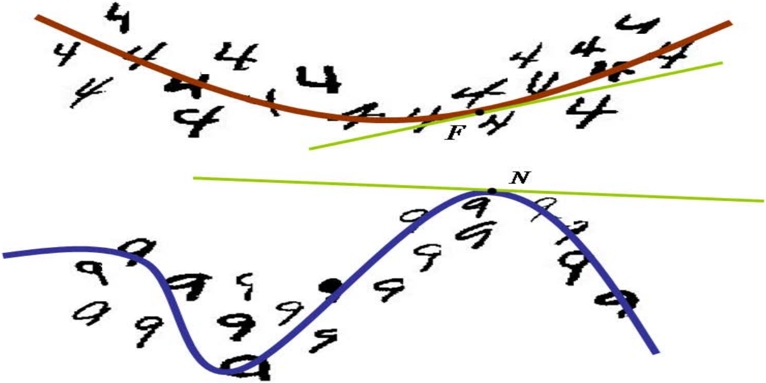
Digit manifolds

Imagine a high-D surface (red curve) where all 4's live on and a high-D surface (blue curve) where all 9's live on.



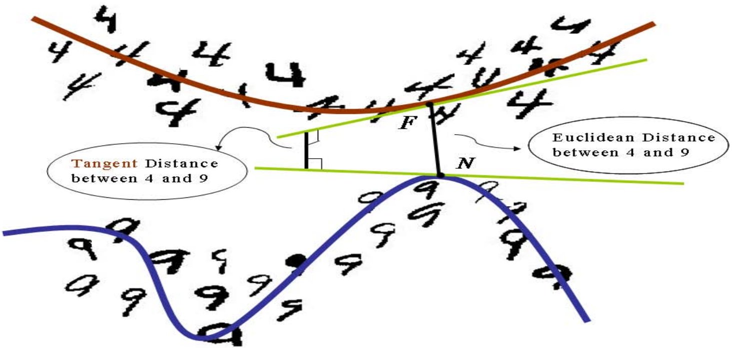
Tangent spaces - training

Create a Tangent Space of the 4's at F and create a Tangent Space of the 9's at N.



Dimensions of the tangent spaces depend on the degree of variations.

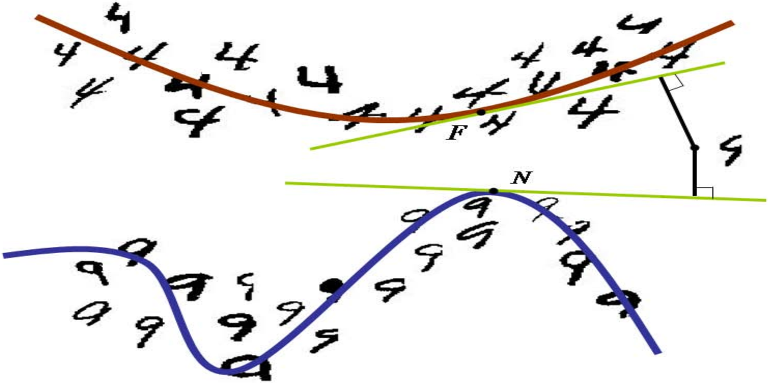
Distances



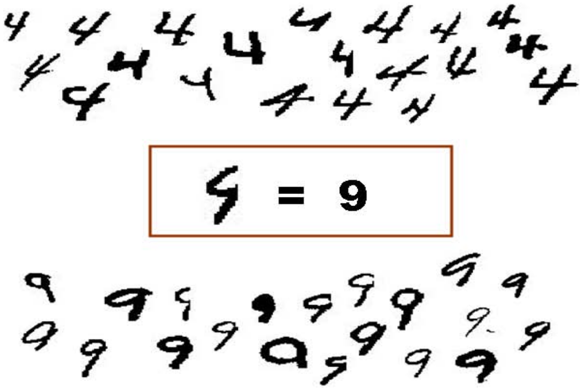
- Euclidean distance between each pair of 4 and 9 varies drastically while tangent distance captures the geometry and is less susceptible to variations.
- Pairwise Euclidean distance is time consuming while the tangent calculation is very efficient.

Classification

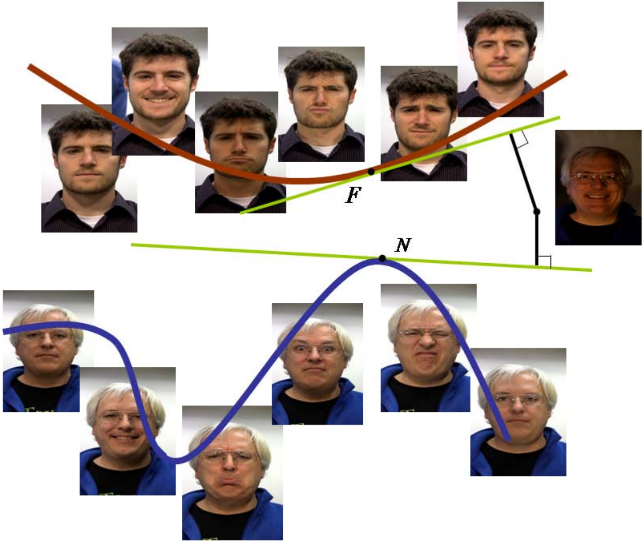
So, is it a 4 or a 9?



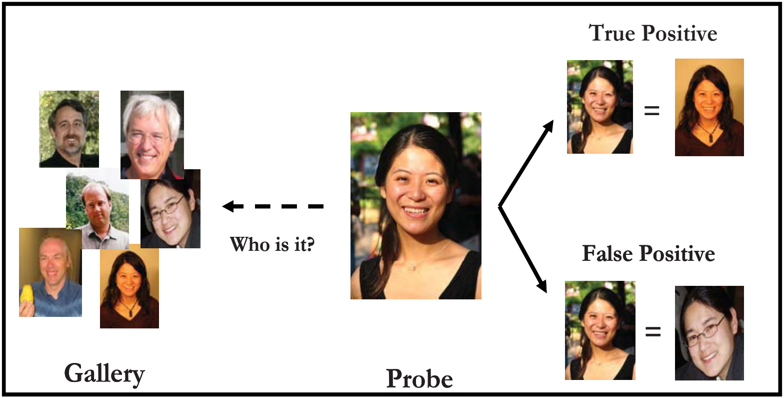
Classification result



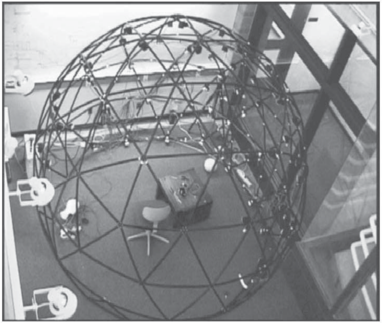
Face recognition



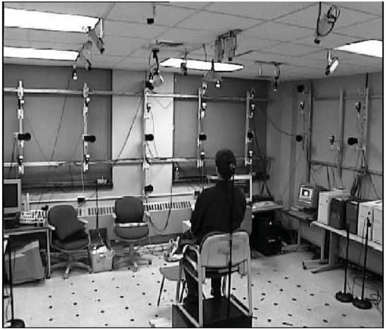
Face recognition paradigm



Illumination apparatus



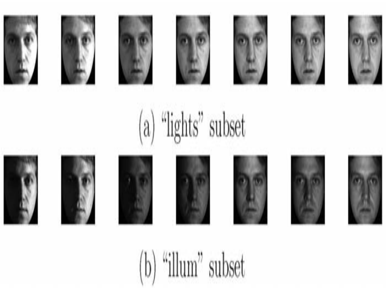
Yale Face Database B



CMU-PIE

Illumination images

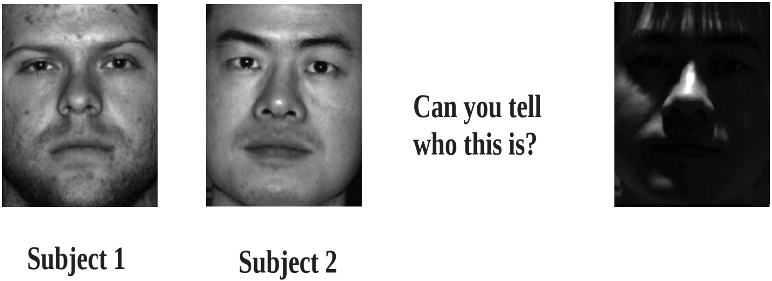
Yale Face Database B



CMU-PIE

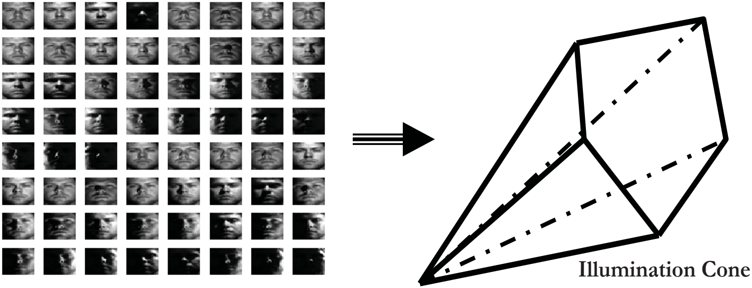
Empirical fact

Images of a single person seen under variations of illumination appear to be more difficult to recognize than images of different people [Zhao et al., 2003].



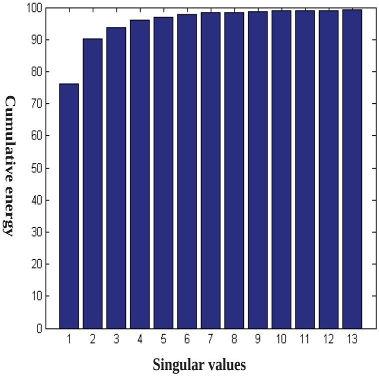
Geometric facts - 1

The set of m-pixel monochrome images of an object seen under general lighting conditions forms a convex polyhedral cone (illumination cone) in $\mathbb{R}^m$ [Belhumeur & Kriegman, 1998].

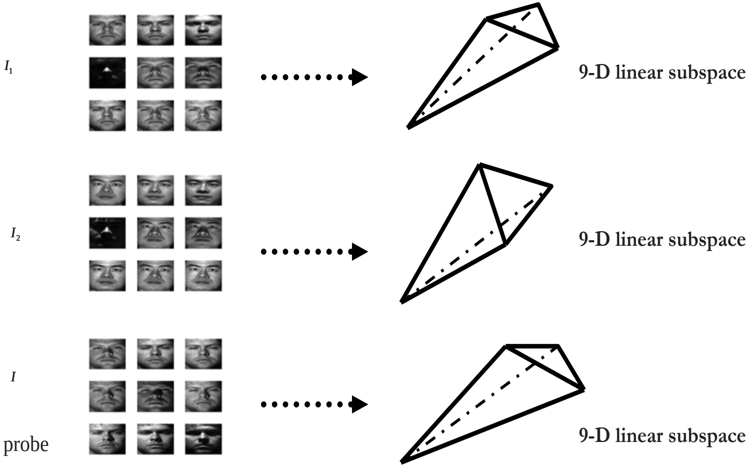


Geometric facts - 2

The illumination cone can be approximated by a 9-dimensional linear subspace [Basri & Jacobs, 2003], i.e., the illumination cone is low-dimensional and linear.

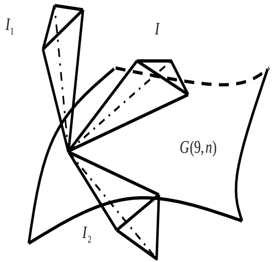


Set-up


I_1
9-D linear subspace
I_2
9-D linear subspace
I
probe
9-D linear subspace

Definition of $G(k, n)$

These illumination cones are all elements of a parameter space called the **Grassmannian (Grassmann manifold)**, $G(9, n)$, where n in the ambient dimension.



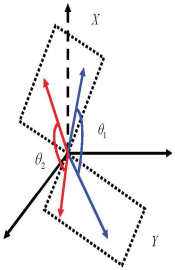
Definition

The *Grassmannian G(k,n)* or the *Grassmann manifold* is the set of k-dimensional subspaces in an n-dimensional vector space K^n for some field K, i.e.,

$$G(k, n) = \{W \subset K^n \mid \dim(W) = k\}.$$

Principal angles [Björck & Golub, 1973]

It turns out that any attempt to construct an unitarily invariant metric on $G(k, n)$ yields something that can be expressed in terms of the **principal angles** [Stewart & Sun, 1990].



Definition

If X and Y are two vector subspaces of $\mathbb{R}^m$, then the principal angles $\theta_k \in \left[0, \frac{\pi}{2}\right]$, $1 \leq k \leq q$ between X and Y are defined recursively by

$$\cos(\theta_k) = \max_{u \in X} \max_{v \in Y} u^T v = u_k^T v_k$$

subject to $\|u\| = \|v\| = 1$, $u^T u_i = 0$ and $v^T v_i = 0$ for $i = 1 : k-1$ and $q = \min\{\dim(X), \dim(Y)\} \geq 1$.

Grassmannian distances [Edelman et al., 1999]

These are the distance functions we will use to compare points on the Grassmann manifold.

Metric Name	Mathematical Expression
Fubini-Study	$d_{FS}(\mathcal{X},\mathcal{Y}) = \cos^{-1}\left(\prod_{i=1}^{k}\cos\theta_i\right)$
Chordal 2-norm	$d_{c2}(\mathcal{X},\mathcal{Y}) = \left\|2\sin\frac{1}{2}\theta\right\|_F$
Chordal F-norm	$d_{cF}(\mathcal{X},\mathcal{Y}) = \left\|2\sin\frac{1}{2}\theta\right\|_2$
Geodesic (Arc Length)	$d_g(\mathcal{X},\mathcal{Y}) = \|\theta\|_2$
Chordal (Projection F-norm)	$d_c(\mathcal{X},\mathcal{Y}) = \|\sin\theta\|_2$
Projection 2-norm	$d_{p2}(\mathcal{X},\mathcal{Y}) = \|\sin\theta\|_\infty$

Empirical result - database

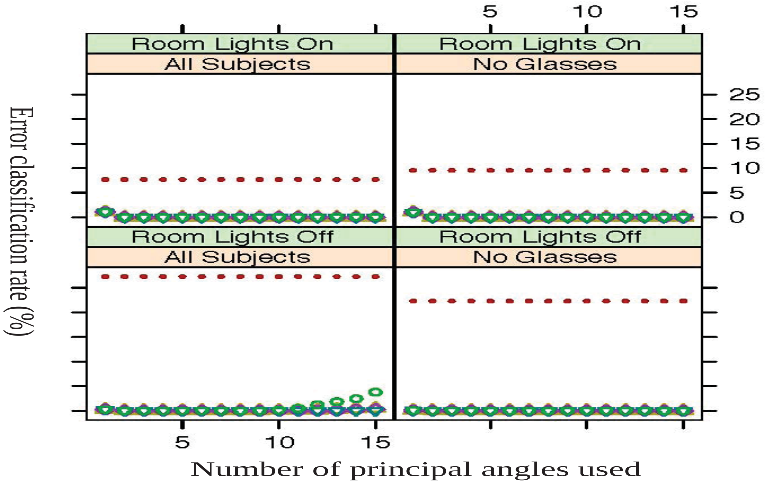
Since we are only concerned with the lighting variations, we fix the frontal pose, neutral expression and select the “illum” and “lights” subsets of CMU-PIE (68 subjects, 13 poses, 43 lightings, 4 expressions) [Sim et al., 2003] for experiments.

- lights: 21 illumination conditions with background lights **on.**
- illum: 21 illumination conditions with background lights **off.**

(a) “lights” subset

(b) “illum” subset

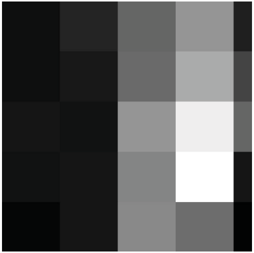
Empirical results



Robustness

If the data set is perfectly separable with the Grassmann method when using this kind of image [Chang et al., 2006a]:

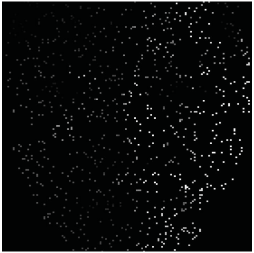
The data set is still perfectly separable with the Grassmann method when using this kind of image [Chang et al., 2007bc]:



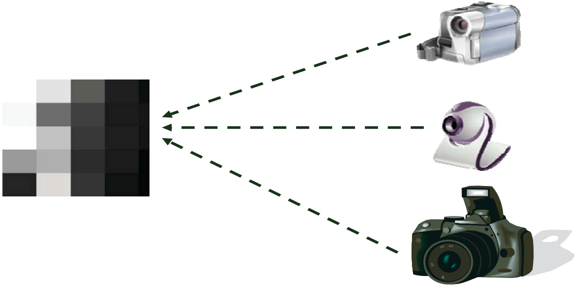
Robustness

If the data set is perfectly separable with the Grassmann method when using this kind of image [Chang et al., 2006a]:

The data set is still perfectly separable with the Grassmann method when using this kind of image [Chang et al., 2007bc]:

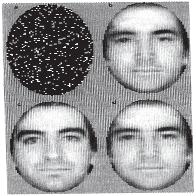


Potential use: low-res. illumination camera



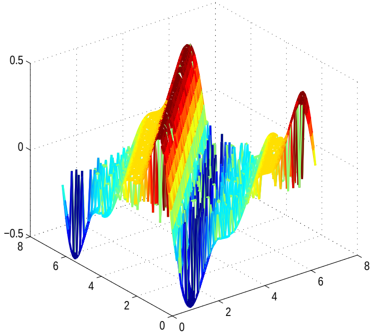
Large private databases of facial imagery can be stored at a resolution that is sufficiently low to prevent recognition by a human operator yet sufficiently high to enable machine recognition.

KL procedure for missing data

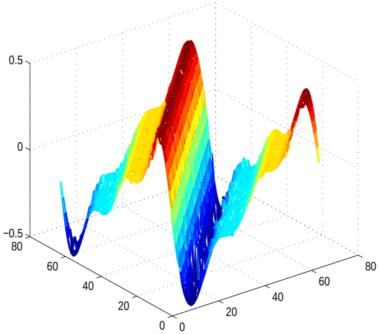


1. Initialize the missing data with the ensemble average.
2. Compute the first estimate of the KL basis.
3. Re-estimate the ensemble using the gappy approximation and the KL basis.
4. Re-compute the KL basis.
5. Repeat Steps 3–4 until stopping criterion is satisfied.

A gappy example

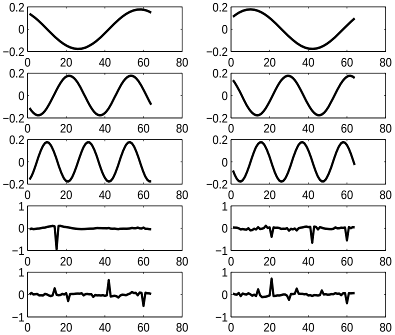


Gappy data

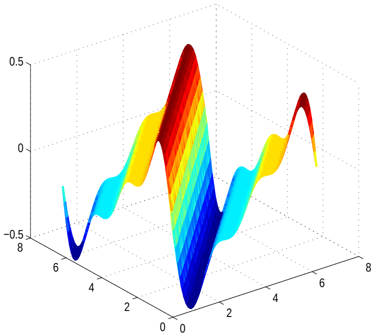


After 1 repair

Gappy example continued



Eigenvectors of repaired data



Repaired

Bankruptcy prediction is the art of predicting bankruptcy and various measures of financial distress of public firms. It is a vast area of finance and accounting research. The importance of the area is due in part to the relevance for creditors and investors in evaluating the likelihood that a firm may go bankrupt[1].

- If we form a feature vector for each firm.
- The problem becomes a two-class classification problem.

[1] adapted from Wikipedia

Bankruptcy prediction is the art of predicting bankruptcy and various measures of financial distress of public firms. It is a vast area of finance and accounting research. The importance of the area is due in part to the relevance for creditors and investors in evaluating the likelihood that a firm may go bankrupt[1].

- If we form a feature vector for each firm.
- The problem becomes a two-class classification problem.

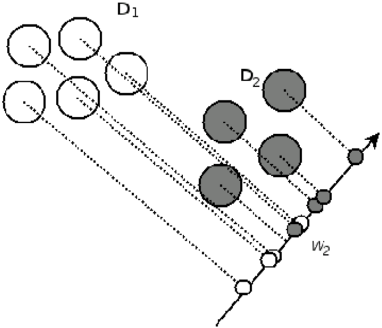
[1] adapted from Wikipedia

Bankruptcy prediction is the art of predicting bankruptcy and various measures of financial distress of public firms. It is a vast area of finance and accounting research. The importance of the area is due in part to the relevance for creditors and investors in evaluating the likelihood that a firm may go bankrupt[1].

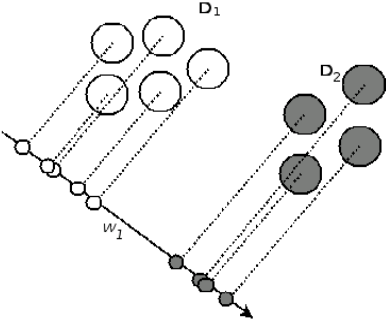
- If we form a feature vector for each firm.
- The problem becomes a two-class classification problem.

[1] adapted from Wikipedia

Linear Discriminant Analysis



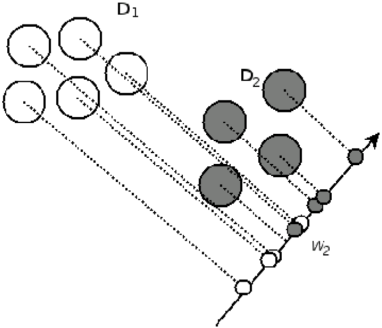
Bad projection



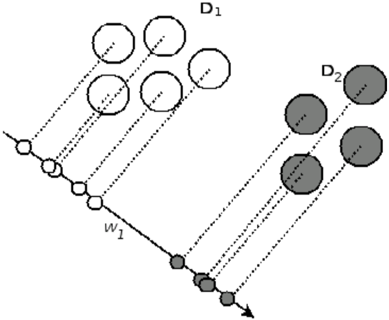
Good projection

Question: Characteristics of a GOOD projection?

Linear Discriminant Analysis



Bad projection

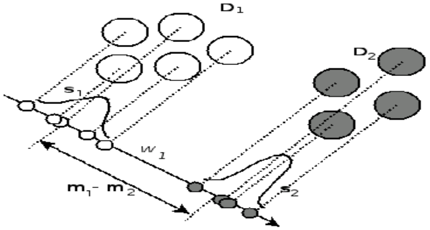


Good projection

Question: Characteristics of a GOOD projection?

Two-Class LDA

$$m_1 = \frac{1}{n_1} \sum_{x \in D_1} w^T x, \quad m_2 = \frac{1}{n_2} \sum_{y \in D_2} w^T y$$



Look for a projection w that

- maximizes (inter-class) distance in the projected space,
- and minimizes the (intra-class) distances in the projected space.

Two-Class LDA

Namely, we desire a w^* such that

$$w^* = \arg\max_w \frac{(m_1 - m_2)^2}{S_1 + S_2},$$

where $S_1 = \sum_{x \in D_1} (w^T x - m_1)^2$ and $S_2 = \sum_{y \in D_2} (w^T y - m_2)^2$.

Alternatively, (with scatter matrices)

$$w^* = \arg\max_w \frac{w^T S_B w}{w^T S_W w}, \tag{1}$$

with $S_W = \sum_{i=1}^{2} \sum_{x \in D_i} (x - \mathbf{m}_i)(x - \mathbf{m_i})^T$, $S_B = (\mathbf{m}_2 - \mathbf{m}_1)(\mathbf{m}_2 - \mathbf{m}_1)^T$.

Two-Class LDA

Namely, we desire a w^* such that

$$w^* = \arg\max_w \frac{(m_1 - m_2)^2}{S_1 + S_2},$$

where $S_1 = \sum_{x \in D_1} (w^T x - m_1)^2$ and $S_2 = \sum_{y \in D_2} (w^T y - m_2)^2$.
Alternatively, (with scatter matrices)

$$w^* = \arg\max_w \frac{w^T S_B w}{w^T S_W w}, \tag{1}$$

with $S_W = \sum_{i=1}^{2} \sum_{x \in D_i} (x - \mathbf{m}_i)(x - \mathbf{m_i})^T$, $S_B = (\mathbf{m}_2 - \mathbf{m}_1)(\mathbf{m}_2 - \mathbf{m}_1)^T$.

LDA

The criterion in Equation (1) is commonly known as the generalized Rayleigh quotient, whose solution can be found via the generalized eigenvalue problem

$$S_B w = \lambda S_W w.$$

LDA for multi-class follows similarly.

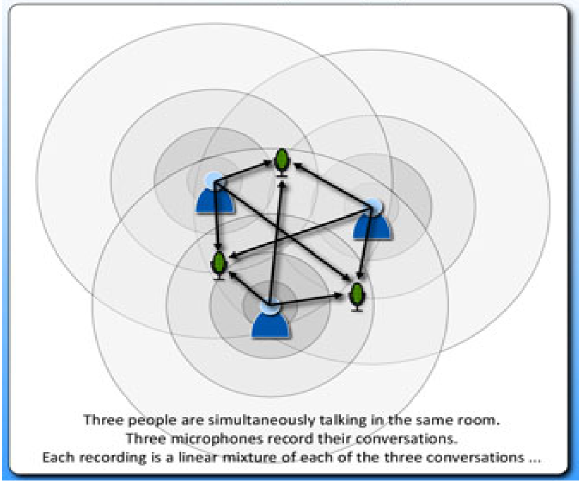
LDA

The criterion in Equation (1) is commonly known as the generalized Rayleigh quotient, whose solution can be found via the generalized eigenvalue problem

$$S_B w = \lambda S_W w.$$

LDA for multi-class follows similarly.

Cocktail Party Problem



(adapted from André Mouraux)

KL procedure for noisy data

- Decompose observed data into its *noise* and *signal* components:

$$\mathbf{x}^{(\mu)} = \mathbf{s}^{(\mu)} + \mathbf{n}^{(\mu)},$$

or, in terms of data matrices,

$$X = S + N. \quad (S = \text{ signal }, N = \text{ noise })$$

- The optimal first basis vector, ϕ, is taken as a superposition of the data, i.e.,

$$\phi = \psi_1 \mathbf{x}^{(1)} + \cdots + \psi_P \mathbf{x}^{(P)} = X\psi.$$

- May decompose ϕ into signal and noise components

$$\phi = \phi_\mathbf{n} + \phi_\mathbf{s},$$

where $\phi_\mathbf{s} = S\psi$ and $\phi_\mathbf{n} = N\psi$.

MNF/BBS

- The basis vector ϕ is said to have maximum noise fraction (MNF) if the ratio

$$D(\phi) = \frac{\phi_{\mathbf{n}}^T \phi_{\mathbf{n}}}{\phi^T \phi}$$

is a maximum.

- A steepest descent method yields the *symmetric definite generalized eigenproblem*

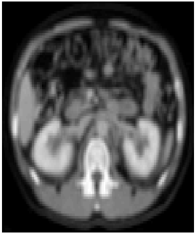
$$N^T N \psi = \mu^2 X^T X \psi.$$

This problem may be solved without actually forming the product matrices $N^T N$ and $X^T X$, using the generalized SVD (gsvd).

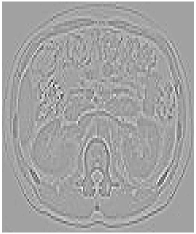
- Note that the same orthonormal basis vector ϕ optimizes the signal-to-noise ratio. And this technique is called **Blind Source Separation (BSS).**

Convolution - sharpening

$$
\begin{aligned}
w(x,y) \star f(x,y) &= \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t) f(x-s, y-t) \\
&= \sum_{s=-a}^{a} \sum_{t=-b}^{b} f(s,t) w(x-s, y-t)
\end{aligned}
$$



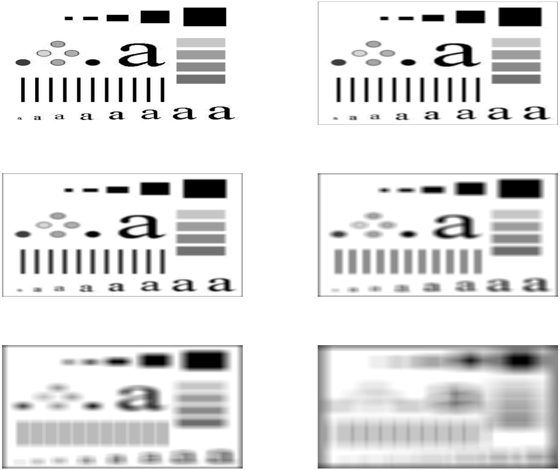
A blurred image



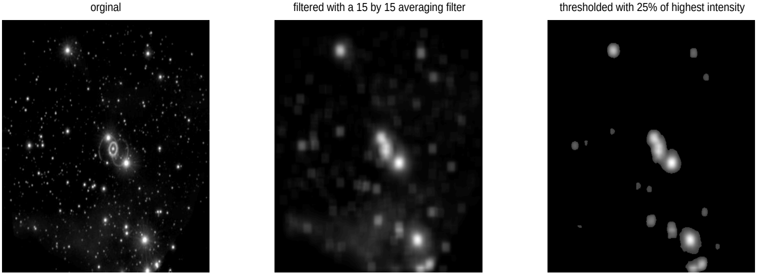
Laplacian edge filter

Enhanced image

Convolution - smoothing

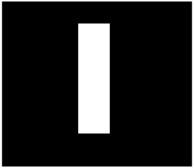


Convolution - threshold smoothing

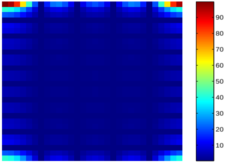


Fourier analysis

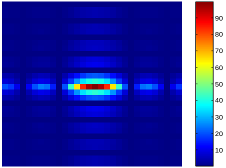
$$F(u,v) = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)e^{-i2\pi\left(\frac{ux}{M}+\frac{vy}{N}\right)}$$



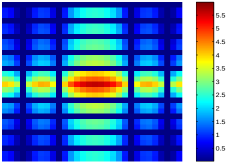
(a) Image.



(b) Spectrum.



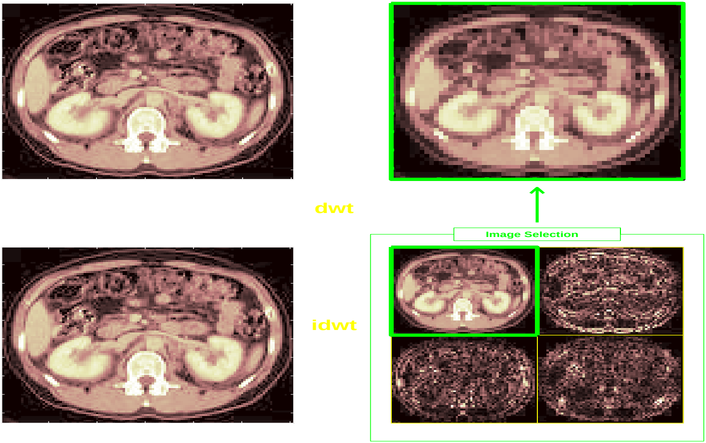
(c) Centered spectrum.



(d) log transform

Multiresolution analysis

$$X(b,a) = \frac{1}{\sqrt{|a|}} \int_{-\infty}^{\infty} x(t)\Psi^{*}\left(\frac{t-b}{a}\right)\, dt$$



References

[Basri & Jacobs, 2003] R. Basri & D. Jacobs, “Lambertian reflectance and linear subspaces”, *PAMI, 25(2):218–233, 2003*.

[Belhumeur & Kriegman, 1998] P. Belhumeur & D. Kriegman, “What is the set of images of an object under all possible illumination conditions”, *IJCV, 28(3):245–260, 1998*.

[Björck & Golub, 1973] A. Björck & G. Golub, “Numerical methods for computing angles between linear subspaces”, *Mathematics of Computation, 27(123):579–594, 1973*.

[Chang et al., 2006a] J.-M. Chang, M. Kirby, H. Kley, J. R. Beveridge, C. Peterson, B. Draper, “Illumination face spaces are idiosyncratic”, *Int’l Conf. on Image Proc. & Comp. Vision, 2: 390–396, 2006*.

[Chang et al., 2007b] J.-M. Chang, M. Kirby, H. Kley, J. R. Beveridge, C. Peterson, B. Draper, “Recognition of digital images of the human face at ultra low resolution via illumination spaces”,*ACCV’07, LNCV, Springer, 4844: 733–743, 2007*.

References

[Chang et al., 2007c] J.-M. Chang, M. Kirby, C. Peterson, "Feature Patch Illumination spaces and Karcher compression for face recognition via Grassmannian", *under review, 2009*.

[Edelman et al., 1999] A. Edelman, T. Arias, & S. Smith, "The Geometry of algorithms with orthogonality constraints", *SIAM J. Matrix Anal. Appl., 20(2):303–353, 1999*.

[Stewart & Sun, 1990] G.W. Stewart & J.-G. Sun, "Matrix Perturbation Theory", *Academic Press, 1990*.

[Zhao et al., 2003] W. Zhao, R. Chellappa, P. J. Phillips, A. Rosenfeld, "Face recognition: A literature survey". *ACM Comp. Surv., 35(4):399–458, 2003*.