Lecture 9: Dimensionality Reduction: PCA, LDA, Fisher

STAT261: Introduction to Machine Learning

Lecture 9, April 25

UCLA

Outline

- Dimensionality Reduction: Linear Approaches
  - Review: linear transformation of Gaussians
  - Review: projection and least squares
- PCA
  - Last time, derive probabilistic "maximal" variance basis
  - Maximize variance of projection of data
  - Minimal approximation error from projection
- SVD
  - Definition & illustration
  - Relate to PCA
- Classification from PCA
- LDA

Dimensionality Reduction: Linear Methods

- Many data sets are high-dimensional

- Want to reduce dimension:
  - Operations become computationally simpler
  - Extracts meaningful component of data
  - Removes noise
  - Simpler to visualize data

- Linear representation toward dimensionality reduction
  - Data $\boldsymbol{x}$ is p-dimensional
  - Find K-dimensional representation: $K \ll p$

$$\boldsymbol{x} \approx \mu + \sum_{k=1}^{K} \alpha_k \mathrm{v}_k$$

  - Approximately express each data vector by K numbers

Review: Jointly Gaussian PDF

- If $\boldsymbol{X}$ is (jointly) Gaussian, then pdf is

$$f(x) = \frac{1}{(2\pi)^{n/2} det^{1/2}(P)} \times \exp\left\{-\frac{1}{2}(x-\mu)^* P^{-1}(x-\mu)\right\}$$

- Gaussian characterized by mean and variance matrix
  - $\mu = E(X), \ P = var(X)$
- Special cases:
  - $n = 1$
  - Independent Gaussian

Bivariate Gaussian ($n = 2$)

- X and Y are jointly Gaussian and zero mean then, pdf is:

$$f(x,y) = \frac{1}{2\pi\sigma_x\sigma_y\sqrt{1-\rho^2}} \times \exp\left[-\frac{1}{2(1-\rho^2)}\left(\frac{x^2}{\sigma_x^2} - \frac{2\rho xy}{\sigma_x\sigma_y} + \frac{y^2}{\sigma_y^2}\right)\right]$$

  - σ_x^2, σ_y^2= variance of X and Y
  - $\rho =$ correlation coefficient

- For non-zero mean replace
  - x with $x - \mu_x$ and y with $y - \mu_y$

Linear Transforms of Gaussian

- Suppose $Y = AX + b$
- Then, Y is also Gaussian with:
  - $\mu_Y = A\mu_X + b$
  - $var(Y) = Avar(X)A^T$

Properties: Jointly Gaussian Random Vectors

- Definition: A random vector $\boldsymbol{X}$ is (jointly) Gaussian if and only if $\boldsymbol{a}^*\boldsymbol{X}$ is a Gaussian scalar for all non-random $\boldsymbol{a}$.

- Jointly Gaussian $\Longrightarrow$ Gaussian components
- But, not converse
- Independent Gaussian $\Longrightarrow$ Jointly Gaussian

- Generalization of 2-dim case

Basics Reminder: Inner Products, Norms & Outer Products

- Given vectors $x, y \in R^p$
- Inner product: $x^T y = \sum_{i=1}^{p} x_i y_i$
- Norm:

$$\|x\|^2 = \sum_{i=1}^{p} x_i^2$$

- Outer product:

$$M = xy^T = [x_i y_j]$$

  - $p \times p$ matrix

- Inner product: $x^T y = \sum_{i=1}^{p} x_i y_i$

Projection Operators** Important in Linear methods

- A matrix (or linear operator) P is a projection if:
  - $P^2 = P$ (idempotent)
  - $P = P^*$ (orthogonal, complex case) or $P = P^T$ (real case)
- Define
  - V = range space of P = $\{Px\}$
  - $V^\perp$ = orthogonal complement = $\{y | y \perp v, \text{for all } v \in V\}$

Orthonormal Representations

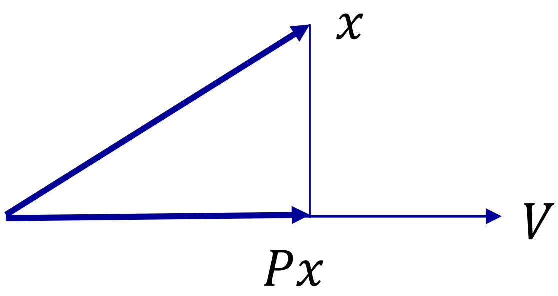
- Linear algebra fact: $\mathrm{v}_1, \dots, \mathrm{v}_p$ are an orthornormal basis

- A set of vectors $\mathrm{v}_1, \dots, \mathrm{v}_p$ are called orthonormal if
  - $\|v_k\|^2 = v_k^T v_k = 1$ for all k (all vectors are unit norm)
  - $v_j^T v_k = 0$ for all $j \neq k$ (different vectors are perpendicular)

- Representation property of orthonormal basis:
  - For any vector x
  - Representation property: $\mathrm{x} = \sum_{k=1}^{p} \alpha_k v_k$, $\alpha_k = \mathrm{v}_k^T \mathrm{x}$
  - Get coefficients in basis from inner product

Projection to a Range Space

- Let P be a projection with range space V
- Lemma (will prove on board):
  - $x \in V \Rightarrow Px = x$
  - $x \in V^\perp \Rightarrow Px = 0$
- P removes the component orthogonal to V
  - Write any vector as $x = u + v \in V \oplus V^\perp$
  - $Px = u$



Linear Least Squares as a Projection

- Least square $x = H\theta + w$
- $\hat{\theta} = \arg\min_{\theta} \|x - H\theta\|^2 = \left(H^T H\right)^{-1} H^T x$
- Let $\hat{x} = H\hat{\theta} = H\left(H^T H\right)^{-1} H^T x = Px$
- $P = H\left(H^T H\right)^{-1} H^T$
- Proposition: P is an orthogonal projection onto $\mathrm{Range}(H)$
- Proof: Can show the following four properties
  - $P^2 = P$
  - $P = P^T$
  - $\mathrm{Range}(H) \subseteq \mathrm{Range}(P)$
  - $\mathrm{Range}(P) \subseteq \mathrm{Range}(H)$

Outline

- Dimensionality Reduction: Linear Approaches
  - Review: linear transformation of Gaussians
  - Review: projection and least squares
- PCA
  - Last time, derive probabilistic "maximal" variance basis
  - Maximize variance of projection of data
  - Minimal approximation error from projection
- SVD
  - Definition & illustration
  - Relate to PCA
- Classification from PCA
- LDA

Dimensionality Reduction: PCA

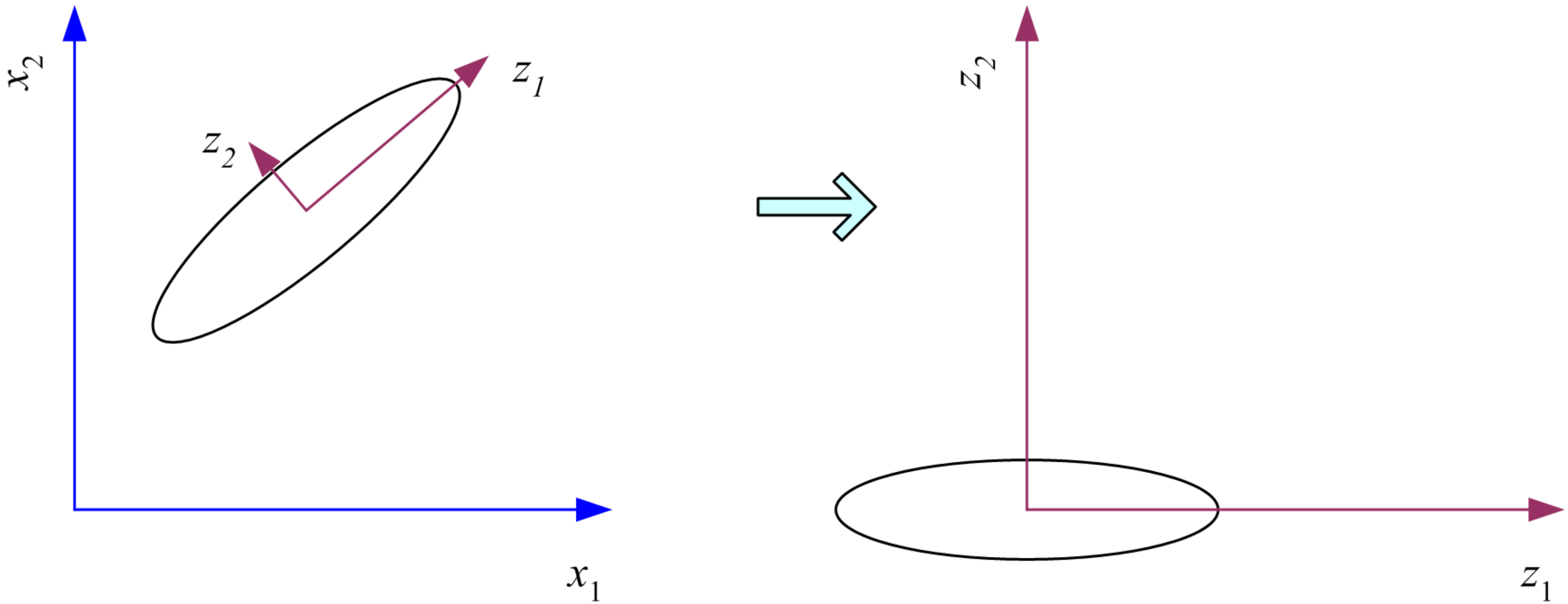
- Linear projection for dimensionality reduction
- Goal: keep most of the variance in the projected domain

- PCA for dimensionality reduction
  - Data $\boldsymbol{x}$ is p-dimensional
  - Find K-dimensional representation: $K \ll p$

$$\boldsymbol{x} \approx \mu + \sum_{k=1}^{K} \alpha_k \mathrm{v}_k$$

  - Approximately express each data vector by K numbers

What does PCA do?



- Transforms and shifts x to bases with maximal variation

Data and Sample Covariance Matrix

- Given data matrix:

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{x}_1^T \\ \vdots \\ \boldsymbol{x}_N^T \end{bmatrix} \quad N \times p$$

  - $\boldsymbol{x_i} = \left[x_{i1} \cdots x_{ip}\right]^T$
  - N records, p dimensions each
  - Each data record is a row of X
- Define
  - Mean: $\boldsymbol{\mu} = \frac{1}{N}\sum_{n=1}^{N} \mathrm{x}_j$, (mean for each column)
  - Sample covariance matrix $\boldsymbol{S} = \frac{1}{N}\sum_{i=1}^{N} (\mathrm{x}_i - \mu)\,(\mathrm{x}_i - \mu)^T$

PCA: End of last lecture, probabilistic…derivation

- Find a low-dimensional space such that when $\boldsymbol{x}$ is projected there, information loss is minimized.
- The projection of $\boldsymbol{x}$ on the direction of $\boldsymbol{w}$ is: $z = \boldsymbol{w}^T\boldsymbol{x}$
- Find $\boldsymbol{w}$ such that Var(z) is maximized

$$
\begin{aligned}
\mathrm{Var}(z) &= \mathrm{Var}(\boldsymbol{w}^T\boldsymbol{x}) = \mathrm{E}[(\boldsymbol{w}^T\boldsymbol{x} - \boldsymbol{w}^T\boldsymbol{\mu})^2] \\
&= \mathrm{E}[(\boldsymbol{w}^T\boldsymbol{x} - \boldsymbol{w}^T\boldsymbol{\mu})(\boldsymbol{w}^T\boldsymbol{x} - \boldsymbol{w}^T\boldsymbol{\mu})] \\
&= \mathrm{E}[\boldsymbol{w}^T(\boldsymbol{x} - \boldsymbol{\mu})(\boldsymbol{x} - \boldsymbol{\mu})^T\boldsymbol{w}] \\
&= \boldsymbol{w}^T\,\mathrm{E}[(\boldsymbol{x} - \boldsymbol{\mu})(\boldsymbol{x} - \boldsymbol{\mu})^T]\boldsymbol{w} = \boldsymbol{w}^T \sum \boldsymbol{w}
\end{aligned}
$$

where $\mathrm{Var}(\boldsymbol{x}) = \mathrm{E}[(\boldsymbol{x} - \boldsymbol{\mu})(\boldsymbol{x} - \boldsymbol{\mu})^T] = \sum$

…..

PCA: maximize empirical variance of the projection

- For each record, give a vector w, let $z_i = (x_i - \mu)^T w$
- Empirical variance in z_i (scalar)
- Var(z_i) $= \frac{1}{N} \sum_{n=1}^{N} w^T (x_n - \mu)(x_n - \mu)^T$w

$$\frac{1}{N} \sum |z_i|^2 = w^T S w$$

- **z**=X**w**, where is the inner product
- Maximal variance is the eigenvector of S
  - Brief explanation on board, similar to last time's probabilistic

- Principal components are the eigenvectors of *K*=$X^T X$

$$K v_k = \lambda_k \mathrm{v}_k$$

  - Order eigenvalues: $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p$
  - Assume $||\mathrm{v}_k|| = 1$

Approximation Property

- Let $v_1, \ldots, v_p$ be an orthonormal basis
- Any vector v has representation: $v = \sum_{k=1}^{p} \alpha_k v_k$,
- Given $K \leq p$, consider K-term approximation:

$$\hat{x} = \sum_{k=1}^{K} \alpha_k v$$

- Theorem: Error in K-term approximation is

$$\|x - \hat{x}\|^2 = \sum_{k=K+1}^{p} \alpha_k^2$$

The PCA Representation

- Write each record in terms of PCs:

$$\boldsymbol{x}_i = \boldsymbol{\mu} + \sum_{k=1}^{p} \alpha_{ik} \boldsymbol{v}_k \,, \qquad \alpha_{ik} = \boldsymbol{v}_k^T (\mathbf{x}_i - \boldsymbol{\mu})$$

- Obtain K-term approximation:

$$\widehat{\boldsymbol{x}}_n = \boldsymbol{\mu} + \sum_{k=1}^{K} \alpha_{nk} \boldsymbol{v}_k$$

- Each data record represented in terms of PCs

Measuring the Average Error

- Suppose we use a K term approximation
- Error in each data record is:

$$\|x_n - \hat{x}_n\|^2 = \sum_{k=K+1}^{p} \alpha_{nk}^2 = \sum_{k=K+1}^{p} v_k^T (x_n - \mu)(x_n - \mu)^T v_k$$

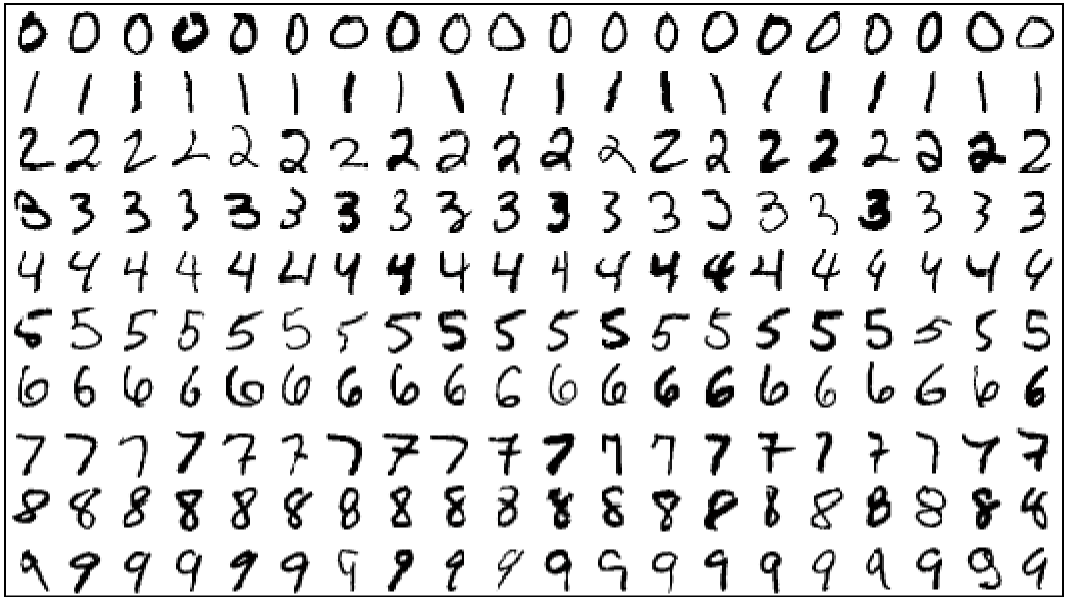
- Average error is:

$$J_K = \frac{1}{N}\sum_{n=1}^{N} \|x_n - \hat{x}_n\|^2 = \sum_{k=K+1}^{p} v_k^T \boldsymbol{K} v_k = \sum_{k=K+1}^{p} \lambda_k$$

- Percentage of Variance:

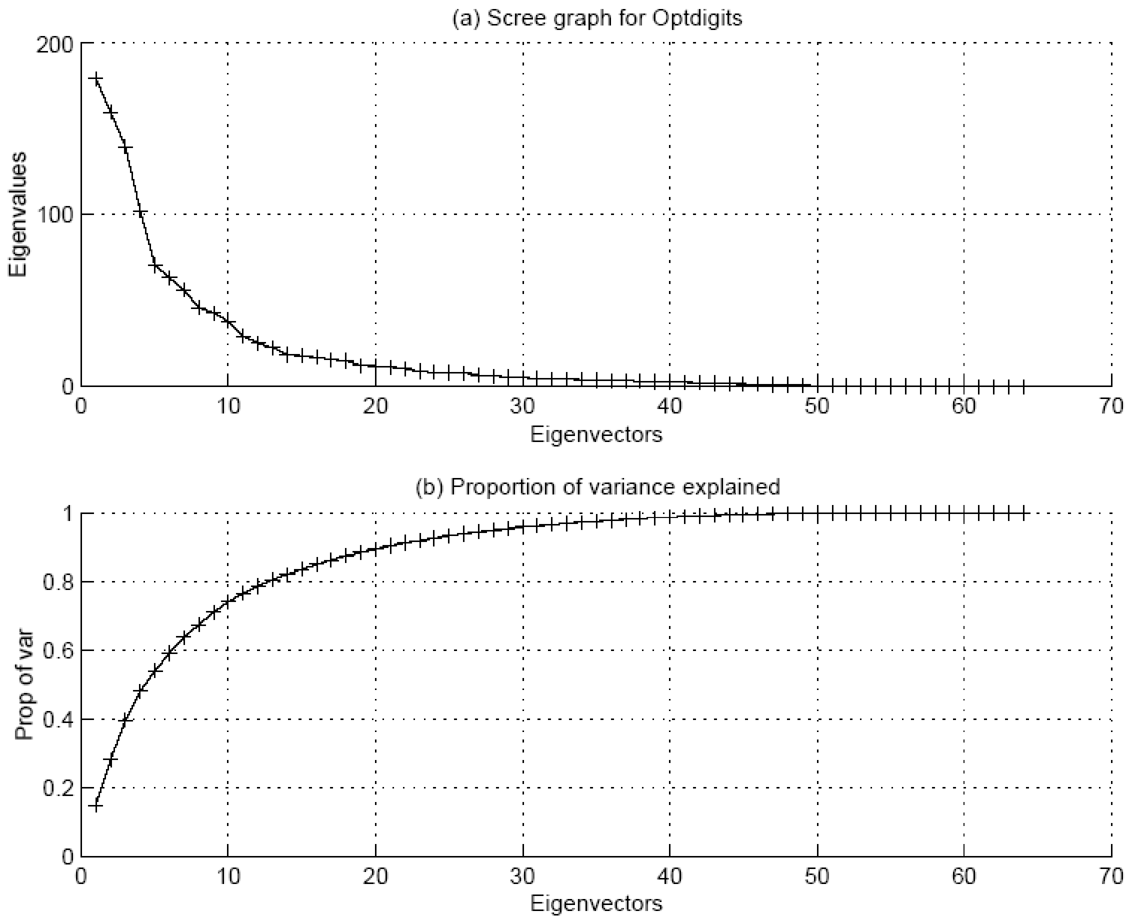
$$POV_K = 1 - \frac{J_K}{J_0} = \frac{\sum_{k=1}^{K} \lambda_k}{\sum_{k=1}^{p} \lambda_k}$$

Example: optical handwritten digits



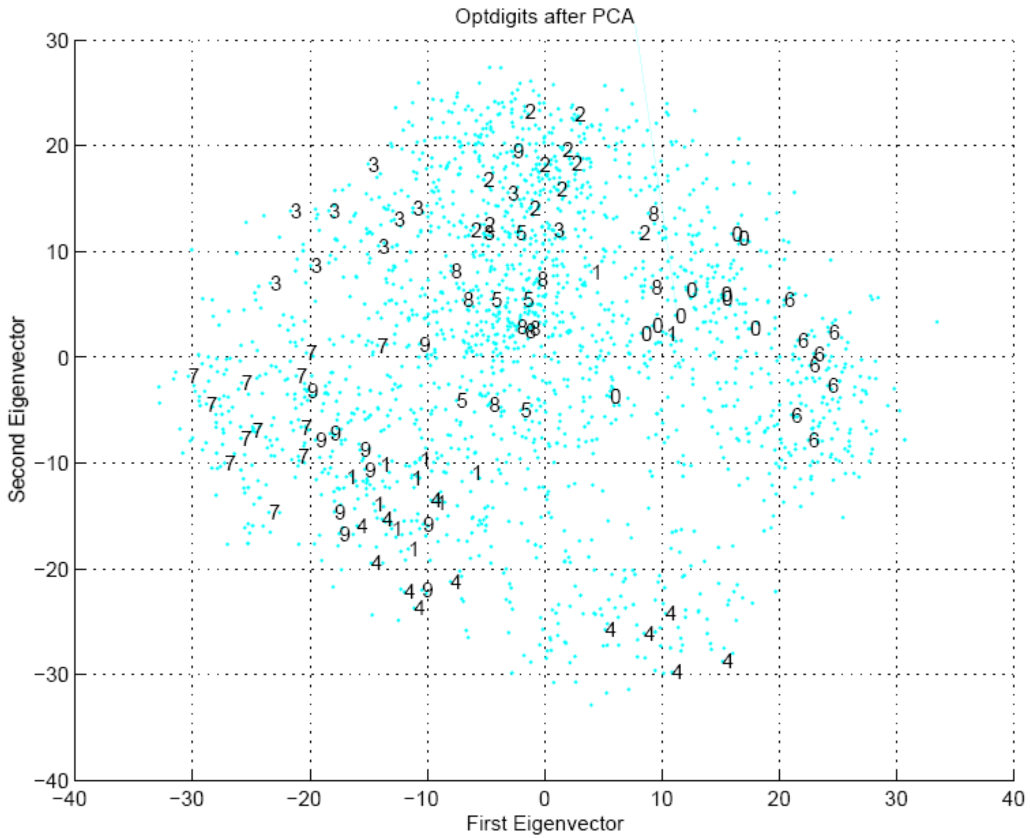
- Data set:images of handwritten digits

Example: OptDigits



- Handwritten digits
- Most variance is in 20 PCs

Visualizing Principal Components

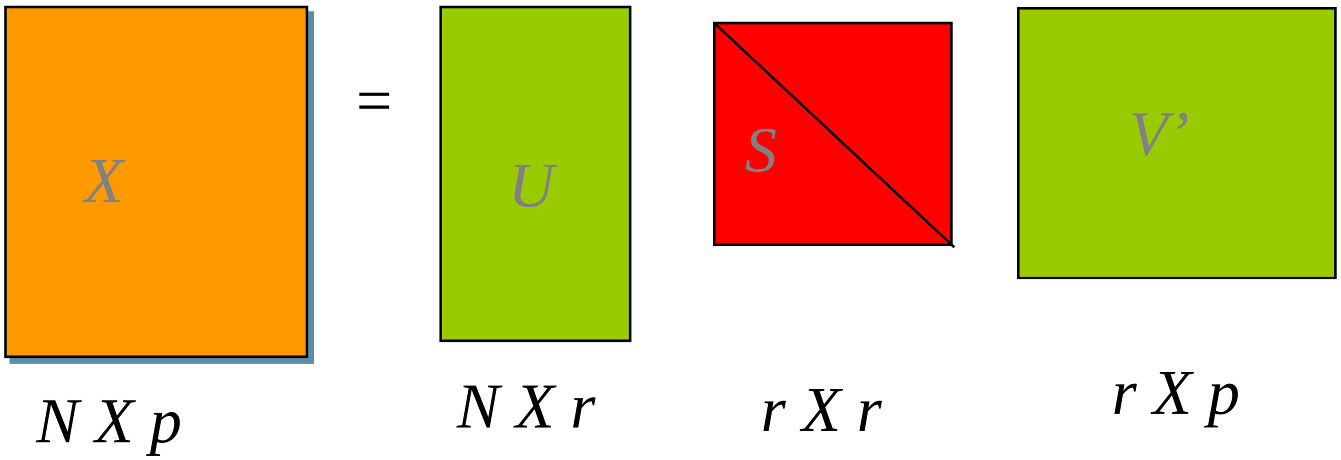


- Take coefficients along two PCs

Outline

- Dimensionality Reduction: Linear Approaches
  - Review: linear transformation of Gaussians
  - Review: projection and least squares
- PCA
  - Last time, derive probabilistic "maximal" variance basis
  - Maximize variance of projection of data
  - Minimal approximation error from projection
- SVD
  - Definition & illustration
  - Relate to PCA
- Classification from PCA
- LDA

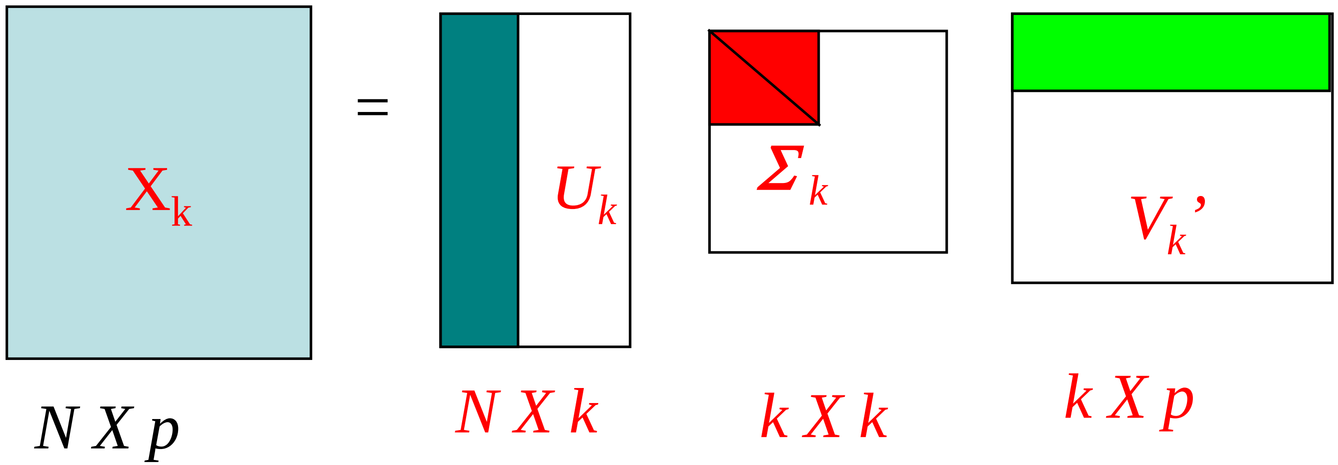
SVD: Mathematical Background



SVD: The mathematical formulation

- X N x p matrix of N p-dimrows, records, pt…
- SVD decomposition : $X = U\ \Sigma\ V^T$
  - U (N x N)
    - U is orthogonal: $U^T U = I$
    - Cols of U are the orthogonal e-vectors of XX^T
    - Left singular vectors of X
  - V(p x p)
    - V is orthogonal: $V^T V = I$
    - Cols of V are the orthogonal e-vectors of $X^T X$
    - Right singular vectors of X
    - Principal components for our data matrix
  - $S\Sigma$ (N x p)
    - diagonal matrix consisting of r non-zero values in descending order
    - square root of the eigenvalues of XX^T (or $X^T X$)
      - r is the rank of the symmetric matrices
    - called the singular values

SVD: Mathematical Background



Reconstructed matrix $X_k = U_k.\boldsymbol{\Sigma}_k.V_k^{\mathrm{T}}$
the closest *rank-k* matrix to the original matrix X

- PCA: :$U_k.\boldsymbol{\Sigma}_k V_k^{\mathrm{T}}$

  Principal direction vectors: columns of V: V_k

  Principal components: $U_k\boldsymbol{\Sigma}_k$

  First row of $U_k\mathbf{S}_k$ are the principal component "weights" for X_1

Outline

- Dimensionality Reduction: Linear Approaches
  - Review: linear transformation of Gaussians
  - Review: projection and least squares
- PCA
  - Last time, derive probabilistic "maximal" variance basis
  - Maximize variance of projection of data
  - Minimal approximation error from projection
- SVD
  - Definition & illustration
  - Relate to PCA
- Classification from PCA
- LDA

PCA for Classification

- PCA finds directions of maximal variation
  - Showed useful for *representation*

- Classification?
  - Let's have labelled data
  - How do we use PCA for classification?

- Take PCA of both data records together?
  - Separately?
  - What makes sense?

Problem Set Up

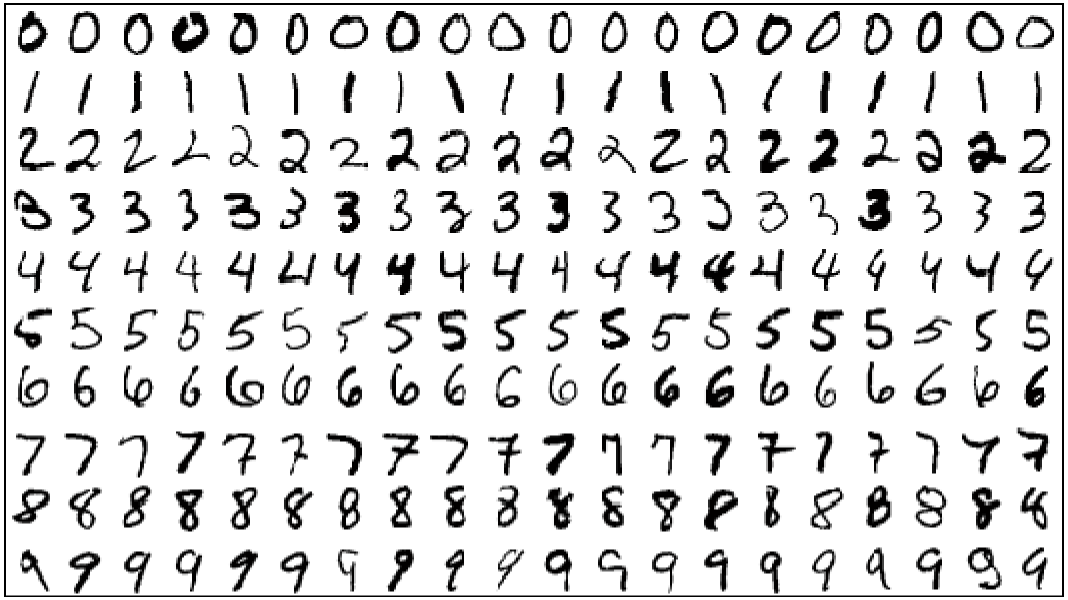
- X_1, X_2: Data matrices from two classes
- Sample mean and covariance in each data set
  - μ_ℓ, K_ℓ
- May not be the directions that are useful in classification
- The PC directions may not discriminate btw the classes
- What are the best directions?

Problem Set Up

- X_1, X_2: Data matrices from two classes

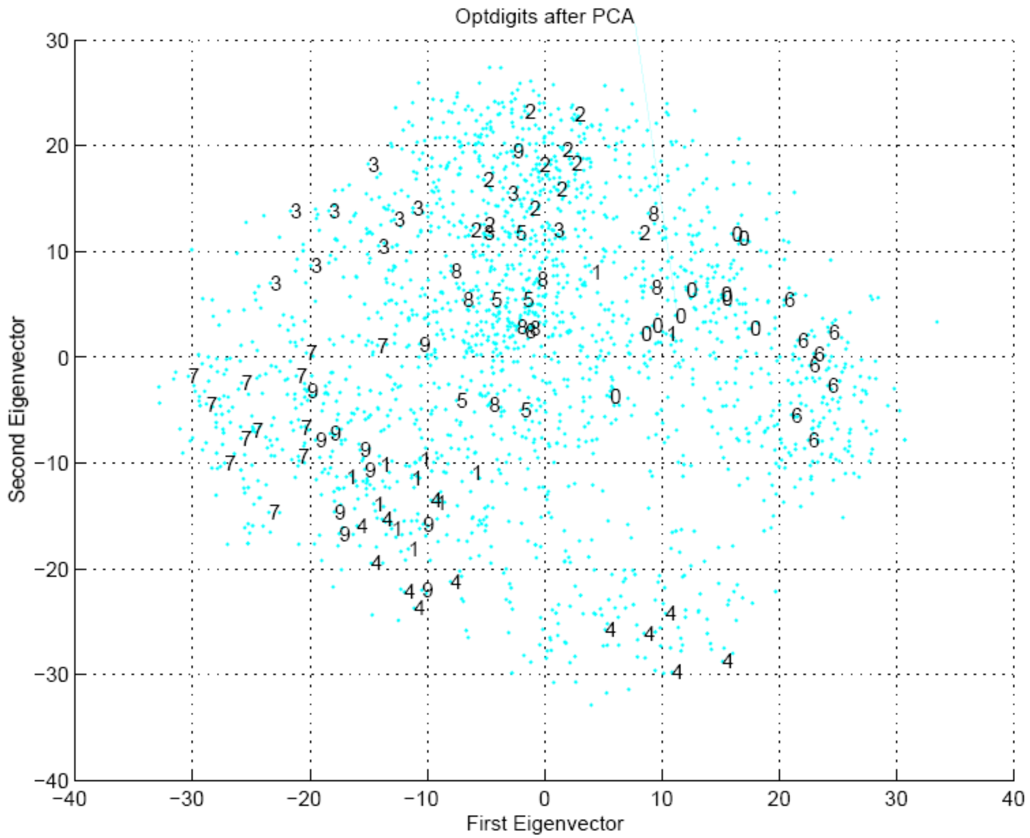
- Sample mean and covariance in each data set
  - μ_ℓ, K_ℓ
- Given vector w define:
  - Mean in each class: $m_\ell = w^T \mu_\ell$
  - Sample variance in each class: $s_\ell^2 = w^T K_\ell w$
  - Number of samples in each class: N_ℓ
- Project via PCA and try to classify

Example: optical handwritten digits



- Data set:images of handwritten digits

Visualizing PCs



- Take coefficients along two PCs
- Classification: some benefits
- But problems already

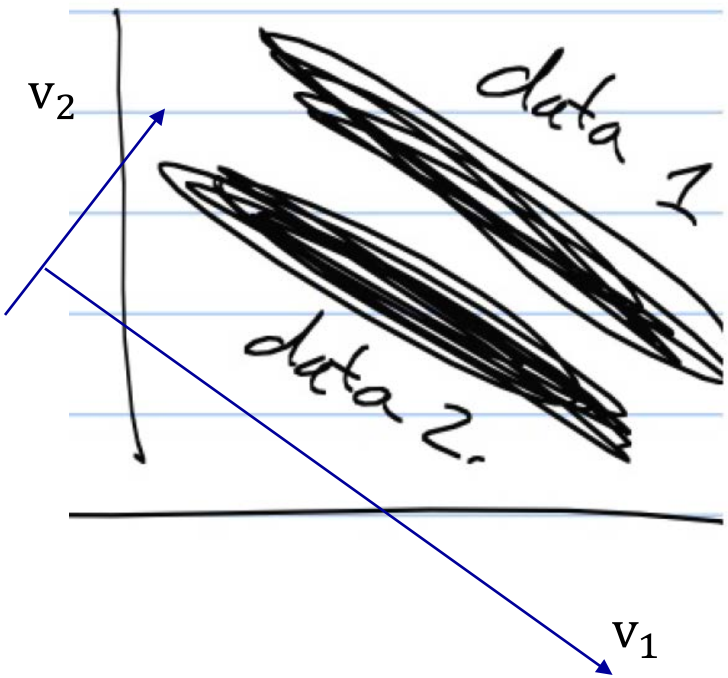
Problems with PCA for Classification

- May not be the directions that are useful in classification
- The PC directions may not discriminate btw the classes
- Can we find better directions?

PCA: Clustering Example

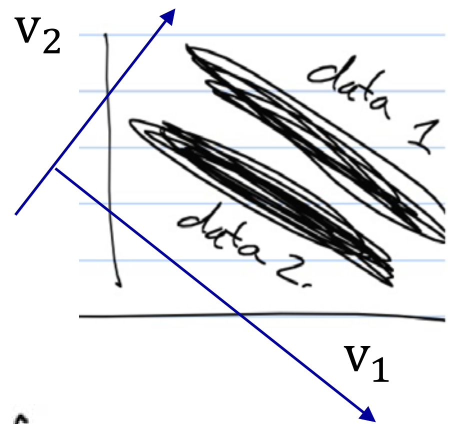
- **PCA will identify:**
- v_1: **dir of max variance**

- v_2: **dir of min variance**

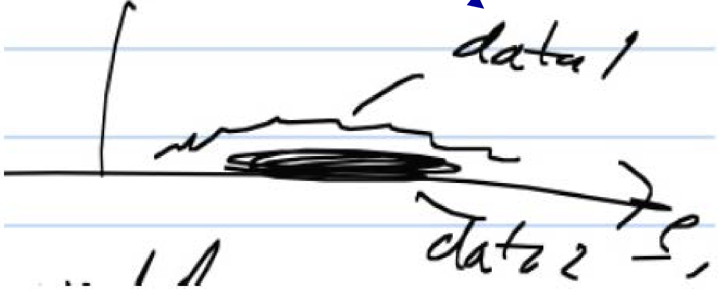


Problems with PCA Illustrated

- **Principal components**

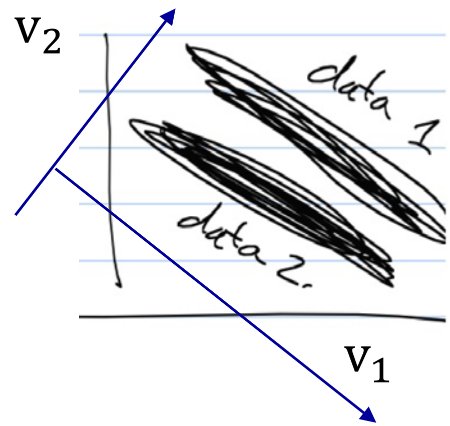


- **Projection onto v_1**
  - **Best PC vector, but**
  - **Very bad for separating classes**

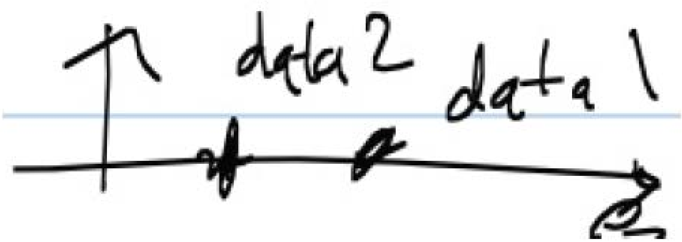


Problems with PCA clustering

- **Principal components**

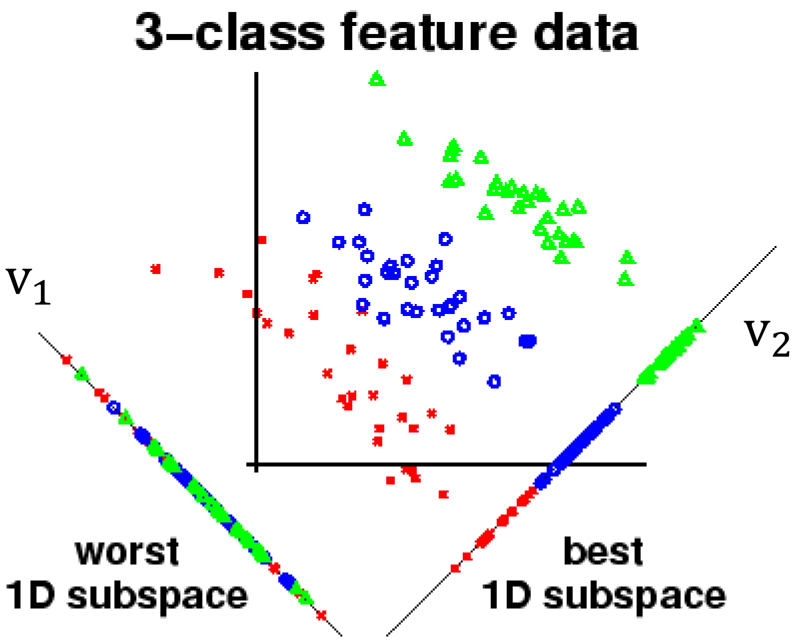


- **Projection onto v_2**
  - **Worst PC vector, but**
  - **very good for separating classes**



Problems with PC Illustrated: multi class

- PCA will identify:
- v_1: dir of max variance
  - Very bad for separating classes
- v_2: dir of min variance
  - Very good for separating classes

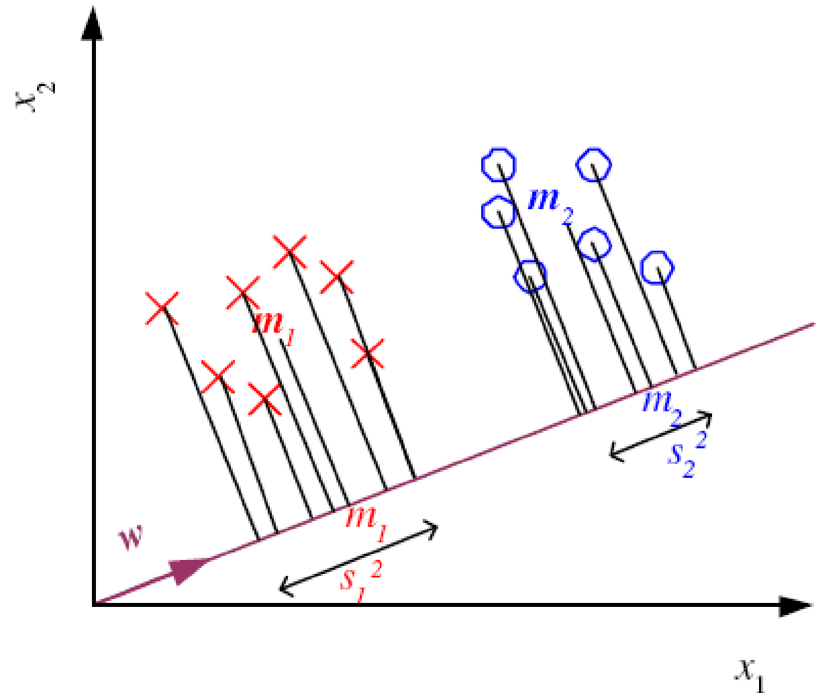


Outline

- Dimensionality Reduction: Linear Approaches
  - Review: linear transformation of Gaussians
  - Review: projection and least squares
- PCA
  - Last time, derive probabilistic "maximal" variance basis
  - Maximize variance of projection of data
  - Minimal approximation error from projection
- SVD
  - Definition & illustration
  - Relate to PCA
- Classification from PCA
- LDA

Linear Discriminant Analysis

- Lower dimension--
  preserves class separation.
  (hence discriminant)
- Project data on vector w aga
  (hence linear!)



- Find ***w*** that maximizes

$$J(\mathbf{w}) = \frac{(m_1 - m_2)^2}{s_1^2 + s_2^2}$$

It turns out that that w is the normal vector to the plane that best separates the classes

Problem Set Up

- X_1, X_2: Data matrices from two classes
- Sample mean and covariance in each data set
  - μ_ℓ, S_ℓ
- Given vector w define:
  - Mean in each class: $m_\ell = w^T \mu_\ell$
  - Sample variance in each class: $s_\ell^2 = w^T S_\ell w$
  - Number of samples in each class: N_ℓ
- Fisher Criteria: Find direction w to maximize:

$$J = \frac{N|m_1 - m_2|^2}{N_1 s_1^2 + N_2 s_2^2}$$

  - Average squared difference normalized by the variance

Solution to the Fisher Criteria (K=2 classes)

- Matrix form of Fisher criteria:

$$J = \frac{N|m_1 - m_2|^2}{N_1 s_1^2 + N_2 s_2^2} = \frac{N\left|w^T(\mu_1 - \mu_2)\right|^2}{N_1 w^T S_1 w + N_2 w^T S_2 w}$$

$$= \frac{w^T S_B w}{w^T S_W w}$$

  - $S_B = (\mu_1 - \mu_2)\,(\mu_1 - \mu_2)^T$, between class scatter, before projection
  - $S_W = \frac{N_1}{N} S_1 + \frac{N_2}{N} S_2$ weighted sum in-class scatter, before projection

Solution to the Fisher Criteria (K=2 classes)

- Optimization of Fisher
- Take derivative and set to zero:
  - $(w^T S_B w) S_W w = (w^T S_W w) S_B w$
  - $(w^T S_B w) S_W w = \left(w^T S_W w\right)(\mu_1 - \mu_2)^T w (\mu_1 - \mu_2)$
  - $S_W w = c(\mu_1 - \mu_2)$
- LDA solution: $w = c S_w^{-1}(\mu_1 - \mu_2)$

K>2 Classes

- With multiple classes, let
  - Overall sample mean: $\mu = \frac{1}{N}\sum_{i=1}^{N} x_i$
  - Sample mean and covariance in each class: μ_ℓ, S_ℓ
- Define:
  - Cross-class variances: $S_B = \sum_{\ell=1}^{K} N_\ell(\mu - \mu_\ell)\,(\mu - \mu_\ell)^T$
  - In-class scatter: $S_W = \sum_{\ell=1}^{K} N_\ell S_\ell$
- LDA components are $K - 1$ eigenvectors of $S_W^{-1} S_B$
- Or Find W that maximizes

$$J(\mathbf{W}) = \frac{\left|\mathbf{W}^T \mathbf{S}_B \mathbf{W}\right|}{\left|\mathbf{W}^T \mathbf{S}_W \mathbf{W}\right|}$$

The largest eigenvectors of $S_W^{-1}S_B$
Maximum rank of *K*-1

Fisher's Linear Discriminant: K>2 Classes

- With multiple classes, let
  - Overall sample mean: $\mu = \frac{1}{N}\sum_{i=1}^{N} x_i$
  - Sample mean and covariance in each class: μ_ℓ, S_ℓ
- Define:
  - Cross-class variances: $S_B = \sum_{\ell=1}^{K} N_\ell(\mu - \mu_\ell)\,(\mu - \mu_\ell)^T$
  - In-class scatter: $S_W = \sum_{\ell=1}^{K} N_\ell S_\ell$
- LDA components are $K - 1$ eigenvectors of $S_W^{-1} S_B$

Fisher's Linear Discriminant: K>2 Classes

- Find ***w*** that max

$$J(\mathbf{W}) = \frac{\left|\mathbf{W}^T \mathbf{S}_B \mathbf{W}\right|}{\left|\mathbf{W}^T \mathbf{S}_W \mathbf{W}\right|}$$

- LDA soln: $K-1$ eigenvectors of $S_W^{-1} S_B$

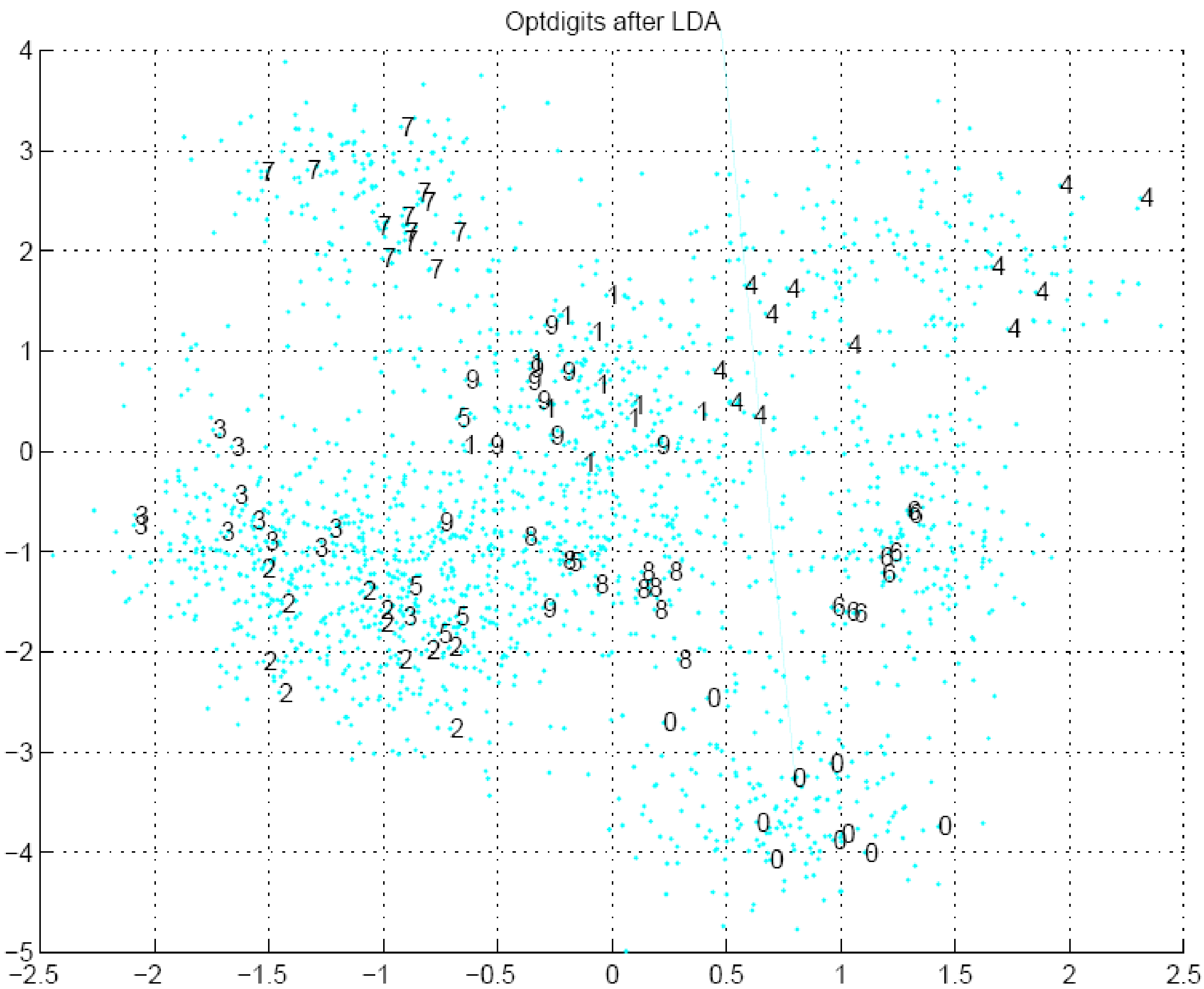
The largest eigenvectors of $S_W^{-1}S_B$
Maximum rank of *K*-1

- Parametric soln:

$$\mathbf{w} = \Sigma^{-1}(\mu_1 - \mu_2)$$

when $p(\mathbf{x} \mid C_i) \sim \mathcal{N}(\mu_i, \Sigma)$

OptDigits Example Revisited: After LDA



- Consider representation with 2 LDA components
- Visually you see a better separation between classes than with PC