Curse of Dimensionality

Dimensionality Reduction

An idea:

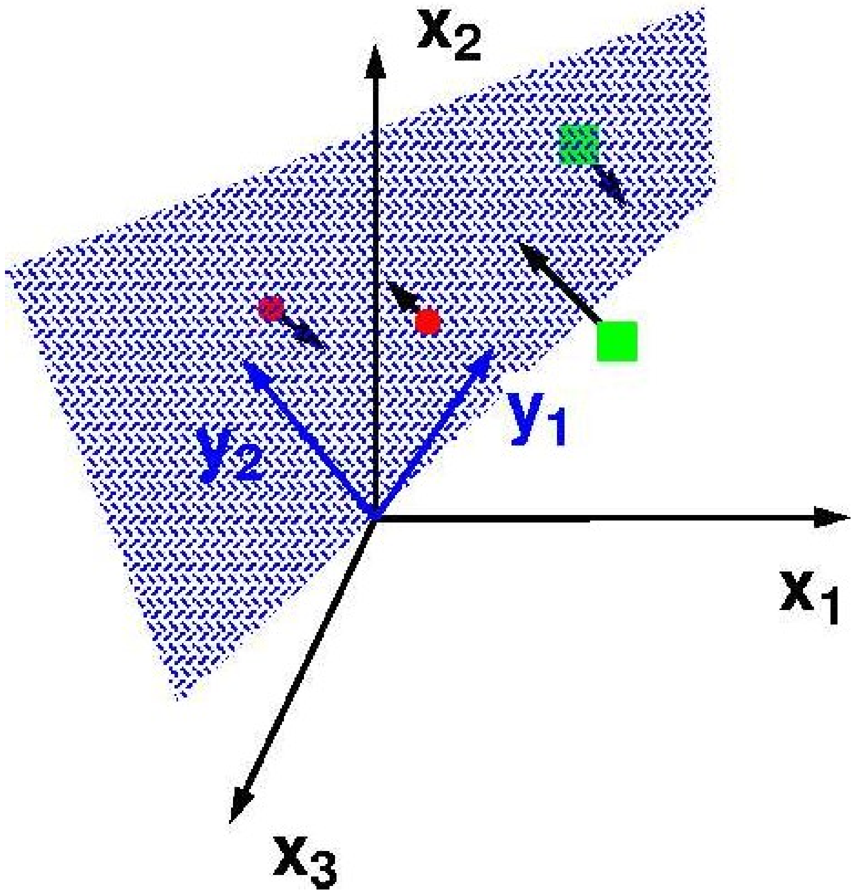
Represent the set of images as a linear subspace

What is a linear subspace?

Let V be a vector space and let W be a subset of V. Then W is a subspace if and only if:

1. The null vector $\mathbf{0}$ is in W
2. If $\mathbf{u}$ and $\mathbf{v}$ are elements of W, then any linear combination of $\mathbf{u}$ and $\mathbf{v}$ is an element of W; $a\mathbf{u} + b\mathbf{v} \in W$
3. If $\mathbf{u}$ is an element of W and c is a scalar, then the scalar product $c\mathbf{u} \in W$

- A k-dimensional subspace is spanned by k linearly independent vectors. It is spanned by a k-dimensional orthogonal basis



Example: A 2-D linear subspace of $\mathbf{R}^3$ spanned by y_1 and y_2

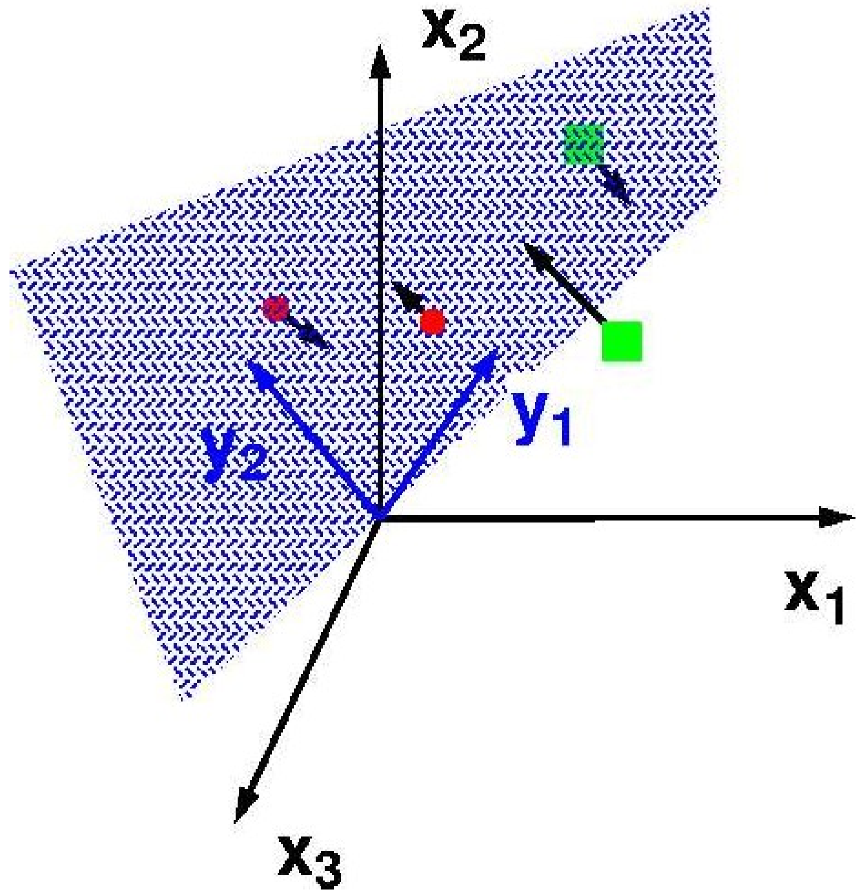
Linear Subspaces & Linear Projection

- A *d*-pixel image $\boldsymbol{x} \in \mathbf{R}^d$ can be projected to a low-dimensional feature space $\boldsymbol{y} \in \mathbf{R}^k$ by

$$\boldsymbol{y} = W\boldsymbol{x}$$

where W is an k by d matrix

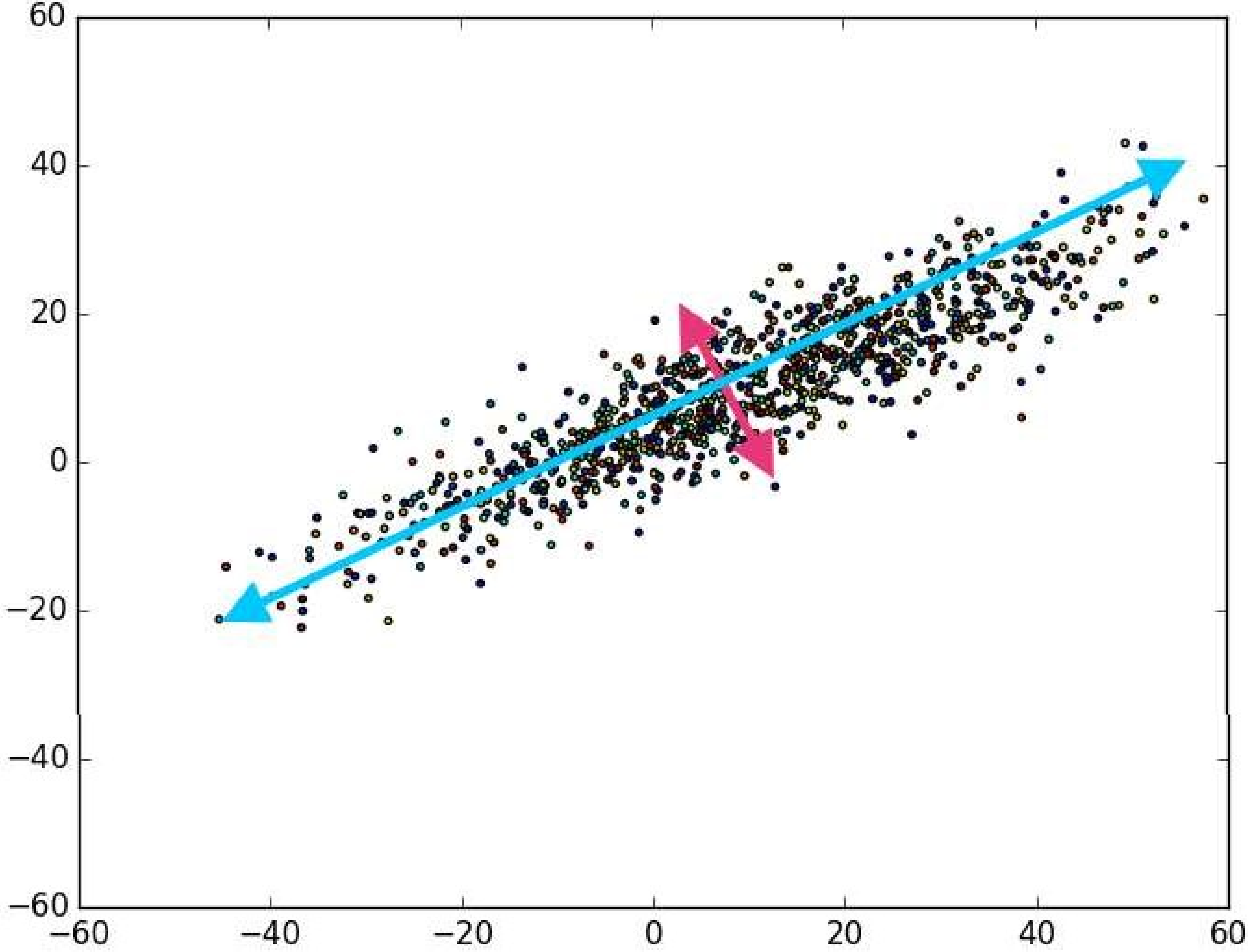
- Each training image is projected to the subspace
- Recognition is performed in $\mathbf{R}^k$ using, for example, nearest neighbor
- How do we choose a good W?



Example: A 2-D linear subspace of $\mathbf{R}^3$ spanned by y_1 and y_2

How do we choose a good W?

- Drop dimensions (feature selection)
- Random projections
- Principal Component Analysis
- Linear Discriminant Analysis
- Independent Component Analysis
- Or non-linear dimensionality reduction


60
40
20
0
−20
−40
−60
−60
−40
−20
0
20
40
60

Principal component analysis (PCA) of covariance matrix

Assume we have a set of n feature vectors $\boldsymbol{x}_i$ $(i = 1, \ldots, n)$ in $\mathbb{R}^d$. Write

Mean $\boldsymbol{\mu}_{\mathbf{x}} = \frac{1}{n} \sum_i \boldsymbol{x}_i$

Covariance $\Sigma_{\mathbf{x}} = \frac{1}{n-1} \sum_i (\boldsymbol{x}_i - \boldsymbol{\mu})(\boldsymbol{x}_i - \boldsymbol{\mu})^T$

Eigen decomposition of covariance matrix

$$\Sigma_{\mathbf{x}} = \mathtt{V} \Lambda \mathtt{V}^\top$$

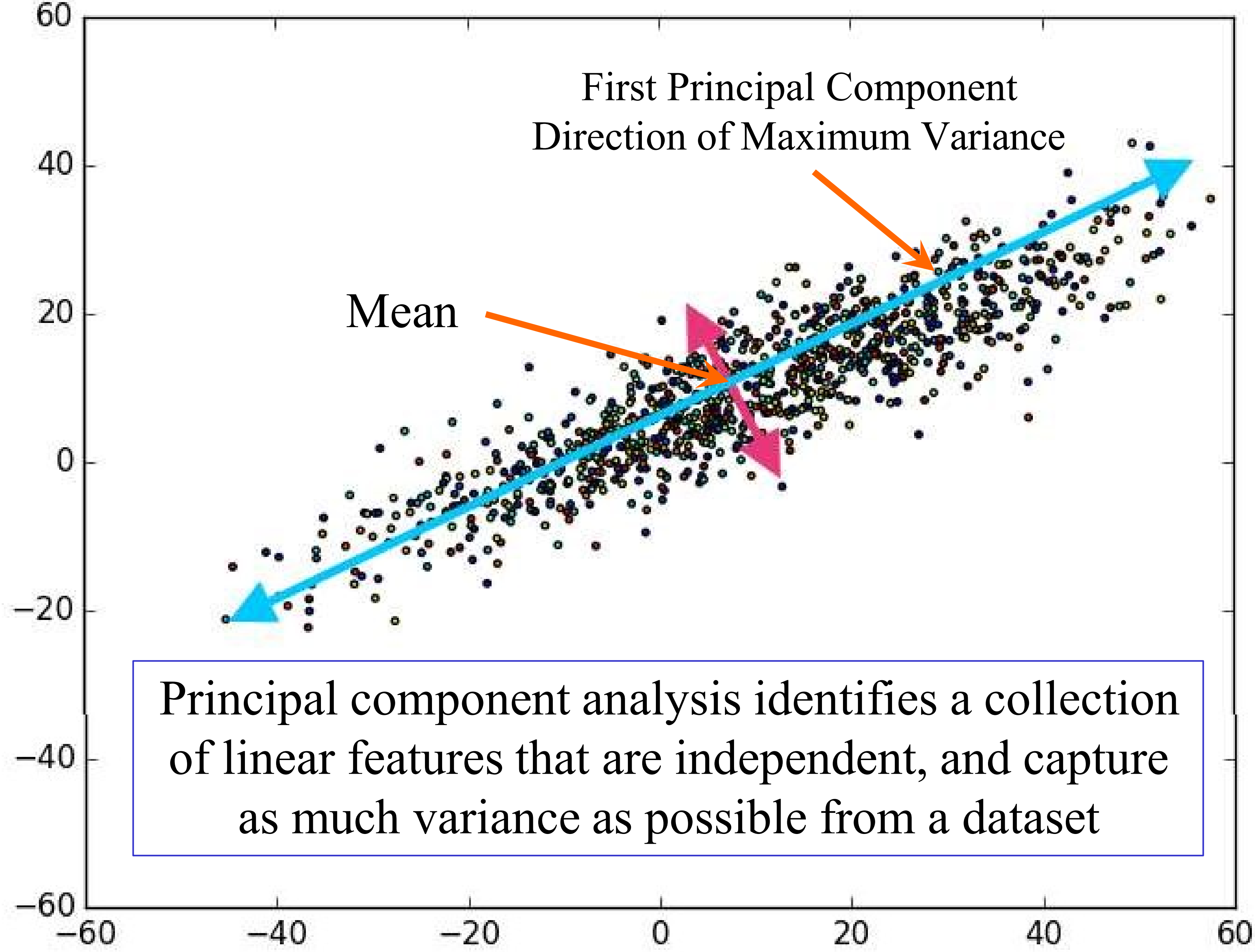
where

$\Sigma_{\mathbf{x}}$ is a positive semidefinite $n \times n$ matrix

$\mathtt{V}$ is an $n \times n$ orthogonal matrix

$\Lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_n)$, where $\lambda_i \geq \lambda_{i+1} \geq 0$

Columns of $\mathtt{V}$ are eigenvectors (also called principal component coefficients) corresponding to eigenvalues (also called principal component variances) $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_n)^\top$.


First Principal Component
Direction of Maximum Variance
Mean
Principal component analysis identifies a collection of linear features that are independent, and capture as much variance as possible from a dataset
60
40
20
0
−20
−40
−60
−60
−40
−20
0
20
40
60

Relationship between singular value decomposition (SVD) and eigen decomposition

$$
\begin{aligned}
\mathtt{A} &= \mathtt{U}\Sigma\mathtt{V}^\top \\
\mathtt{A}^\top\mathtt{A} &= (\mathtt{U}\Sigma\mathtt{V}^\top)^\top\mathtt{U}\Sigma\mathtt{V}^\top \\
\mathtt{A}^\top\mathtt{A} &= \mathtt{V}\Sigma^\top\mathtt{U}^\top\mathtt{U}\Sigma\mathtt{V}^\top \\
\mathtt{A}^\top\mathtt{A} &= \mathtt{V}\Sigma^\top\Sigma\mathtt{V}^\top \\
\mathtt{A}^\top\mathtt{A} &= \mathtt{V}\Lambda\mathtt{V}^\top, \text{ where } \Lambda = \Sigma^\top\Sigma
\end{aligned}
$$

$$
\begin{aligned}
\mathtt{A} &= \mathtt{U}\Sigma\mathtt{V}^\top \\
\mathtt{A}\mathtt{A}^\top &= \mathtt{U}\Sigma\mathtt{V}^\top(\mathtt{U}\Sigma\mathtt{V}^\top)^\top \\
\mathtt{A}\mathtt{A}^\top &= \mathtt{U}\Sigma\mathtt{V}^\top\mathtt{V}\Sigma^\top\mathtt{U}^\top \\
\mathtt{A}\mathtt{A}^\top &= \mathtt{U}\Sigma\Sigma^\top\mathtt{U}^\top \\
\mathtt{A}\mathtt{A}^\top &= \mathtt{U}\Lambda\mathtt{U}^\top, \text{ where } \Lambda = \Sigma\Sigma^\top
\end{aligned}
$$

where

$$
\begin{aligned}
&\mathtt{U} \text{ and } \mathtt{V} \text{ are orthogonal matrices} \\
&\Sigma = \operatorname{diag}(\sigma_1, \ldots, \sigma_n), \text{ where } \sigma_i \geq \sigma_{i+1} \geq 0 \\
&\Lambda = \operatorname{diag}(\lambda_1, \ldots, \lambda_n), \text{ where } \lambda_i \geq \lambda_{i+1} \geq 0 \\
&\text{variances } \lambda_i = \sigma_i^2 \, \forall \, i
\end{aligned}
$$

Data matrix

Data matrix

$$\mathtt{X} \in \mathbb{R}^{m \times n} \quad m \text{ observations, } n \text{ variables}$$

$$\mathtt{X} = \begin{bmatrix} \mathbf{x}^{1\top} \\ \mathbf{x}^{2\top} \\ \vdots \\ \mathbf{x}^{m\top} \end{bmatrix}$$

Data matrix, mean-deviation form

$$\hat{\mathtt{X}} = \begin{bmatrix} \mathbf{x}^{1\top} - \boldsymbol{\mu}_x^\top \\ \mathbf{x}^{2\top} - \boldsymbol{\mu}_x^\top \\ \vdots \\ \mathbf{x}^{m\top} - \boldsymbol{\mu}_x^\top \end{bmatrix} \qquad \boxed{\Sigma_{\mathbf{x}} = \frac{1}{m-1}\hat{\mathtt{X}}^\top \hat{\mathtt{X}}}$$

Singular value decomposition of (mean-deviation form of) data matrix

$$\hat{\mathtt{X}} = \mathtt{UDV}^\top$$

Note: economy SVD can be used

where

$$\mathtt{D} = \mathrm{diag}(\sigma_1, \ldots, \sigma_{\min(m,n)}), \text{ where } \sigma_i \geq \sigma_{i+1} \geq 0$$

columns of $\mathtt{V}$ (rows of $\mathtt{V}^\top$) are principal component coefficients of $\hat{\mathtt{X}}$

Projection of $\hat{\mathtt{X}}$ to principal component axes

$$\hat{\mathtt{A}} = \hat{\mathtt{X}}\mathtt{V} \qquad \text{(forward) projection to principal component scores}$$

$$\hat{\mathtt{A}}\mathtt{V}^\top = \hat{\mathtt{X}} \qquad \text{back projection}$$

Dimensionality reduction
Columns of $\mathtt{V}$ corresponding to smallest singular values can be removed

$$\hat{\mathtt{A}}' = \hat{\mathtt{X}}\mathtt{V}' \qquad \text{(forward) projection to principal component scores}$$

$$\hat{\mathtt{A}}'\mathtt{V}'^\top = \hat{\mathtt{X}}' \qquad \text{back projection (with loss of data)}$$

SVD Properties

- r = rank($\boldsymbol{A}$) = number of non-zero singular values
- $\boldsymbol{U}$, $\boldsymbol{V}$ give an orthonormal bases for the subspaces of $\boldsymbol{A}$:
  - 1st r columns of $\boldsymbol{U}$: Column space of $\boldsymbol{A}$
  - Last m - r columns of $\boldsymbol{U}$: Left nullspace of $\boldsymbol{A}$
  - 1st r columns of $\boldsymbol{V}$: Row space of $\boldsymbol{A}$
  - 1st n - r columns of $\boldsymbol{V}$: (Right) nullspace of $\boldsymbol{A}$
- *For some d where d $\leq$ r, the first d column of U provide the best d-dimensional basis for columns of A in least squares sense.*

PCA for recognition

Modeling

1. Given a collection of n training images x_i, represent each one as a d-dimensional column vector
2. Compute the mean image and covariance matrix
3. Compute k Eigenvectors of the covariance matrix corresponding to the k largest Eigenvalues and form matrix $W^T=[u_1, u_2, \ldots, u_k]$ (Or perform using SVD)
   - Note that the Eigenvectors are images
4. Project the training images to the k-dimensional Eigenspace. $y_i=Wx_i$

Recognition

1. Given a test image x, project the vectorized image to the Eigenspace by $y=Wx$
2. Perform classification of y to the projected training images

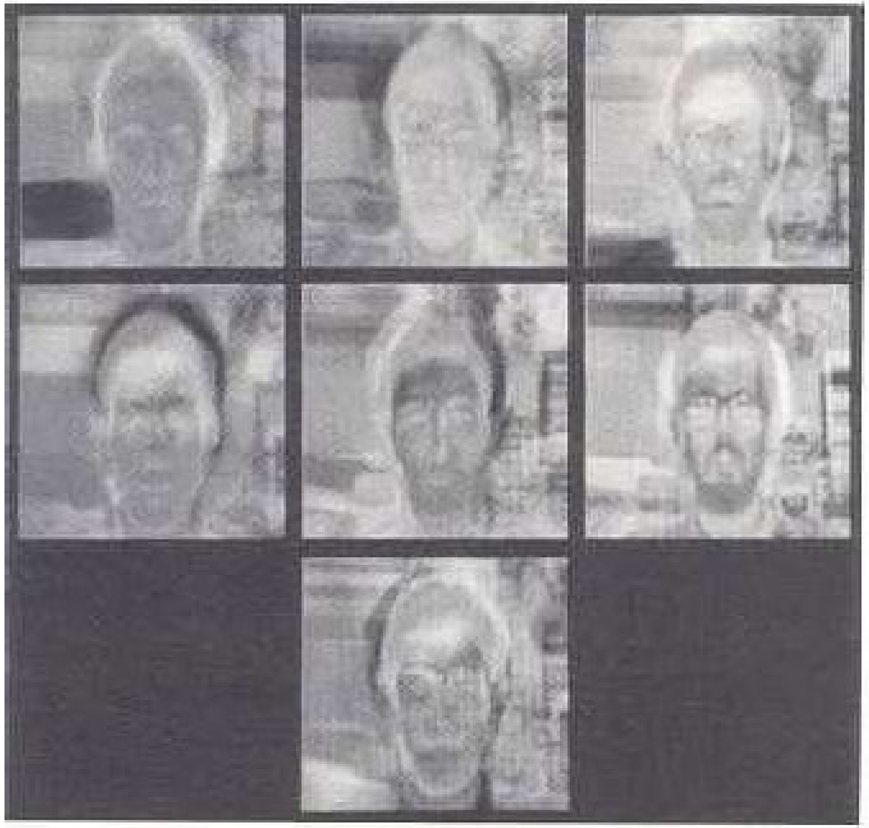
Example: Eigenfaces

Training images

[Turk, Pentland 91]

Eigenfaces

Mean Image



Basis Images

Accuracy of PCA + K-NN

KNN + 3,072 Features	33.86
KNN + 200 PCA Comp.	36.54
KNN + 75 PCA Comp.	39.77
KNN + 50 PCA Comp.	40.12
KNN + 40 PCA Comp.	40.93
KNN + 30 PCA Comp.	**41.78**
KNN + 25 PCA Comp.	41.57
KNN + 15 PCA Comp.	38.75
KNN + 10 PCA Comp.	34.93

Difficulties with PCA

- Projection may suppress important detail
  - smallest variance directions may not be unimportant
- Method does not take discriminative task into account
  - typically, we wish to compute features that allow good discrimination
  - not the same as largest variance or minimizing reconstruction error