Dimensionality Reduction
PCA
continued…

Machine Learning – CSE446
Carlos Guestrin
University of Washington

May 22, 2013

Dimensionality reduction

- Input data may have thousands or millions of dimensions!
  - e.g., text data has

x with 10 000 — 10 000 000 dims

- **Dimensionality reduction**: represent data with fewer dimensions
  - easier learning – fewer parameters
  - visualization – hard to visualize more than 3D or 4D
  - discover "intrinsic dimensionality" of data
    - high dimensional data that is truly lower dimensional

Lower dimensional projections

- Rather than picking a subset of the features, we can new features that are combinations of existing features

$z_7 = 2.5\,x_1 + 2.9\,x_2 - 3.7\,x_3 \ldots$

learn

X 100 → Z 100

1000

model

loss/accuracy

linear

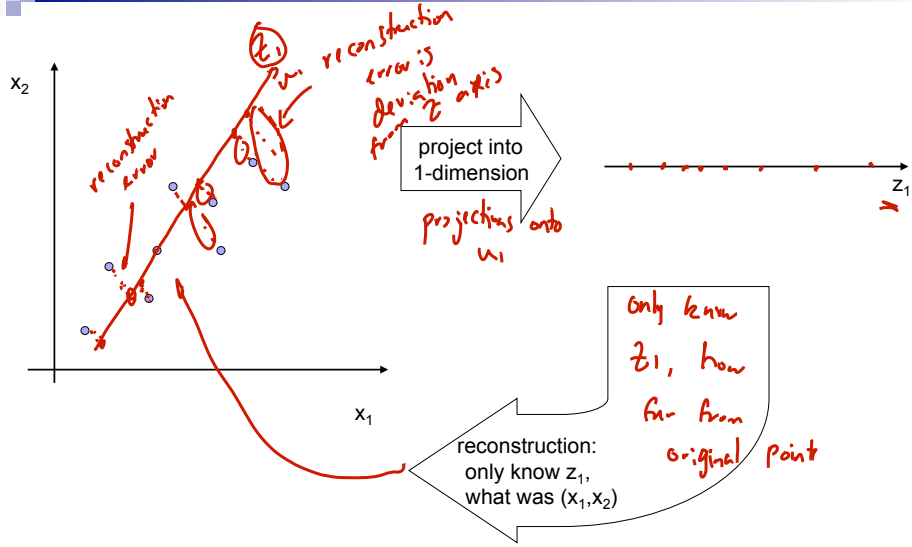
$z = Ax$

reconstruction error

if I project back get back

- Let's see this in the unsupervised setting
  - just **X**, but no Y

Linear projection and reconstruction



Principal component analysis – basic idea

- Project d-dimensional data into k-dimensional space while preserving information:
  - e.g., project space of 10000 words into 3-dimensions
  - e.g., project 3-d into 2-d
- Choose projection with minimum reconstruction error

Linear projections, a review

- Project a point into a (lower dimensional) space:
  - **point**: $\mathbf{x} = (x_1,\ldots,x_d)$ 10,000,00
  - **select a basis** – set of basis vectors – $(\mathbf{u}_1,\ldots,\mathbf{u}_k)$ 100
    - we consider orthonormal basis:
      - $\mathbf{u}_i \bullet \mathbf{u}_i = 1$, and $\mathbf{u}_i \bullet \mathbf{u}_j = 0$ for $i \neq j$
  - **select a center** – $\bar{\mathbf{x}}$, defines offset of space ← mean of X
  - **best coordinates** in lower dimensional space defined by dot-products: $(z_1,\ldots,z_k)$, $z_i = (\mathbf{x}-\bar{\mathbf{x}}) \bullet \mathbf{u}_i$
    - minimum squared error

learn these from data

$z_i = (x-\bar{x}) \cdot u_i$ } $z_i = \underset{z}{\operatorname{argmin}} \left[(x-\bar{x}) - z\, u_i\right]^2$

PCA finds projection that minimizes reconstruction error

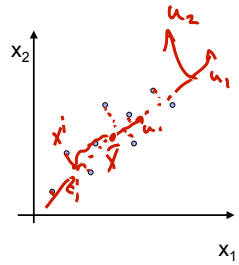
- Given ~~m~~ N data points: $\mathbf{x}^i = (x_1^i,\ldots,x_d^i)$, i=1…N
- Will represent each point as a projection:
  - $\hat{\mathbf{x}}^i = \bar{\mathbf{x}} + \sum_{j=1}^{k} z_j^i \mathbf{u}_j$ where: $\bar{\mathbf{x}} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{x}^i$ and $z_j^i = (\mathbf{x}^i - \bar{\mathbf{x}}) \cdot \mathbf{u}_j$

approx for x^i — own coeff — same basis — avg — projection: coeff from previous slide

- PCA:
  - Given k<d, find $(\mathbf{u}_1,\ldots,\mathbf{u}_k)$ minimizing reconstruction error:

$$error_k = \sum_{i=1}^{N} (\mathbf{x}^i - \hat{\mathbf{x}}^i)^2$$

sum over data points — truth — projection into k-dim space



Understanding the reconstruction error

- Note that $\mathbf{x}^i$ (in d-dims) can be represented exactly by d-dimensional projection:

$$\mathbf{x}^i = \bar{\mathbf{x}} + \sum_{j=1}^{d} z_j^i \mathbf{u}_j$$

$(u_1,\ldots,u_d)$ orthonormal basis

$$\hat{\mathbf{x}}^i = \bar{\mathbf{x}} + \sum_{j=1}^{k} z_j^i \mathbf{u}_j$$ want to learn $u_1 \ldots u_k$

$$z_j^i = (\mathbf{x}^i - \bar{\mathbf{x}}) \cdot \mathbf{u}_j$$

- Given k<d, find $(\mathbf{u}_1,\ldots,\mathbf{u}_k)$ minimizing reconstruction error:

choose u_j

$$error_k = \sum_{i=1}^{N} (\mathbf{x}^i - \hat{\mathbf{x}}^i)^2$$

want to minimize

- Rewriting error:

$$error_k = \sum_{i=1}^{N} (x^i - \hat{x}^i)^2 = \sum_{i=1}^{N}\left[\bar{x} + \sum_{j=1}^{d} z_j^i u_j - \left(\bar{x} + \sum_{j=1}^{k} z_j^i u_j\right)\right]^2 = \sum_{i=1}^{N}\left[\sum_{j=k+1}^{d} z_j^i u_j\right]^2$$

$$= \sum_{i=1}^{N}\left[\sum_{j=k+1}^{d} z_j^i \cancel{u_j}^{1} \cancel{u_j} z_j^i + \sum_{j=k+1}^{d}\sum_{\substack{l=k+1 \\ l\neq j}}^{d} z_j^i u_j \cancel{u_l}^{0} z_l^i\right]$$

$$= \sum_{i=1}^{N}\sum_{j=k+1}^{d} (z_j^i)^2$$ ← minimizing projection into dims that are ignored

Reconstruction error and covariance matrix

choose $u_1 \dots u_k$ to minimize $error_k$

$a \cdot b = a^T b$

$(a)^2 = a^T a$

$(a\ b)^2 = (a^T b)^2 = b^T a a^T b$

$a = (x^i - \bar{x})$ $b = u_j$

$$error_k = \sum_{i=1}^{N} \sum_{j=k+1}^{d} [\mathbf{u}_j \cdot (\mathbf{x}^i - \bar{\mathbf{x}})]^2$$

$$= \sum_{i=1}^{N} \sum_{j=k+1}^{d} u_j^T (x^i - \bar{x})(x^i - \bar{x})^T u_j$$

$$= \sum_{j=k+1}^{d} u_j^T \underbrace{\left[\sum_{i=1}^{N} (x^i - \bar{x})(x^i - \bar{x})^T \right]}_{N\Sigma} u_j$$

$$= N \sum_{j=k+1}^{d} u_j^T \Sigma u_j$$

← choose u_j to minimize this error

$$\Sigma = \frac{1}{N} \sum_{i=1}^{N} (\mathbf{x}^i - \bar{\mathbf{x}})(\mathbf{x}^i - \bar{\mathbf{x}})^T$$

$$\Sigma = \begin{array}{c} \\ r \\ \\ \end{array} \begin{array}{c} \begin{array}{ccc} & r & s \end{array} \\ \begin{pmatrix} \ddots & & \\ & \sigma_r^2 & \sigma_{rs} \\ & & \ddots \end{pmatrix} \end{array}$$

$$\sigma_{rs} \overset{MLE}{=} \frac{1}{N} \sum_{i=1}^{N} (x_r^i - \bar{x}_r)(x_s^i - \bar{x}_s)$$

in vector form

$$\Sigma \overset{MLE}{=} \frac{1}{N} \sum_{i=1}^{N} (x^i - \bar{x})(x^i - \bar{x})^T$$

Minimizing reconstruction error and eigen vectors

- Minimizing reconstruction error equivalent to picking (ordered) orthonormal basis ($\mathbf{u}_1,\dots,\mathbf{u}_d$) minimizing:

$$error_k = N \sum_{j=k+1}^{d} \mathbf{u}_j^T \Sigma \mathbf{u}_j$$

← sum of eigen values of cov. matrix Σ

- Eigen vector:

$\Sigma u = \lambda u$

eigen vector ← eigen value

$u^T \Sigma u = \lambda u^T u = \lambda$

- Minimizing reconstruction error equivalent to picking ($\mathbf{u}_{k+1},\dots,\mathbf{u}_d$) to be eigen vectors with smallest eigen values

$\min_{u_1 \dots u_k} error_k$ = throwing out $d-k$ eigen vectors with smallest eigen values

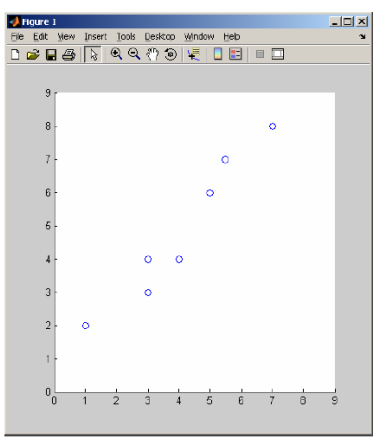
⇒ keep top k eigen vectors of Σ

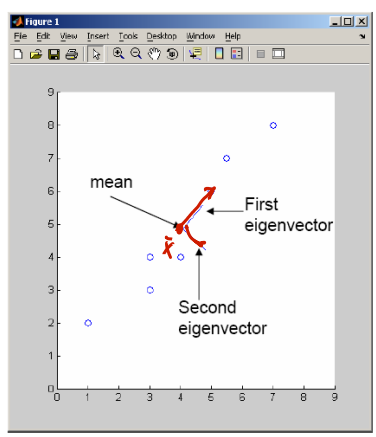
Basic PCA algoritm

- Start from m by n data matrix **X**
- **Recenter**: subtract mean from each row of **X**
  - $\mathbf{X_c} \leftarrow \mathbf{X} - \overline{\mathbf{X}}$
- **Compute covariance matrix**:
  - $\Sigma \leftarrow 1/N\ \mathbf{X_c}^T\ \mathbf{X_c}$
- Find **eigen vectors and values** of Σ
- **Principal components:** k eigen vectors with highest eigen values

PCA example

$$\hat{\mathbf{x}}^i = \bar{\mathbf{x}} + \sum_{j=1}^{k} z_j^i \mathbf{u}_j$$

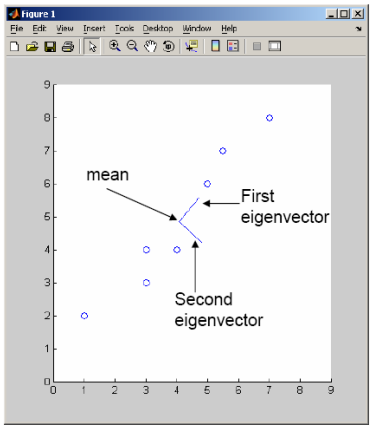


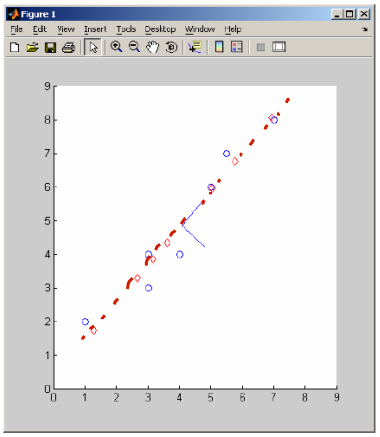


PCA example – reconstruction

$$\hat{\mathbf{x}}^i = \bar{\mathbf{x}} + \sum_{j=1}^{k} z_j^i \mathbf{u}_j$$

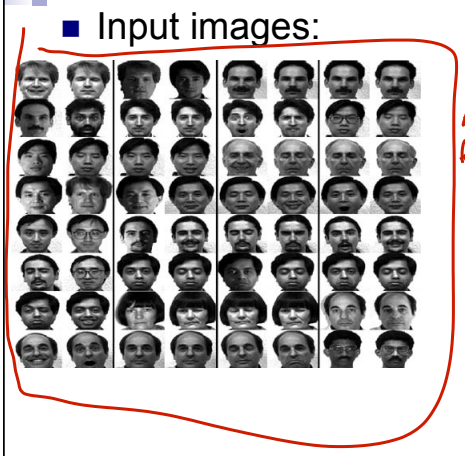
only used first principal component





Eigenfaces [Turk, Pentland '91]

- Input images:

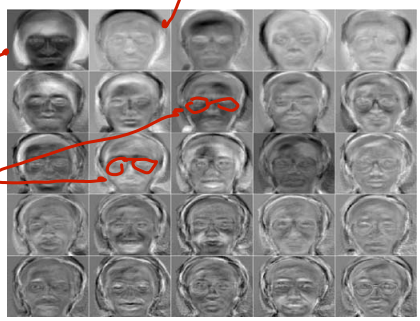


- Principal components:

long hair

Avg face

glasses



Eigenfaces reconstruction

- Each image corresponds to adding 8 principal components:

highly recognizable