CS37300: Dimensionality Reduction using PCA

Nov 13, 2023

Dimensionality reduction

- Identify and describe the “dimensions” that underlie the data
  – May be more fundamental than those directly measured but hidden to the user
- Reduce dimensionality of modeling problem
  – Benefit is simplification, it reduces the number of variables you have to deal with in modeling
- Can identify sets of variables with similar behavior
  – May indicate “unneeded” data derived from other values

Dimensionality reduction methods

- Principal component analysis (PCA)
  - Linear transformation, minimize unexplained variance
- Factor analysis
  - Linear combination of small number of latent variables
- Multidimensional scaling (MDS)
  - Project into low-dimensional subspace while preserving distance between points (can be non-linear)

Principal component analysis (PCA)

- High-level approach, given data matrix **D** with **p** dimensions:
  - Preprocess **D** so that the mean of each attribute is 0, call this matrix **X**
  - Compute pxp covariance matrix: $\Sigma = X^T X$
  - Compute eigenvectors/eigenvalues of covariance matrix:

$$\mathbf{A}\Sigma\mathbf{A}^{-1} = \Lambda$$
$$(\Sigma - \lambda\mathbf{I})\mathbf{a} = 0$$

$\mathbf{A}$: matrix of eigenvectors
Λ : diagonal matrix of eigenvalues
$\mathbf{a}$: 1st principal component, eigenvector assoc. with largest eigenvalue (λ)

  - Eigenvectors $\mathbf{A}$ are the **principal component** vectors, where each is a px1 column vector of projection weights $\mathbf{a}$

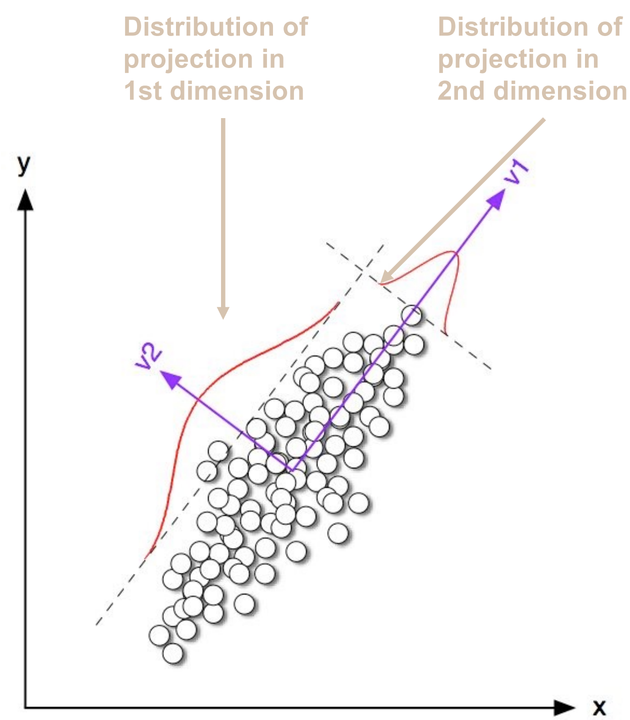
What is the model space for PCA?

$$\mathbf{A\Sigma A^{-1}} = \mathbf{\Lambda}$$

$\mathbf{A}$: matrix of eigenvectors

$\mathbf{\Sigma}$: diagonal matrix of eigenvalues

- **A** is a *p x p* matrix of principal components (if data is p-dimensional)
  - each column is a basis vector, each cell is a projection weight
- Model space: is defined by **A**
  - Method needs to choose the p^2 weights that populate **A**, i.e., set of p basis vectors
  - Constraints: Basis vectors must be **orthonormal**, i.e., each has a norm of 1 and any pair of basis vectors have dot-product of 0
  - E.g., any orthogonal set of v_1 and v_2



What is the score function for PCA?

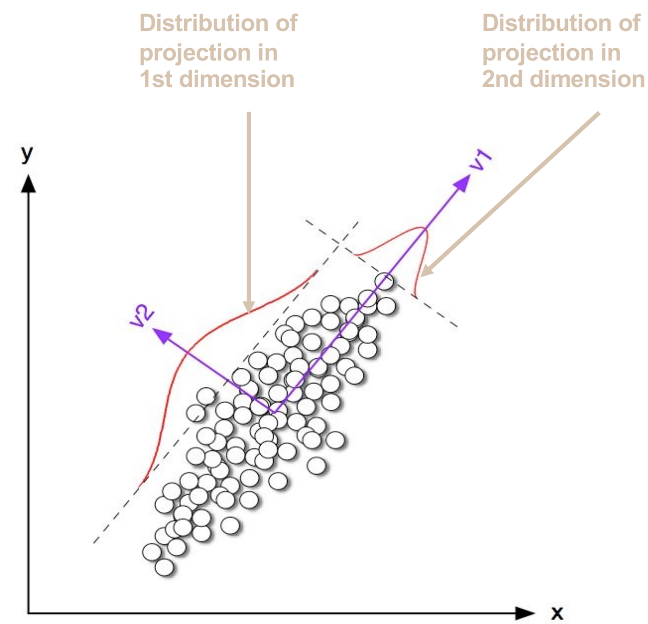
$$\mathbf{A\Sigma A^{-1}} = \mathbf{\Lambda}$$

$\mathbf{A}$: matrix of eigenvectors

$\mathbf{\Sigma}$: diagonal matrix of eigenvalues

- $\mathbf{\Lambda}$ is diagonal matrix of *p* eigenvalues
  - Each eigenvalue λ_i corresponds to the variance of dimension i
- sum of eigenvalues in Λ: $\sum_{j=1}^{p} \lambda_j$
- Sum of eigenvalues is equal to the sum of the variances of the original attributes:

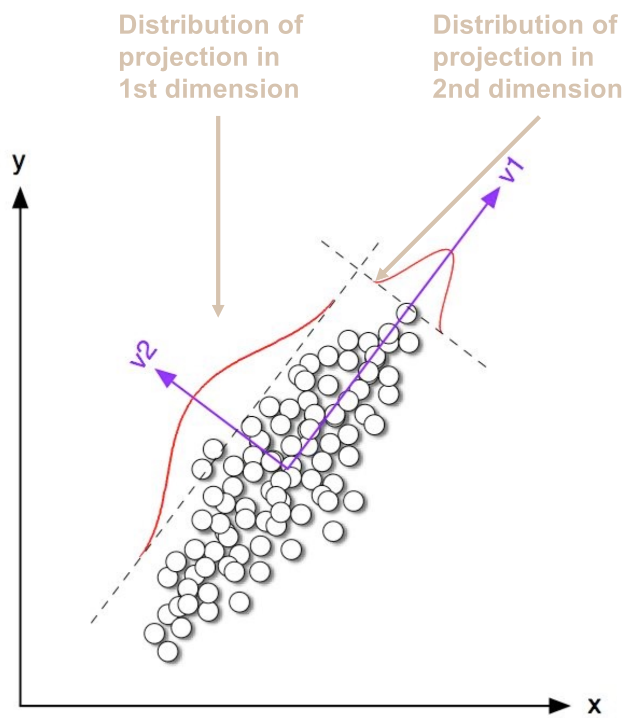
$$\sum_{j=1}^{p} \sigma^2 = \sum_{j=1}^{p} \lambda_j$$



What is the search method for PCA?

- **Goal:** find basis vectors that maximize variance
  - 1st basis (eg. v_1) ***maximizes*** variance of projected data
  - 2nd basis (eg. v_2) again ***maximizes*** variance of projected data, but has to be orthogonal to previous bases, …
  - New dimensions are orthogonal, thus transformed features have 0 covariance
- Search: Solving eigensystem corresponds to finding the orthonormal basis that **maximize** variance of projected data

$$\mathbf{A\Sigma A^{-1}} = \mathbf{\Lambda}$$



Applying PCA

- Choose number of target dimensions (i.e., select m < p)
  - Transform data vectors by projecting them onto the first *m* principal components, which correspond to top m eigenvectors)

$$\mathbf{x} = [x_1, x_2, \ldots, x_p] \quad \text{(original instance)}$$

$$\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_p] \quad \text{(principal components)}$$

$$x'_1 = \mathbf{a}_1 \mathbf{x} = \sum_{j=1}^{p} a_{1j} x_j$$

$$\ldots$$

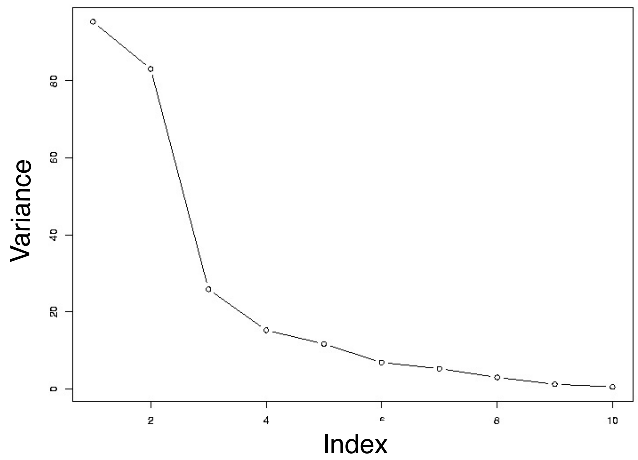
$$x'_m = \mathbf{a}_m \mathbf{x} = \sum_{j=1}^{p} a_{mj} x_j \quad for\ m < p$$

$$\mathbf{x}' = [x'_1, x'_2, \ldots, x'_m] \quad \text{(transformed instance)}$$

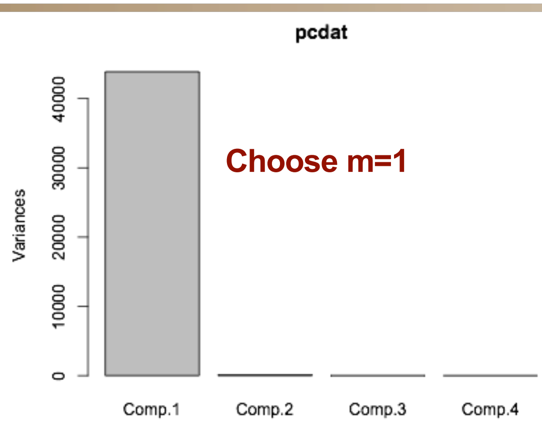
If *m=p* then data is transformed
If *m<p* then transformation is lossy and dimensionality is reduced

Applying PCA (cont')

- Goal: Find a new (smaller) set of dimensions that captures most of the variability of the data
- Can use **scree plot** to choose number of dimensions
  - Choose $m < p$ so projected data captures much of the variance of original data



PCA example on Iris data

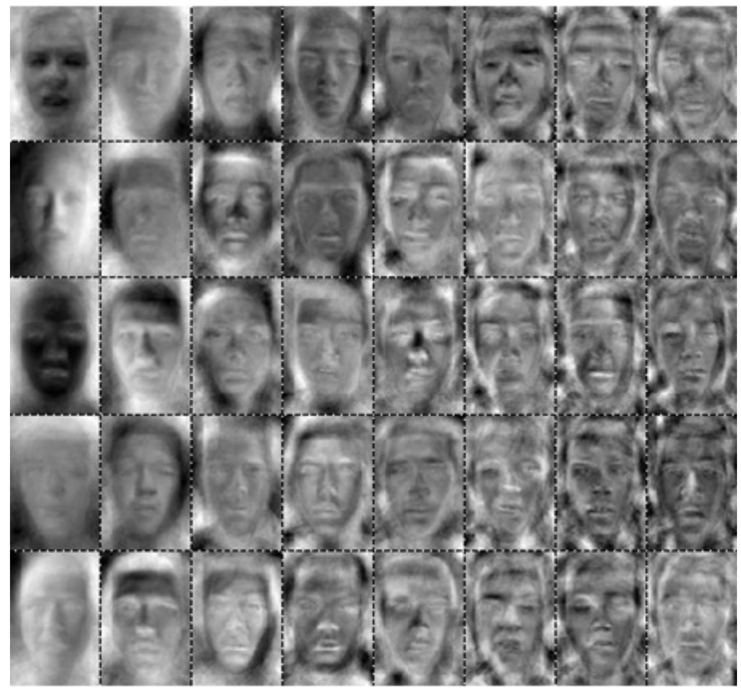



```
> x <- scale(as.matrix(d[,1:4]),scale=FALSE)
> sigma <- t(x)%*% x
> sigma
          V1       V2       V3        V4
V1 102.16833  -5.8510 189.7787  77.01867
V2  -5.85100  28.0126 -47.9352 -17.57920
V3 189.77867 -47.9352 463.8637 193.16173
V4  77.01867 -17.5792 193.1617  86.77973

> pcdat <- princomp(d[,1:4])
> summary(pcdat)
Importance of components:
                          Comp.1     Comp.2     Comp.3      Comp.4
Standard deviation     2.0485788 0.49053911 0.27928554 0.153379074
Proportion of Variance 0.9246162 0.05301557 0.01718514 0.005183085
Cumulative Proportion  0.9246162 0.97763178 0.99481691 1.000000000
> plot(pcdat)
> loadings(pcdat)

Loadings:
   Comp.1 Comp.2 Comp.3 Comp.4
V1  0.362 -0.657 -0.581  0.317
V2        -0.730  0.596 -0.324
V3  0.857  0.176        -0.480
V4  0.359         0.549  0.751

               Comp.1 Comp.2 Comp.3 Comp.4
SS loadings      1.00   1.00   1.00   1.00
Proportion Var   0.25   0.25   0.25   0.25
Cumulative Var   0.25   0.50   0.75   1.00
```

First component explains 92% of data variance

Example: Eigenfaces

- PCA applied to images of human faces
- Reduce dimensionality to set of basis images
- All other images are linear combo of these "eigenpictures"
- Used for facial recognition



First 40 PCA dimensions

Power method

- Iteratively do matrix vector product:
  - $a_{t+1} = A\, a_t / |A a_t|$
- Guaranteed to converge to the largest eigenvector of A not orthogonal to a_0
- Project A onto orthogonal complement of a_∞ and re-start.

Principal component analysis

- **Task**:
  - Reduce dimensionality of data while capturing intrinsic variability
- **Data representation**:
  - **X** data matrix (n x p)
- **Knowledge representation**:
  - p x m matrix (for *m* < *p*) of projection weights that represent:

    Set of *m* alternative dimensions, where each dimension is represented by a p-dimensional vector of weights (e.g., [0.36, -0.08, 0.86, 0.36])

Principal component analysis

- **Learning**:
  - **Scoring function**:
    1) Maximize variance along each orthogonal direction
    2) Minimize squared deviation from original points to projected points
    (can show these two are equivalent)
  - **Search**: *Implicit* search by analytically determining basis vectors with best score (achieved by solving eigensystem with the covariance matrix ∑)

- **Inference**:

PCA complexity
O(np^2+p^3)

  - Project points into new m-dimensional space
    - E.g., $$x'_1 = \mathbf{a}_1 \mathbf{x} = \sum_{j=1}^{p} a_{1j} x_j$$