Principal component analysis

References:
Applied Machine Learning (Forsyth) https://link.springer.com/book/10.1007/978-3-030-18114-7
sections 4.3.2, 5.1.1 - 5.1.4

Primer from section 4.3.2 on diagonalizing matrices

- a matrix M is symmetric if $M = M^T$
- S = $d \times d$ matrix, u: $d \times 1$ vector, λ = scalar

$$Su = \lambda u$$

eigenvalue

eigenvector

- Symmetric matrix: eigenvalues are real
- d distinct eigenvectors normal to one another
- Stack these into a matrix $U = [u_1, \dots, u_d]$
- Now:

$$SU = U\Lambda$$

orthonormal matrix

diagonal matrix

- convention: elements of U are ordered so elements of Λ are sorted along diagonal, largest first

to convert any symmetric matrix S to diagonal form: $U^T S U = \Lambda$ ← result

input

matrix of eigenvectors

PCA

∴ objective: encode high dimensional data into a lower dimensional space for analysis and/or visualization

$\{x\}$ = dataset of N-dimensional vectors

→ subtract the mean from the dataset to get $\{m\}$

$$m_i = x_i - \text{mean}(\{x\})$$

→ diagonalize covariance matrix (note that $\text{cov}(\{m\}) = \text{cov}(\{x\})$)

$$U^T \underbrace{\text{cov}(\{x\})}_{\text{covariance matrix}} \underbrace{U}_{\text{matrix of eigenvectors}} = \underbrace{\Lambda}_{\text{diagonal matrix of eigenvalues}}$$

Note: the book uses $\Sigma = \text{cov}(\{x\})$

→ form dataset $\{r\}$ with:

$$r_i = U^T m_i = U^T(x_i - \text{mean}(\{x\}))$$

→ mean of $\{r\}$ is 0, covariance is diagonal

many, or most of the diagonal entries in the covariance matrix are small

Calculating the error:

→ datapoint r_i has d components, choose an s s.t. $s < d$ to represent r_i in lower (s) dimensions, call this p_i

error: $$\frac{1}{N}\sum_i [(r_i - p_i)^T (r_i - p_i)]$$

$p_i = r_i$ in components $0 - s$

$p_i = 0$ in components $s+1 - d$

so error =

$$\frac{1}{N}\sum_i \left[\sum_{j=s+1}^{j=d} (r_i^{(j)})^2\right]$$

$$= \sum_{j=s+1}^{j=d} \left[\frac{1}{N}\sum_i (r_i^{(j)})^2\right] = \sum_{j=s+1}^{j=d} \text{var}(\{r^{(j)}\})$$

$$= \sum_{j=s+1}^{j=d} \lambda_j$$

- eigenvalues of the covariance matrix give the error when that component is left out
- reprojection error = sum of eigenvalues of components that were left out.

Representing Data on Principal Components

$$\hat{x}_i = U p_i + \text{mean}(\{x\})$$

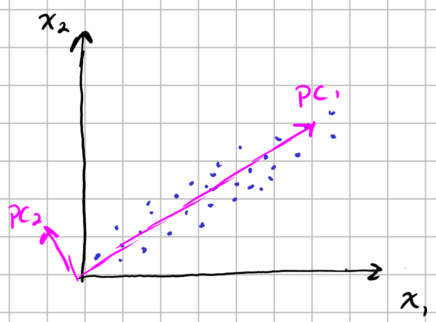
weighted sum of first s columns of U

$$\hat{x}_i = \sum_{j=1}^{s} w_{ij} u_j + \text{mean}(\{x\})$$

$$w_{ij} = r_i^{(j)} = (x_i - \text{mean}(\{x\}))^T u_j$$

$$\hat{x}_i = \text{mean}(\{x\}) + \sum_{j=1}^{s} [u_j^T (x_i - \text{mean}(\{x\}))] u_j$$

Visual explanation:



PC1: Points in direction w/ highest Variance

PC2: Orthogonal to PC1, contains next largest Variation

in Principle, you can collapse this dataset to 1D along PC1 and loose a minimal amount of information

eigenvalues of covariance matrix tell you how much information is encoded in each PC