MATH 250: Mathematical Data Visualization

Singular value decomposition and principal components analysis

Peter A. Gao

February 21, 2024

San José State University

Singular Value Decomposition

Review: SVD

Any matrix $A \in \mathbb{R}^{m \times n}$ can be factorized

$$A = U \Sigma V^\top$$

with

- $U \in \mathbb{R}^{m \times m}$ an orthogonal matrix of the **left singular vectors** of A
- $\Sigma \in \mathbb{R}^{m \times n}$ an $m \times n$ **singular value** matrix
- $V \in \mathbb{R}^{n \times n}$ an orthogonal matrix of the **right singular vectors** of A

Review: SVD

The columns of U form an orthonormal basis for the **column space** of A while the columns of V (rows of $V^\top$) form an orthonormal basis for the **row space**.

The key property to remember for the singular vectors:

$$A\mathbf{v}_i = \sigma_i \mathbf{u}_i, \quad i = 1, \ldots, r$$

Where $\mathbf{v}_i$ and $\mathbf{u}_i$ are the ith right and left singular vectors, respectively, σ_i is the ith singular value, and r is the rank of A.

Example:

$$\begin{bmatrix} 3 & 0 \\ 4 & 5 \end{bmatrix} = \begin{bmatrix} 1/\sqrt{10} & -3/\sqrt{10} \\ 3/\sqrt{10} & 1/\sqrt{10} \end{bmatrix} \begin{bmatrix} 3\sqrt{5} & 0 \\ 0 & \sqrt{5} \end{bmatrix} \begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ -1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix}$$

Decomposing linear transformations

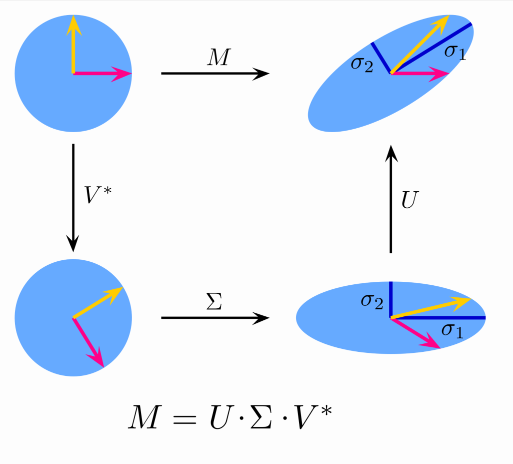


Figure 1: Geometric interpretation of SVD, by Georg-Johann

Review: SVD

In other words:

$$AV = U\Sigma \iff A \begin{bmatrix} & & \\ \mathbf{v}_1 & \cdots & \mathbf{v}_n \\ & & \end{bmatrix} = \begin{bmatrix} & & \\ \mathbf{u}_1 & \cdots & \mathbf{u}_m \\ & & \end{bmatrix} \left[\begin{array}{cc|c} \sigma_1 & & \\ & \sigma_2 & \mathbf{0} \\ \hline & \mathbf{0} & \mathbf{0} \end{array}\right]$$

We can also write A in a reduced SVD form:

$$AV_r = U_r\Sigma_r$$

making Σ_r a diagonal $r \times r$ matrix and removing the last singular vectors from V and U.

Review: SVD as sum of rank-1 matrices

We can also write A as a sum of rank-1 matrices:

$$A = \sigma_1 \mathbf{u}_1 \mathbf{v}_1^\top + \cdots + \sigma_r \mathbf{u}_r \mathbf{v}_r^\top$$

and obtain the best rank-k approximation:

$$A_k = \sigma_1 \mathbf{u}_1 \mathbf{v}_1^\top + \cdots + \sigma_k \mathbf{u}_k \mathbf{v}_k^\top$$

Theorem

(Eckart-Young) If B has rank k, then $||A - A_k|| \leq ||A - B||$ in either the Frobenius or L^2 norm.

Review: Other SVD properties

- V contains orthornormal eigenvectors of $A^\top A$ and U contains orthonormal eigenvectors of $AA^\top$.
- $\sigma_1^2, \ldots, \sigma_r^2$ are the nonzero eigenvalues of both $A^\top A$ and $AA^\top$.
- If S is symmetric positive definite, $U\Sigma V^\top = Q\Lambda Q^\top$.
- $\mathbf{v}_1$ maximizes $||A\mathbf{x}||/||\mathbf{x}||$, achieving a value of σ_1.

Application: Image compression

Figure 2: Stephan's quintet

Application: Image compression

Each pixel is a value from 0-255 representing a color from white to black.

We can thus treat this image as a matrix, compute its SVD and the best rank-k approximation.

Plotting the rank-k approximation yields a compressed version of our original image.

Application: Image compression

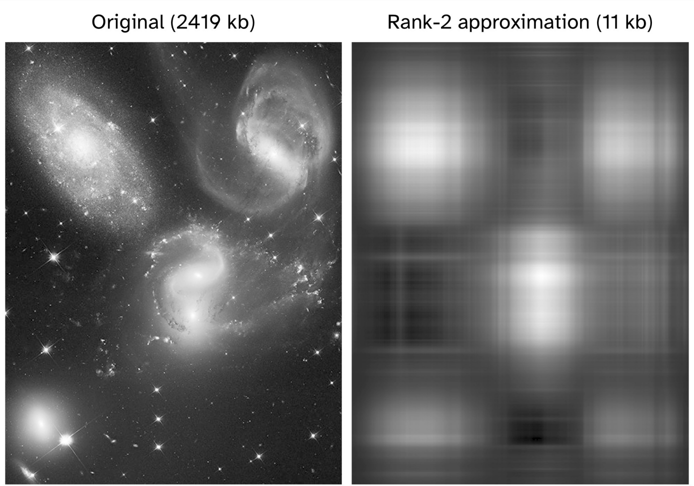


Figure 3: Stephan's quintet

Ranks of common flags

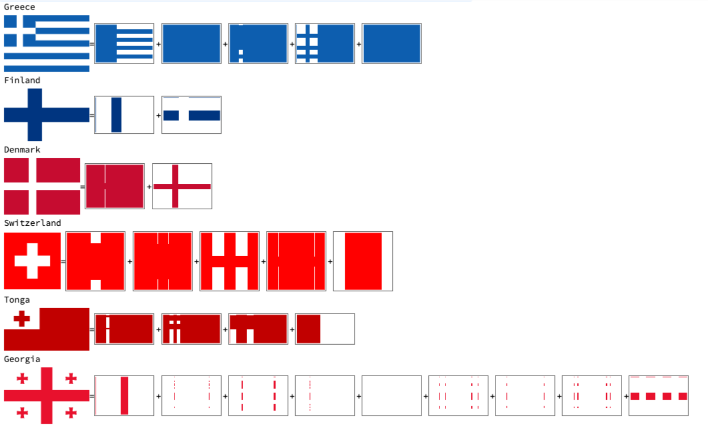


Figure 4: Rank-1 decompositions of common flags by Yaroslav Bulatov

Application: Image compression

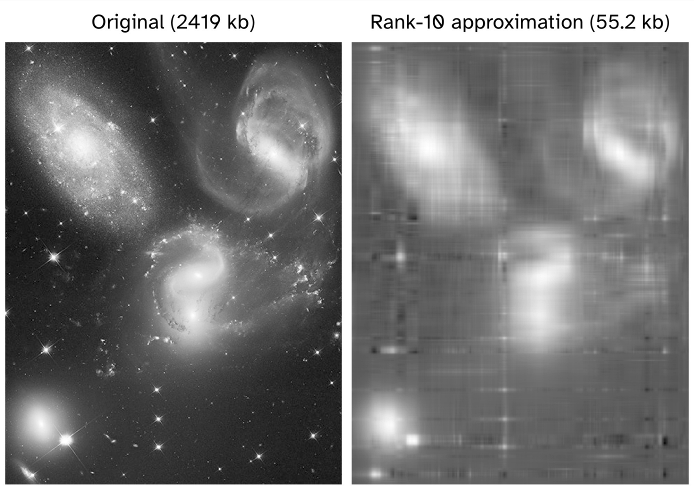


Figure 5: Stephan's quintet

Application: Image compression

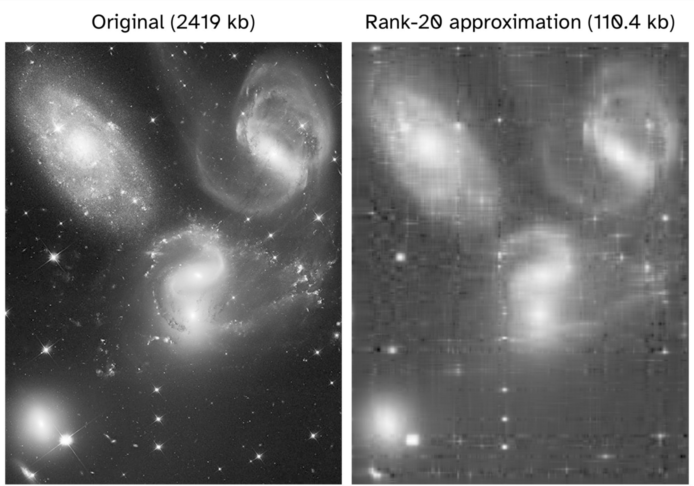


Figure 6: Stephan's quintet

Application: Image compression

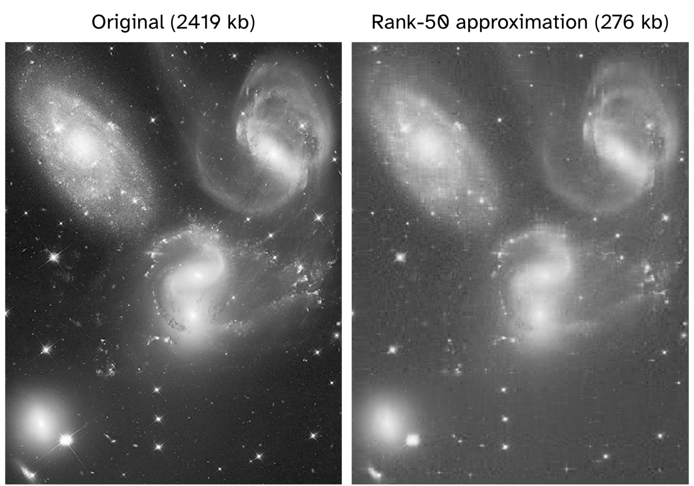


Figure 7: Stephan's quintet

Application: Image compression

The SVD provides a crude approach to image compression, which is the "best" in the sense that it minimizes the matrix distance between these images.

However, when viewing two images, this may not be the right "distance" to be using.

When noise has been added to our image, the SVD can also be used to denoise and clean up images.

Application: Handwritten digits

USPS handwritten digits data:

- 9298 16 x 16 images of handwritten digits, split into training and test datasets.
- Centered and scaled to be the same size.

Example: handwritten digits

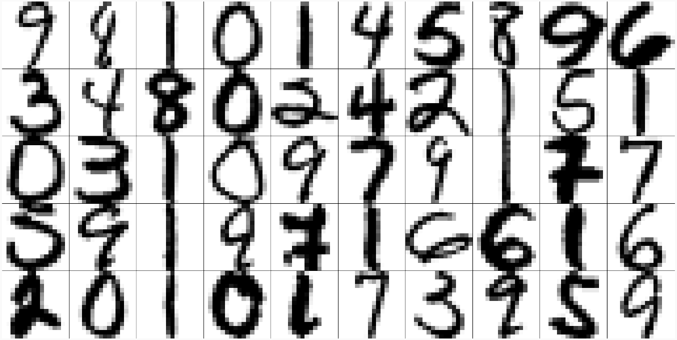


Figure 8: Handwritten 16 x 16 digits from USPS dataset [1]

Example: handwritten digits

	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	[,7]	[,8]	[,9]	[,10]	[,11]	[,12]	[,13]	[,14]	[,15]	[,16]
[1,]	0	0	0	0	0	11	167	197	29	0	0	0	0	0	0	0
[2,]	0	0	0	0	22	207	255	204	0	0	0	0	0	0	0	0
[3,]	0	0	0	95	248	255	160	7	0	0	0	0	0	0	0	0
[4,]	0	58	175	255	255	226	117	146	117	88	88	29	0	0	0	0
[5,]	84	255	255	255	204	145	145	204	145	174	229	255	197	80	0	0
[6,]	32	65	36	7	0	0	0	0	0	0	3	160	255	255	65	0
[7,]	0	0	0	0	0	0	0	0	0	0	0	204	255	236	21	0
[8,]	0	0	0	0	0	0	0	0	0	59	175	255	225	40	0	0
[9,]	0	0	0	0	0	0	22	110	226	255	233	145	0	0	0	0
[10,]	0	0	22	132	190	219	255	255	255	255	190	132	73	0	0	0
[11,]	0	0	7	101	130	72	14	14	14	72	101	159	251	212	37	0
[12,]	0	0	0	0	0	0	0	0	0	0	0	0	25	255	255	44
[13,]	0	0	0	0	0	0	0	0	0	0	0	0	26	255	255	101
[14,]	0	0	116	95	0	0	0	0	0	0	0	44	193	255	247	0
[15,]	0	0	0	138	154	37	37	37	66	125	212	255	236	130	21	0
[16,]	0	0	0	0	50	108	166	196	196	196	137	79	10	0	0	0

Figure 9: Handwritten digit from USPS dataset, as matrix

The "typical" digits

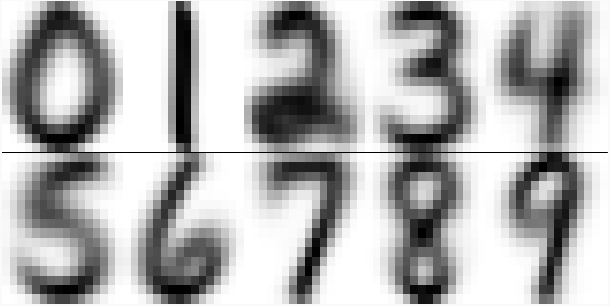


Figure 10: Centroids from USPS dataset (element-wise means)

A naive classification algorithm

1. For each new image i, calculate its distance from the centroids 0-9.
2. Label the new image i based on the closest centroid.

This achieves 75% accuracy. Note the work needed to create all these black-and-white, centered images.

Creating SVD-based representations of each digit

Let n_i be the number of images of digit i in the training set. For each digits, construct a $n_i \times 256$ matrix:

n_i rows $\boxed{A}$

256 columns

The right singular vectors $\mathbf{v}_i$ of A form an orthonormal basis in the space of images.

For a given digit, the first few singular vectors can be used to reconstruct each image in the training set.

SVD basis classification

1. For each new image i, calculate its representation in the SVD basis for each digit.
2. Label the new image i based on the most accurate representation.

For an unknown image $\mathbf{z}$, we can approximate it in a basis using a least squares solution:

$$\min_{\mathbf{c}} ||\mathbf{z} - \sum_{i=1}^{k} c_i \mathbf{v}_i||$$

We can repeat this process for each basis for 0-9 and identify the digit that yields the most accurate representation.

SVD basis classification

The accuracy of this classification method depends on the dimension k of the basis:

# basis images	1	2	4	6	8	10
accuracy	80	86	90	90.5	92	93

Figure 11: Accuracy by number of basis images [2]

Principal components analysis

PCA vs. SVD

You may often hear "PCA is just SVD." It is–sort of.

SVD	**PCA**
- a matrix method	- a data analysis method
- $m \times n$ matrix	- $n \times p$ data matrix

Let's start with PCA and show how it relates to the SVD.

Reviewing statistics

Definition

The **covariance** between two random variables X and Y is defined as

$$\mathrm{Cov}(X, Y) = E[(X - E(X))(Y - E(Y))]$$

Remark

If $\mathbf{x}$ is a p-dimensional random vector, its **covariance matrix** is defined to be

$$\mathrm{Cov}(\mathbf{x}) = E[(\mathbf{x} - E(\mathbf{x}))(\mathbf{x} - E(\mathbf{x}))^\top]$$

Thus, the covariance matrix is positive semidefinite.

Reviewing statistics

$$\mathrm{Cov}(\mathbf{x}) = E[(\mathbf{x} - E(\mathbf{x}))(\mathbf{x} - E(\mathbf{x}))^\top]$$

$$= \begin{bmatrix} \mathrm{Var}(X_1) & \mathrm{Cov}(X_1, X_2) & \cdots & \mathrm{Cov}(X_1, X_p) \\ \mathrm{Cov}(X_1, X_2) & \mathrm{Var}(X_2) & \cdots & \mathrm{Cov}(X_2, X_p) \\ \vdots & \vdots & \ddots & \vdots \\ \mathrm{Cov}(X_1, X_p) & \mathrm{Cov}(X_2, X_p) & \cdots & \mathrm{Var}(X_p) \end{bmatrix}$$

Remark

As a result,

$$\mathrm{Cov}(A\mathbf{x} + \mathbf{b}) = A[\mathrm{Cov}(\mathbf{x})]A^\top$$

Reviewing statistics

If we have a sample of iid random vectors $\mathbf{x}_1, \ldots, \mathbf{x}_n \sim \mathbf{x}$, we can combine them into an $n \times p$ data matrix:

$$X = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{bmatrix}$$

Definition

The **sample covariance** of $\mathbf{x}$ is defined as

$$\widehat{S} = \frac{1}{n}\sum_{i=1}^{n}(\mathbf{x}_i - \overline{\mathbf{x}})(\mathbf{x}_i - \overline{\mathbf{x}})^\top = \frac{1}{n}X_c^\top X_c$$

where $\overline{\mathbf{x}}$ is the sample mean and X_c is a centered version of X (Exercise).

PCA: One-dimensional case

Motivation

We want to project our p-dimensional data into a simpler q-dimensional space. We will try to choose the "most important" q dimensions (principal components) along which the data have maximum variance.

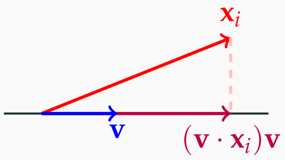
We can start with the example where $q = 1$. What does it mean to find the "optimal" one-dimensional projection of our data? Assume all of our data is centered, so the mean of each column of X is zero.

PCA: One-dimensional case

Idea (from Shalizi [3])

Choose the unit vector $\mathbf{v}$ such that when we project our data vectors $\mathbf{x}_1, \ldots, \mathbf{x}_n$ onto $\mathbf{v}$, the residual error is minimized:

$$\mathrm{MSE}(\mathbf{v}) = \frac{1}{n} \sum_{i=1}^{n} ||\mathbf{x}_i - (\mathbf{v} \cdot \mathbf{x}_i)\mathbf{v}||^2$$



PCA: Outline

Idea (from Shalizi [3])

This turns out to be equivalent to maximizing the sample variance of lengths of the projections onto $\mathbf{v}$ (since the columns of X are centered):

$$
\begin{aligned}
\widehat{\mathrm{Var}}(\mathbf{v}\cdot\mathbf{x}_i) &= \frac{1}{n}\sum_{i=1}^{n}(\mathbf{v}\cdot\mathbf{x}_i)^2 \\
&= \frac{1}{n}(X\mathbf{v})^\top(X\mathbf{v}) \\
&= \frac{1}{n}\mathbf{v}^\top X^\top X\mathbf{v} \\
&= \mathbf{v}^\top\widehat{S}\mathbf{v}
\end{aligned}
$$

PCA: One-dimensional case

In other words, we are simply maximizing the Rayleigh quotient $\mathbf{v}^\top \widehat{S} \mathbf{v}$. How do we find the maximizing $\mathbf{v}$?

PCA: One-dimensional case

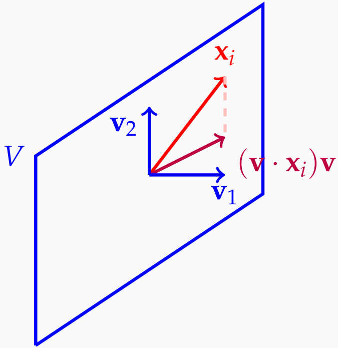
In other words, we are simply maximizing the Rayleigh quotient $\mathbf{v}^\top \widehat{S} \mathbf{v}$. How do we find the maximizing $\mathbf{v}$?

From last week, the maximizing $\mathbf{v}$ is the eigenvector of $\widehat{S}$ with the largest eigenvalue λ_1.

Thus, $\mathbf{v}^\top \widehat{S} \mathbf{v}$ achieves maximum value λ_1.

PCA: Multi-dimensional case

For $q > 1$, we can generalize our approach. Instead of the single vector along which the projected data has maximum variance, we are looking for a k-dimensional plane along which our projected data has maximum variance.



Theorem

The q-dimensional plane along which our projected data has maximum variance has an orthonormal given by the first q eigenvectors of $\widehat{S}$ and the total variance of the projections is given by $\lambda_1 + \cdots + \lambda_k$.

In other words, the q principal components are given by the first q eigenvectors of $\widehat{S} = \frac{1}{n} X^\top X$, or equivalently, of $X^\top X$.

These principal components are orthogonal.

From PCA to SVD

Computing $X^\top X$ is potentially expensive and can lead to an ill-conditioned matrix.

Luckily, the first q eigenvectors of $X^\top X$ are also given by...

From PCA to SVD

Computing $X^\top X$ is potentially expensive and can lead to an ill-conditioned matrix.

Luckily, the first q eigenvectors of $X^\top X$ are also given by... the first q right singular vectors of X (remember, X is centered). The variance captured by the q-dimensional projection plane is $\lambda_1 + \cdots + \lambda_q = \sigma_1^2 + \cdots + \sigma_q^2$.

Summary

Principal components analysis typically involves identifying a set of maximum-variance directions

$$V_q = \begin{bmatrix} \mathbf{v}_1 & \cdots & \mathbf{v}_q \end{bmatrix} \in \mathbb{R}^{p \times q}$$

and the corresponding coordinates of each of the observations in the new basis

$$Y_q = \begin{bmatrix} \mathbf{y}_1 & \cdots & \mathbf{y}_q \end{bmatrix} \in \mathbb{R}^{n \times q}$$

where

$$Y_q = XV_q = U_q \Sigma_q$$

Summary

We can say

- The unit vector $\mathbf{v}_j$ is the jth **principal component** of the data;
- The projected coordinates $Y \in \mathbb{R}^{n \times q}$ are the coefficients obtained by projecting X on the first q **principal components** of the data.

In essence, PCA is a change of coordinate system, where the new axes are the principal components of the data and the new coordinates the projected coefficients

Geometric intepretation

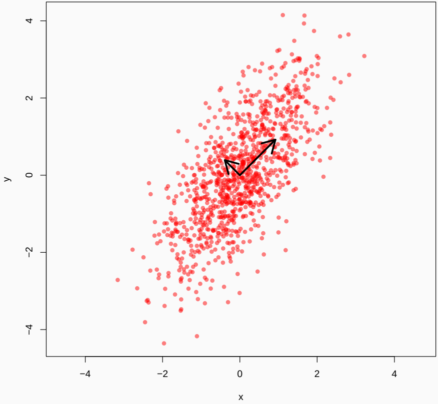


Figure 12: Example data with **principal components** (black)

Geometric interpretation

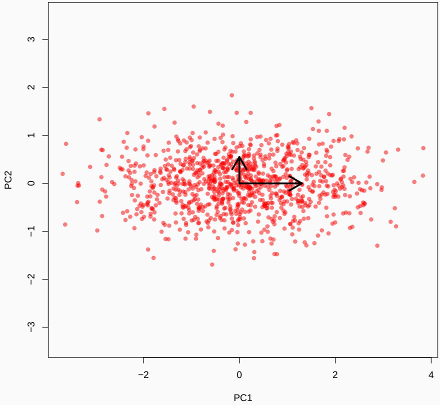


Figure 13: Example **projected data on principal components**

Steps

Given a data matrix $X \in \mathbb{R}^{n \times p}$ and an integer q,

1. Center X by subtracting out the mean, if necessary.
2. Carry out rank-q SVD on $X \approx U_q \Sigma_q V_q^\top$
3. Compute the principal components $Y = U_q \Sigma_q$.

Application: NWSL Teams

In honor of San José’s new soccer team Bay FC, we’ll take a look at data from the National Women’s Soccer League, using the `nwslR` package.

In 2023, there were 12 teams. We can download team-level data for each team including variables like **goals**, **assists**, and **goals allowed**.

Application: Women's Soccer Teams

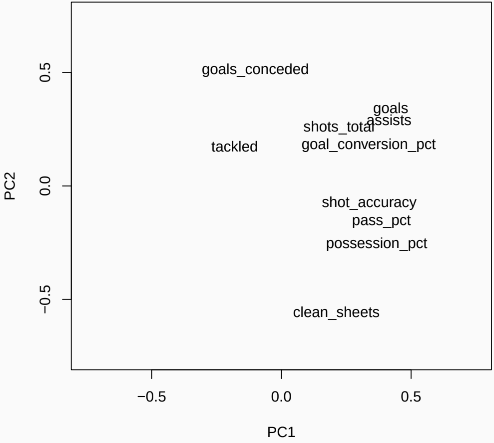


Figure 14: Variable loadings for first two PCs

Application: Women's Soccer Teams

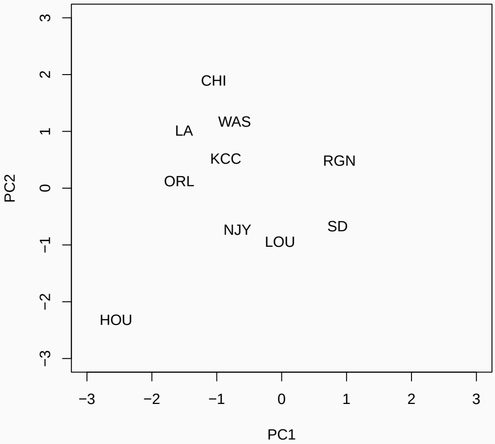


Figure 15: First two PCs NWSL 2023 teams

Application: Women's Soccer Teams

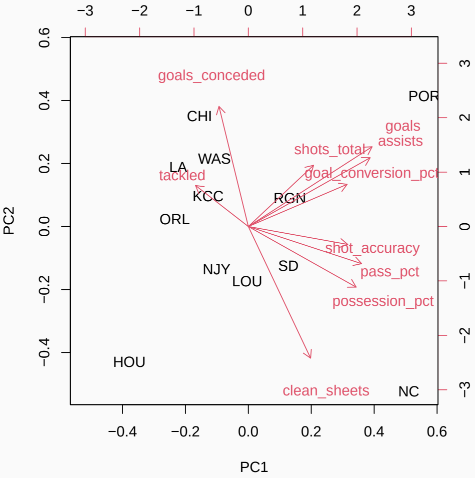


Figure 16: Biplot for NWSL 2023 team-level data

Application: Women's Soccer Teams

Biplots illustrate both:

- data projected on **principal components**: the positions of each observation in the rotated space (columns of U)
- **principal components**: the columns of V contain **variable loadings** (the contribution of each variable to the PCs).

With which variables is the first principal component associated? What about the second principal component?

Application: Women's Soccer Teams

Pos	Team [v·t·e]	Pld	W	D	L	GF	GA	GD	Pts	Qualification
1	San Diego Wave FC	22	11	4	7	31	22	+9	**37**	NWSL Shield, playoffs – semifinals
2	Portland Thorns FC	22	10	5	7	42	32	+10	**35**	Playoffs – semifinals
3	North Carolina Courage	22	9	6	7	29	22	+7	**33**	Playoffs – quarterfinals
4	OL Reign	22	9	5	8	29	24	+5	**32**	
5	Angel City FC	22	8	7	7	31	30	+1	**31**	
6	NJ/NY Gotham FC	22	8	7	7	25	24	+1	**31**	
7	Orlando Pride	22	10	1	11	27	28	−1	**31**	
8	Washington Spirit	22	7	9	6	26	29	−3	**30**	
9	Racing Louisville FC	22	6	9	7	25	24	+1	**27**	
10	Houston Dash	22	6	8	8	16	18	−2	**26**	
11	Kansas City Current	22	8	2	12	30	36	−6	**26**	
12	Chicago Red Stars	22	7	3	12	28	50	−22	**24**	

Figure 17: Standings for NWSL 2023 from Wikipedia

Application: Women's Soccer Teams

The first principal component is associated with several variables about possession and scoring (goals and assists).

The second principal component seems to have more to do with defense (goals conceded and clean sheets).

Application: Women's Soccer Teams

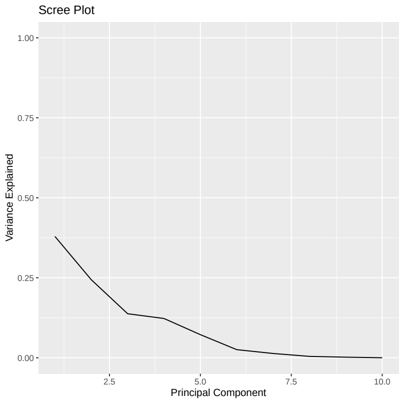


Figure 18: Scree plot for PCA of NWSL 2023 team-level data

Application: State-level characteristics

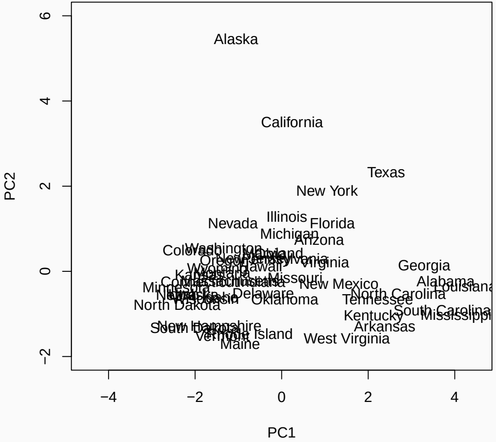


Figure 19: Biplot for state level characteristics, 1977

Application: State-level characteristics

Figure 20: Biplot for state level characteristics, 1977

Application: State-level characteristics

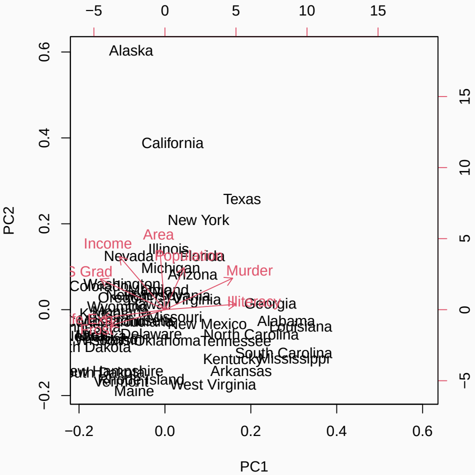


Figure 21: Biplot for state level characteristics, 1977

Application: *The New York Times*

One way to turn documents into numerical data is to represent each document as a **bag of words**: a vector where each component represents the count of a particular word. These vectors are typically quite long and often **sparse**: many values are zero.

We can download a toy dataset from *the New York Times* here [3].

This dataset has 102 rows, and 4432 columns including the class label and and the rest representing the counts for every distinct word that appears in at least one of the stories.

```r
nyt.pca <- prcomp(nyt.frame[, -1])
nyt.latent.sem <- nyt.pca$rotation
```

Application: *The New York Times*

```r
signif(sort(nyt.latent.sem[, 1], decreasing = TRUE)[1:30], 2)
```

```
    music      trio   theater orchestra composers     opera  theaters         m
    0.110     0.084     0.083     0.067     0.059     0.058     0.055     0.054
 festival      east   program         y    jersey   players committee    sunday
    0.051     0.049     0.048     0.048     0.047     0.047     0.046     0.045
     june   concert  symphony     organ   matinee misstated instruments       p
    0.045     0.045     0.044     0.044     0.043     0.042     0.041     0.041
      X.d     april    samuel      jazz   pianist   society
    0.041     0.040     0.040     0.039     0.038     0.038
```

```r
signif(sort(nyt.latent.sem[, 1], decreasing = FALSE)[1:30], 2)
```

```
      she       her        ms         i      said    mother    cooper        my  painting   process
   -0.260    -0.240    -0.200    -0.150    -0.130    -0.110    -0.100    -0.094    -0.088    -0.071
paintings        im        he       mrs        me  gagosian       was   picasso     image sculpture
   -0.070    -0.068    -0.065    -0.065    -0.063    -0.062    -0.058    -0.057    -0.056    -0.056
     baby   artists      work    photos       you    nature    studio       out      says      like
   -0.055    -0.055    -0.054    -0.051    -0.051    -0.050    -0.050    -0.050    -0.050    -0.049
```

Application: *The New York Times*

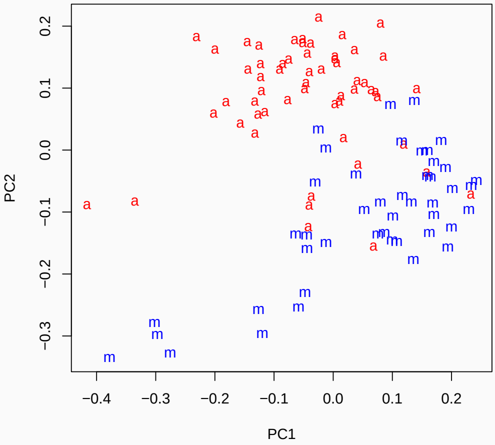


Figure 22: Projection of articles on the first two PCs.

Pitfalls of PCA

Overinterpretation

- It is common to try to interpret the principal components, but it’s important to be cautious. We should be wary of "reifying" concepts.
- A key example comes from Cavalli-Sforza (1997), who describes a PCA with a data matrix where the rows represent locations and columns represent frequency of gene variants.

Cavalli-Sforza et al. (1997): Population migration from PCs?

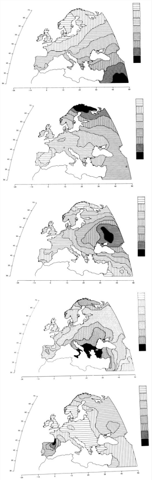


Figure 23: First five PCs by Cavalli-Sforza (1997)

Cavalli-Sforza et al. (1997): Population migration from PCs?

*"Hidden patterns in the geography of Europe shown by the first five principal components, explaining respectively 28%, 22%, 11%, 7%, and 5% of the total genetic variation for 95 classical polymorphisms. **The first component is almost superimposable to the archaeological dates of the spread of farming from the Middle East between 10,000 and 6,000 years ago.** The second principal component parallels a probable spread of Uralic people and/or languages to the northeast of Europe. The third is very similar to the spread of pastoral nomads (and their successors) who domesticated the horse in the steppe towards the end of the farming expansion, and are believed by some archaeologists and linguists to have spread most Indo-European languages to Europe. The fourth is strongly reminiscent of Greek colonization in the first millennium B.C. The fifth corresponds to the progressive retreat of the boundary of the Basque language. Basques have retained, in addition to their language, believed to be descended from an original language spoken in Europe, some of their original genetic characteristics."*

Overinterpretation

Shalizi [3] reviews a paper by Novembre and Stephens [4] that points out that these kinds of patterns are expected when carrying out PCA with **any** spatially correlated data.

Novembre and Stephens simulated data based on genetic diffusion processes, without any migration/population expansion and produced similar maps.

Overinterpretation

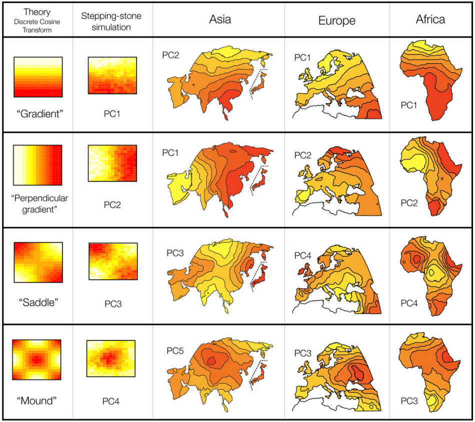


Figure 24: PCs based on simulated data with no migration

Overinterpretation

In other words, Novembre and Stephens do not disprove that migration happened, but they show that PCA of Cavalli-Sforza et al. doesn't provide strong evidence of the migration.

PCs must thus be interpreted with caution.

Other uses of PCA/SVD

1. Timeseries analysis
2. Spatial data analysis
3. Matrix completion
4. … you tell me!

References

- J. Hull, “A database for handwritten text recognition research,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, pp. 550–554, May 1994.
- L. Eldén, *Matrix Methods in Data Mining and Pattern Recognition.* **Fundamentals of Algorithms, Society for Industrial and Applied Mathematics, Jan. 2007.**
- C. R. Shalizi, *Advanced Data Analysis from an Elementary Point of View (draft).*
- J. Novembre and M. Stephens, “Interpreting principal component analyses of spatial population genetic variation,” *Nature genetics*, vol. 40, pp. 646–649, May 2008.