

# Eigenvalue Decomposition

Predictive Modeling & Statistical Learning

Gaston Sanchez

CC BY-SA 4.0

# Matrix Decompositions

# Decompositions

Matrix decompositions, also known as matrix factorizations

$$\mathbf{M} = \mathbf{AB} \quad \text{or} \quad \mathbf{M} = \mathbf{ABC}$$

are a means of expressing a matrix as a product of usually two or three **simpler** matrices.

# Importance of Decompositions

## What for?

Matrix decompositions make it easier to study the properties of matrices. Likewise, many computation tasks become easier with decompositions.

They play a relevant role in multivariate data analysis. Often, the solution to many techniques are obtained (or derived) from a matrix decomposition.

# Decompositions: What for?

- ▶ solving systems of linear equations
- ▶ inverting a matrix
- ▶ analyzing numerical stability of a system
- ▶ understanding the structure of data
- ▶ finding basis for column space (or row space) of a matrix

# Some Assumptions

## Real Matrices

We will assume all matrices to be real matrices, i.e. matrices containing elements in the set of Real numbers.

## Dimensions $n \geq p$

Unless otherwise stated, we will also assume matrices with more rows than columns.

# Decompositions

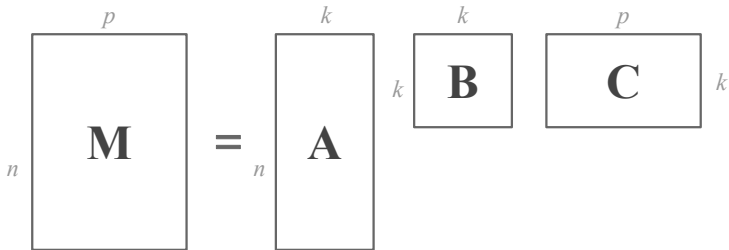
A matrix decomposition can be described by an equation:

$$\mathbf{M} = \mathbf{A}\mathbf{B}\mathbf{C}$$

where the dimensions of the matrices are as follows:

- ▶  $\mathbf{M}$  is  $n \times p$  (assume  $n > p$ )
- ▶  $\mathbf{A}$  is  $n \times k$  (usually  $k < p$ )
- ▶  $\mathbf{B}$  is  $k \times k$  (usually diagonal)
- ▶  $\mathbf{C}$  is  $k \times p$

# Matrix Decomposition





# Interpreting Decompositions

The equation that describes a decomposition:

$$\mathbf{M} = \mathbf{ABC}$$

- ▶ does not explain how to compute one
- ▶ does not explain how such decomposition can reveal the structures implicit in a data matrix.
- ▶ Seeing how a matrix decomposition reveals structure in a dataset is more complicated
- ▶ Each decomposition reveals a different kind of implicit structure

# Types of matrices

## Two types of matrices

We concentrate on the two types of matrices important in statistics:

- ▶ general **rectangular** matrices used to represent data tables.
- ▶ **positive semi-definite** matrices used to represent covariance matrices, correlation matrices, and any matrix that results from a crossproduct.

# Two Special Decompositions

## EVD and SVD

There are many types of matrix decompositions but for now we are going to consider only two:

- ▶ Eigen-Value Decomposition (EVD)
- ▶ Singular Value Decomposition (SVD)

# EVD

## Eigenvalue Decomposition

- ▶ EVD applies to square matrices in general.
- ▶ A special type of square matrices are **symmetric** matrices.
- ▶ In data analysis methods, these matrices usually appear in the form of cross-product association matrices:  
e.g.  $\mathbf{X}^T\mathbf{X}$  and  $\mathbf{X}\mathbf{X}^T$
- ▶ The attractive thing about EVD is that when applied to symmetric matrices the results have a “simple” nice structure.

# Eigenvalue and Eigenvector

Consider the matrix  $\mathbf{A}$ :

$$\mathbf{A} = \begin{bmatrix} 3 & -2 \\ 1 & 0 \end{bmatrix}$$

associated to the linear transformation  $T(\mathbf{x})$  given by:

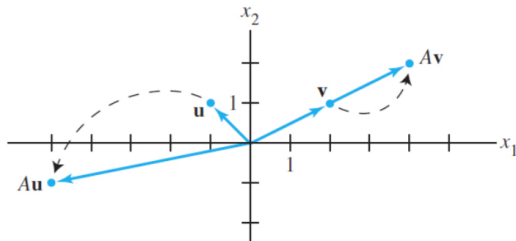
$$T(\mathbf{x}) = \begin{bmatrix} 3 & -2 \\ 1 & 0 \end{bmatrix} \mathbf{x} = \mathbf{A}\mathbf{x}$$

and assume vectors  $\mathbf{v} = (2, 1)$  and  $\mathbf{u} = (-1, 1)$

# Eigenvalue and Eigenvector

$$T(\mathbf{v}) = \begin{bmatrix} 3 & -2 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 2 \\ 1 \end{bmatrix} = \begin{bmatrix} 4 \\ 2 \end{bmatrix}$$

$$T(\mathbf{u}) = \begin{bmatrix} 3 & -2 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} -1 \\ 1 \end{bmatrix} = \begin{bmatrix} -5 \\ -1 \end{bmatrix}$$



$\mathbf{u}$  is changing its direction, but not  $\mathbf{v}$

# Eigenvalue and Eigenvector

Given an  $n \times n$  matrix  $\mathbf{M}$ ,  $\lambda$  is an **eigenvalue** of  $\mathbf{M}$  if there exists a non-trivial solution  $\mathbf{v}$  of the equation:

$$\mathbf{M}\mathbf{v} = \lambda\mathbf{v}$$

The solution  $\mathbf{v}$  is the **eigenvector** associated to the eigenvalue  $\lambda$

# Eigen-Value Decomposition

## EVD

An  $n \times n$  **symmetric matrix**  $\mathbf{M}$  can be decomposed as:

$$\mathbf{M} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$$

where

- ▶  $\mathbf{U}$  is a  $n \times p$  column **orthonormal** matrix containing the eigen-vectors of  $\mathbf{M}$
- ▶  $\mathbf{\Lambda}$  is a  $p \times p$  **diagonal** matrix containing the eigen-values of  $\mathbf{M}$



# EVD

$$\mathbf{M} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$$

$$\mathbf{M} = \begin{bmatrix} u_{11} & \cdots & u_{1p} \\ u_{21} & \cdots & u_{2p} \\ \vdots & \ddots & \vdots \\ u_{n1} & \cdots & u_{np} \end{bmatrix} \begin{bmatrix} \lambda_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \lambda_p \end{bmatrix} \begin{bmatrix} u_{11} & \cdots & u_{n1} \\ u_{12} & \cdots & u_{n2} \\ \vdots & \ddots & \vdots \\ u_{1p} & \cdots & u_{np} \end{bmatrix}$$

# Eigenvectors

Vectors, which under a given transformation  $\mathbf{M}$  map into themselves or multiples of themselves, are called invariant vectors under that transformation. It follows that such vectors satisfy the relation:

$$\mathbf{M}\mathbf{x} = \lambda\mathbf{x}$$

where  $\lambda$  is a scalar.

# Eigenvectors

The matrix equation:

$$\mathbf{M}\mathbf{x} = \lambda\mathbf{x}$$

can be rearranged as follows:

$$\mathbf{M}\mathbf{x} - \lambda\mathbf{x} = \mathbf{0}$$

# Eigenvectors

Given

$$\mathbf{M}\mathbf{x} - \lambda\mathbf{x} = \mathbf{0}$$

We can factor out  $\mathbf{x}$

$$(\mathbf{M} - \lambda\mathbf{I})\mathbf{x} = \mathbf{0}$$

# Eigenvectors

Obtaining the eigenstructure of a (square) matrix involves solving the **characteristic equation**

$$\det(\mathbf{M} - \lambda_i \mathbf{I}) = 0$$

If  $\mathbf{M}$  is of order  $n \times n$ , then we can obtain  $n$  roots of the equation. These roots are called the **eigenvalues**.

# EVD in R

# `eigen()` in R

## `eigen()` function

R provides the function `eigen()` to perform an eigenvalue decomposition of a square matrix.

## `eigen()` output

A list with the following components

- ▶ `values` a vector containing the eigenvalues
- ▶ `vectors` a matrix whose columns contain the eigenvectors

# EVD example in R

```
# X'X matrix
set.seed(22)
X <- as.matrix(USArrests)
XtX <- t(X) %*% X

# eigenvalue decomposition
EVD = eigen(XtX)

# elements returned by eigen()
names(EVD)

## [1] "values" "vectors"

# vector of eigenvalues
(lambdas = EVD$values)

## [1] 2013735.2431 37957.1103 2084.9578 326.5089
```



## EVD example in R (con't)

```
# matrix of eigenvectors
```

```
(V <- EVD$vectors)
```

```
##           [,1]      [,2]      [,3]      [,4]  
## [1,] -0.04239181  0.01616262  0.06588426  0.99679535  
## [2,] -0.94395706  0.32068580 -0.06655170 -0.04094568  
## [3,] -0.30842767 -0.93845891 -0.15496743  0.01234261  
## [4,] -0.10963744 -0.12725666  0.98347101 -0.06760284
```

# Properties of Matrix Eigenstructures

# Properties of Eigenstructures

1. The sum of the eigenvalues of a matrix  $\mathbf{A}$  equals the sum of the main diagonal elements (i.e. the trace) of the matrix.

$$\sum_{i=1}^n \lambda_i = \sum_{i=1}^n a_{ii}$$

2. The product of the eigenvalues of a matrix  $\mathbf{A}$  equals the determinant of  $\mathbf{A}$

$$\prod_{i=1}^n \lambda_i = |\mathbf{A}|$$

# Properties of Eigenstructures

**3.** If we have the matrix  $\mathbf{B} = \mathbf{A} + k\mathbf{I}$ , where  $k$  is a scalar, then the eigenvectors of  $\mathbf{B}$  are the same as those of  $\mathbf{A}$ , and the  $i$ -th eigenvalue of  $\mathbf{B}$  is

$$\lambda_i + k$$

where  $\lambda_i$  is the  $i$ -th eigenvalue of  $\mathbf{A}$

**4.** If we have the matrix  $\mathbf{C} = k\mathbf{A}$ , where  $k$  is a scalar, then  $\mathbf{C}$  has the same eigenvectors as  $\mathbf{A}$  and

$$k\lambda_i$$

is the eigenvalue of  $\mathbf{C}$ , where  $\lambda_i$  is the  $i$ -th eigenvalue of  $\mathbf{A}$

# Properties of Eigenstructures

**5.** If we have the matrix  $\mathbf{A}^p$ , where  $p$  is a positive integer, then  $\mathbf{A}^p$  has the same eigenvectors as  $\mathbf{A}$  and

$$\lambda_i^p$$

is the  $i$ -th eigenvalue of  $\mathbf{A}^p$ , where  $\lambda_i$  is the  $i$ -th eigenvalue of  $\mathbf{A}$

**6.** If  $\mathbf{A}^{-1}$  exists, then  $\mathbf{A}^{-p}$  has the same eigenvectors as  $\mathbf{A}$  and

$$\lambda_i^{-p}$$

is the  $i$ -th eigenvalue of  $\mathbf{A}^{-p}$  corresponding to the  $i$ -th eigenvalue of  $\mathbf{A}$

# Properties of Eigenstructures

7. If a symmetric matrix  $\mathbf{A}$  can be written as the product

$$\mathbf{A} = \mathbf{U}\mathbf{D}\mathbf{U}^T$$

where  $\mathbf{D}$  is a diagonal with all entries nonnegative and  $\mathbf{U}$  is an orthogonal matrix of eigenvectors, then

$$\mathbf{A}^{1/2} = \mathbf{U}\mathbf{D}^{1/2}\mathbf{U}^T$$

and it is the case that  $\mathbf{A}^{1/2}\mathbf{A}^{1/2} = \mathbf{A}$

# Properties of Eigenstructures

8. If a symmetric matrix  $\mathbf{A}^{-1}$  can be written as the product

$$\mathbf{A}^{-1} = \mathbf{U}\mathbf{D}^{-1}\mathbf{U}^T$$

where  $\mathbf{D}^{-1}$  is a diagonal with all entries nonnegative and  $\mathbf{U}$  is an orthogonal matrix of eigenvectors, then

$$\mathbf{A}^{-1/2} = \mathbf{U}\mathbf{D}^{-1/2}\mathbf{U}^T$$

and it is the case that  $\mathbf{A}^{-1/2}\mathbf{A}^{-1/2} = \mathbf{A}^{-1}$

# Power Method



# About the Power Method

One of the basic procedures following a successive approximation approach is precisely the **Power Method**.

In its simplest form, the Power Method (PM) allows us to find **the largest** eigenvector and its corresponding eigenvalue.

# About the Power Method

Choose an arbitrary vector  $\mathbf{w}_0$  to which we will apply the symmetric matrix  $\mathbf{S}$  repeatedly to form the following sequence:

$$\mathbf{w}_1 = \mathbf{S}\mathbf{w}_0$$

$$\mathbf{w}_2 = \mathbf{S}\mathbf{w}_1 = \mathbf{S}^2\mathbf{w}_0$$

$$\mathbf{w}_3 = \mathbf{S}\mathbf{w}_2 = \mathbf{S}^3\mathbf{w}_0$$

$$\vdots$$

$$\mathbf{w}_k = \mathbf{S}\mathbf{w}_{k-1} = \mathbf{S}^k\mathbf{w}_0$$

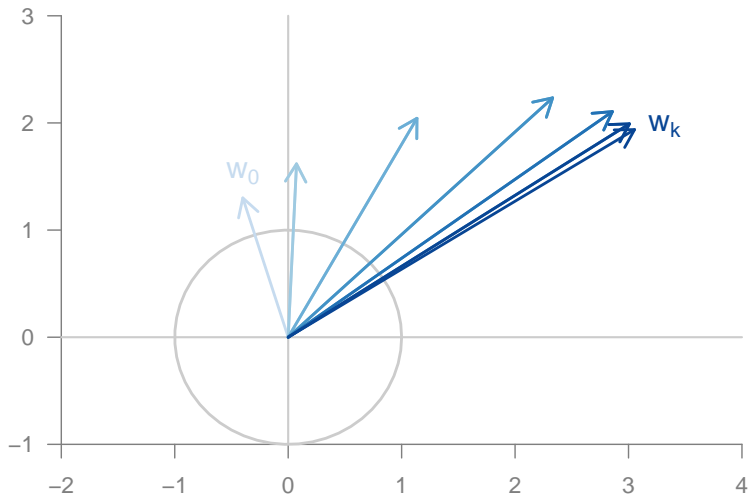
# Power Method: Example

Consider a matrix  $\mathbf{S}$

$$\mathbf{S} = \begin{bmatrix} 3 & 1 \\ 1 & 2 \end{bmatrix}$$

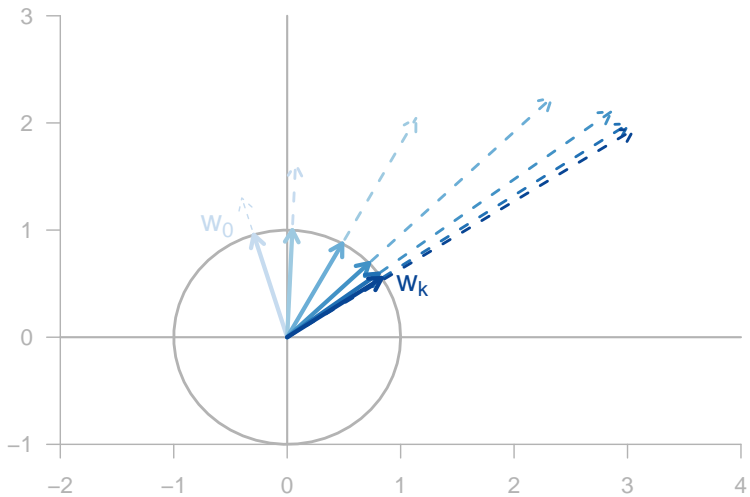
and an initial vector  $\mathbf{w}_0$

$$\mathbf{w}_0 = \begin{bmatrix} -0.4 \\ 1.3 \end{bmatrix}$$



# About the Power Method

- ▶ In practice, we must rescale the obtained vector  $\mathbf{w}_k$  at each step.
- ▶ The rescaling will allows us to judge whether the sequence is converging.
- ▶ After some iterations, the vector  $\mathbf{w}_{k-1}$  and  $\mathbf{w}_k$  will be very similar
- ▶ Assuming a reasonable scaling strategy, the sequence will usually converge to the dominant eigenvector of  $\mathbf{S}$ .



# Dominant Eigenvalue

The obtained vector is the dominant eigenvector. To get the corresponding eigenvalue we calculate the so-called **Rayleigh quotient** given by:

$$\lambda = \frac{\mathbf{w}_k^T \mathbf{S} \mathbf{w}_k}{\mathbf{w}_k^T \mathbf{w}_k}$$

# Remarks

Conditions for the power method to be successfully used:

- ▶ The matrix must have a *dominant* eigenvalue.
- ▶ The starting vector  $\mathbf{w}_0$  must be nonzero.
- ▶ We need to scale each of the vectors  $\mathbf{w}_k$  otherwise the algorithm will “explode”



# PM Pseudocode

Let's consider a more detailed version of the PM algorithm:

1. Start with an arbitrary initial vector  $\mathbf{w}$
2. Obtain product  $\tilde{\mathbf{w}} = \mathbf{S}\mathbf{w}$
3. Normalize  $\tilde{\mathbf{w}}$

e.g. 
$$\mathbf{w} = \frac{\tilde{\mathbf{w}}}{\|\tilde{\mathbf{w}}\|_{p=2}}$$

4. Compare  $\mathbf{w}$  with its previous version
5. Repeat steps 2 till 4 until convergence

# Why does the PM work?

Assume that the matrix  $\mathbf{S}$  has  $p$  eigenvalues  $\lambda_1, \lambda_2, \dots, \lambda_p$ , and that they are ordered in decreasing way

$$|\lambda_1| > |\lambda_2| \geq \dots \geq |\lambda_p|.$$

Note that the first eigenvalue is strictly greater than the second one. This is a very important assumption.

In the same way, we'll assume that the matrix  $\mathbf{S}$  has  $p$  linearly independent vectors  $\mathbf{u}_1, \dots, \mathbf{u}_p$  ordered in such a way that  $\mathbf{u}_j$  corresponds to  $\lambda_j$ .

# Why does the PM work?

The initial vector  $\mathbf{w}_0$  may be expressed as a linear combination of  $\mathbf{u}_1, \dots, \mathbf{u}_p$

$$\mathbf{w}_0 = a_1 \mathbf{u}_1 + \dots + a_p \mathbf{u}_p$$

At every step of the iterative process the vector  $\mathbf{w}_k$  is given by:

$$\mathbf{w}_k = a_1 \lambda_1^k \mathbf{u}_1 + \dots + a_p \lambda_p^k \mathbf{u}_p$$

# Why does the PM work?

Since  $\lambda_1$  is the dominant eigenvalue, the component in the direction of  $\mathbf{u}_1$  becomes relatively greater than the other components as  $k$  increases. If we knew  $\lambda_1$  in advance, we could rescale at each step by dividing by it to get:

$$\left(\frac{1}{\lambda_1^k}\right) \mathbf{w}_k = a_1 \mathbf{u}_1 + \cdots + a_p \left(\frac{\lambda_p^k}{\lambda_1^k}\right) \mathbf{u}_p$$

which converges to the eigenvector  $a_1 \mathbf{u}_1$ , provided that  $a_1$  is nonzero.

# Why does the PM work?

Of course, in real life this scaling strategy is not possible—we don't know  $\lambda_1$ . Consequently, the eigenvector is determined only up to a constant multiple, which is not a concern since the really important thing is the *direction* not the length of the vector.

The speed of the convergence depends on how bigger  $\lambda_1$  is respect with to  $\lambda_2$ , and on the choice of the initial vector  $\mathbf{w}_0$ . If  $\lambda_1$  is not much larger than  $\lambda_2$ , then the convergence will be slow.

# More Remarks

- ▶ The power method is a sequential method.
- ▶ We can obtain  $\mathbf{w}_1, \mathbf{w}_2$ , and so on, step by step.
- ▶ If we only need the first  $k$  vectors, we can stop the procedure at the desired stage.

## Obtaining more eigenvectors?

For **symmetric** matrices, once we've obtained the first eigenvector  $\mathbf{w}_1$  and eigenvalue  $\lambda_1$ , we can compute the second eigenvector by reducing the matrix  $\mathbf{S}$  by the amount explained by the first eigenvector.

This operation of reduction is called **deflation** and the residual matrix is obtained as:

$$\mathbf{S}_1 = \mathbf{S} - \lambda_1 \mathbf{w}_1 \mathbf{w}_1^T$$

To get the second eigenvalue and its corresponding eigenvector, we operate on  $\mathbf{S}_1$  in the same way as the operations on  $\mathbf{S}$ .

# References

- ▶ **Multivariate Analysis** by Maurice Tatsuoka (1988). *Chapter 5: More Matrix Algebra*. Macmillan Publishing.
- ▶ **Mathematical Tools for Applied Multivariate Analysis** by J.D. Carroll, P.E. Green, and A. Chaturvedi (1997). *Chapter 5: Decomposition of Matrix Transformations: Eigenstructures and Quadratic Forms*. Academic Press.
- ▶ **Hand-on Matrix Algebra using R** by Hrishikesh Vinod (2011). *Chapter 9: Eigenvalues and Eigenvectors*. World Scientific.