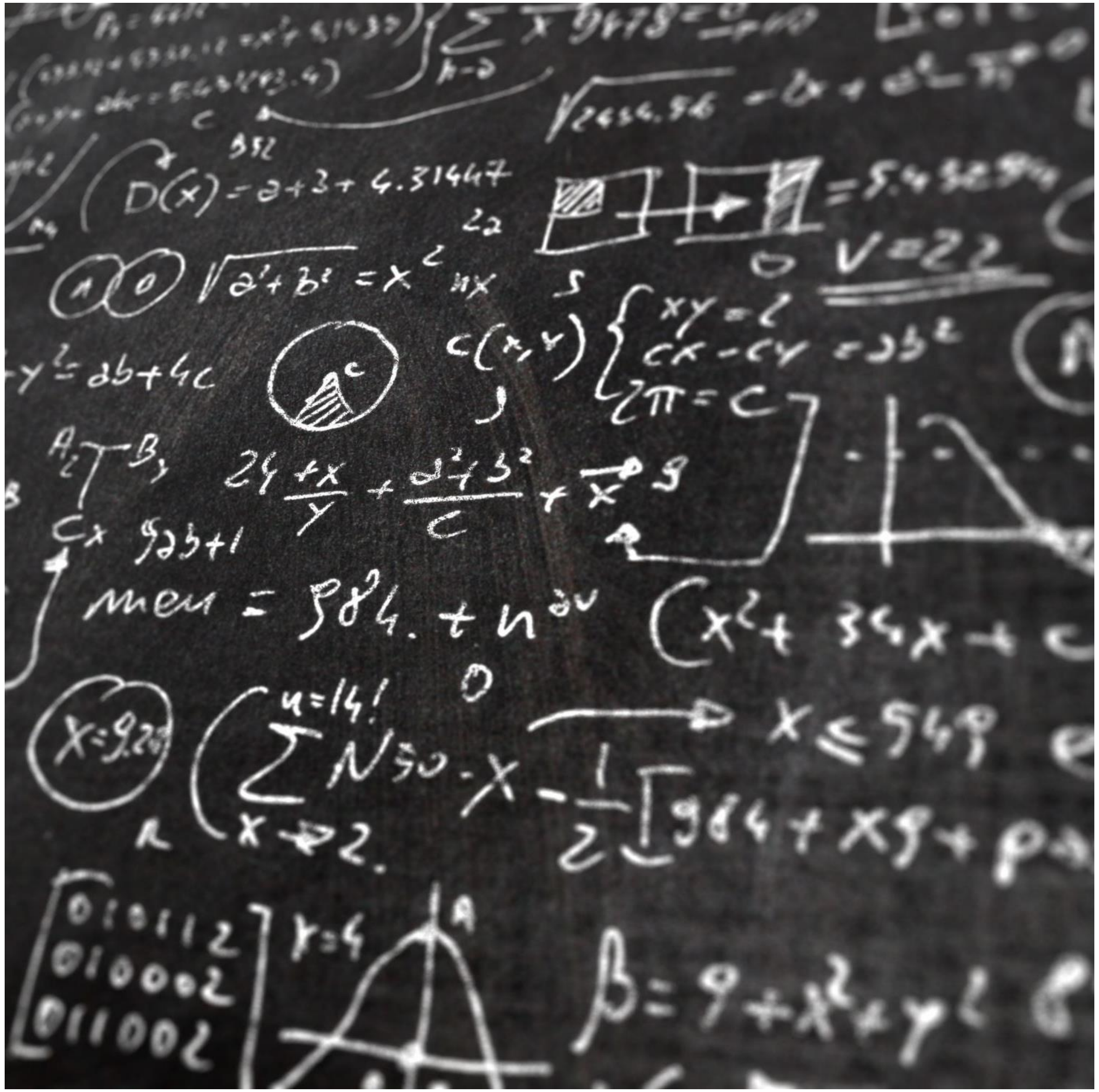
Linear Algebra for data science

Summary of Session 5

Session 5 contents

1. Review: matrices, eigenvalues, and determinants.
2. Review: ‘Solving Systems of Linear Equations’: overdetermined systems.
3. Diagonalization and Eigendecomposition
4. Singular Value Decomposition (SVD)
5. Dimensionality Reduction; Principal Component Analysis (PCA).
6. PCA Using Eigenvectors.
7. PCA Using SVD.



Review

Matrices, Eigenvalues, and Determinants

Determinant of a matrix

You may encounter one of the two definitions below

- If we define M_{ij} = the **submatrix** formed by deleting row $i$and column j (sometimes called the "minor matrix"), then

$$C_{ij} = (-1)^{i+j} \det(M_{ij}) \text{ = cofactor}$$

$$\det(A) = \sum_{i=1}^{n} a_{ij} (-1)^{i+j} \det(M_{ij})$$

- If we define M_{ij} as the minor (the determinant of the **submatrix** formed by deleting row i and column j), then

$$C_{ij} = (-1)^{i+j} M_{ij}$$

$$\det(A) = \sum_{i=1}^{n} a_{ij} (-1)^{i+j} M_{ij}$$

Properties of the determinant

Based on the definition of determinant, for a scalar α,

$$\det(\alpha A) = \alpha^n \det(A)$$

$$\det\left(A^T\right) = \det(A)$$

For an identity matrix

$$\det(\boldsymbol{I}) = 1$$

Also, for two square matrices $\boldsymbol{A}$ and $\boldsymbol{B}$,

$$\det(\boldsymbol{A}\boldsymbol{B}) = \det(\boldsymbol{A})\det(\boldsymbol{B})$$

$$\det\left(AA^{-1}\right) = \det(A)\det\left(A^{-1}\right) = 1$$

For an $m \times n$ matrix $\boldsymbol{A}$ and an $n \times m$ matrix $\boldsymbol{B}$, we have

$$\det(\boldsymbol{I}_m + \boldsymbol{A}\boldsymbol{B}) = \det(\boldsymbol{I}_n + \boldsymbol{B}\boldsymbol{A})$$

Trace of a Matrix

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & & \\ \vdots & & \ddots & \\ a_{n1} & & & a_{nn} \end{bmatrix}$$

The trace of square matrix A is the sum of its diagonal entries

$$\mathrm{Tr}(A) = \sum_{i=1}^{n} a_{i,i}$$

For an $m \times n$ matrix A and an $n \times m$ matrix B, we have

$$\mathrm{Tr}(\boldsymbol{AB}) = \mathrm{Tr}(\boldsymbol{BA})$$

Some properties of the eigenvalues

$$Au = \lambda u$$

The eigenvalues of A are very useful characteristics. In particular,

$$\det(A) = \prod_{i=1}^{n} \lambda_i$$

$$\mathrm{Trace}(A) = \sum_{i=1}^{n} \lambda_i$$

The outer product

- The outer product of two vectors produces a matrix where each element is the product of an element from the first vector and an element from the second vector.
- If the two vectors have dimensions (n) and (m), the resulting matrix will have dimensions $(n \times m)$.

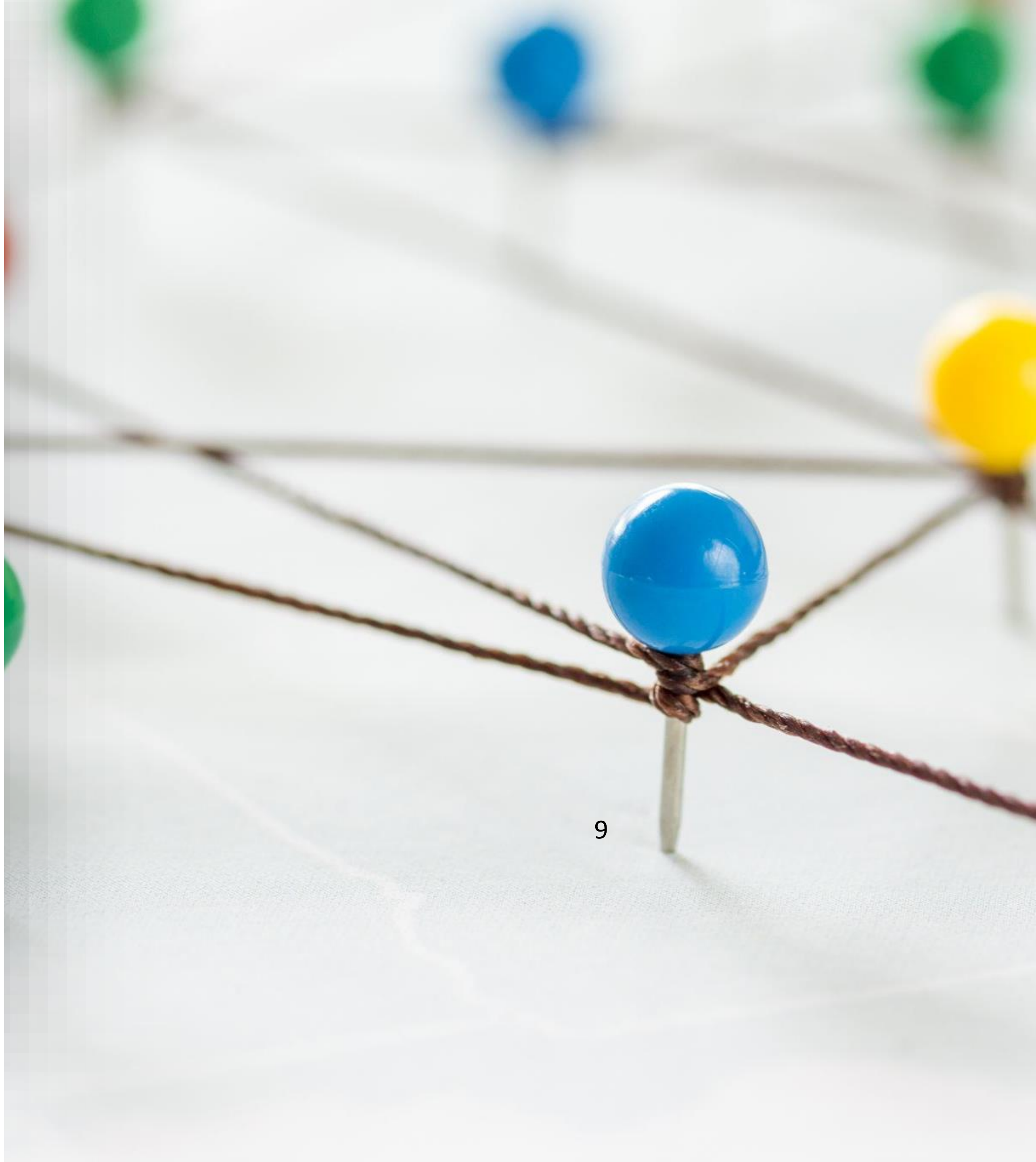
$$\mathbf{u} = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}, \quad \mathbf{v} = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}$$

$$\mathbf{u} \otimes \mathbf{v} = \begin{bmatrix} u_1v_1 & u_1v_2 & u_1v_3 \\ u_2v_1 & u_2v_2 & u_2v_3 \end{bmatrix}$$

$$\mathbf{u} \otimes \mathbf{v} = \mathbf{u}\mathbf{v}^{\mathrm{T}} = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \begin{bmatrix} v_1 & v_2 & v_3 \end{bmatrix}$$

Extra Readings

- https://mathinsight.org/linear_transformation_definition_euclidean
- https://textbooks.math.gatech.edu/ila/linear-transformations.html
- https://www.codeformech.com/determinant-linear-algebra-using-python/



Solving overdetermined systems of linear Equations – Least Squares and Matrix Pseudo Inverse

Review: Solving linear system of equations

Recap for matrices and vectors

$$\begin{bmatrix} 2 & 3 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 5 \\ 7 \end{bmatrix} = \begin{bmatrix} 31 \\ 2 \end{bmatrix}$$

Transformation matrix   Vector X   Output vector

Linear system of equations

$$2x+3y=31$$
$$5x+2y=21$$

Properties

- Variables {x,y} added to each others
- X^2 Is not allowed
- Xy is not allowed
- All variables should be in the left side
- Constants in the right hand side

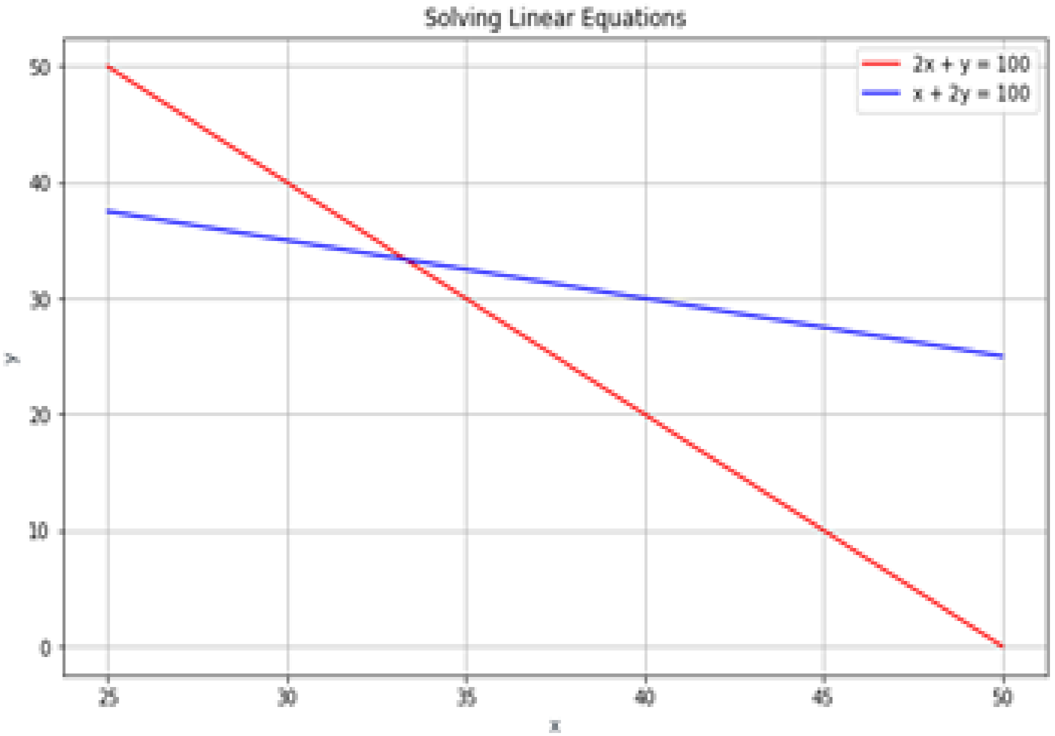
Solving linear system of equations

- $2x + y = 100$
- $x + 2y = 100$

$$\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 100 \\ 100 \end{bmatrix}$$

$AX = V$

$$\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 100 \\ 100 \end{bmatrix}$$



Multiply both sides of this equation times the inverse of the matrix A to find the solution

Inverse of a Non-square Matrix?

- The pseudoinverse (the Moore–Penrose inverse) is a generalized inverse defined for non-square matrices.

Existence of a Generalized Inverse

An inverse (left/right/both) to an MxN matrix **A** only exist when the rank is as large as possible, i.e., when rank(**A**) = min(M,N).

When **A** has full column rank, rank(**A**) = N:

there exists a ***left inverse*** $\mathbf{B} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T$ of size NxM such that $\mathbf{BA} = \mathbf{I}_{NxN}$

Ax = **b** has *at least one* solution **x** for each **b**.

When **A** has full row rank, rank(**A**) = M:

there exists a ***right inverse*** $\mathbf{C} = \mathbf{A}^T(\mathbf{AA}^T)^{-1}$ of size NxM such that $\mathbf{AC} = \mathbf{I}_{MxM}$

Ax = **b** has *at most one* solution **x** for each **b**.

Combine the two:

When **A** is square (M=N) and has full rank, rank(**A**) = M = N:

there exists a left inverse **B** and a right inverse **C** (that are equal)

Ax = **b** has only one solution **x** for each **b**.

Application, Overdetermined System of Equations

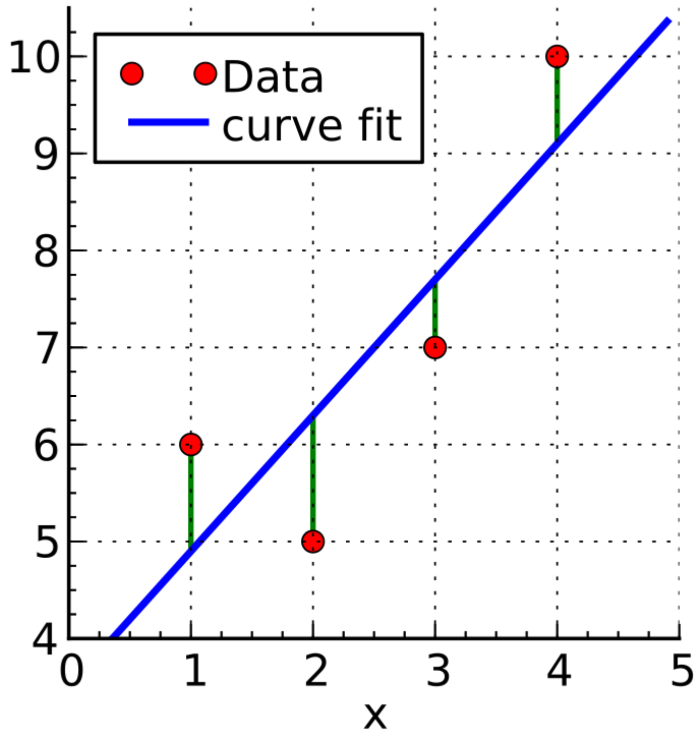
- Curve fitting example, four data points, four equations, two unknowns (the unknowns are the intercept and slope of the line)
- “Least-square solution”

$$\mathbf{y} = \begin{bmatrix} 6 \\ 5 \\ 7 \\ 10 \end{bmatrix}, \quad \mathbf{X} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \\ 1 & 3 \\ 1 & 4 \end{bmatrix}, \quad \beta = \begin{bmatrix} \beta_1 \\ \beta_2 \end{bmatrix}.$$

Intuitively,

$$\mathbf{y} = \mathbf{X}\beta \quad \Rightarrow \quad \mathbf{X}^\top \mathbf{y} = \mathbf{X}^\top \mathbf{X}\beta \quad \Rightarrow \quad \beta = (\mathbf{X}^\top \mathbf{X})^{-1}\mathbf{X}^\top \mathbf{y} = \begin{bmatrix} 3.5 \\ 1.4 \end{bmatrix}.$$

$$y = \beta_1 + \beta_2 x$$



https://en.wikipedia.org/wiki/Linear_least_squares

Diagonalization

Eigendecomposition, eigenvalues and eigenvectors.

Diagonalization using eigendecomposition

An n-square matrix A is similar to a diagonal matrix D if and only if A has n linearly independent eigenvectors. In this case, the diagonal elements of D are the corresponding eigenvalues and $D = P^{-1}AP$, where P is the matrix whose columns are the eigenvectors.

Diagonalization Suppose the n by n matrix A has n linearly independent eigenvectors $x_1, \ldots, x_n$. Put them into the columns of an ***eigenvector matrix*** X. Then $X^{-1}AX$ is the ***eigenvalue matrix*** Λ:

Eigenvector matrix X
Eigenvalue matrix Λ

$$X^{-1}AX = \Lambda = \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{bmatrix}. \qquad (1)$$

Diagonalizing a Matrix

1 The columns of $AX = X\Lambda$ are $A\boldsymbol{x}_k = \lambda_k \boldsymbol{x}_k$. The eigenvalue matrix Λ is diagonal.

2 $\boldsymbol{n}$ independent eigenvectors in X diagonalize A $\boxed{\boldsymbol{A = X\Lambda X^{-1}} \text{ and } \boldsymbol{\Lambda = X^{-1}AX}}$

3 The eigenvector matrix X also diagonalizes all powers A^k : $\boxed{\boldsymbol{A^k = X\Lambda^k X^{-1}}}$

4 Solve $\boldsymbol{u}_{k+1} = A\boldsymbol{u}_k$ by $\boldsymbol{u}_k = A^k \boldsymbol{u}_0 = X\Lambda^k X^{-1}\boldsymbol{u}_0 = \boxed{\boldsymbol{c_1(\lambda_1)^k x_1 + \cdots + c_n(\lambda_n)^k x_n}}$

5 **No equal eigenvalues** $\Rightarrow$ X is invertible and A can be diagonalized.

Equal eigenvalues $\Rightarrow$ A *might* have too few independent eigenvectors. Then X^{-1} fails.

6 Every matrix $C = B^{-1}AB$ has the **same eigenvalues** as A. These C's are "**similar**" to A.

Defective Matrices are not Diagonalizable

Not all *NxN* matrices are diagonalizable, since not all posesses *N* linearly independent eigenvectors.

Example: The matrix

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

Triangular with zeros on the diagonal:
$\lambda_1 = \lambda_2 = 0$
(algebraic multiplicity 2)

is "defective" in the sense that it is of size 2x2 but only has "one" eigenvector.

Do not match!

The non-zero solutions to

$$(\mathbf{A} - \lambda\mathbf{I})\mathbf{x} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}\mathbf{x} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

are all on the form

$$\mathbf{x} = \begin{bmatrix} c \\ 0 \end{bmatrix}$$

All eigenvectors are multiples of $[\ 1\ 0\ ]^T$.
(geometric multiplicity 1)

Applications of Diagonalization

- Powers of a square Matrix
- Principal component analysis
- Markov Chains (to be discussed later)
- Power Control
- Min-Max algorithms
- Signal subspace in Random signal processing
- Whitening (decoupling) of correlated signals/noise

Example, powers of a square Matrix

Example 1 This A is triangular so its eigenvalues are on the diagonal: $\lambda = 1$ and $\lambda = 6$.

Eigenvectors go into X $\begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

$$\underset{X^{-1}}{\begin{bmatrix} 1 & -1 \\ 0 & 1 \end{bmatrix}} \underset{A}{\begin{bmatrix} 1 & 5 \\ 0 & 6 \end{bmatrix}} \underset{X}{\begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}} = \underset{\Lambda}{\begin{bmatrix} 1 & 0 \\ 0 & 6 \end{bmatrix}}$$

In other words $A = X\Lambda X^{-1}$. Then watch $A^2 = X\Lambda X^{-1}X\Lambda X^{-1}$. So $\mathbf{A^2}$ **is** $\mathbf{X\Lambda^2X^{-1}}$.

A^2 has the same eigenvectors in X and squared eigenvalues in Λ^2.

Singular Value Decomposition (SVD)

Singular Value Decomposition

SVD

Would it be possible to replace (the restricted)

$$\mathbf{A} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^T$$

where **Q** is orthogonal and **Λ** is diagonal, with something (more general) like

$$\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$$

where **U** and **V** are still orthogonal (but not necessarily the same) and **Σ** is still diagonal?

This would give us essentially the same "orthogonal transform" property as we have used when we had *symmetric* or *Hermitian* matrices.

... so, does this work?

SVD

Singular Value Decomposition

Any M x N matrix **A** can be factored into

$$\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$$

The columns of **U** (M x M) are eigenvectors of $\mathbf{AA}^T$, and the columns of **V** (N x N) are eigenvectors of $\mathbf{A}^T\mathbf{A}$. The r singular values on the diagonal of $\mathbf{\Sigma}$ (M x N) are the square roots of the nonzero eigenvalues of both $\mathbf{AA}^T$ and $\mathbf{A}^T\mathbf{A}$.

This is a very powerful decomposition ... it works on everything and gives us the practical (orthogonal)(diagonal)(orthogonal) form that simplifies many problems!

SVD

About the structure:

$$
\begin{array}{cccc}
M \times N & M \times M & M \times N & N \times N \\
\mathbf{A} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T = & \begin{bmatrix} | & & | \\ \mathbf{u}_1 & \cdots & \mathbf{u}_m \\ | & & | \end{bmatrix} & \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_r \\ & & \end{bmatrix} & \begin{bmatrix} - & \mathbf{v}_1^T & - \\ & \vdots & \\ - & \mathbf{v}_n^T & - \end{bmatrix}
\end{array}
$$

Are often sorted $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_r$

$\mathbf{u}_1$ to $\mathbf{u}_r$ spans $C(\mathbf{A})$
$\mathbf{u}_{r+1}$ to $\mathbf{u}_M$ spans $N(\mathbf{A}^T)$

Implies that $r = \text{rank}(A)$

Since the singular values are the square-roots of the non-zero eigenvalues of both $\mathbf{AA}^T$ and $\mathbf{A}^T\mathbf{A}$, there can be at most $\min(m,n)$ of them, i.e., $r \leq \min(m,n)$.

$\mathbf{v}_1$ to $\mathbf{v}_r$ spans $C(\mathbf{A}^T)$
$\mathbf{v}_{r+1}$ to $\mathbf{v}_N$ spans $N(\mathbf{A})$

In certain applications, e.g. linear estimation with strong correlation, we have the property that $r \ll \min(m,n)$ and we can use this to simplify calculations.

Calculating the SVD

- The singular value decomposition of a matrix A can be computed using the following observations:
- The left-singular vectors of A are a set of orthonormal eigenvectors of AA^*
- The right-singular vectors of A are a set of orthonormal eigenvectors of A^*A.
- The non-zero singular values of A (found on the diagonal entries of Σ are the square roots of the non-zero eigenvalues of both A^*A and AA^*

Some Applications of SVD

- Rank Reduction
- Image Compression
- PCA

Principal Component Analysis (PCA)

- Dimensionality reduction method.
- Transform a large set of variables into a smaller one that still contains most of the information in the large set.
- Used in
  - Data compression:
    - Save data.
    - Speed up learning algorithm.
  - Data visualization (Reduce high dimension data to 3D or 2D).

Principal Component Analysis (PCA)

Steps (to be revisited after discussing the ‘**Covariance’** in the *Probability and Statistics for Machine Learning* Module)

1. Standardization.
2. Covariance matrix computation.
3. Eigenvectors and Eigenvalues of the covariance matrix.
4. Feature vector.

Principal Component Analysis (PCA): Standardization

Note:
The standard deviation is to be discussed in the probability course

- Given m observations and n number of features, X is the data matrix, and x_i the data from the i^{th} sample.
  - x_{ij} is the j^{th} reading from i^{th} sample
  - $\mu_j = \frac{1}{m}\sum_{i=1}^{m} x_{ij}$
  - $s_j^2 = \frac{1}{m-1}\sum_{i=1}^{m}\left(x_{ij} - \mu_j\right)^2$
  - $z_{ij} = \frac{x_{ij}-\mu_j}{s_j}$
- Z is the scaled data matrix, each feature has mean equal to zero and standard deviation equal to one.

$$X = \begin{bmatrix} - & x_1^T & - \\ - & x_2^T & - \\ - & \vdots & - \\ - & x_m^T & - \end{bmatrix}$$

$$x_i = \begin{bmatrix} x_{i1} \\ x_{i2} \\ \vdots \\ x_{in} \end{bmatrix}$$

$$Z = \begin{bmatrix} - & z_1 & - \\ - & z_2 & - \\ - & \vdots & - \\ - & z_m & - \end{bmatrix}$$

Principal Component Analysis (PCA): Covariance matrix computation

- Covariance matrix is a $n \times n$ symmetric matrix.
- Capture the relationship between the features of the input data set.
- Correlations between all the possible pairs of variables.
- Sometimes, features are highly correlated in such a way that they contain redundant information.

$$cov(a, b) = cov(b, a)$$

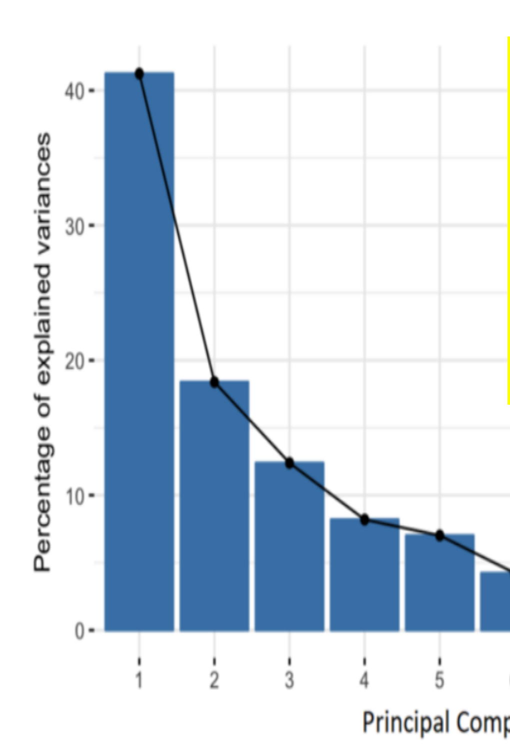
- Positive number: increase or decrease together. (Correlated)
- Negative number: One increase other decrease (Inversely correlated).

Note:
The covariance matrix is to be discussed in the probability course

$$C = Z^T Z = \begin{bmatrix} cov(z_{i1}, z_{i1}) & cov(z_{i1}, z_{i2}) & \cdots & cov(z_{i1}, z_{in}) \\ cov(z_{i2}, z_{i1}) & cov(z_{i2}, z_{i2}) & \cdots & cov(z_{i2}, z_{in}) \\ \vdots & \vdots & \ddots & \vdots \\ cov(z_{in}, z_{i1}) & \cdots & \cdots & cov(z_{in}, z_{in}) \end{bmatrix}$$

PCA: Eigenvectors and Eigenvalues of the covariance matrix.

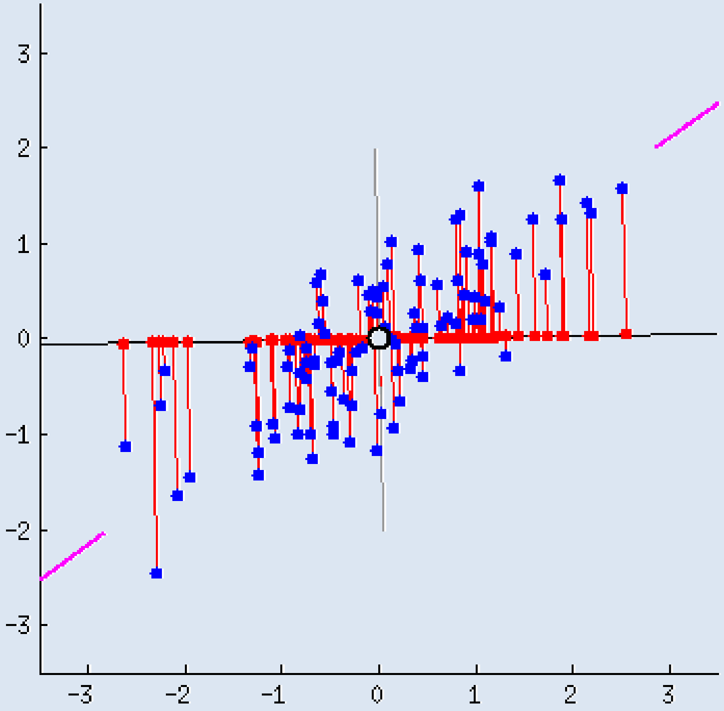
- Compute the Eigenvectors and Eigenvalues of the Covariance matrix to determine the principal component of the data.
- Principal components are new features that are constructed as linear combinations or mixtures of the initial features.
- The principal components are artificial features and are not necessarily easy to interpret.
- These combinations are done in such a way that the new features (i.e., principal components) are uncorrelated and most of the information within the initial variables is squeezed or compressed into the first components.
- To compute how much variance captured in the j^{th}component: $\frac{\lambda_j}{\sum_{k=1}^{n} \lambda_k} = \frac{\lambda_j}{trace(D)}$



Note:
The covariance matrix is to be discussed in the probability course

$$Cv_1 = \lambda_1 v_1, Cv_2 = \lambda_2 v_2, \dots, Cv_n = \lambda_n v_n$$
$$\lambda_1 > \lambda_2 > \dots > \lambda_n$$
$$D = \begin{bmatrix} \lambda_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \lambda_n \end{bmatrix}$$
$$CV = VD$$
$$T = ZV$$

Principal Component Analysis (PCA): Feature vector



- Choose whether to keep all these components or discard those of lesser significance (of low eigenvalues).
- Feature vector is a matrix that has as columns the eigenvectors of the components that we decide to keep.
- If we choose to keep only p eigenvectors (components) out of n, the final data set will have only p dimensions.

Principal Component Analysis (PCA): Example

$$X = \begin{bmatrix} 10 & 6 \\ 11 & 4 \\ 8 & 5 \\ 3 & 3 \\ 2 & 2.8 \\ 1 & 1 \end{bmatrix}$$

$$\mu = \begin{bmatrix} 5.833 & 3.633 \end{bmatrix}$$

$$Z = \begin{bmatrix} 4.166 & 2.366 \\ 5.166 & 0.366 \\ 2.166 & 1.366 \\ -2.833 & -0.633 \\ -3.833 & -0.833 \\ -4.833 & -2.633 \end{bmatrix}$$

$$C = \begin{bmatrix} 94.833 & 32.433 \\ 32.433 & 15.633 \end{bmatrix}$$

$$D = \begin{bmatrix} 106.42 & 0 \\ 0 & 4.0466 \end{bmatrix}$$

$$V = \begin{bmatrix} 0.941 & -0.336 \\ 0.336 & 0.941 \end{bmatrix}$$

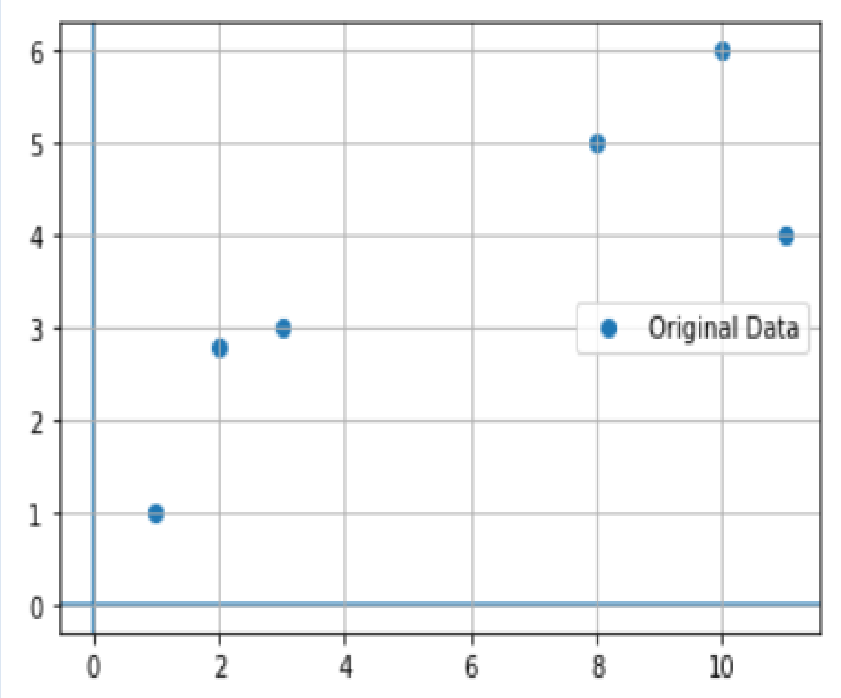
$$T = ZV$$

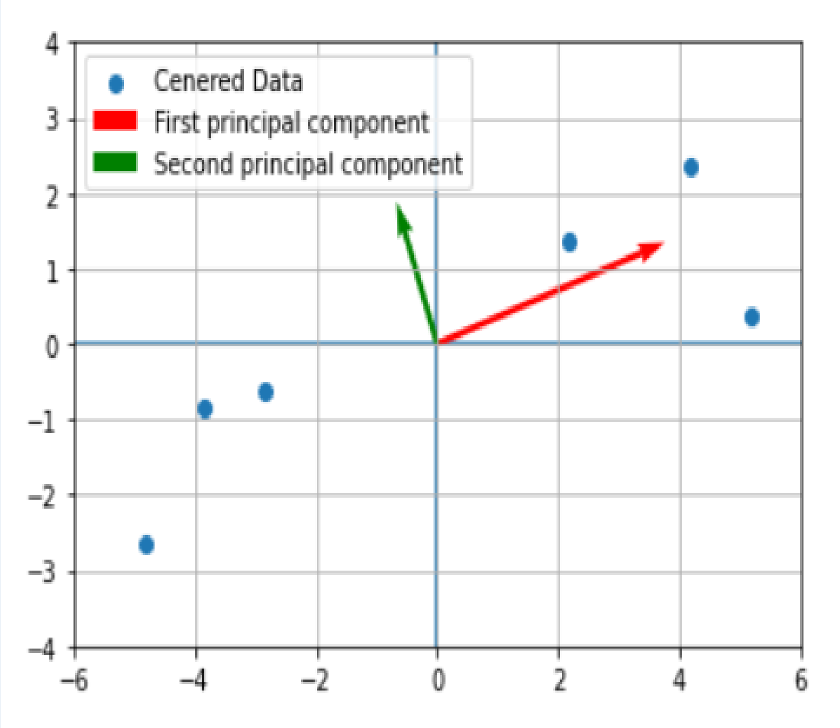
Percentage of variance capture in each component:

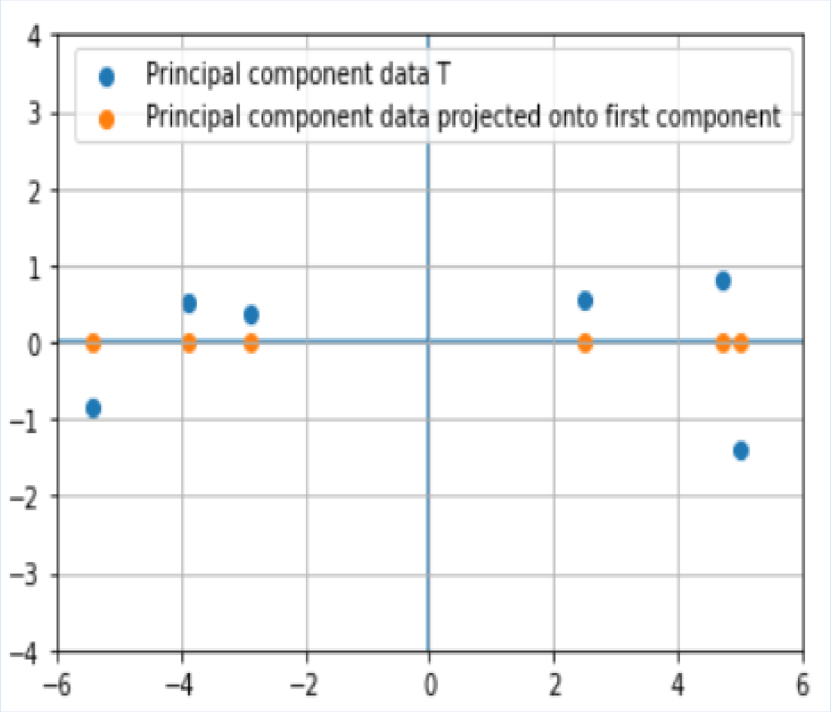
First : $\frac{106.42}{106.42+4.067} = 0.96 \Rightarrow 96\%$

Second: $\frac{4.067}{106.42+4.067} = 0.04 \Rightarrow 4\%$

Principal Component Analysis (PCA): Example







Worked examples, tutorials

- See these:
- https://sebastianraschka.com/Articles/2015_pca_in_3_steps.html
- https://www.youtube.com/watch?v=FgakZw6K1QQ
- https://www.youtube.com/watch?v=g-Hb26agBFg

Principal Component Analysis (PCA): Comments

- Principal components are used to reduce large-dimensional data sets to data sets with a few dimensions that still retain most of the information in the original data.
- Each principal component is a linear combination of the scaled variables.
- Any two principal components are uncorrelated.
- The first few principal components account for a large percentage of the total variance.
- The principal components are artificial variables and are not necessarily easy to interpret.

Principal Component Analysis (PCA): References

- Lawrence Spence, Arnold Insel, Stephen Friedberg - Elementary Linear Algebra - A Matrix Approach- Pearson (2013).
- https://builtin.com/data-science/step-step-explanation-principal-component-analysis.
- https://www.youtube.com/watch?v=fkf4IBRSeEc.

Questions?

Resources

- Books and Articles
  - “Linear Algebra Explained in four Pages”: https://courses.engr.illinois.edu/ece498rc3/fa2016/material/linearAlgebra_4pgs.pdf
  - G. Strang, “Linear Algebra and Its Applications“ Fourth edition, Brooks/Cole
  - Odd K. Moon and Wynn C. Stirling, “Mathematical Methods and Algorithms for Signal Processing”, Prentice Hall 1999
  - Lathi, B. P. “Modern Digital and Analog Communication Systems,” Oxford University Press. 4th / 5th Editions
- Online Courses and Slides
  - Linear Algebra, MIT Open Coursware, Prof. Gilbert Strang, http://ocw.mit.edu/courses/mathematics/18-06-linear-algebra-spring-2010/index.htm *
  - Linear Algebra for Wireless Communication, Ove Edfors, Lund University: http://www.eit.lth.se/index.php?ciuid=384&coursepage=1300&L=1
  - System of Linear Equations (Reference Card) | Brilliant Math & Science Wiki

Thank You