APS1070

Foundations of Data Analytics and Machine Learning

Summer 2020

Wed June 17 / Week 6:

- *Linear Algebra*
- *Analytical Geometry*
- *Example questions*

Jason Riordon, PhD

News

- Project extensions – 3 days
- Midterm quiz next week – Tuesday, June 23, noon to Wednesday, June 24, noon (24 hrs)
- Posted old evaluations (from Fall 2019)

Slide Attribution

These slides contain materials from various sources. Special thanks to Marc Deisenroth.

Linear Algebra
[Chapter 2 – MML book]

Systems of Linear Equations

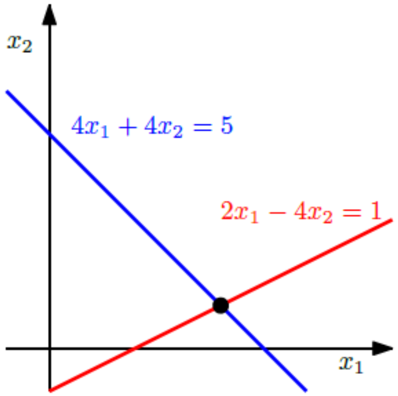
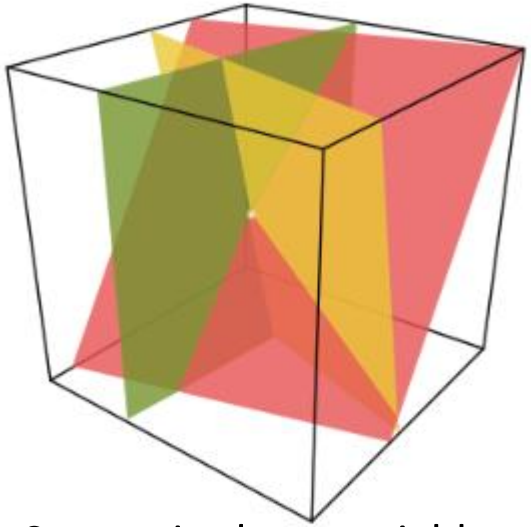
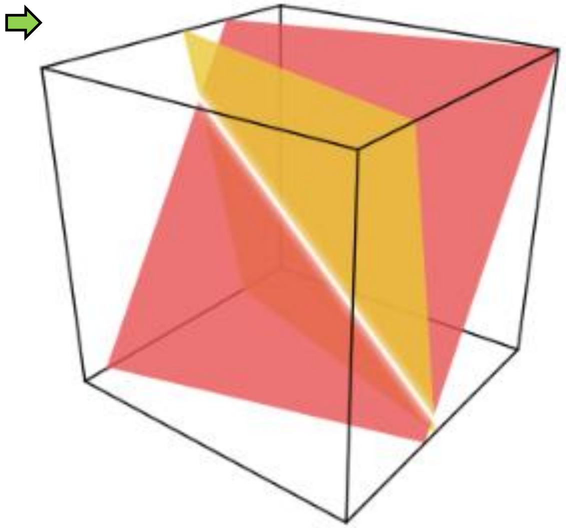


Figure 2.3 The solution space of a system of two linear equations with two variables can be geometrically interpreted as the intersection of two lines. Every linear equation represents a line.



System in three variables – solution is at intersection



With 2 equations and 3 variables – solution is typically a line

Matrices

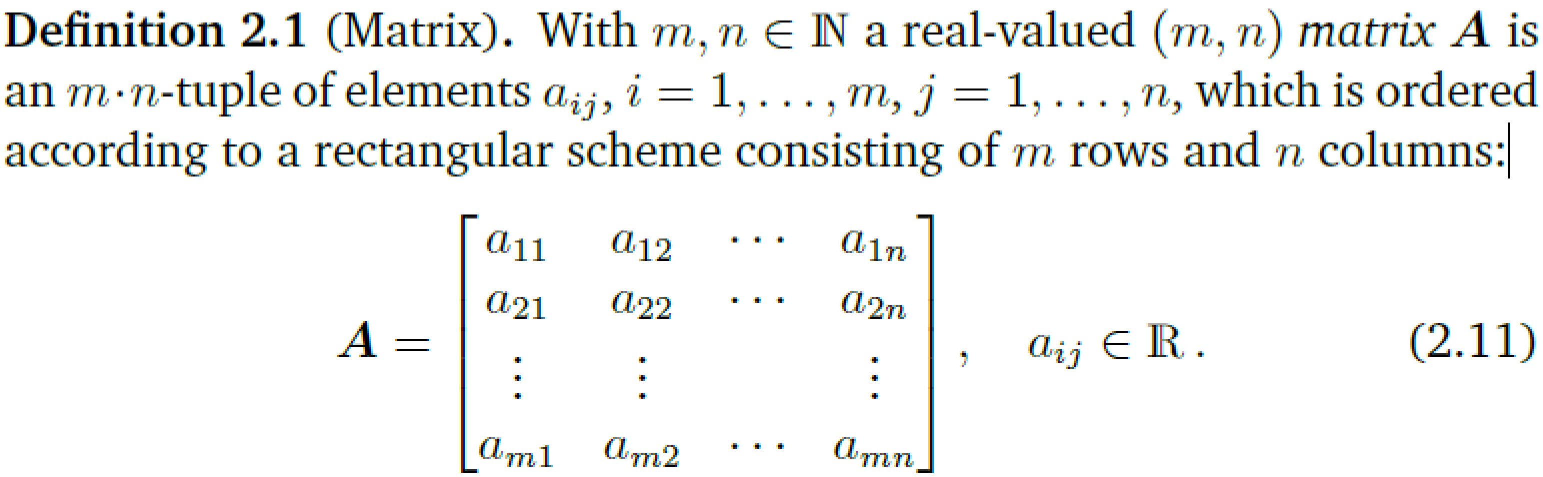
- To solve systems of linear equations more systematically
- This compact notation collects coefficients into vectors, and vectors into matrices.

$$x_1 \begin{bmatrix} a_{11} \\ \vdots \\ a_{m1} \end{bmatrix} + x_2 \begin{bmatrix} a_{12} \\ \vdots \\ a_{m2} \end{bmatrix} + \cdots + x_n \begin{bmatrix} a_{1n} \\ \vdots \\ a_{mn} \end{bmatrix} = \begin{bmatrix} b_1 \\ \vdots \\ b_m \end{bmatrix} \tag{2.9}$$

$$\iff \begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & & \vdots \\ a_{m1} & \cdots & a_{mn} \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ \vdots \\ b_m \end{bmatrix} . \tag{2.10}$$

Matrices

- Matrices can compactly represent systems of linear equations, and also linear functions (mappings)

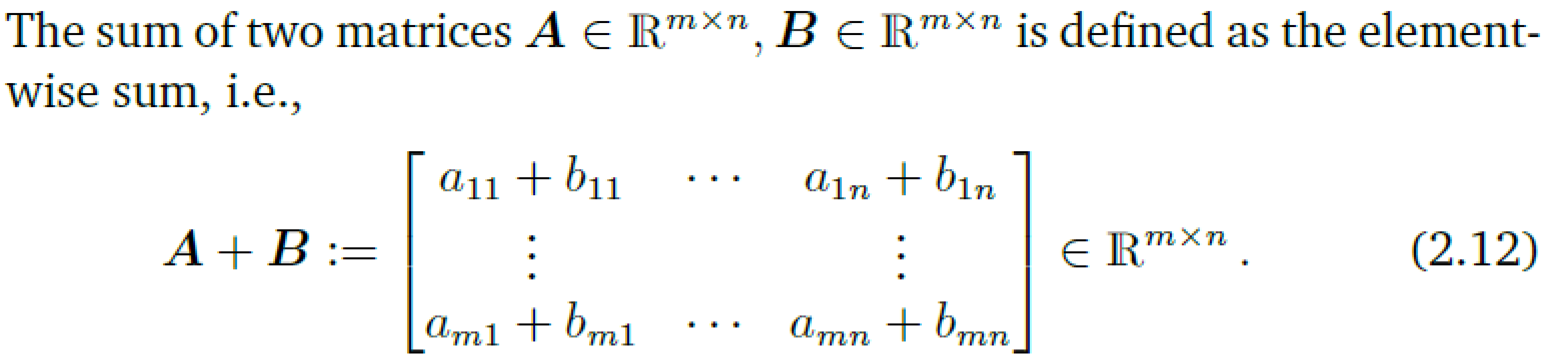


Definition 2.1 (Matrix). With $m, n \in \mathbb{N}$ a real-valued (m, n) *matrix* $\boldsymbol{A}$ is an $m \cdot n$-tuple of elements a_{ij}, $i = 1, \ldots, m$, $j = 1, \ldots, n$, which is ordered according to a rectangular scheme consisting of m rows and n columns:

$$\boldsymbol{A} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}, \quad a_{ij} \in \mathbb{R}\,. \tag{2.11}$$

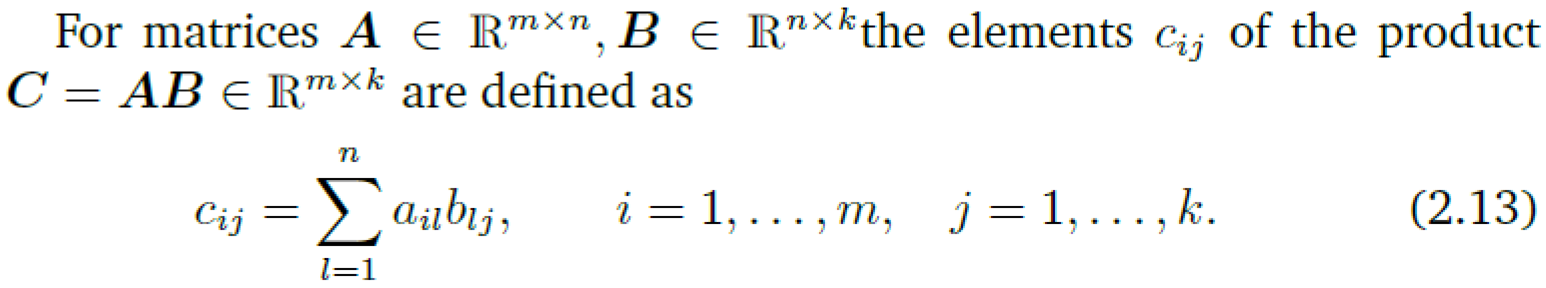
- By convention (1; *n*)-matrices are called ***rows*** and (*m*; 1)-matrices are called ***columns***. These special matrices are also called *row/column vectors*.

Matrices



The sum of two matrices $\boldsymbol{A} \in \mathbb{R}^{m\times n}, \boldsymbol{B} \in \mathbb{R}^{m\times n}$ is defined as the element-wise sum, i.e.,

$$\boldsymbol{A}+\boldsymbol{B} := \begin{bmatrix} a_{11}+b_{11} & \cdots & a_{1n}+b_{1n} \\ \vdots & & \vdots \\ a_{m1}+b_{m1} & \cdots & a_{mn}+b_{mn} \end{bmatrix} \in \mathbb{R}^{m\times n}\,. \tag{2.12}$$

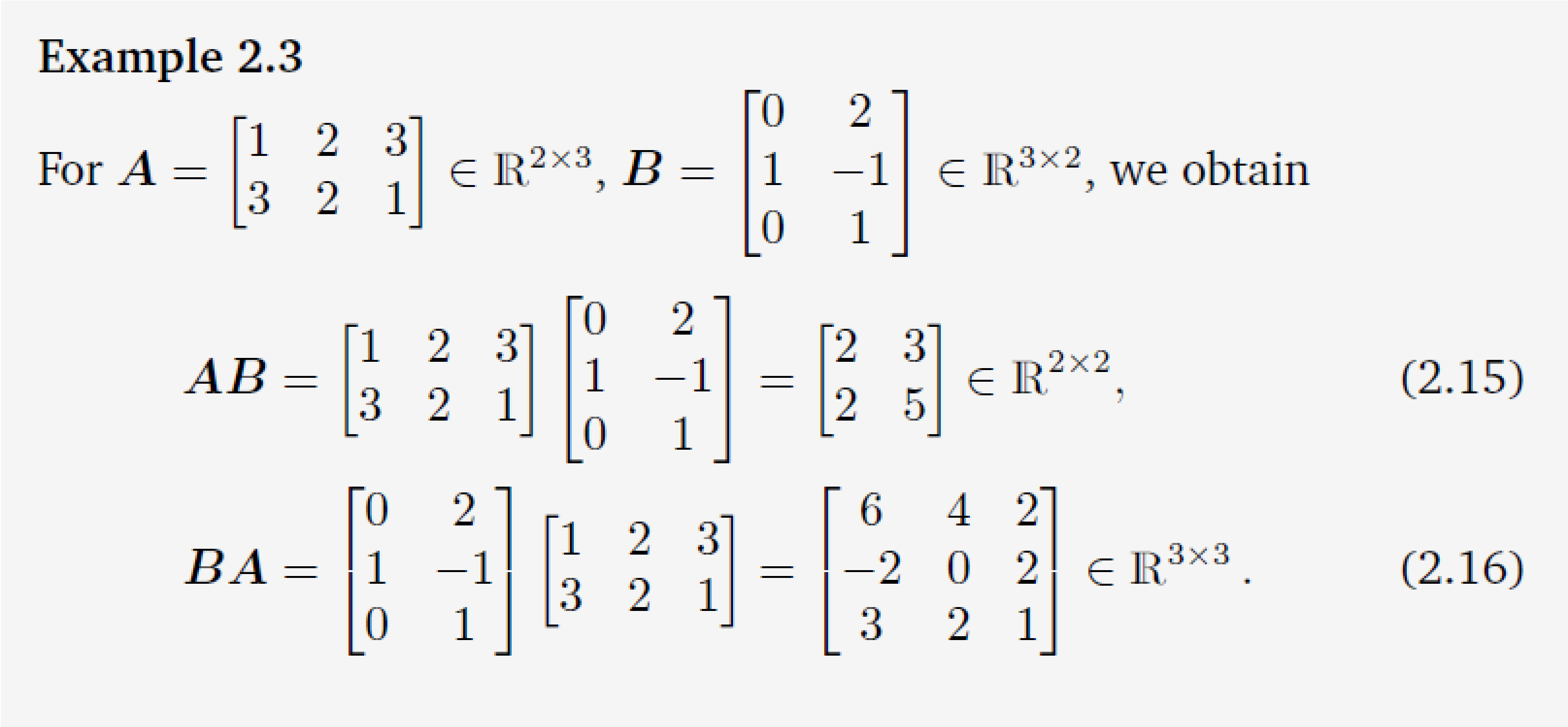


For matrices $\boldsymbol{A} \in \mathbb{R}^{m\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times k}$the elements c_{ij} of the product $\boldsymbol{C} = \boldsymbol{AB} \in \mathbb{R}^{m\times k}$ are defined as

$$c_{ij} = \sum_{l=1}^{n} a_{il}b_{lj}, \qquad i = 1,\ldots,m, \quad j = 1,\ldots,k. \tag{2.13}$$

[m x n] * [n x k] = [m x k]

Matrices



Example 2.3

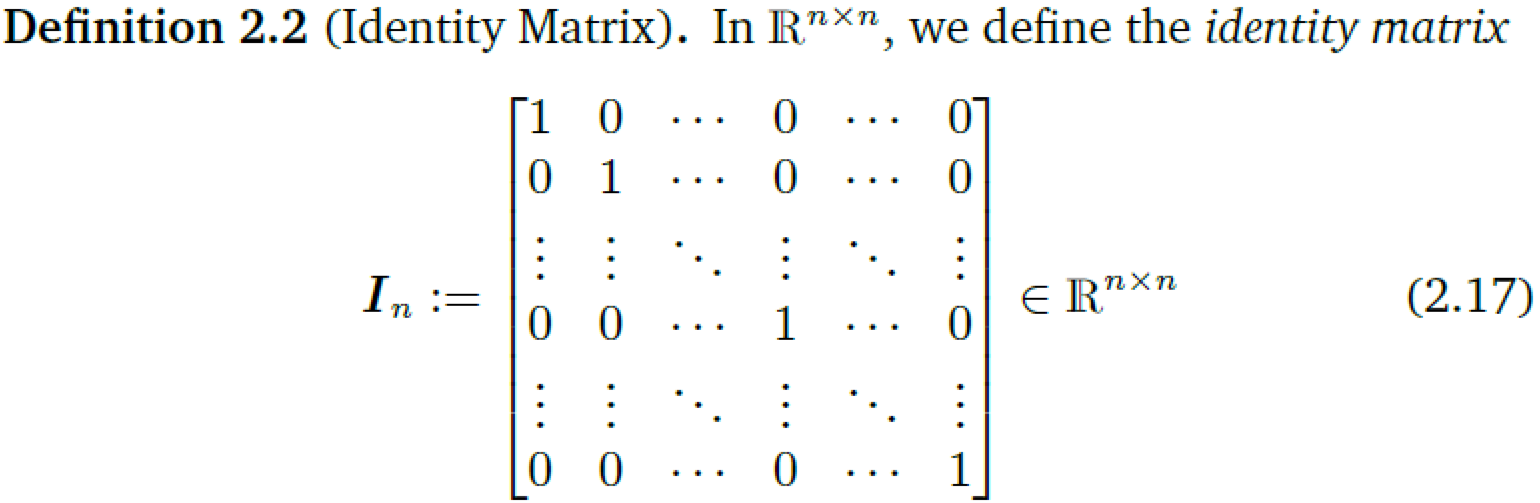
For $\boldsymbol{A} = \begin{bmatrix} 1 & 2 & 3 \\ 3 & 2 & 1 \end{bmatrix} \in \mathbb{R}^{2\times 3}$, $\boldsymbol{B} = \begin{bmatrix} 0 & 2 \\ 1 & -1 \\ 0 & 1 \end{bmatrix} \in \mathbb{R}^{3\times 2}$, we obtain

$$\boldsymbol{AB} = \begin{bmatrix} 1 & 2 & 3 \\ 3 & 2 & 1 \end{bmatrix} \begin{bmatrix} 0 & 2 \\ 1 & -1 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 2 & 3 \\ 2 & 5 \end{bmatrix} \in \mathbb{R}^{2\times 2}, \tag{2.15}$$

$$\boldsymbol{BA} = \begin{bmatrix} 0 & 2 \\ 1 & -1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 2 & 3 \\ 3 & 2 & 1 \end{bmatrix} = \begin{bmatrix} 6 & 4 & 2 \\ -2 & 0 & 2 \\ 3 & 2 & 1 \end{bmatrix} \in \mathbb{R}^{3\times 3}. \tag{2.16}$$

Not commutative! $\boldsymbol{AB} \neq \boldsymbol{BA}$

Matrices

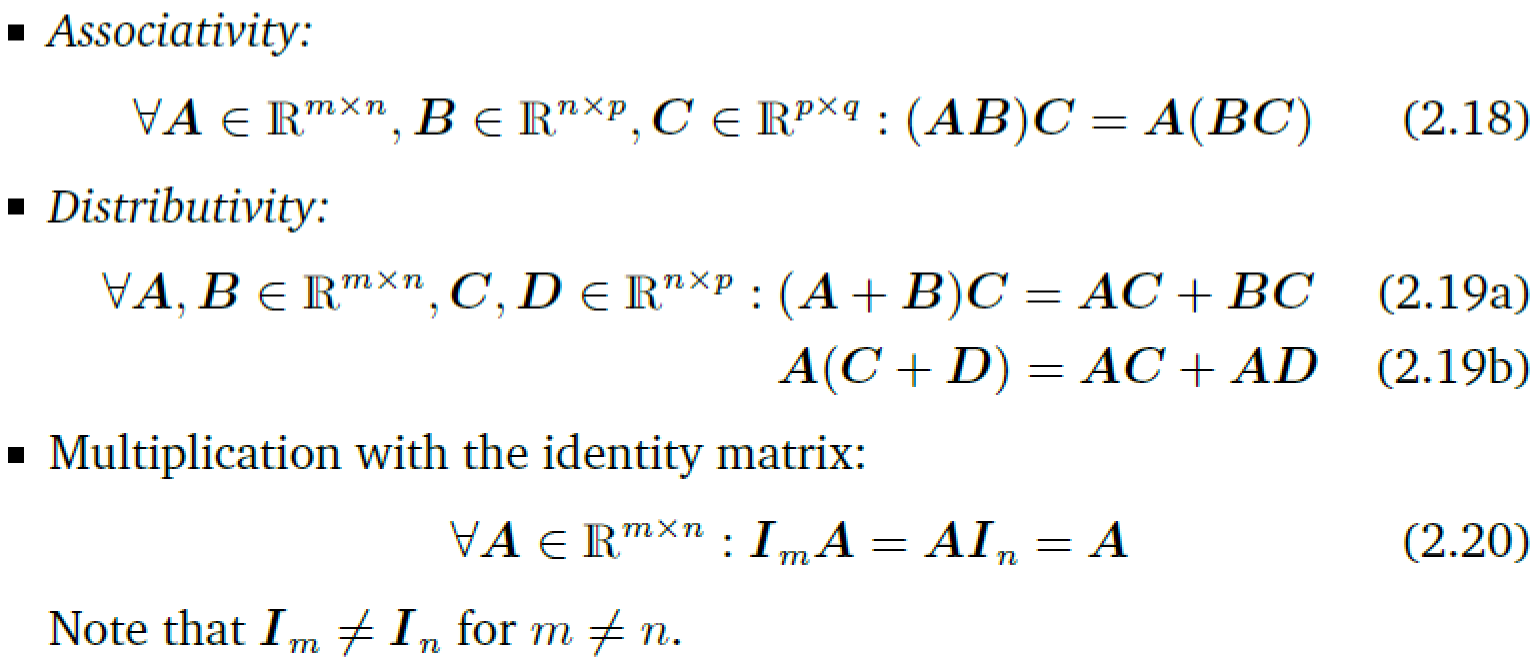


Definition 2.2 (Identity Matrix)**.** In $\mathbb{R}^{n\times n}$, we define the *identity matrix*

$$\boldsymbol{I}_n := \begin{bmatrix} 1 & 0 & \cdots & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & \cdots & 1 \end{bmatrix} \in \mathbb{R}^{n\times n} \tag{2.17}$$

Matrices

A few properties…



- *Associativity:*

$$\forall \boldsymbol{A} \in \mathbb{R}^{m\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times p}, \boldsymbol{C} \in \mathbb{R}^{p\times q} : (\boldsymbol{AB})\boldsymbol{C} = \boldsymbol{A}(\boldsymbol{BC}) \qquad (2.18)$$

- *Distributivity:*

$$\forall \boldsymbol{A}, \boldsymbol{B} \in \mathbb{R}^{m\times n}, \boldsymbol{C}, \boldsymbol{D} \in \mathbb{R}^{n\times p} : (\boldsymbol{A}+\boldsymbol{B})\boldsymbol{C} = \boldsymbol{AC} + \boldsymbol{BC} \qquad (2.19a)$$

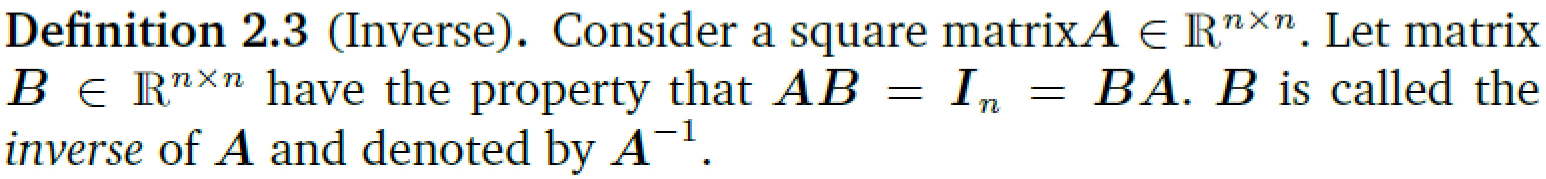
$$\boldsymbol{A}(\boldsymbol{C}+\boldsymbol{D}) = \boldsymbol{AC} + \boldsymbol{AD} \qquad (2.19b)$$

- Multiplication with the identity matrix:

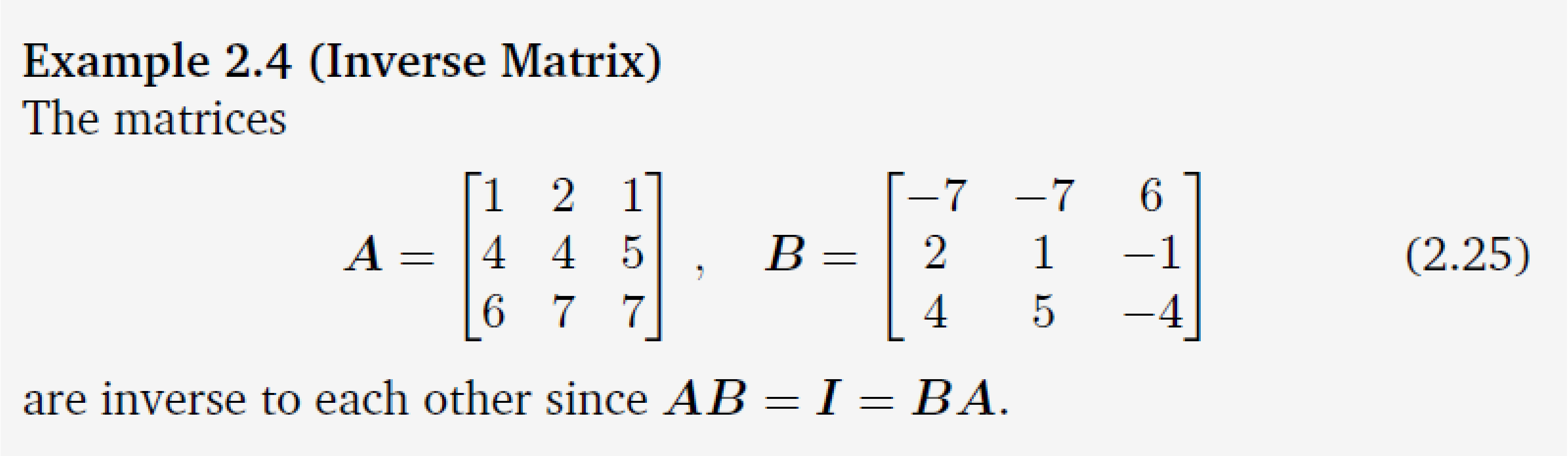
$$\forall \boldsymbol{A} \in \mathbb{R}^{m\times n} : \boldsymbol{I}_m\boldsymbol{A} = \boldsymbol{A}\boldsymbol{I}_n = \boldsymbol{A} \qquad (2.20)$$

Note that $\boldsymbol{I}_m \neq \boldsymbol{I}_n$ for $m \neq n$.

Matrices



Definition 2.3 (Inverse). Consider a square matrix$\boldsymbol{A} \in \mathbb{R}^{n \times n}$. Let matrix $\boldsymbol{B} \in \mathbb{R}^{n \times n}$ have the property that $\boldsymbol{AB} = \boldsymbol{I}_n = \boldsymbol{BA}$. $\boldsymbol{B}$ is called the *inverse* of $\boldsymbol{A}$ and denoted by $\boldsymbol{A}^{-1}$.



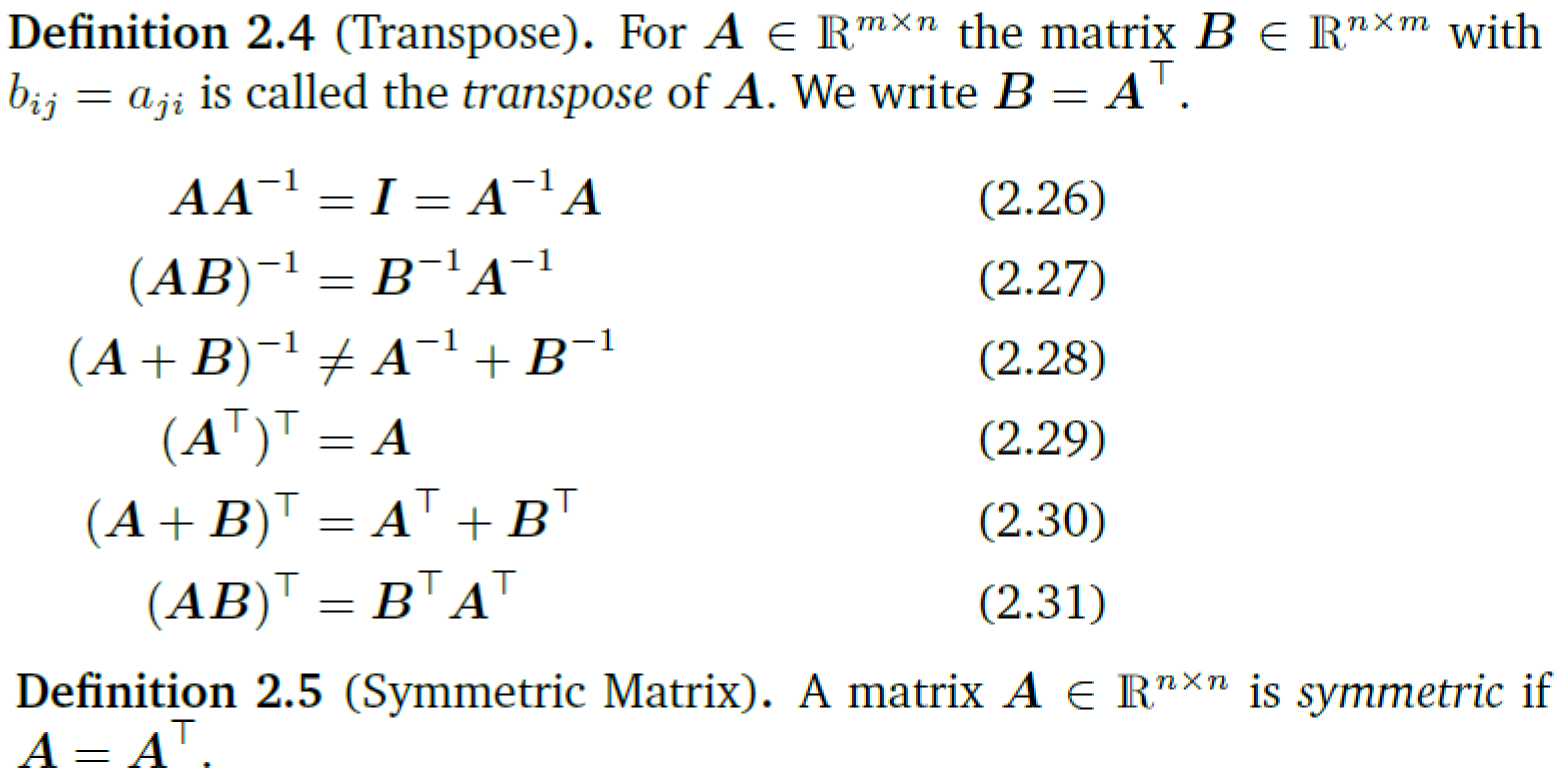
Example 2.4 (Inverse Matrix)
The matrices

$$\boldsymbol{A} = \begin{bmatrix} 1 & 2 & 1 \\ 4 & 4 & 5 \\ 6 & 7 & 7 \end{bmatrix}, \quad \boldsymbol{B} = \begin{bmatrix} -7 & -7 & 6 \\ 2 & 1 & -1 \\ 4 & 5 & -4 \end{bmatrix} \tag{2.25}$$

are inverse to each other since $\boldsymbol{AB} = \boldsymbol{I} = \boldsymbol{BA}$.

We'll look at how to calculate the inverse later

Matrices



Definition 2.4 (Transpose)**.** For $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ the matrix $\boldsymbol{B} \in \mathbb{R}^{n \times m}$ with $b_{ij} = a_{ji}$ is called the *transpose* of $\boldsymbol{A}$. We write $\boldsymbol{B} = \boldsymbol{A}^\top$.

$$\boldsymbol{A}\boldsymbol{A}^{-1} = \boldsymbol{I} = \boldsymbol{A}^{-1}\boldsymbol{A} \tag{2.26}$$

$$(\boldsymbol{A}\boldsymbol{B})^{-1} = \boldsymbol{B}^{-1}\boldsymbol{A}^{-1} \tag{2.27}$$

$$(\boldsymbol{A} + \boldsymbol{B})^{-1} \neq \boldsymbol{A}^{-1} + \boldsymbol{B}^{-1} \tag{2.28}$$

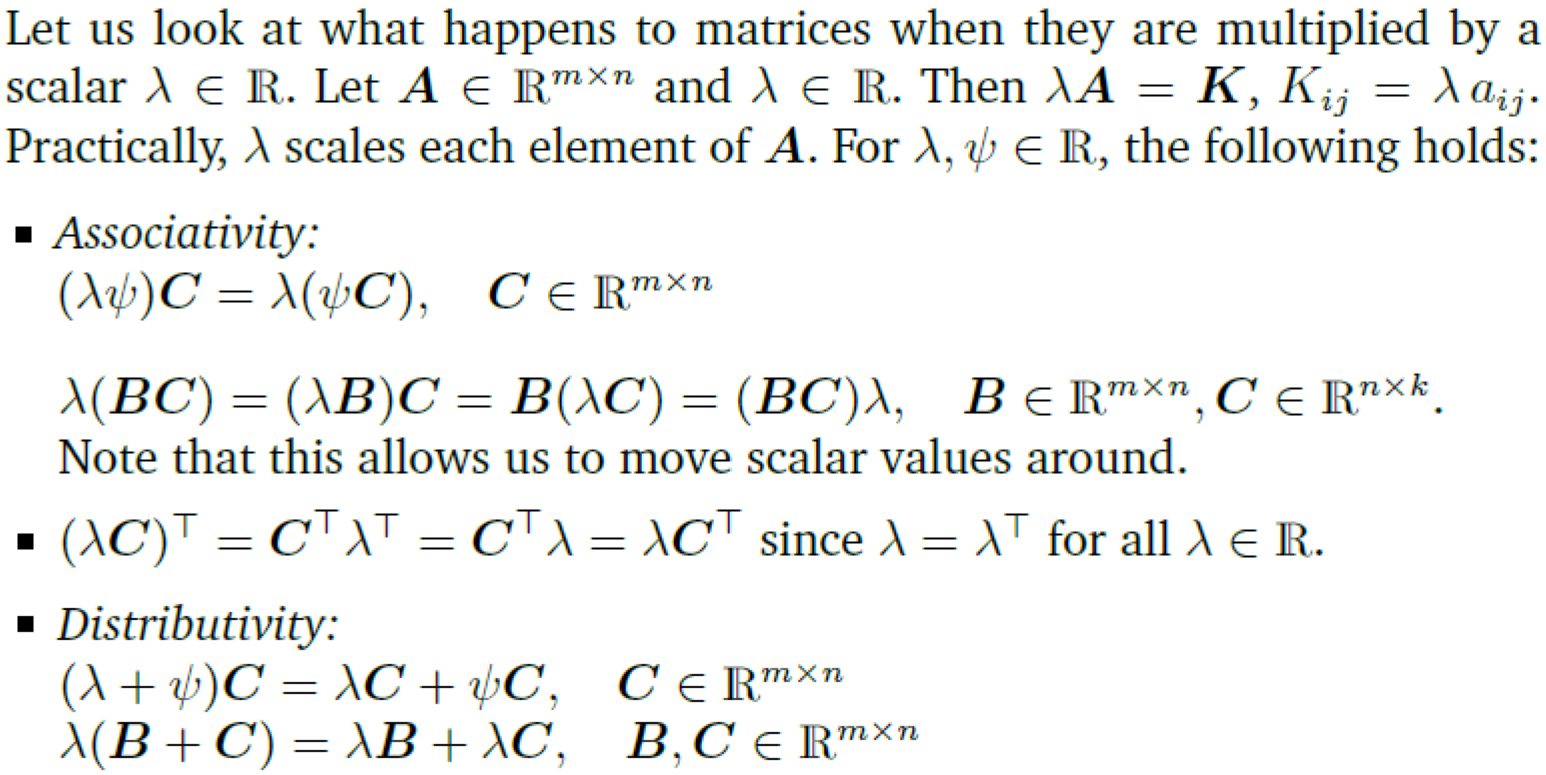
$$(\boldsymbol{A}^\top)^\top = \boldsymbol{A} \tag{2.29}$$

$$(\boldsymbol{A} + \boldsymbol{B})^\top = \boldsymbol{A}^\top + \boldsymbol{B}^\top \tag{2.30}$$

$$(\boldsymbol{A}\boldsymbol{B})^\top = \boldsymbol{B}^\top\boldsymbol{A}^\top \tag{2.31}$$

Definition 2.5 (Symmetric Matrix)**.** A matrix $\boldsymbol{A} \in \mathbb{R}^{n \times n}$ is *symmetric* if $\boldsymbol{A} = \boldsymbol{A}^\top$.

Matrices

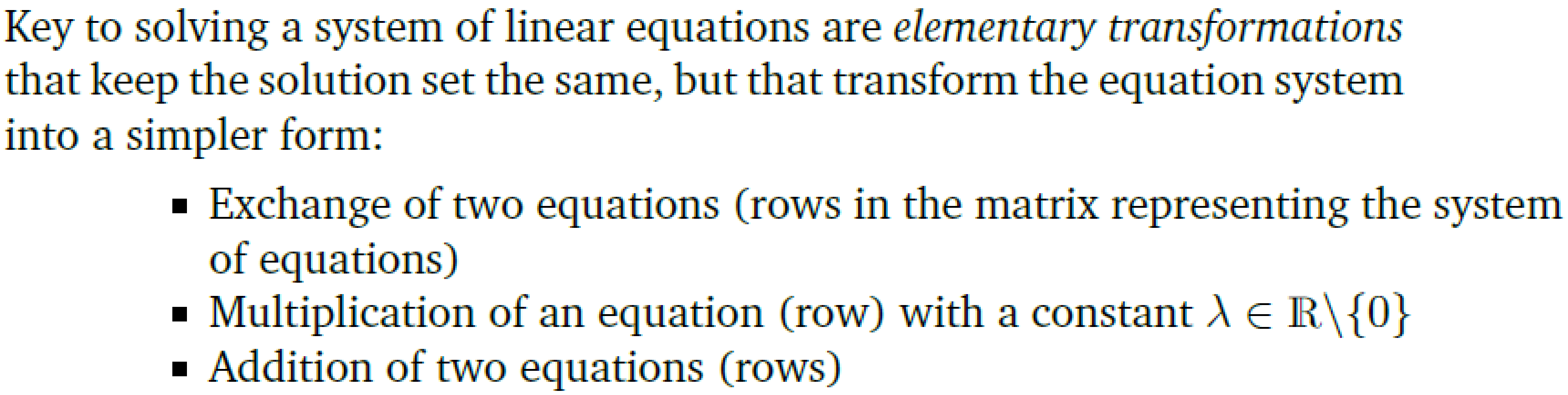


Let us look at what happens to matrices when they are multiplied by a scalar $\lambda \in \mathbb{R}$. Let $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ and $\lambda \in \mathbb{R}$. Then $\lambda\boldsymbol{A} = \boldsymbol{K}$, $K_{ij} = \lambda\, a_{ij}$. Practically, λ scales each element of $\boldsymbol{A}$. For $\lambda, \psi \in \mathbb{R}$, the following holds:

- *Associativity:*
  $(\lambda\psi)\boldsymbol{C} = \lambda(\psi\boldsymbol{C}), \quad \boldsymbol{C} \in \mathbb{R}^{m\times n}$

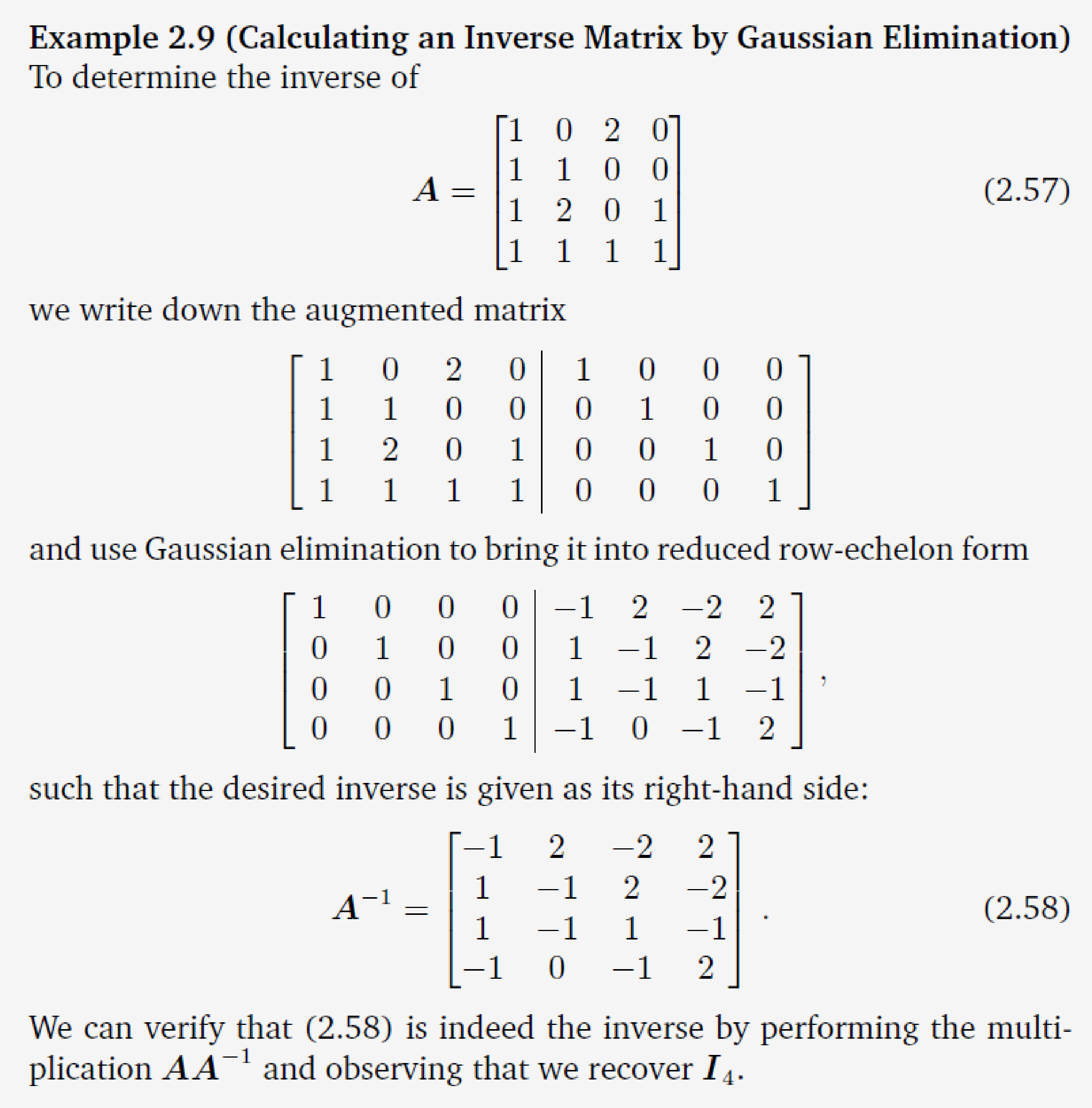
  $\lambda(\boldsymbol{BC}) = (\lambda\boldsymbol{B})\boldsymbol{C} = \boldsymbol{B}(\lambda\boldsymbol{C}) = (\boldsymbol{BC})\lambda, \quad \boldsymbol{B} \in \mathbb{R}^{m\times n}, \boldsymbol{C} \in \mathbb{R}^{n\times k}.$
  Note that this allows us to move scalar values around.
- $(\lambda\boldsymbol{C})^\top = \boldsymbol{C}^\top\lambda^\top = \boldsymbol{C}^\top\lambda = \lambda\boldsymbol{C}^\top$ since $\lambda = \lambda^\top$ for all $\lambda \in \mathbb{R}$.
- *Distributivity:*
  $(\lambda + \psi)\boldsymbol{C} = \lambda\boldsymbol{C} + \psi\boldsymbol{C}, \quad \boldsymbol{C} \in \mathbb{R}^{m\times n}$
  $\lambda(\boldsymbol{B} + \boldsymbol{C}) = \lambda\boldsymbol{B} + \lambda\boldsymbol{C}, \quad \boldsymbol{B}, \boldsymbol{C} \in \mathbb{R}^{m\times n}$

Solving Systems of Linear Equations



Key to solving a system of linear equations are *elementary transformations* that keep the solution set the same, but that transform the equation system into a simpler form:

- Exchange of two equations (rows in the matrix representing the system of equations)
- Multiplication of an equation (row) with a constant $\lambda \in \mathbb{R} \backslash \{0\}$
- Addition of two equations (rows)



Example 2.9 (Calculating an Inverse Matrix by Gaussian Elimination)
To determine the inverse of

$$\boldsymbol{A} = \begin{bmatrix} 1 & 0 & 2 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 2 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix} \tag{2.57}$$

we write down the augmented matrix

$$\left[\begin{array}{cccc|cccc} 1 & 0 & 2 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 2 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \end{array}\right]$$

and use Gaussian elimination to bring it into reduced row-echelon form

$$\left[\begin{array}{cccc|cccc} 1 & 0 & 0 & 0 & -1 & 2 & -2 & 2 \\ 0 & 1 & 0 & 0 & 1 & -1 & 2 & -2 \\ 0 & 0 & 1 & 0 & 1 & -1 & 1 & -1 \\ 0 & 0 & 0 & 1 & -1 & 0 & -1 & 2 \end{array}\right],$$

such that the desired inverse is given as its right-hand side:

$$\boldsymbol{A}^{-1} = \begin{bmatrix} -1 & 2 & -2 & 2 \\ 1 & -1 & 2 & -2 \\ 1 & -1 & 1 & -1 \\ -1 & 0 & -1 & 2 \end{bmatrix}. \tag{2.58}$$

We can verify that (2.58) is indeed the inverse by performing the multiplication $\boldsymbol{A}\boldsymbol{A}^{-1}$ and observing that we recover $\boldsymbol{I}_4$.

Linear independence

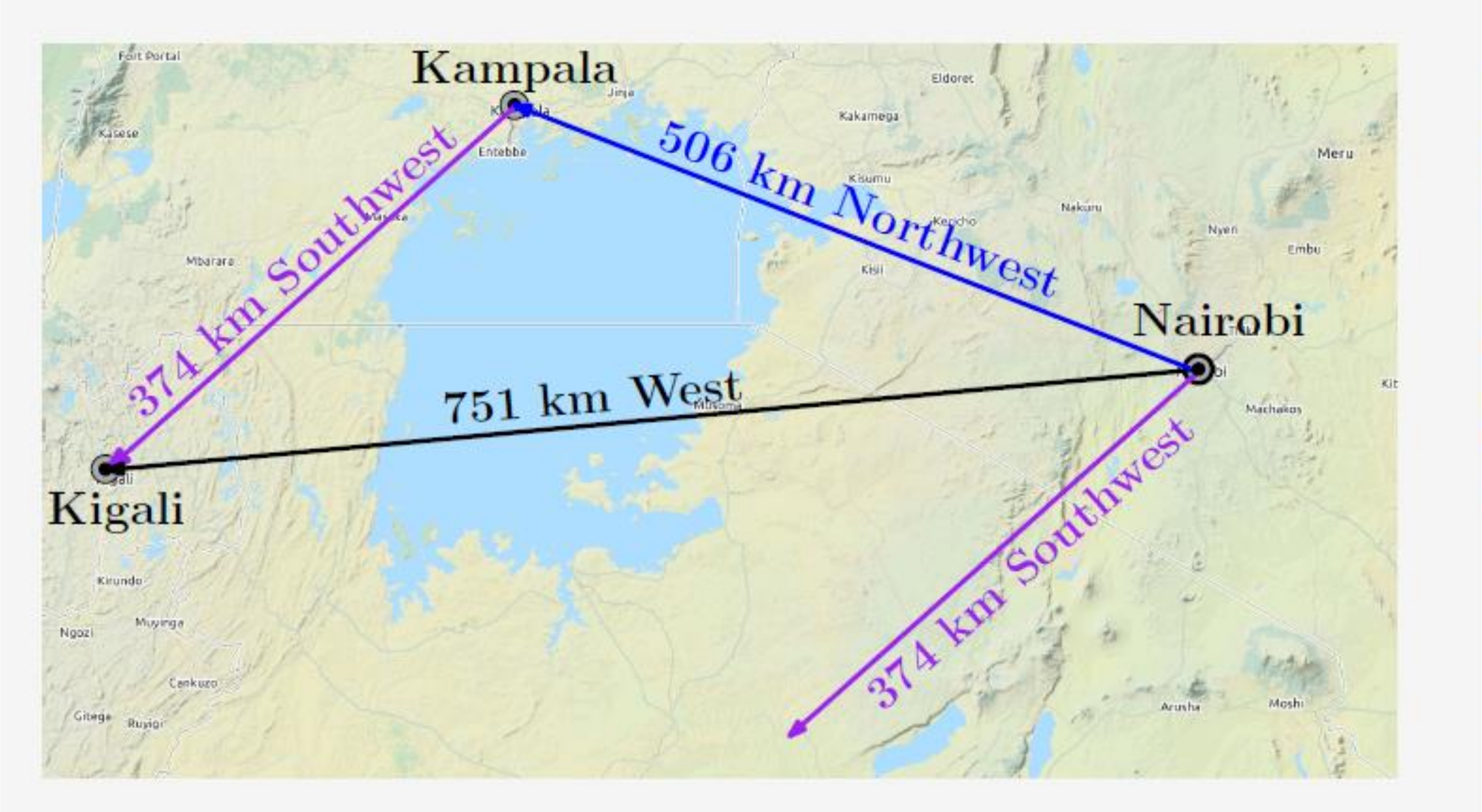
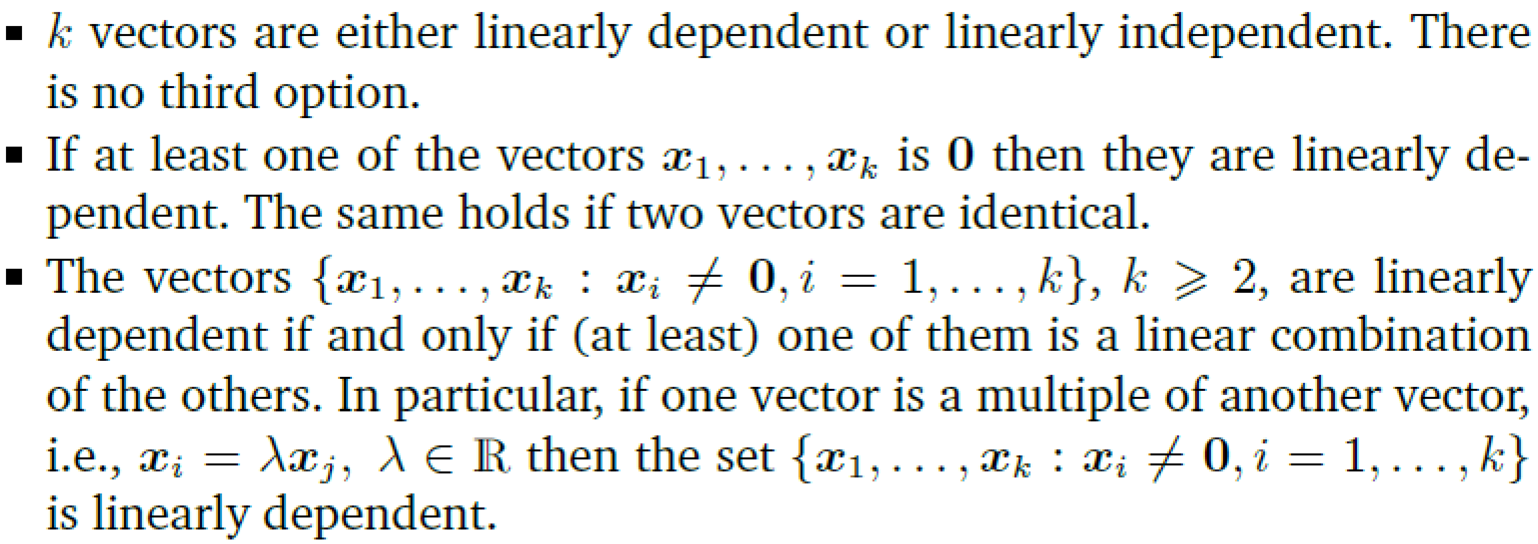


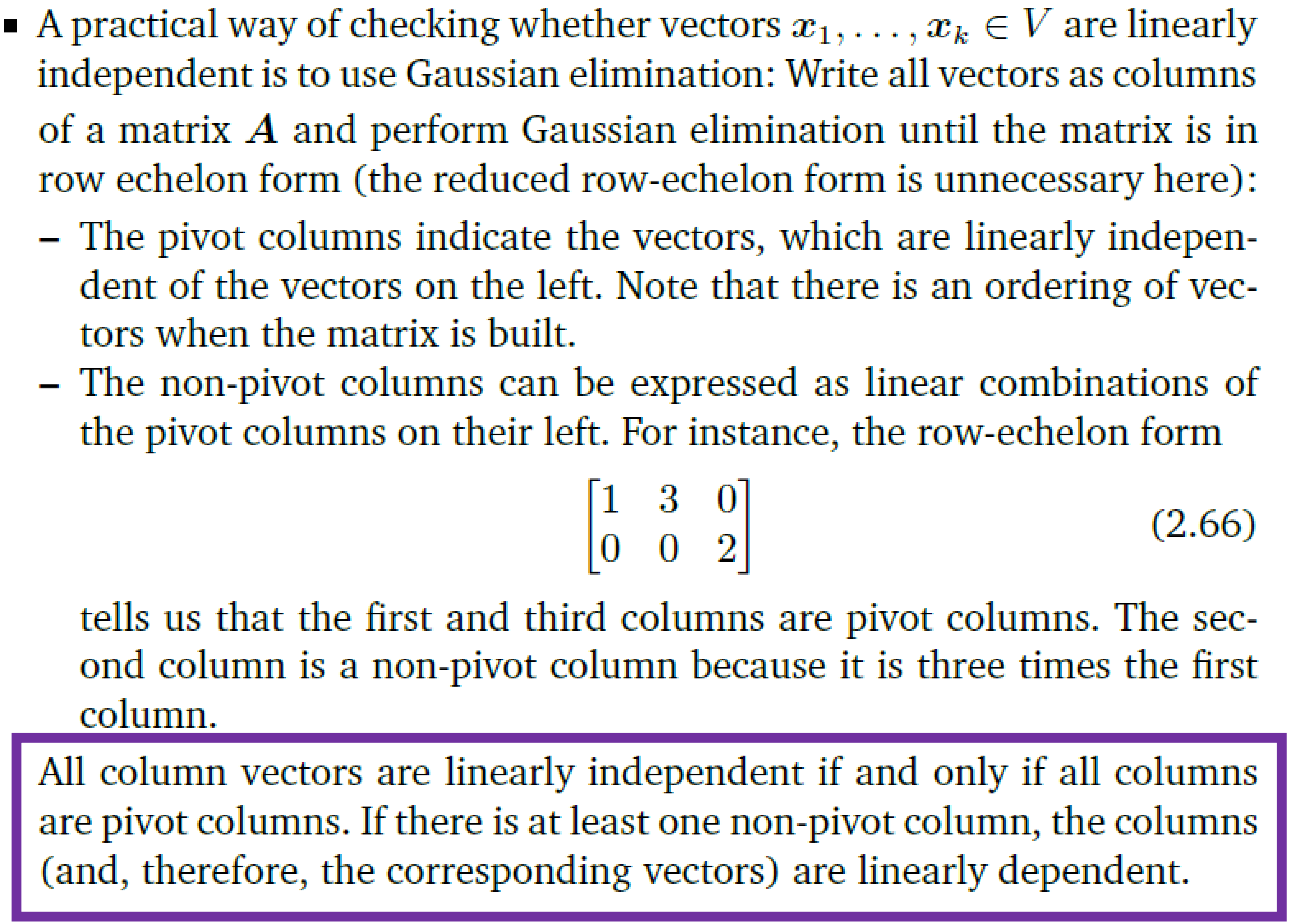
Figure 2.7 Geographic example (with crude approximations to cardinal directions) of linearly dependent vectors in a two-dimensional space (plane).

Linear independence



- k vectors are either linearly dependent or linearly independent. There is no third option.
- If at least one of the vectors $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k$ is $\mathbf{0}$ then they are linearly dependent. The same holds if two vectors are identical.
- The vectors $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k : \boldsymbol{x}_i \neq \mathbf{0}, i = 1, \ldots, k\}$, $k \geqslant 2$, are linearly dependent if and only if (at least) one of them is a linear combination of the others. In particular, if one vector is a multiple of another vector, i.e., $\boldsymbol{x}_i = \lambda \boldsymbol{x}_j$, $\lambda \in \mathbb{R}$ then the set $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k : \boldsymbol{x}_i \neq \mathbf{0}, i = 1, \ldots, k\}$ is linearly dependent.

Linear independence



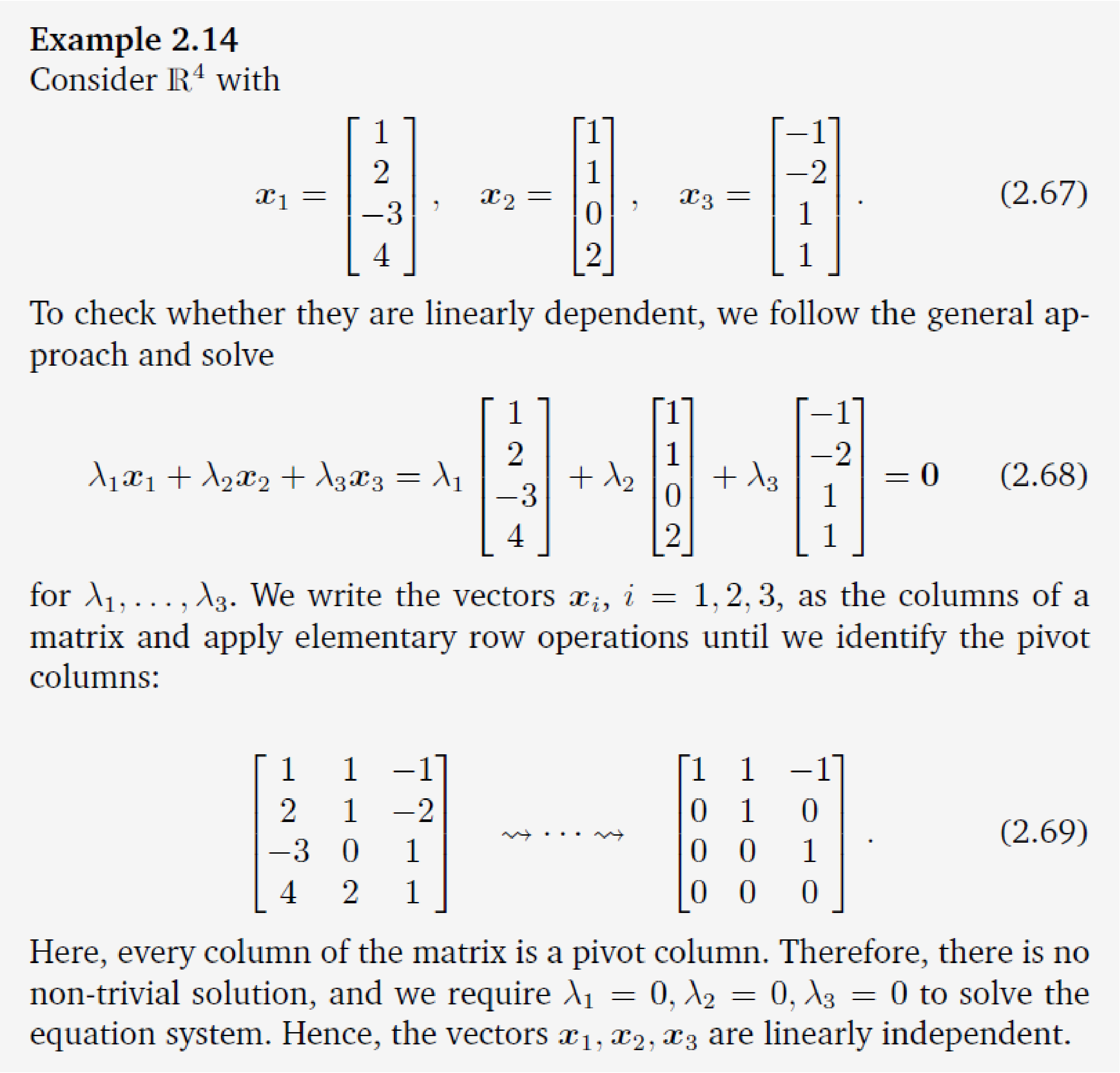
- A practical way of checking whether vectors $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k \in V$ are linearly independent is to use Gaussian elimination: Write all vectors as columns of a matrix $\boldsymbol{A}$ and perform Gaussian elimination until the matrix is in row echelon form (the reduced row-echelon form is unnecessary here):
  - The pivot columns indicate the vectors, which are linearly independent of the vectors on the left. Note that there is an ordering of vectors when the matrix is built.
  - The non-pivot columns can be expressed as linear combinations of the pivot columns on their left. For instance, the row-echelon form

$$\begin{bmatrix} 1 & 3 & 0 \\ 0 & 0 & 2 \end{bmatrix} \tag{2.66}$$

tells us that the first and third columns are pivot columns. The second column is a non-pivot column because it is three times the first column.

All column vectors are linearly independent if and only if all columns are pivot columns. If there is at least one non-pivot column, the columns (and, therefore, the corresponding vectors) are linearly dependent.

Linear independence



Example 2.14
Consider $\mathbb{R}^4$ with

$$\boldsymbol{x}_1 = \begin{bmatrix} 1 \\ 2 \\ -3 \\ 4 \end{bmatrix}, \quad \boldsymbol{x}_2 = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 2 \end{bmatrix}, \quad \boldsymbol{x}_3 = \begin{bmatrix} -1 \\ -2 \\ 1 \\ 1 \end{bmatrix}. \tag{2.67}$$

To check whether they are linearly dependent, we follow the general approach and solve

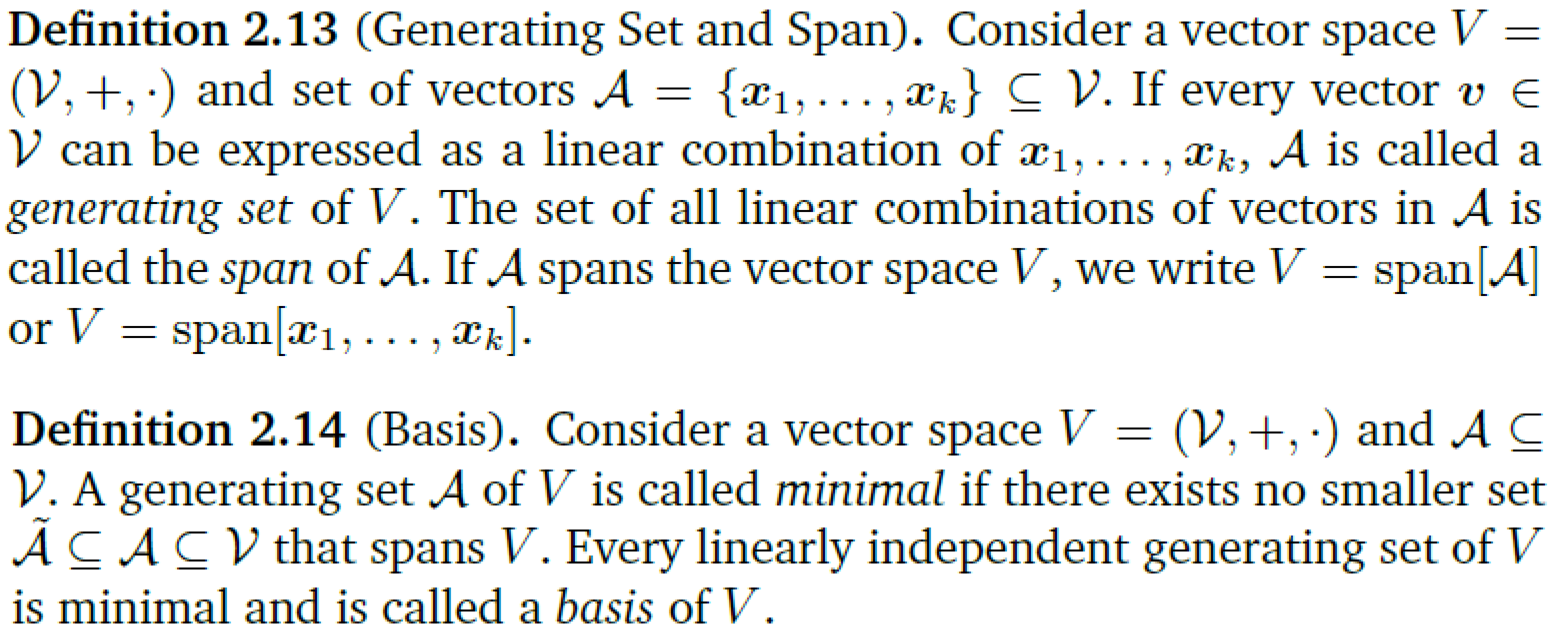
$$\lambda_1 \boldsymbol{x}_1 + \lambda_2 \boldsymbol{x}_2 + \lambda_3 \boldsymbol{x}_3 = \lambda_1 \begin{bmatrix} 1 \\ 2 \\ -3 \\ 4 \end{bmatrix} + \lambda_2 \begin{bmatrix} 1 \\ 1 \\ 0 \\ 2 \end{bmatrix} + \lambda_3 \begin{bmatrix} -1 \\ -2 \\ 1 \\ 1 \end{bmatrix} = \boldsymbol{0} \tag{2.68}$$

for $\lambda_1, \ldots, \lambda_3$. We write the vectors $\boldsymbol{x}_i$, $i = 1, 2, 3$, as the columns of a matrix and apply elementary row operations until we identify the pivot columns:

$$\begin{bmatrix} 1 & 1 & -1 \\ 2 & 1 & -2 \\ -3 & 0 & 1 \\ 4 & 2 & 1 \end{bmatrix} \rightsquigarrow \cdots \rightsquigarrow \begin{bmatrix} 1 & 1 & -1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}. \tag{2.69}$$

Here, every column of the matrix is a pivot column. Therefore, there is no non-trivial solution, and we require $\lambda_1 = 0, \lambda_2 = 0, \lambda_3 = 0$ to solve the equation system. Hence, the vectors $\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3$ are linearly independent.

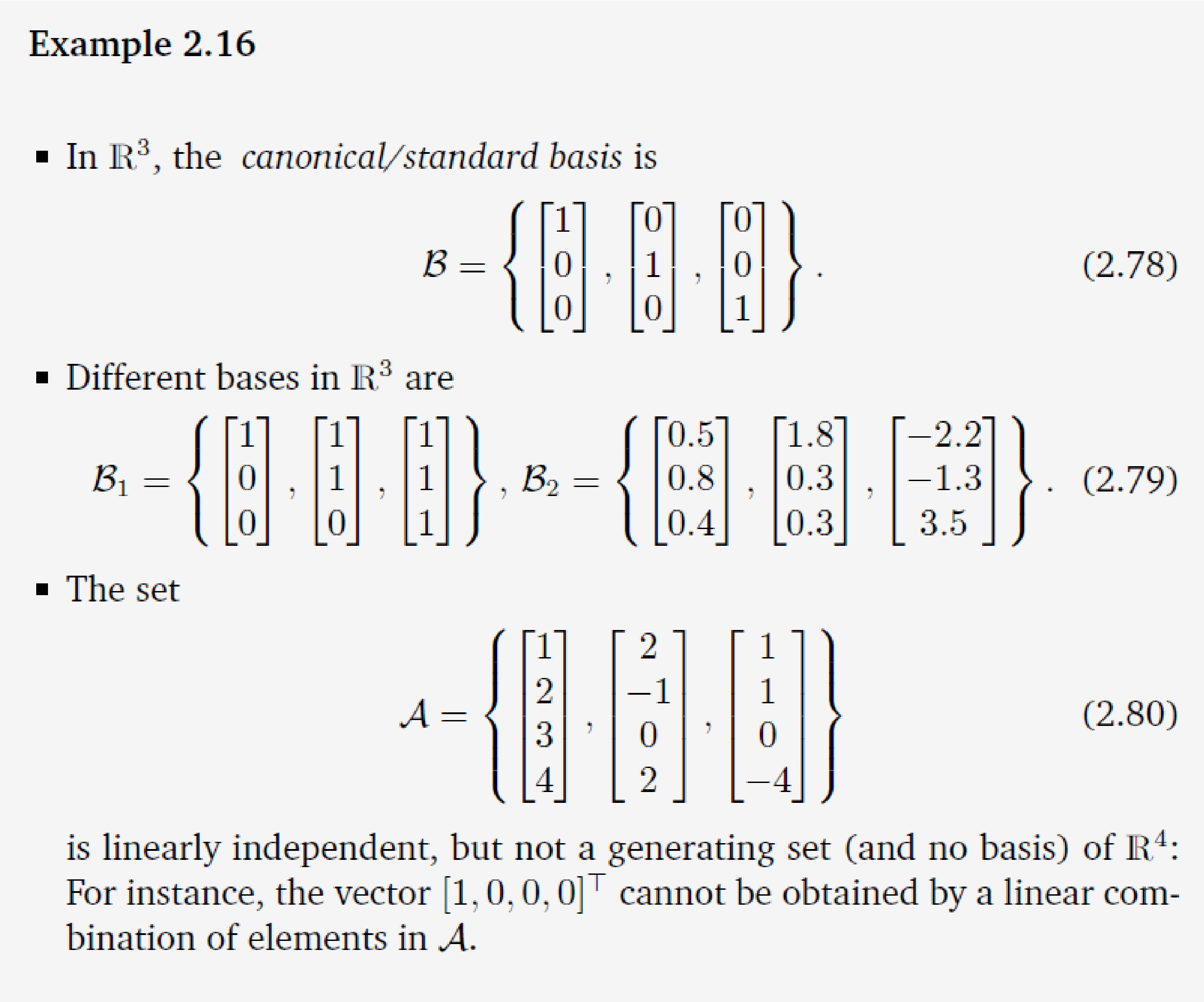
Basis



Definition 2.13 (Generating Set and Span)**.** Consider a vector space $V = (\mathcal{V}, +, \cdot)$ and set of vectors $\mathcal{A} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k\} \subseteq \mathcal{V}$. If every vector $\boldsymbol{v} \in \mathcal{V}$ can be expressed as a linear combination of $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k$, $\mathcal{A}$ is called a *generating set* of V. The set of all linear combinations of vectors in $\mathcal{A}$ is called the *span* of $\mathcal{A}$. If $\mathcal{A}$ spans the vector space V, we write $V = \text{span}[\mathcal{A}]$ or $V = \text{span}[\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k]$.

Definition 2.14 (Basis)**.** Consider a vector space $V = (\mathcal{V}, +, \cdot)$ and $\mathcal{A} \subseteq \mathcal{V}$. A generating set $\mathcal{A}$ of V is called *minimal* if there exists no smaller set $\tilde{\mathcal{A}} \subsetneq \mathcal{A} \subseteq \mathcal{V}$ that spans V. Every linearly independent generating set of V is minimal and is called a *basis* of V.

Basis



Example 2.16

- In $\mathbb{R}^3$, the *canonical/standard basis* is

$$\mathcal{B} = \left\{ \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \right\}. \tag{2.78}$$

- Different bases in $\mathbb{R}^3$ are

$$\mathcal{B}_1 = \left\{ \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \right\}, \mathcal{B}_2 = \left\{ \begin{bmatrix} 0.5 \\ 0.8 \\ 0.4 \end{bmatrix}, \begin{bmatrix} 1.8 \\ 0.3 \\ 0.3 \end{bmatrix}, \begin{bmatrix} -2.2 \\ -1.3 \\ 3.5 \end{bmatrix} \right\}. \tag{2.79}$$

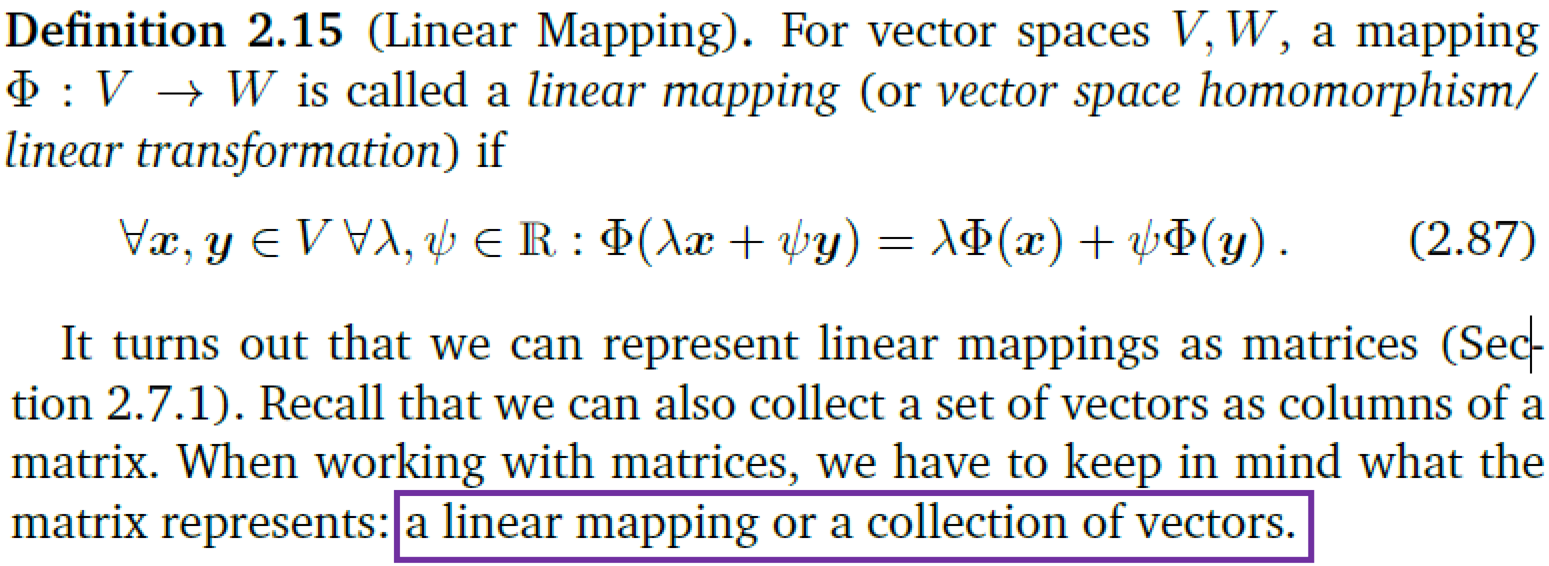
- The set

$$\mathcal{A} = \left\{ \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix}, \begin{bmatrix} 2 \\ -1 \\ 0 \\ 2 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 0 \\ -4 \end{bmatrix} \right\} \tag{2.80}$$

is linearly independent, but not a generating set (and no basis) of $\mathbb{R}^4$: For instance, the vector $[1, 0, 0, 0]^\top$ cannot be obtained by a linear combination of elements in $\mathcal{A}$.

Linear mapping

- Mappings on vector spaces that preserve their structure
- Earlier, we saw that vectors are objects that can be added together and multiplied by a scalar, and the resulting object is still a vector
- Now, we do the same for vector spaces



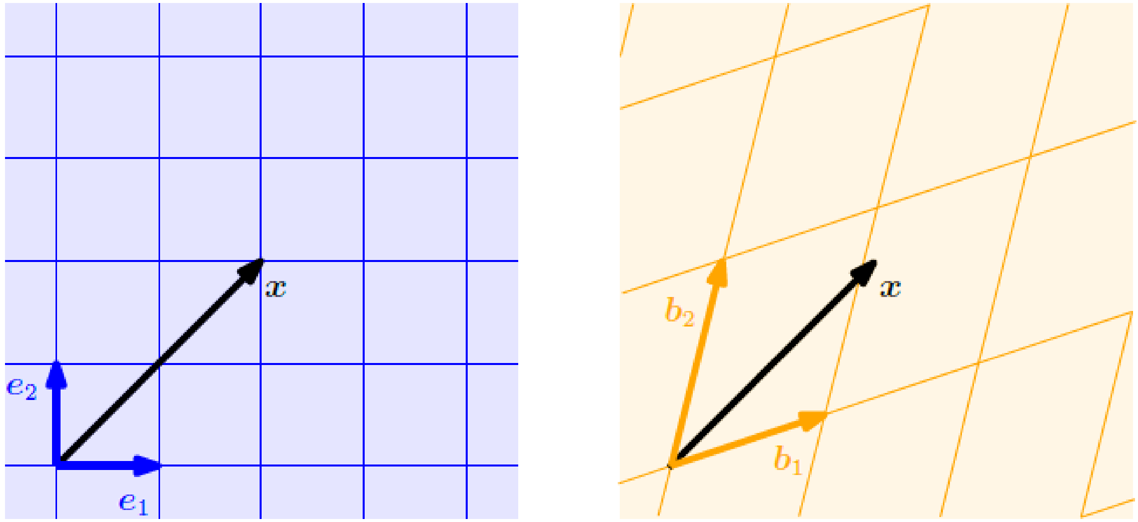
Definition 2.15 (Linear Mapping)**.** For vector spaces V, W, a mapping $\Phi : V \rightarrow W$ is called a *linear mapping* (or *vector space homomorphism/ linear transformation*) if

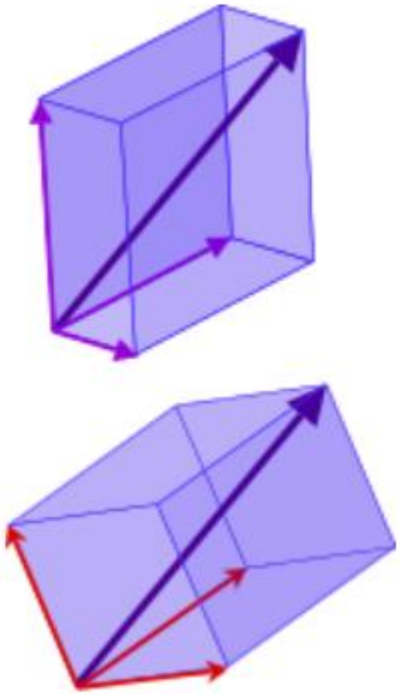
$$\forall \boldsymbol{x}, \boldsymbol{y} \in V \; \forall \lambda, \psi \in \mathbb{R} : \Phi(\lambda \boldsymbol{x} + \psi \boldsymbol{y}) = \lambda \Phi(\boldsymbol{x}) + \psi \Phi(\boldsymbol{y}) \,. \qquad (2.87)$$

It turns out that we can represent linear mappings as matrices (Section 2.7.1). Recall that we can also collect a set of vectors as columns of a matrix. When working with matrices, we have to keep in mind what the matrix represents: a linear mapping or a collection of vectors.

Linear mapping

Figure 2.8 Two different coordinate systems defined by two sets of basis vectors. A vector x has different coordinate representations depending on which coordinate system is chosen.

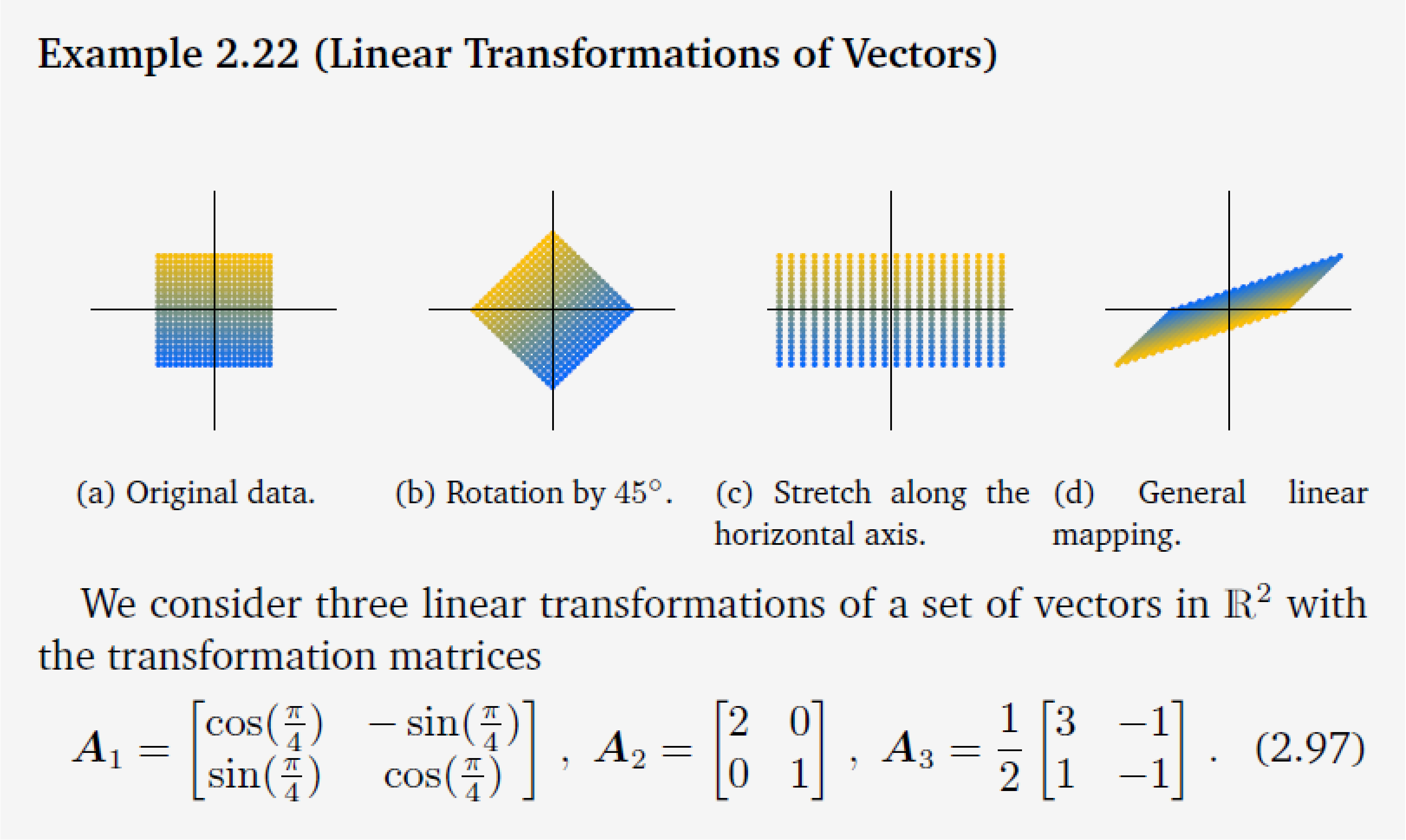




The same vector can be represented in two different bases (purple and red arrows).

Linear mapping

Figure 2.10 Three examples of linear transformations of the vectors shown as dots in (a); (b) Rotation by 45°; (c) Stretching of the horizontal coordinates by 2; (d) Combination of reflection, rotation and stretching.



Example 2.22 (Linear Transformations of Vectors)

(a) Original data.

(b) Rotation by 45°.

(c) Stretch along the horizontal axis.

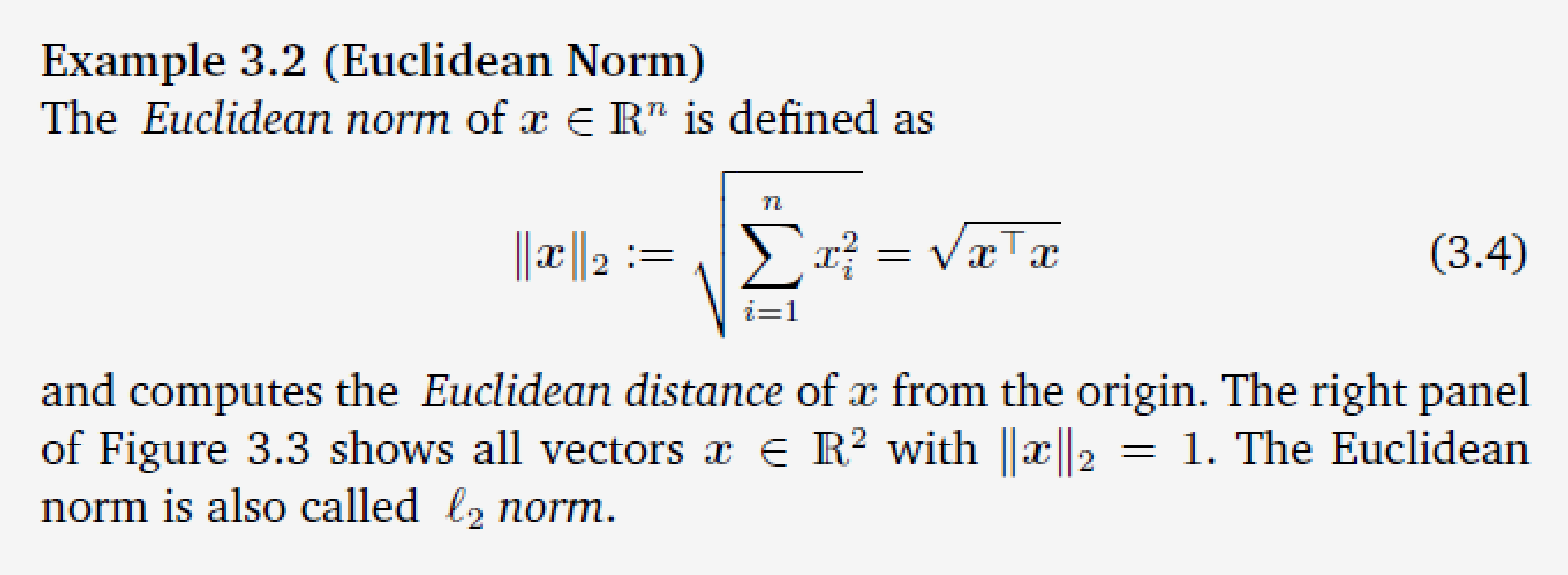
(d) General linear mapping.

We consider three linear transformations of a set of vectors in $\mathbb{R}^2$ with the transformation matrices

$$\boldsymbol{A}_1 = \begin{bmatrix} \cos(\frac{\pi}{4}) & -\sin(\frac{\pi}{4}) \\ \sin(\frac{\pi}{4}) & \cos(\frac{\pi}{4}) \end{bmatrix}, \quad \boldsymbol{A}_2 = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}, \quad \boldsymbol{A}_3 = \frac{1}{2}\begin{bmatrix} 3 & -1 \\ 1 & -1 \end{bmatrix}. \tag{2.97}$$

Analytical Geometry
[Chapter 3 – MML book]

Norms



Example 3.2 (Euclidean Norm)
The *Euclidean norm* of $x \in \mathbb{R}^n$ is defined as

$$\|x\|_2 := \sqrt{\sum_{i=1}^{n} x_i^2} = \sqrt{x^\top x} \tag{3.4}$$

and computes the *Euclidean distance* of x from the origin. The right panel of Figure 3.3 shows all vectors $x \in \mathbb{R}^2$ with $\|x\|_2 = 1$. The Euclidean norm is also called *ℓ_2 norm*.

Norms

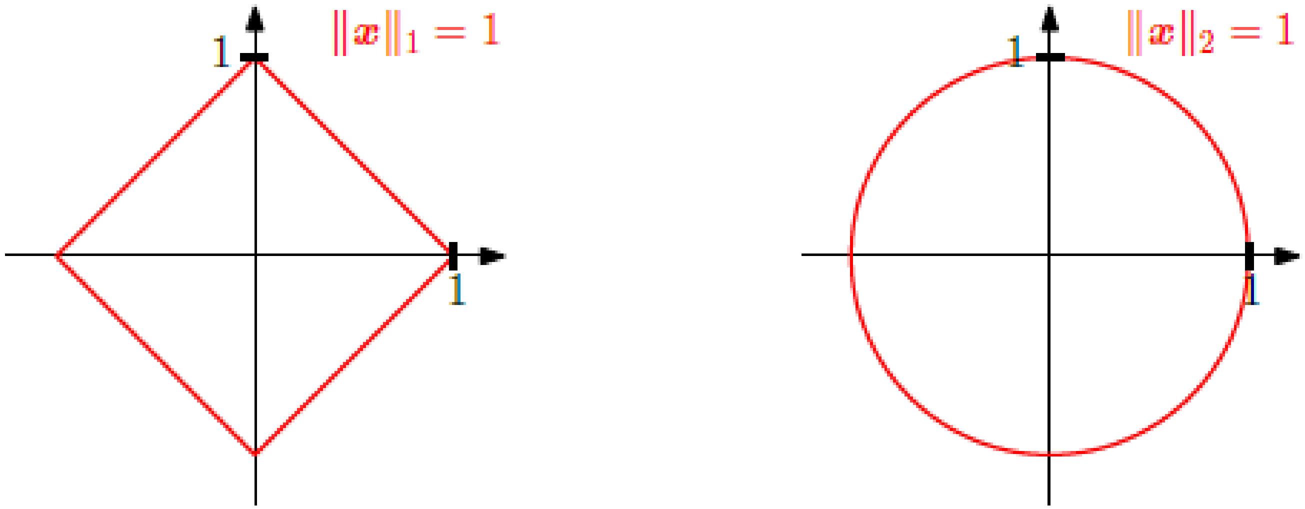


Figure 3.3 For different norms, the red lines indicate the set of vectors with norm 1. Left: Manhattan norm; Right: Euclidean distance.

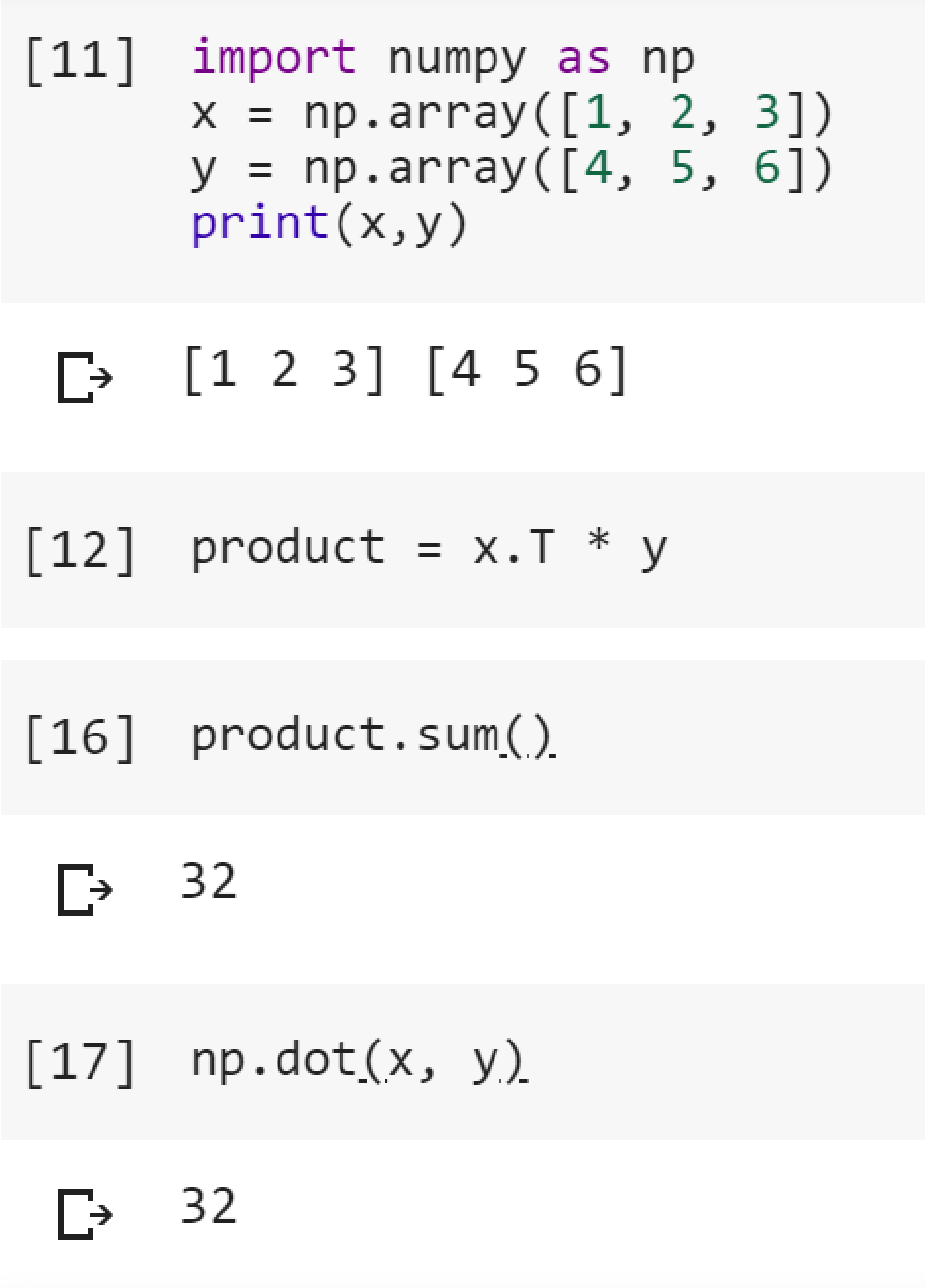
Dot Product

(type of inner product)

Scalar product/dot product

$$x^\top y = \sum_{i=1}^{n} x_i y_i \,. \tag{3.5}$$

$$a_1 \cdot b_1 = \begin{bmatrix} 1 \\ 7 \end{bmatrix} \cdot \begin{bmatrix} 3 \\ 5 \end{bmatrix} = (1 \cdot 3) + (7 \cdot 5) = 38$$



```
[11] import numpy as np
     x = np.array([1, 2, 3])
     y = np.array([4, 5, 6])
     print(x,y)
```

```
[1 2 3] [4 5 6]
```

```
[12] product = x.T * y
```

```
[16] product.sum()
```

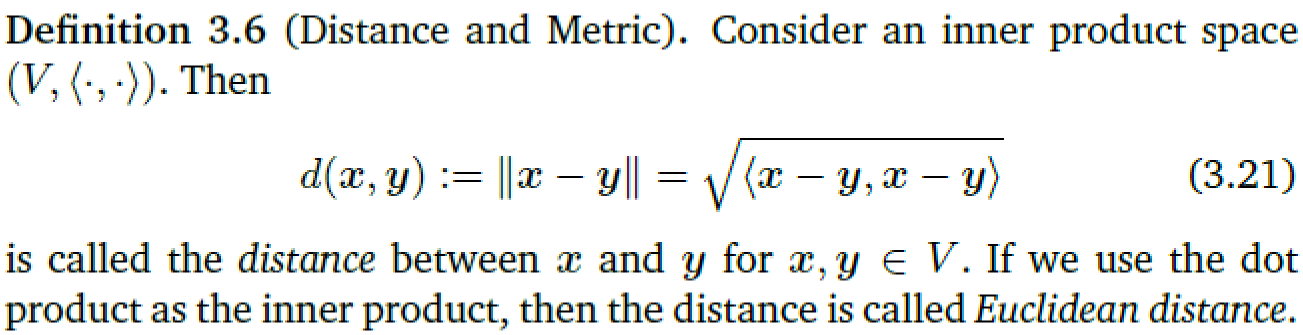
```
32
```

```
[17] np.dot(x, y)
```

```
32
```

https://ml-cheatsheet.readthedocs.io/en/latest/linear_algebra.html

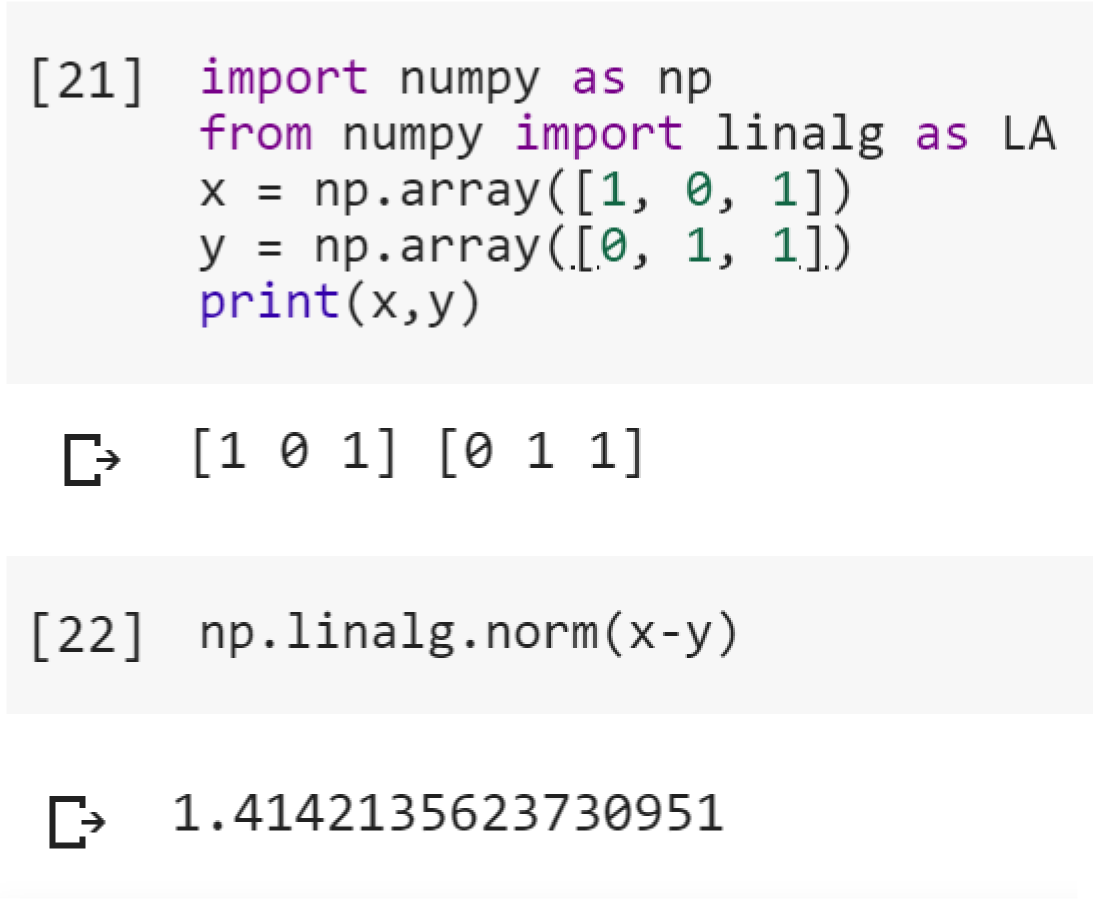
Lengths and Distances



Definition 3.6 (Distance and Metric). Consider an inner product space $(V, \langle\cdot,\cdot\rangle)$. Then

$$d(\boldsymbol{x}, \boldsymbol{y}) := \|\boldsymbol{x} - \boldsymbol{y}\| = \sqrt{\langle \boldsymbol{x} - \boldsymbol{y}, \boldsymbol{x} - \boldsymbol{y} \rangle} \tag{3.21}$$

is called the *distance* between $\boldsymbol{x}$ and $\boldsymbol{y}$ for $\boldsymbol{x}, \boldsymbol{y} \in V$. If we use the dot product as the inner product, then the distance is called *Euclidean distance*.



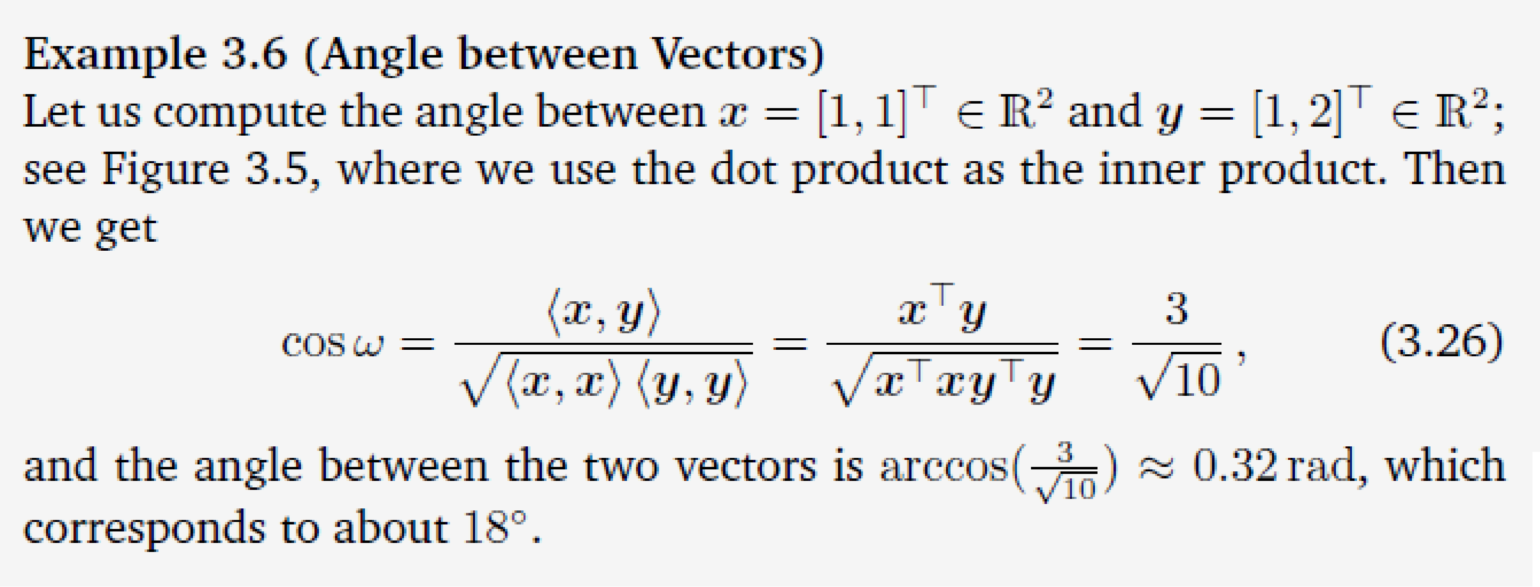
```
[21]  import numpy as np
      from numpy import linalg as LA
      x = np.array([1, 0, 1])
      y = np.array([0, 1, 1])
      print(x,y)
```

```
[1 0 1] [0 1 1]
```

```
[22]  np.linalg.norm(x-y)
```

```
1.4142135623730951
```

Angles and Orthogonality



Example 3.6 (Angle between Vectors)
Let us compute the angle between $\boldsymbol{x} = [1, 1]^\top \in \mathbb{R}^2$ and $\boldsymbol{y} = [1, 2]^\top \in \mathbb{R}^2$; see Figure 3.5, where we use the dot product as the inner product. Then we get

$$\cos\omega = \frac{\langle \boldsymbol{x}, \boldsymbol{y}\rangle}{\sqrt{\langle \boldsymbol{x}, \boldsymbol{x}\rangle \langle \boldsymbol{y}, \boldsymbol{y}\rangle}} = \frac{\boldsymbol{x}^\top \boldsymbol{y}}{\sqrt{\boldsymbol{x}^\top \boldsymbol{x} \boldsymbol{y}^\top \boldsymbol{y}}} = \frac{3}{\sqrt{10}}, \qquad (3.26)$$

and the angle between the two vectors is $\arccos(\frac{3}{\sqrt{10}}) \approx 0.32\,\mathrm{rad}$, which corresponds to about 18°.

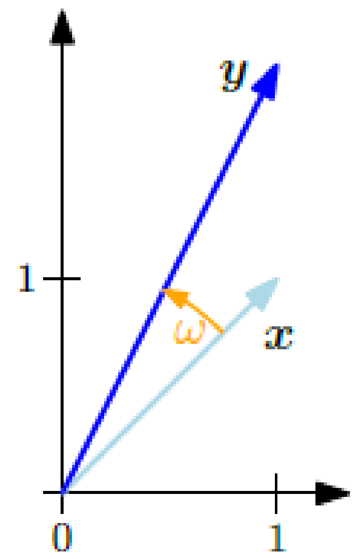
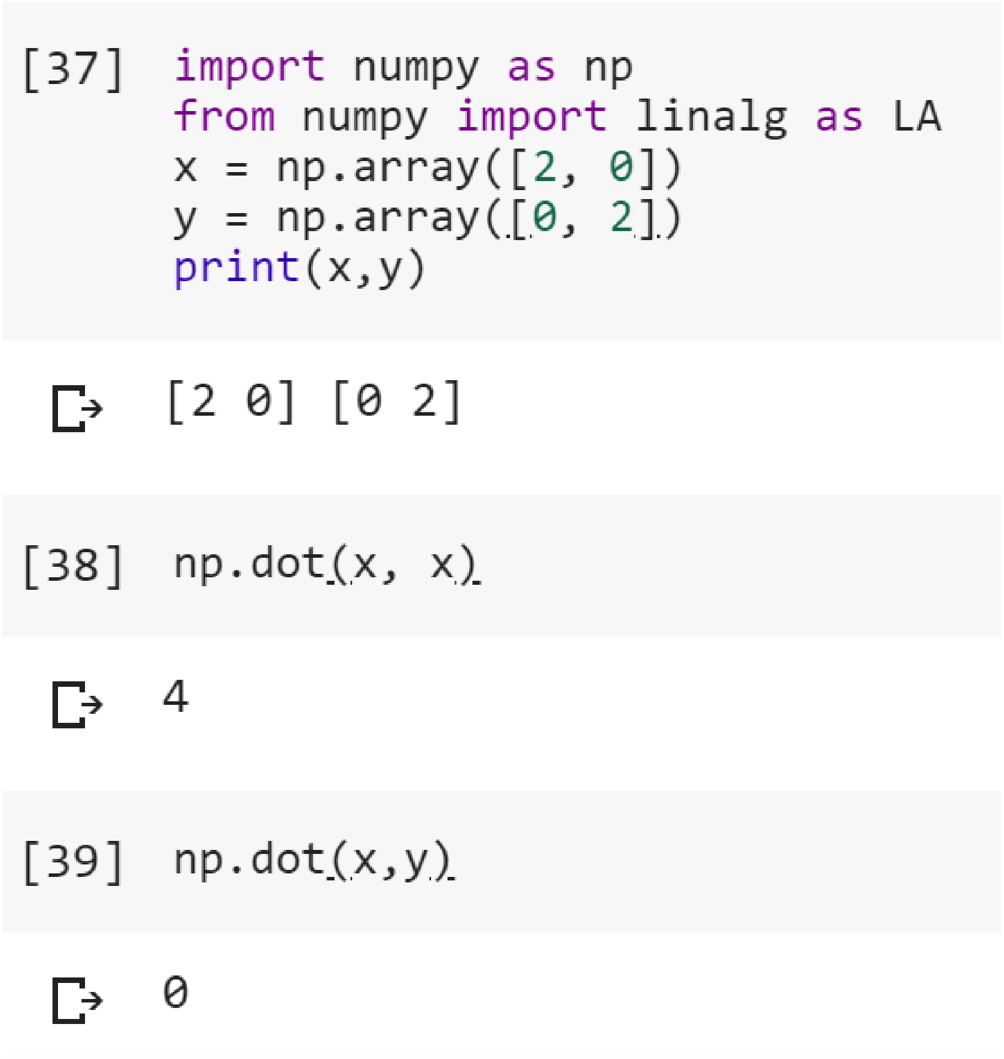


Figure 3.5 The angle ω between two vectors $\boldsymbol{x}, \boldsymbol{y}$ is computed using the inner product.

Angles and Orthogonality

Checking if orthogonal



```
[37]  import numpy as np
      from numpy import linalg as LA
      x = np.array([2, 0])
      y = np.array([0, 2])
      print(x,y)
```

```
[2 0] [0 2]
```

```
[38]  np.dot(x, x)
```

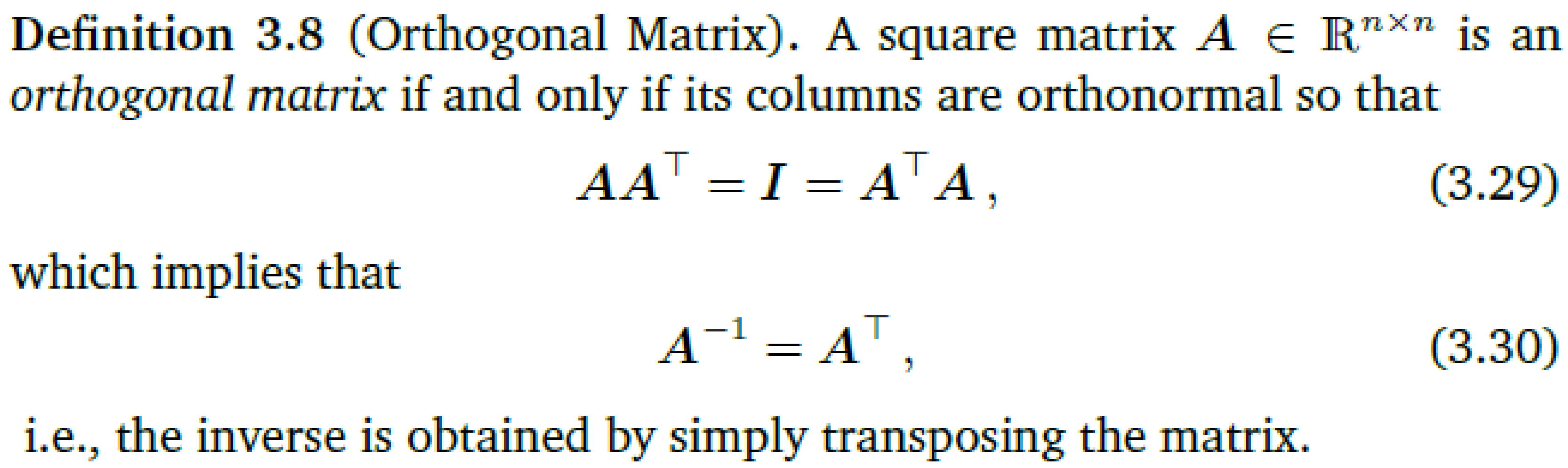
```
4
```

```
[39]  np.dot(x,y)
```

```
0
```

Angles and Orthogonality

Orthonormal = orthogonal and unit vectors



Definition 3.8 (Orthogonal Matrix)**.** A square matrix $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ is an *orthogonal matrix* if and only if its columns are orthonormal so that

$$\boldsymbol{A}\boldsymbol{A}^\top = \boldsymbol{I} = \boldsymbol{A}^\top\boldsymbol{A}\,, \tag{3.29}$$

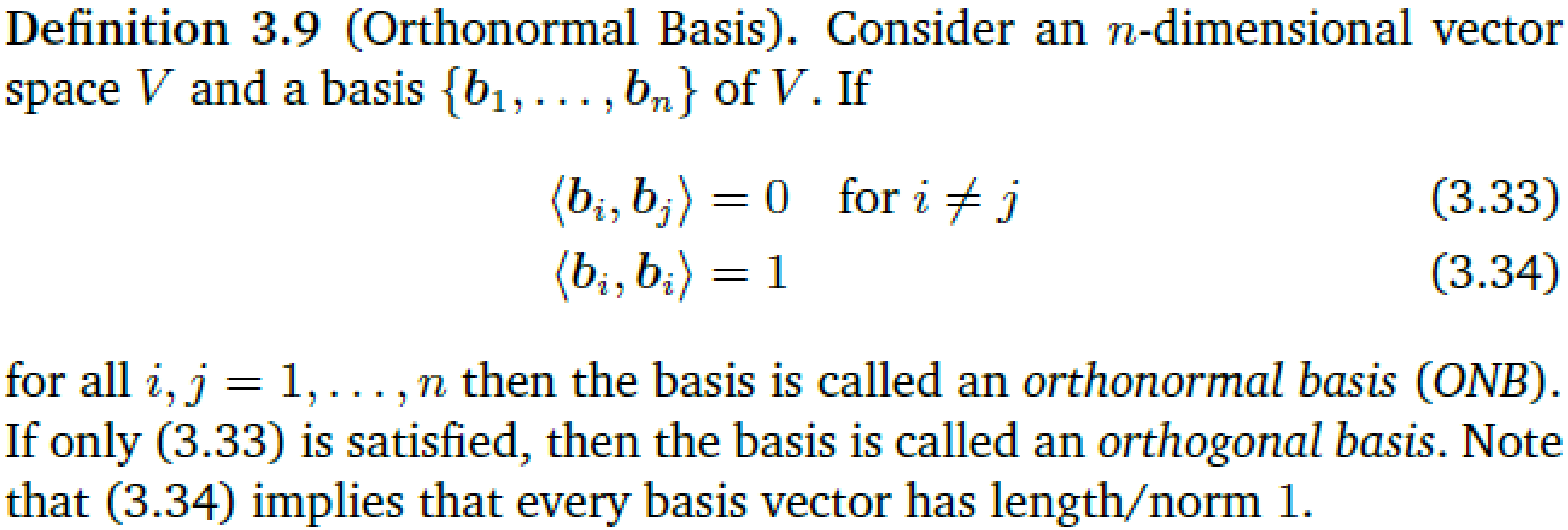
which implies that

$$\boldsymbol{A}^{-1} = \boldsymbol{A}^\top\,, \tag{3.30}$$

i.e., the inverse is obtained by simply transposing the matrix.

Orthonormal Basis

- In *n*-dimensional space, we need *n* basis vectors that are linearly independent
- If these vectors are orthogonal, and each has length 1, it's a special case: ***orthonormal basis***



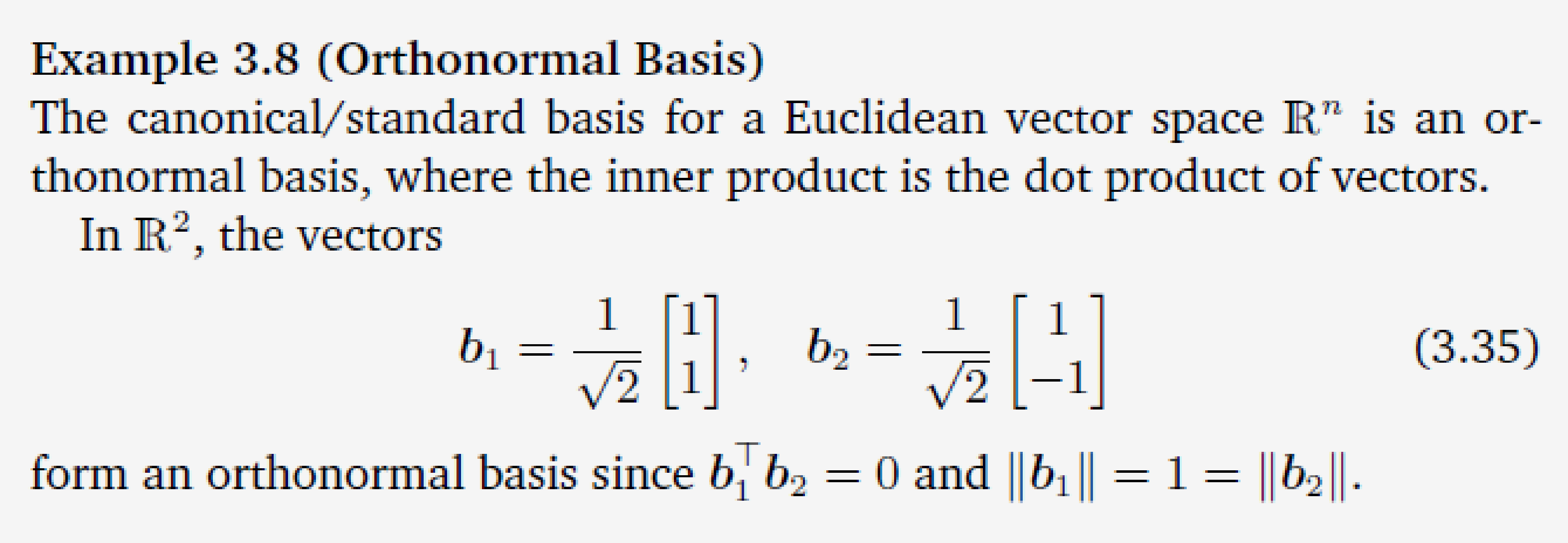
Definition 3.9 (Orthonormal Basis). Consider an n-dimensional vector space V and a basis $\{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n\}$ of V. If

$$\langle \boldsymbol{b}_i, \boldsymbol{b}_j \rangle = 0 \quad \text{for } i \neq j \tag{3.33}$$

$$\langle \boldsymbol{b}_i, \boldsymbol{b}_i \rangle = 1 \tag{3.34}$$

for all $i, j = 1, \ldots, n$ then the basis is called an *orthonormal basis* (*ONB*). If only (3.33) is satisfied, then the basis is called an *orthogonal basis*. Note that (3.34) implies that every basis vector has length/norm 1.

Orthonormal Basis



Example 3.8 (Orthonormal Basis)
The canonical/standard basis for a Euclidean vector space $\mathbb{R}^n$ is an orthonormal basis, where the inner product is the dot product of vectors.

In $\mathbb{R}^2$, the vectors

$$b_1 = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\1\end{bmatrix}, \quad b_2 = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\-1\end{bmatrix} \tag{3.35}$$

form an orthonormal basis since $b_1^\top b_2 = 0$ and $\|b_1\| = 1 = \|b_2\|$.

Questions

Question 1

- As discussed in lecture, when applying a K-nearest-neighbors classifier, it is common to normalize each input dimension to unit variance. Why might it be advantageous to do this?

Question 1

- As discussed in lecture, when applying K-nearest-neighbors, it is common to normalize each input dimension to unit variance.
    - If one feature has higher variance than the others, it will be treated as more important. This is problematic if the features have arbitrary scales, since the importance of a feature would depend on the unit used. Normalizing to unit variance fixes this problem.

Question 2

You are given a labeled dataset and asked to train a classifier model to be as accurate as possible, while having a good idea of how said model would eventually perform on new data. Explain how you would (a) split your data set, (b) train your model and optimize hyperparameters and (c) evaluate performance. Be as descriptive as possible.

Question 2

You are given a labeled dataset and asked to train a classifier model to be as accurate as possible, while having a good idea of how said model would eventually perform on new data. Explain how you would (a) split your data set, (b) train your model and optimize hyperparameters and (c) evaluate performance. Be as descriptive as possible.

(a) I would split data into training/validation (80%) and testing (20%) groups. Within the training/validation group, I would allocate 25% to validation and the rest to training and rotate through during a cross-validation cycle (such that each data point is used once for validation, and three times for testing).

(b) I would perform hyperparameter optimization by rotating the validation set within the training/validation group as described above. For a given cycle, I would use a set of hyperparameters. I would eventually choose the set of hyperparameters with the greatest validation accuracy (averaged over corresponding cross-validation cycles).

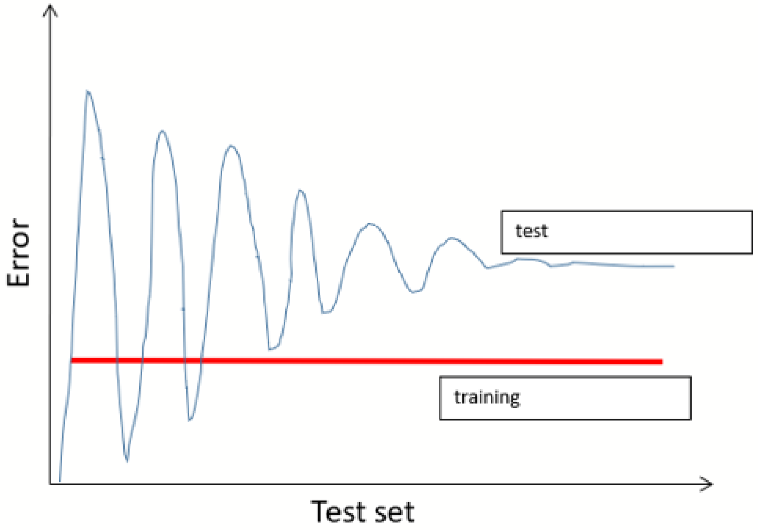
(c) The model would then be applied to the test set, and we'd report the testing accuracy.

Question 3

Plot a graph where x-axis is the test set size and y-axis is (i) the training error and (ii) the test error. Assume a well-trained, typical machine learning model with fixed complexity and a fixed training set. Describe your graph.

Question 3

Plot a graph where x-axis is the test set size and y-axis is (i) the training error and (ii) the test error. Assume a well-trained, typical machine learning model with fixed complexity and a fixed training set. Describe your graph.



The training error will not change if the training set is fixed. The test error will fluctuate initially, for a small number of points, then converge to a value above that of training.