Complete Solutions to Lecture 2 Assignments

Minh Nguyen, Long Le, Giang Hoang, and Hung Ngo

Last update: 20:10, Wednesday 15th April, 2020

Contents

1 From Lecture Slides **1**

2 Coding Exercises **13**

3 Extra Practice **14**

1 From Lecture Slides

P 2.1

> We can turn the affine transformation $T_b x = Ax + b$ with $A \in \mathbb{R}^{n \times n}, x, b \in \mathbb{R}^n$ into a linear transformation using **augmented matrix** with $\bar{A} = \left[\begin{array}{ccc|c} & A & & b \\ 0 & \dots & 0 & 1 \end{array}\right]$ and $\bar{x} = \begin{bmatrix} x \\ 1 \end{bmatrix}$. The last (seems redundant) line allows us to combine any number of affine transformations (e.g., a series of rotation, translation, then rotation) into one by multiplying the respective matrices. Verify this for yourself!

Summary

We first give a proof of equivalence between affine transformation series and products of augmented matrices. Then, some visualization examples for $\mathbb{R}^2$ affine transformation series are included.

Proof of equivalence between affine transformation series and product of augmented matrices

Let's say we have a series of affine transformations $(T_i)_{i=1}^t$ where

$$T_i x = A_i x + b_i \,, \quad i = 1, \dots, t \,.$$

We will prove by induction that for $n = 1, \dots, t$,

$$\bar{A}_n \dots \bar{A}_1 \bar{x} = \begin{bmatrix} T_n \circ \dots \circ T_1 x \\ 1 \end{bmatrix} . \tag{$P(n)$}$$

To begin with, we note down the following matrix multiplication:

$$\bar{A}\bar{x} = \left[\begin{array}{c|c} A & b \\ \text{---}\ \mathbf{0}\ \text{---} & 1 \end{array} \right] \begin{bmatrix} x \\ 1 \end{bmatrix} = \begin{bmatrix} Ax + b \\ 1 \end{bmatrix} . \tag{1.1}$$

The base case $P(1)$ follows directly from (1.1). Suppose $P(n)$ is correct, we can write

$$\begin{aligned} \bar{A}_{n+1} \bar{A}_n \dots \bar{A}_1 \bar{x} &= \bar{A}_{n+1} \begin{bmatrix} T_n \circ \dots \circ T_1 x \\ 1 \end{bmatrix} \\ &= \left[\begin{array}{c|c} A_{n+1} & b_{n+1} \\ \text{---}\ \mathbf{0}\ \text{---} & 1 \end{array} \right] \begin{bmatrix} T_n \circ \dots \circ T_1 x \\ 1 \end{bmatrix} . \end{aligned}$$

Using (1.1), we can deduce that

$$\begin{aligned} \bar{A}_{n+1} \dots \bar{A}_1 \bar{x} &= \begin{bmatrix} A_{n+1} T_n \circ \dots \circ T_1 x + b_{n+1} \\ 1 \end{bmatrix} \\ &= \begin{bmatrix} T_{n+1} \circ T_n \circ \dots \circ T_1 x \\ 1 \end{bmatrix} , \end{aligned}$$

which proves $P(n+1)$. Therefore $P(n)$ holds for $n = 1, \dots, t$ by induction. $\square$

Visualization

Minh Nguyen have created a Python package for visualizing affine transformations, called Avine. Some examples for affine transformation series can be found at the linked GitHub repository.

Notes

1. $v = [\text{---}\ \mathbf{0}\ \text{---}\ 1]$ is the unique solution to

$$\begin{bmatrix} Ax+b \\ 1 \end{bmatrix} = \left[\begin{array}{c|c} A & b \\ \text{---}\ v & \text{---} \end{array}\right] \begin{bmatrix} x \\ 1 \end{bmatrix}.$$

P 2.3

> Prove that **matrix multiplication** $\mathcal{T} : \mathbb{R}^{n\times p} \to \mathbb{R}^{m\times p}$ with $\mathcal{T}C = D$, $\mathcal{T} = B_{m\times n}$ is a linear map. Given $A = \begin{bmatrix} 1 & 2 \\ 0 & 1 \end{bmatrix}$ to transform 2 by 2 images, i.e., $C_{2\times 2} = A_{2\times 2} B_{2\times 2}$. Find matrix representation of this linear transformation in the standard bases. Note that the coordinate vectors of 2D images in standard basis is simply a reshaping (**flattening**, **vectorizing**).

Notation

We denote the rows of $\mathcal{T}$ by

$$\mathcal{T} = \begin{bmatrix} \text{---} & \mathcal{T}_1 & \text{---} \\ & \vdots & \\ \text{---} & \mathcal{T}_m & \text{---} \end{bmatrix}.$$

Let C and C' be $\mathbb{R}^{n\times p}$ matrices. We denote their columns by

$$C = \begin{bmatrix} | & & | \\ C_1 & \dots & C_p \\ | & & | \end{bmatrix}, \quad C' = \begin{bmatrix} | & & | \\ C'_1 & \dots & C'_p \\ | & & | \end{bmatrix}.$$

Proving that $\mathcal{T}$ is linear

For scalars $a, b \in \mathbb{R}$,

$$\begin{aligned}
\mathcal{T}(aC + bC') &= \mathcal{T}\begin{bmatrix} | & & | \\ aC_1 + bC'_1 & \dots & aC_p + bC'_p \\ | & & | \end{bmatrix} \\
&= \begin{bmatrix} a\mathcal{T}_1 C_1 + b\mathcal{T}_1 C'_1 & \dots & a\mathcal{T}_1 C_p + b\mathcal{T}_1 C'_p \\ \vdots & \ddots & \vdots \\ a\mathcal{T}_m C_1 + b\mathcal{T}_m C'_1 & \dots & a\mathcal{T}_m C_p + b\mathcal{T}_m C'_p \end{bmatrix} \\
&= a\mathcal{T}C + b\mathcal{T}C' \,.
\end{aligned}$$

Therefore $\mathcal{T}$ is linear.

Finding matrix representation of A

In this solution, $\mathrm{vec}(B)$ stands for the transformation which turns each **row** of matrix B into a column and stack them on top of one another (vectorization). Furthermore, the standard basis $(\mathbf{h}_i)_{i=1}^4$ of $\mathbb{R}^{2\times 2}$ is ordered as

$$\mathbf{h}_1 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, \quad \mathbf{h}_2 = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad \mathbf{h}_3 = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}, \quad \mathbf{h}_4 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}.$$

A matrix B is represented as $\mathrm{vec}(B)$ in standard basis $(\mathbf{h}_i)_{i=1}^4$. Note that $\mathrm{vec}(\mathbf{h}_i) = \mathbf{e}_i$ the standard basis of $\mathbb{R}^4$, e.g., $\mathbf{e}_1 = (1, 0, 0, 0)^\top$.

Our task is to find the matrix $\mathcal{A}$, representation of A in standard basis $(\mathbf{h}_i)_{i=1}^4$, such that

$$\mathrm{vec}(AB) = \mathcal{A}\,\mathrm{vec}(B)\,,$$

for image $B \in \mathbb{R}^{2\times 2}$. As a linear transformation from $\mathbb{R}^4$ to itself, matrix $\mathcal{A}$ must be of size 4×4. Note that $\mathcal{A}\mathbf{e}_i$ is i-th column of $\mathcal{A}$, and $\mathcal{A}\,\mathrm{vec}(\mathbf{h}_i) = \mathrm{vec}(A\mathbf{h}_i)$. Hence we can specify $\mathcal{A}$ by

$$\begin{aligned}
\mathcal{A} &= [\mathcal{A}\mathbf{e}_1\ \mathcal{A}\mathbf{e}_2\ \mathcal{A}\mathbf{e}_3\ \mathcal{A}\mathbf{e}_4] \\
&= [\mathcal{A}\,\mathrm{vec}(\mathbf{h}_1)\ \mathcal{A}\,\mathrm{vec}(\mathbf{h}_2)\ \mathcal{A}\,\mathrm{vec}(\mathbf{h}_3)\ \mathcal{A}\,\mathrm{vec}(\mathbf{h}_4)] \\
&= [\mathrm{vec}(A\mathbf{h}_1)\ \ \mathrm{vec}(A\mathbf{h}_2)\ \ \mathrm{vec}(A\mathbf{h}_3)\ \ \mathrm{vec}(A\mathbf{h}_4)]\,,
\end{aligned}$$

Hence

$$\mathcal{A} = \begin{bmatrix} 1 & 0 & 2 & 0 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Notes

1. Vectorization often refers to the stacking of **columns**. Here we changed it to the stacking of **rows** so that it is compatible to our ordering of the $\mathbb{R}^{2\times 2}$ standard basis. If one instead defines vectorization the other way and orders the said basis as

$$\mathbf{h}_1 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, \quad \mathbf{h}_2 = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}, \quad \mathbf{h}_3 = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad \mathbf{h}_4 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix},$$

matrix $\mathcal{A}$ is then

$$\mathcal{A} = \begin{bmatrix} 1 & 2 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

while all other arguments are still valid.

2. We assumed that $f : \mathrm{vec}(\mathrm{B}) \mapsto \mathrm{vec}(AB)$ is a linear transformation. You can verify this for yourself.

P 2.4

> Prove that for a linear map $f : \mathbb{R}^5 \to \mathbb{R}^5$, $\ker(f)$ is a subspace of $\mathbb{R}^5$. If $f(x_1, ..., x_5) = (x_1, ..., x_4, 0)$, then is f a linear function? Find the matrix of f, $\ker(f)$ and $\dim(\ker(f))$. Note that f is a projection.

Proving $\ker(f)$ is a subspace of $\mathbb{R}^5$

Since $\ker(f)$ is a subset of $\mathbb{R}^5$, we only focus on the closure of addition and scalar multiplication.

For vectors $u, v \in \mathbb{R}^5$,

$$f(u+v) = f(u) + f(v) = \mathbf{0},$$

hence f is closed under addition.

For vector $u \in \mathbb{R}^5$ and scalar $a \in \mathbb{R}$,

$$f(au) = af(u) = \mathbf{0}\,,$$

hence f is also closed under scalar multiplication.

Verifying if $f(x_1, \dots, x_5) = (x_1, \dots, x_4, 0)$ is a linear function

The specified function is linear: for vector $y = (y_1, \dots, y_5) \in \mathbb{R}^5$ and scalar $a \in \mathbb{R}$,

$$\begin{aligned} f(ax_1 + y_1, \dots ax_5 + y_5) &= (ax_1 + y_1, \dots ax_4 + y_4, 0 + y_5) \\ &= a(x_1, \dots, x_5) + y\,. \end{aligned}$$

The matrix of f is

$$X_f = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}\,.$$

Finding $\ker(f)$ and $\dim(\ker(f))$

Due to the definition of f, $\ker(f)$ consists solely of vectors $v \in \mathbb{R}^5$ such that $x_1 = \cdots = x_4 = 0$, i.e., $\ker(f) = \{(0, 0, 0, 0, x_5) : x_5 \in \mathbb{R}\}$. This subspace has a basis $B = \{(0, \dots, 0, 1)\}$ and therefore is 1-dimensional.

P 2.5

> Verify for yourself the effect of **coordinate scaling** Sx with scaling or **diagonal matrix** $S = \text{diag}(\lambda_1, \dots, \lambda_n) \in \mathbb{R}^{n \times n}$.

Notation

Let the coordinates of x be $(x_1, \dots, x_n)$. We also denote the result of Sx by vector $y = [y_1 \dots y_n]^\top$ and the i^{th} row of matrix S by S_i.

Verifying the scaling effect

For each entry y_i of y,

$$\begin{aligned} y_i &= S_i \cdot x \\ &= [0 \dots 0 \; \lambda_i \; 0 \dots 0] \cdot [0 \dots 0 \; x_i \; 0 \dots 0]^\top \\ &= \lambda_i x_i \, . \end{aligned}$$

Therefore, the diagonal matrix S scales each coordinate x_i of x by λ_i.

$$\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \overset{Sx}{\rightarrow} \begin{bmatrix} \lambda_1 x_1 \\ \lambda_2 x_2 \\ \vdots \\ \lambda_n x_n \end{bmatrix}$$

P 2.6

> Re-derive block matrix multiplications using the general form
>
> $$\begin{aligned} AB &= \left[\begin{array}{c|c} A_{11} & A_{12} \\ \hline A_{21} & A_{22} \end{array}\right] \left[\begin{array}{c|c} B_{11} & B_{12} \\ \hline B_{21} & B_{22} \end{array}\right] \\ &= \left[\begin{array}{c|c} A_{11}B_{11} + A_{12}B_{21} & A_{11}B_{12} + A_{12}B_{22} \\ \hline A_{21}B_{11} + A_{22}B_{21} & A_{21}B_{12} + A_{22}B_{22} \end{array}\right] . \end{aligned} \tag{6.1}$$
>
> That is, how do you split into A_{ij}, B_{kl} in each case?

Applying block matrix multiplication to column view and row view

As for column view

$$Ax = \begin{bmatrix} | & & | \\ A_1 & \dots & A_n \\ | & & | \end{bmatrix} \begin{bmatrix} x_1 \\ \ddots \\ x_n \end{bmatrix} = x_1 \begin{bmatrix} | \\ A_1 \\ | \end{bmatrix} + \dots + x_n \begin{bmatrix} | \\ A_n \\ | \end{bmatrix} ,$$

the blocks of A are its columns while the blocks of x are its entries.

As for row view

$$
\begin{aligned}
x^\top A &= [x_1 \dots x_n] \begin{bmatrix} — & A_1 & — \\ & \vdots & \\ — & A_n & — \end{bmatrix} \\
&= x_1 [— \; A_1 \; —] + \dots + x_n [— \; A_n \; —] \,,
\end{aligned}
$$

the blocks of A are its rows while the blocks of $x^\top$ are its entries.

Proof of block matrix multiplication rule

Notation

For a block matrix C, we denote the block(s) at position (i, j) by $C_{(i,j)}$. Within block $C_{(i,j)}$, the entry/entries at (k, l) is denoted by $C_{(i,j)(k,l)}$. Note that we can use slices for indexing, e.g. $1 : n$ instead of i.

Main proof

Let's say we have block matrices $A \in \mathbb{R}^{M \times N}$ and $B \in \mathbb{R}^{N \times P}$ with block sizes of respectively $m \times n$ and $n \times p$. Now consider their product $C \in \mathbb{R}^{M \times P}$. Due to size alignment in matrix multiplication, C can be divided into blocks containing m rows and p columns. We now take a look into entry $C_{(i,j)(k,l)}$ of $C_{(i,j)}$:

$$
\begin{aligned}
C_{(i,j)(k,l)} &= \sum_{x=1}^{N} A_{(i-1)m+k,x} B_{x,(j-1)p+l} \\
&= \sum_{y=1}^{N/n} A_{(i-1)m+k,(y-1)n+1:yn} B_{(y-1)n+1:yn,(j-1)p+l} \\
&= \sum_{y=1}^{N/n} A_{(i,y)(k,1:n)} B_{(y,j)(1:n,l)} \,.
\end{aligned}
$$

While x represents individual rows/columns, y stands for block indices (column-wise in A and row-wise in B). The change of summation index therefore specifies that we are cutting row/column partitions into blocks.

As for $C_{(i,j)}$, plugging in the worked out formula for individual entries, we get

$$
\begin{aligned}
C_{(i,j)} &= \begin{bmatrix} \sum_{y=1}^{N/n} A_{(i,y)(1,1:n)} B_{(y,j)(1:n,1)} & \cdots & \sum_{y=1}^{N/n} A_{(i,y)(1,1:n)} B_{(y,j)(1:n,p)} \\ \vdots & \ddots & \vdots \\ \sum_{y=1}^{N/n} A_{(i,y)(m,1:n)} B_{(y,j)(1:n,1)} & \cdots & \sum_{y=1}^{N/n} A_{(i,y)(m,1:n)} B_{(y,j)(1:n,p)} \end{bmatrix} \\
&= \sum_{y=1}^{N/n} \begin{bmatrix} A_{(i,y)(1,1:n)} B_{(y,j)(1:n,1)} & \cdots & A_{(i,y)(1,1:n)} B_{(y,j)(1:n,p)} \\ \vdots & \ddots & \vdots \\ A_{(i,y)(m,1:n)} B_{(y,j)(1:n,1)} & \cdots & A_{(i,y)(m,1:n)} B_{(y,j)(1:n,p)} \end{bmatrix} \\
&= \sum_{y=1}^{N/n} A_{(i,y)} B_{(y,j)} ,
\end{aligned}
$$

hence the block matrix multiplication rule. □

P 2.7

> Write down the transformation matrices in homogeneous coordinates and visualize in 2D (**coding**) the sequence of 3 affine transformations: i) rotate in z axis an angle θ, ii) translate in (x, y)-plane an amount $(p_x, p_y, 0)$, iii) then reflect about the origin, of a 2D object of your choice.

Transformation matrices

The transformation matrix for rotation around z-axis an angle θ is

$$
T_\theta = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 & 0 \\ \sin(\theta) & \cos(\theta) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} . \tag{7.1}
$$

The transformation matrix for translation in (x, y)-plane by a vector $(p_x, p_y, 0)$ is

$$
T_{(p_x,p_y,0)} = \begin{bmatrix} 1 & 0 & 0 & p_x \\ 0 & 1 & 0 & p_y \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} . \tag{7.2}
$$

```python
import numpy as np
from avine.Visualize import *

def visualize_sequence(data, theta, p_x, p_y):

    '''
    2D-visualize a sequence of rotation around z-axis,
    translation by [p_x, p_y], and reflection through origin.

    Parameters:
    - data: array_like.
    - theta: float. Rotation angle in radian.
    - p_x: float. Translation vector in x direction.
    - p_y: float. Translation vector in y direction.
    '''

    func_list = [(3, theta), (1, [p_x, p_y]), 6]
    ShowSeries(data, func_list)
```

Code 1: A function for visualizing transformation sequences in **P 2.7**

The transformation matrix for reflection through the origin is

$$T_r = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{7.3}$$

Combining (7.1), (7.2), and (7.3), we get the transformation sequence matrix

$$T_r = T_r T_{(p_x, p_y, 0)} T_\theta = \begin{bmatrix} -\cos(\theta) & \sin(\theta) & 0 & -p_x \\ -\sin(\theta) & -\cos(\theta) & 0 & -p_y \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Visualization

In **code 1**, a function for visualizing the desired transformation sequence is included, using Avine.

P 2.8

> In PCA: input image $X \overset{T}{\rightleftharpoons}$ Coordinate vector $\bar{z} = (1, w_1, \ldots, w_n)^\top$ (no reduction). Then with dimensionality reduction: $\bar{z} \overset{A}{\rightarrow} z = (1, w_1, \ldots, w_k)^\top$, i.e., keeping only first $k+1$ coordinates. Find matrix A of the coordinate transformation.

Because $A : \mathbb{V}^{n+1} \rightarrow \mathbb{V}^{k+1}$, A is $(k+1) \times (n+1)$.Use $P2.5$, we have A is diagonal matrix with the elements in the main diagonal are 1.

P 2.9

> Prove that for any A: $\text{rank}(A^\top A) = \text{rank}(AA^\top) = \text{rank}(A)$.

We first show that nullspaces $N(A^\top) = N(AA^\top)$. Take $x \in N(A^\top)$, we have $A^\top x = 0$, then $AA^\top x = 0$. So, $x \in N(AA^\top)$. Suppose $y \in N(AA^\top)$, $AA^\top y = 0$. We then have $y^\top AA^\top y = 0$, it means $||A^\top y|| = 0$ then $A^\top y = 0$, $y \in N(A^\top)$.

Recall Rank Nullity Theorem for Matrices: $\forall A_{m \times n}$: $\text{rank}(A) + \text{nullity}(A) = n$ the dimension of input space, and similarly $\text{rank}(A) + \text{nullity}(A^\top) = m$. Thus, $\dim N(A^\top) = m - \text{rank}(A^\top)$ and $\text{rank}(AA^\top) = m - \dim(N(AA^\top))$. We conclude that $\text{rank}(AA^\top) = \text{rank}(A^\top)$.

We apply the same reasoning for $\text{rank}(A^\top A) = \text{rank}(A) = \text{rank}(A^\top)$.

P 2.10

> Show that for square & full rank matrices:
>
> - $(ABC)^{-1} = C^{-1}B^{-1}A^{-1}$
> - $(A^{-1})^T = (A^T)^{-1} =: A^{-T}$

- $(AB)(AB)^{-1} = I \Rightarrow B^{-1}A^{-1} = (AB)^{-1}$. Then consider $AB = D$.
- $((A)^{-1}A)^\top = I \Rightarrow (A^\top)^{-1} = A^{(-1)^\top}$.

P 2.11

> Prove again for yourself $\mathsf{rank}(A_{m\times n}) = \mathsf{rank}(A^\top) \leq \min(m, n)$.

From P2.9: $\mathsf{rank}(A^\top) = \mathsf{rank}(A^\top A) = \mathsf{rank}(AA^\top) = \mathsf{rank}(A)$. But $\mathsf{rank}(A^\top A) = \mathsf{rank}(B_{n\times n}) \leq n$ and $\mathsf{rank}(AA^\top) = \mathsf{rank}(C_{m\times m}) \leq m$. Hence QED. See also lecture slides.

2 Coding Exercises

C 2.1

> Reproduce affine transformations in $\mathbb{R}^2$ and make animations by changing the parameters slowly.

Solution

See Minh Nguyen's Avine package on GitHub.

C 2.2–5

Students must continue working on these plotting assignments.

3 Extra Practice

E 2.1

> Let $\beta = \{v_1, \dots, v_n\}$ be a basis of a vector space V, and $T : V \to W$ be a linear transformation from V to W. Is the following proposition true? Why? If the formula is wrong, correct it.
>
> $$T(\sum_{i=1}^{n} a_i v_i) = T(v) \Leftrightarrow \sum_{i=1}^{n} a_i v_i = v. \tag{1.1}$$

Notation

Without loss of generality, the expression $\sum_{i=1}^{n} a_i v_i$ can be replaced with u, a vector of V.

Verifying the proposition

The proposition is not correct. Statement (1.1), which we can rewrite as

$$Tu = Tv \Leftrightarrow u = v,$$

essentially means T is *injective*. This is wrong: not all linear transformations are injective.

Correction for (1.1)

To fix the proposition, we may introduce more constraints. Here are some possible options:

- $\dim(V) = \dim(W)$ and T is surjective. This makes $\{Tv_i\}_{i=1}^{n}$ a basis of W. Let's say $v = \sum b_i v_i$. Then

  $$Tu - Tv = \sum (a_i - b_i) T v_i.$$

  If $Tu = Tv$, we obtain

  $$\mathbf{0}_W = \sum (a_i - b_i) T v_i,$$

  and thus $a_i = b_i, \forall i$.

- $\text{Null}(T) = \{\mathbf{0}_W\}$. As T is linear,

  $$Tu - Tv = T(u - v).$$

  If $Tu = Tv$, we obtain

  $$\mathbf{0}_W = T(u - v),$$

  and thus $u - v = \mathbf{0}_V$.

We may also correct the formula without imposing any new constraint, although it will not be as meaningful:

$$T(\sum_{i=1}^{n} a_i v_i) = T(v) \Leftarrow \sum_{i=1}^{n} a_i v_i = v. \tag{1.1'}$$

E 2.2

> Let V be a finite-dimensional vector space and $\alpha = \{v_1, \ldots, v_n\}$ be a basis of V. Let W be another vector space with some vectors $\beta = \{w_1, \ldots, w_n\}$. Prove that there exists exactly one linear transformation $T : V \to W$ such that
>
> $$T(v_j) = w_j, \forall j. \tag{2.1}$$

Summary

This solution proceeds with first a proof of existence and then a proof of uniqueness.

Proof of existence

We can construct a function by specifying for each value in its domain a corresponding element of its codomain. Equation (2.1) already gave us such specification for vectors in α. To ensure linearity, for each $v = \sum a_i v_i \in V$, we must have

$$Tv = \sum a_i T(v_i) = \sum a_i w_i \,. \tag{2.2}$$

The transformation defined by

$$Tv = \sum a_i w_i$$

is indeed linear; we can check by applying it to $ax + by$, where $x = \sum x_i v_i$ and $y = \sum y_i v_i$ are vectors in V:

$$\begin{aligned} T(ax + by) &= T(\sum (ax_i + by_i) v_i) \\ &= \sum (ax_i + by_i) w_i \\ &= a \sum x_i w_i + b \sum y_i w_i \\ &= aT(x) + bT(y) \,. \end{aligned}$$

As such, we have confirmed the existence of our desired transformation.

Proof of uniqueness

In (2.2), we argued that each linear transformation T in accordance with (2.1) must also satisfy

$$Tv = \sum a_i w_i \,.$$

Since the tuple $(a_i)_{i=1}^{n}$ is unique with respect to v, the value specified by $\sum a_i w_i$ is also unique. Hence there is exactly one such transformation.

E 2.4

> Prove that the null space and the row space of a matrix are orthogonal, i.e. every vector in null space is orthogonal to every vector in row space (zero dot product).

(Moved to Assignment 3).

E 2.6

> *(Generalization of Fibonacci's rabbits - see page 62 of Tao's notes)*
>
> There are 3 characteristic parameters for each population of rabbits:
>
> - fertility rate p_b: the average number of juvenile pairs given birth by an adult pair each year
> - child mortality rate p_c: the probability that a juvenile will not reach their second year
> - adult mortality rate p_a: the probability that an adult will not survive each year
>
> Your first task is to find a function $f([A\ B]^\top \mid t, p_b, p_c, p_a)$ which outputs the total number of rabbit pairs after t years given the aforementioned parameters and an initial population of A juvenile pairs and B adult pairs (written with matrix notation).
>
> Next, let the parameters be uniformly distributed random variables: $p_b \sim \mathcal{U}(b_l, b_u)$, $p_c \sim \mathcal{U}(c_l, c_u)$, and $p_a \sim \mathcal{U}(a_l, a_u)$. Find the linear upper and lower bound function sets $U = \{f_u : \mathbb{R}^{+2} \to \mathbb{R} \mid f_u(v) = [u_0\ u_1]v$ and $f(v \mid t = 1) \leq f_u(v), \forall v\}$ and $L = \{f_l : \mathbb{R}^{+2} \to \mathbb{R} \mid f_l(v) = [l_0\ l_1]v$ and $f_l(v) \leq f(v \mid t = 1), \forall v\}$.
>
> Assumptions:
>
> - Number of rabbit pairs can be non-natural.
> - A rabbit dies as soon as the other in its pair dies.

Notation

Let $g([A\ B]^\top \mid t)$ be the function which gives the number of juvenile and adult pairs in year t as an $\mathbb{R}^2$ vector. Then $f([A\ B]^\top) = [1\ 1]g([A\ B]^\top)$.

Finding f

To know how a rabbit population changes after t years, we first examine how it changes annually, i.e., search for $f([A\ B]^\top \mid t=1)$.

We know that juvenile pairs can only grow into adulthood with a survival rate of p_c:

$$g\left(\begin{bmatrix} A \\ 0 \end{bmatrix} \mid t=1\right) = \begin{bmatrix} 0 \\ (1-p_c)A \end{bmatrix} = A\begin{bmatrix} 0 \\ 1-p_c \end{bmatrix}. \tag{6.1}$$

Meanwhile, adult pairs can affect the direct future population in two aspects–giving birth to new pairs and staying alive:

$$g\left(\begin{bmatrix} 0 \\ B \end{bmatrix} \mid t=1\right) = \begin{bmatrix} p_b B \\ (1-p_a)B \end{bmatrix} = B\begin{bmatrix} p_b \\ 1-p_a \end{bmatrix}. \tag{6.2}$$

Combining (6.1) and (6.2) gives us

$$\begin{aligned} g\left(\begin{bmatrix} A \\ B \end{bmatrix} \mid t=1\right) &= A\begin{bmatrix} 0 \\ 1-p_c \end{bmatrix} + B\begin{bmatrix} p_b \\ 1-p_a \end{bmatrix} \\ &= \begin{bmatrix} 0 & p_b \\ 1-p_c & 1-p_a \end{bmatrix}\begin{bmatrix} A \\ B \end{bmatrix}. \end{aligned}$$

The population vector at $t=n$ is given by the n^{th} power of g's corresponding matrix. The remaining task is just to produce f from g:

$$\begin{aligned} f\left(\begin{bmatrix} A \\ B \end{bmatrix} \mid t=n\right) &= [1\ 1]g\left(\begin{bmatrix} A \\ B \end{bmatrix} \mid t=n\right) \\ &= [1\ 1]\begin{bmatrix} 0 & p_b \\ 1-p_c & 1-p_a \end{bmatrix}^n \begin{bmatrix} A \\ B \end{bmatrix}. \end{aligned}$$

Finding U and L

As for this second part, we solve for U and L by translating their properties into simpler inequalities. Let's start with

$$f(v \mid t = 1) \leq f_u(v), \forall v\,.$$

Utilizing the found formula of f and the given formula of f_u, we obtain an equivalent inequality:

$$[1\ 1] \begin{bmatrix} 0 & p_b \\ 1 - p_c & 1 - p_a \end{bmatrix} \begin{bmatrix} A \\ B \end{bmatrix} \leq [u_0\ u_1] \begin{bmatrix} A \\ B \end{bmatrix}, \forall \begin{bmatrix} A \\ B \end{bmatrix}\,.$$

Moving everything to the right-hand side, we get

$$0 \leq \begin{bmatrix} u_0 - 1 + p_c \\ u_1 - 1 + p_a - p_b \end{bmatrix}^\top \begin{bmatrix} A \\ B \end{bmatrix}, \forall \begin{bmatrix} A \\ B \end{bmatrix}\,.$$

Substitution of respectively $[1\ 0]^\top$ and $[0\ 1]^\top$ for $[A\ B]^\top$ gives us two necessary conditions:

$$\begin{cases} 0 \leq & u_0 - 1 + p_c\,, \\ 0 \leq & u_1 - 1 + p_a - p_b\,. \end{cases}$$

As both A and B are non-negative, these conditions are also sufficient. We now rewrite them as

$$\begin{cases} 1 - p_c & \leq u_0\,, \\ 1 - p_a + p_b & \leq u_1\,. \end{cases}$$

From these inequalities, the set of possible parameter tuples for $f_u \in U$ is the Cartesian product

$$(\sup\{1 - p_c\}, +\infty] \times (\sup\{1 - p_a + p_b\}, +\infty]\,.$$

As $p_b \sim \mathcal{U}(b_l, b_u)$, $p_c \sim \mathcal{U}(c_l, c_u)$, and $p_a \sim \mathcal{U}(a_l, a_u)$, this product is equal to

$$(1 - c_l, +\infty] \times (1 - a_l + b_u, +\infty]\,.$$

Therefore $U = \{f_u : \mathbb{R}^{+2} \to \mathbb{R} \mid f_u(v) = [u_0\ u_1]v,$ where $(u_0, u_1) \in (1 - c_l, +\infty] \times (1 - a_l + b_u, +\infty]\}$.

Similarly, $L = \{f_l : \mathbb{R}^{+2} \to \mathbb{R} \mid f_u(v) = [l_0\ l_1]v,$ where $(l_0, l_1) \in [-\infty, 1 - c_u) \times [-\infty, 1 - a_u + b_l)\}$.