Chapter 1

Vectors and Matrices

The last chapter provided an overview of linear algebra, using many image examples. In this chapter, we'll focus primarily on vector-matrix multiplications. First you'll see the general definition,followed by a few examples. You'll have an opportunity to solve a problem manually and then you'll get to use Python again. In this chapter, we will use two-dimensional matrices for simplicity. But a matrix can have any number of dimensions.

1.1 Vector-Matrix Multiplication

Let's take a look at the general form of vector-matrix multiplication. Given a matrix A of size $m \times n$ and a vector x of size $n \times 1$, the product Ax is a new vector of size $m \times 1$.

You compute the i-th component of the product vector Av by taking the dot product of the i-th row of A and the vector v:

$$(Av)_i = \sum_{j=1}^{n} a_{i,j} x_j$$

where $a_{i,j}$ is the element in the i-th row and j-th column of A, and v_j is the j-th element of v.

This is the general form of a matrix and a vector, written to show the specific components of each:

$$A = \begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} & \dots & a_{1,n} \\ a_{2,1} & a_{2,2} & a_{2,3} & \dots & a_{2,n} \\ \dots \\ a_{m,1} & a_{m,2} & a_{m,3} & \dots & a_{m,n} \end{bmatrix}$$

$$v = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ \dots \\ v_m \end{bmatrix}$$

$$Av = \begin{bmatrix} v_1 * a_{1,1} + v_2 * a_{1,2} + v_3 * a_{1,3} + ... + v_m * a_{1,n} \\ v_1 * a_{2,1} + v_2 * a_{2,2} + v_3 * a_{2,3} + ... + v_m * a_{2,n} \\ ... \\ v_1 * a_{m,1} + v_2 * a_{m,2} + v_3 * a_{m,3} + ... + v_m * a_{m,n} \end{bmatrix}$$

Let's look at a specific example.

$$A = \begin{bmatrix} 2 & 4 & 6 \\ 3 & 5 & 7 \\ 1 & 2 & 3 \\ 8 & 6 & 2 \end{bmatrix}$$

$$v = \begin{bmatrix} -2 \\ 1 \\ 3 \end{bmatrix}$$

Solution:

$$= \begin{bmatrix} -2 * 2 + 1 * 4 + 3 * 6 \\ -2 * 3 + 1 * 5 + 3 * 7 \\ -2 * 1 + 1 * 2 + 3 * 3 \\ -2 * 8 + 1 * 6 + 3 * 2 \end{bmatrix}$$

$$= \begin{bmatrix} 18 \\ 20 \\ 9 \\ -4 \end{bmatrix}$$

1.2 Trail Mix for Mars

Let's look at an applied problem. Three astronauts (Pat, Kai, and River) are getting ready for a trip to Mars. NASA food service is preparing trail mix for the voyage, tailored to each astronaut's taste. The chef needs to submit a budget based on the cost of the trail mix for each astronaut. The mix is made up of raisins, almonds, and chocolate.

Pat prefers a raisins:almonds:chocolate ratio of 6:10:4. Kai likes 2:3:15. River wants 14:1:5. The chef can buy a kg of raisins for $7.50, a kg of almonds for $14.75, and a kg of chocolate for $22.25. Assuming each astronaut will get 20 kg of trail mix, which astronaut will cost more to feed?

First, set up a matrix to represent the raisins:almonds:chocolate ratios. (Conveniently, these ratios already add to 20.)

$$\text{MixRatios} = \begin{bmatrix} 6 & 10 & 4 \\ 2 & 3 & 15 \\ 14 & 1 & 5 \end{bmatrix}$$

Use a vector to represent the cost of each item:

$$\text{IngredientCost} = \begin{bmatrix} 7.50 \\ 14.75 \\ 22.25 \end{bmatrix}$$

To find the cost of trail mix for each astronaut, we simply find the dot product between the mix ratios and the ingredient costs to get:

Pat = \$281.50
Kai = \$615.50
River = \$231.00

Exercise 1 Vector Matrix Multiplication

Multiply the array A with the vector v. Compute this by hand, and make sure to show your computations.

$$A = \begin{bmatrix} 1 & -2 & 3 & 5 \\ -4 & 2 & 7 & 1 \\ 3 & 3 & -9 & 1 \end{bmatrix}$$

$$v = \begin{bmatrix} 2 \\ 2 \\ 6 \\ -1 \end{bmatrix}$$

Working Space

Answer on Page 5

1.2.1 Vector-Matrix Multiplication in Python

Most real-world problems use very large matrices where it becomes impractical to do calculations by hand. As long as you understand how matrix-vector multiplication is performed, you'll be equipped to use a computing language, like Python, to do the calculations for you.

Create a file called `vectors_matrices.py` and enter this code:

```
# import the python module that supports matrices
import numpy as np

# create an array
a = np.array([[5, 1, 3, -2],
              [1, -1, 8, 4],
              [6, 2, 1, 3]])

# create a vector
b = np.array([1, 2, 3, -8])

# calculate the dot product
print(a.dot(b))
```

When you run it, you should see:

```
[16, 6, 8]
```

1.3 Where to Learn More

Watch this video from Khan Academy about matrix-vector products: https://rb.gy/frga5

This is a draft chapter from the Kontinua Project. Please see our website (https://kontinua.org/) for more details.

Appendix A

Answers to Exercises

Answer to Exercise 1 (on page 3)

$$Av = (1137 - 43)$$