

Matematika 3

Scalar and Vector



Name

Dicha Zelianivan Arkana

NIM

2241720002

Class

2i

Department

Information Technology

Study Program

D4 Informatics Engineering

Tasks

1. (a) Scalar
(b) Vector
(c) Scalar
(d) Scalar
2. Using Numpy

```
[1] import numpy as np

# 1 dimensional array
x = np.array([1, 2, 3, 4])
print("1d array", x)

# 2 dimensional array
A = np.array([[1, 2], [3, 4], [5, 6]])
print("2d array", A)

# transpose
A_t = A.T
print("Transpose", A_t)

A_t.shape

1d array [1 2 3 4]
2d array [[1 2]
 [3 4]
 [5 6]]
Transpose [[1 3 5]
 [2 4 6]]
(2, 3)
```

3. Row and Column Vector

```
✓ [4] import numpy as np
    0s

    # vector as row
    vector_row = np.array([1, 2, 3]);

    # vector as column
    vector_column = np.array([[1],
                              [2],
                              [3]])
```

4. Showing results

```
▶ import numpy as np

# create row vector
vector = np.array([1, 2, 3, 4, 5, 6])

# create matrix vector
matrix = np.array([[1, 2, 3],
                   [4, 5, 6],
                   [7, 8, 9]])

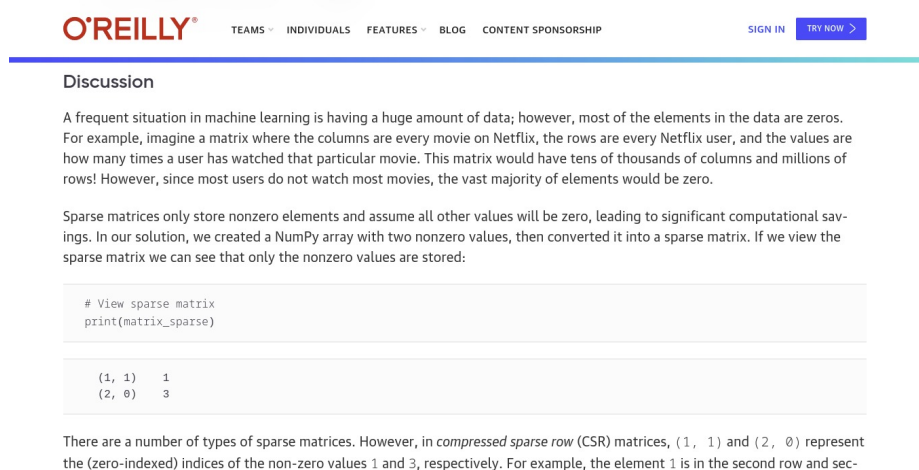
print(vector[2])

print(vector[:])

print(vector[:3])

print(vector[3:])
```

5. Explore



O'REILLY TEAMS ▾ INDIVIDUALS FEATURES ▾ BLOG CONTENT SPONSORSHIP SIGN IN TRY NOW >

Discussion

A frequent situation in machine learning is having a huge amount of data; however, most of the elements in the data are zeros. For example, imagine a matrix where the columns are every movie on Netflix, the rows are every Netflix user, and the values are how many times a user has watched that particular movie. This matrix would have tens of thousands of columns and millions of rows! However, since most users do not watch most movies, the vast majority of elements would be zero.

Sparse matrices only store nonzero elements and assume all other values will be zero, leading to significant computational savings. In our solution, we created a NumPy array with two nonzero values, then converted it into a sparse matrix. If we view the sparse matrix we can see that only the nonzero values are stored:

```
# View sparse matrix
print(matrix_sparse)
```

(1, 1)	1
(2, 0)	3

There are a number of types of sparse matrices. However, in *compressed sparse row (CSR)* matrices, $(1, 1)$ and $(2, 0)$ represent the (zero-indexed) indices of the non-zero values 1 and 3, respectively. For example, the element 1 is in the second row and sec-

6. Scalar can be used while counting money, weight, height, etc. While vector used while counting speed, gravity, etc. Basically anything that has direction and magnitude can be counted as vector while anything that only has magnitude can be counted as scalar.