



12/9/2023

Project Title:

**Analysis of Cricket Tournament for
select players for an upcoming league
match based on their fitness.**

Name: P.Sairam

Branch: CSE

Year: 4th

Project Guide

Mr. Lokesh B. Sir



Problem Statement

A panel wants to select players for an upcoming league match based on their fitness. Players from all significant cricket clubs have participated in a practice match, and their data is collected. Let us now explore NumPy features using the player's data.

Tools:

1. Jupyter Notebook.
2. Python Programming Language.
3. Numpy (Python Library_)
4. Google Colab.

▾ Multidimensional Arrays

A multidimensional array is an array of arrays. For example, a two dimensional array would be an array with each element as one-dimensional array.

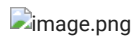
1-D array : [1, 2, 3, 4, 5]

2-D array : [[1, 2, 3, 4, 5], [6, 7, 8, 9, 10]]

Similar for a n-dimensional array

In NumPy, dimension is called axis. In Numpy terminology, for 2-D arrays:

- axis = 0 refers to the rows
- axis = 1 refers to the columns



Multidimensional arrays are indexed using as many indices as the number of dimensions or axes. For instance, to index a 2-D array, you need two indices - array[x, y]. Each axes has an index starting at 0.

```
# import numpy library
import numpy as np

# player information is provided in the lists - players, skills
# players: list of tuples where 1st element is height in inches and the 2nd element is weight in lbs
# skills: the skill of the player in the sport cricket

players = [(74, 180), (74, 215), (72, 210), (72, 210), (73, 188), (69, 176), (69, 209), (71, 200), (76, 231), (71, 180), (73, 188), (73, 180)]
skills = np.array(['Keeper', 'Batsman', 'Bowler', 'Keeper-Batsman', 'Batsman', 'Keeper-Batsman', 'Batsman', 'Batsman', 'Batsman', 'Bowler', ''])

# Creating a 2-D array using the list of tuples
np_players = np.array(players)

# Checking the created array
np_players

array([[ 74, 180],
       [ 74, 215],
       [ 72, 210],
       ...,
       [ 75, 205],
       [ 75, 190],
       [ 73, 195]])

# checking the type of created object
type(np_players)

numpy.ndarray

# printing the number of columns and rows in the array
np_players.shape

(1015, 2)

# printing the dimensions of the array
np_players.ndim

2

# printing the data type of the elements in the array
np_players.dtype

dtype('int64')
```

▼ Slicing a 2D array

```
# printing the entire second row of the 2-D array
np_players[:2]
```

```
array([[ 74, 180],
       [ 74, 215]])
```

```
# printing the second column of the second row of the 2-D array
np_players[2][1]
```

```
210
```

```
# printing the first column (height of the players) with all the rows of the 2-D array
np_players[1:]
```

```
↳ array([[ 74, 215],
        [ 72, 210],
        [ 72, 210],
        ...,
        [ 75, 205],
        [ 75, 190],
        [ 73, 195]])
```

```
# printing only those rows where height of the player is more than 75 inches
import numpy as np
```

```
np_players[np_players[:,0]>75]
```

```
array([[ 76, 231],
       [ 78, 219],
       [ 79, 230],
       [ 76, 205],
       [ 76, 195],
       [ 77, 203],
       [ 78, 200],
       [ 76, 200],
       [ 77, 220],
       [ 76, 212],
       [ 76, 224],
       [ 78, 210],
       [ 76, 195],
       [ 77, 200],
       [ 81, 260],
       [ 78, 228],
       [ 77, 200],
       [ 76, 190],
       [ 76, 230],
       [ 78, 235],
       [ 76, 190],
       [ 79, 205],
       [ 76, 205],
       [ 76, 192],
       [ 77, 235],
       [ 77, 222],
       [ 76, 230],
       [ 80, 220],
       [ 78, 210],
       [ 76, 200],
       [ 76, 210],
       [ 76, 220],
       [ 79, 290],
       [ 76, 200],
       [ 76, 170],
       [ 78, 220],
       [ 76, 220],
       [ 76, 250],
       [ 78, 210],
       [ 77, 220],
       [ 76, 237],
       [ 76, 235],
       [ 76, 195],
       [ 77, 185],
       [ 77, 241],
       [ 77, 225],
       [ 78, 230],
       [ 76, 215],
       [ 79, 215],
```

```
[ 77, 230],
[ 76, 240],
[ 76, 225],
[ 77, 200],
[ 77, 210],
[ 78, 220],
[ 78, 210],
[ 76, 212],
[ 78, 225],
```

```
(np_players[np_players[:,0]>75]).size
```

```
414
```

▼ Slicing one array based on the other

```
# printing those rows where the skill of the player is 'Batsman'
np_players[skills=="Batsman"]
```

```
[ 75, 219],
[ 76, 225],
[ 72, 176],
[ 72, 180],
[ 73, 220],
[ 74, 190],
[ 74, 205],
[ 75, 205],
[ 73, 211],
[ 74, 200],
[ 77, 210],
[ 78, 220],
[ 78, 210],
[ 74, 202],
[ 78, 225],
[ 74, 220],
[ 72, 185],
[ 76, 195],
[ 73, 219],
[ 73, 195],
[ 77, 220],
[ 77, 235],
[ 71, 180],
[ 78, 230],
[ 70, 190],
[ 74, 190],
[ 72, 200],
[ 73, 184],
[ 75, 200],
[ 74, 219],
[ 73, 200],
[ 74, 220],
[ 73, 160],
[ 78, 200],
[ 76, 214],
[ 73, 200],
[ 75, 180],
[ 78, 235],
[ 77, 240],
[ 72, 185],
[ 69, 165],
[ 74, 185],
[ 70, 170],
[ 74, 210],
[ 76, 205],
[ 75, 200],
[ 73, 190],
[ 73, 160],
[ 73, 180],
[ 74, 180],
[ 74, 200],
[ 70, 190],
[ 74, 227],
[ 75, 199],
[ 72, 185],
[ 77, 240],
[ 73, 225],
[ 76, 230],
```

✓ 0s completed at 12:26 PM

● ×