

101 NumPy Exercises for Data Analysis (Python)

The goal of the numpy exercises is to serve as a reference as well as to get you to apply numpy beyond the basics. The questions are of 4 levels of difficulties with L1 being the easiest to L4 being the hardest.



Numpy Tutorial Part 2: Vital Functions for Data Analysis. Photo by Ana Justin Luebke.

If you want a quick refresher on numpy, the [numpy basics](#) (<https://www.machinelearningplus.com/numpy-tutorial-part1-array-python-examples>) and the [advanced numpy tutorials](#) (<https://www.machinelearningplus.com/numpy-tutorial-python-part2>) might be what you are looking for.

Update: A similar set of on [practice exercises with pandas](#) (<https://www.machinelearningplus.com/python/101-pandas-exercises-python/>) is now available.

1. Import numpy as np and see the version

Difficulty Level: L1

Q. Import numpy as `np` and print the version number.

Show Solution

```
import numpy as np
print(np.__version__)
#> 1.13.3
```

You *must* import numpy as np for the rest of the codes in this exercise to work.

To install numpy its recommended to use the installation provided by [anaconda](#). (<https://www.anaconda.com/download/#macos>).

2. How to create a 1D array?

Difficulty Level: L1

Q. Create a 1D array of numbers from 0 to 9

Desired output:

```
#> array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])
```

Show Solution

```
arr = np.arange(10)
arr
#> array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])
```

Feedback

Difficulty Level: L1

Q. Create a 3×3 numpy array of all True's

Show Solution

```
np.full((3, 3), True, dtype=bool)
#> array([[ True,  True,  True],
#>        [ True,  True,  True],
#>        [ True,  True,  True]], dtype=bool)

# Alternate method:
np.ones((3,3), dtype=bool)
```

4. How to extract items that satisfy a given condition from 1D array?

Difficulty Level: L1

Q. Extract all odd numbers from arr

Input:

```
arr = np.array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])
```

Desired output:

```
#> array([1, 3, 5, 7, 9])
```

Show Solution

```
# Input
arr = np.array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])

# Solution
arr[arr % 2 == 1]
#> array([1, 3, 5, 7, 9])
```

Feedback

5. How to replace items that satisfy a condition with another value in numpy array?

Difficulty Level: L1

Q. Replace all odd numbers in arr with -1

Input:

```
arr = np.array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])
```

Desired Output:

```
#> array([ 0, -1,  2, -1,  4, -1,  6, -1,  8, -1])
```

Show Solution

```
arr[arr % 2 == 1] = -1  
arr  
#> array([ 0, -1,  2, -1,  4, -1,  6, -1,  8, -1])
```

6. How to replace items that satisfy a condition without affecting the original array?

Difficulty Level: L2

Q. Replace all odd numbers in arr with -1 without changing arr

Input:

```
arr = np.array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])
```

Desired Output:

Feedback

```
out  
#> array([ 0, -1,  2, -1,  4, -1,  6, -1,  8, -1])  
arr  
#> array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])
```

Show Solution

```
arr = np.arange(10)  
out = np.where(arr % 2 == 1, -1, arr)  
print(arr)  
out  
#> [0 1 2 3 4 5 6 7 8 9]  
array([ 0, -1,  2, -1,  4, -1,  6, -1,  8, -1])
```

7. How to reshape an array?

Difficulty Level: L1

Q. Convert a 1D array to a 2D array with 2 rows

Input:

```
np.arange(10)  
#> array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9])
```

Desired Output:

```
#> array([[0, 1, 2, 3, 4],  
#>         [5, 6, 7, 8, 9]])
```

Show Solution

```
arr = np.arange(10)  
arr.reshape(2, -1) # Setting to -1 automatically decides the number of cols  
#> array([[0, 1, 2, 3, 4],  
#>         [5, 6, 7, 8, 9]])
```

8. How to stack two arrays vertically?

Input

```
a = np.arange(10).reshape(2,-1)
b = np.repeat(1, 10).reshape(2,-1)
```

Desired Output:

```
#> array([[0, 1, 2, 3, 4],
#>        [5, 6, 7, 8, 9],
#>        [1, 1, 1, 1, 1],
#>        [1, 1, 1, 1, 1]])
```

Show Solution

```
a = np.arange(10).reshape(2,-1)
b = np.repeat(1, 10).reshape(2,-1)

# Answers
# Method 1:
np.concatenate([a, b], axis=0)

# Method 2:
np.vstack([a, b])

# Method 3:
np.r_[a, b]
#> array([[0, 1, 2, 3, 4],
#>        [5, 6, 7, 8, 9],
#>        [1, 1, 1, 1, 1],
#>        [1, 1, 1, 1, 1]])
```

9. How to stack two arrays horizontally?

Difficulty Level: L2

Q. Stack the arrays `a` and `b` horizontally.

Input

```
a = np.arange(10).reshape(2,-1)
```

```
b = np.repeat(1, 10).reshape(2,-1)
```

Desired Output:

```
#> array([[0, 1, 2, 3, 4, 1, 1, 1, 1, 1],
#>        [5, 6, 7, 8, 9, 1, 1, 1, 1, 1]])
```

Show Solution

```
a = np.arange(10).reshape(2,-1)
b = np.repeat(1, 10).reshape(2,-1)
```

Answers

Method 1:

```
np.concatenate([a, b], axis=1)
```

Method 2:

```
np.hstack([a, b])
```

Method 3:

```
np.c_[a, b]
```

```
#> array([[0, 1, 2, 3, 4, 1, 1, 1, 1, 1],
#>        [5, 6, 7, 8, 9, 1, 1, 1, 1, 1]])
```

10. How to generate custom sequences in numpy without hardcoded?

Difficulty Level: L2

Q. Create the following pattern without hardcoded. Use only numpy functions and the below input array `a`.

Input:

```
a = np.array([1,2,3])`
```

Desired Output:

```
#> array([1, 1, 1, 2, 2, 2, 3, 3, 3, 1, 2, 3, 1, 2, 3, 1, 2, 3])
```

Feedback

```
np.r_[np.repeat(a, 3), np.tile(a, 3)]
#> array([1, 1, 1, 2, 2, 2, 3, 3, 3, 1, 2, 3, 1, 2, 3, 1, 2, 3])
```

11. How to get the common items between two python numpy arrays?

Difficulty Level: L2

Q. Get the common items between a and b

Input:

```
a = np.array([1,2,3,2,3,4,3,4,5,6])
b = np.array([7,2,10,2,7,4,9,4,9,8])
```

Desired Output:

```
array([2, 4])
```

```
a = np.array([1,2,3,2,3,4,3,4,5,6])
b = np.array([7,2,10,2,7,4,9,4,9,8])
np.intersect1d(a,b)
#> array([2, 4])
```

12. How to remove from one array those items that exist in another?

Difficulty Level: L2

Q. From array a remove all items present in array b

Input:

```
a = np.array([1,2,3,4,5])
b = np.array([5,6,7,8,9])
```

Desired Output:

```
array([1,2,3,4])
```

Show Solution

```
a = np.array([1,2,3,4,5])
b = np.array([5,6,7,8,9])

# From 'a' remove all of 'b'
np.setdiff1d(a,b)
#> array([1, 2, 3, 4])
```

13. How to get the positions where elements of two arrays match?

Difficulty Level: L2

Q. Get the positions where elements of a and b match

Input:

```
a = np.array([1,2,3,2,3,4,3,4,5,6])
b = np.array([7,2,10,2,7,4,9,4,9,8])
```

Desired Output:

```
#> (array([1, 3, 5, 7]),)
```

Show Solution

```
a = np.array([1,2,3,2,3,4,3,4,5,6])
b = np.array([7,2,10,2,7,4,9,4,9,8])

np.where(a == b)
#> (array([1, 3, 5, 7]),)
```

14. How to extract all numbers between a given range from a numpy array?

[Feedback](#)

Difficulty Level: L2

Q. Get all items between 5 and 10 from `a`.

Input:

```
a = np.array([2, 6, 1, 9, 10, 3, 27])
```

Desired Output:

```
(array([6, 9, 10]),)
```

Show Solution

15. How to make a python function that handles scalars to work on numpy arrays?

Difficulty Level: L2

Q. Convert the function `maxx` that works on two scalars, to work on two arrays.

Input:

```
def maxx(x, y):
    """Get the maximum of two items"""
    if x >= y:
        return x
    else:
        return y

maxx(1, 5)
#> 5
```

Desired Output:

```
a = np.array([5, 7, 9, 8, 6, 4, 5])
b = np.array([6, 3, 4, 8, 9, 7, 1])
pair_max(a, b)
#> array([ 6.,  7.,  9.,  8.,  9.,  7.,  5.])
```

Feedback

Show Solution

16. How to swap two columns in a 2d numpy array?

Difficulty Level: L2

Q. Swap columns 1 and 2 in the array arr.

```
arr = np.arange(9).reshape(3,3)  
arr
```

Show Solution

17. How to swap two rows in a 2d numpy array?

Difficulty Level: L2

Q. Swap rows 1 and 2 in the array arr:

```
arr = np.arange(9).reshape(3,3)  
arr
```

Show Solution

18. How to reverse the rows of a 2D array?

Difficulty Level: L2

Q. Reverse the rows of a 2D array arr.

```
# Input  
arr = np.arange(9).reshape(3,3)
```

Show Solution

19. How to reverse the columns of a 2D array?

Feedback

Difficulty Level: L2

Q. Reverse the columns of a 2D array arr.

```
# Input  
arr = np.arange(9).reshape(3,3)
```

Show Solution

20. How to create a 2D array containing random floats between 5 and 10?

Difficulty Level: L2

Q. Create a 2D array of shape 5x3 to contain random decimal numbers between 5 and 10.

Show Solution

21. How to print only 3 decimal places in python numpy array?

Difficulty Level: L1

Q. Print or show only 3 decimal places of the numpy array rand_arr.

Input:

```
rand_arr = np.random.random((5,3))
```

Show Solution

Feedback

```
# Input
rand_arr = np.random.random((5,3))

# Create the random array
rand_arr = np.random.random([5,3])

# Limit to 3 decimal places
np.set_printoptions(precision=3)
rand_arr[:4]
#> array([[ 0.443,  0.109,  0.97 ],
#>        [ 0.388,  0.447,  0.191],
#>        [ 0.891,  0.474,  0.212],
#>        [ 0.609,  0.518,  0.403]])
```

22. How to pretty print a numpy array by suppressing the scientific notation (like 1e10)?

Difficulty Level: L1

Q. Pretty print rand_arr by suppressing the scientific notation (like 1e10)

Input:

```
# Create the random array
np.random.seed(100)
rand_arr = np.random.random([3,3])/1e3
rand_arr
#> array([[ 5.434049e-04,  2.783694e-04,  4.245176e-04],
#>        [ 8.447761e-04,  4.718856e-06,  1.215691e-04],
#>        [ 6.707491e-04,  8.258528e-04,  1.367066e-04]])
```

Desired Output:

```
#> array([[ 0.000543,  0.000278,  0.000425],
#>        [ 0.000845,  0.000005,  0.000122],
#>        [ 0.000671,  0.000826,  0.000137]])
```

Show Solution

Feedback

23. How to limit the number of items printed in output of numpy array?

Difficulty Level: L1

Q. Limit the number of items printed in python numpy array `a` to a maximum of 6 elements.

Input:

```
a = np.arange(15)
#> array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14])
```

Desired Output:

```
#> array([ 0,  1,  2, ..., 12, 13, 14])
```

Show Solution

```
np.set_printoptions(threshold=6)
a = np.arange(15)
a
#> array([ 0,  1,  2, ..., 12, 13, 14])
```

24. How to print the full numpy array without truncating

Difficulty Level: L1

Q. Print the full numpy array `a` without truncating.

Input:

```
np.set_printoptions(threshold=6)
a = np.arange(15)
a
#> array([ 0,  1,  2, ..., 12, 13, 14])
```

Desired Output:

```
a
#> array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14])
```

Feedback

Show Solution

25. How to import a dataset with numbers and texts keeping the text intact in python numpy?

Difficulty Level: L2

Q. Import the [iris dataset](https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data) (<https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data>) keeping the text intact.

Show Solution

```
# Solution
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data' (htt
ps://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'
iris = np.genfromtxt(url, delimiter=',', dtype='object')
names = ('sepallength', 'sepalwidth', 'petallength', 'petalwidth', 'species')

# Print the first 3 rows
iris[:3]
#> array([[b'5.1', b'3.5', b'1.4', b'0.2', b'Iris-setosa'],
#>        [b'4.9', b'3.0', b'1.4', b'0.2', b'Iris-setosa'],
#>        [b'4.7', b'3.2', b'1.3', b'0.2', b'Iris-setosa']], dtype=object)
```

Since we want to retain the species, a text field, I have set the `dtype` to `object`. Had I set `dtype=None`, a 1d array of tuples would have been returned.

26. How to extract a particular column from 1D array of tuples?

Difficulty Level: L2

Q. Extract the text column `species` from the 1D `iris` imported in previous question.

Input:

```
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data' (http
s://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'
iris_1d = np.genfromtxt(url, delimiter=',', dtype=None)
```

Show Solution

Feedback

```
# Input:  
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data (http  
s://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'  
iris_1d = np.genfromtxt(url, delimiter=',', dtype=None)  
print(iris_1d.shape)  
  
# Solution:  
species = np.array([row[4] for row in iris_1d])  
species[:5]  
#> (150,)  
#> array([b'Iris-setosa', b'Iris-setosa', b'Iris-setosa', b'Iris-setosa',  
#>         b'Iris-setosa'],  
#>         dtype='|S18')
```

27. How to convert a 1d array of tuples to a 2d numpy array?

Difficulty Level: L2

Q. Convert the 1D `iris` to 2D array `iris_2d` by omitting the `species` text field.

Input:

```
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data (http  
s://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'  
iris_1d = np.genfromtxt(url, delimiter=',', dtype=None)
```

Show Solution

Feedback

```

# Input:
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data_(http
ps://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'
iris_1d = np.genfromtxt(url, delimiter=',', dtype=None)

# Solution:
# Method 1: Convert each row to a list and get the first 4 items
iris_2d = np.array([row.tolist()[:4] for row in iris_1d])
iris_2d[:4]

# Alt Method 2: Import only the first 4 columns from source url
iris_2d = np.genfromtxt(url, delimiter=',', dtype='float', usecols=[0,1,2,3])
iris_2d[:4]
#> array([[ 5.1,  3.5,  1.4,  0.2],
#>        [ 4.9,  3. ,  1.4,  0.2],
#>        [ 4.7,  3.2,  1.3,  0.2],
#>        [ 4.6,  3.1,  1.5,  0.2]])

```

28. How to compute the mean, median, standard deviation of a numpy array?

Difficulty: L1

Q. Find the mean, median, standard deviation of iris's sepallength (1st column)

```

url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data_(http
ps://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'
iris = np.genfromtxt(url, delimiter=',', dtype='object')

```

Show Solution

```

# Input
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data_(http
ps://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'
iris = np.genfromtxt(url, delimiter=',', dtype='object')
sepallength = np.genfromtxt(url, delimiter=',', dtype='float', usecols=[0])

# Solution
mu, med, sd = np.mean(sepallength), np.median(sepallength), np.std(sepallength)
print(mu, med, sd)
#> 5.84333333333 5.8 0.825301291785

```

29. How to normalize an array so the values range exactly between 0 and 1?

Difficulty: L2

Q. Create a normalized form of `iris`'s `sepallength` whose values range exactly between 0 and 1 so that the minimum has value 0 and maximum has value 1.

Input:

```
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data' (http://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'
sepallength = np.genfromtxt(url, delimiter=',', dtype='float', usecols=[0])
```

Show Solution

```
# Input
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data' (http://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'
sepallength = np.genfromtxt(url, delimiter=',', dtype='float', usecols=[0])

# Solution
Smax, Smin = sepallength.max(), sepallength.min()
S = (sepallength - Smin)/(Smax - Smin)
# or
S = (sepallength - Smin)/sepallength.ptp() # Thanks, David Ojeda!
print(S)
#> [ 0.222  0.167  0.111  0.083  0.194  0.306  0.083  0.194  0.028  0.167
#>   0.306  0.139  0.139  0.        0.417  0.389  0.306  0.222  0.389  0.222
#>   0.306  0.222  0.083  0.222  0.139  0.194  0.194  0.25    0.25    0.111
#>   0.139  0.306  0.25    0.333  0.167  0.194  0.333  0.167  0.028  0.222
#>   0.194  0.056  0.028  0.194  0.222  0.139  0.222  0.083  0.278  0.194
#>   0.75   0.583  0.722  0.333  0.611  0.389  0.556  0.167  0.639  0.25
#>   0.194  0.444  0.472  0.5     0.361  0.667  0.361  0.417  0.528  0.361
#>   0.444  0.5     0.556  0.5     0.583  0.639  0.694  0.667  0.472  0.389
#>   0.333  0.333  0.417  0.472  0.306  0.472  0.667  0.556  0.361  0.333
#>   0.333  0.5     0.417  0.194  0.361  0.389  0.389  0.528  0.222  0.389
#>   0.556  0.417  0.778  0.556  0.611  0.917  0.167  0.833  0.667  0.806
#>   0.611  0.583  0.694  0.389  0.417  0.583  0.611  0.944  0.944  0.472
#>   0.722  0.361  0.944  0.556  0.667  0.806  0.528  0.5     0.583  0.806
#>   0.861  1.      0.583  0.556  0.5     0.944  0.556  0.583  0.472  0.722
#>   0.667  0.722  0.417  0.694  0.667  0.667  0.556  0.611  0.528  0.444]
```

30. How to compute the softmax score?

Difficulty Level: L3

Q. Compute the softmax score of `sepallength`.

```
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data' (http  
s://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'  
sepallength = np.genfromtxt(url, delimiter=',', dtype='float', usecols=[0])
```

Show Solution

```
# Input  
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data' (http  
s://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'  
iris = np.genfromtxt(url, delimiter=',', dtype='object')  
sepallength = np.array([float(row[0]) for row in iris])  
  
# Solution  
def softmax(x):  
    """Compute softmax values for each sets of scores in x.  
    https://stackoverflow.com/questions/34968722/how-to-implement-the-softmax-functi  
on-in-python (https://stackoverflow.com/questions/34968722/how-to-implement-the-soft  
max-function-in-python)"""  
    e_x = np.exp(x - np.max(x))  
    return e_x / e_x.sum(axis=0)  
  
print(softmax(sepallength))  
#> [ 0.002  0.002  0.001  0.001  0.002  0.003  0.001  0.002  0.001  0.002  
#>  0.003  0.002  0.002  0.001  0.004  0.004  0.003  0.002  0.004  0.002  
#>  0.003  0.002  0.001  0.002  0.002  0.002  0.002  0.002  0.002  0.001  
#>  0.002  0.003  0.002  0.003  0.002  0.002  0.003  0.002  0.001  0.002  
#>  0.002  0.001  0.001  0.002  0.002  0.002  0.002  0.001  0.003  0.002  
#>  0.015  0.008  0.013  0.003  0.009  0.004  0.007  0.002  0.01  0.002  
#>  0.002  0.005  0.005  0.006  0.004  0.011  0.004  0.004  0.007  0.004  
#>  0.005  0.006  0.007  0.006  0.008  0.01  0.012  0.011  0.005  0.004  
#>  0.003  0.003  0.004  0.005  0.003  0.005  0.011  0.007  0.004  0.003  
#>  0.003  0.006  0.004  0.002  0.004  0.004  0.004  0.007  0.002  0.004  
#>  0.007  0.004  0.016  0.007  0.009  0.027  0.002  0.02  0.011  0.018  
#>  0.009  0.008  0.012  0.004  0.004  0.008  0.009  0.03  0.03  0.005  
#>  0.013  0.004  0.03  0.007  0.011  0.018  0.007  0.006  0.008  0.018  
#>  0.022  0.037  0.008  0.007  0.006  0.03  0.007  0.008  0.005  0.013  
#>  0.011  0.013  0.004  0.012  0.011  0.011  0.007  0.009  0.007  0.005]
```

31. How to find the percentile scores of a numpy array?

Difficulty Level: L1

Q. Find the 5th and 95th percentile of iris's sepallength

```
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data (http  
s://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'  
sepallength = np.genfromtxt(url, delimiter=',', dtype='float', usecols=[0])
```

Show Solution

```
# Input  
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data (htt  
ps://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'  
sepallength = np.genfromtxt(url, delimiter=',', dtype='float', usecols=[0])  
  
# Solution  
np.percentile(sepallength, q=[5, 95])  
#> array([ 4.6 , 7.255])
```

32. How to insert values at random positions in an array?

Difficulty Level: L2

Q. Insert np.nan values at 20 random positions in iris_2d dataset

```
# Input  
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data (http  
s://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'  
iris_2d = np.genfromtxt(url, delimiter=',', dtype='object')
```

Show Solution

```

# Input
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data_(http
s://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'
iris_2d = np.genfromtxt(url, delimiter=',', dtype='object')

# Method 1
i, j = np.where(iris_2d)

# i, j contain the row numbers and column numbers of 600 elements of iris_x
np.random.seed(100)
iris_2d[np.random.choice((i), 20), np.random.choice((j), 20)] = np.nan

# Method 2
np.random.seed(100)
iris_2d[np.random.randint(150, size=20), np.random.randint(4, size=20)] = np.nan

# Print first 10 rows
print(iris_2d[:10])
#> [[b'5.1' b'3.5' b'1.4' b'0.2' b'Iris-setosa']]
#> [b'4.9' b'3.0' b'1.4' b'0.2' b'Iris-setosa']
#> [b'4.7' b'3.2' b'1.3' b'0.2' b'Iris-setosa']
#> [b'4.6' b'3.1' b'1.5' b'0.2' b'Iris-setosa']
#> [b'5.0' b'3.6' b'1.4' b'0.2' b'Iris-setosa']
#> [b'5.4' b'3.9' b'1.7' b'0.4' b'Iris-setosa']
#> [b'4.6' b'3.4' b'1.4' b'0.3' b'Iris-setosa']
#> [b'5.0' b'3.4' b'1.5' b'0.2' b'Iris-setosa']
#> [b'4.4' nan b'1.4' b'0.2' b'Iris-setosa']
#> [b'4.9' b'3.1' b'1.5' b'0.1' b'Iris-setosa']]
```

33. How to find the position of missing values in numpy array?

Difficulty Level: L2

Q. Find the number and position of missing values in `iris_2d`'s `sepallength` (1st column)

```

# Input
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data_(http
s://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'
iris_2d = np.genfromtxt(url, delimiter=',', dtype='float')
iris_2d[np.random.randint(150, size=20), np.random.randint(4, size=20)] = np.nan
```

Show Solution

Feedback

```

# Input
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data_(http
s://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'
iris_2d = np.genfromtxt(url, delimiter=',', dtype='float', usecols=[0,1,2,3])
iris_2d[np.random.randint(150, size=20), np.random.randint(4, size=20)] = np.nan

# Solution
print("Number of missing values: \n", np.isnan(iris_2d[:, 0]).sum())
print("Position of missing values: \n", np.where(np.isnan(iris_2d[:, 0])))
#> Number of missing values:
#> 5
#> Position of missing values:
#> (array([ 39,  88,  99, 130, 147]),)

```

34. How to filter a numpy array based on two or more conditions?

Difficulty Level: L3

Q. Filter the rows of `iris_2d` that has `petallength` (3rd column) > 1.5 and `sepallength` (1st column) < 5.0

```

# Input
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data_(http
s://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'
iris_2d = np.genfromtxt(url, delimiter=',', dtype='float', usecols=[0,1,2,3])

```

Show Solution

```

# Input
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data_(htt
ps://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'
iris_2d = np.genfromtxt(url, delimiter=',', dtype='float', usecols=[0,1,2,3])

# Solution
condition = (iris_2d[:, 2] > 1.5) & (iris_2d[:, 0] < 5.0)
iris_2d[condition]
#> array([[ 4.8,  3.4,  1.6,  0.2],
#>        [ 4.8,  3.4,  1.9,  0.2],
#>        [ 4.7,  3.2,  1.6,  0.2],
#>        [ 4.8,  3.1,  1.6,  0.2],
#>        [ 4.9,  2.4,  3.3,  1. ],
#>        [ 4.9,  2.5,  4.5,  1.7]])

```

[Feedback](#)

35. How to drop rows that contain a missing value from a numpy array?

Difficulty Level: L3:

Q. Select the rows of iris_2d that does not have any nan value.

```
# Input
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data_(http
s://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'
iris_2d = np.genfromtxt(url, delimiter=',', dtype='float', usecols=[0,1,2,3])
```

Show Solution

```
# Input
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data_(htt
ps://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'
iris_2d = np.genfromtxt(url, delimiter=',', dtype='float', usecols=[0,1,2,3])
iris_2d[np.random.randint(150, size=20), np.random.randint(4, size=20)] = np.nan

# Solution
# No direct numpy function for this.

# Method 1:
any_nan_in_row = np.array([~np.any(np.isnan(row)) for row in iris_2d])
iris_2d[any_nan_in_row][:5]

# Method 2: (By Rong)
iris_2d[np.sum(np.isnan(iris_2d), axis = 1) == 0][:5]
#> array([[ 4.9,  3. ,  1.4,  0.2],
#>        [ 4.7,  3.2,  1.3,  0.2],
#>        [ 4.6,  3.1,  1.5,  0.2],
#>        [ 5. ,  3.6,  1.4,  0.2],
#>        [ 5.4,  3.9,  1.7,  0.4]])
```

36. How to find the correlation between two columns of a numpy array?

Difficulty Level: L2

Q. Find the correlation between SepalLength[1st column] and PetalLength[3rd column] in iris_2d

```
# Input
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data_(http
s://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'
iris_2d = np.genfromtxt(url, delimiter=',', dtype='float', usecols=[0,1,2,3])
```

Show Solution

```
# Input
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data_(htt
ps://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'
iris = np.genfromtxt(url, delimiter=',', dtype='float', usecols=[0,1,2,3])

# Solution 1
np.corrcoef(iris[:, 0], iris[:, 2])[0, 1]

# Solution 2
from scipy.stats.stats import pearsonr
corr, p_value = pearsonr(iris[:, 0], iris[:, 2])
print(corr)

# Correlation coef indicates the degree of linear relationship between two numeric v
ariables.
# It can range between -1 to +1.

# The p-value roughly indicates the probability of an uncorrelated system producing
# datasets that have a correlation at least as extreme as the one computed.
# The lower the p-value (<0.01), stronger is the significance of the relationship.
# It is not an indicator of the strength.
#> 0.871754157305
```

37. How to find if a given array has any null values?

Difficulty Level: L2

Feedback

Q. Find out if `iris_2d` has any missing values.

```
# Input
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data_(http
s://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'
iris_2d = np.genfromtxt(url, delimiter=',', dtype='float', usecols=[0,1,2,3])
```

Show Solution

```
# Input
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data_(htt
ps://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'
iris_2d = np.genfromtxt(url, delimiter=',', dtype='float', usecols=[0,1,2,3])

np.isnan(iris_2d).any()
#> False
```

38. How to replace all missing values with 0 in a numpy array?

Difficulty Level: L2

Q. Replace all occurrences of `nan` with 0 in numpy array

```
# Input
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data_(http
s://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'
iris_2d = np.genfromtxt(url, delimiter=',', dtype='float', usecols=[0,1,2,3])
iris_2d[np.random.randint(150, size=20), np.random.randint(4, size=20)] = np.nan
```

Show Solution

Feedback

```

# Input
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data (http
ps://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'
iris_2d = np.genfromtxt(url, delimiter=',', dtype='float', usecols=[0,1,2,3])
iris_2d[np.random.randint(150, size=20), np.random.randint(4, size=20)] = np.nan

# Solution
iris_2d[np.isnan(iris_2d)] = 0
iris_2d[:4]
#> array([[ 5.1,  3.5,  1.4,  0. ],
#>        [ 4.9,  3. ,  1.4,  0.2],
#>        [ 4.7,  3.2,  1.3,  0.2],
#>        [ 4.6,  3.1,  1.5,  0.2]])

```

39. How to find the count of unique values in a numpy array?

Difficulty Level: L2

Q. Find the unique values and the count of unique values in iris's species

```

# Input
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data (http
ps://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'
iris = np.genfromtxt(url, delimiter=',', dtype='object')
names = ('sepal length', 'sepal width', 'petal length', 'petal width', 'species')

```

Show Solution

```

# Import iris keeping the text column intact
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data (http
ps://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'
iris = np.genfromtxt(url, delimiter=',', dtype='object')
names = ('sepal length', 'sepal width', 'petal length', 'petal width', 'species')

# Solution
# Extract the species column as an array
species = np.array([row.tolist()[4] for row in iris])

# Get the unique values and the counts
np.unique(species, return_counts=True)
#> (array(['Iris-setosa', 'Iris-versicolor', 'Iris-virginica'],
#>       dtype='|S15'), array([50, 50, 50]))

```

[Feedback](#)

40. How to convert a numeric to a categorical (text) array?

Difficulty Level: L2

Q. Bin the petal length [3rd] column of iris_2d to form a text array, such that if petal length is:

- Less than 3 --> 'small'
- 3-5 --> 'medium'
- '>=5 --> 'large'

```
# Input
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data_(http
s://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'
iris = np.genfromtxt(url, delimiter=',', dtype='object')
names = ('sepallength', 'sepalwidth', 'petallength', 'petalwidth', 'species')
```

Show Solution

```
# Input
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data_(htt
ps://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'
iris = np.genfromtxt(url, delimiter=',', dtype='object')
names = ('sepallength', 'sepalwidth', 'petallength', 'petalwidth', 'species')

# Bin petallength
petal_length_bin = np.digitize(iris[:, 2].astype('float'), [0, 3, 5, 10])

# Map it to respective category
label_map = {1: 'small', 2: 'medium', 3: 'large', 4: np.nan}
petal_length_cat = [label_map[x] for x in petal_length_bin]

# View
petal_length_cat[:4]
<#> ['small', 'small', 'small', 'small']
```

41. How to create a new column from existing columns of a numpy array?

Difficulty Level: L2

Q. Create a new column for volume in iris_2d, where volume is $(\pi \times \text{petallength} \times \text{sepal_length}^2)/3$

Feedback

```
# Input
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data' (http
s://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'
iris_2d = np.genfromtxt(url, delimiter=',', dtype='object')
names = ('sepallength', 'sepalwidth', 'petallength', 'petalwidth', 'species')
```

Show Solution

```
# Input
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data' (htt
ps://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'
iris_2d = np.genfromtxt(url, delimiter=',', dtype='object')

# Solution
# Compute volume
sepallength = iris_2d[:, 0].astype('float')
petallength = iris_2d[:, 2].astype('float')
volume = (np.pi * petallength * (sepallength**2))/3

# Introduce new dimension to match iris_2d's
volume = volume[:, np.newaxis]

# Add the new column
out = np.hstack([iris_2d, volume])

# View
out[:4]
#> array([[b'5.1', b'3.5', b'1.4', b'0.2', b'Iris-setosa', 38.13265162927291],
#>        [b'4.9', b'3.0', b'1.4', b'0.2', b'Iris-setosa', 35.200498485922445],
#>        [b'4.7', b'3.2', b'1.3', b'0.2', b'Iris-setosa', 30.0723720777127],
#>        [b'4.6', b'3.1', b'1.5', b'0.2', b'Iris-setosa', 33.238050274980004]], dty
pe=object)
```

42. How to do probabilistic sampling in numpy?

Difficulty Level: L3

Q. Randomly sample iris's species such that setose is twice the number of versicolor and virginica

[Feedback](#)

```
# Import iris keeping the text column intact
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data_(http
s://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'
iris = np.genfromtxt(url, delimiter=',', dtype='object')
```

Show Solution

```
# Import iris keeping the text column intact
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data_(htt
ps://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'
iris = np.genfromtxt(url, delimiter=',', dtype='object')

# Solution
# Get the species column
species = iris[:, 4]

# Approach 1: Generate Probabilistically
np.random.seed(100)
a = np.array(['Iris-setosa', 'Iris-versicolor', 'Iris-virginica'])
species_out = np.random.choice(a, 150, p=[0.5, 0.25, 0.25])

# Approach 2: Probabilistic Sampling (preferred)
np.random.seed(100)
probs = np.r_[np.linspace(0, 0.500, num=50), np.linspace(0.501, .750, num=50), np.li
nspac(.751, 1.0, num=50)]
index = np.searchsorted(probs, np.random.random(150))
species_out = species[index]
print(np.unique(species_out, return_counts=True))

#> (array(['Iris-setosa', 'Iris-versicolor', 'Iris-virginica'], dtype=object), ar
ray([77, 37, 36]))
```

Approach 2 is preferred because it creates an index variable that can be used to sample 2d tabular data.

43. How to get the second largest value of an array when grouped by another array?

Difficulty Level: L2

Q. What is the value of second longest petallength of species setosa

Feedback

```
# Input
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data' (http
s://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'
iris = np.genfromtxt(url, delimiter=',', dtype='object')
names = ('sepal length', 'sepal width', 'petal length', 'petal width', 'species')
```

Show Solution

```
# Import iris keeping the text column intact
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data' (htt
ps://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'
iris = np.genfromtxt(url, delimiter=',', dtype='object')

# Solution
# Get the species and petal length columns
petal_len_setosa = iris[iris[:, 4] == b'Iris-setosa', [2]].astype('float')

# Get the second last value
np.unique(np.sort(petal_len_setosa))[-2]
#> 1.7
```

44. How to sort a 2D array by a column

Difficulty Level: L2

Q. Sort the iris dataset based on sepal length column.

```
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data' (http
s://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'
iris = np.genfromtxt(url, delimiter=',', dtype='object')
names = ('sepal length', 'sepal width', 'petal length', 'petal width', 'species')
```

Show Solution

Feedback

```
# Sort by column position 0: SepalLength
print(iris[iris[:,0].argsort()][:20])
#> [[b'4.3' b'3.0' b'1.1' b'0.1' b'Iris-setosa']]
#> [b'4.4' b'3.2' b'1.3' b'0.2' b'Iris-setosa']
#> [b'4.4' b'3.0' b'1.3' b'0.2' b'Iris-setosa']
#> [b'4.4' b'2.9' b'1.4' b'0.2' b'Iris-setosa']
#> [b'4.5' b'2.3' b'1.3' b'0.3' b'Iris-setosa']
#> [b'4.6' b'3.6' b'1.0' b'0.2' b'Iris-setosa']
#> [b'4.6' b'3.1' b'1.5' b'0.2' b'Iris-setosa']
#> [b'4.6' b'3.4' b'1.4' b'0.3' b'Iris-setosa']
#> [b'4.6' b'3.2' b'1.4' b'0.2' b'Iris-setosa']
#> [b'4.7' b'3.2' b'1.3' b'0.2' b'Iris-setosa']
#> [b'4.7' b'3.2' b'1.6' b'0.2' b'Iris-setosa']
#> [b'4.8' b'3.0' b'1.4' b'0.1' b'Iris-setosa']
#> [b'4.8' b'3.0' b'1.4' b'0.3' b'Iris-setosa']
#> [b'4.8' b'3.4' b'1.9' b'0.2' b'Iris-setosa']
#> [b'4.8' b'3.4' b'1.6' b'0.2' b'Iris-setosa']
#> [b'4.8' b'3.1' b'1.6' b'0.2' b'Iris-setosa']
#> [b'4.9' b'2.4' b'3.3' b'1.0' b'Iris-versicolor']
#> [b'4.9' b'2.5' b'4.5' b'1.7' b'Iris-virginica']
#> [b'4.9' b'3.1' b'1.5' b'0.1' b'Iris-setosa']
#> [b'4.9' b'3.1' b'1.5' b'0.1' b'Iris-setosa']]
```

45. How to find the most frequent value in a numpy array?

Difficulty Level: L1

Q. Find the most frequent value of petal length [3rd column] in iris dataset.

Input:

```
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data' (http
s://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data')
iris = np.genfromtxt(url, delimiter=',', dtype='object')
names = ('sepallength', 'sepalwidth', 'petallength', 'petalwidth', 'species')
```

Show Solution

Feedback

```

# Input:
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data' (https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)
iris = np.genfromtxt(url, delimiter=',', dtype='object')

# Solution:
vals, counts = np.unique(iris[:, 2], return_counts=True)
print(vals[np.argmax(counts)])
#> b'1.5'

```

46. How to find the position of the first occurrence of a value greater than a given value?

Difficulty Level: L2

Q. Find the position of the first occurrence of a value greater than 1.0 in petalwidth 4th column of iris dataset.

```

# Input:
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data' (https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)
iris = np.genfromtxt(url, delimiter=',', dtype='object')

```

Show Solution

```

# Input:
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data' (https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)
iris = np.genfromtxt(url, delimiter=',', dtype='object')

# Solution: (edit: changed argmax to argwhere. Thanks Rong!)
np.argwhere(iris[:, 3].astype(float) > 1.0)[0]
#> 50

```

47. How to replace all values greater than a given value to a given cutoff?

Difficulty Level: L2

Q. From the array `a`, replace all values greater than 30 to 30 and less than 10 to 10.

[Feedback](#)

Input:

```
np.random.seed(100)
a = np.random.uniform(1,50, 20)
```

Show Solution

```
# Input
np.set_printoptions(precision=2)
np.random.seed(100)
a = np.random.uniform(1,50, 20)

# Solution 1: Using np.clip
np.clip(a, a_min=10, a_max=30)

# Solution 2: Using np.where
print(np.where(a < 10, 10, np.where(a > 30, 30, a)))
#> [ 27.63  14.64  21.8   30.    10.    10.    30.    30.    10.    29.18  30.
#>   11.25  10.08  10.    11.77  30.    30.    10.    30.    14.43]
```

48. How to get the positions of top n values from a numpy array?

Difficulty Level: L2

Q. Get the positions of top 5 maximum values in a given array a .

```
np.random.seed(100)
a = np.random.uniform(1,50, 20)
```

Show Solution

Feedback

```

# Input
np.random.seed(100)
a = np.random.uniform(1,50, 20)

# Solution:
print(a.argsort())
#> [18 7 3 10 15]

# Solution 2:
np.argpartition(-a, 5)[:5]
#> [15 10 3 7 18]

# Below methods will get you the values.

# Method 1:
a[a.argsort()][-5:]

# Method 2:
np.sort(a)[-5:]

# Method 3:
np.partition(a, kth=-5)[-5:]

# Method 4:
a[np.argpartition(-a, 5)][:5]

```

49. How to compute the row wise counts of all possible values in an array?

Difficulty Level: L4

Q. Compute the counts of unique values row-wise.

Input:

```

np.random.seed(100)
arr = np.random.randint(1,11,size=(6, 10))
arr
> array([[ 9,  9,  4,  8,  8,  1,  5,  3,  6,  3],
>          [ 3,  3,  2,  1,  9,  5,  1, 10,  7,  3],
>          [ 5,  2,  6,  4,  5,  5,  4,  8,  2,  2],
>          [ 8,  8,  1,  3, 10, 10,  4,  3,  6,  9],
>          [ 2,  1,  8,  7,  3,  1,  9,  3,  6,  2],
>          [ 9,  2,  6,  5,  3,  9,  4,  6,  1, 10]])

```

[Feedback](#)

Desired Output:

```
> [[1, 0, 2, 1, 1, 1, 0, 2, 2, 0],  
> [2, 1, 3, 0, 1, 0, 1, 0, 1, 1],  
> [0, 3, 0, 2, 3, 1, 0, 1, 0, 0],  
> [1, 0, 2, 1, 0, 1, 0, 2, 1, 2],  
> [2, 2, 2, 0, 0, 1, 1, 1, 1, 0],  
> [1, 1, 1, 1, 1, 2, 0, 0, 2, 1]]
```

Output contains 10 columns representing numbers from 1 to 10. The values are the counts of the numbers in the respective rows.

For example, Cell[0,2] has the value 2, which means, the number 3 occurs exactly 2 times in the 1st row.

Show Solution

```
# Input:  
np.random.seed(100)  
arr = np.random.randint(1,11,size=(6, 10))  
arr  
#> array([[ 9,  9,  4,  8,  8,  1,  5,  3,  6,  3],  
#>         [ 3,  3,  2,  1,  9,  5,  1, 10,  7,  3],  
#>         [ 5,  2,  6,  4,  5,  5,  4,  8,  2,  2],  
#>         [ 8,  8,  1,  3, 10, 10,  4,  3,  6,  9],  
#>         [ 2,  1,  8,  7,  3,  1,  9,  3,  6,  2],  
#>         [ 9,  2,  6,  5,  3,  9,  4,  6,  1, 10]])
```

Feedback

```

# Solution

def counts_of_all_values_rowwise(arr2d):
    # Unique values and its counts row wise
    num_counts_array = [np.unique(row, return_counts=True) for row in arr2d]

    # Counts of all values row wise
    return([[int(b[a==i]) if i in a else 0 for i in np.unique(arr2d)] for a, b in num_counts_array])

# Print
print(np.arange(1,11))
counts_of_all_values_rowwise(arr)
#> [ 1  2  3  4  5  6  7  8  9 10]

#> [[1, 0, 2, 1, 1, 1, 0, 2, 2, 0],
#>  [2, 1, 3, 0, 1, 0, 1, 0, 1, 1],
#>  [0, 3, 0, 2, 3, 1, 0, 1, 0, 0],
#>  [1, 0, 2, 1, 0, 1, 0, 2, 1, 2],
#>  [2, 2, 2, 0, 0, 1, 1, 1, 1, 0],
#>  [1, 1, 1, 1, 1, 2, 0, 0, 2, 1]]

```

```

# Example 2:

arr = np.array([np.array(list('bill clinton')), np.array(list('narendramodi')), np.array(list('jjayalalitha'))])
print(np.unique(arr))
counts_of_all_values_rowwise(arr)
#> [ ' ' 'a' 'b' 'c' 'd' 'e' 'h' 'i' 'j' 'l' 'm' 'n' 'o' 'r' 't' 'y']

#> [[1, 0, 1, 1, 0, 0, 0, 2, 0, 3, 0, 2, 1, 0, 1, 0],
#>  [0, 2, 0, 0, 2, 1, 0, 1, 0, 0, 1, 2, 1, 2, 0, 0],
#>  [0, 4, 0, 0, 0, 0, 1, 1, 2, 2, 0, 0, 0, 0, 1, 1]]

```

50. How to convert an array of arrays into a flat 1d array?

Difficulty Level: 2

Q. Convert `array_of_arrays` into a flat linear 1d array.

Input:

[Feedback](#)

```
# Input:  
arr1 = np.arange(3)  
arr2 = np.arange(3,7)  
arr3 = np.arange(7,10)  
  
array_of_arrays = np.array([arr1, arr2, arr3])  
array_of_arrays  
#> array([array([0, 1, 2]), array([3, 4, 5, 6]), array([7, 8, 9])], dtype=object)
```

Desired Output:

```
#> array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])
```

Show Solution

```
# Input:  
arr1 = np.arange(3)  
arr2 = np.arange(3,7)  
arr3 = np.arange(7,10)  
  
array_of_arrays = np.array([arr1, arr2, arr3])  
print('array_of_arrays: ', array_of_arrays)  
  
# Solution 1  
arr_2d = np.array([a for arr in array_of_arrays for a in arr])  
  
# Solution 2:  
arr_2d = np.concatenate(array_of_arrays)  
print(arr_2d)  
#> array_of_arrays: [array([0, 1, 2]) array([3, 4, 5, 6]) array([7, 8, 9])]  
#> [0 1 2 3 4 5 6 7 8 9]
```

51. How to generate one-hot encodings for an array in numpy?

Difficulty Level L4

Q. Compute the one-hot encodings [dummy binary variables for each unique value in the array]

Input:

Feedback

```
np.random.seed(101)
arr = np.random.randint(1,4, size=6)
arr
#> array([2, 3, 2, 2, 2, 1])
```

Output:

```
#> array([[ 0.,  1.,  0.],
#>        [ 0.,  0.,  1.],
#>        [ 0.,  1.,  0.],
#>        [ 0.,  1.,  0.],
#>        [ 0.,  1.,  0.],
#>        [ 1.,  0.,  0.]])
```

Show Solution

```
# Input:
np.random.seed(101)
arr = np.random.randint(1,4, size=6)
arr
#> array([2, 3, 2, 2, 2, 1])

# Solution:
def one_hot_encodings(arr):
    uniqs = np.unique(arr)
    out = np.zeros((arr.shape[0], uniqs.shape[0]))
    for i, k in enumerate(arr):
        out[i, k-1] = 1
    return out

one_hot_encodings(arr)
#> array([[ 0.,  1.,  0.],
#>        [ 0.,  0.,  1.],
#>        [ 0.,  1.,  0.],
#>        [ 0.,  1.,  0.],
#>        [ 0.,  1.,  0.],
#>        [ 1.,  0.,  0.]])
```

Method 2:
arr[:, None] == np.unique(arr).view(np.int8)

52. How to create row numbers grouped by a categorical variable?

Feedback

Q. Create row numbers grouped by a categorical variable. Use the following sample from `iris` species as input.

Input:

```
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data_(http
s://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'
species = np.genfromtxt(url, delimiter=',', dtype='str', usecols=4)
species_small = np.sort(np.random.choice(species, size=20))
species_small
#> array(['Iris-setosa', 'Iris-setosa', 'Iris-setosa', 'Iris-setosa',
#>        'Iris-setosa', 'Iris-setosa', 'Iris-versicolor', 'Iris-versicolor',
#>        'Iris-versicolor', 'Iris-versicolor', 'Iris-versicolor',
#>        'Iris-versicolor', 'Iris-virginica', 'Iris-virginica',
#>        'Iris-virginica', 'Iris-virginica', 'Iris-virginica',
#>        'Iris-virginica', 'Iris-virginica', 'Iris-virginica'],
#>       dtype='<U15')
```

Desired Output:

```
#> [0, 1, 2, 3, 4, 5, 0, 1, 2, 3, 4, 5, 0, 1, 2, 3, 4, 5, 6, 7]
```

Show Solution

```
# Input:
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data_(htt
ps://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)'
species = np.genfromtxt(url, delimiter=',', dtype='str', usecols=4)
np.random.seed(100)
species_small = np.sort(np.random.choice(species, size=20))
species_small
#> array(['Iris-setosa', 'Iris-setosa', 'Iris-setosa', 'Iris-setosa',
#>        'Iris-setosa', 'Iris-versicolor', 'Iris-versicolor',
#>        'Iris-versicolor', 'Iris-versicolor', 'Iris-versicolor',
#>        'Iris-versicolor', 'Iris-virginica', 'Iris-virginica',
#>        'Iris-virginica', 'Iris-virginica', 'Iris-virginica',
#>        'Iris-virginica'], 
#>       dtype='<U15')
```

```
print([i for val in np.unique(species_small) for i, grp in enumerate(species_small[species_small==val])])
```

```
[0, 1, 2, 3, 4, 0, 1, 2, 3, 4, 5, 6, 7, 8, 0, 1, 2, 3, 4, 5]
```

53. How to create group ids based on a given categorical variable?

Difficulty Level: L4

Q. Create group ids based on a given categorical variable. Use the following sample from iris species as input.

Input:

```
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data' #http://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data'
species = np.genfromtxt(url, delimiter=',', dtype='str', usecols=4)
species_small = np.sort(np.random.choice(species, size=20))
species_small
#> array(['Iris-setosa', 'Iris-setosa', 'Iris-setosa', 'Iris-setosa',
#>        'Iris-setosa', 'Iris-setosa', 'Iris-versicolor', 'Iris-versicolor',
#>        'Iris-versicolor', 'Iris-versicolor', 'Iris-versicolor',
#>        'Iris-versicolor', 'Iris-virginica', 'Iris-virginica',
#>        'Iris-virginica', 'Iris-virginica', 'Iris-virginica',
#>        'Iris-virginica', 'Iris-virginica', 'Iris-virginica'],
#>       dtype='|<U15')
```

Desired Output:

```
#> [0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2]
```

Show Solution

Feedback

```

# Input:
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data' (https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)
species = np.genfromtxt(url, delimiter=',', dtype='str', usecols=4)
np.random.seed(100)
species_small = np.sort(np.random.choice(species, size=20))
species_small
#> array(['Iris-setosa', 'Iris-setosa', 'Iris-setosa', 'Iris-setosa',
#>        'Iris-setosa', 'Iris-versicolor', 'Iris-versicolor',
#>        'Iris-versicolor', 'Iris-versicolor', 'Iris-versicolor',
#>        'Iris-versicolor', 'Iris-versicolor', 'Iris-versicolor',
#>        'Iris-versicolor', 'Iris-virginica', 'Iris-virginica',
#>        'Iris-virginica', 'Iris-virginica', 'Iris-virginica',
#>        'Iris-virginica'],
#>       dtype='|<U15')

```

```

# Solution:
output = [np.argwhere(np.unique(species_small) == s).tolist()[0][0] for val in np.unique(species_small) for s in species_small[species_small==val]]

# Solution: For Loop version
output = []
uniqs = np.unique(species_small)

for val in uniqs: # uniq values in group
    for s in species_small[species_small==val]: # each element in group
        groupid = np.argwhere(uniqs == s).tolist()[0][0] # groupid
        output.append(groupid)

print(output)
#> [0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2]

```

54. How to rank items in an array using numpy?

Difficulty Level: L2

Q. Create the ranks for the given numeric array `a`.

Input:

[Feedback](#)

```
np.random.seed(10)
a = np.random.randint(20, size=10)
print(a)
#> [ 9  4 15  0 17 16 17  8  9  0]
```

Desired output:

```
[4 2 6 0 8 7 9 3 5 1]
```

Show Solution

```
np.random.seed(10)
a = np.random.randint(20, size=10)
print('Array: ', a)

# Solution
print(a.argsort().argsort())
print('Array: ', a)
#> Array:  [ 9  4 15  0 17 16 17  8  9  0]
#> [4 2 6 0 8 7 9 3 5 1]
#> Array:  [ 9  4 15  0 17 16 17  8  9  0]
```

55. How to rank items in a multidimensional array using numpy?

Difficulty Level: L3

Q. Create a rank array of the same shape as a given numeric array a.

Input:

```
np.random.seed(10)
a = np.random.randint(20, size=[2,5])
print(a)
#> [[ 9  4 15  0 17]
#>  [16 17  8  9  0]]
```

Desired output:

```
#> [[4 2 6 0 8]
#>  [7 9 3 5 1]]
```

Feedback

Show Solution

```
# Input:  
np.random.seed(10)  
a = np.random.randint(20, size=[2,5])  
print(a)  
  
# Solution  
print(a.ravel().argsort().argsort().reshape(a.shape))  
#> [[ 9  4 15  0 17]  
#>  [16 17  8  9  0]]  
#> [[4 2 6 0 8]  
#>  [7 9 3 5 1]]
```

56. How to find the maximum value in each row of a numpy array 2d?

DifficultyLevel: L2

Q. Compute the maximum for each row in the given array.

```
np.random.seed(100)  
a = np.random.randint(1,10, [5,3])  
a  
#> array([[9,  9,  4],  
#>        [8,  8,  1],  
#>        [5,  3,  6],  
#>        [3,  3,  3],  
#>        [2,  1,  9]])
```

Show Solution

```
# Input  
np.random.seed(100)  
a = np.random.randint(1,10, [5,3])  
a  
  
# Solution 1  
np.amax(a, axis=1)  
  
# Solution 2  
np.apply_along_axis(np.max, arr=a, axis=1)  
#> array([9, 8, 6, 3, 9])
```

Feedback

57. How to compute the min-by-max for each row for a numpy array 2d?

DifficultyLevel: L3

Q. Compute the min-by-max for each row for given 2d numpy array.

```
np.random.seed(100)
a = np.random.randint(1,10, [5,3])
a
#> array([[9, 9, 4],
#>        [8, 8, 1],
#>        [5, 3, 6],
#>        [3, 3, 3],
#>        [2, 1, 9]])
```

Show Solution

```
# Input
np.random.seed(100)
a = np.random.randint(1,10, [5,3])
a

# Solution
np.apply_along_axis(lambda x: np.min(x)/np.max(x), arr=a, axis=1)
#> array([ 0.44444444,  0.125      ,  0.5       ,  1.         ,  0.11111111])
```

58. How to find the duplicate records in a numpy array?

Difficulty Level: L3

Q. Find the duplicate entries {2nd occurrence onwards} in the given numpy array and mark them as `True`. First time occurrences should be `False`.

```
# Input
np.random.seed(100)
a = np.random.randint(0, 5, 10)
print('Array: ', a)
#> Array: [0 0 3 0 2 4 2 2 2 2]
```

Desired Output:

[Feedback](#)

```
#> [False True False True False False True True True True]
```

Show Solution

```
# Input
np.random.seed(100)
a = np.random.randint(0, 5, 10)

## Solution
# There is no direct function to do this as of 1.13.3

# Create an all True array
out = np.full(a.shape[0], True)

# Find the index positions of unique elements
unique_positions = np.unique(a, return_index=True)[1]

# Mark those positions as False
out[unique_positions] = False

print(out)
#> [False True False True False False True True True True]
```

59. How to find the grouped mean in numpy?

Difficulty Level L3

Q. Find the mean of a numeric column grouped by a categorical column in a 2D numpy array

Input:

```
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data' (http://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data)
iris = np.genfromtxt(url, delimiter=',', dtype='object')
names = ('sepallength', 'sepalwidth', 'petallength', 'petalwidth', 'species')
```

Desired Solution:

```
#> [[b'Iris-setosa', 3.418],
#> [b'Iris-versicolor', 2.770],
#> [b'Iris-virginica', 2.974]]
```

Feedback

Show Solution

```
# Input
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data'
      #'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data'
iris = np.genfromtxt(url, delimiter=',', dtype='object')
names = ('sepallength', 'sepalwidth', 'petallength', 'petalwidth', 'species')

# Solution
# No direct way to implement this. Just a version of a workaround.
numeric_column = iris[:, 1].astype('float') # sepalwidth
grouping_column = iris[:, 4] # species

# List comprehension version
[[group_val, numeric_column[grouping_column==group_val].mean()] for group_val in np.unique(grouping_column)]

# For Loop version
output = []
for group_val in np.unique(grouping_column):
    output.append([group_val, numeric_column[grouping_column==group_val].mean()])

output
#> [[b'Iris-setosa', 3.418],
#> [b'Iris-versicolor', 2.770],
#> [b'Iris-virginica', 2.974]]
```

60. How to convert a PIL image to numpy array?

Difficulty Level: L3

Q. Import the image from the following URL and convert it to a numpy array.

```
URL = 'https://upload.wikimedia.org/wikipedia/commons/8/8b/Denali_Mt_McKinley.jpg
      (https://upload.wikimedia.org/wikipedia/commons/8/8b/Denali_Mt_McKinley.jpg)'
```

Show Solution

Feedback

```

from io import BytesIO
from PIL import Image
import PIL, requests

# Import image from URL
URL = 'https://upload.wikimedia.org/wikipedia/commons/8/8b/Denali_Mt_McKinley.jpg'(h
ttps://upload.wikimedia.org/wikipedia/commons/8/8b/Denali_Mt_McKinley.jpg)'
response = requests.get(URL)

# Read it as Image
I = Image.open(BytesIO(response.content))

# Optionally resize
I = I.resize([150,150])

# Convert to numpy array
arr = np.asarray(I)

# Optionally Convert it back to an image and show
im = PIL.Image.fromarray(np.uint8(arr))
Image.Image.show(im)

```

61. How to drop all missing values from a numpy array?

Difficulty Level: L2

Q. Drop all nan values from a 1D numpy array

Input:

```
np.array([1,2,3,np.nan,5,6,7,np.nan])
```

Desired Output:

```
array([ 1.,  2.,  3.,  5.,  6.,  7.])
```

Show Solution

```

a = np.array([1,2,3,np.nan,5,6,7,np.nan])
a[~np.isnan(a)]
#> array([ 1.,  2.,  3.,  5.,  6.,  7.])

```

[Feedback](#)

62. How to compute the euclidean distance between two arrays?

Difficulty Level: L3

Q. Compute the euclidean distance between two arrays `a` and `b`.

Input:

```
a = np.array([1,2,3,4,5])
b = np.array([4,5,6,7,8])
```

Show Solution

```
# Input
a = np.array([1,2,3,4,5])
b = np.array([4,5,6,7,8])

# Solution
dist = np.linalg.norm(a-b)
dist
#> 6.7082039324993694
```

63. How to find all the local maxima (or peaks) in a 1d array?

Difficulty Level: L4

Q. Find all the peaks in a 1D numpy array `a`. Peaks are points surrounded by smaller values on both sides.

Input:

```
a = np.array([1, 3, 7, 1, 2, 6, 0, 1])
```

Desired Output:

```
#> array([2, 5])
```

where, 2 and 5 are the positions of peak values 7 and 6.

Show Solution

Feedback

```
a = np.array([1, 3, 7, 1, 2, 6, 0, 1])
doublediff = np.diff(np.sign(np.diff(a)))
peak_locations = np.where(doublediff == -2)[0] + 1
peak_locations
#> array([2, 5])
```

64. How to subtract a 1d array from a 2d array, where each item of 1d array subtracts from respective row?

Difficulty Level: L2

Q. Subtract the 1d array `b_1d` from the 2d array `a_2d`, such that each item of `b_1d` subtracts from respective row of `a_2d`.

```
a_2d = np.array([[3,3,3],[4,4,4],[5,5,5]])
b_1d = np.array([1,1,1])
```

Desired Output:

```
#> [[2 2 2]
#> [2 2 2]
#> [2 2 2]]
```

Show Solution

```
# Input
a_2d = np.array([[3,3,3],[4,4,4],[5,5,5]])
b_1d = np.array([1,2,3])

# Solution
print(a_2d - b_1d[:,None])
#> [[2 2 2]
#> [2 2 2]
#> [2 2 2]]
```

65. How to find the index of n'th repetition of an item in an array

Difficulty Level L2

Q. Find the index of 5th repetition of number 1 in `x`.

[Feedback](#)

```
x = np.array([1, 2, 1, 1, 3, 4, 3, 1, 1, 2, 1, 1, 2])
```

Show Solution

```
x = np.array([1, 2, 1, 1, 3, 4, 3, 1, 1, 2, 1, 1, 2])
n = 5

# Solution 1: List comprehension
[i for i, v in enumerate(x) if v == 1][n-1]

# Solution 2: Numpy version
np.where(x == 1)[0][n-1]
#> 8
```

66. How to convert numpy's `datetime64` object to datetime's `datetime` object?

Difficulty Level: L2

Q. Convert numpy's `datetime64` object to datetime's `datetime` object

```
# Input: a numpy datetime64 object
dt64 = np.datetime64('2018-02-25 22:10:10')
```

Show Solution

```
# Input: a numpy datetime64 object
dt64 = np.datetime64('2018-02-25 22:10:10')

# Solution
from datetime import datetime
dt64.tolist()

# or

dt64.astype(datetime)
#> datetime.datetime(2018, 2, 25, 22, 10, 10)
```

67. How to compute the moving average of a numpy array?

Difficulty Level: L3

Feedback

Q. Compute the moving average of window size 3, for the given 1D array.

Input:

```
np.random.seed(100)
Z = np.random.randint(10, size=10)
```

Show Solution

```
# Solution
# Source: https://stackoverflow.com/questions/14313510/how-to-calculate-moving-average-using-numpy (https://stackoverflow.com/questions/14313510/how-to-calculate-moving-average-using-numpy)
def moving_average(a, n=3) :
    ret = np.cumsum(a, dtype=float)
    ret[n:] = ret[n:] - ret[:-n]
    return ret[n - 1:] / n

np.random.seed(100)
Z = np.random.randint(10, size=10)
print('array: ', Z)
# Method 1
moving_average(Z, n=3).round(2)

# Method 2: # Thanks AlanLRH!
# np.ones(3)/3 gives equal weights. Use np.ones(4)/4 for window size 4.
np.convolve(Z, np.ones(3)/3, mode='valid') .

#> array: [8 8 3 7 7 0 4 2 5 2]
#> moving average: [ 6.33  6.      5.67  4.67  3.67  2.      3.67  3. ]
```

68. How to create a numpy array sequence given only the starting point, length and the step?

Difficulty Level: L2

Q. Create a numpy array of length 10, starting from 5 and has a step of 3 between consecutive numbers

Show Solution

Feedback

```
length = 10
start = 5
step = 3

def seq(start, length, step):
    end = start + (step*length)
    return np.arange(start, end, step)

seq(start, length, step)
#> array([ 5,  8, 11, 14, 17, 20, 23, 26, 29, 32])
```

69. How to fill in missing dates in an irregular series of numpy dates?

Difficulty Level: L3

Q. Given an array of a non-continuous sequence of dates. Make it a continuous sequence of dates, by filling in the missing dates.

Input:

```
# Input
dates = np.arange(np.datetime64('2018-02-01'), np.datetime64('2018-02-25'), 2)
print(dates)
#> ['2018-02-01' '2018-02-03' '2018-02-05' '2018-02-07' '2018-02-09'
#> '2018-02-11' '2018-02-13' '2018-02-15' '2018-02-17' '2018-02-19'
#> '2018-02-21' '2018-02-23']
```

Show Solution

Feedback

```

# Input
dates = np.arange(np.datetime64('2018-02-01'), np.datetime64('2018-02-25'), 2)
print(dates)

# Solution -----
filled_in = np.array([np.arange(date, (date+d)) for date, d in zip(dates, np.diff(dates))]).reshape(-1)

# add the last day
output = np.hstack([filled_in, dates[-1]])
output

# For loop version -----
out = []
for date, d in zip(dates, np.diff(dates)):
    out.append(np.arange(date, (date+d)))

filled_in = np.array(out).reshape(-1)

# add the last day
output = np.hstack([filled_in, dates[-1]])
output
#> ['2018-02-01' '2018-02-03' '2018-02-05' '2018-02-07' '2018-02-09'
#> '2018-02-11' '2018-02-13' '2018-02-15' '2018-02-17' '2018-02-19'
#> '2018-02-21' '2018-02-23']

#> array(['2018-02-01', '2018-02-02', '2018-02-03', '2018-02-04',
#>        '2018-02-05', '2018-02-06', '2018-02-07', '2018-02-08',
#>        '2018-02-09', '2018-02-10', '2018-02-11', '2018-02-12',
#>        '2018-02-13', '2018-02-14', '2018-02-15', '2018-02-16',
#>        '2018-02-17', '2018-02-18', '2018-02-19', '2018-02-20',
#>        '2018-02-21', '2018-02-22', '2018-02-23'], dtype='datetime64[D]')

```

70. How to create strides from a given 1D array?

Difficulty Level: L4

Q. From the given 1d array `arr`, generate a 2d matrix using strides, with a window length of 4 and strides of 2, like [[0,1,2,3], [2,3,4,5], [4,5,6,7]..]

Input:

[Feedback](#)

```
arr = np.arange(15)
arr
#> array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14])
```

Desired Output:

```
#> [[ 0  1  2  3]
#>  [ 2  3  4  5]
#>  [ 4  5  6  7]
#>  [ 6  7  8  9]
#>  [ 8  9 10 11]
#>  [10 11 12 13]]
```

Show Solution

```
def gen_strides(a, stride_len=5, window_len=5):
    n_strides = ((a.size-window_len)//stride_len) + 1
    # return np.array([a[s:(s+window_len)] for s in np.arange(0, a.size, stride_len)
    [:n_strides]])
    return np.array([a[s:(s+window_len)] for s in np.arange(0, n_strides*stride_len,
    stride_len)])
```



```
print(gen_strides(np.arange(15), stride_len=2, window_len=4))
#> [[ 0  1  2  3]
#>  [ 2  3  4  5]
#>  [ 4  5  6  7]
#>  [ 6  7  8  9]
#>  [ 8  9 10 11]
#>  [10 11 12 13]]
```

To be continued ..

Related



[\[https://www.machinelearningplus.com/pandas/python-tutorial-pandas-part2/\]](https://www.machinelearningplus.com/pandas/python-tutorial-pandas-part2/)
Numpy Tutorial Part 2 - Vital Functions for Data Analysis



[\[https://www.machinelearningplus.com/pandas/python-tutorial-part1-array-python-examples/\]](https://www.machinelearningplus.com/pandas/python-tutorial-part1-array-python-examples/)
Numpy Tutorial Part 1 - Introduction to Arrays



[\[https://www.machinelearningplus.com/pandas-exercises-python/\]](https://www.machinelearningplus.com/pandas-exercises-python/)
101 Pandas Exercises for Data Analysis

Feedback

[https://www.machinelearningplus.com/p/tutorial-python-part2/]	[https://www.machinelearningplus.com/p/tutorial-part1-array-python-examples/]	[https://www.machinelearningplus.com/p/pandas-exercises-python/]
February 14, 2018	February 7, 2018	April 27, 2018
In "Python"	In "Python"	In "Python"

You may also like:

45_Calendar_Heatmap
min

32_PieChart_2_Matplot
min

Python Regex Tutorial -
A Complete Beginners
Guide | ML+

Numpy Tutorial Part 2

Gensim Tutorial - A
Complete Beginners
Guide

Caret Package - A
Complete Guide to
Build Machine Learning
in R

Introduction To Numpy

02_distribution_of_doc
min

Tags:[Numpy](https://www.machinelearningplus.com/tag/numpy/), [Practice Exercise](https://www.machinelearningplus.com/tag/practice-exercise/), [Python](https://www.machinelearningplus.com/tag/python/)

11 Comments [machinelearningplus.com](https://www.machinelearningplus.com)

1 Login ▾

 Recommend 1

 Tweet

 Share

Sort by Best ▾



Join the discussion...

LOG IN WITH

OR SIGN UP WITH DISQUS 

Name



Steve James • 2 months ago

Alternative solution for Q38:

Feedback

```
np.nan_to_num(iris_2d)
```

^ | v · Reply · Share ·



gosta · 3 months ago

I came up with a new solution of Q53.

Is that correct?

```
uniqs, cnts = np.unique(species_small, return_counts=True)  
np.repeat(np.arange(cnts.shape[0]), cnts)
```

^ | v · Reply · Share ·



MANOJ KUMAR · 3 months ago

Alternate Solution of Q39.

```
np.unique(iris[:,4],return_counts=True)
```

^ | v · Reply · Share ·



mayrop · 5 months ago

Keep up the good work! This post is amazing! (and many others from this blog).

^ | v · Reply · Share ·



osawat · 5 months ago

I love your article very very much. Could I translate this into japanese?

^ | v · Reply · Share ·



Selva Prabhakaran Mod → osawat · 5 months ago

Thanks osawat :) Sure please go ahead

^ | v · Reply · Share ·



osawat → Selva Prabhakaran · 5 months ago

Thank you I very much Selva. I will inform you when translation is done.

^ | v · Reply · Share ·



Inayatullah Baloch · a year ago

very informative.

^ | v · Reply · Share ·



Selva Prabhakaran Mod → Inayatullah Baloch · 5 months ago

Thank you :)

^ | v · Reply · Share ·



AMRESH PANDEY · a year ago

Thanks for this awesome blog :)

^ | v · Reply · Share ·



Selva Prabhakaran Mod → AMRESH PANDEY · 5 months ago

Welcome Amresh :)

^ | v · Reply · Share ·

Feedback



Search

(<https://www.machinelearningplus.com/python/python-social-share/>)

Recent Posts

[Mahalanobis Distance – Understanding the math with examples \(python\)](https://www.machinelearningplus.com/statistics/mahalanobis-distance/)
(<https://www.machinelearningplus.com/statistics/mahalanobis-distance/>).

[datetime in Python – Simplified Guide with Clear Examples](https://www.machinelearningplus.com/python/datetime-python-examples/)
(<https://www.machinelearningplus.com/python/datetime-python-examples/>).

[Python Logging – Simplest Guide with Full Code and Examples](https://www.machinelearningplus.com/python/python-logging-guide/)
(<https://www.machinelearningplus.com/python/python-logging-guide/>).

[Matplotlib Histogram – How to Visualize Distributions in Python](https://www.machinelearningplus.com/plots/matplotlib-histogram-python-examples/)
(<https://www.machinelearningplus.com/plots/matplotlib-histogram-python-examples/>).

[ARIMA Model – Complete Guide to Time Series Forecasting in Python](https://www.machinelearningplus.com/time-series/arima-model-time-series-forecasting-python/)
(<https://www.machinelearningplus.com/time-series/arima-model-time-series-forecasting-python/>).

[Time Series Analysis in Python – A Comprehensive Guide with Examples](https://www.machinelearningplus.com/time-series/time-series-analysis-python/)
(<https://www.machinelearningplus.com/time-series/time-series-analysis-python/>).

[Matplotlib Tutorial – A Complete Guide to Python Plot w/ Examples](https://www.machinelearningplus.com/plots/matplotlib-tutorial-complete-guide-python-plot-examples/)
(<https://www.machinelearningplus.com/plots/matplotlib-tutorial-complete-guide-python-plot-examples/>).

[Topic modeling visualization – How to present the results of LDA models?](https://www.machinelearningplus.com/nlp/topic-modeling-visualization-how-to-present-results-lsa-models/)
(<https://www.machinelearningplus.com/nlp/topic-modeling-visualization-how-to-present-results-lsa-models/>).

[Top 50 matplotlib Visualizations – The Master Plots \(with full python code\)](https://www.machinelearningplus.com/plots/top-50-matplotlib-visualizations-the-master-plots-python/)
(<https://www.machinelearningplus.com/plots/top-50-matplotlib-visualizations-the-master-plots-python/>).

[List Comprehensions in Python – My Simplified Guide](https://www.machinelearningplus.com/python/list-comprehensions-in-python/)
(<https://www.machinelearningplus.com/python/list-comprehensions-in-python/>).

[Python @Property Explained – How to Use and When? \(Full Examples\)](https://www.machinelearningplus.com/python/python-property/)
(<https://www.machinelearningplus.com/python/python-property/>).

Feedback

How Naive Bayes Algorithm Works? (with example and full code)
[\(https://www.machinelearningplus.com/predictive-modeling/how-naive-bayes-algorithm-works-with-example-and-full-code/\)](https://www.machinelearningplus.com/predictive-modeling/how-naive-bayes-algorithm-works-with-example-and-full-code/).

Parallel Processing in Python – A Practical Guide with Examples
[\(https://www.machinelearningplus.com/python/parallel-processing-python/\)](https://www.machinelearningplus.com/python/parallel-processing-python/).

Cosine Similarity – Understanding the math and how it works (with python codes)
[\(https://www.machinelearningplus.com/nlp/cosine-similarity/\)](https://www.machinelearningplus.com/nlp/cosine-similarity/).

Gensim Tutorial – A Complete Beginners Guide
[\(https://www.machinelearningplus.com/nlp/gensim-tutorial/\)](https://www.machinelearningplus.com/nlp/gensim-tutorial/).

Lemmatization Approaches with Examples in Python
[\(https://www.machinelearningplus.com/nlp/lemmatization-examples-python/\)](https://www.machinelearningplus.com/nlp/lemmatization-examples-python/).

Feature Selection – Ten Effective Techniques with Examples
[\(https://www.machinelearningplus.com/machine-learning/feature-selection/\)](https://www.machinelearningplus.com/machine-learning/feature-selection/).

101 Pandas Exercises for Data Analysis
[\(https://www.machinelearningplus.com/python/101-pandas-exercises-python/\)](https://www.machinelearningplus.com/python/101-pandas-exercises-python/).

LDA in Python – How to grid search best topic models?
[\(https://www.machinelearningplus.com/nlp/topic-modeling-python-sklearn-examples/\)](https://www.machinelearningplus.com/nlp/topic-modeling-python-sklearn-examples/).

Topic Modeling with Gensim (Python)
[\(https://www.machinelearningplus.com/nlp/topic-modeling-gensim-python/\)](https://www.machinelearningplus.com/nlp/topic-modeling-gensim-python/).

Python debugging with pdb (<https://www.machinelearningplus.com/python/python-debugging/>).

Caret Package – A Practical Guide to Machine Learning in R
[\(https://www.machinelearningplus.com/machine-learning/caret-package/\)](https://www.machinelearningplus.com/machine-learning/caret-package/).

101 NumPy Exercises for Data Analysis (Python)
[\(https://www.machinelearningplus.com/python/101-numpy-exercises-python/\)](https://www.machinelearningplus.com/python/101-numpy-exercises-python/).

Numpy Tutorial Part 2 – Vital Functions for Data Analysis
[\(https://www.machinelearningplus.com/python\(numpy-tutorial-python-part2/\)](https://www.machinelearningplus.com/python(numpy-tutorial-python-part2/).

Numpy Tutorial Part 1 – Introduction to Arrays
[\(https://www.machinelearningplus.com/python\(numpy-tutorial-part1-array-python-examples/\)](https://www.machinelearningplus.com/python(numpy-tutorial-part1-array-python-examples/).

Tags

[@property](https://www.machinelearningplus.com/tag/property/) [<https://www.machinelearningplus.com/tag/property/>] [Bigrams](https://www.machinelearningplus.com/tag/bigrams/) [<https://www.machinelearningplus.com/tag/bigrams/>]

[Classification](https://www.machinelearningplus.com/tag/classification/) [<https://www.machinelearningplus.com/tag/classification/>)

[Corpus](https://www.machinelearningplus.com/tag/corpus/) [<https://www.machinelearningplus.com/tag/corpus/>])

[Cosine Similarity](https://www.machinelearningplus.com/tag/cosine-similarity/) [<https://www.machinelearningplus.com/tag/cosine-similarity/>])

[Data Manipulation](https://www.machinelearningplus.com/tag/data-manipulation/) [<https://www.machinelearningplus.com/tag/data-manipulation/>)

[Debugging](https://www.machinelearningplus.com/tag/debugging/) [<https://www.machinelearningplus.com/tag/debugging/>] [Doc2Vec](https://www.machinelearningplus.com/tag/doc2vec/) [<https://www.machinelearningplus.com/tag/doc2vec/>])

[Evaluation Metrics](https://www.machinelearningplus.com/tag/evaluation-metrics/) [<https://www.machinelearningplus.com/tag/evaluation-metrics/>])

[FastText](https://www.machinelearningplus.com/tag/fasttext/) [<https://www.machinelearningplus.com/tag/fasttext/>])

[Feature Selection](https://www.machinelearningplus.com/tag/feature-selection/) [<https://www.machinelearningplus.com/tag/feature-selection/>])

[Gensim](https://www.machinelearningplus.com/tag/gensim/) [<https://www.machinelearningplus.com/tag/gensim/>])

[KlaR](https://www.machinelearningplus.com/tag/klar/) [<https://www.machinelearningplus.com/tag/klar/>] [LDA](https://www.machinelearningplus.com/tag/lda/) [<https://www.machinelearningplus.com/tag/lda/>])

[Lemmatization](https://www.machinelearningplus.com/tag/lemmatization/) [<https://www.machinelearningplus.com/tag/lemmatization/>])

[Linear Regression](https://www.machinelearningplus.com/tag/linear-regression/) [<https://www.machinelearningplus.com/tag/linear-regression/>])

[Logistic](https://www.machinelearningplus.com/tag/logistic/) [<https://www.machinelearningplus.com/tag/logistic/>] [LSI](https://www.machinelearningplus.com/tag/lsi/) [<https://www.machinelearningplus.com/tag/lsi/>])

[Matplotlib](https://www.machinelearningplus.com/tag/matplotlib/) [<https://www.machinelearningplus.com/tag/matplotlib/>])

[Multiprocessing](https://www.machinelearningplus.com/tag/multiprocessing/) [<https://www.machinelearningplus.com/tag/multiprocessing/>])

[Naive Bayes](https://www.machinelearningplus.com/tag/naive-bayes/) [<https://www.machinelearningplus.com/tag/naive-bayes/>])

[NLP](https://www.machinelearningplus.com/tag/nlp/) [<https://www.machinelearningplus.com/tag/nlp/>])

[NLTK](https://www.machinelearningplus.com/tag/nltk/) [<https://www.machinelearningplus.com/tag/nltk/>])

[Numpy](https://www.machinelearningplus.com/tag(numpy/) [[https://www.machinelearningplus.com/tag\(numpy/](https://www.machinelearningplus.com/tag(numpy/)])

[Pandas](https://www.machinelearningplus.com/tag/pandas/) [<https://www.machinelearningplus.com/tag/pandas/>])

[Parallel Processing](https://www.machinelearningplus.com/tag/parallel-processing/) [<https://www.machinelearningplus.com/tag/parallel-processing/>])

[Phraser](https://www.machinelearningplus.com/tag/phraser/) [<https://www.machinelearningplus.com/tag/phraser/>])

[Practice Exercise](https://www.machinelearningplus.com/tag/practice-exercise/) [<https://www.machinelearningplus.com/tag/practice-exercise/>])

Python

[\(https://www.machinelearningplus.com/tag/python/\)](https://www.machinelearningplus.com/tag/python/)

[R](https://www.machinelearningplus.com/tag/r/) [<https://www.machinelearningplus.com/tag/r/>])

[Regex](https://www.machinelearningplus.com/tag/regex/) [<https://www.machinelearningplus.com/tag/regex/>])

[Regression](https://www.machinelearningplus.com/tag/regression/) [<https://www.machinelearningplus.com/tag/regression/>])

Feedback

[Residual Analysis](https://www.machinelearningplus.com/tag/residual-analysis/)

[Scikit Learn](https://www.machinelearningplus.com/tag/scikit-learn/)

[Significance Tests](https://www.machinelearningplus.com/tag/significance-tests/)

[Soft Cosine Similarity](https://www.machinelearningplus.com/tag/soft-cosine-similarity/)

[spaCy](https://www.machinelearningplus.com/tag/spacy/) [Summarization](https://www.machinelearningplus.com/tag/summarization/)

[TaggedDocument](https://www.machinelearningplus.com/tag/taggeddocument/)

[TextBlob](https://www.machinelearningplus.com/tag/textblob/) [TFIDF](https://www.machinelearningplus.com/tag/tfidf/)

[Time Series](https://www.machinelearningplus.com/tag/time-series/)

[Topic Modeling](https://www.machinelearningplus.com/tag/topic-modeling/)

[Visualization](https://www.machinelearningplus.com/tag/visualization/)

[Word2Vec](https://www.machinelearningplus.com/tag/word2vec/)

[<https://www.ezoic.com/what-is-ezoic/>]

report this ad

[<https://www.ezoic.com/what-is-ezoic/>] report this ad

[HOME](https://www.machinelearningplus.com/)

/ [ALL POSTS](https://www.machinelearningplus.com/blog/)

/ [DATA MANIPULATION](https://www.machinelearningplus.com/category/data-manipulation/)

/ [PREDICTIVE MODELING](https://www.machinelearningplus.com/category/predictive-modeling/)

[\[HTTPS://WWW.MACHINELEARNINGPLUS.COM/CATEGORY/PREDICTIVE-MODELING/\]](https://www.machinelearningplus.com/category/predictive-modeling/)

/ [STATISTICS](https://www.machinelearningplus.com/category/statistics/)

/ [NLP](https://www.machinelearningplus.com/category/nlp/)

/ [PYTHON](https://www.machinelearningplus.com/category/python/)

/ [PLOTS](https://www.machinelearningplus.com/category/plots/)

/ [TIME SERIES](https://www.machinelearningplus.com/category/time-series/)

/ [CONTACT US](https://www.machinelearningplus.com/contact-us/)

© Copyright by Machine Learning Plus | All rights reserved | [Privacy Policy](https://www.machinelearningplus.com/privacy-policy/)

[\[https://www.machinelearningplus.com/privacy-policy/\]](https://www.machinelearningplus.com/privacy-policy/) | [Terms of Use](https://www.machinelearningplus.com/terms-of-use/)

[\[https://www.machinelearningplus.com/terms-of-use/\]](https://www.machinelearningplus.com/terms-of-use/) | [About](https://www.machinelearningplus.com/about/)

[\[https://www.machinelearningplus.com/about\]](https://www.machinelearningplus.com/about/)