

Forum: [Hack Forums](#)

Thread: [how to get average of numeric values from string in list?](#)

bigfatjoe - January 5, 2024, 5:29 pm

I tried splitting each value up into two values and appending average of each two values into an array and casted to integers. its giving me an error. Here is code:

Code import numpy as np

```
cats_api = requests.get('https://api.thecatapi.com/v1/breeds')
cats = cats_api.json()
weight_list = np.array([])
for cat in cats:
    num = cat["weight"]["metric"]
    first_num = int(num.split("-")[0])
    second_num = int(num.split("-")[1])
    value = int((first_num + second_num)/2)
    print(value)
    np.concatenate(weight_list, value)
```

print(weight_list)

Error I am getting:

Code greg@greg-IdeaPad-5-15ARE05:~\$ /bin/python3 "/home/greg/Documents/Python/Day 20.py"
4

Traceback (most recent call last):

```
File "/home/greg/Documents/Python/Day 20.py", line 26, in <module>
    np.concatenate(weight_list, value)
```

```
File "<__array_function__ internals>", line 200, in concatenate
```

ValueError: need at least one array to concatenate

This is for exercise 2 of Day 20 of the 30 Days of Python course:

https://github.com/Asabeneh/30-Days-Of-Python/blob/master/20_Day_Python_package_manager/20_python_package_manager.md

Dirty - January 5, 2024, 10:40 pm

Never used bumpy but from reading the docs and a few stackoverflow posts, my first thing I would try is having a value inside weight lists. This function is to concatenate arrays together. Or a value to an array I guess although for that there is simply better things. In this case you have a empty array and just a value which is what it's triggering over I think.

Perlo - January 8, 2024, 12:23 am

You are looking for a similar function to python's list append, which simply adds something to the end an existing list. np.concatenate isnt the correct function for this. You are better off using numpy's np.append(), which takes in the list to append to and the object to add. In addition to that, since we are using numpy its very easy to use the built in functions (such as np.mean, np.min, np.max, np.std). Also, you can remove the int() in your way of calculating the average, since you

already converted the 2 numbers to integers, and divided by another integer, meaning we already have an integer. Meaning value = $\text{int}((\text{first_num} + \text{second_num})/2)$ can be written shorter as: value = $(\text{first_num} + \text{second_num})/2$. Also even shorter in the notation below using only numpy.

```
Code import numpy as np
import requests
```

```
cats_api = requests.get('https://api.thecatapi.com/v1/breeds')
cats = cats_api.json()
weight_list = np.array([])
```

```
for cat in cats:
    num = cat["weight"]["metric"]
    first_num = int(num.split("-")[0])
    second_num = int(num.split("-")[1])
    value = np.mean([first_num, second_num])
    weight_list = np.append(weight_list, value)
```

```
w_mean = np.mean(weight_list)
w_min = np.min(weight_list)
w_max = np.max(weight_list)
w_std = np.std(weight_list)
```

snoopy717 - January 8, 2024, 4:46 am

It looks like you are trying to use `np.concatenate` incorrectly. The `np.concatenate` function requires at least two arrays to concatenate. Additionally, it's better to use `np.append` in this case, as you are adding elements one by one. Here's the corrected code:

```
Code import numpy as np
import requests
```

```
cats_api = requests.get('https://api.thecatapi.com/v1/breeds')
cats = cats_api.json()
weight_list = np.array([])
```

```
for cat in cats:
    num = cat["weight"]["metric"]
    first_num = int(num.split("-")[0])
    second_num = int(num.split("-")[1])
    value = int((first_num + second_num) / 2)
    print(value)
    weight_list = np.append(weight_list, value)
```

`print(weight_list)` In this corrected version, I replaced `np.concatenate` with `np.append` inside the loop, allowing you to add elements to the `weight_list` one at a time. The resulting array should contain the average values as intended.

If you have any question, PM me

bigfatjoe - January 8, 2024, 6:30 pm

I solved it before you guys posted this with a little hint or two from a friend. Here is right code:

```
Code cats_api = requests.get('https://api.thecatapi.com/v1/breeds')
cats = cats_api.json()
```

```

weight_list = []
for cat in cats:
    num = cat["weight"]["metric"]
    first_num = int(num.split("-")[0])
    second_num = int(num.split("-")[1])
    values = [first_num, second_num]
    value = statistics.mean(values)
    weight_list.append(value)

weight_list = np.array(weight_list)
print(weight_list)
print("Minimum value is: ", np.min(weight_list))
print("Maximum value is: ", np.max(weight_list))
print("Median value is: ", np.median(weight_list))
print("Standard deviation of cats' weight is: ", np.std(weight_list))

```

```

life_span_list = []
for cat in cats:
    num = cat["life_span"]
    first_num = int(num.split("-")[0])
    second_num = int(num.split("-")[1])
    values = [first_num, second_num]
    life_span_list.append(statistics.mean(values))

life_span_list = np.array(life_span_list)
print(life_span_list)
print("Minimum age is:", np.min(life_span_list))
print("Maximum age is: ", np.max(life_span_list))
print("Median age is:", np.median(life_span_list))
print("Standard deviation of cats' age is: ", np.std(life_span_list))

```

```

cats_json = json.dumps(cats)

data = pd.read_json(cats_json)

frequency_table = pd.crosstab(data['country_code'], data['vetstreet_url'])

print(frequency_table)

```

siliconvalley - April 3, 2024, 2:33 am

Try this:

```

import numpy as np
import requests

cats_api = requests.get('https://api.thecatapi.com/v1/breeds')
cats = cats_api.json()

weight_list = np.array([])

for cat in cats:
    num = cat["weight"]["metric"]
    first_num = int(num.split("-")[0])

```

```
second_num = int(num.split("-")[1])
value = int((first_num + second_num) / 2)
weight_list = np.concatenate((weight_list, np.array([value])))

print(weight_list)
```