

NumPy Split İşlemi (Diziyi Bölmek)

🎓 Öğretmene Notlar:

Öğrencilere "Bir dizi çok büyükse, parçalarına ayırmak kolaylık sağlar." benzetmesiyle açıklayın.

Özellikle makine öğrenmesinde veri setlerini eğitim/test olarak bölmede önemlidir.

📌 Konu Anlatımı:

```
import numpy as np
dizi = np.array([1, 2, 3, 4, 5, 6])
parcalar = np.split(dizi, 3)
print(parcalar)
```

🌸 İki boyutlu diziler için:

```
matris = np.arange(16).reshape(4,4)
print(np.vsplit(matris, 2)) # Satırdan bölme
print(np.hsplit(matris, 2)) # Sütundan bölme
```

Sorting İşlemi (Sıralama)

🎓 Öğretmene Notlar:

Öğrencilere liste sıralamayla farkını anlatın: NumPy np.sort() ile yeni dizi döndürür, orijinali değiştirmez.

Öğretmen: "np.sort(): yeni sıralı, sort(): yerinde sıralar"

📌 Örnek:

```
dizi = np.array([5, 1, 8, 3])
print(np.sort(dizi))      # [1 3 5 8]
print(dizi)               # [5 1 8 3] - değişmedi
```

Index (İndeksleme)

🎓 Öğretmene Notlar:

Sıfırdan başlamaya dikkat! ([0] ilk eleman)

Dizi içindeki elemanlara noktasal erişim.

📌 Örnek:

```
a = np.array([10, 20, 30])
print(a[0]) # 10
```

İki boyutlu örnek:

```
b = np.array([[1, 2, 3], [4, 5, 6]])
print(b[1, 2]) # 6
```

Alt Kümeler (Subsets)

🎓 Öğretmene Notlar:

Bu konu slicing ve indexleme ile yakından ilişkili.

Öğrencilere "bir Excel tablosundan satır/sütun seçmek" gibi düşüncelerini önerin.

📌 Örnek:

```
matris = np.array([[10, 20], [30, 40], [50, 60]])
alt_kume = matris[0:2, :] # ilk iki satır, tüm sütunlar
print(alt_kume)
```

Slicing ve İleri Indexleme (Dilimleme)

🎓 Öğretmene Notlar: Python dilimleme ([start:stop:step]) mantığı burada da geçerli.

Öğrencilere [:] tüm diziyi ifade eder, negatif indeks ile sondan ulaşılabileceği gösterilmeli.

```
#Örnek
import numpy as np
a = np.array([0, 1, 2, 3, 4, 5])
```


```
print(a[1:4])      # [1 2 3]
print(a[:,2])      # [0 2 4]
print(a[::-1])     # [5 4 3 2 1 0] (ters çevirir)

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

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

Alıştırmalar

- 1- [10, 20, 30, 40, 50, 60] dizisini 3 parçaya bölün.
- 2- [4, 1, 9, 2] dizisini sıralayın.
- 3- [[5, 10], [15, 20]] dizisinden sadece 20'yi alın.
- 4- 6x3 bir matris oluşturun ve 2. ile 4. satırı ayıklayın.
- 5- [100, 200, 300, 400, 500] dizisini tersten yazdırın.

 1. [10, 20, 30, 40, 50, 60] dizisini 3 parçaya bölün.

```
import numpy as np


dizi = np.array([10, 20, 30, 40, 50, 60])
parcalar = np.split(dizi, 3)

print(parcalar)
```

 2. [4, 1, 9, 2] dizisini sıralayın.


```
dizi = np.array([4, 1, 9, 2])
sirali = np.sort(dizi)

print(sirali) # [1 2 4 9]
```

 3. [[5, 10], [15, 20]] dizisinden sadece 20'yi alın.

```
matris = np.array([[5, 10], [15, 20]])
deger = matris[1, 1]

print(deger) # 20
```

 4. 6x3 bir matris oluşturun ve 2. ile 4. satırı ayıklayın.

```
matris = np.arange(18).reshape(6, 3)
alt_kume = matris[[1, 3], :] # 2. ve 4. satır (indeks 1 ve 3)

print("Matris:\n", matris)
print("Alt Küme:\n", alt_kume)
```

 5. [100, 200, 300, 400, 500] dizisini tersten yazdırın.

```
dizi = np.array([100, 200, 300, 400, 500])
ters = dizi[::-1]

print(ters) # [500 400 300 200 100]
```

Bu Konular Nereelerde İşimize Yarar?

Array (Dizi) Yapısı – Reshape – Split – Concatenate

Gerçek Uygulama

Sensör verilerini okumak ve işlemek Verileri matris yapısına dönüştürme, yeniden boyutlandırma
Eğitim ve test verilerini bölmek `split()` ile veri kümesi ayırma
Görsel veya ses verilerini işlemek Çok boyutlu diziyle çalışabilme yeteneği

Yardımı

Indexleme, Slicing, Alt Kümeler

Gerçek Uygulama

Sadece belirli tarih aralığını analiz etmek Slicing ile zaman serisini kesmek
Excel benzeri veri tabosundan sütun/satır seçmek Alt kümeler ve indexleme ile verileri filtreleme
Görselden sadece bir alanı işlemek (örneğin bir yüz bölgesi) 2D slicing ile alan seçimi

Yardımı

Concatenation, Sorting, Bölme

Gerçek Uygulama

Farklı kaynaklardan gelen veri kümelerini birleştirme

Büyük veri setlerini sıraya koymak (örn. en yüksek not, en sıcak gün)

Verileri kategorilere veya gruplara ayırmak

Yardımı

`concatenate()` ile yatay/dikey birleştirme

`sort()` ve `argsort()` kullanımı

`split()` ve slicing teknikleri ile parçalama

Bu Temeller Ne Sağlar?

Makine Öğrenmesi'nde model eğitimi için veri hazırlama

Pandas ve Matplotlib gibi kütüphaneleri anlamak için sağlam bir alt yapı

Veri Temizleme ve Özellik Mühendisliği (Feature Engineering) işlemlerine giriş

Oyun tasarımı, görsel işleme, finansal analiz gibi özel alanlara temel oluşturma