

#01. 개요

[1] 도수

특정한 구간 또는 범주에 속하는 자료의 개수

예를 들어, 어떤 학급의 학생들의 시험 점수를 기록한다면 각 점수 대역에 속하는 학생의 수를 세어 도수를 구할 수 있다.

[2] 도수분포표

도수들을 정리하여 구간별 도수를 표로 나타낸 것

도수 분포는 평균, 중앙값, 최빈값과 같은 중심경향성 통계량을 계산하거나 자료의 분산과 퍼짐 정도를 파악하는 데에도 사용된다.

이를 통해 데이터의 특성을 더 잘 이해하고 추론할 수 있다.

#02. 준비과정

[1] 패키지 참조

```
import numpy as np
```

[2] 샘플 데이터 준비 어느 학급의 시험 점수

```
point = np.array([100, 91, 89, 86, 84, 79, 78, 77, 74, 71, 69, 66, 65, 60, 58, 57, 55])
```

```
array([100, 91, 89, 86, 84, 79, 78, 77, 74, 71, 69, 66, 65, 60, 58, 57, 55])
```

#03. Numpy의 histogram : 도수분포도

[1] 5개의 구간으로 나누기

```
hist, bins = np.histogram(point, 5)
print("구간별 데이터 수:", hist)
print("구간 경계:", bins)
```

```
구간별 데이터 수: [4 4 4 3 2]
구간 경계: [ 55.  64.  73.  82.  91. 100.]
```

```
# 마지막만 미만이 아닌 '이하' 이므로
s = len(hist)
for i in range(0,s):
    if i + 1 < s:
        tpl = "%d이상 %d미만: %d개"
    else:
        tpl = "%d이상 %d이하: %d개"

    print(tpl % (bins[i], bins[i+1], hist[i]))
```

```
55이상 64미만: 4개
64이상 73미만: 4개
73이상 82미만: 4개
82이상 91미만: 3개
91이상 100이하: 2개
```

[2] 데이터 구간을 직접 정의하기

```
hist, bins = np.histogram(point, [50, 60, 70, 80, 90, 100])
print("구간별 데이터 수:", hist)
print("구간 경계:", bins)
```

```
구간별 데이터 수: [3 4 5 3 2]
구간 경계: [ 50  60  70  80  90 100]
```