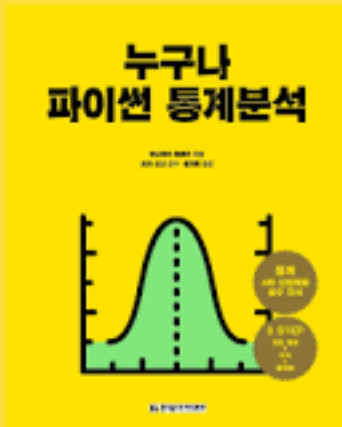


빅데이터 분석

-통계분석2-

강의자료 출처:



임경태

CHAPTER 03

----- 2차원 데이터 정리

CONTENTS

- 3.1 두 데이터 사이의 관계를 나타내는 지표
- 3.2 2차원 데이터의 시각화
- 3.3 앤스컴의 예

2차원 데이터에 대한 이해

- 1차원 데이터: 영어 점수
- 2차원 데이터: 영어, 수학 점수

- 영어, 수학의 대표 값인 평균, 중간 값, 편차, 분산, 표준편차는 어떻게 다를까?

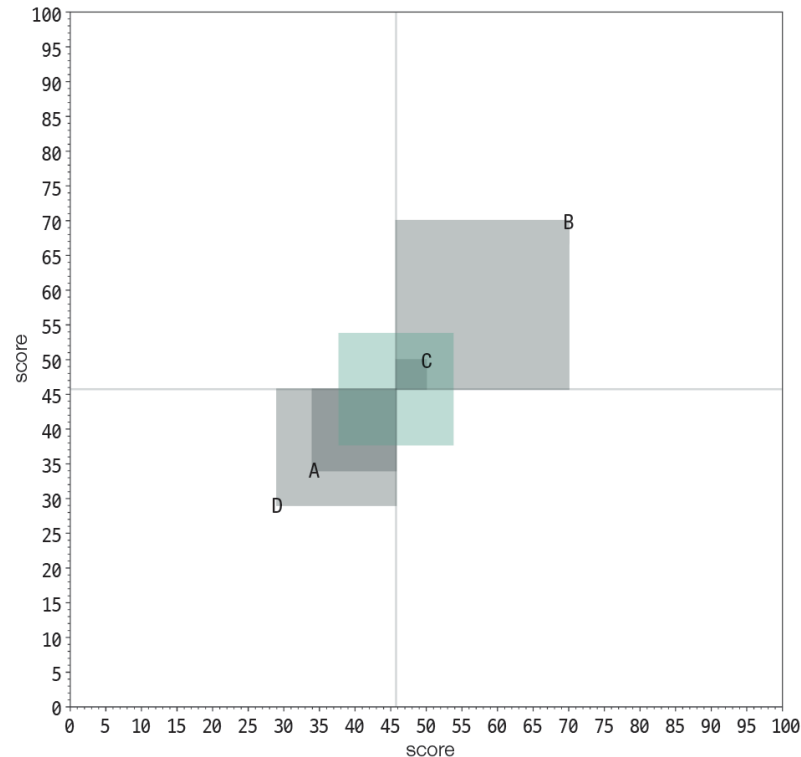
- 영어, 수학 점수는 어떤 상관관계가 있을까? **(분석)**
 - 영어를 잘하는 친구들이 수학도 잘할까?
 - 영어를 잘하면 수학을 잘할까??
 - 아니면 특정 친구들이 그냥 똑똑한 걸까?
 - 머리가 똑똑한 걸까 아니면 노력을 많이 한걸까?
 - 부모가 영어 수학을 잘해서 유전을 타고난 걸까?
 - 잘생기면 영어, 수학을 더 잘할 수 있을까?

- 과연 나의 영어점수, 공부 시간, 외모에 따라 수학을 몇점 받을 수 있을까? **(예측)**

3.0.1 분산과 표준편차

- 분산

- 편차 제곱은 한 변의 길이가 편차인 정사각형의 면적으로 간주하면, 분산은 면적의 평균



[그림 2-3] 분산 SAMPLE CODE ⁴

- 중앙의 가로선과 세로선은 4명의 평균점수
- A, B, C, D 각각은 시험 점수
- 각 회색의 정사각형이 편차 제곱
- 정사각형의 평균이 중앙의 정사각형
- 중앙 정사각형의 면적이 분산

In [1]:

```
import numpy as np
import pandas as pd

%precision 3
pd.set_option( ' precision ' , 3)
```

In [2]:

```
df = pd.read_csv( ' ../data/ch2_scores_em.csv ' ,
                  index_col= ' student number ' )
```

In [3]:

```
en_scores = np.array(df[ ' english ' ][:10])
ma_scores = np.array(df[ ' mathematics ' ][:10])

scores_df = pd.DataFrame({ ' english ' :en_scores,
                           ' mathematics ' :ma_scores},
                          index=pd.Index([ ' A ' , ' B ' , ' C ' , ' D ' , ' E ' ,
                                           ' F ' , ' G ' , ' H ' , ' I ' , ' J ' ],
                                           name= ' student ' ))

scores_df
```

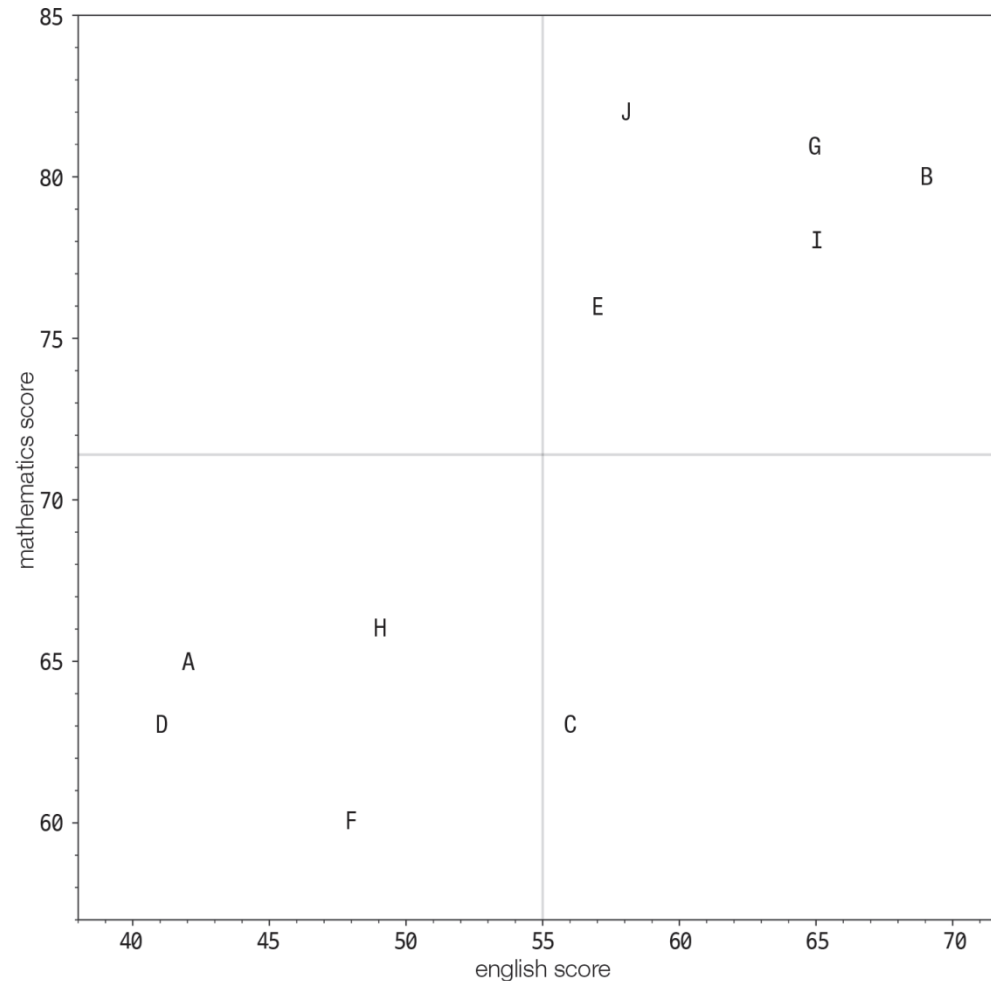
양의 상관 관계: 영어 점수가 높은 학생일수록 수학 점수가 높은 경향이 있다면 영어 점수와 수학 점수는 **양의 상관** 관계

음의 상관 관계: 영어 점수가 높은 학생일수록 수학 점수가 낮은 경향이 있다면 영어 점수와 수학 점수는 **음의 상관** 관계

무상관 관계: 영어 점수가 수학 점수에 직접적으로 영향을 미치지 않을 때, 영어 점수와 수학 점수는 **무상관**

3.1 두 데이터 사이의 관계를 나타내는 지표

3.1.1 공분산

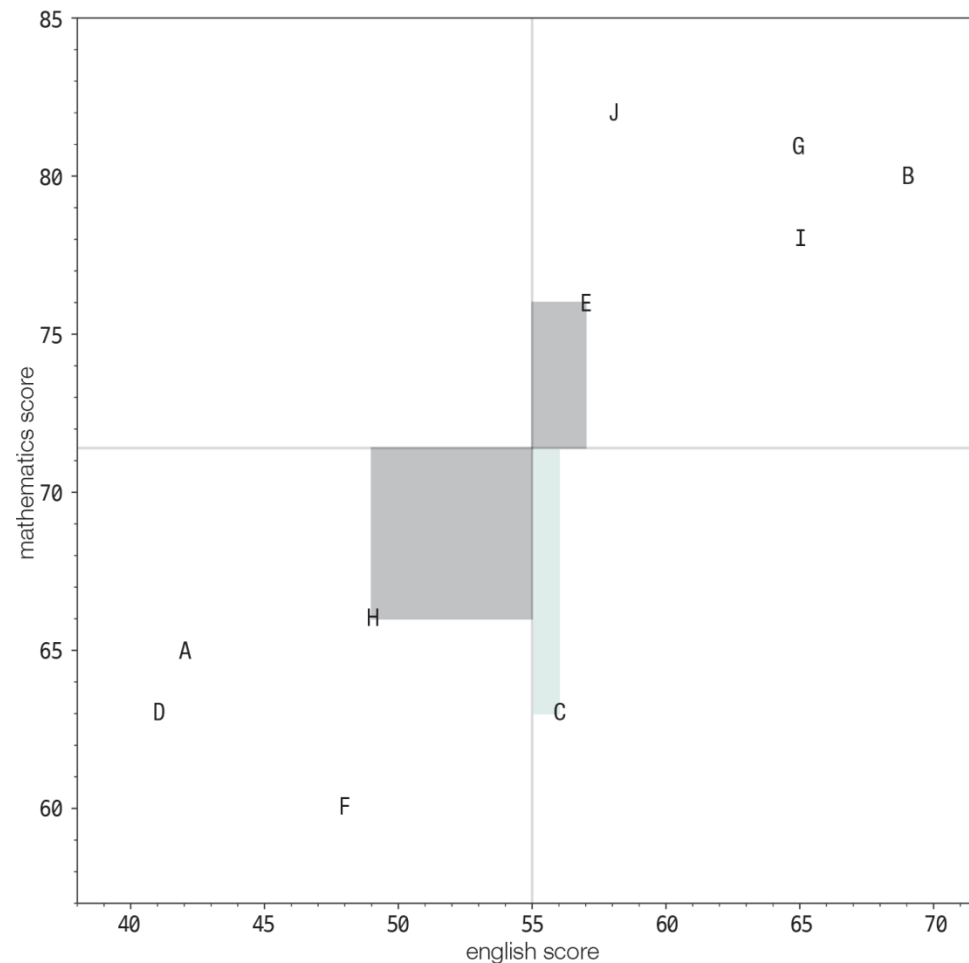


[그림 3-1] 점수의 산점도

- 중간의 가로선과 세로선은 수학과 영어 평균 점수
- 영어 점수와 수학 점수는 양의 상관 관계

3.1 두 데이터 사이의 관계를 나타내는 지표

3.1.1 공분산



[그림 3-2] 점수의 산점도와 부호를 붙인 면적 [SAMPLE CODE](#)

- 직사각형의 가로길이는 영어 점수의 편차, 세로는 수학 점수의 편차
- 공분산은 면적, 음의 면적도 가능(음의 상관)

3.1 두 데이터 사이의 관계를 나타내는 지표

3.1.1 공분산

$$\begin{aligned} S_{xy} &= \frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y}) \\ &= \frac{1}{n} \{ (x_1 - \bar{x})(y_1 - \bar{y}) + (x_2 - \bar{x})(y_2 - \bar{y}) + \cdots + (x_n - \bar{x})(y_n - \bar{y}) \} \end{aligned}$$

In [4] :

```
summary_df = scores_df.copy()
summary_df[ 'english_deviation' ] = \
    summary_df[ 'english' ] - summary_df[ 'english' ].mean()
summary_df[ 'mathematics_deviation' ] = \
    summary_df[ 'mathematics' ] - summary_df[ 'mathematics' ].mean()
summary_df[ 'product of deviations' ] = \
    summary_df[ 'english_deviation' ] * summary_df[ 'mathematics_deviation' ]
summary_df
```

In [5] :

```
summary_df[ 'product of deviations' ].mean()
```

3.1 두 데이터 사이의 관계를 나타내는 지표

3.1.1 공분산

- NumPy의 cov 함수 반환값은 공분산 행렬(분산공분산 행렬)

In [6]:

```
cov_mat = np.cov(en_scores, ma_scores, ddof=0)
cov_mat
```

Out [6]:

```
array([[86.   , 62.8 ],
       [62.8 , 68.44]])
```

- 1행 2열, 2행 1열 성분이 영어 수학의 공분산

In [7]:

```
cov_mat[0, 1], cov_mat[1, 0]
```

3.1.2 상관계수

- 공분산의 단위는 직감적으로 이해하기 어려우므로, 단위에 의존하지 않는 상관을 나타내는 지표
 - 시험 점수간의 공분산 (점수×점수), 키와 점수 (cm×점수)
- 상관계수는 공분산을 각 데이터의 표준편차로 나누어 단위에 의존하지 않음

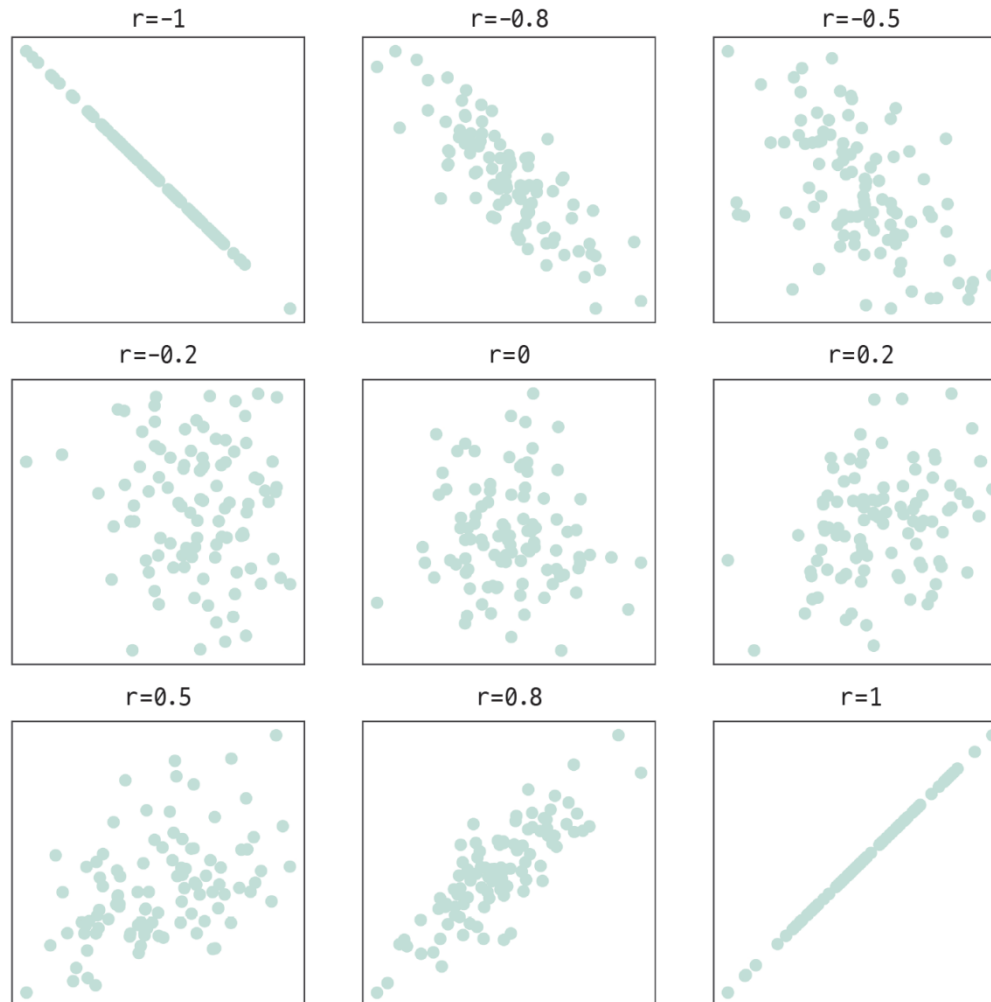
$$\begin{aligned} r_{xy} &= \frac{S_{xy}}{S_x S_y} \\ &= \frac{1}{n} \sum_{i=1}^n \left(\frac{x_i - \bar{x}}{S_x} \right) \left(\frac{y_i - \bar{y}}{S_y} \right) \end{aligned}$$

- 양의 상관은 1에 가까워지고, 음의 상관은 -1에 가까워지고, 무상관은 0

3.1 두 데이터 사이의 관계를 나타내는 지표

3.1.2 상관계수

- 양의 상관은 1에 가까워지고, 음의 상관은 -1에 가까워지고, 무상관은 0
- 상관계수가 -1일 때와 1일 때 데이터는 완전히 직선상에 놓임



[그림 3-3] 상관계수

3.1 두 데이터 사이의 관계를 나타내는 지표

3.1.2 상관계수

- 수식대로 계산하는 영어 점수와 수학 점수의 상관계수

In [10]:

```
np.cov(en_scores, ma_scores, ddof=0)[0, 1] /\n    (np.std(en_scores) * np.std(ma_scores))
```

- NumPy의 `corrcoef` 함수(상관행렬의 [0,1] [1,0] 성분)

In [11]:

```
np.corrcoef(en_scores, ma_scores)
```

- DataFrame의 `corr` 메서드

In [12]:

```
scores_df.corr()
```

3.2.1 산점도

In [13]:

```
import matplotlib.pyplot as plt
```

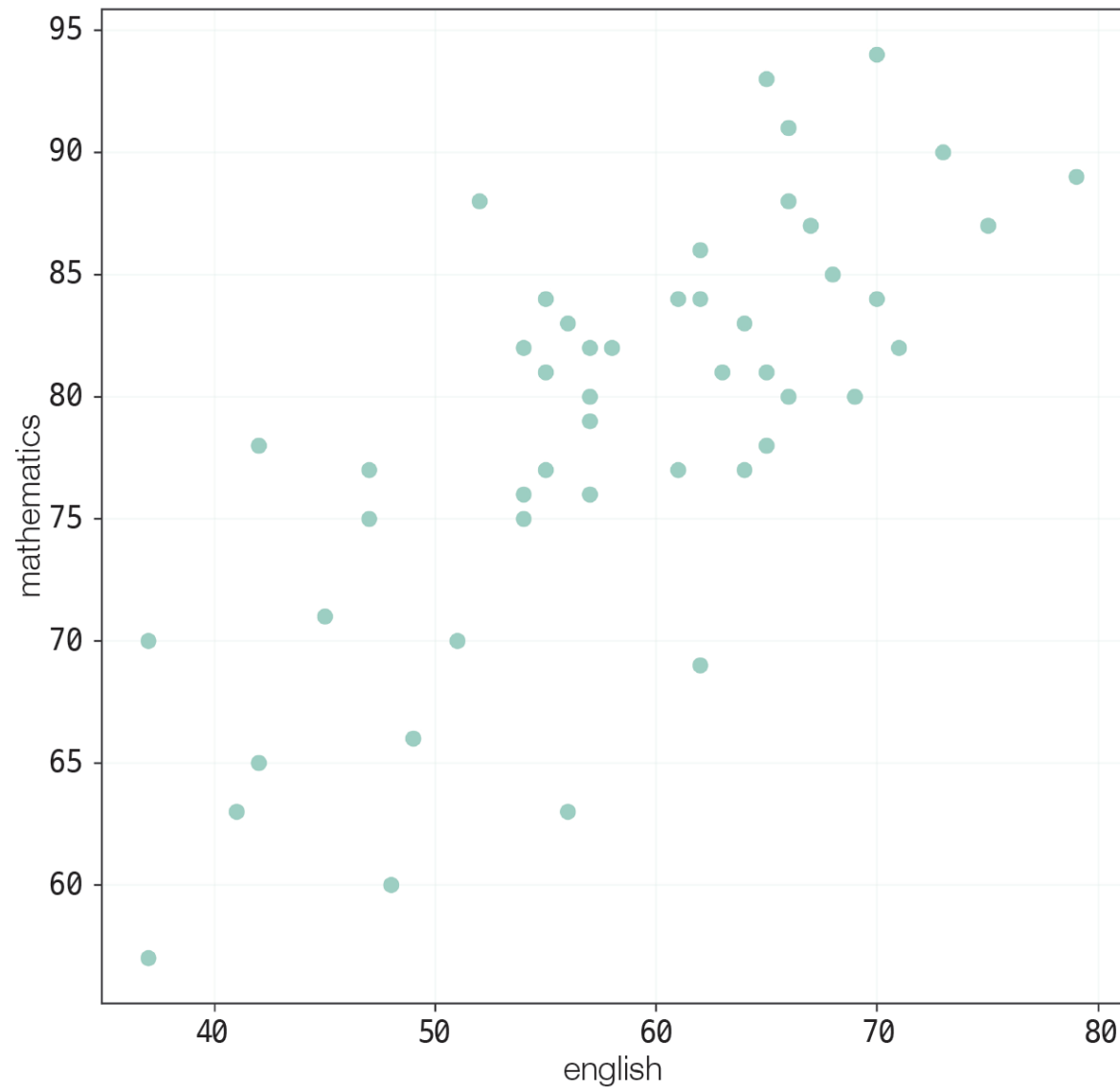
```
%matplotlib inline
```

In [14]:

```
english_scores = np.array(df[ ' english ' ])  
math_scores = np.array(df[ ' mathematics ' ])
```

```
fig = plt.figure(figsize=(8, 8))  
ax = fig.add_subplot(111)  
# 산점도  
ax.scatter(english_scores, math_scores)  
ax.set_xlabel( ' english ' )  
ax.set_ylabel( ' mathematics ' )  
  
plt.show()
```

3.2.1 산점도



[그림 3-4] 산점도

3.2.2 회귀직선

$$y = \beta_0 + \beta_1 x$$

In [15]:

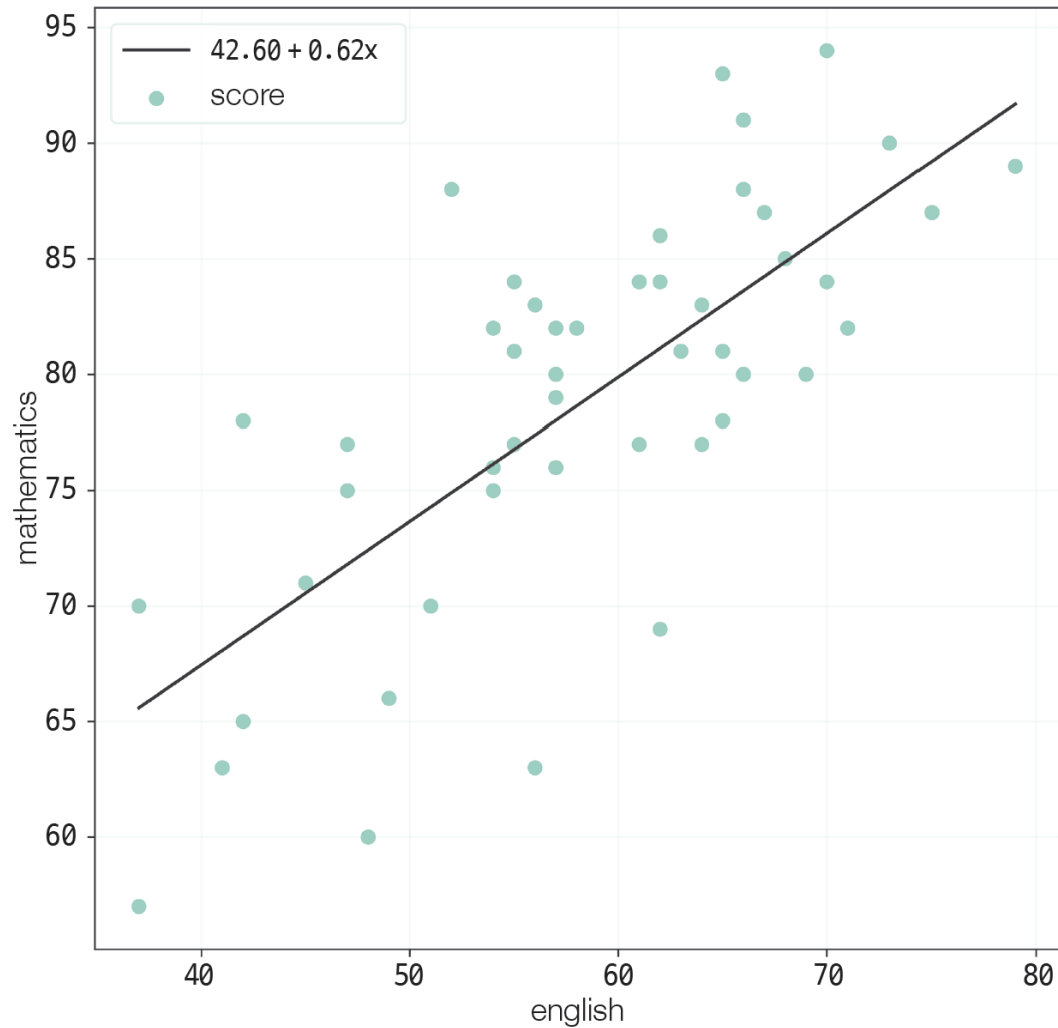
```
# 계수  $\beta_0$ 과  $\beta_1$ 을 구한다
poly_fit = np.polyfit(english_scores, math_scores, 1)
#  $\beta_0 + \beta_1 x$ 를 반환하는 함수를 작성
poly_1d = np.poly1d(poly_fit)
# 직선을 그리기 위해 x좌표를 생성
xs = np.linspace(english_scores.min(), english_scores.max())
# xs에 대응하는 y좌표를 구한다
ys = poly_1d(xs)

fig = plt.figure(figsize=(8, 8))
ax = fig.add_subplot(111)
ax.scatter(english_scores, math_scores, label='score')
ax.plot(xs, ys, color='gray',
        label=f'{poly_fit[1]:.2f}+{poly_fit[0]:.2f}x')
ax.set_xlabel('english')
ax.set_ylabel('mathematics')
# 범례 표시
ax.legend(loc='upper left')

plt.show()
```


3.2.2 회귀직선

$$y = \beta_0 + \beta_1 x$$



[그림 3-5] 산점도와 회귀직선

3.2.3 히트맵

- 히스토그램의 2차원 버전으로 색을 이용해 표현하는 그래프
- 영어 점수 35점부터 80점, 수학 점수 55점부터 95점까지 5점 간격

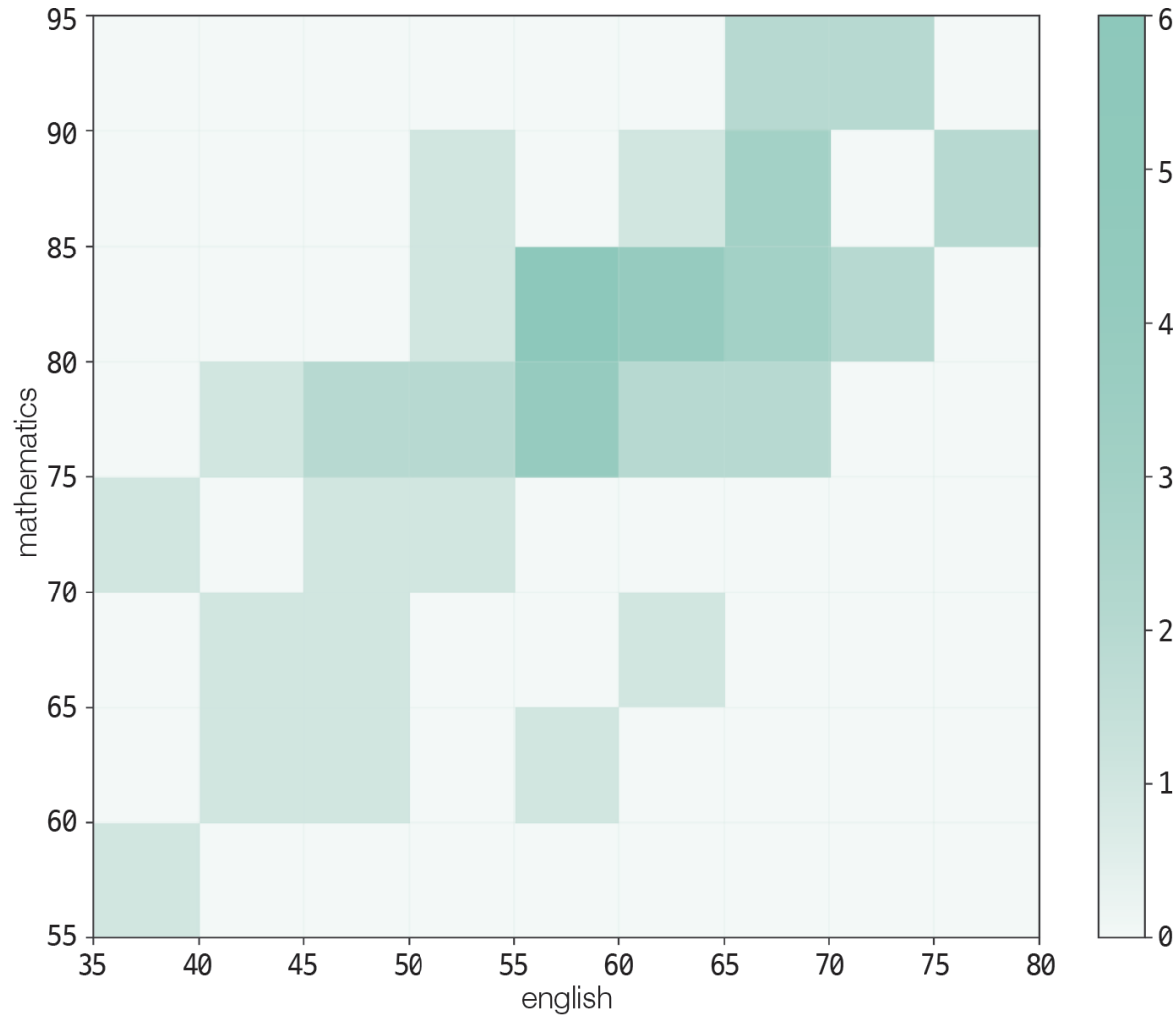
In [16] :

```
fig = plt.figure(figsize=(10, 8))
ax = fig.add_subplot(111)

c = ax.hist2d(english_scores, math_scores,
               bins=[9, 8], range=[(35, 80), (55, 95)])
ax.set_xlabel( ' english ' )
ax.set_ylabel( ' mathematics ' )
ax.set_xticks(c[1])
ax.set_yticks(c[2])
# 컬러 바 표시
fig.colorbar(c[3], ax=ax)
plt.show()
```

3.2.3 히트맵

- 색이 진한 영역일수록 많은 학생이 분포



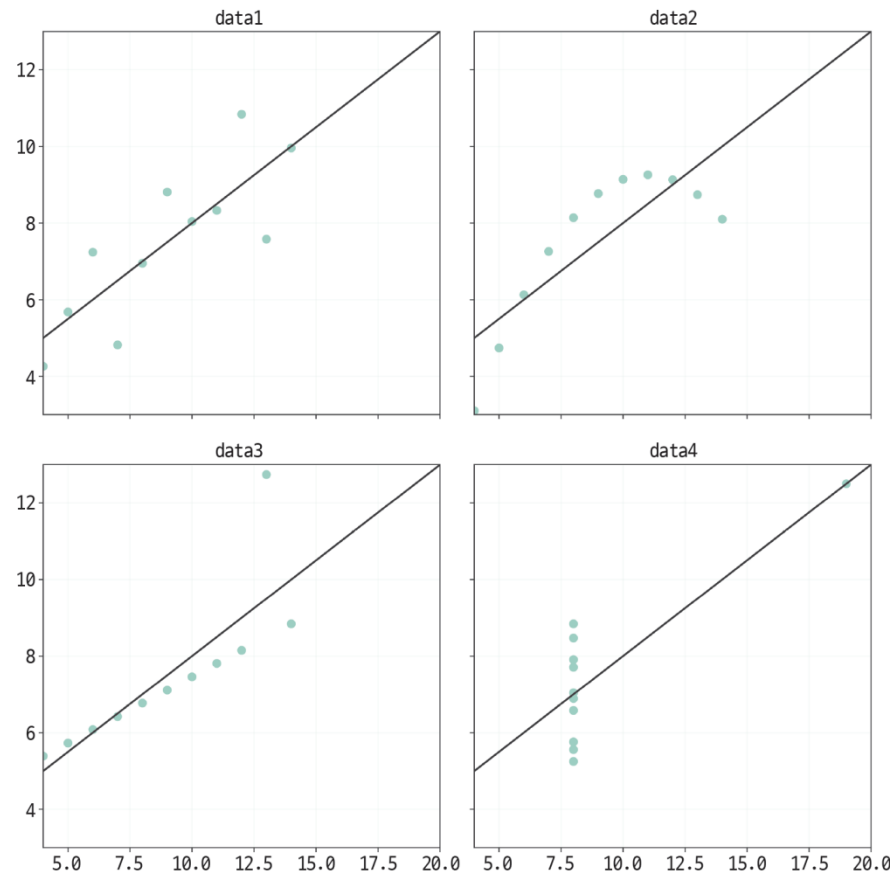
[그림 3-6] 히트맵

동일한 지표를 가지고 있지만 그림으로 표현하면 전혀 다른 데이터

	data1	data2	data3	data4
X_mean	9.00	9.00	9.00	9.00
X_variance	10.00	10.00	10.00	10.00
Y_mean	7.50	7.50	7.50	7.50
Y_variance	3.75	3.75	3.75	3.75
X&Y_correlation	0.82	0.82	0.82	0.82
X&Y_regression line	$3.00+0.50x$	$3.00+0.50x$	$3.00+0.50x$	$3.00+0.50x$

동일한 지표를 가지고 있지만 그림으로 표현하면 전혀 다른 데이터

	data1	data2	data3	data4
X_mean	9.00	9.00	9.00	9.00
X_variance	10.00	10.00	10.00	10.00
Y_mean	7.50	7.50	7.50	7.50
Y_variance	3.75	3.75	3.75	3.75
X&Y_correlation	0.82	0.82	0.82	0.82
X&Y_regression line	$3.00+0.50x$	$3.00+0.50x$	$3.00+0.50x$	$3.00+0.50x$



[그림 3-7] 앤스컴의 예

Q&A

ktlim@hanbat.ac.kr