

상관계수 개념

두 양적(순서형 포함) 변수(X, Y)간의 직선 관계 정도를 계수로 측정함

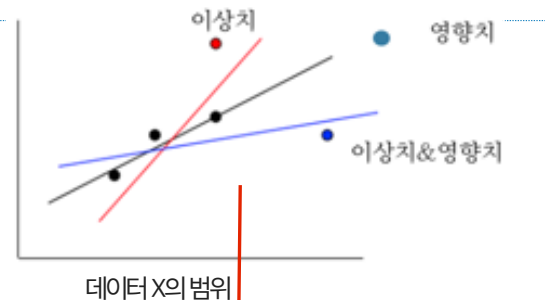
직선(함수)관계가 유의하다는(한 변수가 증가하면 다른 변수도 직선적으로 증가하거나 감소함) 것은 두 변수가 유사하다는 의미 - **변수의 유사성** 척도

산점도 scatter plot

정의

2개의 측정형 변수 데이터를 2차원 공간에 표현하여 두 변수의 함수 관계를 예상함

- X-축 : 결정의 요인, 예측변수, 독립변수, 예측변수
- Y-축 : Output, 목표변수, 목표변수 <- 목표변수가 없는 경우 두 변수 간의 직선관계만 본다.



진단내용

- 두 변수 간의 함수 관계를 본다. -> 각 변수의 확률분포함수도 제공하는 그래픽 툴이 있다.
- 이상치와 영향치를 시각적으로 진단한다.

이상치 outlier

- 선형 함수 관계에서 적합 직선을 많이 벗어난 관측값 - 실제 오차의 분산 기준 $2 \cdot \sigma$ 를 벗어남
- 예측변수 값은 관측 값의 범위 내에 있음

(진단) 오차의 추정치인 Studentized 잔차가 ± 2 벗어남 - 실증적법칙(좌우대칭인 분포의 정규분포 가정)

(해결) 삭제 - 물론 잔차분석 후에 실기

영향치 influential

- 예측변수 값이 극단 값(다른 관측치와 떨어져 있고 두 변수의 함수 관계에 영향을 주는 관측값)
- 순수 영향치 : 함수 회귀 추정 식 상에 있어 함수 관계(기울기 변동)에는 영향을 주지 않으나 결정계수 높여 예측변수의 설명 능력을 과다하게 높은 것으로 판단하게 하는 결과 왜곡

(진단) 잔차분석 - Hat 통계량 활용

(해결) 영향치 주변의 관측값을 추가 수집 후 분석, 영향치 값이 실제 발생 가능하지 않은 경우 제외



상관계수

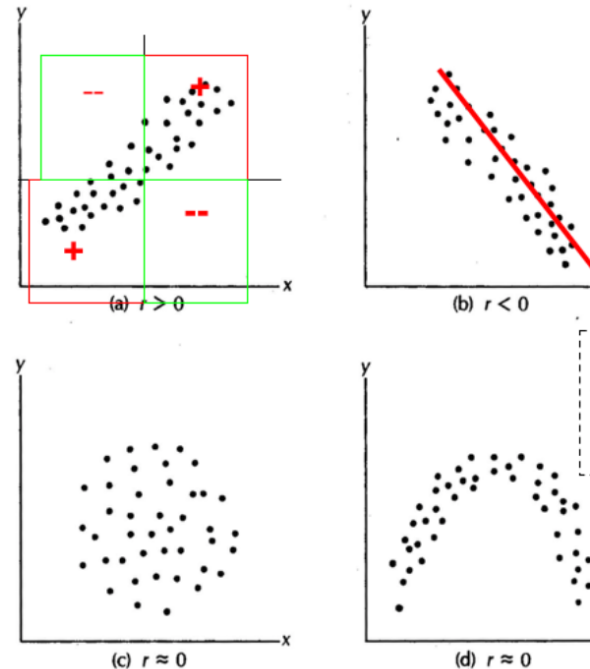
Karl Pearson 공식

$$\rho = \frac{COV(X, Y)}{\sqrt{V(X)}\sqrt{V(Y)}}$$

• 데이터 계산식 :

$$r = \frac{\sum_i^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_i^n (x_i - \bar{x})^2} \sqrt{\sum_i^n (y_i - \bar{y})^2}}$$

• 분모는 확률변수의 표준편차이므로 상관계수의 부호를 결정하는 분자항(공변량) $\sum_i^n (x_i - \bar{x})(y_i - \bar{y})$ 의 부호이다



Spearman 순위 상관계수

• (방법 1) $r_s = \text{Corr}(R_{X_i}, R_{Y_i})$ where R_{X_i} 는 X_i 의 크기 순위이며, R_{Y_i} 는 Y_i 의 크기 순위이다.

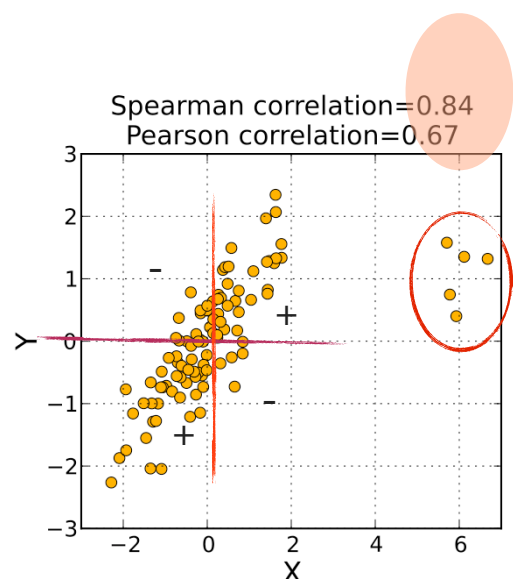
$$\text{(방법 2)} \quad r_s = 1 - \frac{6 \sum d_i^2}{n(n^2 - 1)}, \quad d_i = R_{X_i} - R_{Y_i}$$

• 대부분의 데이터 범위 밖에 있는 관측치(타원형 내 관측치)는 상관계수 값을 높이는 역할을 한다. 그러므로 상관계수를 계산하기 전에 반드시 산점도를 그려 데이터의 범위를 많이 벗어난 관측치가 있는지 확인하여 상관분석의 활용도를 높일 필요가 있음.

두 상관계수의 관계

이상치가 존재하는 경우 값의 크기에 의해 상관관계 척도를 계산하는 피어슨 상관계수 값이 작아진다.

그러므로 피어슨 상관관계를 활용하는 경우



Kendall τ 상관계수

$$\tau = \frac{\#of_concordant_pairs - \#of_discordant_pairs}{n(n-1)/2}$$

- concordant = 만약 $(x_i > x_j), (y_i > y_j)$ 이거나 $(x_i < x_j)$ 이면 $(y_i < y_j)$ 이면 두 관측치는 concordant 쌍이라 함
- τ 값이 클수록 데이터 순위의 일치도는 높아지므로 상관관계가 높아진다.

상관계수 유의성 검정

가설 $H_0 : \rho = 0$

- 귀무가설 : 두 변수의 직선 상관관계는 유의하지 않다. $\Leftrightarrow$ 서로 독립이다. $H_0 : \rho = 0$
- 대립가설 : 두 변수의 직선 상관관계는 유의하다. $H_0 : \rho \neq 0$

데이터 검증

- 데이터는 이변량 정규분포에 근사해야 한다. 단 $n > 20$ 인 대표본에서는 문제 없음
- 산점도를 그려 데이터 범위(X-) 밖의 관측치 존재 여부를 체크한다. - 존재한다면 제외하거나 활용 시 주의해야 한다.

검정통계량

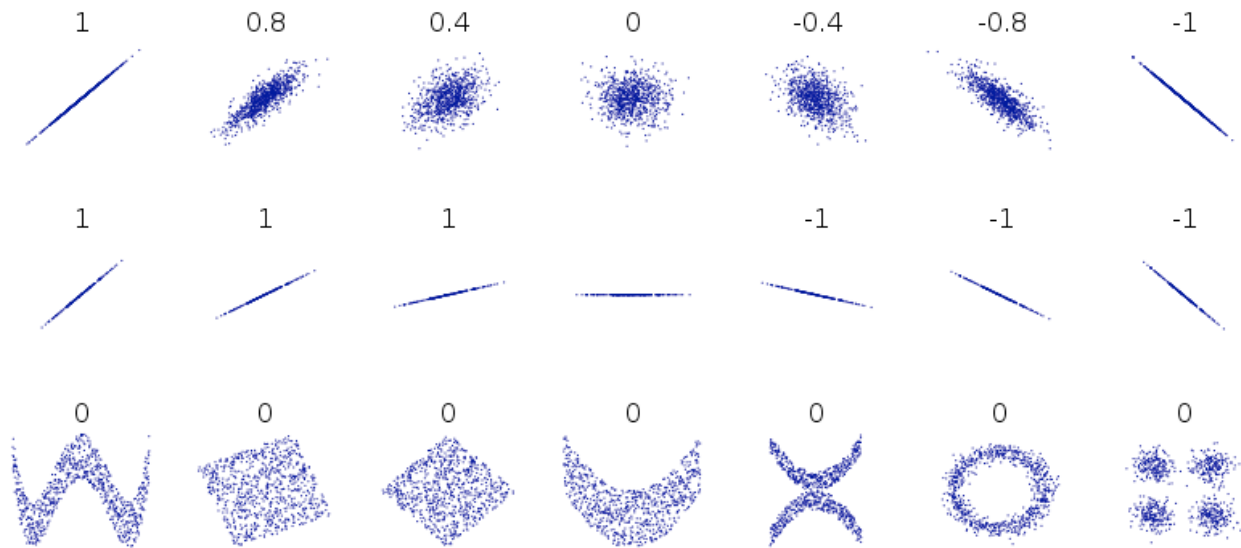
$$TS = \frac{r}{\sqrt{(1-r^2)(n-2)}} \sim t(n-2), n=\text{표본크기}, r=\text{상관계수}$$

결론

- 유의확률 $P(t(n-2) > |TS|)$ 이 유의수준보다 작다면 귀무가설을 기각하여 상관관계의 유의하다고 결론내리고 표본상관계수의 부호를 이용하여 해석
- 귀무가설이 기각, 표본상관계수 부호 $+$ $\Rightarrow$ 두 변수는 양의 상관관계가 있고 한 변수의 값이 증가(감소)하면 다른 변수의 값도 증가(감소)한다.
- 귀무가설이 기각, 표본상관계수 부호 $-$ $\Rightarrow$ 두 변수는 음의 상관관계가 있고 한 변수의 값이 증가(감소)하면 다른 변수의 값도 감소(증가)한다.



상관계수 해석



*) 출처 : 위키피디아

- -1과 1사이의 값이다.
- 1에 가까우면 양의 선형 상관 관계가 존재한다. 한 변수의 값이 증가(감소)하면 다른 변수 값도 증가(감소)한다.
- -1에 가까우면 음의 선형 상관 관계가 존재한다. 한 변수의 값이 증가(감소)하면 다른 변수 값은 감소(증가)한다.
- 두 변수의 상관 관계가 높다는 것은 두 변수가 동일한(comparable) 개념을 측정한다는 의미도 담고 있다(두 변수가 유사함). 그러므로 변수를 축약하거나 개체를 분류하는데 사용되는 다변량 분석에서는 공분산, 혹은 상관 계수 개념 사용

Rule of thumb:

- $0.0 = |r|$: no correlation
- $0.0 < |r| < 0.2$: very weak correlation
- $0.2 \leq |r| < 0.4$: weak correlation
- $0.4 \leq |r| < 0.6$: moderately strong correlation
- $0.6 \leq |r| \leq 0.8$: strong correlation
- $0.8 \leq |r| < 1.0$: very strong correlation
- $1.0 = |r|$: perfect correlation



$H_0 : \rho = \rho_0$ 검정

- 상관관계 유의성 검정이 아니라 임의의 상관계수와 동일한지 검정
- 활용 : 미국의 경우 부자 키의 상관계수는 0.65이다. 한국의 경우 미국과 부자의 키의 상관계수가 같다고 할 수 있나? 귀무가설 : $H_0 : \rho = 0.65$

$$\text{검정통계량 : } TS = \frac{\frac{1}{2} \ln\left(\frac{1+r}{1-r}\right) - \frac{1}{2} \ln\left(\frac{1+\rho_0}{1-\rho_0}\right)}{1/\sqrt{n-3}} \sim N(0,1)$$

서로 독립인 2집단 상관계수 차이 검정 $H_0 : \rho_1 = \rho_2$

- 귀무가설 $H_0 : \rho_x = \rho_y$ vs. 대립가설 $H_0 : \rho_x \neq \rho_y$
- 활용 : 한국 부자 키의 상관계수와 미국 부자 키의 상관계수는 동일한가?

$$z(x) = 0.5 \ln \frac{1+r_x}{1-r_x}, z(y) = 0.5 \ln \frac{1+r_y}{1-r_y}$$

$$z = \frac{z(x) - z(y)}{\sqrt{1/(n_x - 3) + 1/(n_y - 3)}} \sim N(0,1)$$

• 검정통계량 :



다변량 산점도 그리기

예제 데이터 : IRIS 데이터 <https://vincentarelbundock.github.io/Rdatasets/csv/datasets/iris.csv>

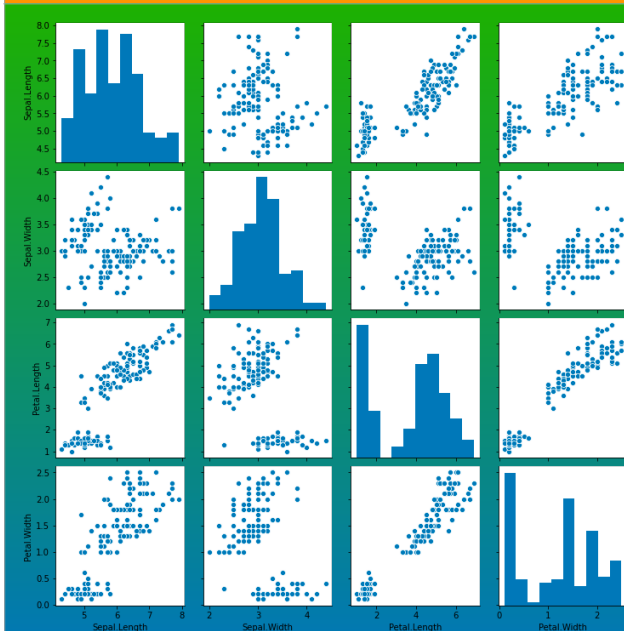
```
import pandas as pd
iris=pd.read_csv("https://vincentarelbundock.github.io/Rdatasets/csv/datasets/iris.csv")
iris=iris.iloc[:,1:6]
iris.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 5 columns):
#   Column          Non-Null Count  Dtype
---  -
0   Sepal.Length    150 non-null   float64
1   Sepal.Width     150 non-null   float64
2   Petal.Length    150 non-null   float64
3   Petal.Width     150 non-null   float64
4   Species         150 non-null   object
```

첫 열은 제외하였음

행렬 산점도

```
import seaborn as sns
import matplotlib.pyplot as plt
sns.pairplot(iris)
plt.show()
```



sepal 꽃받침 길이는 꽃잎 petal 길이와 상관관계 가장 높고 꽃잎 넓이 꽃받침 넓이 순이다.



기초통계량 구하기

```
iris.describe()
```

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width
count	150.000000	150.000000	150.000000	150.000000
mean	5.843333	3.057333	3.758000	1.199333
std	0.828066	0.435866	1.765298	0.762238
min	4.300000	2.000000	1.000000	0.100000
25%	5.100000	2.800000	1.600000	0.300000
50%	5.800000	3.000000	4.350000	1.300000
75%	6.400000	3.300000	5.100000	1.800000
max	7.900000	4.400000	6.900000	2.500000

공분산 구하기

```
iris.cov()
```

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width
Sepal.Length	0.685694	-0.042434	1.274315	0.516271
Sepal.Width	-0.042434	0.189979	-0.329656	-0.121639
Petal.Length	1.274315	-0.329656	3.116278	1.295609
Petal.Width	0.516271	-0.121639	1.295609	0.581006

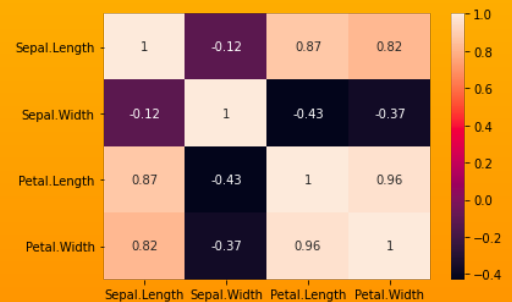
상관계수구하기

```
iris.corr(method='pearson')
```

[참고] spearman, kendall

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width
Sepal.Length	1.000000	-0.117570	0.871754	0.817941
Sepal.Width	-0.117570	1.000000	-0.428440	-0.366126
Petal.Length	0.871754	-0.428440	1.000000	0.962865
Petal.Width	0.817941	-0.366126	0.962865	1.000000

```
iris_cor=iris.corr(method='pearson')
import seaborn as sn
sn.heatmap(iris_cor,annot=True)
plt.show()
```



상관계수 유의성 검정

```
import scipy.stats as stats
stats.pearsonr(iris['Sepal.Length'],iris['Sepal.Width'])
```

```
(-0.11756978413300206, 0.15189826071144766)
```



평균벡터 구하기 $E(\underline{x})_{1 \times p} = \underline{1}^T X$

```
df_mean=np.dot(one,df_array)/n
df_mean
```

```
↳ array([5.84333333, 3.05733333, 3.758      , 1.19933333])
```

공분산행렬 구하기

중심화 행렬 : $C_{n \times p} = X_{n \times p} - \underline{1}_{n \times 1} E(X)$

공분산행렬 : $\Sigma_{p \times p} = (C^T C)/(n - 1)$

```
C=df_array-df_mean
df_cov=np.dot(C.T,C)/(n-1)
df_cov
```

```
↳ array([[ 0.68569351, -0.042434 ,  1.27431544,  0.51627069],
        [-0.042434 ,  0.18997942, -0.32965638, -0.12163937],
        [ 1.27431544, -0.32965638,  3.11627785,  1.2956094 ],
        [ 0.51627069, -0.12163937,  1.2956094 ,  0.58100626]])
```

상관계수행렬 구하기

상관계수 : $Corr(x_i, x_j) = \frac{COV(x_i, x_j)}{\sqrt{V(x_i)}\sqrt{V(x_j)}}$ 그러므로 $R = D^{-1}\Sigma D^{-1}$, 대각행렬 D는 공분산행렬의 대

각원소의 제곱근 값, 즉 각 확률변수 분산의 제곱근이므로 표준편차이다.

```
import numpy.linalg as la
D=np.diag(np.sqrt(np.diag(df_cov)))
D_inv=la.inv(D)
D_inv@df_cov@D_inv
```

```
↳ array([[ 1.          , -0.11756978,  0.87175378,  0.81794113],
        [-0.11756978,  1.          , -0.4284401 , -0.36612593],
        [ 0.87175378, -0.4284401 ,  1.          ,  0.96286543],
        [ 0.81794113, -0.36612593,  0.96286543,  1.          ]])
```