제 4 장. 모집단과 표본

4.1 정규분포

정규분포는 가우스 (Gauss, 1777-1855)에 의해 제시된 분포로써 가우스 분포 (Gauss distribution)라고 불리며 가장 대표적인 연속형 확률 분포이다. 평균이 μ이고 표준편차가 σ인 정규분포의 밀도함수는 다음과 같다.

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2\sigma^2}(x-\mu)^2}, \quad -\infty < x < \infty$$

확률변수 X가 평균이 μ, 표준편차가 σ인 정규분포를 따를 때, 이를 $X \sim N(\mu, \sigma^2)$로 나타낸다. 정규분포는 평균을 중심으로 좌우 대칭의 모양이며 대칭점과 변곡점 사이의 거리가 표준편차이다.

4.1.1 정규분포의 확률밀도함수 그리기

정규분포의 확률밀도는 scipy.stats.norm 모듈의 pdf함수를 사용한다.

In [ ]:

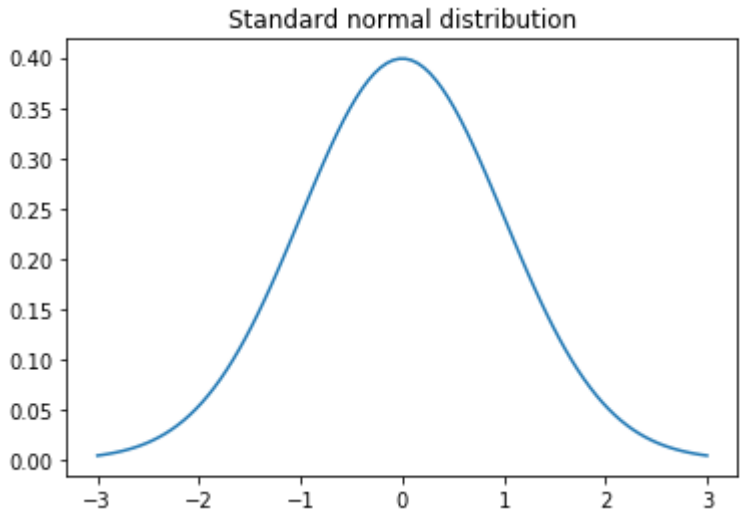
```
pdf(x, loc = 0, scale = 1)
```

: Probability density function. $X \sim N(\mu, \sigma^2)$ 일 때 $\Pr(X = x)$ 값을 구함

다음은 표준 정규분포의 확률밀도함수를 그리는 방법이다.

In [15]:

```
import numpy as np
from scipy.stats import norm
import matplotlib.pyplot as plt

x = np.linspace(-3, 3, 100) # linespace(start, stop, num): start에서 stop까지 num개 만큼의 수열을 생성
fx = norm.pdf(x, loc = 0, scale = 1)
plt.plot(x, fx) # plot(x,y): (x,y)의 산점도를 선으로 연결하여 그림
plt.title("Standard normal distribution")

plt.show()
```



4.1.2 정규분포의 확률 계산

정규분포의 누적분포는 scipy.stats.norm 모듈의 cdf함수를 사용한다.

In [ ]:

```
cdf(x, loc = 0, scale = 1)
```

: Cumulative distribution function. $X \sim N(\mu, \sigma^2)$ 일 때 $\Pr(X \leq x)$ 값을 구함

4.1.3 정규 분위수의 계산

정규분포의 분위수는 scipy.stats.norm 모듈의 ppf함수를 사용한다.

In [ ]:

```
ppf(q, loc=0, scale=1)
```

: Percent point function (inverse of cdf-percentiles). $X \sim N(\mu, \sigma^2)$ 일 때 $\Pr(X \leq x) = q$를 만족하는 x를 구함

4.1.4 정규분포에서 난수 생성하기

정규분포에서의 난수생성은 scipy.stats.norm 모듈의 rvs함수를 사용한다.

In [ ]:

```
rvs(loc=0, scale=1, size=1)
```

4.2 이항 분포

베르누이 시행: 시행의 결과가 (성공, 실패)와 같이 두 가지 결과 중의 하나로 나타나는 시행

이항 분포: 성공 확률이 p인 베르누이 시행을 n회 반복할 때, "X = 성공의 횟수"의 분포

시행 횟수가 n이고 성공의 확률이 p인 이항분포의 확률밀도함수는 다음과 같다.

$$p(x) = \binom{n}{x} p^x (1-p)^{n-x}, \quad x = 0, 1, 2, \cdots, n$$

이항분포의 평균, 분산

: X가 이항분포 $B(n, p)$를 따를 때

$$E(X) = np, \quad Var(X) = np(1-p)$$

이항분포의 정규근사

: X가 이항분포 $B(n, p)$를 따를 때

(1) $X \sim N(np, np(1-p))$

(2) $Z = \frac{X-np}{\sqrt{np(1-p)}} = \frac{\hat{p}-p}{\sqrt{p(1-p)/n}} \simeq N(0, 1)$

참고. 이항분포의 정규근사는 n이 충분히 커서 "$np \geq 5$ 이고 $n(1-p) \geq 5$"일 때 사용하는 것이 안전하다.

4.2.1 이항 분포의 밀도함수 그리기

이항분포의 확률밀도는 scipy.stats.binom 모듈의 pmf함수를 사용한다.

In [ ]:

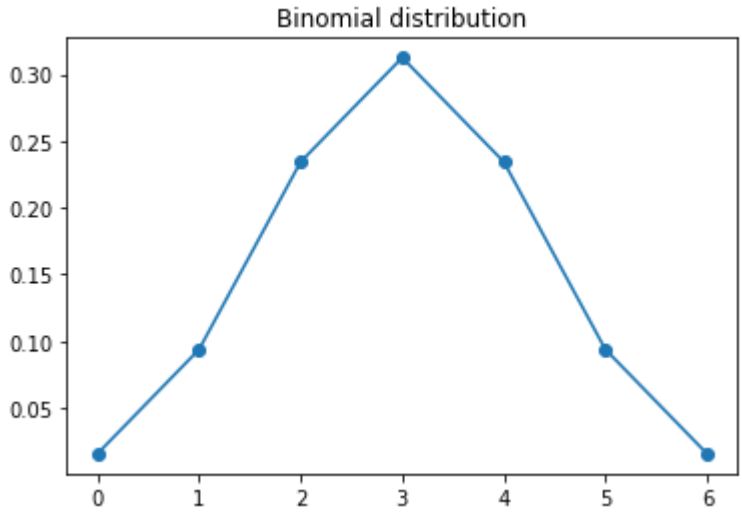
```
pmf(k, n, p, loc = 0)
```

: Probability mass function. $X \sim B(n,p)$일 때 $\Pr(X = k)$ 값을 구함

이항분포의 확률밀도함수는 다음과 같이 그릴 수 있다.

In [5]:

```
import numpy as np
from scipy.stats import binom
import matplotlib.pyplot as plt

k = np.arange(0,7)
px = binom.pmf(k, n = 6, p = 0.5)
plt.plot(k, px, 'COo')
plt.plot(k, px, linestyle="-", color = "CO")
plt.title("Binomial distribution")

plt.show()
```



4.2.2 이항 분포의 확률 계산

이항 분포의 누적분포는 scipy.stats.binom 모듈의 cdf함수를 사용한다.

In [ ]:

```
cdf(k, n, p, loc=0)
```

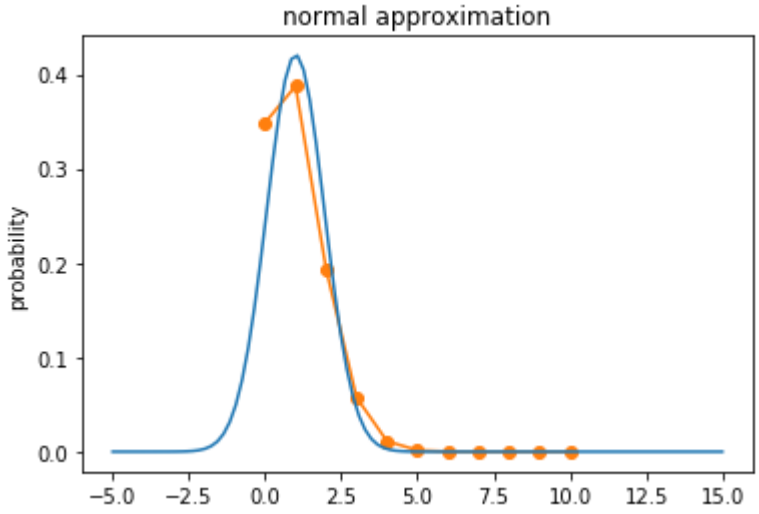
: Cumulative distribution function. $X \sim B(n,p)$일 때 $\Pr(X \leq k)$ 값을 구함

4.2.3 이항 분포의 정규 근사

표본의 크기에 따라 이항분포의 정규근사가 어떻게 달라지는지 확인해보자.

In [4]:

```
import numpy as np
from scipy.stats import binom
from scipy.stats import norm
import matplotlib.pyplot as plt

p = 0.1
n = 10


k = np.arange(0, n+1)
px = binom.pmf(k, n = n, p = p)
plt.plot(k, px, 'C1o')
plt.plot(k, px, linestyle="-", color = "C1") # 이항 분포의 그래프를 그리는 부분

x = np.linspace(-5, 15, 100)
mu = n * p
sd = np.sqrt(n * p * (1-p))
fx = norm.pdf(x, loc = mu, scale = sd)
plt.plot(x, fx, color = "C0", linestyle = "-") # 근사된 정규분포의 그래프를 그리는 부분

plt.title("normal approximation")
plt.ylabel("probability")

plt.show()
```



4.3 표본 평균의 분포

1) 모집단의 분포가 정규분포인 경우

모집단의 분포가 정규분포 $N(\mu, \sigma^2)$ 일 때, 표본평균 $\bar{X}$는 정규분포 $N(\mu, \frac{\sigma^2}{n})$ 를 따른다.

2) 모집단의 분포가 정규분포가 아닌 경우

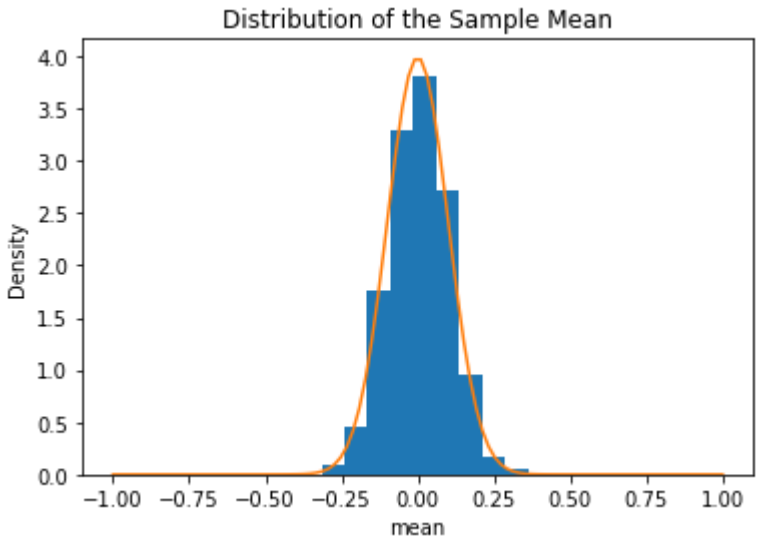
평균이 μ이고 분산이 σ^2 인 임의의 무한 모집단에서 표본의 크기 n이 충분히 크면, 랜덤 표본의 표본평균 $\overline{X}$는 근사적으로 정규분포 $N(\mu, \frac{\sigma^2}{n})$를 따른다.

: **중심극한정리 (Central Limit Theorem)**

예) 정규분포 $N(0,1)$에서 100개의 표본을 추출하여 표본평균 $\overline{X}$를 구하는 실험을 1000회 반복하고, 이들 1000개의 표본평균들을 히스토그램으로 나타내보자.

In [40]:

```
from scipy.stats import norm

np.random.seed(1) # 반복 가능한 난수 생성하기. 같은 시드값을 사용하여 항상 동일한 결과를 보여준다.

n = 100              # 1회 시행에서 추출할 표본의 개수
mean = []          # 각 시행에서 추출된 표본의 평균을 저장할 리스트

for i in range(1000):                          # 총 1000번의 반복시행을 위한 반복문. 강의영상에는 1001로 되어있는데 10000이 맞습니다.
    x = norm.rvs(loc=0, scale=1, size=n)     # (loc,scale)를 모수로 갖는 정규분포에서 n개의 랜덤 표본을 생성한다
    mean.append(x.mean())


plt.hist(mean, bins = 9, color = "C0", density = True)
plt.title("Distribution of the Sample Mean")
plt.xlabel("mean")
plt.ylabel("Density")

x = np.linspace(-1, 1, 100)
mu = 0                                                  # N(0,1) 정규분포의 모평균
sd = 1                                                  # N(0,1) 정규분포의 모표준편차
fx = norm.pdf(x, loc = mu, scale = sd/np.sqrt(n))
plt.plot(x, fx, color = "C1", linestyle="-")   # 표본평균이 근사적으로 따르는 정규 분포

plt.show()
```

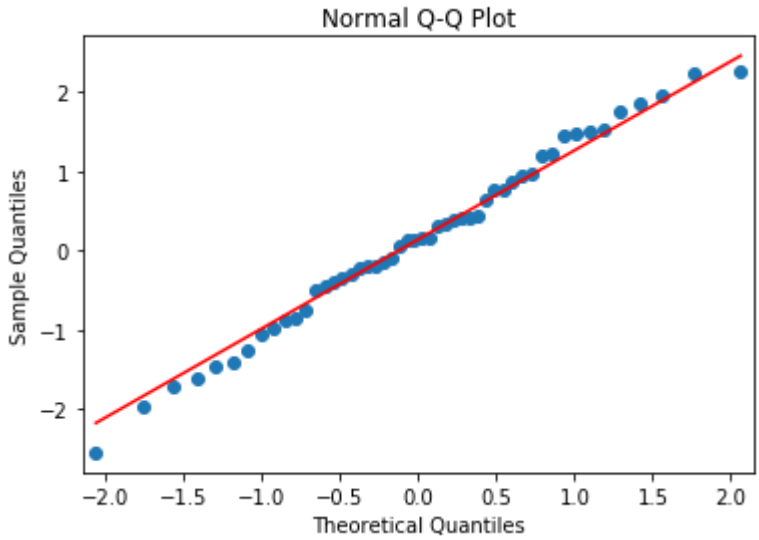


4.4 정규분포 분위수 대조도

: 정규모집단의 가정을 검토하는 방법으로 정규분포의 분위수와 이에 대응하는 자료 분포의 분위수를 좌표평면에 나타낸 그림을 말한다. 점들이 직선 주위에 밀집하여 나타날수록 모집단의 분포가 정규분포라는 가정이 타당하다고 할 수 있다.

In [12]:

```
import matplotlib.pyplot as plt
from statsmodels.graphics.gofplots import ProbPlot
from scipy.stats import norm

np.random.seed(0)
x = norm.rvs(0,1,50)
#x = np.linspace(-1, 1, 50)
QQ = ProbPlot(x)
plot = QQ.qqplot(line = 's', color='C0', lw=1)
plt.title("Normal Q-Q Plot")                    # 정규분포 분위수를 그리는 부분

plt.show()
```



예) (bodydims.csv) 주어진 자료는 247명의 남성과 260명의 여성에 관한 신체 측정 자료이다. 원 자료는 신체의 각 부위를 측정한 총 25개의 변수가 있으며 본 예제에서는 그 중 8개 변수만 간추린 자료를 이용하기로 한다. 아래는 8개의 변수에 대한 설명이다. (원자료에 관한 자세한 설명은 다음을 참조하도록 한다. : http://www.openintro.org/stat/data/bdims.php (http://www.openintro.org/stat/data/bdims.php))

- bii.di : 숫자형 변수, 응답자의 골반의 넓이 (cm)
- che.de : 숫자형 변수, 응답자의 가슴 깊이 (cm)
- elb.di : 숫자형 변수, 응답자의 양쪽 팔꿈치 지름의 합. (cm)
- kne.di : 숫자형 변수, 응답자의 양쪽 무릎의 지름의 합. (cm)
- age : 숫자형 변수, 응답자의 나이 (years)
- wgt : 숫자형 변수, 응답자의 몸무게 (kg)
- hgt : 숫자형 변수, 응답자의 신장 (cm)
- sex : 범주형 변수, 응답자의 성별 (1 = 남성, 0 = 여성)

csv 형태의 자료를 읽기 위해서는 pandas 패키지의 read_csv 함수를 사용한다.

In [ ]:

```
read_csv("파일 저장 경로")
```

다음은 bodydims.csv 파일을 불러와서 bodydims 라는 변수명으로 저장한 결과이다. 자료를 확인해 보면 총 507개의 관찰치와 8개의 변수로 구성된 것을 볼 수 있다.

In [14]:

```
import pandas as pd
bodydims = pd.read_csv("D:/bodydims.csv")
```

In [15]:

```
bodydims.shape
```

Out[15]:

(507, 8)

In [16]:

```
bodydims.head(10)
```

Out[16]:

	bii.di	che.de	elb.di	kne.di	age	wgt	hgt	sex
0	26.0	17.7	13.1	18.8	21	65.6	174.0	1
1	28.5	16.9	14.0	20.6	23	71.8	175.3	1
2	28.2	20.9	13.9	19.7	28	80.7	193.5	1
3	29.9	18.4	13.9	20.9	23	72.6	186.5	1
4	29.9	21.5	15.2	20.7	22	78.8	187.2	1
5	27.0	19.6	14.0	18.8	21	74.8	181.5	1
6	30.0	21.9	16.1	20.8	26	86.4	184.0	1
7	29.8	21.8	15.1	21.0	27	78.4	184.5	1
8	26.5	15.5	14.1	18.9	23	62.0	175.0	1
9	28.0	22.5	15.6	21.1	21	81.6	184.0	1

예) 주어진 자료를 성별에 따라 두 개의 데이터셋으로 나누어 보자. 이 중 여성의 데이터셋을 이용하여 bii.di 변수에 대해 히스토그램과 정규분포 분위수 대조도를 그려보자.

: 먼저 주어진 자료를 성별에 따라 나누어서 각각 bodydims_m 과 bodydims_f 로 저장해보자. 주어진 데이터 프레임의 성별 변수값 중 남성은 1, 여성은 0으로 되어있다. 특정 조건을 만족하는 부분집합을 선택하기 위해서는 논리 연산자를 다음과 같이 이용할 수 있다. 주어진 스크립트를 실행하면 두 개의 새로운 데이터셋이 생성된 것을 확인할 수 있다.

In [17]:

```
bodydims_m = bodydims[bodydims['sex'] == 1]
bodydims_f = bodydims[bodydims['sex'] == 0]
```

정규분포 분위수 대조도를 그리기 위해서는 statsmodels.graphics.gofplots모듈의 ProbPlot 함수를 이용한다. 아래는 주어진 변수의 히스토그램과 정규분포 분위수를 나타낸 것이다.

In [18]:

```
import matplotlib.pyplot as plt
from statsmodels.graphics.gofplots import ProbPlot

x=bodydims_f['bii.di']

plt.hist(x, bins = 8, color = "CO", histtype='bar')
plt.title("histogram of bodydims_f['bii.di']")
plt.ylabel("Frequency")
plt.xlabel("bodydims_f['bii.di']")              # 히스토그램을 그리는 부분

QQ = ProbPlot(x)
plot = QQ.qqplot(line = 's', color='CO', lw=1)
plt.title("Normal Q-Q Plot")                    # 정규분포 분위수를 그리는 부분

plt.show()
```