

이공학도를 위한 확률 및 통계학

- 6장 연속형 확률분포 -

이 성 현(seonghyeon@pel.sejong.ac.kr)

세종대학교 프로토콜공학연구실

목 차

- 연속형 균일분포
- 정규분포와 표준정규분포
- 이항분포의 정규근사
- 감마분포와 지수분포
- 카이제곱분포와 베타분포
- 로그정규분포와 와이블분포

목 차

- 연속형 균일분포
- 정규분포와 표준정규분포
- 이항분포의 정규근사
- 감마분포와 지수분포
- 카이제곱분포와 베타분포
- 로그정규분포와 와이블분포

연속형 균일분포

• 연속형 균일분포(Continuous Uniform Distribution) (1/3)

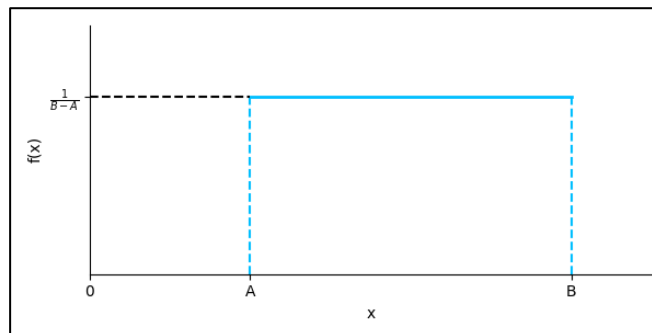
• 정의

- 특정한 구간 내 모든 실수에 대해 균일한 확률을 가지는 분포

$$f(x; A, B) = \begin{cases} \frac{1}{B-A}, & A \leq X \leq B \\ 0, & \text{그 외의 경우} \end{cases}$$

• 특징 (1/2)

- 직사각형 분포(Rectangular Distribution)이라고도 하며, 구간은 닫힌 구간 $[A, B]$ 이든 열린 구간 (A, B) 이든 상관 없음
- $[A, B]$ 내 특정 구간에 속할 확률은 구간의 길이만 같다면 모두 동일함



<그림 1> 균일분포의 확률밀도함수도

연속형 균일분포

• 연속형 균일분포(Continuous Uniform Distribution) (2/3)

• 특징 (2/2)

• 평균 $\mu = \frac{A+B}{2}$, 분산 $\sigma^2 = \frac{(B-A)^2}{12}$ 을 가짐

• $f(x) = \frac{1}{B-A}$ 이므로 평균 $\mu = \int_A^B x f(x) dx = \frac{1}{B-A} \int_A^B x dx = \frac{1}{B-A} \left[\frac{x^2}{2} \right]_A^B$
 $= \frac{1}{B-A} \times \frac{B^2 - A^2}{2} = \frac{A+B}{2}$

• $f(x) = \frac{1}{B-A}, \mu = \frac{A+B}{2}$ 이므로 분산 $\sigma^2 = E[(X - \mu)^2]$
 $= \int_A^B (x - \mu)^2 f(x) dx = \frac{1}{B-A} \int_A^B (x - \mu)^2 dx,$

$h = \frac{B-A}{2}$ 로 하고 $y = x - \mu$ 로 치환적분하면

$$\sigma^2 = \frac{1}{B-A} \int_{-h}^h y^2 dy = \frac{1}{B-A} \left[\frac{y^3}{3} \right]_{-h}^h = \frac{1}{B-A} \times \frac{2h^3}{3} = \frac{(B-A)^2}{12}$$

연속형 균일분포

• 연속형 균일분포(Continuous Uniform Distribution) (3/3)

• 예제 6.1

어느 회사의 대형 회의실은 4 시간을 초과하여 사용할 수 없다. 그 회의실에서는 긴 회의와 짧은 회의가 자주 열리며, 회의시간 X 는 구간 $[0,4]$ 에서 정의되는 균일 분포로 가정할 수 있다고 한다.

(a) 밀도함수를 구하라.

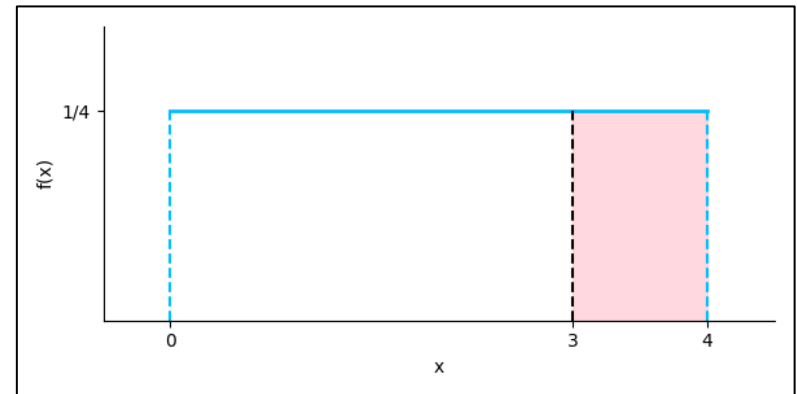
(b) 어떤 회의가 최소한 3 시간 이상 계속될 확률은 얼마인가?

• (a) 균일분포이므로 밀도함수는 다음과 같음

$$f(x) = \begin{cases} \frac{1}{4}, & 0 \leq x \leq 4 \\ 0, & \text{그 외의 경우} \end{cases}$$

$$\bullet (b) P[X \geq 3] = \int_3^4 \frac{1}{4} dx = \frac{1}{4}$$

∴ 회의가 최소 3 시간 이상 계속될 확률은 25%



<그림 2> 예제 6.1의 확률밀도함수도

목 차

- 연속형 균일분포
- 정규분포와 표준정규분포
- 이항분포의 정규근사
- 감마분포와 지수분포
- 카이제곱분포와 베타분포
- 로그정규분포와 와이블분포

정규분포와 표준정규분포

- 정규분포(Normal Distribution) (1/5)

- 정의

- 확률밀도함수가 다음을 따르는 연속형 확률분포

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

(단, μ 는 평균, σ 는 표준편차)

- 필요성

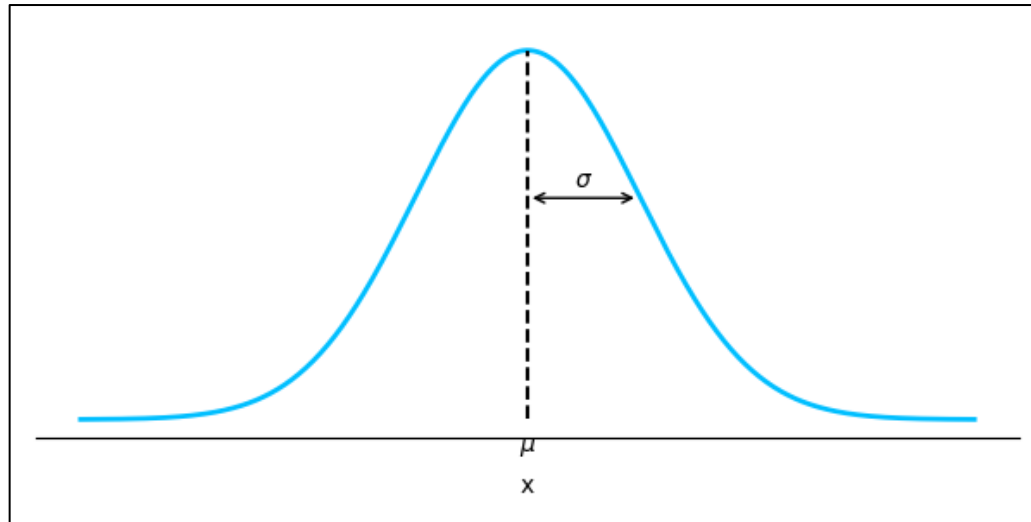
- 과학, 기업, 연구 등의 분야에서 발생하는 여러 현상들을 근사적으로 기술하는 데 사용됨
- 기상실험, 강우량 등의 물리적 실험에 적합하며, 과학적 측정오차는 정규분포와 거의 일치하는 경우가 많음

정규분포와 표준정규분포

- 정규분포(Normal Distribution) (2/5)

- 특징 (1/4)

- 가우스 분포(Gaussian Distribution)라고도 함
- 정규분포가 그리는 종 모양의 좌우대칭 곡선 그래프를 정규 곡선(Normal Curve)이라고 함
- 종 모양 확률분포를 가지는 확률변수를 정규확률변수(Normal Random Variable)이라고 함



<그림 3> 정규분포의 확률밀도함수도

정규분포와 표준정규분포

- 정규분포(Normal Distribution) (3/5)

- 특징 (2/4)

- 평균 μ , 분산 σ^2 , 표준편차 σ 를 가짐

- $z = \frac{x-\mu}{\sigma}$ 로 놓고 치환적분하면 $dx = \sigma dz$ 이고,

$$E(X - \mu) = \int_{-\infty}^{\infty} \frac{x-\mu}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} dx = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} z e^{-\frac{1}{2}z^2} dz = 0 = E(X) - \mu$$

$$\therefore E(X) = \mu$$

- $z = \frac{x-\mu}{\sigma}$ 로 놓고 치환적분하면 $dx = \sigma dz$ 이고,

$$E[(X - \mu)^2] = \frac{1}{\sqrt{2\pi}\sigma} \int_{-\infty}^{\infty} (x - \mu)^2 e^{-\frac{1}{2}\left[\frac{x-\mu}{\sigma}\right]^2} dx = \frac{\sigma^2}{\sqrt{2\pi}} \int_{-\infty}^{\infty} z^2 e^{-\frac{z^2}{2}} dz$$

- $u = z, dv = z e^{-\frac{z^2}{2}} dz$ 로 놓고 부분적분을 하면 $du = dz, v = -e^{-\frac{z^2}{2}}$ 이고,

$$E[(X - \mu)^2] = \frac{\sigma^2}{\sqrt{2\pi}} \left(-ze^{-z^2/2} \Big|_{-\infty}^{\infty} + \int_{-\infty}^{\infty} e^{-\frac{z^2}{2}} dz \right) = \frac{\sigma^2}{\sqrt{2\pi}} (0 + \sqrt{2\pi}) = \sigma^2$$

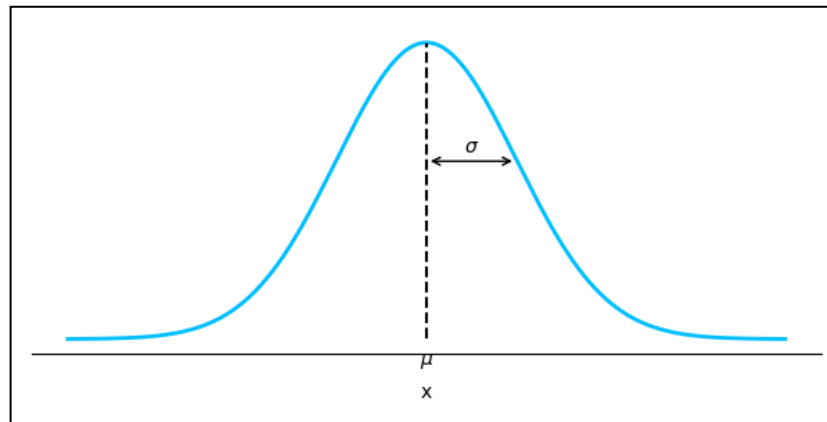
$$\therefore E[(X - \mu)^2] = \sigma^2$$

정규분포와 표준정규분포

- 정규분포(Normal Distribution) (4/5)

- 특징 (3/4)

- $x = \mu$ 에서 곡선이 최대값이 되며, 최빈값(Mode)를 가짐
- 곡선은 평균 μ 를 지나는 수직축에 대하여 대칭임
- 곡선은 $x = \mu \pm \sigma$ 에서 변곡점을 가지며, $\mu - \sigma < X < \mu + \sigma$ 에서 아래로 오목하고 그 외 구간에서 위로 오목함
- 평균에서 멀어질수록 정규곡선은 수평축에 접근함
- 곡선과 수평축 사이의 총 면적은 1이 됨



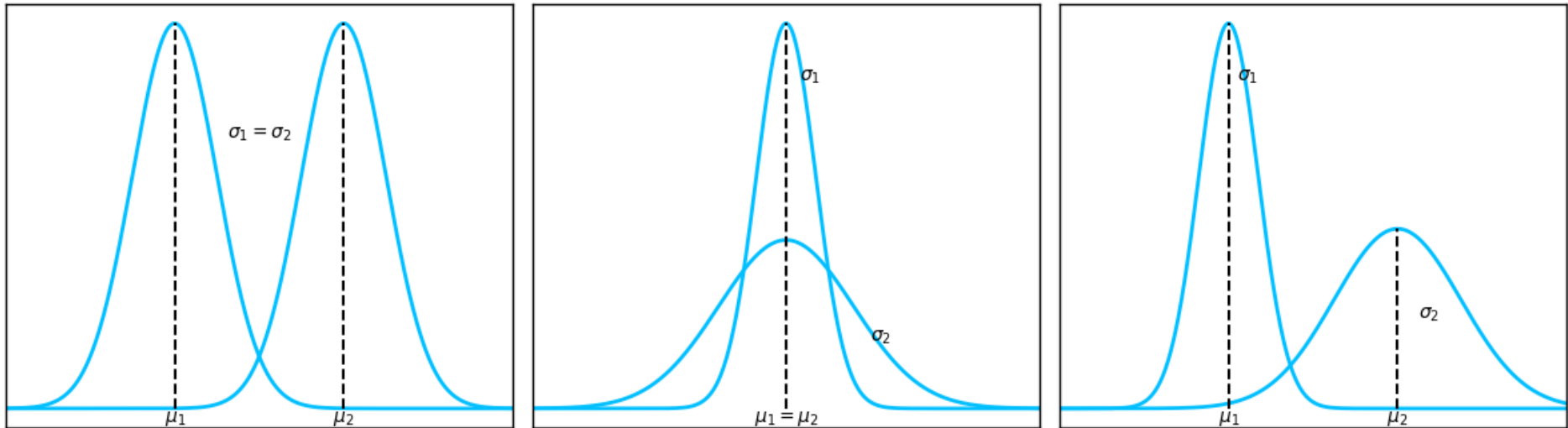
<그림 3> 정규분포의 확률밀도함수도

정규분포와 표준정규분포

- 정규분포(Normal Distribution) (5/5)

- 특징 (4/4)

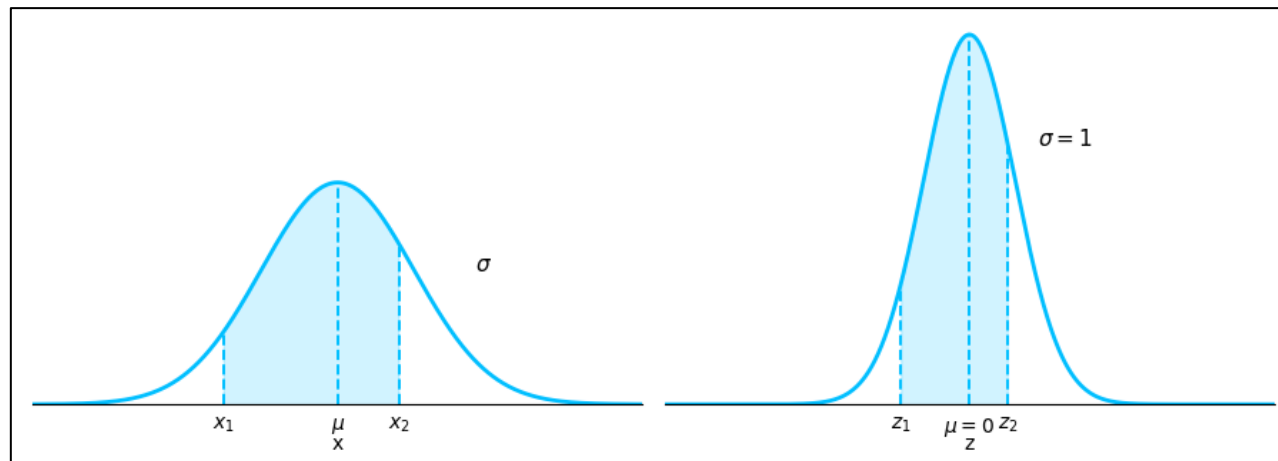
- 평균 μ 는 정규곡선의 중심위치를 결정하고, 표준편차 σ 는 정규곡선의 퍼짐 정도를 결정함



<그림 4> 평균과 표준편차에 따른 정규분포도 변화

정규분포와 표준정규분포

- 표준정규분포(Standard Normal Distribution) (1/9)
 - 정의
 - 평균이 0이고 분산이 1인 정규확률변수의 분포
 - 특징 (1/2)
 - 평균이 μ 이고 표준편차가 σ 인 확률변수 X 를 평균이 0, 분산이 1인 $Z = \frac{X-\mu}{\sigma}$ 로 변환하여 계산할 수 있음



<그림 5> 기존의 정규분포와 변환된 표준정규분포

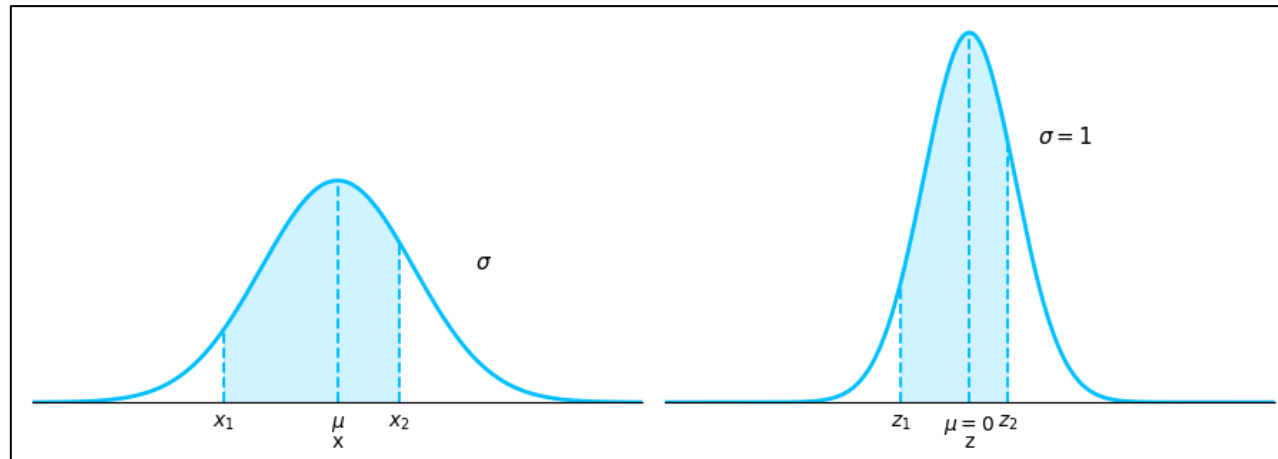
정규분포와 표준정규분포

- 표준정규분포(Standard Normal Distribution) (2/9)

- 특징 (2/2)

- 정규확률변수 X 가 x_1 에서 x_2 사이의 값을 가진다면, 정규확률변수 Z 는 z_1 에서 z_2 사이의 값을 가짐

- $$P(x_1 < X < x_2) = \frac{1}{\sqrt{2\pi}\sigma} \int_{x_1}^{x_2} e^{-\frac{1}{2\sigma^2}(x-\mu)^2} dx = \frac{1}{\sqrt{2\pi}} \int_{z_1}^{z_2} e^{-\frac{1}{2}z^2} dz$$
$$= \int_{z_1}^{z_2} n(z; 0, 1) dz = P(z_1 < Z < z_2)$$



<그림 5> 기존의 정규분포와 변환된 표준정규분포

정규분포와 표준정규분포

• 표준정규분포(Standard Normal Distribution) (3/9)

• 표준정규분포표

- 표준정규분포표를 이용하여 구간에 대한 확률값을 계산할 수 있음
- 주어진 확률값 사이에 해당하는 확률값을 계산해야 하는 경우, 표준정규분포표를 통해 두 개의 z 값의 중간값을 사용함
 - e.g., 확률 0.7975에 대응하는 z 값은 0.7967과 0.7995 중 0.7975에 더 가까운 $z = 0.83$ 을 선택함
 - e.g., 확률 0.7981은 0.7967과 0.7995의 중앙값이므로 $z = 0.835$ 를 선택함

z	.00	.01	.02	.03	.04
0.0	0.5000	0.5040	0.5080	0.5120	0.5160
0.1	0.5398	0.5438	0.5478	0.5517	0.5557
0.2	0.5793	0.5832	0.5871	0.5910	0.5948
0.3	0.6179	0.6217	0.6255	0.6293	0.6331
0.4	0.6554	0.6591	0.6628	0.6664	0.6700
0.5	0.6915	0.6950	0.6985	0.7019	0.7054
0.6	0.7257	0.7291	0.7324	0.7357	0.7389
0.7	0.7580	0.7611	0.7642	0.7673	0.7704
0.8	0.7881	0.7910	0.7939	0.7967	0.7995
0.9	0.8159	0.8186	0.8212	0.8238	0.8264

<그림 6> 확률 0.7975에 가까운 z 값을 찾기 위한 표준정규분포표

정규분포와 표준정규분포

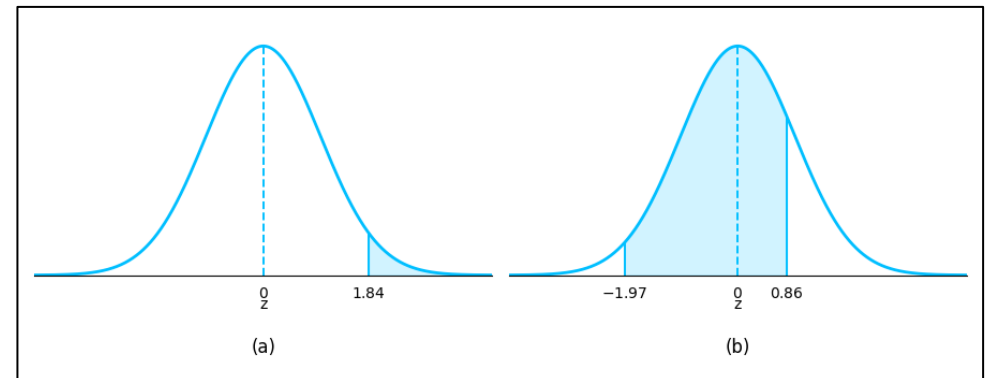
- 표준정규분포(Standard Normal Distribution) (4/9)
- 예제 6.2

표준정규분포가 주어졌을 때, 다음의 면적을 구하라.

(a) $z = 1.84$ 의 오른쪽 면적

(b) $z = -1.97$ 과 $z = 0.86$ 사이의 면적

- (a) $z = 1.84$ 의 오른쪽 면적은 1에서 $z = 1.84$ 의 왼쪽 면적을 뺀 값임
 $\therefore P(Z > 1.84) = 1 - P(Z < 1.84) = 1 - 0.9671 = 0.0329$
- (b) $z = -1.97$ 과 $z = 0.86$ 사이의 면적은 $z = 0.86$ 의 왼쪽 면적에서 $z = -1.97$ 의 왼쪽 면적을 뺀 값임
 $\therefore P(-1.97 < Z < 0.86) = P(Z < 0.86) - P(Z < -1.97)$
 $= 0.8051 - 0.0244 = 0.7807$



<그림 7> 예제 6.2의 표준정규분포도

정규분포와 표준정규분포

• 표준정규분포(Standard Normal Distribution) (5/9)

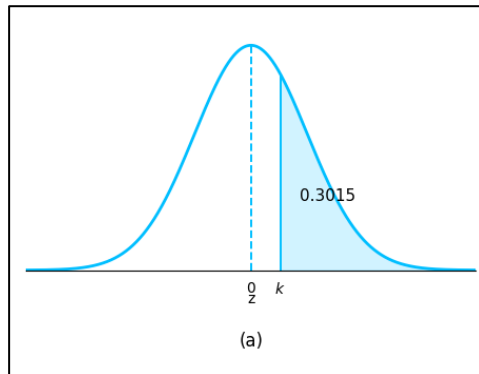
• 예제 6.3 (1/2)

표준정규분포가 주어졌을 때, 다음 각 경우에 대하여 k 값을 구하라.

(a) $P(Z > k) = 0.3015$

(b) $P(k < Z < -0.18) = 0.4197$

- (a) k 값 오른쪽 면적이 0.3015이므로 왼쪽 면적은 0.6985임
 $\therefore k = 0.52 (\because P(Z < k) = 0.6985)$



<그림 8> 예제 6.3 (a)의 표준정규분포도

z	.00	.01	.02	.03
0.0	0.5000	0.5040	0.5080	0.5120
0.1	0.5398	0.5438	0.5478	0.5517
0.2	0.5793	0.5832	0.5871	0.5910
0.3	0.6179	0.6217	0.6255	0.6293
0.4	0.6554	0.6591	0.6628	0.6664
0.5	0.6915	0.6950	0.6985	0.7019
0.6	0.7257	0.7291	0.7324	0.7357
0.7	0.7580	0.7611	0.7642	0.7673
0.8	0.7881	0.7910	0.7939	0.7967
0.9	0.8159	0.8186	0.8212	0.8238

<그림 9> 확률 0.6985에 가까운 z 값을 찾기 위한 표준정규분포표

정규분포와 표준정규분포

• 표준정규분포(Standard Normal Distribution) (6/9)

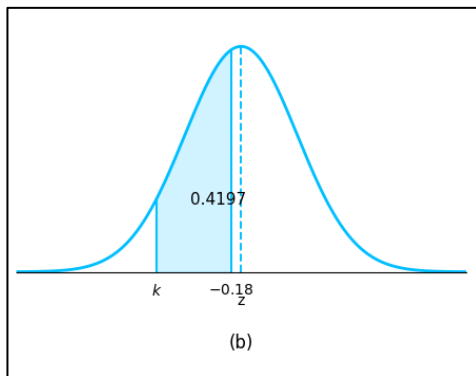
• 예제 6.3 (2/2)

표준정규분포가 주어졌을 때, 다음 각 경우에 대하여 k 값을 구하라.

(a) $P(Z > k) = 0.3015$

(b) $P(k < Z < -0.18) = 0.4197$

- (b) -0.18 의 왼쪽 면적은 0.4286 이고 k 와 -0.18 사이의 면적이 0.4197 이므로 k 의 왼쪽 면적은 0.0089 임
 $\therefore k = -2.37$ ($\because P(Z < k) = 0.0089$)



<그림 10> 예제 6.3 (b)의 표준정규분포도

z	.00	.01	.02	.03	.04	.05	.06	.07
-3.4	0.0003	0.0003	0.0003	0.0003	0.0003	0.0003	0.0003	0.0003
-3.3	0.0005	0.0005	0.0005	0.0004	0.0004	0.0004	0.0004	0.0004
-3.2	0.0007	0.0007	0.0006	0.0006	0.0006	0.0006	0.0006	0.0005
-3.1	0.0010	0.0009	0.0009	0.0009	0.0008	0.0008	0.0008	0.0008
-3.0	0.0013	0.0013	0.0013	0.0012	0.0012	0.0011	0.0011	0.0011
-2.9	0.0019	0.0018	0.0018	0.0017	0.0016	0.0016	0.0015	0.0015
-2.8	0.0026	0.0025	0.0024	0.0023	0.0023	0.0022	0.0021	0.0021
-2.7	0.0035	0.0034	0.0033	0.0032	0.0031	0.0030	0.0029	0.0028
-2.6	0.0047	0.0045	0.0044	0.0043	0.0041	0.0040	0.0039	0.0038
-2.5	0.0062	0.0060	0.0059	0.0057	0.0055	0.0054	0.0052	0.0051
-2.4	0.0082	0.0080	0.0078	0.0075	0.0073	0.0071	0.0069	0.0068
-2.3	0.0107	0.0104	0.0102	0.0099	0.0096	0.0094	0.0091	0.0089
-2.2	0.0139	0.0136	0.0132	0.0129	0.0125	0.0122	0.0119	0.0116
-2.1	0.0179	0.0174	0.0170	0.0166	0.0162	0.0158	0.0154	0.0150
-2.0	0.0228	0.0222	0.0217	0.0212	0.0207	0.0202	0.0197	0.0192

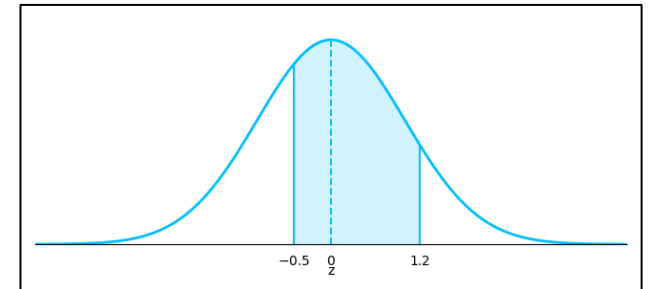
<그림 11> 확률 0.0089에 가까운 z 값을 찾기 위한 표준정규분포표

정규분포와 표준정규분포

- 표준정규분포(Standard Normal Distribution) (7/9)
- 예제 6.4

$\mu = 50$ 이고 $\sigma = 10$ 인 정규분포가 주어졌을 때, X 가 45와 62 사이의 값을 취할 확률을 구하라.

- $x_1 = 45$ 와 $x_2 = 62$ 에 대응하는 z 값은
 $z_1 = \frac{45-50}{10} = -0.5$, $z_2 = \frac{62-50}{10} = 1.2$ 이므로
 $P(45 < X < 62) = P(-0.5 < Z < 1.2)$ 가 됨
- 위 확률은 표준정규분포의 그래프 $z = 1.2$ 의 왼쪽 면적에서 $z = -0.5$ 의 왼쪽 면적을 뺀 값이 되므로 결과는 다음과 같음
 $P(45 < X < 62) = P(-0.5 < Z < 1.2) = P(Z < 1.2) - P(Z < -0.5)$
 $= 0.8849 - 0.3085 = 0.5764$
 $\therefore X$ 가 45와 62 사이 값을 취할 확률은 57.64%



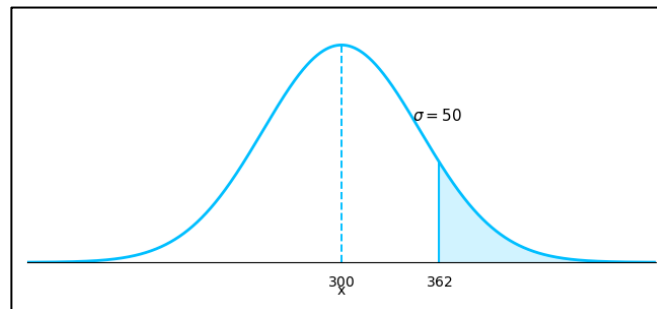
<그림 12> 예제 6.4의 표준정규분포도

정규분포와 표준정규분포

- 표준정규분포(Standard Normal Distribution) (8/9)
- 예제 6.5

$\mu = 300$ 이고 $\sigma = 50$ 인 정규분포가 주어졌을 때, X 가 362보다 큰 값을 취할 확률을 구하라.

- 해당 확률은 $x = 362$ 를 그에 대응하는 z 값으로 변환한 다음, z 값의 왼쪽 면적을 구하고 1에서 그 값을 빼서 얻을 수 있음
- $z = \frac{362-300}{50} = 1.24$ 이므로
 $P(X > 362) = P(Z > 1.24) = 1 - P(Z < 1.24) = 1 - 0.8925 = 0.1075$
 $\therefore X$ 가 362보다 큰 값을 취할 확률은 10.75%



<그림 13> 예제 6.5의 확률 밀도함수도

정규분포와 표준정규분포

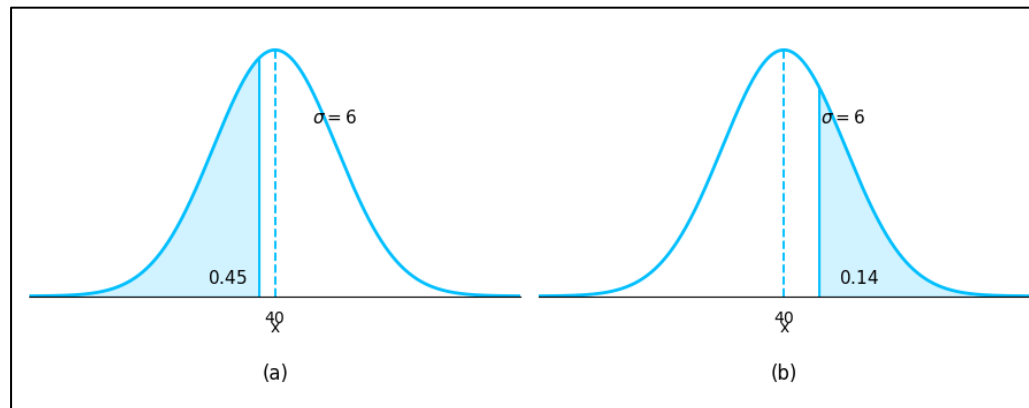
• 표준정규분포(Standard Normal Distribution) (9/9)

• 예제 6.6

$\mu = 40$ 이고 $\sigma = 6$ 인 정규분포가 주어졌을 때, 다음을 구하라.

- (a) 왼쪽 면적이 전체 면적의 45%가 되는 x
- (b) 오른쪽 면적이 전체 면적의 14%가 되는 x

- (a) 왼쪽 면적이 0.45가 되는 z 는 $P(Z < -0.13) = 0.450$ 이므로 $z = -0.13$
 $\therefore$ 왼쪽 면적이 전체 면적의 45%가 되는 $x = (6)(-0.13) + 40 = 39.22$
- (b) 오른쪽 면적이 0.14가 되는 z 는 $P(Z < 1.08) = 0.860$ 이므로 $z = 1.08$
 $\therefore$ 오른쪽 면적이 전체 면적의 14%가 되는 $x = (6)(1.08) + 40 = 46.48$



<그림 14> 예제 6.6의 확률밀도함수도

정규분포와 표준정규분포

• 정규분포의 적용 (1/8)

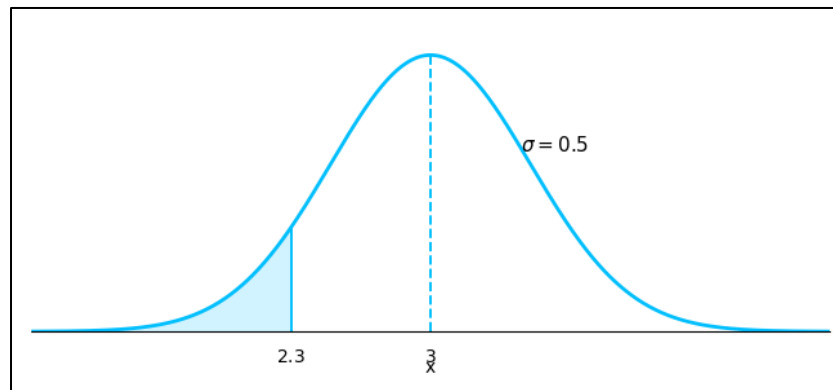
• 예제 6.7

어느 축전지는 평균수명이 3년이고 표준편차가 0.5년인 것으로 알려져 있다. 축전지의 수명이 정규분포를 따른다고 가정할 때, 임의로 주어진 전지의 수명이 2.3년 보다 짧을 확률을 구하라.

- $P(X < 2.3)$ 을 구해야하므로 정규곡선하에서 2.3의 왼쪽 면적을 구해야 하고, 이는 대응하는 z 값이 왼쪽 면적으로 구할 수 있음

$$z = \frac{2.3 - 3}{0.5} = -1.4, P(X < 2.3) = P(Z < -1.4) = 0.0808$$

∴ 전지의 수명이 2.3년보다 짧을 확률은 8.08%



<그림 15> 예제 6.7의 확률밀도함수도

정규분포와 표준정규분포

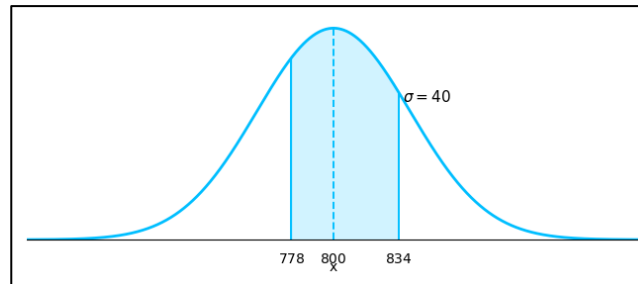
• 정규분포의 적용 (2/8)

• 예제 6.8

어느 전기회사에서는 평균수명이 800 시간이고 표준편차가 40 시간인 정규분포의 수명분포를 가지는 전구를 생산하고 있다. 임의로 선정된 전구의 수명이 778 시간과 834 시간 사이에 있을 확률을 구하라.

- 전구의 수명 $x_1 = 778, x_2 = 834$ 에 대응하는 z 값은 각각
$$x_1 = \frac{778-800}{40} = -0.55, x_2 = \frac{834-800}{40} = 0.85$$
이므로
$$P(778 < X < 834) = P(-0.55 < Z < 0.85)$$
$$= P(Z < 0.85) - P(Z < -0.55)$$
$$= 0.8023 - 0.2912 = 0.5111$$

∴ 전구의 수명이 778 시간과 834 시간 사이에 있을 확률은 51.11%



<그림 16> 예제 6.8의 확률 밀도함수도

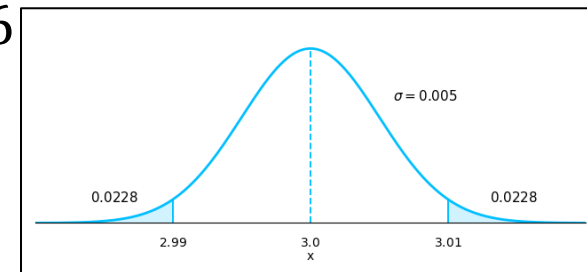
정규분포와 표준정규분포

• 정규분포의 적용 (3/8)

• 예제 6.9

어느 공정에서 볼베어링의 직경이 매우 중요한 품질특성이 된다. 구매자 측은 직경의 규격한계를 $3.0 \pm 0.01\text{cm}$ 로 정해 놓고 이 규격한계를 벗어나는 부품은 불합격처리한다. 볼베어링의 직경은 평균이 3.0, 표준편차 0.005인 정규분포를 따른다고 할 때, 생산된 제품 중 불합격으로 처리되는 것은 얼마나 되겠는가?

- 규격한계인 $x_1 = 2.99$, $x_2 = 3.01$ 에 대응되는 z 값은 각각
$$z_1 = \frac{2.99 - 3.0}{0.005} = -2.0, z_2 = \frac{3.01 - 3.0}{0.005} = 2.0$$
- 따라서 $P(2.99 < X < 3.01) = P(-2.0 < Z < 2.0)$
- $P(Z < -2.0) = 0.0228$ 이고 정규분포의 대칭성을 이용하면
$$P(Z < -2.0) + P(Z > 2.0) = 2 \times 0.0228 = 0.0456$$
- ∴ 생산된 볼베어링의 4.56%가 불합격품 처리됨



<그림 17> 예제 6.9의 확률밀도함수도

정규분포와 표준정규분포

• 정규분포의 적용 (4/8)

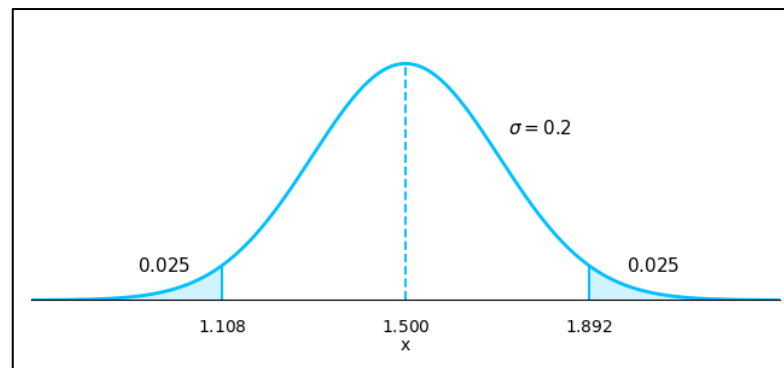
• 예제 6.10

어떤 치수가 규격한계인 $1.50 \pm d$ 내에 들어오지 않으면 모든 부품을 불합격시키는 평가기준이 사용된다고 한다. 측정값은 평균이 1.50이고 표준편차가 0.2인 정규분포를 따른다고 알려져 있을 때, 측정값의 95%가 규격한계 내에 들도록 d 값을 결정하라.

- $P(Z < -1.96) = 0.025$ 이므로 $P(-1.96 < Z < 1.96) = 0.95$

- $1.96 = \frac{(1.50+d)-1.50}{0.2}$ 이므로 $d = (0.2)(1.96) = 0.392$

∴ 규격한계를 1.50 ± 0.392 로 설정하면 부품의 95%가 규격한계 내에 존재함



<그림 18> 예제 6.10의 확률밀도함수도

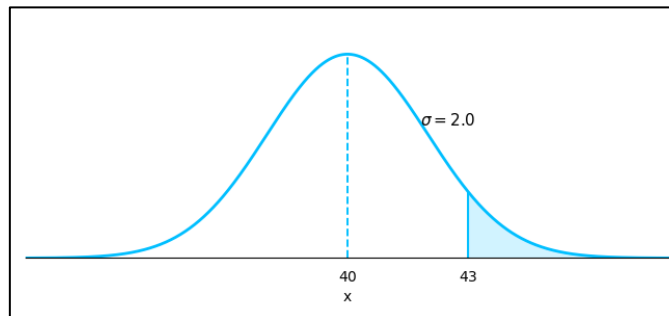
정규분포와 표준정규분포

• 정규분포의 적용 (5/8)

• 예제 6.11

평균저항이 $40\ \Omega$ 이고 표준편차가 $2\ \Omega$ 인 저항기를 만드는 기계가 있다. 저항이 정규분포를 따른다고 가정할 때, $43\ \Omega$ 이 저항을 가지게 되는 저항기는 몇 퍼센트나 되겠는가?

- 저항기의 저항 $x = 43$ 을 그에 대응하는 z 값으로 변환하면
$$z = \frac{43-40}{2} = 1.5$$
- 저항이 $43\ \Omega$ 이상인 경우이므로
$$P(X > 43) = P(Z > 1.5) = 1 - P(Z < 1.5) = 0.9332 = 0.0668$$
$$\therefore 43\ \Omega \text{ 이상의 저항을 가지게 되는 저항기는 } 6.68\%$$



<그림 19> 예제 6.11의 확률밀도함수도

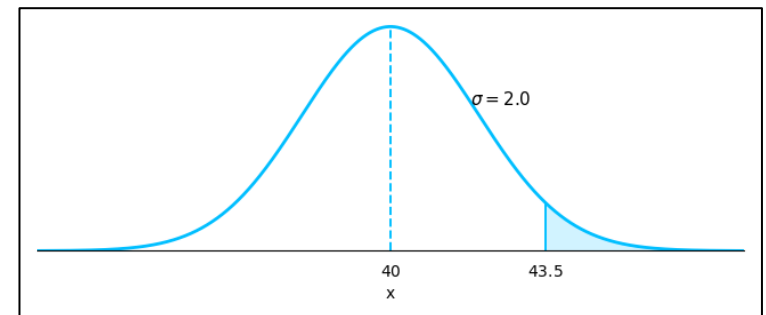
정규분포와 표준정규분포

• 정규분포의 적용 (6/8)

• 예제 6.12

예제 6.11에서의 저항의 측정값을 소수 첫째 자리에서 반올림할 때 43 Ω 이 넘는 저항기의 비율을 구하라.

- 42.5 Ω 에서 43.5 Ω 사이의 저항은 모두 소수 첫째 자리에서 반올림할 때 43 Ω 가 되므로 43 Ω 을 넘기 위해서는 43.5 Ω 보다 커야 함
- 저항기의 저항 $x = 43.5$ 를 그에 대응하는 z 값으로 변환하면
$$Z = \frac{43.5 - 40}{2} = 1.75$$
- $P(X > 43.5) = P(Z > 1.75) = 1 - P(Z < 1.75) = 1 - 0.9599 = 0.0401$
- $\therefore$ 소수 첫째 자리에서 반올림할 때 43 Ω 이 넘는 저항기는 4.01%
- 추가로 이전 예제 결과와의 차이
6.68% - 4.01% = 2.67%는
43 Ω 보다 크고 43.5 Ω 보다 작은
저항을 가지는 저항기의 비율이 됨



<그림 20> 예제 6.12의 확률밀도함수도

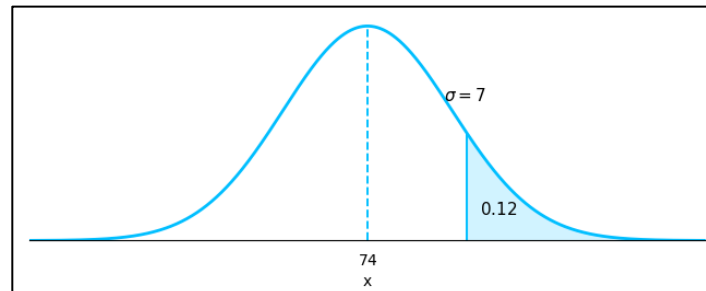
정규분포와 표준정규분포

• 정규분포의 적용 (7/8)

• 예제 6.13

어느 시험 성적이 평균이 74점이고 표준편차가 7인 정규분포를 따른다고 한다. 12%의 학생에게 A 학점이 주어졌다면, A 학점을 받은 학생 중 가장 낮은 점수와 B 학점을 받은 학생 중 가장 높은 점수는 각각 얼마나 되겠는가?

- A 학점을 받은 학생의 비율은 성적 평균 $\mu = 74$, 표준편차 $\sigma = 7$ 일 때 $P(X > k) = 0.12$ 인 경우임
- 대응하는 z 값을 구하면 $P(Z > 1.18) = 1 - P(Z < 1.18) = 1 - 0.88 = 0.12$ 를 통해 $z = 1.18$ 임을 알 수 있음
- $x = \sigma z + \mu$ 에 대입하여 계산하면 $x = (7)(1.18) + 74 = 82.26$
- ∴ A 학점 중 가장 낮은 점수는 83점, B 학점 중 가장 높은 점수는 82점



<그림 21> 예제 6.13의 확률밀도함수도

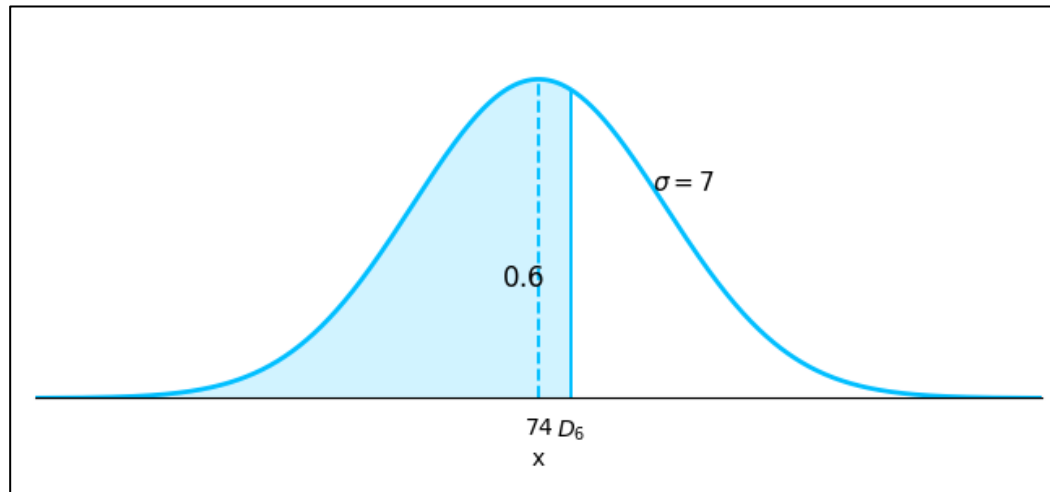
정규분포와 표준정규분포

- 정규분포의 적용 (8/8)

- 예제 6.14

예제 6.13에서 제 6 십분위수를 구하라.

- 제 6 십분위수를 D_6 라 하면, 구해야 하는 것은 $P(X < k) = 0.6$ 임
- $P(Z < 0.25) \approx 0.6$ 이므로 $x = \sigma z + \mu$ 에 대입하면
 $x = (7)(0.25) + 74 = 75.75$
- $\therefore D_6 = 75.75$ 이며, 이는 성적의 60%가 75점 이하임을 의미함



<그림 22> 예제 6.14의 확률밀도함수도

목 차

- 연속형 균일분포
- 정규분포와 표준정규분포
- 이항분포의 정규근사
- 감마분포와 지수분포
- 카이제곱분포와 베타분포
- 로그정규분포와 와이블분포

이항분포의 정규근사

- 이항분포(Binomial Distribution)

베르누이 과정(Bernoulli Process)
성공확률이 일정한 독립적인 이분법적
시행이 반복되는 이산 확률 과정

- 확률이 p 인 n 번의 베르누이 시행(Bernoulli Trial)에서의 성공횟수 X 에 대한 분포
- $b(x; n, p)$ 로 표기하고 평균 $\mu = np$, 분산 $\sigma^2 = npq$ 를 가짐 (단, $q = 1 - p$)

- 개요

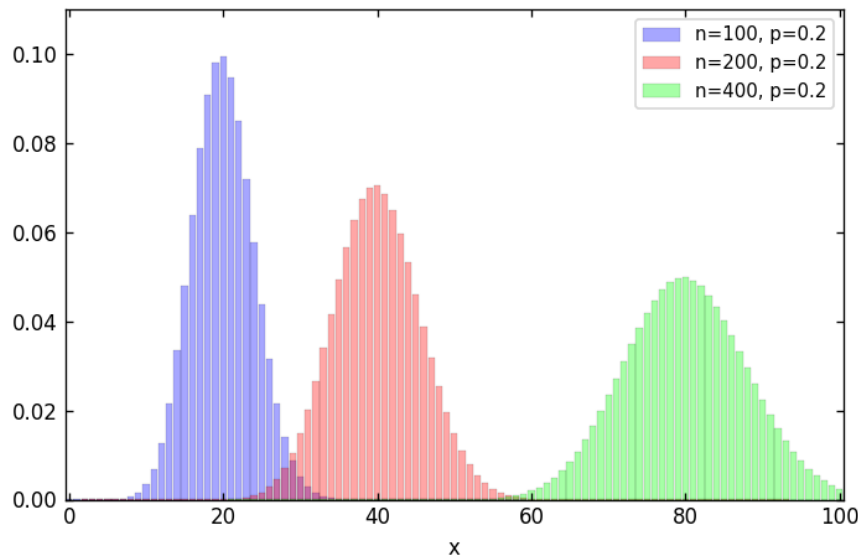
포아송 분포(Poisson Process)
일정한 평균 발생률을 가지는 사건이 어떤 고정된 구간에서 몇 번 발생하는지를 나타내는 이산형 확률분포

- 이항분포에서 n 이 매우 크고 p 가 0이나 1에 매우 가까운 경우, 포아송 분포를 통해 이항확률의 근사값을 계산함
- 마찬가지로, 이산형 분포가 종 모양처럼 대칭을 이루는 경우, 정규분포를 통해 근사값을 계산할 수 있음
 - X 가 $\mu = np$ 이고 $\sigma^2 = npq$ 인 이항확률변수이면, $n \rightarrow \infty$ 일 때 $Z = \frac{X - np}{\sqrt{npq}}$ 의 극한분포는 표준정규분포 $n(z; 0, 1)$ 을 따름

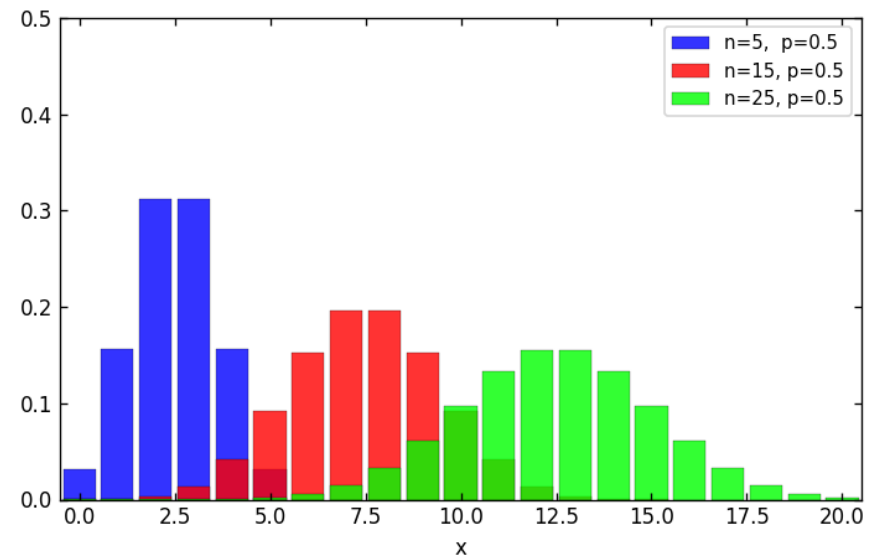
이항분포의 정규근사

- 근사 조건

- 이항분포의 n 이 크고 p 가 0이나 1에 가깝지 않거나, n 이 작더라도 p 가 $\frac{1}{2}$ 에 가까운 경우, 정규분포에 근사가 가능함



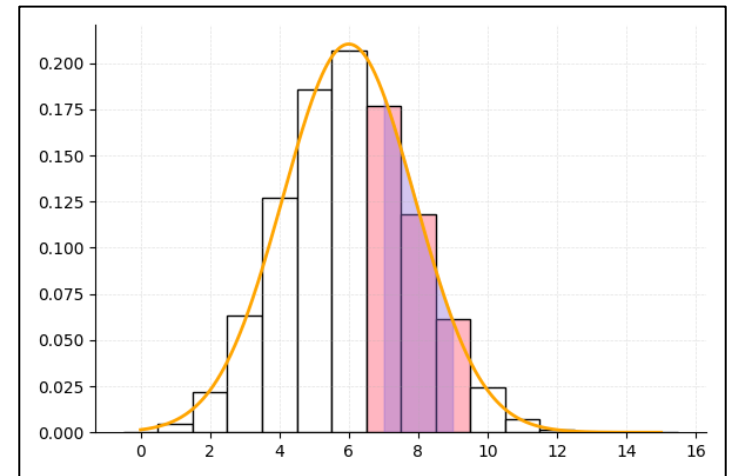
<그림 23> n 이 크고 p 가 0이나 1에 가깝지 않은 경우의 이항분포



<그림 24> p 가 $\frac{1}{2}$ 에 가까운 경우의 이항분포

이항분포의 정규근사

- 연속성 수정(Continuity Correction)
 - 이항분포를 정규분포로 근사시키는 경우, 오차 보정을 위해 x 대신 $x \pm 0.5$ 로 계산하는 것
 - x 이하일 확률과 x 초과일 확률은 $x + 0.5$ 로 계산하고,
 x 이상일 확률과 x 미만일 확률은 $x - 0.5$ 로 계산함
 - 모수 n 과 p 를 갖는 이항확률변수 X 는 평균 $\mu = np$ 이고 분산 $\sigma^2 = npq = np(1 - p)$ 인 정규분포를 근사적으로 따르게 됨
 - $P(X \leq x) = \sum_{k=0}^x b(k; n, p)$
 $\approx P\left(Z \leq \frac{x+0.5-np}{\sqrt{npq}}\right)$ 로 계산할 수
있으며, 해당 정규근사는 np 와 $n(1 - p)$ 가 5 이상일 때 더
적합하게 됨



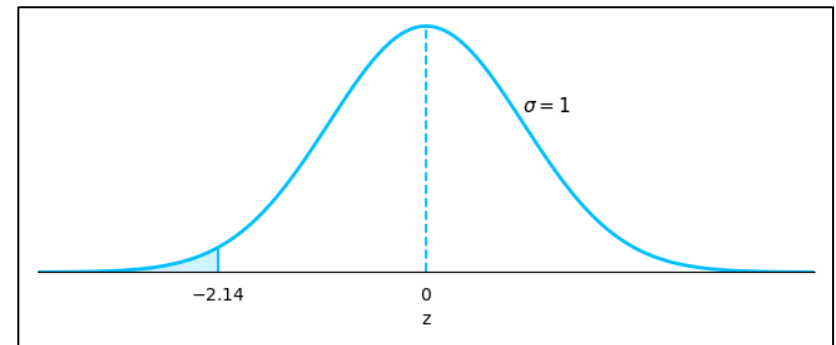
<그림 25> 이항분포 $b(x; 15, 0.4)$ 에 대한 $P(7 < X < 9)$ 의 정규근사

이항분포의 정규근사

• 예제 6.15

빈혈환자가 회복될 확률을 0.4라고 한다. 100명의 빈혈환자 중에서 회복되는 환자의 수가 30보다 적을 확률은 얼마인가?

- 이항확률변수 X 가 회복될 환자의 수일 때, $n = 100$ 이므로
 $\mu = np = (100)(0.4) = 40$, $\sigma = \sqrt{npq} = \sqrt{(100)(0.4)(0.6)} = 4.899$ 인
정규분포에 근사시켜도 정확한 값을 얻을 수 있음
- 연속성 수정으로 $x = 29.5$ 왼쪽 면적을 구하면 됨
- 대응하는 z 값은 $z = \frac{29.5 - 40}{4.899} = -2.14$ 이므로
 $P(X < 30) \approx P(Z < -2.14) = 0.0162$
 $\therefore$ 100명의 빈혈환자 중 회복되는 환자의
수가 30보다 적을 확률은 1.62%



<그림 26> 예제 6.15의 표준정규분포도

이항분포의 정규근사

• 예제 6.16

4개의 보기 중 하나이 정답이 있는 4지선다형 문제 200개가 있다고 하자. 그 시험 문제에 관해서 아무런 지식을 가지고 있지 않은 학생이 200 문제 중 80 문제의 답을 순전히 추측으로 골랐을 때 그 중 정답이 25개에서 30개까지일 확률은 얼마인가?

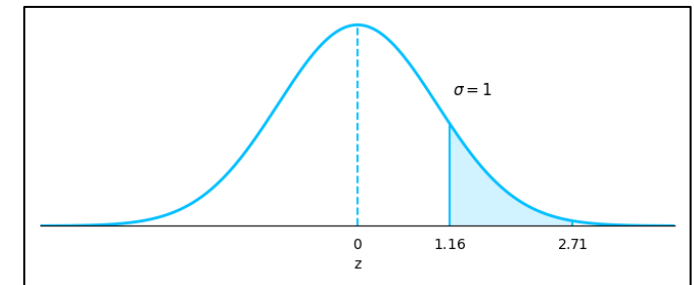
- 확률변수 X 를 학생이 맞춘 정답의 수라 하면,

$$P(25 \leq X < 30) = \sum_{x=25}^{30} b(x; 80, \frac{1}{4}) \text{임}$$

- $\mu = np = (80)(1/4) = 20$, $\sigma = \sqrt{npq} = \sqrt{(80)(1/4)(3/4)} = 3.873$ 인 정규곡선에 근사시키면 $x_1 = 24.5$ 와 $x_2 = 30.5$ 사이의 면적을 구하면 되며, 이에 대응하는 $z_1 = \frac{24.5-20}{3.873} = 1.16$, $z_2 = \frac{30.5-20}{3.873} = 2.71$

$$\begin{aligned} P(25 \leq X \leq 30) &= \sum_{x=25}^{30} b(x; 80, \frac{1}{4}) \\ &\approx P(1.16 < Z < 2.71) \\ &= P(Z < 2.71) - P(Z < 1.16) \\ &= 0.9966 - 0.8770 = 0.1196 \end{aligned}$$

$\therefore$ 4지선다 80 문제를 추측만으로 풀었을 때 정답이 25개에서 30개 사이일 확률은 11.96%



<그림 27> 예제 6.16의 확률밀도함수도

목 차

- 연속형 균일분포
- 정규분포와 표준정규분포
- 이항분포의 정규근사
- 감마분포와 지수분포
- 카이제곱분포와 베타분포
- 로그정규분포와 와이블분포

감마분포와 지수분포

- 감마함수(Gamma Function) (1/4)

- 정의

- 팩토리얼(Factorial)을 실수 영역까지 확장한 함수

- $\alpha > 0$ 인 α 에 대해 $\Gamma(\alpha) = \int_0^{\infty} x^{\alpha-1} e^{-x} dx$

- 특징

- 양의 정수 n 에 대해 $\Gamma(n) = (n-1)(n-2) \cdots (1)\Gamma(1)$

- $\Gamma(1) = 1$

- 양의 정수 n 에 대해 $\Gamma(n) = (n-1)!$

- $\Gamma\left(\frac{1}{2}\right) = \sqrt{\pi}$

감마분포와 지수분포

• 감마함수(Gamma Function) (2/4)

• 증명 (1/3)

1. $\Gamma(n) = (n-1)(n-2)\cdots(1)\Gamma(1)$ (단, n 은 양의 정수)

- $u = x^{\alpha-1}, dv = e^{-x} dx$ 로 놓고 부분적분

- $\Gamma(\alpha) = -e^{-x}x^{\alpha-1}|_0^\infty + \int_0^\infty e^{-x}(\alpha-1)x^{\alpha-2} dx = 0 + (\alpha-1) \int_0^\infty x^{\alpha-2}e^{-x} dx$

- $\alpha > 1$ 인 경우, $\Gamma(\alpha) = (\alpha-1)\Gamma(\alpha-1)$

- 이를 $\Gamma(\alpha) = (\alpha-1)\Gamma(\alpha-1) = (\alpha-1)(\alpha-2)\Gamma(\alpha-2)\cdots$ 과 같이 반복적용하면 $\Gamma(\alpha) = (\alpha-1)(\alpha-2)\cdots(1)\Gamma(1)$

- $\alpha = n$ 이고 n 이 양의 정수이면 $\Gamma(n) = (n-1)(n-2)\cdots(1)\Gamma(1)$

2. $\Gamma(1) = 1$

- $\Gamma(1) = \int_0^\infty e^{-x} dx = -e^{-x}|_0^\infty = 1$

3. $\Gamma(n) = (n-1)!$ (단, n 은 양의 정수)

- $\Gamma(n) = (n-1)(n-2)\cdots(1)\Gamma(1)$ 이고 $\Gamma(1) = 1$ 이므로
 $\Gamma(n) = (n-1)!$

감마분포와 지수분포

• 감마함수(Gamma Function) (3/4)

• 증명 (2/3)

극좌표(Polar Coordinate)

극점(Pole)과의 거리(r)와 극축(Polar Axis)에 대한 각도(θ)로 나타낸 좌표

$$4. \Gamma\left(\frac{1}{2}\right) = \sqrt{\pi} \quad (1/2)$$

$$\bullet \Gamma\left(\frac{1}{2}\right) = \int_0^{\infty} t^{-\frac{1}{2}} e^{-t} dt = \int_0^{\infty} \frac{e^{-t}}{\sqrt{t}} dt$$

$$\bullet t = y^2 \text{으로 치환하면(단, } y > 0) dt = 2y dy, \sqrt{t} = y \text{이므로}$$

$$\Gamma\left(\frac{1}{2}\right) = \int_0^{\infty} \frac{e^{-y^2}}{y} 2y dy = \int_0^{\infty} 2e^{-y^2} dy = 2 \int_0^{\infty} e^{-y^2} dy$$

• 양변 제곱

$$\bullet \Gamma\left(\frac{1}{2}\right)^2 = 4 \left(\int_0^{\infty} e^{-x^2} dx \right) \left(\int_0^{\infty} e^{-y^2} dy \right) = 4 \int_0^{\infty} \int_0^{\infty} e^{-(x^2+y^2)} dx dy$$

• $x^2 + y^2 = r^2, dx dy = r dr d\theta$ 로 하여 극좌표 변환

$$\bullet x \geq 0, y \geq 0 \text{으로 1사분면이므로 } \Gamma\left(\frac{1}{2}\right)^2 = 4 \int_0^{\frac{\pi}{2}} \int_0^{\infty} e^{-r^2} r dr d\theta$$

$$\bullet \int_0^{\frac{\pi}{2}} d\theta = \frac{\pi}{2} \text{이므로 } \Gamma\left(\frac{1}{2}\right)^2 = 4 \times \frac{\pi}{2} \int_0^{\infty} e^{-r^2} r dr = 2\pi \int_0^{\infty} e^{-r^2} r dr$$

감마분포와 지수분포

- 감마함수(Gamma Function) (4/4)

- 증명 (3/3)

4. $\Gamma\left(\frac{1}{2}\right) = \sqrt{\pi}$ (2/2)

- $u = r^2, du = 2r dr$ 로 하여 치환적분하면

$$\Gamma\left(\frac{1}{2}\right)^2 = 2\pi \int_0^\infty e^{-r^2} r dr = 2\pi \times \frac{1}{2} \int_0^\infty e^{-u} du = \pi$$

$$\therefore \Gamma\left(\frac{1}{2}\right)^2 = \pi \text{ 이므로 } \Gamma\left(\frac{1}{2}\right) = \sqrt{\pi}$$

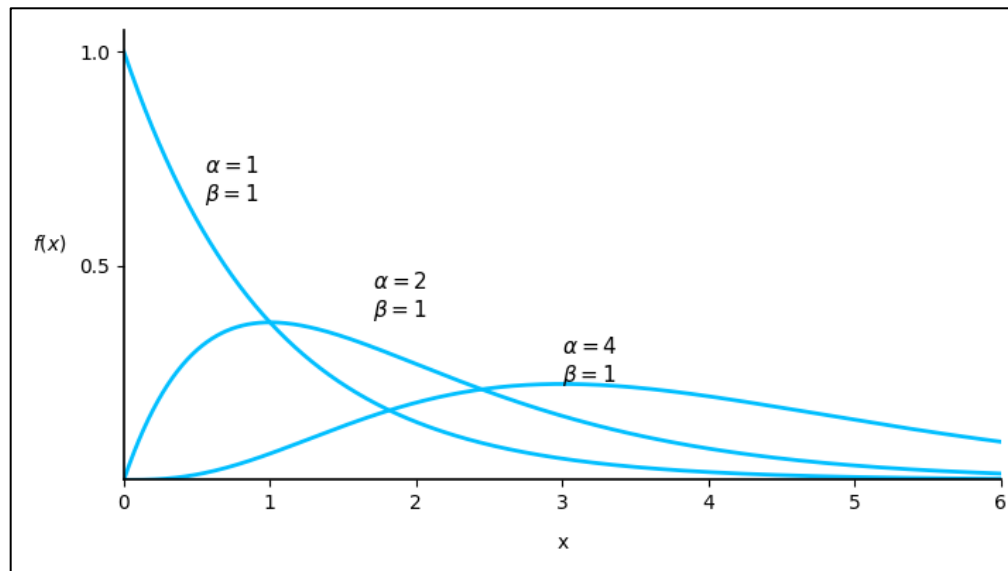
감마분포와 지수분포

- 감마분포(Gamma Distribution) (1/2)

- 정의

- 확률밀도함수가 다음을 따르는 연속형 확률분포

$$f(x; \alpha, \beta) = \begin{cases} \frac{1}{\beta^\alpha \Gamma(\alpha)} x^{\alpha-1} e^{-\frac{x}{\beta}}, & x > 0 \\ 0, & \text{그 외의 경우 (단, } \alpha > 0, \beta > 0) \end{cases}$$



<그림 28> α, β 에 따른 감마분포의 변화

감마분포와 지수분포

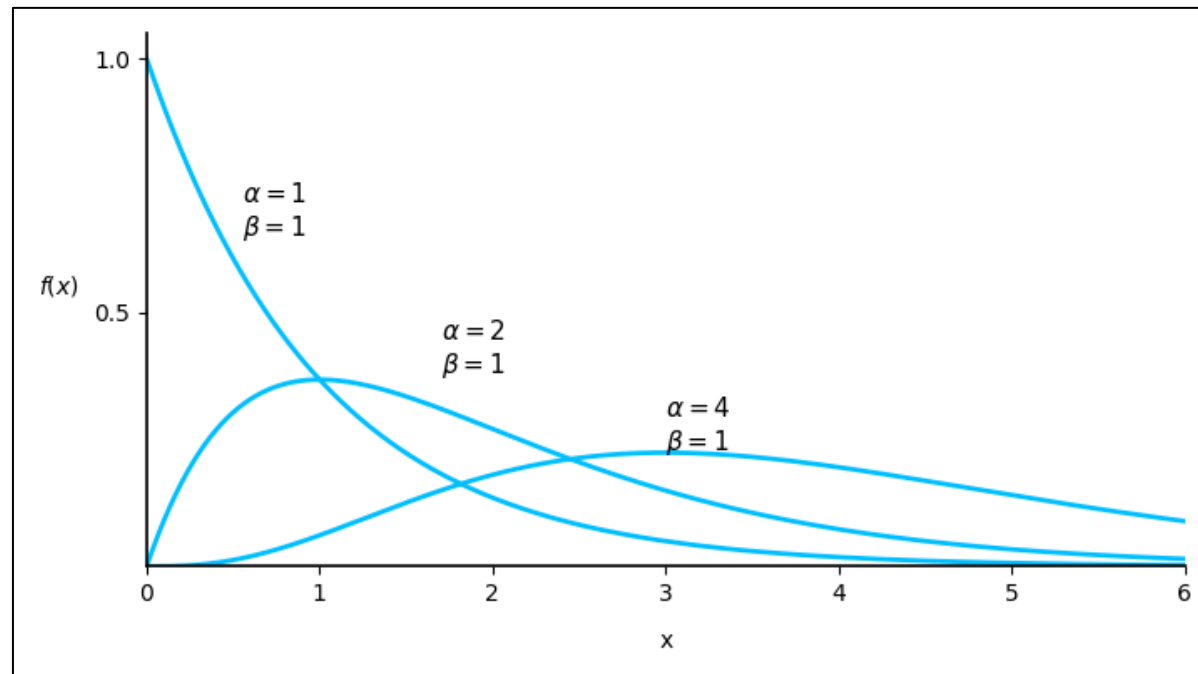
• 감마분포(Gamma Distribution) (2/2)

• 특징

포아송 과정(Poisson Process)

동시에 발생하지 않는 독립된 사건이 일정한 발생률을 가지고 발생하는 확률과정

- 평균 $\mu = \alpha\beta$, 분산 $\sigma^2 = \alpha\beta^2$ 을 가짐
- 포아송 과정에서, 평균 대기시간이 β 인 사건이 α 번 발생할 때까지의 대기시간에 대한 분포로 해석할 수 있음



<그림 28> α, β 에 따른 감마분포의 변화

감마분포와 지수분포

• 지수분포(Exponential Distribution) (1/3)

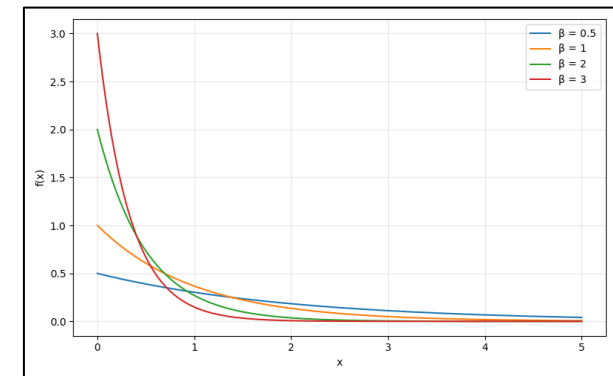
• 정의

- 확률밀도함수가 다음을 따르는 연속형 확률분포

$$f(x; \beta) = \begin{cases} \frac{1}{\beta} e^{-\frac{x}{\beta}}, & x > 0 \\ 0, & \text{그 외의 경우 (단, } \beta > 0) \end{cases}$$

• 특징

- 평균 $\mu = \beta$, 분산 $\sigma^2 = \beta^2$ 을 가짐
- 포아송 과정에서, 평균 대기시간이 β 인 사건이 처음 발생할 때까지의 대기시간에 대한 분포로 해석할 수 있음
- $\alpha = 1$ 일 때의 감마분포임



<그림 29> β 에 따른 지수분포의 변화

감마분포와 지수분포

• 지수분포(Exponential Distribution) (2/3)

• 포아송 과정과의 관계

- 포아송 분포가 특정 시간동안 발생한 사건의 수를 나타낸다면, 지수분포는 사건이 발생하기까지 걸린 시간을 나타냄
- 포아송 과정에서 특정 시간동안 사건이 발생하지 않을 확률로부터 지수분포를 유도할 수 있음
 - t 시간동안 하나의 사건도 발생하지 않을 확률
 - $p(0; \lambda t) = \frac{e^{-\lambda t}(\lambda t)^0}{0!} = e^{-\lambda t}$
 - 첫 번째 포아송 사건이 발생하기까지 소요된 시간을 X 라 할 때, X 가 x 보다 클 확률은 x 시간 내 포아송 사건이 한 건도 발생하지 않을 확률과 같음
 - $P(X > x) = e^{-\lambda x}$
 - X 의 누적분포함수는 다음과 같음
 - $P(0 \leq X \leq x) = 1 - e^{-\lambda x}$
 - 이를 미분하면 $\lambda = \frac{1}{\beta}$ 인 지수분포의 확률밀도함수를 얻게됨
 - $f(x) = \lambda e^{-\lambda x}$

감마분포와 지수분포

• 지수분포(Exponential Distribution) (3/3)

• 건망성(Memoryless)

- 다음 사건 발생까지의 대기시간이 이미 경과한 대기시간에 영향을 받지 않는 성질

- $P(X \geq t_0 + t | X \geq t_0) = P(X \geq t)$

- t_0 시간동안 사건이 발생하지 않은 후, 추가로 t 시간동안 사건이 발생하지 않을 확률이 t 시간동안 사건이 발생하지 않을 확률과 같음

- 포아송 과정과의 관계를 통해 $P(X > x) = e^{-\lambda x}$ 이므로,
 $P(X \geq t_0 + t) = e^{-\lambda(t_0+t)}, P(X \geq t_0) = e^{-\lambda t_0}, P(X \geq t) = e^{-\lambda t}$

- $P(X \geq t_0 + t | X \geq t_0) = \frac{P(X \geq t_0 + t)}{P(X \geq t_0)} = \frac{e^{-\lambda(t_0+t)}}{e^{-\lambda t_0}} = \frac{e^{-\lambda t_0} \times e^{-\lambda t}}{e^{-\lambda t_0}} = e^{-\lambda t}$

- $\therefore P(X \geq t_0 + t | X \geq t_0) = P(X \geq t) = e^{-\lambda t}$

- 이는 이미 t_0 시간동안 사건이 발생하지 않았더라도 다음 사건까지의 대기시간 t 가 영향을 받지 않음을 의미

감마분포와 지수분포

• 예제 6.17

어떤 부품이 고장나기까지의 시간(단위: 년)을 나타내는 확률변수를 T 라고 하자. 그리고 T 는 고장나기까지의 평균시간이 $\beta = 5$ 인 지수분포를 따른다고 하자. 이 부품 5개가 각각 다른 시스템에 설치되었다고 할 때, 8년이 지난 후 적어도 2개의 부품이 여전히 작동하고 있을 확률은 얼마인가?

- 하나의 부품이 8년 이상 작동할 확률은

$$P(T > 8) = \frac{1}{5} \int_8^{\infty} e^{-\frac{t}{5}} dt = e^{-\frac{8}{5}} \approx 0.2$$

- 확률변수 X 가 8년이 지난 후 작동하고 있을 부품의 수라 하면 이는 $n = 5, p \approx 0.2$ 인 이항분포 $b(x; 5, 0.2)$ 로 볼 수 있으므로,

$$\begin{aligned} P(X \geq 2) &= \sum_{x=2}^5 b(x; 5, 0.2) = 1 - \sum_{x=0}^1 b(x; 5, 0.2) \\ &= 1 - 0.7373 = 0.2627 \end{aligned}$$

∴ 평균 수명이 5년이고 서로 다른 시스템에 설치된 부품 5개 중 8년 후에도 2개 이상의 부품이 작동중일 확률은 26.27%

감마분포와 지수분포

• 예제 6.18

전화교환기에 도착되는 호출신호는 분당 평균이 5회인 포아송 과정을 따른다고 한다. 1분 내에 2번 이상의 호출신호가 도착될 확률을 구하라.

- 확률변수 X 를 2번의 호출신호가 도착되기까지 소요된 시간이라 하면, 이는 2번의 포아송 사건이 발생되기까지 소요된 시간으로 $\alpha = 2$, $\beta = \frac{1}{5}$ 인 감마분포를 따르는 확률변수임
- $$P(X \leq 1) = \int_0^1 \frac{1}{\beta^2} x e^{-x/\beta} dx = 25 \int_0^1 x e^{-5x} dx$$
$$= 1 - e^{-5}(1 + 5) = 0.96$$
$$\therefore \text{분당 호출신호가 5회인 전화교환기가 1분 내에 2번 이상 호출신호를 받을 확률은 96\%}$$

감마분포와 지수분포

• 예제 6.19 (1/2)

쥐를 이용한 생의학 실험에서는 독극물이 생존시간에 미치는 영향을 살펴보기 위해 용량반응(Dose-response) 검사를 이용한다. 연구에 의하면 일정량의 독극물에 대한 생존시간(단위: 주)은 $\alpha = 5$ 이고 $\beta = 10$ 인 감마분포를 따른다고 한다. 어떤 쥐에 독극물을 주입했을 때 60주 이상 생존하지 못할 확률은 얼마인가?

- 확률변수 X 를 생존시간이라 하면,

$$P(X \leq 60) = \frac{1}{\beta^5} \int_0^{60} \frac{x^{\alpha-1} e^{-\frac{x}{\beta}}}{\Gamma(5)} dx$$

- 해당 적분은 불완전 감마함수 $F(x; \alpha) = \int_0^x \frac{y^{\alpha-1} e^{-y}}{\Gamma(\alpha)} dy$ 를 이용하여 구할 수 있음

감마분포와 지수분포

• 예제 6.19 (2/2)

쥐를 이용한 생의학 실험에서는 독극물이 생존시간에 미치는 영향을 살펴보기 위해 용량반응(Dose-response) 검사를 이용한다. 연구에 의하면 일정량의 독극물에 대한 생존시간(단위: 주)은 $\alpha = 5$ 이고 $\beta = 10$ 인 감마분포를 따른다고 한다. 어떤 쥐에 독극물을 주입했을 때 60주 이상 생존하지 못할 확률은 얼마인가?

- 구하고자 하는 확률식에서 $y = \frac{x}{\beta}$, $x = \beta y$ 로 놓으면

$$P(X \leq 60) = \int_0^6 \frac{y^4 e^{-y}}{\Gamma(5)} dy \text{이며, 이는 불완전 감마함수표의}$$

$F(6; 5)$ 값임

- $P(X \leq 60) = F(6; 5) = 0.715$
∴ 쥐에게 독극물을 주입했을 때 60주 이상 생존하지 못할 확률은 71.5%

x	α					
	1	2	3	4	5	6
1	0.6320	0.2640	0.0800	0.0190	0.0040	0.0010
2	0.8650	0.5940	0.3230	0.1430	0.0530	0.0170
3	0.9500	0.8010	0.5770	0.3530	0.1850	0.0840
4	0.9820	0.9080	0.7620	0.5670	0.3710	0.2150
5	0.9930	0.9600	0.8750	0.7350	0.5600	0.3840
6	0.9980	0.9830	0.9380	0.8490	0.7150	0.5540
7	0.9990	0.9930	0.9700	0.9180	0.8270	0.6990

<그림 30> $1 \leq x \leq 7, 1 \leq \alpha \leq 6$ 에서의 $F(x; \alpha)$ 의 불완전 감마함수표

감마분포와 지수분포

• 예제 6.20

과거자료로부터 어느 제품에 대한 고객의 불만제기 사이의 시간간격(단위: 개월)은 $\alpha = 2$ 이고 $\beta = 4$ 인 감마분포를 따른다고 알려져 있다. 최근 품질관리를 철저히 실시하고 난 후 첫 번째 불만이 발생할 때까지 20개월이 소요되었다. 이것으로부터 품질관리를 철저히 실시한 것이 효과적이었다고 할 수 있는가?

- 확률변수 X 를 품질관리 전 첫 번째 불만제기까지의 시간이라 하면,

$$P(X \geq 20) = 1 - \frac{1}{\beta^\alpha} \int_0^{20} \frac{x^{\alpha-1} e^{-\frac{x}{\beta}}}{\Gamma(\alpha)} dx$$

- $y = \frac{x}{\beta}, x = \beta y$ 로 놓으면 $F(5; 2) = 0.96$ 이므로

$$\begin{aligned} P(X \geq 20) &= 1 - \int_0^5 \frac{y e^{-y}}{\Gamma(2)} dy = 1 - F(5; 2) \\ &= 1 - 0.96 = 0.04 \end{aligned}$$

- ∴ 이는 자주 발생했다고 보기 어려우며, 따라서 품질관리가 효과적이었다고 볼 수 있음

x	α			
	1	2	3	4
1	0.6320	0.2640	0.0800	0.0190
2	0.8650	0.5940	0.3230	0.1430
3	0.9500	0.8010	0.5770	0.3530
4	0.9820	0.9080	0.7620	0.5670
5	0.9930	0.9600	0.8750	0.7350

<그림 31> $1 \leq x \leq 5, 1 \leq \alpha \leq 4$ 에서의 $F(x; \alpha)$ 의 불완전 감마함수표

감마분포와 지수분포

• 예제 6.21

어느 세탁기가 고장날 때까지의 시간 Y 는 다음의 밀도함수를 따른다.

$$f(y) = \begin{cases} \frac{1}{4} e^{-\frac{y}{4}}, & y \geq 0 \\ 0, & \text{그 외의 경우} \end{cases}$$

이것은 $\mu = 4$ 년인 지수분포이다. 이 세탁기가 6년 이상 고장나지 않을 확률은 얼마인가? 또한 1년 이내에 고장날 확률은 얼마인가?

- 지수분포의 누적분포함수 $F(y) = \frac{1}{\beta} \int_0^y e^{-\frac{t}{\beta}} dt = 1 - e^{-\frac{y}{\beta}}$
- $P(Y > 6) = 1 - F(6) = e^{-\frac{3}{2}} = 0.2231$
∴ 평균 수명 4년인 세탁기가 6년 이상 고장나지 않을 확률 22.31%
- $P(Y < 1) = 1 - e^{-\frac{1}{4}} = 1 - 0.7788 = 0.2212$
∴ 평균 수명 4년인 세탁기가 1년 이내에 고장날 확률 22.12%

목 차

- 연속형 균일분포
- 정규분포와 표준정규분포
- 이항분포의 정규근사
- 감마분포와 지수분포
- 카이제곱분포와 베타분포
- 로그정규분포와 와이블분포

카이제곱분포와 베타분포

• 카이제곱분포(Chi-square Distribution) (1/4)

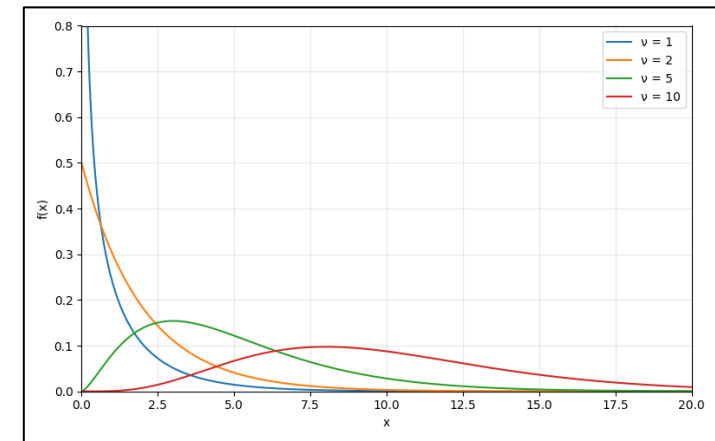
• 정의

- 확률밀도함수가 다음을 따르는 연속형 확률분포

$$f(x; v) = \begin{cases} \frac{1}{2^{\frac{v}{2}} \Gamma(\frac{v}{2})} x^{\frac{v}{2}-1} e^{-\frac{x}{2}}, & x > 0 \\ 0, & \text{그 외의 경우(단, } v \text{는 양의 정수)} \end{cases}$$

• 특징

- 평균 $\mu = v$, 분산 $\sigma^2 = 2v$ 를 가짐
- 자유도 v 에 대해 $\alpha = \frac{v}{2}, \beta = 2$ 일 때의 감마분포임
- 자유도가 작을수록 왼쪽으로 치우침
- 관측도수와 기대도수의 차이를 통해 적합도를 검정할 때 사용될 수 있음



<그림 32> 자유도에 따른 카이제곱분포의 변화

카이제곱분포와 베타분포

- 카이제곱분포(Chi-square Distribution) (2/4)
- 적합도 검정(Goodness-of-fit Test)
 - 관측도수 o_i , 기대도수 e_i 에 대해 카이제곱값 $X^2 = \sum_{i=1}^k \frac{(o_i - e_i)^2}{e_i}$ 가 자유도 v 와 유의수준 α 에 따른 임계값 $X_{\alpha, v}^2$ 을 비교함
 - $X^2 \leq X_{\alpha, v}^2$ 인 경우, 관측도수가 기대분포와 통계적으로 유의미한 차이가 없다고 판단함

카이제곱분포와 베타분포

- 카이제곱분포(Chi-square Distribution) (3/3)
- 예제 6.22 (1/2)

주사위 180번을 던져서 다음과 같은 결과를 얻었다.

x	1	2	3	4	5	6
도수	28	36	36	30	27	23

주사위는 공정하게 만들어졌는가를 유의수준 0.01로 검정하라.

- 기대도수 $e_i = 180 \times \frac{1}{6} = 30$
- 카이제곱값 $X^2 = \sum_{i=1}^6 \frac{(o_i - e_i)^2}{e_i}$ 을 계산하면
1: $\frac{(28-30)^2}{30} = \frac{4}{30} \approx 0.1333$, 2: $\frac{(36-30)^2}{30} = \frac{36}{30} = 1.2$, 3: $\frac{(36-30)^2}{30} = \frac{36}{30} = 1.2$,
4: $\frac{(30-30)^2}{30} = 0$, 5: $\frac{(27-30)^2}{30} = \frac{9}{30} = 0.3$, 6: $\frac{(23-30)^2}{30} = \frac{49}{30} \approx 1.6333$
 $X^2 \approx 0.1333 + 1.2 + 1.2 + 0 + 0.3 + 1.6333 \approx 4.4666 \approx 4.47$

카이제곱분포와 베타분포

- 카이제곱분포(Chi-square Distribution) (4/4)
- 예제 6.22 (2/2)

주사위 180번을 던져서 다음과 같은 결과를 얻었다.

x	1	2	3	4	5	6
도수	28	36	36	30	27	23

주사위는 공정하게 만들어졌는가를 유의수준 0.01로 검정하라.

- 유의수준 $\alpha = 0.01$, 자유도 $\nu = 5$ 에 대한 임계값 $X_{0.01,5}^2 \approx 15.09$
 $\therefore (X^2 \approx 4.47) < (X_{0.01,5}^2 \approx 15.09)$ 이므로 관측도수가 기대분포와 통계적으로 유의미한 차이가 없다고 판단됨

ν	α							
	0.30	0.25	0.20	0.10	0.05	0.025	0.02	0.01
1	1.074	1.323	1.642	2.706	3.841	5.024	5.412	6.635
2	2.408	2.773	3.219	4.605	5.991	7.378	7.824	9.210
3	3.665	4.108	4.642	6.251	7.815	9.348	9.837	11.345
4	4.878	5.385	5.989	7.779	9.488	11.143	11.668	13.277
5	6.064	6.626	7.289	9.236	11.070	12.832	13.388	15.086

<그림 33> 유의수준 $\alpha = 0.01$, 자유도 $\nu = 5$ 에 대한 임계값을 찾기 위한 카이제곱분포표

카이제곱분포와 베타분포

- 베타분포(Beta Distribution) (1/3)

- 베타함수(Beta Function)

- 모수 $\alpha, \beta > 0$ 와 감마함수 $\Gamma(\alpha)$ 에 대해 다음을 만족하는 함수

- $B(\alpha, \beta) = \int_0^1 x^{\alpha-1} (1-x)^{\beta-1} dx = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)}$

- 증명

- $\Gamma(\alpha)\Gamma(\beta) = \left(\int_0^\infty x^{\alpha-1} e^{-x} dx\right) \left(\int_0^\infty y^{\beta-1} e^{-y} dy\right)$
 $= \int_0^\infty \int_0^\infty x^{\alpha-1} y^{\beta-1} e^{-(x+y)} dx dy$

- $t = x + y, s = \frac{x}{x+y}$ 로 하면 $x = st, y = (1-s)t, dx dy = t ds dt$ 이므로,

- $x^{\alpha-1} y^{\beta-1} e^{-(x+y)} = (st)^{\alpha-1} ((1-s)t)^{\beta-1} e^{-t}$
 $= s^{\alpha-1} t^{\alpha-1} (1-s)^{\beta-1} t^{\beta-1} e^{-t} = s^{\alpha-1} (1-s)^{\beta-1} t^{\alpha+\beta-2} e^{-t}$

- $\Gamma(\alpha)\Gamma(\beta) = \int_0^\infty \int_0^1 s^{\alpha-1} (1-s)^{\beta-1} t^{\alpha+\beta-1} e^{-t} ds dt$
 $= \left(\int_0^\infty t^{\alpha+\beta-1} e^{-t} dt\right) \left(\int_0^1 s^{\alpha-1} (1-s)^{\beta-1} ds\right) = \Gamma(\alpha + \beta) B(\alpha, \beta)$

- $\therefore B(\alpha, \beta) = \Gamma(\alpha)\Gamma(\beta)/\Gamma(\alpha + \beta)$

카이제곱분포와 베타분포

• 베타분포(Beta Distribution) (2/3)

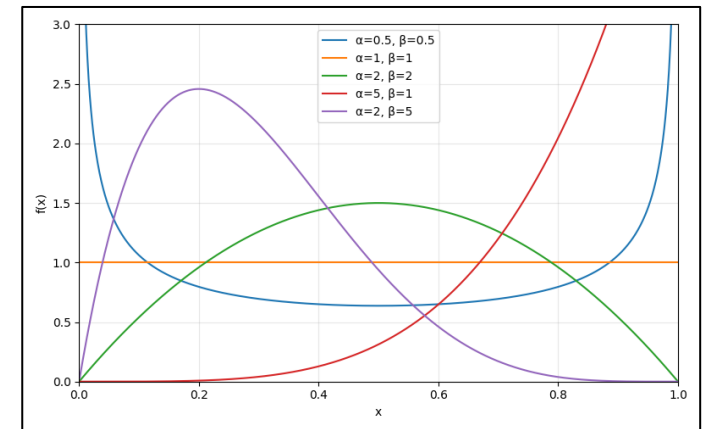
• 정의

- 확률밀도함수가 다음을 따르는 연속형 확률분포

$$f(x) = \begin{cases} \frac{1}{B(\alpha, \beta)} x^{\alpha-1} (1-x)^{\beta-1}, & 0 < x < 1 \\ 0, & \text{그 외의 경우 (단, } \alpha > 0, \beta > 0) \end{cases}$$

• 특징

- 평균 $\mu = \frac{\alpha}{\alpha+\beta}$, 분산 $\sigma^2 = \frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}$ 을 가짐
- $\alpha = 1, \beta = 1$ 인 경우, $[0,1]$ 구간에서 균일분포가 됨
- 정의역이 0~1로, 확률이나 비율에 대한 분포를 다룰 때 주로 사용함
- 베이즈 추정법에서 사전 분포로 사용됨



<그림 34> α, β 에 따른 베타분포의 변화

카이제곱분포와 베타분포

• 베타분포(Beta Distribution) (3/3)

• 예제 6.22

어떤 상표의 텔레비전이 구입 첫 해에 서비스를 받는 비율이 $\alpha = 3$ 이고 $\beta = 2$ 인 베타분포를 따르는 확률변수일 때, 올해에 팔린 이 상표의 텔레비전이 구입 첫 해에 서비스를 받을 확률이 최소 80% 이상일 확률은 얼마인가?

- $\alpha = 3, \beta = 2$ 이므로 해당 베타분포의 확률밀도함수

$$f(x) = \frac{1}{B(\alpha, \beta)} x^{\alpha-1} (1-x)^{\beta-1} = \frac{1}{B(3, 2)} x^{3-1} (1-x)^{2-1} = 12x^2(1-x)$$

- 80% 이상일 확률은 다음과 같음

$$\begin{aligned} P(X > 0.8) &= \int_{0.8}^1 12x^2(1-x) dx = 12 \int_{0.8}^1 (x^2 - x^3) dx \\ &= 12 \left[\frac{x^3}{3} - \frac{x^4}{4} \right]_{0.8}^1 = 12 \left(\frac{1}{12} - \frac{128}{1875} \right) = 12 \times \frac{113}{7500} \\ &= \frac{226}{1250} \approx 0.1808 \end{aligned}$$

$\therefore$ 구입 첫 해에 서비스를 받을 확률이 80% 이상일 확률은 약 18.08%

목 차

- 연속형 균일분포
- 정규분포와 표준정규분포
- 이항분포의 정규근사
- 감마분포와 지수분포
- 카이제곱분포와 베타분포
- 로그정규분포와 와이블분포

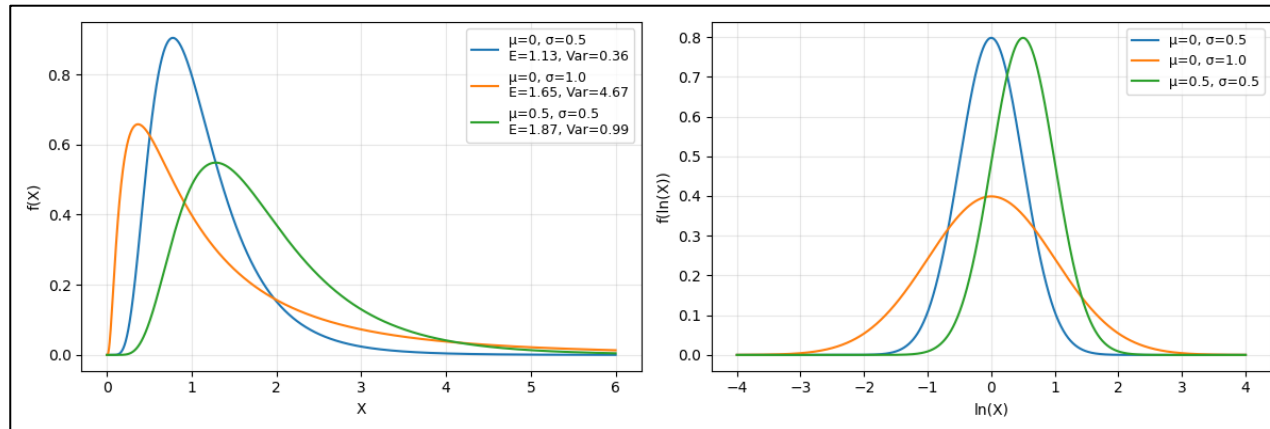
로그정규분포와 와이블분포

• 로그정규분포(Lognormal Distribution) (1/4)

• 정의

- 확률변수 $Y = \ln(X)$ 가 평균 μ , 표준편차 σ 인 정규분포일 때 확률변수 X 가 가지는 확률분포
- 확률밀도함수가 다음을 따르는 연속형 확률분포

$$f(x; \mu, \sigma) = \begin{cases} \frac{1}{\sqrt{2\pi}\sigma x} e^{-\frac{1}{2\sigma^2}[\ln(x)-\mu]^2}, & x \geq 0 \\ 0, & x < 0 \end{cases}$$



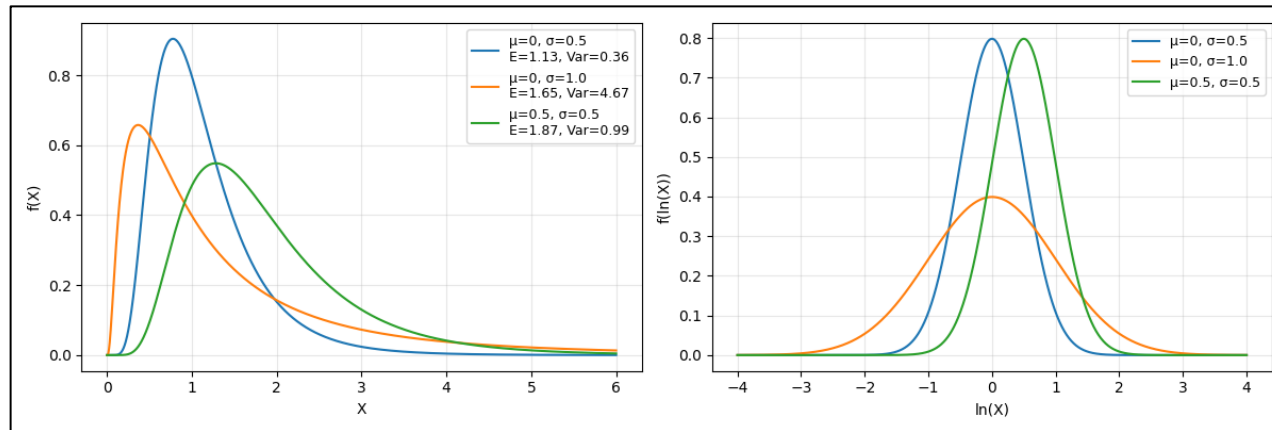
<그림 35> $\ln(X)$ 의 μ, σ 에 따른 로그정규분포의 변화

로그정규분포와 와이블분포

• 로그정규분포(Lognormal Distribution) (2/4)

• 특징

- 평균 $E(X) = e^{\mu + \frac{\sigma^2}{2}}$, 분산 $Var(X) = e^{2\mu + \sigma^2}(e^{\sigma^2} - 1)$ 을 가짐
- 오른쪽에 긴 꼬리가 생기는 비대칭 그래프를 그림
- 항상 양수를 가지며 곱셈적 요인을 가지는 데이터를 분석할 때 주로 사용됨



<그림 35> $\ln(X)$ 의 μ, σ 에 따른 로그정규분포의 변화

로그정규분포와 와이블분포

• 로그정규분포(Lognormal Distribution) (3/4)

• 예제 6.23

화학공장에서 배출되는 오염물질의 농도(ppm)가 $\mu = 3.2$ 이고 $\sigma = 1$ 인 로그정규분포를 따른다고 하자. 농도가 8ppm을 초과할 확률은 얼마인가?

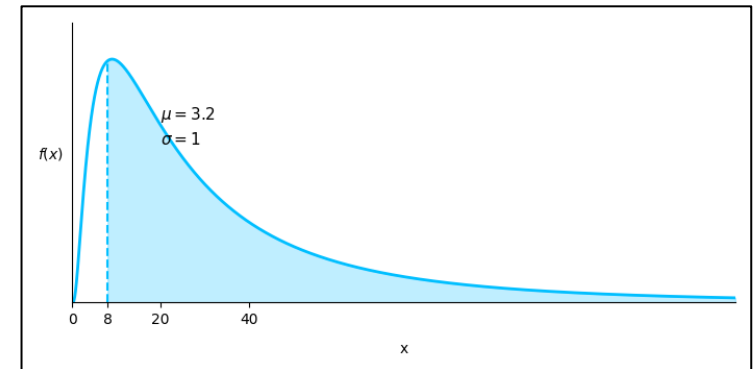
- 확률변수 X 를 오염물질의 농도라 하면, $P(X > 8) = 1 - P(X \leq 8)$
- $\ln(X)$ 는 평균 $\mu = 3.2$ 이고 표준편차 $\sigma = 1$ 인 정규분포를 따르므로

$$P(X \leq 8) = \Phi \left[\frac{\ln(8) - 3.2}{1} \right] = \Phi(-1.12) = 0.1314$$

(단, Φ 는 표준정규분포의 누적분포함수)

- $P(X > 8) = 1 - P(X \leq 8) = 1 - 0.1314 = 0.8686$

$\therefore$ 오염물질 농도가 8ppm을 초과할 확률은 86.86%



<그림 36> 예제 6.23의 확률밀도함수도

로그정규분포와 와이블분포

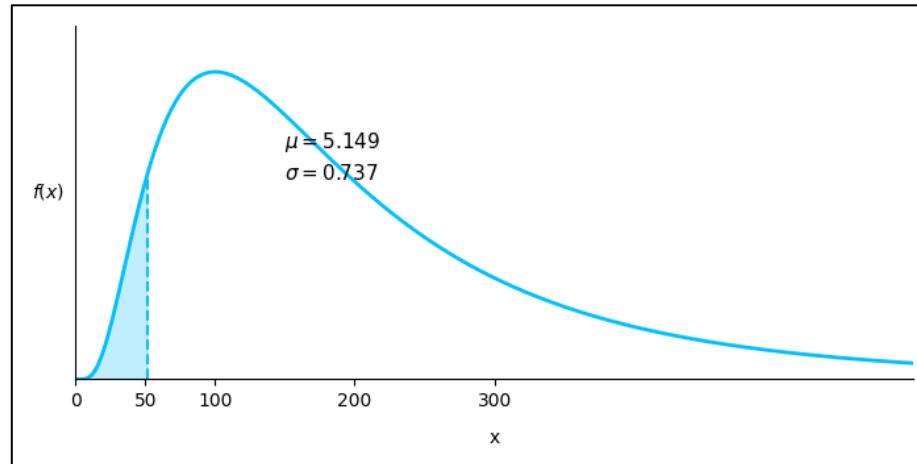
• 로그정규분포(Lognormal Distribution) (4/4)

• 예제 6.24

기관차에 사용되는 어떤 전자제어장치의 수명(단위: 1000마일)은 $\mu = 5.149$ 이고 $\sigma = 0.737$ 인 로그정규분포를 따른다고 한다. 수명의 5 백분위수를 구하라.

- 확률변수 X 를 장치의 수명이라 하면, $P(Z < -1.645) = 0.0500$ 이고 $\ln(x)$ 는 평균 $\mu = 5.149$, 표준편차 $\sigma = 0.737$ 인 정규분포를 따르므로 $\ln(x) = 5.149 + (0.737)(-1.645) = 3.937$

$\therefore x = 51.265$ 이며, 이는 장치의 5%만이 51,265 마일 이하의 수명을 가짐을 의미함



<그림 37> 예제 6.24의 확률밀도함수도

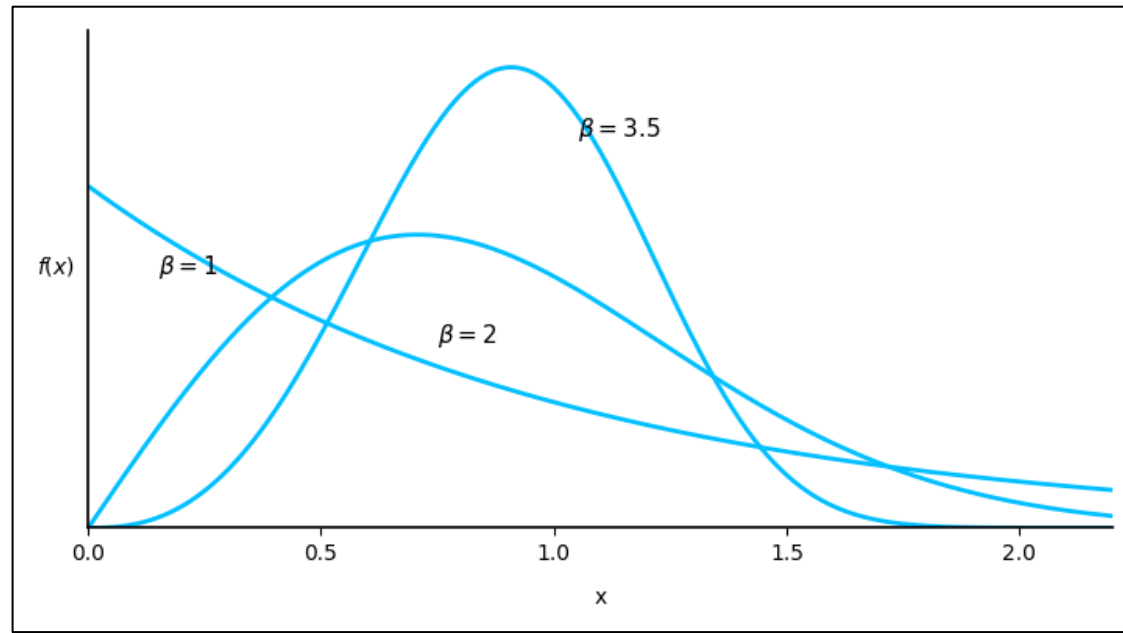
로그정규분포와 와이블분포

- 와이블분포(Weibull Distribution) (1/6)

- 정의

- 확률밀도함수가 다음을 따르는 연속형 확률분포

- $$f(x; \alpha, \beta) = \begin{cases} \alpha \beta x^{\beta-1} e^{-\alpha x^{\beta}}, & x > 0 \\ 0, & \text{그 외의 경우 (단, } \alpha > 0, \beta > 0) \end{cases}$$



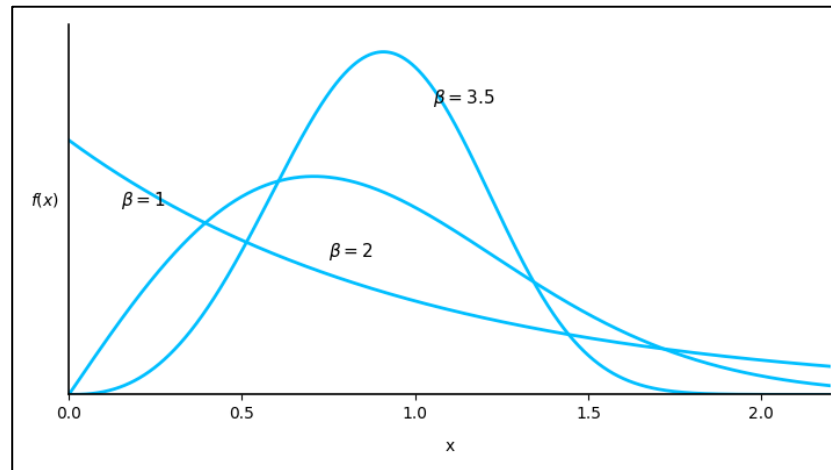
<그림 38> $\alpha = 1$ 일 때 β 에 따른 와이블분포 변화

로그정규분포와 와이블분포

• 와이블분포(Weibull Distribution) (2/6)

• 특징

- 평균 $\mu = \alpha^{-\frac{1}{\beta}} \Gamma\left(1 + \frac{1}{\beta}\right)$, 분산 $\sigma^2 = \alpha^{-\frac{2}{\beta}} \left\{ \Gamma\left(1 + \frac{2}{\beta}\right) - \left[\Gamma\left(1 + \frac{1}{\beta}\right) \right]^2 \right\}$ 을 가짐
- $\beta = 1$ 일 때 지수분포의 형태를 띰
- 누적분포함수 $F(x) = 1 - e^{-\alpha x^\beta}$ (단, $x \geq 0, \alpha > 0, \beta > 0$)
- 시간에 따른 고장률 분포를 나타내기 위해 주로 사용됨



<그림 38> $\alpha = 1$ 일 때 β 에 따른 와이블분포 변화

로그정규분포와 와이블분포

- 와이블분포(Weibull Distribution) (3/6)

- 고장률(Failure Rate 또는 Hazard Rate) (1/2)

- 어느 부품의 신뢰도 $R(t)$ 를 최소한 어느 특정시간 t 까지 부품이 작동할 확률로 정의하면 다음과 같음

- $R(t) = P(T > t) = \int_t^{\infty} f(t) dt = 1 - F(t)$

- $F(t)$ 가 T 의 누적분포이므로 부품이 t 시간 작동했을 때, 구간 $T = t$ 와 $T = t + \Delta t$ 사이에 고장날 조건부확률은 다음과 같음

- $\frac{F(t+\Delta t) - F(t)}{R(t)}$

- 이를 Δt 로 나누고 $\Delta t \rightarrow 0$ 으로 극한을 취하면 고장률 $Z(t)$ 를 다음과 같이 얻을 수 있음

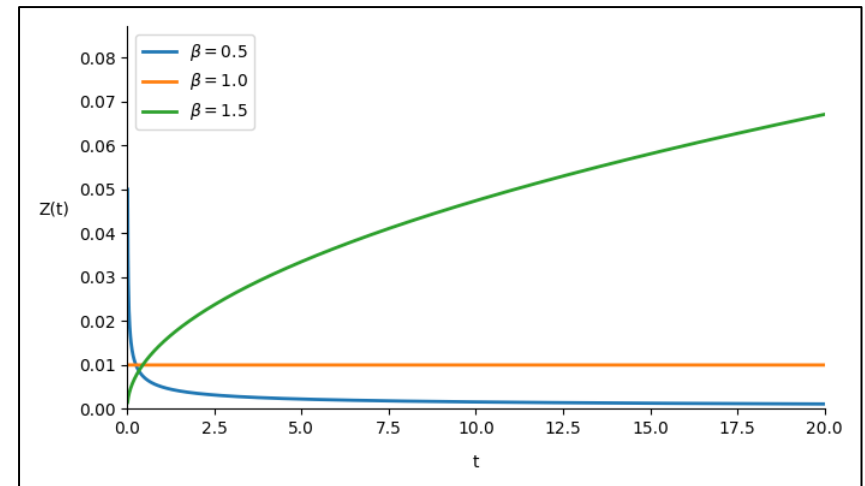
- $$Z(t) = \lim_{t \rightarrow 0} \frac{F(t+\Delta t) - F(t)}{\Delta t} \frac{1}{R(t)} = \frac{F'(t)}{R(t)} = \frac{f(t)}{R(t)} = \frac{f(t)}{1 - F(t)}$$
$$= \frac{\alpha \beta t^{\beta-1} e^{-\alpha t^\beta}}{1 - 1 + e^{-\alpha t^\beta}} = \alpha \beta t^{\beta-1}$$

로그정규분포와 와이블분포

- 와이블분포(Weibull Distribution) (4/6)

- 고장률(Failure Rate, Hazard Rate) (2/2)

- 고장률 $Z(t) = \alpha\beta t^{\beta-1}$ 는 부품이 t 시간동안 고장나지 않았을 때, t 부터 Δt 만큼 더 생존할 조건부확률의 변화율을 의미함
- $\beta = 1$ 이면 고장률은 α 로서 상수가 되고, 지수분포의 형태를 띄며 건망성 특성을 가짐
- $\beta > 1$ 이면 $Z(t)$ 는 증가함수로, 부품이 시간에 따라 마모되는 현상을 표현함
- $\beta < 1$ 이면 $Z(t)$ 는 감소함수로, 시간이 지남에 따라 오히려 강해짐을 의미함



<그림 39> β 에 따른 고장률 변화

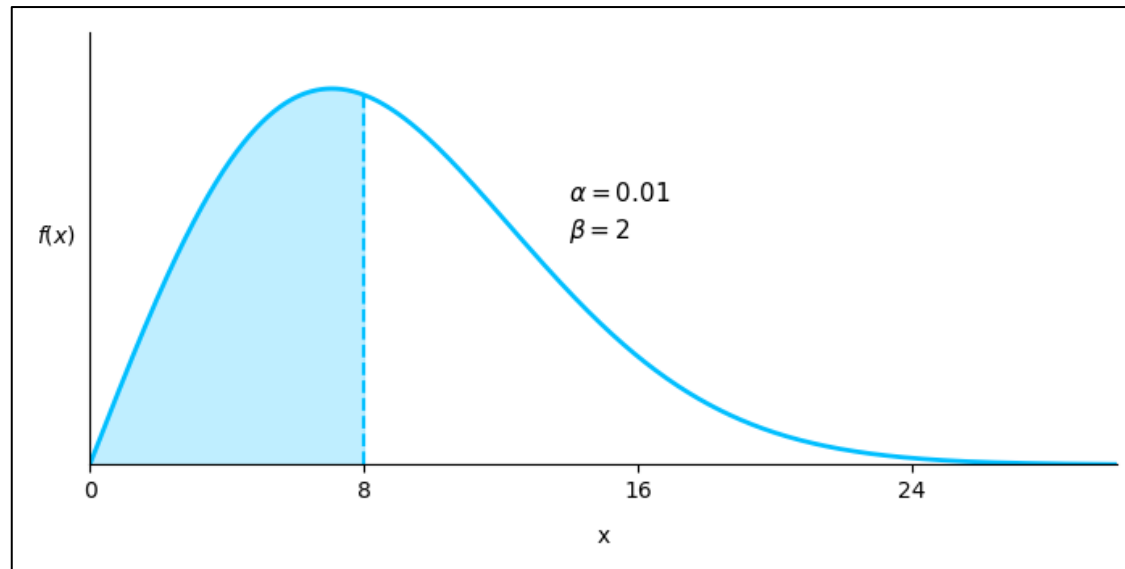
로그정규분포와 와이블분포

- 와이블분포(Weibull Distribution) (5/6)

- 예제 6.25

어느 제품의 수명(단위: 시간)은 $\alpha = 0.01$ 이고 $\beta = 2$ 인 와이블분포를 따른다고 한다. 8시간 이전에 이 제품이 고장날 확률은 얼마인가?

- $P(X < 8) = F(8) = 1 - e^{-(0.01)8^2} = 1 - 0.5273 = 0.4727$



<그림 40> 예제 6.25의 확률밀도함수도

로그정규분포와 와이블분포

• 와이블분포(Weibull Distribution) (6/6)

• 예제 6.26

자동차 배기통의 수명은 고장률이 $Z(t) = \frac{1}{\sqrt{t}}$ 인 와이블분포를 따른다. 배기통이 4년 후에도 계속 사용될 확률을 구하라.

- $Z(t) = \alpha\beta t^{\beta-1} = \frac{1}{\sqrt{t}}$ 이므로 $\beta - 1 = -\frac{1}{2}$ 이어야 함
 - $\beta - 1 = -\frac{1}{2}$ 에서 $\beta = \frac{1}{2}$ 이므로 $\frac{1}{2}\alpha \times \frac{1}{\sqrt{t}} = \frac{1}{\sqrt{t}} \Rightarrow \frac{\alpha}{2} = 1 \Rightarrow \alpha = 2$ 가 됨
 - $P(X > 4) = 1 - P(X < 4) = 1 - F(4) = 1 - 1 + e^{-\alpha x^\beta}$
 $= e^{-2 \times 4^{\frac{1}{2}}} = e^{-4} \approx 0.0183$
- $\therefore$ 자동차 배기통이 4년 후에도 계속 사용될 확률은 1.83%

Thanks!

이 성 현(seonghyeon@pel.sejong.ac.kr)

부록#1

• 표준정규분포표 (1/2)

<i>z</i>	.00	.01	.02	.03	.04	.05	.06	.07	.08	.09
-3.4	0.0003	0.0003	0.0003	0.0003	0.0003	0.0003	0.0003	0.0003	0.0003	0.0002
-3.3	0.0005	0.0005	0.0005	0.0004	0.0004	0.0004	0.0004	0.0004	0.0004	0.0003
-3.2	0.0007	0.0007	0.0006	0.0006	0.0006	0.0006	0.0006	0.0005	0.0005	0.0005
-3.1	0.0010	0.0009	0.0009	0.0009	0.0008	0.0008	0.0008	0.0008	0.0007	0.0007
-3.0	0.0013	0.0013	0.0013	0.0012	0.0012	0.0011	0.0011	0.0011	0.0010	0.0010
-2.9	0.0019	0.0018	0.0018	0.0017	0.0016	0.0016	0.0015	0.0015	0.0014	0.0014
-2.8	0.0026	0.0025	0.0024	0.0023	0.0023	0.0022	0.0021	0.0021	0.0020	0.0019
-2.7	0.0035	0.0034	0.0033	0.0032	0.0031	0.0030	0.0029	0.0028	0.0027	0.0026
-2.6	0.0047	0.0045	0.0044	0.0043	0.0041	0.0040	0.0039	0.0038	0.0037	0.0036
-2.5	0.0062	0.0060	0.0059	0.0057	0.0055	0.0054	0.0052	0.0051	0.0049	0.0048
-2.4	0.0082	0.0080	0.0078	0.0075	0.0073	0.0071	0.0069	0.0068	0.0066	0.0064
-2.3	0.0107	0.0104	0.0102	0.0099	0.0096	0.0094	0.0091	0.0089	0.0087	0.0084
-2.2	0.0139	0.0136	0.0132	0.0129	0.0125	0.0122	0.0119	0.0116	0.0113	0.0110
-2.1	0.0179	0.0174	0.0170	0.0166	0.0162	0.0158	0.0154	0.0150	0.0146	0.0143
-2.0	0.0228	0.0222	0.0217	0.0212	0.0207	0.0202	0.0197	0.0192	0.0188	0.0183
-1.9	0.0287	0.0281	0.0274	0.0268	0.0262	0.0256	0.0250	0.0244	0.0239	0.0233
-1.8	0.0359	0.0351	0.0344	0.0336	0.0329	0.0322	0.0314	0.0307	0.0301	0.0294
-1.7	0.0446	0.0436	0.0427	0.0418	0.0409	0.0401	0.0392	0.0384	0.0375	0.0367
-1.6	0.0548	0.0537	0.0526	0.0516	0.0505	0.0495	0.0485	0.0475	0.0465	0.0455
-1.5	0.0668	0.0655	0.0643	0.0630	0.0618	0.0606	0.0594	0.0582	0.0571	0.0559
-1.4	0.0808	0.0793	0.0778	0.0764	0.0749	0.0735	0.0721	0.0708	0.0694	0.0681
-1.3	0.0968	0.0951	0.0934	0.0918	0.0901	0.0885	0.0869	0.0853	0.0838	0.0823
-1.2	0.1151	0.1131	0.1112	0.1093	0.1075	0.1056	0.1038	0.1020	0.1003	0.0985
-1.1	0.1357	0.1335	0.1314	0.1292	0.1271	0.1251	0.1230	0.1210	0.1190	0.1170
-1.0	0.1587	0.1562	0.1539	0.1515	0.1492	0.1469	0.1446	0.1423	0.1401	0.1379
-0.9	0.1841	0.1814	0.1788	0.1762	0.1736	0.1711	0.1685	0.1660	0.1635	0.1611
-0.8	0.2119	0.2090	0.2061	0.2033	0.2005	0.1977	0.1949	0.1922	0.1894	0.1867
-0.7	0.2420	0.2389	0.2358	0.2327	0.2296	0.2266	0.2236	0.2206	0.2177	0.2148
-0.6	0.2743	0.2709	0.2676	0.2643	0.2611	0.2578	0.2546	0.2514	0.2483	0.2451
-0.5	0.3085	0.3050	0.3015	0.2981	0.2946	0.2912	0.2877	0.2843	0.2810	0.2776
-0.4	0.3446	0.3409	0.3372	0.3336	0.3300	0.3264	0.3228	0.3192	0.3156	0.3121
-0.3	0.3821	0.3783	0.3745	0.3707	0.3669	0.3632	0.3594	0.3557	0.3520	0.3483
-0.2	0.4207	0.4168	0.4129	0.4090	0.4052	0.4013	0.3974	0.3936	0.3897	0.3859
-0.1	0.4602	0.4562	0.4522	0.4483	0.4443	0.4404	0.4364	0.4325	0.4286	0.4247
-0.0	0.5000	0.4960	0.4920	0.4880	0.4840	0.4801	0.4761	0.4721	0.4681	0.4641

부록#1

• 표준정규분포표 (2/2)

<i>z</i>	.00	.01	.02	.03	.04	.05	.06	.07	.08	.09
0.0	0.5000	0.5040	0.5080	0.5120	0.5160	0.5199	0.5239	0.5279	0.5319	0.5359
0.1	0.5398	0.5438	0.5478	0.5517	0.5557	0.5596	0.5636	0.5675	0.5714	0.5753
0.2	0.5793	0.5832	0.5871	0.5910	0.5948	0.5987	0.6026	0.6064	0.6103	0.6141
0.3	0.6179	0.6217	0.6255	0.6293	0.6331	0.6368	0.6406	0.6443	0.6480	0.6517
0.4	0.6554	0.6591	0.6628	0.6664	0.6700	0.6736	0.6772	0.6808	0.6844	0.6879
0.5	0.6915	0.6950	0.6985	0.7019	0.7054	0.7088	0.7123	0.7157	0.7190	0.7224
0.6	0.7257	0.7291	0.7324	0.7357	0.7389	0.7422	0.7454	0.7486	0.7517	0.7549
0.7	0.7580	0.7611	0.7642	0.7673	0.7704	0.7734	0.7764	0.7794	0.7823	0.7852
0.8	0.7881	0.7910	0.7939	0.7967	0.7995	0.8023	0.8051	0.8078	0.8106	0.8133
0.9	0.8159	0.8186	0.8212	0.8238	0.8264	0.8289	0.8315	0.8340	0.8365	0.8389
1.0	0.8413	0.8438	0.8461	0.8485	0.8508	0.8531	0.8554	0.8577	0.8599	0.8621
1.1	0.8643	0.8665	0.8686	0.8708	0.8729	0.8749	0.8770	0.8790	0.8810	0.8830
1.2	0.8849	0.8869	0.8888	0.8907	0.8925	0.8944	0.8962	0.8980	0.8997	0.9015
1.3	0.9032	0.9049	0.9066	0.9082	0.9099	0.9115	0.9131	0.9147	0.9162	0.9177
1.4	0.9192	0.9207	0.9222	0.9236	0.9251	0.9265	0.9279	0.9292	0.9306	0.9319
1.5	0.9332	0.9345	0.9357	0.9370	0.9382	0.9394	0.9406	0.9418	0.9429	0.9441
1.6	0.9452	0.9463	0.9474	0.9484	0.9495	0.9505	0.9515	0.9525	0.9535	0.9545
1.7	0.9554	0.9564	0.9573	0.9582	0.9591	0.9599	0.9608	0.9616	0.9625	0.9633
1.8	0.9641	0.9649	0.9656	0.9664	0.9671	0.9678	0.9686	0.9693	0.9699	0.9706
1.9	0.9713	0.9719	0.9726	0.9732	0.9738	0.9744	0.9750	0.9756	0.9761	0.9767
2.0	0.9772	0.9778	0.9783	0.9788	0.9793	0.9798	0.9803	0.9808	0.9812	0.9817
2.1	0.9821	0.9826	0.9830	0.9834	0.9838	0.9842	0.9846	0.9850	0.9854	0.9857
2.2	0.9861	0.9864	0.9868	0.9871	0.9875	0.9878	0.9881	0.9884	0.9887	0.9890
2.3	0.9893	0.9896	0.9898	0.9901	0.9904	0.9906	0.9909	0.9911	0.9913	0.9916
2.4	0.9918	0.9920	0.9922	0.9925	0.9927	0.9929	0.9931	0.9932	0.9934	0.9936
2.5	0.9938	0.9940	0.9941	0.9943	0.9945	0.9946	0.9948	0.9949	0.9951	0.9952
2.6	0.9953	0.9955	0.9956	0.9957	0.9959	0.9960	0.9961	0.9962	0.9963	0.9964
2.7	0.9965	0.9966	0.9967	0.9968	0.9969	0.9970	0.9971	0.9972	0.9973	0.9974
2.8	0.9974	0.9975	0.9976	0.9977	0.9977	0.9978	0.9979	0.9979	0.9980	0.9981
2.9	0.9981	0.9982	0.9982	0.9983	0.9984	0.9984	0.9985	0.9985	0.9986	0.9986
3.0	0.9987	0.9987	0.9987	0.9988	0.9988	0.9989	0.9989	0.9989	0.9990	0.9990
3.1	0.9990	0.9991	0.9991	0.9991	0.9992	0.9992	0.9992	0.9992	0.9993	0.9993
3.2	0.9993	0.9993	0.9994	0.9994	0.9994	0.9994	0.9994	0.9995	0.9995	0.9995
3.3	0.9995	0.9995	0.9995	0.9996	0.9996	0.9996	0.9996	0.9996	0.9996	0.9997
3.4	0.9997	0.9997	0.9997	0.9997	0.9997	0.9997	0.9997	0.9997	0.9997	0.9998

부록#2

- 불완전 감마함수표

x	α									
	1	2	3	4	5	6	7	8	9	10
1	0.6320	0.2640	0.0800	0.0190	0.0040	0.0010	0.0000	0.0000	0.0000	0.0000
2	0.8650	0.5940	0.3230	0.1430	0.0530	0.0170	0.0050	0.0010	0.0000	0.0000
3	0.9500	0.8010	0.5770	0.3530	0.1850	0.0840	0.0340	0.0120	0.0040	0.0010
4	0.9820	0.9080	0.7620	0.5670	0.3710	0.2150	0.1110	0.0510	0.0210	0.0080
5	0.9930	0.9600	0.8750	0.7350	0.5600	0.3840	0.2380	0.1330	0.0680	0.0320
6	0.9980	0.9830	0.9380	0.8490	0.7150	0.5540	0.3940	0.2560	0.1530	0.0840
7	0.9990	0.9930	0.9700	0.9180	0.8270	0.6990	0.5500	0.4010	0.2710	0.1700
8	1.0000	0.9970	0.9860	0.9580	0.9000	0.8090	0.6870	0.5470	0.4070	0.2830
9		0.9990	0.9940	0.9790	0.9450	0.8840	0.7930	0.6760	0.5440	0.4130
10		1.0000	0.9970	0.9900	0.9710	0.9330	0.8700	0.7800	0.6670	0.5420
11			0.9990	0.9950	0.9850	0.9620	0.9210	0.8570	0.7680	0.6590
12			1.0000	0.9980	0.9920	0.9800	0.9540	0.9110	0.8450	0.7580
13				0.9990	0.9960	0.9890	0.9740	0.9460	0.9000	0.8340
14				1.0000	0.9980	0.9940	0.9860	0.9680	0.9380	0.8910
15					0.9990	0.9970	0.9920	0.9820	0.9630	0.9300

부록#3

• 카이제곱분포표 (1/2)

v	α									
	0.995	0.99	0.98	0.975	0.95	0.90	0.80	0.75	0.70	0.50
1	0.0 ⁴ 393	0.0 ³ 157	0.0 ³ 628	0.0 ³ 982	0.00393	0.0158	0.0642	0.102	0.148	0.455
2	0.0100	0.0201	0.0404	0.0506	0.103	0.211	0.446	0.575	0.713	1.386
3	0.0717	0.115	0.185	0.216	0.352	0.584	1.005	1.213	1.424	2.366
4	0.207	0.297	0.429	0.484	0.711	1.064	1.649	1.923	2.195	3.357
5	0.412	0.554	0.752	0.831	1.145	1.610	2.343	2.675	3.000	4.351
6	0.676	0.872	1.134	1.237	1.635	2.204	3.070	3.455	3.828	5.348
7	0.989	1.239	1.564	1.690	2.167	2.833	3.822	4.255	4.671	6.346
8	1.344	1.647	2.032	2.180	2.733	3.490	4.594	5.071	5.527	7.344
9	1.735	2.088	2.532	2.700	3.325	4.168	5.380	5.899	6.393	8.343
10	2.156	2.558	3.059	3.247	3.940	4.865	6.179	6.737	7.267	9.342
11	2.603	3.053	3.609	3.816	4.575	5.578	6.989	7.584	8.148	10.341
12	3.074	3.571	4.178	4.404	5.226	6.304	7.807	8.438	9.034	11.340
13	3.565	4.107	4.765	5.009	5.892	7.041	8.634	9.299	9.926	12.340
14	4.075	4.660	5.368	5.629	6.571	7.790	9.467	10.165	10.821	13.339
15	4.601	5.229	5.985	6.262	7.261	8.547	10.307	11.037	11.721	14.339
16	5.142	5.812	6.614	6.908	7.962	9.312	11.152	11.912	12.624	15.338
17	5.697	6.408	7.255	7.564	8.672	10.085	12.002	12.792	13.531	16.338
18	6.265	7.015	7.906	8.231	9.390	10.865	12.857	13.675	14.440	17.338
19	6.844	7.633	8.567	8.907	10.117	11.651	13.716	14.562	15.352	18.338
20	7.434	8.260	9.237	9.591	10.851	12.443	14.578	15.452	16.266	19.337
21	8.034	8.897	9.915	10.283	11.591	13.240	15.445	16.344	17.182	20.337
22	8.643	9.542	10.600	10.982	12.338	14.041	16.314	17.240	18.101	21.337
23	9.260	10.196	11.293	11.689	13.091	14.848	17.187	18.137	19.021	22.337
24	9.886	10.856	11.992	12.401	13.848	15.659	18.062	19.037	19.943	23.337
25	10.520	11.524	12.697	13.120	14.611	16.473	18.940	19.939	20.867	24.337
26	11.160	12.198	13.409	13.844	15.379	17.292	19.820	20.843	21.792	25.336
27	11.808	12.878	14.125	14.573	16.151	18.114	20.703	21.749	22.719	26.336
28	12.461	13.565	14.847	15.308	16.928	18.939	21.588	22.657	23.647	27.336
29	13.121	14.256	15.574	16.047	17.708	19.768	22.475	23.567	24.577	28.336
30	13.787	14.953	16.306	16.791	18.493	20.599	23.364	24.478	25.508	29.336
40	20.707	22.164	23.838	24.433	26.509	29.051	32.345	33.66	34.872	39.335
50	27.991	29.707	31.664	32.357	34.764	37.689	41.449	42.942	44.313	49.335
60	35.534	37.485	39.699	40.482	43.188	46.459	50.641	52.294	53.809	59.335

부록#3

• 카이제곱분포표 (2/2)

v	α									
	0.30	0.25	0.20	0.10	0.05	0.025	0.02	0.01	0.005	0.001
1	1.074	1.323	1.642	2.706	3.841	5.024	5.412	6.635	7.879	10.827
2	2.408	2.773	3.219	4.605	5.991	7.378	7.824	9.210	10.597	13.815
3	3.665	4.108	4.642	6.251	7.815	9.348	9.837	11.345	12.838	16.266
4	4.878	5.385	5.989	7.779	9.488	11.143	11.668	13.277	14.860	18.466
5	6.064	6.626	7.289	9.236	11.070	12.832	13.388	15.086	16.750	20.515
6	7.231	7.841	8.558	10.645	12.592	14.449	15.033	16.812	18.548	22.457
7	8.383	9.037	9.803	12.017	14.067	16.013	16.622	18.475	20.278	24.321
8	9.524	10.219	11.030	13.362	15.507	17.535	18.168	20.090	21.955	26.124
9	10.656	11.389	12.242	14.684	16.919	19.023	19.679	21.666	23.589	27.877
10	11.781	12.549	13.442	15.987	18.307	20.483	21.161	23.209	25.188	29.588
11	12.899	13.701	14.631	17.275	19.675	21.920	22.618	24.725	26.757	31.264
12	14.011	14.845	15.812	18.549	21.026	23.337	24.054	26.217	28.300	32.909
13	15.119	15.984	16.985	19.812	22.362	24.736	25.471	27.688	29.819	34.527
14	16.222	17.117	18.151	21.064	23.685	26.119	26.873	29.141	31.319	36.124
15	17.322	18.245	19.311	22.307	24.996	27.488	28.259	30.578	32.801	37.698
16	18.418	19.369	20.465	23.542	26.296	28.845	29.633	32.000	34.267	39.252
17	19.511	20.489	21.615	24.769	27.587	30.191	30.995	33.409	35.718	40.791
18	20.601	21.605	22.760	25.989	28.869	31.526	32.346	34.805	37.156	42.312
19	21.689	22.718	23.900	27.204	30.144	32.852	33.687	36.191	38.582	43.819
20	22.775	23.828	25.038	28.412	31.410	34.170	35.020	37.566	39.997	45.314
21	23.858	24.935	26.171	29.615	32.671	35.479	36.343	38.932	41.401	46.796
22	24.939	26.039	27.301	30.813	33.924	36.781	37.659	40.289	42.796	48.268
23	26.018	27.141	28.429	32.007	35.172	38.076	38.968	41.638	44.181	49.728
24	27.096	28.241	29.553	33.196	36.415	39.364	40.270	42.980	45.558	51.179
25	28.172	29.339	30.675	34.382	37.652	40.646	41.566	44.314	46.928	52.619
26	29.246	30.435	31.795	35.563	38.885	41.923	42.856	45.642	48.290	54.051
27	30.319	31.528	32.912	36.741	40.113	43.195	44.140	46.963	49.645	55.475
28	31.391	32.620	34.027	37.916	41.337	44.461	45.419	48.278	50.994	56.892
29	32.461	33.711	35.139	39.087	42.557	45.722	46.693	49.588	52.335	58.301
30	33.530	34.800	36.250	40.256	43.773	46.979	47.962	50.892	53.672	59.702
40	44.165	45.616	47.269	51.805	55.758	59.342	60.436	63.691	66.766	73.403
50	54.723	56.334	58.164	63.167	67.505	71.420	72.613	76.154	79.490	86.660
60	65.226	66.981	68.972	74.397	79.082	83.298	84.58	88.379	91.952	99.608

부록#4

• 그래프 코드

• <그림 1> 균일분포의 확률 밀도 함수

```
1. import numpy as np
2. import matplotlib.pyplot as plt
3.
4. # 구간 설정
5. x = np.linspace(1, 3, 400)
6. y = np.full_like(x, 0.5)
7.
8. plt.figure(figsize=(6, 3))
9.
10. # 밀도함수 (실선)
11. plt.plot(x, y, color="deepskyblue", linewidth=2)
12.
13. # 점선 (x=1, x=3)
14. plt.vlines([1, 3], ymin=0, ymax=0.5,
15.           colors="deepskyblue",
16.           linestyle="dashed")
17.
18. # y = 1/2 점선
19. plt.hlines(0.5, xmin=0, xmax=1,
20.           colors="black", linestyle="dashed")
```

```
20. # 축 설정
21. plt.xlim(0, 3.5)
22. plt.ylim(0, 0.7)
23. plt.xlabel("x")
24. plt.ylabel("f(x)")
25. plt.xticks([0, 1, 3])
26. plt.yticks([0.5], ["1/2"])
27.
28. ax = plt.gca()
29. ax.spines["top"].set_visible(False)
30. ax.spines["right"].set_visible(False)
31.
32. plt.tight_layout()
33. plt.show()
```


부록#4

• 그래프 코드

• <그림 2> 예제 6.1의 확률밀도함수

```
1. import numpy as np
2. import matplotlib.pyplot as plt
3.
4. # 구간 설정 ( $0 \leq x \leq 4$ )
5. x = np.linspace(0, 4, 400)
6. y = np.full_like(x, 1/4)
7.
8. plt.figure(figsize=(6, 3))
9.
10. # 밀도함수 (실선)
11. plt.plot(x, y, color="deepskyblue", linewidth=2)
12.
13. # 점선 ( $x=0$ ,  $x=4$ )
14. plt.vlines([0, 4], ymin=0, ymax=1/4,
15.           colors="deepskyblue",
16.           linestyle="dashed")
```

```
16. # 축 설정
17. plt.xlim(-0.5, 4.5)
18. plt.ylim(0, 0.35)
19. plt.xlabel("x")
20. plt.ylabel("f(x)")
21.
22. # 눈금
23. plt.xticks([0, 4])
24. plt.yticks([1/4], ["1/4"])
25.
26. ax = plt.gca()
27. ax.spines["top"].set_visible(False)
28. ax.spines["right"].set_visible(False)
29.
30. plt.tight_layout()
31. plt.show()
```

부록#4

• 그래프 코드

• <그림 3> 정규분포의 확률밀도함수

```
1. import numpy as np
2. import matplotlib.pyplot as plt
3. import math
4.
5. mu = 0    # 정규분포 파라미터
6. sigma = 1
7.
8. # x 범위
9. x = np.linspace(mu - 4*sigma, mu + 4*sigma, 500)
10. # 정규분포 PDF
11. y = (1 / (math.sqrt(2 * math.pi) * sigma)) *
    np.exp(
12.     -0.5 * ((x - mu) / sigma) ** 2
13. )
14. plt.figure(figsize=(6, 3))
15.
16. # 정규분포 곡선
17. plt.plot(x, y, color="deepskyblue", linewidth=2)
18.
19. # 평균  $\mu$  점선
20. plt.vlines(mu, ymin=0, ymax=max(y),
21.     colors="black", linestyle="dashed")
```

```
22. plt.annotate(    #  $\sigma$  표시
23.     "",
24.     xy=(mu + sigma, max(y) * 0.6),
25.     xytext=(mu, max(y) * 0.6),
26.     arrowprops=dict(arrowstyle="<->")
27. )
28.
29. plt.text(mu + sigma / 2, max(y) * 0.63,
    r"$\sigma$", ha="center")
30. plt.text(mu, -max(y) * 0.04, r"$\mu$",
    ha="center", va="top")
31.
32. plt.xlabel("x", labelpad=15)
33. plt.yticks([])
34. plt.xticks([])
35.
36. ax = plt.gca()
37. ax.spines["left"].set_visible(False)
38. ax.spines["top"].set_visible(False)
39. ax.spines["right"].set_visible(False)
40.
41. plt.tight_layout()
42. plt.show()
```

부록#4

• 그래프 코드

• <그림 4> 평균과 표준편차에 따른 정규분포도 변화 (1/3)

```
1. import numpy as np
2. import matplotlib.pyplot as plt
3. import math
4.
5. def normal_pdf(x, mu, sigma):
6.     return (1 / (math.sqrt(2 * math.pi) *
7.         sigma)) * np.exp(
8.             -0.5 * ((x - mu) / sigma) ** 2
9.         )
10. # 공통 x 범위
11. x = np.linspace(-6, 6, 800)
12.
13. fig, axes = plt.subplots(1, 3, figsize=(12, 3.5))
```

```
14. # (1)  $\sigma_1 = \sigma_2$ ,  $\mu_1 \neq \mu_2$ 
15. mu1, mu2 = -2, 2
16. sigma = 1
17. axes[0].plot(x, normal_pdf(x, mu1, sigma),
18.     color="deepskyblue", linewidth=2)
19. axes[0].plot(x, normal_pdf(x, mu2, sigma),
20.     color="deepskyblue", linewidth=2)
21. axes[0].vlines([mu1, mu2], 0,
22.     normal_pdf(mu1, mu1, sigma),
23.     colors="black",
24.     linestyle="dashed")
25. axes[0].text(0, max(normal_pdf(x, mu1, sigma))
26.     * 0.7,
27.     r"$\sigma_1=\sigma_2$",
28.     ha="center")
29. axes[0].text(mu1, 0, r"$\mu_1$", ha="center",
30.     va="top")
31. axes[0].text(mu2, 0, r"$\mu_2$", ha="center",
32.     va="top")
```


부록#4

• 그래프 코드

• <그림 4> 평균과 표준편차에 따른 정규분포도 변화 (2/3)

```
26. # (2)  $\mu_1 = \mu_2$ ,  $\sigma_1 \neq \sigma_2$ 
27. mu = 0
28. sigma1, sigma2 = 0.7, 1.6
29. axes[1].plot(x, normal_pdf(x, mu, sigma1),
    color="deepskyblue", linewidth=2)
30. axes[1].plot(x, normal_pdf(x, mu, sigma2),
    color="deepskyblue", linewidth=2)
31. axes[1].vlines(mu, 0,
32.                 normal_pdf(mu, mu, sigma1),
33.                 colors="black",
    linestyle="dashed")
34. axes[1].text(mu + 0.3, max(normal_pdf(x, mu,
    sigma1)) * 0.85, r"\sigma_1$")
35. axes[1].text(mu + 2.0, max(normal_pdf(x, mu,
    sigma2)) * 0.4, r"\sigma_2$")
36. axes[1].text(mu, 0, r"\mu_1=\mu_2$",
    ha="center", va="top")
```

```
37. # (3)  $\mu_1 \neq \mu_2$ ,  $\sigma_1 \neq \sigma_2$ 
38. mu1, sigma1 = -2, 0.7
39. mu2, sigma2 = 2, 1.5
40. axes[2].plot(x, normal_pdf(x, mu1, sigma1),
    color="deepskyblue", linewidth=2)
41. axes[2].plot(x, normal_pdf(x, mu2, sigma2),
    color="deepskyblue", linewidth=2)
42. axes[2].vlines([mu1, mu2], 0,
43.                 [normal_pdf(mu1, mu1, sigma1),
44.                 normal_pdf(mu2, mu2, sigma2)],
45.                 colors="black",
    linestyle="dashed")
46. axes[2].text(mu1 + 0.2, max(normal_pdf(x, mu1,
    sigma1)) * 0.85, r"\sigma_1$")
47. axes[2].text(mu2 + 0.5, max(normal_pdf(x, mu2,
    sigma2)) * 0.5, r"\sigma_2$")
48. axes[2].text(mu1, 0, r"\mu_1$", ha="center",
    va="top")
49. axes[2].text(mu2, 0, r"\mu_2$", ha="center",
    va="top")
```

부록#4

- 그래프 코드

- <그림 4> 평균과 표준편차에 따른 정규분포도 변화 (3/3)

```
50. # 공통 스타일
51. for ax in axes:
52.     ax.set_xlim(-6, 6)
53.     ax.set_yticks([])
54.     ax.set_xticks([])
55.     for spine in ax.spines.values():
56.         spine.set_visible(True)
57.         spine.set_linewidth(1.0)
58.
59. plt.tight_layout()
60. plt.show()
```

부록#4

• 그래프 코드

• <그림 5> 기존의 정규분포와 변환된 표준정규분포 (1/3)

```
1. import numpy as np
2. import matplotlib.pyplot as plt
3. import math
4.
5. def normal_pdf(x, mu, sigma):
6.     return (1 / (math.sqrt(2 * math.pi) *
7.         sigma)) * np.exp(
8.         -0.5 * ((x - mu) / sigma) ** 2
9.     )
10. # 파라미터
11. mu = 0.0
12.
13. sigma_left = 1.0    # 왼쪽
14. sigma_right = 0.6   # 오른쪽
15.
16. # 음영 구간
17. x1_left, x2_left = -1.4, 0.7
18. x1_right, x2_right = -0.8, 0.4
19.
20. # x 범위 (공통)
21. x = np.linspace(-4, 4, 3000)
```

```
22. y_left = normal_pdf(x, mu, sigma_left)
23. y_right = normal_pdf(x, mu, sigma_right)
24.
25. # y축 공통 기준
26. y_max = max(y_left.max(), y_right.max()) * 1.1
27.
28. fig, axes = plt.subplots(1, 2, figsize=(9, 3.5))
29. eps = 0 # 음영 크기 보정
30.
31. # ----- 왼쪽 -----
32. ax = axes[0]
33. ax.plot(x, y_left, color="deepskyblue",
34.         linewidth=2)
35. mask = (x >= x1_left - eps) & (x <= x2_left +
36.     eps)
37. ax.fill_between(x[mask], y_left[mask],
38.                 color="deepskyblue", alpha=0.18)
```

부록#4

• 그래프 코드

• <그림 5> 기존의 정규분포와 변환된 표준정규분포 (2/3)

```
38. # 수직 점선
39. ax.vlines(
40.     [x1_left, x2_left], [0, 0],
41.     [normal_pdf(x1_left, mu, sigma_left),
42.      normal_pdf(x2_left, mu, sigma_left)],
43.     colors="deepskyblue", linestyle="dashed",
44.     linewidth=1.4
45. )
46. ax.vlines(mu, 0, normal_pdf(mu, mu, sigma_left),
47.           colors="deepskyblue",
48.           linestyle="dashed", linewidth=1.4)
49. ax.text(x1_left, -0.02, r"$x_1$", ha="center",
50.         va="top")
51. ax.text(x2_left, -0.02, r"$x_2$", ha="center",
52.         va="top")
53. ax.text(mu, -0.02, r"$\mu$", ha="center",
54.         va="top")
55. ax.text(1.8, y_left.max()*0.6, r"$\sigma$",
56.         fontsize=11)
```

```
57. ax.set_xlabel("x", labelpad=15)
58. ax.set_xlim(-4, 4)
59. ax.set_ylim(0, y_max)
60. ax.set_yticks([])
61. ax.set_xticks([])
62. # ----- 오른쪽 -----
63. ax = axes[1]
64. ax.plot(x, y_right, color="deepskyblue",
65.         linewidth=2)
66. mask = (x >= x1_right - eps) & (x <= x2_right +
67.     eps)
68. ax.fill_between(x[mask], y_right[mask],
69.                 color="deepskyblue", alpha=0.18)
```

부록#4

• 그래프 코드

• <그림 5> 기존의 정규분포와 변환된 표준정규분포 (3/3)

```
65. ax.vlines(  
66.     [x1_right, x2_right], [0, 0],  
67.     [normal_pdf(x1_right, mu, sigma_right),  
68.       normal_pdf(x2_right, mu, sigma_right)],  
69.     colors="deepskyblue", linestyle="dashed",  
       linewidth=1.4  
70. )  
71. ax.vlines(mu, 0, normal_pdf(mu, mu,  
       sigma_right),  
72.           colors="deepskyblue",  
       linestyle="dashed", linewidth=1.4)  
73.  
74. ax.text(x1_right, -0.02, r"$z_1$", ha="center",  
       va="top")  
75. ax.text(x2_right, -0.02, r"$z_2$", ha="center",  
       va="top")  
76. ax.text(mu, -0.02, r"$\mu$", ha="center",  
       va="top")  
77. ax.text(0.9, y_right.max()*0.7, r"$\sigma=1$",  
       fontsize=11)
```

```
78. ax.set_xlabel("z", labelpad=15)  
79. ax.set_xlim(-4, 4)  
80. ax.set_ylim(0, y_max)  
81. ax.set_yticks([])  
82. ax.set_xticks([])  
83.  
84. # ----- 공통 스타일 -----  
85. for ax in axes:  
86.     ax.spines["left"].set_visible(False)  
87.     ax.spines["top"].set_visible(False)  
88.     ax.spines["right"].set_visible(False)  
89.  
90. plt.tight_layout()  
91. plt.show()
```

부록#4

• 그래프 코드

• <그림 7> 예제 6.2의 표준정규분포도 (1/3)

```
1. import numpy as np
2. import matplotlib.pyplot as plt
3. import math
4.
5. def normal_pdf(x, mu=0, sigma=1):
6.     return (1 / (math.sqrt(2 * math.pi) *
7.         sigma)) * np.exp(
8.             -0.5 * ((x - mu) / sigma) ** 2
9.         )
10. # 공통 설정
11. z = np.linspace(-4, 4, 3000)
12. y = normal_pdf(z)
13.
14. fig, axes = plt.subplots(1, 2, figsize=(9, 3.5))
15.
16. # (a) 오른쪽 꼬리:  $P(Z \geq 1.84)$ 
17. z_a = 1.84
18. y_za = normal_pdf(z_a)
```

```
19. ax = axes[0]
20. ax.plot(z, y, color="deepskyblue", linewidth=2)
21. # 음영 (오른쪽 꼬리)
22. mask = z >= z_a
23. ax.fill_between(z[mask], y[mask],
24.                 color="deepskyblue", alpha=0.18)
25. # 수직 점선 (0) + 경계선(1.84)
26. ax.vlines(0, 0, normal_pdf(0),
27.           colors="deepskyblue",
28.           linestyle="dashed", linewidth=1.4)
29. ax.vlines(z_a, 0, y_za,
30.           colors="deepskyblue", linewidth=1.4)
31. ax.text(0, -0.02, r"$0$", ha="center", va="top")
32. ax.text(z_a, -0.02, r"$1.84$", ha="center",
33.         va="top")
```

부록#4

• 그래프 코드

• <그림 7> 예제 6.2의 표준정규분포도 (2/3)

```
33. ax.set_xlabel("z", labelpad=15)
34. ax.set_xlim(-4, 4)
35. ax.set_ylim(0, y.max() * 1.1)
36. ax.set_yticks([])
37. ax.set_xticks([])
38.
39. ax.text(0.5, -0.25, "(a)",
    transform=ax.transAxes,
40.         ha="center", va="top", fontsize=12)
41.
42. # (b) 구간:  $P(-1.97 \leq Z \leq 0.86)$ 
43. z1, z2 = -1.97, 0.86
44. y_z1 = normal_pdf(z1)
45. y_z2 = normal_pdf(z2)
```

```
46. ax = axes[1]
47. ax.plot(z, y, color="deepskyblue", linewidth=2)
48. # 음영 (구간)
49. mask = (z >= z1) & (z <= z2)
50. ax.fill_between(z[mask], y[mask],
51.                 color="deepskyblue", alpha=0.18)
52. # 수직 점선 (0) + 경계선(z1, z2)
53. ax.vlines(0, 0, normal_pdf(0),
54.           colors="deepskyblue",
55.           linestyle="dashed", linewidth=1.4)
56. ax.vlines([z1, z2], [0, 0], [y_z1, y_z2],
57.           colors="deepskyblue", linewidth=1.4)
58. ax.text(z1, -0.02, r"$-1.97$", ha="center",
59.         va="top")
59. ax.text(0, -0.02, r"$0$", ha="center", va="top")
60. ax.text(z2, -0.02, r"$0.86$", ha="center",
61.         va="top")
```

부록#4

- 그래프 코드

- <그림 7> 예제 6.2의 표준정규분포도 (3/3)

```
61. ax.set_xlabel("z", labelpad=15)
62. ax.set_xlim(-4, 4)
63. ax.set_ylim(0, y.max() * 1.1)
64. ax.set_yticks([])
65. ax.set_xticks([])
66.
67. ax.text(0.5, -0.25, "(b)",
68.         transform=ax.transAxes,
69.         ha="center", va="top", fontsize=12)
70. # 공통 스타일
71. for ax in axes:
72.     ax.spines["left"].set_visible(False)
73.     ax.spines["top"].set_visible(False)
74.     ax.spines["right"].set_visible(False)
75.
76. plt.tight_layout()
77. plt.show()
```


부록#4

• 그래프 코드

• <그림 8> 예제 6.3의 표준정규분포도 (1/3)

```
1. import numpy as np
2. import matplotlib.pyplot as plt
3. import math
4.
5. def normal_pdf(x, mu=0, sigma=1):
6.     return (1 / (math.sqrt(2 * math.pi) *
7.         sigma)) * np.exp(
8.         -0.5 * ((x - mu) / sigma) ** 2
9.     )
10. # 공통 설정
11. x = np.linspace(-4, 4, 3000)
12. y = normal_pdf(x)
13. fig, axes = plt.subplots(1, 2, figsize=(9, 3.5))
14.
15. # (a) 오른쪽 꼬리:  $P(X \geq k) = 0.3015$ 
16. k_a = 0.52
17. prob_a = 0.3015
18. ax = axes[0]
19. ax.plot(x, y, color="deepskyblue", linewidth=2)
```

```
20. # 음영
21. mask = x >= k_a
22. ax.fill_between(x[mask], y[mask],
23.                 color="deepskyblue", alpha=0.18)
24. # 수직선
25. ax.vlines(0, 0, normal_pdf(0),
26.           colors="deepskyblue",
27.           linestyle="dashed", linewidth=1.4)
28. ax.vlines(k_a, 0, normal_pdf(k_a),
29.           colors="deepskyblue", linewidth=1.4)
30. ax.text(0, -0.02, r"$0$", ha="center", va="top")
31. ax.text(k_a, -0.02, r"$k$", ha="center",
32.         va="top")
32. ax.text(k_a + 0.35, normal_pdf(k_a) * 0.35,
33.         f"{prob_a}", fontsize=11)
```

부록#4

• 그래프 코드

• <그림 8> 예제 6.3의 표준정규분포도 (2/3)

```
33. ax.set_xlabel("z", labelpad=15)
34. ax.set_xlim(-4, 4)
35. ax.set_ylim(0, y.max() * 1.1)
36. ax.set_yticks([])
37. ax.set_xticks([])
38.
39. ax.text(0.5, -0.25, "(a)",
    transform=ax.transAxes,
40.         ha="center", va="top", fontsize=12)
41.
42. # (b) 구간:  $P(k \leq X \leq -0.18) = 0.4197$ 
43. k_b = -1.5
44. z_b = -0.18
45. prob_b = 0.4197
46. ax = axes[1]
47. ax.plot(x, y, color="deepskyblue", linewidth=2)
48. # 음영
49. mask = (x >= k_b) & (x <= z_b)
50. ax.fill_between(x[mask], y[mask],
51.                 color="deepskyblue", alpha=0.18)
```

```
52. # 수직선
53. ax.vlines(0, 0, normal_pdf(0),
54.           colors="deepskyblue",
55.           linestyle="dashed", linewidth=1.4)
56. ax.vlines([k_b, z_b], [0, 0],
57.           [normal_pdf(k_b), normal_pdf(z_b)],
58.           colors="deepskyblue", linewidth=1.4)
59. ax.text(k_b, -0.02, r"$k$", ha="center",
    va="top")
60. ax.text(z_b, -0.02, r"$-0.18$", ha="center",
    va="top")
61. ax.text(-0.9, normal_pdf(-0.9) * 0.45,
    f"{prob_b}", fontsize=11)
62.
63. ax.set_xlabel("z", labelpad=15)
64. ax.set_xlim(-4, 4)
65. ax.set_ylim(0, y.max() * 1.1)
66. ax.set_yticks([])
67. ax.set_xticks([])
```

부록#4

- 그래프 코드

- <그림 8> 예제 6.3의 표준정규분포도 (3/3)

```
68. ax.text(0.5, -0.25, "(b)",  
    transform=ax.transAxes,  
69.         ha="center", va="top", fontsize=12)  
70.  
71. # 공통 스타일  
72. for ax in axes:  
73.     ax.spines["left"].set_visible(False)  
74.     ax.spines["top"].set_visible(False)  
75.     ax.spines["right"].set_visible(False)  
76.  
77. plt.tight_layout()  
78. plt.show()
```

부록#4

• 그래프 코드

• <그림 9> 예제 6.4의 표준정규분포도 (1/2)

```
1. import numpy as np
2. import matplotlib.pyplot as plt
3. import math
4.
5. def normal_pdf(x, mu=0, sigma=1):
6.     return (1 / (math.sqrt(2 * math.pi) *
7.         sigma)) * np.exp(
8.         -0.5 * ((x - mu) / sigma) ** 2
9.     )
10. # 설정
11. x = np.linspace(-4, 4, 3000)
12. y = normal_pdf(x)
13. x1 = -0.5
14. x2 = 1.2
15. y_x1 = normal_pdf(x1)
16. y_x2 = normal_pdf(x2)
17. y_0 = normal_pdf(0)
18. plt.figure(figsize=(6.5, 3))
19. # 정규분포 곡선
20. plt.plot(x, y, color="deepskyblue", linewidth=2)
```

```
21. # 음영 구간 [-0.5, 1.2]
22. mask = (x >= x1) & (x <= x2)
23. plt.fill_between(x[mask], y[mask],
24.     color="deepskyblue",
25.     alpha=0.18)
26. # 수직선
27. plt.vlines(x1, 0, y_x1,
28.     colors="deepskyblue", linewidth=1.4)
29. plt.vlines(x2, 0, y_x2,
30.     colors="deepskyblue", linewidth=1.4)
31. # 평균 0 (점선)
32. plt.vlines(0, 0, y_0,
33.     colors="deepskyblue",
34.     linestyle="dashed", linewidth=1.4)
35. # 텍스트
36. plt.text(x1, -0.02, r"$-0.5$", ha="center",
37.     va="top")
38. plt.text(0, -0.02, r"$0$", ha="center",
39.     va="top")
40. plt.text(x2, -0.02, r"$1.2$", ha="center",
41.     va="top")
```

부록#4

- 그래프 코드

- <그림 9> 예제 6.4의 표준정규분포도 (2/2)

```
37. # 축 설정
38. plt.xlabel("z", labelpad=15)
39. plt.xlim(-4, 4)
40. plt.ylim(0, y.max() * 1.1)
41. plt.yticks([])
42. plt.xticks([])
43.
44. ax = plt.gca()
45. ax.spines["left"].set_visible(False)
46. ax.spines["top"].set_visible(False)
47. ax.spines["right"].set_visible(False)
48.
49. plt.tight_layout()
50. plt.show()
```

부록#4

• 그래프 코드

• <그림 10> 예제 6.5의 확률밀도함수도 (1/2)

```
1. import numpy as np
2. import matplotlib.pyplot as plt
3. import math
4.
5. def normal_pdf(x, mu, sigma):
6.     return (1 / (math.sqrt(2 * math.pi) *
7.         sigma)) * np.exp(
8.         -0.5 * ((x - mu) / sigma) ** 2
9.     )
10. # 파라미터
11. mu = 300
12. sigma = 50
13. # 기준값
14. x0 = 362
15. # x 범위
16. x = np.linspace(mu - 4*sigma, mu + 4*sigma,
17.     4000)
18. y = normal_pdf(x, mu, sigma)
19. y_mu = normal_pdf(mu, mu, sigma)
20. y_x0 = normal_pdf(x0, mu, sigma)
21. plt.figure(figsize=(6.5, 3))
```

```
22. # 정규분포 곡선
23. plt.plot(x, y, color="deepskyblue", linewidth=2)
24.
25. # 음영
26. mask = x >= x0
27. plt.fill_between(x[mask], y[mask],
28.     color="deepskyblue",
29.     alpha=0.18)
30. # 수직선
31. plt.vlines(mu, 0, y_mu,
32.     colors="deepskyblue",
33.     linestyle="dashed", linewidth=1.4)
34. plt.vlines(x0, 0, y_x0,
35.     colors="deepskyblue", linewidth=1.4)
36. # 텍스트
37. plt.text(mu, -y.max()*0.06, r"$300$",
38.     ha="center", va="top")
39. plt.text(x0, -y.max()*0.06, r"$362$",
40.     ha="center", va="top")
41. plt.text(mu + sigma*0.9, y.max()*0.65,
42.     r"$\sigma = 50$", fontsize=11)
```

부록#4

- 그래프 코드

- <그림 10> 예제 6.5의 확률 밀도 함수도 (2/2)

```
38. # 축 설정
39. plt.xlabel("x", labelpad=15)
40. plt.xlim(mu - 4*sigma, mu + 4*sigma)
41. plt.ylim(0, y.max()*1.1)
42. plt.yticks([])
43. plt.xticks([])
44. ax = plt.gca()
45. ax.spines["left"].set_visible(False)
46. ax.spines["top"].set_visible(False)
47. ax.spines["right"].set_visible(False)
48.
49. plt.tight_layout()
50. plt.show()
```

부록#4

• 그래프 코드

• <그림 11> 예제 6.6의 확률밀도함수도 (1/3)

```
1. import numpy as np
2. import matplotlib.pyplot as plt
3. import math
4.
5. def normal_pdf(x, mu, sigma):
6.     return (1 / (math.sqrt(2 * math.pi) *
7.         sigma)) * np.exp(
8.         -0.5 * ((x - mu) / sigma) ** 2
9.     )
10. # 파라미터
11. mu = 40
12. sigma = 6
13. # 꼬리 기준값
14. k_left = 38.5 # (a) 왼쪽 꼬리
15. k_right = 43.5 # (b) 오른쪽 꼬리
16.
17. # x 범위
18. x = np.linspace(mu - 4*sigma, mu + 4*sigma,
19.     4000)
```

```
20. y_mu = normal_pdf(mu, mu, sigma)
21. y_kl = normal_pdf(k_left, mu, sigma)
22. y_kr = normal_pdf(k_right, mu, sigma)
23. fig, axes = plt.subplots(1, 2, figsize=(9, 3.5))
24. # (a) 왼쪽 꼬리
25. ax = axes[0]
26. ax.plot(x, y, color="deepskyblue", linewidth=2)
27. # 음영: 왼쪽 꼬리
28. mask = x <= k_left
29. ax.fill_between(x[mask], y[mask],
30.     color="deepskyblue", alpha=0.18)
31. # 수직선
32. ax.vlines(mu, 0, y_mu,
33.     colors="deepskyblue",
34.     linestyle="dashed", linewidth=1.4)
35. ax.vlines(k_left, 0, y_kl,
36.     colors="deepskyblue", linewidth=1.4)
37. ax.text(mu, -y.max()*0.06, r"$\mu$", ha="center",
38.     va="top")
39. ax.text(mu + sigma*0.6, y.max()*0.7, r"$\sigma$
40.     = 6$", fontsize=11)
41. ax.text(mu - sigma*1.1, y.max()*0.05, r"$0.45\sigma$",
42.     fontsize=11)
```


부록#4

• 그래프 코드

• <그림 11> 예제 6.6의 확률밀도함수도 (2/3)

```
39. ax.set_xlabel("x", labelpad=15)
40. ax.set_xlim(mu - 4*sigma, mu + 4*sigma)
41. ax.set_ylim(0, y.max()*1.1)
42. ax.set_yticks([])
43. ax.set_xticks([])
44.
45. ax.text(0.5, -0.25, "(a)",
    transform=ax.transAxes,
46.         ha="center", va="top", fontsize=12)
47.
48. # (b) 오른쪽 꼬리
49. ax = axes[1]
50. ax.plot(x, y, color="deepskyblue", linewidth=2)
51. # 음영: 오른쪽 꼬리
52. mask = x >= k_right
53. ax.fill_between(x[mask], y[mask],
54.                 color="deepskyblue", alpha=0.18)
```

```
55. # 수직선
56. ax.vlines(mu, 0, y_mu,
57.           colors="deepskyblue",
58.           linestyle="dashed", linewidth=1.4)
59. ax.vlines(k_right, 0, y_kr,
60.           colors="deepskyblue", linewidth=1.4)
61. ax.text(mu, -y.max()*0.06, r"$\mu$", ha="center",
62.          va="top")
63. ax.text(mu + sigma*0.6, y.max()*0.7, r"$\sigma = 6$",
64.          fontsize=11)
65. ax.text(mu + sigma*0.9, y.max()*0.05, r"$0.14$",
66.          fontsize=11)
67.
68. ax.set_xlabel("x", labelpad=15)
69. ax.set_xlim(mu - 4*sigma, mu + 4*sigma)
70. ax.set_ylim(0, y.max()*1.1)
71. ax.set_yticks([])
72. ax.set_xticks([])
73.
74. ax.text(0.5, -0.25, "(b)",
    transform=ax.transAxes,
75.         ha="center", va="top", fontsize=12)
```

부록#4

- 그래프 코드

- <그림 11> 예제 6.6의 확률 밀도 함수도 (3/3)

```
72. # 공통 스타일
73. for ax in axes:
74.     ax.spines["left"].set_visible(False)
75.     ax.spines["top"].set_visible(False)
76.     ax.spines["right"].set_visible(False)
77.
78. plt.tight_layout()
79. plt.show()
```

부록#4

• 그래프 코드

• <그림 12> 예제 6.7의 확률밀도함수도 (1/2)

```
1. import numpy as np
2. import matplotlib.pyplot as plt
3. import math
4.
5. def normal_pdf(x, mu, sigma):
6.     return (1 / (math.sqrt(2 * math.pi) *
7.         sigma)) * np.exp(
8.             -0.5 * ((x - mu) / sigma) ** 2
9.         )
10. # 파라미터
11. mu = 3
12. sigma = 0.5
13. # 기준값
14. x0 = 2.3
15. # x 범위
16. x = np.linspace(mu - 4*sigma, mu + 4*sigma,
17.     4000)
18. y = normal_pdf(x, mu, sigma)
19. y_mu = normal_pdf(mu, mu, sigma)
20. y_x0 = normal_pdf(x0, mu, sigma)
21. plt.figure(figsize=(6.5, 3))
```

```
22. # 정규분포 곡선
23. plt.plot(x, y, color="deepskyblue", linewidth=2)
24. # 음영: 2.3 이하
25. mask = x <= x0
26. plt.fill_between(x[mask], y[mask],
27.     color="deepskyblue",
28.     alpha=0.18)
29. # 수직선
30. plt.vlines(mu, 0, y_mu,
31.     colors="deepskyblue",
32.     linestyle="dashed", linewidth=1.4)
33. plt.vlines(x0, 0, y_x0,
34.     colors="deepskyblue", linewidth=1.4)
35. plt.text(mu, -y.max()*0.06, r"$\mu$", ha="center",
36.     va="top")
37. plt.text(x0, -y.max()*0.06, r"$x_0$", ha="center", va="top")
38. plt.text(mu + sigma*0.9, y.max()*0.65,
39.     r"$\sigma = 0.5$", fontsize=11)
```

부록#4

- 그래프 코드

- <그림 12> 예제 6.7의 확률밀도함수도 (2/2)

```
36. # 축 설정
37. plt.xlabel("x", labelpad=15)
38. plt.xlim(mu - 4*sigma, mu + 4*sigma)
39. plt.ylim(0, y.max()*1.1)
40. plt.yticks([])
41. plt.xticks([])
42. ax = plt.gca()
43. ax.spines["left"].set_visible(False)
44. ax.spines["top"].set_visible(False)
45. ax.spines["right"].set_visible(False)
46.
47. plt.tight_layout()
48. plt.show()
```

부록#4

• 그래프 코드

• <그림 13> 예제 6.8의 확률밀도함수도 (1/2)

```
1. import numpy as np
2. import matplotlib.pyplot as plt
3. import math
4. def normal_pdf(x, mu, sigma):
5.     return (1 / (math.sqrt(2 * math.pi) *
6.         sigma)) * np.exp(
7.         -0.5 * ((x - mu) / sigma) ** 2
8.     )
9. # 파라미터
10. mu = 800
11. sigma = 40
12. # 구간
13. x1 = 778
14. x2 = 834
15. # x 범위
16. x = np.linspace(mu - 4*sigma, mu + 4*sigma,
17.     4000)
17. y = normal_pdf(x, mu, sigma)
18. y_mu = normal_pdf(mu, mu, sigma)
19. y_x1 = normal_pdf(x1, mu, sigma)
20. y_x2 = normal_pdf(x2, mu, sigma)
21. plt.figure(figsize=(6.5, 3))
```

```
22. # 정규분포 곡선
23. plt.plot(x, y, color="deepskyblue", linewidth=2)
24. # 음영: [778, 834]
25. mask = (x >= x1) & (x <= x2)
26. plt.fill_between(x[mask], y[mask],
27.     color="deepskyblue",
28.     alpha=0.18)
29. # 수직선
30. plt.vlines(x1, 0, y_x1,
31.     colors="deepskyblue", linewidth=1.4)
32. plt.vlines(x2, 0, y_x2,
33.     colors="deepskyblue", linewidth=1.4)
34. # 평균 (점선)
35. plt.vlines(mu, 0, y_mu,
36.     colors="deepskyblue",
37.     linestyle="dashed", linewidth=1.4)
38. plt.text(x1, -y.max()*0.06, r"$778$",
39.     ha="center", va="top")
40. plt.text(mu, -y.max()*0.06, r"$800$",
41.     ha="center", va="top")
42. plt.text(x2, -y.max()*0.06, r"$834$",
43.     ha="center", va="top")
44. plt.text(mu + sigma*0.9, y.max()*0.65,
45.     r"$\sigma = 40$", fontsize=11)
```

부록#4

- 그래프 코드

- <그림 13> 예제 6.8의 확률 밀도 함수도 (2/2)

```
40. # 축 설정
41. plt.xlabel("x", labelpad=15)
42. plt.xlim(mu - 4*sigma, mu + 4*sigma)
43. plt.ylim(0, y.max()*1.1)
44. plt.yticks([])
45. plt.xticks([])
46. ax = plt.gca()
47. ax.spines["left"].set_visible(False)
48. ax.spines["top"].set_visible(False)
49. ax.spines["right"].set_visible(False)
50.
51. plt.tight_layout()
52. plt.show()
```

부록#4

• 그래프 코드

• <그림 14> 예제 6.9의 확률밀도함수도 (1/2)

```
1. import numpy as np
2. import matplotlib.pyplot as plt
3. import math
4. def normal_pdf(x, mu, sigma):
5.     return (1 / (math.sqrt(2 * math.pi) *
6.         sigma)) * np.exp(
7.         -0.5 * ((x - mu) / sigma) ** 2
8.     )
9. # 파라미터
10. mu = 3.0
11. sigma = 0.005
12. # 기준값
13. x_left = 2.99
14. x_right = 3.01
15. # x 범위
16. x = np.linspace(mu - 4*sigma, mu + 4*sigma,
17.     4000)
18. y = normal_pdf(x, mu, sigma)
19. y_mu = normal_pdf(mu, mu, sigma)
20. y_l = normal_pdf(x_left, mu, sigma)
21. y_r = normal_pdf(x_right, mu, sigma)
22. plt.figure(figsize=(6.5, 3))
```

```
23. # 정규분포 곡선
24. plt.plot(x, y, color="deepskyblue", linewidth=2)
25. # 음영: 왼쪽 꼬리
26. mask_left = x <= x_left
27. plt.fill_between(x[mask_left], y[mask_left],
28.     color="deepskyblue",
29.     alpha=0.18)
30. # 음영: 오른쪽 꼬리
31. mask_right = x >= x_right
32. plt.fill_between(x[mask_right], y[mask_right],
33.     color="deepskyblue",
34.     alpha=0.18)
35. # 수직선
36. plt.vlines(mu, 0, y_mu,
37.     colors="deepskyblue",
38.     linestyle="dashed", linewidth=1.4)
39. plt.vlines(x_left, 0, y_l,
40.     colors="deepskyblue", linewidth=1.4)
41. plt.vlines(x_right, 0, y_r,
42.     colors="deepskyblue", linewidth=1.4)
```

부록#4

• 그래프 코드

• <그림 14> 예제 6.9의 확률밀도함수도 (2/2)

```
39. plt.text(x_left, -y.max()*0.08, r"$2.99$",  
    ha="center", va="top")  
40. plt.text(mu, -y.max()*0.08, r"$3.0$",  
    ha="center", va="top")  
41. plt.text(x_right, -y.max()*0.08, r"$3.01$",  
    ha="center", va="top")  
42. plt.text(x_left - sigma*1.2, y.max()*0.12,  
    r"$0.0228$", fontsize=11)  
43. plt.text(x_right + sigma*0.6, y.max()*0.12,  
    r"$0.0228$", fontsize=11)  
44. plt.text(mu + sigma*1.2, y.max()*0.7, r"$\sigma$  
    = 0.005$", fontsize=11)
```

```
45. # 축 설정  
46. plt.xlabel("x", labelpad=22)  
47. plt.xlim(mu - 4*sigma, mu + 4*sigma)  
48. plt.ylim(0, y.max()*1.15)  
49. plt.yticks([])  
50. plt.xticks([])  
51. ax = plt.gca()  
52. ax.spines["left"].set_visible(False)  
53. ax.spines["top"].set_visible(False)  
54. ax.spines["right"].set_visible(False)  
55.  
56. plt.tight_layout()  
57. plt.show()
```


부록#4

• 그래프 코드

• <그림 15> 예제 6.10의 확률밀도함수도 (1/2)

```
1. import numpy as np
2. import matplotlib.pyplot as plt
3. import math
4. def normal_pdf(x, mu, sigma):
5.     return (1 / (math.sqrt(2 * math.pi) *
6.         sigma)) * np.exp(
7.         -0.5 * ((x - mu) / sigma) ** 2
8.     )
9. # 파라미터
10. mu = 1.5
11. sigma = 0.2
12. # 경계값
13. x_left = 1.108
14. x_right = 1.892
15. # x 범위
16. x = np.linspace(mu - 4*sigma, mu + 4*sigma,
17.     4000)
18. y = normal_pdf(x, mu, sigma)
19. y_mu = normal_pdf(mu, mu, sigma)
20. y_l = normal_pdf(x_left, mu, sigma)
21. y_r = normal_pdf(x_right, mu, sigma)
22. plt.figure(figsize=(6.5, 3))
```

```
23. # 정규분포 곡선
24. plt.plot(x, y, color="deepskyblue", linewidth=2)
25. # 음영: 왼쪽 꼬리
26. mask_left = x <= x_left
27. plt.fill_between(x[mask_left], y[mask_left],
28.     color="deepskyblue",
29.     alpha=0.18)
30. # 음영: 오른쪽 꼬리
31. mask_right = x >= x_right
32. plt.fill_between(x[mask_right], y[mask_right],
33.     color="deepskyblue",
34.     alpha=0.18)
35. # 수직선
36. plt.vlines(mu, 0, y_mu,
37.     colors="deepskyblue",
38.     linestyle="dashed", linewidth=1.4)
39. plt.vlines(x_left, 0, y_l,
40.     colors="deepskyblue", linewidth=1.4)
41. plt.vlines(x_right, 0, y_r,
42.     colors="deepskyblue", linewidth=1.4)
```

부록#4

• 그래프 코드

• <그림 15> 예제 6.10의 확률밀도함수도 (2/2)

```
39. plt.text(x_left, -y.max()*0.08, r"$1.108$",  
    ha="center", va="top")  
40. plt.text(mu, -y.max()*0.08, r"$1.500$",  
    ha="center", va="top")  
41. plt.text(x_right, -y.max()*0.08, r"$1.892$",  
    ha="center", va="top")  
42. plt.text(x_left - sigma*0.9, y.max()*0.12,  
    r"$0.025$", fontsize=11)  
43. plt.text(x_right + sigma*0.4, y.max()*0.12,  
    r"$0.025$", fontsize=11)  
44. plt.text(mu + sigma*1.1, y.max()*0.7, r"$\sigma$  
    = 0.2$", fontsize=11)
```

```
45. # 축 설정  
46. plt.xlabel("x", labelpad=22)  
47. plt.xlim(mu - 4*sigma, mu + 4*sigma)  
48. plt.ylim(0, y.max()*1.15)  
49. plt.yticks([])  
50. plt.xticks([])  
51. ax = plt.gca()  
52. ax.spines["left"].set_visible(False)  
53. ax.spines["top"].set_visible(False)  
54. ax.spines["right"].set_visible(False)  
55.  
56. plt.tight_layout()  
57. plt.show()
```

부록#4

• 그래프 코드

• <그림 16> 예제 6.11의 확률밀도함수도 (1/2)

```
1. import numpy as np
2. import matplotlib.pyplot as plt
3. import math
4.
5. def normal_pdf(x, mu, sigma):
6.     return (1 / (math.sqrt(2 * math.pi) *
7.         sigma)) * np.exp(
8.             -0.5 * ((x - mu) / sigma) ** 2
9.         )
10. # 파라미터
11. mu = 40
12. sigma = 2.0
13. # 기준값
14. x0 = 43
15. # x 범위
16. x = np.linspace(mu - 4*sigma, mu + 4*sigma,
17.     4000)
18. y = normal_pdf(x, mu, sigma)
19. y_mu = normal_pdf(mu, mu, sigma)
20. y_x0 = normal_pdf(x0, mu, sigma)
21. plt.figure(figsize=(6.5, 3))
```

```
21. # 정규분포 곡선
22. plt.plot(x, y, color="deepskyblue", linewidth=2)
23. # 음영: 43 이상 (오른쪽 꼬리)
24. mask = x >= x0
25. plt.fill_between(x[mask], y[mask],
26.     color="deepskyblue",
27.     alpha=0.18)
28. # 수직선
29. plt.vlines(mu, 0, y_mu,
30.     colors="deepskyblue",
31.     linestyle="dashed", linewidth=1.4)
32. plt.vlines(x0, 0, y_x0,
33.     colors="deepskyblue", linewidth=1.4)
34. plt.text(mu, -y.max()*0.06, r"$\mu$",
35.     ha="center", va="top")
36. plt.text(x0, -y.max()*0.06, r"$x_0$",
37.     ha="center", va="top")
38. plt.text(mu + sigma*0.9, y.max()*0.65,
39.     r"$\sigma = 2.0$", fontsize=11)
```

부록#4

- 그래프 코드

- <그림 16> 예제 6.11의 확률 밀도 함수도 (2/2)

```
35. # 축 설정
36. plt.xlabel("x", labelpad=22)
37. plt.xlim(mu - 4*sigma, mu + 4*sigma)
38. plt.ylim(0, y.max()*1.15)
39. plt.yticks([])
40. plt.xticks([])
41. ax = plt.gca()
42. ax.spines["left"].set_visible(False)
43. ax.spines["top"].set_visible(False)
44. ax.spines["right"].set_visible(False)
45.
46. plt.tight_layout()
47. plt.show()
```

부록#4

• 그래프 코드

• <그림 17> 예제 6.12의 확률밀도함수도 (1/2)

```
1. import numpy as np
2. import matplotlib.pyplot as plt
3. import math
4.
5. def normal_pdf(x, mu, sigma):
6.     return (1 / (math.sqrt(2 * math.pi) *
7.         sigma)) * np.exp(
8.             -0.5 * ((x - mu) / sigma) ** 2
9.         )
10. # 분포 파라미터
11. mu = 40
12. sigma = 2.0
13. # 기준값
14. x0 = 43.5
15. # x 범위
16. x = np.linspace(mu - 4*sigma, mu + 4*sigma,
17.     4000)
18. y = normal_pdf(x, mu, sigma)
19. y_mu = normal_pdf(mu, mu, sigma)
20. y_x0 = normal_pdf(x0, mu, sigma)
21. plt.figure(figsize=(6.5, 3))
```

```
21. # 정규분포 곡선
22. plt.plot(x, y, color="deepskyblue", linewidth=2)
23. # 음영: 43.5 이상 (오른쪽 꼬리)
24. mask = x >= x0
25. plt.fill_between(x[mask], y[mask],
26.     color="deepskyblue",
27.     alpha=0.18)
28. # 수직선
29. plt.vlines(mu, 0, y_mu,
30.     colors="deepskyblue",
31.     linestyle="dashed", linewidth=1.4)
32. plt.vlines(x0, 0, y_x0,
33.     colors="deepskyblue", linewidth=1.4)
34. plt.text(mu, -y.max()*0.06, r"$\mu$",
35.     ha="center", va="top")
36. plt.text(x0, -y.max()*0.06, r"$x_0$",
37.     ha="center", va="top")
38. plt.text(mu + sigma*0.9, y.max()*0.65,
39.     r"$\sigma = 2.0$", fontsize=11)
```

부록#4

- 그래프 코드

- <그림 17> 예제 6.12의 확률 밀도 함수도 (2/2)

```
35. # 축 설정
36. plt.xlabel("x", labelpad=22)
37. plt.xlim(mu - 4*sigma, mu + 4*sigma)
38. plt.ylim(0, y.max()*1.15)
39. plt.yticks([])
40. plt.xticks([])
41. ax = plt.gca()
42. ax.spines["left"].set_visible(False)
43. ax.spines["top"].set_visible(False)
44. ax.spines["right"].set_visible(False)
45.
46. plt.tight_layout()
47. plt.show()
```

부록#4

• 그래프 코드

• <그림 18> 예제 6.13의 확률밀도함수도 (1/2)

```
1. import numpy as np
2. import matplotlib.pyplot as plt
3. import math
4.
5. def normal_pdf(x, mu, sigma):
6.     return (1 / (math.sqrt(2 * math.pi) *
7.         sigma)) * np.exp(
8.         -0.5 * ((x - mu) / sigma) ** 2
9.     )
10. # 파라미터
11. mu = 74
12. sigma = 7
13. # 꼬리 확률 ~0.12
14. z_tail = 1.175
15. x0 = mu + sigma * z_tail # ≈ 82.23
16. # x 범위
17. x = np.linspace(mu - 4*sigma, mu + 4*sigma,
18.     4000)
19. y = normal_pdf(x, mu, sigma)
20. y_mu = normal_pdf(mu, mu, sigma)
21. y_x0 = normal_pdf(x0, mu, sigma)
22. plt.figure(figsize=(6.5, 3))
```

```
23. # 정규분포 곡선
24. plt.plot(x, y, color="deepskyblue", linewidth=2)
25. # 음영: 오른쪽 꼬리
26. mask = x >= x0
27. plt.fill_between(x[mask], y[mask],
28.     color="deepskyblue",
29.     alpha=0.18)
30. # 수직선
31. plt.vlines(mu, 0, y_mu,
32.     colors="deepskyblue",
33.     linestyle="dashed", linewidth=1.4)
34. plt.vlines(x0, 0, y_x0,
35.     colors="deepskyblue", linewidth=1.4)
36. plt.text(mu, -y.max()*0.06, r"$\mu$",
37.     ha="center", va="top")
38. plt.text(mu + sigma*0.9, y.max()*0.65,
39.     r"$\sigma = 7$", fontsize=11)
40. plt.text(x0 + sigma*0.15, y.max()*0.12,
41.     r"$x_0 = 82.23$", fontsize=11)
```

부록#4

- 그래프 코드

- <그림 18> 예제 6.13의 확률 밀도 함수도 (2/2)

```
36. # 축 설정
37. plt.xlabel("x", labelpad=22)
38. plt.xlim(mu - 4*sigma, mu + 4*sigma)
39. plt.ylim(0, y.max()*1.15)
40. plt.yticks([])
41. plt.xticks([])
42. ax = plt.gca()
43. ax.spines["left"].set_visible(False)
44. ax.spines["top"].set_visible(False)
45. ax.spines["right"].set_visible(False)
46.
47. plt.tight_layout()
48. plt.show()
```


부록#4

• 그래프 코드

• <그림 19> 예제 6.14의 확률밀도함수도 (1/2)

```
1. import numpy as np
2. import matplotlib.pyplot as plt
3. import math
4.
5. def normal_pdf(x, mu, sigma):
6.     return (1 / (math.sqrt(2 * math.pi) *
7.         sigma)) * np.exp(
8.             -0.5 * ((x - mu) / sigma) ** 2
9.         )
10. # 파라미터
11. mu = 74
12. sigma = 7
13. # D6 (표준정규 분위수  $z_{0.6} \approx 0.253$ )
14. z_06 = 0.253
15. D6 = mu + sigma * z_06 #  $\approx 75.77$ 
16. # x 범위
17. x = np.linspace(mu - 4*sigma, mu + 4*sigma,
18.     4000)
19. y = normal_pdf(x, mu, sigma)
20. y_mu = normal_pdf(mu, mu, sigma)
21. y_D6 = normal_pdf(D6, mu, sigma)
22. plt.figure(figsize=(6.5, 3))
```

```
23. # 정규분포 곡선
24. plt.plot(x, y, color="deepskyblue", linewidth=2)
25. # 음영: 왼쪽 누적 ( $\leq D6$ )
26. mask = x <= D6
27. plt.fill_between(x[mask], y[mask],
28.     color="deepskyblue",
29.     alpha=0.18)
30. # 수직선
31. plt.vlines(mu, 0, y_mu,
32.     colors="deepskyblue",
33.     linestyle="dashed", linewidth=1.4)
34. plt.vlines(D6, 0, y_D6,
35.     colors="deepskyblue", linewidth=1.4)
36. plt.text(mu, -y.max()*0.06, r"$74$",
37.     ha="center", va="top")
38. plt.text(D6, -y.max()*0.06, r"$D_6$",
39.     ha="center", va="top")
40. plt.text(mu - sigma*0.3, y.max()*0.35, r"$0.6$",
41.     fontsize=12)
42. plt.text(mu + sigma*0.9, y.max()*0.65,
43.     r"$\sigma = 7$", fontsize=11)
```

부록#4

- 그래프 코드

- <그림 19> 예제 6.14의 확률 밀도 함수도 (2/2)

```
37. # 축 설정
38. plt.xlabel("x", labelpad=22)
39. plt.xlim(mu - 4*sigma, mu + 4*sigma)
40. plt.ylim(0, y.max()*1.15)
41. plt.yticks([])
42. plt.xticks([])
43. ax = plt.gca()
44. ax.spines["left"].set_visible(False)
45. ax.spines["top"].set_visible(False)
46. ax.spines["right"].set_visible(False)
47.
48. plt.tight_layout()
49. plt.show()
```

부록#4

• 그래프 코드

- <그림 20> 이항분포 $b(x; 15, 0.4)$ 에 대한 $P(7 < X < 9)$ 의 정규근사 (1/2)

```
1. import math
2. import numpy as np
3. import matplotlib.pyplot as plt
4.
5. # 파라미터
6. n = 15
7. p = 0.4
8. mu = n * p
9. sigma = math.sqrt(n * p * (1 - p))
10. # 이항분포
11. x = np.arange(0, n + 1)
12. def binom_pmf(k, n, p):
13.     return math.comb(n, k) * (p ** k) * ((1 - p)
14.         ** (n - k))
15. y_binom = np.array([binom_pmf(k, n, p) for k in
16.     x])
17. # 정규분포
18. xx = np.linspace(0, n, 600)
19. y_norm = (1 / (sigma * math.sqrt(2 * math.pi)))
20.     * np.exp(
21.         -0.5 * ((xx - mu) / sigma) ** 2
22.     )
23. )
```

```
20. # 그래프
21. plt.figure(figsize=(6, 4))
22. ax = plt.gca()
23.
24. #  $P(7 \leq X \leq 9)$  막대
25. bar_colors = []
26. for k in x:
27.     if 7 <= k <= 9:
28.         bar_colors.append("lightpink")
29.     else:
30.         bar_colors.append("white")
31.
32. # 이항분포 막대
33. ax.bar(
34.     x,
35.     y_binom,
36.     width=1.0,
37.     color=bar_colors,
38.     edgecolor="black",
39.     linewidth=1
40. )
```

부록#4

• 그래프 코드

- <그림 20> 이항분포 $b(x; 15, 0.4)$ 에 대한 $P(7 < X < 9)$ 의 정규근사 (2/2)

```
41. # 정규곡선
42. ax.plot(xx, y_norm, color="orange", linewidth=2)
43. # P(7<X<9) 음영
44. mask = (xx >= 7) & (xx <= 9)
45. ax.fill_between(
46.     xx[mask],
47.     y_norm[mask],
48.     color="mediumpurple",
49.     alpha=0.4
50. )
51.
52. ax.grid(
53.     True,
54.     linestyle="--",
55.     linewidth=0.5,
56.     alpha=0.35
57. )
58. ax.spines["right"].set_visible(False)
59. ax.spines["top"].set_visible(False)
60. plt.tight_layout()
61. plt.show()
```

부록#4

• 그래프 코드

- <그림 21> n 이 크고 p 가 0이나 1에 가깝지 않은 경우의 이항분포

```
1. import numpy as np
2. import matplotlib.pyplot as plt
3. import math
4.
5. def binom_pmf(k, n, p):
6.     return math.comb(n, k) * (p ** k) * ((1 - p)
7.     ** (n - k))
8.
9. def pmf_array(n, p):
10.     ks = np.arange(0, n + 1)
11.     ys = np.array([binom_pmf(int(k), n, p) for
12.     k in ks], dtype=float)
13.     return ks, ys
14.
15. # 설정
16. p = 0.2
17. cases = [
18.     (100, "blue", "n=100, p=0.2"),
19.     (200, "red", "n=200, p=0.2"),
20.     (400, "lime", "n=400, p=0.2"),
21. ]
22. plt.figure(figsize=(6, 4), dpi=120)
```

```
23. # 막대그래프
24. for n, color, label in cases:
25.     ks, ys = pmf_array(n, p)
26.     plt.bar(
27.         ks, ys,
28.         width=0.85,
29.         color=color,
30.         alpha=0.35,
31.         edgecolor="black",
32.         linewidth=0.2,
33.         label=label
34.     )
35. # 축 / 범례
36. plt.xlim(-0.5, 100.5)
37. plt.ylim(0, 0.11)
38. plt.legend(loc="upper right", fontsize=9,
39.     frameon=True)
40. plt.tick_params(direction="in", top=True,
41.     right=True)
42. plt.xlabel("x")
43. plt.tight_layout()
44. plt.show()
```

부록#4

• 그래프 코드

• <그림 22> p 가 $\frac{1}{2}$ 에 가까운 경우의 이항분포

```
1. import numpy as np
2. import matplotlib.pyplot as plt
3. import math
4.
5. def binom_pmf(k, n, p):
6.     return math.comb(n, k) * (p ** k) * ((1 - p)
7.     ** (n - k))
8.
9. def pmf_array(n, p):
10.     ks = np.arange(0, n + 1)
11.     ys = np.array([binom_pmf(int(k), n, p) for
12.     k in ks], dtype=float)
13.     return ks, ys
14.
15. # 설정
16. p = 0.5
17. cases = [
18.     (5, "blue", "n=5, p=0.5"),
19.     (15, "red", "n=15, p=0.5"),
20.     (25, "lime", "n=25, p=0.5"),
21. ]
22. plt.figure(figsize=(6, 4), dpi=120)
```

```
23. # 막대그래프
24. for n, color, label in cases:
25.     ks, ys = pmf_array(n, p)
26.     plt.bar(
27.         ks, ys,
28.         width=0.85,
29.         color=color,
30.         alpha=0.8,
31.         edgecolor="black",
32.         linewidth=0.2,
33.         label=label
34.     )
35. # 축 / 범례
36. plt.xlim(-0.5, 20.5)
37. plt.ylim(0, 0.5)
38. plt.legend(loc="upper right", fontsize=9,
39.     frameon=True)
40. plt.tick_params(direction="in", top=True,
41.     right=True)
42. plt.xlabel("x")
43. plt.tight_layout()
44. plt.show()
```

부록#4

• 그래프 코드

• <그림 23> 예제 6.15의 표준정규분포도 (1/2)

```
1. import numpy as np
2. import matplotlib.pyplot as plt
3. import math
4.
5. def normal_pdf(x, mu=0, sigma=1):
6.     return (1 / (math.sqrt(2 * math.pi) *
7.         sigma)) * np.exp(
8.         -0.5 * ((x - mu) / sigma) ** 2
9.     )
10. # 파라미터
11. mu = 0
12. sigma = 1
13. # 기준값
14. x0 = -2.14
15. # x 범위
16. x = np.linspace(-4, 4, 4000)
17. y = normal_pdf(x, mu, sigma)
18. y_mu = normal_pdf(mu, mu, sigma)
19. y_x0 = normal_pdf(x0, mu, sigma)
20. plt.figure(figsize=(6.5, 3))
```

```
21. # 정규분포 곡선
22. plt.plot(x, y, color="deepskyblue", linewidth=2)
23. # 음영: 왼쪽 꼬리
24. mask = x <= x0
25. plt.fill_between(x[mask], y[mask],
26.                 color="deepskyblue",
27.                 alpha=0.18)
28. # 수직선
29. plt.vlines(mu, 0, y_mu,
30.           colors="deepskyblue", linewidth=1.4)
31. plt.vlines(x0, 0, y_x0,
32.           colors="deepskyblue", linewidth=1.4)
33. plt.text(mu, -y.max()*0.06, r"$0$", ha="center",
34.          va="top")
35. plt.text(x0, -y.max()*0.06, r"$-2.14$",
36.          ha="center", va="top")
37. plt.text(mu + 1.0, y.max()*0.65, r"$\sigma = 1$",
38.          fontsize=11)
```

부록#4

- 그래프 코드

- <그림 23> 예제 6.15의 표준정규분포도 (2/2)

```
35. # 축 설정
36. plt.xlabel("z", labelpad=22)
37. plt.xlim(-4, 4)
38. plt.ylim(0, y.max()*1.15)
39. plt.yticks([])
40. plt.xticks([])
41. ax = plt.gca()
42. ax.spines["left"].set_visible(False)
43. ax.spines["top"].set_visible(False)
44. ax.spines["right"].set_visible(False)
45.
46. plt.tight_layout()
47. plt.show()
```


부록#4

• 그래프 코드

• <그림 24> 예제 6.16의 확률밀도함수도 (1/2)

```
1. import numpy as np
2. import matplotlib.pyplot as plt
3. import math
4.
5. def normal_pdf(x, mu=0, sigma=1):
6.     return (1 / (math.sqrt(2 * math.pi) *
7.         sigma)) * np.exp(
8.         -0.5 * ((x - mu) / sigma) ** 2
9.     )
10. # 파라미터
11. mu = 0
12. sigma = 1
13. # 구간
14. x1 = 1.16
15. x2 = 2.71
16. # x 범위
17. x = np.linspace(-4, 4, 4000)
18. y = normal_pdf(x, mu, sigma)
19. y_mu = normal_pdf(mu, mu, sigma)
20. y_x1 = normal_pdf(x1, mu, sigma)
21. y_x2 = normal_pdf(x2, mu, sigma)
22. plt.figure(figsize=(6.5, 3))
```

```
23. # 정규분포 곡선
24. plt.plot(x, y, color="deepskyblue", linewidth=2)
25. # 음영: [1.16, 2.71]
26. mask = (x >= x1) & (x <= x2)
27. plt.fill_between(x[mask], y[mask],
28.     color="deepskyblue",
29.     alpha=0.18)
30. # 수직선
31. plt.vlines(mu, 0, y_mu,
32.     colors="deepskyblue",
33.     linestyle="dashed", linewidth=1.4)
34. plt.vlines(x1, 0, y_x1,
35.     colors="deepskyblue", linewidth=1.4)
36. plt.vlines(x2, 0, y_x2,
37.     colors="deepskyblue", linewidth=1.4)
38. plt.text(mu, -y.max()*0.06, r"$\mu$", ha="center",
39.     va="top")
40. plt.text(x1, -y.max()*0.06, r"$x_1$", ha="center", va="top")
41. plt.text(x2, -y.max()*0.06, r"$x_2$", ha="center", va="top")
42. plt.text(mu + 1.2, y.max()*0.65, r"$\sigma = 1$",
43.     fontsize=11)
```

부록#4

- 그래프 코드

- <그림 24> 예제 6.16의 확률 밀도 함수도 (2/2)

```
40. # 축 설정
41. plt.xlabel("z", labelpad=22)
42. plt.xlim(-4, 4)
43. plt.ylim(0, y.max()*1.15)
44. plt.yticks([])
45. plt.xticks([])
46. ax = plt.gca()
47. ax.spines["left"].set_visible(False)
48. ax.spines["top"].set_visible(False)
49. ax.spines["right"].set_visible(False)
50.
51. plt.tight_layout()
52. plt.show()
```

부록#4

• 그래프 코드

• <그림 25> α, β 에 따른 감마분포의 변화

```
1. import numpy as np
2. import matplotlib.pyplot as plt
3. import math
4.
5. def gamma_pdf(x, alpha, beta):
6.     #  $f(x) = x^{\alpha-1} e^{-x/\beta} / (\Gamma(\alpha) \beta^\alpha)$ ,
6.     #  $x > 0$ 
7.     return (x**(alpha - 1) * np.exp(-x / beta))
7.     / (math.gamma(alpha) * beta**alpha)
8.
9. # x 범위
10. x = np.linspace(0, 6, 4000)
11. x[0] = 1e-6
12. beta = 1
13. # 파라미터
14. alphas = [1, 2, 4]
15. plt.figure(figsize=(7.2, 4))
16. for a in alphas:
17.     y = gamma_pdf(x, a, beta)
18.     plt.plot(x, y, color="deepskyblue",
18.     linewidth=2)
```

```
19. plt.text(0.55, 0.65,
19.     r"$\alpha=1$"+"\\n"+r"$\beta=1$", fontsize=11)
20. plt.text(1.7, 0.38,
20.     r"$\alpha=2$"+"\\n"+r"$\beta=1$", fontsize=11)
21. plt.text(3.0, 0.23,
21.     r"$\alpha=4$"+"\\n"+r"$\beta=1$", fontsize=11)
22. plt.xlabel("x", labelpad=10)
23. plt.ylabel(r"$f(x)$", rotation=0, labelpad=15)
24. plt.xlim(0, 6)
25. plt.ylim(0, 1.05)
26. plt.yticks([0.5, 1.0])
27. plt.xticks([0, 1, 2, 3, 4, 5, 6])
28. ax = plt.gca()
29. ax.spines["top"].set_visible(False)
30. ax.spines["right"].set_visible(False)
31. ax.spines["left"].set_linewidth(1.2)
32. ax.spines["bottom"].set_linewidth(1.2)
33.
34. plt.tight_layout()
35. plt.show()
```

부록#4

• 그래프 코드

• <그림 28> μ, σ 에 따른 로그정규분포의 변화

```
1. import numpy as np
2. import matplotlib.pyplot as plt
3. import math
4.
5. def lognormal_pdf(x, mu, sigma):
6.     # f(x) = 1/(x σ √(2π)) * exp( -(ln x - μ)^2 / (2σ^2) ), x>0
7.     return (1 / (x * sigma * math.sqrt(2 *
8.         math.pi)))) * np.exp(
9.         - (np.log(x) - mu)**2 / (2 * sigma**2)
10.     )
11. # x 범위
12. x = np.linspace(0.001, 5, 4000)
13. # 파라미터
14. mu_values = [0, 1]
15. sigma = 1
16.
17. plt.figure(figsize=(7.2, 4))
18. for mu in mu_values:
19.     y = lognormal_pdf(x, mu, sigma)
20.     plt.plot(x, y, color="deepskyblue",
21.         linewidth=2)
```

```
21. plt.text(0.55, 0.55,
22.     r"$\mu=0$"+"\\n"+r"$\sigma=1$", fontsize=11)
23. plt.text(3.0, 0.15,
24.     r"$\mu=1$"+"\\n"+r"$\sigma=1$", fontsize=11)
25. plt.xlabel("x", labelpad=10)
26. plt.ylabel(r"$f(x)$", rotation=0, labelpad=15)
27. plt.xlim(0, 5)
28. plt.ylim(0, 0.7)
29. plt.xticks([0, 1, 2, 3, 4, 5])
30. plt.yticks([0.2, 0.4, 0.6])
31. ax = plt.gca()
32. ax.spines["top"].set_visible(False)
33. ax.spines["right"].set_visible(False)
34. ax.spines["left"].set_linewidth(1.2)
35. ax.spines["bottom"].set_linewidth(1.2)
36. plt.tight_layout()
37. plt.show()
```

부록#4

• 그래프 코드

• <그림 30> 예제 6.23의 확률 밀도 함수도

```
1. import numpy as np
2. import matplotlib.pyplot as plt
3. import math
4.
5. # 로그정규분포
6. def lognormal_pdf(x, mu, sigma):
7.     return (1 / (x * sigma * math.sqrt(2 *
8.         math.pi)))) * np.exp(
9.         - (np.log(x) - mu) ** 2 / (2 * sigma **
10.         2)
11.     )
12.
13. # 파라미터
14. mu = 3.2
15. sigma = 1
16. x0 = 8
17. # x 범위
18. x = np.linspace(0.01, 150, 3000)
19. y = lognormal_pdf(x, mu, sigma)
20. plt.figure(figsize=(7, 3.5))
21. # 밀도함수
22. plt.plot(x, y, color="deepskyblue", linewidth=2)
```

```
22. # 수직 점선
23. y_x0 = lognormal_pdf(x0, mu, sigma)
24. plt.vlines(x0, 0, y_x0, colors="deepskyblue",
25.     linestyle="dashed")
26. # 음영
27. mask = x >= x0
28. plt.fill_between(x[mask], y[mask],
29.     color="deepskyblue", alpha=0.25)
30. plt.text(20, max(y)*0.75, r"$\mu=3.2$",
31.     fontsize=11)
32. plt.text(20, max(y)*0.65, r"$\sigma=1$",
33.     fontsize=11)
34. plt.xlabel("x", labelpad=10)
35. plt.ylabel(r"$f(x)$", rotation=0, labelpad=15)
36. plt.xlim(0, 150)
37. plt.ylim(0, max(y) * 1.15)
38. plt.xticks([0, 8, 20, 40])
39. plt.yticks([])
40. ax = plt.gca()
41. ax.spines["top"].set_visible(False)
42. ax.spines["right"].set_visible(False)
43. plt.tight_layout()
44. plt.show()
```

부록#4

• 그래프 코드

• <그림 31> 예제 6.24의 확률 밀도함수도

```
1. import numpy as np
2. import matplotlib.pyplot as plt
3. import math
4.
5. # 로그정규분포
6. def lognormal_pdf(x, mu, sigma):
7.     return (1 / (x * sigma * math.sqrt(2 *
8.         math.pi))) * np.exp(
9.         - (np.log(x) - mu) ** 2 / (2 * sigma ** 2)
10.     )
11. # 파라미터
12. mu = 5.149
13. sigma = 0.737
14. x_q = 51.265
15. # x 범위
16. x = np.linspace(1, 600, 4000)
17. y = lognormal_pdf(x, mu, sigma)
18. plt.figure(figsize=(7, 3.5))
19.
20. # 밀도함수
21. plt.plot(x, y, color="deepskyblue", linewidth=2)
```

```
22. # 수직 점선
23. y_q = lognormal_pdf(x_q, mu, sigma)
24. plt.vlines(x_q, 0, y_q,
25.             colors="deepskyblue",
26.             linestyle="dashed")
27. # 음영
28. mask = x <= x_q
29. plt.fill_between(x[mask], y[mask],
30.                  color="deepskyblue", alpha=0.25)
31. plt.text(150, max(y)*0.75, r"$\mu=5.149$",
32.          fontsize=11)
33. plt.text(150, max(y)*0.65, r"$\sigma=0.737$",
34.          fontsize=11)
35. plt.xlabel("x", labelpad=10)
36. plt.ylabel(r"$f(x)$", rotation=0, labelpad=15)
37. plt.xlim(0, 600)
38. plt.ylim(0, max(y) * 1.15)
39. plt.xticks([0, 50, 100, 200, 300])
40. plt.yticks([])
41. ax = plt.gca()
42. ax.spines["top"].set_visible(False)
43. ax.spines["right"].set_visible(False)
44. plt.tight_layout()
45. plt.show()
```

부록#4

• 그래프 코드

• <그림 32> $\alpha = 1$ 일 때 β 에 따른 와이블분포 변화

```
1. import numpy as np
2. import matplotlib.pyplot as plt
3. import math
4.
5. def weibull_pdf(x, beta, eta=1):
6.     # f(x) = (β/η) (x/η)^{β-1} exp(-(x/η)^β), x
    ≥ 0
7.     return (beta / eta) * (x / eta)**(beta - 1)
    * np.exp(-(x / eta)**beta)
8.
9. # x 범위
10. x = np.linspace(0, 2.2, 5000)
11. x[0] = 1e-6
12. # 파라미터
13. betas = [1, 2, 3.5]
14. eta = 1
15. # y 최대값 계산
16. y_max = 0
17. ys = {}
18. for b in betas:
19.     y = weibull_pdf(x, b, eta)
20.     ys[b] = y
21.     y_max = max(y_max, y.max())
```

```
22. plt.figure(figsize=(7.2, 4))
23. for b in betas:
24.     plt.plot(x, ys[b], color="deepskyblue",
        linewidth=2)
25. plt.text(0.15, y_max * 0.55, r"$\beta = 1$",
        fontsize=11) # 위로
26. plt.text(0.75, y_max * 0.40, r"$\beta = 2$",
        fontsize=11) # 위로
27. plt.text(1.05, y_max * 0.85, r"$\beta = 3.5$",
        fontsize=11) # 오른쪽으로
28. plt.xlabel("x", labelpad=10)
29. plt.ylabel(r"$f(x)$", rotation=0, labelpad=15)
30. plt.xlim(0, 2.2)
31. plt.ylim(0, y_max * 1.08)
32. plt.xticks([0, 0.5, 1.0, 1.5, 2.0])
33. plt.yticks([])
34. ax = plt.gca()
35. ax.spines["top"].set_visible(False)
36. ax.spines["right"].set_visible(False)
37. ax.spines["left"].set_linewidth(1.2)
38. ax.spines["bottom"].set_linewidth(1.2)
39. plt.tight_layout()
40. plt.show()
```

부록#4

• 그래프 코드

• <그림 33> β 에 따른 고장률 변화

```
1. import numpy as np
2. import matplotlib.pyplot as plt
3.
4. # Hazard rate 정의
5. #  $Z(t) = \alpha \beta t^{\beta-1}$ 
6. def hazard_rate(t, alpha, beta):
7.     return alpha * beta * t**(beta - 1)
8.
9. # 파라미터
10. alpha = 0.01
11. betas = [0.5, 1.0, 1.5]
12. # 시간 범위
13. t = np.linspace(0.01, 20, 2000)
14. plt.figure(figsize=(7, 4))
15.
16. # 곡선
17. for beta in betas:
18.     z = hazard_rate(t, alpha, beta)
19.     plt.plot(t, z, linewidth=2, label=fr"$\beta$
= {beta}$")
```

```
20. plt.xlabel("t", labelpad=10)
21. plt.ylabel("Z(t)", rotation=0, labelpad=15)
22. plt.legend()
23. plt.xlim(0, 20)
24. plt.ylim(0, hazard_rate(t, alpha, 1.5).max() *
1.3)
25. ax = plt.gca()
26. ax.spines["top"].set_visible(False)
27. ax.spines["right"].set_visible(False)
28.
29. plt.tight_layout()
30. plt.show()
```


부록#4

• 그래프 코드

• <그림 34> 예제 6.25의 확률 밀도 함수도

```
1. import numpy as np
2. import matplotlib.pyplot as plt
3.
4. # 와이블분포
5. def weibull_pdf(x, alpha, beta):
6.     #  $f(x) = \alpha \beta x^{\beta-1} \exp(-\alpha x^\beta)$ ,  $x \geq 0$ 
7.     return alpha * beta * x**(beta - 1) *
8.     np.exp(-alpha * x**beta)
9.
10. # 파라미터
11. alpha = 0.01
12. beta = 2
13. # x 범위
14. x = np.linspace(0, 30, 2000)
15. y = weibull_pdf(x, alpha, beta)
16. plt.figure(figsize=(7, 3.5))
17. # 밀도함수
18. plt.plot(x, y, color="deepskyblue", linewidth=2)
19. # 수직선(점선)
20. plt.vlines(8, 0, weibull_pdf(8, alpha, beta),
21.           color="deepskyblue",
22.           linestyle="dashed")
```

```
22. # 음영
23. mask = x <= 8
24. plt.fill_between(x[mask], y[mask],
25.                  color="deepskyblue",
26.                  alpha=0.25)
27. plt.text(14, max(y)*0.7, r"$\alpha=0.01$",
28.          fontsize=11)
29. plt.text(14, max(y)*0.6, r"$\beta=2$",
30.          fontsize=11)
31. plt.xlabel("x", labelpad=10)
32. plt.ylabel(r"$f(x)$", rotation=0, labelpad=15)
33. plt.xlim(0, 30)
34. plt.ylim(0, max(y) * 1.15)
35. plt.yticks([])
36. plt.xticks([0, 8, 16, 24])
37. ax = plt.gca()
38. ax.spines["top"].set_visible(False)
39. ax.spines["right"].set_visible(False)
40. plt.tight_layout()
41. plt.show()
```