I. 표본평균의 표본분포

II. 중심극한정리

III. 표본분산의 표본분포

Ⅰ. 표본평균의 표본분포

- X~ N(μ, σ^2)이면, $\bar{X}$~N(μ, $\frac{\sigma^2}{n}$)

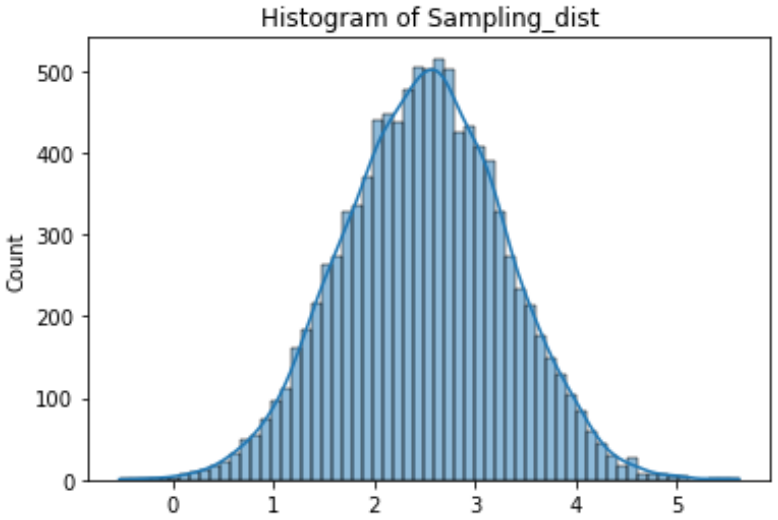
  단, n은 표본의 크기(sample size)

- (예 1) 평균이 2.5이고, 분산이 1.25인 모집단에서 표본크기가 2인 표본을 추출할 경우 표본평균의 평균은 모평균과 동일한 2.5가 되고, 분산은 1.25/2인 분포를 근사적으로 따르는데 10,000개 표본평균의 평균과 분산은 각각 2.498928 및 0.643669로 모집단의 이론적인 평균과 분산에 근접함을 알 수 있음

b1-ch5-1.py

```
import numpy as np
import scipy.stats as stats
import seaborn as sns
# set the random seed:
np.random.seed(123456)
# set sample size:
n=2
# initialize sampling dist. to an array of length r=10000 to later store results:
r = 10000
sample_dist = np.empty(r)
# repeat r times:
for j in range(1,r):
    # draw a sample and store the sample mean in pos. j=0,1,... of sample_dist:
    sample = stats.norm.rvs(2.5, 1.1182, size=n)
    sample_dist[j] = np.mean(sample)
mean = np.mean(sample_dist)
variance = np.var(sample_dist)
print("Mean of sampl mean distribution is :", mean)
print("Variance of sample mean distribution is :", variance)
sns.histplot(data=sample_dist, x=None, kde=True).set(title='Histogram of Sampling_dist')
```

```
Mean of sampl mean distribution is : 2.498928953436337
Variance of sample mean distribution is : 0.6436691927820287
```

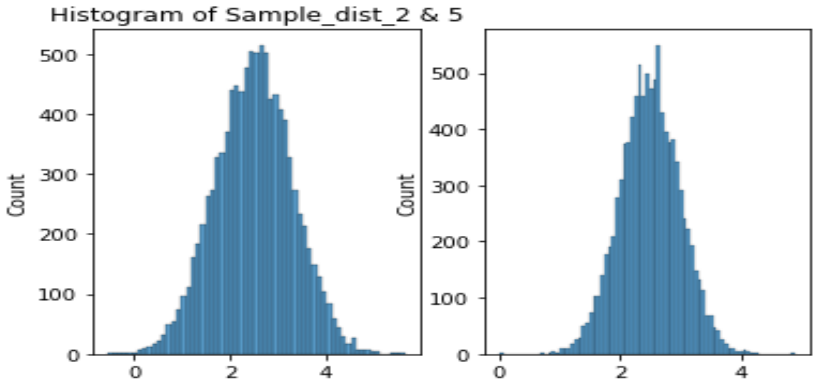


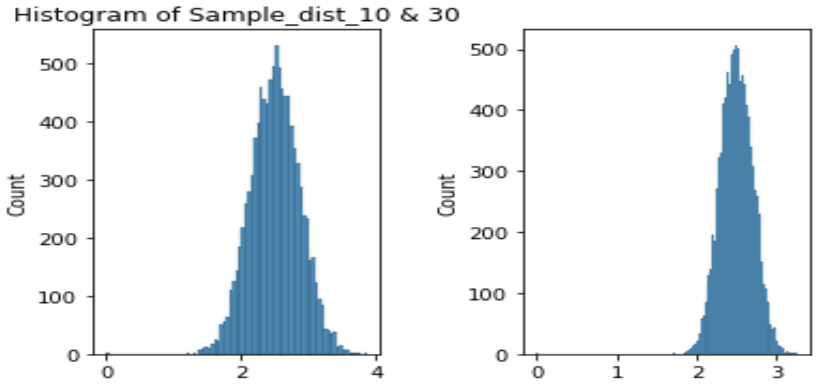
- **(예 2) 표본평균의 평균은 표본크기에 관계없이 모평균과 동일하고, 표본평균의 분산은 모분산을 표본크기로 나눈 값과 같으므로 표본크기가 커짐에 따라 표본평균의 분산은 작아짐을 보일 수 있음**

b1-ch5-2.py

```
(패키지 생략)
# set the random seed:
np.random.seed(123456)
# set sample size:
n1=2
n2=5
n3=10
n4=30
# initialize sampling dist. to an array of length r=10000 to later store results:
r = 10000
sample_dist_2 = np.empty(r)
sample_dist_5 = np.empty(r)
sample_dist_10 = np.empty(r)
sample_dist_30 = np.empty(r)
# repeat r times:
for j in range(1,r):
   # draw a sample and store the sample mean in pos. j=0,1,... of sample_dist:
   sample_2 = stats.norm.rvs(2.5, 1.1182, size=n1)
   sample_dist_2[j] = np.mean(sample_2)
(중략)
mean_2 = np.mean(sample_dist_2)
variance_2 = np.var(sample_dist_2)
print("Mean of sampling distribution w/ n=2 is :", mean_2)
print("Variance of sampling distribution w/ n=2 is :", variance_2)
(중략)
mean_5 = np.mean(sample_dist_5)
variance_5 = np.var(sample_dist_5)
print("Mean of sampling distribution w/ n=5 is :", mean_5)
print("Variance of sampling distribution w/ n=5 is :", variance_5)
fig, ax = plt.subplots()
sns.histplot(data=sample_dist_2, x=None, kde=True).set(title='Histogram of Sampling_dist_2')
ax.set_xlim(-1,6)
plt.show()
(이하 생략)
```

```
Mean of sampling distribution w/ n=2 is : 2.498928953436337
Variance of sampling distribution w/ n=2 is : 0.6436691927820287
Mean of sampling distribution w/ n=5 is : 2.502151496206422
Variance of sampling distribution w/ n=5 is : 0.24916734804551638
Mean of sampling distribution w/ n=10 is : 2.50227657597811
Variance of sampling distribution w/ n=10 is : 0.13186198044252634
Mean of sampling distribution w/ n=30 is : 2.499861103239455
Variance of sampling distribution w/ n=30 is : 0.042058834444569496
```





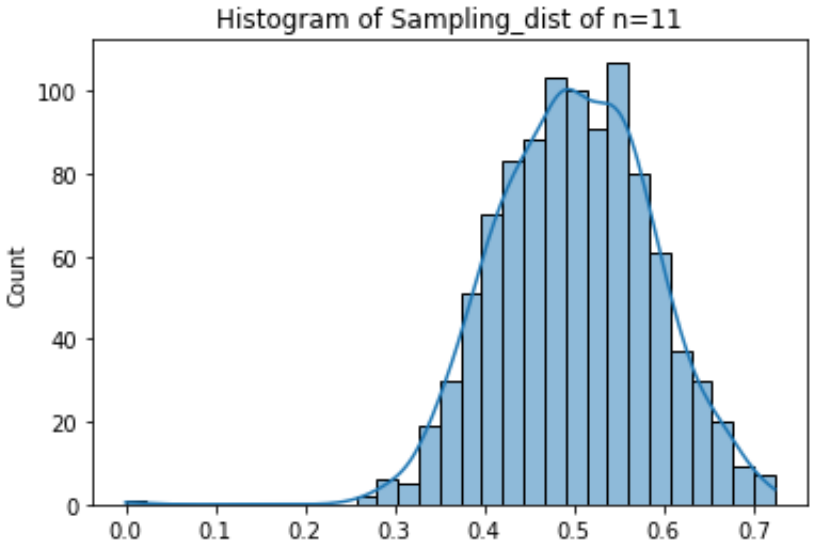
II. 중심극한정리

- X~ (μ, σ^2)이면, n이 커짐에 따라 $\bar{X} \sim N(\mu, \frac{\sigma^2}{n})$, n은 표본의 크기(sample size)
- (실험 1) X~U(0,1)에서 X의 평균은 1/2, 분산은 1/12이므로 n=11인 경우 표본평균 $\bar{X}$는 근사적으로 N(1/2, 1/132)을 따르는데 1,000개 표본평균의 평균과 분산은 각각 0.500009및 0.0073으로 모집단의 이론적인 평균과 표준편차에 근접함을 알 수 있음

b1-ch5-3.py

```
import numpy as np
#import scipy.stats as stats
import seaborn as sns
from numpy import random
# set the random seed:
np.random.seed(1234556)
# set sample size:
n=11
min = 0
max = 1
# initialize sampling dist. to an array of length r=10000 to later store results:
r = 1000
sample_dist = np.empty(r)
# repeat r times:
for j in range(1,r):
    # draw a sample and store the sample mean in pos. j=0,1,... of sample_dist:
    sample = random.uniform(0,1,size=n)
    sample_dist[j] = np.mean(sample)
mean = np.mean(sample_dist)
variance = np.var(sample_dist)
print("Mean of sampling distribution from Unifrom(0,1) is :", mean)
print("Variance of sampling distribution from Uniform(0,1)is :", variance)
sns.histplot(data=sample_dist, x=None, kde=True).set(title='Histogram of Sampling_dist of n=11')
```

```
Mean of sampling distribution from Unifrom(0,1) is : 0.5000949407552135
Variance of sampling distribution from Uniform(0,1)is : 0.0072543239423110 56
```



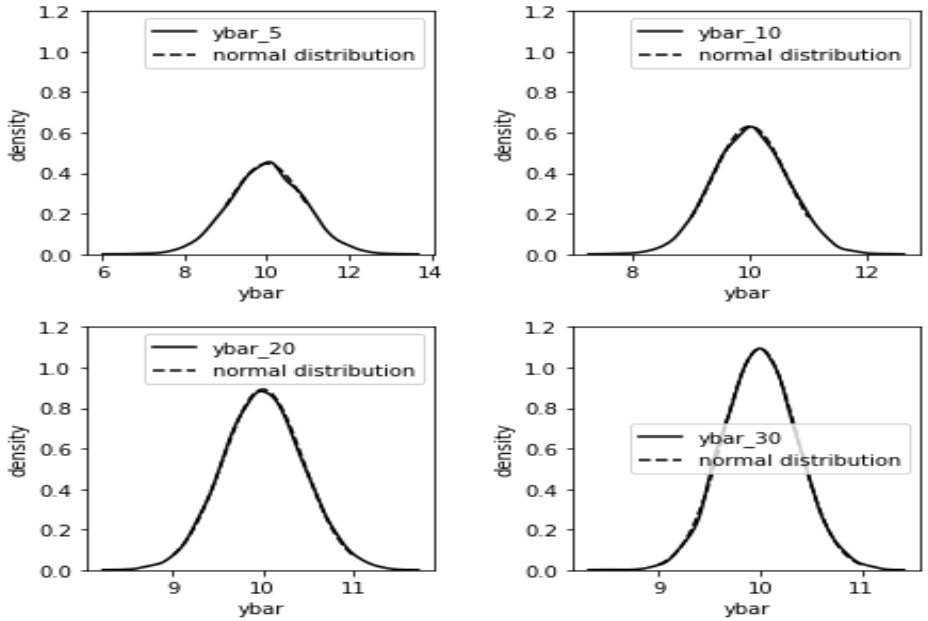
- **(실험 2) 확률변수 X가 평균이 10, 표준편차가 2인 정규분포를 따른다고 할 때 이러한 확률분포로부터 표본크기가 각각 5, 10, 20, 30인 확률표본을 10,000개 추출하는 실험을 통해 표본크기에 따라 10,000개 표본평균의 평균과 분산은 모집단의 이론적인 평균과 분산에 근접함을 알 수 있음**

b1-ch5-4.py

```
(패키지 생략)
# set the random seed:
np.random.seed(123456)
# set sample size:
n_5 = 5
n_10 = 10
n_20 = 20
n_30 = 30
# initialize ybar to an array of length r=10000 to later store results:
r = 10000
ybar_5 = np.empty(r)
ybar_10 = np.empty(r)
ybar_20 = np.empty(r)
ybar_30 = np.empty(r)
# repeat r times:
for j in range(r):
    # draw a sample and store the sample mean in pos. j=0,1,... of ybar:
    sample_5 = stats.norm.rvs(10, 2, size=n_5)
    ybar_5[j] = np.mean(sample_5)
(중략)
mean_5 = np.mean(ybar_5)
variance_5 = np.var(ybar_5)
print("Mean of sampling distribution w/ n=5 is :", mean_5)
print("Variance of sampling distribution w/ n=5 is :", variance_5)
(중략)
# simulated density:
kde_5 = sm.nonparametric.KDEUnivariate(ybar_5)
kde_5.fit()
(중략)
# normal density:
x_range_5 = np.linspace(9, 11)
y_5 = stats.norm.pdf(x_range_5, 10, np.sqrt(4/n_5))
(이하 생략)
```

구분	n=5	n=10	n=20	n=30
표본평균의 평균	10.00254	10.00433	10.00125	10.00179
표본평균의 분산	0.80042	0.4003	0.1985	0.12997

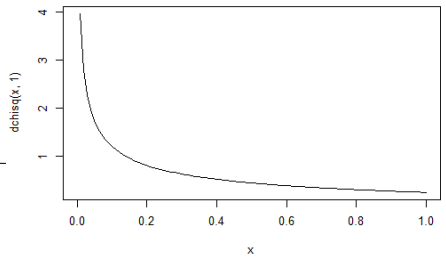
```
Mean of sampling distribution w/ n=5 is : 10.002537846001054
Variance of sampling distribution w/ n=5 is : 0.8004204313463208
Mean of sampling distribution w/ n=10 is : 10.00433456892261
Variance of sampling distribution w/ n=10 is : 0.40030210086547724
Mean of sampling distribution w/ n=20 is : 10.001253312586906
Variance of sampling distribution w/ n=20 is : 0.1985036244242967
Mean of sampling distribution w/ n=30 is : 10.001797270034292
Variance of sampling distribution w/ n=30 is : 0.12997603914791894
```



- **(실험 3) 확률변수 X가 자유도가 1인 χ^2-분포를 따른다고 할 때(따라서 평균은 1, 분산은 2됨) 이러한 확률분포로부터 표본크기가 각각 2, 10, 20, 30인 확률표본을 10,000개 추출하는 실험을 통해 표본크기에 따라 10,000개 표본평균의 평균과 분산은 모집단의 이론적인 평균과 분산에 근접함을 알 수 있음**

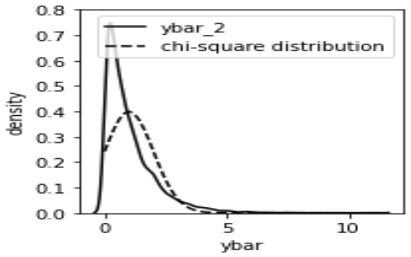
b1-ch5-5.py

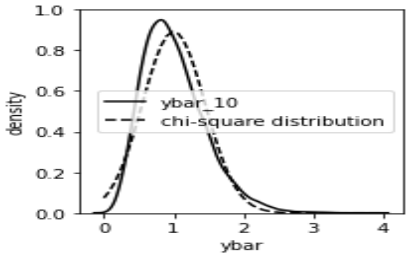
```
(패키지 생략)
# set the random seed:
np.random.seed(123456)
# set sample size:
n_2 = 2
n_10 = 10
n_20 = 20
n_30 = 30
# initialize ybar to an array of length r=10000 to later store results:
r = 10000
ybar_2 = np.empty(r)
ybar_10 = np.empty(r)
ybar_20 = np.empty(r)
ybar_30 = np.empty(r)
# repeat r times:
for j in range(r):
    # draw a sample and store the sample mean in pos. j=0,1,... of ybar:
    sample_2 = random.chisquare(df=1, size=n_2)
    ybar_2[j] = np.mean(sample_2)
(중략)
mean_2 = np.mean(ybar_2)
variance_2 = np.var(ybar_2)
print("Mean of sampling distribution w/ df=1 & n=2 is :", mean_2)
print("Variance of sampling distribution w/ df=1 & n=2 is :", variance_2)
(중략)
# simulated density:
kde_2 = sm.nonparametric.KDEUnivariate(ybar_2)
kde_2.fit()
(이하 생략)
```

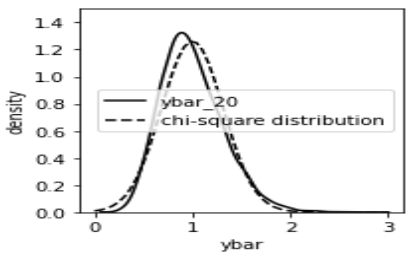


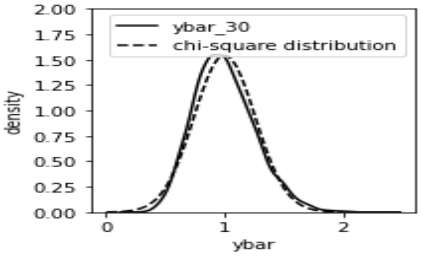
구분	n=2	n=10	n=20	n=30
표본평균의 평균	1.01986	1.00539	0.99734	1.00046
표본평균의 분산	1.06407	0.20082	0.10035	0.06607

```
Mean of sampling distribution w/ df=1 & n=2 is : 1.0198626990259951
Variance of sampling distribution w/ df=1 & n=2 is : 1.064072802545432
Mean of sampling distribution w/ df=1 & n=10 is : 1.0053906557889623
Variance of sampling distribution w/ df=1 & n=10 is : 0.20082294752210522
Mean of sampling distribution w/ df=1 & n=20 is : 0.9973445237790578
Variance of sampling distribution w/ df=1 & n=20 is : 0.10035633077162097
Mean of sampling distribution w/ df=1 & n=30 is : 1.0004656707524442
Variance of sampling distribution w/ df=1 & n=30 is : 0.06607824052889547
```









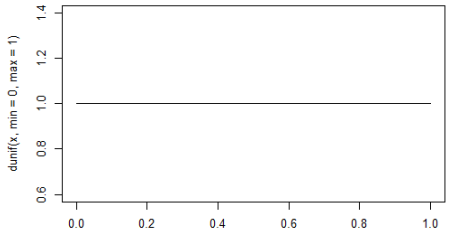
- **(실험 4) 확률변수 X가 균등분포 U(0,1)을 따른다고 할 때(평균은 0.5, 분산은 0.083333) 이러한 확률분포로부터 표본크기가 각각 2, 10, 20, 30인 확률표본을 10,000개 추출하는 실험을 통해 표본크기에 따라 10,000개 표본평균의 평균과 분산은 모집단의 이론적인 평균과 분산에 근접함을 알 수 있음**

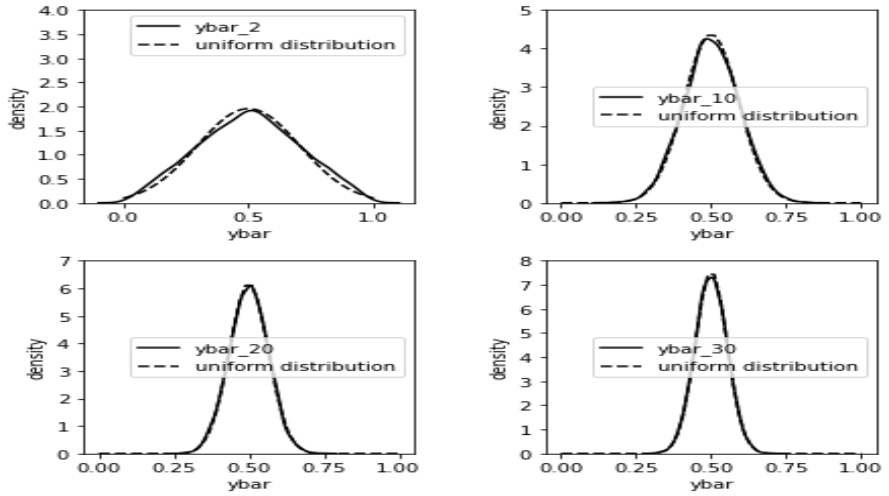
b1-ch5-6.py

```
(패키지 생략)
# set the random seed:
np.random.seed(123456)
# set sample size:
n_2 = 2
n_10 = 10
n_20 = 20
n_30 = 30
# initialize ybar to an array of length r=10000 to later store results:
r = 10000
ybar_2 = np.empty(r)
ybar_10 = np.empty(r)
ybar_20 = np.empty(r)
ybar_30 = np.empty(r)
# repeat r times:
for j in range(r):
    # draw a sample and store the sample mean in pos. j=0,1,... of ybar:
    sample_2 = np.random.uniform(0,1,size=n_2)
    ybar_2[j] = np.mean(sample_2)
(중략)
mean_2 = np.mean(ybar_2)
variance_2 = np.var(ybar_2)
print("Mean of sampling distribution w/ n=2 is :", mean_2)
print("Variance of sampling distribution w/ n=2 is :", variance_2)
(중략)
mean_10 = np.mean(ybar_10)
variance_10 = np.var(ybar_10)
print("Mean of sampling distribution w/ n=10 is :", mean_10)
print("Variance of sampling distribution w/ n=10 is :", variance_10)
# simulated density:
kde_2 = sm.nonparametric.KDEUnivariate(ybar_2)
kde_2.fit()
(이하 생략)
```

구분	n=2	n=10	n=20	n=30
표본평균의 평균	0.502469	0.5009	0.49999	0.49975
표본평균의 분산	0.041391	0.00857	0.00412	0.00279

```
Mean of sampling distribution w/ n=2 is : 0.5024698440268437
Variance of sampling distribution w/ n=2 is : 0.041391861739228514
Mean of sampling distribution w/ n=10 is : 0.5009044104487875
Variance of sampling distribution w/ n=10 is : 0.008579535676001086
Mean of sampling distribution w/ n=20 is : 0.4999959114463433
Variance of sampling distribution w/ n=20 is : 0.0041209806985549125
Mean of sampling distribution w/ n=30 is : 0.499750243998388
Variance of sampling distribution w/ n=30 is : 0.0027946113216498225
```





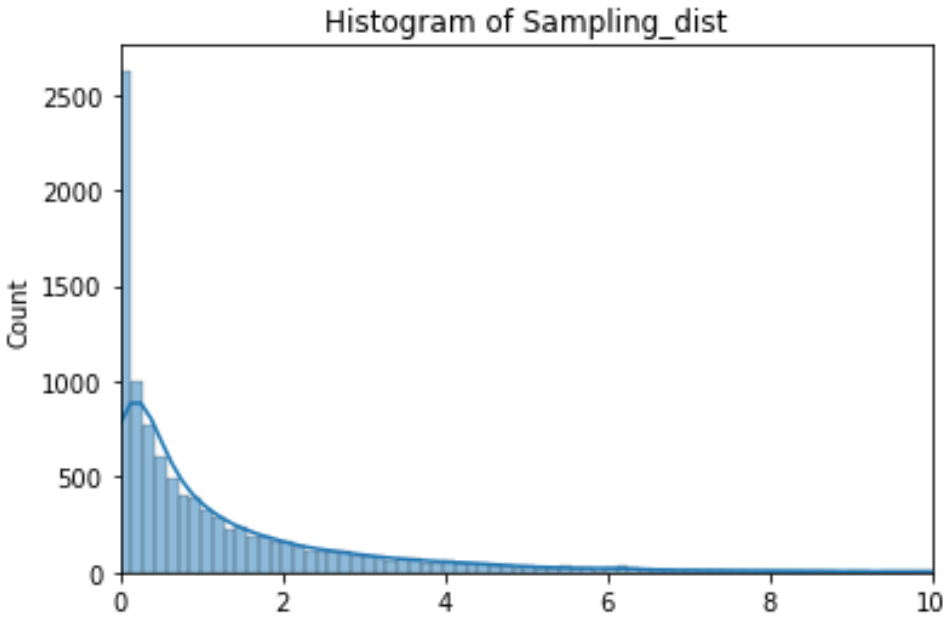
Ⅲ. 표본분산의 표본분포

- 모집단이 정규분포에 따르더라도 표본분산의 표본분포는 정규분포에 따르지 않고 표본분산의 평균은 σ^2, 분산은 $\frac{2\sigma^4}{n-1}$인 분포에 따름
- (예 1) 평균이 2.5이고, 분산이 1.25인 모집단에서 표본크기가 2인 표본을 추출할 경우 표본분산은 근사적으로 평균 1.25, 분산 3.125인 분포를 따르는데 10,000개 표본의 표본분산의 평균 1.260899, 분산 3.176788로 모집단의 이론적인 평균과 분산에 근접함을 알 수 있음

b1-ch5-7.py

```
import numpy as np
import scipy.stats as stats
import seaborn as sns
import matplotlib.pyplot as plt
# set the random seed:
np.random.seed(12345)
# set sample size:
n=2
# initialize sampling dist. to an array of length r=10000 to later store results:
r = 10000
sample_dist = np.empty(r)
# repeat r times:
for j in range(1,r):
    # draw a sample and store the sample mean in pos. j=0,1,... of sample_dist:
    sample = stats.norm.rvs(2.5, 1.118, size=2)
    sample_dist[j] = np.var(sample, ddof=1)
mean = np.mean(sample_dist)
variance = np.var(sample_dist,ddof=1)
print("Mean of sample variance distribution is :", mean)
print("Variance of sample variance distribution is :", variance)
fig, ax = plt.subplots()
sns.histplot(data=sample_dist, ax=ax, kde=True).set(title='Histogram of Sampling_dist')
ax.set_xlim(0,10)
```

```
Mean of sample variance distribution is : 1.2608993584730102
Variance of sample variance distribution is : 3.1767885335022683
```

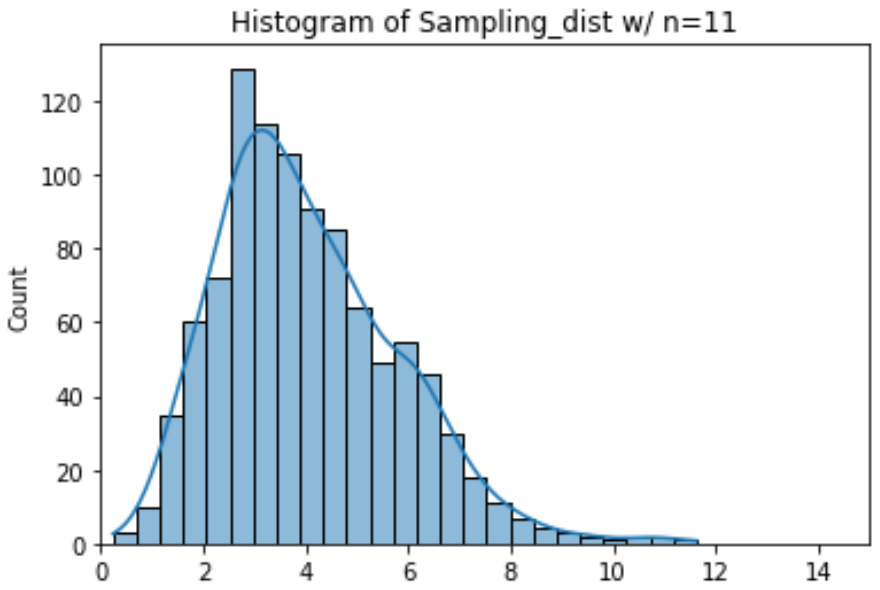


- (예 2) 평균이 10이고, 표준편차가 2인 정규분포에 따르는 모집단에서 표본크기가 11인 1,000개 표본으로부터 구한 표본분산 s_1^2 , s_2^2 , … , s_{1000}^2은 정규분포에 따르지 않는다는 것을 확인할 수 있음

b1-ch5-8.py

```
import numpy as np
#import statsmodels.api as sm
import scipy.stats as stats
import matplotlib.pyplot as plt
import seaborn as sns
# set the random seed:
np.random.seed(123456)
# set sample size:
n = 11
# initialize r to an array of length r=1000 to later store results:
r = 1000
sample_mean = np.empty(r)
sample_var = np.empty(r)
# repeat r times:
for j in range(r):
   # draw a sample and store the sample mean in pos. j=0,1,... of ybar:
   sample = stats.norm.rvs(10, 2, size=n)
   sample_mean[j] = np.mean(sample)
   sample_var[j] = np.var(sample, ddof=1)
mean = np.mean(sample_var)
variance = np.var(sample_var, ddof=1)
print("Mean of sampling distribution from Normal is :", mean)
print("Variance of sampling distribution from Normal is :", variance)
# simulated density:
#kde = sm.nonparametric.KDEUnivariate(sample_var)
#kde.fit()
fig, ax = plt.subplots()
sns.histplot(data=sample_var, x=None, kde=True).set(title='Histogram of Sampling_dist w/ n=11')
ax.set_xlim(0,15)
#plt.savefig('C:/BOOK/PyBasics/PyStat/code/b1-ch5-8.png')
```

```
Mean of sampling distribution from Normal is : 4.043155014435903
Variance of sampling distribution from Normal is : 3.054821584697537
```



- (예 3) 표본크기를 다르게 할 경우 표본분산의 분포는 모집단의 이론적인 평균인 σ^2과 분산인 $\frac{2\sigma^4}{n-1}$에 근접함을 알 수 있음

b1-ch5-9.py

```
(패키지 생략)
# set the random seed:
np.random.seed(123456)
# set sample size:
n1=10
n2=20
n3=30
n4=100
# initialize sampling dist. to an array of length r=10000 to later store results:
r = 10000
yvar_10 = np.empty(r)
yvar_20 = np.empty(r)
yvar_30 = np.empty(r)
yvar_100 = np.empty(r)
# repeat r times:
for j in range(1,r):
    # draw a sample and store the sample mean in pos. j=0,1,... of sample_dist:
    sample_10 = stats.norm.rvs(10, 2, size=n1)
    yvar_10[j] = np.var(sample_10, ddof=1)
(중략)
# repeat r times:
for j in range(1,r):
    # draw a sample and store the sample mean in pos. j=0,1,... of sample_dist:
    sample_100 = stats.norm.rvs(10, 2, size=n4)
    yvar_100[j] = np.var(sample_100, ddof=1)
mean_10 = np.mean(yvar_10)
variance_10 = np.var(yvar_10, ddof=1)
print("Mean of sample variance distribution w/ n=10 is :", mean_10)
print("Variance of sample variance distribution w/ n=10 is :", variance_10)
(이하 생략)
```

구분		n=10	n=20	n=30	n=100
평균	표본분산	3.99581	3.9925	4.00163	3.99624
	모집단	4	4	4	4
분산	표본분산	3.5946	1.70038	1.12352	0.3206
	모집단	3.5555	1.6842	1.1034	0.3232

```
Mean of sample variance distribution w/ n=10 is : 3.9958174541500657
Variance of sample variance distribution w/ n=10 is : 3.5946024817581104
Mean of sample variance distribution w/ n=20 is : 3.992505450439514
Variance of sample variance distribution w/ n=20 is : 1.7003889046087846
Mean of sample variance distribution w/ n=30 is : 4.0016336720185395
Variance of sample variance distribution w/ n=30 is : 1.1235229917302787
Mean of sample variance distribution w/ n=100 is : 3.9962411485858844
Variance of sample variance distribution w/ n=100 is : 0.3206068024401987
```