

母集団の比較

```
In [1]: import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from scipy import stats
```

```
In [2]: # 母集団 1 からのサンプリング
m1=12          # 母平均の設定
s1=3          # 母標準偏差の設定
n1=20         # 母集団から取り出すサンプル数
nsample1=np.random.normal(m1, s1, size=n1) # 正規母集団  $N(m, s^2)$  から取り出した n 個のサンプル
```

```
In [3]: nsample1
```

```
Out[3]: array([ 7.99628937,  8.84060215, 14.65274013, 10.71206459, 17.03714153,
 13.61372559, 13.28163501, 10.69149968, 10.59663027, 17.32715272,
 12.80666847, 14.43002921, 13.49948427, 11.66614713, 10.45081065,
 13.65935175, 13.00266524, 14.40782237, 13.65109749, 16.51574329])
```

```
In [4]: # 母集団 2 からのサンプリング
normal = stats.norm() # 標準正規分布を準備
m2=10          # 母平均の設定
s2=2          # 母標準偏差の設定
n2=15         # 母集団から取り出すサンプル数
nsample2=np.random.normal(m2, s2, size=n2) # 正規母集団  $N(m, s^2)$  から取り出した n 個のサンプル
```

```
In [5]: nsample2
```

```
Out[5]: array([10.90707057, 10.26386215,  9.73551634,  8.94085728,  8.39703177,
  9.85992517, 10.04814588,  8.39912428,  8.62741898,  8.09460722,
 12.24666215,  9.00701112,  9.85086212, 11.95078163, 11.54899302])
```

標本平均の差

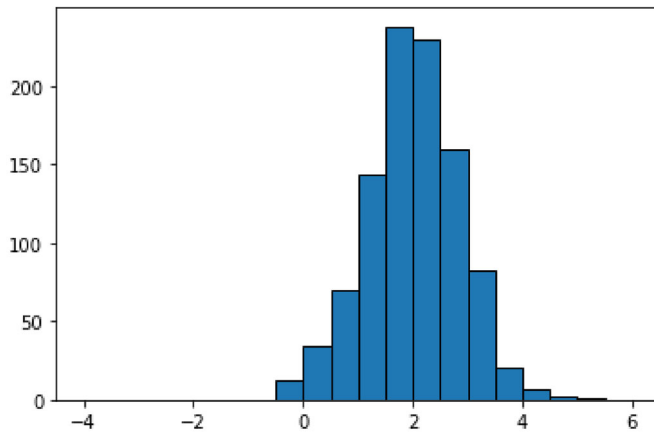
```
In [6]: # 標本平均の差
D=np.mean(nsample1)-np.mean(nsample2)
D
```

```
Out[6]: 3.0834404007267207
```

```
In [7]: # 同様の標本抽出を T 回繰り返して、標本平均の差 D を収集する
T=1000      # トライアルの回数
D_list=[]
for i in range(T):
    nsample1=np.random.normal(m1, s1, size=n1) # 正規母集団  $N(m1, s1^2)$  から取り出した n1 個のサンプル
    nsample2=np.random.normal(m2, s2, size=n2) # 正規母集団  $N(m2, s2^2)$  から取り出した n2 個のサンプル
    D=np.mean(nsample1)-np.mean(nsample2)
    D_list = D_list+[D]
# D_list
```

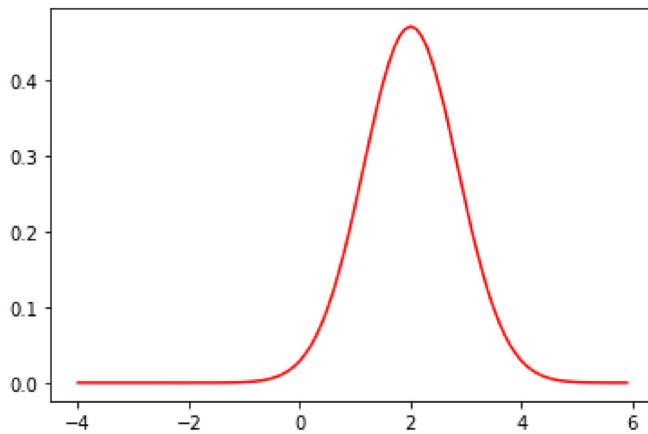
```
In [8]: plt.hist(D_list, range=(-4, 6), bins=20, ec='k')
```

```
Out[8]: (array([ 0.,  0.,  0.,  0.,  0.,  0.,  0., 12., 34., 70., 143.,
          238., 230., 160., 82., 21.,  7.,  2.,  1.,  0.]),
array([-4. , -3.5, -3. , -2.5, -2. , -1.5, -1. , -0.5,  0. ,  0.5,  1. ,
          1.5,  2. ,  2.5,  3. ,  3.5,  4. ,  4.5,  5. ,  5.5,  6. ]),
<BarContainer object of 20 artists>)
```



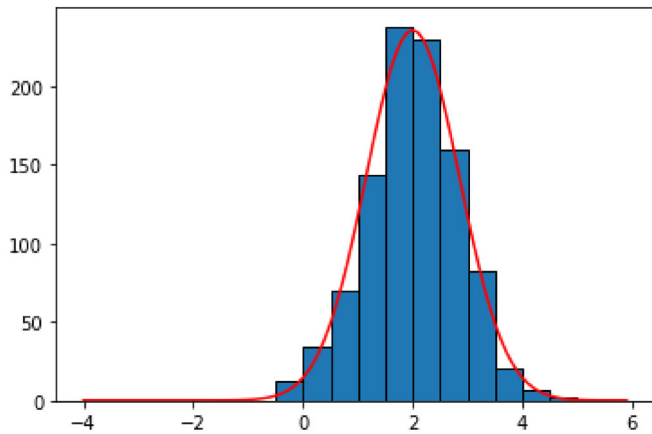
```
In [9]: # D の理論分布
rvD=stats.norm(m1-m2, np.sqrt(s1**2/n1+s2**2/n2))
xs=np.arange(-4, 6, 0.1)
plt.plot(xs, rvD.pdf(xs), color='r')
```

```
Out[9]: [<matplotlib.lines.Line2D at 0x2553c7c1a90>]
```



```
In [10]: plt.hist(D_list, range=(-4, 6), bins=20, ec='k')
plt.plot(xs, rvD.pdf(xs)*T/2, color='r')
```

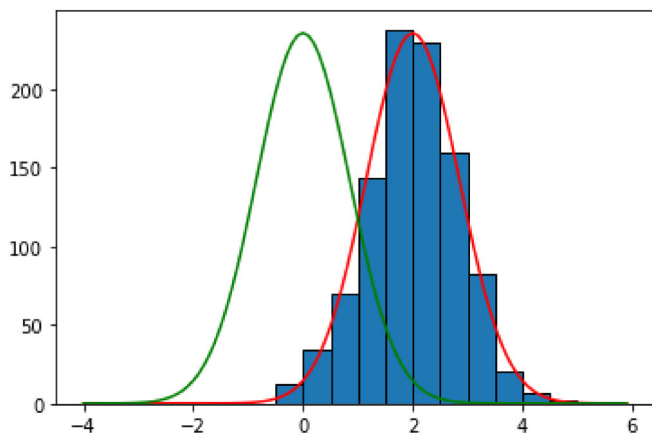
Out[10]: [matplotlib.lines.Line2D at 0x2553c8235b0]



母平均の差の検定

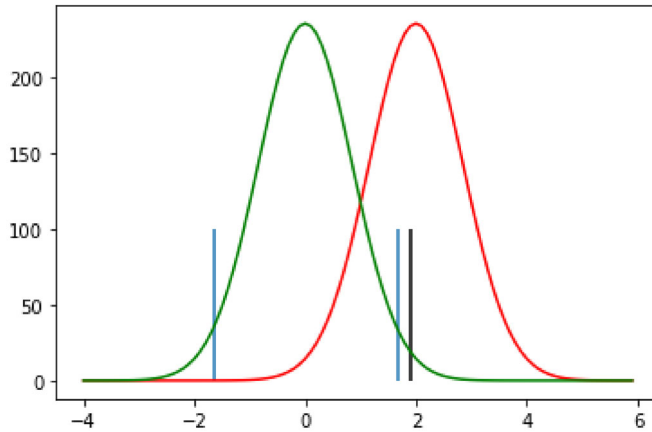
```
In [11]: # 仮説 m1=m2 の下での D の理論分布を重ねる
rvH=stats.norm(0, np.sqrt(s1**2/n1+s2**2/n2))
plt.hist(D_list, range=(-4, 6), bins=20, ec='k')
plt.plot(xs, rvD.pdf(xs)*T/2, color='r')
plt.plot(xs, rvH.pdf(xs)*T/2, color='green')
```

Out[11]: [matplotlib.lines.Line2D at 0x2553c8e5e80]



```
In [12]: # 有意水準 alpha の棄却域
alpha = 0.05
rvH.isf(alpha/2) # 上側 alpha/2 点 = 両側 alpha 点
rvH.isf(1-alpha/2) # 下側 alpha/2 点
plt.plot(xs, rvD.pdf(xs)*T/2, color='r')
plt.plot(xs, rvH.pdf(xs)*T/2, color='green')
plt.vlines(rvH.isf(alpha/2), 0, 100)
plt.vlines(rvH.isf(1-alpha/2), 0, 100)
plt.vlines(D, 0, 100, color='k')
```

Out[12]: <matplotlib.collections.LineCollection at 0x2553c943af0>



```
In [13]: beta=rvD.cdf(rvH.isf(alpha/2))
beta
```

Out[13]: 0.34364564293890904

In []: