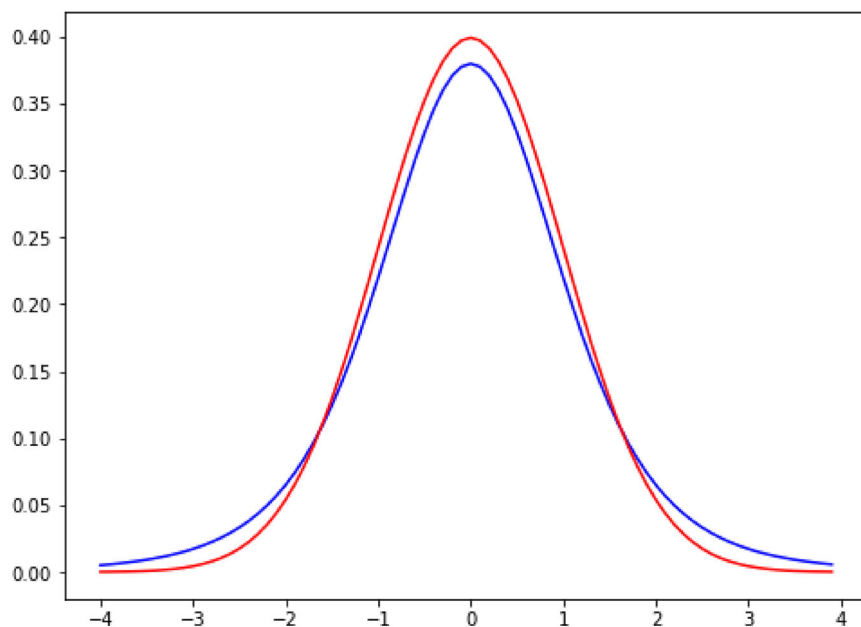


```
In [1]: import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from scipy import stats
```

```
In [2]: n=5          # 自由度
rvT=stats.t(n)     # 自由度 n の t 分布
```

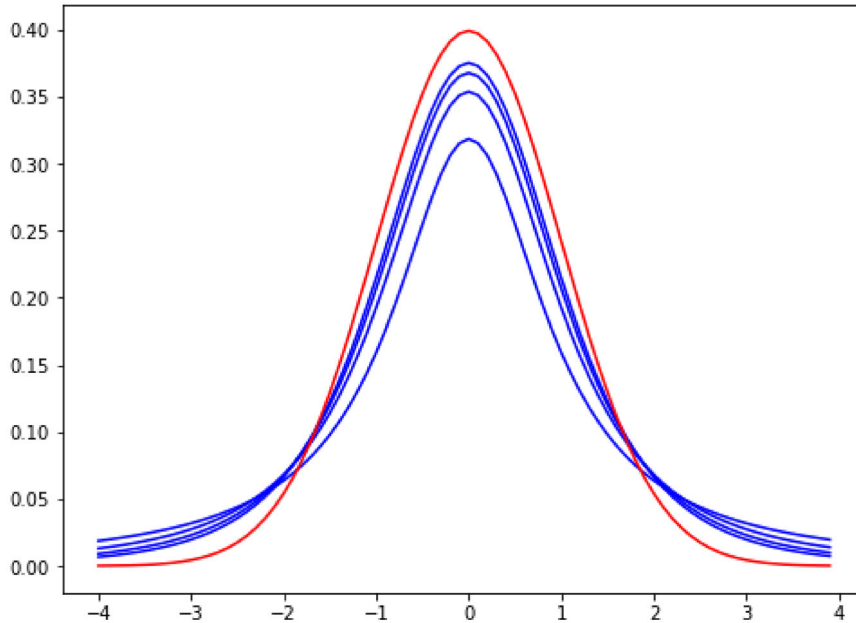
```
In [3]: # 密度関数の描画
fig = plt.figure(figsize=(8, 6))
ax = fig.add_subplot(1, 1, 1)
xs=np.arange(-4, 4, 0.1)
ax.plot(xs, rvT.pdf(xs), color='blue')
# 比較のため N(0, 1)
rvN = stats.norm()
ax.plot(xs, rvN.pdf(xs), color='red')
```

Out[3]: [<matplotlib.lines.Line2D at 0x1eaa74d06a0>]



```
In [4]: # 密度関数の描画 (n=1, 2, 3, 4, 5, ...)
fig = plt.figure(figsize=(8, 6))
ax = fig.add_subplot(1, 1, 1)
xs=np.arange(-4, 4, 0.1)
for n in range(1, 5):
    ax.plot(xs, stats.t(n).pdf(xs), color='blue')
# 比較のため N(0, 1)
rvN = stats.norm()
ax.plot(xs, rvN.pdf(xs), color='red')
```

Out[4]: [matplotlib.lines.Line2D at 0x1eaa75c89d0]

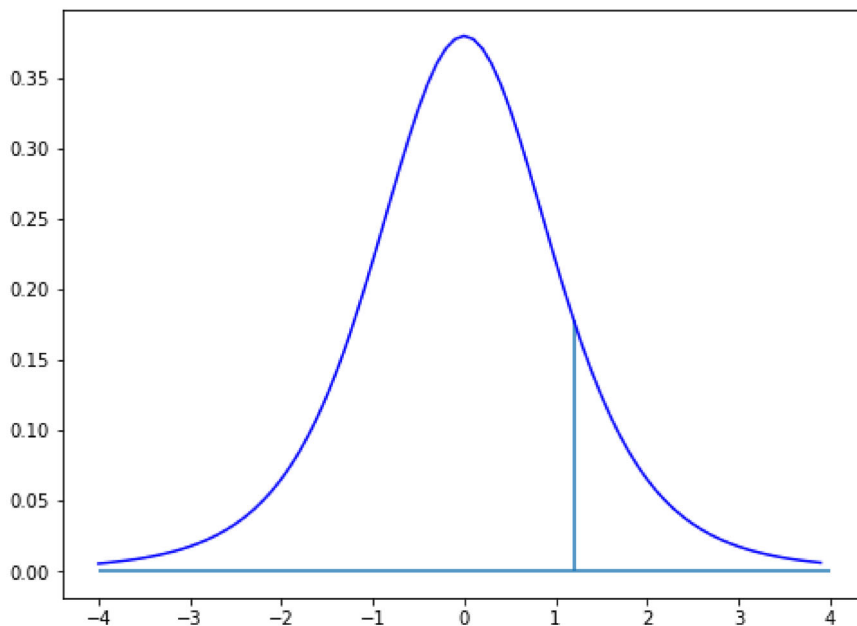


```
In [5]: # 分布関数 (確率計算 P(T<= xx)=rvT.cdf(xx))
n = 5 # 自由度
x = 1.2
stats.t(n).cdf(x)
```

Out[5]: 0.8580544716469489

```
In [6]: plt.figure(figsize=(8,6))
plt.plot(xs, rvT.pdf(xs), color='blue')
plt.vlines(x, 0, rvT.pdf(x))
plt.hlines(0, -4, 4)
```

Out[6]: <matplotlib.collections.LineCollection at 0x1eaa763f790>



```
In [7]: # 上側 alpha 点
n = 5
alpha = 0.05
stats.t(n).isf(alpha)
```

Out[7]: 2.0150483726691575

```
In [8]: # 母分散未知の場合、標本平均の分布のシミュレーション
normal = stats.norm() # 標準正規分布を準備
m=12 # 母平均の設定
s=3 # 母標準偏差の設定
n=8 # 母集団から取り出すサンプル数
nsample=np.random.normal(m,s,size=n) # 正規母集団 N(m, s^2) から取り出した n 個のサンプル
sample_mean = np.mean(nsample) # それらの標本平均
sample_unb_var = np.var(nsample, ddof=1) # それらの不偏分散 ddof=1 が必要
sample_mean, sample_unb_var
```

Out[8]: (11.951728806196112, 10.3640034318335)

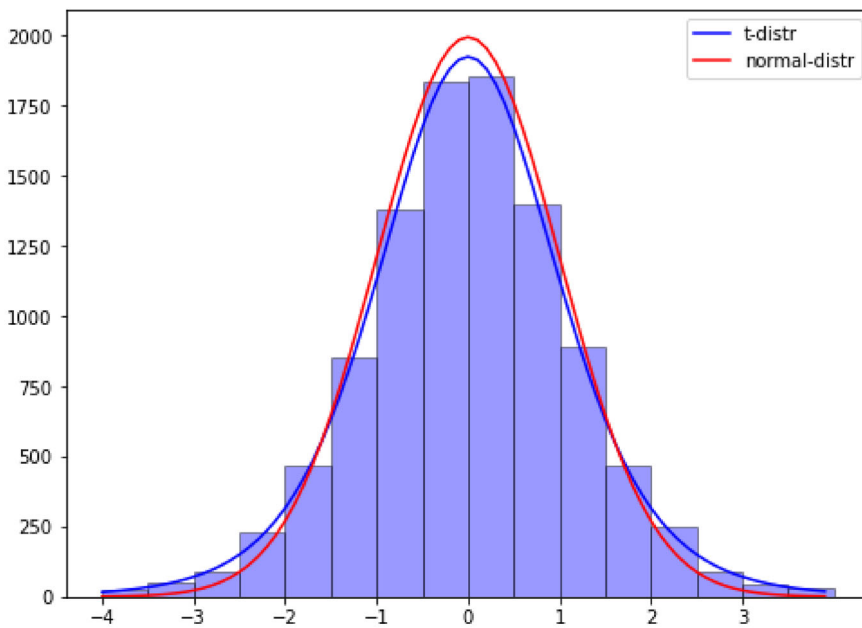
```
In [9]: # 理論では (sample_mean - population_mean)/sqrt(sample_unb_var/n) が 自由度 n-1 の t 分布に従う
t = (sample_mean - m)/np.sqrt(sample_unb_var/n)
t
```

Out[9]: -0.04241009633879092

```
In [10]: # 上記の t を多数 (T 個) 収集して、その分布を調べる
T=10000
t_list=[]
for i in range(T):
    nsample=np.random.normal(m, s, size=n) # 正規母集団 N(m, s^2) から取り出した n 個のサンプル
    sample_mean = np.mean(nsample)
    sample_unb_var = np.var(nsample, ddof=1)
    t = (sample_mean - m)/np.sqrt(sample_unb_var/n)
    t_list = t_list+[t]
# t_list
```

```
In [11]: plt.figure(figsize=(8,6))
F, x, _ = plt.hist(t_list,
    range=(-4,4),          # 幅を指定
    bins=16,              # 階級数を指定
    color='blue',         # 色を指定
    alpha=0.4,           # 色の濃さ (0~1)
    ec='k')              # 棒に枠線つける (k=black)
plt.xticks(np.arange(-4,4,1)) # x軸の目盛
plt.plot(xs, (T/2)*stats.t(n-1).pdf(xs), color='blue', label='t-distr') # 密度関数のフィッティング
plt.plot(xs, (T/2)*stats.norm().pdf(xs), color='red', label='normal-distr') # N(0,1) の誤使用
plt.legend()
```

Out[11]: <matplotlib.legend.Legend at 0x1eaa7652d30>



In []: