MEM-264 Applied Statistics

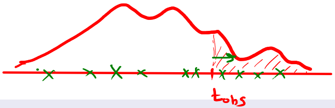
Department of Mathematics and Applied Mathematics, University of Crete

Costas Smaragdakis (kesmarag@uoc.gr)

7th lecture - 01-03-2021

q - simulations

p-value $\approx \frac{4}{10} = 0.4$



$H_0: \theta = 0$

$H_1: \theta > 0$

The Monte-Carlo (MC) approach

Let T be a test statistics. We employ a **MC** approach if $f_T(t)$ is known but its integration is complicated.

- Let t_{obs} is our observation from the data.
- We draw $q >> 1$ independent realizations $\{t_1, \ldots, t_q\}$ of T. (under the Null Hypothesis)

Then we can estimate the p-value as follows:

$$\text{p} - \text{value} \approx \frac{1}{q} \sum_{j=1}^{q} 1\{t_j \in \text{in the area toward the extreme values}\}$$

Bernoulli : Be(θ)

Head : θ (1)

coin

tail : 1-θ (0)

$$X = \begin{cases} 1, & \text{with probability } \theta \\ 0, & \text{with probability } 1-\theta \end{cases}$$

$\mathcal{X} = \{0, 1\}$

$f(x;\theta) = \theta^x (1-\theta)^{1-x}$

if $x=1$ then $f(1;\theta)=\theta$

$x=0$ then $f(0;\theta)=1-\theta$

Test case 1

fair coin

$\operatorname{Var} Y_2 = 2$

$$X = WY_1 + (1-W)Y_2,\ W \sim \text{Be}(0.5),\ Y_1 \sim \mathcal{N}(0,1),\ Y_2 \sim \mathcal{N}(\epsilon, 2)$$

$$X = \begin{cases} Y_1 & \text{if Head} \\ Y_2 & \text{if tail} \end{cases}$$

$$H_0 : \epsilon = 0 \quad \text{vs} \quad H_1 : \epsilon > 0$$

- Let's assume that the (hidden) value of ϵ is 2.

$Y_2 \sim \mathcal{N}(2, 2)$

Simulation $n=6$ observations

$\mathcal{N}$ $\mathcal{B}e$

```
from scipy.stats import norm, bernoulli
import numpy as np
import matplotlib.pyplot as plt
np.random.seed(1000)
n = 6 # We will generate 6 realization of X
Epsilon = 2 # The true value of epsilon
W = bernoulli(0.5)
Y1 = norm(0, np.sqrt(1))
Y2 = norm(Epsilon, np.sqrt(2*1))
weights = W.rvs(n) # n observation from W
X = weights * Y1.rvs(n) + (1 - weights) * Y2.rvs(n)
plt.plot(X, 'o')
plt.show()
```

σ

We can define the following test statistics

$T = \overline{X}_n$

Realizations of X

-0.1074373	1.42588563	0.5950355	2.14593679	0.66728131	2.99799973

$H_0 : \varepsilon = 0$ vs $H_1 : \varepsilon > 0$

The Monte-Carlo (MC) approach

.rvs(n) → generates n random observations


```
from scipy.stats import norm, bernoulli
import numpy as np
import matplotlib.pyplot as plt
q = 1000 # 1000 simulations
n = 6
Epsilon = 0 # under the null hypothesis (H0)
t_obs = np.mean([ -0.1074373, 1.42588563, 0.5950355, 2.14593679, 0.66728131, 2.99799973])

k = 0
for i in range(q):
  W = bernoulli(0.5)
  Y1 = norm(0, 1)
  Y2 = norm(Epsilon, np.sqrt(2))
  weights = W.rvs(n)
  X = weights * Y1.rvs(n) + (1 - weights) * Y2.rvs(n)
  t = X.mean()
  if t > t_obs:
    k += 1
p_value = k/q
```

$t_{obs} = \bar{X}_6$

σ

$\#\ \bar{X}_6$ under the H_0

0.003 ← We can reject the H_0

Test case 2

$$X = W^{(1)}Y_1^{(1)} + (1 - W^{(1)})Y_2^{(1)},\ W^{(1)} \sim \text{Be}(0.5),\ Y_1^{(1)} \sim \mathcal{N}(0,1),\ Y_2^{(1)} \sim \mathcal{N}(\mu+\epsilon, 2)$$

$$Z = W^{(2)}Y_1^{(2)} + (1 - W^{(2)})Y_2^{(2)},\ W^{(2)} \sim \text{Be}(0.25),\ Y_1^{(2)} \sim \mathcal{N}(0,1),\ Y_2^{(2)} \sim \mathcal{N}(\mu, 2)$$

Head 0.25

tail 0.75

$\mathbb{E}\{Z\} = 0.75\mu = 3/4\,\mu$

$\bar{Z}_n \rightarrow \mu = \dfrac{\bar{Z}_n}{0.75}$

- Let's assume that μ is unknown !!

$$H_0 : \epsilon = 0 \quad \text{vs} \quad H_1 : \epsilon > 0$$

Bootstrap test !!

$T = \alpha \bar{X}_n - \beta \bar{Z}_n = \alpha \frac{1}{2}(0 + \mu + \varepsilon) - \beta \frac{3}{4}\mu$

Perform the MC approach using an estimate $\hat{\mu}$ of μ

if $\alpha = 2$, $\beta = 4/3$ then $T = \mu + \varepsilon - \mu = \varepsilon$

$$\hat{\mu} = \bar{z}_n/0.75 \quad \text{(why?)}, \quad T = 2\bar{X}_n - 4\bar{Z}_n/3$$

Simulation

```
        from scipy.stats import norm, bernoulli
        import numpy as np
        import matplotlib.pyplot as plt
        np.random.seed(1000)
        n = 6 # We will generate 6 realization of X
        Mu = 0 # μ = 0
        Epsilon = 2 # The true value of epsilon
W^(1)   W1 = bernoulli(0.5)
Y_1^(1) Y1_1 = norm(0, 1)
Y_2^(1) Y1_2 = norm(Mu + Epsilon, np.sqrt(2))
        weights = W1.rvs(n)
        X = weights * Y1_1.rvs(n) + (1 - weights) * Y1_2.rvs(n)
W^(2)   W2 = bernoulli(0.25)
Y_1^(2) Y2_1 = norm(0, 1)
Y_2^(2) Y2_2 = norm(Mu, np.sqrt(2))
        weights = W2.rvs(n)
        Z = weights * Y2_1.rvs(n) + (1 - weights) * Y2_2.rvs(n)
        plt.plot(X, 'o')
        plt.show()
```

x	-0.1074373	1.42588563	0.5950355	2.14593679	0.66728131	2.99799973	
z	-1.16381427	0.91938754	-0.09797855	0.11767599	-0.46234131	-0.064767	$\to \hat{t}$

```
from scipy.stats import norm, bernoulli
import numpy as np
import matplotlib.pyplot as plt
q = 1000 # 1000 simulations
n = 6
Epsilon = 0 # under the null hypothesis (H0)
Mu_hat = np.mean([-1.16381427, 0.91938754, -0.09797855, 0.11767599, -0.46234131, -0.064767])/0.75
t_obs = 2*np.mean([-0.1074373, 1.42588563, 0.5950355, 2.14593679, 0.66728131, 2.99799973]) - Mu_hat
k = 0
for i in range(q):
  W1 = bernoulli(0.5)
  Y1_1 = norm(0, 1)
  Y1_2 = norm(Mu_hat + Epsilon, np.sqrt(2))
  weights = W1.rvs(n)
  X = weights * Y1_1.rvs(n) + (1 - weights) * Y1_2.rvs(n)
  W2 = bernoulli(0.25)
  Y2_1 = norm(0, 1)
  Y2_2 = norm(Mu_hat, np.sqrt(2))
  weights = W2.rvs(n)
  Z = weights * Y2_1.rvs(n) + (1 - weights) * Y2_2.rvs(n)
  t = 2*X.mean() - 4/3*Z.mean()
  if t > t_obs:
    k += 1
p_value = k/q
```

$4/3\,\bar{Z}_6$

$\hat{\mu} = \bar{Z}_6/0.75$

$t_{obs} = 2\bar{X}_n - 4/3\,\bar{Z}_6$

0.009 We can reject the null hypothesis.