Professional Maters in

Applied Statistics and Data Science

(PM-ASDS)

An Assignment on

Simple Random Sampling and Stratified Sampling

Submitted to:

Mohammad Alamgir Kabir, PhD
Professor & Chairman,
Department of Statistics, JU

Submitted by:
Mohammad Saiduzzaman Sayed
ID: 20215063
Batch: 5th
Sec: A
Department of Statistics, JU

Course Name: Sampling Methodology
Course Code: PM-ASDS02

Example of A2_Leture on simple random sampling

Generating random number list in Python:

```
import pandas as pd
import numpy as np

# user_input
mean_input = float(input("Enter mean                : "))
std_input = float(input("Enter Standard Deviation  : "))

size_column = int(input("Variable                  : "))
size_row = int(input("Observation               : "))

#random number generate using normal distribution (mean, std, size(row, column))
data1 = np.random.normal(mean_input, std_input, size=(size_row, size_column))

data1 = pd.DataFrame(data1)
print(data1)
```

Output:

Enter mean: 5
Enter Standard Deviation: 2.5
Variable: 3
Observation: 10

	0	1	2
0	3.139810	3.193322	8.221730
1	2.354762	-0.127788	5.441622
2	7.047250	5.354266	8.263851
3	4.943709	-0.111039	6.828607
4	9.319299	2.819445	3.966945

Example-1:
Estimation of population characteristics in SRS:
Example: Suppose you want to open a ‘fast food’ shop at Jamuna Future Park. There are 250 fast food shops currently operating in the mall. As a beginner, an idea about monthly sell of the existing shops would be useful. Under this circumstance, you select 10 shops randomly and the total sells (in ‘000 TK) over the last month of the sampled shops are as follows:
75 95 105 99 110 97 85 90 100 101
- Find an estimate of the monthly sells per shop
- Also estimate the standard error and relative errors of the sample mean

Solution:
```
import numpy as np
import scipy.stats as stat

sampled_shop = [75, 95, 105, 99, 110, 97, 85, 90, 100, 101]

#Estimate of the monthly sells per shop:
sampled_shop_mean = np.mean(sampled_shop)
print("Estimate of the monthly sells per shop: x̄ = ", sampled_shop_mean)
#The average monthly sells of fast food shops at Jamuna Future Park is TK 95,700

#Standard error of the sample mean:
SEM = stat.sem(sampled_shop)
print("Standard error of the sample mean: SE(x̄)  = ", "%.2f" % SEM)

#Relative errors of the sample mean:
CV = SEM/sampled_shop_mean
print("Relative errors of the sample mean: CV(x̄) = ", "%.3f" % CV)

#Average monthly sells
Average = CV * 100
print("The average monthly sells may vary", "%.1f" % Average, "%")
```

Output:
```
Estimate of the monthly sells per shop: x̄ =  95.7
Standard error of the sample mean: SE(x̄)  =  3.21
Relative errors of the sample mean: CV(x̄) =  0.034
The average monthly sells may vary 3.4
```

Example-2:
Suppose a social scientist is interested to know the proportion of women watch Indian channels. She selects 1,500 women from Lalmatiya residential area randomly and finds that 975 regularly watch the Indian channels. Suppose the total number of women residing in Lalmatiya area is 1 million.

Solution:

```
import numpy as np

r = 975     #number of cases posses the attribute of interest
n = 1500    #sample size
N = 1000000 #population size

#Estimation of population proportion:
p = r/n
print("Estimation of population proportion: p = ", p)

#Sample variance of P:
variance = (p*(1-p) / (n-1)) * (1- (n/N))
print("Sample variance of P: v(p) = ", "%.5f" % variance)

#Standard error:
SE = np.sqrt(variance)
print("Standard error: SE(p) = ", "%.3f" % SE)
```

Output:

```
Estimation of population proportion: p =  0.65
Sample variance of P: v(p) =  0.00015
Standard error: SE(p) =  0.012
```

Example-3:
We want to estimate the mean mobile bill per month of the 35 PM-ASDS students of Section B. We wish to estimate the true mean to within 10TK with 95% confidence. The standard deviation of monthly mobile bill is around 50TK. What is the required sample size?

Solution:

```
N = 35
z = 1.96  #(1.96 for 5% level of significance)
σ = 50    #population standard deviation
d = 10    #width of the interval desired (precision)

#Estimated sample size:
n_o = z**2 * σ**2/ d**2
print("Estimated sample size: n_o = ", n_o)

#Required sample size:
n = n_o / (1 + n_o/N)
print("Required sample size: n = ", round(n))
```

Output:

```
Estimated sample size: n_o =  96.04
Required sample size: n =  26
```

Example-4: To estimate the proportion of 30 Male students who smoke, what sample size is required to achieve a 95% confidence interval of width ±5% (that is to be within 5% of the true value)? A study some years ago found that approximately 30% were smokers.

Solution:

```
N = 30

z = 1.96    #1.96 for 5 percent level of significance
p = 0.3     #estimated prevalence; (0.5 if unknown)
q = 1 - p
d = 0.05    #precision desired (usually consider 0.05)

#Estimated sample size:
n_o = z**2 * p * q / d**2
print("Estimated sample size: n_o = ", "%.2f" %  n_o)

#Required sample size:
```

```
n = nₒ / (1 + nₒ/N)
print("Required sample size: n = ", int(n))
```

Output:

```
Estimated sample size: nₒ =  322.69
Required sample size: n =  27
```

Asssignment from A2_Leture on simple random sampling

1. Draw a simple random sample of size 20 from a population of size 105 by using random number table.

Solution:

```
import numpy as np

#These value taken from given table in assignment with rejection method
random_sample = [73, 78, 44, 84, 55, 65, 74, 75, 77, 68, 64, 54, 33, 43, 86, 56, 57, 53, 35, 76]

N = 105
n = len(random_sample)
f = n / N  #𝒇 is known as sampling fraction

# ȳ :
y = np.mean(random_sample)
print("ȳ = ", y)

# s² :
s_square = sum((x - y)**2 for x in random_sample) / (n-1)
print("s² = ", "%.2f" % s_square)

# V(ȳ) :
V = (1- f) * s_square/n
print("V(ȳ) = ", "%.3f" % V)
```

Output:

```
ȳ =  62.5
s² =  248.68
V(ȳ) =  10.066
```

2.If we want to draw a simple random sample from a population of 4000items, how large the sample do we need if desire to estimate percent defective within 2% of the true value with 95% confidence level.

Solution:

```
N = 4000
p = 0.5    #estimated prevalence; (0.5 if unknown)
q = 1 - p
z = 1.96   #1.96 for 5 percent level of significance
d = 0.02   #precision desired

#Estimated sample size:
nₒ = z**2 * p * q / d**2
print("Estimated sample size: nₒ = ", "%.2f" %  nₒ)

#Required sample size:
n = nₒ / (1 + (nₒ/N))
print("Required sample size: n = ", int(n))
```

Output:

Estimated sample size: n_o = 2401.00
Required sample size: n = 1500

3. Using the sample variance obtained in problem 1 to estimate the populationvariance, calculate the sample size necessary to achieve it with 95%confidence in a future survey of the same population for estimatingpopulation mean.

Solution:

```
N = 105
p = 0.5    #estimated prevalence; (0.5 if unknown)
q = 1 - p
z = 1.96   #1.96 for 5 percent level of significance
d = 0.1   #precision desired

#Estimated sample size:
nₒ = z**2 * p * q / d**2
print("Estimated sample size: nₒ = ", "%.2f" %  nₒ)

#Required sample size:
n = nₒ / (1 + (nₒ/N))
print("Required sample size: n = ", int(n))
```

Output:

Estimated sample size: n_o = 96.04
Required sample size: n = 50

Example of A3_Lecture on stratified sampling

Example-1: Suppose you want to open a ‘fast food’ shop at Jamuna Future Park. There are 250 fast food shops currently operating in the mall of which 150 are small and 100 are large. As a beginner, an idea about monthly sell of the existing shops would be useful. Under this circumstance, 15 shops are randomly selected from two strata and the total sells (in ‘000 TK) over the last month of the sampled shops are as follows:

Small: 75 90 70 90 80 85 90 75 70

Large: 110 120 150 105 115 125

(i) Find an estimate of the monthly sells per shop

(ii) Also estimate the standard error of the sample mean

Solution:

```
import numpy as np

small = [75, 90, 70, 90, 80, 85, 90, 75, 70]
large = [110, 120, 150, 105, 115, 125]

N = 250
small_Ni = 150
large_Ni = 100
small_ni = len(small)
large_ni = len(large)

small_Wi = small_Ni / N
print("Small Wi = ", small_Wi)

small_mean = np.mean(small)
print("Small X̄i = ", "%.1f" % small_mean)

small_s_square = (1/(small_ni-1)) * sum((x - small_mean)**2 for x in small)
print( "Small si square = ", "%.1f" %small_s_square)

large_Wi = large_Ni / N
print("Large Wi = ", large_Wi)

large_mean = np.mean(large)
print("Large X̄i = ", "%.1f" % large_mean)

large_s_square = (1/(large_ni-1)) * sum((x - large_mean)**2 for x in large)
print( "Large si square = ", "%.1f" % large_s_square)

xofst = (small_Wi * small_mean) + (large_Wi * large_mean)
print(  "x̄_st = ", "%.1f" % xofst)

small_v = small_Wi**2 * (1 - (small_ni/small_Ni)) * (small_s_square / small_ni)
```

```
large_v = large_Wi**2 * (1 - (large_ni/large_Ni)) * (large_s_square / large_ni)
v = small_v + large_v
print("v(x_st) = ", "%.1f" % v)

SE = np.sqrt(v)
print ("SE(x̄st) = ", "%.2f" % SE)
```

Output:

Small Wi = 0.6
Small X̅i = 80.6
Small si square = 71.5

Large Wi = 0.4
Large X̅i = 120.8
Large si square = 254.2

x̄_st = 96.7

v(x_st) = 9.1
SE($\bar{x}_{st}$) = 3.01

Example-2: A sample survey of 800 men and 700 women was conducted to assess their knowledge on HIV/AIDS. The survey findings reveal that 600 men and 400 women have correct knowledge on the issue. Suppose there were 3,500 men and 3,000 women in the study area.

(i) What proportion of study respondents have accurate knowledge on HIV/AIDS?

(ii) Also estimate the standard error of the estimate.

Solution:

```
import numpy as np

N = 6500

men_Ni = 3500

men_ni = 800

men_ri = 600


women_Ni = 3000

women_ni = 700
```

```
women_ri = 400


men_Wi = men_Ni / N

print("men_Wi = ", "%.2f" % men_Wi)

women_Wi = women_Ni / N

print("women_Wi = ", "%.2f" % women_Wi)


men_pi = men_ri / men_ni

print("men_pi = ", "%.2f" % men_pi)

women_pi = women_ri / women_ni

print("women_pi = ", "%.2f" % women_pi)

p_st = (men_Wi * men_pi) + (women_Wi * women_pi)

print("p_st = ", "%.2f" % p_st)

men_v = men_Wi**2 * (1-(men_ni/men_Ni)) * ((men_pi-men_pi**2) / (men_ni - 1))

women_v = women_Wi**2 * (1-(women_ni/women_Ni)) * ((women_pi-women_pi**2) /
(women_ni - 1))

v = men_v + women_v

print("v(p_st) = ", "%.4f" % v)

SE = np.sqrt(v)

print("SE(p_st) = ", "%.2f" % SE)
```

output:

```
men_Wi =  0.54
women_Wi =  0.46

men_pi =  0.75
women_pi =  0.57


p_st =  0.67
v(p_st) =  0.0001
SE(p_st) =  0.01
```

Example-3: Consider a population of 30 families living in two city blocks, block A and block B. Block A consists of 20 families with 120 members and block B consists of 10 families with 50 members. Table below shows the distribution of the family size of these two blocks. We treat the blocks as stratum.

Table given in A3_Lecture on stratified sampling

▪ Compute the stratum mean, stratum variance.

▪ Let us choose 8 families from stratum I and 4 families from stratum II at random. Using the sample values,

(i) Estimate the average family size.

(ii) Estimate the variance of the sample mean.

Solution:

```
import numpy as np

import statistics


#Stratum 1 (Block A)

Y1j = [5, 7, 3, 9, 8, 6, 5, 6, 6, 4, 6, 9, 5, 4, 8, 7, 6, 4, 6, 6]


Y1_mean = np.mean(Y1j)

Y1_variance = statistics.variance(Y1j)


print("Mean of Stratum 1 = ", Y1_mean)

print("Variance of Stratum 1 = ", "%.2f"% Y1_variance)


#Stratum 2 (Block B)

Y2j = [3,9,2,6,11,4,8,2,3,2]


Y2_mean = np.mean(Y2j)

Y2_variance = statistics.variance(Y2j)


print("Mean of Stratum 2 = ", Y2_mean)

print("Variance of Stratum 2 = ", "%.2f"% Y2_variance)
```

```
#Selected sample
#Stratum 1
sample_Y1j = [5,8,6,6,4,7,8,4]

y1_mean = np.mean(sample_Y1j)
y1_variance = statistics.variance(sample_Y1j)

print("Sample mean of Stratum 1 = ", y1_mean)
print("Sample variance of Stratum 1 = ", "%.2f"% y1_variance)

#Stratum 2
sample_Y2j = [11,6,2,4]

y2_mean = np.mean(sample_Y2j)
y2_variance = statistics.variance(sample_Y2j)

print("Sample mean of Stratum 2 = ", y2_mean)
print("Sample variance of Stratum 2 = ", "%.2f"% y2_variance)

#variance of the sample mean
N = 30
N1 = len(Y1j)
N2 = len(Y2j)
n1 = len(sample_Y1j)
n2 = len(sample_Y2j)
s1 = y1_variance
s2 = y2_variance

stra1 = N1 * (N1 - n1) * (s1**2/n1)
```

```
stra2 = N2 * (N2 - n2) * (s2**2/n2)

var = 1/N**2 * (stra1+stra2)

print("variance of the sample mean = ", "%.3f"% var)
```

output:

```
Mean of Stratum 1 =  6.0
Variance of Stratum 1 =  2.74

Mean of Stratum 2 =  5.0
Variance of Stratum 2 =  10.89

Sample mean of Stratum 1 =  6.0
Sample variance of Stratum 1 =  2.57

Sample mean of Stratum 2 =  5.75
Sample variance of Stratum 2 =  14.92

variance of the sample mean =  3.929
```