

Lab09: Sampling

Handout date: Wednesday, day before *Halloween*, 2020

Due date: Friday, November 6, 2020 into the link **SUBMITLAB09** on *eLearning*.

This lab counts 4 % toward your total grade


Part I: Central Limit Theorem

Task 1: Evaluating the distribution of the sample mean (0.9 points)

Following the example on pages 273-275 in Burt, Barber and Rigby. Evaluate the distribution of the sample mean, which itself is a random variable.

You will fully enumerate all possible samples of length 4 with replacement from a population **pop** with 10 data values. For each possible sample its mean is calculated and the distribution of all sample means is evaluated. Use the code below:

```
1 popSd <- function(x) sqrt(sum((x-mean(x))^2)/length(x))
2
3 pop <- c(1,2,3,4,5,5,7,6,8,9)
4 mean(pop); popSd(pop)
5
6 quadruples <- expand.grid(pop,pop,pop,pop)
7 sampleMeans <- rowMeans(quadruples)
8
9 hist(sampleMeans, breaks=seq(1,9,length=32))
10 axis(1,at=1:9)
11 mean(sampleMeans); popSd(sampleMeans)
```

[a] Give the expectation μ and standard deviation σ of the population. Why do you need to use the function **popSd**() here rather than the standard  function **sd**()? Hint: Check the denominator of the function. (0.2 points)

```
mean(pop); popSd(pop)
```

```
[1] 5
```

```
[1] 2.44949
```

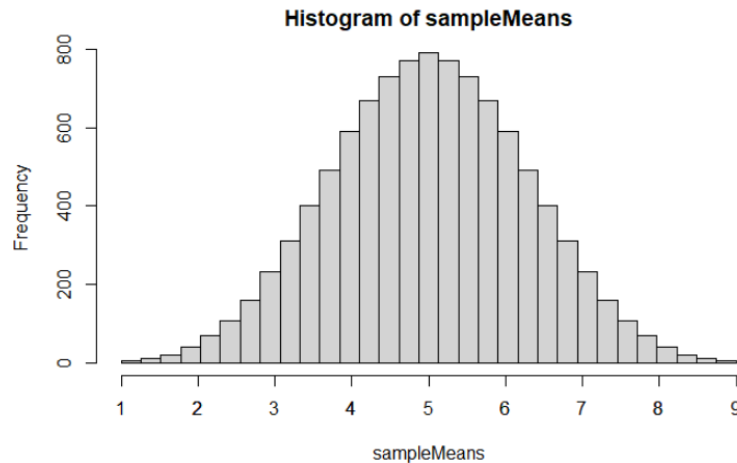
Comment: Because we have the full enumeration of the population, we don't lose one degree of freedom. Thus, we need to use n rather than $n - 1$ as the denominator.

[b] What is the function **expand.grid**() doing and how does its generated sample differ from the other enumeration function **combn**()? (0.2 points)

Comment: **expand.grid**() creates a dataframe with all possible permutations of the vector elements and it allows for repeated sampling. The difference between **expand.grid**() and **combn**() is that **combn**() generates combinations without replacement irrespectively of the order.

[c] Why do the tails of the distribution (plot in lines 9 and 10) have so low frequencies? (0.2 points)

Comment: The plot in lines 9 and 10 shows the distribution of sample means. Because it is very low probability to draw all samples of smallest or largest values, the tails of sample means distribution have low frequencies.



[d] How does the expectation of the sample means $E(\bar{X})$ relate to the population expectation μ . (0.1 points)


Comment: The expectation of the sample means $E(\bar{X})$ and the population expectation μ are identical because the sample mean is an unbiased estimate.


[e] How does the standard error $\sigma_{\bar{X}}$ of the sample means \bar{X} relate to the standard deviation of the population σ . (0.2 points)

```
popSd(sampleMeans)
[1] 1.224745
```

Comment: The standard error of the sample means follows the central limit theorem $\sigma_{\bar{X}} = \frac{\sigma}{\sqrt{4}} = \frac{2.44949}{2} = 1.224745$

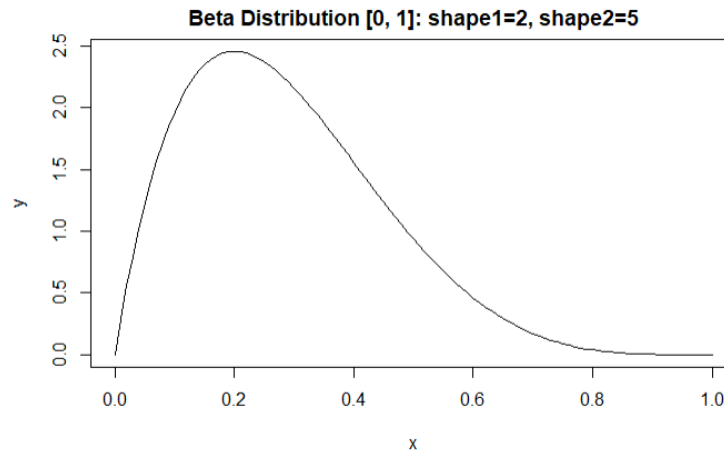
Task 2: Explore Samples (1.5 points)

For task 2 show all relevant  code.

[a] Plot the density function of a beta distribution (see  function `dbeta()`) with the shape parameters shape1=2 and shape2=5 in the default range of $x \in [0,1]$ (you can use `x <- seq(0,1,by=0.001)`). Label the axes of your plot properly. (0.2 points)

Note: You may want to use the `plot()` function with a properly set parameter for the **type**-option.

```
x <- seq(from=0, to=1, by=0.01)
y <- dbeta(x, 2, 5)
plot(x, y, type="l", main="Beta Distribution [0, 1]: shape1=2,
shape2=5")
```



[b] Check what the theoretical **expected value**, **variance** and **skewness** of this beta distribution are at **WIKIPEDIA** and report their values. Note that the shape parameters in **WIKIPEDIA** are called α and β . (0.2 points)

```
alpha<- 2; beta<- 5
(beta.mean<- alpha/(alpha+beta))
(beta.var<- (alpha*beta)/((alpha+beta)^2*(alpha+beta+1)))
(beta.skew<-2*(beta-
alpha)*sqrt(alpha+beta+1)/((alpha+beta+2)*sqrt(alpha*beta)))
```

Mean	Variance	Skewness
0.2857143	0.0255102	0.5962848

[c] Generate two sets of samples from the beta distribution. (0.2 points)

- The first set consists of 10,000 samples of size $n = 4$.
- The second set consists of 10,000 samples of size $n = 256$.

You can generate these sample with the statement `matrix(rbeta(n *10000, ...),
nrow=10000, ncol = n)`.

Calculate the 10,000 means of the samples for both datasets and save them into a vector. You may use the function `rowMeans ()`. **Do not show** the 10,000 calculated means based on either sample size but show your code!

```
sam1<- matrix(rbeta(4*10000, 2, 5), nrow=10000, ncol=4)
sam2<- matrix(rbeta(256*10000,2, 5), nrow=10000, ncol=256)
mean.sam1<- rowMeans(sam1)
mean.sam2<- rowMeans(sam2)
```

[d] Calculate the mean of the distribution of the sample means, the mean's standard error and mean's skewness for both sets (i.e., $n = 4$ and $n = 256$) and report the results in a properly formatted table. (0.2 points)

```
library(e1071)
```

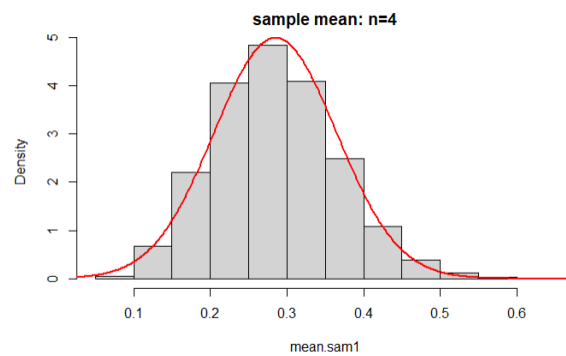
```
mean(mean.sam1); sd(mean.sam1); skewness(mean.sam1)
mean(mean.sam2); sd(mean.sam2); skewness(mean.sam2)
```

	Mean of sample means	The mean's standard error	Mean's skewness
Sam1	0.2850624	0.08000857	0.3106764
Sam2	0.2855376	0.009989509	0.02459543

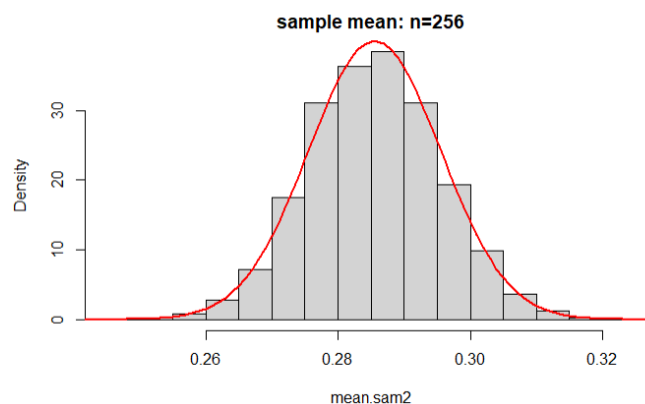
[e] Plot the histograms with the `hist()` function of distributions of the sample means for both sets. Use the option `freq=F` for the histogram. Make sure that you use a proper bin-width and value range for both histograms. (0.2 points)

Superimpose a normal curve within the chosen value range of the histogram. Use the correct theoretical parameters for the mean and standard deviation of the normal curve. The following statements will perform the task:

```
xRange <- seq(..., by=0.001)
lines(xRange, dnorm(xRange, mean=..., sd=...), col="red", lwd=2)
hist(mean.sam1, freq = F, breaks = 20, main="sample mean: n=4")
xRange <- seq(from=0, to=1,by=0.001)
lines(xRange, dnorm(xRange, mean=0.2850624, sd=0.08000857), col="red",
lwd=2)
```



```
hist(mean.sam2, freq = F, breaks = 20, main="sample mean: n=256")
xRange <- seq(from=0, to=1,by=0.001)
lines(xRange, dnorm(xRange, mean=0.2855376, sd=0.009989509),
col="red", lwd=2)
```



[f] Discuss both distributions of the sample means with regards to the central limit theorem and the underlying beta distributed population with the given shape parameters: (0.3 points)

- i. expected value of the distribution of the sample means;
The expected values of the small samples (0. 2005517) and large samples (0. 1999914) are almost equal to the true theoretical expectation $E(X)=0.2$.
- ii. standard error of the distribution of the sample means in dependence to the sample size n ; and
The standard error depends on the sample size: Compared with smaller sample size 4, the standard error of large sample means shrinks by the factor $\frac{\sqrt{256}}{\sqrt{4}} = 8$, $0.008605264 * 8 \approx 0.06813834$.

$$S_{\bar{x}} = \frac{\sigma}{\sqrt{n}}$$

The equation above indicates that the larger the sample size n , the smaller the uncertainty within the estimated mean. The standard error of the mean shrinks by the square-root of the sample size.

For $n = 4$: $S_{\bar{x}} = 0.06813834 \approx 0.1372/\sqrt{4}$


For $n = 256$: $S_{\bar{x}} = 0.008605264 \approx 0.1372/\sqrt{256}$

- iii. shape of the distribution of the sample means compared to that of the normal distribution and their deviations from the normal distribution, measured here by the skewness, in relation to the sample size n .

From the distributions above (Figure 1 and 2), we can clearly see, the distribution of larger sample means ($n=256$) is much closer to normal distribution than that of smaller sample size ($n=4$), which appears to be positively skewed. The large sample here follows central limit theorem, which states that for large sample size n , the sampling distribution of sample mean is normal distributed.


Part II: Spatial Sampling

Task 3: Randomly Sampling a Set of Regions (0.6 points)

For task 3 show all relevant  code.

[a] Open the file **Provinces.shp** from Lab03. Randomly sample a set of 50 provinces out of the 95 Italian provinces **without replacement**. (0.3 points)

Hints:

- Study the function **sample ()** in 's online help.
- The syntax **idx <- sample (1:95, ...)** and **Provinces.shp[idx,]** performs the selection of the provinces.

```
library(GISTools)
pro<-
readShapePoly("Italy/Provinces.shp", IDvar="ID", proj4string=CRS("+proj=
longlat"))
nei<- readShapePoly("Italy/Neighbors.shp",
proj4string=CRS("+proj=longlat"))
```

```

pro.df <- as.data.frame(pro)
pro.bbox <- bbox(pro)
n <- length(pro)
idx <- sample(1:n,50, replace = F)
pro.df[idx, "Sample"] <- "Sampled"
pro.df[-idx, "Sample"] <- "NotSampled"
pro.df$Sample <- as.factor(pro.df$Sample)

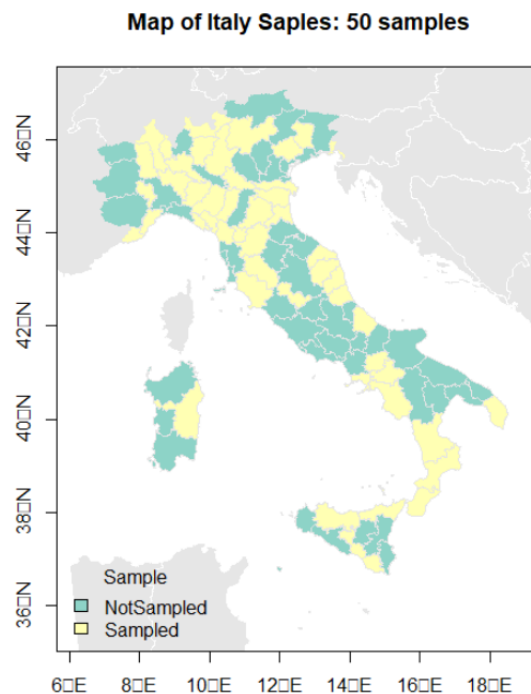
```

[b] Generate a qualitative map for the set of sampled provinces, which shows your sampled and non-sampled provinces using a qualitative map theme. (0.3 points)

```

library(TexMix)
plot(nei, axes = T, col=grey(0.9), border = "white",
xlim=pro.bbox[1,], ylim=pro.bbox[2,])
mapColorQual(pro.df$Sample, pro, map.title=paste("Map of Italy
Samples:", length(pro.df[idx, "Sample"]), "samples"), legend.title =
"Sample", add.to.map=T)

```



Task 4: Area Estimation by Sample Points and Traverses (1 point)

Below you find a map of a study region with a total area of 1 unit. This region has several lakes; one lake also has an island.

Note: the true total lake area happens to be 27 % of the study areas. In total 20 random points were placed into this study region. Furthermore, randomly 10 point pairs were connected by straight lines. Hint: This task can be done manually and using a ruler to evaluate the segment length of the traverses.

[a] Use the 20 sample points to estimate the total lake area. Explain the equation that you have used to estimate the area. (0.5 points)

From the map below, we know, the total study area is 1. There are total 20 points, 7 points in the lake, and 13 points on land. So the proportion of points in lake is $\frac{7}{20} = 0.35$. Therefore, the estimated lake area is 35% compared to the true lake area of 27%. A larger sample size would get closer to the true lake area.

[b] Use a ruler to estimate the total lake area based on the 10 random lines (labeled L:1 to L:10).

Generate a table similar to that on page 286 of Burt, Barber and Rigby. (0.5 points)

	L (length)	L_l (length in lake)	L_{nl} (length not in lake)	C_l (cumulative total length)	C_{L_l} (cumulative length in lake)	$C_{L_{nl}}$ (cumulative length not in lake)
L1	2.8	1.1	1.7	---	---	---
L2	1	0.4	0.6	3.8	1.5	2.3
L3	8.1	4	4.1	11.9	5.5	6.4
L4	5.7	2.8	2.9	17.6	8.3	9.3
L5	12	1.1	10.9	29.6	9.4	20.2
L6	2.8	0.9	1.9	32.4	10.3	22.1
L7	5.4	1.3	4.1	37.8	11.6	26.2
L8	5.8	2.9	2.9	43.6	14.5	29.1
L9	6.4	0	6.4	50	14.5	35.5
L10	4.1	2	2.1	54.1	16.5	37.6

$$\text{The area of lake} = \frac{C_{L_l}}{C_l} \times \text{study area} = \frac{16.5}{54.1} \times 1 = 0.305$$

Because transect sampling uses more information compare to the simple point sample the estimated lake area on average will be closer to the true lake area.

