

FASS512: Statistical estimation

Professor Patrick Rebuschat, p.rebuschat@lancaster.ac.uk

Please work through the following handout at your own pace.

As in the previous handouts, please type the commands in your computer. That is, don't just read the commands on the paper, please type every single one of them.

Before running the commands, think about what you expect to happen. If you are able to do this, that's a good sign that you are starting to understand the R language. 😊

This handout assumes that you have completed all previous handouts. If you haven't, please do this before working on the following handout. Handouts are available on [Moodle](#).

References for this handout

Many of the examples and data files from our class come from these excellent textbooks:

- Andrews, M. (2021). *Doing data science in R*. Sage.
- Brown, D. S. (2021). *Statistics and data visualization using R. The art and practice of data analysis*. Sage.
- Cumming, G. & Calin-Jaegeman, R. J. (2017). *Introduction to the New Statistics: Estimation, Open Science, and Beyond*. Routledge.
- Crawley, M. J. (2013). *The R book*. Wiley.
- Fogarty, B. J. (2019). *Quantitative social science data with R*. Sage.
- Winter, B. (2019). *Statistics for linguists. An introduction using R*. Routledge.

Are you ready? Then let's start on the next page! 📄

Task 1: *Interpreting z-scores*

In this task, we will use the ESCI “Distributions” tool to interpret z-scores. We will focus on the following three z-scores.

$$z_{36} = \frac{36 - 40}{8} = -0.50$$

$$z_{42} = \frac{42 - 40}{8} = 0.25$$

$$z_{50} = \frac{50 - 40}{8} = 1.25$$

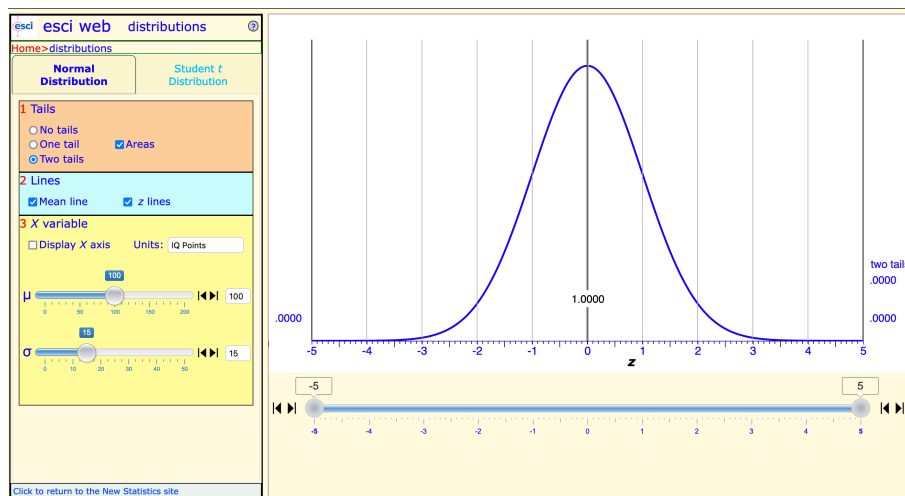
You can access the tool by clicking on the link below:

<https://www.esci.thenewstatistics.com/esci-distributions.html>

On the left side of the page, adjust the controls as follows.

In section 1, select Two tails and Areas. In section 2, select Mean line and z-lines.

Your screen should look like this.



To inspect the z-scores, use the slider underneath the normal distribution.

Adjust the slider to inspect each of the following three z-scores in turn: **-0.50**, **0.25**, and **1.25**.

Each time, ask yourself (and note down your observations):

- How close is each z-score to the mean? (Remember: In a standard normal distribution, the mean is always 0 and the SD is 1.)
- How typical or extreme is each of the three z-scores?

Once you have done this, please rank the three z-scores.

- c) Which score is most likely to occur in a normal distribution, which is least likely?

Discuss your observations with your neighbour(s).

Task 2: *Converting raw scores to z-scores in R*

Let's begin by installing the tidyverse package.

```
library(tidyverse)
```

For this task, we will use an existing data set in R called `MASS::nlschools`.

This data set contains data from eighth-grade pupils in the Netherlands (hence, NL schools). The sample features data from 2,287 pupils (aged about 11, grade 8) in 132 classes in 131 schools in the Netherlands.

There are 2,287 rows (one per participant). Columns include `Lang` (language test score), `IQ` (verbal IQ), `GS` (class size), and `SES` (social-economic status of pupil's family).

You can inspect the entire data set by typing the following command.

```
View(MASS::nlschools)
```

It's also good to compute and observe a few summary statistics.

```
summary(MASS::nlschools)
```

lang		IQ		class		GS	
Min.	: 9.00	Min.	: 4.00	15580	: 33	Min.	:10.00
1st Qu.	:35.00	1st Qu.	:10.50	5480	: 31	1st Qu.	:23.00
Median	:42.00	Median	:12.00	15980	: 31	Median	:27.00
Mean	:40.93	Mean	:11.83	16180	: 31	Mean	:26.51
3rd Qu.	:48.00	3rd Qu.	:13.00	18380	: 31	3rd Qu.	:31.00
Max.	:58.00	Max.	:18.00	5580	: 30	Max.	:39.00
				(Other)	:2100		

SES		COMB	
Min.	:10.00	0:	1658
1st Qu.	:20.00	1:	629
Median	:27.00		
Mean	:27.81		
3rd Qu.	:35.00		
Max.	:50.00		

We will focus exclusively on the `IQ` variable, which contains the verbal IQ score for each pupil. You already have the summary statistics for this variable above. It might be helpful to plot a histogram, too.

```
hist(MASS::nlschools$IQ)
```

Your task

In this task, you will convert five raw scores from the IQ test, i.e. the actual scores of five pupils, into z-scores. This process is also known as “standardizing”.

These are the five raw scores: **4.0, 6.5, 9.5, 12.5, and 13.5**.

The formula to convert each of these into a z-score is below.

$$z = \frac{X - \bar{X}}{s}$$

where X is the raw score, \bar{X} is the sample mean, and s is the standard deviation of the sample.

Use R to convert the numbers into z-scores and complete the table below.

When you are done, please compare your script and your z-scores with your neighbour. Do the z-scores match? Did they use a similar R code to compute the z-scores?

raw score	z-score
4.0	
6.5	
9.5	
12.5	
13.5	

Do NOT go to the next page.

Please wait till we compare notes in class.

As mentioned, there are lots of different ways for completing the same task in R. Here is one way to convert the raw scores into z-scores. You can try it out.

To calculate the z-scores, we need to know the mean and standard deviation of the IQ variable in our sample.

First, we create a new variable called `all_IQ_scores`. This contains the 2,287 raw scores from the IQ column in the `nlschools` data set.

```
all_IQ_scores <- MASS::nlschools$IQ
```

Then, we create another variable called `raw_scores`, which contains a vector with our five raw scores of interest.

```
raw_scores <- c(4.0, 6.5, 9.5, 12.5, 13.5)
```

Finally, we create a third variable called `z_scores`, which contains a computation that takes each raw score in turn, subtracts from each the sample mean, and divides the difference by the standard deviation.

```
z_scores <- (raw_scores - mean(all_IQ_scores)) / sd(all_IQ_scores)
```

If you now run the command below, you will see the five z-scores corresponding to the five raw scores 4.0, 6.5, 9.5, 12.5, 13.5.

```
z_scores
[1] -3.7866021 -2.5782245 -1.1281713  0.3218818  0.8052328
```

If we want to visualize this more easily, we can also create a tibble.

The command below creates a tibble with three columns, `raw_scores`, `z_scores`, and `percentile`.

The `pnorm()` function tells us the percentile for each z-score. This gives us an indication of how exceptional each score is. A high percentile means many scores are lower than the score we are inspecting, a low percentile means only few scores are lower.

```
library(dplyr)
```





```
our_data <-
  tibble(
    raw_scores = raw_scores,
    z_scores = ((raw_scores - mean(all_IQ_scores)) / sd(all_IQ_scores)),
    percentile = pnorm(z_scores)
  )
```

Let's look at our data.

```
View(our_data)
```

You should see a table like this in your script editor.

What do these z-scores and percentiles tell us about each raw score? Please discuss your observations with your neighbour.

	 raw_scores 	z_scores 	percentile 
1	4.0	-3.7866021	0.0000763607
2	6.5	-2.5782245	0.0049654734
3	9.5	-1.1281713	0.1296237833
4	12.5	0.3218818	0.6262288719
5	13.5	0.8052328	0.7896573297

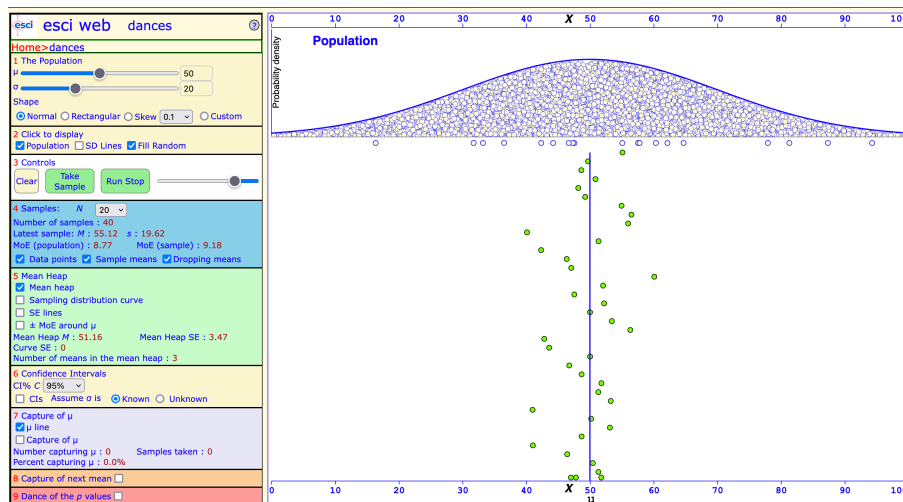
Task 3: *Sampling variability*

We return to the ESCI “Dances” tool. You can access the tool by clicking the following link:
<https://www.esci.thenewstatistics.com/esci-dances.html>

There's no need to change the default settings.

We know the population mean μ is 50 (see section 1 on the left). This is also represented by the blue μ line underneath the population distribution.

To learn more about sampling variability, let's start by drawing 40 random samples. We do this by clicking the Take sample button 40 times. Each time, a new green bubble will appear. These are the sample means of our 40 samples. Your bubbles will look different to the ones below!) Cumming (2012) calls this “the dance of the means”.



To learn more about sampling variability, consider the following:

- First, inspect your 40 sample means (green bubbles). Are they close to each other, or are some closer than others?
- Then, compare the sample means (green bubbles) in relation to the population mean μ (blue line). How close are the sample means to the true population mean? Are some bubbles closer and some further away? Why do you think this is?

Discuss your observations with your neighbour.

Task 4: *Exploring the “Dance of the Means”*

Let's return to the ESCI Dances tool: <https://www.esci.thenewstatistics.com/esci-dances.html>

On the left, please change the settings as follows.

In section 5, make sure the boxes for Mean heap and Sampling distribution curve are ticked.

In section 3, press the Clear button.

As you will see in section 1, the population mean is $\mu = 50$, and the population standard deviation $\sigma = 20$.

Task 4.1. *Simulation with $N = 10$*

Now go to section 4 and change sample size to 10 ($N = 10$).

Click Take sample once. You should see that one sample (with 10 observations) was drawn. The mean of that sample is the green bubble. The 10 clear bubbles immediately under the population curve are the individual observations that form your data set.

Now take another 200 samples with the same settings.

Inspect the “dance of the means”.

- a) Do the sample means (green bubbles) tend to be close to the blue line (the population mean μ)? Or do they tend to be all over the place?
- b) What about the “mean heap”, the pile of green bubbles at the bottom? The mean heap represents your empirical distribution, i.e. the actual scores. Is it a broad heap or a narrow heap?

Task 4.2. Simulation with $N = 50$

Now increase the sample size to 50 ($N = 50$).

First, press Clear in section 3, then increase the sample size N to 50 in section 4.

Again, take 200 samples with these settings.

Inspect the “dance of the means”.

- a) Do the sample means (green bubbles) tend to be close to the blue line (the population mean μ)? Or do they tend to be all over the place?
- b) What about the “mean heap”? Is a broad heap or a narrow heap?

Task 4.3. Simulation with $N = 100$

Let's increase the sample size again, this time to 100 ($N = 100$). Clear the space and increase sample size N to 100 in section 4.

Take 200 samples with these settings, and now inspect the “dance of the means”.

- a) Do the sample means (green bubbles) tend to be close to or further away from the blue line (the population mean μ)?
- b) Do you have a narrow or broad “mean heap”?

Before completing task 4.4., discuss the following with your neighbour:

How did the sample size changes ($N = 10, 50, 100$) affect the dance of the means and the mean heap?

Task 4.4. Comparing the effect of $\sigma = 20$ and $\sigma = 10$

In this task, let's keep the sample size stable (N) but change the population standard deviation σ .

We will compare two conditions. In both, the population mean μ is 50 and the sample size is set to $N = 20$. But we want to see the effect of $\sigma = 10$ and $\sigma = 20$.

First, make sure the population mean μ is still 50. Then change the sample size to 20 (section 4, $N = 20$).

Now run two simulations and write down your observations.

Each time draw 200 samples. Once with population standard deviation σ to 10, then with population standard deviation σ set to 20.

Discuss the following with your neighbour.

- a) How did changing the population standard deviation σ affect the population curve, i.e. the distribution at the top of the webpage?
- b) After you drew 200 samples, how did the change of σ affect the dance of the means and the mean heap, i.e. the sampling distribution of the means, see the bottom of the webpage.