Introduction to Statistics with Python

21-23 June 2021

Working document

**course webpage** : <https://edu.sib.swiss/course/view.php?id=523>

Username: statp21

Password: SIB-statp21

**Zoom link :**  <https://us02web.zoom.us/j/88554115308?pwd=ZlBBVjhJeHdPUDBJZjY3MWlnYUlNUT09>

**Schedule :**

* course from 9:00-12:00 to 13:00-17:00
* 15min breaks **around** 10:30 and 15:30

Day1 :

09-12 : python warmup; data exploration and visualization

13-17: statistical distributions and testing : core concepts behind statistical testing and applications (t-test, normality test)

Day2 :

09-12 : presentation of some common tests (fisher's exact test, chi square, ANOVA, ...)

13-17 : correlation and regression analysis : linear models

### Getting to know you

**Please write up your name as well as 3 keywords describing your research/interests**

Wandrille Duchemin : phylogenomics, python, bioinformatics

Sébastien Wieckowski: onco-immunology, vaccines

Sébastien Herbert: platform support, image analyst, microscopy

Sebastien Boyer : physicist, data analyst, evolutionary biologist

Loïc Jubin: Finance, data analyst

Mateusz Kozak: image analysis, cell biology

Sebastian Bechara: Molecular biology, genomics, RNA biology

Margarita Pertseva: immunology, repertoire analysis, data analysis

Gabriel Dupré: Virology, avian influenza viruses, evolution

Sandeep Grover: Geneticist, R, Epidemiologist

Bruno Fauvet: proteomics, transcriptomics, biochemistry

Romain Feron: comparative genomics, workflow development, genome alignment

Navin Gopaldass: BIochemist, yeast, membrane traffic

Charlotte Huyghe: molecular biology, adaptive radiation, transcriptomics

Wangjie Liu: single cell biology, genetic and genomic biology, data analysis

Solange Denervaud: cognitive neuroscience, education

Pascal Angst: Population genomics, biogeography, metapopulation genomics

Ahmed Sadek: Molecular biology, protein design,

Ellen Edwards: Immunology

Moritz Graeff: molecular biologist, plant genetics, gene expression networks

Diana Wüthrich: Pareto Multi Criteria Optimization in Radiation Therapy Treatment Planning

San Pun: immunology and hematology

Carla Bello: Evolutionary genomics, human genetics, non-coding genome evolution

Amol Panhale: Molecular biology, proteomics, biochemistry

**What is your relationship to python (put a + sign in the relevant line):**

never heard of it :

I a couple of things :+++++++

At ease with the basics : ++++++++++++

super comfortable :++

**How about stats (put a + sign in the relevant line):**

I know nothing :

I know what a mean is :

I heard of tests but theory is hazy : ++++++++

+++++++++

I am comfortable with most concepts : ++++

I do statistical modeling for fun :

## Monday morning - questions

**Q:** **what is the zip file for?**

**A:** It contains all the course material (notebooks, data, images) you will need for this 2-day course.

**Q: What is the format of exam?**

A: let me answer that at the break

The exam will be a couple of exercises centered around statistical questions. Code aspects are not the focus point.

You will have 2 week after the course to send us back your answers, by e-mail.

**Q: Sorry, I had some issues with my computer, where is the zip file?**

**A:** you can find it on the course website , at the bottom of the page https://edu.sib.swiss/course/view.php?id=523

Thanks!

**Q: Why using Python for statistics is better/worse than R?**

**A:** Usually it is comparable but for many applications Python is faster. But for basic stats I would say R is more well developed. Yet python can also open up to great stuff like many deep learning/ machine learning libraries where you could easily do calculation on gpu etc. But for today R is great :-) . Moreover I personally only can code in python so….I might be a bit biased.

A2: I use R and python daily and concur what the above answer. Ultimately, there is nothing that cannot be achieved by both languages, but the way libraries are built, I find that in R you can get to stats a bit faster than with python (in terms of language learning curve). Python shines both when opening to the advanced functionalities mentioned, but also I find when it comes to having more custom operations (going outside classical recipe), and also if you want to deploy your analysis script routinely.

**Q: I have a broader question that might be touched on later… How to deal with non-normal data when it comes to summary statistics, tests and so on?**

A : We will provide alternative tests to non normal data everytime we can. So you will definitely see that soon.

**Q: How can we extract the unique values of a column? Maybe there is some repetition of the same city for different observations?**

**I am wondering how to implement the sum(unique()) R -like code?**

**A:** I am not 100 % sure what you want but you also have the unique operator here just put unique at the end of the columns like df['district name'].unique(). Try that and tell me if this is solving it for you. You also have the sum operator.

R: Yes it solves my question. Thanks

R: Cool! Sorry I am knowledgeable in python but not so much in R so sometimes it is hard for me to be in your shoes.

**Q: I have an error when i am running the code:**

**sns.histplot(dfFractions['Swiss'] )**

**---------------------------------------------------------------------------**

**AttributeError Traceback (most recent call last)**

**<ipython-input-58-07d9be8cad05> in <module>**

**----> 1 sns.histplot(dfFractions['Swiss'] )**

**AttributeError: module 'seaborn' has no attribute 'histplot'**

**A:** pip install -U seaborn . This should solve it, because you probably don’t have the last version of seaborn. pip install -U seaborn in your terminal in the right conda env. Or !pip install -U seaborn in the notebook. You will have to reload the library though, so probably the notebook.

R: it is ok now for me. thanks again

sns.displot

**Q: I get an error NameError: name 'plotWithMeanMedianMode' is not defined** although it seems to be defined in in(35)

AMake extra sure you have successfully executed the cell defining it (maybe re-run it). Tell us if you still see the error

Yes, it seems that it does not run the cell that defines it. it just jumps to the next cell when I run

A: make sure that the cell is in code mode. When you are in this cell what is written in the top bar where you can run with the run button?

Yep, it says code

Weird can you share for a sec?

Wait a sec, I need to get into zoom with my coputer. thanks!

**UPDATED CODE for plotWithMeanMedianMode : previous version had an error in the legend, sorry !**

# Here we just define a small function for plotting a distribution with the mean median and mode

def plotWithMeanMedianMode( dat , ax):

mode=dat.mode()[0] #we only select the first mode

mean=dat.mean()

median=dat.median()

sns.histplot( dat , ax=ax , kde=True) # line for histogram and density line

ax.axvline(mean, color='r', linestyle='--' , label='mean')

ax.axvline(median, color='g', linestyle='-' , label = "median")

ax.axvline(mode, color='b', linestyle='-' , label = "mode ")

ax.legend()

**Q: why do only the group histograms show good automatic bins? How to set it automatic in individual histplot**

<https://seaborn.pydata.org/generated/seaborn.histplot.html>

It does it automatically but you can play with different options like giving the bins or giving the width. Try it.

Thanks!

**Q: What was the exact code of the third part for exercise 2? I had an error when running it.**

A: can you share your error. In the mean time here is the code

maskZH = dfFractions['canton'] == 'ZH'

sns.histplot(dfFractions.loc[maskZH,'Catholic'])

I found the error, I forgot “.loc”

Thanks!

**Q: What is a good way to layer different representations in Seaborn? Ie scatter + mean/SD or something like this.**

A: Else than just stacking them on the same plot?

No, just how to do that.

Plot the two different representations one after another in the same cell . Works for you?

No, produces three separate plots.

Can you share your code?

sns.catplot(data = dfFractions, x = 'canton', y = '60+ y.o.', kind = 'strip')

sns.catplot(data = dfFractions, x = 'canton', y = '60+ y.o.', k ind = 'point')

**Good :** start with something like fig,axes = plt.subplots(1,1 , figsize = (7,7) )

and then in the catplot tehre should be a ax option. Give me a few minutes

you need to define an axe from matplotlib

and add this axe to sns.

Something like that should work (this is from one of my project so change the variables and df of course)

fig, axe = plt.subplots(1, 1,figsize=(5,5))

sns.violinplot(x=df\_to\_model\_full['Experimental/approved'],

y=df\_to\_model\_full['Proportion\_F'],hue=df\_to\_model\_full['did it work?'],ax=axe)

sns.swarmplot(x=df\_to\_model\_full['Experimental/approved'],

y=df\_to\_model\_full['Proportion\_F'],hue=df\_to\_model\_full['did it work?'],ax=axe)

All right, I will play around with this. Thank you!

Let me know! I think it is a really cool idea for you to do that.

It appears that the problem is with catplot; if I use stripplot/pointplot etc directly, your way works and doing one after the other works too. With catplot I get errors (“catplot is a figure-level function and does not accept target axes.”).

**A:** Ah…. is it ok for now if you use my way rather than catplot. I will look more at it later on.

A rapid check on google seems to show that you can not use catplot for that, so the principle behind my bit of code is probably a good hack. You can change the alpha (related to transparency coefficient) of the violin plot for example to make your points in your swarm plot more visible. Also if you want to use point rather than swarm sns.pointplot()

**Q: I get error “TypeError: can only concatenate tuple (not "str") to tuple” if I run following code for exercise 3:**

kinds = ['box','violin','bar','boxen','strip','point']

represented\_variable = "60+ y.o."

category = 'canton name'

>for i,k in enumerate(kinds):

for k in enumerate(kinds):

g=sns.catplot( x = represented\_variable, y = category,

data = dfFractions, kind = k, orient = 'h', height=2, aspects=5)

give me so sec to debug that

ok you found it right it is because enumerate gives you 2 values the counts and the value.

for k in enumerate(kinds):

print(k) (try that)

You could change with for k in kinds

Also aspect not aspects

This is messy but working for me

kinds = ['box','violin','bar','boxen','strip','point']

represented\_variable = "60+ y.o."

category = 'canton name'

for i,k in enumerate(kinds):

g=sns.catplot( x = represented\_variable, y = category,

data = dfFractions, kind = k, orient = 'h', height=2, aspect=5)

Let me know

R: yes it works. Thanks! So even “i” does not appear within for loop, it is still required for enumerate?

A: Yes exactly. So since we dont need i . You can replace the for loop that way

for k in kinds:

g=sns.catplot( x = represented\_variable, y = category,

data = dfFractions, kind = k, orient = 'h', height=2, aspect=5)

R: great!

**Q: This is something I have a problem understanding. I understand that if the underlying population is, non-normal, we can derive a mean from the sampling distribution. But is that mean a good description of the underlying population if the population itself is not normal?**

**A:** It is a good description of the mean of this population not a good description of the population (I am not sure what that means by the way)! If the population mean exists then whatever your distribution is , the distribution of its sample mean will look like a gaussian (given that you calculated the sample mean with enough data). Basically what it means is that you are able to approximate/estimate the population mean by using the sample mean given some confidence interval.

This is called universality classes : the simple fact of adding together those realizations of the sampling makes this sum a normal random variable.

**If this is not clear please pin point what i should explain more**. Central limit theorem is as its name indicates central to what we are going to talk about in this course.

R: Thanks! I was more wondering about whether that mean is still an appropriate measure. For example in a very log-normal distribution the mean is very skewed, so even if we derive a real mean from sampling, is it really representative?

**A:** You are touching on a real good point here : what is representative of a distribution. One number that summarizes well a distribution doesn't exist. The cool thing with the gaussian is that you need only 2 numbers to completely define it : mean and variance. For other distribution which might be highly skewed or fat tailed, the mean the median or whatever single number is going to be a bit misleading . Not that it doesn’t mean anything… it means exactly what it means but you should be aware of the limitation of this representation.

**R: Thank you, a small follow up then: do any other statistics follow the CLT in a similar way? ie median, mode, geometric mean…**

**Q: I’m not sure I understand the question, the sample size could be 1, the important question is the number of independent sample isn’t it?**

**A:** You can have a sample size of one but then your confidence interval on your estimation of the population mean will be bigger than if you used a bigger sample size. Also all your samples are hypothetically independent in what we are doing right now.

Makes sense?

**More or less :) If we had 10 independent sample\_mean from each 10 sample, we wouldn’t consider that the distribution of the averages are representative right? But if we have 10.000 sample\_mean then it matches fairly the expected law for the averages? Or is it really that the std will never accurately represent the expected distribution?**

You can have one independant\_sample with 10 samples in it. You will be able to calculate a sample mean from that and you will be able to calculate a confidence interval around your sample mean that will describe your population mean. What you really need (look at the formula), is the number of data point used to calculate the estimated mean (sample mean).

I think I see, thanks I’ll reread later to make sure :)

ok!

Maybe I see now where your problem is : we usually don’t have access to the sigma and then to estimate this sigma we will need many independent sample\_mean. Is that what cause you problem?

Wandrille : if I understand correctly, you are discussion about the difference of having 100 samples of size 10, or 10 samples of size 100.

For the sake of ~~mental sanity~~ simplicity, we will presume that everything is independent.

**Q: I think I was mixing representativity of a normal law for a single sample distribution and for the TCL in that case.**

**Q: I was wondering similarly to the above, about the importance of sample size vs. number of independently drawn samples (let’s call them ’n’ and ’N’). In many fields of experimental biology, it’s easy to make many observations, of, say, cells (n > 30) in one experiment. In theory we could test hypotheses on that, like we did with coins (n = 10, N = 1). The caveat is that accounting for mistakes and systematic errors, each of the n observations inside one sample is not necessarily independent, so we repeat N times. Individual experiments can be costly in terms of time and money, so the N is small (typically 3). Is the best way to test and summarise such data is to treat the mean of each experiment as an independent observation, regardless of the n and regardless of the distribution of observations within one repeat?**

**A: … have to begin the course back again. I’ll answer after course if Sebastien has not.**

**A :** not my forte but if your N replicates are different from each other because of batch effect for example. Then you should include this batch effect in your modelling : you will still use those t-test kind of test (maybe an ANOVA), but in the calculation of the mean you will try your best to strip the effect of the batch : for example in taking into account the batch effect in you linear model (we will talk about linear modeling tomorrow and how they can be related to a mean calculation,tomorrow). If your N replicates are in fact identical then you should just pool all of those N together.

Am I close to answering your question?

**Q: Something basic which I am not sure about: How is a binomial distrubution different from a normal distribution?**

**A:** For big numbers of draws they are really equivalent : a binomial is going to count the number of success among N trials . Basically it will give you for example 1+0+1+1+0+0+1=4 for one realization of 4 success in 7 tries. Since you are summing a Bernouilli variable (the result of a coin flip i.e. 0 or 1) the Central limit theorem tells you that this sum (so the result of the outcome of a binomial draw) should tend to a gaussian distribution for enough trials. Does that make sense?

The random variable from a Binomial is the sum of Bernouilli variable : by CLT a Binomial random variable should tend to a gaussian variable if the number trials is big enough.

**Yes, thanks; so it is more depending on the kind of data what you have and Binomial distributions are more of a special case for a yes/no question?**

**A:** I am not 100% happy with the wording but I think you got it. Basically binomial count the number of success in a certain amount of draw : so it is an integer. Gaussian random variables can be whatever. So for now let’s just say that you stick to binomial when you want to count the number of success in some number of tries and in most of the case your random variables that you measured really look like a gaussian.

**Thanks**

By the way a yes no question is related to a Bernouilli experiment and generally not a Binomial experiment. Make sure you get the very subtle difference . A Bernouilli experiment is a Binomial experiment with only one try.

**Q: So both p-value and power depend on sample size, is that correct?**

**A:** Yes : it seems intuitive that the more data you have the easier it will be for you to be aware of both type error.

**Yes, indeed :) I guess it will also depend on how strong the differences are, but yes. the more data the better**

Exactly, but this is where there is a bit of conceptual difference : our p value depends on the effect size but we have no available action upon it. Whereas for power it really is a part of the formula : power and p values usually are not involved at the same steps of the experiments . Power is more in the design part so you can actually act on the difference you want to measure. Whereas p value are usually calculated after the experiment so you calculate it with the difference that you have. The same goes for the sample size. So both pvalue and power depends on both effect size and sample size but since they are not involved at the same time of the experiment you usually can’t play with them the same way.

**Oh, I see. Thank you for the clarification!!! Makes sense now.**

Let me try to write something clearer for tomorrow because it is a bit more subtile than that but overall you understood I think when they are used and what variables we can play on to change hem