

# week2\_assessment

June 29, 2020

## 0.1 Creating confidence intervals in python

In this assessment, you will look at data from a study on toddler sleep habits.

The confidence intervals you create and the questions you answer in this Jupyter notebook will be used to answer questions in the following graded assignment.

**IMPORTANT NOTE:** In this assessment, you will need to calculate a standard deviation. The Numpy and Pandas libraries both have functions/methods to perform this calculation. But they use different values for a parameter called the Delta Degrees of Freedom or DDOF.

The appropriate value of DDOF to use for this assignment is 0.

### Numpy

The `numpy.std()` function uses a default DDOF of 0. **Using `numpy.std()` is recommended for this assignment.**

<https://numpy.org/doc/stable/reference/generated/numpy.std.html>

### Pandas

The `pandas.dataframe.std()` and `pandas.dataframe.describe()` methods use a default DDOF of 1, which is NOT appropriate for this assignment. If you want to use these Pandas methods then you need to set the DDOF parameter argument to 0.

<https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.std.html>

```
In [5]: import numpy as np
import pandas as pd
from scipy.stats import t
pd.set_option('display.max_columns', 30) # set so can see all columns of the DataFrame
```

Your goal is to analyse data which is the result of a study that examined differences in a number of sleep variables between napping and non-napping toddlers. Some of these sleep variables included: Bedtime (lights-off time in decimalized time), Night Sleep Onset Time (in decimalized time), Wake Time (sleep end time in decimalized time), Night Sleep Duration (interval between sleep onset and sleep end in minutes), and Total 24-Hour Sleep Duration (in minutes). Note: **Decimalized time** is the representation of the time of day using units which are decimally related.

The 20 study participants were healthy, normally developing toddlers with no sleep or behavioral problems. These children were categorized as napping or non-napping based upon parental report of children's habitual sleep patterns. Researchers then verified napping status with data from actigraphy (a non-invasive method of monitoring human rest/activity cycles by wearing of a sensor on the wrist) and sleep diaries during the 5 days before the study assessments were made.

You are specifically interested in the results for the Bedtime, Night Sleep Duration, and Total 24- Hour Sleep Duration.

Reference: Akacem LD, Simpkin CT, Carskadon MA, Wright KP Jr, Jenni OG, Achermann P, et al. (2015) The Timing of the Circadian Clock and Sleep Differ between Napping and Non-Napping Toddlers. PLoS ONE 10(4): e0125181. <https://doi.org/10.1371/journal.pone.0125181>

```
In [6]: # Import the data (use this if running your Jupyter notebook within Coursera)
df = pd.read_csv("nap_no_nap.csv")
```

```
In [7]: # Import the data (uncomment the line below and use this if you downloaded the Jupyter :
df = pd.read_csv("https://raw.githubusercontent.com/UMstatspy/UMStatsPy/master/Course_2_
```

```
In [8]: # First, look at the DataFrame to get a sense of the data
df
```

```
Out[8]:
```

	id	sex	age (months)	dlmo	time	days napped	napping \
0	1	female	33.7		19.24	0	0
1	2	female	31.5		18.27	0	0
2	3	male	31.9		19.14	0	0
3	4	female	31.6		19.69	0	0
4	5	female	33.0		19.52	0	0
5	6	female	36.2		18.22	4	1
6	7	male	36.3		19.28	1	1
7	8	male	30.0		21.06	5	1
8	9	male	33.2		19.38	2	1
9	10	female	37.1		19.93	3	1
10	11	male	32.9		18.79	4	1
11	12	female	35.0		19.65	5	1
12	13	male	35.1		19.83	3	1
13	14	female	35.6		19.88	4	1
14	15	female	36.6		19.94	4	1
15	16	male	36.5		20.25	3	1
16	17	female	33.7		20.33	5	1
17	18	male	36.4		20.16	5	1
18	19	female	33.6		19.68	3	1
19	20	male	33.8		20.51	3	1

  

	nap	lights	outl	time	nap	sleep	onset	nap	midsleep	nap	sleep	offset \
0				NaN			NaN		NaN			NaN
1				NaN			NaN		NaN			NaN
2				NaN			NaN		NaN			NaN
3				NaN			NaN		NaN			NaN
4				NaN			NaN		NaN			NaN
5				14.00			14.22		15.00			15.78
6				14.75			15.03		15.92			16.80
7				13.09			13.43		14.44			15.46
8				14.41			14.42		15.71			17.01
9				13.12			13.42		14.31			15.19
10				13.99			14.03		14.85			15.68
11				13.18			13.45		14.33			15.21
12				13.94			14.48		15.26			16.03

13	12.68	13.08	13.92	14.76
14	12.71	12.88	13.80	14.72
15	13.74	14.68	15.66	16.64
16	13.15	13.87	14.49	15.11
17	12.47	12.56	13.30	14.05
18	14.71	14.85	15.46	16.07
19	12.68	13.54	14.30	15.07

	nap wake time	nap duration	nap time in bed	night bedtime \
0	NaN	NaN	NaN	20.45
1	NaN	NaN	NaN	19.23
2	NaN	NaN	NaN	19.60
3	NaN	NaN	NaN	19.46
4	NaN	NaN	NaN	19.21
5	16.28	93.75	137.00	19.95
6	16.08	106.00	80.00	20.60
7	15.82	121.60	163.80	22.01
8	16.60	155.50	131.25	20.24
9	15.30	106.67	130.67	20.78
10	16.10	98.75	126.60	19.45
11	15.35	105.80	130.40	20.18
12	15.78	93.33	110.20	20.22
13	15.00	100.75	139.33	20.26
14	14.88	110.75	130.00	20.28
15	16.45	117.33	162.75	20.46
16	15.40	74.20	135.00	20.43
17	14.25	89.80	107.00	20.02
18	16.20	73.00	89.40	19.50
19	15.23	91.67	152.67	20.18

	night sleep onset	sleep onset latency	night midsleep time \
0	20.68	0.23	1.92
1	19.48	0.25	1.09
2	20.05	0.45	1.29
3	19.50	0.05	1.89
4	19.65	0.45	1.30
5	20.25	0.29	1.26
6	20.96	0.36	2.12
7	22.53	0.51	2.92
8	20.37	0.13	1.60
9	21.63	0.84	2.20
10	19.88	0.44	1.34
11	20.84	0.66	1.93
12	20.89	0.67	1.99
13	20.80	0.54	1.96
14	20.92	0.64	1.49
15	21.25	0.79	2.19
16	21.03	0.60	2.44

17	20.45	0.43	1.23
18	19.64	0.14	1.42
19	21.38	1.19	2.51

	night wake time	night sleep duration	night time in bed \
0	7.17	629.40	643.00
1	6.69	672.40	700.40
2	6.53	628.80	682.60
3	8.28	766.60	784.00
4	6.95	678.00	718.00
5	6.28	602.20	653.80
6	7.27	618.40	655.40
7	7.31	526.80	582.40
8	6.82	626.80	660.33
9	6.52	549.50	626.00
10	6.80	655.20	694.80
11	7.03	611.20	660.40
12	7.09	611.80	662.20
13	7.11	618.80	671.20
14	6.33	548.00	595.00
15	7.13	593.25	662.00
16	7.86	649.80	708.60
17	6.01	573.60	614.60
18	7.20	693.40	715.00
19	7.63	615.33	692.00

	24 h sleep duration	bedtime phase difference \
0	629.40	-1.21
1	672.40	-0.96
2	628.80	-0.46
3	766.60	0.23
4	678.00	0.31
5	695.95	-1.73
6	724.40	-1.32
7	648.40	-0.95
8	782.30	-0.86
9	656.17	-0.76
10	753.95	-0.66
11	717.00	-0.53
12	705.13	-0.39
13	719.55	-0.38
14	658.75	-0.34
15	710.58	-0.21
16	724.00	-0.10
17	663.40	0.14
18	766.40	0.18
19	707.00	0.33

	sleep onset phase difference	midsleep phase difference \
0	-1.44	6.68
1	-1.21	6.82
2	-0.91	6.15
3	0.19	6.20
4	-0.13	5.78
5	-2.03	7.05
6	-1.68	6.84
7	-1.47	5.86
8	-0.99	6.22
9	-1.82	6.21
10	-1.09	6.55
11	-1.19	6.28
12	-1.06	6.16
13	-0.92	6.08
14	-0.90	5.64
15	-1.00	5.94
16	-0.70	6.12
17	-0.29	5.07
18	0.04	5.74
19	-0.87	6.00

	wake time phase difference
0	11.93
1	12.42
2	11.39
3	12.59
4	11.43
5	12.06
6	11.99
7	10.25
8	11.44
9	10.59
10	12.01
11	11.38
12	11.26
13	11.23
14	10.39
15	10.88
16	11.53
17	9.85
18	11.52
19	11.12

**Question:** What variable is used in the column 'napping' to indicate a toddler takes a nap?  
**Question:** What is the sample size  $n$ ? What is the sample size for toddlers who nap,  $n_1$ , and toddlers who don't nap,  $n_2$ ?

### 0.1.1 Average bedtime confidence interval for napping and non napping toddlers

Create two 95% confidence intervals for the average bedtime, one for toddler who nap and one for toddlers who don't.

Before any analysis, we will convert 'night bedtime' into decimalized time.

```
In [9]: # Convert 'night bedtime' into decimalized time
        df.loc[:, 'night bedtime'] = np.floor(df['night bedtime'])*60 + np.round(df['night bedtime'], 1)
```

Now, isolate the column 'night bedtime' for those who nap into a new variable, and those who didn't nap into another new variable.

```
In [10]: bedtime_nap = df[df["napping"] == 1]['night bedtime']
```

```
In [11]: bedtime_no_nap = df[df["napping"] == 0]['night bedtime']
```

Now find the sample mean bedtime for nap and no\_nap.

```
In [13]: nap_mean_bedtime = bedtime_nap.mean()
        nap_mean_bedtime
```

```
Out[13]: 1233.0666666666666
```

```
In [14]: no_nap_mean_bedtime = bedtime_no_nap.mean()
        no_nap_mean_bedtime
```

```
Out[14]: 1191.0
```

Now find the standard error for  $\bar{X}_{nap}$  and  $\bar{X}_{no\ nap}$ .

```
In [17]: bedtime_nap.describe()
```

```
Out[17]: count      15.000000
        mean      1233.066667
        std       34.445540
        min      1185.000000
        25%      1218.000000
        50%      1226.000000
        75%      1244.500000
        max      1321.000000
        Name: night bedtime, dtype: float64
```

```
In [18]: bedtime_no_nap.describe()
```

```
Out[18]: count      5.000000
        mean      1191.000000
        std       34.300146
        min      1161.000000
        25%      1163.000000
        50%      1186.000000
        75%      1200.000000
        max      1245.000000
        Name: night bedtime, dtype: float64
```

```
In [20]: nap_se_mean_bedtime = 34.445540/np.sqrt(15)
        nap_se_mean_bedtime
```

```
Out[20]: 8.893800184741428
```

```
In [21]: no_nap_se_mean_bedtime = 34.300146/np.sqrt(5)
        no_nap_se_mean_bedtime
```

```
Out[21]: 15.339491618833499
```

**Question:** Given our sample sizes of  $n_1$  and  $n_2$  for napping and non napping toddlers respectively, how many degrees of freedom ( $df$ ) are there for the associated  $t$  distributions?

To build a 95% confidence interval, what is the value of  $t^*$ ? You can find this value using the percent point function:

```
from scipy.stats import t
```

```
t.ppf(probability, df)
```

This will return the quantile value such that to the left of this value, the tail probability is equal to the input probability (for the specified degrees of freedom).

Example: to find the  $t^*$  for a 90% confidence interval, we want  $t^*$  such that 90% of the density of the  $t$  distribution lies between  $-t^*$  and  $t^*$ .

Or in other words if  $X \sim t(df)$ :

$$P(-t^* < X < t^*) = .90$$

Which, because the  $t$  distribution is symmetric, is equivalent to finding  $t^*$  such that:

$$P(X < t^*) = .95$$

So the  $t^*$  for a 90% confidence interval, and let's say  $df=10$ , will be:

$$t_{\text{star}} = t.\text{ppf}(.95, df=10)$$

```
In [28]: # Find the t_stars for the 95% confidence intervals
        nap_t_star = t.ppf(.975, df=14)
        nap_t_star
```

```
Out[28]: 2.1447866879169273
```

```
In [30]: no_nap_t_star = t.ppf(.975, df=4)
        no_nap_t_star
```

```
Out[30]: 2.7764451051977987
```

**Question:** What is  $t^*$  for nap and no nap?

Now to create our confidence intervals. For the average bedtime for nap and no nap, find the upper and lower bounds for the respective confidence intervals.

```
In [25]: upper_bound_nap=nap_mean_bedtime+(nap_t_star*nap_se_mean_bedtime)
        lower_bound_nap=nap_mean_bedtime-(nap_t_star*nap_se_mean_bedtime)
        print(lower_bound_nap, upper_bound_nap)
```

```
1213.99136242544 1252.1419709078932
```

**Question:** What are the 95% confidence intervals, rounded to the nearest ten, for the average bedtime (in decimalized time) for toddlers who nap and for toddlers who don't nap?

$$CI = \bar{X} \pm t^* \cdot s.e.(\bar{X})$$

**Challenge problem:** Write a function that inputs the column containing the data you want to build your confidence interval from and returns the confidence interval as a list or tuple (i.e. [upper, lower] or (upper, lower)).

```
In [27]: def get_confidence_interval(incoming_data):  
    mean_incoming_data=incoming_data.mean()  
    se_incoming_data = incoming_data.std() / np.sqrt(len(incoming_data))  
    t_star_incoming_data = t.ppf(.975, df=len(incoming_data)-1)  
    return(mean_incoming_data+(t_star_incoming_data*se_incoming_data), mean_incoming_data-(t_star_incoming_data*se_incoming_data))
```