

Σ+ SPSS TUTORIALS

BASICS DATA ANALYSIS T-TEST ANOVA CHI-SQUARE TEST

o Compute Means in SPSS?

You are here: [Home](#)

→ [Basics](#) → [SPSS](#)

[FAQ](#) → [How to](#)

[Compute Means in](#)

[SPSS?](#)

This tutorial shows how to compute means over both variables and cases in a simple but solid way. We encourage you follow along by downloading and opening [restaurant.sav](#), part of which is shown below.

	name	gender	v1	v2	v3	v4	v5
1	Ellie Phillips	0	4	6	6	3	3
2	Audrey Nelson	0	5	2	3	4	4
3	Leah Hernandez	0	.	.	.	4	5
4	Justin Rodriguez	1	3	5	5	5	5
5	Brianna Garcia	0	4	3	3	3	3

Quick Data Check

Before computing anything whatsoever, we always need to know what's in our data in the first place. Skipping this step often results in ending up with wrong results as we'll see in a minute. Let's first inspect some frequencies by running the [syntax](#) below.

***Show data values and value labels in output tables.**
 set tnumbers both.

***Quick data check.**
 frequencies v1 to v5.

Result

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	2 Bad	2	10,0	12,5	12,5
	3 Not good, not bad	8	40,0	50,0	62,5
	4 Good	1	5,0	6,3	68,8
	5 Very good	3	15,0	18,8	87,5
	! 6 No answer	2	10,0	12,5	100,0
	Total	16	80,0	100,0	
Missing	System	4	20,0		
Total		20	100,0		

Right, now there's two things we need to ensure before proceeding.
 Firstly, do all variables have **similar coding schemes**? For the food rating, higher numbers (4 or 5) reflect more positive attitudes ("Good" and "Very good") but does this hold for all variables? If we take a quick peek at our 5 tabels, we see this holds.

Second, do we have any **user missing values**? That is, do we want to include all data values in our computations? In this case, we don't. We

need to exclude 6 ("No answer") from all computations. We'll do so with the syntax below.

AdChoices

SPSS Data Analysis

Free SPSS Software

Analyzing Data

Setting Missing Values

***Set 6 as user missing value.**
missing values v1 to v5 (6).

***Check again.**
frequencies v1 to v5.

Result

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	2 Bad	2	10,0	14,3	14,3
	3 Not good, not bad	8	40,0	57,1	71,4
	4 Good	1	5,0	7,1	78,6
	5 Very good	3	15,0	21,4	100,0
	Total	14	70,0	100,0	
Missing	6 No answer	2	10,0		
	System	4	20,0		
	Total	6	30,0		
Total		20	100,0		

Computing Means over Variables

Right, the simplest way for computing means over variables is shown in the syntax below. Note that we can usually specify variable names separated by spaces but for some odd reason we need to use commas in this case.

```
*Compute mean over v1, v2, v3, v4 and v5.
```

```
compute happy1 = mean(v1, v2, v3, v4, v5).  
execute.
```

If our target variables are adjacent in our data, we don't need to spell out all variable names. Instead, we'll enter only the first and last variable names (which can be copy-pasted from **variable view** into our syntax window) separated by **TO**.

```
*Alternative: use TO keyword for specifying variables
```

```
compute happy2 = mean(v1 to v5).  
execute.
```

Computing Means - Dealing with Missing Values

If we take a good look at our data, we see that some respondents have a lot of missing values on v1 to v5. By default, the mean over v1 to v5 is

computed for any case who has at least one none missing value on those variables. If all five values are (system or user) missing, a mean can't be computed so it will be a **system missing value** as we see in our data.

	name	gender	v1	v2	v3	v4	v5
1	Ellie Phillips	0				3	3
2	Audrey Nel...	0	5	2	3	4	4
3	Leah Hern...	0	.	.	.	4	5

It's quite common to exclude cases with many missings from computations. In this case, the easiest option is using the dot operator. For example `mean.3(v1 to v5)` means "compute means over v1 to v5 but only for cases having at least 3 non missing values on those variables". Let's try it.

Computing Means - Exclude Cases with Many Missings

***Compute mean only for cases having at least 3 valid**

```
compute happy3 = mean.3(v1 to v5).
execute.
```

Result

*restaurant.sav [] - IBM SPSS Statistics Data Editor

compute happy3 = mean.3(v1 to v5).

name	gender	v1	v2	v3	v4	v5	happy3
Ellie Phillips	0	4	6	6	3	3	3,33
Audrey Nel...	0	5	2	3	4	4	3,60
Leah Hern...	0	.	.	.	4	5	.
Justin Rod...	1	3	5	5	5	5	4,60

A more general way that'll work for more complex computations as well is by using **IF** as shown below.

***Alternative way to exclude cases having fewer than 3**

```
if (nvalid (v1 to v5) >= 3) happy4 = mean(v1 to v5).  
execute.
```

SPSS - Compute Means over Cases

So far we computed horizontal means: *means over variables* for each case separately. Let's now compute vertical means: *means over cases* for each variable separately. We'll first create output tables with means and we'll then add such means to our data.

Means over all cases are easily obtained with **DESCRIPTIVES** as in

```
descriptives v1 v2.
```

Result

Descriptive Statistics					
	N	Minimum	Maximum	Mean	Std. Deviation
How did you like the restaurant?	15	2	5	3,53	,990
How did you like to food?	14	2	5	3,36	1,008
Valid N (listwise)	13				

Means for Groups Separately

So what if we want means for male and female respondents separately? One option is **SPLIT FILE** but this is way more work than necessary. A simple **MEANS** command will do as shown below.

***Show only value labels (no data values) in output table.**

```
set tnumbers labels.
```

***Report means for genders separately.**

```
means v1 v2 by gender/cells means.
```

Result

Report		
Mean		
gender	How did you like the restaurant?	How did you like to food?
female	3,80	2,60
male	3,40	3,78
Total	3,53	3,36

SPSS - Add Means to Dataset

Finally, you may sometimes want means over cases as new variables in your data. The way to go here is **AGGREGATE** as shown below.

***Add mean over v1 as new variable to data.**

```
aggregate outfile * mode addvariables
/mean_1 = mean(v1).
```

If you'd like means for groups of cases separately, add one or more **BREAK** variables as shown below. This example also shows how to add means for multiple variables in one go, again by using **TO**.

***Add means over v2 to v5 for genders separately as new variables.**

```
aggregate outfile * mode addvariables
/break gender
/mean_2 to mean_5 = mean(v2 to v5).
```

Result

	name	gender	v1	v2	mean_2	
1	Ellie Phillips	female	Good	No an...	2,60	
2	Audrey Nel...	female	Very g...	Bad	2,60	
3	Leah Hern...	female	.	.	2,60	
4	Justin Rod...	male	Not g...	Very g...	3,78	

Note that we already saw these means (over v2, for genders separately) in our output after running

```
means v2 by gender.
```

Right. That's about all we could think of regarding means in SPSS. If you've any questions or remarks, please feel free to throw in a comment below.

Let me know what you think!

Done!

**Required field. Your comment will show up after approval from a moderator.*

This tutorial has 9 comments

By Eugenia ML on June 12th, 2019

You are so clear and precise! It's a real help!
Thank you so much!



By Muhammad Abu-salih on March 7th, 2019

I need to try this tutorial to find its usefulness, but
computation of means through the usual commands of



Ne
1 -

SPSS are very easy.

AdChoices

SPSS Data Analysis

Free SPSS Software

Excel SPSS

By **William Peck** on November 5th, 2018



I got an error with the last two exercises on the Aggregate commands, ">Error # 10910. Command name: aggregate. The definition of a new variable on

[Expand comment](#) | [all comments](#)

By **Ruben Geert van den Berg** on October 29th, 2018



Use `MEAN ()` for means and `SD ()` for standard deviations. Don't use the `+` operator.

Also see [SPSS Missing Values Tutorial](#) for the reasons for this.

HTH!

1 2

Get In Touch!

SPSS Help (Netherlands)

Ruben Geert van den Berg

Sigma Plus Statistiek

[in](#) LinkedIn

www.sigma-plus-statistiek.nl

[f](#) Facebook

info@sigma-plus-statistiek.nl

SPSS Help (International)

SPSS tutorials

www.spss-tutorials.com

info@spss-tutorials.com