What are some tips for creating an Excel file that can be easily moved to a statistical program for analysis?

Authored by

stats writer

June 30, 2024

Excel is a popular spreadsheet software that is often used to organize and manipulate data. However, when it comes to statistical analysis, it is important to create an Excel file that can be easily transferred to a statistical program for efficient and accurate analysis. To do so, there are a few key tips to keep in mind. Firstly, it is important to ensure that the data is organized in a consistent and logical manner, with proper labeling and clear column and row headings. Secondly, using appropriate formatting such as consistent date and number formats can help prevent data errors during the transfer process. Additionally, it is recommended to avoid merging or splitting cells, as this can cause issues with data analysis. It is also helpful to use relevant and descriptive file names to easily identify the data when transferring to a statistical program. Finally, regularly saving and backing up the Excel file can prevent loss of data and ensure the file is up-to-date for analysis. By following these tips, the process of transferring an Excel file to a statistical program for analysis can be smooth and efficient.

Tips for creating an Excel file that can be easily moved to a statistical program for analysis

Excel is not a statistical package; however, Excel is often the software of choice for inputting data. So, even though we do not advocate its use for statistical analysis, here are some tips on setting up a file that can be easily imported into any statistical program for further analysis. We offer these tips in the hopes that they will ease the process of moving your data out of Excel and into a statistical package appropriate for your data analysis. We will demonstrate these tips by attempting to import

a poorly
designed Excel file into SPSS (version 21).

Here is a link to the data file we will be using
Excel_bad.

Below is snapshot of the data in Excel. It contains 200 unique observations and
13 variables.

	A	B	C	D	E	F	G	H	I	L	M	N	O
1							test scores						
2	respondent sex	id	race	$ses	schtyp	prgtype	READ	#write	1.math	science/socst	Notes	Date	Time
3	0	70	4	1	1	general	57	52	41	47/57		22-Dec-90	15 SECONDS
4	1	121	4	2	1	vocati	68	59	53	63/61		22-Dec-90	1.5 MINUTES
5	0	86	4	3	1	general and voc	44	33	54	58/31	New	1990 DEC 22	1.75 SECONDS
6	0	141	4	3	1	vocati	63	44	47	53/56		902212	90 SECONDS
7	0	172	4	2	1	academic	47	52	57	53/61		22.12.1990	16 SECONDS
8	0	113	4	2	1	academic	44	52	51	63/61		22/12/90	1.5 MINUTES
9	0	50	3	2	1	general	50	59	42	53/61		22/12/1990	1.75 SECONDS
10	0	11	1	2	1	Academic	34	46	45	39/36		12/22/1990	91 SECONDS

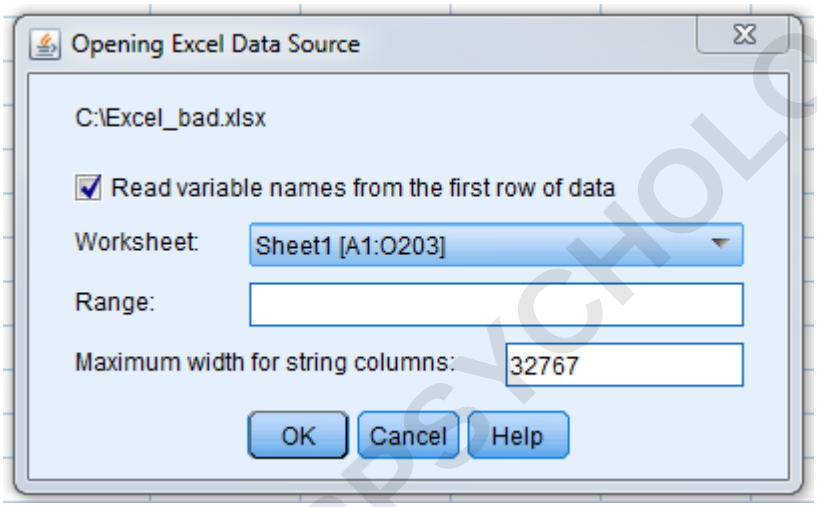
Our first step will be to try to open our Excel data file in SPSS.

To Open a file in SPSS : Click on the File tab -> Choose Open -> Choose Data

An dialogue box titled "Open Data" will appear. Navigate to the directory where you have saved the Excel file. Next, in the text box next to
"Files of

type" choose Excel. Then, in the text box next to "File
name" enter the name of our data file "Excel_Bad". Finally, click
Open.

A second dialogue box (shown below) will appear called "Opening Excel Data Source."



Make sure there is a check in the box next to "Read variables names from
first row of data." That way SPSS will know what our variables names
are automatically, by reading them from the top of our Excel data worksheet.
You will also see a box where you can choose the

"Worksheet"
you want SPSS to open. Our Excel file has two worksheets, one called "Excel_bad"
and one called "Test". Always make sure that SPSS is reading the correct
worksheet. Next to the worksheet name you will see . This indicates the
range of data SPSS is detecting in your Excel file. SPSS will be reading in data
that ranges from columns A through O and rows 1 through 203.

It is important to take a look at these data ranges and make sure they are what
you expect. For example, if you know your data only has 300 rows, but appears
that SPSS is detecting 500 rows, then there maybe additional data in other rows
of your Excel file that you were not expecting. You should correct this before
attempting to bring the data into SPSS, otherwise SPSS will read in a data file
with 200 empty rows. You will notice that 203 observations were detected by

SPSS, we will discuss why in a subsequent section.

After you have chosen the correct Excel file, click on OK.

If we take a look at our data file, we will immediately see that something went wrong!

	V1	V2	V3	V4	V5	V6	testscores	V8	V9	V10	V11	V12	V13	V14	V15
1	respondent sex	id	race	$ses	schtyp	prgtype	READ	#write	1.math	science	socst	science/socst	Notes	Date	Time
2	0	70	4	1	1	general	57	52	41	47	57	4		33229	15 SECONDS
3	1	121	4	2	1	vocati	68	59	53	63	61	6		33229	1.5 MINUTES
4	0	86	4	3	1	general and voc	44	33	54	58	31	5	New	1990 DEC 22	1.75 SECONDS
5	0	141	4	3	1	vocati	63	44	47	53	56	5		902212	90 SECONDS
6	0	172	4	2	1	academic	47	52	57	53	61	5		22.12.1990	16 SECONDS
7	0	113	4	2	1	academic	44	52	51	63	61	6		22/12/90	1.5 MINUTES
8	0	50	3	2	1	general	50	59	42	53	61	5		22/12/1990	1.75 SECONDS
9	0	11	1	2	1	Academic	34	46	45	39	36	3		33229	91 SECONDS
10	0	84	4	2	1	general	63	57	54		51	/51		XII 22 1990	17 SECONDS
11	0	48	3	2	1	academic	57	55	52	50	51	5		33229	1.5 MINUTES

SPSS appears to have included our row of variable names as an observation in our data file instead of reading them in as variable names like we specified! Let's investigate how that happened.

Well if you look back our Excel file, you will see that the first row actually was not our variables names. The first row included a merged column called "Test Scores." We will need to delete that row, re-save our

Excel file, and
then open our updated file in SPSS.

Now, as we can see below, our variables names have been read correctly by SPSS, but some of the variable names appear to have been changed by SPSS.

	respondentsex	id	race	@$ses	schtyp	prgtype	READ	write	@1math	science	socst	sciencesocst	Notes	Date	Time
1	0	70	4	1	1	general	57	52.000	41	47	57	4.00		33229	15 SECONDS
2	1	121	4	2	1	vocati	68	59.000	53	63	61	6.00		33229	1.5 MINUTES
3	0	86	4	3	1	general and ...	44	33.000	54	58	31	5.00	New	1990 DEC 22	1.75 SECONDS
4	0	141	4	3	1	vocati	63	44.000	47	53	56	5.00		902212	90 SECONDS
5	0	172	4	2	1	academic	47	52.000	57	53	61	5.00		22.12.1990	16 SECONDS
6	0	113	4	2	1	academic	44	52.000	51	63	61	6.00		22/12/90	1.5 MINUTES
7	0	50	3	2	1	general	50	59.000	42	53	61	5.00		22/12/1990	1.75 SECONDS
8	0	11	1	2	1	Academic	34	46.000	45	39	36	3.00		33229	91 SECONDS
9	0	84	4	2	1	general	63	57.000	54	.	51	.00		XII 22 1990	17 SECONDS
10	0	48	3	2	1	academic	57	55.000	52	50	51	5.00		33229	1.5 MINUTES

Variable Names

All of the variables that started with a "$" , "#", or a number, have either had an "@" added or the character removed upon converting the datafile from an Excel file to SPSS. The respondent sex variables had the space removed. Additionally, the science/socst variable had the "/" removed and as a consequence of SPSS having to reformat the variable for conversion we only have the first digit from our original science/socst

variable
preserved.

What happened? First, variable names in SPSS (as in most
statistical programs) cannot contain spaces, start with number, or includes
slashes. Second, a "#" character in the first position of a variable name
defines a special variables type of variable in SPSS called a
scratch variable. You should not specify a "#" as the first character
of a user-defined variable. A "$" sign in the first position indicates
that the variable is a system variable. The "$" sign is not allowed as the
initial character of a user-defined variable. Additionally, periods and
underscores should not be used at the end of variable names in SPSS.
However, the period, the underscore, as well the characters "$" and "#" can be
used within variable names (e.g. science#socst or

respondents_sex).

More information on specifying variable names can be found
here on the SPSS website.

Let's go ahead and fix those variable names in our Excel file and then
re-open the dataset in SPSS.

	respondent_sex	id	race	ses	schtyp	prgtype	READ	write	math	science	socst	sciencesocst	Notes	Date	Time
1	0	70	4	1	1	general	57	52.000	41	47	57	4.00		33229	15 SECONDS
2	1	121	4	2	1	vocati	68	59.000	53	63	61	6.00		33229	1.5 MINUTES
3	0	86	4	3	1	general and voc	44	33.000	54	58	31	5.00	New	1990 DEC 22	1.75 SECONDS
4	0	141	4	3	1	vocati	63	44.000	47	53	56	5.00		902212	90 SECONDS
5	0	172	4	2	1	academic	47	52.000	57	53	61	5.00		22.12.1990	16 SECONDS
6	0	113	4	2	1	academic	44	52.000	51	63	61	6.00		22/12/90	1.5 MINUTES
7	0	50	3	2	1	general	50	59.000	42	53	61	5.00		22/12/1990	1.75 SECONDS
8	0	11	1	2	1	Academic	34	46.000	45	39	36	3.00		33229	91 SECONDS
9	0	84	4	2	1	general	63	57.000	54	.	51	.00		XII 22 1990	17 SECONDS
10	0	48	3	2	1	academic	57	55.000	52	50	51	5.00		33229	1.5 MINUTES

Working with String Variables

This is starting to look better, but we still have a few more data management
issues to address.

First, you will notice that in addition to reformatting the original
science/socst, information was lost. This occurs because a mixture of
string and numeric values such as "63/61" can confuse

SPSS.
SPSS uses the first value that it sees in a column to decide if that column
should be stored using a string, date, or numeric format. If any further values in
that column do not match the format of your first value, SPSS may convert that
value to missing (as is the case with observation 9) or it may
truncate the information to match the detected format. Additionally, special symbols like the dash "-" or "/"
can often cause problems because they imply some sort of mathematical operation.
For this variable, it is best to use the individual science and
socst variables. You can
perform mathematical functions such as ratios, division etc. later in SPSS using the
compute command. For more information on the compute command take a look at our
learning module on
computing variables.

Next take a look at our prgtype variable. It appears that

some of our categories are spelled differently for different observations. For example, observations 5 and 6 are labeled as "academic" (with a lower case a) while observation 8 is labeled "Academic" (with a uppercase A). This can become a problem when trying to use this variable in an analysis or recode it. Let's demonstrate this using the Count Values within Cases command.

```
COUNT academic=prgtype('academic').
EXECUTE.
```

This syntax will create a new variable called academic, which will be a 1 if prgtype = academic and a 0 if not.

Below is some output showing our original variable prgtype and our newly created variable academic. Let's take a look at

the same three observations. As you can see below, observations 5
and 6
have the value of 1 for the variable academic
but
observation 8 has the value of 0 for the variable academic.

academic prgtype

.00
general
.00
vocati
.00
general and voc
.00
vocati
1.00 academic
1.00 academic
.00
general
.00
Academic

.00
general
1.00 academic

Number of cases listed:
10

What happened? The problem with this, is that SPSS will
interpret the values of "academic" and "Academic" as different categories.
Therefore, it is important to be consistent in the spelling when
using string or character information. This will make subsequent analysis and data management much
easier and more efficient. Additionally, this is also important if the
variable(s) of interest are stored on different Excel worksheets. For example,
if information on prgtype for men and women were stored on
separate worksheets, it would be important that the variable and group names be
spelled the same. This way the information can be

appended or merged easily.

Next is our Note column. Often time researchers may make
notes to themselves in a data file. For example, you might want to label a
respondent as "New" if they are new patient. This type of open-ended text
information typically cannot be analyzed in most statistical programs including
SPSS. This information is best stored elsewhere in a separate data file, not
used for analysis.

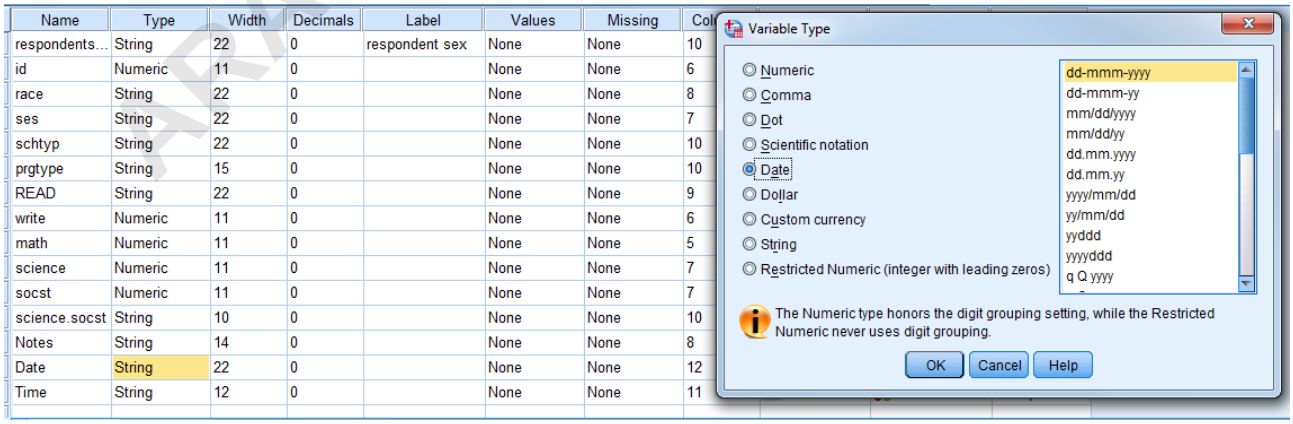
Date and Time

Now let's take a look at our Date and Time
columns. SPSS can recognize several different formats for data and time
information. In SPSS, as in many statistical programs, date/time variables are
stored as the variable type numeric with formats applied that display them in a
form we are used to (e.g. dd-mm-yyyy hh:mm:ss). A comprehensive

list of the data and time formats recognized by SPSS can be found on their

website. Whichever format you choose, it is best to be consistent for all
data that you input into your data file.

As you can see in the picture below, under the Variable View
tab, SPSS will allow you to declare a variable date and then apply a format
based on how the dates were entered into the data file. Unfortunately in our
mock dataset, the date information has been stored in several different formats
making it impossible to apply a single format in SPSS.



Similarly, Time is also stored in different formats

(seconds
or minutes). Just like dates, you can tell SPSS how the information in Time
is to be stored. If you look
above, you will see that SPSS has declared the
variable Time as string (or character). This is occurs because
the words "seconds" and "minutes" are stored in the same field
as the actual numeric time. SPSS does not
recognize these to mean numeric time, it reads them as text and therefore stores it accordingly.

If you would like more information about manipulating date/time information in
SPSS, take a look at our learning module Using dates in SPSS.

Summations and Averages:

When we opened our "Excel_bad" data you
may have noticed, that SPSS imported 203 observations. But our
data file only has 200 unique observations. Two of those extra observations was

the miss-read rows of variable names that we corrected, but that still leaves one unexpected observation. Let's investigate what happened.

If you go to Data View in SPSS and scroll down to the last record (observation 201), you will see two numeric values in the columns READ and write.

Where do these values come from? In our original Excel data file, we summed the values in the READ column and averaged the values in write column. Even thought they are not true observations, these values are retained when opening the data in SPSS, and recognized as an additional observation. This type of information should be removed from your original Excel data file before attempting to open it in SPSS.

Mathematical operations such as these and a number of others, can be performed in SPSS using the descriptives or compute commands. If you would like more information on how do this, please take a look at our learning module on Descriptive statistics in SPSS or the Class Notes from our Introduction to SPSS seminar.

After all those changes are new Excel file is now formatted appropriately for analysis:

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	respondent_sex	id	race	ses	schtyp	prgtype	READ	write	math	science	socst	Date_ddmmyyyy	Time_Seconds
2	0	70	4	1	1	general	57	52	41	47	57	12/22/1990	15
3	1	121	4	2	1	vocati	68	59	53	63	61	12/22/1990	90
4	0	86	4	3	1	general and voc	44	33	54	58	31	12/22/1990	1.75
5	0	141	4	3	1	vocati	63	44	47	53	56	12/22/1990	90
6	0	172	4	2	1	academic	47	52	57	53	61	12/12/1990	16
7	0	113	4	2	1	academic	44	52	51	63	61	12/12/1990	90
8	0	50	3	2	1	general	50	59	42	53	61	12/12/1990	1.75
9	0	11	1	2	1	academic	34	46	45	39	36	12/22/1990	91
10	0	84	4	2	1	general	63	57	54		51	12/22/1990	17

Summary - Things to look out for: