Mini-Dissertation Write-up Guide

Part 03 - Open Data

Dr. Gordon Wright

January 30, 2024

Open Data

Open Data requirements

Open Data is the concept of making research data accessible to everyone, promoting transparency, reproducibility, and collaboration in research - all critical features of an Open Science. In the field of psychology, sharing data is crucial for advancing scientific knowledge. This guide will provide step-by-step instructions for creating an Open Data submission for a psychology research project such as the Mini-Dissertation (and your Final Year Dissertations for that matter!).

> ❗ Important
>
> It is a compulsory that you submit your data. If you do not submit something, then your submission is not complete and the mark awarded will be impacted.
> By taking the time to produce a clear Open Data submission, you will enhance the overall quality of your submission, and this will be reflected in your mark.

1. Prepare your data: Before submitting your data, ensure that it is well-organized, accurate, and complete. You might be fixated on doing the analysis, but in many ways, the way you treat and pre-process your data (prepare it for the analysis) is even more critical to reliable results.

2. This includes:

   a. Cleaning and validating the data: Remove any errors, inconsistencies, or missing values. This may be down to participant error or a result of anything you might have done in this, your first data collection exercise (mistakes are to be EXPECTED! Don't hide them!). Make sure the data is properly formatted (an example here might be consistent decimal places, or labelling in your variable headers.

b. Anonymizing the data: To protect participants' privacy, remove any personally identifiable information (PII), such as names, addresses, or user generated codes. Replace these with unique identifiers, if necessary. This can be as simple as a number from 1 - n (where n is your total sample size).

3. Create a data dictionary or codebook: A data dictionary is a simple document that describes the variables in your dataset, their definitions, units of measurement, and any coding schemes used, such as values for any categorised data e.g. gender (1 = female, 2 = male, 3 = prefer not to say).

   This will help others understand and use your data more effectively. Consider including the following information for each variable:

   a. Variable name as it exists in your dataset
   b. Variable description - what is the variable? Score on what measure of your Open Materials?
   c. Data type (e.g., categorical, continuous, binary)
   d. Units of measurement (if applicable, milliseconds, hours per day)
   e. Consider count information, or summary information (mean, range etc)
   f. Coding scheme (if applicable, 1 = strongly disagree, 2 = disagree etc.)

4. Information on any pre-processing you performed:

   What were your decision rules on missing data or participant exclusion? Your data set can be the original data set which includes data you later do not analyse, or the data set that had undergone pre-processing and had had all the missing values removed… in either case, I want to know the process involved, so that I will be able to do it if I try to replicate your analysis!

5. Choose a suitable format (or formats) for submission: In real research, you could host this on OSF.io or a similar data repository, but you are going to submit it as a supplementary file. Choose a format (or formats) that are usable. Ideally, I would like to be able to import your data directly, so it should be in a .csv, .xlsx, or .sav file, but Jamovi, R or equivalent is fine too.

   Any descriptive content (such as 3 or 4) could be in a pdf and submitted alongside the Mini-Dissertation (you can submit up to 5 files)

By following these guidelines, you will contribute to a more transparent and collaborative research environment in the field of psychology, ultimately promoting the advancement of scientific knowledge.

Week 14 - Getting set up to run your analysis

Goals and outcomes of this week:

- A quick review of the SPSS Data editor window and some useful menu options
- Inputting data into SPSS for your specific ANOVA design
  - 2x2 Independent (between-groups) ANOVA
  - 2x2 Repeated measures ANOVA
  - 2x2 Mixed ANOVA
- Some useful calculations in SPSS
  - Performing a Median Split in SPSS
  - Calculating a mean score variable

Analysing your data starts here... with good set up!

I know that you have used SPSS plenty already, but it is normally the case that you've only worked on downloaded pre-existing data sets, or followed step-by-step guides as a class. This time, these are YOUR data, of which you should be proud! Nobody has ever seen these data before.

Examples of the 3 flavours of dataset can be found on the VLE.

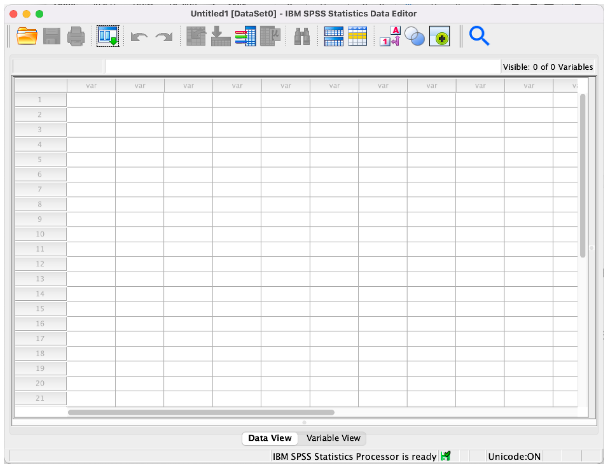
For your Mini-Dissertations you are going through all the stages of a research project, from idea origination, through to the final presentation of the report. One of the parts that often (unnecessarily) causes anxiety is the analysis. Don't let it! You'll get lots of help in the labs after Reading Week!

It is important that you go about the process of data preparation and analysis methodically and with care. You've worked hard to generate these data, give them the attention they deserve. And at the end of the process... you get to see what happened in your experiment.

This worksheet will offer you tools to start off well. I don't doubt that you will have an occasional mishap, but that is how we learn and improve. And I still google things sometimes!

Remember, the mini-dissertation is carefully constrained to a 2x2 design, but by using this as a case-study, we can equip you with all the skills necessary for something more exotic next year.

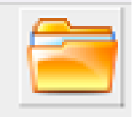
When you open SPSS, you'll see the familiar Data Editor screen. Let me point a few things out.

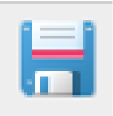

Untitled1 [DataSet0] - IBM SPSS Statistics Data Editor

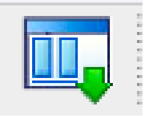
Visible: 0 of 0 Variables

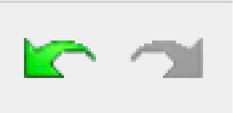
IBM SPSS Statistics Processor is ready

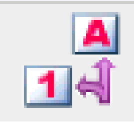
Unicode:ON


Open data document. This is where you can navigate to open your data set. Obvious, I know.


This button saves your data set. Use this regularly. And save your data set in the cloud. You do not want to have to redo data preparation. And use version control. If you spend time on a data set and reach the end of a session, save it with the date and start the next session with a copy. I dearly hope you listen to this, and don't learn the hard way.


Recently used dialogues. Allows you to pick from a dropdown menu of operations you have performed previously. Most people don't get to know this button exists and spend hours clicking through the menus. Using this button is the easiest way to do what you did last time, but make a slight modification (e.g. you forgot to select something).


Most things that you do in the data editor window can be undone and redone. Anything that can't be undone will give you a warning window. If you are going to try stuff out, use a copy of your dataset and keep a safe copy of the last one you were working on – see Save button above.



Flip between Value labels. You'll see how this can be helpful later on. Just pointing it out for now.

Sometimes you want to look at your data (the numbers) and sometimes you want to look at what the data are (labels and types of variables). `Data View` lets you look at the numbers. Variable View lets you look at the labels and types of variables in your data. Let’s look at Variable View now.

Now that you are reminded of those, let’s get things ready for when your data are ready.

Data input

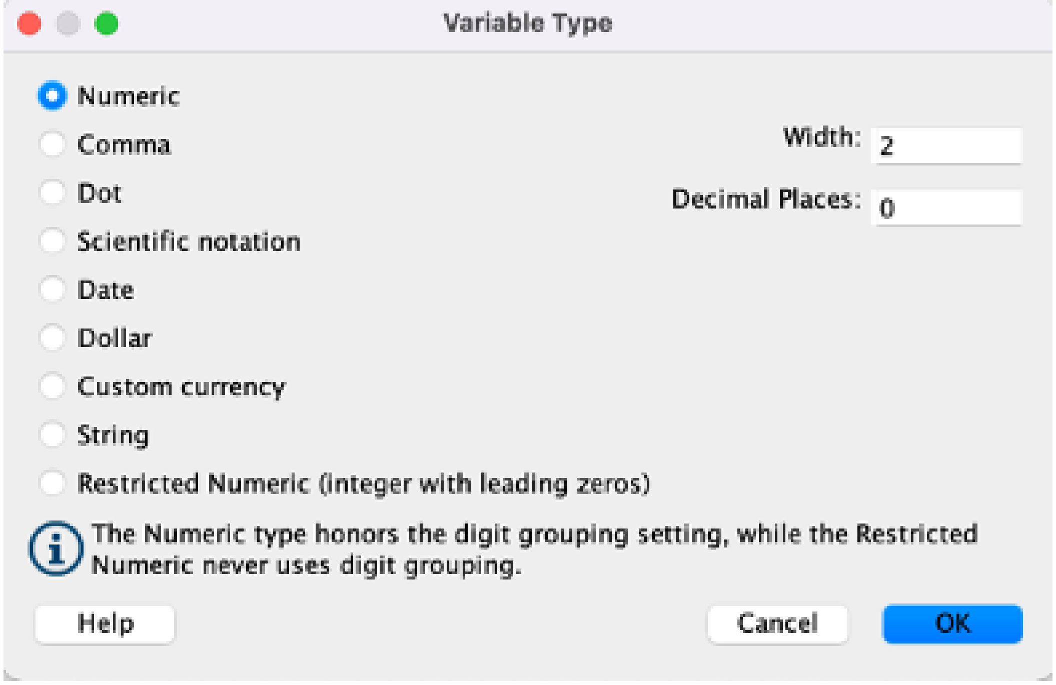
It may seem supremely dull, but properly formatting and labelling your data is important. Remember that you will be uploading a dataset as part of your submission under Open Data. It is important that an outsider can understand your dataset and use it to reproduce your analysis, should they wish.

We use the `Variable View` in SPSS to add and define the variables used in our study. Each row represents one of your variables. Here, we can give them sensible names and specify the type of data we are entering for each one.

So start by thinking about `Name`**.** A variable name should be easy to understand and informative. You can’t use spaces or symbols, so you should think about using underscores or hyphens to-make-things_easier_to_understand.

`Type` specifies the type of data that is contained in that variable.

When inputting your data, you will likely mostly use `Numeric`**,** which simply means it mostly numbers. It could be scores on a test, or categories which might be relevant for gender or group membership `(1=Female, 2= Male, 3= Prefer not to say`, etc)



● Numeric
○ Comma
○ Dot
○ Scientific notation
○ Date
○ Dollar
○ Custom currency
○ String
○ Restricted Numeric (integer with leading zeros)

Width: 2

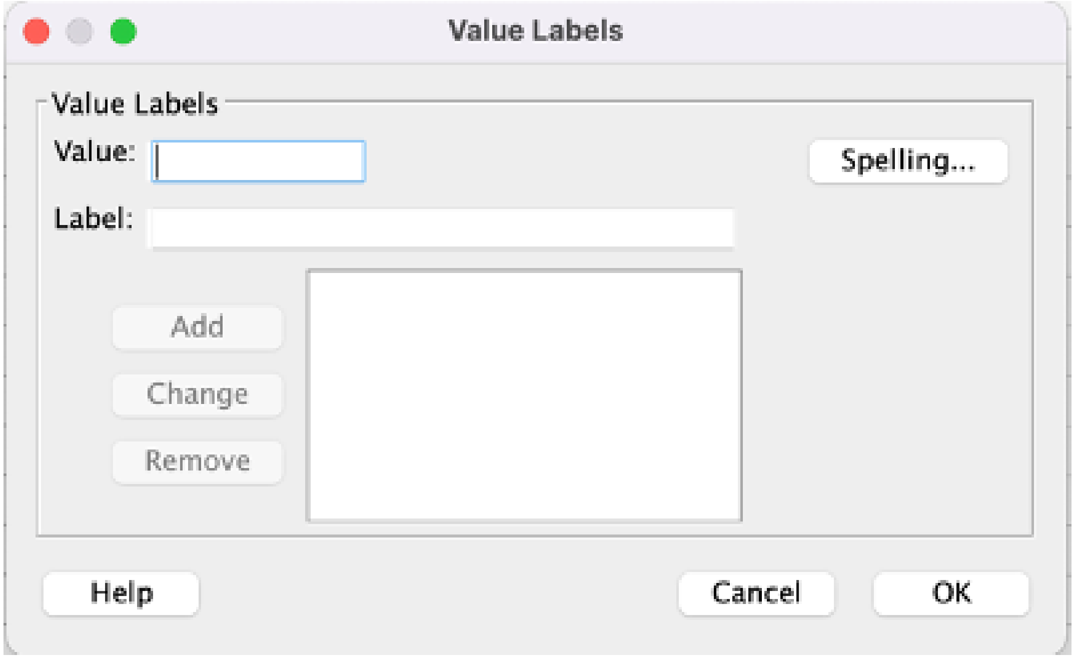
Decimal Places: 0

The Numeric type honors the digit grouping setting, while the Restricted Numeric never uses digit grouping.

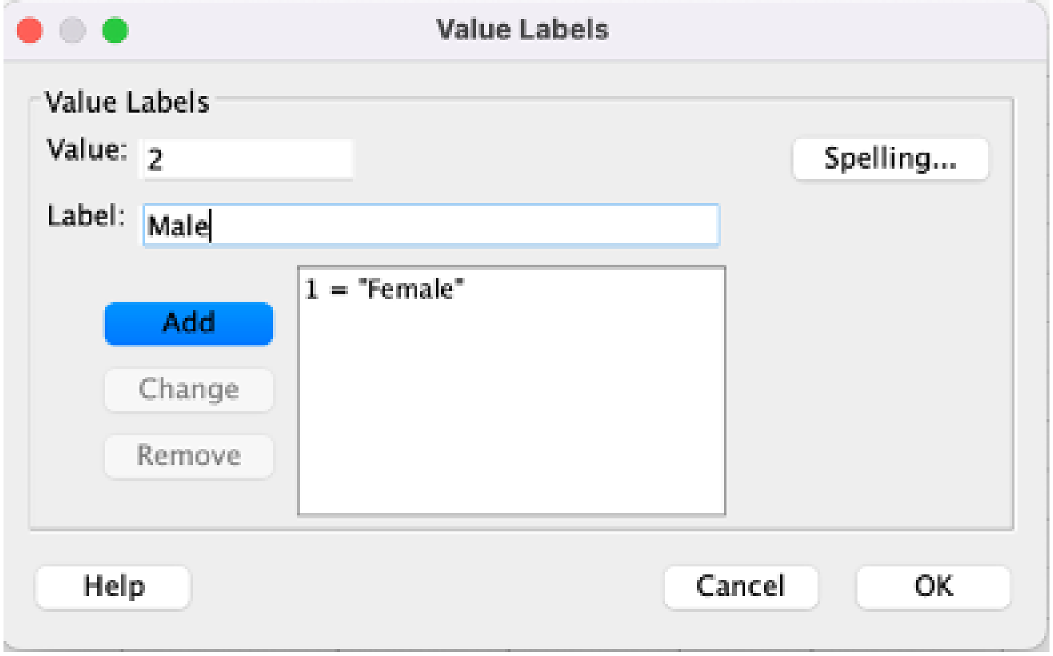
`Width` refers to the maximum number of characters in a given variable. I don't think I've ever changed this from 8.

`Decimals` allows you to specify if a value shows decimal places or not. I've used `0` in some screengrabs below, so that decimal places are not shown, and 2 decimal places in others. I would suggest sticking with the default of 2, unless you are using very precise measurements.

`Label` is VERY handy. It doesn't have the restrictions of `Name`, and so you can give a variable a really clear informative label. And it shows up in the output, meaning that the output is more easily understood. Do take the time to use this field, as once you step away from a dataset for even a couple of hours, it's possible to forget what a variable is or represents.



`Value` is particularly useful for categorical variables (or nominal, named variables as SPSS prefers). Assign every group a name and SPSS will attach that name to the output too. It's easier to compare output with the labels 'Females' and 'Males' than '1' and '2'.



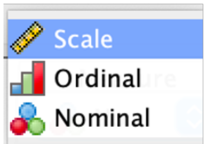
Value: 2

Label: Male

1 = "Female"

Enter the numeric value for your group, enter the label and click 'add' and your label will be stored.

`Measure` allows you to specify the measurement type for a variable, and sometimes this allows SPSS to deal with the data more accurately, so it is important.

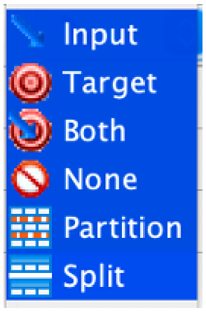


Use `Scale` if you are entering interval or ration data (i.e. where the difference between scores on the scale is meaningful and standardised; and intervals across the scale are equal) such as response times or accuracy rates.

`Ordinal` if you have a categorical variable where the order of the categories matter, such as Lickert Scales,

and `Nominal` if you have a categorical variable where the order of categories does not matter.

You can also further classify your data according to `Role` , but this is not crucial.



`Input`. The variable will be used as an input (e.g., predictor, independent variable).

`Target`. The variable will be used as an output or target (e.g., dependent variable).

`Both`. The variable will be used as both input and output. This is the case with some aspects of the 2x2 design, but don't get caught up with this.

`None`. The variable has no role assignment.

`Partition`. The variable will be used to partition the data into separate samples for training, testing, and validation. Or if you will exclude some participants based on the value.

`Split`. Is used for advanced functions I won't go into here.

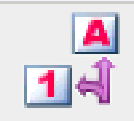
Inputting your data for the various 2x2 designs.

In this series of toy examples, I'm running an experiment examining the effect of Puzzle Difficulty (IV1 with 2 levels - Easy, Hard) and Background Music (IV2 with 2 levels - Slow, Fast) on Solving Time (DV in Seconds).

You will see that I couldn't quite make up my mind on what the optimal 2x2 design was for this study, and so we can try it in all 3 'flavours' of 2x2 or two-way ANOVA

- Independent (or between-groups) ANOVA
- Repeated Measures (paired samples or within-participant) ANOVA
- Mixed ANOVA

For each design, I'll include a participant ID number (which is good practice) and I shall show the **Data View** with **numeric values** on the left and with the **variable labels** on the right.



This button toggles between the two and allows you to quickly see what you are working with.

I'll then show you the `Variable View`, so that you can see how they are set up.

If you know what your specific design is, you can jump to that section below, but I recommend you take the time to think about how data are organised for all of the 3 'flavours', as this will be useful next year.

Tip: Syntax

Even though your dataset may not yet be complete, you can dry-run the analysis, and then re-run it using 'Syntax'. Simply put, syntax is the code that runs the SPSS analysis behind the scenes.
This is something you could ask about in the labs if you are interested. It's not something we usually teach.

2x2 Independent (between-groups) ANOVA

> In an independent ANOVA, each participant belongs in only one condition or cell in your 2x2 design grid, meaning that they are allocated to only one condition for each of your IVs out of the possible 4.

So in the example of my experiment examining the effect of Puzzle Difficulty (IV1 with 2 levels - Easy, Hard) and Background Music (IV2 with 2 levels - Slow, Fast) on Solving Time (DV in Seconds), I would randomly allocate people to solve one puzzle ONLY after informed consent. So there are four conditions and a participant in my study will only do one of them.

An easy puzzle with slow background music, an easy puzzle with fast background music, a difficult puzzle with slow background music or a difficult puzzle with fast background music.

You need to label the condition each participant was in for each of your IVs separately so that SPSS knows what puzzle condition they got and what music condition they got.

As always, each row corresponds to a single participant.

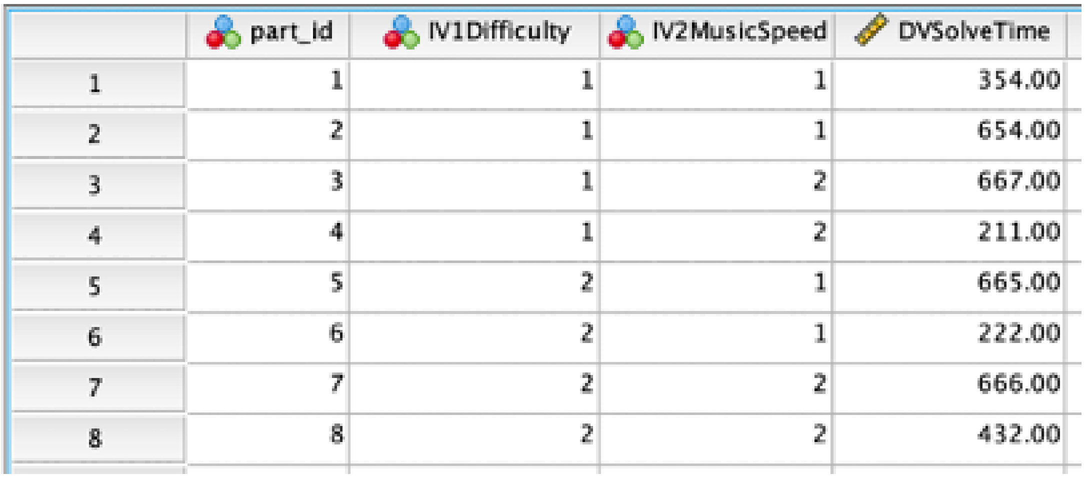
In this example, I've called Puzzle Difficulty `IV1Difficulty` so that you can replace this in your mind with your first IV. I've used the `Label` and `Values` to identify Easy puzzle group with 1 and the difficult puzzle group with 2. `IV2MusicSpeed` uses 1 to signify the Slow music condition and 2 to signify the Fast music condition

The first panel shows this in data view with `numeric values` , the right panel with `value labels`. The `Variable View` shows how I set this up.

The DV in my study is `DVSolveTime` and is how many seconds it took to solve the puzzle.

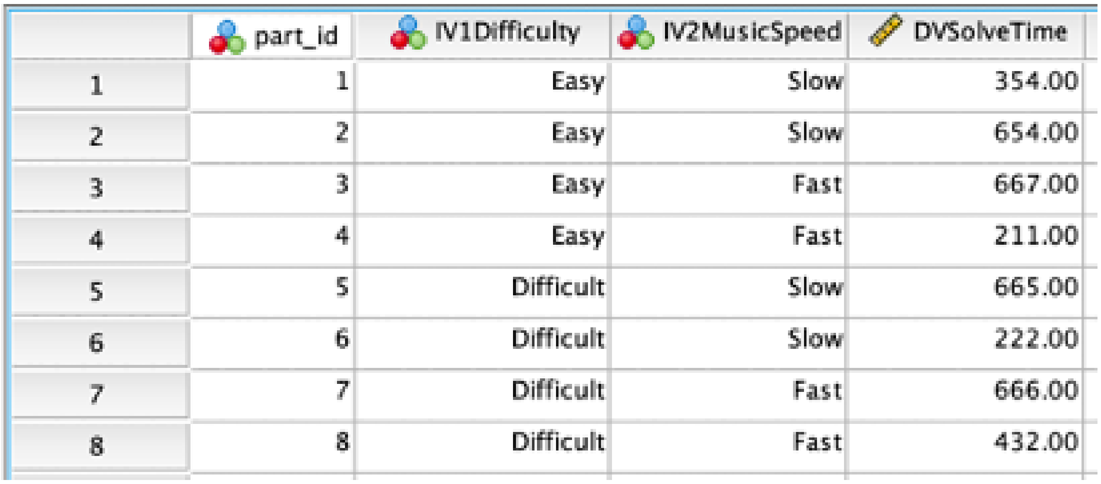
Download the 2x2 Independent ANOVA dataset here

Data view numeric values



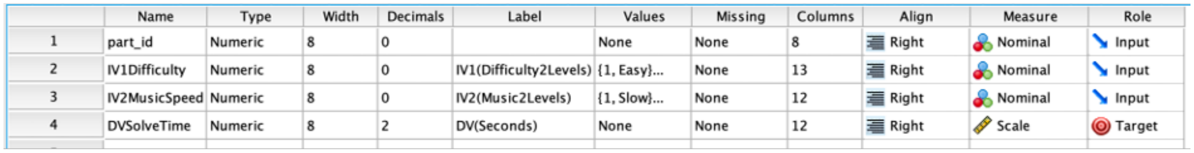
	part_id	IV1Difficulty	IV2MusicSpeed	DVSolveTime
1	1	1	1	354.00
2	2	1	1	654.00
3	3	1	2	667.00
4	4	1	2	211.00
5	5	2	1	665.00
6	6	2	1	222.00
7	7	2	2	666.00
8	8	2	2	432.00

Data view with variable labels



	part_id	IV1Difficulty	IV2MusicSpeed	DVSolveTime
1	1	Easy	Slow	354.00
2	2	Easy	Slow	654.00
3	3	Easy	Fast	667.00
4	4	Easy	Fast	211.00
5	5	Difficult	Slow	665.00
6	6	Difficult	Slow	222.00
7	7	Difficult	Fast	666.00
8	8	Difficult	Fast	432.00

Variable view



	Name	Type	Width	Decimals	Label	Values	Missing	Columns	Align	Measure	Role
1	part_id	Numeric	8	0		None	None	8	Right	Nominal	Input
2	IV1Difficulty	Numeric	8	0	IV1(Difficulty2Levels)	{1, Easy}...	None	13	Right	Nominal	Input
3	IV2MusicSpeed	Numeric	8	0	IV2(Music2Levels)	{1, Slow}...	None	12	Right	Nominal	Input
4	DVSolveTime	Numeric	8	2	DV(Seconds)	None	None	12	Right	Scale	Target

2x2 Repeated Measures ANOVA

> In a repeated measures ANOVA, each participant does all four conditions or cells in your 2x2 design grid. They would have been given all four conditions from a possible four.

So, in the example of my experiment examining the effect of Puzzle Difficulty (IV1 with 2 levels - Easy, Hard) and Background Music (IV2 with 2 levels - Slow, Fast) on Solving Time (DV in Seconds), I would randomly sequence the puzzles and music, but a participant in my study would solve four puzzles making up all the possible combinations of difficulty and music. In short, both levels of both of my IVs.

Every participant would have solved an easy puzzle with slow background music, an easy puzzle with fast background music, a difficult puzzle with slow background music and a difficult puzzle with fast background music. Busy day!

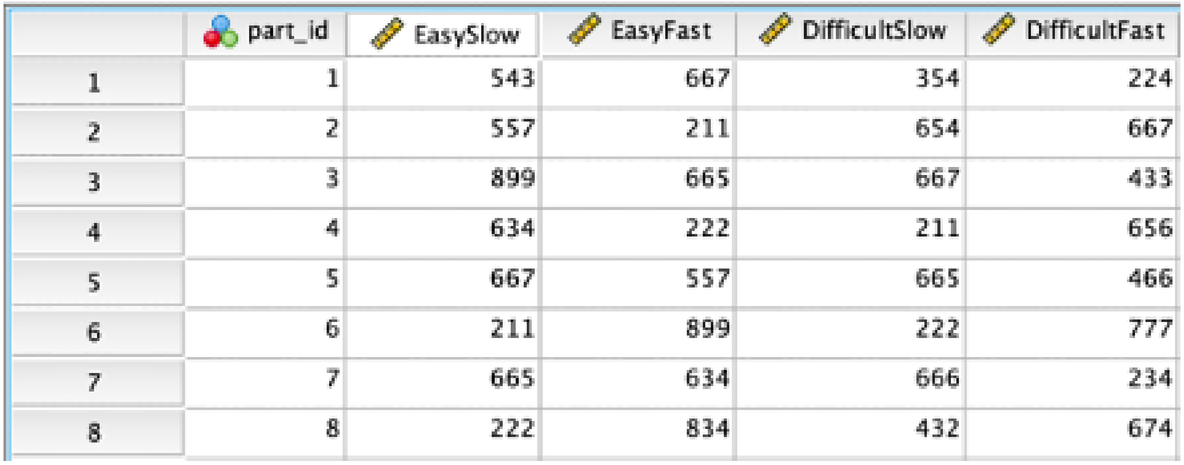
In the previous example, we had one DV measure and needed to tell SPSS which condition the participant was in. In this example, we have 4 measures of the DV (how long it took to solve each puzzle) so we need to let SPSS see which response time corresponds to each condition.

As always, each row corresponds to a single participant.

In a repeated measures ANOVA setup, we don't need to use numbered groups or labels, instead we rely on using `Name` in `Variable View` to identify each of the 4 conditions the participant experienced. So this is one of the simplest data sets.

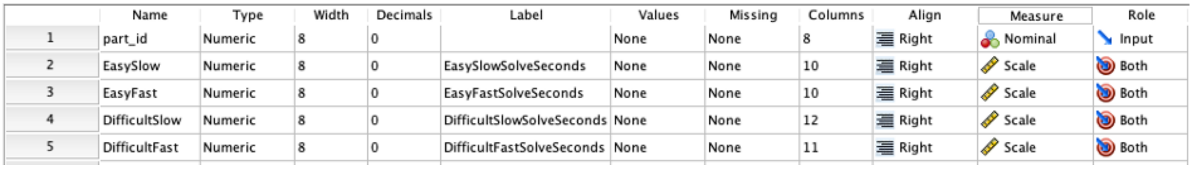
Download the 2x2 Repeated Measures ANOVA dataset here

Data view numeric values



	part_id	EasySlow	EasyFast	DifficultSlow	DifficultFast
1	1	543	667	354	224
2	2	557	211	654	667
3	3	899	665	667	433
4	4	634	222	211	656
5	5	667	557	665	466
6	6	211	899	222	777
7	7	665	634	666	234
8	8	222	834	432	674

Variable View



	Name	Type	Width	Decimals	Label	Values	Missing	Columns	Align	Measure	Role
1	part_id	Numeric	8	0		None	None	8	Right	Nominal	Input
2	EasySlow	Numeric	8	0	EasySlowSolveSeconds	None	None	10	Right	Scale	Both
3	EasyFast	Numeric	8	0	EasyFastSolveSeconds	None	None	10	Right	Scale	Both
4	DifficultSlow	Numeric	8	0	DifficultSlowSolveSeconds	None	None	12	Right	Scale	Both
5	DifficultFast	Numeric	8	0	DifficultFastSolveSeconds	None	None	11	Right	Scale	Both

2x2 Mixed ANOVA

As you know, a repeated-measures ANOVA contains only **within-participant** variables (where participants take part in *all* conditions).

An independent ANOVA uses only **between-group** variables (where participants only take part in one condition),

> A mixed ANOVA contains **BOTH** variable types. In this case, one of each. You will necessarily have one within-participant IV and one between-group IV.

In this last toy example using the puzzle music experiment, I decided that participants were allocated to **either** the Easy or the Difficult Puzzle condition (puzzle difficulty is the between-group IV with 2 levels)

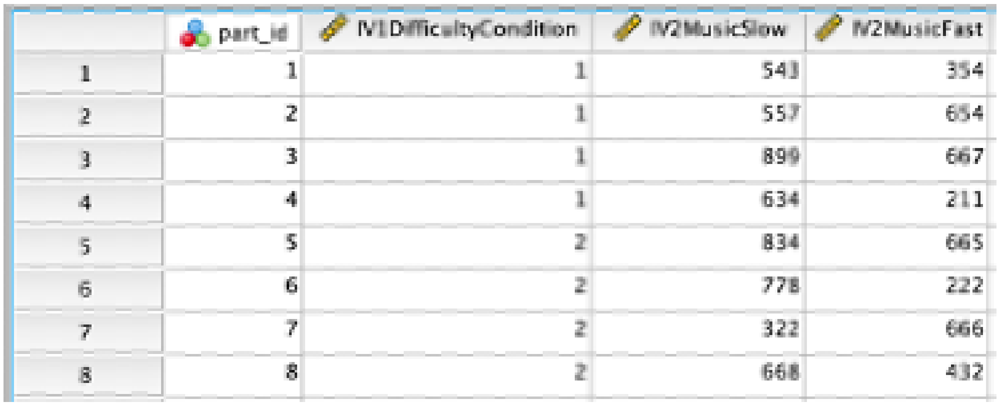
Each participant then did two puzzles with both of the levels of the Background Music condition, one puzzle with Slow background music, and one puzzle with Fast background music (music is the within-participant IV with 2 levels).

As always, each row corresponds to a single participant.

In this case, each participant solved 2 puzzles and so needs two measures, but we need to identify the condition of the between-group IV that they were in Easy Puzzles or Difficult Puzzles.

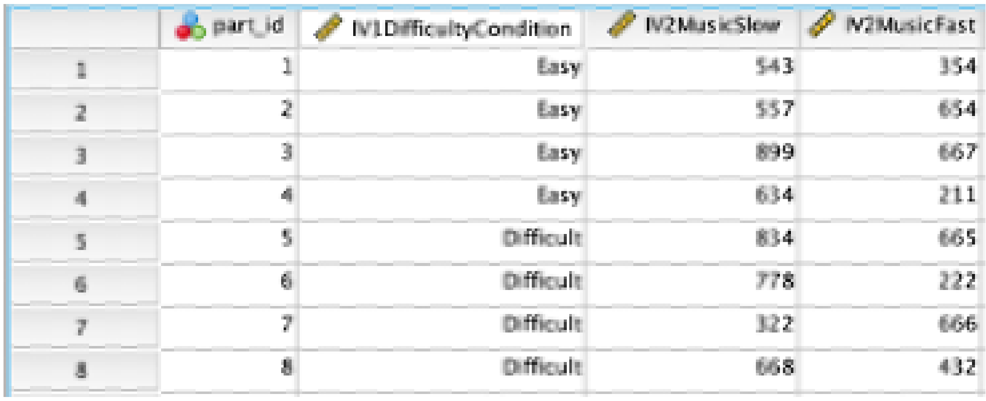
Download the 2x2 Mixed ANOVA dataset here

Data view numeric values



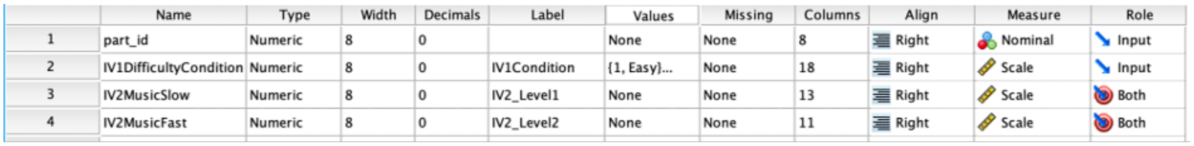
	part_id	IV1DifficultyCondition	IV2MusicSlow	IV2MusicFast
1	1	1	543	354
2	2	1	557	654
3	3	1	899	667
4	4	1	634	211
5	5	2	834	665
6	6	2	778	222
7	7	2	322	666
8	8	2	668	432

Data view with variable labels



	part_id	IV1DifficultyCondition	IV2MusicSlow	IV2MusicFast
1	1	Easy	543	354
2	2	Easy	557	654
3	3	Easy	899	667
4	4	Easy	634	211
5	5	Difficult	834	665
6	6	Difficult	778	222
7	7	Difficult	322	666
8	8	Difficult	668	432

Variable view



	Name	Type	Width	Decimals	Label	Values	Missing	Columns	Align	Measure	Role
1	part_id	Numeric	8	0		None	None	8	Right	Nominal	Input
2	IV1DifficultyCondition	Numeric	8	0	IV1Condition	{1, Easy}...	None	18	Right	Scale	Input
3	IV2MusicSlow	Numeric	8	0	IV2_Level1	None	None	13	Right	Scale	Both
4	IV2MusicFast	Numeric	8	0	IV2_Level2	None	None	11	Right	Scale	Both

Some useful calculations in SPSS

Sometimes your data is not yet entirely ready for analysis and you need to calculate a value to make your data make sense or to permit further analysis. I shall present two useful calculations here.

It might be that you need to sum all the questions in a questionnaire to get a grand total. Or you need to find the average score on a group of questions or trials in an experiment.

I shall use that last example, **calculating a mean score** on a set of trials, to illustrate this functionality, but you can do lots of useful things the same way, simply by inserting variables in the SPSS dataset into a calculation.

Another thing that many of you may need to do is to perform a **median split** , where you take a continuous variable such as a score on a personality measure and 'cut it in half' so that 50% of your sample is considered low, and 50% is considered high.

So here's an experiment that needs me to do both of these things.

It's a 2x2 Mixed design experiment I ran recently, with 16 individuals in my dataset.

I have allocated each participant a unique ID number, and I have age and gender as demographic characteristics.

I have a variable to confirm their having given Informed Consent, and at this point I have already removed any identifying information, and so this is anonymous.

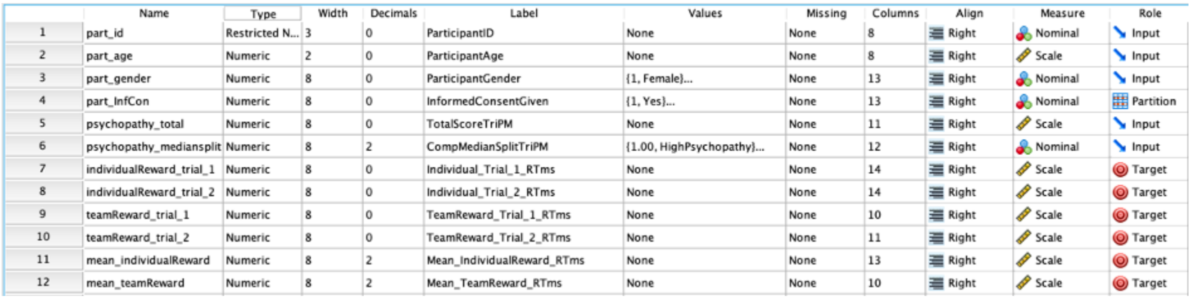
Participants completed a Psychopathy Questionnaire (The Triarchic Psychopathy Measure – TriPM) giving me a possible score of between 0 and 174 for each participant. I shall return to this measure when talking about SPSS Scoring Syntax and Syntax more generally.

They then took part in a decision-making task, where they responded to choices in 2 different scenarios. In one scenario they would be rewarded as part of a team, in another they would be rewarded individually. They each responded to two trials in each of the two conditions (within-subject IV with 2 levels - manipulation of reward type: team reward or individual reward).

Reaction Time was recorded for each trial to indicate how quickly the participant made the decision.

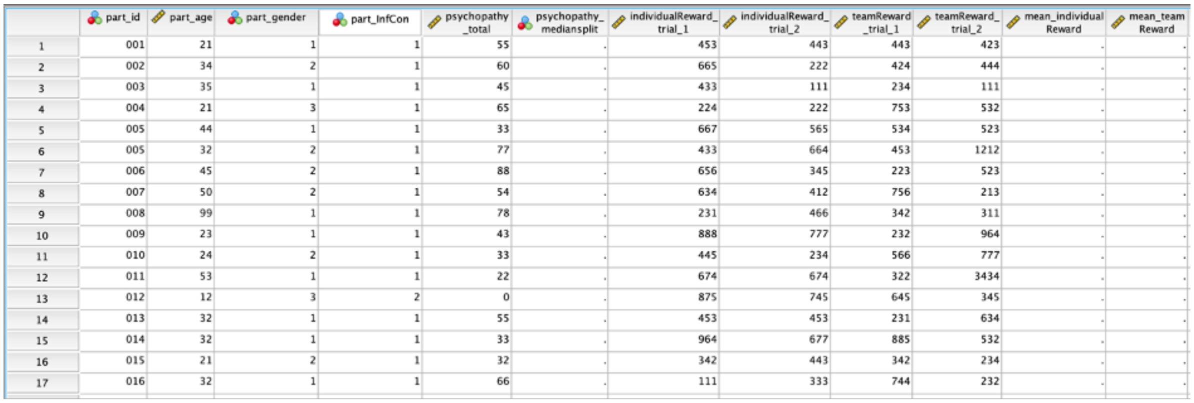
Below is a snapshot of the Variable View and Data View. You'll see that I need to complete the Median Split variable, and the two mean scores variables at the right hand side.

Variable View.



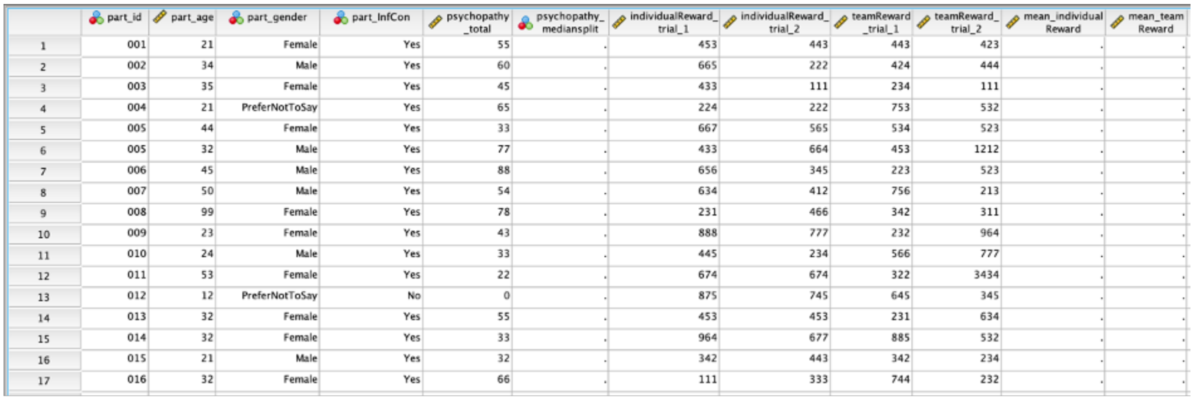
	Name	Type	Width	Decimals	Label	Values	Missing	Columns	Align	Measure	Role
1	part_id	Restricted N...	3	0	ParticipantID	None	None	8	Right	Nominal	Input
2	part_age	Numeric	2	0	ParticipantAge	None	None	8	Right	Scale	Input
3	part_gender	Numeric	8	0	ParticipantGender	{1, Female}...	None	13	Right	Nominal	Input
4	part_InfCon	Numeric	8	0	InformedConsentGiven	{1, Yes}...	None	13	Right	Nominal	Partition
5	psychopathy_total	Numeric	8	0	TotalScoreTriPM	None	None	11	Right	Scale	Input
6	psychopathy_mediansplit	Numeric	8	2	CompMedianSplitTriPM	{1.00, HighPsychopathy}...	None	12	Right	Nominal	Input
7	individualReward_trial_1	Numeric	8	0	Individual_Trial_1_RTms	None	None	14	Right	Scale	Target
8	individualReward_trial_2	Numeric	8	0	Individual_Trial_2_RTms	None	None	14	Right	Scale	Target
9	teamReward_trial_1	Numeric	8	0	TeamReward_Trial_1_RTms	None	None	10	Right	Scale	Target
10	teamReward_trial_2	Numeric	8	0	TeamReward_Trial_2_RTms	None	None	11	Right	Scale	Target
11	mean_individualReward	Numeric	8	2	Mean_IndividualReward_RTms	None	None	13	Right	Scale	Target
12	mean_teamReward	Numeric	8	2	Mean_TeamReward_RTms	None	None	10	Right	Scale	Target

Data View with numeric values (e.g. gender 1,2,3)



	part_id	part_age	part_gender	part_InfCon	psychopathy_total	psychopathy_mediansplit	individualReward_trial_1	individualReward_trial_2	teamReward_trial_1	teamReward_trial_2	mean_individualReward	mean_teamReward
1	001	21	1	1	55	.	453	443	443	423	.	.
2	002	34	2	1	60	.	665	222	424	444	.	.
3	003	35	1	1	45	.	433	111	234	111	.	.
4	004	21	3	1	65	.	224	222	753	532	.	.
5	005	44	1	1	33	.	667	565	534	523	.	.
6	005	32	2	1	77	.	433	664	453	1212	.	.
7	006	45	2	1	88	.	656	345	223	523	.	.
8	007	50	2	1	54	.	634	412	756	213	.	.
9	008	99	1	1	78	.	231	466	342	311	.	.
10	009	23	1	1	43	.	888	777	232	964	.	.
11	010	24	2	1	33	.	445	234	566	777	.	.
12	011	53	1	1	22	.	674	674	322	3434	.	.
13	012	12	3	2	0	.	875	745	645	345	.	.
14	013	32	1	1	55	.	453	453	231	634	.	.
15	014	32	1	1	33	.	964	677	885	532	.	.
16	015	21	2	1	32	.	342	443	342	234	.	.
17	016	32	1	1	66	.	111	333	744	232	.	.

Data View with value labels (using the value labels button above) (see gender)



	part_id	part_age	part_gender	part_InfCon	psychopathy_total	psychopathy_mediansplit	individualReward_trial_1	individualReward_trial_2	teamReward_trial_1	teamReward_trial_2	mean_individualReward	mean_teamReward
1	001	21	Female	Yes	55	.	453	443	443	423	.	.
2	002	34	Male	Yes	60	.	665	222	424	444	.	.
3	003	35	Female	Yes	45	.	433	111	234	111	.	.
4	004	21	PreferNotToSay	Yes	65	.	224	222	753	532	.	.
5	005	44	Female	Yes	33	.	667	565	534	523	.	.
6	005	32	Male	Yes	77	.	433	664	453	1212	.	.
7	006	45	Male	Yes	88	.	656	345	223	523	.	.
8	007	50	Male	Yes	54	.	634	412	756	213	.	.
9	008	99	Female	Yes	78	.	231	466	342	311	.	.
10	009	23	Female	Yes	43	.	888	777	232	964	.	.
11	010	24	Male	Yes	33	.	445	234	566	777	.	.
12	011	53	Female	Yes	22	.	674	674	322	3434	.	.
13	012	12	PreferNotToSay	No	0	.	875	745	645	345	.	.
14	013	32	Female	Yes	55	.	453	453	231	634	.	.
15	014	32	Female	Yes	33	.	964	677	885	532	.	.
16	015	21	Male	Yes	32	.	342	443	342	234	.	.
17	016	32	Female	Yes	66	.	111	333	744	232	.	.

Performing a Median Split in SPSS

Performing a median split allows you to categorise participants into discreet groups based on a continuous variable. You turn a continuous variable into a dichotomous variable (low & high, for example). The median is used because it is the number at which point 50% of values lie above, and 50% lie below. This means that usually, you get a nice equal sample size in both groups.

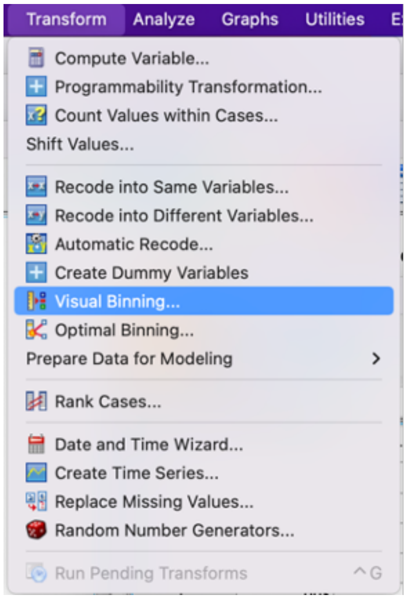
The median split procedure is sometimes better than using a cut off score or threshold, as a median split uses the data that you have in hand, and doesn't assume anything about the sample of scores that you have.

I want to categorise my 16 participants into 'LowPsychopathy' and 'HighPsychopathy' groups of equal size (hopefully) using a **median split** procedure (essentially turning this into a quasi-experimental between-group IV with 2 levels).

I could just work out the median with descriptive statistics, and then by hand label each person as lower than the median or higher than the median. With 16 participants, it might be quicker to do it that way, but with more it would be very dull indeed and I'd probably still get mixed up and make a mistake. So, I want to find a way to do it automatically and accurately.

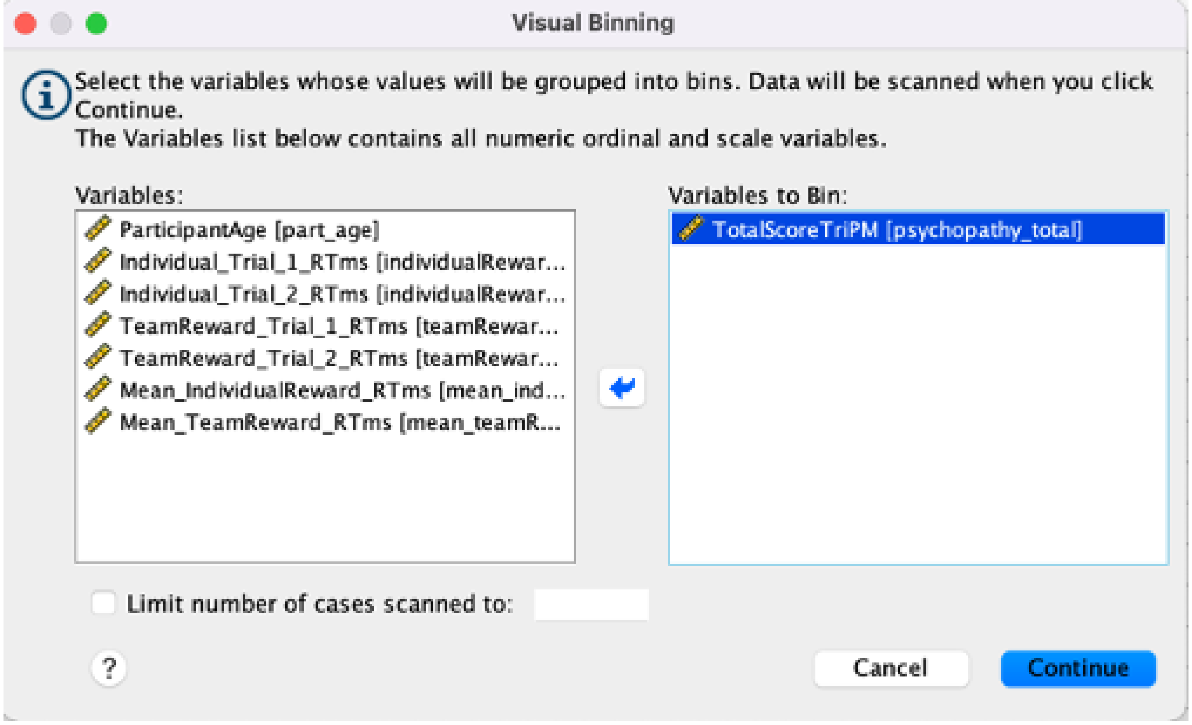
Here is the step-by-step procedure.

Go to the `Transform` menu and select `Visual Binning`



Simply put, this allows you to put participants into 'bins' based on a set of rules you make.

Select the variable you are going to use to divide people up. In this case, I choose the total score for my TriPM Psychopathy measure (`TotalScoreTriPm`), move it into the `Variables to Bin` field and click `continue`.



Select the variables whose values will be grouped into bins. Data will be scanned when you click Continue.
The Variables list below contains all numeric ordinal and scale variables.

Variables:

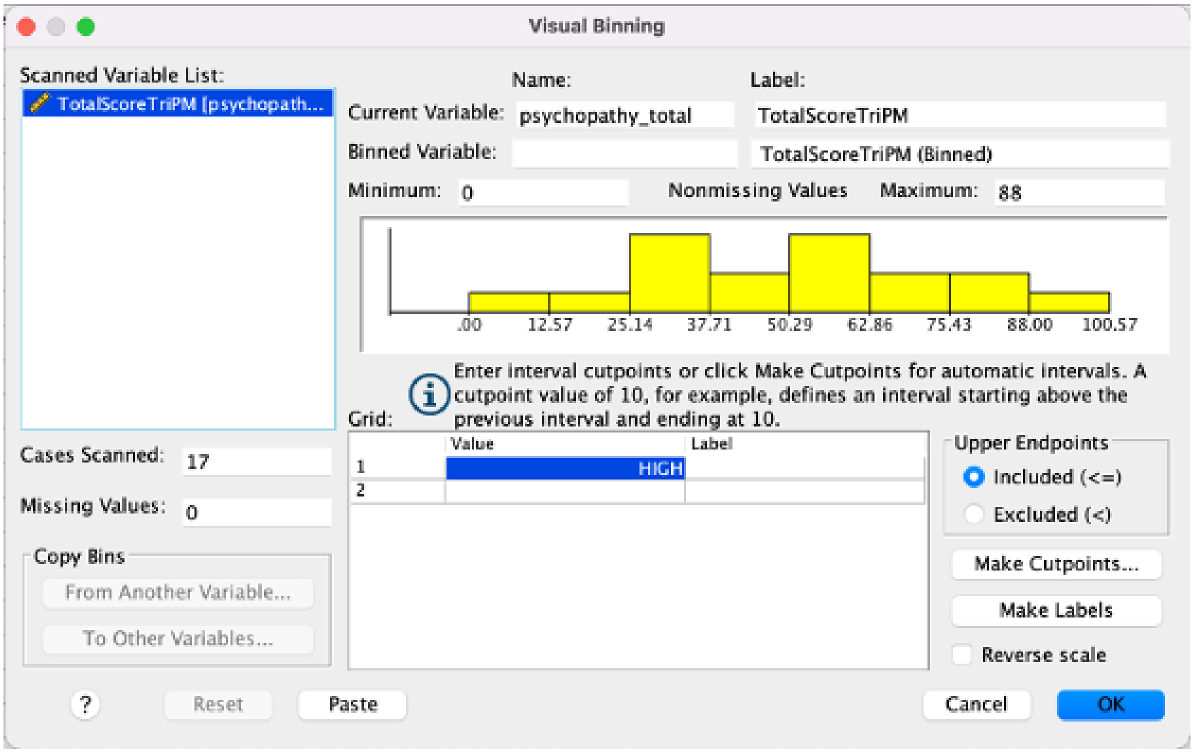
- ParticipantAge [part_age]
- Individual_Trial_1_RTms [individualRewar...
- Individual_Trial_2_RTms [individualRewar...
- TeamReward_Trial_1_RTms [teamRewar...
- TeamReward_Trial_2_RTms [teamRewar...
- Mean_IndividualReward_RTms [mean_ind...
- Mean_TeamReward_RTms [mean_teamR...

Variables to Bin:

- TotalScoreTriPM [psychopathy_total]

□ Limit number of cases scanned to:

I'm then presented with this view of the distribution of the data.



Scanned Variable List: TotalScoreTriPM [psychopath...

	Name:	Label:
Current Variable:	psychopathy_total	TotalScoreTriPM
Binned Variable:		TotalScoreTriPM (Binned)

Minimum: 0 Nonmissing Values Maximum: 88

Enter interval cutpoints or click Make Cutpoints for automatic intervals. A cutpoint value of 10, for example, defines an interval starting above the previous interval and ending at 10.

Grid:

	Value	Label
1	HIGH	
2		

Cases Scanned: 17

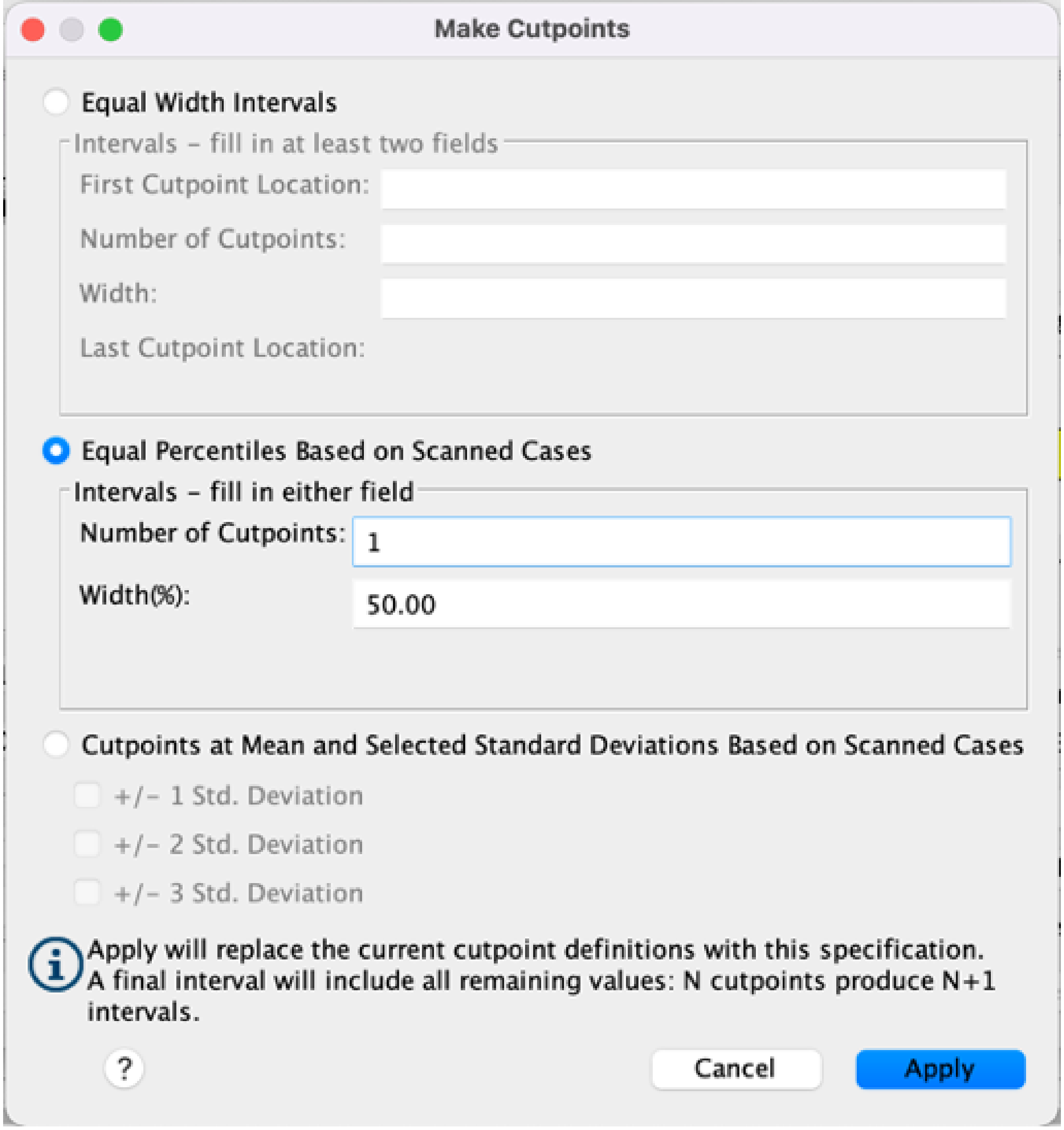
Missing Values: 0

Upper Endpoints
● Included (<=)
○ Excluded (<)

□ Reverse scale

So we want to find the median, or the point at which we cut the data in half. We click the `Make Cutpoints` button.

In the section marked `Equal percentiles based on scannedcases` **we input 1 in the** `number of cutpoints` **box and it automagically fills** 50% **in the** width **box. Click** `Apply`



Make Cutpoints

○ Equal Width Intervals

Intervals – fill in at least two fields

First Cutpoint Location:

Number of Cutpoints:

Width:

Last Cutpoint Location:

● Equal Percentiles Based on Scanned Cases

Intervals – fill in either field

Number of Cutpoints: 1

Width(%): 50.00

○ Cutpoints at Mean and Selected Standard Deviations Based on Scanned Cases

□ +/– 1 Std. Deviation

□ +/– 2 Std. Deviation

□ +/– 3 Std. Deviation

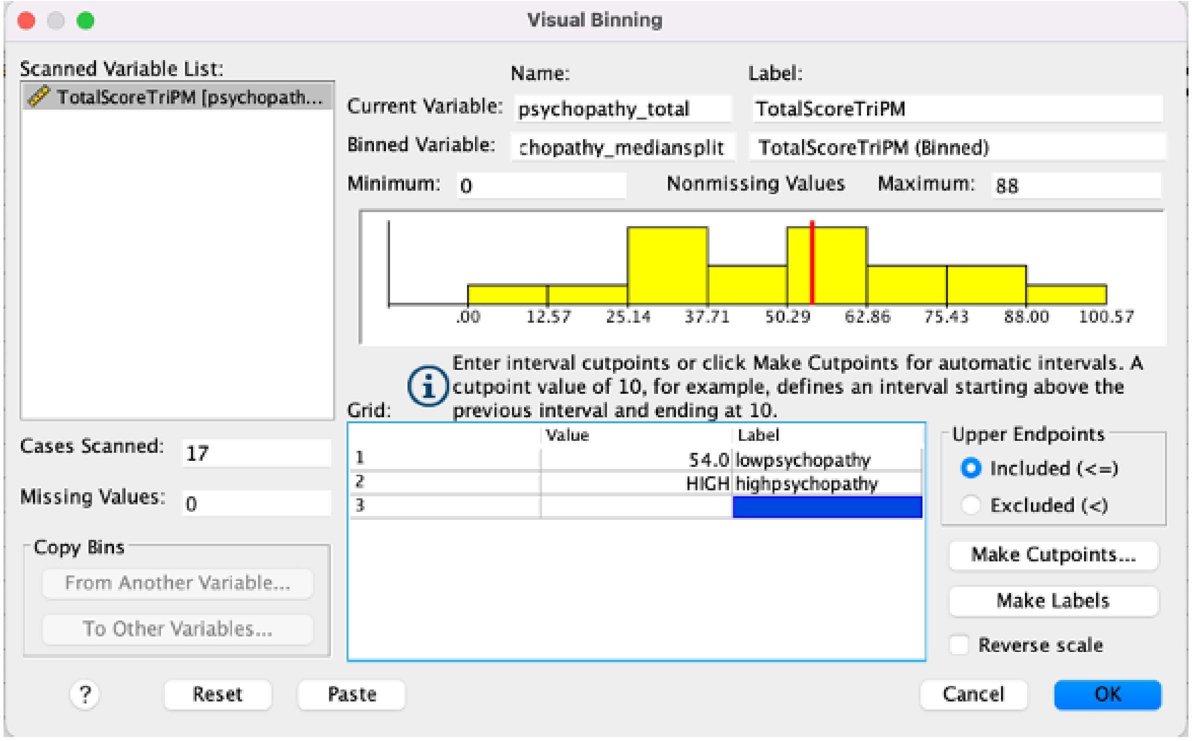
Apply will replace the current cutpoint definitions with this specification. A final interval will include all remaining values: N cutpoints produce N+1 intervals.

Think about that. We asked SPSS to cut our variable into equal groups based on observed data with a single chop! It could only ever be at the Median! It's a funny set of commands, but makes sense, right?

If we asked for 2 cutpoints, it would give Low, Medium and High groups cut at 33.33% and 66.66%. Two cuts with equal numbers on each side = 3 groups – a tertile split or trichotomized variable. But I digress.

Now just label your new variable in the `Binned Variable` box and label the values.

You'll notice that for the data I used, the value is up to and including 54 as low and anything over 54 is considered high. Bearing in mind the scale goes up to 174, and cutoff are usually much higher than 54, the median split is better for my data than that cutoff. There wouldn't be ANYBODY in the high group.



Scanned Variable List: TotalScoreTriPM [psychopath...

	Name:	Label:
Current Variable:	psychopathy_total	TotalScoreTriPM
Binned Variable:	chopathy_mediansplit	TotalScoreTriPM (Binned)

Minimum: 0 Nonmissing Values Maximum: 88

Enter interval cutpoints or click Make Cutpoints for automatic intervals. A cutpoint value of 10, for example, defines an interval starting above the previous interval and ending at 10.

Grid:

	Value	Label
1	54.0	lowpsychopathy
2	HIGH	highpsychopathy
3		

Cases Scanned: 17

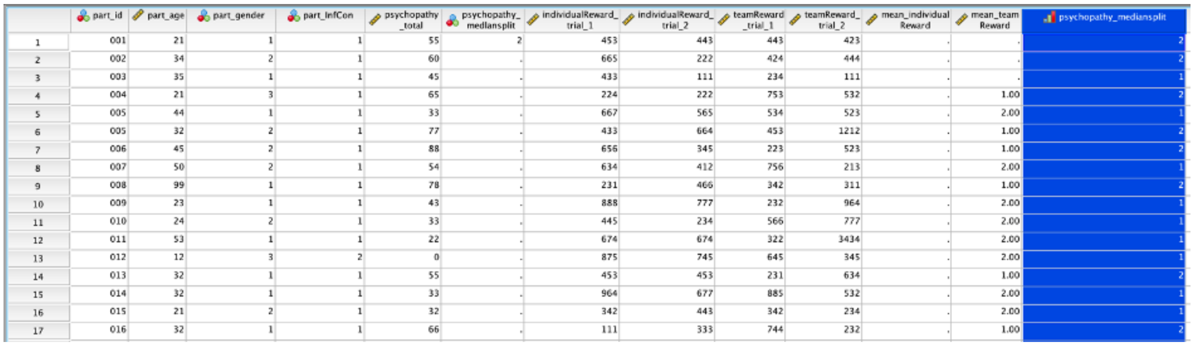
Missing Values: 0

Copy Bins

Upper Endpoints
● Included (<=)
○ Excluded (<)

□ Reverse scale

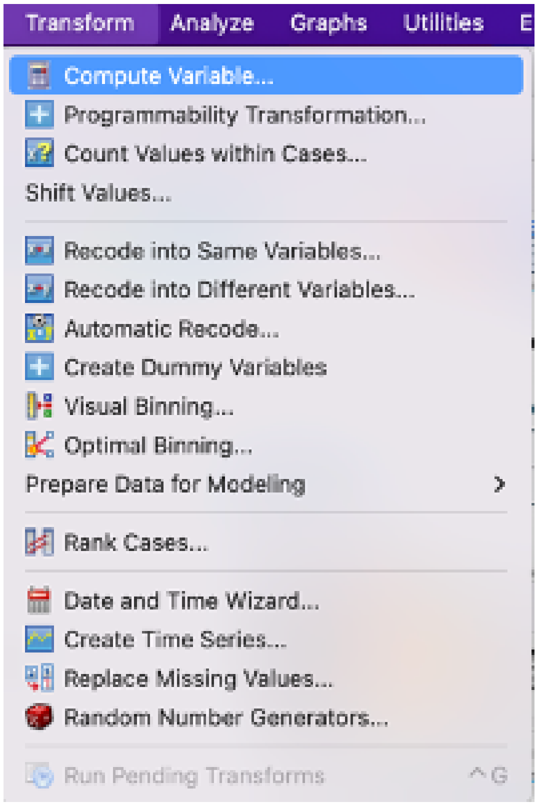
When you click on `OK`, it pops a new variable at the end of the data set with the information you need (highlighted in blue below). From here, I would move that variable to it's rightful place in my dataset, or copy and paste the values into the relevant column if I had already set it up. Alternatively, I could have given it exactly the same name in the `Binned Variable` label setting and it would have replaced the empty cells. But I wanted you to see that new variables usually pop to the far end of the data set.



	part_id	part_age	part_gender	part_InfCon	psychopathy_total	psychopathy_mediansplit	individualReward_trial_1	individualReward_trial_2	teamReward_trial_1	teamReward_trial_2	mean_individualReward	mean_teamReward	psychopathy_mediansplit
1	001	21	1	1	55	2	453	443	443	423	.	.	2
2	002	34	2	1	60	.	665	222	424	444	.	.	2
3	003	35	1	1	45	.	433	111	234	111	.	.	1
4	004	21	3	1	65	.	224	222	753	532	.	1.00	2
5	005	44	1	1	33	.	667	565	534	523	.	2.00	1
6	005	32	2	1	77	.	433	664	453	1212	.	1.00	2
7	006	45	2	1	88	.	656	345	223	523	.	1.00	2
8	007	50	2	1	54	.	634	412	756	213	.	2.00	1
9	008	99	1	1	78	.	231	466	342	311	.	1.00	2
10	009	23	1	1	43	.	888	777	232	964	.	2.00	1
11	010	24	2	1	33	.	445	234	566	777	.	2.00	1
12	011	53	1	1	22	.	674	674	322	3434	.	2.00	1
13	012	12	3	2	0	.	875	745	645	345	.	2.00	1
14	013	32	1	1	55	.	453	453	231	634	.	1.00	2
15	014	32	1	1	33	.	964	677	885	532	.	2.00	1
16	015	21	2	1	32	.	342	443	342	234	.	2.00	1
17	016	32	1	1	66	.	111	333	744	232	.	1.00	2

Calculating a mean score variable

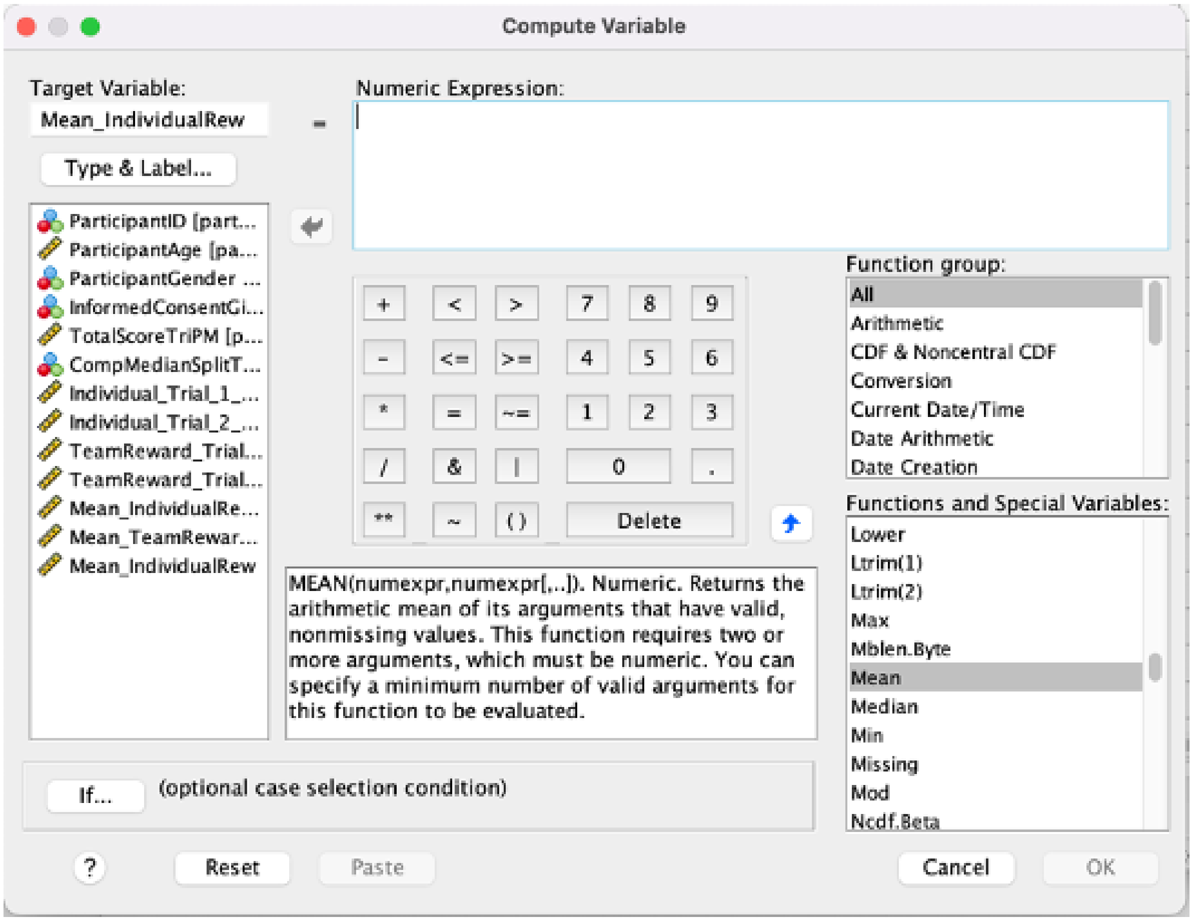
Now to calculate the mean score across a series of variables. In this instance, I want to calculate the mean response time for decisions in the two individual reward trials, so that for each participant I have a mean response time across all trials of that type.



From the `Transform` menu, you want to select `Compute Variable` and the following window will open up.

Firstly, in `Target Variabl`**e** type in the name of the variable you want to calculate. I've chosen '`Mean_IndividualReward`'.

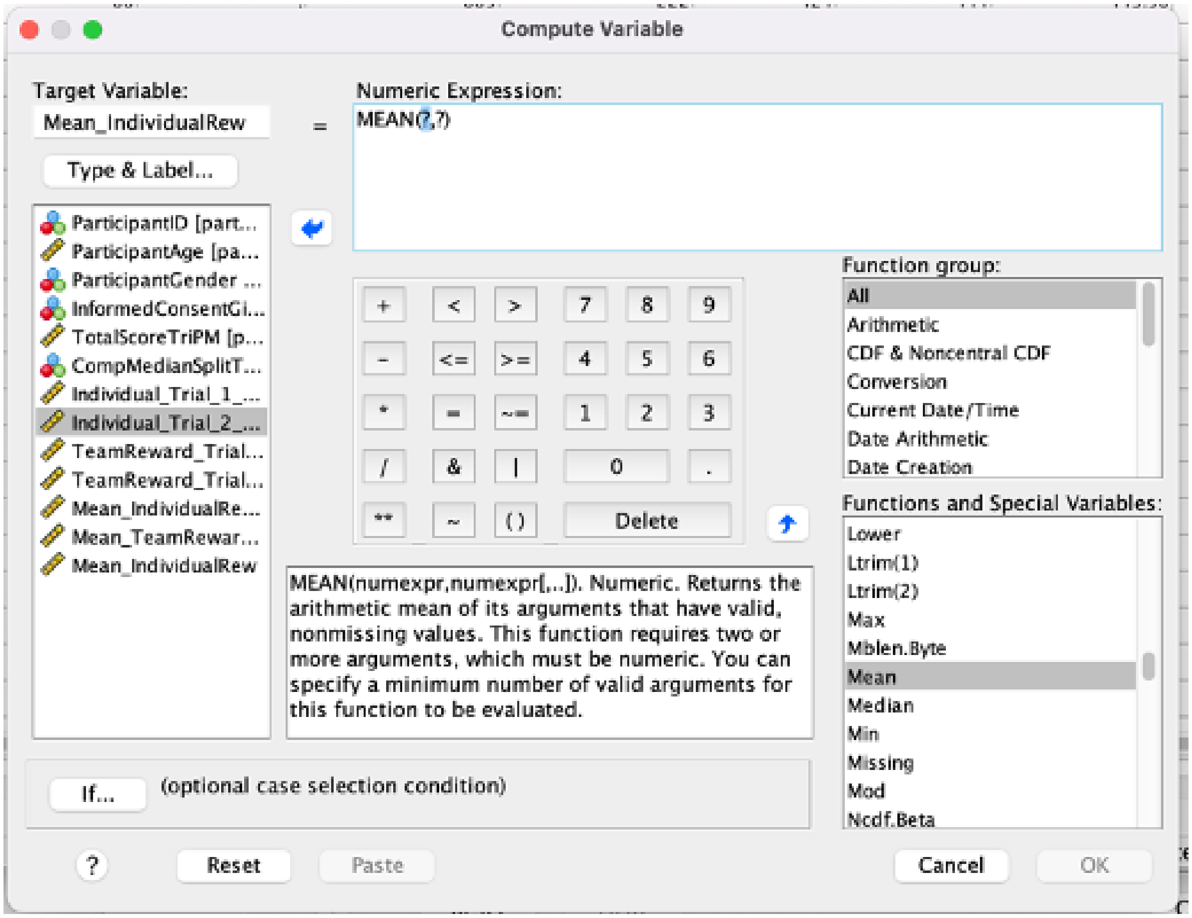
Then double click `All` in the `Function group`**:** dropdown to see all the calculations you could use. Scroll down to `Mean` in the `Functions and Special Variables` dropdown. Have a look at some of the other options. You can see `Sum` and `Sqrt` for example, which is a calculated sum total and square root, respectively.



Target Variable: Mean_IndividualRew

MEAN(numexpr,numexpr[,..]). Numeric. Returns the arithmetic mean of its arguments that have valid, nonmissing values. This function requires two or more arguments, which must be numeric. You can specify a minimum number of valid arguments for this function to be evaluated.

Using the **little blue arrow** pointing upwards, put `Mean` into the `Numeric Expression` box. This is where you put in a simple formula to tell SPSS how to calculate the new variable. You can use this just like a calculator, and insert a variable then + then another variable using the calculator keyboard. It's super easy.



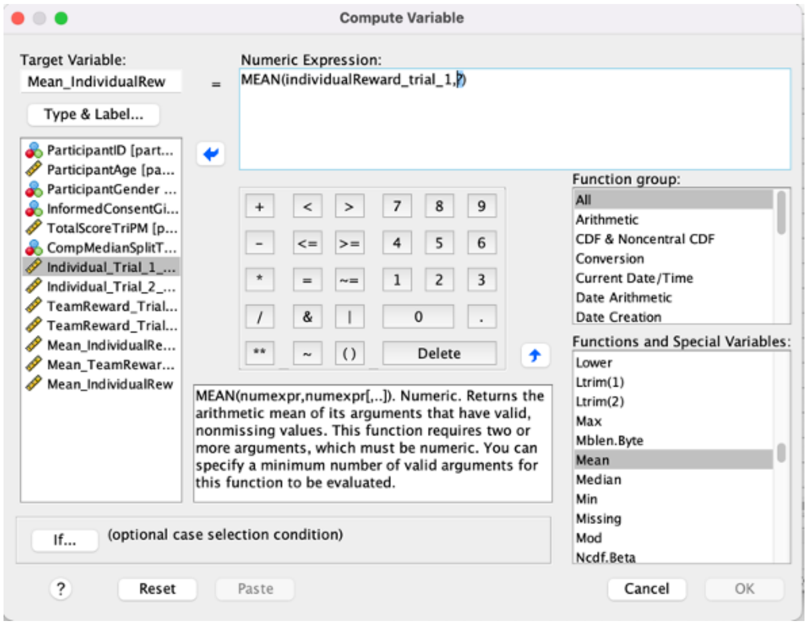
Target Variable: Mean_IndividualRew =

Numeric Expression: MEAN(?,?)

MEAN(numexpr,numexpr[,..]). Numeric. Returns the arithmetic mean of its arguments that have valid, nonmissing values. This function requires two or more arguments, which must be numeric. You can specify a minimum number of valid arguments for this function to be evaluated.

If... (optional case selection condition)

It should look like the window above, and quite simply, you replace the '?' with the variables you want to use from the left hand column. So, you can double click on the variables in the left hand column to send them up to the `Numeric Expression` box and separate them with commas.



Compute Variable

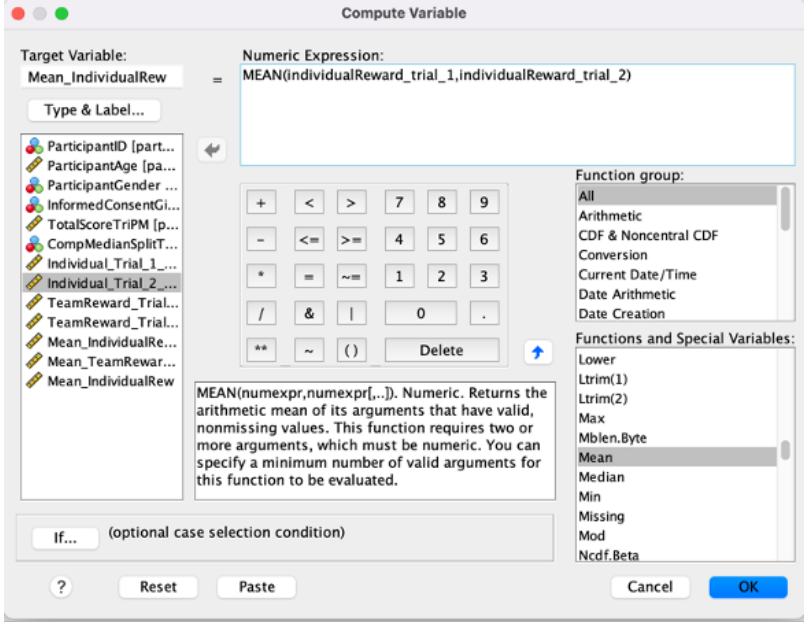
Target Variable: Mean_IndividualRew

Numeric Expression: MEAN(individualReward_trial_1,?)

Function group: All

Functions and Special Variables: Mean

MEAN(numexpr,numexpr[,..]). Numeric. Returns the arithmetic mean of its arguments that have valid, nonmissing values. This function requires two or more arguments, which must be numeric. You can specify a minimum number of valid arguments for this function to be evaluated.



Compute Variable

Target Variable: Mean_IndividualRew

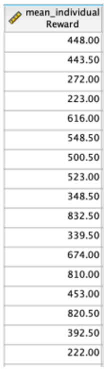
Numeric Expression: MEAN(individualReward_trial_1,individualReward_trial_2)

Function group: All

Functions and Special Variables: Mean

MEAN(numexpr,numexpr[,..]). Numeric. Returns the arithmetic mean of its arguments that have valid, nonmissing values. This function requires two or more arguments, which must be numeric. You can specify a minimum number of valid arguments for this function to be evaluated.

You see I have calculated the mean of 2 trials, but you can do this for any number of variables. This is a useful tool and works like excel formulae. It will then just pop a new column at the end of your dataset with the relevant output of the calculation.



mean_individual Reward
448.00
443.50
272.00
223.00
616.00
548.50
500.50
523.00
348.50
832.50
339.50
674.00
810.00
453.00
820.50
392.50
222.00

Do remember that if you want to perform other calculations, you can find excellent help on the internet. Don't be afraid of trying things out and exploring the features of SPSS. It is confusing at times, but once you understand what it is trying to do, you'll find that it is very clever and can save you time.

⚠ Warning

NOW SAVE YOUR DATA! AND STORE IT IN THE CLOUD!
You do NOT want to have to redo things because your computer crashes!

Well done and stay awesome!!