Mapping Task behavioral data Script

The automated data script for the Mapping Task was made using Python 3.9 and will likely not work on older version. I would suggest opening the script in an editor such as PyCharm with a python 3.9 environment to run it (at the time of writing this, psychopy is still using Python 3.6 which will not work).

The script has two parts:
Data_Extraction_Mapping.py – This is the main script that loops through the files
Behavioral_Mapping_Script.py – This script extracts the data from a single file

You only need to run the *Data_Extraction_Mapping.py* file, which will call a function written in the *Behavioral_Mapping_Script.py* file for each file in the folder. Once it is done it will then save all the data into an excel sheet called "Mapping_Behavioral_Data.xlsx" in a "Results" folder. If these last two do not exist it will create a results folder (in the same directory as the script) and the excel file.

The folder should look something like below, with the scrips and the data in one folder. The data in the data-folder need to be text files, but the naming does not matter that much. Currently it is set up to look for a folder named *"07 – Raw Data/Mapping Task/"* which is part of the overall Data management data structure. This can be changed on lines 15 and 16 of the *Data_Extraction_Mapping.py* file.

Within this raw data folder, it looks for "Canonical" and "Non-Canonical" subfolders. These are created by the mapping task as well. The mapping task also supports "Symbols", to add this folder to the script you simply add the folder name between the brackets on line 15 next to "Canonical" and "Non-Canonical".

…Folder/
- ▪ Data_Extraction_Mapping.py
- ▪ Behavioral_Mapping_Script.py
- ▪ 07 – Raw Data/
  - ↪ ▪ Mapping Task
    - ↪ ▪ Canonical/
      - ↪ ▪ ppn001_MappingTask.dat
      - ↪ ▪ ppn002_MappingTask.dat
    - ↪ ▪ Non-Canonical/
      - ↪ ▪ ppn001_MappingTask.dat
      - ↪ ▪ ppn002_MappingTask.dat
    - ↪ ▪ Results/
      - ↪ ▪ Mapping_Behavioral_Data.xlsx

Data_Extraction_Mapping.py
For the script to work it needs to be in the same folder as the "data" folder, not in the data folder but next to the data folder, because it will try to find a "data" folder in the same directory as itself. This can be specified on lines 15 and 16 though.

It will loop through all the folders specified in the settings structure (written as `[‘Canonical’, ‘Non-Canonical’]` on line 15). Within each folder, it will loop through every .dat file. If the IDE has a debug-window or something similar, a message will appear

```
Processing Folder: Canonical
```

It will proceed by loading in all the .dat files in the data folder and loop through each file, processing it using *Behavioral_Mapping_Script.py*. A message will appear showing which file is currently being processed

```
Processing: PP01.txt
```

If something goes wrong, the message will indicate which file caused an error

```
Error Occurred at file: PP01.txt
```

Upon completion, all the data is saved to an excel file in the Results folder. This can be loaded into an SPSS file or processed further in excel.

Sheet	Description
Bins	A key for the description codes
Univariate_Canonical	▪ A univariate data structure. Each participant has 2 rows, one for each range. ▪ Options: the choices given ▪ Stimulus: the single representation shown at the top ▪ Mean_Acc: Average accuracy (across both ranges) ▪ Mean_RT: Average Response Time (across both ranges) ▪ Mean_RT_Cor: Average Response Time Correct only (across both ranges) ▪ Range: Numerical Range ▪ Acc: Average accuracy ▪ RT_Overall: Average Response Time ▪ RT_Correct: Average Response Time Correct only
Univariate_Non-Canonical	
Multivariate_Canonical	▪ A multivariate version of the above data
Multivariate_Non-Canonical	

Behavioral_Mapping_Script.py

This script extract the data for individual files, the main script will call the “behavioral_mapping_runner” function that outputs the data from a single text file.

The first thing it does is check which line contains “Trial_Number” which, at the time of writing, denotes where the participant data starts.

```
Number-Hand Mapping Task
Participant;1
Options;Numbers
Stimulus;Canonical
Reps;2
Blocks;1
Version;3.02021
Date and Time;10-02-2021 10:28:40

Mapping Task


Block;Trial_Number;Number Shown;Key Pressed;Correct;RT;Mirror    ← find this line
1;1;2;0;1;0.471;0
1;2;1;0;0;0.78;1
1;3;3;0;1;0.322;0
```

Participant Information is the first block, everything above the data. It reads this in, transposes the rows into columns, and saves it for the output.

Task data is the second block. The script reads all the lines and separates the data entries by semicolons. A new column is created based on the number shown, denoting if the number is equal to or higher than 5 (1 = yes, 0 = no).

Block	Trial_Number	Number Shown	Key Pressed	Correct	RT	Mirror	**LowHigh**
1	1	2	0	1	0.471	0	**0**
1	2	1	0	0	0.78	1	**0**
1	3	3	0	1	0.322	0	**0**

Based on the structure shown above, it is possible to calculate the accuracy and the response times separately for low and high, but also overall.

- Accuracy is the average of the “correct” column
- Response time is the average of the RT column
- Response time for correct only is the average of the RT column is Correct = 1

The script then creates a multivariate and univariate version of the results and returns this to the main script.