EPRIME CODE DOCUMENTATION

Generate .csv files from eprime .txt files

Run these Python codes in the given order –

1) batch_ce.py – Generates the .csv file from the .txt eprime file (saved in the main directory containing the log and originals directories-accepts data from originals/ePrimeData) input in a new data directory named eprime_csvfiles, which is created in the input folder directory.
    a) --path: Takes the full path directory to the .txt eprime files.[The **main directory** containing all the subjectIDs with subdirectories – log, originals.]
    b) --subjectID: Takes the subject ID name and generates the .txt file for the particular subject. [accepts sub- ID number/ID number, you can either be in the **main directory** and enter only the ID number or enter the full path to the subjectID]

Example:
batch_ce.py --subjectID=”80026b”
batch_ce.py --subjectID=”sub-80026b”
batch_ce.py –subjectID=”/projects……/MB6/sub-80026b or 80026b” [MB6 contains all the subject folders]
batch_ce.py --path=”/projects…/MB6”

Generate .tsv files from .csv files

2) csv2tsv.py – Generate the .tsv files form the .csv input file (the .csv directory is taken in as eprime_csvfiles – generated form batch_ce.py arrangement). The inputs to this code are:

a) --input_dir: The pathway to the input directory containing the .csv files. [It takes the **main directory** as input, which contains all the subject directories]

b)--output_dir: The pathway to the output directory where you would like to save the .tsv files [This is accordance to the **BIDS directory** arrangement, and takes the main directory containing all the BIDS converted subject directories as input here]

c)--subID: Takes the subjectID as input.

Example:

csv2tsv.py –-input_dir=”/projects/……/MB6” --output_dir = “/projects/…../rawData” [MB6 consiss of all the subject folders, rawData consists of the BIDS converted datasets, where the code searches for the func directory to store the .tsv files]
csv2tsv.py –-input_dir="/projects/…../MB6" --output_dir = "/projects/..../rawData" --subID=”sub-80026b”
csv2tsv.py –-subID=”/projects/……/MB6/sub-80026b” --output_dir=”/projects/…/rawData”
csv2tsv.py --subID=”sub-80026b” --output_dir=”/projects/…./rawData” [Make sure your current directory contains the subID folder]

Generate the feat onset vectors (3 column format) and subject accuracy summary file list in the directory containing the rawData folder.

3) feat_excel.py – Generate feat onset vectors in .txt file.

a)--input_dir: Contains the .csv file input directory. [The **main directory** containing the subject lists which have log and originals folders.]

b)--output_dir: Contains the output directory in accordance with the **BIDS structure**.[The main rawData folder]

c)--subjectID: Enter the subjectID

examples:

feat_excel.py –-input_dir=”/projects/…….../MB6” --output_dir=”/projects/…./rawData”
feat_excel.py –-input_dir=”/projects/…./MB6” --subjectID=”sub-80026b” --output_dir=”/projects/…../rawData”
feat_excel.py –-subjectID=”/projects/…../MB6/sub-80026b” --output_dir=”/projects/……rawData”
feat_excel.py –-subjectID=”sub-80026b” --output_dir=”/projects/….rawData” [Make sure your current diretory contains the subjectID folder which contains the log and originals folders.]