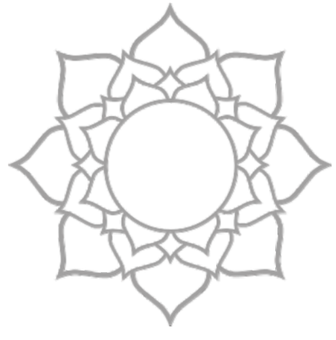


LOTUS Toolkit

Longitudinal Timeseries Unification & Signal Processing Toolkit

Author
Dr Jack Fogarty
Research Scientist, National Institute of Education, Singapore
jack.fogarty@nie.edu.sg

LOTUS Toolkit

The **LOTUS Toolkit** is a set of GUI's featuring a data *reader* and data *analyser* that were designed to facilitate organisation, management, processing, and analysis of data recorded using the EmbracePlus, or other devices that capture longitudinal recordings of continuous physiological signals.

1. LOTUS *Reader*
2. LOTUS *Analyser*

LOTUS READER

The **LOTUS Reader** is a GUI that enables the selective reconstitution of fragmented timeseries data by collating relevant ‘chunks’ of signal data into a continuous timeseries.

This is critical for processing continuous raw data from the EmbracePlus and other devices that output data in arbitrary *chunks* (discrete files) to improve memory processing and efficiency of data management. That is, for most standard signal processing, users will first need to ensure that relevant periods of raw signal data are restored as a continuous timeseries (i.e., without discontinuities). Unfortunately, this process can be challenging, particularly when handling large and irregular datasets, which is often the case when dealing with wearable timeseries data across numerous individuals.

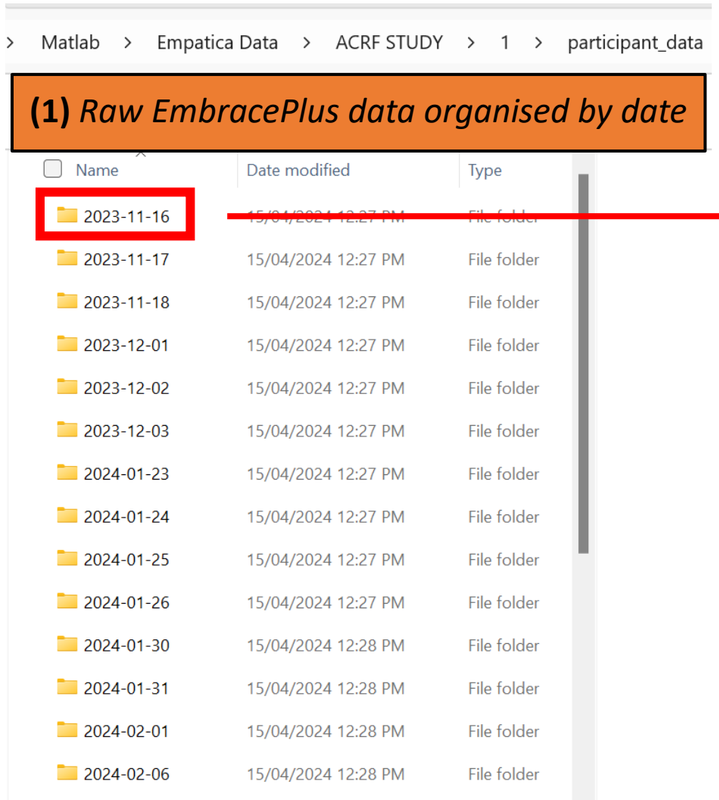
Handling fragmented EmbracePlus data

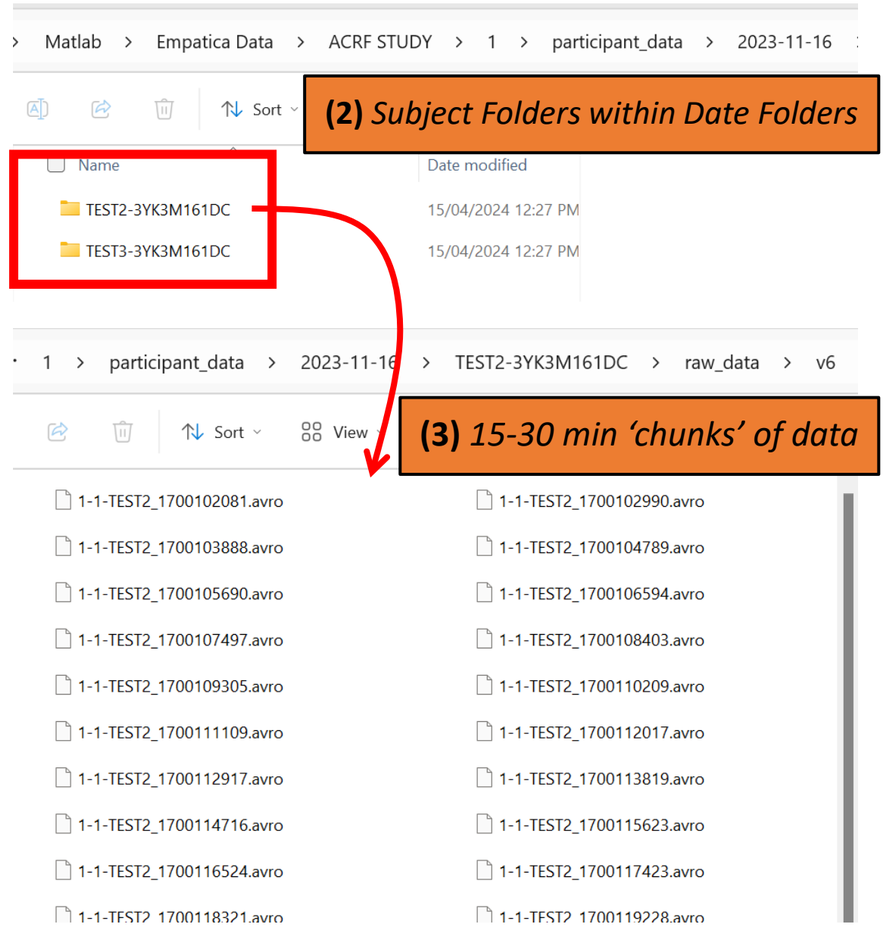
The challenge is how to selectively and systematically extract and compile relevant periods of data from irregular and fragmented datasets - a programmatic solution is required that can allow users to have interactive control over this procedure to output relevant data efficiently.

EmbracePlus data is organised by Site, date, Participant, and device serial number…

Note:

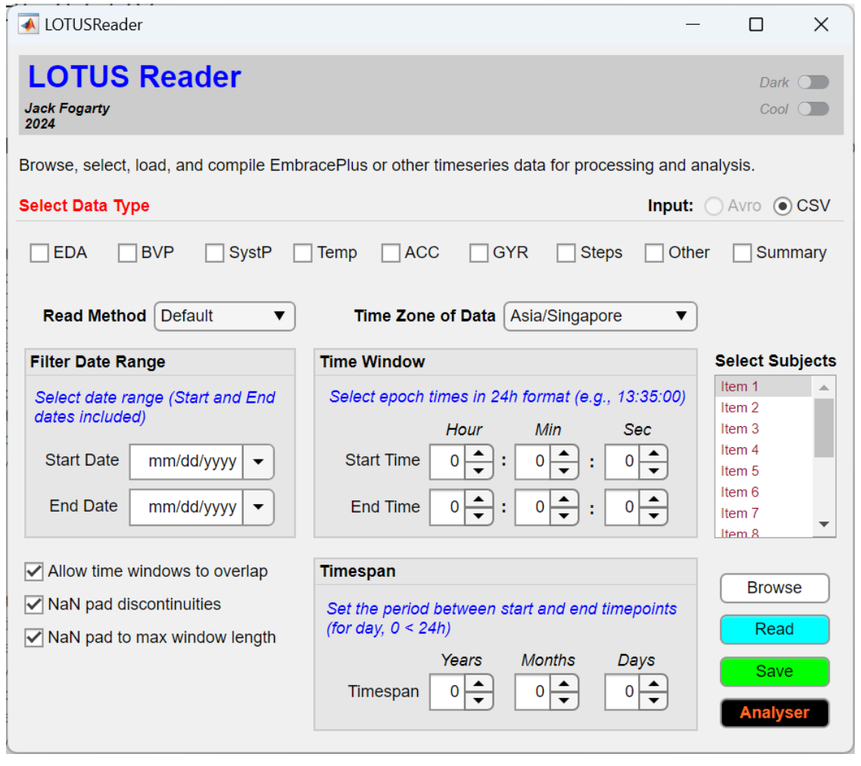
Avro files are a special type of text file containing raw signal data… these can be converted to CSV files for easier preprocessing or manual interaction (e.g. in excel)



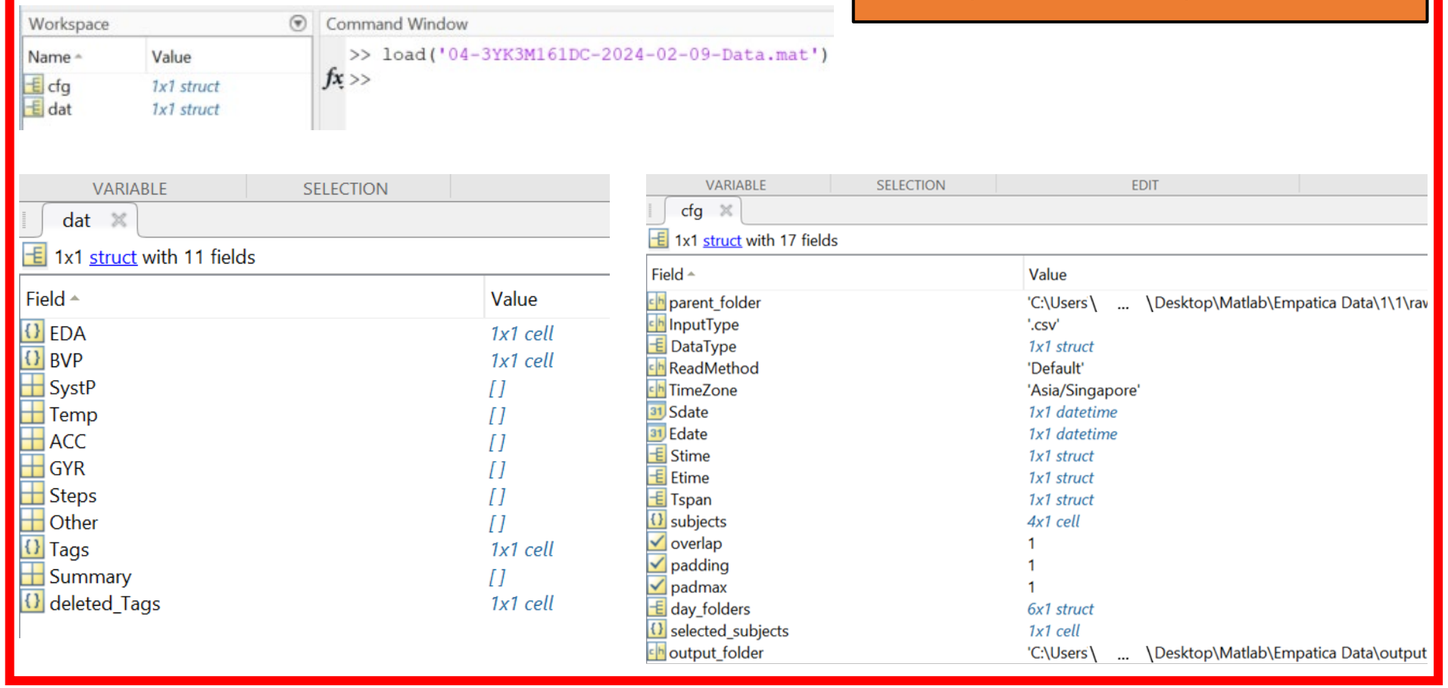


LOTUS READER (solution)

- The LOTUS Reader is a GUI built in Matlab 2023b with options to interactively filter, read, and compile data.
- For simplicity, the options for filtering, reading, and compiling data are all organised in a single window.
- **Output:** A mat file for each participant featuring two variables;
    1. **dat** - A mat structure containing reconstituted raw data and events (tags) over a selected period of time
    2. **cfg** - A mat structure containing file configurations (processing parameters and file history)



Example of mat output structures



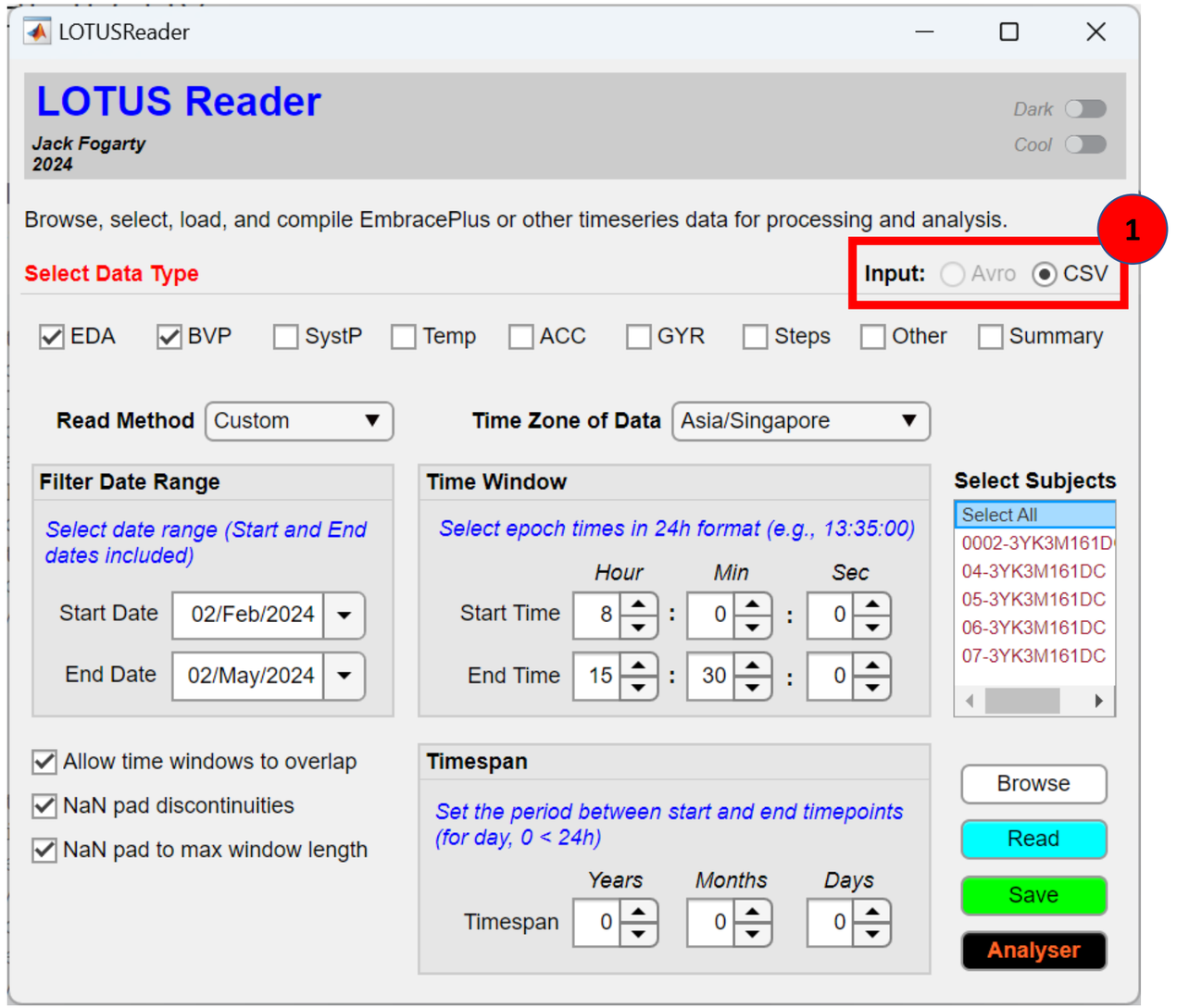
LOTUS READER

Core Procedure

1. Select input file type
2. Select required output
3. Choose read method
   a) Default
   b) Custom
   c) Event-based
4. Time zone of data
5. Browse
6. Read
7. Select Subjects
8. Save

Optional Filters

9. Filter dates
10. Set time window (Custom)
11. Set timespan (Custom)



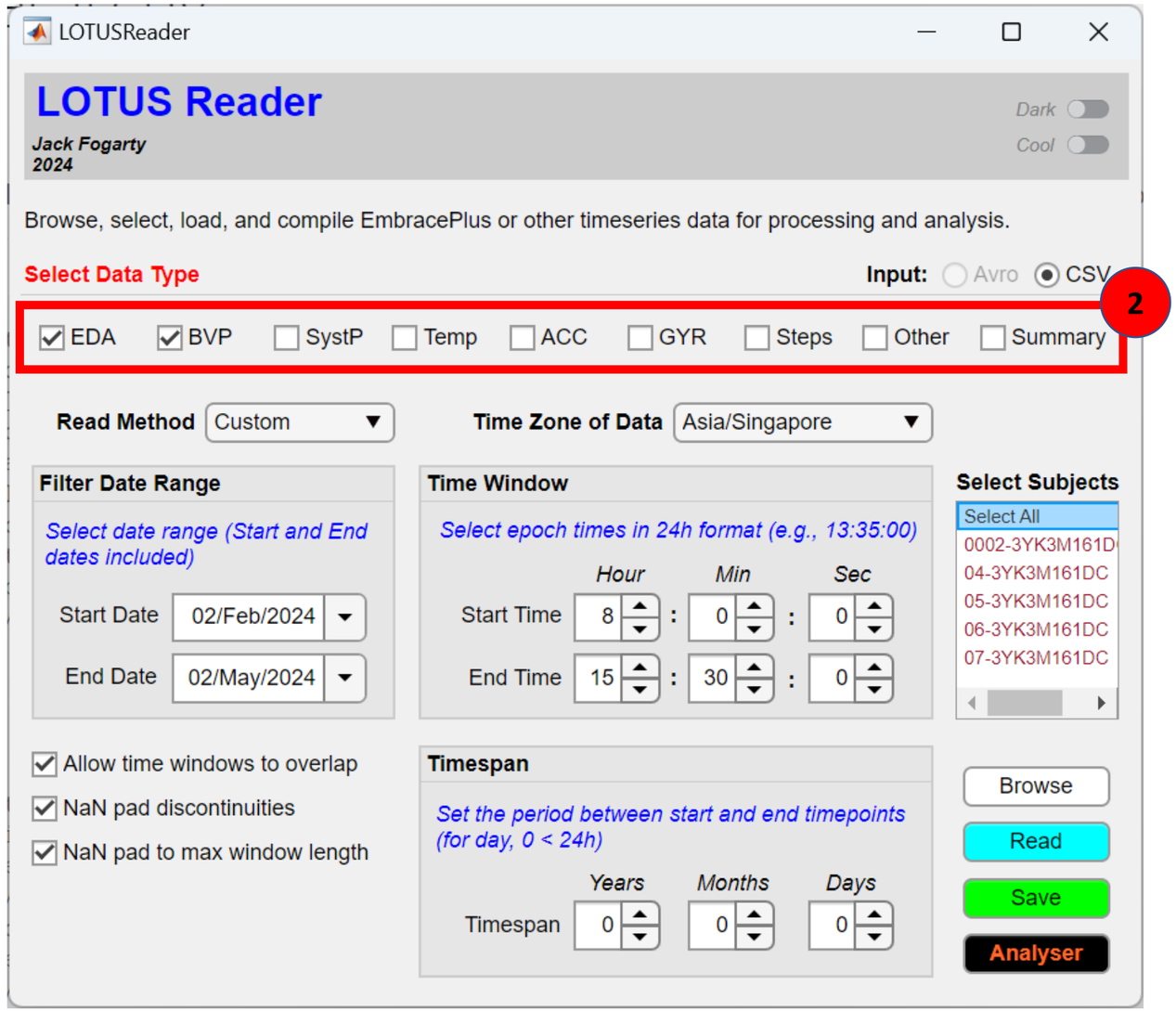
LOTUS READER

Core Procedure

1. Select input file type
2. Select required output
3. Choose read method
   a) Default
   b) Custom
   c) Event-based
4. Time zone of data
5. Browse
6. Read
7. Select Subjects
8. Save

Optional Filters

9. Filter dates
10. Set time window (Custom)
11. Set timespan (Custom)



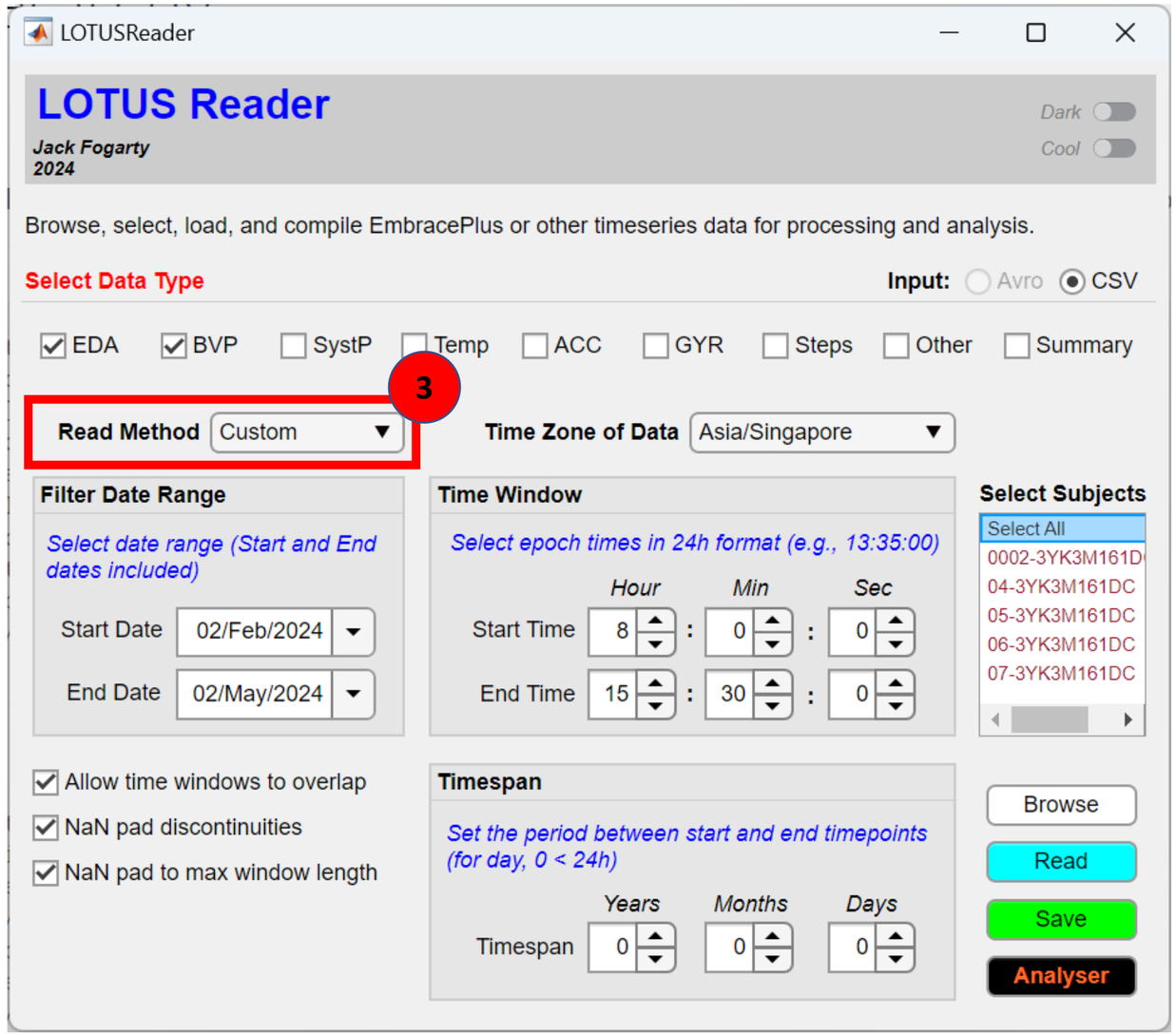
LOTUS READER

Core Procedure

1. Select input file type
2. Select required output
3. Choose read method
   a) Default
   b) Custom
   c) *Event-based*
4. Time zone of data
5. Browse
6. Read
7. Select Subjects
8. Save

Optional Filters

9. Filter dates
10. Set time window (Custom)
11. Set timespan (Custom)



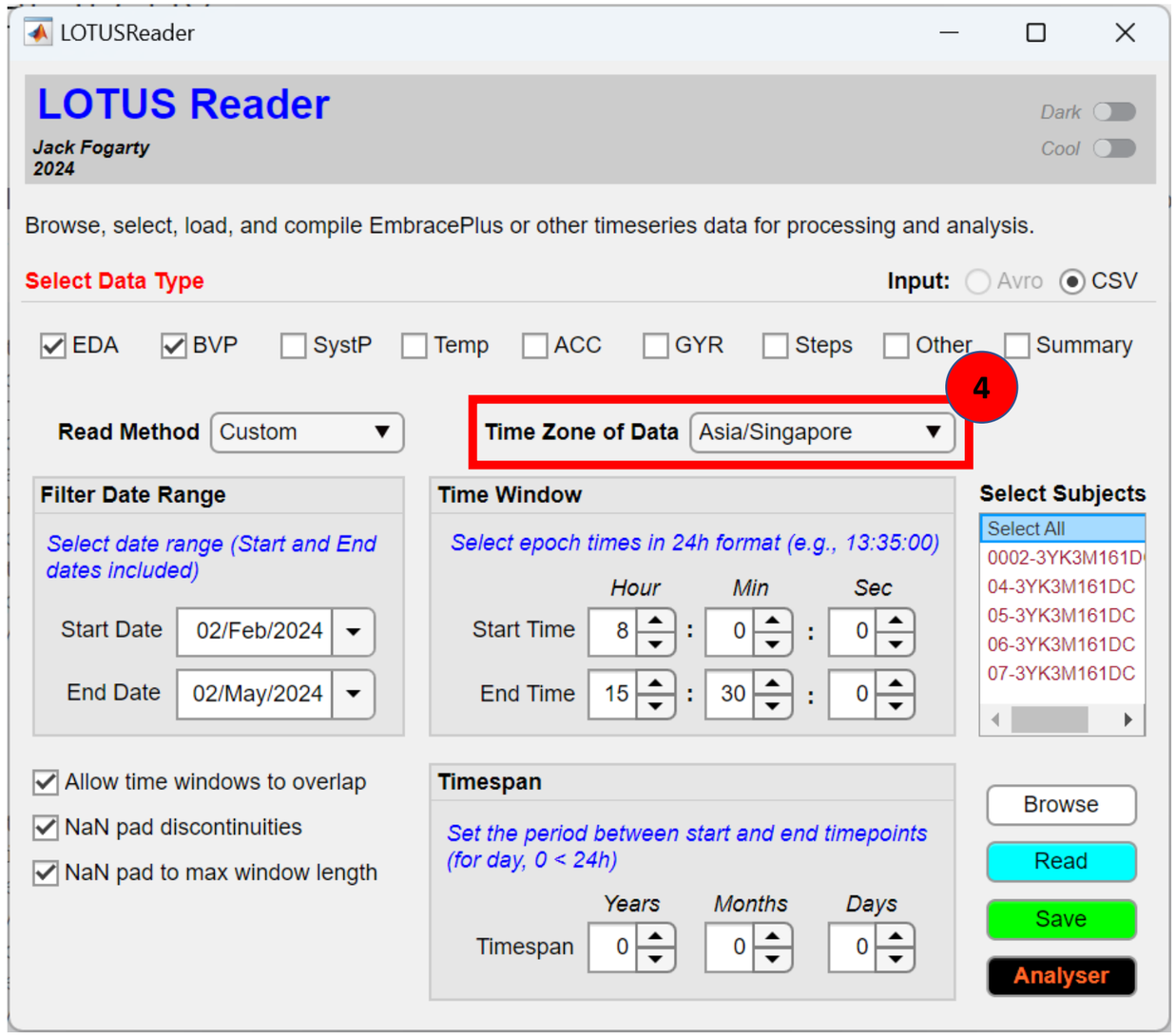
LOTUS READER

Core Procedure

1. Select input file type
2. Select required output
3. Choose read method
   a) Default
   b) Custom
   c) Event-based
4. Time zone of data
5. Browse
6. Read
7. Select Subjects
8. Save

Optional Filters

9. Filter dates
10. Set time window (Custom)
11. Set timespan (Custom)



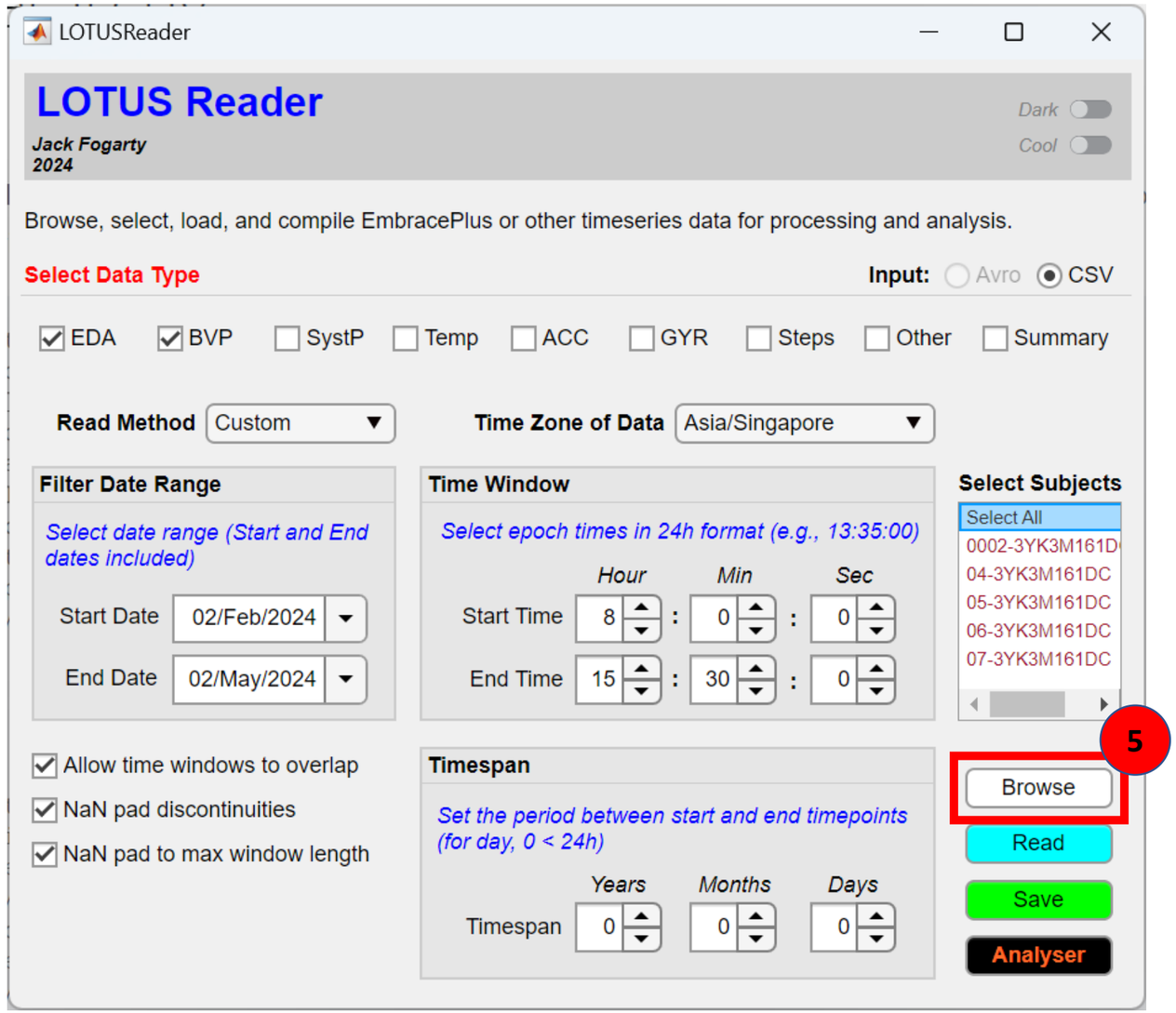
LOTUS READER

Core Procedure

1. Select input file type
2. Select required output
3. Choose read method
   a) Default
   b) Custom
   c) Event-based
4. Time zone of data
5. Browse
6. Read
7. Select Subjects
8. Save

Optional Filters

9. Filter dates
10. Set time window (Custom)
11. Set timespan (Custom)



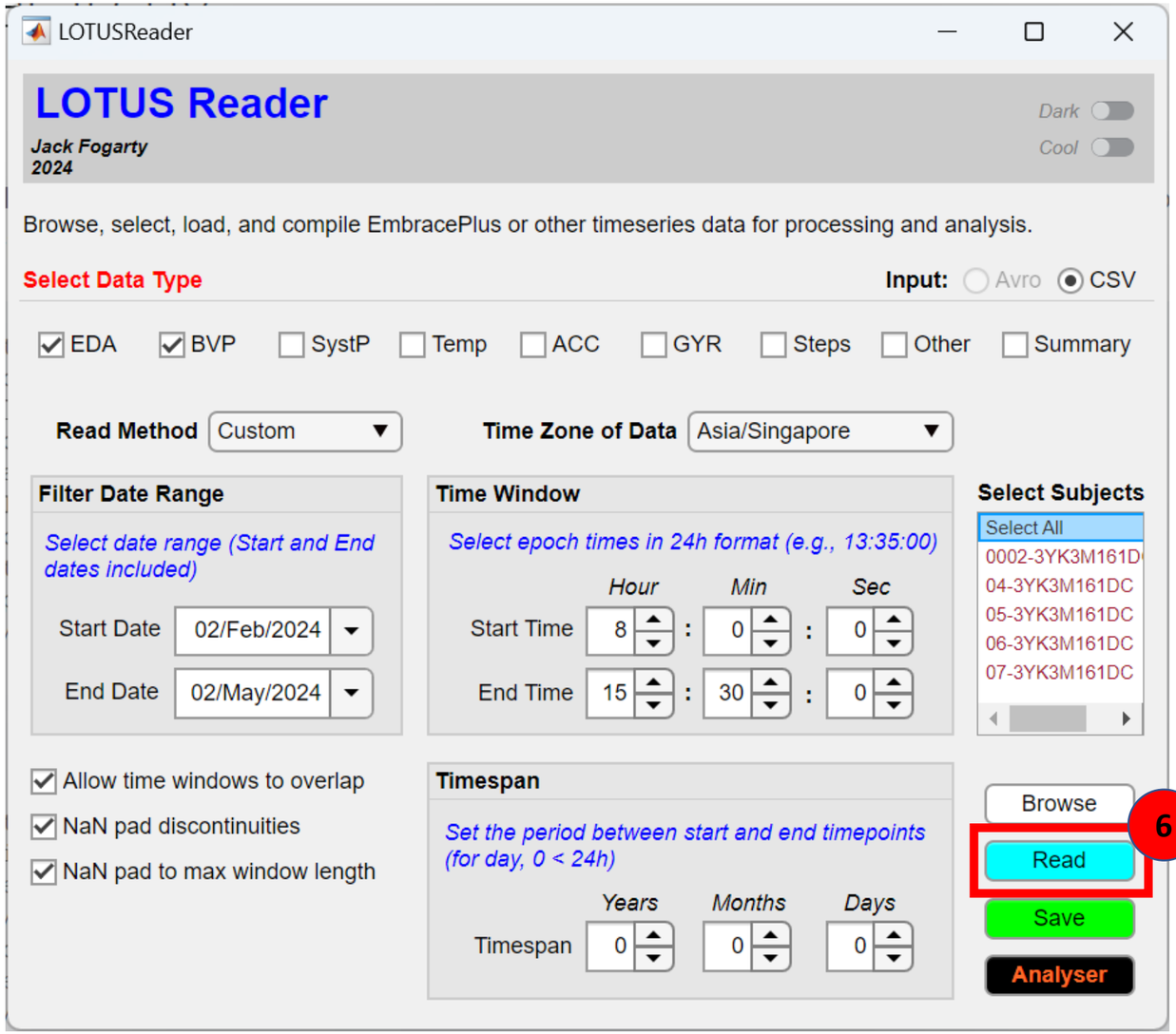
LOTUS READER

Core Procedure

1. Select input file type
2. Select required output
3. Choose read method
   a) Default
   b) Custom
   c) Event-based
4. Time zone of data
5. Browse
6. Read
7. Select Subjects
8. Save

Optional Filters

9. Filter dates
10. Set time window (Custom)
11. Set timespan (Custom)



NOTE

'Read' identifies potential subjects and updates the GUI configuration. Press this before selecting subjects and clicking 'Save' to ensure settings are up-to-date.

If GUI settings are altered on the fly, press 'Read' again to ensure GUI is up-to-date.

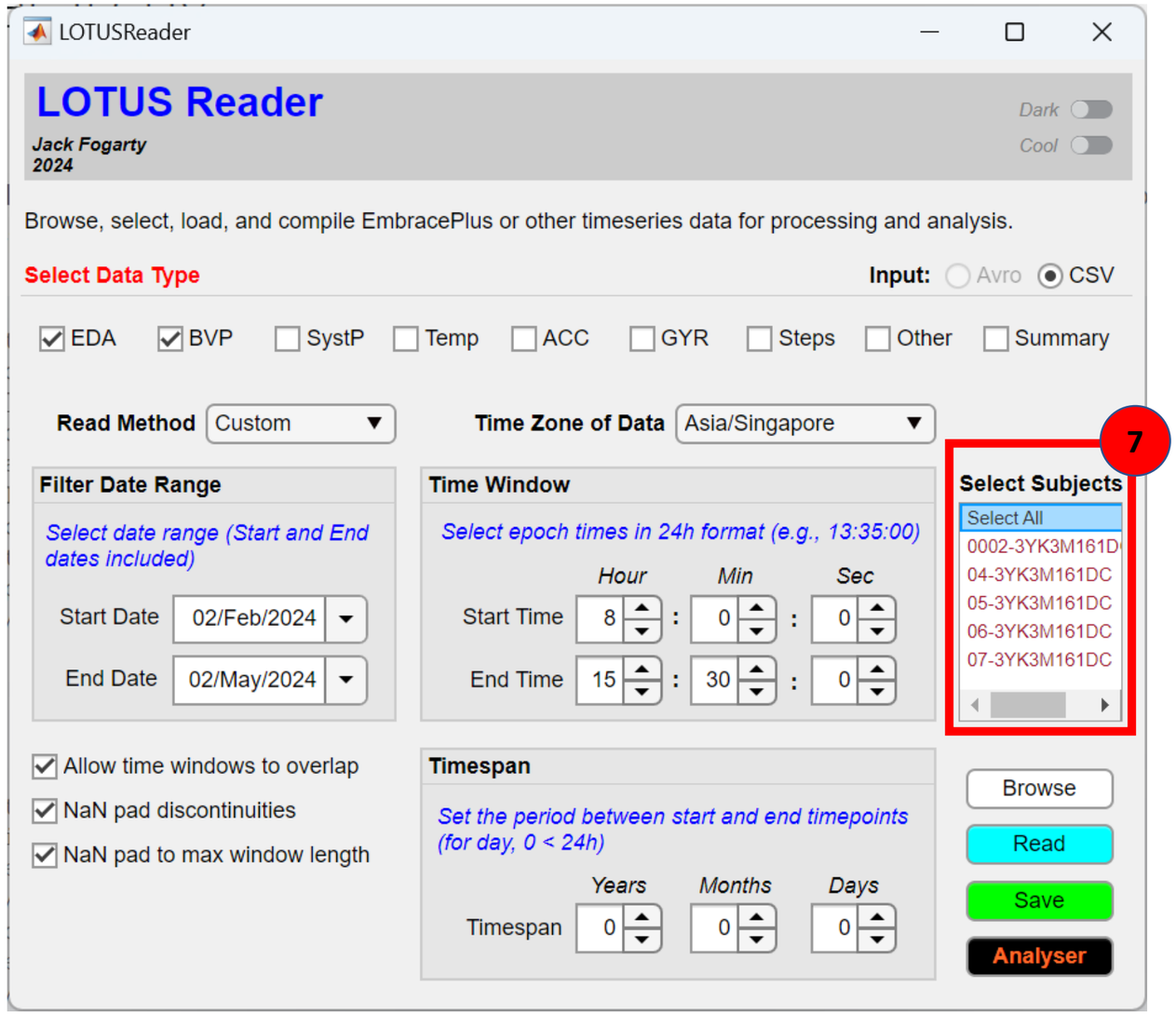
LOTUS READER

Core Procedure

1. Select input file type
2. Select required output
3. Choose read method
   a) Default
   b) Custom
   c) Event-based
4. Time zone of data
5. Browse
6. Read
7. Select Subjects
8. Save

Optional Filters

9. Filter dates
10. Set time window (Custom)
11. Set timespan (Custom)



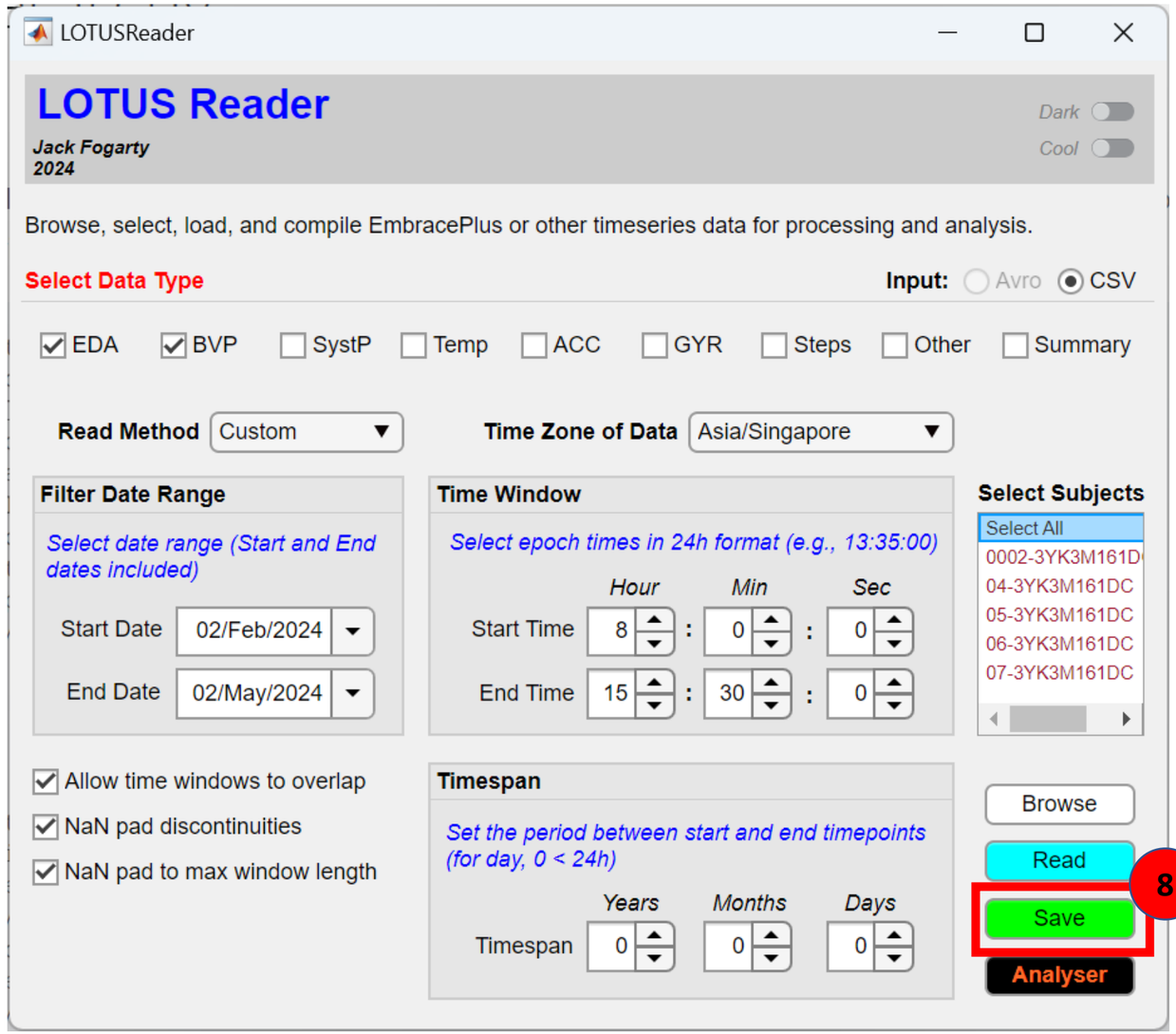
LOTUS READER

Core Procedure

1. Select input file type
2. Select required output
3. Choose read method
   a) Default
   b) Custom
   c) Event-based
4. Time zone of data
5. Browse
6. Read
7. Select Subjects
8. Save

Optional Filters

9. Filter dates
10. Set time window (Custom)
11. Set timespan (Custom)



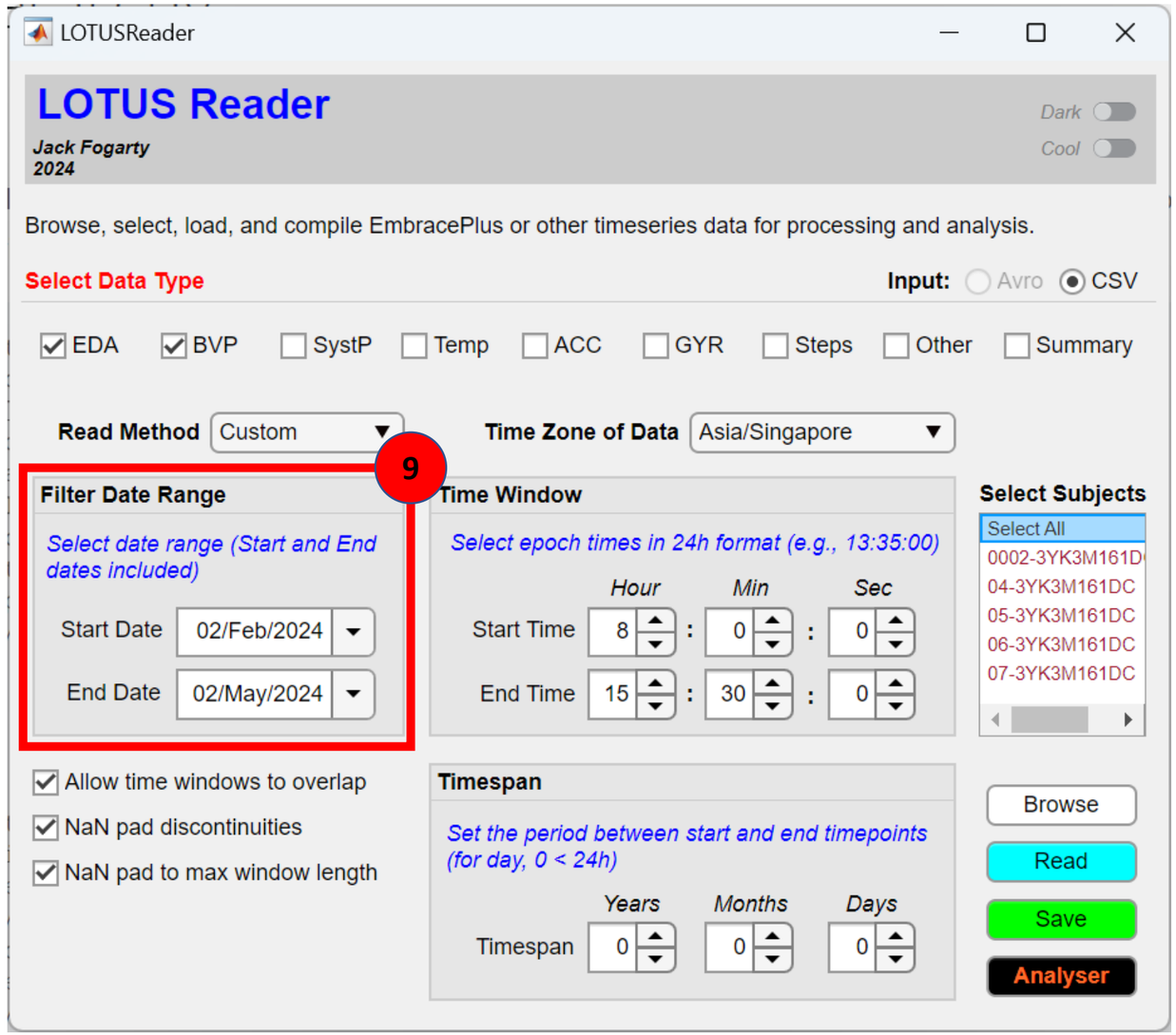
LOTUS READER

Core Procedure

1. Select input file type
2. Select required output
3. Choose read method
   a) Default
   b) Custom
   c) Event-based
4. Time zone of data
5. Browse
6. Read
7. Select Subjects
8. Save

Optional Filters

9. Filter dates
10. Set time window (Custom)
11. Set timespan (Custom)



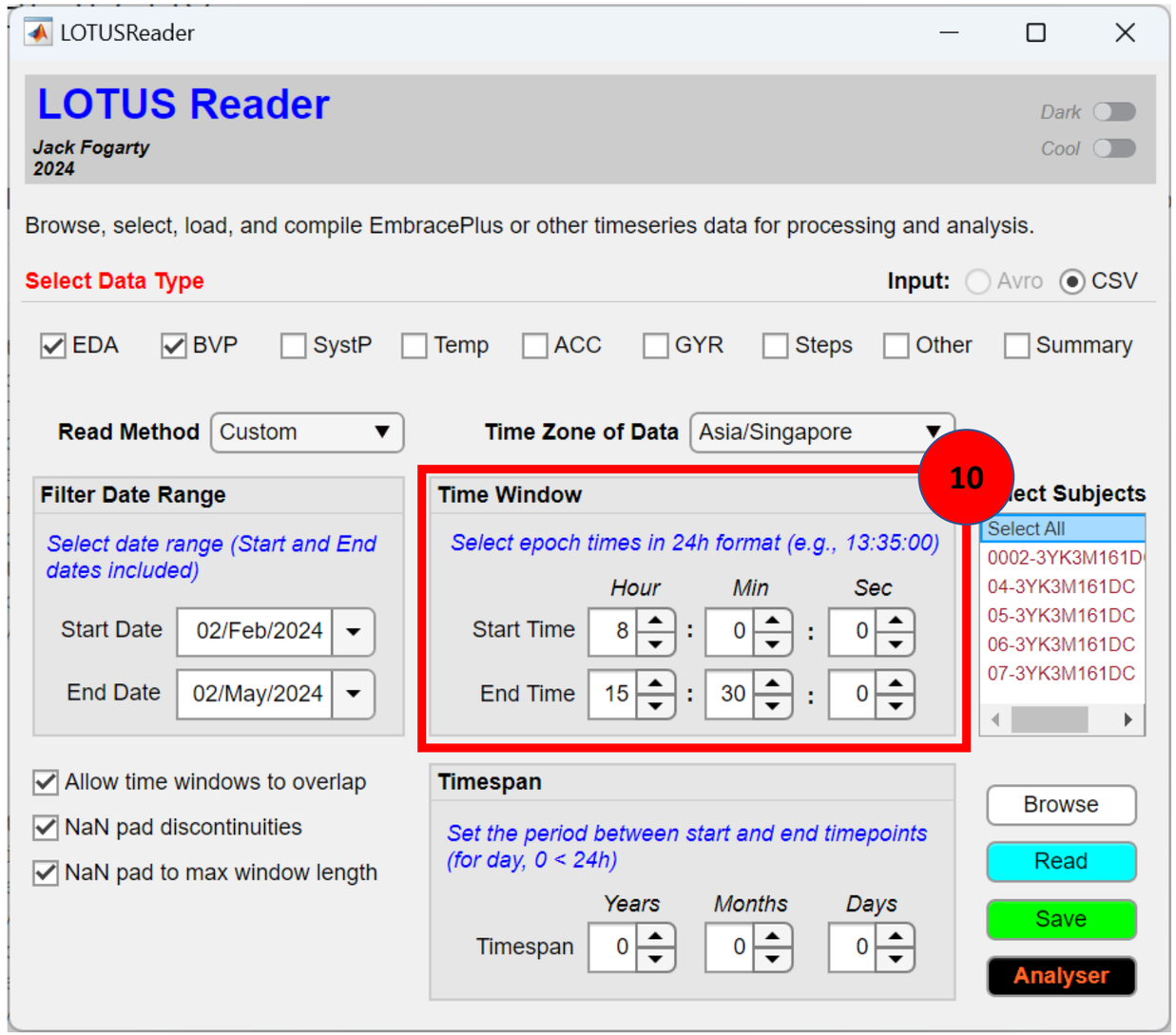
LOTUS READER

Core Procedure

1. Select input file type
2. Select required output
3. Choose read method
   a) Default
   b) Custom
   c) Event-based
4. Time zone of data
5. Browse
6. Read
7. Select Subjects
8. Save

Optional Filters

9. Filter dates
10. Set time window (Custom)
11. Set timespan (Custom)



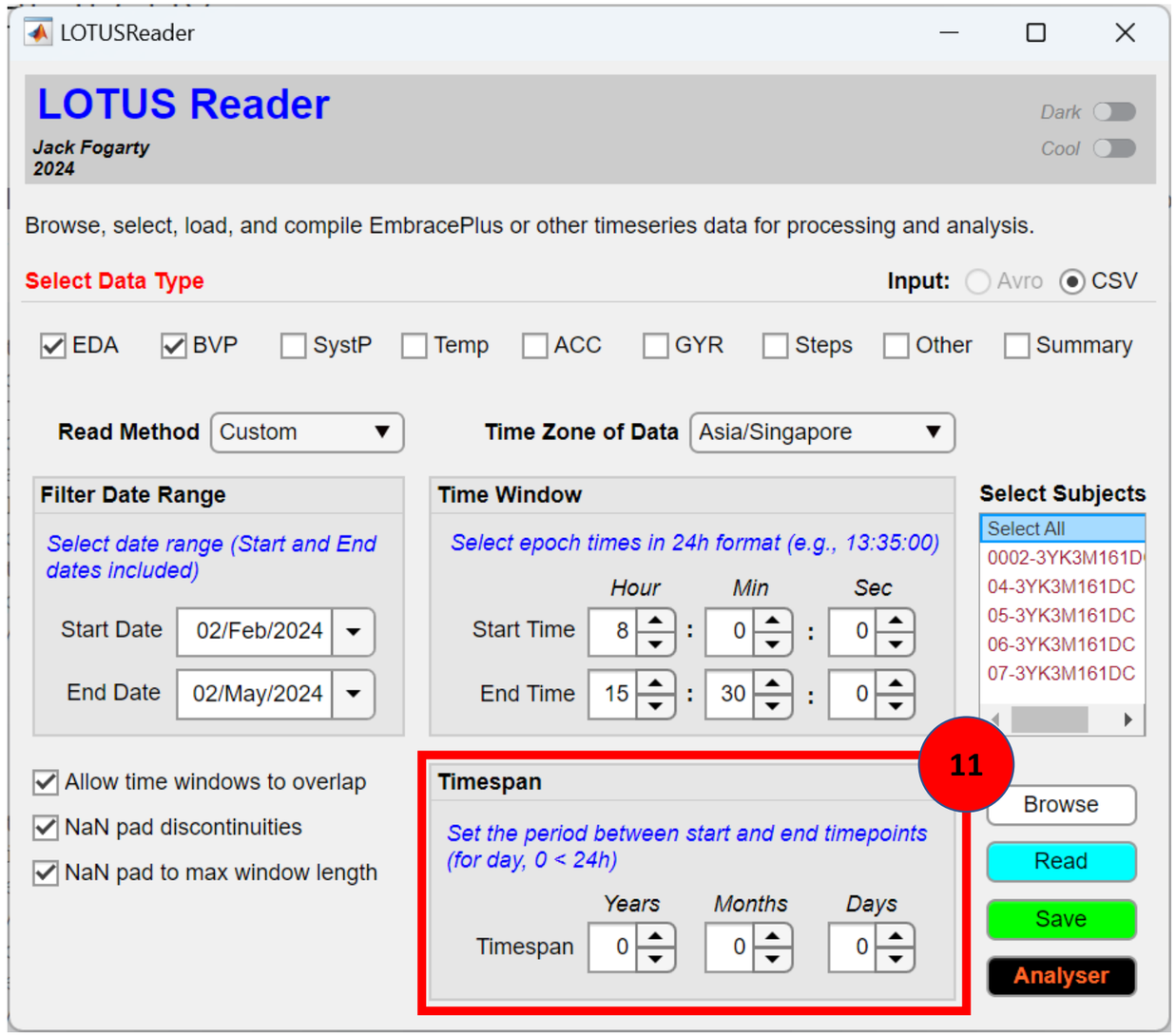
LOTUS READER

Core Procedure

1. Select input file type
2. Select required output
3. Choose read method
   a) Default
   b) Custom
   c) Event-based
4. Time zone of data
5. Browse
6. Read
7. Select Subjects
8. Save

Optional Filters

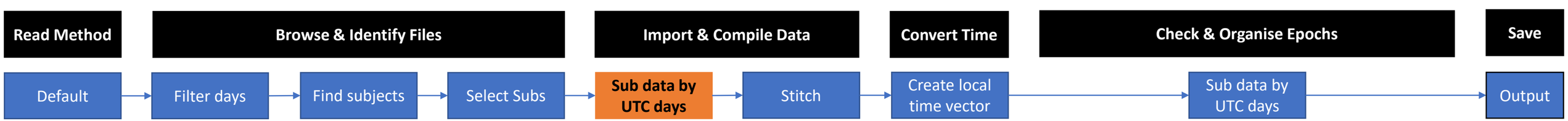
9. Filter dates
10. Set time window (Custom)
11. Set timespan (Custom)

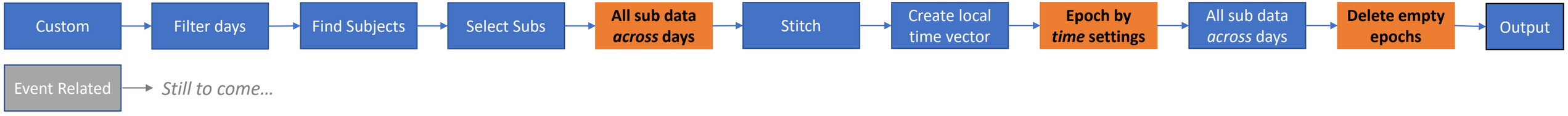


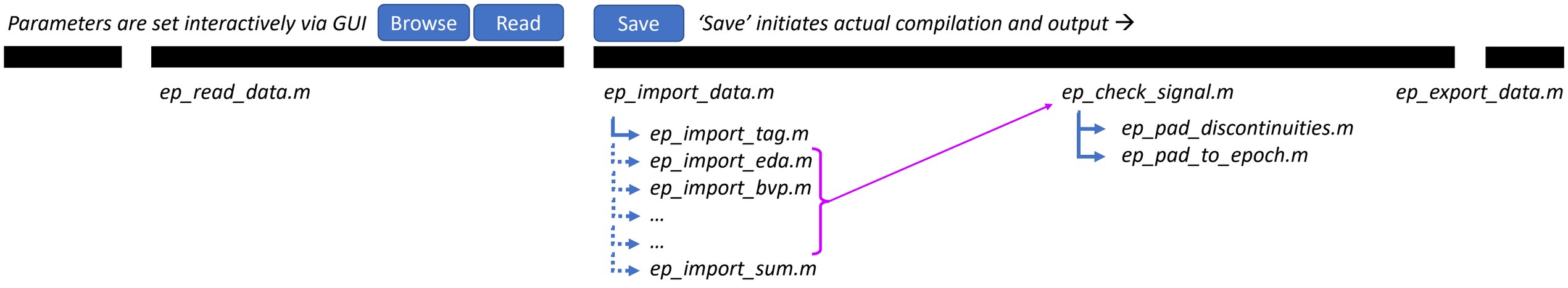
Basic Pipeline and Backend Code

The backend code (Matlab functions) behind the GUI are in early stages of development and this pipeline and underlying code will likely be subject to change as improvements are made.

This basic pipeline below is implemented after selecting the type of input (i.e., CSV) and desired output options…







LOTUS READER: Data Output

Data structure (dat)

Output EDA signal

VARIABLE | SELECTION

dat

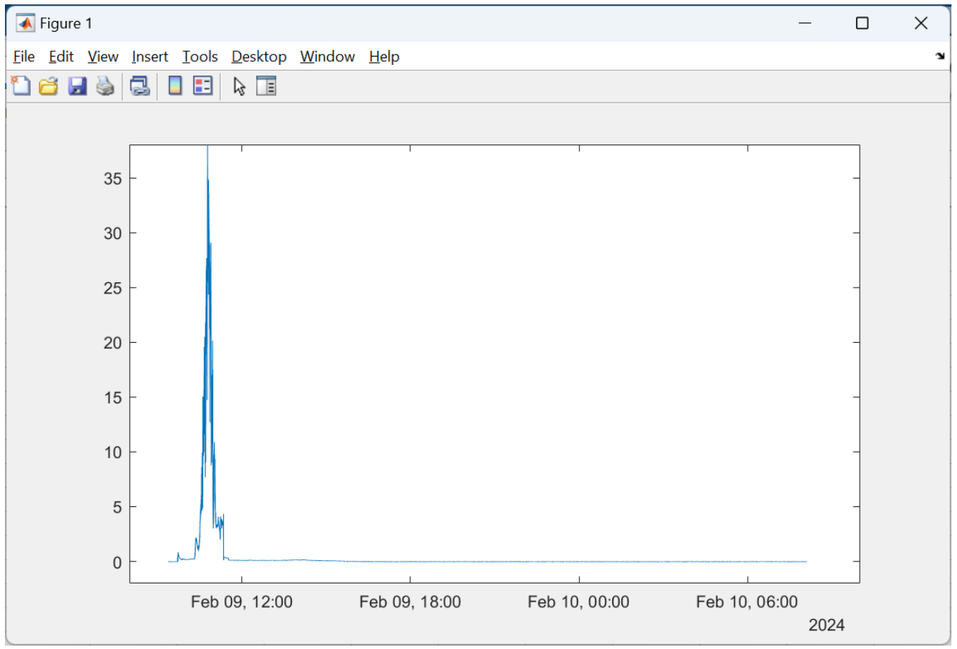
1x1 struct with 11 fields

Field	Value
EDA	1x1 cell
BVP	1x1 cell
SystP	[]
Temp	[]
ACC	[]
GYR	[]
Steps	[]
Other	[]
Tags	1x1 cell
Summary	[]
deleted_Tags	1x1 cell

dat | dat.EDA | dat.EDA{1, 1}

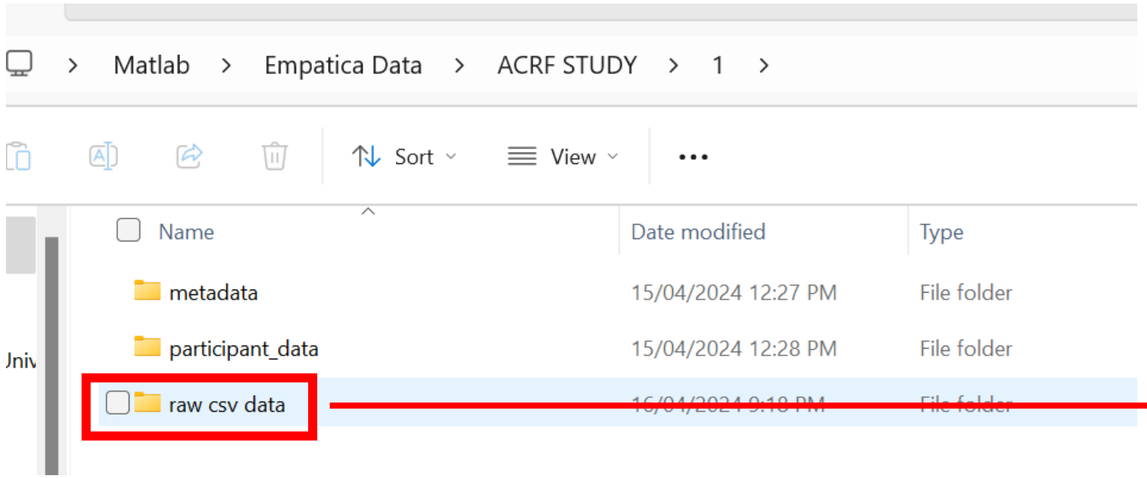
dat.EDA{1, 1}

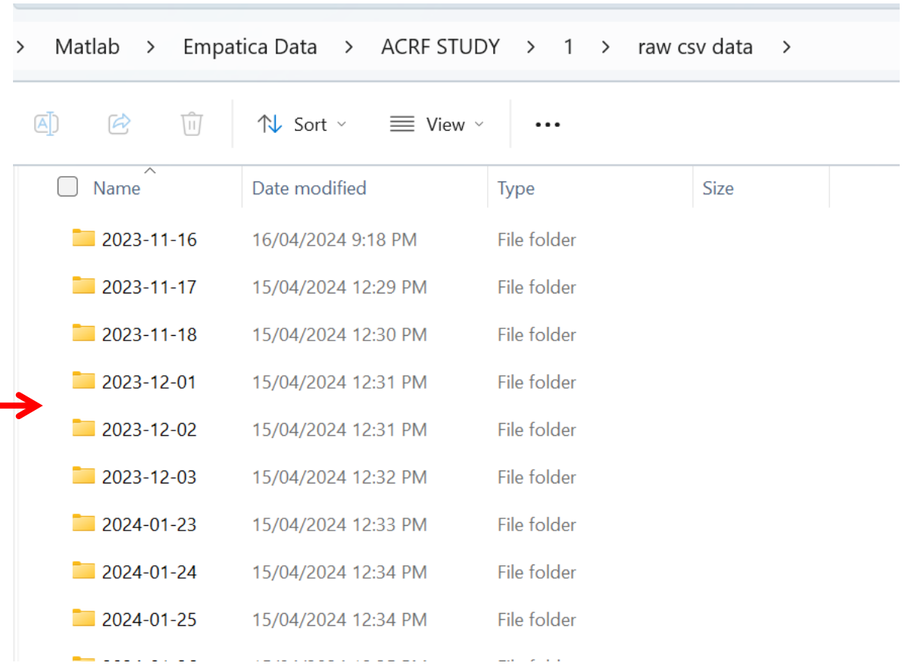
	1 unix_time	2 local_time	3 data
77677	1.7075e+15	02/09/24 13:23:3...	0.1020
77678	1.7075e+15	02/09/24 13:23:3...	0.1029
77679	1.7075e+15	02/09/24 13:23:3...	0.1030
77680	1.7075e+15	02/09/24 13:23:3...	0.1030
77681	1.7075e+15	02/09/24 13:23:3...	0.1034
77682	1.7075e+15	02/09/24 13:23:4...	0.1031
77683	1.7075e+15	02/09/24 13:23:4...	0.1043
77684	1.7075e+15	02/09/24 13:23:4...	0.1028
77685	1.7075e+15	02/09/24 13:23:4...	0.1028
77686	1.7075e+15	02/09/24 13:23:4...	0.1037
77687	1.7075e+15	02/09/24 13:23:4...	0.1047
77688	1.7075e+15	02/09/24 13:23:4...	0.1031
77689	1.7075e+15	02/09/24 13:23:4...	0.1022
77690	1.7075e+15	02/09/24 13:23:4...	0.1032
77691	1.7075e+15	02/09/24 13:23:4...	0.1046
77692	1.7075e+15	02/09/24 13:23:4	0.1036



Notes on Input Data

- LOTUS Reader can read in avro or CSV data converted from avro files.
  - Conversion from avro to csv is can be done using python code provided in Empatica manuals (see their documentation)
  - Python code has also been customized to directly read avro into matlab (more memory efficient but potentially slower for LOTUS)
- Users should *browse* to the parent folder containing the raw CSV data grouped by date.
  - This raw csv folder structure is a product of a custom version of the Empatica python script for converting avro to csv
  - To read avro directly, browse to the 'participant_data' folder instead





Data Output

Data structure (dat)

VARIABLE | SELECTION

dat

1x1 struct with 11 fields

Field	Value
EDA	*1x1 cell*
BVP	*1x1 cell*
SystP	*[ ]*
Temp	*[ ]*
ACC	*[ ]*
GYR	*[ ]*
Steps	*[ ]*
Other	*[ ]*
Tags	*1x1 cell*
Summary	*[ ]*
deleted_Tags	*1x1 cell*

Output events (tags)

dat | dat.Tags | dat.Tags{1, 1}

dat.Tags{1, 1}

	1 Tag	2 UTC_FileTime	3 TZ_FileTime	4 TZ_TagTime	5 Annotation
1	1	'1707442807'	02/09/24	09:44:57.876	''
2	2	'1707443710'	02/09/24	09:59:08.055	''
3	3	'1707443710'	02/09/24	10:09:42.733	''
4	4	'1707444607'	02/09/24	10:12:06.119	''
5	5	'1707444607'	02/09/24	10:18:30.321	''
6	6	'1707444607'	02/09/24	10:24:53.189	''
7	7	'1707446412'	02/09/24	10:50:52.988	''
8	8	'1707447319'	02/09/24	10:57:05.333	''
9	9	'1707447319'	02/09/24	10:57:37.392	''
10	10	'1707447319'	02/09/24	11:03:15.614	''
11	11	'1707448221'	02/09/24	11:13:41.233	''
12	12	'1707448221'	02/09/24	11:14:26.468	''
13	13	'1707448221'	02/09/24	11:20:44.445	''

Cfg Output

Configuration structure (cfg)

VARIABLE | SELECTION | EDIT

cfg

1x1 struct with 17 fields

Field	Value
parent_folder	'C:\Users\jfogarty\Desktop\Matlab\Empatica Data\1\1\rav
InputType	'.csv'
DataType	*1x1 struct*
ReadMethod	'Default'
TimeZone	'Asia/Singapore'
Sdate	*1x1 datetime*
Edate	*1x1 datetime*
Stime	*1x1 struct*
Etime	*1x1 struct*
Tspan	*1x1 struct*
subjects	*4x1 cell*
overlap	1
padding	1
padmax	1
day_folders	*6x1 struct*
selected_subjects	*1x1 cell*
output_folder	'C:\Users\jfogarty\Desktop\Matlab\Empatica Data\output

Contains details regarding the file/dataset that will be kept for posterity or used in subsequent analysis steps within the LOTUS *analyzer*

Use Requirements

LOTUS Reader was built in Matlab 2023b and utilises python code via Matlab, using python version 3.11

Users without Matlab can install LOTUS Reader via a compiled version but will still need a compatible version of python (i.e. 3.9, 3.10, or 3.11) installed to read avro data directly into organised mat structures.

Dependencies to read avro via python also need to be installed for:

- Avro (run: pip install avro in a command prompt or other terminal)

Reading in Avro Data

Option 1

- Run the adjusted Empatica Python script provided to transform to Avro to CSV
- Import CSV data via LOTUS

Requirements

- Python installed
- Use ‘external’ python script [extractavro_matlab.py]
    - Dependencies (libraries) to install:
        - Avro (run: pip install avro in a command prompt or other terminal)
        - json (comes with python)
        - csv (comes with python)
        - os (comes with python)
- Ability to run python script in external environment (e.g., via Jupyter Notebook)

Notes

- Reading in CSV is currently a lot faster than reading in Avro directly to LOTUS via Option 2 (next slide)

Reading in Avro Data

Option 2

- Call avro data into Matlab directly via LOTUS Reader

Requirements

- Python version 3.11 installed
- Be using Matlab 2023 or later (as this can utilise Python version 3.11)
    - Dependencies (libraries) to install:
        - Avro (run: pip install avro in a command prompt or other terminal)
        - json (comes with python)
        - csv (comes with python)
        - os (comes with python)

Notes

- Reading in Avro files is slower (but more memory efficient) as Matlab is calling python [ep_read_avro.py] to read avro data directly into mat files (i.e., no CSV files necessary)
- Need to ensure compatible versions of Matlab and python are installed
- LOTUS was built with Matlab 2023 [which is 'installed' in the LOTUS_reader.exe] and this is compatible with python versions 3.9, 3.10, and 3.11
- Python 3.11 was installed during the development of LOTUS

On the Horizon for Future Releases…

- LOTUS Analyzer is currently in development and to be released soon to enable signal preprocessing

LOTUS Reader

- Backwards compatibility with E4 data to come
- Modular output formats (i.e., mat or CSV)
- Refined ‘summary’ of input datasets
- Read functions for GYR, steps, and other signals from alternate devices may be built in
- Update for status and warning messages