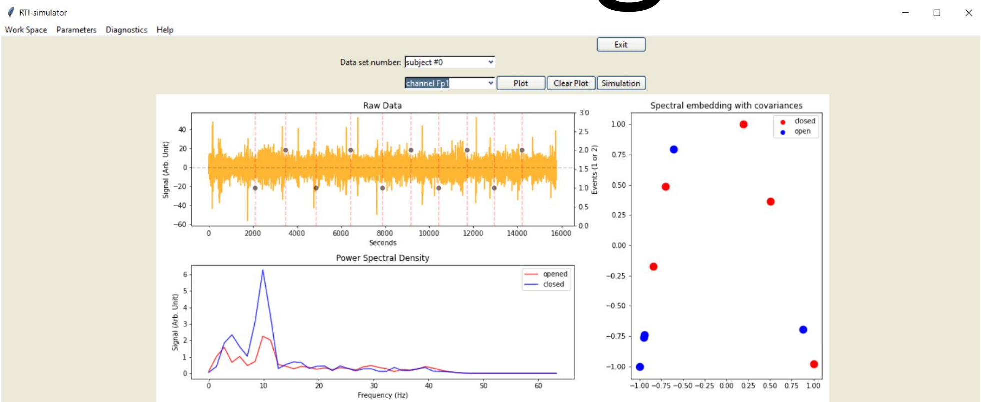
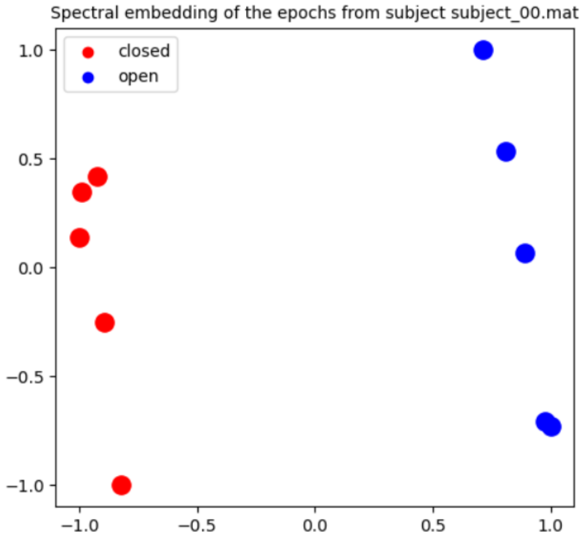
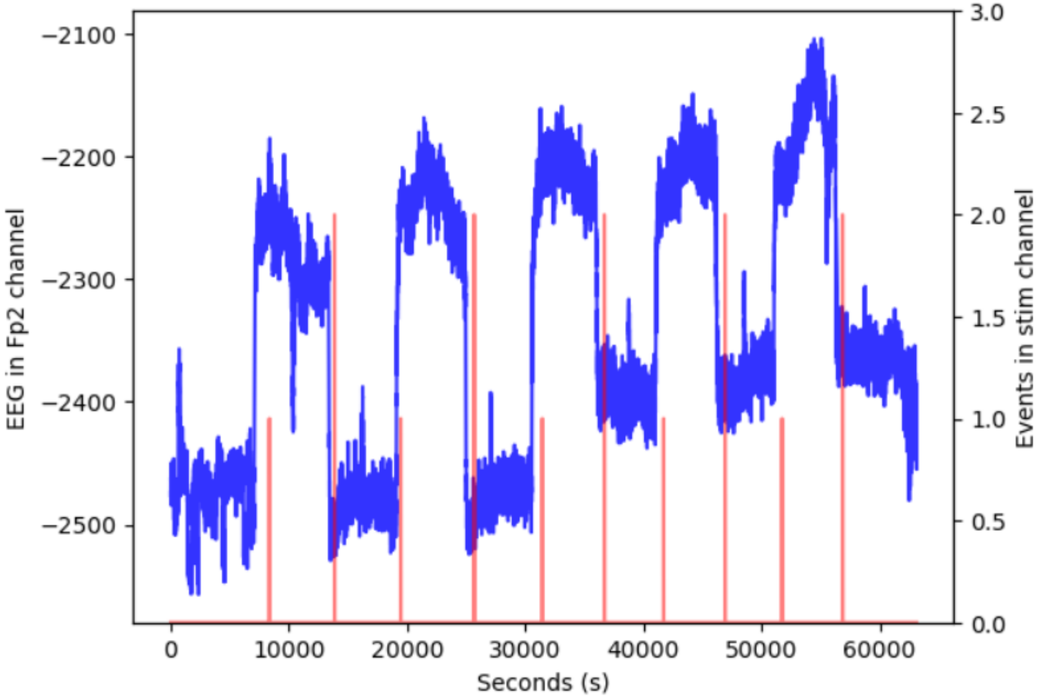
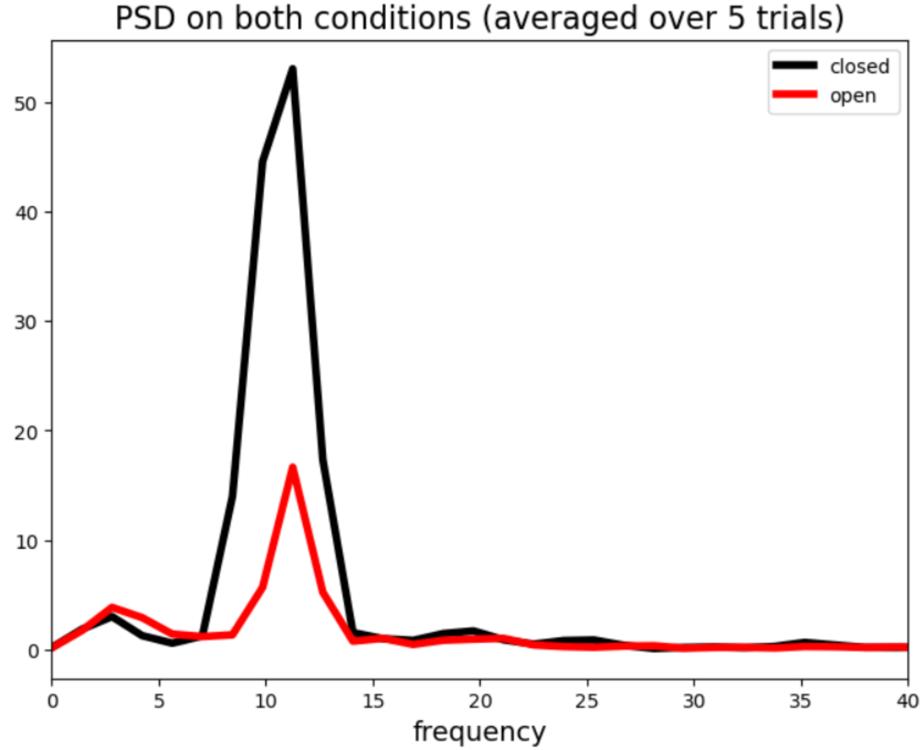
The Detection of Driver's Fatigue Level



Kiman Park





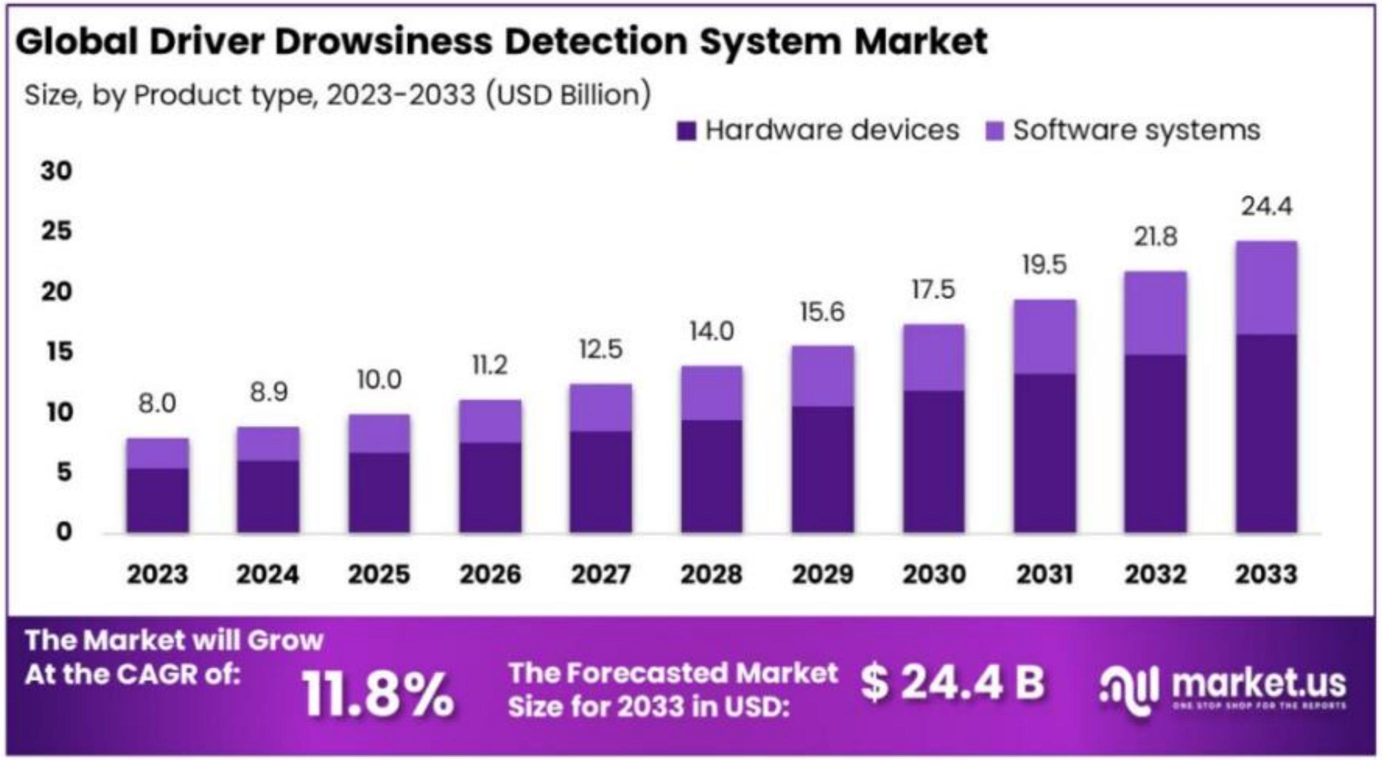


Market Value

Driver Fatigue Monitoring System Market Size and Projections

The **Driver Fatigue Monitoring System Market** Size was valued at USD 4.39 Billion in 2023 and is expected to reach **USD 16.52 Billion by 2031**, growing at a **18% CAGR from 2024 to 2031.**The positive momentum in market dynamics, coupled with the expected prolonged expansion, points to robust growth rates throughout the forecasted period. In essence, the market is on the verge of noteworthy and substantial development.

4.39 Billior



on by 2031,

https://www.marketresearchintellect.com/product/global-driver-fatigue-monitoring-system-market?utm_source=Artrocker&utm_medium=027/

https://market.us/report/driver-drowsiness-detection-system-market/

Market Value

Driver Fatigue Monitoring System Market Size And Analysis Research Report 2031

Data Wisdom Network
99 followers

Who is the largest Manufacturers of Driver Fatigue Monitoring System Market worldwide?

- Robert Bosch GmbH
- Continental AG
- Valeo
- Hella GmbH
- Autoliv Inc
- Denso Corporation
- Magna International Inc
- Aisin Seiki Co.
- Ltd
- TRW Automotive

https://www.linkedin.com/pulse/driver-fatigue-monitoring-system-market-size-analysis-6dbxc/

Market Value

Which regions are leading the Driver Fatigue Monitoring System Market?

- North America (United States, Canada and Mexico)
- Europe (Germany, UK, France, Italy, Russia and Turkey etc.)
- Asia-Pacific (China, Japan, Korea, India, Australia, Indonesia, Thailand, Philippines, Malaysia and Vietnam)
- South America (Brazil, Argentina, Columbia etc.)
- Middle East and Africa (Saudi Arabia, UAE, Egypt, Nigeria and South Africa)

What are the types of Driver Fatigue Monitor Equipment available in the Market?

- Face Monitoring
- Pupil Monitoring
- Other

https://www.linkedin.com/pulse/driver-fatigue-monitoring-system-market-size-analysis-6dbxc/

Market Value

What are the types of Driver Fatigue Monitoring System available in the Market?

Based on Product Types the Market is categorized into Below types that held the largest Driver Fatigue Monitoring System market share In 2022.

- Hardware Devices
- Software System

What are the factors driving the growth of the Driver Fatigue Monitoring System Market?

Growing demand for below applications around the world has had a direct impact on the growth of the Driver Fatigue Monitoring System

- Passenger Cars
- Commercial Vehicles

https://www.linkedin.com/pulse/driver-fatigue-monitor-equipment-market-size-emerging-13ctc/
https://www.linkedin.com/pulse/driver-fatigue-monitoring-system-market-size-analysis-6dbxc/

Market Value

How Big of a Problem is Driver Fatigue?

Multiple studies indicate fatigue as the principal cause of between 13% to 40% of trucking accidents. On the conservative end, the Federal Motor Carrier Safety Administration (FMCSA) estimates that fatigue is the cause of 13% of all trucking accidents, while the European Commission on Road Safety pegs the number at 25%. A study by the National Transportation Safety Board reports it as a cause of up to 40% of cases.

In more human terms, even by the most conservative estimations, in the United States alone each year there are approximately 100,000 fatigue-related motoring accidents per year, resulting in 71,000 injuries and 800 tragic, largely preventable deaths.

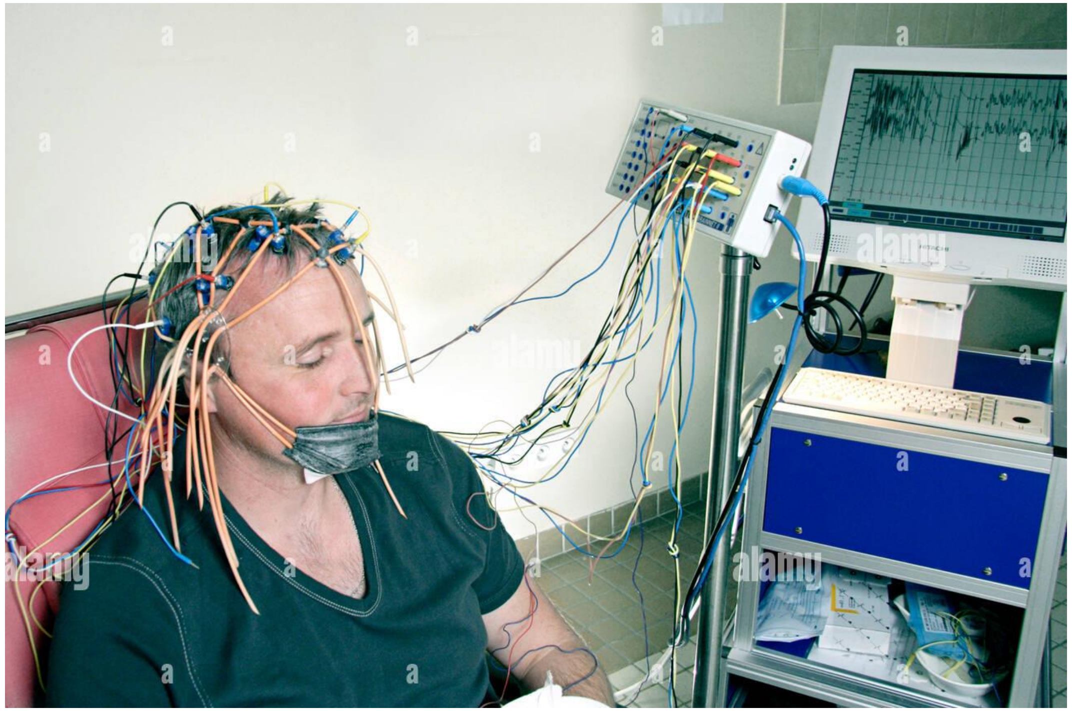
The Financial Cost of Trucker Fatigue

Beyond the human factor, cost is a major consequence of fatigue-related accidents. According to a study by the FMCSA, the total cost of trucking accidents involving driver fatigue is approximately $20 billion per year. This includes costs such as medical expenses, property damage, and lost productivity.

The National Transportation Safety Board (NTSB) indicates that the average cost per trucking accident involving driver fatigue is approximately $91,000. However, the cost of a fatigue-related accident can vary depending on the severity of the accident. For example, a minor accident with no injuries may cost less than $10,000, while a serious accident with multiple fatalities can cost multiple millions of dollars.

https://fatiguescience.com/blog/the-true-cost-of-driver-fatigue-for-your-trucking-fleet/

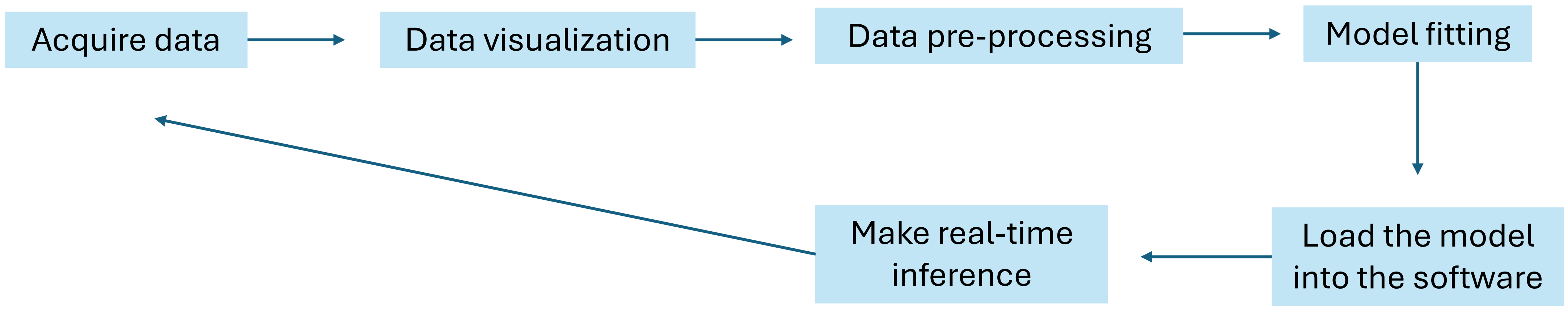
Electroencephalogram (EEG)



- Measures brain electrical activity
- A lot of literature on measuring sleepiness using EEG measurements
- Karolinska sleepiness scale (KSS) level is from 0-10
- 0 for fully awake
- 10 for extremely drowsy

Brainwave Type	Frequency Range (Hz)	State of the brain
Delta (δ)	0.1Hz to 3Hz	Deep, dreamless sleep, non-REM sleep, unconscious
Theta (θ)	4Hz to7Hz	Intuitive, creative, recall, fantasy, imaginary, dream
Alpha (α)	8Hz to12Hz	Relaxed, but not drowsy, tranquil, conscious
Low-range Beta (β)	12Hz to 15Hz	Formerly SMR, relaxed yet focused, integrated
Mid-range Beta (β)	16Hz to 20Hz	Thinking, aware of self & surroundings
High-range Beta (β)	21Hz to 30Hz	Alertness, agitation
Gamma (γ)	30Hz to 100,Hz	Motor Functions, higher mental activity

Machine learning operation



https://zenodo.org/records/2348892

Dataset

EEG Alpha Waves dataset

Grégoire Cattan[1]; Pedro L. C. Rodrigues[1]; Marco Congedo[1]

```
file_path = r'C:\Users\brian\OneDrive - University of Tennessee\Desktop\Research\Python program\RTI-simulator\data_files
files = [x for x in os.listdir(file_path) if re.search('.mat',x)]
print(files)
```

```
['subject_00.mat', 'subject_01.mat', 'subject_02.mat', 'subject_03.mat', 'subject_04.mat', 'subject_05.mat', 'subject_06.ma
t', 'subject_08.mat', 'subject_09.mat', 'subject_10.mat', 'subject_11.mat', 'subject_12.mat', 'subject_13.mat', 'subject_14.
mat', 'subject_15.mat', 'subject_16.mat', 'subject_17.mat', 'subject_18.mat', 'subject_19.mat', 'subject_20.mat']
```

- 21 subjects

https://zenodo.org/records/2348892

Description of data

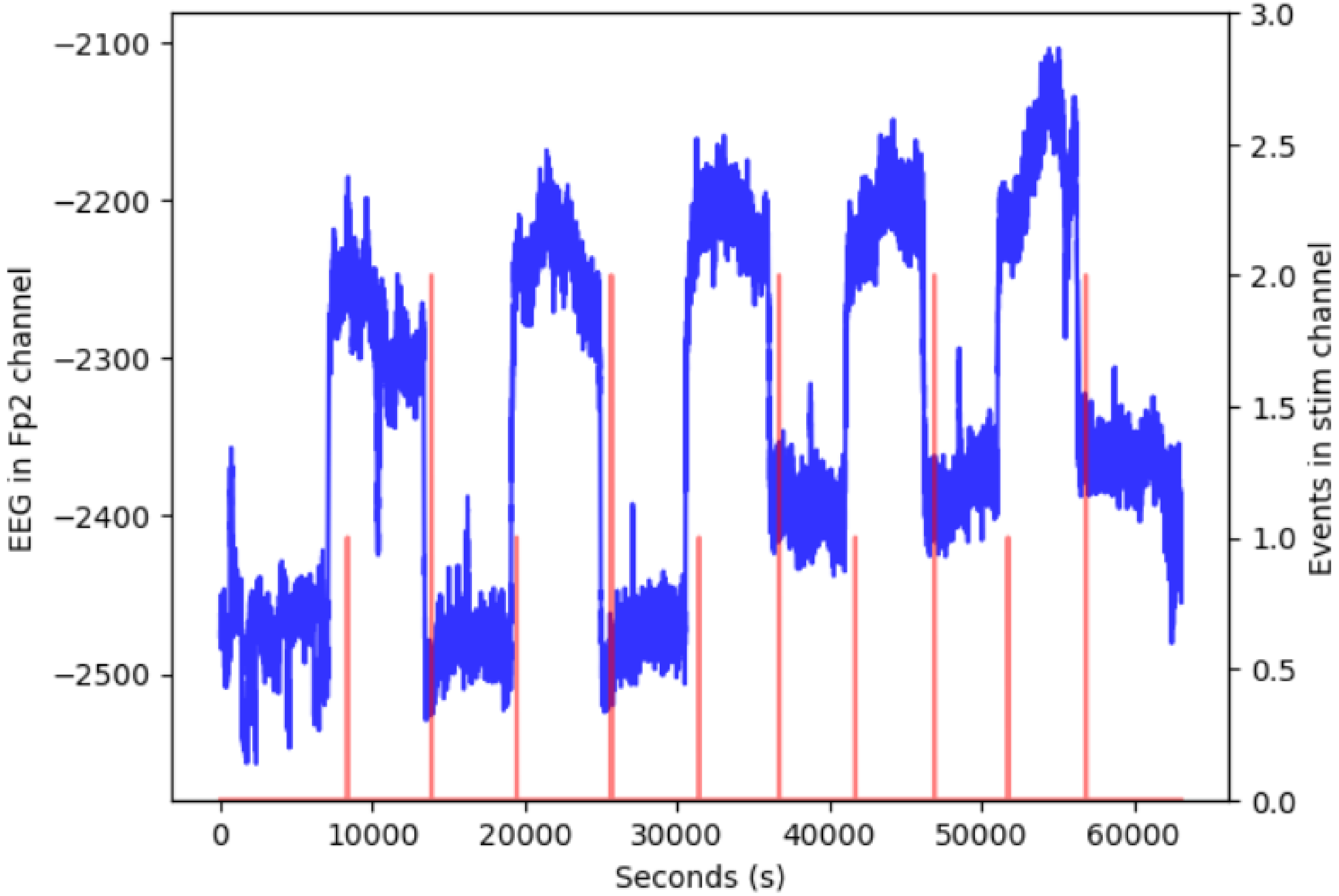
```
1  raw.describe()
```

```
<RawArray | 17 x 63104 (123.2 s), ~8.2 MB, data loaded>
ch  name  type  unit          min             Q1         median             Q3            max
 0  Fp1   EEG   µV     1919117919.90  2181048034.68  2275398925.80  2422695739.77  2577585205.10
 1  Fp2   EEG   µV    -2557157959.00  -2456960937.47  -2362735351.55  -2231222045.88  -2104002685.50
 2  Fc5   EEG   µV     217782211.30  430085990.91  510669525.15  560539947.51  795885375.98
 3  Fz    EEG   µV    -7108479492.20  -6816313232.43  -6781700683.60  -6714244018.57  -6612576660.20
 4  Fc6   EEG   µV    -6534169433.60  -6374752075.20  -6266024902.35  -6208651123.05  -6095611328.10
 5  T7    EEG   µV    -11849524414.00  -11518110351.50  -11400988769.50  -11323991699.50  -11208399414.00
 6  Cz    EEG   µV    -2934821289.10  -2712221618.65  -2555023925.80  -2486022583.05  -2391815185.50
 7  T8    EEG   µV    -14278633789.00  -13982040039.00  -13955098633.00  -13930370361.50  -13821010742.00
 8  P7    EEG   µV    -2057885986.30  -1844913513.15  -1780144714.35  -1737875030.55  -1659822143.60
 9  P3    EEG   µV    -4158770019.50  -3908614929.20  -3739749511.70  -3644156066.90  -3515339599.60
10  Pz    EEG   µV    -6366761230.50  -6145006347.68  -5971622802.75  -5855889892.60  -5726430175.80
11  P4    EEG   µV    -12492496094.00  -12344897460.75  -12311807129.00  -12282636963.00  -12200150391.00
12  P8    EEG   µV    -7381578613.30  -7282695434.55  -7263285156.25  -7240133667.03  -7144030273.40
13  O1    EEG   µV    -59928363.80  26797467.71  52249719.62  84224887.85  241197326.66
14  Oz    EEG   µV     303357452.39  401499443.06  438953948.98  475364578.25  592525390.63
15  O2    EEG   µV    -3355604248.00  -3259574646.00  -3237737304.65  -3215229431.15  -3124399169.90
16  stim  STIM  V              0.00            0.00            0.00            0.00           2.00
```

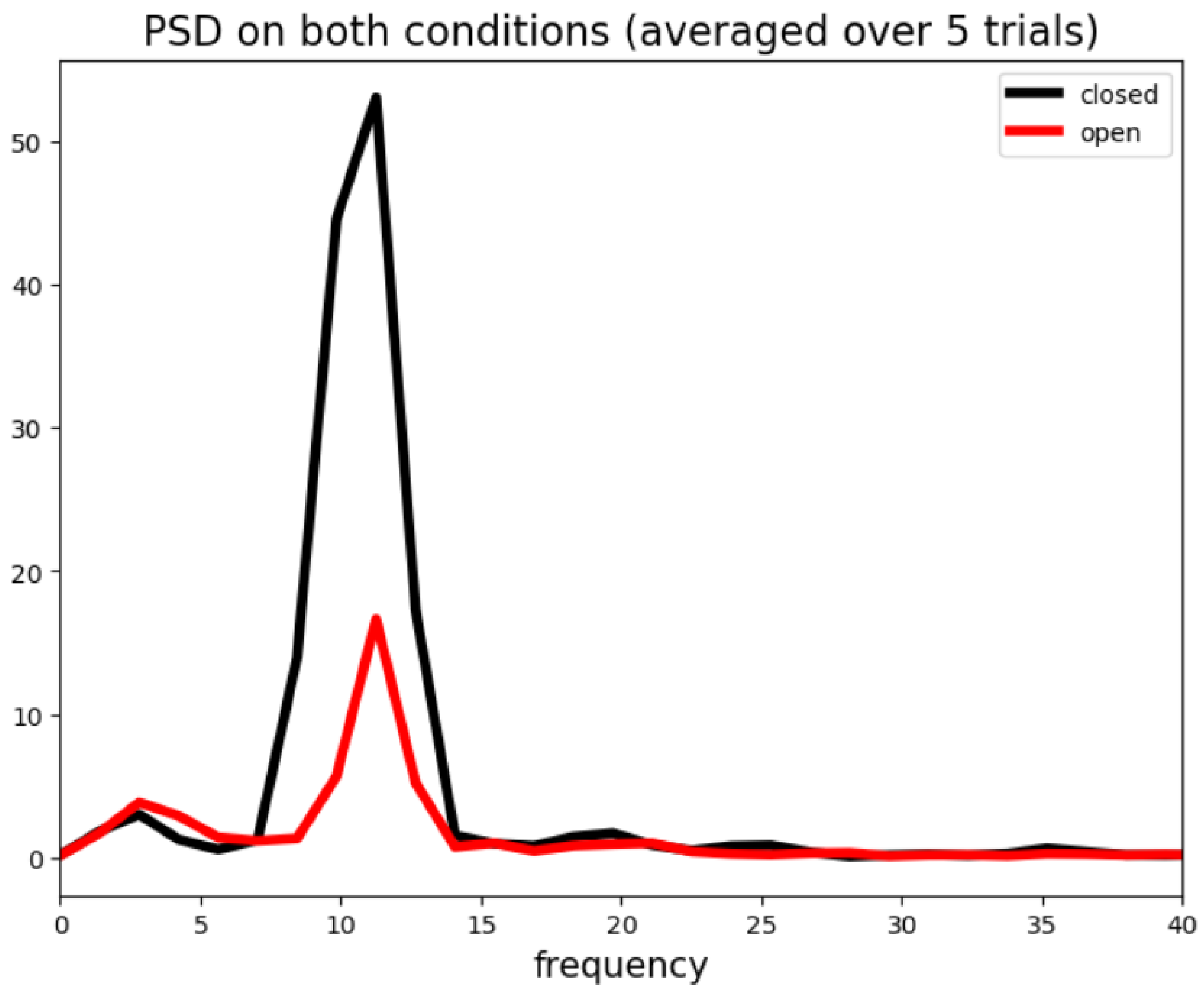
Data visualization

```
1  raw.describe()
```

```
<RawArray | 17 x 63104 (123.2 s), ~8.2 MB, data loaded>
ch  name  type  unit        min          Q1       median          Q3          max
 0  Fp1   EEG   µV    1919117919.90  2181048034.68  2275398925.80  2422695739.77  2577585205.10
 1  Fp2   EEG   µV    -2557157959.00  -2456960937.47  -2362735351.55  -2231222045.88  -2104002685.50
 2  Fc5   EEG   µV    217782211.30  430085990.91  510669525.15  5605
 3  Fz    EEG   µV    -7108479492.20  -6816313232.43  -6781700683.60
 4  Fc6   EEG   µV    -6534169433.60  -6374752075.20  -6266024902.35
 5  T7    EEG   µV    -11849524414.00  -11518110351.50  -11400988769.
 6  Cz    EEG   µV    -2934821289.10  -2712221618.65  -2555023925.80
 7  T8    EEG   µV    -14278633789.00  -13982040039.00  -13955098633.
 8  P7    EEG   µV    -2057885986.30  -1844913513.15  -1780144714.35
 9  P3    EEG   µV    -4158770019.50  -3908614929.20  -3739749511.70
10  Pz    EEG   µV    -6366761230.50  -6145006347.68  -5971622802.75
11  P4    EEG   µV    -12492496094.00  -12344897460.75  -12311807129.
12  P8    EEG   µV    -7381578613.30  -7282695434.55  -7263285156.25
13  O1    EEG   µV    -59928363.80  26797467.71  52249719.62  8422488
14  Oz    EEG   µV    303357452.39  401499443.06  438953948.98  47536
15  O2    EEG   µV    -3355604248.00  -3259574646.00  -3237737304.65
16  stim  STIM  V          0.00        0.00        0.00        0.00
```



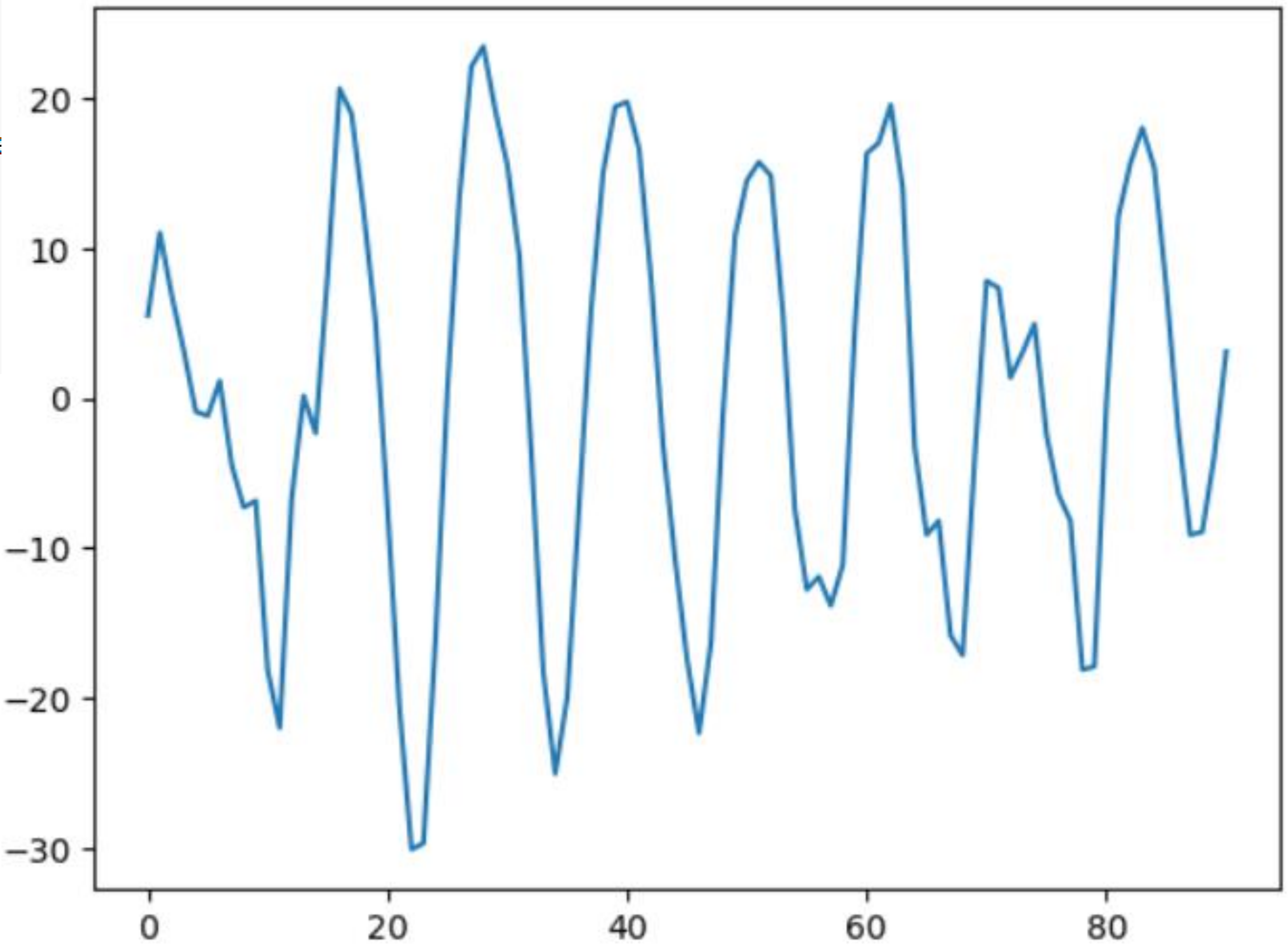
Data visualization



Data pre-processing

```
def processData(raw):
    # filter data and resample
    fmin = 3
    fmax = 40
    raw.filter(fmin, fmax, verbose=False)
    raw.resample(sfreq=128, verbose=False)

    # detect the events and cut the signal into epochs
    events = mne.find_events(raw=raw, shortest_event=1, ve
    event_id = {'closed': 1, 'open': 2}
    epochs = mne.Epochs(raw, events, event_id, tmin=-0.2,
                        verbose=False,preload=True)
    epochs.pick_types(eeg=True)
    return epochs,events
```



Model fitting

- Pre-processing parameters

```
fmin = 3
fmax = 40
```

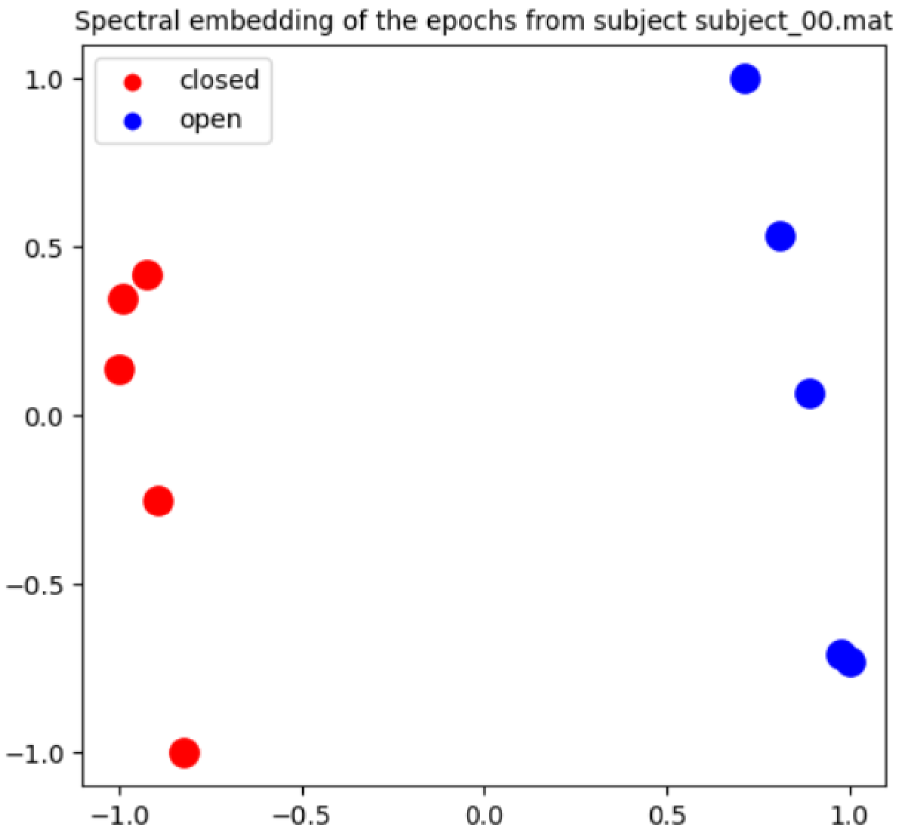
```
tmin=-0.2, tmax=0.2,
```

- One subject, one channel
- 10 measurements
- Standard Scaler
- Cross-validation
- SVC

```
subject subject_00.mat
mean SVC accuracy : 0.9
```

- One subject, one channel
- 10 measurements
- Standard Scaler
- Train/test split
- SVC

```
SCV accuracy score:  1.0
```



Model fitting

- Pre-processing parameters

```
fmin = 3
fmax = 40
```

```
tmin=-0.2, tmax=0.2,
```

- All subjects, All channels
- 200 measurements
- Covariance
- Cross-validation
- SVC

```
X shape:  (200, 16, 91)
y shape:  (200,)
```

```
X_trans = Covariances(estimator='lwf').fit_transform(X,y)
```

```
X_trans = np.array(X_trans)
print('X shape: ', X_trans.shape)
```

```
X shape:  (200, 16, 16)
```

```
mean accuracy : 0.53
```

Model fitting

- All subjects, One channels
- 200 measurements
- Covariance
- Cross-validation
- SVC

```
------------------
Fp1
mean accuracy : 0.
Best parameters se
{'C': 3, 'degree':

Grid best score:
0.8099999999999999
------------------
Fp2
mean accuracy : 0.
Best parameters se
{'C': 1, 'degree':

Grid best score:
0.7700000000000001
------------------
Fc5
mean accuracy : 0.
Best parameters se
{'C': 1, 'degree':

Grid best score:
0.7799999999999999
```

```
P4
mean accuracy : 0.545
Best parameters set found on development set:
{'C': 1, 'degree': 1, 'gamma': 'auto', 'kernel':

Grid best score:
0.76
--------------------------------------------------
P8
mean accuracy : 0.515
Best parameters set found on development set:
{'C': 2, 'degree': 1, 'gamma': 'auto', 'kernel':

Grid best score:
0.8150000000000001
--------------------------------------------------
O1
mean accuracy : 0.51
Best parameters set found on development set:
{'C': 1, 'de      1, 'gamma': 'auto', 'kernel':

Grid best score:
0.86
```

```
Fz
mean accuracy : 0.5399999999999999
                     n development set:
                     a': 'auto', 'kernel': 'poly'}
```

```
--------------------------------------------------
T8
mean accuracy : 0.505
Best parameters set found on development set:
{'C': 3, 'degree': 1, 'gamma': 'scale', 'kernel': 'rbf'}
```

```
------------------------------------------------------------
Oz
mean accuracy : 0.52
Best parameters set found on development set:
{'C': 3, 'degree': 1, 'gamma': 'auto', 'kernel': 'linear'}

Grid best score:
0.8100000000000002
------------------------------------------------------------
O2
mean accuracy : 0.51
Best parameters set found on development set:
{'C': 1, 'degree': 2, 'gamma': 'auto', 'kernel': 'poly'}

Grid best score:
0.8200000000000001
```

```
----------------
et:
rnel': 'poly'}
----------------
et:
rnel': 'linear'}
----------------
et:
ernel': 'rbf'}
```

```
--------------------------------
Grid best score:
0.77
```

Model fitting

- All subjects, One channels
- 200 measurements
- Covariance
- Cross-validation
- Random Forest

```
------------------------------------------------------------------------------------------
Fp1
T8
mean accuracy : 0.65
Best parameters set found on development set:
{'max_depth': 10, 'max_features': 1, 'min_samples_leaf': 1, 'min_samples_split': 2, 'n_estimators': 140}
mators': 120}

Grid best score:
0.715
--------------------------
P7
mean accuracy : 0.755
```

```
Pz
mean accuracy : 0.72
Best parameters set found on development set:
{'max_depth': 5, 'max_features': 7, 'min_samples_leaf': 3, 'min_samples_split': 2, 'n_estimators': 140}

Grid best score:
0.765
```

```
--------------------------
Fc6
mean accuracy : 0.725
Best parameters set found
```

```
Oz
mean accuracy : 0.755
Best parameters set found on development set:
{'max_depth': None, 'max_features': 1, 'min_samples_leaf': 2, 'min_samples_split': 2, 'n_estimators': 100}

Grid best score:
0.78
---------------------------------------------------------------------------------
O2
mean accuracy : 0.75
Best parameters set found on development set:
{'max_depth': 5, 'max_features': 1, 'min_samples_leaf': 1, 'min_samples_split': 2, 'n_estimators': 200}

Grid best score:
0.78
```

```
------------
stimators': 100}
------------
stimators': 140}
------------
timators': 140}
```

Model fitting

- All subjects, One channels
  - 200 measurements
    - Covariance
  - Cross-validation
    - SVC

- All subjects, One channels
  - 200 measurements
    - Covariance
  - Cross-validation
    - Random Forest

O1 channel seems to be the best for SVC (86% accuracy) and P8 channel (83.6% accuracy) for RF.

Model fitting

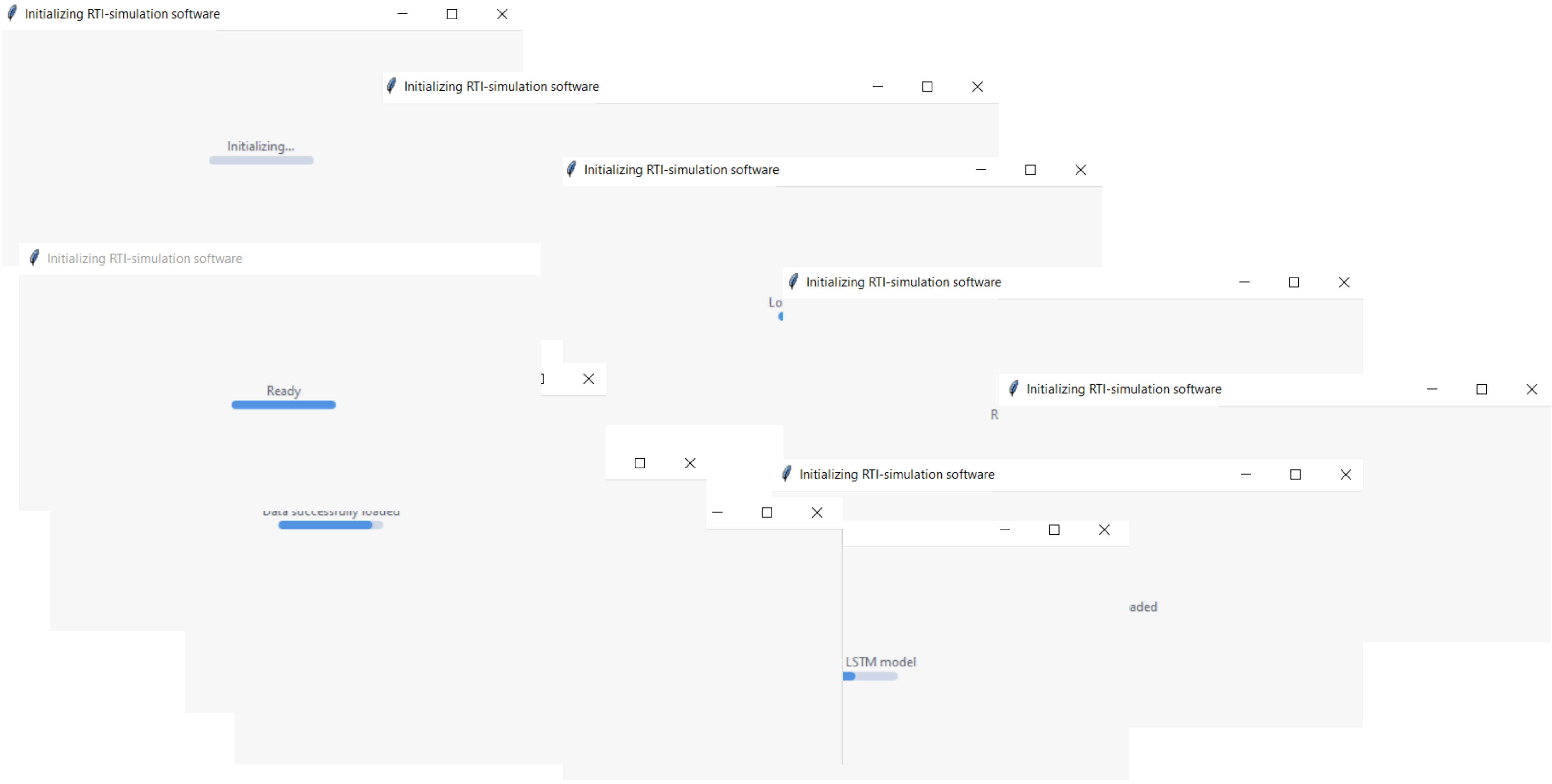
```
def build_LSTM(hidden=16, window=16):
    # Define the model
    LSTM_model = Sequential()
    LSTM_model.name="LSTM"
    LSTM_model.add(LSTM(hidden, input_shape=(window,window),name='LSTM_hidden1'))
    LSTM_model.add(Dense(32,name='Dense_hidden1'))
    LSTM_model.add(Dense(1,name='Output'))
    LSTM_model.compile(optimizer='adam', loss='mae')
    #LSTM_model.summary()
    return LSTM_model

LSTM_model = KerasClassifier(build_fn=build_LSTM(32,16))
LSTM_model
```

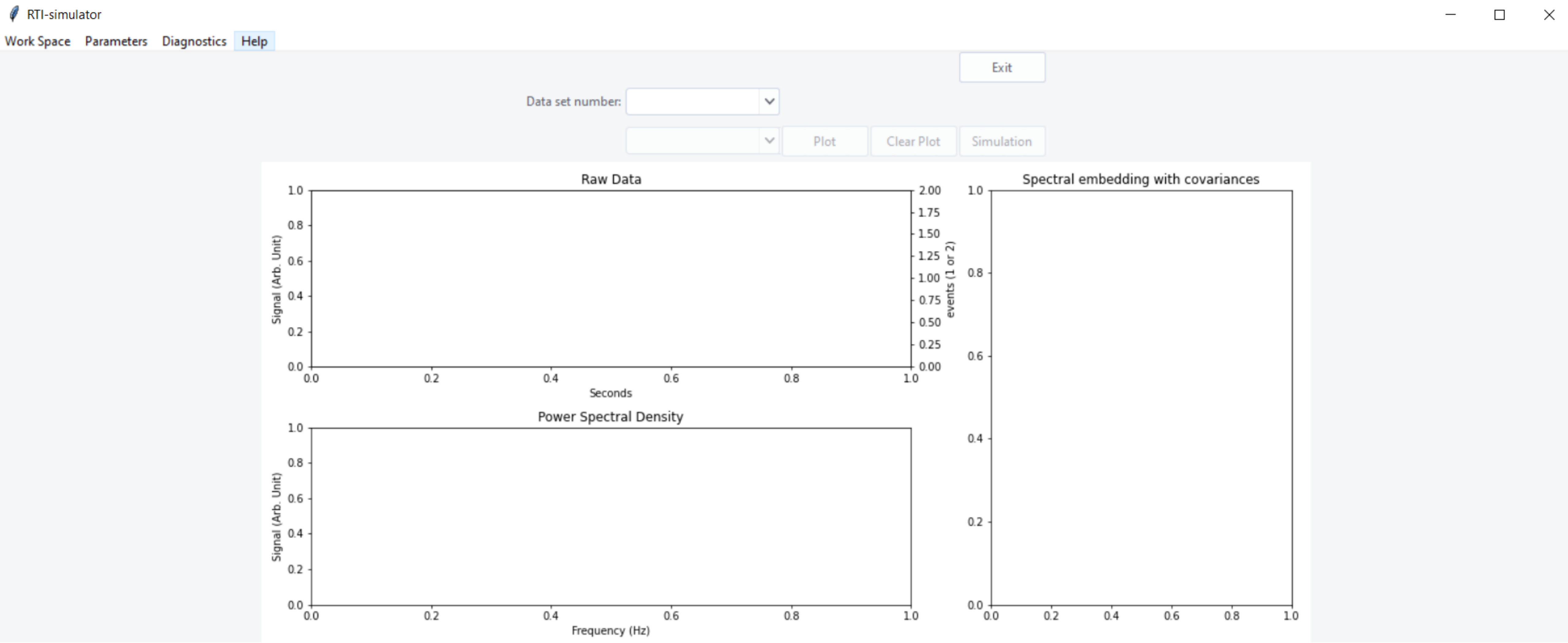
- All subjects, All channels
- 200 measurements
- Covariance
- Cross-validation
- LSTM

```
mean accuracy : 0.97
```

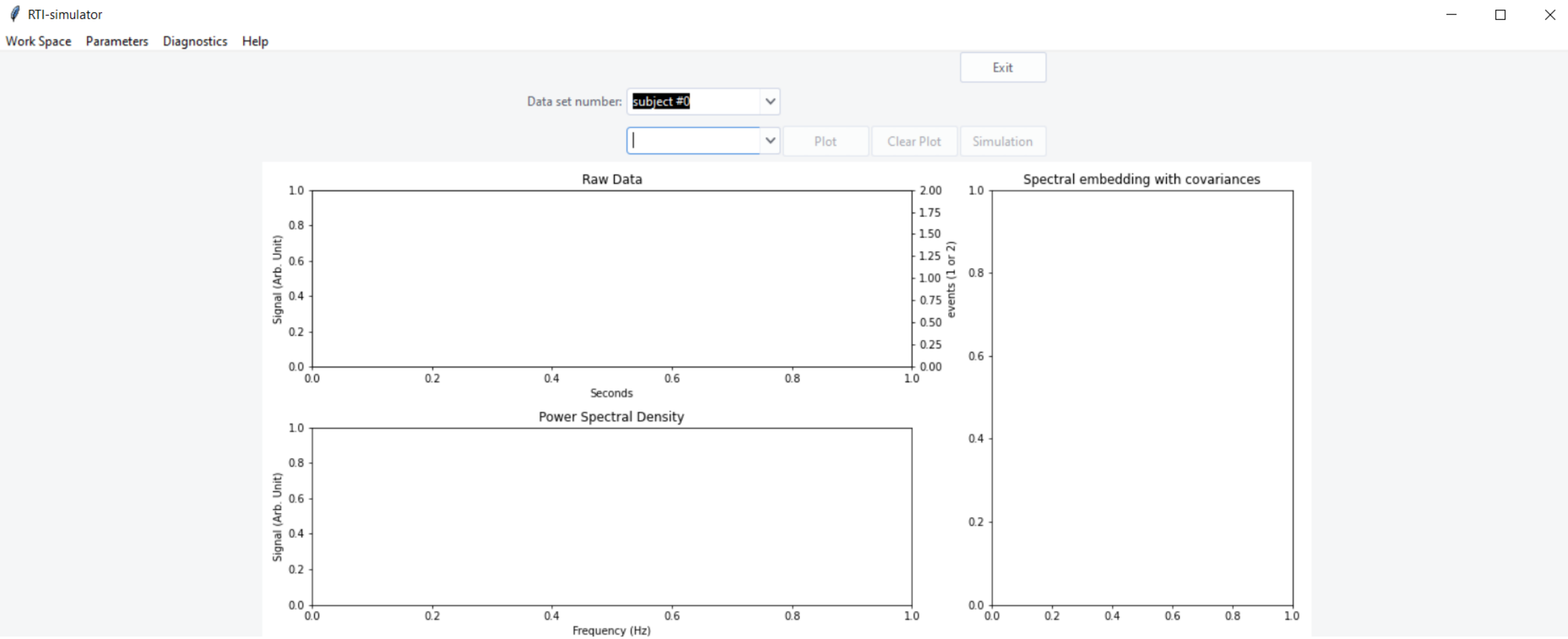
Software Development



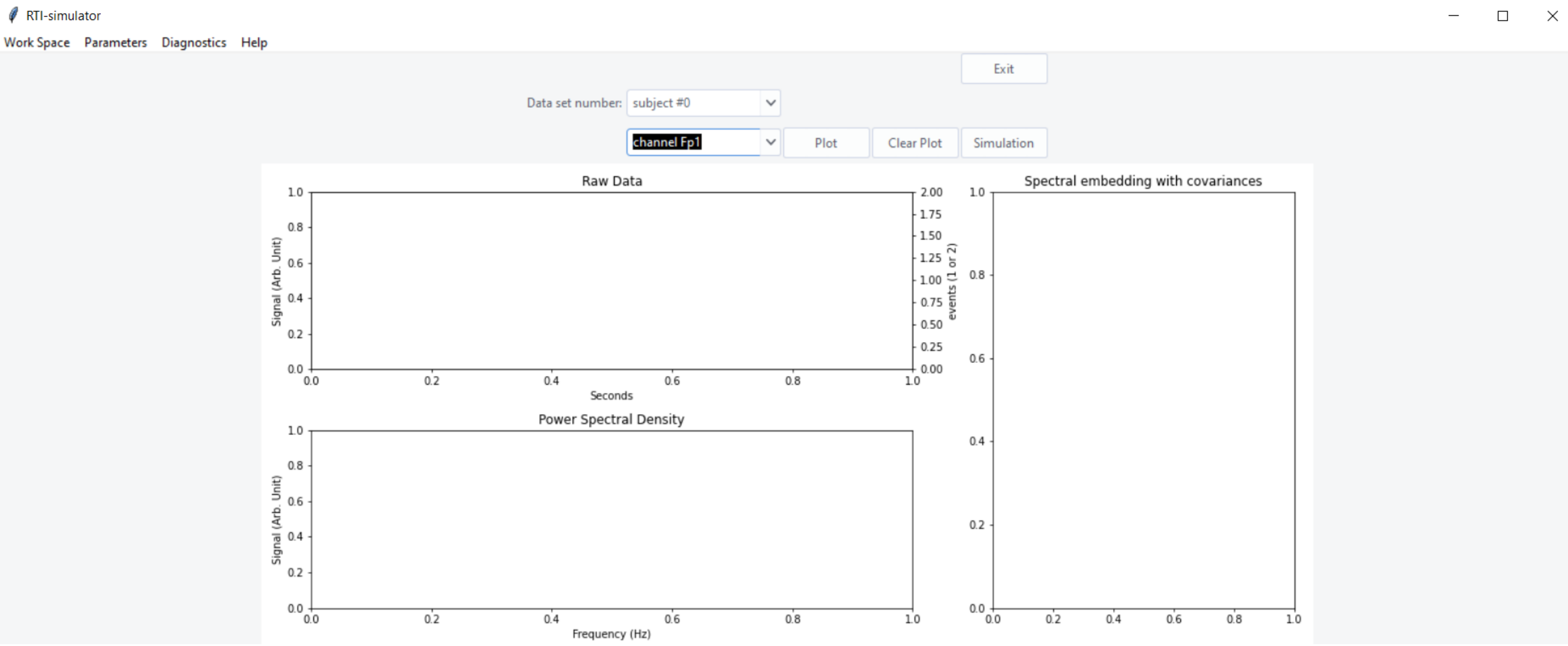
Software Development



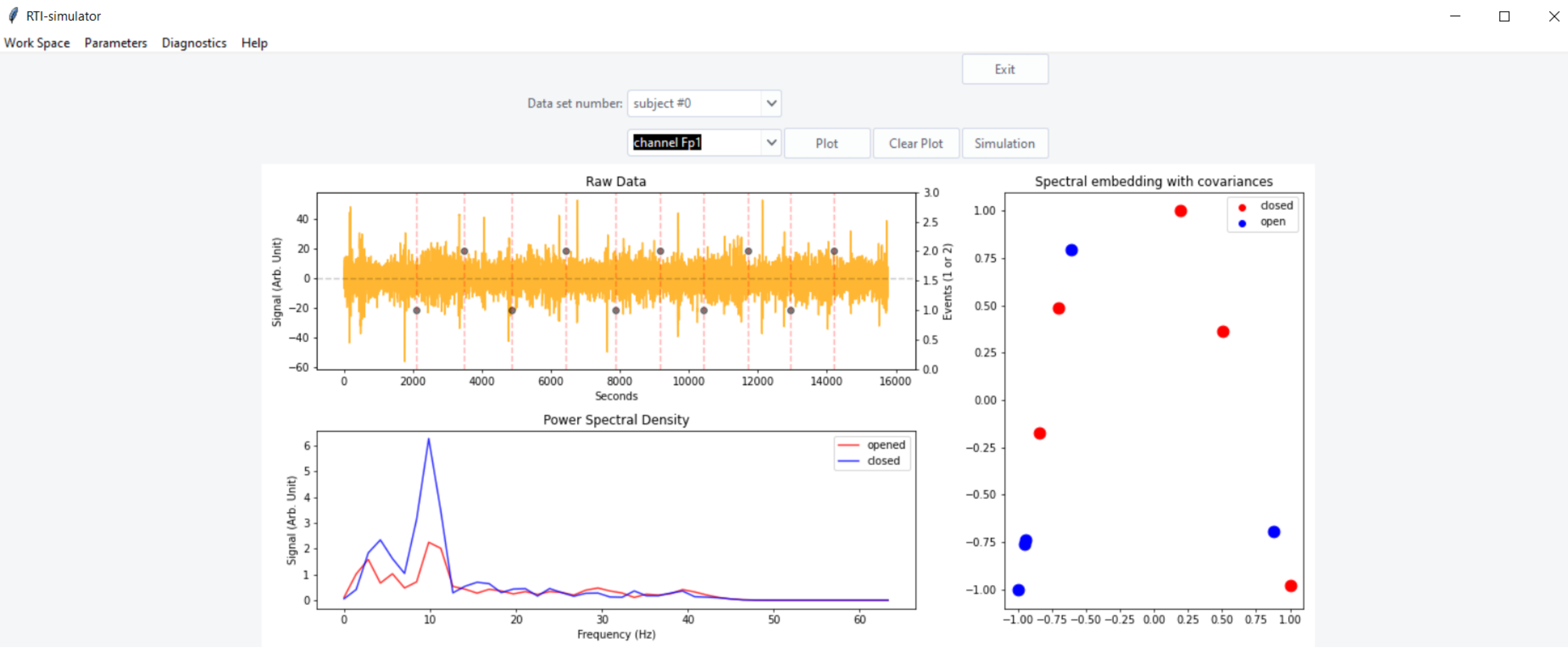
Software Development



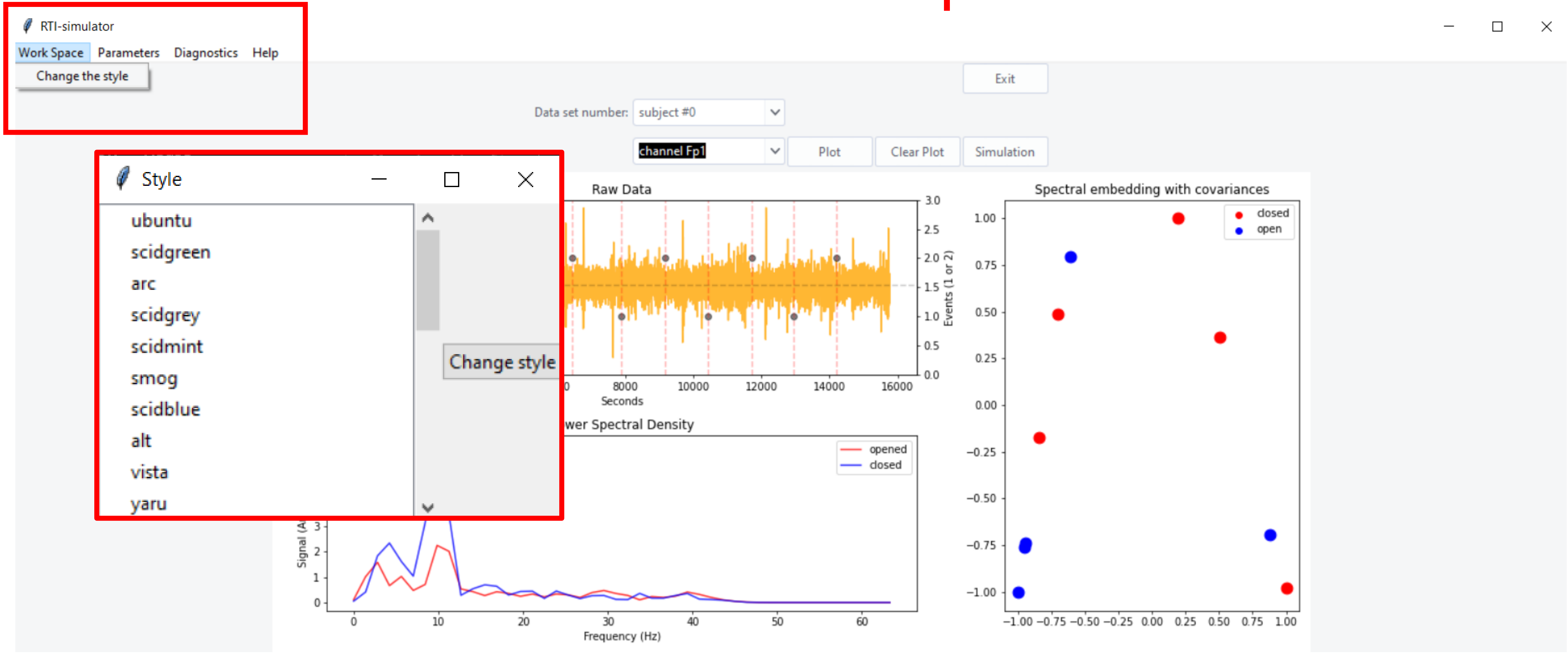
Software Development



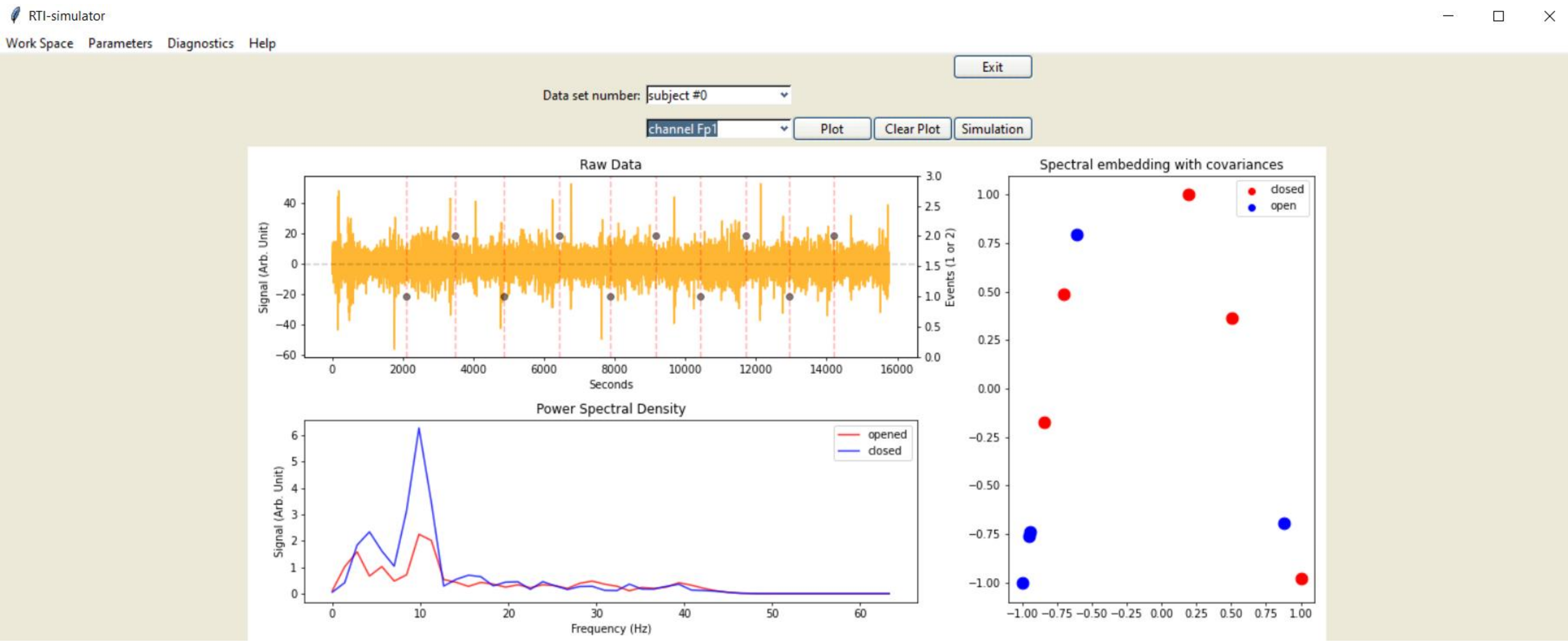
Software Development



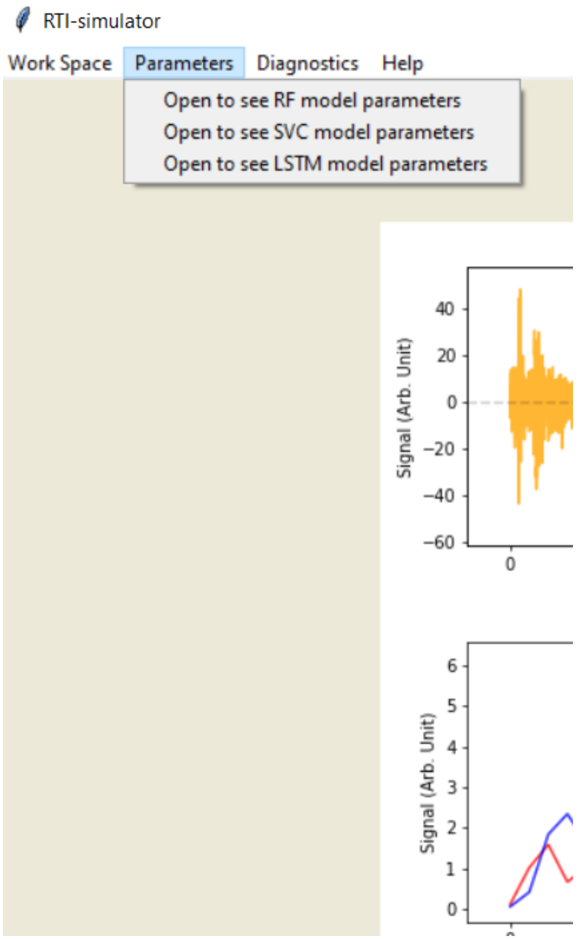
Software Development

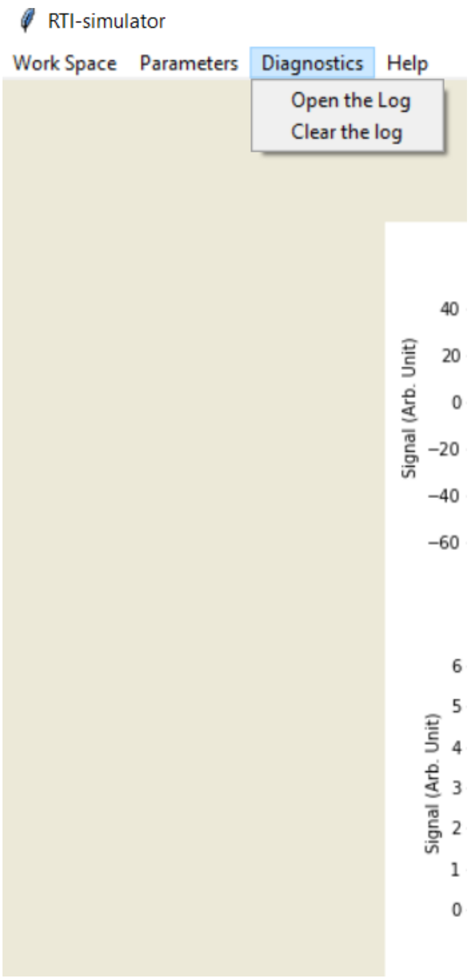


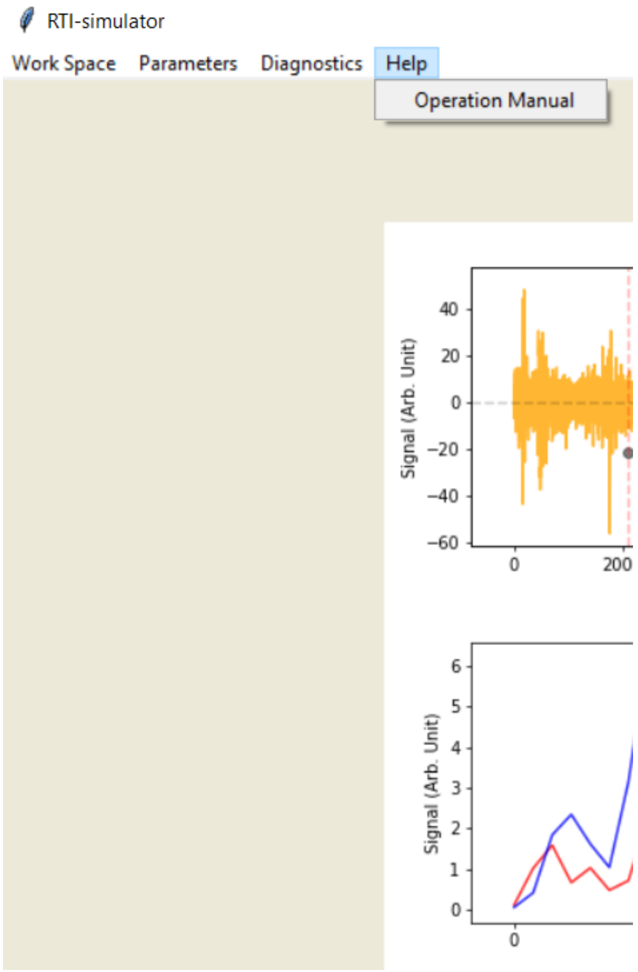
Software Development



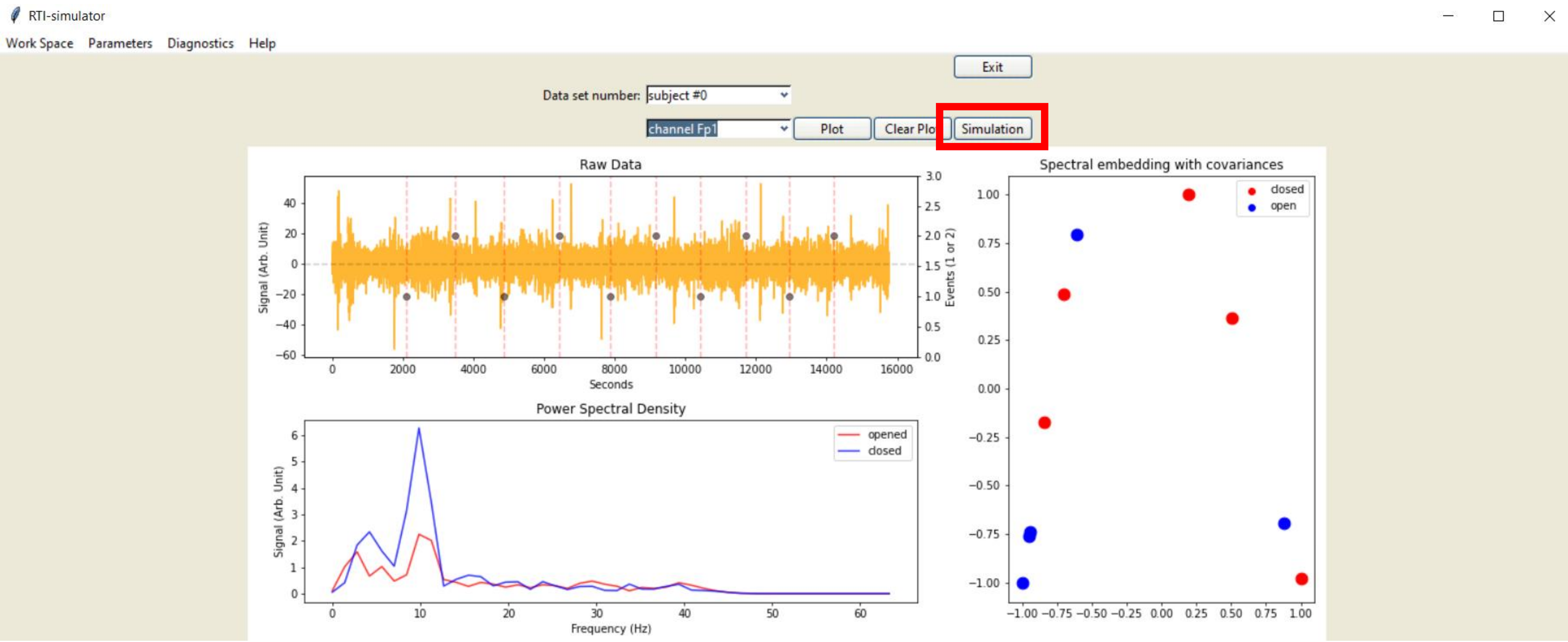
Software Development



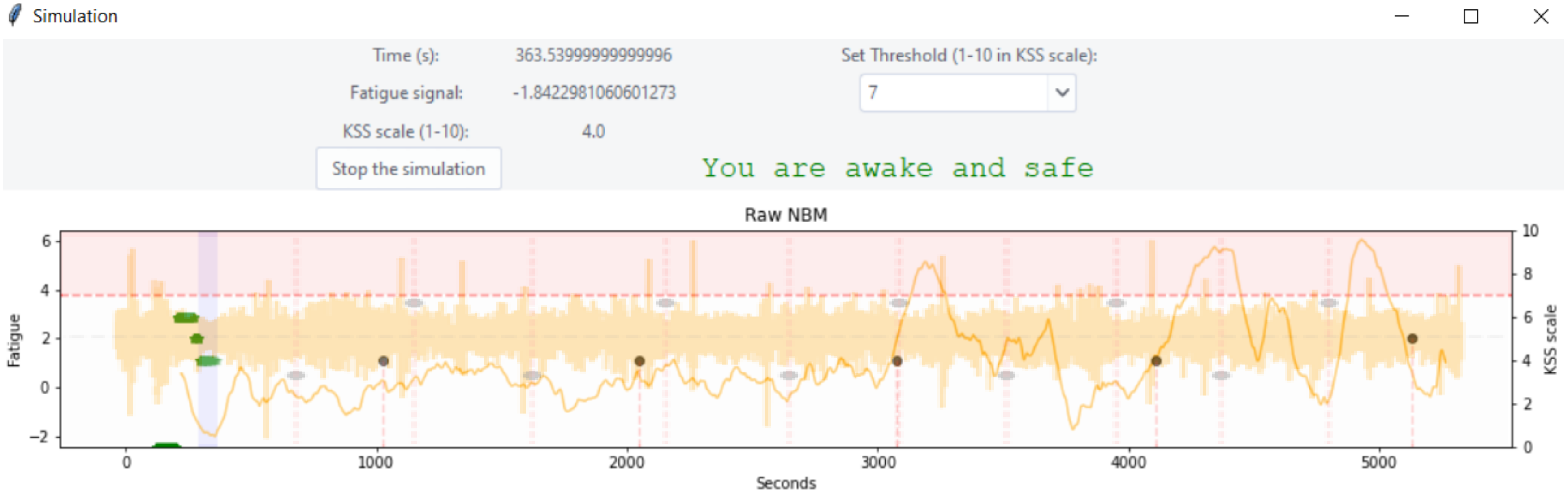


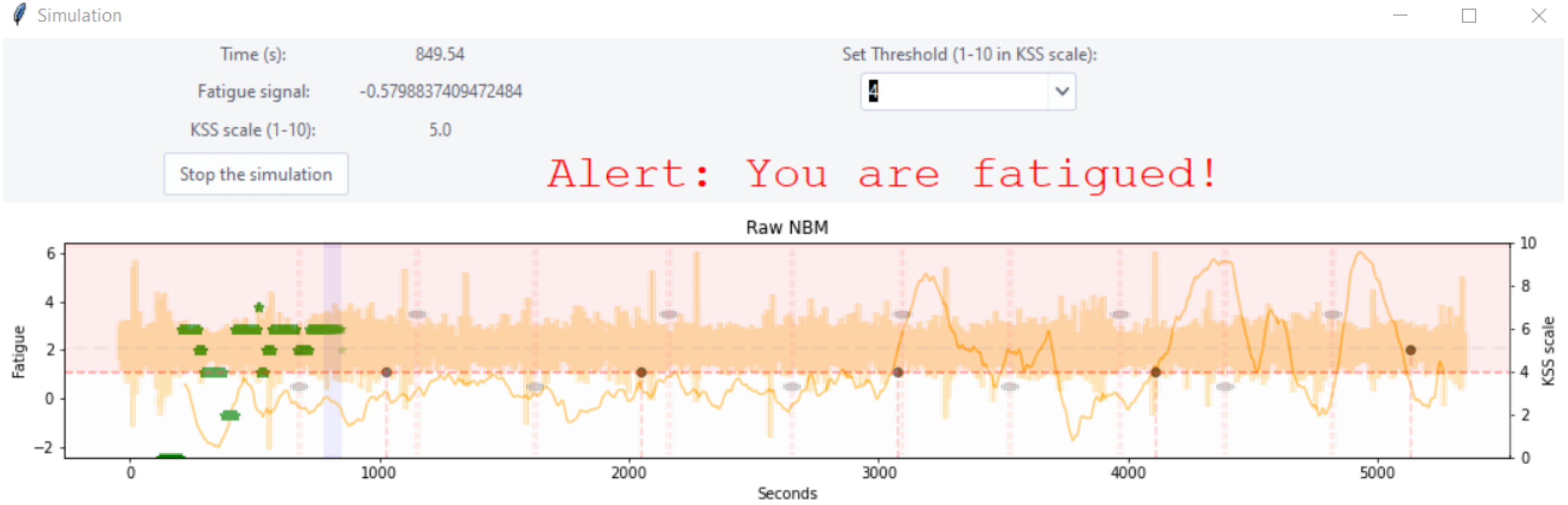


Software Development



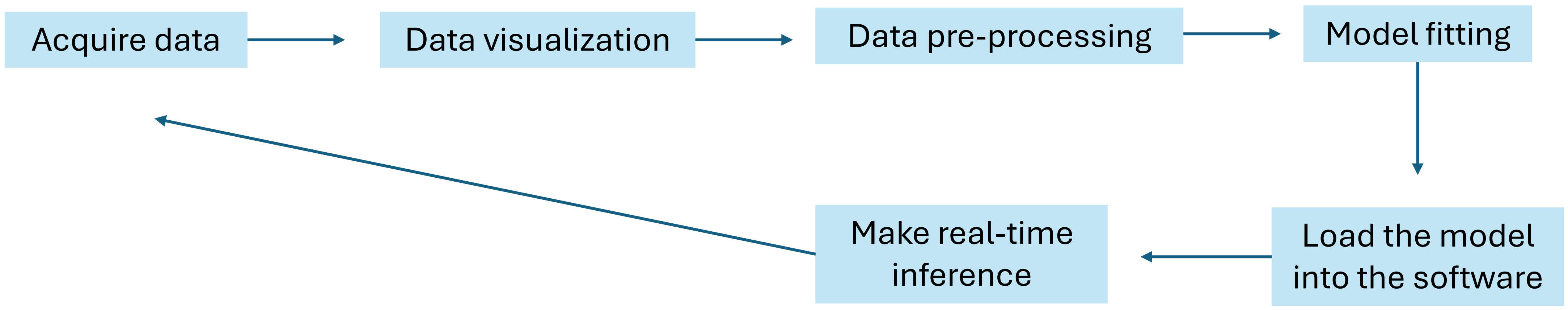
Software Development





Also, the alert has the sound!

Machine learning operation



https://zenodo.org/records/2348892