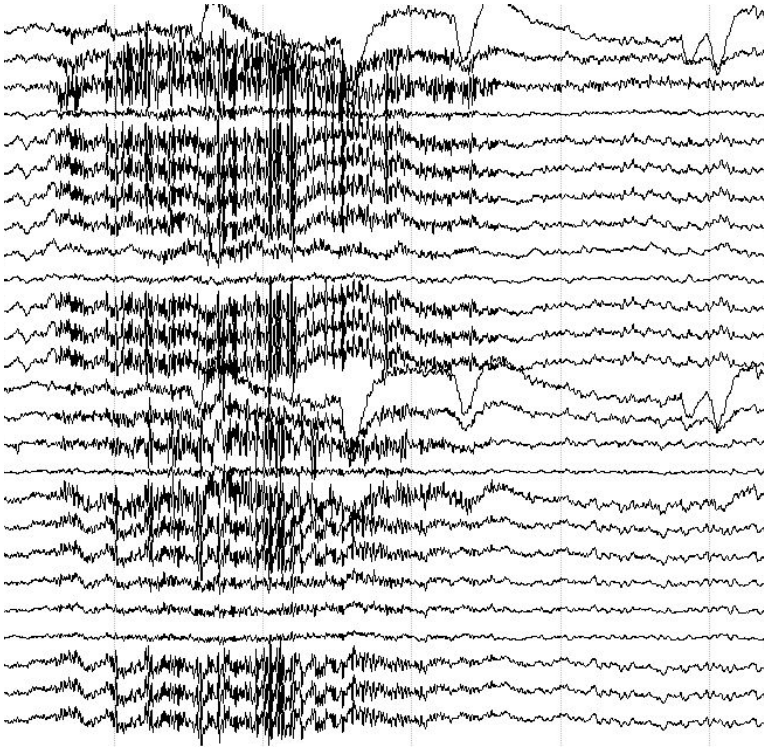
Signals and Noise

“Data is the new gold”

What is a signal? What is noise?



What is a signal? What is noise?

Signal

- Something that carries information
- A phenomenon that is not completely random
- It’s relevant for the observer

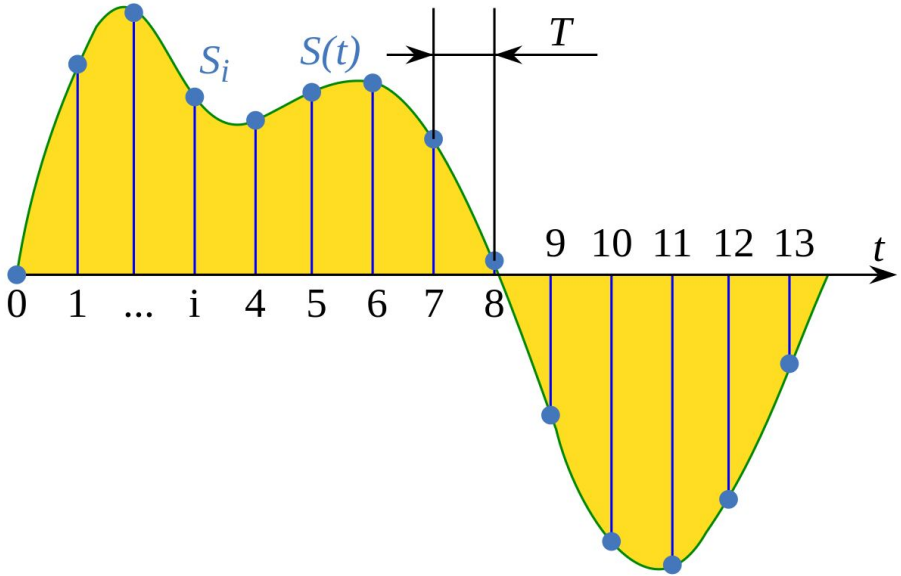
Noise

- Everything else…

One person’s noise is another person’s signal

Sampling

Examining/inspecting/measuring a physical property

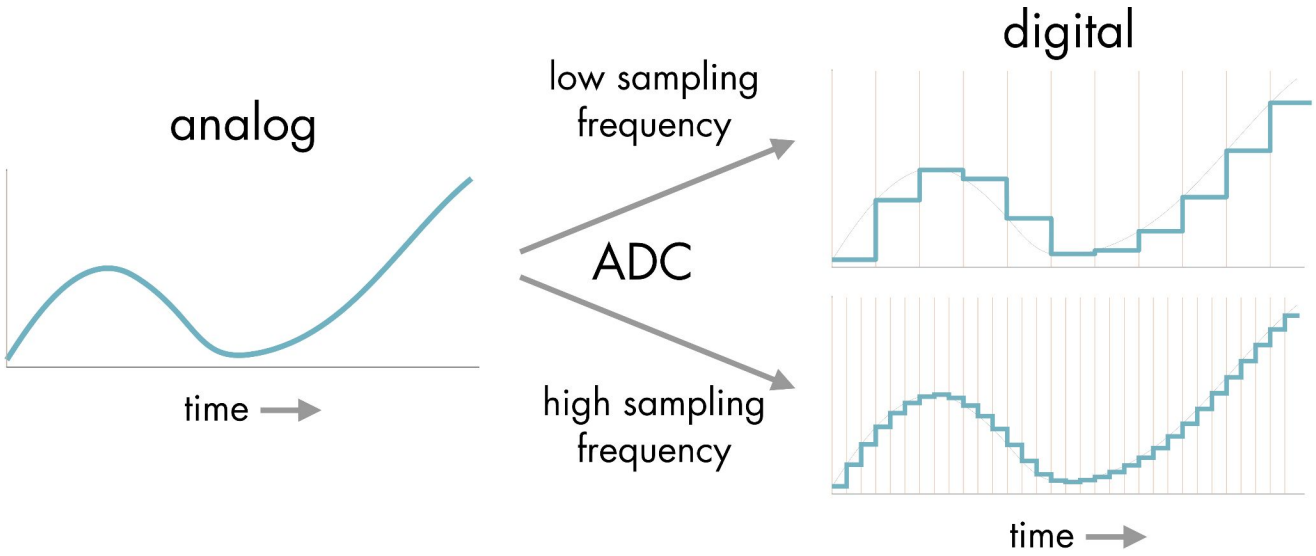


Sampling frequency

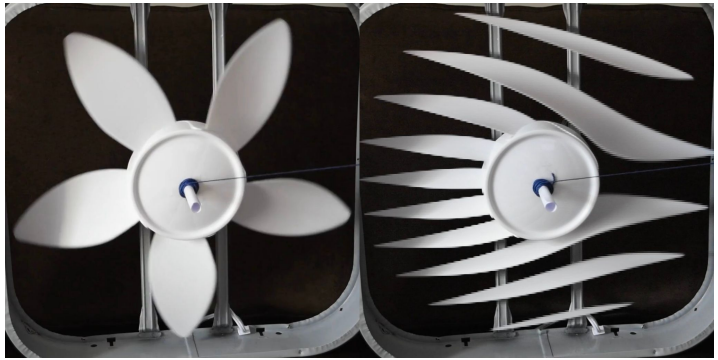
How often? How many times per second? At what rate? How *frequently* do you sample?

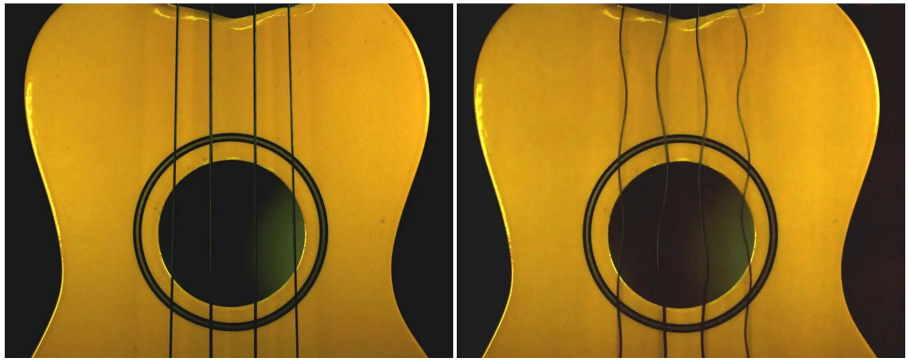
Analog signal	**Sampling frequency**
Sound/audio	
Video	
Motion capture (films)	
Temperature (weather)	
Single Cell Electrophysiology	
BOLD response (MRI)	
Stock price	
Seismology	

Sampling frequency

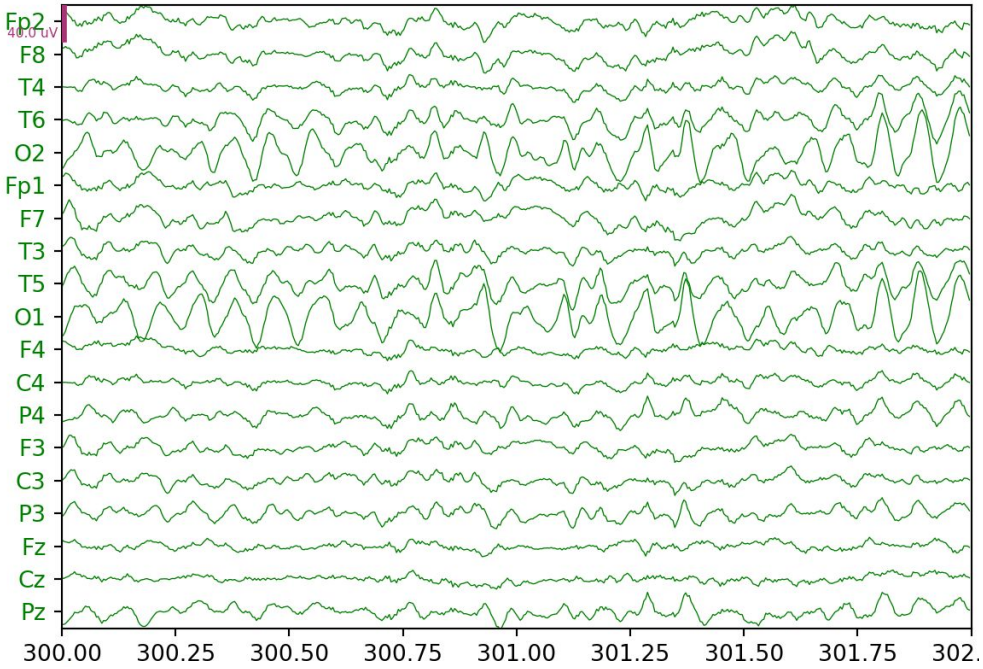


Sampling isn’t always perfect… What’s going on here?



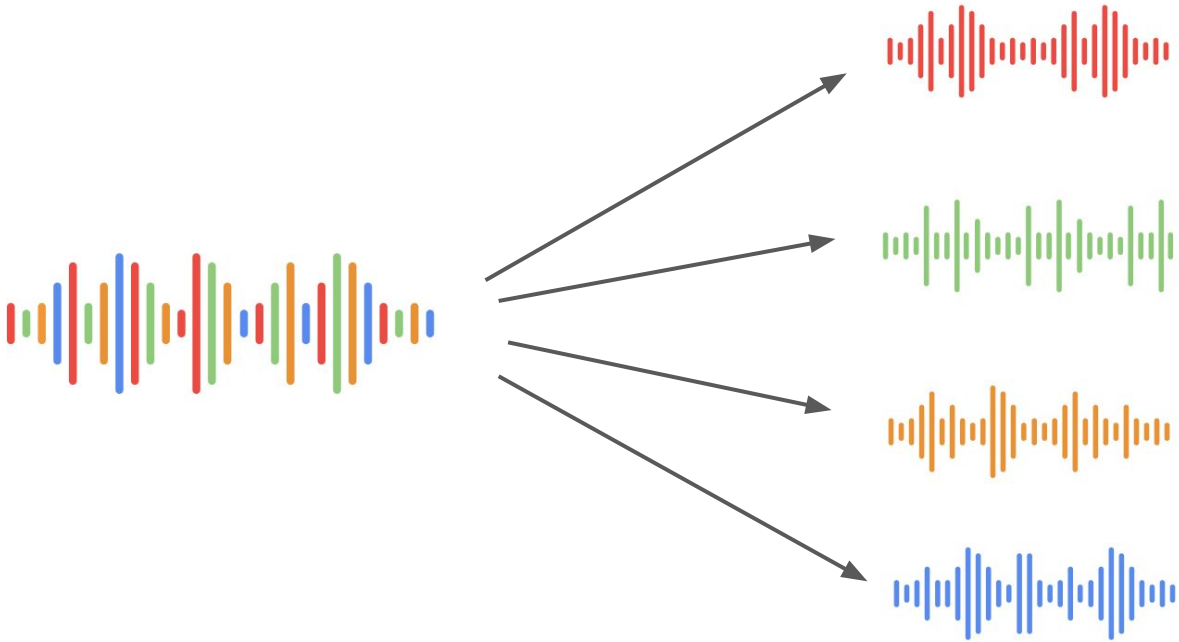


EEG is just another signal

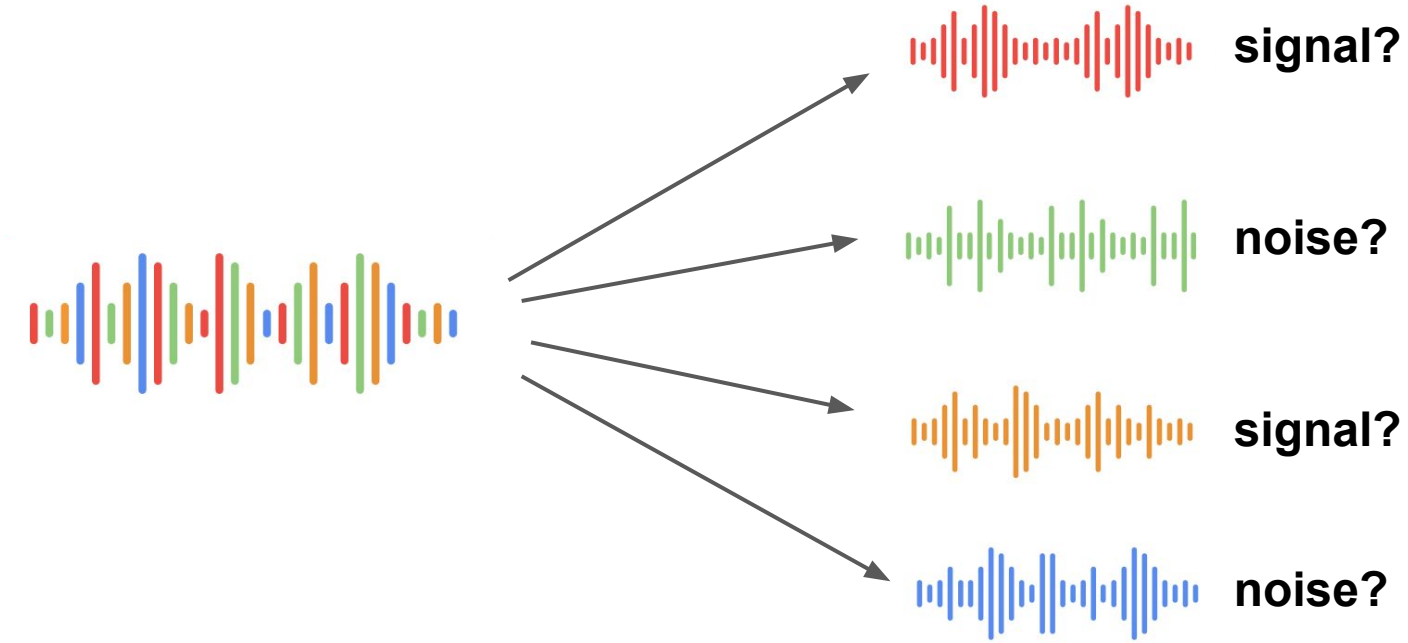


… but it’s very noisy!

Signals are additive



Signals are additive

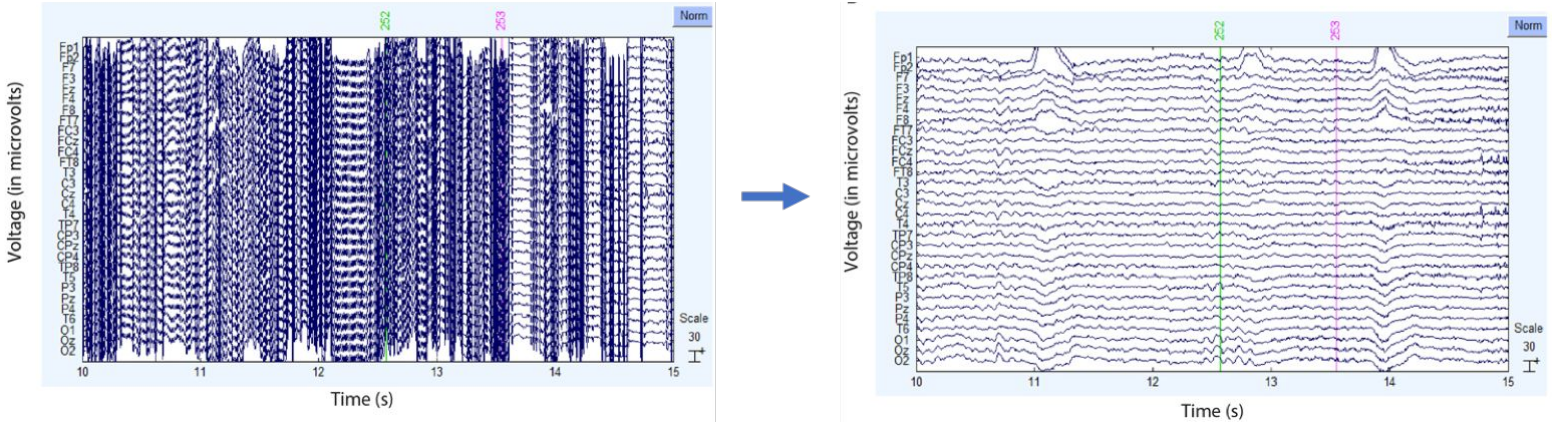


Sources of noise/artefacts in EEG

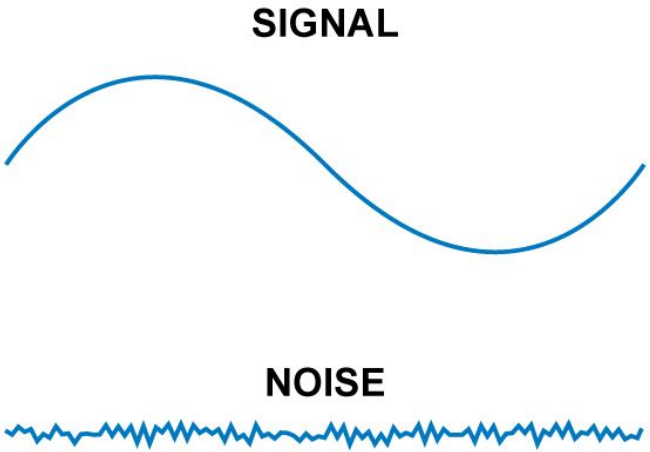
- Biophysical
  - Blinking and eye movement
  - Muscle movement
- Environmental noise
  - Power line alternating current (AC)
  - Room lighting and computer equipment
  - Any electronic equipment close to the sensors
- Instrumentation artifacts
  - Electromagnetic interference (e.g: headphones)
  - Sensor malfunction (random jumps, constant zero signal)

Dealing with noise

- **Ignore** the artifact and carry on with analysis
- **Exclude** the corrupted portion of the data and analyze the remaining data
- **Repair** the artifact by suppressing artifactual part of the recording while (hopefully) leaving the signal of interest intact



Signal-to-Noise Ratio (SNR)



$$SNR = \frac{P_{signal}}{P_{noise}}$$

Wanted component

Unwanted component

Hansen’s Axiom

“There is no substitute for clean data”

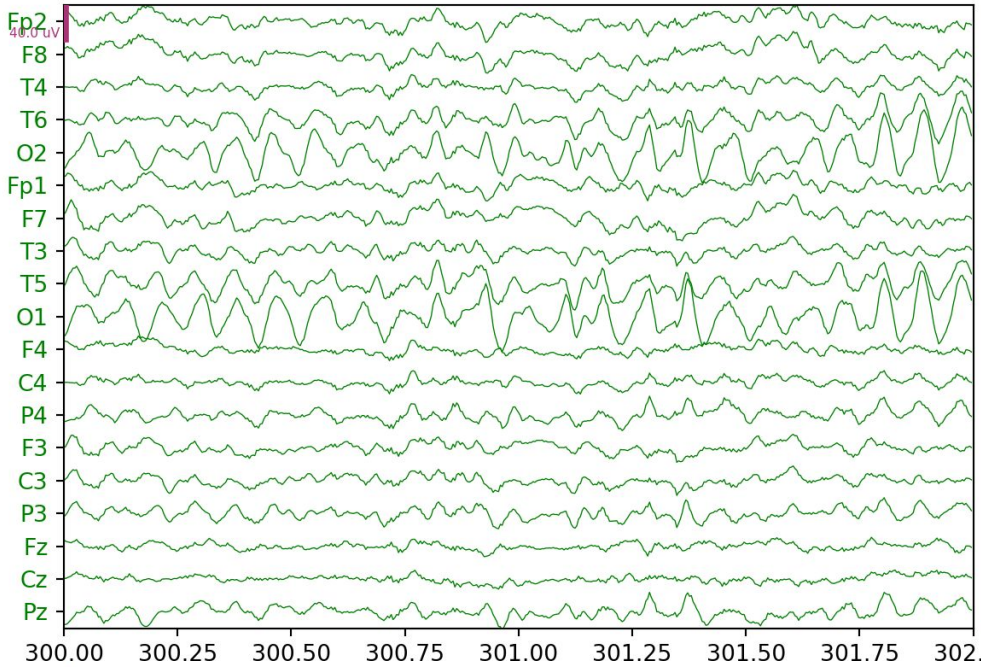
Summary

- Signal vs Noise
  - Carries information
  - Relevant for the observer
- Sampling frequency
  - How often do you measure?
  - Undersampling
- Removing noise
  - Signals are additive
  - EEG is noisy

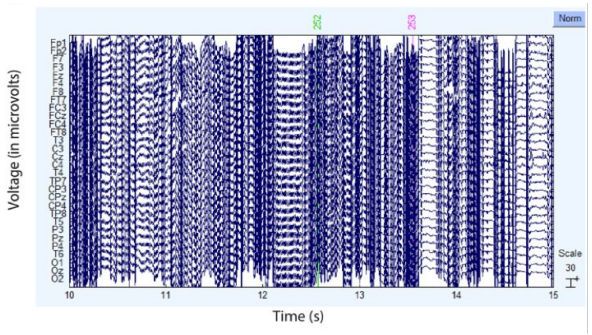
Good luck with the EEG data mining!

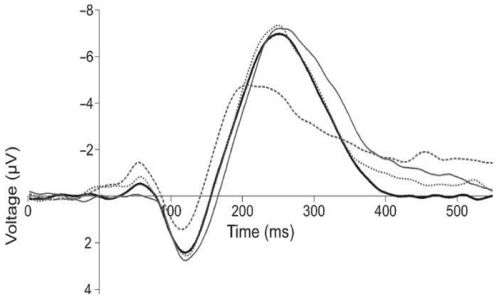
EEG signals in python

Reminder: EEG is just another signal…



From signal to meaning → we need smart tools

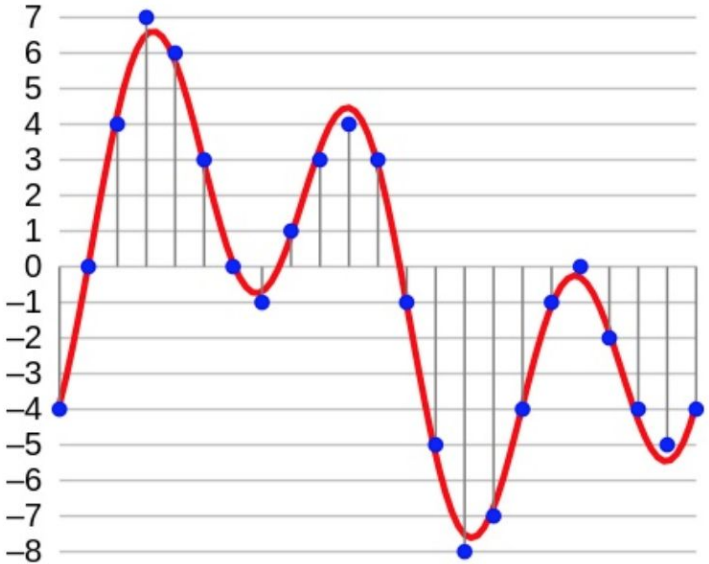




python is all about lists

```
# Define a list of strings
list_1 = ["New York", "Tokyo", "Montreal", "Berlin"]

# List of integers
list_2 = list((1, 5, 8, 9))

# List of booleans
list_3 = [True, False, False, True]

# Mixed list
list_4 = list((True, 25, False, "hello"))
```

Signals in python

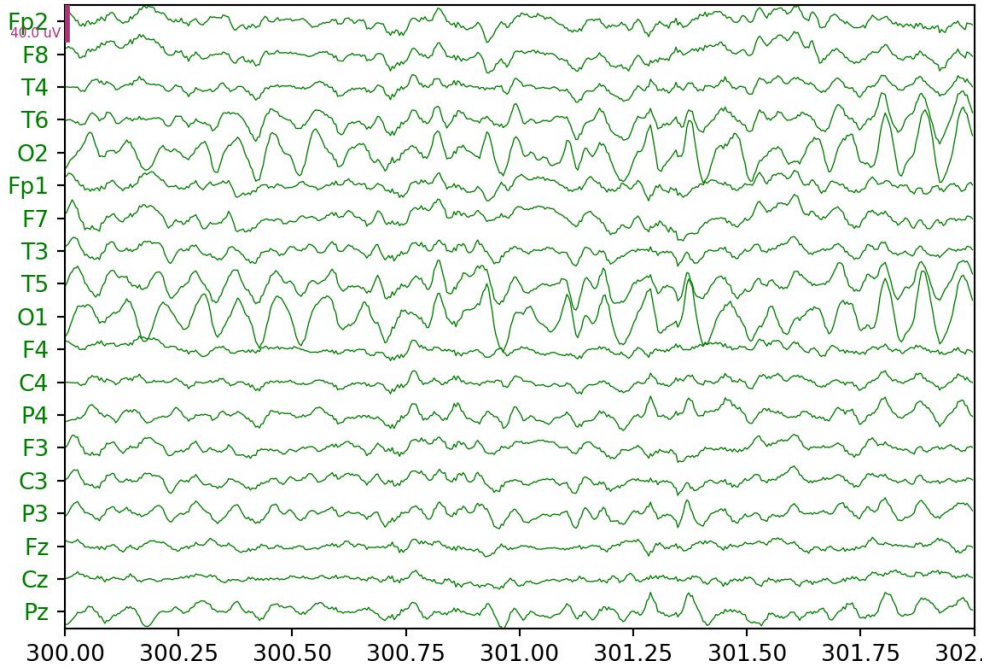



```
list = [-4, 0, 4, 7, 6, 3, 0, -1, …]
```

What else do you need to be able to tell **when** things are happening?

!!! SAMPLE RATE !!!

Signals in python - EEG



⟹ `list_Fp2 = […, …, …]`

⟹ `list_F8 = […, …, …]`

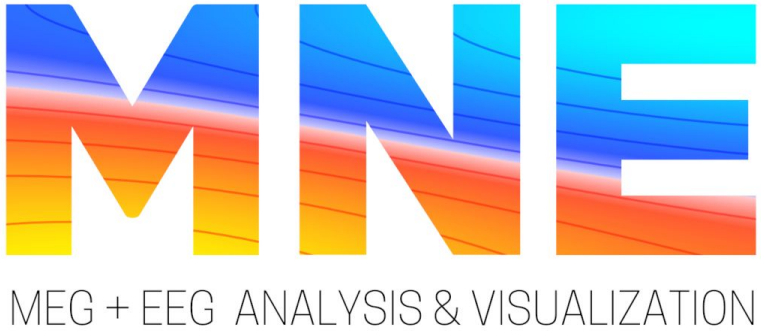
⟹ `list_T4 = […, …, …]`

⟹ …

MNE python

“Open-source Python package for exploring, visualizing, and analyzing human neurophysiological data: MEG, EEG, sEEG, ECoG, NIRS, and more”

Use the documentation as a reference



File types in our lab

Brainvision files

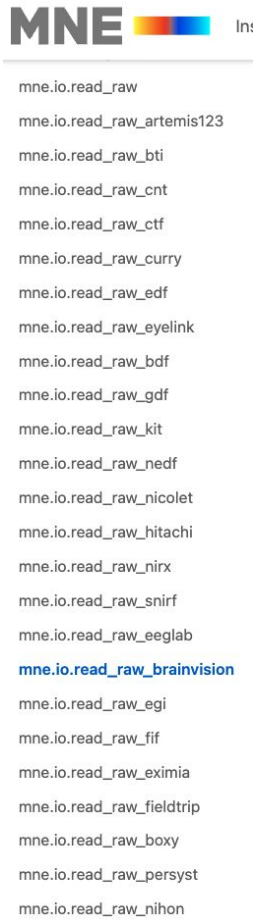
- Header file (.vhdr)
  - Recording parameters and meta-information
- Marker file (.vmrk)
  - Text file describing the events
- EEG data file (.eeg)
  - Recorded EEG signals

File types in our lab

Brainvision files

- Header file (.vhdr)
  - Recording parameters and meta-information
- Marker file (.vmrk)
  - Text file describing the events
- EEG data file (.eeg)
  - Recorded EEG signals

But other file types also exist



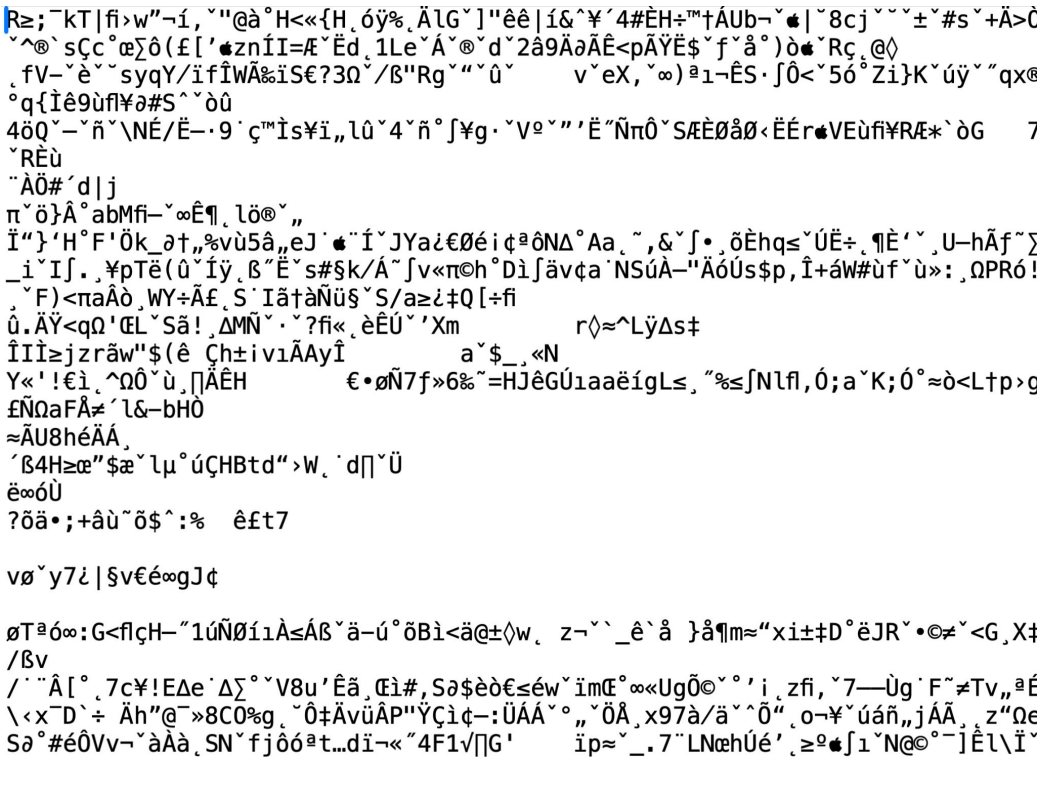
Header file (.vhdr)

```
Brain Vision Data Exchange Header File Version 1.0
; Data created by the Vision Recorder

[Common Infos]
Codepage=UTF-8
DataFile=sub25_main1.eeg
MarkerFile=sub25_main1.vmrk
DataFormat=BINARY
; Data orientation: MULTIPLEXED=ch1,pt1, ch2,pt1 ...
DataOrientation=MULTIPLEXED
NumberOfChannels=64
; Sampling interval in microseconds
SamplingInterval=2000

[Binary Infos]
BinaryFormat=INT_16

[Channel Infos]
; Each entry: Ch<Channel number>=<Name>,<Reference channel name>,
; <Resolution in "Unit">,<Unit>, Future extensions..
; Fields are delimited by commas, some fields might be omitted (empty).
; Commas in channel names are coded as "\1".
Ch1=1,,0.1,µV
Ch2=2,,0.1,µV
Ch3=3,,0.1,µV
Ch4=4,,0.1,µV
Ch5=5,,0.1,µV
Ch6=6,,0.1,µV
Ch7=7,,0.1,µV
Ch8=8,,0.1,µV
Ch9=9,,0.1,µV
Ch10=10,,0.1,µV
Ch11=11,,0.1,µV
Ch12=12,,0.1,µV
Ch13=13,,0.1,µV
Ch14=14,,0.1,µV
Ch15=15,,0.1,µV
Ch16=16,,0.1,µV
Ch17=17,,0.1,µV
Ch18=18,,0.1,µV
Ch19=19,,0.1,µV
Ch20=20,,0.1,µV
Ch21=21,,0.1,µV
Ch22=22,,0.1,µV
```

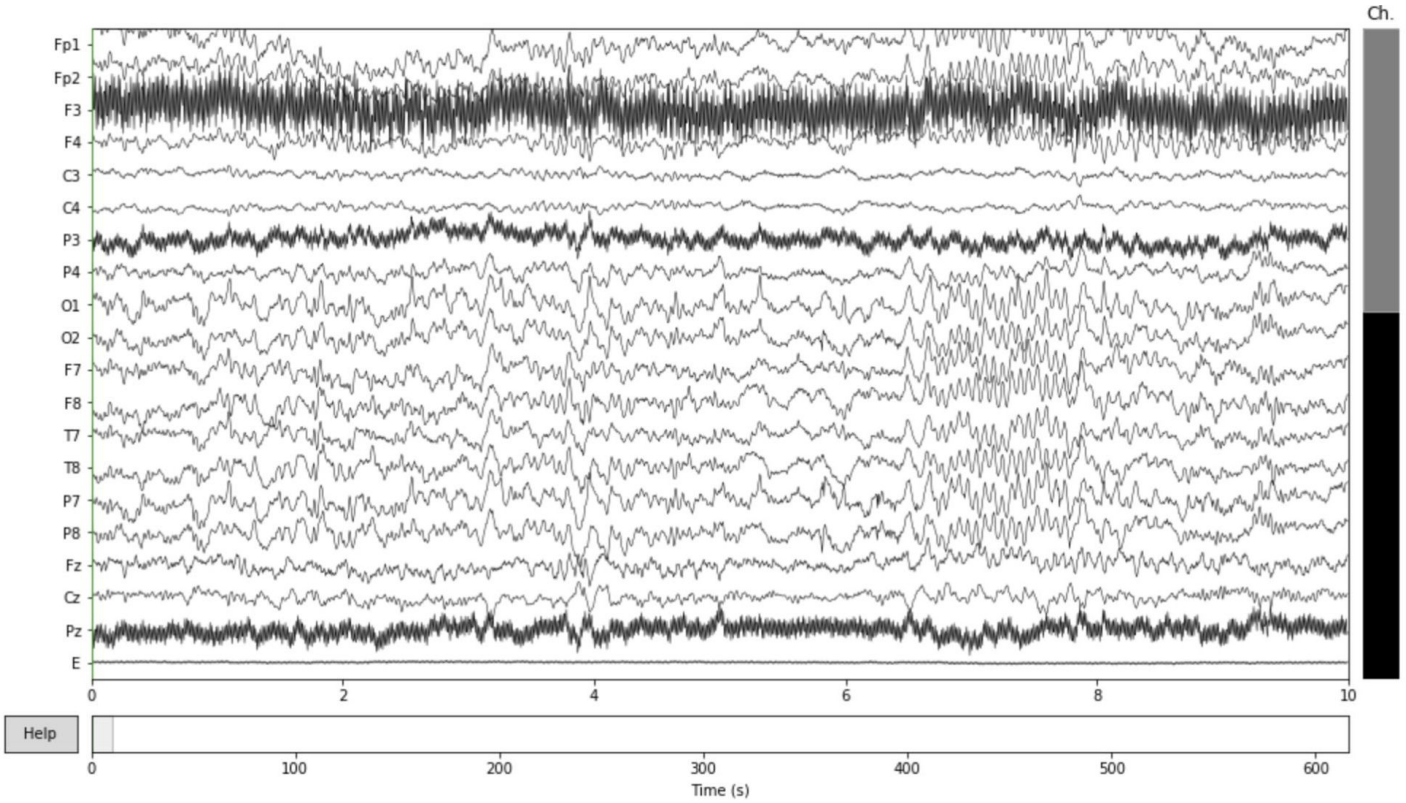
Marker file (.vmrk)

```
Brain Vision Data Exchange Marker File, Version 1.0

[Common Infos]
Codepage=UTF-8
DataFile=sub25_main1.eeg

[Marker Infos]
; Each entry: Mk<Marker number>=<Type>,<Description>,<Position in data points>,
; <Size in data points>, <Channel number (0 = marker is related to all channels)>
; Fields are delimited by commas, some fields might be omitted (empty).
; Commas in type or description text are coded as "\1".
Mk1=New Segment,,1,1,0,20240419173324132949
Mk2=Stimulus,S 68,18843,1,0
Mk3=Stimulus,S  4,19341,1,0
Mk4=Stimulus,S  4,19839,1,0
Mk5=Stimulus,S 16,20840,1,0
Mk6=Stimulus,S  4,21338,1,0
Mk7=Stimulus,S  4,21838,1,0
Mk8=Stimulus,S 16,22338,1,0
Mk9=Stimulus,S 64,22836,1,0
Mk10=Stimulus,S  4,23336,1,0
Mk11=Stimulus,S  4,23842,1,0
Mk12=Stimulus,S 16,24342,1,0
Mk13=Stimulus,S  4,24840,1,0
Mk14=Stimulus,S  4,25342,1,0
Mk15=Stimulus,S  4,25841,1,0
Mk16=Stimulus,S 16,26341,1,0
Mk17=Stimulus,S 64,26839,1,0
Mk18=Stimulus,S  4,27337,1,0
Mk19=Stimulus,S 16,27836,1,0
Mk20=Stimulus,S  4,28339,1,0
Mk21=Stimulus,S  4,28837,1,0
Mk22=Stimulus,S 16,29336,1,0
Mk23=Stimulus,S  4,29842,1,0
Mk24=Stimulus,S  4,30340,1,0
```

EEG data file (.eeg)



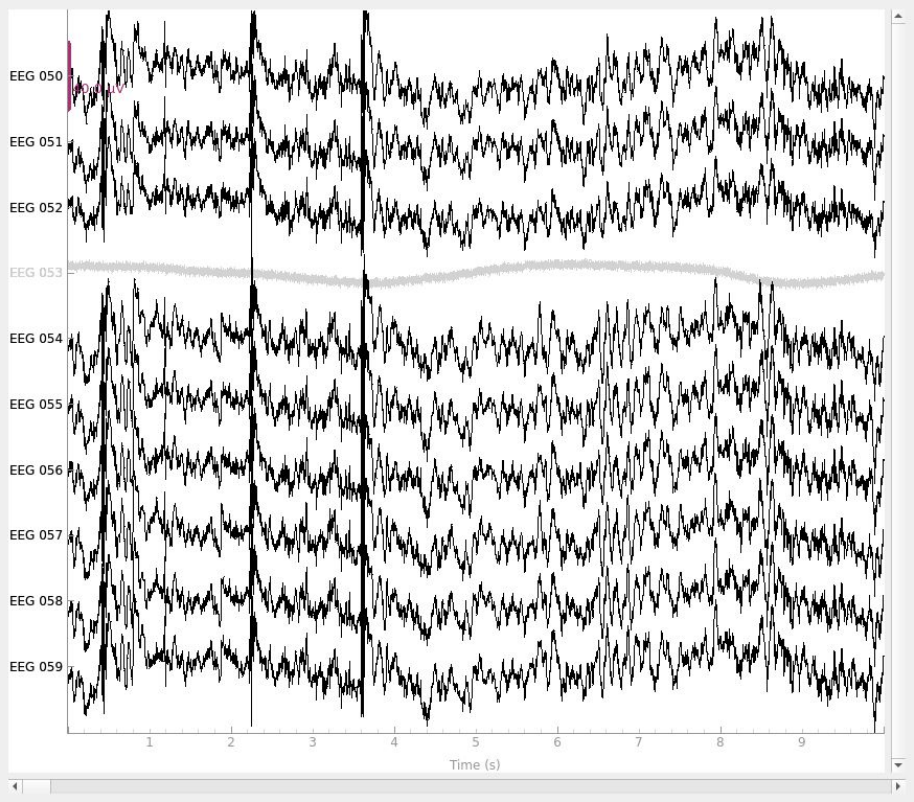
MNE Raw


Ch.
Fp1
Fp2
F3
F4
C3
C4
P3
P4
O1
O2
F7
F8
T7
T8
P7
P8
Fz
Cz
Pz
E
0
2
4
6
8
10
Help
0
100
200
300
400
500
600
Time (s)

Extracting data from Raw objects

Python code	Result
`raw.get_data()`	`NumPy array` (n_chans × n_samps)
`raw[:]`	`tuple` of (data (n_chans × n_samps), times (1 × n_samps))
`raw.get_data(return_times=True)`	
`raw[0, 1000:2000]`	`tuple` of (data (1 × 1000), times (1 × 1000))
`raw['MEG 0113', 1000:2000]`	
`raw.get_data(picks=0, start=1000, stop=2000, return_times=True)`	
`raw.get_data(picks='MEG 0113', start=1000, stop=2000, return_times=True)`	
`raw[7:9, 1000:2000]`	`tuple` of (data (2 × 1000), times (1 × 1000))
`raw[[2, 5], 1000:2000]`	
`raw[['EEG 030', 'EOG 061'], 1000:2000]`	

Bad channels



MNE Info object

```
<Info | 14 non-empty values
 bads: 2 items (MEG 2443, EEG 053)
 ch_names: MEG 0113, MEG 0112, MEG 0111, MEG 0122, MEG 0123, MEG 0121, MEG ...
 chs: 204 Gradiometers, 102 Magnetometers, 9 Stimulus, 60 EEG, 1 EOG
 custom_ref_applied: False
 dev_head_t: MEG device -> head transform
 dig: 146 items (3 Cardinal, 4 HPI, 61 EEG, 78 Extra)
 highpass: 0.1 Hz
 hpi_meas: 1 item (list)
 hpi_results: 1 item (list)
 lowpass: 40.0 Hz
 meas_date: 2002-12-03 19:01:10 UTC
 meas_id: 4 items (dict)
 nchan: 376
 projs: PCA-v1: off, PCA-v2: off, PCA-v3: off, Average EEG reference: off
 sfreq: 150.2 Hz
>
```

Use the MNE documentation

- MNE documentation website
- Introductory tutorials
- The Raw data structure
- The Info data structure