# Introduction to EEG data collection & analysis

**Yi Chun Ko 柯逸均**
Graduate Institute of Linguistics, National Taiwan University
Brain and Language Processing lab

# Objectives

This session aims to provide introduction to:

- data structure of EEG/ERP
- steps of pre-processing
- scripts to analyze EEG/ERP data

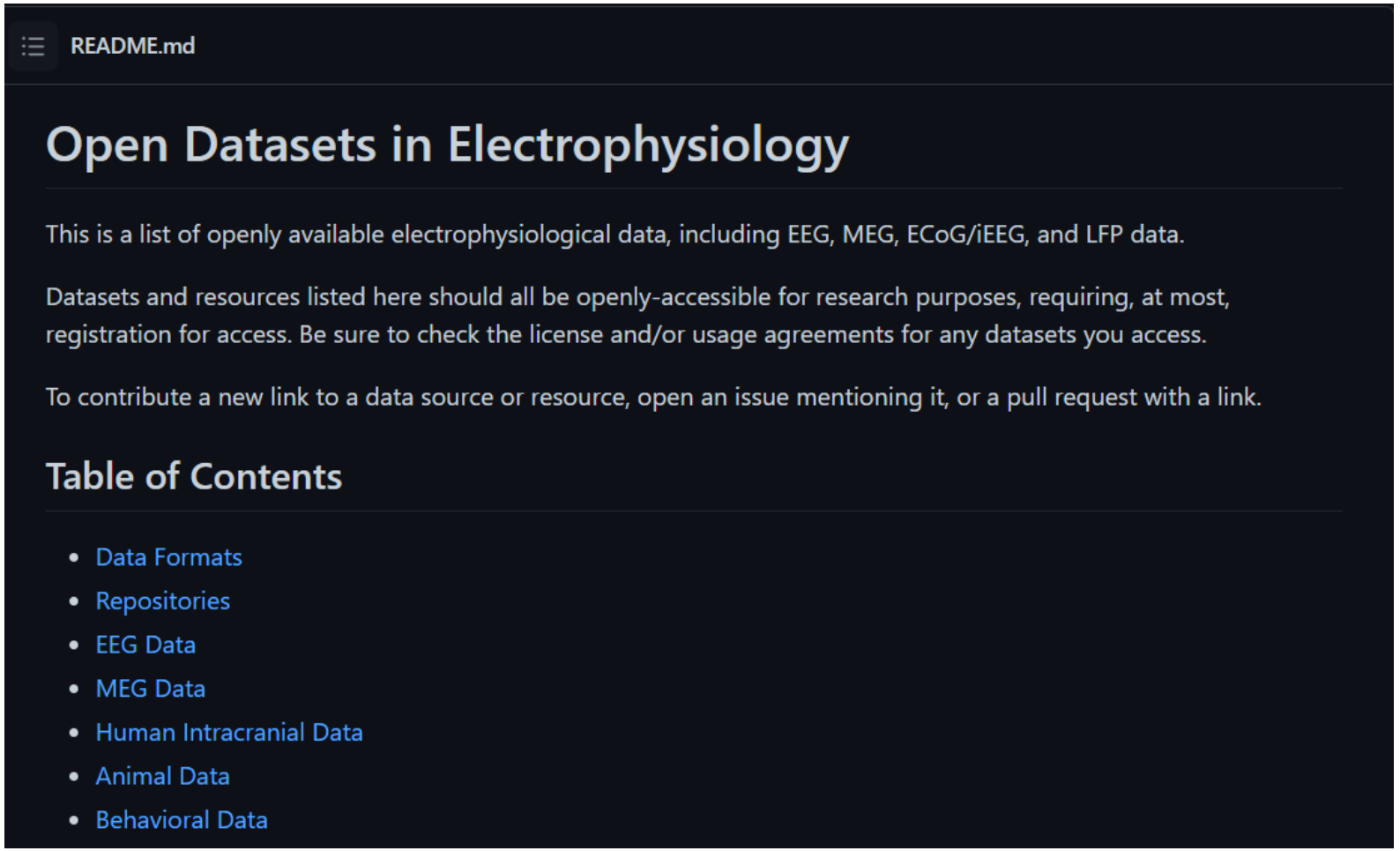
# Objectives

This introduction may help you to:

- understand how other multi-modal EEG studies perform data analysis
- analyze open datasets
- design EEG experiments using **MultiMoCo**

# Open Datasets in Electrophysiology

- https://github.com/openlists/ElectrophysiologyData


README.md

## Open Datasets in Electrophysiology

This is a list of openly available electrophysiological data, including EEG, MEG, ECoG/iEEG, and LFP data.

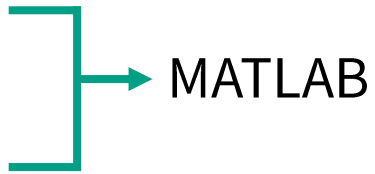
Datasets and resources listed here should all be openly-accessible for research purposes, requiring, at most, registration for access. Be sure to check the license and/or usage agreements for any datasets you access.

To contribute a new link to a data source or resource, open an issue mentioning it, or a pull request with a link.

### Table of Contents

# Packages for EEG analysis

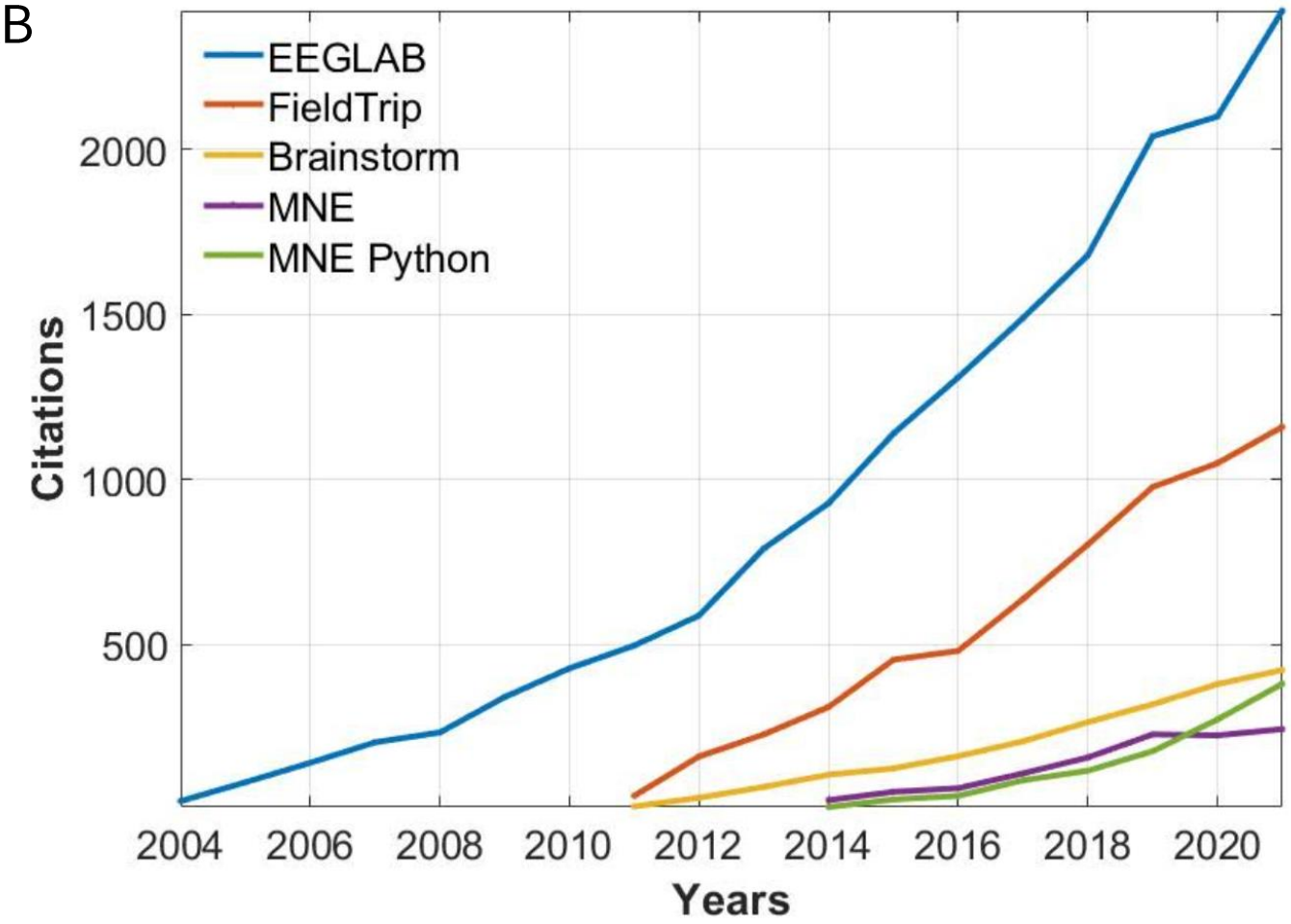
- EEGLAB (& ERPLAB)
- Fieldtrip
- MNE-Python



# Packages for EEG analysis

- EEGLAB (& ERPLAB)
- Fieldtrip
- MNE-Python

→ MATLAB

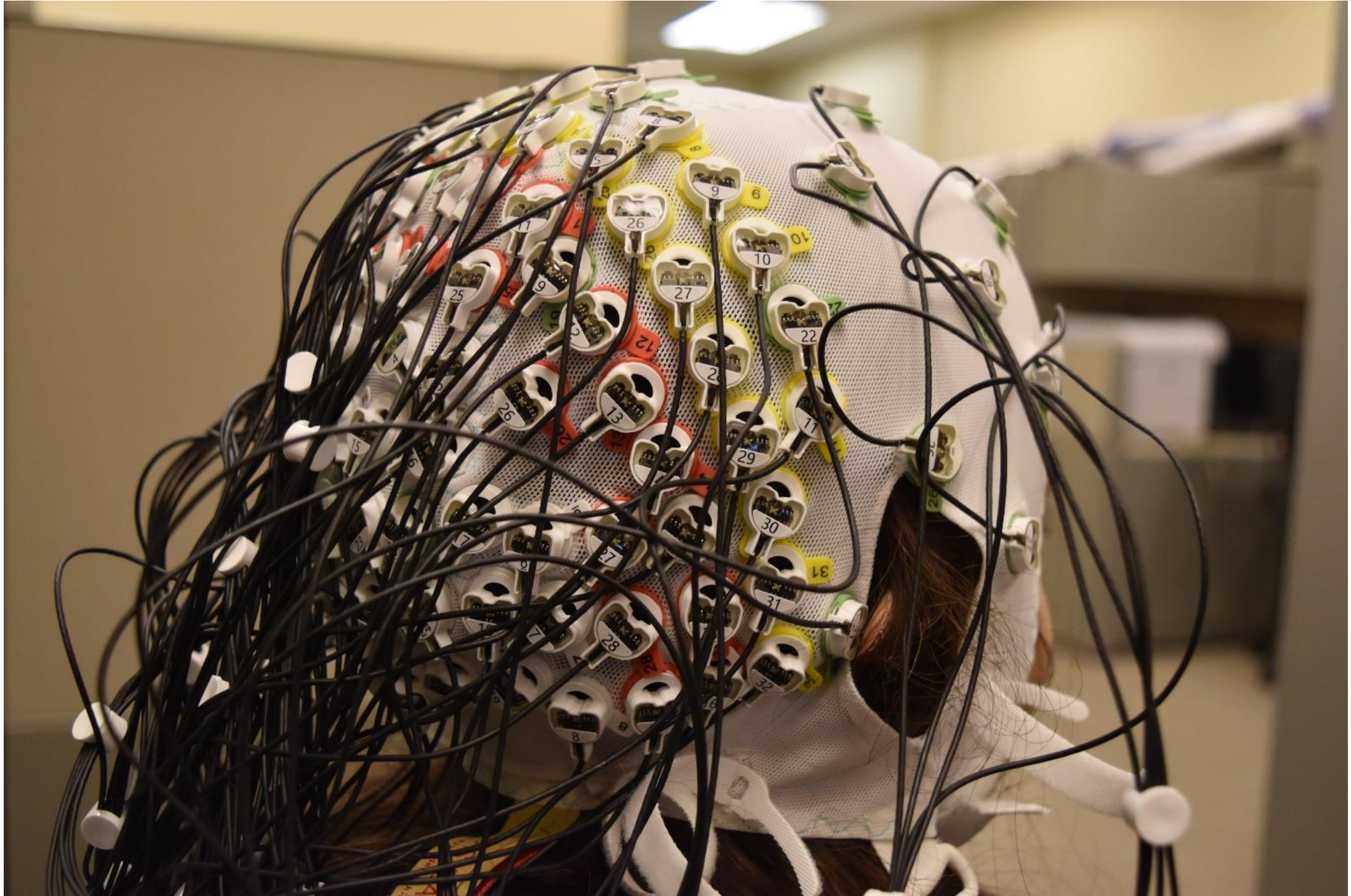



https://eeglab.org/others/EEGLAB_and_python.html

# EEG & ERP

- EEG = Electroencephalogram

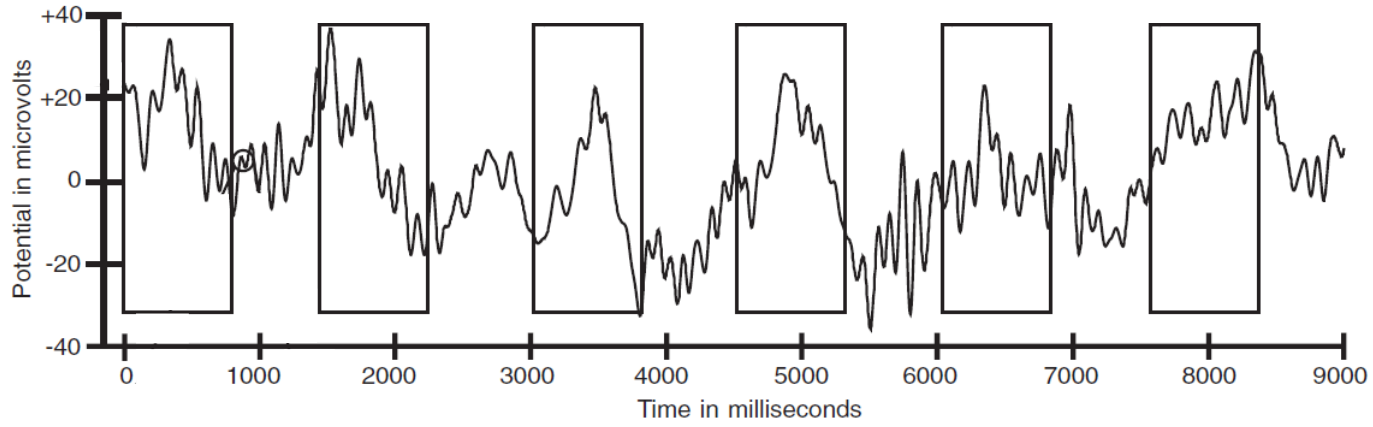

- A method to record electrical activity in the brain
- EEG signal is recorded by placing electrodes on the scalp
- Non-invasive



# EEG & ERP

- ERP = event-related potential (事件相關電位)

- **How do we get ERP?**
  - Cut the continuous EEG signals into segments

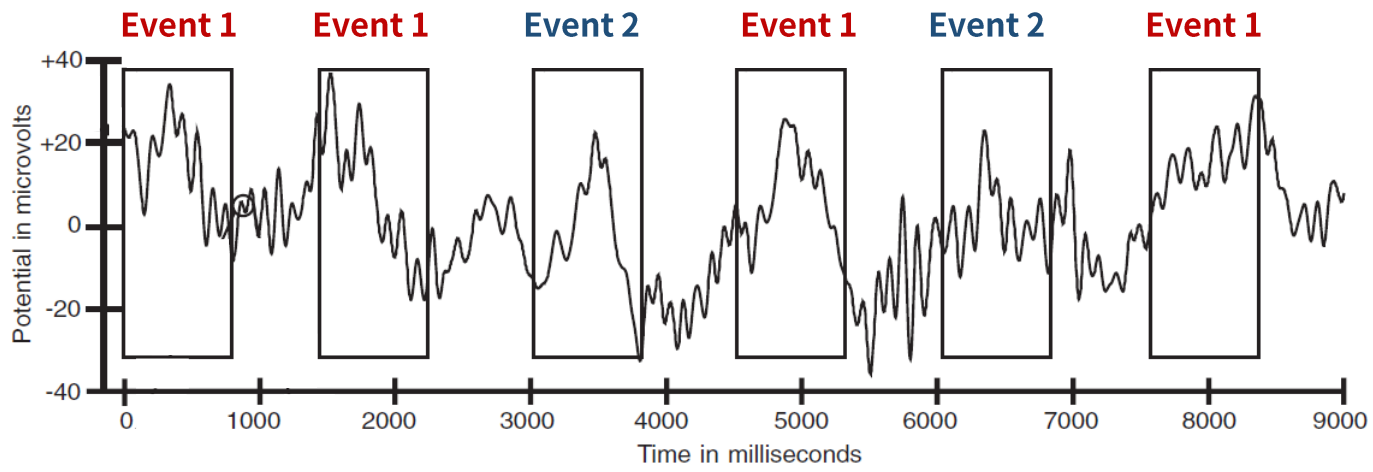



Adapted from Luck (2014)

# EEG & ERP

- ERP = event-related potential (事件相關電位)

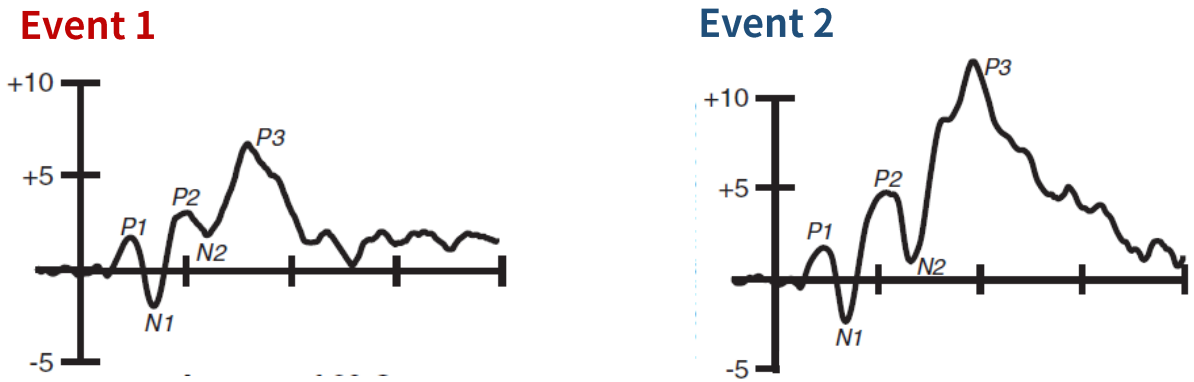
- **How do we get ERP?**
  - Cut the continuous EEG signals into segments
  - Average these EEG segments based on the "events" (stimulus type)



# EEG & ERP

- ERP = event-related potential (事件相關電位)

- **How do we get ERP?**
  - Cut the continuous EEG signals into segments
  - Average these EEG segments based on the "events" (stimulus type)
  - Then, we will get the signals that are **related to that specific event**



# EEG & ERP

- ERP = event-related potential (事件相關電位)

- **How do we get ERP?**
  - Cut the continuous EEG signals into segments
  - Average these EEG segments based on the "events" (stimulus type)
  - Then, we will get the signals that are **related to that specific event**

  → That's why it's called “event-related potentials”

# EEG & ERP

- ERP = event-related potential (事件相關電位)

- **How do we get ERP?**
  - Cut the continuous EEG signals into segments
  - Average these EEG segments based on the "events" (stimulus type)
  - Then, we will get the signals that are **related to that specific event**

  → That's why it's called “event-related potentials”
- **But why?**
  - Noise is random, so it will **cancel out each other** after averaging
  - Signals that are elicited by that event will **remain** even after averaging

# Tutorials

- Introduction to ERPs (by Steven Luck)
- ERP online courses (by Steven Luck)
- An introduction to the event-related potential technique (again, by Steven Luck)

# Preprocessing – outline

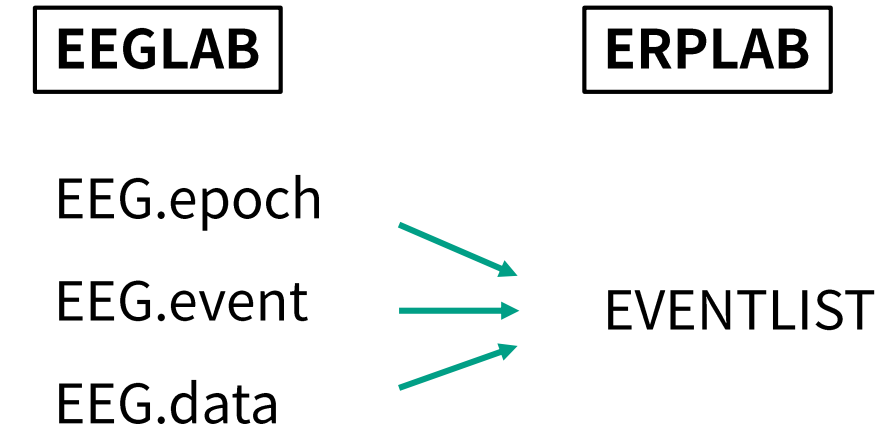
- Load dataset
- Channel location
- Create eventlist
- Assign bin
- Re-reference
- Epoch & baseline correction
- Artifact detection (AD)
- Filter

**→ Compute averaged ERP → Grand Average**

# Preprocessing – outline

- Load dataset
- Channel location
- Create eventlist
- Assign bin
- Re-reference
- Epoch & baseline correction
- Artifact detection (AD)
- Filter

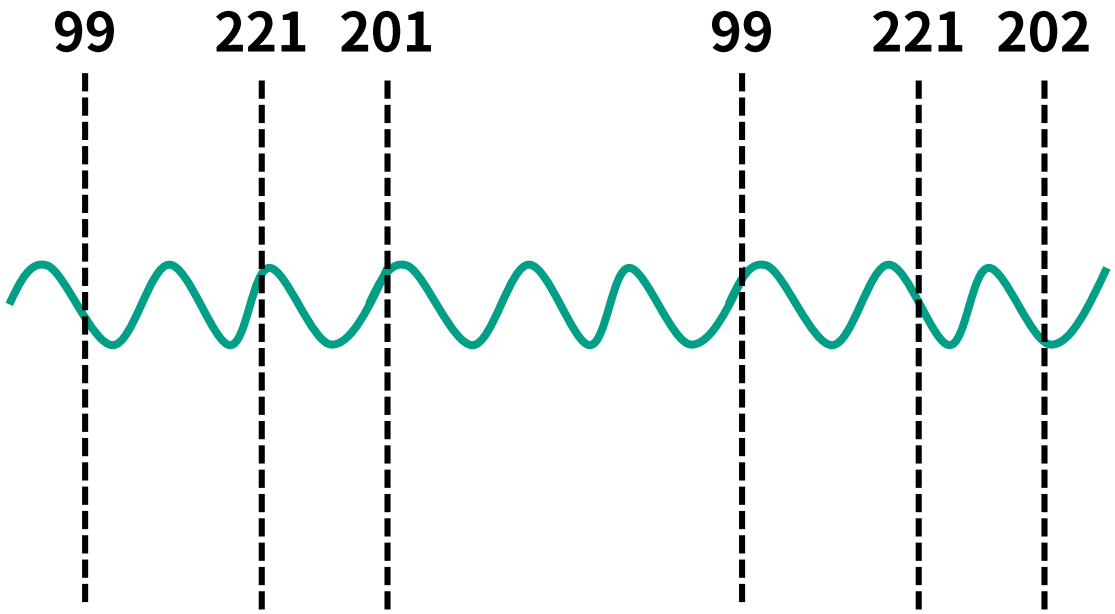
**→ Compute averaged ERP → Grand Average**



# Preprocessing – outline

- Load dataset
- Channel location
- Create eventlist
- Assign bin
- Re-reference
- Epoch & baseline correction
- Artifact detection (AD)
- Filter

**→ Compute averaged ERP → Grand Average**



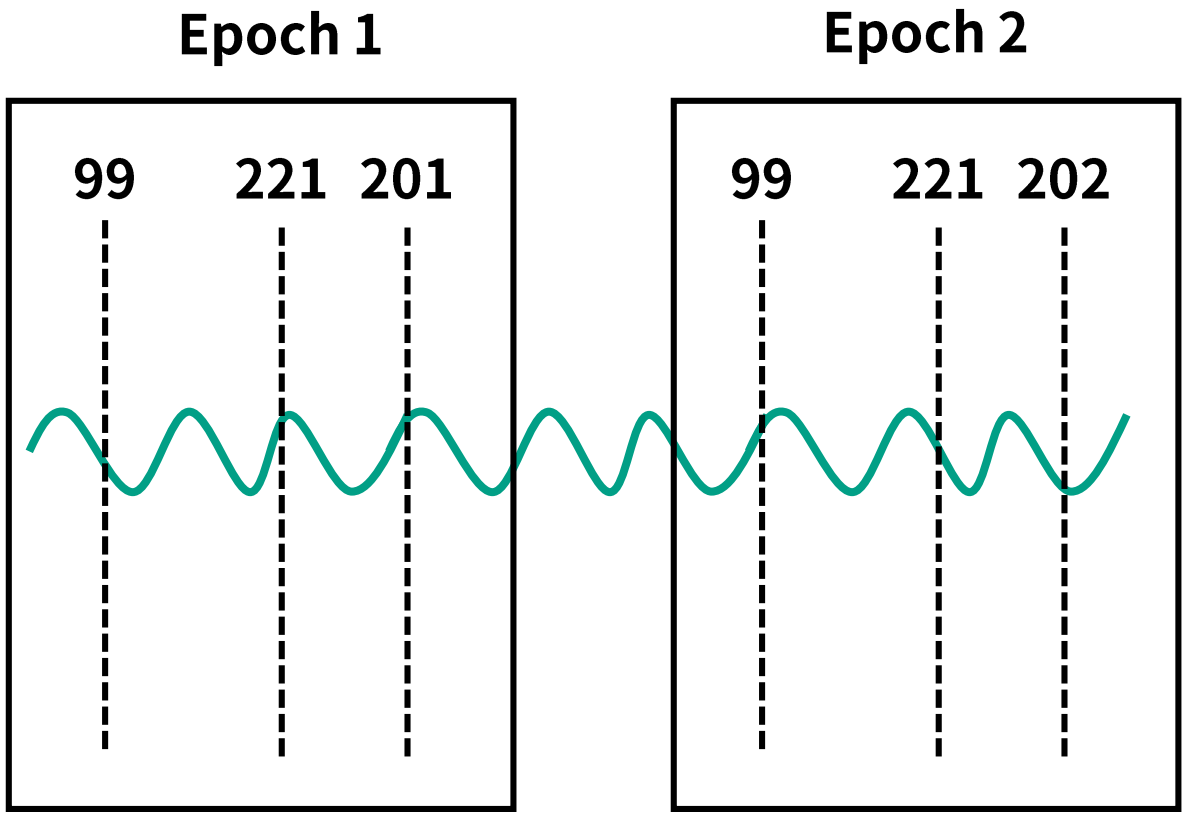
# Preprocessing – outline

- Load dataset
- Channel location
- Create eventlist
- Assign bin
- Re-reference
- Epoch & baseline correction
- Artifact detection (AD)
- Filter

**→ Compute averaged ERP → Grand Average**

# Preprocessing – outline

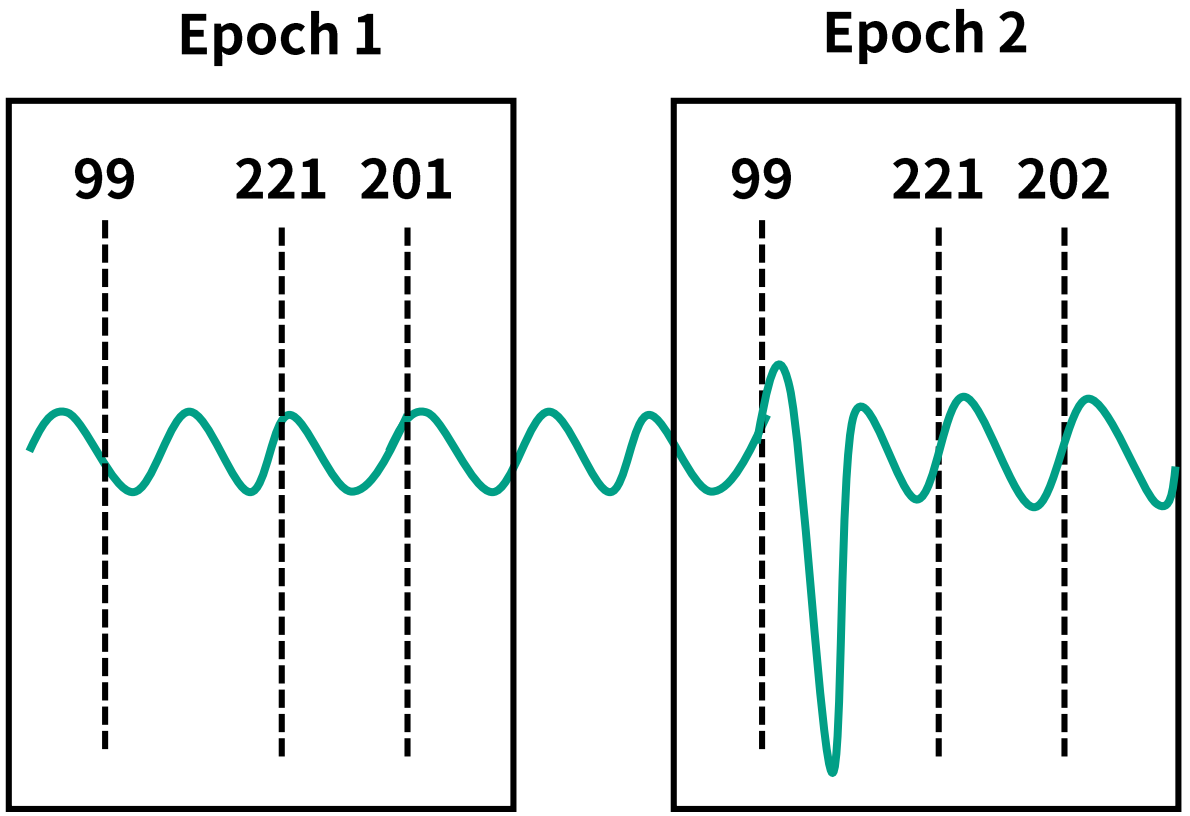
- Load dataset
- Channel location
- Create eventlist
- Assign bin
- Re-reference
- Epoch & baseline correction
- Artifact detection (AD)
- Filter




**→ Compute averaged ERP → Grand Average**

# Preprocessing – outline

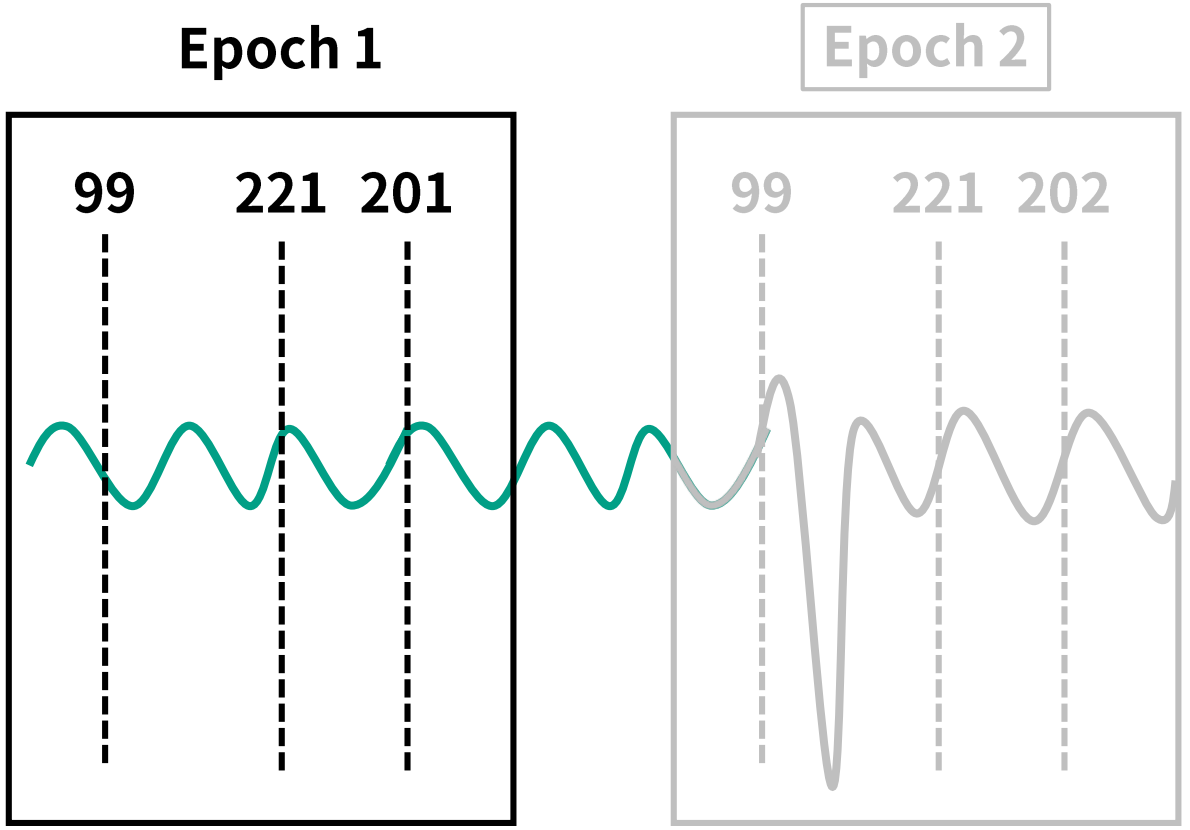
- Load dataset
- Channel location
- Create eventlist
- Assign bin
- Re-reference
- Epoch & baseline correction
- Artifact detection (AD)
- Filter




**→ Compute averaged ERP → Grand Average**

# Preprocessing – outline

- Load dataset
- Channel location
- Create eventlist
- Assign bin
- Re-reference
- Epoch & baseline correction
- Artifact detection (AD)
- Filter




**→ Compute averaged ERP → Grand Average**

C:\Users\YCK\Desktop\20221114 NTU MultiMoco Workshop\1_EEG data structure\Preprocessing_template.m

EDITOR PUBLISH VIEW

New Open Save Find Files Compare Print — FILE
Go To Find — NAVIGATE
Insert Comment Indent — EDIT
Breakpoints — BREAKPOINTS
Run Run and Advance Run Section Advance Run and Time — RUN

```
1  clear; clc;
2
3  %Launch EEGLAB
4  [ALLEEG EEG CURRENTSET ALLCOM] = eeglab;
5
6  %Location of the main study directory
7  %This method of specifying the study directory only works if you run the script; for running individual lines of code, replace the study directory with the path on your computer, e.g.:
8  DIR = fileparts(mfilename('fullpath'))
9
10 %Set parameters
11 filename = 'S1';
12 BDF = 'BDF.txt';
13 Rereference = 'Rereference_list.txt';
14 ChanLocation = 'standard_1005.elc';
15
16 %Load your dataset
17 EEG = pop_loadset('filename', [filename '.set'], 'filepath', DIR); %%
18
19 %Create event list from the dataset
20 EEG  = pop_creabasiceventlist(EEG , 'AlphanumericCleaning', 'on', 'Eventlist', [DIR '.txt'], 'Newboundary', { -99 }, 'Stringboundary', { 'boundary' }, 'Warning', 'on' ); %%
21
22 %Load your channel location file
23 EEG = pop_chanedit(EEG, 'lookup', ChanLocation); %%
24
25 %Assign bins
26 EEG  = pop_binlister(EEG , 'BDF', BDF, 'ExportEL', [DIR '.txt'], 'ImportEL', 'no', 'Saveas', 'on', 'SendEL2', 'EEG&Text', 'Warning', 'on' ); %%
27
28 %Channel operations for offline rereference
29 EEG = pop_eegchanoperator(EEG, Rereference);
30
31 %Epoching
32 EEG = pop_epochbin(EEG , [-200.0  1200.0],  'pre');
```

UTF-8 | script | Ln 12 | Col 17

# Scripts

- Open folder “**1_EEG data structure**”
  - Preprocessing_teamplate.m
  - S1.set
  - S1.fdt
  - BDF.txt
  - Rereference_list.txt
  - standard_1005.elc

# Scripts

- Open folder “**1_EEG data structure**”
  - Preprocessing_teamplate.m
  - S1.set
  - S1.fdt
  - BDF.txt
  - Rereference_list.txt
  - standard_1005.elc

# Scripts

- Open folder “**1_EEG data structure**”
  - Preprocessing_teamplate.m
  - S1.set → This one will be loaded
  - S1.fdt → But this one need to be placed in the same folder as the “.set” one
  - BDF.txt
  - Rereference_list.txt
  - standard_1005.elc

# Scripts

- Open folder “**1_EEG data structure**”
  - Preprocessing_teamplate.m
  - S1.set
  - S1.fdt
  - BDF.txt
  - Rereference_list.txt
  - standard_1005.elc

# EEGLAB data structure

**EEG** - the current EEG dataset

**ALLEEG** - array of all loaded EEG datasets

**CURRENTSET** - the index of the current dataset (EEG = ALLEEG(CURRENTSET))

**LASTCOM** - the last command issued from the EEGLAB menu

**ALLCOM** - all the commands issued from the EEGLAB menu

# EEGLAB data structure

- **EEG** contains all the information of currently loaded dataset in EEGLAB
  - Load **S1.set** and type **EEG**
  - Load **S1_preprocessed.set** and type **EEG**
  - Examine the structure of each dataset

  - What's the difference?
    - trials
    - pnts
    - data
    - epoch

# EEGLAB data structure

- **EEG** contains all the information of currently loaded dataset in EEGLAB
  - Type **EEG.trials**
  - Type **EEG.pnts**
  - Type **EEG.epoch**
  - Type **EEG.epoch(1)**
  - Type **EEG.epoch(1).eventtype**
  - Type **EEG.epoch(1).eventlatency**

  - What about typing **EEG.data**?

    → Don’t try!

# EEGLAB data structure

- **ALLEEG** holds all the loaded datasets in the EEGLAB
  - Type **ALLEEG(1)** and **ALLEEG(2)**

# Preprocessing – outline

- Load dataset
- Channel location
- Create eventlist
- Assign bin
- Re-reference
- Epoch & baseline correction
- Artifact detection (AD)
- Filter

**→ Compute averaged ERP → Grand Average**

# Preprocessing – outline

- Load dataset
- Channel location
- Create eventlist
- Assign bin
- Re-reference
- Epoch & baseline correction
- Artifact detection (AD)
- Filter

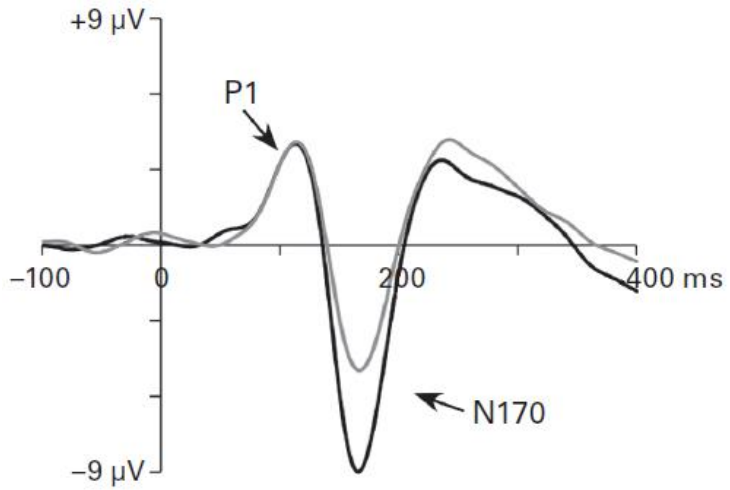
→ **Compute averaged ERP →** **Grand Average**

# ERP CORE – N170 dataset

**N170**

- Face-related component
- Typically peaks around 170 ms after stimulus onset
- Larger when the stimulus is a face compared to when the stimulus is a non-face object

→ The human brain is able to distinguish between **faces** and **other objects within 150 ms!**



# ERP CORE – N170 dataset

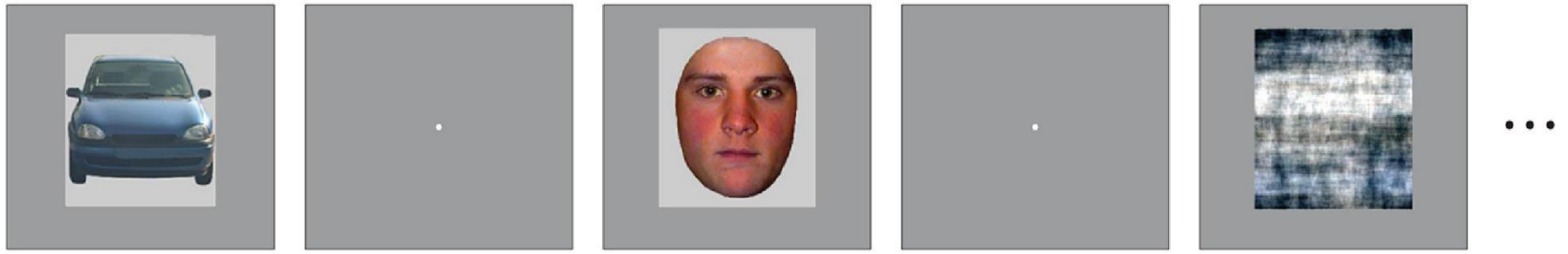
Previous studies have used N170 as **an index of face processing** to ask interesting questions

- Face processing is at least partially **automatic**
- Face processing can still be modulated by attention

→ Will the appearances of gestures reduce the N170 amplitudes?

# ERP CORE – N170 dataset

- Face perception task
- An image of a face/car/scrambled face/scrambled car was presented
- Participants responded whether the stimulus was an “object” (face or car) or a “texture” (scrambled face or scrambled car)



# Scripts

- Open folder “**2_ERP-core_N170**”
  - Script10_Plot_Grand_Average_ERPs.m
  - Script11_Plot_Grand_Average_Topomaps.m
  - GA_N170_erp_ar_diff_waves.erp
  - GA_N170_erp_ar_diff_waves_lpfilt.erp

# Scripts

- Open folder “**2_ERP-core_N170**”
  - Script10_Plot_Grand_Average_ERPs.m
  - Script11_Plot_Grand_Average_Topomaps.m
  - GA_N170_erp_ar_diff_waves.erp
  - GA_N170_erp_ar_diff_waves_lpfilt.erp

# Scripts

- Open folder “**2_ERP-core_N170**”
  - Script10_Plot_Grand_Average_ERPs.m
  - Script11_Plot_Grand_Average_Topomaps.m
  - GA_N170_erp_ar_diff_waves.erp
  - GA_N170_erp_ar_diff_waves_lpfilt.erp