Classifying brain activity using electroencephalography and automated time tracking of computer use

Master's thesis presentation

Erik Bjäreholt

Department of Computer Science
Faculty of Engineering
Lund University

October 28, 2021

LUND UNIVERSITY

Abstract

We investigate the ability of EEG to distinguish between different activities users engage in on their devices, building on previous research which showed a considerable difference in brain activity between code- and prose-comprehension, as well as differences during code- and prose-synthesis. We perform a replication study and improve upon past results using state-of-the-art machine learning classifiers based on Riemannian geometry.

Furthermore, we extend the scope of previous work by introducing the automated time tracking application ActivityWatch, to track the device activities that the user is engaging in. This lets us label EEG data with naturalistic device activity, which we then use to train classifiers to discern activities such as code writing vs prose writing, or work vs media consumption. Our results indicate that a consumer-grade EEG device can discern between different activities that a user performs at the computer. Among other results, we show that not only can code and prose *comprehension* be distinguished, but also code and prose *writing*.

A full replication package, including source code and a sample dataset, is available at

github.com/ErikBjare/thesis

LUND UNIVERSITY

Outline

LUND UNIVERSITY

Introduction

Before we begin, we will present two technologies used:

- Functional brain imaging
- Automated time trackers

Introduction

Functional brain imaging

Functional brain imaging is used to measure aspects of brain function.

Examples:

- Electroencephalography (EEG)
- Magnetoencephalography (MEG)
- Functional Magnetic Resonance Imaging (fMRI)
- Functional Near-Infrared Spectroscopy (fNIRS)

Introduction

Functional brain imaging > EEG

Developments in EEG the last ∼10 years:

- Cost reduction
- Consumer availability

Rough timeline:

- 2013: OpenBCI kickstarter
- 2016: InteraXon releases the Muse
- 2021: Neurosity releases the Crown

Introduction

Functional brain imaging > EEG

Applications:

- Clinical (sleep, epilepsy)
- Brain-Computer Interfaces
- Neurolinguistics research
  - Discerning code vs prose comprehension
    - Using MRI, by Floyd et al. [1]
    - Using EEG & various biosensors, by Fucci et al. [2]
- Biofeedback / meditation aid
- Quantified self (measuring mood, focus)

Introduction

Automated time trackers

We use an automated time tracker to track which device activity a user is engaging in.

Examples:

- Screen Time (Apple)
- Digital Wellness (Android)
- RescueTime (commercial use)
- TimeAware (research use)

Introduction

Automated time trackers

Issues with existing solutions:

- Data detail & temporal resolution
- Source availability / licensing
- Privacy concerns

Introduction

Automated time trackers

Our solution:

ActivityWatch

- *“The world’s best free & open-source automated time tracker”*
- Started in 2015 by me
- My brother joined in 2016
- >100,000 downloads
- >90 contributors
- Available on Windows, macOS, Linux, and Android.

Theory

Now a very brief introduction to underlying theory within electroencephalography and machine learning.

Theory

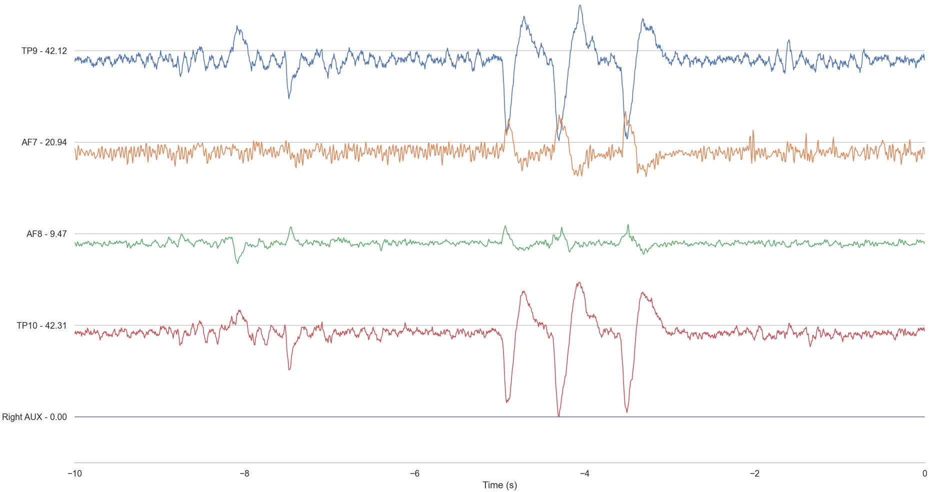
Electroencephalography

Electroencephalography works by measuring tiny amounts of electrical potential (voltage) on the skull, which is caused by the activation of underlying neurons.

Measurements are taken about 256 times per second, using one or more electrodes.

Theory

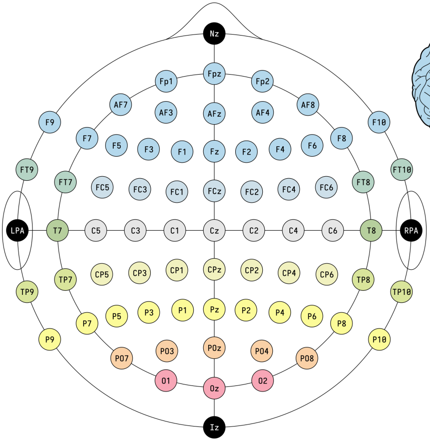
Electroencephalography

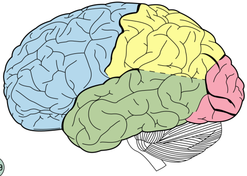


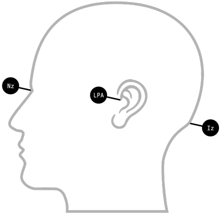
Theory

Electroencephalography

The 10–20 system is a standard for electrode placements.







Theory

Electroencephalography

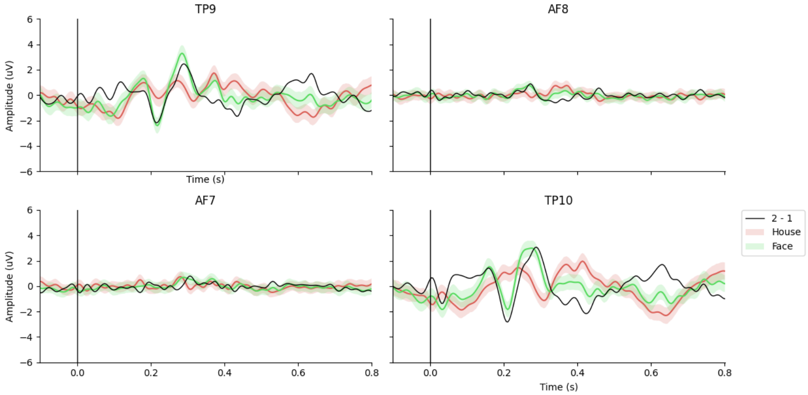
ERPs, of Event-Related Potentials, are stereotyped responses to a stimulus.

ERP	Elicited by
N170	Processing of faces, familiar objects or words.
N400	Words and other meaningful stimuli.
P300	Decision making, oddball paradigm.
P600	Hearing or reading grammatical errors and other syntactic anomalies.

Theory

Electroencephalography

Example analysis of the N170 ERP:



The signal can be broken down into constituent frequencies. They can be roughly grouped into frequency bands, which are associated with certain brain states.

Frequency band	Frequencies	Brain states
Gamma (γ)	>35 Hz	Concentration
Beta (β)	16–35 Hz	Active, external attention, relaxed
Sigma (σ)	12–16 Hz	Sleep spindles
Alpha (α)	8–12 Hz	Very relaxed, passive attention
Theta (θ)	4–8 Hz	Deeply relaxed
Delta (δ)	0.5–4 Hz	Sleep

Theory

Machine learning

Machine learning can be used to classify EEG signals.

Common approaches:

- Riemannian methods
- Deep learning
- Common Spatial Pattern
- Bandpower-features

Theory

Machine learning > Riemannian geometry

Riemannian methods in EEG utilizes the spatial information encoded in covariance matrices to estimate the similarity between two signals.

In the simple *Minimum Distance to Mean* (MDM) method, covariance matrices for each class are averaged in Riemannian space. For a new signal's matrix, the distance to each class is calculated, and whichever class distance is smaller becomes the predicted class.

Theory

Machine learning > Riemannian geometry

The Riemannian distance metric δ_G for two symmetric positive definite matrices A and B (such as covariance matrices) is [3]:

$$\delta_G(A, B) = \sqrt{\sum_{i=1}^{N} \log^2 \lambda_i(A, B)}$$

Theory

Machine learning > Riemannian geometry

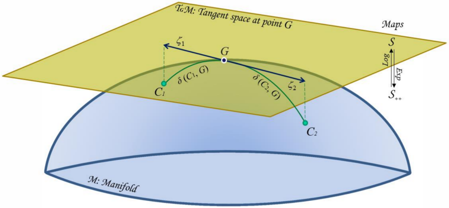


Figure: Schematic representation of the symmetric positive definite matrix manifold, the geometric mean G of two points and the tangent space at G. The geometric mean of these points is the midpoint on the geodesic connecting C_1 and C_2, i.e. it minimizes the sum of the two squared distances. The map from the tangent space to the manifold is an exponential map. The inverse map is a logarithmic map.

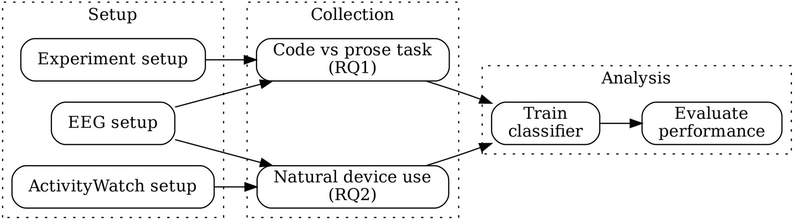
Source: *Congedo et al. [4]*

LUND UNIVERSITY

Method

We perform two different experiments:

1. Controlled code vs prose experiment
2. Naturalistic device use

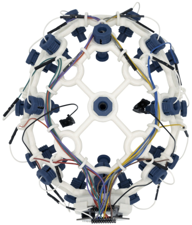


Method

Devices

Muse S

Neurosity Crown



OpenBCI Cyton + Ultracortex

Manufacturer	Device	Channels	Sampling rate	Comfort
InteraXon	Muse S (2020)	4	256Hz	High
Neurosity	Crown (2021)	8	256Hz	Medium
OpenBCI	Cyton (2013) + Ultracortex	8–16	125–250Hz	Low

LUND UNIVERSITY

Method

Collection

Next up: Collecting data for our code vs prose experiments, followed by naturalistic use experiments.

Method

Collection > Code vs prose

```
if (l < 0) {
    l = (Z_STRLEN_P(orig_str) - f) + l;
    if (l < 0) {
        l = 0;
    }
}


Given the following values for variables, the
value of l after executing this code will be 0.
-------------------------------------------------
l = -2
Z_STRLEN_P(orig_str) = 10
f = 9
```

(a) Code comprehension

Knuth offered a ~~proved~~ proven algorithm for simulating an elevator system. This algorithm is different ~~than~~ from previous attempts in several ways. ~~This algorithm should be used~~Use this algorithm to model scenarios with more than one elevator.

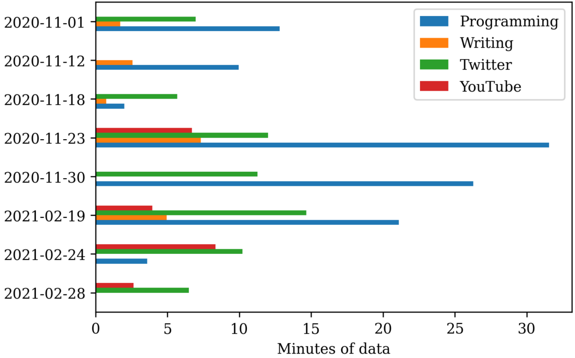
(b) Prose review

Figure: Sample of the tasks used as stimuli.

Method

Collection > Naturalistic

A single subject (me) measured EEG while engaging in natural device use (both work and leisure). We define 4 categories of device activity.



Method

Analysis

We train two classifiers:

- Riemannian geometry
- Bandpower-features (benchmark)

General software libraries used: scikit-learn, numpy, pandas.

Domain-specific libraries used: pyriemann, MNE, yasa.

Method

Analysis > The classifiers

Our Riemannian classifier pipeline is constructed like this:

```
from sklearn.pipeline import make_pipeline
from sklearn.linear_model import LogisticRegression
from pyriemann.estimation import Covariances
from pyriemann.spatialfilters import CSP
from pyriemann.tangentspace import TangentSpace

clf = make_pipeline(
    Covariances(),
    CSP(4, log=False),
    TangentSpace(),
    LogisticRegression(),
)
```

LUND UNIVERSITY

Method

Analysis > The classifiers

Our bandpower-based classifier computes the bandpower of each frequency band, and puts the values and their ratios in a feature vector.

We then use common ML methods for the actual learning and classification.

Method

Analysis > Windows and epochs

To train on and classify the EEG signal, we first need to label it and split into fixed-size windows.

We divide the EEG-signal into epochs (according to their stimuli markers), and then split those down into 5s windows which we train on.

We then also aggregate the predictions back into their epochs by taking the mean prediction of each window in the epoch, yielding predictions for entire epochs.

Method

Analysis > Performance evaluation

To evaluate our classifiers, we need a suitable performance metric and cross-validation method.

Method

Analysis > Performance evaluation

- Studies using EEG often use *balanced accuracy* (BAC).
- Balanced accuracy deals with imbalanced datasets.

For binary classification, BAC is defined as:

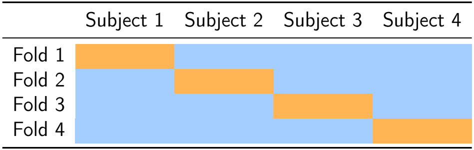
$$BAC = \frac{Sensitivity + Specificity}{2} = \frac{\frac{TP}{TP+FN} + \frac{TN}{TN+FP}}{2}$$

This implies that, for the binary case, $BAC = 0.5$ is no better than chance.

Method

Analysis > Performance evaluation > Validation

To ensure our classifier generalizes across subjects, we perform *Leave-One-Run-Out* (LORO) cross validation.



For our naturalistic device use experiment we instead use Stratified K-Fold cross-validation, as there is only one subject.

Method

Comparison with previous studies

Setting	**Study**		
	This study	**Fucci et al.** (2019)	**Floyd et al.** (2017)
Experiment site	Lund Univ. (Sweden)	Univ. of Bari (Italy)	Univ. of Virginia (USA)
# Participants	10	28	29
Participants experience	Grads	Undergrads	Grads & Undergrads
# Tasks	Variable	36 tasks	27 tasks
Task type	Code comprehension Prose review	Code comprehension Prose comprehension	Code comprehension Code review Prose review
Physiological signal	Neural	Neural Skin Heart	Neural
Physiological measure	EEG	EEG EDA HR, HRV, BVP	BOLD
Device	Muse S	BrainLink Headset Empatica wristband	fMRI
Classifier	Riemannian geometry	8 algorithms	Gaussian Process
Classifier validation	LORO-CV	LORO-CV Hold-out	LORO-CV
Classifier metric	Balanced accuracy (BAC)	Balanced accuracy (BAC)	Balanced accuracy (BAC)

Results

- Controlled code vs prose experiment
- Naturalistic device use

Results

Controlled code vs prose experiment

Our results are:

Subject	Riemannian		Bandpower	
	Window-level	Epoch-level	Window-level	Epoch-level
#0	0.673	0.727	0.511	0.541
#1	0.895	0.955	0.689	0.809
#5	0.616	0.542	0.628	0.750
#6	0.864	0.908	0.739	0.737
#7	0.749	0.900	0.733	0.733
Median	0.749	0.900	0.689	0.737

Table: The balanced accuracy for each LORO fold/subject. Excluding subjects 3, 4, 8, 9, and 10.

Results

Controlled code vs prose experiment

Compared to previous studies:

	This study		Fucci et al.	Floyd et al.
	Riemannian	Bandpower		
Overall	0.75	0.69	0.66	0.79

Table: Result comparison between the previous studies and this study. Best balanced accuracy scores are reported. For this study, we used the best window-level score. For Fucci et al. we chose the best EEG-only score.

LUND UNIVERSITY

Results

Naturalistic device use

Our naturalistic device use results:

Experiment	Score	Support	Hours
Programming vs Writing	0.676	(1386, 209)	2.22h
Programming vs Twitter	0.695	(1386, 949)	3.24h
Programming vs YouTube	0.672	(1386, 266)	2.29h
Twitter vs Writing	0.833	(949, 209)	1.61h
Twitter vs YouTube	0.604	(949, 266)	1.69h
YouTube vs Writing	0.889	(266, 209)	0.66h

LUND UNIVERSITY

Conclusions

We conclude that we can discern code from prose using EEG, in both settings.

We also find that...

- Using a Riemannian approach outperforms the use of bandpower-features.
- It seems easier to discern work from leisure, than inter-work or inter-leisure tasks.

LUND UNIVERSITY

Conclusions

Future work

- Collect more data
  - NeuroTech Challenge
  - More subjects for naturalistic use
- Implement classification in moabb
- Create prose comprehension stimuli in English
- Use even better EEG devices, or even try fNIRS
- Turn brainwatch into a proper app to complement ActivityWatch

Discussion

Time permitting, we will briefly go over threats and ethical considerations.

Discussion

Threats to validity

- Dataset
  - Size
  - Cherry-picked subjects
- Stimuli
  - Prose *review* (not *comprehension*) stimuli

Discussion

Ethical considerations

- Research ethics
- Commercial ethics

Acknowledgements

- My advisor Markus Borg.
- My brother Johan Bjäreholt and all the ActivityWatch contributors, for working with me all these years.
- The NeuroTechX crowd, specifically John Griffiths and Morgan Hough, for their support and time spent helping me.
- Pex Tufvesson and Carolina Bergeling at the Department for Automatic Control, for providing early guidance.
- Andrew Jay Keller at Neurosity, for gifting me a refurbished Notion DK1 to work with.
- Alex K. Chen, for referring me to all the right people.
- All the test subjects, for their time and interest.
- Everyone who has contributed to the open source tools I have used.
- Everyone who have supported me at LTH.
- Friends and family, for their neverending love and support.

LUND UNIVERSITY

Appendix I

Data overview

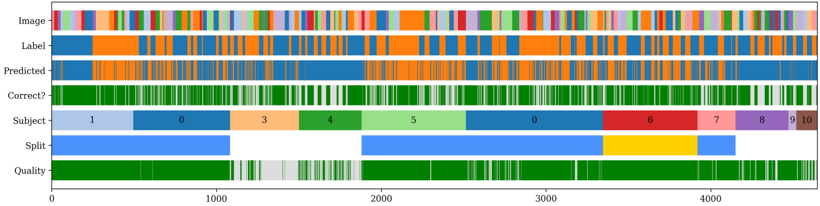


Figure: Visualization of the labeled data with classifications from one example subject-fold. Shows the *Image* (stimuli), the class *Label* for that stimuli (**blue** is code, **orange** is prose), the *Predicted* class, whether the prediction is *Correct*, the *Subject*, the *Split*/Fold (**blue** shows the training set, yellow the test set), and our threshold measure for signal *Quality* (**green** indicates acceptable quality). The x-axis is the window index, sorted by acquisition time.

It can be seen that (1) subjects #3 and #4 have bad signal quality, and have therefore been excluded from the training set. (2) The subjects #9 and #10 have also been excluded from training due to issues during data collection. (3) For subject #1 the stimuli images were not shuffled. (4) Subject #0 appears twice, as they did two sessions (using unseen stimuli).

References I

[1] Benjamin Floyd, Tyler Santander, and Westley Weimer. "Decoding the Representation of Code in the Brain: An fMRI Study of Code Review and Expertise". In: *2017 IEEE/ACM 39th International Conference on Software Engineering (ICSE)*. 2017 IEEE/ACM 39th International Conference on Software Engineering (ICSE). May 2017, pp. 175–186. DOI: 10.1109/ICSE.2017.24.

[2] Davide Fucci et al. "A Replication Study on Code Comprehension and Expertise Using Lightweight Biometric Sensors". In: *2019 IEEE/ACM 27th International Conference on Program Comprehension (ICPC)*. 2019 IEEE/ACM 27th International Conference on Program Comprehension (ICPC). May 2019, pp. 311–322. DOI: 10.1109/ICPC.2019.00050.

[3] Wolfgang Förstner and Boudewijn Moonen. "A Metric for Covariance Matrices". In: *Geodesy-The Challenge of the 3rd Millennium*. Ed. by Erik W. Grafarend, Friedrich W. Krumm, and Volker S. Schwarze. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 299–309. ISBN: 978-3-642-07733-3 978-3-662-05296-9. DOI: 10.1007/978-3-662-05296-9_31. URL: http://link.springer.com/10.1007/978-3-662-05296-9_31 (visited on 07/07/2021).

References II

[4] Marco Congedo, Alexandre Barachant, and Rajendra Bhatia. "Riemannian Geometry for EEG-Based Brain-Computer Interfaces; a Primer and a Review". In: *Brain-Computer Interfaces* 4.3 (July 3, 2017), pp. 155–174. ISSN: 2326-263X. DOI: 10.1080/2326263X.2017.1297192. URL: https://doi.org/10.1080/2326263X.2017.1297192 (visited on 10/11/2021).

LUND UNIVERSITY