Detecting eye closure from EEG signals using a Recurrent Neural Network

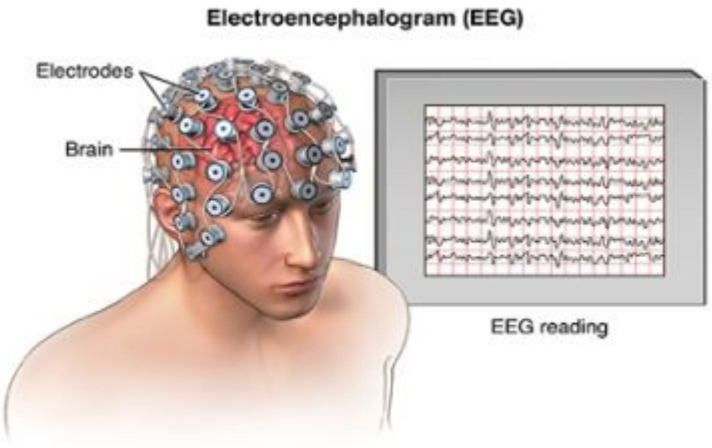
Presenters:
Ben Stanislav[1], Susmi Sharma[2], Dona Lontok[3], Priya Nalliah[4]
Cognitive Computational Neuroscience

Contributions:
1: Introduction and limitations.
2: Pre-processed the data for—converted the data into excel, assigned the correct labeling for the output of models, and averaged the EEG data across time. Worked on method section.
3: Data filtering, exploratory data analysis, visualization, model training and testing, evaluation of results, and worked on result slides.
4: Discussion implications and future directions.

Introduction

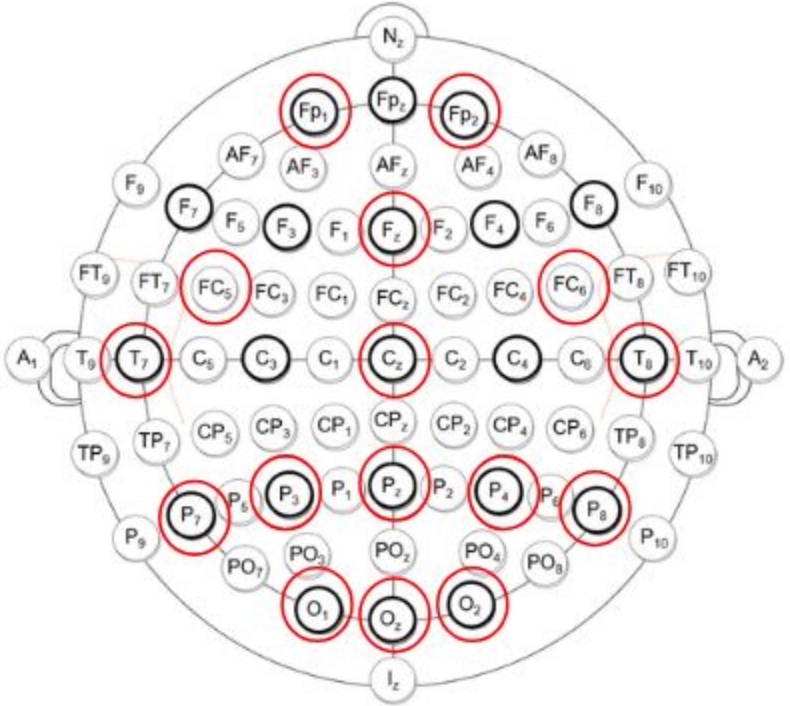
EEG Data

- EEG uses scalp electrodes to detect rhythmic alterations in the brain's electrical activity
- EEG has good temporal but limited spatial resolution
- EEG data is sequential data (relevant to use of Recurrent Neural Networks)



Our Data

- obtained from zenodo.org/records/2348892,
- EEG recordings were taken from an array of 16 scalp electrodes in 20 participants
- Each subject participated in 10 blocks (10 seconds each) of recordings
- The blocks alternated between eyes closed (condition 1) and eyes open (condition 2)

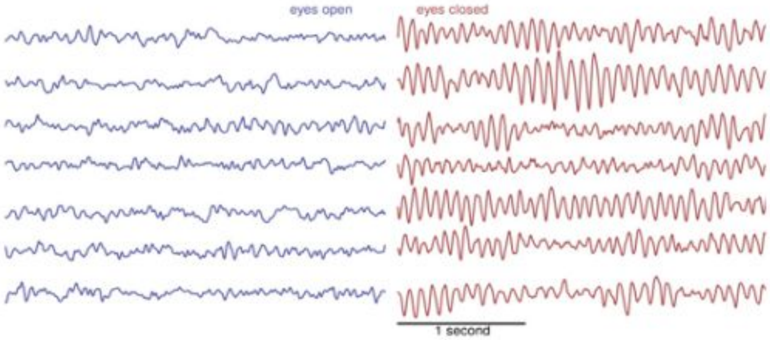


**Grégoire Cattan, Pedro L. C. Rodrigues, & Marco Congedo. (2018). EEG Alpha Waves dataset [Data set]. Zenodo. https://doi.org/10.5281/zenodo.2348892

Using machine learning, can we categorize when eyes are closed or open based on the brain’s electrical activity?

Past Approach

- Previous work with time frequency analysis looking at alpha and beta waves
- Open or closed eyes is indicative of different brain states
  - “The alpha rhythm is typically recorded in awake individuals with their eyes closed. By definition, the frequency of the alpha rhythm is 8 to 13 Hz, with amplitudes that are typically 10 to 50 mV. Lower-amplitude beta activity is defined by frequencies of 14 to 60 Hz and is indicative of mental activity and attention.” - Purves, Neuroscience, 6th ed. p. 647



Berger, H. (1929). Über das elektroenkephalogramm des menschen. Archiv für psychiatrie und nervenkrankheiten, 87(1), 527-570.
Adrian, E. D., & Matthews, B. H. (1934). The Berger rhythm: potential changes from the occipital lobes in man. Brain, 57(4), 355-385.
Othmani, A., Sabri, A. Q. M., Aslan, S., Chaieb, F., Rameh, H., Alfred, R., & Cohen, D. (2023). EEG-based neural networks approaches for fatigue and drowsiness detection: A survey. Neurocomputing, 126709.

Methods Overview

- 3 data processing approaches
  - Raw data
  - Averaged by block data
  - Top 5 electrode raw data
    - based on exploratory data analysis
- 2 machine learning algorithms
  - Multilayer perceptron
  - Recurrent Neural Network
- Questions
  - Which data preprocessing is best?
  - Which machine learning algorithm is best?

Multilayer Perceptron (MLP)

- Machine learning approach which can learn to categorize input using multiple hidden layers in a feed forward manner
- Used for when time component not preserved in data processing (data simplified to not be sequential) which allows for lower computational cost

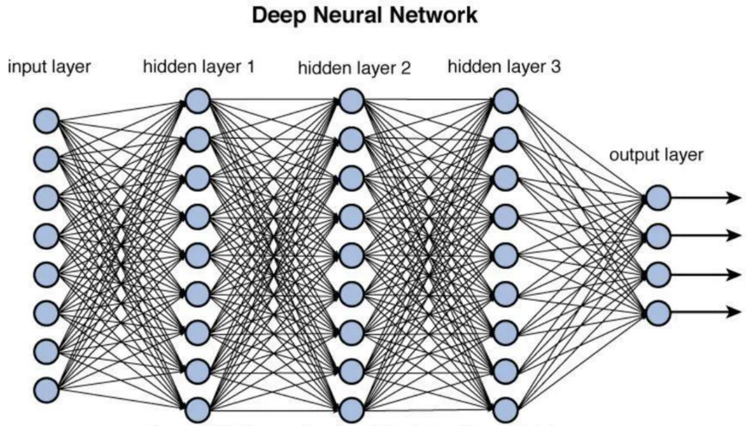
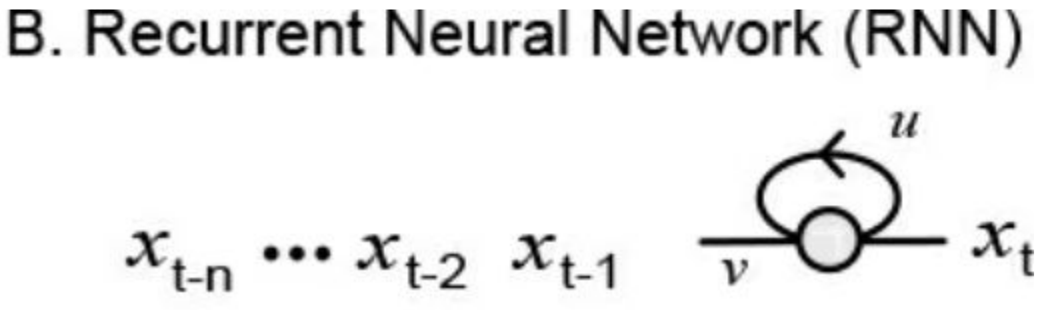


Figure 12.2 Deep network architecture with multiple layers.

Recurrent Neural Networks

- Best fit for sequential or time series data like that seen in EEG, where what comes next in the sequence depends on what came before
- RNNs capture this idea because they involve a memory component by which the input and output at a current time is influenced by the prior input in the sequence.

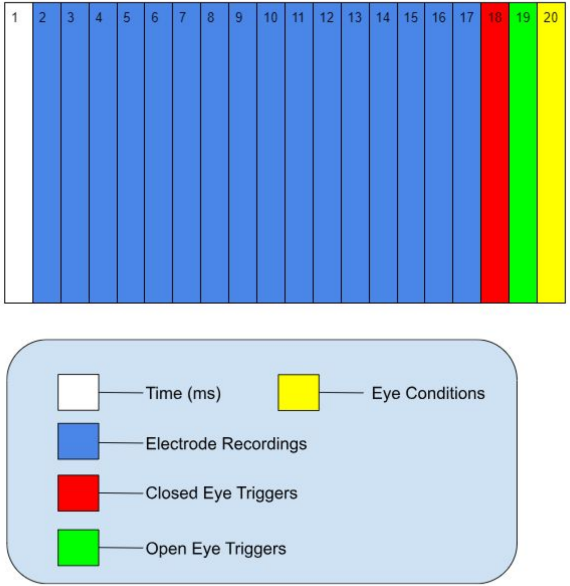


Hypothesis: The Recurrent Neural Network should do better than the Multilayer Perceptron at classifying when eyes are closed

Methods

EEG Data Pre-processing

- In the first stage of data preprocessing, the multidimensional array datasets stored in .mat format for each subject were converted into Excel format.
- To enhance usability and preliminary data analysis.
- The excel dataset included a total of 20 columns.
  - Added column 20 = eye condition (Closed (1), Open (0))
- Discarded the portion of the unlabeled data before the first block started for every participant.



**Grégoire Cattan, Pedro L. C. Rodrigues, & Marco Congedo. (2018). EEG Alpha Waves dataset [Data set]. Zenodo. https://doi.org/10.5281/zenodo.2348892

Two types of neural network models with 3 different datasets. (2*3)

- Two different types of Deep Learning Neural Network—Multiple Perceptron Network and Recurrent Neural Network with GRU layer.
- The EEG dataset were fed into the two neural networks 3 different ways. “Raw data” dataset preserved all the milliseconds electrode information; “Averaged across blocks” included data points from averaging across blocks; and finally, the last dataset comes from exploratory data analysis.

Multilayer Perceptron Model	**Recurrent Neural Network**
Raw Data	Raw Data
Averaged across blocks	Averaged across blocks
Input found useful after Exploratory Data Analysis	Input found useful after Exploratory Data Analysis

Dataset I. (Raw Data, 10 subjects combined)

We used all the *raw data* with 16 channels from each subject. In this model, each data from 16 channels were utilized as the input variables to predict the output variable—to predict eyes open vs eyes close. For instance, the model using subject 1 utilizes the following input and output.

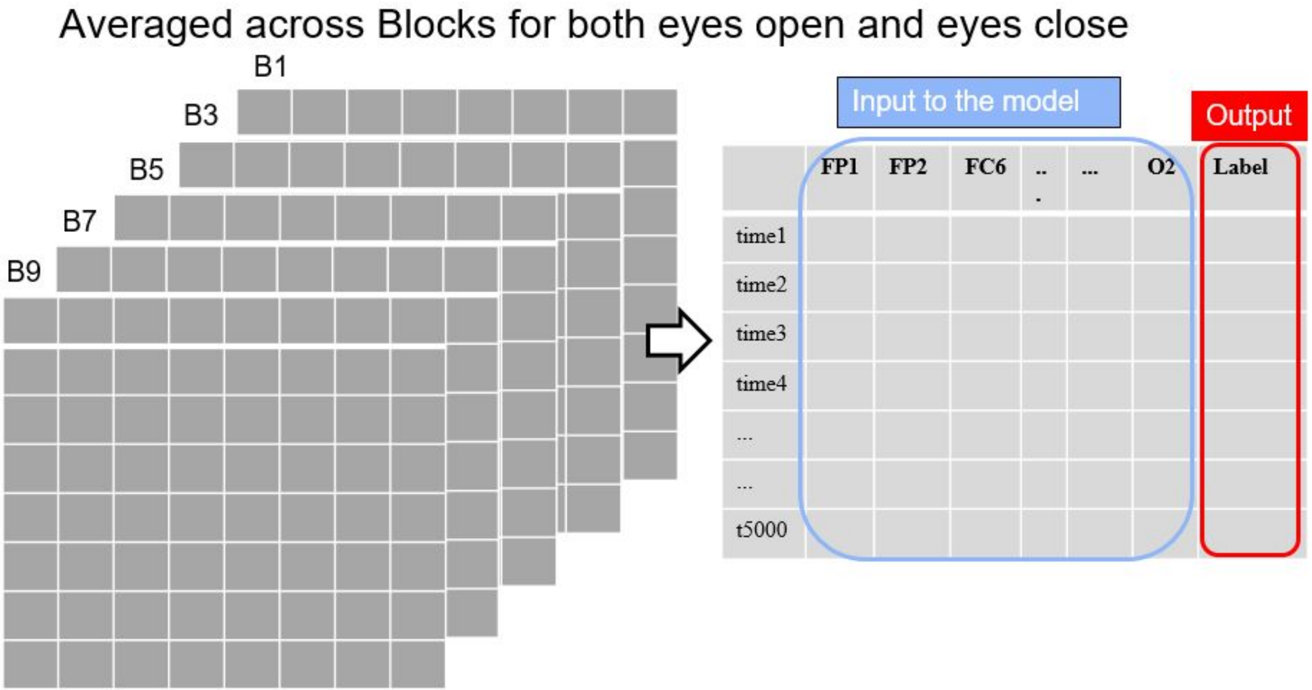
Input to the model

Output

	timestamp	FP1	FP2	FC5	FC6	FZ	T7	CZ	T8	P7	P3	PZ	P4	P8	O1	Oz	O2	stim_close	stim_open	Close_Label
2	0	20976.43	3298.686	4736.395	12961.01	5785.235	5856.096	17377.79	41.07761	11498.37	17605.34	22601.16	18115.68	15631.65	18219.4	9485.403	20960.26	0	0	0
3	0.001953	20974.38	3292.085	4739.685	12954.86	5775.713	5849.043	17374.61	29.44381	11490.84	17599.71	22595.9	18110.94	15629.73	18209.22	9478.368	20953.74	0	0	0
4	0.003906	20991	3310.958	4753.391	12972.35	5798.898	5864.392	17390.72	59.11332	11500.43	17612.22	22613.25	18130.78	15644.83	18215.57	9487.954	20965.55	0	0	0
5	0.005859	20969.73	3294.44	4728.652	12950.52	5770.265	5841.399	17361.05	46.31168	11470	17582.12	22578.85	18093.56	15619.2	18169.88	9445.059	20935.28	0	0	0
6	0.007813	20972.8	3293.226	4729.788	12951.72	5772.721	5837.273	17362.41	51.59474	11454.56	17573.66	22578.83	18096.44	15623.4	18152.24	9443.227	20937.71	0	0	0
7	0.009766	20958.1	3282.415	4719.713	12939.31	5749.752	5831.192	17353.15	28.06894	11450.27	17567.54	22572.45	18089.78	15616.66	18151.64	9440.426	20934.21	0	0	0
8	0.011719	20968.17	3296.872	4729.535	12949.61	5777.387	5832.318	17360.94	42.93429	11463.43	17580.04	22582.3	18099.79	15628.3	18169.41	9452.316	20945.62	0	0	0
9	0.013672	20972.11	3298.042	4729.956	12947.95	5784.151	5836.695	17363.06	46.11057	11465.08	17582.05	22584.49	18104.02	15631.62	18170.91	9455.08	20947.74	0	0	0
10	0.015625	20970.75	3298.972	4724.954	12950.03	5788.62	5828.304	17363.38	47.92031	11462.17	17584.1	22588.11	18108.67	15638.2	18175.93	9459.156	20953.42	0	0	0
11	0.017578	20967.13	3292.432	4725.922	12946.57	5764.85	5820.176	17357.2	43.39471	11460.11	17576.5	22580.03	18098.74	15627.63	18170.46	9452.796	20946.92	0	0	0
12	0.019531	20971.25	3296.529	4732.858	12953.6	5761.791	5834.434	17365.03	59.52311	11479.28	17589.52	22591.53	18112.98	15644.56	18184.71	9466.74	20960.96	0	0	0
13	0.021484	20968.98	3301.331	4729.292	12953.32	5779.382	5828.489	17363.44	48.47882	11462.25	17581.97	22584.79	18101.1	15623.22	18173.33	9453.48	20945.79	0	0	0
14	0.023438	20968.85	3294.833	4735.35	12949.34	5769.405	5829.569	17360.41	18.97475	11454.93	17580.73	22577.99	18094.71	15610.61	18174.01	9454.611	20942.86	0	0	0
15	0.025391	20969.21	3296.617	4732.769	12950.06	5772.344	5823.36	17363.35	44.21218	11452.82	17582.14	22583.72	18104.1	15629.6	18176.61	9460.538	20954.83	0	0	0
16	0.027344	20971.72	3293.594	4742.733	12952.17	5761.557	5834.656	17363.46	38.73988	11457.97	17584.89	22584.3	18105.51	15629.8	18184.21	9462.83	20956.97	0	0	0
17	0.029297	20965.98	3285.288	4734.733	12944.7	5752.091	5823.772	17359.25	39.71462	11448.93	17578.1	22578.37	18099.42	15624.62	18176.93	9454.823	20947.27	0	0	0
18	0.03125	20979.54	3300.75	4744.456	12959.85	5764.985	5838.837	17371.03	46.31102	11463.59	17593.59	22591.8	18113.3	15624.84	18193.26	9470.275	20957.38	0	0	0
19	0.033203	20979.33	3299.713	4749.266	12962	5765.686	5843.827	17376.15	26.84717	11465.83	17596.92	22598.69	18127.36	15641.06	18195.13	9469.71	20958.98	0	0	0
20	0.035156	20971.76	3293.786	4740.137	12955.48	5762.126	5839.502	17367.88	32.59253	11458.06	17585.71	22584.49	18107.31	15624.55	18181.75	9459.127	20950.04	0	0	0
21	0.037109	20965.54	3292.41	4732.986	12953.09	5764.688	5834.025	17359.06	50.82139	11448.87	17577.8	22572.61	18081.21	15617.52	18170.71	9446.064	20937.55	0	0	0
22	0.039063	20969.18	3291.795	4728.779	12957.71	5798.295	5839.209	17371.66	47.42571	11458.12	17586.73	22584.83	18104.89	15628.54	18177.93	9455.883	20944.17	0	0	0
23	0.041016	20980.93	3305.914	4738.313	12968.62	5796.167	5841.917	17377.88	53.26928	11465.06	17593.04	22589.7	18109.1	15631.33	18184.38	9460.417	20948.3[illegible]	0	0	0
24	0.042969	20[illegible]8.28	3320.948	4770.866	12988.58	5793.685	5873.533	17398.82	93.79349	11492.76	17616.41	22610.8	18128.98	15654.46	18204.23	9483.588	209[illegible].8	0	0	0

Dataset II (Averaging across blocks, 19 subjects)

- We *averaged eyes open and eyes close data across blocks* and collected the averaged data on all 16 channels.



Dataset III. from Exploratory Data Analysis

- The previous input and output — utilized before EDA
- The final input and output utilized in our project comes after EDA.
- Analyzed temporal pattern across different electrodes
- Idea — maybe some electrodes are enough to detect the signal

Data partitioning and model training

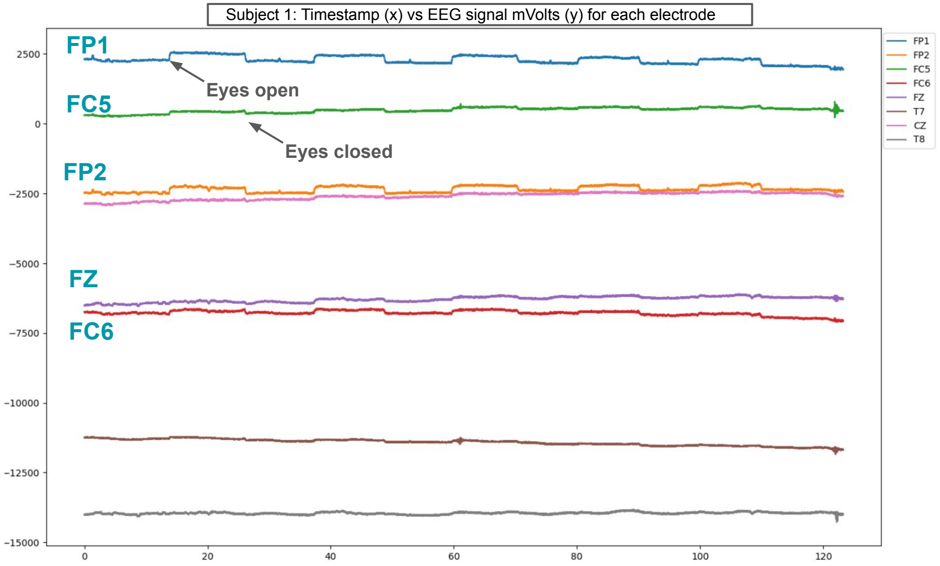
- 80% training set and 20% testing set (Participant-wise or raw-data wise)
- The training set — create the model.
- Then the testing set — evaluate the model's accuracy performance.
- Multilayer Perceptron Model — Ritesh code**
- Recurrent Neural Network — Publicly available code detecting emotional states***

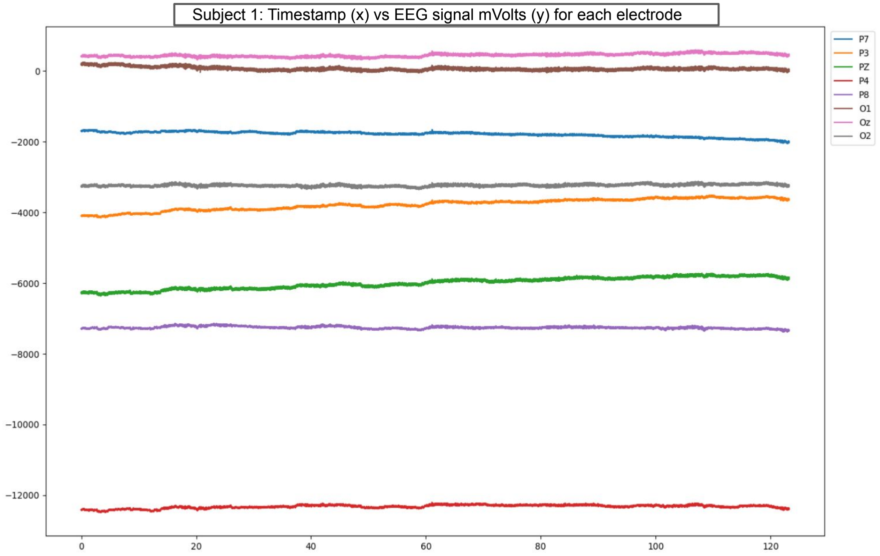
Links to the Code:
** https://github.com/coinslab/ComputationalCognitiveModeling/blob/main/python-scripts/MNISTmlpKeras.py
*** https://medium.com/geekculture/predicting-emotions-using-eeg-data-with-recurrent-neural-networks-8acf384896f5

Results

EEG data visualization for one subject

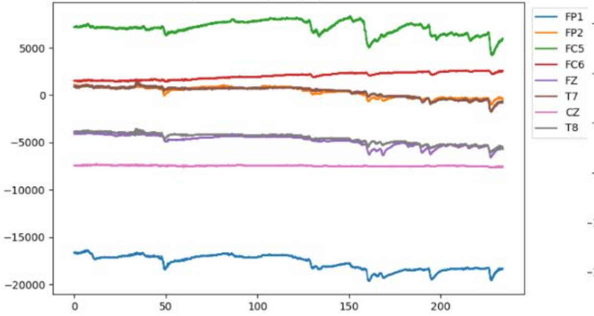
- Plot of all 16 electrodes during alternating eyes closed / eyes open conditions
- Visible pattern observed from electrodes: FP1, FP2, FC5, FC6, FZ
- Feature selection was done to test the models using only the 5 electrodes with visible pattern



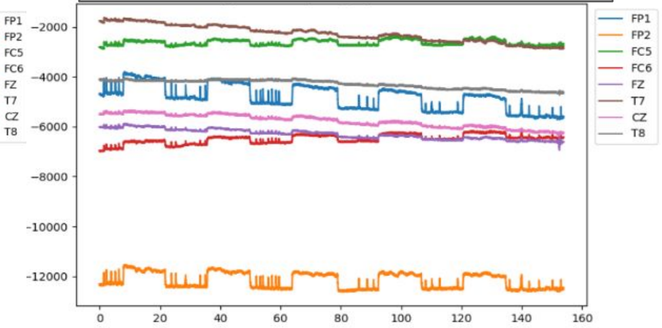


EEG data visualization for **multiple subjects**

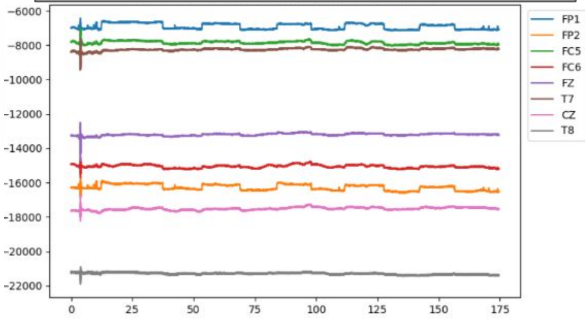
Subject 2: Timestamp (x) vs EEG signal mVolts (y) for each electrode



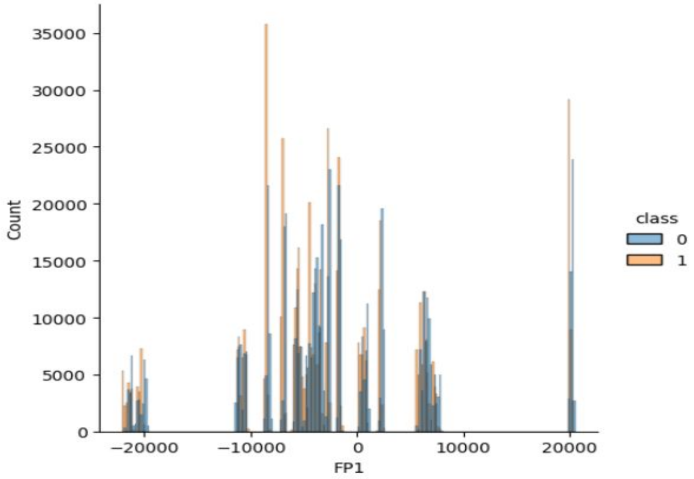
Subject 3: Timestamp (x) vs EEG signal mVolts (y) for each electrode



Subject 4: Timestamp (x) vs EEG signal mVolts (y) for each electrode



- Sampled three subjects to plot the first 8 electrodes
- Data shows **large variation in EEG signals**
- This will impact the modeling for all subjects given that not all subjects shows the same strengths and pattern.



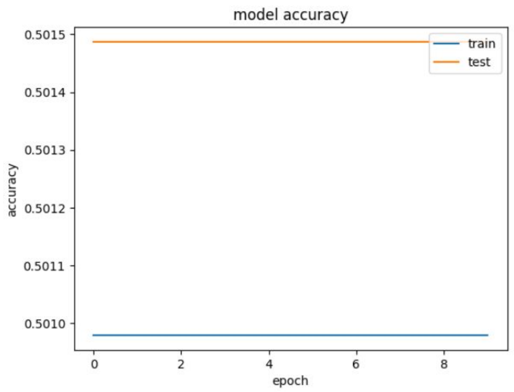
- Sampled one electrode FP1 to show data distribution for 10 subjects
- Data shows large variance in EEG signals ranging from -20,000 to 20,000 mVolts
- There is still some observed segregation of data between class 0 (closed) and class 1 (open), so model may still be predictive

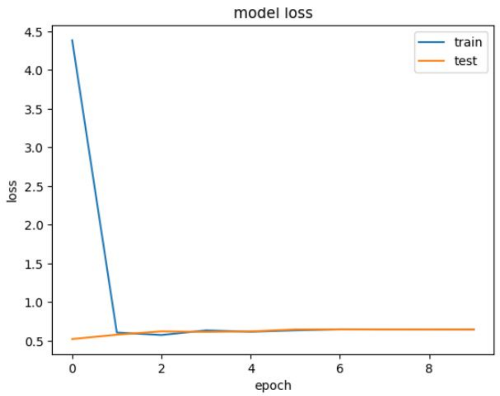
Results for Multilayer Perceptron (MLP)

Using COINS MnistMlpKeras.py

	Dataset	Test Loss	Test Accuracy
1	Raw data, 10 subjects, 16 electrodes	0.64	50.19%
2	Raw data, 10 subjects, 5 electrodes	0.57	50.15%
3	Averaged data, 19 subjects, 16 electrodes	13.30	50.04%

1. Raw data, 10 subjects, 16 electrodes






```
K = 1
input_nodes = X_train.shape[1]
inputs=layers.Input(shape=(input_nodes,))
x=layers.Dense(128,activation='relu')(inputs)
x=layers.Dense(128,activation='relu')(x)
x=layers.Dense(128,activation='relu')(x)
x=layers.Dense(128,activation='relu')(x)
outputs=layers.Dense(K,activation='softmax')(x)

model=models.Model(inputs=inputs,outputs=outputs)

model.compile(loss='binary_crossentropy',
              optimizer='Nadam',metrics=['accuracy'])

history = model.fit(X_train,y_train,
                    batch_size=128,
                    epochs=10,
                    validation_data=(X_test,y_test))

score=model.evaluate(X_test,y_test)
print('Testloss:',score[0],'Testaccuracy:',score[1])
```

Other test scenarios

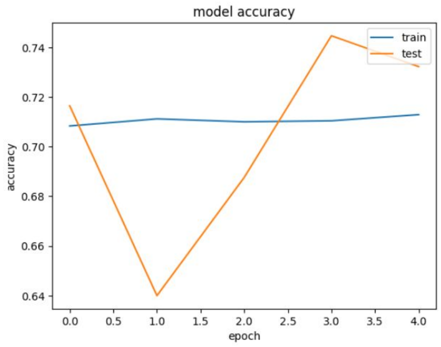
1. Tested 20 layers - degraded performance
2. Tested 20 & 50 epochs - no change

Results for Recurrent Neural Network (RNN)

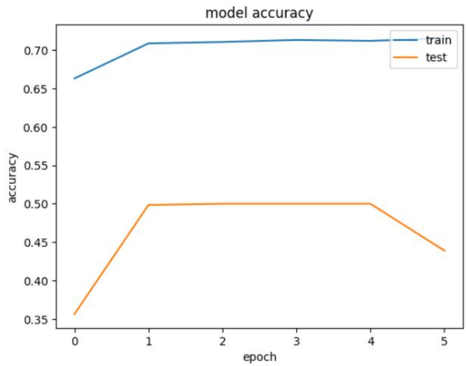
Sample modeling code online for EEG data

	Dataset	Test Loss	Test Accuracy
1	Raw data, 10 subjects, 16 electrodes	0.48	**73.15%**
2	Raw data, 10 subjects, 5 electrodes	0.62	60.54%
3	Averaged data, 19 subjects, 16 electrodes	2.02	50.15%

1. Raw data, 10 subjects, 16 electrodes



3. Average data, 20 subjects, 16 electrodes




```
inputs = tf.keras.Input(shape=(X_train.shape[1],))
expand_dims = tf.expand_dims(inputs, axis=2)
gru = tf.keras.layers.GRU(256, return_sequences=True)(expand_dims)
flatten = tf.keras.layers.Flatten()(gru)
outputs = tf.keras.layers.Dense(4, activation='softmax')(flatten)
model = tf.keras.Model(inputs=inputs, outputs=outputs)

model.compile(
    optimizer='adam',
    loss='sparse_categorical_crossentropy',
    metrics=['accuracy']
)

history = model.fit(
    X_train,
    y_train,
    validation_split=0.2,
    batch_size=32,
    epochs=5,
    callbacks=[
        tf.keras.callbacks.EarlyStopping(
            monitor='val_loss',
            patience=5,
            restore_best_weights=True
        )
    ])

model_acc = model.evaluate(X_test, y_test, verbose=0)
print('Testloss:',model_acc[0],'Testaccuracy:',model_acc[1])
```

Other test scenarios:

1. Tested 10 epochs - degraded performance

Discussions

Summary of Results

- RNN with GRU model using raw data for 10 subjects and all 16 electrodes produced the highest accuracy at 73.15%
- MLP did not improve on performance across the three datasets, showing that feed forward neural network may not be the best approach for large dataset with large variability such as EEG recordings
- As expected, averaged data has the lowest prediction accuracy for both models due to smaller training data and lost signals caused by data aggregation

Subjects	Electrodes	Data Type	Train size	MLP accuracy	RNN accuracy
10	16	Raw	597,071	50.19%	**73.15%**
10	5	Raw	597,071	50.15%	61.54%
19	16	Average	69,134	50.04%	50.15%

Other observations:

- Splitting per subject for training and testing performed worse (~50% accuracy) than randomization of combined data for all subjects
  - Splitting per subject was done to preserve the sequence of records given that RNN is best used for sequential data analysis.
  - However, the large variability of EEG data per subject likely caused the degraded performance, given that a sample of test data is unseen during training

Summary and Limitations

Overall Summary

- ***Using machine learning, can we categorize when eyes are closed or open based on the brain's electrical activity?***
  - Yes. Machine learning algorithm such as RNN can be useful in predicting conditions of eyes open or eyes closed based on EEG data of multiple subjects, however subject variability of baseline EEG signals heavily impacts the accuracy of the model.
- ***Hypothesis: Recurrent Neural Network should do better than the Multilayer Perceptron (MLP)***
  - True. Based on the experiments done, RNN performed better when using raw data compared to MLP.

Limitations

- Limited Colab GPU resources impacted the following:
  - Ability to run more tests and plotting of raw data
  - Ability to include all raw data for 20 subjects
- Pre-processing of EEG data was not done.
  - Interblock variability
  - Individuals differences in baseline
  - Removal of motion and ocular artifacts

Implications

To conclude, our research is slightly different but it complements other approaches done for analysing EEG raw data:

- Although we didn't convert our data to frequency, we were still able to see a pattern of change in our raw data, as we used an RNN code to predict eye conditions.

Applications:

- Increased Reliability of EEG Based Testing through Machine Learning
- Optimization of Mental States
  - Eye breaks

References:

- Department of Psychiatry, Nagoya University Graduate School of Medicine, Nagoya Japan - Application of eye trackers for understanding mental disorders: Cases for schizophrenia and autism spectrum disorder - https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7292297/#:~:text=If%20someone%20has%20a%20defect,pathogenic%20mechanisms%20underlying%20mental%20disorders.

Future Directions

- Personalized EEG detection through BCI
  - Utilizing BCI for uniquely tailored outputs based on an individual's daily neural activity
- Using Similar Approaches
  - Utilizing this data to look for emotional regulation, sleep monitoring, etc
  - Using different grouping styles (gender, age, etc)

References:

- Cornell University - *Deep Convolutional Neural Network for Automated Detection of Mind Wandering using EEG Signals* https://arxiv.org/abs/1902.01799
- Brain-Machine Interface Systems Lab, Systems Engineering and Automation Department, Miguel Hernández University of Elche, Elche, Spain - *Personalized Offline and Pseudo-Online BCI Models to Detect Pedaling Intent* https://www.frontiersin.org/articles/10.3389/fninf.2017.00045/full

Questions/Concerns?