Reconstruction of lost EEG data to predict and classify Alzheimer's disease

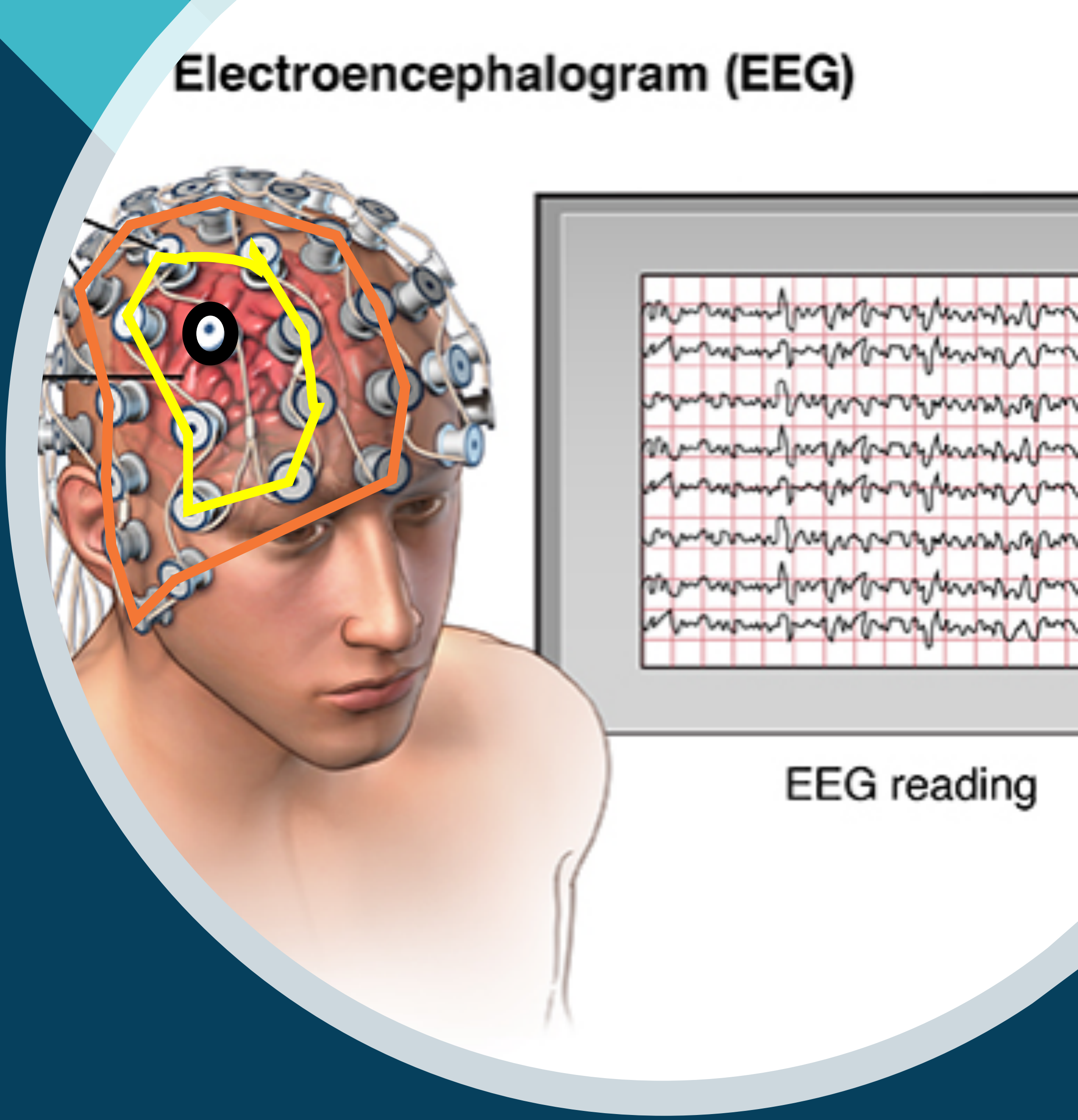
Jeremy-LIU Zihuan-City University of Hong Kong

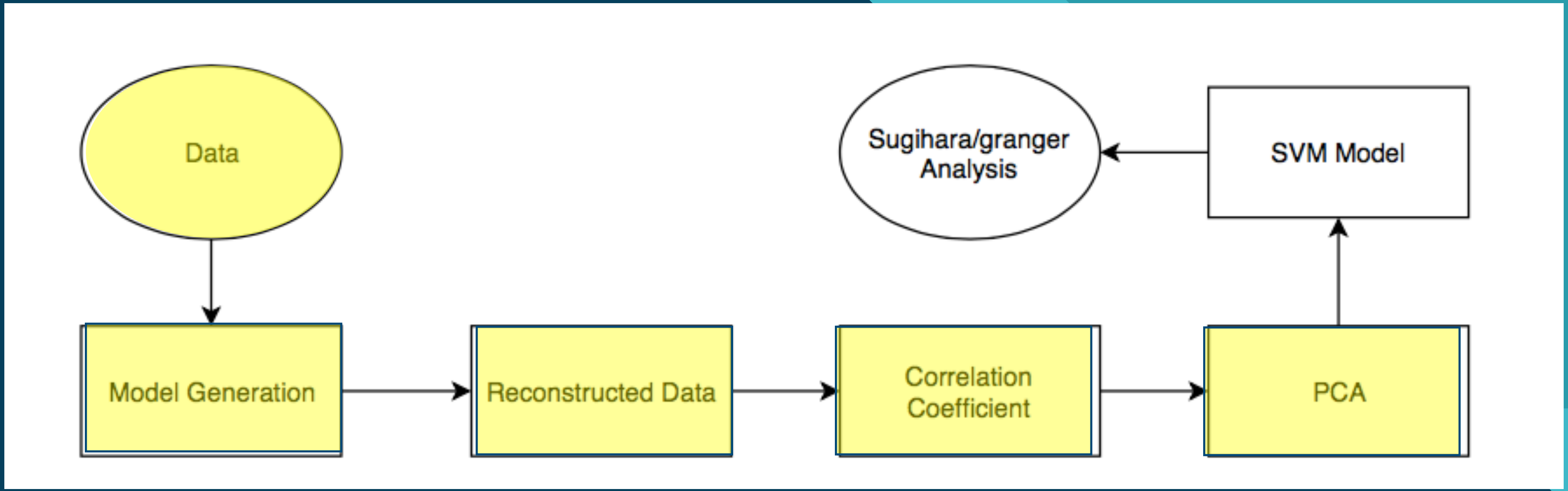
Joshua Dunkley –University of Tennessee, Knoxville

Arden-GUO Jinxu-City University of Hong Kong

Objective

- Build a reconstruction model to minimize loss of data for specific channels in scalp EEG
  - Build regression models to construct channel i for normal control (NC), mild cognitive impairment (MCI), or Alzheimer's disease (AD).
- Calculate the correlation coefficients of the predicted data and the original data
- Use PCA to find the principle components and graph the data to see if the data clusters by cognitive state
- Create a visual representation of the brain activity for each of the NC-, MCI-, AD- models to better visualize the structure of the cognitive state.





48 subjects
15 NC
16 MCI
17 AD

1 . . 9 . . . 16
subjects

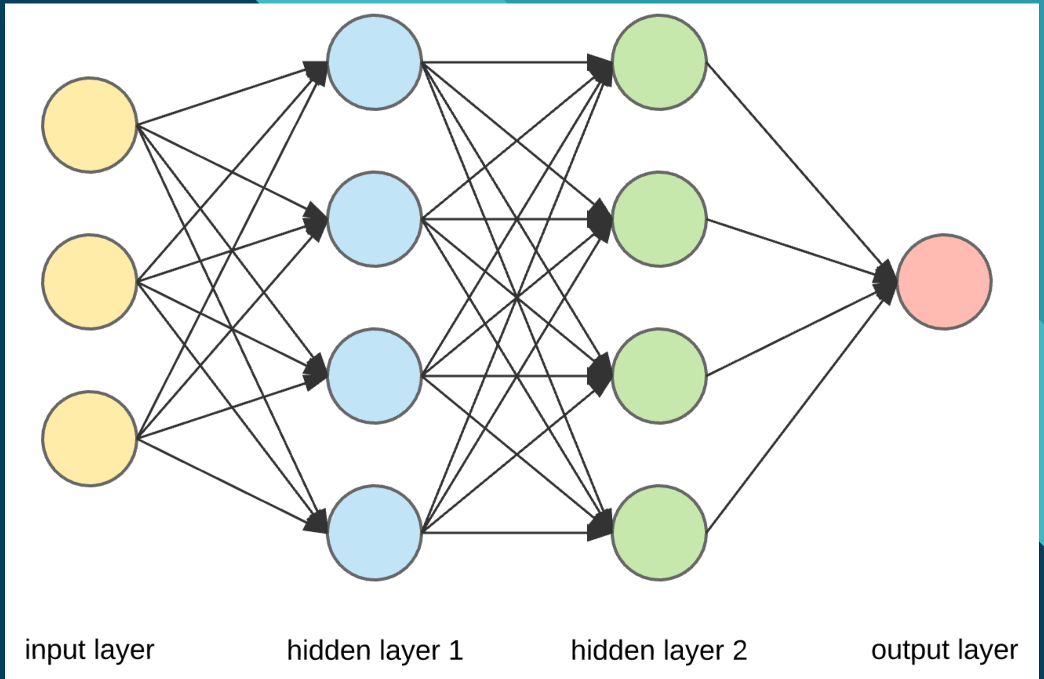
1 30
channels in a signal

3 reconstruction models : NC, MCI, AD

NC
MCI
AD

Artificial neural network

Model Type	# of hidden layers	# of nodes of hidden layers	Activation functions	Batch_size	epoch	loss
MCI	1	40	Selu linear	8	100	45
MCI	2	50 50	Relu, hard_sigmoid linear	100	1000	28.3007
MCI	2	50 50	Relu Relu linear	100	1000	25.6433
MCI	2	50 100	Relu Tanh linear	100	1000	20.9690
MCI	2	57 114	Relu Relu Linear	100	1000	26.8425
MCI	2	57 114	Relu Tanh linear	100	1000	19.1075
MCI	2	57 114	Relu Hard_sigmoid linear	100	1000	22.1408
MCI	2	60 120	Relu Relu linear	100	1000	14.7909
AD	2	60 120	Relu Relu linear	100	1000	3.1272
AD	1	5	Relu linear	5000	1000	25
AD	1	5	Relu linear	10000	200	26.72



Research Plan

- Steps

1. Separate the subjects into training sets based on the cognitive state
   - Keep a record of the subject ID, subject data, and subject cognitive state in a class object
2. Build Reconstruction models for all 3 training sets to save time with building models during actual training
3. Loop through the NC-training set and build reconstruction models for Nc_i using the data from the other n subjects in the NC-training set
4. Make predictions for each subjects along the way and calculate the correlation coefficient between the original data and the predicted data.
5. Repeat Steps 3-4 for the MCI- and AD-training sets.
6. Reorganize the correlation coefficients for all the subjects into a matrix, and perform feature reduction and PCA on the newly restructured data.
7. Create an SVM Model for principle components

Research Plan

- Algorithms/Math
  - Leave-one-out principle
  - Correlation Coefficient
  - Principle Component Analysis (PCA)

Research Plan

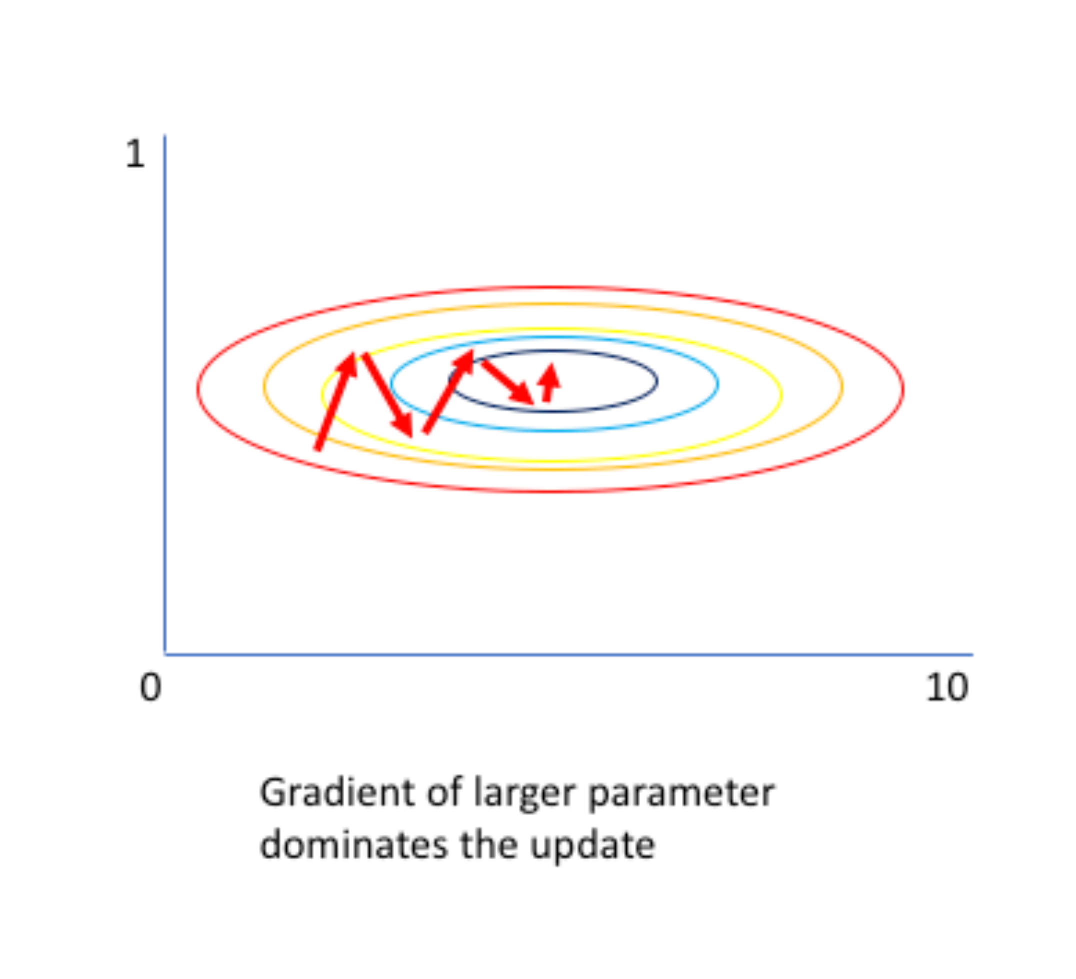
- Implementation
  - Currently working with Python script and IPython Notebooks in Google Collaboratory
  - Built Reconstruction Models using Keras with a Tensorflow backend
  - Made graphs and figures using Matplotlib
  - Use sklearn to normalize the data, and perform feature reduction and Principle Component Analysis on reconstructed data

Data normalization

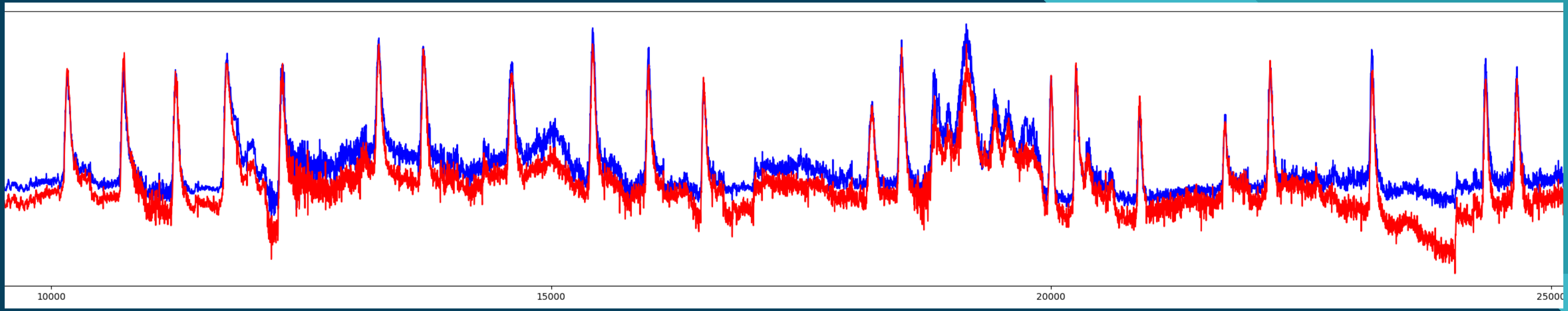
Standardization

Standardize features by removing the mean and scaling to unit variance

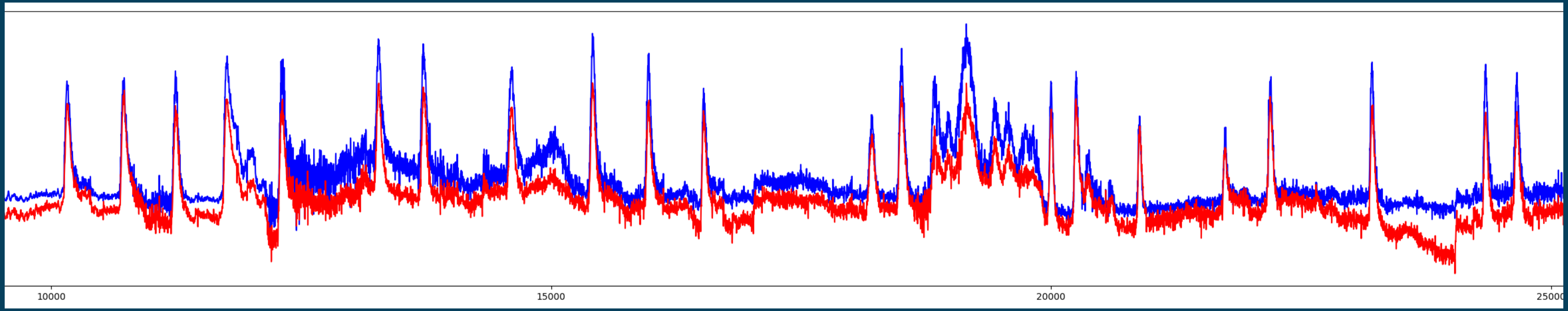
- Faster learning speed
- a feature has a variance that is orders of magnitude larger that others, it might dominate the objective function, causing the estimator unable to learn from other features correctly as expected.



Results



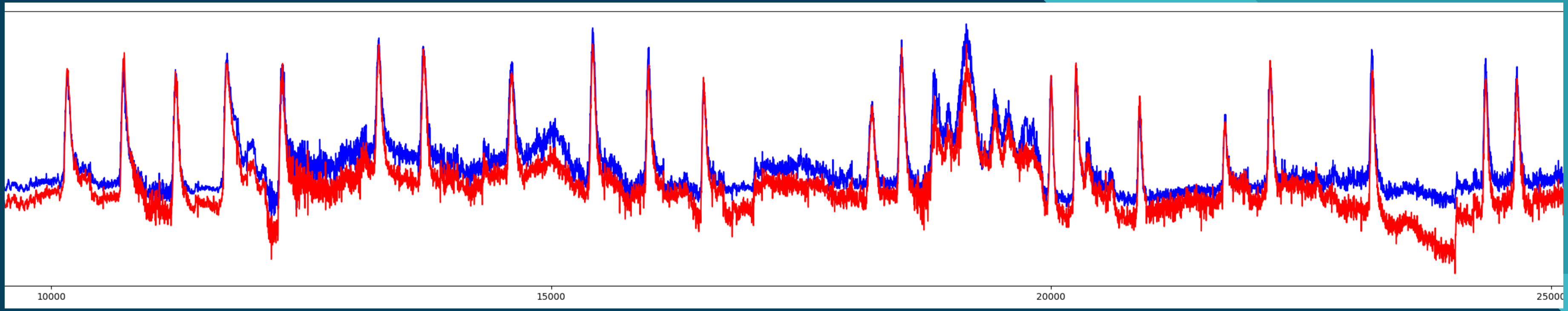
Normalized data



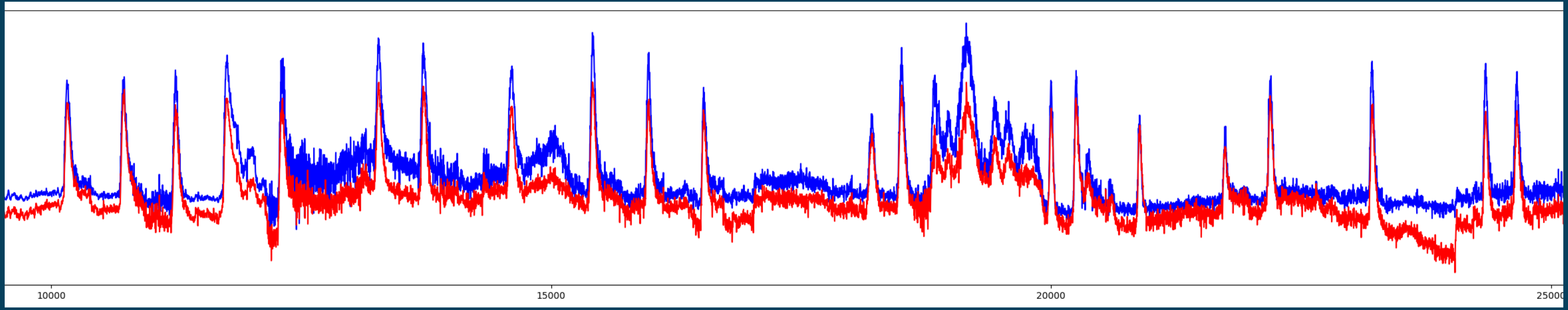
Without preprocessing

Original Data   Predicted data

Results



Normalized data

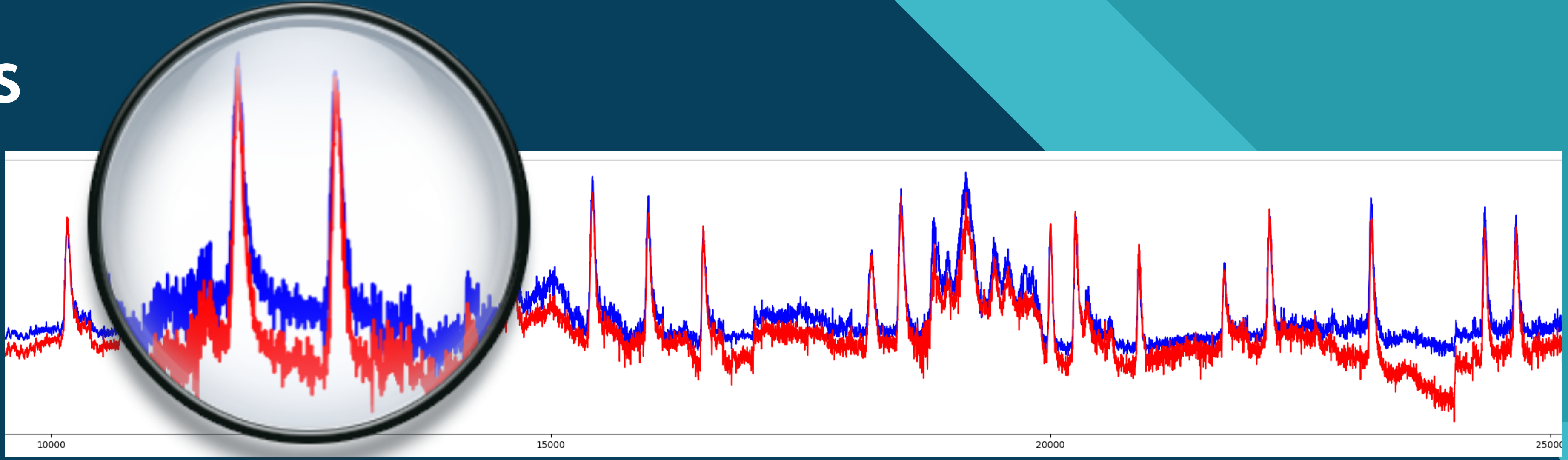


Without preprocessing

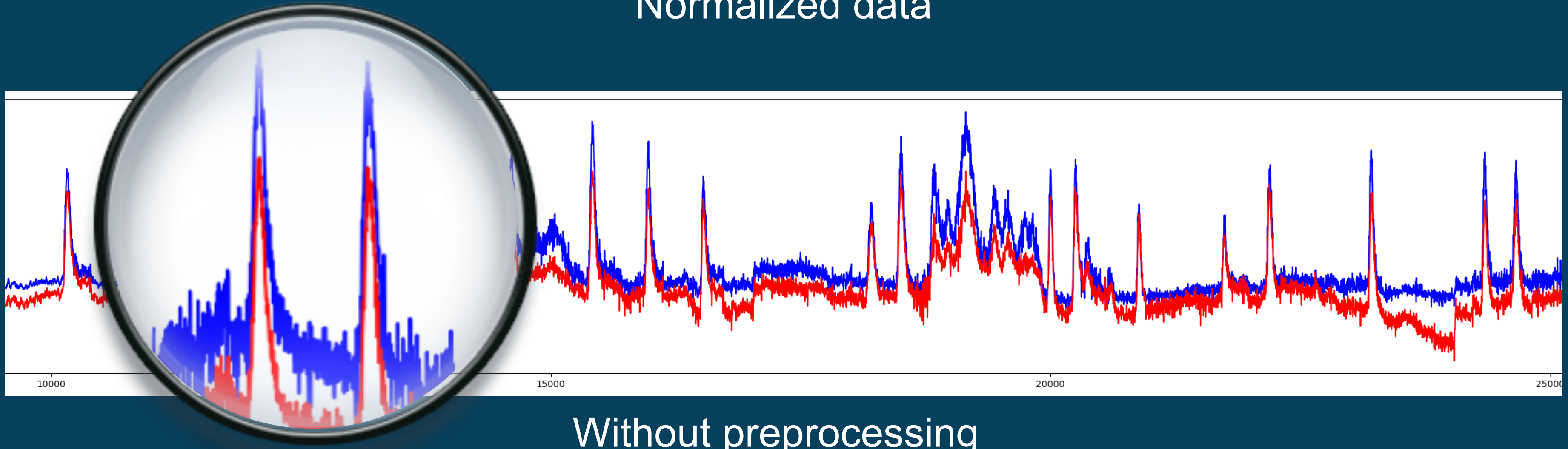
Original Data

Predicted data

Results



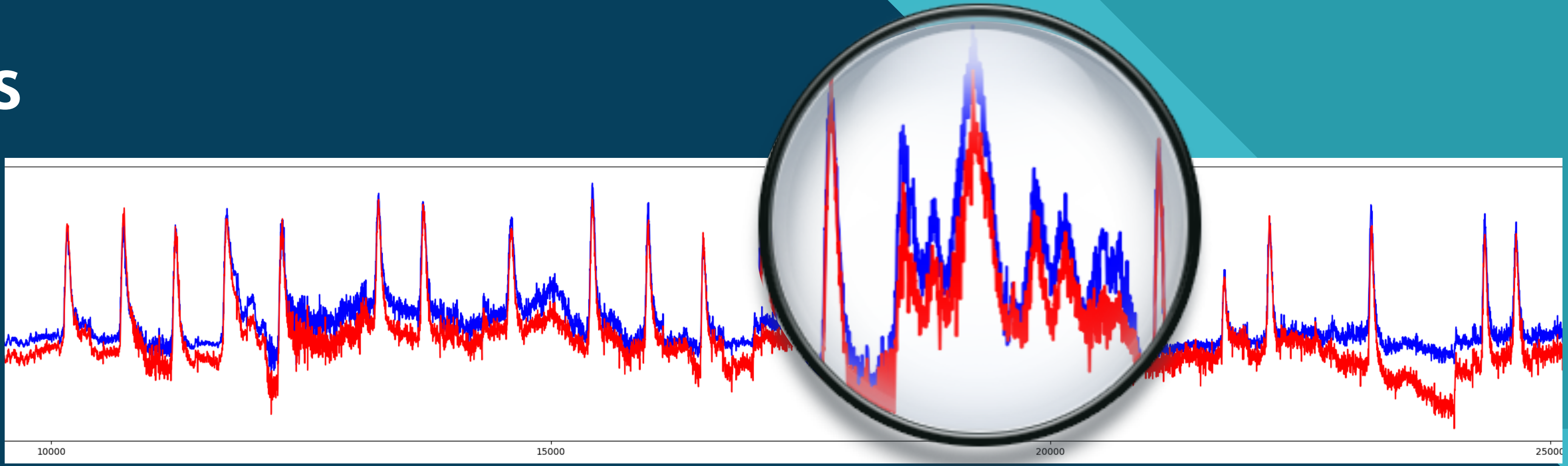
Normalized data



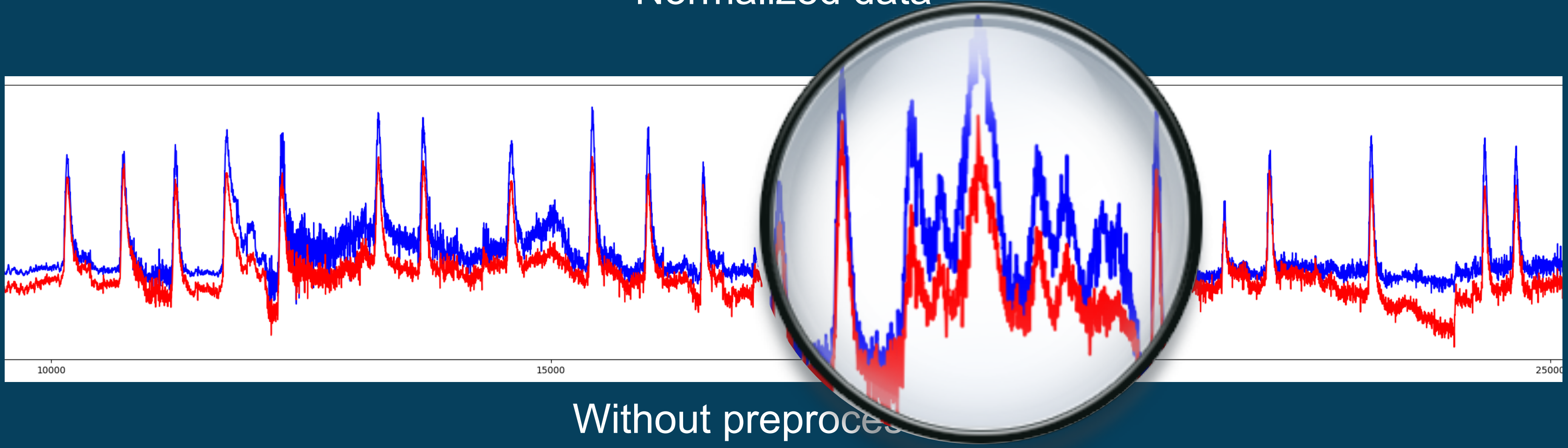
Without preprocessing

Original Data   Predicted data

Results



Normalized data



Without preproce

Original Data

Predicted data

PCA results

Correlation Matrix

- Row subjects
- Column status

PCA result

- 2 principle components
- Plot diagram
- Most of subjects clustered togrther