ECE-407 Final Project

Epileptic Seizure Detection Project

Arash Fouman Ajirlou - Spring 2020

Outline

- Overview of the dataset
- Preprocessing
- Gaussian classifier
- K-Nearest-Neighbors classifier
- Logistic Regression classifier
- Conclusion

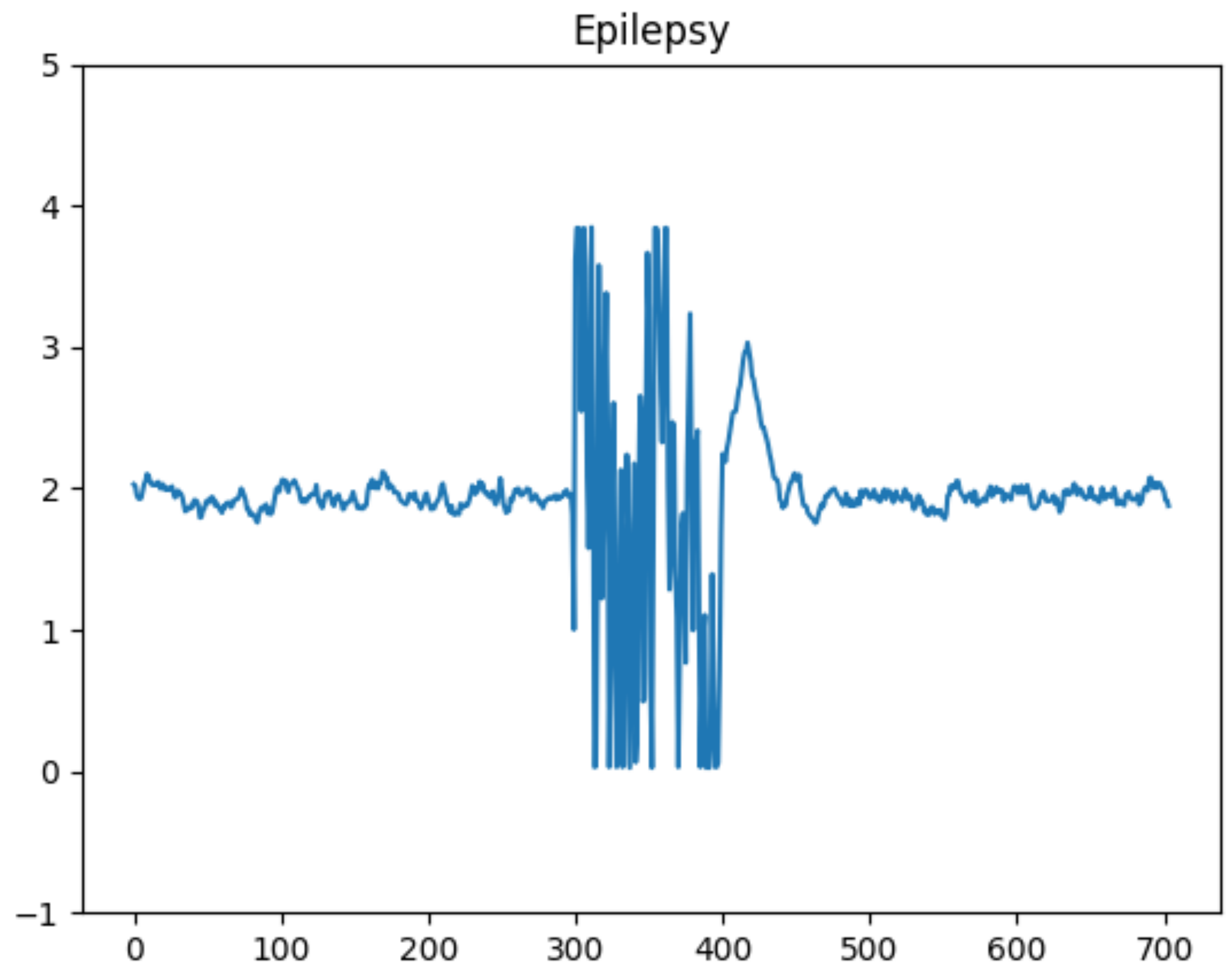
Overview of the dataset

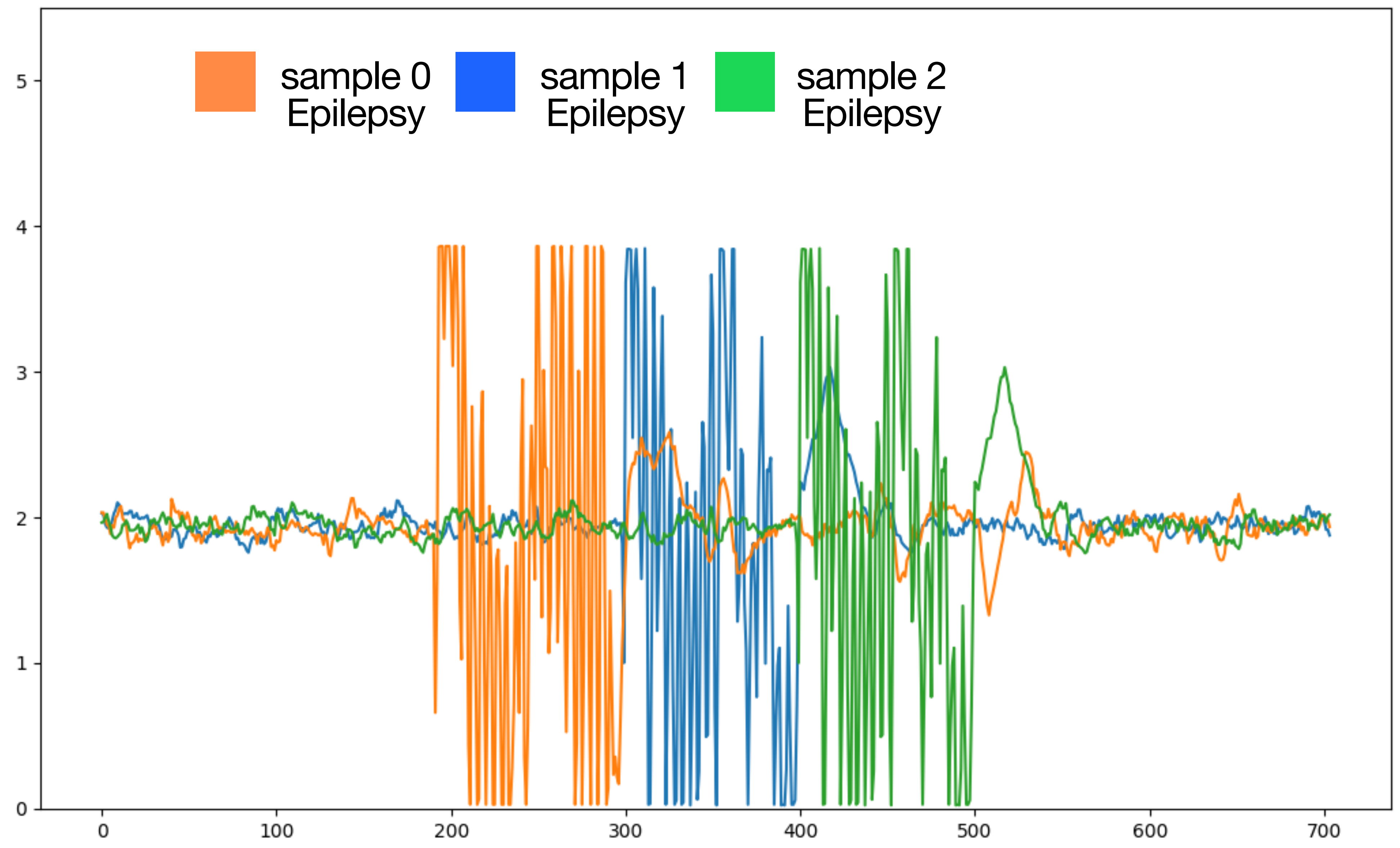
- Real-time data from Epileptic seizure detection
- Three classes : {Epilepsy, Normal, Nothing (No-Movement)}
- 240 samples in train set and 60 samples in test, splitted uniformly over the three classes
- Each sample includes 704 time stamps

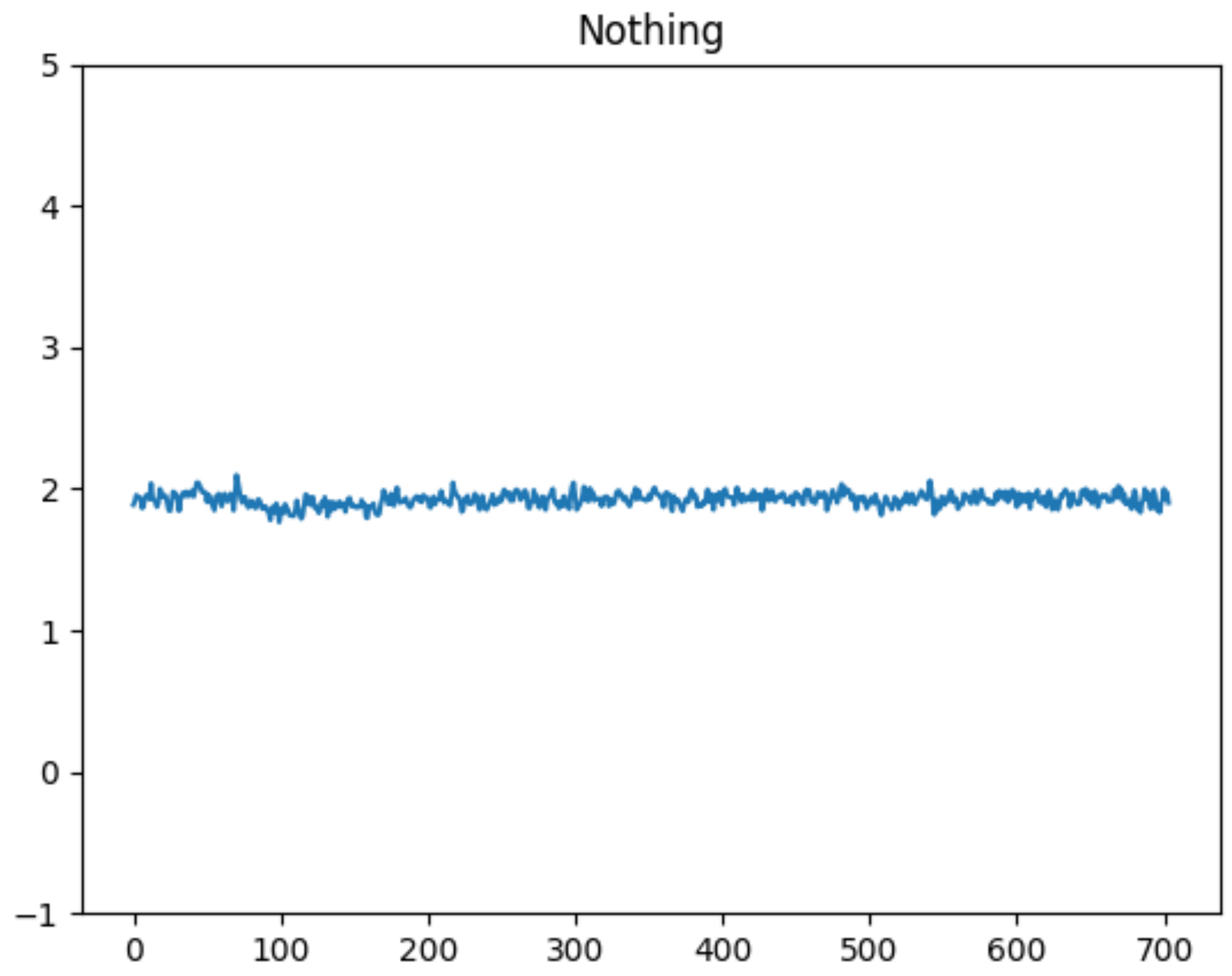
Overview of the dataset (cnt'd)

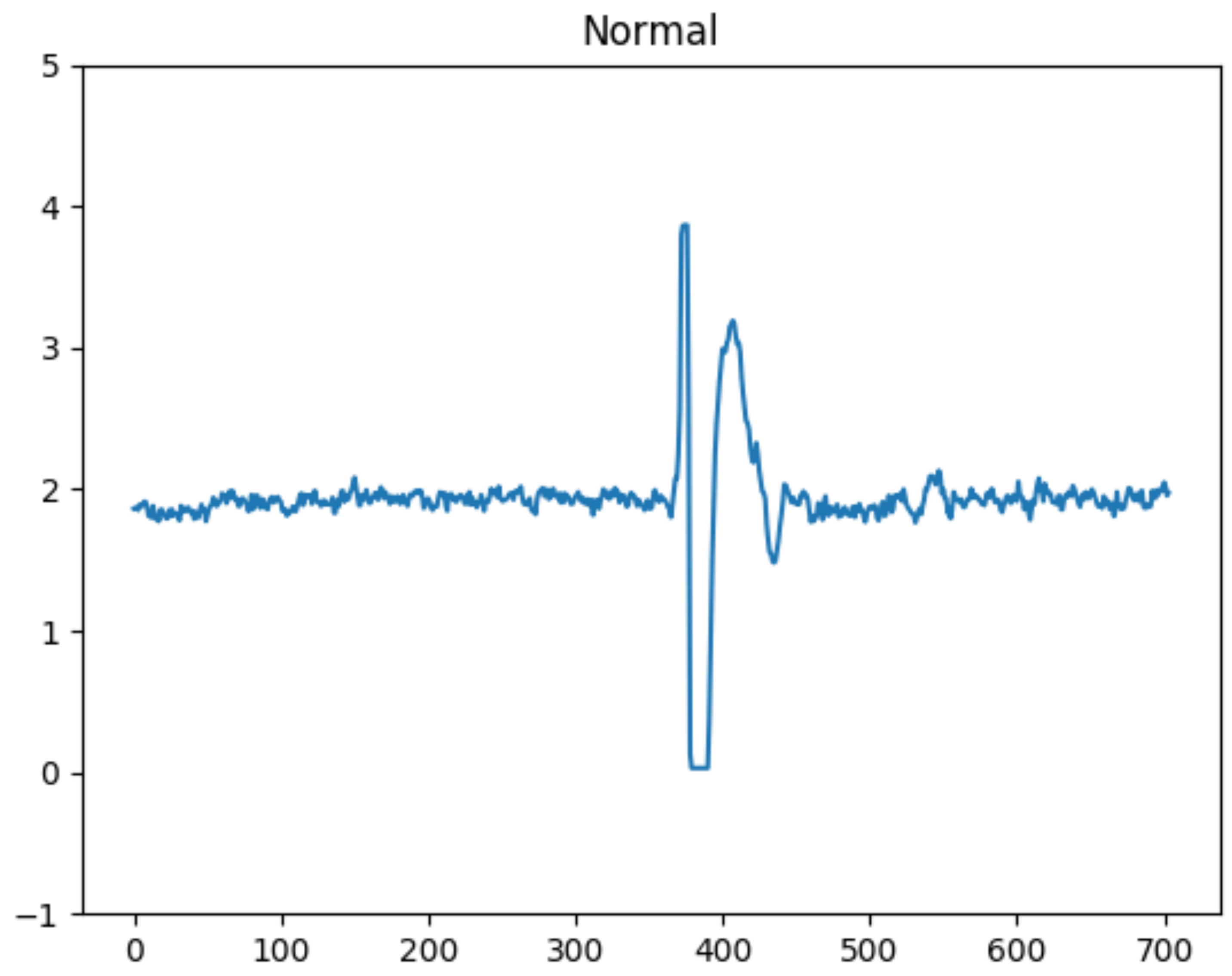
Data samples

- Only a small fraction of the captured dataframe contains useful information
- The time stamps containing information walk through the timeline





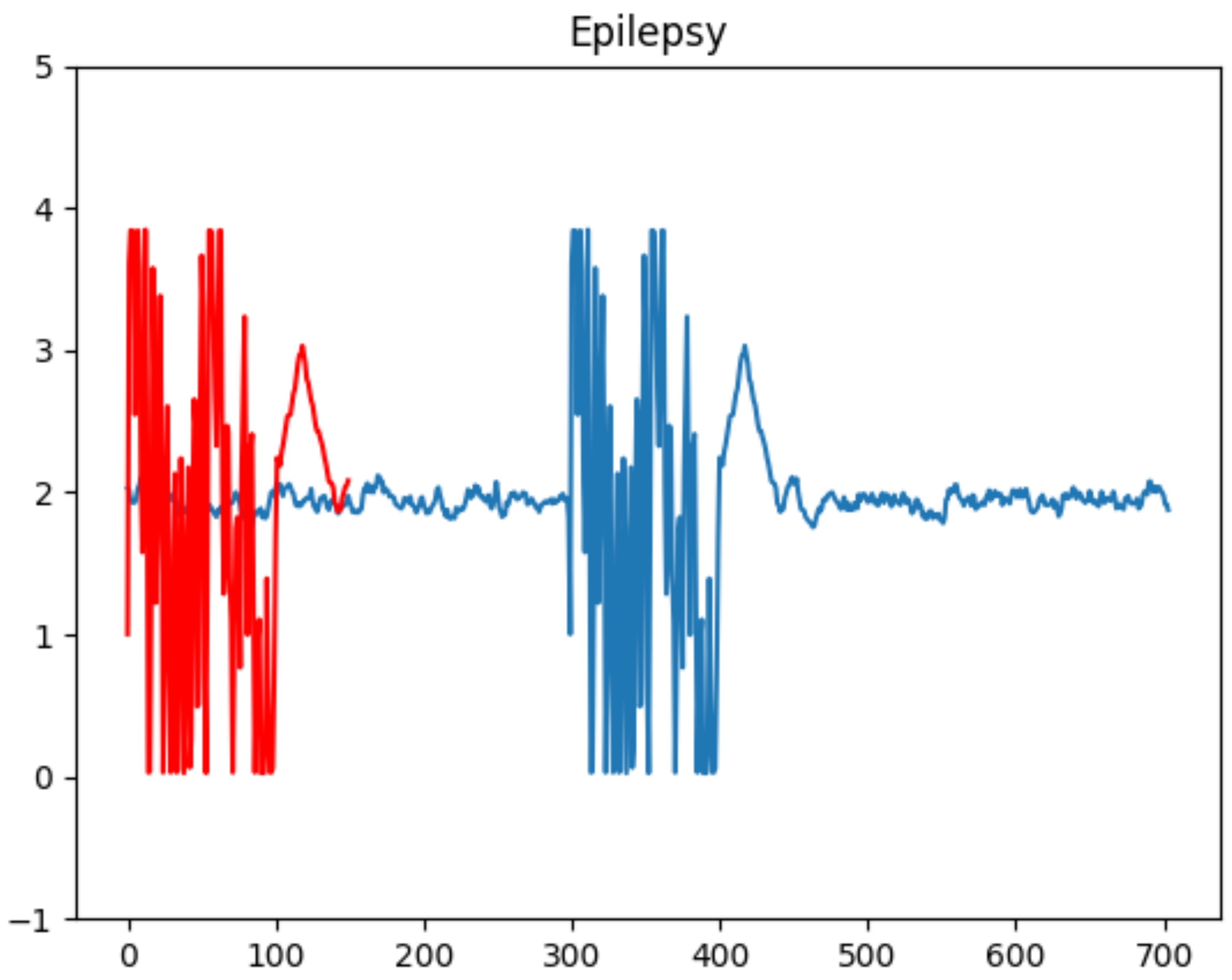


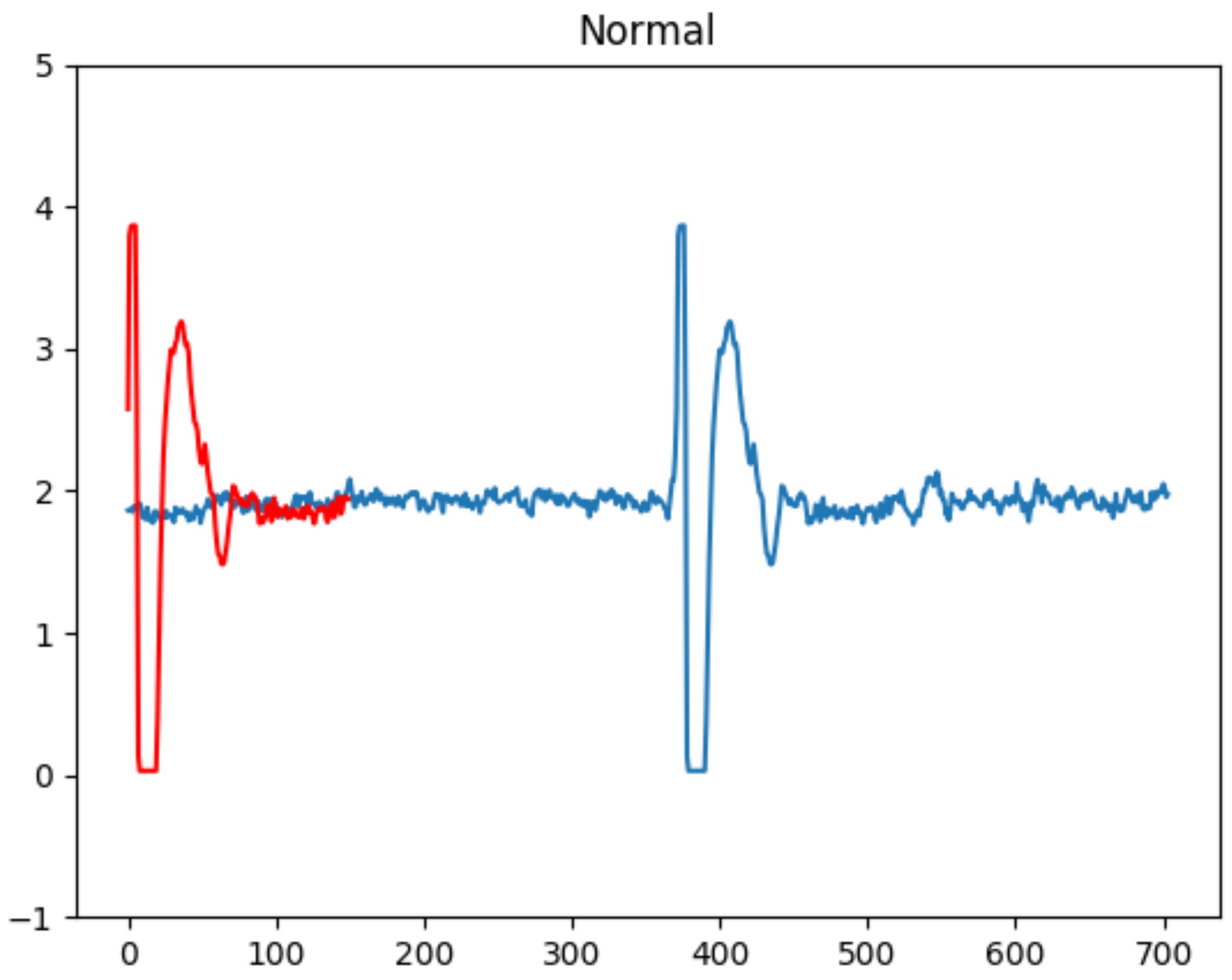


Preprocessing

Crop the data frame

- The blue waveforms are the original waveforms
- The red waveforms are the cropped portions
- Cropping has no effect on samples from *Nothing* class.
- Perform *scaling* on the data for Gaussian classifier to meet probability theory requirements





Gaussian Classifier

Mutliclass

$$p(x \mid \mu, \Sigma) = \frac{1}{\sqrt{2\pi^p \mid \Sigma \mid}} exp(-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu))$$

- Preprocess and scale the dataset
- Calculate the mean (u) and covariance matrix (sigma) for each class (according to the formulas learned in the class)
- A hint of Principal Component Analysis
- For each test sample (*x_test*) from test dataset:

    1. calculate the likelihood of *x_test* for each class
    2. the highest likelihood determines the class.
    3. Note that prior probabilities are equal

Gaussian Classifier (cnt'd)

Results

$$p(x|\mu,\Sigma) = \frac{1}{\sqrt{2\pi^p|\Sigma|}} exp(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu))$$

- After preprocessing:
  - For *Epilepsy* class: sub-features 1 to 11
  - For *Normal* class: sub-features 16 to 25

contribute the most to the likelihood of a test sample

- 91.7% accuracy

Confusion Matrix

	Epilepsy	Normal	Nothing
Epilepsy	15	5	0
Normal	0	20	0
Nothing	0	0	20

K-Nearest-Neighbors Classifier

- Preprocess dataset
- Load the train dataset
- A hint of Principal Component Analysis (the first 25 features contain the most information)
- For each test sample (*x_test*) from test dataset:
    1. calculate the distance of *x_test* with all the points
    2. sort the distances in descending order
    3. vote among the top *k* indices in the sorted list

K-Nearest-Neighbors Classifier

Results

- K = 1 : 91.7% accuracy
- K = 3 : 91.7% accuracy
- K = 5 : 91.7% accuracy

Confusion Matrix (K=1)

	Epilepsy	Normal	Nothing
Epilepsy	20	0	0
Normal	0	15	5
Nothing	0	0	20

Confusion Matrix (K=3)

	Epilepsy	Normal	Nothing
Epilepsy	20	0	0
Normal	0	15	5
Nothing	0	0	20

Confusion Matrix (K=5)

	Epilepsy	Normal	Nothing
Epilepsy	20	0	0
Normal	0	15	5
Nothing	0	0	20

Logistic Regression Classifier

- Preprocess dataset
- Load and train using Scikit-learn library in python
- A hint of Principal Component Analysis (the first 16 features contain the most information)
- Save the weight matrix (*w*)
- For each test sample (*x_test*) from test dataset:
    1. multiply *x_test* by *w*
    2. apply *softmax*
    3. the highest-value element determines the class

Logistic Regression Classifier

Results

- 83.3% accuracy

Confusion Matrix

	Epilepsy	Normal	Nothing
Epilepsy	20	0	0
Normal	10	10	0
Nothing	0	0	20

Conclusion

- Gaussian, KNN, and LR provide competitive accuracies
- A combination of these approaches can increase the prediction accuracy. E.g.:
    A. Gaussian only misclassifies as class *Normal*
    B. KNN never misclassifies as class *Normal*
    C. use Gaussian, if the prediction was *Normal* verify it by KK
    D. For the given dataset, an accuracy of 100% is achievable

Algorithm		Accuracy
Gaussian		91.7%
KNN	K = 1	91.7%
	K = 3	91.7%
	K = 5	91.7%
Logistic Regression		83.4%