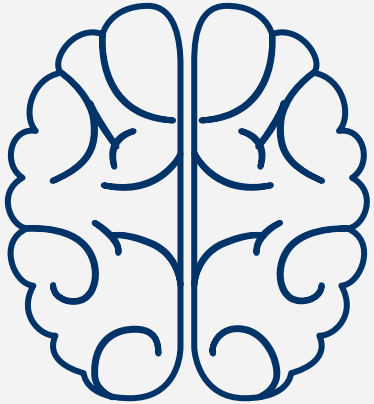
universität wien

Detecting Button presses with Autoencoders

Neuroinformatics: Machine Learning for Neuronal Data Analysis

Luis Caumel Morales
Lukas Karner
Alexander Stähle
Patrick Pollek

INTRODUCTION



Why Anomaly Detection using (Convolutional) Autoencoders on diverse datasets of button press EEG data makes sense:

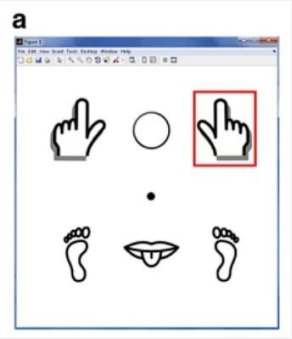
- real-time detection of button presses in EEG data
- varying types of presses can be detected (duration, motion)
- applicable to a range of applications (HCI, industrial, security systems)

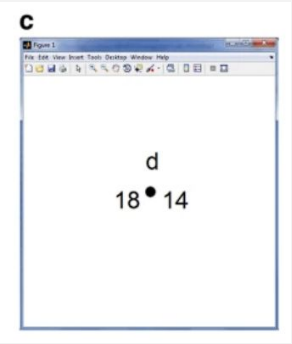
Overall: more robust and generalizable way to detect decision making

DATA

We have two datasets to analyze:

- NoMT (Image a): Experiments where recording sessions were passively watched the computer screen throughout the recording session where a greater number of imageries were shown.
- FreeForm (Image c): Examining the discrimination of voluntary motor movements prior to their physical manifestation,
    - Participants were asked to press the "d" or "l" keys voluntarily, using the left or right hand, at arbitrary times.





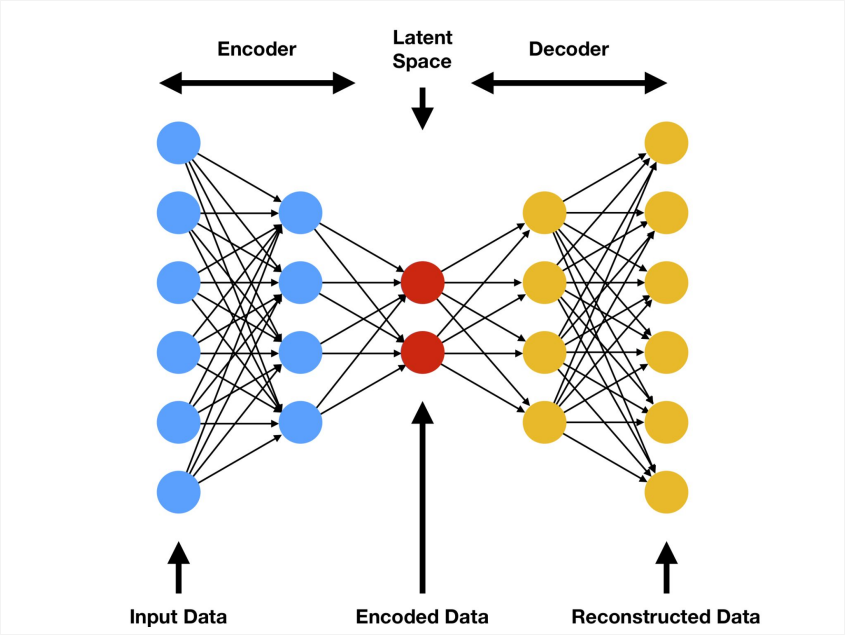
PROCESSING DATA

- We transform the data to a mne array to make it easier to deal with the data and set eeg reference average as we are working with EEG data.
- Create annotations, for annotating segments of our data.
- Creation of epochs, to divide our dataset and identify when an experiment is taking place, since the eGUI state is easy to identify.
- Creation of datasets, splitting into training and testing
    - Data NoMT , our baseline and where the subject is in a resting position which no input on the screen, event with the ID 0
    - Data NoMT2, will correspond to the data which comes from the same trials and session of the previous data but we select here epochs where there is something shown on the screen, events 2
    - FREEFORM data, here the subject can voluntarily press buttons (either left or right)
- Scale the data, as we observed a large difference between the observed datasets, the data have been normalised allowing for a more accurate comparison.

METHODS

Idea : Train AE on resting state data.
Observe high reconstruction error on button press data.

- Brain activity which is generated by planning to press a button should change the EEG signal in a way such that is is different enough to create a higher reconstruction error.
- How big should the autoencoder be ?
- Should we use Convolutional autoencoder ?



SIZES OF AE

We tried several sizes of AE to see which one performs best:

	small:	medium:	large:
Encoder	Input 8421 neurons	Input 8421	Input 8421
		dense layer 256	dense layer 1024
		dense layer 128	dense layer 512
			dense layer 128
	dense layer 64	**dense layer 64**	**dense layer 64**
Decoder			dense layer 128
		dense layer 128	dense layer 512
		dense layer 256	dense layer 1024
	output 8421 neurons	output 8421	output 8421
Parameter :	1,08 mio	2,19 mio	18,4 mio

CONVOLUTIONAL AE

Encoder

CONV ae:	MORE LATENT VAR:	MORE LAYERS:
Input (401,21)	Input (401,21)	Input (401,21)
Conv1D(42,3)	Conv1D(42,3)	Conv1D(42,3)
MaxPooling1D(2)	MaxPooling1D(2)	MaxPooling1D(2)
Conv1D(84,3)	Conv1D(84,3)	Conv1D(84,3)
MaxPooling1D(2)	MaxPooling1D(2)	MaxPooling1D(2)
Flatten	Flatten	Conv1D(168,3)
dense layer 50	**dense layer 256**	MaxPooling1D(2)
		Flatten
		dense layer 50
Parameter : 900k	4.4 mio	1 mio.

TRAINING OF THE AUTOENCODERS

- Error measure : MSE
- optimizer : ADAM (several different lr's)
- dropout after every layer
- 1000 epochs (none conv ones)
- batchsize 512
- training of one AE in the range of 2 to 10 min
- activation functions: Relu,Gelu

RESULTS

normal Autoencoder, unscaled data

model size	NoMT	NoMT2	FREEFORM
small	90.32	89.93	19.91
medium	128.32	125.52	24.48
large	178.56	174.09	30.59

Table: Reconstruction errors on different test sets.

RESULTS

normal Autoencoder, scaled data

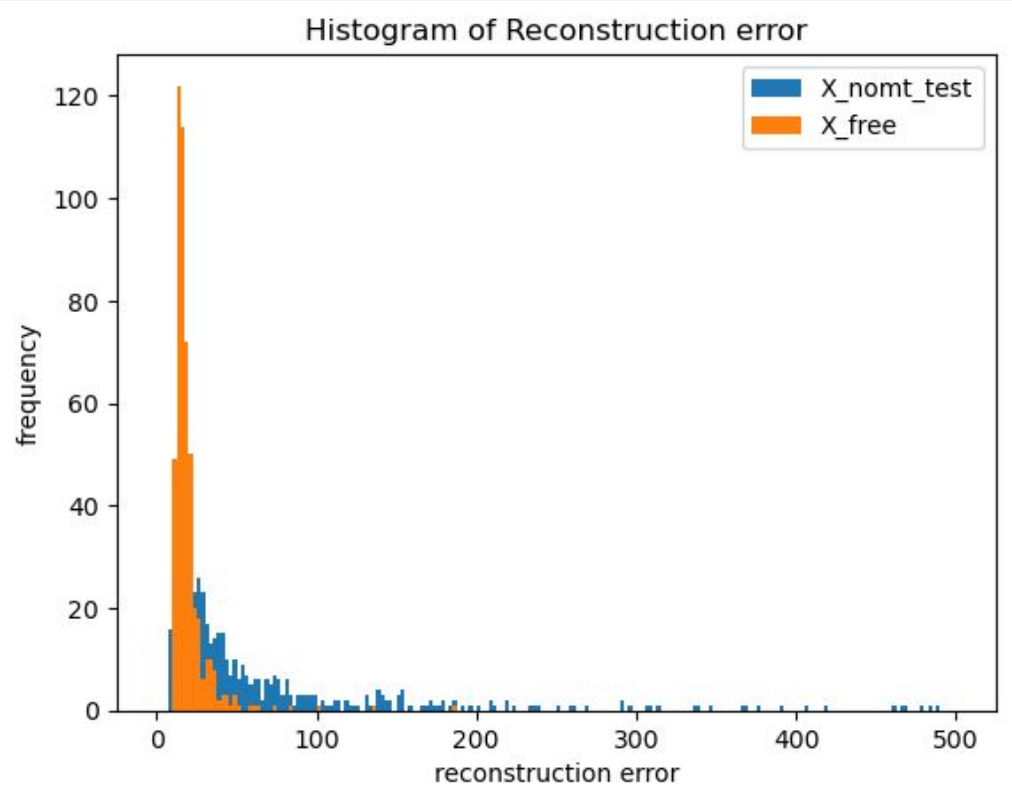
model size	NoMT	NoMT2	FREEFORM
small	0.4276	0.4523	0.6897
medium	0.5591	0.5889	0.8299
large	0.6307	0.6506	0.8386

Table: Reconstruction errors on different test sets.

SMALL AUTOENCODER - UNSCALED DATA

Reconstruction Error of: X_nomt_test
Mean = 89.639534
Median = 37.58262

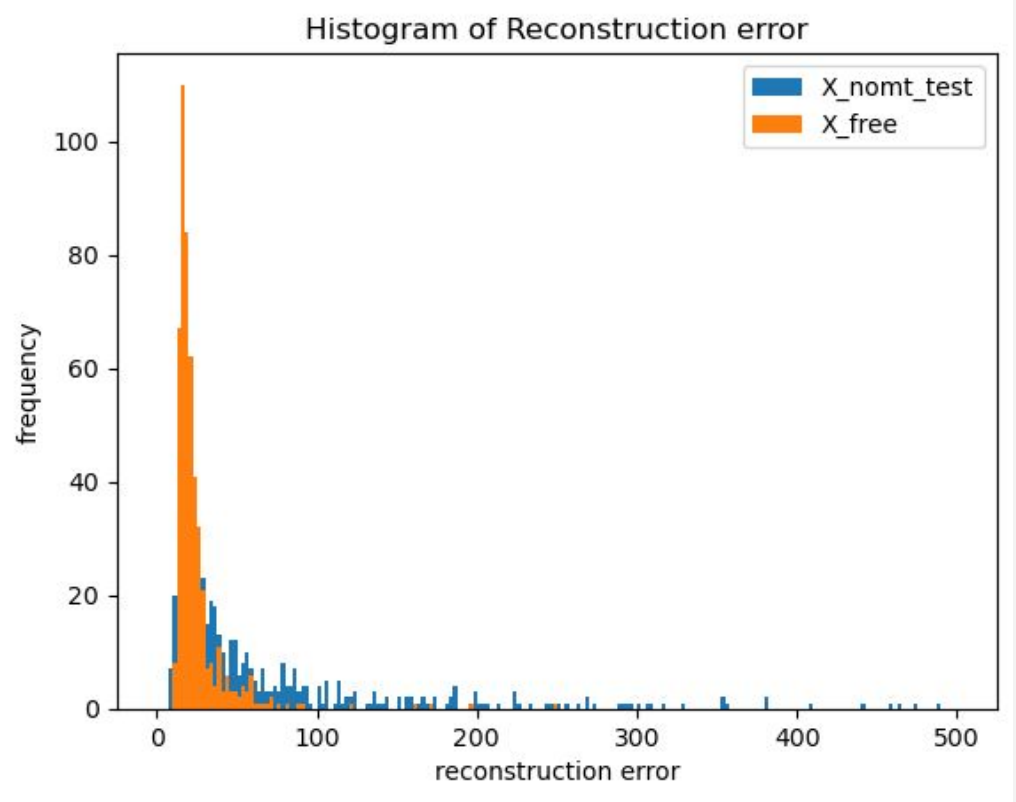
Reconstruction Error of : X_free
Mean = 19.766518
Median = 16.590572



MEDIUM AUTOENCODER - UNSCALED DATA

Reconstruction Error of : X_nomt_test
Mean = 125.83797
Median = 43.448288

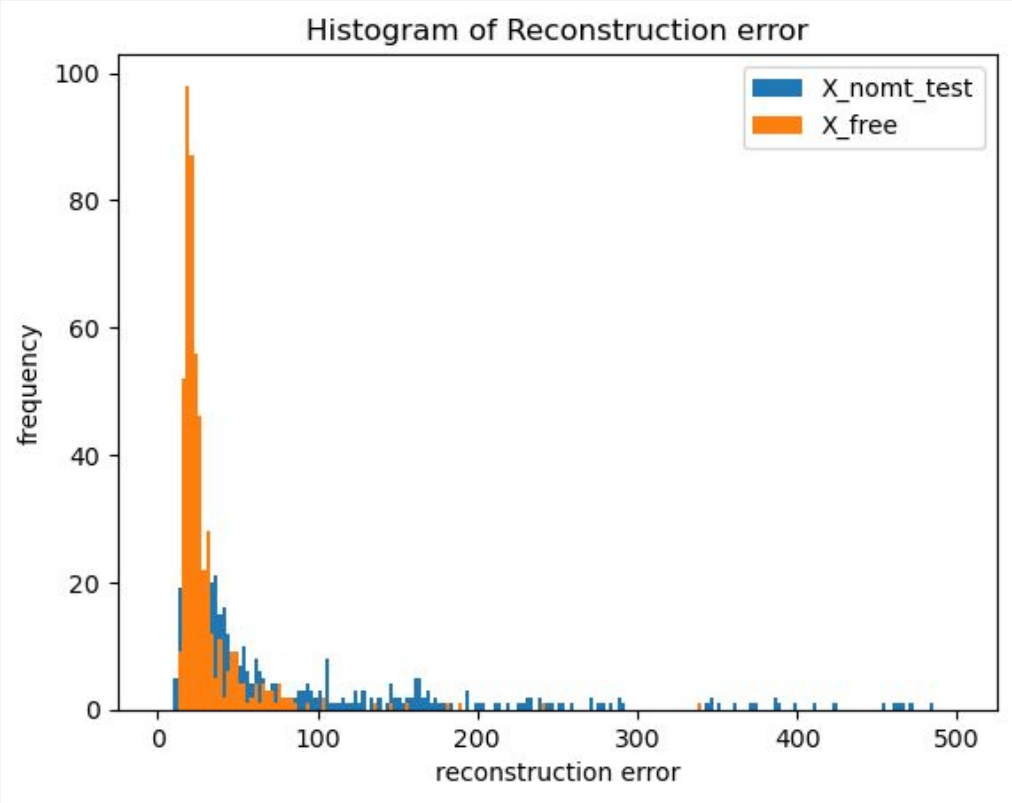
Reconstruction Error of : X_free
Mean = 24.586403
Median = 19.243282



LARGE AUTOENCODER - UNSCALED DATA

Reconstruction Error of : X_nomt_test
Mean = 174.92694
Median = 48.188297

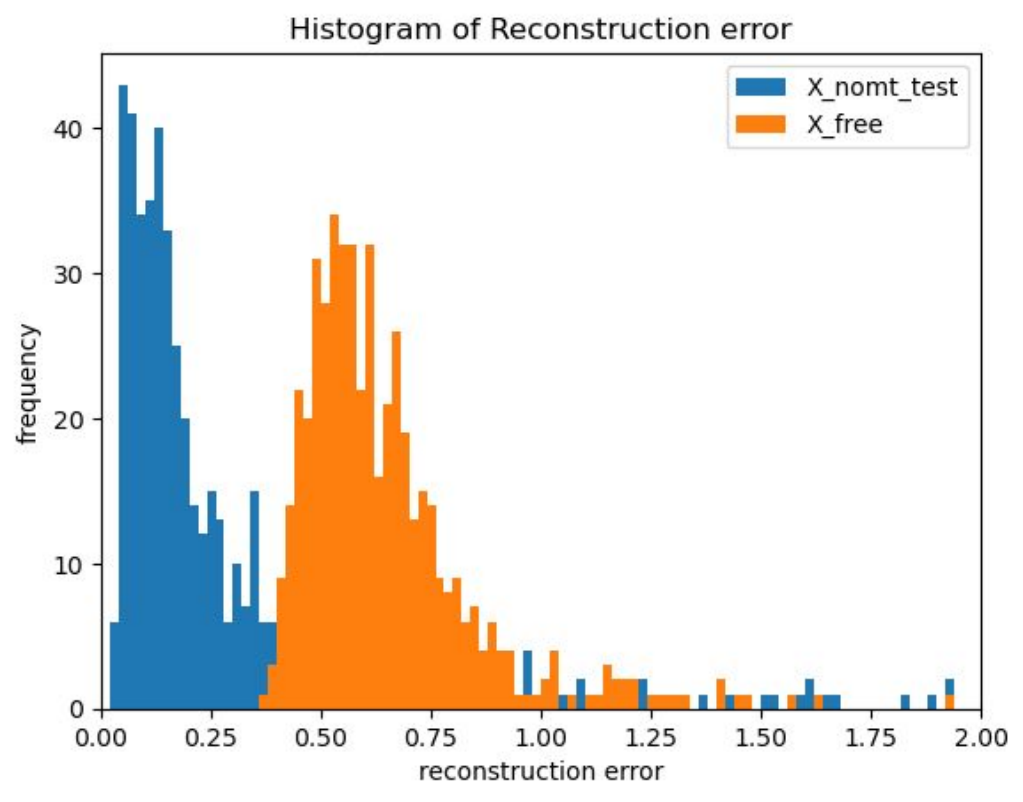
Reconstruction Error of : X_free
Mean = 30.278997
Median = 22.69931



SMALL AUTOENCODER - SCALED DATA

Reconstruction Error of : X_nomt_test
Mean = 0.42486674
Median = 0.17336366

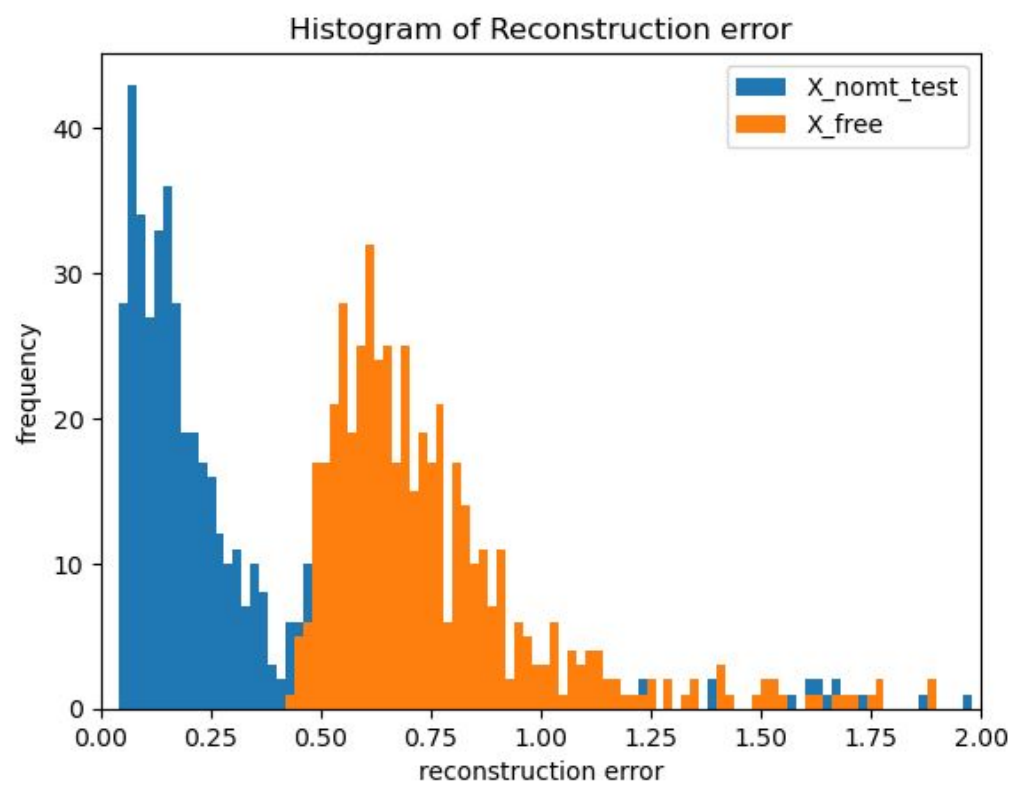
Reconstruction Error of : X_free
Mean = 0.68259436
Median = 0.60096836



MEDIUM AUTOENCODER - SCALED DATA

Reconstruction Error of : X_nomt_test
Mean = 0.5512001
Median = 0.20311476

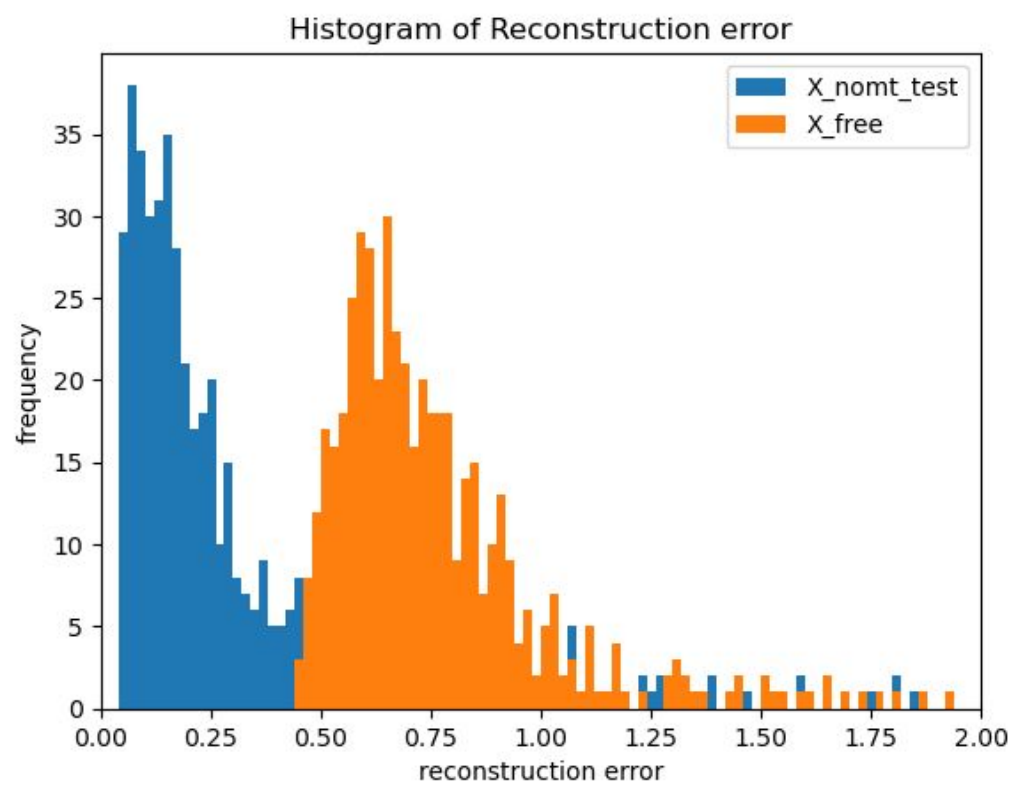
Reconstruction Error of : X_free
Mean = 0.8318454
Median = 0.69081753



LARGE AUTOENCODER - SCALED DATA

Reconstruction Error of : X_nomt_test
Mean = 0.6125158
Median = 0.20422757

Reconstruction Error of : X_free
Mean = 0.84346235
Median = 0.7008301



RESULTS - AUTOENCODER

Based on the histograms we take the thresholds to be:

- small → 0.4
- medium → 0.42
- large → 0.42

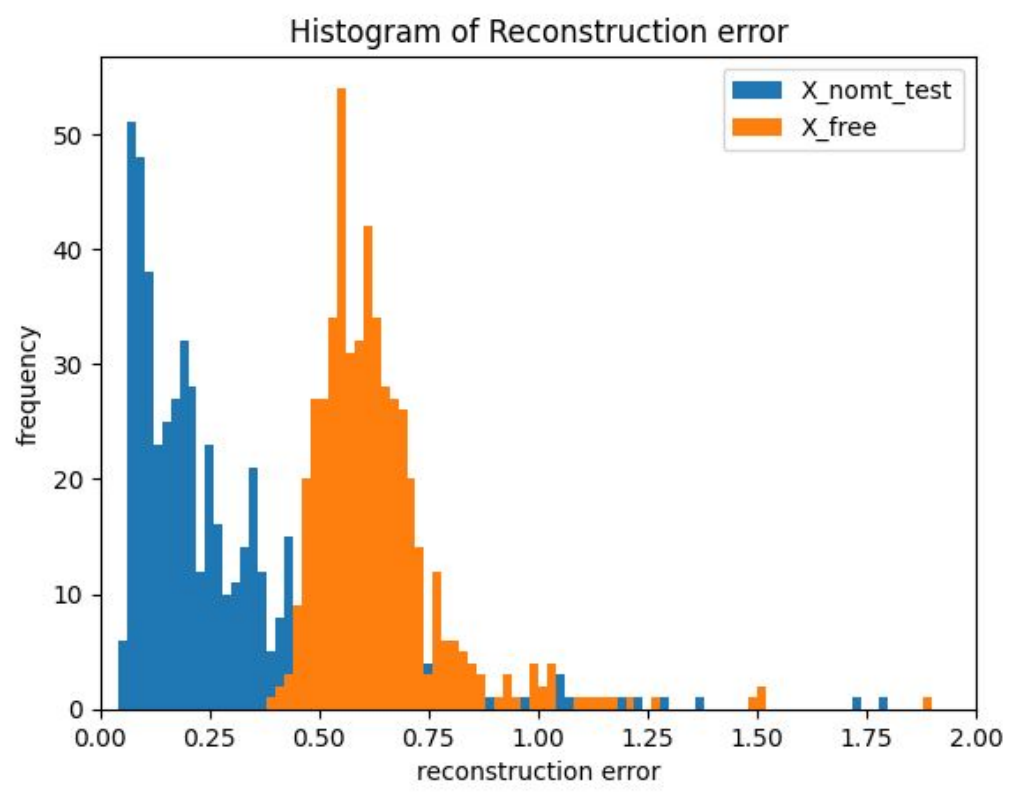
and get the classification accuracies:

- small → 0.8718
- medium → 0.8526
- large → 0.8563

CONVOLUTIONAL AUTOENCODER

Reconstruction Error of : X_nomt_test
Mean = 0.32027975
Median = 0.1996785

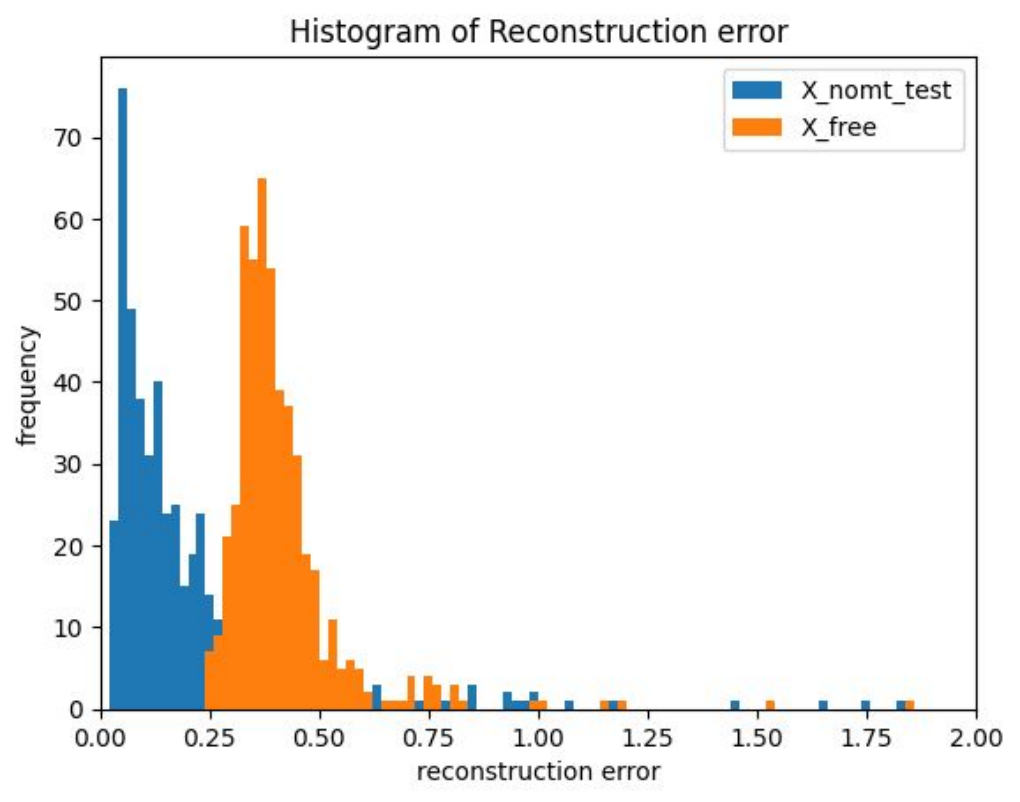
Reconstruction Error of : X_free
Mean = 0.64917845
Median = 0.60304785



CONV. AUTOENCODER - MORE LATENT VARIABLES

Reconstruction Error of : X_nomt_test
Mean = 0.24402161
Median = 0.13725924

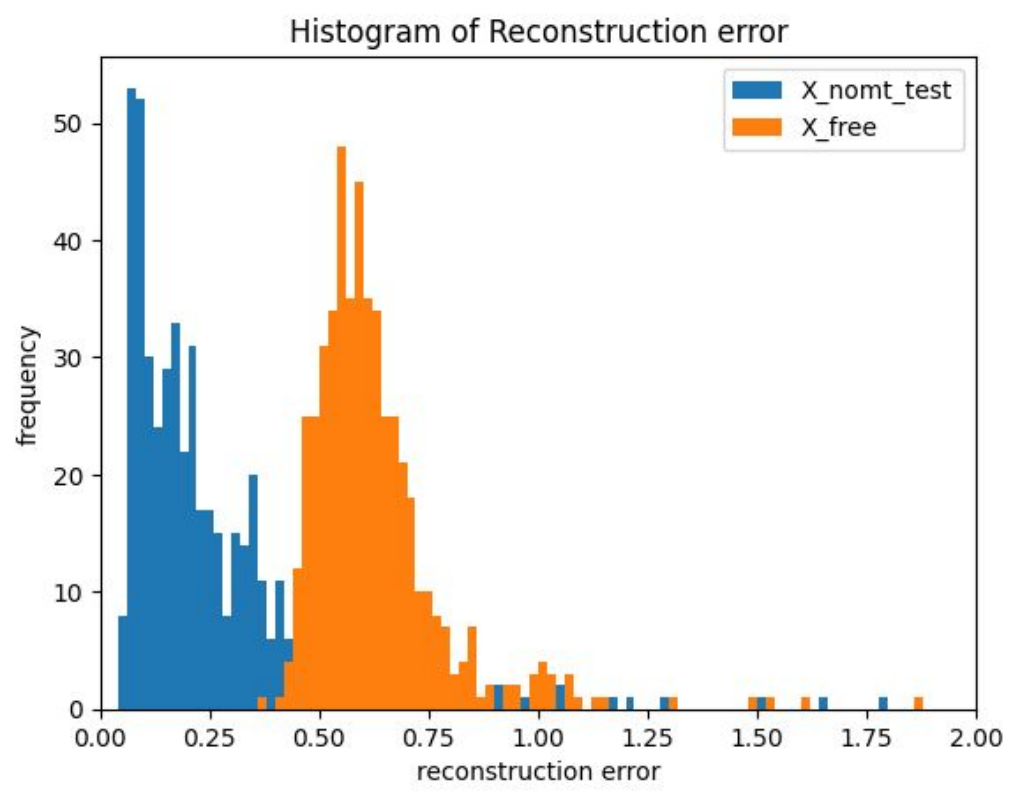
Reconstruction Error of : X_free
Mean = 0.42459446
Median = 0.38303673



CONVOLUTIONAL AUTOENCODER - MORE LAYERS

Reconstruction Error of : X_nomt_test
Mean = 0.31996152
Median = 0.19738552

Reconstruction Error of : X_free
Mean = 0.6446158
Median = 0.5971612



RESULTS - CONVOLUTIONAL AUTOENCODER

Based on the histograms we take the thresholds to be:

- normal → 0.4
- more latent variables → 0.25
- more layers → 0.4

and get the classification accuracies:

- normal → 0.9113
- more latent variables → 0.8841
- more layers → 0.9161

DISCUSSION

- different scale of data poses a problem when using MSE
- does good accuray come from scaling the data instead of “structure” ?
- More Data
- Questions :
  - How to scale in online setting (hopefully we don't have to)

SOURCES

- https://www.compthree.com/blog/autoencoder/