Intelliquench status update

Duc -- Fermilab

Outline

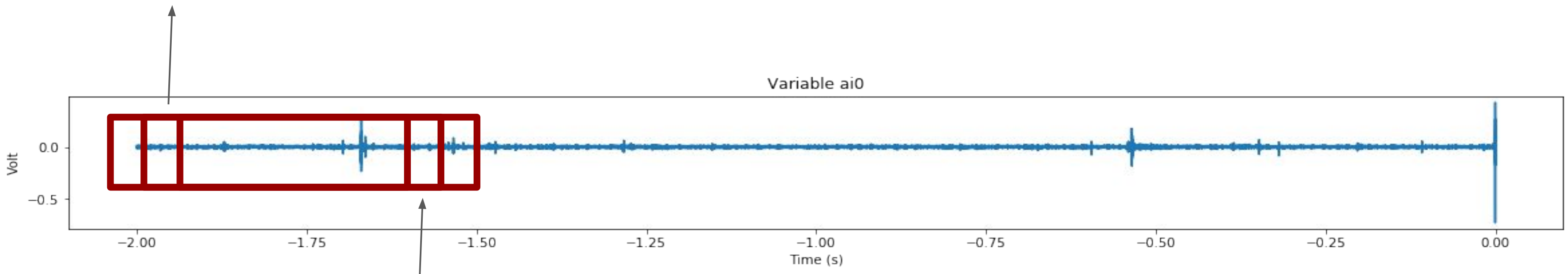
1. Generating statistical features from acoustic data.
    a. Standard deviation
    b. Kurtosis
    c. Skew
2. Predicting quench using Principal Component Analysis (PCA) and Mahalanobis distance.
    a. PCA
    b. Mahalanobis distance
3. Predicting quench using autoencoder deep neural network (DNN).
    a. Deep Neural Network
    b. Auto-encoder
    c. Reconstruction loss
4. Summary & Outlook

1. Generating statistical features from acoustic data

- There are five acoustic sensors around the magnet.
- Acoustic data rate: **100kHz**
- We use a rolling window to calculate statistical features across all the acoustic data.
- The statistical features are: Variance, Kurtosis, Skew.

Calculating statistical features using rolling window.

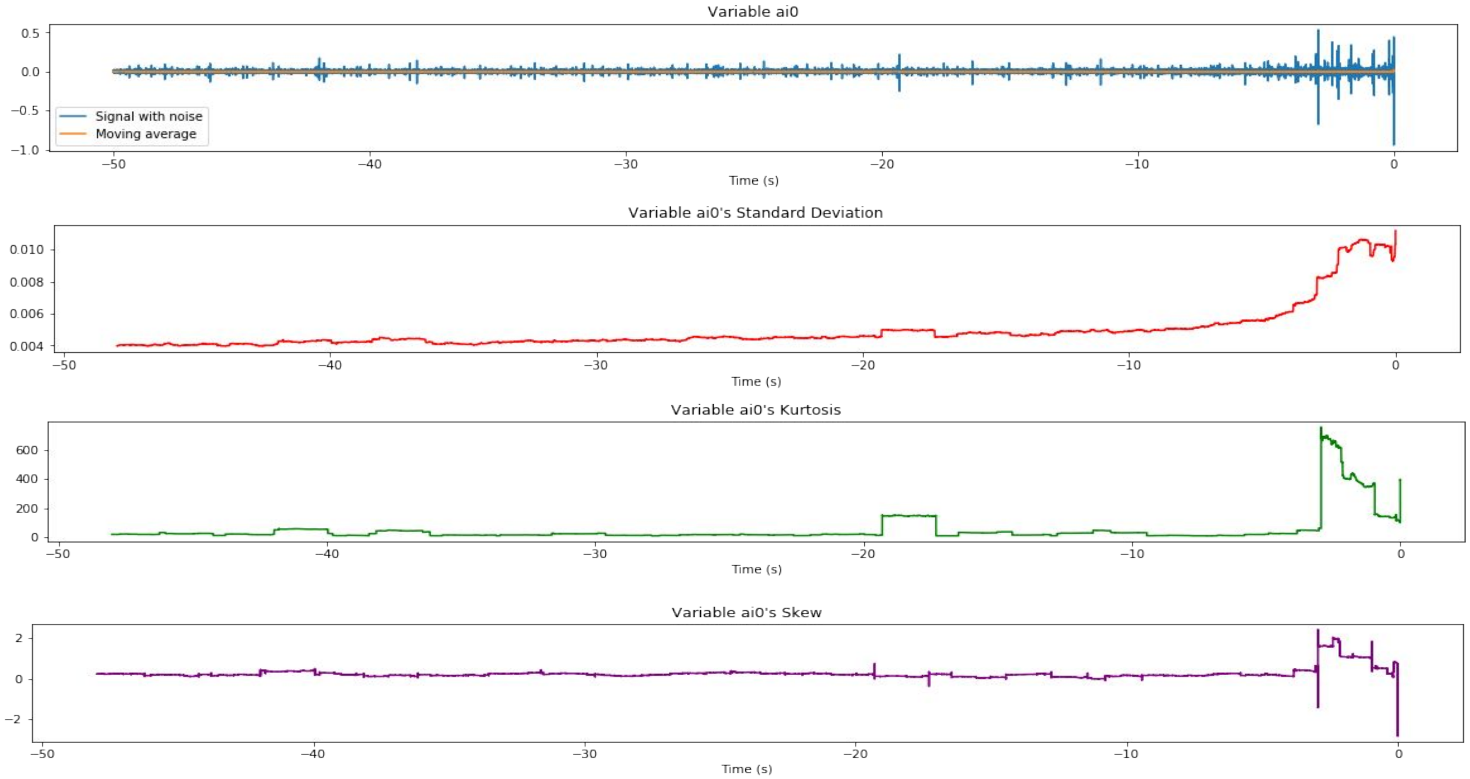
Calculate the raw data's distribution and extract statistical features



Shift over by a step. (we are experimenting with steps of 100us)

Statistical value calculated in a window is assigned to the time of the last data point.

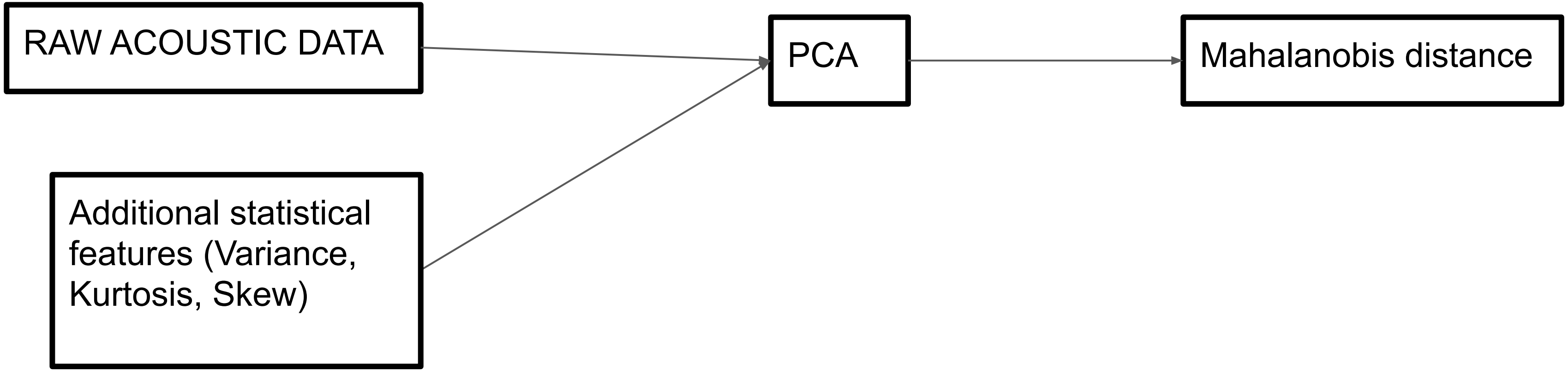
Combining them all!



Each channel's data will thus be a matrix of dimension: (# of data points, 4)

2. Predicting quench using Principal Component Analysis and Mahalanobis distance.

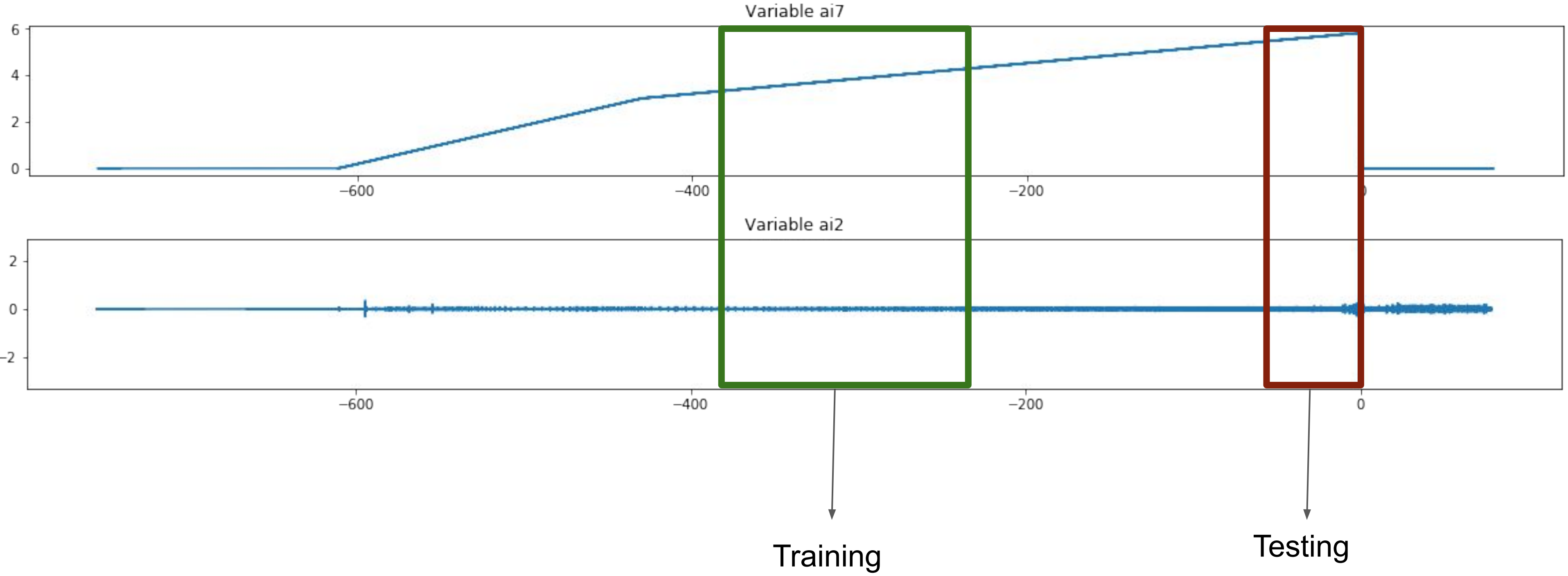
Procedure


RAW ACOUSTIC DATA
Additional statistical features (Variance, Kurtosis, Skew)
PCA
Mahalanobis distance

Principal Component Analysis (PCA)

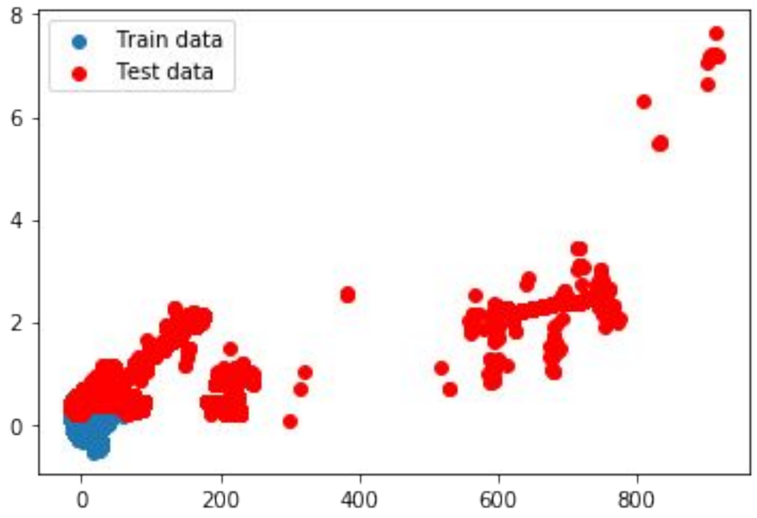
- Intuition: PCA is a method for **compressing** a lot of data into something that **captures the essence of original data** by reducing its dimensionality.
- Principal components are the axes that span the **most variations** of the data (so the axes that does not have a lot of variations can be eliminated).
- Example: we watch movies on 2D screen but can still infer the same information about a 3D world.

1. Choosing training and testing data



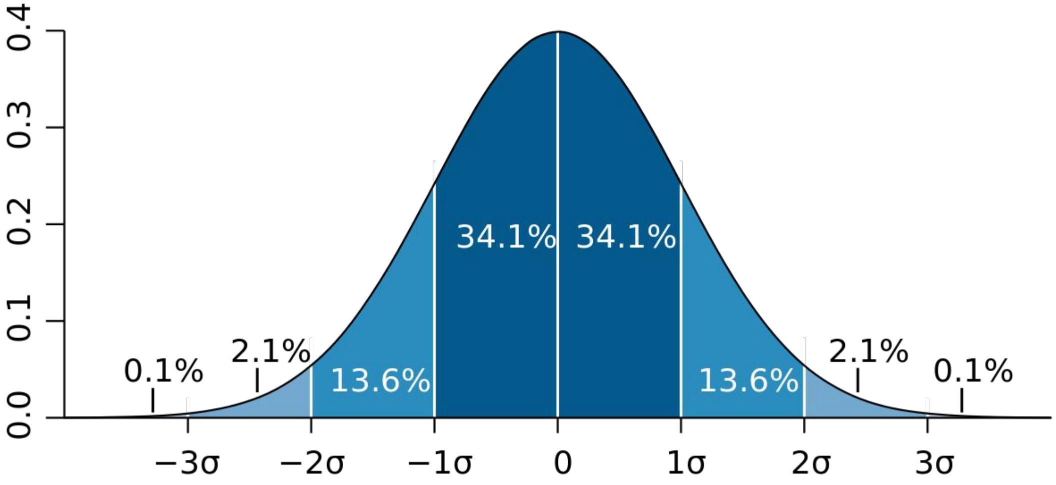
2. Use PCA to compress dimensions.

- The data's dimensionality was reduced from 4 to 2. In other words, the matrix's shape changes from (# data points, 4) -> (# data points, 2)

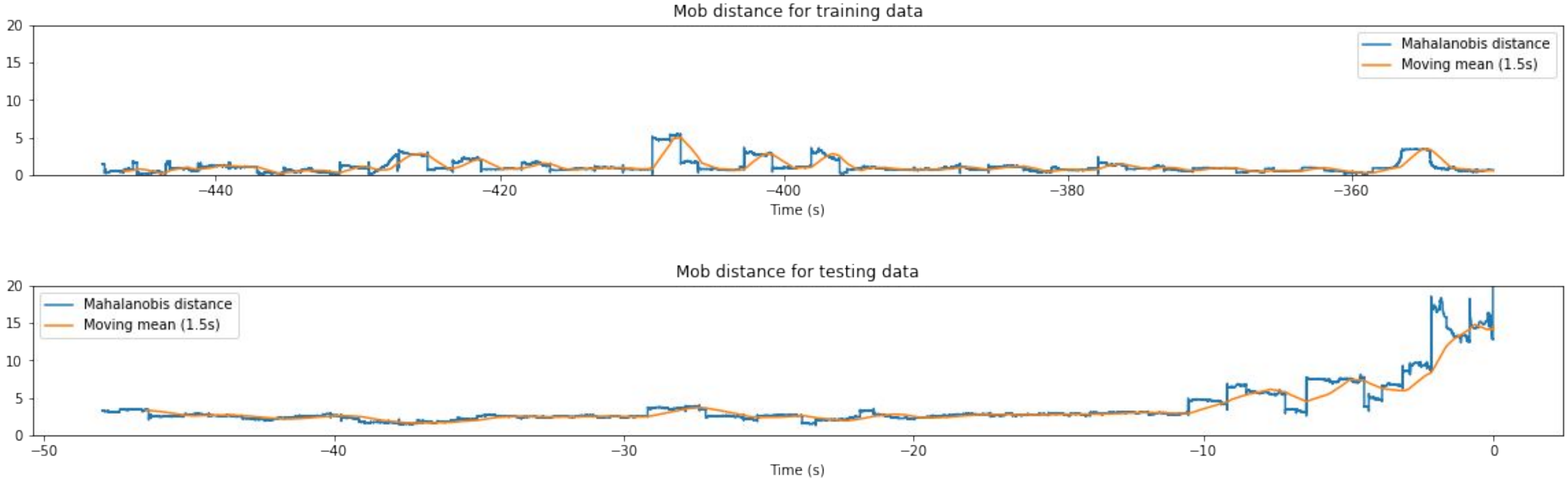


3. Mahalanobis distance

- Intuition: How many standard deviations away a data point from its mean of the distribution.



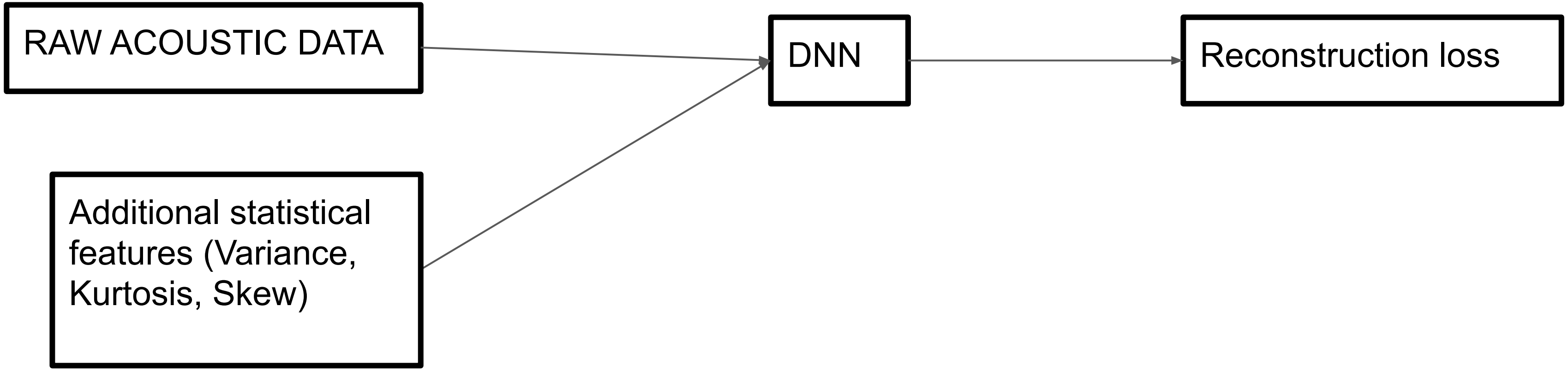
4. Visualize the results



Results from training on a single sensor's data

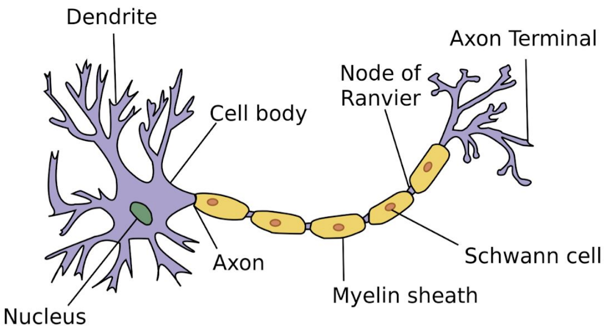
3. Predicting quench using Deep Neural Network as Auto-encoder.

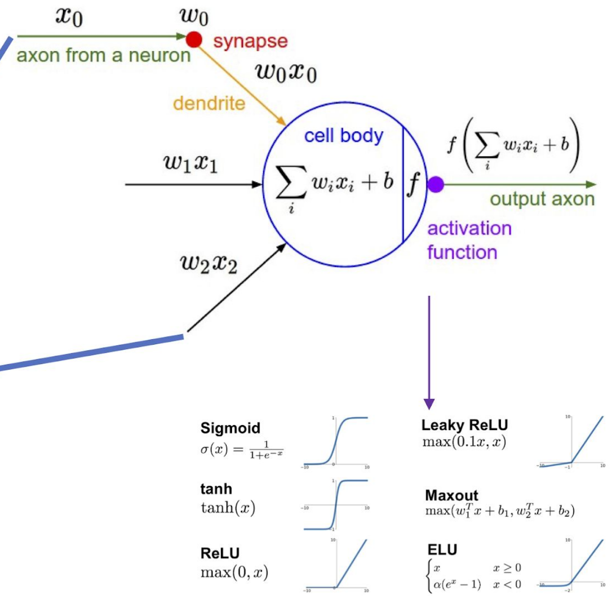
Procedure

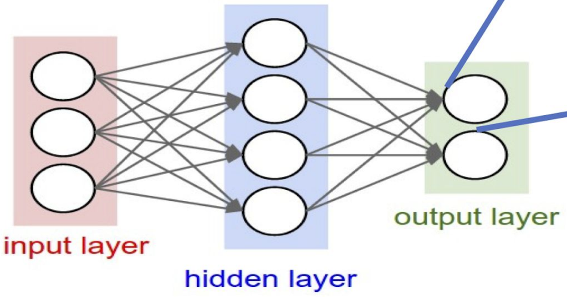

RAW ACOUSTIC DATA
Additional statistical features (Variance, Kurtosis, Skew)
DNN
Reconstruction loss

Deep Neural Networks

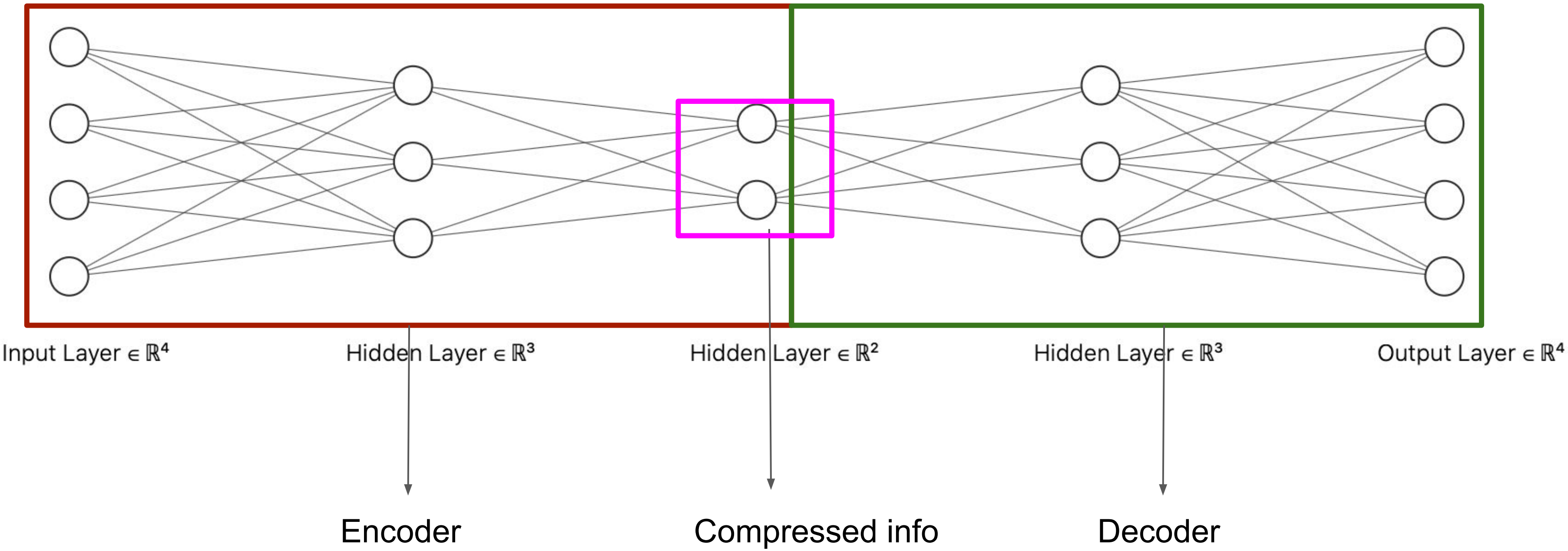
- Fully connected architecture
- Each **input** multiplied by a **weight**.
- **Weighted values** are summed, **Bias** is added.
- Non-linear **activation function** is applied
- Trained by varying the **parameters** to minimize a loss function (quantifies how many mistakes the network makes)







Deep neural network as auto-encoder

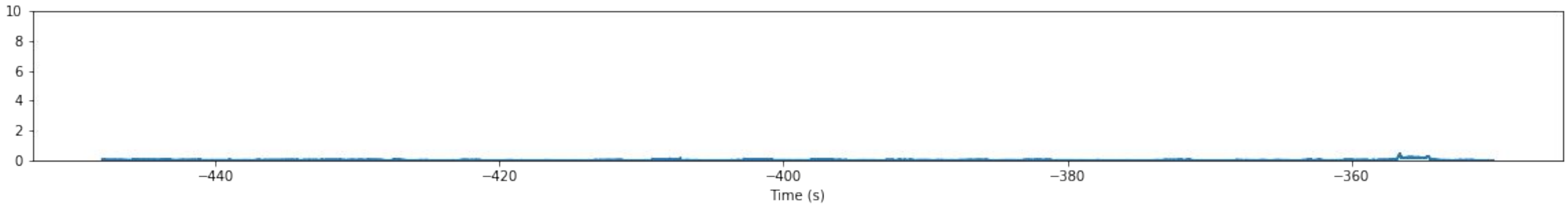


Deep neural network as auto-encoder

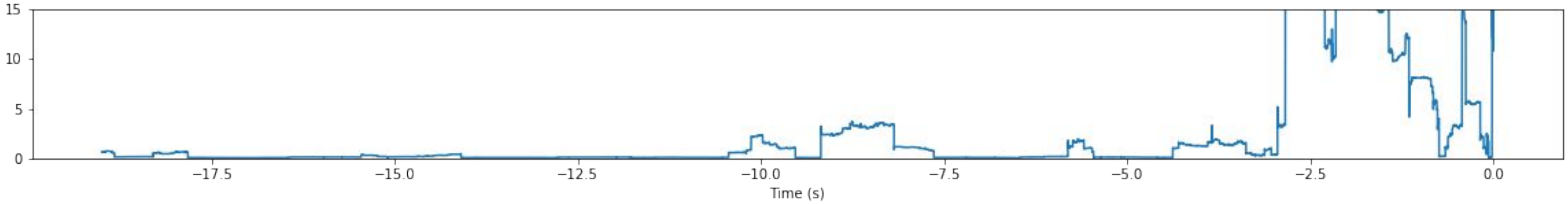
- The **reconstructed output** was compared with the **original input** using **mean square error**.
- We try to **minimize** this error on the **training data**, and then **calculate the error on the testing data**.
- The idea is similar to PCA method, but this takes into account relationships between the inputs and also add non-linearity to it.
- DNN is also less susceptible to noise that it has already seen during training.

Reconstruction loss on training and testing set

Training data



Testing data



All sensors' data are combined

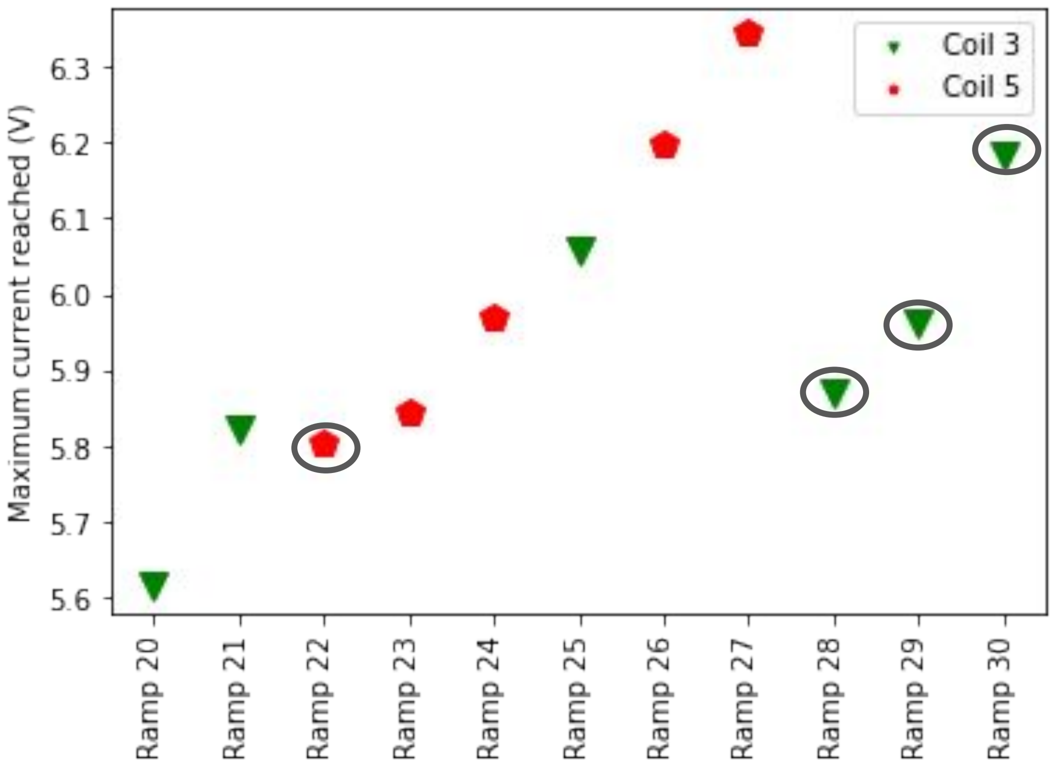
Summary & Outlook

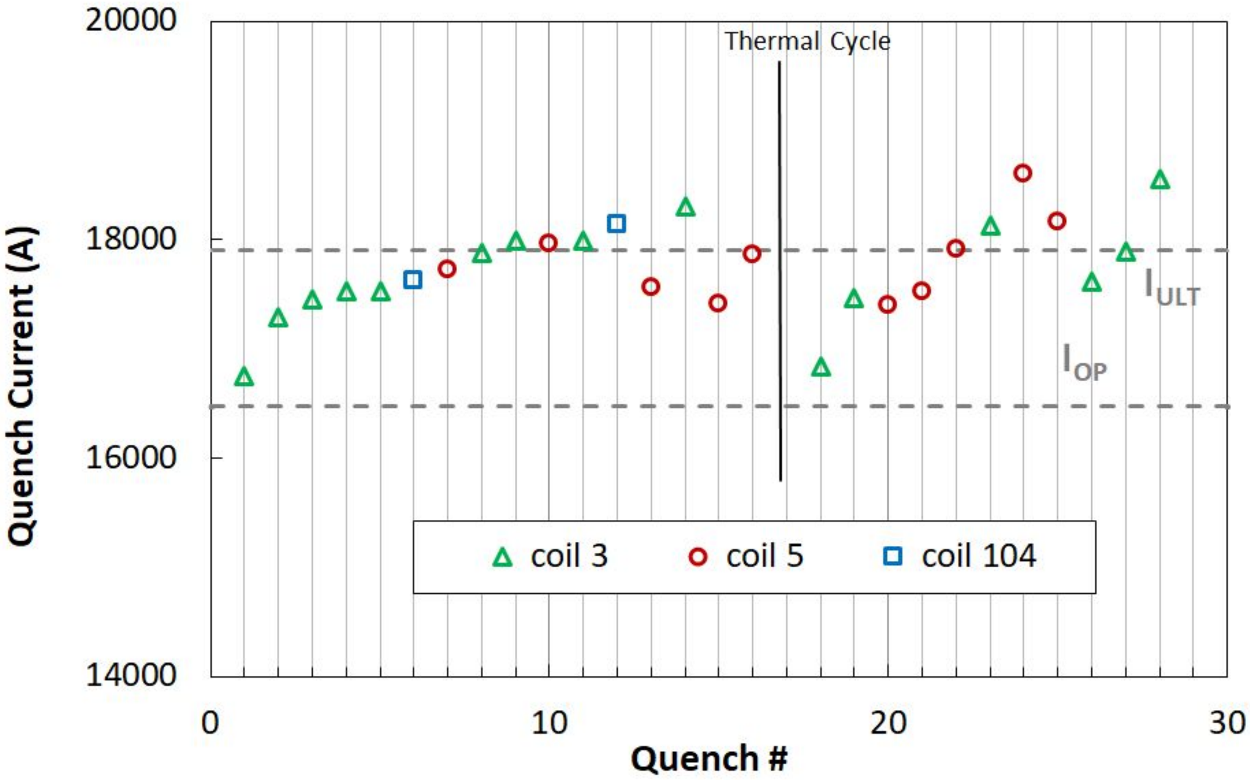
1. Some promising machine learning methods to try on acoustic data for early quench detection.
2. There are still lots of work to do, including figuring out how exactly the anomaly points are related to the quench, or whether they are related at all.
3. Have to take into account magnet training's effects on acoustic data, and whether these effects are different from the quench's precursors.

Back-ups

1. Magnet training effects’ on acoustic data

Maximum current reached in each quench

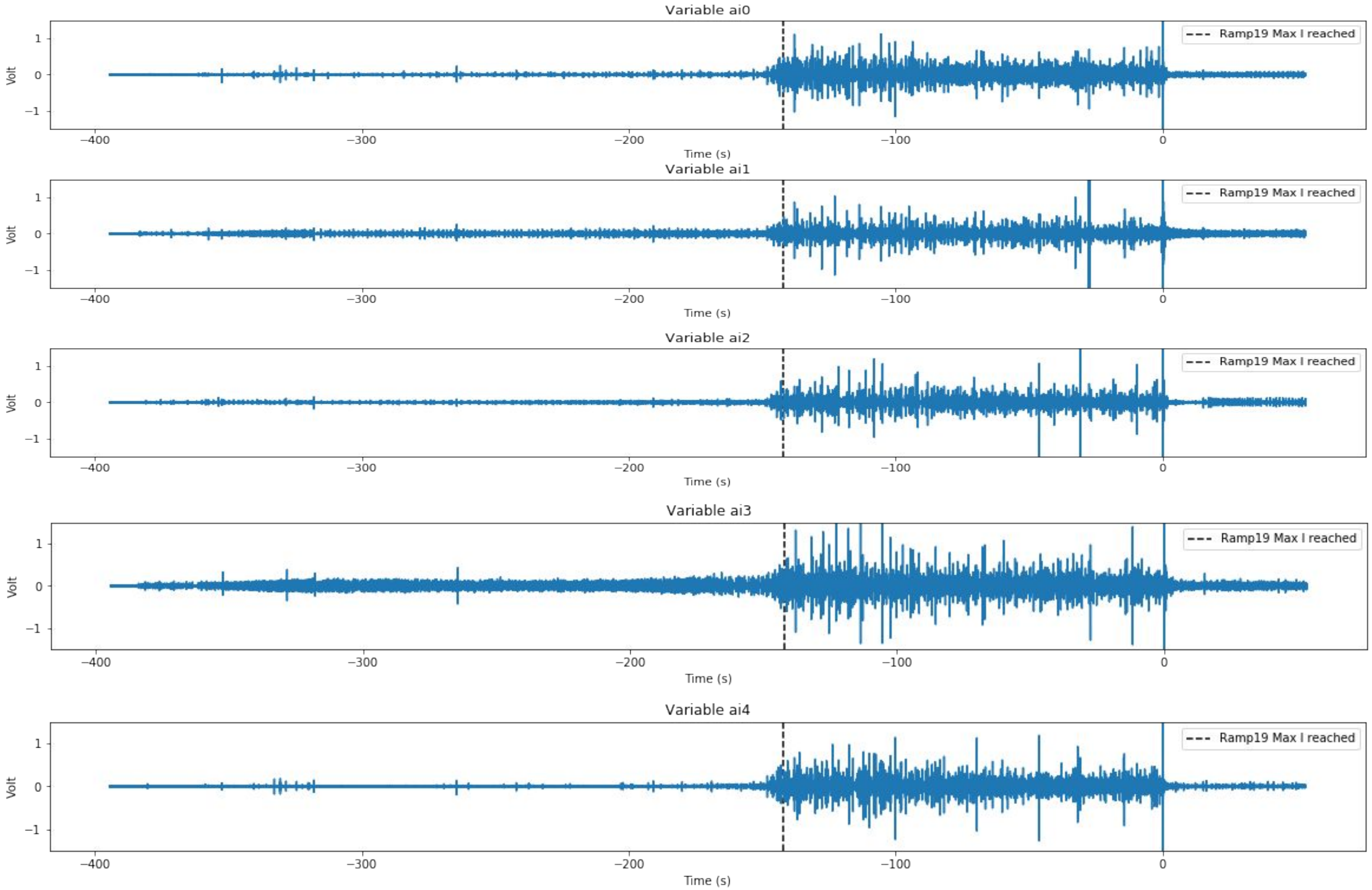



Thermal Cycle
Quench Current (A)
20000
18000
16000
14000
0
10
20
30
Quench #
I_ULT
I_OP
coil 3
coil 5
coil 104

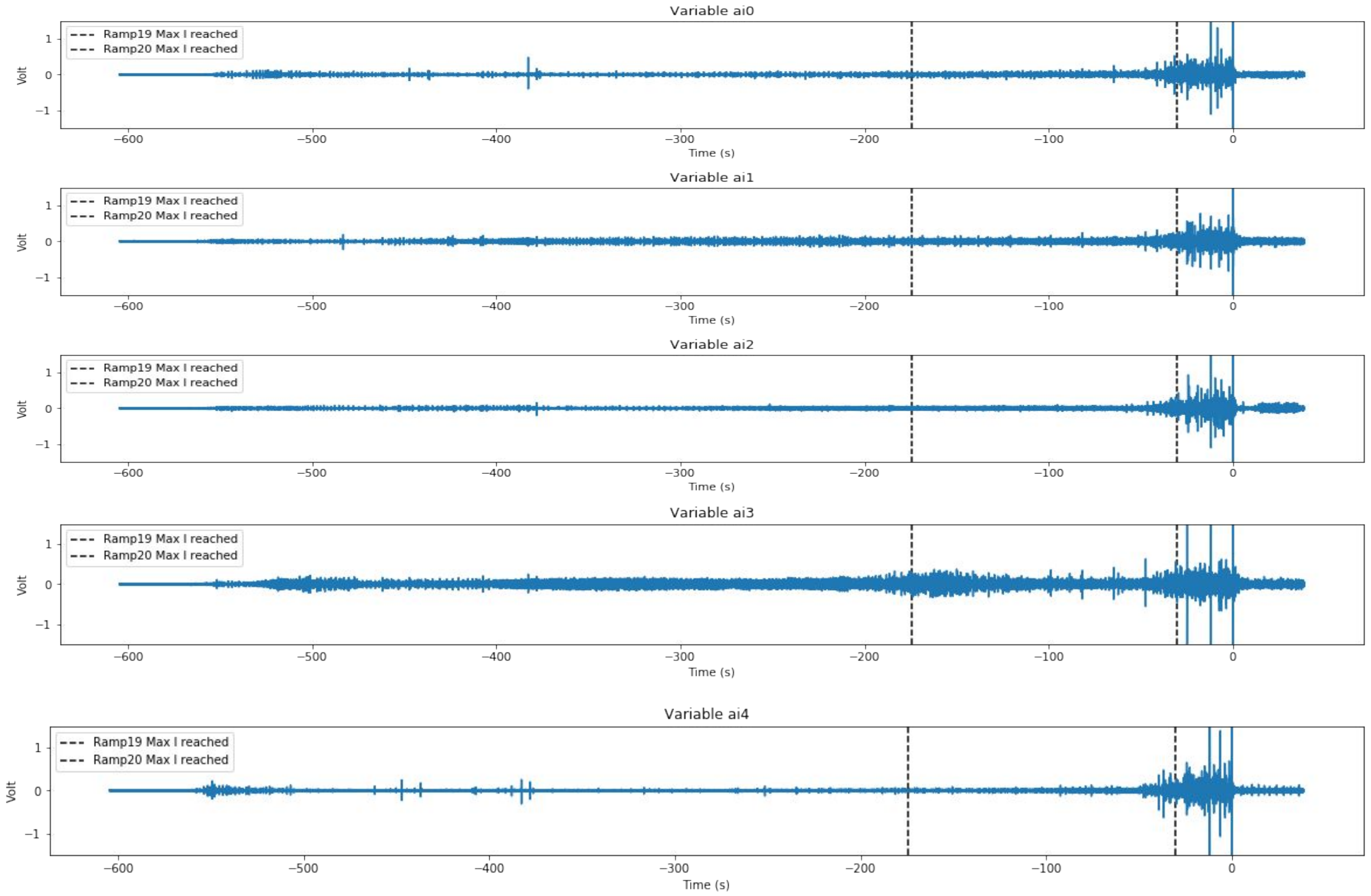
MQXFS1D -- Ramp 19 -- 24.May.2018 15:24

- Max current (in Volt): 4.672324V
- A trip, not a quench

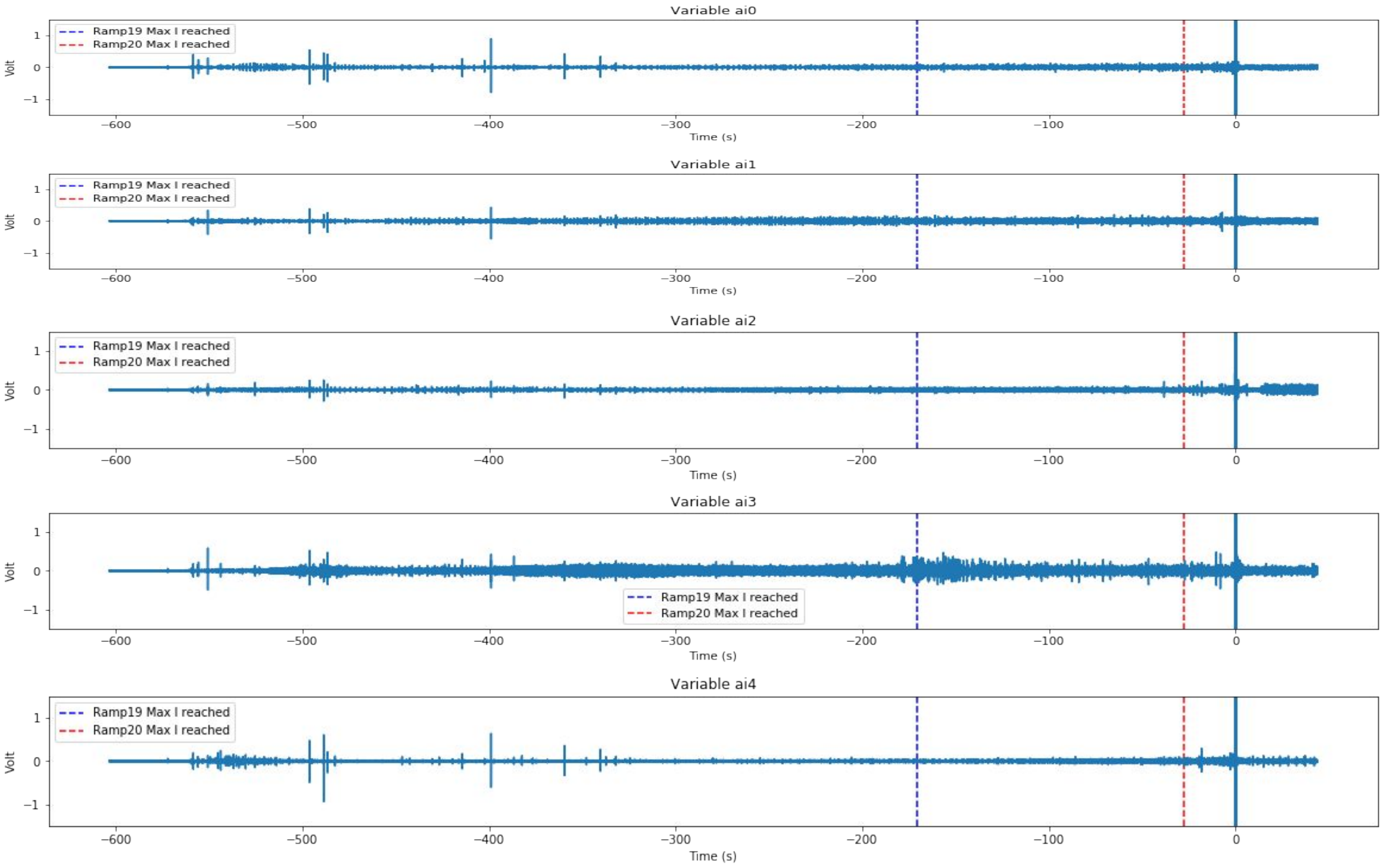
Ramp 20 -- Acoustic data with max current from 19



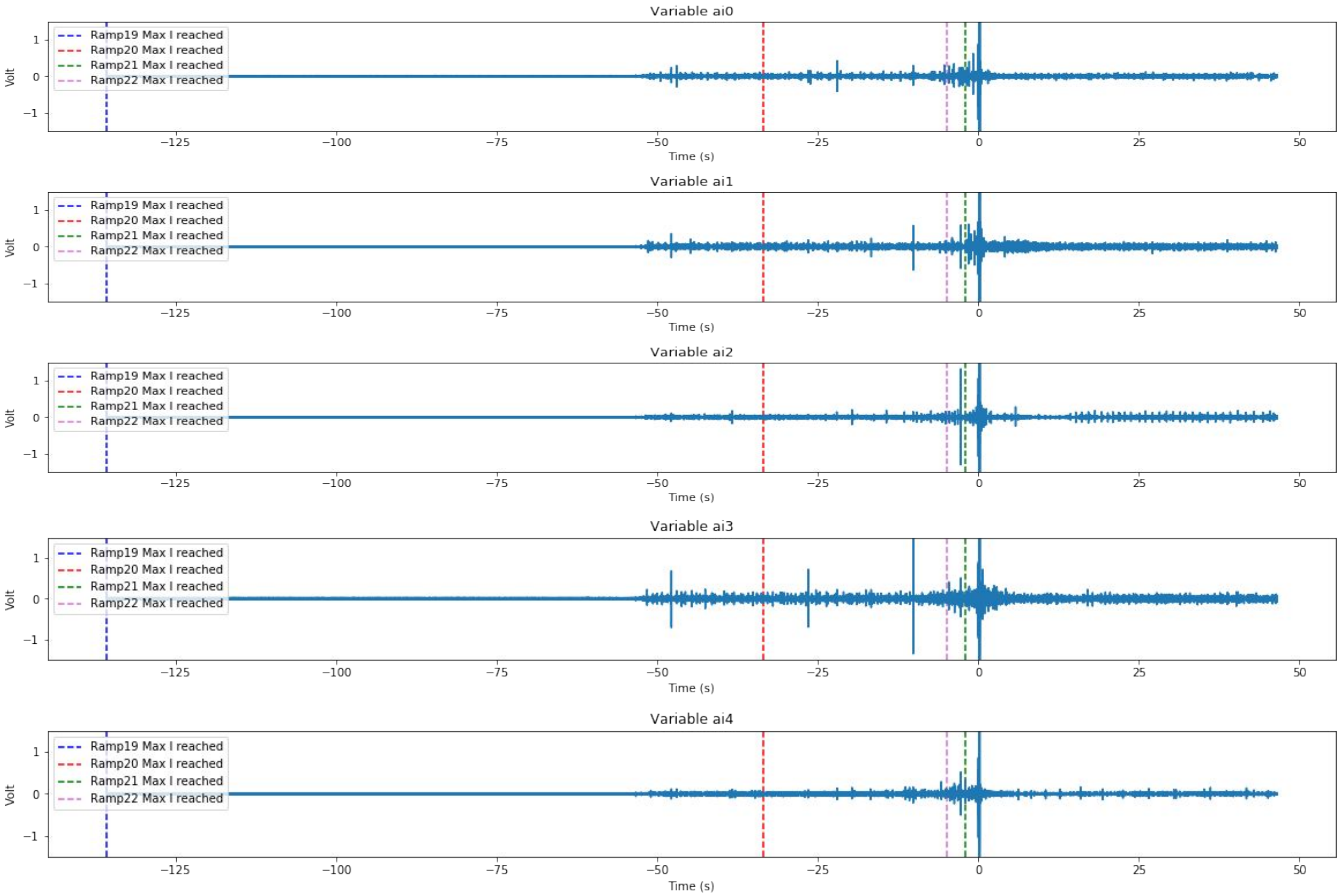
Ramp 21 -- Acoustic data with max current reached



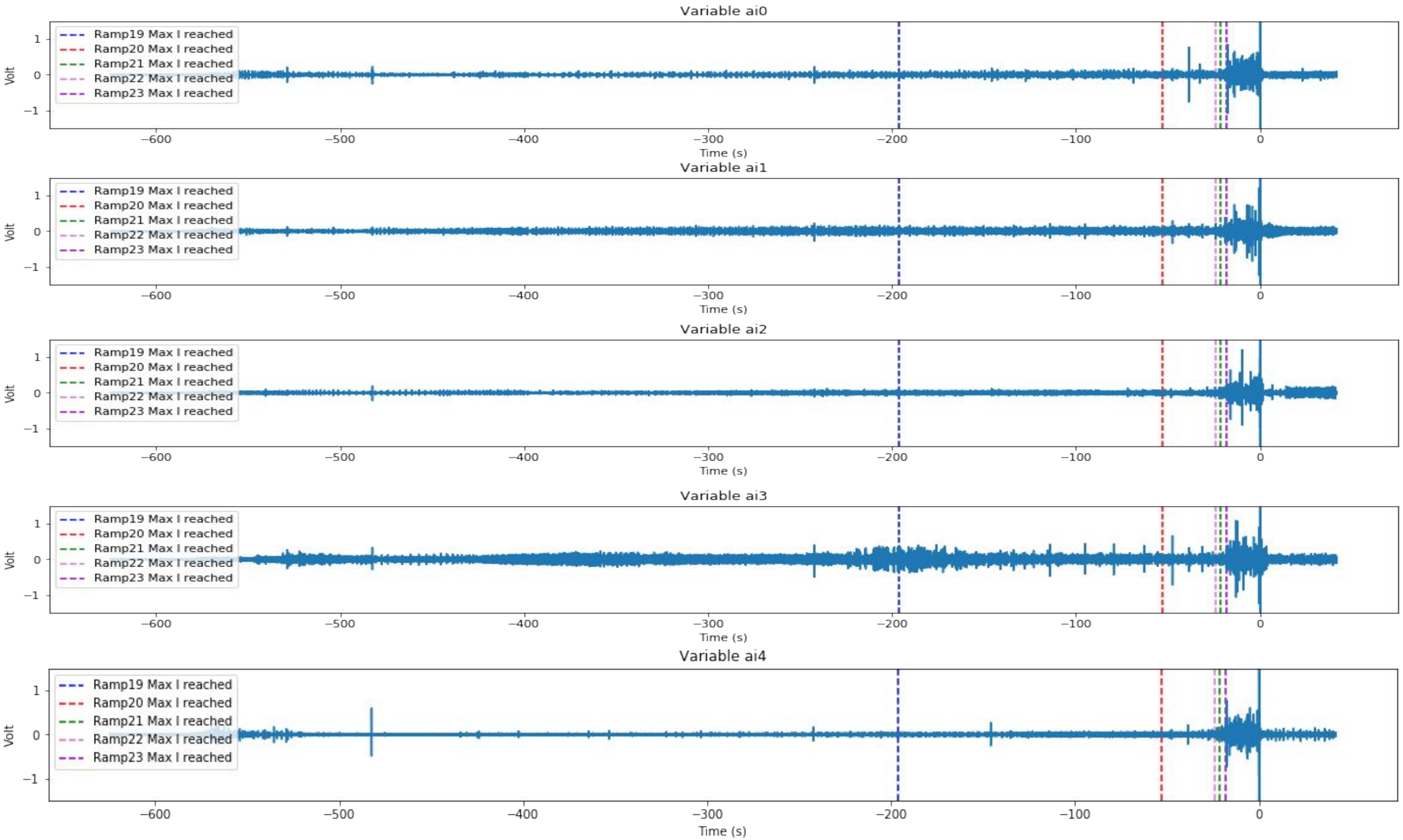
Ramp 22 -- Acoustic data with max current reached



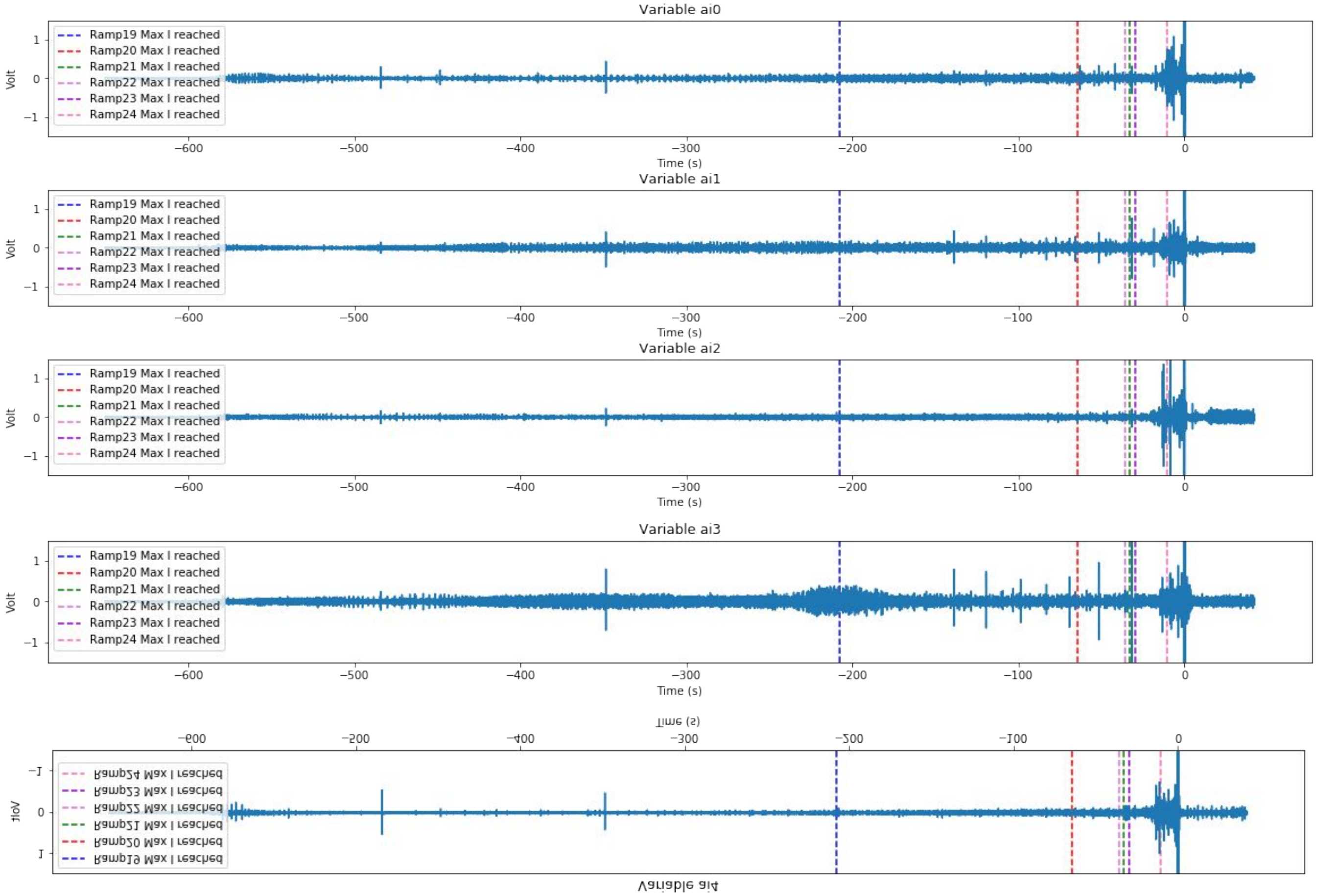
Ramp 23 -- Acoustic data with max current



Ramp 24 -- Acoustic data with max current



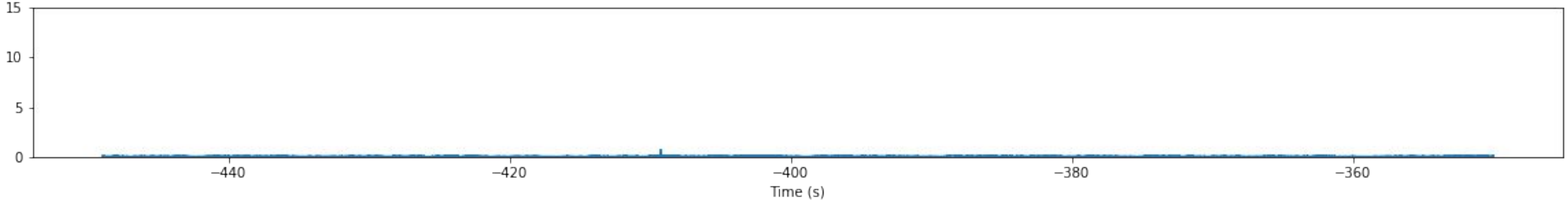
Ramp 25 -- Acoustic data with max current



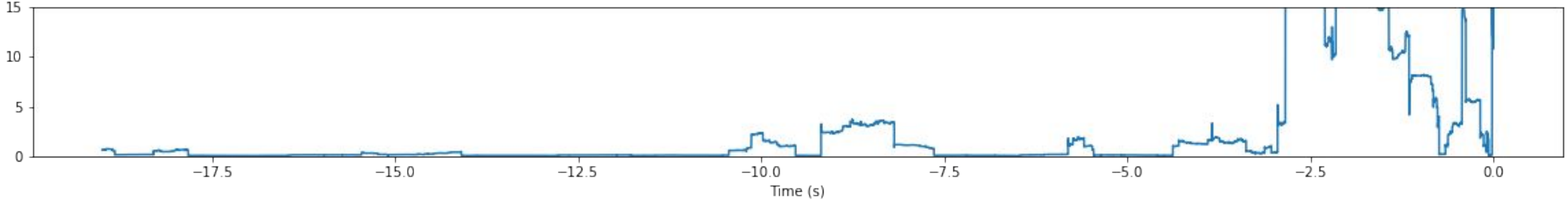
Potential effects on reconstruction loss

Ramp4 - Window size: **1s** - Steps: **100us**

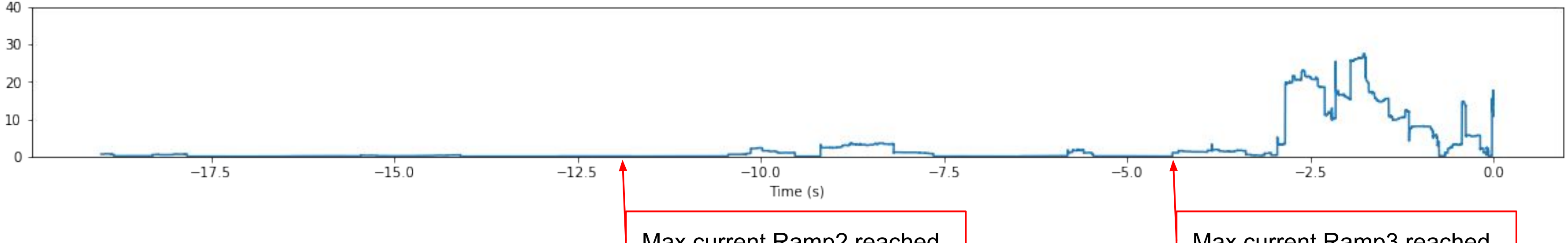
Training loss:



Testing loss:



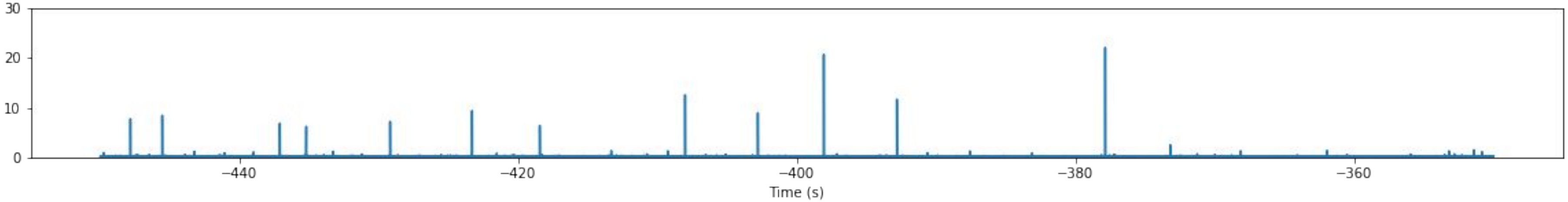
Test loss (**Changed y scale**)



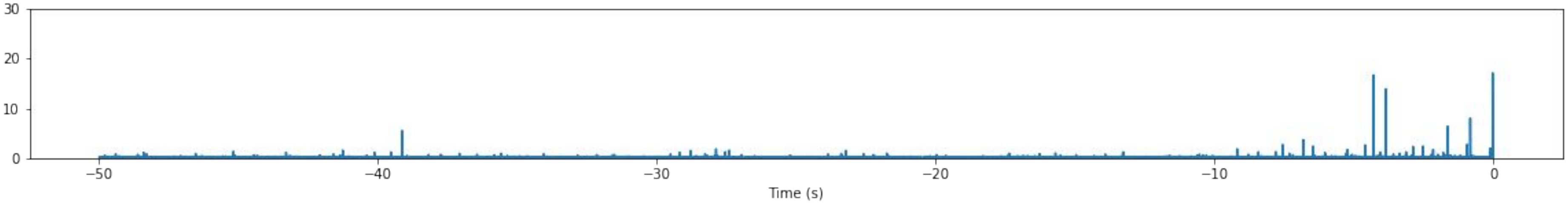
1. Different window size + steps for single sensor

Ramp4 - Window size: **10ms** - Step: **100us**

- Train loss:

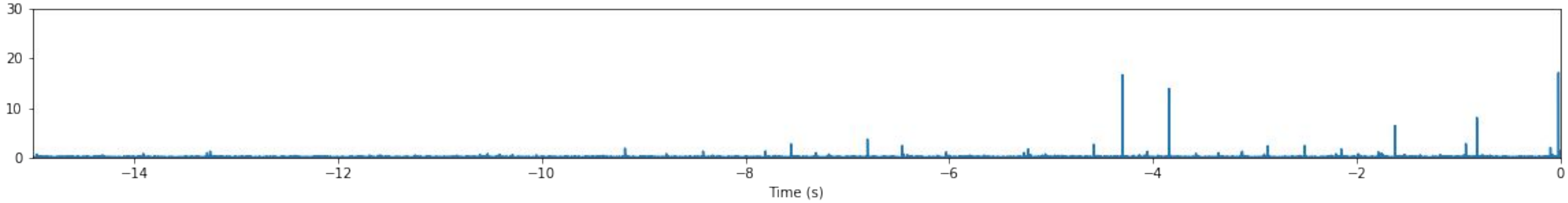


- Test loss (-50,0)s:

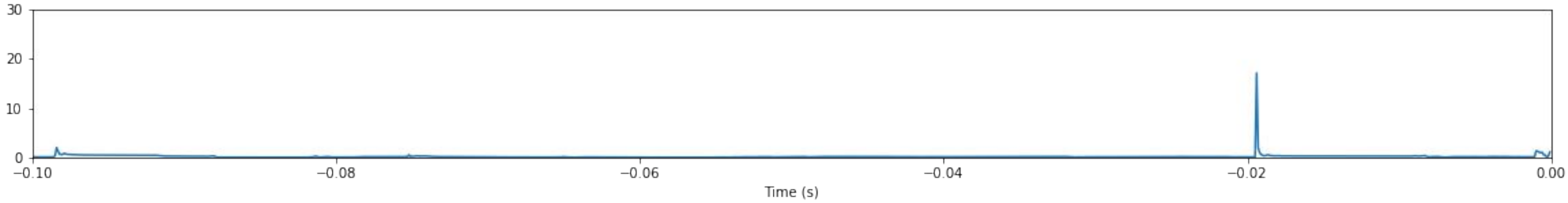


Ramp4 - Window size: **10ms** - Step: **100us**

- Test loss (-15,0)s:

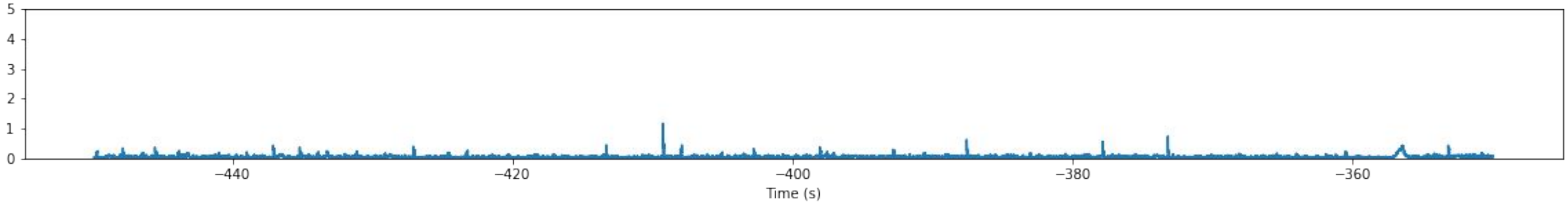


- Test loss (-100,0)ms:

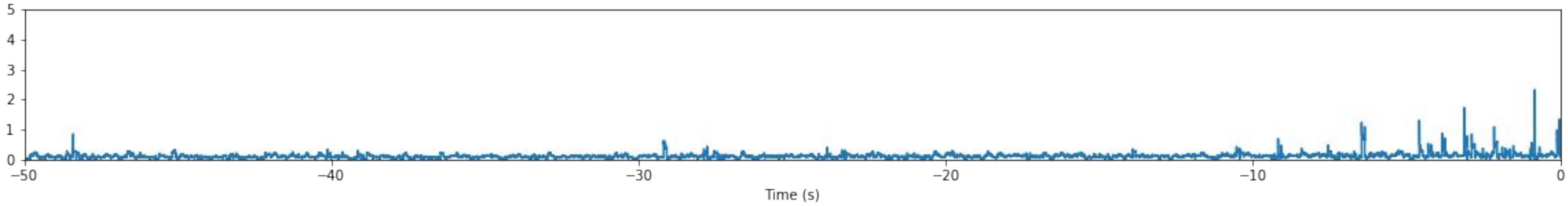


Ramp 4 -- Window size: **100ms** -- Steps: **100us**

- Train loss:

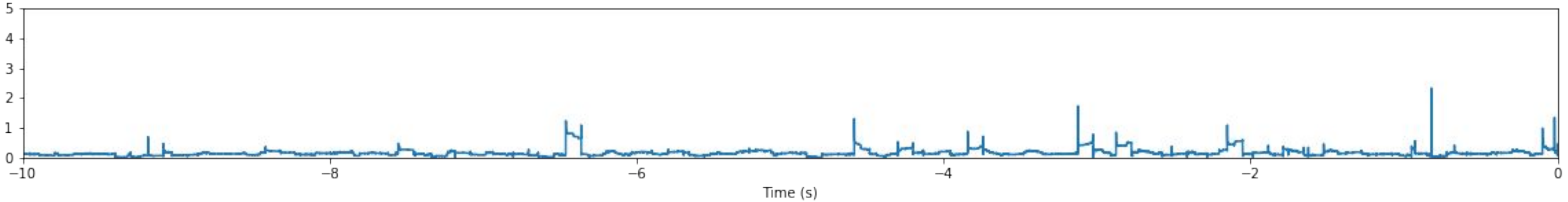


- Test loss:

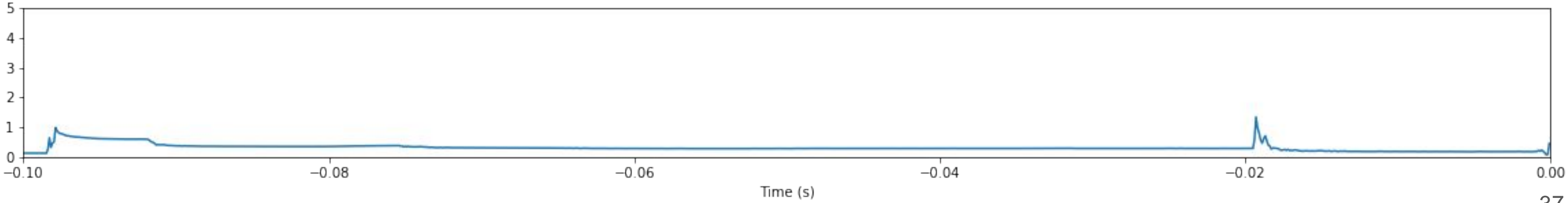


Ramp 4 -- Window size: **100ms** -- Steps: **100us**

- Test loss (-10,0)s:

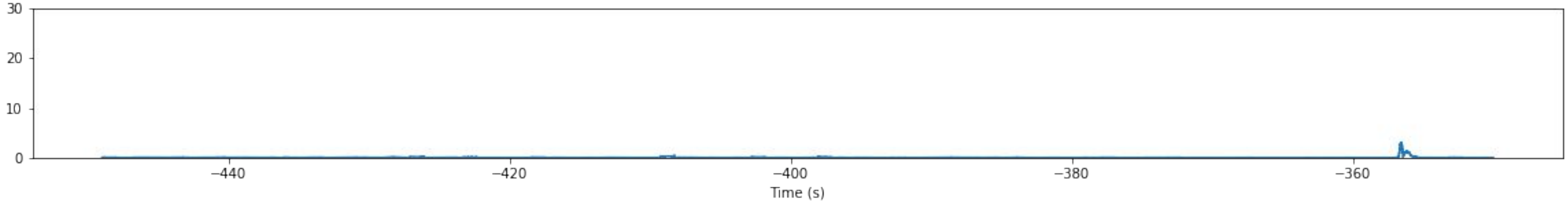


- Test loss (-100,0)ms:

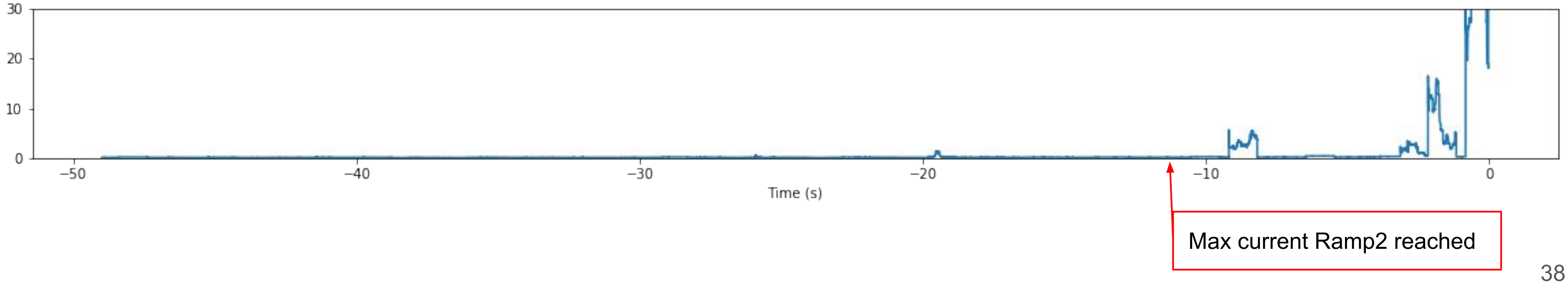


Ramp4 - Window size: **1s** - steps: **100us**

Train loss:

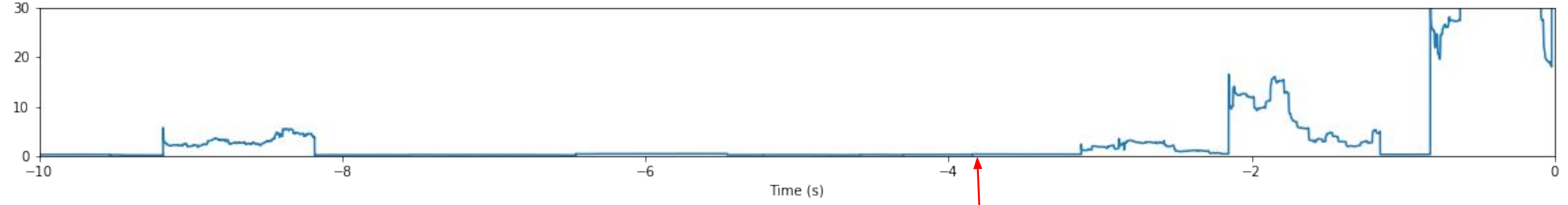


Test loss



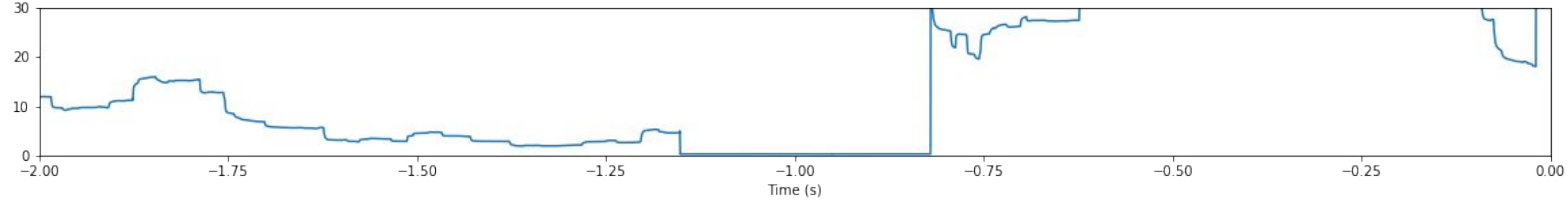
Ramp4 - Window size: **1s** - steps: **100us**

Test loss (-10,0)s

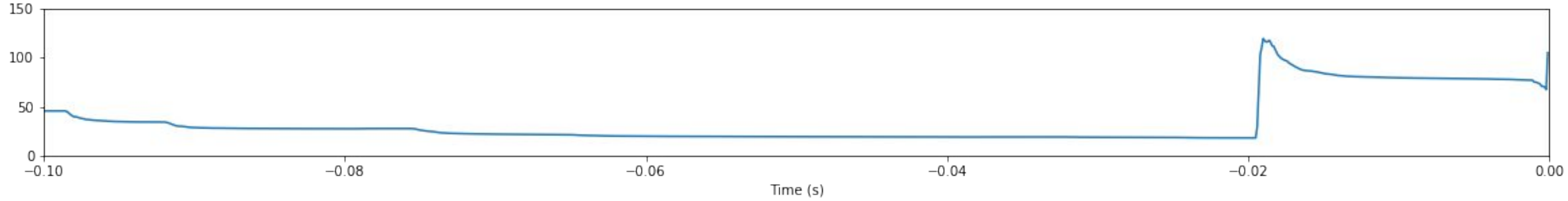


Max current Ramp3 reached

Test loss (-2,0)s

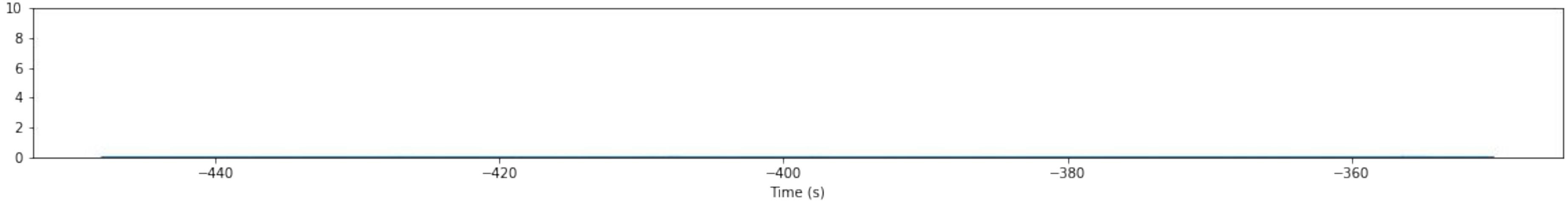


Test loss (-100,0)ms (**Y axis scale changed**)



Ramp4 - Window size: **2s** - steps: **100us**

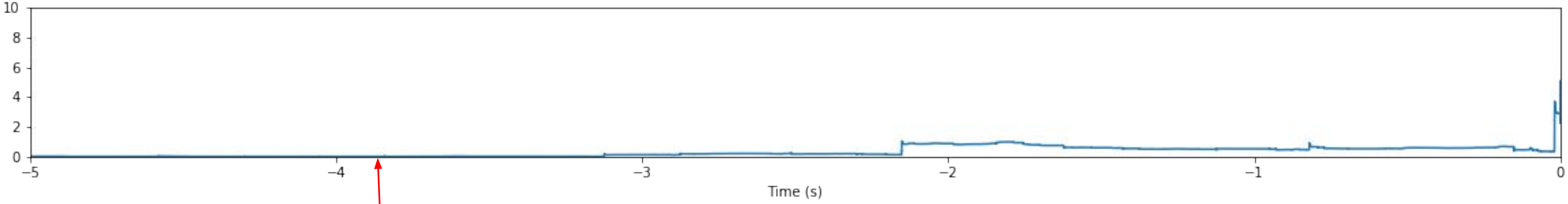
Training loss:



Test loss (-50,0)s:

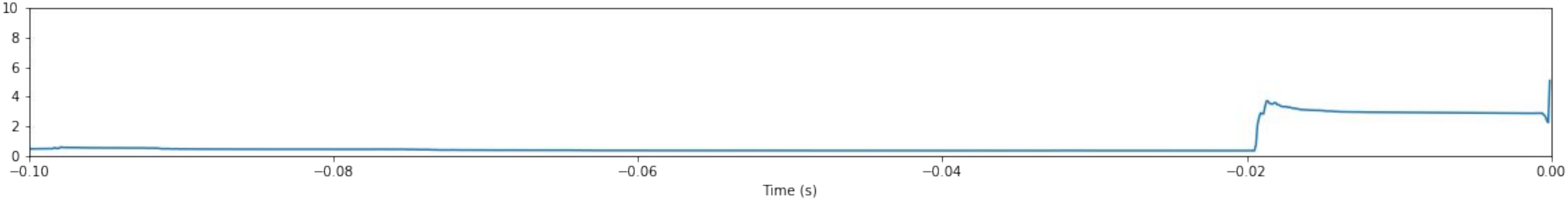
Ramp4 - Window size: **2s** - steps: **100us**

Test loss (-5,0)s:



Max current Ramp3 reached

Test loss(-100,0)ms:



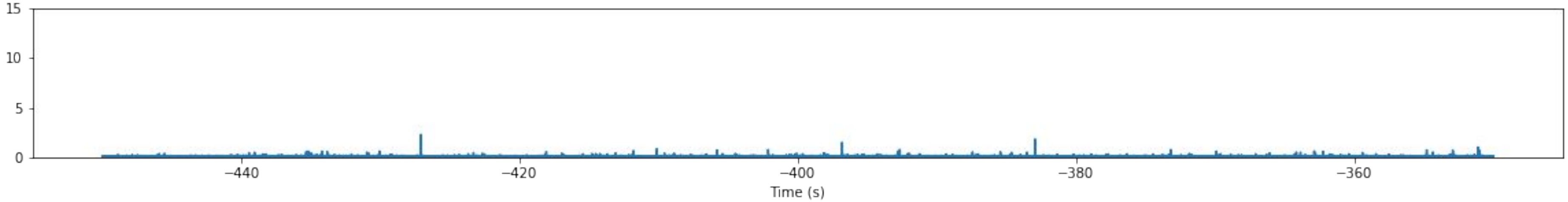
2. Different window sizes and steps on all sensors - Ramp 4

Network architecture used

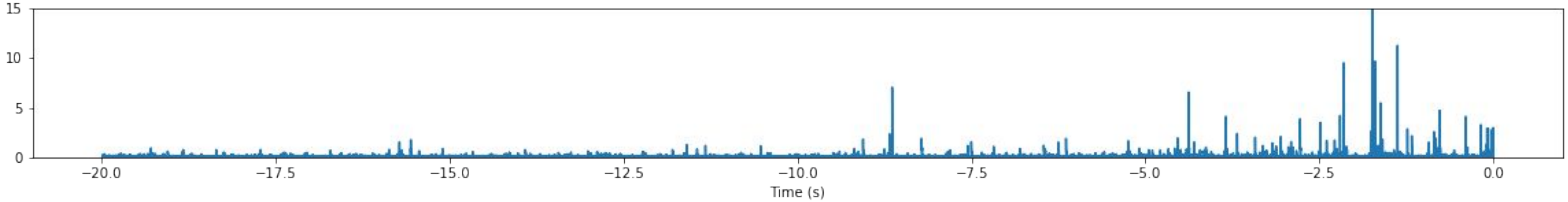
```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense_1 (Dense)              (None, 15)                315
_________________________________________________________________
dense_2 (Dense)              (None, 10)                160
_________________________________________________________________
dense_3 (Dense)              (None, 7)                 77
_________________________________________________________________
dense_4 (Dense)              (None, 10)                80
_________________________________________________________________
dense_5 (Dense)              (None, 15)                165
_________________________________________________________________
dense_6 (Dense)              (None, 20)                320
=================================================================
Total params: 1,117
Trainable params: 1,117
Non-trainable params: 0
_________________________________________________________________
```

Ramp4 - Window size: **10ms** - Steps: **100us**

Training loss:

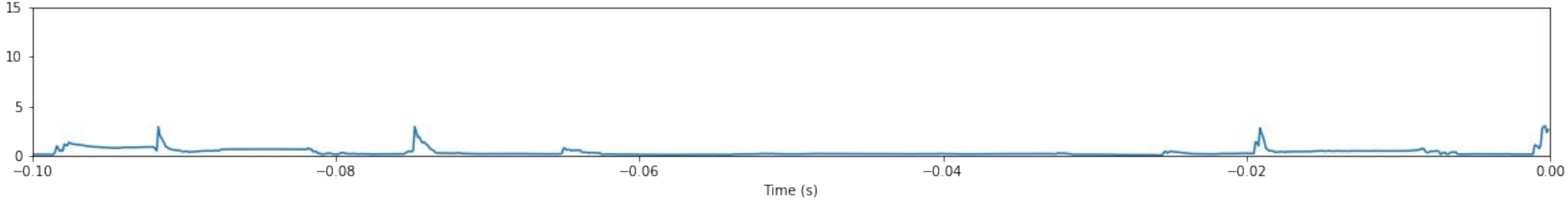


Testing loss (-20,0)s:

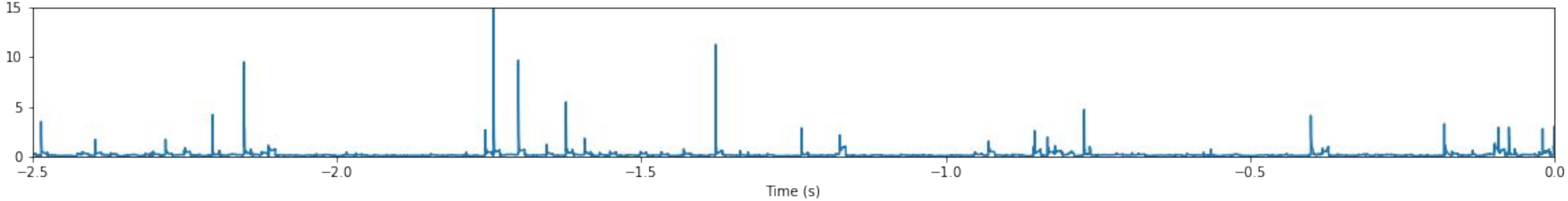


Ramp4 - Window size: **10ms** - Steps: **100us**

Testing loss (-100,0)ms:

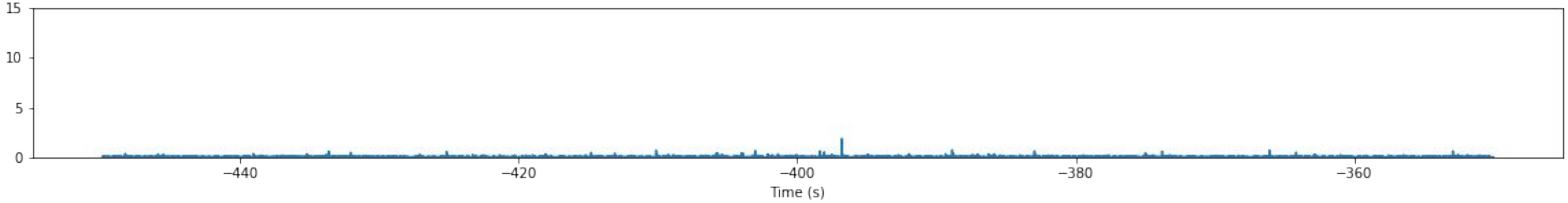


Testing loss (-2.5,0)s:

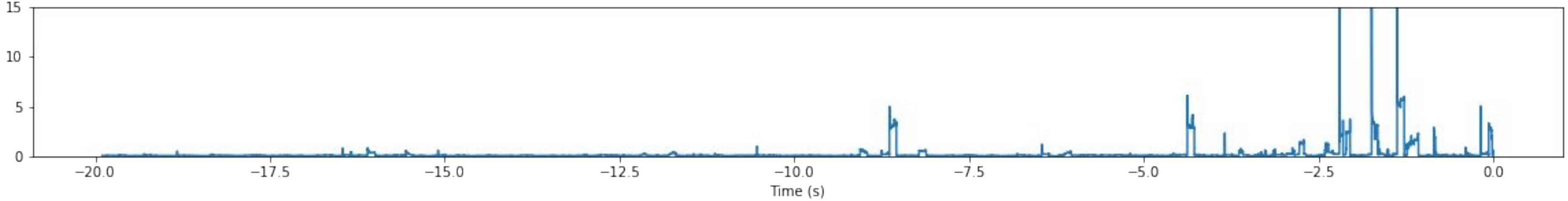


Ramp4 - Window size: **100ms** - Steps: **100us**

Training loss

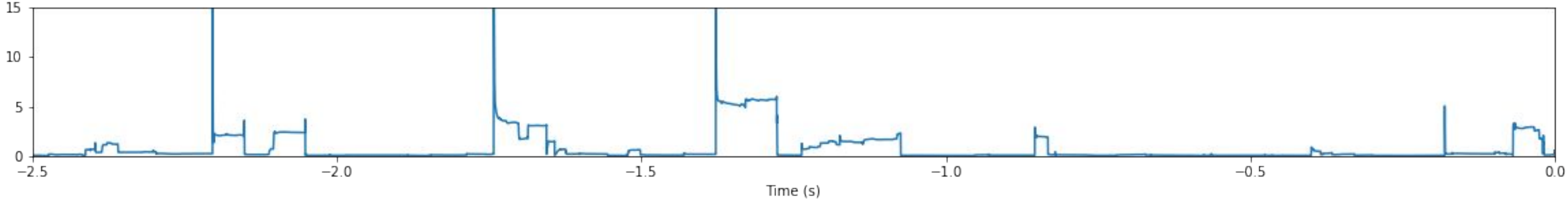


Testing loss (-20,0)

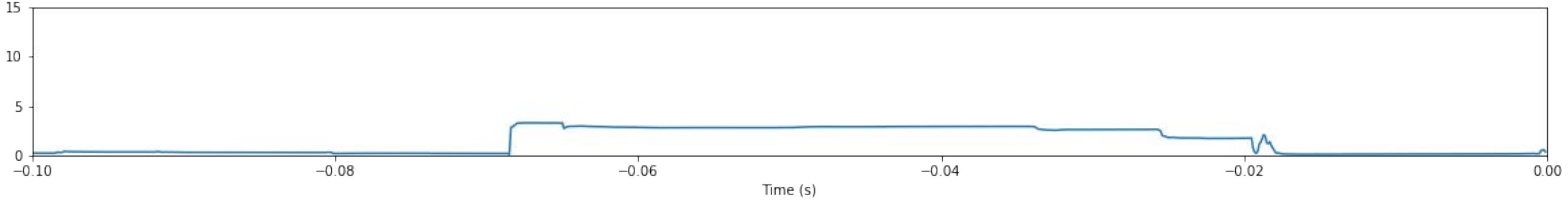


Ramp4 - Window size: **100ms** - Steps: **100us**

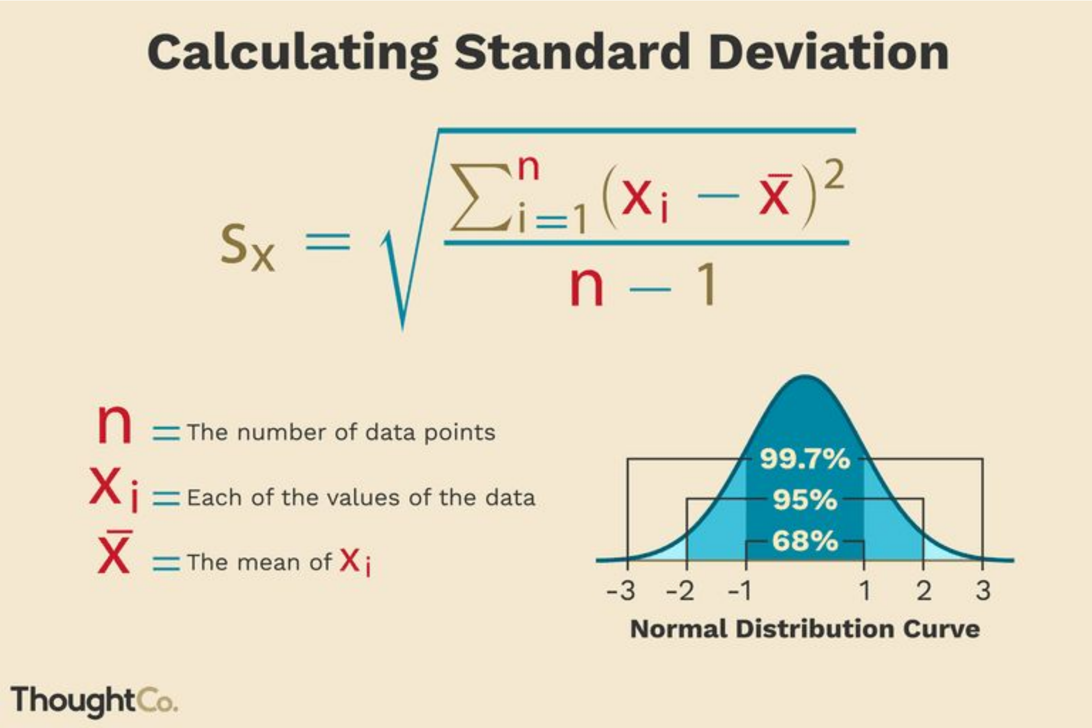
Testing loss (-2.5,0)s



Testing loss (-100,0)ms



Statistical features -- Standard deviation



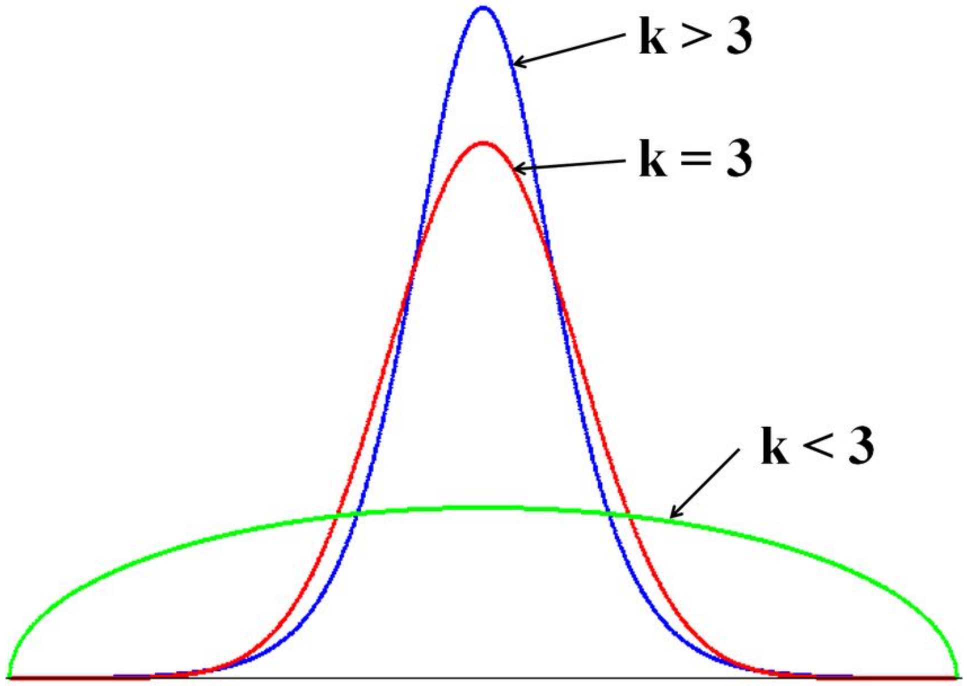
Source

Statistical features -- Kurtosis -- definition

- Intuition: measurement of tailedness of the distribution.
- Kurtosis of normal distribution = 3
- Kurtosis decreases as the tailedness becomes lighter.
- If kurtosis < 3: distribution produces fewer and less extreme outliers than does the normal distribution.
- So the higher the kurtosis, there are more extreme outliers in the distribution.
- Formal formula:

$$\mathrm{Kurt}[X] = \mathrm{E}\left[\left(\frac{X-\mu}{\sigma}\right)^4\right] = \frac{\mathrm{E}[(X-\mu)^4]}{(\mathrm{E}[(X-\mu)^2])^2}$$

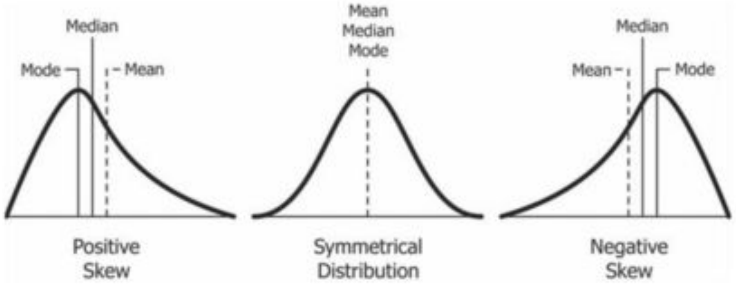
Statistical features -- Kurtosis -- example



Source

Statistical features -- Skew

- Intuition: measurement of asymmetry of the distribution about its mean.
- Negative skew commonly indicates that the tail is on the left side of the distribution, and positive skew indicates that the tail is on the right.



$$\tilde{\mu}_3 = \mathrm{E}\left[\left(\frac{X-\mu}{\sigma}\right)^3\right] = \frac{\mu_3}{\sigma^3} = \frac{\mathrm{E}\left[(X-\mu)^3\right]}{(\mathrm{E}[(X-\mu)^2])^{3/2}} = \frac{\kappa_3}{\kappa_2^{3/2}}$$

Source