CSE-555
Deep Learning and Applications
Synthetic Sensor Data Generation

Final Project

HAKKI YAVUZ ERZURUMLU

Introduction

In this report, I will try to implement **"SenseGen: A Deep Learning Architecture for Synthetic Sensor Data Generation"** paper. Our aim is generating synthetic data similar to **HAR dataset**.

You can access the whole source code and paper which used in this report from:

- https://github.com/freeloki/CSE555-VirtualSensorGenerator
- https://arxiv.org/abs/1701.08886

Tools

Here is the list of tools which I have used in this project.

- Human Activity Recoginition(HAR) dataset for training,validation and testing.
- keras deep learning framework with tensorflow backend.
- tensorflow framework for training and testing.

Data Preparation

Data is the most important part of any deep learning application. For this project, we used HAR Inertial Accelerometer and Gyroscope signal values.

Data	Training	Test
Inertial Total Acc X	(7352,128)	(2947,128)
Inertial Body Gyro X	(7352,128)	(2947,128)

Model

SenseGen consists of two deep learning models:

• Generator (G): The generator G is capable of generating new synthetic time series data from random noise input.

• Discriminator (D): The goal of the discriminator D is to assess the quality of the examples generated by the generator G.

Generative Model

Model consists 3 stack of 256 LSTM unit, top of that 128 fully connected layer and top it’s another 72 fully connected layer.

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
lstm (LSTM)                  (None, 1, 256)            264192
_________________________________________________________________
lstm_1 (LSTM)                (None, 1, 256)            525312
_________________________________________________________________
lstm_2 (LSTM)                (None, 1, 256)            525312
_________________________________________________________________
DENSE1 (Dense)               (None, 1, 128)            32896
_________________________________________________________________
DENSE2 (Dense)               (None, 1, 72)             9288
_________________________________________________________________
dense (Dense)                (None, 1, 24)             1752
=================================================================
Total params: 1,358,752
Trainable params: 1,358,752
Non-trainable params: 0
_________________________________________________________________
```

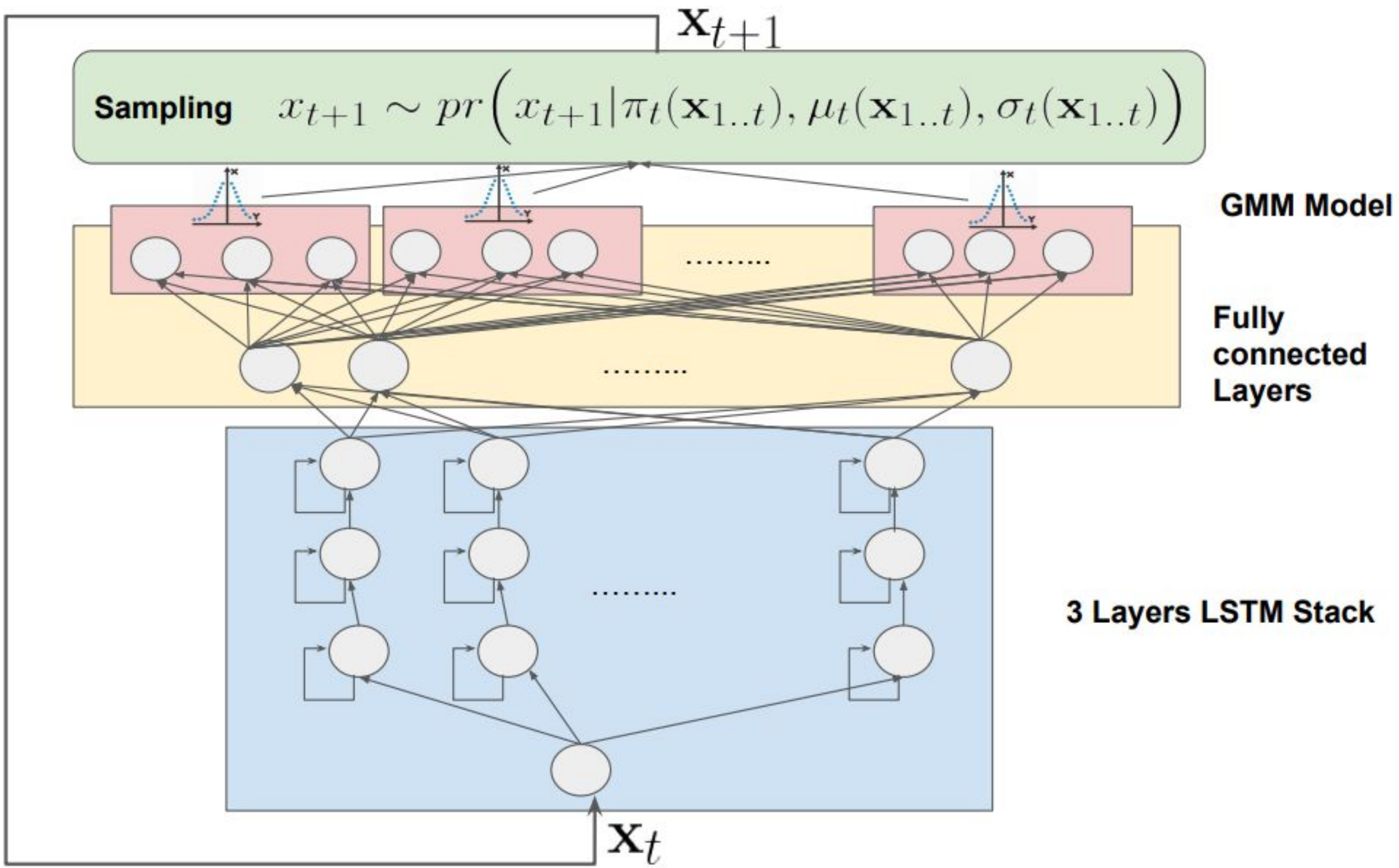
Generative Loss and GMM Parameters

$$\mathcal{L}^{\mathcal{G}}(\theta_{\mathcal{G}}) = -\sum_{t=1}^{T} \log\left(pr(x_{t+1}|\pi_t(\mathbf{x}_{1..t}), \mu_t(\mathbf{x}_{1..t}), \sigma_t(\mathbf{x}_{1..t}))\right)$$

$$\pi_t(\mathbf{x}_{1..t}) = softmax(l_t^{(5)}[1...24])$$

$$\mu_t(\mathbf{x}_{1..t}) = l_t^{(5)}[25...48]$$

$$\sigma_t(\mathbf{x}_{1..t}) = e^{(l_t^{(5)}[49...72])}$$



$\mathbf{x}_{t+1}$
Sampling $x_{t+1} \sim pr\Big(x_{t+1} | \pi_t(\mathbf{x}_{1..t}), \mu_t(\mathbf{x}_{1..t}), \sigma_t(\mathbf{x}_{1..t})\Big)$
GMM Model
Fully connected Layers
3 Layers LSTM Stack
$\mathbf{x}_t$

Discriminative Model

In order to quantify the similarity between the generated time-series and the real sensor timeseries collected from users. We build another model D whose goal is to distinguish between samples generated by G. The discriminative model D is trained to distinguish between the samples coming from the dataset for real sensor traces Xtrue and others samples from the dataset Xgen which is generated by the model G.

$$\mathcal{L}^{\mathcal{D}}(\theta_{\mathcal{D}}) = -\left(\sum_{i=1}^{m} \log\left(\mathcal{D}(\mathcal{X}_{true}^{(i)})\right) + \log\left(1 - \mathcal{D}(\mathcal{X}_{gen}^{(i)})\right)\right)$$

Discriminative RNN Model

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
lstm (LSTM)                  (None, 1, 128)            66560
_________________________________________________________________
DENSE1 (Dense)               (None, 1, 32)             4128
_________________________________________________________________
dense (Dense)                (None, 1, 1)              33
=================================================================
Total params: 70,721
Trainable params: 70,721
Non-trainable params: 0
```

Generative Model Training Parameters

Learning rate : 0.0003

Optimizer: Adam

Loss: Negative Log Likelihood

Batch Size: 4

Epoch Size: 1500

Parameters to be trained: 1,358,752

Discriminative Model Training Parameters

Learning rate : 0.0003

Optimizer: Adam

Loss: MSE

Batch Size: 128

Epoch Size: 1500

Parameters to be trained: 70,721

Training

Algorithm 1 Training algorithm

```
for t = 1, 2, ..., T do
    Sample X_true minibatch from true data
    Sample X_gen minibatch from the generative model G
    Train the discriminative model D on the training set
    (X_true, X_gen) for 200 epochs
    Sample another X_true minibatch from true data
    Sample another X_gen minibatch from the generative
    model G
    Train the generative model G on the training set (X_true)
    for 100 epochs
end for
```

Problems

Paper Epoch: 25000

NLL: -6

MyWork Epoch: ~1600

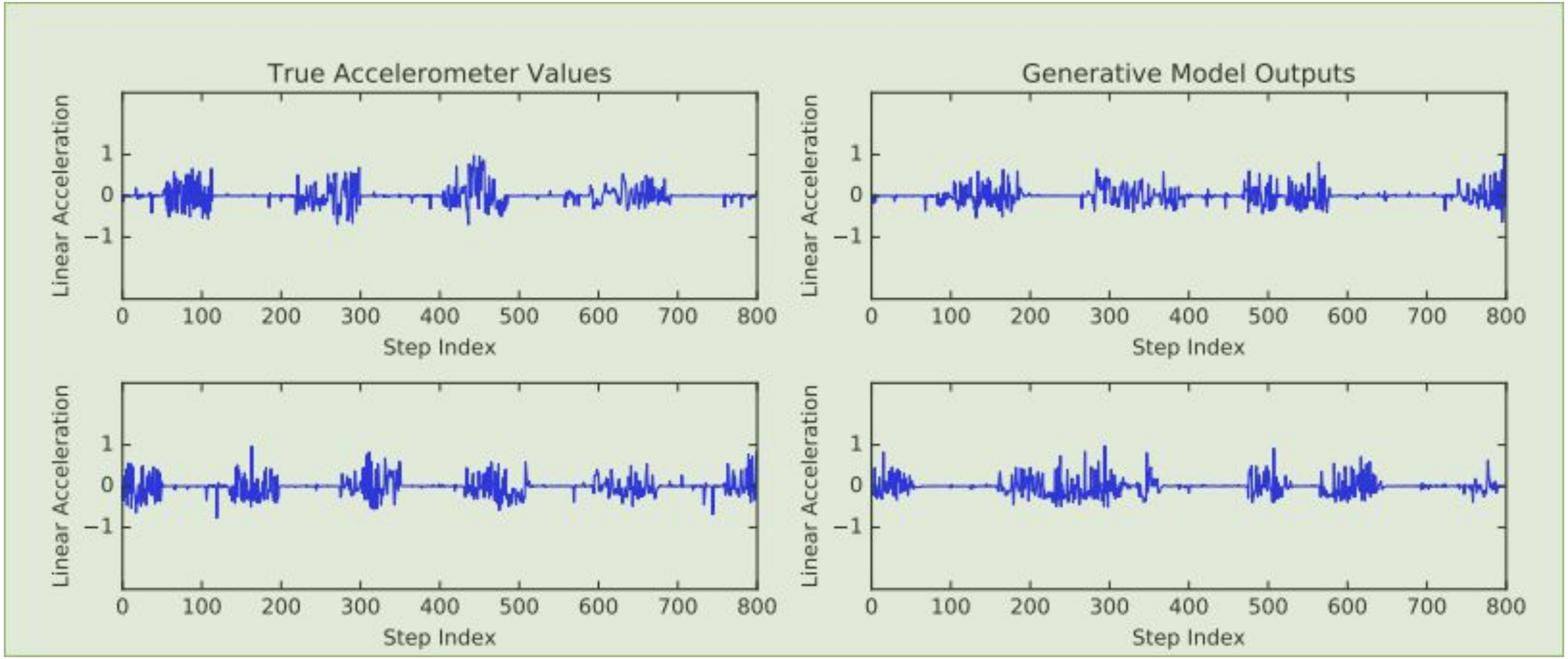
NLL: ~ -2.5

After that NaN

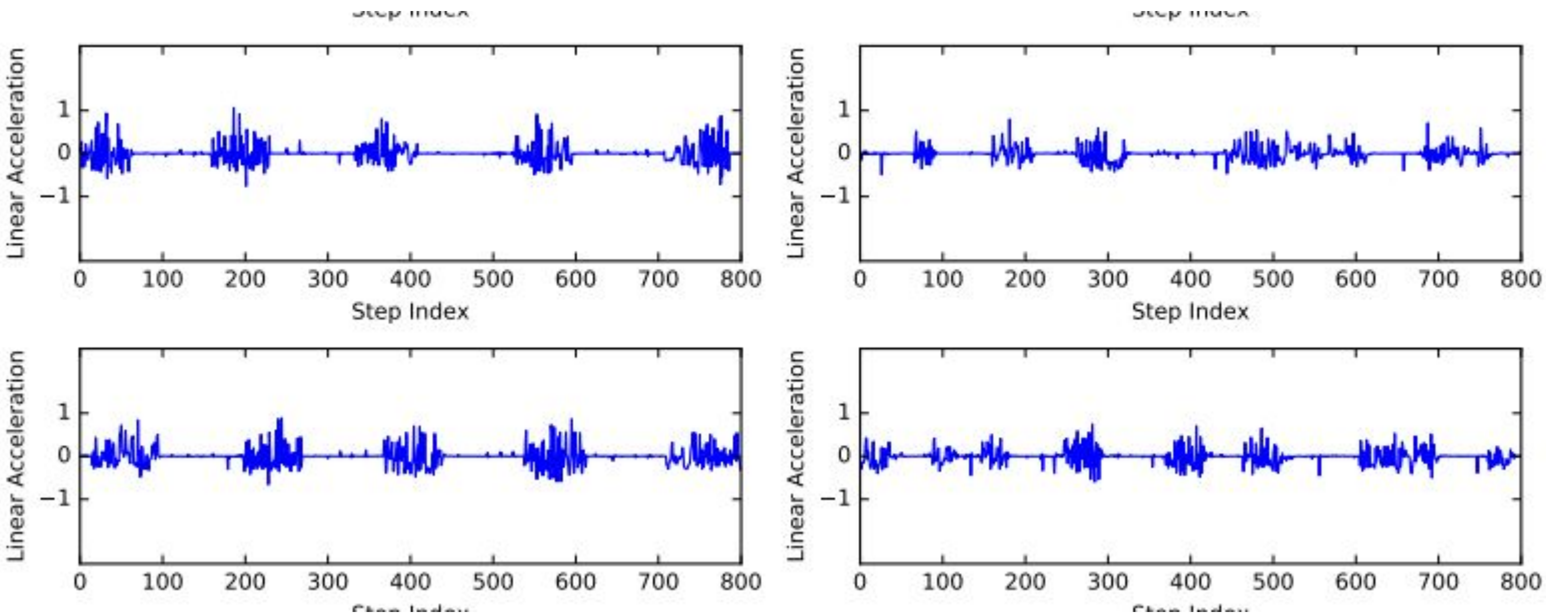
Sample Results

Epoch	Data	Generative Loss	Discriminative Loss	LSTM Unit Size
500	Accelerometer	-1.9754791	0.0021356253	256
1000	Accelerometer	-2.848514	0.0017647267	256
1500	Accelerometer	-3.2386086	0.0113765512	128
2000	Accelerometer	NaN	NaN	256
1500	Gyroscope	-3.6386086	0.001872581	256

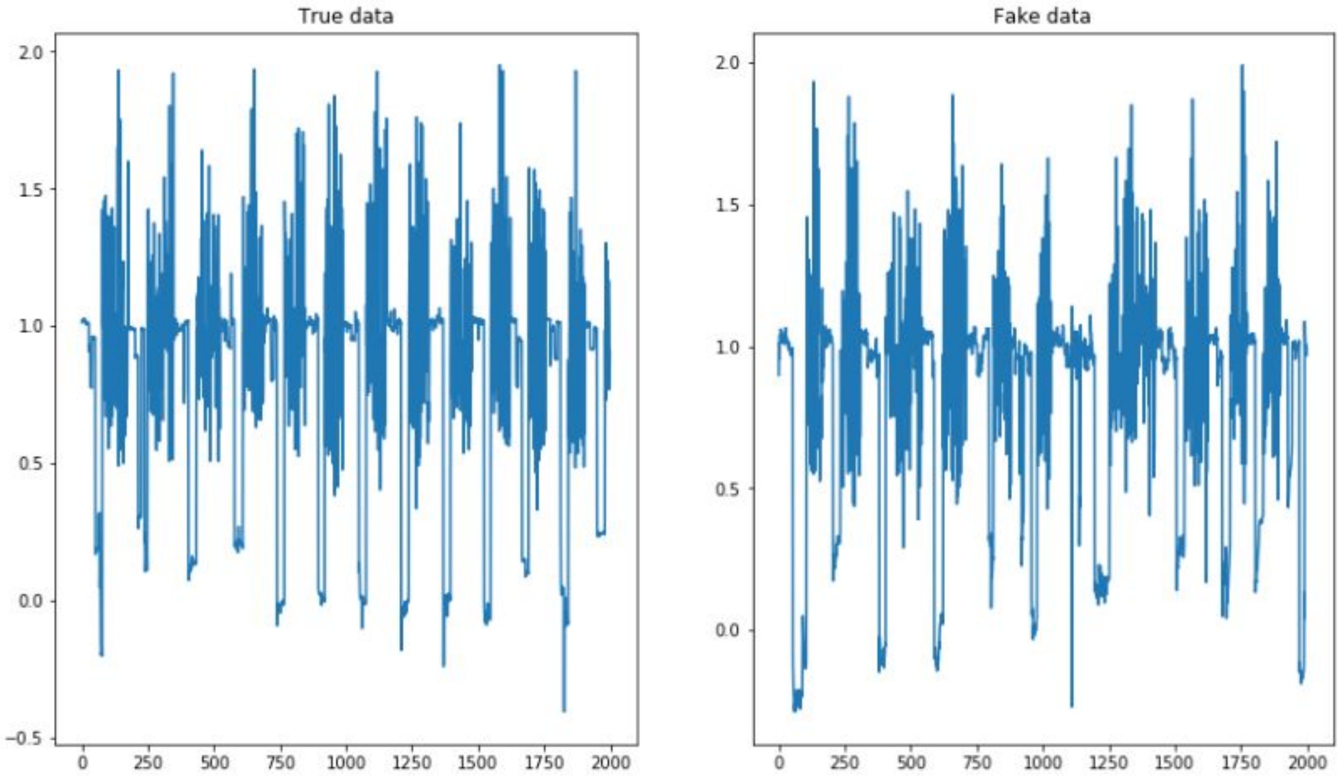
Accelerometer Results (Paper)



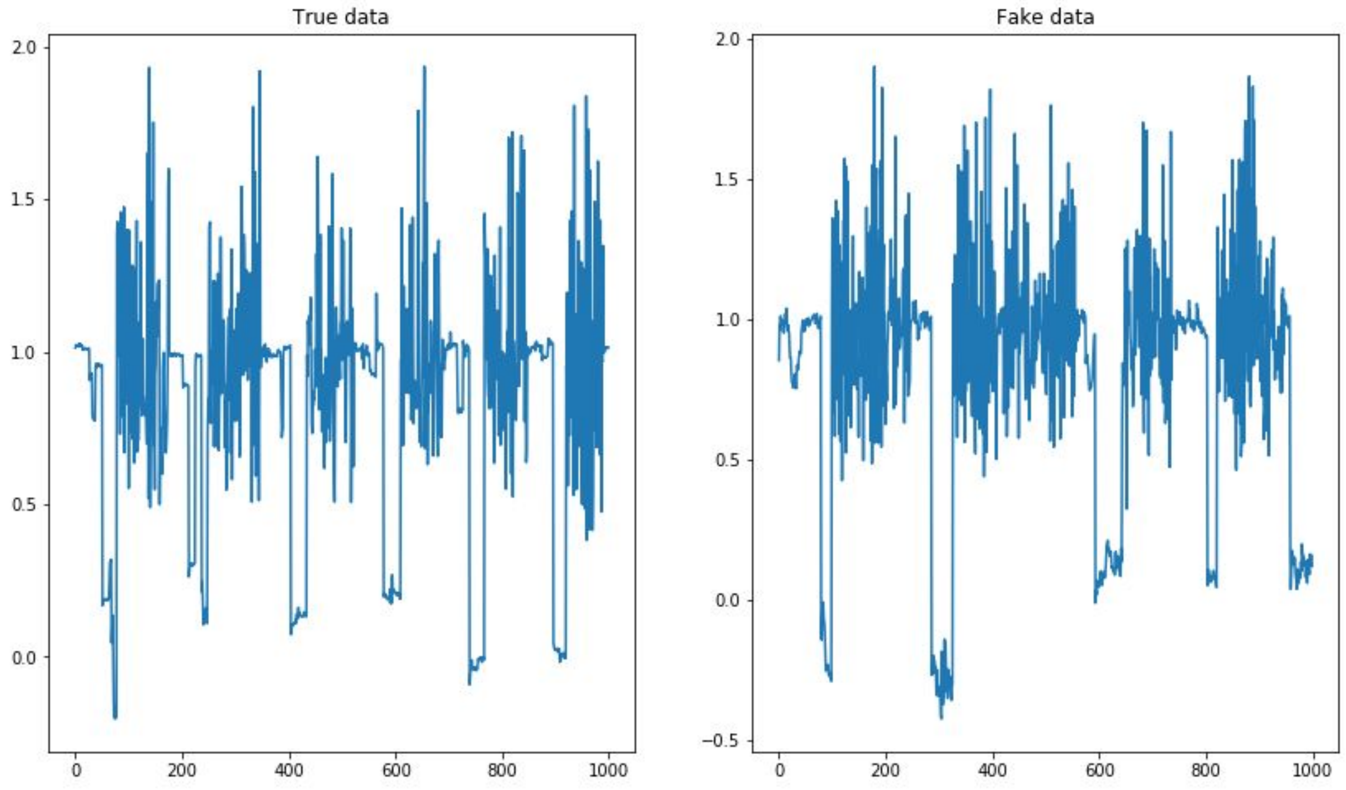
Accelerometer Results (Paper)



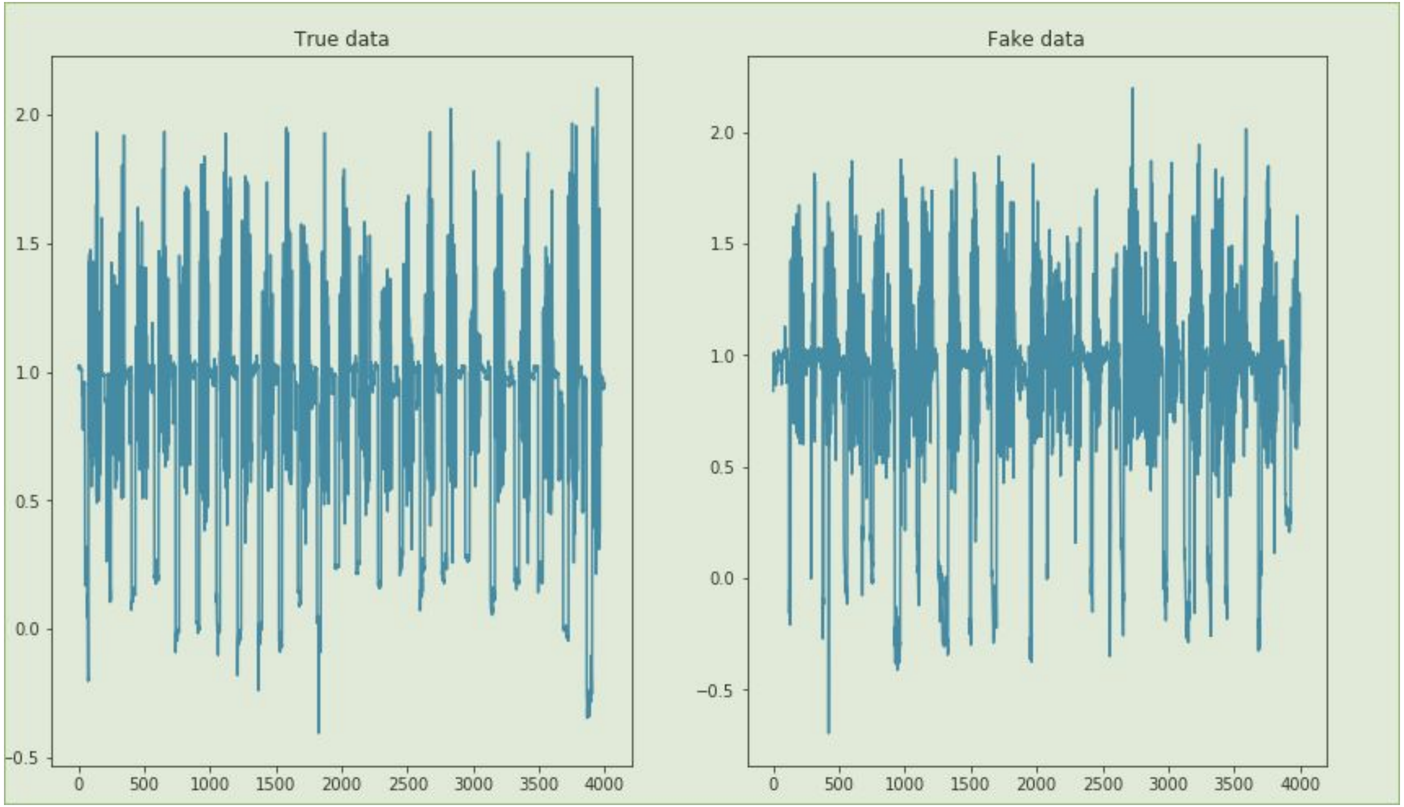
Accelerometer Results (MyWork)



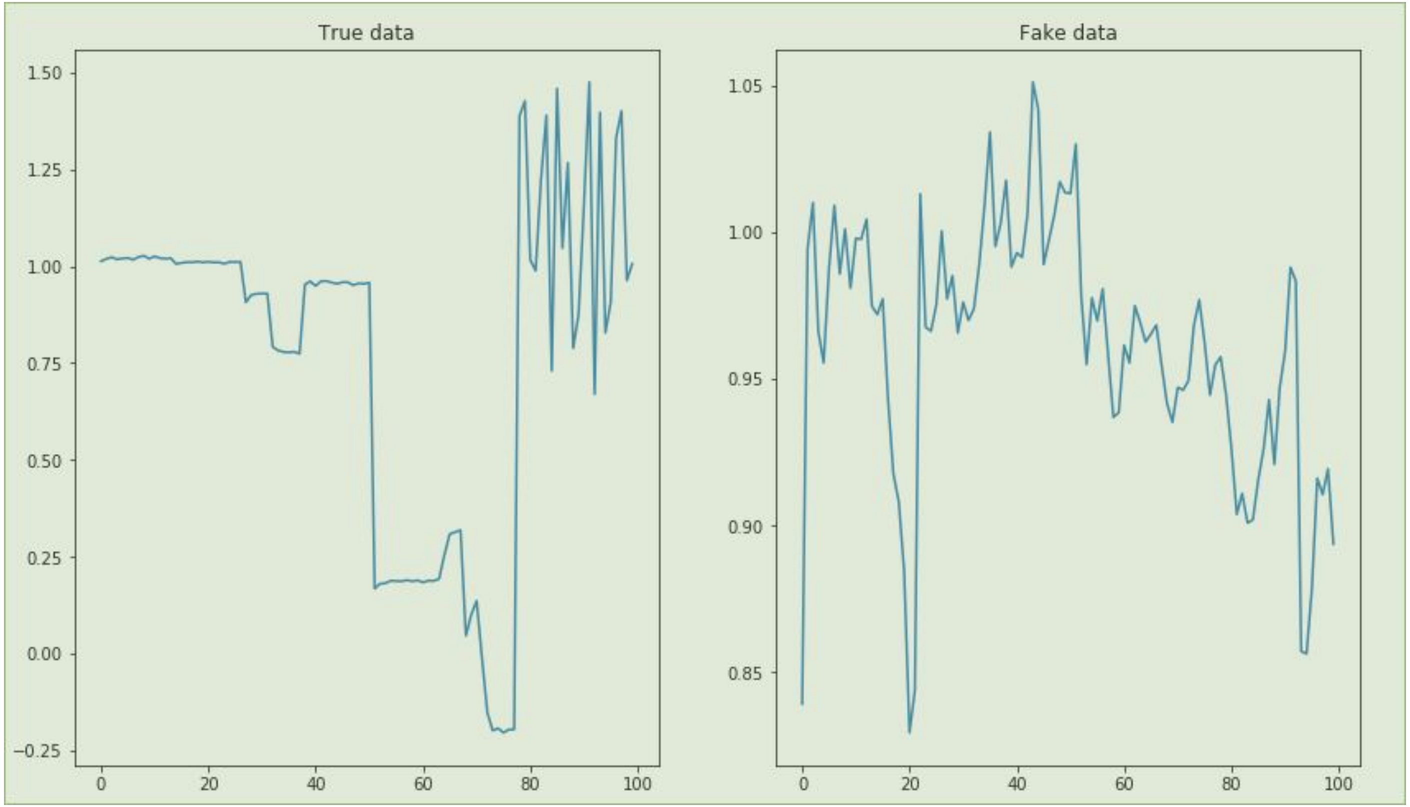
Accelerometer Results (MyWork)



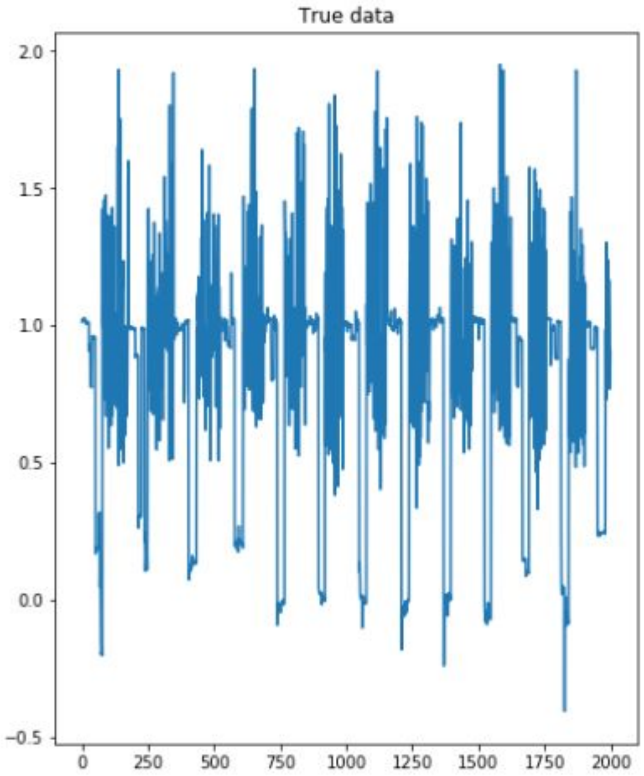
Accelerometer Results (MyWork)

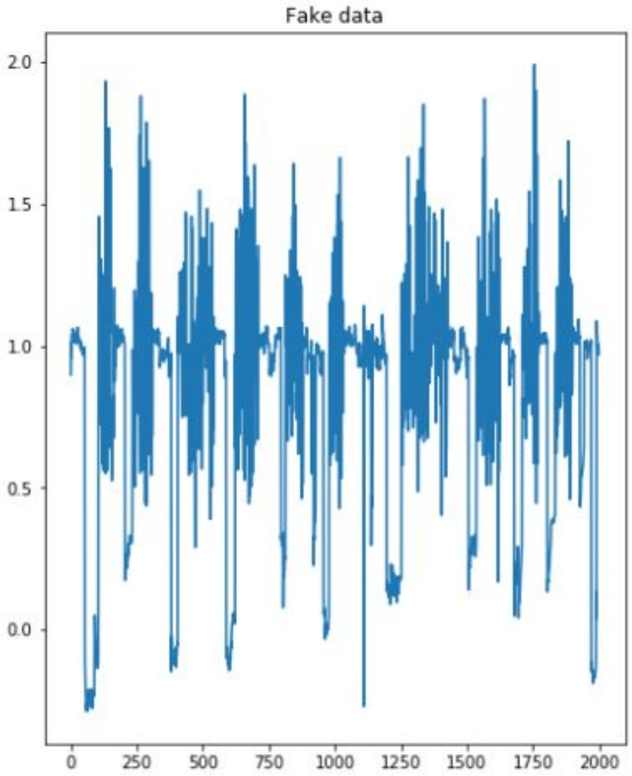


Accelerometer Results (MyWork-Fail Case)



Accelerometer Results (MyWork)





Gyroscope Results (MyWork)