Data Drift Management

Under the guidance of

Supervisor- Dr. Sumit Kalra

Examiners-Dr. Suman Kundu
Dr.Arpit Khandelwal

Indian Institute of Technology,Jodhpur

By
Shobhit Sharma
M.Tech. Ai
Dept. CSE,IIT jodhpur

Overview

- According to the study,there will be more than 55 billion IoT devices by 2025.
- Artificial intelligence (AI) and the Internet of Things (IoT) are a perfect example of two technologies that are interlinked.
- Data management is a huge task.
- Data from sensor devices changes time to time.
- Static machine learning model can be easily fooled in such situations, resulting in poor performances.

Most significant cause of data drift….

- Data drift = Change in data distribution
- concept drift = Change in relationship between input and output

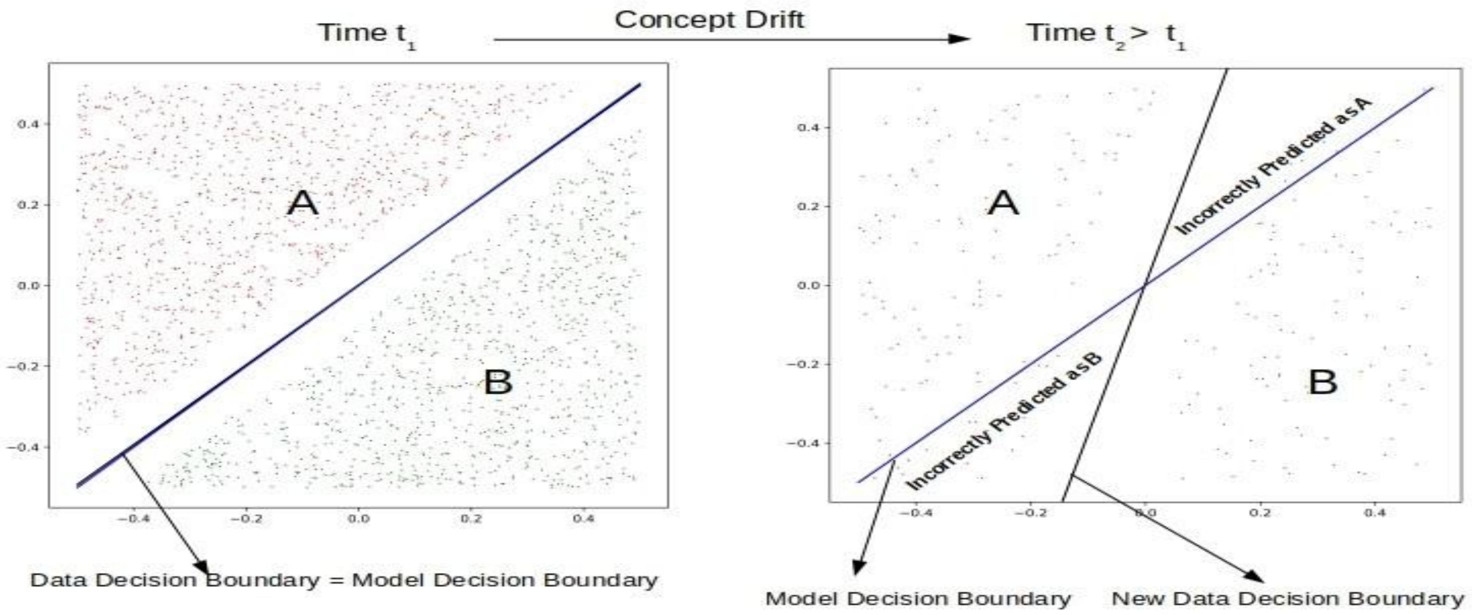


image source

- Deep learning models are effective in terms of their generalization power.
- Deep Neural Networks (DNNs) requires the entire training data to be made available prior to the learning task.
- This is not scalable for many real-world scenarios where new data arrives sequentially in a stream form...

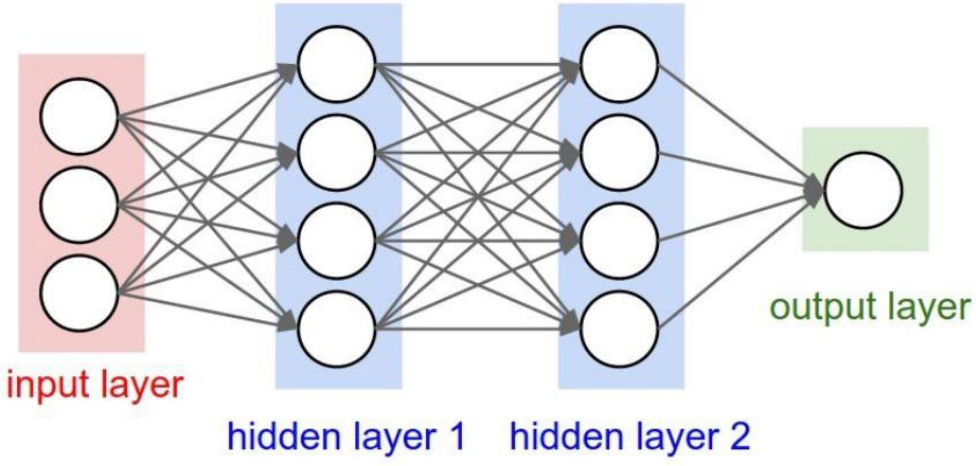


image source

Challenge

- Depth of deep learning model can’t be changed once fixed.
- Real world data arrives in a stream.

To overcome these issues:

1. Build a Deep learning model that can operate in dynamic environments.
2. Build a Model for drift data detection.

1.Build a dynamic deep learning model...

- Add output layer to each hidden layer separately.
- Associate each layer with its reward factor.
- Get prediction from each layer and loss of each layer.
- Reward is updated based on the loss obtained from each layer.
- Update model parameter.
- Final output is weighted sum of all the classifiers.

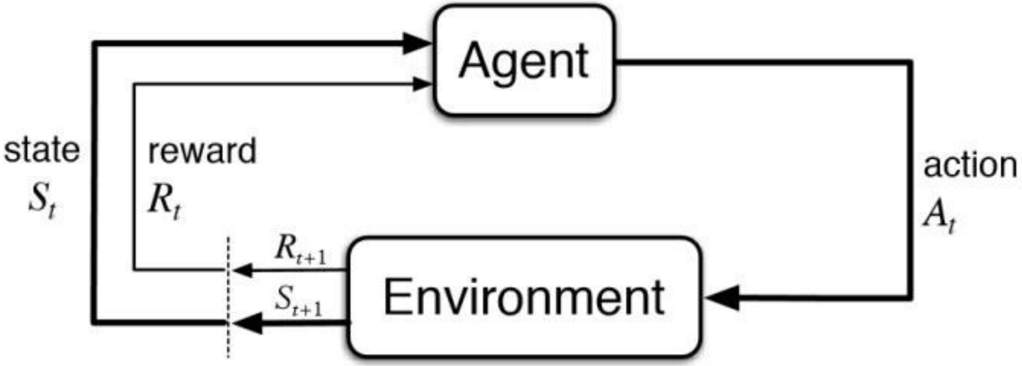


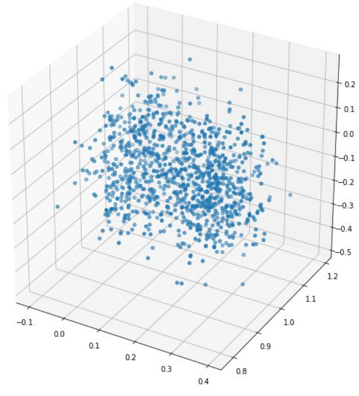
Image Source: Artificial Intelligence: A Modern Approach

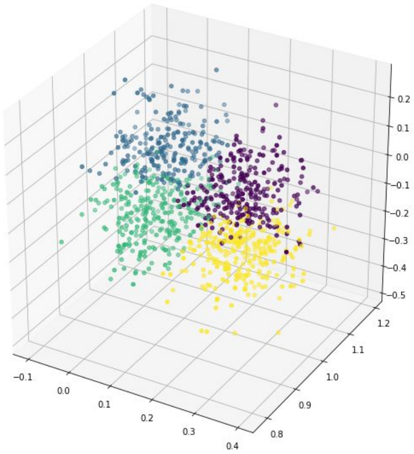
2.Model for drift detection...

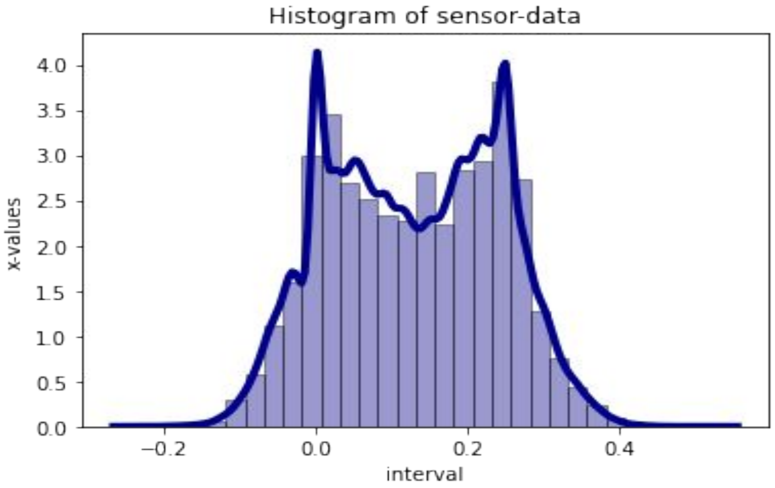
- Training...Testing...
- If instance is misclassified then 1 else 0.
- Accuracy matrices A=[0,0,1,1,1,0,0,0,1,1] -> B=[0,0,1,1,1],C=[0,0,0,1,1]
- Hoeffding error bound for each matrix is calculated.
- If mean(A)+(Hoeffding error A) < mean(B)+(Hoeffding error B)

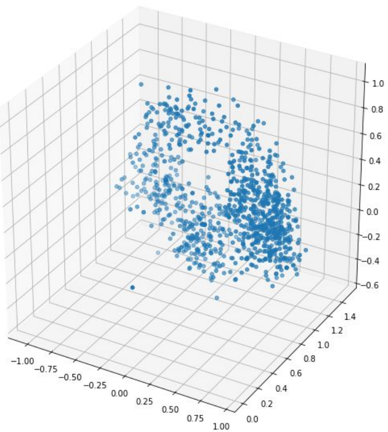
➔ Hoeffding (warning level)< mean(C) - mean(B)< Hoeffding (drift level)

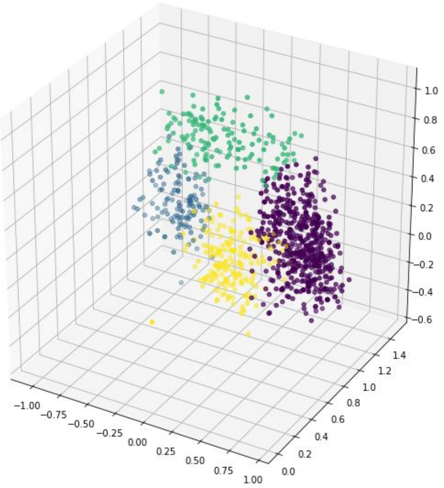
➔ mean(C) - mean(B) > Hoeffding (drift level)

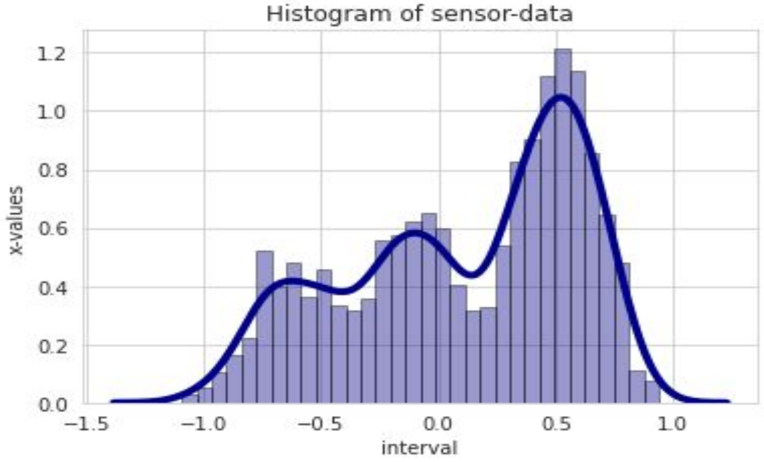





Histogram of sensor-data
x-values
interval
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
−0.2
0.0
0.2
0.4

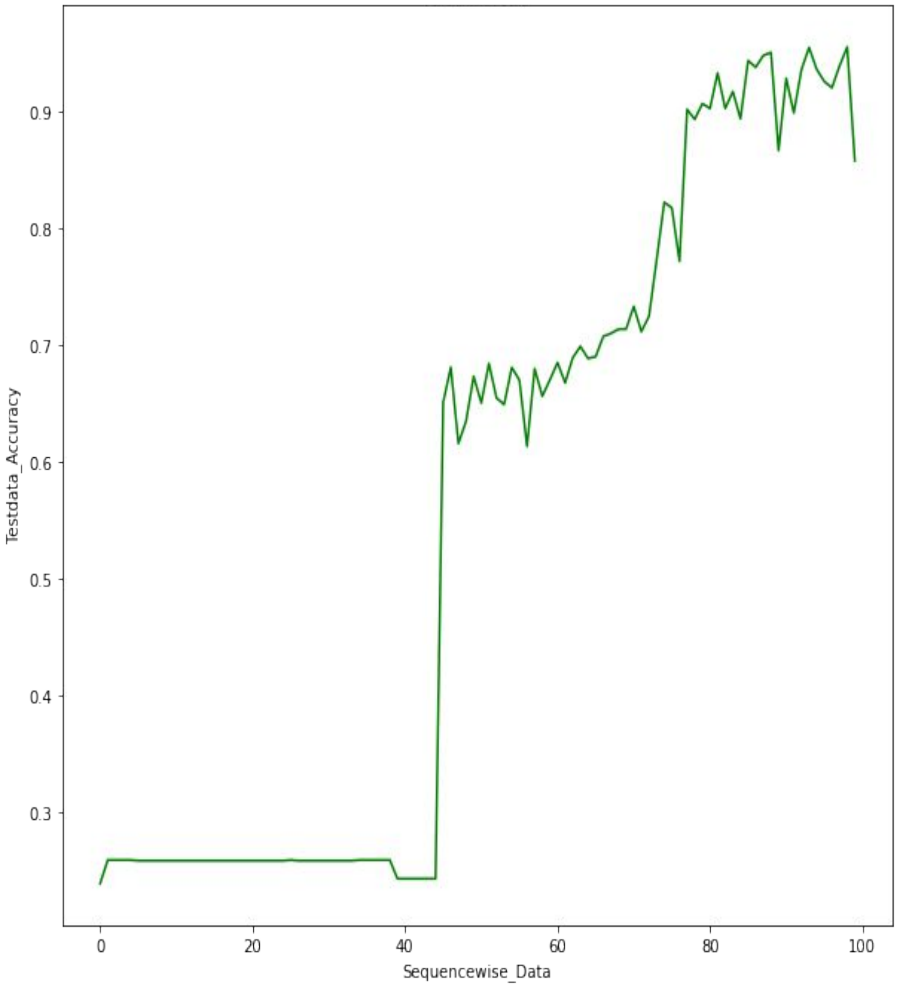




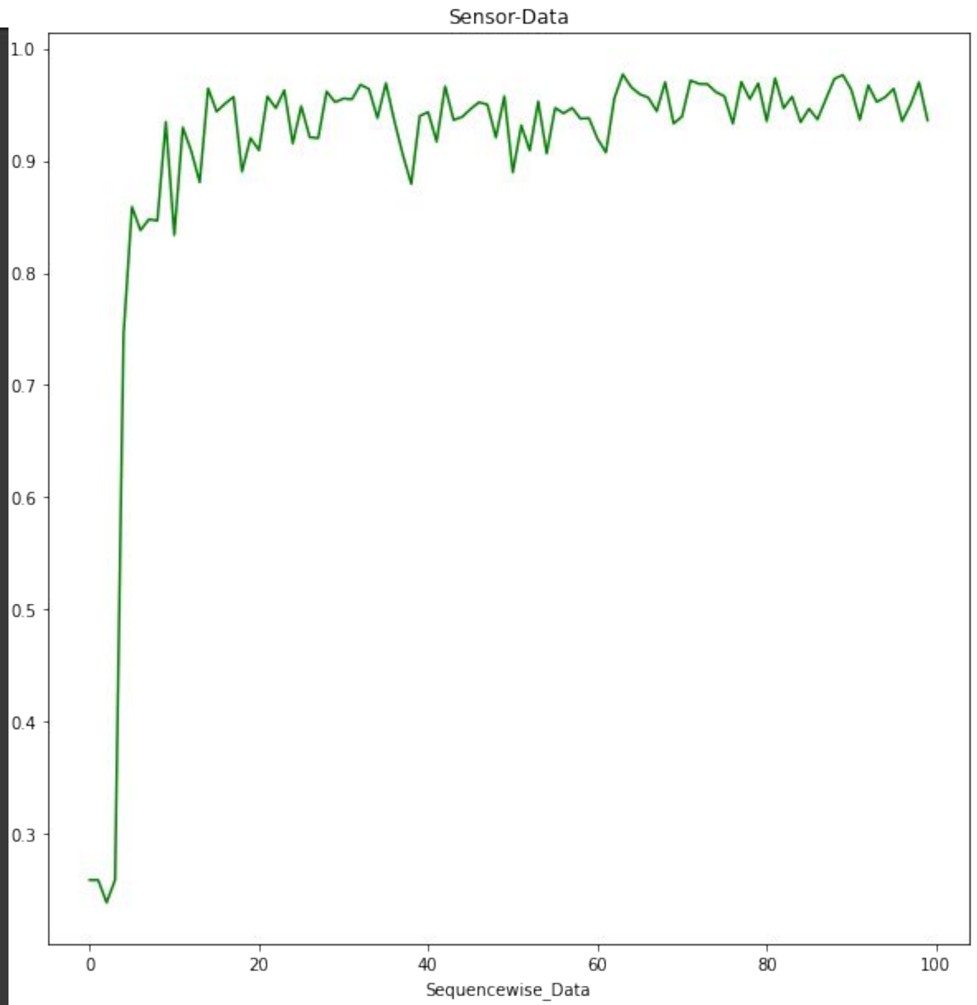

Histogram of sensor-data
x-values
interval
0.0
0.2
0.4
0.6
0.8
1.0
1.2
−1.5
−1.0
−0.5
0.0
0.5
1.0

Test Accuracy of sensor data with normal conditions

```
training loss tensor(1.3790)
Training examples 44000 are processed
Online Accuracy: 0.2591551507537688


Drift state: STABLE

training loss tensor(1.3759)
Training examples 45000 are processed
Online Accuracy: 0.2591551507537688


Drift state: STABLE

training loss tensor(1.3728)
Training examples 46000 are processed
Online Accuracy: 0.2591551507537688


Drift state: STABLE

training loss tensor(1.3684)
Training examples 47000 are processed
Online Accuracy: 0.5083542713567839


Drift state: STABLE

training loss tensor(1.3623)
Training examples 48000 are processed
Online Accuracy: 0.4685694095477387


Drift state: STABLE
```

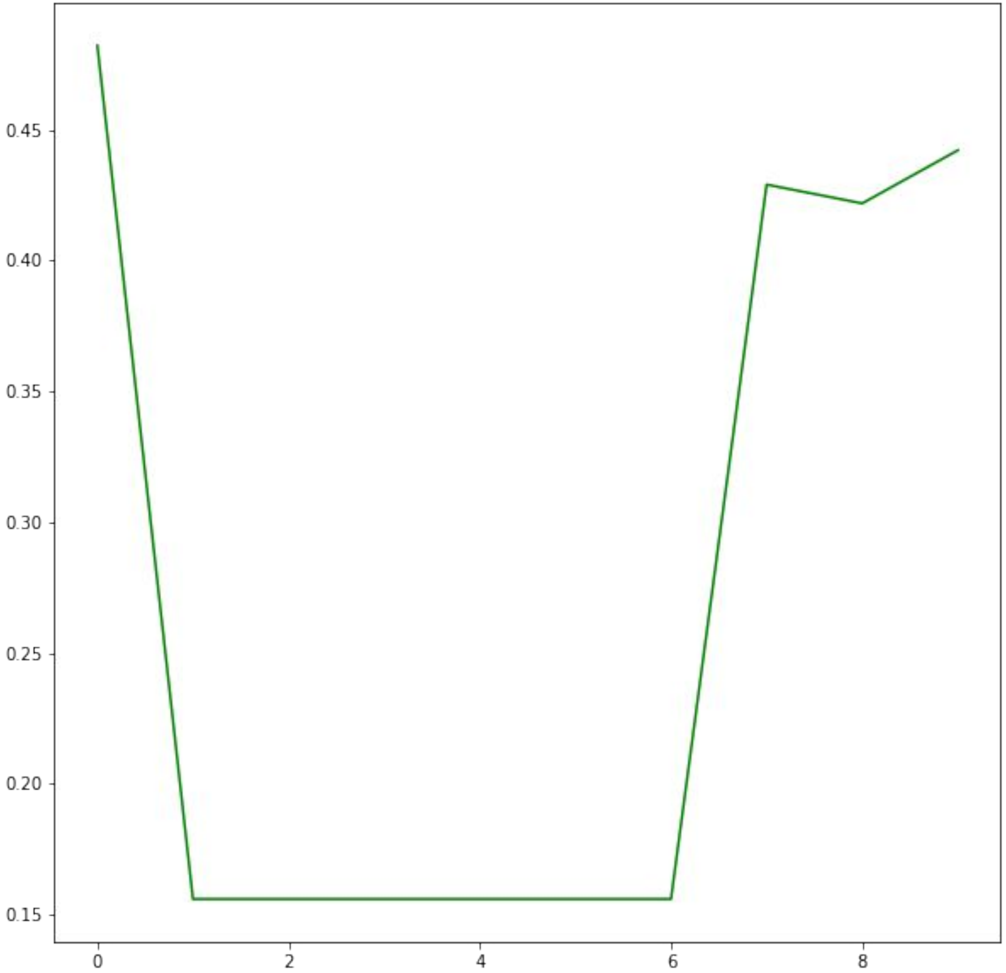


```
Training examples 1000 are processed

Online Accuracy: 0.2591551507537688

Drift state: STABLE

Alpha:[0.25539473 0.29447427 0.21638454 0.23374646]
Training Loss: 1.410154
Training examples 2000 are processed

Online Accuracy: 0.23907820351758793

Drift state: STABLE

Alpha:[0.29799931 0.28796478 0.1839748  0.23006112]
Training Loss: 1.3902878
Training examples 3000 are processed

Online Accuracy: 0.259375

Drift state: STABLE

Alpha:[0.95590642 0.01818054 0.01121726 0.01469577]
Training Loss: 1.0712863
Training examples 4000 are processed

Online Accuracy: 0.7665829145728643

Drift state: STABLE
```

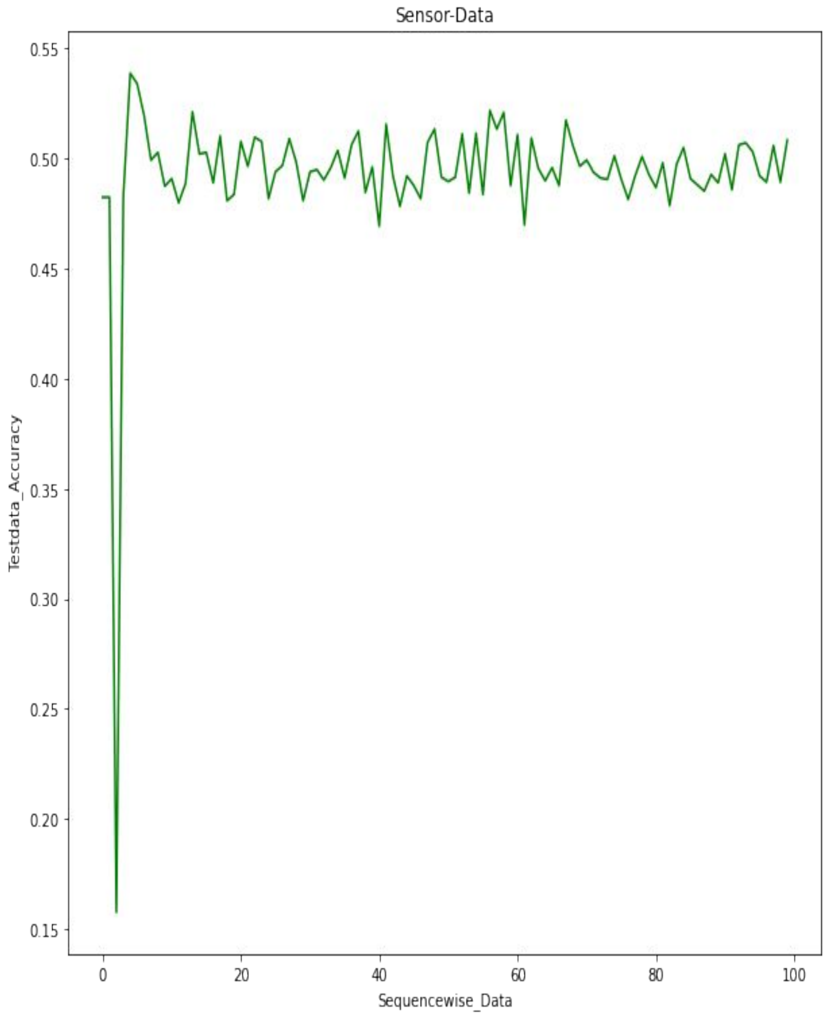


Test Accuracy of sensor data with drift conditions

```
training loss tensor(1.3875)
Training examples 5000 are processed
Online Accuracy: 0.4821875


Drift state: STABLE

training loss tensor(1.3873)
Training examples 6000 are processed
Online Accuracy: 0.1559375


cut 95
(1.0, 0.8392914653784219) 0.13015933201208676
Drift state: DRIFT

training loss tensor(1.3872)
Training examples 7000 are processed
Online Accuracy: 0.1559375


cut 95
(1.0, 0.8392914653784219) 0.13015933201208676
Drift state: DRIFT

training loss tensor(1.3869)
Training examples 8000 are processed
Online Accuracy: 0.1559375


cut 95
(1.0, 0.8392914653784219) 0.13015933201208676
Drift state: DRIFT
```



```
Drift state: STABLE

Alpha:[0.2549021  0.29422696 0.21642609 0.23444485]
Training Loss: 1.410153
Training examples 2000 are processed

Online Accuracy: 0.1578125

0.9880952380952381 0.8382541720154044 0.13841956201355718
instance 84
Drift state: DRIFT

Alpha:[0.30269261 0.28579617 0.18240191 0.22910931]
Training Loss: 1.3894299
Training examples 3000 are processed

Online Accuracy: 0.4821875

Drift state: STABLE

Alpha:[0.9708815  0.01190039 0.00750075 0.00971736]
Training Loss: 1.0288513
Training examples 4000 are processed

Online Accuracy: 0.451875

Drift state: STABLE

Alpha:[9.99998461e-01 6.28047468e-07 3.97472282e-07 5.13814334e-07]
Training Loss: 0.40532893
Training examples 5000 are processed

Online Accuracy: 0.5340625
```



```
Alpha:[1.00000000e+00 4.21525937e-42 2.61256409e-42 3.41463225e-42]
Training Loss: 0.20091999
Training examples 12000 are processed

Online Accuracy: 0.908071608040201

Drift state: STABLE

Alpha:[1.00000000e+00 2.52804457e-47 1.56684984e-47 2.04787932e-47]
Training Loss: 0.18989632
Training examples 13000 are processed

Online Accuracy: 0.9198570979899497

Drift state: STABLE

Alpha:[1.00000000e+00 1.26228247e-52 7.82346605e-53 1.02253030e-52]
Training Loss: 0.17166214
Training examples 14000 are processed

Online Accuracy: 0.968498743718593

Drift state: STABLE
```

1.Normal working data:

Traditional Deep Learning model:

Accuracy:0.508 (After 47000 instances trained)

Dynamic Deep Learning model :

Accuracy:0.77 (After 4000 instances trained) and 0.96 (After 14000 instances trained)

2.Drift Data:

Traditional Deep Learning model:

Accuracy:0.15 (Drift signal at 95th instance)

After that you can see drift is not handled in a manner and the same accuracy and drift is signalled in next testbatches)

Dynamic Deep Learning model :

Accuracy:0.15 (Drift signal at 84th instance)

you can see drift is stable after that iteration in a manner and 0.53 accuracy is achieved.

Thank you

(all image sources are attached)