University of Glasgow | School of Computing Science

ESSENCE

Technical Report 2019-08/01/A

V1.0 – August 2019

Application of Optimal Stopping Time on Large-scale Data Streams in Distributed Computing Environments

Name of Research Fellow: Mr Valentin Orru

Researcher's Institute/University: Phelma, Grenoble INP, France

Host School & University: School of Computing, University of Glasgow, Essence Research Group

Name of Supervisor: Dr Christos Anagnostopoulos

A. Abstract

This report presents an exploration of several methods based on edge-computing to reduce the communication costs in an information system collecting data to build predictive models.

These methods were tested using an algorithm simulating the behavior of an edge node of the system for various data sets and parameters.

The results of these tests allowed us to build a set of rules that indicate the best method to use depending on the situation.

B. Introduction

As technological evolution skyrockets, we live in an evermore connected world where the IoT as become a major field of research. However, these technological evolutions comes with an ecological price. In this regard, this study focus on the data transfer in the IoT to create new ways to collect and process data while reducing the ecological cost and keeping the accuracy loss at a minimum.

For our study, we consider a system composed of several nodes whose purpose is to collect data to build predictive models (where the data is a multidimensional vector that might be anything). The simplest way to do that is to have several edge nodes with sensors collect data to send all of these to a sync node which will compute them to build models. However, in this process, the communication cost is way higher than the computation cost. It is therefore very expensive to send all the data to the sync node.

A solution is to send only the models to the sync nodes, while computing the data directly on the edge nodes. This is what is called edge computing. To do so, we keep track of the time were each t time instance corresponds to the censors acquiring one new data. We build a model every W time instance where W is the size of our buffer i.e. the number of data we can store on the edge node.

With this solution, we send a model every W time instances, which reduce drastically the cost of communication as we send a few parameters to replace W instances of data (with the data possibly being a multidimensional vector). But it comes with a slight accuracy loss as we will only be able to get a new model every W time instance. If W is very high, then the last model that was sent might not match the data at all. Likewise, if W is too small, then it might be useless to send the new model as it will be very close to the last one. And we can't forget that the value of W will often be limited by the storage space on the sync node.

To offer a better compromise between cost and accuracy without being limited by the W value, we give ourselves the possibility to not send the model after computing it if it is too similar to the last one sent. The question here is to define exactly what being « similar » means for two models and how to evaluate it best. The whole purpose of this study is to research several sending policies to reach the best compromise between accuracy and cost.

C. METHODOLOGY

To test and compare several methods, the first step was to build an algorithm allowing us to simulate the behaviour of one edge node. That algorithm was made in Java using the weka[4] library to deal with the data sets. We used different parameters and values in our simulation:

-W, the buffer size on the edge node i.e. the number of time instances between each computation of a new model.
-m, the cost payed to compute the model.
-M, the cost payed to send the model.
-$e_0[t]$, the error calculated at the time instance t using the last model sent and the W last data. (We calculate the RMSE between the current last W values and the same values obtained with the model). It represents the error of the sync node over the last W data.
-$e_i[t]$, the error caculated at the time instance t using the last model computed (i.e. the model number i =t/W) and the W last data (same method with the RMSE).
-Z the error discrepancy which is equal to $|e_0[t] - e_i[t]|$
-BASE-(COST/RMSE/Z)which are the average values of the cost/$e_0[t]$/Z in the case we always send the model after computation.
-(COST/RMSE/Z)-RATIO which are the averages values of the cost/$e_0[t]$/Z with the actual method divided by BASE-COST/RMSE/Z.

Then, with entry parameters such as W, m and M and one entry data set, we were able to simulate the behaviour of the node and get the error and the decision made by the node for each time instance.

We compiled our results to get different ratios, allowing use to judge the efficiency of the sending policies over the entire data set.

We came up with 6 different methods. For each of them, we ran the simulation with different entry parameters to obtain detailed results.

From these observations, we built a set of rules. Then, we repeated the process with different data sets to test and adjust these rules in order to come up with the more accurate conclusions.

D. EXPERIMENTATION

I) EXPERIMENTING WITH DIFFERENT METHODS

1.) Method 1 : Discrepancy limitation

In this method, we define a threshold θ which is the maximum discrepancy tolerated.
Each W time instance, we calculate the new Z and if it exceeds θ, we send the model.

First, we can observe that the cost ratio is affected by the change in the M/m ratio. When m gets closer to M, not sending is less effective and therefore, the cost ratio is increasing.

Then, when we observe the evolution of the θ parameter we can isolate two trends. When θ increase, at first, the cost ratio is quickly decreasing while the Z and RMSE ratio only slightly increase. And then, the cost ratio starts to decrease more slowly while the RMSE and Z ratio are increasing faster. This tells us that there is an optimal θ value to get the best compromise where the cost ratio has significantly decreased, but the RMSE and Z ratio have not increased much yet.
We can also observe that the only difference between the RMSE and the Z ratio is that the second is increasing much faster when θ goes up, which means that even if the models are very different, they can very well create a similar error.

If we compare different W values we can also see that this « optimal point » does not occur for the same θ. For example, we can locate it between 1.0 and 2.0 when W=10 and between 0.5 and 1 when W=50. But for W=100, it gets harder as the RMSE and Z ratio are increasing much faster when θ goes up as the time window is bigger and so is the error. Moreover, the cost ratio is decreasing at a slower pace since a bigger time window already saves a lot by not sending or computing for W time instances, so the base cost is much smaller.

W	M/m	Theta-Z	COST-RATIO	RMSE-RATIO	Z-RATIO
10	20	0,1	0,662	0,999	0,995
10	20	0,5	0,338	1,004	1,017
10	20	1	0,187	1,001	1,007
10	20	2	0,116	1,081	1,329
10	20	5	0,069	1,347	2,411
10	20	10	0,056	1,585	3,379
10	10	0,1	0,678	0,999	0,995
10	10	0,5	0,368	1,004	1,017
10	10	1	0,224	1,001	1,007
10	10	2	0,156	1,081	1,329
10	10	5	0,111	1,347	2,411
10	10	10	0,098	1,585	3,379
10	5	0,1	0,705	0,999	0,995
10	5	0,5	0,421	1,004	1,017
10	5	1	0,289	1,001	1,007
10	5	2	0,227	1,081	1,329
10	5	5	0,185	1,347	2,411
10	5	10	0,174	1,585	3,379
10	2	0,1	0,764	0,999	0,995
10	2	0,5	0,536	1,004	1,017
10	2	1	0,431	1,001	1,007
10	2	2	0,381	1,081	1,329
10	2	5	0,348	1,347	2,411
10	2	10	0,339	1,585	3,379
100	20	0,1	0,709	1,006	1,025
100	20	0,5	0,471	1,02	1,083
100	20	1	0,339	1,097	1,407
100	20	2	0,233	1,152	1,639
100	20	5	0,153	1,468	2,972
100	20	10	0,074	1,459	2,931
100	10	0,1	0,722	1,006	1,025
100	10	0,5	0,495	1,02	1,083
100	10	1	0,369	1,097	1,407
100	10	2	0,268	1,152	1,639
100	10	5	0,192	1,468	2,972
100	10	10	0,116	1,459	2,931
100	5	0,1	0,745	1,006	1,025
100	5	0,5	0,537	1,02	1,083
100	5	1	0,421	1,097	1,407
100	5	2	0,329	1,152	1,639
100	5	5	0,259	1,468	2,972
100	5	10	0,19	1,459	2,931
100	2	0,1	0,796	1,006	1,025
100	2	0,5	0,63	1,02	1,083
100	2	1	0,537	1,097	1,407
100	2	2	0,463	1,152	1,639
100	2	5	0,407	1,468	2,972
100	2	10	0,352	1,459	2,931

Figure 1: Method 1 – Reduced ratio table

From that, we can conclude that this « optimal point » is easier to reach for lower W, but is it better ?

Even in the worst case scenario when M and m are close and it gets expensive to compute often so the cost is getting higher for a small W, the RMSE and Z ratio are really increasing too fast at high W for it to make a difference. We can look at the highlighted rows on Fig.1 : if we want to keep a decent error such as less than 10 %, we will still get a smaller cost with W=10 than with W=100, even with a small M/m ratio.
Anyway, it seems the best compromise is always at a small W if we want a reasonable error. But even so, it all depends on the conditions(M, m) and the limitations (Wmax, needs in accuracy).

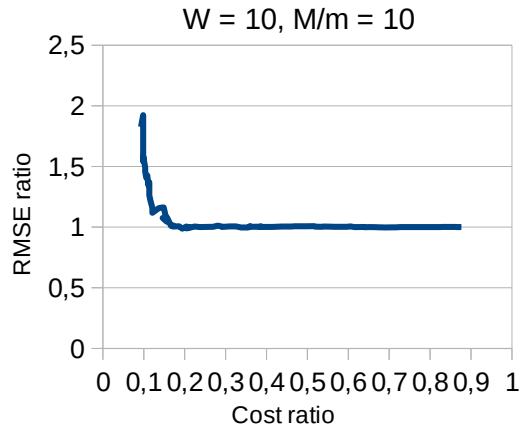

Figure 2: Method 1 - RMSE for cost

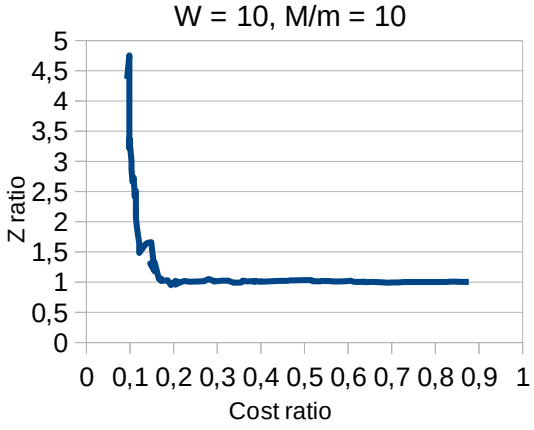

Figure 3: Method 1 - Discrepancy for cost

2.) Method 2 : Cumulative error limitation

Once again, we define a threshold θ which is different here, it is the maximum cumulative discrepancy tolerated before sending the model.
Each W time instance, we add the discrepancy Z over the last W data to a cumulative sum S.
Then, if S exceeds θ, we send the model and reset S to 0.

Here, the error ratio seems to increase even faster at a high W so it is even more obvious than the best compromise is at a low W, even if the M/m ratio is low (That is if we still want a reasonable error).

Then we can compare this method to the first one. To do so, let's fix an RMSE-ratio threshold (we can imagine it being a condition from the client) and get the better cost-ratio possible with this limitation.
The first thing we notice is that we can't really get a global trend out of the result since the best model seems to change for each parameters, and it is often very hard to determine as they are pretty close, and it gets even harder when W goes up.
So overall, the difference between the two models is not that obvious.

However, when W is small, the second model seems to give better result than the first one even though the values are pretty close (see highlighted rows again). To confirm it, we can look at the comparative plot between the two models for W=10 and M/m=10.

W	M/m	Theta-SUM	COST-RATIO	RMSE-RATIO	Z-RATIO
10	20	1	0,346	1,012	1,048
10	20	5	0,143	0,985	0,94
10	20	10	0,103	1,03	1,121
10	20	20	0,09	1,094	1,384
10	20	30	0,077	1,117	1,476
10	20	50	0,071	1,219	1,892
10	10	1	0,375	1,012	1,048
10	10	5	0,182	0,985	0,94
10	10	10	0,144	1,03	1,121
10	10	20	0,131	1,094	1,384
10	10	30	0,119	1,117	1,476
10	10	50	0,114	1,219	1,892
10	5	1	0,428	1,012	1,048
10	5	5	0,25	0,985	0,94
10	5	10	0,215	1,03	1,121
10	5	20	0,204	1,094	1,384
10	5	30	0,192	1,117	1,476
10	5	50	0,187	1,219	1,892
10	2	1	0,542	1,012	1,048
10	2	5	0,4	0,985	0,94
10	2	10	0,372	1,03	1,121
10	2	20	0,363	1,094	1,384
10	2	30	0,354	1,117	1,476
10	2	50	0,35	1,219	1,892
100	20	1	0,418	1,068	1,285
100	20	5	0,233	1,142	1,597
100	20	10	0,18	1,251	2,058
100	20	20	0,153	1,388	2,634
100	20	30	0,127	1,416	2,751
100	20	50	0,127	1,819	4,449
100	10	1	0,444	1,068	1,285
100	10	5	0,268	1,142	1,597
100	10	10	0,217	1,251	2,058
100	10	20	0,192	1,388	2,634
100	10	30	0,167	1,416	2,751
100	10	50	0,167	1,819	4,449
100	5	1	0,491	1,068	1,285
100	5	5	0,329	1,142	1,597
100	5	10	0,282	1,251	2,058
100	5	20	0,259	1,388	2,634
100	5	30	0,236	1,416	2,751
100	5	50	0,236	1,819	4,449
100	2	1	0,593	1,068	1,285
100	2	5	0,463	1,142	1,597
100	2	10	0,426	1,251	2,058
100	2	20	0,407	1,388	2,634
100	2	30	0,389	1,416	2,751
100	2	50	0,389	1,819	4,449

Figure 4: Method 2 - Reduced ratio table

The second model seems overall better, especially when we want to keep a reasonable error.

That result is really interesting as we established earlier that a low W gave better compromise, so that's where we ideally want to work to get the best output.

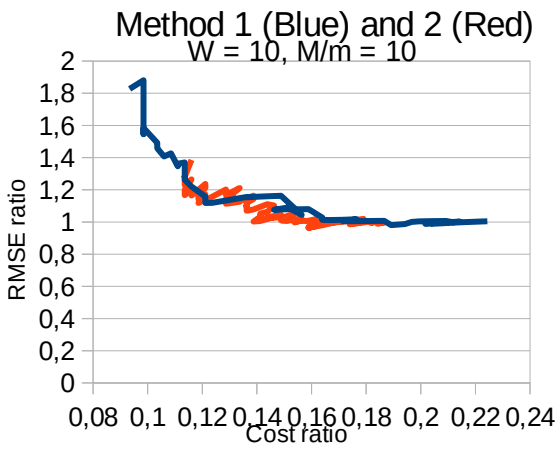

Figure 5: Method 1 and 2 comparison - RMSE for cost

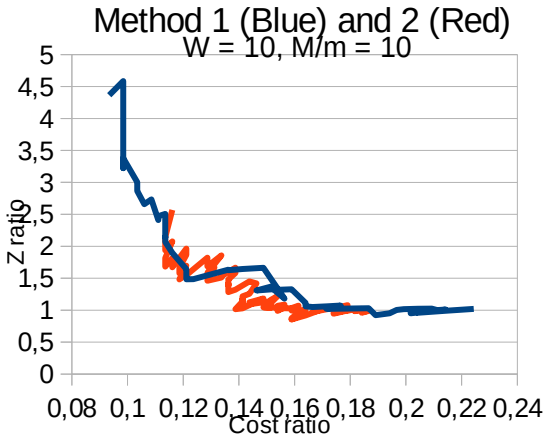

Figure 6: Method 1 and 2 comparison - Discrepancy for cost

Therefore, so far, to get the best result, it is better working with a small W and the second model.

3.1) Method 3 : Application of OST theory

In this method, we want to use the Optimal Stopping Time theory to maximize the outcome which is the cost-accuracy ratio. To do so, we use the sum S defined earlier in the method 2.

We also need to define and compute the probability density function of Z in order to predict its value. In our simulation, we are doing it by running through the data once before launching the actual simulation. In a real situation, the PDF will be built over the first data samples and update itself with the new ones. So we consider the system once it is stabilized and the PDF is already built.

Next, instead of simply sending when the sum exceeds a value we apply the OST theory to try maximizing the outcome. To do so, we define a gain V_t which is the overall benefit we got from not sending the model until the time t.

It can take two values :

- $V_t = Mt$ if we don't exceed the threshold at time t.
- $V_t = Mt - B$ else with B a penalty that we define.

Each W time instances, we update S :
- If S > θ , then we immediately send and set S to 0.

Theta-OST	B/M	COST-RATIO	RMSE-RATIO	Z-RATIO
10	0,01	0,144	1,03	1,121
10	1	0,144	1,03	1,121
10	5	0,159	0,967	0,865
10	10	0,295	1,005	1,021
10	20	0,401	1,002	1,008
10	30	0,411	1,001	1,005
10	40	1	1	1
10	50	1	1	1
10	100	1	1	1
10	1000	1	1	1
15	0,01	0,136	1,07	1,284
15	1	0,136	1,07	1,284
15	5	0,141	1,033	1,134
15	10	0,161	0,998	0,992
15	20	0,179	1,02	1,08
15	30	0,179	1,02	1,08
15	40	0,179	0,976	0,902
15	50	0,179	0,976	0,902
15	100	0,194	0,991	0,963
15	1000	0,194	0,991	0,963
20	0,01	0,131	1,094	1,384
20	1	0,131	1,094	1,384
20	5	0,136	1,104	1,425
20	10	0,144	1,046	1,186
20	20	0,139	1,003	1,014
20	30	0,139	1,003	1,014
20	40	0,144	1,031	1,127
20	50	0,144	1,031	1,127
20	100	0,151	1,039	1,157
20	1000	0,151	1,039	1,157
30	0,01	0,119	1,117	1,476
30	1	0,119	1,117	1,476
30	5	0,126	1,177	1,72
30	10	0,131	1,155	1,631
30	20	0,129	1,113	1,46
30	30	0,129	1,113	1,46
30	40	0,134	1,126	1,511
30	50	0,134	1,126	1,511
30	100	0,139	1,167	1,679
30	1000	0,139	1,167	1,679
50	0,01	0,114	1,219	1,892
50	1	0,114	1,219	1,892
50	5	0,114	1,2	1,815
50	10	0,116	1,184	1,748
50	20	0,116	1,187	1,761
50	30	0,116	1,187	1,761
50	40	0,116	1,185	1,754
50	50	0,116	1,185	1,754
50	100	0,116	1,167	1,679
50	1000	0,116	1,167	1,679

Figure 7: Method 3 - Ratio table for W = 10 and M/m = 10

- Else, we calculate E[V_t+1] thanks to the PDF and if V_t>E[V_t+1], then we send.

$$E[V_{t+1}]=M(t+1)q+(M(t+1)-B)(1-q)$$

with q the probability w don't exceed the threshold, calculated thanks to the PDF $q=P(\theta>S_{t+1})$

Then, if we calculated E[V_t+1] , it means we didn't exceed the threshold and $V_t=Mt$.

Therefore, our condition to send is $Mt>M(t+1)-B(1-q)$

Which simplifies to $q<1-M/B$

From that, we can define 2 extreme cases :

- If M > B, then the condition is never met, and we never send in advance, but only when S exceeds the threshold. This is the 2nd method.
- If B >> M, then the condition is always met, and we always send. This is the baseline model.

This method allows us to move from the baseline model to the second one using the parameter B. It is actually an improvement of the last method which gives us a wider range of parameter from which we might pick an even better solution.
Whenever we use it, when B goes up, the cost goes up as well and the accuracy gets better as we evolve from the 2nd model that send less to the baseline solution which always send.

Then, for a fixed W, M and m, we can imagine choosing a higher θ value (which means sending less) but using the penalty B to counterbalance the rise of θ to somehow reach a better compromise.
We can see an example with the highlighted rows on Fig.7.
It is very hard to determine from these data alone if the compromise will always be better as it highly depend on the situation (data set and parameters).

However, it is only an improvement of the last model and we can just set B to a small value to retrieve the 2nd model.

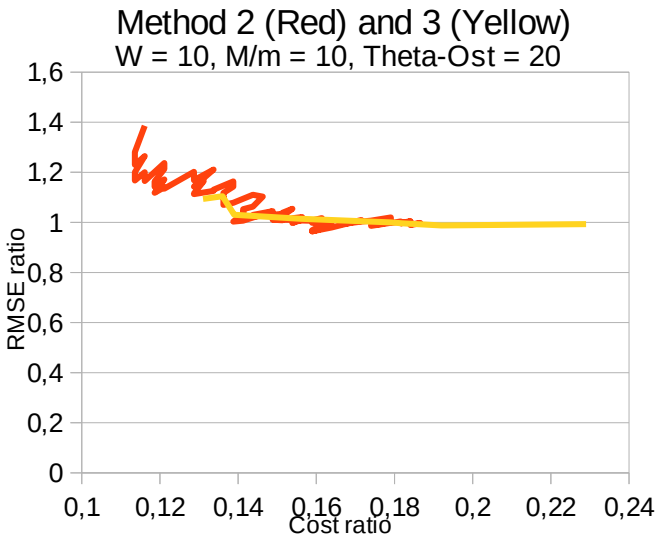

Figure 8: Method 2 and 3 comparison - RMSE for cost

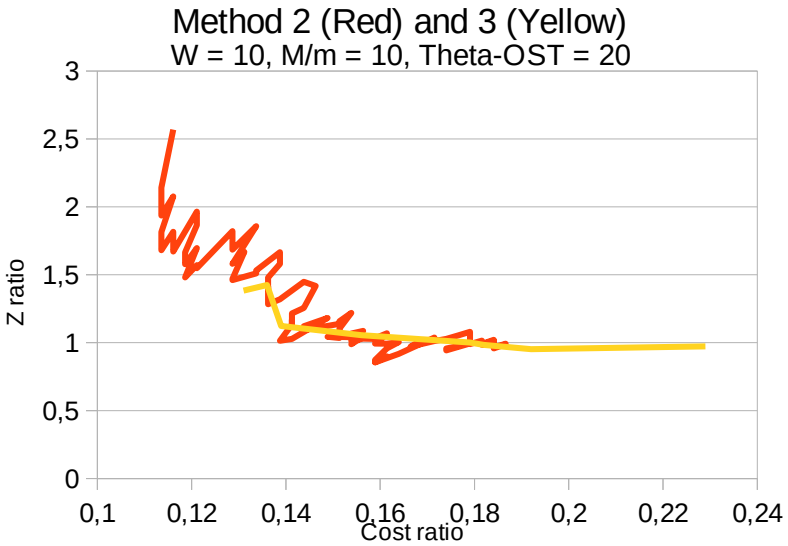

Figure 9: Method 2 and 3 comparison - Discrepancy for cost

3.2) Method 3-bis : Application of OST theory-bis

While defining the method 3, we used the simplest gain V possible, but there are other possibilities to define this gain. Here is an other one that we will call method 3-bis.

- $V_t = (M-m)t$ if we don't exceed the threshold at time t.
- $V_t = -B-(M+m)$ else with B a penalty that we define.

Then, $E[V_{t+1}] = (M-m)(t+1)q-(B+M+m)(1-q)$

with still $q = P(\theta > S_{t+1})$

Then, if we calculated $E[V_t+1]$, it means we didn't exceed the threshold and $V_t = (M-m)t$.

Therefore, our condition to send is $(M-m)t > (M-m)(t+1)q-(B+M+m)(1-q)$

Which simplifies to $q < \frac{(M-m)t+(B+M+m)}{(M-m)(t+1)+(B+M+m)}$

The results obtained using this method are only slightly different from those obtained with the last method (see highlighted rows).

It can be worse or better but we can't make a rule out of it as it highly depends on the situations and the data.

So if a better compromised can be reached by the third method, it will probably also be possible to find it with this method, maybe using different parameters and obtaining a slightly different result.

Modifying the gain V might be a lead to follow if we ever want to go further, but for the rest of this study, we will choose to keep only the third method as it is simpler and really close to this one.

Theta-OST	B/M	COST-RATIO	RMSE-RATIO	Z-RATIO	COST-RATIO	RMSE-RATIO	Z-RATIO
10	0,01	0,144	1,03	1,121	0,237	0,998	0,993
10	1	0,144	1,03	1,121	0,252	1,003	1,014
10	5	0,159	0,967	0,865	0,403	1,024	1,096
10	10	0,295	1,005	1,021	0,713	1,005	1,021
10	20	0,401	1,002	1,008	1	1	1
10	30	0,411	1,001	1,005	1	1	1
10	40	1	1	1	1	1	1
10	50	1	1	1	1	1	1
10	100	1	1	1	1	1	1
10	1000	1	1	1	1	1	1
15	0,01	0,136	1,07	1,284	0,177	1,011	1,044
15	1	0,136	1,07	1,284	0,177	1,004	1,018
15	5	0,141	1,033	1,134	0,194	1,013	1,053
15	10	0,161	0,998	0,992	0,202	0,989	0,957
15	20	0,179	1,02	1,08	0,227	1,017	1,071
15	30	0,179	1,02	1,08	0,237	0,994	0,976
15	40	0,179	0,976	0,902	0,242	1,006	1,023
15	50	0,179	0,976	0,902	0,245	0,996	0,983
15	100	0,194	0,991	0,963	0,403	1,013	1,052
15	1000	0,194	0,991	0,963	1	1	1
20	0,01	0,131	1,094	1,384	0,149	1,027	1,109
20	1	0,131	1,094	1,384	0,149	1,027	1,112
20	5	0,136	1,104	1,425	0,146	1,005	1,019
20	10	0,144	1,046	1,186	0,146	0,976	0,901
20	20	0,139	1,003	1,014	0,149	0,972	0,886
20	30	0,139	1,003	1,014	0,151	0,98	0,919
20	40	0,144	1,031	1,127	0,159	0,999	0,998
20	50	0,144	1,031	1,127	0,154	0,979	0,915
20	100	0,151	1,039	1,157	0,184	0,991	0,963
20	1000	0,151	1,039	1,157	0,224	1,011	1,044
30	0,01	0,119	1,117	1,476	0,136	1,153	1,624
30	1	0,119	1,117	1,476	0,139	1,168	1,683
30	5	0,126	1,177	1,72	0,136	1,138	1,56
30	10	0,131	1,155	1,631	0,136	1,136	1,555
30	20	0,129	1,113	1,46	0,136	1,136	1,555
30	30	0,129	1,113	1,46	0,136	1,113	1,458
30	40	0,134	1,126	1,511	0,136	1,094	1,383
30	50	0,134	1,126	1,511	0,136	1,094	1,381
30	100	0,139	1,167	1,679	0,139	1,07	1,284
30	1000	0,139	1,167	1,679	0,144	1,064	1,259
50	0,01	0,114	1,219	1,892	0,119	1,193	1,784
50	1	0,114	1,219	1,892	0,119	1,193	1,784
50	5	0,114	1,2	1,815	0,116	1,152	1,619
50	10	0,116	1,184	1,748	0,116	1,152	1,619
50	20	0,116	1,187	1,761	0,116	1,152	1,62
50	30	0,116	1,187	1,761	0,119	1,173	1,705
50	40	0,116	1,185	1,754	0,119	1,173	1,705
50	50	0,116	1,185	1,754	0,116	1,14	1,57
50	100	0,116	1,167	1,679	0,119	1,155	1,632
50	1000	0,116	1,167	1,679	0,119	1,149	1,608

Figure 10: Ratio table comparison between method 3 and 3-bis

4.) Method 4 : Discount factor Beta

We define a last method where we use a new parameter β which is between 0 and 1.
Then we use the sum of the discrepancies S that we defined earlier and the PDF function as well.

Each W time instance, we want to send only if
$S > \beta/(1-\beta) * E[Z]$

It is actually a way to replace our older θ factor with something that is directly linked to the actual values of the discrepancy. The performance are actually exactly the same since we can reach the same θ values.

Its real utility is to replace a very arbitrary value such as θ with the β factor which is linked to the accuracy.
Indeed, one β value will give close error ratios with different W and/or data sets while the same θ can make use always or never send depending on the data set and W.

Here are results with M/m = 10.
We can see that for the same β values, RMSE and Z ratios are pretty close.

W	Beta	COST-RATIO	RMSE-RATIO	Z-RATIO
10	0,1	0,574	1,006	1,023
10	0,5	0,25	1	0,999
10	0,8	0,154	1,054	1,22
10	0,85	0,144	1,064	1,259
10	0,9	0,129	1,113	1,46
10	0,95	0,116	1,216	1,88
30	0,1	0,492	1,004	1,027
30	0,5	0,242	1,053	1,33
30	0,8	0,167	1,136	1,842
30	0,85	0,159	1,232	2,438
30	0,9	0,152	1,345	3,135
30	0,95	0,129	1,49	4,034
50	0,1	0,495	1,015	1,1
50	0,5	0,255	1,099	1,666
50	0,8	0,205	1,352	3,367
50	0,85	0,179	1,281	2,888
50	0,9	0,167	1,435	3,921
50	0,95	0,141	1,679	5,561
100	0,1	0,571	1,002	1,01
100	0,5	0,343	1,109	1,46
100	0,8	0,217	1,219	1,921
100	0,85	0,217	1,336	2,413
100	0,9	0,192	1,388	2,634
100	0,95	0,167	1,647	3,726

Figure 11: Method 4 - Ratio table

It is a better way to do what we did before with θ.
This method is actually equivalent to the 2nd method as we only use the sum of the discrepancies S.
But we can imagine replacing θ by β in the OST method as well.

5.) The decision factor : using the error instead of the discrepancy

One alternative to all the models presented earlier is to directly use the flat RMSE e0[t] instead of the discrepancy Z to make the decision to send and define a threshold θ for the RMSE.

In fact , we can apply every single method until now using the RMSE instead of Z.
It is an alternative way to evaluate the error and make the decision but it has the interesting particularity to not need the current model.

Then, each W time instances, we can make the decision of sending or not and compute the model only if we need to send, it saves us the computation cost every time we actually don't send.

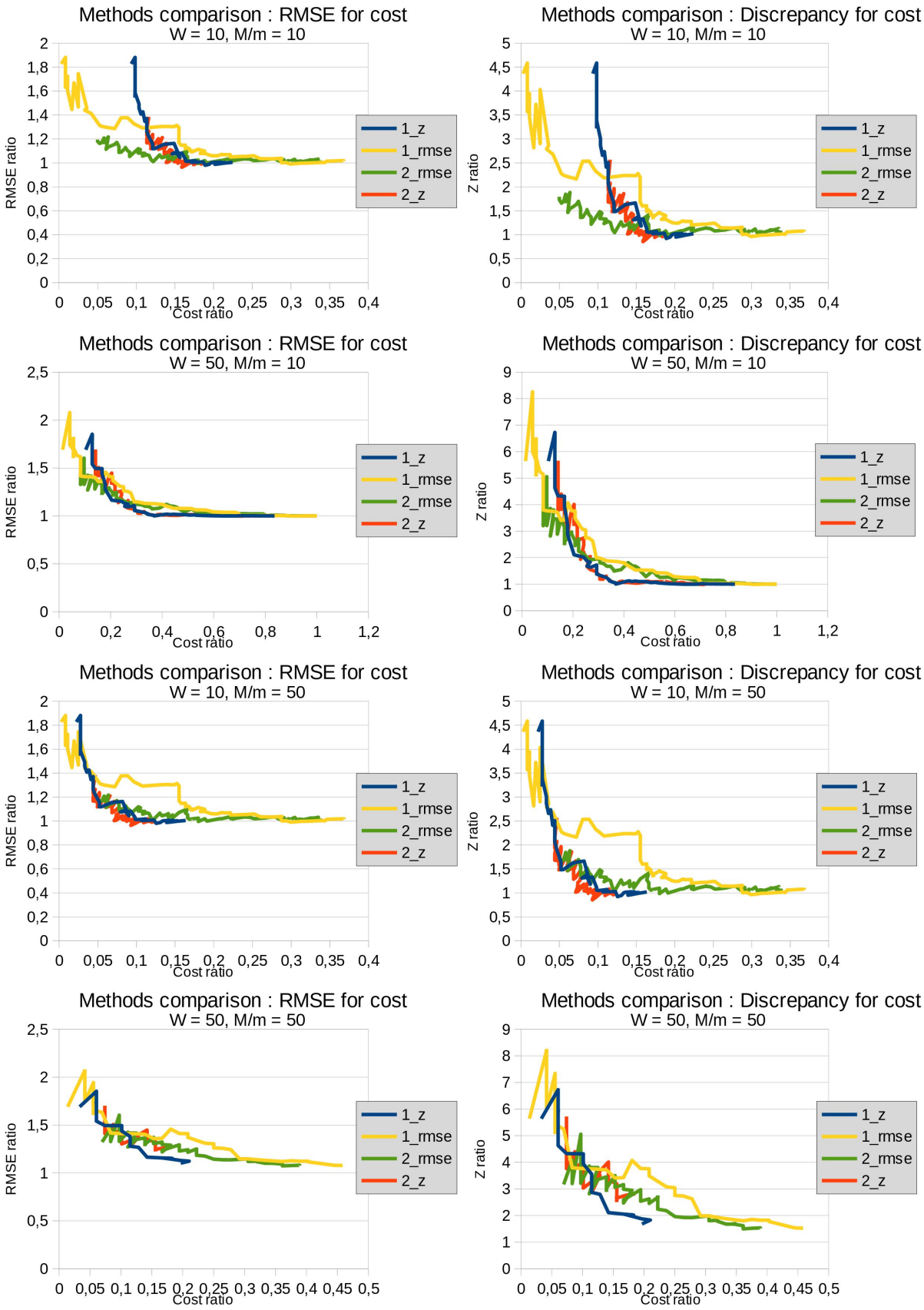

Figure 12: Comparison of RMSE vs Discrepancy

The first thing we can notice is that theses plots confirm some fact we established earlier with the ratio tables. For instance, it is way harder to tell which model is the best for a high W and we can also confirm that the compromise are overall way better using a small W.

Aside from that, we can see that using the 1st method with the RMSE seems to always be the worst choice. However, it becomes interesting to use the RMSE with the 2nd method in some situations. Indeed, when M and m are not too different, it gets really interesting to use the RMSE as we suppress the systematic computation cost. Here we can see that using the RMSE is better when M/m = 10 but that it is actually better to use the discrepancy Z when M/m = 50 .

According to our simulation, we can conclude that it is better to use the RMSE when M/m is not too high.

However, this result is to take in regards of the limitations of our simulation. Indeed, we don't take into account the computation cost needed to make the decision to send or not which might be relevant here.

Conclusion

To conclude on what we learned, let's define rules from the fact we established, we will then be able to test the validity of theses rules with other data sets. To be able to test them easily, we will replace θ by β in every method the same way we did in what was our 4th method.

RULES

1 : The best compromise is always found for a small W.

2 : The 2nd model is better than the 1st for a small W.

3 : The 3rd model can reach a better compromise than the 2nd.

4 : The models provide similar ratios for one β value.

5 : Using the RMSE provides better result if M/m is not too big.

II) EXPERIMENTING WITH DIFFERENT DATA SETS

Up until now, we used two values of a data set presenting a concept drift[1] to build our models. It was the first one we used, but we have to test our methods with other data sets and compare our results.

To experiment with other data sets, we will replace the θ parameter which was highly linked to the actual values by the β parameter as described in the 4th method. We will do so in every method and stop generating ratio table for the 4th method which will become the 2nd.

For this part, we won't show all of the results since the amount of data is significantly higher than before. We will only discuss the rules and their validity one by one by looking at the relevant plots. To look at the actual results, please refer to the data files linked to this report.

1.) Second data set : Integer values

The second data set[2] we are using to compare our result is composed of integer values of humidity and temperature which are linked through a model. Since these data are integers, they have very small variations, the values are increasing 1 by 1 and often remain constant, which makes it a really peculiar data set.

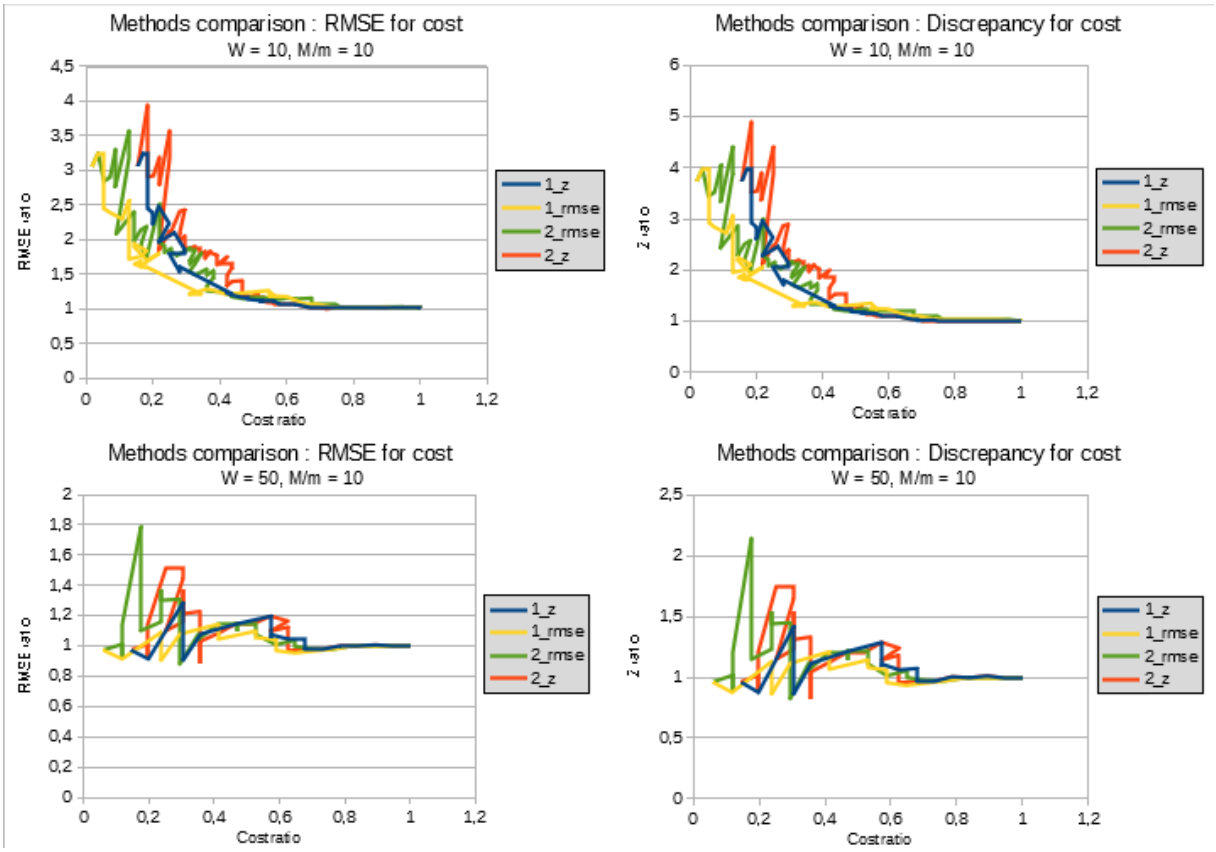

Figure 13: Second dataset - Methods comparison with M/m = 10

Even the first rule doesn't seem to hold true for this data set. Indeed, when W gets too high, the models built get really strange and we often have a better precision while not sending, but it is a completely random phenomenon linked to the fact that we are using integers, which creates bad models using the linear regression with too much values. So it is not a better compromise since the outcome will be completely random. So the first rule does hold true even for this peculiar data set.

The second rule, however, seems to be proven false. It looks like using the 1st model will always be better with this data set. We can understand this phenomenon better when we look at the data set. The values remain constant and suddenly change at one time instance. So the discrepancy is often zero and suddenly gets high when the values change. This is detected by the 1st model but not necessarily by the 2nd which use the sum of the discrepancies. We can conclude that the 1st model can be better when we deal with a data that knows sudden variations, which is always the case with integer values.

The third rule encounter the same problem than the second, the distribution of the error is not uniform over the data. Therefore, the 3rd method using the PDF and trying to predict the discrepancy/ error values doesn't work well. So using it is also pretty random. But the rule as it is stays true, as it is possible to find a better compromise, even if it will be a random occurrence.

Once again, the fourth rule encounter the same problem, the error values are pretty random, therefore hardly linked to the value. So it is not true for this data set.

However, we can observe on the plots that the fifth rule stays true, RMSE will always be better when M/m gets small.
Let's update the rules with these models. For that, we will add one exception as this is a really peculiar data set.

RULES

1 : The best compromise is always found for a small W.

If the data values over time look continuous *then :

2-1 : The 2nd model is better than the 1st for a small W.

2-2 : The models provide similar ratios for one β value.

If not, and the data values are not continuous :

2-3 : The 1st model will always be the best.

3 : The 3rd model can reach a better compromise than the 2nd.

4 : Using the RMSE provides better result if M/m is not too big.

* Here, looking continuous means that the values are precise enough to avoid huge discrepancies between two values due to the sampling of the analogical data.

2.) Third data set : Linear relation

Here, we are looking at a data set linking power and voltage in a household installation[3].
When plotted, the relation between these two values is almost linear, or it should be but it isn't because of the noise and errors of the measurements. It could be seen as the reverse of the previous data set as this one is almost too precise.
The consequences are that not sending the model will provide a better error than sending it and the results show error ratio that are below 1. However, despite this phenomenon, does our rules holds true ?

RMSE and discrepancy ratio below 1 are observed only for a low W. When the time window is small enough, not sending means not taking into account local variations and therefore having a better precision when the values have a linear relation. So the first rule holds true, the best compromise is indeed at a low W since we can have a lower cost while also lowering the error.

Then, we can observe a similar phenomenon than with the second data set, in these conditions, the first model seems to always be the best and one β value does not always give a similar error as we get a ratio below 1 in some situations. So we can say it follow the rules if we change our conditions for the rules definitions and adapt it to this new case.

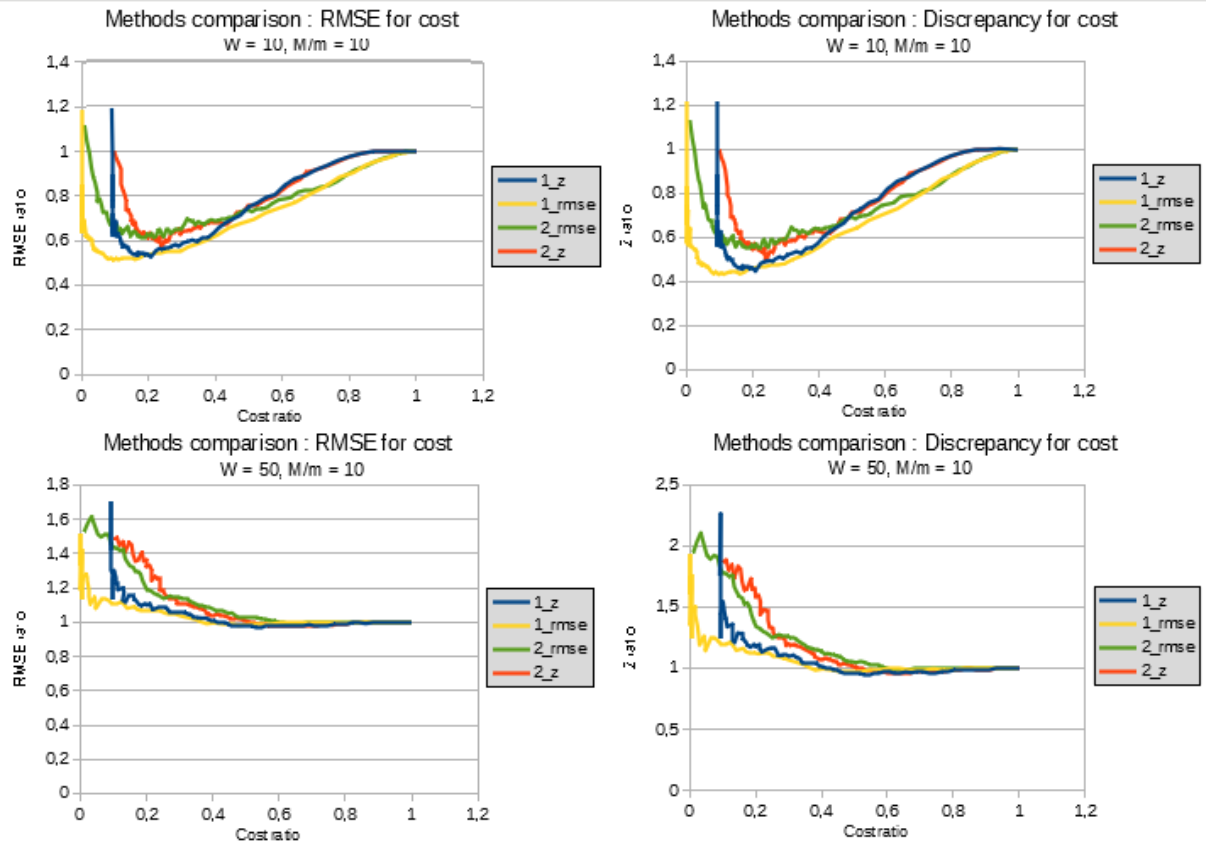

Figure 14: Third dataset - Methods comparison with M/m = 10

About the 3rd model, there are some cases where it can indeed reach a better compromise even if those seems to be harder to find as a penalty will quickly lead to the situation where we always send, most likely because we have a high probability of exceeding the threshold with the random noise. But the rule does hold true.

Finally, the last rule seems to be exactly the same. The RMSE will always be better for a small M/m ratio.

To update the rules, we only need to change our condition about « looking continuous » to make it more precise regarding these new results.
In our case, not « being continuous » occurs when the data presents high local variations(often due to noise or a sampling that's not precise enough) that might create local models far away from the expected models and values in these situations.

3.) Multi-variable data sets

To expand our study, we experimented with multi-variable data sets. To do so, we used the exact same data sets than before, but without reducing them to two values. We used the drift data set[1] with 8 values and the household data set[3] with 4 values, always using the last value as the target of the model and all the others as the parameters. By doing so, we could validate the rules in both the continuous and not continuous cases. It turns out that even if the plot and the results are obviously slightly different, all the rules hold true and the models behave in the same way than with a single parameter.

CONCLUSION

Finally, we can settle on the following rules on which model to use :

RULES

1 : The best compromise is always found for a small W.

2 : The 3rd model can reach a better compromise than the 2nd.

3 : Using the RMSE provides better result if M/m is not too big.

Then, looking at the values, we need to evaluate if they can be considered continuous, i.e. they do not presents high local variations linked to the sensors precision or noise and unrelated to the actual expected values.

If the values over time are looking continuous:

4-1 : The 2nd model is better than the 1st for a small W.

4-2 : The models provide similar ratios for one β value.

If not :

5 : The 1st model will always be the best.

Besides, we have to keep in mind that the best solutions is not the same in every situation and finding it will require an early study of the problem, which will also be necessary to establish the best parameters to use.

To go further, we could define more precisely the computation cost which also includes the computation made to make the decision and not only the cost to compute the model.
We could also experiment with the R2 or other errors instead of the RMSE and see if it gives better results.
Finally, the CDF built in our simulation use the very first model to get all the discrepancies as it is built before the real simulation starts. We could build it in the same way as in an actual node by using a machine learning model to get a more accurate simulation.

All the results that were used to draw these conclusions can be found attached to this report. It includes all the ratio tables and the data used to plot the method comparison of the first data set using θ and from all the data sets using β.

E. Data Sets & Resources

[1] :Gas Sensor Array Drift Data set at Different Concentrations Data Set :http://archive.ics.uci.edu/ml/datasets/Gas+Sensor+Array+Drift+Dataset+at+Different+Concentrations
[2] :GNFUV Unmanned Surface Vehicles Sensor Data Set 2 Data Set :https://archive.ics.uci.edu/ml/datasets/GNFUV+Unmanned+Surface+Vehicles+Sensor+Data+Set+2
[3] :Individual household electric power consumption Data Set :http://archive.ics.uci.edu/ml/datasets/Individual+household+electric+power+consumption
[4] :The WEKA library : https://www.cs.waikato.ac.nz/ml/weka/

ACKNOWLEDGMENT

My deepest thanks to the University of Glasgow for welcoming me for this internship.

I would like to express my sincere gratitude toward my tutor Dr.Christos Agnastopoulos for their guidance, comments and suggestions throughout the course of the project.