

```
load_data = true
```

```
load_data = logical
1
```

```
% save_data_from_scratch = true
```

```
save_data_from_scratch = logical
1
```

```
save_data_from_scratch = false
if load_data
    load ryan5_battery_dataset.mat
end
```

```
length(ryan5_battery_dataset)
```

```
ans = 119
```

```
size(ryan5_battery_dataset)
```

```
ans = 1x2
      1   119
```

```
[ryan5_battery_dataset.key]
```

```
ans = 1x119 string
"b1c6"      "b1c7"      "b1c8"      "b1c10"     "b1c12"     "b1c15"     "b . . .
```

```
{[ryan5_battery_dataset.key]}
```

```
ans = 1x1 cell array
      {"b1c6"      "b1c7"      "b1c8"      "b1c10"     "b1c12"     "b1c15"     "b1c16"     "b1c17"     "b1c18"     "b1c19"     "b1c20"     "b1c21"     "b1c22"     "b1c23"     "b1c24"     "b1c25"     "b1c26"     "b1c27"     "b1c28"     "b1c29"     "b1c30"     "b1c31"     "b1c32"     "b1c33"     "b1c34"     "b1c35"     "b1c36"     "b1c37"     "b1c38"     "b1c39"     "b1c40"     "b1c41"     "b1c42"     "b1c43"     "b1c44"     "b1c45"     "b1c46"     "b1c47"     "b1c48"     "b1c49"     "b1c50"     "b1c51"     "b1c52"     "b1c53"     "b1c54"     "b1c55"     "b1c56"     "b1c57"     "b1c58"     "b1c59"     "b1c60"     "b1c61"     "b1c62"     "b1c63"     "b1c64"     "b1c65"     "b1c66"     "b1c67"     "b1c68"     "b1c69"     "b1c70"     "b1c71"     "b1c72"     "b1c73"     "b1c74"     "b1c75"     "b1c76"     "b1c77"     "b1c78"     "b1c79"     "b1c80"     "b1c81"     "b1c82"     "b1c83"     "b1c84"     "b1c85"     "b1c86"     "b1c87"     "b1c88"     "b1c89"     "b1c90"     "b1c91"     "b1c92"     "b1c93"     "b1c94"     "b1c95"     "b1c96"     "b1c97"     "b1c98"     "b1c99"     "b1c100"     "b1c101"     "b1c102"     "b1c103"     "b1c104"     "b1c105"     "b1c106"     "b1c107"     "b1c108"     "b1c109"     "b1c110"     "b1c111"     "b1c112"     "b1c113"     "b1c114"     "b1c115"     "b1c116"     "b1c117"     "b1c118"     "b1c119"     "b1c120"     "b1c121"     "b1c122"     "b1c123"     "b1c124"     "b1c125"     "b1c126"     "b1c127"     "b1c128"     "b1c129"     "b1c130"     "b1c131"     "b1c132"     "b1c133"     "b1c134"     "b1c135"     "b1c136"     "b1c137"     "b1c138"     "b1c139"     "b1c140"     "b1c141"     "b1c142"     "b1c143"     "b1c144"     "b1c145"     "b1c146"     "b1c147"     "b1c148"     "b1c149"     "b1c150"     "b1c151"     "b1c152"     "b1c153"     "b1c154"     "b1c155"     "b1c156"     "b1c157"     "b1c158"     "b1c159"     "b1c160"     "b1c161"     "b1c162"     "b1c163"     "b1c164"     "b1c165"     "b1c166"     "b1c167"     "b1c168"     "b1c169"     "b1c170"     "b1c171"     "b1c172"     "b1c173"     "b1c174"     "b1c175"     "b1c176"     "b1c177"     "b1c178"     "b1c179"     "b1c180"     "b1c181"     "b1c182"     "b1c183"     "b1c184"     "b1c185"     "b1c186"     "b1c187"     "b1c188"     "b1c189"     "b1c190"     "b1c191"     "b1c192"     "b1c193"     "b1c194"     "b1c195"     "b1c196"     "b1c197"     "b1c198"     "b1c199"     "b1c200"     "b1c201"     "b1c202"     "b1c203"     "b1c204"     "b1c205"     "b1c206"     "b1c207"     "b1c208"     "b1c209"     "b1c210"     "b1c211"     "b1c212"     "b1c213"     "b1c214"     "b1c215"     "b1c216"     "b1c217"     "b1c218"     "b1c219"     "b1c220"     "b1c221"     "b1c222"     "b1c223"     "b1c224"     "b1c225"     "b1c226"     "b1c227"     "b1c228"     "b1c229"     "b1c230"     "b1c231"     "b1c232"     "b1c233"     "b1c234"     "b1c235"     "b1c236"     "b1c237"     "b1c238"     "b1c239"     "b1c240"     "b1c241"     "b1c242"     "b1c243"     "b1c244"     "b1c245"     "b1c246"     "b1c247"     "b1c248"     "b1c249"     "b1c250"     "b1c251"     "b1c252"     "b1c253"     "b1c254"     "b1c255"     "b1c256"     "b1c257"     "b1c258"     "b1c259"     "b1c260"     "b1c261"     "b1c262"     "b1c263"     "b1c264"     "b1c265"     "b1c266"     "b1c267"     "b1c268"     "b1c269"     "b1c270"     "b1c271"     "b1c272"     "b1c273"     "b1c274"     "b1c275"     "b1c276"     "b1c277"     "b1c278"     "b1c279"     "b1c280"     "b1c281"     "b1c282"     "b1c283"     "b1c284"     "b1c285"     "b1c286"     "b1c287"     "b1c288"     "b1c289"     "b1c290"     "b1c291"     "b1c292"     "b1c293"     "b1c294"     "b1c295"     "b1c296"     "b1c297"     "b1c298"     "b1c299"     "b1c300"     "b1c301"     "b1c302"     "b1c303"     "b1c304"     "b1c305"     "b1c306"     "b1c307"     "b1c308"     "b1c309"     "b1c310"     "b1c311"     "b1c312"     "b1c313"     "b1c314"     "b1c315"     "b1c316"     "b1c317"     "b1c318"     "b1c319"     "b1c320"     "b1c321"     "b1c322"     "b1c323"     "b1c324"     "b1c325"     "b1c326"     "b1c327"     "b1c328"     "b1c329"     "b1c330"     "b1c331"     "b1c332"     "b1c333"     "b1c334"     "b1c335"     "b1c336"     "b1c337"     "b1c338"     "b1c339"     "b1c340"     "b1c341"     "b1c342"     "b1c343"     "b1c344"     "b1c345"     "b1c346"     "b1c347"     "b1c348"     "b1c349"     "b1c350"     "b1c351"     "b1c352"     "b1c353"     "b1c354"     "b1c355"     "b1c356"     "b1c357"     "b1c358"     "b1c359"     "b1c360"     "b1c361"     "b1c362"     "b1c363"     "b1c364"     "b1c365"     "b1c366"     "b1c367"     "b1c368"     "b1c369"     "b1c370"     "b1c371"     "b1c372"     "b1c373"     "b1c374"     "b1c375"     "b1c376"     "b1c377"     "b1c378"     "b1c379"     "b1c380"     "b1c381"     "b1c382"     "b1c383"     "b1c384"     "b1c385"     "b1c386"     "b1c387"     "b1c388"     "b1c389"     "b1c390"     "b1c391"     "b1c392"     "b1c393"     "b1c394"     "b1c395"     "b1c396"     "b1c397"     "b1c398"     "b1c399"     "b1c400"     "b1c401"     "b1c402"     "b1c403"     "b1c404"     "b1c405"     "b1c406"     "b1c407"     "b1c408"     "b1c409"     "b1c410"     "b1c411"     "b1c412"     "b1c413"     "b1c414"     "b1c415"     "b1c416"     "b1c417"     "b1c418"     "b1c419"     "b1c420"     "b1c421"     "b1c422"     "b1c423"     "b1c424"     "b1c425"     "b1c426"     "b1c427"     "b1c428"     "b1c429"     "b1c430"     "b1c431"     "b1c432"     "b1c433"     "b1c434"     "b1c435"     "b1c436"     "b1c437"     "b1c438"     "b1c439"     "b1c440"     "b1c441"     "b1c442"     "b1c443"     "b1c444"     "b1c445"     "b1c446"     "b1c447"     "b1c448"     "b1c449"     "b1c450"     "b1c451"     "b1c452"     "b1c453"     "b1c454"     "b1c455"     "b1c456"     "b1c457"     "b1c458"     "b1c459"     "b1c460"     "b1c461"     "b1c462"     "b1c463"     "b1c464"     "b1c465"     "b1c466"     "b1c467"     "b1c468"     "b1c469"     "b1c470"     "b1c471"     "b1c472"     "b1c473"     "
```

```
contains([ryan5_battery_dataset.key], 'b1', 'IgnoreCase', true)
```

[illegible]

```
contains([ryan5_battery_dataset.key], 'b2', 'IgnoreCase', true)
```

[illegible]

```
contains([ryan5_battery_dataset.key], 'b3', 'IgnoreCase', true)
```

[illegible]

```
idx1 = find(contains([ryan5_battery_dataset.key], 'b1', 'IgnoreCase', true))
```

```
idx1 = 1×36
      1      2      3      4      5      6      7      8      9     10     11     12     13 ...
```

```
idx2 = find(contains([ryan5_battery_dataset.key], 'b2', 'IgnoreCase', true))
```

```
idx2 = 1x43
```

37 38 39 40 41 42 43 44 45 46 47 48 49 ...

```
idx3 = find(contains([ryan5_battery_dataset.key], 'b3', 'IgnoreCase', true))
```

```
idx3 = 1x40
80 81 82 83 84 85 86 87 88 89 90 91 92 ...
```

```
idx12 = [idx1 idx2]
```

```
idx12 = 1x79
1 2 3 4 5 6 7 8 9 10 11 12 13 ...
```

```
idx13 = [idx1 idx3]
```

```
idx13 = 1x76
1 2 3 4 5 6 7 8 9 10 11 12 13 ...
```

```
idx23 = [idx2 idx3]
```

```
idx23 = 1x83
37 38 39 40 41 42 43 44 45 46 47 48 49 ...
```

```
idxAll = (1:length(ryan5_battery_dataset))
```

```
idxAll = 1x119
1 2 3 4 5 6 7 8 9 10 11 12 13 ...
```

```
idx123 = [idx1 idx2 idx3]
```

```
idx123 = 1x119
1 2 3 4 5 6 7 8 9 10 11 12 13 ...
```

```
key1 = [ryan5_battery_dataset(idx1).key]';
key2 = [ryan5_battery_dataset(idx2).key]';
key3 = [ryan5_battery_dataset(idx3).key]';
cycle_life = [ryan5_battery_dataset(idxAll).cycle_life]';
bat_index = [1:length(idxAll)]';
index_cyclelife = [ bat_index, cycle_life];
% https://kr.mathworks.com/help/matlab/ref/double.sortrows.html#bt8bz9j-3
index_cyclelife_table = table(bat_index, cycle_life)
```

index_cyclelife_table = 119x2 table

	bat_index	cycle_life
1	1	428
2	2	229
3	3	253
4	4	338
5	5	416
6	6	331
7	7	278

	bat_index	cycle_life
8	8	235
9	9	284
10	10	462
11	11	269
12	12	379
13	13	300
14	14	354

⋮

```
[index_cyclelife_table_sort,index] = sortrows(index_cyclelife_table',{'cycle_life'},{'ascend'})
```

```
index_cyclelife_table_sort = 119x2 table
```

	bat_index	cycle_life
1	38	109
2	78	115
3	71	118
4	69	129
5	65	137
6	76	140
7	61	146
8	74	146
9	73	152
10	66	153
11	70	153
12	64	154
13	68	155
14	75	157

⋮

```
index = 119x1
```

```
38
78
71
69
65
76
61
74
73
66
⋮
```

```
%%%%%%%%%
```

```
total_index = length(idxAll)
```

```
total_index = 119
```

```
index_cyclelife_table_sort.Variables
```

```
ans = 119x2
```

```
38 109
78 115
71 118
69 129
65 137
76 140
61 146
74 146
73 152
66 153
:
```

```
index_cyclelife_table_sort.Properties
```

```
ans =
```

```
TableProperties with properties:
```

```
    Description: ''
    UserData: []
    DimensionNames: {'Row' 'Variables'}
    VariableNames: {'bat_index' 'cycle_life'}
    VariableDescriptions: {}
    VariableUnits: {}
    VariableContinuity: []
    RowNames: {}
    CustomProperties: No custom properties are set.
    Use addprop and rmprop to modify CustomProperties.
```

```
[~, maxrow] = max(index_cyclelife_table_sort.bat_index)
```

```
maxrow = 116
```

```
%maxrowname = index_cyclelife_table_sort.Properties.RowNames(1:maxrow-1)
```

```
%index_cyclelife_table_sort.Properties.RowNames
```

```
index_vec = [1:total_index]
```

```
index_vec = 1x119
```

```
1 2 3 4 5 6 7 8 9 10 11 12 13 ...
```

```
idxVal_ = index_vec(5:10:end)
```

```
idxVal_ = 1x12
```

```
5 15 25 35 45 55 65 75 85 95 105 115
```

```
idxTest_ = index_vec(6:10:end)
```

```
idxTest_ = 1x12
```

```
6 16 26 36 46 56 66 76 86 96 106 116
```

```
idxTrain_ = setdiff(index_vec,idxVal_);
idxTrain_ = setdiff(idxTrain_,idxTest_)
```

```
idxTrain_ = 1×95
    1     2     3     4     7     8     9    10    11    12    13    14    17...
```

```
idxTestTable = index_cyclelife_table_sort(idxTest_,{'bat_index'})
```

```
idxTestTable = 12×1 table
```

	bat_index
1	76
2	48
3	62
4	28
5	26
6	47
7	16
8	99
9	92
10	106
11	84
12	119

```
idxTrainTable = index_cyclelife_table_sort(idxTrain_,{'bat_index'})
```

```
idxTrainTable = 95×1 table
```

	bat_index
1	38
2	78
3	71
4	69
5	61
6	74
7	73
8	66
9	70
10	64
11	68
12	75
13	52

	bat_index
14	60

⋮

```
idxValTable = index_cyclelife_table_sort(idxVal_,{'bat_index'})
```

```
idxValTable = 12×1 table
```

	bat_index
1	65
2	63
3	56
4	59
5	40
6	11
7	46
8	14
9	94
10	10
11	100
12	115

```
idxTest = idxTestTable.Variables'
```

```
idxTest = 1×12
    76    48    62    28    26    47    16    99    92   106    84   119
```

```
idxValid = idxValTable.Variables'
```

```
idxValid = 1×12
    65    63    56    59    40    11    46    14    94    10   100   115
```

```
idxTrain = idxTrainTable.Variables'
```

```
idxTrain = 1×95
    38    78    71    69    61    74    73    66    70    64    68    75    52 ...
```

```
idxAll
```

```
idxAll = 1×119
     1     2     3     4     5     6     7     8     9    10    11    12    13 ...
```

```
idxTrainSort = sort(idxTrain)
```

```
idxTrainSort = 1×95
     1     2     3     4     5     6     7     8     9    12    13    15    17 ...
```

```
idxValidSort = sort(idxValid)
```

```
idxValidSort = 1×12
    10    11    14    40    46    56    59    63    65    94   100   115
```

```
idxTestSort = sort(idxTest)
```

```
idxTestSort = 1×12
    16    26    28    47    48    62    76    84    92    99   106   119
```

```
%https://kr.mathworks.com/matlabcentral/answers/101996-how-can-i-sort-a-vector-in-a-random-manner
```

```
idxTestRand = idxTest(randperm(length(idxTest)))
```

```
idxTestRand = 1×12
    28    99    48   106    92   119    76    26    84    47    16    62
```

```
idxTrainRand = idxTrain(randperm(length(idxTrain)))
```

```
idxTrainRand = 1×95
    34    41    35    55    30    44     2    24    39    22    61    70    69 ...
```

```
idxValidRand = idxValid(randperm(length(idxValid)))
```

```
idxValidRand = 1×12
    63    10    94    56    11   100   115    59    65    40    46    14
```

```
%sortrows(idxValTable,{'cycle_life'},{'ascend'})
input_size = 100;
```

```
ryan5_gen_data_desc = 'ryan5_data_feature';
if save_data_from_scratch
save('ryan5_data.mat','ryan5_gen_data_desc');
end

% ryan5_battery_dataset_raw 는 지워도 된다.
%save('ryan5_data.mat','ryan5_battery_dataset_raw' ,'-append')
%save('ryan5_data.mat','data_conversion_param' ,'-append')
save('ryan5_data.mat','idx1' ,'-append')
save('ryan5_data.mat','idx2' ,'-append')
save('ryan5_data.mat','idx3' ,'-append')
save('ryan5_data.mat','idx12' ,'-append')
save('ryan5_data.mat','idx23' ,'-append')
save('ryan5_data.mat','idx13' ,'-append')
save('ryan5_data.mat','idxAll' ,'-append')
save('ryan5_data.mat','idx123' ,'-append')
save('ryan5_data.mat','key1' ,'-append')
save('ryan5_data.mat','key2' ,'-append')
save('ryan5_data.mat','key3' ,'-append')
save('ryan5_data.mat','input_size' ,'-append')

save('ryan5_data.mat','idxTrain' ,'-append')
save('ryan5_data.mat','idxValid' ,'-append')
```

```

save('ryan5_data.mat','idxTest' ,'-append')
save('ryan5_data.mat','idxTrainRand' ,'-append')
save('ryan5_data.mat','idxValidRand' ,'-append')
save('ryan5_data.mat','idxTestRand' ,'-append')
save('ryan5_data.mat','idxTrainSort' ,'-append')
save('ryan5_data.mat','idxValidSort' ,'-append')
save('ryan5_data.mat','idxTestSort' ,'-append')

```

```

cycle_life_list = zeros();
cycle_life_list = [ 1 1 1 ] ;

```

```

a = ryan5_battery_dataset(idxTest(1)).QDischargeSmooth;
b = ryan5_battery_dataset(idxTest(end)).QDischargeSmooth;
a1 = ryan5_battery_dataset(idxTest(2)).QDischargeSmooth;
b1 = ryan5_battery_dataset(idxTest(end-1)).QDischargeSmooth;

```

```

%https://kr.mathworks.com/matlabcentral/answers/146685-find-index-where-value-exceeds-threshold
[idxFirst] = find(a >= 1,1,'last')

```

```

idxFirst = 73

```

```

[idxLast] = find(b >= 1,1,'last' )

```

```

idxLast = 669

```

```

idxLast = idxLast + 1

```

```

idxLast = 670

```

```

a(idxFirst-1)

```

```

ans = 1.0019

```

```

a(idxFirst)

```

```

ans = 1.0007

```

```

a(idxFirst+1)

```

```

ans = 0.9995

```

```

b(idxLast-1)

```

```

ans = 1.0001

```

```

b(idxLast)

```

```

ans = 1.0000

```

```

b(idxLast+1)

```

```

ans = 0.9998

```



```
[idxFirst] = find(a1 >= 1,1,'last')
```

```
idxFirst = 91
```

```
[idxLast] = find(b1 >= 1,1,'last' )
```

```
idxLast = 462
```

```
idxLast = idxLast + 1
```

```
idxLast = 463
```

```
a1(idxFirst-1)
```

```
ans = 1.0013
```

```
a1(idxFirst)
```

```
ans = 1.0004
```

```
a1(idxFirst+1)
```

```
ans = 0.9995
```

```
b1(idxLast-1)
```

```
ans = 1.0000
```

```
b1(idxLast)
```

```
ans = 0.9998
```

```
b1(idxLast+1)
```

```
ans = 0.9995
```

```
idxTestCount = length(idxTest)
```

```
idxTestCount = 12
```

```
nLines = length(idxTest);  
legend_str = cell(nLines,1);
```

```
figure
```

```
hold on
```

```
for j = 1 : idxTestCount
```

```
    i = idxTest(j);
```

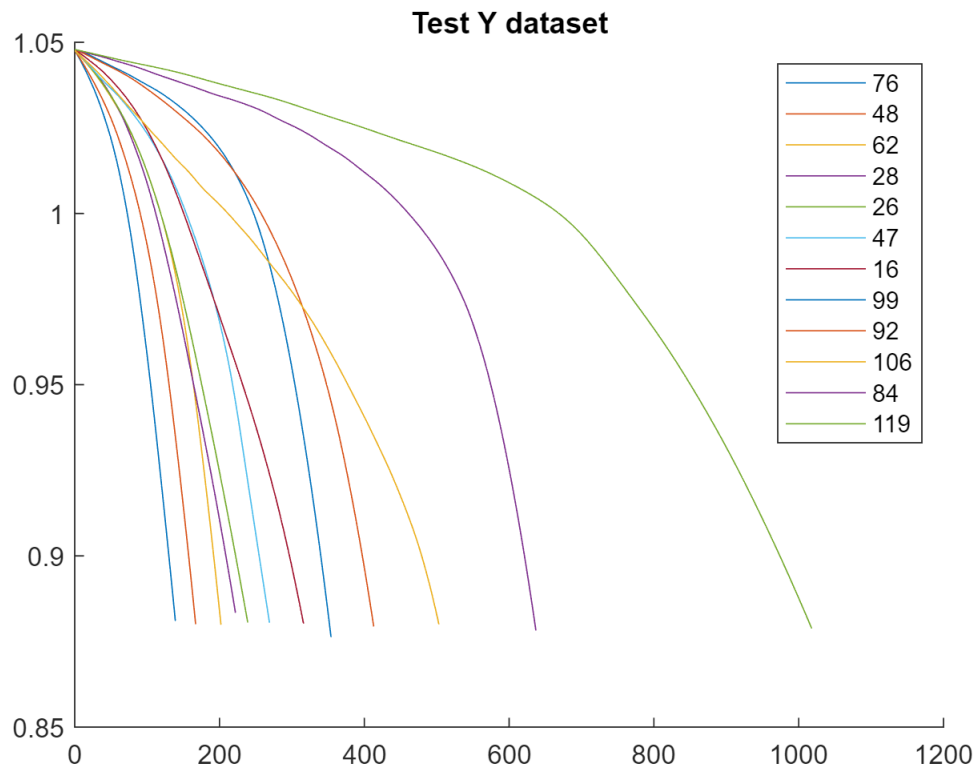
```
    plot(ryan5_battery_dataset(i).cycle,ryan5_battery_dataset(i).QDischargeSmooth)
```

```
    cycle_life_list(j,:) = [j i ryan5_battery_dataset(i).cycle_life];
```

```

legend_str{j} = num2str(i);
end
hold off
title 'Test Y dataset'
legend(legend_str)

```



cycle_life_list

```

cycle_life_list = 12x3
    1    76   140
    2    48   168
    3    62   203
    4    28   222
    5    26   239
    6    47   270
    7    16   316
    8    99   355
    9    92   414
   10   106   504
    ⋮

```

```

battery_testset_org = zeros();
battery_testset_org = ryan5_battery_dataset(idxTest(1));
idxTestCount = length(idxTest)

```

```
idxTestCount = 12
```

```
idxTestStopCount = idxTestCount / 2
```

```
idxTestStopCount = 6
```

```
idxTestReverse = idxTestCount
```

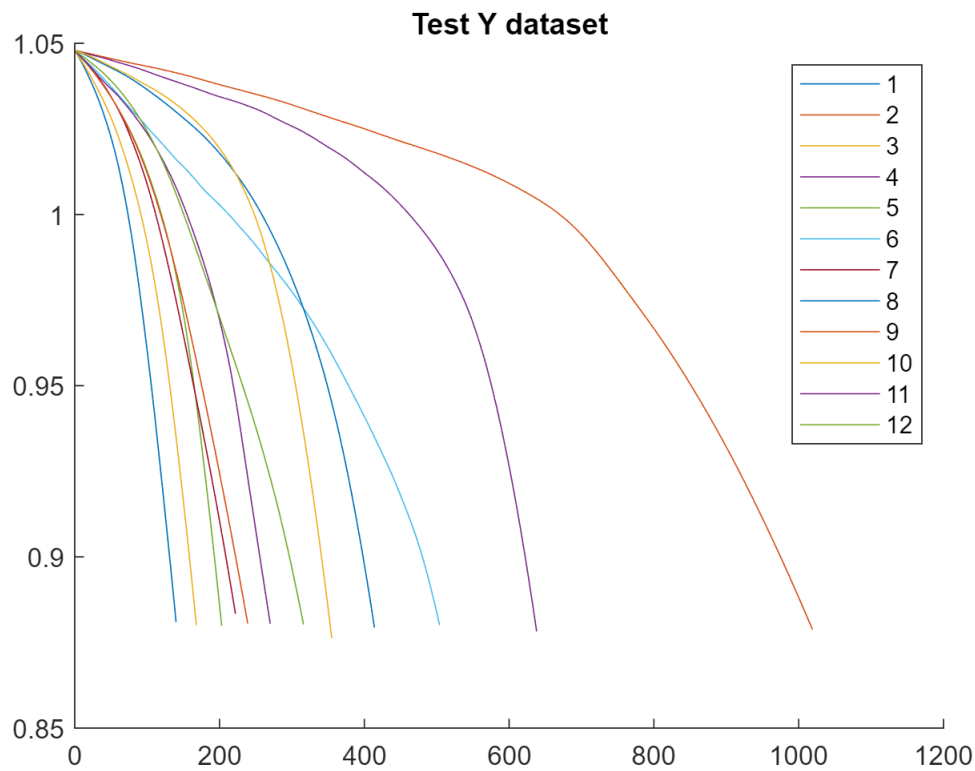
```
idxTestReverse = 12
```

```
k = 1;
kk = idxTestCount;
for j = 1 : idxTestStopCount
    pre = idxTest(j);
    post = idxTest(idxTestReverse);
    idxTestReverse = idxTestReverse - 1;
    battery_testset_org(k,:) = ryan5_battery_dataset(pre);
    k=k+1;
    battery_testset_org(k,:) = ryan5_battery_dataset(post);
    k=k+1;
end

cycle_life_list = zeros();
cycle_life_list = [ 1 1 1 ] ;

nLines = size(battery_testset_org,1);
legend_str = cell(nLines,1);
figure
hold on
for i = 1 : nLines
    %plot(battery_testset(i).cycle,battery_testset(i).QDischargeSmooth)
    plot(battery_testset_org(i).QDischargeSmooth)
    cycle_life_list(i,:) = [i i battery_testset_org(i).cycle_life];
    legend_str{i} = num2str(i);
end

hold off
title 'Test Y dataset'
legend(legend_str)
```



```
cycle_life_list_org = cycle_life_list
```

```
cycle_life_list_org = 12x3
    1      1      140
    2      2     1019
    3      3     168
    4      4     638
    5      5     203
    6      6     504
    7      7     222
    8      8     414
    9      9     239
   10     10     355
    ⋮
```

```
battery_testset = zeros();
battery_testset = ryan5_battery_dataset(idxTest(1));

battery_testset_before = [ ...
    1, ...
    ryan5_battery_dataset(1).cycle_life , ...
    ryan5_battery_dataset(1).policy , ...
    ryan5_battery_dataset(1).policy_readable , ...
    ryan5_battery_dataset(1).key ...
    ];

battery_testset_after = [ ...
```

```

1, ...
ryan5_battery_dataset(1).cycle_life , ...
ryan5_battery_dataset(1).policy , ...
ryan5_battery_dataset(1).policy_readable , ...
ryan5_battery_dataset(1).key ...
];

```

```

%bat_pre = ryan5_battery_dataset(idxTest(1))
%bat_post = ryan5_battery_dataset(idxTest(12))
%battery_testset = bat_pre

```

```

idxTestCount = length(idxTest)

```

```

idxTestCount = 12

```

```

idxTestStopCount = idxTestCount / 2

```

```

idxTestStopCount = 6

```

```

idxTestReverse = idxTestCount

```

```

idxTestReverse = 12

```

```

k = 1;
kk = idxTestCount;
for j = 1 : idxTestStopCount
    pre = idxTest(j);
    post = idxTest(idxTestReverse);
    idxTestReverse = idxTestReverse - 1;

    %ryan5_battery_dataset(pre).QDischargeSmooth
    %cycle_life_list(j,:) = [j pre ryan5_battery_dataset(pre).cycle_life];
    %legend_str{j} = num2str(pre);

    [idxPre] = find(ryan5_battery_dataset(pre).QDischargeSmooth >= 1,1,'last');
    [idxPost] = find(ryan5_battery_dataset(post).QDischargeSmooth >= 1,1,'last' );
    idxPost = idxPost + 1;
    idxPreAfter = idxPre + 1;
    idxPreBefore = idxPre ;
    idxPostBefore = idxPost -1;
    idxPostAfter = idxPost ;

    battery_testset_before(k,:) = [ ...
    pre, ...
    ryan5_battery_dataset(pre).cycle_life , ...
    ryan5_battery_dataset(pre).policy , ...
    ryan5_battery_dataset(pre).policy_readable , ...
    ryan5_battery_dataset(pre).key ...
    ];

    battery_testset_after(k,:) = [ ...

```

```

post, ...
ryan5_battery_dataset(post).cycle_life , ...
ryan5_battery_dataset(post).policy , ...
ryan5_battery_dataset(post).policy_readable , ...
ryan5_battery_dataset(post).key ...
];

battery_testset_before(k+1,:) = [ ...
post, ...
ryan5_battery_dataset(post).cycle_life , ...
ryan5_battery_dataset(post).policy , ...
ryan5_battery_dataset(post).policy_readable , ...
ryan5_battery_dataset(post).key ...
];

battery_testset_after(k+1,:) = [ ...
pre, ...
ryan5_battery_dataset(pre).cycle_life , ...
ryan5_battery_dataset(pre).policy , ...
ryan5_battery_dataset(pre).policy_readable , ...
ryan5_battery_dataset(pre).key ...
];

battery_testset(k,:) = ryan5_battery_dataset(pre);

battery_testset(k).IR([idxPreAfter:end],:) = [] ;
battery_testset(k).QCharge([idxPreAfter:end],:) = [] ;
battery_testset(k).QDischarge([idxPreAfter:end],:) = [] ;
battery_testset(k).SOH_c([idxPreAfter:end],:) = [] ;
battery_testset(k).SOH_d([idxPreAfter:end],:) = [] ;
battery_testset(k).RUL([idxPreAfter:end],:) = [] ;
battery_testset(k).Time_c([idxPreAfter:end],:) = [] ;
battery_testset(k).Time_d([idxPreAfter:end],:) = [] ;
battery_testset(k).Time_d1([idxPreAfter:end],:) = [] ;
battery_testset(k).Tavg([idxPreAfter:end],:) = [] ;
battery_testset(k).Tmin([idxPreAfter:end],:) = [] ;
battery_testset(k).Tmax([idxPreAfter:end],:) = [] ;

battery_testset(k).Vc([idxPreAfter:end],:) = [] ;
battery_testset(k).Vd([idxPreAfter:end],:) = [] ;
battery_testset(k).Tc([idxPreAfter:end],:) = [] ;
battery_testset(k).Td([idxPreAfter:end],:) = [] ;
battery_testset(k).Ic([idxPreAfter:end],:) = [] ;
battery_testset(k).Id([idxPreAfter:end],:) = [] ;

battery_testset(k).discharge_dQdV([idxPreAfter:end],:) = [] ;
battery_testset(k).Qdlin([idxPreAfter:end],:) = [] ;
battery_testset(k).Tdlin([idxPreAfter:end],:) = [] ;

```

```

battery_testset(k).chargetime([idxPreAfter:end],:) = [] ;
battery_testset(k).QDischargeSmooth([idxPreAfter:end],:) = [] ;

battery_testset(k).IR = vertcat(battery_testset(k).IR,ryan5_battery_dataset(post).IR([idxPreAfter:end],:));
battery_testset(k).QCharge = vertcat(battery_testset(k).QCharge,ryan5_battery_dataset(post).QCharge([idxPreAfter:end],:));
battery_testset(k).QDischarge = vertcat(battery_testset(k).QDischarge,ryan5_battery_dataset(post).QDischarge([idxPreAfter:end],:));
battery_testset(k).SOH_c = vertcat(battery_testset(k).SOH_c,ryan5_battery_dataset(post).SOH_c([idxPreAfter:end],:));
battery_testset(k).SOH_d = vertcat(battery_testset(k).SOH_d,ryan5_battery_dataset(post).SOH_d([idxPreAfter:end],:));
battery_testset(k).RUL = vertcat(battery_testset(k).RUL,ryan5_battery_dataset(post).RUL([idxPreAfter:end],:));
battery_testset(k).Time_c = vertcat(battery_testset(k).Time_c,ryan5_battery_dataset(post).Time_c([idxPreAfter:end],:));
battery_testset(k).Time_d = vertcat(battery_testset(k).Time_d,ryan5_battery_dataset(post).Time_d([idxPreAfter:end],:));
battery_testset(k).Time_d1 = vertcat(battery_testset(k).Time_d1,ryan5_battery_dataset(post).Time_d1([idxPreAfter:end],:));
battery_testset(k).Tavg = vertcat(battery_testset(k).Tavg,ryan5_battery_dataset(post).Tavg([idxPreAfter:end],:));
battery_testset(k).Tmin = vertcat(battery_testset(k).Tmin,ryan5_battery_dataset(post).Tmin([idxPreAfter:end],:));
battery_testset(k).Tmax = vertcat(battery_testset(k).Tmax,ryan5_battery_dataset(post).Tmax([idxPreAfter:end],:));
battery_testset(k).Vc = vertcat(battery_testset(k).Vc,ryan5_battery_dataset(post).Vc([idxPreAfter:end],:));
battery_testset(k).Vd = vertcat(battery_testset(k).Vd,ryan5_battery_dataset(post).Vd([idxPreAfter:end],:));
battery_testset(k).Tc = vertcat(battery_testset(k).Tc,ryan5_battery_dataset(post).Tc([idxPreAfter:end],:));
battery_testset(k).Td = vertcat(battery_testset(k).Td,ryan5_battery_dataset(post).Td([idxPreAfter:end],:));
battery_testset(k).Ic = vertcat(battery_testset(k).Ic,ryan5_battery_dataset(post).Ic([idxPreAfter:end],:));
battery_testset(k).Id = vertcat(battery_testset(k).Id,ryan5_battery_dataset(post).Id([idxPreAfter:end],:));

battery_testset(k).discharge_dQdV = vertcat(battery_testset(k).discharge_dQdV,ryan5_battery_dataset(post).discharge_dQdV([idxPreAfter:end],:));
battery_testset(k).Qdlin = vertcat(battery_testset(k).Qdlin,ryan5_battery_dataset(post).Qdlin([idxPreAfter:end],:));
battery_testset(k).Tdlin = vertcat(battery_testset(k).Tdlin,ryan5_battery_dataset(post).Tdlin([idxPreAfter:end],:));

battery_testset(k).chargetime = vertcat(battery_testset(k).chargetime,ryan5_battery_dataset(post).chargetime([idxPreAfter:end],:));
battery_testset(k).QDischargeSmooth = vertcat(battery_testset(k).QDischargeSmooth,ryan5_battery_dataset(post).QDischargeSmooth([idxPreAfter:end],:));

battery_testset(k).cycle_life = idxPre + (ryan5_battery_dataset(post).cycle_life - idxPost);
battery_testset(k).cycle = [];
battery_testset(k).cycle = linspace(1,battery_testset(k).cycle_life,battery_testset(k).cycle_life);

battery_testset(k).QDischargePolyMdl = [];
% battery_testset(k).QDischargePolyMdl = polyfit(battery_testset(k).cycle, battery_testset(k).QDischarge,battery_testset(k).cycle_life);
battery_testset(k).QDischargePolyfit = [];
% battery_testset(k).QDischargePolyfit = polyval(battery_testset(k).QDischargePolyMdl,battery_testset(k).cycle);

%%%%%%%%%%%%%%
k = k+1;
battery_testset(k,:) = ryan5_battery_dataset(post);

battery_testset(k).IR([idxPostAfter:end],:) = [] ;
battery_testset(k).QCharge([idxPostAfter:end],:) = [] ;
battery_testset(k).QDischarge([idxPostAfter:end],:) = [] ;
battery_testset(k).SOH_c([idxPostAfter:end],:) = [] ;
battery_testset(k).SOH_d([idxPostAfter:end],:) = [] ;
battery_testset(k).RUL([idxPostAfter:end],:) = [] ;
battery_testset(k).Time_c([idxPostAfter:end],:) = [] ;
battery_testset(k).Time_d([idxPostAfter:end],:) = [] ;

```

```

battery_testset(k).Time_d1([idxPostAfter:end],:) = [] ;
battery_testset(k).Tavg([idxPostAfter:end],:) = [] ;
battery_testset(k).Tavg([idxPostAfter:end],:) = [] ;
battery_testset(k).Tmin([idxPostAfter:end],:) = [] ;
battery_testset(k).Tmax([idxPostAfter:end],:) = [] ;
battery_testset(k).Vc([idxPostAfter:end],:) = [] ;
battery_testset(k).Vd([idxPostAfter:end],:) = [] ;
battery_testset(k).Tc([idxPostAfter:end],:) = [] ;
battery_testset(k).Td([idxPostAfter:end],:) = [] ;
battery_testset(k).Ic([idxPostAfter:end],:) = [] ;
battery_testset(k).Id([idxPostAfter:end],:) = [] ;

battery_testset(k).discharge_dQdV([idxPostAfter:end],:) = [] ;
battery_testset(k).Qdlin([idxPostAfter:end],:) = [] ;
battery_testset(k).Tdlin([idxPostAfter:end],:) = [] ;

battery_testset(k).chargetime([idxPostAfter:end],:) = [] ;
battery_testset(k).QDischargeSmooth([idxPostAfter:end],:) = [] ;

battery_testset(k).IR = vertcat(battery_testset(k).IR,ryan5_battery_dataset(pre).IR([idxPreAfter:end],:));
battery_testset(k).QCharge = vertcat(battery_testset(k).QCharge,ryan5_battery_dataset(pre).QCharge([idxPreAfter:end],:));
battery_testset(k).QDischarge = vertcat(battery_testset(k).QDischarge,ryan5_battery_dataset(pre).QDischarge([idxPreAfter:end],:));
battery_testset(k).SOH_c = vertcat(battery_testset(k).SOH_c,ryan5_battery_dataset(pre).SOH_c([idxPreAfter:end],:));
battery_testset(k).SOH_d = vertcat(battery_testset(k).SOH_d,ryan5_battery_dataset(pre).SOH_d([idxPreAfter:end],:));
battery_testset(k).RUL = vertcat(battery_testset(k).RUL,ryan5_battery_dataset(pre).RUL([idxPreAfter:end],:));
battery_testset(k).Time_c = vertcat(battery_testset(k).Time_c,ryan5_battery_dataset(pre).Time_c([idxPreAfter:end],:));
battery_testset(k).Time_d = vertcat(battery_testset(k).Time_d,ryan5_battery_dataset(pre).Time_d([idxPreAfter:end],:));
battery_testset(k).Time_d1 = vertcat(battery_testset(k).Time_d1,ryan5_battery_dataset(pre).Time_d1([idxPreAfter:end],:));
battery_testset(k).Tavg = vertcat(battery_testset(k).Tavg,ryan5_battery_dataset(pre).Tavg([idxPreAfter:end],:));
battery_testset(k).Tmin = vertcat(battery_testset(k).Tmin,ryan5_battery_dataset(pre).Tmin([idxPreAfter:end],:));
battery_testset(k).Tmax = vertcat(battery_testset(k).Tmax,ryan5_battery_dataset(pre).Tmax([idxPreAfter:end],:));
battery_testset(k).Vc = vertcat(battery_testset(k).Vc,ryan5_battery_dataset(pre).Vc([idxPreAfter:end],:));
battery_testset(k).Vd = vertcat(battery_testset(k).Vd,ryan5_battery_dataset(pre).Vd([idxPreAfter:end],:));
battery_testset(k).Tc = vertcat(battery_testset(k).Tc,ryan5_battery_dataset(pre).Tc([idxPreAfter:end],:));
battery_testset(k).Td = vertcat(battery_testset(k).Td,ryan5_battery_dataset(pre).Td([idxPreAfter:end],:));
battery_testset(k).Ic = vertcat(battery_testset(k).Ic,ryan5_battery_dataset(pre).Ic([idxPreAfter:end],:));
battery_testset(k).Id = vertcat(battery_testset(k).Id,ryan5_battery_dataset(pre).Id([idxPreAfter:end],:));

battery_testset(k).discharge_dQdV = vertcat(battery_testset(k).discharge_dQdV,ryan5_battery_dataset(pre).discharge_dQdV([idxPreAfter:end],:));
battery_testset(k).Qdlin = vertcat(battery_testset(k).Qdlin,ryan5_battery_dataset(pre).Qdlin([idxPreAfter:end],:));
battery_testset(k).Tdlin = vertcat(battery_testset(k).Tdlin,ryan5_battery_dataset(pre).Tdlin([idxPreAfter:end],:));

battery_testset(k).chargetime = vertcat(battery_testset(k).chargetime,ryan5_battery_dataset(pre).chargetime([idxPreAfter:end],:));
battery_testset(k).QDischargeSmooth = vertcat(battery_testset(k).QDischargeSmooth,ryan5_battery_dataset(pre).QDischargeSmooth([idxPreAfter:end],:));

battery_testset(k).cycle_life = idxPost + (ryan5_battery_dataset(pre).cycle_life - idxPreAfter);
battery_testset(k).cycle = [];
battery_testset(k).cycle = linspace(1,battery_testset(k).cycle_life,battery_testset(k).cycle_life);

battery_testset(k).QDischargePolyMd1 = [];

```



```

%    battery_testset(k).QDischargePolyMdl = polyfit(battery_testset(k).cycle, battery_testset(k).QDischargeSmooth, 1);
%    battery_testset(k).QDischargePolyfit = [];
%    battery_testset(k).QDischargePolyfit = polyval(battery_testset(k).QDischargePolyMdl, battery_testset(k).cycle);
    k = k + 1;
end
%battery_testset

```

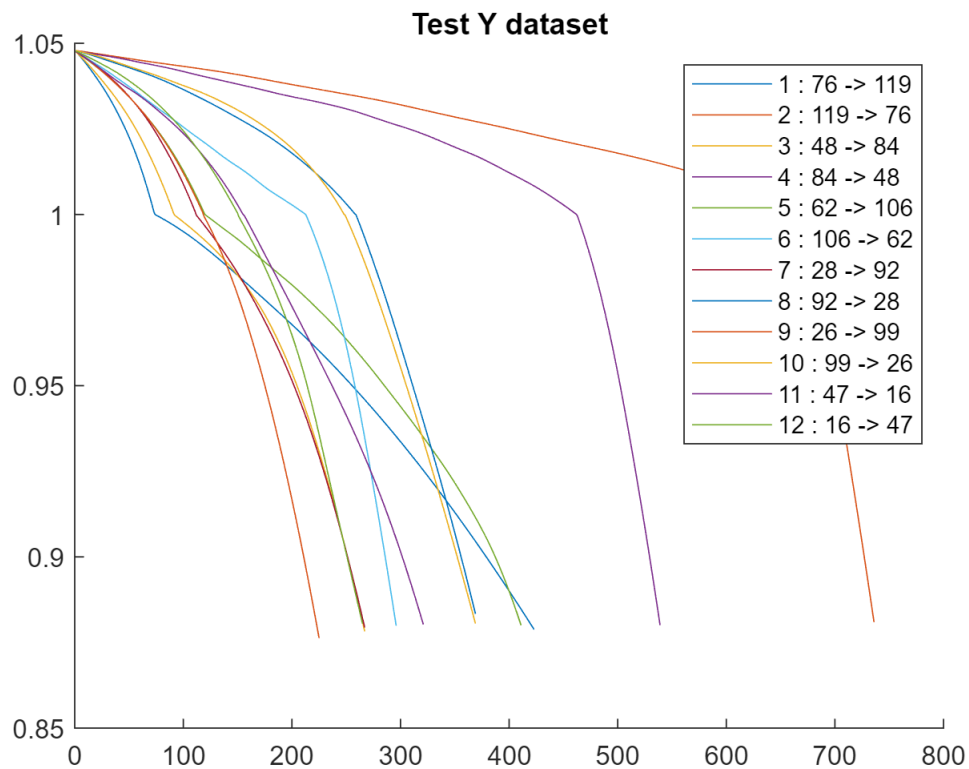
```

cycle_life_list = zeros();
cycle_life_list = [ 1 1 1 ] ;

nLines = size(battery_testset,1);
legend_str = cell(nLines,1);
figure
hold on
for i = 1 : nLines
    %plot(battery_testset(i).cycle,battery_testset(i).QDischargeSmooth)
    plot(battery_testset(i).cycle,battery_testset(i).QDischargeSmooth)
    cycle_life_list(i,:) = [i i battery_testset(i).cycle_life];
    legend_str{i} = num2str(i) + " : " + num2str(battery_testset(i).cycle_life) + " -> " + num2str(battery_testset(i).QDischargeSmooth);
end

hold off
title 'Test Y dataset'
legend(legend_str)

```



```
cycle_life_list_change = cycle_life_list
```

```
cycle_life_list_change = 12x3
```

```

1      1      423
2      2      738
3      3      267
4      4      541
5      5      411
6      6      298
7      7      267
8      8      371
9      9      225
10     10      371
⋮

```

```

%{
cycle_life_list = zeros();
cycle_life_list = [ 1 1 1 ] ;

nLines = length(idxTestSort);
legend_str = cell(nLines,1);
figure
hold on
for j = 1 : length(idxTestSort)
    i = idxTestSort(j);
    plot(ryan5_battery_dataset(i).cycle,ryan5_battery_dataset(i).QDischargeSmooth)

```

```

    cycle_life_list(j,:) = [j i ryan5_battery_dataset(i).cycle_life];
    legend_str{j} = num2str(i);
end

hold off
title 'Test Y dataset'
legend(legend_str)
cycle_life_list
%}

```

```

save('ryan5_battery_dataset.mat','battery_testset_before' ,'-append')
save('ryan5_battery_dataset.mat','battery_testset_after' ,'-append')
save('ryan5_battery_dataset.mat','battery_testset' ,'-append')
save('ryan5_battery_dataset.mat','battery_testset_org' ,'-append')

```

```

%% load ryan5_gen_data.mat

%{
load('ryan5_battery_dataset.mat','ryan5_battery_dataset' )
load('ryan5_data.mat','idxAll' )

%}

```

첫번째 cycle의 충전전류를 얻는다.

```

idx = 1; % idxAll;

ryan5_battery_dataset(idx).Ic

```

```

ans = 428x100
    4.4001    4.3999    4.4005    4.4008    4.3997    4.3999    4.4003    4.4000 ...
    4.4001    4.3999    4.4002    4.3998    4.4002    4.3995    4.4002    4.4000
    4.4006    4.3998    4.4001    4.4000    4.4002    4.3994    4.4001    4.4000
    4.4003    4.3998    4.3999    4.3997    4.3999    4.3999    4.4002    4.4001
    4.3997    4.3998    4.4000    4.3998    4.3998    4.4002    4.4005    4.4000
    4.4001    4.4002    4.3996    4.3993    4.4000    4.3999    4.3999    4.4000
    4.3998    4.3991    4.4000    4.4006    4.3999    4.3999    4.4001    4.4002
    4.4001    4.4002    4.4003    4.3998    4.4001    4.4000    4.4001    4.4000
    4.4004    4.4000    4.3999    4.3998    4.3999    4.3997    4.4000    4.3999
    4.4002    4.4000    4.4000    4.4002    4.3999    4.3999    4.3998    4.4000
    ⋮

```

```

ryan5_battery_dataset(idx).Ic(1,:)

```

```

ans = 1x100
    4.4001    4.3999    4.4005    4.4008    4.3997    4.3999    4.4003    4.4000 ...

```

```

% 열벡터로 바꾼다.
Ic1 = ryan5_battery_dataset(idx).Ic(1,:)

```

```

Ic1 = 100x1
    4.4001

```

```

4.3999
4.4005
4.4008
4.3997
4.3999
4.4003
4.4000
4.3999
4.4007
⋮

```

```

% 전체 데이터에 대해서
IcFirstFull = zeros()

```

```

IcFirstFull = 0

```

```

IcFirstFull = ryan5_battery_dataset(1).Ic(1,:)

```

```

IcFirstFull = 1×100
    4.4001    4.3999    4.4005    4.4008    4.3997    4.3999    4.4003    4.4000 ...

```

```

VdFirstFull = zeros()

```

```

VdFirstFull = 0

```

```

VdFirstFull = ryan5_battery_dataset(1).Vd(1,:)

```

```

VdFirstFull = 1×100
    3.2974    3.2346    3.2033    3.1864    3.1762    3.1695    3.1647    3.1611 ...

```

```

%IcFirstFull = ryan5_battery_dataset(2).Ic(1,:)
for i=1:length(ryan5_battery_dataset)
    IcFirstFull(i,:) = ryan5_battery_dataset(i).Ic(1,:);
    VdFirstFull(i,:) = ryan5_battery_dataset(i).Vd(1,:);
end
IcFirstFull

```

```

IcFirstFull = 119×100
    4.4001    4.3999    4.4005    4.4008    4.3997    4.3999    4.4003    4.4000 ...
    4.8000    4.8000    4.8000    4.7999    4.8001    4.8000    4.8001    4.8001
    4.8000    4.8001    4.7998    4.8001    4.8001    4.7999    4.7999    4.8001
    5.4007    5.3993    5.4001    5.4003    5.3996    5.3999    5.3990    5.3997
    5.3984    5.3995    5.3998    5.3989    5.3991    5.4003    5.3996    5.4001
    5.4014    5.4002    5.4005    5.3991    5.3997    5.3996    5.4012    5.4003
    5.3968    5.3989    5.3993    5.3999    5.3989    5.4010    5.4007    5.3997
    5.3995    5.4008    5.4003    5.3991    5.4000    5.4002    5.3999    5.4005
    5.4024    5.3999    5.3998    5.3994    5.4004    5.3996    5.3994    5.4007
    5.4000    5.4002    5.3996    5.3999    5.3997    5.4001    5.3990    5.3994
    ⋮

```

```

VdFirstFull

```

```

VdFirstFull = 119×100
    3.2974    3.2346    3.2033    3.1864    3.1762    3.1695    3.1647    3.1611 ...

```

3.2991	3.2361	3.2046	3.1877	3.1774	3.1708	3.1659	3.1620
3.3029	3.2397	3.2078	3.1907	3.1801	3.1732	3.1679	3.1640
3.3038	3.2409	3.2087	3.1914	3.1811	3.1744	3.1695	3.1658
3.3053	3.2440	3.2146	3.1992	3.1902	3.1844	3.1801	3.1765
3.2988	3.2360	3.2043	3.1875	3.1775	3.1709	3.1661	3.1620
3.3052	3.2434	3.2133	3.1974	3.1883	3.1822	3.1778	3.1743
3.3044	3.2416	3.2094	3.1923	3.1823	3.1754	3.1707	3.1668
3.2953	3.2325	3.2010	3.1839	3.1736	3.1669	3.1620	3.1580
3.2974	3.2350	3.2041	3.1876	3.1777	3.1708	3.1659	3.1620
⋮							

```
%IcFirstFull = IcFirstFull'
```

```
RulFirst = [ryan5_battery_dataset(idxAll).cycle_life ]'
```

```
RulFirst = 119x1
```

```
428
229
253
338
416
331
278
235
284
462
⋮
```

```
blockSize = [1, 5] % 다섯개를 묶어서.
```

```
blockSize = 1x2
1      5
```

```
meanFilterFunction = @(theBlockStructure) mean2(theBlockStructure.data(:))
```

```
meanFilterFunction = function_handle with value:
@(theBlockStructure)mean2(theBlockStructure.data(:))
```

```
IcFirst = blockproc(IcFirstFull, blockSize, meanFilterFunction)
```

```
IcFirst = 119x20
4.4002    4.4002    4.4000    4.4002    4.4000    4.3999    4.3999    4.4000 ...
4.8000    4.8001    4.8000    4.7999    4.8000    4.8000    4.8000    4.8001
4.8000    4.8000    4.7999    4.8000    4.8000    4.8000    4.8000    4.8000
5.4000    5.3999    5.3998    5.3999    5.4002    5.4000    5.3997    4.3203
5.3991    5.4004    5.3996    5.4003    5.3998    5.4004    5.4000    5.4001
5.4002    5.4002    5.3996    5.3997    5.4000    5.3997    5.3999    5.4000
5.3987    5.4003    5.4001    5.3997    5.3999    5.4001    5.3999    5.4007
5.3999    5.3998    5.3997    5.4001    5.4002    5.3997    5.3998    5.4002
5.4004    5.4000    5.4002    5.4000    5.3999    5.3991    5.4003    5.4003
5.3999    5.3996    5.4003    5.3997    5.3997    5.4004    5.4003    5.4001
⋮
```

```
VdFirst = blockproc(VdFirstFull, blockSize, meanFilterFunction)
```

```
VdFirst = 119x20
3.2196    3.1616    3.1480    3.1376    3.1274    3.1164    3.1044    3.0915 ...
3.2210    3.1627    3.1486    3.1379    3.1272    3.1157    3.1032    3.0898
3.2243    3.1647    3.1502    3.1395    3.1291    3.1180    3.1061    3.0930
3.2252    3.1664    3.1529    3.1429    3.1330    3.1224    3.1109    3.0982
```

3.2307	3.1771	3.1643	3.1540	3.1438	3.1328	3.1208	3.1078
3.2208	3.1628	3.1492	3.1387	3.1281	3.1168	3.1045	3.0913
3.2295	3.1749	3.1620	3.1517	3.1415	3.1306	3.1188	3.1061
3.2260	3.1674	3.1535	3.1431	3.1326	3.1215	3.1094	3.0964
3.2173	3.1588	3.1451	3.1346	3.1242	3.1130	3.1008	3.0875
3.2203	3.1626	3.1482	3.1370	3.1258	3.1138	3.1007	3.0868
:							
:							

```
%RulFirst = cycleLife;
%clear cycleLife
```

```
%{
load("ryan5_data.mat",'RulFirst')
load("ryan5_data.mat",'IcFirstFull')
load("ryan5_data.mat",'IcFirst')
load("ryan5_data.mat",'VdFirstFull')
load("ryan5_data.mat",'VdFirst')

load("ryan5_data.mat",'idxA;')
load("ryan5_data.mat",'idxTrain')
load("ryan5_data.mat",'idxValid')
load("ryan5_data.mat",'idxTest')

%}

[ynorm_IcFirst_RUL,ymax_IcFirst_RUL,ymin_IcFirst_RUL,yrate_IcFirst_RUL,xnorm_IcFirstFull_RUL,xnorm_IcFirst_RUL] =
minmax_norm_new(RulFirst,IcFirstFull);
[ynorm_IcFirst_RUL1,xnorm_IcFirst_RUL] = minmax_norm_new_with_param(RulFirst,IcFirst,ymin_IcFirst_RUL,ymax_IcFirst_RUL);
% https://stackoverflow.com/questions/40053450/compare-if-two-vectors-are-the-same
all(ynorm_IcFirst_RUL == ynorm_IcFirst_RUL1)
```

```
ans = logical
     1
```

```
[ynorm_VdFirst_RUL,ymax_VdFirst_RUL,ymin_VdFirst_RUL,yrate_VdFirst_RUL,xnorm_VdFirstFull_RUL,xnorm_VdFirst_RUL] =
minmax_norm_new(RulFirst,VdFirstFull);
[ynorm_VdFirst_RUL1,xnorm_VdFirst_RUL] = minmax_norm_new_with_param(RulFirst,VdFirst,ymin_VdFirst_RUL,ymax_VdFirst_RUL);
% https://stackoverflow.com/questions/40053450/compare-if-two-vectors-are-the-same
all(ynorm_VdFirst_RUL == ynorm_VdFirst_RUL1)
```

```
ans = logical
     1
```

```
all(ynorm_IcFirst_RUL == ynorm_VdFirst_RUL)
```

```
ans = logical
     1
```

```
all(ymax_IcFirst_RUL == ymax_VdFirst_RUL)
```

```
ans = logical
      1
```

```
all(ymin_IcFirst_RUL == ymin_VdFirst_RUL)
```

```
ans = logical
      1
```

```
all(yrate_IcFirst_RUL == yrate_VdFirst_RUL)
```

```
ans = logical
      1
```

```
ynorm_First_RUL = ynorm_IcFirst_RUL
```

```
ynorm_First_RUL = 119x1
    0.3004
    0.1130
    0.1356
    0.2156
    0.2891
    0.2090
    0.1591
    0.1186
    0.1648
    0.3324
    ⋮
```

```
ymax_First_RUL = ymax_IcFirst_RUL
```

```
ymax_First_RUL = 1171
```

```
ymin_First_RUL = ymin_IcFirst_RUL
```

```
ymin_First_RUL = 109
```

```
yrate_First_RUL = yrate_IcFirst_RUL
```

```
yrate_First_RUL = 1062
```

```
x_trainValid_IcFirstFull_RUL = xnorm_IcFirstFull_RUL([idxTrain idxValid],:);
x_test_IcFirstFull_RUL = xnorm_IcFirstFull_RUL([idxTest],:);
x_trainValid_IcFirst_RUL = xnorm_IcFirst_RUL([idxTrain idxValid],:);
x_test_IcFirst_RUL = xnorm_IcFirst_RUL([idxTest],:);
```

```
y_trainValid_IcFirst_RUL = ynorm_IcFirst_RUL([idxTrain idxValid]);
y_test_IcFirst_RUL = ynorm_IcFirst_RUL([idxTest]);
Y_trainValid_IcFirst_RUL = RulFirst([idxTrain idxValid]);
Y_test_IcFirst_RUL = RulFirst([idxTest]);
```

```
x_trainValid_VdFirstFull_RUL = xnorm_VdFirstFull_RUL([idxTrain idxValid],:);
x_test_VdFirstFull_RUL = xnorm_VdFirstFull_RUL([idxTest],:);
x_trainValid_VdFirst_RUL = xnorm_VdFirst_RUL([idxTrain idxValid],:);
```

```

x_test_VdFirst_RUL = xnorm_VdFirst_RUL([idxTest],:);

y_trainValid_VdFirst_RUL = ynorm_VdFirst_RUL([idxTrain idxValid]);
y_test_VdFirst_RUL = ynorm_VdFirst_RUL([idxTest]);
Y_trainValid_VdFirst_RUL = RulFirst([idxTrain idxValid]);
Y_test_VdFirst_RUL = RulFirst([idxTest]);

all(y_trainValid_IcFirst_RUL == y_trainValid_VdFirst_RUL)

```

```

ans = logical
     1

```

```

all(y_test_IcFirst_RUL == y_test_VdFirst_RUL)

```

```

ans = logical
     1

```

```

all(Y_trainValid_IcFirst_RUL == Y_trainValid_VdFirst_RUL)

```

```

ans = logical
     1

```

```

all(Y_test_IcFirst_RUL == Y_test_VdFirst_RUL)

```

```

ans = logical
     1

```

```

y_trainValid_First_RUL = y_trainValid_IcFirst_RUL ;
y_test_First_RUL = y_test_IcFirst_RUL ;
Y_trainValid_First_RUL = Y_trainValid_IcFirst_RUL ;
Y_test_First_RUL = Y_test_IcFirst_RUL ;

```

변수 추출

```

Cap = zeros();
Cap_c = zeros();
%SOH = zeros();
% = zeeros();
IcFull = zeros();
IdFull = zeros();
VcFull = zeros();
VdFull = zeros();
TcFull = zeros();
TdFull = zeros();

%VdFull = [1:100];
ryan5_battery_dataset(1).QDischargeSmooth

```

```

ans = 428x1
     1.0479
     1.0478

```



```

1.0477
1.0476
1.0475
1.0474
1.0473
1.0472
1.0471
1.0470
:

```

```
ryan5_battery_dataset(1).Vd
```

```

ans = 428x100
    3.2974    3.2346    3.2033    3.1864    3.1762    3.1695    3.1647    3.1611 ...
    3.2967    3.2340    3.2026    3.1858    3.1756    3.1692    3.1643    3.1604
    3.2978    3.2347    3.2036    3.1866    3.1766    3.1700    3.1651    3.1613
    3.2971    3.2345    3.2032    3.1861    3.1760    3.1693    3.1645    3.1607
    3.2972    3.2343    3.2028    3.1859    3.1759    3.1692    3.1643    3.1606
    3.2968    3.2340    3.2026    3.1857    3.1756    3.1689    3.1641    3.1603
    3.2971    3.2344    3.2032    3.1862    3.1760    3.1694    3.1645    3.1606
    3.2970    3.2341    3.2029    3.1857    3.1757    3.1690    3.1642    3.1603
    3.2972    3.2343    3.2030    3.1860    3.1758    3.1691    3.1643    3.1605
    3.2971    3.2343    3.2031    3.1861    3.1760    3.1692    3.1645    3.1605
    :

```

```
size(ryan5_battery_dataset(1).QDischargeSmooth)
```

```

ans = 1x2
    428     1

```

```
length(ryan5_battery_dataset(1).QDischargeSmooth)
```

```
ans = 428
```

```

idxAll = (1:length(ryan5_battery_dataset));
idxTrainValid = [ idxTrain idxValid ];
idxTrainAll = []; %zeros();
idxValidAll = []; %zeros();
idxTestAll = []; %zeros();
idxAllAll = []; %zeros();
idxTrainAll_batIndex = []; %zeros();
idxValidAll_batIndex = []; %zeros();
idxTestAll_batIndex = []; %zeros();
idxAllAll_batIndex = []; %zeros();
%batIndex_temp = [];
j =1;
startIndx = j ;
GenerateFrameList = @(A,S,N) A+S*(0:N-1);

for i = 1: length(idxAll)

    cap_temp = ryan5_battery_dataset(i).QDischargeSmooth;
    cap_c_temp = ryan5_battery_dataset(i).QDischargeSmooth;

```

```

%SOH_temp = max(cap_temp);
RUL_temp = linspace(ryan5_battery_dataset(i).cycle_life,1,ryan5_battery_dataset(i).cycle_1);
leng = length(cap_temp);

%startIndx = j ;
% https://kr.mathworks.com/matlabcentral/answers/478553-how-can-i-have-a-starting-number-a
append_index = GenerateFrameList(startIndx,1,leng);

Vd_temp = ryan5_battery_dataset(i).Vd ;
Vc_temp = ryan5_battery_dataset(i).Vc ;
Td_temp = ryan5_battery_dataset(i).Td ;
Tc_temp = ryan5_battery_dataset(i).Tc ;
Id_temp = ryan5_battery_dataset(i).Id ;
Ic_temp = ryan5_battery_dataset(i).Ic ;
batIndex_temp = [];
batIndex_temp(1:leng, 1) = i;
%VdFull = [VdFull ; Vd_temp];
if i == 1

    Cap = cap_temp;
    Cap_c = cap_c_temp;
    RUL = RUL_temp;

    VdFull = Vd_temp ;
    VcFull = Vc_temp ;
    TdFull = Td_temp ;
    TcFull = Tc_temp ;
    IdFull = Id_temp ;
    IcFull = Ic_temp ;

else
    Cap = vertcat(Cap, cap_temp);
    Cap_c = vertcat(Cap_c, cap_c_temp);
    RUL = vertcat(RUL, RUL_temp);

    VdFull = vertcat(VdFull , Vd_temp );
    VcFull = vertcat(VcFull , Vc_temp );
    TdFull = vertcat(TdFull , Td_temp );
    TcFull = vertcat(TcFull , Tc_temp );
    IdFull = vertcat(IdFull , Id_temp );
    IcFull = vertcat(IcFull , Ic_temp );
end

idxAllAll = [idxAllAll append_index ] ;
idxAllAll_batIndex = [idxAllAll_batIndex batIndex_temp'];

if any(idxTrain(:) == i)
    % https://kr.mathworks.com/matlabcentral/answers/283821-add-single-element-to-array-or-v
    idxTrainAll = [idxTrainAll append_index ] ;

```

```

        idxTrainAll_batIndex = [idxTrainAll_batIndex batIndex_temp'];
    end
    if any(idxValid(:) == i)
        idxValidAll = [idxValidAll append_index ] ;
        idxValidAll_batIndex = [idxValidAll_batIndex batIndex_temp'];
    end
    if any(idxTest(:) == i)
        idxTestAll = [idxTestAll append_index ] ;
        idxTestAll_batIndex = [idxTestAll_batIndex batIndex_temp'];
    end
    %VdFull(i) = Vd_temp;

    startIndx = startIndx + leng;
end

```

idxTrainAll

```

idxTrainAll = 1×33964
    1     2     3     4     5     6     7     8     9    10    11    12    13 ...

```

idxTrainAll_batIndex

```

idxTrainAll_batIndex = 1×33964
    1     1     1     1     1     1     1     1     1     1     1     1     1 ...

```

idxValidAll

```

idxValidAll = 1×4403
    2793    2794    2795    2796    2797    2798 ...

```

idxValidAll_batIndex

```

idxValidAll_batIndex = 1×4403
    10    10    10    10    10    10    10    10    10    10    10    10 ...

```

idxTestAll

```

idxTestAll = 1×4488
    4962    4963    4964    4965    4966    4967 ...

```

idxTestAll_batIndex

```

idxTestAll_batIndex = 1×4488
    16    16    16    16    16    16    16    16    16    16    16    16 ...

```

Cap

```

Cap = 42855×1
    1.0479
    1.0478
    1.0477
    1.0476
    1.0475
    1.0474

```

```
1.0473
1.0472
1.0471
1.0470
⋮
```

VdFull

VdFull = 42855×100

```
3.2974    3.2346    3.2033    3.1864    3.1762    3.1695    3.1647    3.1611 ...
3.2967    3.2340    3.2026    3.1858    3.1756    3.1692    3.1643    3.1604
3.2978    3.2347    3.2036    3.1866    3.1766    3.1700    3.1651    3.1613
3.2971    3.2345    3.2032    3.1861    3.1760    3.1693    3.1645    3.1607
3.2972    3.2343    3.2028    3.1859    3.1759    3.1692    3.1643    3.1606
3.2968    3.2340    3.2026    3.1857    3.1756    3.1689    3.1641    3.1603
3.2971    3.2344    3.2032    3.1862    3.1760    3.1694    3.1645    3.1606
3.2970    3.2341    3.2029    3.1857    3.1757    3.1690    3.1642    3.1603
3.2972    3.2343    3.2030    3.1860    3.1758    3.1691    3.1643    3.1605
3.2971    3.2343    3.2031    3.1861    3.1760    3.1692    3.1645    3.1605
⋮
```

IcFull

IcFull = 42855×100

```
4.4001    4.3999    4.4005    4.4008    4.3997    4.3999    4.4003    4.4000 ...
4.4001    4.3999    4.4002    4.3998    4.4002    4.3995    4.4002    4.4000
4.4006    4.3998    4.4001    4.4000    4.4002    4.3994    4.4001    4.4000
4.4003    4.3998    4.3999    4.3997    4.3999    4.3999    4.4002    4.4001
4.3997    4.3998    4.4000    4.3998    4.3998    4.4002    4.4005    4.4000
4.4001    4.4002    4.3996    4.3993    4.4000    4.3999    4.3999    4.4000
4.3998    4.3991    4.4000    4.4006    4.3999    4.3999    4.4001    4.4002
4.4001    4.4002    4.4003    4.3998    4.4001    4.4000    4.4001    4.4000
4.4004    4.4000    4.3999    4.3998    4.3999    4.3997    4.4000    4.3999
4.4002    4.4000    4.4000    4.4002    4.3999    4.3999    4.3998    4.4000
⋮
```

```
% https://kr.mathworks.com/matlabcentral/answers/80480-how-do-i-take-the-average-of-every-n-values
blockSize = [1, 5] % 다섯개를 묶어서.
```

```
blockSize = 1x2
         1     5
```

```
meanFilterFunction = @(theBlockStructure) mean2(theBlockStructure.data(:))
```

```
meanFilterFunction = function_handle with value:
    @(theBlockStructure)mean2(theBlockStructure.data(:))
```

```
Ic = blockproc(IcFull, blockSize, meanFilterFunction)
```

Ic = 42855×20

```
4.4002    4.4002    4.4000    4.4002    4.4000    4.3999    4.3999    4.4000 ...
4.4000    4.4001    4.3998    4.4001    4.4001    4.4000    4.4001    4.3999
4.4002    4.3999    4.3997    4.3999    4.3999    4.3999    4.4001    4.3999
4.3999    4.4001    4.3999    4.4000    4.4001    4.4000    4.4000    4.3999
4.3998    4.4001    4.4001    4.4001    4.4003    4.4000    4.3999    4.4000
4.3998    4.3999    4.4002    4.4000    4.4002    4.4000    4.4000    4.4000
4.3999    4.4001    4.4000    4.4000    4.4001    4.4002    4.4000    4.3998
4.4001    4.4000    4.4000    4.3999    4.4000    4.4001    4.4000    4.4000
```

```

4.4000    4.4000    4.4000    4.4002    4.3999    4.4001    4.4000    4.4000
4.4000    4.3999    4.4000    4.4001    4.4000    4.4000    4.4000    4.3998
:

```

```
%clear IcFull
```

```
Vd = blockproc(VdFull, blockSize, meanFilterFunction)
```

```

Vd = 42855x20
    3.2196    3.1616    3.1480    3.1376    3.1274    3.1164    3.1044    3.0915 ...
    3.2189    3.1612    3.1476    3.1373    3.1271    3.1161    3.1041    3.0912
    3.2198    3.1619    3.1482    3.1378    3.1275    3.1165    3.1045    3.0915
    3.2194    3.1614    3.1478    3.1375    3.1272    3.1162    3.1042    3.0912
    3.2192    3.1612    3.1477    3.1374    3.1272    3.1162    3.1042    3.0913
    3.2189    3.1609    3.1474    3.1371    3.1269    3.1160    3.1040    3.0910
    3.2194    3.1613    3.1477    3.1373    3.1271    3.1160    3.1040    3.0910
    3.2191    3.1610    3.1473    3.1370    3.1268    3.1157    3.1038    3.0908
    3.2193    3.1611    3.1475    3.1372    3.1269    3.1159    3.1038    3.0908
    3.2193    3.1613    3.1475    3.1372    3.1270    3.1159    3.1039    3.0909
:

```

```
%clear VdFull
```

```

Vc = blockproc(VcFull, blockSize, meanFilterFunction);
Td = blockproc(TdFull, blockSize, meanFilterFunction);
Tc = blockproc(TcFull, blockSize, meanFilterFunction);
Id = blockproc(IdFull, blockSize, meanFilterFunction);

```

```

%ryan5_gen_data_raw_desc = "ryan5_gen_data_raw_desc";
%save('ryan5_data_raw.mat', 'ryan5_gen_data_raw_desc');
%save('ryan5_data_raw.mat', 'ryan5_battery_dataset', '-append');

```

```

% https://stackoverflow.com/questions/39149677/how-to-delete-a-variable-from-mat-file-in-matlab
%rmrmatvar('ryan5_gen_data.mat', 'ryan5_battery_dataset');

```

```
%clear ryan5_battery_dataset
```

```

[y_norm_Ic_RUL, y_max_Ic_RUL, y_min_Ic_RUL, y_rate_Ic_RUL, x_norm_Ic_RUL, x_max_Ic_RUL, x_min_Ic_RUL, x_rate_Ic_RUL] = ...
[y_norm_Vd_RUL, y_max_Vd_RUL, y_min_Vd_RUL, y_rate_Vd_RUL, x_norm_Vd_RUL, x_max_Vd_RUL, x_min_Vd_RUL, x_rate_Vd_RUL] = ...

Y_trainValid_Ic_RUL = RUL([idxTrainAll idxValidAll],:);
Y_test_Ic_RUL = RUL([idxTestAll],:);

```

```
x_trainValid_Ic_RUL = xnorm_Ic_RUL([idxTrainAll idxValidAll],:)
```

x_trainValid Ic RUL = 38367x20

0.5870	0.5870	0.5870	0.5870	0.5870	0.5870	0.5870	0.5870 . . .
0.5870	0.5870	0.5870	0.5870	0.5870	0.5870	0.5870	0.5870
0.5870	0.5870	0.5870	0.5870	0.5870	0.5870	0.5870	0.5870
0.5870	0.5870	0.5870	0.5870	0.5870	0.5870	0.5870	0.5870
0.5870	0.5870	0.5870	0.5870	0.5870	0.5870	0.5870	0.5870
0.5870	0.5870	0.5870	0.5870	0.5870	0.5870	0.5870	0.5870
0.5870	0.5870	0.5870	0.5870	0.5870	0.5870	0.5870	0.5870
0.5870	0.5870	0.5870	0.5870	0.5870	0.5870	0.5870	0.5870
0.5870	0.5870	0.5870	0.5870	0.5870	0.5870	0.5870	0.5870
0.5870	0.5870	0.5870	0.5870	0.5870	0.5870	0.5870	0.5870
0.5870	0.5870	0.5870	0.5870	0.5870	0.5870	0.5870	0.5870
⋮							

```
x_test_Ic_RUL = xnorm_Ic_RUL([idxTestAll],:)
```

$$x_test_Ic_RUL = 4488 \times 20$$

0.7703	0.7703	0.7702	0.7703	0.7702	0.7703	0.7702	0.5641 . . .
0.7703	0.7703	0.7702	0.7702	0.7702	0.7703	0.7702	0.5641
0.7702	0.7702	0.7703	0.7703	0.7703	0.7702	0.7703	0.5641
0.7703	0.7703	0.7702	0.7702	0.7703	0.7703	0.7703	0.5641
0.7703	0.7703	0.7703	0.7703	0.7703	0.7702	0.7702	0.5641
0.7703	0.7702	0.7702	0.7703	0.7702	0.7703	0.7702	0.5641
0.7703	0.7702	0.7703	0.7703	0.7702	0.7702	0.7703	0.5640
0.7703	0.7703	0.7702	0.7702	0.7702	0.7703	0.7703	0.5641
0.7703	0.7703	0.7702	0.7702	0.7702	0.7702	0.7702	0.5641
0.7703	0.7702	0.7703	0.7702	0.7702	0.7703	0.7702	0.5641
⋮							

```
y_trainValid_Ic_RUL = ynorm_Ic_RUL([idxTrainAll idxValidAll])
```

y_trainValid_Ic_RUL = 38367x1

$$\begin{pmatrix} 0.3650 \\ 0.3641 \\ 0.3632 \\ 0.3624 \\ 0.3615 \\ 0.3607 \\ 0.3598 \\ 0.3590 \\ 0.3581 \\ 0.3573 \\ \vdots \\ \vdots \end{pmatrix}$$

```
y_test_Ic_RUL = ynorm_Ic_RUL([idxTestAll])
```

y test Ic RUL = 4488x1

$$\begin{array}{c} 0.2692 \\ 0.2684 \\ 0.2675 \\ 0.2667 \\ 0.2658 \\ 0.2650 \\ 0.2641 \\ 0.2632 \\ 0.2624 \\ 0.2615 \\ \vdots \\ \vdots \end{array}$$

```

Y_trainValid_Vd_RUL = RUL([idxTrainAll idxValidAll],:);
Y_test_Vd_RUL = RUL([idxTestAll],:);
x_trainValid_Vd_RUL = xnorm_Vd_RUL([idxTrainAll idxValidAll],:)

```

```

x_trainValid_Vd_RUL = 38367x20
    0.9175    0.8739    0.8637    0.8559    0.8482    0.8400    0.8309    0.8212 ...
    0.9170    0.8736    0.8634    0.8556    0.8480    0.8397    0.8307    0.8210
    0.9177    0.8742    0.8638    0.8560    0.8483    0.8400    0.8310    0.8212
    0.9173    0.8737    0.8635    0.8558    0.8481    0.8398    0.8308    0.8210
    0.9172    0.8736    0.8635    0.8557    0.8480    0.8398    0.8308    0.8211
    0.9170    0.8734    0.8632    0.8555    0.8478    0.8397    0.8306    0.8209
    0.9173    0.8737    0.8634    0.8556    0.8480    0.8397    0.8307    0.8209
    0.9171    0.8734    0.8632    0.8554    0.8477    0.8394    0.8305    0.8207
    0.9172    0.8735    0.8633    0.8556    0.8478    0.8396    0.8305    0.8207
    0.9173    0.8737    0.8633    0.8556    0.8479    0.8396    0.8306    0.8208
    ⋮

```

```

x_test_Vd_RUL = xnorm_Vd_RUL([idxTestAll],:)

```

```

x_test_Vd_RUL = 4488x20
    0.9222    0.8786    0.8682    0.8605    0.8527    0.8443    0.8352    0.8253 ...
    0.9218    0.8783    0.8680    0.8601    0.8523    0.8440    0.8348    0.8250
    0.9217    0.8781    0.8678    0.8599    0.8521    0.8438    0.8346    0.8247
    0.9221    0.8784    0.8680    0.8602    0.8524    0.8440    0.8349    0.8249
    0.9222    0.8786    0.8682    0.8603    0.8525    0.8441    0.8349    0.8250
    0.9219    0.8783    0.8679    0.8600    0.8522    0.8438    0.8347    0.8248
    0.9220    0.8783    0.8678    0.8600    0.8522    0.8438    0.8347    0.8247
    0.9216    0.8779    0.8676    0.8597    0.8520    0.8436    0.8344    0.8246
    0.9217    0.8780    0.8676    0.8597    0.8519    0.8436    0.8344    0.8245
    0.9217    0.8780    0.8676    0.8597    0.8519    0.8436    0.8344    0.8245
    ⋮

```

```

y_trainValid_Vd_RUL = ynorm_Vd_RUL([idxTrainAll idxValidAll])

```

```

y_trainValid_Vd_RUL = 38367x1
    0.3650
    0.3641
    0.3632
    0.3624
    0.3615
    0.3607
    0.3598
    0.3590
    0.3581
    0.3573
    ⋮

```

```

y_test_Vd_RUL = ynorm_Vd_RUL([idxTestAll])

```

```

y_test_Vd_RUL = 4488x1
    0.2692
    0.2684
    0.2675
    0.2667
    0.2658
    0.2650
    0.2641
    0.2632
    0.2624

```

```
0.2615  
:  
:
```

```
all(Y_trainValid_Ic_RUL == Y_trainValid_Vd_RUL)
```

```
ans = logical  
1
```

```
all(Y_test_Ic_RUL == Y_test_Vd_RUL)
```

```
ans = logical  
1
```

```
all(y_trainValid_Ic_RUL == y_trainValid_Vd_RUL)
```

```
ans = logical  
1
```

```
all(y_test_Ic_RUL == y_test_Vd_RUL)
```

```
ans = logical  
1
```

```
all(ynorm_Ic_RUL == ynorm_Vd_RUL)
```

```
ans = logical  
1
```

```
ynorm_RUL = ynorm_Ic_RUL ;  
ymax_RUL = ymax_Ic_RUL ;  
ymin_RUL = ymin_Ic_RUL ;  
yrate_RUL = yrate_Ic_RUL ;
```

```
Y_trainValid_RUL = Y_trainValid_Vd_RUL ;  
Y_test_RUL = Y_test_Vd_RUL ;  
y_trainValid_RUL = y_trainValid_Vd_RUL ;  
y_test_RUL = y_test_Vd_RUL ;
```

```
[ynorm_SOH,ymax_SOH,ymin_SOH,yrate_SOH,xnorm_Ic_SOH,xmax_Ic_SOH,xmin_Ic_SOH,xrate_Ic_SOH] = min(ynorm_SOH,ymax_SOH,ymin_SOH,yrate_SOH,xnorm_Ic_SOH,xmax_Ic_SOH,xmin_Ic_SOH,xrate_Ic_SOH);  
[ynorm_SOH,ymax_SOH,ymin_SOH,yrate_SOH,xnorm_Vd_SOH,xmax_Vd_SOH,xmin_Vd_SOH,xrate_Vd_SOH] = min(ynorm_SOH,ymax_SOH,ymin_SOH,yrate_SOH,xnorm_Vd_SOH,xmax_Vd_SOH,xmin_Vd_SOH,xrate_Vd_SOH);
```

```
Y_trainValid_SOH = Cap([idxTrainAll idxValidAll],:);  
Y_test_SOH = Cap([idxTestAll],:);  
y_trainValid_SOH = ynorm_SOH([idxTrainAll idxValidAll]);  
y_test_SOH = ynorm_SOH([idxTestAll]);  
  
x_trainValid_Ic_SOH = xnorm_Ic_SOH([idxTrainAll idxValidAll],:);
```



```
x_test_Ic_SOH = xnorm_Ic_SOH([idxTestAll],:);
x_trainValid_Vd_SOH = xnorm_Vd_SOH([idxTrainAll idxValidAll],:);
x_test_Vd_SOH = xnorm_Vd_SOH([idxTestAll],:);
```

```
[xnorm_Tc,xmax_Tc,xmin_Tc,xrate_Tc] = minmax_norm_x(Tc);
x_trainValid_Tc = xnorm_Tc([idxTrainAll idxValidAll],:);
x_test_Tc = xnorm_Tc([idxTestAll],:);
```

```
[xnorm_Td,xmax_Td,xmin_Td,xrate_Td] = minmax_norm_x(Td);
x_trainValid_Td = xnorm_Td([idxTrainAll idxValidAll],:);
x_test_Td = xnorm_Td([idxTestAll],:);
```

```
[xnorm_Vc,xmax_Vc,xmin_Vc,xrate_Vc] = minmax_norm_x(Vc);
x_trainValid_Vc = xnorm_Vc([idxTrainAll idxValidAll],:);
x_test_Vc = xnorm_Vc([idxTestAll],:);
```

```
[xnorm_Id,xmax_Id,xmin_Id,xrate_Id] = minmax_norm_x(Id);
x_trainValid_Id = xnorm_Id([idxTrainAll idxValidAll],:);
x_test_Id = xnorm_Id([idxTestAll],:);
```

```
[ynorm_SOHc,ymax_SOHc,ymin_SOHc,yrate_SOHc] = minmax_norm_x(Cap_c);
```

```
Y_trainValid_SOHc = Cap_c([idxTrainAll idxValidAll],:);
Y_test_SOHc = Cap_c([idxTestAll],:);
y_trainValid_SOHc = ynorm_SOHc([idxTrainAll idxValidAll]);
y_test_SOHc = ynorm_SOHc([idxTestAll]);
```

```
cap_min = zeros();
for i = 1: length(idxAll)
    cap_min = [cap_min min(ryan5_battery_dataset(i).QDischargeSmooth)];
end
cap_min = cap_min(2:end)
```

```
cap_min = 1×119
    0.8797    0.8776    0.8811    0.8808    0.8807    0.8799    0.8829    0.8812 ...
```

```
min(cap_min)
```

```
ans = 0.8763
```

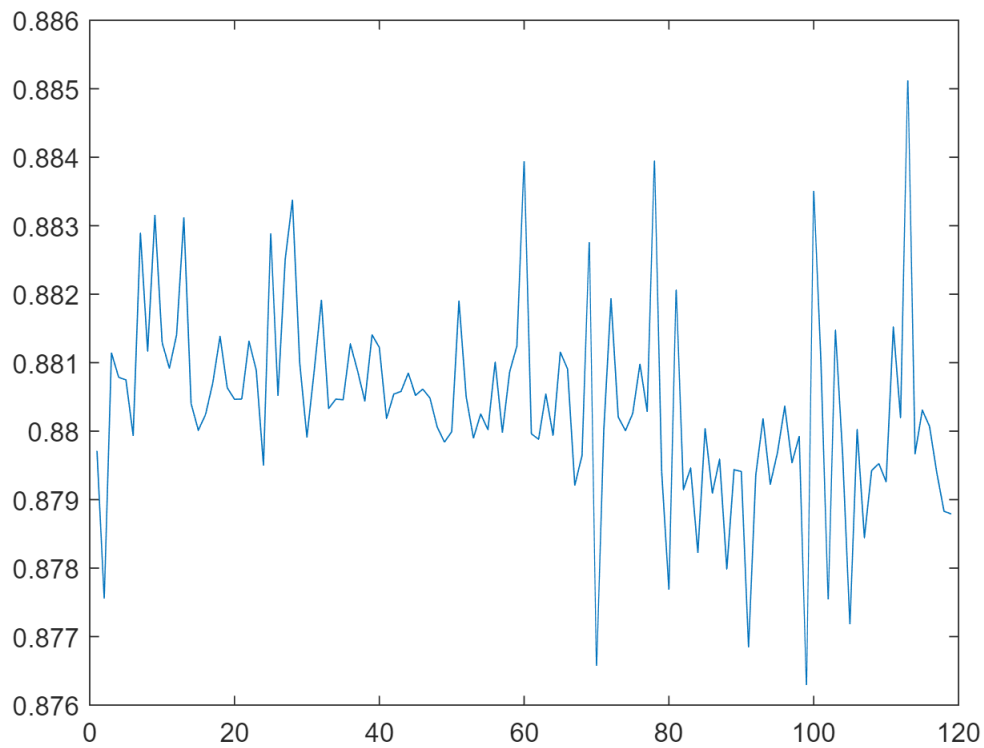
```
max(cap_min)
```

```
ans = 0.8851
```

```
Cap_EOL = mean(cap_min)
```

```
Cap_EOL = 0.8804
```

```
plot(cap_min)
```



```
cap_max = zeros();
for i = 1: length(idxAll)
    cap_max = [cap_max max(ryan5_battery_dataset(i).QDischargeSmooth)];
end
cap_max = cap_max(2:end)
```

```
cap_max = 1×119
    1.0479    1.0477    1.0476    1.0478    1.0479    1.0478    1.0478    1.0478 ...
```

```
min(cap_max)
```

```
ans = 1.0463
```

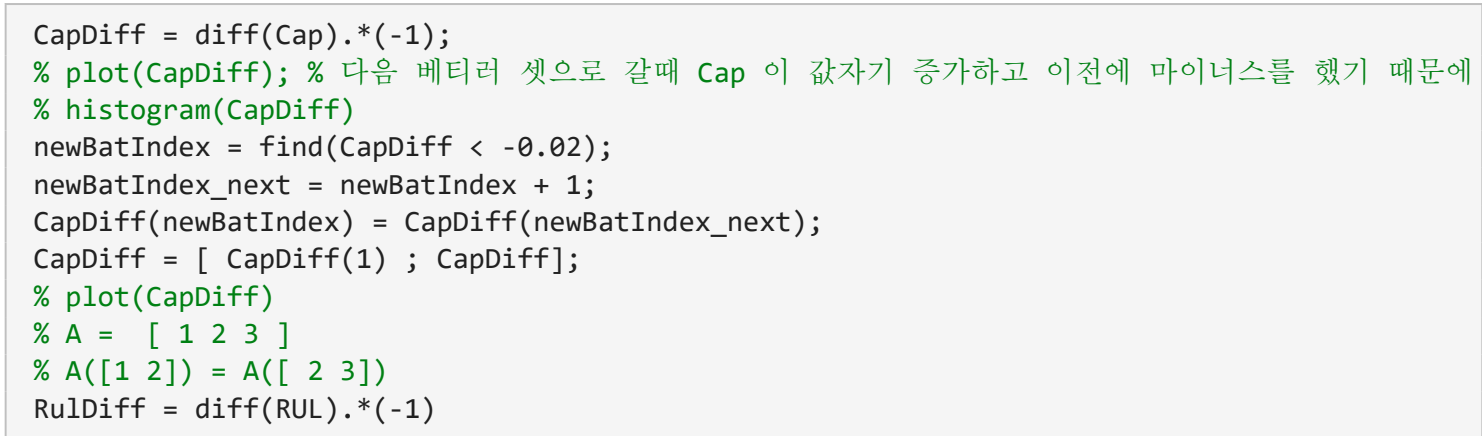
```
max(cap_max)
```

```
ans = 1.0500
```

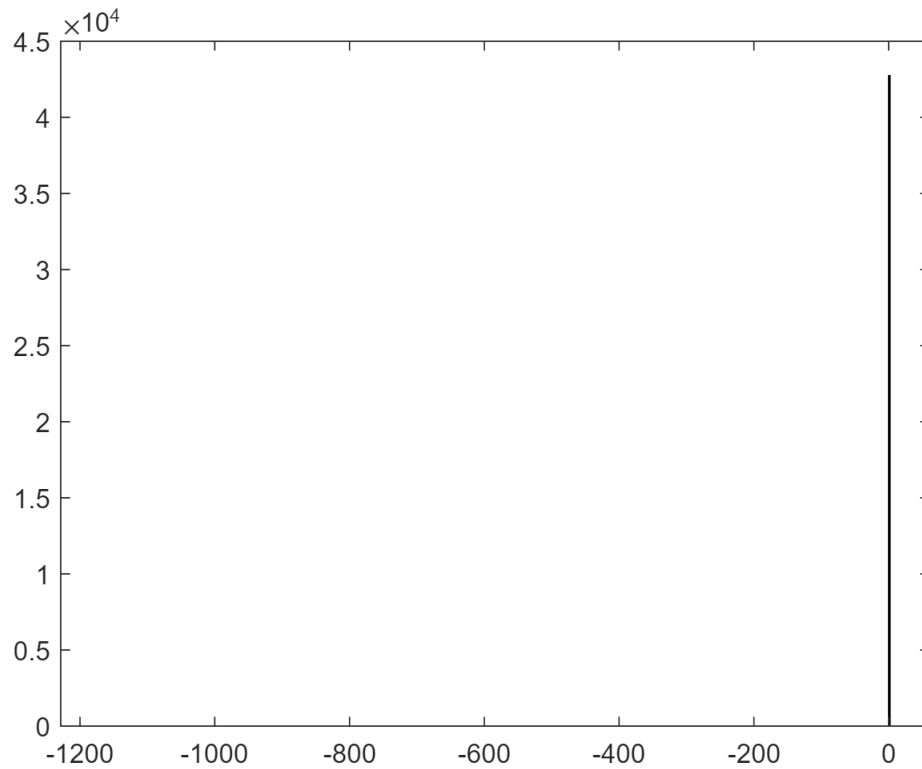
```
Cap_Rated = mean(cap_max)
```

```
Cap_Rated = 1.0477
```

```
plot(cap_max)
```


$$\begin{matrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ \vdots \end{matrix}$$

35



```
newBatIndex = find(RulDiff < -10);
newBatIndex_next = newBatIndex + 1;
RulDiff(newBatIndex) = RulDiff(newBatIndex_next);
RulDiff = [ RulDiff(1) ; RulDiff];
% plot(RulDiff);
```

```
[ynorm_DeltaSOH,ymax_DeltaSOH,ymin_DeltaSOH,yrate_DeltaSOH] = minmax_norm_x(CapDiff);

Y_trainValid_DeltaSOH = CapDiff([idxTrainAll idxValidAll],:);
Y_test_DeltaSOH = CapDiff([idxTestAll],:);
y_trainValid_DeltaSOH = ynorm_DeltaSOH([idxTrainAll idxValidAll]);
y_test_DeltaSOH = ynorm_DeltaSOH([idxTestAll]);

[ynorm_DeltaRUL,ymax_DeltaRUL,ymin_DeltaRUL,yrate_DeltaRUL] = minmax_norm_x(RulDiff);

Y_trainValid_DeltaRUL = RulDiff([idxTrainAll idxValidAll],:);
Y_test_DeltaRUL = RulDiff([idxTestAll],:);
y_trainValid_DeltaRUL = ynorm_DeltaRUL([idxTrainAll idxValidAll]);
y_test_DeltaRUL = ynorm_DeltaRUL([idxTestAll]);
```

```
Y_trainValid_SOH_DeltaSOH = [ Y_trainValid_SOH Y_trainValid_DeltaSOH ];
Y_test_SOH_DeltaSOH = [ Y_test_SOH Y_test_DeltaSOH ];
y_trainValid_SOH_DeltaSOH = [ y_trainValid_SOH y_trainValid_DeltaSOH ];
y_test_SOH_DeltaSOH = [ y_test_SOH y_test_DeltaSOH ];
```

```
ynorm_SOH_DeltaSOH = [ ynorm_SOH ynorm_DeltaSOH ];
% y_trainValid_RUL
```

실제 실험에 사용될 데이터

```
save('ryan5_data.mat','Ic','-append');
save('ryan5_data.mat','Vd','-append');
save('ryan5_data.mat','Cap','-append');
save('ryan5_data.mat','IcFull','-append');
save('ryan5_data.mat','Ic','-append');
save('ryan5_data.mat','VdFull','-append');
save('ryan5_data.mat','Vd','-append');

save('ryan5_data.mat','VcFull','-append');
save('ryan5_data.mat','TdFull','-append');
save('ryan5_data.mat','TcFull','-append');
save('ryan5_data.mat','IdFull','-append');
save('ryan5_data.mat','Vc','-append');
save('ryan5_data.mat','Td','-append');
save('ryan5_data.mat','Tc','-append');
save('ryan5_data.mat','Id','-append');
save('ryan5_data.mat','Cap_c','-append');

save('ryan5_data.mat','idxTrainAll','-append');
save('ryan5_data.mat','idxValidAll','-append');
save('ryan5_data.mat','idxTestAll','-append');
save('ryan5_data.mat','idxAllAll','-append');

save('ryan5_data.mat','idxTrainAll_batIndex','-append');
save('ryan5_data.mat','idxValidAll_batIndex','-append');
save('ryan5_data.mat','idxTestAll_batIndex','-append');
save('ryan5_data.mat','idxAllAll_batIndex','-append');

save('ryan5_data.mat','RUL','-append');

%%%%%% First - common
save('ryan5_data.mat','RulFirst','-append');
save('ryan5_data.mat','IcFirstFull','-append');
save('ryan5_data.mat','IcFirst','-append');
save('ryan5_data.mat','VdFirstFull','-append');
save('ryan5_data.mat','VdFirst','-append');
save('ryan5_data.mat','ynorm_First_RUL','-append');
save('ryan5_data.mat','ymax_First_RUL','-append');
save('ryan5_data.mat','ymin_First_RUL','-append');
save('ryan5_data.mat','yrate_First_RUL','-append');
save('ryan5_data.mat','y_trainValid_First_RUL','-append');
save('ryan5_data.mat','y_test_First_RUL','-append');
save('ryan5_data.mat','Y_trainValid_First_RUL','-append');
```

```

save('ryan5_data.mat','Y_test_First_RUL','-append');

%%%%%%%%%% IcFirst / RUL
save('ryan5_data.mat','ynorm_IcFirst_RUL','-append');
save('ryan5_data.mat','ymax_IcFirst_RUL','-append');
save('ryan5_data.mat','ymin_IcFirst_RUL','-append');
save('ryan5_data.mat','yrate_IcFirst_RUL','-append');
save('ryan5_data.mat','xnorm_IcFirstFull_RUL','-append');
save('ryan5_data.mat','xmax_IcFirst_RUL','-append');
save('ryan5_data.mat','xmin_IcFirst_RUL','-append');
save('ryan5_data.mat','xrate_IcFirst_RUL','-append');
save('ryan5_data.mat','xnorm_IcFirst_RUL','-append');

save('ryan5_data.mat','x_trainValid_IcFirstFull_RUL','-append');
save('ryan5_data.mat','x_test_IcFirstFull_RUL','-append');
save('ryan5_data.mat','x_trainValid_IcFirst_RUL','-append');
save('ryan5_data.mat','x_test_IcFirst_RUL','-append');
save('ryan5_data.mat','y_trainValid_IcFirst_RUL','-append');
save('ryan5_data.mat','y_test_IcFirst_RUL','-append');
save('ryan5_data.mat','Y_trainValid_IcFirst_RUL','-append');
save('ryan5_data.mat','Y_test_IcFirst_RUL','-append');

%%%%%%%%%% VdFirst / RUL
save('ryan5_data.mat','ynorm_VdFirst_RUL','-append');
save('ryan5_data.mat','ymax_VdFirst_RUL','-append');
save('ryan5_data.mat','ymin_VdFirst_RUL','-append');
save('ryan5_data.mat','yrate_VdFirst_RUL','-append');
save('ryan5_data.mat','xnorm_VdFirstFull_RUL','-append');
save('ryan5_data.mat','xmax_VdFirst_RUL','-append');
save('ryan5_data.mat','xmin_VdFirst_RUL','-append');
save('ryan5_data.mat','xrate_VdFirst_RUL','-append');
save('ryan5_data.mat','xnorm_VdFirst_RUL','-append');

%save('ryan5_data.mat','x_trainValid_IcFirstFull_RUL','-append');
%save('ryan5_data.mat','x_test_IcFirstFull_RUL','-append');
save('ryan5_data.mat','x_trainValid_VdFirstFull_RUL','-append');
save('ryan5_data.mat','x_trainValid_VdFirst_RUL','-append');
save('ryan5_data.mat','x_test_VdFirst_RUL','-append');
save('ryan5_data.mat','y_trainValid_VdFirst_RUL','-append');
save('ryan5_data.mat','y_test_VdFirst_RUL','-append');
save('ryan5_data.mat','Y_trainValid_VdFirst_RUL','-append');
save('ryan5_data.mat','Y_test_VdFirst_RUL','-append');

```

```

%%%%%%%%%% Ic Vd RUL common
save('ryan5_data.mat','ynorm_RUL','-append');
save('ryan5_data.mat','ymax_RUL','-append');

```

```

save('ryan5_data.mat','ymin_RUL','-append');
save('ryan5_data.mat','yrate_RUL','-append');
save('ryan5_data.mat','y_trainValid_RUL','-append');
save('ryan5_data.mat','y_test_RUL','-append');
save('ryan5_data.mat','Y_trainValid_RUL','-append');
save('ryan5_data.mat','Y_test_RUL','-append');

%%%%%%%%%% Ic / RUL
save('ryan5_data.mat','ynorm_Ic_RUL','-append');
save('ryan5_data.mat','ymax_Ic_RUL','-append');
save('ryan5_data.mat','ymin_Ic_RUL','-append');
save('ryan5_data.mat','yrate_Ic_RUL','-append');
save('ryan5_data.mat','xnorm_Ic_RUL','-append');
save('ryan5_data.mat','xmax_Ic_RUL','-append');
save('ryan5_data.mat','xmin_Ic_RUL','-append');
save('ryan5_data.mat','xrate_Ic_RUL','-append');

save('ryan5_data.mat','x_trainValid_Ic_RUL','-append');
save('ryan5_data.mat','x_test_Ic_RUL','-append');
save('ryan5_data.mat','y_trainValid_Ic_RUL','-append');
save('ryan5_data.mat','y_test_Ic_RUL','-append');
save('ryan5_data.mat','Y_trainValid_Ic_RUL','-append');
save('ryan5_data.mat','Y_test_Ic_RUL','-append');

%%%%%%%%%% Vd / RUL
save('ryan5_data.mat','ynorm_Vd_RUL','-append');
save('ryan5_data.mat','ymax_Vd_RUL','-append');
save('ryan5_data.mat','ymin_Vd_RUL','-append');
save('ryan5_data.mat','yrate_Vd_RUL','-append');
save('ryan5_data.mat','xnorm_Vd_RUL','-append');
save('ryan5_data.mat','xmax_Vd_RUL','-append');
save('ryan5_data.mat','xmin_Vd_RUL','-append');
save('ryan5_data.mat','xrate_Vd_RUL','-append');

save('ryan5_data.mat','x_trainValid_Vd_RUL','-append');
save('ryan5_data.mat','x_test_Vd_RUL','-append');
save('ryan5_data.mat','y_trainValid_Vd_RUL','-append');
save('ryan5_data.mat','y_test_Vd_RUL','-append');
save('ryan5_data.mat','Y_trainValid_Vd_RUL','-append');
save('ryan5_data.mat','Y_test_Vd_RUL','-append');

```

```

%%%%%%%%%% SOH
save('ryan5_data.mat','ynorm_SOH','-append');
save('ryan5_data.mat','ymax_SOH','-append');
save('ryan5_data.mat','ymin_SOH','-append');

```

```

save('ryan5_data.mat','yrate_SOH','-append');
save('ryan5_data.mat','y_trainValid_SOH','-append');
save('ryan5_data.mat','y_test_SOH','-append');
save('ryan5_data.mat','Y_trainValid_SOH','-append');
save('ryan5_data.mat','Y_test_SOH','-append');

%%%%%% Ic / SOH
save('ryan5_data.mat','xnorm_Ic_SOH','-append');
save('ryan5_data.mat','xmax_Ic_SOH','-append');
save('ryan5_data.mat','xmin_Ic_SOH','-append');
save('ryan5_data.mat','xrate_Ic_SOH','-append');
save('ryan5_data.mat','x_trainValid_Ic_SOH','-append');
save('ryan5_data.mat','x_test_Ic_SOH','-append');
%%%%%% Vd / SOH
save('ryan5_data.mat','xnorm_Vd_SOH','-append');
save('ryan5_data.mat','xmax_Vd_SOH','-append');
save('ryan5_data.mat','xmin_Vd_SOH','-append');
save('ryan5_data.mat','xrate_Vd_SOH','-append');
save('ryan5_data.mat','x_trainValid_Vd_SOH','-append');
save('ryan5_data.mat','x_test_Vd_SOH','-append');

```

```

%%%% SOH == RUL   for x value
x_trainValid_Ic = x_trainValid_Ic_SOH;
x_trainValid_Vd = x_trainValid_Vd_SOH;
x_test_Ic = x_test_Ic_SOH;
x_test_Vd = x_test_Vd_SOH;

xnorm_Ic = xnorm_Ic_SOH;
%xmax_Ic = xmax_Ic_SOH;
xmin_Ic = xmin_Ic_SOH;
xrate_Ic = xrate_Ic_SOH;

xnorm_Vd = xnorm_Vd_SOH;
%xmax_Vd = xmax_Vd_SOH;
xmin_Vd = xmin_Vd_SOH;
xrate_Vd = xrate_Vd_SOH;

%%%%%%%% common
save('ryan5_data.mat','x_trainValid_Ic','-append');
save('ryan5_data.mat','x_trainValid_Vd','-append');
save('ryan5_data.mat','x_test_Ic','-append');
save('ryan5_data.mat','x_test_Vd','-append');

save('ryan5_data.mat','xnorm_Ic','-append');
%save('ryan5_data.mat','xmax_Ic','-append');
save('ryan5_data.mat','xmin_Ic','-append');
save('ryan5_data.mat','xrate_Ic','-append');

```



```
save('ryan5_data.mat','xnorm_Vd','-append');
%save('ryan5_data.mat','xmax_Vd','-append');
save('ryan5_data.mat','xmin_Vd','-append');
save('ryan5_data.mat','xrate_Vd','-append');
```

```
save('ryan5_data.mat','x_trainValid_Id','-append');
save('ryan5_data.mat','x_test_Id','-append');
save('ryan5_data.mat','xnorm_Id','-append');
%save('ryan5_data.mat','xmax_Id','-append');
save('ryan5_data.mat','xmin_Id','-append');
save('ryan5_data.mat','xrate_Id','-append');
```

```
save('ryan5_data.mat','x_trainValid_Vc','-append');
save('ryan5_data.mat','x_test_Vc','-append');
save('ryan5_data.mat','xnorm_Vc','-append');
%save('ryan5_data.mat','xmax_Vc','-append');
save('ryan5_data.mat','xmin_Vc','-append');
save('ryan5_data.mat','xrate_Vc','-append');
```

```
save('ryan5_data.mat','x_trainValid_Tc','-append');
save('ryan5_data.mat','x_test_Tc','-append');
save('ryan5_data.mat','xnorm_Tc','-append');
%save('ryan5_data.mat','xmax_Tc','-append');
save('ryan5_data.mat','xmin_Tc','-append');
save('ryan5_data.mat','xrate_Tc','-append');
```

```
save('ryan5_data.mat','x_trainValid_Td','-append');
save('ryan5_data.mat','x_test_Td','-append');
save('ryan5_data.mat','xnorm_Td','-append');
%save('ryan5_data.mat','xmax_Td','-append');
save('ryan5_data.mat','xmin_Td','-append');
save('ryan5_data.mat','xrate_Td','-append');
```

```
save('ryan5_data.mat','CapDiff','-append');
save('ryan5_data.mat','RulDiff','-append');
```

```
save('ryan5_data.mat','ynorm_SOHc','-append');
save('ryan5_data.mat','ymax_SOHc','-append');
save('ryan5_data.mat','ymin_SOHc','-append');
save('ryan5_data.mat','yrate_SOHc','-append');
save('ryan5_data.mat','y_trainValid_SOHc','-append');
save('ryan5_data.mat','y_test_SOHc','-append');
save('ryan5_data.mat','Y_trainValid_SOHc','-append');
save('ryan5_data.mat','Y_test_SOHc','-append');
```

```
save('ryan5_data.mat','ynorm_DeltaSOH','-append');
save('ryan5_data.mat','ymax_DeltaSOH','-append');
save('ryan5_data.mat','ymin_DeltaSOH','-append');
save('ryan5_data.mat','yrate_DeltaSOH','-append');
save('ryan5_data.mat','y_trainValid_DeltaSOH','-append');
save('ryan5_data.mat','y_test_DeltaSOH','-append');
save('ryan5_data.mat','Y_trainValid_DeltaSOH','-append');
save('ryan5_data.mat','Y_test_DeltaSOH','-append');

save('ryan5_data.mat','ynorm_DeltaRUL','-append');
save('ryan5_data.mat','ymax_DeltaRUL','-append');
save('ryan5_data.mat','ymin_DeltaRUL','-append');
save('ryan5_data.mat','yrate_DeltaRUL','-append');
save('ryan5_data.mat','y_trainValid_DeltaRUL','-append');
save('ryan5_data.mat','y_test_DeltaRUL','-append');
save('ryan5_data.mat','Y_trainValid_DeltaRUL','-append');
save('ryan5_data.mat','Y_test_DeltaRUL','-append');
```

```
save('ryan5_data.mat','Cap_Rated','-append');
save('ryan5_data.mat','Cap_EOL','-append');

save('ryan5_data.mat','Y_trainValid_SOH_DeltaSOH','-append');
save('ryan5_data.mat','Y_test_SOH_DeltaSOH','-append');
save('ryan5_data.mat','y_trainValid_SOH_DeltaSOH','-append');
save('ryan5_data.mat','y_test_SOH_DeltaSOH','-append');
save('ryan5_data.mat','ynorm_SOH_DeltaSOH','-append');
```

```
Cap_change = (Cap_Rated + Cap_EOL )/2
```

```
Cap_change = 1.9281
```

```
save('ryan5_data.mat','Cap_change','-append')
```