Test of the WEASEL algorithm

Szymon Górski

16 January 2022

Introduction

The test is to determine if the implemented WEASEL algorithm works in the same way as the original implementation. The "Beef" data set from 2015 UCR Time Series Classification Archive was chosen. Unfortunately, due to R language and environment limitations it was impossible to perform the full test of the data set. The issue is probably caused by prolonged time of modification and reassignment of large objects in R. Therefore, the test was limited to a smaller number of windows' lengths.

The code against which the new implementation was checked was the Python implementation shared by the algorithm designer (Patrick Schäfer) on his GitHub account. To make the comparison possible, the following modifications to the original code were introduced:

- The maximal window length to check was lowered from 249 to 19 (files `WEASELClassifier.py`, `WEASEL.py`),
- A piece of code responsible for printing calculated values was added in function predict (file `WEASELClassifier.py`),
- The randomization of trained samples were turned off to achieve identical results with both programs (file `WEASELClassifier.py`):

```
# random.shuffle(perm)
perm = [11, 13, 18, 9, 28, 29, 5, 15, 3, 6, 26, 25, 4, 2,
22, 1, 21, 0, 17, 7, 10, 20, 27, 23, 8, 19, 16, 24, 14, 12]
```

The same modifications were introduced in R implementation:

```
# perm <- sample(perm)
perm = c(12, 14, 19, 10, 29, 30, 6, 16, 4, 7, 27, 26, 5, 3, 23, 2,
         22, 1, 18, 8, 11, 21, 28, 24, 9, 20, 17, 25, 15, 13)
```

Evaluation

Separated Fast Fourier Transform (FFT) coefficients with respect to their position and labels of independent windows:

```
[Python: LINE 284]
ORDER LINE:
[[(-2.82, 2), (-2.81, 1),
  (-2.81, 2), (-2.81, 3),
  (-2.81, 3), (-2.81, 3),
  (-2.81, 5), (-2.81, 5),
  (-2.8, 2), (-2.8, 2),
  (-2.79, 1), (-2.79, 4),
  (-2.79, 5), (-2.79, 5),
  (-2.78, 1), (-2.78, 3),
  (-2.77, 2), (-2.77, 2),
  (-2.77, 4), (-2.77, 4),
  (-2.76, 1), (-2.76, 1),
  (-2.76, 2), (-2.75, 1),
  (-2.75, 3), (-2.74, 1),
  (-2.74, 3), (-2.74, 3),
  (-2.74, 3), (-2.74, 4),
  (-2.73, 1), (-2.73, 1),
  (-2.73, 3), (-2.73, 5),
  (-2.72, 1), (-2.72, 2),
  (-2.72, 5), (-2.72, 5),
  (-2.71, 1), (-2.71, 2),
  (-2.71, 3), (-2.71, 3),
  (-2.71, 4), (-2.7, 2),
  (-2.7, 5), (-2.69, 3),
  (-2.68, 2), (-2.68, 2),
  (-2.68, 5), (-2.67, 5),
  (-2.66, 1), (-2.66, 1),
  (-2.66, 2), (-2.66, 2),
  (-2.66, 3), (-2.66, 5),
  (-2.65, 1), (-2.65, 1),
  (-2.65, 4), (-2.65, 4),
  (-2.65, 5), (-2.65, 5),
  (-2.65, 5), (-2.64, 4),
  (-2.64, 5), (-2.64, 5),
  (-2.63, 1), (-2.63, 2),
  (-2.63, 4), (-2.63, 4),
  (-2.62, 1), (-2.62, 2),
 ...
```

```
[R: LINE 210]
$wordModel$signature[[1]]
  $orderLine
$wordModel$signature[[1]]
  $orderLine[[1]]
$wordModel$signature[[1]]
  $orderLine[[1]][[1]]
$wordModel$signature[[1]]
  $orderLine[[1]][[1]]$value
[1] -2.82

$wordModel$signature[[1]]
  $orderLine[[1]][[1]]$label
[1] 2


$wordModel$signature[[1]]
  $orderLine[[1]][[2]]
$wordModel$signature[[1]]
  $orderLine[[1]][[2]]$value
[1] -2.81

$wordModel$signature[[1]]
  $orderLine[[1]][[2]]$label
[1] 1


$wordModel$signature[[1]]
  $orderLine[[1]][[3]]
$wordModel$signature[[1]]
  $orderLine[[1]][[3]]$value
[1] -2.81

$wordModel$signature[[1]]
  $orderLine[[1]][[3]]$label
[1] 2
...
```

Values most efficiently separating the set of independent windows with respect to information gain (IG), defining letters:

```
[Python: LINE 317]
SPLIT POINTS:
[702, 1404, 2106]

[564, 1128, 1692]

[468, 936, 1404]

[402, 804, 1206]

[348, 696, 1044]

[312, 624, 936]

[282, 564, 846]

[252, 504, 756]

[234, 468, 702]

[216, 432, 648]

[198, 396, 594]

[186, 372, 558]

[174, 348, 522]

[162, 324, 486]

[156, 312, 468]

[144, 288, 432]
```

```
[R: LINE 112,540]
$wordModel$signature[[1]]
  $splitPoints
[1]  703 1405 2107

[LINE 225,536]
$wordModel$signature[[2]]
  $splitPoints
[1]  565 1129 1693

[LINE 338,062]
$wordModel$signature[[3]]
  $splitPoints
[1]  469  937 1405

[LINE 450,830]
$wordModel$signature[[4]]
  $splitPoints
[1]  403  805 1207

[LINE 562,401]
$wordModel$signature[[5]]
  $splitPoints
[1]  349  697 1045

[LINE 674,929]
$wordModel$signature[[6]]
 $splitPoints
[1] 313 625 937

[LINE 787,941]
$wordModel$signature[[7]]
  $splitPoints
[1] 283 565 847

[LINE 899,035]
$wordModel$signature[[8]]
  $splitPoints
[1] 253 505 757
...
```

Separation of values of independent windows with respect to split points and Fast Fourier Transform (FFT) coefficients:

```
[Python: LINE 1]
BINS:
     0     1     2     3
0 -inf -2.25 -2.03  2.13
1 -inf  2.67  2.78  2.78
2 -inf  0.00  0.00  0.00
3 -inf  0.00  0.00  0.00

     0     1     2     3
0 -inf  1.73  1.75  1.75
1 -inf -3.38 -3.35  3.36
2 -inf -3.18 -3.01  3.13
3 -inf  0.00  0.00  0.00
4 -inf  0.00  0.00  0.00

     0     1     2     3
0 -inf -3.65 -3.40  3.67
1 -inf -4.06 -3.72  3.70
2 -inf -1.86 -1.86  1.98
3 -inf -1.10  1.32  1.32
4 -inf  0.00  0.00  0.00
5 -inf  0.00  0.00  0.00

     0     1     2     3
0 -inf -0.63  0.01  0.01
1 -inf -4.75 -4.58  4.49
2 -inf -2.39 -2.30  2.58
3 -inf  1.92  3.08  3.09
4 -inf -2.78 -2.21  2.68
5 -inf  0.00  0.00  0.00
6 -inf  0.00  0.00  0.00

     0     1     2     3
0 -inf -4.83 -4.56  5.12
1 -inf -4.60 -4.59  4.78
2 -inf -1.79  2.64  2.65
3 -inf -2.29 -2.23  2.16
4 -inf  2.17  2.18  2.18
5 -inf -1.04 -1.00  0.87
6 -inf  0.00  0.00  0.00
7 -inf  0.00  0.00  0.00
...
```

```
[R: LINE 77]
$wordModel$signature[[1]]$bins
    X1    X2    X3   X4
1 -Inf -2.25 -2.03 2.13
2 -Inf  2.67  2.78 2.78
3 -Inf  0.00  0.00 0.00
4 -Inf  0.00  0.00 0.00

[LINE 112,597]
$wordModel$signature[[2]]$bins
    X1    X2    X3   X4
1 -Inf  1.73  1.75 1.75
2 -Inf -3.38 -3.35 3.36
3 -Inf -3.18 -3.01 3.13
4 -Inf  0.00  0.00 0.00
5 -Inf  0.00  0.00 0.00

[LINE 225,593]
$wordModel$signature[[3]]$bins
    X1    X2    X3   X4
1 -Inf -3.65 -3.40 3.67
2 -Inf -4.06 -3.72 3.70
3 -Inf -1.86 -1.86 1.98
4 -Inf -1.10  1.32 1.32
5 -Inf  0.00  0.00 0.00
6 -Inf  0.00  0.00 0.00

[LINE 338,119]
$wordModel$signature[[4]]$bins
    X1    X2    X3   X4
1 -Inf -0.63  0.01 0.01
2 -Inf -4.75 -4.58 4.49
3 -Inf -2.39 -2.30 2.58
4 -Inf  1.92  3.08 3.09
5 -Inf -2.78 -2.21 2.68
6 -Inf  0.00  0.00 0.00
7 -Inf  0.00  0.00 0.00
...
```

Best Fourier coefficients with respect to information gain (IG) which were chosen to create words:

```
[Python: LINE 350]
BEST VALUES:
[2, 3, 1, 0]

[3, 4, 0, 2, 1]

[4, 5, 3, 2, 1, 0]

[5, 6, 0, 2, 3, 4]

[6, 7, 4, 1, 3, 0]

[7, 8, 5, 2, 3, 6]

[8, 9, 2, 7, 1, 5]

[9, 10, 5, 0, 2, 8]

[10, 11, 7, 1, 5, 6]

[11, 12, 9, 4, 10, 6]

[12, 13, 0, 11, 2, 5]

[13, 14, 11, 4, 0, 6]

[14, 15, 8, 13, 4, 9]

[15, 16, 4, 13, 3, 5]

[16, 17, 13, 11, 12, 6]

[17, 18, 15, 9, 8, 12]
```

```
[R: LINE 112,565]
$wordModel$signature[[1]]
  $bestValues
[1] 3 4 2 1

[LINE 225,561]
$wordModel$signature[[2]]
  $bestValues
[1] 4 5 1 3 2

[LINE 338,087]
$wordModel$signature[[3]]
  $bestValues
[1] 5 6 4 3 2 1

[LINE 450,855]
$wordModel$signature[[4]]
  $bestValues
[1] 6 7 1 3 4 5

[LINE 562,426]
$wordModel$signature[[5]]
  $bestValues
[1] 7 8 5 2 4 1

[LINE 674,954]
$wordModel$signature[[6]]
  $bestValues
[1] 8 9 6 3 4 7

[LINE 787,966]
$wordModel$signature[[7]]
  $bestValues
[1]  9 10  3  8  2  6

[LINE 899,060]
$wordModel$signature[[8]]
  $bestValues
[1] 10 11  6  1  3  9
...
```

Normalization of sliding windows:

```
[Python: LINE 218]
MEANS:
[-0.15638300792125698,
 -0.16733518147607576,
 -0.17752057508539965,
 -0.18636154785695588,
 -0.19534565288500444,
 -0.20246137077919713,
 -0.2071668437113842,
 -0.21017773296402745,
 -0.2107655975889069,
 -0.21015217363251099,
 -0.2094671835478689,
 -0.20833746109483975,
 -0.20629015864036837,
 -0.20283964888564127,
 -0.19776612157961665,
 -0.19137117683418917,
 -0.18323564161248823,
 -0.17424131285183306,
 -0.1639613497158981,
 -0.151697982454283,
 -0.13857326572056197,
 -0.12432905026641838,
 -0.10961454311237122,
 -0.09537543952453094,
 -0.08140982078391716,
 -0.0693023654445527,
 -0.06098535896908474,
 -0.056203208042348185,
 -0.05511438051974547,
 -0.05608052325106902,
 -0.05542876029739834,
 -0.053419796840201754,
 -0.048977585022634586,
 -0.04084460573408537,
 -0.03050585613566248,
 -0.017851431101844947,
 -0.003596991915094754,
 0.01165170526764716,
 ...
```

```
[R: LINE 94]
$wordModel$signature[[1]]
  $transformation$means
 [1] -0.1563830079 -0.1673351815
     -0.1775205751 -0.1863615479
     -0.1953456529 -0.2024613708
     -0.2071668437 -0.2101777330
 [9] -0.2107655976 -0.2101521736
     -0.2094671835 -0.2083374611
     -0.2062901586 -0.2028396489
     -0.1977661216 -0.1913711768
[17] -0.1832356416 -0.1742413129
      -0.1639613497 -0.1516979825
      -0.1385732657 -0.1243290503
      -0.1096145431 -0.0953754395
[25] -0.0814098208 -0.0693023654
      -0.0609853590 -0.0562032080
      -0.0551143805 -0.0560805233
      -0.0554287603 -0.0534197968
[33] -0.0489775850 -0.0408446057
      -0.0305058561 -0.0178514311
      -0.0035969919  0.0116517053
      0.0284569657  0.0466322064
[41]  0.0652419557  0.0829034537
       0.0979834593  0.1105535386
       0.1195376436  0.1235402349
       0.1224820786  0.1147094859
[49]  0.1016026607  0.0859347905
       0.0687921468  0.0516009404
       0.0341183577  0.0165923240
       -0.0009311536 -0.0179613362
[57] -0.0326553959 -0.0456804313
       -0.0559271673 -0.0622863289
       -0.0656422692 -0.0646147840
       -0.0608371148 -0.0570262185
[65] -0.0538134105 -0.0531616476
      -0.0557942587 -0.0621048577
      -0.0729241227 -0.0872501280
      -0.1035697612 -0.1190357127
...
```

```
[Python: LINE 251]
STDS:
[0.012417236478639503,
 0.0115881284976611,
 0.010858630143560392,
 0.010035615614176763,
 0.009156196995716657,
 0.00761403991791489,
 0.005089000905482175,
 0.001345586537645807,
 0.0003857342640174462 4,
 0.000748145989694039,
 0.0010476457897072933,
 0.0014677823054724314,
 0.002638806337285787,
 0.004991927206189352,
 0.007000011727654186,
 0.008008475614655971,
 0.009435356963754344,
 0.010907818375628143,
 0.012554067174573054,
 0.014170289788845864,
 0.015818034686588132,
 0.016570053710160106,
 0.016335751410489766,
 0.015832288417041793,
 0.015220759394513186,
 0.013047758860238037,
 0.008366160336738804,
 0.00291956939397324 8,
 0.0011520087202441206,
 0.0011626771064576966,
 0.002153732770308441,
 0.004078406831636457,
 0.006776925049634054,
 0.010249192186123812,
 0.014017966844923853,
 0.0157735038369915,
 0.015967487805749588,
 0.017305193378512992,
 ...
```

```
[R: LINE 155]
$wordModel$signature[[1]]
  $transformation$STDs
  [1] 0.0124172365 0.0115881285
      0.0108586301 0.0100356156
      0.0091561970 0.0076140399
      0.0050890009 0.0013455865
      0.0003857343
 [10] 0.0007748146 0.0010476458
      0.0014677823 0.0026388063
      0.0049919272 0.0070000117
      0.0080084756 0.0094353570
      0.0109078184
 [19] 0.0125540672 0.0141702898
      0.0158180347 0.0165700537
      0.0163357514 0.0158322884
      0.0152207594 0.0130477589
      0.0083661603
 [28] 0.0029195694 0.0011520087
      0.0011626771 0.0021537328
      0.0040784068 0.0067769250
      0.0102491922 0.0140179668
      0.0157735038
 [37] 0.0159674878 0.0173051934
      0.0199919822 0.0217364640
      0.0211137336 0.0186575268
      0.0154404840 0.0125374738
      0.0085939717
 [46] 0.0043118403 0.0059886054
      0.0126352625 0.0177979114
      0.0198487882 0.0188884906
      0.0184197482 0.0201217220
      0.0204999636
 [55] 0.0193698461 0.0178043251
      0.0161251347 0.0135025367
      0.0096210946 0.0058465852
      0.0020038174 0.0037661473
      0.0062926197
 [64] 0.0058656143 0.0025579400
      0.0014821680 0.0039819896
...
```

Words created by transforming sliding windows and assigning letters to them:

```
[Python: LINE 383]
WORDS:
[[143, 207, 143, 143, 143, 143,
  15, 143, 143, 79, 143, 143, 79,
  143, 143, 143, 143, 143, 143,
  79, 143, 143, 143, 143, 15,
  143, 143, 79, 15, 143, 143, 15,
  143, 143, 15, 143, 143, 79,
  143, 143, 143, 143, 143, 143,
  143, 79, 143, 143, 15, 143,
  143, 143, 143, 207, 143, 143,
  143, 207, 143, 143, 207, 207,
  15, 143, 143, 207, 143, 143,
  207, 143, 143, 207, 143, 143,
  143, 143, 79, 143, 143, 143,
  143, 79, 143, 143, 79, 143,
  143, 15, 143, 143, 143, 207,
  143, 143, 143, 143, 143, 143,
  143, 207, 143, 143, 207, 207,
  143, 143, 143, 207, 207, 143,
  143, 79, 143, 143, 79, 79,
  143, 143, 15, 143, 143, 143,
  79, 79, 143, 143, 15, 191,
  143, 143, 143, 143, 143, 143,
  143, 143, 143, 207, 143, 143,
  143, 143, 143, 207, 207, 143,
  143, 143, 207, 143, 143, 143,
  207, 143, 15, 143, 143, 207,
  143, 143, 143, 143, 15, 79,
  143, 207, 143, 143, 143, 143,
  143, 143, 143, 207, 207, 143,
  143, 143, 79, 15, 207, 143, 79,
  143, 143, 15, 143, 143, 143,
  79, 79, 143, 143, 79, 143, 143,
  143, 143, 143, 79, 79, 143,
  143, 79, 143, 143, 143, 79, 15,
  191, 143, 143, 143, 143, 143,
  143, 143, 207, 143, 143, 143,
  143, 207, 143, 143, 143, 15,
 ...
```

```
[R: LINE 1,811,314]
$wordModel$words
$wordModel$words[[1]]
$wordModel$words[[1]][[1]]
  [1] 143 207 143 143 143 143  15
      143 143  79 143 143  79 143
      143 143 143 143 143  79 143
      143 143 143  15 143 143  79
       15 143
 [31] 143  15 143 143  15 143 143
       79 143 143 143 143 143 143
      143  79 143 143  15 143 143
      143 143 207 143 143 143 207
      143 143
 [61] 207 207  15 143 143 207 143
      143 207 143 143 207 143 143
      143 143  79 143 143 143 143
       79 143 143  79 143 143  15
      143 143
 [91] 143 207 143 143 143 143 143
      143 143 207 143 143 207 207
      143 143 143 207 207 143 143
       79 143 143  79  79 143 143
       15 143
[121] 143 143  79  79 143 143  15
      191 143 143 143 143 143 143
      143 143 143 207 143 143 143
      143 143 207 207 143 143 143
      207 143
[151] 143 143 207 143  15 143 143
      207 143 143 143 143  15  79
      143 207 143 143 143 143 143
      143 143 207 207 143 143 143
       79  15
[181] 207 143  79 143 143  15 143
      143 143  79  79 143 143  79
      143 143 143 143 143  79  79
      143 143  79 143 143 143  79
       15 191
...
```

Features selected for recognition of classes (labels):

```
[Python: LINE 416]
FEATURES:
1 2
2 2
6 1
10 1
11 3
13 1
16 4
18 1
20 1
21 2
36 1
37 3
39 1
41 1
42 2
43 1
44 1
45 2
47 2
63 1
69 1
70 2
71 1
76 5
77 1
82 1
84 1
89 2
93 2
94 2
97 1
99 1
104 3
105 1
109 1
110 1
111 2
 ...
```

```
[R: LINE 1,822,287]
$features
$features[[1]]
$features[[1]][[1]]
$features[[1]][[1]]$index
[1] 1

$features[[1]][[1]]$value
[1] 2


$features[[1]][[2]]
$features[[1]][[2]]$index
[1] 2

$features[[1]][[2]]$value
[1] 2


$features[[1]][[3]]
$features[[1]][[3]]$index
[1] 6

$features[[1]][[3]]$value
[1] 1


$features[[1]][[4]]
$features[[1]][[4]]$index
[1] 10

$features[[1]][[4]]$value
[1] 1


$features[[1]][[5]]
$features[[1]][[5]]$index
[1] 11

$features[[1]][[5]]$value
[1] 3
...
```

Results

Labels assigned to tested time series:

```
correct labels:
[1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 3, 3, 3, 3, 3, 3,
                   4, 4, 4, 4, 4, 4, 5, 5, 5, 5, 5, 5]

[Python: LINE 4,741]
LABELS:
[3, 1, 1, 1, 1, 1, 2, 2, 2, 4, 2, 2, 2, 2, 3, 3, 2, 5,
                   4, 4, 4, 4, 4, 4, 5, 5, 5, 5, 2, 5]

[R: LINE 1,856,701]
[1] 3 1 1 1 1 1 2 2 2 4 2 2 2 2 3 3 2 5 4 4 4 4 4 4 5 5 5 5 2 5
```

Labels were assigned identically in both Python and R implementation. They were correct for 23 out of 30 time series (76.7% accuracy). Running the algorithm with non-fixed permutations can yield higher results. Even higher accuracy can be obtained by increasing the number of windows' lengths.

Additional information

"Beef" data set from 2015 "The UCR Time Series Classification Archive":
Yanping Chen, Eamonn Keogh, Bing Hu, Nurjahan Begum,
Anthony Bagnall, Abdullah Mueen and Gustavo Batista (2015),
`https://www.cs.ucr.edu/~eamonn/time_series_data/`

Original implementation by Patrick Schäfer is available at:
`https://github.com/patrickzib/SFA`

Python implementation by Samuel Harford is available at:
`https://github.com/patrickzib/SFA_Python`