A Rule Extraction and Classification Example

This is an example of the rule extraction process and the subsequent use of the rules to build a classifier. It contains the data, the configuration of the model and the obtained results using 10-fold cross-validation. The classifier is build by placing either a trapezoidal or a triangular membership function for each parameter of the rules, depending if they have an interval or a single value in that parameter. The extracted rules and the associated membership functions are also discussed.

Data

The data used for the experiment is shown below. It is the same data we have been using.

obs	freq	nharm	class
1	0.2	152	1
2	0.2	189	1
3	0.2	425	1
4	0.2	858	1
5	0.3	253	1
6	0.4	136	1
7	0.5	18	1
8	0.5	205	1
9	0.7	618	1
10	0.9	433	1
11	1	41	1
12	1	77	1
13	1	526	1
14	1	949	1
15	2	224	1
16	2	519	1
17	2	813	1
18	2	984	1
19	3	21	1
20	3	33	1
21	3	659	1
22	4	9	1
23	4	19	1
24	4	22	1
25	4	25	1
26	4	26	1
27	4	349	1
28	4	609	1

obs	freq	nharm	class
29	5	14	1
30	5	361	1
31	5	937	1
32	6	7	1
33	6	8	1
34	6	24	1
35	7	12	1
36	7	27	1
37	8	16	1
38	8	48	1
39	8	783	1
40	9	21	1
41	9	342	1
42	9	417	1
43	10	1	1
44	10	16	1
45	10	20	1
46	10	28	1
47	10	88	1
48	11	11	1
49	11	121	1
50	11	274	1
51	12	5	2
52	14	10	1
53	14	13	1
54	14	418	1
55	14	930	1
56	15	5	2
57	15	22	2
58	15	23	1
59	15	58	1
60	16	99	1
61	17	751	2
62	18	5	1
63	18	28	1
64	18	242	1
65	18	576	2
66	19	10	2
67	19	13	2
68	19	275	2
69	19	740	2
70	20	6	2
71	22	9	2
72	23	7	2

obs	freq	nharm	class
73	23	700	2
74	23	803	2
75	27	9	2
76	27	829	2
77	27	874	2
78	28	2	3
79	28	42	2
80	29	451	2
81	30	796	2
82	31	445	2
83	31	610	2
84	32	220	2
85	32	221	2
86	32	289	2
87	34	471	2
88	35	7	2
89	35	674	2
90	36	229	2
91	37	8	2
92	37	27	2
93	37	600	2
94	38	3	2
95	38	394	2
96	38	665	2
97	39	37	2
98	40	874	2
99	41	33	2
100	41	646	2
101	42	3	3
102	42	7	2
103	42	101	2
104	42	149	2
105	42	506	2
106	43	17	2
107	43	455	2
108	44	11	2
109	46	133	2
110	47	335	2
111	47	838	2
112	48	664	2
113	50	15	2
114	50	34	2
115	53	61	2
116	55	10	2

obs	freq	nharm	class
117	56	5	2
118	56	16	2
119	57	38	2
120	57	838	2
121	58	42	2
122	59	1	3
123	65	97	2
124	70	16	2
125	70	35	2
126	71	38	2
127	73	1	3
128	73	10	3
129	73	14	2
130	73	42	2
131	75	4	3
132	78	6	3
133	80	1	3
134	80	17	2
135	81	0	3
136	87	2	3
137	87	13	2
138	90	2	3
139	92	20	2
140	94	12	2
141	95	31	2
142	96	27	2
143	97	0	3
144	98	10	2
145	98	49	2
146	101	3	3
147	104	498	2
148	105	4	3
149	108	233	2
150	109	3	3
151	109	4	3
152	114	9	3
153	116	1	3
154	118	67	2
155	120	47	2
156	127	113	2
157	128	15	3
158	131	368	2
159	133	8	3
160	136	7	3

obs	freq	nharm	class
161	138	487	2
162	145	8	3
163	145	361	2
164	151	356	2
165	156	595	2
166	158	817	2
167	165	687	2
168	168	376	2
169	170	486	2
170	201	834	2
171	202	483	2
172	223	229	2
173	229	607	2
174	255	653	3
175	315	359	3
176	331	4	3
177	331	397	3
178	333	513	3
179	335	17	3
180	377	5	3
181	391	435	3
182	393	173	3
183	446	834	3
184	453	118	3
185	464	4	3
186	471	280	3
187	480	868	3
188	482	31	3
189	493	676	3
190	531	890	3
191	582	4	3
192	586	4	3
193	597	506	3
194	614	3	3
195	614	947	3
196	637	17	3
197	672	2	3
198	751	99	3
199	761	2	3
200	792	766	3
201	806	3	3
202	854	811	3
203	900	8	3
204	944	10	3

obs	freq	nharm	class
205	980	6	3
206	996	755	3

Please note that, although the data were constructed to favor the formation of rules (see, for example, observations 1 to 4). These process, at least at that moment, did not follow any particular methodology.

The idea is to propose a specific process for the exploration process.So far, the current instructions for this step are:

1. Explore the space by using the graphical interface (spyder chart in processing) until you find a desired/interesting combination.
2. From that point tweak (slowly vary) one parameter in turn, getting back to the original point.

-Any other ideas-???

The data was split for 10-Folds cross-validation using the scikit K-Folds cross-validator. The accuracy is calculated by using the sklearn.metrics accuracy_score (both available python libraries).

Parameters

The parameters used for the rule extraction and the classifier are the following: For the rule extraction algorithm: frequency: from 0 to inf, threshold = 1000 number of upper harmonics: form 0 to inf, threshold = 1000

For the classifier: gamma = 2

Results

The individual accuracies for the individual folds are:

[1.0, 1.0, 1.0, 0.952, 0.904, 1.0, 0.949, 0.900, 0.900, 0.949]

And the accuracy of 10-fold cross-validation is: 0.955

Analysis of the Results

Rules

The rules extracted by the rule extraction function on the first training set are the following:

Obtained rules
0 [[8 10] 16 1]
1 [[10 18] 28 1]
2 [[15 56] 5 2]
3 [[19 55] 10 2]
4 [[23 35] 7 2]
5 [[27 40] 874 2]
6 [[28 87 90 672] 2 3]
7 [[28 58] 42 2]
8 [[36 223] 229 2]
9 [[42 101 109 614] 3 3]
10 [[56 70] 16 2]
11 [[59 73 80 116] 1 3]
12 [[73 944] 10 3]
13 [[75 109 331 464 582 586] 4 3]
14 [[81 97] 0 3]
15 [[133 145] 8 3]
16 [[335 637] 17 3]
17 [0.2 [152 189 425 858] 1]
18 [0.5 [18 205] 1]
19 [1 [41 77 526 949] 1]
20 [2 [519 813] 1]
21 [3 [21 33] 1]
22 [4 [9 19 22 25 26 349] 1]
23 [5 [14 937] 1]
24 [6 [8 24] 1]
25 [8 [48 783] 1]
26 [9 [342 417] 1]
27 [10 [20 88] 1]
28 [11 [11 121] 1]
29 [14 [10 13 418 930] 1]
30 [15 [23 58] 1]
31 [19 [13 275] 2]
32 [23 [700 803] 2]
33 [31 [445 610] 2]
34 [32 [220 221] 2]
35 [37 [8 27] 2]
36 [38 [3 394] 2]
37 [41 [33 646] 2]
38 [42 [7 101 149 506] 2]
39 [43 [17 455] 2]
40 [50 [15 34] 2]
41 [73 [14 42] 2]
42 [98 [10 49] 2]
43 [8 [16 48] 1]

Obtained rules
44 [10 [16 20 28 88] 1]
45 [15 [5 22] 2]
46 [18 [28 242] 1]
47 [19 [10 13 275 740] 2]
48 [23 [7 700] 2]
49 [27 [829 874] 2]
50 [35 [7 674] 2]
51 [56 [5 16] 2]
52 [70 [16 35] 2]
53 [73 [1 10] 3]
54 [109 [3 4] 3]
55 [331 [4 397] 3]
56 [614 [3 947] 3]
57 [[42 101] 3 3]
58 [[59 80] 1 3]
59 [[75 464 582 586] 4 3]

Some of the rules can stil be compressed (though not many). – I am currently working on a script intended to this task considering the multi-threshold issue — For example rules 27, 44, 46, and 1 came from different permutations of the data during the compression. Rule 27 is contained in 44 and rule 1 is contained in rule 46.

Membership functions

The trapezoidal membership function for the parameter j or the rule k is given by the equation:

membership_kj(xj,vkj,wkj,gamma):
membership = 1/2 * (max(0,1 - max(0,gamma * min(1,xj-wkj))) + max(0,1 - max(0,gamma * min(1,vkj-xj))))

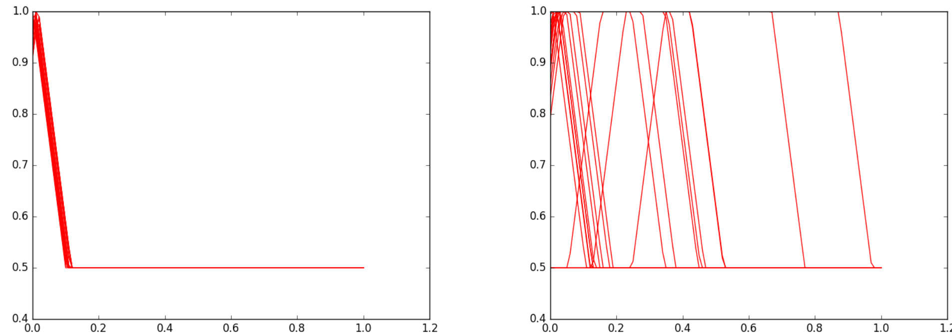
Where xj is the jth value of any input instance. vkj and wkj are the minimum and maximum values of the midium par of the trapezoid in the jth parameter of rule k.

These values are set equal to the min and max of the interval [vkj,. . . , wkj] if the rule has an interval in the jth parameter or vkj = wkj = value, if the rule has a single value at the jth parameter. Finaly, gamma is the parameter that control the spread of the "rim" of the trapezoidal functions.

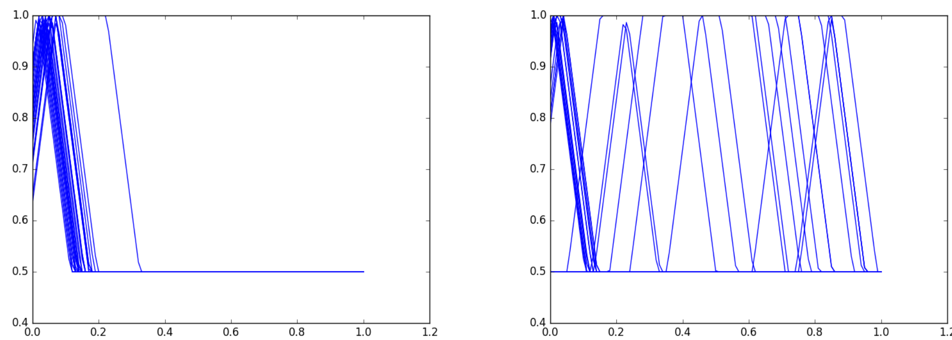
Membership functions plots

Plots for the membership functions of the parameters 1 and 2, for classes rhythmic, rough and tone, are display below. The data and the rules are normalized after the rule extraction process.

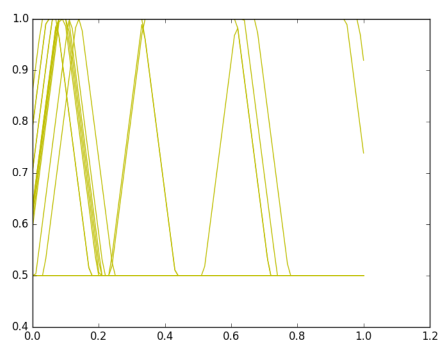
Class 1. Parameters 1 and 2

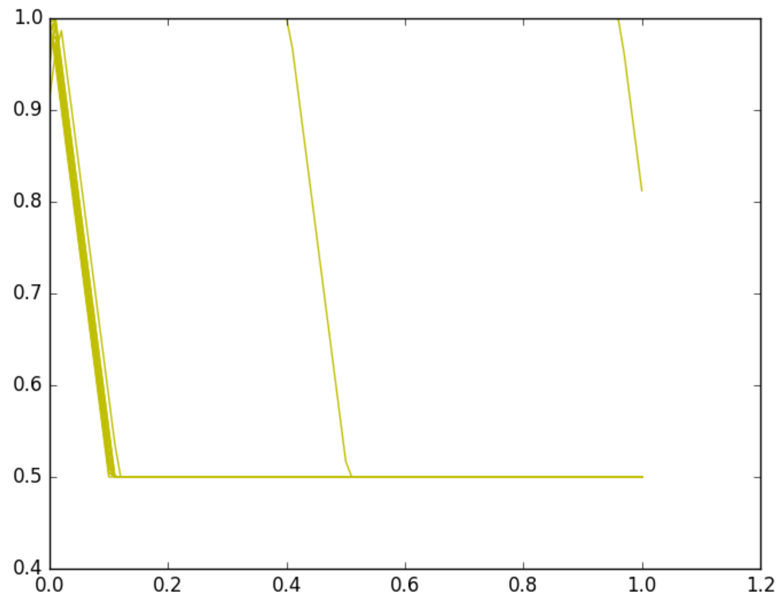


Class2. Parameters 1 and 2.



Class 3. Parameters 1 and 2.




1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.0
0.2
0.4
0.6
0.8
1.0
1.2

The parameter 1 is the frequency and parameter 2 is the number of upper harmonics added to the fundamental frequency. In class 1 we can see that the membership functions of the frequency are well defined in the space of low frequencies. In contrast, in the second parameter (num of harm) the membership functions are disperse. This tow facts could be translated in the expression. If the values of the frequency are from 0 to <= 15 (depending, close to 15, a little bit on the number of hermonics), then no matter what happends with the number of harmonics the perceptual sensation is rhythmic. More or less the same behavior is observed looking at class 2. Here "as long as the frequency is in the rough zone, the number of harmonics do not change the perception. In class 3, which describes combinations perceibed as"tone", it can be observed that combinations with low number of upper harmonics were chosen.

Eliminate Redundant Rules

In the example above rules 27 and 44 are redundant

Rules
27 [10 [20 88] 1]
44 [10 [16 20 28 88] 1]

The "redundant rules" function eliminate those rules. In the example rules 59, 58, 57, 31, and 27 are eliminated with this criteria.

Non redundant rules
0 [[8 10] 16 1]
1 [[10 18] 28 1]
2 [[15 56] 5 2]
3 [[19 55] 10 2]
4 [[23 35] 7 2]
5 [[27 40] 874 2]
6 [[28 87 90 672] 2 3]
7 [[28 58] 42 2]
8 [[36 223] 229 2]
9 [[42 101 109 614] 3 3]
10 [[56 70] 16 2]
11 [[59 73 80 116] 1 3]
12 [[73 944] 10 3]
13 [[75 109 331 464 582 586] 4 3]
14 [[81 97] 0 3]
15 [[133 145] 8 3]
16 [[335 637] 17 3]
17 [0.2 [152 189 425 858] 1]
18 [0.5 [18 205] 1]

Non redundant rules
19 [1 [41 77 526 949] 1]
20 [2 [519 813] 1]
21 [3 [21 33] 1]
22 [4 [9 19 22 25 26 349] 1]
23 [5 [14 937] 1]
24 [6 [8 24] 1]
25 [8 [48 783] 1]
26 [9 [342 417] 1]
27 [11 [11 121] 1]
28 [14 [10 13 418 930] 1]
29 [15 [23 58] 1]
30 [23 [700 803] 2]
31 [31 [445 610] 2]
32 [32 [220 221] 2]
33 [37 [8 27] 2]
34 [38 [3 394] 2]
35 [41 [33 646] 2]
36 [42 [7 101 149 506] 2]
37 [43 [17 455] 2]
38 [50 [15 34] 2]
39 [73 [14 42] 2]
40 [98 [10 49] 2]
41 [8 [16 48] 1]
42 [10 [16 20 28 88] 1]
43 [15 [5 22] 2]
44 [18 [28 242] 1]
45 [19 [10 13 275 740] 2]
46 [23 [7 700] 2]
47 [27 [829 874] 2]
48 [35 [7 674] 2]
49 [56 [5 16] 2]
50 [70 [16 35] 2]
51 [73 [1 10] 3]
52 [109 [3 4] 3]
53 [331 [4 397] 3]
54 [614 [3 947] 3]

Unification of rule intervals

In the table above, intervals of rules 25 and 41 as well as rules 30 and 46 can be unified. In the case of rules 25 and 41

[8, [48, 783], 1] and [8, [16,48], 1] they can be expressed as [8, [16, 783], 1].

With these reductions the new set of rules is:

Rules after unify the possible intervals
0 [[8 10] 16 1]
1 [[10 18] 28 1]
2 [[15 56] 5 2]
3 [[19 55] 10 2]
4 [[23 35] 7 2]
5 [[27 40] 874 2]
6 [[28 87 90 672] 2 3]
7 [[28 58] 42 2]
8 [[36 223] 229 2]
9 [[42 101 109 614] 3 3]
10 [[56 70] 16 2]
11 [[59 73 80 116] 1 3]
12 [[73 944] 10 3]
13 [[75 109 331 464 582 586] 4 3]
14 [[81 97] 0 3]
15 [[133 145] 8 3]
16 [[335 637] 17 3]
17 [0.2 [152 189 425 858] 1]
18 [0.5 [18 205] 1]
19 [1 [41 77 526 949] 1]
20 [2 [519 813] 1]
21 [3 [21 33] 1]
22 [4 [9 19 22 25 26 349] 1]
23 [5 [14 937] 1]
24 [6 [8 24] 1]
25 [9 [342 417] 1]
26 [11 [11 121] 1]
27 [14 [10 13 418 930] 1]
28 [15 [23 58] 1]
29 [31 [445 610] 2]
30 [32 [220 221] 2]
31 [37 [8 27] 2]
32 [38 [3 394] 2]
33 [41 [33 646] 2]
34 [42 [7 101 149 506] 2]
35 [43 [17 455] 2]
36 [50 [15 34] 2]
37 [73 [14 42] 2]
38 [98 [10 49] 2]
39 [10 [16 20 28 88] 1]
40 [15 [5 22] 2]
41 [18 [28 242] 1]

Rules after unify the possible intervals
42 [19 [10 13 275 740] 2]
43 [27 [829 874] 2]
44 [35 [7 674] 2]
45 [56 [5 16] 2]
46 [70 [16 35] 2]
47 [73 [1 10] 3]
48 [109 [3 4] 3]
49 [331 [4 397] 3]
50 [614 [3 947] 3]
51 [8 [16 783] 1]
52 [23 [7 803] 2]