

January 3, 2019

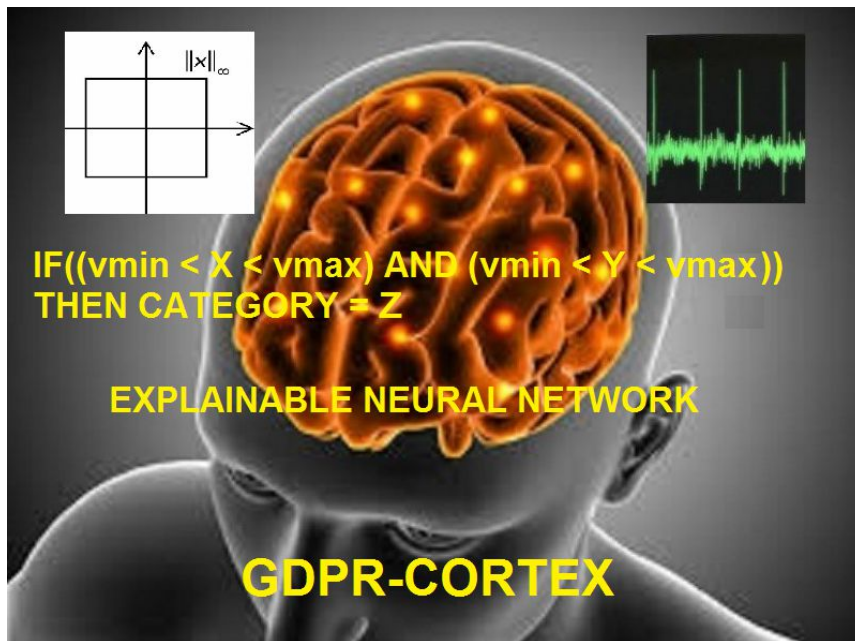
© Luca Marchese luca.marchese@synaptics.org
www.synaptics.org

RULENN

THE RULES BASED NEURAL NETWORK

EXPLAINABLE ARTIFICIAL
INTELLIGENCE

GDPR COMPLIANT NEURAL NETWORK



Copyright © Luca Marchese. ORC-ID: <http://orcid.org/0000-0001-7903-7540>

NCAGE (NATO Commercial And Government Entity Code : AK845

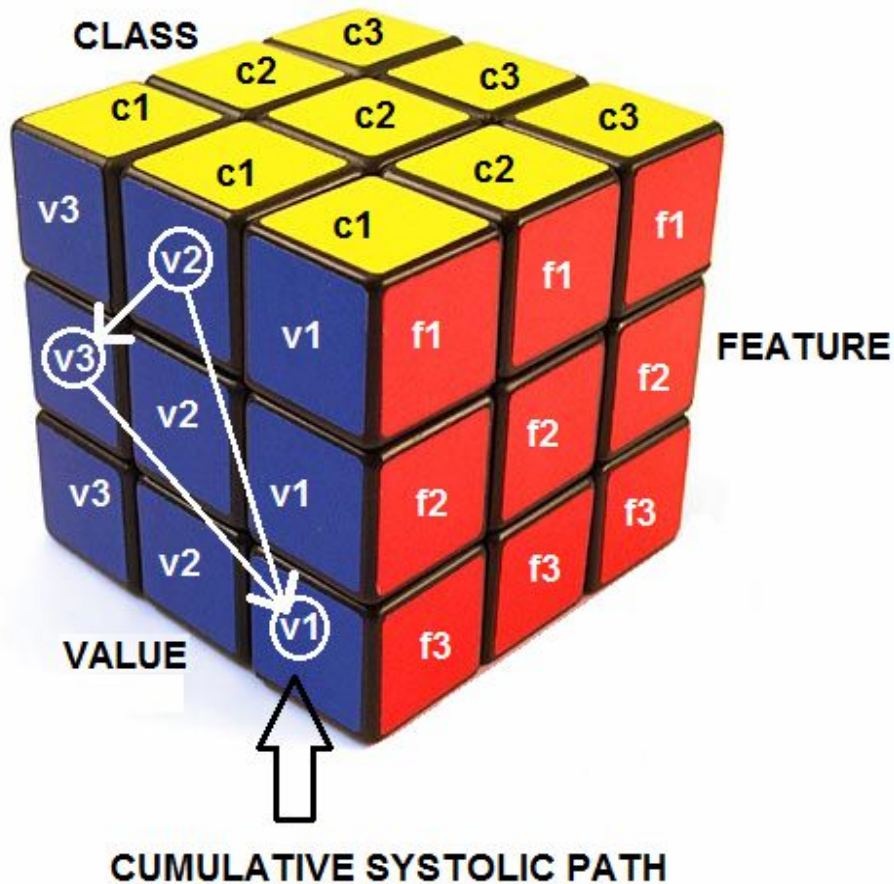
Email: luca.marchese@synaptics.org web: www.synaptics.org

IEEE Senior Member – INNS Senior Member

Copyright © note: the entire document with the copyright note can be freely reproduced, stored and distributed. Parts of the document can be reproduced with the obligation to cite the source.

Based on the SHARP algorithm

Systolic Hebb Agnostic Resonance Perceptron + Automatic Rules Store/Extraction



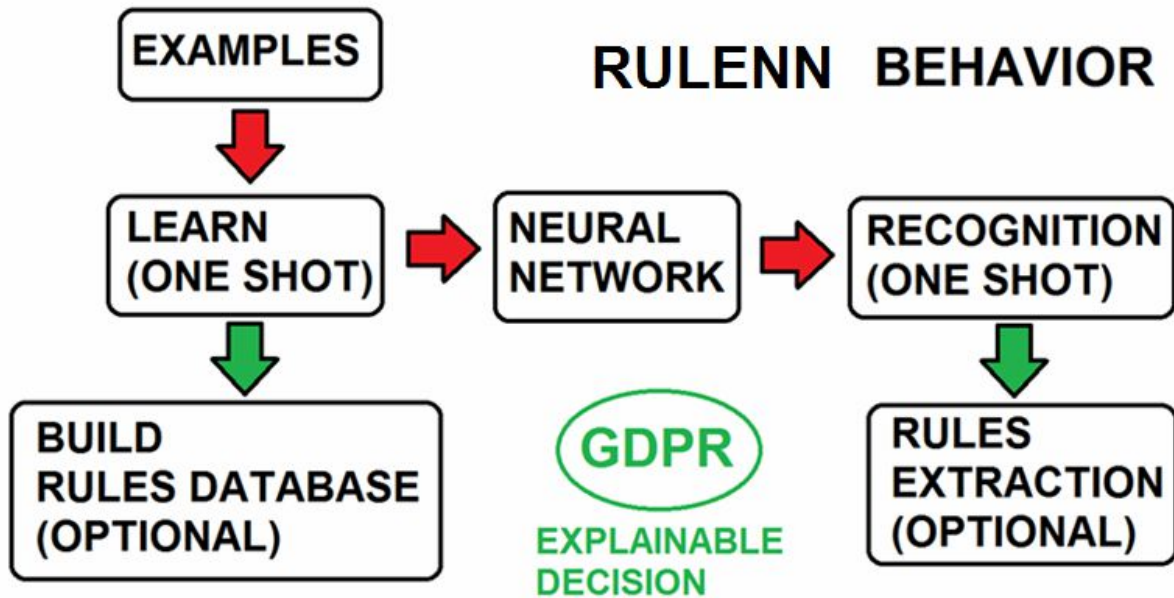
Copyright © Luca Marchese. ORC-ID: <http://orcid.org/0000-0001-7903-7540>

NCAGE (NATO Commercial And Government Entity Code : AK845

Email: luca.marchese@synaptics.org web: www.synaptics.org

IEEE Senior Member – INNS Senior Member

Copyright © note: the entire document with the copyright note can be freely reproduced, stored and distributed. Parts of the document can be reproduced with the obligation to cite the source.



An expert system that learns from the data and builds rules on which the expert user can intervene, by relating plausibility ranges (BOX-Norm) of the individual variables. The range for each variable increases during learning: initial small (also \emptyset) ranges are suitable working with BIG_DATA while initial larger ranges are suitable working with SMALL-DATA. The rules of the expert system are not evaluated serially as in a Von

Copyright © Luca Marchese. ORCID-ID: <http://orcid.org/0000-0001-7903-7540>

NCAGE (NATO Commercial And Government Entity Code : AK845

Email: luca.marchese@synaptics.org web: www.synaptics.org

IEEE Senior Member – INNS Senior Member

Copyright © note: the entire document with the copyright note can be freely reproduced, stored and distributed. Parts of the document can be reproduced with the obligation to cite the source.

Neumann machine but are mapped onto a Neural Net (SHARP) that learns and infers in a single shot. If the "GDPR" flag is set to "ON" during learning and during the inference, the network stores/extracts the rules that determined the decision (s) from the synapses. During learning there is no loss of performance if the "GDPR" flag is set to "ON" while the inference on the single data is much faster if the "GDPR" flag is set to "OFF".

Actually, the system is set up to acquire large amounts of data quickly by building a database of rules (GDPR ON) and inferring equally quickly on large amounts of data without giving an explanation of the applied rule (GDPR OFF). When it is necessary to provide an explanation on one or more decisions, one can perform an inference with the GDPR ON flag, losing speed performance but explicating all the rules (if-then) involved in the decision.

Copyright © Luca Marchese. ORCID-ID: <http://orcid.org/0000-0001-7903-7540>

NCAGE (NATO Commercial And Government Entity Code) : AK845

Email: luca.marchese@synaptics.org web: www.synaptics.org

IEEE Senior Member – INNS Senior Member

Copyright © note: the entire document with the copyright note can be freely reproduced, stored and distributed. Parts of the document can be reproduced with the obligation to cite the source.

API USER MANUAL

RULENN Rel 1.1 Notes

This software is an optimized digital emulation of the S.H.A.R.P neural network. This software works on RAM and synapses are saved/loaded on mass storage with apposite functions. This simple software is composed of six functions performing the following operations:

- 1) Initialization
- 2) Learning
- 3) Recognition
- 4) Synaptic saving
- 5) Synaptic loading
- 6) Report

Initialization

```
//-----//  
// Initializing the RULENN architecture //  
//-----//  
unsigned short int RULENN_INIT(const unsigned char pattern_comps);
```

PARAMETERS:

pattern_comps: (INPUT) the number of components of the input vector

This function must be called only one time when it is required to set the number of components of the input vector and initialize the synaptic values. When the network is uploaded with the RULENN_LOAD function (see the next part of the document) the function RULENN_INIT must not be called.

Copyright © Luca Marchese. ORC-ID: <http://orcid.org/0000-0001-7903-7540>

NCAGE (NATO Commercial And Government Entity Code : AK845

Email: luca.marchese@synaptics.org web: www.synaptics.org

IEEE Senior Member – INNS Senior Memeber

Copyright © note: the entire document with the copyright note can be freely reproduced, stored and distributed. Parts of the document can be reproduced with the obligation to cite the source.

Learning

```
//-----//  
// Pattern learning operation //  
//-----//  
unsigned short int RULENN_LEARN(const unsigned int build_rule_flag,  
                                const char *db_rule_file,  
                                const unsigned char *pattern,  
                                const unsigned short int cat,  
                                const unsigned char *nif,  
                                const unsigned char thresh,  
                                unsigned char *rec);
```

PARAMETERS:

build_rule_flag: (INPUT) 0 = rules not saved; 1 = rules saved;

db_rule_file: (INPUT) the name of the file where rules are written if build_rule_flag = 1

pattern: (INPUT) the vector to learn (vector size: pattern_comps); component value range: 0-128

cat: (INPUT) category assigned to the pattern; value range: 1-8

nif: (INPUT) the vector of L-SUP generalization required (vector size: pattern_comps); value range: 0-128 (maximum value is actually limited by #define MAX_NIF 50). See Fig.1.

thresh: (INPUT) the threshold value of the recognition strength used to check if the pattern should be considered already recognized (*rec= 1 and no learning operation is performed); range: 0-255 (the pattern is learned if recognized with strength < thresh)

rec: (OUTPUT) when *rec = 1 then the pattern is already recognized and no learning operation is required; when *rec = 0 then the pattern has been learned.

Copyright © Luca Marchese. ORC-ID: <http://orcid.org/0000-0001-7903-7540>

NCAGE (NATO Commercial And Government Entity Code : AK845

Email: luca.marchese@synaptics.org web: www.synaptics.org

IEEE Senior Member – INNS Senior Memeber

Copyright © note: the entire document with the copyright note can be freely reproduced, stored and distributed. Parts of the document can be reproduced with the obligation to cite the source.

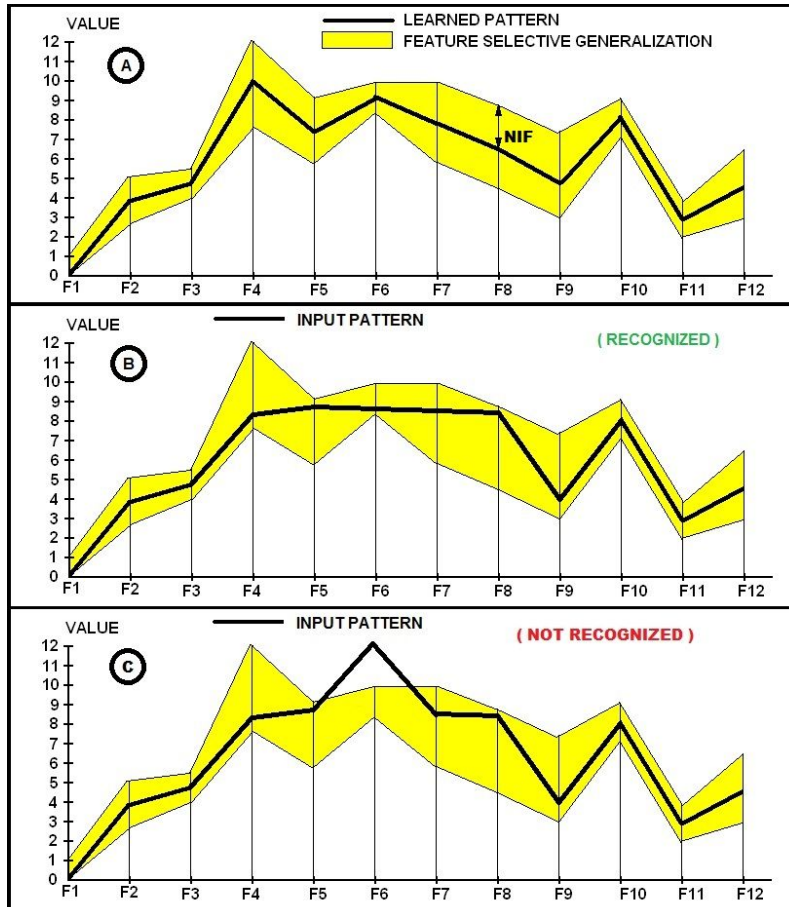


Fig.1 BOX GENERALIZATION WITH SPECIFIC VALUES FOR EACH FEATURE

Recognition

```
//-----//
// Pattern recognition operation //
//-----//
unsigned short int RULENN_RECOGNIZE(const unsigned int extract_rule_flag,
                                   const char *db_rule_file,
                                   const unsigned char *pattern,
                                   unsigned short int *cat_num,
                                   unsigned short int *cat,
                                   unsigned char *recognition_strength);
```

Copyright © Luca Marchese. ORC-ID: <http://orcid.org/0000-0001-7903-7540>

NCAGE (NATO Commercial And Government Entity Code : AK845

Email: luca.marchese@synaptics.org web: www.synaptics.org

IEEE Senior Member – INNS Senior Memeber

Copyright © note: the entire document with the copyright note can be freely reproduced, stored and distributed. Parts of the document can be reproduced with the obligation to cite the source.

PARAMETERS:

extract_rule_flag: (INPUT) 0 = rules not extracted (faster) ; 1 = rules extracted;

db_rule_file: (INPUT) is the name of the file where the rules are written if the extract_rule_flag = 1

pattern: (INPUT) the vector to learn (vector size: pattern_comps); component value range: 0-128

cat_num: (OUTPUT) number of categories where the pattern has been recognized

cat: (OUTPUT) the vector of categories where the pattern has been recognized ordered by recognition strength

recognition_strength: (OUTPUT) the vector of the recognition strength values associated to the categories
range: 0-255

Synaptic saving

```
//-----//  
// Saving a RULENN trained architecture on binary file //  
//-----//  
unsigned short int RULENN_SAVE(const char *filename);
```

This function saves the RULENN NN with all the synaptic values in a file.

filename: (INPUT) the name of the file where to save the network

Synaptic loading

```
//-----//  
// Loading a RULENN trained architecture from binary file //  
//-----//  
unsigned short int RULENN_LOAD(const char *filename);
```

This function loads the RULENN NN with all the synaptic values from a file.

filename: (INPUT) the name of the file where to save the network

Report

Copyright © Luca Marchese. ORC-ID: <http://orcid.org/0000-0001-7903-7540>

NCAGE (NATO Commercial And Government Entity Code : AK845

Email: luca.marchese@synaptics.org web: www.synaptics.org

IEEE Senior Member – INNS Senior Memeber

Copyright © note: the entire document with the copyright note can be freely reproduced, stored and distributed. Parts of the document can be reproduced with the obligation to cite the source.


```
//-----  
// Saving a RULENN info report  
//-----  
unsigned short int RULENN_REPORT(void);
```

This function generates a report file containing information on the RULENN NN after initialization and learning:

Example:

FEATURES: 64

CATEGORIES: 1 2 3 4 5 6 7 8

CLUSTER NEURONS: 8

FEATURES is the number of features used for the initialization of the network

CATEGORIES is the number of categories that have been utilized in the training set

CLUSTER NEURONS is the number of neurons used as intermediate layer between the mini-columns neurons and the category neurons. This feature is not mandatory in the RULENN neural network but has been inserted in this software implementation for future developments on mechanisms aimed to limit crosstalk phenomena. The role of the intermediate neurons (called also “category cluster neurons”) is limiting the number of connections with a single category neuron. Each “category cluster neuron” can learn a maximum number of patterns. When, during the learning task, this limit is reached a new “cluster neuron” associated with the category is committed. If the number of cluster neurons is equal to the number of categories, then each category is represented by a single cluster neuron. If the number of category cluster neurons is greater than the number of categories, some categories could be associated with more cluster neurons.

Copyright © Luca Marchese. ORC-ID: <http://orcid.org/0000-0001-7903-7540>

NCAGE (NATO Commercial And Government Entity Code : AK845

Email: luca.marchese@synaptics.org web: www.synaptics.org

IEEE Senior Member – INNS Senior Member

Copyright © note: the entire document with the copyright note can be freely reproduced, stored and distributed. Parts of the document can be reproduced with the obligation to cite the source.

The possible return codes of the functions are:

```
#define RULENN_NO_ERROR 0
#define RULENN_OPEN_FILE 1
#define RULENN_READ_FILE 2
#define RULENN_WRITE_FILE 3
#define RULENN_INV_PAR 4
```

EXAMPLE OF USE

```
//initialization of the network for a task with 7 features
unsigned short int i_RULENN_Ret;
const unsigned char pattern_comps = 7;
i_RULENN_Ret = RULENN_INIT(pattern_comps);
if(!i_RULENN_Ret) { RULENN_process_error(i_RULENN_Ret); return i_RULENN_Ret; }

//learning
unsigned short int i_RULENN_Ret;
unsigned short int i_READ_Ret;
const unsigned char pattern[7];
const unsigned short int cat;
const unsigned char nif[8];
const unsigned char thresh;
unsigned char rec;
i_READ_Ret = read_example (pattern, nif, cat, thresh);
if(!i_READ_Ret) { READ_process_error(i_READ_Ret); return i_READ_Ret; }
i_RULENN_Ret = RULENN_LEARN (1,"rule_db.txt",pattern, cat, nif, thresh, &rec);
if(!i_RULENN_Ret) { RULENN_process_error(i_RULENN_Ret); return i_RULENN_Ret; }
if(!rec) { printf("the pattern has been learned\n"); }
else { printf("the pattern has been recognized\n"); }
```

the example file could contain a record like this:

```
123,010, 020,010,040,005,098,015,100,015,070,010,030,010,200,5
```

The meaning of the fields is:

```
VAL[0],LSUP[0],VAL[1],LSUP[1],VAL[2],LSUP[2],VAL[3],LSUP[3],VAL[4],LSUP[4],VAL[5],LSUP[5],VAL[6],LSUP[6],THRESH,CAT
```

Copyright © Luca Marchese. ORC-ID: <http://orcid.org/0000-0001-7903-7540>

NCAGE (NATO Commercial And Government Entity Code : AK845

Email: luca.marchese@synaptics.org web: www.synaptics.org

IEEE Senior Member – INNS Senior Memeber

Copyright © note: the entire document with the copyright note can be freely reproduced, stored and distributed. Parts of the document can be reproduced with the obligation to cite the source.

The record contains the values of 7 features. For each feature a BOX (LSUP) generalization value is set. If the pattern is recognized with a recognition strength >200 (thresh) in the category 5 (cat) then the pattern is not learned (it is considered already part of the know-how of the network). If the recognition strength is < 200 (thresh) or the recognized cat is different from 5 (cat) then the pattern is learned and associated with the category 5.

```
//recognition of a 7 components input pattern
#define MAX_CAT_NUMBER 8 //it must be <= MAX_CAT in RULENN.c
unsigned short int i_RULENN_Ret;
unsigned short int i_READ_Ret;
unsigned short int i_RESULT_Ret;

const unsigned char pattern[7];
unsigned short int cat_num;
unsigned short int cat[CAT_NUMBER];
unsigned char recognition_strength[CAT_NUMBER];
i_READ_Ret = read_input (pattern);
if(!i_READ_Ret) { READ_process_error(i_READ_Ret); return i_READ_Ret; }
i_RULENN_Ret = RULENN_RECOGNIZE(1, "rules.txt", pattern, &cat_num, cat, recognition_strength);
if(!i_RULENN_Ret) { RULENN_process_error(i_RULENN_Ret); return i_RULENN_Ret; }
i_RESULT_Ret = read_result(cat_num, cat, recognition_strength);
if(!i_RESULT_Ret) { RESULT_process_error(i_RESULT_Ret); return i_RESULT_Ret; }
```

the read_input function could read from a file containing records like this:

123,020,040,098,100,070,030

The meaning of the fields is:

VAL[0],VAL[1],VAL[2],VAL[3],VAL[4],VAL[5],VAL[6]

the read_result function should read the number of categories (cat_num) and then scan cat_num elements of the vectors cat[] and recognition_strength[]. These vectors contain respectively the category number and the associated recognition strength. The vectors are already sorted with decreasing recognition strength. The first element is the strongest recognition. There could be an uncertainty if more elements have the same recognition strength of the first element.

An example of result could be:

cat_num = 3

Copyright © Luca Marchese. ORC-ID: <http://orcid.org/0000-0001-7903-7540>

NCAGE (NATO Commercial And Government Entity Code : AK845

Email: luca.marchese@synaptics.org web: www.synaptics.org

IEEE Senior Member – INNS Senior Member

Copyright © note: the entire document with the copyright note can be freely reproduced, stored and distributed. Parts of the document can be reproduced with the obligation to cite the source.

```
cat[] = 7, 4, 5, 0, 0, 0, 0, 0  
recognition_strength[] = 200, 80, 60
```

The user can consider only the first elements having the same recognition strength (if only the first element has the higher recognition strength value then the pattern is recognized without uncertainty and this is the most frequent case).

The user can also work with the fuzzy classification on multiple categories (Fig.2). By using the vector of recognition strengths it is possible to build a deep learning architecture where the output vectors (recognition strengths) of multiple RULENN modules in one layer are sent together as input of one or more RULENN modules in the following layer.

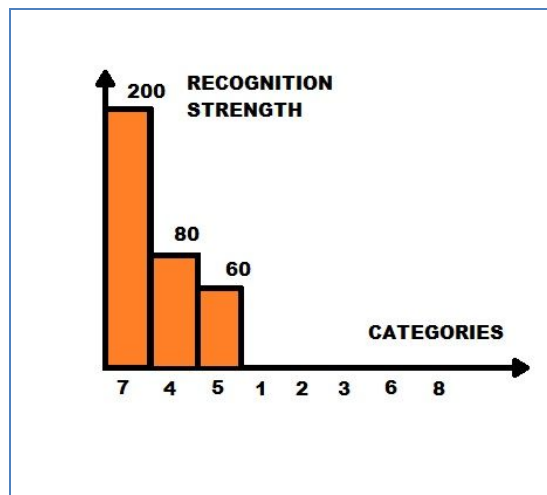


Fig.2 RECOGNITION STRENGTH VECTOR

HOW TO WORK WITH A LARGER NUMBER OF CATEGORIES

With this simple implementation only 8 categories can be utilized. By using multiple RULENN module, it is possible to manage a larger number of categories. The input pattern must be multiplexed to all the RULENN modules and the final category is composed by the ensemble of the single categories (used as 0-8 digits). Fig.3 could explain better the simple concept of composite categorization. With 3 networks you can manage 8×3 categories. Fig.4B explains the same concept of Fig.3. During the learning task, an input pattern can be assigned to the category 768. The first network learns the pattern as assigned to category 7, the second

Copyright © Luca Marchese. ORC-ID: <http://orcid.org/0000-0001-7903-7540>

NCAGE (NATO Commercial And Government Entity Code : AK845

Email: luca.marchese@synaptics.org web: www.synaptics.org

IEEE Senior Member – INNS Senior Memeber

Copyright © note: the entire document with the copyright note can be freely reproduced, stored and distributed. Parts of the document can be reproduced with the obligation to cite the source.

network learns the pattern as assigned to category 6 and the third network learns the pattern as assigned to category 8. The value cannot be used. When a recognition process produces a category with one or more digit set to 0, the input pattern is not recognized.

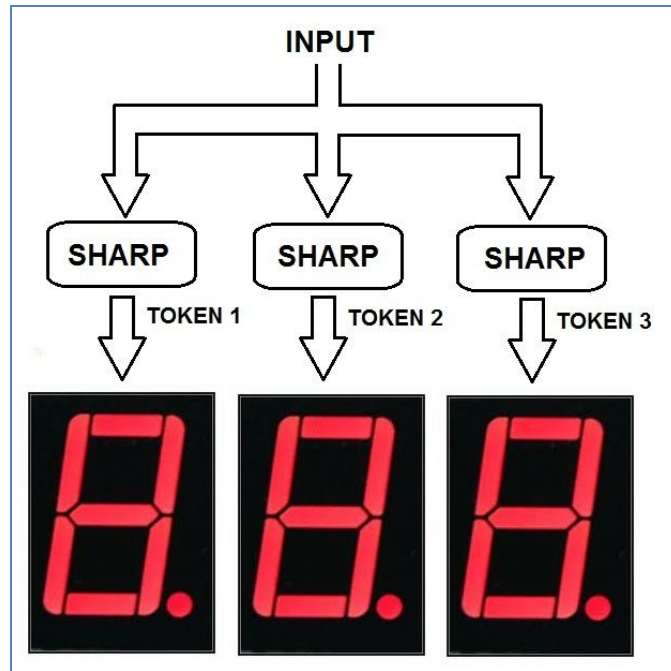


Fig.3 COMPOSITE CATEGORY EXAMPLE: 8**3 CATEGORIES

HOW TO WORK WITH LARGER INPUT PATTERNS

With this simple implementation it is possible to use patterns composed of 64 features. If your application needs to manage larger patterns, you can use multiple RULENN modules inputting different tokens of the input to different modules (Fig.4 A).

Copyright © Luca Marchese. ORC-ID: <http://orcid.org/0000-0001-7903-7540>

NCAGE (NATO Commercial And Government Entity Code : AK845

Email: luca.marchese@synaptics.org web: www.synaptics.org

IEEE Senior Member – INNS Senior Memeber

Copyright © note: the entire document with the copyright note can be freely reproduced, stored and distributed. Parts of the document can be reproduced with the obligation to cite the source.

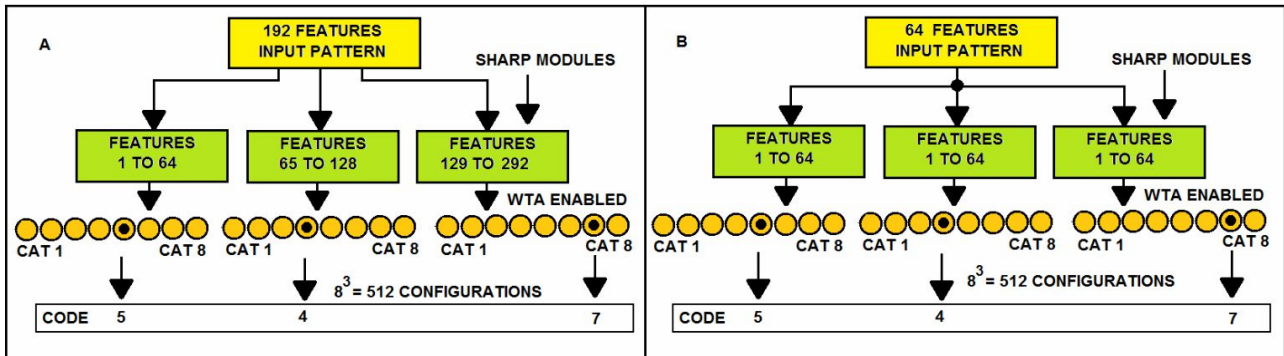


Fig.4 A) A LARGE INPUT PATTERN IS MULTIPLEXED ON THREE RULENN MODULES. WTA (Winner Takes All) ENABLED MEANS (FOR THE API USER) THAT ONLY THE WINNING CATEGORY IS CONSIDERED AND NOT THE VECTOR OF RECOGNITION STRENGTHS. B) THIS SCHEME SHOWS THE SAME CONCEPT OF Fig.3.

Larger input patterns can be managed also with a DEEP-LEARNING approach by using multiple layers of RULENN modules. The recognition strengths vectors that are output of the layer N are used as the input of the layer N+1. The concept is shown in Fig.5. This approach becomes particularly interesting when working with BIG DATA: when the crosstalk between categories becomes important, the recognition strengths vectors of the first layers are more complex (few 0 values are present / more categories activated).

Copyright © Luca Marchese. ORC-ID: <http://orcid.org/0000-0001-7903-7540>

NCAGE (NATO Commercial And Government Entity Code : AK845

Email: luca.marchese@synaptics.org web: www.synaptics.org

IEEE Senior Member – INNS Senior Memeber

Copyright © note: the entire document with the copyright note can be freely reproduced, stored and distributed. Parts of the document can be reproduced with the obligation to cite the source.

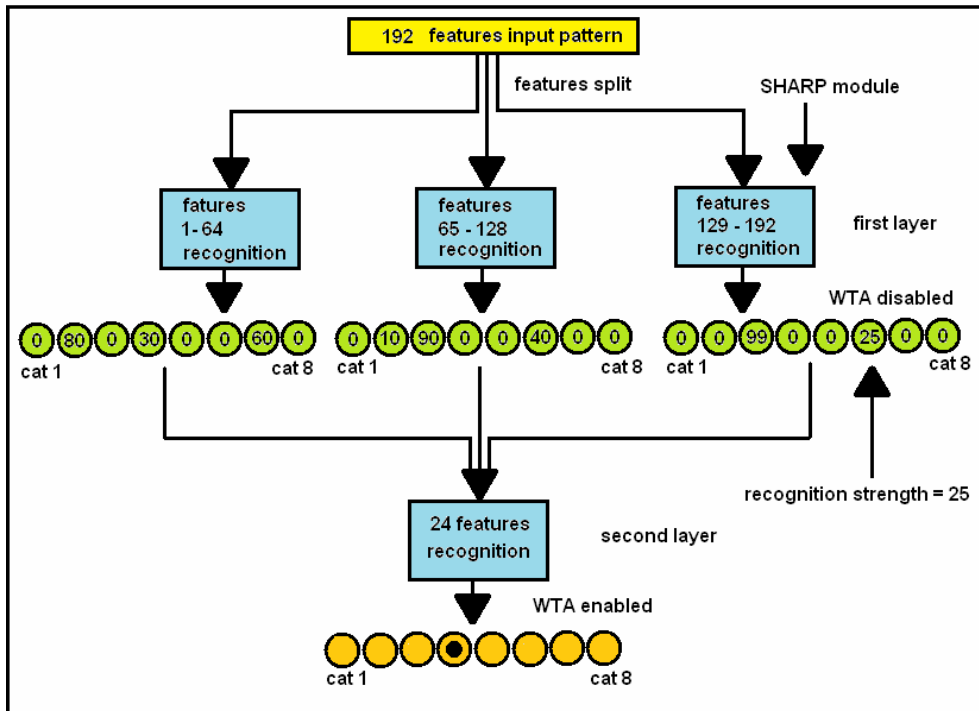


Fig.5 AN INPUT VECTOR OF 192 FEATURES IS MULTIPLEXED TO THREE RULENN MODULES. THE OUTPUT VECTORS OF RECOGNITION STRENGTHS ARE USED AS INPUT FOR THE RULENN MODULE IN THE SECOND LAYER. WTA (Winner Takes All) ENABLED MEANS (FOR THE API USER) THAT ONLY THE WINNING CATEGORY IS CONSIDERED AND NOT THE VECTOR OF RECOGNITION STRENGTHS. WTA (Winner Takes All) DISABLED MEANS (FOR THE API USER) THAT THE VECTOR OF RECOGNITION STRENGTHS IS CONSIDERED AS OUTPUT.

Copyright © Luca Marchese. ORC-ID: <http://orcid.org/0000-0001-7903-7540>

NCAGE (NATO Commercial And Government Entity Code : AK845

Email: luca.marchese@synaptics.org web: www.synaptics.org

IEEE Senior Member – INNS Senior Member

Copyright © note: the entire document with the copyright note can be freely reproduced, stored and distributed. Parts of the document can be reproduced with the obligation to cite the source.

USING THE TESTING SOFTWARE

The test software provided with the RULENN code is a simple console application containing functions that generate artificial databases of examples for training and validation.

```
-----
1) TOY_TEST
2) EXTENSIVE_TEST
3) EXIT
-----
choice:
```

By selecting the TOY_TEST the following menu is shown:

```
-----
1) INIT
2) LEARN <5 patterns>
3) RECON <5 noisy patterns + 1 extr>
4) SAVE
5) LOAD
6) EXIT
-----
choice:
```

There are 5 patterns hard-coded inside the C-source and 5 related noisy patterns. One extra pattern is completely different.

Try INIT => LEARN => RECON:

```
choice:3
pattern_1: cat_num = 1 cat = 1 strength = 250
pattern_2: cat_num = 1 cat = 2 strength = 251
pattern_3: cat_num = 1 cat = 3 strength = 252
pattern_4: cat_num = 2 cat = 4 strength = 253
pattern_4: cat_num = 2 cat = 2 strength = 251
pattern_5: cat_num = 1 cat = 5 strength = 251
pattern_6: NOT IDENTIFIED
```

Then try SAVE (the network with synapses is saved on the binary file "RULENN.snps")

Then try INIT=>RECON:

Copyright © Luca Marchese. ORC-ID: <http://orcid.org/0000-0001-7903-7540>

NCAGE (NATO Commercial And Government Entity Code : AK845

Email: luca.marchese@synaptics.org web: www.synaptics.org

IEEE Senior Member – INNS Senior Memeber

Copyright © note: the entire document with the copyright note can be freely reproduced, stored and distributed. Parts of the document can be reproduced with the obligation to cite the source.

January 3, 2019

© Luca Marchese luca.marchese@synaptics.org
www.synaptics.org

```
choice:3
pattern_1: NOT IDENTIFIED
pattern_2: NOT IDENTIFIED
pattern_3: NOT IDENTIFIED
pattern_4: NOT IDENTIFIED
pattern_5: NOT IDENTIFIED
pattern_6: NOT IDENTIFIED
```

(the INIT operation deleted all the synapses)

Try LOAD (the network with synapses previously saved is reloaded)

Try RECON:

```
choice:3
pattern_1: cat_num = 1 cat = 1 strength = 250
pattern_2: cat_num = 1 cat = 2 strength = 251
pattern_3: cat_num = 1 cat = 3 strength = 252
pattern_4: cat_num = 2 cat = 4 strength = 253
pattern_4: cat_num = 2 cat = 2 strength = 251
pattern_5: cat_num = 1 cat = 5 strength = 251
pattern_6: NOT IDENTIFIED
```

NOTE: in this toy problem the GDPR flag is set ON by default and a “rules_db.bin” binary file is created during learning. During the recognition process the GDPR flag is ON by default and the files rules_ex1.txt, rules_ex2.txt, rules_ex3.txt, rules_ex4.txt and rules_ex5.txt are created. They contain the rules that explain the inference of the neural network.

Following the contents of the rules files:

rules_ex1.txt:

```
IF (029 < VAR[00]=039 < 039) AND
  (095 < VAR[01]=100 < 105) AND
  (045 < VAR[02]=050 < 055) AND
  (035 < VAR[03]=040 < 045) AND
  (025 < VAR[04]=030 < 035) AND
  (120 < VAR[05]=125 < 127) AND
  (075 < VAR[06]=080 < 085) AND
  (005 < VAR[07]=010 < 015) AND
  (095 < VAR[08]=100 < 105) AND
  (095 < VAR[09]=100 < 105)
THEN CATEGORY=01
```

rules_ex2.txt:

```
IF (095 < VAR[00]=104 < 105) AND
  (015 < VAR[01]=024 < 025) AND
  (025 < VAR[02]=034 < 035) AND
```

Copyright © Luca Marchese. ORC-ID: <http://orcid.org/0000-0001-7903-7540>

NCAGE (NATO Commercial And Government Entity Code : AK845

Email: luca.marchese@synaptics.org web: www.synaptics.org

IEEE Senior Member – INNS Senior Memeber

Copyright © note: the entire document with the copyright note can be freely reproduced, stored and distributed. Parts of the document can be reproduced with the obligation to cite the source.

January 3, 2019

© Luca Marchese luca.marchese@synaptics.org
www.synaptics.org

```
(085 < VAR[03]=094 < 095) AND  
(062 < VAR[04]=063 < 072) AND  
(095 < VAR[05]=096 < 105) AND  
(095 < VAR[06]=104 < 105) AND  
(095 < VAR[07]=104 < 105) AND  
(095 < VAR[08]=096 < 105) AND  
(094 < VAR[09]=096 < 106)  
THEN CATEGORY=02
```

rules_ex3:

```
IF (005 < VAR[00]=012 < 015) AND  
(015 < VAR[01]=022 < 025) AND  
(025 < VAR[02]=032 < 035) AND  
(035 < VAR[03]=038 < 045) AND  
(045 < VAR[04]=052 < 055) AND  
(055 < VAR[05]=058 < 065) AND  
(065 < VAR[06]=068 < 075) AND  
(075 < VAR[07]=082 < 085) AND  
(085 < VAR[08]=093 < 095) AND  
(093 < VAR[09]=102 < 107)  
THEN CATEGORY=03
```

rules_ex4:

```
IF (095 < VAR[00]=097 < 105) AND  
(015 < VAR[01]=024 < 025) AND  
(025 < VAR[02]=028 < 035) AND  
(085 < VAR[03]=094 < 095) AND  
(062 < VAR[04]=064 < 072) AND  
(095 < VAR[05]=104 < 105) AND  
(095 < VAR[06]=104 < 105) AND  
(095 < VAR[07]=104 < 105) AND  
(095 < VAR[08]=104 < 105) AND  
(094 < VAR[09]=096 < 106)  
THEN CATEGORY=02
```

```
IF (091 < VAR[00]=097 < 101) AND  
(021 < VAR[01]=024 < 031) AND
```

Copyright © Luca Marchese. ORC-ID: <http://orcid.org/0000-0001-7903-7540>

NCAGE (NATO Commercial And Government Entity Code : AK845

Email: luca.marchese@synaptics.org web: www.synaptics.org

IEEE Senior Member – INNS Senior Memeber

Copyright © note: the entire document with the copyright note can be freely reproduced, stored and distributed. Parts of the document can be reproduced with the obligation to cite the source.

January 3, 2019

© Luca Marchese luca.marchese@synaptics.org
www.synaptics.org

```
(022 < VAR[02]=028 < 032) AND  
(091 < VAR[03]=094 < 101) AND  
(057 < VAR[04]=064 < 067) AND  
(101 < VAR[05]=104 < 111) AND  
(101 < VAR[06]=104 < 111) AND  
(101 < VAR[07]=104 < 111) AND  
(101 < VAR[08]=104 < 111) AND  
(086 < VAR[09]=096 < 102)  
THEN CATEGORY=04
```

rules_ex5:

```
IF (025 < VAR[00]=026 < 035) AND  
(085 < VAR[01]=091 < 095) AND  
(025 < VAR[02]=031 < 035) AND  
(085 < VAR[03]=089 < 095) AND  
(025 < VAR[04]=026 < 035) AND  
(085 < VAR[05]=094 < 095) AND  
(025 < VAR[06]=034 < 035) AND  
(085 < VAR[07]=092 < 095) AND  
(025 < VAR[08]=034 < 035) AND  
(081 < VAR[09]=094 < 099)  
THEN CATEGORY=05
```

Then EXIT => EXTENSIVE_TEST:

Copyright © Luca Marchese. ORC-ID: <http://orcid.org/0000-0001-7903-7540>

NCAGE (NATO Commercial And Government Entity Code : AK845

Email: luca.marchese@synaptics.org web: www.synaptics.org

IEEE Senior Member – INNS Senior Memeber

Copyright © note: the entire document with the copyright note can be freely reproduced, stored and distributed. Parts of the document can be reproduced with the obligation to cite the source.

```
| 0) DATAGEN RANDOM (8CAT/64COMP)
| 1) DATAGEN XOR (2CAT/2COMP)
| 2) DATAGEN CIRCLE_IN_THE_SQUARE(2CAT/2COMP)
| 3) DATAGEN COMPLEX (8CAT/64COMP)
| 4) INITIALIZE NEURAL NETWORK
| 5) READ CURRENT DATASET TYPE
| 6) SET EXAMPLES NUM (DATAGEN 0,2,3)
| 7) LEARN
| 8) RECALL
| 9) RECOGNITION
| 10) SAVE NEURAL NETWORK SYNAPSES
| 11) LOAD NEURAL NETWORK SYNAPSES
| 12) REPORT
| 13) GDPR ON (STORE/EXTRACT RULES)
| 14) GDPR OFF (FASTER INFERENCE)(DEFAULT)
| 15) EXIT
```

choice:

The option 0 generates the text file "RULENN_ext_test.dat" that contains records for both learning and validation. The number of examples can be (previously) selected with the option SET EXAMPLES NUM (default=30,000).

The program will ask for the NIF (the BOX generalization) for each component. If the size of the pattern is 64 you will have to put a value of NIF for each component as shown in the following screen snapshot. When the process will be terminated, a file called "RULENN_ext_test.dat" will be created.

Copyright © Luca Marchese. ORCID-ID: <http://orcid.org/0000-0001-7903-7540>

NCAGE (NATO Commercial And Government Entity Code : AK845

Email: luca.marchese@synaptics.org web: www.synaptics.org

IEEE Senior Member – INNS Senior Memeber

Copyright © note: the entire document with the copyright note can be freely reproduced, stored and distributed. Parts of the document can be reproduced with the obligation to cite the source.

```

| 0) DATAGEN RANDOM (8CAT/64COMP)
| 1) DATAGEN XOR (2CAT/2COMP)
| 2) DATAGEN CIRCLE_IN_THE_SQUARE(2CAT/2COMP)
| 3) DATAGEN COMPLEX (8CAT/64COMP)
| 4) INITIALIZE NEURAL NETWORK
| 5) READ CURRENT DATASET TYPE
| 6) SET EXAMPLES NUM (DATAGEN 0,2,3)
| 7) LEARN
| 8) RECALL
| 9) RECOGNITION
| 10) SAVE NEURAL NETWORK SYNAPSES
| 11) LOAD NEURAL NETWORK SYNAPSES
| 12) REPORT
| 13) GDPR ON (STORE/EXTRACT RULES)
| 14) GDPR OFF (FASTER INFERENCE)(DEFAULT)
| 15) EXIT

```

```

choice:0
NIF[000](0-50):5
NIF[001](0-50):10
NIF[002](0-50):10
NIF[003](0-50):15
NIF[004](0-50):10
NIF[005](0-50):5

```

Continue up to NIF[063] selecting the desired range for any variable

The text file "RULENN_ext_test.dat" contains records of this type:

```

ACO 039(010) 015(010) 000(010) 090(010) 070(010) 087(005) 053(005) 034(005)
118(005) 024(005) 044(005) 117(005) 105(005) 065(005) 124(005) 107(005)
105(005) 006(005) 042(005) 109(005) 065(005) 046(005) 056(005) 109(005)
008(005) 023(005) 117(005) 088(005) 102(005) 030(005) 048(005) 091(005)
092(005) 035(005) 000(005) 119(005) 070(005) 063(005) 042(005) 071(005)
068(005) 080(005) 062(005) 123(005) 084(005) 074(005) 087(005) 093(005)
061(005) 114(005) 083(005) 009(005) 078(005) 121(005) 026(005) 092(005)
104(005) 077(005) 087(005) 027(005) 013(005) 046(005) 056(005) 094(005)
CAT:0001

```

Copyright © Luca Marchese. ORC-ID: <http://orcid.org/0000-0001-7903-7540>

NCAGE (NATO Commercial And Government Entity Code : AK845

Email: luca.marchese@synaptics.org web: www.synaptics.org

IEEE Senior Member – INNS Senior Memeber

Copyright © note: the entire document with the copyright note can be freely reproduced, stored and distributed. Parts of the document can be reproduced with the obligation to cite the source.

January 3, 2019

© Luca Marchese luca.marchese@synaptics.org
www.synaptics.org

AC1 043 (010) 022 (010) 003 (010) 083 (010) 068 (010) 083 (005) 049 (005) 037 (005)
118 (005) 025 (005) 040 (005) 115 (005) 108 (005) 062 (005) 124 (005) 105 (005)
101 (005) 003 (005) 043 (005) 105 (005) 064 (005) 047 (005) 059 (005) 110 (005)
008 (005) 027 (005) 115 (005) 088 (005) 100 (005) 026 (005) 052 (005) 095 (005)
090 (005) 038 (005) 003 (005) 123 (005) 070 (005) 063 (005) 046 (005) 067 (005)
071 (005) 081 (005) 059 (005) 124 (005) 080 (005) 075 (005) 090 (005) 090 (005)
065 (005) 112 (005) 079 (005) 005 (005) 080 (005) 123 (005) 024 (005) 096 (005)
101 (005) 081 (005) 086 (005) 028 (005) 016 (005) 046 (005) 056 (005) 097 (005)
CAT : 0001

Each AC0 record is followed by a AC1 record. AC0 is the learning pattern and AC1 is the associated validation pattern. The validation pattern contains the components of the learning pattern with added noise limited by the “generalization box” defined after any component between parentheses.

(**LEARNING COMPONENT VALUE**, **BOX GENERALIZATION(NIF)**, **VALIDATION COMPONENT VALUE**)

The option 3 (DATAGEN COMPLEX (8 CAT / 64 FEATURES) generates a file with the same format but the patterns are not completely random. Different complex and relative constraints are applied to the values for the 8 categories (like in a real problem).

Independently by the database created (options 0,1,2,3, DATAGEN) the sequence of operations that should be performed is the following:

SET EXAMPLES NUM	(optional or default – the XOR test has 4 examples)
DATAGEN	(generation of the file containing the training set)
INITIALIZE NEURAL NETWORK	(initialization of the network)
LEARN	(learning task)
RECALL	(recall of the training examples)
RECON	(recognition of the validation examples)

The INITIALIZE NEURAL NETWORK operation will ask for the number of components contained in the pattern. You must put the same number of components of the dataset (created by DATAGEN):

Copyright © Luca Marchese. ORC-ID: <http://orcid.org/0000-0001-7903-7540>

NCAGE (NATO Commercial And Government Entity Code : AK845

Email: luca.marchese@synaptics.org web: www.synaptics.org

IEEE Senior Member – INNS Senior Memeber

Copyright © note: the entire document with the copyright note can be freely reproduced, stored and distributed. Parts of the document can be reproduced with the obligation to cite the source.


```
|
| 0) DATAGEN RANDOM (8CAT/64COMP)
| 1) DATAGEN XOR (2CAT/2COMP)
| 2) DATAGEN CIRCLE_IN_THE_SQUARE(2CAT/2COMP)
| 3) DATAGEN COMPLEX (8CAT/64COMP)
| 4) INITIALIZE NEURAL NETWORK
| 5) READ CURRENT DATASET TYPE
| 6) SET EXAMPLES NUM (DATAGEN 0,2,3)
| 7) LEARN
| 8) RECALL
| 9) RECOGNITION
| 10) SAVE NEURAL NETWORK SYNAPSES
| 11) LOAD NEURAL NETWORK SYNAPSES
| 12) REPORT
| 13) GDPR ON (STORE/EXTRACT RULES)
| 14) GDPR OFF (FASTER INFERENCE)(DEFAULT)
| 15) EXIT
|
|-----|
choice:4
pattern comps[2 or 64]:64
```

If the number of examples is large the operation could require some minutes to be completely performed. When the learning task is performed, the program shows the number of patterns that have been read and the number of patterns that have been recognized and not learned:

```
choice:8
EXAMPLES READ: 8000
EXAMPLES RECOGNIZED: 0
```

If the learning task is executed a second time all the patterns are recognized and not learned:

Copyright © Luca Marchese. ORC-ID: <http://orcid.org/0000-0001-7903-7540>

NCAGE (NATO Commercial And Government Entity Code : AK845

Email: luca.marchese@synaptics.org web: www.synaptics.org

IEEE Senior Member – INNS Senior Memeber

Copyright © note: the entire document with the copyright note can be freely reproduced, stored and distributed. Parts of the document can be reproduced with the obligation to cite the source.

January 3, 2019

© Luca Marchese luca.marchese@synaptics.org
www.synaptics.org

```
choice:8
EXAMPLES READ: 8000
EXAMPLES RECOGNIZED: 8000
```

Executing the RECALL operation, the same patterns of the training set (AC0) are inputted to the network. The result of each recognition is reported in the file “RULENN_test.res” (due to all these file writing operations the process requires some time more). The format of the file “RULENN_test.res” is the following:

```
CAT: 00001 CRE: 00001 STR: 255 ->00001(255)
CAT: 00001 CRE: 00001 STR: 255 ->00001(255)
CAT: 00001 CRE: 00001 STR: 255 ->00001(255)
CAT: 00001 CRE: 00001 STR: 255 ->00001(255)->00004(250)->00006(250)
.....
```

Each row is related to an input pattern. The fields of the record have this meaning:

CAT: <expected category>

CRE/WRE/UNC: <recognized category> or NID

(CRE=correct, WRE=wrong, UNC=uncertainty, NID=not identified)

STR: <maximum recognition strength>

-><first recognized category><strength>-><second recognized category><strength>-> ...

At the end there is a resume line like this:

```
CRE:30000(100.000000%) UNC:0(0.000000%) NID:0(0.000000%) WRE:0(0.000000%)
```

Execution the RECON operation, the noisy patterns (AC1) are inputted to the network. The result of each recognition is reported in the file “RULENN_test.res”.

PLEASE NOTE THAT reports are always APPENDED to the existing file “RULENN_test.res”. If you want a file with only the last results you must delete the file before the execution of the RECALL/RECON operations that you want record.

The use of the RANDOM database (option 0) produces bad RECON results with a large dataset (>15000). With a RANDOM database you should use also small BOX distance generalization values (i.e. <=5) because the values of the components of the generated vectors are completely random and the vectors are assigned to random categories in the range 0-8.

Copyright © Luca Marchese. ORC-ID: <http://orcid.org/0000-0001-7903-7540>

NCAGE (NATO Commercial And Government Entity Code : AK845

Email: luca.marchese@synaptics.org web: www.synaptics.org

IEEE Senior Member – INNS Senior Memeber

Copyright © note: the entire document with the copyright note can be freely reproduced, stored and distributed. Parts of the document can be reproduced with the obligation to cite the source.

January 3, 2019

© Luca Marchese luca.marchese@synaptics.org
www.synaptics.org

By using the option 3 (DATAGEN COMPLEX (8 CAT / 64 FEATURES)) the generated patterns are associated with the 8 categories following some criteria (range of the values of the components and relations between the values of different components). In such a context, also with large datasets (>15000) there are good RECON results.

The constraints are managed inside the function `void datagen_complex(void)` inside the `test_RULENN.c` code: you can change the code to build personalized classes.

The Datasets options DATAGEN XOR and DATAGEN CIRCLE_IN_THE_SQUARE produce 2 components vectors related to the known basic problems.

NOTE: The network must be initialized setting a number of components matching the type Dataset that will be used.

USING THE TESTING SOFTWARE (EXAMPLE)

```
1> TOY_TEST
2> EXTENSIVE_TEST
3> EXIT

choice:2
```

(we select the EXTENSIVE TEST)

Copyright © Luca Marchese. ORC-ID: <http://orcid.org/0000-0001-7903-7540>

NCAGE (NATO Commercial And Government Entity Code : AK845

Email: luca.marchese@synaptics.org web: www.synaptics.org

IEEE Senior Member – INNS Senior Memeber

Copyright © note: the entire document with the copyright note can be freely reproduced, stored and distributed. Parts of the document can be reproduced with the obligation to cite the source.

January 3, 2019

© Luca Marchese luca.marchese@synaptics.org
www.synaptics.org

```

RULES BASED NEURAL NETWORK
-----Author: Luca Marchese-----
-----luca.marchese@synaptics.org-----
-----www.synaptics.org-----
-----
|
| 0) DATAGEN RANDOM (8CAT/64COMP)
| 1) DATAGEN XOR (2CAT/2COMP)
| 2) DATAGEN CIRCLE_IN_THE_SQUARE(2CAT/2COMP)
| 3) DATAGEN COMPLEX (8CAT/64COMP)
| 4) INITIALIZE NEURAL NETWORK
| 5) READ CURRENT DATASET TYPE
| 6) SET EXAMPLES NUM (DATAGEN 0,2,3)
| 7) LEARN
| 8) RECALL
| 9) RECOGNITION
| 10) SAVE NEURAL NETWORK SYNAPSES
| 11) LOAD NEURAL NETWORK SYNAPSES
| 12) REPORT
| 13) GDPR ON (STORE/EXTRACT RULES)
| 14) GDPR OFF (FASTER INFERENCE)(DEFAULT)
| 15) EXIT
|
-----
choice:6
ACTUAL EXAMPLES NUMBER: 30000
EXAMPLES NUMBER[9-30000]:30000

```

(we leave the default number of examples)

Copyright © Luca Marchese. ORCID-ID: <http://orcid.org/0000-0001-7903-7540>

NCAGE (NATO Commercial And Government Entity Code : AK845

Email: luca.marchese@synaptics.org web: www.synaptics.org

IEEE Senior Member – INNS Senior Memeber

Copyright © note: the entire document with the copyright note can be freely reproduced, stored and distributed. Parts of the document can be reproduced with the obligation to cite the source.

```

|-----|
| 0) DATAGEN RANDOM (8CAT/64COMP) |
| 1) DATAGEN XOR (2CAT/2COMP) |
| 2) DATAGEN CIRCLE_IN_THE_SQUARE(2CAT/2COMP) |
| 3) DATAGEN COMPLEX (8CAT/64COMP) |
| 4) INITIALIZE NEURAL NETWORK |
| 5) READ CURRENT DATASET TYPE |
| 6) SET EXAMPLES NUM (DATAGEN 0,2,3) |
| 7) LEARN |
| 8) RECALL |
| 9) RECOGNITION |
| 10) SAVE NEURAL NETWORK SYNAPSES |
| 11) LOAD NEURAL NETWORK SYNAPSES |
| 12) REPORT |
| 13) GDPR ON (STORE/EXTRACT RULES) |
| 14) GDPR OFF (FASTER INFERENCE)(DEFAULT) |
| 15) EXIT |
|-----|
choice:0
NIF[000](0-50):5
NIF[001](0-50):5
NIF[002](0-50):5
NIF[003](0-50):5
NIF[004](0-50):5

```

...set the NIF=5 for all the 64 components

The file "RULENN_ext_test.dat" is created with 30,000 examples (following the first example associated with category 1 and the last example associated with category 8

```

AC0 039(005) 015(005) 000(005) 090(005) 070(005) 087(005) 053(005) 034(005)
118(005) 024(005) 044(005) 117(005) 105(005) 065(005) 124(005) 107(005)
105(005) 006(005) 042(005) 109(005) 065(005) 046(005) 056(005) 109(005)
008(005) 023(005) 117(005) 088(005) 102(005) 030(005) 048(005) 091(005)
092(005) 035(005) 000(005) 119(005) 070(005) 063(005) 042(005) 071(005)
068(005) 080(005) 062(005) 123(005) 084(005) 074(005) 087(005) 093(005)
061(005) 114(005) 083(005) 009(005) 078(005) 121(005) 026(005) 092(005)
104(005) 077(005) 087(005) 027(005) 013(005) 046(005) 056(005) 094(005)
CAT:0001
AC1 041(005) 018(005) 001(005) 087(005) 069(005) 083(005) 049(005) 037(005)
118(005) 025(005) 040(005) 115(005) 108(005) 062(005) 124(005) 105(005)
101(005) 003(005) 043(005) 105(005) 064(005) 047(005) 059(005) 110(005)
008(005) 027(005) 115(005) 088(005) 100(005) 026(005) 052(005) 095(005)
090(005) 038(005) 003(005) 123(005) 070(005) 063(005) 046(005) 067(005)

```

Copyright © Luca Marchese. ORC-ID: <http://orcid.org/0000-0001-7903-7540>

NCAGE (NATO Commercial And Government Entity Code : AK845

Email: luca.marchese@synaptics.org web: www.synaptics.org

IEEE Senior Member – INNS Senior Memeber

Copyright © note: the entire document with the copyright note can be freely reproduced, stored and distributed. Parts of the document can be reproduced with the obligation to cite the source.

January 3, 2019

© Luca Marchese luca.marchese@synaptics.org
www.synaptics.org

071 (005) 081 (005) 059 (005) 124 (005) 080 (005) 075 (005) 090 (005) 090 (005)
065 (005) 112 (005) 079 (005) 005 (005) 080 (005) 123 (005) 024 (005) 096 (005)
101 (005) 081 (005) 086 (005) 028 (005) 016 (005) 046 (005) 056 (005) 097 (005)
CAT:0001

.....

AC0 074 (005) 105 (005) 016 (005) 126 (005) 005 (005) 085 (005) 038 (005) 037 (005)
108 (005) 054 (005) 057 (005) 077 (005) 019 (005) 014 (005) 109 (005) 037 (005)
059 (005) 096 (005) 076 (005) 112 (005) 084 (005) 050 (005) 109 (005) 112 (005)
059 (005) 037 (005) 079 (005) 051 (005) 072 (005) 074 (005) 068 (005) 063 (005)
058 (005) 104 (005) 038 (005) 053 (005) 088 (005) 062 (005) 050 (005) 114 (005)
075 (005) 095 (005) 021 (005) 112 (005) 088 (005) 008 (005) 029 (005) 070 (005)
019 (005) 046 (005) 103 (005) 116 (005) 113 (005) 074 (005) 065 (005) 066 (005)
121 (005) 039 (005) 103 (005) 051 (005) 111 (005) 085 (005) 067 (005) 109 (005)
CAT:0008

AC1 075 (005) 104 (005) 012 (005) 127 (005) 001 (005) 089 (005) 034 (005) 037 (005)
108 (005) 052 (005) 061 (005) 077 (005) 019 (005) 015 (005) 106 (005) 035 (005)
059 (005) 093 (005) 080 (005) 112 (005) 084 (005) 049 (005) 110 (005) 116 (005)
056 (005) 033 (005) 075 (005) 053 (005) 069 (005) 073 (005) 068 (005) 065 (005)
054 (005) 106 (005) 035 (005) 052 (005) 084 (005) 066 (005) 047 (005) 114 (005)
076 (005) 094 (005) 023 (005) 110 (005) 092 (005) 009 (005) 033 (005) 069 (005)
023 (005) 046 (005) 107 (005) 113 (005) 111 (005) 070 (005) 067 (005) 063 (005)
122 (005) 039 (005) 106 (005) 047 (005) 111 (005) 087 (005) 067 (005) 109 (005)
CAT:0008

Then we delete the "RULENN_test.res" file in order to have the results of the RECALL and RECON not appended to old data.

Copyright © Luca Marchese. ORCID-ID: <http://orcid.org/0000-0001-7903-7540>

NCAGE (NATO Commercial And Government Entity Code : AK845

Email: luca.marchese@synaptics.org web: www.synaptics.org

IEEE Senior Member – INNS Senior Memeber

Copyright © note: the entire document with the copyright note can be freely reproduced, stored and distributed. Parts of the document can be reproduced with the obligation to cite the source.

```
|-----|
| 0) DATAGEN RANDOM (8CAT/64COMP)
| 1) DATAGEN XOR (2CAT/2COMP)
| 2) DATAGEN CIRCLE_IN_THE_SQUARE(2CAT/2COMP)
| 3) DATAGEN COMPLEX (8CAT/64COMP)
| 4) INITIALIZE NEURAL NETWORK
| 5) READ CURRENT DATASET TYPE
| 6) SET EXAMPLES NUM (DATAGEN 0,2,3)
| 7) LEARN
| 8) RECALL
| 9) RECOGNITION
| 10) SAVE NEURAL NETWORK SYNAPSES
| 11) LOAD NEURAL NETWORK SYNAPSES
| 12) REPORT
| 13) GDPR ON (STORE/EXTRACT RULES)
| 14) GDPR OFF (FASTER INFERENCE)(DEFAULT)
| 15) EXIT
|-----|
choice:4
pattern comps[2 or 64]:64
```

(we initialize the neural network with 64 features as required by the dataset)

```
|-----|
| 0) DATAGEN RANDOM (8CAT/64COMP)
| 1) DATAGEN XOR (2CAT/2COMP)
| 2) DATAGEN CIRCLE_IN_THE_SQUARE(2CAT/2COMP)
| 3) DATAGEN COMPLEX (8CAT/64COMP)
| 4) INITIALIZE NEURAL NETWORK
| 5) READ CURRENT DATASET TYPE
| 6) SET EXAMPLES NUM (DATAGEN 0,2,3)
| 7) LEARN
| 8) RECALL
| 9) RECOGNITION
| 10) SAVE NEURAL NETWORK SYNAPSES
| 11) LOAD NEURAL NETWORK SYNAPSES
| 12) REPORT
| 13) GDPR ON (STORE/EXTRACT RULES)
| 14) GDPR OFF (FASTER INFERENCE)(DEFAULT)
| 15) EXIT
|-----|
choice:7
EXAMPLES READ: 30000
EXAMPLES RECOGNIZED: 0
```

Copyright © Luca Marchese. ORCID-ID: <http://orcid.org/0000-0001-7903-7540>

NCAGE (NATO Commercial And Government Entity Code : AK845

Email: luca.marchese@synaptics.org web: www.synaptics.org

IEEE Senior Member – INNS Senior Memeber

Copyright © note: the entire document with the copyright note can be freely reproduced, stored and distributed. Parts of the document can be reproduced with the obligation to cite the source.

(we learn the 30,000 examples)

```
| 0) DATAGEN RANDOM (8CAT/64COMP)
| 1) DATAGEN XOR (2CAT/2COMP)
| 2) DATAGEN CIRCLE_IN_THE_SQUARE(2CAT/2COMP)
| 3) DATAGEN COMPLEX (8CAT/64COMP)
| 4) INITIALIZE NEURAL NETWORK
| 5) READ CURRENT DATASET TYPE
| 6) SET EXAMPLES NUM (DATAGEN 0,2,3)
| 7) LEARN
| 8) RECALL
| 9) RECOGNITION
| 10) SAVE NEURAL NETWORK SYNAPSES
| 11) LOAD NEURAL NETWORK SYNAPSES
| 12) REPORT
| 13) GDPR ON (STORE/EXTRACT RULES)
| 14) GDPR OFF (FASTER INFERENCE)(DEFAULT)
| 15) EXIT
|
|-----|
choice:8
```

(we execute a RECALL operation and a file "RULENN_test.res" is created)

Opening with WordPad the file RULENN_test.res we find 30,000 records like this:

```
CAT: 00001 CRE: 00001 STR: 255 ->00001(255)->00002(252)->00003(252)->00004(252)->00005(252)->00006(251)->00007(251)->00008(251)
```

All the 30,000 patterns have been recalled also in wrong categories with lower strength (i.e. 252, 251). Do not forget that these patterns are randomly generated and randomly assigned to categories: this crosstalk phenomenon is normal with this dataset.

We find a final record that shows 100% patterns have been recalled correctly:

```
CRE:30000(100.000000%) UNC:0(0.000000%) NID:0(0.000000%) WRE:0(0.000000%)
```

Copyright © Luca Marchese. ORC-ID: <http://orcid.org/0000-0001-7903-7540>

NCAGE (NATO Commercial And Government Entity Code : AK845

Email: luca.marchese@synaptics.org web: www.synaptics.org

IEEE Senior Member – INNS Senior Memeber

Copyright © note: the entire document with the copyright note can be freely reproduced, stored and distributed. Parts of the document can be reproduced with the obligation to cite the source.

January 3, 2019

© Luca Marchese luca.marchese@synaptics.org
www.synaptics.org

```
|-----|
| 0) DATAGEN RANDOM (8CAT/64COMP) |
| 1) DATAGEN XOR (2CAT/2COMP) |
| 2) DATAGEN CIRCLE_IN_THE_SQUARE(2CAT/2COMP) |
| 3) DATAGEN COMPLEX (8CAT/64COMP) |
| 4) INITIALIZE NEURAL NETWORK |
| 5) READ CURRENT DATASET TYPE |
| 6) SET EXAMPLES NUM (DATAGEN 0,2,3) |
| 7) LEARN |
| 8) RECALL |
| 9) RECOGNITION |
| 10) SAVE NEURAL NETWORK SYNAPSES |
| 11) LOAD NEURAL NETWORK SYNAPSES |
| 12) REPORT |
| 13) GDPR ON (STORE/EXTRACT RULES) |
| 14) GDPR OFF (FASTER INFERENCE)(DEFAULT) |
| 15) EXIT |
|-----|
choice:9
```

(then we execute the recognition of noisy patterns with up to +/-5 in each component)

The results are appended to the RULENN_test.res file. There are other 30,000 records like this:

```
CAT: 00001 CRE: 00001 STR: 255 ->00001 (255) ->00002 (252) ->00003 (252) ->00004 (252) ->00005 (252) ->00006 (252) ->00007 (252) ->00008 (252)
```

```
CAT: 00001 #WRE: 00008 STR: 252 ->00008 (252) ->00001 (251) ->00002 (251) ->00003 (251) ->00004 (251) ->00007 (251) ->00005 (250) ->00006 (250)
```

```
CAT: 00001 @UNC: 00002 STR: 252 ->00002 (252) ->00003 (252) ->00004 (252) ->00005 (252) ->00008 (252) ->00001 (251) ->00006 (251) ->00007 (251)
```

And the final record:

```
CRE:5527 (18.423333%) UNC:24077 (80.256667%) NID:0 (0.000000%)
WRE:396 (1.320000%)
```

NOTE: as stated, the case of 30,000 random examples is the worst case (only 18.4% correctly recognized). With DATAGEN RANDOM database, many patterns are actually overlapped on the box distance but associated with different categories. The use of DATAGEN COMPLEX database is more similar to a real problem.

Copyright © Luca Marchese. ORC-ID: <http://orcid.org/0000-0001-7903-7540>

NCAGE (NATO Commercial And Government Entity Code : AK845

Email: luca.marchese@synaptics.org web: www.synaptics.org

IEEE Senior Member – INNS Senior Memeber

Copyright © note: the entire document with the copyright note can be freely reproduced, stored and distributed. Parts of the document can be reproduced with the obligation to cite the source.

SETTING GDPR ON

```
|-----|
| 0) DATAGEN RANDOM (8CAT/64COMP)
| 1) DATAGEN XOR (2CAT/2COMP)
| 2) DATAGEN CIRCLE_IN_THE_SQUARE(2CAT/2COMP)
| 3) DATAGEN COMPLEX (8CAT/64COMP)
| 4) INITIALIZE NEURAL NETWORK
| 5) READ CURRENT DATASET TYPE
| 6) SET EXAMPLES NUM (DATAGEN 0,2,3)
| 7) LEARN
| 8) RECALL
| 9) RECOGNITION
| 10) SAVE NEURAL NETWORK SYNAPSES
| 11) LOAD NEURAL NETWORK SYNAPSES
| 12) REPORT
| 13) GDPR ON (STORE/EXTRACT RULES)
| 14) GDPR OFF (FASTER INFERENCE)(DEFAULT)
| 15) EXIT
|-----|
choice:13
```

If GDPR is set ON (option 13) before a LEARN operation, then a binary file “rules_db.bin” is created during the learning phase. This binary file contains rules extracted from data. A piece of a “rules_db_bin” edited with a binary editor is shown in the following picture. There is no loss of performance in the speed of the learning process.

Copyright © Luca Marchese. ORCID-ID: <http://orcid.org/0000-0001-7903-7540>

NCAGE (NATO Commercial And Government Entity Code : AK845

Email: luca.marchese@synaptics.org web: www.synaptics.org

IEEE Senior Member – INNS Senior Memeber

Copyright © note: the entire document with the copyright note can be freely reproduced, stored and distributed. Parts of the document can be reproduced with the obligation to cite the source.

Address	00	01	02	03	04	05	06	07	08	09	0a	0b	0c	0d	0e	0f	ASCII
00000000	01	00	0a	00	22	64	32	28	1e	7d	50	0a	64	64	05	05	...\"d2(.)P.dd..
00000010	05	05	05	05	05	05	05	05	02	00	0a	00	64	14	1e	5ad..Z
00000020	43	64	64	64	64	64	05	05	05	05	05	05	05	05	05	06	Cdddd.....
00000030	03	00	0a	00	0a	14	1e	28	32	3c	46	50	5a	64	05	05(2<FPZd..
00000040	05	05	05	05	05	05	05	07	04	00	0a	00	60	1a	1b	60`..`
00000050	3e	6a	6a	6a	6a	5e	05	05	05	05	05	05	05	05	05	08	>jjjj^.....
00000060	05	00	0a	00	1e	5a	1e	5a	1e	5a	1e	5a	1e	5a	05	05Z.Z.Z.Z.Z..
00000070	05	05	05	05	05	05	05	09
00000080

If GDPR is set ON (option 13) before a LEARN operation, then a binary file “rules_db.bin” should exist.

If GDPR is set ON (option 13) before a RECOGNITION, then a file “rules_ex.txt” is created. It contains all the rules that are used for making inference on the input data. Typically GDPR=ON in the phase of recognition is used with only one input to validate the inference with readable rules. The recognition with GDPR=ON has a lower speed performance. This program is an evaluation tool: use GDPR=ON on small dataset(<200).

```
IF (056 < VAR[00]=063 < 066) AND
(025 < VAR[01]=028 < 035) AND
(003 < VAR[02]=011 < 013) AND
(056 < VAR[03]=061 < 066) AND
(085 < VAR[04]=090 < 095) AND
(079 < VAR[05]=088 < 089) AND
(081 < VAR[06]=086 < 091) AND
(010 < VAR[07]=019 < 020) AND
(117 < VAR[08]=123 < 127) AND
(038 < VAR[09]=045 < 048) AND
(119 < VAR[10]=122 < 127) AND
(080 < VAR[11]=085 < 090) AND
(079 < VAR[12]=085 < 089) AND
(041 < VAR[13]=047 < 051) AND
(103 < VAR[14]=105 < 113) AND
(069 < VAR[15]=074 < 079) AND
(043 < VAR[16]=047 < 053) AND
(024 < VAR[17]=025 < 034) AND
(023 < VAR[18]=032 < 033) AND
```

Copyright © Luca Marchese. ORC-ID: <http://orcid.org/0000-0001-7903-7540>

NCAGE (NATO Commercial And Government Entity Code : AK845

Email: luca.marchese@synaptics.org web: www.synaptics.org

IEEE Senior Member – INNS Senior Memeber

Copyright © note: the entire document with the copyright note can be freely reproduced, stored and distributed. Parts of the document can be reproduced with the obligation to cite the source.

(029 < VAR[19]=030 < 039) AND
(007 < VAR[20]=013 < 017) AND
(046 < VAR[21]=054 < 056) AND
(083 < VAR[22]=092 < 093) AND
(058 < VAR[23]=060 < 068) AND
(090 < VAR[24]=097 < 100) AND
(027 < VAR[25]=032 < 037) AND
(080 < VAR[26]=085 < 090) AND
(068 < VAR[27]=074 < 078) AND
(000 < VAR[28]=005 < 006) AND
(051 < VAR[29]=060 < 061) AND
(063 < VAR[30]=071 < 073) AND
(114 < VAR[31]=119 < 124) AND
(000 < VAR[32]=003 < 006) AND
(041 < VAR[33]=045 < 051) AND
(095 < VAR[34]=098 < 105) AND
(047 < VAR[35]=056 < 057) AND
(038 < VAR[36]=041 < 048) AND
(066 < VAR[37]=072 < 076) AND
(097 < VAR[38]=102 < 107) AND
(038 < VAR[39]=040 < 048) AND
(017 < VAR[40]=018 < 027) AND
(067 < VAR[41]=073 < 077) AND
(039 < VAR[42]=044 < 049) AND
(012 < VAR[43]=018 < 022) AND
(050 < VAR[44]=055 < 060) AND
(038 < VAR[45]=045 < 048) AND
(110 < VAR[46]=115 < 120) AND
(104 < VAR[47]=107 < 114) AND
(065 < VAR[48]=068 < 075) AND
(035 < VAR[49]=042 < 045) AND
(055 < VAR[50]=064 < 065) AND
(084 < VAR[51]=090 < 094) AND
(041 < VAR[52]=050 < 051) AND
(087 < VAR[53]=092 < 097) AND
(046 < VAR[54]=051 < 056) AND
(068 < VAR[55]=073 < 078) AND
(119 < VAR[56]=120 < 127) AND

Copyright © Luca Marchese. ORC-ID: <http://orcid.org/0000-0001-7903-7540>

NCAGE (NATO Commercial And Government Entity Code : AK845

Email: luca.marchese@synaptics.org web: www.synaptics.org

IEEE Senior Member – INNS Senior Memeber

Copyright © note: the entire document with the copyright note can be freely reproduced, stored and distributed. Parts of the document can be reproduced with the obligation to cite the source.

January 3, 2019

© Luca Marchese luca.marchese@synaptics.org
www.synaptics.org

```
(044 < VAR[57]=051 < 054) AND  
(044 < VAR[58]=052 < 054) AND  
(100 < VAR[59]=106 < 110) AND  
(087 < VAR[60]=088 < 097) AND  
(069 < VAR[61]=076 < 079) AND  
(097 < VAR[62]=101 < 107) AND  
(064 < VAR[63]=071 < 074)  
THEN CATEGORY=01
```

Copyright © Luca Marchese. ORC-ID: <http://orcid.org/0000-0001-7903-7540>

NCAGE (NATO Commercial And Government Entity Code : AK845

Email: luca.marchese@synaptics.org web: www.synaptics.org

IEEE Senior Member – INNS Senior Memeber

Copyright © note: the entire document with the copyright note can be freely reproduced, stored and distributed. Parts of the document can be reproduced with the obligation to cite the source.