Appendix I
Analog Complexing

5.2.3 Application to classification

The word "classification" is ambiguous. Along with the following considerations it means assigning a new object/case to one of an existing set of possible classes. "Classification is learning a function that maps (classifies) a data item into one of several predefined classes " [Fayyad, 96].

Finding the classes themselves from a given set of "unclassified" objects/cases (unsupervised classification) is the task of cluster analysis, which was considered in section 5.1, and which will be solved by means of Analog Complexing in section 5.2.4. Often, the set of classes is known or given from empirical or theoretical systems analysis.

Once such a set of classes C_i (i=1, 2, ..., n) has been found, it can be used for classifying new cases $\underline{x}_{N+k} = \{x_{1N+k}, x_{2N+k}, \ldots, x_{mN+k}\}$, k=1, 2, ..., M.

Given a sample of objects $\underline{x}_j = \{x_{1j}, \ldots, x_{mj}\}$, j=1, 2, ..., N, N+1, ..., N+M, where objects j=1, 2, ..., N are already classified, the membership function

$$\mu_{C_i}(\underline{x}_j) = \left\{ \begin{matrix} 1 & \underline{x}_j & C_i \\ 0 & \underline{x}_j & C_i \end{matrix} \right\}$$

can be calculated for each object $\underline{x}_j$. The task of classification is to estimate the membership function μ_{C_i} for new, not yet classified observations $\underline{x}_{N+k}$, k=1, 2, ..., M, either in a bivalent (binary) form

$$\mu_{C_i}(\underline{x}_{N+k}) = \left\{ \begin{matrix} 1 & \underline{x}_{N+k} & C_i \\ 0 & \underline{x}_{N+k} & C_i \end{matrix} \right\}$$

or as a real-valued estimation

$$\mu_{C_i}(\underline{x}_{N+k}), \text{ where } 0 \le \mu_{C_i}(\underline{x}_{N+k}) \le 1.$$

Using the Analog Complexing algorithm, each row of cases $P_1(i)=\{x_{1i}, x_{2i}, \ldots, x_{mi}\}$, i=1, 2, ..., N, represents a pattern of length m (number of variables describing an object/case) with N as the number of known classified objects/cases.

Each new, unclassified object/case can be considered a reference pattern $P_1(N+k)=\{x_{1N+k}, \ldots, x_{mN+k}\}$, k=1, 2, ..., M.

Applying Analog Complexing considering all N possible analogous patterns, the similarity measure s_{ik} (total sum of squares) of the k-th unclassified object/case (reference pattern) can be calcula ted relative to the i-th object/case (analogous pattern). Using these similarities, the "memberhip function" of each unclassified object/case

$$\mu_{C_i}(\underline{x}_{N+k}) = \frac{1}{n_{C_i}} \sum_{j \in N_{C_i}} s_{jk}$$

is estimated, where N_{C_i} - set of indexes j of objects/cases $\underline{x}_j$ included in class C_i ($\underline{x}_j$ C_i), and n_{C_i} their number of indexes .

If it is necessary to get a binary membership function

$$\mu_{C_r}(\underline{x}_{N+k}) = \begin{Bmatrix} 1 & \underline{x}_{N+k} & C_r \\ 0 & \underline{x}_{N+k} & C_r \end{Bmatrix}$$

an estimation is given by means of the maximum operator

$$\mu_{C_r}(\underline{x}_{N+k}) = \max\{\mu_{C_i}(\underline{x}_{N+k}) / i = 1, 2, ..., n\}.$$

But often, without preselection of variables, the membership functions do not differ very sharply. The reason is that when using *all* variables x_1, x_2, ..., x_m (i.e. relevant *and* nonrelevant) to repre sent the objects/cases, the classification power may be diluted. Using a preselection of variables (section 5.3.1: "set of variables used"), a so-called nucleus is obtained that, if then taken to des cribe the objects/cases, may improve the classification results.

Example: *Solvency checking*

In section 8.1.5 an example about solvency checking of companies by a bank is explained. This is a typical classification problem. Given are 71 (36 positive, 35 negative) decisions of a bank(objects) derived from 19 balance sheet characteristics. The classification task is to decide (classifiy) that 10 new companies' solvency. Using the above algorithm on all 19 variables (table 5.3.a) and a nucleus of variables (table 5.3.b), both the real-valued and the binary membership functions were estimated.

case	negative	positive	decision
1	0.13	0.35	p
2	0.04	0.09	p
3	0.17	0.45	p
4	0.14	0.04	n
5	0.10	0.27	p
6	0.36	0.06	n
7	0.29	0.03	n
8	0.18	0.34	p
9	0.23	0.43	p
10	0.38	0.33	n

Table 5.3.a: Membership functions of 10 companies using all variables

case	negative	positive	decision
1	0.18	0.36	p
2	0.24	0.27	p
3	0.23	0.48	p
4	0.31	0.03	n
5	0.13	0.27	p
6	0.36	0.06	n
7	0.29	0.02	n
8	0.31	0.42	p
9	0.27	0.47	p
10	0.45	0.38	n

Table 5.3.b: Membership functions of 10 companies using a nucleus of variables (5.2.4)

In Table 5.3.a, the test cases 2, 4 and 10 are uncertain. Using the nucleus described in section 5.2.4, test case 4, especially, but also the 10-th case are becoming a more certain decision while for the 2nd case its undecided nature is confirmed.

5.2.4 APPLICATION TO CLUSTERING

The goal of clustering is finding an optimal partitioning of the distribution in the $\underline{x}$-space into a unknown number of regions (clusters). "Clustering is a common descriptive task where one seeks to identify a finite set of categories to describe the data " [Fayyad, 96].

Given are N objects or N samples (of an object or time process) and m variables $\underline{x}_i = \{x_{1i}, ..., x_{ji}, ..., x_{mi}\}$, i=1, 2, ..., N:

$$X = \begin{bmatrix} x_{11} & . \; . \; x_{j1} & . \; . \; x_{m1} \\ . & . \; . \; . & . \; . \; . \\ x_{1i} & . \; . \; x_{ji} & . \; . \; x_{mi} \\ . & . \; . \; . & . \; . \; . \\ x_{1N} & . \; . \; x_{jN} & . \; . \; x_{mN} \end{bmatrix}$$

The i-th row represents the object O_i or the i-th sample, and column j represents the variable x_j.

a. Clustering in the state space

The state space has as the axes the variables $x_1, x_2, ..., x_m$. The objects O_i, i=1, 2, ..., N are distributed points in the state space. Each object O_k differ more or less from all other objects O_h, the differences (distances) can be estimated by s^1_{kh}, k=1, 2, ..., N, h=1, 2, ..., N. The basis for clustering therefore is the symmetrical similartiy matrix $S^1_{NN}=\{s^1_{kh}\}$. The task of clustering is to subdivide the state space into n clusters of similar objects.

b. Clustering in the sample space

The sample space has as the axes the objects $O_1, O_2, ..., O_N$ (or several points of time T=1, 2, ..., N for time processes). The variables $x_1, ..., x_m$ that are describing the objects (process) are distributed points in the sample space. Each variable x_u differ more or less from all other varibales x_v, the differences (distances) can be estimated by s^2_{uv}, u=1, 2, ..., m, v=1, 2, ..., m. The basis for clustering is now the symmetrical matrix $S^2_{mm}=\{s^2_{uv}\}$. The task of clustering is to subdivide the sample space into n clusters of variables that similary influence the objects or the process. This sort of clustering was used to estimate the nucleus in section 1.3, for example.

Analog Complexing evaluates the similarity matrix S^1 or S^2. Here, each row i (i=1, 2, ..., N, clustering in state space) or column j (j=1, 2, ..., m, clustering in the sample space) is a reference pattern with a pattern length of m (N), all other rows (colums) are patterns, which the similarity measure s^1_{ik}, (k=1, 2, ..., N) (s^2_{jh}, h=1, 2, ..., m), has to be evaluated for. From the similarity matrix S^1 (S^2) a Boolean structure matrix $B=\{b_{ij}\}$ can be calculated using a threshold value τ

$$b_{ij} = \begin{cases} 1 & s_{ij} \geq \tau \\ 0 & s_{ij} < \tau \end{cases}$$

where b_{ii}=0 as for avoiding self-cycles. Using the known algorithms of structure analysis [Weil, 68], the

- number of clusters and the
- elements each cluster consists of

can be derived.

The result of clustering depends on the threshold value τ. For $\tau < \tau_0$, there is only one cluster that includes all objects/variables. For $\tau > \tau_0$, the number of clusters increases up to τ_1, where each object/variable is a cluster of its own.

Example: *Solvency checking*

Applying the AC Clustering algorithm on the 81 objects (35 negative, 36 positive and 10 unknown decisions) and all 19 variables, these results were obtained:

a. clustering in the state space

Here, the pattern length is 19, the number of objects is N=81. Table 5.4 shows an extract from the generated 21 clusters for τ=0.70. From this follows:

- t6, t7 are included in a cluster with exclusively negative decisions,
- t1, t3, t5, t8, t9, t10 are included in a cluster with major positive decisions (18 negative and 24 positive decisions),
- t2, t4 are not included in other clusters.

The test cases t6, t7, in all tests, have always been clear negative decisions (see also 8.1.5). This is confirmed by this clustering result. The inclusion of companies t1, t3, t5, t8, t9 in cluster with major positive decisions corresponds with classifiaction results above. Also, the specific character of t2, t4 and t10 is reflected (see section. 5.2.3).

cluster	objects
1	n1, n4, n8, n10, n17, n22, n25, n32, n33, n35, n36, t6, t7
2	n2, n5, n6,..., n34, p1, p3, p4, ..., p35, t1, t3, t5, t8, t9, t10
3	t2
4	t4
...	...

Tab. 5.4: Extracted results of clustering in the state space

b. Clustering in the sample space

In this case the pattern length is 81 while the number of variables is m=19. Table 5.5 shows some results of clustering.

From this clustering results, a nucleus of variables can be selected. For $\tau = 0.4$, for example, the nucleus can be composed of x_1, x_2, x_3, x_{11}, x_{12}, x_{14}, x_{15}, x_{16}, x_{17}, x_{18}, x_{19}. Using this nucleus, the classification results are listed in table 5.3.b (section 5.2.3).

cluster	$\tau = 0.4$	$0.5 \le \tau \le 0.6$	$\tau = 0.7$
1	X1	X1	X1
2	X2	X2	X2
3	X3,X4,X5,X6,X7,X8,X9,X10,X13	X3,X4,X7,X8,X9,X10	X3,X9
4	X11	X5,X6,	X4
5	X12	X11	X5
6	X14	X12	X6
7	X15	X13	X7,X8,X10
8	X16	X14	X11
9	X17	X15	X12
10	X18	X16	X13
11	X19	X17	X14
12		X18	X15
13		X19	X16
14			X17
15			X18
16			X19

Tab. 5.5: Results of clustering in the sample space (variables $x_1, ..., x_{19}$)

8.3.3 The Two Spirals Classification Benchmark

Data Source

Gesellschaft zur Foerderung angewandter Informatik e.V.,
Adaptive Modelling and Statistical Pattern Recognition Department

http://zarnow.gfai.de/twsp.htm

Problem

The task is to learn to discriminate between two sets of training points which lie on two distinct spirals in the x-y plane. These spirals coil three times around the origin and around one another. This appears to be a very difficult task for back-propagation networks and their relatives. Problems like this one, whose inputs are points on the 2-D plane, are interesting because we can display the 2-D "receptive field" of any unit in the network. The problem was first published by Scott E.

Fahlman, CMU.

Information Used

Given are training (fig. 8.27) and testing data of two sets of points, each with 96 members (three complete revolutions at 32 points per revolution).

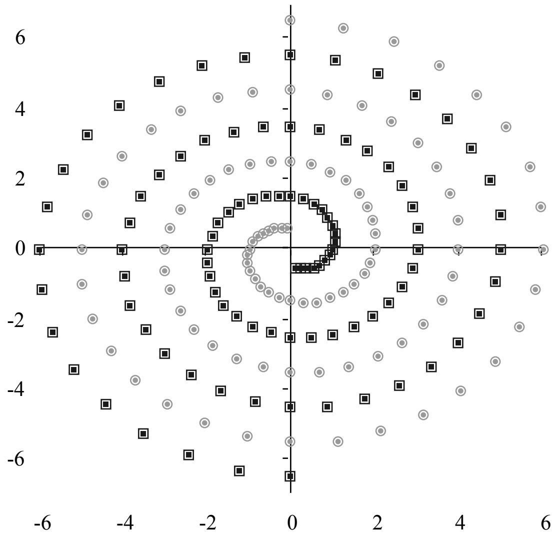

Fig. 8.27: Training data of the two spirals

Each point is represented by two floating-point numbers, the X and Y coordinates, plus an output value of 0.5 for one set, -0.5 for the other set. For Neural Networks, the task is to train on the 192 I/O pairs of the training data set until the learning system can produce the correct output for all of the inputs. The time required for learning can be immense since several network topologies have to be trained and tested until a 'satisfying' result is obtained.

Solution

To solve this classification task we used Analog Complexing (section 5.2.3). Since the given X and Y coordinates do not form a sufficiently distinguishable pattern, we added variables $\underline{v}_i$, i=1, 2, to get a pattern length of 4, finally:

$$x_j;\ y_j;\ v_{1j}=x_j^3;\ v_{2j}=y_j^3,\ j=1, 2, ..., N.$$

Using this extended data base, each point of the testing data set was used as a reference pattern to find the corresponding most similar pattern from the training data. The known class of the selected similar pattern is then assigned to the reference pattern as the resulting class.

Results

Analog Complexing does not need to train a model as it works using existing data to model new data samples. Therefore, it is a very fast classification method, and - with all testing samples classified correctly in this example within a few minutes (tab. 8.29) - a very accurate too (fig. 8.28).

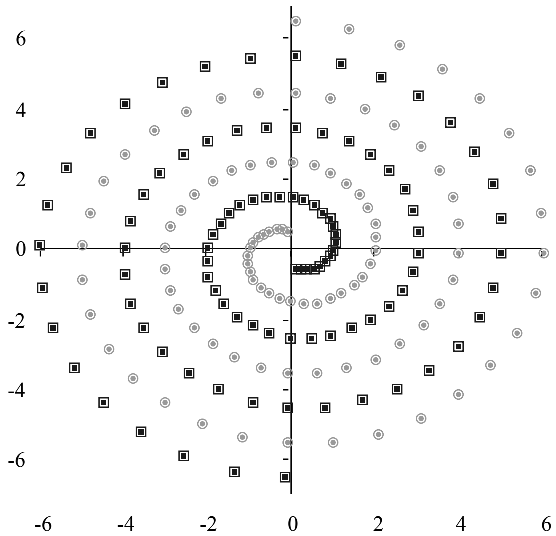

Fig. 8.28: Classified testing samples using Analog Complexing

	noise free testing samples	noisy testing samples
false classified	0	6
average similarity of the best pattern [%]	99.99	99.80
classification accuracy [%]	100.0	93.8

Table 8.29: Classification results using Analog Complexing

Fig. 8.29: Classification result using TACOMA NN learning method
(http://zarnow.gfai.de/twsp.htm)

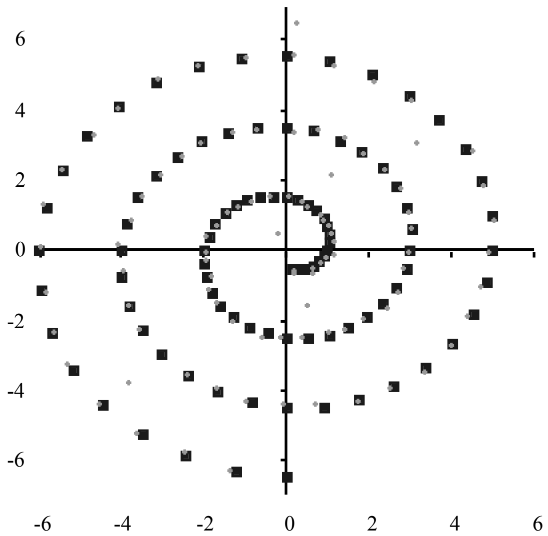

Fig. 8.30: Classification results: Training samples vs. classified noisy testing samples

Figure 8.29 shows the result of a NN model that uses TACOMA (http:// zarnow.gfai.de/tacoma.htm), an advanced NN learning algorithm. Figure 8.30 displays the classification result of Analog Complexing for one spiral when we added 5% uniform distributed noise to the X and Y coordinates of the testing data samples (tab. 8.29). This classification task extends the initial task of identifying identical but inverse patterns of two spirals by assigning new points having no exact equivalents in the learning data set to the corresponding spiral.

Summary

This example has shown that Analog Complexing is a very fast working and comfortable method for classification tasks. The difference between AC and Neural Networks is that AC does not generate a continuous valued classification model from a set of training data samples, but works as discrete mapping of new cases and known data representations of the objects, instead. The advantage of this 'similar cases' approach is that it does not make any declaration about unknown cases except their similarity to existing cases of the learning data set. When dealing with ill-defined systems, this may be of particular interest, because this method supports the way humans usually think and make decisions: Concluding from given facts. In this sense, this technology, along with the other self-organising modelling technologies, has strong potential to really function as what data mining claims to do: Giving some objective, timely decision aid on all-day problems by extracting hidden knowledge from data.