휴리스틱 진화 알고리즘을 이용한 클러스터링 알고리즘

류정우, 강명구°, 김명원
숭실대학교 컴퓨터학과
ryu0914@channeli.net, zycrome@chollian.net, mkim@computing.soongsil.ac.kr

A Clustering Algorithm based on Heuristic Evolution Algorithm

Joung Woo Ryu, Myung Ku Kang°, Myung Won Kim
School of Computing, Soongsil University

요 약

클러스터링이란 주어진 데이터들을 유사한 성질을 가지는 군집으로 나누는 것으로 많은 분야에서 응용되고 있으며, 특히 최근 관심의 대상인 데이터 마이닝의 중요한 기술로서 활발히 응용되고 있다. 클러스터링에 있어서 기존의 알고리즘들은 지역적 최적해에 수렴하는 것과 사전에 클러스터 개수를 미리 결정해야 하는 문제점을 가지고 있다. 본 논문에서는 병렬 탐색을 통해 최적해를 찾는 진화알고리즘을 사용하여 지역적 최적해에 수렴되는 문제점을 개선하였으며, 자동으로 적절한 클러스터 개수를 결정할 수 있게 하였다. 또한 진화알고리즘의 단점인 탐색공간의 확대에 따른 탐색시간의 증가는 휴리스틱 연산을 정의하여 개선하였다. 제안한 알고리즘의 성능 및 타당성을 보이기 위해 가우시안 분포 데이터를 사용하여 제안한 알고리즘의 성능이 우수함을 보였다.

1. 서론

관찰이나 실험 등을 통해 얻은 데이터들을 분류한다는 것은 과학적 연구의 가장 기본이 되는 목표중의 하나라고 볼 수 있다. 실제 d개의 변수로 구성된 N개의 개체들은 d-차원 공간에 흩어진 N개의 점으로 생각될 수 있으며, 이들이 어떤 의미의 유사성을 가지고 군집을 이루고 있는지에 대한 정보는 다변량 자료의 구조를 이해하는 데 매우 중요한 의미를 가지고 있다. 클러스터링이란 주어진 데이터를 군집화 하는 것으로, 한 군집 내에 있는 데이터들은 유사성이 높은 반면 다른 군집에 속하는 데이터들과는 차별성이 높도록 데이터를 분류하는 것이다. 클러스터링은 특별한 정보나 배경지식 없이 데이터들 간의 주어진 척도를 이용하여 결과를 이끌어 내므로 비 교사 학습에 속하는 패턴 분류 방법으로써 현재 패턴인식, 영상처리 등의 공학분야에 널리 적용되고 있을 뿐 아니라, 최근 많은 관심의 대상이 되고 있는 데이터 마이닝분야에서 핵심기술로서 활발히 연구되고 있다. 클러스터링 알고리즘은 크게 분할적 클러스터링과 계층적 클러스터링으로 나눌 수 있다.

본 논문에서는 분할적 클러스터링 알고리즘의 문제점을 개선한 알고리즘을 제안한다. 제안한 알고리즘에서는 병렬탐색을 통해 최적의 해를 찾는 진화알고리즘을 이용하여 전역적 최적해를 찾을 뿐만 아니라 클러스터링의 특성인 '클러스터내의 유사성과 클러스터간의 차별성'을 각각 분산도와 분리도로 나타내고, 입력 데이터들의 분포에 따라 자동으로 적절한 클러스터 개수를 결정하도록 하였다. 또한 진화알고리즘이 가지고 있는 단점인 탐색공간의 확대에 따른 탐색시간의 증가를 휴리스틱 연산을 정의하여 개선하였다.

본 논문의 구성은 다음과 같다. 2장에서는 기존의 알고리즘과 문제점들을 살펴보고, 3장에서는 휴리스틱 진화알고리즘을 이용한 클러스터링 알고리즘을 제안한다. 4장에서는 몇 가지 실험을 통한 결과를 살펴보고 5장에서 결론을 맺는다.

2. 관련연구

분할적 클러스터링의 대표적인 알고리즘인 K-means 알고리즘[1]과 Fuzzy C-Means(FCM)[2]알고리즘은 사전에 클러스터 개수를 정해주어야 하며, 초기 클러스터의 중심 설정과 잡음에 따라 알고리즘의 성능이 민감하게 좌우되는 문제점이 있다.

이와 같은 문제점을 해결하기 위해 최근 클러스터링 알고리즘에 진화알고리즘을 적용하는 연구가 이루어지고 있다.[3][6][7][8] 그러나 진화알고리즘을 사용할 경우 탐색공간이 커짐에 따라 수렴시간이 오래 걸리는 단점을 가지고 있다.

본 논문에서는 기존의 교배연산대신 휴리스틱 연산을 사용한 진화성이 우수한 진화알고리즘을 이용하여 클러스터링 알고리즘의 문제점을 개선하였다.

본 연구는 한국과학재단 특정기초연구과제
(과제번호 : 98-0102-01-01-3)의 지원을 받았음.

3. 휴리스틱 진화알고리즘을 이용한 클러스터링 알고리즘

클러스터링 문제는 입력데이터를 포함하는 입력 공간에서 유사한 데이터들 끼리 그룹화하는 문제로 생각할 수 있다. N개의 데이터를 c개의 클러스터로 그룹화 할 수 있는 경우의 수는 식(1)과 같다.[4]

이처럼 클러스터링 문제에 있어서 최적의 클러스터를 찾는 것은 NP-complete문제로 알려져 있으며 또한 어떤 클러스터링이 최적이냐에 대한 수학적 모델이 아직 알려지지 않았다.

$$\frac{1}{c!}\sum_{i=0}^{c}\binom{c}{i}(-1)^{c-i}\,i^{N} \qquad (1)$$

본 논문에서 제안한 알고리즘은 진화알고리즘을 사용하여 클러스터링을 하였다. 진화알고리즘은 자연도태와 진화의 메카니즘(mechanism)에 기반을 둔 확률적인 탐색 알고리즘으로서 특히 최적화 문제에 효율적인 알고리즘[5]이다.

제안한 알고리즘의 흐름은 (그림1)과 같이 진화알고리즘의 흐름과 비슷하나 탐색시간을 줄이기 위해 즉, 진화속도를 향상시키기 위해 휴리스틱 연산을 사용하였다는 점에서 차이가 있다.

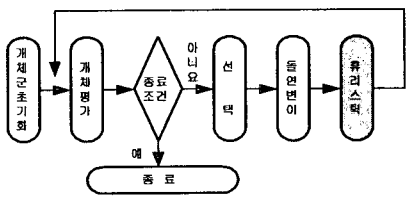


그림 1. 진화 과정

3.1. 개체군 초기화

진화알고리즘은 문제에 대한 개체 또는 후보해들의 집단인 개체군을 형성한다. 일반적으로 진화알고리즘에서의 각 개체들은 순서화된 고정길이이며, 한 유전인자를 나타내는 값들의 배열로 표현된다. 제안한 알고리즘에서는 최적의 클러스터 개수를 찾아주기 위해 각 클러스터의 중심점으로 개체를 표현하였으며, 각 개체는 가변적인 길이를 가지도록 코딩하였다.

3.2. 개체평가

진화 알고리즘에서는 각 개체의 성능을 평가하기 위해서는 적합도를 계산한다. 적합도의 관점에서 해가 될 가능성이 있는 것들을 평가하는 환경 역할을 수행하는 것이 적합도 함수이다. 따라서 진화알고리즘의 성능은 적합도 함수에 좌우되므로 문제 해의 특성을 고려한 적절한 적합도 함수를 정의하는 것이 매우 중요하다.

모든 입력데이터에 대한 각 클러스터의 소속정도를 구하기 위해 임의

의 데이터 x_j 에 대해 각 클러스터의 소속정도 u_{ij} 를 (식2)와 같이 각 클러스터 중심 v_i과의 거리의 비율로 나타낸다. 여기서 m, c는 각각 클러스터링의 퍼지정도와 클러스터 개수를 의미한다.

$$u_{ij}(x_j, v_i) = \sum_{l=1}^{c} \left(\frac{\|x_j - v_i\|}{\|x_j - v_l\|} \right)^{\frac{2}{(1-m)}} \quad (2)$$

본 논문에서 S_i는 중심이 v_i인 클러스터에 소속정도가 최대인 입력데이터들의 집합으로 정의한다. 이것은 v_i을 중심으로하는 클러스터를 의미하며 N_i은 클러스터i (S_i)에 포함되어 있는 데이터의 개수를 의미한다.

이와 같이 형성된 각각의 클러스터에 대해서 중심v_i을 평균(mean)으로 하고 표준편차 벡터를 계산한다. 이는 클러스터의 특성 중 유사성에 의하여 클러스터 중심이 될 가능성이 있는 데이터의 주위에는 관련된 많은 데이터들이 분포될 가능성이 높다고 할 수 있다. 따라서 본 논문에서는 중심(v_i)에 대해서 각 차원별로 표준편차를 계산하여 데이터의 분산를 측정하였다. 계산된 모든 클러스터의 표준편차 벡터 요소를 합함으로써 클러스터 중심으로부터 데이터들이 얼마나 분산되어 있는가를 나타내는 분산도(dispersion) $disp(X, V)$을 식(3)와 같이 정의한다.

$$disp(X, V) = \sum_{i=1}^{c} \sum_{k=1}^{d} \left(\sqrt{\frac{1}{N_i} \sum_{x_j \in S_i} (x_{jk} - v_{ik})^2} \right) \quad (3)$$

적합도 함수로서 분산도만을 고려할 경우 최적의 해로는 항상 각각의 입력데이터가 클러스터 중심이 되는 경우이다. 이는 기존 분할적 클러스터링 알고리즘에 사용된 목적함수을 사용했을 때와 같은 결과를 갖게된다. 따라서 클러스터 개수를 자동으로 결정하기 위해서 유사성뿐만 아니라, 클러스터간의 차별성을 동시에 고려해야만 한다. 본 논문에서는 클러스터간의 차별성을 FCM에서 고려하고 있는 소속정도를 이용하여 고려하였다. 즉, 각각의 클러스터에 포함된 데이터에 대한 평균 소속정도를 합한 것을 분리도(separation) $sep(X, V)$으로 식(4)과 같이 정의하였다.

$$sep(X, V) = \sum_{i=1}^{c} \left(\frac{N_i}{N} \right)^n \left\{ \frac{1}{N_i} \left(\sum_{x_j \in S_i} u_{ij}^m \right) \right\} \quad (4)$$

(단, $0 \le n \le 1$)

이와 같이 정의된 평가 척도를 사용하여 적합도가 작으면 작을수록 클러스터 특성을 잘 나타내는 개체로 평가되기 위해서 식(5)와 같이 적합도 함수 $fit(X, V)$를 정의하였다.

$$fit(X, V) = c^{1-l} \times \frac{disp(X, V)}{sep(X, V)} \quad (5)$$

지금까지 정의된 적합도 함수는 각 개체들의 특성을 평가하여 다음 세대의 개체집단을 선택하기 위한 척도가 된다. 본 논문에서는 다음 세대의 개체집단을 선택하기 위한 방법으로 룰렛휠(roulette wheel)방법과 최상의 개체를 보존하는 엘리트 방법(elitist model)을 사용하였다.

3.3. 연산

기존의 진화알고리즘을 이용한 클러스터링 알고리즘에서는 전통적인 교배(cross over)연산과 돌연변이(mutation) 연산을 사용하였다. 그러나 이 두 연산은 어떤 특별한 정보 없이 임의의 값에 의해 연산을 수행함으로 진화속도가 저하된다. 본 논문에서는 이처럼 맹목적인 연산을 하는 교배 연산 대신, 특정한 상황에 따라 연산을 적용할 수 있도록 휴리스틱(heuristic) 연산을 정의하여 진화속도를 향상시키고 있다. 그러나 휴리스틱 연산은 잘못된 정보를 사용하게 되면 지역적 최적해에 빠지기 쉬운 단점을 가지고 있으므로 이를 보완해 주기 위해 기존의 돌연변이 연산을 그대로 사용한다.

3.3.1. 진화 연산(Genetic Operation)

3.3.1.1. 돌연변이 (Mutation)

진화알고리즘에서 돌연변이 연산은 한 개체에서 임의로 선택된 유전인자를 임의의 가능한 다른 값으로 바꿈으로써, 현재 개체군에 존재하지 않는 새로운 개체를 생성하며 개체군의 다양성을 유지한다.

3.3.2. 휴리스틱 연산(Heuristic Operation)

휴리스틱 연산은 교배 연산과 돌연변이 연산처럼 확률 값에 근거하여 모든 개체에 적용하는 맹목적인 연산이 가질 수 있는 진화 시간 지연의 문제점을 해결할 수 있도록 정의하였다.

본 논문에서 사용하는 휴리스틱 연산은 가까이 있는 두 클러스터를 합병(merge)하고 , 큰 클러스터는 두 개의 클러스터로 분할(split)하는 개념에서 비롯된다.

3.3.2.1. 합병 연산(merge operation)

본 논문에서는 정의된 합병 연산을 적용하기 위해 두 클러스터간의 거리를 계산하여 두 클러스터의 차별성을 조사한다. 여기서 두 클러스터간의 차별성은 임계값 θ_M보다 작으면 낮다고 보고 합병 연산을 적용하여 (그림2. 왼쪽)와 같이 두 클러스터의 중심점 v_1, v_2을 합하여 한 개의 클러스터로 만든다. 여기서 임계값 θ_M는 모든 중심간의 평균거리에 상수 α를 곱한 것으로서 식(6)과 같다.

$$\theta_M = \alpha \left[\frac{2}{c(c-1)} \sum_{i=1}^{c} \sum_{j=i+1}^{c} D_E(v_i, v_j) \right], \quad 0 < \alpha \le 1 \quad (6)$$

단, $D_E(v_i, v_j) = \sqrt{\sum_{k=1}^{d} (v_{ik} - v_{jk})^2}$

3.3.2.2. 분할 연산 (split operation)

본 논문에서는 모든 클러스터의 표준편차 백터를 계산하여 클러스터 내의 유사성을 정의하였다. 즉, 클러스터의 표준편차 백터 요소들 중 임계값 θ_S보다 크면 클러스터 내의 유사성이 낮다고 판단하고 분할연산을 적용하여 (그림2. 오른쪽)와 같이 중심 v를 두 중심 v_1, v_2 로 분할한다. 분할된 중심좌표는 표준편차가 가장 큰 차원만 고려한다.

이때 임계값 θ_S는 모든 클러스터의 표준편차 백터 요소 합의 평균값에 상수 β를 곱한 값으로 식(7)과 같다.

$$\theta_S = \beta \left[\frac{1}{c} \sum_{i=1}^{c} SD(v_i) \right], \quad 1 \le \beta \quad (7)$$

단, $SD(v_i) = \frac{1}{d} \sum_{k=1}^{d} \left(\sqrt{\frac{1}{N_i} \sum_{x_j \in S_i} (x_{ik} - v_{ik})^2} \right)$

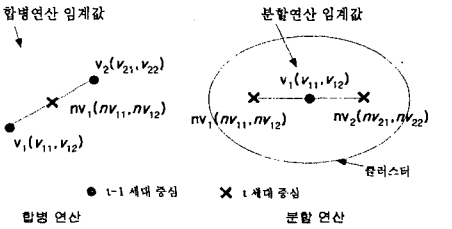


그림 2. 합병 / 분할 연산

3.3.2.3. K-means 연산 (K-means operation)[7]

돌연변이 연산, 합병 연산, 분할 연산에 의해 생성된 중심은 클러스터 내에서 정확한 중심이 아닌 대략적인 중심에 위치하고 있어 정확한 중심을 찾기 위해서 더 진화를 시켜야할 필요가 있다. 이러한 문제를 개선하기 위해서 K-means 알고리즘을 한 단계 적용한 연산을 사용하였다. K-means 연산을 통해 매 세대마다 클러스터 중심들을 입력데이터 분포에 가장 적합한 중심으로 교정함으로써 진화 속도를 개선한다.

4. 실험결과

본 논문에서는 제안한 알고리즘의 타당성을 검증하기 위해 기존의 알고리즘, 즉 K-means, FCM, Isodata 과 비교 실험하였다. 제안한 알고리즘이 전역적 최적해에 수렴하면서 가장 적합한 클러스터 개수를 자동으로 찾는 것을 시각적으로 확인할 수 있도록 2차원 가우시안 분포 데이터를 이용하여 기존의 알고리즘과 비교하였으며 다차원 가우시안 분포 데이터에서는 클러스터링 후의 클러스터 분류율을 기존의 알고리즘과 비교하였고, 생성된 클러스터 중심은 가우시안 분포 데이터의 원 중심간의 오차를 계산하여 타당한지 확인하였다. 또한 휴리스틱 연산을 사용하지 않고 교배 연산과 돌연변이 연산을 사용한 알고리즘[3]과 휴리스틱 연산을 사용한 제안한 알고리즘과의 진화속도를 비교하였다.

본 실험에서는 실험 데이터 변수로 표1과 같이 선언하였다.

표 1. 실험 데이터 변수

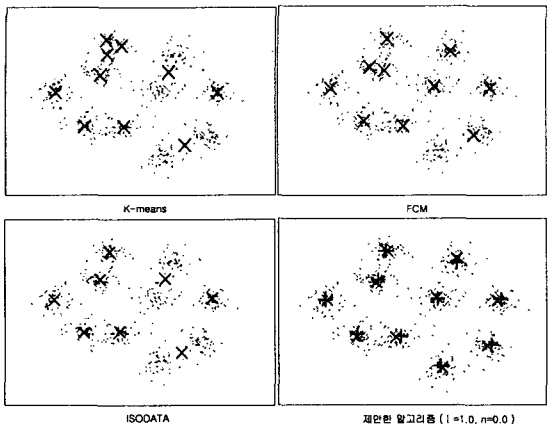
개체집단크기	30	α	0.5
돌연변이확률	0.2	β	0.5

4.1. 2차원 가우시안(Gaussian) 분포 데이터

본 데이타는 입력공간에 대한 각 차원의 영역이 [0, 10]인 2차원 가우

시안 분포 데이터로써 10개의 원 중심으로부터 가우시안 분포를 갖도록 각각 50개씩 데이터를 생성하였다. 따라서 총 500개의 데이터를 가지고 있다. 단, 원 중심은 초기 클러스터의 평균 중심은 아니다.

실험결과는 (그림3)와 같이 기존의 알고리즘은 지역적 최적해에 수렴하여 적절한 중심을 찾지 못한 반면 제안한 알고리즘은 자동적으로 클러스터 개수를 찾아낼 뿐만 아니라 적합한 클러스터링을 형성하고 있는 것을 보여주고 있다



×: 클러스터 중심좌표
+: 가우시안 분포데이터의 원 중심좌표
그림 3. 데이터C에 대한 실험결과

표2 에서는 생성된 클러스터 중심과 가우시안 분포의 원 중심간의 오차를 나타내고 있다. 오차는 입력데이터에 대한 각 차원의 최소값으로 이루어진 점과 입력공간의 대각선 거리로 나눈 상대적 거리(%)이다. 따라서 생성된 클러스터들이 적절하게 입력데이터의 특성을 표현하고 있는 것을 알 수 있다.

표 2. 중심간의 오차

	가우시안 분포데이터의 원중심	클러스터 중심	오차 (%)
클러스터1	(2.09, 2.99)	(2.13, 2.94)	0.45
클러스터2	(4.46, 2.86)	(4.26, 2.78)	1.52
클러스터3	(6.50, 5.79)	(6.45, 5.84)	0.50
클러스터4	(0.37, 5.77)	(0.40, 5.72)	0.41
클러스터5	(7.49, 8.72)	(7.47, 8.77)	0.38
클러스터6	(3.22, 7.30)	(3.90, 7.36)	1.01
클러스터7	(9.85, 5.82)	(9.86, 5.69)	0.92
클러스터8	(9.40, 2.17)	(9.20, 2.08)	1.55
클러스터9	(6.64, 0.50)	(6.69, 0.37)	0.98
클러스터10	(3.63, 9.76)	(3.72, 9.72)	0.69
평균			0.84

4.2 다차원데이터

다차원데이터는 2차원 데이터와 생성과정이 같고 단지 차원만 10차원, 20차원으로 확장하여 생성시킨 데이터로서 각 차원의 영역은 각각 [0,1], [0,10]으로 되어 있다.

표 3. 기존알고리즘과 제안한 알고리즘 비교

	10차원데이터 분류율(%)	20차원데이터 분류율(%)
K-means	80	87
FCM	88	90
제안한 알고리즘	100	100

따라서 제안한 알고리즘은 입력데이터에 대해 기존의 알고리즘과는 다르게 자동적으로 입력데이터의 분포에 따라 클러스터 개수를 결정할 수 있으며 최적의 클러스터 중심 위치를 찾을 수 있음을 알 수 있다.

또한 진화알고리즘의 단점인 탐색공간이 커짐에 따라 진화속도가 느려지는 문제점을 휴리스틱 연산을 적용하여 개선하였다. 10차원 가우시안 분포 데이터로 1000세대 실험한 결과는 (그림 4)와 같이 기존의 진화연산을 적용하였을 때보다 휴리스틱 연산을 적용하였을 때의 수렴속도가 훨씬 빠르며 지역적 최소해에서도 빨리 벗어남을 알 수 있다. 즉, (그림5)에서 보는바와 같이 휴리스틱 연산자를 사용할 경우 8세대에 수렴함을 볼 수 있으나, 일반 진화연산자를 사용할 경우 1000세대가 지나도 수렴되지 않음을 알 수 있다.

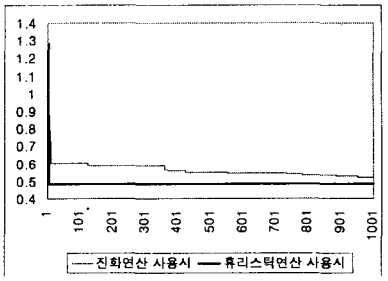


그림 5 1000세대 진화된 적합도값의 변화

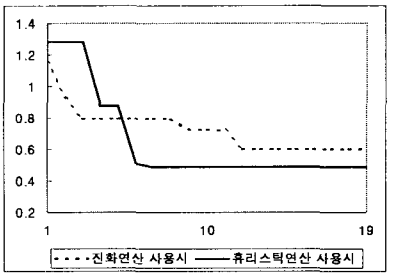


그림 6 20세대 진화된 적합도값의 변화

5. 결 론

본 논문에서는 휴리스틱 연산을 사용한 진화 알고리즘을 이용하여 자동으로 클러스터 개수를 결정하는 클러스터링 알고리즘을 제안하였다. 제안한 알고리즘은 기존의 진화알고리즘을 사용한 클러스터링의 클러스터의 수를 자동으로 찾아주며, 초기값에 민감하지 않는다는 장점은 유지하면서 문제점인 수렵속도가 느리다는 점을 개선하였다.

향후 계획으로는 기존의 분할적 클러스터링 알고리즘에서 다루기 힘들었던 기호적이나 이산적인 값을 포함한 데이터를 적용하는 부분을 연구할 계획이다. 기존의 분할적 클러스터링 방법들은 기호적인 데이터를 수치 데이터로 변환하여 사용하였으며, 그 과정에서 데이터의 성질이 왜곡되어 적절한 클러스터링이 되지 않는 단점을 가지고 있다. 이러한 단점 역시 클러스터링의 응용분야를 축소시키는 요인이 될 수 있다. 이러한 데이터들을 올바르게 분석하기 위해서는 제안한 알고리즘에 기호적 데이터를 분석할 수 있는 척도를 고려하여 보다 일반적인 클러스터링 알고리즘으로 확장되어야 할 것이다.

참 고 문 헌

[1] J. T. Tou, R. C. Gonzalez, "Pattern Recognition Principles", Addison-Wesley Publishing Company, Inc. pp. 75-109, 1974
[2] George J. Klir, Bo Yuan, "Fuzzy Sets and Fuzzy Logic", Prentice-Hall Inc. 1995
[3] 김명원, 류정우, "진화 알고리즘을 이용한 클러스터링 알고리즘", 2000봄 학술발표논문집(B) 제 27권 1호 pp. 313-315, 2000
[4] Knuth, D.(1973). The art of computer programming, vol. 1. Fundamental Algorithms of Addison-Wesley Series in Computer Science and Information Processing. Addison-Wesley, Reading, MA.
[5] Z. Michalewicz, "Genetic Algorithm + Data Structures = Evolution Programs", Third, Extended Edition, Springer-Verlag, 1995
[6] Susu Yao, "Evolutionary Search Based Fuzzy Self-Organising Clustering", Congress on Evolutionary Computation,pp. 185-188. 1999
[7] K.Krishna and M. Narasimha Murty, "Genetic K-Means Algorithm", IEEE Trans. Syst., Man, Cybern., VOL. 29, No. 3, pp. 433-439,1999
[8] G.Phanendra Babu and M. Narasimha Murty, "Clustering with Evolution Strategies", Pattern Recognition VOL 27, No.2 pp. 321-329, 1994