경영정보학연구
제11권 제1호
2001년 3월

유전자 알고리즘을 이용한 비모수 회귀분석*

김 병 도**, 노 상 규**

Nonparametric Regression with Genetic Algorithm

Kim, Byung-Do, Rho, Sang-kyu

Predicting a variable using other variables in a large data set is a very difficult task. It involves selecting variables to include in a model and determining the shape of the relationship between variables. Nonparametric regression such as smoothing splines and neural networks are widely-used methods for such a task. We propose an alternative method based on a genetic algorithm (GA) to solve this problem. We applied GA to regression splines, a nonparametric regression method, to estimate functional forms between variables. Using several simulated and real data, our technique is shown to outperform traditional nonparametric methods such as smoothing splines and neural networks.

* 서울대학교 발전기금의 일반연구비 지원에 의해 수행되었음.
** 서울대학교 경영대학 경영학과

Ⅰ. 서 론

선형회귀분석(Linear Regression)은 간단한 구조와 간결한 이론을 가지고 있기 때문에 여러 학문 또는 데이터베이스 마케팅 등의 실무 분야에서 가장 널리 쓰이는 데이터 분석 방법이다. 선형회귀분석은 독립변수와 종속변수의 관계가 선형이라고 가정한다. 그러나 그 관계가 직선으로 모형화하기 어려운 경우에 선형회귀분석을 사용하게 되면 회귀모형은 적합도가 떨어지고 추정된 모수는 편중되게 된다. 이러한 회귀분석의 문제점을 해결하기 위해 인공지능 분야의 연구자들은 인공신경망과 같은 데이터 중심의 방법을 제안하였고 통계학자들은 비모수 회귀분석(Nonparametric Regression)과 같은 데이터 중심의 방법을 제안하였다. 비모수 회귀분석은 변수간의 관계가 연속적(smooth)이라고 가정함으로써 선형모형의 제한적 구조를 극복하고 관계의 형태를 유연하게 결정할 수 있다.

이 논문에서는 비모수 회귀분석방법의 하나인 Regression Splines를 연구의 대상으로 삼았다. 비선형 관계를 찾아내는 다양한 방법들이 있지만 Regression Splines를 선택한 것은 이 방법이 간단하지만 뛰어난 아이디어에 근거하고 있기 때문이다. 이 방법은 변수간의 관계가 구간별 비선형함수(piecewise nonlinear function)라고 가정한다. 각 독립변수의 구간을 여러 개의 구분점(breakpoint 또는 knot)으로 구분하고 각 소구간 별로 비선형함수를 추정한다. Regression Splines를 추정하는데 있어 가장 중요하고도 어려운 문제가 구분점의 위치와 개수를 결정하는 것이다. 이 논문에서는 구분점을 선정하는 방법으로 처음에 많은 수의 구분점을 도입한 후 통계학적으로 일군의 구분점을 선택하는 방법을 택했다. 많은 수의 가능한 구분점 중에서 진정한 구분점을 선택하는 것이 0/1 정수계획 문제(integer programming problem)임에 착안하여 이 논문에서는 이 문제에 유전자 알고리즘(genetic alogirthm)을 적용하고자 한다.

이 논문은 Regression Splines에서 구분점의 위치와 개수를 결정하는 데 유전자 알고리즘을 적용한 것이다. Sudjianto, et al.[1996]와 노상규[1998]의 선형회귀에서의 유전적 변수선택법(genetic variable selection)을 확장, 비모수 회귀분석에 적용하여 변수선택(variable selection) 및 모형선택(model selection)을 수행하는 효율적 방법을 제안한다. 본 논문은 다음과 같이 구성되어 있다. 제2장에서는 Regression Splines를 중심으로 비모수 회귀분석에 대해 설명한다. 제3장에서는 유전자 알고리즘에 대해 간단히 설명하고 Regression Splines문제에 어떻게 적용되는가를 설명한다. 제4장과 5장에서는 유전자 알고리즘을 이용한 비모수 회귀분석을 인위적 데이터와 실제 데이터를 이용하여 평가한다. 제6장에서는 본 연구를 요약하고 향후 연구 과제를 제시한다.

Ⅱ. 비모수 회귀분석

우선 독립변수가 하나인 경우(univariate case)를 고려해 보자. n 개의 x_i 와 y_i 관측치($i = 1, 2, ..., n$)가 있고 x 와 y 의 관계는 다음과 같다고 가정한다.

$$y_i = f(x_i) + \varepsilon_i \qquad (1)$$

식 (1)에서 f는 추정하려 하는 알려지지 않은 함수이고 ε_i는 평균 0, 분산 σ^2의 표준정규분포를 따른다고 가정한다. f를 모수적(parametric)으로 추정하는 방법에서는 f의 함수유형(functional form)을 미리 가정하고 그 내에서 몇 가지의 모수를 추정하는 방법이다. 가장 잘 알려져 있는 사례가 선형회귀분석으로 $f(x_i) = \alpha + \beta x_i$로 가정하는 것이며, 여기서 α와 β는 추정되어야 할 모수이다. 하지만 진정한 관계가

선형이라는 것을 누가 확신할 수 있는가? 만약 진정한 f가 선형이 아닌 경우에 회귀모형은 부적절할 것이고 모수의 추정치는 편중될 것이다.

비모수 회귀분석에서는 f를 선형으로 가정하는 것이 아니라 f는 비선형이며 데이터에 의해 적절한 f를 결정할 수 있다고 가정한다. 이러한 비선형 함수 f를 추정하는 방법은 여러 가지가 있는데[Hastie and Tibshirani, 1990; H ≅ rdle and Turlach, 1992] 그 중 이 논문에서는 Regression Splines를 이용하여 함수 f를 추정하고자 한다[Hastie and Tibshirani, 1990; Fahrmeir and Tutz, 1994]. 다항회귀분석(polynomial regression)을 이용한 추정이 전체적(global)인데 반해 Kernel smoothing이나 running-mean 같은 방법은 추정이 부분적(local)이다. Regression Sp-lines의 경우는 구간별 다항식(piecewise polynomial)이기 때문에 추정이 전체적이면서도 부분적이다. x의 지역이 여러 개의 구분점, $\xi_1 <, \cdots < \xi_S$으로 나누어진다. 일군의 구분점이 주어지면 다항식의 차원이 결정되어져야 하고 구간별 다항식이 이 구분점에서 연결되어져야 한다. 이 논문에서는 f를 추정하기 위해 truncated power series basis[Hastie and Tibshirani, 1990 ; Smith and Kohn, 1996]를 적용하였다.

$$f(x) \approx \alpha_0 + \alpha_1 x + \alpha_2 x^2 + \alpha_3 x^3 + \sum_{i=1}^{S} \beta_i (x - \xi_i)_+^3 \quad (2)$$

위의 식 (2)에서 $(x-\xi_i)_+ = \max\{0, x-\xi_i\}$이다. 식 (2)의 함수는 모든 (ξ_i, ξ_{i+1}) 구간(subinterval)에서 3차 다항식(cubic polynomial)이고 구분점에서 연속적인 1, 2차 도함수(first and second derivatives)를 가지고 있다. 위의 식은 $(s+4)$개의 모수를 가진 선형식으로 선형회귀분석을 적용할 수 있다. 위의 Regression Splines를 적용하는데 있어 어려운 점은 smoothing의 정도에 영향을 미치는 구분점의 개수와 위치를 결정하는 것이다. 만약 구분점의 개수가 너무 적거나 위치가 부적절하다면 곡선의 자세한 부분을 잡아내지 못할 수도 있다. 하지만 구분점이 너무 많은 경우 함수를 overfit 하게 되어 모수 추정치가 큰 분산을 가지게 될 것이다. 이 문제에 접근하는 가장 합리적인 방법은 처음에는 많은 수의 가능한 구분점에서 출발하여 일군의 구분점을 통계적으로 선택하는 것이다. 즉 구분점의 선택을 변수선택의 문제로 보는 것이다.

Friedman and Silverman[1989]이 이러한 접근방법을 제안하였고 구분점은 단계별 방법(*stepwise procedure*)을 이용하여 선택하였다. Smith and Kohn[1996]은 베이지언 변수선택법을 사용하여 구분점을 선택하였다. 많은 수의 가능한 구분점 중에서 진정한 구분점을 찾는 것이 0/1 정수계획 문제임에 착안하여 이 논문에서는 유전자 알고리즘을 이용한 문제해결방법을 제안하고자 한다.

위에서 설명한 방법은 다변수(multivariate)의 경우에도 쉽게 적용될 수 있다. x_i와 $y_i (i=1, 2, \cdots, n)$의 관측치가 n개라고 가정하고 이제 각 x_i가 벡터 $(x_{1i}, \cdots, x_{1m})'$라고 가정하자. y와 벡터 x 사이의 관계는 다음과 같다고 할 수 있다.

$$\begin{aligned} y_i &= f(x_{1i}, \cdots, x_{mi}) + \epsilon_i \\ &= f_1(x_{1i}) + \cdots + f_m(x_{mi}) + \epsilon_i \end{aligned} \quad (3)$$

이와 같은 방법을 additive modeling이라 하는데 이는 m차원의(m-dimensional) 변수 x의 함수 f를 추정하는 문제를 m개의 1차원 함수 f_i를 추정하는 문제로 단순화하는 것이다. Additive modeling은 완전히 일반적이지는 않지만 여러 문제에 효과적으로 적용되어 왔다[Hastie and Tibshirani, 1987].

Ⅲ. 유전자 알고리즘과 Regression Splines

3.1 유전자 알고리즘

유전자 알고리즘은 자연에서의 생명체의 진화 개념에 기초한 효율적이며 robust한 탐색 방법(search method)으로 최적화 및 인공 학습의 여러 분야에 성공적으로 적용되어 왔다[Hou et al., 1994 ; March and Rho, 1995, Tam, 1992 ; Uckun, 1993 ; Wasserman and Sudjianto, 1988]. 유전자 알고리즘의 기본적인 개념은 다음과 같다[Goldberg, 1989 ; Davis, 1991, De Jong, 1990, Holland, 1975].

1) 생명체의 유전자에 해당하는 해의 표현(a representation of solutions)
2) 서로 다른 유전적 형질을 지닌 생명체들의 군에 해당하는 해집단(population)
3) 다음 세대에 자식을 생산할 부모를 선택하는 기준이 되는 적응도 개념(fitness)
4) 자식의 유전적 형질을 부모로부터 물려받는 유전적 연산(genetic operators)
5) 적자만이 살아남는 적자생존의 법칙(survival of the fittest)

위의 개념을 설명하기 위해 간단한 최적화 문제를 고려해 보자. 함수 $f(x) = -x^2 + 22x + 279$를 정수 간격 [0, 31]에서 최대화 하고자 한다고 가정하자. 해 x는 이진수로 5bit을 이용하여 나타낼 수 있다. 예를 들어, 01001은 9를 나타낸다.

유전자 알고리즘은 무작위로 최초 해집단을 생성함으로써 시작된다. 집단의 크기는 해공간을 충분히 샘플할 수 있을 정도로 커야한다. <표 1>은 크기 4인 예제의 최초 해집단을 보여주고 있다. 각 세대에서 해집단의 해는 적응도에 의해 평가되어진다. 예제에서는 함수 값, $f(x)$, 로 해를 평가할 수 있다. 하지만 대부분의 최적화 문제에서는 함수 값을 그대로 사용할 수 없기 때문에 적응도를 조정하여야 한다. 예를 들어 함수 값이 음이거나 최소화(minimization) 문제인 경우는 함수 값을 그대로 사용하지 않고 조정(scale)을 하게 된다. 적응도를 어떻게 정의 하는가는 유전자 알고리즘의 효율성에 큰 영향을 미친다.

<표 1> 최초 해집단(Initial Population)

해	x	f(x) (적응도)
11101	29	76
00101	5	364
01110	14	391
10100	20	319

해집단의 해를 적응도에 의해 평가한 후, 해집단 중의 몇 해는 자식(offspring)을 생산할 수 있는 부모(parent)로 선택되어진다. 이 때, 부모로 선택되어질 확률이 적응도에 비례하는 확률적 방법으로 부모를 선택한다.

선택된 부모들은 짝이 지워지고 유전적 연산에 의해 자식이 생성된다. 자식을 생성하기 위해 사용되는 대표적인 유전적 연산으로 교배(crossover)와 돌연변이(mutation)가 있다. 교배는 유전자 알고리즘에서 가장 중요한 연산이다. 교배는 두 부모에 적용하며 두 부모의 일부분을 결합하여 하나 또는 두 개의 자식을 생성한다. 교배를 하는 가장 간단한 방법(1점교배(1-point crossover))은 <그림 1.a>에서와 같이 교배점(crossover point)을 무작위로 선택하고 한 부모의 왼쪽 부분과 다른 부모의 오른쪽 부분을 연결하는 것이다. 두 번째 자식은 반대 부분을 결합함으로써 생성될 수 있다. 돌연변이는 <그림 1.b>에서와 같이 한 해의 유전형질을 무작위로 변형시키는 것이다. 돌연변이는 해공간(solution space)의 특정부분을 탐색하지 않을 확률이 0이 되지 않도록 보장하는 역할을 한다.

a. 교배(Crossover)		b. 돌연변이(Mutation)	
부모1	0 0 1 0 1		
부모2	0 1 1 1 0	부모	0 0 1 0 1
	↑		↑
	교배점		돌연변이
자식1	0 0 1 1 0	자식	0 0 1 1 1
자식2	0 1 1 0 1		

<그림 1> 유전적 연산(Genetic Operators)

새로운 해들이 유전적 연산에 의해 생성되면 이전 세대의 해들을 새로운 해(자식)들로 대체함으로써 새로운 세대(generation)를 형성한다. 이때 대체되는 해는 적응도가 떨어지는 해가 된다. <표 2>는 두 자식을 생성하여 적응도가 떨어지는 두 해를 대체함으로써 형성된 제 2세대이다. 마지막으로 유전자 알고리즘은 주어진 정지조건을 만족하면 끝나게 된다. 정지조건은 대부분의 경우 최대 세대 수이다.

<표 2> 제2세대 해집단(Second-generation Population)

해	x	f(x) (적응도)
00101	5	364
01110	14	391
00110	6	375
01101	13	396

단순하기는 하지만 위의 예는 유전자 알고리즘이 왜 효과적인가를 잘 나타내고 있다. 교배가 해를 결합하면서, 높은 적응도를 지닌 스키마(schema) 또는 부분해(partial solution)가 (예를 들어 0****, *1**1, 01***) 여러 해에 나타나기 시작한다. 평균이상의 적응도를 지닌 부모들은 좋은 스키마를 가지고 있을 것이고 그렇지 못한 부모들은 좋은 스키마를 가지고 있지 않을 것이다. 확률적 선택과정을 통해 높은 적응도를 지닌 부모들이 그렇지 못한 부모들 보다 많은 자식을 생성하게 될 것이고 세대가 지나면서 좋은 스키마의 숫자는 늘고 나쁜 스키마의 숫자는 줄어 들 것이다. 따라서 해집단의 평균 적응도는 향상되고 최적해를 찾게 될 가능성이 높아지는 것이다.

3.2 유전자 알고리즘을 이용한 기존 연구

유전자 알고리즘을 회귀분석에 적용한 기존 연구는 변수선택 문제에 국한되었다. Sudjianto et al[1996]은 선형회귀분석에서 빈번히 사용되는 단계적 회귀분석법(stepwise regression)의 대안으로 유전자 알고리즘을 이용한 변수선택법을 제안하였다. 실험결과 유전자 알고리즘을 이용한 변수선택법이 우수한 것으로 나타났다.

노상규[1998]는 단계적 회귀분석법(stepwise regression), 유전자 알고리즘(genetic algorithm)을 이용한 변수선택법 및 베이지언(Bayesian) 변수선택법을 비교·평가하였다. 실험결과 유전자 알고리즘을 이용한 변수선택법과 베이지언 변수선택법이 단계적 회귀분석법보다 우수한 것으로 나타났다. 본 연구에서는 기존 연구를 확장하여 유전자알고리즘을 변수선택 뿐 아니라 모형선택에도 적용하였다.

3.3 유전자 알고리즘을 이용한 Regression Splines

식 (2)의 Regression Splines에 적용한 유전자 알고리즘에서는 해(즉 회귀모형)를 이진수의 string으로 표현한다. j번째 위치의 값이 1이면 j번째 변수가 회귀모형에 포함되는 것이고 0이면 포함되지 않는 것을 나타낸다. 예를 들어 유전자 알고리즘을 수행한 결과가 이진 string {101010000…1}이라면 회귀모형은 다음과 같다.

$$y = \hat{\alpha}_0 + \hat{\alpha}_2 x^2 + \hat{\beta}_1 (x-\xi_1)_+^3 + \hat{\beta}_s (x-\xi_s)_+^3 .$$

본 연구에서는 *BIC* (Bayesian Information Cri-teria[Schwarz, 1978])를 각 회귀모형을 평가하는데 사용하였다. 처음에는 Adjusted R^2, *AIC*, corrected *AIC* 등 여러 평가기준을 사용하였으나 실험결과 *BIC*가 다른 기준에 비해 우수하였다. *BIC*는 다음과 같이 정의된다.

$$BIC = \log(SSE/n) + p \log n / n$$

위의 식에서 n은 표본의 수이고, p는 추정하는 모수의 수이고, *SSE*는 오차 제곱합(error sum of squares)이다.

본 연구의 유전자 알고리즘에서는 최초 해집단은 무작위로 생성된다. 각 세대에서 부모해 두 개가 적응도(fitness)와 비례하여 확률적으로 선택(i. e., stochastic selection without replacement)되고 이 부모들로부터 하나의 자식이 평등교배(uniform crossover)에 의해 생성된다. 평등교배란 전 절에서 설명한 1점 교배와는 달리 두 부모로부터 각 유전형질을 동일한 확률로 물려받는 방법이다[Syswerda, 1989]. 본 연구에서 평등교배를 사용한 이유는 본 연구에서 사용한 steady state접근법(다음 단락 참조)이 exploitive한 성격이 강하므로 explorative한 성격이 강한 평등교배를 사용함으로써 exploration과 exploitation의 균형을 유지하기 위해서다[March and Rho, 1996].

다음에는 생성된 자식이 전 세대에서 적응도가 가장 나쁜 해를 대체(i. e., Elitism)함으로써 새로운 세대를 형성한다. 이렇게 한 세대에서 하나 또는 둘의 자식을 생성하여 대체하는 방식을 steady state 접근법이라고 하며 이는 해집단이 자식들에 의해 완전히 대체되는 단순 유전자 알고리즘(simple genetic algorithm)에 비해 deterministic 한 함수의 최적화에 우수하기 때문에[Davis, 1991; Whitley, 1988] 본 연구에서 사용하였다. 본 연구의 유전자 알고리즘은 한 해가 해집단의 대부분을 차지할 때 중단하도록 하였다. 이 경우에는 세대가 지속되더라도 나은 해가 나오기 어렵기 때문이다.

Ⅳ. 시뮬레이션 데이터를 이용한 평가

이 장에서는 지금까지 설명한 유전자 알고리즘을 이용한 Regression Splines의 유용성을 시뮬레이션을 통한 인위적 데이터에 의해 평가하고자 한다. 다음의 모든 평가에서 유전자 알고리즘의 해집단 크기는 2000을 사용하였고 알고리즘은 한 해가 해집단의 99% 이상을 차지할 때 중지하였다.

4.1 변수가 하나인 경우의 비모수 회귀분석 (Univariate Nonparametric Regression)

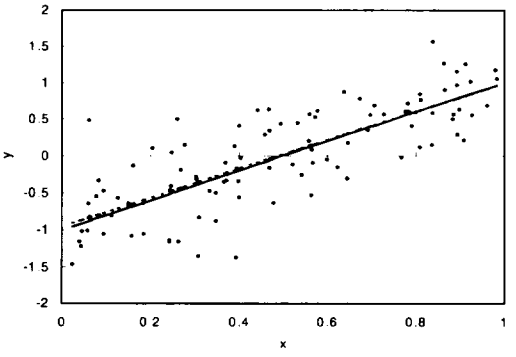
우선 유전자알고리즘을 변수가 하나인 문제에 적용해 보자. 다음의 두 함수를 이용하여 인위적 데이터를 생성하였다.

1. $y = f(x) + \varepsilon = 2x - 1 + \varepsilon$
2. $y = f(x) + \varepsilon = \sin(10\pi x) + \varepsilon$

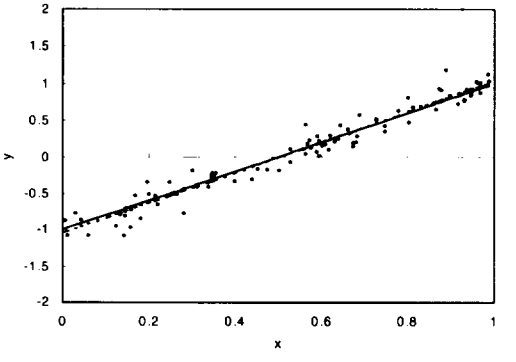
첫번째 함수는 직선이고 두번째 함수는 싸인 곡선이다. 각 함수별로 $x \sim$ Uniform(0, 1)와 $\varepsilon \sim N(0, \sigma_\varepsilon^2)$을 따르는 n = 100개의 표본을 생성하였다. 각 함수별로 두 개의 데이터 군을 생성하였다. 하나는 잡음(noise)이 큰(high) 경우이고 ($\sigma_\varepsilon = 1/2$) 다른 하나는 잡음이 적은(low) 경우이다($\sigma_\varepsilon = 1/8$).

유전자 알고리즘을 적용하여 본 결과 두 함수 모두 좋은 결과를 얻었다. 선형함수의 경우 <그림 2>에서와 같이 두 가지 잡음 수준 모두 추정된 함수가 진정한 함수에 매우 가까웠다. 잡음이 큰 경우 함수는 $y = 1.94x - 0.95$로 추정되

었고 R^2는 0.62이었고 잡음이 적은 경우 함수는 $y=2.09x-1.05$로 추정되었고 R^2는 0.96이었다. 최초의 구분점의 개수를 4에서 40개까지 바꾸어 보았으나 결과는 같았다. 유전자 알고리즘은 절편과 선형계수만 선택하였다.



a) High Noise

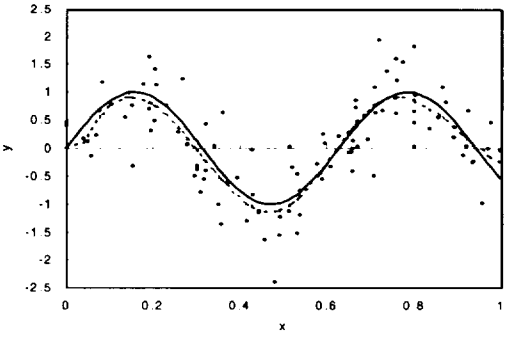


• Observation —TRUE ··· GA estimate

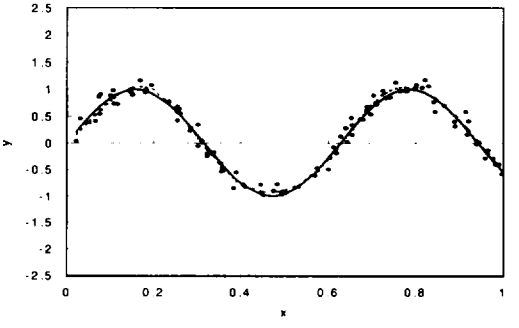
b) Low Noise

<그림 2> 선형함수 추정결과

유전자 알고리즘은 싸인곡선의 경우에도 두 가지 잡음 경우 모두 좋은 결과를 얻었다. <그림 3>에서와 같이 싸인 함수의 궤적(locus)을 추적하였는데 잡음이 큰 경우는 R^2가 0.62이었고 잡음이 적은 경우는 R^2가 0.97이었다. 잡음이 큰 경우 회귀모형은 5개의 변수(x^3와 4개의 구분점)를 포함하였고 잡음이 적은 경우도 역시 5개의 변수(x와 4개의 구분점)를 포함하였다.



a) High Noise



• Observation —TRUE ··· GA estimate

b) Low Noise

<그림 3> 싸인함수 추정결과

4.2. 다변수 비모수 회귀분석(Multivariate Nonparametric Regression)

이 절에서는 유전자 알고리즘을 보다 어려운 다변수 문제에 적용해 보고자 한다. 우선 표본수 100개인 $x_i \sim U(0, 1)$를 따르는 독립변수 10개, $x_1, \ldots, x_{10}$를 생성하였다. 그리고 다음의 함수를 이용하여 종속변수 y를 인위적으로 생성하였다.

$$y = 0.1e^{4x_1} + \frac{4}{1+\exp[-(x_2-0.5)/0.05]} + 3x_3 + 2x_4 + x_5 + \epsilon$$

위의 식에서 x_6에서 x_{10}는 포함되지 않았음을 유의하여야 한다. 이 평가의 첫번째 목적은 유전자 알고리즘이 진정한 독립변수인 x_1에서 x_5만을 선택하는가를 보는 것이다. 그 다음에는 각

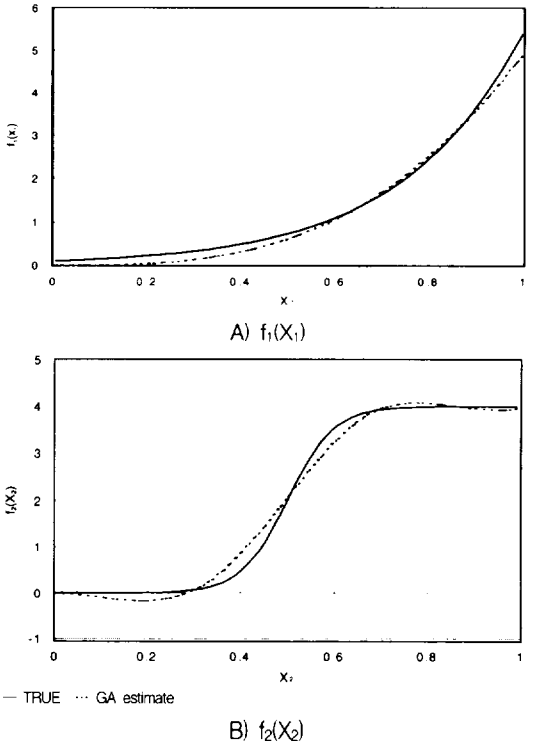
각의 진정한 독립변수와 종속변수와의 관계의 모양을 정확하게 도출해 내는가를 보는 것이다. 일변수의 경우와 마찬가지로 오차 분산이 다른 두 데이터 군을 생성하였다. 식 (3)의 가법적(additive) 가정을 사용하면 다음의 다변수 모형이 100개의 인위적 표본을 이용하여 추정된다.

$$y = \alpha_0 + \sum_{i=1}^{10}(\alpha_{i1}x_i + \alpha_{i2}x_i^2 + \alpha_{i3}x_i^3 + \beta_{i1}(x_i - \xi_{i1})^3 + \beta_{i2}(x_i - \xi_{i2})_+^3) + \varepsilon \quad (4)$$

<표 3> 다변수 함수의 해

Variable	High Noise						Low Noise					
	α_0	α_{i1}	α_{i2}	α_{i3}	β_{i1}	β_{i2}	α_0	α_{i1}	α_{i2}	α_{i3}	β_{i1}	β_{i2}
Intercept	0						0					
x_1		0	0	1	0	0		1	0	0	1	0
x_2		0	1	1	1	1		1	1	1	1	1
x_3		1	0	0	0	0		1	0	0	0	0
x_4		1	0	0	0	0		1	0	0	0	0
x_5		1	0	0	0	0		0	1	0	0	0
$x_{6\text{-}10}$		0	0	0	0	0		0	0	0	0	0

유전자 알고리즘은 다변수 함수의 경우에도 효과적이었다. 잡음이 큰 경우와 적은 경우의 R^2가 각각 0.9667과 0.9966이었다. 잡음이 큰 경우 유전자 알고리즘은 5개의 진정한 독립변수를 선택했다(<표 3> 참조). 결과를 자세히 분석한 결과 유전자 알고리즘은 각각의 독립변수 수준에서도 정확하게 함수 관계를 찾았다. x_1와 x_2의 경우 비선형 함수를 <그림 4>에서처럼 매우 유사하게 추정하였다. x_3에서 x_5의 경우는 정확하게 종속변수와의 관계를 선형으로 추정하였다. 잡음이 적은 경우에도 <표 3>과 <그림 5>에 나타난 것처럼 유사한 결과가 나타났다. 단지 x_5의 경우 선형인데도 유전자 알고리즘은 비선형(자승)으로 추정하였다. <표 3>을 보면 잡음이 큰 경우와 작은 경우의 결과가 다른 것처럼 보이나 사실 그렇게 다르지 않다. 첫째, 시뮬레이션의 가장 핵심 사항으로 두 조건 모두에서 유전자 알고리즘은 x_6부터 x_{10}까지의 허구 변수들을 하나도 선택하지 않았다. 둘째, 변수 x_3과 x_4의 경우 두 시뮬레이션 조건 모두에서 선형으로 올바르게 추정되었다. 사실 두 시뮬레이션에서 차이점이 있다면 10개의 독립변수 중 x_1, x_2, x_5의 세 독립변수에 해당되는 문제이다. 그러나 이 경우도 변수 선택에 있어서 약간의 차이가 있었지만, 선택된 변수에 대한 계수(coefficient) 추정이 이루어지기 때문에 최종적으로 추정된 모형의 모습은 매우 유사하다(<그림 4>와 <그림 5> 참조). 100개의 표본을 가지고 10개 중 5개의 진정한 변수를 찾아내서 함수의 형태를 추정하였고 그 중 2개가 비선형인 것을 감안할 때 유전자 알고리즘은 매우 우수하다 할 수 있다.

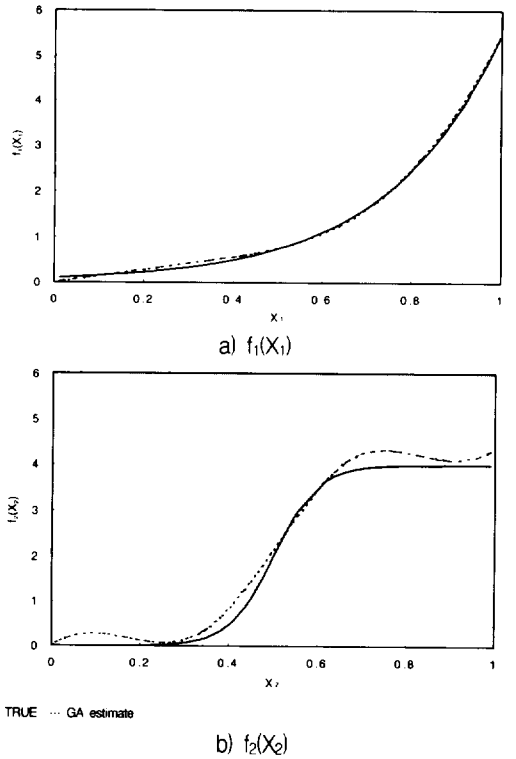


A) $f_1(X_1)$

B) $f_2(X_2)$

<그림 4> 다변수 함수 추정결과: 잡음이 큰 경우

마지막으로 이 논문에서 제안한 유전자 알고리즘을 전통적인 방법인 Smoothing Splines[Has



<그림 5> 다변수 함수 추정결과: 잡음이 적은 경우

tie and Tibshirani, 1990]와 인공신경망[Fausett, 1994]과 비교해 보았다. Smoothing Splines는 통계학자들이 새로운 비모수 회귀분석 모형을 비교하는 벤치마크로 주로 사용되는 비모수 회귀분석법이다. Smoothing Splines는 다음의 최적화 문제의 해로부터 도출될 수 있다[Hastie and Tibshirani, 1990] 1994).

$$\sum_{i=1}^{n}\{y_i - f(x_i)\}^2 + \lambda \int_a^b \{f''(t)\}^2 dt \qquad (5)$$

위 식에서 λ는 고정된 값이고 $a \le x_1 \le \cdots \le x_n \le b$이고, smoothing spline f는 위 식을 최소화하는 f를 찾는 문제로 요약될 수 있다. 위 식에서 첫 번째 항은 실제 데이터와 모형이 예측한 값이 얼마나 차이가 나는가를 측정하는 값으로 일반 회귀분석의 오차 자승합(error sum of squres)에 해당한다. 반면 두 번째 항은 비선형 함수 f의 굴곡(curvature) 정도를 제어하는 항으로 만약 이 크다면 굴곡이 별로 없는 비선형 함수가 도출될 것이고 λ이 적다면 굴곡이 매우 심한 비선형 함수가 도출될 것이다. 위 식을 최소화하는 문제에 있어 λ의 값을 결정하는 방법에는 여러 접근법이 있으나 본 논문에서는 통계학자들 간에 널리 쓰이는 cross-validation 기법을 사용하였다. 즉 아래에 제시된 목적함수를 최소화하는 λ 값을 선택한다는 것이다.

$$CV(\lambda) = \frac{1}{n}\sum_{i=1}^{n}\left\{y_i - \widehat{f_\lambda}^{i}(x_i)\right\}^2 \qquad (6)$$

위 식에서 $\widehat{f_\lambda}^{i}(x_i)$은 i번째 데이터를 제외한 후 추정된 모형의 예측치를 의미한다.

인공신경망은 생물학적 신경망을 수리적 모형으로 일반화한 인공지능(Artificial Intelligence)의 한 분야로[Fausett, 1994] 다양한 예측문제에 성공적으로 적용되어왔다. 인공신경망은 2-layer feed-forward backpropagation network을 사용하였다. 인공신경망의 경우 표본의 수가 적기 때문에 적합한 모형을 도출하기가 어려웠다. 따라서 다양한 transfer 함수, 다양한 개수의 hidden node 등을 시도하여 가장 우수한 결과를 나타낸 모형을 비교에 이용하였다. 비교에 사용된 모형의 hidden node는 5개, transfer function은 sigmoid함수이다. Learning rate, momentum term 및 tolerance는 각각 0.45, 0.9, 0.1을 사용하였다.

각 방법의 성과를 측정하기 위해 *Square Root of MSE (RMSE)* $= \sqrt{\sum_{i=1}^{n}(f(x_i) - \hat{f}(x_i))^2/n}$를 사용하였다. 비교결과 유전자알고리즘이 Smoothing Splines와 인공신경망보다 우수한 것으로 나타났다(<표 4> 참조). 선형함수에서는 유전자알고리즘과 Smoothing Splines가 같은 결과를 가져온 반면 인공신경망은 좋지 않은 결과를 가

져왔다. 싸인함수에서도 유사한 결과가 나타났다. 유전자알고리즘과 Smoothing Splines는 유사한 결과를 나타냈고 인공신경망은 좋지 않은 결과를 가져왔다. 다변수 함수의 경우에는 유전자 알고리즘이 Smoothing Splines와 인공신경망보다 잡음이 큰 경우와 작은 경우 모두 우수한 결과를 나타내었다.

<표 4> 시뮬레이션 데이터 비교결과(RMSE)

Method	Linear		Sine		Multivariate	
	High	Low	High	Low	High	Low
GA	0.0245	0.0258	0.1246	0.0426	0.2348	0.1306
Smoothing Splines	0.0245	0.0258	0.1265	0.0307	0.2843	0.1890
Neural Network	0.0792	0.0639	0.5549	0.3652	0.4542	0.6101

V. 실제 데이터를 이용한 평가

이 장에서는 유전자 알고리즘을 이용한 Regression Splines의 유용성을 실제 데이터를 이용하여 평가하고자 한다. 평가에 사용된 데이터는 기업의 연 통신비 데이터로 표본수는 7,636이고 7개의 독립변수를 포함한다. 변수에 대한 설명은 다음과 같다.

$Y(TEL_{it})$ = 기업 i의 t 년도 통신비용
$X_1(TEL_{i\ t-1})$ = 기업 i의 $(t-1)$년도 통신비용
$X_2(DURATION_i)$ = 기업i의 존속년수
$X_3(CAPITAL_i)$ = 본사가 수도에 위치하면 1, 그렇지 않으면 0
$X_4(EMP_{it})$ = 기업 i의 t 년도 종업원수
$X_5(SALE_{it})$ = 기업 i의 t 년도 매출액
$X_6(RND_{it})$ = 기업 i 의 t 년도 매출액 대비 R&D 투자비율
$X_7(PROFIT_{it})$ = 기업 i 의 t 년도 수익률

데이터를 양분하여 추정에 3,818 관측치(estimation sample)를 사용하였고 나머지 3,818 관측치(validation sample)를 모형의 예측력을 검증하는데 사용하였다. 모든 변수는 모형을 추정하기 전에 정규화 하였다. 0/1 변수인 X_3만 제외하고 Y와 각 독립변수의 관계는 비선형으로 모형화 하였다. 따라서 다음과 같은 회귀모형을 추정하고자 한다.

$$Y = \beta_0 + f_1(X_1) + f_2(X_2) + \beta_3 X_3 + f_4(X_4) + f_5(X_5) + f_6(X_6) + f_7(X_7) \quad (5)$$

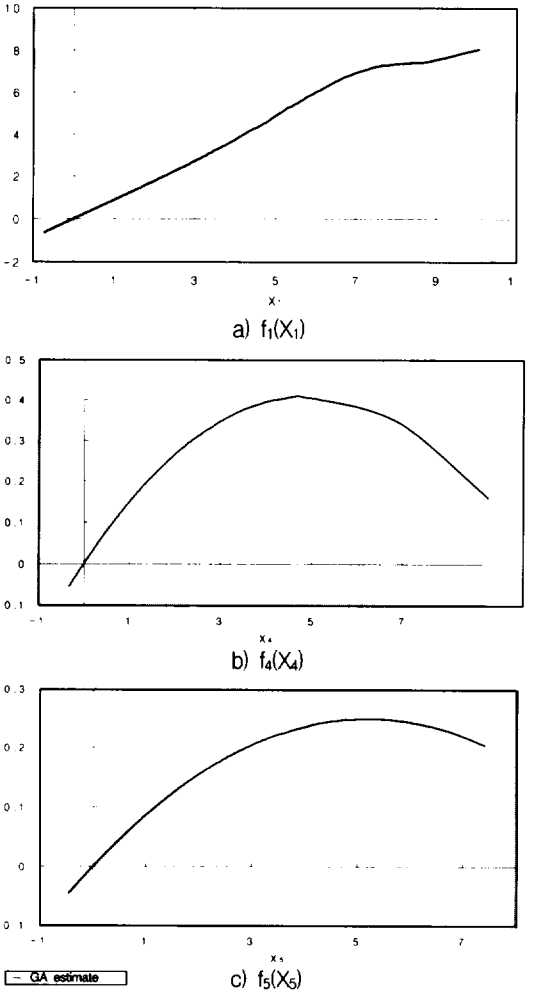
비선형 변수의 경우 5개의 구분점을 사용하였다. 비선형성을 모델링하기에 5개의 구분점이 충분한 것으로 나타났다. 실제 5개보다 많은 구분점을 사용한 경우에는 $(X'X)$매트릭스가 거의 singular하기 때문에 최소자승법이 실행되지 않는 경우가 많았다.

추정결과는 <표 5>에 요약되어 있다. 우선 절편과 X_3는 회귀모형에서 제외되었다. 두 번째로 독립변수 X_2, X_6 및 X_7과 종속변수 Y와의 관계는 선형으로 추정되었다. 세 번째로 X_1, X_4 및 X_5와 Y와의 관계는 비선형으로 추정되었다(<그림 6> 참조). 선형회귀모형을 사용하였다면 회귀모형을 제대로 추정할 수 없었을 것이다.

<표 5> 통신비 데이터의 해

Variable	α_0	α_1	α_2	α_3	β_1	β_2	β_3	β_4	β_5
Intercept	0								
x_1		0	0	1	0	1	1	0	0
x_2		1	0	0	0	0	0	0	0
x_3		0	-	-	-	-	-	-	-
x_4		1	1	0	1	1	0	0	0
x_5		1	1	0	0	0	1	1	1
x_6		1	0	0	0	0	0	0	0
x_7		1	0	0	0	0	0	0	0

마지막으로 유전자 알고리즘을 Smoothing Sp



<그림 6> 통신비 데이터 추정결과

lines, 인공신경망 및 polynomial regression[1] 과 비교해 보았다. 위에서 언급하였듯이 3,818 표본을 추정에 사용하였고 나머지 3,818 표본을 검증에 사용하였다. <표 6>에 요약되었듯이 두 표본 모두에서 유전자 알고리즘이 Smoothing Splines, 인공신경망, polynomial regression보다 우수하였다. 추정샘플(estimation sample)에서의 차이는 적지만 예측샘플(validation sample)에서 유전자 알고리즘은 Smoothing splines에 비해 5 퍼센트 이상 RMSE(root mean squared error)가 적다. 모형의 평가는 예측샘플에서 비교하는 것이 타당하며 Smoothing splines 모형이 매우 유연한 비선형 모형이라는 점을 고려할 때 5퍼센트 정도의 예측력 차이는 큰 차이라고 할 수 있다.

<표 6> 통신비 데이터 비교결과(RMSE)

Method	Estimation Sample	Validation Sample
GA	0.2657	0.2809
Smoothing Splines	0.2701	0.2933
Neural Networks	0.2791	0.3581
Polynomial Regression	0.2793	0.2977

VI. 결 론

이 연구에서는 유전자 알고리즘을 비모수 회귀분석중의 하나인 Regression Splines에 적용하였다. 즉 유전자 알고리즘을 변수선택의 문제뿐만 아니라 모형선택의 문제에도 적용한 것이다. 인위적 데이터와 실제 데이터를 이용한 평가에서 유전자 알고리즘은 전통적인 비모수 회귀분석 방법인 Smoothing Splines나 인공신경망보다 우수한 결과를 나타냈다.

향후 연구는 다양한 방향으로 진행될 것이다. 우선 이 연구에서는 독립변수간의 상호작용이 없는 것으로 가정하였다. 그러나 인공신경망과 같은 데이터마이닝 기법은 독립변수간의 상호작용을 설명할 수 있다[Berry and Linoff, 1997]. 향후에는 이 연구에서 제안한 유전자알고리즘을 독립변수간의 상호작용을 고려하도록 향상시키는 연구를 진행할 것이다. 진행되어야 할 것은 이 연구의 알고리즘을 각 변수간의 상호작용이 존재하는 경우를 고려하도록 향상시키는 것이다. 마지막으로 유전자 알고리즘을 현실 데이터에 적용해 보는 것이다. 두 번째로 다른 유전자 알고리즘과 마찬가지로 이 유전자알고리즘의 통계학적 효율성을 정확히 알지 못하는 것이 사실

1) 7개의 각 독립변수의 선형(linear) 항과 자승(square) 항을 포함한 polynomial regression 모형을 사용하였다.

이다. 따라서 다양한 상황에서 보다 많은 시뮬레이션이 필요하다. 마지막으로 이 연구에서는 대표적인 비모수 회귀분석 방법인 Smoothing Splines와 인공신경망과 유전자알고리즘을 비교하였는데 향후 연구에서는 유전자프로그램, Kernel Smoothing과 같은 다양한 방법과 비교하고자 한다.

〈참 고 문 헌〉

[1] 노상규, "유전자 알고리즘을 이용한 변수선택법," 경영논집, 제32권, 제4호, 1998년 12월, pp. 108-122.

[2] Berry, M. and Linoff, G., *Data Mining Techniques for Marketing, Sales and Customer Support*, John Wiley & Sons, Inc.: New York, 1997.

[3] Davis, L., ed., *Handbook of Genetic Algorithms*, Van Nostrand Reinhold, New York, 1991.

[4] De Jong, K.A., "Genetic-Algorithm-Based Learning," in Kodratoff, Y. and Michalski, R.S.(eds.), *Machine Learning*, Morgan Kaufmann Publishers, 1990, pp. 611-638.

[5] Fahrmeir, L. and Tutz, G., *Multivariate Statistical Modelling Based on Generalized Linear Models*, Springer Series in Statistics, Springer-Verlag: New York, 1994.

[6] Friedman, J. and Silverman, B., "Flexible Parsimonious Smoothing and Additive Modeling,"(with discussions) *Technometrics*, Vol. 31, No. 1, 1989, pp. 3-39.

[7] Goldberg, D.E., *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, 1989.

[8] H ≅rdle, W. and Turlach, B., "Nonparametric Approaches to Generalized Linear Models," *Springer Lecture Notes*, Vol. 78, 1992, pp. 213-225.

[9] Hastie, T. and Tibshirani, R., "*Generalized Additive Models* : Some Applications," Journal of American Statistical Association, Vol.82, 1987, pp. 371-86.

[10] Hastie, T. and Tibshirani, R., *Generalized Additive Models*, Monographs on Statistics and Applied Probability 43, Chapman & Hall : London, UK, 1990.

[11] Holland, J.H., *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor, MI., 1975.

[12] Hou, E.S.H., Ansari, N., and Ren, H., "A Genetic Algorithm for Multiprocessor Scheduling," *IEEE Transactions on Parallel and Distributed Systems*, Vol. 5, No. 2, 1994, pp. 113-120.

[13] March, S.T. and Rho, S., "Allocating Data and Operations to Nodes in Distributed Database Design," *IEEE Transactions on Knowledge and Data Engineering*, Vol. 7, No. 2, 1995, pp. 305-317.

[14] March, S.T. and Rho, S., "Characterization and Analysis of a Nested Genetic Algorithm for Distributed Database Design," *Seoul Journal of Business*, Vol. 2, No. 1, Fall 1996, pp. 85-122.

[15] Schwarz, G., "Estimating the Dimension of a Model," *Annals of Statistics*, Vol. 6, 1978, pp. 461-464.

[16] Wasserman, G.S., and Sudjianto, A., "All Subsets Regression Using a Genetic Search Algorithm," *Computers and Industrial Engineering*, Vol. 27, 1994, pp. 489-492.

[17] Whitley, D., "GENITOR: A Different Genetic Algorithm," *Proceedings of the Rocky Mountain Conference on Artificial Intelligence*, 1988.

◈ 이 논문은 2000년 8월 22일 접수하여 1차 수정을 거쳐 2001년 1월 18일 게재확정되었습니다.

◆ 저자소개 ◆

김병도 (Kim, Byung-Do)
현재 서울대학교 경영대학에 재직중이다. 서울대학교 경영대학을 졸업하고 시카고 대학에서 박사학위를 취득하였다. 서울대학에 오기 전 카네기멜론 대학에서 4년간 교수로 재직하기도 하였다. 최근의 주요 연구관심사로는 소비자 선택행위 분석, 데이터베이스마케팅, 추천모형 등 다양한 경영통계적 문제를 들 수 있다. 또한 Journal of Business & Economic Statistics, Journal of Interactive Marketing, Journal of Marketing Research, Journal of Retailing, Marketing Letters, Marketing Science, Journal of Database Marketing 등 다양한 학술지에 논문을 발표하였다.

노상규 (Rho, Sang-kyu)
서울대학교 경영학과를 졸업하고 미국 미네소타 대학에서 경영학 석사 및 박사학위를 취득하였으며 현재 서울대학교 경영학과에 재직중이다. 주요 연구분야로는 분산 데이터베이스 시스템, 객체지향 시스템 개발, 데이터 마이닝 등이 있으며 IEEE Transactions on Knowledge and Data Engineering, Information Systems, Annals of Operations Research, Journal of Database Management 등 다양한 학술지에 논문을 게재하였다.