혼합모형을 이용한
예측방법에 관한 연구

연세대학교 대학원
의학전산통계학협동과정
의학통계학전공
배 장 섭

혼합모형을 이용한 예측방법에 관한 연구

지도 변 혜 란 교수

이 논문을 석사 학위논문으로 제출함

2003년 6월 일

연세대학교 대학원
의학전산통계학협동과정
의학통계학전공
배 장 섭

배장섭의 석사 학위논문을 인준함

심사위원 _____ 인

심사위원 _____ 인

심사위원 _____ 인

연세대학교 대학원

2003년 6월 일

감사의 글

어떤 일을 마치고 난 뒤에는 항상 뒤를 돌아다보기 마련인 것 같습니다. 매번 그러했지만, 2년 남짓 되는 대학원 생활만큼 후회와 아쉬움이 남은 적은 없었던 것 같습니다.

이 논문이 완성되기까지 부족한 저에게 자상한 지도와 격려를 아끼지 않으셨던 김동기 교수님께 감사 드립니다. 논문 지도에 선 뜻 허락해주신 변혜란 교수님, 그리고 멀리 떨어져 있다는 이유만으로 자주 찾아뵙지 못했던 저에게 따뜻하게 대해 주셨던 김동건 교수님께도 감사 드립니다.

대학원 생활을 함께 보낸 여러 선후배 동기들 특히, 힘들 때마다 커피 한잔으로 서로를 위로했던 후배 봉섭이에게 고맙다는 말을 전합니다.

항상 옆에서 힘이 대주었던 누나와 매형, 그리고 힘든 순간에도 웃음을 잃지 않게 해 준 조카들에게도 고맙다는 말을 하고 싶습니다.

마지막으로 지금까지 제가 하고 싶어하는 것을 할 수 있도록 묵묵히 지켜 봐 주신 부모님께 부족하나마 감사와 사랑하는 마음을 전합니다.

차 례

그림 차례

표 차례

국 문 요 약

혼합모형을 이용한 예측방법에 관한 연구

기존의 연구에서는 DA, Logit, Probit 등과 같은 통계적 기법을 주로 사용하여 분류문제를 해결하고자 하였지만, 통계적 기법이 갖고 있는 엄격한 가정이 만족되어야 한다는 방법론적인 한계를 갖고 있었다. 따라서 최근에는 의사결정나무, 신경망 등과 같은 인공지능기법을 분류문제에 적용하는 연구가 활발히 소개되고 있다.

특히 기존에는 단일모형을 통한 해석에 초점을 둔 분석에 관심을 가졌지만, 최근에는 구조가 다소 복잡하고 해석은 어려우나 분류정확도에 초점을 둔 혼합모형을 이용한 분석이 실시되고 있다. 따라서 본 논문은 이러한 추세에 발맞추어 분류문제를 해결하기 위한 방법으로 혼합모형을 제시하고 그 성과를 살펴보고자 한다.

본 연구에서 제시하는 혼합모형은 세 가지 관점에서 바라보고 있다. 첫 번째 제시하는 혼합모형 I은 신경망모형이 가지고 있지 않은 변수선택능력을 단일모형에 의해서 해결하고자 하는 시도이고 두 번째 제시하는 혼합모형 II는 신경망모형의 부족한 부분을 의사결정나무(CART)로 보완하고자 하는 시도이다. 마지막으로 세 번째 제시하는 혼합모형 III는 개별기법의 결과를 최적의 연결가중치를 통하여 결합하고자 하는 시도이다.

제시된 혼합모형은 실제 자료에 적용되었으며 그 결과를 단일모형의 결과와 비교하였다. 실험결과 본 연구에서 제시된 혼합모형이 각각의 단일모형에 비해서 분류력이 향상되었음이 입증되었다.

핵심 되는 말 : 혼합모형(Hybrid Model), 변수선택, CART, 유전자 알고리즘(Genetic Algorithm), 신경망(Neural Network)

제 1 장 서 론

1.1 연구의 배경 및 의의

기존의 연구에서는 DA, Logit, Probit 등과 같은 통계적 기법을 주로 사용하여 분류문제를 해결하고자 하였지만, 통계적 기법이 갖고 있는 엄격한 가정이 만족되어야 한다는 방법론적인 한계를 갖고 있었다. 따라서 최근에는 귀납적 학습방법, 신경망 등과 같은 인공지능기법을 분류문제에 적용하는 연구가 활발히 소개되고 있다.

특히 기존에는 단일모형을 통한 해석에 초점을 둔 분석에 관심을 가졌지만, 최근에는 구조가 다소 복잡하고 해석은 어려우나 분류정확도에 초점을 둔 혼합모형을 이용한 분석이 실시되고 있다. 따라서 본 논문은 이러한 추세에 발맞추어 분류문제를 해결하기 위한 방법으로 혼합모형을 제시하고 그 성과를 살펴보고자 한다.

본 연구에서 제시하는 혼합모형은 세 가지 관점에서 바라보고 있다. 첫 번째 제시하는 혼합모형 I은 신경망모형이 가지고 있지 않은 변수선택능력을 단일모형에 의해서 해결하고자 하는 시도로서 단일모형을 이용한 변수선택결과를 신경망의 입력변수로 사용하는 모형을 말한다.

실제 대용량의 데이터를 이용하여 분류분석을 실시할 경우, 모형에 모든 입력변수를 사용한다는 것은 효율적이지도 않을뿐더러 적합값의 분산이 증가하여 정도(precision)가 떨어지게 되고 분류 예측력의 향상을 가져오지도 못한다. 따라서 이를 해결하기 위해서는 결국 모형에 포함할 입력변수(input variable)를 선택해야 하는 변수선택(variable selection) 문제에 봉착하게 된다.

그러나 데이터에 대한 사전 정보가 부족하거나 없을 때, 또는 변수의 수가 너무 많은 경우, 최적의 변수 군을 찾기란 매우 어렵다. 이러한 경우에 현재 빈번히 사용되고 있는 변수선택 방법인 단계적 선택법(Stepwise)과 모든 가능한 회귀(all possible regression)는 시간상, 계산상 단점을 내포하고 있고 데이터의 부분적인

정보만을 이용하는가 하면 변수가 선택된 순서에 따라 다른 결과가 나오는 등 만족할만한 결과를 가져다 주지 못하고있다.

따라서 본 논문에서는 인공지능기법인 의사결정나무(Decision Tree)와 최적화 기법인 유전자 알고리즘(Genetic Algorithm)을 이용하여 최적의 변수를 선택하는데, 특히 유전자 알고리즘은 확률적으로 탐색(stochastic search)함으로서 2^p 개의 모형을 계산해야 하는 모든 가능한 회귀의 단점을 극복하게 된다.

두 번째 제시하는 혼합모형 II는 신경망모형의 부족한 부분을 의사결정나무(CART)로 보완하고자 하는 시도로서, 이는 제1종 오류(Type I Error)와 제2종 오류(Type II Error)를 이용해서 의사결정나무와 신경망모형의 성과를 결합하는 방법을 채택하고 있다. 이것은 한 기법의 부족한 부분을 다른 기법을 통하여 보완해 보고자 하는 데에서 착안된 것이다.

세 번째 제시하는 혼합모형 III는 최적의 연결가중치를 통하여 개별기법의 결과를 결합함으로서 분류 예측력을 향상시켜 보고자 하는 것이다. 따라서 각 개별기법의 예측 결과간의 최적의 연결가중치(weight)를 구하는 것이 핵심이라 할 수 있다.

1.2 연구의 목적

본 연구는 분류분석에 있어서 혼합모형을 이용하는 방법을 시도하고 평가하고자 한다.

본 연구의 목적은 세 가지 이다. 첫째는, 제시된 혼합모형(I, II, III)이 분류 예측력의 향상을 가져오는지 살펴보는 것이고 둘째는, 각 모형의 분류 정확도를 비교하여 가장 좋은 분류 예측력을 갖는 모형을 찾는 것이며 셋째는, 정확도의 측면에서 볼 때 혼합모형 I, II, III중 어느 것이 더 좋은 모형인지 살펴보는 것이다.

이와 같은 내용을 바탕으로 기존에 사용된 단일모형보다는 혼합모형을 이용해 효율적이면서도 분류 예측력에서도 떨어지지 않는 방법을 제시하였다. 특히 혼합

모형 I에서 CART-assisted Neural Network, GA-assisted Neural Network은 각각 CART와 GA에 의해 선택된 변수를 입력변수로 하는 신경망모형으로서, 단계적 선택법에 비해 성과가 우수함을 알 수 있었다.(Kun Chang Lee, Ingoo Han and Youngsig Kwon, 1996)

제 2 장 분류 분석에 사용되는 통계적 방법들

2.1 판별분석(Discriminant Analysis)

판별분석이란 두 집단 혹은 그 이상의 집단에 대하여 얻어진 여러 개의 변수 자료를 이용하여, 각 집단의 특성을 나타내는 변수의 선형결합을 도출하고, 이를 이용하여 특정 관측치가 어느 집단에 소속되는 지를 예측하는 통계적 기법으로서 독립변수들은 다변량정규분포를 따르고 종속변수들은 사전에 알려진 명목변수라고 가정한다.

판별분석의 기본원리는 두 개 이상의 집단을 구분하는 데 있어서 그 오류를 최소화할 수 있는 변수들의 선형결합, 즉 판별함수(discriminant function)를 도출하고, 이를 이용하여 주어진 관측치를 각 집단에 분류하기 위한 분류함수(classification function)를 계산하는 것이다.

판별분석의 목적은 미리 규정된 그룹들의 판별점수의 평균들이 통계적으로 유의한 차이가 있는가를 검정하거나 판별함수를 이용하여 새로운 개체를 분류하거나 그룹간의 차이를 어떤 독립변수가 가장 많이 설명하여 주는가를 찾아내는데 있다.

2.1.1 두 다변량정규모집단의 분류

일반적으로 최적의 분류기준이란 오분류확률과 오분류비용을 모두 고려한 오분류 기대 비용(Expected Cost of Misclassification:ECM)을 최소로 하는 방법이라고 할 수 있는데, 이와 더불어 총오분류확률(Total Probability of Misclassification:TPM)을 최소로 하는 방법과 사후확률(posterior probability)를

이용한 방법도 있다.

각 집단의 공분산행렬의 동일성 여부에 따라 분류규칙은 달라지게 되므로, 먼저 공분산행렬의 동일성 검정을 통해서 공분산행렬의 동일성 여부를 확인하게 된다.

2.1.2 공분산행렬이 같은 경우($\Sigma_1=\Sigma_2=\Sigma$)

관찰값 $\mathbf{x}_0$가 부등식

$$(\bar{\mathbf{x}}_1-\bar{\mathbf{x}}_2)' \mathbf{S}_{pooled}^{-1}\mathbf{x}_0-\frac{1}{2}(\bar{\mathbf{x}}_1-\bar{\mathbf{x}}_2)' \mathbf{S}_{pooled}^{-1}(\bar{\mathbf{x}}_1+\bar{\mathbf{x}}_2)\geq \ln\left[\left(\frac{c(1|2)}{c(2|1)}\right)\left(\frac{p_2}{p_1}\right)\right] \quad (2\text{-}1)$$

단,

$$\mathbf{S}_{pooled}=\frac{(n_1-1)\mathbf{S}_1+(n_2-1)\mathbf{S}_2}{(n_1+n_2-2)} \quad (2\text{-}2)$$

을 만족하면 모집단 π_1에 할당하고 그렇지 않으면 모집단 π_2에 할당하게 된다.

2.1.3 두 집단 공분산행렬이 다른 경우($\Sigma_1\neq\Sigma_2$)

관찰값 $\mathbf{x}_0$가 부등식

$$\frac{1}{2}\mathbf{x}'(\Sigma_1^{-1}-\Sigma_2^{-1})\mathbf{x}+(\mu'_1\Sigma_1^{-1}-\mu'_2\Sigma_2^{-1})\mathbf{x}-k\geq \ln[(\frac{c(1|2)}{c(2|1)})(\frac{p_2}{p_1})] \quad (2\text{-}3)$$

단,

$$k=\frac{1}{2}\ln\left(\frac{|\boldsymbol{\Sigma}_1|}{|\boldsymbol{\Sigma}_2|}\right)+\frac{1}{2}(\boldsymbol{\mu}_1'\boldsymbol{\Sigma}_1^{-1}\boldsymbol{\mu}_1-\boldsymbol{\mu}_2'\boldsymbol{\Sigma}_2^{-1}\boldsymbol{\mu}_2)$$

을 만족하면 모집단 π_1에 할당하고 그렇지 않으면 모집단 π_2에 할당하게 된다.

이러한 분류규칙에 의하여 분류함수들이 추정되면 분류함수의 타당성 및 설명력을 검토하여야 하는데, 분류함수들의 분류력을 측정하는데 사용되는 오류율로는 최적오류율(Optimum Error Rate:OER), 실제 오류율(Actual Error Rate:AER), 겉보기 오류율(APparent Error Rate:APER)등이 있다.

$$OER=p_1\int_{R_2}f_1(\mathbf{x})d\mathbf{x}+p_2\int_{R_1}f_2(\mathbf{x})d\mathbf{x} \tag{2-4}$$

$$AER=p_1\int_{\mathcal{R}_2}f_1(\mathbf{x})d\mathbf{x}+p_2\int_{\mathcal{R}_1}f_2(\mathbf{x})d\mathbf{x} \tag{2-5}$$

$$APER=\frac{n_{1M}+n_{2M}}{n_1+n_2} \tag{2-6}$$

2.1.4 Fisher의 판별함수 (부분집단이 두 개인 경우)

피셔는 다변량 관찰값 $\mathbf{x}$를 일변량 관찰값 y로 변환하여 두 모집단 π_1, π_2에서의 표본평균 $\bar{y}_1$과 $\bar{y}_2$가 최대한 멀리 떨어져 있도록 분리하는 문제에 착안하였다. 이와 같이 일변량 자료로 변환하게 되면 다변량 자료에 비해 다루기가 훨씬 쉽게 된다. 피셔의 모집단 분리는 앞에서 소개된 판별 절차와 전혀 다른 관점에서 출발하고 있지만 그 결과는 앞 절에서 논의된 일차판별식과 같아진다. 여기서는 다변량정규분포의 가정은 필요하지 않은 반면 두 모집단의 공분산이 서로

같다는 가정이 필요하다.

첫 번째 모집단의 관찰값 $\mathbf{x}$를 변환한 일변량 관찰값을 $y_{11}, y_{12}, \cdots, y_{1n_1}$, 두 번째 모집단의 $\mathbf{x}$를 변환한 관찰값을 $y_{21}, y_{22}, \cdots, y_{2n_2}$ 라하고 이들의 표본평균을 각각 $\bar{y}_1$, $\bar{y}_2$라 하자. 표준편차에 대한 표본평균간의 상대적인 거리는

$$\text{거리} = \frac{|\bar{y}_1 - \bar{y}_2|}{s_y}$$

이다. 단, s_y는 합동추정량

$$s_y^2 = \frac{\sum_{j=1}^{n_1}(y_{1j} - \bar{y}_1)^2 + \sum_{j=1}^{n_2}(y_{2j} - \bar{y}_2)^2}{n_1 + n_2 - 2}$$

의 양의 제곱근이다.

[정리 2.1] 표본평균간의 상대적인 거리를 최대로 하는 일차변환은

$$y = \hat{l}' \mathbf{x} = (\bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_2)' \mathbf{S}_{pooled}^{-1} \mathbf{x} \tag{2-7}$$

가 되고, 이 때 거리 제곱의 최대값은

$$D^2 = (\bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_2)' \mathbf{S}_{pooled}^{-1} (\bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_2) \tag{2-8}$$

이다. 식 (2-7)에 의한 일차변환을 피셔 선형판별함수라 한다.

피셔의 선형판별함수를 이용한 할당 규칙은 어떤 관찰값 $\mathbf{x}_0$에 대해 다음 부등

식

$$y_0 = (\bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_2)' \mathbf{S}_{pooled}^{-1} \mathbf{x}_0 \geq \hat{m} = \frac{1}{2}(\bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_2)' \mathbf{S}_{pooled}^{-1}(\bar{\mathbf{x}}_1 + \bar{\mathbf{x}}_2) \quad (2\text{-}9)$$

를 만족하면 $\mathbf{x}_0$를 모집단 π_1에 배정하고 그렇지 않으면 모집단 π_2에 배정한다.

끝으로 판별분석의 가정(다변량정규분포)이 위배되는 경우에는 오분류율이 매우 커지게 되므로, 자료를 변환하여 정규분포를 따르도록 한 후 등분산성을 만족하는지 여부에 따라 선형판별함수나 이차분류 규칙을 적용하는 것이 좋다. 또한 자료의 크기가 충분히 큰 경우에는 자료를 “훈련표본(training sample)”과 “확인표본(validation sample)”으로 나누어 훈련표본은 분류함수를 구성할 때만 이용하고, 확인표본은 분류 규칙의 수행 정도를 평가하는데 이용하는 것이 바람직하다.

2.2 로지스틱 회귀분석(Logistic Regression Analysis)

로지스틱 회귀분석은 반응범주가 “성공”, “실패” 등과 같이 이항형(binary-type)인 반응변수와 설명변수의 관계를 설명하고자 하는 경우에 사용하는 분석이다. 로지스틱 회귀분석은 독립변수의 정규분포 가정이 만족하지 못한 경우나 독립변수들이 범주형과 연속형으로 이루어진 경우에 사용될 수 있기 때문에 판별분석과 비교해 좀 더 유연한 기법이라고 할 수 있는데, 일반적으로 독립변수의 정규분포 가정이 만족하지 못한 경우에는 로지스틱 회귀분석이 판별분석 보다 더 낳은 결과를 가져다 준다고 알려져 있다(Press and Wilson, 1978).

2.2.1 로지스틱 회귀모형

이항반응변수의 경우에 단순선형회귀모형을 적용하는 것은 다음과 같은 두 가지 문제점을 가지고 있다. 첫째 반응변수의 예측 값이 이항형이 아니라는 점, 둘째 반응변수가 이항형이기 때문에 베르누이(Bernoulli) 분포와 같이 이진변수(binary variable)를 가지는 분포에 의해서 모형 화되는 것이 타당한데, 선형회귀모형에서는 반응변수를 연속형인 것으로 간주되기 때문에 흔히 정규분포로 모형 화된다.

로지스틱 회귀모형은 반응변수가 이항형일 때 선형회귀모형의 이러한 단점을 극복하기 위해 확률에 대한 로짓변환(logit transformation)을 고려하여 분석하는 것으로 설명변수 $x_1, x_2, \cdots, x_p$ 에 대한 다중 로지스틱 회귀모형은 다음과 같다.

$$\log \frac{p(y=1|x_1,\cdots,x_p)}{1-p(y=1|x_1,\cdots,x_p)} = \alpha+\beta_1 x_1+\cdots+\beta_p x_p \qquad (2\text{-}10)$$

식 (2-10)로부터 추정된 회귀계수 $a, b_1, \cdots, b_p$ 를 이용하여 다음과 같이 사후확률에 대한 추정 식을 얻을 수 있다.

$$\hat{p}(y=1|x_1,\cdots,x_p) = \frac{\exp(a+b_1x_1+\cdots+b_px_p)}{1+\exp(a+b_1x_1+\cdots+b_px_p)} \qquad (2\text{-}11)$$

이렇게 얻어진 각 개체에 대한 사후확률(posteria probability)은 그 개체를 분류하기 위해 사용될 수 있다 $\{\hat{p}(y=0|x_1,\cdots,x_p)=1-\hat{p}(y=1|x_1,\cdots,x_p)\}$. 즉, 추정된 사후확률은 0과 1사이의 값을 가지게 되므로, 적절한 절단 값(cutoff value)을 정하여 이 값을 기준으로 각 개체를 분류하는 것이다.

2.2.2 로지스틱 회귀계수의 추정과 검정

로지스틱 회귀모형에서는 모수인 회귀계수를 추정하기 위해서 최대우도(maximum likelihood)방법을 이용한다. 최대우도방법이란 우도함수(likelihood function)를 최대로 하는 β값을 추정하는 것을 말하는데 자연대수로 변환시킨 우도함수는 다음과 같다.

$$L(\beta) = \ln[l(\beta)] = \sum\{Y_p \ln P_p(X) + (1-Y_p)\ln(1-P_p(X))\} \qquad (2\text{-}12)$$

β값은 식 (2-12)을 미분함으로써 구할 수 있는데, 로지스틱 회귀모형에서는 미분한 방정식이 β에 관하여 2차 방정식이 되기 때문에 generalized weighted least square method나 iterative weighted least square method와 같은 반복법(iterative method)으로 계산해야만 해답을 얻을 수 있다(Hosmer and Lemeshow, 1989).

이와 같은 방법으로 모형이 얻어지면 다음으로 모형 안에 있는 특정 변수가 통계적으로 의미가 있는지를 검정해야 되는데, 이때 사용하는 방법으로 우도비 검정(likelihood ratio test), Wald test, Score test등이 있다. 이 중 검정력이 가장 좋은 것으로 알려져 있는 우도비 검정이 많이 사용되며 가설은 다음과 같다.

$$H_0 : \beta_1 = \beta_2 = \cdots = \beta_k = 0$$
$$H_1 : \beta_i \neq \beta_j \ , \ (i \neq j \ , \ j = 1,2,3,\cdots,k)$$

귀무가설(H_0) 하에서 검정통계량(test statistics) $\triangle G$는

$$\triangle G = -2\ln\left[\frac{l(\beta)\ without\ the\ variable}{l(\beta)\ with\ the\ variable}\right] \sim \chi^2 \qquad (2\text{-}13)$$

로 정의되어 지는데, 우도비 검정(2-13)이란 결국 관심 있는 변수를 포함하지 않는 모형과 포함하는 모형의 우도 $l(\beta)$를 비교함으로써 특정 변수의 결과변수에 대한 기여도를 평가하는 방법이라고 할 수 있다.

2.3 의사결정나무(Decision Tree)

2.3.1 의사결정나무의 개요

의사결정나무는 의사결정규칙(decision rule)을 도표화하여 관심대상이 되는 집단을 몇 개의 소집단으로 분류(classification)하거나 예측(prediction)을 수행하는 분석방법이다. 의사결정나무분석의 과정 및 결과가 나무구조에 의해서 표현되기 때문에, 분류 또는 예측을 목적으로 하는 다른 방법들(신경망, 판별분석, 로지스틱 회귀분석)에 비해 분석과정을 쉽게 이해하고 설명할 수 있다는 장점을 가지고 있다.

의사결정나무분석이 유용하게 활용될 수 있는 응용분야로는 (1) 차원축소 및 변수선택 (2) 교호작용효과의 파악 (3) 범주의 병합 또는 연속형 변수의 이산화 (4) 세분화 (5) 분류 (6) 예측을 들 수 있는데, (1)~(3)은 데이터마이닝의 단계 중 '탐색'에 포함되고 (4)~(6)은 '모형화'에 포함된다고 할 수 있다. 이러한 측면에서 볼 때 의사결정나무는 광고인쇄물의 응답자 분석(direct mailing), 고객들의 신용점수화(credit scoring), 의학연구(medical research), 시장분석(market analysis), 품질관리(quality control) 등 다양한 분야에서 이용될 수 있다고 하겠다.

의사결정나무의 장점은 해석이 쉽고 회귀분석이나 판별분석과 같은 모수적 모형에서는 거의 불가능한 교호효과(interaction)를 쉽게 파악할 수 있으며 정규성(normality)이나 선형성(linearity) 또는 등분산성(equal variance) 등의 가정을 필요로 하지 않는 비모수적 방법이라는 것이다. 아울러 이상치(outlier)에 민감하지 않다는 것 또한 의사결정나무가 가지는 장점이자 강점이라 할 수 있다.

반면에 단점은 선형(linear) 또는 주효과(main effect) 모형에서와 같은 결과를 얻을 수 없다는 것과, 분석용 자료(training data)에만 의존하기 때문에 새로운 자료의 예측에서는 불안정(unstable)할 가능성이 높다는 것인데, 이것은 검증용 자료(test data)에 의한 교차타당성(cross validation) 평가나 가지치기를 통하여 해결될 수 있다. 끝으로 연속형 변수를 비연속적인 값으로 취급하기 때문에 분리의 경계점 근방에서는 예측오류가 클 가능성이 있게 되는데, 이러한 단점을 극복하기 위하여 최근에는 모수적 모형(Logit)을 의사결정나무와 결합하는 혼합 나무-로짓 모형(Hybrid Tree-Logit Model)이 연구되고 있다.(Steinberg, 1999).

의사결정나무를 형성하는 알고리즘으로는 CHAID(Kass, 1910), CART (Breiman et al., 1914), C4.5(Quinlan, 1993), QUEST 등이 있는데 분리기준(Splitting Criterion), 정지규칙(Stoping rule), 가지치기(Pruning) 등을 어떻게 지정하느냐에 따라서 서로 다른 의사결정나무가 형성되게 된다.

2.3.2 분리기준과 정지규칙(Splitting Criterion and Stoping rule)

분리기준은 하나의 부모마디로부터 자식마디들이 형성될 때, 입력변수(input variable)의 선택과 범주(category)의 병합이 이루어 질 기준을 의미한다. 즉, 어떤 입력변수를 이용하여 어떻게 분리하는 것이 목표변수의 분포를 가장 잘 구별해 주는지를 파악하여 자식마디가 형성되는데, 목표변수의 분포를 구별하는 정도를 순수도(purity) 또는 불순도(impurity)에 의해서 측정하는 것이다. 이 때 순수도란 목표변수의 특정 범주에 개체들이 포함되어 있는 정도를 의미하는데, 결국 부모마디의 순수도에 비해서 자식마디들의 순수도가 증가하도록 자식마디를 형성해 나가게 되는 것이다.

이때 사용되는 분리기준은 목표변수의 형태에 따라서 달라지게 되는데, 이산형인 경우에는 카이제곱 통계량(Chi-Square statistic), 지니 지수(Gini index), 엔트로피 지수(Entropy index) 등이 사용되고, 연속형인 경우에는 F-통계량

(F-statistics), 분산의 감소량(reduction of variance)등이 사용된다.

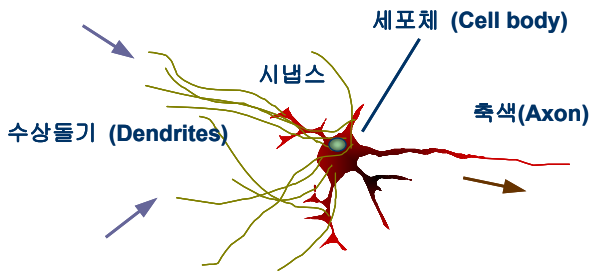
정지규칙이란 더 이상 분리가 일어나지 않고 현재의 마디가 끝마디가 되도록 하는 규칙을 의미하고 가지치기란 부모마디에 비해 분류능력(또는 순수도)이 떨어지는 마디를 제거하는 것을 의미하는데, 이는 새로운 자료에 적용할 때 예측오차(prediction error)가 매우 커질 가능성을 줄이기 위해서 사용된다.

2.4 신경망(Neural Network)

2.4.1 신경망의 개요

신경망(Neural Network)이란, 인간 두뇌의 신경망을 흉내내어 실제 자신이 가진 데이터로부터의 반복적인 학습 과정을 거쳐 데이터에 숨어 있는 패턴을 찾아내는 모델링 기법으로서 1943년 McCulloch와 Pits에 의해 최초의 신경망 모형이 제안된 뒤, 1980년대에 Hopfield에 의해 각광을 받기 시작했다.

[그림 2-1] 신경세포(Neuron)



신경망은 병렬분산처리 방식이고 연상기억(Associative memory) 능력을 가지며 학습능력이 있는 비 선형 모형이라는 특징을 가지고 있는데, 기존의 다른 방법들에 비해 다음과 같은 장점이 있다.

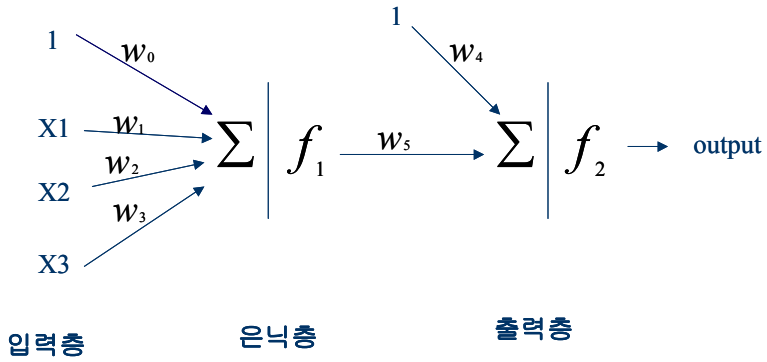
· 입력, 출력마디에 이산형, 연속형 변수 모두 사용 가능하며 기법을 적용할 수 있는 문제의 영역이 고전적인 통계기법이나 의사결정나무에 비해 넓다.
· 통계기법이 가지고 있는 가정에 민감하지 않아 로버스트(Robust)하다.
· 주어진 입력에 대해 자신의 내부 구조를 스스로 조직해 나감으로써 학습해 나가는 능력 즉, 자기조직화(self-organizing) 능력이 있다.
· 상용화된 데이터마이닝 제품이 많으며 제품 선택의 폭이 넓다.

반면에 신경망이 가진 단점은 다음과 같다.

· 분류나 예측 결과만을 제공할 뿐 결과에 대한 근거를 설명하지 못한다. 따라서 법칙의 설명이 매우 중요한 경우에는 사용될 수 없다.
· 복잡한 학습과정을 거치기 때문에 모형 구축 시 많은 시간이 소요된다. 따라서 입력변수의 수가 너무 많으면 판별분석, 로짓분석과 같은 고전적인 통계기법이나 의사결정나무, 유전자 알고리즘과 같은 인공지능기법를 이용하여 변수를 선별 후 신경망을 구축하는 대안을 고려 할 수 있다.

2.4.2 신경망의 구조

[그림 2-2] 신경망의 구조



$$\sum_1 = w_0 + \sum_{i=1}^{3} w_i x_i$$

$$f_1 = \frac{1}{1+\exp(-(w_0 + \sum_{i=1}^{3} w_i x_i))}$$

$$\sum_2 = w_4 + w_5 f_1$$

$$f_2 = \frac{1}{1+\exp(-\sum_2)} \quad : \quad Output(= \Pr edicted\ value)$$

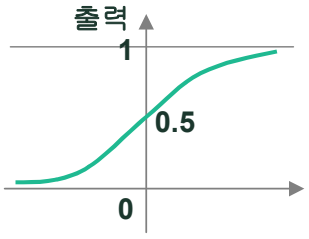
신경망은 세 개의 층 즉, 입력층(input layer), 은닉층(hidden layer), 출력층(output layer)으로 구성되어 있고 각 층은 여러 개의 노드들로 이루어져 있는데 이러한 신경망을 다층 신경망(Multi Layer Perceptron:MLP)이라고 한다. 반면에 은닉층이 없는 신경망을 단층 신경망(Single Layer Perceptron:SLP)이라고 하는데, 이 경우 일반적인 선형회귀모형과 같게 된다.

각 노드는 연결강도(weight)에 의해 연결되어 있는데, 입력신호의 강도를 나타내는 것으로 통계에서의 모수(Parameter)에 해당한다. 신경망의 목적이라 하면 결

국 최적의 연결강도(weight)를 구하는 것이라 할 수 있다.

은닉층과 출력층은 결합함수(Combination Function)와 활성함수(Activation Function)로 구성되어 있는데, 결합함수(combination function) $\sum$는 입력층 또는 은닉층의 마디들을 결합하는 형태를 의미하는 것으로 선형(Linear)함수가 사용된다. 활성함수(Activation Function) f_1과 f_2는 입력 또는 은닉마디의 결합을 변환하는 함수로서, 회귀모형인 경우에는 $f_2=x$ 가 쓰이고 분류모형인 경우에는 시그모이드 (sigmoid) 함수가 사용된다. 시그모이드(sigmoid) 함수는 로지스틱(Logistic) 함수와 유사하고 출력값은 0과 1사이의 값을 갖게 되는데, 이분형(Binary) 종속변수인 경우 0.5를 기준으로 분류를 하게 된다.

[그림 2-3] 시그모이드(sigmoid) 함수



$$f(x) = \frac{1}{1+\exp(-x)}$$

2.4.3 학습(Learning)

2.4.3.1 역전파(Back-Propagation) 알고리즘

학습(Learning)이란 노드(뉴런) 들간의 연결강도(Weight)를 조절해서 목적함수를 최적화하는 것이라고 말할 수 있다. 학습방식에 따라 다양한 학습 알고리즘이 존재하는데, 본 논문에서는 가장 많이 사용되는 방법인 역전파(Back-Propagation) 알고리즘을 사용하였다. 그 절차를 살펴보면 다음과 같다.

첫째, weight를 초기화한다.
둘째, 주어진 연결강도를 이용하여 예측 값을 계산한다.
셋째, 예측 값과 실제 값 사이의 오차를 계산한다.
넷째, 오차를 은닉층과 입력층으로 역전파 시켜서 연결강도를 새로 조절하게 되는데 오차를 최소화 할 때까지 이 과정을 반복한다. 이 때 오차는 오차제곱합 $\left(SSE = \sum(y_i - f_2)^2\right)$ 을 많이 사용한다.
단, 분류모형에서는 오차제곱합 (SSE)이 적절하지 않을 수 있기 때문에 로그우도 함수($-\sum[y_i \ln P_i + (1-y_i)\ln(1-P_i)]$) 를 사용한다.

2.4.3.2 학습 방식에 따른 신경망의 분류

학습은 크게 교사학습(지도학습 : supervised learning)과 비교사학습(비지도 학습 : unsupervised learning)으로 나누어진다. 교사학습이란 외부에서 교사신호로써 입력에 대한 정답을 출력으로 주는 학습방식을 말하며 분류(classification), 예측(prediction) 등에 유용한데 Hopfield network, Back-Propagation등이 해당된다. 반면에 비교사학습이란 평가기준은 있으나 외부에서 일일이 교사 신호를 주지 않은 학습방식을 말하며 군집화(clustering), 분리(segmentation)등에 유용한데 Adaptive Resonance Theory:ART, Self-Organized Map:SOM등이 해당된다.

2.4.4 신경망의 응용분야

신경망은 그 뛰어난 성능 때문에 다양한 산업 분야에서 응용되고 있다. 특히, 의학분야에서는 의료 진단 수단(Medical Diagnostic Aides), 생화학 분석(Biochemical Analysis), 의료 영상 분석(Medical Image Analysis), 의약품 개발(Drug Development) 등에서 활발히 사용되고 있다.

2.5 유전자 알고리즘(Genetic Algorithm:GA)

1975년 J. Holland의 논문 "Adaptation in Natural and Artificial Systems" 에서 처음 소개된 유전자 알고리즘은 생태계의 진화과정을 모방한 전역적인 최적해 탐색 기법(global optimal solution search technique)으로서, 진화가 거듭될수록 주어진 환경에 더 적합한 유전자들만이 남아있게 되는 적자 생존(survival of fittest) 이론에 바탕을 두고 있다.

유전자 알고리즘은 매우 큰공간을 탐색하는데 있어서 효과적이고 로버스트하다고 입증되었고, 특히 많은 제한에 얽매인 목적함수(objective function)를 가지는 다모수(multi-parameter) 최적화문제에 적합하다(Colin, 1992).

신경망이나 퍼지와 마찬가지로 유전자 알고리즘은 과학, 공학, 비즈니스, 사회과학 등 많은 분야에서 응용되고 있는데 기존의 다른 방법들에 비해 유전자 알고리즘이 가진 장점은 다음과 같다.

· 유전자 알고리즘은 복수의 개체 사이에서 선택이나 교배 등의 유전적 조작에 의해서 상호 협력적으로 해의 탐색을 수행한다. 따라서, 단순한 병렬적 해의 탐색과 비교하여 보다 좋은 해를 발견하기 쉽다.
· 신경망(Neural Network), 특히 역전파(Back-Propagation:BP) 알고리즘 등에서는 평가 함수의 미분값을 필요로 한다. 그러나 유전자 알고리즘에서는 현재 적응도를 분별할 수만 있으면 되기 때문에 알고리즘이 단순하고, 평가 함수가 불연속인 경우에도 적용이 가능하다.
· 어려운 비선형 문제에서 최적 해를 찾는데 적합하다.
· Local optimization을 피해갈 능력이 있는 Global optimization이다.

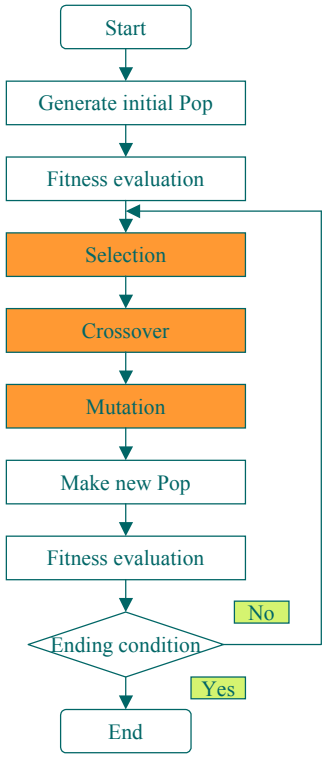
반면에 유전자 알고리즘이 가진 단점은 다음과 같다.

· 대상으로 하는 문제를 유전자 알고리즘으로 해결하기 위한 일반적인 방법이 없다.
· 개체수, 선택 방법이나 교배법의 결정, 돌연변이의 비율 등 파라미터의 수가 많다.

2.5.1 유전자 알고리즘의 연산과정

유전자 알고리즘은 풀고자 하는 문제에 대한 가능한 해들을 정해진 형태의 자료구조로 표현한 다음 이들을 점차적으로 변형함으로써 점점 더 좋은 해들을 생성하게 되는데, 각각의 가능한 해를 하나의 개체(individual)로 보며 이들의 집합을 개체군(population)이라 한다. 개체군 중에서 환경에 대한 적응도(fitness)가 높은 개체가 높은 확률로 살아남아 재생(reproduction)할 수 있게 된다.

[그림 2-4] 유전자 알고리즘 흐름도



[1 단계] 초기 개체군 생성

[2 단계] 각 개체의 적응도 평가

[3 단계] 선택

[4 단계] 교배

[5 단계] 돌연변이

[6 단계] 새로운 개체군 생성

[7 단계] 각 개체의 적응도 평가

[8 단계] 수렴성 체크

[9 단계] 반복

2.5.2 유전자 알고리즘의 구성요소

2.5.2.1 적합도 함수(fitness function)

최적화하고자 하는 함수 즉, 목적 함수(objective function)는 각 개체의 적합도를 평가하는 기반이다. 그러나 목적함수의 값의 범위는 문제마다 다르기 때문에 보통 정해진 구간 사이의 양수값을 갖도록 표준화된 값을 사용하는데, 이때 표준화되어서 실제로 개체 선택의 기준이 되는 함수를 적합도 함수 (fitness function)라고 한다.

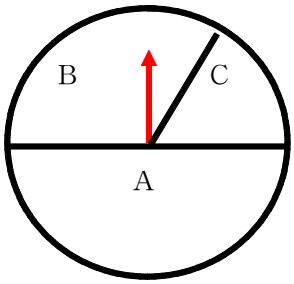
2.5.2.2 유전 연산자(Genetic Operators)

유전자 알고리즘은 기본적으로 세 가지의 연산자(operator) 즉, 선택(Selection), 교배(Crossover), 돌연변이(Mutation)로 구성되는데, 이를 유전연산자라고 한다.

2.5.2.3 선택(Selection)

선택(Selection)은 교배(crossover)를 위해 개체군(population)에서 2개의 개체를 선택하는 것을 말하는데 Roulette wheel selection 개념이 적용된다. Roulette wheel selection이란 각 개체의 적응도에 비례하는 만큼 roulette의 영역을 할당한 다음, roulette을 돌려 화살표가 가리키는 영역의 개체를 선택하는 것으로, 적응도가 높은 것(A)은 선택될 확률이 그만큼 많고 적응도가 낮은 것(C)은 선택될 확률이 상대적으로 낮게 되는 개념이다.

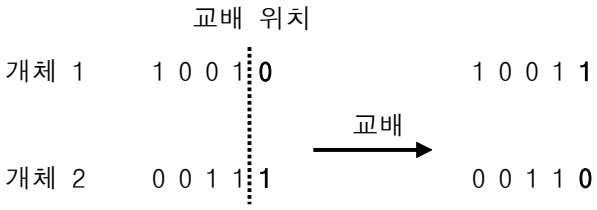
[그림 2-5] Roulette wheel



2.5.2.4 교배(Crossover)

교배(crossover)는 2개의 염색체 사이에서 유전자를 바꾸어 넣어 새로운 개체를 발생시키는 것으로, 너무 낮게 설정되면 다음 세대에서 새로운 개체 발생이 적게 되어 탐색이 침체되고 너무 높게 설정되면 탐색공간을 빨리 탐색하는 특징을 갖게 된다. 일반적으로 교배 확률은 0.8~0.95이다.

[그림 2-6] 교배(crossover) 과정

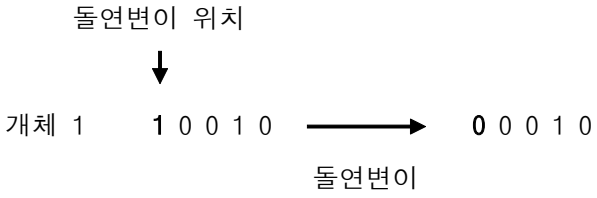


2.5.2.5 돌연변이(Mutation)

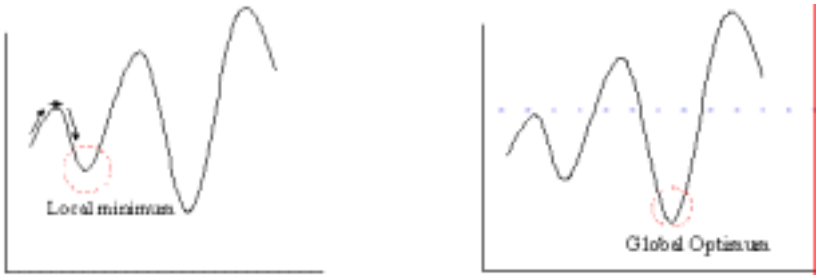
교배는 개체군 내에서의 개체 진화에 한계가 있다. 다시 말해, 주어진 환경에 어느 한계까지는 진화하여 적응할 수 있지만, 개체군내의 개체의 유전자 schema를 극복할 수는 없다. 예를 들어 11110 & 11100 두 개체가 교배를 하더라도

11111이라는 개체는 생길 수가 없는데, 이렇게 구조적인 문제점을 보완하고 혹시라도 집단에서 잃어버린 중요한 개체를 복구하여 다양성을 유지하기 위해 사용된다. 돌연변이 확률을 너무 낮게 설정하면 지역해(local solution)로 수렴할 수 있고, 너무 높게 설정하면 탐색이 불규칙적으로 점프하게 되어 불안정한 탐색 결과를 초래하게 되는 무작위 탐색(random search)이 될 수 있는데 그 결과 수렴성이 떨어져 수행 시간이 많이 걸리게 된다. 일반적으로 돌연변이 확률은 0.5~0.01이하이다.

[그림 2-7] 돌연변이(Mutation) 과정



[그림 2-8] Local Search vs Global Search



2.5.3 유전자 알고리즘의 이론적 기반 ; Schema Theorem

Schema Theorem이란 유전자 알고리즘의 수학적 배경이 되는 이론이라 할 수 있다. 이것은 스키마의 크기 등이 다음 세대에서 그 스키마를 가지는 개체들의 개수에 미치는 영향을 공식화한 것으로서 Holland에 의해서 처음으로 제안되었다. 여기서 Schema란 개체에 들어 있는 패턴을 말하는데 특별한 이진값을 가진 모든 다른 이진 열들의 집합이다. Schema는 문자 {0, 1, *} 로 만들어진 이진 열이다. "*"의 위치에는 0과1 중 어떤 것이나 올 수 있다. 예를 들어, 다음과 같은 Schema $H = 1, 0, *, 1, *$ 는 다음과 같은 4가지 이진열의 집합을 가질 수 있다.

1 0 0 1 0　　1 0 0 1 1
1 0 1 1 0　　1 0 1 1 1

Schema Theorem은 '유전자 알고리즘의 기초 정리'라고도 부르는데, 공식은 다음과 같다.

$$m(H,\ t+1) \geq m(H,\ t)\frac{f(H)}{\bar{f}}\left[1-p_c\frac{\delta(H)}{l-1}-o(H)p_m\right] \qquad (2\text{-}14)$$

$m(H,\ t)$: 세대 t에서 스키마 H를 포함하는 개체의 수

$f(H)$: 세대 t에서 스키마 H를 포함하는 개체들의 평균 적합도

$\bar{f}$: 세대 t에서 모든 개체들의 평균 적합도

p_c : 교배 확률

p_m : 돌연변이 확률

$\delta(H)$: 스키마 H의 길이

$o(H)$: 스키마 H의 차수

l : 개체의 길이

$\frac{\delta(H)}{l-1}$: 스키마가 파손될 확률

$o(H)p_m$: 스키마 H가 돌연변이 대상으로 선택될 확률

이 이론은 세대 $t+1$ 에서 존재하는 스키마 H의 개수에 대한 기대치의 하한 값을 제시하는데, 결국 스키마의 생존 가능성에 대한 하한선을 제공해 준다고 볼 수 있다. 스키마 이론에 입각하여 볼 때, 유전자 알고리즘이란 작은 스키마들이 결합하여 점점 더 큰 스키마를 이루어 가는 과정이라고 말할 수 있다.

제 3 장 모형에 관한 연구

현재까지 분류분석이 모수적인 방법으로는 판별분석과 로지스틱 회귀분석이 사용되었고 비모수적인 방법으로는 의사결정나무와 신경망이 사용되었다. 특히, 기존에는 단일모형을 통한 해석에 초점을 둔 분석에 관심을 가졌지만, 최근에는 구조가 다소 복잡하고 해석은 어려우나 분류정확도에 초점을 둔 혼합모형을 이용한 분석이 실시되고 있는데, 최근에 혼합모형이 사용된 예로는 부도예측(Kun Chang Lee, Ingoo Han and Youngsig Kwon, 1996), 사기방지(Dorronsoro, Ginel, Sanchez and Santa, 1997), 신용관련분석, 사망률에 관한 연구(Lijia, Morgan and Wang, 2000) 등이 있다.

본 연구에서 제시하는 혼합모형 이외에도 혼합 나무-로짓 모형, 혼합 나무-최인접 모형 등이 있는데, Samuel and Ciril(2002)은 혼합 나무-최인접 모형이 분류예측력 향상의 결과를 가져옴을 보였고, Steinberg(1999)는 단일모형에 비해서 혼합 나무-로짓 모형과 혼합 나무-신경망 모형의 결과가 유의적으로 좋다고 주장한 바 있다.

본 연구에서 제시하는 혼합모형은 세 가지 관점에서 바라보고 있다. 첫 번째 제시하는 혼합모형 I은 신경망모형이 가지고 있지 않은 변수선택능력을 단일모형에 의해서 해결하고자 하는 시도이고, 두 번째 제시하는 혼합모형 II는 신경망모형의 부족한 부분을 의사결정나무(CART)로 보완하고자 하는 시도이며, 세 번째 제시하는 혼합모형 III는 개별기법의 결과를 최적의 연결 가중치를 사용하여 결합하고자 하는 시도이다.

3.1. 혼합모형(Hybrid Model) I

본 연구에서 처음에 제시된 혼합모형의 의미는 단일모형을 이용한 변수선택결과를 신경망의 입력변수로 사용하는 모형을 말한다. 단일모형으로는 크게 로짓분석과 같은 통계적 기법과 의사결정나무, 유전자 알고리즘과 같은 인공지능 기법으로 나누어서 사용하였고 이렇게 선택된 변수들에 신경망 모형을 적용하여 최종적인 모형을 구축하였다.

3.1.1. Logit-assisted Neural Network

일반적으로 판별분석(DA)은 입력변수에 대해서 엄격한 통계적 가정을 요구하는 반면, 로짓분석(Logit)은 입력변수에 대한 특별한 가정이 없기 때문에 분류문제에 있어서 많이 사용되는 기법이다. Logit-assisted NN Model[1)]의 근본적인 의미는 위와 같은 장점을 가지고 있는 Logit 모형을 신경망모형에 사용될 중요한 입력변수를 선택하기 위한 사전처리기(preprocessor)로 사용한다는 것이다.

3.1.2. CART-assisted Neural Network

CART-assisted NN Model은 사전처리기로 CART를 사용하여 나중에 신경망모형에서 사용될 입력변수를 선택한다. DA, Logit의 단계적 선택법이 기하학적인 거리(geometric distance)의 관점에서 입력변수를 선택하는 반면 CART는 엔트로피(entropy)의 관점에서 입력변수를 선택한다.

---

1) Kun Chang Lee, Ingoo Han, Youngsig Kwon, "Hybrid neural network models for bankruptcy predictions"에서 사용된 용어를 인용하였다.

3.1.3. GA-assisted Neural Network

GA-assisted NN Model은 신경망모형에 사용될 중요한 입력변수를 선택하기 위한 사전처리기로 유전자 알고리즘(Genetic Algorithm)을 사용한다.

3.2. 혼합모형(Hybrid Model) II

혼합모형을 바라보는 관점은 여러 가지가 있는데 그 중 하나가 앞서 살펴본 변수선택을 위해서 개별모형을 사용한 뒤 또 다른 개별모형으로 모형을 구축하는 것이다. 지금부터는 이것과는 좀 다르게 하나의 기법이 다른 하나의 기법을 보완해 줄 수 있는 개념의 혼합모형을 소개하려고 한다.

본 연구에서 소개하는 혼합모형 II는 의사결정나무와 신경망모형을 결합한 '의사결정나무지원 신경망'(Decision Tree-Assisted Neural Network)기법으로서 이는 제1종 오류(Type I Error)와 제2종 오류(Type II Error)를 이용해서 의사결정나무와 신경망모형의 성과를 결합하는 방법을 채택하고 있다.(Lee, K.C., H. Kim and M. Kim, 1995)

3.2.1. 의사결정나무지원 신경망(DTANN)

Coakley & Brown (1991)은 신경망의 시스템오류[2] 수준을 변경시켜가며 제1종 오류(암을 정상으로 분류)와 제2종 오류(정상을 암으로 분류)간의 관계를 분석하였다. 그들의 연구결과를 살펴보면 신경망은 시스템오류 수준이 낮아짐에 따라 제2종 오류는 증가하고 제1종 오류는 감소한다는 사실을 제시하였다. 본 연구는

2) 신경망을 학습시킬 때 발생되는 오류로서 신경망에서 계산된 출력값과 학습자료의 실제값 사이의 차이를 말한다.

Coakley & Brown (1991)의 연구결과를 바탕으로 하여 신경망의 결과와 의사결정나무의 결과를 연결하여 하나의 모형을 구성하려고 한다.

[표 3-1] 경우의 수

신경망	의사결정나무	의사 결정
암	암	신경망
정상	정상	
암	정상	[표 3-2]에 따라 결정
정상	암	

1) 신경망의 결과와 의사결정나무의 결과가 동일한 경우, 의사결정나무의 결과는 신경망의 결과를 더욱 확증하여 주는 것으로 결과를 제시한다.
2) 신경망의 결과와 의사결정나무의 결과가 다른 경우, 다음과 같은 기준에 따라 신경망과 의사결정나무의 결과중 하나를 선택하여 결과를 제시한다.

① 시스템오류의 수준이 정해진 기준 이하일 때
제1종 오류발생 시 의사결정나무를 이용하여 예측을 하도록 하며, 제2종 오류 발생 시 신경망을 이용하여 예측을 하도록 한다.

② 시스템오류의 수준이 정해진 기준 이상일 때
제1종 오류발생 시 신경망을 이용하여 예측을 하도록 하며, 제2종 오류 발생 시 의사결정나무 이용하여 예측을 하도록 한다.

이를 도표화 한 것이 [표 3-2]에 제시되어 있다.

[표 3-2] DTANN모형의 의사 결정 방식

오류형태 / 시스템오류	제1종 오류	제2종 오류
시스템오류 ↓	의사결정나무	신경망
시스템오류 ↑	신경망	의사결정나무

시스템오류	신경망 결과	의사결정나무 결과	결과제시
↓	정상	암	신경망
	암	정상	의사결정나무
↑	암	정상	신경망
	정상	암	의사결정나무

본 실험에서 사용되는 시스템오류 수준은 연구에 사용되는 데이터의 특성에 맞게 분석을 통하여 적절한 값을 찾아 설정하는 것으로서 연구자의 환경에 따라 변하게 된다.

3.3. 혼합모형(Hybrid Model) III

세 번째 제시하는 혼합모형 III는 개별기법의 결과를 최적의 연결가중치(weight)를 통하여 결합하고자 하는 것으로서 최적의 연결가중치를 구하는 것이 핵심이라 할 수 있다.

개별모형으로서는 CART와 NN이 사용되었고 각 분류 기법간의 최적의 연결가중치를 찾기 위해 유전자 알고리즘(GA)이 적용되었는데, 이 유전자 알고리즘은 Visual C++ 6.0에 의해서 구현되었다.

이렇게 수행된 혼합모형의 결과는 추후 개별모형(CART, NN)의 결과와 비교를 한다.

혼합모형 III의 방법론에 대해 설명하면 다음과 같다. 각각의 개별기법을 연결할 수 있는 최적의 가중치를 훈련용 자료에서 구한 뒤, 이를 검증용 자료에 적용하여 분류력을 비교해 보는 것이다.

가중치 조절 규칙은 관측치와 예측치사이의 오차제곱합(SSE)를 최소화하는 것인데, 결과적으로 오분류 에러(Misclassification Error)를 최소화하고자 하는 것이다.

$$SSE = \sum_{i=1}^{n}(Y_i - P_i)^2 \qquad (2\text{-}15)$$

$$P_i = f\left\{\sum_{j=1}^{m} W_j p_{ij}\right\}$$

$$f(X) = \begin{pmatrix} 0 & X < 0.5 \\ 1 & X \geq 0.5 \end{pmatrix}$$

$$Y_i,\ P_i = 0\ or\ 1$$

$$0 \leq W_j,\ p_{ij} \leq 1$$

$$\sum_{j=1}^{m} W_j = 1$$

SSE : 관측치와 예측치사이의 제곱근오차, $Hit\ ratio = 100*\left\{1-\frac{SSE}{n}\right\}$

Y_i : i번째 데이터에서의 관측치

P_i : i번째 데이터에서의 예측치의 가중합을 f함수에 의해 변환된 예측치

p_{ij} : i번째 데이터와 j번째 분류방법에서의 예측치

W_j : j번째 분류방법의 가중치

f : 이진형태의 결과를 발생시키기 위한 활성함수

제 4 장 실험 연구

4.1. 실험 연구 계획

기존에 분류목적으로 흔히 사용되던 로지스틱 회귀분석, 의사결정나무, 신경망 분석을 실시한 결과를 본 연구에서 제시된 혼합모형을 사용해서 얻은 결과와 비교하고자 한다. 분류문제에 있어서의 모의실험은 정규성, 등분산성 등 피셔의 가정을 만족시키는 경우와 그렇지 못한 경우를 나누어 표본을 만들어 비교해 보는 것이 일반적이다. 그러나 앞에서도 밝혔듯이 실제로 사용되는 자료들은 이러한 가정을 만족시키지 못하는 경우가 대부분이다. 그래서 이번 모의실험에서는 모의표본을 만드는 대신 실제 자료를 이용하여 분석을 실시하여 보았다.

실험절차에 대해서 설명하면 다음과 같다. 혼합모형 I의 경우, 먼저 변수선택 방법으로서 Ordered Search(로짓분석), Heuristic Search(CART), Genetic Search(GA)를 사용하여 변수를 선택한 뒤, 이를 신경망의 입력변수로 사용하여 분류를 실시한다. 이렇게 분류 예측한 결과와 개별모형의 분류 예측 결과를 비교하게 되는데, 이러한 비교를 통해서 혼합모형이 이들 개별 기법에 비해 우수한 분류성능을 가지고 있음을 보이고자 하였고, 또한 제시된 세 개의 혼합모형간의 결과를 비교하고자 하였다. 혼합모형 II의 경우, 혼합모형 I에서 사용된 데이터를 그대로 사용해서 신경망분석과 의사결정나무분석을 한 뒤, 이 두 기법을 서로 보완한 혼합모형(의사결정나무지원 신경망)의 결과와 비교한다. 혼합모형 III의 경우 역시 동일한 데이터를 사용해서 신경망분석과 의사결정나무분석을 한 뒤, 최적의 연결가중치를 사용하여 두 기법을 통합하려고 한다. 더불어 정확도의 측면에서 볼 때 혼합모형 I, II, III중 어느 것이 더 좋은 모형인지 살펴보고자 한다.

모형이 과적합 되는 것을 막기 위해서 표본을 분석용(Train), 평가용(Validation), 검증용(Test)자료로 나누어 사용하였고 안정된 분류율을 얻기 위해서 새롭게 분할해 반복적으로 실험을 하였다.

최종적으로 이렇게 반복적으로 실험해서 얻어진 분류율을 가지고 각 방법에 대해 t-검정을 실시하여 유의적인 차이를 보이는지 확인해 보았다.

4.2. 실험 자료

본 연구에 사용된 실험자료에 대한 설명은 다음과 같다.

[표 4-1] 자료 설명

※ 간암 자료(Liver Cancer Data)

자료출처	연세대학교 의학통계학과	
자료내용	1990년 1월부터 10년 동안 연세대학교 의과대학 부속 세브란스병원 소화기 내과에 방문하여 간암 발생 위험 군으로 판단되어 정기적으로 복부 초음파검사와 혈청 검사를 포함한 검진을 받아온 환자에 대한 자료	
자료의 크기	간암 데이터 994개	
자료의 형태	연속형(3개) + 범주형(5개)	
목표변수	hepatoma (0:non-cancer, 1:cancer)	
입력변수	변수명	설명
	diag	진단명 (carrier:보균자, CH:만성간염증상, LC:간경화)
	cause	간염원인 (B, C, other)
	sex	성별 (M:남자, F:여자)
	age	연령
	afp	20이상이면 이상이 있다고 판단
	alt	40이상이면 이상이 있다고 판단
	drink	음주력 (no, social, heavy)
	par_echo	초음파 소견 (normal, moderate, severe)

※ 심장 질환 자료(Heart Disease Data)

자료출처	http://www.ics.uci.edu/~mlearn/MLRepository.html	
자료의 크기	심장 질환 자료 270개	
자료의 형태	연속형(6개) + 범주형(7개)	
목표변수	heart disease (0:absence, 1:presence)	
입력변수	변수명	
	x1	age
	x2	sex
	x3	chest pain type
	x4	resting blood pressure
	x5	serum cholestoral in mg/dl
	x6	fasting blood sugar > 120 mg/dl
	x7	resting electrocardiographic results
	x8	maximum heart rate achieved
	x9	exercise induced angina
	x10	oldpeak
	x11	the slope of the peak exercise ST segment
	x12	number of major vessels (0-3) colored by flourosopy
	x13	thal

※ 유방암 자료(Breast Cancer Data)

자료출처	http://www.ics.uci.edu/~mlearn/MLRepository.html
자료의 크기	유방암 자료 569개
자료의 형태	연속형(30개)
목표변수	diagnosis (0:malignant, 1:benign)

4.3. 연구 모형 구축

교차 타당성 평가방법을 사용하는 대신, 자료를 각각 3 : 1 : 1로 임의로 3등분하여 나누고 이를 각각 훈련용(Train), 평가용(Validation), 검증용(Test)자료로 삼았다. 보다 안정적이고 정확한 분류율을 얻기 위해서 자료를 3등분 할 때 난수를 이용하여 새롭게 3등분하고 이를 10회 반복하였다.

데이터 분할 시, 각 자료에 대해서 각 집단의 비율은 같게 구성되었는데 [표 4-2]에 제시된 바와 같다.

[표 4-2] 데이터 분할

	간암 자료			심장 질환 자료			유방암 자료		
	암	정상	총계	심장 질환	정상	총계	악성	양성	총계
훈련용 자료	54	542	596	90	72	162	128	215	341
평가용 자료	18	181	199	30	24	54	42	71	114
검증용 자료	18	181	199	30	24	54	42	71	114
총 자료	90	904	994	150	120	270	212	357	569

각각의 분석에서 로지스틱 회귀분석에서의 절단점(cut-off value), 의사결정나무와 신경망에서의 정지규칙(stop rule)을 결정하기 위해서 평가용 자료를 사용하였다.

최종분류율은 위와 같은 방법에 의하여 만들어진 모형에서의 검증용 자료의 분류율을 사용한다.

4.3.1. 혼합모형(Hybrid Model) I

4.3.1.1. 변수선택 방법

기존 연구들에서는 변수선택을 주로 단계적 선택법(Stepwise)과 같은 통계적 기법을 사용하였다. 본 연구에서는 이와 같은 방법과 더불어 CART나 GA와 같은 인공지능 기법을 활용한 변수선택 방법으로 입력변수를 선택하였다.

통계적 기법으로는 단계적 선택법(Stepwise)을 사용하여 변수를 선택하였는데, 이때 분석기법은 로짓분석(Logit)을 사용하였다. 통계적 기법을 통한 입력변수 선택에는 SPSS 11.0을 사용하였다.

인공지능 기법으로는 의사결정나무(CART)와 최적화 기법인 유전자 알고리즘을 이용하여 적응도(Fitness value)가 가장 우수한 변수로 입력변수를 선택하였다. 의사결정나무의 경우 분리기준으로 엔트로피를 사용하였고, 유전자 알고리즘의 경우 집단의 크기와 교배율, 돌연변이율은 각각 1000, 0.5 0.1로 고정시켜서 사용하였으며 가장 우수하다고 알려진 베이지안 정보기준(Bayesian Information Criterion:BIC)을 목적함수로 설정하여 이 BIC를 최소화하는 방향으로 변수를 선택하였다. 의사결정나무 분석을 위해서는 SAS E-Miner 가 사용되었고 유전자 알고리즘을 활용한 입력변수 선택을 위해서는 Evolver 4.0 이 사용되었다.

입력변수 선택을 위한 각 방법들을 정리하면 [표 4-3]과 같다.

[표 4-3] 입력변수 선택방법

선택방법		선택기준
통계적 기법 (Statistical method)	Ordered search	로짓분석(Logit)에 의해 선택된 변수
인공지능 기법 (AI method)	Heuristic search	CART에 의해 선택된 변수
	Genetic search	유전자 알고리즘(GA)에 의해 선택된 변수

4.3.1.2. 신경망 모형 구축

본 연구에서는 [표 4-3]에서 제시된 방법들을 통하여 선택된 입력변수로 신경망 모형을 구축하였다.

본 연구에서 사용한 신경망 모형은 다층 퍼셉트론(MLP)과 역전파(Back-propagation) 알고리즘으로 은닉층의 노드수는 시행착오(Trial and Error)를 통해 최적의 값을 찾는 것이 원칙이지만 은닉층의 노드수가 결과에 미치는 영향을 배제시키기 위해서 입력변수의 수와 동수를 사용하였다. 본 연구에서 사용된 모형의 출력층은 1개의 노드로 구성되며 출력값은 [0,1]의 범위에 존재한다. 신경망 실험시 모멘텀과 학습률은 모두 0.1로 고정시켜서 사용하였고 학습반복횟수는 100,000번으로 하였다. 신경망 모형의 구축을 위해 사용된 소프트웨어는 Neuro Shell V2.0 이다.

4.3.2. 혼합모형(Hybrid Model) II, III

혼합모형 I, II, II의 분류 정확도를 비교하기 위해서 데이터에서부터 옵션 설정까지 혼합모형 I과 동일하게 구성하였고 또한 원활한 자료처리를 위해 Excel을 사용하였다.

4.4. 분석 결과

4.4.1. 혼합모형(Hybrid Model) I

4.4.1.1. 변수선택 결과

[표 4-3]의 방법론에 의해 선택된 입력변수를 정리하면 [표 4-4]와 같다.

[표 4-4] 입력변수 선택방법에 따른 결과

※ 간암 자료

선택방법		결과
통계적 기법 (Statistical method)	로짓분석(Logit)	sex, age, afp, par_echo
인공지능 기법 (AI method)	CART	age, afp
	GA	age, afp

※ 심장 질환 자료

선택방법		결과
통계적 기법 (Statistical method)	로짓분석(Logit)	x2, x3, x4, x5, x7, x8, x9, x10, x12, x13
인공지능 기법 (AI method)	CART	x3, x4, x8, x12, x13
	GA	x2, x3, x10, x12, x13

※ 유방암 자료

선택방법		결과
통계적 기법 (Statistical method)	로짓분석(Logit)	x2, x7, x8, x15, x18, x19, x21, x22, x23, x30, x31
인공지능 기법 (AI method)	CART	x3, x9, x24
	GA	x8, x9, x19, x23, x24

[표 4-4] 결과에 의하면 간암 자료의 경우, 각 모형별로 선택된 변수군의 변수의 개수와 선택된 변수들의 종류가 약간의 차이를 보이고 있지만 인공지능 기법간에는 같은 양상을 보이고 있으며 모든 방법에서 AGE, AFP는 공통적 중요 입력변수로 선택되었다. 심장 질환 자료의 경우, 각 모형별로 선택된 변수군의 변수의 개수와 선택된 변수들의 종류가 확연한 차이를 보이고 있으며 모든 방법에서 x3, x12, x13은 공통적 중요 입력변수로 선택되었다. 또한 간암 자료의 경우와 마찬가지로 선택된 변수의 개수에 있어서 인공지능 기법간에 유사한 양상을 보이고 있다. 유방암 자료의 경우, 심장 질환 자료의 경우와 마찬가지로 각 모형별로 선택된 변수군의 변수의 개수와 선택된 변수들의 종류가 확연한 차이를 보이고 있다.

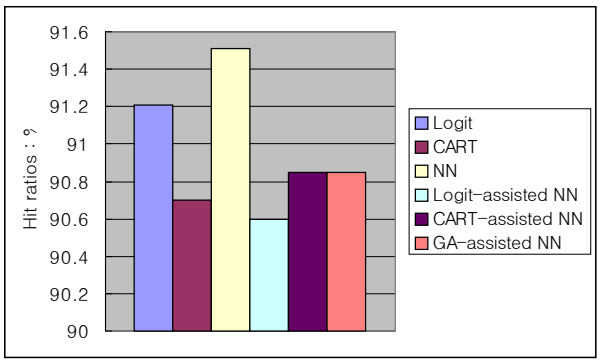
4.4.1.2. 신경망 모형 구축 결과

[표 4-3]의 방법론에 의해 선택된 변수를 입력변수로 하여 구축된 신경망 모형을 실험하였다. 결과값의 신뢰성을 높이기 위하여 임의선정을 통해 자료를 달리하여 10번의 실험을 실시하였다.

[표 4-5] 각 분류 모형의 예측 정확도(Hit ratios:%) - 간암 자료

시뮬레이션	Logit	CART	NN	Hybrid Model I		
				Logit-assisted NN	CART-assisted NN	GA-assisted NN
1	92.96	91.46	92.96	91.46	90.95	90.95
2	90.95	90.45	90.95	90.45	87.44	87.44
3	91.96	85.93	92.46	87.44	92.96	92.96
4	89.95	90.95	90.95	92.96	83.92	83.92
5	93.47	93.47	93.47	92.46	88.44	88.44
6	90.95	90.45	91.46	93.46	91.96	91.96
7	93.97	93.97	93.97	88.44	95.98	95.98
8	91.96	92.46	91.46	85.43	93.97	93.97
9	84.42	92.96	84.92	89.45	92.46	92.46
10	91.46	84.92	92.46	94.47	90.45	90.45
평균	91.21	90.70	91.51	90.60	90.85	90.85
		91.14			**90.77**	

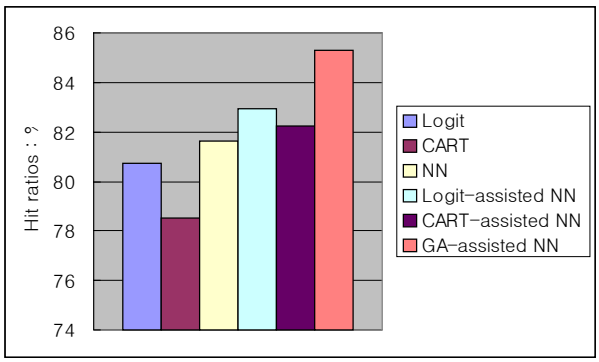
[그림 4-1] 예측 정확도의 평균 (Hit ratios:%) - 간암 자료



[표 4-6] 각 분류 모형의 예측 정확도(Hit ratios:%) - 심장 질환 자료

시뮬레이션	Logit	CART	NN	Hybrid Model I		
				Logit-assisted NN	CART-assisted NN	GA-assisted NN
1	85.19	79.63	85.19	81.49	81.49	87.04
2	77.78	79.63	81.49	81.48	79.63	83.34
3	88.89	81.49	85.19	79.63	81.49	88.00
4	75.93	81.49	77.78	88.89	85.19	85.19
5	81.49	74.08	83.34	83.34	87.04	79.63
6	81.49	79.63	81.48	85.19	85.19	87.04
7	79.63	74.08	78.73	81.49	79.63	83.33
8	77.78	81.49	83.34	81.49	83.34	88.89
9	75.76	75.76	77.78	84.75	79.63	87.04
10	83.34	77.78	83.78	81.48	79.63	83.34
평균	80.73	78.51	81.81	82.96	82.23	85.28
		80.35			83.49	

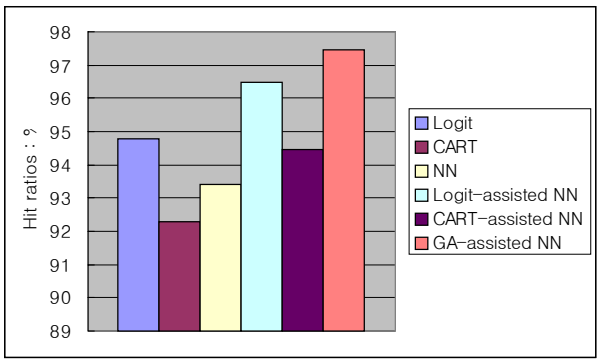
[그림 4-2] 예측 정확도의 평균 (Hit ratios:%) - 심장 질환 자료



[표 4-7] 각 분류 모형의 예측 정확도(Hit ratios:%) - 유방암 자료

시뮬레이션	Logit	CART	NN	Hybrid Model I		
				Logit-assisted NN	CART-assisted NN	GA-assisted NN
1	97.37	94.74	92.11	96.49	93.86	97.37
2	92.11	91.23	93.86	96.49	92.98	98.25
3	93.86	87.72	92.11	98.25	96.49	95.61
4	94.74	89.47	97.37	97.37	93.86	98.25
5	97.37	95.61	95.61	95.61	94.74	97.37
6	95.61	90.35	88.60	96.49	93.86	97.37
7	92.98	92.11	92.98	93.86	92.98	94.74
8	91.82	92.98	93.86	94.74	95.61	99.12
9	97.37	93.86	96.49	99.12	93.86	97.37
10	94.74	94.74	91.23	96.49	96.49	99.12
평균	94.80	92.28	93.42	96.49	94.47	97.46
		93.50			96.14	

[그림 4-3] 예측 정확도의 평균 (Hit ratios:%) - 유방암 자료



[표 4-5]는 간암 자료에 대한 결과인데, 총 10번의 모의실험에서 단일모형의 분류 예측력이 혼합모형 보다 조금 높은 것을 볼 수 있다. 또한 혼합모형의 경우, Logit-assisted NN과 CART-assisted NN, GA-assisted NN는 변수의 개수에서 두 배 차이가 남에도 불구하고 분류 예측력에서는 차이가 없는 것으로 나타났다. 결과적으로 혼합모형의 분류 예측력이 단일모형에 비해서 조금 떨어지기는 하였지만 변수의 개수를 $\frac{1}{2} \sim \frac{1}{4}$ 로 줄이면서 모든 변수를 사용한 단일모형에 버금가는 분류 예측력을 갖고있어 혼합모형을 사용하는 것이 효율적임을 알 수 있었다.

[표 4-6]은 심장 질환 자료에 대한 결과인데, 총 10번의 모의실험에서 혼합모형의 분류 예측력이 단일모형 보다 오히려 높은 것을 볼 수 있다. 또한 혼합모형의 경우, Logit-assisted NN과 CART-assisted NN, GA-assisted NN는 변수의 개수에서 두 배 차이가 남에도 불구하고 분류 예측력에서는 차이가 없는 것으로 나타났고, 특히 GA-assisted NN은 CART-assisted NN과 같은 개수의 변수를 사용했음에도 불구하고 더 우수한 분류 예측력을 갖고있어 변수선택법으로 유전자 알고리즘이 유용한 기법이 될 수 있음이 입증되었다.

[표 4-7]의 유방암 자료에 대한 결과 또한 심장 질환 자료의 결과와 대체로 유사하다.

4.4.1.3. 유의성 검정

본 연구에서는 각 모형간의 분류력 차이가 통계적으로 유의한지를 알아보기 위하여 t-test를 실시하였다.

각 단일모형과 혼합모형간의 정확도에 대한 t-test를 하기에 앞서, 먼저 두 독립된 집단간의 분산 동일성 여부를 확인한 뒤 유의확률을 얻었다.

[표 4-8] 분류 모형간의 유의성 검정(t-test) - 간암 자료

	Logit	CART	NN	Logit-assisted NN	CART-assisted NN	GA-assisted NN
Logit	-	0.6998	0.7995	0.6351	0.8034	0.8034
CART		-	0.5293	0.9409	0.9191	0.9191
NN			-	0.4675	0.6383	0.6383
Logit-assisted NN				-	0.8632	0.8632
CART-assisted NN					-	1
GA-assisted NN						-

(** 유의수준 1%이내, * 유의수준 5%이내)

[표 4-9] 분류 모형간의 유의성 검정(t-test) - 심장 질환 자료

	Logit	CART	NN	Logit-assisted NN	CART-assisted NN	GA-assisted NN
Logit	-	0.1891	0.5937	0.1831	0.3620	0.0112*
CART		-	0.0322	0.0025**	0.0095**	0.0001**
NN			-	0.3283	0.6522	0.0128*
Logit-assisted NN				-	0.5770	0.0730
CART-assisted NN					-	0.0260*
GA-assisted NN						-

(** 유의수준 1%이내, * 유의수준 5%이내)

[표 4-10] 분류 모형간의 유의성 검정(t-test) - 유방암 자료

	Logit	CART	NN	Logit-assisted NN	CART-assisted NN	GA-assisted NN
Logit	-	0.0285*	0.2142	0.0568*	0.6868	0.0040**
CART		-	0.3393	0.0003**	0.0275*	0.0001**
NN			-	0.0051**	0.2717	0.0004**
Logit-assisted NN				-	0.0056**	0.1603
CART-assisted NN					-	0.0001**
GA-assisted NN						-

(** 유의수준 1%이내, * 유의수준 5%이내)

[표 4-8]의 결과를 통해 각 모형들간의 차이가 통계적으로는 유의한 것이 아님을 알 수 있다. 따라서 적은 수의 변수를 가지는 혼합모형을 사용하는 것이 효율적인 측면에서 바람직하다고 본다.

[표 4-9]의 결과를 통해 GA-assisted NN은 Logit-assisted NN을 제외한 모든 모형과 5% 이내의 유의수준을 보이고 있어 다른 모형들과의 차이가 통계적으로 유의함을 확인할 수 있었다. 결과적으로 심장 질환 자료에 대해서 혼합모형의 분류 예측력이 단일모형에 비해서 더 나은 분류 예측력을 갖고 있으며 그 중에서도 GA-assisted NN이 가장 우수함을 알 수 있었다.

[표 4-10]의 유방암 자료에 대한 결과 또한 심장 질환 자료의 결과와 대체로 유사하다.

혼합모형 I에 대한 모의실험의 결과를 종합하면 다음과 같다.

첫째는, 변수의 개수를 상당수 줄일 수 있는 혼합모형이 모든 변수를 사용한 단일모형 이상의 분류 예측력을 갖고있어 혼합모형을 사용하는 것이 효율적임을 알 수 있었다.

둘째는, 혼합모형 중 GA-assisted NN이 가장 우수한 분류 예측력을 갖고있어 변수선택법으로 유전자 알고리즘이 유용한 기법이 될 수 있음을 확인할 수 있었다.

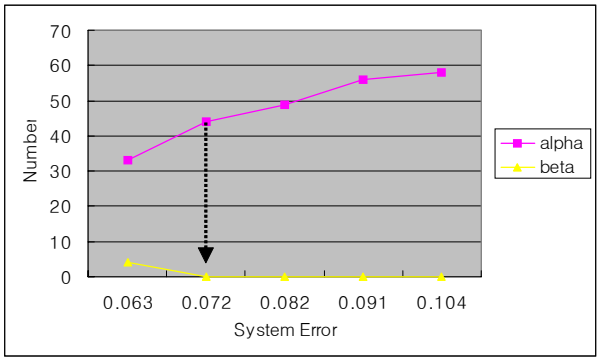
셋째는, 연속형과 범주형이 혼합되어 있는 데이터뿐만 아니라 연속형으로만 이루어진 데이터에서도 유사한 결과를 얻을 수 있었다.

4.4.2. 혼합모형(Hybrid Model) II

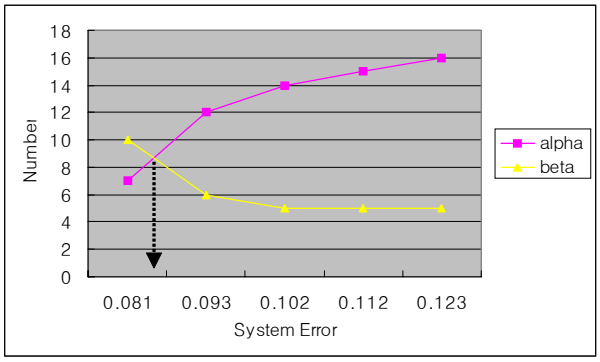
4.4.2.1. 분석 결과

본 연구에서 설정한 시스템오류는 다음과 같다.

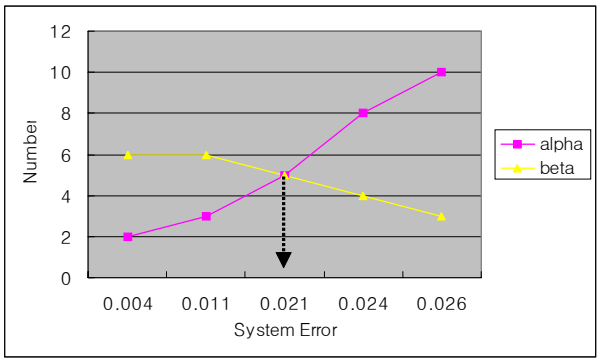
[그림 4-4] 시스템오류와 α, β 의 관계 - 간암 자료



[그림 4-5] 시스템오류와 α, β 의 관계 - 심장 질환 자료



[그림 4-6] 시스템오류와 α, β 의 관계 - 유방암 자료

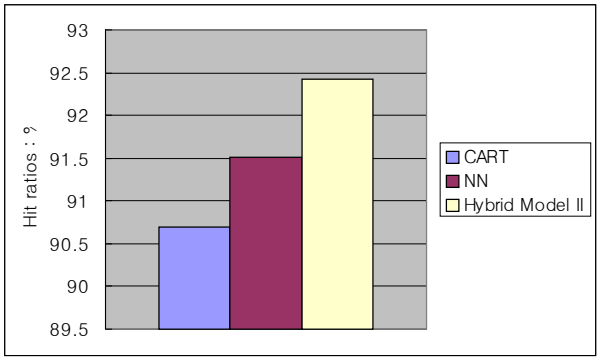


간암 자료, 심장 질환 자료, 유방암 자료 모두 α, β 의 방향이 반대의 성향을 띠고 있는 것을 볼 수 있는데 α, β 가 교차하는 시점을 시스템오류값으로 설정한다. 이렇게 해서 설정한 시스템오류값이 각각 0.072, 0.085, 0.021이다. 따라서 이 값을 기준으로 해서 시스템오류가 크거나 작을 때, 각각에 해당되는 의사결정을 내리게 된다.

[표 4-11] 각 분류 모형의 예측 정확도(Hit ratios:%) - 간암 자료

시뮬레이션	CART	NN	Hybrid Model II CART-assisted NN
1	91.46	92.96	94.47
2	90.45	90.95	93.97
3	85.93	92.46	94.85
4	90.95	90.95	89.69
5	93.47	93.47	88.65
6	90.45	91.46	88.76
7	93.97	93.97	95.47
8	92.46	91.46	88.76
9	92.96	84.92	94.85
10	84.92	92.46	94.85
평균	**90.70**	**91.51**	**92.43**

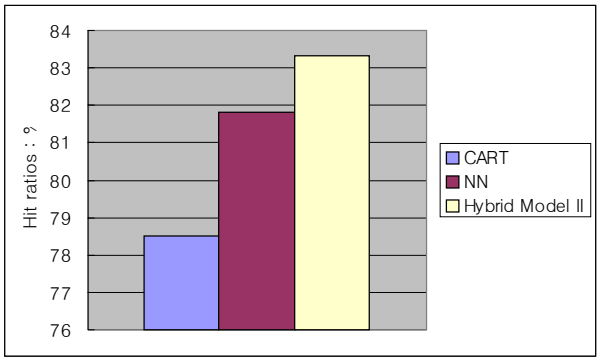
[그림 4-7] 예측 정확도의 평균 (Hit ratios:%) - 간암 자료



[표 4-12] 각 분류 모형의 예측 정확도(Hit ratios:%) - 심장 질환 자료

시뮬레이션	CART	NN	Hybrid Model II CART-assisted NN
1	79.63	85.19	87.04
2	79.63	81.49	83.34
3	81.49	85.19	85.19
4	81.49	77.78	83.34
5	74.08	83.34	85.19
6	79.63	81.48	81.48
7	74.08	78.73	79.63
8	81.49	83.34	83.34
9	75.76	77.78	79.63
10	77.78	83.78	85.19
평균	**78.51**	**81.81**	**83.34**

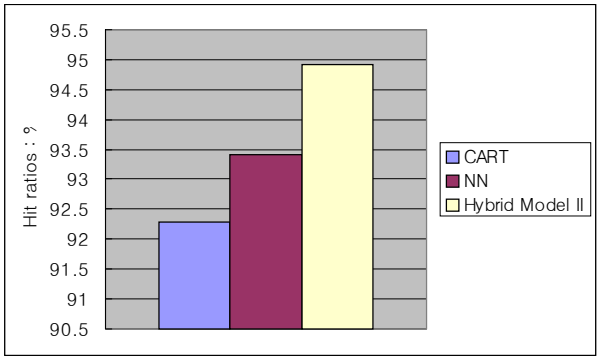
[그림 4-8] 예측 정확도의 평균 (Hit ratios:%) - 심장 질환 자료



[표 4-13] 각 분류 모형의 예측 정확도(Hit ratios:%) - 유방암 자료

시뮬레이션	CART	NN	Hybrid Model II CART-assisted NN
1	94.74	92.11	95.61
2	91.23	93.86	94.74
3	87.72	92.11	93.86
4	89.47	97.37	97.37
5	95.61	95.61	96.49
6	90.35	88.60	92.11
7	92.11	92.98	94.74
8	92.98	93.86	94.74
9	93.86	96.49	96.49
10	94.74	91.23	92.98
평균	**92.28**	**93.42**	**94.91**

[그림 4-9] 예측 정확도의 평균 (Hit ratios:%) - 유방암 자료



[표 4-11], [표 4-12], [표 4-13]은 각각 간암 자료, 심장 질환 자료, 유방암 자료에 대한 결과인데, 총 10번의 모의실험에서 혼합모형의 분류 예측력이 단일모형보다 높은 것을 볼 수 있다.

4.4.2.2. 유의성 검정

본 연구에서는 각 모형간의 분류력 차이가 통계적으로 유의한지를 알아보기 위하여 t-test를 실시하였다.

각 단일모형과 혼합모형간의 정확도에 대한 t-test를 하기에 앞서, 먼저 두 독립된 집단간의 분산 동일성 여부를 확인한 뒤 유의확률을 얻었다.

[표 4-14] 분류 모형간의 유의성 검정(t-test) - 간암 자료

	CART	NN	CART-assisted NN
CART	-	0.5293	0.2184
NN		-	0.4673
CART-assisted NN			-

(** 유의수준 1%이내, * 유의수준 5%이내)

[표 4-15] 분류 모형간의 유의성 검정(t-test) - 심장 질환 자료

	CART	NN	CART-assisted NN
CART	-	0.0322*	0.0009**
NN		-	0.2176
CART-assisted NN			-

(** 유의수준 1%이내, * 유의수준 5%이내)

[표 4-16] 분류 모형간의 유의성 검정(t-test) - 유방암 자료

	CART	NN	CART-assisted NN
CART	-	0.3393	0.0139*
NN		-	0.1449
CART-assisted NN			-

(** 유의수준 1%이내, * 유의수준 5%이내)

[표 4-14]의 결과를 통해 혼합모형과 각각의 단일모형과의 차이가 존재하기는 하지만 통계적으로는 유의한 것이 아님을 알 수 있다. 그러나 이것은 암과 정상의 데이터 비율이 1:10이라는 큰 차이에서 기인한 것으로 보여진다.

[표 4-15], [표 4-16]의 결과를 통해 혼합모형이 CART와는 5% 이내의 유의수준을 보이고 있지만 NN과는 유의적이지 않은 것을 확인할 수 있었다.

혼합모형 II에 대한 모의실험의 결과를 종합하면 다음과 같다.

첫째는, 제시된 혼합모형의 분류력이 단일모형에 비해 전체적으로 높은 것을 알 수 있었다.

둘째는, 데이터의 형태에 상관없이 CART와는 유의적인 반면 NN과는 유의적이지 않았다. 이것을 통하여 연속형인 데이터의 경우에 NN의 분류력이 우수하다는 기존의 연구를 재확인하였다.

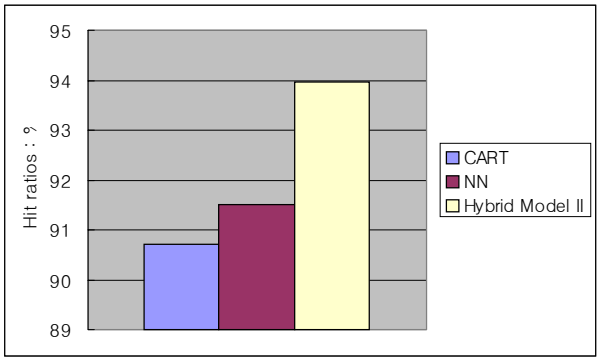
4.4.3. 혼합모형(Hybrid Model) III

4.4.3.1. 분석 결과

[표 4-17] 각 분류 모형의 예측 정확도(Hit ratios:%) - 간암 자료

시뮬레이션	CART	NN	Hybrid Model III	
				Weights (CART, NN)
1	91.46	92.96	97.94	(0.462, 0.538)
2	90.45	90.95	99.48	(0.412, 0.588)
3	85.93	92.46	94.85	(0.235, 0.765)
4	90.95	90.95	88.76	(0.398, 0.602)
5	93.47	93.47	88.76	(0.644, 0.356)
6	90.45	91.46	91.96	(0.639, 0.361)
7	93.97	93.97	97.94	(0.695, 0.305)
8	92.46	91.46	90.45	(0.441, 0.559)
9	92.96	84.92	93.97	(0.219, 0.781)
10	84.92	92.46	95.48	(0.416, 0.584)
평균	**90.70**	**91.51**	**93.96**	

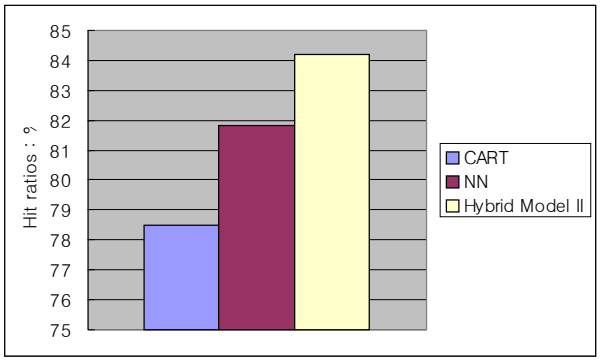
[그림 4-10] 예측 정확도의 평균 (Hit ratios:%) - 간암 자료



[표 4-18] 각 분류 모형의 예측 정확도(Hit ratios:%) - 심장 질환 자료

시뮬레이션	CART	NN	Hybrid Model III	
				Weights (CART, NN)
1	79.63	85.19	87.04	(0.045, 0.955)
2	79.63	81.49	83.78	(0.135, 0.865)
3	81.49	85.19	85.19	(0.421, 0.579)
4	81.49	77.78	83.78	(0.694, 0.306)
5	74.08	83.34	85.19	(0.299, 0.701)
6	79.63	81.48	81.48	(0.312, 0.688)
7	74.08	78.73	83.33	(0.358, 0.642)
8	81.49	83.34	85.19	(0.025, 0.975)
9	75.76	77.78	83.33	(0.191, 0.809)
10	77.78	83.78	83.78	(0.031, 0.969)
평균	**78.51**	**81.81**	**84.21**	

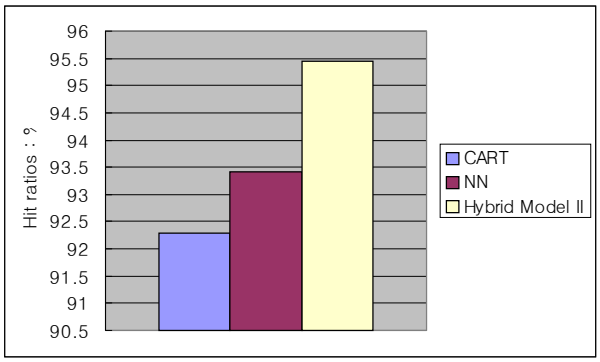
[그림 4-11] 예측 정확도의 평균 (Hit ratios:%) - 심장 질환 자료



[표 4-19] 각 분류 모형의 예측 정확도(Hit ratios:%) - 유방암 자료

시뮬레이션	CART	NN	Hybrid Model III	
				Weights (CART, NN)
1	94.74	92.11	95.61	(0.751, 0.249)
2	91.23	93.86	95.61	(0.466, 0.534)
3	87.72	92.11	96.49	(0.012, 0.988)
4	89.47	97.37	97.37	(0.134, 0.866)
5	95.61	95.61	96.49	(0.512, 0.488)
6	90.35	88.60	93.86	(0.621, 0.379)
7	92.11	92.98	92.98	(0.359, 0.641)
8	92.98	93.86	94.74	(0.491, 0.509)
9	93.86	96.49	96.49	(0.223, 0.777)
10	94.74	91.23	94.74	(0.615, 0.385)
평균	**92.28**	**93.42**	**95.44**	

[그림 4-12] 예측 정확도의 평균 (Hit ratios:%) - 유방암 자료



[표 4-17], [표 4-18], [표 4-19]은 각각 간암 자료, 심장 질환 자료, 유방암 자료에 대한 결과인데, 훈련용 자료에서 얻어진 가중치(weights)가 검증용 자료에 적용되었을 때의 분류정확도를 나타낸다. 총 10번의 모의실험에서 혼합모형의 분류 예측력이 각각의 단일모형에 비해서 조금 향상이 있음을 보여주고 있다.

4.4.3.2. 유의성 검정

본 연구에서는 각 모형간의 분류력 차이가 통계적으로 유의한지를 알아보기 위하여 t-test를 실시하였다.

각 단일모형과 혼합모형간의 정확도에 대한 t-test를 하기에 앞서, 먼저 두 독립된 집단간의 분산 동일성 여부를 확인한 뒤 유의확률을 얻었다.

[표 4-20] 분류 모형간의 유의성 검정(t-test) - 간암 자료

	CART	NN	Hybrid Model III
CART	-	0.5293	0.0514
NN		-	0.1116
Hybrid Model III			-

(** 유의수준 1%이내, * 유의수준 5%이내)

[표 4-21] 분류 모형간의 유의성 검정(t-test) - 심장 질환 자료

	CART	NN	Hybrid Model III
CART	-	0.0291*	0.0004**
NN		-	0.0449*
Hybrid Model III			-

(** 유의수준 1%이내, * 유의수준 5%이내)

[표 4-22] 분류 모형간의 유의성 검정(t-test) - 유방암 자료

	CART	NN	Hybrid Model III
CART	-	0.3393	0.0030**
NN		-	0.0446*
Hybrid Model III			-

(** 유의수준 1%이내, * 유의수준 5%이내)

[표 4-20]의 결과를 통해 혼합모형과 각각의 단일모형과의 차이가 존재하기는 하지만 통계적으로는 유의한 것이 아님을 알 수 있다. 그러나 이것은 암과 정상의 데이터 비율이 1:10이라는 큰 차이에서 기인한 것으로 보여진다.

[표 4-21]의 결과를 통해 혼합모형이 CART와는 1%이내의 유의수준을 보이고 있고 NN과는 5%이내의 유의수준을 보이고 있어 분류력의 향상이 유의적임을 확인할 수 있었다.

[표 4-22]의 유방암 자료에 대한 결과 또한 심장 질환 자료의 결과와 유사하다. 결론적으로 두 기법간의 연결 가중치를 구하여 두 기법을 통합한 방법을 사용하였을 경우, 데이터의 형태에 상관없이 각각의 단일모형과 비교해 통계적으로 의미 있는 분류력의 향상을 가져 올 수 있었다.

혼합모형 III에 대한 모의실험의 결과를 종합하면 다음과 같다.

첫째는, 제시된 혼합모형의 분류력이 단일모형에 비해 전체적으로 높은 것을 알 수 있었다.

둘째는, 혼합모형 II 와는 달리, 데이터의 형태와 상관없이 모든 단일모형과 비교해 분류력의 향상이 확인되었다.

제 6 장 결론

분류분석으로 주로 사용되었던 단일모형과 비교해보았을 때 실제로 본 연구에서 제시한 혼합모형이 더 나은 분류 예측력을 갖는 것을 확인할 수 있었다.

실제 자료를 토대로 본 연구의 혼합모형이 실제로 분류력의 향상을 가지고 오는지에 대해서 확인을 해 보았고, 분류력 향상의 결과 또한 통계적으로 의미가 있다는 것도 알 수 있었다.

본 연구가 기존의 연구와 다른 점은 보통의 연구들이 단일모형을 통한 해석에 초점을 둔 분석에 관심을 가졌지만, 최근에는 해석에는 다소 복잡하지만 분류 예측력 향상에 초점을 둔 분석접근이 시도되고 있다. 이러한 분위기에 맞춰 본 연구에서도 분류력 향상에 초점을 맞추고 개별기법을 통합한 혼합된 기법을 사용하여 분류력의 향상을 꾀하였는데 다음과 같은 세 가지 관점에서 접근을 하였다.

첫째는, 단일모형을 이용한 변수선택결과를 신경망의 입력변수로 사용하는 혼합모형을 제시함으로서 분류 예측력을 향상시킬 수 있는 접근방법을 시도하였고 이 방법론이 다른 통계기법에 비해 우수함을 신경망 모형에 적용한 결과의 비교를 통해서 보여 주었으며, 이들간의 분류력의 차이가 유의함을 통계적 검정을 통하여 확인하였다. 둘째는, 신경망모형의 부족한 부분을 의사결정나무(CART)로 보완하고자 하는 시도로서 의사결정나무와 신경망모형을 결합한 '의사결정나무지원 신경망'(Decision Tree-Assisted Neural Network)기법이라 칭한다. 모의실험 결과 혼합모형의 분류력 향상은 개별기법(CART)과 유의적임을 확인 할 수 있었다. 셋째는, 개별기법의 결과를 최적의 연결 가중치를 사용하여 결합하고자 하는 시도로서 개별모형으로서는 CART와 NN이 사용되었고 각 분류 기법을 통합하기 위한 최적의 연결 가중치를 찾기 위해서는 유전자 알고리즘(GA)이 적용되었는데, 모의실험 결과 혼합모형의 분류력 향상은 개별기법(CART)과 유의적임을 확인 할 수 있었다.

본 연구에서 제시한 세 개의 혼합모형 중 가장 우수한 모형을 선택하기란 어려운 문제다. 그러나 분류 예측력 측면에서 보았을 때에는 혼합모형 III가 가장 우수한 모형이라고 말할 수 있는데 특히 데이터의 형태에 상관없이 분류력의 향상이 있음은 주목할만한 결과이다. 한편 효율성 측면에서 보았을 때에는 변수의 개수를 상당수 줄이면서 모든 변수를 사용한 단일모형에 버금가는 분류 예측력을 갖고있는 혼합모형 I이 우수한 모형이라고 말할 수 있을 것이다.

본 연구가 가지는 한계점은 다음과 같다. 첫째는, 실제 사용하기에는 다소 번거롭다는 단점을 가지고 있다. 둘째는, 본 연구에 사용된 자료는 연속형과 범주형의 혼합된 형태와 모두 연속형인 자료로 이루어져 있는데, 만약 모두 이산형 자료를 사용할 경우에도 유사한 결과가 나올 것인가는 미지수이다. 셋째는, 본 연구에서 제시한 혼합모형은 모형구축으로 신경망을 사용하기 때문에 해석의 어려움을 가지고 있다. 그러므로 해석이 중요한 연구에 적용하기에는 다소 무리가 있는 분석방법이라는 한계점을 가지고 있다. 따라서 해석의 편의를 위해서는 모형구축시 신경망 대신에 로지스틱 회귀분석을 사용하는 CART-assisted Logit Model이나 GA-assisted Logit Model이 도움이 될 것이다.

향후의 연구방향은 첫째는, 본 연구에서 이용한 다층신경망(MLP)기법 이외에 보다 많은 다양한 신경망기법들을 이용해 보는 것이다. 둘째는, 모형구축시 신경망 이외에 의사결정나무, 사례기반추론(Case-Based Reasoning, CBR), 전문가 시스템(Expert System), 퍼지(Fuzzy)등을 이용한 혼합모형에 관한 연구이다.

참고 문헌

박성수, 박해영, “(C++로 구현한) 유전자 알고리즘”, 한올출판사, 2001.

최종후, 권기만, 김수택, “신용평점 모형”, 세창출판사, 2002.

Breslow N.E., Day N.E., "Statistical methods in cancer research", Lyon, IARC Sci Publ, 1980.

Coakley, J. R. and C. E. Brown, "Neural Networks Applied to Ratio Analysis in the Analytical Review Process", The 4th International Symposium on Expert Systems in Accounting, Finance and Management(1991), 1-35.

Colin, A. "Neural Networks and Genetic Algorithms for Exchange Rate Forecasting." Proceedings IJCNN (Beijing), 1992.

David E. Goldberg, "Genetic Algorithms in Search, Optimization, and Machine Learning", ADDISON-WESLEY PUBLISHING COMPANY, INC, 1989.

D.C. Indro, C.C. Jiang, B.E. Patuwo, G.P. Zhang, "Predicting mutual fund performance using artificial neural networks" Omega, Int. J. Mgmt. Sci. 27(1999), 373-380.

Deboeck, G.J., "Trading On The Edge (ed), 1994.

Dorronsoro, J., Ginel, F., Sanchez, C., and Santa, C. C., "Neural Fraud Detection in Credit Card Operations", IEEE Trans. Neural Networks, 8, pp. 827-834, 1997.

Guoqiang Zhang, Michael Y. Hu, B. Eddy Patuwo, Daniel C. Indro, "Artificial neural networks in bankruptcy prediction: General framework and cross-validation analysis", European Journal of Operational Research, 116(1999), 16-32.

Hongkyu Jo, Ingoo Han, "Integration of Case-Based Forecasting, Neural Network, and Discriminant Analysis for Bankruptcy Prediction", Expert Systems With Application, Vol. 11, No. 4, PP. 415-422, 1996.

Hosmer D.W., Lemeshow S, "Applied logistic regression", John Wiley and Sons, 1989.

Ingoo Han, Cheolsoo Park, Chulhong Kim, "Bankruptcy Predictions for Korea Medium-Sized Firms using Neural Networks and Case Based Reasoning".

Ingoo Han, Hongkyu Jo, Kyung Shik Shin, "The Hybrid Systems for Credit Rating", Journal of the Korean OR/MS Society Vol. 22, No. 3, September 1997.

Javed Khan, Jun S. Wei,..., "Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks", Nature Medicine Vol. 7, No. 6, June 2001.

John J. Maher, Tarun K. Sen, "Predicting Bond Ratings Using Neural Networks: A Comparison with Logistic Regression", INTELLIGENT SYSTEMS IN ACCOUNTING, FINANCE AND MANAGEMENT VOL. 6(1997): 59-72.

J.R. Quinlan, "Induction of Decision Trees", Machine Learning, 1(1986), 81-106.

Kun Chang Lee, Ingoo Han, Youngsig Kwon, "Hybrid neural network models for bankruptcy predictions", Decision Support Systems, 18(1996), 63-72.

Kyoung-jae Kim, Ingoo Han, "Genetic algorithms approach to feature discretization in artificial neural networks for the prediction of stock price index", Expert Systems with Applications, 19(2000), 125-132.

Lee, K.C., H. Kim and M. Kim, "An inductive learning-assisted neural network approach to bankruptcy prediction: Comparison with MDA, inductive learning, and neural network models", Korean Management Study, 1995.

Lijia, G. and Morgan, C. W., "Data Mining Techniques for Mortality at Advanced Age", 2000.

Press, S. J. and Wilson, S., "Choosing Between Logistic Regression and Discriminant Regression", Journal of the American Statistical Association, 73, 1978.

Samuel, E. B. and Ciril, K., "Using k-nearest-neighbor classification in the leaves of a tree", Computational Statistics & Data Analysis 40, pp27-37, 2002.

Steinberg, D., "Hybrid CART-logit and CART-neural nets for classification and regression", American Statistical Association, 1999.

Will Dwinnell, "Model Input Selection", PCAI, January/February 1998, Volume 12, No 1, 23-26.

부 록

Genetic Algorithms - Visual C++ Source

```
#include <stdlib.h>
#include <time.h>

#define MAXPOP 25  //  교배할 개체군이 25개란 의미.

struct gene {
        int alleles[4]; // W1, W2 값 각각 소수 둘째 자리까지 있으므로 공간을
4개 확부했다.
        int fitness;   // SSE 계산값.
        float likelihood; // 적합도// 이 값이 높아야 좋은 부모 개체가 된다.


        // Test for equality.
        operator==(gene gn) {
                for (int i=0;i<4;i++) {
                        if (gn.alleles[i] != alleles[i]) return false;
                }

                return true;
        }
};

class CDiophantine {
        public:
                CDiophantine(double*, double*, int *,int, int, double, int);
                // Constructor with coefficients for a,b,c,d.
                int Solve();
                // Solve the equation.

                // Returns a given gene.
                gene GetGene(int i) { return population[i];}

                double Ave_test_x[250];
                int test_max;
                int test_min;
                int dol_per ; // 돌연변이 확률
```

```
        int SSE_min;  //
        int W1_1;
        int W1_2;
        int W2_1;
        int W2_2;


    protected:
        double *PJ2, *PJ1;
        int *Data_Y;
        int result;
        int line; // 파일에서 읽어들인 데이터는 몇 줄인가?
        double Loof_Num; // 몇 세대를 교배할 것인가?
        int dol; // 돌연변이 확률

        gene population[MAXPOP];            // 개체군.

        int Fitness(gene &);                // 계산값
        int ToDigital(double a);
        void GenerateLikelihoods();         // 적합도 평가
        float MultInv();

        int CreateFitnesses();
        void CreateNewPopulation();
        int GetIndex(float val);

        gene Breed(int p1, int p2);

};

CDiophantine::CDiophantine(double *a, double *b, int *c, int res , int d , double
loof, int dol)
{
        PJ1 =a ;
        PJ2 = b;
        Data_Y = c;
        result = res; // SSE 가 목표로 하는값.
        line = d; // 앞서 파일에서 읽어들인 데이터가 몇 줄인가?
        Loof_Num  = loof; // 몇 세대를 교배 할 것인가
        dol_per = dol; // 돌연변이 확률. // 이상.. 생성자 초기값
}
```

```
int CDiophantine::Solve() {
	int fitness = -1;

	// Generate initial population.
	srand((unsigned)time(NULL));
	SSE_min =100000; // SSE 최소값 저장 공간

	for(int i=0;i<MAXPOP;i++)
	{                                                    // Fill the population
with numbers between
		for (int j=0;j<4;j++)
		{                                            // 0 and the result.
			//population[i].alleles[j] = rand() % (result + 1);
			population[i].alleles[j] = rand() % (10); // 처음 개체는
랜덤하게 구성한다.
		}
	}

	//fitness = CreateFitnesses();
	//int Loof_Num = 10;
	int iterations = 0;                    // Keep record of the iterations.
	int Step = 0;
	test_max=0;
	test_min=100000;

	while (fitness != 0 || iterations < Loof_Num) {    //   Loof_Num   만큼
반복한다.
		GenerateLikelihoods();    // 부모자격 평가.
		CreateNewPopulation();  // 차세대 개체 생성
		if (fitness = CreateFitnesses()) {
			return fitness;
		}

		iterations++;

	/*	if(iterations%(Loof_Num/200) == 0)
		{
			double test_x = 0;
			for(i=0;i<MAXPOP;i++)
			{
				test_x += Fitness(population[i]);
```

```
                        }
                        Ave_test_x[Step] = test_x;
                        Step++;
                        if(test_x>test_max)
                        {
                                test_max = test_x;
                        }
                        if(test_x<test_min)
                        {
                                test_min = test_x;
                        }
                }*/
        }


        return -1;
}
int CDiophantine::ToDigital(double a) // 0 - 1 사이의 소수를 0 과 1로 구분 디지털화한다.
{
        if(a<0.5)
        {
                return 0;
        }
        if(a>=0.5)
        {
                return 1;
        }
}
int CDiophantine::Fitness(gene &gn)  // 실질적인 SSE 계산. // 앞서 파일을 열어 얻은 // 변수들을 사용해 연산하였다.

{
        int total = 0;
        int tmp_total = 0;

        for(int i=0;i<line;i++)
        {
                tmp_total = 0;
                tmp_total =abs( (Data_Y[i] - ToDigital(PJ1[i]* ( gn.alleles[0] * 0.1 + gn.alleles[1] * 0.01)))
```

```
                                        + (Data_Y[i] - ToDigital(PJ2[i] * (
gn.alleles[2] * 0.1 + gn.alleles[3] * 0.01))));
                total += tmp_total;
                // SSE 연산.
                // alleles[0] alleles[1] 은 각각  W1 의 소수 첫번째 자리, 소수
두번째 자리.
                // alleles[1] alleles[2] 은 각각  W2 의 소수 첫번째 자리, 소수
두번째 자리.
                // Data_Y 는 Y 값

        }
        if(total<SSE_min && ((gn.alleles[0]+gn.alleles[2])*10 + gn.alleles[1] +
gn.alleles[3]==100)) // 여태까지 평가했던 값 중에 가장  작은 값인가?
        {                       // 그리고 W1 과 W2 값의 합이 1인가?
                SSE_min = total;
                W1_1 = gn.alleles[0];
                W1_2 = gn.alleles[1];
                W2_1 = gn.alleles[2];
                W2_2 = gn.alleles[3];
        }

        return gn.fitness = abs(total);  // 계산 값을 리턴 한다. 이 값으로 부모
의 자격을 평가한다.
        // 프로젝트의 목적은 이 값이 작아야 하므로 작을수록 좋은 평가를 받아
차세대 개체를 생산할
        // 부모로 선정될 확률이 많다.
}

int CDiophantine::CreateFitnesses() {
        float avgfit = 0;
        int fitness = 0;
        for(int i=0;i<MAXPOP;i++) {
                fitness = Fitness(population[i]);
//              avgfit += fitness;
                if (fitness == 0) {
                        return i;
                }
        }
        return 0;
}
```

```
float CDiophantine::MultInv() {
	float sum = 0;

	for(int i=0;i<MAXPOP;i++) {
		sum += 1/((float)population[i].fitness); // 룰렛식으로 각 개체의 likelihood 의 영역을 결정한다.
	}

	return sum;
}

void CDiophantine::GenerateLikelihoods() {
	float multinv = MultInv();

	float last = 0;
	for(int i=0;i<MAXPOP;i++)
	{
		last = last + ((1/((float)population[i].fitness) / multinv) * 100);
		population[i].likelihood = last; // 정답에 가까운 부모에게 큰 likelihood 부여.
	}
}

int CDiophantine::GetIndex(float val) {
	float last = 0;
	for(int i=0;i<MAXPOP;i++)
	{
		if (last <= val && val <= population[i].likelihood) return i;
		else last = population[i].likelihood;  // val 값을 받아 합당한 부모를 골라 번호를 리턴.
	}
	return 4;
}

gene CDiophantine::Breed(int p1, int p2) {
	int crossover = rand() % 3+1;			// 잘라낼 길이 생성
	int first = rand() % 100; // 앞에서 잘라낼 것인가 뒤에서 잘라낼 것인가

	gene child = population[p1];			// p1 을 기본 부모로 설정.

	int initial = 0, final = 3;
```

```
        if (first < 50) initial = crossover;
        else final = crossover+1;

        for(int i=initial;i<final;i++) {                    // 크로스 오버
                child.alleles[i] = population[p2].alleles[i];
                if (rand() % 101 < dol_per) child.alleles[i] = rand() % 10; //
돌연변이
        }

        return child;
                // 생성된 자식 리턴.
}

void CDiophantine::CreateNewPopulation()
{
        gene temppop[MAXPOP];

        for(int i=0;i<MAXPOP;i++)
        {
                int parent1 = 0, parent2 = 0, iterations = 0;
                while(parent1 == parent2 || population[parent1] ==
population[parent2])
                {
                        parent1 = GetIndex((float)(rand() % 101)); // 확률로
첫 번째 부모를 구한다.
                        parent2 = GetIndex((float)(rand() % 101)); // 확률로
두 번째 부모를 구한다.
                        // SSE 값이 작을수록 부모가 될 확률이 높다.
                        if (++iterations > 25)
                        {
                                break;
                        }
                }
                temppop[i] = Breed(parent1, parent2);       // p1. p2 교배
        }
        for(i=0;i<MAXPOP;i++) population[i] = temppop[i];
}
```

ABSTRACT

A study on Predictive Method using the Hybrid Model

Bae, Jang Seob
Dept. of Biostatistics and Computing
The Graduate School
Yonsei University

In a conventional research, researchers tried to solve the classification problems by utilizing statistical methods such as DA, Logit, Probit, however these trials had the methodological limitation in that they should satisfy the strict assumptions of statistical techniques. For this reason, recently researchers are introducing the method applying artificial intelligence such as decision tree and neural network to classification problems.

Especially, conventional research has been focused on the analysis through single model, however, recently researchers are more experimenting the analysis using hybrid model, which is rather complex in structure, difficult in interpretation but focused on hit ratios. In consequence, I try to introduce hybrid models in this research as a method of solving classification problems and review the performance of these hybrid model according to the current academic trend.

In this research paper, three perspectives of hybrid models are introduced. First perspective suggests hybrid model I, which uses variable selection result, using single model as input variable. Second perspective suggests hybrid model II, which tries to supplement lacking parts of neural network with decision tree(CART). Last hybrid model III is an experiment to combine the each

individual result through optimal linkage weight.

Hybrid models were applied to real data in this research. Result of hybrid model was also compared with that of single model. Importantly, result of this experiment proved that hybrid model was improved in classification in comparison with single model.

Key words : Hybrid Model, Variable Selection, CART, Genetic Algorithm, Neural Network