韓國OA學會論文誌
第1卷 第1號, 1996. 8

論文 96-1-1-3-1

신경망이론을 이용한 비중심 F분포 확률계산

구 선 희*

Computation of Noncentral F Probabilities using Neural Network Theory

Son Hee Gu*

요 약

ANOVA 검정에서 검정통계량은 단일 또는 이중 비중심F분포를 따르며 비중심F분포는 일반적인 선형 가설 검정에서 검정함수 계산에 적용되고 있다.

본 논문에서는 단일 비중심F분포의 누적함수 계산에 신경망이론을 적용하였다. 신경망 구조는 다층 퍼셉트론이며 학습과정은 역전파 학습알고리즘이다. 신경망이론에 의하여 계산한 결과와 Patnaik 이 제시한 확률값을 비교하여 제시하였다.

ABSTRACT

The test statistic in ANOVA tests has a single or doubly noncentral F distribution and the noncentral F distribution is applied to the calculation of the power functions of tests of general linear hypotheses.

In this paper, the evaluation of the cumulative function of the single noncentral F distribution is applied to the neural network theory. The neural network consists of the multi-layer perceptron structure and learning process has the algorithm of the backpropagation. Numerical comparisons are made between the results obtained by neural network theory and the Patnaik's values.

Ⅰ. 서 론

비중심F분포는 1938년 Tang에 의하여 처음으로 제시되었으며 일반적으로 교호작용이 있는 분산분 석에서 검정력계산 및 선형 가설 검정에 적용되고 있다[10]. 비중심F분포에서 분포함수 (distribution function)의 값을 계산하는데에는 일반적으로 많은 시간이 소요될 뿐만 아니라 정확성이 문제가 되고 있다. 이로 인하여 이 분포함수에 대한 연구가 Price (1964),

* 전주대학교 전기전자컴퓨터 공학부 객원교수

Bulgren(1971), Tiku(1972, 1974, 1975), Chou et al(1985), Ennis와 Johnson (1993), Chattamvelli(1994) 등에 의하여 이루어졌다[4].

본 논문에서는 비중심F분포의 확률계산을 하기 위하여 인공지능(artificial intelligence)분야에서 많은 연구가 이루어진 인공 신경망(artificial neural network : 이하 신경망)이론을 적용하여 먼저 모형을 설정하고 이 계산결과를 Patnaik의 누적함수값과 비교하고자 한다.

Ⅱ. 신경망 이론

2.1 신경망의 구조

신경망은 네트워크의 위상(topology), 유니트(unit), 학습규칙(learning rule)의 세가지 요소로 구성된다[1][2]. 네트워크의 위상은 각 유니트들이 연결된 형태로서 기본적으로 입력층(input layer)과 출력층(output layer), 그리고 이 두개의 층을 연결하는 연결강도(connection strength)로 구성되어 있다. 그러나 이러한 네트워크의 위상은 항상 일정한 것이 아니며, 경우에 따라 그 위상을 다르게 구성할 수 도 있다.

유니트는 각 층을 이루고 있는 노드이며, 변수의 수 만큼 노드 수가 존재한다. 예를 들어 p차원의 변수가 입력되어 학습을 하는 경우 이 네트워크는 p개의 입력노드가 필요하다.

네트워크의 입력층과 출력층, 또는 은닉층을 연결하는 연결강도의 값은 반복이 이루어지면서 계 조정을 하며 이러한 과정을 학습이라고 하고, 이 학습은 일정한 공식에 의해서 이루어지는데 이것을 학습규칙이라고 한다.

2.2 역전파 알고리즘

역전파 학습 알고리즘은 평균제곱오차(mean square error) 알고리즘의 비선형적인 확장으로 미분의 반복규칙(chain rule)을 여러번 반복적으로 적용하여 오차의 최저점을 결정하는 최적의 연결 강도의 집합을 구하는 데 있다. 그러나 특정영역에 대하여 최적의 연결강도를 결정하는 것이 매우 어려움으로 예제(example)를 가지고 적절한 연결강도를 찾는다[13].

먼저 입력노드의 값을 입력받아 은닉층의 연결강도을 곱하고 편의연결강도을 더한 후 시그모이드 함수(sigmoid function)를 통하여 출력층에 전달한다. 출력층은 은닉층의 각 노드의 값을 받아들여 다시 출력층의 연결강도을 곱하고 편의연결강도을 더한 후 시그모이드 함수를 통하여 출력한다. 최종에 제시된 출력값을 가지고 사전에 주어진 목표값과 비교하는 데 평균제곱오차를 이용한다. 평균제곱오차가 가장 작은 경우에 목표값에 가장 근사함으로 최종 출력값이 목표값에 가장 근사하게 되는 각 층 각 노드의 연결강도와 편의연결강도을 조절하여 학습을 한다[1][5].

학습은 출력값과 목표값의 차이가 최소가 되도록 연결강도를 계속하여 변화하는 과정인데 연결 강도의 변화는 다음과 같다.

$$w_m = w_{m-1} + C_m \qquad (1)$$

여기서 w_m : 학습횟수(epoch) m에서의 연결강도 w의 값

C_m : 학습횟수 m에서의 연결강도 w의 변화량 이다.

변화량 C_m은 연결강도의 수정이 현재의 수정값 뿐만 아니라 이전 단계에서의 수정값도 고려하여 학습하는 모멘텀(momentum)을 이용한 적합적인 학습규칙 방법(adaptive learning rates method)을 사용하였다. 이것은 일명 delta-bar-delta 라고도 한다.

$$C_m = \mu C_{m-1} - (1 - \mu) e_m d_m \qquad (2)$$

여기서 $0 \le \mu \le 1$로서 $\mu = 0$이면 C_m은 과거의 연결강도의 변화량에는 영향을 받지 않으며 μ가 클수록 과거의 연결강도의 변화량에 크게 영향을 받는다. 또한 $d_m = \sum_{n=1}^{N} \left(\frac{\partial E}{\partial w_m} \right)_n$ 은 학습횟수 m에서의 모든 예제에 대한 연결강도에 대한 오차의 편미분의 합이다. d_m의 부호가 양의 방향이면 연결강도가 내려감에 따라 오차가 감소하지만 d_m의 부호가 음의 방향이면 연결강도가 올라감에 따라 오차가 감소하게 된다. e_m은 학습횟수 m에서의 d_m의 방향과 과거오차의 편미분의 방향 f_m에 영향을 받는다.

$$\text{즉 } e_m = \begin{cases} e_{m-1} + x, & d_m f_m > 0 \\ e_{m-1} \times \phi, & d_m f_m \le 0 \end{cases} \qquad (3)$$

이다.

오차가 현재 연결강도에서 감소하는 방향이 최근 감소한 방향과 같을 때는 e값을 x만큼 더해주고 현재 감소하는 방향이 최근 감소한 방향과 반대일 때는 e값을 ϕ만큼 곱해준다. 여기서 x와 ϕ의 값은 0과 1사이의 값을 선택하며 변수 x, ϕ, θ, μ는 사용자가 적절히 조절하여 최적해를 찾는다[13][14][16].

Ⅲ. 비중심F분포의 확률계산식

본 논문에서는 비중심F분포의 확률계산을 하기 위하여 입력층의 입력노드(독립변수)가 4개 (분자와 분모의 자유도, 비중심모수, 백분위수) 이며 출력층의 출력노드(종속변수)가 1개 (확률) 그리고 1개의 은닉층으로 구성된 다층퍼셉트론 네트워크로 부터 역전파 학습알고리즘을 적용한 결과를 제시하였으며 비중심F분포는 다음과 같이 정의한다.

■ 정 의

자유도가 n_1과 n_2인 두 독립인 x^2통계량에 대하여 다음과 같은 비(ratio)를 F분포라 한다.

$$F_{n_1, n_2} = \left\{ \frac{x^2_{n_1}}{n_1} \right\} \left\{ \frac{x^2_{n_2}}{n_2} \right\}^{-1}$$

여기서 중심 x^2이 비중심 x^2일때 자유도가 n_1, n_2이고 비중심모수 λ_1, λ_2를 갖는 이중비중심F분포(doubly noncentral F distribution)는 다음과 같다.

$$F^{\lambda_1, \lambda_2}_{n_1, n_2} = \left\{ \frac{x^{12}_{n_1}(\lambda_1)}{n_1} \right\} \left\{ \frac{x^{12}_{n_2}(\lambda_2)}{n_2} \right\}^{-1}$$

이중비중심F분포에서 $\lambda_2 = 0$(또는 $\lambda_1 = 0$)인 경우 단일비중심F분포(singly noncentral F distribution)라 하며 일반적으로 단일비중심F분포를 간단히 비중심F분포라 하며 $F^{\lambda}_{n_1, n_2}$로 나타낸다[10].

1949년 Patnaik가 제시한 비중심F분포의 누적함수는 다음과 같다.

$$F^{\lambda}_{n_1, n_2}(x) = \sum_{j=0}^{\infty} P(j) I_w \left(\frac{n_1}{2} + j, \frac{n_2}{2} \right)$$

여기서 $P(j) = \frac{e^{-\lambda} \lambda^j}{j!}$, $w = \frac{n_1 x}{n_2 + n_1 x}$ 이며,

I는 불완전 베타함수(incomplete beta function)이다[3][10].

3.1 모형

비중심F분포의 누적함수(Patnaik,1949)을 Mathematica Program을 이용하여 생성시킨 자료를 다음과 같은 모형으로 분류하였다.

〈표 1〉 모형 1

〈Table 1〉 Model 1

자 료	확 률 (P)	분자자유도(n_1)	분모자유도(n_2)	비중심모수(λ)
학습자료	P = 0.95 ~ 0.99	$n_1 = 1$	$n_2 = 2$	λ = 1 λ = 5
			$n_2 = 5$	λ = 3 λ = 9
		$n_1 = 2$	$n_2 = 2$	λ = 2 λ = 6
			$n_2 = 5$	λ = 4 λ = 12
검정자료	P = 0.95 ~ 0.99	$n_1 = 1$	$n_2 = 2$	λ = 3 λ = 7
			$n_2 = 5$	λ = 6 λ = 12
		$n_1 = 2$	$n_2 = 2$	λ = 4 λ = 8
			$n_2 = 5$	λ = 8 λ = 16

〈표 2〉 모형 2

〈Table 2〉 Model 2

자 료	확 률 (P)	분자자유도(n_1)	분모자유도(n_2)	비중심모수(λ)
학습자료	P = 0.95 ~ 0.99	$n_1 = 5$	$n_2 = 5$	λ = 5 λ = 15
		$n_1 = 10$	$n_2 = 10$	λ = 10 λ = 30
			$n_2 = 15$	λ = 13 λ = 39
			$n_2 = 20$	λ = 15 λ = 45
검정자료	P = 0.95 ~ 0.99	$n_1 = 5$	$n_2 = 5$	λ = 10 λ = 20
		$n_1 = 10$	$n_2 = 10$	λ = 20 λ = 40
			$n_2 = 15$	λ = 26 λ = 52
			$n_2 = 20$	λ = 30 λ = 60

〈표 3〉 모형 3

〈Table 3〉 Model 3

자 료	확 률 (P)	분자자유도(n_1)	분모자유도(n_2)	비중심모수(λ)
학습자료	P = 0.95 −0.99	n_1 = 15	n_2 = 15	λ = 15 λ = 45
			n_2 = 20	λ = 17 λ = 51
		n_1 = 20	n_2 = 20	λ = 20 λ = 60
검정자료	P = 0.95 −0.99	n_1 = 15	n_2 = 15	λ = 30 λ = 60
			n_2 = 20	λ = 34 λ = 68
		n_1 = 20	n_2 = 20	λ = 40 λ = 80

3.2 확률계산

일반적으로 분포함수에서 발생하는 문제로는 서론에서도 제시되었듯이 하나의 함수값을 계산하는데에도 많은 시간이 소요된다는 것이다.

본 논문에서도 비중심F분포의 누적함수값을 Patnaik가 제시한 누적함수식으로부터 직접 계산하여 정확한 함수값을 구할 수 있지만 하나의 함수값을 계산하는데에도 많은 시간이 소요됨으로 효율적이라고 할 수 없다. 그러나 신경망이론을 적용하는 경우 비록 최적의 모수를 찾는 데 있어 많은 시간과 노력이 필요하지만 한 번 학습이 끝난 후에는 비중심F분포의 함수값을 소요되는 시간없이 구할 수 있게 된다.

신경망을 적용하기 위하여 먼저 Patnaik가 제시한 비중심F분포의 누적함수식으로부터 직접 계산한 Patnaik의 함수값을 학습자료로 사용하였다.

다음은 각 모형에 대한 학습자료이다.

〈표 4〉 모형 1의 학습자료

〈Table 4〉 Training data of Model 1

분자의 자유도	분모의 자유도	비중심모수	백분위수	확 률
1	2	1	37.5	0.949997
1	2	1	47.5	0.959998
1	2	1	64.1	0.969969
1	2	1	97.5	0.979999
1	2	1	197	0.989974
1	2	5	115	0.949940
1	2	5	145	0.959960
1	2	5	195	0.969976

1	2	5	295	0.979989
1	2	5	595	0.989997
1	5	3	22	0.946416
1	5	3	25	0.957506
1	5	3	30	0.969900
1	5	3	36	0.978999
1	5	3	51	0.989816
1	5	9	50	0.949205
1	5	9	56	0.959330
1	5	9	65	0.969897
1	5	9	79	0.979959
1	5	9	108	0.989830
2	2	2	38	0.949692
2	2	2	48	0.959802
2	2	2	64	0.969583
2	2	2	98	0.979950
2	2	2	198	0.989987
2	2	6	77	0.949931
2	2	6	97	0.959955
2	2	6	130	0.969898
2	2	6	197	0.979988
2	2	6	397	0.989996
2	5	4	15	0.944367
2	5	4	17	0.956045
2	5	4	20	0.967986
2	5	4	25	0.979657
2	5	4	34	0.989426
2	5	12	33	0.947046
2	5	12	37	0.957838
2	5	12	43	0.969007
2	5	12	52	0.979251
2	5	12	72	0.989833

〈표 5〉 모형 2의 학습자료

〈Table 5〉 Training data of Model 2

분자의 자유도	분모의 자유도	비중심모수	백분위수	확 률
5	5	5	9	0.940882
5	5	5	10	0.951846
5	5	5	12	0.966627
5	5	5	15	0.979068
5	5	5	21	0.989952
5	5	15	18	0.945523
5	5	15	20	0.955969
5	5	15	24	0.969857
5	5	15	29	0.979911
5	5	15	39	0.989579

10	10	10	5.7	0.948388
10	10	10	6.1	0.958201
10	10	10	6.7	0.969111
10	10	10	7.6	0.979806
10	10	10	9.2	0.989789
10	10	30	10.9	0.948724
10	10	30	11.7	0.959261
10	10	30	12.8	0.969881
10	10	30	14.3	0.979551
10	10	30	17.3	0.989848
10	15	13	5.5	0.948625
10	15	13	5.8	0.958053
10	15	13	6.3	0.969832
10	15	13	6.9	0.979404
10	15	13	8.1	0.989967
10	15	39	11	0.948399
10	15	39	11.7	0.959918
10	15	39	12.5	0.969757
10	15	39	13.7	0.979895
10	15	39	15.8	0.989756
10	20	15	5.4	0.947068
10	20	15	5.7	0.958100
10	20	15	6.1	0.969192
10	20	15	6.6	0.978868
10	20	15	7.6	0.989784
10	20	45	11.2	0.949991
10	20	45	11.7	0.959374
10	20	45	12.4	0.969525
10	20	45	13.4	0.979631
10	20	45	15.2	0.989883

〈표 6〉 모형 3의 학습자료

〈Table 6〉 Training data of Model 3

분자의 자유도	분모의 자유도	비중심모수	백분위수	확 률
15	15	15	4.6	0.947679
15	15	15	4.9	0.959252
15	15	15	5.2	0.968086
15	15	15	5.7	0.978490
15	15	15	6.7	0.989762
15	15	45	8.9	0.949977
15	15	45	9.3	0.958349
15	15	45	10	0.969551
15	15	45	10.9	0.979365
15	15	45	12.6	0.989687
15	15	17	4.5	0.949671
15	20	17	4.7	0.958670

15	20	17	5	0.969135
15	20	17	5.4	0.978934
15	20	17	6.2	0.989912
15	20	51	8.8	0.949604
15	20	51	9.2	0.959382
15	20	51	9.7	0.968886
15	20	51	10.5	0.979525
15	20	51	11.9	0.989895
20	20	20	4.1	0.948839
20	20	20	4.3	0.959034
20	20	20	4.5	0.967122
20	20	20	4.9	0.978654
20	20	20	5.6	0.989694
20	20	60	7.9	0.949908
20	20	60	8.2	0.958318
20	20	60	8.7	0.969209
20	20	60	9.4	0.979681
20	20	60	10.6	0.989784

자료를 앞에서 제시한 모형에 따라 분류하여 학습한 결과 다음과 같은 모수에서 학습오차가 최저가 됨을 알 수 있었다.

모형 1, 2, 3 모두가 12개의 은닉노드 그리고 모수가 $x = 0.009$, $\phi = 0.3$, $\mu = 0.4$, $\theta = 0.3$에서 최적의 값을 갖는다. 따라서 이 모수로부터 구한 최종 학습결과의 연결강도를 가지고 검정자료로부터 비중심F 분포의 확률계산을 한다.

다음은 자유도와 비중심모수에 대하여 학습자료(110개)와 검정자료(110개)중에서 일부만을 비교하고 있으며 각 각의 전체자료에 대한 오차는 플롯을 이용하여 보여주고 있다.

〈표 7〉 Patnaik의 함수값과 신경망이론에 의한 함수값 비교(학습자료)

〈Table 7〉 Comparisons of the Patnaik's values and the results obtained neural network theory(training data)

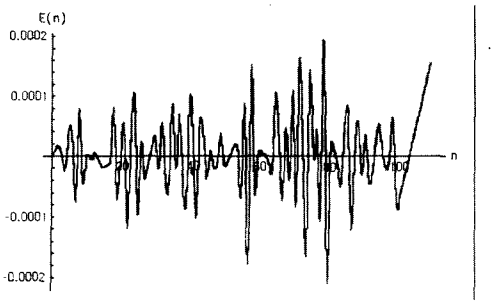
분자의자유도	분모의자유도	비중심모수	백분위수	Patnaik의확률	신경망의확률	오 차
1	2	1	37.5	0.949997	0.950004	0.000007
1	2	5	145	0.959960	0.960039	0.000079
1	5	3	30	0.969900	0.969875	0.000025
1	5	9	79	0.979959	0.980032	0.000073
2	2	2	198	0.989987	0.990087	0.000100
2	5	4	20	0.967986	0.968018	0.000032
2	5	12	52	0.979251	0.979337	0.000086
5	5	5	21	0.989952	0.989991	0.000039
5	5	15	24	0.969857	0.969870	0.000013
10	10	10	7.6	0.979806	0.979795	0.000011
10	10	30	17.3	0.989848	0.989844	0.000004
10	15	39	11	0.948399	0.948376	0.000023

10	20	15	5.7	0.958100	0.957930	0.000170
10	20	45	12.4	0.969525	0.969639	0.000114
15	15	15	5.7	0.978490	0.978542	0.000052
15	15	45	12.6	0.989687	0.989721	0.000034
15	20	17	6.2	0.989912	0.989860	0.000052
15	20	51	10.5	0.979525	0.979459	0.000066
20	20	20	4.5	0.967122	0.967095	0.000027
20	20	60	8.2	0.958318	0.958185	0.000133

〈표 8〉 Patnaik의 함수값과 신경망이론에 의한 함수값 비교(검정자료)

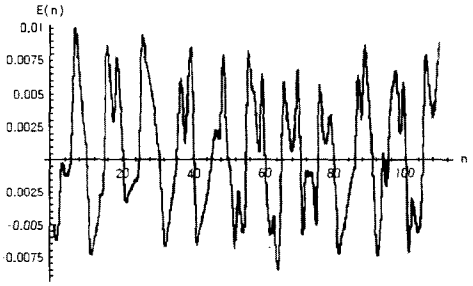
〈Table 8〉 Comparisons of the Patnaik's values and the results obtained neural network theory(validation data)

분자의자유도	분모의자유도	비중심모수	백분위수	Patnaik의확률	신경망의확률	오 차
1	2	3	196	0.979975	0.982733	0.00275
1	2	7	794	0.989998	0.988550	0.001448
1	5	6	80	0.989814	0.991718	0.001904
1	5	12	99	0.979789	0.972091	0.007698
2	2	4	297	0.989977	0.990701	0.000724
2	2	8	121	0.959812	0.952130	0.007682
2	5	8	32	0.969388	0.974584	0.005196
2	5	16	90	0.989775	0.981642	0.008133
5	5	10	30	0.989640	0.991179	0.001539
5	5	20	23	0.949092	0.946322	0.002770
10	10	20	8.9	0.958540	0.955746	0.002784
10	10	40	15.7	0.969402	0.964809	0.004593
10	15	26	10.3	0.979399	0.989041	0.009642
10	15	52	19.6	0.989802	0.983200	0.006602
10	20	30	8.3	0.948188	0.944346	0.003842
10	20	60	14.6	0.959086	0.957297	0.001789
15	15	30	7.6	0.968684	0.973720	0.005036
15	15	60	13.5	0.979869	0.971408	0.008461
15	20	34	9	0.989366	0.991707	0.002341
20	20	68	10.9	0.949250	0.945679	0.003571
20	20	40	6.3	0.959703	0.952970	0.006733
20	20	80	10.7	0.968614	0.965680	0.002934



(그림 1) 학습결과의 오차 : 오차 $E(n)$은 Patnaik의 함수값과 신경망 방법을 이용한 함수값과의 차이이며, $E(80) = 2.14 \times 10^{-4}$의 최대오차를 나타내고 있다.

(Fig. 1) Errors for training data : Maximum error is $E(80) = 2.14 \times 10^{-4}$



(그림 2) 검정결과의 오차 : 오차 $E(n)$은 Patnaik의 함수값과 신경망 방법을 이용한 함수값과의 차이이며, $E(7) = 9.799 \times 10^{-3}$의 최대오차를 나타내고 있다.

(Fig. 2) Errors for validation data : Maximum error is $E(7) = 9.799 \times 10^{-3}$

Patnaik의 함수값과 비교하여 신경망을 이용한 학습과 검정결과는 다음과 같다.

학습자료의 함수값은 대략 소수점 세째자리까지 Patnaik의 함수값과 동일하며 Patnaik의 함수값과의 최대오차는 2.14×10^{-4}이다. 또한 검정자료의 함수값은 대략 소수점 둘째자리까지 Patnaik의 함수값과 동일하며 Patnaik의 함수값과의 최대오차는 9.799×1^{-3}임을 알 수 있다.

Patnaik의 함수값과 신경망에 의한 함수값과의 차이인 검정자료 오차로부터 좋은 결과임을 알 수 있으며 또한, 최적의 모수로부터 구한 최종 학습결과의 연결강도를 가지고 우리가 구하고자 하는 비중심F분포의 누적함수값 여러개를 동시에 소요되는 시간 없이 구할 수 있다. 이로부터 Patnaik 누적함수식에 의한 방법보다 신경망을 이용한 접근 방법이 효율적임을 알 수 있다.

Ⅳ. 결 론

본 논문은 새로운 접근방법으로 신경망 이론을 적용하여 비중심F분포의 확률 계산을 하였다. 이 신경망 이론은 다층의 구조를 갖는 복잡한 신경망 역전파 학습 알고리즘으로 연결강도 공간에서 오차표면의 기울기 감소를 기본으로 하였다.

비중심F분포의 확률 계산을 Patnaik의 함수값과 신경망을 이용한 함수값과 비교하여 학습결과와 검정결과 그리고 그에 따른 오차들을 플롯을 이용하여 제시하였다. 신경망을 이용한 함수값과 Patnaik의 함수값을 비교한 결과 다음과 같은 결론을 얻을 수 있었다.

첫째, 학습자료의 함수값은 대략 소수점 세째자리까지 Patnaik의 함수값과 정확하며 검정자료의 함수값은 소수점 둘째자리까지 Patnaik의 함수값과 정확하게 나타남을 알 수 있다. 둘째, 학습하는데 소요되는 시간, 즉 신경망의 최적의 모수를 찾는 데 있어 많은 시간과 노력이 필요하지만 한 번 학습이 끝난 후에는 최종의 연결강도를 가지고 소요되는 시간 없이 비중심F분포의 확률계산을 한다. 셋째, 정

확성과 계산속도를 고려하여 볼 때 비중심F분포의 확률계산을 Patnaik의 누적함수식에 의한 방법보다 신경망을 이용한 방법이 효율적임을 알 수 있다.

참 고 문 헌

[1] 김대수, 신경망 이론과 응용 (Ⅰ) (Ⅱ), 하이테크정보, 1994

[2] 이성환, 패턴인식의 원리 (Ⅰ) (Ⅱ), 홍릉과학출판사, 1994

[3] Bickel, P. J. and Doksum, K. A., Mathematical Statistics : Basic ideas and selected topics, SanFrancisco : Holden-Day, 1977

[4] Chattamvelli, R., On the doubly non-central F distribution, Computational Statistics and Data Analysis, Vol. 20, pp. 481-489, 1995

[5] Cheng, B. and Titterington, D. M., Neural Networks : A review from a statistical perspective, Statistical Science, Vol.9, pp.2-54, 1994

[6] Friedman, J. H. and Stuetzle, W., Projection pursuit regression, Journal of the American Statistical Association, Vol.76, pp.817-823, 1981

[7] Hwang, J. N and Maechler, H.Li.M. and Martin, D. and Schimert, J., Regression modeling in backpropagation and projection pursuit learning, IEEE transactions on neural networks, Vol.5, pp.342-353, 1994

[8] Jones, L.K., A simple lemma on greedy approximation in Hilbert space and convergence rates for projection pursuit regression and neural network training, The Annals of Statistics, Vol.20, pp.608-613, 1992

[9] John, H. and Anders, K. and Richard, G.P., Introduction to the Theory Neural Computation, Addison-Wesley Publishing Co. Inc, 1991

[10] Johnson, N. L. and Kotz, S., Continuous Univariate Distributions, Boston : Houghton Mifflin, 1970

[11] Pao, Y.H., Adaptive Pattern Recognition and Neural Network, Addison Wesley Publishing Co. Inc, 1989

[12] Refenes, A. N. and Zapranis, A. and Francis, G., Stock Preformance Modeling Using Neural Networks : A Comparative Study With Regression Models, Neural Networks, Vol.7, pp.375-388, 1994

[13] Rumelhart, D. E. and Geoffrey, E. H. and Ronald, J. W., Learning representations by backpropagating errors, Nature, Vol.323, pp.533-536, 1986

[14] Simth, M., Neural Networks for Statistical Modeling, New York : Van Nostrand Reinhold, 1993

[15] Specht, D. F., A general regression neural network, IEEE transactions on neural networks, Vol.2, pp.568-576, 1991

[16] White, H., Learning in artificial networks : A Statistical perspective, Neural Computation, Vol.1(4), pp.425-469, 1989

[17] White, H., Connectionist nonparametric regression : multilayer feedforward networks can learn arbitrary mappings, Neural Networks, Vol.3, pp.535-549, 1990

□ 著者紹介

구 선 희(具 宣 喜)

1986년 숙명여자대학교 수학과 졸업(이학사)
1989년 숙명여자대학교 일반대학원 수학과 졸업(이학석사)
1996년 성균관대학교 일반대학원 통계학과 졸업(경제학박사)
1996년 - 현재 전주대학교 전기전자컴퓨터 공학부 객원교수

※ 관심 분야 : 신경망 이론, 인공지능 응용분야, 전산수학