Monthly User Guide from JMP Korea

제 26호 (2019년 9월)

인공신경망(Neural Network)

* 본 Guide 의 내용과 관련한 문의는 ikju.Shin@jmp.com 으로 연락 바랍니다

인공 신경망(Neural Network)

이번 호에서는 Machine Learning 기법 중의 하나인 **인공 신경망(Neural Network)**을 활용하여 예측 모델링(Predictive Modeling)을 하는 방법에 대해 간략하게 배워 보겠습니다.

참고로 JMP에서 지원되는 Machine Learning 방법은 아래와 같습니다(Analyze / Predictive Modeling 기준)
JMP Pro에 보다 다양한 Machine Learning 방법이 탑재되어 있으며,
2019년 10월 출시될 JMP 15에는 Pattern 탐색, SVM(Support Vector Machine) 방법 등이 추가될 예정입니다.

JMP의 Machine Learning 기능	JMP 14		JMP 15	
	Machine Learning	Modeling Utilities	Machine Learning	Modeling Utilities
JMP	Partition (Decision Tree) Neural Network	**Explore Outliers** (이상치 탐색, 처리) **Explore Missing Values** (결측치 탐색 처리)		**Explore Patterns** (특이한, 기대되지 않은 패턴의 탐색, 처리)
JMP Pro	Bootstrap Forest Boosted Tree K Nearest Neighbors Naïve Bayes	**Model Comparison**(각 예측 기법들의 예측 능력 비교), **Make Validation Column** (교차 검증-Holdback 등-을 하기 위한 Column 생성), **Formula Depot**(예측 모델을 JMP 밖-Python, JAVA 등-에서 활용할 수 있는 Code 생성)	Support Vector Machine	

인공 신경망(Neural Network)*

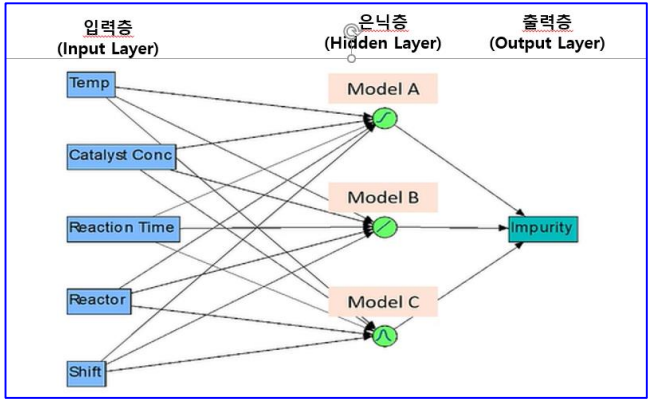
1. Neural Network의 장점

Neural Network는 인공 지능 및 통계적 모델링 분야에서 광범위하게 사용되고 있습니다. 대용량 Data, 많은 수의 예측 변수를 가지고 있는 Data에 대해서도 잘 적용될 수 있으며 다음과 같은 몇 가지 장점을 가지고 있습니다.

1) 연속형 및 범주형 반응치에 대해서도 모델링 가능
2) 다양한 형태의 예측 변수를 취급할 수 있음.
3) 교호 작용(Interaction)을 가진 Data에 대해서도 쉽게 모델링할 수 있으며
4) 비선형(Nonlinear) 관계에 대한 모델링도 가능함.

2. 입력층(Input Layer), 은닉층(Hidden Layer), 출력층(Output Layer)

Neural Network은 인간 신경 세포의 기본 단위인 Neuron의 동작 원리를 컴퓨터로 구현한 모형으로 입력층, 은닉층, 출력층이라는 개념을 활용한다. 간단히 예를 들면, 만약 대학의 학과를 선택하는 기준이 다섯 가지가 있고, A,B,C 세 명의 학생이 있다고 가정하면, 학생들마다 최종적으로 선택하는 학과는 매우 다양할 수 있는 데, 이는 학생들마다 다섯 가지 학과 선택 기준에 대한 그들의 가중치가 다르기 때문에 개별 학과에 대한 그들 각각의 최종적인 점수가 다르게 된다. Neural Network는 이러한 점수를 계산하는 과정 (은닉층의 함수)를 고유한 몇 가지 복잡한 함수를 사용하여 계산한다.



3. JMP와 JMP Pro의 차이

1) 은닉층의 수, 은닉층에 사용되는 활성 함수(Activation Function) 등에 있어서 JMP와 JMP Pro는 차이가 있다.
2) JMP는 한 개의 Hidden Layer에 대해 Hyperbolic Tangent 함수만을 사용하는 반면, JMP Pro의 경우는 두 개의 Hidden Layer를 설정할 수 있고, 다른 함수(Linear, Gaussian)를 활용할 수 있다.
3) 아래는 JMP Pro의 실행 화면 중 일부이다.

Hidden Layer Structure

Number of nodes of each activation type

Activation	Sigmoid	Identity	Radial
Layer	TanH	Linear	Gaussian
First	3	0	0
Second	0	0	0

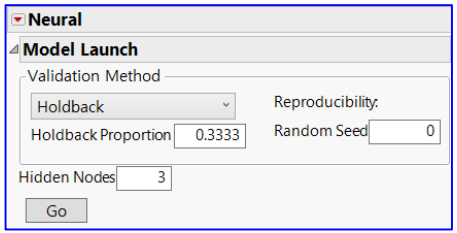
Second layer is closer to X's in two layer models.

Source : https://www.jmp.com/en_us/online-statistics-course.html

예제 1 : 연속형 Y

Sample Data : Help / Sample Data Library / Boston Housing.jmp
-주택 가격과 관련된 Data로 mvalue 변수가 Y Data이고,
나머지는 이외 관련된 지리적, 인구 통계학적 변수이다
-주택 가격(mvalue)과 다른 변수들에 대해 Neural Network 기능을 활용하여 모델링하고자 한다.

1. Analyze / Predictive Modeling / Neural에 들어가서 mvalue 변수를 Y로, 나머지 모든 변수를 X로 선택한 후 선택하여 OK를 클릭한 다음, Model Launch 화면에서 아래와 같이 입력한다. Holdback Proportion은 검증용(Validation) Data로 사용하고자 하는 Data의 비율을 말한다 (Random하게 선택된다), Hidden Nodes에 3을 입력하고 Go를 선택한다.



1) Validation Method
-Excluded rows holdback : Data Table에 Excluded / Unexcluded 로 row를 구분하였을 경우 Unexcluded row는 Training 용으로, Excluded row는 Validation set 으로 활용된다
-Holdback : Training set과 Validation set을 랜덤하게 분할 (Holdback proportion : Validation set의 비율)
-K Fold(전체 Data를 K개의 subset으로 분할하여 Validation 하는 방법)
2) Hidden Nodes : 은닉층

2. 분석용(Training) Data와 검증용(Validation) Data의 R-Square 값이 모두 높으므로 예측력이 높은 모델이 생성되었음을 알 수 있다.

Neural

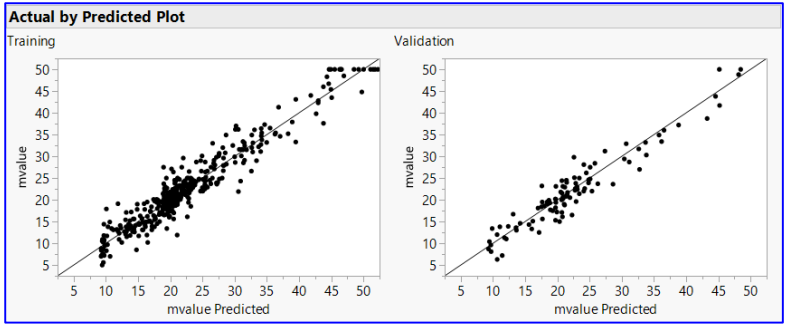
Validation: Random KFold

Model Launch

Model NTanH(3)

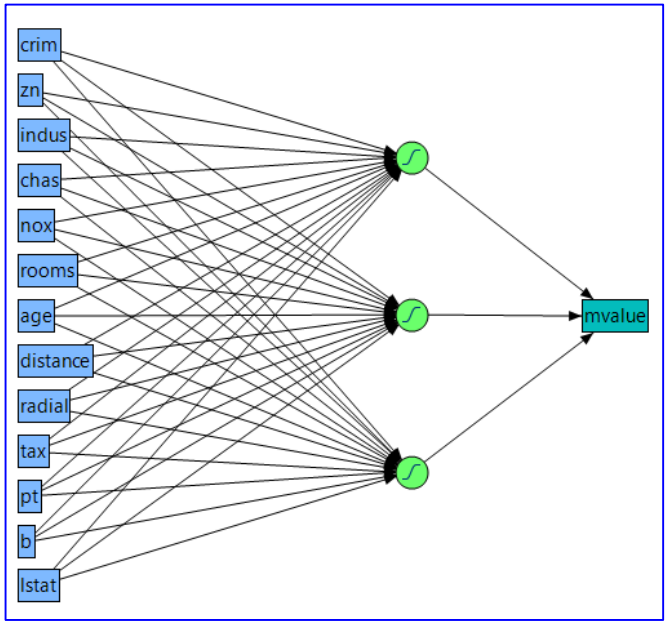
Training		Validation	
mvalue		mvalue	
Measures	Value	Measures	Value
RSquare	0.9213977	RSquare	0.9206037
RMSE	2.5813047	RMSE	2.5672311
Mean Abs Dev	1.9616048	Mean Abs Dev	2.0108876
-LogLikelihood	958.72956	-LogLikelihood	238.53841
SSE	2698.5692	SSE	665.65821
Sum Freq	405	Sum Freq	101

3. 그 외에도 실제 값과 예측 값을 그래프로 표현한 Actual by Predicted Plot을 통해서도 모형의 예측력을 확인할 수 있다
(▼Model NTanH / Actual by Predicted Plot 선택)

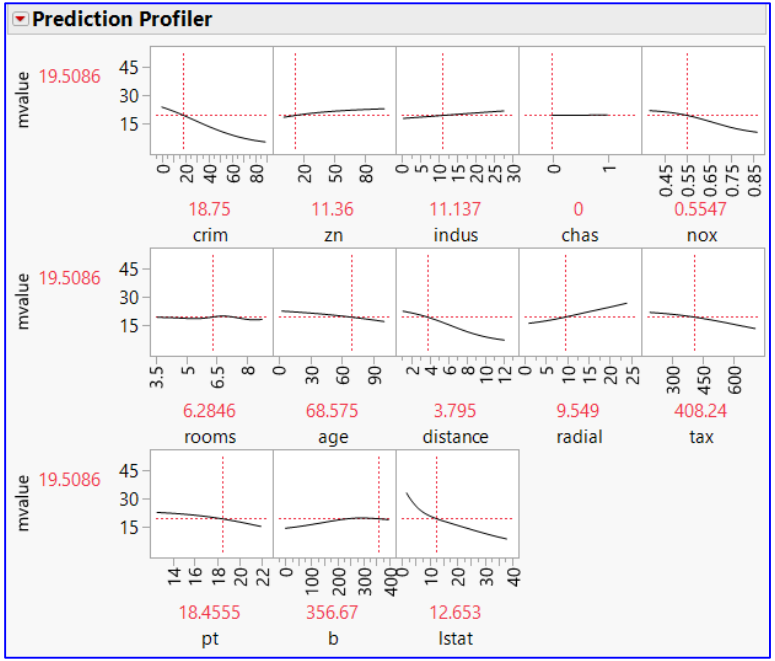


예제 1 : 연속형 Y

4. ▼Model NTanH의 Diagram을 클릭하면 Neural Network의 모델을 시각적으로 확인할 수 있다. 왼쪽부터 입력층(Input Layer), 은닉층(Hidden Layer), 출력층(Output Layer)이라고 한다.



5. ▼Model NTanH / Profiler 기능을 활용하여 각 X인자별로 Y 변수에 어떠한 영향을 주는 주는지를 상세하게 분석할 수 있다.



예제 1 : 연속형 Y

6. 예측 결과의 저장과 활용

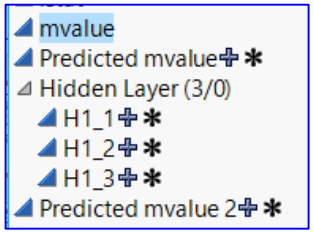
▼Model NTanH / Save Profile Formulas(및 Save Formulas)를 클릭하면, Data Table에 Y 변수에 대한 예측 값이 저장된다. 예측 값에 Formula가 저장되어 있으므로 다양한 방식으로 활용 가능하다. 예를 들어 Data Table에서 X인자에 대한 값을 입력하면, Y 변수에 대한 예측 값이 자동으로 생성된다.

	crim	zn	indus	chas	nox	rooms	age	distance	radial	tax	pt	b	lstat	mvalue	Predicted mvalue
506	0.047...	0	11.93	0	0.573	6.03	80.8	2.505	1	273	21	396.9	7.88	11.9	20.871228027
507	0.1	0	12	0	0.5	5	80	3	4	300	21	300	25	•	17.974767005

새로운 값 입력

예측된 Y 값

7. Column Panel에서 예측값 및 은닉층(Hidden Layer)의 함수를 확인할 수도 있다.



1) 예측된 Y값의 Formula는 다음과 같다

28.101252169 + 5.1203820487 • *H1_1* + -7.64527069 • *H1_2* + 11.480638198 • *H1_3*

2) 은닉층의 Formula는 다음과 같다(아래는 H1_1의 Formula의 일부분)

TanH(0.5 • (-11.58951417 + 0.012620198 • *crim* + -0.014132793 • *zn* + 0.1079437554 • *indus* + Match(*chas*, 0 ⇒ -3.846930464, 1 ⇒ -(-3.846930464 + 0)) + 6.7375926382 • *nox* + 2.2050293854 • *rooms* + -0.02455719 • *age*))

7. 위의 Formula의 결과는 ▼Model NTanH / Show Estimate를 클릭하여 확인할 수도 있다.

Estimates

Parameter	Estimate
H1_1:crim	0.01262
H1_1:zn	-0.01413
H1_1:indus	0.107944
H1_1:chas:0	-3.84693

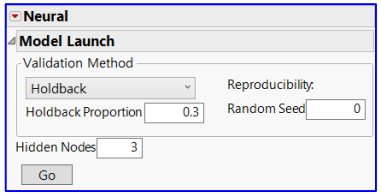
~~~~

mvalue_1:H1_1	5.120382
mvalue_2:H1_2	-7.64527
mvalue_3:H1_3	11.48064
mvalue_4:Intercept	28.10125
~~~~

예제 2 : 범주형 Y

Sample Data : Help / Sample Data Library / Diabetes.jmp
(당뇨병과 관련하여 수집된 data로서, 본 사례에서는 반응치(Response)로 혈당 수치를 상/하로 구분한 Y(Binary)을 사용한다. Age ~ Glucose까지 10개의 factor가 있다)

1. Analyze / Predictive Modeling / Neural에 들어가서 Y(Binary) 변수를 Y로, 나머지 모든 변수를 X로 선택한 후 선택하여 OK를 클릭한 다음, Model Launch 화면에서 아래와 같이 Go를 선택한다.



2. 분석용(Training) Data와 검증용(Validation) Data에 대한 분석 결과는 (R-Square 값 등) 은 아래와 같다

Model NTanH(3)

Training		Validation	
Y Binary		Y Binary	
Measures	Value	Measures	Value
Generalized RSquare	0.547519	Generalized RSquare	0.5299418
Entropy RSquare	0.4050426	Entropy RSquare	0.3878316
RMSE	0.3372323	RMSE	0.3351543
Mean Abs Dev	0.2271344	Mean Abs Dev	0.2261339
Misclassification Rate	0.1558442	Misclassification Rate	0.1716418
-LogLikelihood	107.37399	-LogLikelihood	48.336577
Sum Freq	308	Sum Freq	134

3. Confusion Matrix (Misclassification Rate, 오 분류표), Confusion Rate에 대한 결과가 있는 데, 간략히 설명하면 아래와 같다.

1) Training Data를 기준으로 설명하면 전체 Data가 308 개이므로 정확하게 판단한 비율, 즉 Low를 Low로, High를 High로 판단한 비율은
(208+52) / 308 = 0.844 이다. 이를 보통 **정확도(Accuracy)**라고 표현한다.
2) 범주별 정확도를 나타내는 **민감도(Sensitivity)**의 경우, Low 범주의 민감도는
208 / (208+18)=0.92 으로 계산된다.
이처럼 민감도는 실제 사실을 정확하게 사실로 판단해 내는 능력이라 할 수 있다.
3) 민감도(Sensitivity) 외에 **특이도(Specificity)**라는 개념도 있는 데, 예를 들어 Row 범주의 특이도는 52 / (32+52) = 0.619 로 계산된다. 이처럼 특이도는
사실이 아닌 것을 사실이 아닌 것으로 판단하는 능력이라 할 수 있다.

Confusion Matrix

Actual	Predicted Count	
Y Binary	Low	High
Low	208	16
High	32	52

Confusion Matrix

Actual	Predicted Count	
Y Binary	Low	High
Low	86	11
High	12	25

Confusion Rates

Actual	Predicted Rate	
Y Binary	Low	High
Low	0.929	0.071
High	0.381	0.619

Confusion Rates

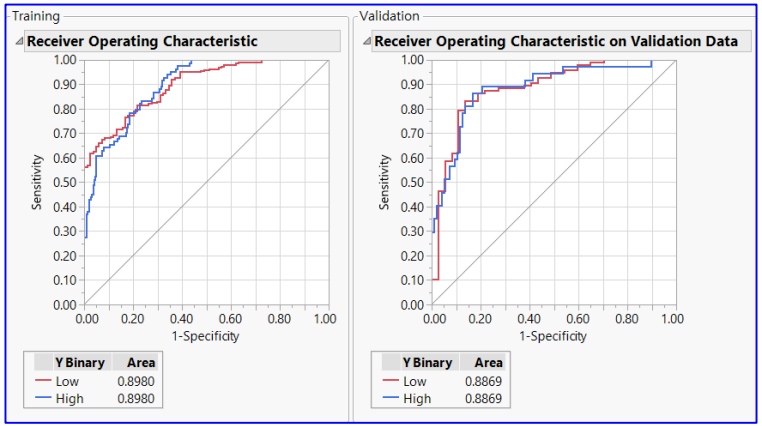
Actual	Predicted Rate	
Y Binary	Low	High
Low	0.887	0.113
High	0.324	0.676

4) 민감도와 특이도를 요약하면 아래 표와 같다.
다시 말하면, 민감도(Sensitivity)는 True Positive Rate로 TP/(TP+FN)으로 계산되며, 특이도(Specificity)는 TN/(FP+TN)으로 계산되고, '1-Specificity'는 False Positive Rate로 FP/(FP+TN)으로 계산된다.

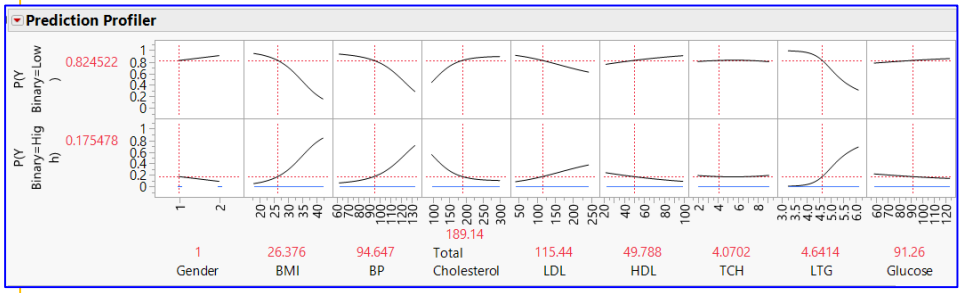
실제(Actual)	예측(Predict)	
	Positive	Negative
Positive	TP(True Positive)	FN(False Negative)
Negative	FP(False Positive)	TN(True Negative)

예제 2 : 범주형 Y

4. ▼Model NTanH / ROC Curve를 선택하면 아래와 같이 ROC Curve가 Display된다. ROC(Receiver Operating Characteristic)는 범주형 모델에서 모델의 정확도를 평가하는 방법으로, ROC 곡선의 아랫부분(아래 그래프의 왼쪽 상단에 있는 곡선이 ROC 곡선이다)의 면적이 1에 가까울수록 정확한 모델이라 할 수 있다.



5. ▼Model NTanH / Profiler (또는 Categorical Profiler) 기능을 활용하여 각 X 인자별로 Y 변수에 어떠한 영향을 주는 지를 상세하게 분석, 예측할 수 있다.



6. ▼Profiler (또는 Categorical Profiler) / Variable Importance 에서 Independent Uniform Inputs 등을 선택하면 인자별 효과를 확인할 수 있다.