Portland State
UNIVERSITY

MATLAB을 이용한
고강도강 겹치기 레이저 용접부의 모델링

- SNN (shallow), DNN (deep) -

안 내

- 본 자료는 아래의 사람들이 만들었습니다.
  유현정 (Portland State University)
  김철희 (한국생산기술연구원, Portland State University)

- 예제 파일은 아래에서 받을 수 있습니다.
  https://deepjoining.github.io/

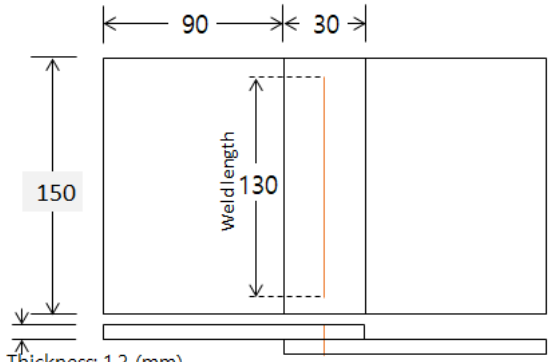
- 문의사항 및 의견: deepjoining@gmail.com

- 자료는 한국생산기술연구원 용접접합그룹 신입대학원생 교육자료입니다.
  일체의 다른 용도 사용을 금지합니다.

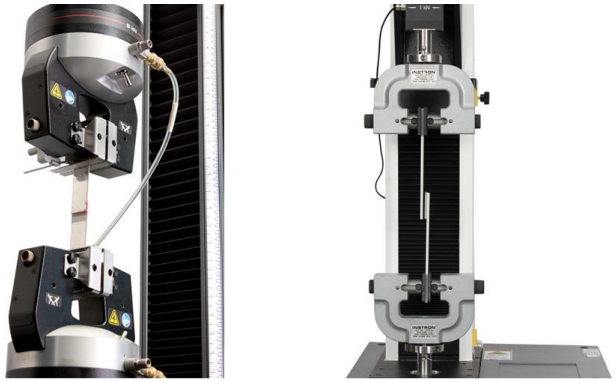
0. 풀어야 할 문제

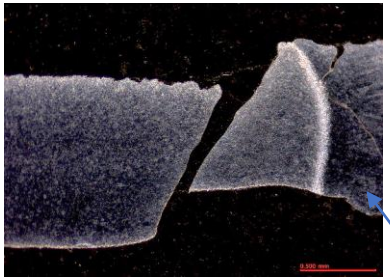
- 소재: 인장강도 590~1500 MPa급 자동차용 강판
  (cf. 연강의 경우 인장강도 270~300 MPa)
- 용접방법: 레이저 겹치기 용접
- 용접부 시험방법: 인장-전단 강도 평가
- 품질판단 기준: 파단의 위치

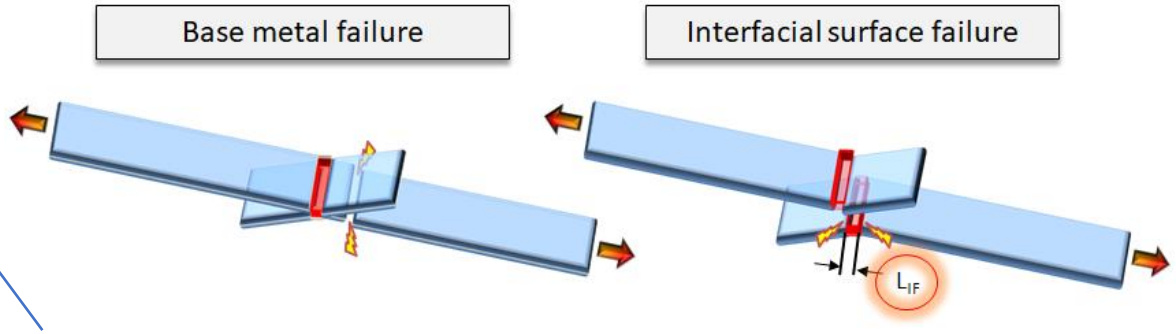
- 모델링할 문제
  * 다양한 소재 조합 및 다양한 레이저 용접조건하에서
  (1) 용접 후 용접 비드폭은 얼마인가? (회귀)
  (2) 인장-전단 시험에서 강도는? (회귀)
  (3) 인장-전단 시험에서 파단위치는? (분류)

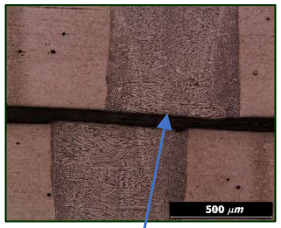


용접 실험 모식도


인장-전단 시험


HPF2.0G
용접부





용접 계면파단

1. 머신 러닝 모델 구축에 사용된 Input, Output parameter

- Input parameter

No.	1~7	8~14	15	16
Input parameter	Chemical composition of the upper sheet	Chemical composition of the lower sheet	Welding speed	Focal position

- Output parameter

	Regression model		Classification model
Output parameter	Bead width at the faying surface	Fracture load	Fracture location

- Chemical compositions

Base materials (thickness)	C	Si	Mn	P	S	Cr	B
590 DP (1.2 mm)	0.078	0.363	1.808	0.011	0.001	-	-
780 DP (1.2 mm)	0.070	0.977	2.264	0.010	0.015	-	-
980 DP (1.2 mm)	0.170	1.340	2.000	0.016	0.001	-	-
1180 CP (1.2 mm)	0.110	0.110	2.790	0.019	0.004	1.040	-
1500 HPF (1.1 mm)	0.216	0.240	1.255	0.002	0.002	0.001	0.003

1. 머신 러닝 모델 구축에 사용된 데이터

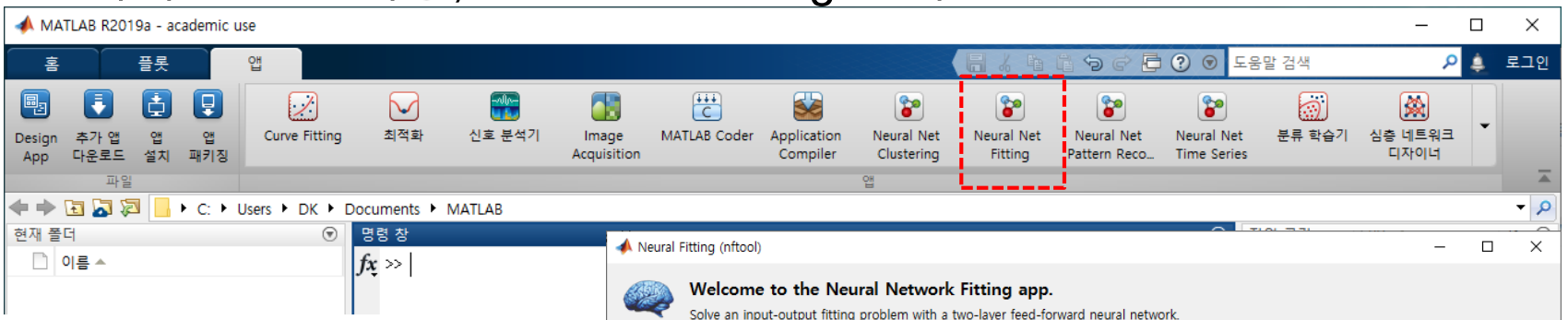
- Input parameter

- Output parameter

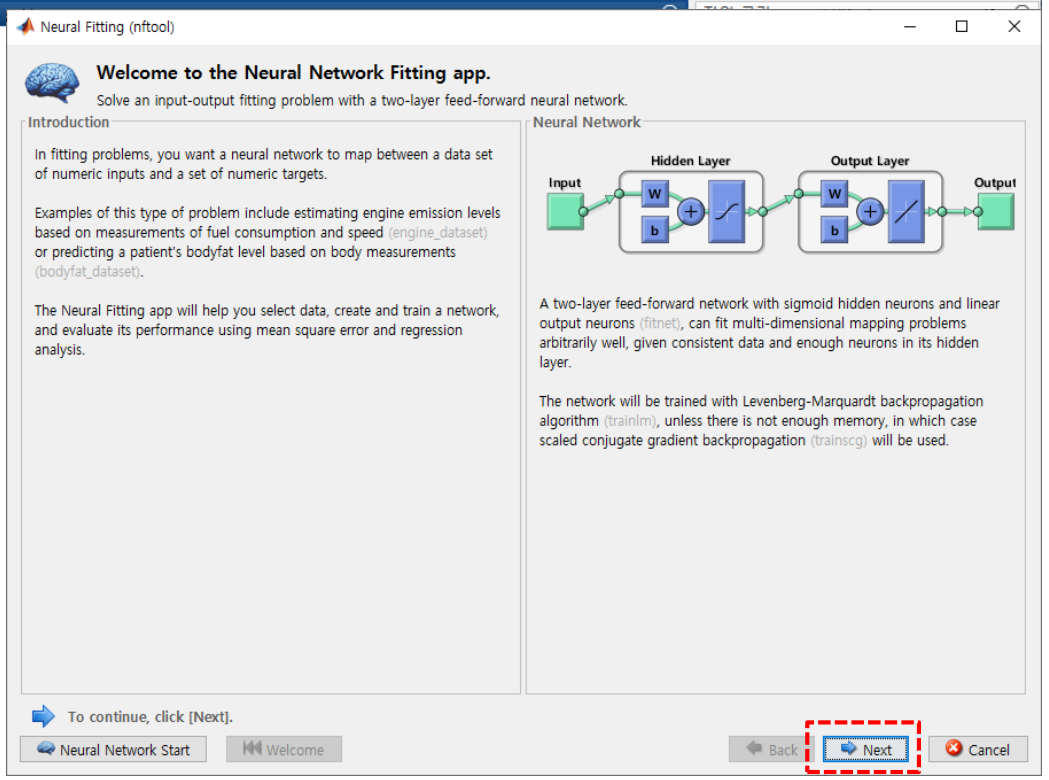
Chemical composition of upper sheet							Chemical composition of lower sheet											
C	Si	Mn	P	S	Cr	B	C	Si	Mn	P	S	Cr	B	Weldingsp	Focalposit	Bead widtl	Fracture lo	Fracture lo
0.078	0.363	1.808	0.011	0.001	0	0	0.078	0.363	1.808	0.011	0.001	0	0	70	0	0.82	15659.47	0
0.078	0.363	1.808	0.011	0.001	0	0	0.078	0.363	1.808	0.011	0.001	0	0	60	-5	0.87	16660.73	0
0.078	0.363	1.808	0.011	0.001	0	0	0.078	0.363	1.808	0.011	0.001	0	0	48	-10	1.02	18593.63	0
0.078	0.363	1.808	0.011	0.001	0	0	0.078	0.363	1.808	0.011	0.001	0	0	37	-15	1.33	18619.1	1
0.078	0.363	1.808	0.011	0.001	0	0	0.078	0.363	1.808	0.011	0.001	0	0	26	-20	1.99	18859.83	1
0.078	0.363	1.808	0.011	0.001	0	0	0.078	0.363	1.808	0.011	0.001	0	0	20	-25	2.02	18765.03	1
0.078	0.363	1.808	0.011	0.001	0	0	0.07	0.977	2.264	0.01	0.015	0	0	70	0	0.76	14681.73	0
0.078	0.363	1.808	0.011	0.001	0	0	0.07	0.977	2.264	0.01	0.015	0	0	60	-5	0.81	15620.37	0
0.078	0.363	1.808	0.011	0.001	0	0	0.07	0.977	2.264	0.01	0.015	0	0	48	-10	1.1	18561.8	1
0.078	0.363	1.808	0.011	0.001	0	0	0.07	0.977	2.264	0.01	0.015	0	0	40	-10	1.09	18555.93	1
0.078	0.363	1.808	0.011	0.001	0	0	0.07	0.977	2.264	0.01	0.015	0	0	32	-15	1.41	18899.8	1
0.078	0.363	1.808	0.011	0.001	0	0	0.07	0.977	2.264	0.01	0.015	0	0	24	-20	2.01	18918.9	1
0.078	0.363	1.808	0.011	0.001	0	0	0.17	1.34	2	0.016	0.001	0	0	70	0	0.74	16050.63	0
0.078	0.363	1.808	0.011	0.001	0	0	0.17	1.34	2	0.016	0.001	0	0	60	-5	0.86	17751.67	0

2. MATLAB에서 신경회로망 fitting (SNN)

1. Matlab 실행
2. Tab에서 앱으로 이동, Neural net fitting 선택

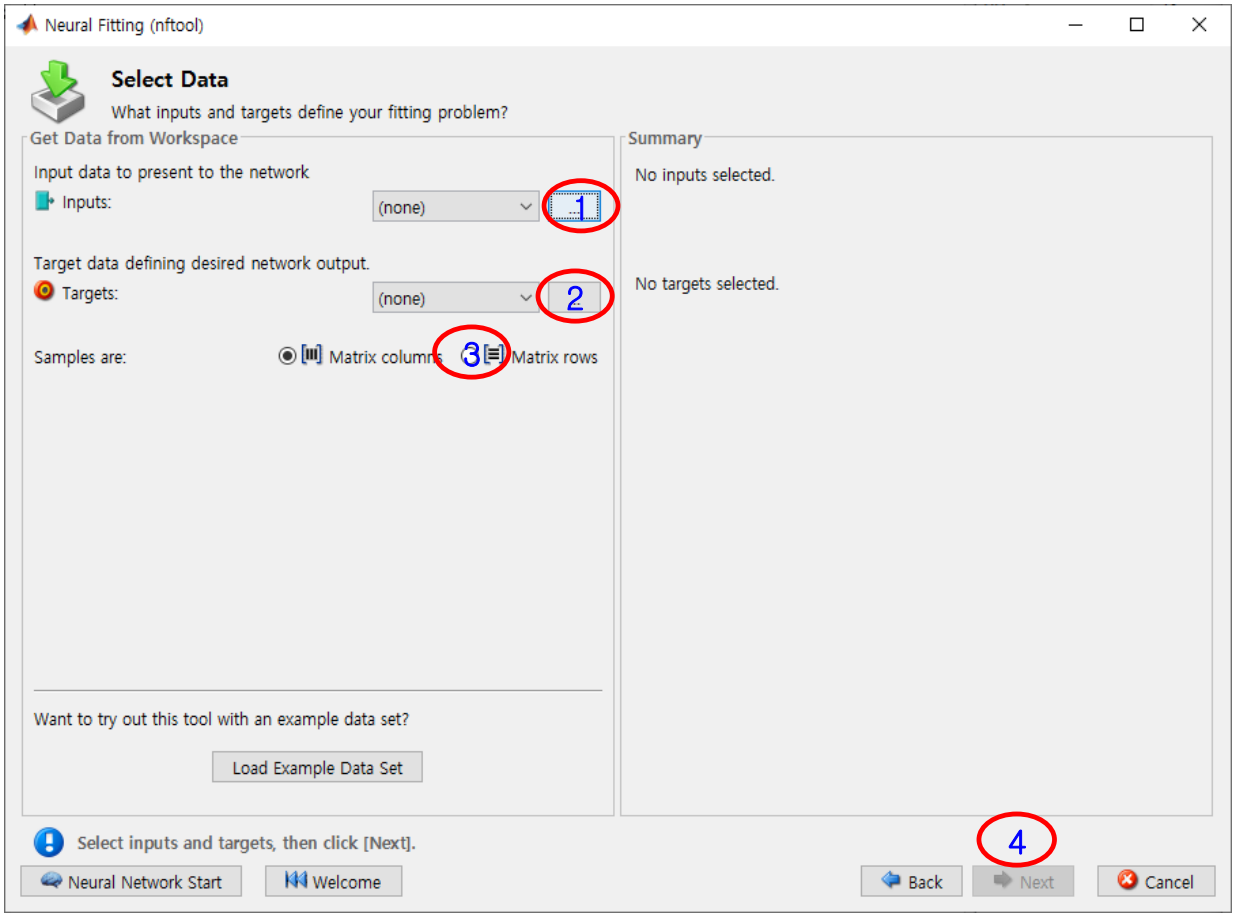


3. 새로 뜬 창에서 Next 클릭



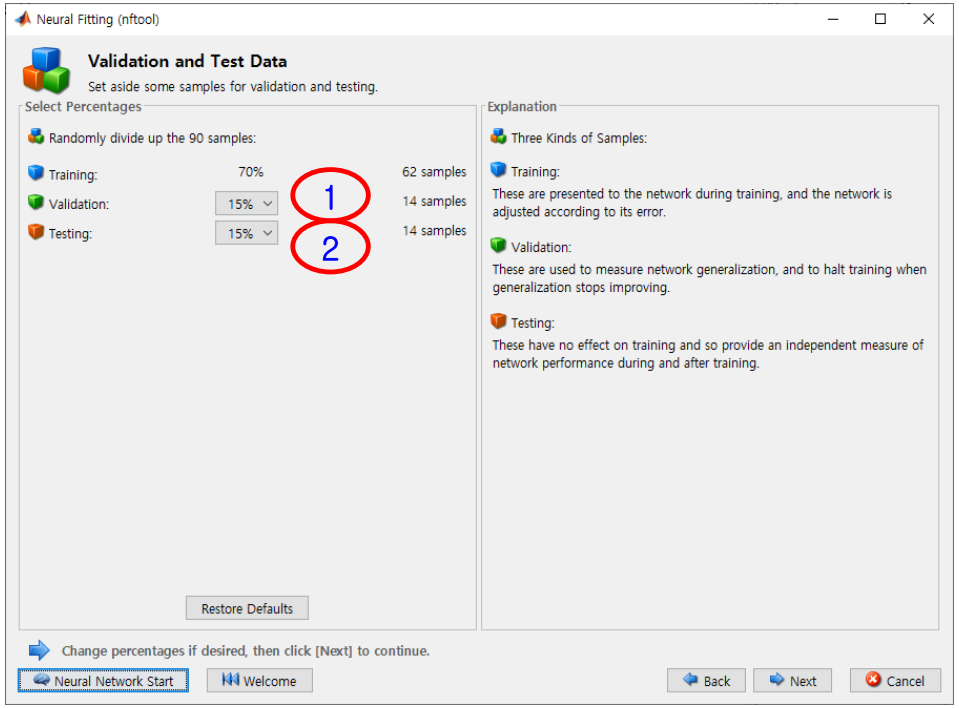
2. MATLAB에서 신경회로망 fitting (SNN)

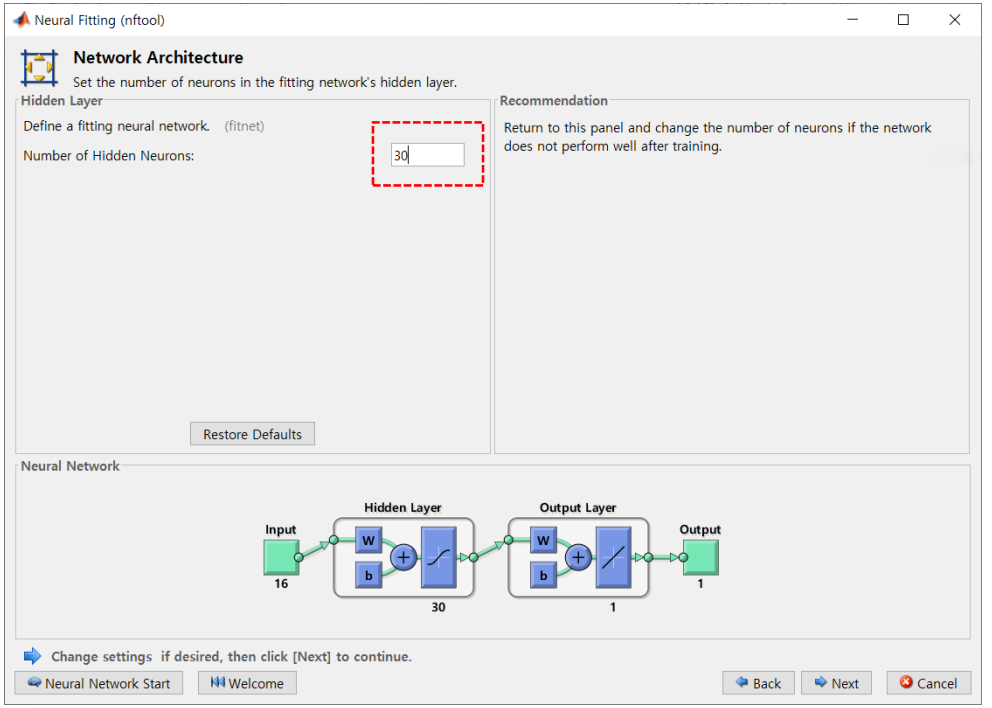
4. 1을 눌러서 input 데이터만 있는 파일을 불러옴
5. 2를 눌러서 output 데이터만 있는 파일을 불러옴
6. 3에서 Matrix rows선택하고 4-Next 선택



2. MATLAB에서 신경회로망 fitting (SNN)

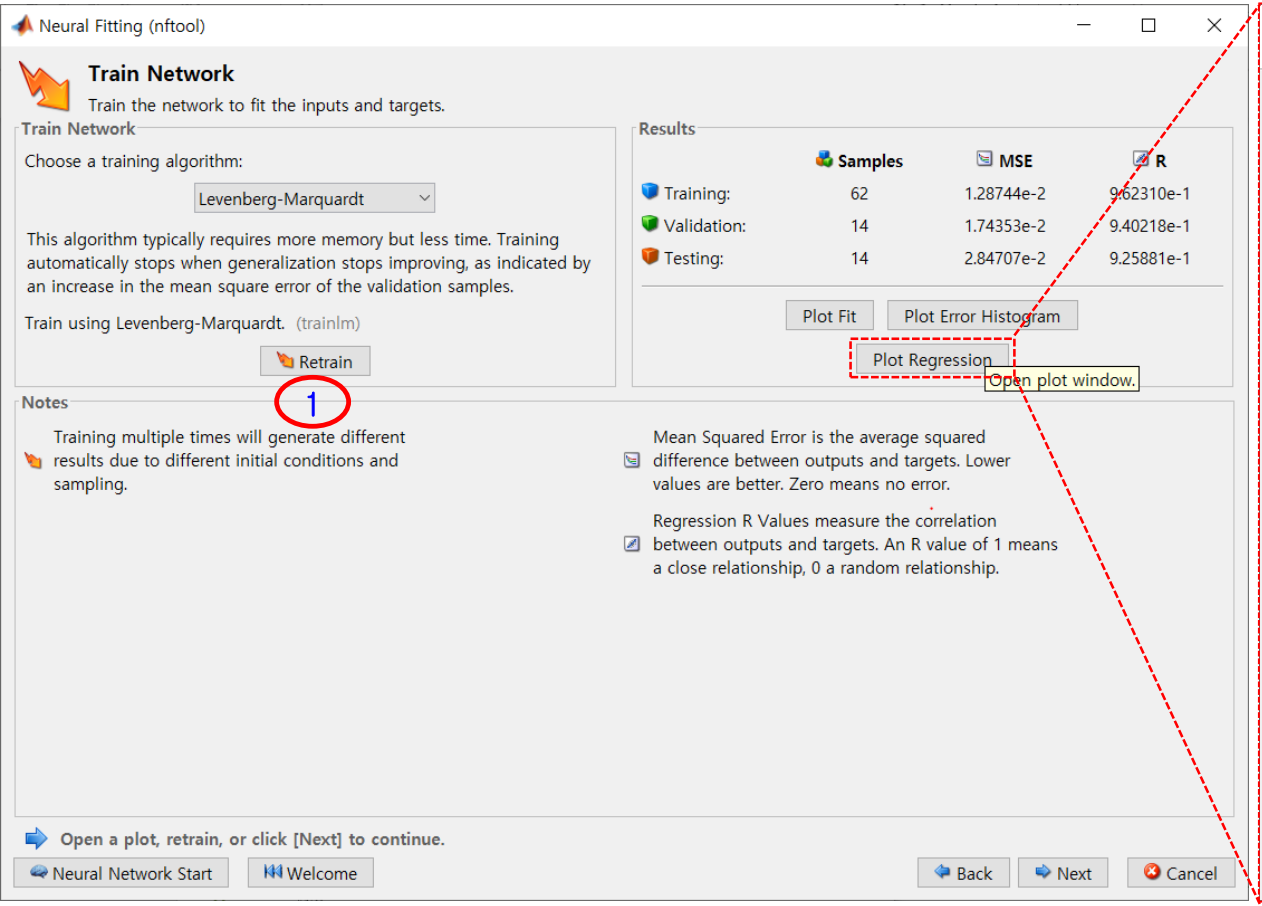
7. Validation과 testing data의 퍼센트 변경 가능
8. Next 로 넘어가면
9. Hidden layer의 노드 개수 변경 가능
10. Next로 넘어갈 것

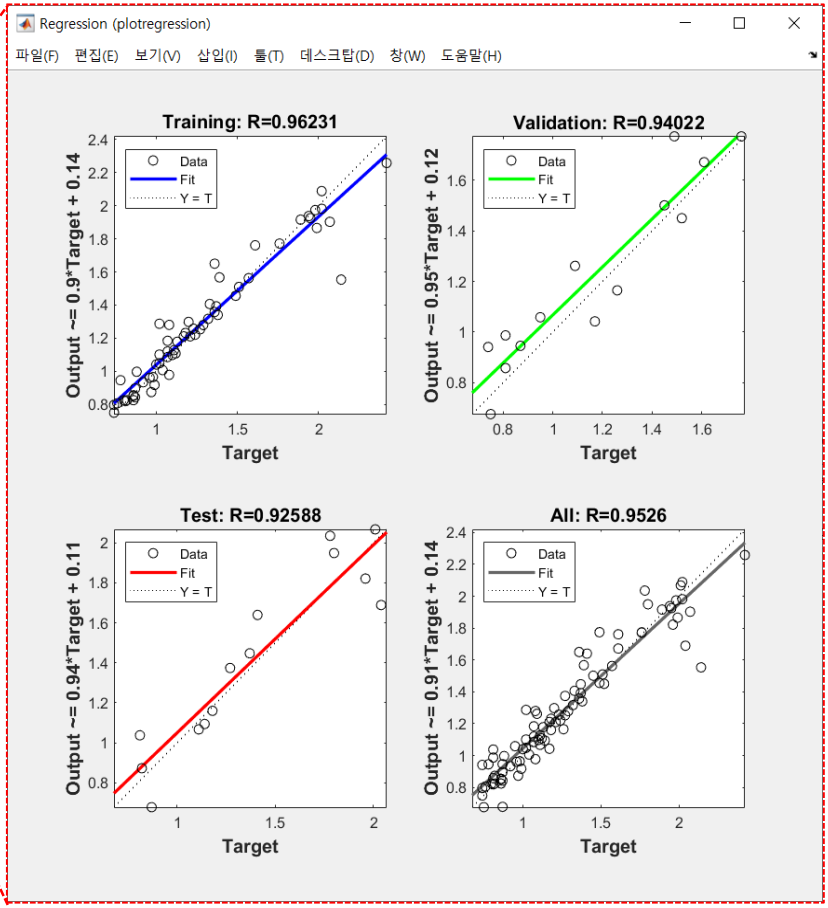




2. MATLAB에서 신경회로망 fitting (SNN)

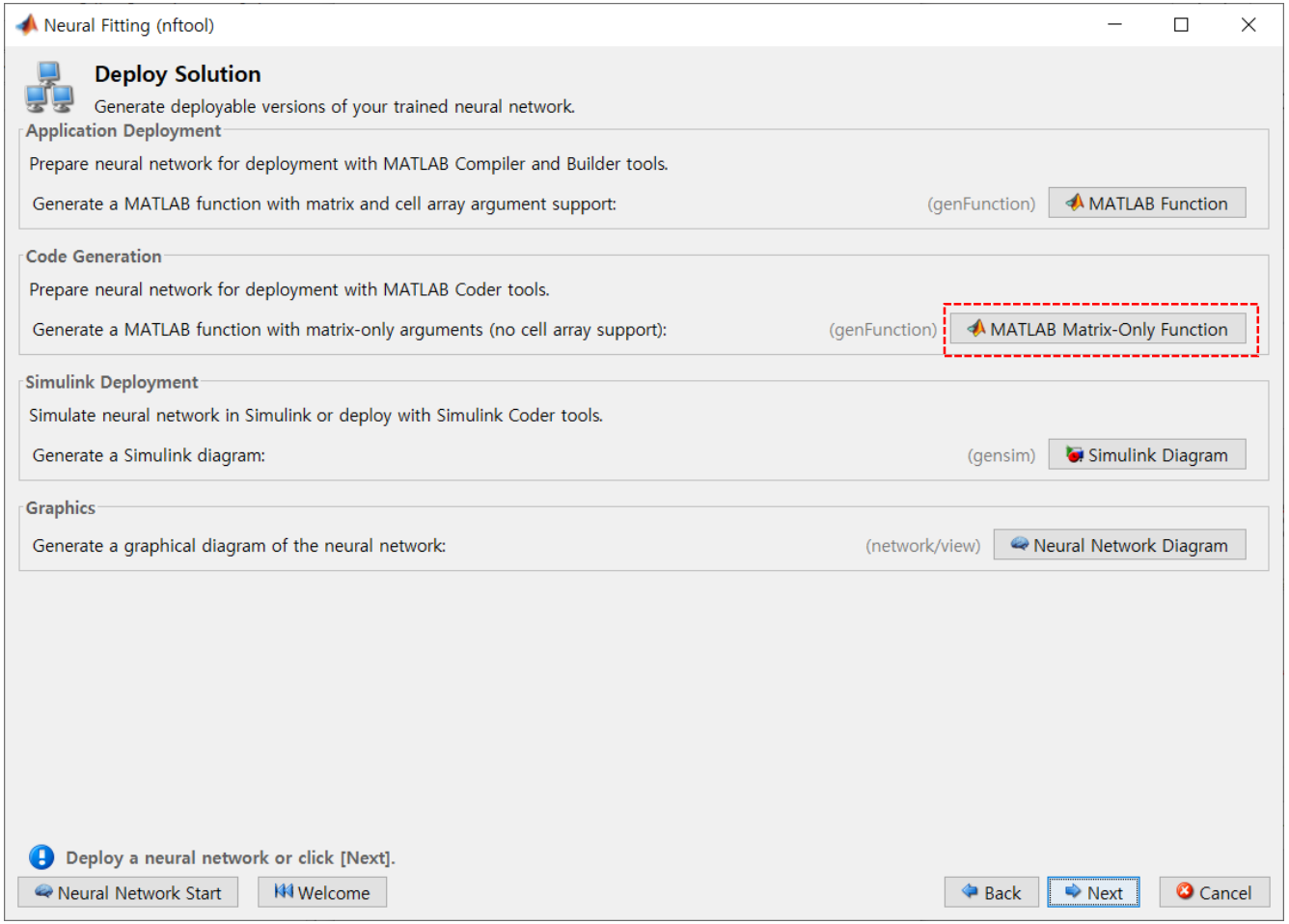
11. Train을 눌러서 학습
12. 결정계수 확인





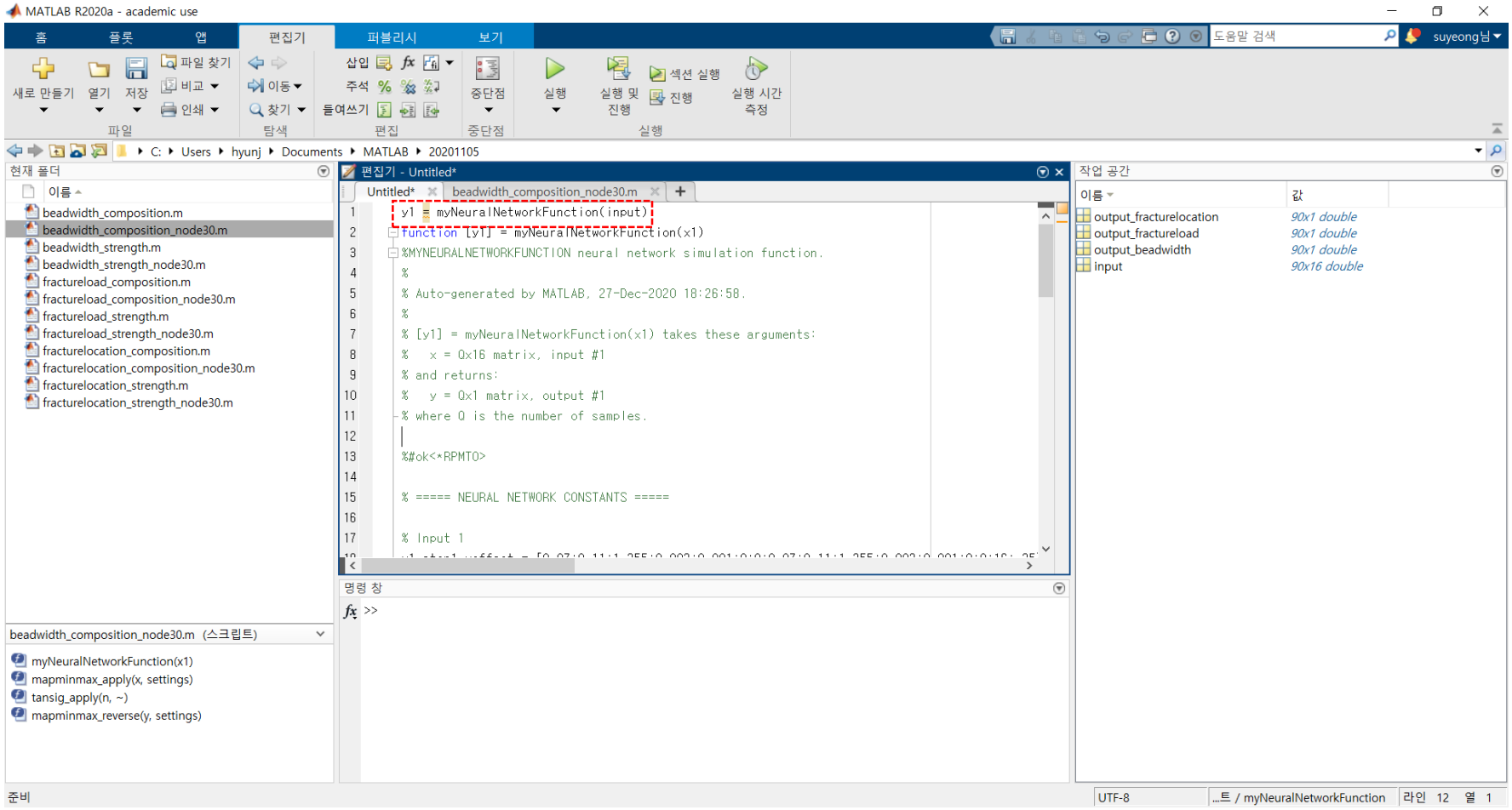
2. MATLAB에서 신경회로망 fitting (SNN)

11. Next를 두 번 클릭하면 아래와 같은 화면이 나옴
12. MATLAB Matrix-Only Function 선택



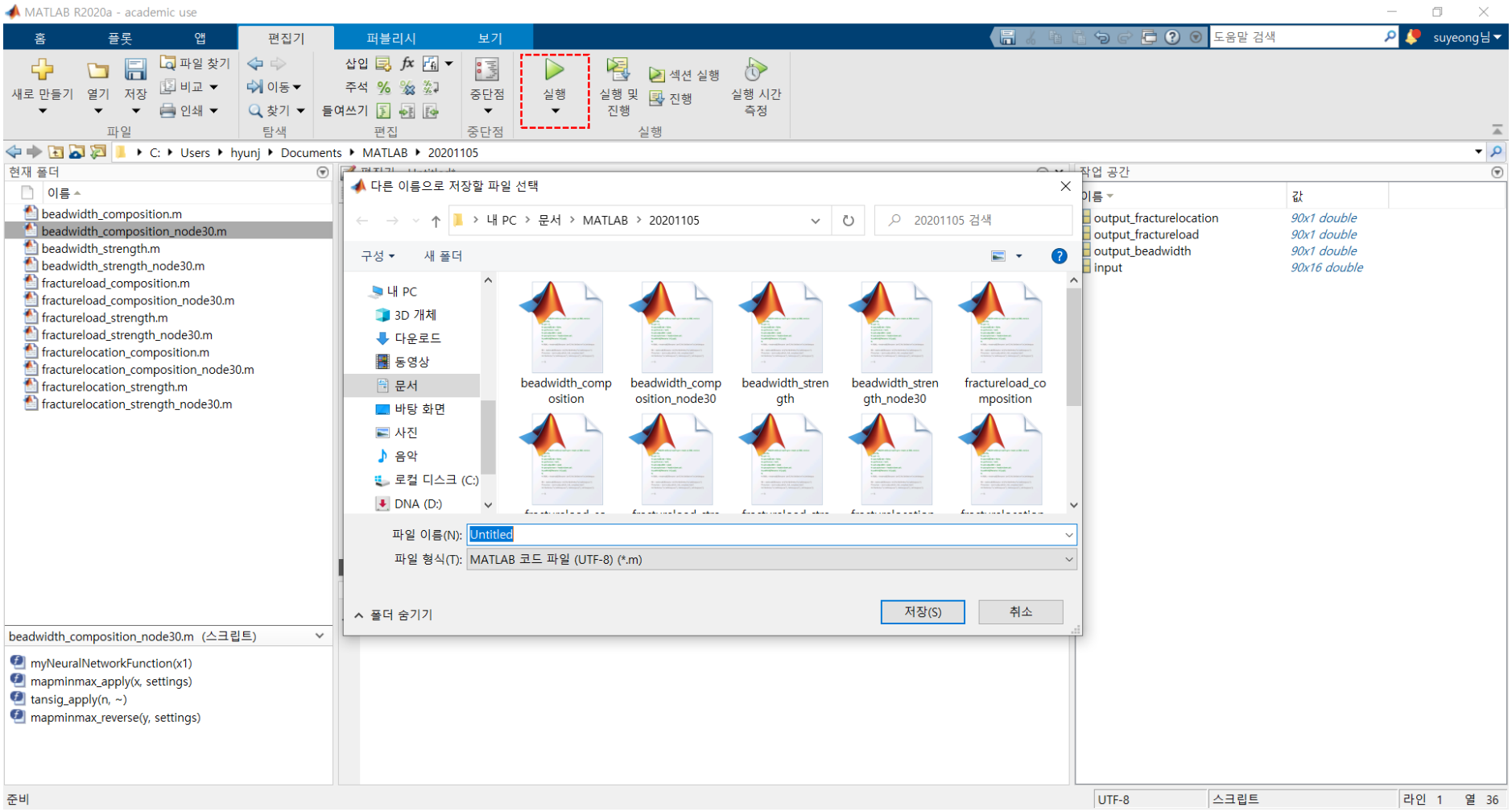
2. MATLAB에서 신경회로망 fitting (SNN)

13. 아래와 같은 편집기가 생성되며, 그 수식 위에 y1=myNeuralNetworkFunction(x1)을 작성해줘야 하며 괄호 안에 x1 대신 작업 공간에서 확인할 수 있는 실제 사용된 input data의 이름을 작성



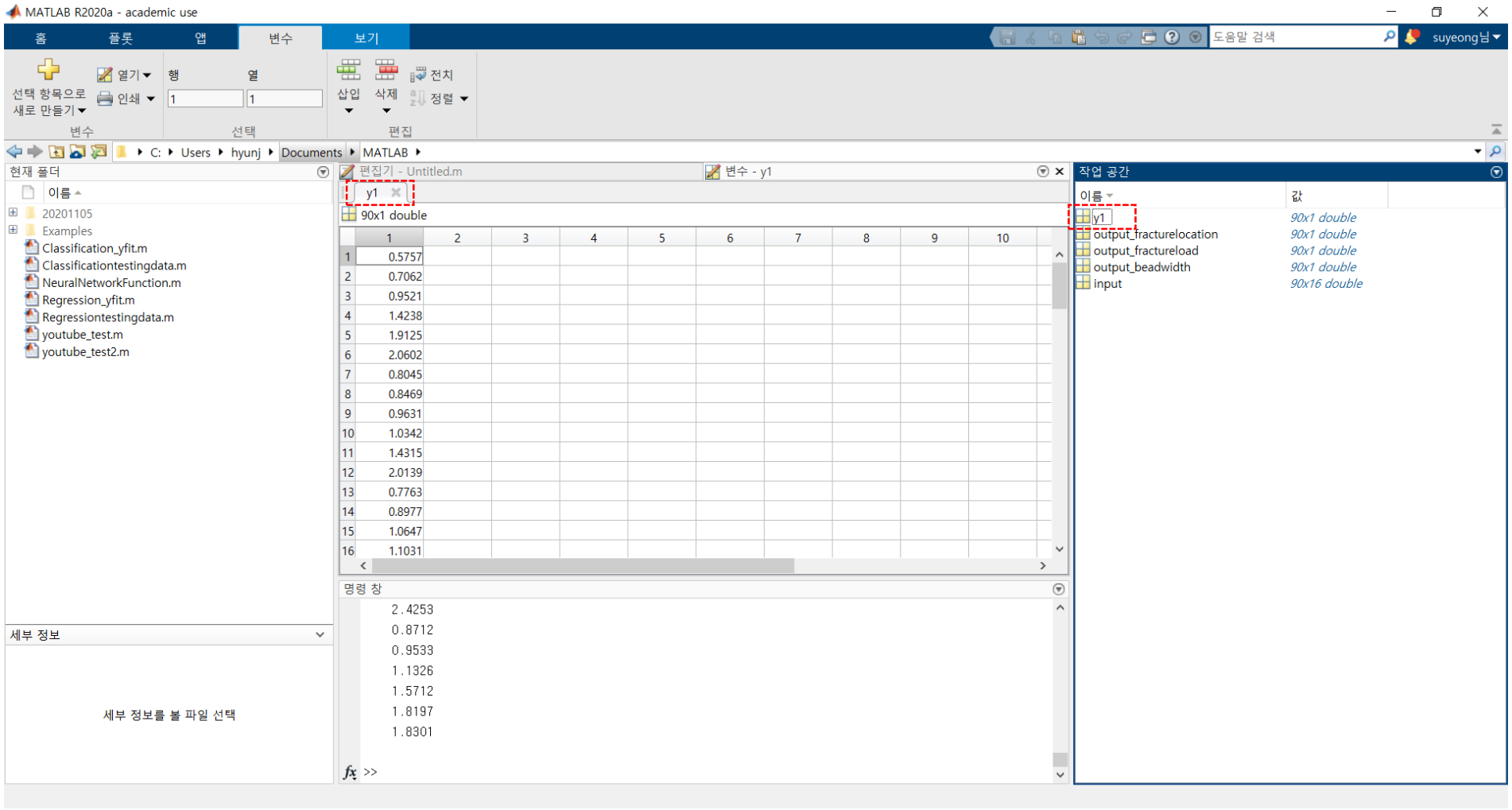
2. MATLAB에서 신경회로망 fitting (SNN)

14. 실행을 클릭하면 다음과 같이 코드 파일을 저장하라는 화면이 생성



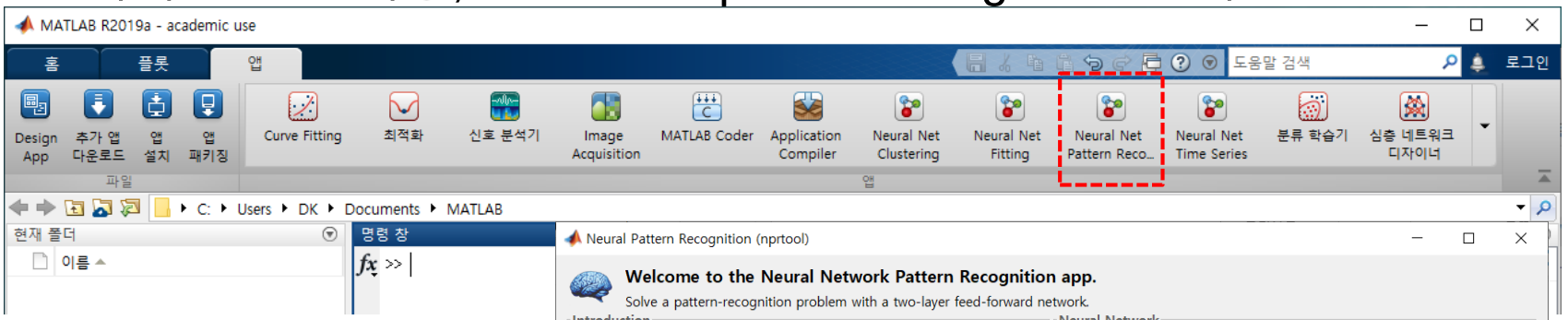
2. MATLAB에서 신경회로망 fitting (SNN)

15. y1이라는 SNN 모델을 통해 예측된 값이 생성됨

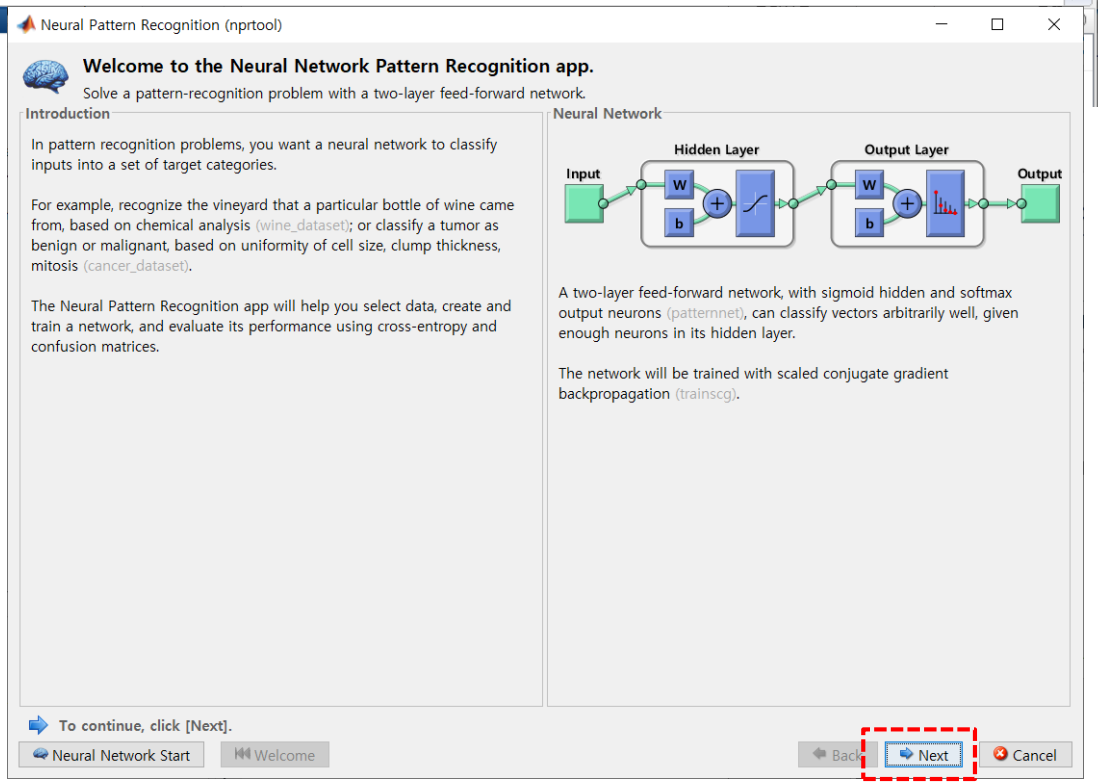


3. MATLAB에서 신경회로망 분류

1. Matlab 실행
2. Tab에서 앱으로 이동, Neural net pattern recognition 선택

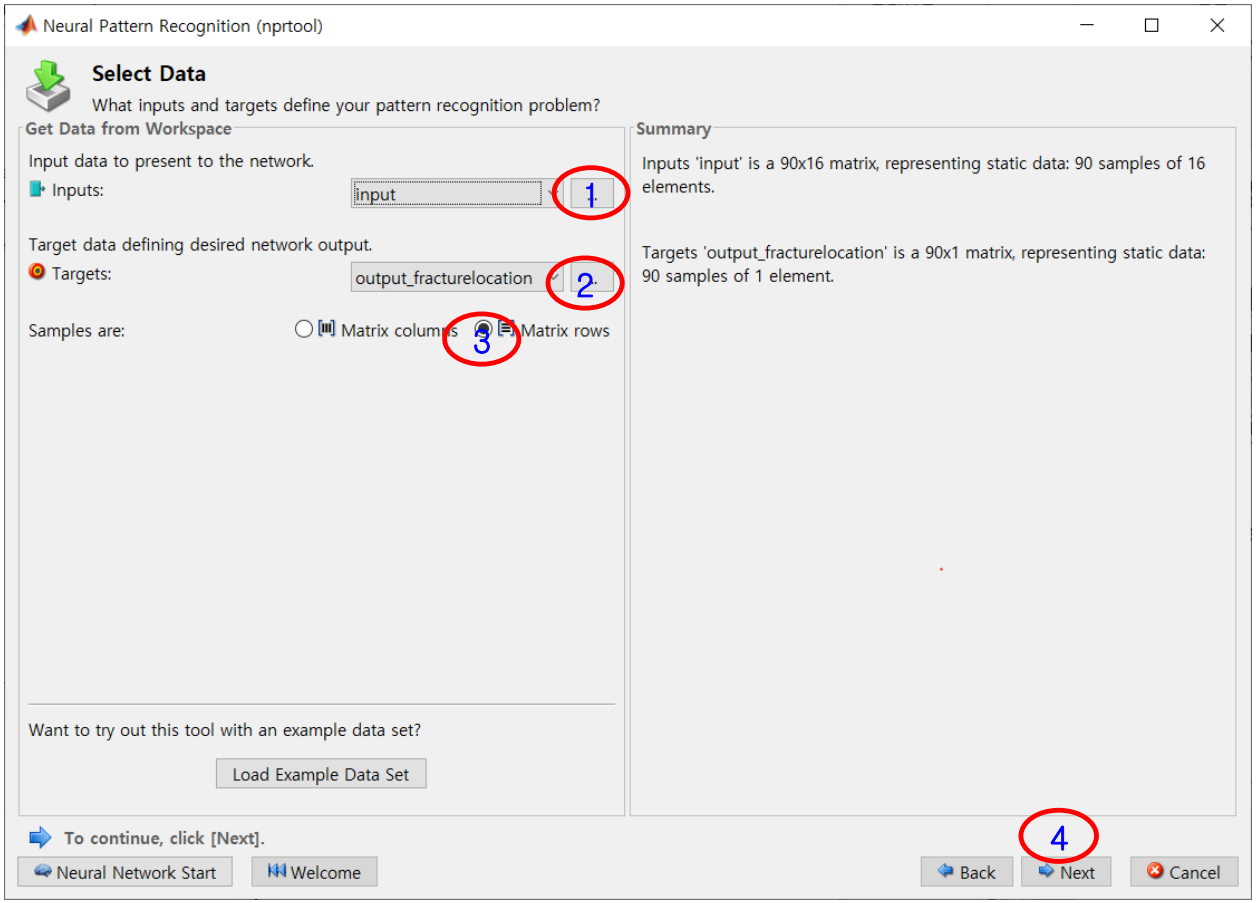


3. 새로 뜬 창에서 Next 클릭



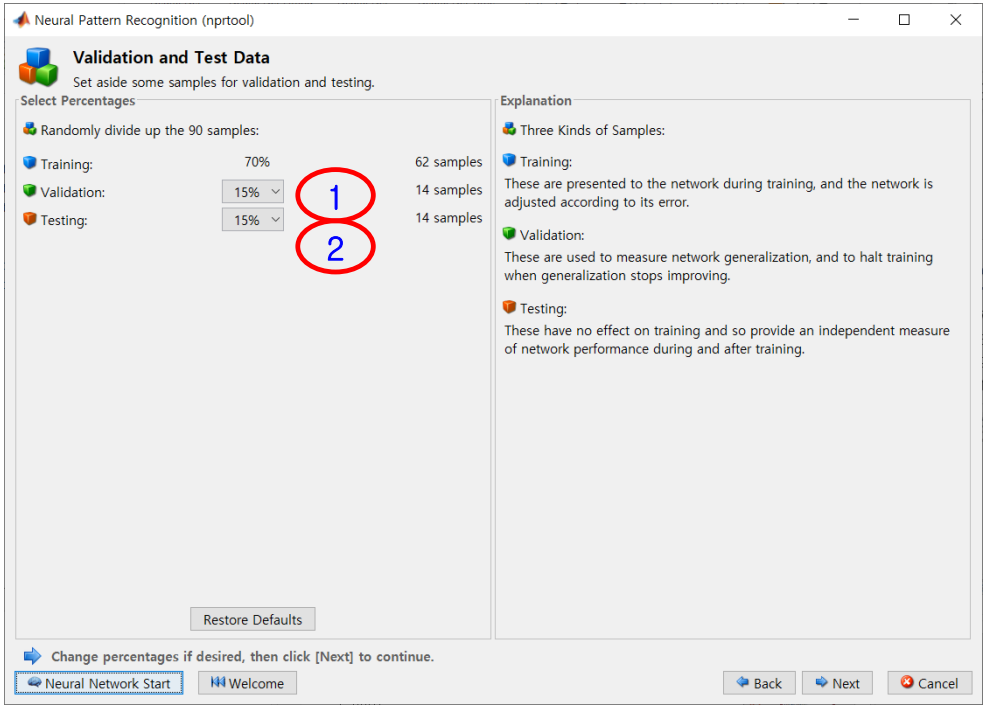
3. MATLAB에서 신경회로망 분류

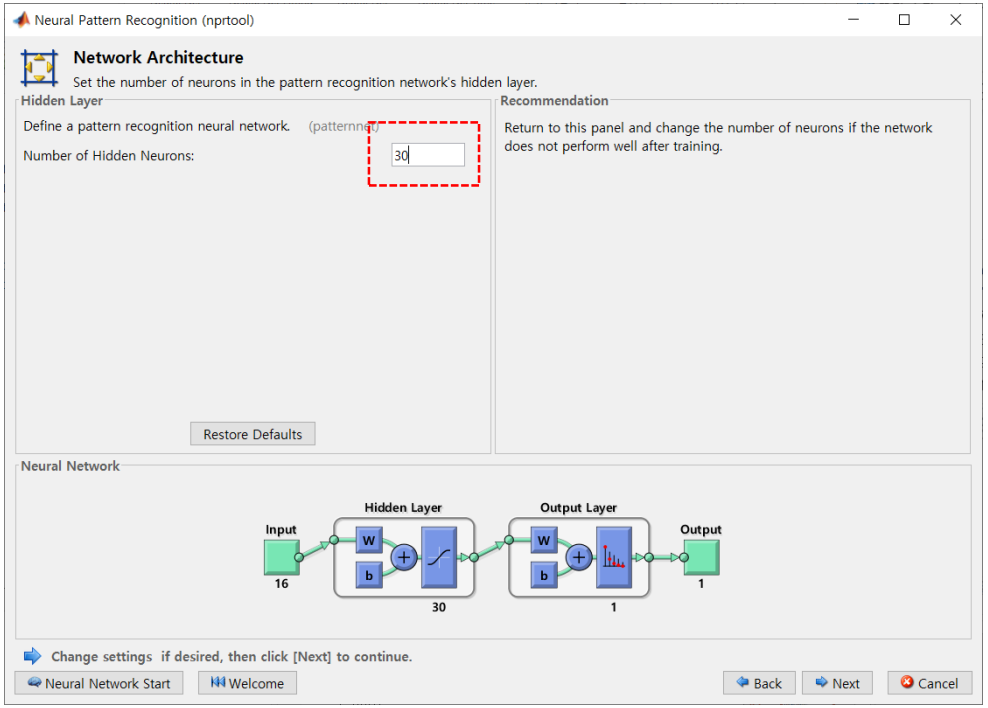
4. 1을 눌러서 input 데이터만 있는 파일을 불러옴
5. 2를 눌러서 output 데이터만 있는 파일을 불러옴
6. 3에서 Matrix rows선택하고 4-Next 선택



3. MATLAB에서 신경회로망 분류

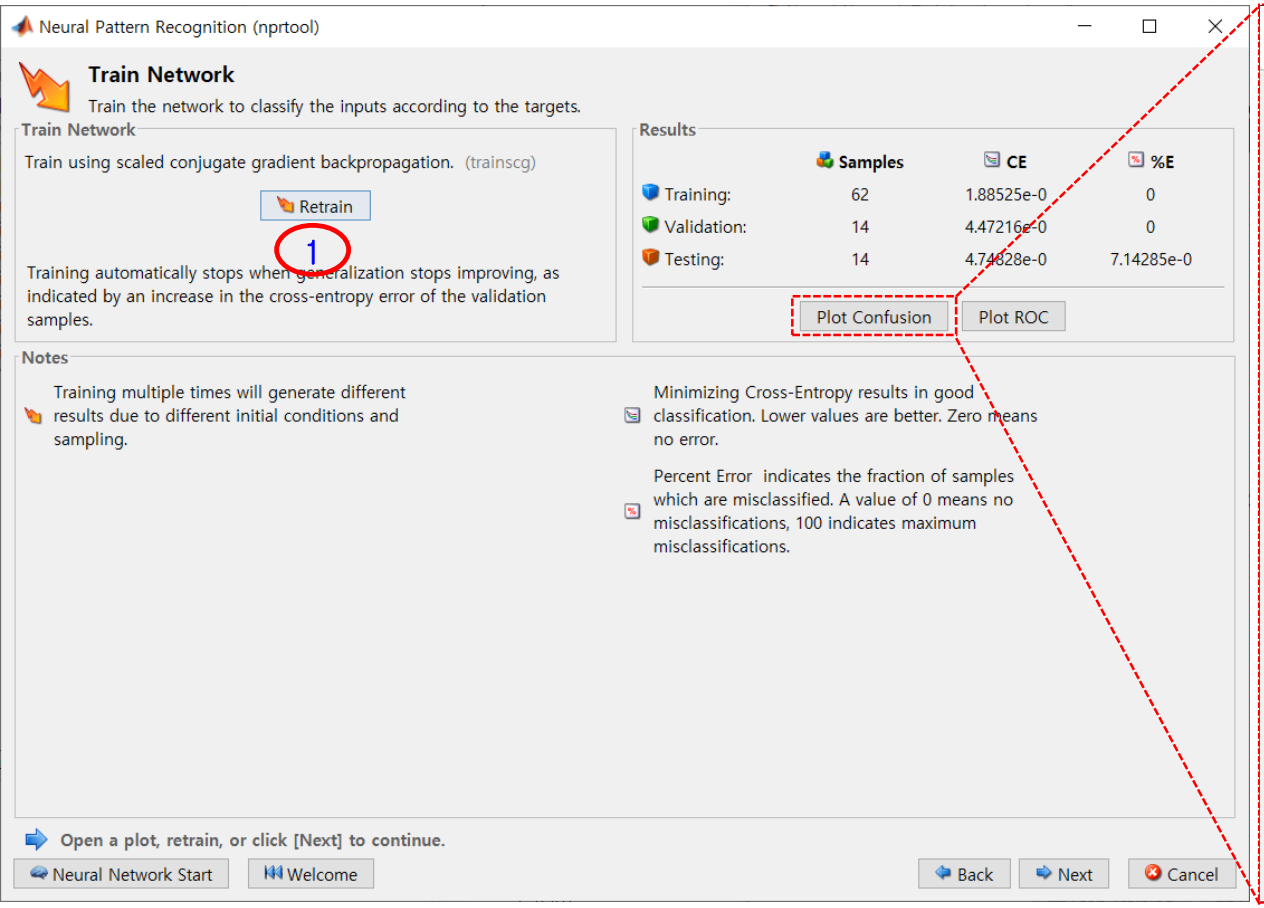
7. Validation과 testing data의 퍼센트 변경 가능
8. Next 로 넘어가면
9. Hidden layer의 노드 개수 변경 가능
10. Next로 넘어갈 것

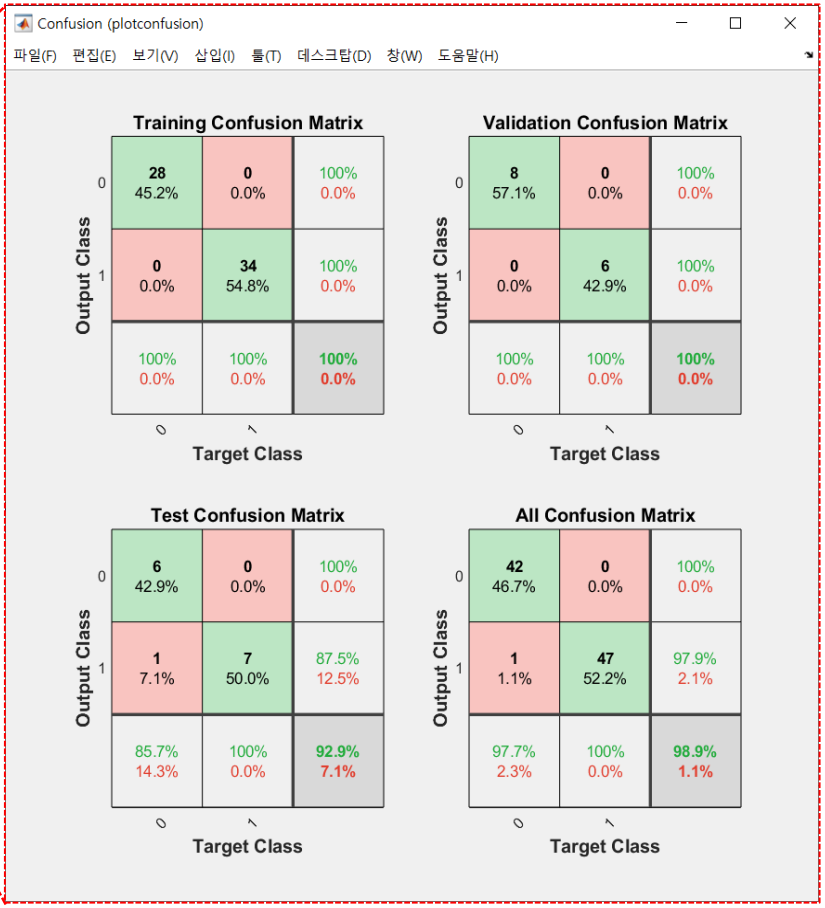




3. MATLAB에서 신경회로망 분류

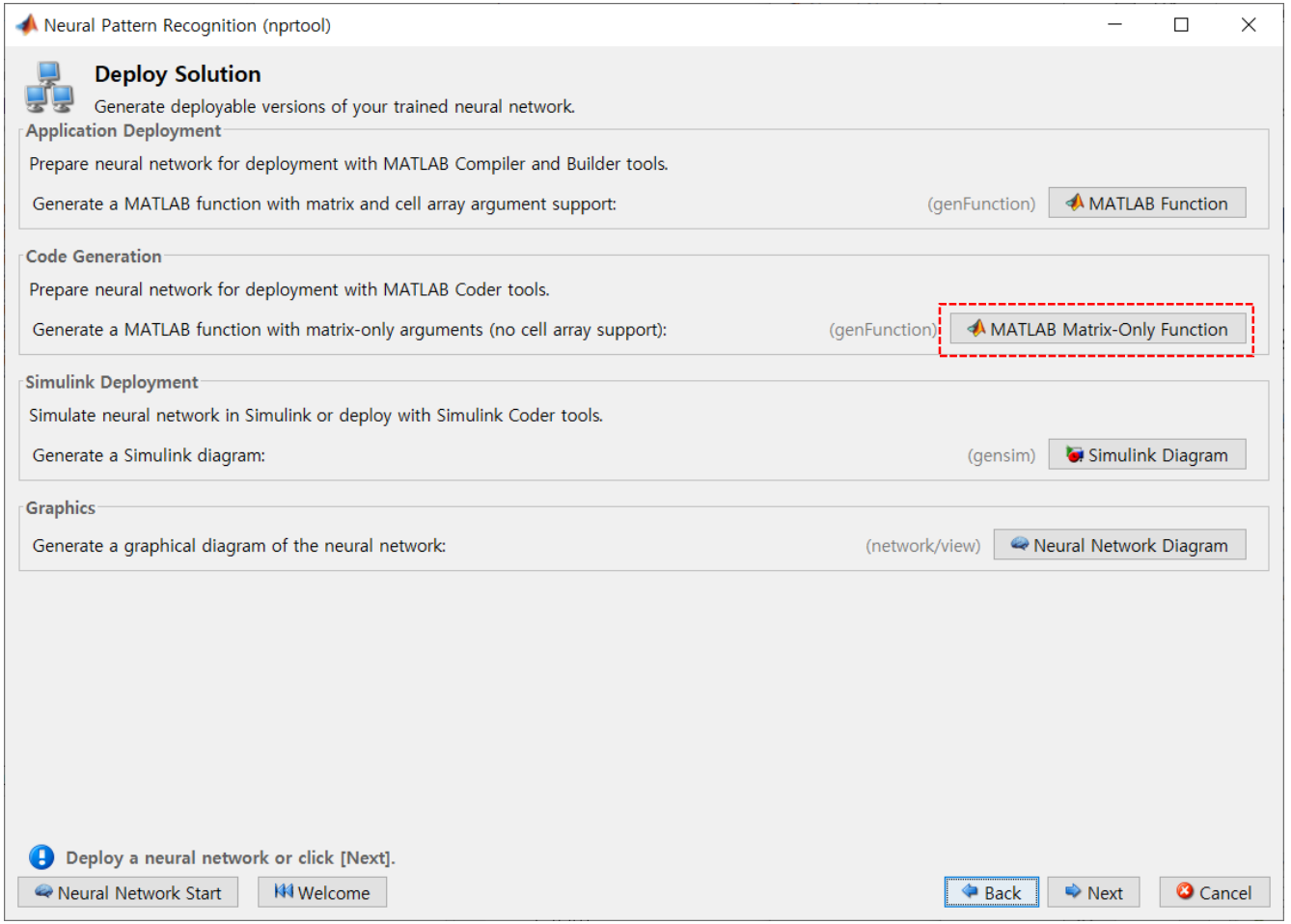
11. Train을 눌러서 학습
12. 오차율 확인





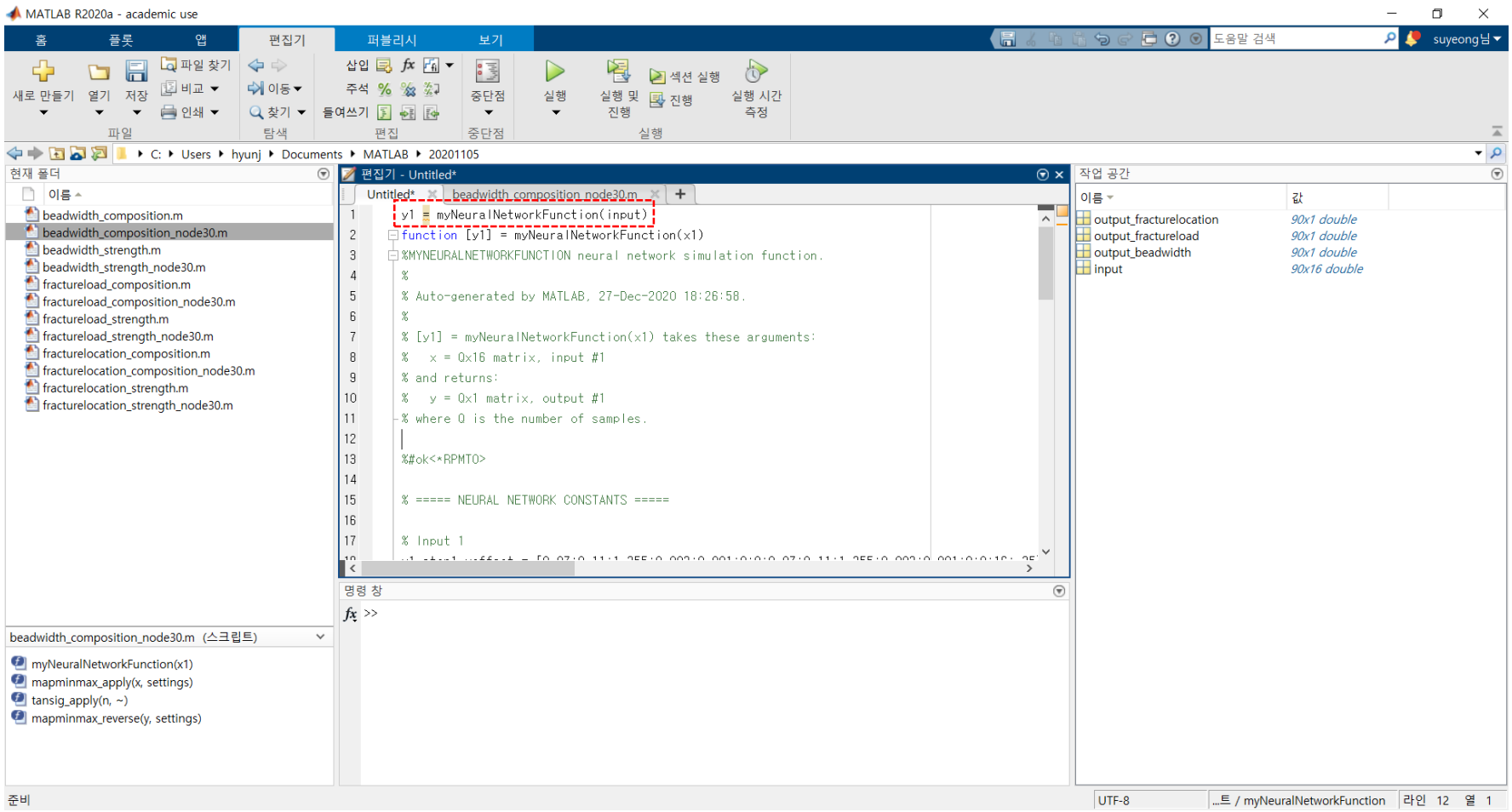
3. MATLAB에서 신경회로망 분류

11. Next를 두 번 클릭하면 아래와 같은 화면이 나옴
12. MATLAB Matrix-Only Function 선택



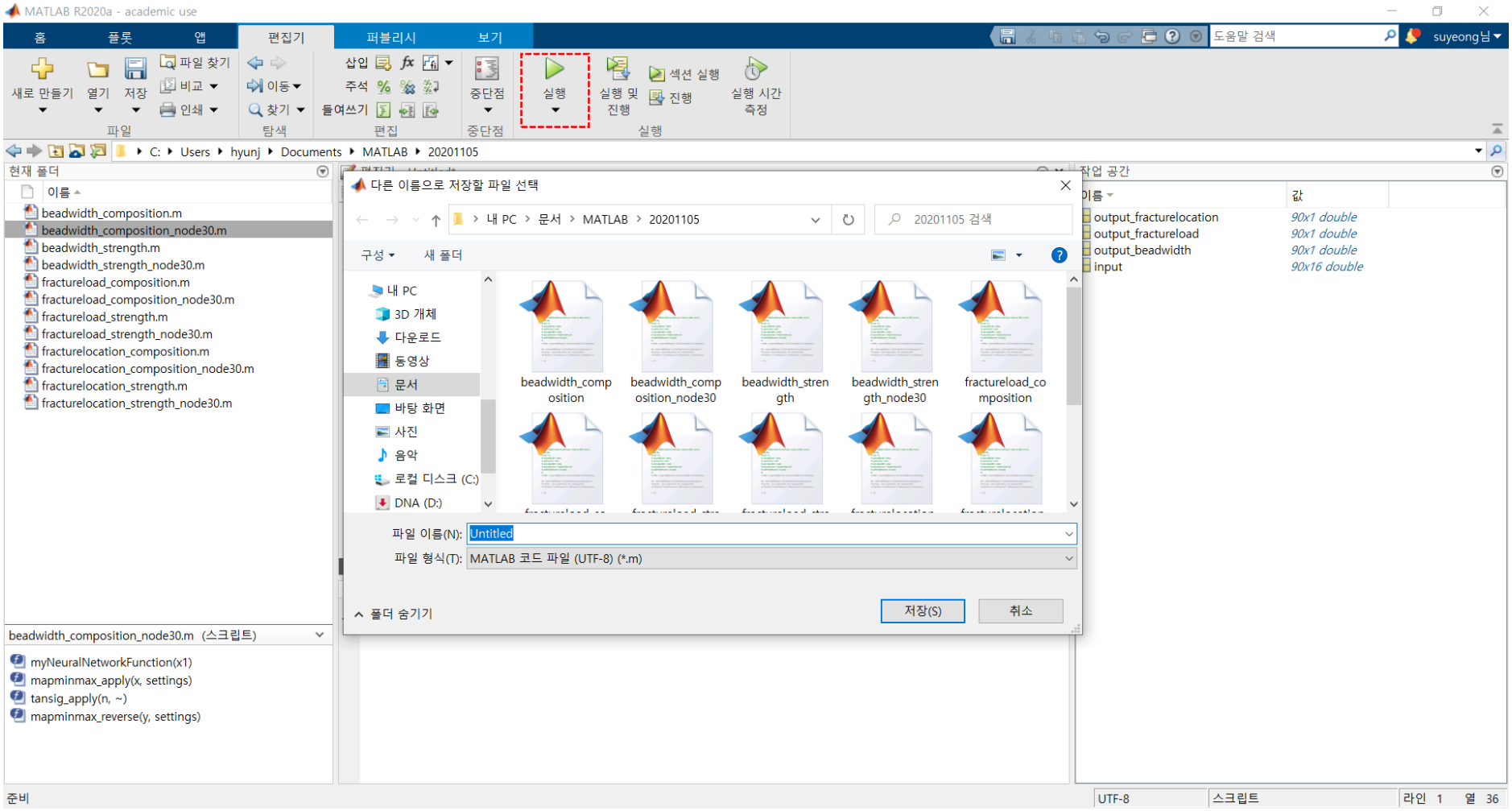
3. MATLAB에서 신경회로망 분류

13. 아래와 같은 편집기가 생성되며, 그 수식 위에 y1=myNeuralNetworkFunction(x1)을 작성해줘야 하며 괄호 안에 x1 대신 작업 공간에서 확인할 수 있는 실제 사용된 input data의 이름을 작성



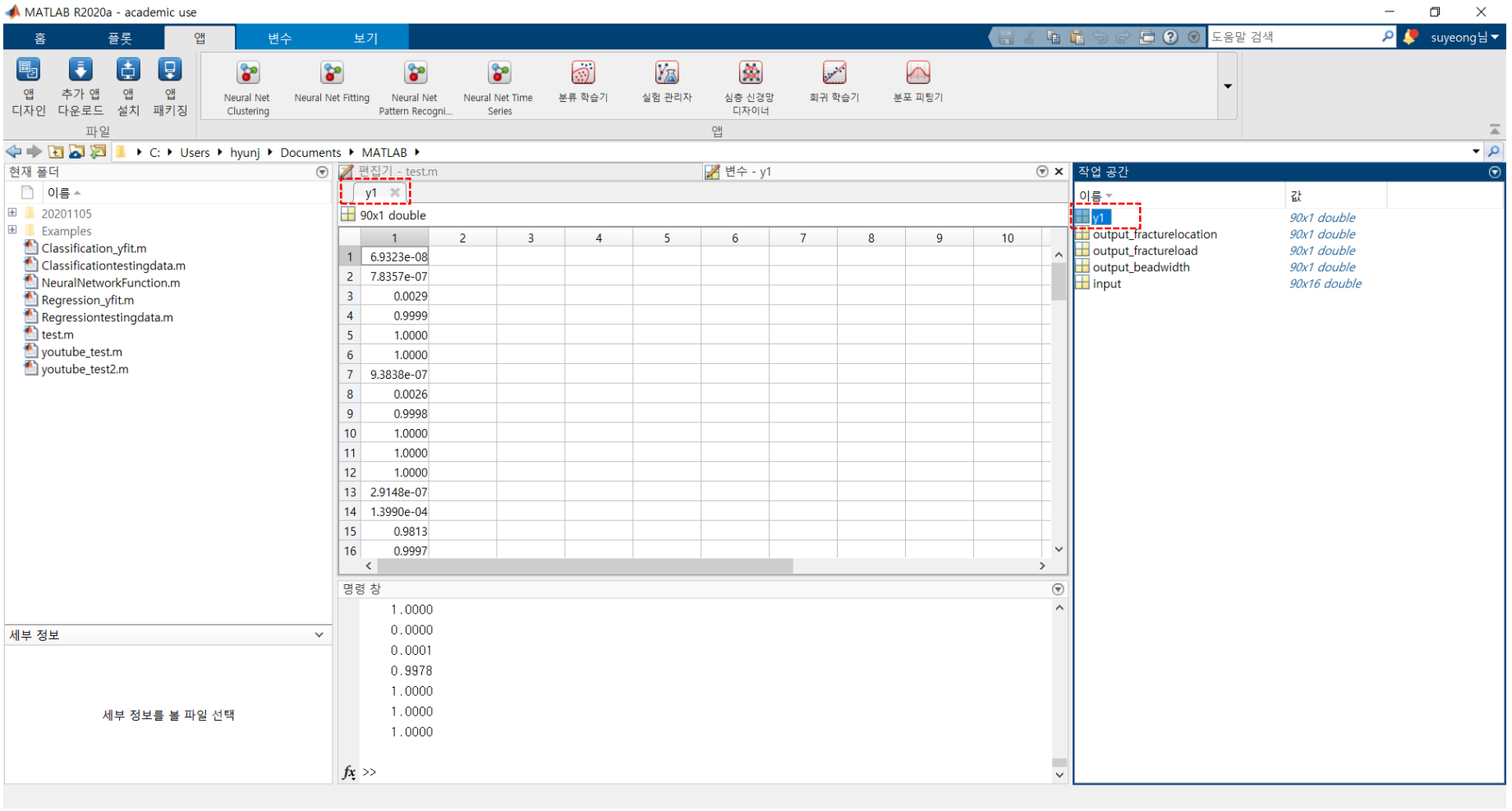
3. MATLAB에서 신경회로망 분류

14. 실행을 클릭하면 다음과 같이 코드 파일을 저장하라는 화면이 생성



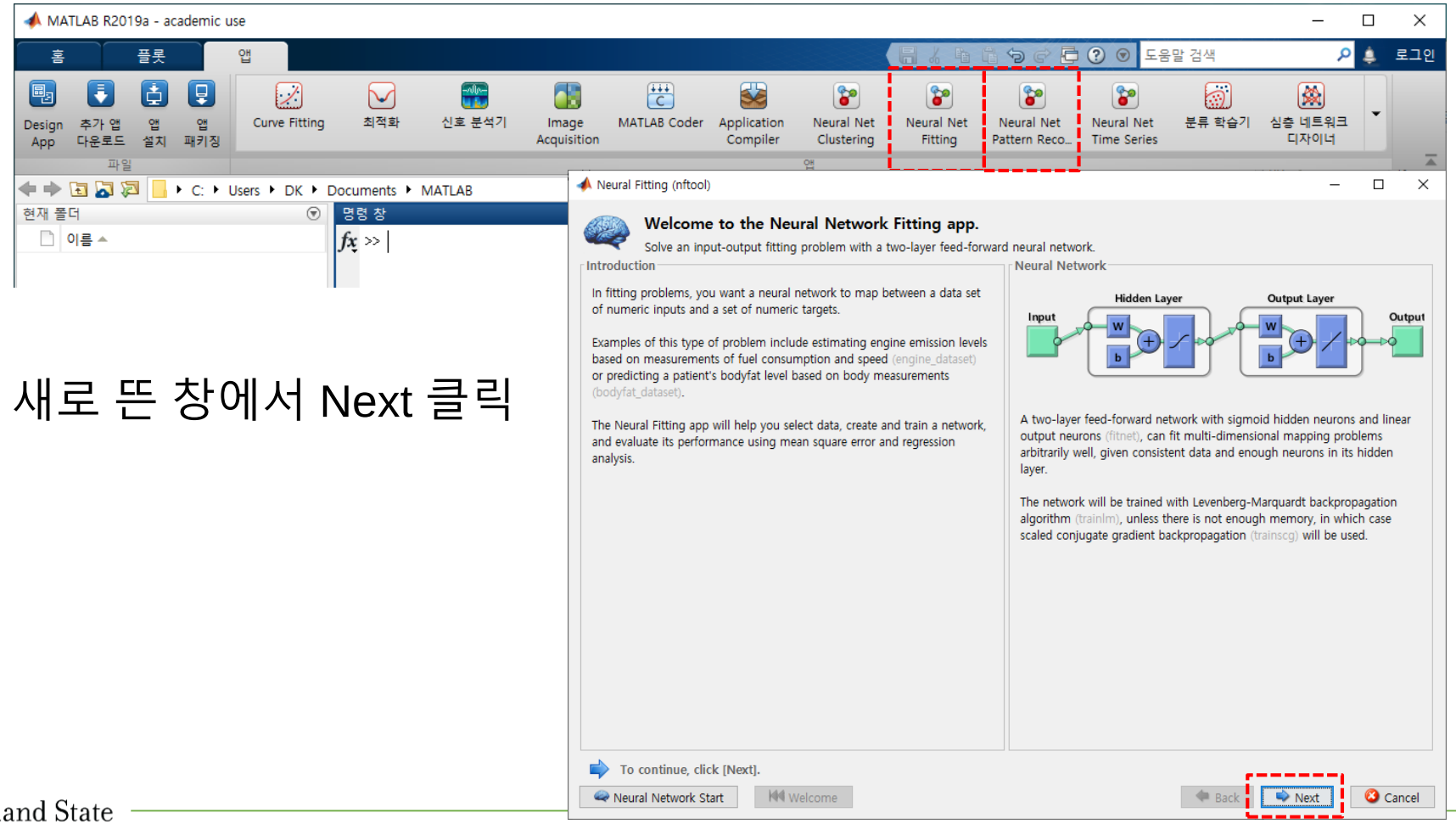
3. MATLAB에서 신경회로망 분류

15. y1이라는 SNN 모델을 통해 예측된 값이 생성됨



4. MATLAB에서 신경회로망 fitting (DNN)

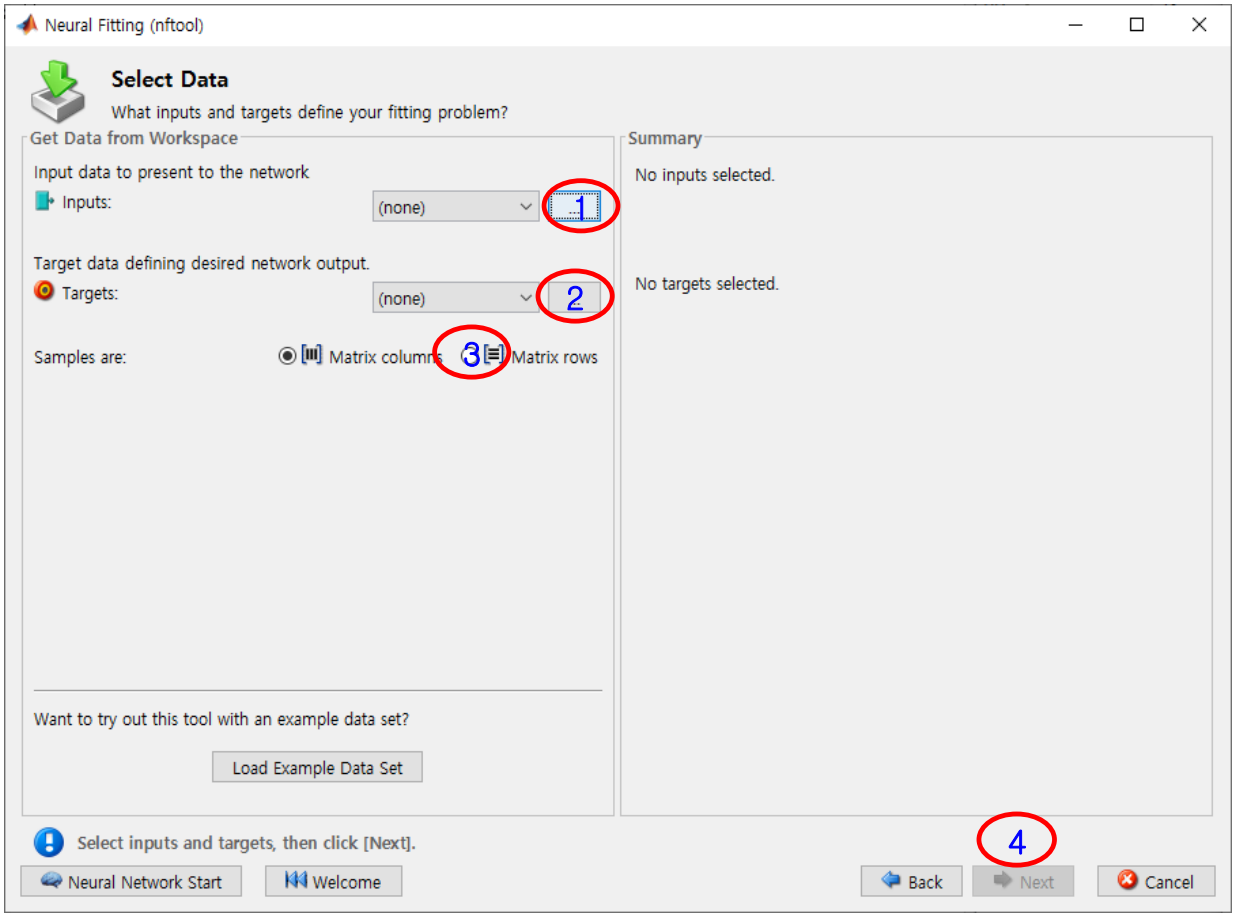
1. Matlab 실행
2. Tab에서 앱으로 이동, 회귀 문제의 경우 Neural net fitting을 분류 문제의 경우 Neural net pattern recognition 선택



3. 새로 뜬 창에서 Next 클릭

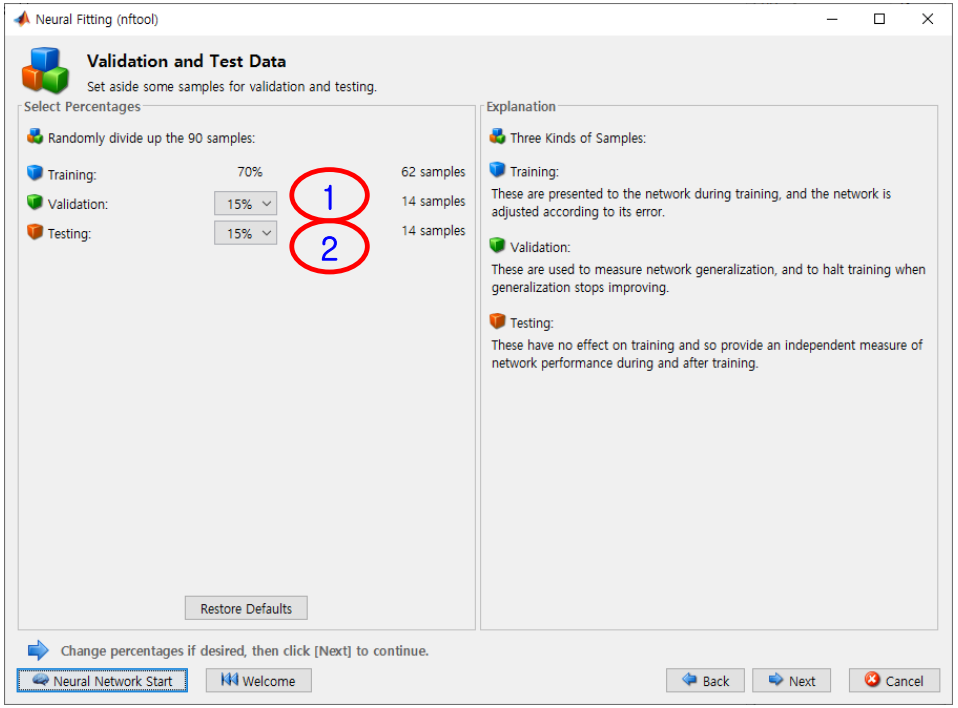
4. MATLAB에서 신경회로망 fitting (DNN)

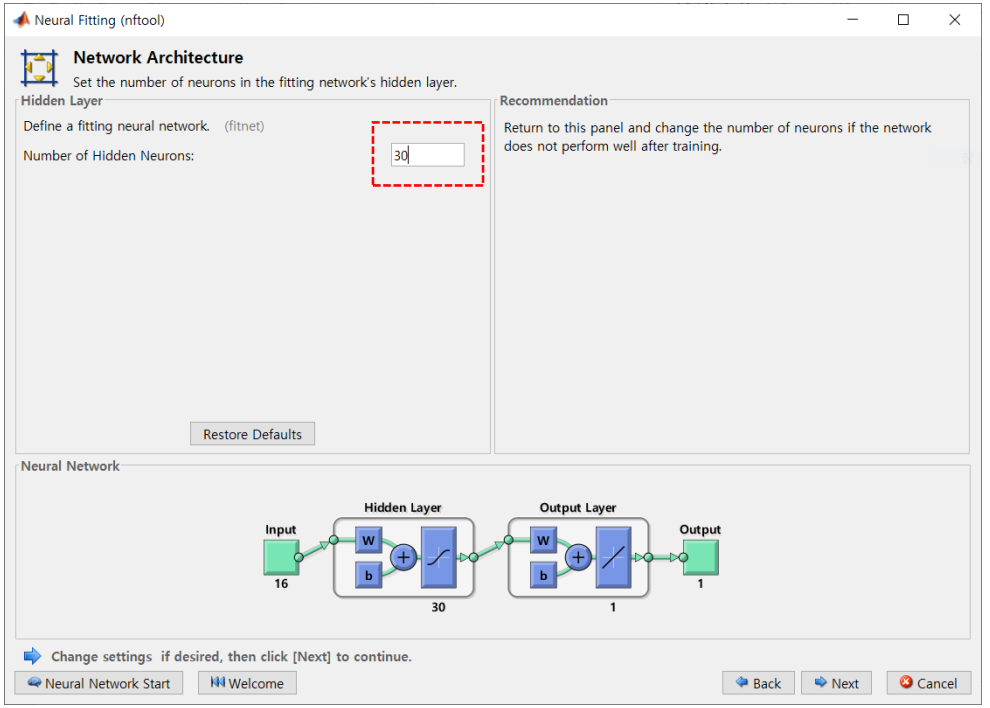
4. 1을 눌러서 input 데이터만 있는 파일을 불러옴
5. 2를 눌러서 output 데이터만 있는 파일을 불러옴
6. 3에서 Matrix rows선택하고 4-Next 선택



4. MATLAB에서 신경회로망 fitting (DNN)

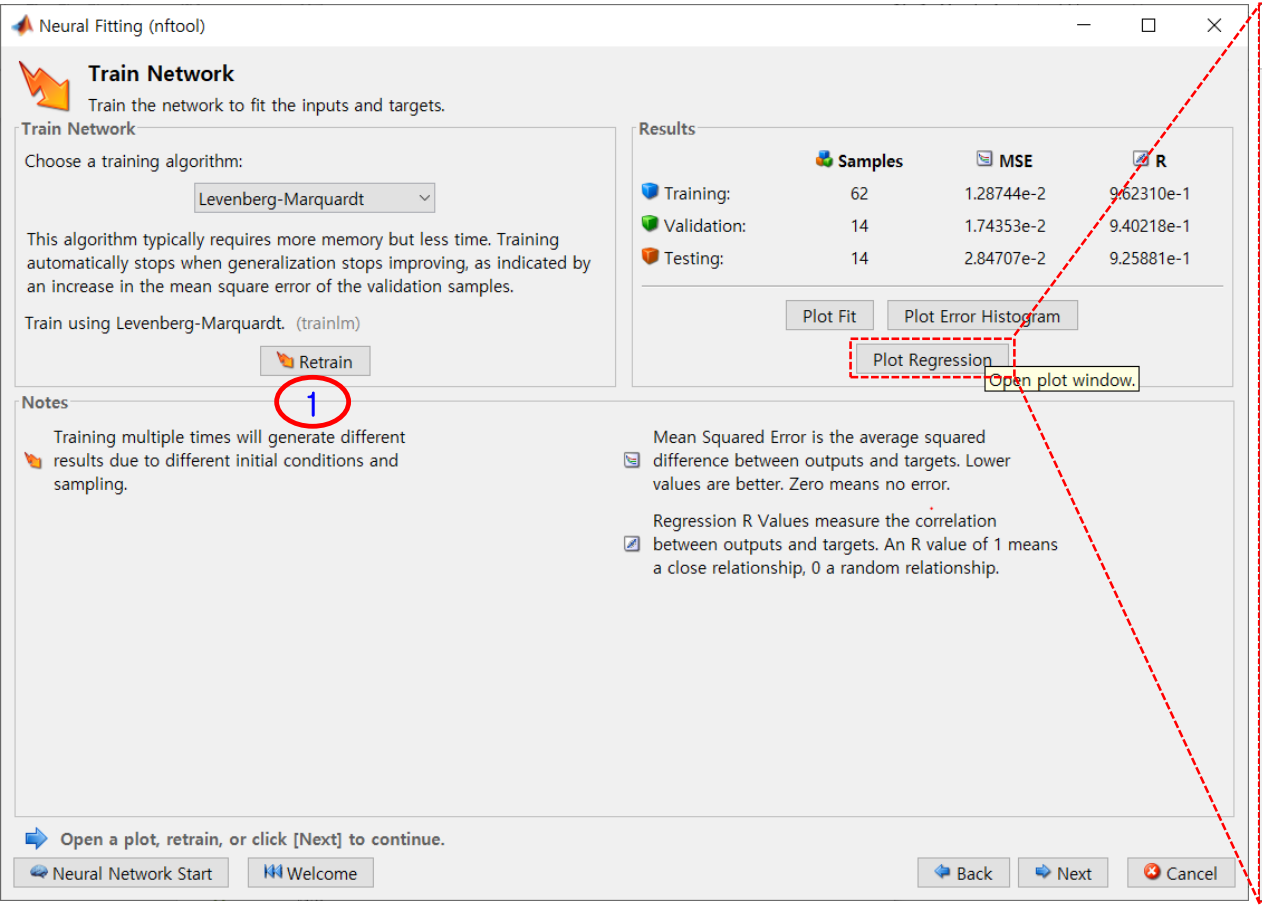
7. Validation과 testing data의 퍼센트 변경 가능
8. Next 로 넘어가면
9. Hidden layer의 노드 개수 변경 가능
10. Next로 넘어갈 것

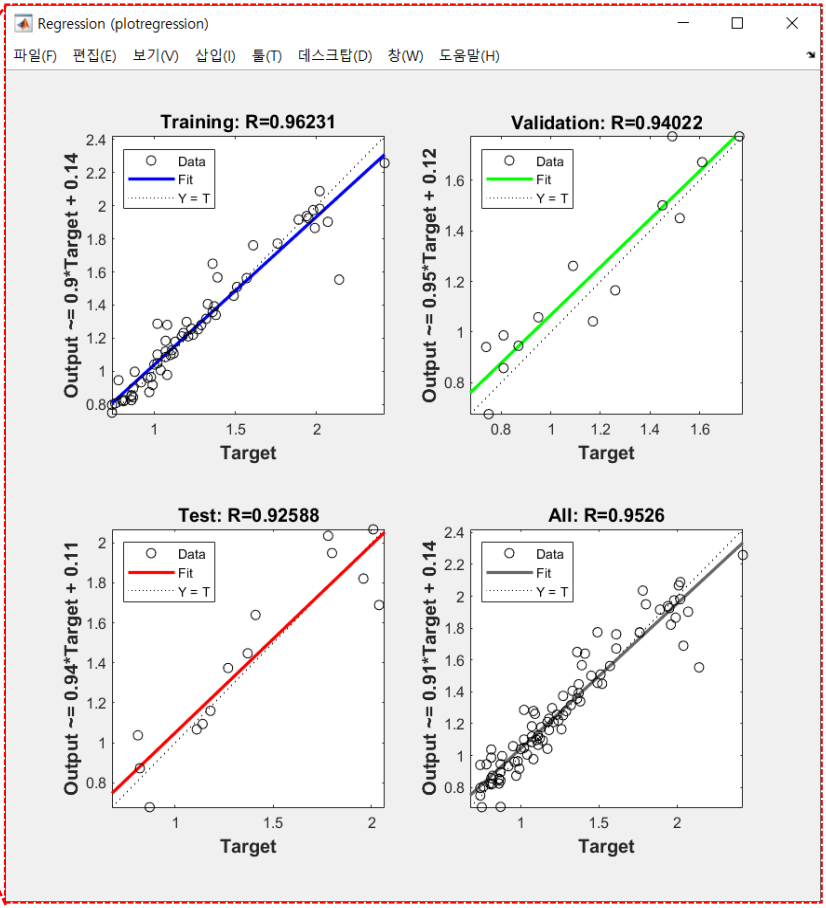




4. MATLAB에서 신경회로망 fitting (DNN)

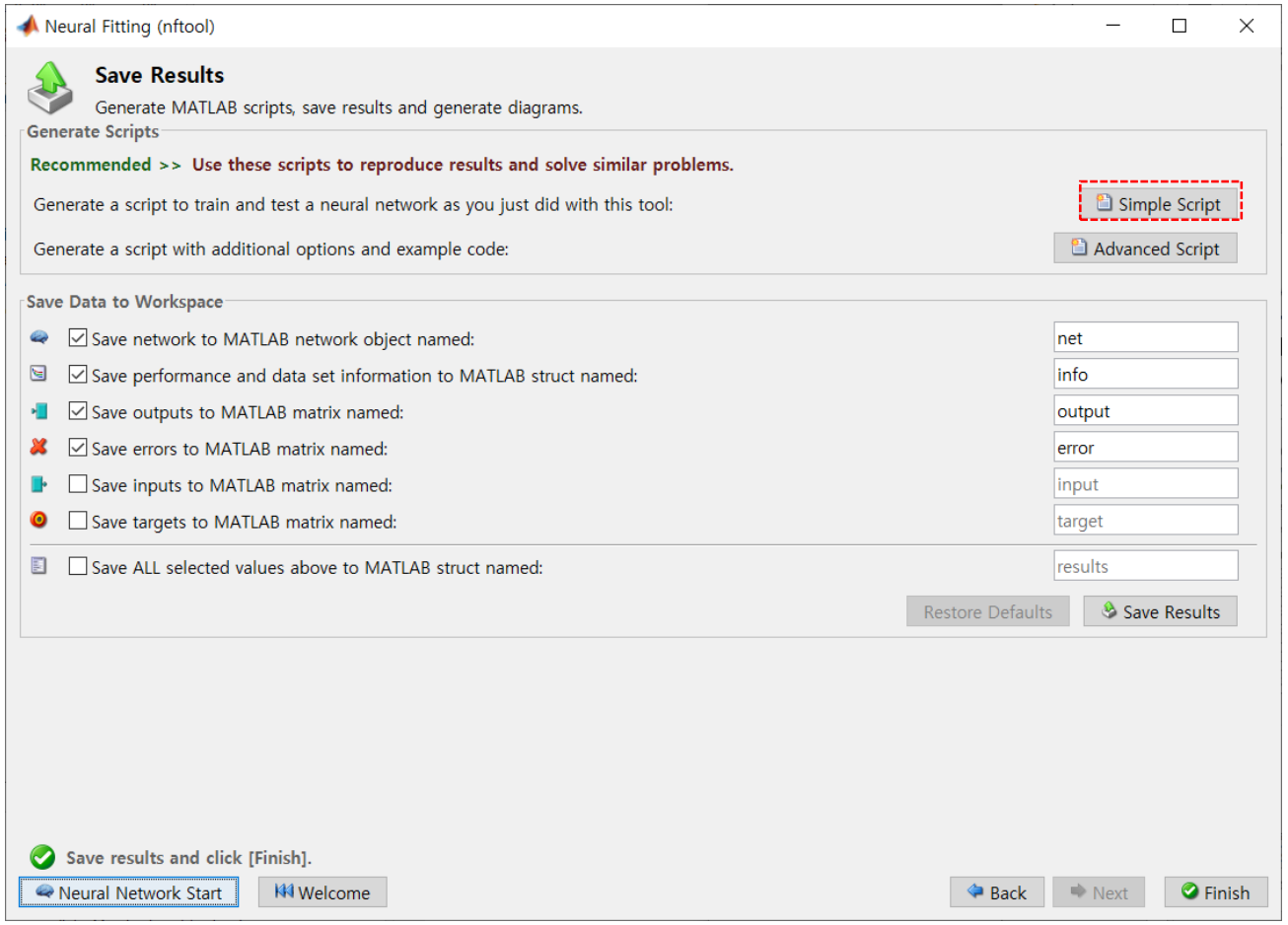
11. Train을 눌러서 학습
12. 결정계수 확인





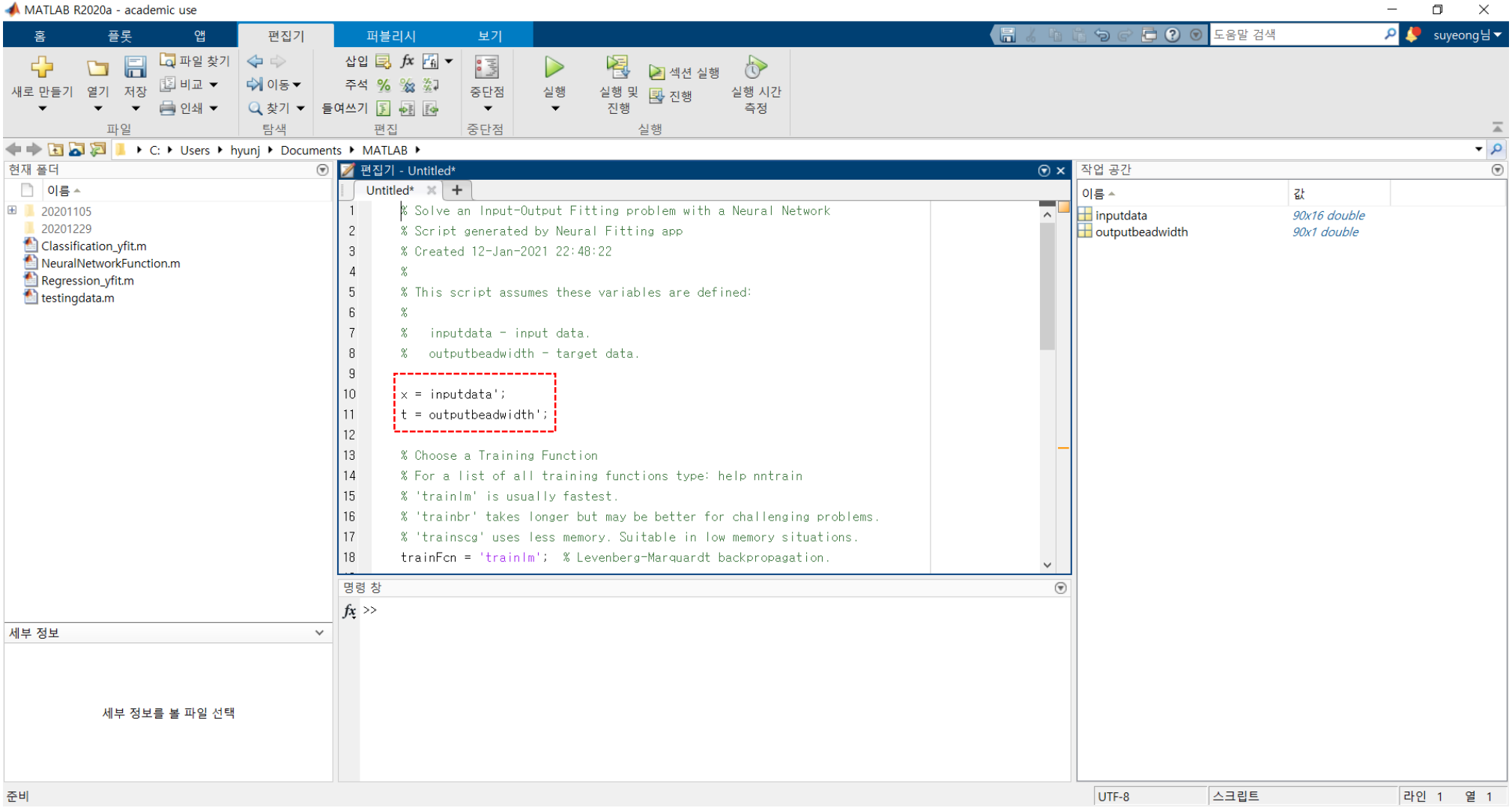
4. MATLAB에서 신경회로망 fitting (DNN)

13. Next를 **세 번** 클릭하면 아래와 같은 화면이 나옴
14. Simple script 선택



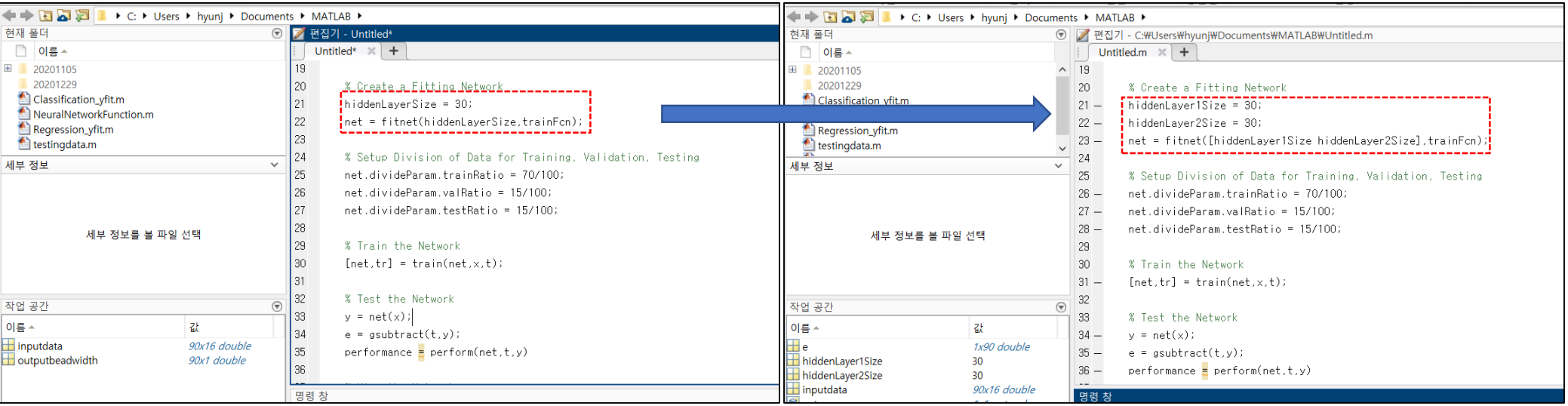
4. MATLAB에서 신경회로망 fitting (DNN)

15. 아래와 같은 편집기가 생성됨



4. MATLAB에서 신경회로망 fitting (DNN)

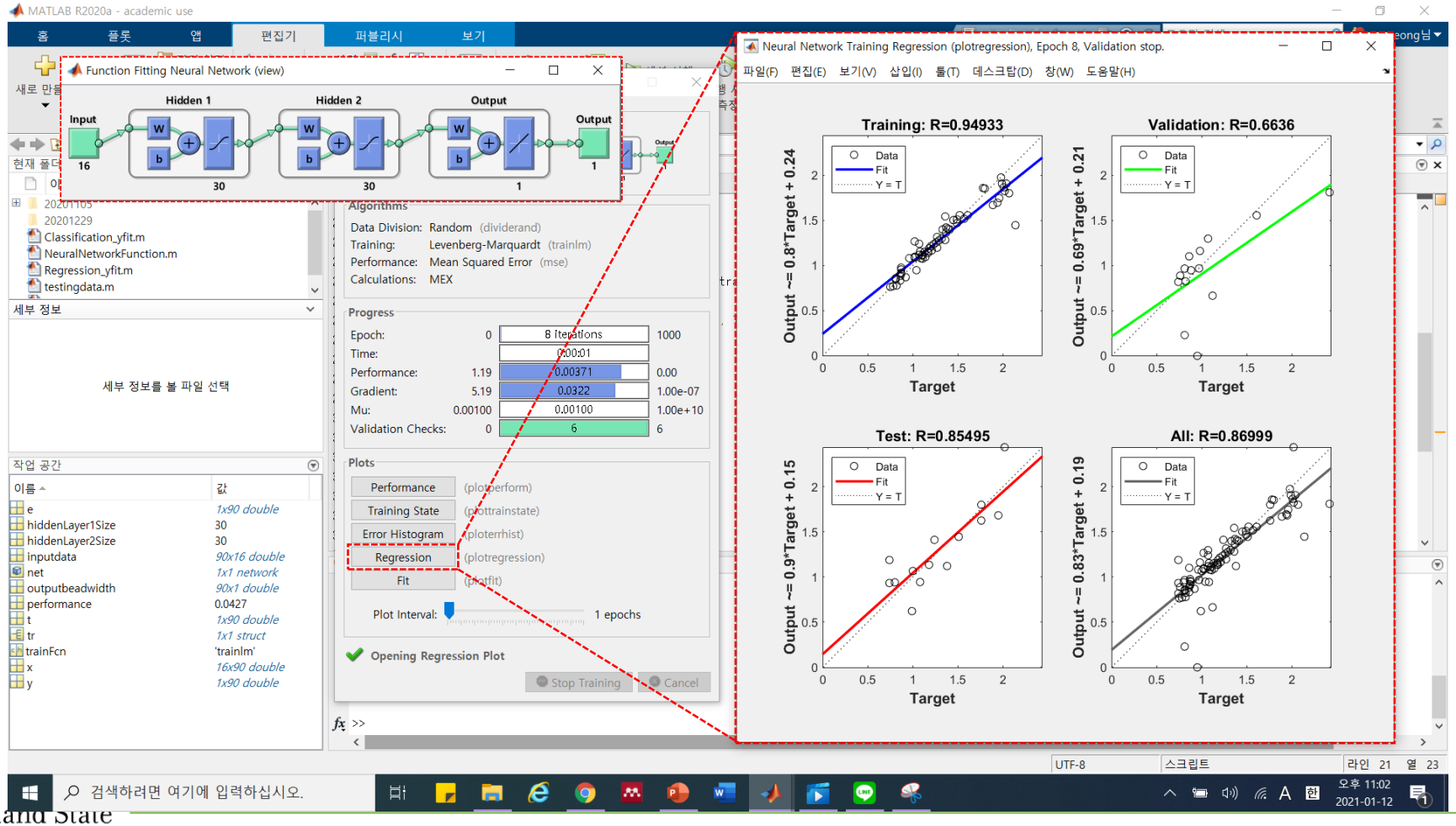
16. Hidden Layer의 개수와 Training, Validation, Test 데이터의 비율을 확인할 수 있음
17. 여기서 hidden layer의 층을 하나 이상을 추가하게 되면 DNN 모델을 구축할 수 있음



18. 상단에 실행 버튼 클릭

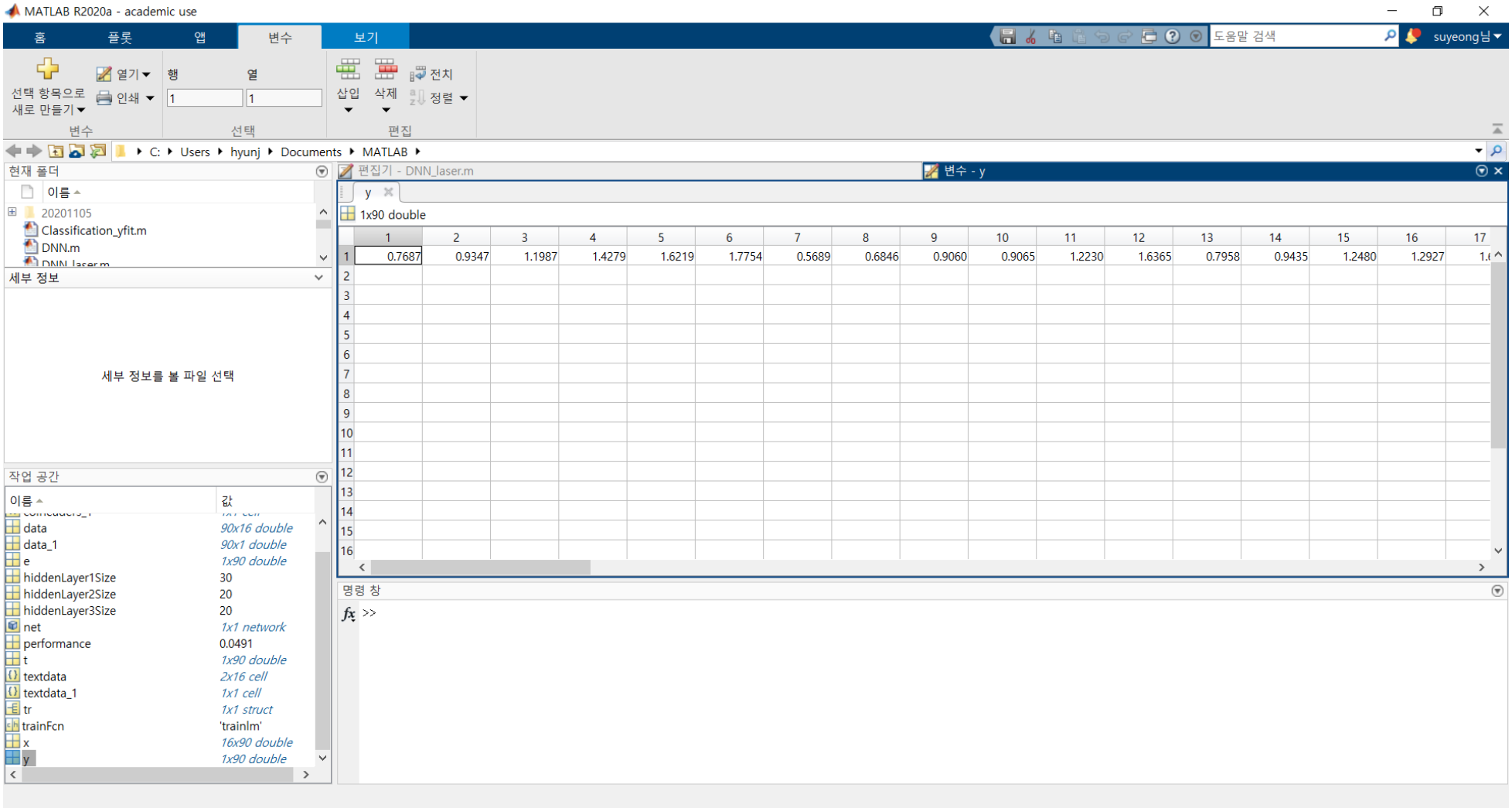
4. MATLAB에서 신경회로망 fitting (DNN)

19. 사용한 모델의 구조를 보여주는 창이 뜸
20. Regression을 클릭하면 구축된 모델의 training, validation, test의 결정계수를 확인할 수 있음



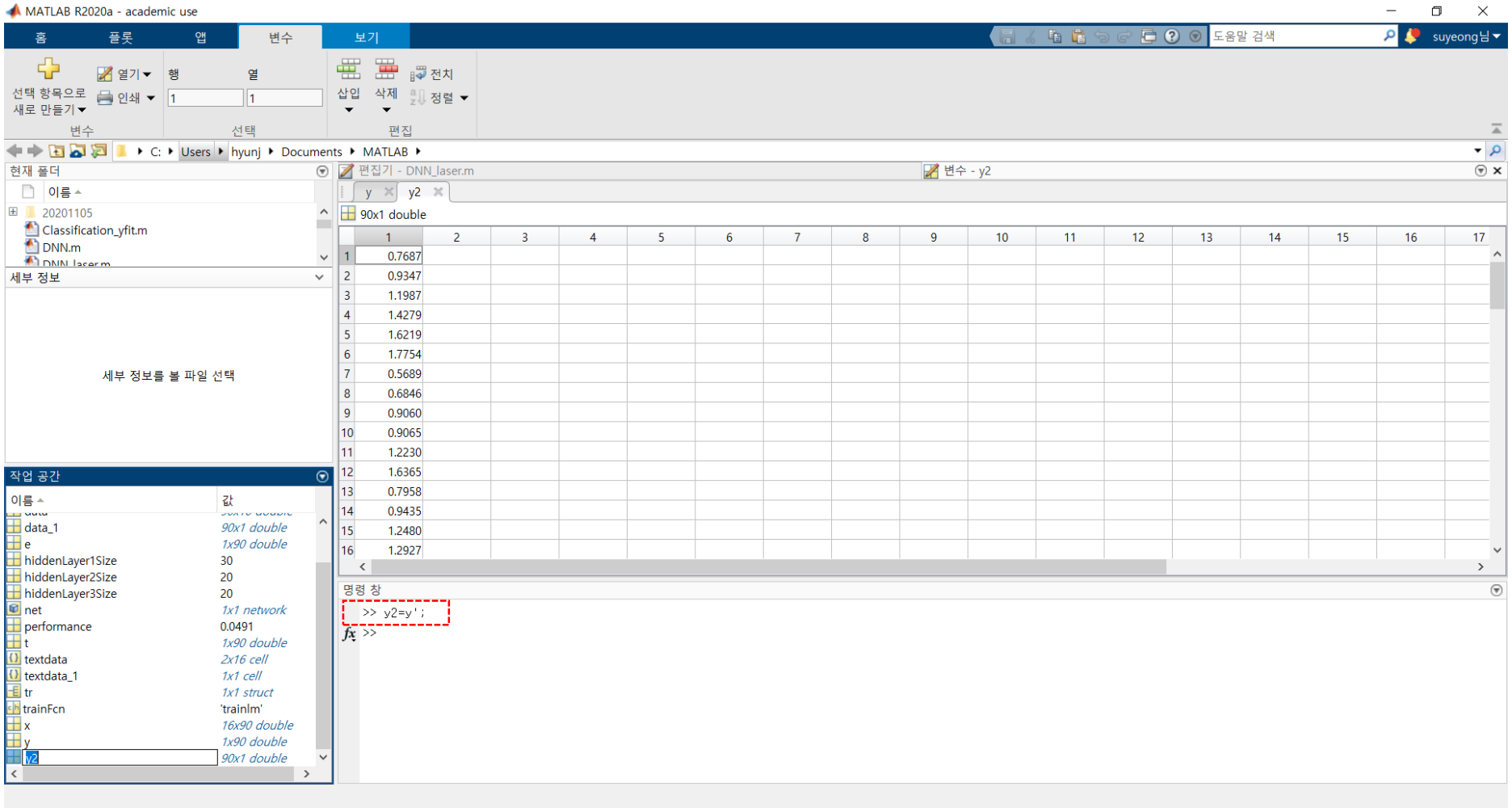
4. MATLAB에서 신경회로망 fitting (DNN)

21. 왼쪽 하단의 작업공간에 생성된 y를 클릭하면 DNN을 통해 학습된 값을 확인할 수 있음

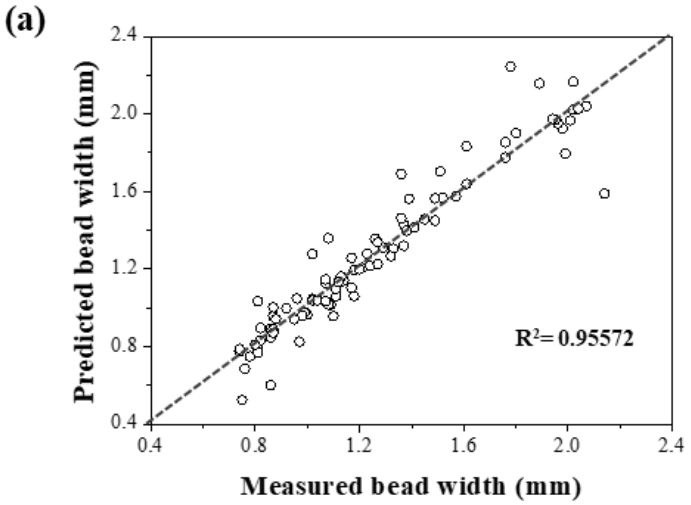


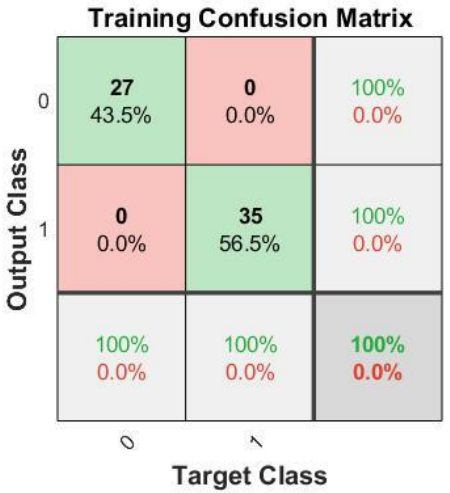
4. MATLAB에서 신경회로망 fitting (DNN)

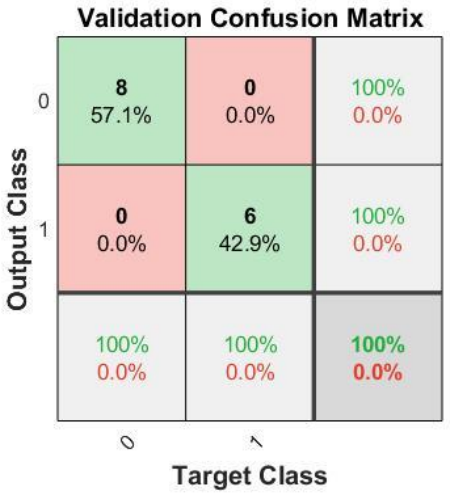
22. 데이터의 행과 열을 바꾸기 위해서 **y2=y'** 라는 명령어를 작성해주면 작업공간에 y2가 생기면서 행과 열이 바뀐 데이터를 받을 수 있음

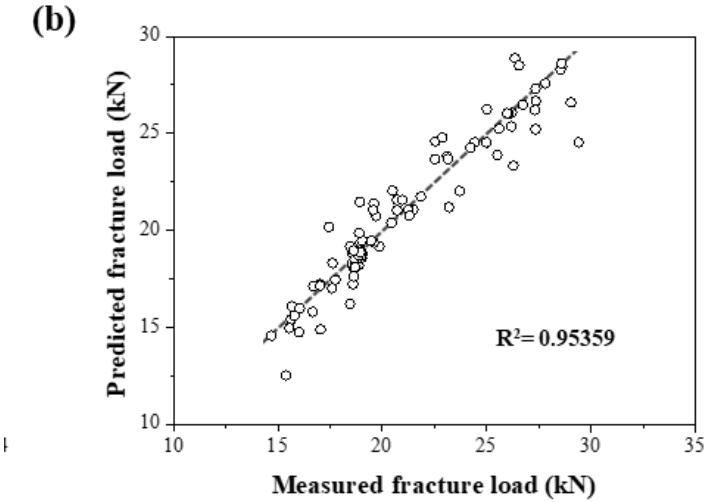


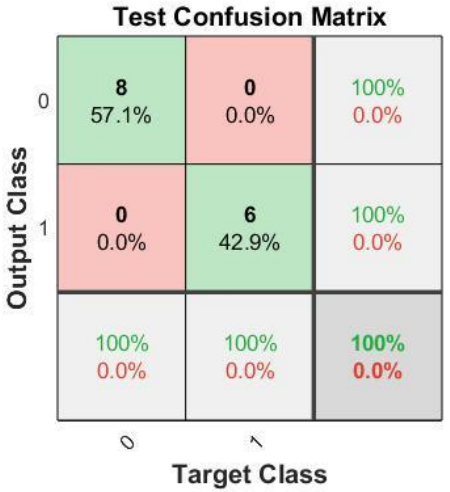
5. 모델의 정확성 판단하기

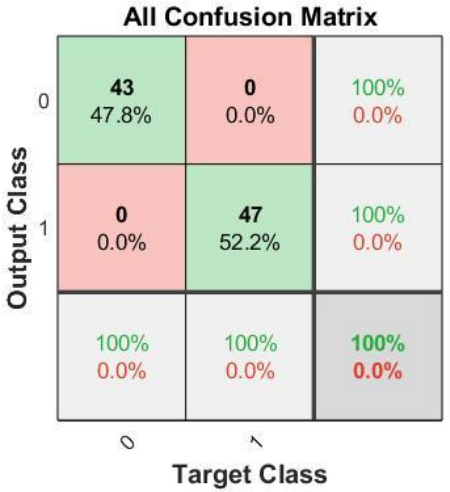












감 사 합 니 다

다음 강의 내용 : 파이썬을 이용한 SNN, DNN 모델 분석 방법