

گزارش شبکه عصبی

استاد راهنما : دکتر شادرو

اعضای گروه :

آریان فرمان

پیمان شبیری

دانیال نهفته

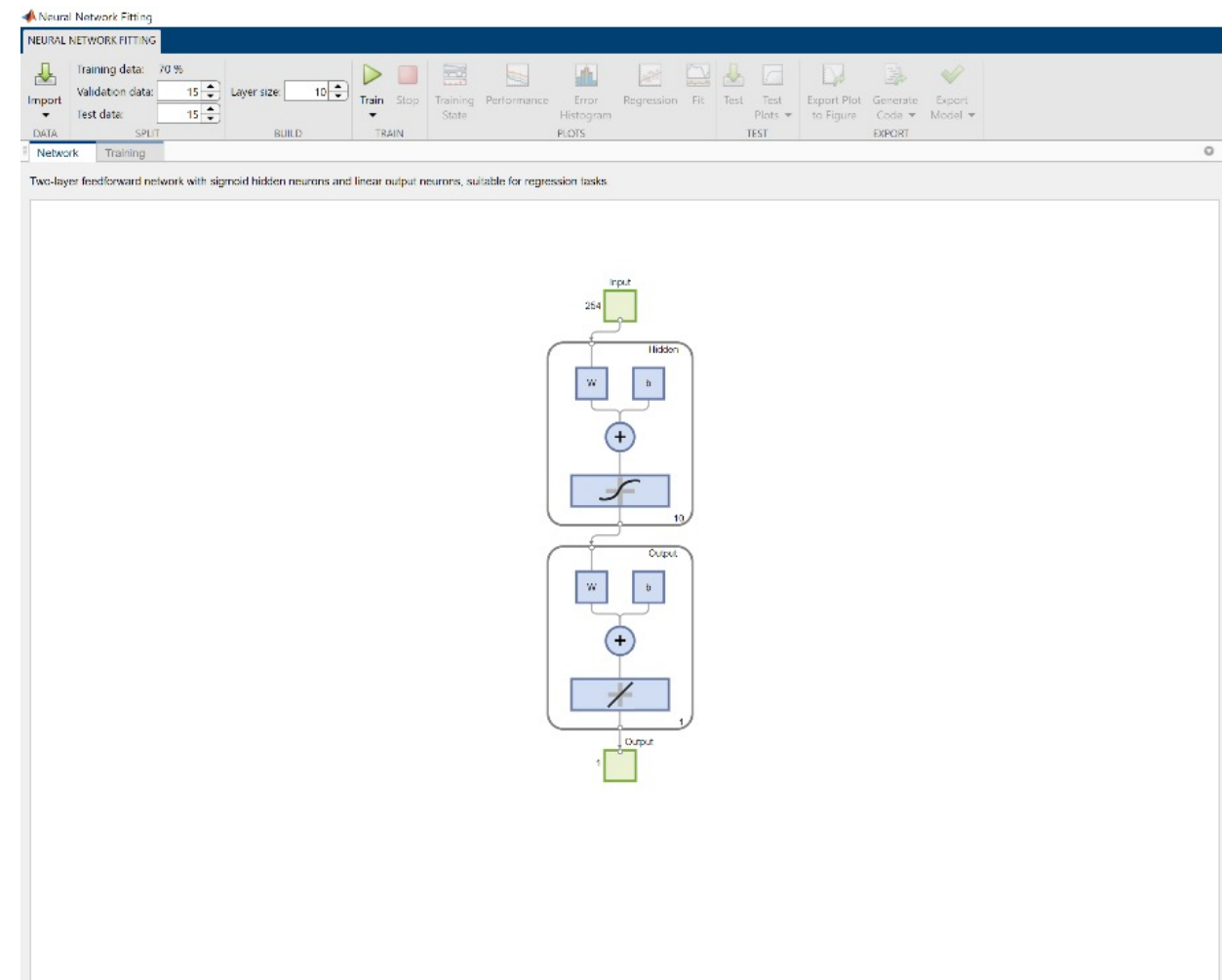
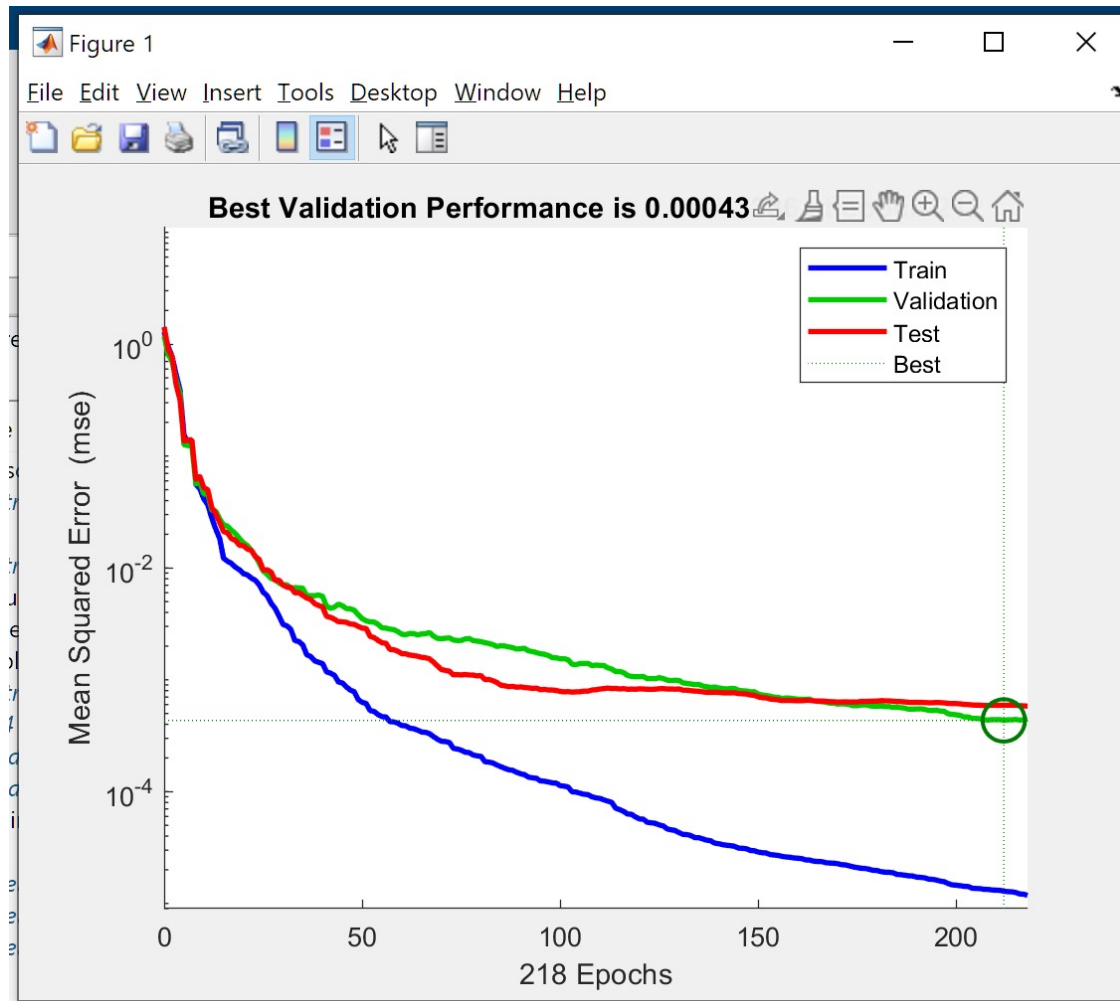
شبکه feed forward 2layers

از آنجایی که شبکه هایی که ما برای توضیح انتخاب کردیم در شبیه ساز وجود ندارد و توی این قبیل از مسائل هم کاربرد ندارند.

باید با استفاده از شبکه های خود متلب آموزش را انجام بدهیم. پس ابتدا از دیتاست ردیف آخر را برای لیبیل ها جدا میکنیم و بعد از 218 اپوک کار به اتمام میرسد. از MSE برای خطا استفاده شده است. از مجموع کل داده ها 70 درصد را برای آموزش شبکه و 15 درصد واسه ارزیابی و 15 درصد واسه تست استفاده شده است.

و دلیل توقف شبکه رسیدن به مقدار validation بود. 254 عدد فیچر داشتیم همراه با 420 داده در دیتاست.

با بررسی نمودار زیر میبینیم که میزان خطای شبکه خوب است و همچنین **overfit** هم اتفاق نیوفتاده است و روی داده آموزشی خطای شبکه به صفر میل میکند و روی داده های ارزیابی و تست میزان خطا بسیار کم میباشد. در قسمت Hidden layer از یک لایه با 10 نورون برای توپولوژی شبکه استفاده شده است.



254 عدد ورودی داریم و یک خروجی و تابع فعالیت لایه خروجی نیز همانی است در واقع اصلا وجود ندارد.

تعداد داده های آموزشی : 294

تعداد داده های ارزیابی : 63

تعداد داده های تست : 63

NEURAL NETWORK FITTING

Import Training data: 70 % Validation data: 15 Layer size: 10 Test data: 15 Train Stop Training State Performance Error Histogram Regression Fit Test Test Plots Export Plot to Figure Generate Code Export Model

DATA SPLIT BUILD TRAIN PLOTS TEST EXPORT

Model Summary

Train a neural network to map predictors to continuous responses.

Data
Predictors: all_data - [420x254 double]
Responses: labels - [420x1 double]
all_data: double array of 420 observations with 254 features.
labels: double array of 420 observations with 1 features.

Algorithm
Data division: Random
Training algorithm: Scaled conjugate gradient
Performance: Mean squared error

Training Results
Training start time: 16 Jan 2022 12:16:08
Layer size: 10

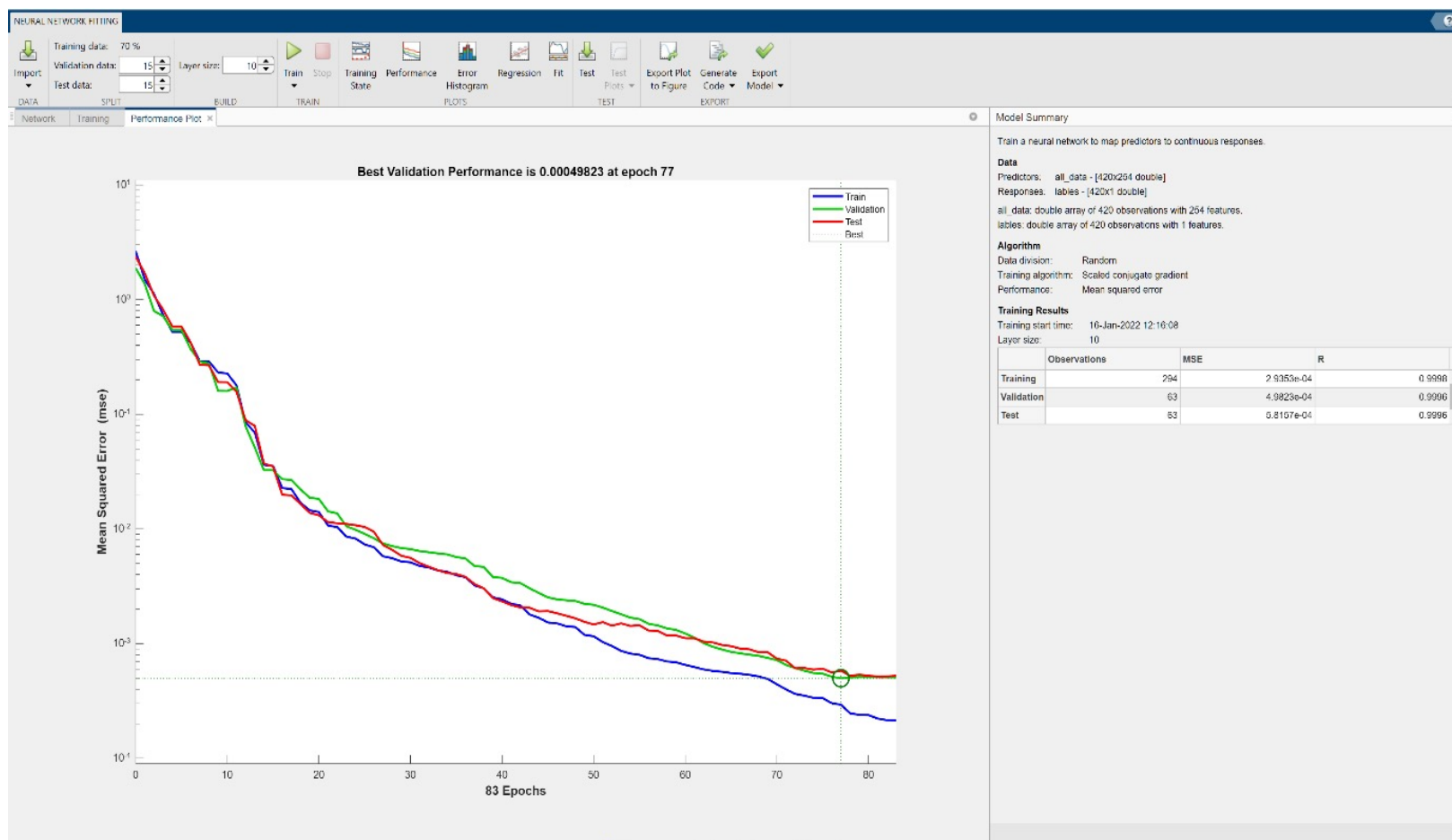
	Observations	MSE	R
Training	294	2.9353e-04	0.9998
Validation	63	4.9823e-04	0.9996
Test	63	5.8167e-04	0.9996

Training Results
Training finished: Met validation criterion

Training Progress

Unit	Initial Value	Stopped Value	Target Value
Epoch	0	83	1000
Elapsed Time	-	00:00:00	-
Performance	2.65	0.000215	0
Gradient	14	0.00298	1e-06
Validation Checks	0	6	6

آموزش همان شبکه با مقادیر متفاوت و الگوریتم آموزش Scaled conjugate gradient



تعداد اپوک روی 83 اپوک تموم میشود . با بررسی نتایج حاصل شده در نمودار بالا پی میبریم که نتایج بهبود داشته و Performance شبکه در حدود 99 درصد شده است.

نتایج مدل اول :

results_1_model.TrainingResults	
Field ^	Value
trainFcn	'trainscg'
trainParam	1x1 struct
performFcn	'mse'
performParam	1x1 struct
derivFcn	'defaultderiv'
divideFcn	'dividerand'
divideMode	'sample'
divideParam	1x1 struct
trainInd	1x294 double
valInd	1x63 double
testInd	1x63 double
stop	'Training finished: Met validation criterion'
num_epochs	83
trainMask	1x1 cell
valMask	1x1 cell
testMask	1x1 cell
best_epoch	77
goal	0
states	1x7 cell
epoch	1x84 double
time	1x84 double
perf	1x84 double
vperf	1x84 double
tperf	1x84 double
gradient	1x84 double
val_fail	1x84 double
best_perf	2.9353e-04
best_vperf	4.9823e-04
best_tperf	5.8157e-04

کد : فایل .m code
لیبل ها : فایل labeles.mat
دیتا ها : فایل all_dat.mat

شبکه feed forward 2layers

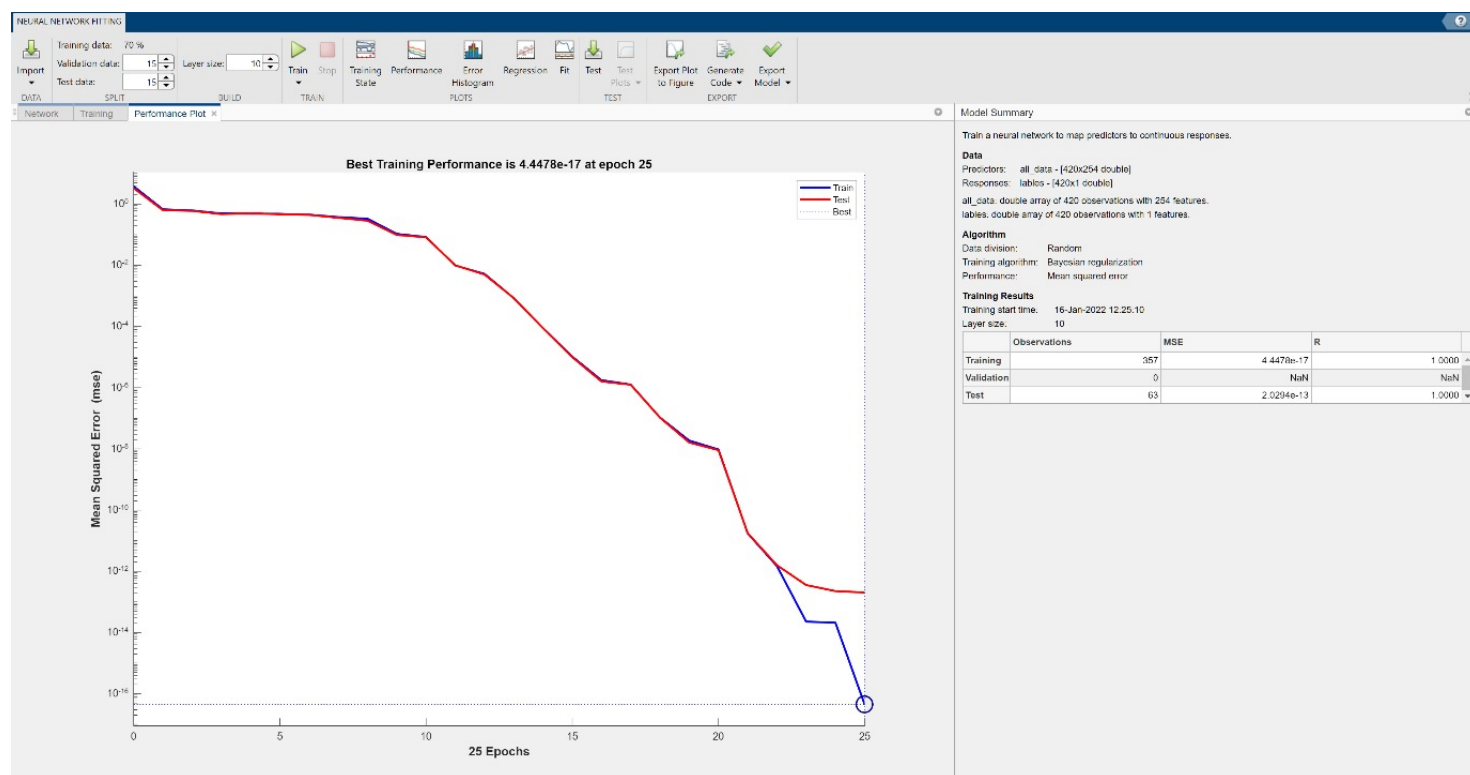
مدل شبکه در این دفعه تغییر نمیکند فقط الگوریتم به bayesian regularization تغییر میدهیم .
و این بار به واسطه کمینه شدن گرادیانت تموم میشه (بعد از 25 اپوک به مینیمم میرسیم) و همین طور که مشاهده میشود زمان آموزش افزایش داشته است . و در این الگوریتم دیگر validation نداریم . و 70 درصد را برای آموزش و 30 درصد باقی مانده را جهت تست استفاده میکنیم .

تعداد داده های آموزشی : 357

تعداد داده های ارزیابی : 0

تعداد داده های تست : 63

و با بررسی نتیجه میبینیم که خطا به سمت صفر میل میکند .



NEURAL NETWORK FITTING

Import

Training data: 70 %

Validation data: 15

Test data: 15

Layer size: 10

Train

Stop

Training State

Performance

Error Histogram

Regression

Fit

Test

Test Plots

Export Plot to Figure

Generate Code

Export Model

DATA

SPLIT

BUILD

TRAIN

PLOTS

TEST

EXPORT

Network

Training

Performance Plot

Training Results

Training finished: Reached minimum gradient

Training Progress

Unit	Initial Value	Stopped Value	Target Value
Epoch	0	25	1000
Elapsed Time	-	00:00:31	-
Performance	3.81	4.45e-17	0
Gradient	26.2	4.41e-09	1e-07
Mu	0.005	5e+03	1e+10
Effective # Param	2.48e+03	355	0
Sum Squared Param	30.8	1.49	0

Model Summary

Train a neural network to map predictors to continuous responses.

Data

Predictors: all_data - [420x254 double]

Responses: labies - [420x1 double]

all_data: double array of 420 observations with 254 features.

labies: double array of 420 observations with 1 features.

Algorithm

Data division: Random

Training algorithm: Bayesian regularization

Performance: Mean squared error

Training Results

Training start time: 16-Jan-2022 12:25:10

Layer size: 10

	Observations	MSE	R
Training	357	4.4478e-17	1.0000
Validation	0	NaN	NaN
Test	63	2.0291e-13	1.0000

HOME PLOTS APPS VARIABLE VIEW

Open Rows Columns Insert Field Delete Field Transpose
New from Selection Print Sort

VARIABLE SELECTION EDIT

C:\Users\aryan\Downloads\Telegram Desktop\mn

Current Folder

- Name*
- matlab.mat
- lables.mat
- all_data.mat
- result1
- res3
- res2

lables.mat (MAT-file)

Name	Value
lables	420x1 double

Editor - code.m

results.TrainingResults.num_epochs x results_1_model x results_1_model.TrainingResults x results_3_model x results_3_model.TrainingResults

results_3_model.TrainingResults

Field	Value
trainFcn	'trainbr'
trainParam	1x1 struct
performFcn	mse
performParam	1x1 struct
divFcn	'defaultdiv'
divFcnParam	divicerend
divideMode	'sample'
divideParam	1x1 struct
trainInd	1x357 double
valInd	1
testInd	1x53 double
stop	'Training finished: Reached minimum gradient'
num_epochs	25
trainMask	1x1 cell
valMask	1x1 cell
testMask	1x1 cell
best_epoch	25
goal	0
states	1x10 cell
epoch	1x26 double
time	1x26 double
perf	1x26 double
vperf	1x26 double
tperf	1x26 double
mu	1x26 double
gradient	1x26 double
gamk	1x26 double
soX	1x26 double
val_fail	1x26 double
best_perf	4.4478e-17
best_vperf	NaN
best_tperf	2.0294e-13

Workspace

Name	Value
all_data	420x254 double
lables	420x1 double
results	1x1 struct
results_1_model	1x1 struct
results_3_model	1x1 struct

Command Window

```
>> load('matlab.mat')
>> load('matlab.mat')
>> load('matlab.mat', 'results')
>> load('all_data.mat')
>> load('lables.mat')
>> mlsart.
fx>>
```