Електротехнички факултет – Универзитет у Београду

Катедра за сигнале и системе

Неуралне мреже (13E054NM)

Први пројектни задатак

P = 3 (*Real estate*)

Студенти:

Матија Јевтић 0114/2017

Никола Васиљевић 0095/2017

1. Улазни сет података

У овом задатку решавамо проблем одређивања цене некретнине (стана) у зависности од неколико улазних познатих података . У нашем *dataset*-у имамо конкретно 7 улазних сигнала који представљају *годину изградње, величину, спрат, врсту грејања, постојање лифтова, постојање парка у близини и даљина школе.* На излазу из неуралне мреже треба наравно да буде цена некретнине (стана).

Посматрајући улазне податке закључујемо да они нису балансирани из разлога што се одређени подаци не појављују равноправно у свим комбинацијама. Проћи ћемо кроз све улазне податке понаособ:

Година изградње стана је податак који нам говори када је стан изграђен и из нашег *dataset*-а који поседујемо видимо да имамо станове из само неколико година тј постоје године за које немамо ниједан стан да је тада изграђен. На слици *Слика1.1* приказани су све вредности године изградње стана које имамо у нашем *dataset*-у док су у табели *Табела1.1* избројана појављивања свих година из *dataset*-а. Из те табеле видимо да подаци нису толико балансирани јер и поред тога што не поседујемо податке за неке године ми за неке године имамо доста више података као што је 2007. година док за неке имамо мањи број података као што је 2003. година.

Слика1.1 Година изградње стана

Год.	**1978.**	**1980.**	**1985.**	**1986.**	**1992.**	**1993.**	**1997.**	**2003.**
N	65	4	65	378	469	678	106	9
Год.	**2005.**	**2006.**	**2007.**	**2008.**	**2009.**	**2013.**	**2014.**	**2015.**
N	737	665	1225	270	238	468	387	131

Табела1.1 Број појављивања варијанти првог улаза – година изградње

Величина стана (квадратура) је други улазни податак и у нашем *dataset* -у поседујемо разноврсне вредности квадратуре које се крећу у опсегу:

$$12.542\,m^2 \;\div\; 217.114\,m^2.$$

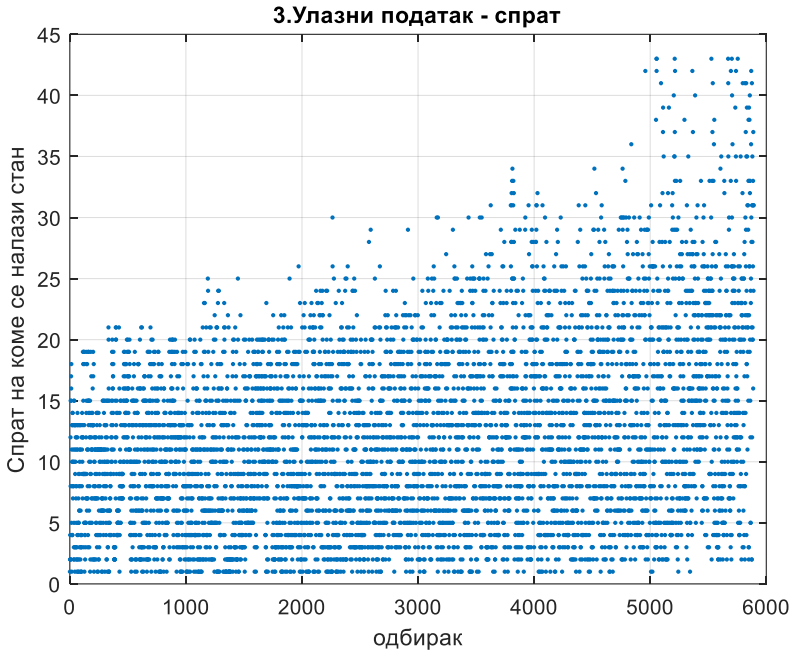
На слици *Слика1.2* приказани су сви улазни подаци за вредност површине стана.

Слика1.2 Величина стана (квадратура)

Спрат је трећи улазни податак који говори на ком спрату се налази стан. Наши подаци имају станове чија се спратност креће у опсегу:

$$1 \;\div\; 43.$$

На слици *Слика1.3* приказани су сви улазни подаци за спратност из нашег *dataset*-а. Јасно се види са слике да имамо пуно више станова који су на спрату нижем од 25-ог спрата што можемо узетио као прихватљиво јер имамо мање високих зграда.



Слика1.3 Спратност стана

Тип грејања је четврти податак који говори о томе на који начин се греју просторије стана. Како улазни податак у неуралну мрежу коју пројектујемо не може бити стринг ослучујемо се да податке заменимо бројевима и то:

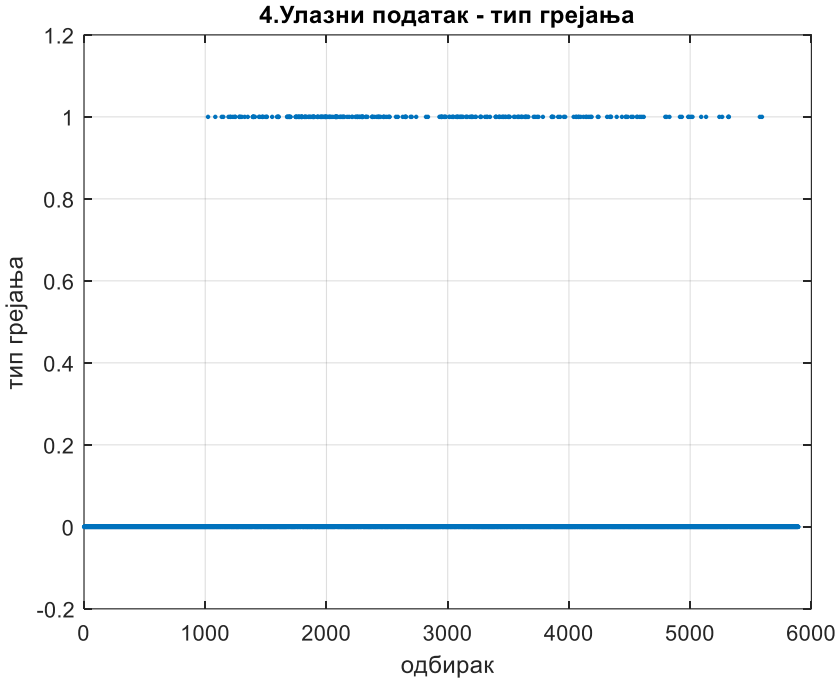
$$Individual\text{-}heating \equiv 0,$$

$$Central\text{-}heating \equiv 1.$$

На слици *Слика1.4* приказани су улазни подаци из *dataset*-а који говоре о типу грејања. Видимо да су ови подаци небалансирани јер имамо много мањи број станова који имају централно грејање што можемо видети и из табеле *Табела1.2* у којој су приказани бројеви понављања ове две врсте грејања.

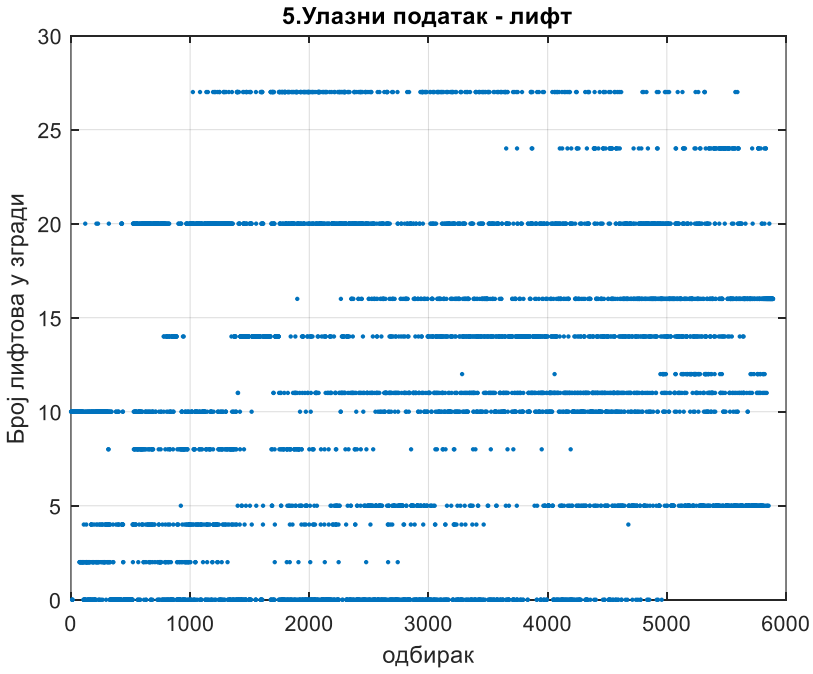
type	***Individual-heating***	***Central-heating***
N	5591	300

Табела1.2 Број појављивања типа грејања у становима



Слика1.4 Тип грејања у стану

Број лифтова је пети параметар који фигурише на улазу система. Видимо да су у опсегу од 0 до 27 и да нису сви бројеви у оптицају. Такође видимо и да је различит распоред самих вредности на улазу који можемо прецизно видети у табели *Табела1.3.*

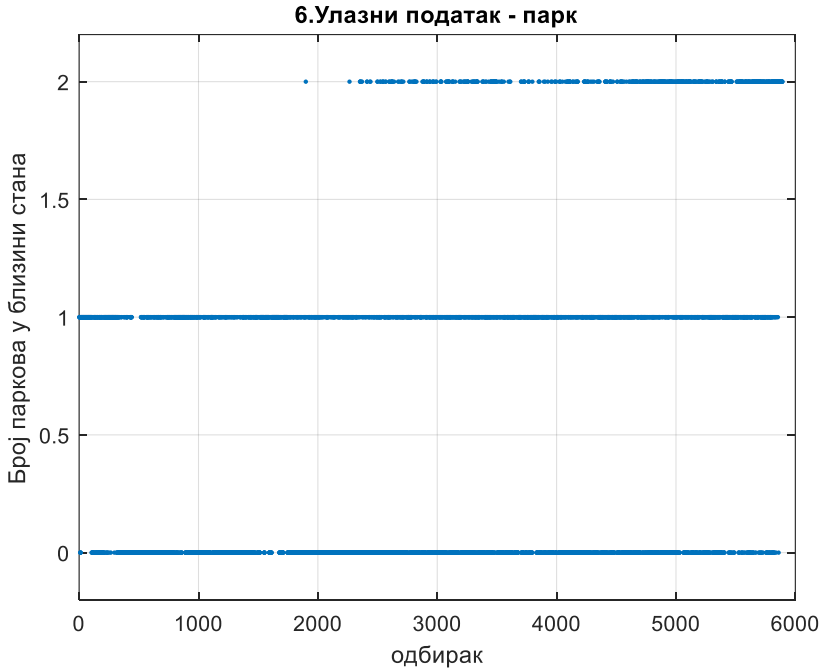


Слика1.5 Број лифтова у згради

лифт	**0**	**2**	**4**	**5**	**8**	**10**	**11**
N	1062	143	180	541	164	637	464
лифт	**12**	**14**	**16**	**20**	**24**	**27**	---
N	64	738	610	877	111	310	---

Табела1.3 Број лифтова

Број паркова у близини је шести податак на улазу и говори о томе колико се паркова налази у близини стана. Из нашег *dataset*-а видимо да имамо могућности 0, 1 или 2. На слици *Слика1.6* приказани су ови улазни подаци док је у табели *Табела1.4* приказан број станова у чијој близини се налазе 0, 1 или 2 парка.

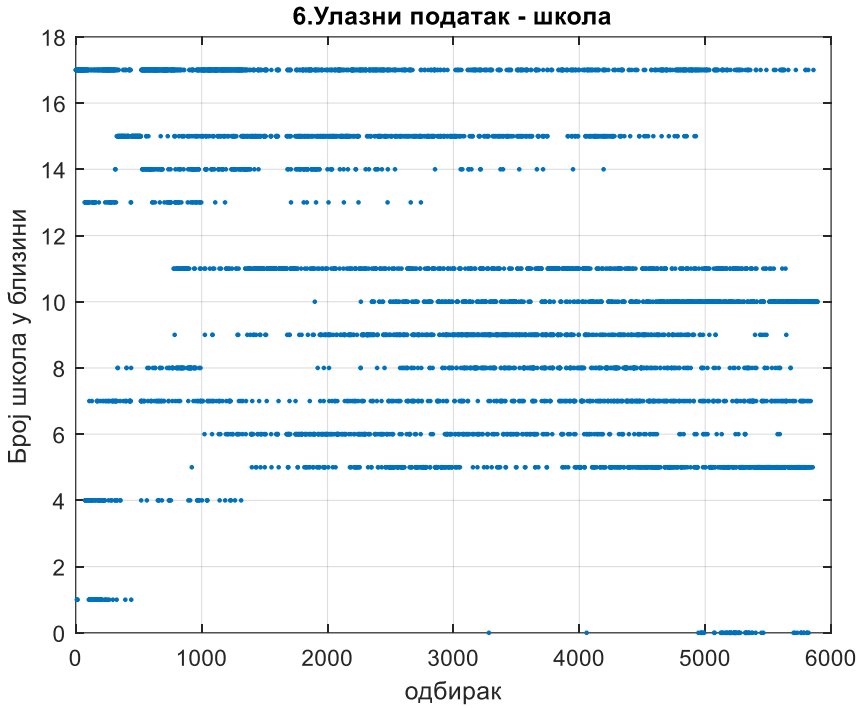


Слика1.6 Број паркова

Бр. паркова	**0**	**1**	**2**
N	2647	2634	610

Табела1.4 Расподела броја паркова у близини стана

Близина школе је последњи улазни податак који се креће у опсегу од 0 до 17 и приказани су на слици *Слика1.7.*



Слика1.7 Број школа у близини

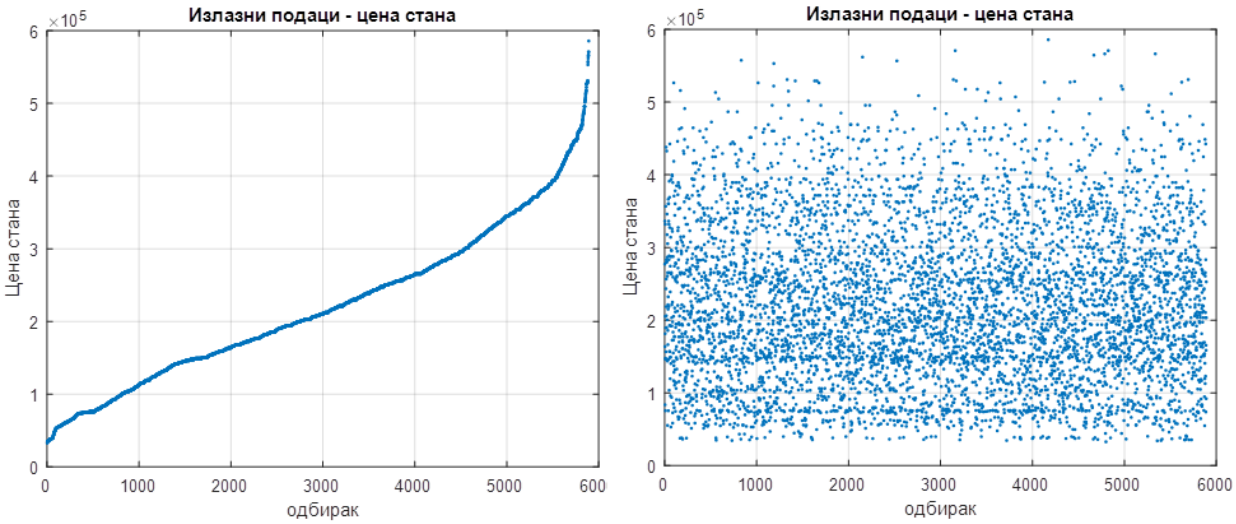
2. Излазни подаци

Потребно је да на излазу из наше неуралне мреже имамо цену стана. Оно што имамо у нашем *dataset*-у за тренирање наше неуралне мреже је приказано на слици *Слика2.1*. Излазни подаци су сортирани и видимо да се цене станова крећу у опсегу:

$$32{,}743 \div 585{,}840.$$

Како су излазни подаци сортирани у растућим вредностима цене станова морамо их мало помешати како неурална мрежа не би кренула да се обучава прво са јефтиним становима па тек на крају са скупљим. Након мешања добијамо распоред излазних података на слици *Слика2.1* десно.

Како би смо добро обучили мрежу све улазне и излазне податке морамо одговарајуће скалирати тако да буду у опсегу $[\,0 \div 1\,]$.



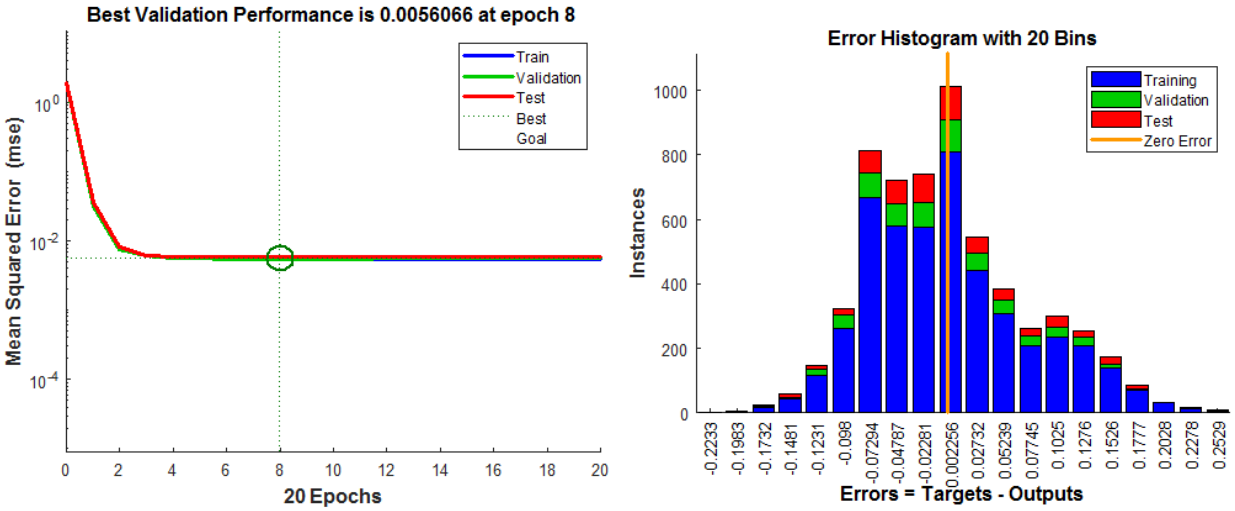
Слика2.1 Излазни подаци пре (лево) и након мешања (десно)

3. Обучавање неуралне мреже

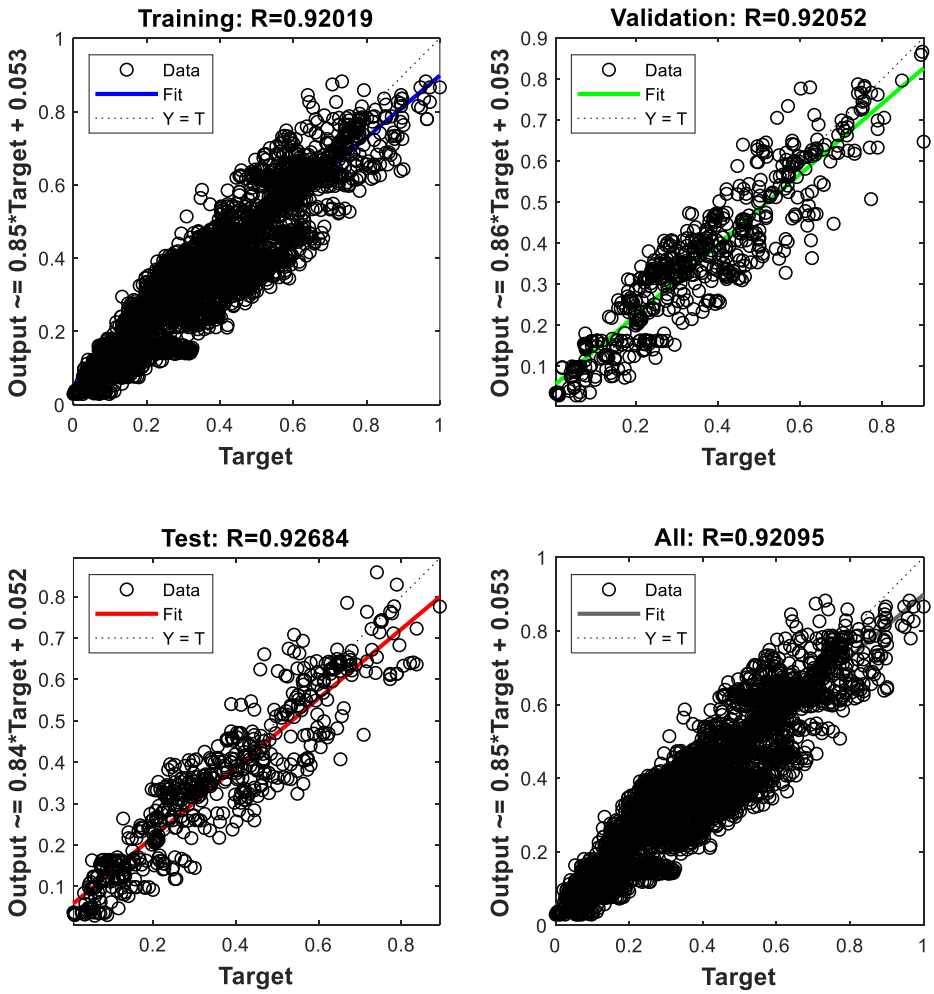
Осим тога што смо податке нормализовали током тренирања неуралне мреже их делимо на скупове за тестирање, валидацију и тест скуп и то у размери 80%, 10% и 10%, респективно. Битно је да овако поделимо податке како би били сигурни да се мрежа током тренирања не преобучи.

Прво смо за тренирање користили ***trainlm*** *(Levenberg-Marquardt backpropagation)* тренинг функцију и као критеријум за мерење перформанси користимо *средњу квадратну грешку (mse – Mean squared error).* Овако тренирана неуралн мрежа не захтева подешавање параметара и она у себи има уграђен алгоритам за њихову оптимизацију. Тренирањем неуралне мреже за разне њене структуре добијамо да је најоптималније да наша неуралн мрежа има структуру од 3 слоја у којима се налази по 25 неурона и излазним слојем са линеарном активационом функцијом као што је то приказано на слици *Слика3.1.* испод.

Слика3.1 Структура наше неуралне мреже



Слика3.2 Тренирање неуралне мреже trainlm тренинг функцијом



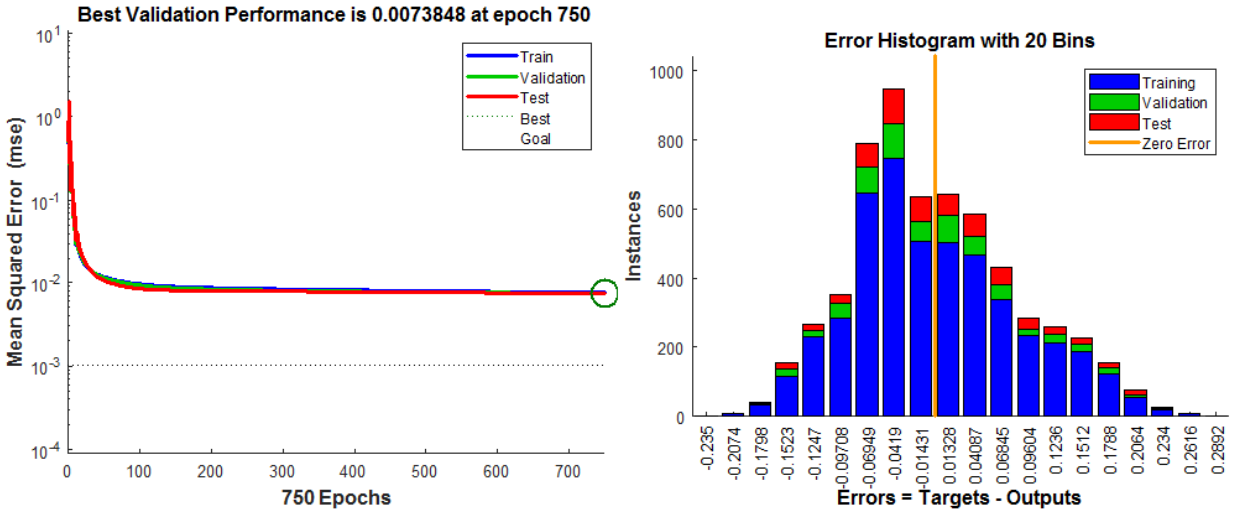
Слика3.3 Regression plot

На сликама *Слика3.2* и *Слика3.3* приказане су перформансе неуралне мреже током њеног тренирања *trainlm* тренинг функцијом. Како се у задатку захтева да помоћу кросвалидације изаберемо сет оптималних параметара поново ћемо тренирати нашу неуралну мрежу али сада са ***traingdm*** *(Gradient descent with momentum)* тј методом градијентног спуста као тренинг функцијом. Трудићемо се да добијемо сличне резултате као када смо неуралну мрежу тренирали са *trainlm* тренинг функцијом. Поново као критеријум за мерење перформанси узимамо *средњу квадратну грешку (mse).*

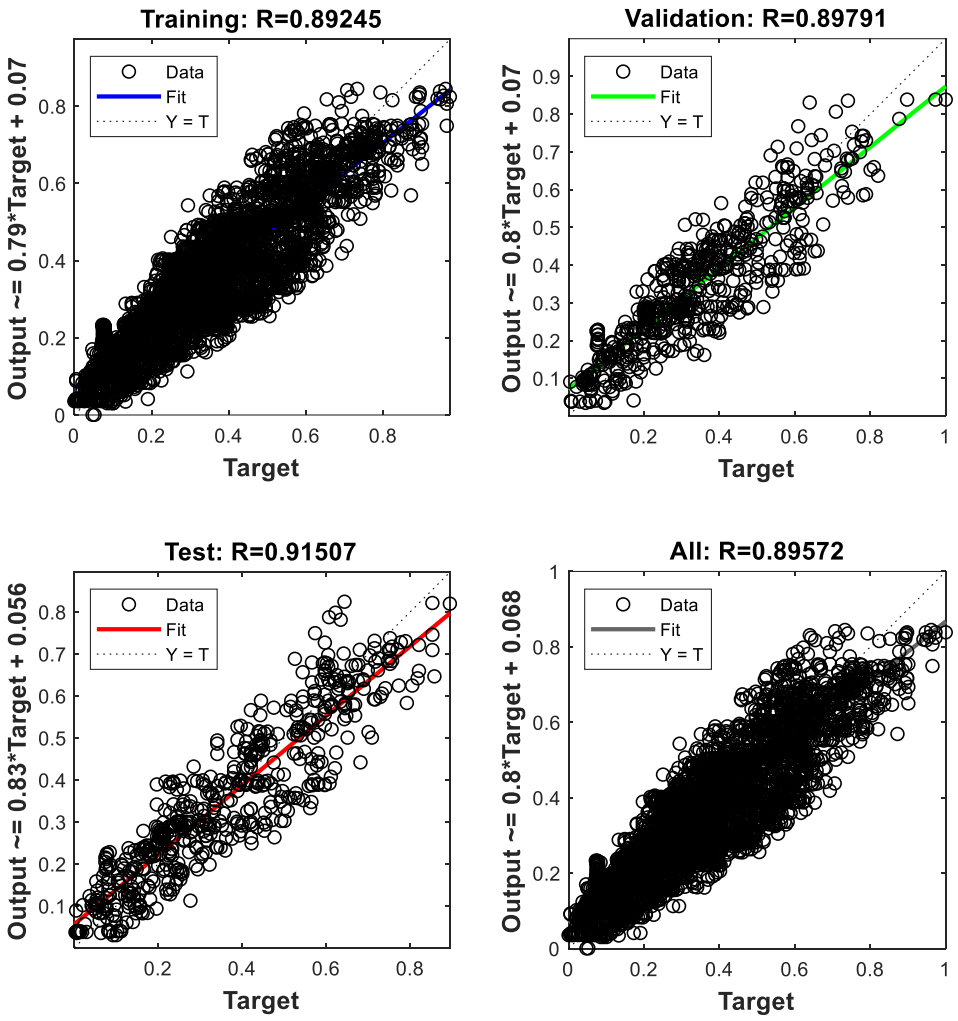
Поново смо податке поделили на скупове за тестирање, валидацију и тест скуп и то поново у истој размери 80%, 10% и 10%, респективно. За хиперпараметре чије оптималне вредности тражимо смо изабрали *константу обучавања (η), коефицијент регуларизације(γ) и моментум(α)*. Након спровођења поступка кросвалидације добијамо да оптималне вредности ових параметара износе:

$$\eta = 0.35, \qquad \gamma = 0.001, \qquad \alpha = 0.75.$$

На сликама *Слика3.4* и *Слика3.5* можемо видети перформансе неуралне мреже током тестирања за оптималне вреднсти хиперпараметара. Са ових слика видимо да смо добили сличне резултате као и код првобитног тренирања неуралне мреже *trainlm* тренинг функцијом.

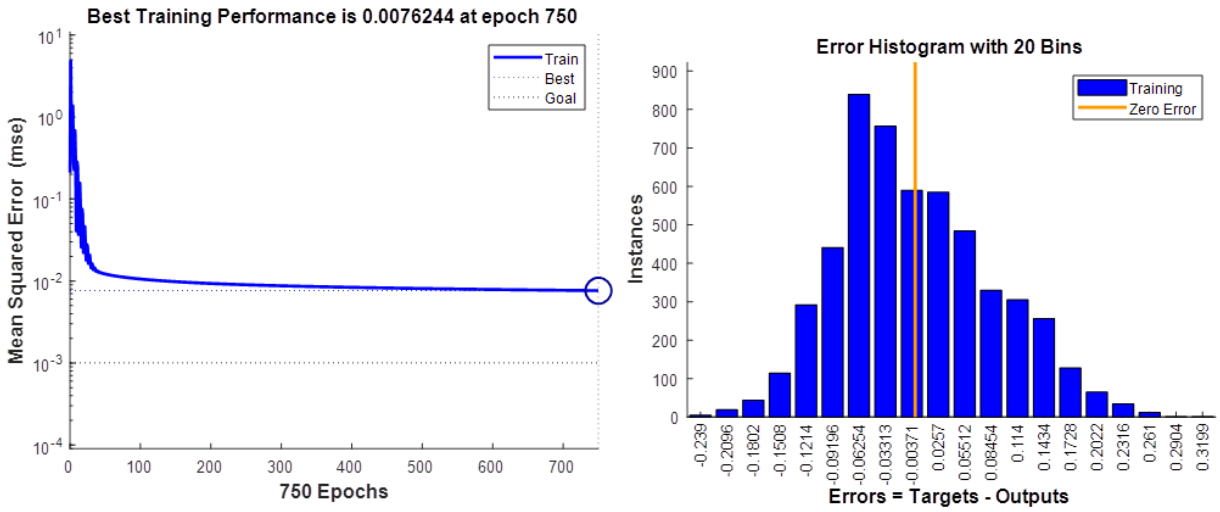


Слика3.4 Тренирање неуралне мреже оптималним параметрима

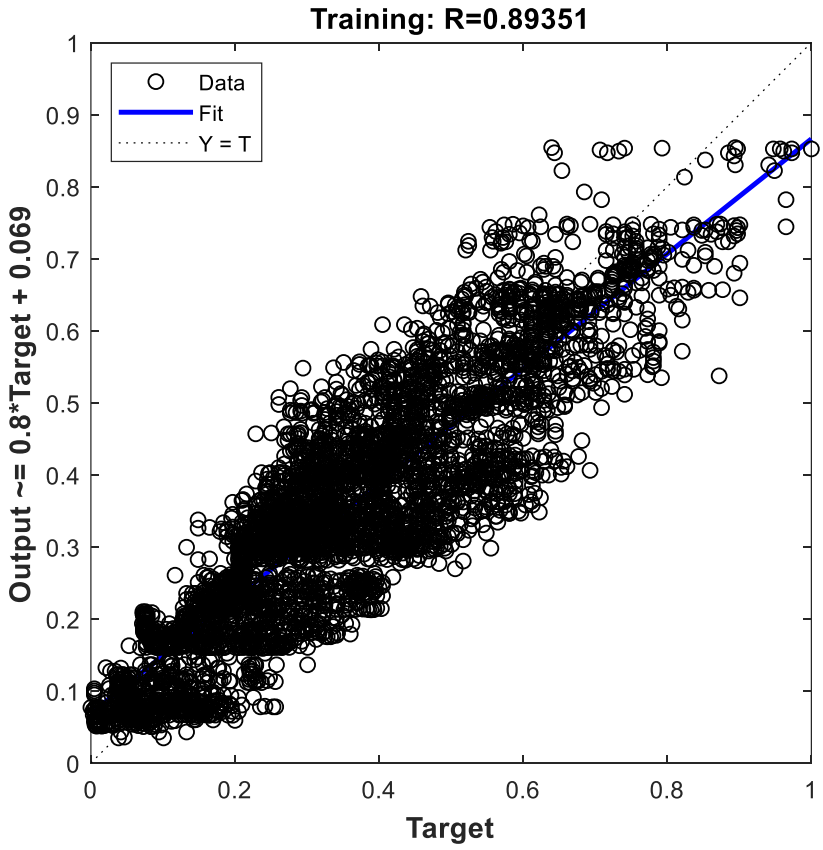


Слика3.5 Regression plot

Након овога закључујемо да неуралну мрежу можемо да тренирамо са већим тренинг скупом тако што ћемо да спојимо скупове за тренинг и за валидацију а да неће доћи до њеног преобучавања. Након тренирања неуралне мреже са проширеним скупом за тренирање и истим оптималним параметрима добијамо резултате приказане на сликама *Слика3.6* и *Слика3.7*. На слици *Слика3.6* видимо хистограм грешке на којој видимо да постоје и грешке које имају величину од 0.3 (160,000) али да већина одбирака има грешку мању од 0.15 (80,000) док највећи број одбирака има грешку мању од 0.09 (50,000). Сам податак да велики број одбирака има грешку већу од 50,000 говори да мрежа не ради баш задивљујуће али да ради солидно. Наиме огроман кривац за овакве резултате јесте улазни сет података који је небалансиран и који нема све могућности улазних података.

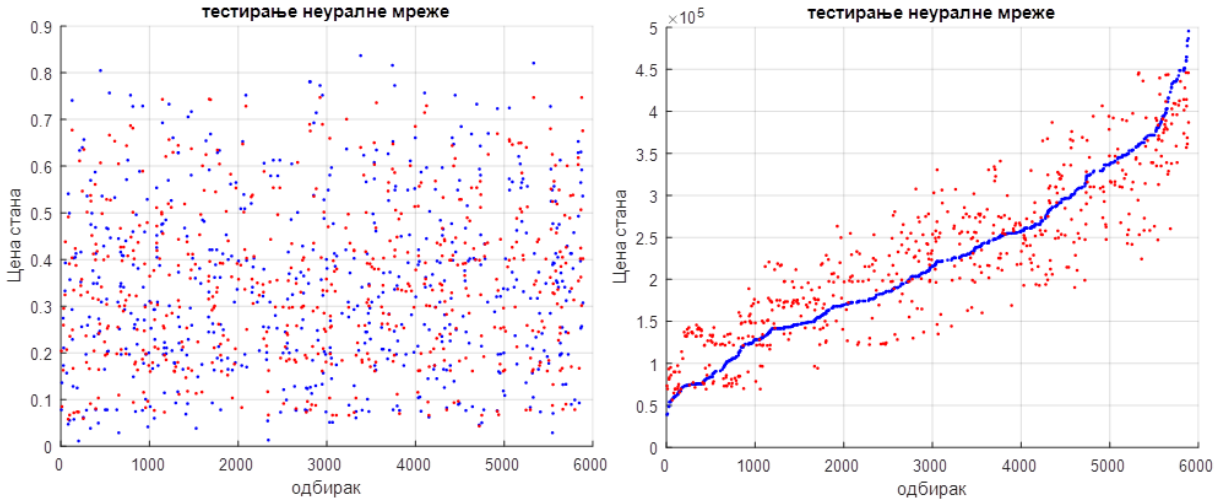


Слика3.6 Тренирање неуралне мреже са проширеним скупом за тренирање

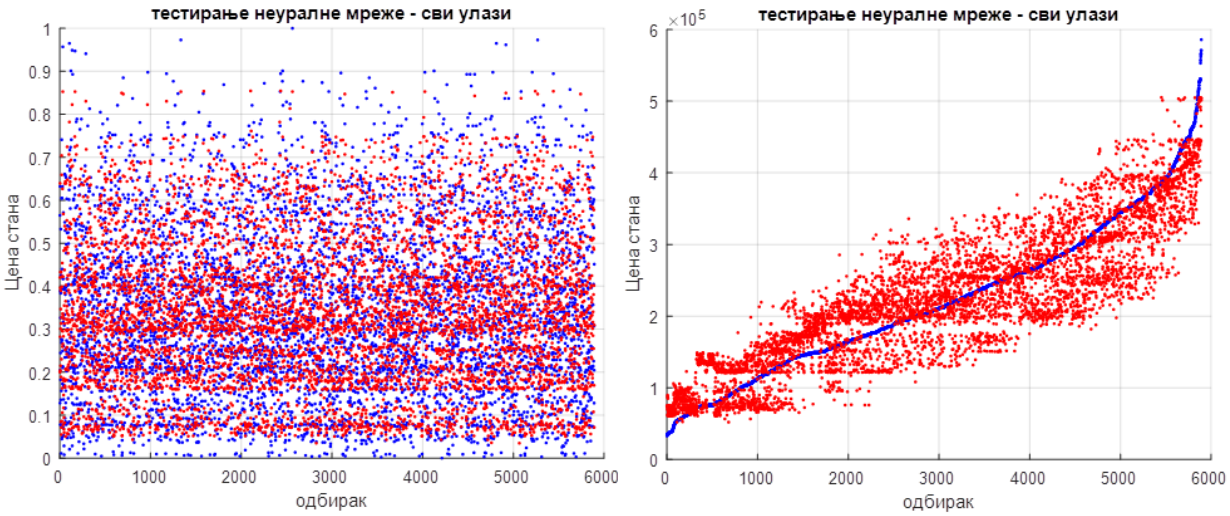


Слика3.7 Regression plot

Након обучавања наше неуралне мреже можемо тестирати одбирке који су одвојени у тест скупу. Након довођења ових података на улаз неуралне мреже добијамо добијамо излазе који су приказани на слици *Слика3.8* лево на којој видимо плавом бојом обележене потребне вредности излаза док црвеном бојом имамо обележене излазе наше неуралне мреже. Пошто су подаци на улазу хаотични тј очекују се хаотични излази из којих не можемо ништа закључити а и добијамо нормализоване излазе морамо их поређати по растућим вредностима жељених излаза као што смо то имали на самом почетку и вратити у опсег који одговара правим вредностима цена станова. Након тога добијамо резултате који су приказани на слици *Слика3.8* десно. Можемо сада пропустити све податке које имамо кроз неуралну мрежу и добијамо излазе који су приказани на слици *Слика3.9.* Са ових слика можемо закључити да мрежа ради солидно и да прати тренд у ценама који се јавља на излазу.



Слика3.8 Тестирање неуралне мреже – тест скуп



Слика3.9 Тестирање неуралне мреже – цео скуп података