PROJECT REPORT FOR

MULTIPLE LAYER PERCEPTRON

BY

ANSHUMAN VAIDYANATHAN

WITAUT BAJARYN

Faculty of Mathematics and Information Sciences

WARSAW UNIVERSITY OF TECHNOLOGY

Author Note

Report submitted as part of Neural Networks course

ACKNOWLEDGEMENT

We thank Prof." Jacek Mandziuk " for presenting us with the opportunity to work on this project "Multiple Layer Perceptron" and we are very grateful for the timed help provided by him for completion of this project.

We also would like to thank our friends and family for giving moral support without which completion of this project was not possible.

Anshuman Vaidyanathan

Witaut Bajaryn

Contents

Multiple Layer Perceptron

Introduction:

A multilayer perceptron (MLP) is a deep, artificial neural network. It is composed of more than one perceptron. They are composed of an input layer to receive the signal, an output layer that makes a decision or prediction about the input, and in between those two, an arbitrary number of hidden layers that are the true computational engine of the MLP. MLPs with one hidden layer are capable of approximating any continuous function.

Multilayer perceptrons are often applied to supervised learning problems: they train on a set of input-output pairs and learn to model the correlation (or dependencies) between those inputs and outputs. Training involves adjusting the parameters, or the weights and biases, of the model in order to minimize error. Backpropagation is used to make those weight and bias adjustments relative to the error, and the error itself can be measured in a variety of ways, including by root mean squared error (RMSE).

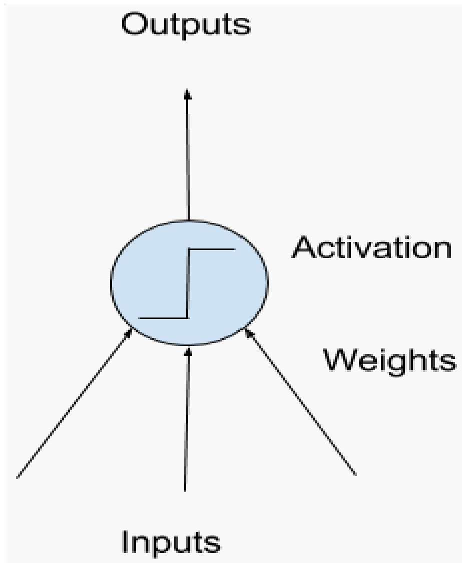
In the forward pass, the signal flow moves from the input layer through the hidden layers to the output layer, and the decision of the output layer is measured against the ground truth labels.

In the backward pass, using backpropagation and the chain rule of calculus, partial derivatives of the error function w.r.t. the various weights and biases are back-propagated through the MLP. That act of differentiation gives us a gradient, or a landscape of error, along which the parameters may be adjusted as they move the MLP one step closer to the error minimum. This can be done with any gradient-based optimization algorithm such as stochastic gradient descent.

1. Neurons:

The building block for neural networks are artificial neurons.

These are simple computational units that have weighted input signals and produce an output signal using an activation function.



Model of a Simple Neuron

2. Neuron Weights:

You may be familiar with linear regression, in which case the weights on the inputs are very much like the coefficients used in a regression equation.

Like linear regression, each neuron also has a bias which can be thought of as an input that always has the value 1.0 and it too must be weighted.

For example, a neuron may have two inputs in which case it requires three weights. One for each input and one for the bias.

Weights are often initialized to small random values, such as values in the range 0 to 0.3, although more complex initialization schemes can be used.

Like linear regression, larger weights indicate increased complexity and fragility. It is desirable to keep weights in the network small and regularization techniques can be used

3. Activation:

The weighted inputs are summed and passed through an activation function, sometimes called a transfer function. An activation function is a simple mapping of summed weighted input to the output of the neuron. It is called an activation function because it governs the threshold at which the neuron is activated and strength of the output signal.

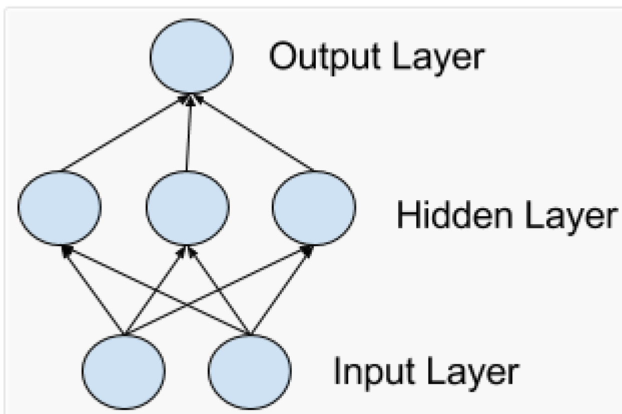
Historically simple step activation functions were used where if the summed input was above a threshold, for example 0.5, then the neuron would output a value of 1.0, otherwise it would output a 0.0.

Traditionally non-linear activation functions are used. This allows the network to combine the inputs in more complex ways and in turn provide a richer capability in the functions they can model. Non-linear functions like the logistic also called the sigmoid function were used that output a value between 0 and 1 with an s-shaped distribution.

4. Networks of Neurons:

Neurons are arranged into networks of neurons.

A row of neurons is called a layer and one network can have multiple layers. The architecture of the neurons in the network is often called the network topology.



Model of a Simple Network

1. Input or Visible Layers
   The bottom layer that takes input from the dataset is called the visible layer, because it is the exposed part of the network. Often a neural network is drawn with a visible layer with one neuron per input value or column in the dataset. These are not neurons as described above, but simply pass the input value through to the next layer.

2. Hidden Layers
   Layers after the input layer are called hidden layers because they are not directly exposed to the input. The simplest network structure is to have a single neuron in the hidden layer that directly outputs the value.

3. Output Layer
   The final hidden layer is called the output layer and it is responsible for outputting a value or vector of values that correspond to the format required for the problem.

The choice of activation function in the output layer is strongly constrained by the type of problem that is being modeled.

In this case, it was a classification problem, and a sigmoid activation function was used. Additionally, the network's output had to be made discrete. Thresholding was used for this: the output range was divided into equally-sized subranges (one for each possible category), and the respective category was chosen based on which subrange the real-valued output fell into.

5. Training Networks:

Once configured, the neural network needs to be trained on the dataset.

1. Data Preparation:

The datasets have categorical and numerical data. Categorical data had to be made numerical to apply the network on it. For the flags dataset, color names had to be converted to integers. For the ionosphere dataset, the "good/bad" value was converted to 1 or 0. For the horse colic dataset, the unknown values were replaced with -1.

2. Stochastic Gradient Descent:

The classical and still preferred training algorithm for neural networks is called stochastic gradient descent.

This is where one row of data is exposed to the network at a time as input. The network processes the input upward activating neurons as it goes to finally produce an output value. This is called a forward pass on the network. It is the type of pass that is also used after the network is trained in order to make predictions on new data.

The output of the network is compared to the expected output and an error is calculated. This error is then propagated back through the network, one layer at a time, and the weights are updated according to the amount that they contributed to the error. This clever bit of math is called the **backpropagation algorithm**.

The process is repeated for all of the examples in the training data. One iteration of updating the network for the entire training dataset is called an epoch. A network may be trained for tens, hundreds or many thousands of epochs.

3. Weight Updates:

The weights in the network can be updated from the errors calculated for each training example and this is called online learning. It can result in fast but also chaotic changes to the network.

Alternatively, the errors can be saved up across all of the training examples and the network can be updated at the end. This is called batch learning and is often more stable.

Typically, because datasets are so large and because of computational efficiencies, the size of the batch, the number of examples the network is shown before an update is often reduced to a small number, such as tens or hundreds of examples.

The amount that weights are updated is controlled by a configuration parameters called the learning rate. It is also called the step size and controls the step or change made to network weight for a given error. Often small weight sizes are used such as 0.1 or 0.01 or smaller.

The update equation can be complemented with additional configuration terms that you can set.

- Momentum is a term that incorporates the properties from the previous weight update to allow the weights to continue to change in the same direction even when there is less error being calculated.
- Learning Rate Decay is used to decrease the learning rate over epochs to allow the network to make large changes to the weights at the beginning and smaller fine tuning changes later in the training schedule.

4. Prediction:

Once a neural network has been trained it can be used to make predictions.

You can make predictions on test or validation data in order to estimate the skill of the model on unseen data. You can also deploy it operationally and use it to make predictions continuously.

The network topology and the final set of weights is all that you need to save from the model. Predictions are made by providing the input to the network and performing a forward-pass allowing it to generate an output that you can use as a prediction.

Datasets:

1. Flag database:

This data file contains details of various nations and their flags. In this file the fields are separated by spaces (not commas). With this data you can try things like predicting the religion of a country from its size and the colors in its flag.

-- 10 attributes are numeric-valued. The remainder are either Boolean- or nominal-valued.

Number of attributes: 30 (overall)

In this work, the attributes related to the shape and colors of the flag were used as inputs and predicting the country's religion was attempted.

Attribute Information:

1. name Name of the country concerned

2. landmass 1=N.America, 2=S.America, 3=Europe, 4=Africa, 4=Asia, 6=Oceania

3. zone Geographic quadrant, based on Greenwich and the Equator

   1=NE, 2=SE, 3=SW, 4=NW

4. area in thousands of square km

5. population in round millions

6. language 1=English, 2=Spanish, 3=French, 4=German, 5=Slavic, 6=Other

   Indo-European, 7=Chinese, 8=Arabic,

   9=Japanese/Turkish/Finnish/Magyar, 10=Others

7. religion 0=Catholic, 1=Other Christian, 2=Muslim, 3=Buddhist, 4=Hindu,
    5=Ethnic, 6=Marxist, 7=Others
8. bars Number of vertical bars in the flag
9. stripes Number of horizontal stripes in the flag
10. colours Number of different colours in the flag
11. red 0 if red absent, 1 if red present in the flag
12. green same for green
13. blue same for blue
14. gold same for gold (also yellow)
15. white same for white
16. black same for black
17. orange same for orange (also brown)
18. mainhue predominant colour in the flag (tie-breaks decided by taking
    the topmost hue, if that fails then the most central hue,
    and if that fails the leftmost hue)
19. circles Number of circles in the flag
20. crosses Number of (upright) crosses
21. saltires Number of diagonal crosses
22. quarters Number of quartered sections
23. sunstars Number of sun or star symbols
24. crescent 1 if a crescent moon symbol present, else 0
25. triangle 1 if any triangles present, 0 otherwise
26. icon 1 if an inanimate image present (e.g., a boat), otherwise 0
27. animate 1 if an animate image (e.g., an eagle, a tree, a human hand)
    present, 0 otherwise
28. text 1 if any letters or writing on the flag (e.g., a motto or
    slogan), 0 otherwise
29. topleft colour in the top-left corner (moving right to decide
    tie-breaks)

30. botright Colour in the bottom-left corner (moving left to decide

tie-breaks)

Missing values: None

2. Horse-Colic Database:

2 data files

-- horse-colic.data: 300 training instances

-- horse-colic.test: 68 test instances

-- Possible class attributes: 24 (whether lesion is surgical)

-- others include: 23, 25, 26, and 27

-- Many Data types: (continuous, discrete, and nominal)

Number of attributes: 28

In this work, the attributes known before treatment were used as inputs and predicting the outcome (attribute 23) was attempted.

Attribute Information:

1: surgery?

1 = Yes, it had surgery

2 = It was treated without surgery

2: Age

1 = Adult horse

2 = Young (< 6 months)

3: Hospital Number

- numeric id

- the case number assigned to the horse

(may not be unique if the horse is treated > 1 time)

4: rectal temperature

- linear

- in degrees celsius.

- An elevated temp may occur due to infection.

- temperature may be reduced when the animal is in late shock

- normal temp is 37.8

- this parameter will usually change as the problem progresses

  eg. may start out normal, then become elevated because of the lesion, passing back through the normal range as the horse goes into shock

5: pulse

- linear

- the heart rate in beats per minute

- is a reflection of the heart condition: 30 -40 is normal for adults

- rare to have a lower than normal rate although athletic horses may have a rate of 20-25

- animals with painful lesions or suffering from circulatory shock may have an elevated heart rate

6: respiratory rate

- linear

- normal rate is 8 to 10

- usefulness is doubtful due to the great fluctuations

7: temperature of extremities

- a subjective indication of peripheral circulation

- possible values:

  1 = Normal

  2 = Warm

  3 = Cool

  4 = Cold

- cool to cold extremities indicate possible shock

- hot extremities should correlate with an elevated rectal temp.

8: peripheral pulse

- subjective

- possible values are:

1 = normal

2 = increased

3 = reduced

4 = absent

- normal or increased p.p. are indicative of adequate circulation while reduced or absent indicate poor perfusion

9: mucous membranes

- a subjective measurement of colour
- possible values are:

1 = normal pink

2 = bright pink

3 = pale pink

4 = pale cyanotic

5 = bright red / injected

6 = dark cyanotic

- 1 and 2 probably indicate a normal or slightly increased circulation
- 3 may occur in early shock
- 4 and 6 are indicative of serious circulatory compromise
- 5 is more indicative of a septicemia

10: capillary refill time

- a clinical judgement. The longer the refill, the poorer the circulation
- possible values

1 = < 3 seconds

2 = >= 3 seconds

11: pain - a subjective judgement of the horse's pain level

- possible values:

1 = alert, no pain

2 = depressed

3 = intermittent mild pain

4 = intermittent severe pain

5 = continuous severe pain

- should NOT be treated as a ordered or discrete variable!

- In general, the more painful, the more likely it is to require surgery

- prior treatment of pain may mask the pain level to some extent

12: peristalsis

- an indication of the activity in the horse's gut. As the gut becomes more distended or the horse becomes more toxic, the activity decreases

- possible values:

1 = hypermotile

2 = normal

3 = hypomotile

4 = absent

13: abdominal distension

- An IMPORTANT parameter.

- possible values

1 = none

2 = slight

3 = moderate

4 = severe

- an animal with abdominal distension is likely to be painful and have reduced gut motility.

- a horse with severe abdominal distension is likely to require surgery just to relieve the pressure

14: nasogastric tube

- this refers to any gas coming out of the tube
- possible values:

  1 = none

  2 = slight

  3 = significant

- a large gas cap in the stomach is likely to give the horse discomfort

15: nasogastric reflux

- possible values

  1 = none

  2 = > 1 liter

  3 = < 1 liter

- the greater amount of reflux, the more likelihood that there is some serious obstruction to the fluid passage from the rest of the intestine

16: nasogastric reflux PH

- linear
- scale is from 0 to 14 with 7 being neutral
- normal values are in the 3 to 4 range

17: rectal examination - feces

- possible values

  1 = normal

  2 = increased

  3 = decreased

4 = absent

- absent feces probably indicates an obstruction

18: abdomen

- possible values

  1 = normal

  2 = other

  3 = firm feces in the large intestine

  4 = distended small intestine

  5 = distended large intestine

- 3 is probably an obstruction caused by a mechanical impaction and is normally treated medically

- 4 and 5 indicate a surgical lesion

19: packed cell volume

- linear

- the # of red cells by volume in the blood

- normal range is 30 to 50. The level rises as the circulation becomes compromised or as the animal becomes dehydrated.

20: total protein

- linear

- normal values lie in the 6-7.5 (gms/dL) range

- the higher the value the greater the dehydration

21: abdominocentesis appearance

- a needle is put in the horse's abdomen and fluid is obtained from the abdominal cavity

- possible values:

  1 = clear

  2 = cloudy

  3 = serosanguinous

- normal fluid is clear while cloudy or serosanguinous indicates a compromised gut

22: abdomcentesis total protein

- linear
- the higher the level of protein the more likely it is to have a compromised gut. Values are in gms/dL

23: outcome

- what eventually happened to the horse?
- possible values:

  1 = lived

  2 = died

  3 = was euthanized

24: surgical lesion?

- retrospectively, was the problem (lesion) surgical?
- all cases are either operated upon or autopsied so that this value and the lesion type are always known
- possible values:

  1 = Yes

  2 = No

25, 26, 27: type of lesion

- first number is site of lesion

  1 = gastric

  2 = sm intestine

  3 = lg colon

  4 = lg colon and cecum

  5 = cecum

  6 = transverse colon

  7 = retum/descending colon

  8 = uterus

  9 = bladder

11 = all intestinal sites

00 = none

- second number is type

1 = simple

2 = strangulation

3 = inflammation

4 = other

- third number is subtype

1 = mechanical

2 = paralytic

0 = n/a

- fourth number is specific code

1 = obturation

2 = intrinsic

3 = extrinsic

4 = adynamic

5 = volvulus/torsionr

6 = intussuption

7 = thromboembolic

8 = hernia

9 = lipoma/slenic incarceration

10 = displacement

0 = n/a

28: cp_data

- is pathology data present for this case?

1 = Yes

2 = No

- this variable is of no significance since pathology data is not included or collected for these cases

Missing values

30% of the values are missing

3. Ionosphere database:

This radar data was collected by a system in Goose Bay, Labrador. This system consists of a phased array of 16 high-frequency antennas with a total transmitted power on the order of 6.4 kilowatts. The targets were free electrons in the ionosphere. "Good" radar returns are those showing evidence of some type of structure in the ionosphere. "Bad" returns are those that do not; their signals pass through the ionosphere.

Received signals were processed using an autocorrelation function whose arguments are the time of a pulse and the pulse number. There were 17 pulse numbers for the Goose Bay system. Instances in this database are described by 2 attributes per pulse number, corresponding to the complex values returned by the function resulting from the complex electromagnetic signal.

Number of Instances: 351

Number of Attributes

34 plus the class attribute

-- All 34 predictor attributes are continuous

Attribute Information:

-- All 34 are continuous, as described above

-- The 35th attribute is either "good" or "bad" according to the definition

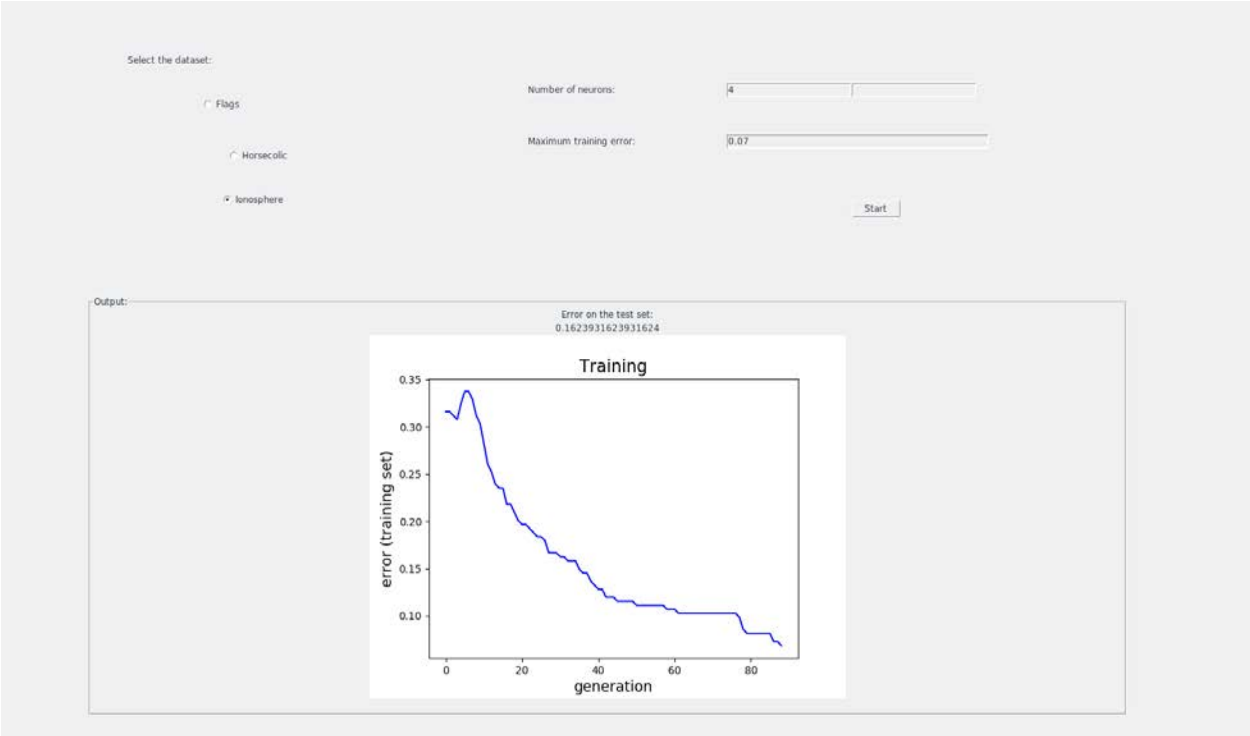
summarized above. This is a binary classification task.

Missing Values: None

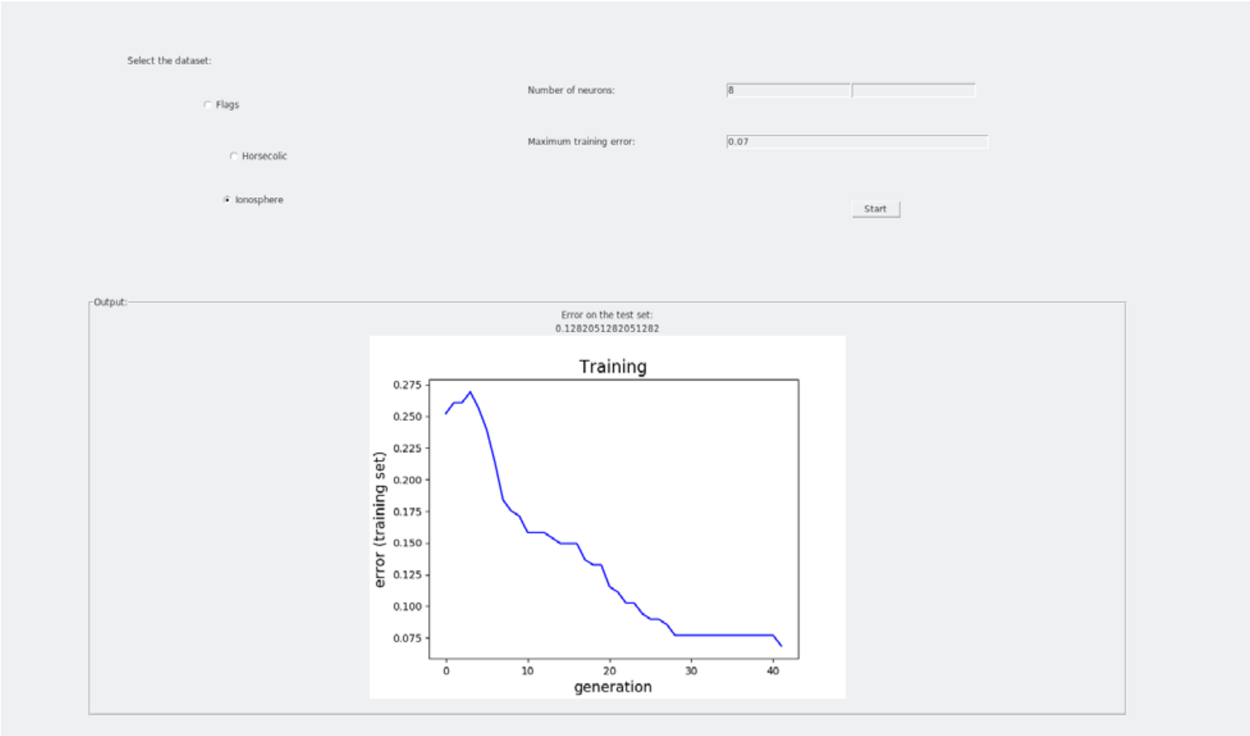
Output Results:

Ionosphere Dataset Result:

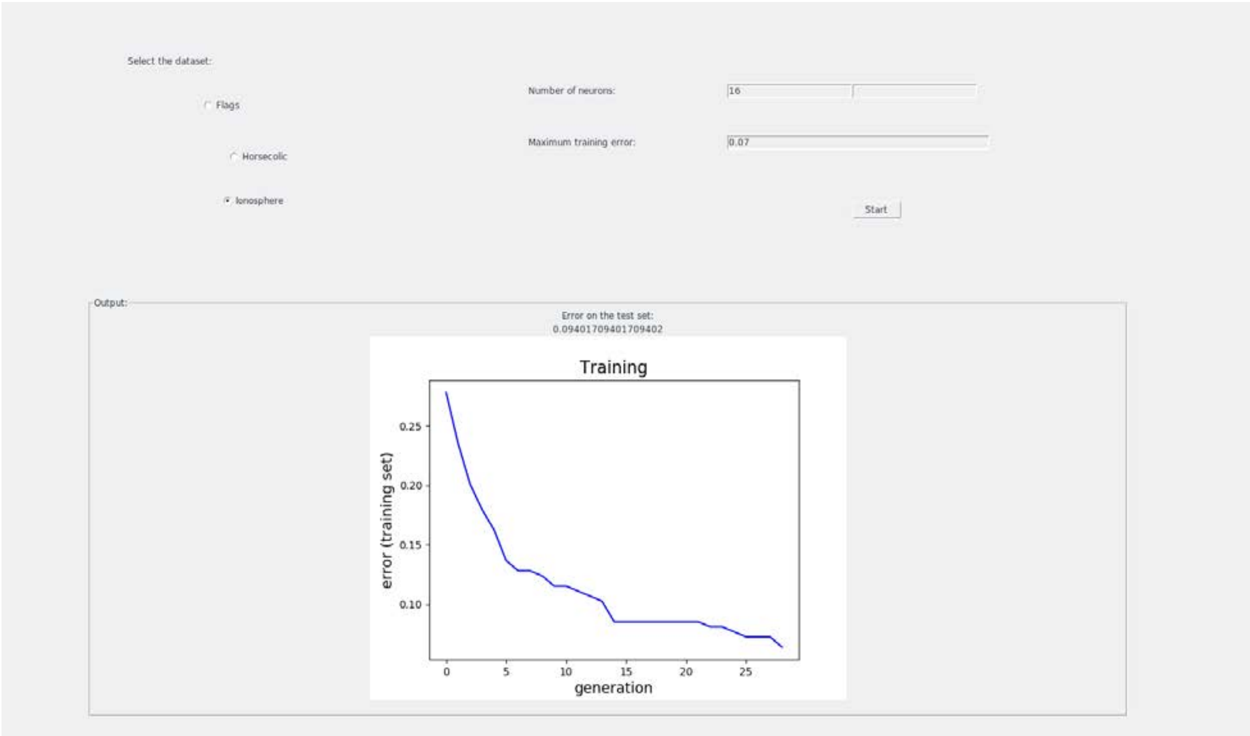
The accuracy on the test set is 84% with a single hidden layer of 4 neurons.



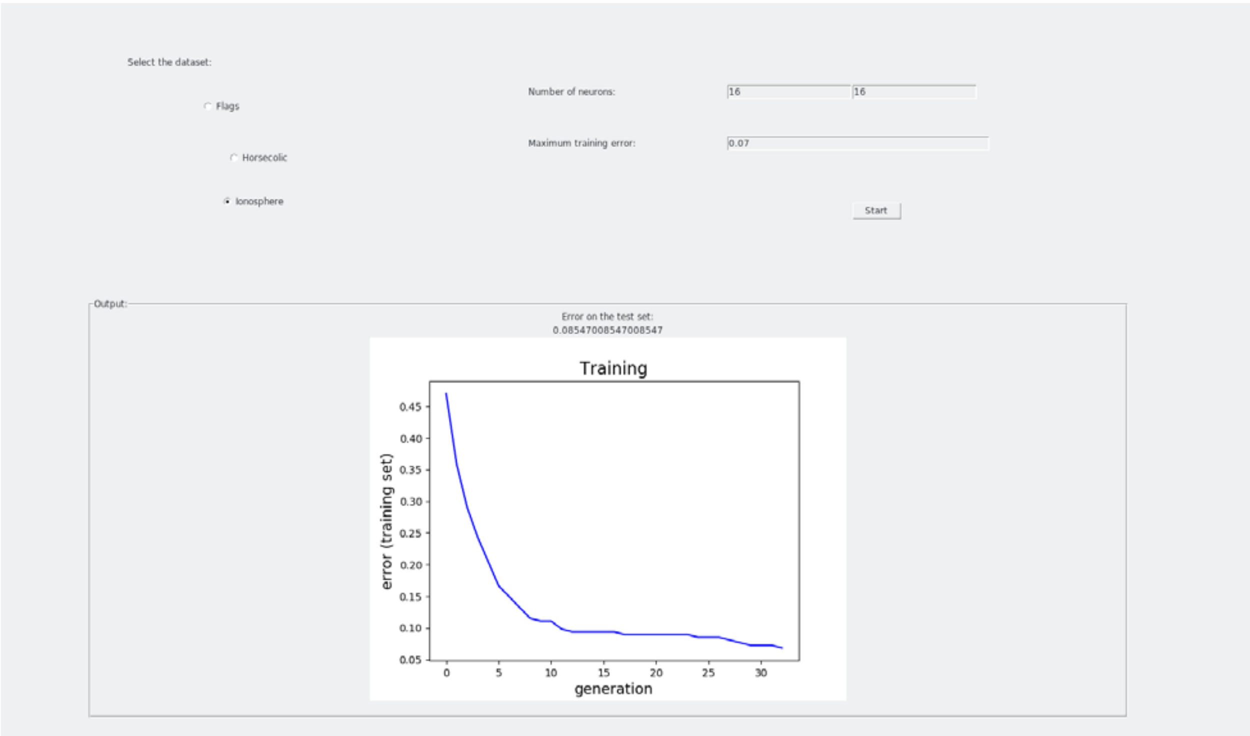
The accuracy on the test set is 87% with a larger hidden layer of 8 neurons.



The accuracy on the test set is 91% with a larger hidden layer of 16 neurons.

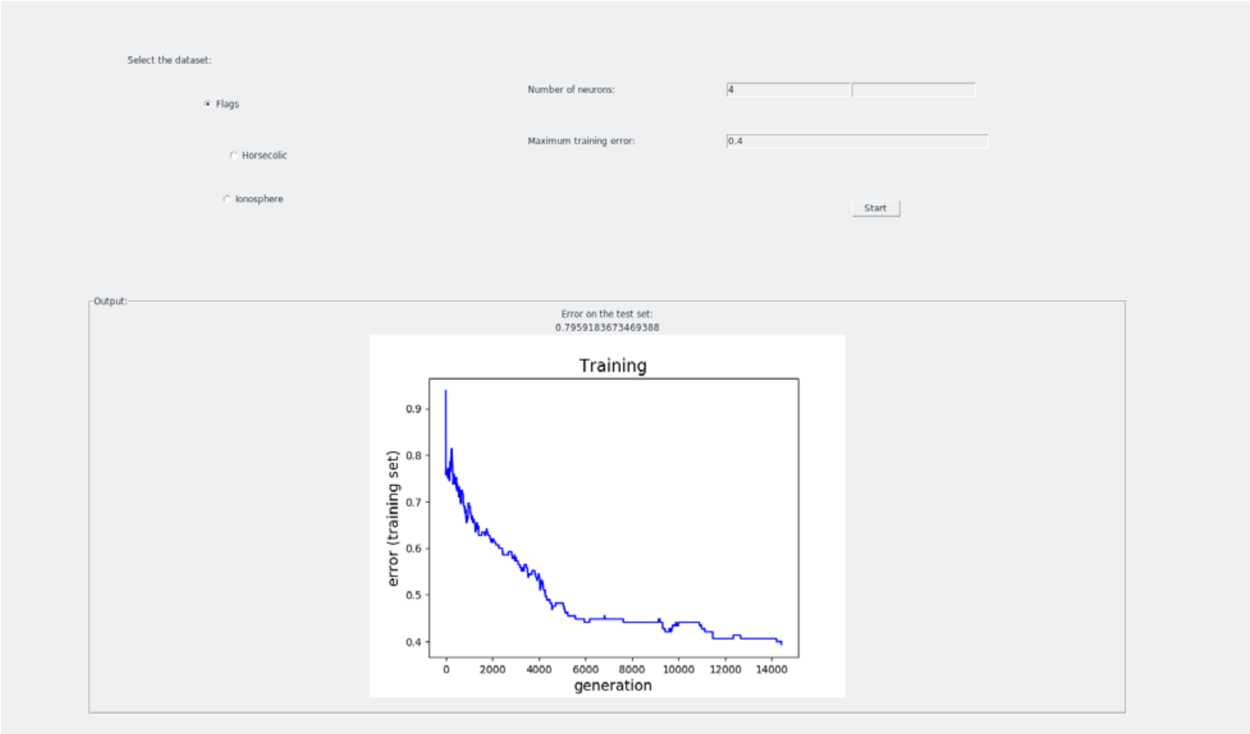


The accuracy on the test set is 91% with two hidden layers of 16 neurons.

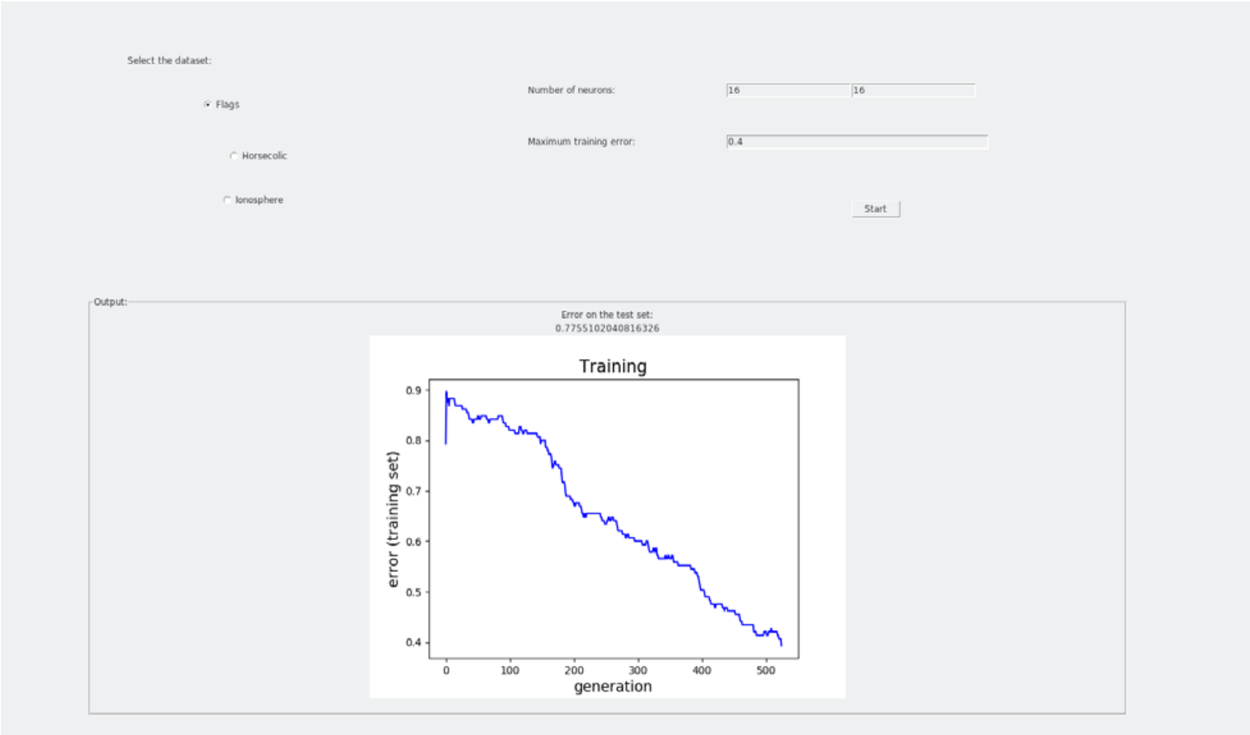


Results for flag dataset:

The accuracy on the test set is 21% with a single hidden layer of 4 neurons.

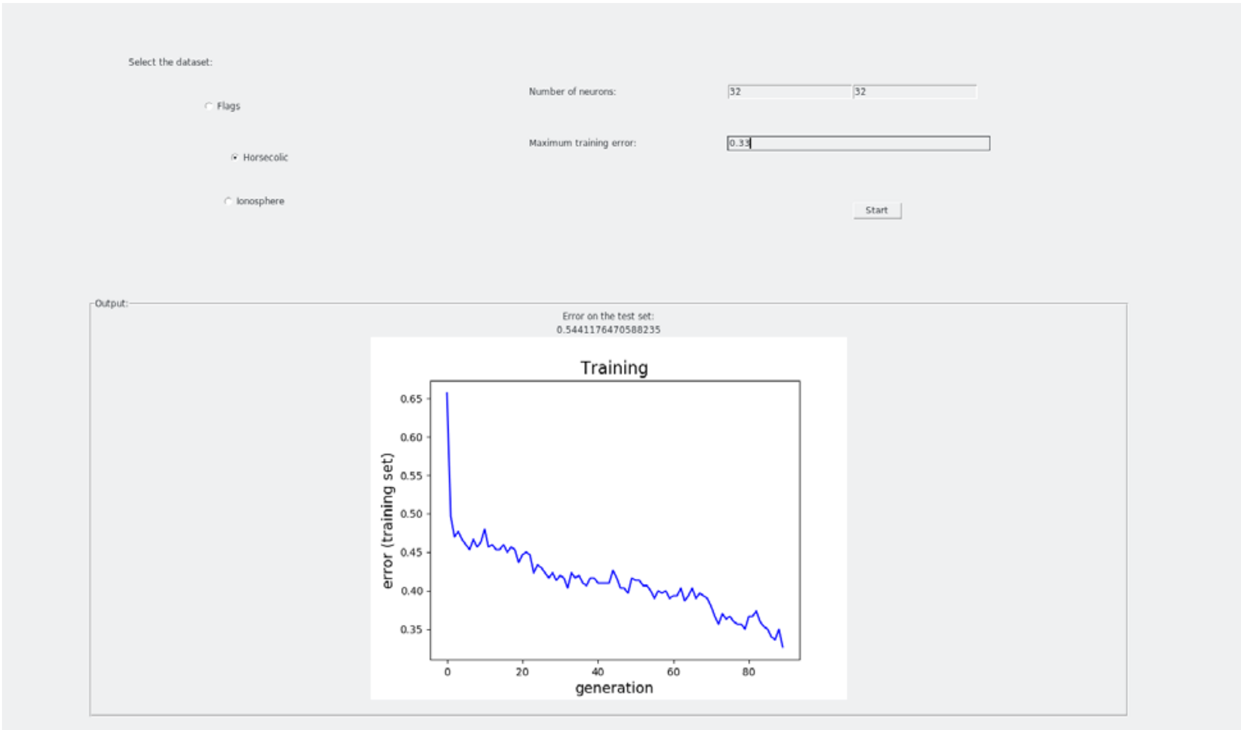


The accuracy on the test set is 23% with two hidden layers of 16 neurons.



Result for horse-colic dataset:

The accuracy on the test set is 46% with two hidden layers of 16 neurons.



CONCLUSION

Ionosphere dataset:

Increasing the number of neurons in the single hidden layer makes training achieve the desired accuracy on the testing set in fewer iterations; using two hidden layers makes training get close to the desired accuracy in fewer iterations. The achieved accuracy on the test set is close to the one reported for a multilayer perceptron in the dataset's description.

Flags dataset:

Using two hidden layers dramatically reduces the number of iterations needed to achieve the desired accuracy on the training set. However, the accuracy on the test set is disproportionately low in either case.

Horse colic dataset:

The desired accuracy on the training set was only achieved with two hidden layers. Using one layer made it impossible to achieve the desired accuracy in a reasonable number of iterations. Increasing the number of neurons in the two hidden layers had the same effect.