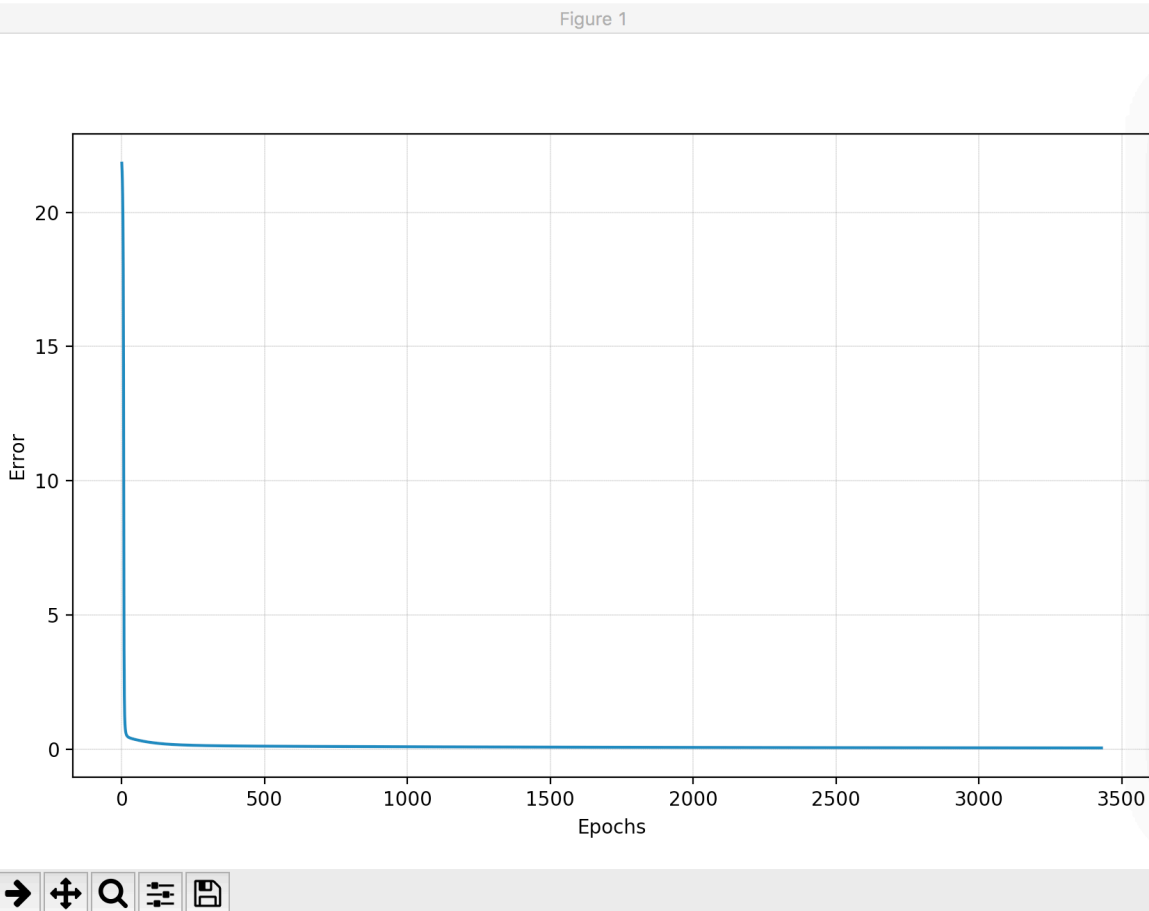


For 3 Hidden Neurons

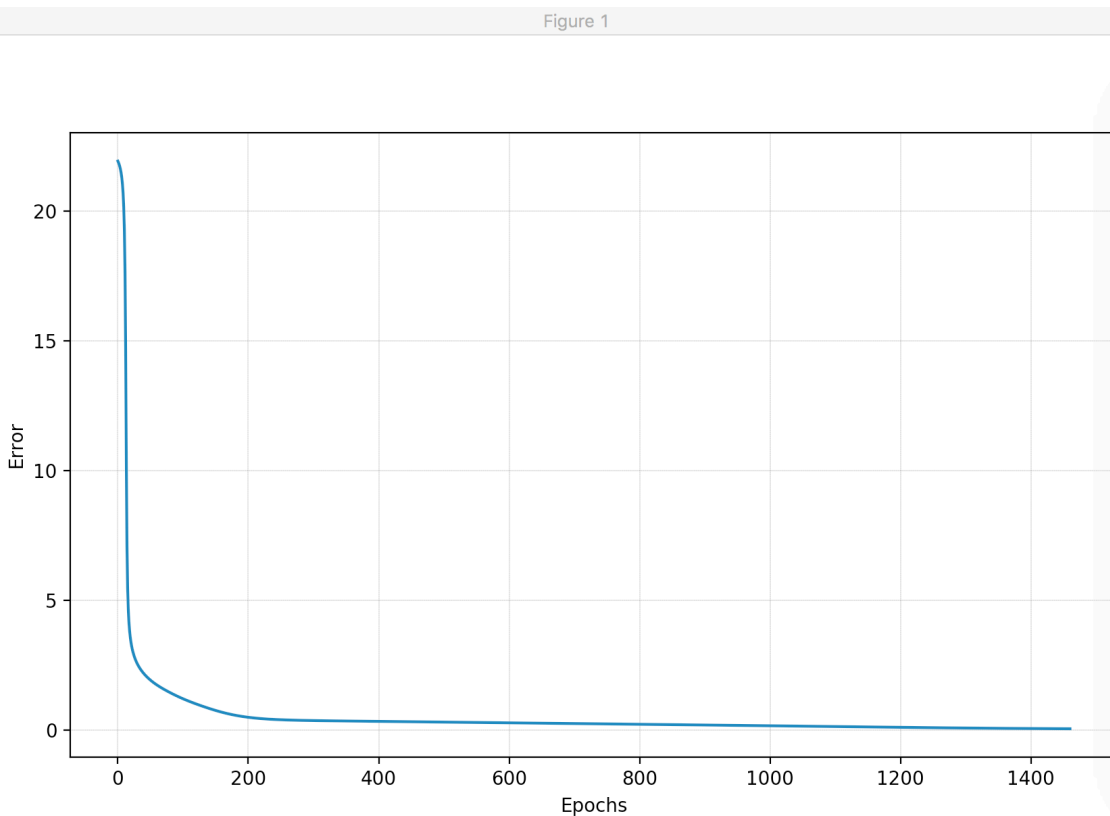
Error in testing data is 0.0515090018635



```
Back Propogration
0.0400102319445
0.0399963079738
Traceback (most recent call last):
  File "Implement_MultiNeuron.py", line 10, in program.train()
    program.train()
  File "Implement_MultiNeuron.py", line 10, in self.graph(x, values_j)
    self.graph(x, values_j)
  File "Implement_MultiNeuron.py", line 10, in eeta = self.eeta
    eeta = self.eeta
AttributeError: Neural_Network instance has no attribute 'eeta'
[Prakhar:Back Propogration prakya$ python 3.py]
Enter Eta: ^[[A
Traceback (most recent call last):
  File "Implement_MultiNeuron.py", line 10, in eeta = input("Enter Eta: ")
    eeta = input("Enter Eta: ")
  File "<string>", line 1
    ^
SyntaxError: invalid syntax
[Prakhar:Back Propogration prakya$ python 3.py]
Enter Eta: 0.01
Enter number of hidden neurons: 3
Traceback (most recent call last):
  File "Implement_MultiNeuron.py", line 10, in program.train()
    program.train()
  File "Implement_MultiNeuron.py", line 10, in self.graph(x, values_j)
    self.graph(x, values_j)
  File "Implement_MultiNeuron.py", line 10, in eeta = self.eeta
    eeta = self.eeta
AttributeError: Neural_Network instance has no attribute 'eeta'
[Prakhar:Back Propogration prakya$ python 3.py]
Enter Eta: 0.01
Enter number of hidden neurons: 3
```

For 6 Hidden Neurons

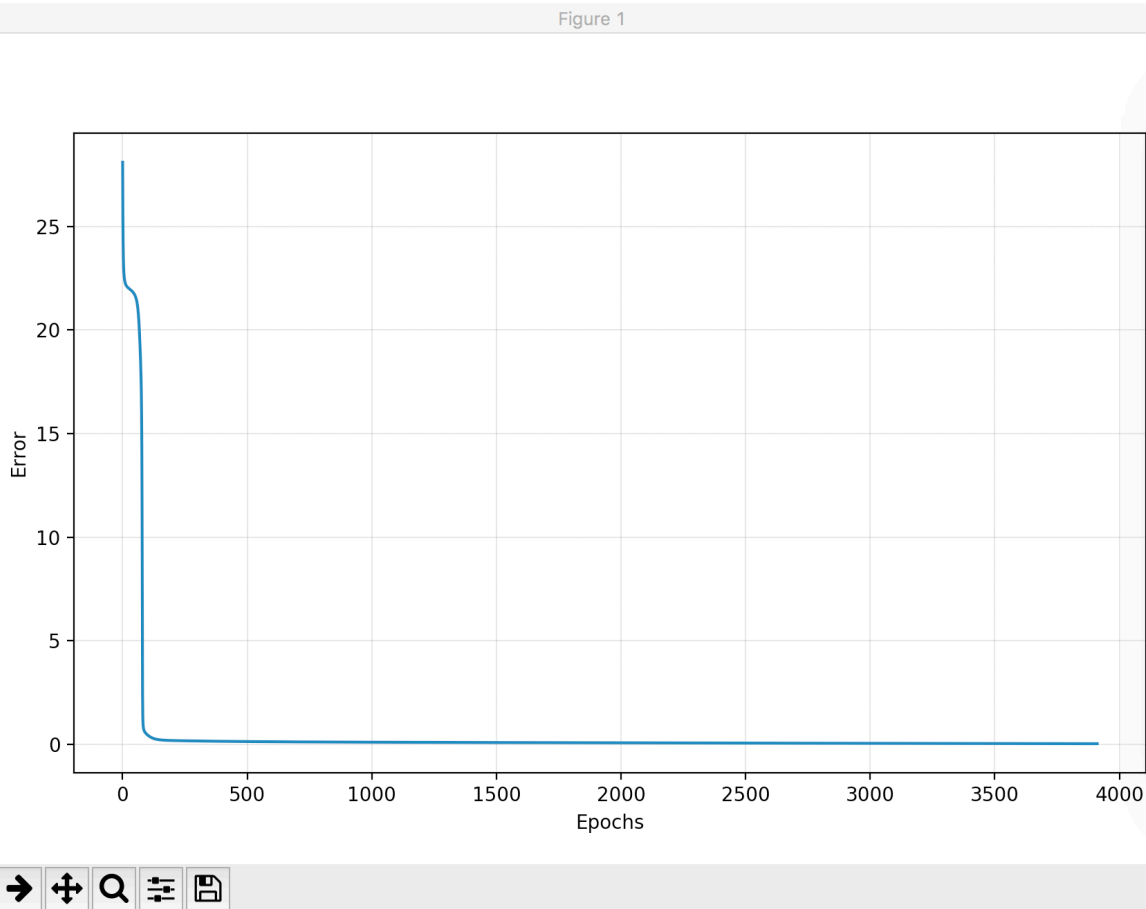
Error in testing data is 0.0620994257181



```
Back Propogration
program.train()
File "Implement_MultiNeuron.py", line 10, in self.graph(x, values_j)
    self.graph(x, values_j)
File "Implement_MultiNeuron.py", line 10, in eeta = self.eeta
    eeta = self.eeta
AttributeError: Neural_Network instance has no attribute 'eeta'
[Prakhar:Back Propogration prakya$ python 3.py]
Enter Eta: ^[[A
Traceback (most recent call last):
  File "Implement_MultiNeuron.py", line 10, in eeta = input("Enter Eta: ")
    eeta = input("Enter Eta: ")
  File "<string>", line 1
    ^
SyntaxError: invalid syntax
[Prakhar:Back Propogration prakya$ python 3.py]
Enter Eta: 0.01
Enter number of hidden neurons: 3
Traceback (most recent call last):
  File "Implement_MultiNeuron.py", line 10, in program.train()
    program.train()
  File "Implement_MultiNeuron.py", line 10, in self.graph(x, values_j)
    self.graph(x, values_j)
  File "Implement_MultiNeuron.py", line 10, in eeta = self.eeta
    eeta = self.eeta
AttributeError: Neural_Network instance has no attribute 'eeta'
[Prakhar:Back Propogration prakya$ python 3.py]
Enter Eta: 0.01
Enter number of hidden neurons: 3
Error in testing data is 0.0515090018635
[Prakhar:Back Propogration prakya$ python 3.py]
Enter Eta: 0.01
Enter number of hidden neurons: 6
$
```

For 9 Hidden Neurons

Error in testing data is 0.0521766569587



```
Back Propogration
SyntaxError: invalid syntax
Prakhar:Back Propogration prakhy$ py
Enter Eeta: 0.01
Enter number of hidden neurons: 3
Traceback (most recent call last):
  File "Implement_MultiNeuron.py", l
    program.train()
  File "Implement_MultiNeuron.py", l
    self.graph(x, values_j)
  File "Implement_MultiNeuron.py", l
    eeta = self.eeta
AttributeError: Neural_Network insta
Prakhar:Back Propogration prakhy$ py
Enter Eeta: 0.01
Enter number of hidden neurons: 3
Error in testing data is 0.051509001
Prakhar:Back Propogration prakhy$ py
Enter Eeta: 0.01
Enter number of hidden neurons: 6
$
Error in testing data is 0.062099423
Prakhar:Back Propogration prakhy$
-bash: $: command not found
Prakhar:Back Propogration prakhy$
Prakhar:Back Propogration prakhy$ py
Enter Eeta: 0.01
Enter number of hidden neurons: 9
Error in testing data is 0.065341522
Prakhar:Back Propogration prakhy$ py
Enter Eeta: 0.01
Enter number of hidden neurons: 9
```

There is no correlation between testing data error and the number of neurons. I tried running the code with 20 and 30 hidden neurons but the error remains in the range of 0.5 and 0.6

I would thus conclude that hidden neurons for such a small dataset are not able to make much of an impact how many ever of them are used. The error as well as the number of epochs to reach the same is more or less the same.

This is possibly due to overfitting, since the data is small. The model starts counting in for the noises which is why the error in the testing is becoming larger (0.07) at a very large number of neurons (ie, 100)