PROGRAM DESCRIPTION

Newton’s Method

In this exercise we will implement the gradient descent method and Newton’s method.In the first part we calculate the gradient and the Hessian on paper(see the result on the attached papers).

Then we pick an initial weigths (w0,w1) w that in this case will be:

w=[0.5,0.2]

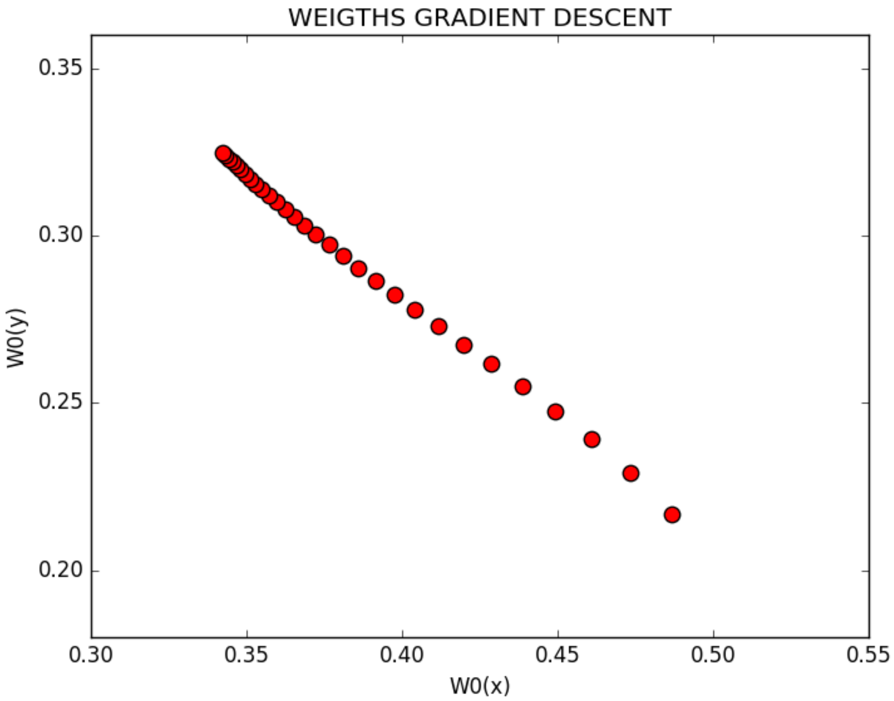
eta=0.01

In order to allow the weights to converge,we use also a threshold:

threshold=0.

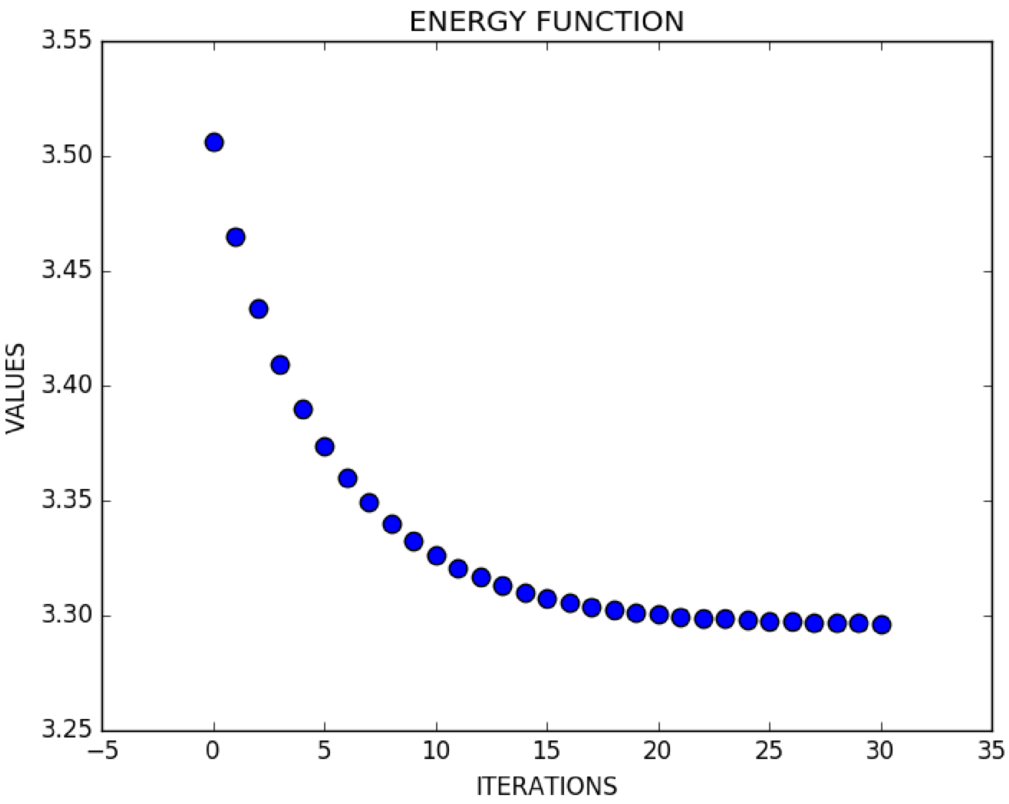
We apply the gradient descent and we obtain the following graphs:

- Weights VS Iterations



- Energy

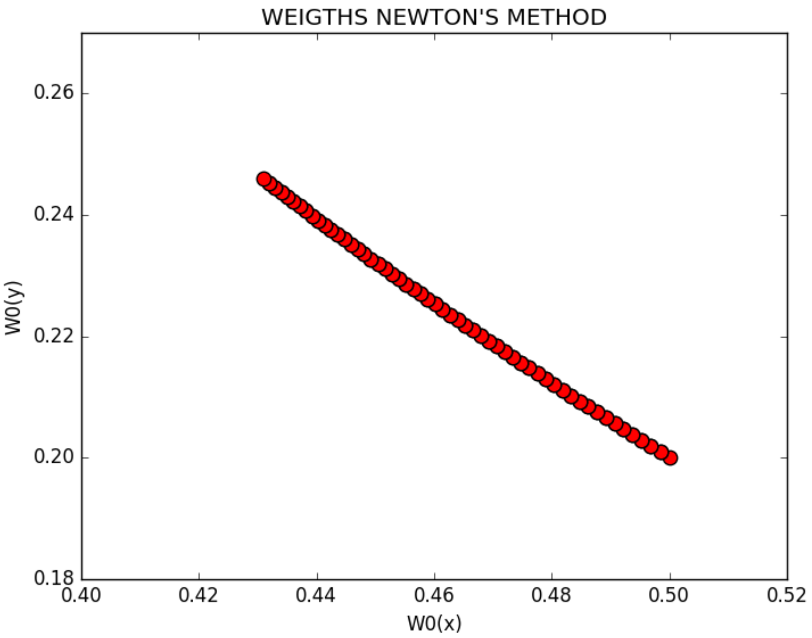
Function VS Iterations



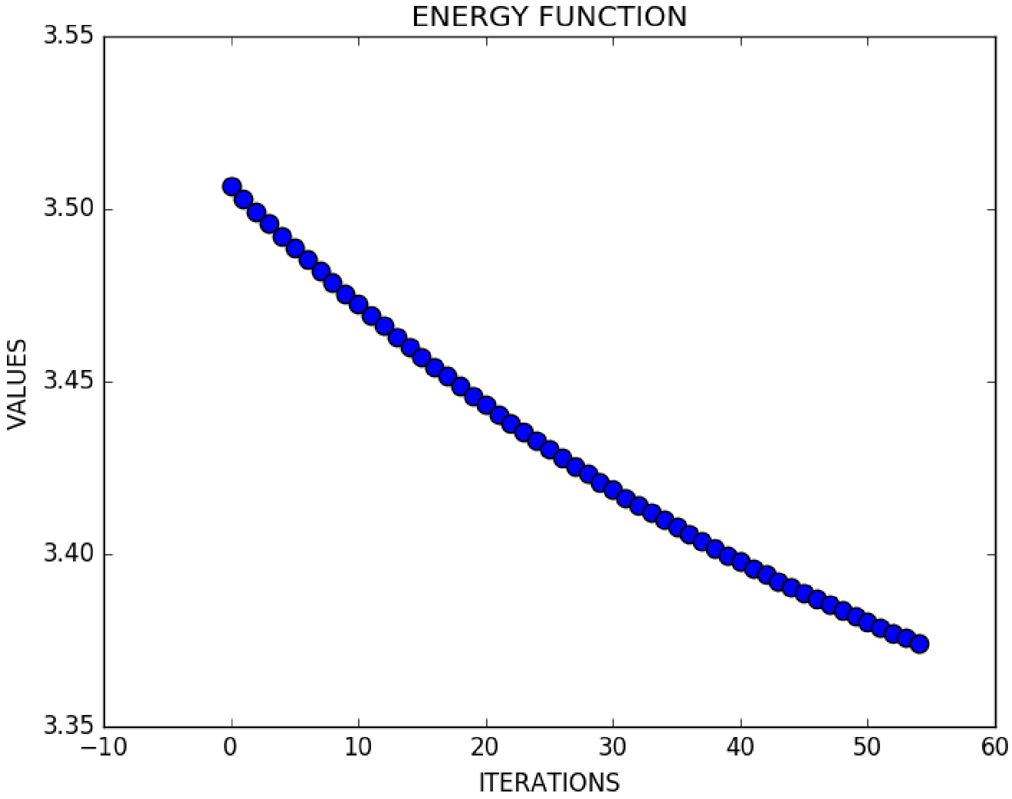
Now we repeat the same steps with Newton’s method:

We use the same w0,eta and threshold for this experiment and we obtain the following graphs:

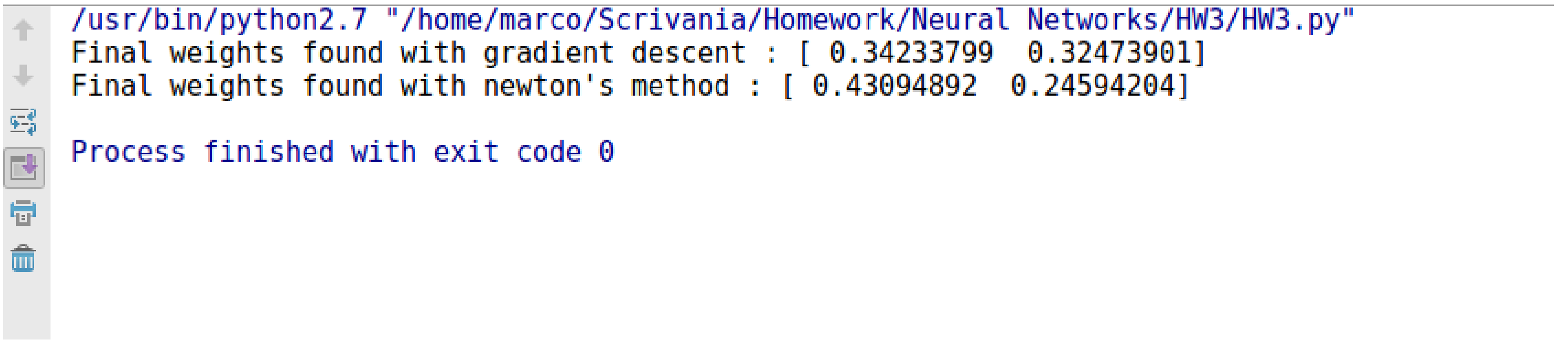
- Weights vs Iterations



- Energy function vs Iterations



The final output of the program is the following:



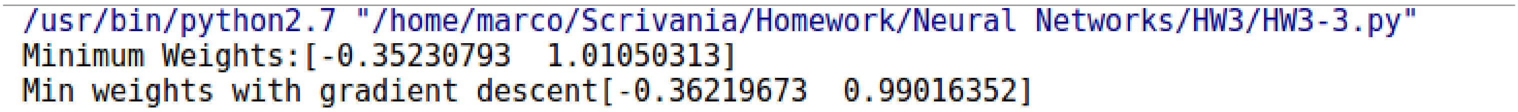
```
/usr/bin/python2.7 "/home/marco/Scrivania/Homework/Neural Networks/HW3/HW3.py"
Final weights found with gradient descent : [ 0.34233799  0.32473901]
Final weights found with newton's method : [ 0.43094892  0.24594204]

Process finished with exit code 0
```

If we compare the speed of convergence of the 2 methods we can notifce that the gradient descent is faster,in fact thw weights converge in around 30 iterations while with the Newton’s method around 55. This is not validin general but for this specific function yes.

Gradient Descent

In this exercise we generate xi = i with i = 1, . . . , 50 and yi = i + ui with i = 1, . . . , 50, where each u i should be chosen to be an arbitrary real number between −1 and 1(In this case i used a random seed(2) to generate them.

Now we find linear least squares fit to (xi , yi), i = 1, . . . , 50, and then we do the same applying gradient descent. The result found are the following:



```
/usr/bin/python2.7 "/home/marco/Scrivania/Homework/Neural Networks/HW3/HW3-3.py"
Minimum Weights:[-0.35230793  1.01050313]
Min weights with gradient descent[-0.36219673  0.99016352]
```

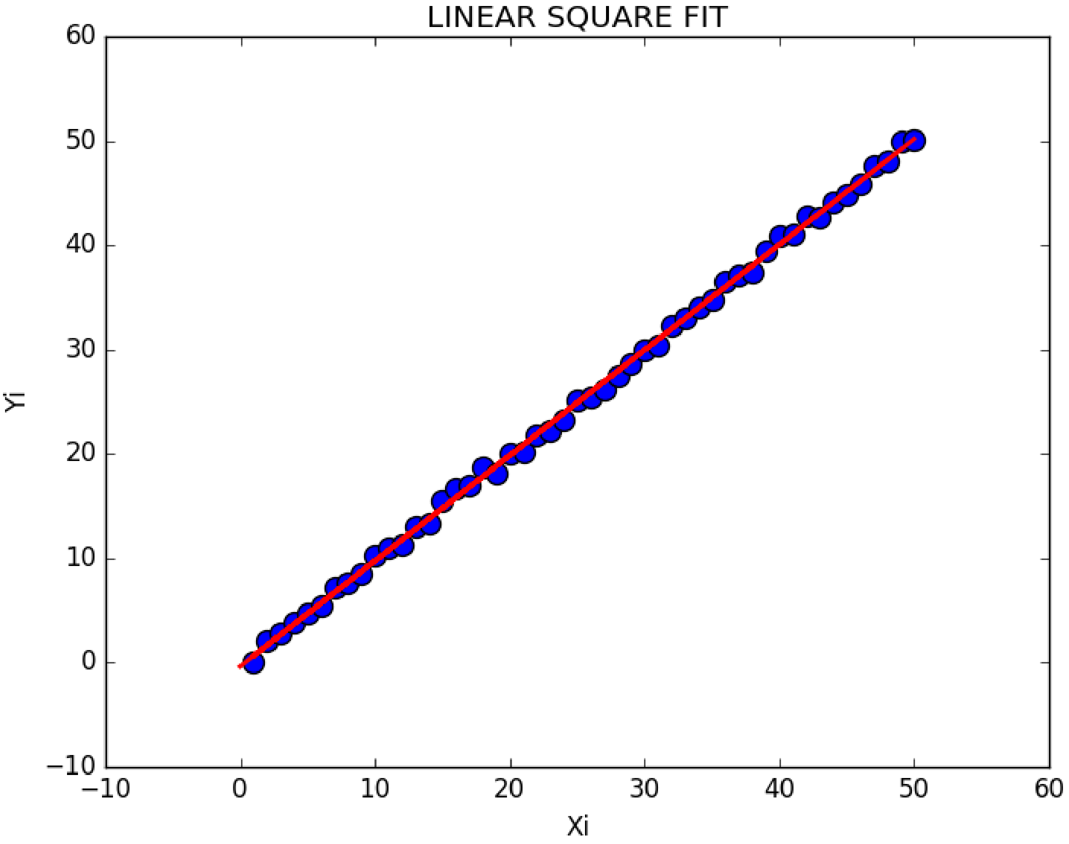
The first 2 weights are the one found using the pseudo inverse matrix, while the second using gradient descent. For the gradient descent, in order to allow it to converge we choose the following parameters:

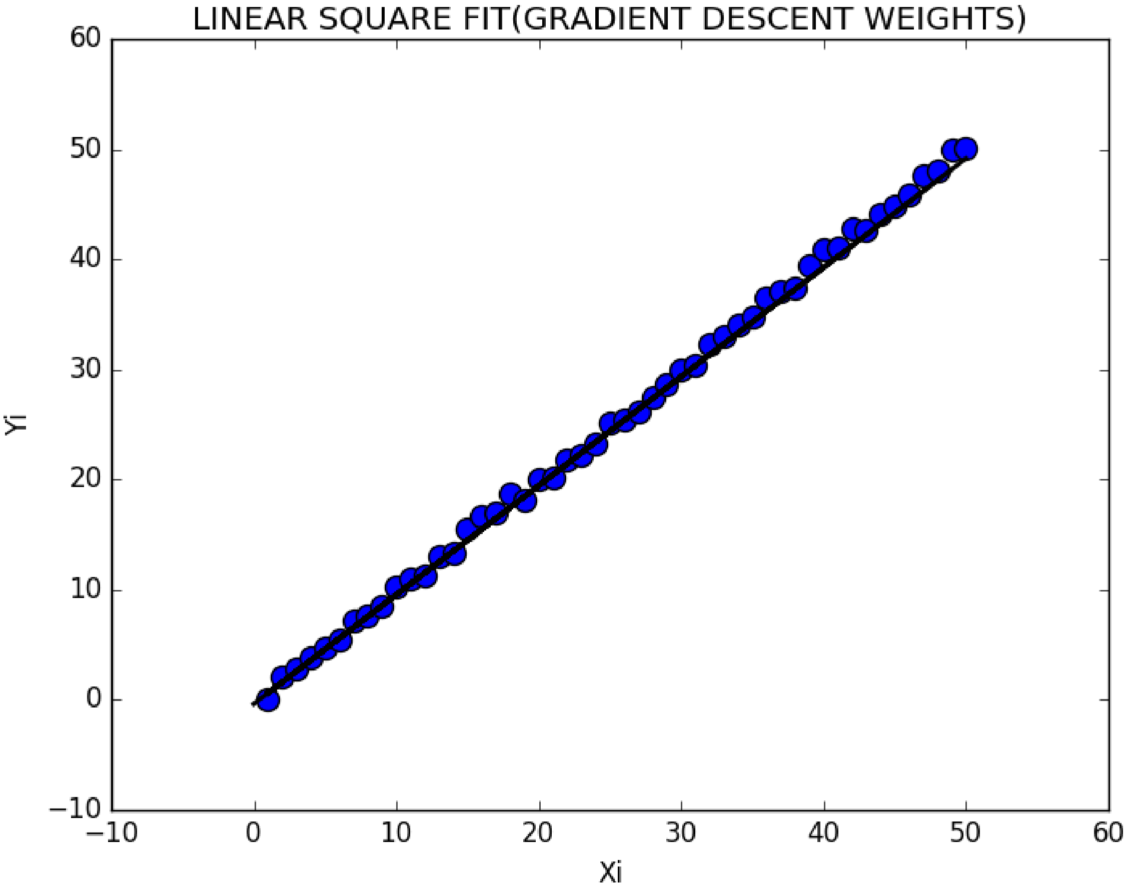
- w=[-0.4, -0.9] where -0.4 is w0 and -0.9 is w1
- eta=0,00001
- threshold=0,001

We choose a very small eta in order to allow the weights to converge because otherwise they diverge to infinity.

Comparing the weights found with the 2 methods we can see that they are almost the same.

In fact if we plot the points together linear least square fit lines, we can notice that the 2 lines are very similar and fit very good the points.


LINEAR SQUARE FIT
60
50
40
30
20
10
0
−10
−10
0
10
20
30
40
50
60
Xi
Yi


LINEAR SQUARE FIT(GRADIENT DESCENT WEIGHTS)
60
50
40
30
20
10
0
−10
−10
0
10
20
30
40
50
60
Xi
Yi