**Part 1**

**Task 1**

First, we shall write the network function explicitly:

Therefore:  
Now we can easily derive the loss.  
We shall calculate the derivative of tanh as well:

Last thing we want to define before calculating the gradients is the layers’ outputs:

w.r.t

w.r.t

w.r.t

w.r.t

w.r.t :

w.r.t

We randomly picked weights and input values for the model and calculated numerically and analytically so we can verify the above expressions.   
The results of are as follow:



We may see that the difference for each one of the 31 weights is bounded by meaning that the gradients were calculated correctly.

**Task 2:**

The function we’d like to estimate look like:  


**Task 3**

We have trained our model using BFGS, we may see that the model has converged very fast:  
Afterwards we inserted the test set to the model and plotted the results on the target function:

