**Ivancich Stefano 1227846**

**Neural Network and deep learning course 2020/21**

**Homework 1**

1. **Regression Task**

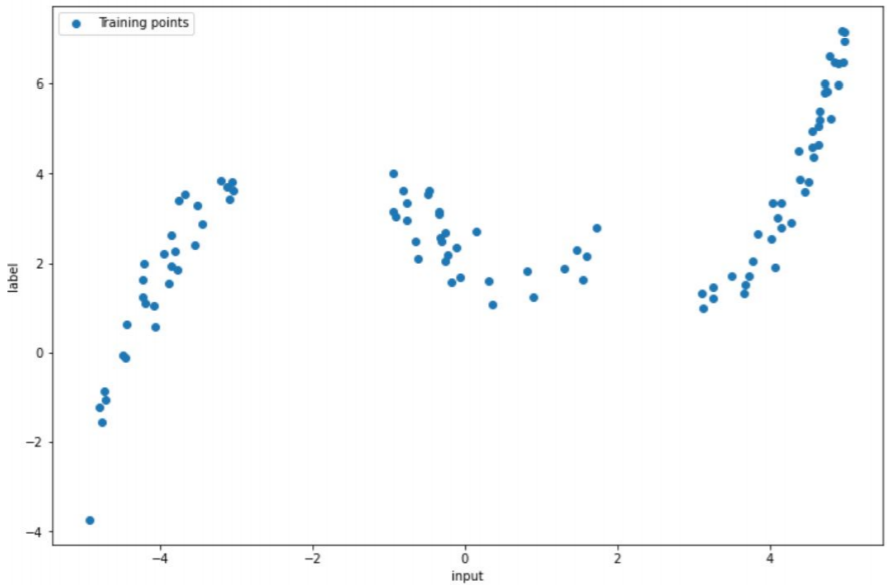
The goal is to train a neural network to approximate an unknown function:

As training point, its only given a noisy measure from the target function.

The training set its composed of only 100 points in the range of for the coordinate and for the coordinate.

This information tell us that we probably have to use cross validation because the training set is very small and that we can’t use tanh or sigmoid as activation functions in the output layer of the network because the output can be grater than 1.

This was the training set given:



We choose to use a network with 2 fully connected layers.

A grid search of the following hyperparameters with a cross fold validation of 3 folds was ran:

* **First layer number of neurons:** 8, 16, 32, 48
* **Second layer number of neurons:** 8, 16, 32, 48
* **Layers activation:** ReLu or no activation (because the output can be grater that 1 so we can’t use sigmoid in the last layer)
* **Optimizer:** Adam
* **Learning rates:** 0.1, 0.01, 0.001
* **Regularization:** "L2" with values 0, 1e-3, 1e-4, 1e-5
* **Max epochs:** 3000 (we did not choose to tune this value, because the early stopping will take care of it)
* **Early stopping:** 100 epochs without improvement

The validation errors of all training were saved, and the "best network" was chosen to be that with

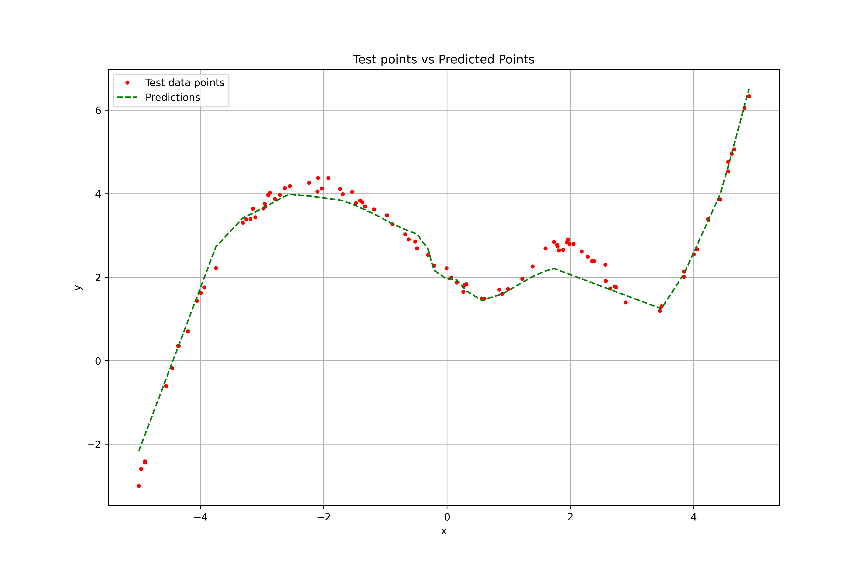
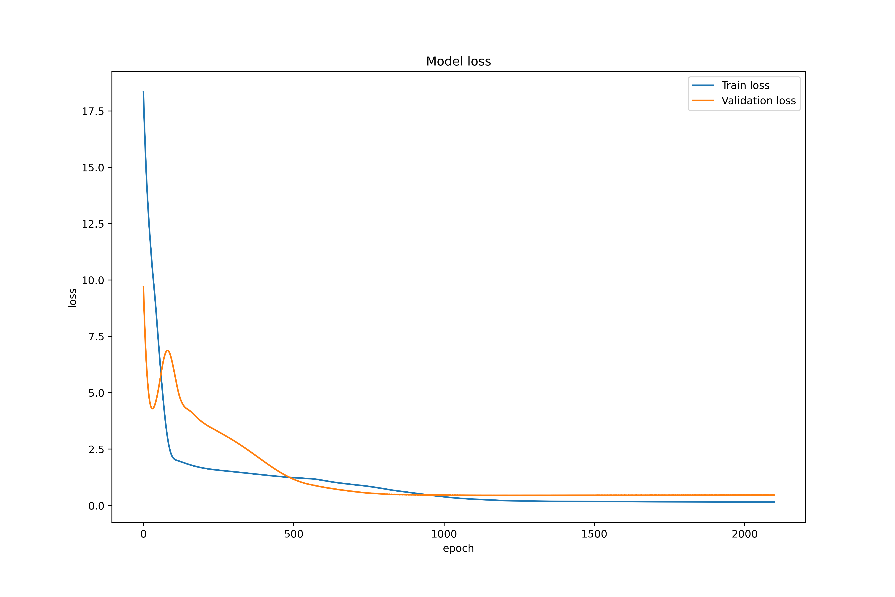
smallest average validation error.

The best hyperparameters turned out to be:

* **First layer number of neurons:** 32
* **Second layer number of neurons:** 32
* **Layers activation:** ReLu except for the last layer that has no activation
* **Optimizer:** Adam
* **Learning rates:** 0.001
* **Regularization:** 1e-5 (L2)

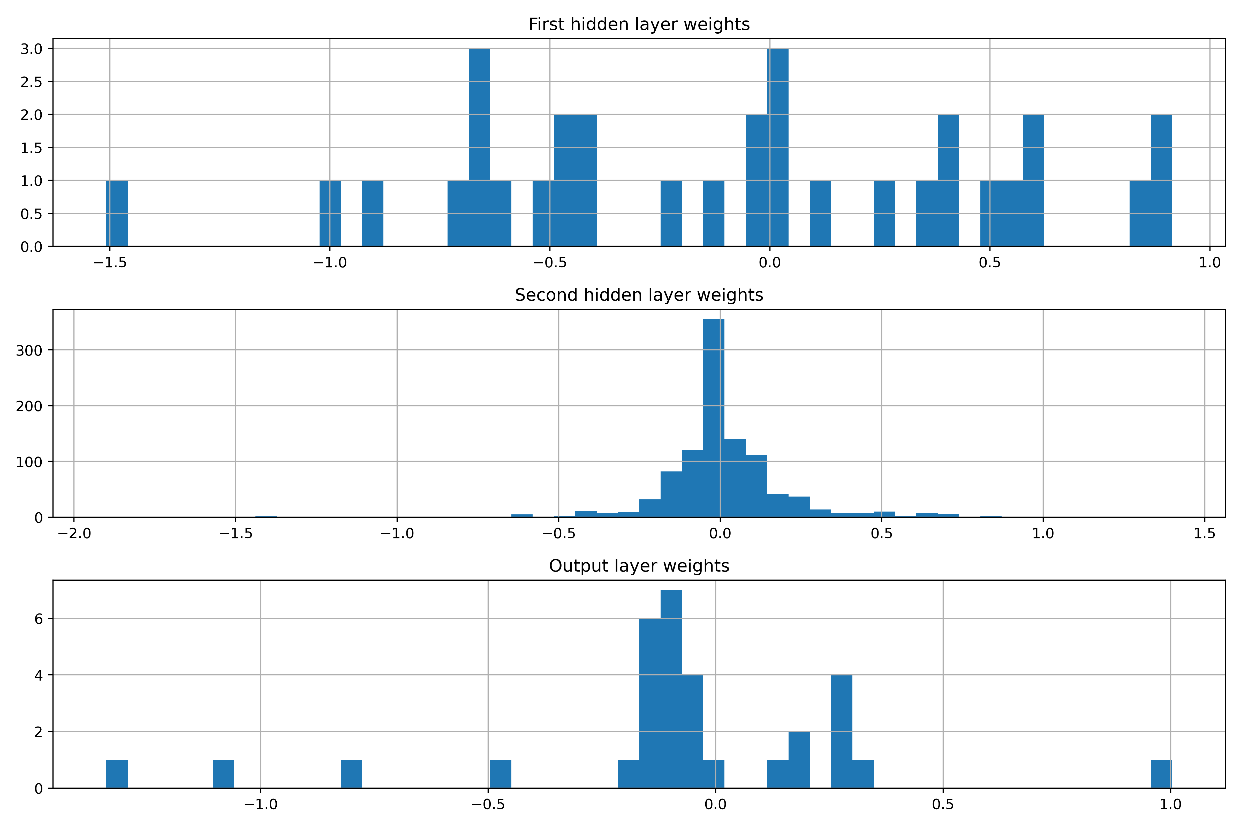
This NN was trained again using the whole train dataset with a splitting of 80-20 train-val, giving the following results:

* **Train Loss:** 0.159
* **Val Loss:** 0.457
* **Test Loss:** 0.11

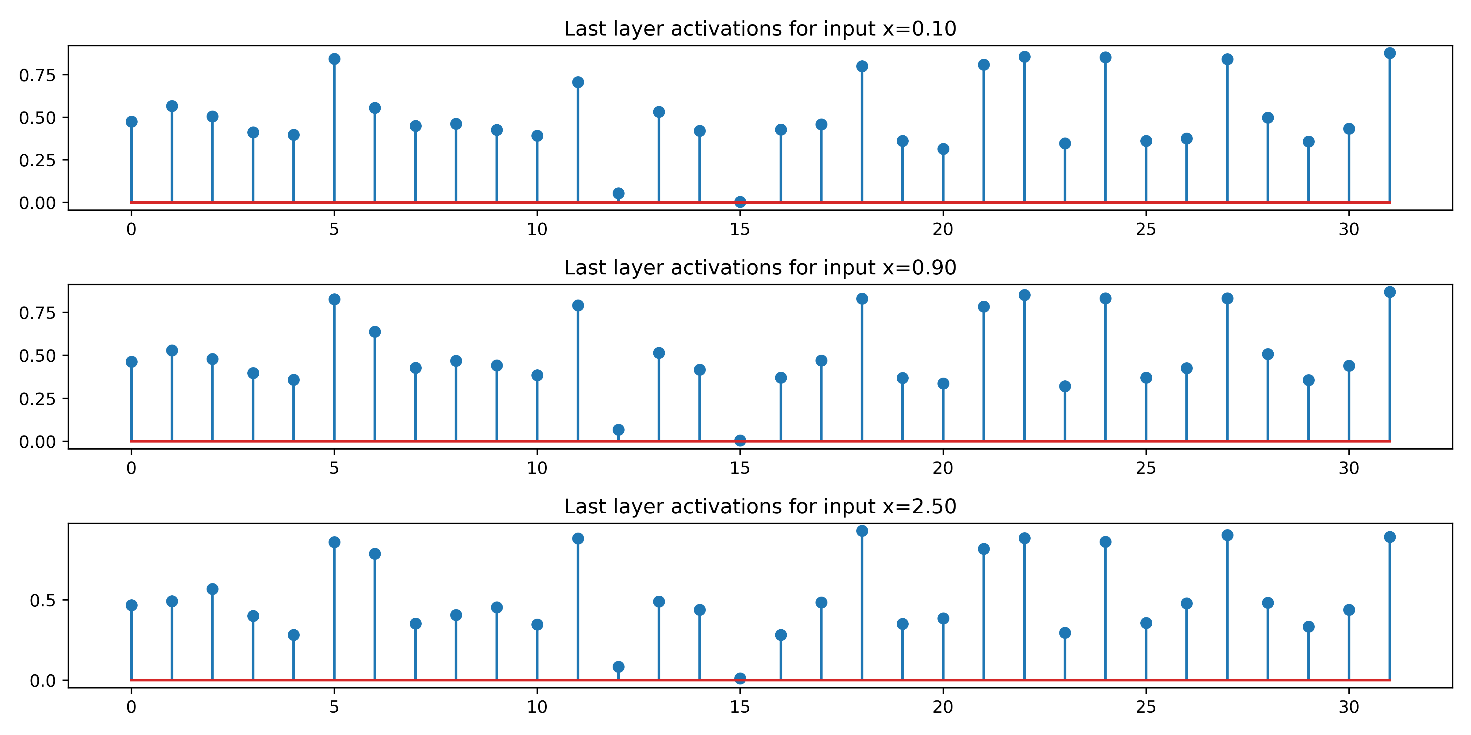


As we can see the model has problems predicting the points around and , that is because in the training set those range of points are missing, and given the fact that the Train loss and the test loss are quite near to each other, this implies that this model is generalizing the training data and probably with this kind of architecture we can’t get way better results.

The following **Weights histograms** tell us that the weights are in an acceptable range.



Finally, the **layer activations** for different parameters are telling us that all neurons are being used, so probably would be difficult to use a smaller network.



1. **Classification Task**

3 FC layers

….

CNN

…

Weights histogram

…..

Analyzing Activations

….

Receptive Fields

……