Deep Learning Lab: Report Assignment #1

Marco Latella
marco.latella@usi.ch

October 28, 2021

1 Polynomial Regression

Task 1 The assignment statement was completed by adding the i-th element of x to the power of j. In this way the rows in the dataset were initialized with: 1, x, x^2, x^3 where x is a random number that changes for each row.

Task 2 I created Validation set and Training set using the function created in Task 1.

Task 3 The resulting plot show that the data generated for the Training set and the Validation set are correct because they correctly follow the polynomial's equation.

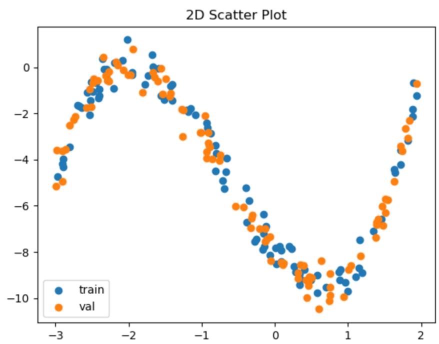


Figure 1: Training and Validation points

Task 4 Given the equation: $p(x) = x^3 + 2x^2 - 4x - 8$, the bias parameter, if set to True, allows the Neural Network to learn the known term of the equation which is -8. In this case, the bias parameter should be set to False, because the Training Dataset has been created with the known term as an input of the Neural Network and not as an additive bias.

Task 5 Adapting the linear regression, training, and Validating the Neural Network I obtained the following estimate of w*: [-7.8384943, -4.057679, 1.967556, 1.0037894].

Task 6 Running the training phase with different values of Learning Rate and number of iterations I found that the most suitable values are respectively 0.01 and 1000.

Task 7 The two losses curves are very similar and reaching 1000 steps the losses values are pretty close to zero. This means that the model can predict well both on training data and unseen data.

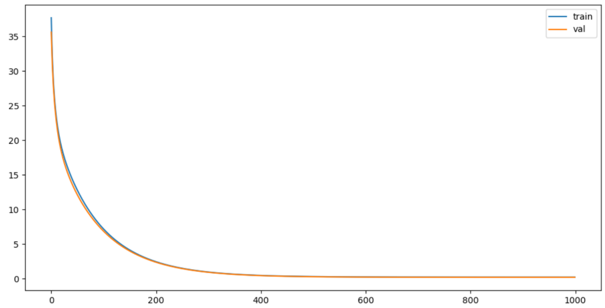


Figure 2: Training and Validation losses

Task 8 In support of the above, as we can see in the figure below, the two polynomials functions are almost overlaid, meaning that the estimated $\hat{w}$ is almost exactly w^*.

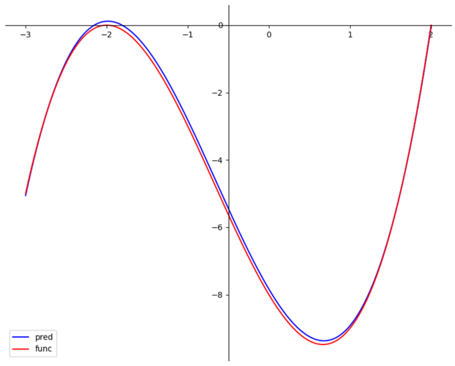
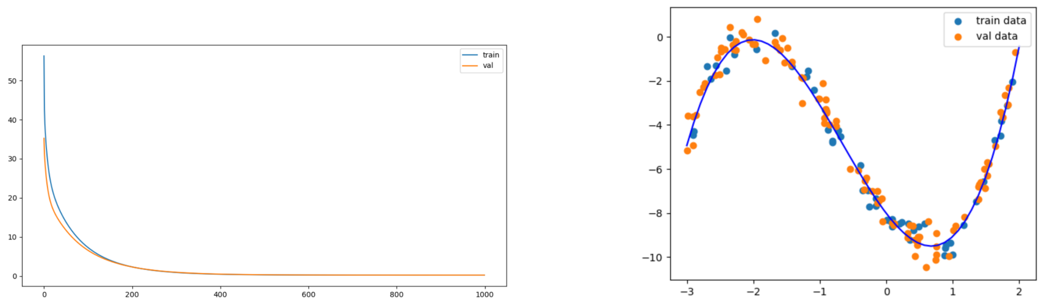


Figure 3: w^* and $\hat{w}$ polynomials

Task 9 With 50 instances in the training set, the model is still able to predict well on the training set and validation set.

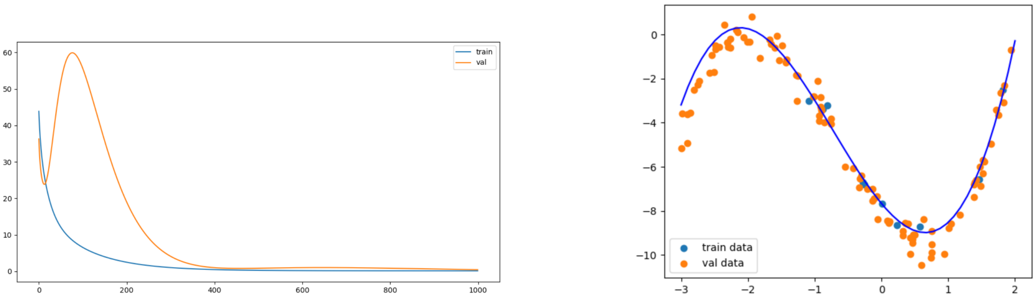


(a) Losses Curves

(b) Model

Figure 4: Model performance, Training set dim = 50

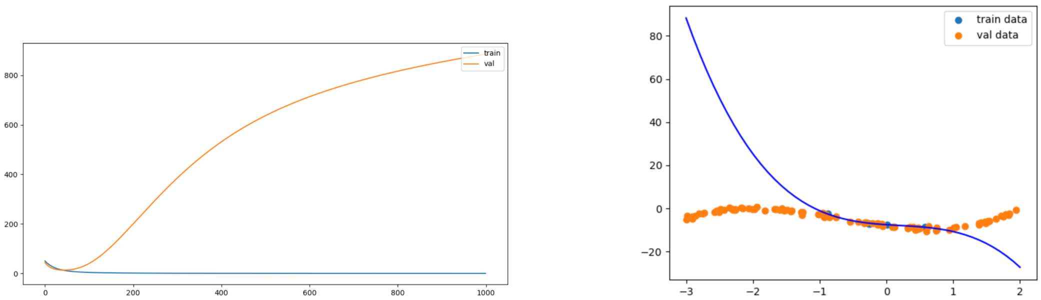
With a training set of 10 instances, we are starting to have overfitting, with a 0.1 of train loss and a 0.4 of val loss.



(a) Losses Curves (b) Model

Figure 5: Model performance, Training set dim = 10

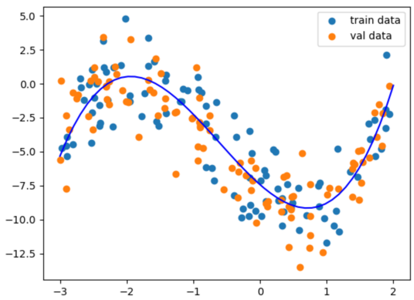
Instead, with a training set of 5 instances, we have very exaggerated overfitting. The training data are not enough to train the model.



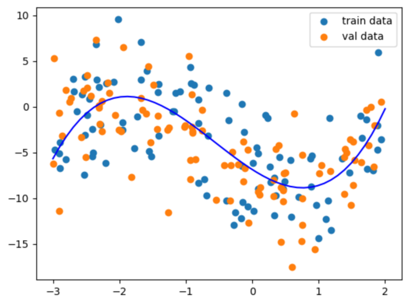
(a) Losses Curves (b) Model

Figure 6: Model performance, Training set dim = 5

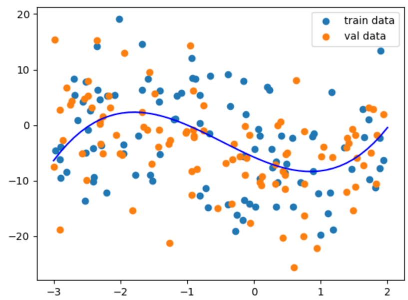
Task 10 Rising the standard deviation of the generated data, we can see that the model is too simple to represent the data, thus getting a high error on both the prediction on training data and the prediction on validation data. This lead to the underfitting.



(a) Standard deviation of 2



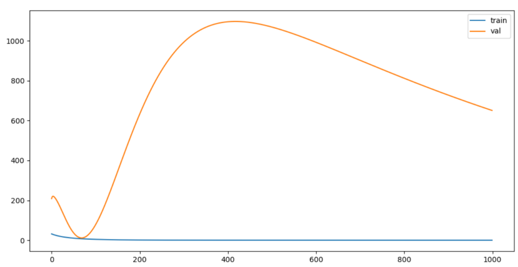
(b) Standard deviation of 4



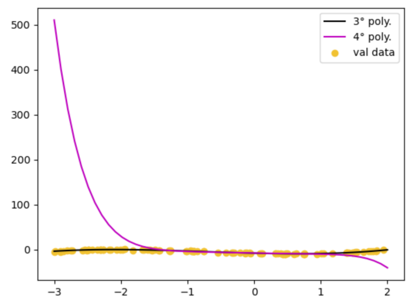
(c) Standard deviation of 8

Figure 7: Variations of data's Standard Deviation

Task 11 Fitting a polynomial of degree four on data distributed by a polynomial of degree three, we get a very high validation loss error and a low training loss error, meaning that the model is overfitting the data.



(a) Losses Curves



(b) Model comparisons

Figure 8: Model performance

2 Questions

1 - Given a function, the global minimum is the point at which the function takes its minimum value. Instead, the local minimum is the point at which the function takes the minimum value in a specific domain range.

2 - If the model is overfitting, means that it can predict well on training data, but it can't generalize with unseen data, thus it has a high loss on the validation set. If the model is underfitting, means that it can't predict well both the training data and the validation data, probably because the chosen model is too simple and can't get the relation between the input and the output variables.

3 - To avoid the overfitting problem you can simply train your model with more data, expanding the training set. Instead, to solve the underfitting problem you can rise the complexity of the model.