TD 4 Regularized Linear Regression and Bias vs Variance

Adam Goux--Gateau

31 October 2023

All the code is available here : `https://github.com/Gougaaate/Machine_Learning/tree/main/TD4`

1 Introduction

We are going to predict the amount of water flowing out a dam according to its level. The notions of bias and variance will also be studied

2 Propagation and Prediction

The first step is to plot the initial set of data :

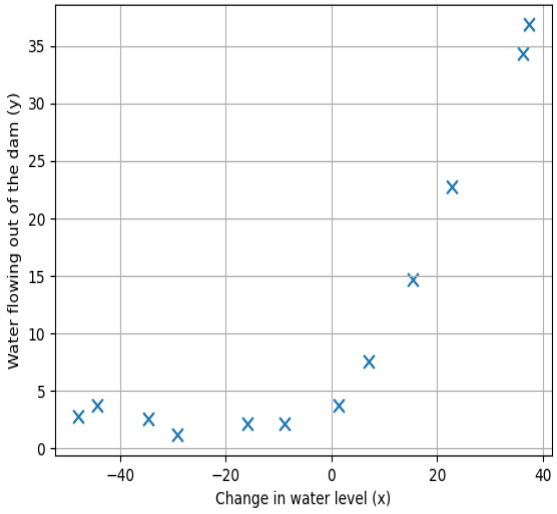

Figure 1: Water flowing according to its level

We need then to find the best model for our points. We are going to use a classial cost function :

$$J(\theta) = \frac{1}{2m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2 + \frac{\lambda}{2m}\sum_{j=1}^{n}\theta_j^2.$$

We only have to add a full column of 1 to X, and hence we can compute the gradient of this cost function :

$$\frac{\partial J(\theta)}{\partial \theta_j} = \frac{1}{m}\sum_{i=1}^{m}\left(h_\theta(x^{(i)}) - y^{(i)}\right)x_j^{(i)} (j = 0)$$
$$\frac{\partial J(\theta)}{\partial \theta_j} = \frac{1}{m}\left(\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})x_j^{(i)} + \lambda\theta_j\right) (j > 0)$$

Finally we obtain this cost, which is consistent with the value expected.
We can also find the best value of theta for our dataset by using scipy.optimize

```
Cost at theta = [1  1]: 303.993192
(this value should be about 303.993192)
```

Figure 2: Computing the cost

We can also plot the line corresponding to the best value :

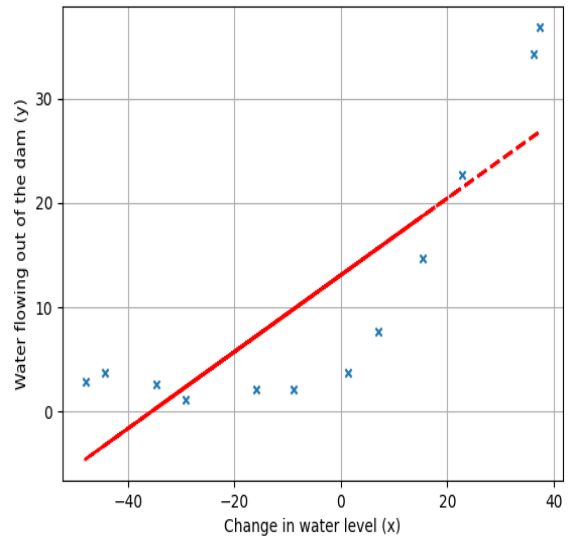

Figure 3: Water flowing according to its level

In dimension 2, we don't need a regulation parameter, hence its value is 0. The line doesn't seem to be a great model for our data set.

3 Bias and variance

We will plot learning curves that allow us to visualize training and validation errors as a function of the sample size. Remember that our database has been divided into three parts:

-The training data, which will be used to find the best parameters for our model.

-The validation data, which is used to determine the regularization parameter.

-The test data, which is used to evaluate the performance of our model.

I will then compute θ by each time adding a training data. The cost function is the same as before. We finally put the errors into vectors :

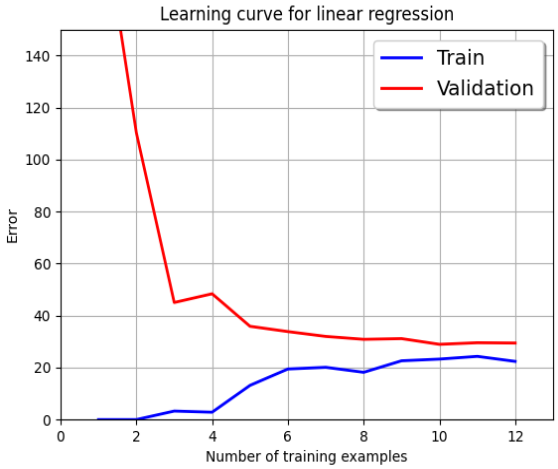

Figure 4: Learning curves

The trainer error increases as the number of training values does so. Indeed, there are only a few points at the beginning, if this number becomes too large, it is impossible to find a lite that fits perfectly. However, the validation error decreases with the number of training values : the more example we give, the more accuracy we can get on θ

4 Polynomial regression

Here, we need to change h_θ, because estimating a polynomial function by a line is really imprecise. The formula chosen is :

$$\sum_{i=0}^{p} \theta_i.(\text{waterLevel})^i$$

We chose the parameter to have reduced center ones ($\sigma = 1, m = 0$), hence the same weight is applied to all variables. With the `minimize` function of `scipy.optimize`, we obtain :

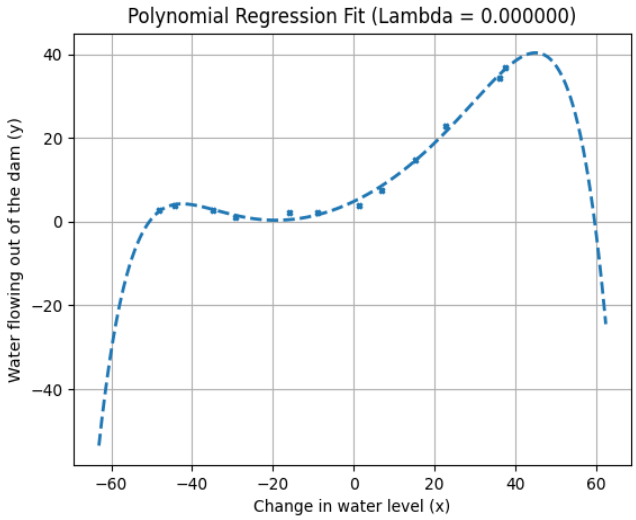

Figure 5: Polynomial model

The errors are :

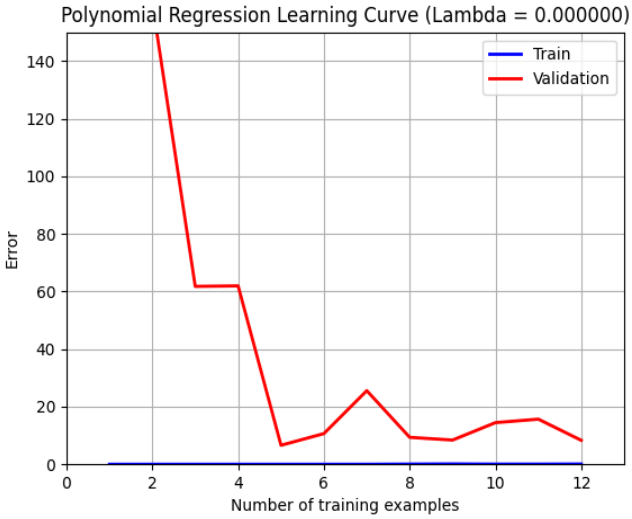

Figure 6: Errors for polynomial model

Here, we can observe that the training error is very low at each iteration. This indicates that the polynomial model is more accurate than the linear model. However, we also notice that the validation error is very high. Thus, there is indeed an overfitting issue in our model, and our variance is too high. The goal is to adjust the regularization parameter λ, which has been set to zero so far.

In the next step, we can change the parameter λ to observe its influence on the polynomial regression and the learning curves :

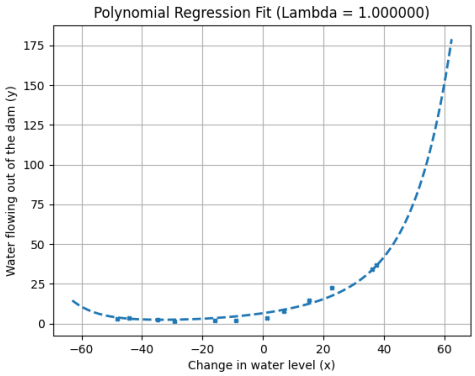

Figure 7: $\lambda = 1$

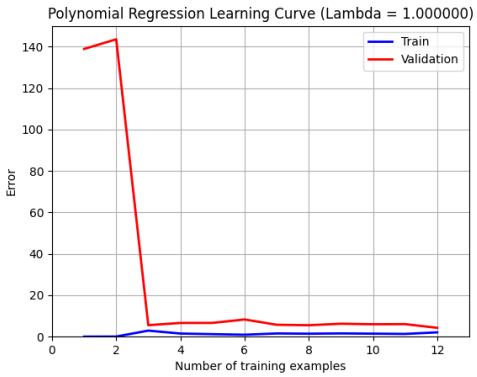

Figure 8: Error with $\lambda = 1$

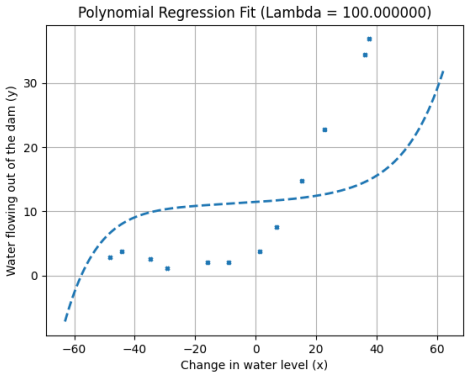

Figure 9: $\lambda = 100$

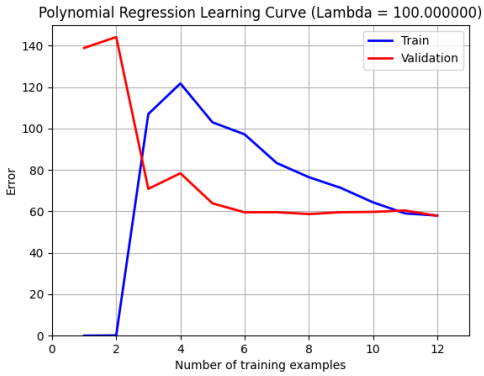

Figure 10: Error with $\lambda = 100$

With $\lambda = 1$, there is way less overfitting, and the regression curve seems to get close enough to each point. With $\lambda = 100$, thie bias is too elevated, and the data are not well modelized. It is a phenomenon of underfitting. We are now going to look for λ from the validation set.
The best regulation parameter is around 0.3.

5 More questions

5.1 Question 1

We cut the data set into 3 parts :
- Training data to find the best parameters for our model
- Validation data to find the regulation parameter
- Test data to evaluate our model

5.2 Question 2

The capacity of a prediction model corresponds to the accuracy of the relationship established between the data. When we want to perform regression, the predictive model's capacity is the accuracy of the function that connects the data together (the best function f such that $y = f(X)$). It is measured using the validation data set. In other words, the model's capacity is a measure of its ability to accurately and consistently represent the observed data in the validation set.

5.3 Question 3

The generalization error is the error of the prediction model about data that it has not viewed before. We can measure it on test data.

5.4 Question 4

The validation set is used to evaluate the performance of a model during the training phase. It enables us to adjust the regularization parameter

5.5 Question 5

The learning curve is used to evaluate a model as the size of the training set increases. It can hence detect a bias problem, and gives us data to adjust the regularization parameter.