Assignment 1

Computational Intelligence, SS2023

Team Members		
Last name	First name	Matriculation Number
Carlos Franco	Verde Arteaga	
Nedžma	Mušović	

Contents

1 Maximum Likelihood Estimation

1.1 Maximum Likelihood Model Estimation

1.1.1 Histograms

Given the three data arrays, the first task was to assume distribution based on their histograms. As visible from the following plots, the first two data samples show Gaussian distribution, whereas the third data array appears exponentially distributed.

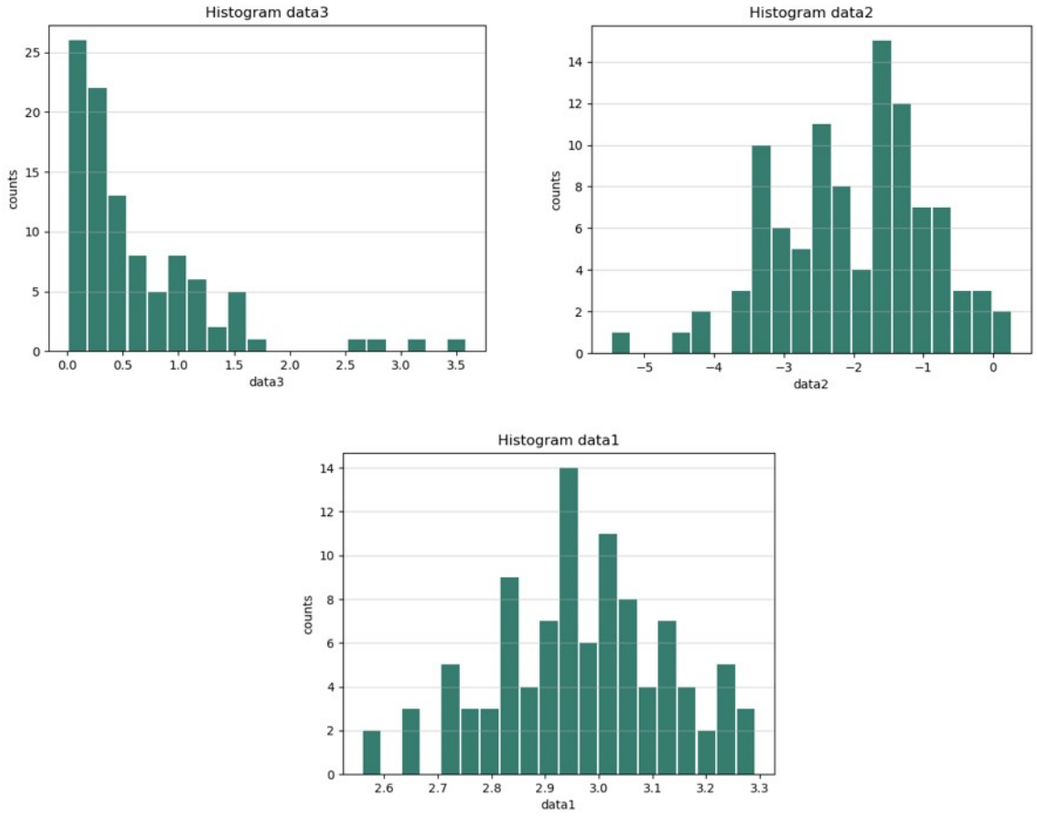

Figure 1: Histogram plots of all data sets

1.1.2 Analytical derivation of λ_{ML}

In the second step, we have analytically derived lambda in the case of the exponential distribution. As usual, the observation starts with the likelihood function:

$$P(\mathcal{X}|\theta) = \prod_{n=1}^{N} P(x_n|\theta) = \prod_{n=1}^{N} \lambda \cdot e^{-\lambda x_n}$$

After that, the log-likelihood function gets determined for the facilitated further derivation:

$$L(\mathcal{X}|\theta) = \ln P(\mathcal{X}|\theta) = \ln \prod_{n=1}^{N} P(x_n|\theta) = \ln \prod_{n=1}^{N} \lambda \cdot e^{-\lambda x_n}$$

$$L(\mathcal{X}|\theta) = \sum_{n=1}^{N} \ln \lambda \cdot e^{-\lambda x_n} = \sum_{n=1}^{N} (\ln \lambda - \lambda x_n) = N \cdot \ln \lambda - \lambda \sum_{n=1}^{N} x_n$$

To find extreme values we observe the first derivative in the next step:

$$\frac{\partial L(\mathcal{X}|\theta)}{\partial \theta} \quad \rightarrow \quad \frac{\partial L}{\partial \lambda} = \frac{N}{\lambda} - \sum_{n=1}^{N} x_n$$

In the last step, we set the first derivative to 0 and determine the extreme.

$$\frac{N}{\lambda_{ML}} - \sum_{n=1}^{N} x_n \stackrel{!}{=} 0$$

$$\lambda_{ML} = \frac{N}{\sum_{n=1}^{N} x_n}$$

Additionally, we can observe the second derivative as follows:

$$\frac{\partial^2 L}{\partial \lambda^2} = -\frac{N}{\lambda^2} < 0$$

Since the second derivative is negative for all values of λ, we can conclude the extreme to be the maxima.

1.1.3 True model parameters estimation

After deriving the parameters in exponential and Gaussian distribution, the maximum likelihood estimation was to be applied, and the parameters correspondingly estimated.

The **ML_estimation** function checks for the particular distribution (using anderson function from the scipy library) and applies derived formulas.
If the returned statistic from the anderson function is greater than the chosen significance level we reject the null hypothesis and assume data not to be normally/exponentially distributed (lower significance level chosen, to avoid false rejections).

The provided code results in the following means, variances, and lambda values, which seem to fit assumptions based on previously shown histograms.

Table 1: Estimated model parameters

$\sigma^2_{ML,1}$	$\mu_{ML,1}$	$\sigma^2_{ML,2}$	μ_{ML_2}	λ_{ML}
0.025866	2.9699	1.18167	-2.00167	1.62372

1.1.4 Probability density functions plots

After estimating parameters we plot the probability density functions by applying the extracted characteristic values on built-in python functions and obtain the following results.

The pdf for the second data set seems to fit the data points quite well, whereas that does not apply to the pdf of the first data array. The estimated mean value looks fine, but the variance does not cover all data points. However, in this data set, we observe a rather low variance, and the impact of every possible inaccuracy is significantly higher.
In the next chapter, we will observe the behavior of the numerically computed variance and see how a small correction leads to a perfectly fitting pdf.
The pdf of the exponential function shows the expected flow.

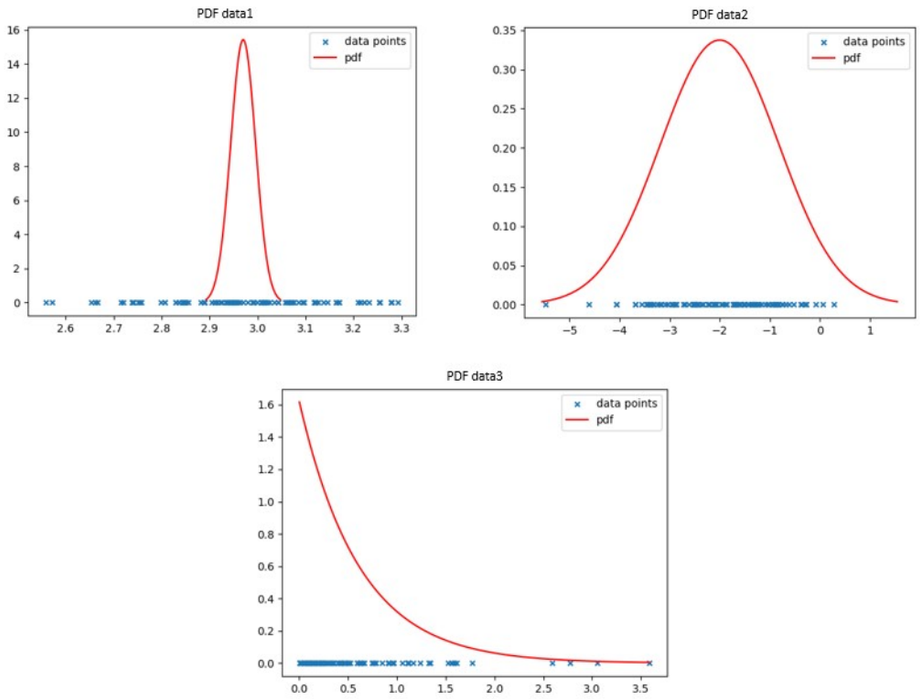

Figure 2: PDF plots of all data sets based on analytically estimated parameters

1.2 Evaluation and Visualization of the Likelihood Function

1.2.1 Gaussian

In the next step, we had to plot the 3D representation of the likelihood function for the first and second data sets showing Gaussian distribution where the x-axis represents the mean, the y-axis shows variance, whereas the likelihood applies on the z-axis.
The mean and variance ranges were defined such that the plot offers the best overview and is based on the previous parameters estimations to decrease the unnecessary computation time and increase the accuracy with a relatively low step size. The likelihood function was evaluated for every mean and variance combination respectively.
The mesh grid function from the numpy library was used to create grids for the 3D plot, with the following output:

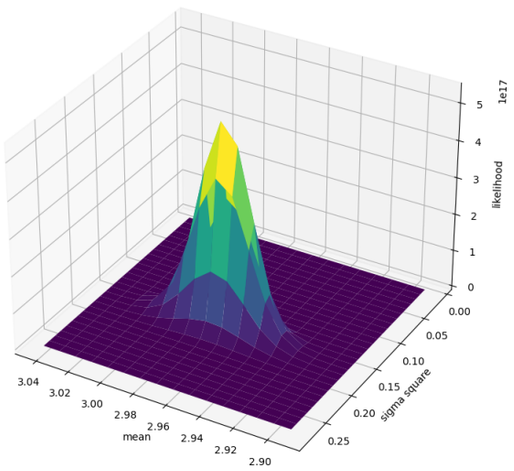

Figure 3: 3D plot of the likelihood function - data1

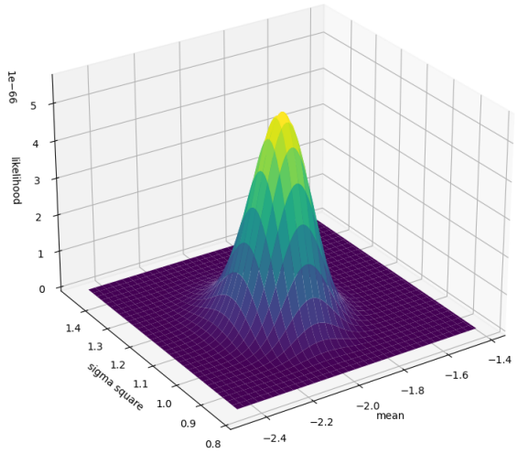

Figure 4: 3D plot of the likelihood function - data2

The given plots show expected behavior with the respect to estimated mean and variance values, as we can observe the maximal likelihood in the very range of the estimated parameters.

1.2.2 Exponential

The same applies to the 2d likelihood plot of the data set showing exponential distribution, whereas the x-axis represents lambda and y-axis likelihood values. The likelihood function gets evaluated for the defined range of λ and the following plot clearly shows that the likelihood function reaches its maximum at the estimated lambda value.

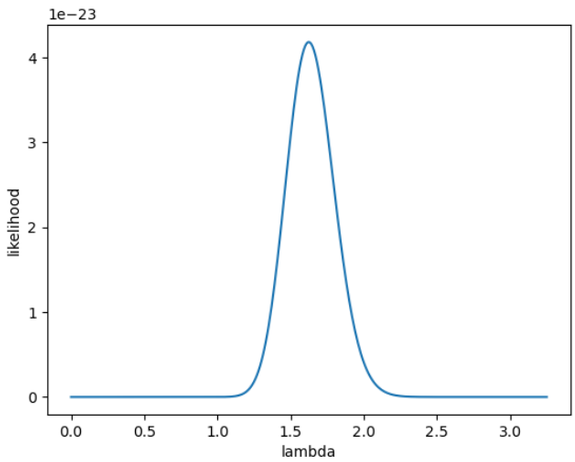

Figure 5: Likelihood function for the exponentially distributed data set

1.2.3 Numerical model parameters

For the numerical computation of the parameters of interest, the likelihood function was evaluated for various means and variances (considering a higher range here) such that we can explicitly look for a maximum using the argmax numpy function. The computed parameters match well the estimated values, except for the variance of the first data set, where the numerical approach seems to deliver better results, as shown in the corrected pdf.

Table 2: Numerically calculated model parameters

$\mu^2_{num,1}$	$\sigma_{num,1}$	$\sigma^2_{ML,2}$	μ_{ML_2}	λ_{ML}
2.9699	0.16	-1.996	1.09	1.62

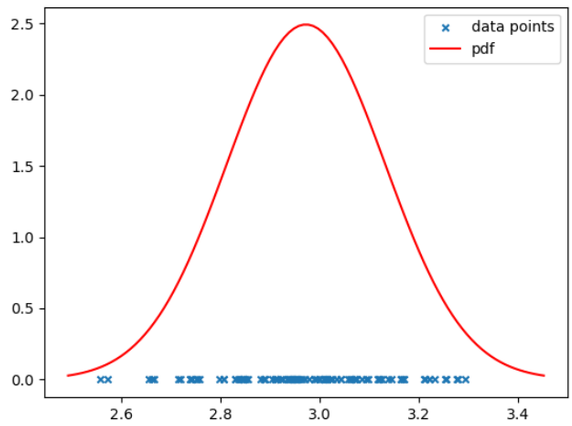

Figure 6: PDF for the second data set after variance correction

1.3 Bayesian Model Estimation

In the first step, we computed the posterior mean and the variance of the mean distribution for the first and second data arrays using formulas from the assignment sheet and derived the following results:

Table 3: Parameters of the posterior distribution

$\mu_{N,1}$	$\sigma^2_{N,1}$	$\mu_{N,2}$	$\sigma^2_{N,2}$
2.96595	0.001605	-1.97476	0.0107

As shown in the following figure, the posterior distributions are compared to the prior distributions differently shaped and highly concentrated which implies informative data observed estimations which updated prior beliefs (prior distribution) significantly and reduced the prior distribution uncertainty.

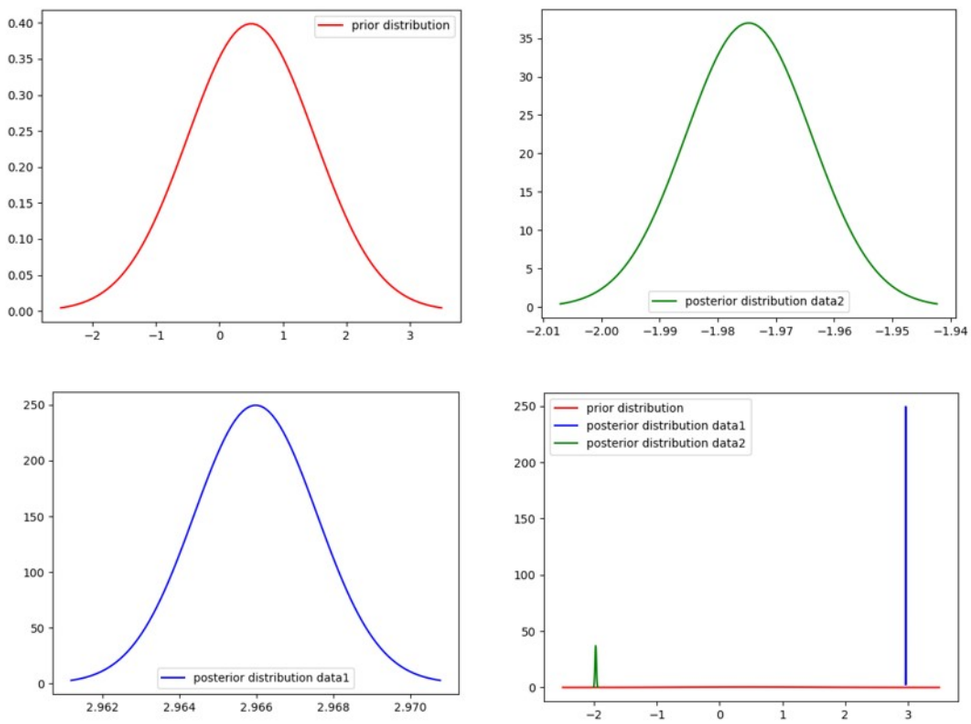

Figure 7: Posterior distribution as the main form of support

2 Linear Regression

2.1 Linear Regression with Regularization

$$\frac{\partial}{\partial w_d}\overrightarrow{E}(w) = \frac{\partial}{\partial w_d}\Big[E(w) + \frac{\lambda}{2}\cdot||w||^2\Big] = \frac{\partial}{\partial w_d}E(w) + \frac{\partial}{\partial w_d}\Big[\frac{\lambda}{2}\cdot||w||^2\Big]$$

$$\tfrac{\partial}{\partial w_d}E(w) = \tfrac{\partial}{\partial w_d}\Big[\tfrac{1}{2}\cdot||t-\phi w^2||\Big] \overset{\mathrm{z=t-\phi w}}{=} \tfrac{\partial}{\partial w_d}\Big[\tfrac{1}{2}||z||^2\Big] \overset{||\mathrm{z}||^2=\mathrm{z^T z}}{=} \tfrac{\partial}{\partial w_d}\Big[\tfrac{1}{2}\cdot z^T z\Big] =$$

$$\tfrac{\partial}{\partial w_d}\Big[\tfrac{1}{2}(t-\phi w)^T\cdot(t-\phi \mathrm{w})\Big] = \tfrac{\partial}{\partial w_d}\Big[\tfrac{1}{2}(t^Tt - \phi t^T w - \phi^T t w^T + \phi^T\phi w^T w\Big] \overset{||w||^2=w^{\mathrm{T}}w}{=} \tag{1}$$

$$\tfrac{\partial}{\partial w_d}\Big[\tfrac{1}{2}(t^Tt - \phi t^T w - \phi^T t w^T + \phi^T\phi||w||^2)\Big]$$

$$\frac{\partial}{\partial w_d}\Big[E(w) + \frac{\lambda}{2}\cdot||w||^2\Big] = \frac{\partial}{\partial w_d}\Big[\ \frac{1}{2}(t^Tt - \phi t^T w - \phi^T t w^T + \phi^T\phi||w||^2 + \lambda||w||^2)\ \Big] \tag{2}$$

$$\frac{\partial}{\partial w_d}\Big[\ t^Tt\ \Big] = 0 \tag{3}$$

$$\frac{\partial}{\partial w_d}\Big[\ \phi^T\phi||w||^2\ \Big] = 2\phi^T\phi||w|| \tag{4}$$

$$\frac{\partial}{\partial w_d}\Big[\ \lambda||w||^2\ \Big] = 2\lambda||w|| \tag{5}$$

$$\tfrac{\partial}{\partial w_d}\Big[\ -\phi t^T w - \phi^T t w^T\ \Big] = \quad \Big[\phi w t^T = (\phi^T w^T t)^T = \phi^T w t^T\ ;\ \phi w t^T = (\phi w t^T)^T\Big] \tag{6}$$

$$= \tfrac{\partial}{\partial w_d}\Big[\ -\phi w t^T - \phi^T t w^T\ \Big] = \tfrac{\partial}{\partial w_d}\Big[\ -\phi w t^T - \phi w t^T\ \Big] = \tfrac{\partial}{\partial w_d}\Big[\ -2\phi w t^T\ \Big] = -2\phi t^T$$

$$\tfrac{\partial}{\partial w_d}(\overrightarrow{E}(w)) \overset{!}{=} 0$$

$$\tfrac{1}{2}\Big[0 - 2\phi t^T + 2\phi^T\phi||w|| + 2\lambda||w||\Big] \overset{!}{=} 0 \tag{7}$$

$$-\phi t^T + \phi^T\phi||w|| + \lambda||w|| \overset{!}{=} 0$$

$$||w||(\phi^T\phi + \lambda I) = \phi t^T \quad \rightarrow \quad ||w|| = \tfrac{\phi t^T}{\phi^T\phi + \lambda I} = (\phi^T\phi + \lambda I)^{-1}\phi t^T$$

2.2 Linear Regression with Polynomial Features

2.2.1 Polynomial model functions

We show in the pictures Figure 8 and 9 the result of the degree D = 1,2,9 and 16. For doing this step, we calculated w and phi of each degree. Basically, we used the data train for calculating w train and phi train, then we obtain the result y. The result is plotted in green.

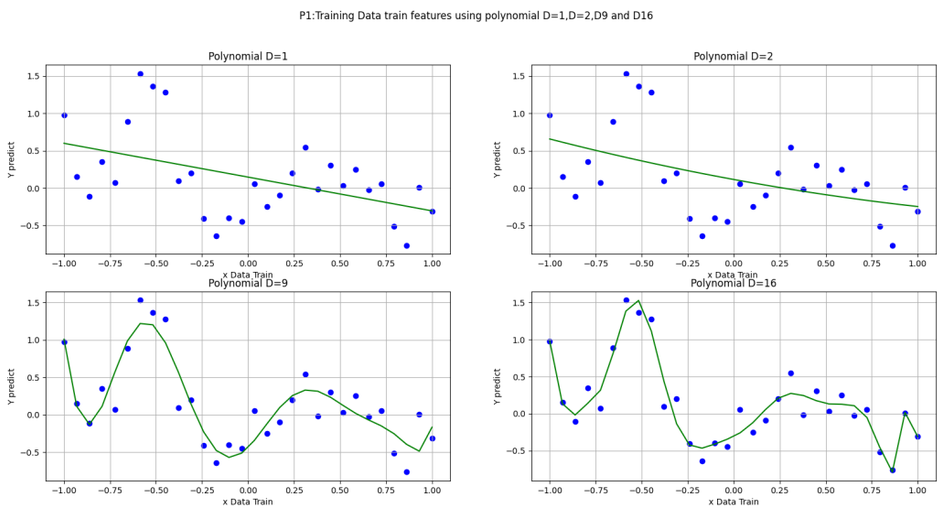

Figure 8: Y predicted using data train and plotted data train

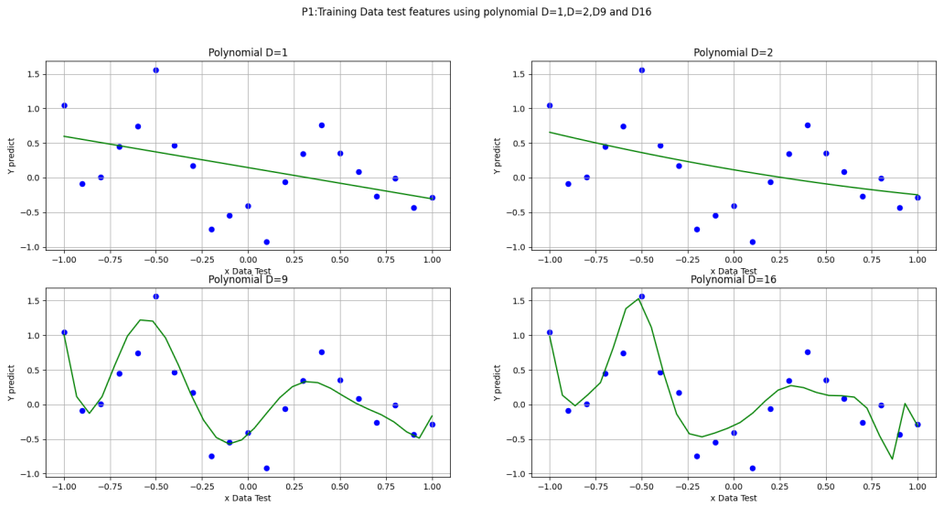

Figure 9: Y predicted using data train and plotted data test

Conclusion: Depending on the degree that we are using, we will obtain a value that fit better the samples that we have.

We used the formula that was proportionate for solving the problem $E(w) = \frac{1}{2}||t - \phi w||^2$, depending on t(t train left and t test right) we will have different solution that we are presenting.

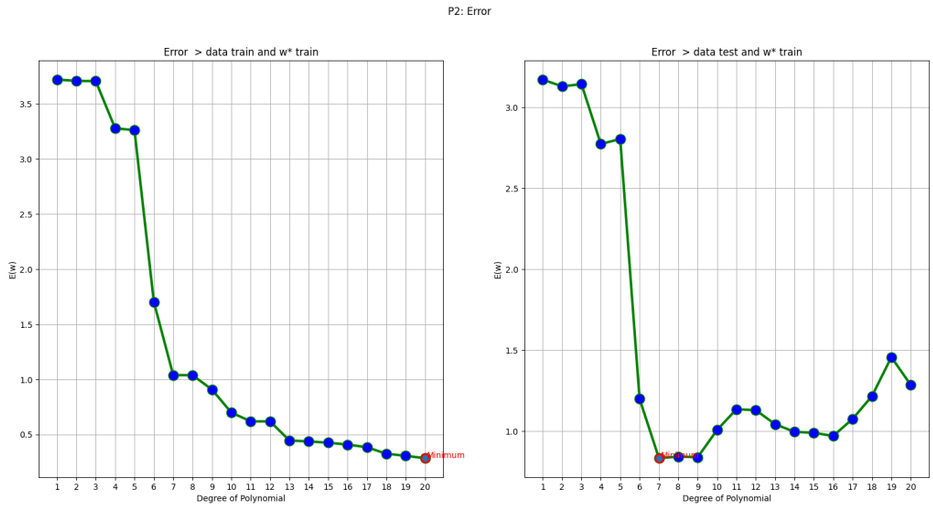

Figure 10: Error using W train and Phi Train for data train and data test

Conclusion: For the left graphic, the value of the error will decrease depending on the value of the degree (height degree low error). On the other hand, when you want to apply w train and phi train for other targets (test) / other data set, the behavior is different, the minimum error will be found in a lower degree.

In the next exercise, we will repeat the steps that we made before, but this time we will use data test features for calculating w test and phi test. Finally, we will plot the calculate Y in green, the data test (Figure 11) and data train (Figure 12) in blue. On Figure 11 and 12 the result of the degree D = 1,2,9 and 16 are included.

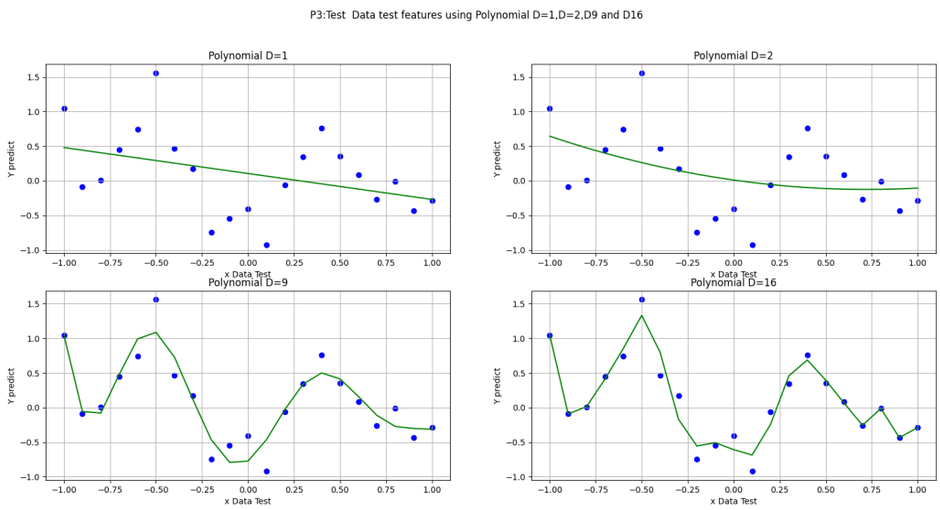

Figure 11: Y predicted using data test and plotted data test

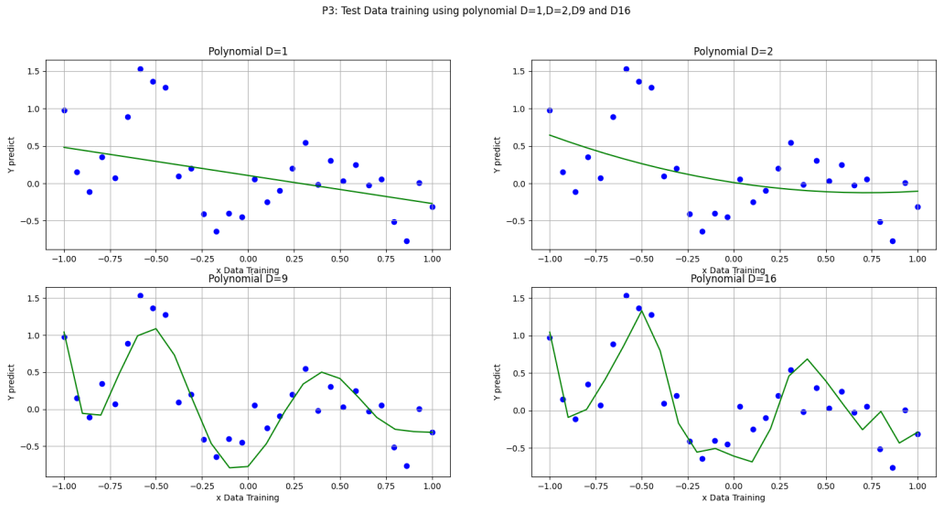

Figure 12: Y predicted using data test and plotted data train

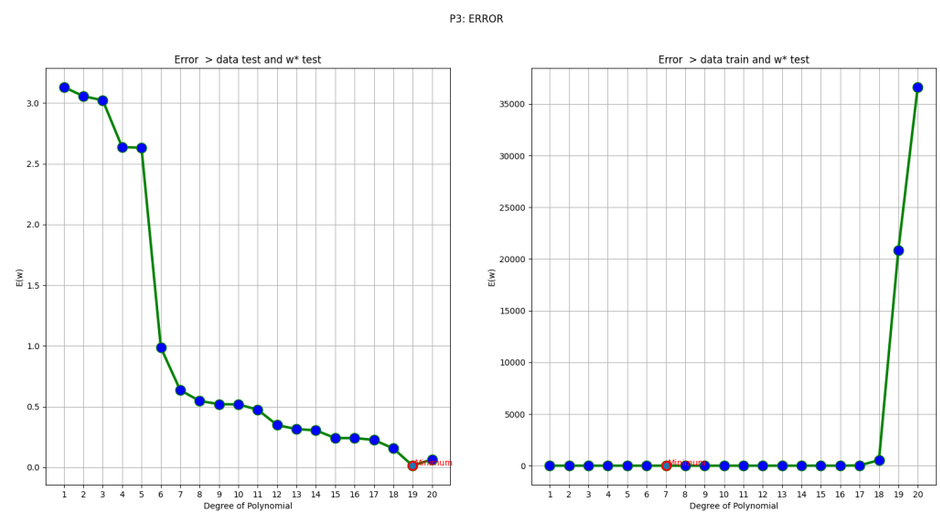

Figure 13: Error using W test and Phi test for data test and data train

Conclusion: For the left graphic, the value of the error will decrease depending on the value of degree, this time we will find a minimum before the degree 20 (degree 19). On the other hand, in the right graphic, the error will increase so much around the degree 19. A minimum using data train, means an increase using the model applied (using w test phi test)

2.2.2 Linear Regression based on regularized error function

In the next exercise, we repeated the steps that we made before for the Figure 8 and Figure 9, but we applied to a regularization parameter lambda = 0.1. Finally, we will plot the calculate Y in green and the data train (Figure 14) and data test (Figure 15) in blue. On Figure 14 and 15 the result of the degree D = 1,2,9 and 16 are included.

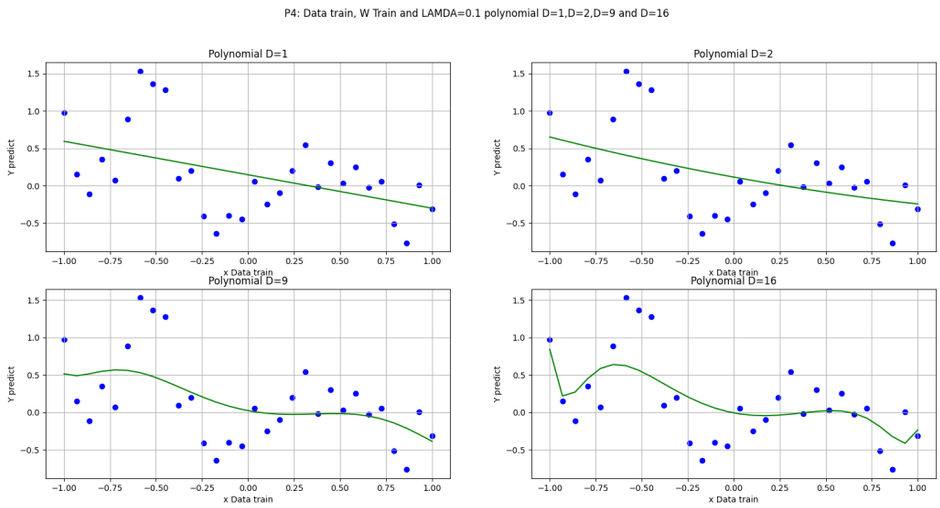

Figure 14: Y predicted using data train and lambda=0.1 and plotted data train

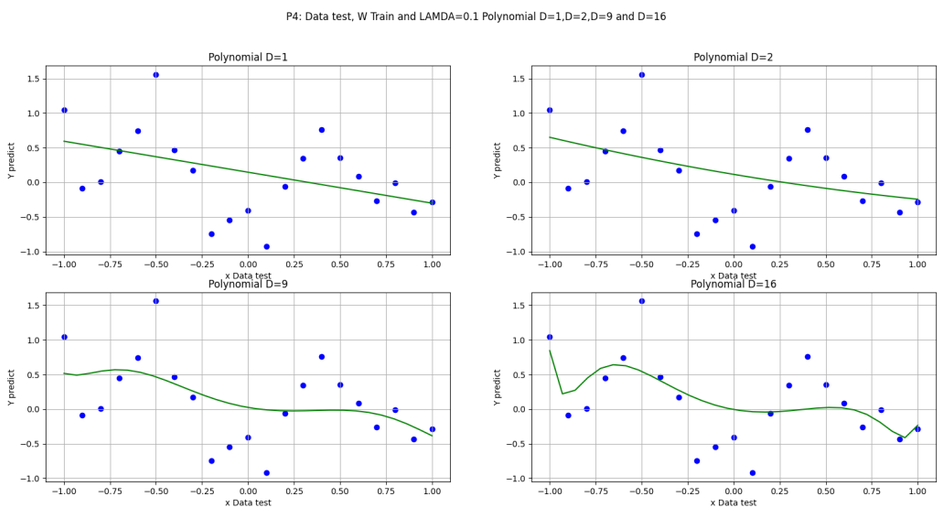

Figure 15: Y predicted using data train and lambda=0.1 and plotted data test

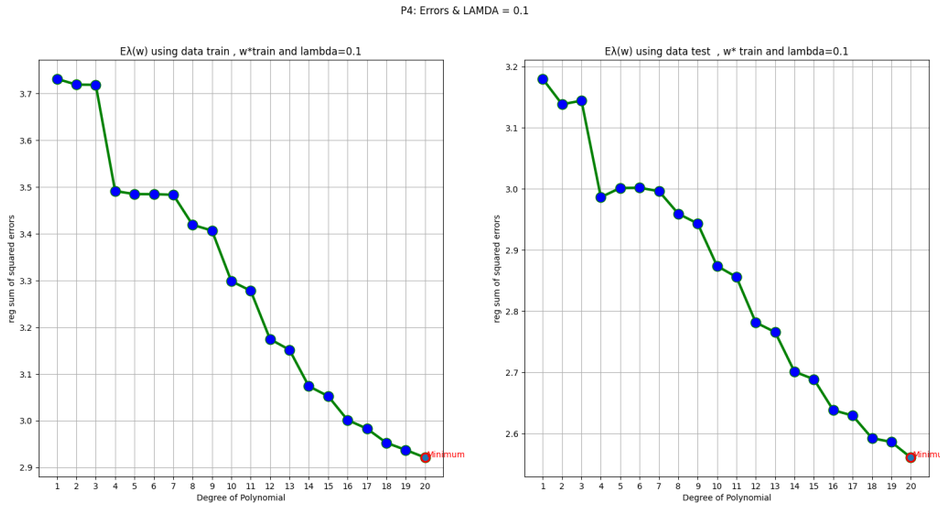

Figure 16: Error using W train,Phi Train and lambda=0.1 for data train and data test

Conclusion: We will attenuate the Y depending on the lambda that we want applied. When the lambda has a higher value, Y will be attenuated, otherwise when lambda is so low you will have the same value that you made in the Figure 8 and Figure 9, you are changing the Y but not so much because you are applying lower values. It will also affect to the error, because if you attenuate the Y, the error between the Y and the values plotted in blue will increase. We can see it in the figure 16.

On the figure 17 and figure 18, we plotted the error using different lambda. Figure 17 using lambda= 0.0001 and Figure 18 using lambda = 100.

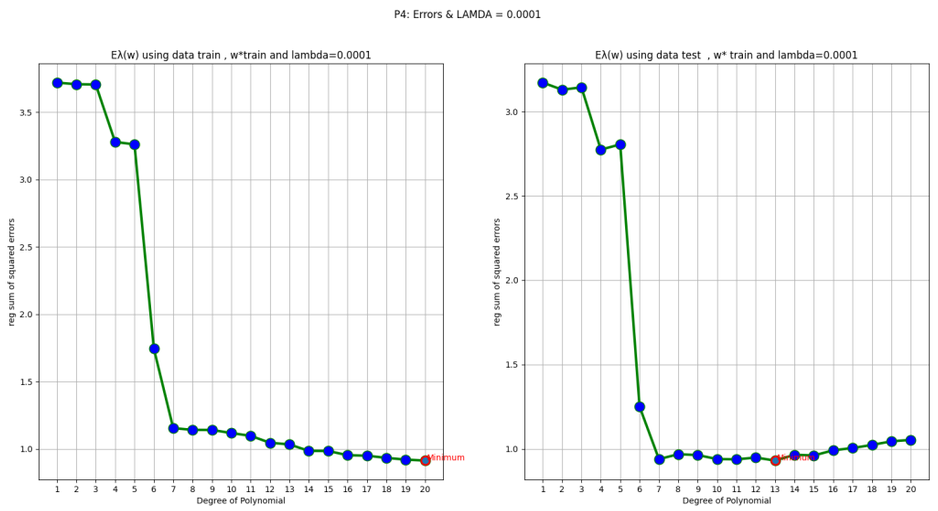

Figure 17: Error using W train,Phi Train and lambda=0.0001 for data train and data test

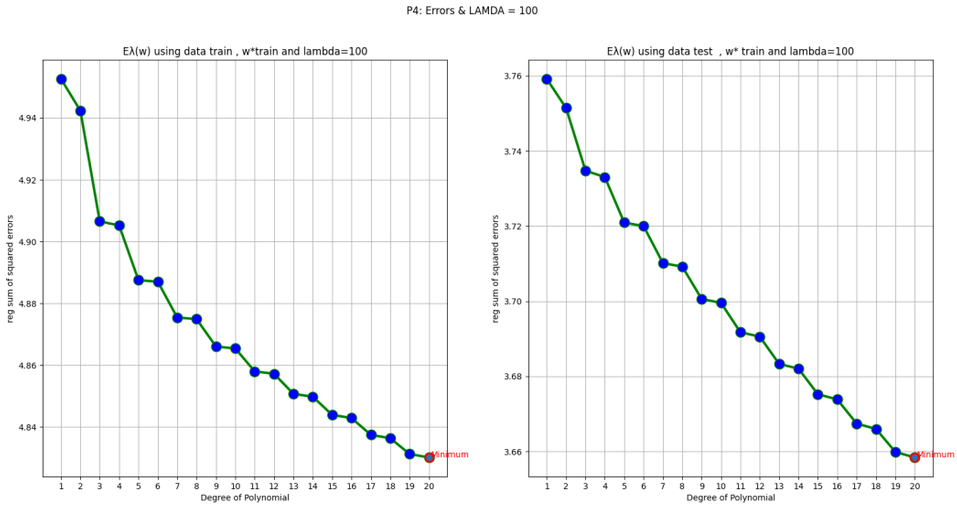

Figure 18: Error using W train,Phi Train and lambda=100 for data train and data test

Conclusion : We verified all the conclusion that we wrote in the last conclusion. Firstly, the error is higher when the lambda is higher and the error is different, the error that we have in the Figure 10. The Y is different for higher degrees. On the other hand, using lower lambdas (in this case 0.0001) we will obtain an error similar to the error that we found in the Figure 10. All that we wrote before make sense, because we applied the formula $E_\lambda(w) = E(w) + \frac{\lambda}{2}||w||^2$ for solving this problem.