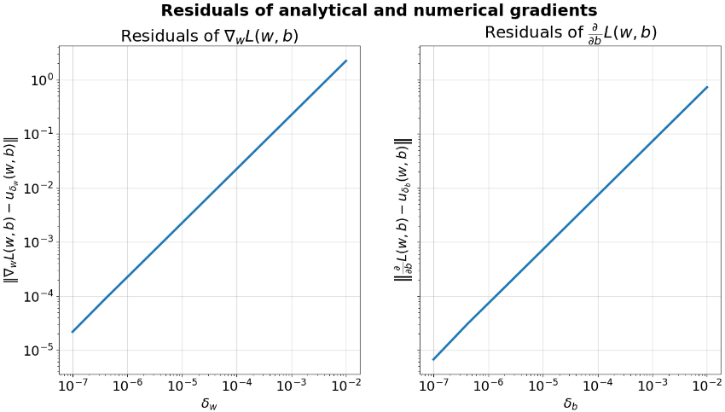
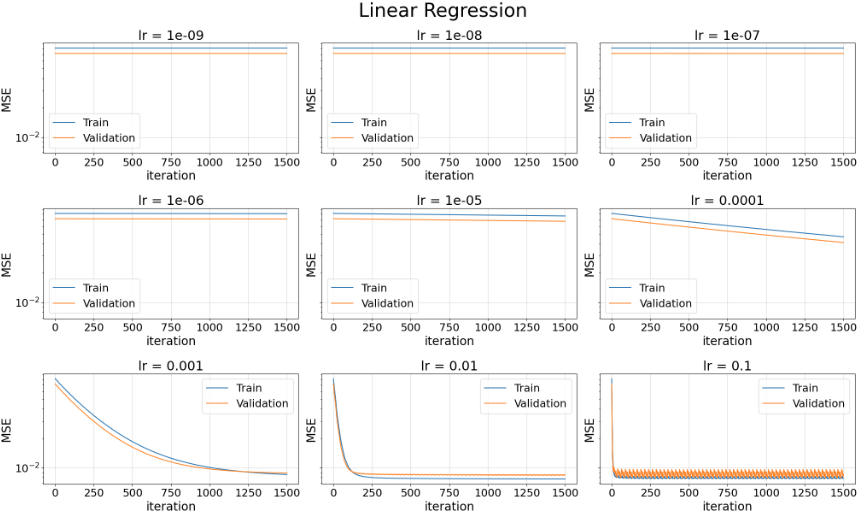
***Section 1***

***(A1)***

***(A2)***

***(A3)***

*lr size = 0.01, Best train loss = 0.508229717973801, Best validation loss = 0.14791003436298927*

*lr = 0.01 is the best learning rate since it has the best validation loss out of all other learning rates, it’s not jittery as lr = 0.1(step size is too big to find settle for a minimum) and out of all non-jittery graphs – it takes the least steps to get to the minimum.*

*It doesn’t make sense for it to increase the number of steps above 1500 since it’s way more then enough for it to get to it to descent to the minimum*

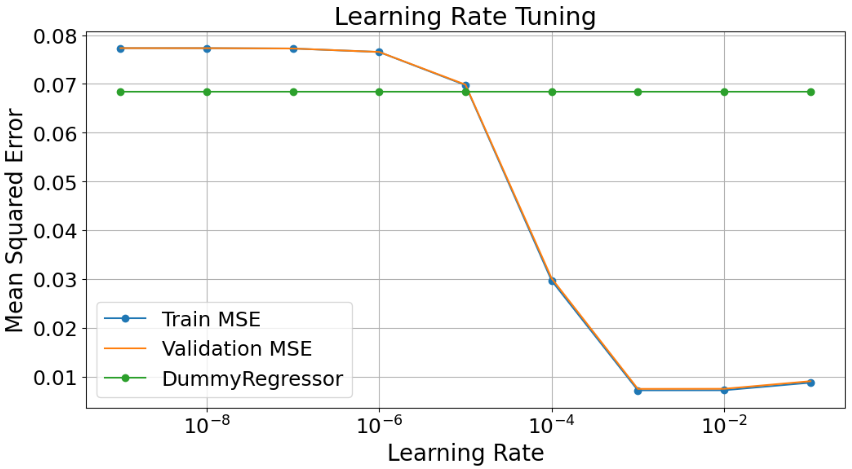
***Section 2***

***(A4)***

|  |  |  |  |
| --- | --- | --- | --- |
| ***Model*** | ***Section*** | ***Train MSE*** | ***Valid MSE*** |
|  |  |  | |
| ***Dummy*** | ***2*** | ***0.0692*** | ***0.0693*** |

***(A5)***

|  |  |  |  |
| --- | --- | --- | --- |
| ***Model*** | ***Section*** | ***Train MSE*** | ***Valid MSE*** |
|  |  | ***Cross validated*** | |
| ***Dummy*** | ***2*** | ***0.0692*** | ***0.0693*** |
| ***Linear*** | ***2*** | ***0.0762*** | ***0.0762*** |

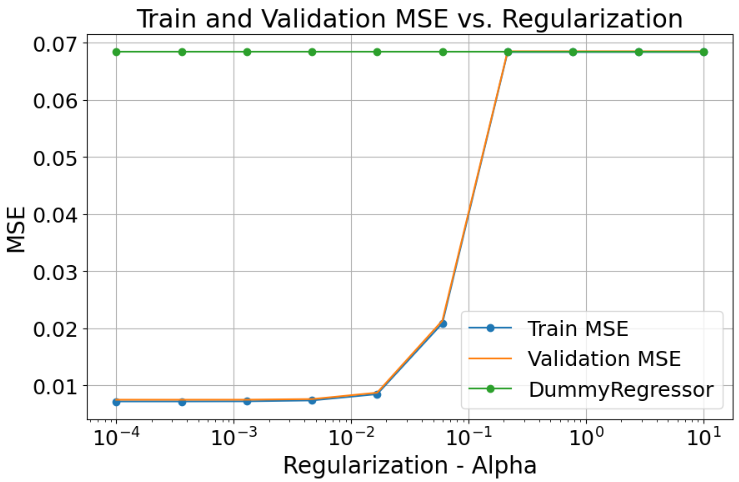
* *****Optimal Learning Rate: 0.001; Validation MSE: 0.0762***

***(A6)***

***Linear Regressor:*** *no, because the Normalization is Linear for both minmax and normal therefore linear models are not affected by normalization of linear features.*

***Dummy Regressor:*** *no, because it always predicts the mean of the target values and does not use the features for predictions.*

***Section 3***

***(A7)***

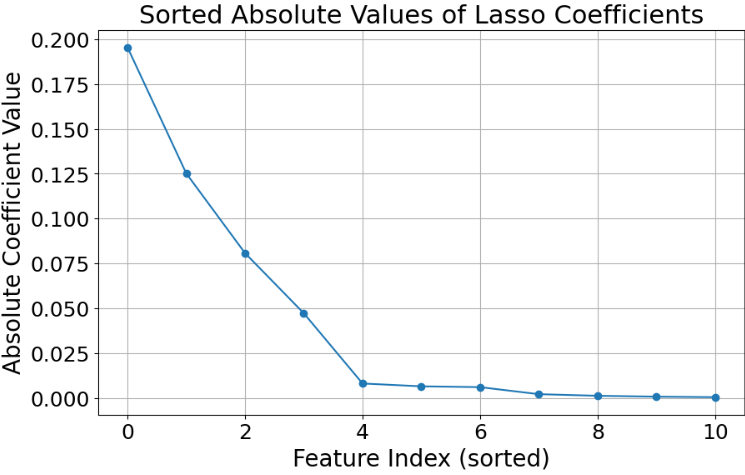
* ***Optimal Regularization (Alpha): 0.0018; Validation MSE: 0.0079***

***(A8)***

|  |  |  |  |
| --- | --- | --- | --- |
| ***Model*** | ***Section*** | ***Train MSE*** | ***Valid MSE*** |
|  |  | ***Cross validated*** | |
| ***Dummy*** | ***2*** | ***0.0692*** | ***0.0693*** |
| ***Linear*** | ***2*** | ***0.0762*** | ***0.0762*** |
| ***Lasso*** | ***3*** | ***0.0078*** | ***0.0079*** |

***(A9)*** *Top 5 Features by Coefficient (in absolute value):*

1. *PCR\_04: 0.1947559693612877*
2. *sugar\_levels: 0.13228166292553778*
3. *PCR\_06: 0.07825556680416525*
4. *PCR\_02: 0.0778865600522928*
5. *PCR\_09: 0.005492021817041533*

***(A10)***

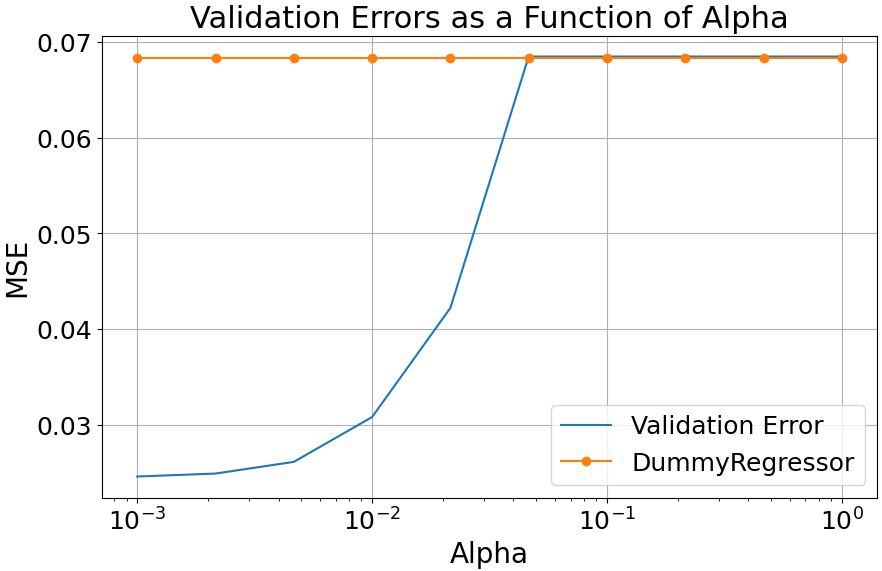
***(A11)*** *as was taught in the Tutorial, the regularization in the Lasso Regression causes “variable selection” by shrinking some coefficients to zero; meaning that larger magnitudes suggest that the feature has a stronger effect on the target variable. It helps us find the features that are a significant indicator of contamination\_level.*

***(A12)*** *Had we not normalized the feature beforehand, the training performance of the Lasso model could have changed. The model would prefer features with higher range of values because their values are the dominating values w.r.t the features with the lower value. This could make the model biased towards those features, normalizing the data address this issue.*

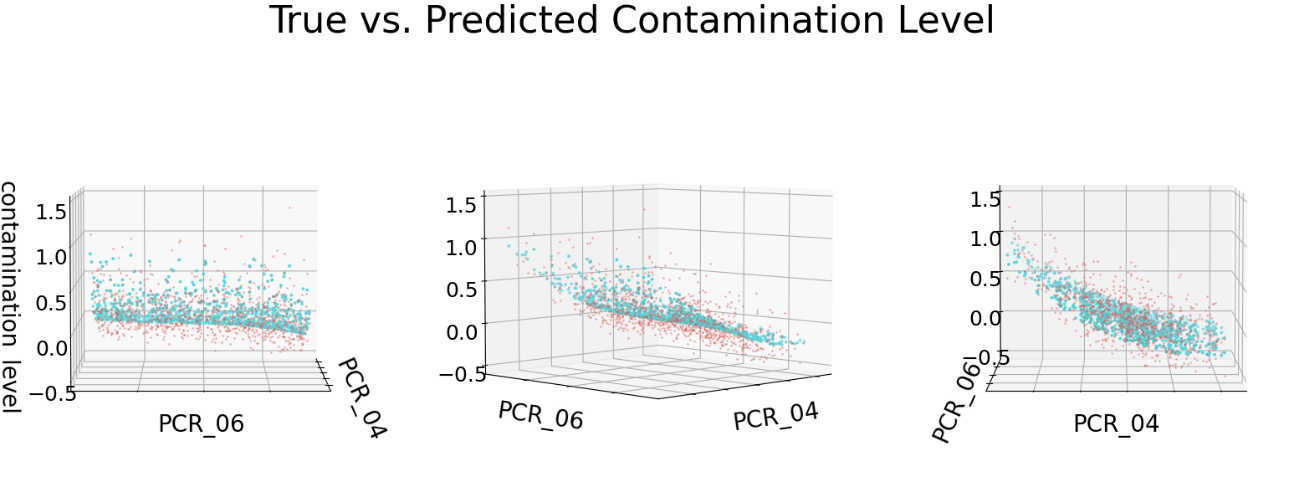
***(A13)*** *Should we used Ridge regressor instead, the coefficients of the trained model would not turn zero, which will make it more difficult to recognize the importance of the features, opposing to the Lasso regressor.*

***Section 4***

***(A14)*** *after the poly-mapping, the features will have different scales because of their different degrees* , *we will see a big difference if their original values are large; without normalization they will contribute unevenly to the model’s prediction. Therefore, we should normalize to ensure that the features are on a similar scale - which improves the performance of gradient-based optimization algorithms.*

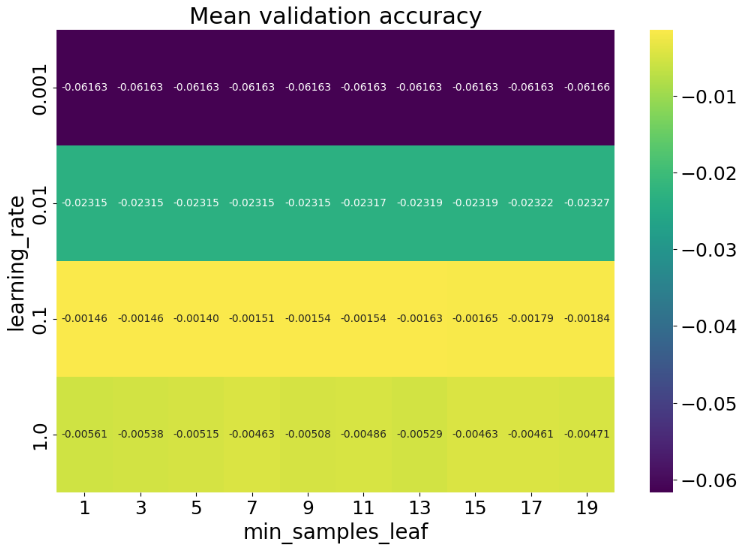
***(A15)***

* ***Optimal Regularization (Alpha): 0.00001; Validation MSE: 0.0048***

***(A16)***

***(A17)***

|  |  |  |  |
| --- | --- | --- | --- |
| ***Model*** | ***Section*** | ***Train MSE*** | ***Valid MSE*** |
|  |  | ***Cross validated*** | |
| ***Dummy*** | ***2*** | ***0.0692*** | ***0.0693*** |
| ***Linear*** | ***2*** | ***0.0762*** | ***0.0762*** |
| ***Lasso*** | ***3*** | ***0.0078*** | ***0.0079*** |
| ***Polynomial Lasso*** | ***4*** | ***0.0047*** | ***0.0048*** |

***Section 5***

***(A18)***

***Best Parameters:*** *gbm\_\_learning\_rate: 0.1, gbm\_\_min\_samples\_leaf: 5*

***Best Training Error (MSE):******0.0013***

***Best Validation Error (MSE):******0.0004***

***(A19)***

|  |  |  |  |
| --- | --- | --- | --- |
| ***Model*** | ***Section*** | ***Train MSE*** | ***Valid MSE*** |
|  |  | ***Cross validated*** | |
| ***Dummy*** | ***2*** | ***0.0692*** | ***0.0693*** |
| ***Linear*** | ***2*** | ***0.0762*** | ***0.0762*** |
| ***Lasso*** | ***3*** | ***0.0078*** | ***0.0079*** |
| ***Polynomial Lasso*** | ***4*** | ***0.0047*** | ***0.0048*** |
| ***GBM Regressor*** | ***5*** | ***0.0013*** | ***0.0004*** |

***Section 6***

***(A20)***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***Model*** | ***Section*** | ***Train MSE*** | ***Valid MSE*** | ***Test MSE*** |
|  |  | ***Cross validated*** | | ***Retrained*** |
| ***Dummy*** | ***2*** | ***0.0692*** | ***0.0693*** | ***0.0639*** |
| ***Linear*** | ***2*** | ***0.0762*** | ***0.0762*** | ***0.0061*** |
| ***Lasso*** | ***3*** | ***0.0078*** | ***0.0079*** | ***0.0060*** |
| ***Polynomial Lasso*** | ***4*** | ***0.0047*** | ***0.0048*** | ***0.0220*** |
| ***GBM Regressor*** | ***5*** | ***0.0013*** | ***0.0004*** | ***0.0008*** |

***GBM Regressor performed with an error of 0.0008 on the test data.***

***We can see from the table that no model suffers from over or under fitting, since all the MSE are relatively small and no major difference between the train/valid MSE and the test MSE.***