Advanced Data Analysis Techniques

Project Presentation

Carlos Brito - 2018286603
Eduardo Ribeiro - 2017254671

Supervisor: Tiago Cerqueira

11th of January 2023

Universidade de Coimbra

Index

Introduction

In this work we use the algorithm Crystal Graph Convolution Neural Networks (CGCNN) as base to predict properties from a given dataset. We compare two different models implemented in the algorithm and see how the predictions of the propertie studied compares with the target value.

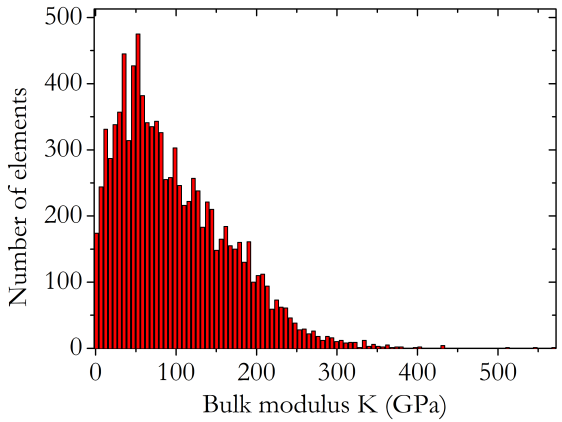
Dataset analysis and characterization

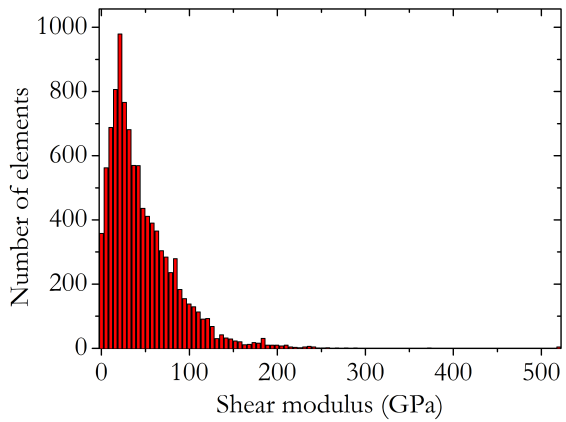
The dataset that we used is composed by 10 000 different compounds. We have three properties available: Bulk and Shear modulus and Vickers hardness.

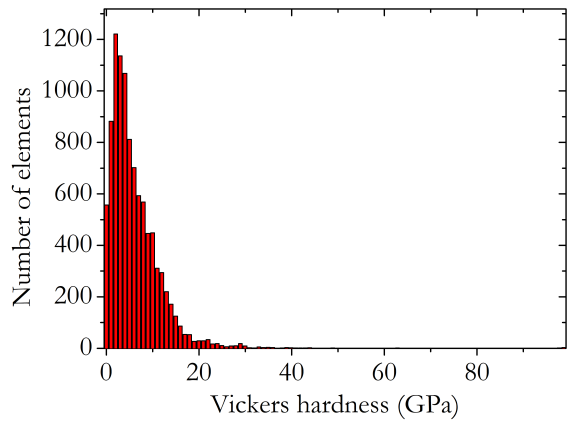
- Bulk modulus (K) is a measure of how resistant to compression the material is
- Shear modulus (G) is a measure of the elastic shear stiffness of a material
- Vickers hardness (HV) is a method to measure the hardness of a material

Dataset analysis and characterization

We started by obtaining an histogram for all this properties in order to know our data set:







Looking to the histograms we see asymmetric distributions similar to Poisson or Gaussian Distributions. Materials with high values for this properties are rare and of interest. To have an ideia, the values of this properties, for diamond, are the following: $K = 443GPa$, $G = 478GPa$ and $HV = 115GPa$.

Dataset analysis and characterization

We present a table with some important parameters about the dataset represented in the histograms.

	Shear modulus (GPa)	Bulk modulus (GPa)	Vickers hardness (GPa)
Mean	48.68	100.53	6.54
Standard Deviation	39.20	69.12	5.43
Minimum Value	0.00	1.00	0.02
Maximum Value	525.00	575.00	99.71

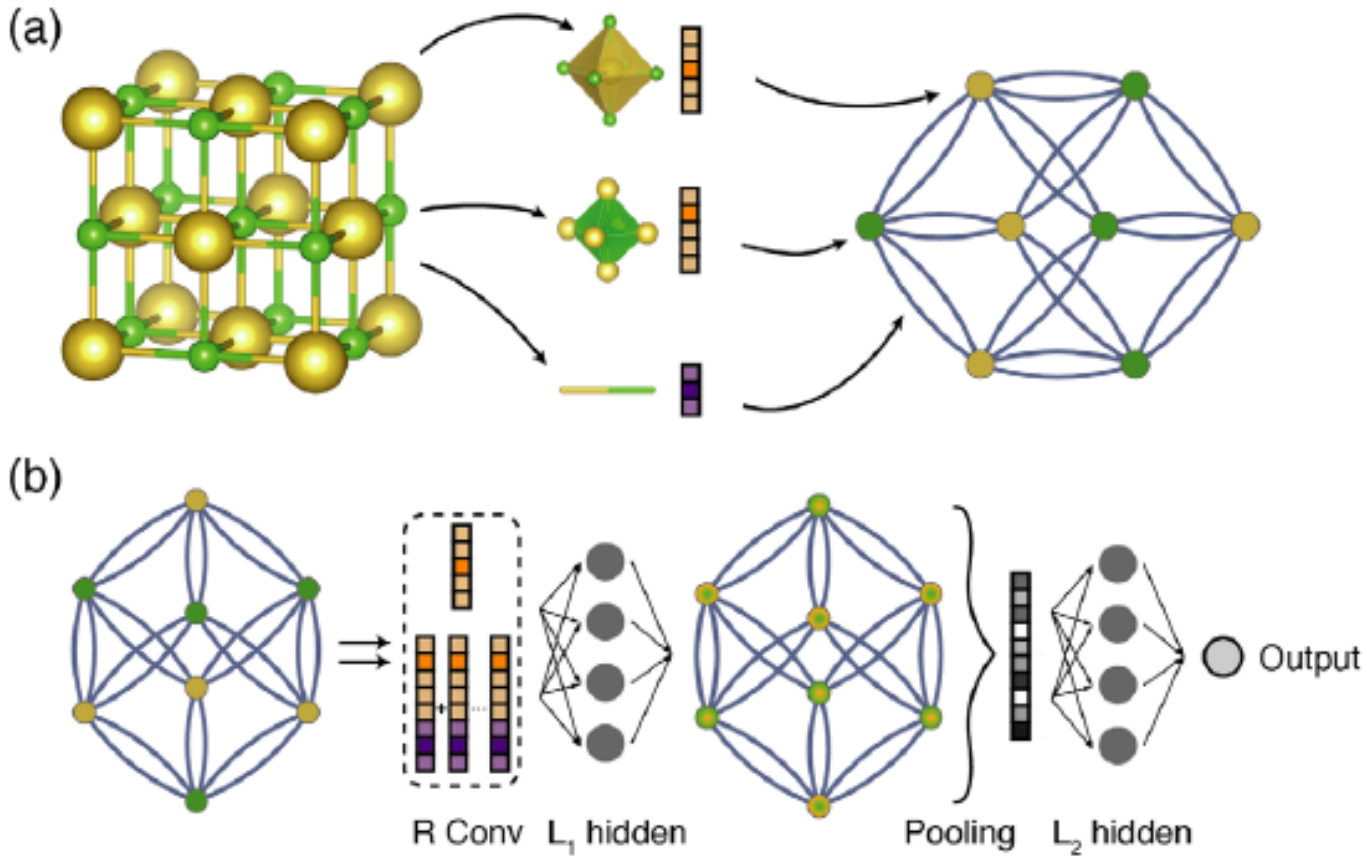
We clearly see that the dataset is unbalanced, since the majority of the parameters values are in the first half of the interval of values and the distributions are not uniform.

Dataset analysis and characterization

- From this dataset we can have some problems. Overfitting or underfitting and, depending on the sampling, loss of information
- In order to work with this dataset we have to start to do some small manipulations. First, we need it to be normalized. For that, we've simply used python.
- We've also done a logarithmic scaling of the data, thus reducing the range of the data which makes the computing easier.
- In this work, we will only concentrate on the bulk modulus.

Crystal Graph Convolution Neural Networks

The several crystallographic structures of the dataset are the input of the the base algorithm used: Crystal Graph Convolution Neural Networks (CGCNN). In the figure we present a schematics of this algorithm.



Regression and Distribution Models

- We call Regression Model to the original algorithm from CGCNN code.

- The Distribution Model is the algorithm that was implemented to yield an estimation of standard deviation associated to the predicted value.

Loss Functions

The loss functions are a method to evaluate how the algorithm models the data. Since we have two different models, the loss function for each model is different.

- For the regression model, the loss function is the mean square error (MSE) that is the summation of the square of the difference between the prediction and the true value, for all, divided by the total number of values.

$$MSE = \frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{n}$$

Loss Functions

- For the distribution model, we use the Gaussian loss function where the targets are treated as samples from Gaussian distributions with expectations and variances predicted by the neural network and is given by

$$\mathcal{L}_D = \frac{1}{2}\sum_{i=0}^{|D|}\left[\log\ \sigma(x_i)^2 + \frac{(y_i - \mu(x_i))^2}{\sigma(x_i)^2}\right]$$

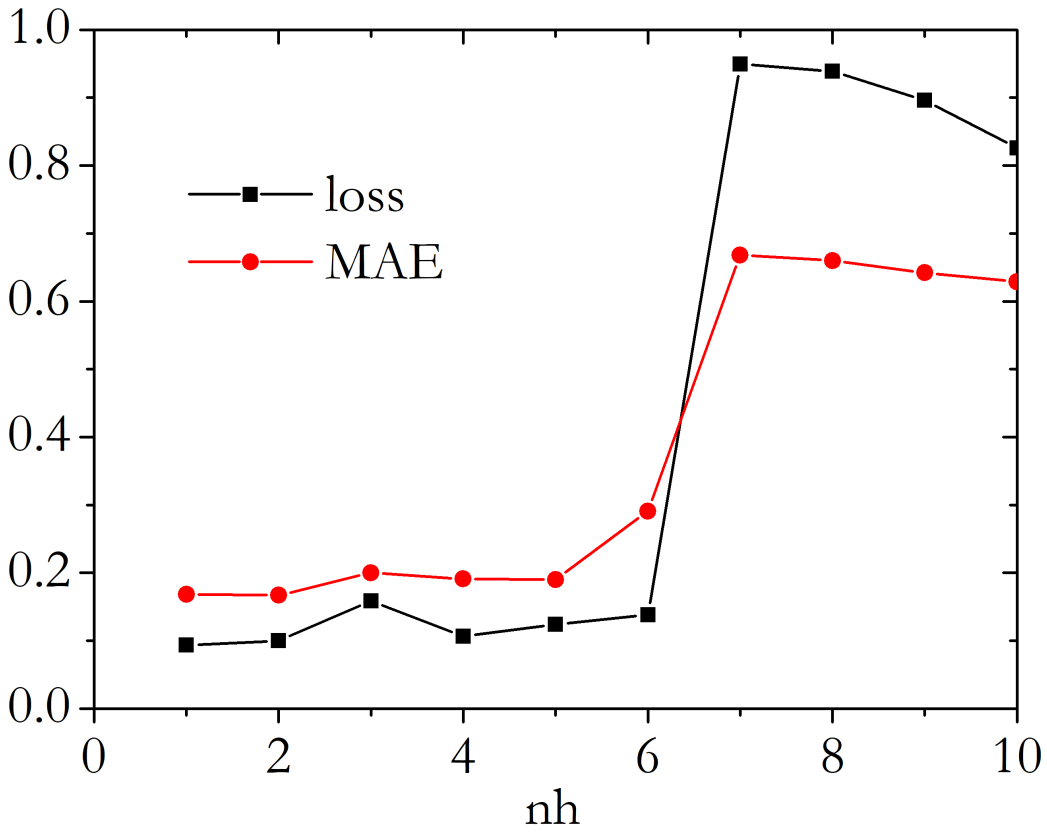
Mean Absolute Error (MAE)

- MAE is the mean of the absolute errors. Is the summation of module of the difference between the prediction and the true value, for all, divided by the total number of values. We will use it to compare both models.

$$MAE = \frac{\sum_{i=1}^{n} \mid y_i - \hat{y}_i \mid}{n}$$

Variation of the Number of Hidden Layers

Firstly, to get familiarized with the algorithm, we only varied the number of hidden layers in the model. We used 60% of the dataset for training, 20% for validation and other 20% for test.



Model Training with Multiple Hyperpameters

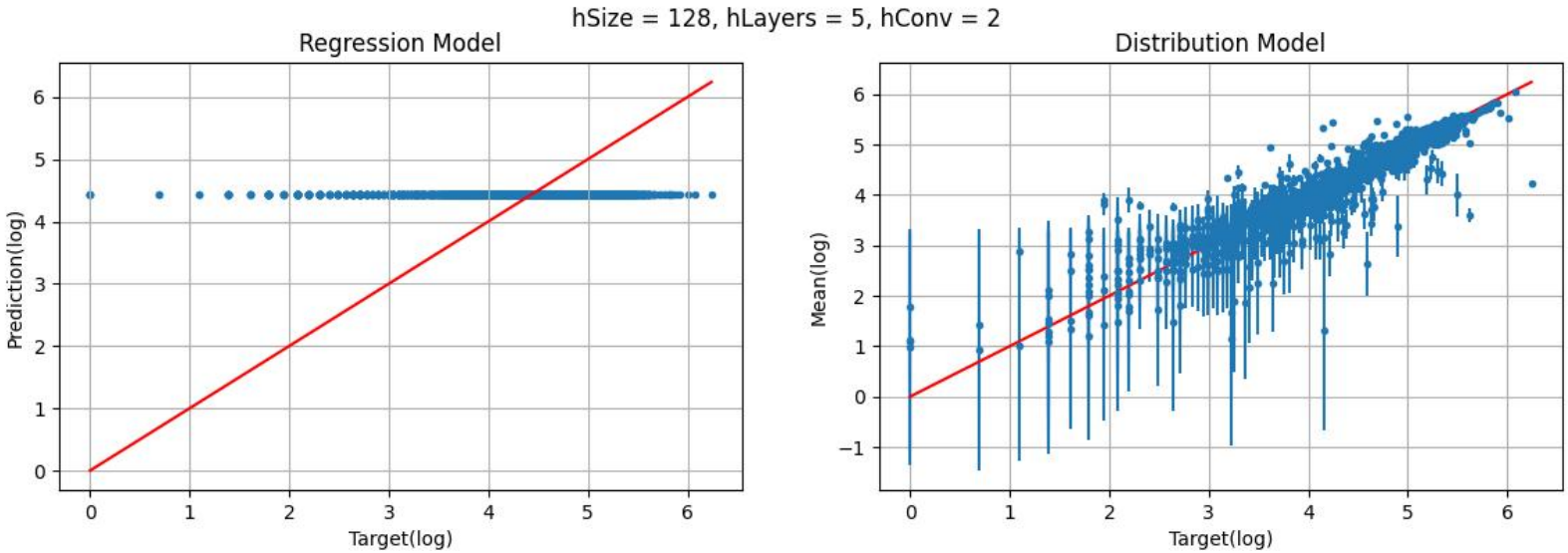
- We tested multiples combinations of hyperparameters, such as the hidden layer sizes, the number of hidden layers and the number of hidden convolution layers and chose the combinations with the least mean absolute error (MAE) and least loss function for more attentive analysis.
- From the previous graphic, we see that we can only vary the number of hidden layers from 1 to 6, because after these the MAE and the loss values are considerably higher.

So, the hyperparameters tested were all the possible combinations of the following:

- models - regression and distribution
- hiddenSizes - 16, 32, 64, 128
- hiddenLayers - 1, 2, 3, 4, 5, 6
- hiddenConvs - 2, 3, 4

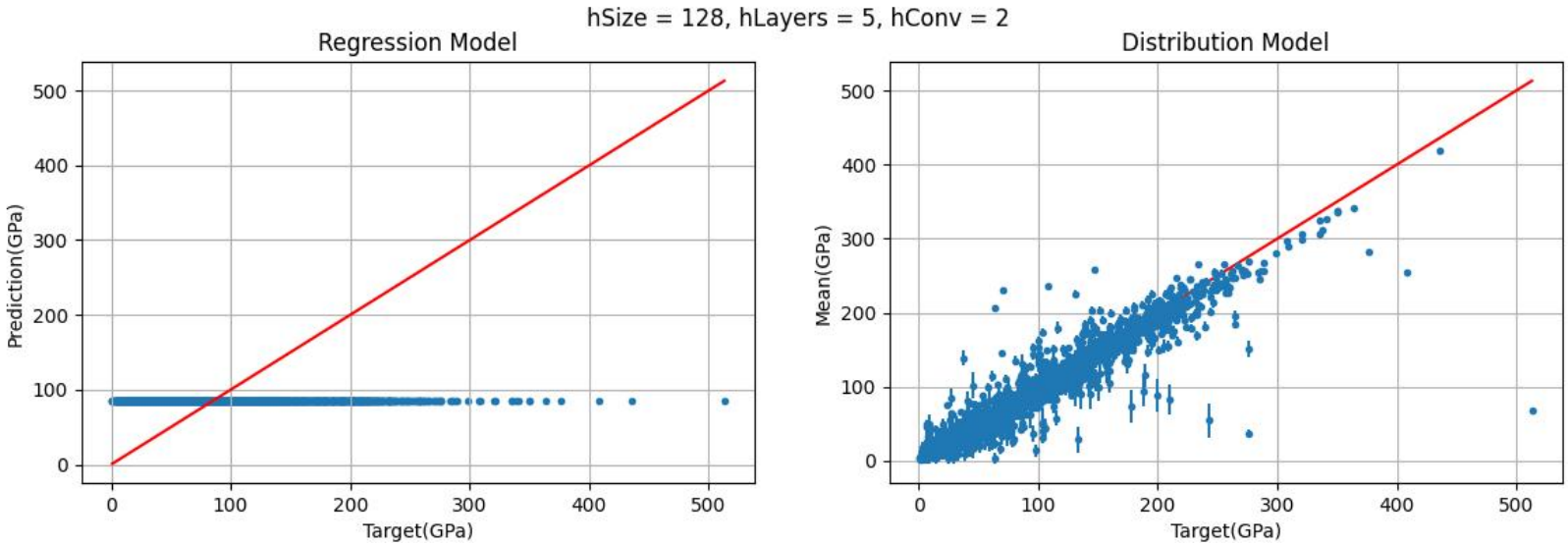
Problems with the Regression Model

Fitting problems are apparent in the Regression Model for some combinations of hyperparameters, this tends to not occur for the Distribution Model.



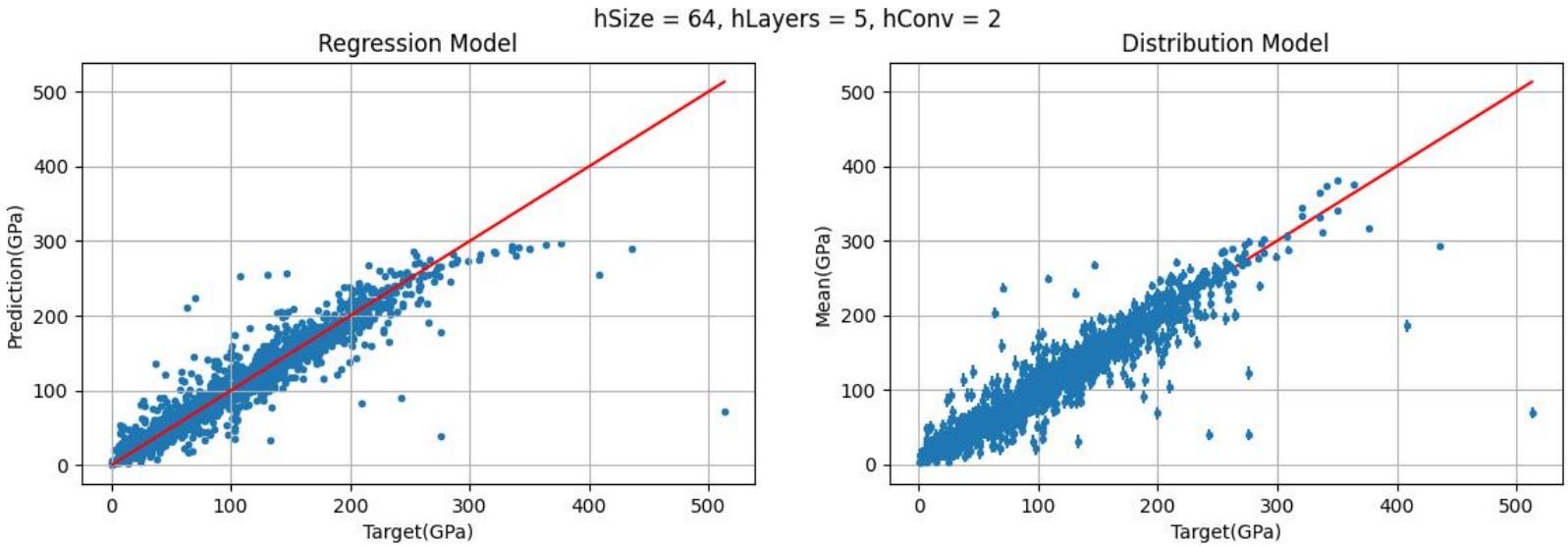
Problems with the Regression Model

Converting the logarithmic values to GPa units the representation of the previous fitting problem is presented bellow.



Problems with the Regression Model

For some combination of hyperparameters, it can seen that the results from the Regression Model tend to "saturate" after a certain threshold. This "saturation" is not present in the results from the Distribution Model.



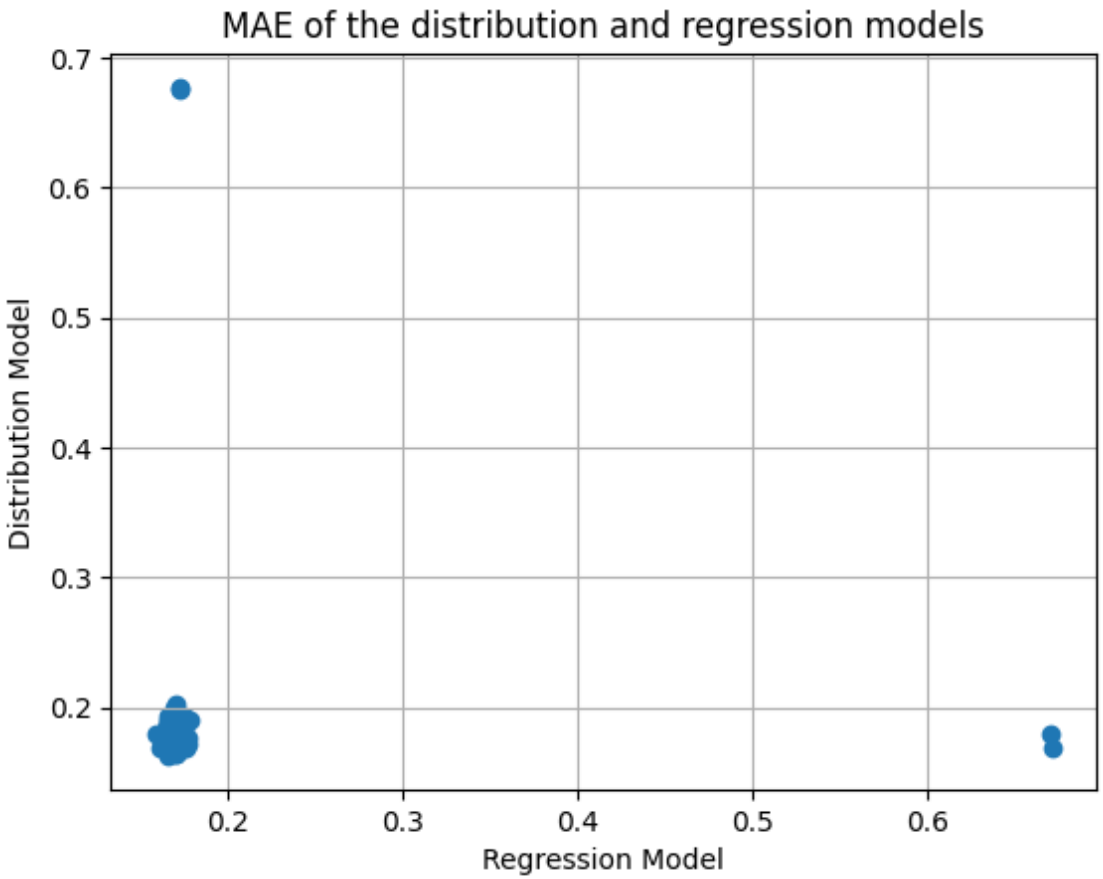
Mean Absolute Error (MAE) Analysis

Bellow is the minimal value of MAE of the trained distribution models and the corresponding hyperparameters. For comparison, it is also shown the regression model for this hyperparameters.

Model type	Hidden Size	Hidden Layers	Hidden Conv	Loss	MAE
Regression	64	4	4.0	0.1116	0.166
Distribution	64	4	4.0	-0.7334	0.163

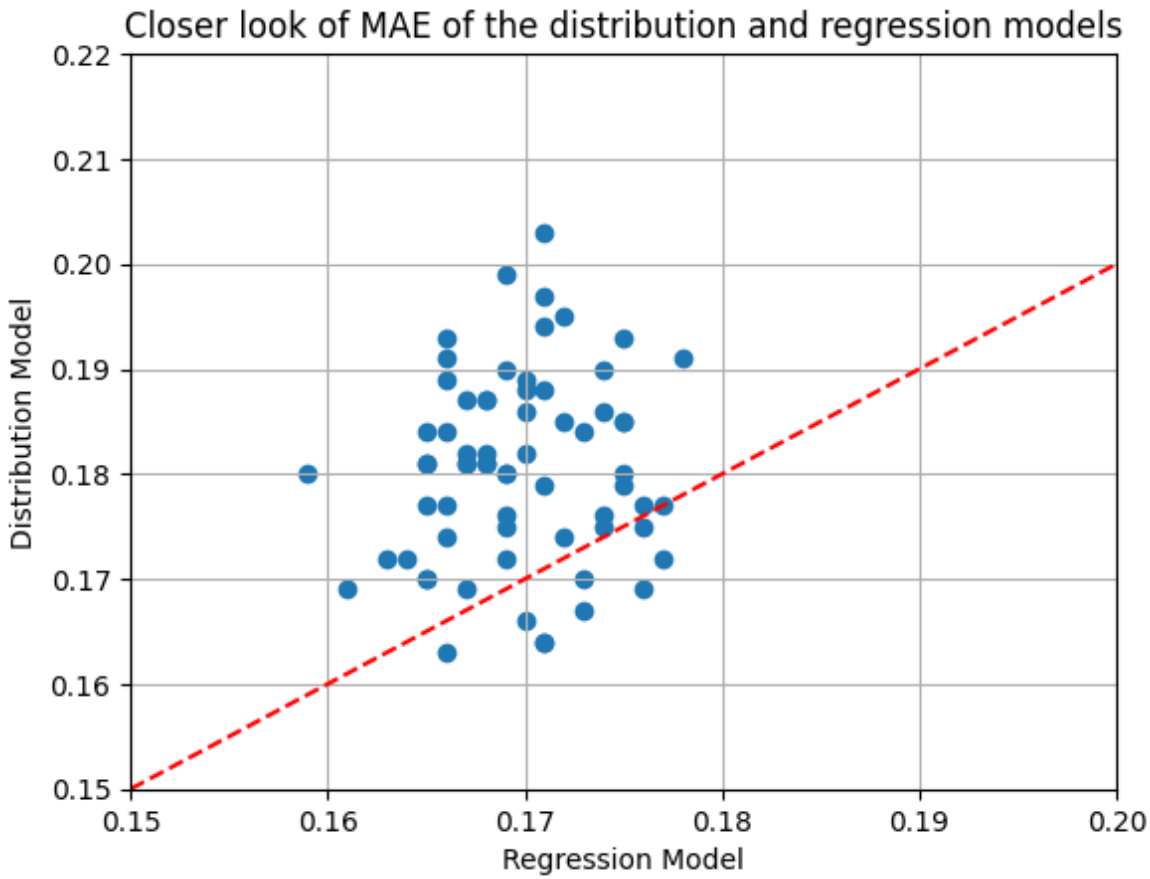
Mean Absolute Error (MAE) Analysis

Bellow we represent all the MAE obtained for both models. We see that most of the points are in the bottom left corner.



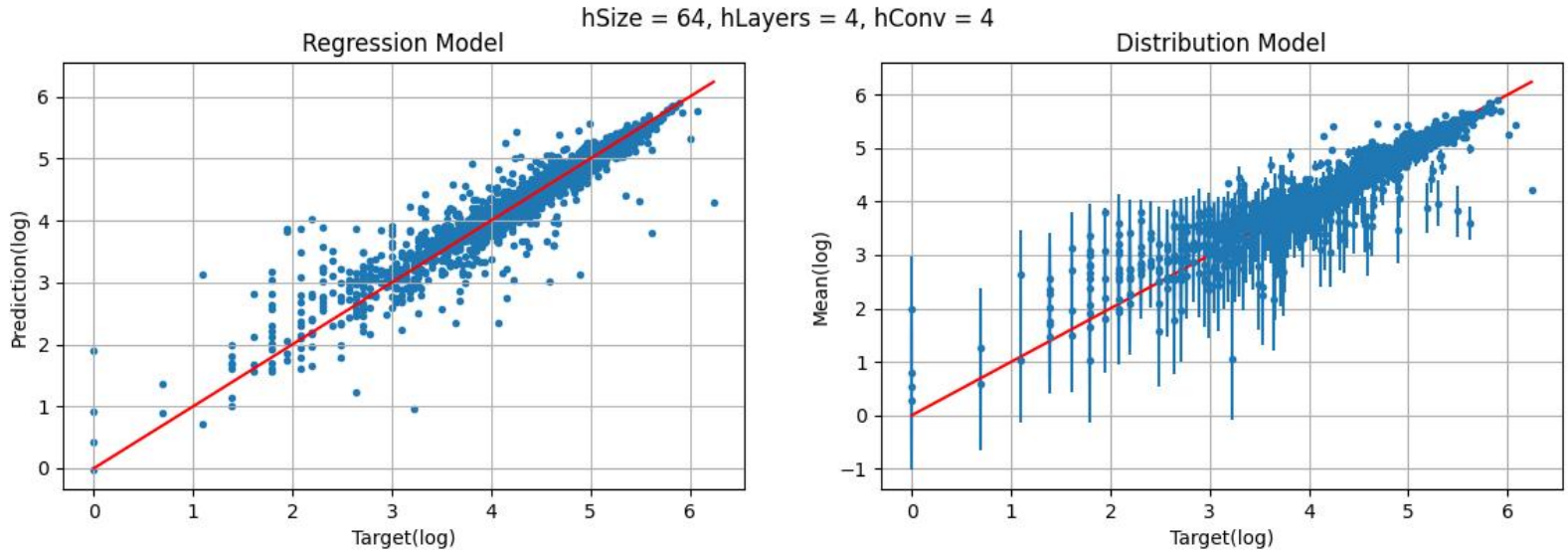
Mean Absolute Error (MAE) Analysis

Here we see with more detail most of the points and we can conclude that the MAE is higher for the Distribution Model.



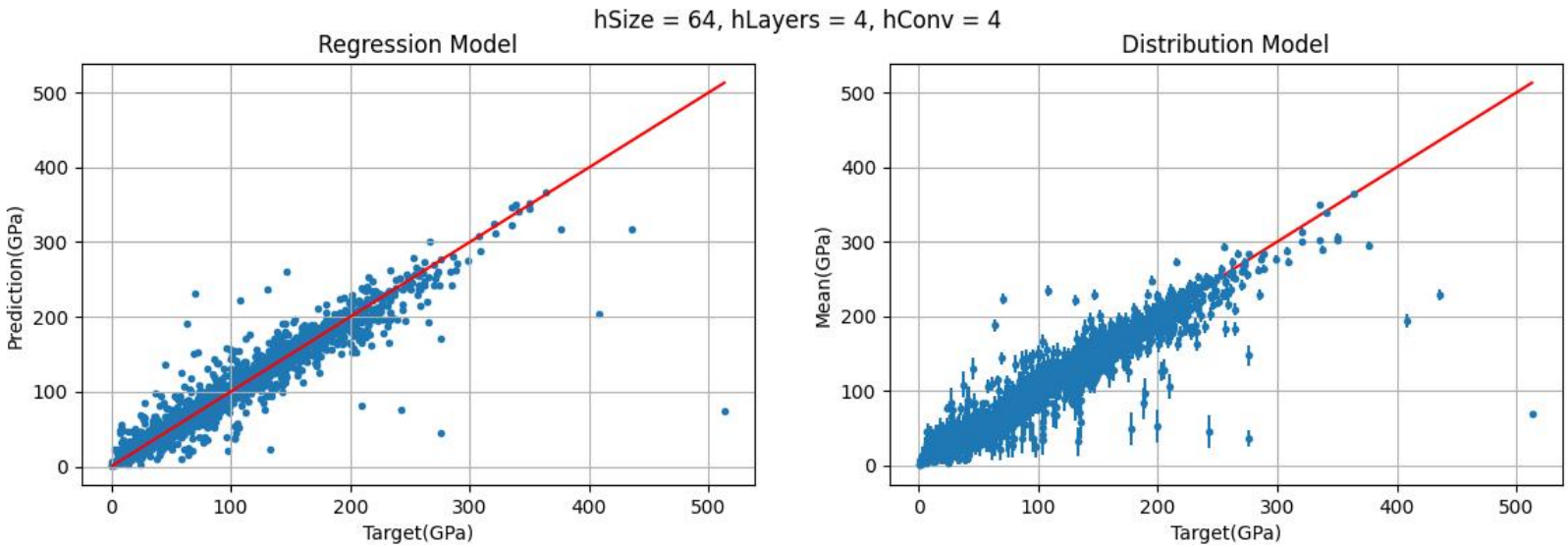
MAE Analysis, Graphs

For the parameters with the lowest MAE, we represent the logarithmic values given by the algorithm in function of the target values, from both Regression and Distribution Models.



MAE Analysis, Graphs

Converting the logarithmic values to GPa units the representation bellow is obtained.



As we can see, the results from both models are in agreement.

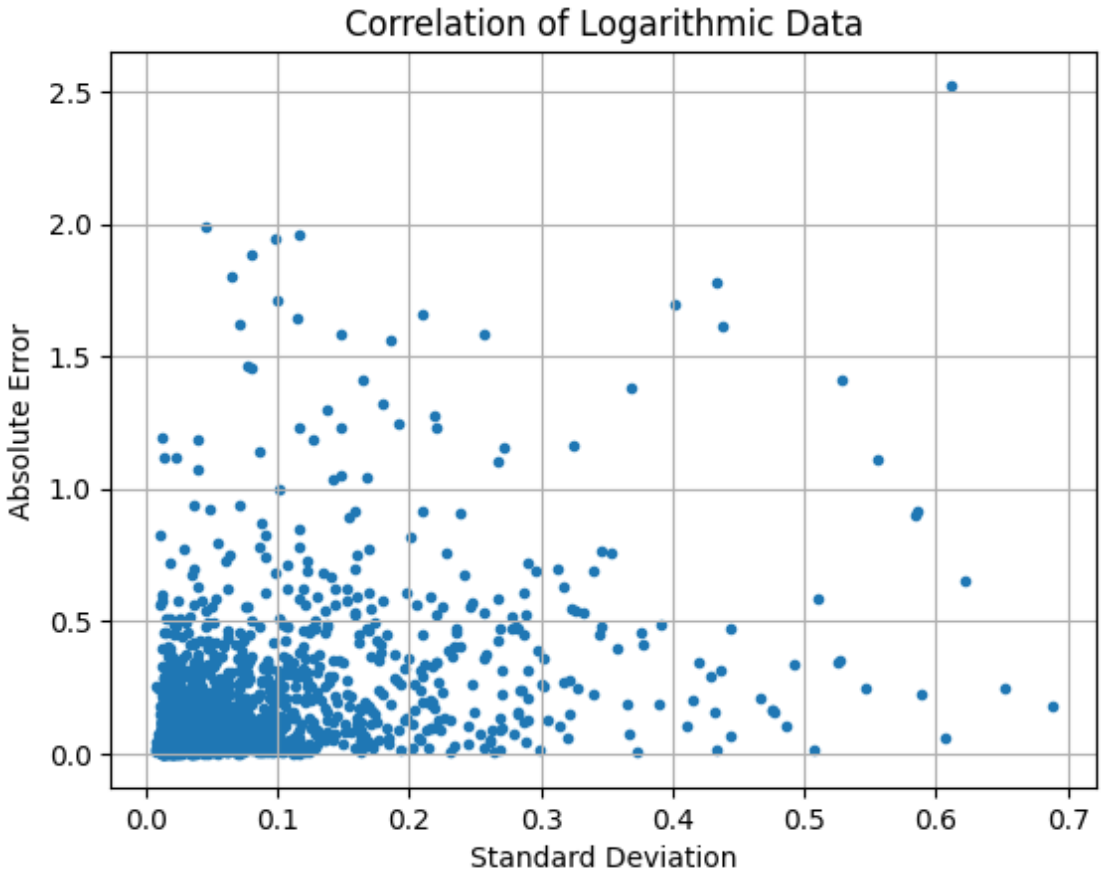
MAE Analysis, Confidence Interval

The results pertaining to the confidence interval of 95%
(target inside mean $\pm$ 1.96$\times$std) were determined and are presented below.

Model type	Units	Total Results	Results inside the CI	Ratio (%)
Regression	Log	2000	1749	87.45
	GPa		1757	87.85
Distribution	Log		931	46.65
	GPa		928	46.40

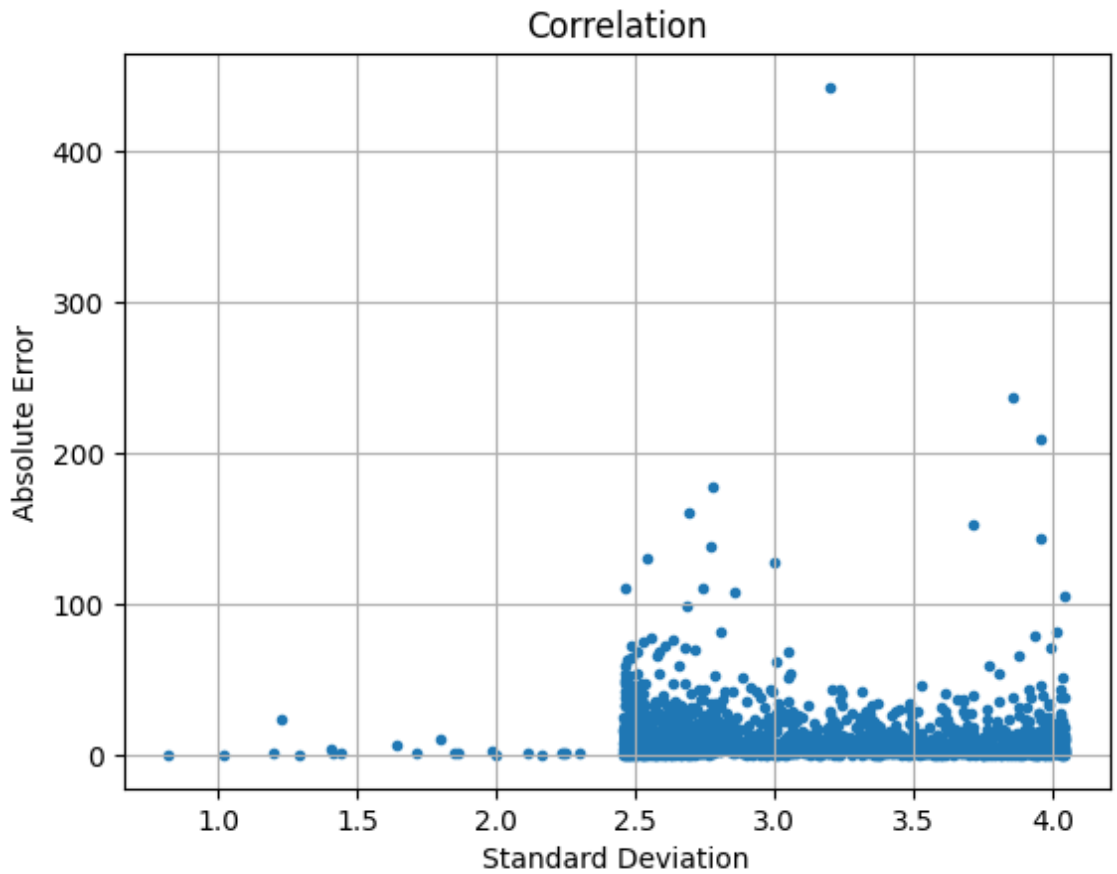
MAE Analysis, Correlation between STD and Absolute Errors

The Pearson correlation between the standard deviation and absolute error of the logarithmic values from the Distribution Model with least MAE value was determined and is of 0.400.

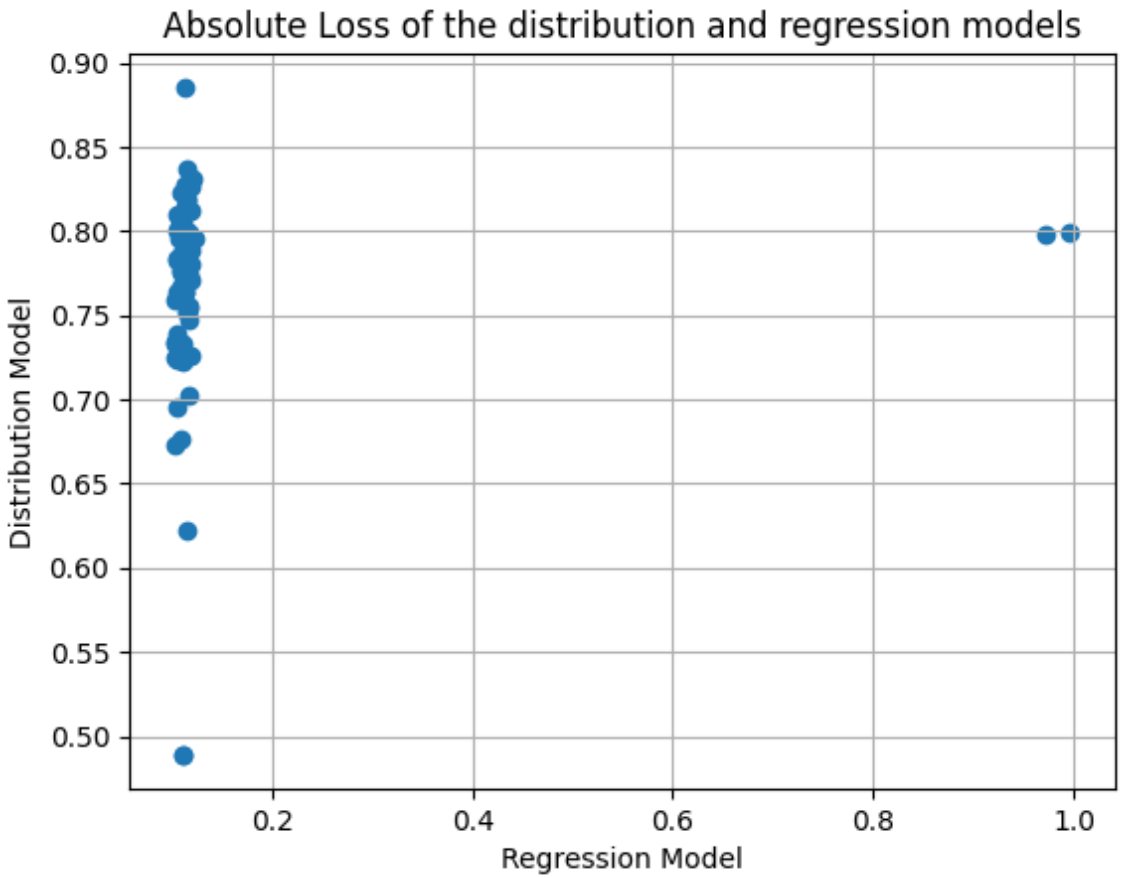


MAE Analysis, Correlation between STD and Absolute Errors

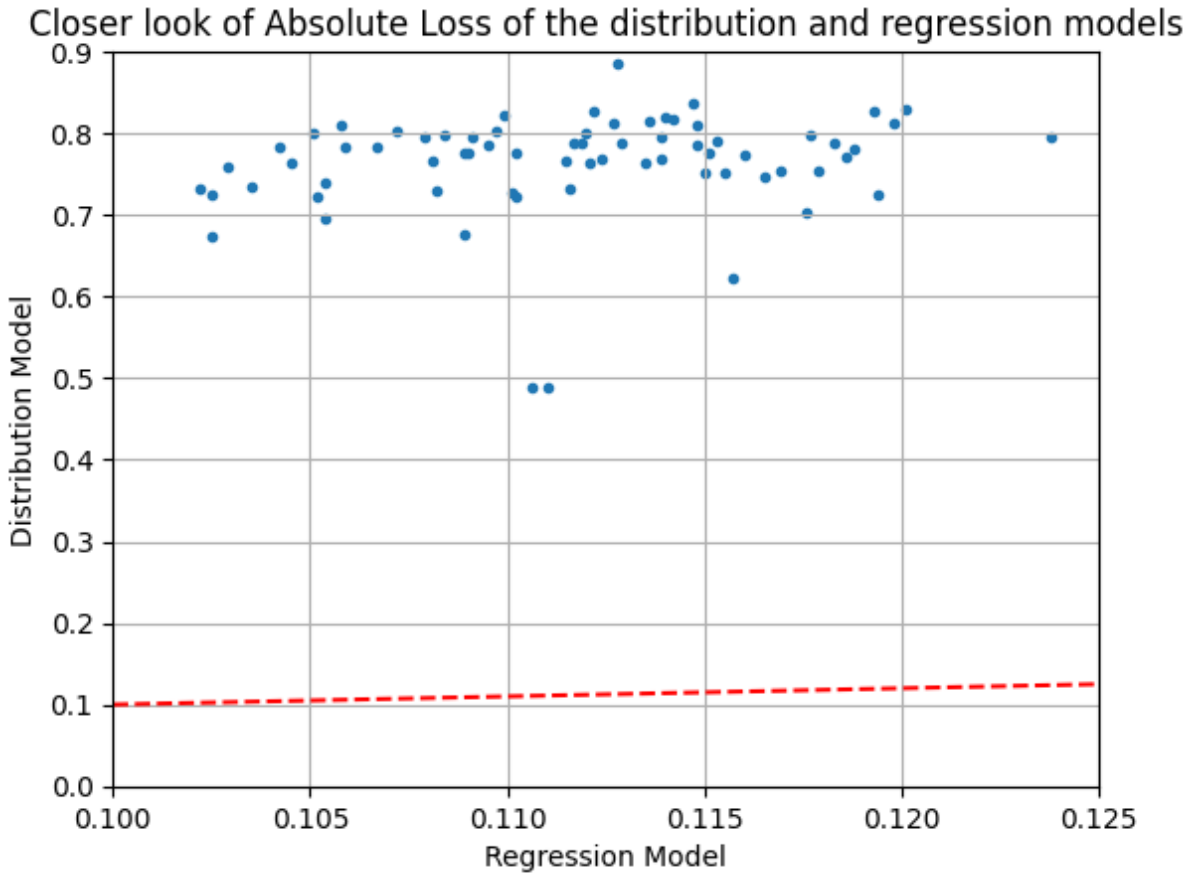
Posteriorly, the correlation of values in GPA was determined and is -0.067.



Loss Function Analysis



Loss Function Analysis



Conclusion

- For high values both models do not describe well the bulk modulus.
- For the Distribution Model only 50% of the predicted values describe a CI of 95%.
- The Regression Model has returned 87% of the values in the CI of 95%.
- By these results, we think that the estimation of the standard deviation is still not viable for serious studies of bulk modulus but further analysis is required.

Thank you for your attention!