StandardScaler (Z-score normalization)- we have one coordinate with X and Y. And these X and Y have values from the given dataset. And sometimes our parameters and target values will have different values and the difference between them will be bigger. In this case, we can use StandardScaler to make our parameters a little same with each other.

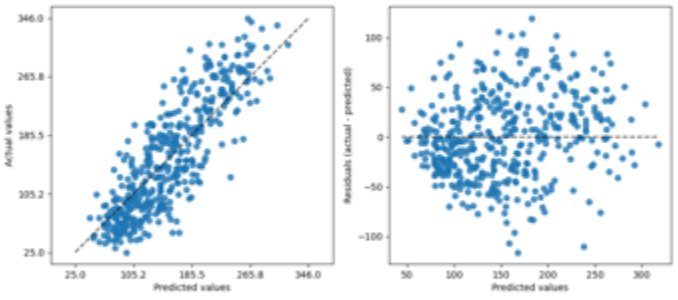
```
z = (x - u) / s      z - new value, x - current value, u - average of col. Values , s - col. Hosilasi.
```

By using this formula, we can find new values which can be a little smaller than the original and it can be around 0 : 0 point of the coordinate and we can use them.

Formula Explanation :

$$\text{Standardization:}$$
$$z = \frac{x-\mu}{\sigma}$$
$$\text{with mean:}$$
$$\mu = \frac{1}{N}\sum_{i=1}^{N}(x_i)$$
$$\text{and standard deviation:}$$
$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i-\mu)^2}$$

How it looks like in graph :



Fit vs Transform vs Fit_transform

Fit - can be use when I want to transform my data value into Standard Deviation with `std = sqrt(mean(abs(x - x.mean())**2))`

In python we can use build in function: numpy.std(

The fit() method helps in fitting the data into a model, transform() method helps in transforming the data into a form that is more suitable for the model. Fit_transform() method, on the other hand, combines the functionalities of both fit() and transform() methods in one step. So let's imagine you are planning to go to a party and you are given a dress code to follow. Firstly you will examine your wardrobe and based on the dress code, you would plan your outfit. Finally, just before your party you would wear the outfit and go.