**Linear Regression**

The simplest form of regression is the linear regression, which assumes that the predictors have a linear relationship with the target variable.

* The input variables are assumed to have a Gaussian distribution.
* Another assumption is that the predictors are not highly correlated with each other (a problem called multi-collinearity).

The linear regression equation can be expressed in the following form:

y = a1x1 + a2x2 + a3x3 + ..... + anxn + b

Where the following is true:

* y is the target variable.
* x1, x2, x3,...xn are the features.
* a1, a2, a3,..., an are the coefficients.
* b is the parameter of the model.

The parameters a and b of the model are selected through the Ordinary least squares (OLS) method. It works by minimizing the sum of squares of residuals (actual value - predicted value).

**Regularized Regression**

Linear regression works by selecting coefficients for each independent variable that minimizes a loss function.

However, if the coefficients are too large, it can lead to model over-fitting on the training dataset.

To overcome this shortcoming, we do regularization which penalizes large coefficients.

The following sections of the guide will discuss the various regularization algorithms.

**Ridge Regression**

Ridge regression is an extension of linear regression where the loss function is modified to minimize the complexity of the model.

This modification is done by adding a penalty parameter that is equivalent to the square of the magnitude of the coefficients.

Loss function = OLS + alpha \* summation (squared coefficient values)

In the above loss function, alpha is the parameter we need to select.

A low alpha value can lead to over-fitting, whereas a high alpha value can lead to under-fitting.

In scikit-learn, a ridge regression model is constructed by using the Ridge class.

**Lasso Regression**

Lasso regression, or the Least Absolute Shrinkage and Selection Operator, is also a modification of linear regression.

In Lasso, the loss function is modified to minimize the complexity of the model by limiting the sum of the absolute values of the model coefficients (also called the l1-norm).

The loss function for Lasso Regression can be expressed as below:

Loss function = OLS + alpha \* summation (absolute values of the magnitude of the coefficients)

In the above loss function, alpha is the penalty parameter we need to select. Using an l1 norm constraint forces some weight values to zero to allow other coefficients to take non-zero values.

**ElasticNet Regression**

ElasticNet combines the properties of both Ridge and Lasso regression. It works by penalizing the model using both the l2-norm and the l1-norm.

In scikit-learn, an ElasticNet regression model is constructed by using the ElasticNet class. The *first line of code* below instantiates the ElasticNet Regression with an alpha value of 0.01. The *second line* fits the model to the training data.