**Machine Learning with Python**

**Week 1 – Introduction to Machine Learning**

Machine learning is the subfield of computer science that gives computers the ability to learn without being explicitly programmed.

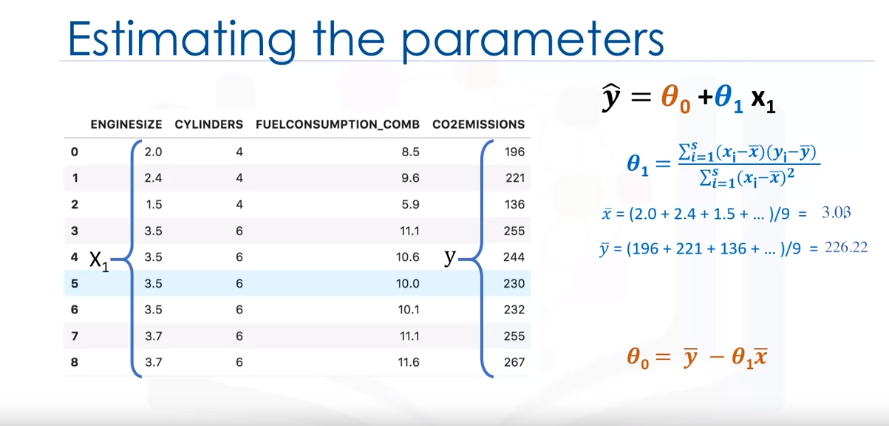
**Week 2 – Supervised Learning – Linear Regression**

**Simple Linear Regression:**

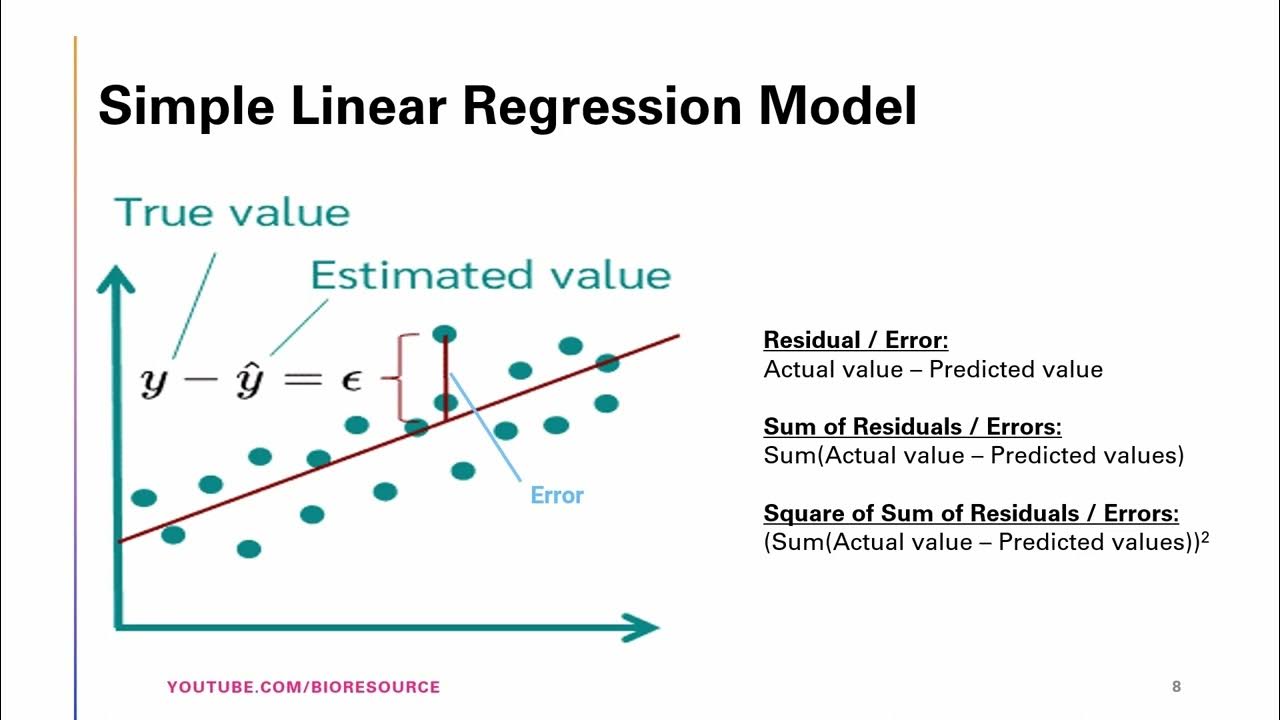
**Yhat = a + bx**

**Residual Error = Yactual - YPredicted**

The mean of all residual errors shows how poorly the line fits with the whole data set. Mathematically it can be shown by the equation Mean Squared Error, shown as MSE. Our objective is to find a line where the meaning of all these errors is minimized. In other words, the mean error of the prediction using the fit line should be minimized. Let's reword it more technically. The objective of linear regression is to minimize this MSE equation and to minimize it, we should find the best parameters; a and b.



As mentioned before, **Coefficient** and **Intercept** in the simple linear regression, are the parameters of the fit line. Given that it is a simple linear regression, with only 2 parameters, and knowing that the parameters are the intercept and slope of the line, sklearn can estimate them directly from our data. Notice that all of the data must be available to traverse and calculate the parameters

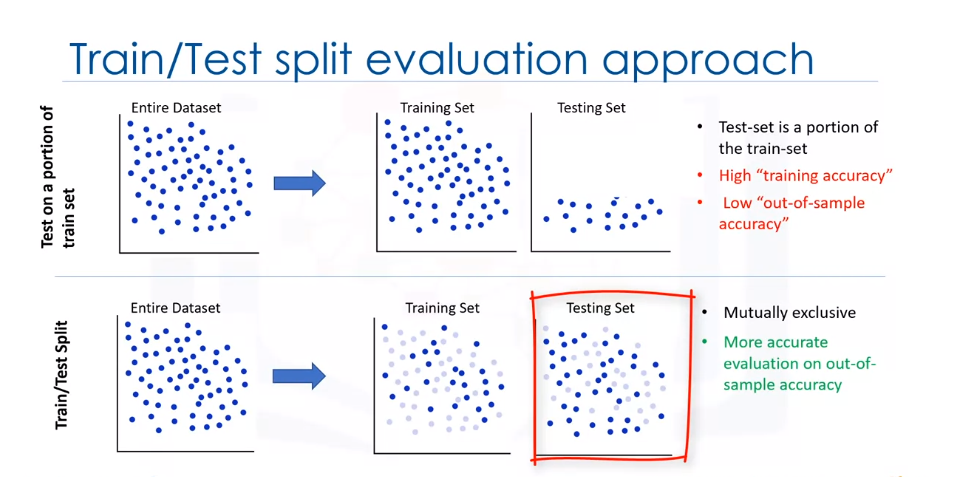


**Training Accuracy:**

Training accuracy is the percentage of correct predictions that the model makes when using the test dataset. However, a high training accuracy isn't necessarily a good thing. For instance, having a high training accuracy may result in an over-fit the data. This means that the model is overly trained to the dataset, which may capture noise and produce a non-generalized model.

**Out-of-sample Accuracy:**

Out-of-sample accuracy is the percentage of correct predictions that the model makes on data that the model has not been trained on. Doing a train and test on the same dataset will most likely have low out-of-sample accuracy due to the likelihood of being over-fit. It's important that our models have high out-of-sample accuracy because the purpose of our model is, of course, to make correct predictions on unknown data.



**Creating train and test dataset**

Train/Test Split involves splitting the dataset into training and testing sets that are mutually exclusive. After which, you train with the training set and test with the testing set. This will provide a more accurate evaluation on out-of-sample accuracy because the testing dataset is not part of the dataset that have been used to train the model. Therefore, it gives us a better understanding of how well our model generalizes on new data.

This means that we know the outcome of each data point in the testing dataset, making it great to test with! Since this data has not been used to train the model, the model has no knowledge of the outcome of these data points. So, in essence, it is truly an out-of-sample testing.

Let's split our dataset into train and test sets. 80% of the entire dataset will be used for training and 20% for testing. We create a mask to select random rows using **np.random.rand()** function:

**Evaluation Metrics in Regression Models:**

Error: It is the measure of how far the data point of Y is from the fitted regression line (Yhat).

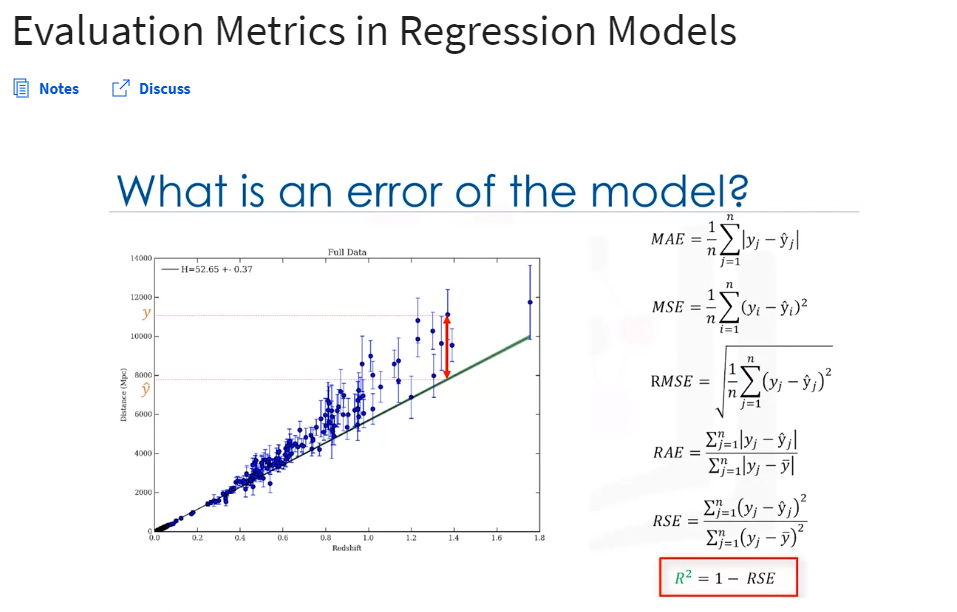
We compare the actual values and predicted values to calculate the accuracy of a regression model. Evaluation metrics provide a key role in the development of a model, as it provides insight to areas that require improvement.

There are different model evaluation metrics, lets use MSE here to calculate the accuracy of our model based on the test set:

* Mean Absolute Error: It is the mean of the absolute value of the errors. This is the easiest of the metrics to understand since it’s just average error.
* Mean Squared Error (MSE): Mean Squared Error (MSE) is the mean of the squared error. It’s more popular than Mean Absolute Error because the focus is geared more towards large errors. This is due to the squared term exponentially increasing larger errors in comparison to smaller ones.
* Root Mean Squared Error (RMSE).
* R-squared is not an error, but rather a popular metric to measure the performance of your regression model. It represents how close the data points are to the fitted regression line. The higher the R-squared value, the better the model fits your data. The best possible score is 1.0 and it can be negative (because the model can be arbitrarily worse).

MAE(Mean Absolute Error):

MSE(Mean Squared Error):

RMSE(Root Mean Squared Error):

R-Squared:

Adjusted R-squared:

RAE(Relative Absolute Error):

RSE(Relative Squared Error):

**Multiple Linear Regression/Model**

In reality, there are multiple variables that impact the co2emission(dependent variable). When more than one independent variable is present, the process is called multiple linear regression. An example of multiple linear regression is predicting co2emission using the features (independent variables) FUELCONSUMPTION\_COMB, EngineSize and Cylinders of cars and more. The good thing here is that multiple linear regression model is the extension of the simple linear regression model.

There can be two methods to optimize the parameters of the multiple linear regression model:

1. Ordinary Least Square method (takes a long time to optimize the parameters vector if the data is too large (data >=10k). It is a good practice to go with OLS if data is less than < 10K.
2. Optimization Approach: it is used for the larger set of data. Technique can be Gradient Descent, Stochastic Gradient Descent, Newton’s Method, etc.

As mentioned before, **Coefficient** and **Intercept** are the parameters of the fitted line. Given that it is a multiple linear regression model with 3 parameters and that the parameters are the intercept and coefficients of the hyperplane, sklearn can estimate them from our data. Scikit-learn uses plain Ordinary Least Squares method to solve this problem.

**Ordinary Least Squares (OLS)**

OLS is a method for estimating the unknown parameters in a linear regression model. OLS chooses the parameters of a linear function of a set of explanatory variables by minimizing the sum of the squares of the differences between the target dependent variable and those predicted by the linear function. In other words, it tries to minimize the sum of squared errors (SSE) or mean squared error (MSE) between the target variable (y) and our predicted output (^y�^) over all samples in the dataset.

OLS can find the best parameters using of the following methods:

* Solving the model parameters analytically using closed-form equations
* Using an optimization algorithm (Gradient Descent, Stochastic Gradient Descent, Newton’s Method, etc.)

**Explained variance regression score:**   
Let ^y�^ be the estimated target output, y the corresponding (correct) target output, and Var be the Variance (the square of the standard deviation). Then the explained variance is estimated as follows:

explainedVariance(y, ^y)=1−Var{y−^y} / Var{y}

The best possible score is 1.0, the lower values are worse.

**---------Explained Variance is equal to the R-squared.-----------**

**Week 3 – Supervised Learning - Classification**

It can be thought of as a means of categorizing or classifying some unknown items into a discrete set of classes.

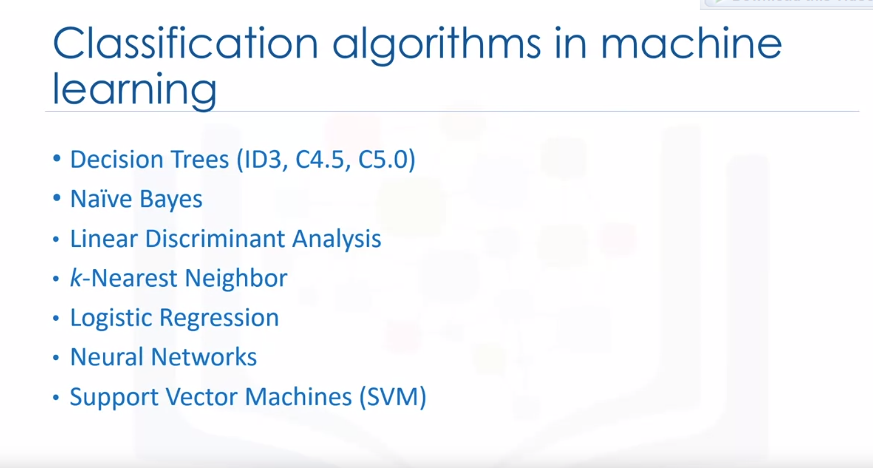
Target Attribute/variable is a categorical variable.

Classification determines the class label for an unlabeled test case.

Data classification has several applications in a wide variety of industries.

Essentially, many problems can be expressed as associations between feature and target variables, especially when labelled data is available.

This provides a broad range of applicability for classification. For example, classification can be used for email filtering, speech recognition, handwriting recognition, biometric identification, document classification and much more.

Here we have the types of classification algorithms and machine learning.