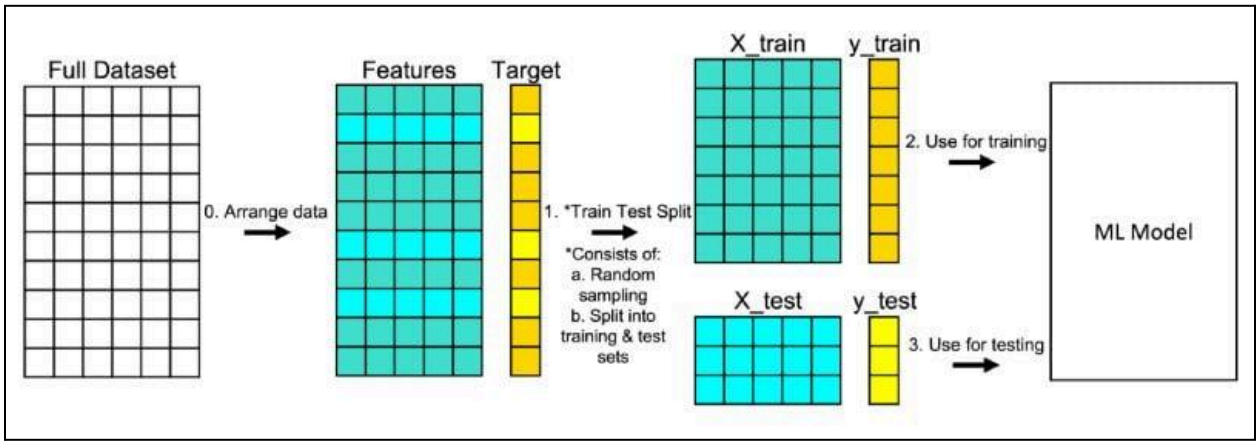

42/100
03 FEB 24| DAY - 42| MACHINE LEARNING
#100DAYSOFDATA SCIENCE
PYTHON | NUMPY | PANDAS | MATPLOTLIB | SEABORN | SQL | STATS | MACHINE LEARNING |

Train Test Split

- A train test split is when you split your data into a training set and a testing set.
- The training set is used for training the model, and the testing set is used to test your model. This allows you to train your models on the training set, and then test their accuracy on the unseen testing set.
- There are a few different ways to do a train test split, but the most common is to simply split your data into two sets.
- For example 80% for training and 20% for testing.
- This ensures that both sets are representative of the entire dataset, and gives you a good way to measure the accuracy of your models.

Train Test Split Procedure



1. ARRANGE THE DATA

Make sure your data is arranged into a format acceptable for train test split. In scikit-learn, this consists of separating your full data set into “Features” and “Target.”

2. SPLIT THE DATA

Split the data set into two pieces — a training set and a testing set. This consists of random sampling without replacement about 75 percent of the rows (you can vary this) and putting them into your training set. The remaining 25 percent is put into your test set. Note that the colors in “Features” and “Target” indicate where their data will go (“X_train,” “X_test,” “y_train,” “y_test”) for a particular train test split.

3. TRAIN THE MODEL

Train the model on the training set. This is “X_train” and “y_train” in the image.

4. TEST THE MODEL

Test the model on the testing set (“X_test” and “y_test” in the image) and evaluate the performance.

train_test_split Parameters	Description	Options/Values	Default
test_size	Size of the testing subset	Float (0.0 to 1.0) or int	0.25
train_size	Size of the training subset	Float (0.0 to 1.0) or int	None
random_state	Random seed for reproducibility	int or RandomState instance	None
shuffle	Whether to shuffle the data before splitting	bool	True
stratify	Array-like or None. If not None, split data in a stratified fashion	array-like or None	None