Splitting Data Into Train And Test

- When you are working on a model and you want to train it, you have a dataset. But after training, we have to test the model on some test dataset. For this, you will need a dataset that is different from the training set you used earlier. But it might not always be possible to have so much data during the development phase. In such cases, the solution is to split the dataset into two sets, one for training and the other for testing.
- But the question is, how do you split the data?
- For time-series data, the sequence of values is important. A simple method that we can use is to split the ordered dataset into train and test datasets. The code below calculates the index of the split point and separates the data into the training datasets with 65% of the observations that we can use to train our model, leaving the remaining 30% for testing the model.

```
training_size=int(len(data_oil)*0.65)
test_size=len(data_oil)-training_size
train_data,test_data=data_oil[0:training_size,:],data_oil[training_size:len(data_oil),:1]
```

The size of the train and test data after splitting
The size of train and test data after splitting

```
training_size,test_size

(5340, 2876)

train_data.shape

(5340, 1)
```

If you want to create a prediction model, you typically use a table or a view that contains historical data. If you decide to use the Easy Mining procedures for classification or regression, you might want to split this table into the following disjoint data sets:

- One data set to train the prediction model
- One data set to test the prediction model