

In this article, we will train a model using the linear regression algorithm to predict house prices, after exploring the dataset, performing exploratory data analysis (EDA) and normalizing the features

### 1. Import Necessary Libraries

```
from sklearn.linear_model import LinearRegression  
  
from sklearn.metrics import mean_squared_error, r2_score
```

### 2. Initialize the Linear Regression Model

```
# Initialize the Linear Regression model  
  
regression = LinearRegression()
```

### 3. Fit the Model to the Training Data

```
# Fit the model to the scaled training data  
  
regression.fit(X_train_norm, y_train)
```

In a linear regression model, the equation for the predicted output ( $y_{pred}$ ) given the input features ( $x$ ) is:

$$y_{pred} = mx + c$$

Here:

- $m$  is the slope or coefficient of the regression line
- $c$  is the y-intercept

When you fit a linear regression model using the `LinearRegression` class in scikit-learn, the `coef_` attribute gives you the  $m$  values (slopes), and the `intercept_` attribute gives you the  $c$  value (intercept).

Here's how you can access these values after fitting the model:

```
# Retrieve the slope (m) values  
  
slope_values = regression.coef_
```

```
# Retrieve the y-intercept (c) value
```

```
intercept_value = regression.intercept_
```

Here, in this model we'll get 8 slope values and `slope_values` will contain an array of slopes corresponding to each feature in your dataset, and `intercept_value` will be a single value representing the y-intercept.