



# Machine Learning - Multiple Regression

[< Previous](#)[Next >](#)

## Multiple Regression

Multiple regression is like linear regression, but with more than one independent value, meaning that we try to predict a value based on **two or more** variables.

Take a look at the data set below, it contains some information about cars.

| Car        | Model      | Volume | Weight | CO2 |
|------------|------------|--------|--------|-----|
| Toyota     | Aygo       | 1000   | 790    | 99  |
| Mitsubishi | Space Star | 1200   | 1160   | 95  |
| Skoda      | Citigo     | 1000   | 929    | 95  |
| Fiat       | 500        | 900    | 865    | 90  |
| Mini       | Cooper     | 1500   | 1140   | 105 |
| VW         | Up!        | 1000   | 929    | 105 |
| Skoda      | Fabia      | 1400   | 1109   | 90  |
| Mercedes   | A-Class    | 1500   | 1365   | 92  |
| Ford       | Fiesta     | 1500   | 1112   | 98  |
| Audi       | A1         | 1600   | 1150   | 99  |
| Hyundai    | I20        | 1100   | 980    | 90  |

We can predict the CO2 emission of a car based on the size of the engine, but with multiple regression we can throw in more variables, like the weight of the car, to make the prediction more accurate.

---

## How Does it Work?

In Python we have modules that will do the work for us. Start by importing the Pandas module.

```
import pandas
```

The Pandas module allows us to read csv files and return a DataFrame object.

The file is meant for testing purposes only, you can download it here: [cars.csv](#)

```
df = pandas.read_csv("cars.csv")
```

Then make a list of the independent values and call this variable **X**.

Put the dependent values in a variable called **y**.

```
X = df[['Weight', 'Volume']]  
y = df['CO2']
```

**Tip:** It is common to name the list of independent values with a upper case X, and the list of dependent values with a lower case y.

We will use some methods from the sklearn module, so we will have to import that module as well:

```
from sklearn import linear_model
```

From the sklearn module we will use the `LinearRegression()` method to create a linear regression object.

This object has a method called `fit()` that takes the independent and dependent values as parameters and fills the regression object with data that describes the relationship:

```
regr = linear_model.LinearRegression()
regr.fit(X, y)
```

Now we have a regression object that are ready to predict CO2 values based on a car's weight and volume:

```
#predict the CO2 emission of a car where the weight is 2300kg, and the volume
is 1300ccm:
predictedCO2 = regr.predict([[2300, 1300]])
```

## Example

See the whole example in action:

```
import pandas
from sklearn import linear_model

df = pandas.read_csv("cars.csv")

X = df[['Weight', 'Volume']]
y = df['CO2']

regr = linear_model.LinearRegression()
regr.fit(X, y)

#predict the CO2 emission of a car where the weight is 2300kg, and the
volume is 1300ccm:
predictedCO2 = regr.predict([[2300, 1300]])

print(predictedCO2)
```

## Result:

```
[107.2087328]
```

[Run example »](#)

We have predicted that a car with 1.3 liter engine, and a weight of 2300 kg, will release approximately 107 grams of CO2 for every kilometer it drives.

## Coefficient

The coefficient is a factor that describes the relationship with an unknown variable.

Example: if  $x$  is a variable, then  $2x$  is  $x$  two times.  $x$  is the unknown variable, and the number  $2$  is the coefficient.

In this case, we can ask for the coefficient value of weight against CO2, and for volume against CO2. The answer(s) we get tells us what would happen if we increase, or decrease, one of the independent values.

## Example

Print the coefficient values of the regression object:

```
import pandas
from sklearn import linear_model

df = pandas.read_csv("cars.csv")

X = df[['Weight', 'Volume']]
y = df['CO2']

regr = linear_model.LinearRegression()
regr.fit(X, y)

print(regr.coef_)
```

## Result:

```
[0.00755095 0.00780526]
```

[Run example »](#)

## Result Explained

The result array represents the coefficient values of weight and volume.

Weight: 0.00755095

Volume: 0.00780526

These values tells us that if the weight increases by 1g, the CO2 emission increases by 0.00755095g.

And if the engine size (Volume) increases by 1 ccm, the CO2 emission increases by 0.00780526 g.

I think that is a fair guess, but let test it!

We have already predicted that if a car with a 1300ccm engine weighs 2300kg, the CO2 emission will be approximately 107g.

What if we increase the weight with 1000kg?

## Example

Copy the example from before, but change the weight from 2300 to 3300:

```
import pandas
from sklearn import linear_model

df = pandas.read_csv("cars.csv")

X = df[['Weight', 'Volume']]
y = df['CO2']

regr = linear_model.LinearRegression()
regr.fit(X, y)

predictedCO2 = regr.predict([[3300, 1300]])

print(predictedCO2)
```

## Result:

```
[114.75968007]
```

[Run example »](#)

We have predicted that a car with 1.3 liter engine, and a weight of 3300 kg, will release approximately 115 grams of CO2 for every kilometer it drives.

Which shows that the coefficient of 0.00755095 is correct:

$$107.2087328 + (1000 * 0.00755095) = 114.75968$$

[< Previous](#)[Next >](#)

## COLOR PICKER



## HOW TO

- Tabs
- Dropdowns
- Accordions
- Side Navigation
- Top Navigation
- Modal Boxes
- Progress Bars
- Parallax
- Login Form
- HTML Includes
- Google Maps
- Range Sliders
- Tooltips
- Slideshow

[Filter List](#)

[Sort List](#)

SHARE



## CERTIFICATES

[HTML](#)

[CSS](#)

[JavaScript](#)

[SQL](#)

[Python](#)

[PHP](#)

[jQuery](#)

[Bootstrap](#)

[XML](#)

[Read More »](#)

---

[REPORT ERROR](#)

[PRINT PAGE](#)

[FORUM](#)

[ABOUT](#)

---

## Top Tutorials

[HTML Tutorial](#)

[CSS Tutorial](#)

[JavaScript Tutorial](#)

[How To Tutorial](#)

[SQL Tutorial](#)

[Python Tutorial](#)

[W3.CSS Tutorial](#)

[Bootstrap Tutorial](#)  
[PHP Tutorial](#)  
[jQuery Tutorial](#)  
[Java Tutorial](#)  
[C++ Tutorial](#)

## Top References

[HTML Reference](#)  
[CSS Reference](#)  
[JavaScript Reference](#)  
[SQL Reference](#)  
[Python Reference](#)  
[W3.CSS Reference](#)  
[Bootstrap Reference](#)  
[PHP Reference](#)  
[HTML Colors](#)  
[jQuery Reference](#)  
[Java Reference](#)  
[Angular Reference](#)

## Top Examples

[HTML Examples](#)  
[CSS Examples](#)  
[JavaScript Examples](#)  
[How To Examples](#)  
[SQL Examples](#)  
[Python Examples](#)  
[W3.CSS Examples](#)  
[Bootstrap Examples](#)  
[PHP Examples](#)  
[jQuery Examples](#)  
[Java Examples](#)  
[XML Examples](#)

## Web Certificates

[HTML Certificate](#)  
[CSS Certificate](#)  
[JavaScript Certificate](#)  
[SQL Certificate](#)  
[Python Certificate](#)  
[jQuery Certificate](#)  
[PHP Certificate](#)  
[Bootstrap Certificate](#)  
[XML Certificate](#)

[Get Certified »](#)

---

W3Schools is optimized for learning, testing, and training. Examples might be simplified to improve reading and basic understanding. Tutorials, references, and examples are constantly reviewed to avoid errors, but we cannot warrant full correctness of all content. While using this site, you agree to have read and accepted our terms of use, cookie and privacy policy. Copyright 1999-2020 by Refsnes Data. All Rights Reserved.

Powered by W3.CSS.



