

How do you perform logarithmic regression in Python step-by-step?

Authored by
stats writer

April 27, 2024

RECOMMENDED CITATION

stats writer (2024). *How do you perform logarithmic regression in Python step-by-step?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=140181>

Logarithmic regression is a statistical method used to fit a curve to a set of data points that follow a logarithmic relationship. This technique is commonly used in data analysis and machine learning to model non-linear relationships between variables.

Performing logarithmic regression in Python involves the following steps:

1. Import the necessary libraries: The first step is to import the required libraries such as numpy, pandas, and matplotlib. These libraries provide functions and tools for data manipulation and visualization.
2. Load the data: Next, the data set needs to be loaded into the Python environment. This can be done using the `read_csv()` function from the pandas library.
3. Visualize the data: It is always helpful to visualize the data before performing any analysis. Plotting the data using a scatter plot can help identify any potential logarithmic relationships.
4. Define the independent and dependent variables: In logarithmic regression, the independent variable is the one that is being used to predict the dependent variable. Therefore, it is important to identify and define these variables in the data set.
5. Transform the data: Since logarithmic regression models a logarithmic relationship, it is necessary to transform the data into a logarithmic scale. This can be done using the `log()` function from numpy.
6. Fit the model: Using the linear regression function from the statsmodels library, the logarithmic regression model can be fitted to the transformed data.
7. Evaluate the model: After fitting the model, it is important to evaluate its performance. This can be done by calculating the R-squared value, which measures the goodness of fit of the model.
8. Plot the regression line: Finally, the regression line can be plotted on the scatter plot to visualize how well it fits the data.

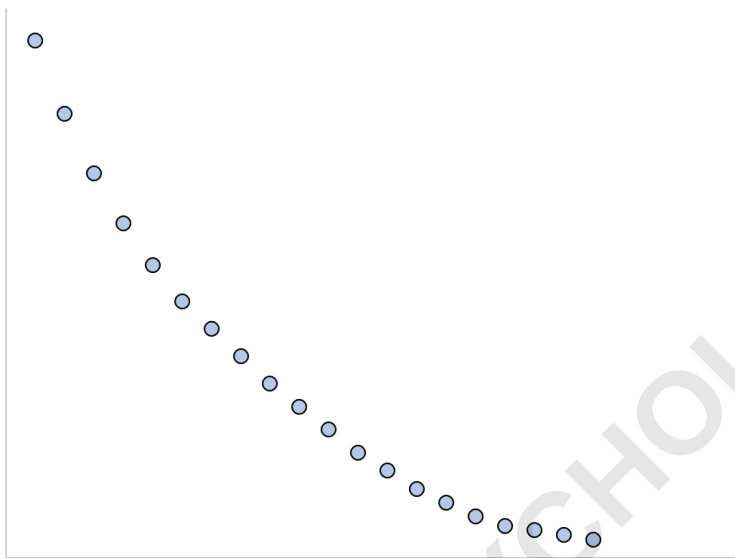
In conclusion, logarithmic regression in Python involves importing libraries, loading and visualizing the data, transforming it into a logarithmic scale, fitting the model, and evaluating its performance. It is a useful tool for analyzing non-linear relationships and making predictions based on the data.

Logarithmic Regression in Python (Step-by-Step)

Logarithmic regression is a type of regression used to

model situations where growth or decay accelerates rapidly at first and then slows over time.

For example, the following plot demonstrates an example of logarithmic decay:



For this type of situation, the relationship between a predictor variable and a could be modeled well using logarithmic regression.

The equation of a logarithmic regression model takes the following form:

$$y = a + b \cdot \ln(x)$$

where:

y: The response variable
x: The predictor variable
a, b: The regression coefficients that describe the relationship between x and y

The following step-by-step example shows how to perform logarithmic regression in Python.

Step 1: Create the Data

First, let's create some fake data for two variables: x and y :

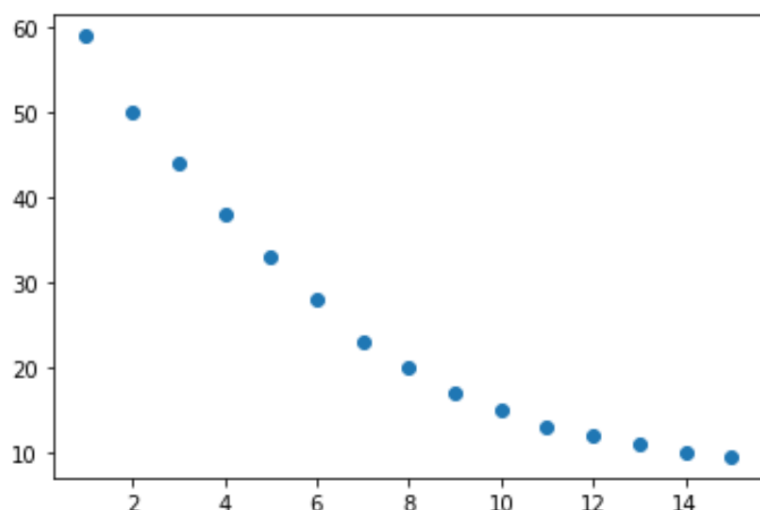
```
import numpy as np
x = np.arange(1, 16, 1)
y = np.array()
```

Step 2: Visualize the Data

Next, let's create a quick to visualize the relationship between x and y :

```
import matplotlib.pyplot as plt

plt.scatter(x, y)
plt.show()
```



From the plot we can see that there exists a logarithmic decay pattern between the two variables. The value of the response variable, y , decreases rapidly at first and then slows over time.

Step 3: Fit the Logarithmic Regression Model

Next, we'll use the `polyfit()` function to fit a logarithmic regression model, using the natural log of x as the predictor variable and y as the response variable:

```
#fit the model
```

```
fit = np.polyfit(np.log(x), y, 1)
```

```
#view the output of the model
```

```
print(fit)
```

We can use the coefficients in the output to write the following fitted logarithmic regression equation:

$$y = 63.0686 - 20.1987 * \ln(x)$$

We can use this equation to predict the response variable, y , based on the value of the predictor variable, x . For example, if $x = 12$, then we would predict that y would be 12.87:

$$y = 63.0686 - 20.1987 * \ln(12) = 12.87$$

Bonus: Feel free to use this online to automatically compute the logarithmic regression equation for a given predictor and response variable.