TASK

Capstone Project II —
Machine Learning V

Visit our website

Introduction

WELCOME TO THE MACHINE LEARNING CAPSTONE TASK!

In the previous tasks, we assumed that the real relationship between the explanatory variables and the response variable is linear. This assumption is not always true, as you will see in the **polynomial-regression.py** file. Thus, in this task we will introduce polynomial regression; a special case of multiple linear regression that adds terms with degrees greater than one to the model.

Get in touch

Connect for support

Remember that with our courses, you're not alone! You can contact an expert code reviewer to get support on any aspect of your course.

The best way to get help is to login to Discord at **https://discord.com/invite/hyperdev** where our specialist team is ready to support you.

Our team is happy to offer you support that is tailored to your individual career or education needs. Do not hesitate to ask a question or for additional support!

A note from the

HyperionDev Team

Over time, there have been several myths surrounding Machine Learning such as "Machine Learning is just summarising data" or "Learning algorithms just discover correlations between events", but probably one of the most prominent is that "Machine Learning can't predict unseen events". Well, let's debunk this.

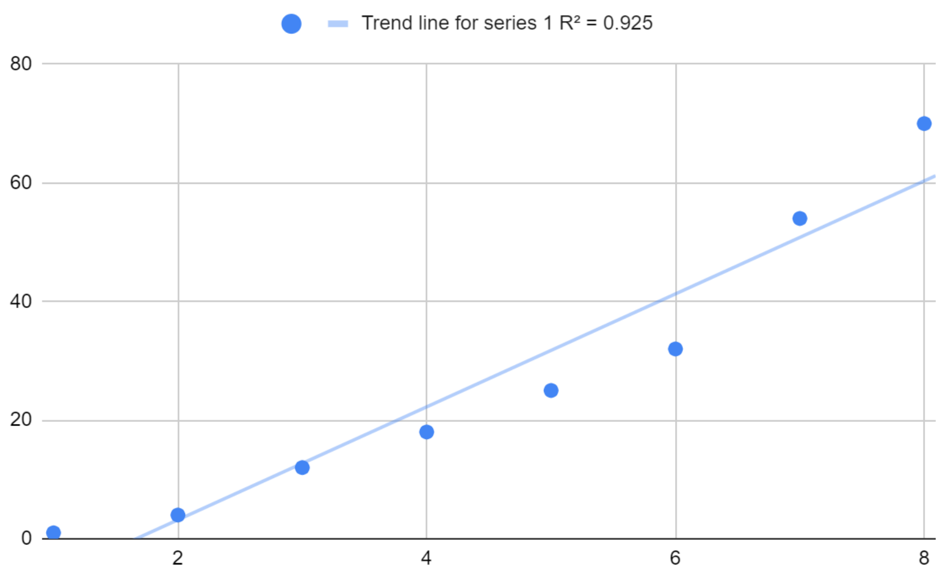
If something has never happened before, its predicted probability must be zero, right? What else could it be? On the contrary, machine learning is the art of predicting rare events with high accuracy. If A is one of the causes of B and B is one of the causes of C, A can lead to C, even if we've never seen it happen before. For example, every day, spam filters correctly flag freshly concocted spam emails.

POLYNOMIAL REGRESSION

Polynomials are mathematical expressions that contain sums and differences of terms, which have variables with whole numbered positive powers. For example, $x^2 + 2x - 4$ is a polynomial of "order" or "degree" 2 (i.e. the biggest power of x is 2) and has 3 terms, and $6x^3 - 3x^2 + 2x + 1$ is a polynomial of order 3 and has 4 terms.

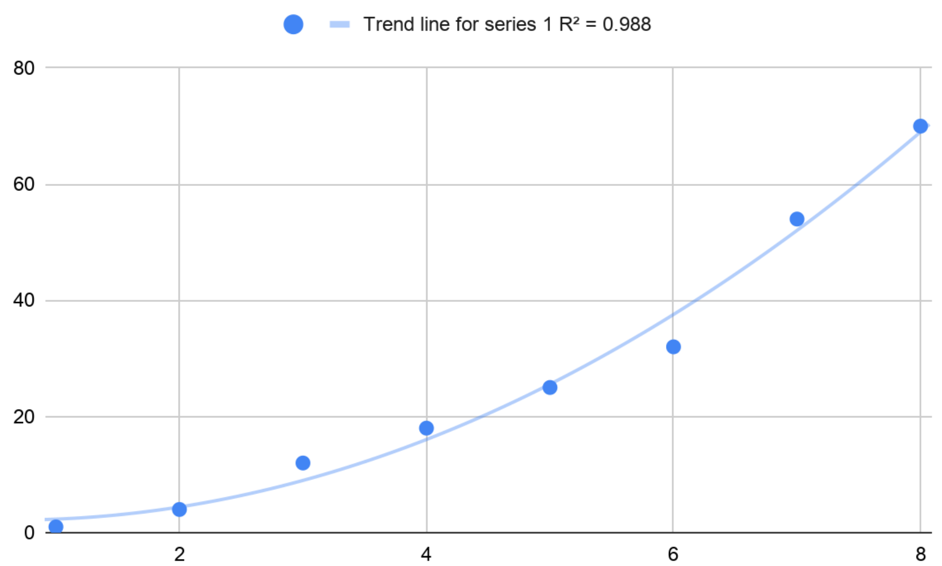
Like linear regression, polynomial regression is used to find a relationship between two variables. Indeed, linear regression is actually a special case of polynomial regression, $2x + 1$is a polynomial of order 1 (since x is to the power of one, so we don't have to write the exponent) with two terms, which is in the form of a straight line, $mx + c$. The general form for a polynomial expression is $a_n x^n + a_{n-1}x^{n-1} + ... + a_2 x^2 + a_1 x + a_0$. Note that the last two terms here are the same thing as $mx + c$.

Sometimes, the relationship between two variables isn't just a simple linear relationship. Have a look at the graph below:



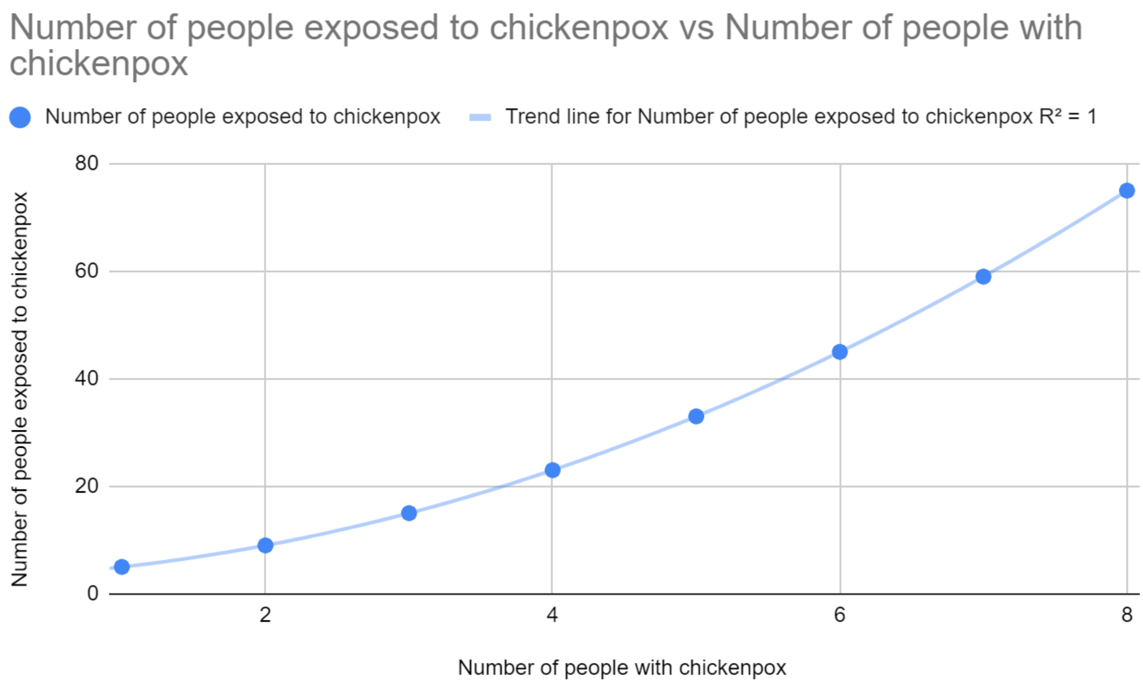
As you can see, a straight line doesn't really fit the data very well. This is because there is a loosely exponential relationship between the x and y variables. We can see this because the slope of the data gets steeper as we move from left to right (although not as quickly as a true exponential curve).

Now let's look at the line of best fit when its formula is a polynomial of order 2:



This line fits much better with the current data. We can see that because the R^2 number is closer to 1. This means that the formula for the line of best fit is a polynomial of order 2.

Let us put this example in context. Let's say that for every person who has chickenpox (x), $x^2 + x + 3$ people get exposed to the virus. That means that one infected person results in 5 people being exposed ($1^2 + 1 + 3 = 5$) and two infected results in 9 being exposed, ($2^2 + 2 + 3 = 9$), while three results in 15, ($3^2 + 3 + 3 = 15$). The exponent 2 in the formula shows a polynomial relationship (when it is to the power of 2 it is referred to as "quadratic"), and so the graph of the relationship between people with chickenpox and the number of people exposed will look like this:



A note from Riaz...

Sorry to interrupt, but did you know that cutting-edge startups, as well as established tech companies and Universities, are increasingly finding new, novel, and exciting ways to apply powerful machine learning tools such as neural networks to existing problems in many different industries.

Cornell University, for example, is working on an algorithm to identify whales in the ocean based on audio recordings so that ships can avoid hitting them. Also, Oregon State University is working on software that will determine which bird species is/are on a given audio recording collected in field conditions.

Compulsory Task

Follow these steps:

- Read the example file.
- Try to think of a relationship you can model and create a new Python file in this folder called **poly.py**.
- Inside **poly.py**, identify a relationship, and use Polynomial regression to train, predict, and plot your results.

Rate us

Share your thoughts

HyperionDev strives to provide internationally-excellent course content that helps you achieve your learning outcomes.

Think that the content of this task, or this course as a whole, can be improved or think we've done a good job?

Click here to share your thoughts anonymously.