02: Power Series

Chapter Goal: To build the visual intuition of how the approximation process works, layer by layer, before we get into the actual calculation of the Taylor Series.

---

1. Core Idea: The "Lego Bricks" of a Polynomial

- **Power Series:** Is another, more general name for a Taylor Series.
- **Definition:** A Power Series is a polynomial function that can potentially be infinitely long.
- **The "Lego Bricks":** It consists of a sum of terms with progressively increasing powers of `x`:

> $g(x) = c_0 + c_1x + c_2x^2 + c_3x^3 + \dots$
> *(In the video, the coefficients are called `a, b, c, d`, but we will use $c_0, c_1, \dots$ to be more general).*

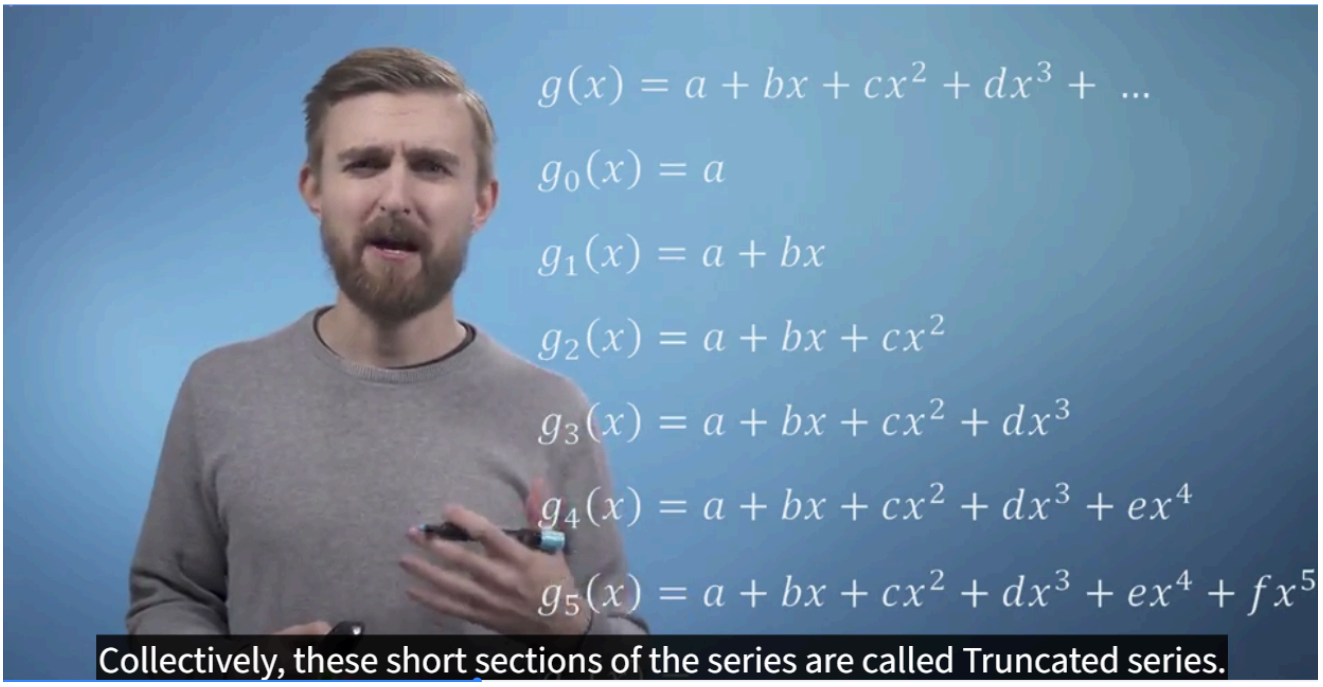
- **Goal:** We will build our "imitator" function by adding these "Lego bricks" one by one, where each addition will make the approximation better.

---

2. "Chopping" the Series (Truncated Series)

In many practical applications, we don't need an infinite series. We often only use the first few terms.

- "Chopping" a series after a few terms is called a **Truncated Series**.
- We name each "chop" based on the highest power we use.



---

3. The Layer-by-Layer Approximation Process (Visual)

Imagine we want to imitate a complex red curve around one specific point.

Layer 1: The 0th-Order Approximation (c_0)

- **Formula:** $P_0(x) = c_0$
- **Shape:** A horizontal line.
- **The Task:** To be the most basic approximation, this line must pass through the point we want to imitate.
- **Result:** This is a very rough approximation, but at least its value is correct at that one specific point.

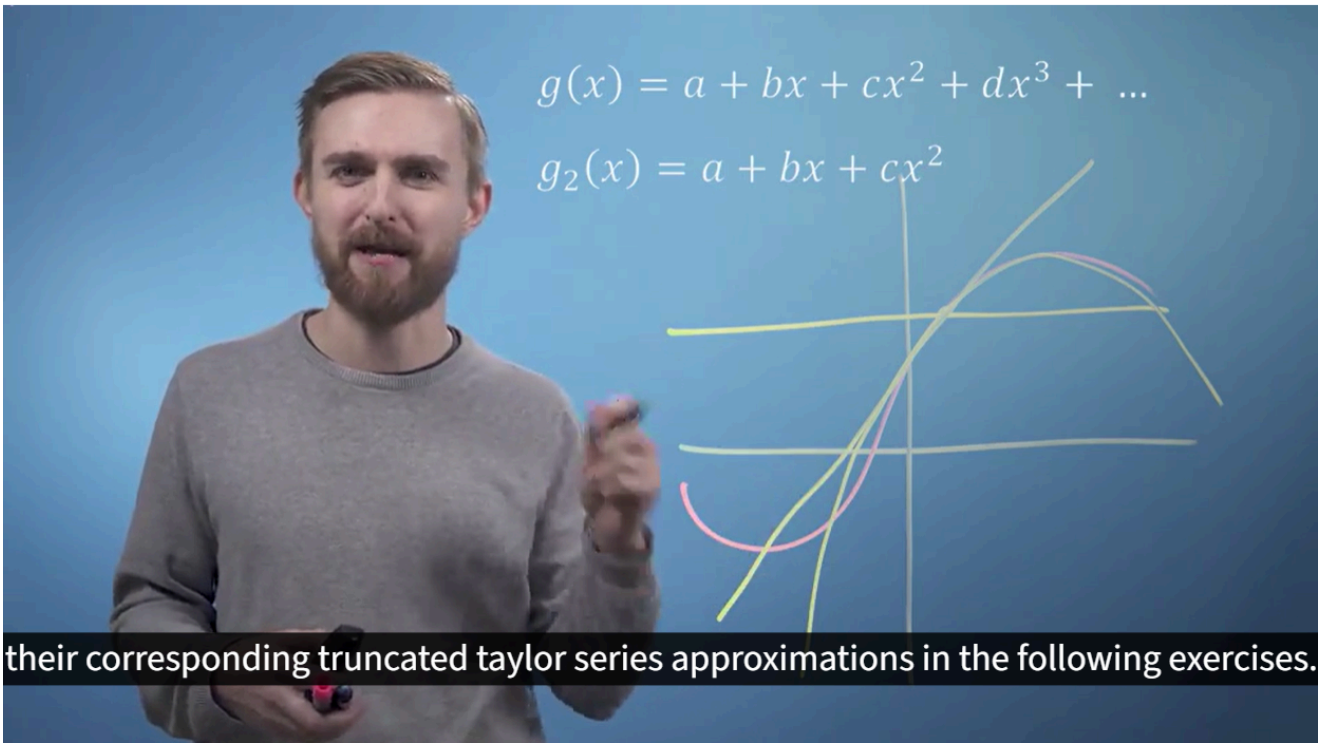
Layer 2: The 1st-Order Approximation ($c_0 + c_1x$)

- **Formula:** $P_1(x) = c_0 + c_1x$
- **Shape:** A slanted straight line.
- **The Task:** To be a better approximation, this line must not only pass through the point, but it must also have the same **slope** as the red curve at that point. This is the tangent line.
- **Result:** The approximation becomes much better in the area very close to the point of contact.

Layer 3: The 2nd-Order Approximation ($c_0 + c_1x + c_2x^2$)

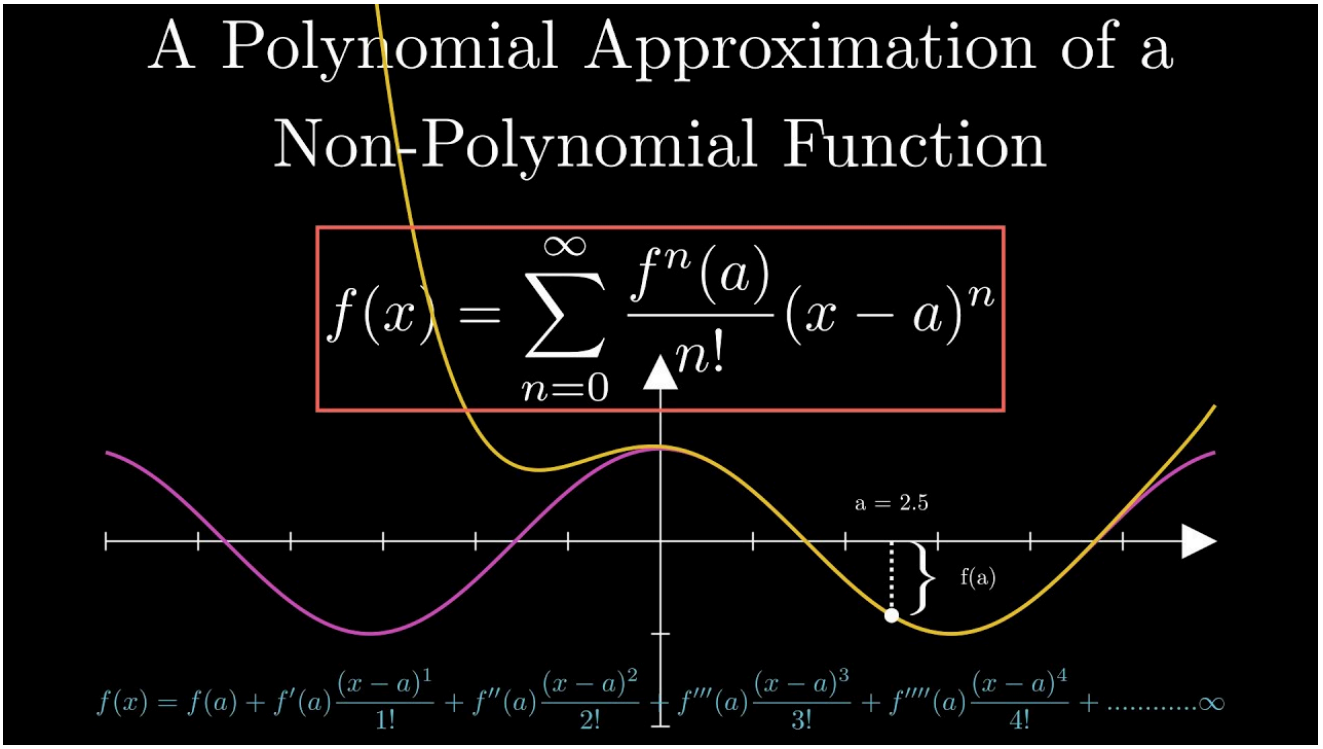
- **Formula:** $P_2(x) = c_0 + c_1x + c_2x^2$
- **Shape:** A parabola.

- **The Task:** To be an even better approximation, this parabola must not only have the same value and slope, but it must also have the same **curvature** as the red curve at that point.
- **Result:** This parabola will "hug" the red curve much more tightly and for longer than the straight line did.



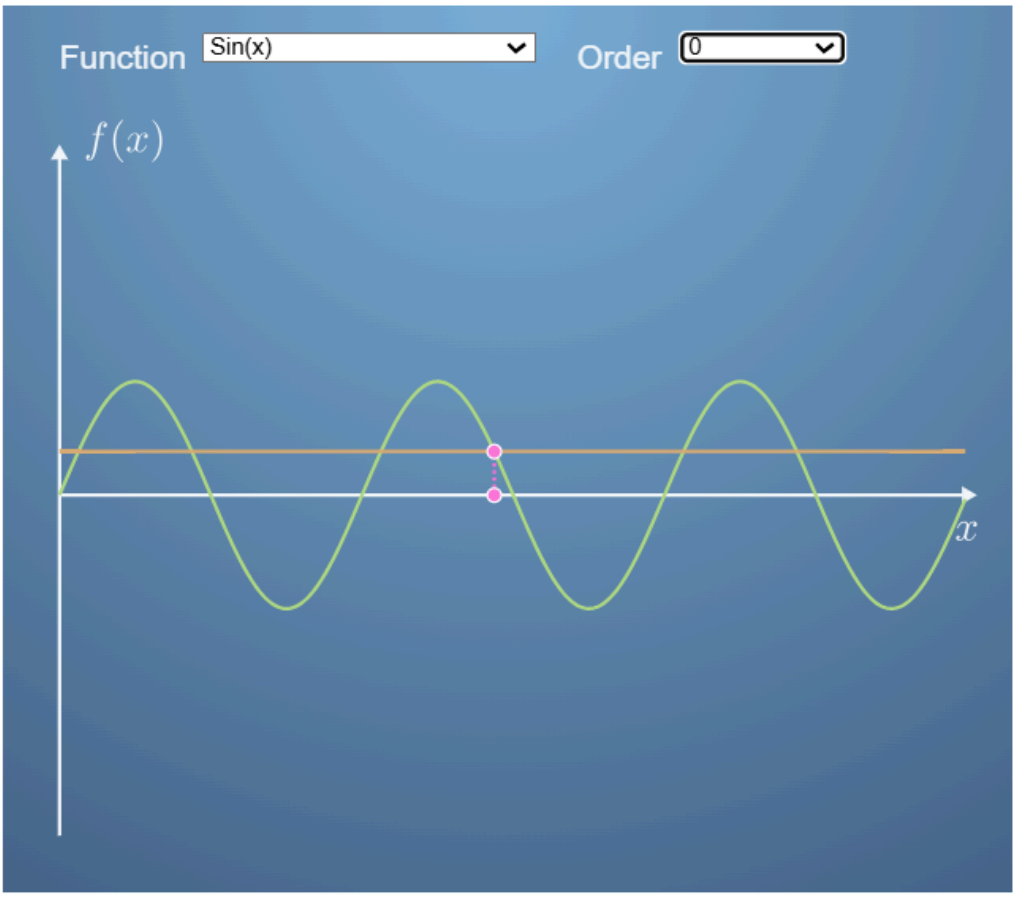
...and so on...

- **3rd-Order Approximation (Cubic):** Will match the "change in curvature" (the jerk).
- **nth-Order Approximation:** Will match the nth-derivative.

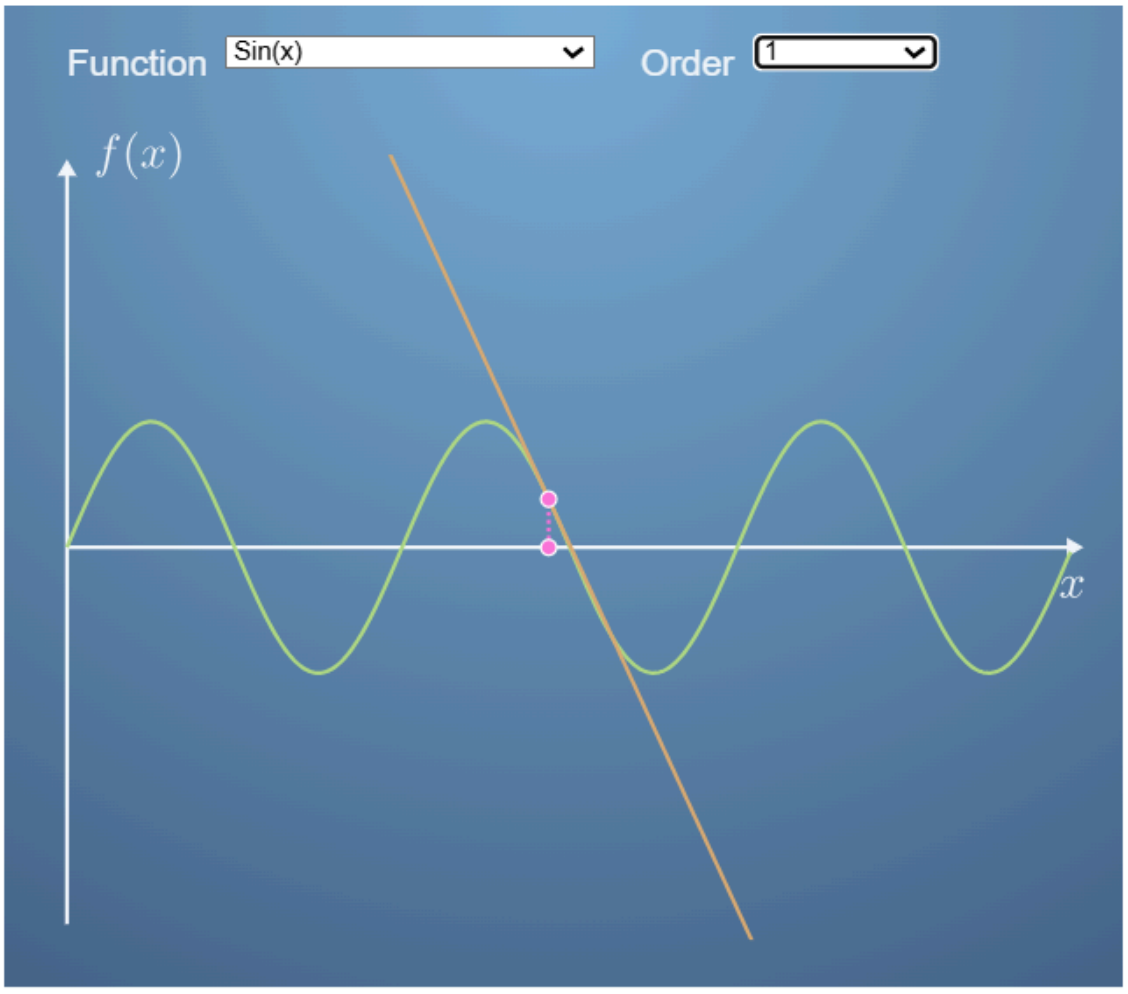


Example :

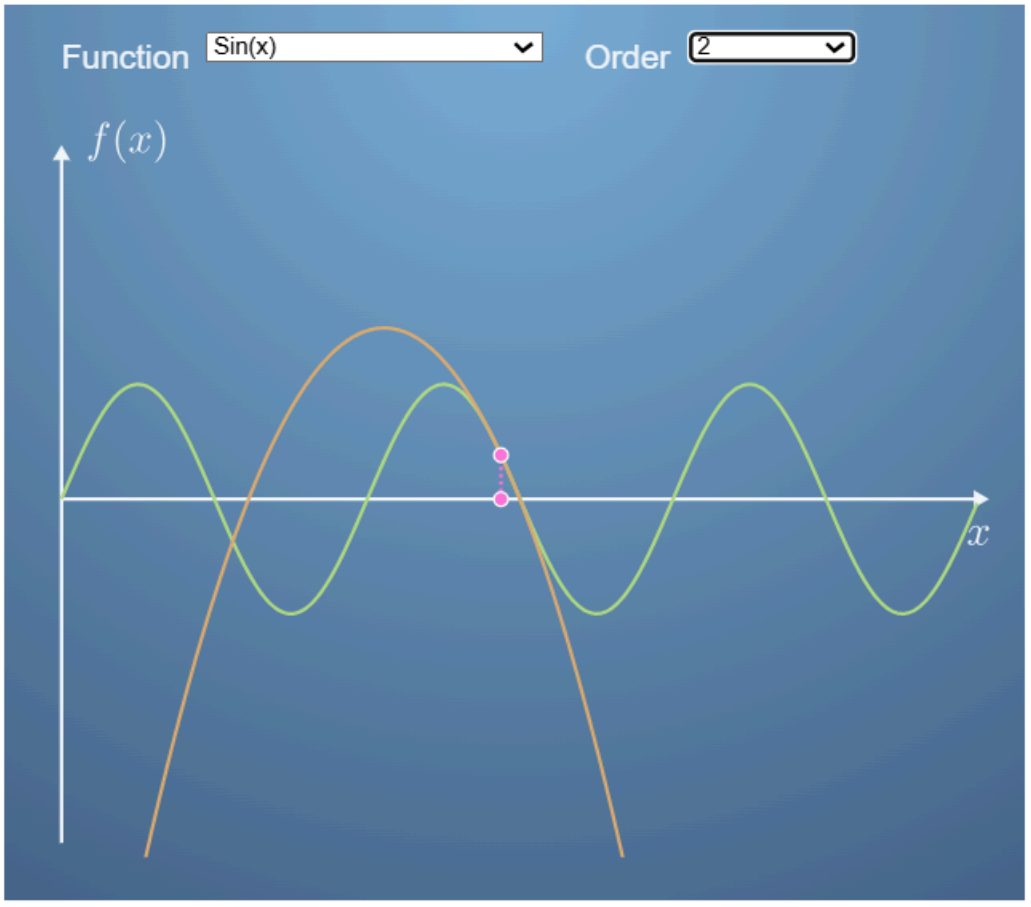
Visualising Taylor Series

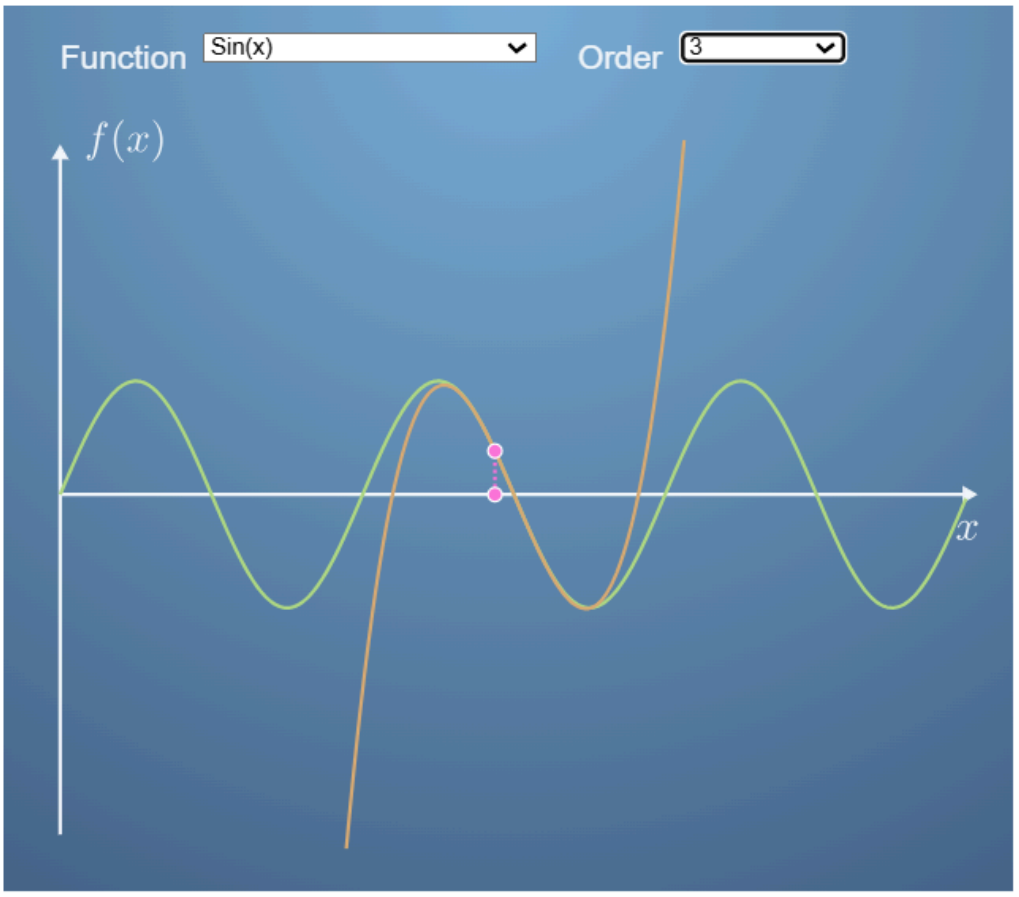


Visualising Taylor Series



Visualising Taylor Series

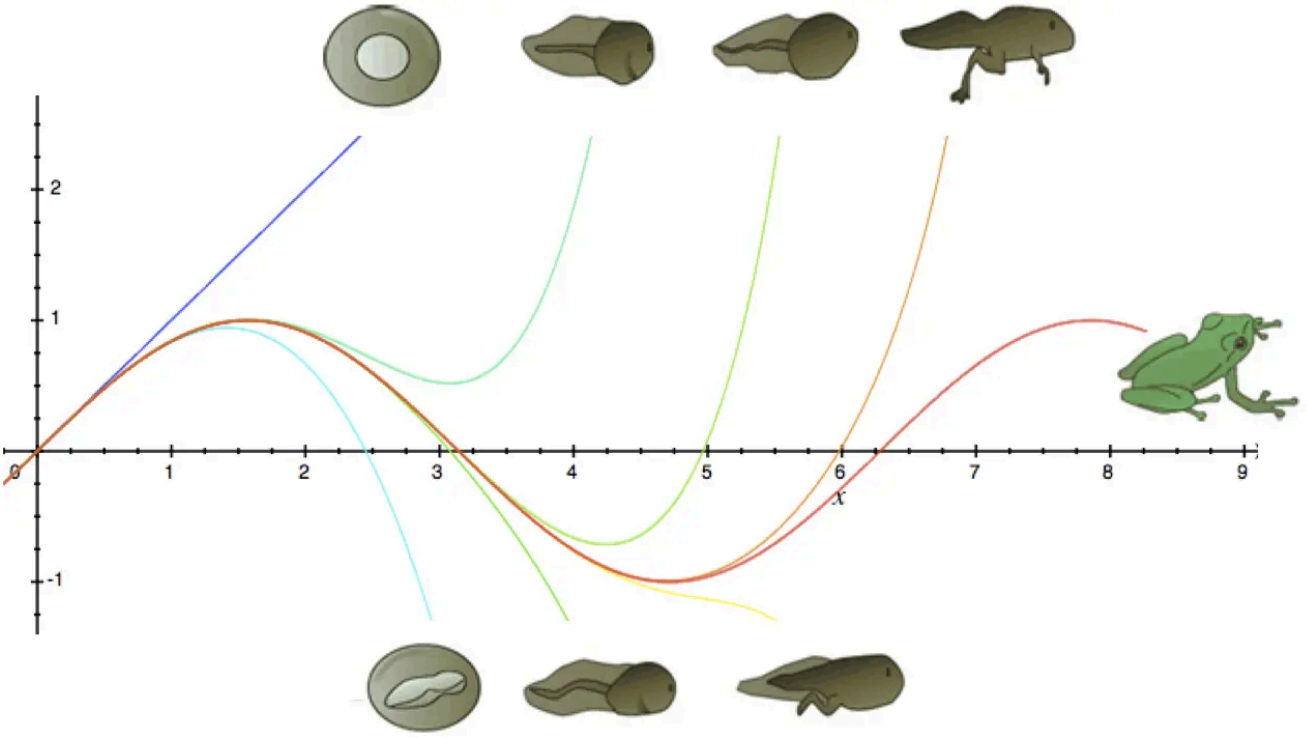




---

4. Key Message

- A Taylor Series builds its approximation **incrementally**.
- Each new term that is added (cx) has the job of matching a **higher-order derivative** of the original function.
- The more terms we use, the "tighter the hug" of our polynomial imitator is to the original function around the point of contact.
- The next video will discuss how to systematically calculate the values of these coefficients, $c_0, c_1, c_2, \ldots$.



https://betterexplained.com/articles/taylor-series/

Tags: #mml-specialization #multivariate-calculus #taylor-series #power-series #approximation