

00: Welcome to Multivariate Calculus - The "Compass" for the Data Landscape

You've successfully built the "map" with [Linear Algebra](#). You now understand what "parameter space," "basis," and "vectors" are. This course, as explained by the instructors, will give you the "compass and hiking boots" to actually explore that map.

Here is the adventure roadmap for this course, based on this introductory video:

1. First Step: Building an Intuitive Foundation (Back to Basics)

- **What we will do:** We will start again from the fundamentals of calculus, much like the 3Blue1Brown approach, using animations to build intuition. We won't jump directly into complex formulas.
 - **Why it's important:** This ensures our foundation of "tiny nudges" (dx) is rock-solid before we take it to the next level.
-

2. Second Step: Leveling Up to Multi-Dimensions ("Navigating Mountains in the Dark")

- **What we will do:** This is where the "Multivariate" part comes in. We will generalize the idea of a [derivative](#) to functions with multiple inputs.
 - **Key Analogy:** Remember the "badness landscape" from the Linear Algebra course, where we wanted to find the bottom of the valley? This video calls it "navigating mountains in the dark." Multivariate Calculus will give us the "compass" (the **Gradient**) that, at any point, always points in the direction of the steepest ascent, so we know which way to step to get to the bottom of the valley (by going in the opposite direction).
-

3. Third Step: Real Application #1 - Dissecting the "Guts" of a Neural Network

- **What we will do:** This is one of the biggest "payoffs" of this course. We will see exactly how calculus is used to **"train" a Neural Network**.
- **The Connection:** The "learning" process of a Neural Network is essentially the process of "descending a valley" in a parameter landscape that has millions of dimensions. We

will even try to write the code for it in Python.

4. Fourth Step: Real Application #2 - Returning to the Initial Problem

- **What we will do:** We will circle back to the "data fitting" problem that motivated us at the beginning, but this time with a more complex case. We will use everything we've learned to fit a complex function to real data.
 - **The Climax:** This is the final project where [Linear Algebra](#) (to represent the data) and [Calculus](#) (to find the best fit) will work together to solve a practical problem.
-

Conclusion:

This course is the bridge that will connect all the mathematical theory you've learned with its practical applications in Machine Learning. The goal is to give you the confidence that you don't just "know" the formulas, but you understand what they mean and how they work behind the scenes.

Our learning method will remain the same. I'm ready to accompany you every step of the way.

Let's begin this new adventure.

Tags: [#mml-specialization](#) [#multivariate-calculus](#) [#course-introduction](#) [#roadmap](#)