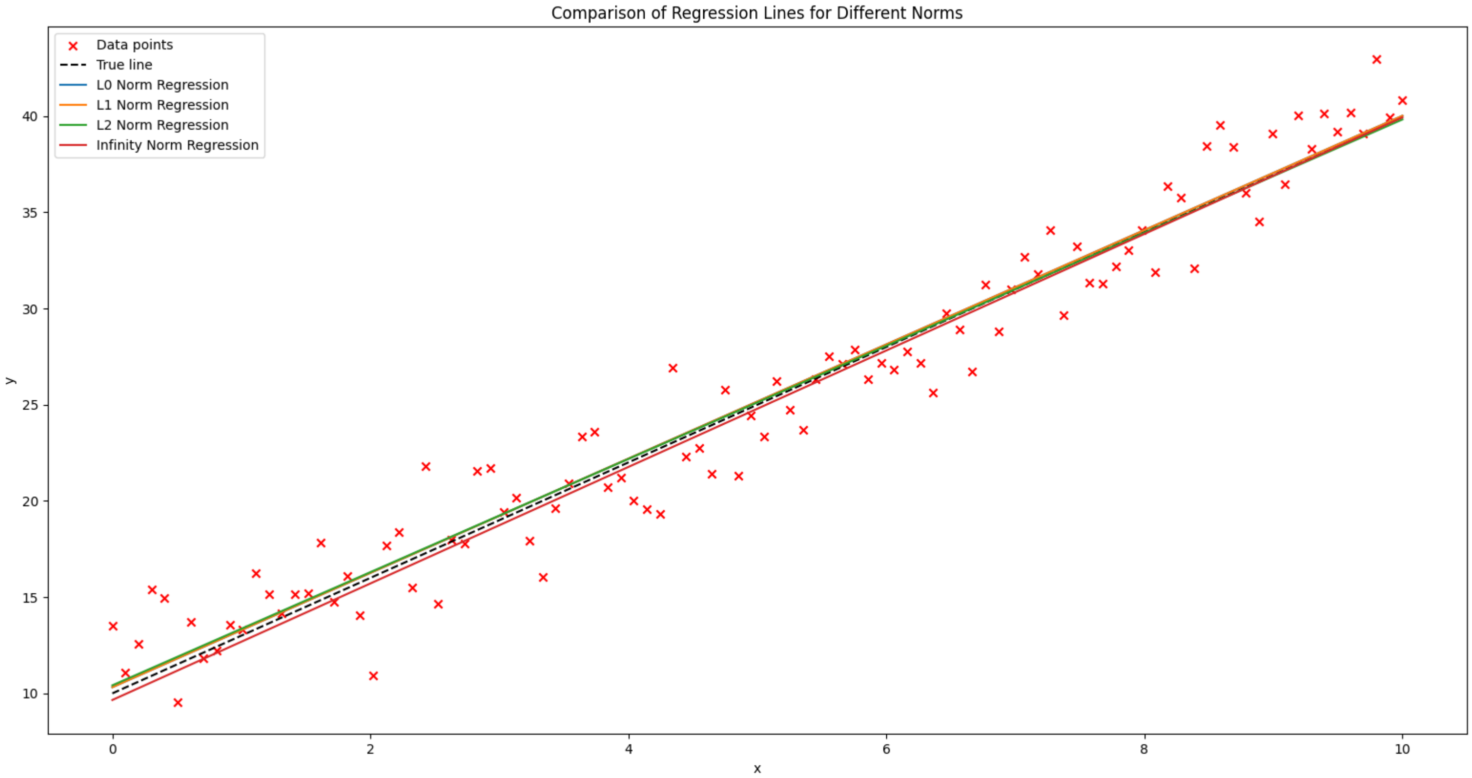
Mathematics of AI

Fereshteh Baradaran

Assignment #2



Introduction

This documentation outlines the development and functionality of a Python program designed to minimize various error norms in the context of linear regression. Linear regression is a fundamental statistical and machine learning technique used to model the relationship between a dependent variable and one or more independent variables. In this specific case, the program tackles a linear regression problem where the goal is to find the best fit line through a set of data points. The fit is evaluated based on different error norms, providing a comprehensive view of model accuracy.

Implementation

The implementation of this assignment is carried out in Python. The complete source code is available in a GitHub repository that can be accessed at the following link.

The program performs the following key tasks:

- **Data Generation**

  The dataset is generated using the *numpy* library. A linear relationship is simulated with added Gaussian noise to mimic real-world data variability:

- **Linear Model and Error Calculation**

  The linear model is defined as a function, calculatePredicted_Y, which calculates $\hat{y}$ based on input x, and model parameters α and β. The signed error is then calculated for each data point.

- **Error Norm Minimization**

  Four different error norms are minimized:

  - L0 Norm: Counts the number of non-zero errors.
  - L1 Norm: Sum of absolute values of errors, leading to median regression.
  - L2 Norm: Sum of squares of errors, equivalent to ordinary least squares regression.
  - Infinity Norm: Maximum absolute error.

  The scipy.optimize.minimize function is used for minimization, employing the Sequential Least Squares Programming (SLSQP) method.

* Sequential Least Squares Programming (SLSQP) is an optimization algorithm that is particularly well-suited for constrained optimization problems. SLSQP works by approximating the objective function locally as a quadratic and the constraints linearly, solving this approximate problem, and then iteratively updating the solution.

It’s a good general-purpose optimizer that can efficiently handle the different types of norms (L0, L1, L2, Infinity) which can vary greatly in their behavior.

- **Visualization**

  The dataset and the resulting regression lines for each error norm are visualized using matplotlib.pyplot