ECE 105: Introduction to Electrical Engineering

Lecture 16

Curve Fitting

Yasser Khan

Rehan Kapadia

Outline

- **Objective**: Understand how to fit data using linear and non-linear techniques and gain an introductory understanding of neural networks.

- **Topics Covered**:
  - Linear Curve Fitting
  - Non-Linear Curve Fitting

Linear Regression

- **Linear Curve Fitting**
- **Definition**: Approximating a set of data points with a straight line.
- **Theory**:
  - **Least Squares Method**: Find the line of best fit by minimizing the sum of squared residuals.
  - **Equation**: $\mathbf{y = mx + b}$, where $\mathbf{m}$ is the slope and $\mathbf{b}$ is the intercept.
- **Steps to Perform Linear Curve Fitting**:
  - Organize data points.
  - Calculate slope and intercept using least squares.
  - Plot data with the fitted line.
- **Examples**:
  - Fit data points representing the relationship between temperature and energy consumption.
- **Applications**:
  - Predictive modeling in economics and engineering.

Math Behind Linear Regression

- **Definition**: Linear curve fitting is the process of finding the best-fit line that approximates a set of data points.
- **Goal**: Minimize the distance between the observed data points and the fitted line.
- **Equation of a Line**: $y = mx + b$
  - **m**: Slope (rate of change)
  - **b**: Intercept (value when $x = 0$)

Math Behind Linear Regression

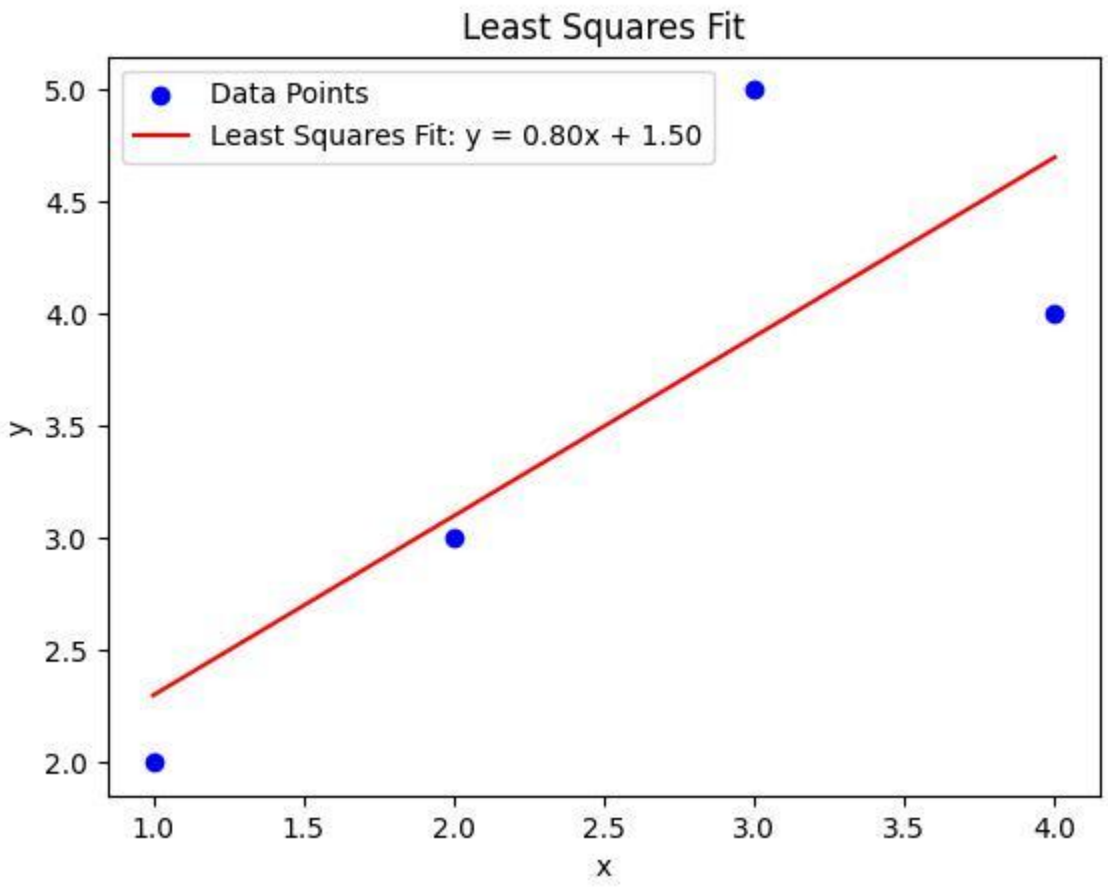
- **Linear Regression Equation**: $y = mx + b$
  - **Dependent Variable (y)**: The value we want to predict.
  - **Independent Variable (x)**: The value that influences **y**.
- **Least Squares Method**:
  - Objective: Minimize the **sum of squared residuals**.
  - **Residual**: The difference between the observed value and the value predicted by the model.
  - **Mathematical Representation**: Minimize $\sum(y_i - (mx_i + b))^2$ for all data points **i**.

Math Behind Linear Regression

- **Slope Formula (m)**:
  - $\mathbf{m = \sum((x_i - \bar{x})(y_i - \hat{y})) / \sum(x_i - \bar{x})^2}$
  - $\bar{x}$: Mean of the x-values
  - $\hat{y}$: Mean of the y-values
- **Intercept Formula (b)**:
  - $\mathbf{b = \hat{y} - m\bar{x}}$
- **Finding the Line of Best Fit**:
  - Once **m** and **b** are known, substitute back into **y = mx + b**.

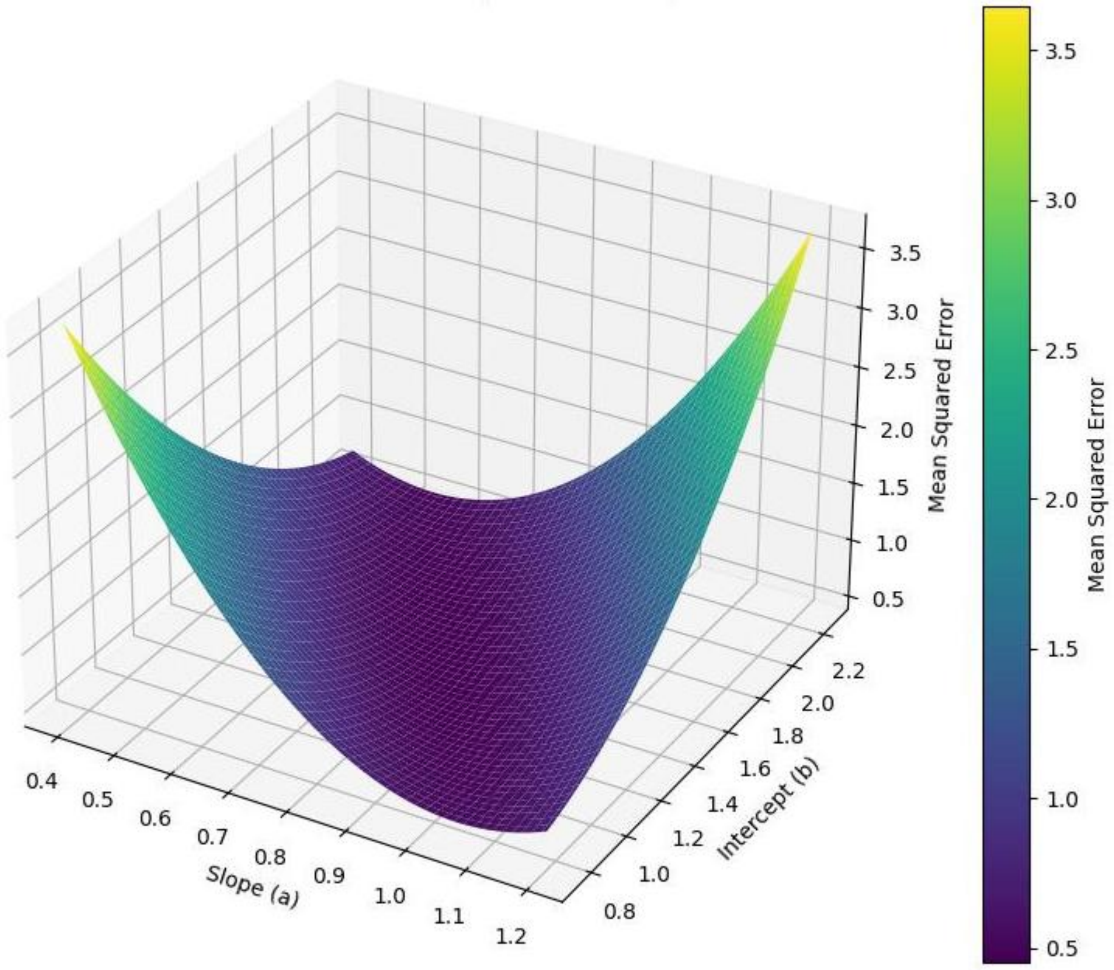
Finding the Best Fit

- **Given Data Points**: (1, 2), (2, 3), (3, 5), (4, 4)
- **Step 1**: Calculate **x̄** and **ȳ**:
  - **$\bar{x}$ = (1 + 2 + 3 + 4) / 4 = 2.5**
  - **$\bar{y}$ = (2 + 3 + 5 + 4) / 4 = 3.5**
- **Step 2**: Calculate **m** and **b**:
  - **m = 0.8**
  - **b = 1.5**
- **Step 3**: Line Equation:
  - **y = 0.8x + 1.5**



Finding the Best Fit

- **Given Data Points**: (1, 2), (2, 3), (3, 5), (4, 4)
- **Step 1**: Calculate $\bar{x}$ and $\bar{y}$:
  - $\bar{x} = (1 + 2 + 3 + 4) / 4 = 2.5$
  - $\bar{y} = (2 + 3 + 5 + 4) / 4 = 3.5$
- **Step 2**: Calculate **m** and **b**:
  - **m = 0.8**
  - **b = 1.5**
- **Step 3**: Line Equation:
  - **y = 0.8x + 1.5**



Non-linear Regression

- **Non-Linear Curve Fitting**
- **Definition**: Finding a curve that best fits a set of data points when a straight line is not appropriate.
- **Theory**:
  - **Polynomial Fitting**: Use higher-order polynomials (**$y = a + bx + cx^2 + \ldots$**).
  - **Exponential and Logarithmic Models**: Fit data using non-linear functions like **y = a * e^(bx)** or **y = a * log(bx)**.
  - **Optimization Methods**: Use gradient descent or numerical optimization to find the best fit.
- **Steps to Perform Non-Linear Fitting**:
  - Choose an appropriate model based on data behavior.
  - Use iterative methods to minimize the error.
- **Examples**:
  - Fit a sine curve to periodic data such as seasonal temperature variations.
- **Applications**:
  - Any space where the relationships are non-linear, which is pretty much most areas

Common types of non-linear regresion

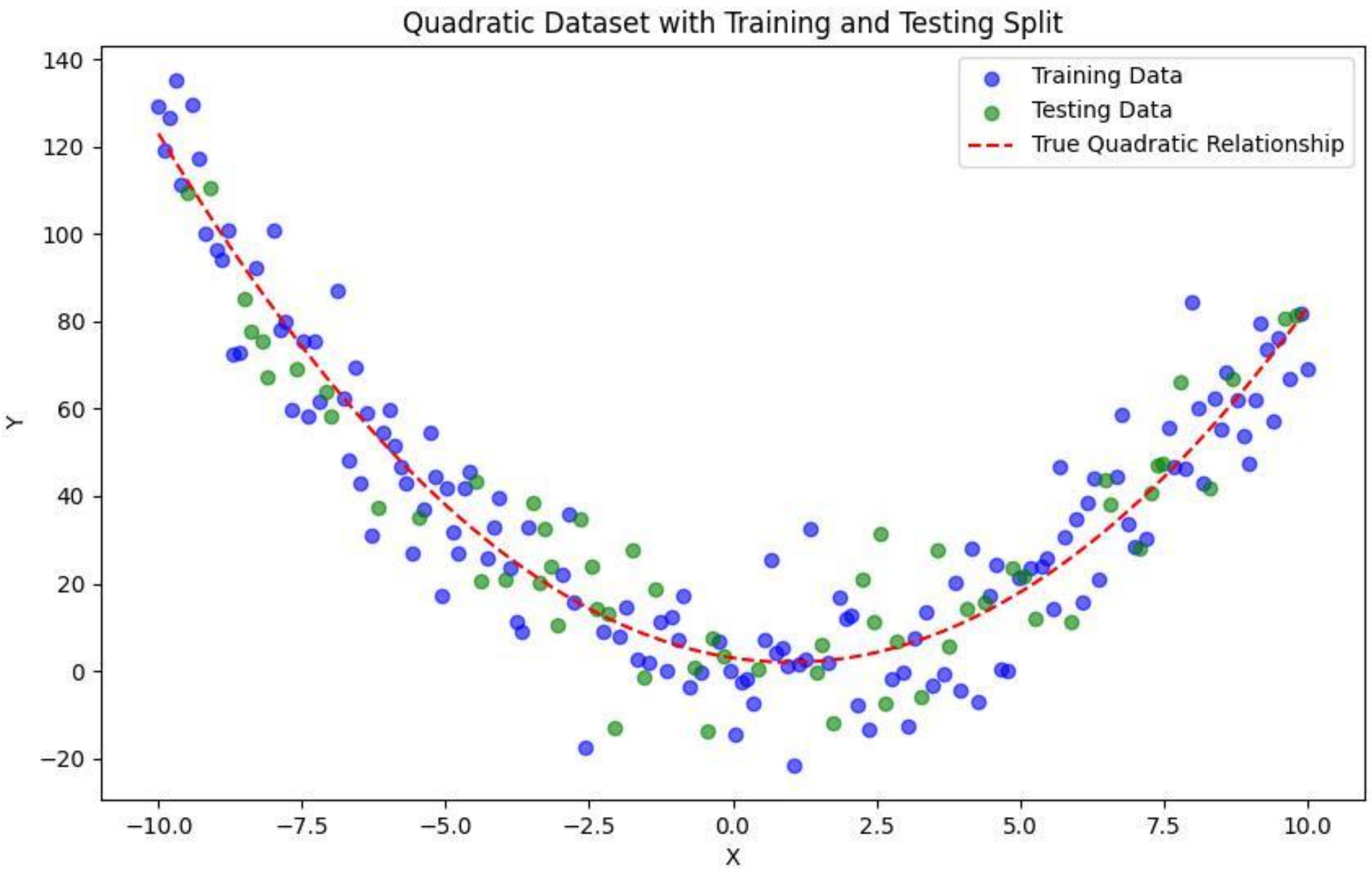
- **Model Types**:
  - **Polynomial Fitting**: Use higher-order polynomials (**$y = a + bx + cx^2 + \ldots$**).
  - **Exponential Models**: **y = a * e^(bx)**, commonly used for growth models.
  - **Logarithmic Models**: **y = a * log(bx)**, often used for decay or diminishing returns.
- **General Approach**:
  - Choose an appropriate model type.
  - Use **iterative methods** (e.g., gradient descent) to minimize the error.
  - Unlike linear regression, there is no closed-form solution.

Minimizing the error

- **Objective**: Minimize the sum of squared differences between observed values and predicted values.
  - **Mathematical Representation**: Minimize $\sum(y_i - f(x_i, \theta))^2$, where $f(x_i, \theta)$ is the non-linear model with parameters θ.
- **Gradient Descent**:
  - An iterative optimization algorithm used to find the parameters that minimize the error function.
  - Adjusts parameters by moving in the direction of the negative gradient.

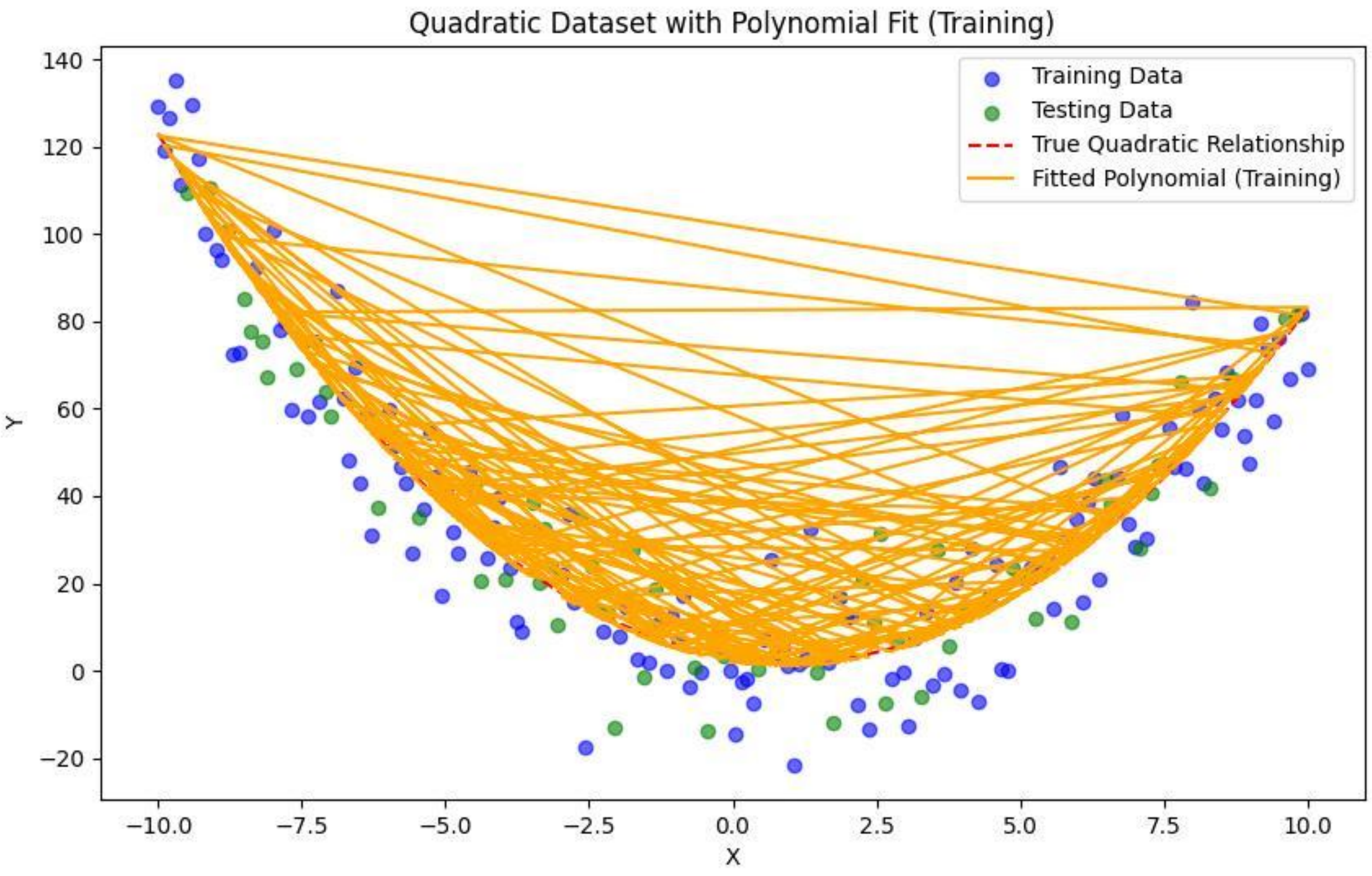
Let's look at an example

- Assume we have a dataset which is a quadratic function with some noise
- Split the dataset into training data and testing data
- Fit your model using training data, and then test your model using testing data



Let’s look at an example

- Assume we have a dataset which is a quadratic function with some noise



Let’s look at an example

- We can see the error during the iterations for both the training set and the testing set