House Price Prediction using Linear Regression (From Scratch)

Name: Abhishek
Program: Data Science & Machine Learning Internship
Project Title: House Price Prediction using Linear Regression (From Scratch)

Course Overview

Introduction

This learning module is designed to help learners understand how Linear Regression works internally by implementing the algorithm from scratch using Python, NumPy, and Pandas without relying on machine learning libraries such as scikit-learn.

Instead of using built-in ML functions, this project demonstrates how predictions are calculated mathematically, how errors are measured, and how model parameters are updated using gradient descent.

The learning experience is structured as a sequence of conceptual videos, each corresponding to a section in the Jupyter Notebook.

Summary

By the end of this module, learners will be able to:

- Understand regression as a supervised learning problem
- Preprocess real-world datasets
- Implement Linear Regression mathematically
- Apply Gradient Descent optimization
- Evaluate models using RMSE and R^2

Video 1: Dataset Loading and Understanding

Introduction

In this video, learners are introduced to the housing dataset used for price prediction. The dataset is explored using Pandas functions such as head(), info(), and columns. Each feature is examined to understand whether it is numerical or categorical and how it may influence house prices.

Understanding the dataset structure is the first step in building any reliable machine learning model.

Summary

After this video, learners will:

- Inspect datasets effectively
- Identify feature types
- Understand the regression problem setup

Video 2: Exploratory Data Analysis (EDA)

Introduction

This video focuses on visualizing the dataset to understand relationships between features and house prices. Histograms analyze price distribution, scatter plots show relationships between area and price, box plots compare bedroom counts, and heatmaps reveal correlations between features.

EDA helps validate whether a linear model is suitable for the dataset.

Summary

Learners will:

- Interpret feature distributions
- Identify correlations
- Justify using Linear Regression

Video 3: Data Preprocessing and Encoding

Introduction

Machine learning algorithms require numerical input. In this video, binary categorical variables such as “yes” and “no” are mapped to 1 and 0. The multi-category feature “furnishing status” is transformed using one-hot encoding.

This step ensures all features are converted into a format suitable for mathematical computation.

Summary

Learners will:

- Understand why encoding is required
- Apply binary mapping
- Implement one-hot encoding

Video 4: Feature Scaling and Train-Test Split

Introduction

This video explains why feature scaling is necessary for gradient descent optimization. Standardization is applied using only training data statistics to avoid data leakage. The dataset is divided into training (80%) and testing (20%) sets to ensure fair evaluation.

Summary

Learners will:

- Understand feature scaling importance
- Prevent data leakage
- Perform proper train-test splitting

Video 5: Linear Regression Model Implementation

Introduction

This video explains the mathematical equation of Linear Regression and how predictions are computed using weights and bias. The prediction function is implemented manually using NumPy's dot product.

Summary

Learners will:

- Understand regression equations
- Implement prediction logic in Python

Video 6: Loss Function (Mean Squared Error)

Introduction

Mean Squared Error (MSE) is introduced as a way to measure prediction error. The formula is implemented manually to understand how the model's performance is evaluated during training.

Summary

Learners will:

- Compute MSE
- Understand why squared errors are used

Video 7: Training Using Gradient Descent

Introduction

Gradient Descent is implemented to iteratively update weights and bias to minimize error. The model starts with zero weights and gradually reduces loss over multiple iterations. A loss curve is plotted to observe convergence.

Summary

Learners will:

- Understand optimization using gradients
- Implement weight updates
- Interpret training loss curves

Video 8: Model Evaluation (RMSE and R^2)

Introduction

The trained model is evaluated on unseen test data. Metrics such as MSE, RMSE, and R^2 score are computed to measure how well the model generalizes.

Summary

Learners will:

- Interpret RMSE in price units
- Understand R^2 as variance explanation
- Evaluate regression models objectively

Video 9: Residual Analysis

Introduction

Residual plots are used to visualize prediction errors. This helps check whether errors are randomly distributed around zero and whether the model shows systematic bias.

Summary

Learners will:

- Understand residual distribution
- Identify model weaknesses

Video 10: Example Prediction and Model Conclusion

Introduction

The trained model is used to predict the price of a sample house. Predicted and actual prices are compared to demonstrate real-world application and limitations of Linear Regression.

Summary

Learners will:

- Apply trained models
- Understand limitations and future improvements

Assessment Questions

1. What is the main goal of Linear Regression?
   A. To classify data into categories
   B. To predict continuous numerical values
   C. To cluster similar data points
   D. To reduce dimensionality
   Correct Answer: B
   Feedback: Linear Regression is used to predict continuous outcomes such as house prices.

2. Why is Exploratory Data Analysis (EDA) performed before modeling?
   A. To train the model
   B. To remove all features
   C. To understand data patterns and relationships
   D. To increase dataset size
   Correct Answer: C
   Feedback: EDA helps identify trends, correlations, and anomalies before training.

3. Why must categorical variables be encoded?
   A. Linear Regression only works on text
   B. Algorithms require numerical input
   C. Encoding improves visualization
   D. Encoding increases dataset size
   Correct Answer: B
   Feedback: ML algorithms perform mathematical operations, so inputs must be numeric.

4. What is the purpose of feature scaling?
   A. To increase feature values
   B. To make features comparable in magnitude
   C. To remove noise
   D. To reduce dataset size
   Correct Answer: B
   Feedback: Scaling helps gradient descent converge faster and more reliably.

5. Which metric measures average squared prediction error?
   A. MAE
   B. RMSE
   C. MSE
   D. R^2
   Correct Answer: C
   Feedback: Mean Squared Error measures the average of squared prediction errors.

6. What does Gradient Descent do?
   A. Sorts the dataset
   B. Minimizes the loss function
   C. Visualizes data
   D. Encodes features

Correct Answer: B
Feedback: Gradient Descent optimizes model parameters to reduce error.

7. What does a high R^2 value indicate?
A. Poor model performance
B. No relationship between variables
C. Strong explanatory power
D. High error
Correct Answer: C
Feedback: A higher R^2 means the model explains more variance in the data.

8. Why is train-test split important?
A. To make dataset smaller
B. To improve visualization
C. To evaluate model generalization
D. To encode features
Correct Answer: C
Feedback: It ensures the model performs well on unseen data.

9. What does a residual plot help detect?
A. Feature importance
B. Model bias and patterns in errors
C. Data types
D. Dataset size
Correct Answer: B
Feedback: Residual plots reveal whether prediction errors are random or systematic.

10. What is the main benefit of implementing Linear Regression from scratch?
A. Faster execution
B. Less memory usage
C. Better understanding of algorithm internals
D. Higher accuracy
Correct Answer: C
Feedback: Writing algorithms manually improves conceptual understanding.