CS412 - Machine Learning: Homework 2

Notebook Link: https://colab.research.google.com/drive/1z6nvCxZrjGWHxwvHDtL61cI2t_oLFcah?usp=sharing

Title: Linear and Polynomial Regression Analysis

Name: Mete Kerem Berk
Student ID: 30933
Course: CS412 - Machine Learning
Homework Number: HW2
Submission Date: March 16, 2025

---

1. Introduction

Linear regression is a fundamental method in machine learning used to model relationships between variables. This homework focuses on implementing and analyzing linear and polynomial regression techniques on two datasets. The primary objectives are:

- Understanding and applying linear regression using three different approaches: scikit-learn's built-in method, ordinary least squares (OLS), and gradient descent.
- Extending regression analysis to polynomial functions to model nonlinear relationships.
- Evaluating model performance using the mean squared error (MSE) metric.
- Comparing different regression techniques to derive key insights.

---

2. Datasets & Experimental Setup

Dataset 1: Linear Data

- A dataset of (x, y) pairs where y is generated from a linear function with added Gaussian noise.
- The dataset was split into 50% training and 50% validation sets.

Dataset 2: Nonlinear Data

- A dataset of (x, y) pairs where y follows a nonlinear function with added Gaussian noise.
- Data was read from provided `.npy` files and split using the same 50-50 ratio.

---

3. Methods & Implementation

Part 1: Linear Regression on Dataset 1

(a) Scikit-Learn's Linear Regression (Part 1.a)

- Utilized `LinearRegression` from `sklearn.linear_model`.
- Trained the model on Dataset 1 and evaluated predictions on the validation set.

(b) Ordinary Least Squares (Part 1.b)

- Implemented OLS manually.
- Computed coefficients and used them for prediction.

(c) Gradient Descent (Part 1.c)

- Implemented gradient descent to iteratively optimize weights.
- Used a learning rate of 0.1 and 1000 iterations.
- Converged to optimal parameters and evaluated performance.

Part 2: Polynomial Regression on Dataset 2

(a) Polynomial Regression using Scikit-Learn (Part 2.a)

- Applied `PolynomialFeatures` from `sklearn.preprocessing` to generate polynomial features of degrees 1, 3, 5, and 7.
- Fitted linear regression models to the transformed data and computed validation MSE.

(b) Manual Polynomial Regression (Part 2.b)

- Implemented polynomial regression manually for degree 3.
- Constructed the polynomial feature matrix and applied the OLS method.

---

4. Results & Discussion

Performance Comparison (MSE Analysis)

Method	MSE
Scikit-Learn Linear Regression	0.00795462682779033
Manual OLS	0.00795462682779037
Gradient Descent	0.00527514849082713
Polynomial (Degree 1)	0.06362852709262727
Polynomial (Degree 3)	0.01205922374286829
Polynomial (Degree 5)	0.00747546307928118
Polynomial (Degree 7)	0.01154922783882886

Visualization & Insights

- **Dataset 1:** Linear regression methods performed well, with slight variations in MSE values.
- **Dataset 2:** Higher-degree polynomial regression showed improved fitting but risked overfitting for degrees 5 and 7.
- **Gradient descent:** Required careful tuning of hyperparameters for convergence.

---

5. Conclusion

This homework provided hands-on experience in implementing and analyzing regression methods. Key takeaways:

- Linear regression methods performed well on dataset 1.
- Polynomial regression effectively captured nonlinearity in dataset 2, but higher degrees led to overfitting.
- Gradient descent, while powerful, required careful tuning.

Overall, this assignment reinforced the importance of selecting appropriate models based on data characteristics and balancing bias-variance tradeoffs.