Homework 2

March 20, 2025

Instructions

This assignment is due on [**March 29, 2025**].

1 Introduction

In this challenge, you will explore the full machine learning pipeline for a regression problem. You will work with a real-world dataset, perform data cleaning and transformation, and implement two regression algorithms: KNN and Linear Regression, by building your own versions. In addition, you will use scikit-learn's implementations to benchmark your work.

2 Task Overview

Your assignment is to:

- Acquire and Understand the Data: Download this dataset and perform exploratory data analysis.
- Preprocess the Data: Address missing values, encode categorical features, and scale numerical features.
- Develop Models:
  - Custom Implementation: Build KNN and Linear Regression models from scratch.
  - Library Models: Train and evaluate the same models using scikit-learn.

- Optimize Performance: Use hyperparameter tuning methods to find the best configurations.
- Compare and Analyze: Assess the performance and runtime of your custom algorithms versus scikit-learn's models.

3 Detailed Instructions

3.1 Exploratory data analysis

- Conduct a thorough exploratory data analysis (EDA).
- Visualize feature distributions, identify correlations, and spot outliers.
- Summarize your findings in a brief report.

3.2 Data Preprocessing

- Handling Missing Data: Identify missing values and decide on an imputation or removal strategy.
- Categorical Encoding: Detect categorical variables and apply appropriate encoding (e.g., one-hot encoding).
- Feature Scaling: Standardize or normalize your numerical features as needed, especially to improve KNN performance.

3.3 Model Development

- Using scikit-learn:
  - Train both KNN and Linear Regression models.
  - Experiment with hyperparameters (e.g., varying neighbors in KNN, choosing the best solver or adding regularization in Linear Regression) using techniques like Grid Search or Random Search.
  - Record performance metrics (e.g., Mean Squared Error, Mean Absolute Error, R^2 Score).
- Custom Implementation:
  - KNN Regression: Implement the algorithm from scratch including distance computations and neighbor selection. Include at least two options for distance (e.g. Euclidean and Manhattan).

Integrate hyperparameter tuning (e.g., choosing the number of neighbors).

- Linear Regression: Create your own model using an approach of your choice (e.g., closed-form solution or gradient descent).

- Evaluation and Comparison: Evaluate both the custom and scikit-learn models on a test set. Compare metrics and runtime and discuss potential reasons for any discrepancies. Reflect on trade-offs between custom code and mature library implementations.

4 Important Note

Avoid using Python *for loops* as much as possible. Instead, rely on NumPy functionality.