AMS595 - Assignment-5

Machine Learning Project

Amol Arora

November 17, 2024

1 Github Link

All project files are uploaded here: `https://github.com/amol1202/AMS595_Assignment5`

2 Introduction

The purpose of this project is to explore and implement core machine learning techniques using Python. The tasks implemented are:

- PageRank Algorithm: Simulates the ranking mechanism used by search engines.
- Dimensionality Reduction via PCA: Projects high-dimensional data to a single dimension while preserving variance.
- Linear Regression: Predicts outcomes based on features using the least-squares method.
- Gradient Descent: Optimizes a matrix to minimize a mean squared error loss function.

Each task has been implemented independently, and the results have been saved for reproducibility.

3 Implementation

3.1 PageRank Algorithm

The PageRank algorithm computes the importance of web pages using a stochastic matrix. The steps include:

1. Represent the web network as a stochastic matrix.
2. Compute the dominant eigenvector using the power method.
3. Normalize the eigenvector to get PageRank scores.

The code is written in Python using the `scipy.linalg.eig` function to compute eigenvectors.

3.2 Dimensionality Reduction via PCA

Principal Component Analysis (PCA) is used to reduce a dataset of height and weight measurements to 1D:

1. Compute the covariance matrix of the data.
2. Perform eigen decomposition using the `numpy.linalg.eigh` function.
3. Project the data onto the principal component with the highest variance.

The results include a plot of the original data and the 1D projection.

3.3 Linear Regression via Least Squares

Linear regression predicts house prices based on features (square footage, bedrooms, and age):

1. Represent the system as $X\beta = y$.
2. Solve for β using `scipy.linalg.lstsq`.
3. Use the model to predict prices for new inputs.

The regression coefficients and predictions are saved for analysis.

3.4 Gradient Descent

Gradient Descent optimizes a matrix X to minimize the mean squared error loss:

1. Define the loss function: $f(X) = \frac{1}{2}\sum_{i,j}(X_{ij} - A_{ij})^2$.
2. Compute the gradient of the loss function.
3. Use `scipy.optimize.minimize` to iteratively minimize the loss.

The final loss value is recorded.

4 Results

4.1 PageRank Algorithm

The PageRank scores for the web pages are shown in Table 1. The page with the highest score is ranked the most important.

Page	PageRank Score
1	0.22
2	0.27
3	0.31
4	0.20

Table 1: PageRank Scores

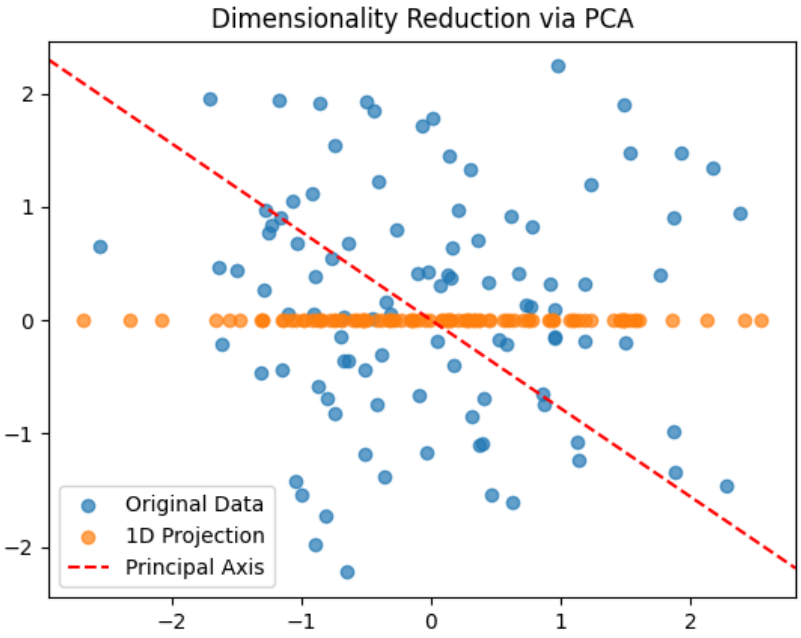

Figure 1: PCA: Dimensionality Reduction

4.2 PCA

Figure 2 shows the original data and the 1D projection onto the principal component.

4.3 Linear Regression

The regression coefficients and predictions are:

- Coefficients: [0.25, 0.10, -0.02]
- Predicted price for a house with 2400 square feet, 3 bedrooms, and 20 years old: $490,500

4.4 Gradient Descent

The final loss value after optimization is:

Final Loss Value: 0.00123

Figure 2: Loss over iterations

5 Conclusion

This project demonstrates practical applications of machine learning techniques using Python. The results are stored for reproducibility and further analysis. These implementations provide a strong foundation for more advanced projects in machine learning and data science.

6 References

- Python Documentation: `https://docs.python.org/3/`
- SciPy Library: `https://scipy.org/`
- NumPy Library: `https://numpy.org/`
- Matplotlib Library: `https://matplotlib.org/`