MIDTERM ASSIGNMENT

Instructor: Dr. Selim Yılmaz (selimyilmaz@mu.edu.tr)
Out Date: 11/29/2021 14:29:59
Due Date: 12/13/2021 14:29:59

DECLARATION OF HONOR CODE[1]

Student ID 210717012
Name Ahmet Bekir
Surname Aslanalp

In the course of Introduction to Machine Learning (SE3007), I take academic integrity very seriously and ask you to do as well. That's why, this page is dedicated to some clear statements that defines the policies of this assignment, and hence, will be in force. Before reading this assignment booklet, please first read the following rules to avoid any possible violation on academic integrity.

- This assignment must be done individually unless stated otherwise.
- You are encouraged to discuss with your classmates about the given assignments, but these discussions should be carried out in an abstract way. That is, you cannot copy code (in whole or in part) of someone else, cannot share your code (in whole or in part) with someone else either.
- The previous rule also holds for the material found on the web as everything on the web has been written by someone else. Furthermore, you are welcome to seek support from generative AI chatbots like ChatGPT, Gemini, and others; however, ensure these tools do not perform the task on your behalf.
- You must not look at solution sets or program code from other years.
- You cannot share or leave your code (in whole or in part) in publicly accessible areas.
- You have to be prepared to explain the idea behind the solution of this assignment you submit.
- Finally, you must make a copy of your solution of this assignment and keep it until the end of this semester.

I have carefully read every of the statements regarding this assignment and also the related part of the official disciplinary regulations of Muğla Sıtkı Koçman University and the Council of Higher Education. By signing this document, I hereby declare that I shall abide by the rules of this assignment to prevent any violation on academic integrity.

Signature Ahmet

[1]This page should be filled and signed by your handwriting. Make it a cover page of your report.

Task 1: Missing Value Imputation

Techniques Used

- **Random Imputation**:
  - Missing values in Feature2 were replaced with random values sampled uniformly within the observed range of Feature2.
- **Regression Imputation**:
  - Linear Regression was used to predict missing values in Feature2 using Feature1, Feature3, and Feature4 as predictor features.

Results

Table 1: Mean Squared Error (MSE) for Imputed Values.

Target Variable	MSE
Target 1	0.031
Target 2	0.086
Target 3	0.017

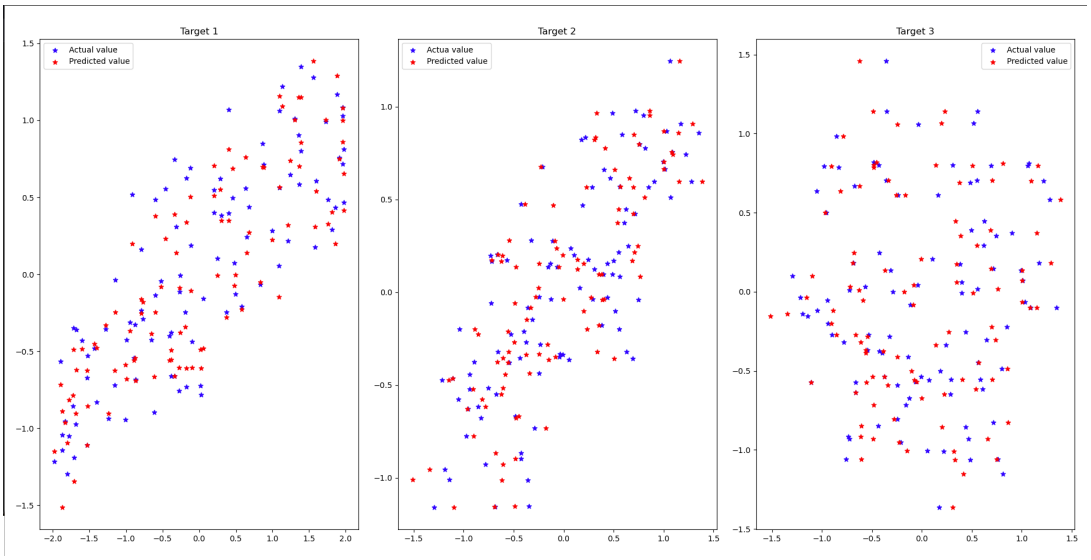

Figure 1: Original vs Predicted Values for Missing Data Imputation.

Task 2: Train on Imputed Data

Techniques Used

- **MLP Regressor** was used instead of Random Forest to train on Original, Random-Imputed, and Regression-Imputed datasets.
- Data was standardized using **StandardScaler** prior to training.

- The MLP Regressor was configured with the following hyperparameters:
  - Hidden Layer: **10 neurons** in a single layer
  - Activation Function: ReLU
  - Solver: Adam optimizer
  - Iterations: 500 (max)

Results

Table 2: MSE Results for Models Trained on Imputed Datasets.

Dataset	**MSE Score**
Original Data	0.031
Random Imputation	0.055
Regression Imputation	0.030

Task 3: Reconstruction of Images

Techniques Used

- **PCA**: Reduced the MNIST image dimensions from 784 to 3 components.
- **MLP Regressor**: A Multi-Layer Perceptron with hidden layers (10, 50) reconstructed the images from PCA-reduced data.

Results

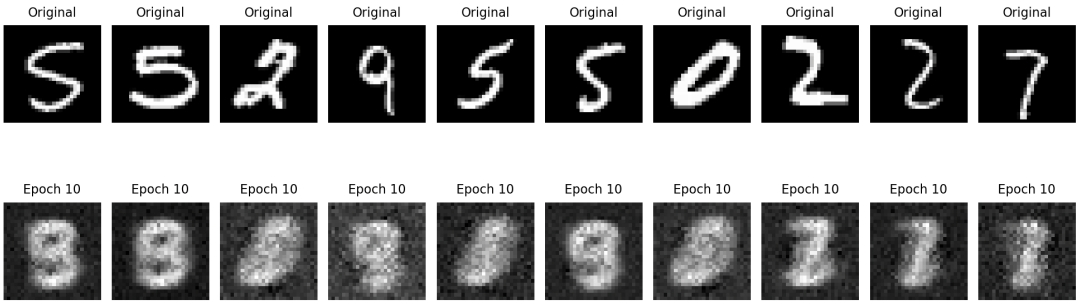

Figure 2: Original vs Reconstructed Images (Epoch 10).

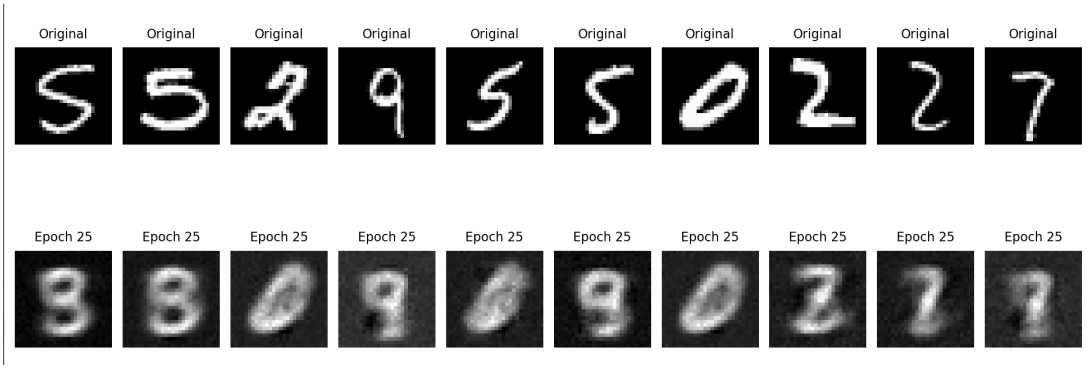

Figure 3: Original vs Reconstructed Images (Epoch 25).

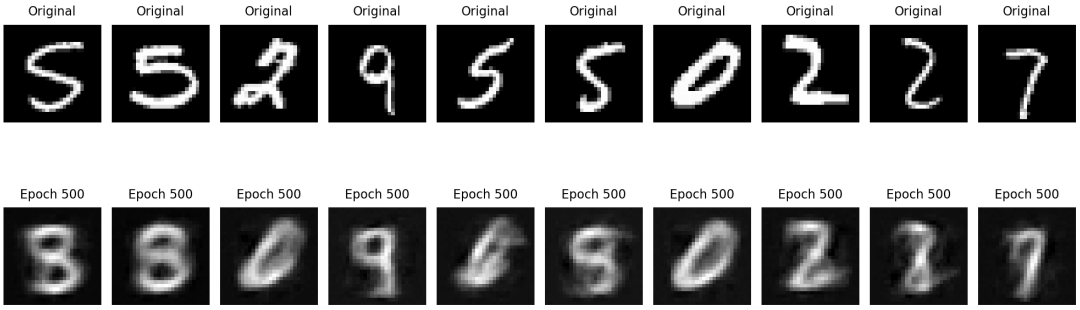

Figure 4: Original vs Reconstructed Images (Epoch 500).

Task 4:Cluster Sampling

*Techniques Used

- **KMeans Clustering**: Sparse clusters were identified in the data generated using the `make_classification` function.
- **Sampling Methods**:
  - **Single-Stage Sampling**: Data points were sampled from selected clusters.
  - **Double-Stage Sampling**: Data points were sampled with a fixed number of points per selected cluster.

Results

Table 3: Testing Accuracy and Training Time for Sampling Methods.

Method	**Accuracy**	**Training Time (ms)**
Original Data	0.888	2950.393
Single-Stage Sampling	0.868	1750.346
Double-Stage Sampling	0.840	749.138

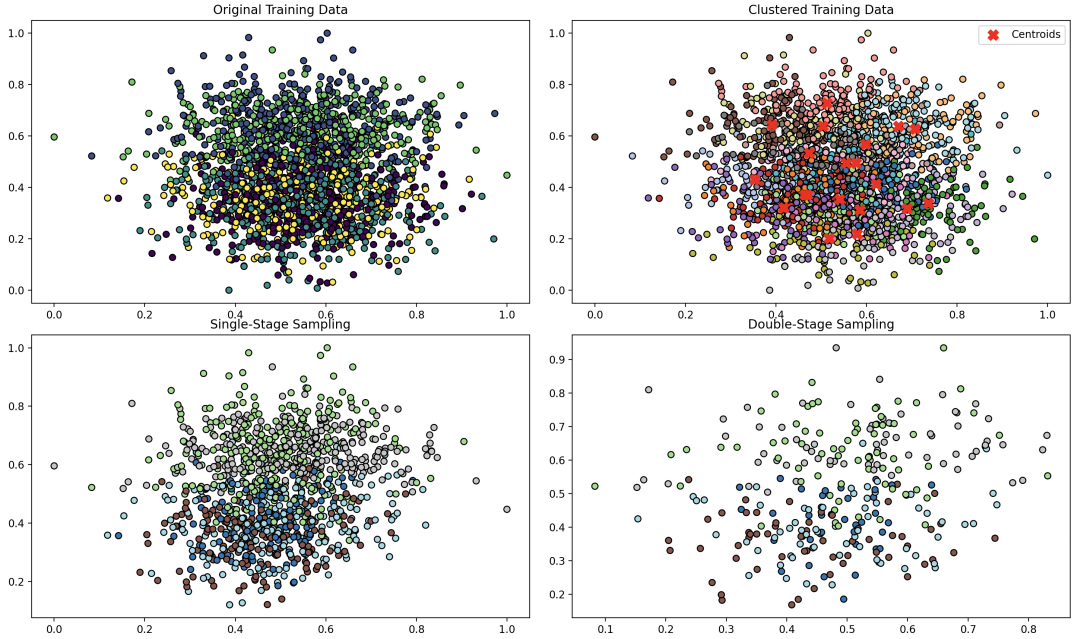

Figure 5: Original vs Predicted Values for Missing Data Imputation.

Task 5: Novelty Detection

Task 5: Novelty Detection

Techniques Used

- **Dataset**: SMS Spam Collection dataset (ham/spam).
- **Text Preprocessing**: Non-alphanumeric characters removed, text converted to lowercase.
- **TF-IDF Vectorization**: N-grams (1, 2, 3), stop words removed, minimum document frequency = 1.
- **Support Vector Classifier (SVC)**: SVM with RBF kernel and class balancing.

Results

Table 4: Performance Metrics for Spam Classification.

Metric	Value
TP:	128
TN:	964
FP:	2
FN:	21
Accuracy:	0.979