Project Description Document

Part A: Numerical Dataset (Regression Models)

Dataset: Bank Loan Prediction

Model 1: Linear Regression

---

a. General Information on Dataset

The numerical dataset used in this part is the **Bank Loan Prediction dataset**.
It contains customer financial and personal information and is used to predict whether a customer will accept a personal loan.

- **Target variable:** Personal.Loan (0 = No, 1 = Yes)
- **Total number of samples:** Dataset contains customer records after removing irrelevant columns.
- **Features:** Numerical and categorical attributes related to income, education, family size, and banking information.

The dataset was split into:

- **Training set:** 80%
- **Testing set:** 20%

No separate validation set was used; instead, cross-validation was applied during loss curve analysis.

---

b. Implementation Details

Feature Extraction and Processing

- Columns ID and ZIP.Code were removed as they do not contribute to prediction.
- Missing values were checked, and no significant missing data was found.
- **Encoding:**
  - Education was encoded using **Ordinal Encoding**.
  - Family was encoded using **One-Hot Encoding**.
- **Polynomial Features:**
  - Polynomial features of degree **2** were generated to capture non-linear relationships.
- **Feature Scaling:**
  - Standardization was applied using **StandardScaler**.

The final feature matrix includes polynomial-expanded and scaled features.

Cross-Validation

- **Cross-validation used:** Yes
- **Number of folds:** 5-fold cross-validation
- **Purpose:** Used to generate training and validation loss curves.

Model Hyperparameters

- **Model:** Linear Regression
- **Regularization:** L2 Regularization
- **Optimizer:** Closed-form solution (library default)
- **Learning rate / epochs:** Not applicable

c. Results Details

The Linear Regression model achieved the following results on the testing data:

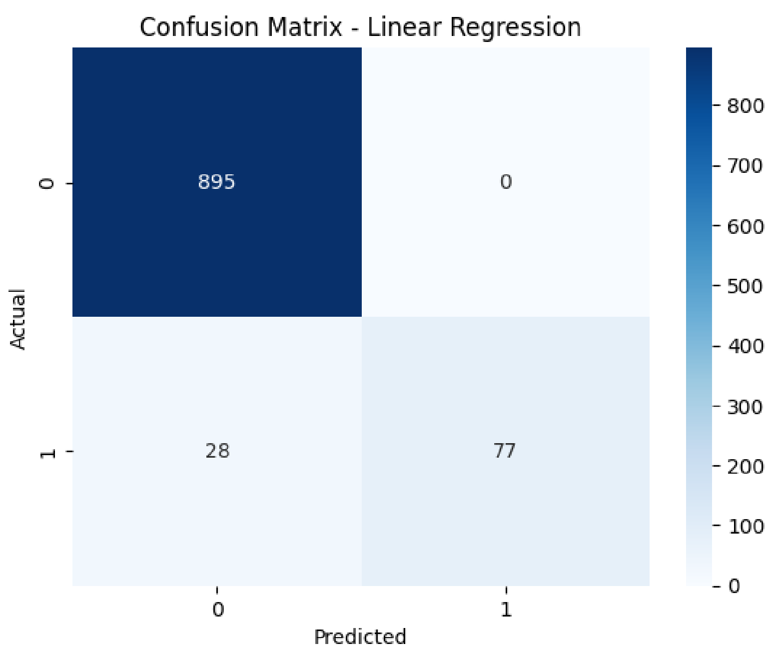
- **R^2 Score:** ~0.67
- **MSE:** ~0.031
- **MAE:** ~0.11
- **RMSE:** ~0.18

To evaluate classification-like performance, predictions were thresholded at **0.5**:

- **Accuracy:** ~97%

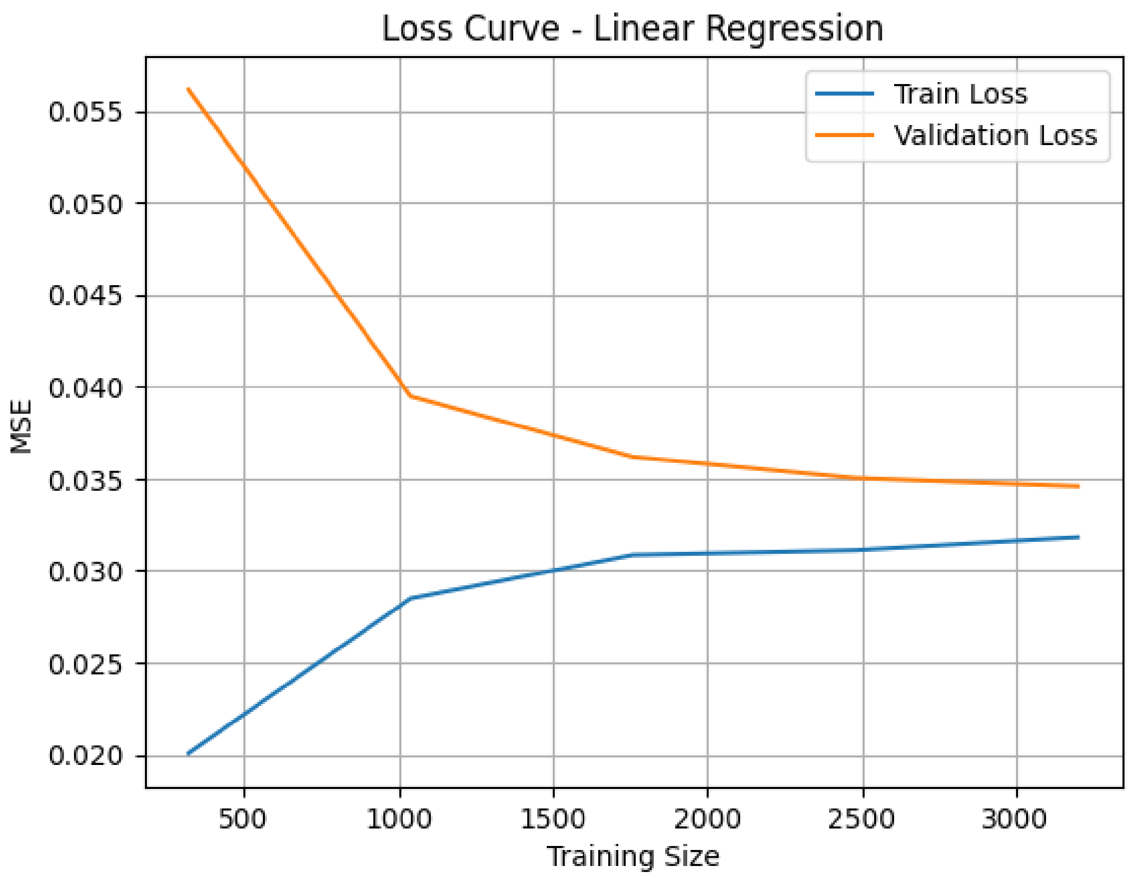
Confusion Matrix

The confusion matrix shows that most loan approvals and rejections were predicted correctly, with very few misclassifications.



Loss Curve

The loss curve shows that both training and validation errors decrease as the training size increases, indicating stable learning and no major overfitting.



Model 2: KNN Regressor

---

a. General Information on Dataset

The same **Bank Loan Prediction dataset** was used for the KNN regressor with the same preprocessing steps and data split.

b. Implementation Details

Feature Extraction

- The same polynomial features and scaled inputs used for Linear Regression were reused.
- This ensures fair comparison between models.

Cross-Validation

- **Cross-validation used:** Yes
- **Number of folds:** 5

Model Hyperparameters

- **Model:** K-Nearest Neighbors Regressor
- **Number of neighbors (k):** 7
- **Distance metric:** Euclidean (default)
- **Weighting:** Uniform

---

c. Results Details

The KNN Regressor produced the following results:

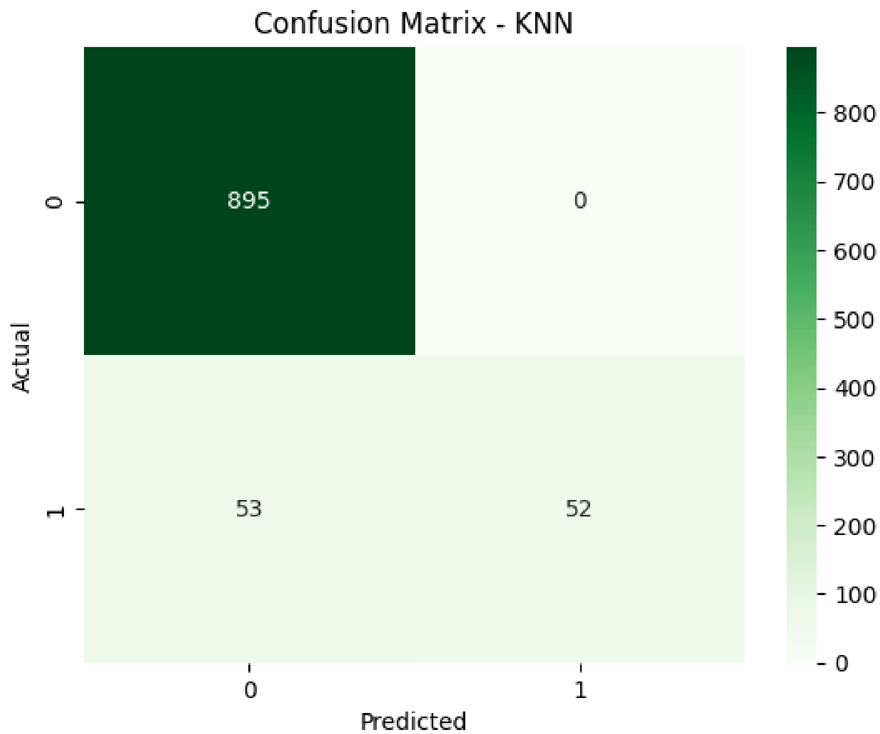
- **R^2 Score:** ~0.62
- **MSE:** ~0.036
- **MAE:** ~0.06
- **RMSE:** ~0.19

Using the same thresholding approach:

- **Accuracy:** ~95%

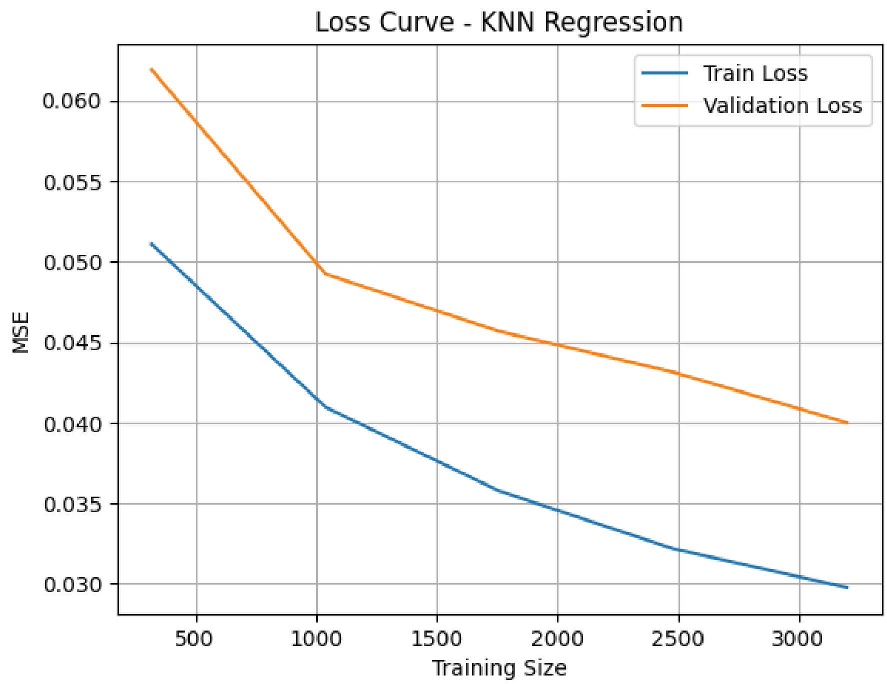
Confusion Matrix

The confusion matrix shows good performance, though slightly weaker than Linear Regression, especially near the decision boundary.



Loss Curve

The loss curve indicates that KNN benefits from more training data but shows higher variance compared to Linear Regression.



Comparison and Discussion

- Linear Regression achieved **higher R^2 and accuracy** compared to KNN.
- KNN showed lower MAE but higher overall error variance.
- Linear Regression provided more stable and consistent performance on this dataset.

Overall Conclusion (Numerical Dataset)

Both regression models performed well on the Bank Loan dataset. However, **Linear Regression** achieved better overall performance and stability, while **KNN** was more sensitive to data distribution and parameter selection.

Part B: Image Dataset (Classification Models)

Dataset: Fashion-MNIST

Model 3: Logistic Regression

a. General Information on Dataset

The image dataset used in this part is **Fashion-MNIST**, which consists of grayscale images of fashion items.
Each image has a resolution of **28 × 28 pixels**, represented as flattened pixel values.

Originally, the dataset contains **10 classes**, but only **5 classes** were selected to simplify the classification task:

Label	Class
1	Trouser
3	Dress
5	Sandal
6	Shirt
8	Bag

Dataset size:

- **Original training samples:** 60,000

- **Original testing samples:** 10,000

After filtering:

- **Training samples:** 30,000
- **Testing samples:** 5,000

The training data was split into:

- **80% training (No of samples:** 23,985**)**
- **20% validation (No of samples:** 5997**)**

Duplicate samples were removed, and pixel values were normalized to the range **[0, 1]**.

---

b. Implementation Details

Feature Extraction

- **Original features:** 784 pixel values per image
- **Normalization:** Pixel values divided by 255

To reduce dimensionality, **PCA (Principal Component Analysis)** was applied:

- **Number of PCA components:** 200
- **Final feature dimension:** 200

Cross-Validation

- Cross-validation was **not used**
- A fixed **train/validation split (80/20)** was applied

Model Hyperparameters

- **Model:** Logistic Regression
- **Solver:** LBFGS
- **Regularization:** L2 (default)

- **Maximum iterations:** Default

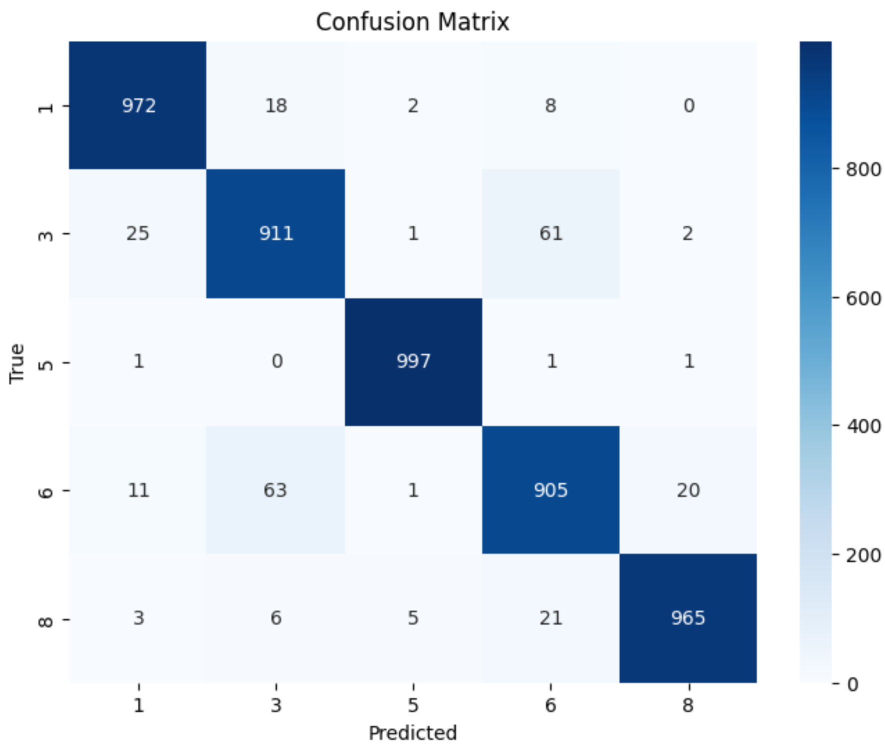
---

c. Results Details

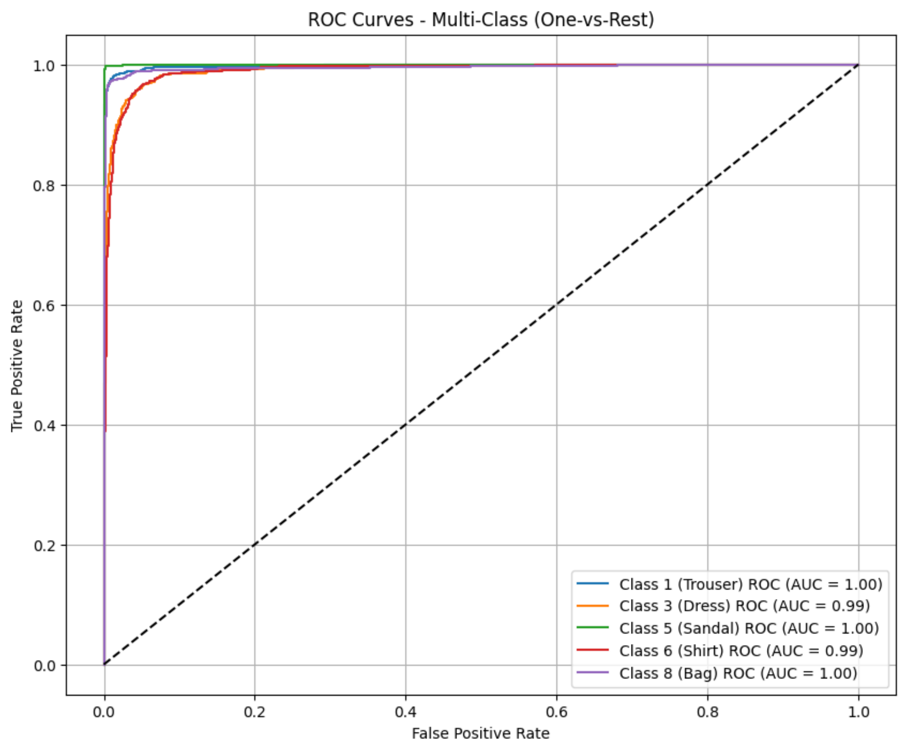
Performance on testing data:

- **Training accuracy:** ~96%
- **Validation accuracy:** ~95%
- **Testing accuracy:** ~95%

The confusion matrix shows strong classification performance, with most errors occurring between visually similar classes such as *Dress* and *Shirt*.



ROC curves were plotted for all classes, achieving a **macro ROC-AUC score of ~0.99**, indicating excellent class separation.



Model 4: K-Means Clustering

---

a. General Information on Dataset

The same **Fashion-MNIST filtered dataset** with 5 classes was used. PCA-reduced features (200 dimensions) were used as input.

---

b. Implementation Details

- **Algorithm:** K-Means
- **Number of clusters:** 5
- **Maximum iterations:** 100
- **Random state:** 42

Each cluster was mapped to a class label based on the **majority class** inside the cluster.

Cross-Validation

- Cross-validation was **not used**

---

c. Results Details

Model performance:

- **Training accuracy:** ~71%
- **Validation accuracy:** ~71%
- **Testing accuracy:** ~70%

The confusion matrix shows that K-Means can capture general patterns but struggles to fully separate similar classes due to its unsupervised nature.

Overall Conclusion (Image Dataset)

Logistic Regression significantly outperformed K-Means on the image dataset.
While K-Means provided reasonable clustering results, supervised learning proved to be much more effective for image classification tasks.