TIA 2024 Competition Report

Șeicărin Alin Ștefan Tudorică Antonio-Adrian

22nd of December 2024

1 Introduction

The goal of this project is to analyze and classify images from a truncated version of the Fashion MNIST dataset.

The dataset consists of 56,000 grayscale images divided into 10 classes, representing different categories of clothing and accessories. This report documents the entire process, including data analysis, statistical evaluation, model development, and results.

2 Data Analysis and Statistical Analysis

2.1 Dataset Overview

The dataset used in this study is the Fashion MNIST dataset, consisting of 56,000 grayscale image samples, each labeled in one of 10 classes:

0: T-shirt

1: Trousers

2: Pullover

3: Dress

4: Coat

5: Sandal

6: Shirt

7: Sneakers

8: Bag

9: Ankle boot

Each image is 28x28 pixels, flattened into a single vector of 784 features. The dataset provides a balanced distribution of samples across the 10 classes, as shown in Table 1.

2.2 Class Distribution

To understand the structure of the dataset, we examined the distribution of samples across the 10 classes. The dataset is mostly balanced. The class distribution is summarized in Table 1.

Table 1: Class Distribution in Training Data

Class ID	Class Name	Count
0	T-shirt/top	4,499
1	Trouser	4,487
2	Pullover	4,465
3	Dress	4,428
4	Coat	4,449
5	Sandal	4,462
6	Shirt	4,468
7	Sneaker	4,560
8	Bag	4,544
9	Ankle boot	4,393

2.3 Data Visualization

To gain an intuitive understanding of the data, Principal Component Analysis (PCA) was applied to reduce the dimensionality of the dataset to two components. Figure 1 visualizes the distribution of the data in this reduced feature space, showing the overlap and separability among classes. This indicates that the dataset is not entirely linearly separable.

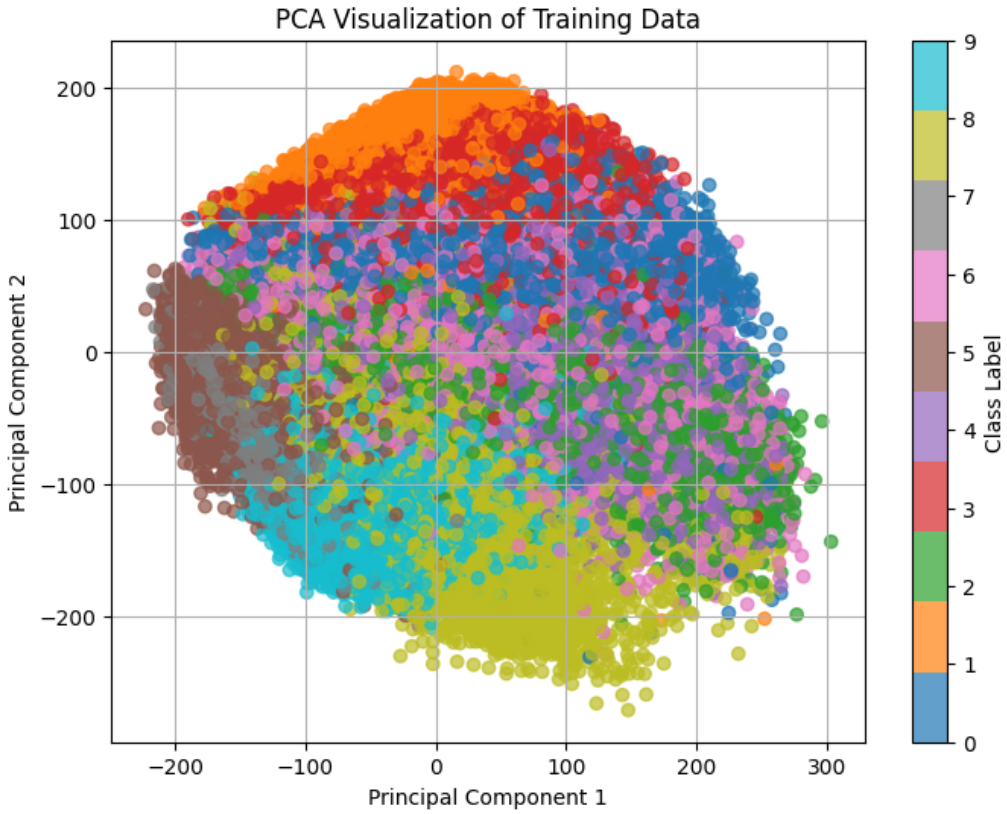

Figure 1: PCA Visualization of the dataset

2.4 Statistical Insights

The PCA visualization in Figure 1 highlights the overlap between certain classes, such as "T-shirt/top" and "Shirt," suggesting that these categories may share similar feature distributions.

3 Methodology

3.1 Model Selection

We chose to use an SVM model, which we think to be very suitable given the nature of the dataset. The training set consists of 56.000 flattened 28x28 grayscale images, meaning a matrix of 56.000 samples, each having 784 features. Given the high dimension of the set, we chose SVM due to the fact that it focuses on finding the optimal separating hyperplane, and is thus not necessarily affected by the large number of input features. Moreover, the images are visually similar, meaning the boundaries would have to be non linear, so we went with an RBF kernel.

In order to achieve better performance, we performed hyperparameter tuning using GridSearchCV to optimise the regularization parameter C and the kernel coefficient γ.

3.2 Model Implementation

The training set consists of 56,000 samples, each represented as a vector of 784 features. The following steps were performed:

- Normalized pixel intensities to the range [0, 1] to improve model convergence.
- Applied an RBF kernel to handle non-linear separability.
- Used GridSearchCV for hyperparameter tuning, optimizing the regularization parameter C and kernel coefficient γ.

4 Performance

In order to test the performance of the model, we used a number of metrics, specifically accuracy, precision, recall and f1.

- Accuracy is the proportion of correctly predicted labels out of the total number of predictions . It's a quick overview of the overall performance, but seeing as we called all the score functions using the average = 'weighed' parameter, it's quite a reliable metric
- Precision is the proportion of true positive predictions out of all positive predictions, it accounts for false positives, ensuring that the model is not overly confident in its predictions without being accurate
- Recall is the proportion of true positives out of all actual positives. It accounts for false negatives in the same way that precision accounts for false positives, and thus reflects the model's ability to detect all relevant instances for each class
- F1 is the harmonic mean of precision and recall, useful for when the two need to be balanced, or when there are imbalances in the classes

5 Interpreting the results

Initial performance scores were around 0.85, which the model achieved using a default C of 0.1, default γ, and unscaled data in general. Then, after normalising the data we saw a strong increase in performance, up to around 0.88, and the final score of 0.906 we achieved by running GridSearchCV and finding the optimal $C = 6$.

6 Error Analysis

Misclassifications were primarily observed in visually similar classes, consistent with the statistical and PCA insights. These findings suggest that further feature engineering or alternate kernels could improve results.