TAA Group Project - Delivery 2

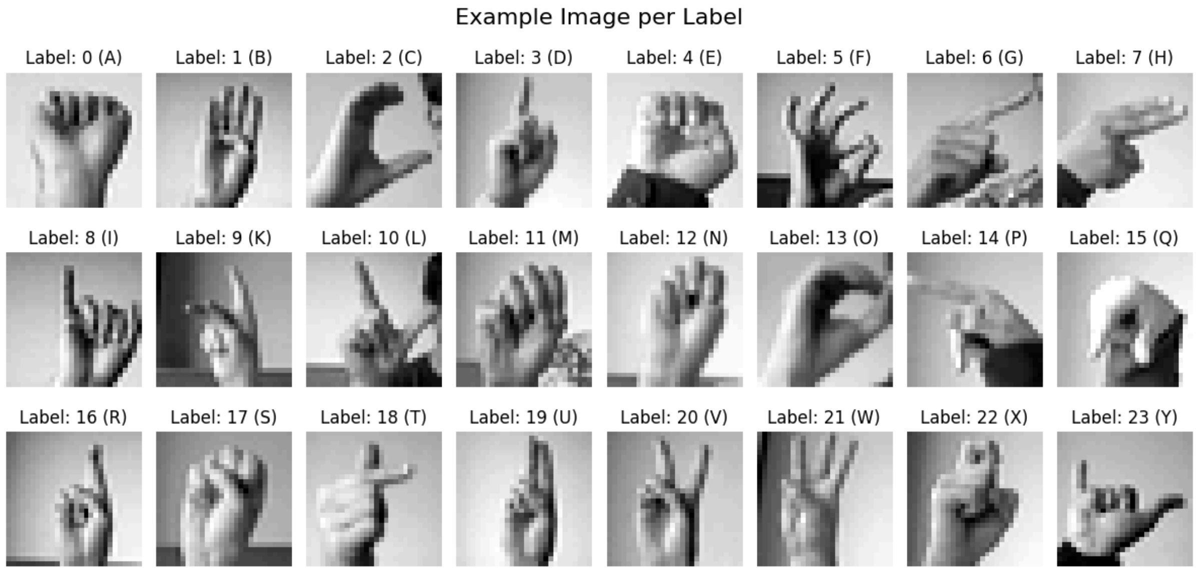
Rodrigo Abreu, 113626
Eduardo Lopes, 103070
João Neto, 113482

Data description

The American Sign Language letter database of hand gestures represents a multi-class problem with 24 classes of letters (excluding J and Z which require motion). For the development of the American Sign Language Understanding project we used the *Sign Language MNIST* dataset.

The dataset has two files: 1 training file and a test file. Each training and test case represents a label (0-25) as a one-to-one map for each alphabetic letter A-Z (and no cases for 9=J or 25=Z because of gesture motions). The training data (27,455 cases) and test data (7172 cases) have a header row of label, pixel1,pixel2….pixel784 which represent a single 28x28 pixel image with grayscale values between 0-255.

Each data image contains a hand expressing the correspondent letter sign with different angles, backgrounds and light exposures.
Image example per class/letter:



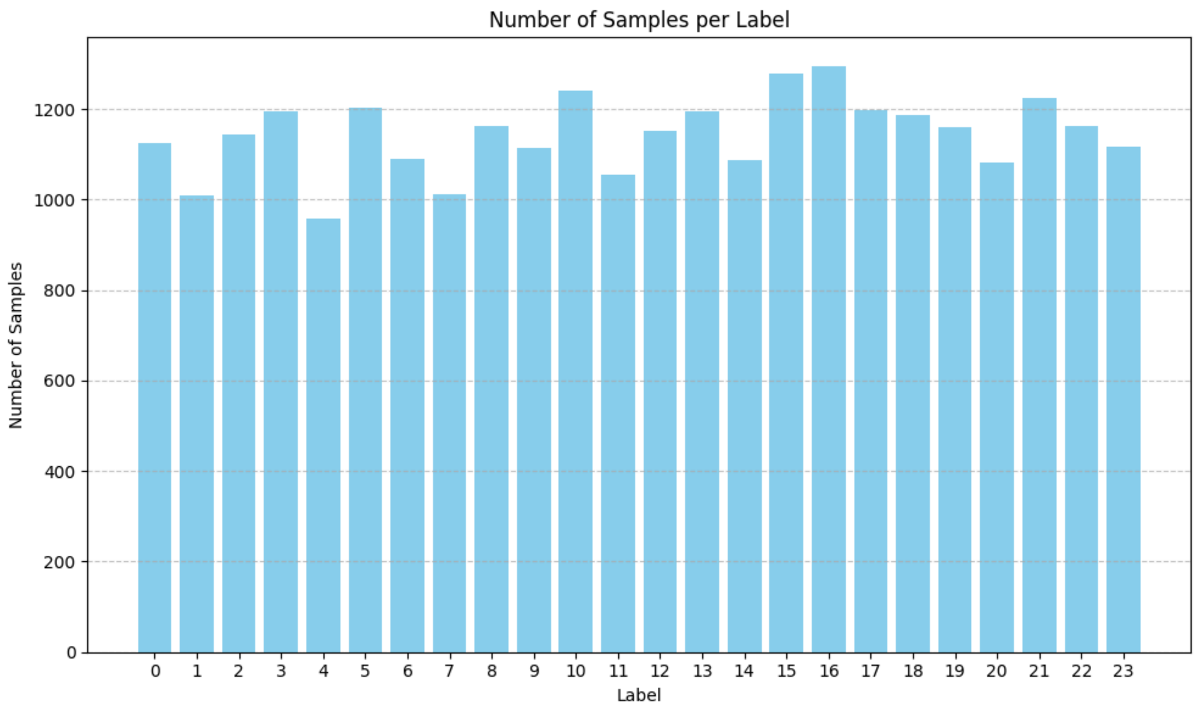
Data Handling Methodology

To handle the data we start by downloading the training data and the test data (csv format). Then we load the data to a variable, storing it in memory, then we separate the training data and respective labels in different variables, and repeat the same procedure for the test data.

We apply normalization on the data by dividing each pixel value by 255 so that each value is in the [0,1] range. This helps to improve performance and convergence speed of the algorithms.

We then removed the label gaps caused by signals that are not present in the dataset, and went from a label of 0 to 25 with gaps on 9 (“j”) and 25 (“z”) to a label of 0 to 23 without gaps. After this, we count each class sample on the training data and put the data on a plot to analyse the dataset balance.

We got the following results:



Even though each class does not have the same number of samples, it looks balanced enough.

Selected Algorithms

After analysing the dataset, we discussed Machine Learning algorithms that could be suitable for this problem. Since the objective was to classify accurately the letter represented by the hand sign represented on the image, classic multiclass algorithms are suitable for this approach, so we selected the following:

- **One vs All with Logistic Regression** - Easy to implement and interpret. Performs surprisingly well when each class is clearly separable (which is the case).
- **SVM** - The algorithm maximizes the margin between the closest points of different classes (which often results in good generalization). With the right values, SVMs can avoid overfitting even in high-dimensional data.
- **KNN** - Assigns a class based on the majority vote of its K closest neighbors in the feature space. It is effective for image data because it leverages similarity in pixel patterns.
- **Softmax Regression** - Generalization of logistic regression for multiclass problems using the softmax function. Suitable for linearly separable multiclass problems.

After this, we also tried a deep learning implementation using **Convolutional Neural Networks with Keras** in order to get better accuracy results. This algorithm automatically learns spatial hierarchies of features from raw images, making them ideal for visual recognition tasks.