# Introduction to Machine Learning
# Assignment 1

Group 06
Lucas Pereira (s4507983) & Matteo Wohlrapp (s4974921)

September 23, 2021

## Introduction

In this practical, we applied the Learning Vector Quantization to a data set which is composed of 100 data points in a 2D plane. The training was based on prototypes, with the squared euclidean distance as a reference. The data is divided into two different classes, which are represented in Figure 1.

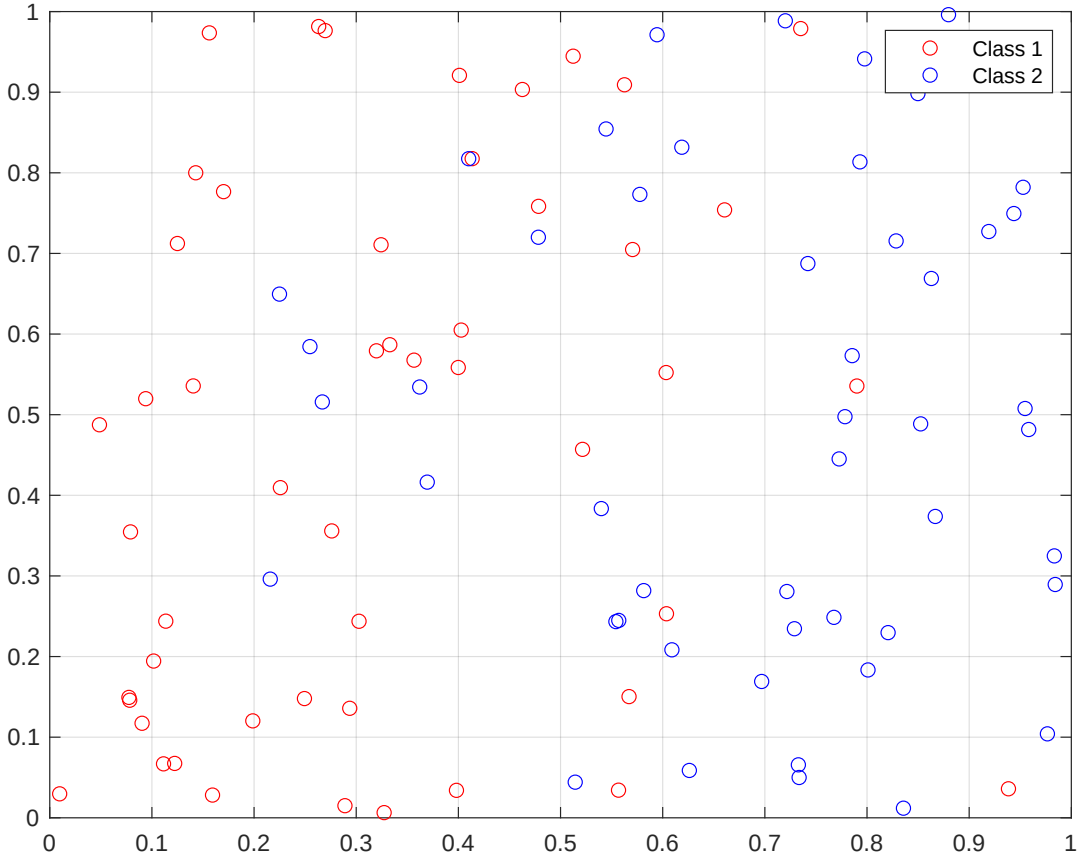


**Figure 1:** *Visualization of the classification of the provided data*

The prototypes are randomly picked from the data set, with an equal amount of prototypes for each class. During the practical, we experimented with different amounts of prototypes, as well as

varying factors for the learning rate and considerations for constant error rates. As a bonus to two and four prototypes, we also included a version with four prototypes per class.

## Method

To solve the given problem, we use MATLAB. Our approach firstly includes loading the respective data, then classifying it. That means we assign the first 50 of the 100 data points to class 1, and points 51 to 100 to class 2. Afterwards, we choose random points, an equal amount for both classes, as prototypes. Then we simulate epochs until the ratio of errors is constant or the maximum amount of epochs, which we declare as 300, is reached. A constant error ratio is defined as a period where the error ratios of 50 consecutive epochs don't change.

In each epoch we then iterate randomly over the data points and search the next closest prototype. If the prototype is in the same class as the data point, the psi-function returns 1 and therefore the prototype moves closer to the data point, otherwise it moves further away. After each epoch, we track the amount of wrong classifications in comparison to the original one.

In general it is important to keep the code very flexible on input constants. That enables us to increase the size of prototypes, the learning rate, or the definition of constant error rate easily.

## Results

In the following section, we present our findings by means of graphs depicting either the visualization of the classification of the provided data and the respective prototypes or the learning curves regarding the training epochs. The learning curves and LVQ1 label graphics were both retrieved from the same training session.

### 0.1. Learning curves

The following subsection depicts the learning curves for training sessions with one, two and four prototypes per class. The learning curve describes how many wrong classification we assume for each epoch, following the distance based classification with prototypes, normalized to the interval [0,1].

0.1.1. (A)

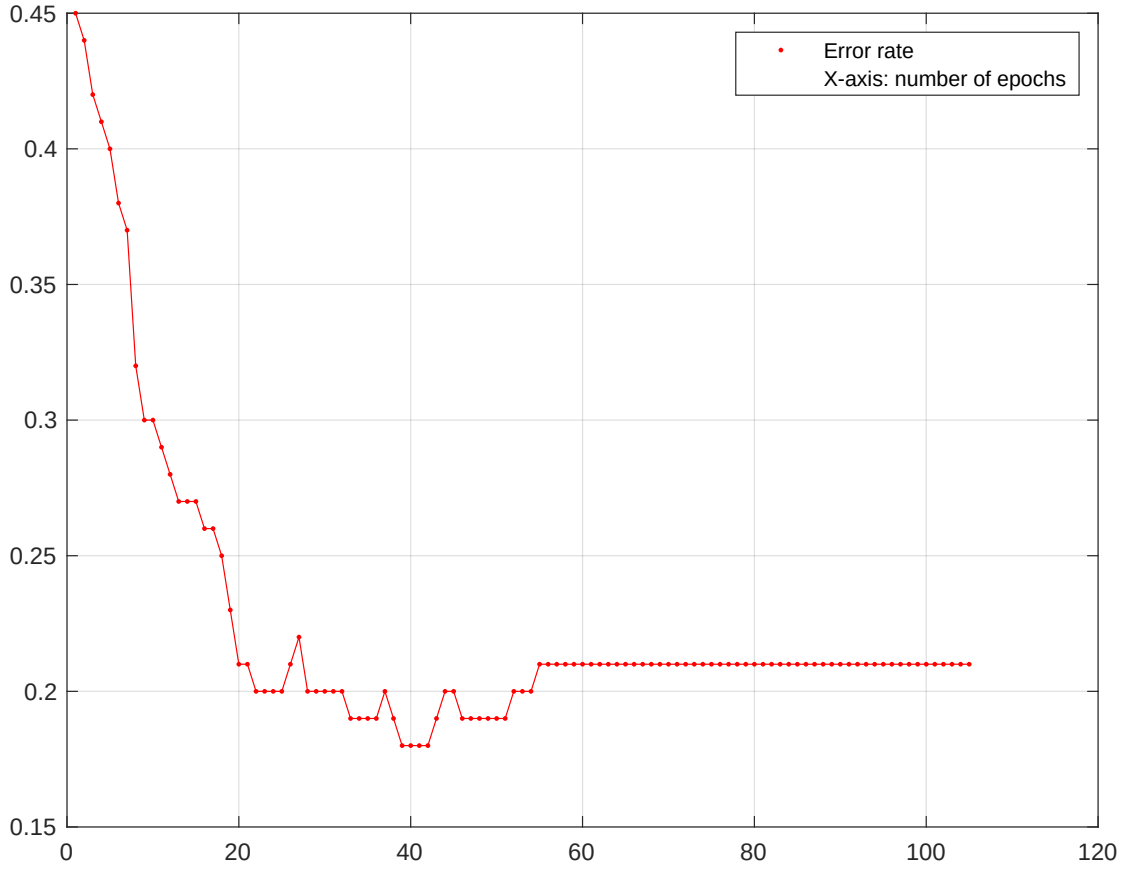


**Figure 2:** *Visualization of the learning curve with two prototypes*

0.1.2. (B)

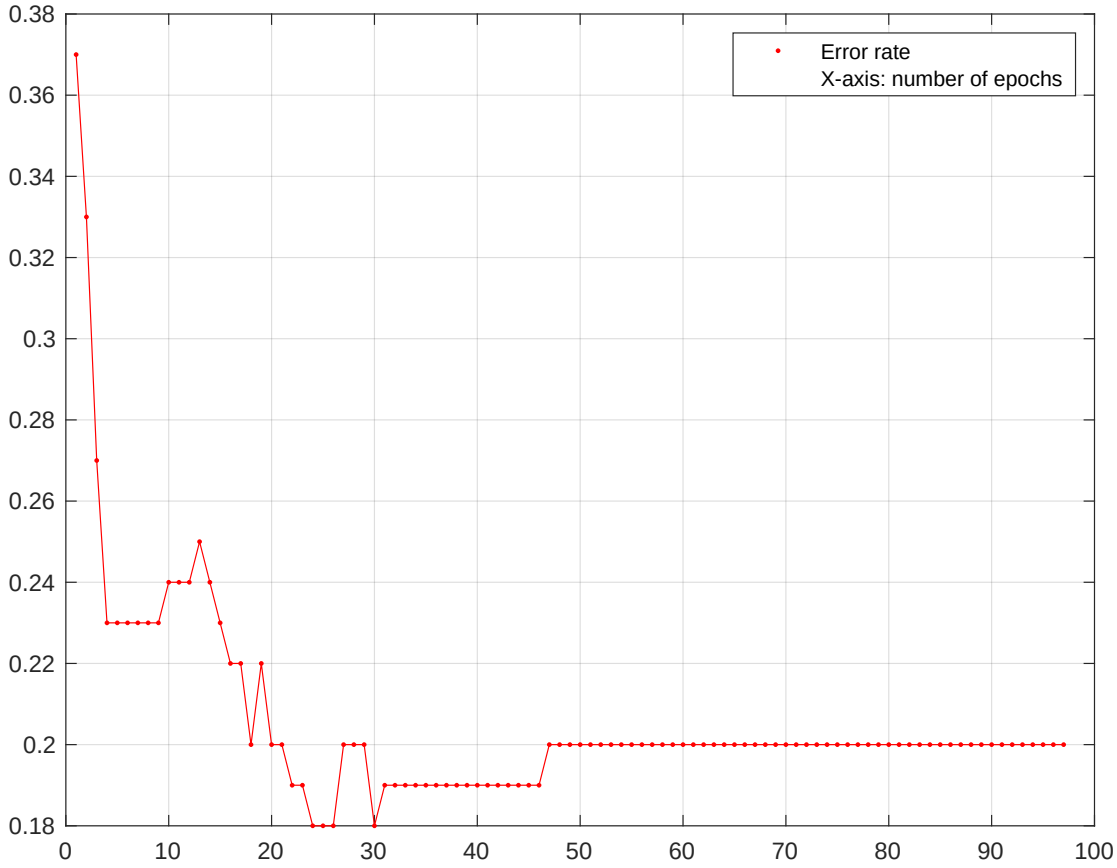


**Figure 3:** *Visualization of the learning curve with four prototypes*

### 0.1.3. (Bonus)

As a bonus, the following figure shows the learning curves for training with four prototypes in each class.

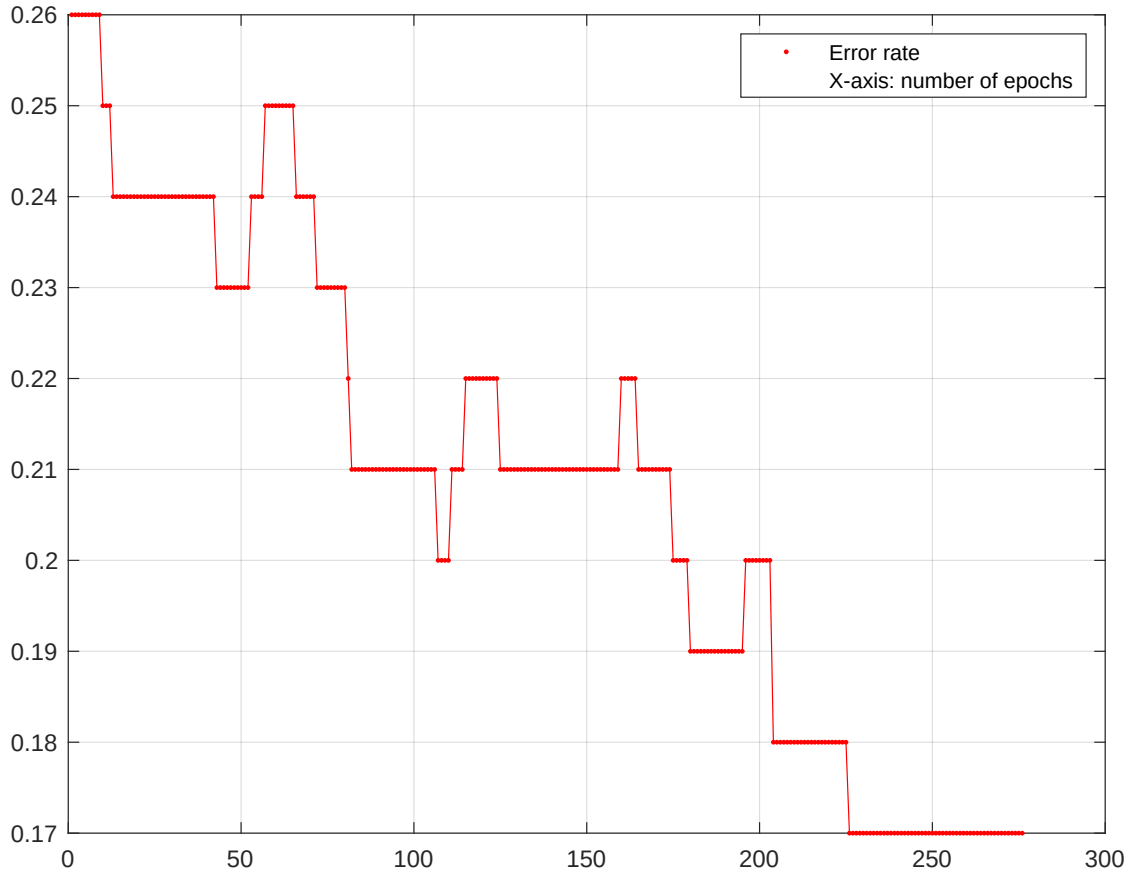


**Figure 4:** *Visualization of the learning curve with eight prototypes*

## 0.2. Data with LVQ1 labels

The following subsection depicts the LVQ1 labels for the given data set after the training. In comparison to Figure 1, you can clearly see that some labels are labeled different than originally intended. Due to the fact, that the corresponding error rate is around 20% for both (B) and (Bonus), you can assume that roughly 20 data points are coloured differently.

### 0.2.1. (B)

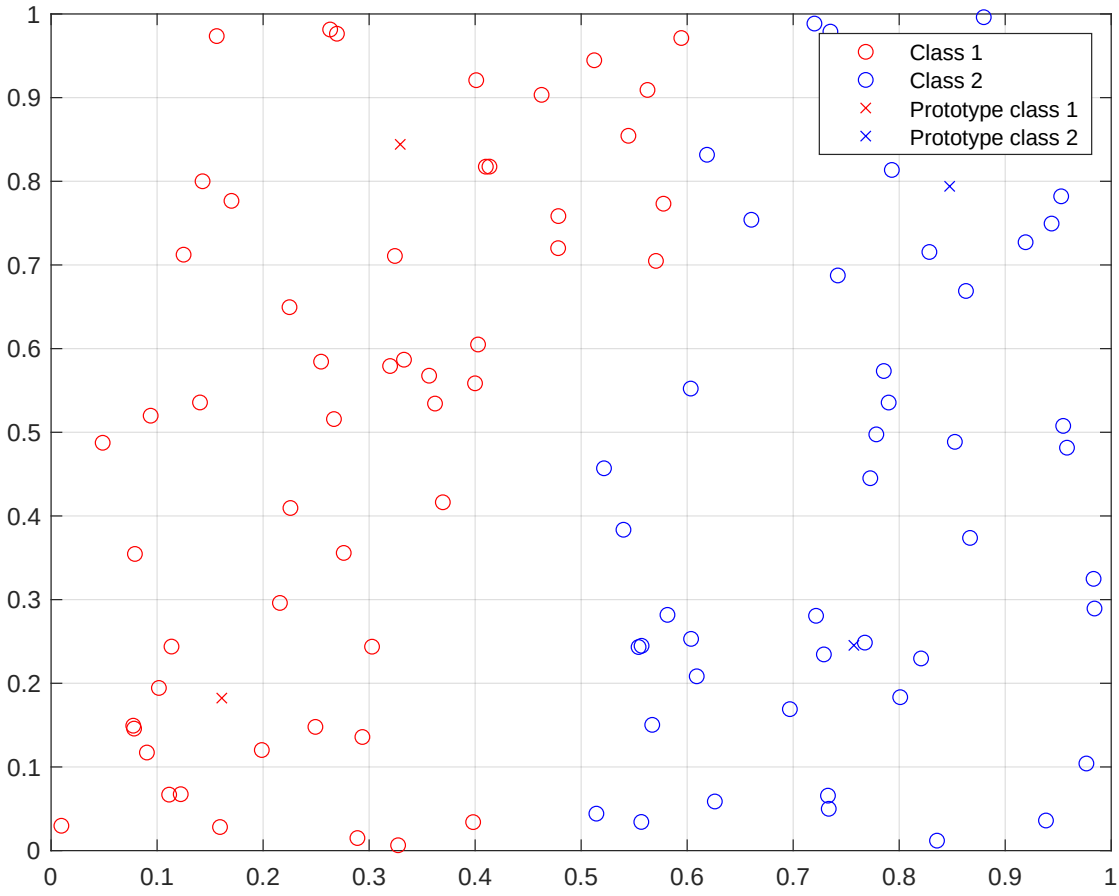


**Figure 5:** *Visualization of the LVQ1 labels with four prototypes*

### 0.2.2. (Bonus)

As a bonus, we added the labels for training with four prototypes for each class.

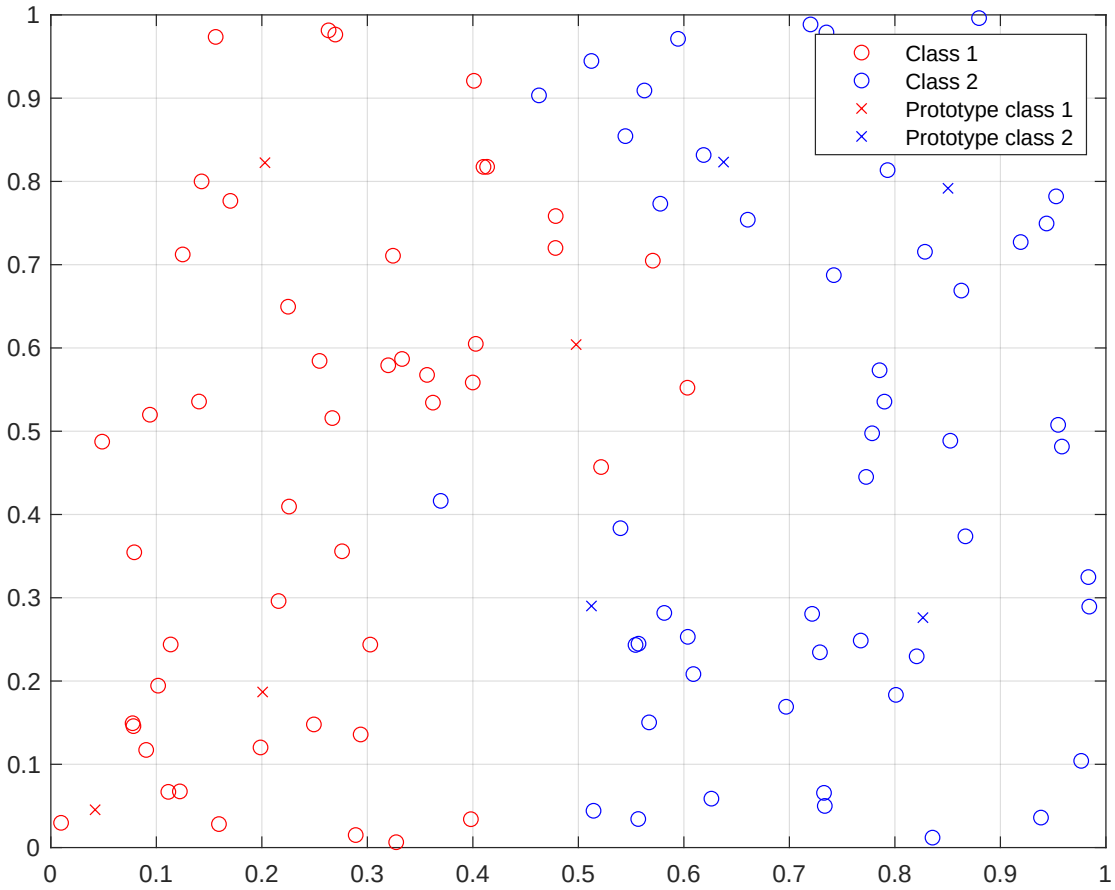


**Figure 6:** *Visualization of the LVQ1 labels with eight prototypes*

## Discussion

As expected, the general trend of the learning curves is decreasing until it reaches a certain threshold. Suggested by the figures and empirical observation, that values seems to be roughly at 20%. Through trial and error, we tried to verify the suggested learning rate and found as expected, that 0.002 is an appropriate value. If the rate is too big, the prototypes move more during the course of continuous epochs, while a value which is too small accounts for very few movement and therefore a higher number of needed epochs to reach a constant error rate.

We also found, that the choice of prototypes has a grand impact on the LVQ1 labels and the learning rate. While we discovered that with only two prototypes the result seems very predictable, in that the prototypes roughly move to the same position for different starting positions, a higher number of prototypes led to a more chaotic outcome with different ratios class one and class two classifications.

## Workload

The workload was shared equally, with Lucas Peireira focusing more on the mathematical part, while Matteo Wohlrapp focused more on the graphical and textual aspects of the work. However, the results were discussed in detail and changes implemented in a team environment.