AI RESEARCH EXERCICE REPORT

Derbel Mohamed aziz
January 2, 2023

Abstract

This papers demonstrates the followed work methodology and its outcomes.

1 Introduction

- Pytorch deep learning framework was used all along the exercice
- Development environment : Python 3.10.8 , VsCode editor
- Tools and libraries : pandas, matplotlib, seaborn, numpy
- Estimated time : 2h

2 Workflow

2.1 Question 1 and 2

- The MNIST dataset was imported using the the torchvision repository
- A convolutional autoencoder was created with 2 convolutional blocks and with an embedding size of 2
- the model was trained during 100 epochs with a learning rate of 0.004

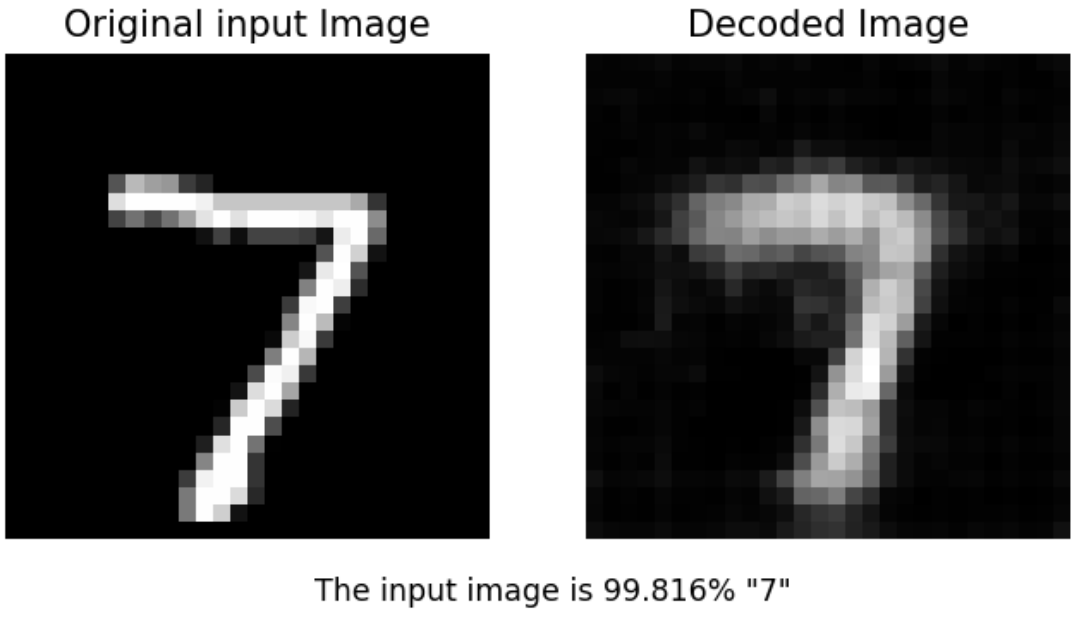

Figure 1: The autoencoder performance

2.2 Question 3

My method consists of 2 parts

- Create the feature space using the trained autoencoder
- Train a seperate classifier on the generated feature space

To predict the class of a particular image

- The autoencoder will generate the corresponding latent variable
- The trained classifier will try to match the correct label or class

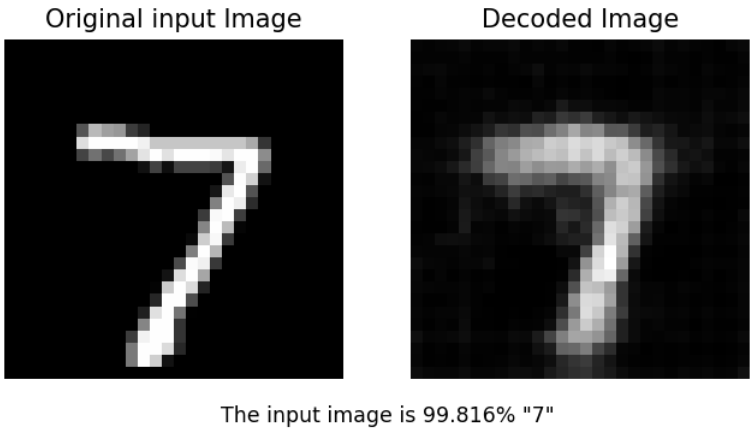

Figure 2: The autoencoder performance

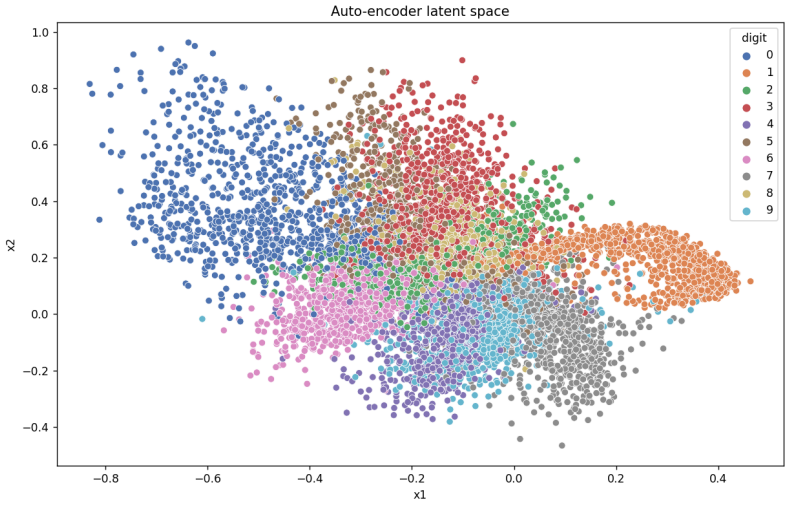

Figure 3: The autoencoder feature space visualization

2.3 Question 4 and 5

- I used the trained classifier to map between the latent space and the probability space

- In fact, this task is sort of generalization of the previous one
- I followed the same pattern but this time for all latent variables present in the latent space.
- To show the class of a particular image, the autoencoder will generate the corresponding latent variable
- Then the trained classifier will try to match the correct label or class
- As a result, the classifier will build the probability space mapped with the latent feature
- I chose this graph, a KDEplot, to visualize probability distributions of all latent variables

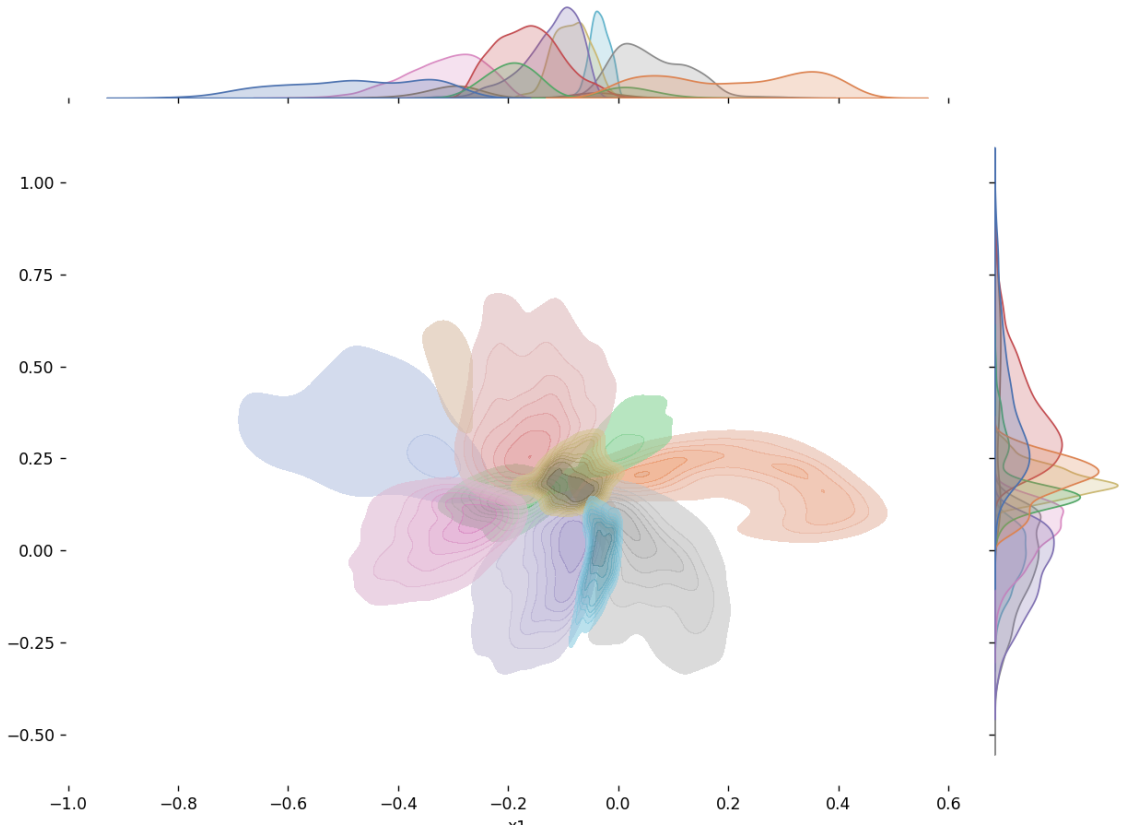

Figure 4: probability space mapped to the latent space