# Deep learning lab-meeting 2- Self supervised learning-Final report

Izar Hasson , Aviv Shem-Tov

## 1.1

We can assume that supervised learning will perform poorly here because there is simply not enough labeled data for training the module.
Its a good fit for self-supervised learning because we have a lot of unlabeled data that can be used to train the module that describes the data and then using the low amount of labeled data to train the last fully connected layer.

## 1.2

RandomCrop- Crop the given image at a random location.

ColorJitter- Randomly changes the brightness, contrast, saturation, hue, and other properties of an image.

RandomGrayscale- Randomly convert image to grayscale with a probability of p.

RandomHorizontalFlip - Horizontally flip the given image randomly with a given probability.

RandomPerspective- Performs random perspective transform on an image.

RandomRotation- Rotates an image with random angle.

RandomInvert- Randomly inverts the colors of the given image.

## 1.3

ColorJitter, RandomGrayscale, RandomHorizontalFlip, RandomPerspective, RandomInvert.
RandomCrop is not safe because we can crop out the object itself and the module will think that the background is a positive example, which we know will worsen the module.

## 1.4

Our batch size is 256, batch size will mainly affect the training time, the bigger the batch the faster and the other way around. The advantage of choosing a relatively small batch size is that the batch will be a bit noisy, and that helps to generalize the model.

## 1.5

The contents of the batch is 256 couples of images with their label, each image of size (3,64,64), when 3 is the channels (RGB) and 64x64 is the height and width of the image. Each couple is considered a positive couple.

2.1

The role of m is how much weight we give to the new parameters, the bigger the m the less weight we give those parameters.

Updated_param = m * param + (1- m) * new_param

3.1

Because in the BYOL model the input is augmented and considered as positive couples, each one of them forwarded in a different path, one in the main model and the other in the momentum path.

5.1

We want to represent the positive couples as close to each other as possible, our distance is determined by the cosine similarity, which means we want to reach distance of 1.
When we train we search for the minimum of the loss function and thus we put negative before the expression, so we are searching for the module that reaches –1.

5.2

When we use the pre-trained model, we already receive a good presentation of the images from it because the module has a prior knowledge of how images look like, thus it needs only few epochs to adapt to the new dataset.
The weights of the pre-trained model will need fine tuning for the model to work even better on the new dataset, we don't want to differ too much from the original weights values thus choosing a smaller starting learning rate can help.

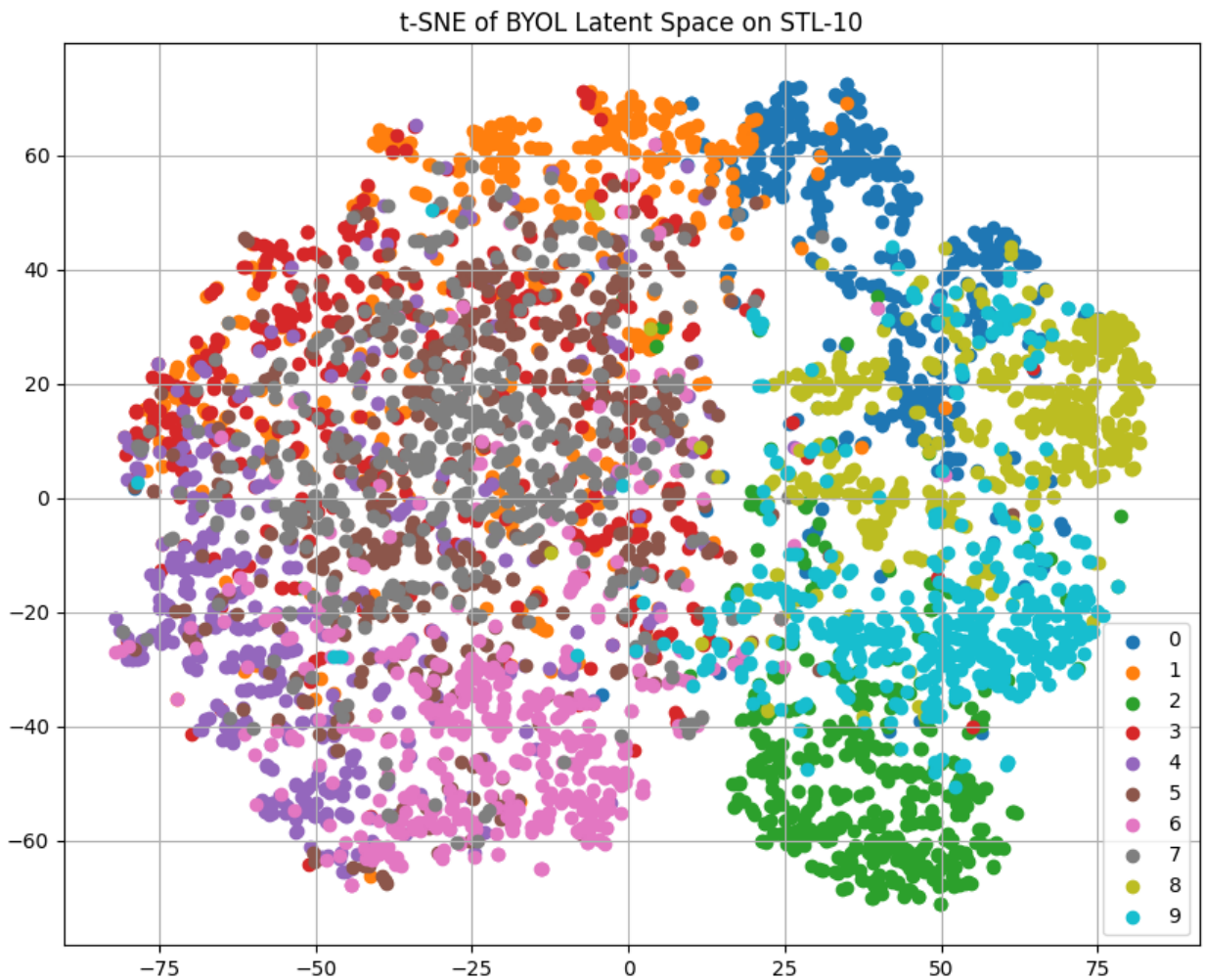
We lowered the learning rate from $3 \cdot 10^{-4}$ to $2 \cdot 10^{-4}$

6.1

We should ideally see the classes separated as clusters.

6.2

The classes are not perfectly separated, but we can see that samples of the same class are close to each other. This happens because the module is trained to represent the same class as closely as possible, but it wasn't trained for increasing the distance between different classes thus letting everything be close to one another.



7.1

We would expect the data to be separated even in the low dimensional space thus making linear classification work well.

7.2

The data could be in different scales, thus making certain features “more important” than others by influencing the loss function more, without a good reason. We solve that by scaling the data before we train the network.

7.3

Our linear classifier with BYOL from scratch (without a pretrained backbone) got a 67.5% train accuracy and 33% test accuracy.

We didn’t finish training BYOL with the pre-trained backbone, but we would assume that the linear classifier will perform better with the pre-trained backbone, because the model was trained to represent images well.