Lab 8 – Exploring Latent Spaces
By: Abhash Shrestha (35464984) as14n23@soton.ac.uk

1. Systematically sample a VAE

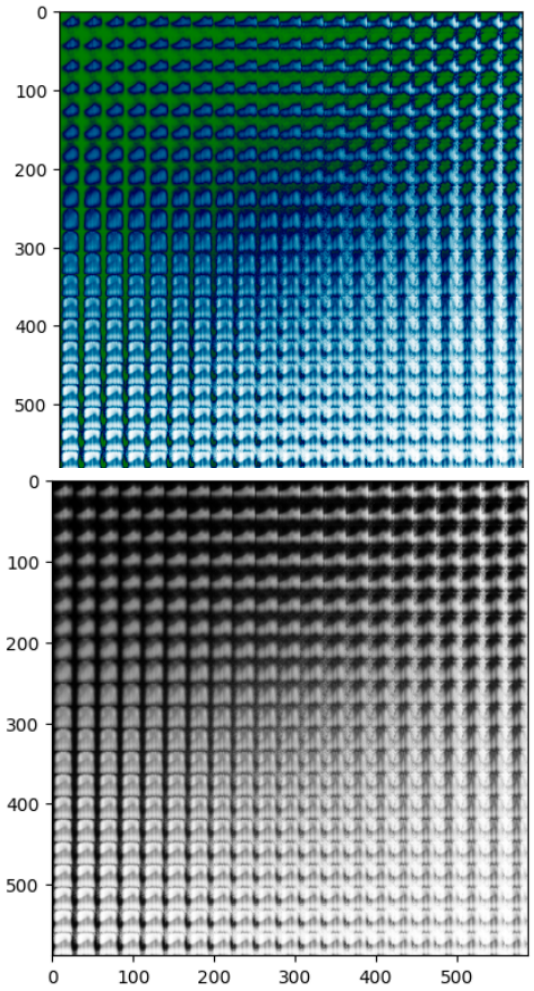

Figure 1 : Latent Images - VAE

Above, we can see the reconstructed image

```
x = np.linspace(-4, 4, 21)
y = np.linspace(-4, 4, 21)
xx, yy = np.meshgrid(x, y)
latent_image = np.zeros((21*28, 21*28))
for j in reversed(range(len(yy))):
    for i in range(len(xx)):
        z = torch.tensor([xx[j, i], yy[j, i]],
                         dtype=torch.float32).view(1, 2)
        output = dec(z)
        image = output.view(28, 28).detach().numpy()
        row_start = j*28
        row_end = (j+1)*28
        col_start = i*28
        col_end = (i+1)*28

latent_image[row_start:row_end, col_start:col_end] =
image
plt.imshow(latent_image, cmap=plt.get_cmap('gray'))
plt.show()
```

2.1 Systematically sample an Autoencoder

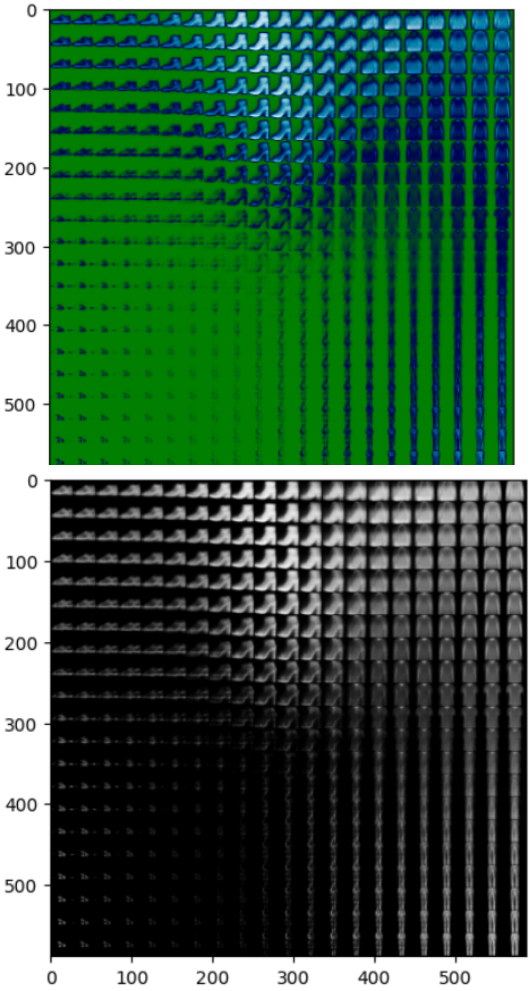

Figure 2: Autoencoder Latent Images

The top rows(0-200) display a stark transition from very dark to very bright images as one moves horizontally. This suggests that the model's latent variables are highly sensitive to changes in this region of the latent space. The bright regions might represent common shapes of items, while the darker regions might represent less typical or more ambiguous forms.

2.2 Compare the latent spaces of the VAE and autoencoder

Fig 1 shows that the VAE learns the representation of object such as shoes more than other objects.
Fig 2, the Autoencoder learns the representation of objects such as shoes, boots, and jackets etc. better than the VAE. It has a more distinguishable combination of the objects such as shirt, boots etc.
This can be because AEs are optimized to minimize the reconstruction error directly. They tend to learn a deterministic and potentially overfitting mapping from inputs to the latent space. This could be leading to crisper, more detailed outputs where distinctions among different object types like shirts and boots are more clearly preserved.