DL Lab 7 – Hettiarachchige H.I.A Y4.S1.WE.SE.2

Github: [IT21305900/DL-Lab7 (github.com)]

2. A linear AutoEncoder (AE) without activation functions is closely related to Principal Component Analysis (PCA). Both techniques aim to reduce the dimensionality of data using linear transformations. In PCA, the goal is to find directions (called principal components) that capture the most variance in the data, while a linear AE learns a lower-dimensional representation by minimizing the reconstruction error of the data. The weights learned by the encoder in a linear AE span the same subspace as the principal components found by PCA, meaning they both find similar low-dimensional representations. The main difference lies in how they achieve this: PCA uses a mathematical technique called Singular Value Decomposition (SVD), while linear AEs use gradient descent to learn the optimal weights. Despite the different methods, both achieve the same objective of finding the most important features in the data.

3. In comparison increase of no of epochs against the loss model trained with the 2D CNN layer AE has shown higher accuracy than the model with dense layer based AE.

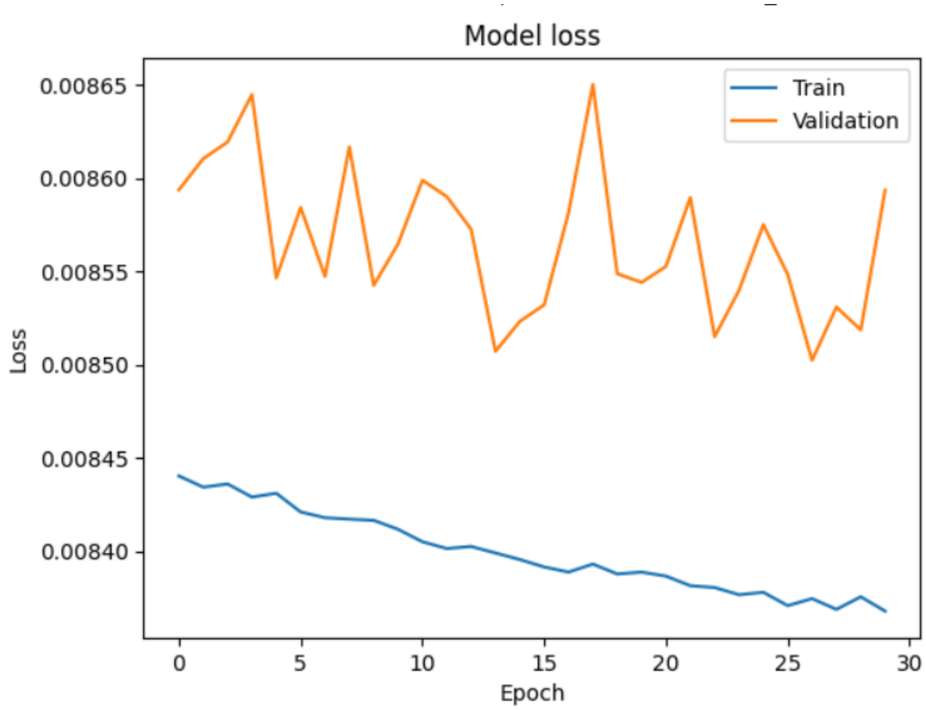

Figure 1 Model with Dense Layer AE

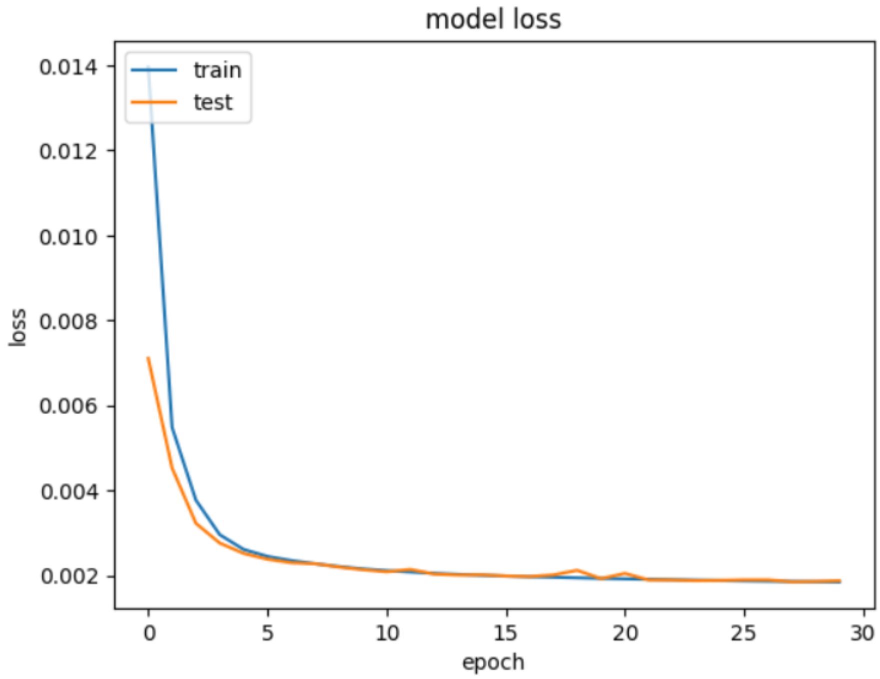

Figure 2 Model with 2D CNN AE

1. Feature Extraction:

   Dense Layer AE: In a Dense Layer AE, each neuron is fully connected to the neurons in the adjacent layers, which makes the model learn global features from the input. This approach may be less effective in capturing local patterns, especially in image data like Fashion MNIST, where spatial relationships are critical.

   2D CNN AE: In contrast, the 2D CNN AE uses convolutional layers, which are designed to capture local spatial features through the application of filters (kernels). CNNs are highly effective in processing image data because they preserve spatial hierarchies by learning local features at various levels of granularity. As a result, the CNN AE typically learns more efficient feature representations for images, leading to better performance.

2. Weight Sharing and Parameter Efficiency:

   Dense Layer AE: Dense layers do not have weight sharing, meaning each connection has a separate weight. This results in a large number of parameters, especially with high-dimensional input like images, making it harder to train the model effectively.

   2D CNN AE: CNN layers share weights through the convolutional filters, which leads to fewer parameters and more efficient learning. This makes CNNs better suited for image-based tasks, allowing the model to generalize better and achieve higher accuracy.

3. Spatial Information Preservation:

Dense Layer AE: By flattening the input image to pass it through the fully connected layers, the dense layer AE loses the spatial arrangement of the pixels. This loss of spatial information is a significant drawback in tasks like image reconstruction, where maintaining the structural integrity of the image is crucial.

2D CNN AE: CNNs preserve spatial structure through convolutional layers and pooling, ensuring that important spatial relationships between pixels are maintained. This preservation of structure is key in improving the quality of image reconstruction, which explains why CNN AEs generally outperform dense-layer AEs in such tasks.

4. Performance Metrics:

   Dense Layer AE: The performance of the Dense Layer AE may plateau or improve only marginally as the number of epochs increases, primarily because of its limited ability to capture the complex structure of image data.

   2D CNN AE: The 2D CNN AE tends to achieve higher accuracy with the same number of epochs due to its enhanced feature extraction capabilities and efficient learning mechanisms. As seen in the comparison, the CNN AE exhibits lower reconstruction error and improved image fidelity.

6 In general comparison Image De-noising AE outperforms Vanilla CNN AE in training with low loss value. However, one thing to notice in the De-noising AE is with the change of the noice the model validation loss change in noticeable way expressing an scenario of model overfitting situation. Compared to this situation Vanilla CNN AE in perform well in both validation & training stages.

Figure 3 Noice with 0.6 De Noice AE

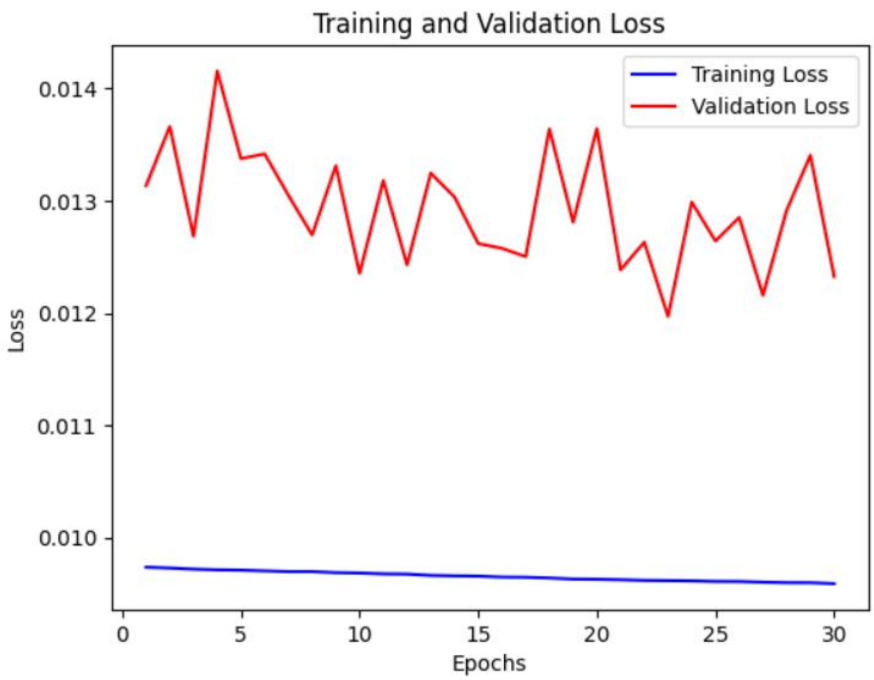

Figure 4 Noice with 0.3 Denoice AE

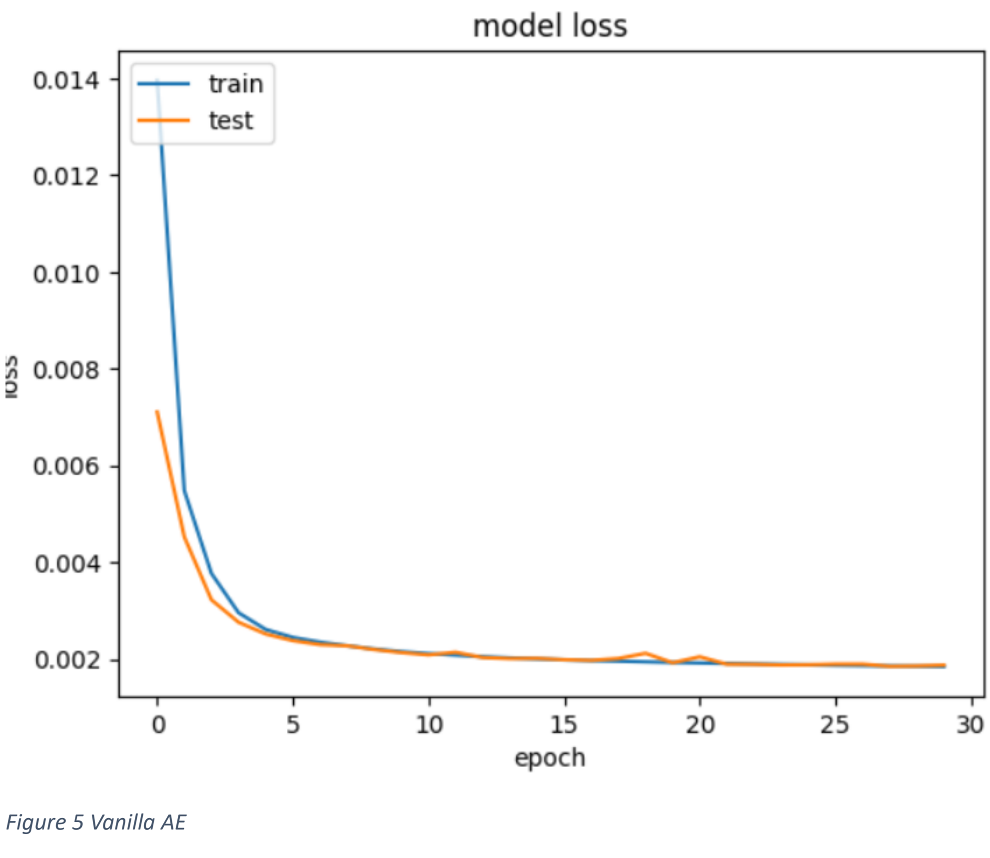

Figure 5 Vanilla AE

7. An Autoencoder (AE) and a Variational Autoencoder (VAE) are both neural networks designed for dimensionality reduction and data reconstruction.

Autoencoder (AE): A regular AE works by compressing input data into a lower-dimensional compressed version of the data), then reconstructing the original input from that compressed version. It learns to find important patterns in the data, but it doesn't enforce any specific structure on the latent space, meaning the compressed representations might not follow any particular distribution.

Variational Autoencoder (VAE): A VAE is a more advanced type of AE that adds a probabilistic twist. Instead of just learning a single compressed version of the input, a VAE learns a distribution (usually a Gaussian distribution) for each data point in the latent space. This allows it to generate new, similar data points by sampling from this distribution. Essentially, VAEs are better for tasks like generating new images or creating smooth transitions between different data points, because they ensure that the latent space has a well-organized structure.

In simple terms, an AE is like a regular compressor that shrinks data to a smaller version and restores it. A VAE, on the other hand, not only compresses but also learns to represent the data in a more flexible and structured way, making it better for generating new, similar data from what it has learned.