DL lab 7 -Autoencoders

1. Upload the Autoencoder (AE) jupyter notebook file (i.e., lab\_7\_AE\_FFNN.ipynb) to google colab root directory.
   * In this code, an image reconstruction is done using dense layers-based AE.
   * Fashion MNIST dataset is used for this task (also for the subsequent tasks as well).
   * Run the above code and understand it.
   * Train the model with 30 epochs.
   * Write the code implementation to calculate the loss (Mean Squared Error) for the test dataset.
   * Write the code implementation to plot the train and validation loss against number of epochs.
2. When above AE is used without activation functions, it is called a linear AE. Explain the relationship between linear AE and principal component analysis (PCA). Write the answer in a word file.

A linear Autoencoder (without activation functions) is equivalent to PCA in that both perform linear dimensionality reduction. They both aim to compress the data into a lower-dimensional representation while preserving key information, and both minimize reconstruction errors in the process. The main difference lies in how the two are implemented: PCA uses an algebraic approach, while linear Autoencoders rely on neural network training.

1. Upload the Vanilla CNN AE jupyter notebook file (i.e., lab\_7\_AE\_Vanilla\_CNN.ipynb) to google colab root directory.
   * In this code, instead of dense layers, 2D CNN layers are used.
   * Task in the same as before with the same Fashion MNIST dataset.
   * Run the above code and understand it.
   * Train the model with 30 epochs.
   * Write the code implementation to calculate the loss (Mean Squared Error) for the test dataset.
   * Write the code implementation to plot the train and validation loss against number of epochs.
2. Observe the model performance improvements between the above two models and give reasons for the observed improvements.

The Vanilla CNN Autoencoder outperforms the Dense Autoencoder because,

* Better Reconstruction - CNNs capture spatial features like edges and textures, leading to more accurate image reconstruction.
* Fewer Parameters - CNNs use parameter sharing with filters, making them more efficient than Dense layers, which have more parameters.
* Spatial Information - CNNs preserve the spatial relationships between pixels, while Dense layers treat the image as a flat vector, losing this information.
* Better Generalization - CNNs are less prone to overfitting and generalize better on test data, resulting in improved performance.
* Efficient Training - CNNs learn local patterns faster and more efficiently, while Dense layers are slower and less effective for image data.

1. Upload the Image De-noising AE jupyter notebook file (i.e., lab\_7\_AE\_CNN\_Image\_Denoising.ipynb) to google colab root directory.
   * In this code, noise is first added to the images before the reconstruction.
   * This is a method to overcome the overfitting that happens in AEs.
   * Run the above code and understand it.
   * Train the model with 30 epochs.
   * Write the code implementation to calculate the loss (Mean Squared Error) for the test dataset.
   * Write the code implementation to plot the train and validation loss against number of epochs.
   * Experiment with “noise\_factor” value and use the best value you find in the final implementation. (Pay attention to how this value affect the images by observing the noise added images in the code.

When reduce the value of noise\_factor images are getting clear.

1. Observe the model performance improvements between the Image De-noising AE and the Vanilla CNN AE.

The Image De-noising Autoencoder improves performance compared to the Vanilla CNN Autoencoder,

* Noise Robustness: The Image De-noising AE adds noise during training, helping the model learn more robust features and improving its ability to generalize to unseen data.
* Overfitting Prevention: Adding noise serves as a form of regularization, reducing overfitting and making the model more resilient to variations in the data.
* Better Reconstruction: The denoising process helps the model focus on essential image features, leading to cleaner and more accurate reconstructions.
* Handling Noisy Data: Image De-noising AE performs better in real-world scenarios where input data might be noisy, unlike the Vanilla CNN AE, which may struggle without noise resistance training.

1. Explain the differences between AE and Variational AE (VAE).

**Latent Space Representation**

* AE: Encodes inputs into a fixed set of points in the latent space, directly reconstructing them.
* VAE: Encodes inputs into a distribution (mean and variance), sampling from this distribution to generate more diverse and flexible representations.

**Generative Capabilities:**

* AE: Not inherently generative; focuses on reconstructing input data.
* VAE: A generative model that can create new data by sampling from the learned latent space distribution.

**Loss Function:**

* AE: Uses only reconstruction loss (e.g., Mean Squared Error) to measure how well the input is reconstructed.
* VAE: Combines reconstruction loss with a KL Divergence term, which ensures the latent space follows a Gaussian distribution for better sampling and generation