VQ-VAE As Defense On Adversarial Attacks

Harsh Gupta
20961
MTech, CSA

Abstract

The objective of this this project is to investigate the effectiveness of various adversarial attacks and defense mechanisms on a pre-trained image classification model. For this project, I have chosen to focus on adversarial attacks using the **Fast Gradient Sign Method (FGSM)**, **Iterative FGSM (IFGSM)**, and **Momentum Iterative FGSM (MIFGSM)** algorithms. Specifically, I aim to analyze the impact of these attacks on the model's accuracy and robustness against different defense mechanisms, including image compression using **KMeans** and **Vector Quantized Variational AutoEncoder (VQVAE)** with different levels of noise (**0**, **0.25**, and **0.75**). Our experiments demonstrate that VQVAE can effectively mitigate adversarial attacks without requiring retraining of the classifier model. We evaluate the defense mechanism's performance by measuring the model's accuracy and robustness against the attacks. Our results suggest that VQVAE is a promising defense mechanism against adversarial attacks on image classification systems.

1 Introduction

The broad project topic is **Adversarial Attacks and Defenses in Deep Learning**, with a focus on **Implementing Efficient Defenses Against Adversarial Attacks on Pre-Trained Image Classification Models**

Deep neural networks are vulnerable to **adversarial attacks**, which are designed to mislead the model's predictions by adding imperceptible perturbations to the input data. Adversarial attacks can pose significant threats to the security and reliability of deep learning models in real-world applications. Thus, improving the adversarial robustness of deep neural networks has become a crucial research direction in recent years. One approach to enhance the robustness of models against adversarial attacks is to train them on a combination of natural and adversarial examples, a technique called **adversarial training**. However, adversarial training is computationally intensive and challenging to scale, making it less practical for real-world applications.

To address this issue, researchers have proposed various efficient **adversarial attack and defense mechanisms** that can improve the model's robustness without incurring significant computational costs. In this project, we focus on studying the impact of adversarial attacks, specifically **Fast Gradient Sign Method (FGSM)**, **Iterative FGSM (IFGSM)**, and **Momentum Iterative FGSM (MIFGSM)**, on the accuracy and robustness of a pre-trained image classification model. We also investigate the effectiveness of different defense mechanisms, including **KMeans image compression** and **Vector Quantized Variational AutoEncoder (VQVAE)** with different levels of noise (**0**, **0.25**, and **0.75**) in protecting the model against these attacks.

This project aims to contribute to the broader objective of improving the adversarial robustness of deep neural networks for secure and reliable deployment in real-world applications.

The code can be found at:

https://github.com/Harsh-Gupta9897/VQVAE_as_Defense_Against_Adverserial_Attack

2 Methodologies

2.1 Attacks

2.2 Adversarial Attacks

a) **FGSM**: Fast Gradient Sign Method (FGSM) is a popular white-box attack that perturbs the input image by adding noise proportional to the sign of the gradient of the loss function with respect to the input. The perturbation is scaled by a hyperparameter ϵ which controls the size of the noise. The formula for generating adversarial examples using FGSM is:

$$x_{adv} = x + \epsilon sign(\nabla_x L(x, y, \theta))$$

b) **IFGSM**: Iterative Fast Gradient Sign Method (IFGSM) is an iterative variant of FGSM, which adds small perturbations to the original input image for multiple iterations. The perturbations are computed using the same formula as FGSM, but applied iteratively for a certain number of iterations.

c) **MIFGSM**: Momentum Iterative Fast Gradient Sign Method (MIFGSM) is a variant of IFGSM which adds momentum to the iterative perturbations to make them more consistent across iterations. The addition of momentum allows the perturbations to follow a more direct path towards the decision boundary of the model, resulting in more effective adversarial examples.

2.3 Defense

a) **Image Compression using K-means**: One way to defend against adversarial attacks is to use data pre-processing techniques like image compression. In this project, we explore the use of K-means clustering as a compression technique. K-means clustering reduces the dimensionality of the image by grouping similar pixels together and replacing them with the cluster centroid. This compression makes the image less susceptible to adversarial perturbations, as the small perturbations applied to the input image will have less effect on the overall image after compression.

b) **Vector Quantized Variational Autoencoder (VQ-VAE)**: Another way to defend against adversarial attacks is to use generative models like VQ-VAE. VQ-VAE is a type of autoencoder that learns to compress and reconstruct images. The encoder network maps the input image to a low-dimensional latent code, and the decoder network maps the latent code back to a reconstructed image. The novelty of our approach is that we propose using VQ-VAE with different values of noise added to input, i.e., $noise\_added = 0, 0.25, 0.75$, to achieve better defense against adversarial attacks.

We have evaluated the performance of our approach by generating adversarial attacks using FGSM, IFGSM, and MIFGSM methods on the MNIST datasets, and then measuring the accuracy of the VQ-VAE model with varying noise added and tunning hyperparameters in correctly classifying the attacked images.

VQVAE is typically trained to reproduce the same image from an input image. As for adversarial defense, reproducing the same adversarial images is an undesirable task as the adversarial perturbations may be preserved during the image reconstruction. Instead, in Defense-VQVAE, we modify the encoder and the decoder of the latent variable model(inspired from Defense-VAE) as follows:

$$z \sim Encoder(\hat{x}) = q(z|\hat{x}), \quad x \sim Decoder(\hat{z}) = p(x|\hat{z})$$

Here, $\hat{z}$ is the output of the vector quantizer which quantizes the latent code z obtained from the encoder network for the perturbed image $\hat{x}$. The outputs $\hat{x}$ is obtained by adding noise to input image.

2.4 Pipeline

- **Input Image**: The pipeline takes an input image from the MNIST dataset as input.

- **Adversarial Attacks**: The input image is subjected to adversarial attacks using FGSM, I-FGSM, and MI-FGSM with different values of epsilon. These attacks generate perturbed images that are used to evaluate the defense mechanisms.

- **Defense Mechanisms**: The perturbed images are then passed through defense mechanisms, K-means clustering and VQVAE, which attempt to reconstruct the original image while removing the adversarial perturbations.

- **Pretrained Model**: Finally, the reconstructed images are passed through a pre-trained MNIST model to evaluate the effectiveness of the defense mechanisms.

The pipeline is illustrated in the following figure:

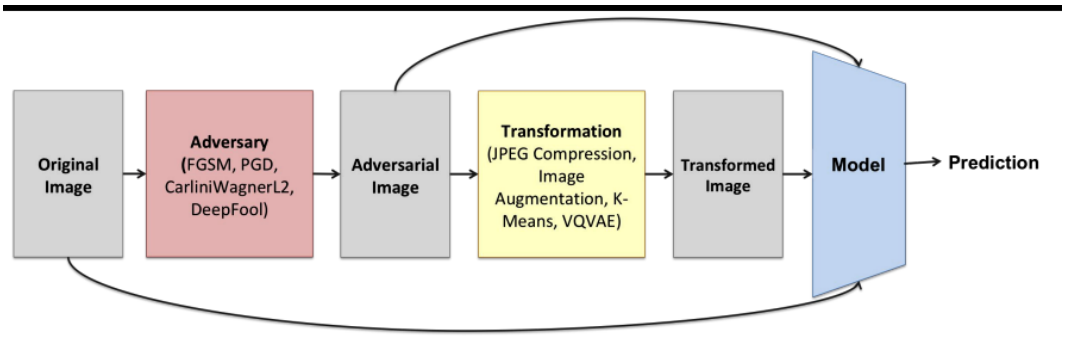

Figure 1: The pipeline for evaluating defense mechanisms against adversarial attacks.

The pipeline takes an input image from the MNIST dataset and generates adversarial examples using FGSM, I-FGSM, and MI-FGSM attacks. The adversarial examples are then passed through two different defense mechanisms, K-means clustering and VQVAE, to reconstruct the original image. Finally, the reconstructed image is passed through a pre-trained MNIST model to evaluate the effectiveness of the defense mechanisms against the adversarial attacks.

3 Experiments

In this section, we evaluate the performance of our defense mechanism against various attacks on a pre-trained model on the MNIST dataset. We apply three different types of attacks: Fast Gradient Sign Method (FGSM), Iterative Fast Gradient Sign Method (IFGSM), and Momentum Iterative Fast Gradient Sign Method (MIFGSM), with varying epsilon values ranging from 0 to 0.3.

For each attack, we apply our defense mechanism using either K-means clustering or VQVAE to reconstruct the perturbed image. We then pass the reconstructed image through the pre-trained MNIST model and calculate the accuracy of the model on the reconstructed image.

3.1 Attack using FGSM, IFGSM, and MIFGSM

We perform attacks using FGSM, IFGSM, and MIFGSM with varying epsilon values on the pre-trained model. The resulting images after the attacks are shown in Figure 2. We can observe that the attacks cause distortions in the images, making them difficult for humans to classify correctly.For these attacks, there is not retraining of model required.

Fig1: IFGSM Attack

Fig2: IFGSM Attack

Fig3: IMFGSM Attack

Figure 2: Images after attacks using FGSM, IFGSM, and MIFGSM with varying epsilon values

3.1.1 Accuracy plot with Attacks Only

We plot the accuracies of the model against the epsilon values used in the attacks in Figure 3. We can observe that the accuracy of the model decreases as the epsilon values increase, indicating that the attacks are successful in fooling the model.

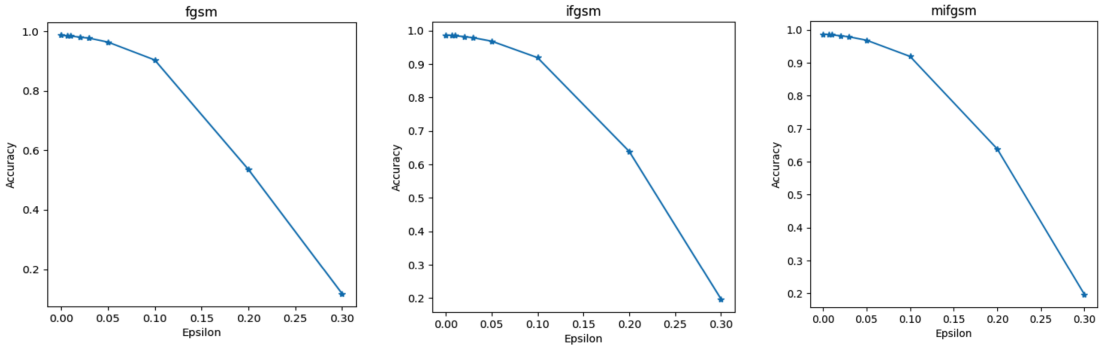

Figure 3: Accuracy plot against the epsilon values used in the attacks

3.1.2 Accuracy Plot with Attacks and VQVAE as Defense with 0.25 Noise

We performed three different types of attacks, FGSM, IFGSM, and MIFGSM, with varying epsilon values ranging from 0 to 0.3 on a pre-trained MNIST model. We then applied VQVAE as a defense mechanism with a 0.25 noise added to the reconstructed images.

The accuracies of the model against the epsilon values used in the attacks were plotted and are shown in Figure 4. It can be observed that **the accuracy with defense VQVAE improved by 20-30%** compared to the model without any defense mechanism.

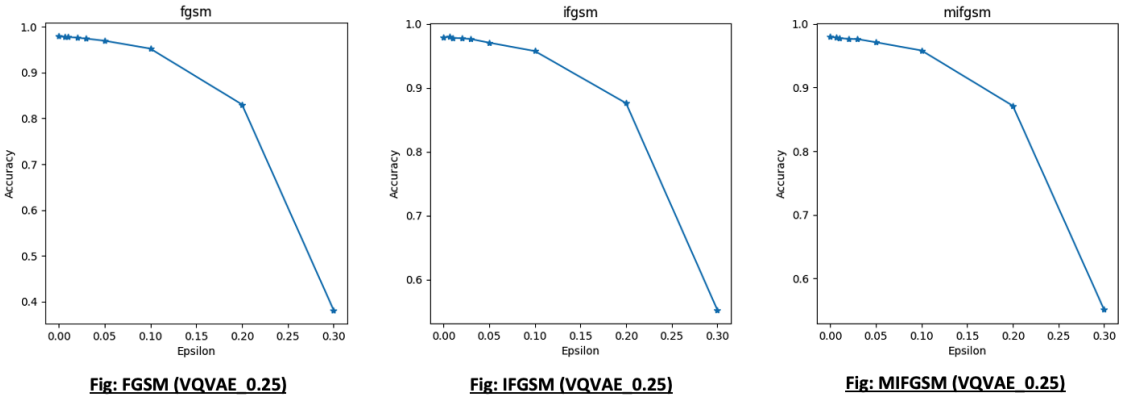

Figure 4: Accuracy plot against the epsilon values used in the attacks with defense VQVAE_0.25

Furthermore, the VQVAE defense mechanism has an advantage of maintaining the quality of the image, even for non-perturbed images. This is because VQVAE is designed to reconstruct images better than traditional VAE and autoencoder methods. Thus, it provides an effective defense mechanism for improving the robustness of pre-trained models against adversarial attacks.

3.1.3 Accuracy Plot with Attacks and KMeans as Defense

We also experimented with KMeans as a defense mechanism against adversarial attacks. We used KMeans clustering with k=20 to cluster the images and then assigned each pixel to its nearest cluster

center. We then reconstructed the image by assigning the value of each pixel to the center of its assigned cluster.

However, due to the time-consuming nature of KMeans clustering, we only performed experiments on the FGSM attack with epsilon values of 0.2 and 0.3. The accuracy results for the KMeans defense are shown in Figure 5. We can observe that the KMeans defense does not perform as well as the VQVAE defense with a noise level of 0.25.

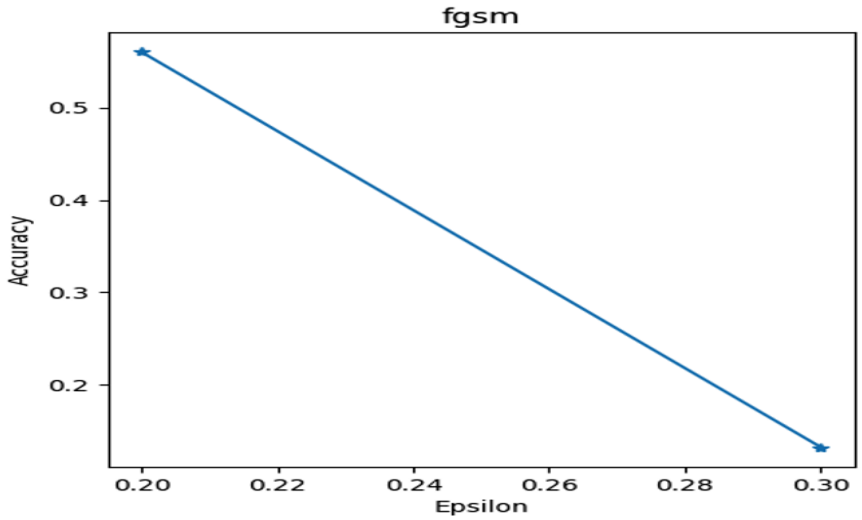

Figure 5: Accuracy plot against the epsilon values used in the FGSM attack with defense KMeans(k=20)

We would like to note that the KMeans defense has the advantage of not reducing the accuracy of non-perturbed images, but it is not as effective as the VQVAE defense in terms of maintaining the quality of the reconstructed image. It should also be noted that our experiments were performed using only 20 iterations(because of very time consuming process) of KMeans clustering, which may not have been sufficient for optimal performance.The improvement is also very less, but it performs well, if we train it for more iterations.

Also the perturbed Image generated after reconstruction using KMeans as defense is as follows:

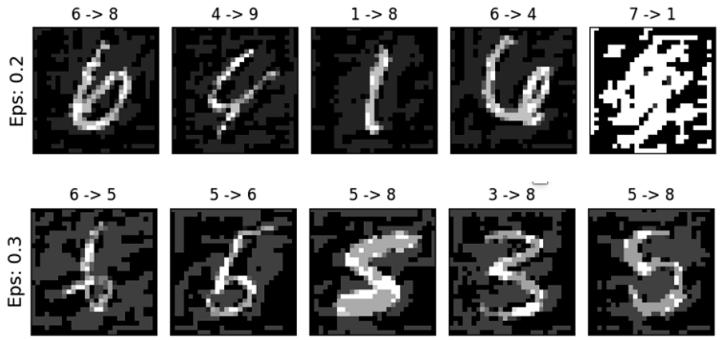

Figure 6: Image Reconstructed using KMeans k=20 after Attack

The above image shows, that kmeans requires more number of iterations to produce result, which is very time consuming and also , we can better results using VQVAE which takes lesser time to train on given dataset, as it is trained earlier to reproduce images.

3.2 VQVAE Image Reconstruction

In this section, we show the reconstruction of images using VQVAE after adding noise to the original image. We use the VQVAE model trained on the MNIST dataset and add noise to the input images. We then feed the noisy images to the VQVAE model for reconstruction.

Figure 7 shows some sample input images from the MNIST dataset, and Figure 8 shows the corresponding reconstructed images after adding noise. We can observe that the VQVAE model is able to reconstruct the images with a high level of accuracy, even after adding noise to the input images.

Figure 7: Sample input images from the MNIST dataset

Figure 8: Corresponding reconstructed images using VQVAE after adding noise after training

These reconstructed images can be used as a defense mechanism against adversarial attacks, as the VQVAE model is able to reconstruct the images accurately even after adding noise. This ensures that the model is able to make accurate predictions even in the presence of adversarial examples.

Now , The reconstructed image after applying defense VQVAe_0.75 after attack by FGSM with epsilon 0.2 and 0.3 is as follows:

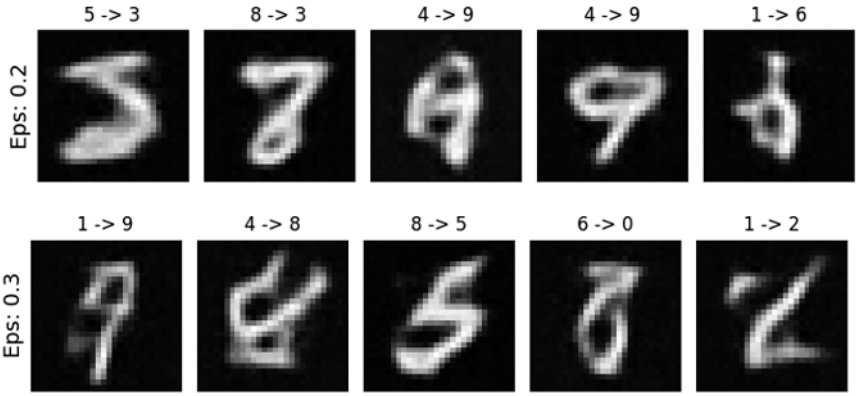

Figure 9: Reconstructed images after attack FGSM and then VQVAE_0.75 as defense

From,above image, we can see the difference between the images generated by KMeans and VQ-VAE_0.75 ,where VQVAE_0.75 is trained for just 6 epochs, if we train VQVAE for more iterations , it will give much better results.But it takes very less time as compared to KMeans to learn.In this , we don't need to perform anything on pretrained model of MNIST, which also takes more time than VQVAE to classify the problem with good accuracy.

Hence, VQVAE produces the promising results, by raising the accuracy by 30-40% by training just for 6 epochs.

3.3 Comparison between different VQVAE Models as Defense

This is the comparison of FGSM with different epsilon value, highlighted text shows the best result.It performs well and give promising results.

Attack\Defense (%)	No Defense	VQVAE_0	VQVAE_0.25	VQVAE_0.5	VQVAE_0.75
No Attack	98.39	**98.39**	97.91	96.39	91.94
FGSM (0.001)	98.23	**98.39**	97.99	96.25	92.08
FGSM (0.005)	96.94	**98.34**	97.79	96.05	91.45
FGSM (0.2)	54.20	82.2	**82.39**	75.84	64.92
FGSM (0.3)	8.92	26.92	35.49	**41.71**	38.53

Figure 10: Table fGSM attack with VQVAE defense

This is the comparison of I-FGSM with different epsilon value, highlighted text shows the best result.

Attack\Defense	No Defense	VQVAE_0	VQVAE_0.25	VQVAE_0.5	VQVAE_0.75
No Attack	98.7	**98.46**	98.03	96.12	92.06
IFGSM (0.001)	98.13	**98.43**	97.97	96.27	92.18
IFGSM (0.005)	98.01	**98.38**	97.87	96.25	91.46
IFGSM (0.2)	63.41	86.28	**86.87**	80.33	69.33
IFGSM (0.3)	18.57	46.84	**52.32**	**52.43**	46.44

Figure 11: Table I-FGSM attack with VQVAE defense

This is the comparison of MI-FGSM with different epsilon value, highlighted text shows the best results in Figure 12.

Attack\Defense	No Defense	VQVAE_0	VQVAE_0.25	VQVAE_0.5	VQVAE_0.75
No Attack	98.39	98.37	98.00	96.32	91.92
MIFGSM (0.001)	98.39	**98.39**	97.88	96.5	91.98
MIFGSM (0.005)	97.69	**98.39**	97.95	96.2	91.47
MIFGSM (0.2)	82.53	86.20	**86.89**	80.81	69.05
MIFGSM (0.3)	34.02	46.43	**53.4**	53.43	46.59

Figure 12: Table MI-FGSM attack with VQVAE defense

3.3.1 Conclusion

In all three attacks, if we increase the value of noise , the accuracies improve for higher perturbation input by 1-3% but the accuracies of no perturbation input class also decreases, so it is kind of tradeoff, so we will chose the model according to our requirement.

If we increase the perturbation, then there is tradeoff between accuracies and robustness. But as perturbation increases , we can see the changes are visible, so it will not cause problem in the attacks we have discussed. **The best result is given by VQVAE with noise=0.25 for all attacks FGSM,I-FGSM, MI-FGSM attack on MNIST dataset**.

4 Code References

The following code repositories and packages were used in this project:

- PyTorch: an open source machine learning library used for building and training neural networks.
- FGSM, I-FGSM, MI-FGSM: a Python implementation of the Fast Gradient Sign Method adversarial attack(https://github.com/as791/Adversarial-Example-Attack-and-Defense)
- K-Means : a Python implementation of the K-Means algorithm for image compression.
- VQ-VAE : a PyTorch implementation of the Vector Quantized Variational AutoEncoder model
- MNIST dataset : a dataset of handwritten digits commonly used for machine learning experiments.

The code for the VQ-VAE model and the K-Means algorithm was modified to incorporate our defense mechanism. The modified code is available in the above mentioned link for project.

References

[1] Goodfellow, I.J., Shlens, J., Szegedy, C. (2014). Explaining and harnessing adversarial examples. In *International Conference on Learning Representations* (ICLR). Retrieved from `https://arxiv.org/abs/1412.6572`

[2] Madry, A., Makelov, A., Schmidt, L., Tsipras, D., Vladu, A. (2018). Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations* (ICLR). Retrieved from `https://arxiv.org/abs/1706.06083`

[3] Learning Adversarerially Robust and Rich Image Transformations for Object Detection`https://cs229.stanford.edu/proj2019aut/data/assignment_308875_raw/26505582.pdf`

[4] Xiang Li, Shihao Ji(2019),Defense-VAE: A Fast and Accurate Defense against Adversarial Attacks `https://arxiv.org/pdf/1812.06570.pdf`

[5] Abdul Jabbar, Xi Li, Bourahla Omar,A Survey on Generative Adversarial Networks: Variants, Applications, and Training
`https://arxiv.org/pdf/2006.05132.pdf`

[6] Anirban Chakraborty, Manaar Alam, Vishal Dey, Anupam Chattopadhyay, Debdeep Mukhopadhyay,A survey on adversarial attacks and defences `https://ietresearch.onlinelibrary.wiley.com/doi/10.1049/cit2.12028`