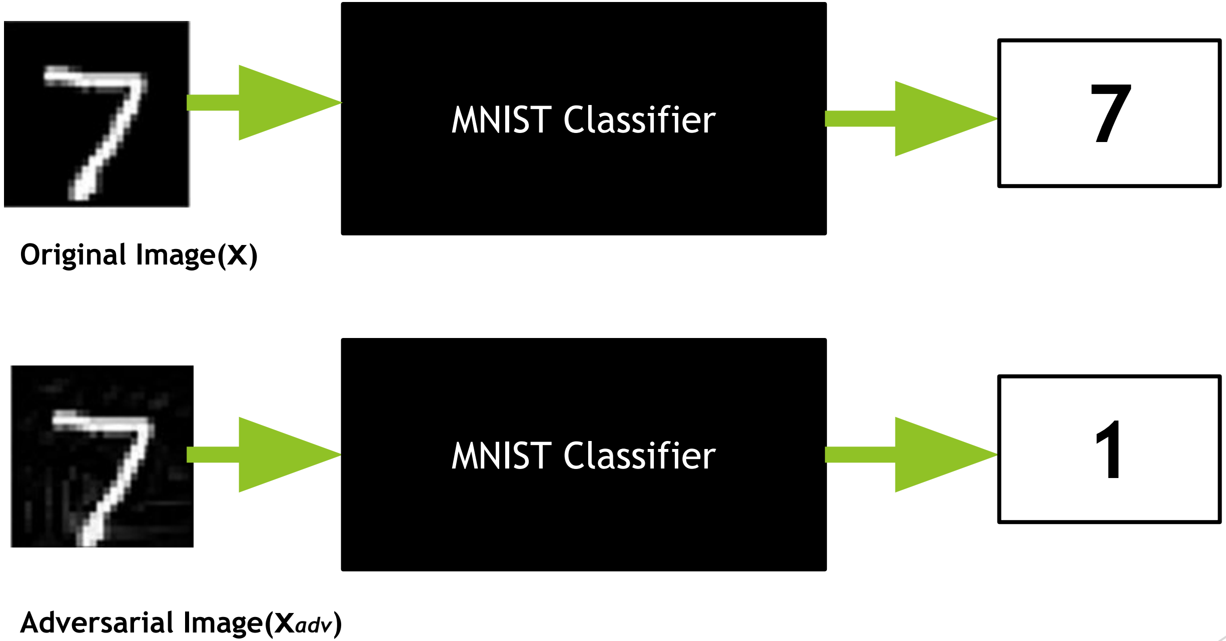
Variational Inference with Latent Space Quantization for Adversarial Resilience

Vinay Kyatham, Deepak Mishra, Prathosh A.P

What is an Adversarial Example ?



Original Image(X)

MNIST Classifier

7

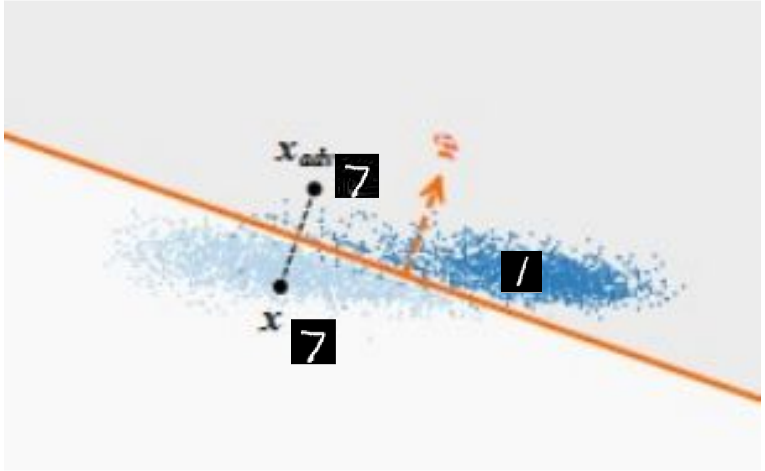
Adversarial Image(X_{adv})

MNIST Classifier

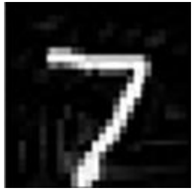
1

What is an Adversarial Example ?

- Below Figure shows the classification boundary, which separates classes 7 and 1.
- Adversarial noise is added to real sample whose label is 7,
  in such a way that it falls into other class region, making the classifier to misclassify it as 1

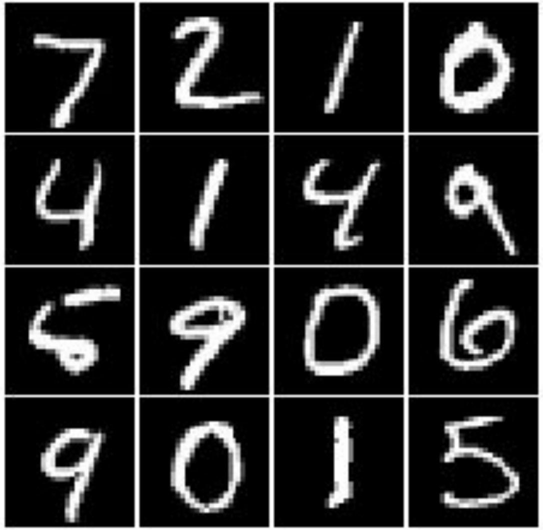


Original Image(X)

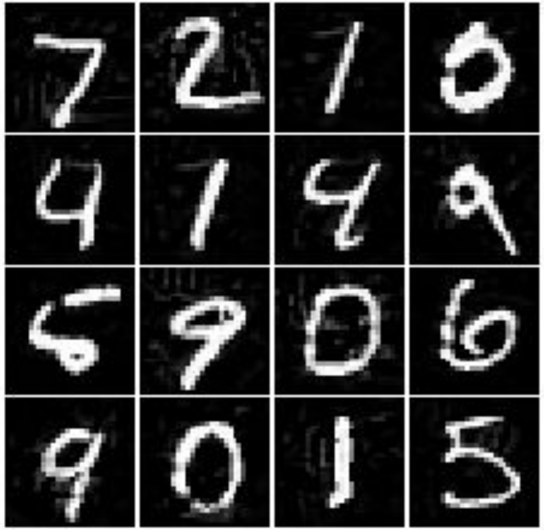

Adversarial Image(X_{adv})

MNIST: CW (Carlini-Wagner)

- Left block shows the real mnist images, whose classification accuracy is 100%
- Right block images are adversarial images generated by using the CW (Carlini-Wagner) method, classification accuracy has dropped to 12%



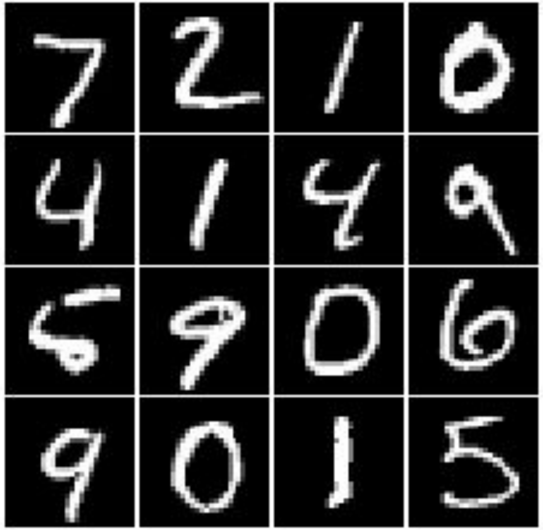
100%



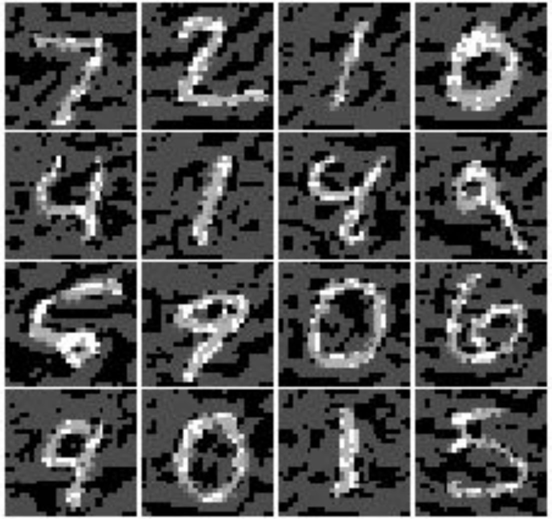
12%

MNIST: FGSM (Fast gradient sign method)

- Left block shows the real mnist images, whose classification accuracy is 100% and
- Right block images are adversarial images generated by using the FGSM method, classification accuracy has dropped to 6%



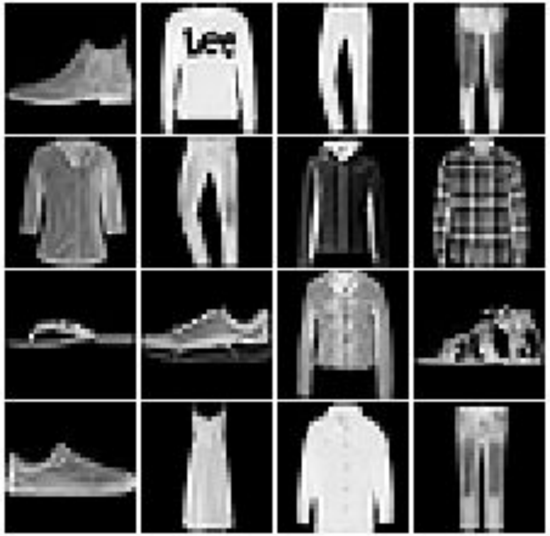
100%



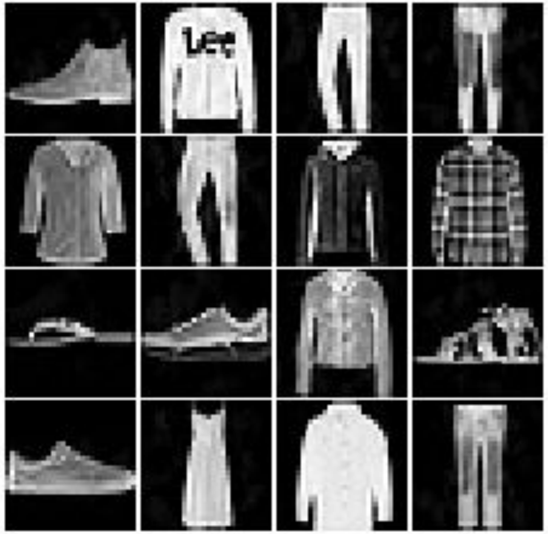
6%

FMNIST: CW (Carlini-Wagner)

- Left block shows the real fmnist images, whose classification accuracy is 100%
- Right block images are adversarial images generated by using the CW (Carlini-Wagner) method, classification accuracy has dropped to 20%



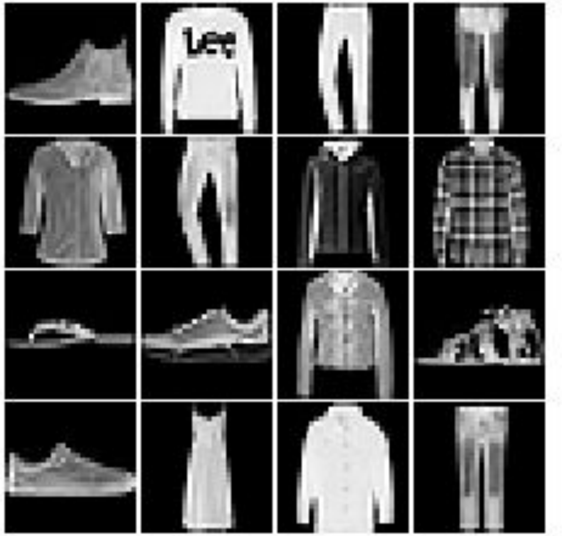
100%



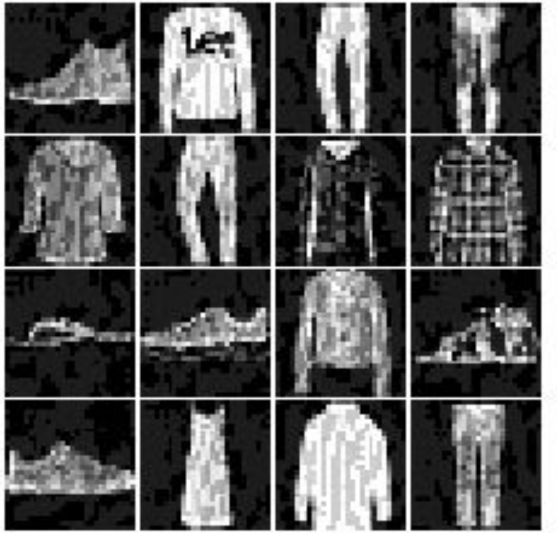
20%

FMNIST: FGSM (Fast gradient sign method)

- Left block shows the real fmnist images, whose classification accuracy is 100%
- Right block images are adversarial images generated by using the CW (Carlini-Wagner) method, classification accuracy has dropped to 12%



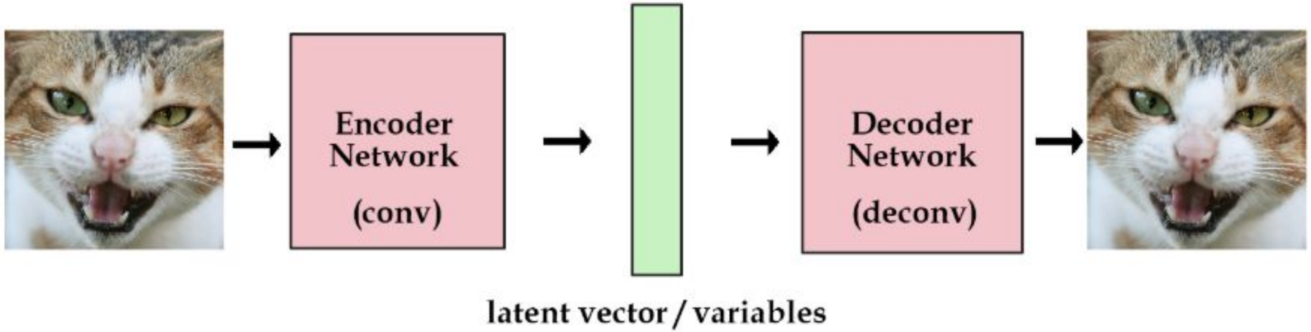
100%



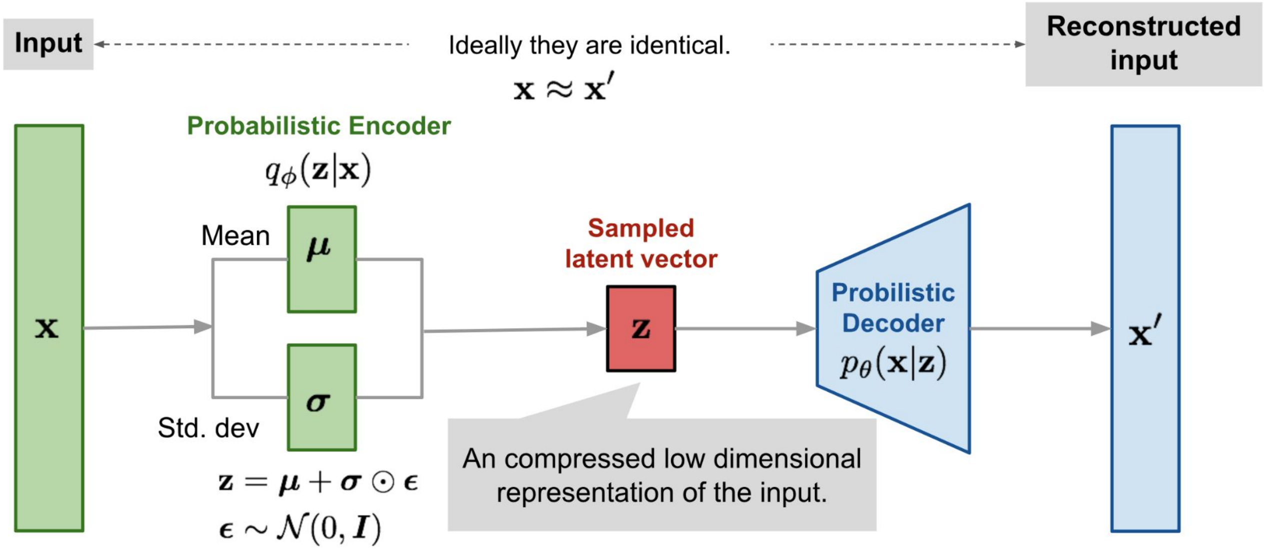
12%

VAE (Variational AutoEncoder)

- VAE is a Generative Model
- It has an Encoder and decoder
- Encoder encodes the given image and produces a latent vector
- Decoder decodes the latent vector and produces an image.
- Latent space is forced to follow normal distribution
- Any Randomly sampled latent vector from normal distribution,
  when decode using decoder produces an image from training data distribution

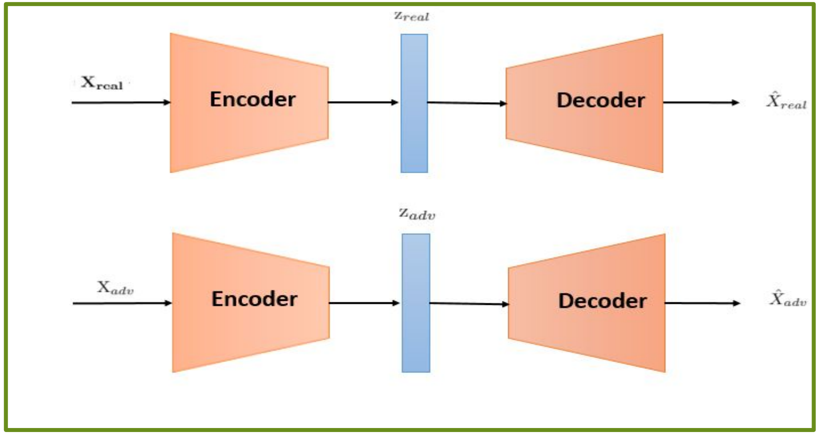


VAE (Variational AutoEncoder)


Input
Ideally they are identical.
$\mathbf{x} \approx \mathbf{x}'$
Reconstructed input
$\mathbf{x}$
Probabilistic Encoder
$q_\phi(\mathbf{z}|\mathbf{x})$
Mean
$\boldsymbol{\mu}$
$\boldsymbol{\sigma}$
Std. dev
$\mathbf{z} = \boldsymbol{\mu} + \boldsymbol{\sigma} \odot \boldsymbol{\epsilon}$
$\boldsymbol{\epsilon} \sim \mathcal{N}(0, \boldsymbol{I})$
Sampled latent vector
$\mathbf{z}$
An compressed low dimensional representation of the input.
Probilistic Decoder
$p_\theta(\mathbf{x}|\mathbf{z})$
$\mathbf{x}'$

Lipschitz constrained latent space

- Real image and its Adversarial image are very close in the image space.
- Lipschitz constrained latent encoder makes sures that real image and its adversarial image are very close in the latent space.
- Gradient norm penalty is used to enforce Lipschitz constrain



Real image and its Adversarial image are very close in the image space.

$$X_{adv} = \mathbf{X}_{\mathbf{real}} + \eta$$

Lipschitz constrain

$$\|\mathbf{z}_{\mathbf{real}} - \mathbf{z}_{\mathbf{adv}}\|_2 \leq K\|\mathbf{X}_{\mathbf{real}} - \mathbf{X}_{\mathbf{adv}}\|_2$$

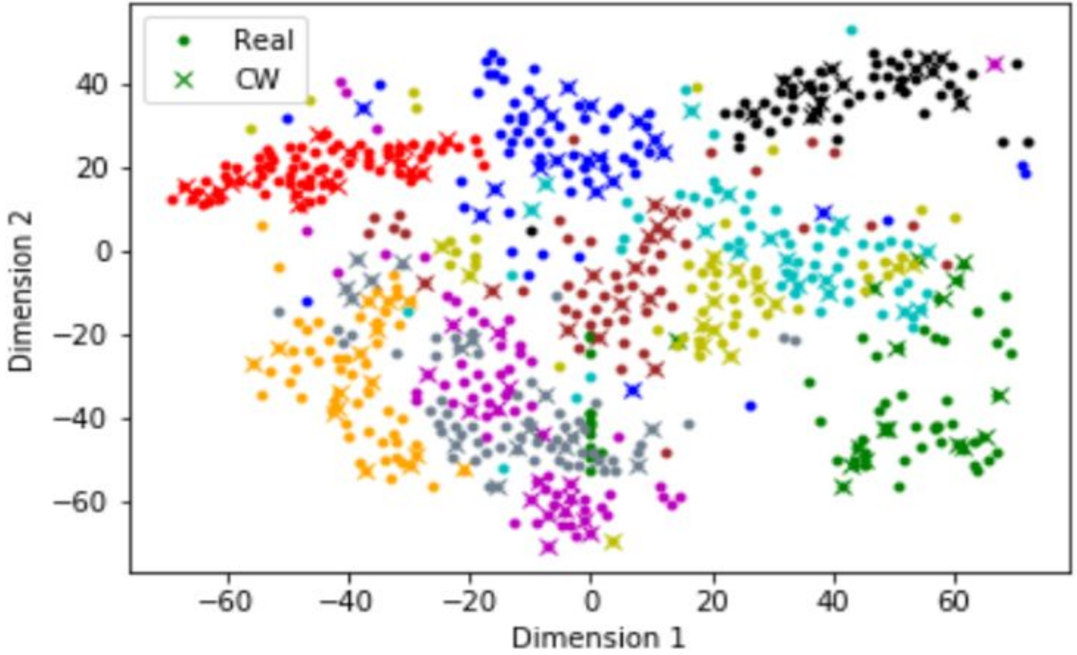
Lipschitz constrained latent encoder preserves the distances under a metric space on the latent and the data manifolds.

Gradient Norm Penalty to enforce **Lipschitz constrain**

$$\mathcal{L}_l = \sum_{i=1}^{B} \left(\left\|\nabla_x f_\theta(x^{(i)})\right\|_2 - K\right)^2$$

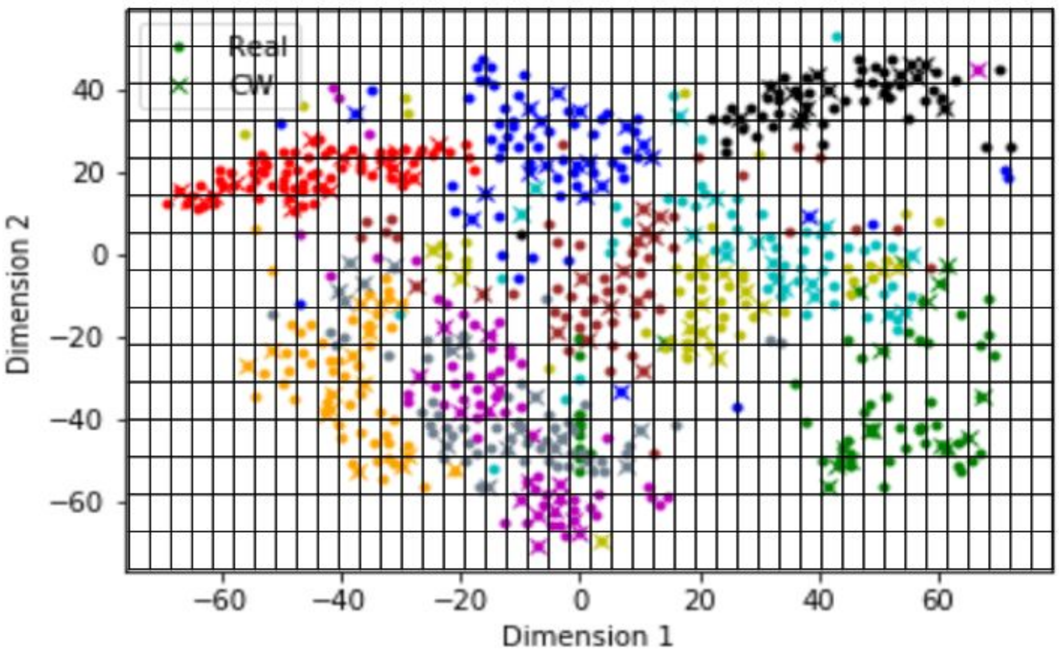
Lipschitz constrained latent space

- Below Figure is a 2D t-SNE plot of the latent encodings (from the Lipschitz constrained encoder) of the true and the CW attacked adversarial samples from the MNIST data.
- It can be seen that embeddings of the adversaries are extremely close to those of the true samples.



Lipschitz constrained Quantized latent space

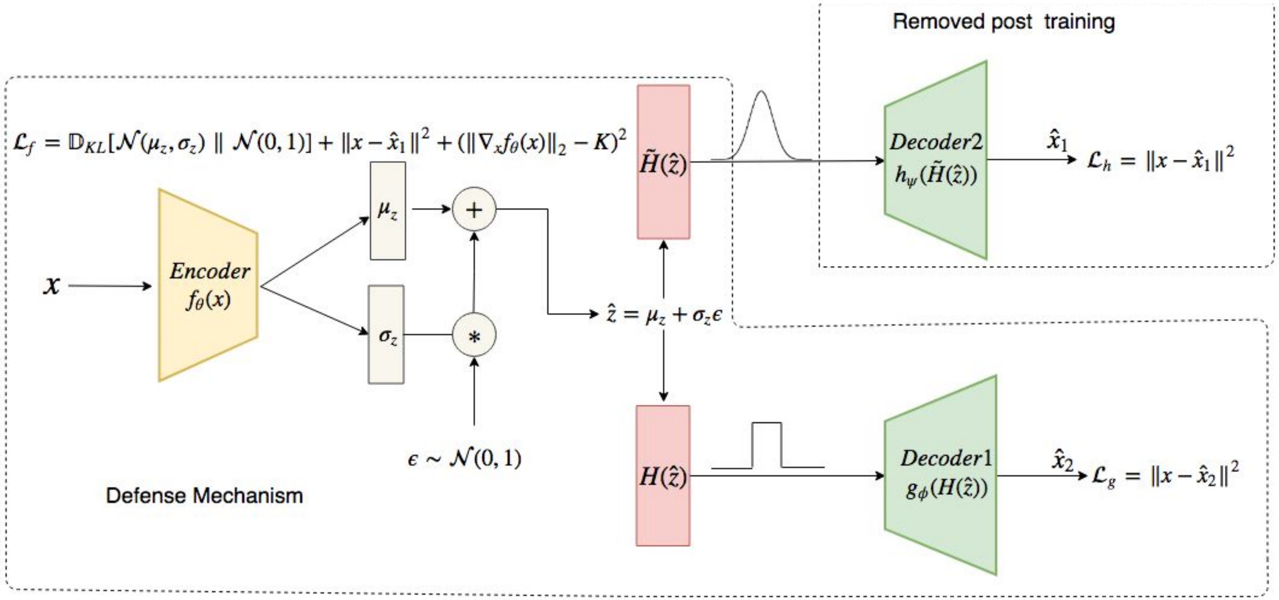
- Below Figure shows the Overlay of Grid on the 2D t-SNE plot of the latent encodings (from the Lipschitz constrained encoder) of the true and the CW attacked adversarial samples from the MNIST data.
- It can be seen that embeddings of the adversaries and true samples fall into the same grid cell.
- If we quantize the latent space, then latent embeddings of the adversaries and true samples will be same.



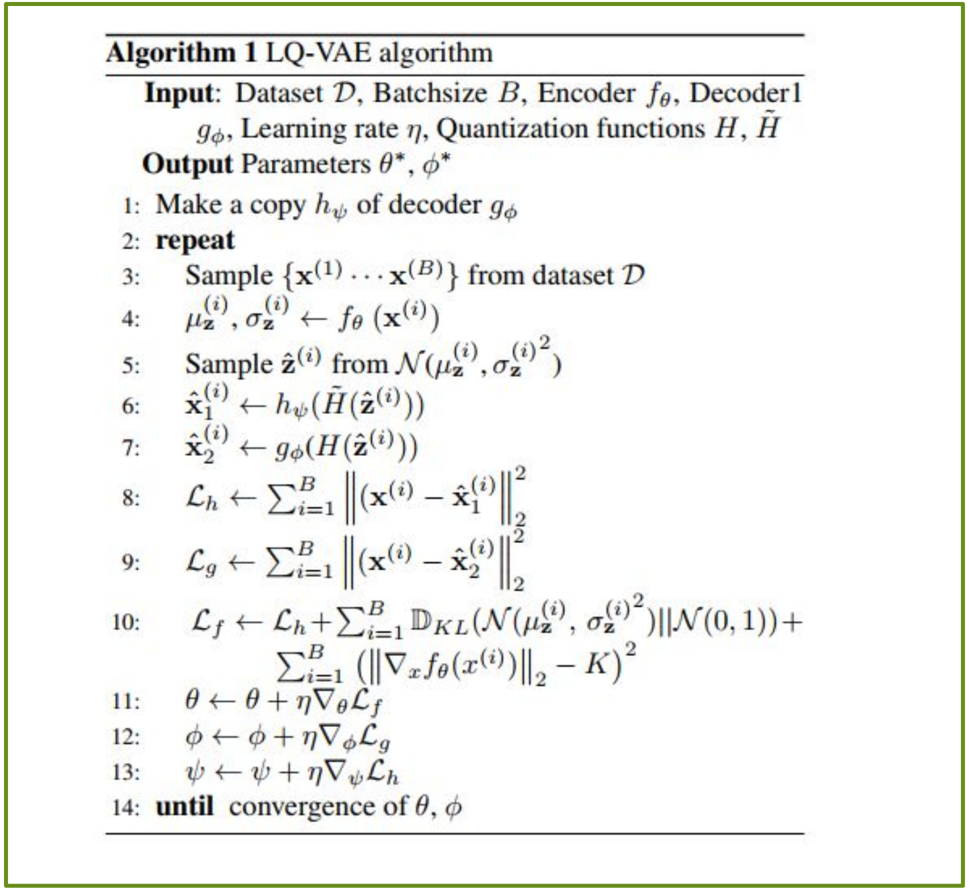
Proposed Architecture

- Encoder encodes the given image and produces latent vector, we quantize it and feed it to decoder.
- As quantization is non differentiable, loss propagation cannot happen, so for training we use another decoder which works on soft quantized latent vector, which is differentiable.

Fig. 1. **Proposed LQ-VAE - A Lipschitz constrained encoder (L_f) encodes the input image into a latent space quantized by the function H which is explored through a stochastic perturbation (ϵ). During inference, Decoder1 maps the quantized latent codes generated by the adversarial images back to the image space. Training is done only on real data samples by using an approximate differentiable version of Decoder1 (i.e. Decoder2).**


Removed post training
$\mathcal{L}_f = \mathbb{D}_{KL}[\mathcal{N}(\mu_z, \sigma_z) \parallel \mathcal{N}(0,1)] + \|x - \hat{x}_1\|^2 + (\|\nabla_x f_\theta(x)\|_2 - K)^2$
x
Encoder
$f_\theta(x)$
μ_z
σ_z
+
*
$\epsilon \sim \mathcal{N}(0,1)$
$\hat{z} = \mu_z + \sigma_z \epsilon$
$\tilde{H}(\hat{z})$
$H(\hat{z})$
Decoder2
$h_\psi(\tilde{H}(\hat{z}))$
$\hat{x}_1$
$\mathcal{L}_h = \|x - \hat{x}_1\|^2$
Decoder1
$g_\phi(H(\hat{z}))$
$\hat{x}_2$
$\mathcal{L}_g = \|x - \hat{x}_2\|^2$
Defense Mechanism

Proposed Algorithm

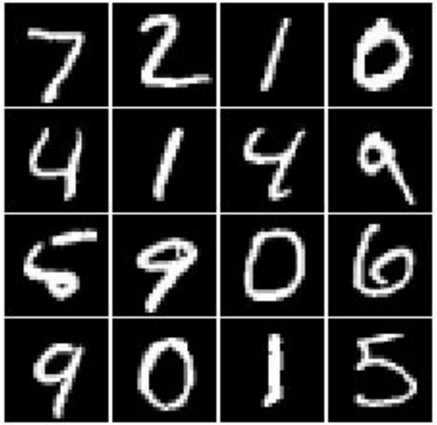


Algorithm 1 LQ-VAE algorithm

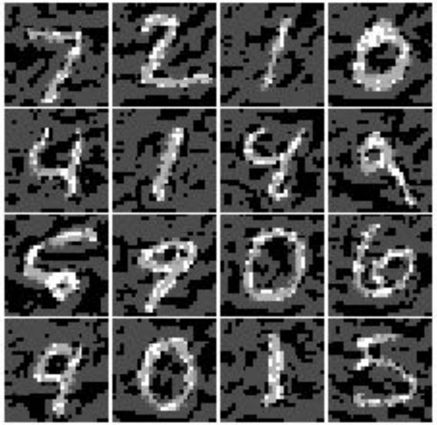
```
Input: Dataset $\mathcal{D}$, Batchsize $B$, Encoder $f_\theta$, Decoder1 $g_\phi$, Learning rate $\eta$, Quantization functions $H$, $\tilde{H}$
Output Parameters $\theta^*$, $\phi^*$
Make a copy $h_\psi$ of decoder $g_\phi$
repeat
    Sample $\{\mathbf{x}^{(1)} \cdots \mathbf{x}^{(B)}\}$ from dataset $\mathcal{D}$
    $\mu_{\mathbf{z}}^{(i)}, \sigma_{\mathbf{z}}^{(i)} \leftarrow f_\theta\left(\mathbf{x}^{(i)}\right)$
    Sample $\hat{\mathbf{z}}^{(i)}$ from $\mathcal{N}(\mu_{\mathbf{z}}^{(i)}, {\sigma_{\mathbf{z}}^{(i)}}^2)$
    $\hat{\mathbf{x}}_1^{(i)} \leftarrow h_\psi(\tilde{H}(\hat{\mathbf{z}}^{(i)}))$
    $\hat{\mathbf{x}}_2^{(i)} \leftarrow g_\phi(H(\hat{\mathbf{z}}^{(i)}))$
    $\mathcal{L}_h \leftarrow \sum_{i=1}^{B} \left\|(\mathbf{x}^{(i)} - \hat{\mathbf{x}}_1^{(i)}\right\|_2^2$
    $\mathcal{L}_g \leftarrow \sum_{i=1}^{B} \left\|(\mathbf{x}^{(i)} - \hat{\mathbf{x}}_2^{(i)}\right\|_2^2$
    $\mathcal{L}_f \leftarrow \mathcal{L}_h + \sum_{i=1}^{B} \mathbb{D}_{KL}(\mathcal{N}(\mu_{\mathbf{z}}^{(i)}, {\sigma_{\mathbf{z}}^{(i)}}^2)||\mathcal{N}(0,1)) + \sum_{i=1}^{B} \left(\left\|\nabla_x f_\theta(x^{(i)})\right\|_2 - K\right)^2$
    $\theta \leftarrow \theta + \eta \nabla_\theta \mathcal{L}_f$
    $\phi \leftarrow \phi + \eta \nabla_\phi \mathcal{L}_g$
    $\psi \leftarrow \psi + \eta \nabla_\psi \mathcal{L}_h$
until convergence of $\theta$, $\phi$
```

Results: MNIST

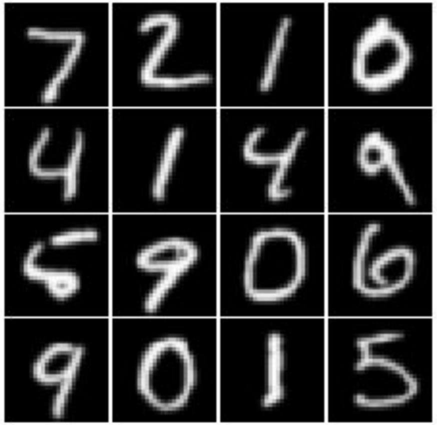
- Left block shows the real mnist images, whose classification accuracy is 100% and
- Middle block images are adversarial images generated by using the FGSM method, whose classification accuracy has dropped to 6%
- Right block shows the LQ-VAE output on the Adversarial images, whose classification accuracy is 94%



100%



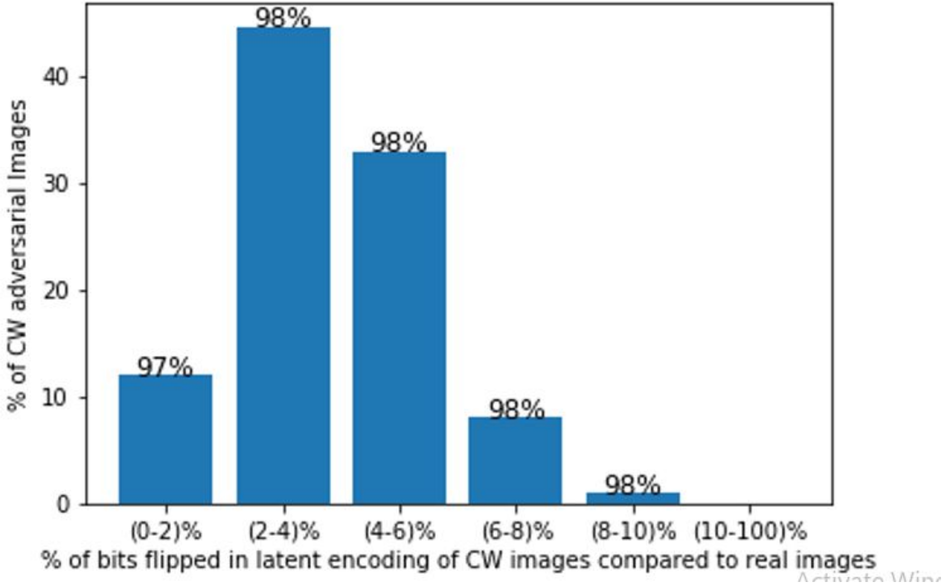
6%



94%

Analysis of Lipschitz Quantized Latent Vectors of Real and Adversarial Example

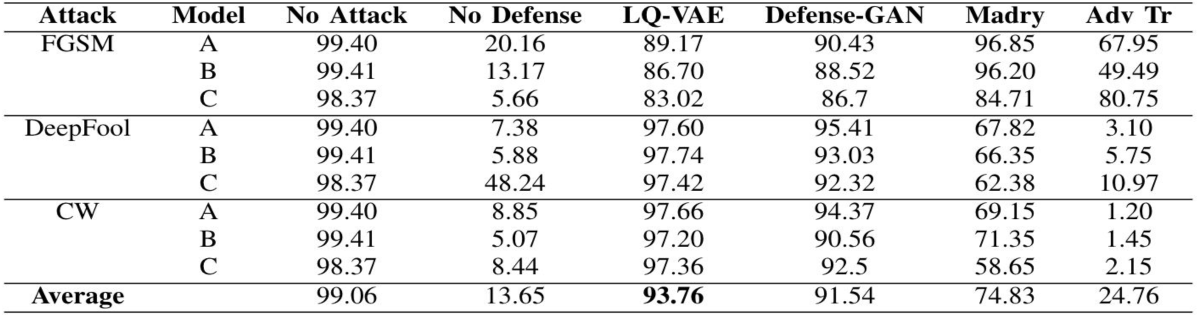
- This is the analysis of latent vectors of real image and its adversarial image in LQ-VAE
- Figure depicts the distribution of the bit-flippings in the latent codes of the CW adversaries on MNIST data
- It can be seen that about 90% of the total adversaries undergo less than 6% of bits being flipped resulting in high classification accuracy (seen on top of the bars).
- First Bar denotes that 12% of adversarial images have undergone less than 2% of bit flipping in latent space compared to real images. For this bucket the classification accuracy of LQ-VAE filtered images is 97%



Results: MNIST - White Box Attack

- White Box Attack means Attacker has the access to original classifier for generating adversarial images
- This shows the performance of LQ-VAE against various attacks like FGSM, DeepFool and CW on different types of classifiers A, B, C.
- It can be seen that on an average LQ-VAE is better than Defense-GAN and other defense mechanisms

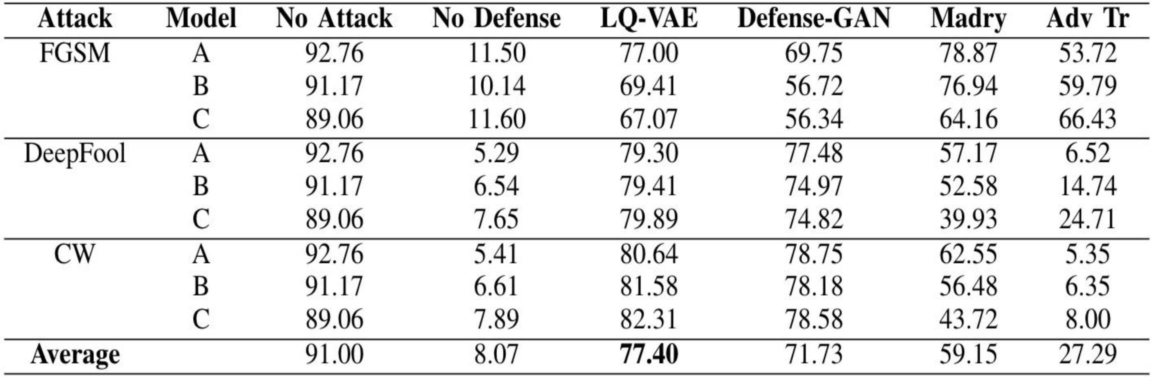
CLASSIFICATION ACCURACY OF THE MNIST CLASSIFIERS ON WHITE BOX ATTACKS WITH VARIOUS DEFENSE STRATEGIES.



Attack	Model	No Attack	No Defense	LQ-VAE	Defense-GAN	Madry	Adv Tr
FGSM	A	99.40	20.16	89.17	90.43	96.85	67.95
	B	99.41	13.17	86.70	88.52	96.20	49.49
	C	98.37	5.66	83.02	86.7	84.71	80.75
DeepFool	A	99.40	7.38	97.60	95.41	67.82	3.10
	B	99.41	5.88	97.74	93.03	66.35	5.75
	C	98.37	48.24	97.42	92.32	62.38	10.97
CW	A	99.40	8.85	97.66	94.37	69.15	1.20
	B	99.41	5.07	97.20	90.56	71.35	1.45
	C	98.37	8.44	97.36	92.5	58.65	2.15
Average		99.06	13.65	**93.76**	91.54	74.83	24.76

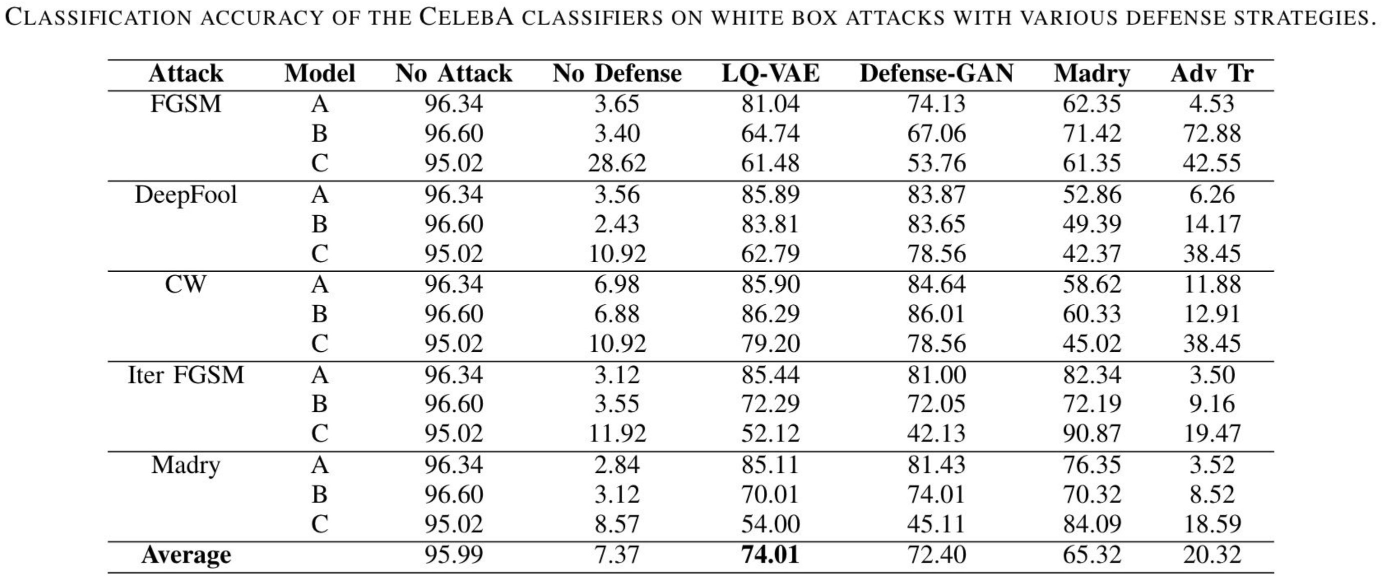
Results: FMNIST - White Box Attack

CLASSIFICATION ACCURACY OF THE FMNIST CLASSIFIERS ON WHITE BOX ATTACKS WITH VARIOUS DEFENSE STRATEGIES



Attack	Model	No Attack	No Defense	LQ-VAE	Defense-GAN	Madry	Adv Tr
FGSM	A	92.76	11.50	77.00	69.75	78.87	53.72
	B	91.17	10.14	69.41	56.72	76.94	59.79
	C	89.06	11.60	67.07	56.34	64.16	66.43
DeepFool	A	92.76	5.29	79.30	77.48	57.17	6.52
	B	91.17	6.54	79.41	74.97	52.58	14.74
	C	89.06	7.65	79.89	74.82	39.93	24.71
CW	A	92.76	5.41	80.64	78.75	62.55	5.35
	B	91.17	6.61	81.58	78.18	56.48	6.35
	C	89.06	7.89	82.31	78.58	43.72	8.00
Average		91.00	8.07	**77.40**	71.73	59.15	27.29

Results: CelebA - White Box Attack

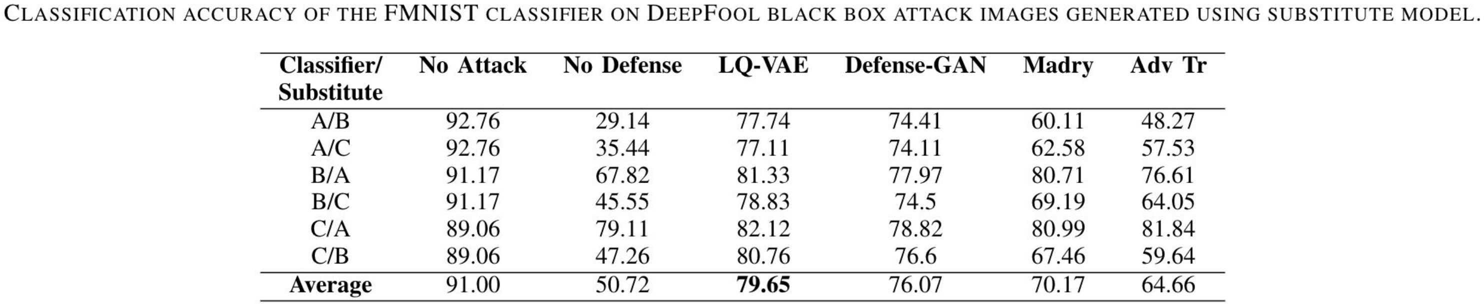


CLASSIFICATION ACCURACY OF THE CELEBA CLASSIFIERS ON WHITE BOX ATTACKS WITH VARIOUS DEFENSE STRATEGIES.

Attack	**Model**	**No Attack**	**No Defense**	**LQ-VAE**	**Defense-GAN**	**Madry**	**Adv Tr**
FGSM	A	96.34	3.65	81.04	74.13	62.35	4.53
	B	96.60	3.40	64.74	67.06	71.42	72.88
	C	95.02	28.62	61.48	53.76	61.35	42.55
DeepFool	A	96.34	3.56	85.89	83.87	52.86	6.26
	B	96.60	2.43	83.81	83.65	49.39	14.17
	C	95.02	10.92	62.79	78.56	42.37	38.45
CW	A	96.34	6.98	85.90	84.64	58.62	11.88
	B	96.60	6.88	86.29	86.01	60.33	12.91
	C	95.02	10.92	79.20	78.56	45.02	38.45
Iter FGSM	A	96.34	3.12	85.44	81.00	82.34	3.50
	B	96.60	3.55	72.29	72.05	72.19	9.16
	C	95.02	11.92	52.12	42.13	90.87	19.47
Madry	A	96.34	2.84	85.11	81.43	76.35	3.52
	B	96.60	3.12	70.01	74.01	70.32	8.52
	C	95.02	8.57	54.00	45.11	84.09	18.59
Average		95.99	7.37	**74.01**	72.40	65.32	20.32

Results: FMNIST - Black Box Attack

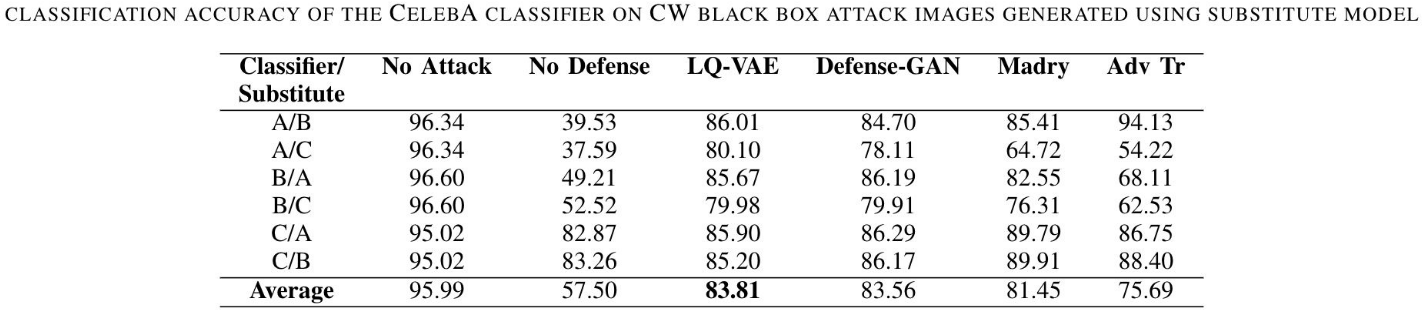
- Black Box Attack means Attacker has no access to original classifier, so attacker has to generate the adversarial images using substitute model
- This shows the performance of LQ-VAE against DeepFool Black Box attack on different types of classifiers A, B, C.
- It can be seen that on an average LQ-VAE is better than Defense-GAN and other defense mechanisms



CLASSIFICATION ACCURACY OF THE FMNIST CLASSIFIER ON DEEPFOOL BLACK BOX ATTACK IMAGES GENERATED USING SUBSTITUTE MODEL.

Classifier/ Substitute	**No Attack**	**No Defense**	**LQ-VAE**	**Defense-GAN**	**Madry**	**Adv Tr**
A/B	92.76	29.14	77.74	74.41	60.11	48.27
A/C	92.76	35.44	77.11	74.11	62.58	57.53
B/A	91.17	67.82	81.33	77.97	80.71	76.61
B/C	91.17	45.55	78.83	74.5	69.19	64.05
C/A	89.06	79.11	82.12	78.82	80.99	81.84
C/B	89.06	47.26	80.76	76.6	67.46	59.64
Average	91.00	50.72	**79.65**	76.07	70.17	64.66

Results: CelebA - Black Box Attack

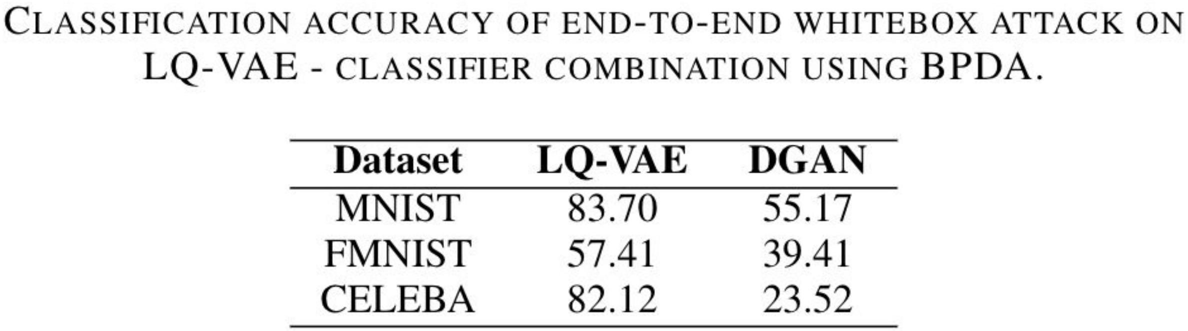


CLASSIFICATION ACCURACY OF THE CELEBA CLASSIFIER ON CW BLACK BOX ATTACK IMAGES GENERATED USING SUBSTITUTE MODEL

Classifier/ Substitute	**No Attack**	**No Defense**	**LQ-VAE**	**Defense-GAN**	**Madry**	**Adv Tr**
A/B	96.34	39.53	86.01	84.70	85.41	94.13
A/C	96.34	37.59	80.10	78.11	64.72	54.22
B/A	96.60	49.21	85.67	86.19	82.55	68.11
B/C	96.60	52.52	79.98	79.91	76.31	62.53
C/A	95.02	82.87	85.90	86.29	89.79	86.75
C/B	95.02	83.26	85.20	86.17	89.91	88.40
Average	95.99	57.50	**83.81**	83.56	81.45	75.69

Results: BPDA (Backward Pass Differentiable Approximation)

This shows that On BPDA LQ-VAE Performs better than Defense Gan



CLASSIFICATION ACCURACY OF END-TO-END WHITEBOX ATTACK ON LQ-VAE - CLASSIFIER COMBINATION USING BPDA.

Dataset	LQ-VAE	DGAN
MNIST	83.70	55.17
FMNIST	57.41	39.41
CELEBA	82.12	23.52

Conclusions

- Proposed a novel generative model based defense mechanism LQ-VAE
- Through Experiments we have shown our method works very well across various datasets, different types of classifiers and defends various adversarial attacks
- The experiments shows that the proposed method surpasses the state-of-the-art techniques in several cases.
- Proposed method is faster than the current best state of art method Defense GAN, as Defense GAN involves a run time search on the latent space
- so the proposed method is faster and accurate

Thank You