The Autoencoding Variational Autoencoder

Probabilistic Machine Learning Reading Group

Roman Kouznetsov

University of Michigan Department of Statistics

Table of Contents

1. Review of VAE (and AE)
2. Composing Optimization for AVAE
3. Big Picture Impact
4. Applications
5. Conclusion

1 Review of VAE (and AE)

2 Composing Optimization for AVAE

3 Big Picture Impact

4 Applications

5 Conclusion

What's in an AE?

- An autoecoder lets us learn **representations**.
- Representations frequently take on lower dimensions the "represents" a compressed form of the input; this is what we call a **bottleneck**.

Why Bottlenecks?

- Bottlenecks allow us to learn low-dimensional features of the input data.
- **Goal**: Use the bottlenecks to maximize reconstruction BUT not just memorize inputs; i.e. actually learn the representation.

How do we prevent overfitting?

There are several ways we can (try to) prevent our AE from just memorizing inputs.

- Limit the number of nodes in the bottleneck layer.
  - This does not guarantee a prevention of input memorization; certain AEs with 1 node in the bottleneck layer can reconstruct just fine.
- Use a Regularizer!!
  - L1 Regularization
  - KL Regularization

AE Applications

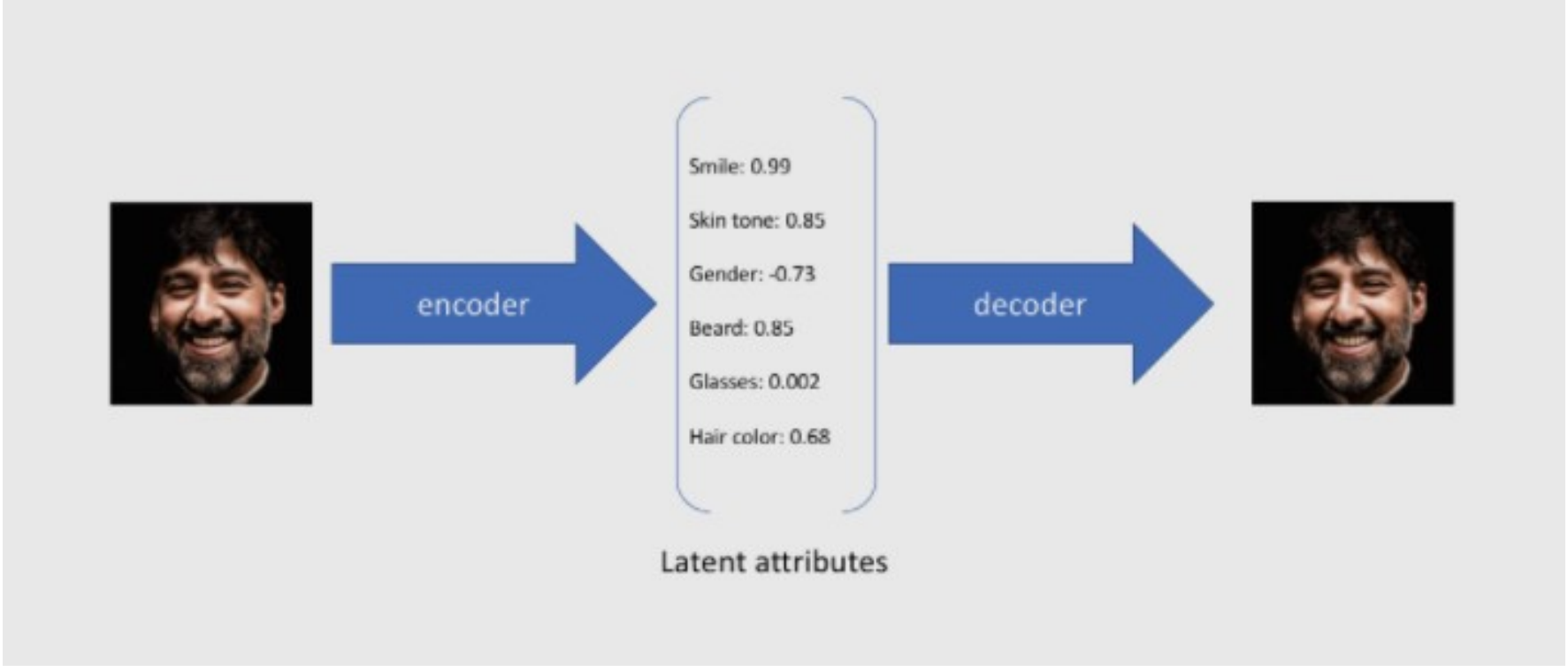
- Data Denoising
- Feature Extraction
- Data Generation

Shortcomings of AE

- Effectively only learns a point estimate.
- Can be very sensitive to input perturbations.

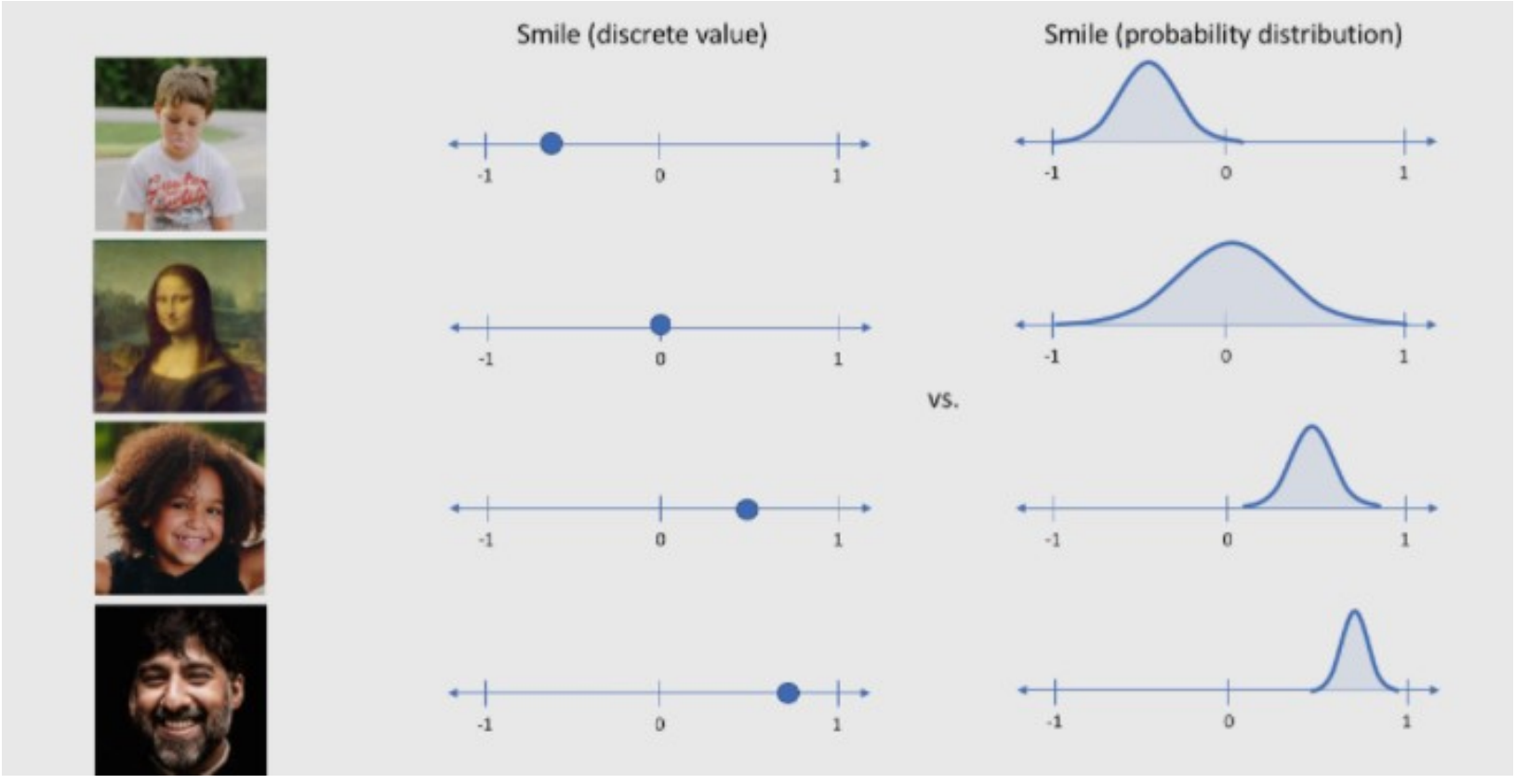
Enter VAE

- VAEs allow latent representation to be defined probabilistically!



Enter VAE (cont.)

- VAEs allow latent representation to be defined probabilistically!



The VAE Model

- Prior: $Z \sim p(Z) = \mathcal{N}(0, 1)$
- Decoder: $X|Z \sim p(X|Z, \theta) = \mathcal{N}(g(Z|\theta), vI)$
- Encoder: $q(Z|X, \eta) \sim \mathcal{N}(f^{\mu}(X|\eta), f^{\Sigma}(X|\eta))$

The VAE model attempts to use an approximation to the exact posterior $P(Z|X)$ that is more tractable and easy to work with (hence the Gaussian distribution), while still getting good reconstructions.

Review: The ELBO

The ELBO is the objective we use to maximize reconstruction, but keep the approximated distribution close to the true posterior.

This ensures that we are learning features probabilistically and not just memorizing inputs.

Maximizing the ELBO is the same as minimizing the KL.

Review: The ELBO, Proof

$$
\begin{aligned}
KL(\mathcal{Q}||\mathcal{P}) &= E_{\mathcal{Q}} log\left(\frac{\mathcal{Q}}{\mathcal{P}}\right) \\
&= E_{\mathcal{Q}} log\left(\frac{\pi(X) q(Z|X,\eta)}{p(X,Z|\theta)}\right) \\
&= \underbrace{E_{\mathcal{Q}} log(\pi(X))}_{\text{This is just } log(\pi(X))} + E_{\mathcal{Q}} log(q(Z|X,\eta)) - E_{\mathcal{Q}} log(p(X,Z|\theta))
\end{aligned}
$$

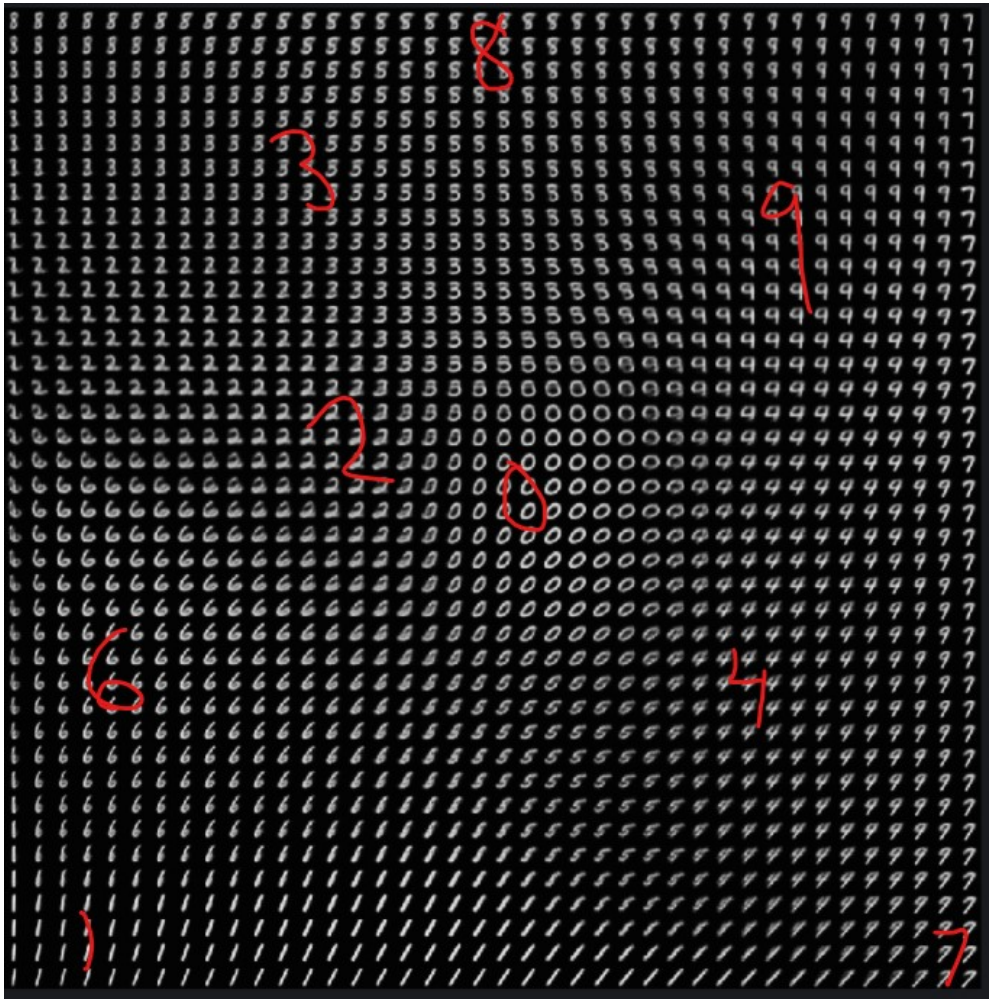
Some basic algebra yields:

$$
\underbrace{log(\pi(X))}_{\perp Z} - KL(\mathcal{Q}||\mathcal{P}) = \underbrace{E_{\mathcal{Q}} log(p(X,Z|\theta) - E_{\mathcal{Q}} log(q(Z|X,\theta)}_{ELBO}
$$

Note: Sometimes we don't have access to $\pi(X)$ so we need to use the empirical distribution as a plug-in estimate.

Applications of VAE

The applications of the VAE incorporate many of those of AEs, but add the convenience of random sampling!



Benefits of VAE

- Random sampling of continuous latent space, which is Gaussian.
- Creation of better defined boundaries between classes of outputs.
  - This allows for us to understand what the computer deems as close and to actually obtain these boundary condition inputs (e.g. 4s that look like 9s).
- Better interpolation. Here was a mind blowing example: MusicVAE

Is VAE the best we can do?

Corny Answer: No, otherwise why are we here right now.

Copout Answer: It depends on what you want to use the model for.

Author's Answer: If the goal is to probabilistically encoder an input, no.

1 Review of VAE (and AE)

2 Composing Optimization for AVAE

3 Big Picture Impact

4 Applications

5 Conclusion

What's Wrong with VAE?

- Can suffer from model misspecification.
- Doesn't actually encode inputs correctly sometimes.

9..4..8... what?

(a)

(b)

1. (a) Iterated forward passes of a VAE (hence the title)
2. (b) Iterated forward passes of Autoencoded VAE (very consistent 9)

Extended VAE Model

$$(Z, Z') \sim p_\rho(Z, Z') = \mathcal{N}\left(\begin{pmatrix} Z \\ Z' \end{pmatrix}; \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} I & \rho I \\ \rho I & I \end{pmatrix}\right)$$

$$X \mid Z \sim p(X \mid Z, \theta) = \mathcal{N}(X; g(Z; \theta), vI)$$

$$X' \mid Z' \sim p(X' \mid Z', \theta) = \mathcal{N}(X'; g(Z'; \theta), vI)$$

Q: What's different from the vanilla VAE?

A:

- Model is now specified as a joint of 2 observations.
- Model gains a correlation (coupling) parameter ρ.

The VAE Embedded

$$\bar{\mathcal{P}} \equiv p(X'|Z';\theta)p(X|Z;\theta)p_\rho(Z',Z) \tag{6}$$

Proposition 2.1. *Approximating $\bar{\mathcal{P}}$ with a distribution of form*

$$\bar{\mathcal{Q}} \equiv \hat{\pi}(X)q(Z|X,\eta)p_\rho(Z'|Z)p(X'|Z',\theta) \tag{7}$$

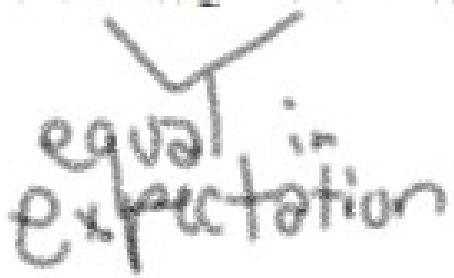
where $\hat{\pi}$ is the empirical data distribution and q and p are the encoder and the decoder models respectively gives the original VAE objective in (3).

Proof: *See Appendix A.1.*

Proof. By definition

$$\mathcal{KL}(\bar{\mathcal{Q}}|\bar{\mathcal{P}}) = -\left\langle \log \frac{p(X|Z;\theta)p(Z)p_\rho(Z'|Z)p(X'|Z';\theta)}{\hat{\pi}(X)q(Z|X,\eta)p_\rho(Z'|Z)p(X'|Z',\theta)} \right\rangle_{\bar{\mathcal{Q}}} = \mathcal{KL}(\mathcal{Q}|\mathcal{P})$$

Equality follows as for any test function of form $f(X,Z)$, that does not depend on X' and Z' we have $\langle f(X,Z)\rangle_{\bar{\mathcal{Q}}} = \langle f(X,Z)\rangle_{\mathcal{Q}}$. □



The AVAE Model

- $\mathcal{P}_\rho = p(x \mid z;\theta)p(z)p_\rho\left(z' \mid z\right) u(\tilde{x})$
- $\mathcal{Q}_{\text{AVAE}} = \underbrace{q\left(z' \mid \tilde{x};\eta\right) p_\theta(\tilde{x} \mid z)}_{\text{auxiliary data}} \underbrace{q(z \mid x;\eta)\pi(x)}_{\text{training data}}$

Some careful scrutiny will reveal that the AVAE is effectively the reformed VAE model **but with the auxiliary observation no longer depending on a latent variable.**

Philosophy

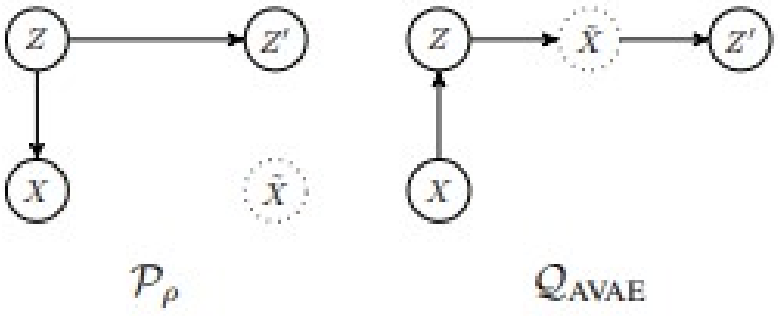


Figure 2: Graphical model of the extended target distribution $\mathcal{P}_\rho$, and the variational approximation $\mathcal{Q}_{\text{AVAE}}$. Here $\tilde{X}$ is a sample generated by the decoder that is subsequently encoded by the encoder.

The philosophy of the AVAE model is that in truth, the auxiliary sample $\tilde{X}$ will be independent from X, but we want our approximation to actually keep them similar, so $\tilde{X}$ is dependent on X in the approximation because $\tilde{X}$ is X's reconstruction.

tl;dr: The AVAE ensures that our encoder is not exclusively performing well on training data, but is also very consistent on UNSEEN data that is similar to training data.

Derivation of Objective (as used by authors)

Just apply the ELBO to our new target and approximation!

Approximation: $Q_{AVAE} = \pi(X)q(z|X,\eta)q(Z'|\tilde{X},\eta)p_\theta(\tilde{X}|Z)$
Target: $P_\rho = p(X|Z,\theta)p(Z)p_\rho(Z'|Z)u(\tilde{X})$

$$-KL(Q_{AVAE}||P_\rho) = -E_{Q_{AVAE}}\left(log\left(\frac{Q_{AVAE}}{P_\rho}\right)\right)$$
$$= E_{Q_{AVAE}}\left(log\left(\frac{P_\rho}{Q_{AVAE}}\right)\right)$$

Derivation (cont.)

$$
\begin{aligned}
-KL(Q_{AVAE}||P_\rho) &= -E_{Q_{AVAE}}\left(log\left(\frac{Q_{AVAE}}{P_\rho}\right)\right) \\
&= E_{Q_{AVAE}}\left(log\left(\frac{P_\rho}{Q_{AVAE}}\right)\right) \\
&= E_{Q_{AVAE}}\left(log\left(\frac{p(X|Z,\theta)p(Z)p_\rho(Z'|Z)\underbrace{u(\tilde{X})}_{1}}{\pi(X)q(z|X,\eta)q(Z'|\tilde{X},\eta)p_\theta(\tilde{X}|Z)}\right)\right) \\
&= E_{Q_{AVAE}}\left(-log\left(\frac{\pi(X)q(Z|X,\eta)}{p(x|Z,\theta)p(Z)}\right)\right) \\
&\quad + E_{Q_{AVAE}}\left(log(p(Z'|Z)\right) - E_{Q_{AVAE}}\left(log(q(Z'|\tilde{X},\eta)\right) \\
&\quad - \underbrace{E_{Q_{AVAE}}(log(p_\theta(\tilde{X}|Z)}
\end{aligned}
$$

can ignore b/c independent of variational params

Derivation (cont.)

$$
\begin{aligned}
-KL(Q_{AVAE}||P_\rho) &= E_{Q_{AVAE}}\left(-log\left(\frac{\pi(X)q(Z|X,\eta)}{p(x|Z,\theta)p(Z)}\right)\right) \\
&\quad + E_{Q_{AVAE}}\left(log(p(Z'|Z)) - E_{Q_{AVAE}}\left(log(q(Z'|\tilde{X},\eta)\right)\right) \\
&= -KL(\pi(X)q(Z|X,\eta)||p(x|Z,\theta)p(Z)) \\
&\quad + E_{Q_{AVAE}}\left(log(p(Z'|Z)) - E_{Q_{AVAE}}\left(log(q(Z'|\tilde{X},\eta)\right)\right) \\
&:= \mathcal{B}_{AVAE}
\end{aligned}
$$

Penalty Interpretation

$$
\begin{aligned}
\mathcal{B}_{\mathrm{AVAE}} =^{+} & -\mathcal{KL}(\pi(X)q(Z \mid X;\eta)\|p(X \mid Z;\theta)p(Z)) \text{ original ELBO} \\
& + \left\langle \log p_{\rho}\left(Z' \mid Z\right)\right\rangle_{\tilde{q}(Z;\eta)\tilde{q}_{\theta}(Z'|Z;\eta)} \text{ ensure consistent encoding} \\
& - \left\langle \log q\left(Z' \mid \tilde{X};\eta\right)\right\rangle_{q(Z'|\bar{X};\eta)\tilde{q}_{\theta}(\bar{X};\eta)} \text{ ensure good decoding}
\end{aligned}
$$

1 Review of VAE (and AE)

2 Composing Optimization for AVAE

3 Big Picture Impact

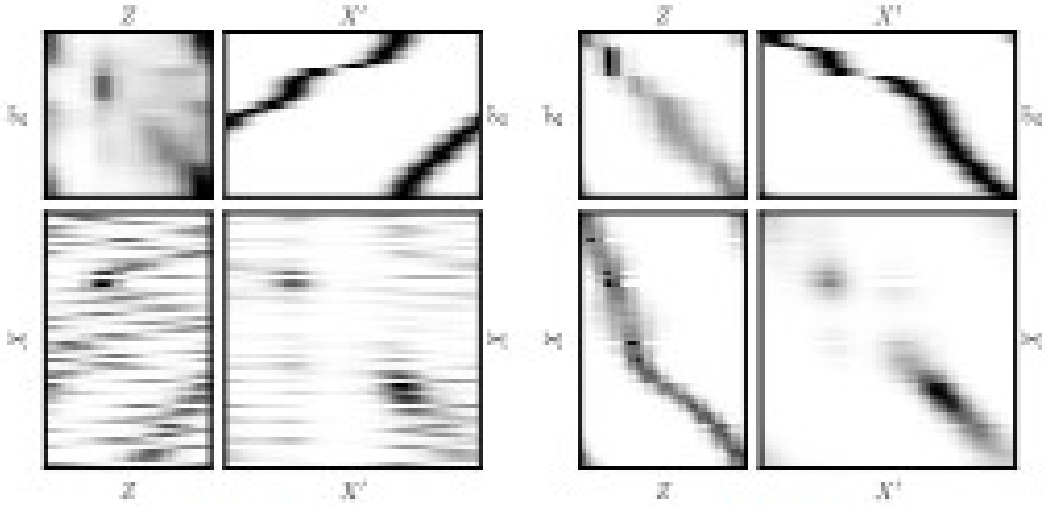
4 Applications

5 Conclusion

VAE v. AVAE

In cases of coupled observations, we can see that AVAE imposes a more symmetric distribution on the joint of (X, X') and - therefore - (Z, Z').

AVAE allows for better reconstruction, and theory says that in the limit $v \to 0$ the approximation $Q_{AVAE}(Z'|Z, \eta, \theta) = p(Z'|Z, \rho = 1)$. In other words, if observations are perfectly known, the approximation is equal to the true posterior in the limit.



1 Review of VAE (and AE)

2 Composing Optimization for AVAE

3 Big Picture Impact

4 Applications

5 Conclusion

Applied Experiments

The AVAE and a handful of others were trained to perform classification on

- colorMNIST
- celebA

with and without adversarial attacks.

Flow of Evaluation

Evaluating these models for classification can be done in many ways so it is important to understand why this method evaluates what we want to evaluate:

Procedure:

1. Train the encoder-decoder pair normally.
2. At some point, free the encoder parameters and use them to train a linear classifier without attacks. This is done by passing values from the latent space to the decoder and using those outputs to solve the classification.
3. Add the PGD attack and see the change!

Q: Why this evaluation in performance?

A: Remember, we want our VAE (or AVAE or whatever else) to be very good at understanding boundary conditions. Small perturbations ideally would not completely confuse our working model to think an input belongs to a different class. So, we evaluate classification in this environment.

What is a PGD Attack?

A PGD attack is an attack that has access to your model parameters.

The idea is that if we attack input data, our model would screw up.

Q: Isn't this obvious? Modifying input data will change how the network works.

A: Yes, but could you make an attack that wasn't noticeable to humans?

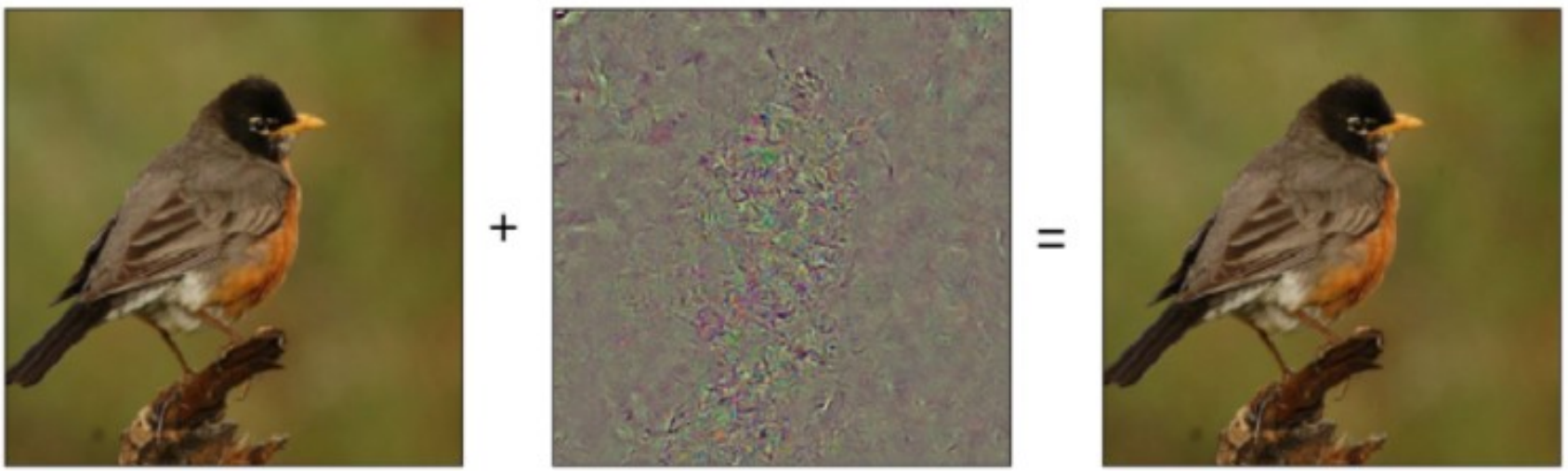
What is a PGD Attack? (cont.)

A: Yes, but could you make an attack that wasn't noticeable to humans?

We can accomplish this by taking gradient steps in loss maximizing directions WHILE controlling for the size of the perturbation.

This constraint of size (ϵ) is why the attack follows a projected gradient descent. We project onto the L^p ball of radius ϵ after taking a gradient step so that the attack is of controlled size post-update.

PGD Attack Example



Left: natural image. Middle: Adversarial perturbation found by PGD attack against ResNet50 model, size of perturbation is magnified x100 to be more visible. Right: adversarial example.

Results

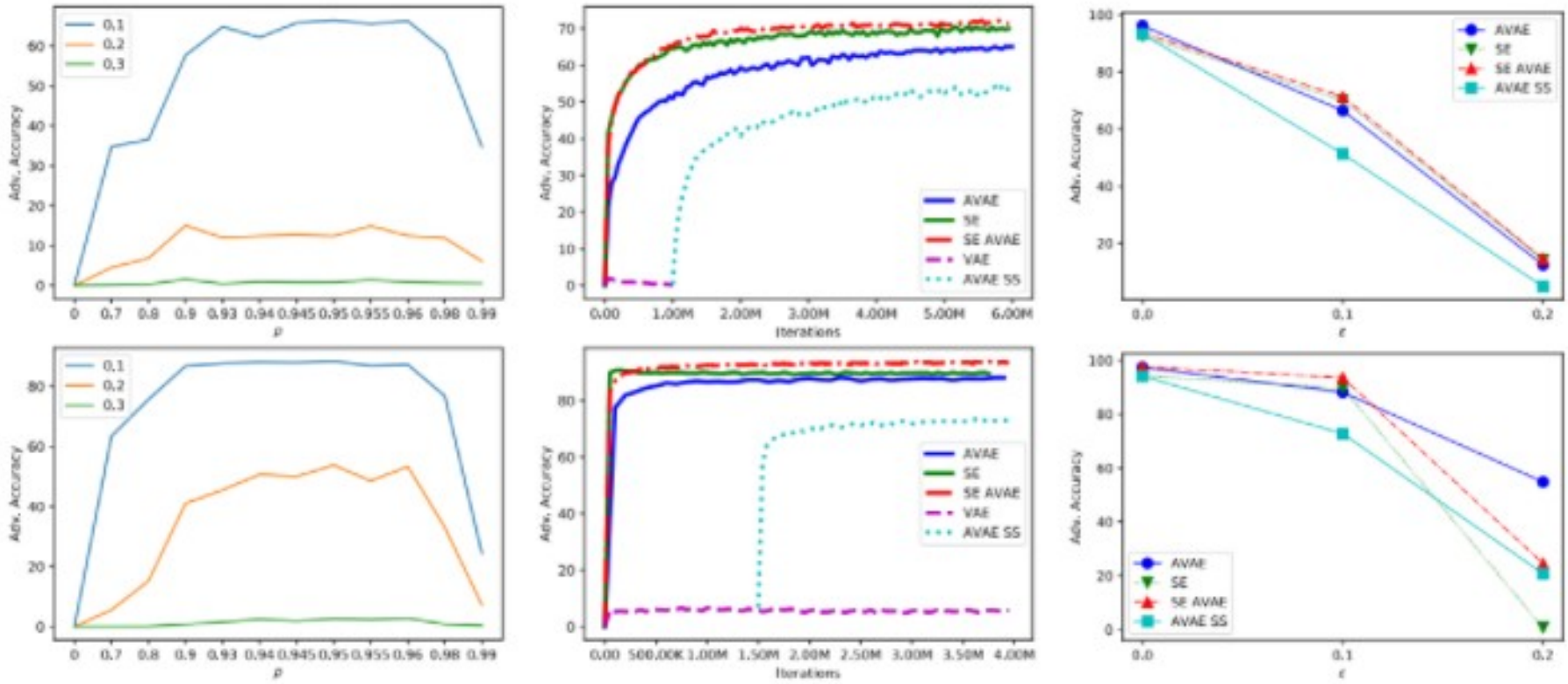
Performance on ColorMNIST Classification

we expect this to be high as a baseline

Task	digit			color			Time	MSE	FID
ϵ	0.0	0.1	0.2	0.0	0.1	0.2			
VAE	93.8	5.8	0.0	100.0	19.9	2.0	× 1	1369.2	12.44
$SE^5_{0.1}$	94.3	89.6	1.8	100.0	100.0	21.8	× 4	1372.5	13.01
$SE^5_{0.2}$	95.7	92.6	87.3	100.0	99.9	99.9	× 4	1374.9	11.72
AVAE	97.3	88.1	54.8	100.0	99.8	87.7	× 1.5	1371.9	15.46
$SE_{0.1}$-AVAE	97.4	93.6	24.5	100.0	100.0	60.0	× 4.7	1373.3	13.90
$SE_{0.2}$-AVAE	97.6	94.2	79.8	100.0	100.0	83.2	× 4.7	1374.3	13.89
AVAE SS	94.1	72.8	20.8	100.0	99.6	56.8	× 1.5	1379.3	12.44

Results (cont.)

Performance on ColorMNIST Classification Breakdown



Results (cont.): ☹ Models

Performance on celebA Classification

	AVAE		SE^5		SE^{20} [5]		SE^5 AVAE		AVAE SS	
Task / ϵ	0.0	0.1	0.0	0.1	0.0	0.1	0.0	0.1	0.0	0.1
Bald	97.9	85.2	97.9	72.0	97.4	86.5	97.9	87.0	97.8	70.0
Mustache	96.1	91.5	95.0	69.5	95.7	84.4	96.0	92.3	94.9	74.3
Necklace	86.1	78.4	87.8	56.7	88.0	78.9	86.1	80.3	88.0	59.7
Eyeglasses	95.4	68.9	95.9	20.3	95.7	33.0	95.4	67.5	94.4	57.1
Smiling	77.7	3.6	87.0	3.10	85.7	1.1	77.9	6.3	81.4	0.9
Lipstick	81.0	7.3	83.9	2.0	80.3	0.6	80.2	11.5	80.7	0.9
Time	×2.2		×3.1		×7.8		×4.3		×2.2	
MSE	7276.6		7208.8		N/A		7269.2		7347.3	
FID	97.92		98.00		N/A		109.4		99.8	

Table 2: Adversarial test accuracy (in percentage) of the representations for subset of classification tasks on CelebA. For SE methods, superscript L in SE^L denotes the number of PGD iterations used during training of the model.

1 Review of VAE (and AE)

2 Composing Optimization for AVAE

3 Big Picture Impact

4 Applications

5 Conclusion

Major Takeaways

- VAEs are awesome but might not actually learn consistent encodings.
- AVAE at the end of the day is similar to a VAE but with additional regularizers that enforce the encoder to be consistent and as a result yield consistent reconstructions.
- The time cost of AVAE is 50% larger than VAEs but is more robust against adversarial attacks.
- AVAE is an alternative model that offers robustness and consistency, especially in coupled batch inputs.

Lingering Thoughts

- The image comparing VAE and AVAE seems a bit unfair because it is generated under an assumption of X and $\tilde{X}$ being similar (symmetric in joint distribution). Will we always expect similar inputs?
- Could an extension of the VAE model to 3+ dimensions have even better improvements over vanilla VAE?
- Are there any theoretical guarantees of AVAE?
- Could the coupling parameter ρ be something that we can learn rather than set as a hyperparameter?
- Speaking of coupling, what happens if we generate data s.t. $\rho < 0.25$? Will AVAE still outperform VAE?

Lingering Thoughts (cont.)

- I largely excluded it from this presentation as it wasn't too important to the big picture, but why do the authors stress invariance to likelihood $P(X|\theta)$ and priors $p(Z)$?
- If ϵ got really large and the SE-AVAE model was trained alongside these adversarial inputs, would the model simply learn the adversarial attacks too? What happens if we used an attack with $\epsilon/4$ or $\epsilon/10$? Given provided results, I would still assume accuracy decreases as ϵ increases, but I'm not fully convinced.