

Auto-Encoding Variational Bayes (Variational Autoencoders)

Faisal Mohamed

- Core idea.
- Implementation.
- Experimental results.

Scalable Variational Inference

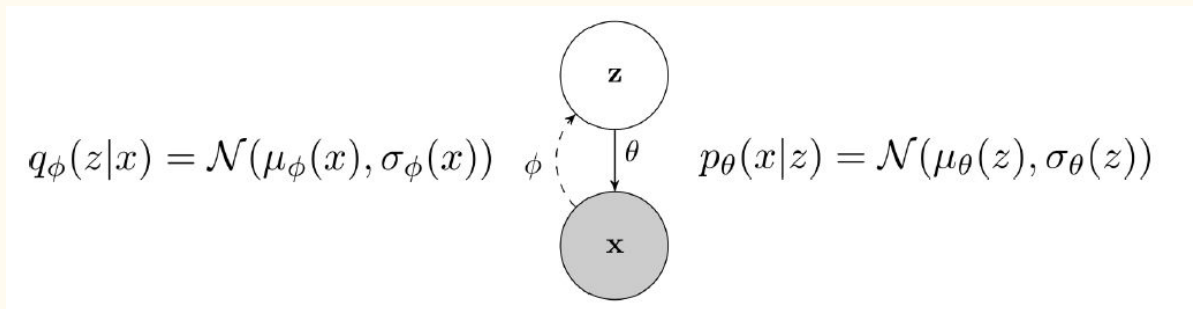
- Works on the intractable case.
- Can be optimized with SGD.

Problem setup

- We have a dataset \mathbf{X} of size N .
- We want to learn the data distribution $P(\mathbf{X})$
- We assume that there is some hidden lower dimensional variable called \mathbf{z} that can represent the data efficiently.
- The problem now is $P(\mathbf{X}) = \int P(\mathbf{X}|\mathbf{Z})P(\mathbf{Z}) d\mathbf{z}$
- This integral is intractable for vectors.

Varitional auto conoders

- Solve this problem by learning both $P(Z)$ and $P(X|Z)$.
- We sample z from $P(Z)$ network and pass it to $P(X|Z)$ to return a data sample.



Algorithm

$\theta, \phi \leftarrow$ Initialize parameters

repeat

$\mathbf{X}^M \leftarrow$ Random minibatch of M datapoints (drawn from full dataset)

$\epsilon \leftarrow$ Random samples from noise distribution $p(\epsilon)$

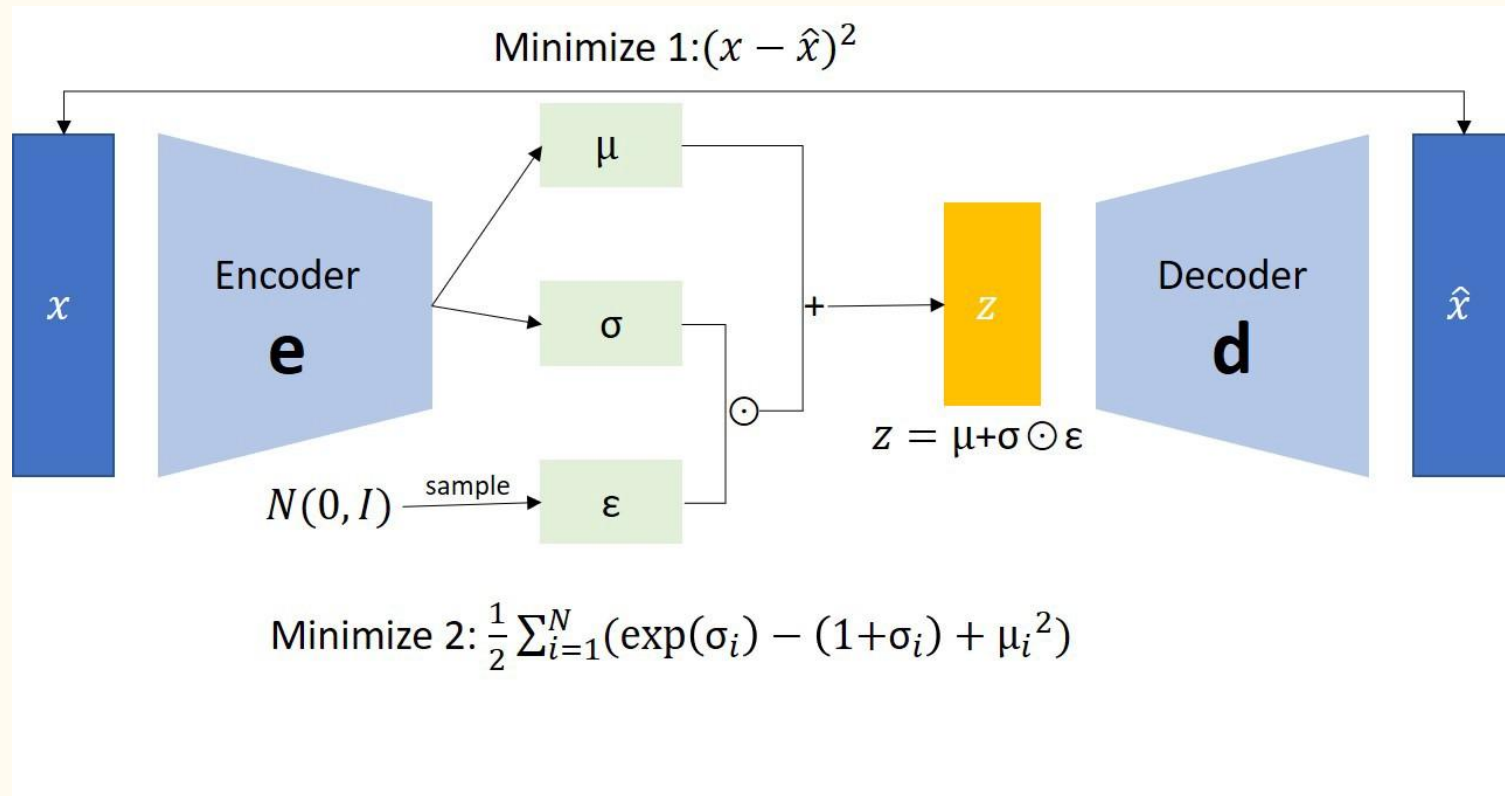
$\mathbf{g} \leftarrow \nabla_{\theta, \phi} \tilde{\mathcal{L}}^M(\theta, \phi; \mathbf{X}^M, \epsilon)$ (Gradients of minibatch estimator (8))

$\theta, \phi \leftarrow$ Update parameters using gradients \mathbf{g} (e.g. SGD or Adagrad [DHS10])

until convergence of parameters (θ, ϕ)

return θ, ϕ

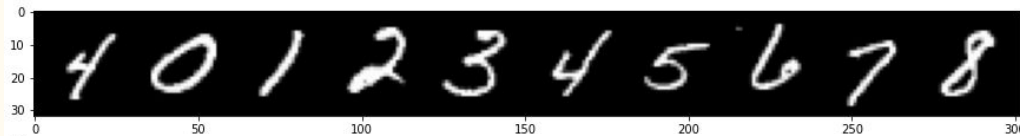
Algorithm



- Core idea.
- Implementation.
- Experimental results.

Experiment

- Dataset: MNIST
- Model: Single layer MLP with 500 units.



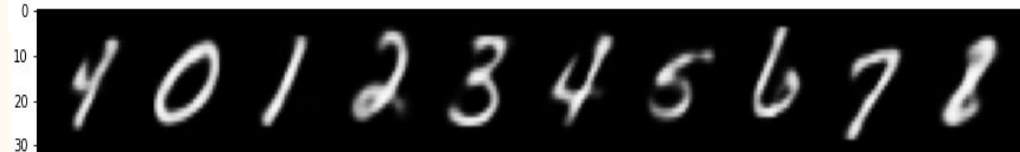
Original Image



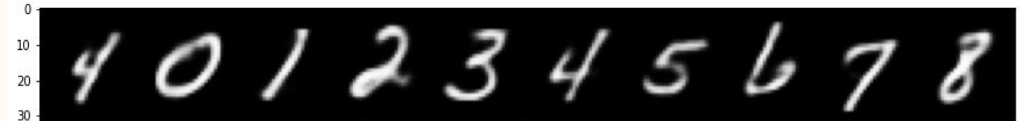
Size of $z = 3$



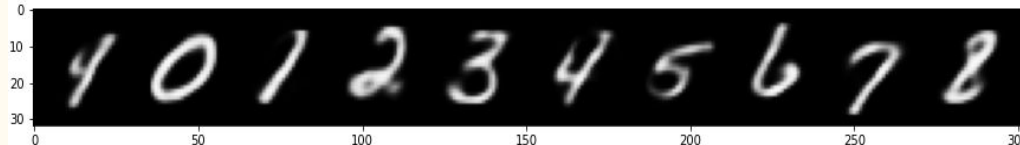
Size of $z = 5$



Size of $z = 10$



Size of $z = 20$



Size of $z = 200$