Atari, Anime and MADE Auto-encoders

Rogen George
Vivswan Shitole
Roli Khanna

The problem

Density Estimation

Density estimation is the construction of an estimate, based on observed data, of an unobservable underlying probability density function.

Probabilistic Modelling

We solve the problem of density estimation using probabilistic generative models.

It works on the principle of training a model to learn variables and their associated projected probabilities.

MADE

MADE, or Masked Auto-regressive Density Estimation, is a form of Probabilistic modelling, and works on the model of auto-encoders, and their adaptation to the concept of auto-regression.

Hypothesis

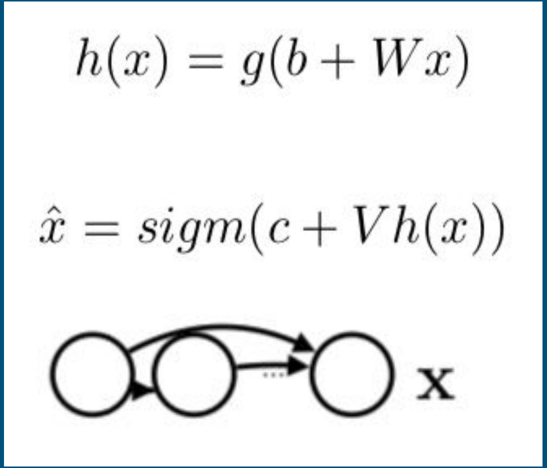
We propose that Deep MADE generates richer samples in comparison to Gaussian mixture models. It is a more complex and robust model for Density Estimation.

Approaches for Density Estimation

1. Neural Networks (Generative Adversarial Networks): Rather than using an intractable log-likelihood, this discriminator network provides the training signal in an adversarial fashion. Successfully trained GAN models can consistently generate sharp and realistically looking samples.
2. Probabilistic Modelling: Density estimation can be achieved using methods such as Auto-encoders, and Auto-encoders in the context of the Auto-regression property. Another proposed method for density estimation is using Gaussian Mixture Models.

Probabilistic Modelling for Density Estimation

Auto-encoders and Auto-regression



$$h(x) = g(b + Wx)$$

$$\hat{x} = sigm(c + Vh(x))$$

An Auto-encoder tries to learn hidden representations of the inputs, which points towards the original probability distribution that generated them.

Auto-regression builds upon Auto-encoders by using the "autoregressive property": A connection is made between the distribution of the new model by applying negative log likelihood to the definition of the probability of the Auto-encoder.

Gaussian Mixture Models

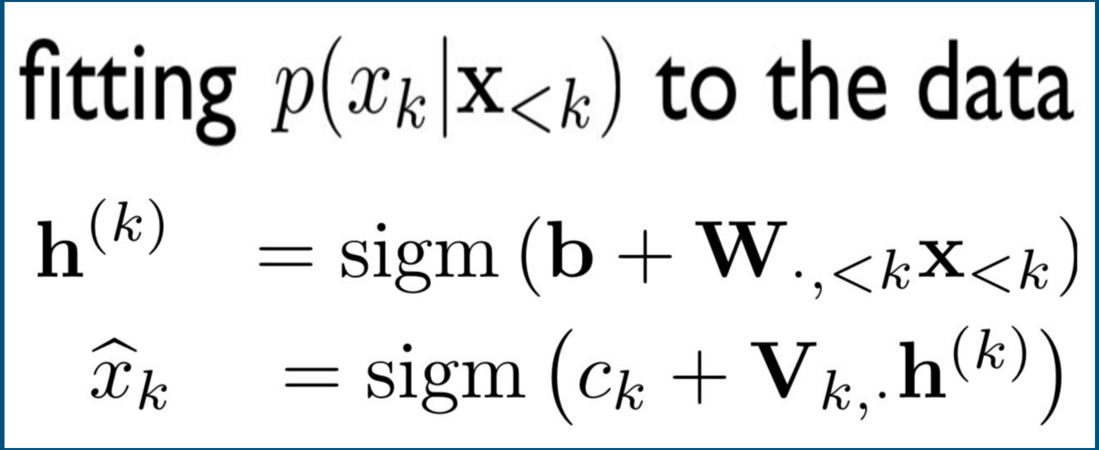
$$p(x) = \sum_{i=1}^{K} \phi_i N(x|\mu_i, \sigma_i)$$

Gaussian mixture models are a probabilistic model for representing a normal sub-distribution within an overarching probability distribution. Mixture models generally don't require knowing which subpopulation a data point belongs to, allowing the model to learn the subpopulations automatically. This makes GMMs a useful variant for estimating density as an unsupervised learning problem.

NADE

Neural Auto-regressive Density Estimation

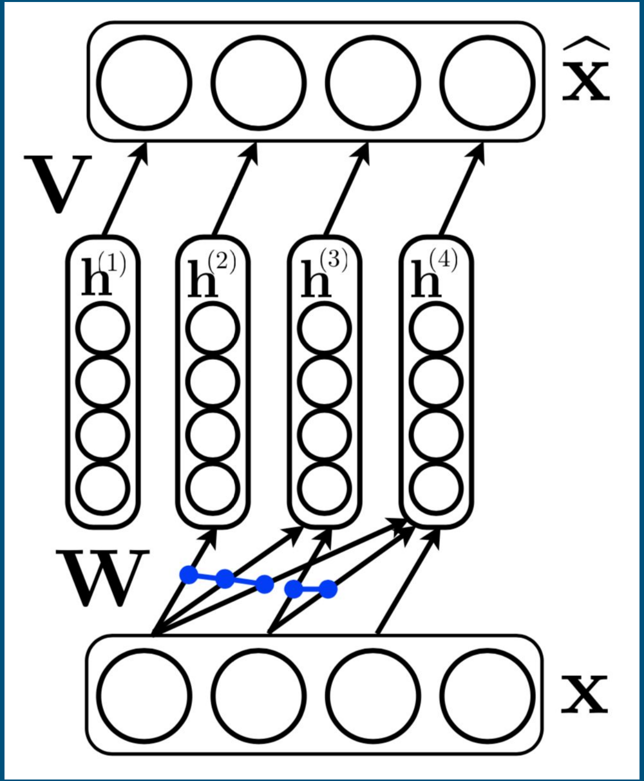
NADE: The Idea and Architecture



fitting $p(x_k|\mathbf{x}_{<k})$ to the data

$$\mathbf{h}^{(k)} = \mathrm{sigm}\left(\mathbf{b} + \mathbf{W}_{\cdot,<k}\mathbf{x}_{<k}\right)$$

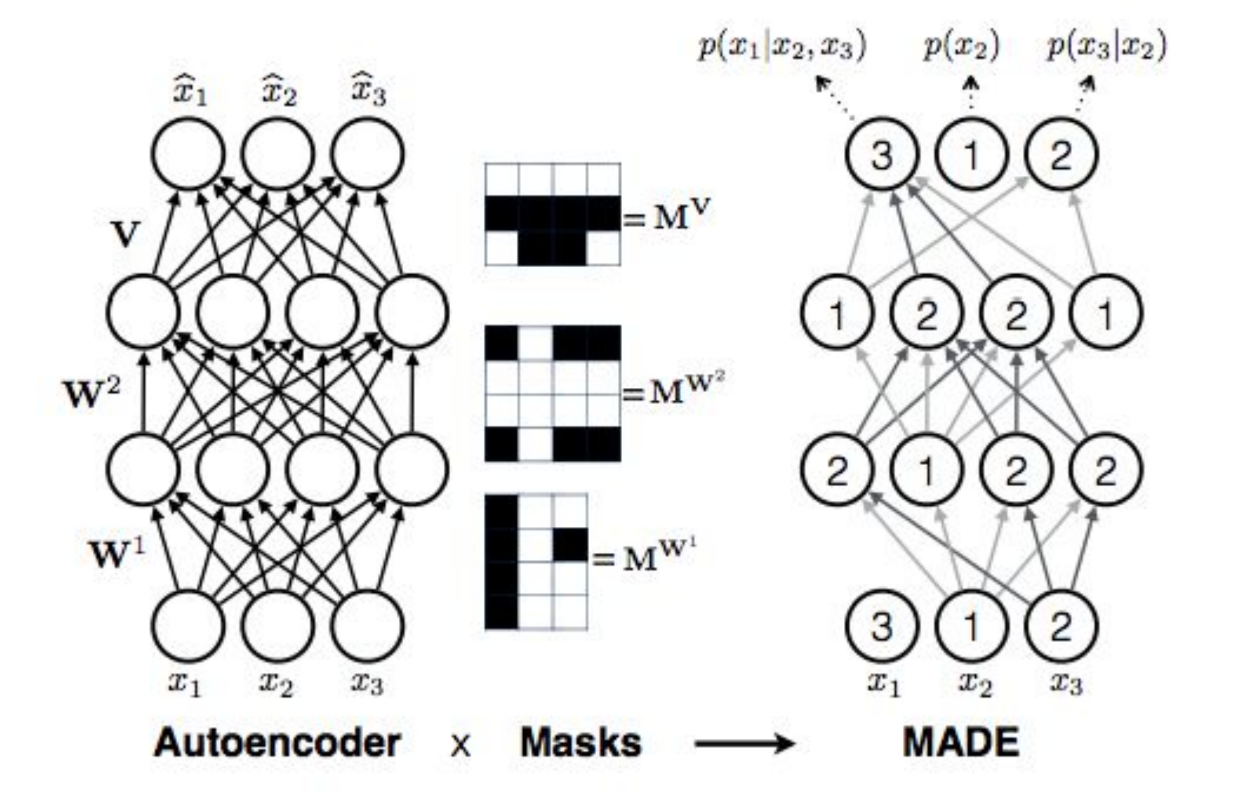
$$\widehat{x}_k = \mathrm{sigm}\left(c_k + \mathbf{V}_{k,\cdot}\mathbf{h}^{(k)}\right)$$

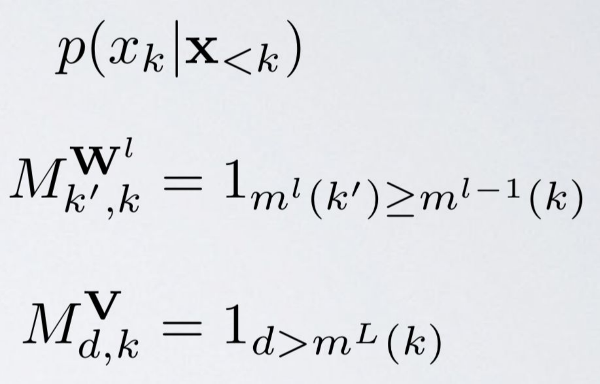


MADE

Masked Auto-regressive Density Estimation

Idea: Constrain the output so that it can be used for conditionals





$$p(x_k|\mathbf{x}_{<k})$$

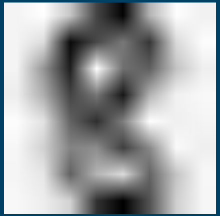
$$M^{\mathbf{W}^l}_{k',k} = 1_{m^l(k')\geq m^{l-1}(k)}$$

$$M^{\mathbf{V}}_{d,k} = 1_{d>m^L(k)}$$

Experiments

Phase 1

Experiments on MNIST Data: Gaussian Mixture Model



Standard MNIST digits dataset - 28X28

We split the dataset into 50000 training images and 10000 test images.

Applied PCA - preserving 99% variance

Result fed into GMM Model

110 Gaussians with a full covariance matrix.

Took a sample and applied Inverse PCA.

Phase 1

Experiments on MNIST Data: MADE

MADE model - single hidden layer

500 hidden units each layer.

50000 training images - batch size 100.

Masks changed every 20th sample of batch.

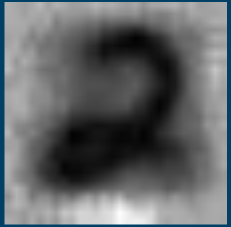
Cross entropy of the batch calculated.

Back propagated with Adam optimizer.

Learning rate of 0.001.

Phase 1

Experiments on MNIST Data: MADE



Training Epochs - 100

Test samples - 10000 test samples fed into the network.

Took a sample and generated a sample image.

Image was richer than the GMM model

Phase 2

Experiments on Atari Breakout: Gaussian Mixture Model

Atari Space Invaders Dataset atarigrandchallenge.com.

82700 game frame images, each of 240X160 pixels in RGB color format.

Converted to Binary.

Applied PCA - preserving 99% variance

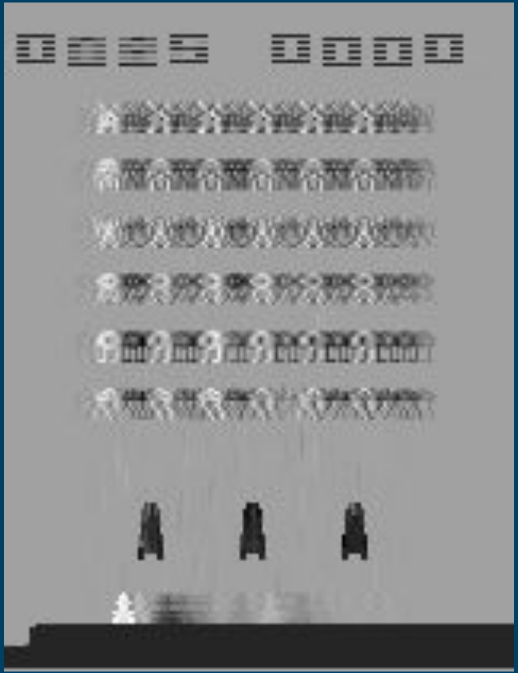
Result fed into GMM Model

110 Gaussians with a full covariance matrix.

Took a sample and applied Inverse PCA.

Phase 2

Experiments on Atari Breakout: Gaussian Mixture Model



Sampled Image

Phase 2

Experiments on Atari Breakout: MADE

MADE model - single hidden layer

500 hidden units each layer.

50000 training images - batch size 100.

Masks changed every 20th sample of batch.

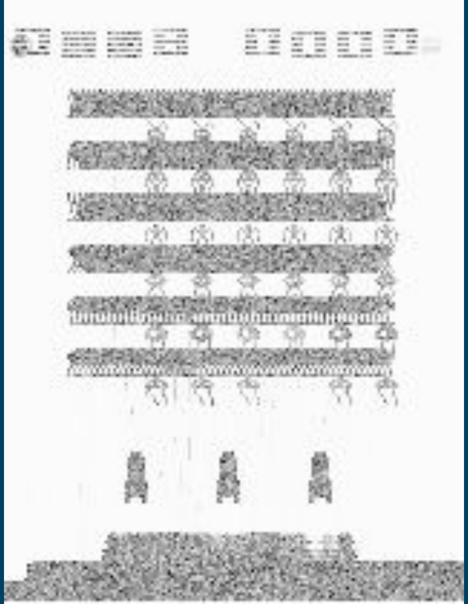
Cross entropy of the batch calculated.

Back propagated with Adam optimizer.

Learning rate of 0.001.

Phase 2

Experiments on Atari Breakout: MADE



Training Epochs - 100

Took a sample and generated a sample image.

Phase 3

Experiments on Anime faces: Gaussian Mixture Model

Inspired by the success on MNIST and ATARI, we thought of applying MADE to different datasets.

Searched for Datasets :

Less complex features.

Datasets applied to GAN.

Fruits, Pokemon,...

Anime Faces !!!

Phase 3

Experiments on Anime faces: Gaussian Mixture Model

21551 anime faces scraped from www.getchu.com

Cropped using the anime face detection algorithm by Nagadomi.

All images are resized to 64 X 64 pixels.

Images were grey scaled.

Phase 3

Experiments on Anime faces: Gaussian Mixture Model

Phase 3

Experiments on Anime faces: MADE

500 hidden layers - Single Hidden layer - Blurry images.

At least one unit for a pixel.

8000 hidden units * 3 layers.

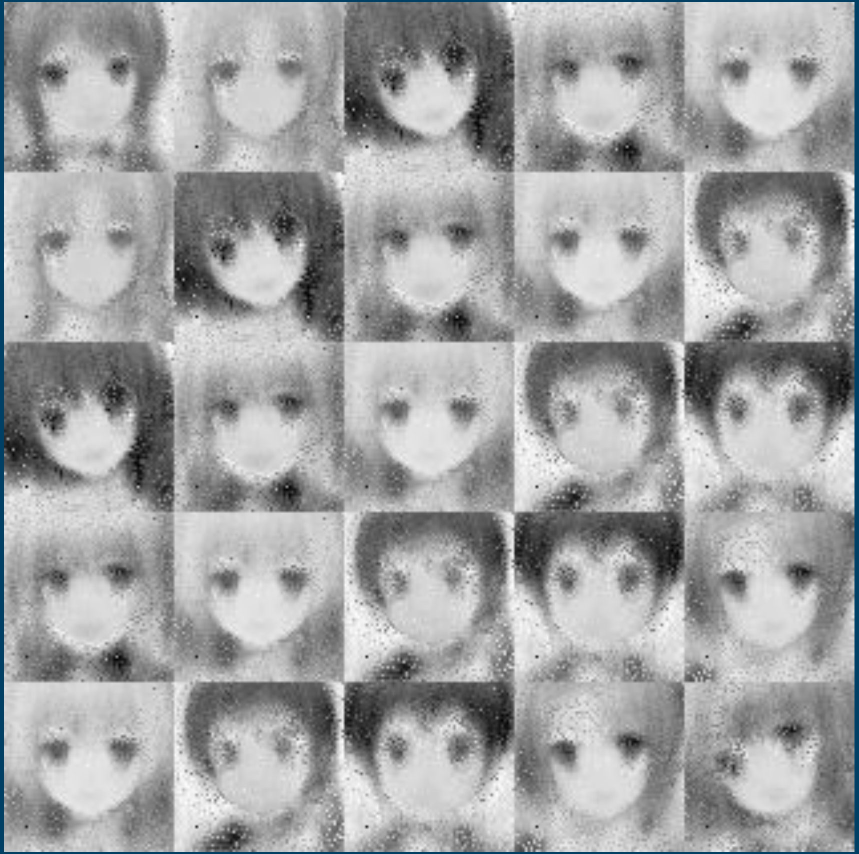
Number of Epochs - 100.

Loss Function - Mean Squared Loss.

Adam Optimizer.

Phase 3

Experiments on Anime faces: MADE



Phase 3

Experiments on Anime faces: MADE

Increased number of Epochs

No significant advantage

Increased Number of layers and Hidden units

No significant advantage.

Tried changing masks more frequently - over regularize MADE and provoke overfitting.

Insights

1. Robustness:

MADE models are very robust - the same hyperparameters can be used to model the distributions of diverse datasets such as MNIST digits, Atari game frames and Anime faces.

GMM can work for diverse datasets if we either have a mixture model with a large number of components (110 components used in our experiments) or if we change the number of components in the mixture according to the dataset whose distribution we are trying to model.

2. Accuracy:

We observe that the samples generated from the trained MADE model are closer to the binary input images compared to the samples generated from the trained GMM, which are greyscale.

Moreover, we observe that MADE model has captured distributions of individual pixels (the learnt distribution is in pixel space), while the GMM has captured distributions only in the principal component space as obtained from the PCA transform as a pre-processing step.

Future Scope.

Comparison with GAN 2.0 NVIDIA’s Hyper realistic Face Generator.

Dataset of 70,000 High quality Images at 1024 resolution.

Freely available at https://github.com/NVlabs/stylegan

Limiting Factors:

GPU Availability and Time.

Thank you!