LG Advanced Data Scientists Program
Deep Learning

[6: Generative Models]

Prof. Sungroh Yoon

Electrical & Computer Engineering | Seoul National University

(last compiled at 22:47:00 on 2020/02/25)

Outline

References

- *Deep Learning* by Goodfellow, Bengio and Courville ▸ Link
  - Chapter 19: Approximate Inference
  - Chapter 20: Deep Generative Models
- *Pattern Recognition and Machine Learning* by Bishop
  - Chapter 10: Approximate Inference
- online resources:
  - *Stanford CS231n: CNN for Visual Recognition* ▸ Link
  - *CVPR 2018 GAN Tutorial* ▸ Link
  - *NIPS 2016 Variational Inference Tutorial* ▸ Link
  - *NIPS 2016 GAN Tutorial* ▸ Link

Outline

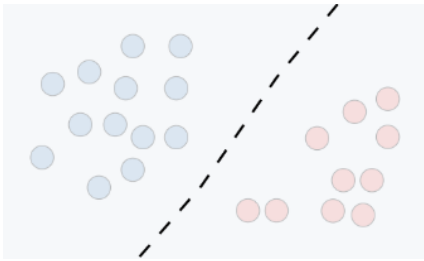
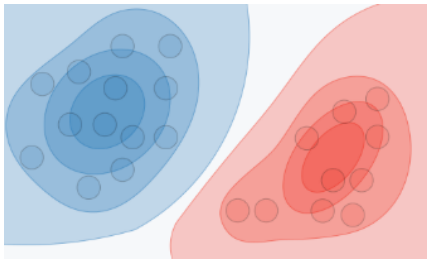
Supervised vs unsupervised learning

	supervised	**unsupervised**
data	$(\boldsymbol{x}, y)$ $\boldsymbol{x}$: data, y: label	$\boldsymbol{x}$ just data, no _____
goal	learn a *function* to map $\boldsymbol{x} \mapsto y$	learn inherent *structure* of the data
examples	classification regression object detection semantic segmentation	clustering dimensionality reduction representation learning density estimation

(source: cs231n)

Discriminative vs generative models

- assume supervised learning
  - goal: learn a *function* f to map $\boldsymbol{x} \mapsto y$

	discriminative	**generative**
goal	directly estimate $p(y \mid \boldsymbol{x})$	estimate $p(\boldsymbol{x} \mid y)$; then deduce[1] $p(y \mid \boldsymbol{x})$
should evaluate	$f(\boldsymbol{x}) = \operatorname{argmax}_y p(y \mid \boldsymbol{x})$	$f(\boldsymbol{x}) = \operatorname{argmax}_y p(\boldsymbol{x} \mid y)p(y)$
what's learned	decision boundary	probability _____ of data
		
examples	SVM, neural nets	Gaussian mixture, Bayes nets

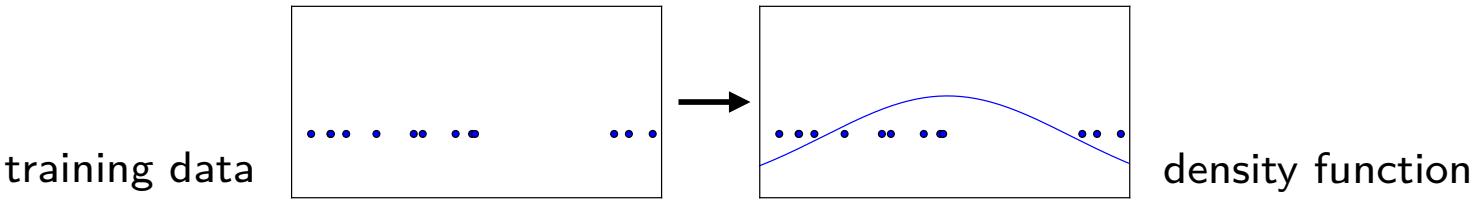
(source: [Ng and Jordan, 2002][2], stackoverflow)

[1] *e.g.* using Bayes rule: $p(y \mid \boldsymbol{x}) = \frac{p(\boldsymbol{x} \mid y)p(y)}{p(\boldsymbol{x})}$

[2] Ng, A. Y. and Jordan, M. I. (2002). On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. In *Advances in neural information processing systems*, pages 841–848

Generative modeling in unsupervised learning

- density estimation
  - goal: learn $p(\boldsymbol{x})$



- _____ generation
  - given training data, generate new samples from the same distribution

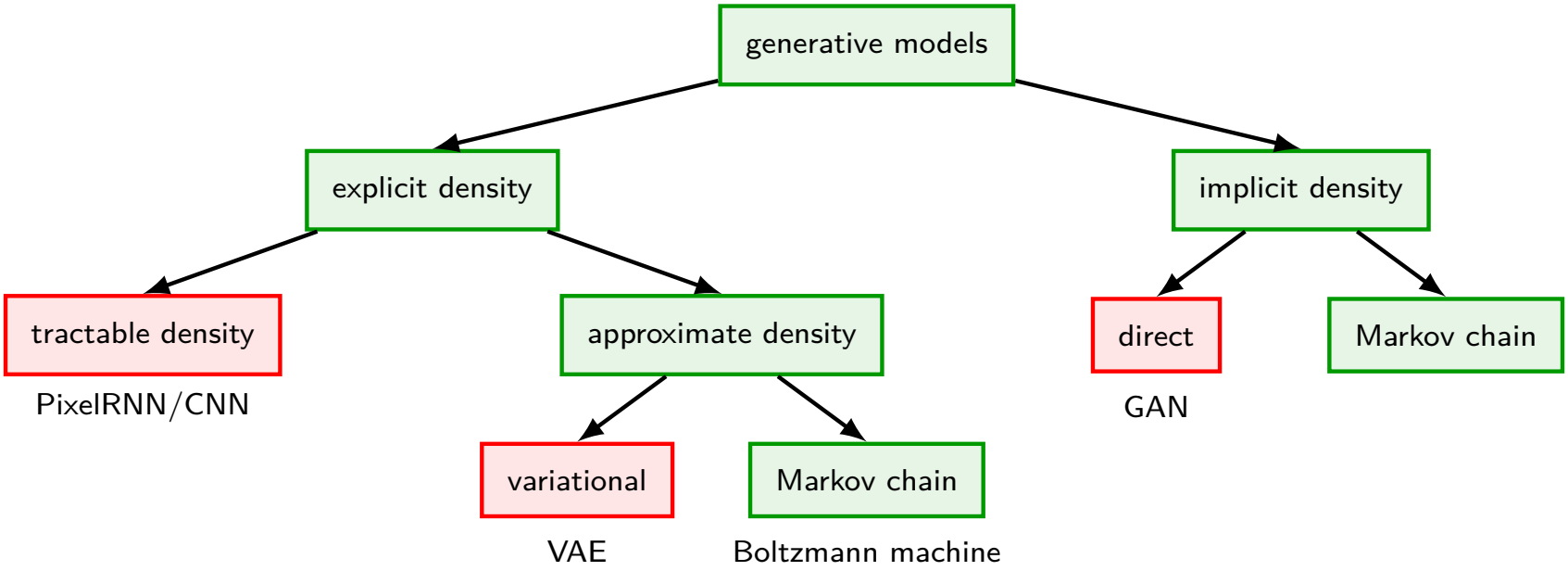
training data $\sim p_{\text{data}}(\boldsymbol{x})$

generated samples $\sim p_{\text{model}}(\boldsymbol{x})$

  - want to learn $p_{\text{model}}(\boldsymbol{x})$ similar to $p_{\text{data}}(\boldsymbol{x})$

(source: Goodfellow (2018), Karras (2017))

Taxonomy of generative models



(source: Goodfellow, 2017)

1. explicit density estimation
   - explicitly define and solve for $p_{\text{model}}(\boldsymbol{x})$

2. implicit density estimation
   - learn a model that can sample from $p_{\text{model}}(\boldsymbol{x})$ w/o explicitly defining it

Trends in generative models

- conventional GM approaches have one of three drawbacks[3]:
  1. require strong assumptions about structure in data
  2. make approximations $\Rightarrow$ suboptimal results
  3. rely on computationally expensive inference procedures (*e.g.* MCMC)

- recent advances
  - train neural nets as powerful function approximators through backprop

  $\Rightarrow$ gives framework for backprop-based function approximators to build GMs

e.g. variational autoencoder: one of the most popular such frameworks
  - assumptions of this model: weak
  - training: fast via _____
  - make an approximation but error is small given high-capacity models

[3] Doersch, C. (2016). Tutorial on variational autoencoders. *arXiv preprint arXiv:1606.05908*

- generative models based on neural nets:

1. autoregressive models (or fully visible belief nets)
   - model $p(\boldsymbol{x})$ as $\prod_{i=1}^{n} p(x_i \,|\, x_1, \ldots, x_{i-1})$

   e.g. PixelRNN, PixelCNN, WaveNet

2. Helmholtz machines
   - model $p(\boldsymbol{x})$ as $\int p(\boldsymbol{z}) p(\boldsymbol{x} \,|\, \boldsymbol{z}) d\boldsymbol{z}$
   - two components: _____ net (encoder) + generative net (decoder)
   - use variational inference to maximize ELBO

   e.g. variational autoencoder (VAE)

3. generative adversarial network (GAN)
   - no explicit density modeling
   - two components: generator + discriminator
   - train models by solving minimax problem

Why study generative models?

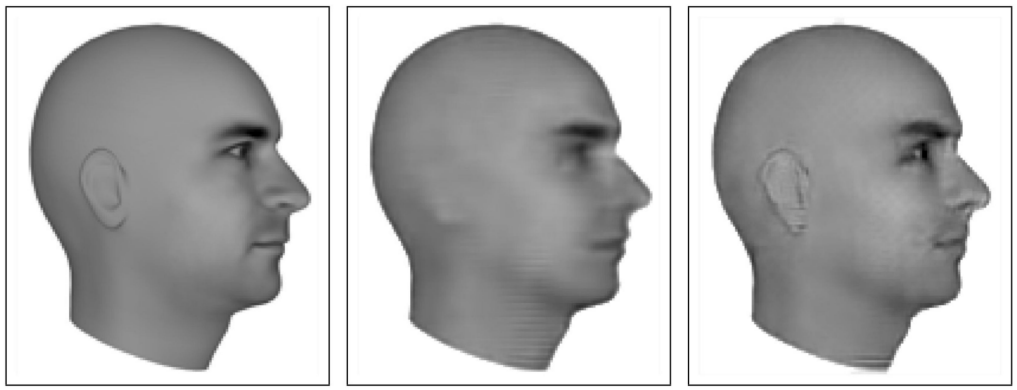
- excellent test of our ability
  - to represent/manipulate high-dim/complicated probability distributions
- can be incorporated into reinforcement learning (RL)
  - simulate possible futures for planning
- can be trained with missing data
  - can provide predictions on inputs that are missing data

$\Rightarrow$ facilitate ____-supervised learning

(source: Chen+, 2018)

- enable machine learning to work with multi-modal outputs

  e.g. a single input $\rightarrow$ many different correction answers



(source: Lotter+, 2015)

$\leftarrow$ video frame prediction

- MSE: averaging

$\Rightarrow$ blurry

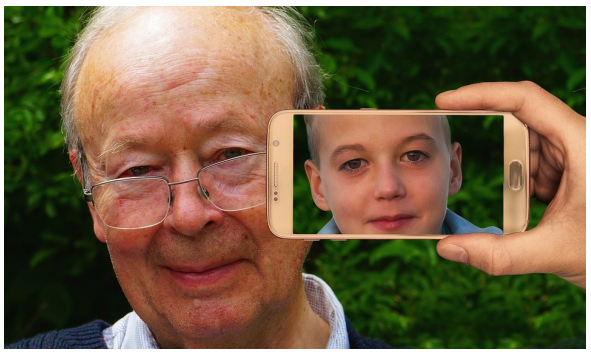
- GAN: sharp

- enable inference of _____ representations

  $\Rightarrow$ can be useful as general features

  e.g. identity-preserving latent representations $\rightarrow$

(source: Antipov+, 2017)



age conditional GAN

- many tasks require realistic generation of samples

  e.g. single image super-resolution, art creation, image-to-image translation

first AI art sold at Christie ($432,000)

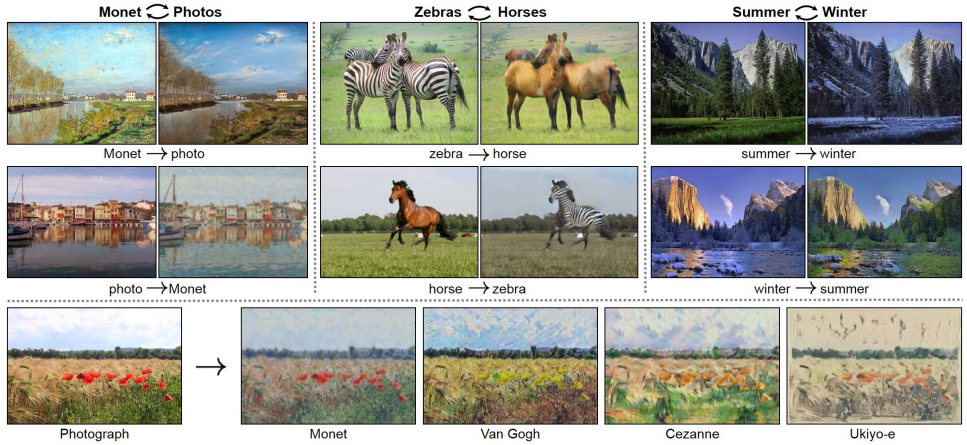
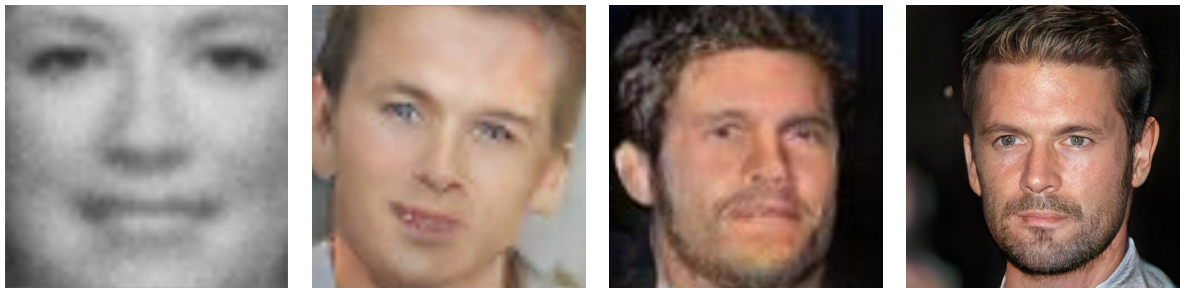


image-to-image translation



2014 2015 2016 2017

3.5 years of progress on faces

(source: Christie's, Zhu+ (2017), Brundage+ (2018))

Outline

Fully visible belief network (FVBN)

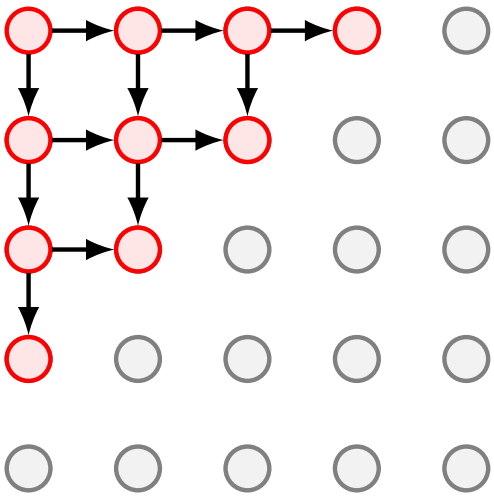
- explicit density model
- how it works:
  1. decompose likelihood of each input $\boldsymbol{x}$ into a product of 1D distributions

$$\underbrace{p(\boldsymbol{x})}_{\substack{\text{likelihood} \\ \text{of input } \boldsymbol{x}}} = \underbrace{\prod_{i=1}^{n} p(x_i \mid x_1, \ldots, x_{i-1})}_{\substack{\text{probability of } i\text{-th feature} \\ \text{given all previous features}}}$$

  2. maximize likelihood of training data
- complex distribution over feature values
  - we express it using a neural net
- main issue:
  - need to define _____ of "previous features"

PixelRNN

- context: image (ICML 2016 best paper[4])
- idea: the same as _____ modeling applied to image
  - generate image pixels starting from corner
  - model dependency on previous pixels using 2D LSTM
- limitation
  - sequential generation $\Rightarrow$ slow



(source: [van den Oord et al., 2016c][5])

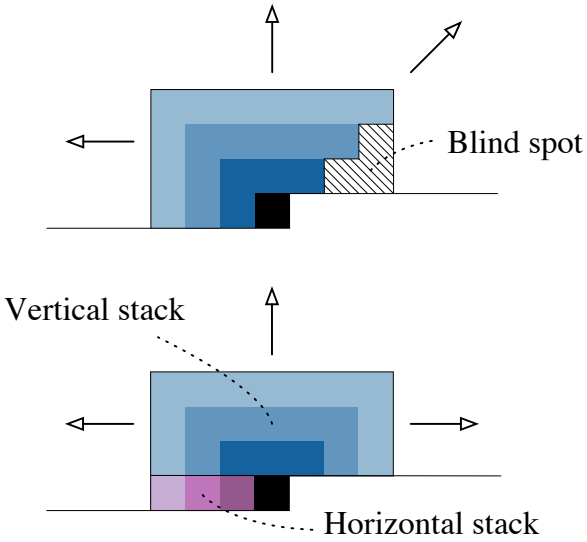
[4] this paper also proposed a simple PixelCNN, which was revised later in their NIPS paper (next page)

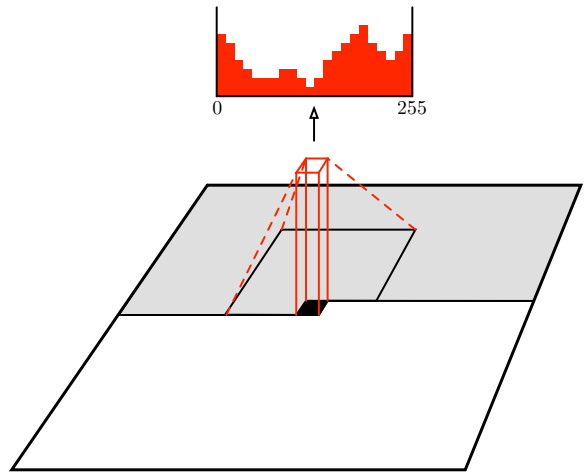
[5] van den Oord, A., Kalchbrenner, N., and Kavukcuoglu, K. (2016c). Pixel recurrent neural networks. *arXiv preprint arXiv:1601.06759*

PixelCNN

- still generate image pixels starting from corner
- what's new
  - use CNN (not LSTM) to model dependency on previous pixels
  - two conv stacks to remove ____ spots
- training
  - maximize likelihood of training images
  - loss: softmax loss at each pixel
  - faster[6] than PixelRNN
- image generation
  - must still proceed sequentially

  $\Rightarrow$ still slow



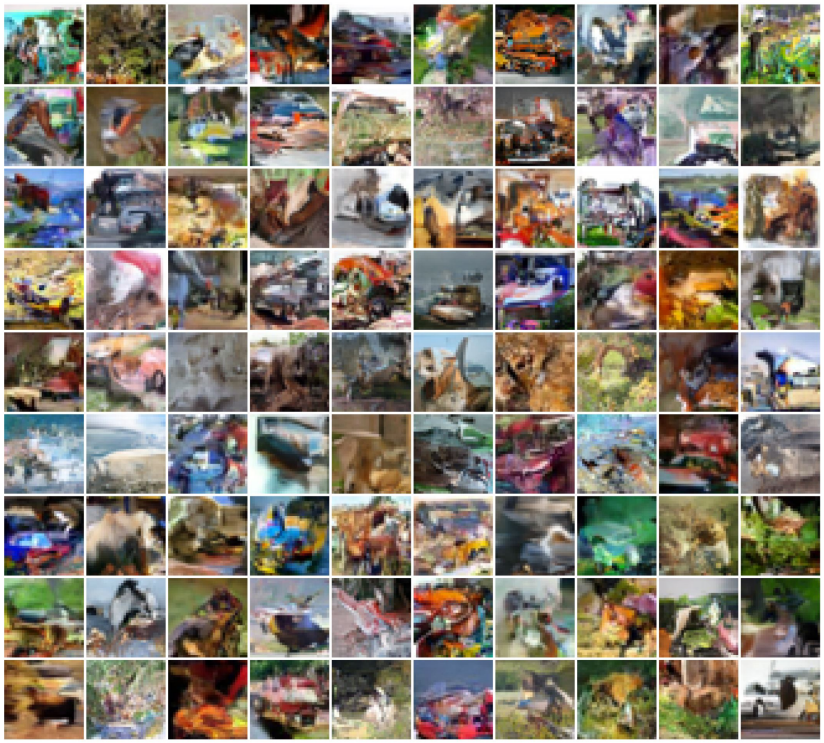


(source: [van den Oord et al., 2016b][7])

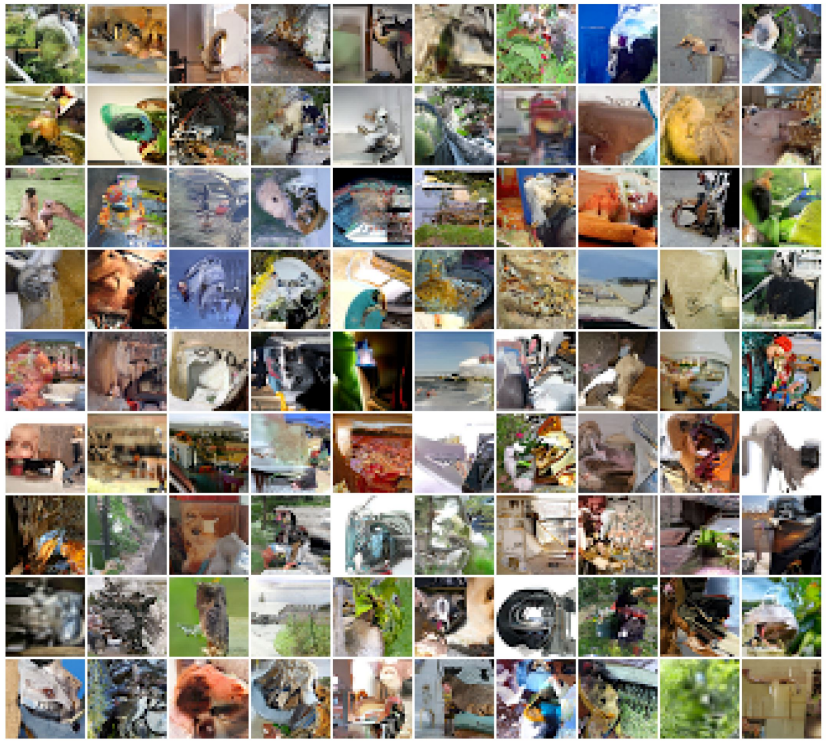
[6] we can parallelize convolutions since context region values are known from training images

[7] van den Oord, A., Kalchbrenner, N., Espeholt, L., Vinyals, O., Graves, A., et al. (2016b). Conditional image generation with pixelcnn decoders. In *Advances in Neural Information Processing Systems*, pages 4790–4798

Examples of generated samples



32×32 CIFAR-10



32×32 ImageNet

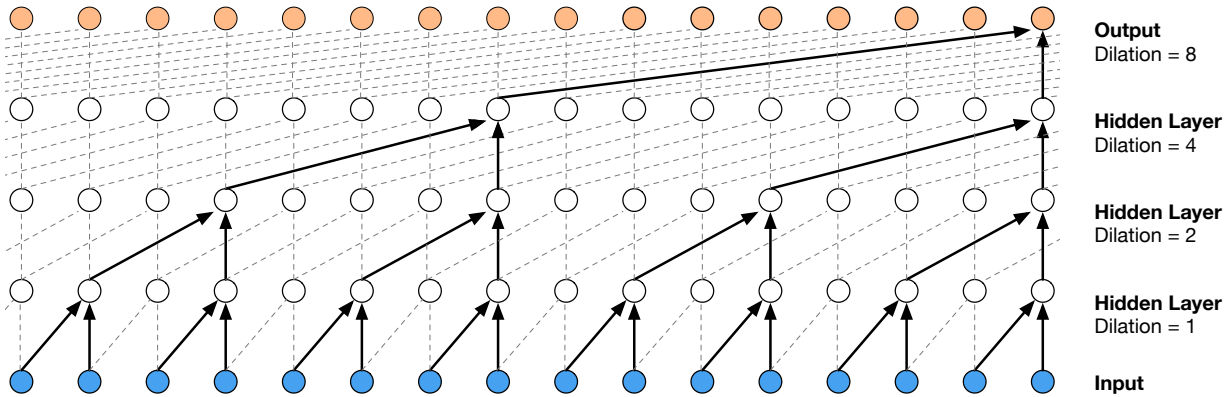
(source: [van den Oord et al., 2016c][8])

[8] van den Oord, A., Kalchbrenner, N., and Kavukcuoglu, K. (2016c). Pixel recurrent neural networks. *arXiv preprint arXiv:1601.06759*

WaveNet

- can generate raw audios (trained on raw audio waveforms)
  - similar to PixelCNN in structure, but much more successful
  - amazing quality but slow generation (2 min to synthesize 1 sec audio)

- architecture
  - use dilated convolution to vastly _____ receptive field



(source: [van den Oord et al., 2016a][9])

  - stack many layers like the above together
  - use classification (not regression) to generate next sample point
  - convert continuous audio $\rightarrow$ 256-level quantized classes

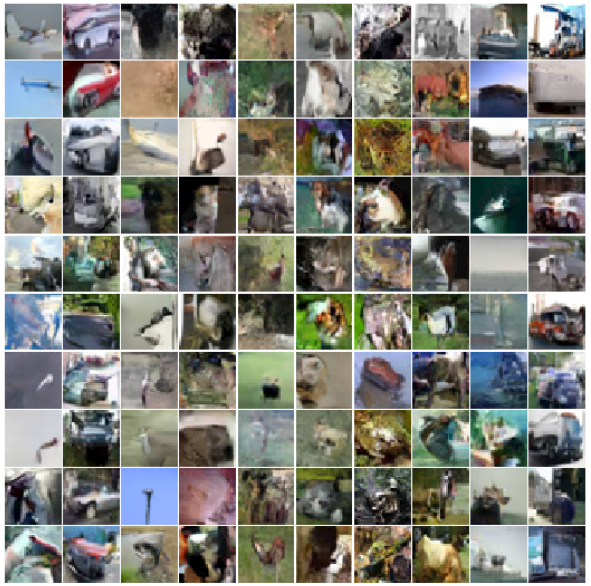
[9] van den Oord, A., Dieleman, S., et al. (2016a). Wavenet: A generative model for raw audio. In *SSW*, page 125

Remarks

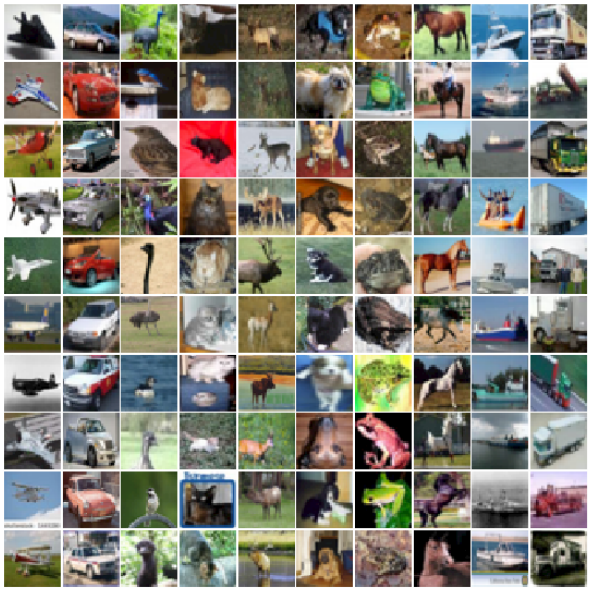
- autoregressive models: pros and cons
  - can explicitly compute $p(\boldsymbol{x})$
  - good samples
  - sequential generation: ____
- to improve performance
  - gated convolution layers
  - short-cut connections
  - discretized logistic loss
  - multi-scale
  - parallelization
  - and many more!
- more information: ▸ Link

(source: cs231n, [Salimans et al., 2017][10])

PixelCNN++ samples from CIFAR-10



real CIFAR-10 images



[10] Salimans, T., Karpathy, A., Chen, X., and Kingma, D. P. (2017). Pixelcnn++: Improving the pixelcnn with discretized logistic mixture likelihood and other modifications. *arXiv preprint arXiv:1701.05517*

Outline

Probabilistic machine learning

- a probabilistic model
  - a joint distribution of hidden variables $\mathbf{z}$ and observed variables $\mathbf{x}$

$$p(\boldsymbol{z}, \boldsymbol{x})$$

  - describes how (a portion of) the world works

- inference about the unknowns:
  - through **posterior** _____

i.e. conditional distribution of the hidden variables given the observations

$$p(\boldsymbol{z} \mid \boldsymbol{x}) = \frac{p(\boldsymbol{z}, \boldsymbol{x})}{p(\boldsymbol{x})} \begin{array}{l} \leftarrow \text{model} \\ \leftarrow \text{data} \end{array}$$

  - the posterior links data and model $\Rightarrow$ used in all downstream analyses

- posterior inference:
  - therefore a central task in probabilistic models

Posterior inference

- refers to
  - (1) computing posterior distribution $p(\boldsymbol{z} \mid \boldsymbol{x})$ or
  - (2) taking expectations computed wrt this distribution

e.g. expectation-maximization (EM) algorithm

- evaluates expectation of complete-data log likelihood wrt posterior distribution of latent variables:

$$\mathcal{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^{\text{old}}) \triangleq \sum_{\mathbf{z}} p(\boldsymbol{z} \mid \boldsymbol{x}, \boldsymbol{\theta}^{\text{old}}) \log p(\boldsymbol{x}, \boldsymbol{z} \mid \boldsymbol{\theta})$$

- for many models of practical interest
  - it is intractable to do (1)/(2) $\Rightarrow$ called "challenge of inference"

- challenge of inference
  - makes it difficult to ____ probabilistic models

Challenge of inference

- general reasons
  - dimensionality of latent space: too high to work with directly
  - posterior: highly complex $\Rightarrow$ expectations are not analytically tractable
- reasons in deep learning
  - interactions between _____ variables (*e.g.* connections between layers)
  - most neural nets with multiple layers of hidden variables
    - have intractable posterior distributions
- a solution: approximate posterior inference

Bayesian view

- we setup the general problem
  - consider a joint density of latent variables $\mathbf{z}$ and observations $\mathbf{x}$

$$p(\boldsymbol{z}, \boldsymbol{x}) = p(\boldsymbol{z})p(\boldsymbol{x} \mid \boldsymbol{z})$$

  - latent variables help govern distribution of data in Bayesian models

- a Bayesian model
  - draws _____ variables from a prior $p(\boldsymbol{z})$, and then
  - relates them to observations through likelihood $p(\boldsymbol{x} \mid \boldsymbol{z})$

- inference in a Bayesian model
  - conditions on data and $\underbrace{\text{computes posterior } p(\boldsymbol{z} \mid \boldsymbol{x})}_{\uparrow \text{ this often requires approximate inference}}$

Approximate posterior inference

1. stochastic (*e.g.* Markov chain Monte Carlo: MCMC)
   - given infinite computational resource, can generate exact results
   - approximation arises from using a finite amount of processor time
   - 😟 sampling methods can be computationally demanding
   - ⇒ their use is often limited to small-scale problems
   - 😟 difficult to know whether _____ samples are being generated

2. deterministic (*e.g.* variational inference: VI)
   - some of which scale well to large applications
   - based on analytical approximations to posterior $p(\boldsymbol{z} \mid \boldsymbol{x})$

   e.g. assume it factorizes or has a _____ form (like Gaussian)

   ⇒ never generate exact results

Outline

Summary

- generative models: density estimation and sample generation
  - explicit density: PixelRNN/CNN, variational autoencoder (VAE)
  - implicit density: generative adversarial network (GAN)
- generative models are versatile and useful for many tasks
  - representation/manipulation of high-dim distributions
  - reinforcement learning, semi-supervised learning
  - multi-modal outputs, inference of latent representations
- approximate inference schemes fall into two classes: stochastic or deterministic
  - their strengths and weaknesses are complementary to each other