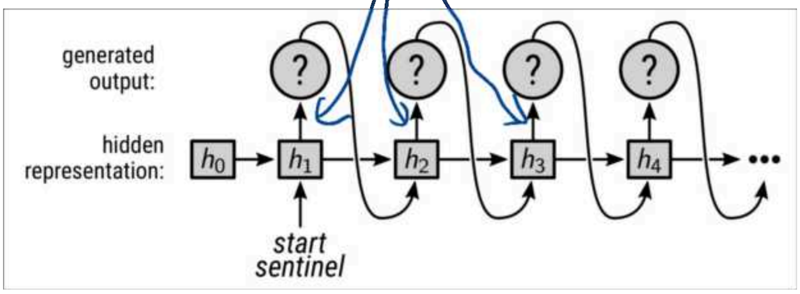
Lecture notes from the course "Data Compression With Deep Probabilistic Models" by Prof. Robert Bamler, University of Tuebingen, 16 June 2021; more course materials at https://robamler.github.io/teaching/compress21/

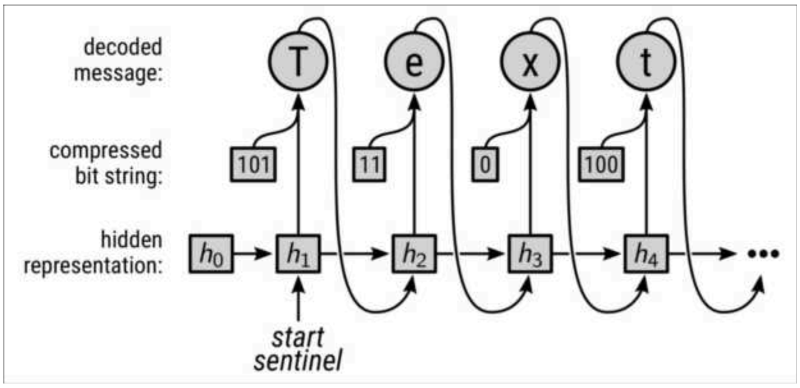
Data Compression with Deep Probabilistic Models

Reminder: Problem 3.2: compression with a learned autoregressive model

parameterizes a probability dist.
⇒ can be used for compression



→ when used for compression (here: decoder side):



autoregressive models: $P_\vartheta(\underline{X}) = P_\vartheta(X_1)\, P_\vartheta(X_2 | X_1)\, P_\vartheta(X_3 | X_1, X_2)$

model parameters (neural network weights)

→ optimize ϑ by minimizing an empirical estimate of cross entropy $H(P_{data}, P_\vartheta)$

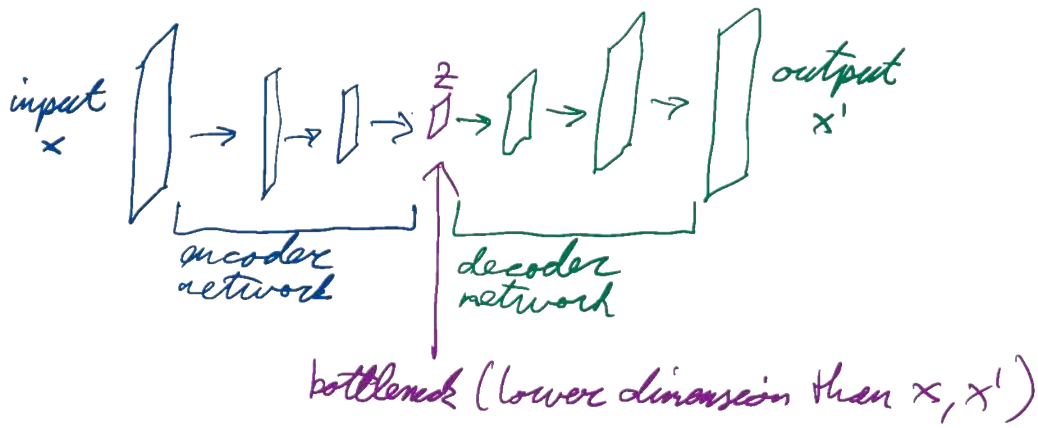
→ can we do the same thing with latent variable models

Deep Latent Variable Models & Scalable Approximate Bayesian Inference

Spoiler: variational autoencoders (VAEs)

→ a form of representation learning

→ often introduced with the following explanation:
"learn to map data to itself while squeezing it through a bottleneck"



use cases of VAEs for compression

↳ lossless compression

1) map x to z & encode z

2) map z to x' & encode residual $x - x'$

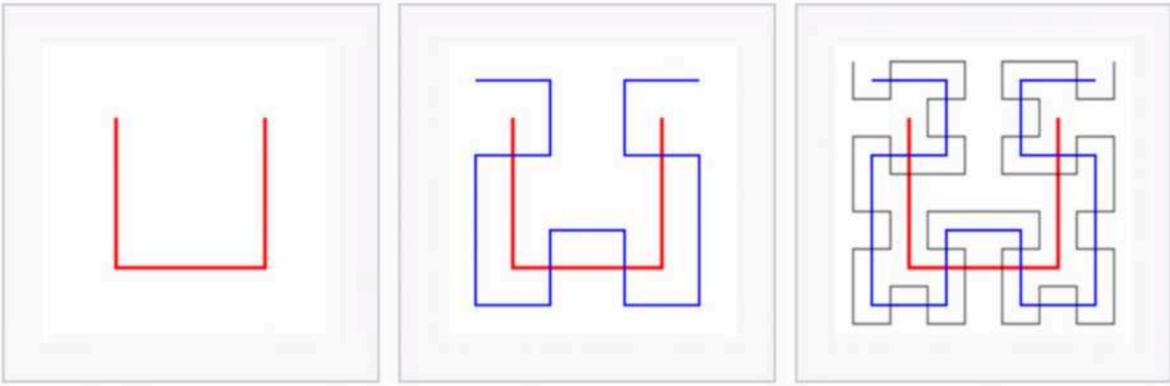
↳ lossy compression: leave out residual

⇒ 3 training objectives

(i) decoder network should reconstruct the data well
(⇒ residual $x' - x$ small / low entropy)

(ii) encoder network decorrelates data
→ need probabilistic model (we want $P(z) = \prod_{i=1}^{k} P(z_i)$)

Note: just squeezing data through a lower-dimensional bottleneck does not in itself imply compression
→ think about information theoretical measures rather than dim.
(iii) keep $H(Z)$ low to enable effective compression



Hilbert curve, first order | Hilbert curves, first and second orders | Hilbert curves, first to third orders

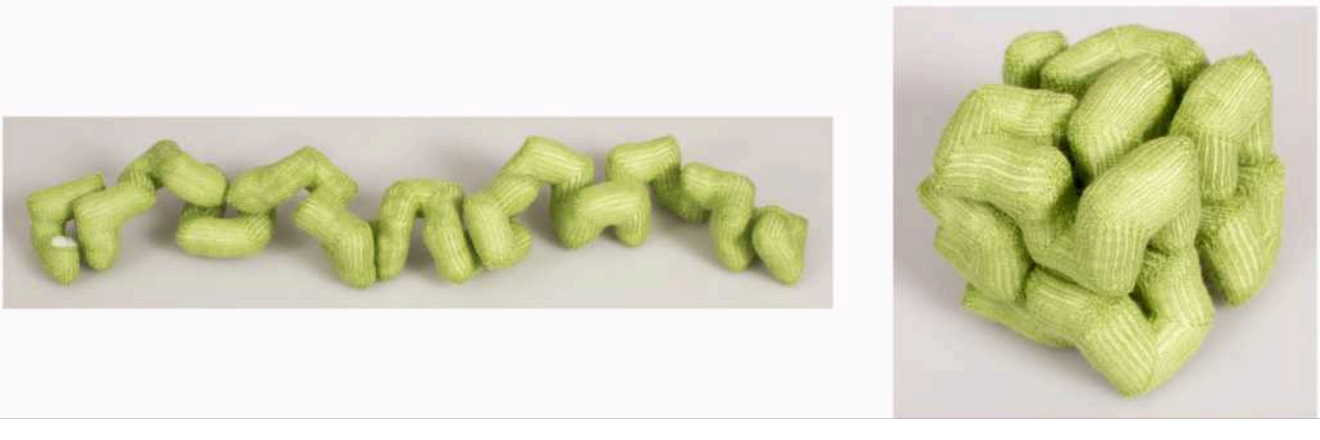
"Hilbert curve" (drawings from Wikipedia)

[ACM Transactions on Graphics (TOG) 35.4 (2016)]

A Compiler for 3D Machine Knitting

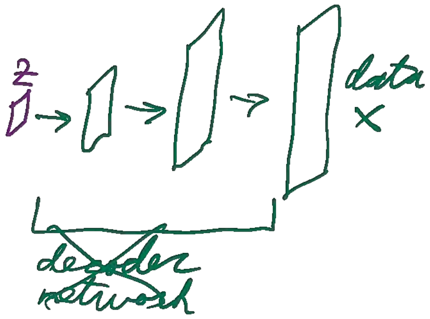
James McCann[1] Lea Albaugh[1] Vidya Narayanan[1] April Grow[1,2]
Wojciech Matusik[3] Jen Mankoff[1,4] Jessica Hodgins[1]

[1]Disney Research [2]UC Santa Cruz [3]Massacusetts Institute of Technology [4]Carnegie Mellon University



Deep Latent Variable Models

· look at decoder network only



→ interpret as a latent variable model:

$$P_\theta(X,Z) = P_\theta(Z)\, P_\theta(X|Z)$$

learned model parameters (e.g. neural network weights)

common example:

lower case: density fct

↳ prior is fully factorized, i.e., $p_\theta(z) = \prod_{i=1}^{k} p_\theta(z_i)$

↳ likelihood: $p_\theta(x|z) = \mathcal{N}(x;\ f_\theta(z),\ \sigma^2 I)$

normal dist. (= gaussian)

neural network

fixed or learned

Goal: minimize $H(P_{data}(X),\ P_\theta(X)) = \mathbb{E}_{P_{data}(X)}\left[\underbrace{-\log P_\theta(X)}_{\text{"evidence"}}\right]$

Problem $P_\theta(X=x) = \int p_\theta(x,z)\, dz$

prohibitively expensive

high dimensional

We want to maximize evidence $P_\theta(X=x)$ when evaluated on data x from the training set.

Recall: bits-back coding

$$R_{net}(x) = -\log P_\vartheta(X=x)$$
$$= -\log P_\vartheta(Z=z) - \log P_\vartheta(X=x \mid Z=z) + \log \underbrace{P_\vartheta(Z=z \mid X=x)}_{\text{posterior}}$$

problem: $P_\vartheta(Z \mid X) = \dfrac{P_\vartheta(X, Z)}{P_\vartheta(X)}$ ← intractable

- Idea: replace posterior with some other dist. $Q_{\lambda_x}(Z)$
  (e.g.: $Q_{\lambda_x}(Z) = \prod_{i=1}^{k} \mathcal{N}(z_i; \mu_i, \sigma_i^2)$)
  μ_i, σ_i^2 make up λ_x

$$\Rightarrow \tilde{R}^{(z)}_{net}(x) = -\log P_\vartheta(X=x, Z=z) + \log Q_{\lambda_x}(Z=z)$$

$$\mathbb{E}_{z \sim Q_{\lambda_x}(Z)}\left[\tilde{R}^{(z)}_{net}(x)\right] \geq R_{net}(x) = \underbrace{-\log P_\vartheta(X=x)}_{\text{we want to minimize this}}$$

equality if $Q_{\lambda_x}(Z) = P_\vartheta(Z \mid X=x)$

Notation & Naming Conventions

- $\log P_\vartheta(X=x)$ is called <u>evidence</u> (we want this to be high)
- $-\mathbb{E}_{z \sim Q_{\lambda_x}(Z)}\left[\tilde{R}^{(z)}_{net}(x)\right] = \mathbb{E}_{z \sim Q_{\lambda_x}(Z)}\left[\log P_\vartheta(X=x, Z=z) - \log Q_{\lambda_x}(Z=z)\right]$
  is called the <u>evidence lower bound</u> (ELBO)
  $\Rightarrow \mathrm{ELBO}(\vartheta, \lambda_x) \leq \underbrace{\log P_\vartheta(X=x)}_{\text{evidence}}$
- parameters λ_x of the distribution $Q_{\lambda_x}(Z)$ are called "<u>variational parameters</u>"
- $Q_{\lambda_x}(Z)$ is called "<u>variational distribution</u>"
- Variational Inference (VI): approximate evidence $\log P_\vartheta(X=x)$ by $\mathrm{ELBO}(\vartheta, \lambda_x^*)$ where
  $$\lambda_x^* := \arg\max_{\lambda_x} \mathrm{ELBO}(\vartheta, \lambda_x)$$

→ observation: this typically leads to a $Q_{\lambda_x^*}(z)$ which is "close" to true posterior $P_\vartheta(Z \mid X=x)$.

(Reviews: Blei et al. 2016, Zhang et al. 2018)

→ we now can approximate $\log P_\vartheta(X=x)$, but we still have to maximize it over ϑ.

→ idea: maximize our approximation $ELBO(\vartheta, \lambda_x^*)$ over ϑ.

Pseudocode:

for t in training_steps:
  sample a minibatch B of training points
  initialize λ_x randomly $\forall x \in B$
  for t' in inner_training_steps:
    perform gradient step for λ_x $\forall x \in B$
  perform gradient step for ϑ on $ELBO(\vartheta, \lambda_x^*)$

nested loop → extremely expensive

VI, finds λ_x^*

Remember:
- model params ϑ are global (i.e., the same for all data points x)
- variational params λ_x parameterize an approximation of $P(Z \mid X=x)$ ⇒ they are local (i.e., different for all data points x)
- we want to maximize $\mathbb{E}_{x \sim P_{data}}[\log P_\vartheta(X=x)]$
  ⇒ we have to sample a new minibatch in each iteration of outer loop
  → invalidates λ_x^* from previous iteration of outer loop.

→ "Variational Expectation Maximization"
(Dempster et al 1977, Beal & Ghahramani 2003)

<u>Final additional trick</u>: learn how to do variational inference

i.e., learn a function $g_\phi : x \mapsto \lambda_x$

set $\lambda_x = g_\phi(x)$ in the ELBO

notation: $Q_\phi(Z|x) = Q_{\lambda_x}(Z)$ with $\lambda_x = g_\phi(x)$

$$\mathrm{ELBO}(\underset{\substack{\uparrow\\ \text{now both are global params}}}{\vartheta, \phi}) = \mathbb{E}_{z \sim Q_\phi(z|x)}\left[\log P_\vartheta(X=x, Z) - \log Q_\phi(Z|x)\right]$$

$$\leq \underbrace{\log P_\vartheta(X=x)}_{\text{evidence}}$$

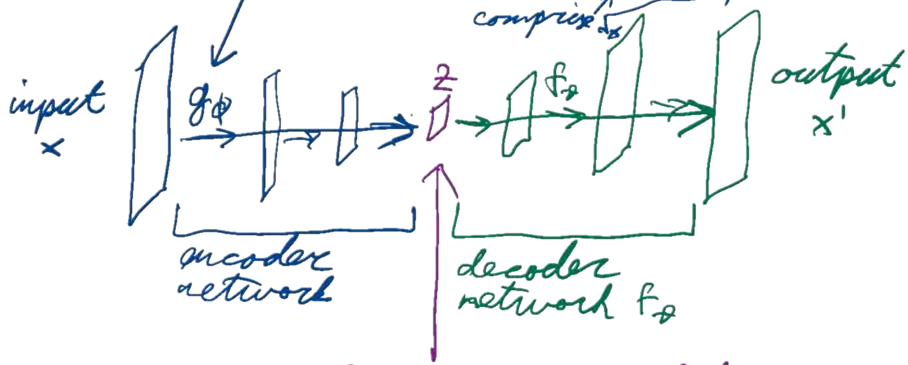
→ maximize $\underbrace{\mathbb{E}_{x \sim P_{data}}\left[\mathrm{ELBO}(\vartheta, \phi)\right]}_{\text{often also just called "ELBO"}}$ over both ϑ, ϕ

→ "Amortized Variational Expectation Maximization"
= "Variational Autoencoders" (VAEs)
(Kingma & Welling 2013)

parameterizes $q_\phi(z|x) = q_{\lambda_x}(z)$ with $\lambda_x = g_\phi(x)$

e.g. $q_{\lambda_x}(z) = \mathcal{N}(z; \mu, \mathrm{diag}(\sigma^2))$

recall: likelihood
$p_\vartheta(x|z) = \mathcal{N}(x; \underset{\substack{\uparrow\\ \text{mean}}}{f_\vartheta(z)}, \sigma^2 I)$



- minimize entropy of this representation → more precisely D_{KL}
- we inject noise here: $z \sim Q_\phi(z|x)$

Interpretations of the ELBO (i.e. the objective function)

$$\mathrm{ELBO}(\theta,\phi) = \mathbb{E}_{z\sim Q_\phi(Z|x)}\left[\log p_\theta(z) + \log p_\theta(x|z) - \log q_\phi(z|x)\right]$$

we maximize this

$$= +\underbrace{\mathbb{E}_{z\sim Q_\phi(Z|x)}\left[\log p_\theta(x|z)\right]} - \underbrace{D_{KL}\left(Q_\phi(Z|x)\,\|\,P_\theta(Z)\right)}$$

maximizing only this part would be maximum likelihood estimation (MLE)
→ it would make $Q_\phi(Z|x)$ collapse to a δ-function peaked at the MLE $= \arg\max_z \log p_\theta(x|z)$

think of this as a regularizer
→ tries to make $Q_\phi(Z|x)$ similar to $P_\theta(Z)$
→ at compression: want to encode z using $P_\theta(Z)$; this term ensures that z's obtained from encoder have high $P_\theta(Z)$

$$= \underbrace{\log P_\theta(X=x)}_{\text{evidence}} - \underbrace{D_{KL}\left(Q_\phi(Z|x)\,\|\,P_\theta(Z|X=x)\right)}$$

evidence
→ maximizing this minimizes the inf. content of x under our model P_θ, i.e. the theoretical lower bound of the bitrate

minimizing this makes the variational dist. Q_ϕ similar to the true posterior
⇒ Q_ϕ can be called the "approximate posterior"

→ Goal: maximise ELBO over θ & ϕ

- issue: $\mathrm{ELBO}(\theta,\phi) = \mathbb{E}_{z\sim Q_\phi(Z|x)}[\ldots]$

distribution from which we have to sample depends on ϕ, by which we want to differentiate
→ see Problem set
(reparameterization grad: Kingma & Welling 2013
REINFORCE-gradients: Ranganath et al. 2014)

Why all this fuss?

ongoing research on VI & related methods may be applicable to compression - or it may not be
⇒ look into that literature & try out if it improves compression methods

Examples: • lots of research on tighter bounds of the evidence (tighter than the standard ELBO):

→ e.g. importance weighted VI, recently applied to compression by Theis & Ho 2021

• iterative amortized inference

↳ Marino et al 2018

↳ Campos et al 2019

• other approximate Bayesian inference methods (alternatives to VI) exist (in particular: Markov Chain Monte Carlo = MCMC)

→ nontrivial how to use these for compression (pioneering work: Havasi et al., 2018)

References

[Beal and Ghahramani, 2003] Beal, M. J. and Ghahramani, Z. (2003). The variational bayesian em algorithm for incomplete data: with application to scoring graphical model structures. *Bayesian statistics*, 7(453-464):210.

[Blei et al., 2017] Blei, D. M., Kucukelbir, A., and McAuliffe, J. D. (2017). Variational inference: A review for statisticians. *Journal of the American statistical Association*, 112(518).

[Campos et al., 2019] Campos, J., Meierhans, S., Djelouah, A., and Schroers, C. (2019). Content adaptive optimization for neural image compression. *arXiv preprint arXiv:1906.01223*.

[Dempster et al., 1977] Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1).

[Havasi et al., 2018] Havasi, M., Peharz, R., and Hernández-Lobato, J. M. (2018). Minimal random code learning: Getting bits back from compressed model parameters. *arXiv preprint arXiv:1810.00440*.

[Kingma and Welling, 2014] Kingma, D. P. and Welling, M. (2014). Auto-encoding variational Bayes. In *International Conference on Learning Representations*.

[Marino et al., 2018] Marino, J., Yue, Y., and Mandt, S. (2018). Iterative amortized inference. In *International Conference on Machine Learning*.

[Ranganath et al., 2014] Ranganath, R., Gerrish, S., and Blei, D. (2014). Black box variational inference. In *Artificial intelligence and statistics*, pages 814–822.

[Theis and Ho, 2021] Theis, L. and Ho, J. (2021). Importance weighted compression. In *Neural Compression: From Information Theory to Applications–Workshop@ ICLR 2021*.

[Zhang et al., 2018] Zhang, C., Bütepage, J., Kjellström, H., and Mandt, S. (2018). Advances in variational inference. *IEEE transactions on pattern analysis and machine intelligence*, 41(8).