RPTU

Deep Learning for Natural Language Processing

Jun.-Prof. Sophie Fellenz

20.01.2025

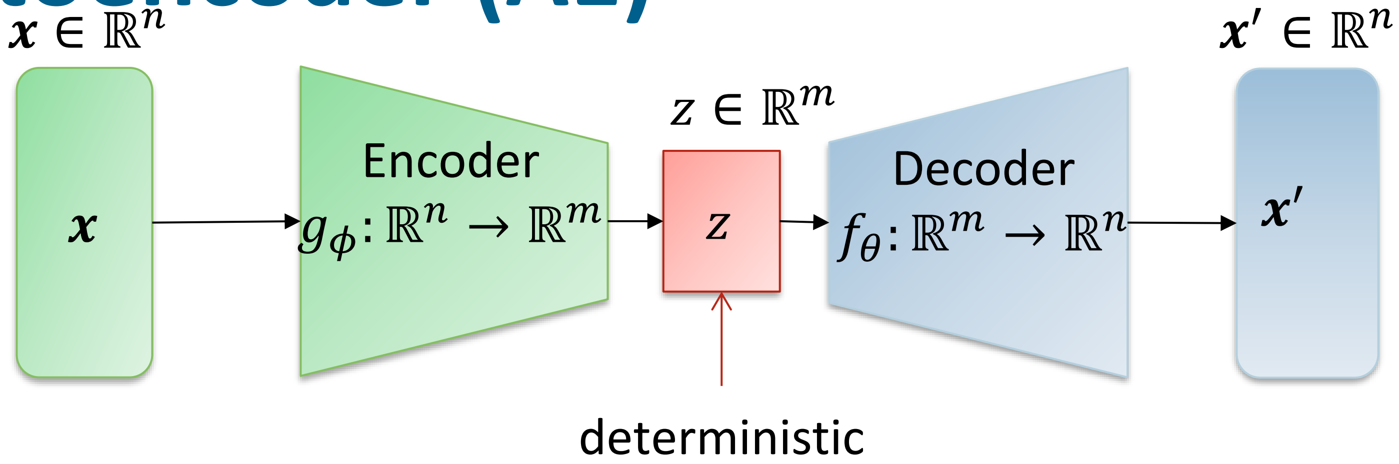
Week 11 - Neural Topic Model

Outline

- Autoencoder
- Variational Auto Encoder
- Topic model
- Variational Auto Encoder based topic model

Autoencoder

Autoencoder (AE)



- **Encoder**: map input x from an n-dimensional space into a smaller m-dimensional space
- **Decoder**: re-map compressed representation from the m-dimensional space back into original n-dimensional input data space.

Autoencoder (AE)

- encoder function: $z = g_\phi(\boldsymbol{x})$
- decoder function: $\boldsymbol{x}' = f_\theta(z)$
- **objective** is to minimize the sum of squared differences between every input and output.

- Reconstruction error

$$\mathcal{L}_{AE}(\phi, \theta, \boldsymbol{x}) = \frac{1}{N} \sum_{i=1}^{N} \left(x^{(i)} - f_\theta \left(g_\phi \left(\boldsymbol{x}^{(i)} \right) \right) \right)^2$$

- Objective

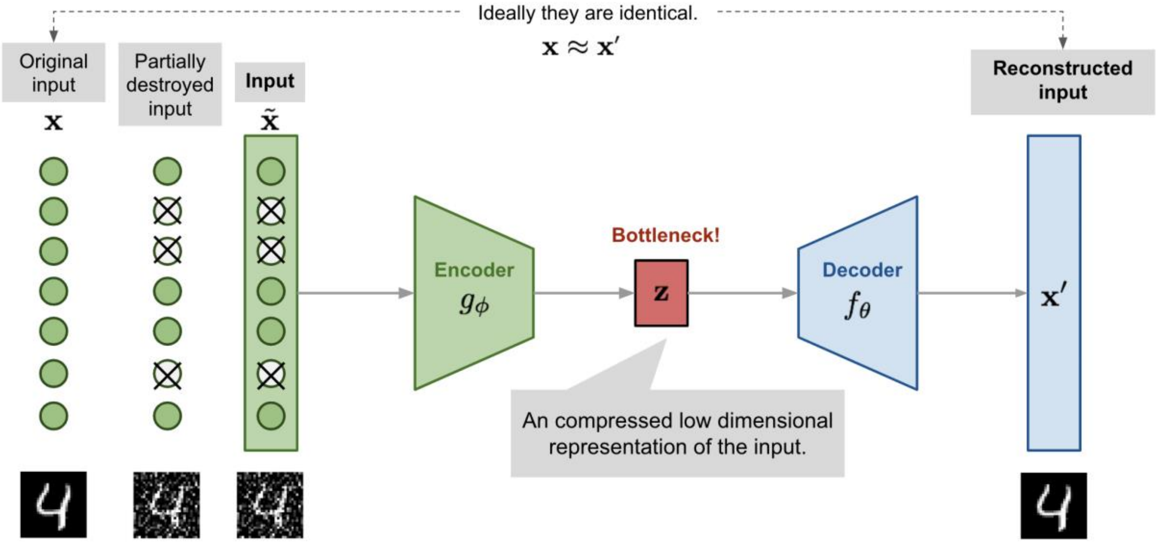
$$\text{Find} \arg\min_{\phi,\theta} \mathcal{L}_{AE}(\phi, \theta, \boldsymbol{x})$$

Autoencoder (AE)

- A deterministic AE compresses data
- lossy (here also: blurry due to $\mathcal{L}_{AE}$= MSE)
- unsupervised
- data-specific
- memorizes data but has no concept of the data

- **Question**: How can we augment a DAE to learn a model of the data?

Denoising Autoencoder (DAE)



Source: https://lilianweng.github.io/posts/2018-08-12-vae/

Variational Autoencoder

Variational Autoencoder

- Dataset $X = \{x^{(i)}\}_{i=1}^{N}$ is generated by some random variable x
- x is influenced by some latent (hidden) random variable z
- $p_\theta(x)$ is the data distribution

Generative process:

1. Draw $z^{(i)} \sim p_\theta(z)$ (prior)
2. Draw $x^{(i)} \sim p_\theta(x|z)$ (likelihood)

$$\rightarrow p_\theta(x, z) = p_\theta(z) p_\theta(x|z)$$

Model

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

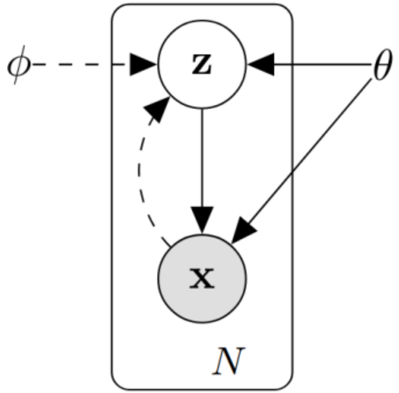
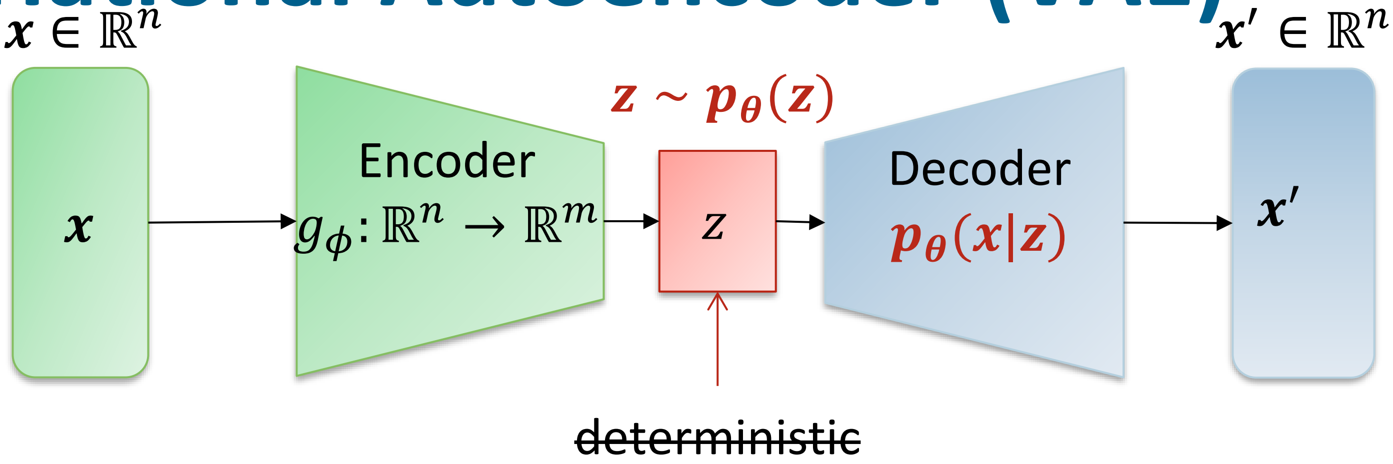


Figure: Kingma & Welling, 2014

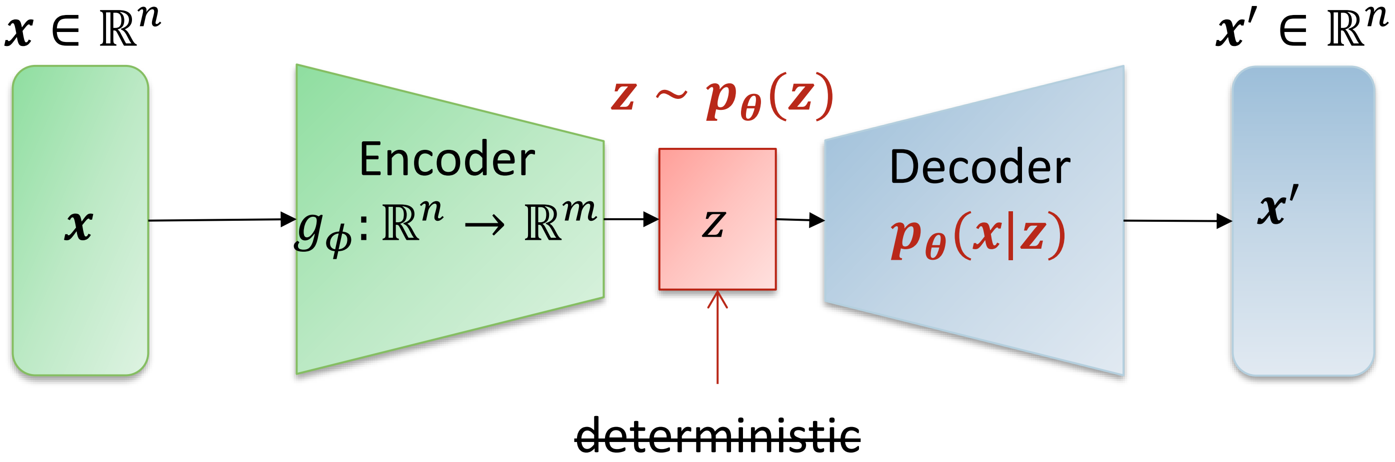
Variational Autoencoder (VAE)



- **Decoder**: parameterizes generative probability distribution

Now we have a probabilistic model with latent variable z, what is the **objective** of this model?

Variational Autoencoder (VAE)

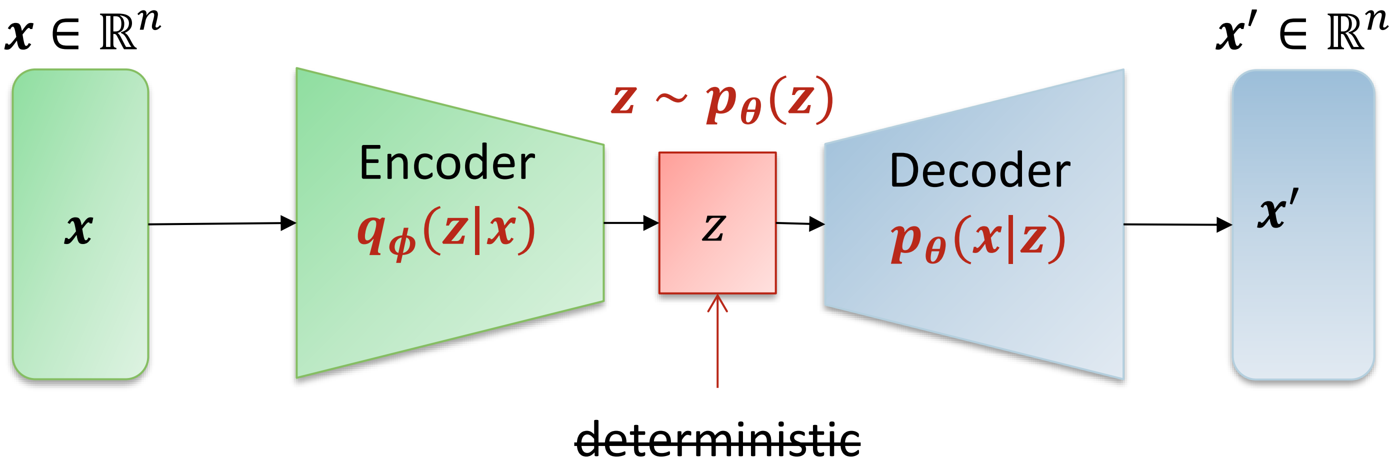


- **Encoder**: For the encoder we need the posterior distribution

$$p(\mathbf{z} \mid \mathbf{x}) = \frac{p(\mathbf{z},\mathbf{x})}{p(\mathbf{x})} = \frac{p(\mathbf{x}|\mathbf{z})p(\mathbf{z})}{\int p(\mathbf{x}|\mathbf{z})p(\mathbf{z})dz}$$

Problem: Computing this quantity is intractable

Variational Autoencoder (VAE)

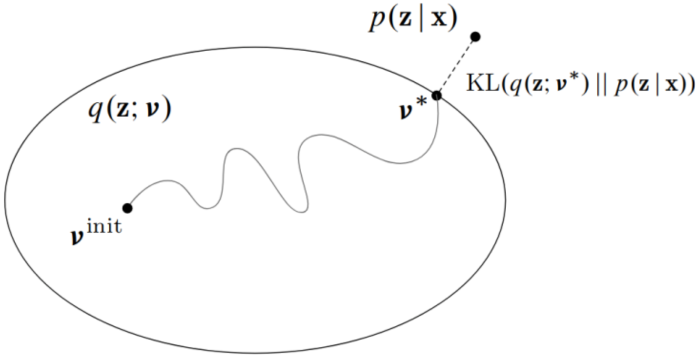


Solution: approximate posterior $q_\phi(\boldsymbol{z} \mid \boldsymbol{x})$

Variational Inference

- Consider a generative model $p_\theta(x|z)$ and prior $p(z)$
  - Joint distribution: $p_\theta(x, z) = p_\theta(x|z)p(z)$
- Assume variational distribution $q_\phi(z|x)$
- Objective: Minimize the difference between the distributions q and p:

$$D_{KL}(q_\phi(z|x)||p_\theta(z|x))$$



KL Divergence

- For two probability distributions the KL divergence is given by:

$$D_{KL}\left(q_\phi(\mathbf{z} \mid \mathbf{x}) \parallel p_\theta(\mathbf{z} \mid \mathbf{x})\right) = \int q_\phi(\mathbf{z} \mid \mathbf{x}) \log\left(\frac{q_\phi(\mathbf{z} \mid \mathbf{x})}{p_\theta(\mathbf{z} \mid \mathbf{x})}\right) d\mathbf{z}$$

- By Jensen‘s inequality, the KL divergence is always non-negative

$$D_{KL}(p \parallel q) \geq 0$$

Other names: information gain, relative entropy

KL Divergence

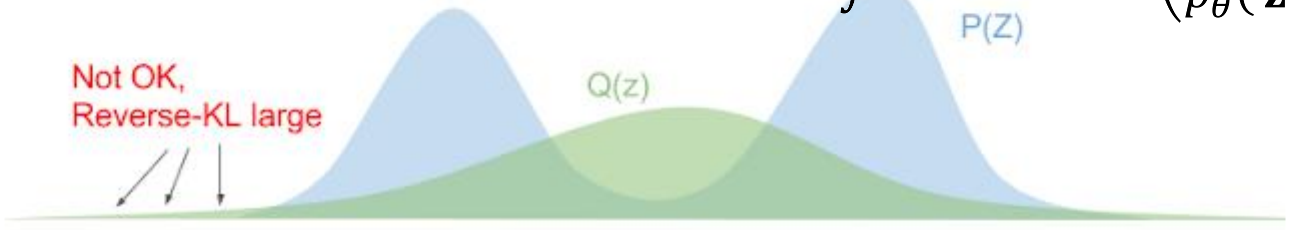
$$D_{KL}\left(p_\phi(\mathbf{z} \mid \mathbf{x}) \parallel q_\theta(\mathbf{z} \mid \mathbf{x})\right)$$
$$= \int p_\phi(\mathbf{z} \mid \mathbf{x}) \log\left(\frac{p_\phi(\mathbf{z} \mid \mathbf{x})}{q_\theta(\mathbf{z} \mid \mathbf{x})}\right) d\mathbf{z}$$

Forward-KL: $D_{KL}\left(p_\theta(\mathbf{z} \mid \mathbf{x}) || q_\phi(\mathbf{z} \mid \mathbf{x})\right)$

Source: https://blog.evjang.com/2016/08/variational-bayes.html

KL Divergence

$$D_{KL}\left(q_\phi(\mathbf{z} \mid \mathbf{x}) \parallel p_\theta(\mathbf{z} \mid \mathbf{x})\right)$$
$$= \int q_\phi(\mathbf{z} \mid \mathbf{x}) \log\left(\frac{q_\phi(\mathbf{z} \mid \mathbf{x})}{p_\theta(\mathbf{z} \mid \mathbf{x})}\right) d\mathbf{z}$$



Reverse-KL: $D_{KL}\left(q_\theta(\mathbf{z} \mid \mathbf{x}) || p_\phi(\mathbf{z} \mid \mathbf{x})\right)$

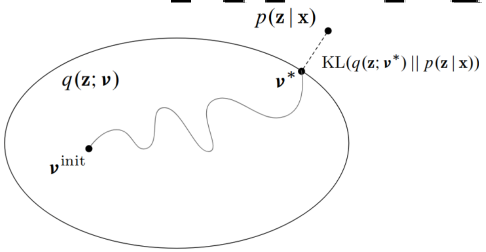
Source: https://blog.evjang.com/2016/08/variational-bayes.html

KL Divergence

$$D_{KL}\left(q_\phi(\mathbf{z} \mid \mathbf{x}) \parallel p_\theta(\mathbf{z} \mid \mathbf{x})\right)$$
$$= \int q_\phi(\mathbf{z} \mid \mathbf{x}) \log\left(\frac{q_\phi(\mathbf{z} \mid \mathbf{x})}{p_\theta(\mathbf{z} \mid \mathbf{x})}\right) d\mathbf{z}$$

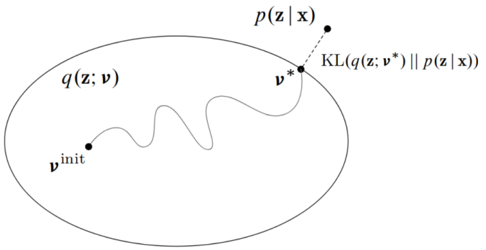
minimizing forward-KL "stretches" your variational distribution Q(Z) to cover over the entire P(Z) like a tarp, while minimizing reverse-KL "squeezes" the Q(Z) under P(Z).

Variational Inference



$$
\begin{aligned}
&D_{KL}(q_\phi(z|x)||p_\theta(z|x)) \\
&= \mathbb{E}_{q_\phi(Z|x)}(\log q_\phi(z|x) - \log p_\theta(z|x)) \\
&= \mathbb{E}_{q_\phi(Z|x)}(\log q_\phi(z|x) - \log \frac{p_\theta(x|Z)p_\theta(z)}{p_\theta(x)}) \\
&= \underbrace{\mathbb{E}_{q_\phi(Z|x)}\big(\log q_\phi(z|x) - \log p_\theta(x|z) - \log p_\theta(z)\big)}_{-ELBO} + \overbrace{\log p_\theta(x)}^{\text{constant w.r.t. z}}
\end{aligned}
$$

Variational Inference



$$
\begin{aligned}
&D_{KL}(q_\phi(z|x)||p_\theta(z|x)) \\
&= \mathbb{E}_{q_\phi(z|x)}(\log q_\phi(z|x) - \log p_\theta(z|x)) \\
&= \mathbb{E}_{q_\phi(z|x)}(\log q_\phi(z|x) - \log \frac{p_\theta(x|z)p_\theta(z)}{p_\theta(x)}) \\
&= \underbrace{\mathbb{E}_{q_\phi(z|x)}\big(\log q_\phi(z|x) - \log p_\theta(x|z) - \log p_\theta(z)\big)}_{-ELBO} + \overbrace{\log p_\theta(x)}^{\text{constant w.r.t. z}}
\end{aligned}
$$

Rearranging terms shows we are doing Maximum likelihood estimation:

$$\log p_\theta(x) = ELBO + \underbrace{D_{KL}(q_\phi(z|x)||p_\theta(z|x))}_{\text{>=0 by definition}}$$

Variational Inference

$$= \underbrace{\mathbb{E}_{q_\phi(z|x)}\big(\log q_\phi(z|x) - \log p_\theta(x|z) - \log p_\theta(z)\big)}_{-ELBO} + \overbrace{\log p_\theta(x)}^{\text{constant w.r.t. z}}$$

$$\text{- ELBO} = \mathbb{E}_{q_\phi(z|x)}\big(\log q_\phi(z|x) - \log p_\theta(x|z) - \log p_\theta(z)\big)$$

$$= \mathbb{E}_{q_\phi(z|x)}\big(\log q_\phi(z|x) - \log p_\theta(z)\big) - \mathbb{E}_{q_\phi(z|x)}(\log p_\theta(x|z))$$

$$= \underbrace{\mathbb{E}_{q_\phi(z|x)}\left(\frac{\log q_\phi(z|x)}{\log p(z)}\right)}_{= KL[q_\phi(z|x)||p(z)]} - \mathbb{E}_{q_\phi(z|x)}(\log p_\theta(x|z))$$

Variational Inference

Maximize the variational lower bound:

$$\mathcal{L}(\theta, \phi; x) = \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)] - D_{KL}(q_\phi(z|x)||p(z))$$

E-step: maximize $\mathcal{L}$ w.r.t. ϕ, with θ fixed

$$\max_{\phi} \mathcal{L}(\theta, \phi; x)$$

M-step: maximize $\mathcal{L}$ w.r.t. θ, with ϕ fixed

$$\max_{\theta} \mathcal{L}(\theta, \phi; x)$$

Variational Inference

Maximize the variational lower bound:

$$\mathcal{L}(\theta, \phi; x) \underbrace{\boxed{= \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)]}}_{\text{Reconstruction loss}} - D_{KL}(q_\phi(z|x)||p(z))$$

Gaussian case: $\log p_\theta(x|z)$

$$= \log\left[\ (2\pi)^{-\frac{k}{2}} \det(\Sigma)^{-\frac{1}{2}} \exp\left[-\frac{1}{2}(\mu - \mathrm{x})^{\mathrm{T}} \overset{I}{\Sigma^{-1}} (\overset{Wz}{\mu} - \mathrm{x})\right]\right]$$

$$= \underbrace{\log\left[\ (2\pi)^{-\frac{k}{2}} \det(\Sigma)^{-\frac{1}{2}}\right]}_{\text{Constant with respect to } \mu} + \underbrace{\left[-\frac{1}{2}(\mu - \mathrm{x})^{\mathrm{T}}(\mu - \mathrm{x})\right]}_{\text{Standard L2 Autoencoder loss!}}$$

Variational Inference

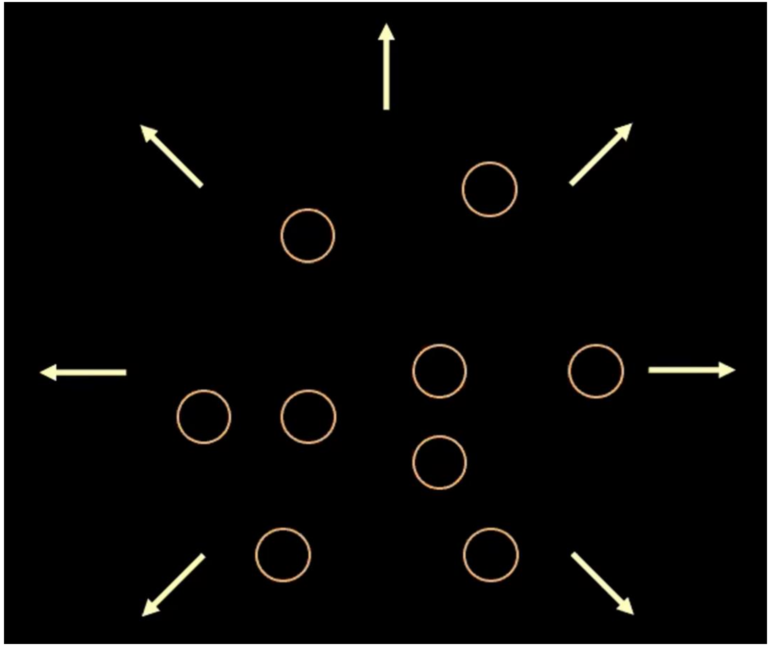
Maximize the variational lower bound:

$$\mathcal{L}(\theta, \phi; x) = \underbrace{\boxed{\mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)]}}_{\text{Reconstruction loss}} - D_{KL}(q_\phi(z|x)||p(z))$$

Categorical case: $\log p_\theta(x|z) = \log[\mu^x] = x \log \mu$

$softmax(Wz)$

Variational Inference



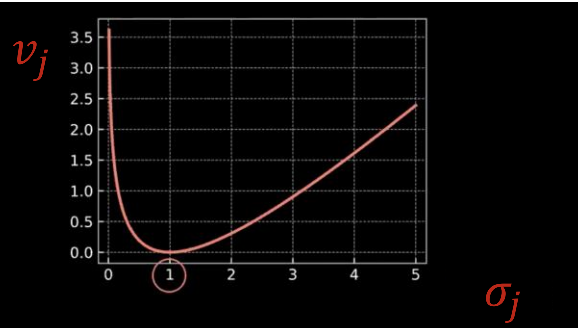
Source: https://atcold.github.io/NYU-DLSP20/en/week08/08-3/

Variational Inference

Maximize the variational lower bound:

$$\mathcal{L}(\theta,\phi;x) = \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)] - \underbrace{\boxed{D_{KL}(q_\phi(z|x)||p(z))}}_{\text{Regularization}}$$

Gaussian case:

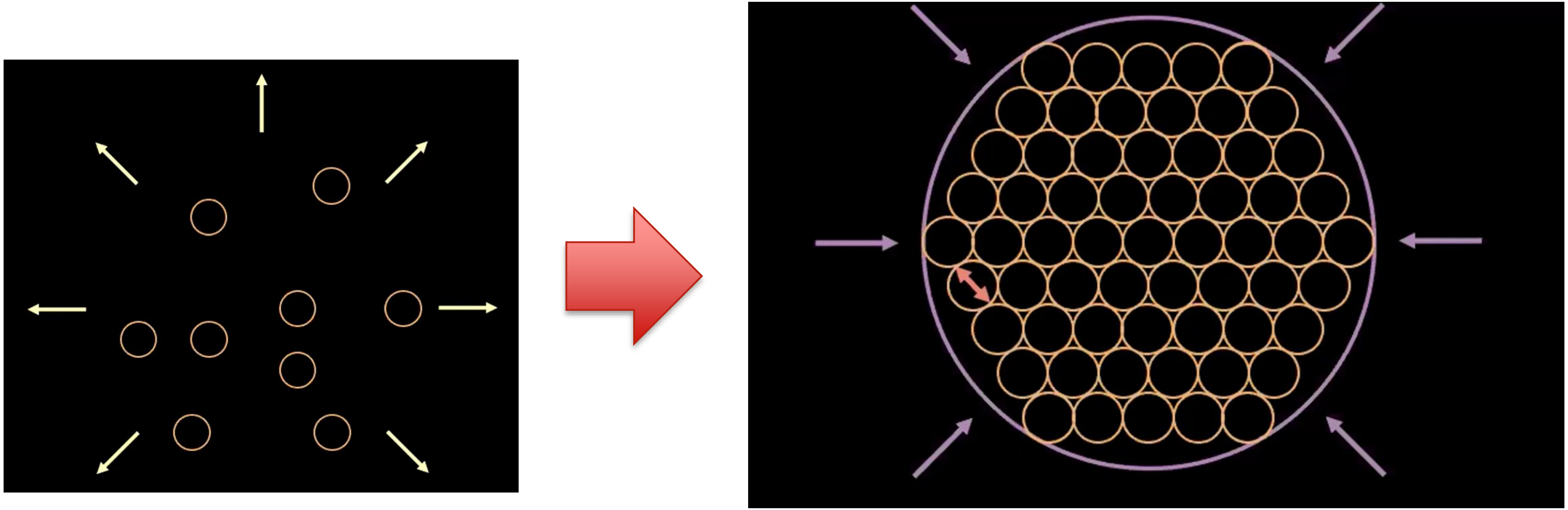


$$D_{KL}(q_\phi(z|x)||p(z)) = D_{KL}(\boldsymbol{N}(\boldsymbol{\mu},\boldsymbol{\sigma})||\boldsymbol{N}(\boldsymbol{0},\boldsymbol{I}))$$

$$= \frac{\mathbf{1}}{\mathbf{2}}\sum_{j=1}^{J}\left(\underbrace{-\mathbf{1} - \boldsymbol{log}\left((\boldsymbol{\sigma}_j)^{\mathbf{2}}\right) + (\boldsymbol{\sigma}_j)^{\mathbf{2}}}_{v_i} + (\boldsymbol{\mu}_j)^{\mathbf{2}}\right)$$

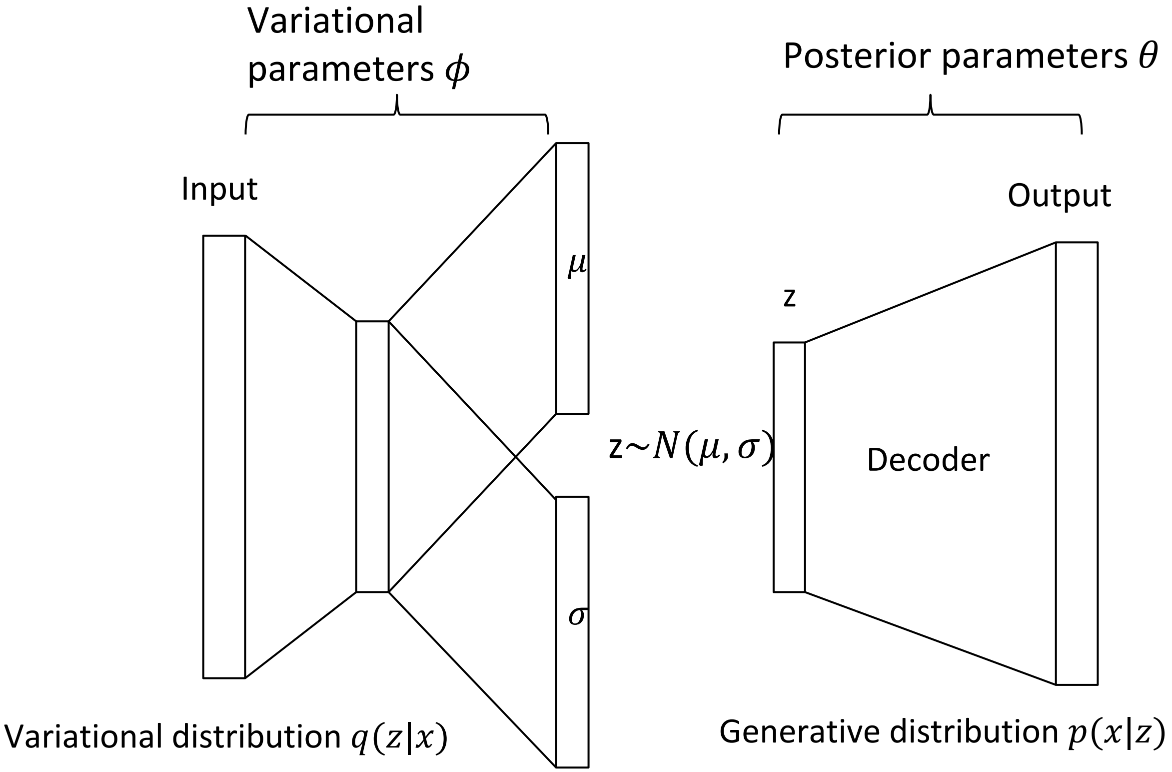
Source: https://atcold.github.io/NYU-DLSP20/en/week08/08-3/

Variational Inference



Source: https://atcold.github.io/NYU-DLSP20/en/week08/08-3/

Gaussian VAEs



$$L(\theta, \phi; x) = \underbrace{\mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)]}_{\text{reconstruction performance}} - \underbrace{KL\big(q_\phi(z|x) || p(z)\big)}_{\text{regularizer}}$$

Synonyms in the Literature

Posterior distribution -> Inference model

- Variational approximation
- Recognition model
- Inference network (if parameterized as neural networks)
- Recognition network (if parameterized as neural networks)
- (Probabilistic) encoder

Synonyms in the Literature

„The Model“ (prior + conditional, or joint) -> generative model

- The (data) likelihood model
- Generative network (if parameterized as neural networks)
- Generator
- (Probabilistic) decoder

Variational Autoencoders (VAEs)

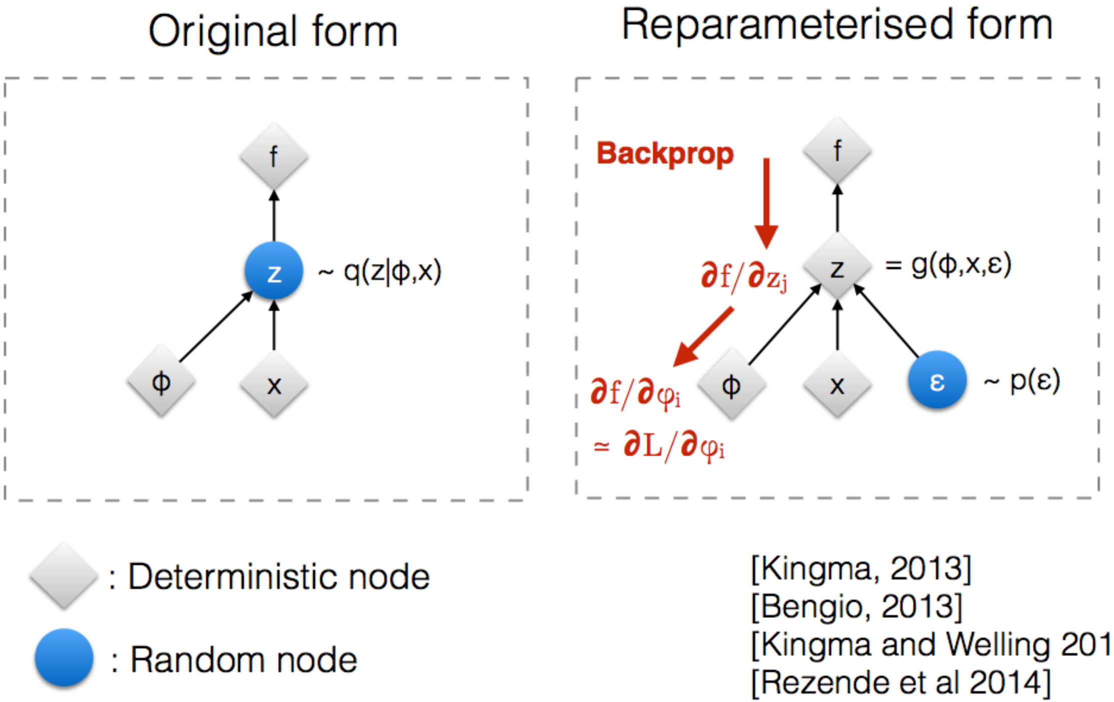
- Variational lower bound
  $L(\theta, \phi; x) = \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)] - KL\big(q_\phi(z|x)||p(z)\big)$
  $= \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x, z)] + H(q_\phi(z|x))$
- Optimize $L(\theta, \phi; x) wrt. \theta$ of $p_\theta(x|z)$

- Optimize $L(\theta, \phi; x)$ wrt. ϕ of $q_\phi(z|x)$

$$\nabla_\phi L(\theta, \phi; x) = \cdots + \nabla_\phi \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)] + \cdots$$

Use reparameterization trick to reduce variance

Reparameterization trick



[Kingma, 2013]
[Bengio, 2013]
[Kingma and Welling 2014]
[Rezende et al 2014]

Reparameterized Gradient

- Optimize $L(\theta, \phi; x)$ wrt. ϕ of $q_\phi(z \mid x)$

This follows from the fact that
$\nabla_\phi q_\phi(z|x) = q(z|x)\nabla_\phi \log q_\phi(z|x)$

Also called **score function gradient** or **REINFORCE** gradient

- ELBO: $L(\theta, \phi; x) = \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x, z)] + H\big(q_\phi(z \mid x)\big)$
  - gradient estimate with log-derivative trick:
    $\nabla_\phi \mathbb{E}_{q_\phi}[\log p_\theta(x, z)] = \mathbb{E}_{q_\phi}\left[\log p_\theta(x, z) \nabla_\phi \log q_\phi\right]$

  - High variance: $\nabla_\phi \mathbb{E}_{q_\phi}[\log p_\theta] \approx \mathbb{E}_{z_i \sim q_\phi}\left[\log p_\theta(x, z_i) \nabla_\phi q_\phi(z_i \mid x)\right]$
    - The scale factor $\log p_\theta(x, z_i)$ can have arbitrarily large magnitude.

Reparameterized Gradient

- Gradient estimate with reparameterization trick

$$z \sim q_\phi(z \mid x) \Leftrightarrow z = g_\phi(\epsilon, x), \epsilon \sim p(\epsilon)$$

$$\nabla_\phi \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x, z)] = \mathbb{E}_{\epsilon \sim p(\epsilon)}\left[\nabla_\phi \log p_\theta\left(x, z_\phi(\epsilon)\right)\right]$$

- (Empirically) lower variance of the gradient estimate

- E.g., $t \sim N(\mu(x), L(x)L(x)^T) \Leftrightarrow \epsilon \sim N(0, \mathrm{I}), z = \mu(x) + L(x)\epsilon$

Reparameterized Gradient

- Score function gradient is broadly applicable to nearly any variational distribution, regardless of whether z is discrete or continuous
- BUT: high variance estimates
- Lower variance with reparameterization trick
- BUT: only possible if differentiable reparameterization function available
- Easy for Gaussian, but e.g. Dirichlet is not as straight-forward (some solutions exist)

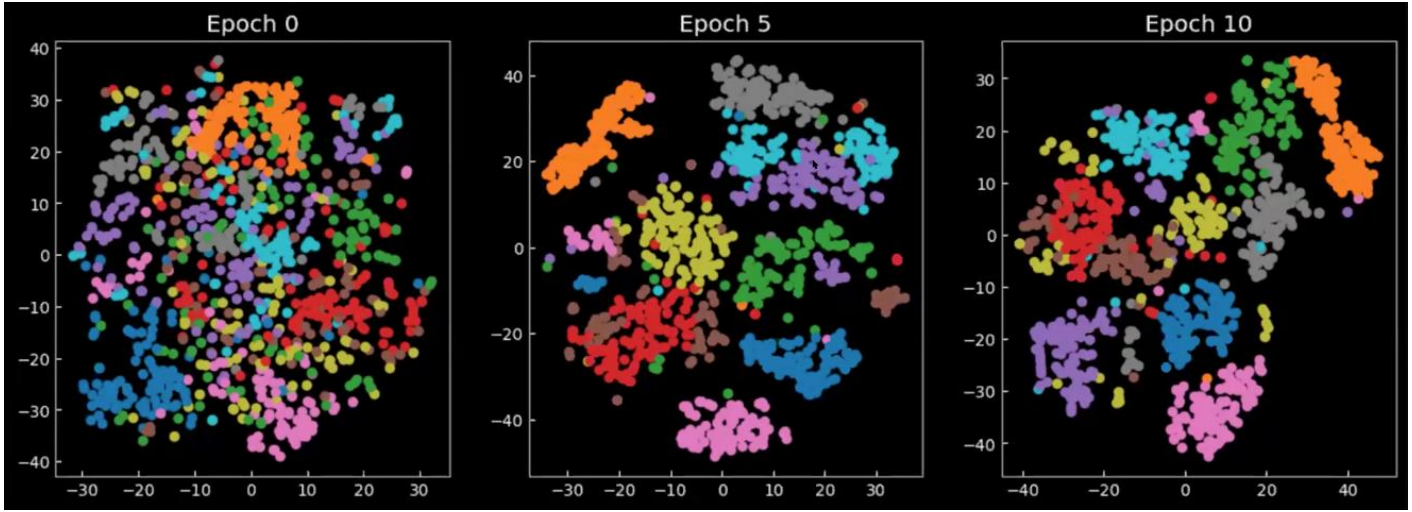
VAEs: Algorithm

Algorithm 1 Minibatch version of the Auto-Encoding VB (AEVB) algorithm. Either of the two SGVB estimators in section 2.3 can be used. We use settings $M = 100$ and $L = 1$ in experiments.

$\boldsymbol{\theta}, \boldsymbol{\phi} \leftarrow$ Initialize parameters
repeat
  $\mathbf{X}^M \leftarrow$ Random minibatch of M datapoints (drawn from full dataset)
  $\boldsymbol{\epsilon} \leftarrow$ Random samples from noise distribution $p(\boldsymbol{\epsilon})$
  $\mathbf{g} \leftarrow \nabla_{\boldsymbol{\theta},\boldsymbol{\phi}} \widetilde{\mathcal{L}}^M(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{X}^M, \boldsymbol{\epsilon})$ (Gradients of minibatch estimator (8))
  $\boldsymbol{\theta}, \boldsymbol{\phi} \leftarrow$ Update parameters using gradients $\mathbf{g}$ (e.g. SGD or Adagrad [DHS10])
until convergence of parameters $(\boldsymbol{\theta}, \boldsymbol{\phi})$
return $\boldsymbol{\theta}, \boldsymbol{\phi}$

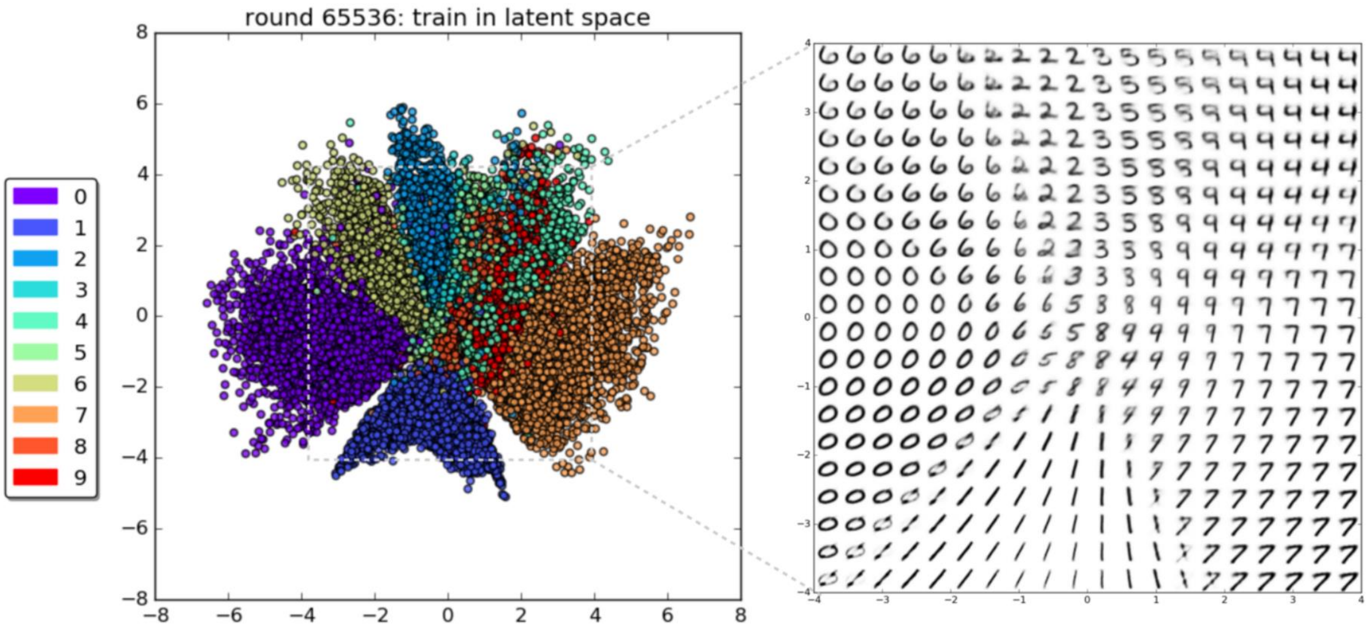
[Kingma & Welling, 2014]

Projecting Means in Latent Space



Source: https://atcold.github.io/pytorch-Deep-Learning/en/week08/08-3/

Projecting Means in Latent Space



Source: https://atcold.github.io/pytorch-Deep-Learning/en/week08/08-3/

VAE: Example Results

Latent code interpolation and sentences generation from VAEs [Bowman et al., 2015] [5]

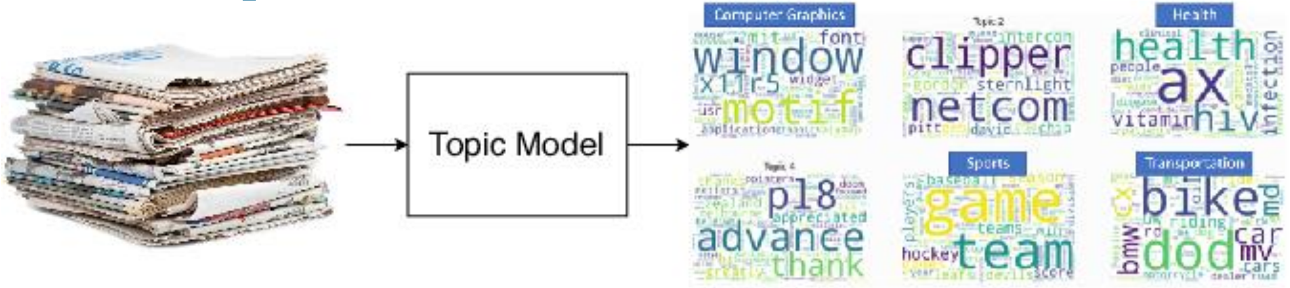
" i want to talk to you . "
"i want to be with you . "
"i do n't want to be with you . "
i do n't want to be with you .
she did n't want to be with him .
he was silent for a long moment .
he was silent for a moment .
it was quiet for a moment .
it was dark and cold .
there was a pause .
it was my turn .

Topic Models

Topics

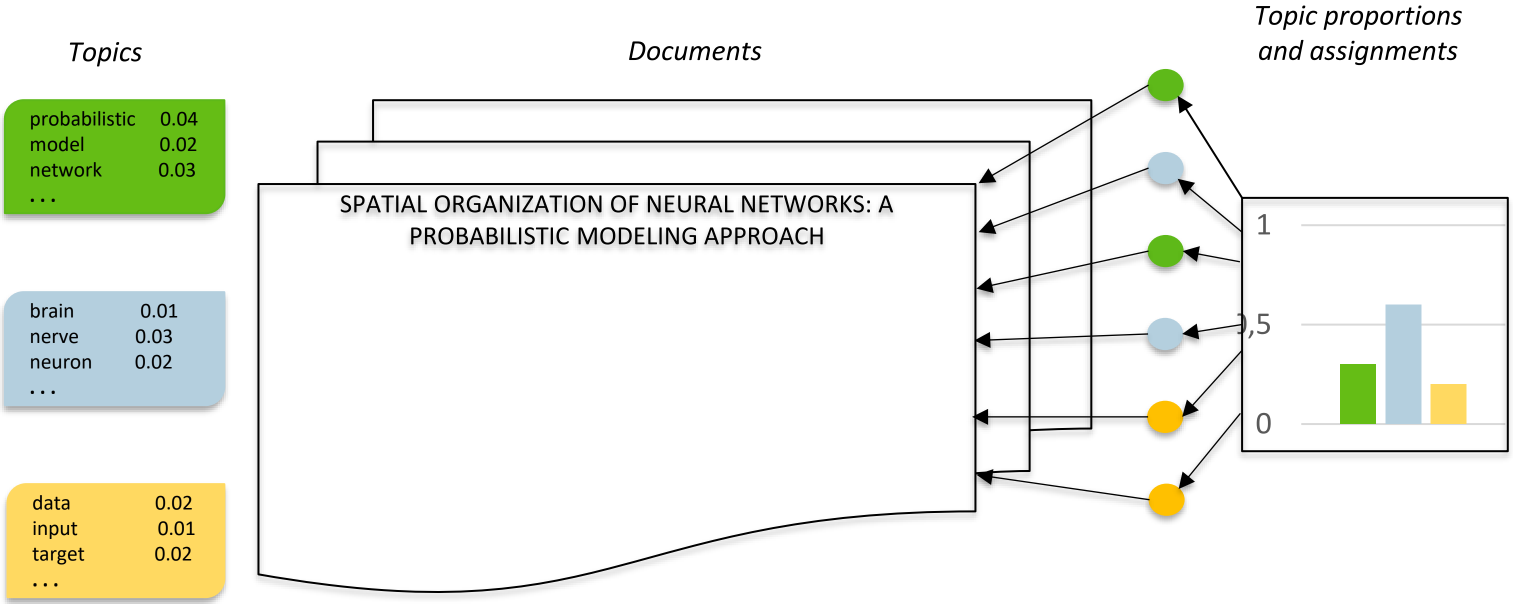
- Word clouds (important words are bigger)
- Probability distributions over words

Topic Models



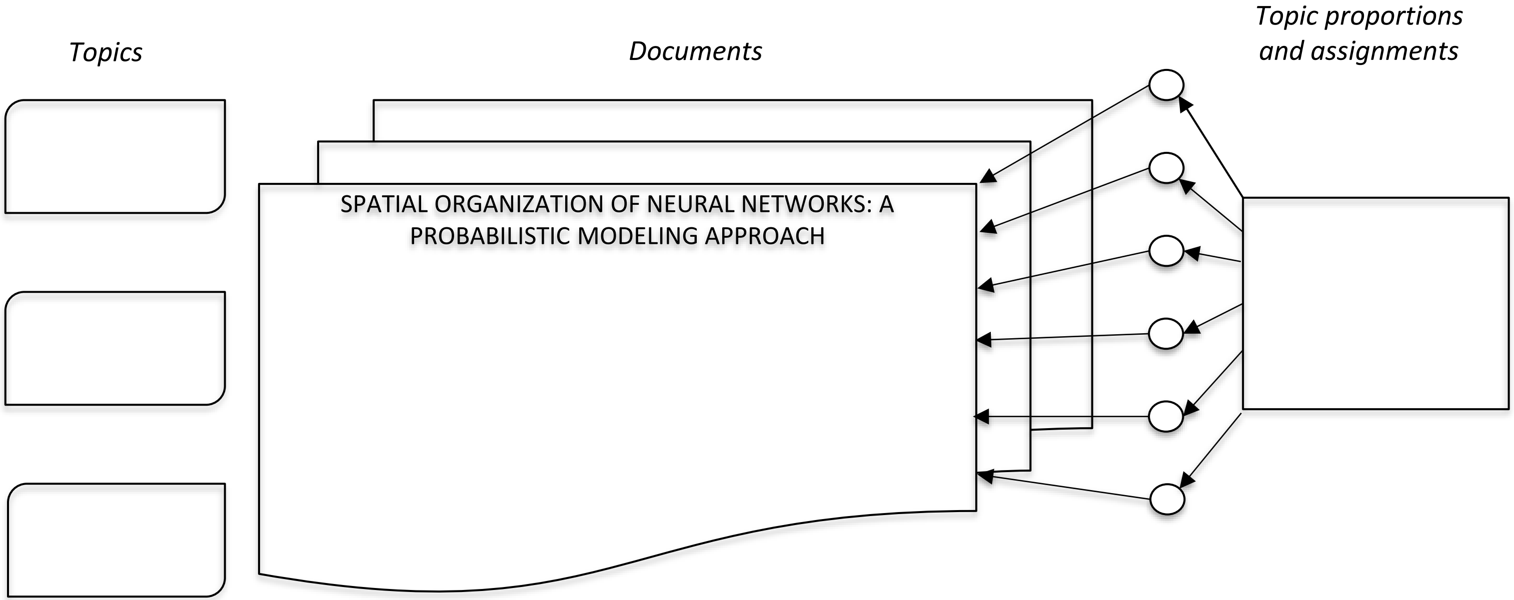
- Input: unstructured text data
- Output: Topics
- No annotations, labels, tags …
- Group documents automatically

What is a "topic"?



- Each **topic** is a distribution of words, each **document** is a mixture of corpus-wide topics; each **word** is drawn from one of those topics.

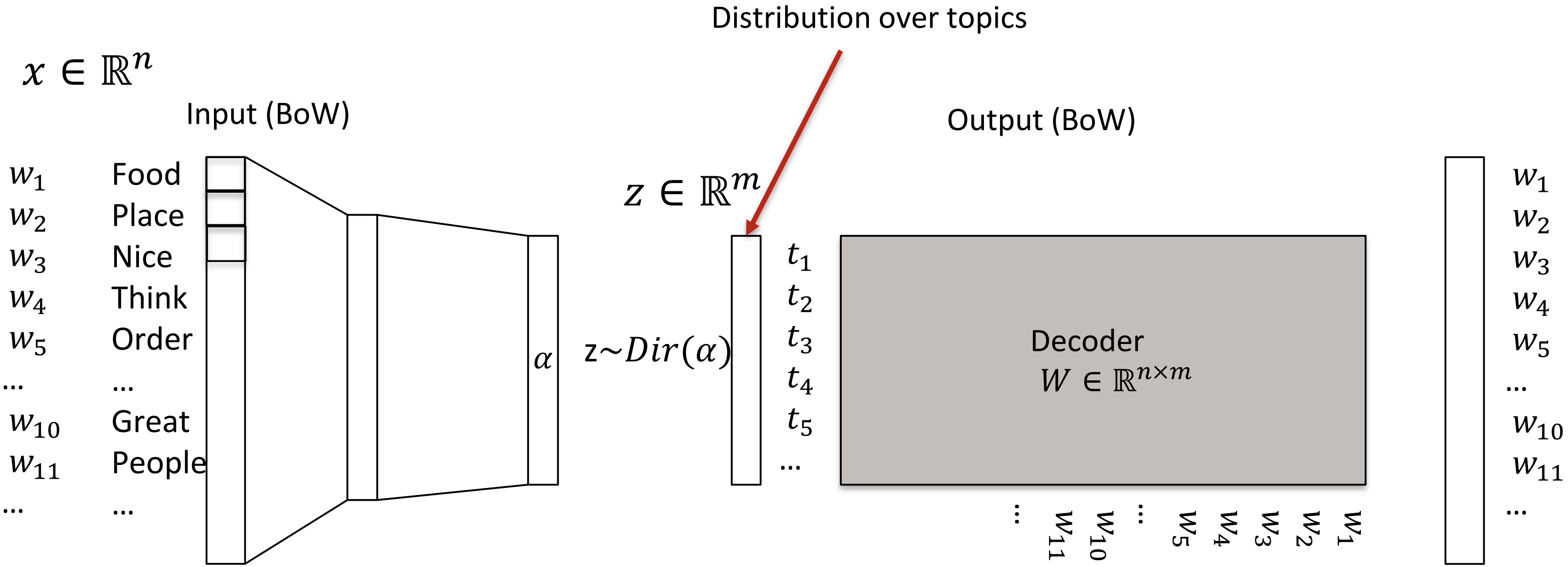
What is a "topic"?



- In reality, we only observe documents. The other structures are hidden variables. Our goal is to **infer** the hidden variables.

VAE-based Topic Models

Dirichlet VAE

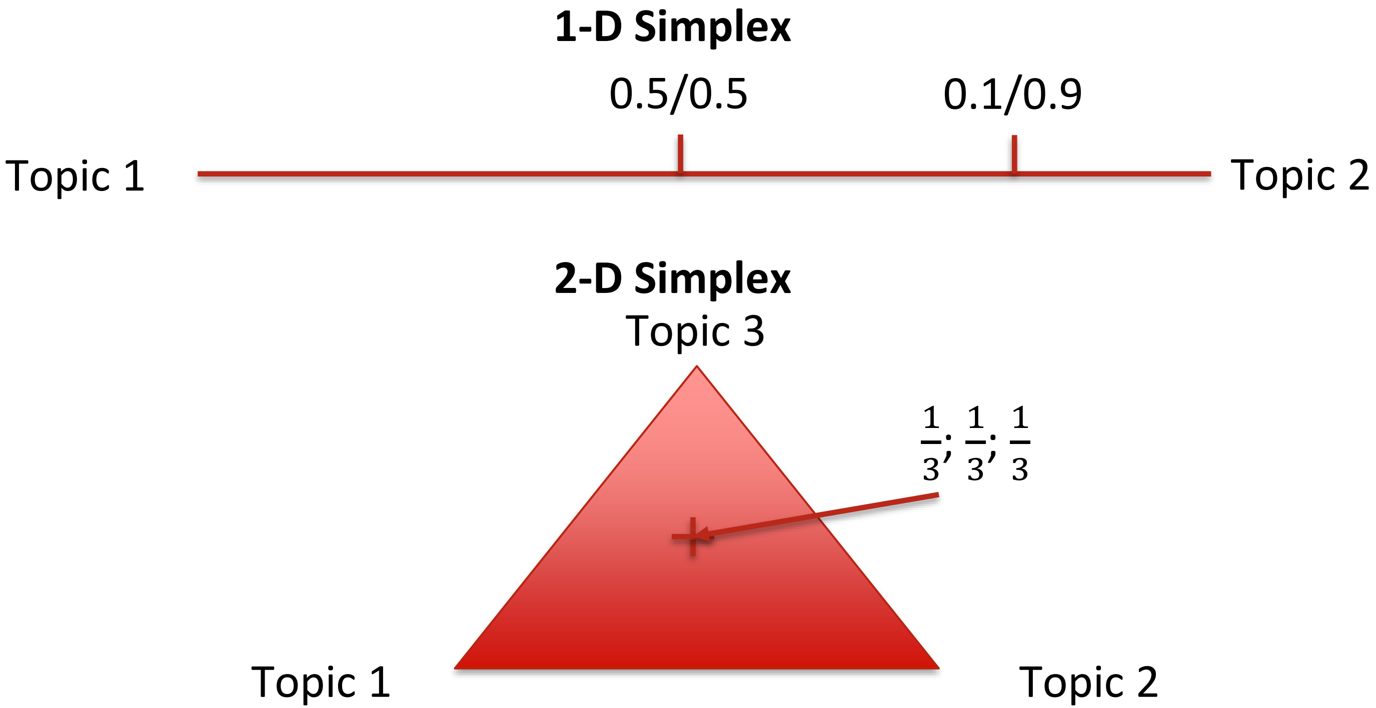


The Dirichlet Distribution

- The Dirichlet distribution is a distribution over the (K-1)-dimensional simplex
- It is parameterized by a K-dimensional vector $(\alpha_1, \dots, \alpha_K)$ such that $\alpha_k \geq 0, k = 1, \dots, K$ and $\sum_k \alpha_k > 0$
- Its distribution is given by

$$\frac{\prod_{k=1}^{K} \Gamma(\alpha_k)}{\Gamma(\sum_{k=1}^{K} \alpha_k)} \prod_{k=1}^{K} \pi_k^{\alpha_k - 1}$$

Probability Simplex



Samples from the Dirichlet

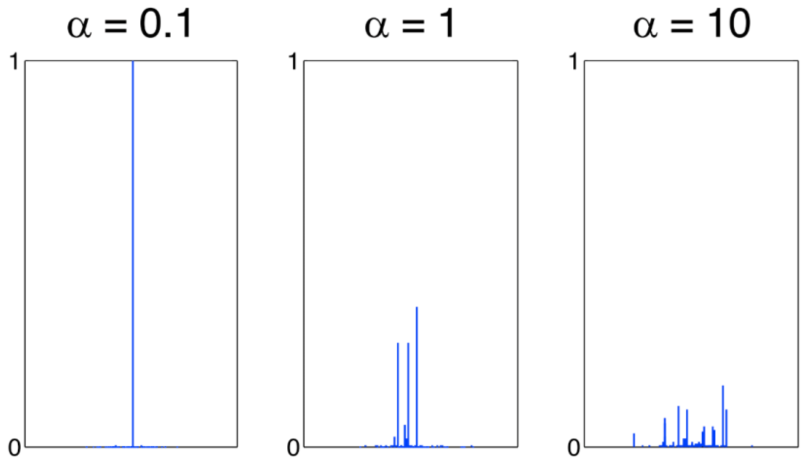
- If $\pi \sim Dirichlet(\alpha_1, \dots, \alpha_K)$ then $\pi_k \geq 0$ for all k, and $\sum_{k=1}^{K} \pi_k = 1$

- Expectation: $\mathbb{E}[(\pi_1, \dots, \pi_K)] = \frac{(\alpha_1, \dots, \alpha_K)}{\sum_k \alpha_k}$

$\alpha = (0.01, 0.01, 0.01)$ $\alpha = (100, 100, 100)$ $\alpha = (5, 50, 100)$

Samples from the Dirichlet Process

- The concentration parameter α determines the distribution over atom sizes
- Small values of α give sparse distributions



Why Dirichlet?

- One document should be assigned to **as few topics as possible**
- Why?
- If one document is assigned to many topics, what would/could happen?

VAE-Based Topic Models

- Basic neural topic modelling (NTM) architecture, was proposed by Miao et al. (2015) (Gaussian).
- Laplace approximation by Srivastava and Sutton (2017)
- **Implicit reparameterization (Figurnov et al. 2018): this is implemented in Pytorch and will be used per default**
- Weibull autoencoder (Zhang et al. 2018)
- Rejection sampling variational inference (Burkhardt and Kramer 2019)
- Inverse CDF (Joo et al. 2019): uses inverse CDF as reparameterization function

Component collapsing

- menu minutes service ordered new order came went table way
- wait try minutes going time good vegas right table got
- find ordered menu want got little great bar vegas went
- location vegas come new pretty think order drinks minutes table
- great love [UNK] went better right little best want staff
- find menu want think pretty cheese [UNK] ordered drinks staff
- new went try nice staff best like find [UNK] better
- try best pretty think love good wait staff want bar
- wait good try new came minutes ordered better order food
- people come food good love way service time drinks vegas
- try come love food restaurant minutes like ordered staff cheese

Extreme Component collapsing

- nice service went wait great [UNK] think want food time
- ordered [UNK] nice going like people went think food great
- time want great food going got [UNK] place wait order
- food going wait good nice got ordered think people order
- food order like people want ordered wait think time good
- ordered place time food good order people wait think want
- good place great like order nice wait ordered time people
- going got time service place like went people order [UNK]
- order ordered like food time got people think service nice
- got went good people time great place going service order

Beta-VAE

$$\mathcal{L}(\theta, \phi; x) = \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)] - D_{KL}(q_\phi(z|x)||p(z))$$

$$\mathcal{L}(\theta, \phi; x) = \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)] - \boldsymbol{\beta} D_{KL}(q_\phi(z|x)||p(z))$$

Stopwords

- is are they your do buy always enjoy go favorite
- to look 's really so have looking pretty burger see
- salad of chicken little the sauce lunch cheese i beans
- she __num__ asked told for did would her % __alpha_num__
- knowledgeable stroll keeping shaped cochon darin create peter inflated central
- ahí una tardaron fuimos entramos hubiera hospedando cómo todos mismo
- we warm our were us first ordered stopped potato __num__
- uh curries sweaty mill televisions tee fortunately landing tasters buttered

Good Topics

- Thai, soup, rice, tuna, roll
- Bartender, tables, drinks, server, restaurant
- Burger, fries, eat, long, bun
- Room, nail, bathroom, rooms, salon
- Coffee, breakfast, cute, brunch, cafe

Summary

- VAEs are generative models that learn a generative distribution for the data
- Reparameterization trick allows for stable training
- VAEs can be used to generate text, as sequential text or in topic models
- The Dirichlet distribution gives better topics because it enforces sparsity

References

- Kingma, Diederik, and Max Welling. “Auto-encoding variational Bayes” https://arxiv.org/abs/1312.6114 (2014).
- Srivastava, Akash, and Charles Sutton. "Autoencoding variational inference for topic models." *arXiv preprint arXiv:1703.01488* (2017).
- Bowman, Samuel R., et al. "Generating sentences from a continuous space." *arXiv preprint arXiv:1511.06349* (2015).