Artificial Intelligence for Business Research @Antai

# Unsupervised Learning: Clustering, Topic Modeling, Variational Auto-Encoder

Renyu (Philip) Zhang

1

## Unsupervised Learning

- Reference: https://sites.google.com/view/berkeley-cs294-158-sp24/home
- We use a collection of observations $X_1$, $X_2$, … $X_n$ sampled from a distribution to describe the property/pattern of the distribution.
- Different types of unsupervised learning:
  - Clustering (k-means, Gaussian mixture, etc.)
  - Topic modelling (Latent Dirichlet Allocation, Variational Inference, etc.)
  - Generative models (VAE, Diffusion, GAN)
  - Principal component analysis (PCA, ICA, etc.)
  - Auto-regressive and self-supervised learning (GPTs)
  - Compression (Huffman Code, etc.)
  - Flow models
  - Etc.

2

2

# Why Do We Care About Unsupervised Learning?

"The brain has about $10^{14}$ synapses, and we only live for about $10^9$ seconds. So, we have a lot more parameters than data. This motivates the idea that we must do a lot of unsupervised learning since the perceptual input (including proprioception) is the only place we can get $10^5$ dimensions of constraint per second."

Geoffrey Hinton (2014) @ AMA Reddit

Reference: https://www.reddit.com/r/MachineLearning/comments/2lmo0l/ama_geoffrey_hinton/

3

3

# Why Do We Care About Unsupervised Learning?

LeCun (2016) @ NeurIPS Keynote

Need tremendous amount of information to build machines that have common sense and generalize.

- "Pure" Reinforcement Learning (cherry)
  - The machine predicts a scalar reward given once in a while.
  - **A few bits for some samples**
- Supervised Learning (icing)
  - The machine predicts a category or a few numbers for each input
  - Predicting human-supplied data
  - **10→10,000 bits per sample**
- Unsupervised/Predictive Learning (cake)
  - The machine predicts any part of its input for any observed part.
  - Predicts future frames in videos
  - **Millions of bits per sample**

LeCake

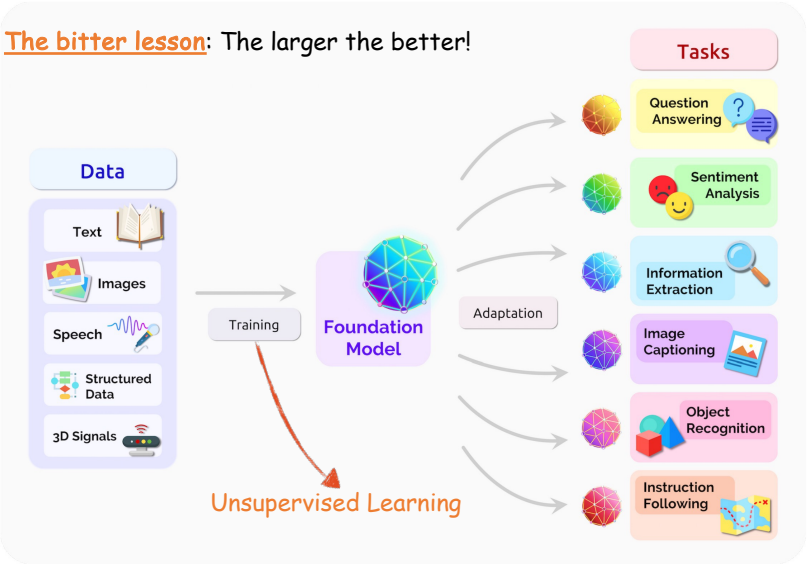
- (Yes, I know, this picture is slightly offensive to RL folks. But I'll make it up)

4

4

# Pretraining: Scaling Unsupervised Learning on the Internet

References: Stanford CS224N, Lecture 9: https://web.stanford.edu/class/cs224n/slides/cs224n-2024-lecture09-pretraining.pdf
Princeton COS 597G, Lecture 2: https://www.cs.princeton.edu/courses/archive/fall22/cos597G/lectures/lec02.pdf




- The model can process large-scale, diverse data sets.
- Never use labelled data; otherwise, you cannot scale.

5

5

# Agenda

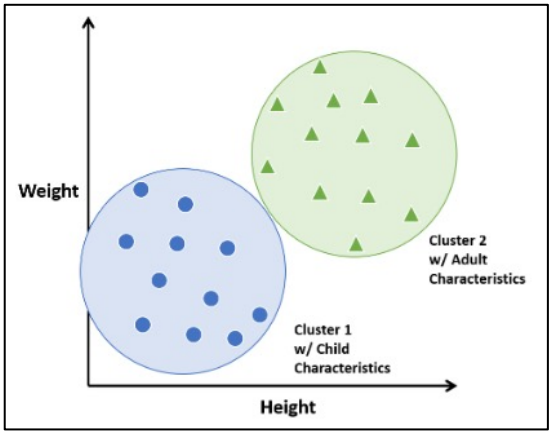
- Clustering: K-Means, Gaussian Mixture Models, EM Algorithm
- Topic Modeling: Latent Dirichlet Allocation
- Variational Auto-Encoder

6

6

# Clustering

- Data = {$X_i \in R^d$: i = 1, 2, 3, …, n}
- Output: A (non-overlapping) partition of the dataset $C_1$, $C_2$, $C_3$, … $C_k$



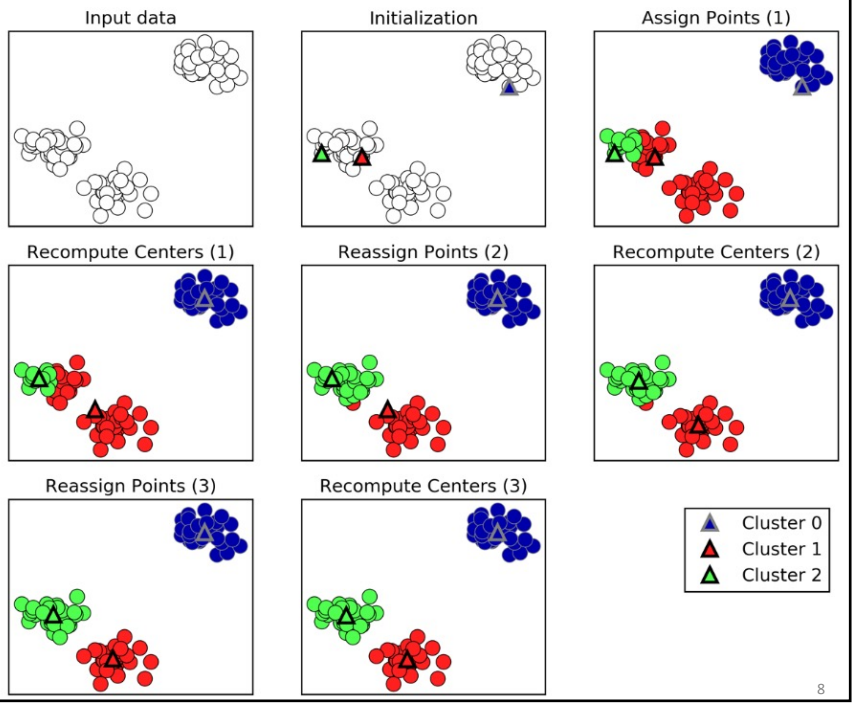

Customer Clustering

7

7

# K-Means Clustering

- Data = {$X_i \in R^d$: i = 1, 2, 3, …, n}
- Output: A (non-overlapping) partition of the dataset $C_1$, $C_2$, $C_3$, … $C_k$
- Equivalent output for K-means, cluster centers: $y_1$, $y_2$, $y_3$, … $y_k$
- $y_j = \frac{\sum_{i \in Cj} X_i}{|Cj|}$ for j = 1, 2, …, k




8

8

# K-Means Algorithm

**Algorithm 1** K-Means Clustering (Lloyd's Algorithm) *Note: written for clarity, not efficiency.*

**Input:** Data vectors $\{\boldsymbol{x}_n\}_{n=1}^N$, number of clusters $K$
**for** $n \leftarrow 1 \ldots N$ **do** ▷ Initialize all of the responsibilities.
  $\boldsymbol{r}_n \leftarrow [0, 0, \cdots, 0]$ ▷ Zero out the responsibilities.
  $k' \leftarrow \text{RandomInteger}(1, K)$ ▷ Make one of them randomly one to initialize.
  $r_{nk'} = 1$
**end for**
**repeat**
  **for** $k \leftarrow 1 \ldots K$ **do** ▷ Loop over the clusters.
    $N_k \leftarrow \sum_{n=1}^N r_{nk}$ ▷ Compute the number assigned to cluster $k$.
    $\boldsymbol{\mu}_k \leftarrow \frac{1}{N_k} \sum_{n=1}^N r_{nk} \boldsymbol{x}_n$ ▷ Compute the mean of the $k$th cluster.
  **end for**
  **for** $n \leftarrow 1 \ldots N$ **do** ▷ Loop over the data.
    $\boldsymbol{r}_n \leftarrow [0, 0, \cdots, 0]$ ▷ Zero out the responsibilities.
    $k' \leftarrow \arg\min_k ||\boldsymbol{x}_n - \boldsymbol{\mu}_k||^2$ ▷ Find the closest mean.
    $r_{nk'} = 1$
  **end for**
**until** none of the $\boldsymbol{r}_n$ change
**Return** assignments $\{\boldsymbol{r}_n\}_{n=1}^N$ for each datum, and cluster means $\{\boldsymbol{\mu}_k\}_{k=1}^K$.

Expectation

EM Algorithm

Maximization

9

9

# Convergence of K-Means

- Criteria of k-means (sometimes called inertia): The squared distance between each data observation and the center of the cluster it belongs to.

$$L(\mu) = \sum_{i=1}^{n} \min_{j \in \{1,\ldots,k\}} \|x_i - \mu_j\|_2^2.$$

- Each maximization step will always (weakly) reduce the criteria.

$$L(\mu, z^*) - L(\mu, z) = \sum_{i=1}^{n} \left( \|x_i - \mu_{z_i^*}\|_2^2 - \|x_i - \mu_{z_i}\|_2^2 \right) \leq 0$$

- Each expectation step will also always (weakly) reduce the criteria.

$$L(\mu^*, z) - L(\mu, z) = \sum_{j=1}^{k} \left( \left( \sum_{i:z_i=j} \|x_i - \mu_j^*\|_2^2 \right) - \left( \sum_{i:z_i=j} \|x_i - \mu_j\|_2^2 \right) \right) \leq 0$$

- Therefore, k-means will always converge.

10

10

Data aggregation and demand prediction
MC Cohen, R Zhang, K Jiao
Operations Research, 2022 · pubsonline.informs.org

We study how retailers can use data aggregation and clustering to improve demand prediction. High accuracy in demand prediction allows retailers to effectively manage their inventory as well as mitigate stock-outs and excess supply. A typical retail setting involves predicting demand for hundreds of items simultaneously. Although some items have a large amount of historical data, others were recently introduced and, thus, transaction data can be scarce. A common approach is to cluster several items and estimate a joint model for each cluster. In this vein, one can estimate some model parameters by aggregating the data from several items and other parameters at the individual-item level. We propose a practical method referred to as *data aggregation with clustering* (DAC), which balances the tradeoff between data aggregation and model flexibility. DAC allows us to predict demand while optimally identifying the features that should be estimated at the (i) item, (ii) cluster, and (iii) aggregate levels. We show that the DAC algorithm yields a consistent and normal estimate, along with improved prediction errors relative to the decentralized benchmark, which estimates a different model for each item. Using both simulated and real data, we illustrate DAC's improvement in prediction accuracy relative to a wide range of common benchmarks. Interestingly, the DAC algorithm has theoretical and practical advantages and helps retailers uncover meaningful managerial insights.

INFORMS

SHOW LESS

☆ Save 99 Cite Cited by 18 Related articles All 7 versions Web of Science: 2 »

A **model-based embedding technique** for **segmenting customers**
S Jagabathula, L Subramanian… - Operations …, 2018 - pubsonline.informs.org
… recommending new movies to **customers**. We show that **segmenting customers** using our **method** and customizing recommendations to each **segment** improves the recommendation …
☆ Save 99 Cite Cited by 23 Related articles All 9 versions Web of Science: 10 »

# Demand Prediction with Clustering

- Apply clustering to the coefficients of different features in a generalized linear model to determine whether the coefficient should be estimated at a department, cluster, or individual level.
- Fundamentally, this is a bias-variance trade-off.
- Practical values of DAC are demonstrated with the real sales data from a US online retailer.

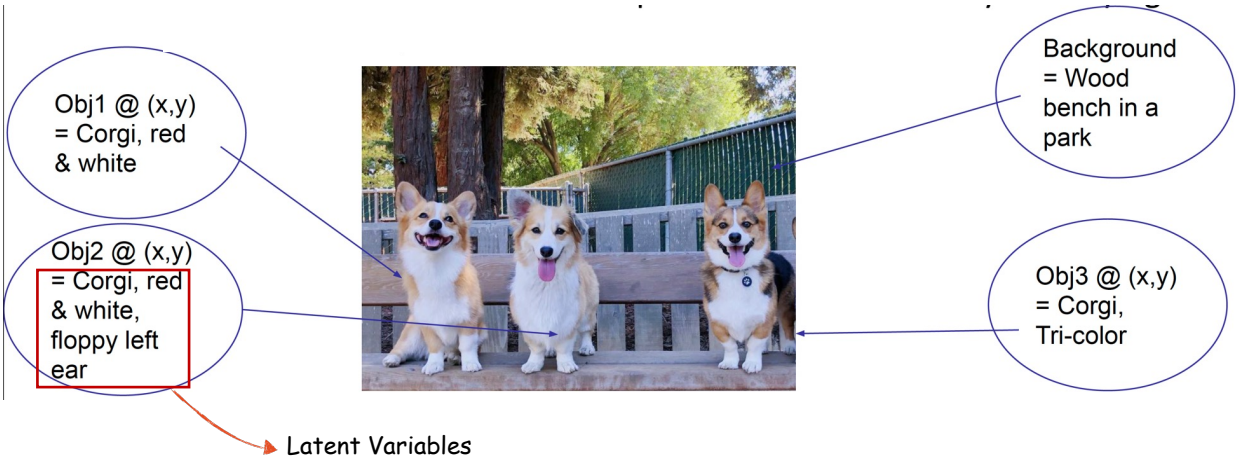
A **dynamic clustering approach** to **data-driven assortment personalization**
F Bernstein, S Modaresi, D Sauré - Management Science, 2019 - pubsonline.informs.org
… **approach** that integrates **dynamic clustering** (segmentation) and demand learning with **dynamic assortment personalization**. The … : **dynamic assortment** planning with demand learning, …
☆ Save 99 Cite Cited by 110 Related articles All 8 versions Web of Science: 40 »

11

11

# Latent Variable Models

- Reference: https://sites.google.com/view/berkeley-cs294-158-sp24/home
- The data generating process may have some low-dimensional hidden representations which could be automatically identified by latent variable models.



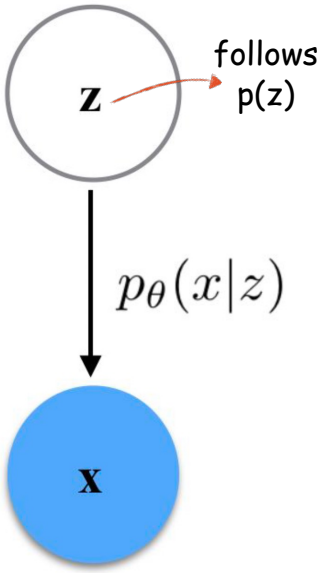

12

12

# Latent Variable Models

- Reference: https://sites.google.com/view/berkeley-cs294-158-sp24/home

1. Generate the latent variable z from prior distribution p(z).
2. Estimate $\theta$ and generate x conditioned on z from the distribution $p_\theta(x|z)$.
3. Update the distribution P(z) based on $p_\theta(x|z)$ and data.
4. Repeat Steps 1 ~ 3 until we cannot find a better P(z) to generate z.

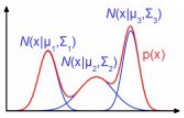



13

13

# Gaussian Mixture Models (GMM)

- Primitives of GMM:
  - A generative model for data clustering
  - Data assumed generated from a mixture of $K$ Gaussians




  - Let $0 \le \pi_k \le 1$ denote the "mixing weight" of the $k$-th Gaussian. It means:
    - $\pi_k$ is the fraction of points generated from the $k$-th Gaussian
    - $\pi_k = p(\boldsymbol{z}_n = k)$ is the prior prob. of $\boldsymbol{x}_n$ belonging to the $k$-th Gaussian
  - Let $\pi = (\pi_1, \pi_2, \ldots, \pi_K)$ denote the vector of mixing wts of $K$ Gaussians. This is a probability vector and sums to 1, i.e., $\sum_{k=1}^{K} \pi_k = 1$
  - Notation $\boldsymbol{z}_n = k$ is equivalent to a size $K$ one-hot vector $\boldsymbol{z}_n$

$$\boldsymbol{z}_n = \underbrace{[0\ 0\ \ldots\ 1\ 0\ 0]}_{\text{all zeros except the k-th bit, i.e., } z_{nk} = 1}$$

Reference: https://sites.google.com/view/berkeley-cs294-158-sp24/home

14

14

# Gaussian Mixture Models (GMM)

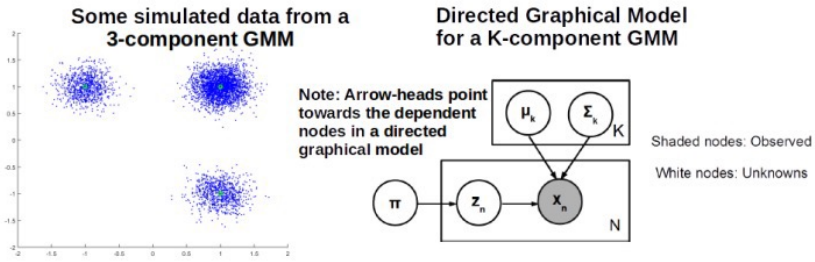
- Data generation process:
  - First choose one of the $K$ mixture components as

$$z_n \sim \text{Multinomial}(z_n|\pi) \qquad \text{(from the prior } p(z) \text{ over } z)$$

  - Suppose $z_n = k$. Now generate $x_n$ from the $k$-th Gaussian as

$$x_n \sim \mathcal{N}(x_n|\mu_k, \Sigma_k) \qquad \text{(from the data distr. } p(x|z))$$

Pdf of normal




Reference: https://sites.google.com/view/berkeley-cs294-158-sp24/home

15

15

# Gaussian Mixture Models (GMM)

- We use maximum likelihood (MLE) to estimate GMM:

$$p(x) = \sum_z p(x, z) = \sum_{k=1}^{K} p(x, z = k) = \sum_{k=1}^{K} p(z = k)p(x|z = k) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x|\mu_k, \Sigma_k)$$

$$\mathcal{L} = \log \prod_{n=1}^{N} p(x_n) = \sum_{n=1}^{N} \log p(x_n) = \sum_{n=1}^{N} \log \underbrace{\sum_{k=1}^{K} \pi_k \mathcal{N}(x_n|\mu_k, \Sigma_k)}_{\text{params get coupled!}}$$

$$\mathcal{L} = \sum_{n=1}^{N} \log \underbrace{\int_{z_n} p(x_n|z_n)p(z_n)dz_n}_{\text{Ouch! Intractable integral!!!}}$$

- We do not observe the latent variable z, so we must take integration over all possible permutations of z.
  - Use Gibbs sampling or MCMC.

Reference: https://sites.google.com/view/berkeley-cs294-158-sp24/home

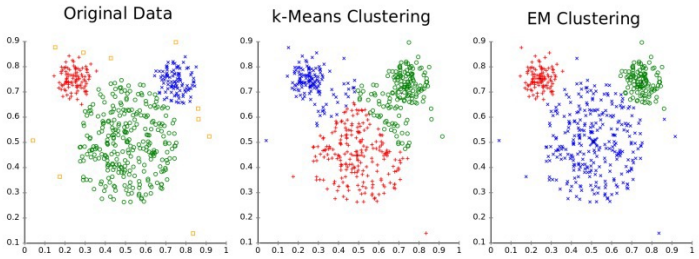
16

16

# GMM Estimation

- We use EM algorithm to estimate:

$$\mathcal{L} = \log \prod_{n=1}^{N} p(\boldsymbol{x}_n) = \sum_{n=1}^{N} \log p(\boldsymbol{x}_n) = \sum_{n=1}^{N} \log \underbrace{\sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}_{\text{params get coupled!}}$$




- The essence is to first guess z, then use MLE to estimate the parameters of p(x|z) (M-step), then based on the estimated parameters to re-guess z (E-step), repeat.

- Initialize the parameters $\Theta = \{\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}_{k=1}^{K}$ randomly, or using $K$-means
- Iterate until convergence (e.g., when $\log p(\boldsymbol{x}|\Theta)$ ceases to increase)

E-step
- Given $\Theta$, compute each expectation $z_{nk}$ (post. prob. of $z_{nk} = 1$), $\forall n, k$

By Bayes rule; called responsibilities. $\gamma_{nk} = \mathbb{E}[z_{nk}] \propto \pi_k \mathcal{N}(\boldsymbol{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ (and re-normalize s.t. $\sum_{k=1}^{K} \gamma_{nk} = 1$)

M-step
- Given $\gamma_{nk} = \mathbb{E}[z_{nk}]$ and $N_k = \sum_{n=1}^{N} \gamma_{nk}$, update $\Theta = \{\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}_{k=1}^{K}$ as

By MLE and setting the derivatives of log-likelihood with respect to each parameter to 0.

$$\boldsymbol{\mu}_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma_{nk} \boldsymbol{x}_n$$

$$\boldsymbol{\Sigma}_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma_{nk} (\boldsymbol{x}_n - \boldsymbol{\mu}_k)(\boldsymbol{x}_n - \boldsymbol{\mu}_k)^\top$$

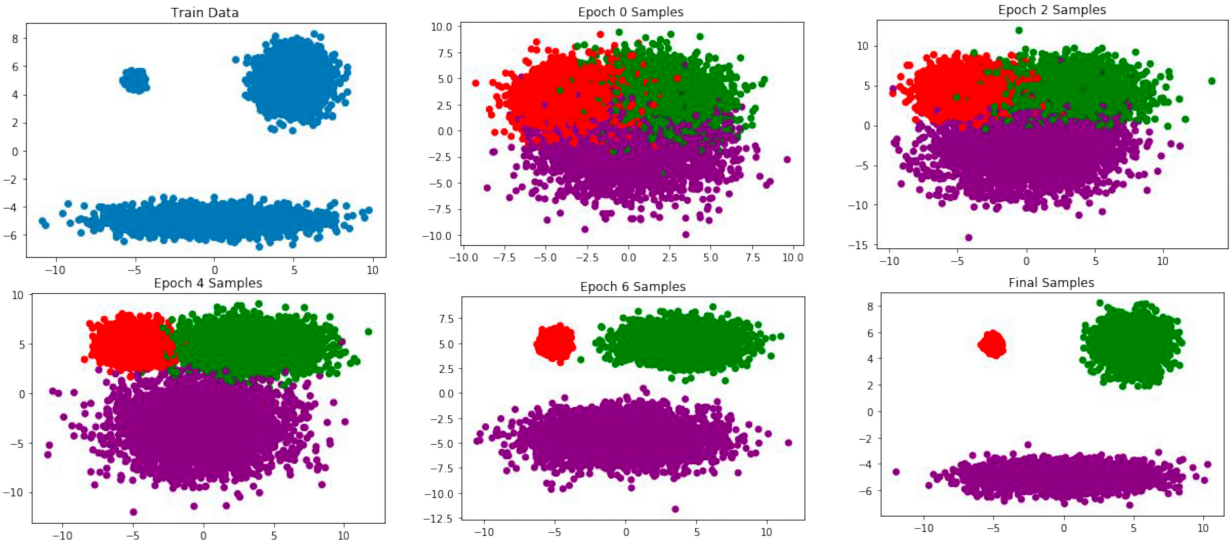
$$\pi_k = \frac{N_k}{N}$$

Reference: https://sites.google.com/view/berkeley-cs294-158-sp24/home

17

17

# GMM Estimation




Reference: https://sites.google.com/view/berkeley-cs294-158-sp24/home

18

18

# EM Algorithm Framework

- A general latent variable framework:

  - Consider a generative model with joint distr. $p(\mathbf{X}, \mathbf{Z}|\Theta) = \prod_{n=1}^{N} p(\boldsymbol{x}_n, \boldsymbol{z}_n)$
    - Observed data: $\mathbf{X} = \{\boldsymbol{x}_n\}_{n=1}^{N}$
    - Latent variables: $\mathbf{Z} = \{\boldsymbol{z}_n\}_{n=1}^{N}$. All the model parameters: $\Theta$ → $\pi, \mu, \Sigma$ in GMM
  - Goal: Estimate the model parameters $\Theta$ via MLE (or MAP) MAP: Maximum A Posteriori

$$\hat{\Theta} = \arg\max_{\Theta} \log p(\mathbf{X}|\Theta) = \arg\max_{\Theta} \log \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\Theta) \quad \text{(when } \mathbf{Z} \text{ is discrete)}$$

$$= \arg\max_{\Theta} \log \int_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\Theta) d\mathbf{Z} \quad \text{(when } \mathbf{Z} \text{ is continuous)}$$

19

19

# EM Algorithm Framework

- EM algorithm to estimate the latent variable model:

Initialize the parameters: $\Theta^{old}$. Then alternate between these steps:

- **E (Expectation) step:** → $\gamma_{nk}$ in GMM
  - Compute the posterior $p(\mathbf{Z}|\mathbf{X}, \Theta^{old})$ over latent variables $\mathbf{Z}$ using $\Theta^{old}$
  - Compute the expected complete data log-likelihood w.r.t. *this* posterior

$$\mathcal{Q}(\Theta, \Theta^{old}) = \mathbb{E}_{p(\mathbf{Z}|\mathbf{X}, \Theta^{old})}[\log p(\mathbf{X}, \mathbf{Z}|\Theta)] = \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \Theta^{old}) \log p(\mathbf{X}, \mathbf{Z}|\Theta)$$

- **M (Maximization) step:**
  - Maximize the expected complete data log-likelihood w.r.t. $\Theta$

$$\Theta^{new} = \arg\max_{\Theta} \mathcal{Q}(\Theta, \Theta^{old}) \quad \text{(if doing MLE)}$$

$$\Theta^{new} = \arg\max_{\Theta} \{\mathcal{Q}(\Theta, \Theta^{old}) + \log p(\Theta)\} \quad \text{(if doing MAP)}$$

- If the log-likelihood or the parameter values not converged then set $\Theta^{old} = \Theta^{new}$ and go to the E step.

20

20

# Latent Class Models

- Similarly, we have used the latent class model to capture heterogeneity in many structural estimations, e.g., mixed logit with customer segmentation.
  - In these models, we assume there exists a probability vector ($p_1$,$p_2$,…,$p_k$), which describes the probability of each data point belonging to each of k segments.
- EM-type of algorithms are the standard approach to estimate such models.
- EM algorithms for demand estimation:

**Estimation** of **consumer demand** with **stock-out based substitution**: An **application** to **vending machine products**
R Anupindi, M Dada, S Gupta - Marketing Science, 1998 - pubsonline.informs.org
… in which times of **stock-out** occurrence and cumulative sales of all **goods** up to these times … in retail **vending** provide only periodic data, ie, data in which times of **stock-out** occurrence are …
☆ Save 99 Cite Cited by 378 Related articles All 15 versions Web of Science: 136 »

Estimating primary demand for substitutable products from sales transaction data
G **Vulcano**, G **Van Ryzin**, R Ratliff - Operations Research, **2012** - pubsonline.informs.org
We propose a method for estimating substitute and lost demand when only sales and product availability data are observable, not all products are displayed in all periods (eg, due to …
☆ Save 99 Cite Cited by 348 Related articles All 14 versions Web of Science: 142 »

21

21

# Agenda

- Clustering: K-Means, Gaussian Mixture Models, EM Algorithm
- Topic Modeling: Latent Dirichlet Allocation
- Variational Auto-Encoder

22

22

# Topic Modeling

- Topic modeling: An unsupervised way of simultaneously (a) finding topics from a set of documents and (b) classify these documents into these topics.

- Input: A bunch of documents without labels.
- Output:
  1. Topics and representation of topics by words.
  2. Classification of documents into topics.
- Applications:
  - Organize the documents into thematic categories/topics.
  - Describe the evolution of those topics over time.
  - Enable a domain expert to analyze and understand the content.
  - Find relationships between the categories.
  - Understand how authorship influences the content/topics.

23



# Latent Dirichlet Allocation

[PDF] **Latent dirichlet allocation**
DM Blei, AY Ng, MI Jordan - Journal of machine Learning research, 2003 - jmlr.org
We describe **latent Dirichlet allocation** (LDA), a generative probabilistic model for collections of discrete data such as text corpora. LDA is a three-level hierarchical Bayesian model, in …
☆ Save 99 Cite Cited by 51694 Related articles All 84 versions Web of Science: 20749 »

- LDA: Latent Dirichlet Allocation
  - A latent (generative) model that can generate new data instances using latent variables.
- Each document will have a multinomial topic distribution.
- The parameters for the multinomial distributions of words (for each topic) as well as the topics (for each document) are drawn from a Dirichlet distribution, a probability distribution over a vector of random variables on a (k-1)-simplex:

Dirichlet($\alpha$): $$p(\theta \mid \alpha) = \frac{\Gamma\left(\sum_{i=1}^{k} \alpha_i\right)}{\prod_{i=1}^{k} \Gamma(\alpha_i)} \theta_1^{\alpha_1 - 1} \cdots \theta_k^{\alpha_k - 1},$$

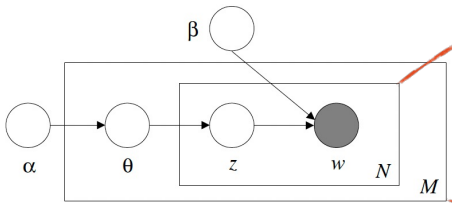
$$0 \le \theta_i \le 1 \text{ for } i = 1,2,\ldots,k \text{ and } \sum_{i=1}^{k} \theta_i = 1$$

- Choose the topic distribution $\theta$ according to Dirichlet($\alpha$); choose a topic z based on Multinomial($\theta$).
- Choose a document length N from Poisson($\xi$); choose a word $w_n$ from $p(wn|zn, \beta)$ with $\beta_{i,j} = p(w = i|z = j)$.

24

# Topic Modeling Generative Process

- Another way to visualize the generative model of LDA:




One document of N words.

$$p(\theta, \mathbf{z}, \mathbf{w} \mid \alpha, \beta) = p(\theta \mid \alpha) \prod_{n=1}^{N} p(z_n \mid \theta) p(w_n \mid z_n, \beta),$$

M documents.

Figure 1: Graphical model representation of LDA. The boxes are "plates" representing replicates. The outer plate represents documents, while the inner plate represents the repeated choice of topics and words within a document.

$$p(\mathbf{w} \mid \alpha, \beta) = \int p(\theta \mid \alpha) \left( \prod_{n=1}^{N} \sum_{z_n} p(z_n \mid \theta) p(w_n \mid z_n, \beta) \right) d\theta.$$

Marginal distribution of the word sequence w in each document.

$$p(D \mid \alpha, \beta) = \prod_{d=1}^{M} \int p(\theta_d \mid \alpha) \left( \prod_{n=1}^{N_d} \sum_{z_{dn}} p(z_{dn} \mid \theta_d) p(w_{dn} \mid z_{dn}, \beta) \right) d\theta_d.$$

Marginal distribution of all documents D.

25

25

# Topic Modeling Inference

- The key inferential problem we need to solve is the posterior distribution of the hidden variables:

$$p(\theta, \mathbf{z} \mid \mathbf{w}, \alpha, \beta) = \frac{p(\theta, \mathbf{z}, \mathbf{w} \mid \alpha, \beta)}{p(\mathbf{w} \mid \alpha, \beta)}$$ , where $$p(\mathbf{w} \mid \alpha, \beta) = \frac{\Gamma(\sum_i \alpha_i)}{\prod_i \Gamma(\alpha_i)} \int \left( \prod_{i=1}^{k} \theta_i^{\alpha_i - 1} \right) \left( \prod_{n=1}^{N} \sum_{i=1}^{k} \prod_{j=1}^{V} (\theta_i \beta_{ij})^{w_n^j} \right) d\theta$$

$K^n$ topic combinations

- It is intractable to compute $p(\mathbf{w} \mid \alpha, \beta)$ because we must go through all possible combinations of the topics.
- So, we need to do approximate inference, Variational Inference (VI) in particular.
- VI does posterior inference based on approximating the posterior by a nice distribution.
- Very general idea:

choose a nice family of distributions $\mathcal{Q}$,
find a $q \in \mathcal{Q}$ that is as close as possible to the posterior, and
use $q$ to quantify uncertainty, as a proxy for the posterior.

Closeness is determined by KL divergence.

$$(\gamma^*, \phi^*) = \arg\min_{(\gamma, \phi)} \mathrm{D}(q(\theta, \mathbf{z} \mid \gamma, \phi) \parallel p(\theta, \mathbf{z} \mid \mathbf{w}, \alpha, \beta))$$

$$q(\theta, \mathbf{z} \mid \gamma, \phi) = q(\theta \mid \gamma) \prod_{n=1}^{N} q(z_n \mid \phi_n)$$

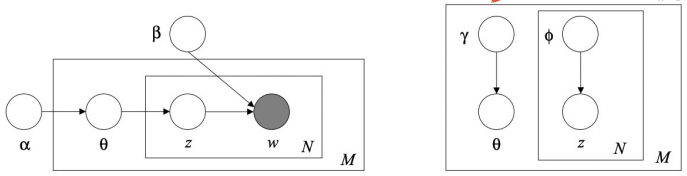



Figure 5: (Left) Graphical model representation of LDA. (Right) Graphical model representation of the variational distribution used to approximate the posterior in LDA.

26

26

# Variational Inference for LDA

$$(\gamma^*, \phi^*) = \arg\min_{(\gamma,\phi)} \mathrm{D}(q(\theta, \mathbf{z} \mid \gamma, \phi) \parallel p(\theta, \mathbf{z} \mid \mathbf{w}, \alpha, \beta))$$

- Setting the derivatives of the KL divergence to 0, we have:

$$\begin{aligned} \phi_{ni} &\propto \beta_{iw_n} \exp\{\mathrm{E}_q[\log(\theta_i) \mid \gamma]\} \\ \gamma_i &= \alpha_i + \textstyle\sum_{n=1}^{N} \phi_{ni}. \end{aligned}$$

$$\mathrm{E}_q[\log(\theta_i) \mid \gamma] = \Psi(\gamma_i) - \Psi\left(\textstyle\sum_{j=1}^{k} \gamma_j\right)$$

where $\Psi(.)$ is the derivative of the $\log\Gamma$ function.

(1) initialize $\phi_{ni}^0 := 1/k$ for all $i$ and $n$
(2) initialize $\gamma_i := \alpha_i + N/k$ for all $i$
(3) **repeat**
(4) **for** $n = 1$ **to** $N$
(5) **for** $i = 1$ **to** $k$
(6) $\phi_{ni}^{t+1} := \beta_{iw_n} \exp(\Psi(\gamma_i^t))$
(7) normalize $\phi_n^{t+1}$ to sum to 1.
(8) $\gamma^{t+1} := \alpha + \sum_{n=1}^{N} \phi_n^{t+1}$
(9) **until** convergence

The algorithm obtains $(\gamma^*(\boldsymbol{w}), \phi^*(\boldsymbol{w}))$.

Figure 6: A variational inference algorithm for LDA.

27

27

# Variational EM Algorithm

- To estimate the parameters $(\alpha, \beta)$, we want to minimize the intractable $\ell(\alpha, \beta) = \sum_{d=1}^{M} \log p(\mathbf{w}_d \mid \alpha, \beta)$, where M is the number of documents.
- We instead apply the variational EM algorithm:

$$\ell(\alpha, \beta) = \sum_{d=1}^{M} \log p(\mathbf{w}_d \mid \alpha, \beta).$$

As we have described above, the quantity $p(\mathbf{w} \mid \alpha, \beta)$ cannot be computed tractably. However, variational inference provides us with a tractable lower bound on the log likelihood, a bound which we can maximize with respect to $\alpha$ and $\beta$. We can thus find approximate empirical Bayes estimates for the LDA model via an alternating *variational EM* procedure that maximizes a lower bound with respect to the variational parameters $\gamma$ and $\phi$, and then, for fixed values of the variational parameters, maximizes the lower bound with respect to the model parameters $\alpha$ and $\beta$.

We provide a detailed derivation of the variational EM algorithm for LDA in Appendix A.4. The derivation yields the following iterative algorithm:

1. (E-step) For each document, find the optimizing values of the variational parameters $\{\gamma_d^*, \phi_d^* : d \in \mathcal{D}\}$. This is done as described in the previous section.
2. (M-step) Maximize the resulting lower bound on the log likelihood with respect to the model parameters $\alpha$ and $\beta$. This corresponds to finding maximum likelihood estimates with expected sufficient statistics for each document under the approximate posterior which is computed in the E-step.

These two steps are repeated until the lower bound on the log likelihood converges.

$$\beta_{ij} \propto \sum_{d=1}^{M} \sum_{n=1}^{N_d} \phi_{dni}^* w_{dn}^j. \quad (9)$$

We further show that the M-step update for Dirichlet parameter $\alpha$ can be implemented using an efficient Newton-Raphson method in which the Hessian is inverted in linear time.

Adjust $(\alpha, \beta)$ to maximize the log-likelihood:

$$L_{[\beta]} = \sum_{d=1}^{M} \sum_{n=1}^{N_d} \sum_{i=1}^{k} \sum_{j=1}^{V} \phi_{dni} w_{dn}^j \log \beta_{ij} + \sum_{i=1}^{k} \lambda_i \left(\textstyle\sum_{j=1}^{V} \beta_{ij} - 1\right).$$

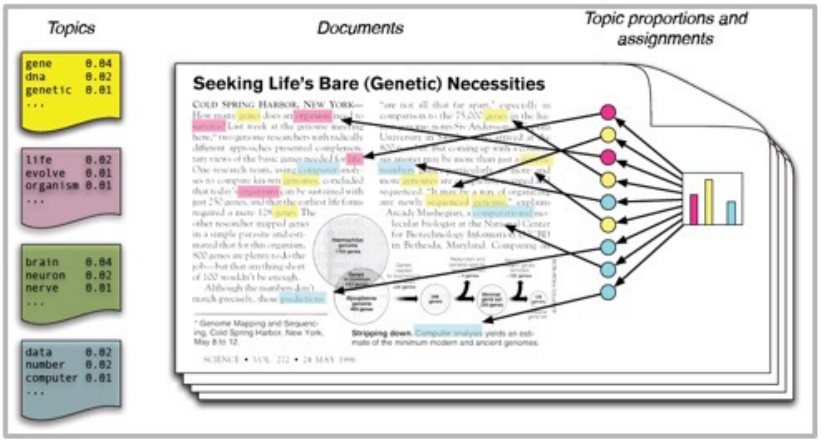
$$L_{[\alpha]} = \sum_{d=1}^{M} \left(\log\Gamma\left(\textstyle\sum_{j=1}^{k} \alpha_j\right) - \sum_{i=1}^{k} \log\Gamma(\alpha_i) + \sum_{i=1}^{k} \left((\alpha_i - 1)\left(\Psi(\gamma_{di}) - \Psi\left(\textstyle\sum_{j=1}^{k} \gamma_{dj}\right)\right)\right)\right)$$

28

28

# Other Topic Modeling Methods

- There are more topic modelling methods:
  - Hierarchical topic modelling.
  - Dynamic topic modelling.
  - Fat-tail topic modelling.
  - Etc.




29

29

CEO behavior and firm performance

O Bandiera, A Prat, S Hansen, R Sadun

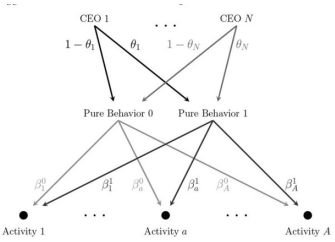
Journal of Political Economy, 2020 · journals.uchicago.edu

We develop a new method to measure CEO behavior in large samples via a survey that collects high-frequency, high-dimensional diary data and a machine learning algorithm that estimates behavioral types. Applying this method to 1,114 CEOs in six countries reveals two types: "leaders," who do multifunction, high-level meetings, and "managers," who do individual meetings with core functions. Firms that hire leaders perform better, and it takes three years for a new CEO to make a difference. Structural estimates indicate that productivity differentials are due to mismatches rather than to leaders being better for all firms.

The University of Chicago Press

SHOW LESS

☆ Save 99 Cite Cited by 360 Related articles All 36 versions Web of Science: 1 »




# CEO Behavior

- Apply LDA to a large panel of CEO diary data and identify two behavioral types.
- "Leaders" who focus on coordination and communication.
- "Managers" who focus on production-related activities.
- Properly matched CEOs and firms enjoy better firm performances.

30

30

## FOMC Transparency

Transparency and deliberation within the FOMC: A computational linguistics approach

S Hansen, M McMahon, A Prat

The Quarterly Journal of Economics, 2018 · academic.oup.com

**Abstract**

How does transparency, a key feature of central bank design, affect monetary policy makers' deliberations? Theory predicts a positive discipline effect and negative conformity effect. We empirically explore these effects using a natural experiment in the Federal Open Market Committee in 1993 and computational linguistics algorithms. We first find large changes in communication patterns after transparency. We then propose a difference-in-differences approach inspired by the career concerns literature, and find evidence for both effects. Finally, we construct an influence measure that suggests the discipline effect dominates.

Oxford University Press

SHOW LESS

☆ Save 99 Cite Cited by 726 Related articles All 37 versions Web of Science: 163 »

- Use LDA to study 149 FOMC meeting transcripts during Alan Greenspan's tenure.
- Measure the proportion of language devoted to K topics (estimated from the fitted topic model) for each FOMC member at each meeting.
- DiD suggests that moving to a more transparent system prompts inexperienced FOMC members to discuss a wider range of topics.

31

31

## Brand Analysis Using Product Reviews

Mining marketing meaning from online chatter: Strategic brand analysis of big data using latent dirichlet allocation

S Tirunillai, GJ Tellis

Journal of marketing research, 2014 · journals.sagepub.com

Online chatter, or user-generated content, constitutes an excellent emerging source for marketers to mine meaning at a high temporal frequency. This article posits that this meaning consists of extracting the key latent dimensions of consumer satisfaction with quality and ascertaining the valence, labels, validity, importance, dynamics, and heterogeneity of those dimensions. The authors propose a unified framework for this purpose using unsupervised latent Dirichlet allocation. The sample of user-generated content consists of rich data on product reviews across 15 firms in five markets over four years. The results suggest that a few dimensions with good face validity and external validity are enough to capture quality. Dynamic analysis enables marketers to track dimensions' importance over time and allows for dynamic mapping of competitive brand positions on those dimensions over time. For vertically differentiated markets (e.g., mobile phones, computers), objective dimensions dominate and are similar across markets, heterogeneity is low across dimensions, and stability is high over time. For horizontally differentiated markets (e.g., shoes, toys), subjective dimensions dominate but vary across markets, heterogeneity is high across dimensions, and stability is low over time.

Sage Journals

SHOW LESS

☆ Save 99 Cite Cited by 870 Related articles All 6 versions Web of Science: 441 »

- Use LDA to extract latent dimensions of consumer satisfaction from online product reviews.
- These dimensions are validated by crowdsourced raters to capture product quality.
- For vertically differentiated markets, objective dimensions dominate with low heterogeneity across markets and high stability over time.
- For horizontally differentiated markets, subjective dimensions dominate with high heterogeneity across markets and low stability over time.

32

32

A semantic approach for estimating consumer content preferences from online search queries

J Liu, O Toubia

Marketing Science, 2018 · pubsonline.informs.org

We extend latent Dirichlet allocation by introducing a topic model, hierarchically dual latent Dirichlet allocation (HDLDA), for contexts in which one type of document (e.g., search queries) are semantically related to another type of document (e.g., search results). In the context of online search engines, HDLDA identifies not only topics in short search queries and web pages, but also how the topics in search queries relate to the topics in the corresponding top search results. The output of HDLDA provides a basis for estimating consumers' content preferences on the fly from their search queries given a set of assumptions on how consumers translate their content preferences into search queries. We apply HDLDA and explore its use in the estimation of content preferences in two studies. The first is a lab experiment in which we manipulate participants' content preferences and observe the queries they formulate and their browsing behavior across different product categories. The second is a field study, which allows us to explore whether the content preferences estimated based on HDLDA may be used to explain and predict click-through rates in online search advertising.

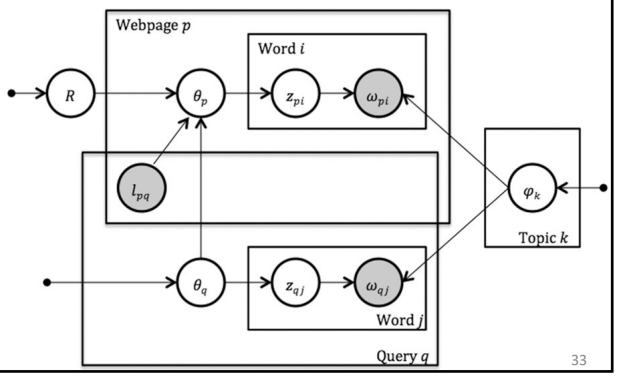
INFORMS

SHOW LESS

☆ Save 99 Cite Cited by 93 Related articles All 10 versions Web of Science: 45 »

## LDA for Consumer Preference Estimation

- Hierarchically dual LDA: Connect search queries and search results with the same set of topics.
- Apply HDLDA to lab experiments data and Google search data to showcase how it can be used to estimate consumer preference.
- The content preferences estimated by HDLDA can help predict the CTR of sponsored ads.




33

33

Reading between the lines: Prediction of political violence using newspaper text

H Mueller, C Rauh

American Political Science Review, 2018 · cambridge.org

This article provides a new methodology to predict armed conflict by using newspaper text. Through machine learning, vast quantities of newspaper text are reduced to interpretable topics. These topics are then used in panel regressions to predict the onset of conflict. We propose the use of the within-country variation of these topics to predict the timing of conflict. This allows us to avoid the tendency of predicting conflict only in countries where it occurred before. We show that the within-country variation of topics is a good predictor of conflict and becomes particularly useful when risk in previously peaceful countries arises. Two aspects seem to be responsible for these features. Topics provide depth because they consist of changing, long lists of terms that make them able to capture the changing context of conflict. At the same time, topics provide width because they are summaries of the full text, including stabilizing factors.

Cambridge University Press

SHOW LESS

☆ Save 99 Cite Cited by 192 Related articles All 26 versions Web of Science: 43 »

## Political Violence Prediction

- Use LDA to reduce newspaper texts into interpretable topics.
- Regress the onset of conflict on these topics.
- Within-country variation of these topics is a good predictor of conflict, providing both depth and width of information into onset of conflict.

34

34

# Agenda

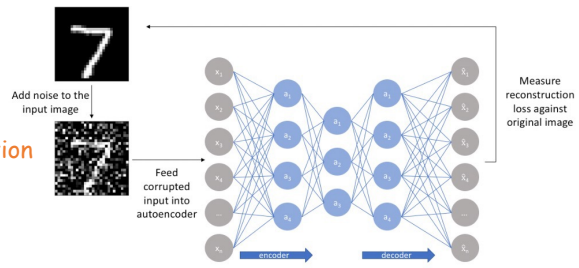
- Clustering: K-Means, Gaussian Mixture Models, EM Algorithm
- Topic Modeling: Latent Dirichlet Allocation
- Variational Auto-Encoder

35

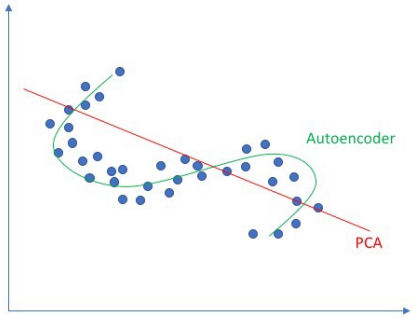
35

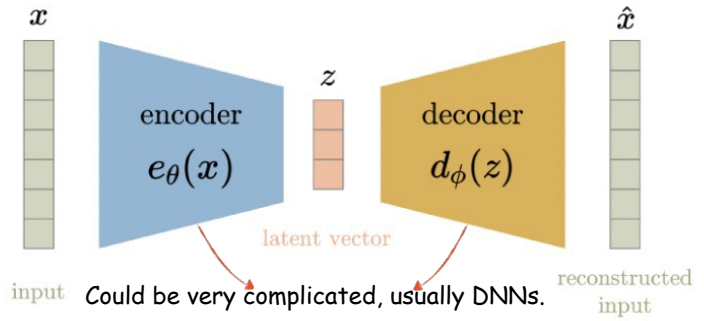
# Autoencoder

- Data = $\{X_i \in R^d: i = 1, 2, 3, \ldots, n\}$
- Output: an encoder and decoder mapping as representation learning for the data.
- Loss: Reconstruction noise + regularization.




Linear vs nonlinear dimensionality reduction




Could be very complicated, usually DNNs.

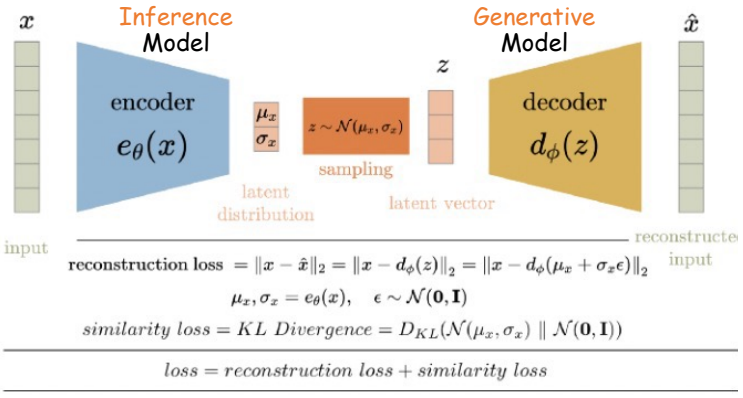
$$loss = \|x - \hat{x}\|_2 = \|x - d_\phi(z)\|_2 = \|x - d_\phi(e_\theta(x))\|_2$$

36

36

# Variational Auto-Encoder (VAE)

- Autoencoder is not generative, but VAE is generative.




**Auto-encoding variational** bayes ICLR 2024 Test-of-Time Award

DP Kingma, M Welling - arXiv preprint arXiv:1312.6114, 2013 - arxiv.org

… We introduce a stochastic **variational** inference and learning … datapoint, we propose the **AutoEncoding** VB (AEVB) algorithm… recognition model, we arrive at the **variational auto-encoder**. …

☆ Save 99 Cite Cited by 34339 Related articles All 44 versions »

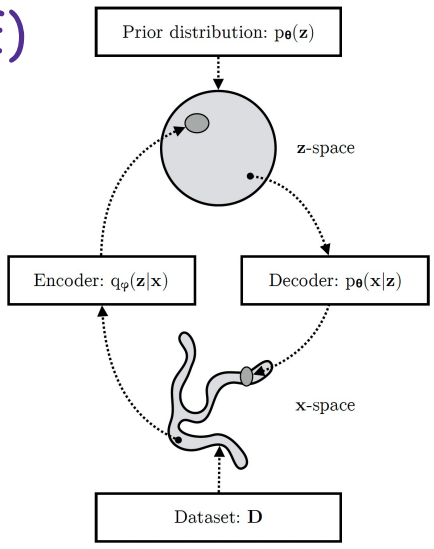



**Figure 2.1:** A VAE learns stochastic mappings between an observed $\mathbf{x}$-space, whose empirical distribution $q_{\mathcal{D}}(\mathbf{x})$ is typically complicated, and a latent $\mathbf{z}$-space, whose distribution can be relatively simple (such as spherical, as in this figure). The generative model learns a joint distribution $p_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{z})$ that is often (but not always) factorized as $p_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{z}) = p_{\boldsymbol{\theta}}(\mathbf{z})p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})$, with a prior distribution over latent space $p_{\boldsymbol{\theta}}(\mathbf{z})$, and a stochastic decoder $p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})$. The stochastic encoder $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})$, also called *inference model*, approximates the true but intractable posterior $p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x})$ of the generative model.

37

37

# Variational Lower Bound

- How can we estimate a VAE model?

$$\log p_{\boldsymbol{\theta}}(\mathbf{x}) = \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})}\left[\log p_{\boldsymbol{\theta}}(\mathbf{x})\right] \tag{2.5}$$

$$= \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})}\left[\log \left[\frac{p_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{z})}{p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x})}\right]\right] \tag{2.6}$$

$$= \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})}\left[\log \left[\frac{p_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{z})}{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})} \frac{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})}{p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x})}\right]\right] \tag{2.7}$$

$$= \underbrace{\mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})}\left[\log \left[\frac{p_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{z})}{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})}\right]\right]}_{=\mathcal{L}_{\boldsymbol{\theta},\boldsymbol{\phi}}(\mathbf{x})\ \text{(ELBO)}} + \underbrace{\mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})}\left[\log \left[\frac{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})}{p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x})}\right]\right]}_{=D_{KL}(q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})||p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x}))} \tag{2.8}$$

VAE with Gaussian noise.

Maximizing Evidence Lower Bound (ELBO) increases the log-likelihood of data and decreases the distance between the approximate prior and the true prior.

Let $z = \Sigma^{1/2}(x; \phi)\epsilon + \mu(x; \phi)$

$$\text{VLB} = \mathbb{E}_{\epsilon \sim \mathcal{N}(0,I)}\left[\log p_\theta(x|z) - \log q_\phi(z|x) + \log p(z)\right]$$

$$= \mathbb{E}_{\epsilon \sim \mathcal{N}(0,I)}\left[\log p_\theta(x|z)\right] - KL(q_\phi(z|x)||p(z))$$

$\nabla_\theta\,[\text{VLB}]$ and $\nabla_\phi\,[\text{VLB}]$ can now be efficiently computed with SGD.

38

38

Mega or Micro? Influencer Selection Using Follower Elasticity

Z Tian, R Dew, R Iyengar

Journal of Marketing Research, 2023 · journals.sagepub.com

Influencer marketing, in which companies sponsor social media personalities to promote their brands, has exploded in popularity in recent years. One common criterion for selecting an influencer partner is popularity. While some firms collaborate with "mega" influencers with millions of followers, other firms partner with "micro" influencers with only several thousand followers, but who also cost less to sponsor. To quantify this trade-off between popularity and cost, the authors develop a framework for estimating the follower elasticity of impressions (FEI), which measures a video's percentage gain in impressions (i.e., views) corresponding to a percentage increase in the number of followers of its creator. Computing FEI involves estimating the causal effect of an influencer's popularity on the view counts of their videos, which is achieved through a combination of (1) a unique data set collected from TikTok, (2) a representation learning model for quantifying video content, and (3) a machine learning–based causal inference method. The authors find that FEI is always positive, averaging .10, but often nonlinearly related to follower size. They examine the factors that predict variation in these FEI curves and show how firms can use these results to better determine influencer partnerships.

Sage Journals

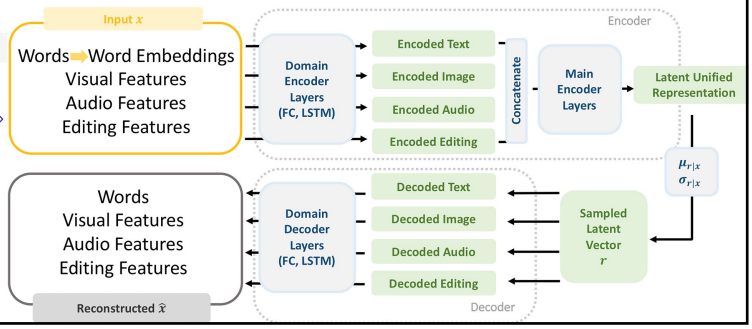
SHOW LESS

Save Cite Cited by 8 Related articles All 2 versions Web of Science: 1

## Follower Elasticity of Impressions for Influencers

- Construct Structured Multimodal VAE (SMVAE) to extract latent representations of TikTok videos.
- Use the latent representations and DeepIV to estimate the follower elasticity of impressions (FEI), i.e.,

$$\frac{\Delta\%\ of\ Impression\ Count}{\Delta\%\ of\ Follower\ Count}$$




39