Generative Models

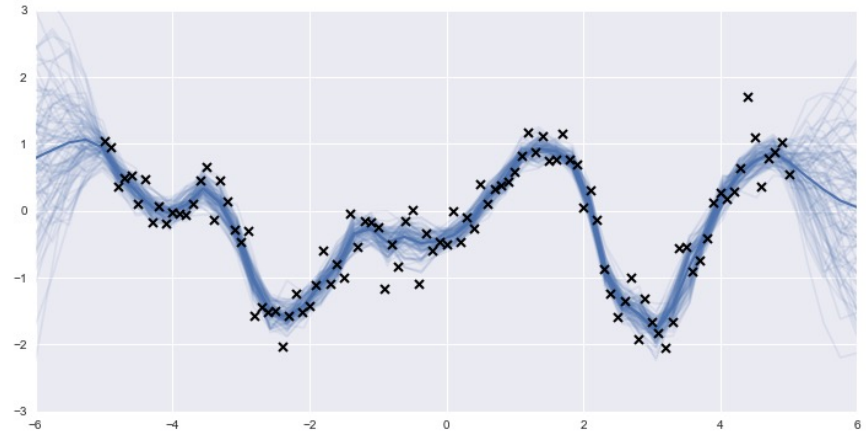
Introduction

Yingzhen Li (yingzhen.li@imperial.ac.uk)

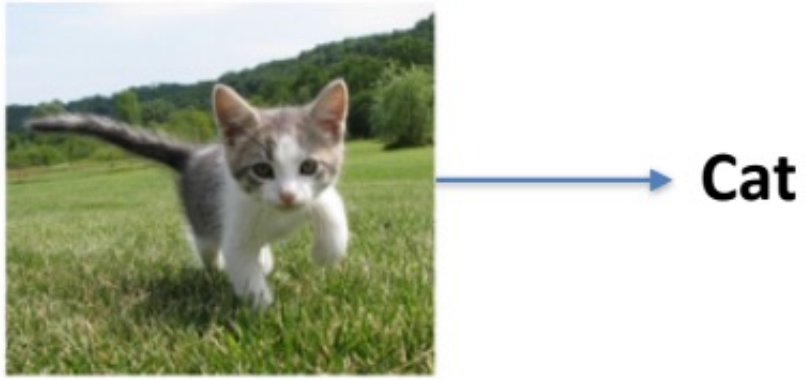
Supervised Learning

Data: $(x_1, y_1), \ldots, (x_N, y_N) \sim p_{data}(x, y)$

Goal: learn a function to map $x \rightarrow y$



Regression



Classification

Semantic Segmentation

Object detection

Most of the data are unlabelled

Unsupervised Learning

Data: $x_1, \dots, x_N \sim p_{data}(x)$ (no supervision signal)

Goal: inferring a function that describes the hidden structure of unlabelled data

Examples:

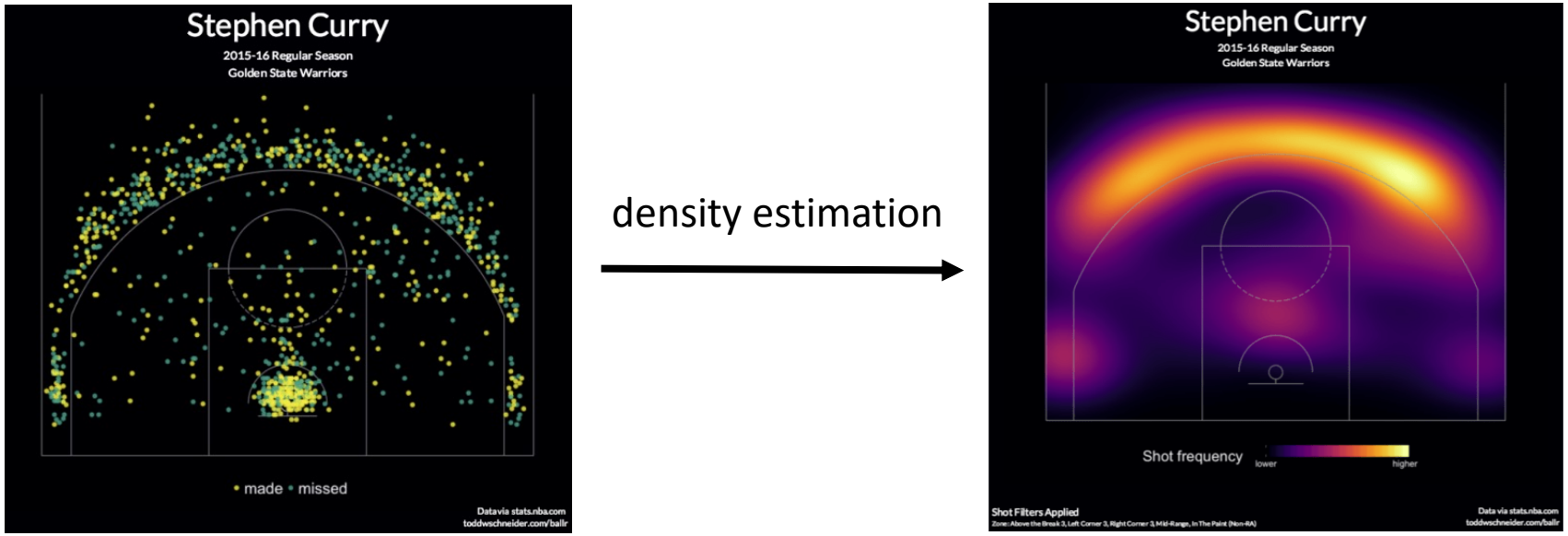
- Probability distribution/density estimation
- Dimensionality Reduction
- Clustering

All of them can be achieved by generative modelling!

Probability distribution/density estimation

Data: $x_1, \ldots, x_N \sim p_{data}(x)$

Goal: learn a distribution $p_\theta(x) \approx p_{data}(x)$ with data $x_1, \ldots, x_N$



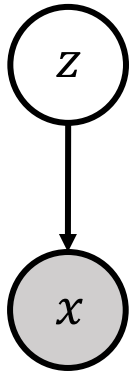
https://www.r-bloggers.com/2016/03/ballr-interactive-nba-shot-charts-with-r-and-shiny/

Generative Latent Variable Models

- Design $p_\theta(x)$ as a generative latent variable model (LVM):

$$z \sim p_\theta(z), \qquad x \sim p_\theta(x|z)$$

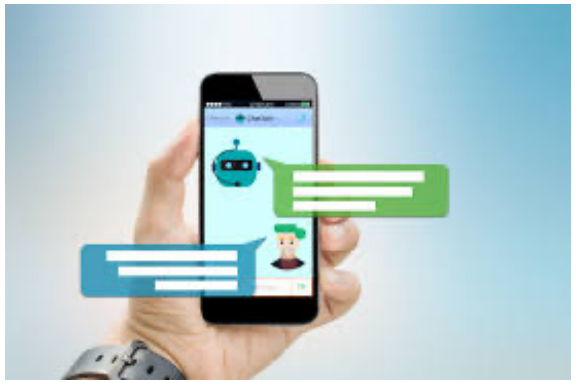
$$\Rightarrow\ p_\theta(x) = \int p_\theta(x|z)p_\theta(z)dz$$



z: latent variable (unobserved) x: observation variable

z: digit label, writing style, …
x: hand-written digit

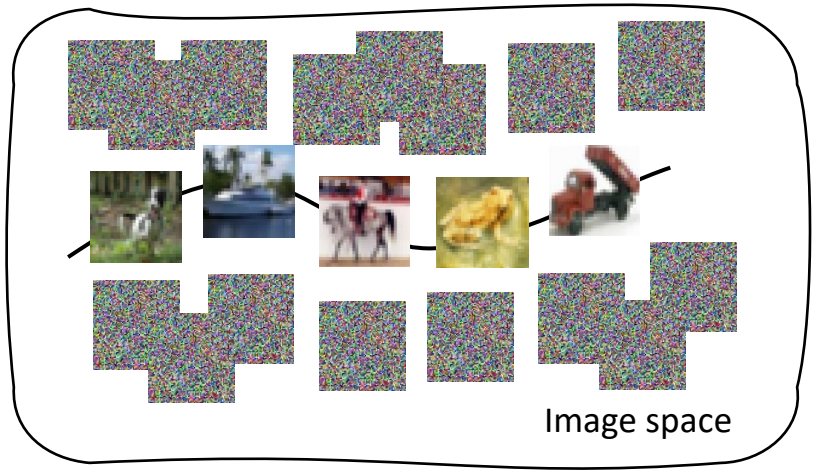
z: scene, viewing angle, lighting condition, …
x: photo image



z: semantics, sentiment, …
x: generated text

Dimensionality reduction

- High-dimensional raw data are often sparse, perhaps lying on a low-dimensional manifold:



natural images vs all RGB images

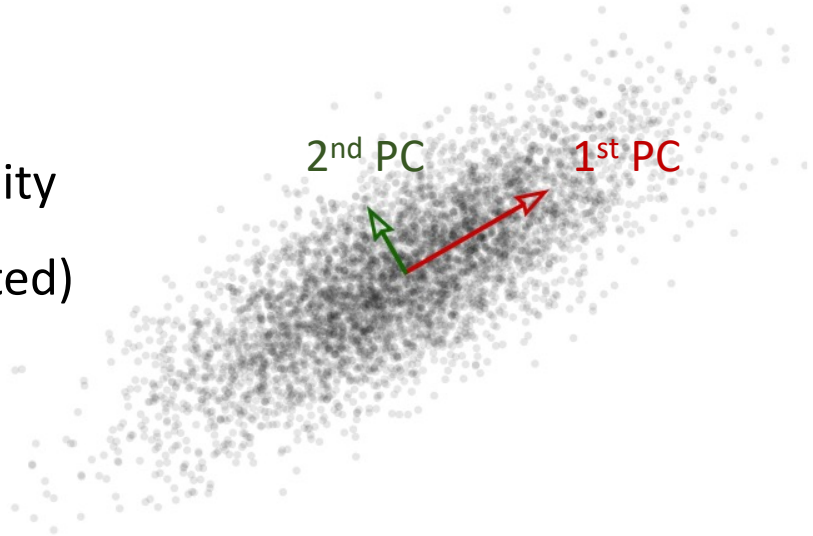
User ratings on items

Dimensionality reduction

- Principal Component Analysis (PCA):

  Find principal components – orthogonal directions that capture most of the variance in the data

  - 1st principal component – direction of greatest variability
  - 2nd principal component – next orthogonal (uncorrelated) direction of greatest variability
  - And so on ...



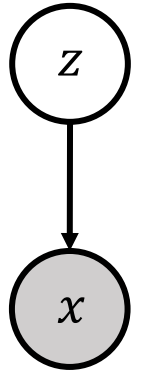
Dimensionality reduction is achieved by projecting the data on the top $K < d$ principal components ($x \in R^d$)

Dimensionality reduction

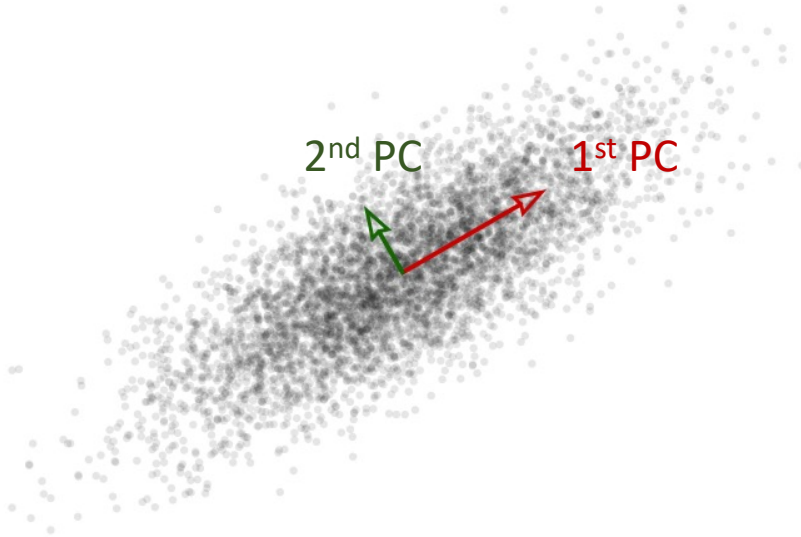
- Probabilistic Principal Component Analysis (Prob PCA):

$$p(z) = N(z; 0, I), \quad z \in R^K, \quad K < d$$

$$p_\theta(x|z) = N(x; Wz, \sigma^2 I), \quad x \in R^d$$



- Parameters to optimize: $\theta = W \in R^{d \times K}$, with the row vectors in W orthogonal to each other
- Trained using Maximum Likelihood
- Optimal W contains the top K principal components – the top K eigenvectors of the data covariance matrix

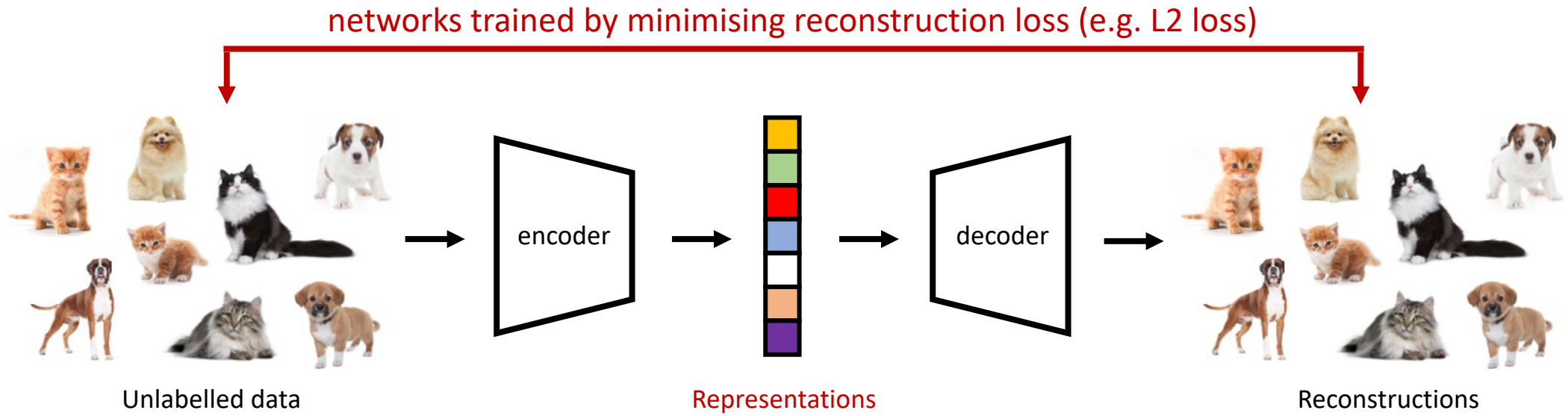


Tipping and Bishop. Probabilistic principal component analysis. Journal of the Royal Statistical Society: Series B, 1999

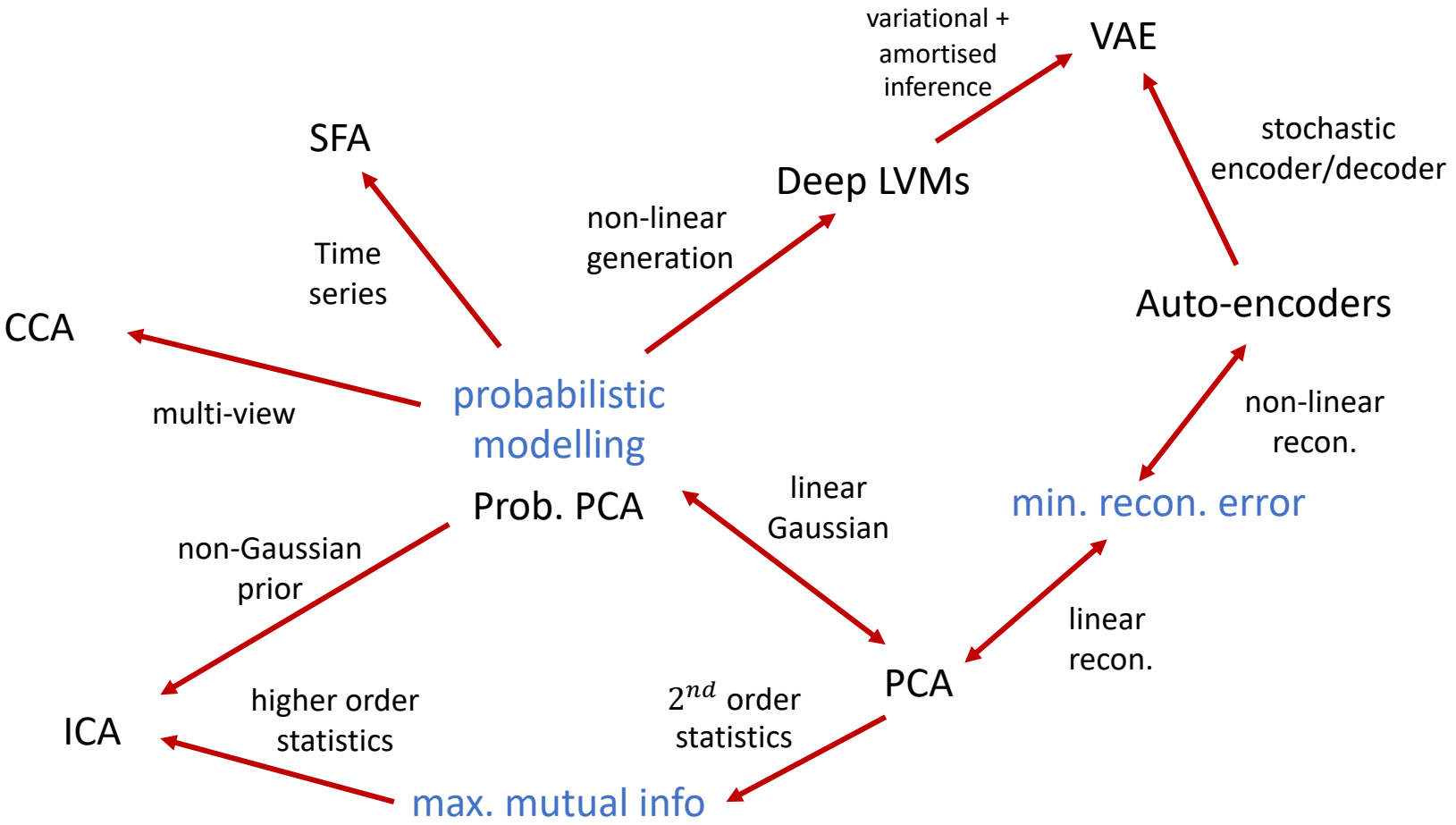
Dimensionality reduction

- Auto-encoders for dimensionality reduction:
  - Encoder network to extract data representations (often with lower dimensionality)
  - Decoder network to reconstruct data given the representations

A probabilistic version: Variational auto-encoder (VAE)



Dimensionality reduction



Clustering

- Clustering: discover “group structure”
  - grouping datapoints into several clusters
  - Datapoints in the same cluster are similar
  - Datapoints in different clusters are “dissimilar”

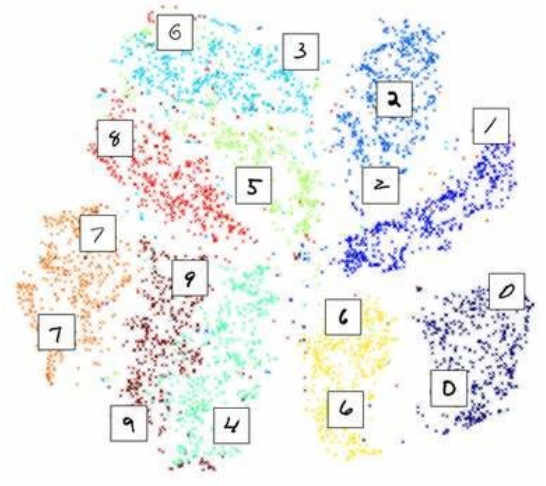
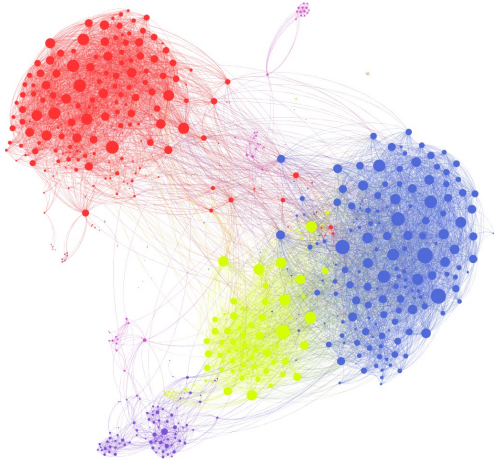
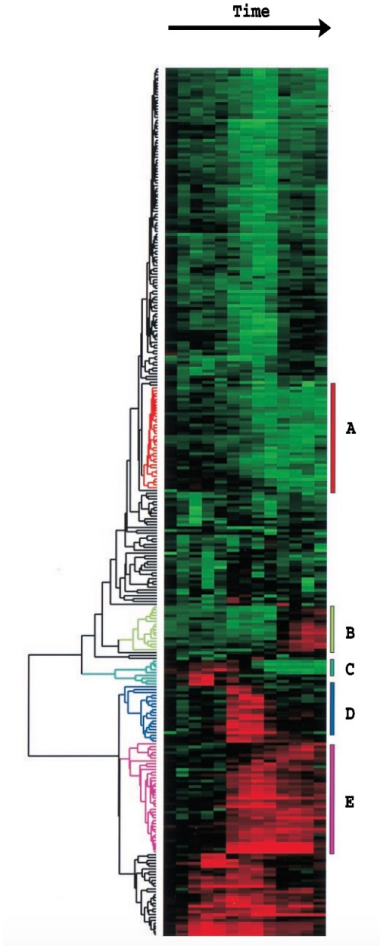


Image clustering



social network analysis



gene data analysis

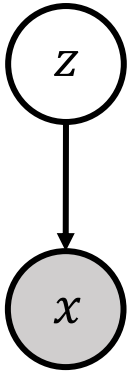
Clustering

- Gaussian mixture model (GMM):

$$p_\theta(z) = Categorical(\pi),$$

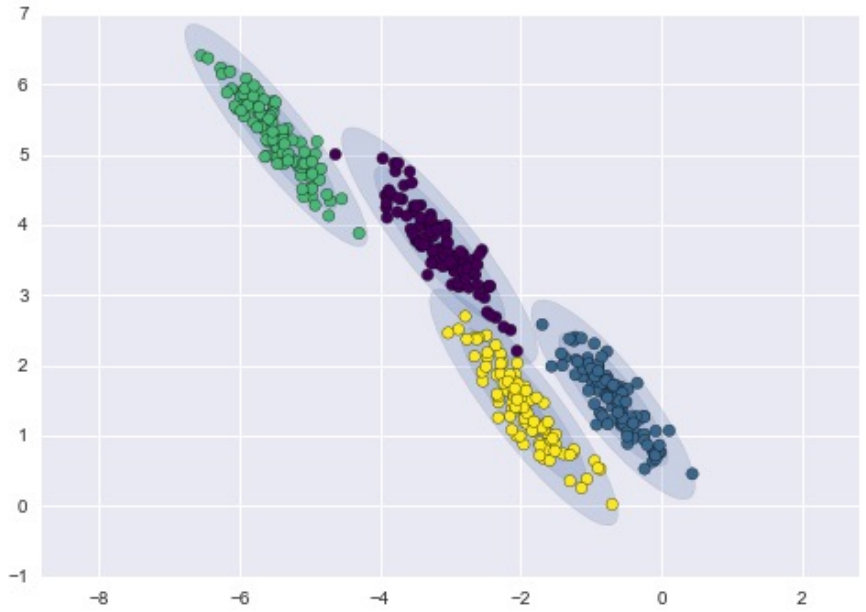
$$\pi = (\pi_1, \dots, \pi_K), \pi_i = p_\theta(z = i), \sum_{i=1}^{K} \pi_i = 1$$

$$p_\theta(x|z) = N(x; \mu_z, \Sigma_z)$$



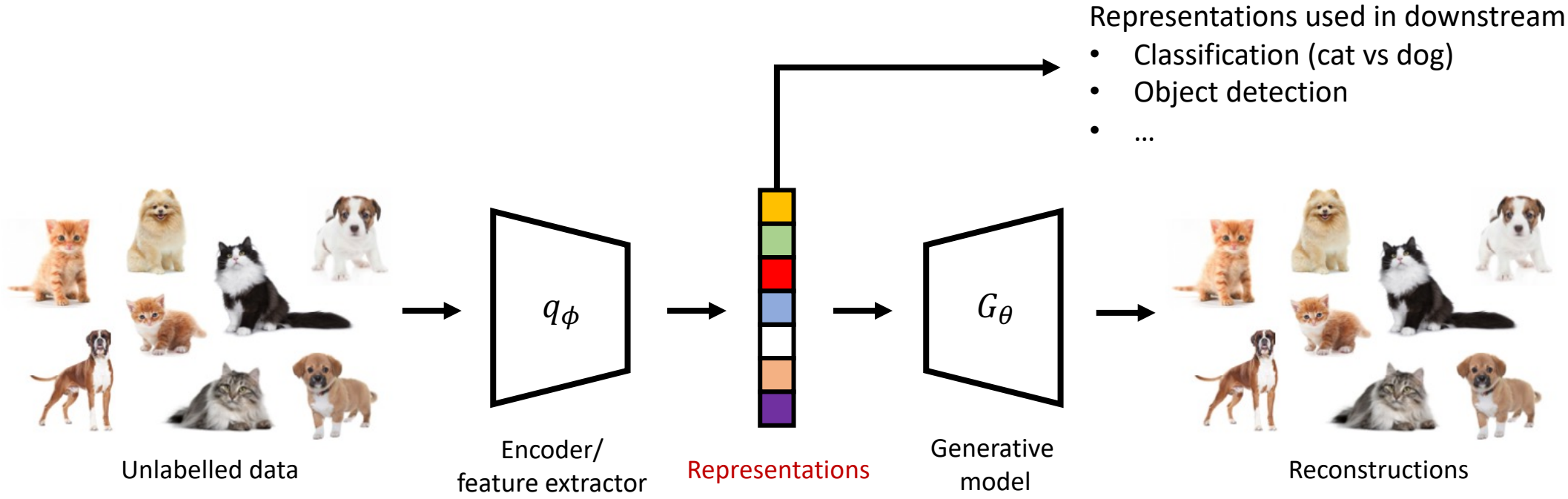
- $z \in \{1, \dots, K\}$: index of the Gaussian component
- μ_z: mean of the i^{th} Gaussian component if $z = i$
- Σ_z: Covariance matrix of the i^{th} Gaussian component if $z = i$

⇒ Clustering can be done by fitting a GMM model to the data



Representation learning

- Both dimensionality reduction and clustering can be viewed as representation learning
  - Hope: useful for downstream tasks



Taxonomy of Generative Models