AA 598B Special Topics

Decision-Making & Control for Safe Interactive Autonomy

Instructor: Prof. Karen Leung

Autumn 2024

https://faculty.washington.edu/kymleung/aa598/

Announcements

- Next week we have long paper summaries (~35 minutes each)
  - [**Paper #1**] Kretzschmar et al 2016 Socially compliant navigation via IRL
    - Shubham Mittal, Anubhav Vishwakarma
  - [**Paper #2**] Rhinehart et al 2019 Goal-conditioned deep multi-agent trajectory forecasting
    - Ethan Pronovost, Ruiqi Li(?) (sign up sheet)
  - Submit slides to canvas afterwards

- **Rest of class**: Short paper summaries (2 x 5%)
  - PACES, ½ - 1 page each
  - Submit on Canvas
  - 7 opportunities for short paper/talk summaries, maximum 30% of grade)

Last time

- Start of the prediction module
  - Behavior prediction for HRI
  - Types of prediction models
    - Ontological vs phenomenological

Today

- Continue with ontological vs phenomenological
- Deep generative models
  - Latent space models and CVAEs
  - Sketch-RNN demo!
- Current landscape of prediction models
- Quickfire paper summary

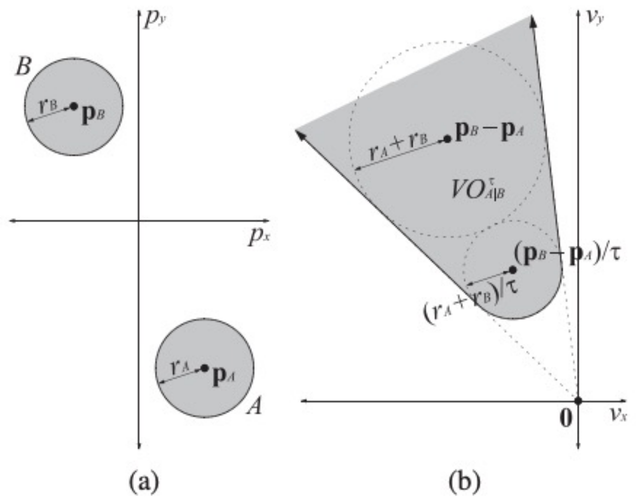
Ontological vs Phenomenological approaches

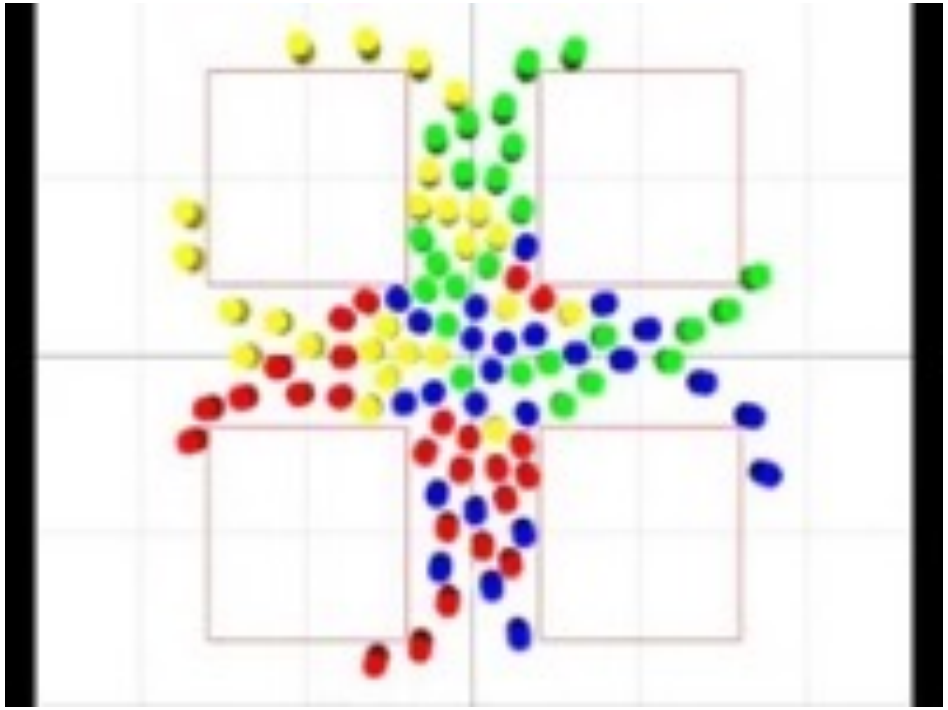
https://ai.stanford.edu/blog/trajectory-forecasting/

- **Ontological**: “Theory of mind”, ”first principles”, explicit model that describes the interactions in a very interpretable way
  - Social forces model (SFM) [Helbing & Molnar 1998]
  - Optimal reciprocal collision avoidance (ORCA) [van den Berg et al 2011]
  - Intelligent driver model (IDM) [Treiber, Hennecke, Helbing, 2000]
  - MOBIL lane changing models [Kesting, Treiber, Helbing 2006]
  - Game theory [von Neumann 1928]
  - Optimal control, assume agents are optimal planners
  - Rationality model (maximum entropy inverse reinforcement learning) [Ziebart et al 2008, Levine & Koltun 2012, Sadigh et al 2016]

Optimal Reciprocal Collision Avoidance

- Variation on velocity obstacles, reciprocal collision avoidance,…
- Main idea: Agents change direction if maintaining constant velocity leads to collision.





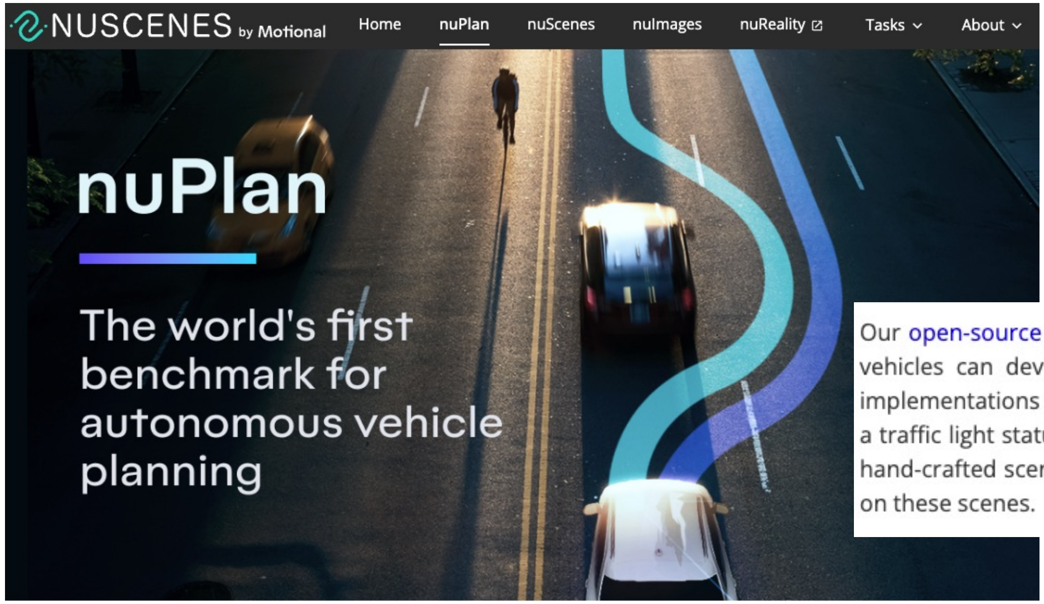
https://gamma.cs.unc.edu/ORCA/publications/ORCA.pdf

Intelligent Driver Model

$$\frac{dv}{dt} = a\left[1 - \left(\frac{v}{v_0}\right)^{\delta} - \left(\frac{s^*(v,\Delta v)}{s}\right)^2\right]$$

where

$$s^*(v,\Delta v) = s_0 + \max\left[0,\ \left(vT + \frac{v\Delta v}{2\sqrt{ab}}\right)\right]$$



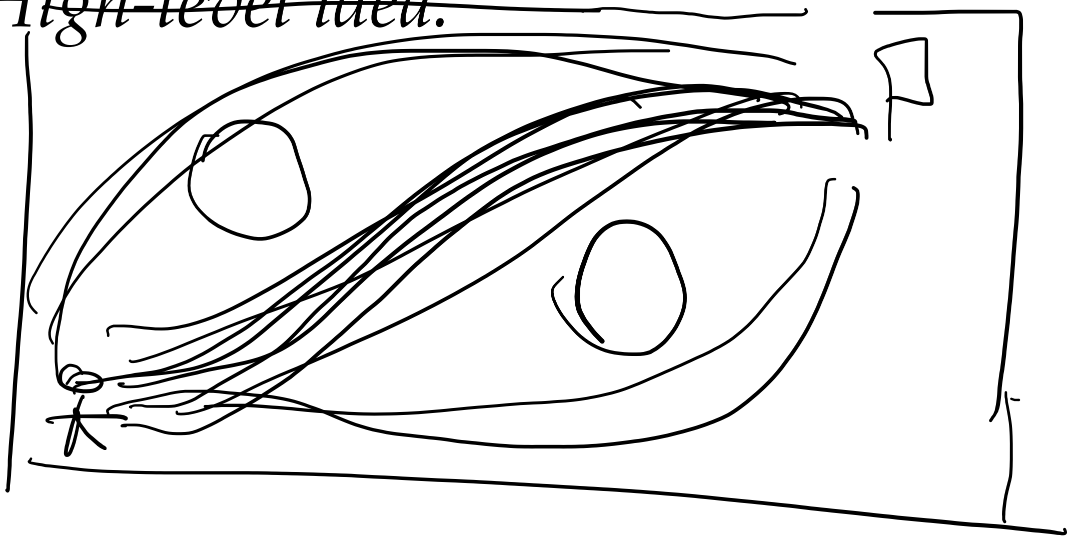
Our open-source simulation framework supports closed-loop and open-loop simulation. Closed-loop means that the ego vehicle and other agent vehicles can deviate from what was recorded in the original log. Besides baseline implementations for planners, we also provide baseline implementations for traditional (Intelligent Driver Model, IDM) and ML-based smart agents. To realistically simulate the flow of traffic, we developed a traffic light status inference system, that infers the status of a traffic light from the observed motion of vehicles in the scene. Finally we mine for hand-crafted scenarios (e.g. lane change, pedestrian-car interaction) to find interesting scenes and evaluate common and scenario-specific metrics on these scenes.

https://traffic-simulation.de/info/info_IDM.html

Maximum entropy inverse RL (MaxEnt-IRL)

- Given noisily optimal demonstrations, what was the reward function that explains the observed demonstrations
  - Assumes human is *noisily rational*

High-level idea.



Maximum entropy inverse RL (MaxEnt-IRL)

$$\max_{P} \int -P(\xi)\log P(\xi)d\xi$$

Entropy: not favoring any trajectory over any other

$$\text{s.t.} \quad \mathbb{E}_{\xi\sim P(\xi)}[f(\xi)] = \int P(\xi)f(\xi)d\xi = f_D$$

Feature matching: features from learned distribution should match the data

$$\int P(\xi)d\xi = 1$$

$$P(\xi) \geq 0,\ \forall \xi \in \Xi$$

Valid probability distribution

Maximum entropy inverse RL (MaxEnt-IRL)

- We define the features $r(\xi) = \sum_i \lambda_i f_i(\xi)$ linear combination of features
- Solving the optimization problem yield…

$$P(\xi \mid \lambda) = \frac{e^{\lambda^T f(\xi)}}{\int e^{\lambda^T f(\hat{\xi})} d\hat{\xi}}$$

$$P(\xi \mid \lambda) \propto e^{r(\xi)}$$

Advantages of ontological methods

(Discussion)

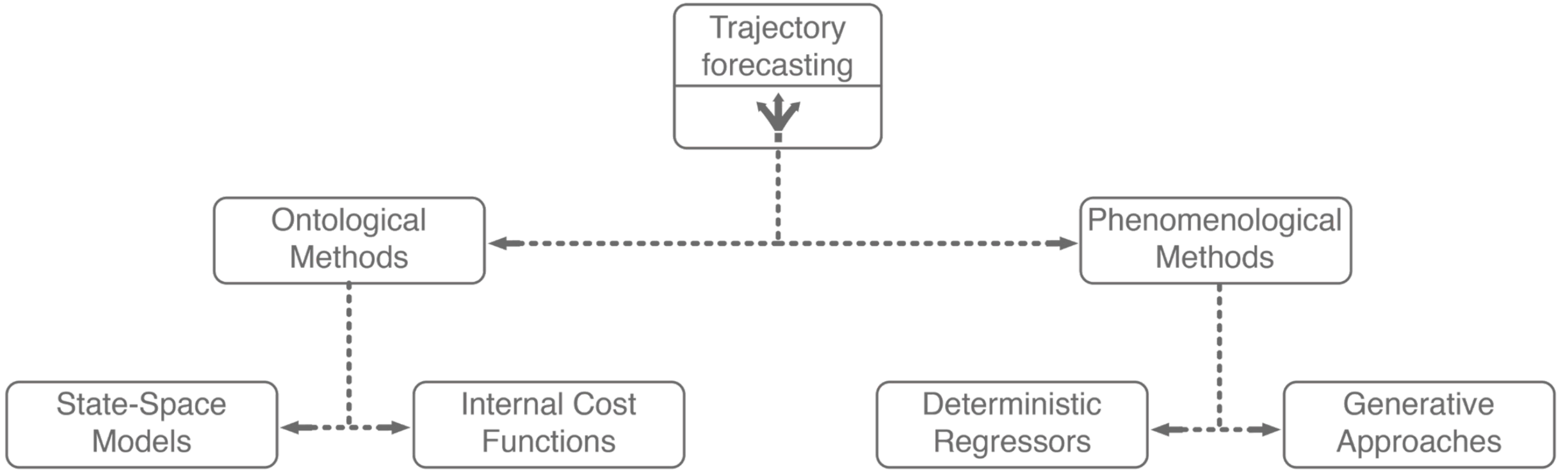
- Potentially better generalization capabilities in new/out of distribution scenarios.
- Interpretable!
- Simple to work with

- Limited expressivity
- Not good at capturing uncertainty

Ontological vs Phenomenological approaches

https://ai.stanford.edu/blog/trajectory-forecasting/

- **Phenomenological**: make minimal assumptions about the structure of agents' decision-making process.
  - Rely on powerful general modeling techniques (i.e., DNN) and a wealth of observation data
  - Regression or generative modeling



https://ai.stanford.edu/blog/trajectory-forecasting/

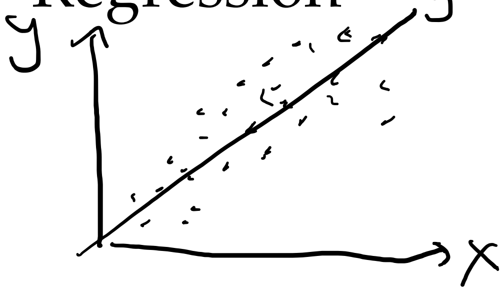
Quick review: ML

- Supervised learning: each datapoint has a corresponding label; $(x, y) \in D$

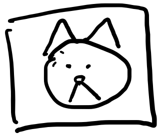
  f?

  $$\min_f \frac{1}{N}\sum_{i=1}^{N} \|f(x_i) - y_i\|_2^2 \quad \text{MSE}$$

  - Regression $y = f(x)$



  - Classification

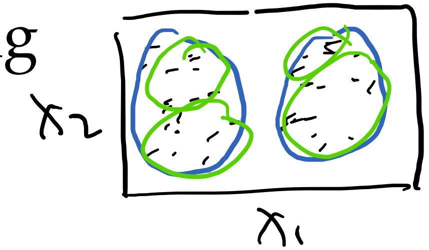


$\xrightarrow{f}$

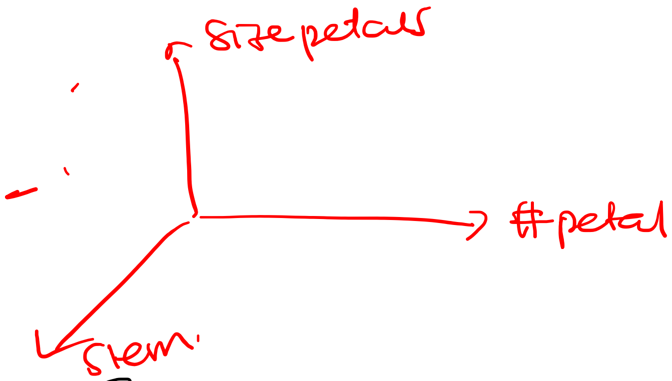
fish	0	0.1
$\boxed{\text{cat}}$	1	0.9
dog	0	0.05
rabbit	0	0.05

Quick review: ML

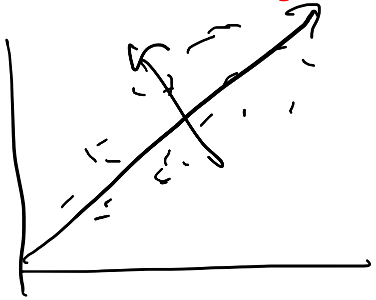
- Unsupervised learning: each datapoint does not have a corresponding label; $x \in D$
  - Clustering



flowers



  - Principal Component Analysis



  - Generative modeling

Quick review: ML

- Self-supervised: Combined with some sort of auto-labeling technique
- Semi-supervised learning: Only some data is labeled

Regression

- ”Line of best fit”
- Find parameters of a function that best fits the data
- E.g., Linear least squares
- E.g., Learning dynamics from data

x_t, x_{t+1}, u_t

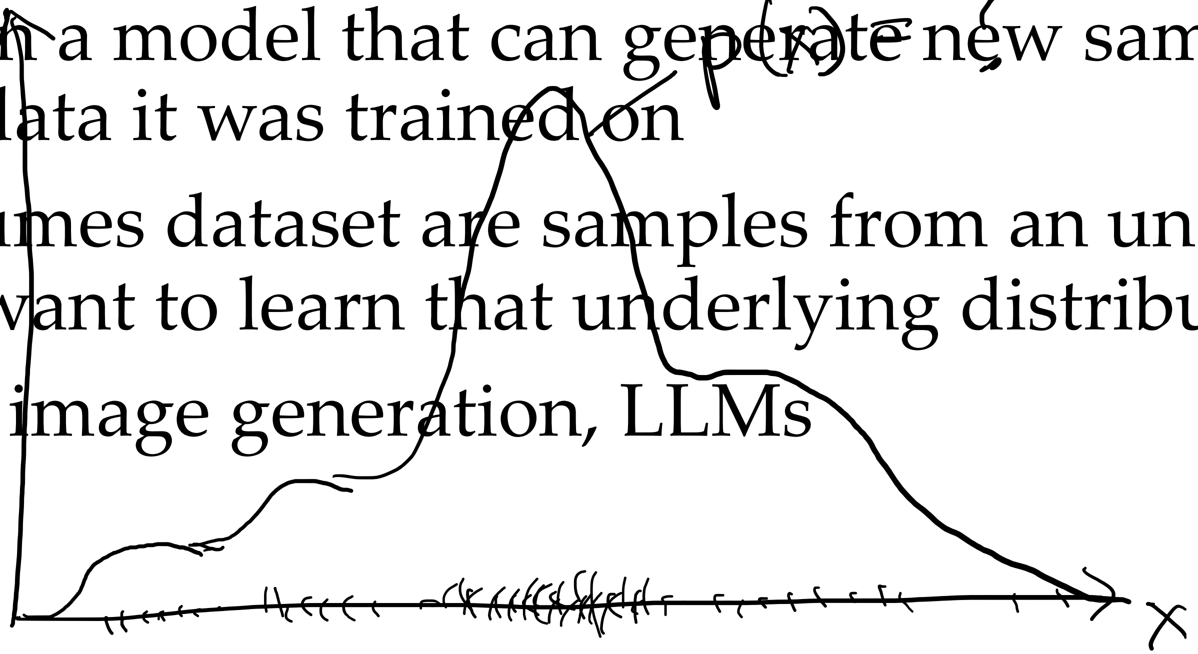
learn model $x_{t+1} = f(x_t, u_t)$

$$\min_{\theta} \frac{1}{N} \sum_{i=1}^{N} \text{error}(f(x_i), y_i) + \text{reg}(\theta)$$

Generative models

$p(x) = ?$ or $p(y|x)$ for conditional distributions

- Learn a model that can generate new samples that resembles the data it was trained on
- Assumes dataset are samples from an underlying distribution. We want to learn that underlying distribution
- E.g., image generation, LLMs

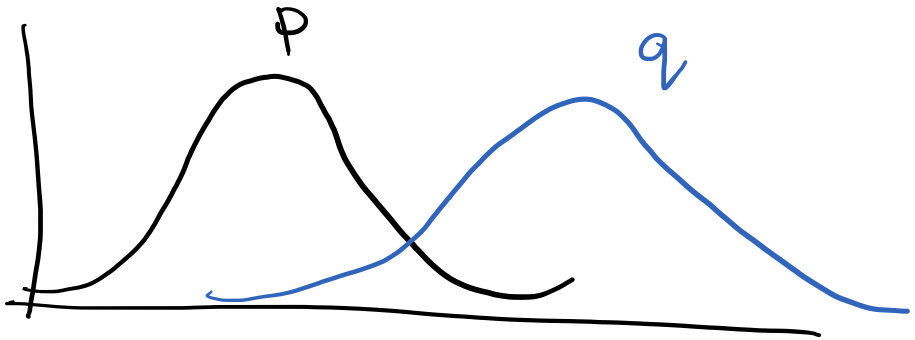


CS236

https://deepgenerativemodels.github.io/

Generative models

propose a distribution $p_\theta(x)$, θ parameters of distributions



divergence metrics

$$\max_\theta \sum_{i=1}^{N} \log p_\theta(x_i)$$

Maximum log-likelihood

$$\min_\theta D_{\mathrm{KL}}(p_{\mathrm{data}}(x) || p_\theta(x))$$

Minimize KL-divergence

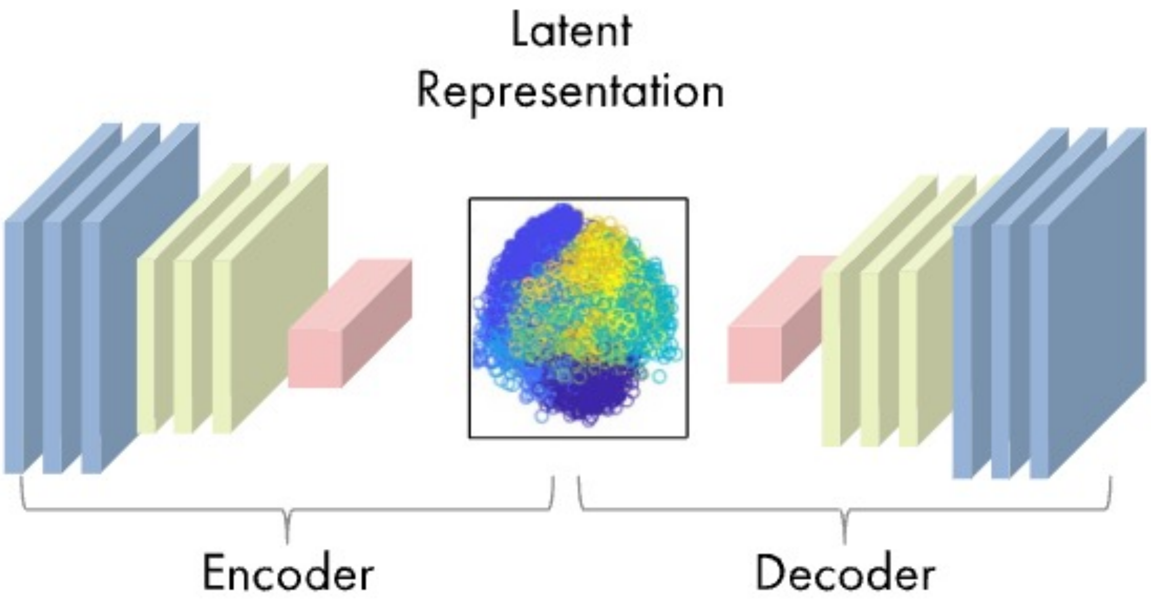
https://deepgenerativemodels.github.io/

Types of generative models

- **Variational autoencoders:** We will discuss more
- **Normalizing flows**: “morph” a gaussian distribution into something more complicated
- **Generative Adversarial Networks (GANs):** Generator and discriminator. G tries to fool the D.
- **Energy-based models (EBMs)**: Learn an energy function that assigns ”how good” that point is, and then we can sample from the energy landscape
- **Diffusion models**: Learn to denoise from noise

Autoencoders

High dimensional data

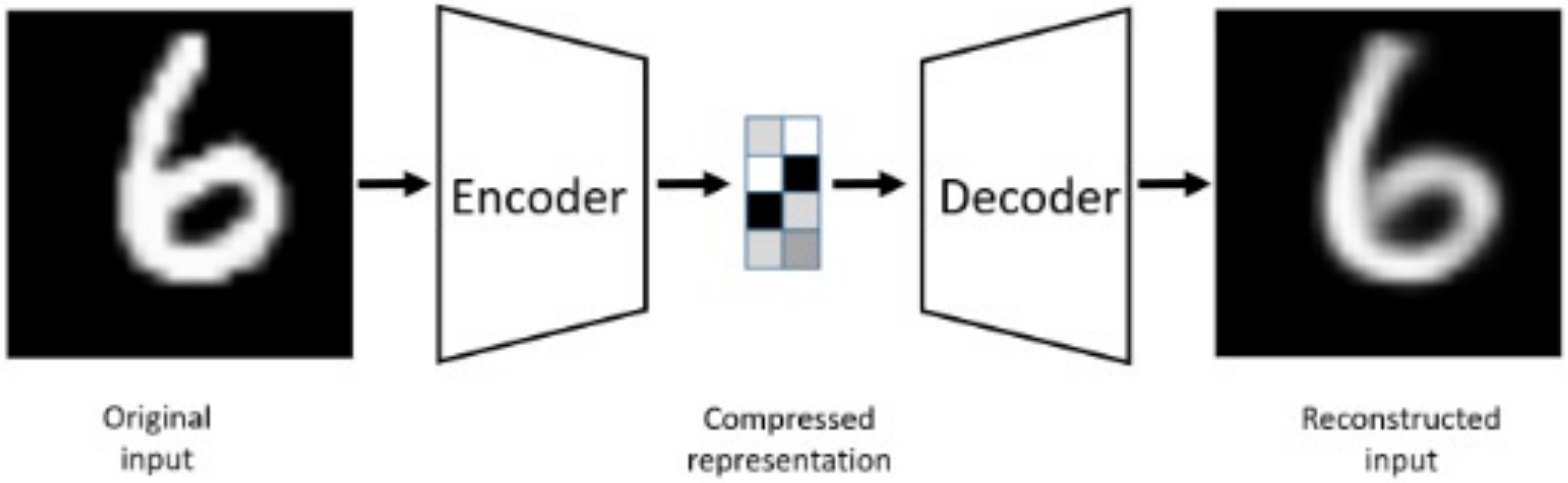


Reconstruct input

Enforce a bottleneck via a
low-dimensional layer

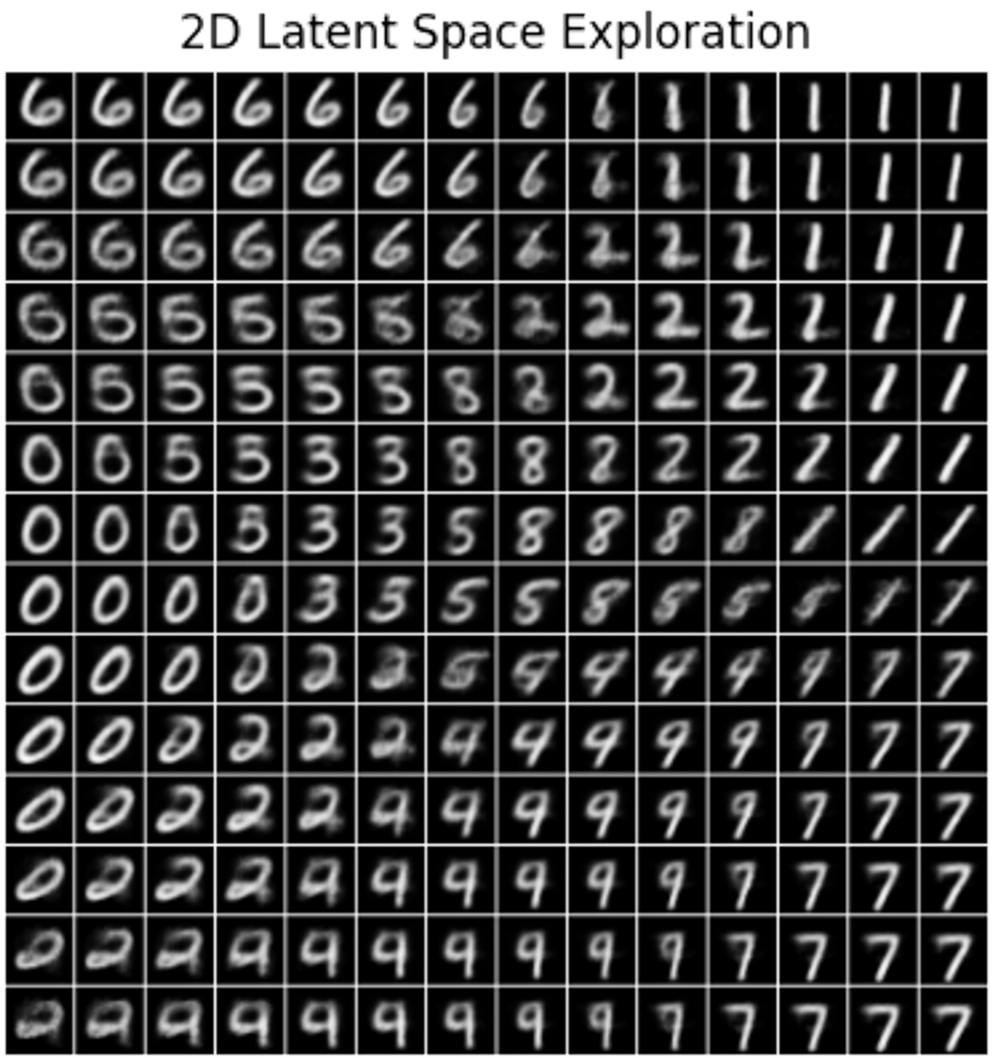
An approach to perform clustering, especially on high-dimensional data

Image compressions



Murat et al 2023

MNIST dataset



https://arxiv.org/abs/1312.6114

https://kikaben.com/vae-2013/

Conditional Variational Autoencoders

want to learn $\rightarrow$ $$\boxed{p(y \mid x)} = \int \underbrace{p(y \mid x, z)p(z \mid x)}_{p(y,z|x)} dz$$

z: latent variable

$$\max_{\theta} \sum_{i=1}^{N} \log p_\theta(y_i \mid x_i) \Rightarrow \max_{\phi,\psi} \sum_{i=1}^{N} \underbrace{\int p_\phi(y_i \mid x_i, z) p_\psi(z \mid x_i) dz}_{\text{intractable to compute}}$$

Evidence lower bound $$\log p(y \mid x) \geq \underbrace{\mathbb{E}_{q(z|x,y)}[\log p(y \mid x, z)] - D_{\mathrm{KL}}(q(z \mid x, y) \mid p(z \mid x)])}_{\text{ELBO}}$$

Robot conditioned human trajectory predictor

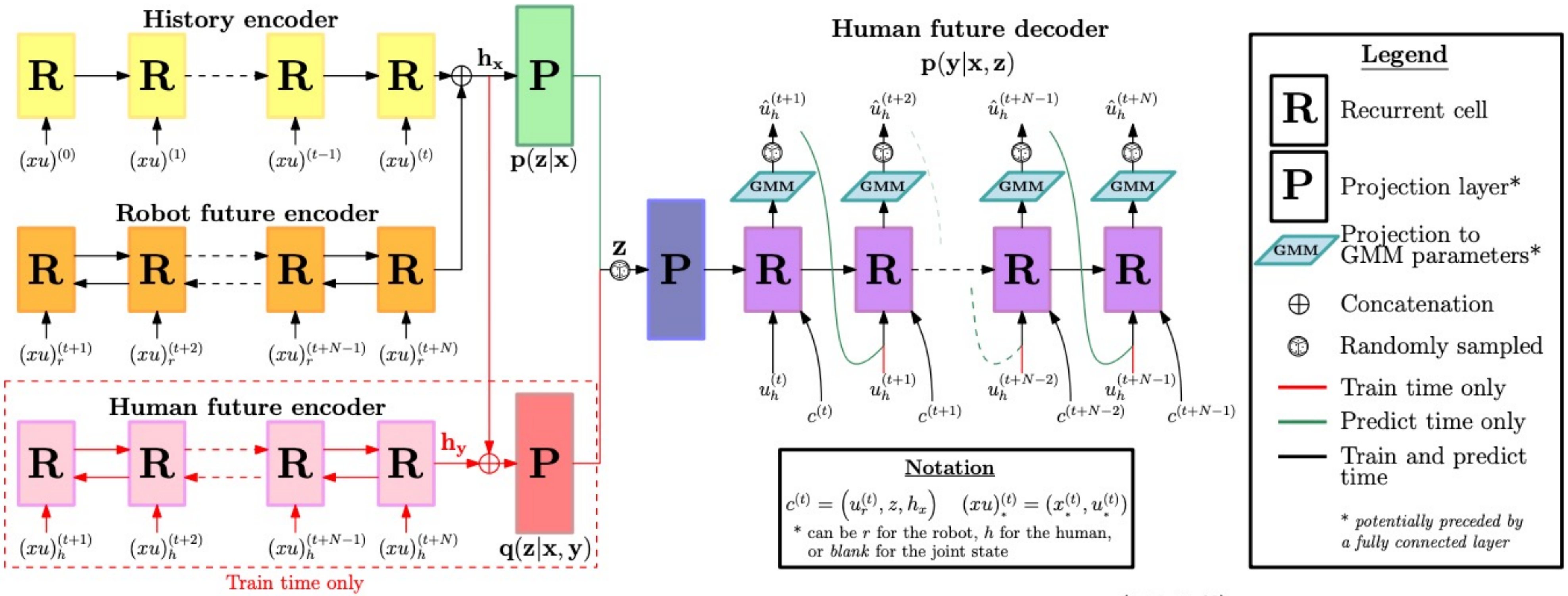


Fig. 2. CVAE architecture for sequence-to-sequence generative modeling of future human actions $\mathbf{y} = u_h^{(t+1:t+N)}$ conditioned on joint interaction history $(x^{(0:t)}, u^{(0:t)})$ and candidate robot future actions $u_r^{(t+1:t+N)}$ (together, $\mathbf{x}$). The random variable $\mathbf{z}$ is a latent mixture component index.

Schmerling et al 2018

CVAE demo: Sketch RNN

- https://magenta.tensorflow.org/sketch-rnn-demo