DSC291: Machine Learning with Few Labels

"Standard Model" of ML

Zhiting Hu

Lecture 24, May 31, 2024

UC San Diego
HALICIOĞLU DATA SCIENCE INSTITUTE

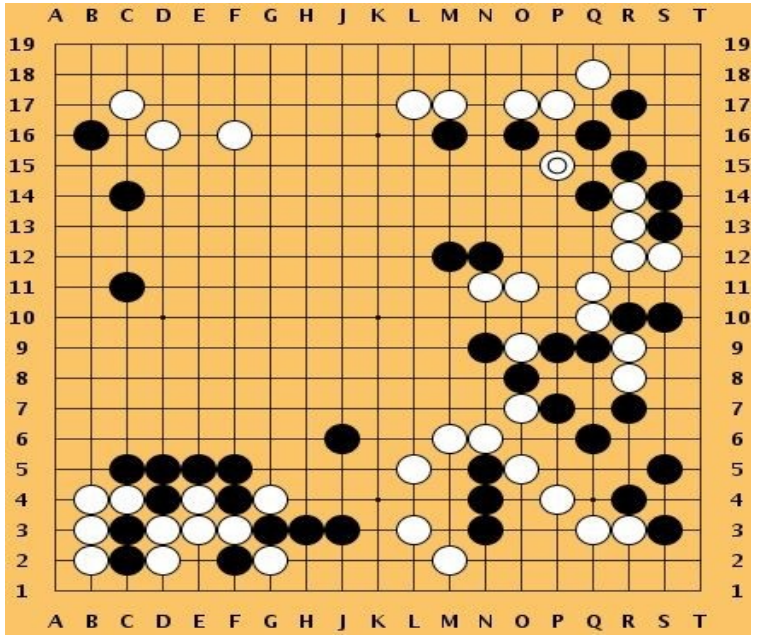
This Lecture

- "Standard Model" (20mins)

- Presentation #1 (10mins):
  - Shanglin Zeng, What is important about the No Free Lunch theorems?
- Presentation #2 (10mins):
  - Jiatu Li, TBD
- Presentation #3 (10mins):
  - Jiaxian Xiang, ChineseBERT: Chinese Pretraining Enhanced by Glyph and Pinyin Information

Google form for presentation questions and feedback:

Policy gradients: AlphaGo



Overview:

- Mix of supervised learning and reinforcement learning
- Mix of old methods (Monte Carlo Tree Search) and recent ones (deep RL)

How to beat the Go world champion:

- Featurize the board (stone color, move legality, bias, …)
- Initialize policy network with supervised training from professional go games, then continue training using policy gradient (play against itself from random previous iterations, +1 / -1 reward for winning / losing)
- Also learn value network (critic)
- Finally, combine policy and value networks in a Monte Carlo Tree Search algorithm to select actions by lookahead search

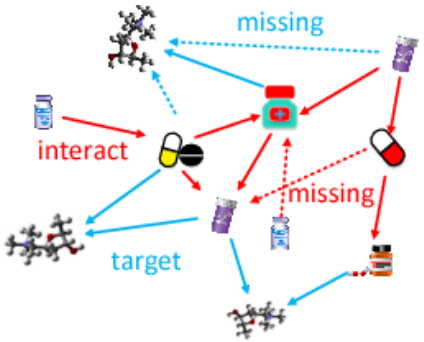
[Silver et al., Nature 2016]

Experience of all kinds

Data examples

Type-2
diabetes is 90%
more common
than type-1

Rules/Constraints

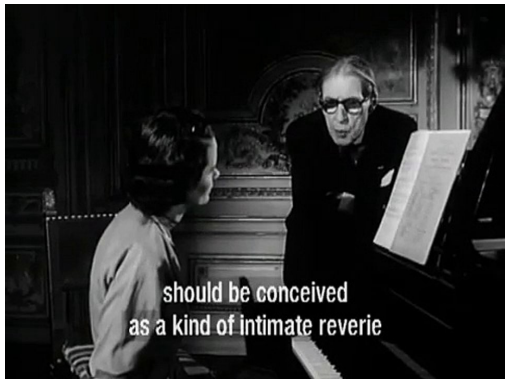


Knowledge graphs

Rewards

Auxiliary agents

Adversaries

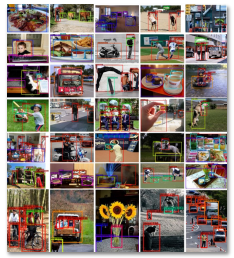


Master classes

…

- *And all combinations of such*
- *Interpolations between such*
- *…*

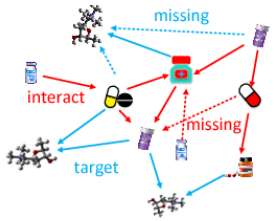
Human learning vs machine learning



Data examples

Type-2 diabetes is 90% more common than type-1

Rules/Constraints



Knowledge graphs

Rewards

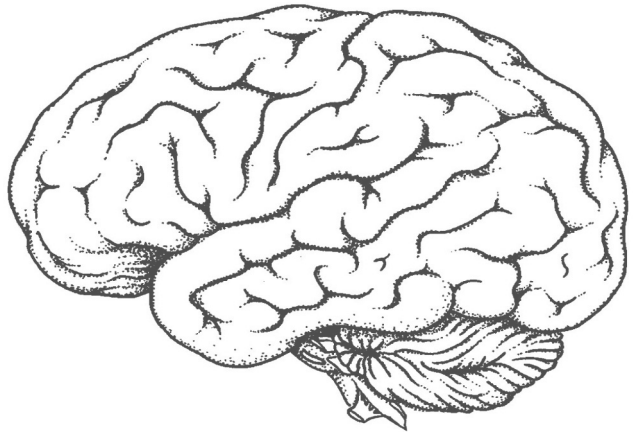
Auxiliary agents

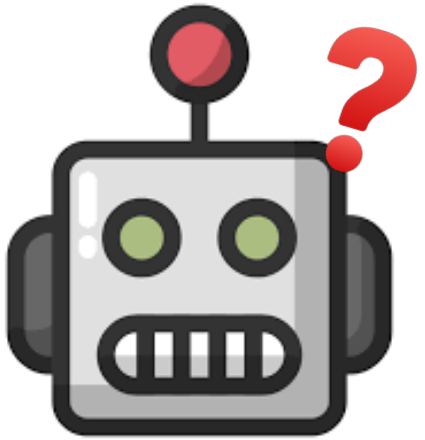
Adversaries

Master classes

…

- *And all combinations of such*
- *Interpolations between such*
- *…*





The zoo of ML/AI models

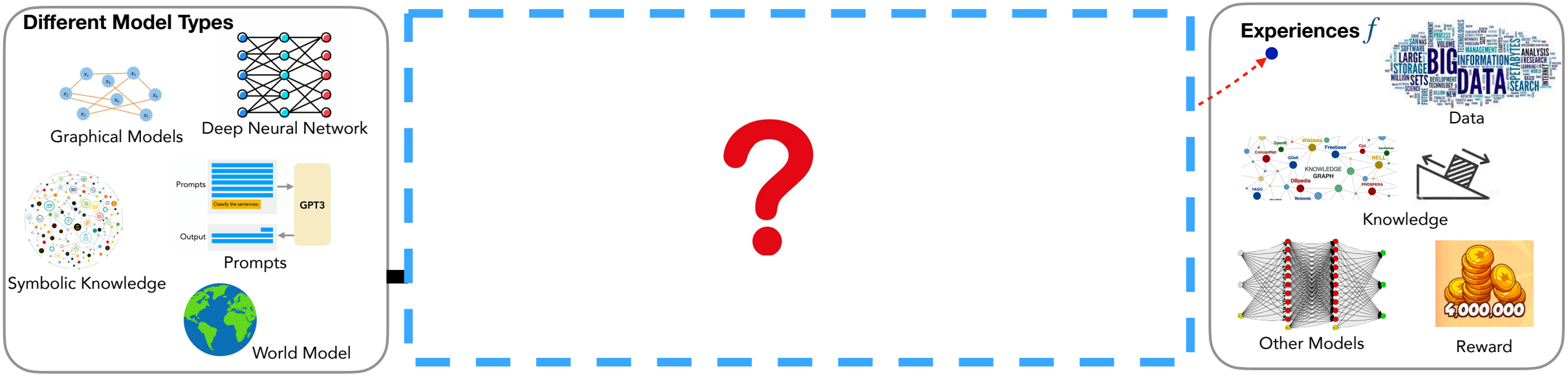
- Neural networks
  - Convolutional networks
  - AlexNet, GoogleNet, ResNet
  - Recurrent networks, LSTM
  - Transformers
  - BERT, GPTs

- Graphical models
  - Bayesian networks
  - Markov Random fields
  - Topic models, LDA
  - HMM, CRF

- Kernel machines
  - Radial Basis Function Networks
  - Gaussian processes
  - Deep kernel learning
  - Maximum margin
  - SVMs

- Decision trees
- PCA, Probabilistic PCA, Kernel PCA, ICA
- Boosting

The zoo of ML/AI algorithms



$$\min_\theta \mathcal{L}(\theta, \mathcal{E})$$

Optimization solver

Loss

Model architecture

Experience

The zoo of ML/AI algorithms

maximum likelihood estimation

reinforcement learning as inference

data re-weighting

inverse RL

policy optimization

active learning

data augmentation

actor-critic

reward-augmented maximum likelihood

label smoothing

softmax policy gradient

imitation learning

adversarial domain adaptation

posterior regularization

GANs

constraint-driven learning

knowledge distillation

intrinsic reward

generalized expectation

prediction minimization

regularized Bayes

learning from measurements

energy-based GANs

weak/distant supervision

Standard Model in Physics

Diverse electro-magnetic theories

Maxwell's Eqns: original form

$e+\frac{df}{dx}+\frac{dg}{dy}+\frac{dh}{dz}=0$	(1)	Gauss' Law
$\mu\alpha=\frac{dH}{dy}-\frac{dG}{dz}$ $\mu\beta=\frac{dF}{dz}-\frac{dH}{dx}$ $\mu\gamma=\frac{dG}{dx}-\frac{dF}{dy}$	(2)	Equivalent to Gauss' Law for magnetism
$P=\mu\left(\gamma\frac{dy}{dt}-\beta\frac{dz}{dt}\right)-\frac{dF}{dt}-\frac{d\Psi}{dz}$ $Q=\mu\left(\alpha\frac{dz}{dt}-\gamma\frac{dx}{dt}\right)-\frac{dG}{dt}-\frac{d\Psi}{dy}$ $R=\mu\left(\beta\frac{dx}{dt}-\alpha\frac{dy}{dt}\right)-\frac{dH}{dt}-\frac{d\Psi}{dz}$	(3)	Faraday's Law (with the Lorentz Force and Poisson's Law)
$\frac{d\gamma}{dy}-\frac{d\beta}{dz}=4\pi p'$ $\quad p'=p+\frac{df}{dt}$ $\frac{d\alpha}{dz}-\frac{d\gamma}{dx}=4\pi q'$ $\quad q'=q+\frac{dg}{dt}$ $\frac{d\beta}{dx}-\frac{d\alpha}{dy}=4\pi r'$ $\quad r'=r+\frac{dh}{dt}$	(4)	Ampère-Maxwell Law
$P=-\xi p \quad Q=-\xi q \quad R=-\xi r$		Ohm's Law
$P=kf \quad Q=kg \quad R=kh$		The electric elasticity equation ($\mathbf{E}=\mathbf{D}/\varepsilon$)
$\frac{de}{dt}+\frac{dp}{dx}+\frac{dq}{dy}+\frac{dr}{dz}=0$		Continuity of charge

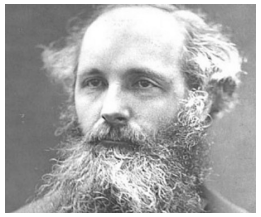
Simplified w/ rotational symmetry

$$\nabla\cdot\mathbf{D}=\rho_V$$

$$\nabla\cdot\mathbf{B}=0$$

$$\nabla\times\mathbf{E}=-\frac{\partial\mathbf{B}}{\partial t}$$

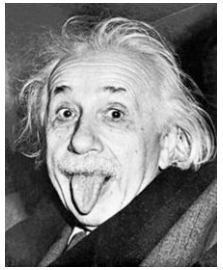
$$\nabla\times\mathbf{H}=\frac{\partial\mathbf{D}}{\partial t}+\mathbf{J}$$



Further simplified w/ symmetry of special relativity

$$\varepsilon^{uvk\lambda}\partial_v F_{k\lambda}=0$$

$$\partial_v F^{uV}=\frac{4\pi}{c}j^u$$

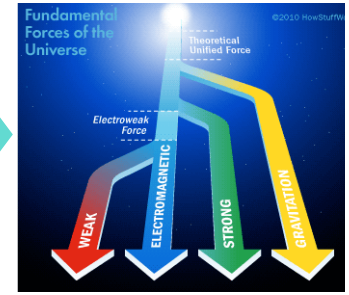


Standard Model w/ Yang-Mills theory and US(3) symmetry

$$\mathcal{L}_{\text{gf}}=-\frac{1}{2}\,\text{Tr}(F^2)$$

$$=-\frac{1}{4}F^{a\mu\nu}F^a_{\mu\nu}$$

Unification of fundamental forces?



1861 — 1910s — 1970s — ?

Toward a 'Standard Model' of Machine Learning

Zhiting Hu[†,*], Eric P. Xing[‡,◇,♮,**]

[†] Halıcıoğlu Data Science Institute, University of California San Diego, San Diego, USA

[‡] Machine Learning Department, Carnegie Mellon University, Pittsburgh, USA

[♮] Mohamed bin Zayed University of Artificial Intelligence, Abu Dhabi, UAE

[◇] Petuum Inc., Pittsburgh, USA

[Hu & Xing, Harvard Data Science Review, 2022]: https://arxiv.org/abs/2108.07783

$$\min_{q,\theta} \; -\mathbb{E} + \mathbb{D} - \mathbb{H}$$

Experience Divergence Uncertainty

Maximum likelihood estimation (MLE) at a close look:

- The most classical learning algorithm

- Supervised:
  - Observe data $\mathcal{D} = \{(\boldsymbol{x}^*, \boldsymbol{y}^*)\}$
  - Solve with SGD

$$\min_{\theta} -\mathbb{E}_{(\boldsymbol{x}^*, \boldsymbol{y}^*) \sim \mathcal{D}} \left[\log p_\theta(\boldsymbol{y}^* | \boldsymbol{x}^*) \right]$$

- Unsupervised:
  - Observe $\mathcal{D} = \{(\boldsymbol{x}^*)\}$, $\boldsymbol{y}$ is latent variable
  - Posterior $p_\theta(\boldsymbol{y}|\boldsymbol{x})$
  - Solve with EM:
    - E-step imputes latent variable $\boldsymbol{y}$ through expectation on complete likelihood
    - M-step: supervised MLE

$$\min_{\theta} -\mathbb{E}_{\boldsymbol{x}^* \sim \mathcal{D}} \left[\log \int_{\boldsymbol{y}} p_\theta(\boldsymbol{x}^*, \boldsymbol{y}) \right]$$

MLE as Entropy Maximization

- Equivalence between supervised MLE and maximum entropy (when p_θ is an exponential family distribution)

$$\min_\theta - \mathbb{E}_{(\boldsymbol{x}^*,\boldsymbol{y}^*)\sim\mathcal{D}}\left[\log p_\theta(\boldsymbol{y}^*|\boldsymbol{x}^*)\right]$$

Shannon entropy H

$$\min_{p(\boldsymbol{x},\boldsymbol{y})} H(p)$$

features $T(\boldsymbol{x},\boldsymbol{y})$

$$s.t.\ \mathbb{E}_p[T(\boldsymbol{x},\boldsymbol{y})] = \mathbb{E}_{(x^*,y^*)\sim\mathcal{D}}[T(\boldsymbol{x},\boldsymbol{y})]$$

data as constraints

MLE as Entropy Maximization

- Unsupervised MLE can be achieved by maximizing the negative free energy:
  - Introduce an auxiliary variational distribution $q(\boldsymbol{y}|\boldsymbol{x})$ (and then play with its entropy and cross entropy, etc.)

$$\log \int_{\boldsymbol{y}} p_\theta(\boldsymbol{x}^*, \boldsymbol{y}) \geq H\big(q(\boldsymbol{y}|\boldsymbol{x}^*)\big) + \mathbb{E}_{q(\boldsymbol{y}|\boldsymbol{x}^*)}[\log p_\theta(\boldsymbol{x}^*, \boldsymbol{y})]$$

MaxEnt perspective

The general expression as a constrained optimization:

(auxiliary) distribution q

loss

$$\min_{q,\boldsymbol{\theta}} \; \mathcal{L}(q, \boldsymbol{\theta})$$

constrained set

$$s.t. \;\; q \in \mathcal{Q}.$$

- Supervised MLE and maximum entropy
- Unsupervised MLE and maximum entropy
- Bayesian inference and maximum entropy
  - Bayesian inference as optimization

MaxEnt perspective

The general expression as a constrained optimization:

(auxiliary) distribution q

loss

$$\min_{q,\boldsymbol{\theta}} \; \mathcal{L}(q, \boldsymbol{\theta})$$

constrained set

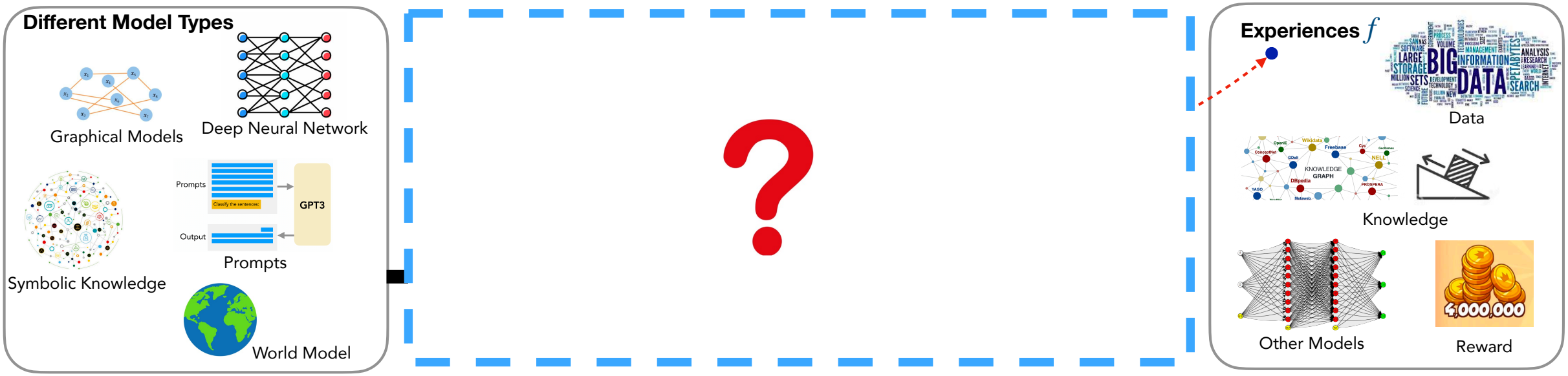
$$s.t. \;\; q \in \mathcal{Q}.$$

- Supervised MLE and maximum entropy
- Unsupervised MLE and maximum entropy
- Bayesian inference and maximum entropy

$$\min_{q(\boldsymbol{z})} \; -\mathbf{H}(q(\boldsymbol{z})) + \log p(\mathcal{D}) - \mathbb{E}_{q(\boldsymbol{z})}\left[\log \pi(\boldsymbol{z}) + \sum\nolimits_{\boldsymbol{x}^* \in \mathcal{D}} \log p(\boldsymbol{x}^* | \boldsymbol{z})\right]$$

$$s.t.\; q(\boldsymbol{z}) \in \mathcal{P}$$

A "Standard Model" of Machine Learning



The Standard Equation (SE)

- Let $\boldsymbol{t}$ be the variable of interest
  - E.g., the input-output pair $\boldsymbol{t} = (\boldsymbol{x}, \boldsymbol{y})$ in a prediction task
  - or $\boldsymbol{t} = \boldsymbol{x}$ in generative modeling
- $p_{\theta}(\boldsymbol{t})$: the target model to be learned
- $q(\boldsymbol{t})$: auxiliary distribution
- The SE:

$$\min_{q, \boldsymbol{\theta}, \boldsymbol{\xi}} -\alpha \mathbb{H}(q) + \beta \mathbb{D}\Big(q(\boldsymbol{t}), p_{\theta}(\boldsymbol{t})\Big) + U(\boldsymbol{\xi})$$

$$s.t. \; -\mathbb{E}_{q(\boldsymbol{t})}\Big[\, f_k(\boldsymbol{t}) \,\Big] < \boldsymbol{\xi}_k, \quad k = 1, \ldots, K$$

  - Experience function f represents external experiences of different kinds for training the model
    - $f_k(\boldsymbol{t}) \in \mathbb{R}$: measures the goodness of a configuration $\boldsymbol{t}$ in light of any given experiences
    - Data, constraints, reward, adversarial discriminators, etc., can all be formulated as an experience function (later)
    - Maximizing $\mathbb{E}_{q(\boldsymbol{t})}[f_k(\boldsymbol{t})]$ -> q is encouraged to produce samples receiving high scores

[Hu & Xing, 2021]

The Standard Equation (SE)

- Let $\boldsymbol{t}$ be the variable of interest
  - E.g., the input-output pair $\boldsymbol{t} = (\boldsymbol{x}, \boldsymbol{y})$ in a prediction task
  - or $\boldsymbol{t} = \boldsymbol{x}$ in generative modeling
- $p_\theta(\boldsymbol{t})$: the target model to be learned
- $q(\boldsymbol{t})$: auxiliary distribution
- The SE:

$$\min_{q, \boldsymbol{\theta}, \boldsymbol{\xi}} -\alpha \mathbb{H}(q) + \beta \mathbb{D}\Big(q(\boldsymbol{t}), p_\theta(\boldsymbol{t})\Big) + U(\boldsymbol{\xi})$$

$$s.t. \; -\mathbb{E}_{q(\boldsymbol{t})}\Big[\, f_k(\boldsymbol{t}) \,\Big] < \boldsymbol{\xi}_k, \quad k = 1, \ldots, K$$

  - Divergence $\mathbb{D}$: measures the distance between the target model p_θ to be trained and the auxiliary model q
    - E.g., cross entropy

[Hu & Xing, 2021]

The Standard Equation (SE)

- Let $\boldsymbol{t}$ be the variable of interest
  - E.g., the input-output pair $\boldsymbol{t} = (\boldsymbol{x}, \boldsymbol{y})$ in a prediction task
  - or $\boldsymbol{t} = \boldsymbol{x}$ in generative modeling
- $p_{\theta}(\boldsymbol{t})$: the target model to be learned
- $q(\boldsymbol{t})$: auxiliary distribution
- The SE:

$$\min_{q, \boldsymbol{\theta}, \boldsymbol{\xi}} -\alpha \mathbb{H}(q) + \beta \mathbb{D}\left(q(\boldsymbol{t}), p_{\theta}(\boldsymbol{t})\right) + U(\boldsymbol{\xi})$$

$$s.t. \; -\mathbb{E}_{q(\boldsymbol{t})}\left[f_k(\boldsymbol{t}) \right] < \boldsymbol{\xi}_k, \quad k = 1, \ldots, K$$

  - Uncertainty $\mathbb{H}$: controls the compactness of the model
    - E.g., Shannon entropy

[Hu & Xing, 2021]

The Standard Equation (SE)

$$\min_{q,\boldsymbol{\theta},\boldsymbol{\xi}} -\alpha\mathbb{H}(q) + \beta\mathbb{D}\left(q(\boldsymbol{t}), p_\theta(\boldsymbol{t})\right) + U(\boldsymbol{\xi})$$

$$s.t. \; -\mathbb{E}_{q(\boldsymbol{t})}\left[\, f_k(\boldsymbol{t}) \,\right] < \xi_k, \quad k = 1, \ldots, K$$

Assuming penalty $U = \sum_k \xi_k$, and $f = \sum_k f_k$:

$$\min_{q,\theta} -\alpha\mathbb{H}(q) + \beta\mathbb{D}\left(q(\boldsymbol{t}), p_\theta(\boldsymbol{t})\right) - \mathbb{E}_{q(\boldsymbol{t})}\left[\, f(\boldsymbol{t}) \,\right]$$

3 terms:

Uncertainty
(self-regularization)
e.g., Shannon entropy

Uncertainty

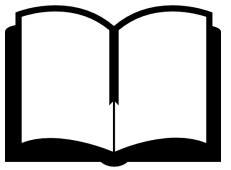
Divergence
(fitness)
e.g., Cross Entropy

Teacher $q(\boldsymbol{t})$

Student $p_\theta(\boldsymbol{t})$

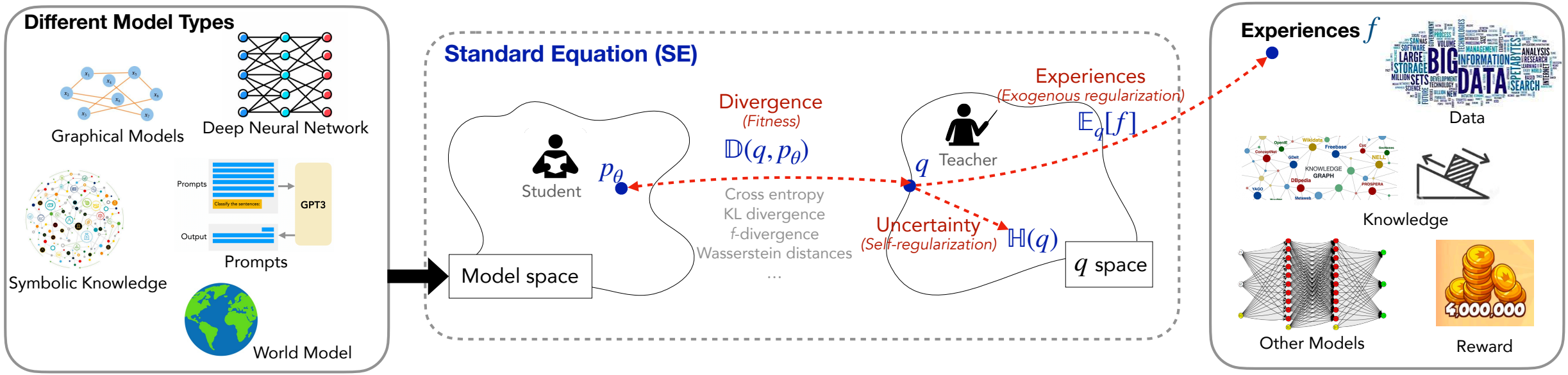
Experiences
(exogenous regularizations)
e.g., data examples, rules

Textbook $f(\boldsymbol{t})$



The Standard Equation (SE)

$$\min_{q,\theta} -\alpha\mathbb{H}(q) + \beta\mathbb{D}\Big(q(\boldsymbol{t}), p_\theta(\boldsymbol{t})\Big) - \mathbb{E}_{q(\boldsymbol{t})}\Big[\, f(\boldsymbol{t})\,\Big]$$



[Note: in SE, experience function f can also depends on θ. See the paper for mor details]

The Standard Equation (SE)

$$\min_{q,\theta} -\alpha\mathbb{H}(q) + \beta\mathbb{D}\left(q(\boldsymbol{t}), p_\theta(\boldsymbol{t})\right) - \mathbb{E}_{q(\boldsymbol{t})}\left[f(\boldsymbol{t})\right]$$

- Formulates a large space of learning algorithms, which encompasses many well-known algorithms

SE encompasses many well-known algorithms (more later)

Experience type	Experience function f	Divergence $\mathbb{D}$	α	β	Algorithm
Data instances	$f_{\text{data}}(\boldsymbol{x};\mathcal{D})$	CE	1	1	Unsupervised MLE
	$f_{\text{data}}(\boldsymbol{x},\boldsymbol{y};\mathcal{D})$	CE	1	ϵ	Supervised MLE
	$f_{\text{data-self}}(\boldsymbol{x},\boldsymbol{y};\mathcal{D})$	CE	1	ϵ	Self-supervised MLE
	$f_{\text{data-w}}(\boldsymbol{t};\mathcal{D})$	CE	1	ϵ	Data Re-weighting
	$f_{\text{data-aug}}(\boldsymbol{t};\mathcal{D})$	CE	1	ϵ	Data Augmentation
	$f_{\text{active}}(\boldsymbol{x},\boldsymbol{y};\mathcal{D})$	CE	1	ϵ	Active Learning (Ertekin et al., 2007)
Knowledge	$f_{rule}(\boldsymbol{x},\boldsymbol{y})$	CE	1	1	Posterior Regularization (Ganchev et al., 2010)
	$f_{rule}(\boldsymbol{x},\boldsymbol{y})$	CE	$\mathbb{R}$	1	Unified EM (Samdani et al., 2012)
Reward	$\log Q^{\theta}(\boldsymbol{x},\boldsymbol{y})$	CE	1	1	Policy Gradient
	$\log Q^{\theta}(\boldsymbol{x},\boldsymbol{y}) + Q^{in,\theta}(\boldsymbol{x},\boldsymbol{y})$	CE	1	1	+ Intrinsic Reward
	$Q^{\theta}(\boldsymbol{x},\boldsymbol{y})$	CE	$\rho > 0$	$\rho > 0$	RL as Inference
Model	$f^{\text{mimicking}}_{\text{model}}(\boldsymbol{x},\boldsymbol{y};\mathcal{D})$	CE	1	ϵ	Knowledge Distillation (G. Hinton et al., 2015)
Variational	binary classifier	JSD	0	1	Vanilla GAN (Goodfellow et al., 2014)
	discriminator	f-divergence	0	1	f-GAN (Nowozin et al., 2016)
	1-Lipschitz discriminator	W_1 distance	0	1	WGAN (Arjovsky et al., 2017)
	1-Lipschitz discriminator	KL	0	1	PPO-GAN (Y. Wu et al., 2020)
Online	$f_{\tau}(\boldsymbol{t})$	CE	$\rho > 0$	$\rho > 0$	Multiplicative Weights (Freund & Schapire, 1997)

SE Component: Experience Function f

Different choices of experience function f lead to different algorithms:

$$\min_{q,\theta} -\mathbb{E}_{q(\boldsymbol{x},\boldsymbol{y})}\Big[\, f(\boldsymbol{x},\boldsymbol{y}) \,\Big] + \beta\mathbb{D}\Big(q(\boldsymbol{x},\boldsymbol{y}),\, p_\theta(\boldsymbol{x},\boldsymbol{y})\Big) - \alpha\mathbb{H}(q)$$

Experience
(exogenous regularizations)
e.g., data examples, rules

Set *Divergence* to *Cross Entropy*
$\mathbb{D}(q, p_\theta) = -\mathbb{E}_q[\,\log p_\theta\,]$

Set *Uncertainty* to *Shannon Entropy*
$\mathbb{H}(q) = H(q) := -\mathbb{E}_q[\,\log q\,]$

SE with supervised data experience

$$\min_{q,\theta} -\alpha\mathbb{H}(q) + \beta\mathbb{D}\Big(q(\boldsymbol{t}), p_\theta(\boldsymbol{t})\Big) - \mathbb{E}_{q(\boldsymbol{t})}\Big[\, f(\boldsymbol{t}) \,\Big]$$

- Input-output variables $\boldsymbol{t} = (\boldsymbol{x}, \boldsymbol{y})$
- Experience: dataset $\mathcal{D} = \{(\boldsymbol{x}^*, \boldsymbol{y}^*)\}$ of size N
  - defines the empirical distribution

$$\tilde{p}(\boldsymbol{x}, \boldsymbol{y}) = \frac{m(\boldsymbol{x}, \boldsymbol{y})}{N} = \mathbb{E}_{(\boldsymbol{x}^*, \boldsymbol{y}^*)\sim\mathcal{D}}[\mathbb{1}_{(\boldsymbol{x}^*, \boldsymbol{y}^*)}(\boldsymbol{x}, \boldsymbol{y})]$$

- Define the experience function

$$f := f_{data}(\boldsymbol{x}, \boldsymbol{y}; \mathcal{D}) = \log \mathbb{E}_{(\boldsymbol{x}^*, \boldsymbol{y}^*)\sim\mathcal{D}}\big[\, \mathbb{1}_{(\boldsymbol{x}^*, \boldsymbol{y}^*)}(\boldsymbol{x}, \boldsymbol{y}) \,\big]$$

- Let $\mathbb{D}$ cross entropy, $\mathbb{H}$ Shannon entropy, $\alpha = 1, \beta = \epsilon$ (a very small value)

$$\min_{q,\theta} -H(q) - \epsilon\, \mathbb{E}_q\Big[\log p_\theta(\boldsymbol{x}, \boldsymbol{y})\Big] - \mathbb{E}_q\Big[\, f_{data}(\boldsymbol{x}, \boldsymbol{y}; \mathcal{D}) \,\Big]$$

SE with supervised data experience

$$f := f_{data}(\boldsymbol{x}, \boldsymbol{y}; \mathcal{D}) = \log \mathbb{E}_{(\boldsymbol{x}^*, \boldsymbol{y}^*) \sim \mathcal{D}} \left[\mathbb{1}_{(\boldsymbol{x}^*, \boldsymbol{y}^*)}(\boldsymbol{x}, \boldsymbol{y}) \right]$$

$$\min_{q, \theta} -H(q) - \epsilon \, \mathbb{E}_q \left[\log p_\theta(\boldsymbol{x}, \boldsymbol{y}) \right] - \mathbb{E}_q \left[f_{data}(\boldsymbol{x}, \boldsymbol{y}; \mathcal{D}) \right]$$

- At each iteration n:

$$\text{Teacher:} \quad q^{(n+1)}(\boldsymbol{t}) = \exp \left\{ \frac{\beta \log p_{\theta^{(n)}}(\boldsymbol{t}) + f(\boldsymbol{t})}{\alpha} \right\} / Z \approx \tilde{p}(\boldsymbol{x}, \boldsymbol{y})$$

$$\text{Student:} \quad \boldsymbol{\theta}^{(n+1)} = \operatorname*{argmax}_{\boldsymbol{\theta}} \mathbb{E}_{q^{(n+1)}(\boldsymbol{t})} \left[\log p_\theta(\boldsymbol{t}) \right],$$

q reduces to the empirical distribution

Maximizes data log-likelihood

- Recovers **supervised MLE!**

SE with unsupervised data experience

$$\min_{q,\theta} -\alpha\mathbb{H}(q) + \beta\mathbb{D}\Big(q(\boldsymbol{t}), p_\theta(\boldsymbol{t})\Big) - \mathbb{E}_{q(\boldsymbol{t})}\Big[\, f(\boldsymbol{t})\,\Big]$$

- Input-output variables $\boldsymbol{t} = (\boldsymbol{x}, \boldsymbol{y})$
- Experience: dataset $\mathcal{D} = \{(\boldsymbol{x}^*)\}$ of size N, I,e., we only observe the $\boldsymbol{x}$ part
  - defines the empirical distribution

$$\tilde{p}(\boldsymbol{x}) = \frac{m(\boldsymbol{x})}{N} = \mathbb{E}_{\boldsymbol{x}^*\sim\mathcal{D}}[\mathbb{1}_{\boldsymbol{x}^*}(\boldsymbol{x})]$$

- Define the experience function

$$f := f_{data}(\boldsymbol{x}\,; \mathcal{D}) = \log \mathbb{E}_{\boldsymbol{x}^*\sim\mathcal{D}}[\mathbb{1}_{\boldsymbol{x}^*}(\boldsymbol{x})]$$

- Let $\mathbb{D}$ cross entropy, $\mathbb{H}$ Shannon entropy, $\alpha = 1, \beta = 1$

$$\min_{q,\theta} -H(q) - \mathbb{E}_q\Big[\log p_\theta(\boldsymbol{x}, \boldsymbol{y})\Big] - \mathbb{E}_q\Big[\; f_{data}(\boldsymbol{x}; \mathcal{D})\;\Big]$$

  - Assume $q(\boldsymbol{x}, \boldsymbol{y}) = \tilde{p}(\boldsymbol{x})q(\boldsymbol{y}|\boldsymbol{x})$

Recovers **unsupervised** MLE (EM)!

SE with manipulated data experience

- Input-output variables $\boldsymbol{t} = (\boldsymbol{x}, \boldsymbol{y})$
- Experience: dataset $\mathcal{D} = \{(\boldsymbol{x}^*, \boldsymbol{y}^*)\}$ of size N
  - defines the empirical distribution

$$\tilde{p}(\boldsymbol{x}, \boldsymbol{y}) = \frac{m(\boldsymbol{x}, \boldsymbol{y})}{N} = \mathbb{E}_{(\boldsymbol{x}^*, \boldsymbol{y}^*) \sim \mathcal{D}}[\mathbb{1}_{(\boldsymbol{x}^*, \boldsymbol{y}^*)}(\boldsymbol{x}, \boldsymbol{y})]$$

- Define the experience function

$$f := f_{data}(\boldsymbol{x}, \boldsymbol{y}; \mathcal{D}) = \log \mathbb{E}_{(\boldsymbol{x}^*, \boldsymbol{y}^*) \sim \mathcal{D}}\left[\mathbb{1}_{(\boldsymbol{x}^*, \boldsymbol{y}^*)}(\boldsymbol{x}, \boldsymbol{y})\right]$$

- The similarity measure $\mathbb{1}_a(b)$ is too restrictive. Let's enrich it:
  - Don't have to be 0/1, we can scale it

$$f := f_{data-w}(\boldsymbol{x}, \boldsymbol{y}; \mathcal{D}) = \log \mathbb{E}_{(\boldsymbol{x}^*, \boldsymbol{y}^*) \sim \mathcal{D}}\left[w(\boldsymbol{x}^*, \boldsymbol{y}^*) \cdot \mathbb{1}_{(\boldsymbol{x}^*, \boldsymbol{y}^*)}(\boldsymbol{x}, \boldsymbol{y})\right]$$

  - Plug f_{data-w} into SE, keep all other configurations the same as supervised MLE, we recover **data re-weighting** in the "student" step

$$\max_{\boldsymbol{\theta}} \mathbb{E}_{\boldsymbol{t}^* \sim \mathcal{D}}\left[w(\boldsymbol{t}^*) \cdot \log p_\theta(\boldsymbol{t}^*)\right]$$

SE with manipulated data experience

- Input-output variables $\boldsymbol{t} = (\boldsymbol{x}, \boldsymbol{y})$
- Experience: dataset $\mathcal{D} = \{(\boldsymbol{x}^*, \boldsymbol{y}^*)\}$ of size N
  - defines the empirical distribution

$$\tilde{p}(\boldsymbol{x}, \boldsymbol{y}) = \frac{m(\boldsymbol{x}, \boldsymbol{y})}{N} = \mathbb{E}_{(\boldsymbol{x}^*, \boldsymbol{y}^*) \sim \mathcal{D}}[\mathbb{1}_{(\boldsymbol{x}^*, \boldsymbol{y}^*)}(\boldsymbol{x}, \boldsymbol{y})]$$

- Define the experience function

$$f := f_{data}(\boldsymbol{x}, \boldsymbol{y}; \mathcal{D}) = \log \mathbb{E}_{(\boldsymbol{x}^*, \boldsymbol{y}^*) \sim \mathcal{D}}\left[\mathbb{1}_{(\boldsymbol{x}^*, \boldsymbol{y}^*)}(\boldsymbol{x}, \boldsymbol{y})\right]$$

- The similarity measure $\mathbb{1}_a(b)$ is too restrictive. Let's enrich it:
  - Don't have to match exactly, we can relax it

$$f := f_{data-aug}(\boldsymbol{x}, \boldsymbol{y}; \mathcal{D}) = \log \mathbb{E}_{(\boldsymbol{x}^*, \boldsymbol{y}^*) \sim \mathcal{D}}\left[a_{(\boldsymbol{x}^*, \boldsymbol{y}^*)}(\boldsymbol{x}, \boldsymbol{y})\right]$$

    - $a_{(\boldsymbol{x}^*, \boldsymbol{y}^*)}(\boldsymbol{x}, \boldsymbol{y})$: assigns non-zero probability to not only the exact $(\boldsymbol{x}^*, \boldsymbol{y}^*)$ but also other $(\boldsymbol{x}, \boldsymbol{y})$ configurations
  - Plug $f_{data-aug}$ into SE, keep all other configurations the same as supervised MLE, we recover **data augmentation** in the "student" step $\max_{\boldsymbol{\theta}} \mathbb{E}_{\boldsymbol{t}^* \sim \mathcal{D},\, \boldsymbol{t} \sim a_{\boldsymbol{t}^*}(\boldsymbol{t})} [\log p_\theta(\boldsymbol{t})]$.

SE with reward experience -- Policy Gradient

$$\min_{q,\theta} \; -\alpha H(q) - \beta \mathbb{E}_q \Big[\log p_\theta(\boldsymbol{x}, \boldsymbol{y}) \Big] - \mathbb{E}_q \Big[\quad f(\boldsymbol{x}, \boldsymbol{y}) \quad \Big]$$

- Policy gradient

$$f^\theta(\boldsymbol{x}, \boldsymbol{y}) := \log Q^\theta(\boldsymbol{x}, \boldsymbol{y}) \qquad \alpha = \beta = 1$$

  - Teacher step: $q^{(n)}(\boldsymbol{x}, \boldsymbol{y}) = p_{\theta^{(n)}}(\boldsymbol{x}, \boldsymbol{y}) Q^{\theta^{(n)}}(\boldsymbol{x}, \boldsymbol{y}) \,/\, Z$
  - Student step:

$$\mathbb{E}_{q^{(n)}(\boldsymbol{x},\boldsymbol{y})} \left[\nabla_\theta \log p_\theta(\boldsymbol{x}, \boldsymbol{y}) \right] + \mathbb{E}_{q^{(n)}(\boldsymbol{x},\boldsymbol{y})} \left[\nabla_\theta f^\theta_{\text{reward},1}(\boldsymbol{x}, \boldsymbol{y}) \right] \Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}^{(n)}}$$

$$= 1/Z \cdot \sum_{\boldsymbol{x}} p_0(\boldsymbol{x}) \nabla_\theta \sum_{\boldsymbol{y}} p_\theta(\boldsymbol{y}|\boldsymbol{x}) Q^\theta(\boldsymbol{x}, \boldsymbol{y}) \Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}^{(n)}} \qquad \textit{(log-derivative trick)}$$

$$= 1/Z \cdot \sum_{\boldsymbol{x}} \mu^\theta(\boldsymbol{x}) \sum_{\boldsymbol{y}} Q^\theta(\boldsymbol{x}, \boldsymbol{y}) \nabla_\theta p_\theta(\boldsymbol{y}|\boldsymbol{x}) \Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}^{(n)}} \qquad \textit{(policy gradient theorem)}$$

policy gradient

(auxiliary) distribution q — loss

$$\min_{q,\boldsymbol{\theta}} \ \mathcal{L}(q, \boldsymbol{\theta})$$

$$s.t. \ \ q \in \mathcal{Q}.$$

constrained set

Key Takeaways

- The MaxEnt perspective converts learning into a constrained optimization problem
- The standard equation (SE):

$$\min_{q,\theta} -\alpha \mathbb{H}(q) + \beta \mathbb{D}\Big(q(\boldsymbol{t}), p_\theta(\boldsymbol{t})\Big) - \mathbb{E}_{q(\boldsymbol{t})}\Big[f(\boldsymbol{t}) \Big]$$

3 terms:

Uncertainty
(self-regularization)
e.g., Shannon entropy

Uncertainty

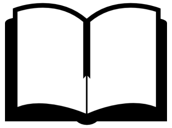
Divergence
(fitness)
e.g., Cross Entropy

Teacher $q(\boldsymbol{t})$

Student $p_\theta(\boldsymbol{t})$

Experiences
(exogenous regularizations)
e.g., data examples, rules

Textbook $f(\boldsymbol{t})$



Questions?