Statistical applications of contrastive (self-supervised) learning

Michael U. Gutmann

michael.gutmann@ed.ac.uk

School of Informatics, University of Edinburgh

August 28 2024

Main messages

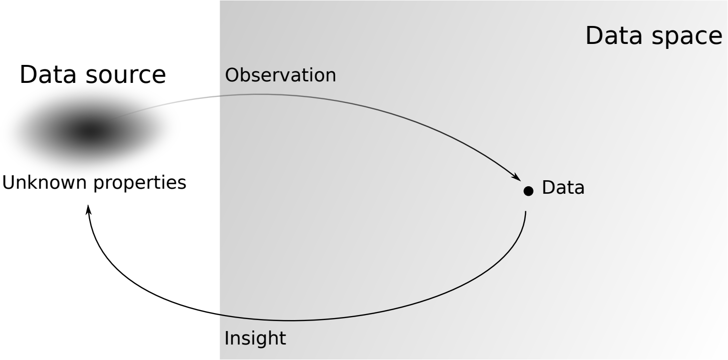
1. The likelihood function is a main workhorse in statistics and ML but becomes easily computationally intractable.
2. Contrastive learning is an intuitive and computationally feasible alternative to likelihood-based approaches.
3. It is broadly applicable. Here: (1) parameter estimation, (2) Bayesian inference, and (3) Bayesian experimental design.

Contents

Contents

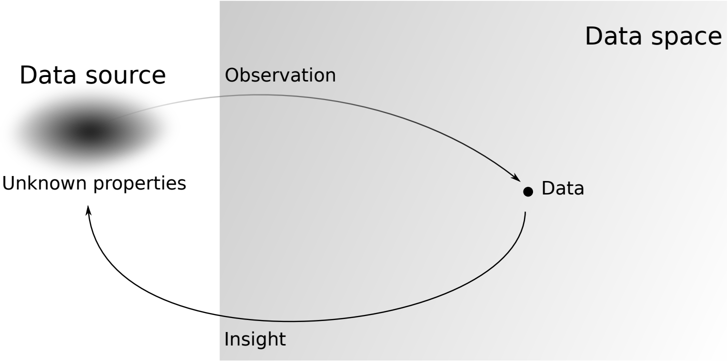
Overall goal

- Goal: Understanding properties of some data source
- Enables predictions, decision making under uncertainty, . . .



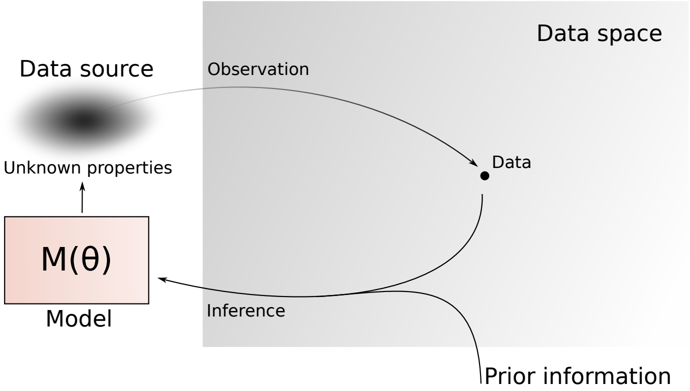
Two fundamental tasks

- Data analysis : Given data $\mathcal{D}$, what can we robustly say about the properties of the source?
- Experimental design : How to obtain data $\mathcal{D}$ that is maximally useful for learning about the properties?



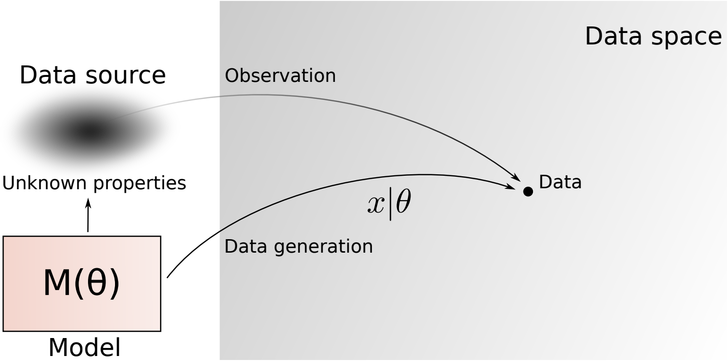
Approaching the tasks via parametric models

- Set up a model with properties that the unknown data source might have.
- The potential properties are induced by the parameters θ of the model.



The likelihood function $L(\boldsymbol{\theta})$

- Probability that the model generates data like the observed one when using parameter value $\boldsymbol{\theta}$
- Classically, the main workhorse in statistics/ML but intractable for the models we would like to work with.

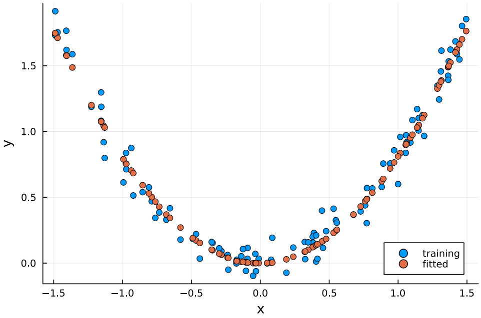


Contents

Contents

From deep supervised to deep unsupervised learning

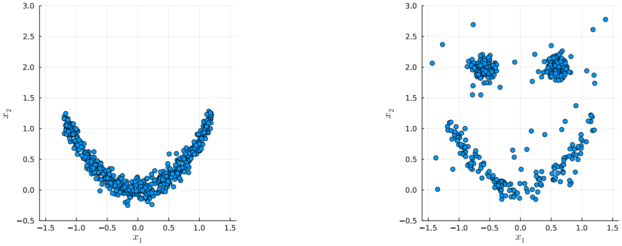
- Deep neural networks have transformed supervised learning.
- Allow us to specify complex parameterised functions $f_\theta(\mathbf{x})$ mapping the inputs (covariates) $\mathbf{x}$ to the target variables.
- Fitting is supported by a rich code infrastructure.
- Simple regression example:



($f_\theta(\mathbf{x})$ was a multi-layer NN with relu activation functions)

From deep supervised to deep unsupervised learning

- "All models are wrong" but deep neural networks are broadly applicable to different supervised learning tasks.
- The situation is a bit different in unsupervised learning (density estimation).
- Consider task of learning the parameters $\boldsymbol{\theta}$ of a density model $p(\mathbf{x}|\boldsymbol{\theta})$ for the following two data sets.



- We may need rather different models and frameworks (e.g. mixture models etc).

Energy-based models

- We would like to use the same model-class $p(\mathbf{x}|\boldsymbol{\theta})$ for both data sets.
- One approach is to write

$$p(\mathbf{x}|\boldsymbol{\theta}) = \frac{\exp(-f_{\boldsymbol{\theta}}(\mathbf{x}))}{Z(\boldsymbol{\theta})} \qquad Z(\boldsymbol{\theta}) = \int \exp(-f_{\boldsymbol{\theta}}(\mathbf{x}))\,\mathrm{d}\mathbf{x} \tag{1}$$

  where $f_{\boldsymbol{\theta}}$ is a deep neural network (sometimes called the energy)
- Models specified in terms of $f_{\boldsymbol{\theta}}$ are called energy-based models.
- Widely used:
  - computer vision and modelling of images
  - natural language processing and machine translation
  - modelling social or biological networks
  - ...

Log-likelihood for energy-based models

- Given iid data $\mathcal{D} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, the log-likelihood function is

$$\ell(\boldsymbol{\theta}) = \sum_{i=1}^{n} \log p(\mathbf{x}_i|\boldsymbol{\theta}) = -\sum_{i=1}^{n} f_{\boldsymbol{\theta}}(\mathbf{x}_i) - n \log Z(\boldsymbol{\theta}) \qquad (2)$$

- Problem: The partition function $Z(\boldsymbol{\theta})$ is defined in terms of a high-dimensional integral

$$Z(\boldsymbol{\theta}) = \int \exp(-f_{\boldsymbol{\theta}}(\mathbf{x}))\,\mathrm{d}\mathbf{x} \qquad (3)$$

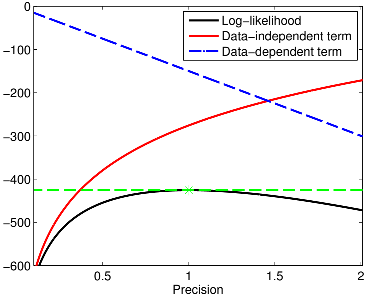
  that is typically impossible to compute.
- Makes evaluating $\ell(\boldsymbol{\theta})$ intractable.

We cannot just ignore the partition function

- Consider $p(x|\theta) = \frac{\exp(-f_{\boldsymbol{\theta}}(x))}{Z(\theta)} = \frac{\exp\left(-\theta\frac{x^2}{2}\right)}{\sqrt{2\pi/\theta}}$ with $x \in \mathbb{R}$.
- Log-likelihood function for precision (inverse variance) $\theta \geq 0$

$$\ell(\theta) = -n \log \sqrt{\frac{2\pi}{\theta}} - \theta \sum_{i=1}^{n} \frac{x_i^2}{2} \quad (4)$$

- Data-dependent (blue) and independent part (red) balance each other.
- Ignoring $Z(\boldsymbol{\theta})$ leads to meaningless estimates.



Question 1: estimation of deep energy-based models

- Consider an energy-based model specified as

$$p(\mathbf{x}|\boldsymbol{\theta}) = \frac{\exp(-f_{\boldsymbol{\theta}}(\mathbf{x}))}{Z(\boldsymbol{\theta})} \qquad Z(\boldsymbol{\theta}) = \int \exp(f_{\boldsymbol{\theta}}(-\mathbf{x}))\,\mathrm{d}\mathbf{x} \tag{5}$$

  where $f_{\boldsymbol{\theta}}$ is a deep neural network.
- Problem: Likelihood-based learning requires us to compute or approximate $Z(\boldsymbol{\theta})$ (or related quantities).
- Question: What learning principles can we use to efficiently estimate $\boldsymbol{\theta}$ when the model pdf $p(\mathbf{x}|\boldsymbol{\theta})$ is only available up to $Z(\boldsymbol{\theta})$?

Contents

Simulator models

- Widely used:
  - computer models/simulators in the natural sciences
  - evolutionary biology to model evolution
  - neuroscience to model neural processing
  - epidemiology to model the spread of an infectious disease
  - ...
- Specified via a measurable function g, typically not known in closed form but implemented as a computer programme.

$$\mathbf{x} = g(\boldsymbol{\theta}, \boldsymbol{\omega}), \quad \boldsymbol{\omega} \sim p(\boldsymbol{\omega}) \tag{6}$$

  Maps parameters $\boldsymbol{\theta}$ and "noise" $\boldsymbol{\omega}$ to data $\mathbf{x}$
- Equals the basic definition of a random variable in terms of a measurable function.

Simulator models

Some examples:

- $p(\omega) = \mathcal{N}(\omega; 0, 1)$, $g(\boldsymbol{\theta}, \omega) = \mu + \sigma\omega$, with $\boldsymbol{\theta} = (\mu, \sigma)$.
- $p(\omega) = \mathcal{U}(\omega; 0, 1)$, $g(\boldsymbol{\theta}, \omega) =$ inverse cdf of some target distribution with parameters $\boldsymbol{\theta}$.
- $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_n)$ is obtained by solving a parameterised ODE subject to noise, e.g.

  $$\dot{\mathbf{z}} = f(\mathbf{z}, t, \boldsymbol{\theta}) \qquad \mathbf{x}_i = \mathbf{z}(t_i) + \boldsymbol{\omega}_i, \quad i = 1, \ldots, n \tag{7}$$

  where $\boldsymbol{\omega}_i \sim \mathcal{N}(\boldsymbol{\omega}_i; 0, \boldsymbol{\Sigma})$ iid.
- $\mathbf{x}$ is the solution to a stochastic differential equation with parameters $\boldsymbol{\theta}$.
- $\mathbf{x}$ is the output of some graphics rendered with parameters $\boldsymbol{\theta}$.
- ...

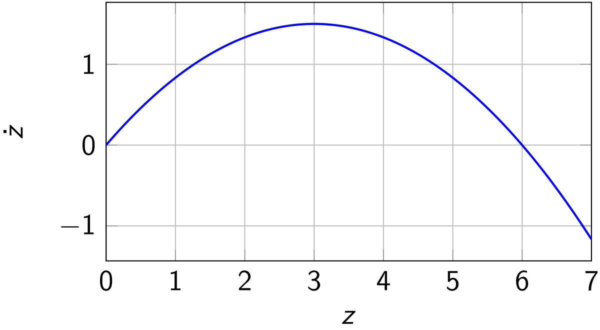
Example from ecology

- A classical model for population growth is

$$\dot{z} = rz(1 - \frac{z}{k}) \tag{8}$$

where r is the growth rate and k is the carrying capacity.

- Defines a dynamics with a fixed point at 0 (unstable) and at k (stable). For example, for $k = 6$:



Example from ecology

- Denote by z_i the solution of the ODE evaluated at times $t_1, \ldots, t_n$.
- Let the observed data $x_1, \ldots, x_n$ be the z_i corrupted by some noise:

$$x_i = z_i + \omega_i \qquad\qquad \omega_i = \mathcal{N}(\omega_i; 0, \sigma^2) \tag{9}$$

  In other words, $x_i | z_i \sim \mathcal{N}(x_i; z_i, \sigma^2)$
- Note that the z_i, and hence the x_i, depend on the values of k and r.
- They are the parameters $\boldsymbol{\theta}$ of the model.

Key strengths and weaknesses of simulator models

- Strengths:
  - Most general definition of a statistical model
  - Connects statistics to the natural sciences and engineering
- Weaknesses:
  - Model pdf implicitly defined in terms of the inverse image of $g(\boldsymbol{\theta}, \boldsymbol{\omega})$:

    $$\Pr(\mathbf{x} \in \mathcal{A} | \boldsymbol{\theta}) = \Pr(\{\omega : g(\boldsymbol{\theta}, \boldsymbol{\omega}) \in \mathcal{A}\})$$

    for some event $\mathcal{A}$.
  - Computing inverse image and the associated probability is typically not possible, which makes the model pdf $p(\mathbf{x}|\boldsymbol{\theta})$ intractable.

Intractable model pdf implies intractable likelihood

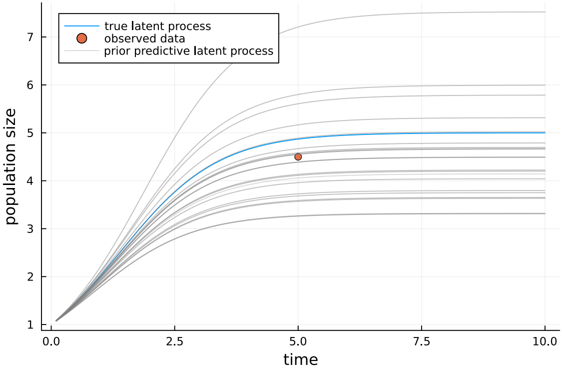
- For models explicitly expressed as a family of pdfs $\{p(\mathbf{x}|\boldsymbol{\theta})\}$ indexed by $\boldsymbol{\theta}$: $L(\boldsymbol{\theta}) = p(\mathcal{D}|\boldsymbol{\theta})$.
- For models implicitly expressed in terms of a simulator, $p(\mathbf{x}|\boldsymbol{\theta})$ and hence $L(\boldsymbol{\theta})$ are typically not available.
- This causes problems in likelihood-based inference, which requires $L(\boldsymbol{\theta})$:

$$\hat{\boldsymbol{\theta}} = \operatorname*{argmax}_{\boldsymbol{\theta}} L(\boldsymbol{\theta}) \qquad \text{or} \qquad p(\boldsymbol{\theta}|\mathcal{D}) = \frac{L(\boldsymbol{\theta})}{p(\mathcal{D})} p(\boldsymbol{\theta}) \qquad (10)$$

- In some cases, we can obtain $p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta})$ for some unobserved variable $\mathbf{z}$ and then use MCMC or variational methods for inference. We here do not assume that the model allows for such an expression.

Ecology example

- A latent process $z(t)$ follows the ODE $\dot{z} = rz(1 - z/k)$. We observe $x \sim \mathcal{N}(x; z(t), \sigma^2)$ at a known time t (say $t = 5$).
- Assuming a Gamma prior on k (and r known), what are plausible values of the carrying capacity k given x?



(Gamma prior has a shape parameter 9, and scale parameter 0.5, giving a prior mean of 4.5 and std 1.5. "True" value of k: 5, std of observation noise: 0.3)

Question 2: Bayesian inference for simulator models

- Consider a simulator model specified as

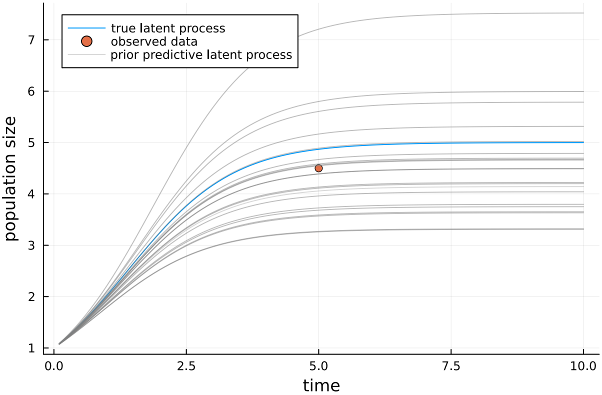
$$\mathbf{x} = g(\boldsymbol{\theta}, \boldsymbol{\omega}), \quad \boldsymbol{\omega} \sim p(\boldsymbol{\omega}) \tag{11}$$

  where g is not known in closed form but implemented as a computer programme.
- We are given data $\mathcal{D} = (\mathbf{x}_1, \ldots, \mathbf{x}_n)$ and have a prior $p(\boldsymbol{\theta})$ on $\boldsymbol{\theta}$. We would like to determine which values of $\boldsymbol{\theta}$ are plausible given $\mathcal{D}$.
- Problem: Likelihood-based inference would require us to numerically compute the likelihood or run e.g. MCMC, which may not be feasible for complex simulator models.
- Question: How can we compute or sample from $p(\boldsymbol{\theta}|\mathcal{D})$ without access to the model pdf $p(\mathbf{x}|\boldsymbol{\theta})$?

Contents

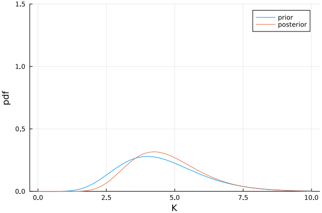
Ecology example: when to measure?

- In the previous example, we took the measurement at $t = 5$. Was that a good choice? Could it have been better?
- Deciding about t corresponds to experimental design. What is a criterion to measure optimality of an experimental design?

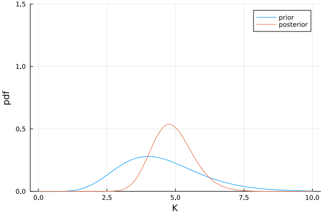


Ecology example: when to measure?

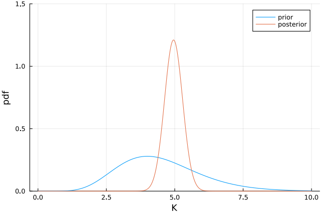
We want experimental data from which we can learn something, i.e. data that can change our belief.



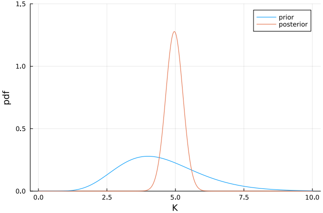
(a) Measurement at $t = 1$



(b) Measurement at $t = 2$



(c) Measurement at $t = 5$



(d) Measurement at $t = 8$

Expected information gain

- Assume now that we have some control over the data collection process. Denote the control (design) variables by $\mathbf{d}$ and include $\mathbf{d}$ in the model as an additional parameter:

$$p(\mathbf{x}|\boldsymbol{\theta}, \mathbf{d}) \Longleftrightarrow \mathbf{x} = g(\boldsymbol{\theta}, \mathbf{d}, \boldsymbol{\omega}), \quad \boldsymbol{\omega} \sim p(\boldsymbol{\omega}) \tag{12}$$

- While $\boldsymbol{\theta}$ is unknown (e.g. the carrying capacity k), $\mathbf{d}$ is controllable (e.g. the measurement time).
- We can assess the value of some data $\mathcal{D}$ obtained with design $\mathbf{d}$ by computing how much it can change our belief about $\boldsymbol{\theta}$.

Expected information gain

- Let us use the Kullback-Leibler divergence to measure the difference between our belief before seeing the data, $p(\boldsymbol{\theta}|\mathbf{d})$, and our belief after seeing the data, $p(\boldsymbol{\theta}|\mathcal{D}, \mathbf{d})$ when using design $\mathbf{d}$:

$$\text{value}(\mathcal{D}, \mathbf{d}) = \text{KL}(p(\boldsymbol{\theta}|\mathcal{D}, \mathbf{d})||p(\boldsymbol{\theta}|\mathbf{d})) \tag{13}$$

$$= \int p(\boldsymbol{\theta}|\mathcal{D}, \mathbf{d}) \log \frac{p(\boldsymbol{\theta}|\mathcal{D}, \mathbf{d})}{p(\boldsymbol{\theta}|\mathbf{d})} \, \mathrm{d}\boldsymbol{\theta} \tag{14}$$

  We call this the information gain.
- Quantifies how much information we gain about $\boldsymbol{\theta}$ by analysing the data $\mathcal{D}$.
- Often but not necessarily: $p(\boldsymbol{\theta}|\mathbf{d}) = p(\boldsymbol{\theta})$ (belief about $\boldsymbol{\theta}$ is independent of the design $\mathbf{d}$).

Expected information gain

- $\text{value}(\mathcal{D}, \mathbf{d})$ can be used to assess the value of some data $\mathcal{D}$ that we have gathered with design $\mathbf{d}$.
- When deciding about what design $\mathbf{d}$ to use, $\mathcal{D}$ is not yet observed.
- However, we can average over possible data sets $\mathcal{D}$ that we may observe when using $\mathbf{d}$ and compute the expected information gain (EIG):

$$\text{EIG}(\mathbf{d}) = \int p(\mathbf{x}|\mathbf{d}) \text{value}(\mathbf{x}, \mathbf{d}) \, \mathrm{d}\mathbf{x} \tag{15}$$

$$= \int p(\mathbf{x}|\mathbf{d}) \int p(\boldsymbol{\theta}|\mathbf{x}, \mathbf{d}) \log \frac{p(\boldsymbol{\theta}|\mathbf{x}, \mathbf{d})}{p(\boldsymbol{\theta}|\mathbf{d})} \, \mathrm{d}\boldsymbol{\theta} \, \mathrm{d}\mathbf{x} \tag{16}$$

$$= \int \int p(\mathbf{x}, \boldsymbol{\theta}|\mathbf{d}) \log \frac{p(\boldsymbol{\theta}|\mathbf{x}, \mathbf{d})}{p(\boldsymbol{\theta}|\mathbf{d})} \, \mathrm{d}\boldsymbol{\theta} \, \mathrm{d}\mathbf{x} \tag{17}$$

Expected information gain

- Equals an expectation with respect to $p(\mathbf{x}, \boldsymbol{\theta}|\mathbf{d})$, hence

$$\mathrm{EIG}(\mathbf{d}) = \mathbb{E}_{p(\mathbf{x},\boldsymbol{\theta}|\mathbf{d})}\left[\log \frac{p(\boldsymbol{\theta}|\mathbf{x},\mathbf{d})}{p(\boldsymbol{\theta}|\mathbf{d})}\right] \tag{18}$$

- Since

$$p(\boldsymbol{\theta}|\mathbf{x},\mathbf{d}) = \frac{p(\mathbf{x},\boldsymbol{\theta}|\mathbf{d})}{p(\mathbf{x}|\mathbf{d})} = \frac{p(\mathbf{x}|\boldsymbol{\theta},\mathbf{d})p(\boldsymbol{\theta}|\mathbf{d})}{p(\mathbf{x}|\mathbf{d})} \tag{19}$$

  we also have

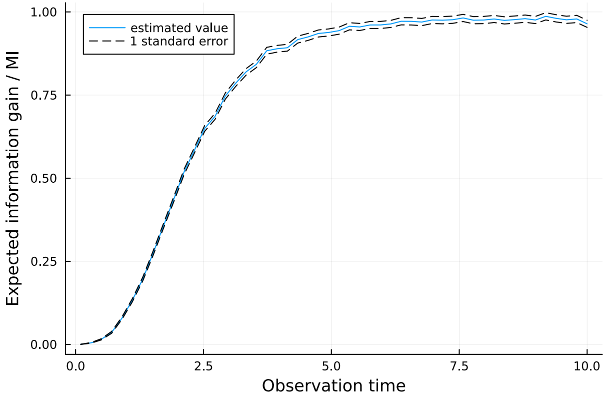
$$\mathrm{EIG}(\mathbf{d}) = \mathbb{E}_{p(\mathbf{x},\boldsymbol{\theta}|\mathbf{d})}\left[\log \frac{p(\mathbf{x},\boldsymbol{\theta}|\mathbf{d})}{p(\mathbf{x}|\mathbf{d})p(\boldsymbol{\theta}|\mathbf{d})}\right] \tag{20}$$

$$= \mathrm{KL}(p(\mathbf{x},\boldsymbol{\theta}|\mathbf{d})||p(\mathbf{x}|\mathbf{d})p(\boldsymbol{\theta}|\mathbf{d})) \tag{21}$$

  which is known as the mutual information (MI) between $\mathbf{x}$ and $\boldsymbol{\theta}$ (for fixed $\mathbf{d}$). Measures the dependency between $\mathbf{x}$ and $\boldsymbol{\theta}$ for a given $\mathbf{d}$.
- We choose $\mathbf{d}$ such that the EIG / MI is maximised.

Ecology example: when to measure?

- For the simple toy example, we can numerically compute the EIG as a function of the measurement time.



- EIG is larger for later measurements, which is in line with posterior vs prior plots.

Question 3: experimental design for simulator models

- Consider a simulator model specified as

$$\mathbf{x} = g(\boldsymbol{\theta}, \mathbf{d}, \boldsymbol{\omega}), \quad \boldsymbol{\omega} \sim p(\boldsymbol{\omega}) \tag{22}$$

  where g is not known in closed form but implemented as a computer programme so that $p(\mathbf{x}|\boldsymbol{\theta}, \mathbf{d})$ is not available.
- We would like to compute the value of $\mathbf{d}$ that maximises the expected information gain about $\boldsymbol{\theta}$.
- Problem: The expected information gain cannot be computed/maximised when $p(\mathbf{x}|\boldsymbol{\theta}, \mathbf{d})$ is not tractable.
- Question: How to obtain a design $\mathbf{d}$ that approximately maximises the expected information gain without access to the model pdf $p(\mathbf{x}|\boldsymbol{\theta}, \mathbf{d})$?

Contents

Summary so far

- Not all models are specified as a family of pdfs.
- Two important classes considered here:
  1. Energy-based (unnormalised) models
  2. Simulator models
- The models are rather different, common point:

  Multiple integrals needed to be solved to represent the models in terms of pdfs.
- Solving the integrals exactly is computationally impossible (curse of dimensionality)

  $\Rightarrow$ No model pdfs
  $\Rightarrow$ A wall of intractable likelihoods that prevents inference and experimental design

Summary so far

- We considered diverse kinds of problems and associated questions:
  1. Deep energy-based models: What learning principles can we use to efficiently estimate $\boldsymbol{\theta}$ when the model pdf $p(\mathbf{x}|\boldsymbol{\theta})$ is only available up to $Z(\boldsymbol{\theta})$?
  2. Inference for simulator models: How can we compute or sample from $p(\boldsymbol{\theta}|\mathcal{D})$ without access to the model pdf $p(\mathbf{x}|\boldsymbol{\theta})$?
  3. Exp design for simulator models: How to obtain a design $\mathbf{d}$ that approximately maximises the expected information gain without access to the model pdf $p(\mathbf{x}|\boldsymbol{\theta}, \mathbf{d})$?
- Contrastive learning provides a single answer to the above questions.

Main messages

1. The likelihood function is a main workhorse in statistics and ML but becomes easily computationally intractable. ✓
2. Contrastive learning is an intuitive and computationally feasible alternative to likelihood-based approaches.
3. It is broadly applicable. Here: (1) parameter estimation, (2) Bayesian inference, and (3) Bayesian experimental design.

Contents

Contents

Question 1: estimation of deep energy-based models

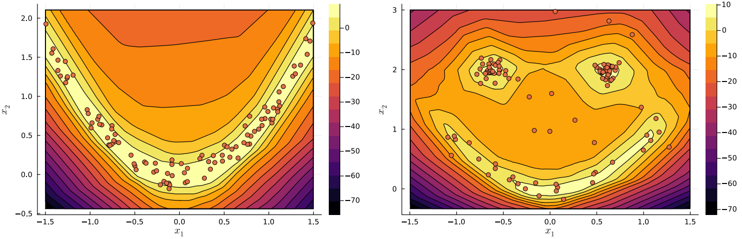
- Consider an energy-based model specified as

$$p(\mathbf{x}|\boldsymbol{\theta}) = \frac{\exp(-f_{\boldsymbol{\theta}}(\mathbf{x}))}{Z(\boldsymbol{\theta})} \qquad Z(\boldsymbol{\theta}) = \int \exp(f_{\boldsymbol{\theta}}(-\mathbf{x}))\,\mathrm{d}\mathbf{x} \tag{23}$$

  where $f_{\boldsymbol{\theta}}$ is a deep neural network.
- Problem: Likelihood-based learning requires us to compute or approximate $Z(\boldsymbol{\theta})$ (or related quantities).
- Question: What learning principles can we use to efficiently estimate $\boldsymbol{\theta}$ when the model pdf $p(\mathbf{x}|\boldsymbol{\theta})$ is only available up to $Z(\boldsymbol{\theta})$?

Preview 1: contrastive deep energy-based learning

- Let $p(\mathbf{x}|\boldsymbol{\theta}) \propto \exp(-f_{\boldsymbol{\theta}}(\mathbf{x}))$ where $f_{\boldsymbol{\theta}}(\mathbf{x})$ is a deep neural network.
- Contour plot of the log-density obtained with contrastive learning (up to additive constant). Obtained with the same model and training procedure.



- Main point: contrastive learning allows us to use flexible deep neural networks for unsupervised learning (density estimation) in exactly the same way as in supervised learning.

Question 2: Bayesian inference for simulator models

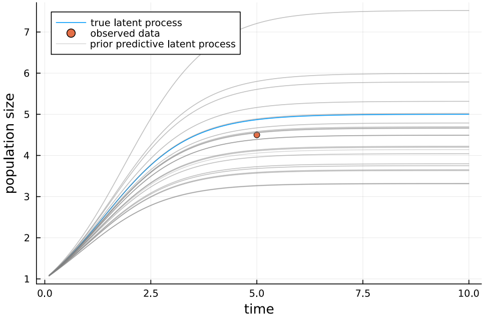
- Consider a simulator model specified as

$$\mathbf{x} = g(\boldsymbol{\theta}, \boldsymbol{\omega}), \quad \boldsymbol{\omega} \sim p(\boldsymbol{\omega}) \tag{24}$$

  where g is not known in closed form but implemented as a computer programme.
- We are given data $\mathcal{D} = (\mathbf{x}_1, \ldots, \mathbf{x}_n)$ and have a prior $p(\boldsymbol{\theta})$ on $\boldsymbol{\theta}$. We would like to determine which values of $\boldsymbol{\theta}$ are plausible given $\mathcal{D}$.
- Problem: Likelihood-based inference would require us to numerically compute the likelihood or run e.g. MCMC, which may not be feasible for complex simulator models.
- Question: How can we compute or sample from $p(\boldsymbol{\theta}|\mathcal{D})$ without access to the model pdf $p(\mathbf{x}|\boldsymbol{\theta})$?

Ecology example

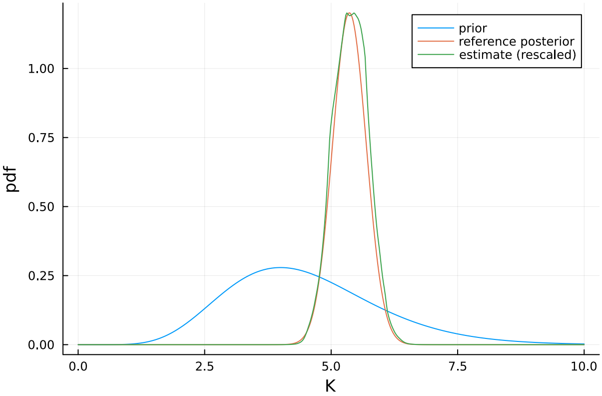
- A latent process $z(t)$ follows the ODE $\dot{z} = rz(1 - z/k)$. We observe $x \sim \mathcal{N}(x|z(t), \sigma^2)$ at some fixed time t (say $t = 5$).
- Assuming a Gamma prior on k (and r known), what are plausible values of the carrying capacity k given x?



(Gamma prior has a shape parameter 9, and scale parameter 0.5, giving a prior mean of 4.5 and std 1.5. "True" value of k: 5, std of observation noise: 0.3)

Preview 2: contrastive Bayesian inference

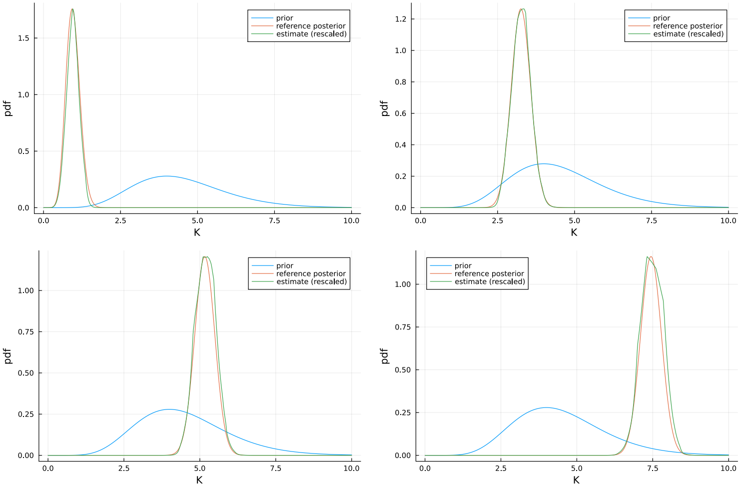
- Reference posterior (via numerical integration) and posterior estimated via contrastive learning.



- Main point: contrastive learning allows us to estimate posteriors $p(\boldsymbol{\theta}|\mathcal{D})$ for simulator models without access to $L(\boldsymbol{\theta})$.

Preview 2: contrastive Bayesian inference

- The method is amortised with respect to the observed data: it returns $p(\boldsymbol{\theta}|\mathcal{D})$ for any value of $\mathcal{D}$ without new learning.



Question 3: experimental design for simulator models

- Consider a simulator model specified as

$$\mathbf{x} = g(\boldsymbol{\theta}, \mathbf{d}, \boldsymbol{\omega}), \quad \boldsymbol{\omega} \sim p(\boldsymbol{\omega}) \tag{25}$$

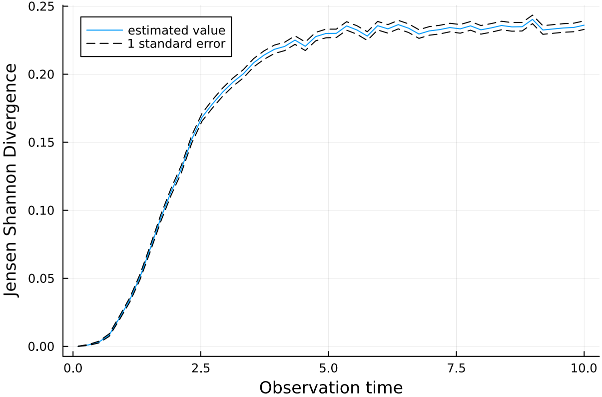
  where g is not known in closed form but implemented as a computer programme so that $p(\mathbf{x}|\boldsymbol{\theta}, \mathbf{d})$ is not available.
- We would like to compute the value of $\mathbf{d}$ that maximises the expected information gain about $\boldsymbol{\theta}$.
- Problem: The expected information gain cannot be computed/maximised when $p(\mathbf{x}|\boldsymbol{\theta}, \mathbf{d})$ is not tractable.
- Question: How to obtain a design $\mathbf{d}$ that approximately maximises the expected information gain without access to the model pdf $p(\mathbf{x}|\boldsymbol{\theta}, \mathbf{d})$?

Preview 3: contrastive experimental design

- We optimise another measure of information gain: While the EIG is defined in terms of the KL-divergence, we use a proxy measure that is defined in terms of another divergence, the Jensen-Shannon divergence (JSD).
- The JSD is a symmetrized and smoothed version of the KL divergence. Considered more robust.

Preview 3: contrastive experimental design

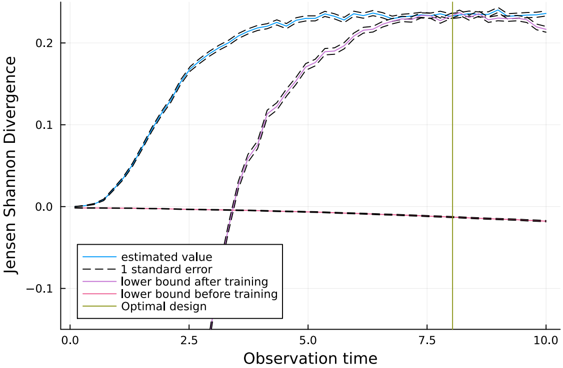
- For the simple toy example, we can numerically compute the JSD as a function of the measurement time.



- Similar behaviour as the EIG: later measurements are optimal.

Preview 3: contrastive experimental design

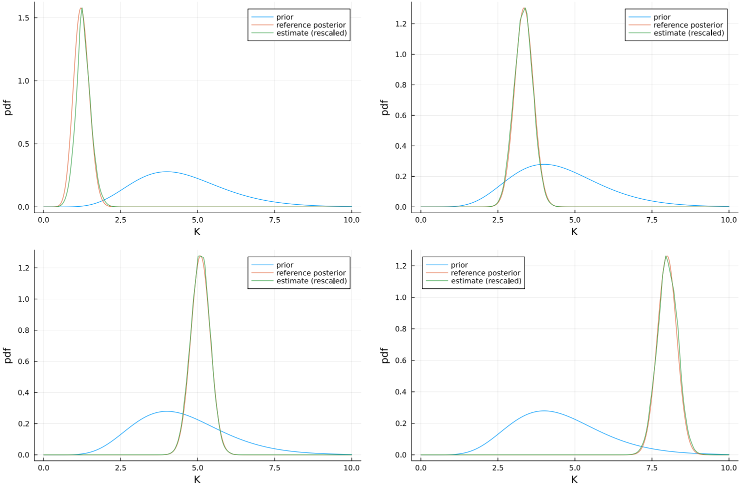
- To find the optimal design, we learn a lower bound on the JSD and jointly tighten the bound and determine its maximiser.



- Main point: Contrastive learning enables and accelerates exp design with simulator models by only approximating the JSD around its maximiser $\hat{\mathbf{d}}$.

Preview 3: contrastive experimental design

- The method also returns posteriors $p(\boldsymbol{\theta}|\mathcal{D}, \hat{\mathbf{d}})$ that are amortised with respect to the observed data.



Contents

Basic idea

- The basic idea in contrastive learning is to learn the difference between the data of interest and some reference data.
- Properties of the reference are typically known or not of interest; by learning the difference we focus the (computational) resources on learning what matters.
- As straightforward as

$$\underbrace{b}_{\text{reference}} + \underbrace{a - b}_{\text{difference}} \Rightarrow \underbrace{a}_{\text{interest}} \tag{26}$$

- Contrastive learning has two main ingredients:
  1. Learning/measuring the difference
  2. Constructing the reference

Connection to other frameworks

- Link to (log) ratio estimation (see e.g. Sugiyama et al's textbook "Density Ratio Estimation in Machine Learning".)

$$\underbrace{\log p_b}_{\text{reference}} + \underbrace{\log p_a - \log p_b}_{\text{difference}} \Rightarrow \underbrace{\log p_a}_{\text{interest}} \tag{27}$$

- Link to Bayes' rule

$$\underbrace{\log p(\boldsymbol{\theta})}_{\text{reference}} + \underbrace{\log p(\mathbf{x}|\boldsymbol{\theta}) - \log p(\mathbf{x})}_{\text{difference}} \Rightarrow \underbrace{\log p(\boldsymbol{\theta}|\mathbf{x})}_{\text{interest}} \tag{28}$$

- Link to classification: learning differences between data sets can be seen as a classification problem.

Ingredient 1: learning the difference

- Let $\mathcal{D} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ be the data of interest, $\mathbf{x}_i \sim p$ (iid), and $\{\mathbf{y}_1, \ldots \mathbf{y}_m\}$ be reference data, $\mathbf{y}_i \sim q$ (iid).
- Label the data: $(\mathbf{x}_i, 1)$, $(\mathbf{y}_i, 0)$ and learn a classifier h by minimising the (rescaled) logistic loss $J(h)$

$$J(h) = \frac{1}{n}\sum_{i=1}^{n} \log\left[1 + \nu \exp(-h(\mathbf{x}_i))\right] + \frac{\nu}{m}\sum_{i=1}^{m} \log\left[1 + \frac{1}{\nu}\exp(h(\mathbf{y}_i))\right] \tag{29}$$

  where $\nu = m/n$
- For large sample sizes n and m (and fixed ratio ν), the optimal h is

$$h^* = \log \frac{p}{q} \tag{30}$$

Ingredient 1: learning the difference

Two key points:

1. The optimisation is done without any constraints (e.g. normalisation constraint that leads to a partition function). The optimal h is automagically the ratio between two densities

$$h^* = \log \frac{p}{q} \tag{31}$$

2. We only need samples from p and q; we do not need their densities or a model for them (but we do need an appropriate model for the ratio)

Proof that $h^* = \log p - \log q$

- When n and m are large, $J(h) \to \bar{J}(h)$,

$$\bar{J}(h) = \mathbb{E}_{p(\mathbf{x})} \log \left[1 + \nu e^{-h(\mathbf{x})}\right] + \nu \mathbb{E}_{q(\mathbf{y})} \log \left[1 + \frac{1}{\nu} e^{h(\mathbf{y})}\right] \quad (32)$$

- With the definitions $p(.|C=1) = p(.)$ and $p(.|C=0) = q(.)$

$$\bar{J}(h) = \mathbb{E}_{p(\mathbf{u}|C=1)} \log \left[1 + \nu e^{-h(\mathbf{u})}\right] + \nu \mathbb{E}_{p(\mathbf{u}|C=0)} \log \left[1 + \frac{1}{\nu} e^{h(\mathbf{u})}\right] \quad (33)$$

- ν is kept fixed as n and m increase. It equals the ratio of the prior class probabilities:

$$\nu = \frac{m}{n} = \frac{\frac{m}{m+n}}{\frac{n}{m+n}} = \frac{p(C=0)}{p(C=1)} = \frac{p_0}{p_1} \quad (34)$$

Proof that $h^* = \log p - \log q$

- Insert $\nu = p_0/p_1$:

$$\bar{J}(h) = \mathbb{E}_{p(\mathbf{u}|C=1)} \log \left[1 + \frac{p_0}{p_1} e^{-h(\mathbf{u})}\right] + \frac{p_0}{p_1} \mathbb{E}_{p(\mathbf{u}|C=0)} \log \left[1 + \frac{p_1}{p_0} e^{h(\mathbf{u})}\right] \quad (35)$$

- It follows that

$$p_1 \bar{J}(h) = p_1 \mathbb{E}_{p(\mathbf{u}|C=1)} \log \left[1 + \frac{p_0}{p_1} e^{-h(\mathbf{u})}\right] + p_0 \mathbb{E}_{p(\mathbf{u}|C=0)} \log \left[1 + \frac{p_1}{p_0} e^{h(\mathbf{u})}\right] \quad (36)$$

$$= -p_1 \mathbb{E}_{p(\mathbf{u}|C=1)} \log \left[\frac{1}{1 + \frac{p_0}{p_1} e^{-h(\mathbf{u})}}\right] - p_0 \mathbb{E}_{p(\mathbf{u}|C=0)} \log \left[\frac{1}{1 + \frac{p_1}{p_0} e^{h(\mathbf{u})}}\right] \quad (37)$$

Proof that $h^* = \log p - \log q$

- By manipulating the terms in the logs:

$$p_1 \bar{J}(h) = -p_1 \mathbb{E}_{p(\mathbf{u}|C=1)} \log \left[\frac{p_1 e^{h(\mathbf{u})}}{p_0 + p_1 e^{h(\mathbf{u})}} \right] - p_0 \mathbb{E}_{p(\mathbf{u}|C=0)} \log \left[\frac{p_0}{p_0 + p_1 e^{h(\mathbf{u})}} \right] \quad (38)$$

- Let

$$\Pr(C|\mathbf{u}; h) = \begin{cases} \frac{p_1 e^{h(\mathbf{u})}}{p_0 + p_1 e^{h(\mathbf{u})}} & \text{if } C = 1 \\ \frac{p_0}{p_0 + p_1 e^{h(\mathbf{u})}} & \text{if } C = 0 \end{cases} \quad (39)$$

- With $p_1 \mathbb{E}_{p(\mathbf{u}|C=1)} \cdots + p_0 \mathbb{E}_{p(\mathbf{u}|C=0)} = \mathbb{E}_{p(\mathbf{u},C)}$

$$p_1 \bar{J}(h) = -\mathbb{E}_{p(\mathbf{u},C)} \left[\log \Pr(C|\mathbf{u}; h) \right] \quad (40)$$

- Note that $p_1 J(h)$ is just the sample version of $p_1 \bar{J}(h)$.

Proof that $h^* = \log p - \log q$

- Whilst $\Pr(C|\mathbf{u}; h)$ is our model of the conditional distribution of the class C given an input $\mathbf{u}$, let $\Pr(C|\mathbf{u})$ be the true conditional (obtained via Bayes' rule),

$$\Pr(C|\mathbf{u}) = \begin{cases} \frac{p_1 p(\mathbf{u})}{p_0 q(\mathbf{u}) + p_1 p(\mathbf{u})} & \text{if } C = 1 \\ \frac{p_0 q(\mathbf{u})}{p_0 q(\mathbf{u}) + p_1 p(\mathbf{u})} & \text{if } C = 0 \end{cases} \tag{41}$$

  Denominator is the marginal $m(\mathbf{u}) = \sum_C p(\mathbf{u}, C) = p_0 p(\mathbf{u}|C = 0) + p_1 p(\mathbf{u}|C = 1) = p_0 q(\mathbf{u}) + p_1 p(\mathbf{u})$.
- Add $\mathbb{E}_{p(\mathbf{u},C)}[\log \Pr(C|\mathbf{u})]$ to $p_1 \bar{J}(h)$:

$$p_1 \bar{J}(h) + \mathbb{E}_{p(\mathbf{u},C)} \log \Pr(C|\mathbf{u}) = -\mathbb{E}_{p(\mathbf{u},C)} \left[\log \frac{\Pr(C|\mathbf{u}; h)}{\Pr(C|\mathbf{u})} \right] \tag{42}$$

Proof that $h^* = \log p - \log q$

- Introduce abbreviation $\mathcal{L}(h) = p_1 \bar{J}(h) + \mathbb{E}_{p(\mathbf{u},C)} \log \Pr(C|\mathbf{u})$:

$$\mathcal{L}(h) = -\mathbb{E}_{p(\mathbf{u},C)} \left[\log \frac{\Pr(C|\mathbf{u};h)}{\Pr(C|\mathbf{u})} \right] \tag{43}$$

- $\operatorname{argmin}_h \mathcal{L}(h) = \operatorname{argmin}_h \bar{J}(h)$.
- By the chain rule $p(\mathbf{u}, C) = m(\mathbf{u}) \Pr(C|\mathbf{u})$, which gives

$$\mathcal{L}(h) = -\mathbb{E}_{m(\mathbf{u})} \mathbb{E}_{\Pr(C|\mathbf{u})} \left[\log \frac{\Pr(C|\mathbf{u};h)}{\Pr(C|\mathbf{u})} \right] \tag{44}$$

$$= \mathbb{E}_{m(\mathbf{u})} \mathbb{E}_{\Pr(C|\mathbf{u})} \left[\log \frac{\Pr(C|\mathbf{u})}{\Pr(C|\mathbf{u};h)} \right] \tag{45}$$

$$= \mathbb{E}_{m(\mathbf{u})} \mathrm{KL}(\Pr(C|\mathbf{u}) || \Pr(C|\mathbf{u};h)) \tag{46}$$

- Optimal $h(\mathbf{u})$ minimises $\mathrm{KL}(\Pr(C|\mathbf{u}) || \Pr(C|\mathbf{u};h)$ for all $\mathbf{u}$ where $m(\mathbf{u}) > 0$.

Proof that $h^* = \log p - \log q$

- The KL divergence is 0 iff $\Pr(C|\mathbf{u}) = \Pr(C|\mathbf{u}; h)$.
- Recall:

$$\Pr(C|\mathbf{u}; h) = \begin{cases} \frac{p_1 e^{h(\mathbf{u})}}{p_0 + p_1 e^{h(\mathbf{u})}} & \text{if } C = 1 \\ \frac{p_0}{p_0 + p_1 e^{h(\mathbf{u})}} & \text{if } C = 0 \end{cases} \tag{47}$$

$$\Pr(C|\mathbf{u}) = \begin{cases} \frac{p_1 p(\mathbf{u})}{p_0 q(\mathbf{u}) + p_1 p(\mathbf{u})} & \text{if } C = 1 \\ \frac{p_0 q(\mathbf{u})}{p_0 q(\mathbf{u}) + p_1 p(\mathbf{u})} & \text{if } C = 0 \end{cases} \tag{48}$$

- $\Pr(C|\mathbf{u}; h) = \Pr(C|\mathbf{u})$ iff for all $\mathbf{u}$ where $m(\mathbf{u}) > 0$:

$$\exp(h(\mathbf{u})) = \frac{p(\mathbf{u})}{q(\mathbf{u})} \quad \Longleftrightarrow \quad h(\mathbf{u}) = \log \frac{p(\mathbf{u})}{q(\mathbf{u})} \tag{49}$$

This is the result that we wanted to show and concludes the proof.

Logistic loss lower bounds a divergence between p and q

- The optimal h sets $\mathcal{L}(h)$ to zero so that

$$-p_1 \bar{J}(h^*) = \mathbb{E}_{p(\mathbf{u},C)} \log \Pr(C|\mathbf{u}) \tag{50}$$

- Writing the right-hand-side out gives

$$\begin{aligned} -p_1 \bar{J}(h^*) &= p_1 \mathbb{E}_{p(\mathbf{u}|C=1)} \log \Pr(C=1|\mathbf{u}) \\ &\quad + p_0 \mathbb{E}_{p(\mathbf{u}|C=0)} \log \Pr(C=0|\mathbf{u}) & (51) \\ &= p_1 \mathbb{E}_{p(\mathbf{x})} \log \Pr(C=1|\mathbf{x}) + p_0 \mathbb{E}_{q(\mathbf{y})} \log \Pr(C=0|\mathbf{y}) & (52) \\ &= p_1 \mathbb{E}_{p(\mathbf{x})} \log \left[\frac{p_1 p(\mathbf{x})}{p_0 q(\mathbf{x}) + p_1 p(\mathbf{x})} \right] \\ &\quad + p_0 \mathbb{E}_{q(\mathbf{y})} \log \left[\frac{p_0 q(\mathbf{y})}{p_0 q(\mathbf{y}) + p_1 p(\mathbf{y})} \right] & (53) \end{aligned}$$

Logistic loss lower bounds a divergence between p and q

- Continuing from the previous slide

$$
\begin{aligned}
-p_1 \bar{J}(h^*) = & \; p_1 \mathbb{E}_{p(\mathbf{x})} \log \left[\frac{p(\mathbf{x})}{p_0 q(\mathbf{x}) + p_1 p(\mathbf{x})} \right] \\
& + p_0 \mathbb{E}_{q(\mathbf{y})} \log \left[\frac{q(\mathbf{y})}{p_0 q(\mathbf{y}) + p_1 p(\mathbf{y})} \right] \\
& + p_1 \log p_1 + p_0 \log p_0
\end{aligned}
\tag{54}
$$

- The term in red is a generalisation of the KL-divergence known as λ-divergence $D_\lambda(p||q)$ (typically p_1 is denoted by λ).

$$
-p_1 \bar{J}(h^*) = D_\lambda(p||q) + p_1 \log p_1 + p_0 \log p_0 \tag{55}
$$

Logistic loss lower bounds a divergence between p and q

- Since $\bar{J}(h^*) \leq \bar{J}(h)$, we have $-p_1\bar{J}(h^*) \geq -p_1\bar{J}(h)$ and

$$-p_1\bar{J}(h^*) = D_\lambda(p||q) + p_1 \log p_1 + p_0 \log p_0 \geq -p_1\bar{J}(h) \quad (56)$$

- Hence

$$D_\lambda(p||q) \geq -p_1\bar{J}(h) - p_1 \log p_1 - p_0 \log p_0 \quad (57)$$

  Negative logistic loss provides a lower bound on the λ-divergence between p and q.

- For $p_1 = 1/2$, corresponding to $m = n$, the λ-divergence $D_\lambda(p||q)$ equals the Jensen-Shannon divergence (JSD).

$$\text{JSD}(p||q) \geq -\frac{1}{2}\bar{J}(h) + \log 2 \quad (58)$$

  Negative logistic loss provides a lower bound on the JSD.

Summary

- Basic idea of contrastive learning

$$\underbrace{b}_{\text{reference}} + \underbrace{a - b}_{\text{difference}} \Rightarrow \underbrace{a}_{\text{interest}} \tag{59}$$

- Contrastive learning has two main ingredients:
  1. Learning/measuring the difference
  2. Constructing the reference
- Minimising the logistic loss allows us to learn the difference between two distributions p and q.
- Key properties:
  - $h^* = \text{argmin}_h \bar{J}(h) = \log p - \log q$
  - $\text{JSD}(p||q) \geq -\frac{1}{2}\bar{J}(h) + \log 2$ and the bound is tight for h^*.

Ingredient 2: constructing reference data

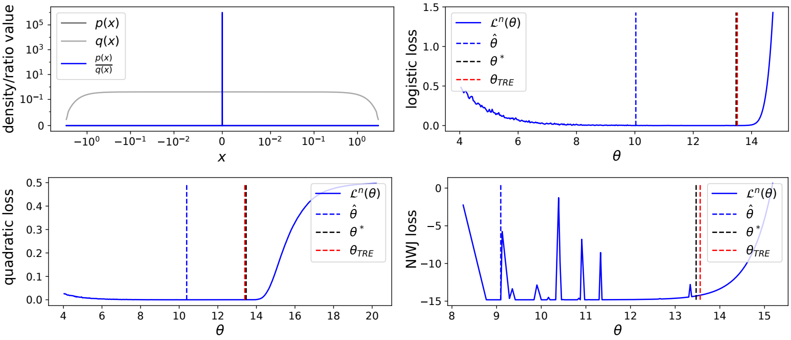
Choice depends on the specific application of contrastive learning.

- Deep energy-based models: Fit a preliminary model and keep it fixed or iterate such that the fitted model becomes the reference (Gutmann and Hyvärinen, AISTATS 2010; JMLR 2012)
- Inference for simulator models: Use the prior or another proposal distribution, and the corresponding predictive distribution (Thomas et al, 2016; Thomas et al, Bayesian Analysis, 2020)
- Exp design for simulator-models: Use the product of the prior and the prior predictive distribution

  (Kleinegesse and Gutmann, AISTATS 2019; ICML 2020; arXiv:2105.04379)
- . . .

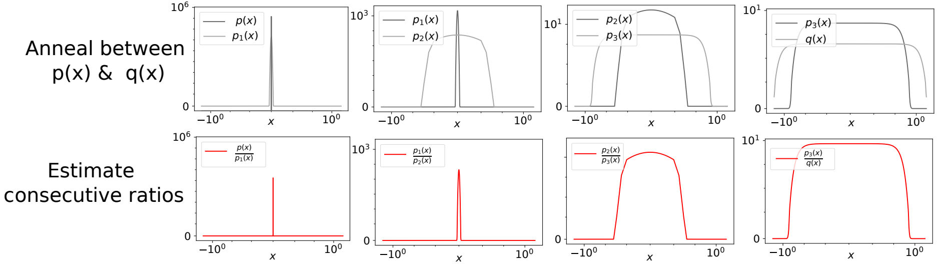
Something to watch out for: the density-chasm problem

- Logistic loss and other single ratio methods struggle if the two distributions are very different ("density chasm")
- Consider ratio between two zero-mean Gaussians. 10'000 samples from each distribution. Ratio parameterised by $\theta \in \mathbb{R}$.
- Solution in red bridges the "gap" using telescopic ratio estimation (TRE) (Rhodes, Xu, and Gutmann, NeurIPS 2020)



A single density-ratio fails to "bridge" the density-chasm.

Let us thus use multiple bridges.

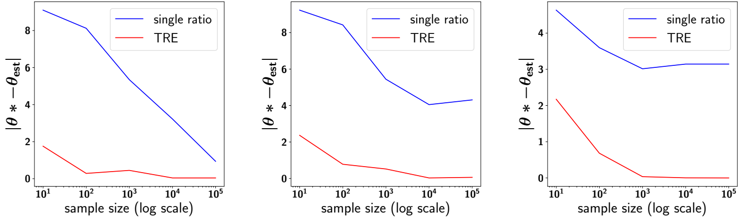


(relabel $p \equiv p_0$ and $q \equiv p_4$) and compute *telescoping* product

$$\frac{p(\mathbf{x})}{q(\mathbf{x})} = \frac{p_0(\mathbf{x})}{p_4(\mathbf{x})} = \frac{p_0(\mathbf{x})}{p_1(\mathbf{x})} \frac{p_1(\mathbf{x})}{p_2(\mathbf{x})} \frac{p_2(\mathbf{x})}{p_3(\mathbf{x})} \frac{p_3(\mathbf{x})}{p_4(\mathbf{x})}. \tag{60}$$

Telescoping density-ratio estimation (Rhodes, Xu, and Gutmann, NeurIPS 2020)

- Sample efficiency curves for the 1d peaked ratio experiment



(a) Logistic loss (b) NWJ loss (c) Quadratic loss

- More results in the paper!
- For further improvements: Srivastava et al, TMLR 2023, *Estimating the Density Ratio between Distributions with High Discrepancy using Multinomial Logistic Regression[...]*.
- Use as replacement of the standard logistic loss if you suspect a density chasm.

Summary

- Basic idea of contrastive learning

$$\underbrace{b}_{\text{reference}} + \underbrace{a - b}_{\text{difference}} \Rightarrow \underbrace{a}_{\text{interest}} \tag{61}$$

- Contrastive learning has two main ingredients:
  1. Learning/measuring the difference
  2. Constructing the reference
- Minimising the logistic loss allows us to learn the difference between two distributions p and q.
- Key properties:
  - $h^* = \text{argmin}_h \bar{J}(h) = \log p - \log q$
  - $\text{JSD}(p||q) \geq -\frac{1}{2}\bar{J}(h) + \log 2$ and the bound is tight for h^*.
- Mind the gap (density chasm).

Contents

Question 1: estimation of deep energy-based models

- Consider an energy-based model specified as

$$p(\mathbf{x}|\boldsymbol{\theta}) = \frac{\exp(-f_{\boldsymbol{\theta}}(\mathbf{x}))}{Z(\boldsymbol{\theta})} \qquad Z(\boldsymbol{\theta}) = \int \exp(f_{\boldsymbol{\theta}}(-\mathbf{x}))\, \mathrm{d}\mathbf{x} \quad (62)$$

  where $f_{\boldsymbol{\theta}}$ is a deep neural network.
- Problem: Likelihood-based learning requires us to compute or approximate $Z(\boldsymbol{\theta})$ (or related quantities).
- Question: What learning principles can we use to efficiently estimate $\boldsymbol{\theta}$ when the model pdf $p(\mathbf{x}|\boldsymbol{\theta})$ is only available up to $Z(\boldsymbol{\theta})$?

Contrastive approach

(Gutmann and Hyvärinen, AISTATS 2010; JMLR 2012)

- Let $\mathcal{D} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ be random sample from $\mathbf{x} \sim p_{\mathbf{x}}$
- Introduce reference data $\mathbf{y}_1, \ldots, \mathbf{y}_m$, sampled iid from a reference distribution with a known distribution q
- Parameterise h as $h_{\boldsymbol{\theta}} = -f_{\boldsymbol{\theta}} - \log q$. Learn $\boldsymbol{\theta}$ by minimising $J(h_{\boldsymbol{\theta}})$.
- After learning: $h_{\hat{\boldsymbol{\theta}}} = -f_{\hat{\boldsymbol{\theta}}} - \log q \approx \log p_{\mathbf{x}} - \log q$
- Hence

$$\exp(-f_{\hat{\boldsymbol{\theta}}}) \approx p_{\mathbf{x}} \tag{63}$$

(We here assume that $f_{\boldsymbol{\theta}}$ is parameterised such that it can change is magnitude freely. Can always be ensured by adding a learnable constant.)

- We can use flexible deep neural networks in unsupervised learning as in supervised learning.
- Formulates unsupervised learning as a supervised learning problem, which is what self-supervised learning is all about.

Illustration on the toy example

Julia code "EBM-contrastive-learning.jl".

How good is the estimation procedure?

- We can characterise the asymptotic distribution and estimation error of the estimator $\hat{\boldsymbol{\theta}} = \text{argmax}_{\boldsymbol{\theta}} J(h_{\boldsymbol{\theta}})$
- I won't go into this here. For those interested, please see the paper *Gutmann and Hyvärinen, Noise-contrastive estimation of Unnormalized Statistical Models, with Applications to Natural Image Statistics, JMLR 2012*.
- As $\nu \to \infty$, $\hat{\boldsymbol{\theta}}$ converges to the maximum likelihood estimator.

Contents

Question 2: Bayesian inference for simulator models

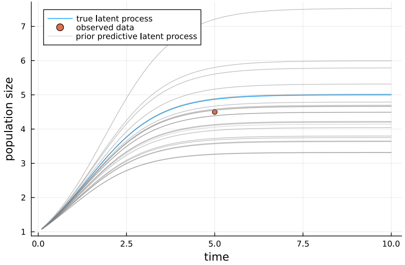
- Consider a simulator model specified as

$$\mathbf{x} = g(\boldsymbol{\theta}, \boldsymbol{\omega}), \quad \boldsymbol{\omega} \sim p(\boldsymbol{\omega}) \tag{64}$$

  where g is not known in closed form but implemented as a computer programme.
- We are given data $\mathcal{D} = (\mathbf{x}_1, \ldots, \mathbf{x}_n)$ and have a prior $p(\boldsymbol{\theta})$ on $\boldsymbol{\theta}$. We would like to determine which values of $\boldsymbol{\theta}$ are plausible given $\mathcal{D}$.
- Problem: Likelihood-based inference would require us to numerically compute the likelihood or run e.g. MCMC, which may not be feasible for complex simulator models.
- Question: How can we compute or sample from $p(\boldsymbol{\theta}|\mathcal{D})$ without access to the model pdf $p(\mathbf{x}|\boldsymbol{\theta})$?

Ecology example

- A latent process $z(t)$ follows the ODE $\dot{z} = rz(1 - z/k)$. We observe $x \sim \mathcal{N}(\mathbf{x}|z(t), \sigma^2)$ at some fixed time t (say $t = 5$).
- Assuming a Gamma prior on k (and r known), what are plausible values of the carrying capacity k given x?



(Gamma prior has a shape parameter 9, and scale parameter 0.5, giving a prior mean of 4.5 and std 1.5. "True" value of k: 5, std of observation noise: 0.3)

Contrastive approach

(Likelihood-Free Inference by Ratio Estimation, Thomas et al, 2016; 2020)
(Dinev and Gutmann, arXiv:1810.09899, 2018)

- Contrastive interpretation of Bayes' rule:

$$\underbrace{\log p(\boldsymbol{\theta})}_{\text{reference}} + \underbrace{\log p(\mathbf{x}|\boldsymbol{\theta}) - \log p(\mathbf{x})}_{\text{difference}} \Rightarrow \underbrace{\log p(\boldsymbol{\theta}|\mathbf{x})}_{\text{interest}} \tag{65}$$

- We use the logistic loss to learn the difference/log-ratio

$$r(\mathbf{x}, \boldsymbol{\theta}) = \log \frac{p(\mathbf{x}|\boldsymbol{\theta})}{p(\mathbf{x})} \tag{66}$$

- We need data from the numerator (class $C = 1$) and denominator (class $C = 0$) distribution.
- Can be generated with the simulator model:

$$C = 1 : \mathbf{x} \sim p(\mathbf{x}|\boldsymbol{\theta}) \Leftrightarrow \boldsymbol{\omega} \sim p(\boldsymbol{\omega}), \mathbf{x} = g(\boldsymbol{\omega}, \boldsymbol{\theta}) \tag{67}$$

$$C = 0 : \mathbf{x} \sim p(\mathbf{x}) \Leftrightarrow \boldsymbol{\omega} \sim p(\boldsymbol{\omega}), \boldsymbol{\theta} \sim p(\boldsymbol{\theta}), \mathbf{x} = g(\boldsymbol{\omega}, \boldsymbol{\theta}) \tag{68}$$

Contrastive approach

- Learned nonlinearity $\hat{h} = \text{argmin}_h J(h)$ provides an estimate of $r(\mathbf{x}, \boldsymbol{\theta})$:

$$\hat{h}(\mathbf{x}, \boldsymbol{\theta}) \approx r(\mathbf{x}, \boldsymbol{\theta}) = \log \frac{p(\mathbf{x}|\boldsymbol{\theta})}{p(\mathbf{x})} \tag{69}$$

- Hence

$$\underbrace{\log \hat{p}(\boldsymbol{\theta}|\mathbf{x})}_{\text{interest}} = \underbrace{\hat{h}(\mathbf{x}, \boldsymbol{\theta})}_{\text{learned difference}} + \underbrace{\log p(\boldsymbol{\theta})}_{\text{reference}} \tag{70}$$

- We can re-use the learned ratio $\hat{h}(\mathbf{x}, \boldsymbol{\theta})$ for any value of $\mathbf{x}$ (amortisation with respect to the data).

Contrastive approach

- Let us have a closer look at the loss $\bar{J}(h)$: (using the large-sample formulation for ease of the argument)

$$\bar{J}(h) = \mathbb{E}_{p(\mathbf{x}|\boldsymbol{\theta})} \log \left[1 + \nu e^{-h(\mathbf{x})}\right] + \nu \mathbb{E}_{p(\mathbf{x})} \log \left[1 + \frac{1}{\nu} e^{h(\mathbf{x})}\right] \quad (71)$$

- The nonlinearity only takes $\mathbf{x}$ as input and not also $\boldsymbol{\theta}$. Small tweak: $h(\mathbf{x}) \to h(\mathbf{x}, \boldsymbol{\theta})$
- The loss above is formulated for a specific (fixed) $\boldsymbol{\theta}$. That is ok if we would like to learn the ratio and evaluate the posterior for a specific $\boldsymbol{\theta}$.
- But we can also learn it for a range of $\boldsymbol{\theta}$ by averaging $\bar{J}(h)$ over an auxiliary distribution $f(\boldsymbol{\theta})$.
- Learns the complete posterior function rather than the value of the posterior at a specific $\boldsymbol{\theta}$. Sometimes called amortisation with respect to $\boldsymbol{\theta}$.

Contrastive approach

- Denote the averaged loss by $\bar{\mathcal{J}}_f(h)$

$$\bar{\mathcal{J}}_f(h) = \mathbb{E}_{f(\boldsymbol{\theta})}\left[\bar{J}(h)\right] \tag{72}$$

$$= \mathbb{E}_{f(\boldsymbol{\theta})}\mathbb{E}_{p(\mathbf{x}|\boldsymbol{\theta})} \log\left[1 + \nu e^{-h(\mathbf{x},\boldsymbol{\theta})}\right]$$

$$+ \nu \mathbb{E}_{f(\boldsymbol{\theta})}\mathbb{E}_{p(\mathbf{x})} \log\left[1 + \frac{1}{\nu} e^{h(\mathbf{x},\boldsymbol{\theta})}\right] \tag{73}$$

- Equivalent to using $\bar{J}(h)$ and targetting the ratio

$$r(\mathbf{x}, \boldsymbol{\theta}) = \log \frac{p(\mathbf{x}|\boldsymbol{\theta}) f(\boldsymbol{\theta})}{p(\mathbf{x}) f(\boldsymbol{\theta})} \tag{74}$$

  Learns $\log \frac{p(\mathbf{x}|\boldsymbol{\theta})}{p(\mathbf{x})}$ due to cancellation of $f(\boldsymbol{\theta})$.
- As before

$$\log \hat{p}(\boldsymbol{\theta}|\mathbf{x}) = \hat{h}(\mathbf{x}, \boldsymbol{\theta}) + \log p(\boldsymbol{\theta}) \tag{75}$$

Illustration on the toy example

Julia code "population-growth-contrastive-learning.jl".

Contents

Question 3: experimental design for simulator models

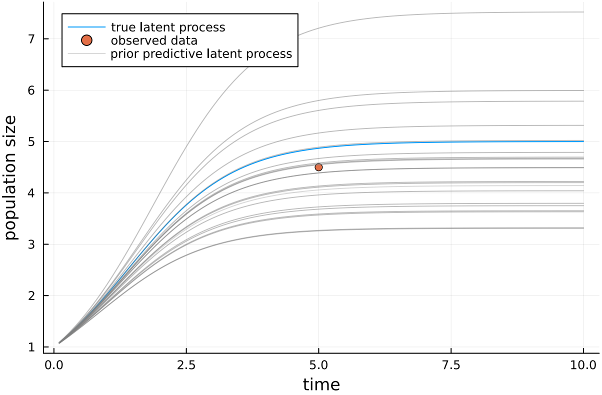
- Consider a simulator model specified as

$$\mathbf{x} = g(\boldsymbol{\theta}, \mathbf{d}, \boldsymbol{\omega}), \quad \boldsymbol{\omega} \sim p(\boldsymbol{\omega}) \tag{76}$$

  where g is not known in closed form but implemented as a computer programme so that $p(\mathbf{x}|\boldsymbol{\theta}, \mathbf{d})$ is not available.
- We would like to compute the value of $\mathbf{d}$ that maximises the expected information gain about $\boldsymbol{\theta}$.
- Problem: The expected information gain cannot be computed/maximised when $p(\mathbf{x}|\boldsymbol{\theta}, \mathbf{d})$ is not tractable.
- Question: How to obtain a design $\mathbf{d}$ that approximately maximises the expected information gain without access to the model pdf $p(\mathbf{x}|\boldsymbol{\theta}, \mathbf{d})$?

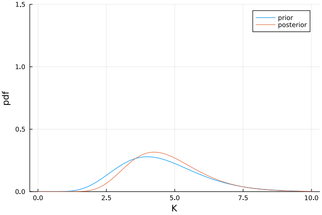
Ecology example: when to measure?

- The figure shows realisations of the population growth $z(t)$ for different values of the parameter of the model, the carrying capacity K.
- We asked: When should we best measure the population to learn about K?

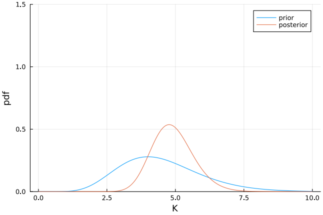


Ecology example: when to measure?

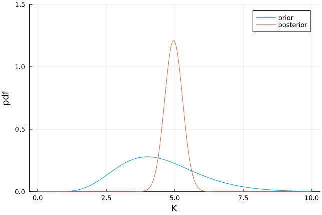
- $t = 5$ is not bad but later seems better



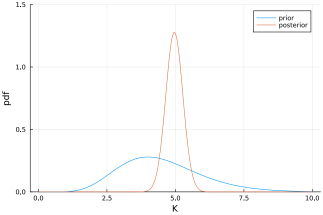
(a) Measurement at $t = 1$



(b) Measurement at $t = 2$



(c) Measurement at $t = 5$

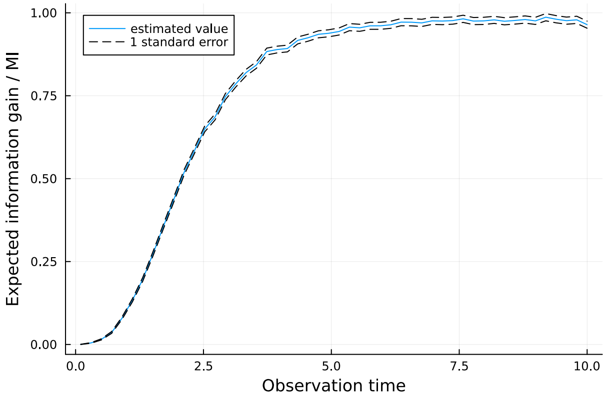


(d) Measurement at $t = 8$

Ecology example: when to measure?

$$\mathrm{EIG}(\mathbf{d}) = \mathbb{E}_{p(\mathbf{x},\boldsymbol{\theta}|\mathbf{d})}\left[\log \frac{p(\mathbf{x},\boldsymbol{\theta}|\mathbf{d})}{p(\mathbf{x}|\mathbf{d})p(\boldsymbol{\theta}|\mathbf{d})}\right] = \mathrm{KL}(p(\mathbf{x},\boldsymbol{\theta}|\mathbf{d})||p(\mathbf{x}|\mathbf{d})p(\boldsymbol{\theta}|\mathbf{d}))$$

- We can use the expected information gain (EIG) to decide when to take the measurement.
- Typically intractable to compute. In the toy example, numerical integration can be used:



Contrastive approach (the direct way)

- The EIG features density ratios that we can estimate by contrastive learning:

$$\text{EIG}(\mathbf{d}) = \mathbb{E}_{p(\mathbf{x},\boldsymbol{\theta}|\mathbf{d})} \log \left[\frac{p(\mathbf{x},\boldsymbol{\theta}|\mathbf{d})}{p(\mathbf{x}|\mathbf{d})p(\boldsymbol{\theta}|\mathbf{d})} \right] = \mathbb{E}_{p(\mathbf{x},\boldsymbol{\theta}|\mathbf{d})} \log \left[\frac{p(\mathbf{x}|\boldsymbol{\theta},\mathbf{d})}{p(\mathbf{x}|\mathbf{d})} \right] \tag{77}$$

- For **d** fixed, we estimate

$$h_{\mathbf{d}}(\mathbf{x},\boldsymbol{\theta}) = \log p(\mathbf{x}|\boldsymbol{\theta},\mathbf{d}) - \log p(\mathbf{x}|\mathbf{d}), \tag{78}$$

  and maximise the sample average of $h_{\mathbf{d}}(\mathbf{x},\boldsymbol{\theta})$ with respect to **d**
- Static setting: Kleinegesse and Gutmann, AISTATS 2019
- Sequential setting where we update our belief about $\boldsymbol{\theta}$ as we sequentially acquire the data: Kleinegesse, Drovandi and Gutmann, Bayesian Analysis 2020

Contrastive approach (with lower bound)

$$\hat{\mathbf{d}} = \operatorname{argmax}_{\mathbf{d}} \mathbb{E}_{p(\mathbf{x},\theta|\mathbf{d})} \log \left[\frac{p(\mathbf{x}|\theta,\mathbf{d})}{p(\mathbf{x}|\mathbf{d})} \right]$$

- Learning the ratio $h_{\mathbf{d}}(\mathbf{x}, \theta)$ and approximating the EIG is computationally costly.
- But we do not need to estimate the EIG accurately everywhere! Only around it's maximum.
- Suggests an approach where we lower bound the EIG (or proxy quantities), and then concurrently tighten the bound and maximise the (proxy) EIG.

Contrastive approach (with lower bound)

- While the EIG is defined in terms of the KL-divergence, we use a proxy measure that is defined in terms of another divergence, the Jensen-Shannon divergence.

$$\mathrm{EIG}(\mathbf{d}) = \mathrm{KL}(p(\mathbf{x}, \boldsymbol{\theta}|\mathbf{d})||p(\mathbf{x}|\mathbf{d})p(\boldsymbol{\theta}|\mathbf{d})) \tag{79}$$

$$\mathrm{proxy}(\mathbf{d}) = \mathrm{JSD} p(\mathbf{x}, \boldsymbol{\theta}|\mathbf{d})||p(\mathbf{x}|\mathbf{d})p(\boldsymbol{\theta}|\mathbf{d})) \tag{80}$$

$$= \frac{1}{2}\Big(\mathrm{KL}(p(\mathbf{x}, \boldsymbol{\theta}|\mathbf{d})||m(\mathbf{x}, \boldsymbol{\theta}|\mathbf{d})) + \mathrm{KL}(p(\mathbf{x}|\mathbf{d})p(\boldsymbol{\theta}|\mathbf{d})||m(\mathbf{x}, \boldsymbol{\theta}|\mathbf{d}))\Big) \tag{81}$$

$$m(\mathbf{x}, \boldsymbol{\theta}|\mathbf{d}) = \frac{1}{2}\left(p(\mathbf{x}, \boldsymbol{\theta}|\mathbf{d}) + p(\mathbf{x}|\mathbf{d})p(\boldsymbol{\theta}|\mathbf{d})\right) \tag{82}$$

- The JSD is a symmetrized and smoothed version of the KL divergence. Considered more robust.

Contrastive approach (with lower bound)

(Kleinegesse and Gutmann, ICML 2020; arXiv:2105.04379)

- Recall:

$$\text{JSD}(p, q) \geq \log 2 - \frac{1}{2}\bar{J}(h) \tag{83}$$

where h is the regression function and $\bar{J}$ the logistic loss.

- Use with

$$p \equiv p(\mathbf{x}, \boldsymbol{\theta}|\mathbf{d}) \qquad\qquad q \equiv p(\mathbf{x}|\mathbf{d})p(\boldsymbol{\theta}|\mathbf{d}) \tag{84}$$

- The loss is, using $\nu = 1$ and making the $\mathbf{d}$ dependency explicit:

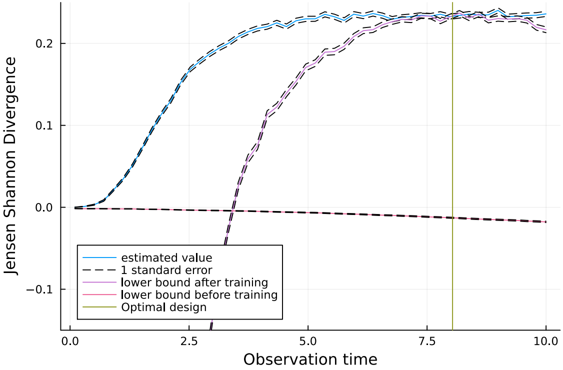
$$\bar{J}(h, \mathbf{d}) = \mathbb{E}_{p(\mathbf{x},\boldsymbol{\theta}|\mathbf{d})} \log \left[1 + e^{-h(\mathbf{x},\boldsymbol{\theta},\mathbf{d})}\right] + \mathbb{E}_{p(\mathbf{x}|\mathbf{d})p(\boldsymbol{\theta}|\mathbf{d})} \log \left[1 + e^{h(\mathbf{x}.\boldsymbol{\theta},\mathbf{d})}\right] \tag{85}$$

- Minimise sample version jointly with respect to h and $\mathbf{d}$:

$$\hat{h}, \hat{\mathbf{d}} = \underset{h,\mathbf{d}}{\text{argmin}}\, J(h, \mathbf{d}) \tag{86}$$

Contrastive aproach (with lower bound)

- Optim with respect to h tightens the bound to approximate the JSD. Optim with respect to $\mathbf{d}$ for optimal design.
- Allows for computational savings as we only aim to approximate the JSD accurately around its maximiser $\hat{\mathbf{d}}$. (This is because we optimise iteratively, changing $\mathbf{d}$ and h as we proceed)
- Result for the ecology example:



Contrastive approach (with lower bound)

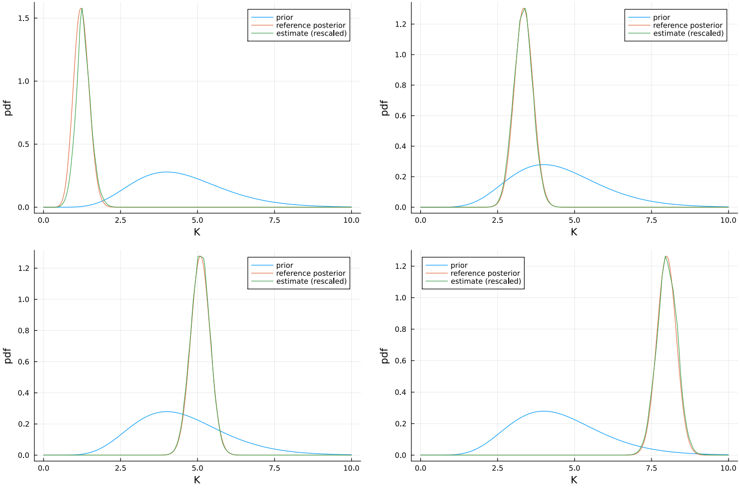
- $\hat{\mathbf{d}}$ is the optimal design.
- As before, $\hat{h}$ approximates the log-ratio of the distributions in the expectations of the logistic loss.
- Provides and estimate of the posterior: Since

$$\hat{h}(\mathbf{x}, \boldsymbol{\theta}, \mathbf{d}) \approx \log \frac{p(\mathbf{x}, \boldsymbol{\theta}|\mathbf{d})}{p(\mathbf{x}|\mathbf{d})p(\boldsymbol{\theta}|\mathbf{d})} = \log \frac{p(\boldsymbol{\theta}|\mathbf{x}, \mathbf{d})}{p(\boldsymbol{\theta}|\mathbf{d})} \tag{87}$$

  we have $\log \hat{p}(\boldsymbol{\theta}|\mathbf{x}, \mathbf{d}) = \hat{h}(\mathbf{x}, \boldsymbol{\theta}, \mathbf{d}) + \log p(\boldsymbol{\theta}|\mathbf{d})$
- Use for values of $\mathbf{d}$ around $\hat{\mathbf{d}}$. May not be accurate for other $\mathbf{d}$.
- Estimated posterior is amortised with respect to $\boldsymbol{\theta}$ and the data $\mathbf{x}$.

Contrastive approach (with lower bound)

- Ecology example: estimated posteriors for different data sets.



Contents

Summary

- Contrastive learning has two main ingredients:
  1. Learning/measuring the difference
  2. Constructing the reference
- Minimising the logistic loss allows us to learn the difference between two distributions p and q.
- Key properties:
  - $h^* = \text{argmin}_h \bar{J}(h) = \log p - \log q$
  - $\text{JSD}(p||q) \geq -\frac{1}{2}\bar{J}(h) + \log 2$ and the bound is tight for h^*.
- A number of diverse kinds of problems can be solved with contrastive learning.

Summary

1. Deep energy-based models: What learning principles can we use to efficiently estimate $\boldsymbol{\theta}$ when the model pdf $p(\mathbf{x}|\boldsymbol{\theta})$ is only available up to $Z(\boldsymbol{\theta})$?
   $\Rightarrow$ Use contrastive learning to target $\log \frac{\exp(-f_{\boldsymbol{\theta}}(\mathbf{x}))}{q(\mathbf{x})}$ where q is a preliminary model, e.g. representing our current belief about $\mathbf{x}$.
2. Inference for simulator models: How can we compute or sample from $p(\boldsymbol{\theta}|\mathcal{D})$ without access to the model pdf $p(\mathbf{x}|\boldsymbol{\theta})$?
   $\Rightarrow$ Use contrastive learning to target $\log \frac{p(\mathbf{x}|\boldsymbol{\theta})\cancel{f(\boldsymbol{\theta})}}{p(\mathbf{x})\cancel{f(\boldsymbol{\theta})}}$ where $f(\boldsymbol{\theta})$ is an auxiliary distribution.
3. Exp design for simulator models: How to obtain a design $\mathbf{d}$ that approximately maximises the expected information gain without access to the model pdf $p(\mathbf{x}|\boldsymbol{\theta}, \mathbf{d})$?
   $\Rightarrow$ Use contrastive learning to lower bound and maximise $\text{JSD}(p(\mathbf{x}, \boldsymbol{\theta}|\mathbf{d})||p(\mathbf{x}|\mathbf{d})p(\boldsymbol{\theta}|\mathbf{d}))$ with respect to $\mathbf{d}$. Targets $\log \frac{p(\mathbf{x},\boldsymbol{\theta}|\mathbf{d})}{p(\mathbf{x}|\mathbf{d})p(\boldsymbol{\theta}|\mathbf{d})}$.

Main messages

1. The likelihood function is a main workhorse in statistics and ML but becomes easily computationally intractable. ✓
2. Contrastive learning is an intuitive and computationally feasible alternative to likelihood-based approaches. ✓
3. It is broadly applicable. Here: (1) parameter estimation, (2) Bayesian inference, and (3) Bayesian experimental design. ✓

Contents

Directions to go from here

$$\underbrace{b}_{\text{reference}} + \underbrace{a - b}_{\text{difference}} \Rightarrow \underbrace{a}_{\text{interest}}$$

- Contrastive learning has two main ingredients:
  1. Learning/measuring the difference
  2. Constructing the reference
- Multiple directions are possible. Classify them broadly into three:
  1. Other loss functions to learn the difference.
  2. Construction of the reference distribution.
  3. Applications.

Other loss functions

- Other loss functions than logistic loss can be used.
- Multinomial logistic loss where we contrast more than two data points:
  - Ma and M. Collins, Conference on Empirical Methods in Natural Language Processing 2018. *Noise contrastive estimation and negative sampling for conditional models: Consistency and statistical efficiency.*
  - Srivastava et al, TMLR 2023. *Estimating the Density Ratio between Distributions with High Discrepancy using Multinomial Logistic Regression.*
- Bregman and other divergences:
  - Pihlaja et al, UAI, 2010. *A family of computationally efficient and simple estimators for unnormalized statistical models*
  - Gutmann and Hirayama, 2011. *Bregman divergence as general framework to estimate unnormalized statistical models*
  - Uehera et al, AISTATS 2020. *A Unified Statistically Efficient Estimation Framework for Unnormalized Models*

Construction of the reference distribution

- The reference depends on the problem-class studied.
- Research has mostly focussed on the case of energy-based models.
  - We can iterate and choose as reference the model from the previous iteration (Gutmann and Hyvärinen, 2010).
  - Iterate and as use as reference a normalising flow
    (Gao et al, NeurIPS 2019. Flow-contrastive estimation.)
  - Use a kernel-density estimate of the data distribution
    (Uehera et al, AISTATS 2020)
  - We can generate the reference data conditionally on the observed data
    (Ceylan and Gutmann, ICML 2019. Conditional noise-contrastive estimation of unnormalised models)
  - We can investigate which fixed reference distribution gives the smallest error
    (Chehab et al, AISTATS 2022. The optimal noise in noise-contrastive learning is not what you think)
- Adaptive construction of the reference distribution gives raise to GANs if a simulator model instead of a EBM is used.

Further applications

- Change-point detection (e.g. Puchkin et al, AISTATS 2023)
- Recommendation systems (e.g. Wu et al, SIGIR 2019)
- Representation learning, e.g. Word2Vec (Mikolov et al, 2013), InfoNCE (van den Oord, et al, arXiv:1807.03748), or SimCL (Chen et al, ICML 2020). For a recent review paper in this domain, see *A Cookbook of Self-Supervised Learning* (Balestriero et al, arXiv:2304.12210)
- Sequential experimental design

  (e.g. Ivanova et al, NeurIPS 2021. Implicit Deep Adaptive Design [...])
- ...

Conclusions

- Introduced energy-based and simulator models.
- Pointed out that their likelihood function is typically computationally intractable, which hampers inference and experimental design.
- Contrastive learning is an intuitive and computationally feasible alternative to likelihood-based approaches.
- Contrastive learning is closely related to classification, logistic regression, and ratio estimation.
- Explained how to use it to solve various difficult statistical problems:
  1. Parameter estimation for energy-based models
  2. Bayesian inference for simulator models
  3. Bayesian experimental design for simulator models
- For papers and code, see `https://michaelgutmann.github.io`