Entropy-Based Learning

Principles, Methods, and Scientific Applications

Yatao Bian

RELEASED
August 2, 2026

LAST UPDATED
August 24, 2026

Blue Whale Lab

Preface

Welcome to **Entropy-Based Learning**. This short book is an introductory, evolving companion to a research program on *modern entropy- and energy-based learning*, developed at the Blue Whale Lab led by Yatao Bian.

Entropy is one of the deepest bridges between physics, information theory, and machine learning — a single quantity reinvented, again and again, for heat engines, telephone lines, quantum ensembles, and even theories of human anxiety. Three complementary principles organize the material in this book:

- **The Maximum Entropy Principle** — among all models consistent with the observed constraints, prefer the one that is maximally noncommittal (highest entropy). This principle underlies statistical physics, Boltzmann machines, and a principled class of game-theoretic valuation methods.
- **The Minimum Entropy Principle** — when a model's own predictions can serve as a training signal, reducing predictive uncertainty can elicit capabilities such as reasoning, *without* external supervision.
- **The Minimax Entropy Principle** — maximize entropy given the features a model includes, then choose the features that make that entropy as small as possible. This couples the two principles above and turns model selection itself into a well-posed problem (Zhu et al. 1997; Carcamo et al. 2025).

Binding them together is the **free energy** $F = U - TS$, in which temperature T sets the exchange rate between energy and entropy. It is the equation that carried statistical physics into machine learning, and it makes the three principles above three settings of one dial rather than three separate doctrines.

How complete is this draft?

> ⚠ Work in Progress
>
> This book is an ongoing project, released early so you can follow its development. Most chapters are incomplete. The table below shows the current status of each chapter. Please keep this in mind before quoting or citing any part of the text, or before judging drafts that are still being written.

Chapter	Status
1 — Introduction	early draft
2 — Entropy and the Maximum Entropy Principle	**mostly complete**
3 — Energy, Entropy, and Free Energy	**mostly complete**
4 — Energy-Based Models	**mostly complete**
5 — Maximum-Entropy Learning	early draft
6 — Minimum-Entropy Learning	**mostly complete**
7 — Minimax-Entropy Learning	early draft
8 — Energy-Based Reasoning, Guidance, and Refinement	early draft
9 — Entropy-Based Learning for Science	early draft
10 — Summary	early draft

"Mostly complete" means the argument is in place and has been checked, though details may still move. "Early draft" means little more than a scaffold. The table will be updated as chapters mature. Feedback and contributions are welcome at any stage.

A printable PDF of the current draft is available from the download button in the sidebar, or directly as Entropy-Based-Learning.pdf.

To learn more about the underlying research, visit https://yataobian.com/ and the Blue Whale Lab.

Citation

If you found this book useful, please cite it as:

```
@book{bian2026entropy,
  title     = {Entropy-Based Learning},
  author    = {Yatao Bian},
  year      = {2026},
  publisher = {Blue Whale Lab},
  url       = {https://yataobian.github.io/entropy-based-learning/}
}
```

Or, for attribution in text: Bian, Yatao. 2026. *Entropy-Based Learning*. Blue Whale Lab. https://yataobian.github.io/entropy-based-learning/.

Contents

1 Introduction

1.1 Why entropy?

Modern machine learning has long been nurtured by scientific disciplines. A canonical example is the **Boltzmann machine**, rooted in statistical physics and motivated by the *Maximum Entropy Principle*. This book takes entropy — and its close relative, energy — as the organizing thread for a family of learning methods that are both principled and practically effective.

That lineage is unusually direct. In 1982 John Hopfield borrowed the energy function of a magnetic spin system to build an associative memory (Hopfield 1982); within a year, simulated annealing had turned the competition between energy and entropy into an optimization schedule (Kirkpatrick et al. 1983); and by 1985 the Boltzmann machine placed Hopfield's network at a finite temperature and learned its couplings from data (Ackley et al. 1985). The same trade-off resurfaced as the evidence lower bound (Dayan et al. 1995; Neal and Hinton 1998), as the contrastive objectives of energy-based learning (LeCun et al. 2006), and as the temperature parameter of every language-model decoder. In 2024 the Nobel Prize in Physics was awarded to Hopfield and Hinton for this line of work (The Royal Swedish Academy of Sciences 2024). Chapter 3 develops the single equation, $F = U - TS$, that runs beneath all of it.

We study three directions that are connected:

1. **Maximizing entropy** to obtain the least-biased model consistent with given constraints. This yields exponential-family / Gibbs distributions and, as we will see, a principled framework for valuation problems in machine learning (Bian et al. 2022).
2. **Minimizing entropy** to sharpen a model's own predictions. When applied to large language models in a latent semantic space, this idea enables fully *unsupervised* elicitation of reasoning capabilities.
3. **Minimaxing entropy** to decide which features a model should include at all. The optimal features are those whose maximum-entropy model has the *minimum* entropy, which makes model selection a well-posed problem (Carcamo et al. 2025).

1.2 Roadmap

- **Part I — Foundations** surveys the thermodynamic, information-theoretic, quantum, and psychological notions of entropy, develops the maximum-entropy principle together with the free-energy relationship $F = U - TS$ that connects entropy to energy, and introduces energy-based models and the minimum-description-length view that leads to minimax entropy.

- **Part II — Methods** develops maximum-entropy, minimum-entropy, and minimax-entropy learning, together with energy-based reasoning / guidance / refinement.
- **Part III — Applications** connects these ideas to scientific foundation models and AI-for-science.

Part I

Part I — Foundations

2 Entropy and the Maximum Entropy Principle

2.1 The many faces of entropy

Entropy was invented three separate times — for heat engines in the 1860s, for quantum ensembles in the 1920s, and for telephone lines in the 1940s — and has been rediscovered many times since. The names, the units, and the motivating problems all differ, but each version measures the same thing: given what we have specified about a system, how much remains unspecified. This section walks through those notions before the rest of the book settles on one of them.

2.1.1 Information: Shannon entropy

For a discrete random variable X with probability mass function $p(x)$, the **Shannon entropy** is

$$H(p) = -\sum_{x} p(x) \log p(x),$$

which quantifies the average uncertainty (in nats or bits) of X (Shannon 1948). Entropy is maximized by the uniform distribution, where it equals $\log |\mathcal{X}|$, and minimized (equal to zero) by a deterministic one.

This functional form is not an arbitrary choice. Shannon showed that three requirements — continuity in the probabilities, monotonicity in the number of equally likely outcomes, and consistency when outcomes are grouped — determine H uniquely up to the base of the logarithm. It also carries a concrete operational meaning: $H(p)$ is the expected code length of an optimal lossless encoding of samples from p — exactly, in the limit of coding long blocks, and to within one bit if each symbol must be coded on its own. Uncertainty and description length are one quantity seen from two sides, a duality that returns in Chapter 7.

2.1.2 Heat: thermodynamic entropy

The original entropy is older and, on its face, unrelated to information. Studying the irreversibility of heat flow, Clausius defined a state function whose differential along a reversible path is

$$dS = \frac{\delta Q_{\text{rev}}}{T},$$

and named it from the Greek $\tau\rho o\pi\square$, transformation, deliberately shaping the word to rhyme with *energy* because the two quantities seemed to him so closely related (Clausius 1865). That 1865 paper closes with the two sentences that became the standard statement of the laws of thermodynamics: the energy of the universe is constant, and its entropy tends towards a maximum.

Boltzmann supplied the microscopic reading. A macrostate — a temperature, a pressure — is compatible with an enormous number W of microstates, and entropy simply counts them (Boltzmann 1877):

$$S = k_B \log W.$$

Gibbs then generalized the count to microstates that are not equally likely (Gibbs 1902):

$$S = -k_B \sum_i p_i \log p_i.$$

The Gibbs expression and the Shannon expression are the same formula, separated only by the constant k_B that carries the physical units. The coincidence is not accidental, and Jaynes turned it into the argument developed in Section 2.2: thermodynamic entropy is the Shannon entropy of our ignorance about the microstate, which makes equilibrium statistical mechanics a problem of inference rather than of mechanics (Jaynes 1957).

2.1.3 Quantum states: von Neumann entropy

Quantum mechanics replaces a probability distribution with a density matrix ρ, a positive semidefinite operator of unit trace. Von Neumann extended entropy to this setting in 1927, two decades before Shannon (Neumann 1927):

$$S(\rho) = -\operatorname{Tr}(\rho \log \rho).$$

Written in the eigenbasis of ρ, this reduces to the Shannon entropy of its eigenvalues, so the same functional form reappears with a different argument. It vanishes exactly for pure states, which are completely specified, and is maximal for the maximally mixed state.

The quantum case does break the analogy in one instructive place. Classically, conditional entropy is never negative: learning Y cannot leave us knowing more than everything about X. Quantum mechanically it can be — the joint state may be more sharply specified than either of its parts. A negative value therefore certifies entanglement, though the implication runs one way only: entangled states with non-negative conditional entropy exist.

2.1.4 One knob, a family: Rényi and Tsallis

Shannon's requirements can be relaxed, and each relaxation produces a one-parameter family. Rényi kept additivity over independent systems but dropped the requirement that entropy be a linear average of $-\log p$ (Rényi 1961):

$$H_\alpha(p) = \frac{1}{1-\alpha} \log \sum_x p(x)^\alpha, \qquad \alpha > 0,\ \alpha \neq 1.$$

The order α is a knob controlling how much weight rare outcomes receive. The limit $\alpha \to 1$ recovers Shannon entropy; $\alpha = 0$ gives the Hartley entropy $\log |\operatorname{supp} p|$, which counts possibilities and ignores their probabilities; $\alpha = 2$ gives the **collision entropy**

$$H_2(p) = -\log \sum_x p(x)^2,$$

whose argument $\sum_x p(x)^2$ is the probability that two independent draws coincide — a quantity that returns as a training objective in Section 6.5; and $\alpha \to \infty$ gives the **min-entropy**

$$H_\infty(p) = -\log \max_x p(x),$$

the worst-case measure used in cryptography, where what matters is not average uncertainty but the chance that an adversary guesses right on the first try.

A warning about that last name, since both readings appear later in this book. The min-entropy H_∞ is a *specific functional*, the most pessimistic member of the Rényi family. It should not be confused with the **minimum-entropy principle** of Chapter 6, which is not a functional at all but a class of learning *objectives* that reduce the entropy — usually the Shannon or the collision entropy — of a model's predictive distribution. The two are unrelated, and the shared prefix is an accident of terminology in two literatures that developed apart.

Tsallis relaxed the other requirement, giving up additivity itself (Tsallis 1988):

$$S_q(p) = \frac{1}{q-1}\Big(1 - \sum_x p(x)^q\Big).$$

Here $q \to 1$ again recovers the Boltzmann–Gibbs form, but for $q \neq 1$ two independent systems combine as $S_q(A, B) = S_q(A) + S_q(B) + (1-q)\, S_q(A)\, S_q(B)$. This *non-extensivity* was introduced to describe systems with long-range interactions and multifractal structure, where the usual assumption that entropy scales with system size fails.

Both families are useful in their own domains, and both make the same point by contrast: Shannon entropy is the unique member satisfying the full set of Shannon–Khinchin axioms, including the chain rule that makes conditional entropy and mutual information well behaved. That is why it is the form used throughout the remainder of this book.

2.1.5 Minds: psychological entropy

The most surprising reappearance is in psychology. Hirsh, Mar, and Peterson propose the **entropy model of uncertainty**, which applies Shannon's formula directly to the human information system (Hirsh et al. 2012). At any moment an organism faces a set of competing *affordances*: possible interpretations of the sensory input, and possible actions to take. The model treats these as a subjectively weighted probability distribution and defines **psychological entropy** as

its Shannon entropy.

The reading is unusually literal. A familiar situation places nearly all the weight on one dominant affordance, giving low entropy and a fast habitual response. An unfamiliar or ambiguous one spreads the weight across many competing frames, giving high entropy and sustained neural competition — which, the authors argue, is what anxiety *is* subjectively, with anterior cingulate activity and noradrenaline release as its physiological correlates.

Two features of the account matter for what follows. First, goals act as constraints: adopting a goal reweights the distribution towards the actions that serve it, and thereby lowers entropy. This is structurally the same move as imposing moment constraints in Section 2.2, since a constraint is whatever narrows the space of live possibilities. Second, the model makes entropy reduction a *drive* rather than a calculation. That idea runs from Schrödinger's argument that organisms persist by exporting entropy to their environment (Schrödinger 1944) through to the free-energy principle (Friston 2010), and it is the biological precedent for the minimum-entropy learning objectives of Chapter 6.

2.1.6 One form, many readings

Notion	Probabilities over	Definition	Origin
Thermodynamic	no explicit distribution	$dS = \delta Q_{\text{rev}}/T$	Clausius, 1865
Statistical	microstates, equally likely	$k_B \log W$	Boltzmann, 1877
Gibbs	microstates, unequally likely	$-k_B \sum_i p_i \log p_i$	Gibbs, 1902
Information	a random variable or message source	$-\sum_x p(x) \log p(x)$	Shannon, 1948
Quantum	eigenvalues of a density matrix	$-\operatorname{Tr}(\rho \log \rho)$	von Neumann, 1927
Rényi	any distribution, at order α	$\frac{1}{1-\alpha} \log \sum_x p(x)^\alpha$	Rényi, 1961
Tsallis	non-extensive systems, at order q	$\frac{1}{q-1}(1 - \sum_x p(x)^q)$	Tsallis, 1988
Psychological	competing perceptions and actions	$-\sum_x p(x) \log p(x)$	Hirsh et al., 2012

Two observations follow. The first is mathematical: almost every row is the same functional $-\sum p \log p$, differing only in what the probabilities range over and what constant multiplies the result. Entropy is less a family of analogies than a single quantity applied to different objects.

The second is interpretive, and it is the one that does work in this book. The same number reads as dispersal of energy to a physicist, as expected code length to an engineer, and as felt uncertainty to a psychologist. Those readings are what make the two principles that follow feel necessary rather than arbitrary: maximizing entropy is being honest about what we have not specified, and minimizing it is committing to what we have.

2.2 The maximum entropy principle

Given constraints — typically moment constraints of the form $\mathbb{E}_p[f_k(X)] = \mu_k$ — the **maximum entropy** distribution solves

$$\max_p \; H(p) \quad \text{s.t.} \quad \sum_x p(x) = 1, \;\; \mathbb{E}_p[f_k(X)] = \mu_k.$$

Its solution is the **Gibbs / Boltzmann** distribution

$$p(x) = \frac{1}{Z(\lambda)} \exp\Big(\textstyle\sum_k \lambda_k f_k(x) \Big),$$

where λ are Lagrange multipliers and Z is the partition function. This is the least-biased distribution consistent with what we know (Jaynes 1957).

The single most important instance of this framework is the one in which the constrained feature is the energy of a physical system. That case, worked out in Chapter 3, is what turns the multipliers above into a temperature and connects the whole construction to thermodynamics.

2.3 Minimum description length and minimax entropy

The maximum-entropy principle can be *derived* rather than postulated. Under the **minimum description length** (MDL) view (Grünwald 2007), a model P encodes each state with a code word of length $-\log P(x)$, and its description length is the expected code length across all datasets consistent with the measured features. Among models matching those features, description length is minimized by the maximum-entropy model (Feder 1986).

MDL also poses a question that maximum entropy alone cannot answer: which features should be measured in the first place? For maximum-entropy models the description length coincides with the model's own entropy,

$$L(P_F) = S(P_F),$$

so the best feature set F is the one whose maximum-entropy model has the *smallest* entropy. This is the **minimax entropy** principle, developed in Chapter 7.

2.4 Why it matters for learning

The maximum-entropy principle explains why exponential families are ubiquitous, grounds the Boltzmann machine in statistical physics, and — as developed in Bian et al. (2022) — provides a principled basis for defining valuations in machine learning. The next chapter supplies the missing half of the picture: what happens when entropy is traded off against energy, and what the exchange rate between them turns out to be.

3 Energy, Entropy, and Free Energy

The previous chapter maximized entropy subject to abstract moment constraints and obtained a Gibbs distribution written in terms of Lagrange multipliers. The next chapter writes probabilities as $p(x) \propto e^{-E(x)}$ and calls the exponent an energy. Something has to connect the two, and that something is one of the most useful equations in science:

$$F = U - TS.$$

At a fixed temperature, nature neither minimizes energy nor maximizes entropy. It minimizes a combination of the two, and the temperature is the exchange rate between them. This chapter derives that statement, illustrates it with three classical examples, and follows it into artificial intelligence.

3.1 Maximum entropy with an energy constraint

The maximum-entropy problem of Section 2.2 was stated for arbitrary features f_k. Historically the original problem had exactly one constraint: the average energy of the system is known to be U. Written out,

$$\max_{p} \; H(p) \quad \text{s.t.} \quad \sum_{x} p(x) = 1, \;\; \sum_{x} p(x)\, E(x) = U,$$

and solved with Lagrange multipliers, it yields the **Boltzmann distribution**

$$p(x) = \frac{1}{Z}\, e^{-\beta E(x)}, \qquad Z(\beta) = \sum_{x} e^{-\beta E(x)},$$

where the multiplier enforcing the energy constraint is the inverse temperature, $\beta = 1/k_B T$.

That identification is worth pausing on, because it is what gives temperature a meaning outside physics. A Lagrange multiplier is a *price*: it measures how much of the objective must be surrendered to buy one unit of the constrained quantity. Here it measures how much entropy must be given up to raise the expected energy by one unit. Temperature is that price. A system at high temperature buys energy cheaply and stays spread out; a system at low temperature pays dearly for it and concentrates on low-energy states.

This also settles a notational discrepancy the attentive reader will have noticed. In Section 2.2 the solution appeared as $p(x) \propto \exp\big(\sum_k \lambda_k f_k(x)\big)$, with a plus sign; here it appears as $\exp(-\beta E(x))$, with a minus. The two are the same object under the correspondence $E = -\sum_k \lambda_k f_k$. Energy is negative log-probability up to a constant, and the sign convention comes

from physics, where stable configurations should sit at *low* energy. Chapter 4 follows the physics convention and absorbs β into E by setting $k_B T = 1$.

3.2 Free energy

Given the Boltzmann distribution, the two quantities of interest follow at once: the average energy $U = \langle E \rangle$ and the entropy $S = -k_B \sum_x p(x) \log p(x)$. The combination that governs equilibrium is the **Helmholtz free energy** $F = U - TS$, and it has a compact closed form (Callen 1985):

$$F = -k_B T \log Z.$$

Free energy is the log partition function. What Chapter 4 will present as the central computational obstacle of energy-based modelling is, read from the other side, the central thermodynamic potential of the system.

The relationship has a third face, and it is the one that matters most for learning. Let q be any distribution over the same states, and define its **variational free energy**

$$F(q) = \langle E \rangle_q - T\, S(q).$$

A short calculation relates this to the equilibrium value:

$$F(q) = F + k_B T \cdot \mathrm{KL}(q \,\|\, p),$$

where p is the Boltzmann distribution. Because the Kullback–Leibler divergence is non-negative, $F(q) \geq F$ with equality exactly when $q = p$. Equilibrium is therefore not merely *described* by the Boltzmann distribution, it is *characterized* by it, as the unique minimizer of a variational objective. Nearly everything in later chapters that goes by the name of a variational bound is this one inequality in disguise.

3.3 What entropy buys you

The trade-off is easiest to believe after seeing cases in which entropy alone decides the outcome.

3.3.1 Isothermal expansion: a process with no energy change

An ideal gas of N particles expands from volume V_1 to V_2 while held at temperature T. The internal energy of an ideal gas depends only on temperature, so $\Delta U = 0$: not one joule of stored energy is released. The gas expands regardless, because

$$\Delta S = N k_B \log \frac{V_2}{V_1} > 0 \quad \Longrightarrow \quad \Delta F = -T\Delta S < 0.$$

The work extracted, $W = N k_B T \log(V_2/V_1) = T\Delta S$, is drawn entirely from the heat bath. An account of this process in terms of energy alone cannot even say which direction it runs.

3.3.2 The rubber band: a force made of entropy

Stretch a rubber band and it pulls back. The restoring force is not stored in stretched chemical bonds. A polymer chain is a random walk of N segments of length a, and there are vastly more coiled configurations than extended ones, so extending the chain to end-to-end length L costs entropy. The force is the gradient of that cost:

$$f = -T\frac{\partial S}{\partial L} \approx \frac{3k_B T}{N a^2}\, L.$$

The energy landscape here is essentially flat, and yet there is a real force, one measurable with a spring scale. The formula also makes two predictions that run against intuition, and both hold. The force is *proportional to temperature*, so heating a stretched rubber band makes it contract harder — the opposite of a metal rod, which expands. And stretching a band quickly warms it, an effect readily confirmed by holding one against your lip.

3.3.3 Melting: where the competition changes hands

Ice has lower energy than liquid water, since melting must pay $\Delta H > 0$ to break hydrogen bonds; water has the higher entropy. At the melting point the two contributions balance exactly, $\Delta H - T_m \Delta S = 0$, so

$$T_m = \frac{\Delta H}{\Delta S}.$$

Below T_m the energy term dominates and ice is the stable phase; above it the entropy term dominates and water is. A phase transition is precisely the temperature at which the exchange rate tips the competition from one side to the other.

3.4 From spins to networks

The simplest system with an energy is a single two-state variable $s = \pm 1$ with energy $\mp\varepsilon$. Its partition function and average value are

$$Z = 2\cosh(\beta\varepsilon), \qquad \langle s \rangle = \tanh(\beta\varepsilon),$$

and the temperature limits are exactly what the trade-off predicts: as $T \to 0$ the variable freezes into its low-energy state, and as $T \to \infty$ it becomes a fair coin with entropy $k_B \log 2$.

Now couple many such variables together. The **Ising model** assigns to a configuration $s = (s_1, \dots, s_N)$ the energy

$$E(s) = -\sum_{i<j} J_{ij}\, s_i s_j - \sum_i h_i s_i,$$

with J_{ij} the interaction between units i and j and h_i a local bias. When the couplings are disordered, the landscape acquires a large number of local minima and the system is called a *spin glass*.

That equation is the hinge of this chapter. It is a model of magnetism, and it is also, term for term, the energy function of a Hopfield network and of a Boltzmann machine. What follows is what happened when physicists noticed.

3.5 How this reached AI

Year	Contribution	Role of the entropy–energy trade-off
1982	Hopfield network (Hopfield 1982)	The Ising energy becomes an associative memory: stored patterns are local minima and recall is descent. The $T = 0$ limit — energy only
1983	Simulated annealing (Kirkpatrick et al. 1983)	Turns the trade-off into a *schedule*: start hot so entropy drives exploration, then cool so energy drives commitment
1985	Boltzmann machine (Ackley et al. 1985)	Places the Hopfield network at finite temperature, $p(s) \propto e^{-E(s)/T}$; learning compares correlations in a clamped and a free phase
1995–98	Helmholtz machine (Dayan et al. 1995); free-energy view of EM (Neal and Hinton 1998)	The evidence lower bound is identified as a negative variational free energy
2002	Contrastive divergence (Hinton 2002)	Makes the intractable $\log Z$ gradient practical and revives deep generative models
2006	Energy-based learning (LeCun et al. 2006)	Names the log-partition term "the Helmholtz free energy of the ensemble" and unifies probabilistic and non-probabilistic learning

Year	Contribution	Role of the entropy–energy trade-off
2024	Nobel Prize in Physics (The Royal Swedish Academy of Sciences 2024)	Awarded to Hopfield and Hinton; the citation rests on the correspondence between neural networks and spin models in statistical physics

Two threads in that table deserve to be stated plainly, because the rest of the book leans on them.

The first is that **the evidence lower bound is a free energy**. Take a latent variable z, set $E(z) = -\log p(x, z)$ and $T = 1$, and the variational inequality of the previous section reads

$$-\log p(x) \;\le\; \big\langle -\log p(x, z) \big\rangle_q - H(q) \;=\; F(q),$$

with the gap equal to $\mathrm{KL}(q \,\|\, p(\cdot \mid x))$. Variational autoencoders, variational inference, and the training objectives of diffusion models are all instances of minimizing $U - TS$ at unit temperature. The physics is not an analogy here; it is the same algebra.

The second is that **annealing is the trade-off used as a control signal**. Simulated annealing, the noise schedules of Langevin and diffusion samplers, and the practice of decoding a language model at a temperature that falls during generation are one idea: begin where entropy dominates and the search is broad, end where energy dominates and the answer is definite.

3.6 The temperature dial

The modern instance is worth making explicit, because it is the version most readers use daily. A language model produces logits $\ell(x)$ and samples

$$p(x) \propto \exp\big(\ell(x)/T\big),$$

which is the Boltzmann distribution with energy $E = -\ell$. Greedy decoding is $T \to 0$; sampling uniformly from the vocabulary is $T \to \infty$. The `temperature` field in every inference API is the same T that appears in $F = U - TS$.

Read this way, the three principles that organize this book are three settings of a single dial.

Principle	Where the dial sits	What it buys
Maximum entropy (Chapter 5)	High: entropy dominates	Honesty about what the constraints leave undetermined
Minimum entropy (Chapter 6)	Low: energy dominates	Commitment — sharpened predictions and elicited reasoning

Principle	Where the dial sits	What it buys
Minimax entropy (Chapter 7)	Both, nested	Maximize on the inside to model the world, minimize on the outside to choose what to model

The principles are not competing claims about the right amount of uncertainty. They are claims about *when* to be uncertain, and the free energy is what makes that question well posed.

3.7 Why it matters for learning

The equation $F = U - TS$ supplies three things the rest of this book uses. It explains where the Gibbs form of Chapter 4 comes from and why its exponent carries a minus sign. It identifies the partition function — the central computational difficulty of energy-based modelling — as a free energy, which is why variational bounds are the natural way around it. And it gives *temperature* a precise meaning, so that annealing schedules, decoding parameters, and the coalition temperature τ of Chapter 5 can all be recognized as one knob rather than three coincidences.

The next chapter takes the Gibbs distribution as its starting point and asks what happens when the energy function is learned from data.

4 Energy-Based Models

The previous chapter left us with an open question: what changes when the Gibbs distribution's energy function is no longer prescribed by physics, but instead *learned from data*? This chapter answers that question.

Two things change the moment the energy becomes learnable. First, the energy stops being handed to us by physics — the Ising couplings, the hydrogen bonds of Chapter 3 — and becomes a neural network $E_\theta(x)$ whose parameters we must fit. Second, the partition function, which for a physicist is a bookkeeping device, becomes for us the central computational villain of the story: nearly every algorithm in this chapter is best understood as a different strategy for *not computing it*.

An **energy-based model (EBM)** associates to each configuration x a scalar energy $E_\theta(x)$ and defines the Gibbs distribution

$$p_\theta(x) = \frac{e^{-E_\theta(x)}}{Z(\theta)}, \qquad Z(\theta) = \int e^{-E_\theta(x)}\, dx, \tag{4.1}$$

with a sum in place of the integral when x is discrete. This is the Boltzmann distribution of Chapter 3 at unit temperature, with a learnable energy in place of a physical one. Low energy means high probability; the sign convention and the absorbed temperature were settled at the end of that chapter.

The material here draws chiefly on two tutorials, written fifteen years apart and complementary in spirit. The first (LeCun et al. 2006) treats the energy as a *decision surface*: inference is energy minimization, learning is shaping the landscape, and probabilities are an optional purchase. The second (Song and Kingma 2021) treats the energy as a *density*: the model is Equation 4.1 taken literally, and the question is how to estimate it when $Z(\theta)$ cannot be computed. Reading them together — decision surface first, density second — is the plan of this chapter. A regularly updated companion list of papers, tutorials, and software is maintained at *Awesome-EBM* (Bian 2020).

4.1 The energy view of inference

Start with the decision surface. In the energy-based view, a model is a function $E_\theta(x, y)$ that measures the *compatibility* between an observed input x and a candidate answer y: small energy means "these two belong together," large energy means "they do not." Inference is then not a forward pass but an *optimization*:

$$y^* = \arg\min_{y \in \mathcal{Y}} E_\theta(x, y). \tag{4.2}$$

When $\mathcal{Y}$ is a handful of class labels, the arg min is a table lookup and Equation 4.2 is just classification wearing unfamiliar clothes. The formulation earns its keep when $\mathcal{Y}$ is large or structured: all consistent segmentations of an image, all sentences of English, all conformations of a protein. In those cases no model can afford to enumerate answers; what it can afford is to *score* any proposed answer, and to search — by dynamic programming, by gradient descent on a continuous y, by annealing — for a good one. The energy function is precisely the object that makes such a search well posed.

This view immediately clarifies what a model owes us, and it is less than one might think. LeCun et al. distinguish the questions a model may be asked (LeCun et al. 2006):

1. **Decision.** "Which y is best for this x?" Only the *ordering* of energies matters. Any strictly increasing transform of E_θ gives the same answers.
2. **Ranking.** "Is y_1 better than y_2?" Again only comparisons matter.
3. **Detection.** "Is this y good enough?" Now energies are compared to a threshold, so their *scale* begins to matter.
4. **Probability.** "How confident are we in each y?" Only here must energies be converted into calibrated numbers that sum to one.

Only the fourth question requires the Gibbs conversion

$$p_\theta(y \mid x) = \frac{e^{-\beta E_\theta(x,y)}}{\int_{y'} e^{-\beta E_\theta(x,y')}\, dy'}, \tag{4.3}$$

(a sum over y' when $\mathcal{Y}$ is discrete), restoring the inverse temperature β of Chapter 3 as an explicit dial. The rest of the chapter works at $\beta = 1$ unless a limit in β is at issue, matching the unit-temperature convention of Equation 4.1. The denominator is the partition function — the free-energy bill of Chapter 3. This is the first strategic lesson of energy-based learning: *probabilities are expensive, and you should buy them only when the application actually needs them* (LeCun et al. 2006). A robot that must steer left or right needs the arg min, not a posterior. Much of this chapter is about what becomes possible when we decline to pay, and what it costs to pay when we must.

4.2 Why learn an energy at all?

If unnormalized models create such trouble, why prefer them? Three reasons, in increasing order of depth.

Freedom of form. A normalized density must integrate to one, and maintaining that constraint dictates the architecture: autoregressive models must factor into conditionals, normalizing flows must be invertible with tractable Jacobians, variational autoencoders must route everything through a directed latent-variable structure. An energy function must merely be a scalar-valued

function, bounded below well enough for Equation 4.1 to make sense. Any regression architecture qualifies — a convolutional network for images, a graph neural network for molecules, a transformer for text. In the words of Song and Kingma, density estimation is thereby "reduced to a nonlinear regression problem" (Song and Kingma 2021). The modeling question stops being "what distributions can my architecture normalize?" and becomes "what shape should the landscape have?"

Compatibility is often what we actually know. Domain knowledge usually arrives as statements about what configurations are good — physical constraints, grammatical constraints, consistency conditions — not as recipes for sampling. An energy function absorbs such knowledge directly, as terms that raise the energy of violating configurations.

Energies add. If E_1 scores grammaticality and E_2 scores factual accuracy, then $E_1 + E_2$ scores both at once, and at the level of distributions the sum of energies is the *product* of the corresponding Gibbs factors — a product of experts (Hinton 2002), in which each expert can veto a candidate by assigning it high energy. Normalized models do not compose this way: the product of two normalized densities is not normalized, and renormalizing it is exactly the partition-function problem again. This additivity is what makes energies the natural currency for *inference-time* composition — steering a generator with extra constraints — a theme that returns in force in Chapter 8.

The price of this freedom is stated by Equation 4.1 itself: $Z(\theta)$ is an integral over every configuration the model can represent, and for any interesting architecture it is hopelessly intractable. Neither exact likelihoods nor exact samples are available. Everything that follows is about living well anyway.

4.3 Learning is shaping a landscape

What should a learned energy landscape look like? Configurations resembling the training data should sit in valleys; everything else should sit high. The subtlety — and it is *the* central subtlety of EBM training — is that a learning rule which only digs valleys will destroy the landscape.

Consider the most innocent possible objective, the **energy loss**: for each training pair (x^i, y^i), minimize $E_\theta(x^i, y^i)$. Push down on the data; done. For some architectures this works — in least-squares regression, $E_\theta(x, y) = \|y - G_\theta(x)\|^2$, pushing the correct answer's energy toward zero automatically raises every other answer's, because the architecture allows only one minimum per input. But give the model more freedom and the innocent objective finds the cheat: make the energy *low everywhere*. A constant landscape assigns the data the lowest possible energy and is perfectly useless — it has **collapsed** (LeCun et al. 2006). Flat landscapes answer every question with "everything is equally fine."

The cure is as old as the perceptron: every push down on the correct answer must be paired with a **pull up** on incorrect ones. Losses that implement this pairing are called *contrastive*, and the zoo of them is organized by one question — *which* incorrect answers get pulled up, and how hard?

The most offending incorrect answer. The sharpest choice is to pull up only on the incorrect

answer the model currently likes best, $\bar{y}^i = \arg\min_{y \neq y^i} E_\theta(x^i, y)$ — the *most offending incorrect answer* (LeCun et al. 2006). The **perceptron loss** $E_\theta(x^i, y^i) - \min_y E_\theta(x^i, y)$ does this but demands no gap, and so can still be satisfied by a nearly flat landscape. Margin losses fix that. The **hinge loss** $\max\left(0,\ m + E_\theta(x^i, y^i) - E_\theta(x^i, \bar{y}^i)\right)$ insists the correct answer sit at least m below its closest rival — the loss of support vector machines, re-read as landscape shaping. The **log loss** $\log\left(1 + e^{E_\theta(x^i, y^i) - E_\theta(x^i, \bar{y}^i)}\right)$ is its soft, infinite-margin sibling. The **square–square loss** $E_\theta(x^i, y^i)^2 + \max(0, m - E_\theta(x^i, \bar{y}^i))^2$ pins correct answers near zero and rivals above m, which is natural when the energy is a distance and hence bounded below.

All incorrect answers at once. The other extreme is the **negative log-likelihood (NLL) loss**. Write the Gibbs conversion Equation 4.3, take $-\log$, and per sample:

$$\mathcal{L}_{\text{nll}} = E_\theta(x^i, y^i) \ + \ \frac{1}{\beta} \log \int_y e^{-\beta E_\theta(x^i, y)}\, dy. \tag{4.4}$$

(Replace the integral by a sum when $\mathcal{Y}$ is finite.) The second term is *minus* the Helmholtz free energy $F = -(1/\beta)\log \int e^{-\beta E}\, dy$ of Chapter 3 — at $\beta = 1$ it is $\log Z$, not F itself. The first term pushes down on the data; the $\log Z$ term pulls up on *every* answer. How hard? Differentiate (at $\beta = 1$):

$$\frac{\partial \mathcal{L}_{\text{nll}}}{\partial \theta} = \frac{\partial E_\theta(x^i, y^i)}{\partial \theta} - \int_y p_\theta(y \mid x^i)\, \frac{\partial E_\theta(x^i, y)}{\partial \theta}\, dy. \tag{4.5}$$

Each answer is pulled up with force proportional to *the probability the model currently assigns it*. The landscape's own valleys attract the correction: wherever the model is confidently wrong, the pull is strongest. This is a beautiful rule and an expensive one — the integral in Equation 4.5 is an expectation over the model distribution, which is the partition-function problem in gradient form. Two limits knit the zoo together (LeCun et al. 2006). As $\beta \to \infty$, $(1/\beta)\log \int e^{-\beta E}\, dy \to -\min_y E_\theta(x^i, y)$, so Equation 4.4 becomes the perceptron loss (only the minimum-energy rival matters). And when $\mathcal{Y}$ has just two elements $\{y^i, \bar{y}^i\}$, the $\beta = 1$ case collapses algebraically onto the log loss: $E(y^i) + \log\left(e^{-E(y^i)} + e^{-E(\bar{y}^i)}\right) = \log\left(1 + e^{E(y^i) - E(\bar{y}^i)}\right)$.

> **i** Collapse is a recurring character
>
> Hold on to the failure mode, because this book meets it twice more. In Section 6.6, a language model trained on its own agreement signal can collapse to confident nonsense — entropy minimization with the pull-up term missing. And in modern self-supervised representation learning, joint-embedding architectures collapse to constant embeddings unless a contrastive or regularization term props the landscape up (LeCun 2022; Dawid and LeCun 2024). The lesson is always the same: *an objective that only rewards fitting what you see will be gamed by a model that makes everything look equally good.* The interesting design space is in how — and how cheaply — you pull up.

The remainder of the chapter follows the density view (Song and Kingma 2021): three families of estimators, distinguished by how they approximate or evade the pull-up integral of Equation 4.5. In caricature: **sample it** (maximum likelihood with MCMC), **differentiate it away** (score

matching), or **turn it into a classification problem** (noise contrastive estimation).

4.4 Strategy one: sample the pull-up

Take maximum likelihood seriously and estimate the intractable term by Monte Carlo. For the unconditional model Equation 4.1, the log-likelihood gradient splits into the same two forces as Equation 4.5:

$$\nabla_\theta \log p_\theta(x) = -\nabla_\theta E_\theta(x) \; + \; \mathbb{E}_{\tilde{x}\sim p_\theta}\big[\nabla_\theta E_\theta(\tilde{x})\big]. \tag{4.6}$$

The derivation is worth doing once by hand, because the punchline is neat. Start from $\nabla_\theta \log Z(\theta)$, use the chain rule, and watch the model distribution assemble itself:

$$\nabla_\theta \log Z = \frac{1}{Z}\nabla_\theta \int e^{-E_\theta(x)} dx = \int \frac{e^{-E_\theta(x)}}{Z}\big(-\nabla_\theta E_\theta(x)\big)\, dx = -\,\mathbb{E}_{x\sim p_\theta}\big[\nabla_\theta E_\theta(x)\big]. \tag{4.7}$$

The intractable normalizer has become an *expectation under the model* — not computable in closed form, but estimable from a single model sample, without bias. Substituting into $\nabla_\theta \log p_\theta = -\nabla_\theta E_\theta - \nabla_\theta \log Z$ gives Equation 4.6.

Read the two terms as the two phases of learning. The **positive phase** lowers the energy of real data. The **negative phase** raises the energy of whatever the model itself likes to produce — its *fantasies*, in the old Boltzmann-machine vocabulary, where the phases were called *clamped* and *free* (Ackley et al. 1985) (the 1985 row of the table in Chapter 3). Learning stops exactly when the fantasies become statistically indistinguishable from the data, at which point the two forces cancel in expectation. Confabulate, compare, correct.

The entire difficulty hides in four words: *sample from the model*. The standard tool is **Langevin dynamics**, which needs only the gradient of the log-density in x — and here EBMs catch a genuine break, because the normalizer does not depend on x:

$$\nabla_x \log p_\theta(x) = -\nabla_x E_\theta(x) - \underbrace{\nabla_x \log Z(\theta)}_{=0} = -\nabla_x E_\theta(x). \tag{4.8}$$

This quantity — the gradient of the log-density with respect to the *input*, not the parameters — is called the **score**, and Equation 4.8 says an EBM hands it to us free of charge. Langevin MCMC then iterates

$$x_{k+1} = x_k - \frac{\epsilon^2}{2}\nabla_x E_\theta(x_k) + \epsilon\, z_k, \qquad z_k \sim \mathcal{N}(0, I)\,: \tag{4.9}$$

roll downhill, but shake while doing so. The reader will recognize finite-temperature descent — energy pulling toward minima, injected noise buying the entropy that keeps the walker exploring (Chapter 3). The *ratio* of noise amplitude to drift sets the temperature; Equation 4.9 is the unit-temperature case already built into Equation 4.1. The step size ϵ is a discretization

parameter, not a temperature: as $\epsilon \to 0$ and the number of steps grows, the iterates distribute as p_θ (Song and Kingma 2021).

"Long enough" is the problem: fresh chains per gradient step are ruinously slow, so practice truncates. **Contrastive divergence (CD)** starts chains at data points and runs a handful of steps (Hinton 2002); **persistent CD** never restarts, carrying chains across parameter updates (Tieleman 2008); replay buffers reinitialize chains from a reservoir of past samples (Du and Mordatch 2019); short-run MCMC accepts non-convergence and reinterprets the truncated sampler itself as the generative model (Nijkamp et al. 2019). These approximations power most large-scale EBM image models, but the bias of truncated chains is real, and taming it remains a research area of its own (Song and Kingma 2021).

4.5 Strategy two: match slopes, not heights

Here is a different idea, and it begins with an observation about Equation 4.8. The score $\nabla_x \log p_\theta(x)$ does not contain $Z(\theta)$ at all. Two normalized densities with the same score everywhere on a connected domain are the same density — if their log-densities have equal gradients, they differ by a constant, and normalization forces that constant to zero. (When the domain falls into well-separated pieces the argument fails, and that failure is the mode-weight blind spot below.) So instead of matching *heights* of the log-density (which requires knowing sea level, i.e. Z), match *slopes*. Picture two hikers comparing terrain maps: they need not agree on altitude above sea level to verify they hold maps of the same mountain — agreeing on the steepness and direction of the slope at every point is enough.

Score matching (Hyvärinen 2005) makes this precise by minimizing the **Fisher divergence** between data and model:

$$D_F(p_{\text{data}} \,\|\, p_\theta) = \mathbb{E}_{p_{\text{data}}}\left[\tfrac{1}{2}\left\|\nabla_x \log p_{\text{data}}(x) - \nabla_x \log p_\theta(x)\right\|^2\right]. \tag{4.10}$$

An objection leaps out: Equation 4.10 contains the score of the *data distribution*, which nobody knows. The resolution is a small miracle of integration by parts, and it is worth seeing in one dimension. Expand the square in Equation 4.10; the term $\frac{1}{2}\mathbb{E}[(\partial_x \log p_{\text{data}})^2]$ is a constant in θ and can be dropped, and the term $\frac{1}{2}\mathbb{E}[(\partial_x \log p_\theta)^2]$ is computable. The troublesome cross-term is where the trick lands:

$$-\mathbb{E}_{p_{\text{data}}}[\partial_x \log p_{\text{data}} \cdot \partial_x \log p_\theta] = -\int p_{\text{data}}(x)\, \frac{\partial_x p_{\text{data}}(x)}{p_{\text{data}}(x)}\, \partial_x \log p_\theta(x)\, dx = -\int \partial_x p_{\text{data}}(x) \cdot \partial_x \log p_\theta(x)\, dx.$$

The unknown density has been reduced to a plain factor, and one integration by parts (boundary terms vanishing for well-behaved densities) moves the derivative off it entirely:

$$= \int p_{\text{data}}(x)\, \partial_x^2 \log p_\theta(x)\, dx = \mathbb{E}_{p_{\text{data}}}[\partial_x^2 \log p_\theta(x)].$$

The unknown score has become a second derivative of the *model* — which for an EBM is a second

derivative of the energy — averaged over data samples we have. In d dimensions (Hyvärinen 2005):

$$D_F = \mathbb{E}_{p_{\text{data}}}\left[\sum_{i=1}^{d} \frac{1}{2}\left(\frac{\partial E_\theta(x)}{\partial x_i}\right)^2 - \frac{\partial^2 E_\theta(x)}{\partial x_i^2}\right] + \text{const.} \tag{4.11}$$

Signs deserve a moment of care here, because they are easy to slip on. Since $\log p_\theta = -E_\theta - \log Z$, second derivatives of the log-density are *negatives* of second derivatives of the energy, which is where the minus sign in Equation 4.11 comes from; the squared first-derivative term is immune because squaring erases the sign. A one-line sanity check: fit a Gaussian $E_\theta(x) = x^2/2\sigma^2$ to standard-normal data. Equation 4.11 gives $\frac{1}{2\sigma^4} - \frac{1}{\sigma^2}$, minimized at $\sigma^2 = 1$ as it must be; with the sign flipped, the objective would be minimized by sending $\sigma^2 \to \infty$ — a model dissolving into uniformity. (For *location* families the second-derivative term is constant in the parameter, so both signs give the same fit; that is exactly why the slip is easy to miss.)

No sampling, no partition function, and a consistent estimator. The catch is the Hessian trace: for deep networks and high-dimensional x, exact second derivatives cost roughly d backpropagations. Two descendants remove the bottleneck, one by adding noise and one by throwing dice.

Denoising score matching (DSM). Perturb each data point with Gaussian noise, $\tilde{x} = x + \sigma z$, and match the model's score to the score of the *noisy* data distribution. Vincent's identity (Vincent 2011) shows this equals (up to a constant) a beautifully simple objective:

$$\mathbb{E}_{x\sim p_{\text{data}},\, z\sim\mathcal{N}(0,I)}\left[\frac{1}{2}\left\|\nabla_{\tilde{x}} \log p_\theta(\tilde{x}) + \frac{z}{\sigma}\right\|^2\right], \qquad \tilde{x} = x + \sigma z. \tag{4.12}$$

Intuitively: given a noisy point, the score of the noised distribution points back toward the clean data that produced it, so *learning the score is learning to denoise*. Second derivatives are gone. The costs are a bias — the optimum matches the noisy distribution, not the data — and, if one shrinks σ to reduce that bias, a variance in the z/σ term that blows up as $\sigma \to 0$ (Song and Kingma 2021). Noise is a knob with no free setting.

Sliced score matching (SSM). Keep the clean data, and check the score match only along random directions: project both scores onto a random vector v and match the resulting scalars (Song et al. 2019). The Hessian-trace term of Equation 4.11 is replaced by the directional second derivative $v^\top(\nabla_x^2 E_\theta)\, v$ — the Skilling–Hutchinson estimator — *keeping the minus sign*. Both the squared directional derivative and that Hessian–vector product are computable with two backpropagations at any dimension. SSM is consistent, like exact score matching, at the price of extra estimator variance from the random projections.

The blind spot, and how it launched diffusion. Score matching has one structural weakness worth internalizing. Suppose the data has two well-separated modes — a mixture $\pi\, p_0 + (1-\pi)\, p_1$ with essentially disjoint supports. *Inside* each mode's region, the mixture's score is just that component's own score: the weight π enters the density as a constant factor, and gradients of log-densities kill constant factors. The relative weight of the modes lives precisely in the no-man's-land *between* them, where there is no data to enforce it. A score-matched model can therefore reproduce each island perfectly and still apportion probability between the islands

arbitrarily (Song and Ermon 2019; Song and Kingma 2021).

The fix is to flood the landscape: perturb the data with noise large enough to bridge the modes — making the weights visible in the score — and, since heavy noise distorts fine detail, do it at *many scales*, learning a noise-conditional score for each. Sampling then anneals from coarse to fine (Song and Ermon 2019). The reader of Chapter 3 will recognize the annealing schedule — start where entropy dominates, finish where energy does — and the reader of the wider literature will recognize the result: this multi-scale, score-based recipe, together with its diffusion-process formulation (Sohl-Dickstein et al. 2015; Ho et al. 2020; Song et al. 2021), became the generative-modeling workhorse of the 2020s. Diffusion models are the energy-based lineage's most successful child: what they inherit is the score Equation 4.8, the Langevin walk Equation 4.9, and the annealed temperature dial of Chapter 3.

4.6 Strategy three: learn by telling things apart

The third strategy is the slyest. If estimating a density is hard, compare it to one you know.

Noise contrastive estimation (NCE) (Gutmann and Hyvärinen 2010) introduces a noise distribution p_n with tractable density and easy sampling — a Gaussian, say — and mixes noise samples with data samples. Now play a game: given a point, was it data or noise? By Bayes' rule, the ideal referee computes

$$P(\text{data} \mid x) = \frac{\nu\, p_{\text{data}}(x)}{\nu\, p_{\text{data}}(x) + p_n(x)}, \tag{4.13}$$

with ν the data-to-noise mixing ratio. NCE builds a classifier of exactly this functional form, but with the model density in the data slot,

$$P_\theta(\text{data} \mid x) = \frac{\nu\, p_\theta(x)}{\nu\, p_\theta(x) + p_n(x)}, \tag{4.14}$$

and trains it by ordinary logistic regression on the mixed samples. If the classifier family is rich enough, training drives Equation 4.14 to the true posterior Equation 4.13 for every x, and matching posteriors at fixed p_n and ν forces $p_\theta = p_{\text{data}}$. Density estimation has been smuggled inside a classification problem — the one problem deep learning solves in its sleep.

Two features distinguish NCE in the EBM toolkit (Song and Kingma 2021). First, the normalizer Z_θ can be treated as an explicit learnable scalar — the classification game is only won when the model's *absolute* density matches the data's, so the game itself calibrates Z. Alternatively, fix $Z = 1$ and obtain a *self-normalized* model, a trick beloved of language modeling, where NCE-style objectives trained word embeddings and early neural language models at vocabulary scales where softmax normalization was unaffordable (Mnih and Teh 2012). Second, the choice of p_n is the whole game in practice: the classification task must be neither trivial nor impossible, which means the noise should resemble the data — a requirement that becomes brutal in high dimension, and has spawned a cottage industry of learned, adaptive noise distributions (Song and Kingma 2021).

One more connection ties the room together. Choose the noise to be the data itself, shifted by a small vector v, and the NCE objective Taylor-expands — as $\|v\| \to 0$ — into the sliced score matching objective with projection direction v (Song and Kingma 2021). Telling the data apart from a nudged copy of itself means, in the limit, matching directional slopes.

4.7 One family, three estimators

The three strategies look like three separate inventions. They are better seen as one family, distinguished by which divergence they minimize and what they demand of us.

Table 4.1: Three roads around the partition function. Each row answers: how does this method avoid computing $Z(\theta)$, and what does it pay instead?

	MCMC-based MLE (Section 4.4)	Score matching (Section 4.5)	NCE (Section 4.6)
Minimizes	KL divergence	Fisher divergence	KL of classification posteriors
Pull-up term handled by	sampling it	differentiating it away	canceling it against a reference
Requires	model samples (MCMC)	input derivatives of E_θ	a tractable noise density
Access to Z?	no	no (energy determined up to a constant)	yes (Z learned, or fixed to 1)
Characteristic failure	biased truncated chains	invisible mode weights	noise too far from data

And the roads interconnect. Contrastive divergence run with a single infinitesimal Langevin step *is* score matching, up to rescaling (Hyvärinen 2007). NCE against infinitesimally shifted data is sliced score matching, as we just saw. Underneath both coincidences sits a single identity — a relative form of de Bruijn's identity — relating the two divergences. Perturb data and model with the *same* Gaussian noise of variance t, write q_t and $p_{\theta,t}$ for the smoothed densities, and

$$\frac{d}{dt} D_{\mathrm{KL}}(q_t \,\|\, p_{\theta,t}) = -\tfrac{1}{2} D_F(q_t \,\|\, p_{\theta,t}) : \tag{4.15}$$

the Fisher divergence is the *rate* at which the KL divergence dissolves under noise (Song and Kingma 2021). (The classical de Bruijn identity is the special case that tracks entropy rather than KL.) KL is the integral view, Fisher the differential view, of one and the same discrepancy — which is why an MCMC shortcut and a derivative trick keep converging on the same estimator.

There are further roads — minimizing Stein discrepancies, adversarially training a sampler network against the energy, differences of KL divergences — for which Song and Kingma's survey (Song and Kingma 2021) and the running list at *Awesome-EBM* (Bian 2020) are the entry points. The taxonomy above is enough to read that literature: whenever a new EBM training method appears, ask first *how it implements the pull-up*.

4.8 Latent variables, and the free energy again

Often the energy needs helper variables that are observed never: the pose of a face, the segmentation of a sentence, the rotation of a molecule. Give the energy an extra argument $E_\theta(x, y, z)$ and let inference minimize over z as well. But minimizing over z is a $T = 0$ policy — it credits y with only its single best explanation. The finite-temperature policy sums over explanations:

$$F_\beta(x, y) = -\frac{1}{\beta} \log \int_z e^{-\beta E_\theta(x,y,z)}\, dz, \tag{4.16}$$

which the reader will recognize as a free energy — Equation 4.16 *is* $F = -\frac{1}{\beta} \log Z$ with the latent variable playing the role of microstate, and it interpolates between marginalization ($\beta = 1$: an answer is good if its explanations are *collectively* probable) and minimization ($\beta \to \infty$: an answer is as good as its best explanation) (LeCun et al. 2006). Latent-variable EBMs thereby fold the machinery of Chapter 3 inside the energy function itself, and the variational bound of that chapter — minimize $F(q) = \langle E \rangle_q - T\, S(q)$ over tractable q, with $T = 1/\beta$ — becomes the standard training route when the integral is intractable. The evidence lower bound of variational autoencoders is this construction at $T = 1$, as Chapter 3 already made explicit.

This is also where the energy-based view connects to the present frontier of self-supervised learning. LeCun's proposal for world-model architectures — JEPA, the joint-embedding predictive architecture — is at bottom a latent-variable EBM over *representation* space: energy measures prediction error between embeddings, latent variables absorb what cannot be predicted, and the central design problem is once again preventing collapse, either by contrastive pull-up or by regularizers that keep the embedding distribution spread out (LeCun 2022; Dawid and LeCun 2024). Decades of this subject can be compressed into one sentence: *dig valleys where the data is, and find an affordable way to keep the rest of the landscape high.*

4.9 The modern landscape

A brief tour of where energies appear in current practice, with pointers rather than treatments — the companion list (Bian 2020) tracks the full literature.

Your classifier was an EBM all along. A softmax classifier computes logits $f_\theta(x)[y]$ and normalizes over labels only. Reuse those logits as a joint energy, $E_\theta(x, y) = -f_\theta(x)[y]$, and the *marginal* energy of an input is $E_\theta(x) = -\log \sum_y e^{f_\theta(x)[y]}$ — the free energy of the label ensemble. Training this hidden generative model alongside the classifier improves calibration and robustness (Grathwohl et al. 2020). Even without retraining, the same marginal energy is a useful *score* for out-of-distribution detection, with higher energy read as more unfamiliar (Liu et al. 2020). (A classifier that was never trained as a generative model does not automatically place OOD inputs high; the energy score is still often a better OOD detector than the raw softmax confidence.) This is the "energy dial" of Chapter 3 read in reverse — the logits every deep classifier already emits are energies in thin disguise.

Generation and its guidance. Diffusion and score-based models (Section 4.5) carry the EBM inheritance in generation, and the additivity of energies (Section 4.2) becomes *composability at*

inference time: adding a classifier's gradient to a sampler's score steers generation toward a class (Dhariwal and Nichol 2021); adding hand-built constraint energies composes concepts (Du et al. 2020); in text, a lightweight energy can re-rank or steer a powerful autoregressive proposal — residual EBMs (Deng et al. 2020), constraint-guided decoding (Qin et al. 2022). Discrete spaces lack Langevin's gradient walk, and a lively literature builds gradient-informed proposals for them (Grathwohl et al. 2021). This inference-time story is the subject of Chapter 8.

Science. Molecules are the native habitat of energy functions — the word itself came from physics, and here it goes home. Denoising-style score matching over molecular geometries drives conformation generation (Shi et al. 2021); SE(3)-equivariant energies score protein structures end-to-end (Wu et al. 2021). The learned energy (or its score) sits in the same seat as the physical force field it approximates or replaces — Chapter 9 takes up this thread.

4.10 Where this sits in the book

This chapter closes Part I, and it is the load-bearing wall for Part II. Backward: Chapter 2 supplied entropy and the maximum-entropy principle; Chapter 3 derived the Gibbs distribution and identified $-\log Z$ as a free energy; this chapter made the energy learnable and catalogued the ways around the normalizer. Forward, each methods chapter picks up a specific thread:

- **Maximum-entropy learning** (Chapter 5) is the large tent in which Gibbs distributions are *used* as models: classical MaxEnt, probabilistic graphical models as an explanatory family (Section 5.1), maximum-entropy RL, energy-based guidance, and — as the worked example written so far — a Gibbs distribution over coalitions in a cooperative game, from which game-theoretic valuations are derived (Bian et al. 2022).
- **Minimum-entropy learning** (Chapter 6) trains a model on its own agreement signal; its collapse mode (Section 6.6) is the energy loss's flat-landscape failure in a new costume, and the free-energy decomposition used to analyze it is Equation 4.4 seen from the other side.
- **Minimax-entropy learning** (Chapter 7) asks which *features* an exponential-family model should include — which is precisely the architecture-selection problem for the EBMs of this chapter, posed variationally.
- **Energy-based reasoning** (Chapter 8) takes the landscape as given and spends inference-time compute navigating it: guidance is gradient descent on a composed energy, refinement is repeated local descent, and both inherit the annealing wisdom of Chapter 3.

4.11 Takeaway

An energy-based model is a learnable landscape: inference rolls downhill on it, probability — when needed — is bought from it at the price of a partition function. Training is landscape-shaping, and its non-negotiable rule is that digging valleys under the data must be paired with pulling up elsewhere, lest the landscape collapse into a useless plain. The three classical training strategies are three ways to afford the pull-up: sample it (MCMC-based maximum likelihood), sidestep it by matching slopes (score matching, the road that led to diffusion models), or fold

it into a classification game against known noise (NCE). One identity — a relative form of de Bruijn's — reveals them as views of a single discrepancy, and one design question organizes the whole field, from Boltzmann machines to JEPA: *how do you keep the rest of the world high?*

Part II

Part II — Methods

5 Maximum-Entropy Learning

i Scope and status of this chapter

This chapter is an early scaffold; it is worth saying what it will eventually hold. *Maximum-entropy learning* is where the energy-based viewpoint of Chapter 4 meets the maximum-entropy principle of Chapter 2: among all models consistent with what we know, prefer the maximally noncommittal one. When what we know is a set of expectations $\mathbb{E}_p[f_k] = \mu_k$, that distribution is the Gibbs form $p(x) \propto \exp(-E(x))$ **with the energy confined to the span of the constrained features**, $E = -\sum_k \lambda_k f_k$ — the content of the theorem is the confinement, not the exponential shape, since any strictly positive p can trivially be written as e^{-E} with $E := -\log p$. Maximum-entropy *learning* is what happens when that confinement is relaxed and the energy is learned instead of derived; the hinge between the two is the maximum-entropy / maximum-likelihood duality (Section 5.1). Read that broadly and it is a large tent. A complete treatment should span at least:

- **Classical maximum-entropy models** — the Jaynes programme of Chapter 2, exponential families, and Gibbs distributions as the least-committal fit to moment constraints.
- **Probabilistic graphical models** — Markov random fields and Bayesian networks. This is a classical field in its own right, still central after several decades (Pearl 1988; Koller and Friedman 2009; Wainwright and Jordan 2008). This book reads it as an *explanatory family*: a graph is a hypothesis about which independencies the joint must satisfy. The identification with classical maximum entropy is tight for undirected models and only partial for directed ones; that distinction is recorded in Section 5.1, not asserted as community consensus.
- **Maximum-entropy reinforcement learning** — first a distribution over trajectories $p(\tau) \propto e^{R(\tau)}$ that matches demonstrated feature counts and nothing else (Ziebart et al. 2008) (exactly so for deterministic dynamics; with stochastic transitions Ziebart et al. carry an explicit transition factor and an approximation assumption, and *maximum causal entropy* is the principled repair), then energy-based softmax policies $\pi(a \mid s) \propto e^{Q(s,a)/\tau}$ and soft actor-critic (Haarnoja et al. 2017, 2018).
- **Energy-based guidance and refinement** — the currently active, inference-time face of the same idea, where a (possibly composed) energy steers or polishes a generative process. This book gives it its own home in Chapter 8; it belongs to the maximum-entropy family because guidance is descent on an energy and refinement is annealed sampling from a Gibbs distribution (Chapter 3).
- **Energy-based cooperative games** — our own line of work, recasting valuation through a maximum-entropy lens. This is the one strand written out in full below,

as a worked example; it will be expanded in a later revision.

Section 5.1 records where graphical models sit relative to the maximum-entropy programme, and what is and is not licensed by that identification; it is not a treatment of the field itself, which is a book of its own. The other fully worked-out strand below is the last item.

5.1 Graphical models as an explanatory family

Probabilistic graphical models (PGMs) organize a joint $p(x)$ by a graph that *encodes* a set of conditional independencies (Pearl 1988; Koller and Friedman 2009). The encoding is not the same in the two standard realizations. On an undirected graph, a missing edge is pairwise independence given all other variables (and, when p is strictly positive, the pairwise, local, and global Markov properties coincide). On a directed acyclic graph the criterion is *d-separation*. A missing arrow still carries an independence — the local Markov property says each variable is independent of its non-descendants given its parents — but not the *same* independence as a missing undirected edge: conditioning on all the remaining variables can *open* a path through a collider, so the MRF's pairwise reading has no directed analogue (Pearl 1988). The field is older than the modern energy-based revival of Chapter 4, larger than this chapter, and still the working language for structured uncertainty in statistics and in much of classical machine learning — vision, speech, computational biology, and expert systems among them.

An **undirected** model, or Markov random field (MRF), scores *compatibilities*. On a finite graph G, a strictly positive distribution is Markov with respect to G if and only if it is a Gibbs distribution that factors over the cliques of G (Hammersley–Clifford; the published proof usually cited is Besag (1974)). In energy language that is already the definition of Chapter 4, with the extra discipline that E is a sum of local terms:

$$p(x) \;=\; \frac{1}{Z}\,\exp\Bigl(-\sum_{C\in\mathrm{cliques}(G)} E_C(x_C)\Bigr).$$

(The sum may equally be taken over maximal cliques: smaller clique terms can be absorbed.) The Gibbs $\Rightarrow$ Markov direction is elementary and needs no positivity; it is Markov $\Rightarrow$ Gibbs that does. Without it the implication fails: Moussouris' four-cycle example is a distribution that is Markov with respect to G but admits no Gibbs factorization over the cliques of G (Moussouris 1974).

A **directed** model, or Bayesian network, tells a *generative* story. If pa_i are the parents of coordinate i in a directed acyclic graph,

$$p(x) \;=\; \prod_i p(x_i \mid \mathrm{pa}_i) \tag{5.1}$$

(Pearl 1988). The arrows are often read as explanation or cause; the undirected edges of an MRF are not. Pearl's own contrast is that a Bayes net is a knowledge base of directed influences, while a Markov network is a system of symmetric associations. That reading is a stretch beyond what

the joint distribution alone licenses: from observational data a DAG is identifiable only up to its **Markov equivalence class** — every DAG with the same skeleton and the same v-structures — so "the graph is the explanation" should be heard as a claim about which independencies the graph asserts, not about which arrow points which way.

This book treats the pair as an **explanatory family**, in a local sense that is not a standard name in the literature. The graph is the explanation: it says which independencies the joint is required to satisfy, and the factorization is what makes the model intelligible rather than a mere scoring rule. That reading is closest to Pearl's directed case. It is more of a stretch for MRFs, whose native language is compatibility rather than cause. Both, however, explain the joint by a sparse independence hypothesis, and that is the sense in which they belong together here.

How much of this is classical maximum entropy? Enough to be worth saying, not enough to be a slogan. The maximum-entropy distribution of Section 2.2 that matches prescribed expectations of clique-supported features is an exponential-family MRF: the Lagrange multipliers are the feature weights, and the solution is the Gibbs form above (with $E_C = -\sum_k \lambda_k f_k$ over the features supported on C; the sign is the usual energy convention) (Jaynes 1957; Della Pietra et al. 1997; Wainwright and Jordan 2008). On a finite discrete state space every strictly positive MRF arises this way, because the clique-configuration indicators are a sufficient statistic. The modern synthesis of Wainwright and Jordan writes undirected graphical models, exponential families, and entropy duality as one subject. Feature induction for random fields (Della Pietra et al. 1997) and the maximum-entropy models of classical NLP (Berger et al. 1996) are the same *exponential-family* construction; they are MRFs only when the features have local graphical support — Della Pietra, Della Pietra, and Lafferty are explicit that their fields may be non-Markovian. Conditional random fields (Lafferty et al. 2001) take the construction over to $p(y \mid x)$ and became, for a decade, the default of structured prediction. None of that is controversial.

The duality itself has a real regularity condition, not just a name: the program $\max_p H(p)$ subject to $\mathbb{E}_p[f] = \mu$ is dual to maximum likelihood in the exponential family exactly when μ sits in the *interior* of the marginal polytope; on the boundary the MLE fails to exist, and the maximum-entropy optimum is attained only in the closure of the exponential family (Wainwright and Jordan 2008, sec. 3.4).

What is *not* settled — and should not be written as if it were — is the claim that "PGM *is* classical maximum entropy." Three qualifications matter.

1. The PGM literature organizes itself around *representation, inference, and learning*, not around Jaynes. Standard treatments (Pearl 1988; Koller and Friedman 2009) may discuss log-linear Markov networks, whose maximum-likelihood dual *is* a maximum-entropy problem, without taking that dual as the field's founding slogan. The maxent identification is a precise reading of the exponential-family parameterization, not the community's self-description.
2. The tight theorem is undirected, finite, and strictly positive. Directed models are a different organizing principle: Equation 5.1 is not a Jaynes problem. A Bayes net can be *moralized* into an MRF on the graph obtained by marrying co-parents and dropping directions; the resulting undirected model is an **I-map** (every independence it asserts by missing edges does hold in p), but it need not be a *perfect* map — moralization can lose

independencies that the DAG had (v-structures are the usual example, since marrying the parents removes exactly the edge whose absence encoded that independence). The rewriting $E(x) = -\sum_i \log p(x_i \mid \mathrm{pa}_i)$ is an energy in the sense of Chapter 4, and in this case $Z = 1$ already, so the network sits inside the energy-based tent as a *normalized* model. That does not make it a maximum-entropy model, and the precise reason is the dimensional cousin of the loss of independencies above: a Jaynes problem with *linear* constraints has a solution confined to a *linear* exponential family. Fully observed discrete **undirected** graphical models are exactly such linear exponential families; fully observed discrete **directed** (DAG) models are in general only a *curved* exponential family (Geiger et al. 2001), with equality precisely when the DAG has no immoralities (v-structures) — that is, when its Markov equivalence class contains a decomposable (chordal) undirected model. Away from finite discrete variables the identification loses its grip for a further, dimensional reason: the clique potentials now range over an infinite-dimensional function space, so the MRFs on a fixed graph no longer form a *finite-dimensional* exponential family. (A single Gibbs distribution is always trivially exponential-family — the claim has content only for the model class.) Gaussian MRFs remain a finite-dimensional exponential family; potentials given by mixtures do not.

3. Even on the undirected side, maxent fixes the *distribution given the constraints*. Choosing the graph, or which clique features to include, is a *feature-selection* question with more than one answer — Della Pietra, Della Pietra, and Lafferty's greedy KL-gain feature induction (Della Pietra et al. 1997), and Zhu, Wu, and Mumford's minimax entropy (Zhu et al. 1997) (Chapter 7) — and neither is part of Jaynes' original programme.

So the accurate statement, and the one this chapter will use, is narrower than a slogan and stronger than a metaphor: **a strictly positive exponential-family MRF on a finite graph is a classical maximum-entropy model with structured constraints**; Bayesian networks are an explanatory sibling that can be rewritten as (already normalized) energies but should not be renamed maxent; the PGM field as a whole is larger than either identification, and still doing work that this book will not reproduce. A future revision should include full derivations of both the Hammersley–Clifford theorem and the moralization construction. For now, this chapter simply points to them as essential references.

5.2 A worked example: energy-based cooperative games

The variational machinery of the previous section is not only for graphs. The same mean-field relaxation that Wainwright and Jordan use to approximate a graphical model's log-partition function applies verbatim to a Gibbs distribution over coalitions, and what it produces there turns out to be a familiar object.

As one concrete instance of the programme above, consider a **valuation problem** in machine learning: how much does a feature, data point, or player contribute to an outcome? The classical answer is the **Shapley value** from cooperative game theory, but it is one particular choice among many.

In *Energy-Based Learning for Cooperative Games* (Bian et al. 2022), the valuation problem is recast

through an energy / maximum-entropy lens. A cooperative game with characteristic function $v(S)$ over coalitions $S \subseteq N$ is turned into a distribution over coalitions by a single Jaynes constraint: match a prescribed expected payoff and nothing else. The solution is the Gibbs distribution

$$p(S) \propto \exp\left(v(S)/\tau\right),$$

with energy $-v(S)$. Here τ is a temperature in exactly the sense of Chapter 3: as $\tau \to 0$ the mass concentrates on the highest-value coalitions, while as $\tau \to \infty$ it flattens towards uniform. In the strict Jaynes reading $\tau = 1/\lambda$ is not a free knob — it is pinned by whichever expected payoff the constraint matches — so sweeping τ below is the same move as sweeping that matched level.

Player valuations are then read from a mean-field approximation to p. The payoff term of that mean-field ELBO is exactly Owen's **multilinear extension** of the game (Owen 1972, 1975). What the variational reading adds is the entropy term — which turns the multilinear extension from an interpolation device into an ELBO with a temperature — and the trajectory beyond one step: further mean-field iterations define a family of variational valuations whose fixed point — the minimizer of the mean-field KL divergence, i.e. the member with the best conceivable decoupling error — is the **Variational Index**. Shapley is therefore not a special case of the maximum-entropy *distribution*, but a one-step mean-field reading of it.

5.2.1 Why this is useful

- **Principled interpretation.** Valuations become variational parameters of a mean-field approximation to a maximum-entropy distribution over coalitions, rather than isolated axiomatic scores.
- **A family, not a point.** The temperature τ and the number of mean-field steps trace a spectrum of valuations, with Shapley and Banzhaf as one-step readings rather than the only options.
- **Bridges to physics.** The same Gibbs machinery is the one that carried the Ising energy into neural networks in the first place, by way of the Hopfield network (Hopfield 1982) and the Boltzmann machine (Ackley et al. 1985); see Chapter 3.

5.3 Takeaway

Maximizing entropy turns the choice of a distribution over coalitions into the choice of a constraint; mean-field inference on that distribution then turns "which valuation should I use?" into "how far should I iterate the decoupling?" — a shift that places Shapley and Banzhaf on one trajectory rather than as isolated axioms. It is the same PGM lesson from earlier in the chapter, one level down: a strictly positive MRF is maxent with structured constraints, and here the mean-field relaxation of that MRF is what turns a menu of valuation axioms into one trajectory.

6 Minimum-Entropy Learning

A model that answers the same question five different ways is telling us something. This chapter is about turning that signal — a model's disagreement with itself, measurable without any ground truth answer — into a training objective, and about what such an objective can and cannot deliver.

The true result is more limited than people often say, and it is important to say this clearly from the beginning. Minimising the entropy of a model's own answers is a strikingly effective way to *elicit* capability the model already has: on mathematics it matches verifier-based reinforcement learning while using no labels at all. It cannot add capability, and that is not a defect of present implementations but a consequence of a training loop that never touches ground truth (Section 6.7). Its characteristic failure is equally structural. Left unchecked, the objective drives the model to a single confident answer, right or wrong; Section 6.6 shows that this is not an implementation nuisance to be tuned away but a first-order transition in a mean-field free energy — an abrupt jump rather than a gradual drift — with a computable critical point and, whenever three or more distinct answers compete, hysteresis: a memory effect that makes the jump one-way. That is why the methods that survive their own objective all hold the entropy above a floor — by reset, by fairness term, or by filter — and why no choice of regularisation strength can be made to serve instead. In the implementation this chapter examines most closely, the floor currently sits at zero — a configuration its own ablations arrived at, and one whose published runs completed stably. Section 6.6 argues there is no contradiction in that, once the floor's role is named correctly: it is a stopping rule that decides how much of the model's diversity sharpening spends, not a stability device, and in the published runs a cruder stopping rule — the finite training budget — was doing its work.

Three cautions run through what follows. The name is unfortunate: minimum-entropy learning is a family of *objectives* and has nothing to do with the Rényi min-entropy H_∞ of Chapter 2. The physics is routinely oversold: Schrödinger's "order from order" turns out to be a genuine and clarifying reading of what these methods do, whereas Prigogine's minimum-entropy-production principle does not survive contact with the details, and Section 6.9 says why in more detail. And the history is longer than the modern literature usually recounts — the objective and the term that balances it appear side by side as early as 1991, and the pairing has since been arrived at independently in several communities (Section 6.3).

6.1 Two regimes, and a warning about the name

The maximum-entropy principle of Section 2.2 answers a question about *honesty*: given that we have measured a few things and nothing else, which distribution commits us to the least? The

minimum-entropy principle answers a question about *commitment*, and it applies in a regime the first one never contemplates. Suppose the distribution in front of us is not a summary of our ignorance but the output of a model that has already absorbed an enormous amount of information. Its dispersion is then no longer a virtue to be preserved; it is a symptom. A model that answers the same question five different ways is telling us something, and what it is telling us can be turned into a training signal.

The two principles are therefore not rivals. They act on different objects at different moments: maximum entropy shapes the distribution we *assume*, minimum entropy sharpens the distribution a trained model *produces*. A system that only maximises entropy generates candidates forever and never decides; one that only minimises it becomes rigid and confidently wrong. This chapter mainly discusses the second failure, as it is the main problem for the methods covered here.

Before going further, a terminological issue has to be cleared. Chapter 2 introduced the **min-entropy** $H_\infty(p) = -\log \max_x p(x)$, the $\alpha \to \infty$ member of the Rényi family, used in cryptography to bound an adversary's first-guess probability. That quantity has nothing to do with the subject of this chapter. Here, "minimum-entropy learning" names a class of *objectives* — procedures that reduce the entropy of a model's predictive distribution — not a particular entropy functional. The collision is unfortunate and it is now standard in both literatures, so it is best met head on. As it happens, the connection is not quite empty: Section 6.5 shows that the objective actually implemented by the method at the centre of this chapter is the Rényi entropy at $\alpha = 2$, a different member of the same family.

6.2 From word-level to meaning-level uncertainty

Turning "the model contradicts itself" into a number requires deciding what counts as a contradiction, and the obvious choice fails immediately. Ask a language model for the capital of France five times and it may return *Paris*, *It is Paris*, *The capital of France is Paris*, *Paris, France*, and *Paris — it has been since 987*. These are five distinct token sequences. If we calculate entropy at the token level, based on the distribution over next words, it might suggest high uncertainty—even when, in reality, the model is certain. For example, if you ask which paper first documented an obscure phenomenon, the model might return five distinct and seemingly plausible citations, each actually incompatible with the others. In such cases, the token-level entropy could be the same as, or even lower than, situations where the correct answer is always given. This happens because the model generates convincing but fabricated responses with confidence on a word-by-word basis.

The solution is to group together outputs with the same meaning before making any measurement. Let $\sim$ be an equivalence relation on responses that identifies those with the same meaning, and let $\mathcal{C}$ be the resulting set of clusters. Sampling G responses $o_1, \dots, o_G$ from the model for a prompt q and writing $\hat{p}_c = |c|/G$ for the empirical frequency of cluster c, the **semantic entropy** is

$$H_{\text{sem}}(q) = -\sum_{c \in \mathcal{C}} \hat{p}_c \log \hat{p}_c.$$

This is Shannon's formula applied to a coarser σ-algebra, which is the whole of the idea. In the Paris case all five responses fall into one cluster and $H_{\text{sem}} = 0$. The measurement now tracks what we care about, and the repetition exploit disappears, because repeating oneself does not create a new cluster.

Semantic entropy was introduced for uncertainty quantification (Kuhn et al. 2023) and reached wide attention as a hallucination detector (Farquhar et al. 2024): resample an answer ten times, cluster by meaning, and flag the high-entropy cases before a user sees them. Its appeal is that the signal is available with no answer key, no reference database, and no change to the model — which makes it deployable exactly where verification is hardest. But it is only a detector. It says when not to trust an answer; it does not produce a better one. The rest of this chapter concerns the obvious next question: if this uncertainty can be measured, can it be optimised?

6.3 Entropy minimisation as an objective: a short history

The idea of adding an entropy term to a loss and descending it is not new, and its history is worth tracing, because the same pair of ideas has surfaced independently in several communities, under different names and with different emphases — a pattern that is common wherever one piece of mathematics serves problems with different constraints. Everything the next sections struggle with has been met before.

Both terms are already present in 1991. Bridle et al. (1992) asked how to train a classifier on unlabelled data alone, and proposed maximising the mutual information between the input and the predicted class. Decomposed, that objective reads

$$I(c\,;x) \;=\; \underbrace{H(\bar{y})}_{\text{fair}} \;-\; \underbrace{\overline{H(y)}}_{\text{firm}},$$

where y is the predictive distribution for one input and $\bar{y}$ its average over the dataset. The second term is entropy minimisation exactly as it is practised today: make each individual prediction decisive. The first is the thing that stops the second from being satisfied by fraud, since a classifier that answers "class 3" to everything is perfectly decisive and carries no information at all. Bridle and his co-authors state the point in so many words, observing that it is trivial to achieve either desideratum alone, and their summary of what a working objective must be — *firm but fair* — remains the clearest available description of what is at stake.

The two terms then travelled separately. Grandvalet and Bengio (2004) brought entropy minimisation to semi-supervised classification: augment the likelihood on labelled data with $-\lambda \sum_{x \in \mathcal{U}} H(p_\theta(y \mid x))$ over the unlabelled set, pushing the classifier towards confident predictions on data it has never been told the answer for. The justification is the cluster assumption — that decision boundaries should fall in low-density regions — and under it, confident predictions on unlabelled points are evidence of a boundary in the right place. It is the firm term on its own, and the setting explains why that is a reasonable formulation there: the labelled likelihood anchors the classifier to all K classes, so the firm term can stand alone in a way it cannot in the fully unsupervised case, and a fairness term has no obvious work to do. This anchored formulation became the standard reference point for the modern literature, and

pseudo-labelling (Lee 2013) — whose author explicitly identifies it with entropy regularisation — operates in the same anchored regime. The fairness term returned to wide use in 2020, when Berthelot et al. (2020) introduced it under the name *distribution alignment* and traced the idea back to Bridle's mutual-information objective, observing that it had been proposed more than twenty-five years earlier and had, to their knowledge, not been in use in contemporary semi-supervised learning. Read as a whole, the record is less a story of anything being lost than of the two terms mattering in different regimes: as long as a supervised anchor was present, the firm term alone was enough; the fair term becomes essential exactly when the anchor is removed — which is the situation of this chapter.

What the field eventually converged on is not one fix but a three-term structure. Drawing on what has proven effective, any objective for unlabelled data must include all of the following components, with each omission leading to its own characteristic failure:

term	what it buys	what its absence costs
conditional entropy down — *firmness*	decisive predictions	predictions stay diffuse and carry no usable signal
marginal entropy up, or an equipartition constraint — *fairness*	use of the whole label space	**everything collapses onto one class**
a capacity, smoothness or invariance constraint	classes that mean something	the partition fragments, or becomes balanced and arbitrary

That two of the three do not suffice is not a conjecture. Gomes et al. (2010) demonstrate both failures side by side on data constructed to expose them: dropping fairness lets decision boundaries vanish and clusters disappear, dropping the capacity term shatters the data into a large number of categories. The equipartition methods make the third failure vivid — a constant encoder paired with a uniform code satisfies a balance constraint *exactly*, which is why Asano et al. (2020) and Caron et al. (2020) must couple it to an augmentation-invariance objective. And Caron et al. (2021) separate the two collapse attractors quantitatively: without centring the output entropy falls to zero, without sharpening it rises to $\log K$, and only the two together hold a representation in between.

One mechanism, six notations. The most striking feature of this literature is how many communities arrived at the fairness term independently — as a bonus in the loss, as a reweighting of the pseudo-label, as an edit to the logits, and as a constraint on the assignment matrix. These are objects of four different kinds, and it is worth saying precisely in what sense they are the same thing, because the ways in which they are *not* turn out to matter.

Fix the notation for the rest of this section. A network maps an input x to logits $z_c(x)$ over K classes, giving predictions $p_c(x) = \mathrm{softmax}(z(x))_c$, and over a batch or dataset of N inputs the **class marginal** is the average prediction

$$\bar{p}_c \;=\; \frac{1}{N}\sum_{n=1}^{N} p_c(x_n), \qquad \textstyle\sum_c \bar{p}_c = 1.$$

π_c denotes the **target class distribution** for class c—that is, the desired proportion of examples

assigned to each class. In most typical cases, we want all classes to be equally represented, so π_c is set to the uniform distribution: $\pi_c = 1/K$, where K is the total number of classes. In some scenarios, prior knowledge may justify setting a non-uniform target distribution.

Collapse is the statement that $\bar{p}$ concentrates on one class; fairness is any mechanism that resists it. The unifying observation is that all six act on the predictions in the same way:

> **i** Each replaces the logits $z_c(x)$ by $z_c(x) - \gamma_c$ for a **per-class offset** γ_c increasing in how over-represented class c currently is. What distinguishes them is how γ_c is estimated, and how far the resulting marginal is driven towards its target.

mechanism	offset γ_c	enforcement	acts on
marginal-entropy bonus $+\lambda H(\bar{p})$	$\propto p_c(x)(\log \bar{p}_c - \mathbb{E}_{p(x)} \log \bar{p})$	soft penalty, one gradient step	the loss
$-\lambda D_{\mathrm{KL}}(\bar{p} \,\|\, \mathrm{Unif})$	as above	soft penalty, one gradient step	the loss
distribution alignment	$\log \bar{p}_c - \log \pi_c$	one Sinkhorn step	the pseudo-label
logit debiasing	$\lambda \log \bar{p}_c$	one Sinkhorn step ($\lambda = 1$)	the logits
centring	$\overline{z_c}$, a moving average	one step, on a lagged estimate	the teacher's logits
Sinkhorn equipartition	the γ making $\bar{p}$ exactly uniform	hard constraint, at convergence	the assignments

Reading in order: the marginal-entropy bonus of Bridle et al. (1992), Hu et al. (2017) and Liang et al. (2020); the same quantity written as a divergence from uniform; the distribution alignment of Berthelot et al. (2020); logit debiasing (Menon et al. 2021; Wang et al. 2022); the centring operation of Caron et al. (2021); and the Sinkhorn equipartition constraint of Asano et al. (2020) and Caron et al. (2020).

Rows three, four and six are the same operation at different stopping times. With a uniform target $\pi_c = 1/K$, distribution alignment reweights the prediction to $\tilde{q}_c \propto p_c(x)/\bar{p}_c$, which in logits is $z_c(x) - \log \bar{p}_c$ up to a per-example constant — logit debiasing at $\lambda = 1$. And if the predictions are arranged into the matrix $Q_{cn} = p_c(x_n)/N$, whose columns already sum to $1/N$, its row sums are exactly $\bar{p}_c$, so the first row-normalisation of the Sinkhorn iteration towards $U(\frac{1}{K}\mathbf{1}, \frac{1}{N}\mathbf{1})$ rescales row c by $1/(K\bar{p}_c)$: the same offset once more. **Logit debiasing is one Sinkhorn step; Sinkhorn is logit debiasing run to convergence.**

This distinction is important, and it is a common source of confusion. One debiasing step does not make the marginal uniform. It makes the *row* sums uniform, but the per-example softmax that follows is a column renormalisation, and that undoes part of the correction — which is precisely why Sinkhorn alternates rather than stopping. The residual shrinks by a roughly constant factor of about three per pass: with $K = 10$ classes and $N = 256$ examples drawn from a moderately confident network, the total-variation distance from uniform runs $0.060 \to 0.020 \to 0.0064 \to 0.0021$, falling to about 10^{-7} after a dozen passes and to machine

precision after about thirty. A single step therefore buys most of the correction and none of the guarantee, which is the operative difference between a method that *discourages* collapse and one that *forbids* it. Row two, by contrast, is identical to row one: $D_{\mathrm{KL}}(\bar{p} \,\|\, \mathrm{Unif}) = \log K - H(\bar{p})$, so the two objectives differ by a constant and have the same gradient everywhere.

The remaining two relations are approximations, and both gaps have a definite sign. The gradient of $H(\bar{p})$ is *not* the flat offset $\lambda \log \bar{p}_c$: it carries a factor $p_c(x)$, so it acts on each example only through the classes that example is already predicted to belong to, and it is centred, so within an example it merely redistributes. The hand-designed offset applies the full correction to every example unconditionally. They agree on sign and on monotonicity in $\bar{p}_c$, and on little else — which is also why the two are calibrated differently: $\lambda = 1$ is a canonical, tuning-free setting for the logit correction, being exactly the Bayes adjustment that maps a model with implicit prior $\bar{p}$ onto a uniform one (Menon et al. 2021), whereas the entropy bonus has no distinguished coefficient and must be tuned.

Centring differs along another axis entirely. Subtracting the mean *logit* $\overline{z_c}$ is, because $\log p_c(x) = z_c(x) - \log Z(x)$ and the log-partition is common to all classes, the same as subtracting $\overline{\log p_c}$ — the logarithm of the **geometric** mean of the predictions, where debiasing subtracts the logarithm of the **arithmetic** mean:

$$\begin{aligned} \gamma_c^{\text{debias}} &= \log\left(\tfrac{1}{N}\textstyle\sum_n p_c(x_n)\right) \\ &\geq \tfrac{1}{N}\textstyle\sum_n \log p_c(x_n) \;=\; \gamma_c^{\text{centring}} \;-\; \tfrac{1}{N}\textstyle\sum_n \log Z(x_n), \end{aligned}$$

by Jensen, with equality only if p_c is constant across the dataset; the trailing term is the same for every class and so invisible to the softmax. The gap widens with the variance of p_c over examples, so centring discounts classes that dominate a few inputs strongly and penalises only classes predicted uniformly often — a weaker and smoother intervention, which is consistent with Caron et al. (2021) needing to pair it with sharpening rather than with an explicit capacity term.

i Gradient of the marginal-entropy bonus

Differentiating $H(\bar{p}) = -\sum_c \bar{p}_c \log \bar{p}_c$ through $\bar{p}_c = \frac{1}{N}\sum_n p_c(x_n)$ and the softmax Jacobian $\partial p_c / \partial z_j = p_c(\delta_{cj} - p_j)$ gives, for one example x,

$$\begin{aligned} \frac{\partial H(\bar{p})}{\partial z_j(x)} &= \frac{1}{N}\sum_c \left(-\log \bar{p}_c - 1\right) p_c(\delta_{cj} - p_j) \\ &= -\frac{p_j(x)}{N}\left(\log \bar{p}_j - \sum_c p_c(x) \log \bar{p}_c\right), \end{aligned}$$

the -1 terms cancelling because $\sum_c p_c = 1$. Ascending H therefore *subtracts* this quantity from the logits, which is the offset in the first row of the table. Two features are worth noting. The bracket has p-weighted mean zero, so the update is a pure redistribution within each example; and the prefactor $p_j(x)$ vanishes for classes the example is confidently not in, so an example already committed elsewhere contributes almost nothing to correcting

> an over-represented class. Neither property is shared by $\gamma_j = \lambda \log \bar{p}_j$.

Six communities, six notations, one mechanism — with the caveat that the two notations most often treated as interchangeable, the entropy bonus and the logit correction, are the two that are genuinely distinct.

A second and orthogonal question — what the penalty does *at* the boundary of the simplex — matters for what follows, and appears not to be remarked on anywhere. Tanaka et al. (2018) penalise $D_{\mathrm{KL}}(\pi \,\|\, \bar{p})$, a fixed prior π against the marginal prediction $\bar{p}$ — with the prior in the *first* slot, so the term *diverges* as any class's marginal mass approaches zero: an infinitely high barrier that no finite reward can buy through. The marginal entropy $H(\bar{p})$ is bounded on the simplex, so collapsing costs at most $\log K$, and a sufficiently strong firmness term will simply pay it. One is a hard floor and the other a soft one, and the distinction decides whether collapse is impossible or merely expensive.

Test-time adaptation is the closest analogue to the unsupervised setting. When a deployed network adapts to a shifted input distribution by minimising the entropy of its own predictions on the incoming stream (Wang et al. 2021), the labelled anchor is gone and collapse stops being hypothetical. Niu et al. (2023) devote a subsection to it, under a heading that says all that needs saying — online entropy minimisation tends to predict all samples to the same class. Their measurements are the strongest empirical statement available that this objective can be actively harmful when nothing holds it — accuracy on ImageNet-C at the highest corruption severity, averaged over all fifteen corruptions, with a batch-normalised ResNet-50:

stream	no adaptation	Tent	EATA
imbalanced label order, batch 64	18.0	2.1	0.9
balanced labels, batch size 1	18.0	0.1	0.1

Adaptation by entropy minimisation takes a network from 18% to a fraction of one per cent — far worse than leaving it alone. Both of these are deliberately adverse streams rather than the standard benchmark, which is the point: the objective is fine until the conditions that were implicitly propping it up are withdrawn. Note also that the weight regularisation of Niu et al. (2022) does not rescue it here, though it does help on backbones whose normalisation is not batch-dependent, which is itself evidence that anchoring is a weaker remedy than a structural one.

Two details of the fix Niu et al. (2023) propose are worth carrying forward. The first is that they monitor a moving average of the entropy loss and *reset the model* when it falls below a threshold, set to 0.2 — an entropy floor placed strictly inside the simplex, implemented as a discrete intervention rather than a filter. The same object returns in Section 6.4 with its role transformed: here it is a stability device rescuing a network from destruction; there, where the degenerate endpoint turns out to be quiet rather than catastrophic, a floor would serve as a stopping rule — and Section 6.6 argues that distinction is the key to reading its current setting of zero correctly. The second is that in their diagnostic experiment, on a single corruption with a group-normalised backbone, the model collapses at severity level five while at level three

the gradients stay in a stable range throughout: the behaviour changes sharply as a control parameter is turned. That is the closest thing in the empirical literature to the critical value derived in Section 6.6, though it should not be read as a general safe threshold — under mixed corruptions the same backbone collapses at level three as well. Running the other way, Lee et al. (2024) argue that entropy is not a trustworthy reliability signal in the first place, since under spurious correlation confidently wrong samples are exactly the harmful ones — a result that weakens the entire filtering family and, by elimination, strengthens the case for structural remedies.

The language-model literature has adopted two of the three families and not the third. Reinforcement learning from internal feedback has taken up filtering — the two-sided entropy band described in Section 6.4, the substitution of self-certainty for entropy in Zhao et al. (2025) — and anchoring, in the form of covariance-targeted clipping and KL penalties (Cui et al. 2025). A fairness term is absent: no work in this line maximises a dataset-level marginal entropy. Part of the reason is a genuine type mismatch. The answer space of a language model is prompt-dependent, so "cluster 1" for one question has nothing to do with "cluster 1" for another and the marginal $\bar{p}$ has no well-defined referent. Candidate substitutes exist — a prompt-independent output attribute such as length, format or refusal rate; a population-level coverage or pass@k target; or a floor on cluster mass within each prompt, which is what δ_{low} would approximate if it were set anywhere above zero — but none has been tried. That the gap is real is confirmed by what happens without them: in the plain entropy-minimisation baseline that Zhao et al. (2025) run against their own method, the model "converges to producing the same character regardless of the prompt" after a few updates. That is the vision literature's collapse, arriving in the language literature under a different name and from the same cause.

What is new in the method described next is precisely the removal of the supervised anchor. There is no labelled term at all. This is what makes the approach interesting, and it is also what returns the degeneracy in full force — a point developed in Section 6.6, where it turns out to be sharper than a mere risk of a bad optimum.

6.4 EMPO: reward agreement, not correctness

EMPO — entropy-minimised policy optimisation — applies the objective to the elicitation of reasoning in large language models, in a fully unsupervised setting (Zhang et al. 2025). The context is the reinforcement-learning recipe behind recent reasoning models (Shao et al. 2024; DeepSeek-AI 2025): sample several attempts at a question, score each against a ground-truth answer with an automatic verifier, and reinforce the winners,

$$r_i = \mathbb{1}\big[\,\text{verify}(o_i, a) = \text{true}\,\big].$$

The recipe works remarkably well, but the bottleneck sits entirely in the symbol a. Every training question needs a gold answer and a cheap, reliable checker. Mathematics has one — compare the final expression — and code has a test suite. A differential diagnosis does not; neither does a proposed mechanism, a study design, or a molecule nobody has synthesised. The algorithm is not the scarce resource. The verifier is.

EMPO removes a from the loop. Given an unlabelled question q, it samples G responses, partitions them by meaning as in Section 6.2, and pays each response the share of the group that agrees with it:

$$r_i = \hat{p}_{\,c(i)} = \frac{|c(i)|}{G},$$

where $c(i)$ is the cluster containing o_i. Responses in the majority cluster are reinforced and outliers are suppressed, with no reference to whether the majority is right. In the released implementation, the equivalence relation is a pipeline of answer extraction and symbolic equality for mathematics, and a small dedicated verifier model for open-ended reasoning — cheap in both cases, and cheap in a way that does not require knowing the answer, though for the learned verifier that last clause turns out to need a careful audit, taken up in Section 6.7.

Two guards accompany the reward. The first is a band on the entropy: training proceeds only on questions with $\delta_{\text{low}} < H_{\text{sem}}(q) < \delta_{\text{high}}$. Too much dispersion and the majority cluster is noise; too little and there is nothing left to sharpen but overconfidence. The second is a set of anti-gaming penalties — an empty answer is penalised outright, and an answer that merely echoes the question earns nothing — which prevent the model from manufacturing agreement without saying anything.

It is worth examining how the first guard is realized in practice, because the implementation ended up stricter than the design intended—and understanding why is revealing. The results above were obtained with the band active. When the experiments were later rebuilt on a different training framework, an ablation found the thresholds made limited difference, and they were relaxed to keep any group whose entropy is neither zero nor close to its ceiling: $0 < H_{\text{sem}} < 0.6 \log G$, over a group of G samples. **The two endpoints** (0 **and** $\log G$) **are redundant.** A group in which all G responses agree has $H_{\text{sem}} = 0$ and pays every response the same reward, 1; a group in which no two agree has $H_{\text{sem}} = \log G$ and again pays every response the same reward, $1/G$. Since the advantage subtracts the group mean, both cases contribute an identically zero update whether they are filtered or not. A filter acting at the endpoints removes only groups the optimiser was already ignoring — consistent with the ablation's verdict that the original thresholds could be relaxed at little cost.

What remains is the upper bound alone, and it is a real constraint: at $G = 16$ it admits a group only if its meanings are no more dispersed than roughly five equally-sized clusters. But it guards against one failure only — a majority that is really noise. Nothing in the current configuration bounds the entropy from *below* at any value strictly inside the simplex. That asymmetry is not a detail, but reading it correctly takes some care, and Section 6.6 supplies the reading: the floor turns out to be a *stopping rule* — the mechanism that decides where sharpening halts, per prompt — rather than a stability device, which is why its absence coexists peacefully with stable published runs governed by a short horizon, and why the ablation that retired it, run at exactly such a horizon, was not positioned to see what it does.

Empirically the method closes most of the gap to supervision without using any. On five mathematics benchmarks with a 7B base model, averaging over the suite:

Method	What it needs	Score
Base model	—	33.7
Supervised fine-tuning	$\{q, r, a\}$	39.0
Verifier-based RL (GRPO)	$\{q, a\}$	51.7
EMPO	$\{q\}$	51.6

Here q is the question, a the gold answer, and r a reasoning trace. Two things in this table deserve to be said plainly. The first is that supervised fine-tuning, which sees the most, does worst of the three — imitation of traces is a weaker signal than selection among the model's own attempts. The second is that the tie between EMPO and verifier-based RL holds on mathematics; on open-ended reasoning a real gap of several points remains. The honest summary is that removing labels costs little in a domain where the model is already strong, and more elsewhere.

6.5 What "reward agreement" actually optimises

It is worth computing what the reward above does in expectation, because the answer is both cleaner and different from the one usually stated.

Averaging r_i over the group, each response contributes the frequency of its own cluster, so a cluster of size $|c|$ contributes $|c|$ times $\hat{p}_c$:

$$\bar{r} \;=\; \frac{1}{G}\sum_{i=1}^{G} \hat{p}_{\,c(i)} \;=\; \sum_{c\in\mathcal{C}} \hat{p}_c^{\,2} \;=\; \exp\big(-H_2(\hat{p})\big),$$

where $H_2(p) = -\log \sum_c p_c^2$ is the **Rényi entropy of order two**, also called the collision entropy (Rényi 1961). The quantity being maximised is the *collision probability*: the chance that two independent samples from the model land on the same meaning.

This deserves a frank note, and then a correction to the way such paper-versus-code mismatches are usually described. The objective as written in the paper is the Shannon semantic entropy of Section 6.2; the objective the code implements is H_2. These are genuinely different functions: their gradients differ, and H_2 weights the dominant cluster more heavily than Shannon does. But they are not different *kinds* of object, and naming the family they belong to turns the discrepancy from an inconsistency into a choice that can be argued for.

A unified family, parametrized by order. Consider the Rényi entropy H_α from Section 2.1.4. Instead of rewarding responses based on their cluster's simple share, generalize by awarding a power of that share:

$$r_\alpha(c) \;=\; \frac{p_c^{\alpha-1}-1}{\alpha-1}, \qquad \mathbb{E}_{c\sim p}\big[r_\alpha\big] \;=\; \frac{\sum_c p_c^{\alpha}-1}{\alpha-1},$$

an expectation monotone in H_α for every $\alpha > 1$, so that maximising it minimises the entropy of that order. Two members are already in front of us. As $\alpha \to 1$ the reward tends to $\log p_c$ and its expectation to $-H_{\text{sem}}$, which is the paper's objective. At $\alpha = 2$ the reward is $p_c - 1$, which

differs from the implemented reward p_c by a constant — and constants are invisible here, since the advantage subtracts the group mean before anything is done with it. **The stated objective and the implemented one are the $\alpha \to 1$ and $\alpha = 2$ members of a single one-parameter family.**

A third member turns up in the filters. The surviving band is written in terms of H_{sem}, but the majority-share cutoffs that have at various points stood in for it are thresholds on $\max_c p_c$, which is e^{-H_∞}. Between the derivation, the reward and the gate, the algorithm therefore touches three orders at once: $\alpha \to 1$, $\alpha = 2$, $\alpha = \infty$. The min-entropy that Section 2.1.4 was at pains to separate from minimum-entropy learning turns out to appear here after all — not as the objective, but as the gatekeeper.

Why order two is the defensible choice. Three reasons, of which the first is decisive. A group holds only G samples — seven to sixteen in the released configurations — so p_c is never known, only estimated by $\hat{p}_c = |c|/G$. Shannon entropy admits *no unbiased estimator at any finite G*: the expectation of any estimator is a polynomial in p, and entropy is not a polynomial (Paninski 2003, Prop. 8). Collision probability is a polynomial, of degree two, and is therefore estimable without bias by $\sum_c |c|(|c|-1)/G(G-1)$. The gap is not academic: at $G = 8$ over five clusters the plug-in Shannon estimator is biased downward by 0.28 nats, a fifth of the quantity being measured, while the degree-two estimator is exact. Second, the reward stays bounded: a singleton cluster earns $1/G$, where the order-one reward $\log \hat{p}_c$ hands it $-\log G$, a spike that grows with the group size. Third —

$$\log K - H_\alpha \;=\; \frac{\alpha}{2K} \sum_c \varepsilon_c^2 \;+\; O(\varepsilon^3), \qquad p_c = \frac{1+\varepsilon_c}{K}, \quad \textstyle\sum_c \varepsilon_c = 0,$$

so near the uniform distribution every order is the *same* function of the distribution, differing only by the factor α. A Shannon derivation and a collision-entropy implementation give parallel gradients wherever the meanings are near-balanced, which — since the band admits only high-entropy questions — is where training begins. They part company as p concentrates, which is precisely the regime Section 6.6 is about.

They are not, for all that, the same function. Both are Schur-concave, so they agree on any pair ordered by majorization, and both are maximised by the uniform distribution and minimised by a point mass. But majorization is a partial order, and on incomparable pairs the two can disagree outright. A group placing half its mass on one meaning and scattering the rest over nine others, $p = (0.5,\ 0.056^{\times 9})$, has Shannon entropy 1.79 against 1.39 for a group split evenly four ways — while its collision entropy is 1.28 against 1.39. One says the first group is the more diverse; the other says it is the less. Which order is used is a modelling decision, not a formality.

A second and independent axis: from a partition to a kernel. Everything above holds the hard clustering fixed. Relaxing it is a separate generalisation, and the two compose. Replace the equivalence relation with a positive semi-definite similarity kernel over the sampled answers, normalise it to unit trace as $\rho = K/\operatorname{tr} K$, and read the entropy off the spectrum, $H_\alpha(\rho) = \frac{1}{1-\alpha} \log \operatorname{Tr} \rho^\alpha$, with $\alpha \to 1$ giving the von Neumann entropy of Chapter 2. The classical cases are recovered exactly rather than approximately: when K is block-diagonal with all-ones

blocks, the non-zero eigenvalues of ρ *are* the cluster masses p_c (Pasarkar and Dieng 2024), so $\alpha \to 1$ returns H_{sem} and $\alpha = 2$ returns $-\log \sum_c p_c^2$. The resulting two-parameter family — an order, and a choice of hard or soft — is the order-q Vendi score (Friedman and Dieng 2023; Pasarkar and Dieng 2024), which coincides with the matrix-based Rényi entropy introduced independently by Sánchez Giraldo et al. (2015), and which is, under the name Hill numbers, the standard family of diversity indices in ecology (Hill 1973; Leinster and Cobbold 2012).

This is worth spelling out because of one claim it would be easy to make and wrong. The kernel relaxation is *not* available only for collision probability. Nikitin et al. (2024) build exactly the $\alpha \to 1$ version for language models, under the name kernel language entropy, and prove that for any semantic clustering there exists a kernel whose von Neumann entropy equals the semantic entropy — by the same block-diagonal construction. What distinguishes $\alpha = 2$ is cost, not existence: $\operatorname{Tr} \rho^2 = \|\rho\|_F^2$ is a sum of squares needing no eigendecomposition, while every other order needs the spectrum. Over a group of sixteen that is free either way; over a corpus it is not.

The gap worth recording is the empty cell. The kernel literature has developed the $\alpha \to 1$ corner and the clustering literature the $\alpha = 2$ one, and a soft, order-two semantic objective — the natural home for a reward that is already a collision probability — appears not to have been tried.

There is also a pleasing coincidence in the expression $\sum_c p_c^2$. It is exactly the chance-agreement term p_e in Cohen's κ, the standard correction applied when two human raters are scored for agreement (Cohen 1960). In the inter-rater setting one computes the agreement two independent raters would reach by luck alone, and subtracts it. Here the same expression is computed over one model's repeated answers and *maximised*. The statistical object that measures how much of an agreement is worthless is, read in the other direction, the training signal.

6.6 The free-energy view, and why entropy collapses

Here is the conclusion of this section, before the argument for it. Rewarding agreement is a rich-get-richer process, and whether it runs away is decided by a single number crossing one. On one side of that threshold, an accidental imbalance between meanings dies out and the model stays honest about what it does not know; on the other side, the imbalance feeds on itself and the model commits to one meaning — and nothing in the objective checks whether that meaning is correct. Two properties of the crossing make it worse than an ordinary tuning problem. It is abrupt: the entropy barely moves for most of the approach, then falls off a cliff. And when more than two meanings compete, it has memory: the collapsed state is already stable on the *safe* side of the threshold, so a single large fluctuation can tip the system over, and moving the knob back afterwards does not undo it. Setting the knob near its critical value is therefore not a defence. A floor on the state is — though what exactly it defends, and what stood in for it in the published runs, takes the rest of the section to state precisely.

Everything below is the derivation, and the machinery is that of Chapter 3.

The control parameter is a temperature, and it is a familiar one. Anyone who has sampled from a language model has already met the object doing the work here. Sampling at temperature τ

replaces the model's distribution by $p(o)^{1/\tau}$ and renormalises: large τ flattens the distribution, small τ sharpens it, and $\tau \to 0$ is arg-max decoding. What follows is that same operation with two modifications. The exponent is a reward rather than the model's own log-probability, and — this is the entire source of the difficulty — the reward is itself the probability being adjusted. It is a softmax whose logits are its own outputs.

Start from the canonical form of the policy-optimisation problem, in which reward is traded against a Kullback–Leibler penalty that keeps the updated policy near a reference π_0:

$$\max_{\pi} \;\; \mathbb{E}_{\pi}[\, r \,] \;-\; \beta\, D_{\mathrm{KL}}(\pi \,\|\, \pi_0). \tag{6.1}$$

This is the minimisation of a free energy, with $-r$ in the role of energy and β in the role of temperature, and it can be solved in closed form. Add a Lagrange multiplier λ to enforce that the probabilities sum to one, and differentiate with respect to $\pi(o)$:

$$\begin{aligned} &\frac{\partial}{\partial \pi(o)} \left[\sum_{o'} \pi(o') r(o') - \beta \sum_{o'} \pi(o') \log \frac{\pi(o')}{\pi_0(o')} + \lambda \Big(1 - \sum_{o'} \pi(o') \Big) \right] \\ &\qquad = r(o) - \beta \log \frac{\pi(o)}{\pi_0(o)} - \beta - \lambda = 0. \end{aligned}$$

Solving for $\pi(o)$ and absorbing everything independent of o into a normaliser gives the **Gibbs tilt of the reference**,

$$\pi^{\star}(o) = \frac{\pi_0(o)\, e^{\, r(o)/\beta}}{Z}, \qquad Z = \sum_{o} \pi_0(o)\, e^{\, r(o)/\beta}, \tag{6.2}$$

which is the unique maximiser, since the KL term is strictly convex in π and the reward term is linear. Substituting back, the optimal value is $\beta \log Z$ — a log-partition function, exactly the free energy of Chapter 3. This is the same exponential family that Section 2.2 produces from the other direction: there, maximising entropy under a moment constraint; here, maximising reward under a divergence constraint.

The temperature reading is worth making concrete. Take three semantic clusters, a uniform reference $\pi_0 = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$, and rewards $r = (1.0,\, 0.5,\, 0.0)$. Then Equation 6.2 gives

β	$\pi^{\star}$	$\mathbb{E}[r]$
∞	$(0.333,\, 0.333,\, 0.333)$	0.500
2	$(0.419,\, 0.326,\, 0.254)$	0.582
1	$(0.506,\, 0.307,\, 0.186)$	0.660
0.25	$(0.867,\, 0.117,\, 0.016)$	0.925
$\to 0$	$(1,\, 0,\, 0)$	1.000

At high temperature the update does nothing and the policy stays at its reference; at zero temperature it becomes greedy and places all mass on the best cluster. The KL coefficient is precisely the knob that decides how far towards the arg-max a single update is allowed to

travel — the role played by τ in temperature sampling, now attached to a reward instead of a log-probability.

Nothing so far is specific to entropy minimisation. What *is* specific is that under EMPO the reward is not an external function of the response: by Section 6.4, r is the cluster frequency of the very policy being optimised. Writing p_c for the probability the policy assigns to cluster c and p_c^0 for the reference, substituting $r(o) = p_{c(o)}$ into Equation 6.2 turns an explicit formula into a **self-consistency condition**:

$$p_c \;=\; \frac{p_c^0\, e^{\,p_c/\beta}}{\sum_{c'} p_{c'}^0\, e^{\,p_{c'}/\beta}}. \tag{6.3}$$

This is a mean-field equation, of exactly the kind that describes a magnet whose spins feel the average field they themselves generate. The consequences are worth spelling out, because they turn a vague worry about instability into a quantitative statement. Take the KL anchor seriously for the moment and suppose π_0 is genuinely fixed; the paragraphs after next remove that supposition, which is what real implementations do.

A critical value. Take a reference that is uniform over K clusters, $p_c^0 = 1/K$, and nudge the balanced solution slightly: $p_c = 1/K + \varepsilon_c$ with $\sum_c \varepsilon_c = 0$. Substituting into Equation 6.3, the factor $e^{1/(K\beta)}$ is common to every cluster and cancels, leaving $p_c \propto e^{\varepsilon_c/\beta} \approx 1 + \varepsilon_c/\beta$ to first order, so that after normalisation

$$\varepsilon_c \;\longmapsto\; \frac{\varepsilon_c}{K\beta}.$$

An imbalance between meanings is thus multiplied by $1/(K\beta)$ at every step. This is the number promised at the opening, and the rest of the section is a matter of where it sits relative to one. The balanced state survives a perturbation if and only if $K\beta > 1$, so there is a critical value

$$\beta_c = \frac{1}{K},$$

below which the policy commits to a single cluster. The appearance of K is not an artefact: the more distinct meanings a model is entertaining, the smaller each one's share of the group, the weaker the positive feedback any one of them can exert, and so the less braking is needed to contain it. For $K = 2$ the self-consistency condition can be solved exactly — writing $m = p_1 - p_2$, Equation 6.3 becomes

$$m = \tanh\!\left(\frac{m}{2\beta}\right),$$

the Curie–Weiss equation — the textbook description of a magnet that spontaneously magnetises when cooled below a critical temperature — with β in the place of temperature and $\beta_c = 1/2$ in the place of that critical temperature (the Curie point). One warning for readers arriving from physics, where β conventionally denotes *inverse* temperature: here it is the KL coefficient and it plays the role of T itself. Large β means strong regularisation, a policy held near its reference and high entropy — the hot, disordered phase; small β is the cold, ordered, collapsed one.

i Derivation of the $K = 2$ case

With two clusters and a uniform reference, the factor $\frac{1}{2}$ cancels from numerator and denominator of Equation 6.3, leaving a softmax whose logits are the cluster masses themselves. Since $p_1 + p_2 = 1$ there is only one degree of freedom; write $p_{1,2} = \bar{p} \pm \delta$ with $\bar{p} = \frac{1}{2}$ and $\delta = m/2$. Then

$$m = \frac{e^{p_1/\beta} - e^{p_2/\beta}}{e^{p_1/\beta} + e^{p_2/\beta}} = \frac{e^{\bar{p}/\beta}\left(e^{\delta/\beta} - e^{-\delta/\beta}\right)}{e^{\bar{p}/\beta}\left(e^{\delta/\beta} + e^{-\delta/\beta}\right)} = \tanh\left(\frac{\delta}{\beta}\right) = \tanh\left(\frac{m}{2\beta}\right),$$

by the definition of the hyperbolic tangent. The cancellation of $e^{\bar{p}/\beta}$ is worth pausing on, because it is the same one that occurs at general K in the perturbation argument above: a softmax is blind to a common shift of its logits, so only *differences* between clusters drive the dynamics. It is also why subtracting the group baseline $\bar{r}$ leaves the update direction untouched.

Whether a non-zero solution exists is then settled entirely by the slope at the origin,

$$\frac{d}{dm}\tanh\left(\frac{m}{2\beta}\right)\Big|_{m=0} = \frac{1}{2\beta},$$

which is the amplification factor $1/(K\beta)$ evaluated at $K = 2$, as it must be — linear stability analysis is exactly this derivative. Because tanh is concave for $m > 0$ and bounded by one, a slope below one lets the curve fall away from the diagonal immediately and never return, so $m = 0$ is the only solution; a slope above one carries the curve above the diagonal, and boundedness forces it back across, creating a pair of solutions $\pm m^\star$. The two regimes are separated by $1/(2\beta) = 1$.

Expanding $\tanh x = x - \frac{1}{3}x^3 + O(x^5)$ and dividing through by $m \neq 0$ gives $1 = 1/(2\beta) - m^2/(24\beta^3)$, hence

$$m^\star = 2\beta\sqrt{3\,(1 - 2\beta)} \simeq \sqrt{3\left(1 - \frac{\beta}{\beta_c}\right)}$$

near the critical point. The imbalance m — the *order parameter*, in physics language — grows continuously out of zero with exponent $1/2$, the standard mean-field value. That continuity is what makes the $K = 2$ transition *second order*: the state changes smoothly, with no jump. It is precisely this smoothness that fails for $K \geq 3$.

The two regimes, step by step. The algebra is easier to trust once one has watched it happen. Both runs start from the same barely-perturbed state $p = (0.36,\ 0.32,\ 0.32)$ with $K = 3$; only β changes.

Above the threshold ($\beta = 1.5\,\beta_c$), the perturbation is erased and the entropy returns to its ceiling $\log 3 = 1.0986$:

step	p	H
0	0.360, 0.320, 0.320	1.0970
2	0.345, 0.327, 0.327	1.0983

step	p	H
4	$0.339,\ 0.331,\ 0.331$	1.0985
6	$0.336,\ 0.332,\ 0.332$	1.0986
8	$0.334,\ 0.333,\ 0.333$	1.0986

Below the threshold ($\beta = 0.75\,\beta_c$), the same perturbation compounds and the distribution is most of the way to a point mass:

step	p	H
0	$0.360,\ 0.320,\ 0.320$	1.0970
2	$0.384,\ 0.308,\ 0.308$	1.0931
4	$0.432,\ 0.284,\ 0.284$	1.0775
6	$0.539,\ 0.231,\ 0.231$	1.0097
8	$0.750,\ 0.125,\ 0.125$	0.7358

The column that deserves attention is the entropy in the second table. Over the first four steps it falls by less than two per cent while the gap between the leading cluster and the others nearly quadruples, and only then does it drop. Entropy is a *lagging* indicator of a process that is already well advanced, which is exactly why collapse is reported as sudden: from the outside it looks like training proceeding normally and then failing without warning. The warning was in the gap between clusters, not in the entropy.

For more than two clusters the transition is discontinuous. At $K = 2$ the dominant cluster's mass grows continuously from $1/2$ as β falls through β_c. For $K \geq 3$ it does not, and the cleanest evidence is a decisive experiment: hold β fixed, start the iteration from two different places, and see whether they agree. The balanced start carries a 10^{-3} perturbation, because starting from perfect symmetry would test nothing: a perfectly balanced state stays balanced by symmetry at every β. For $K = 3$, reporting the converged $\max_c p_c$:

β	from balanced start	from concentrated start	
$1.200\,\beta_c$	0.333	0.333	balanced only
$1.080\,\beta_c$	0.333	0.674	**both stable**
$1.035\,\beta_c$	0.333	0.768	**both stable**
$1.020\,\beta_c$	0.333	0.788	**both stable**
$1.000\,\beta_c$	0.811	0.811	collapsed only

Throughout the window $\beta \in [1.000,\ 1.093]\,\beta_c$ the two states are *both* stable, and which one the system occupies depends on where it came from. That path dependence is called *hysteresis* — the system remembers its history — and it is the defining signature of a *first-order* transition, one where the state jumps rather than drifts. The window is empty at $K = 2$, widens to $[1.000,\ 1.403]\,\beta_c$ at $K = 5$ and $[1.000,\ 1.882]\,\beta_c$ at $K = 8$ — the more meanings in play, the larger the region of ambiguity. This matches a classical model of statistical physics, the

mean-field Potts model, whose transition is likewise smooth at $q = 2$ states and discontinuous for $q \geq 3$ (Wu 1982) — so the behaviour is a structural property of self-referential dynamics of this type rather than a peculiarity of the present problem.

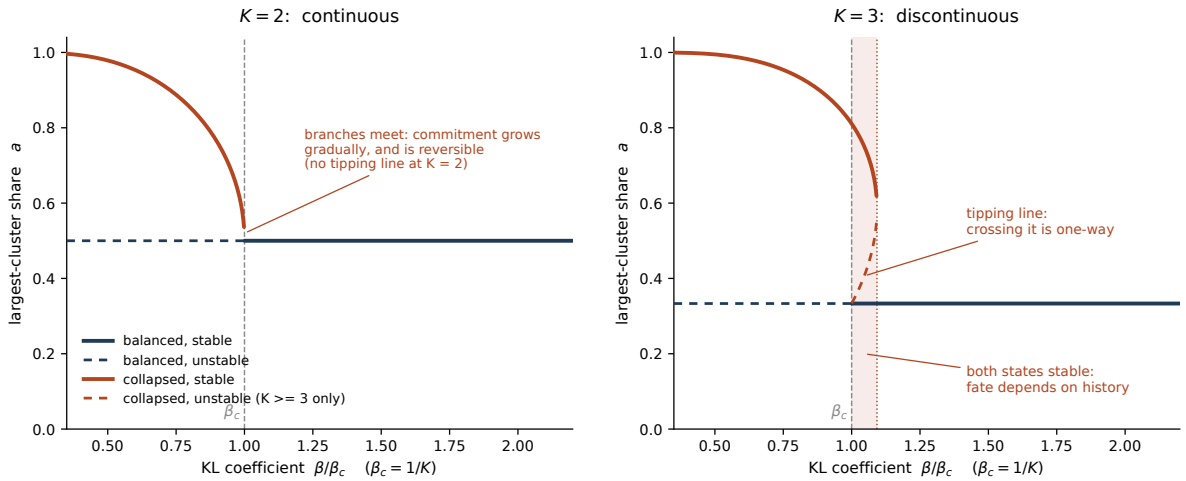

Figure 6.1: Where training can come to rest, and why the number of competing meanings changes everything. Each panel shows the resting states of the self-consistency condition Equation 6.3 as the anchor strength β varies: strong anchor on the right, weak on the left. Height is the share of the largest meaning cluster — $1/K$ means the model is keeping its options open (navy), 1 means it has committed to a single answer (rust). Solid curves are states the system can rest in; dashed curves are unstable and cannot be occupied. **Left** ($K = 2$)**:** weaken the anchor past β_c and commitment grows *gradually* out of the balanced state; strengthen it again and the system walks back. No red dashed line appears because none exists at $K = 2$ — there is no tipping line to cross, so the transition is continuous and reversible. **Right** ($K = 3$)**:** the committed state is already available and stable *before* the threshold is reached. In the shaded window both states are stable, separated by an unstable ridge (red dashed): which state the system occupies depends on where it came from, and a fluctuation across the ridge is one-way. This is the figure's practical message — with three or more meanings in play, setting β "just above the critical value" parks the system inside the window where a single large fluctuation commits it irreversibly. Generated by `code/min-entropy-phase-diagram.py`.

Figure 6.1 makes the practical consequence visible. The threshold $\beta_c = 1/K$ is not where the danger begins; it is only the last point of the safe regime — what physicists call a *spinodal*, the place where the balanced state finally stops surviving even small knocks. The collapsed state has already been available and stable for some distance above it. Tuning β to sit just above β_c is therefore not a safety margin — it places the system exactly in the region where both states coexist, and where a single fluctuation large enough to cross the dividing line is never recovered from. What is needed is not a carefully chosen distance from the critical point but a mechanism that acts on the state itself.

In practice there is no such reference, and collapse becomes unconditional. The stabilising mechanism above depends entirely on the KL term in Equation 6.1 anchoring the policy to a *fixed* π_0. It is worth checking whether real implementations have one, and the answer is that they do not. In the released EMPO code both available training stacks disable the penalty outright: the reference-model implementation sets its KL coefficient `beta: 0.0`, and the large-scale implementation switches the KL penalty off everywhere it can appear — both the penalty added to the reward and the auxiliary KL loss, along with both of their coefficients — in every one of its published run scripts. The entropy bonus is likewise set to zero. There is no term

anywhere in the optimiser that opposes entropy reduction.

What remains is only the implicit restraint of taking small steps. There is a standard way to read each gradient step in the language of this section (formally, as *mirror descent*): every update applies Equation 6.2 with the *current* policy in place of π_0 and the step size in place of β — instead of tilting a fixed reference, each step tilts wherever the policy already stands, so the cluster probabilities evolve as $p_c \propto p_c\, e^{p_c/\beta}$. The same perturbation analysis now gives

$$\varepsilon_c \;\longmapsto\; \varepsilon_c\Big(1 + \frac{1}{K\beta}\Big),$$

an amplification factor strictly greater than one for *every* $\beta > 0$. No step size stabilises the balanced state. A trust region — whether imposed by a small learning rate or by the unusually tight clipping range the released configuration uses — bounds how fast the policy moves, not where it ends up. Within this idealised dynamics, collapse is not a risk that a step size can tune away; it is the asymptotic behaviour, on a clock the trust region can stretch but not stop.

Two features of the group-relative advantage sharpen this rather than soften it. Advantages are computed as $A_i = (r_i - \bar{r})/s_r$ within the group of G samples. The baseline $\bar{r}$ cancels exactly in Equation 6.2, since a constant shift in the exponent is absorbed by the normaliser, so subtracting the mean has no effect on the direction of the update. The division by the group standard deviation s_r does have an effect: it acts as an adaptive inverse temperature — rescaling the pull without redirecting it — and because $s_r \to 0$ as the group becomes homogeneous, the effective temperature *shrinks* precisely as collapse proceeds. The pull towards the dominant cluster does not weaken near the degenerate state; it is held back only by clipping.

This is the sharpest justification available for the entropy floor of Section 6.4, and stating it correctly also dissolves what looks, at first sight, like a contradiction. The chapter has argued that a floor strictly inside the simplex is structurally necessary; the released configuration sets $\delta_{\text{low}} = 0$; and yet the published runs completed stably and an ablation found the thresholds made little difference. If the floor were a *stability device* — a thing that keeps the optimiser from blowing up — those three facts could not sit together. They sit together because that is not what the floor is, and seeing what it is instead requires separating three questions that the word *collapse* runs together.

Does training diverge without a floor? No — and not because anything opposes the dynamics, but because the endpoint silences itself. A prompt whose group has fully agreed pays every response the same reward, the advantage is identically zero, and the prompt stops contributing gradient. The fully-agreed state is what dynamicists call *absorbing* — once a prompt reaches it, it never leaves — but it is also quiet: nothing visibly breaks, and reaching it looks like convergence, not failure. The catastrophic collapses of Section 6.3 — a classifier sending every input to one class, a policy emitting one character for every prompt — are *cross-input* degeneracies, and EMPO's structure blocks the direct route to them: per-prompt clusters leave no shared label space for a marginal to collapse onto, and the anti-gaming penalties tax the cheapest constant policies.

Where does each prompt end without a floor? With all of its probability on a single meaning — at a corner of the simplex. That is what the mean-field analysis establishes: once the anchor is gone

the balanced state is unstable at every β, the transition is first order, and past the point of no return the commitment cannot be undone by a luckier sample later. Nothing in the optimiser selects a stopping point short of full commitment, and no setting of β can be made to select one — with an anchor it merely positions the system inside the window where both states coexist, and without one there is nothing to position.

Is full commitment what the method wants? Half of it is. Sharpening all the way to that corner is exactly the elicitation the method is for, on prompts where the majority is right. The other half is the cost, and it is not hypothetical, because the paper's own measurements price it: on prompts where the majority is wrong, the commitment is equally final — and the pass@k curves of Section 6.7, converging to the base model and crossing below it on hard benchmarks, are the coverage bill for sharpening run to the end.

On this reading the floor is a **stopping rule**, not a safety rail: it decides how much of the distribution's diversity sharpening is allowed to spend, per prompt, and it is the only instrument in the design that can make that decision at the level of the state. The ceiling cannot substitute — it rejects dispersion, and full commitment approaches from the other side. What operated in its place in the published runs was a second, cruder stopping rule: the finite training budget, a global and undirected version of what the floor would do per prompt. A fixed horizon stops the average prompt in time, the already-committed prompt too late, and the still-diffuse prompt too early; but under a horizon short enough, and clipping as tight as the released configuration's, most prompts never enter the region where a per-prompt floor would bind. The ablation's null verdict is then the expected signature rather than a refutation — it compared two configurations both governed, in effect, by the horizon — and the same reasoning says where the floor's value should become measurable: longer runs, sparser prompts, weaker base models, and any deployment in which locking in the current majority early is expensive. The degeneracy is the one Section 6.3 traced through three decades of this idea, now out in the open because the supervised anchor that used to contain it has been removed. Read against that history, even the band as designed is a *filtering* remedy — the weakest of the three families Section 6.3 describes, since it acts on which examples are used rather than on the objective, and it has no counterpart to the fairness term that the semi-supervised and self-supervised literatures found necessary. The analysis also predicts the failure mode reported in follow-up work on this family of methods, in which performance improves for a while and then falls below the untrained model: that is what running past a first-order transition looks like from the outside — a horizon that outlasted its usefulness as a stopping rule.

Three caveats on the scope of this analysis. It is an idealised treatment of the *reward dynamics* in the space of cluster probabilities, not of the neural network; it treats each prompt independently, ignoring the fact that all prompts share one set of weights, so an update for one prompt shifts the policy for all others. It treats p_c as known rather than estimated from G samples, which for the group sizes actually used adds sampling noise of the same order as the effects being described. And it reads the optimiser as taking the idealised update of Equation 6.2 exactly, whereas the clipped surrogate actually used departs from that whenever the clip binds. What survives these caveats is the qualitative structure — self-consistency, a critical temperature, a discontinuous transition, and the absence of any stabilising term — not the precise numerical threshold.

A note on what here is new, since the ingredients are not. Reading a learning objective as a free energy and locating its critical temperature is the method of deterministic annealing, where Rose et al. (1990) computed the temperature at which a cluster splits in two — the same construction, run in the opposite direction, since there cooling *creates* structure out of a single-cluster symmetric state whereas here it *destroys* the balanced one. The information bottleneck has its own sharp transition, below which the trivial representation becomes the global optimum (Wu et al. 2019; Wu and Fischer 2020), and entropy-regularised learning dynamics are known to switch behaviour abruptly at a critical exploration rate (Kianercy and Galstyan 2012). Closest of all, GX-Chen et al. (2025) analyse precisely the object of Equation 6.2 and conclude, as this section does, that collapse is a property of the objective rather than of the optimiser, and that β is the parameter controlling it. What that work characterises is the optimum; what is added here is the self-consistency that arises when the reward is the policy's own cluster frequency, and with it a critical value, an order of transition, and a stability analysis. The claim worth defending is not that this is a phase transition — any physicist would frame it so — but the chain that follows from the transition being first order: collapse has memory, the threshold at which the balanced state loses stability is consequently the wrong safety criterion, and no choice of β can substitute for a floor on the state.

6.7 The information ceiling

There is a limit on what any of this can achieve, and it is worth stating formally because it is easy to lose sight of amid the empirical results.

Let π_0 be the base policy, π_T the policy after training, and $a^\star$ the true answer to a question q. Every update EMPO performs is computed from samples drawn from the current policy and from the equivalence relation $\sim$; the ground truth enters nowhere. Two facts about that loop do all the work: the randomness of sampling — seeds, batch order; call it R — carries no information about the answer, and $\sim$ is frozen before training begins — whatever it is, it does not change in response to anything the run produces. Everything the run ends up with is therefore determined by four ingredients: the base model, the question, the metric, and the dice. The trained policy is a deterministic function $\pi_T = f(\pi_0, q, \sim, R)$, and the data-processing inequality — the rule that no amount of computation can create information about a quantity its inputs do not already carry — gives

$$I(\pi_T;\, a^\star \mid \pi_0,\, q,\, \sim\,) = 0.$$

All the content sits in the words after the bar. The statement is not that the trained policy knows nothing about the truth — it plainly does — but that *given what the base model already was and what the equivalence relation already knew*, the training run added nothing. The same argument, run once more (the callout below has the steps), yields the inequality of this section's title, and it has two terms:

$$I(\pi_T;\, a^\star \mid q) \;\le\; \underbrace{I(\pi_0;\, a^\star \mid q)}_{\text{pre-training}} \;+\; \underbrace{I(\sim;\, a^\star \mid \pi_0,\, q)}_{\text{the metric}}.$$

Information about the truth can be rearranged and expressed more reliably, but the budget is fixed before the first update — by pre-training, plus whatever the equivalence relation itself carries about the answer. On mathematics the second term is not merely small but identically zero: the clustering there is answer extraction followed by symbolic equality, a hand-written rule with no trained parameters and no access to the question, so the ceiling is pre-training alone, exactly as the one-term reading suggests. On open-ended reasoning the second term cannot simply be declared zero, because the equivalence relation is itself a trained model — and that turns out to deserve a section of its own, taken up below. Either way, whatever the trained policy gained, it gained by redistributing probability mass over a support that the base model already had.

i Derivation, and a world small enough to check it

The assumption doing the quiet work is that the training randomness R — sampling seeds, batch order — is independent of $a^\star$ given $(\pi_0, q, \sim)$, and that $\sim$ is frozen before the run. Both hold in the released code, where rewards and filters are computed from cluster frequencies alone. Then, in three steps. *Determinism*: every update uses only the G samples and their cluster frequencies, so $\pi_T = f(\pi_0, q, \sim, R)$ with f deterministic. *Markov chain*: given $(\pi_0, q, \sim)$, π_T is a function of R alone and R is conditionally independent of $a^\star$, so $P(\pi_T \mid \pi_0, q, \sim, a^\star) = P(\pi_T \mid \pi_0, q, \sim)$ — the chain $a^\star \to (\pi_0, q, \sim) \to \pi_T$. *Data processing*: $0 \le I(\pi_T; a^\star \mid \pi_0, q, \sim) \le I(R; a^\star \mid \pi_0, q, \sim) = 0$. The two-term ceiling then takes two short steps: since π_T is a function of $(\pi_0, \sim, R)$ once q is fixed, data processing gives $I(\pi_T; a^\star \mid q) \le I(\pi_0, \sim, R;\ a^\star \mid q)$; and expanding the right side with the chain rule, the R term vanishes and the other two are the budget and the metric term. Supervision breaks the chain at the first step: the verifier reward $\mathbb{1}[\text{verify}(o_i, a)]$ makes π_T a function of $a^\star$ as well, and each verification deposits bits about the answer into the weights.

The theorem is easiest to trust in a world small enough to enumerate. Let $a^\star$ be uniform on $\{0, 1\}$; let pre-training set a parameter θ_0 equal to $a^\star$ with probability 0.8; let the base policy answer θ_0 with probability 0.7; and let training sample $G = 5$ answers and sharpen towards their majority, so that θ_T is a function of θ_0 and the sampling noise alone. Exact enumeration of the joint distribution gives $I(\theta_T; a^\star \mid \theta_0) = 0$ — machine-exact, not approximately — while $I(\theta_T; a^\star) = 0.121$ bits against a pre-training budget of $I(\theta_0; a^\star) = 0.278$ bits, and replacing the majority vote by a verifier-guided update lifts the conditional information to 0.358 bits. Two details repay attention. The inherited information is *less* than the budget: majority voting is a noisy compression of θ_0, and processing loses information more readily than it preserves it. And one loophole sits outside the theorem entirely — selecting a checkpoint on a labelled validation set is itself an update that touches ground truth, worth at most $\log_2$ of the number of candidates in bits; small, but not zero. The numbers are reproduced by `code/min-entropy-information-ceiling.py`.

The metric is not free. The second term of the ceiling deserves to be followed into the implementation, because what sits there is more interesting than a technicality. For open-ended reasoning, the released code clusters answers with a small generative verifier (Ma et al. 2025)

— a mathematics-specialised 1.5B model fine-tuned on roughly 230,000 questions *with ground-truth answers*, whose equivalence verdicts were annotated by a much larger commercial model. The weights of $\sim$ are, in other words, a deterministic function of a labelled dataset. Three details of how it is called sharpen the point. The prompt contains the full question, not just the two answers being compared. One of the two policy outputs is placed in a slot literally named *Ground Truth Answer* and the other in a slot named *Student Answer* — the model's native supervised-verification format, with a sampled answer standing in for the gold one. And the prompt instructs the model not to solve the question itself, which is an instruction, not an architectural constraint: nothing prevents the forward pass from forming its own belief about the answer on the way to a verdict.

So can the metric smuggle supervision into a "fully unsupervised" run? The answer has a clean structure: leakage requires two conditions *jointly*, and either one alone transmits nothing. First, **knowledge** — the verifier must actually carry information about this question's answer, $I(\sim; a^\star \mid \pi_0, q) > 0$, whether from partially solving q, from having seen related questions in its training set, or from the annotation model whose judgments it distils. Second, a **behavioural channel** — its same-or-not-the-same verdicts must *vary* with that knowledge: merging a pair more readily when both members agree with its own belief, or fragmenting clusters of answers it considers wrong. A verifier that knows the answer but judges equivalence without ever consulting that knowledge leaks nothing: if the verdicts do not vary with what it knows, nothing of what it knows can reach the training run. The toy world above extends to check this exactly: add a verifier belief V that equals $a^\star$ with probability 0.9, and a merging rule that consolidates clusters agreeing with V while fragmenting the rest.

clustering metric	$I(\theta_T; a^\star \mid \theta_0)$	$I(\theta_T; a^\star)$ vs. one-term ceiling 0.278
uninformed rule (regex-like)	0	0.044 — holds
informed, correctness-blind verdicts	0	0.121 — holds
informed, verdicts follow the belief	0.114	0.340 — **exceeded**

In the third row the training run genuinely adds information about the truth — the quantity the one-term reading says is zero is 0.114 bits, and the trained model ends up knowing more than pre-training supplied. No theorem is violated: conditioning on the verifier still gives exactly zero in every row, and the two-term ceiling (0.636 bits in this world) holds throughout. The bits came from the verifier's training labels, carried into the run through pairwise equivalence verdicts. The numbers are reproduced by `code/min-entropy-verifier-leak.py`.

Two things follow. The defence offered for such metrics — they answer only "are these two the same", never "which is better", and provide no gradient — is right in spirit, because it narrows the behavioural channel; the middle row of the table is exactly that defence working. But it does not close the channel while the question sits in the prompt, and how far the verdicts of this particular verifier follow its beliefs has not yet been measured — a natural piece of follow-up work, and a cheap one. The measurements are easy to specify: cluster with the question replaced by an empty string and compare training curves; measure how often the

verdict flips when the two answers swap slots; on a labelled held-out set, compare the merge rate for pairs of equivalent *wrong* answers against pairs of equivalent correct ones. Until then, a careful accounting prices the two settings differently. The mathematics results — including the pass@k curves below — rest on a parameter-free symbolic rule and inherit the one-term ceiling exactly. The open-ended results are unsupervised *at the level of per-question labels*, with a metric distilled from labelled verification data; their ceiling is the two-term one, and the second term — plausibly small, since the verifier is a 1.5B model and the benchmarks in question sit well above its own solving ability — awaits measurement.

A last distinction keeps the accounting honest in the other direction. Even a metric carrying zero bits about $a^\star$ imports something real: a definition of *what counts as the same meaning*, learned from labelled entailment or equivalence data (Kuhn et al. 2023). That is supervision about how the space of answers is carved into meanings, not about which meaning is the true one — and it is load-bearing, since an earlier attempt with a weaker entailment model produced clusters poor enough that training collapsed into reward hacking. The theorem is silent about this kind of supervision, and should be: it is the same status as the tokeniser, fixed machinery that defines the problem rather than labels that answer it. The information ceiling concerns the other kind, and for the metric actually deployed on open-ended tasks, the other kind is not ruled out — only unmeasured.

The ceiling is not a technicality, and it has a testable consequence: a method that adds no information cannot enlarge the set of problems the model is able to solve given enough attempts. Measuring pass@k — the probability that at least one of k sampled attempts is correct — the trained and untrained models should converge as k grows, and on some benchmarks the untrained model should win, since sharpening a distribution costs coverage. That cost is not a side effect — the objective works at it directly: a correct answer rare enough to appear once among G samples earns the group's lowest reward and a negative advantage, so the objective actively pushes down exactly the answers that only pass@k can see. Both predictions are observed — on mathematics benchmarks, where the metric is the parameter-free rule and the ceiling argument applies in its exact form. At $k = 1$ the trained model is far ahead; by $k = 256$ the curves meet, and on harder benchmarks the base model overtakes. Minimum-entropy learning changes which answer comes out first. It does not change which answers are reachable.

6.8 Schrödinger, read carefully

The preceding result has an old precedent, and getting the precedent right requires more care than the usual invocation supplies.

In *What Is Life?* (Schrödinger 1944), Schrödinger asks how an organism avoids the decay to thermodynamic equilibrium that the second law seems to demand. His answer, in the sections titled "It feeds on 'negative entropy' " and "The statistical meaning of entropy," is that it does not evade the law but pays for its exemption:

> "Thus a living organism continually increases its entropy — or, as you may say, produces positive entropy — and thus tends to approach the dangerous state of

> maximum entropy, which is death. It can only keep aloof from it, i.e. alive, by continually drawing from its environment negative entropy."

and, a few pages later, that the organism "attract[s], as it were, a stream of negative entropy upon itself, to compensate the entropy increase it produces by living and thus to maintain itself on a **stationary and fairly low entropy level**."

Two details in that sentence are usually dropped and both matter here. First, the claim is a *balance*, not a minimisation: a level is held constant because what flows out offsets what is produced inside. Nothing is being pushed to its lowest possible value. Second, the level is "fairly low," not zero. Schrödinger is describing a floor, and Section 6.6 gives the machine-learning analogue of why a floor rather than a minimum is the right target.

The part of the book that bears most directly on Section 6.7 is a different argument, the one Schrödinger frames as **order from order** as against order from disorder. Ordinary physical law, he observes, is statistical: the regularity of a gas law is order extracted from the disorder of enormous numbers. Heredity cannot work that way, because the structure carrying it is far too small for statistics to smooth anything. It must instead be order copied from pre-existing order — his "aperiodic crystal," a structure whose information resides in not repeating.

The mapping onto the pass@k result is exact enough to be worth stating as a point-by-point correspondence rather than a loose analogy. Pre-training is the aperiodic crystal: the reservoir of structure, acquired at enormous cost from an external source. Entropy minimisation is the copying mechanism: it makes that structure come out reliably, and it makes it come out first. It creates nothing. An organism does not synthesise its own order either; it imports it, and Schrödinger's whole point is that the import is what has to be accounted for.

Where the analogy must stop is also clear. Schrödinger's entropy is the physical entropy of an open system exchanging real heat with a real environment, carrying units of k_B and constrained by the second law. The entropy of Section 6.2 is the Shannon entropy of a predictive distribution over meanings. Chapter 2 argued that these are the same functional applied to different objects, which licenses the shared vocabulary and the shared mathematics — but not the transfer of a thermodynamic theorem. Nothing in this chapter is derived from the second law, and nothing in it needs to be.

One further detail is worth recording, because it turns out to favour the treatment given here. In a note appended to the chapter in later editions, Schrödinger conceded to his critics that he would have framed the discussion in terms of **free energy** had he been writing for physicists alone, choosing entropy only because it connects more transparently to Boltzmann's order–disorder principle for a general reader. The concession is narrower than it is often reported to be — he does not retract the argument — but it points in a useful direction. Section 6.6 shows that the object doing the real work in minimum-entropy learning *is* a free energy, with the KL coefficient as its temperature. On this point Schrödinger's second thoughts and the mathematics agree.

6.9 Prigogine, and an analogy that does not hold

There is a second thermodynamic principle with "minimum entropy" in its name, formulated a year after Schrödinger's book by a scientist who worked on open systems and later won a Nobel Prize for the thermodynamics of self-organisation. The temptation to enlist it here is considerable. It should be resisted, and setting out why is more instructive than the endorsement would have been.

Prigogine's theorem (Prigogine 1945, 1947) concerns the **entropy production rate**. For an open system the second law is written as a balance, $dS/dt = d_eS/dt + d_iS/dt$, separating entropy exchanged across the boundary from entropy generated internally, with the internal term $\sigma \equiv d_iS/dt \geq 0$. Near equilibrium, each flow responds in proportion to the push that drives it — heat flow to a temperature difference, diffusion to a concentration difference: $J_i = \sum_j L_{ij}X_j$, and

$$\sigma = \sum_i J_iX_i = \sum_{i,j} L_{ij}X_iX_j.$$

The theorem states that if some pushes are held fixed by external constraints and the rest are free, minimising σ over the free ones is equivalent to the vanishing of the flows they drive — that is, to the steady state. In that regime σ only ever decreases as the system evolves (it is what dynamicists call a Lyapunov function), which is what makes the steady state stable.

Three things about this result rule it out as a foundation for anything in this chapter.

It is about a rate, not a level. The quantity minimised is d_iS/dt, with units of entropy per unit time, and it is not the system's entropy. Schrödinger's "fairly low entropy level" and Prigogine's minimum entropy production are different claims about different quantities, and they are logically independent: a system could hold Schrödinger's balance while producing entropy at any rate whatever (Gnaiger 2009). What Section 6.2 minimises is a level — the entropy of a distribution — with no rate anywhere in sight.

Its hypotheses have no counterpart here, and they are more restrictive than usually advertised. The theorem requires the flow–push relations to be linear, symmetric ($L_{ij} = L_{ji}$, Onsager reciprocity), with coefficients L_{ij} that are *constant* rather than state-dependent — and it requires the system's variables to look the same when the film is run backwards (Verschaffelt 1954; Maes and Netočný 2013). That last condition is the one most often omitted, and it is precisely the one Landauer (1975) exploited: a resistance in series with an inductance settles at the current $I = E/R$, which manifestly does not minimise $\sigma = RI^2/T$ — because a current reverses sign when the film runs backwards, and the theorem's hypotheses exclude such variables. Jaynes noted that the principle already fails for a network of resistors held at different temperatures (Jaynes 1980). Extending the statement from a point to a whole region, by minimising $\int_V \sigma\, dV$ to determine spatial profiles, produces field equations that contradict the conservation laws (Barbera 1999; Martyushev et al. 2007). Modern work has settled the status of the whole construction: the theorem is a small-driving approximation to an exact variational principle, accurate only when the system is barely pushed (Maes and Netočný 2007). It is a narrow theorem, not a general law.

It does not apply to living systems, and Prigogine said so. In his Nobel lecture he was explicit that the result holds only in "a 'strictly' linear theory in which the deviations from equilibrium are so small that the phenomenological coefficients may be treated as constants," and that "far from equilibrium the thermodynamic behavior could be quite different, in fact, even opposite to that indicated by the theorem of minimum entropy production" (Prigogine 1978). Elsewhere he noted that the linear relations, adequate for heat conduction and diffusion, "are generally not valid for the conditions of chemical kinetics" — which is to say, for biochemistry. The modern verdict is blunter. Martyushev and Seleznev (2013) point out that if minimum entropy production governed development, the goal of an organism's life would be the soonest possible death, since σ attains its minimum value of zero at equilibrium.

The history here contains its own warning. Prigogine did apply the principle to biology, in a 1946 paper with Wiame proposing that living systems evolve towards states of least entropy production and offering, in the authors' words, "a physicochemical interpretation of Lamarckism" (Prigogine and Wiame 1946). That paper is a substantial part of why the principle took hold among biologists, and its destination is a fair indication of what happens when a theorem with narrow hypotheses is carried into a domain that satisfies none of them.

Two further cautions belong here, since they generalise beyond this case. The first is that minimum and maximum entropy production are not the contradiction they appear to be: one minimises and the other maximises the same function, under different constraints and with respect to different variables, and which applies depends on the choice of thermodynamic variables and their behaviour under time reversal (Martyushev and Seleznev 2006). The second is a piece of history that ought to make anyone cautious about arguing from the word "entropy" in a title. The founding paper of the maximum-entropy-production literature in climate science was published as a theory of *minimum* entropy exchange; the two descriptions are the same physics, differing only in whether one counts the flux at the boundary as leaving or entering (Paltridge 1975).

The conclusion for this chapter is that the shared phrase is a coincidence. Minimum-entropy learning is not an instance of the minimum entropy production principle, does not inherit its guarantees, and would gain nothing if it did, since those guarantees hold only in a regime with no analogue here. There is one genuine structural inversion worth noticing on the way out. In Prigogine's setting, low entropy production is where the dynamics *ends up on its own*; the principle is descriptive. In the setting of Section 6.6, low entropy is what we *pay* for, and having paid, we must then pay again to stop the system going lower. The two are not the same shape of statement at all.

6.10 So why call it minimum-entropy learning?

Given the preceding two sections, the name deserves a defence. It is defensible on two of three possible grounds, and it is worth being explicit about which.

As a literal description, it holds. The method minimises a named entropy functional over a well-defined space — the Shannon entropy of a distribution over semantic clusters as stated, the Rényi entropy at $\alpha = 2$ as implemented. Nothing metaphorical is happening; a specific

entropy of a specific distribution goes down.

As a structural claim, it holds, and this is the more interesting case. The objects that appear when the method is analysed are not borrowed imagery. The KL-regularised optimum is a Gibbs distribution; the objective is a free energy; the coefficient β enters exactly where a temperature enters; the self-consistency condition is a mean-field equation with a genuine critical point and a genuine first-order transition. These are the same mathematics as statistical physics, not a picture of it — which is the same standard that justifies calling a Boltzmann machine a Boltzmann machine.

As a claim to physical law, it does not hold and is not needed. No thermodynamic theorem is being invoked, no second law is doing any work, and no result transfers from Prigogine's setting or from Schrödinger's. The biological reading of Section 6.8 is an analogy about the *provenance of order* — a claim about where structure comes from, which the pass@k result independently establishes on its own terms — and not a derivation.

Keeping these three levels apart is the discipline this whole area most needs. The entropy vocabulary is powerful enough to describe learning objectives precisely, and seductive enough to make unearned claims sound rigorous. The first two grounds are earned. The third would not be.

6.11 When agreement means truth, and when it does not

The method rests on treating consensus as a proxy for correctness, and it is worth being precise about when that substitution is legitimate.

It works when the knowledge is already in the model but unstably expressed. The model can reach the answer; it simply does not do so reliably, and sharpening the distribution converts an occasional success into a dependable one. This is the regime Section 6.7 describes: the answer is already among the things the model can say, and training moves probability onto it. It also requires the question to have a stable meaning, since grouping by meaning is the only thing preventing "be consistent" from degenerating into "repeat yourself."

It fails in two ways. The model may never have seen the material, in which case there is nothing to sharpen — no amount of self-agreement conjures knowledge the model never had. More dangerously, the model may be confidently and consistently wrong, in which case low entropy certifies a falsehood and training reinforces it. The paper's own appendix contains a clean instance: on a probability problem the model silently substitutes a different goal, then twice announces that something looks wrong and reworks the solution — each rework landing on the arithmetic and never on the substituted goal. It returns 325 with high confidence; the answer is 265. Nothing in the reward can see the substitution: a model that re-derives the same misreading on resampling lands in the same cluster, the semantic entropy is low, and agreement is paid for being wrong. Fluent self-correction is not verification.

Three design rules follow, and they summarise most of what this chapter has established.

Threshold, do not minimise. Stop reducing entropy at a floor, leaving the model room to change its mind. Section 6.6 shows this is not a matter of taste: without a floor the fixed point is degenerate

and the approach to it is discontinuous. Schrödinger's "fairly low," not zero.

Group meanings, not strings. Otherwise consistency collapses into repetition, which is free to produce and worth nothing.

Never evaluate on the training signal. Consistency drives the optimisation; held-out ground truth is what gets reported. The moment those become the same number, the method is measuring its own success by the criterion it was trained to satisfy, and the failure case above becomes invisible.

6.12 Takeaway

Minimum-entropy learning turns a model's disagreement with itself into a training signal, which buys reliable elicitation of existing capability without any labels at all — and buys nothing beyond that, by an information-theoretic argument that the pass@k curves on mathematics confirm in its exact form. Its central pathology, degeneration to a single confident answer, is not an implementation nuisance but a phase transition in a mean-field free energy, which is why the durable remedies are floors on the entropy of the state — stopping rules for how much diversity sharpening may spend — rather than settings of a knob.

Set against Chapter 5, the pair is now complete: maximum entropy supplies possibility and minimum entropy supplies commitment, and neither is a general recipe for intelligence on its own. The next chapter shows that the two are not merely complementary but composable — that choosing *which features to model* turns out to require maximising entropy on the inside and minimising it on the outside, simultaneously.

7 Minimax-Entropy Learning

7.1 Which features should a model include?

Maximum entropy (Chapter 5) answers a precise question: given a set of measured features, which model is least biased? It is silent on a prior question — *which features should we measure at all?*

Every model is a lossy compression that trades complexity against uncertainty. Include too few features and the model stays vague; include too many and it ceases to be a compression. For a fixed budget of features, we want the description that leaves the least uncertainty.

7.2 The minimax entropy principle

Starting from the **minimum description length** (MDL) principle (Grünwald 2007), two results follow in sequence. First, among all models consistent with a feature set F, the shortest description is the maximum-entropy model (Feder 1986). Second, for such models the description length equals the model's own entropy, $L(P_F) = S(P_F)$. Choosing features to minimize description length therefore means choosing them to minimize that entropy.

The optimal features are the ones producing the maximum-entropy model with *minimum* entropy (Carcamo et al. 2025):

$$F^* = \arg\min_F \; S(P_F), \qquad P_F = \arg\max_P \; \{\, S(P) \;:\; \mathbb{E}_P[f_\mu] = \langle f_\mu \rangle_{\text{exp}},\; f_\mu \in F \,\}.$$

The nesting gives the principle its name: maximize entropy on the inside, minimize it on the outside. The best features are those that maximally constrain our expectations while leaving the model maximally random about everything unobserved. The principle was first proposed for texture modeling in computer vision (Zhu et al. 1997).

7.3 Three equivalent views

Minimizing $S(P_F)$ optimizes three quantities at once. In what follows, P_{true} denotes the distribution that actually generated the data, and P_{ind} the featureless baseline model in which the variables are treated as independent.

- **Shortest description.** Since $L(P_F) = S(P_F)$, the optimum is the most compressed account of the data.

- **Closest to the truth.** The entropy gap to the true distribution is exactly a divergence, $S(P_F) - S(P_{\text{true}}) = D_{\text{KL}}(P_{\text{true}} \| P_F)$, so the optimum is also the most accurate model.
- **Most informative features.** The information carried by the selected features, $I_F = S(P_{\text{ind}}) - S(P_F)$, is maximized at the same point.

Compression, accuracy, and informativeness coincide — a reassuring sign that the criterion is the right one.

7.4 Solving it

The difficulty is combinatorial. Each candidate feature set requires solving a maximum-entropy problem for its parameters, and the number of candidate sets grows super-exponentially, so brute-force search is infeasible beyond small systems (Carcamo et al. 2025). The practical workhorse is a **greedy** algorithm that repeatedly adds whichever remaining feature reduces the entropy most.

Structure sometimes restores exactness. For trees of pairwise correlations the entropy reduction decomposes into a sum of mutual informations, the problem becomes a maximum-spanning-tree problem, and greedy is optimal. More generally, if entropy reduction is **submodular** — diminishing returns as features accumulate — greedy comes within $1 - 1/e$ of the optimum (Nemhauser et al. 1978). Whether the minimax entropy objective is submodular in general remains open; where it is not, guarantees can still be recovered in terms of how far the objective departs from submodularity (Bian et al. 2017).

7.5 Takeaway

Minimax entropy couples the two preceding principles into a single criterion: **maximize entropy to model the world, minimize entropy to commit to what matters**. Maximum entropy supplies possibility, minimum entropy supplies certainty, and the minimax formulation decides which details deserve to be modelled at all.

The principle has recently yielded exactly solvable models of large neuronal populations (Lynn et al. 2025), yet its reach in machine learning is largely untapped. Open directions include scalability and sample complexity in high dimensions, realizations for LLMs, agents, and multimodal models, and the relationship to the free-energy principle (Friston 2010).

8 Energy-Based Reasoning, Guidance, and Refinement

Energy-based thinking is not only a training paradigm; it is also a powerful *inference-time* tool. Given an energy function $E_\theta(x)$ that scores how "good" a candidate solution x is — whether trained by the methods of Chapter 4 or assembled by composing pretrained models and constraints, exploiting the additivity of energies noted there — we can **guide**, **refine**, and **search** at test time.

This chapter is the inference-time face of the maximum-entropy programme sketched in Chapter 5: guidance is descent on an energy, and refinement is annealed sampling from the Gibbs distribution $p(x) \propto e^{-E(x)}$, so both are the maximum-entropy machinery of Chapter 3 applied at test time rather than during training.

8.1 Guidance

An energy (or a gradient of it) can steer a generative process toward desiderata:

$$x \leftarrow x - \eta \nabla_x E_\theta(x),$$

nudging samples toward lower-energy, higher-quality regions. This is the mechanism behind classifier / energy guidance in diffusion-style generation and, more broadly, behind constraint-aware generation.

Note that pure descent is the zero-temperature limit of Chapter 3: it consults energy and ignores entropy entirely, and so inherits the classic weakness of getting trapped in local minima. Adding noise and lowering it over the course of the process — annealing (Kirkpatrick et al. 1983) — is the finite-temperature generalization, and it is the same schedule that diffusion samplers implement under a different name.

8.2 Refinement

Rather than accepting a single forward pass, we can iteratively **refine** a candidate by repeatedly lowering its energy — an optimization view of "thinking longer." Refinement trades extra compute for higher-quality outputs, and connects naturally to the minimum-entropy view of self-consistent reasoning in Chapter 6.

8.3 Reasoning as energy minimization

A unifying perspective:

- A **reasoning problem** defines an (implicit) energy landscape over candidate solution trajectories.
- **Reasoning** is the process of finding low-energy trajectories — via search, guidance, or refinement.
- **Learning** shapes the landscape so that correct or self-consistent solutions sit at the minima.

This viewpoint ties together the threads of the book: maximum entropy defines principled landscapes, minimum entropy sharpens them from self-supervision, and energy-based inference navigates them. The next part turns to where these ideas pay off — scientific intelligence.

Part III

Part III — Applications

9 Entropy-Based Learning for Science

The ultimate aim of this research program is **scientific intelligence** — AI that is not only accurate and robust, but increasingly endowed with reasoning capabilities that enable new modes of scientific inquiry. Entropy- and energy-based learning offer principled tools for this goal.

9.1 Two synergistic directions

- **Advancing AI with Science.** Fundamental scientific and mathematical theories — statistical physics, information theory, the maximum-entropy principle — inspire principled AI methodologies. Energy-based games (Chapter 5), entropy-minimized reasoning (Chapter 6), and minimax-entropy model selection (Chapter 7) are three examples.
- **Advancing Science with AI.** The same methods accelerate discovery in the sciences, from molecular representation learning to chemical reasoning.

9.2 Molecular foundation models

A concrete thread runs from **self-supervised graph transformers** for molecular representation learning toward **molecule–language** models with reasoning capabilities. The trajectory mirrors the broader shift from *pretraining–finetuning* to *LLM-style reasoning*, and entropy-based objectives provide label-efficient signals where annotated data are scarce.

9.3 Outlook

A deeper unification may be within reach. The minimax view (Chapter 7) suggests that modelling the world and committing to a decision are two halves of one objective — an idea that echoes the **free-energy principle** proposed as a unified account of perception and action in the brain (Friston 2010). The quantity that principle asks the brain to minimize is the variational free energy of Chapter 3, which is what makes the correspondence more than a metaphor. Making it precise, and scaling it to LLMs, agents, and multimodal scientific models, is an open and appealing direction.

As the research matures, this book will grow to include worked examples, executable code, and reproducible experiments.

See the Blue Whale Lab and https://yataobian.com/ for the latest developments.

10 Summary

This book introduced **entropy-based learning** through three complementary principles and their shared energy-based substrate:

- **Maximum entropy** yields the least-biased model consistent with given constraints — grounding exponential families, Boltzmann machines, and a principled family of game-theoretic valuations (Bian et al. 2022).
- **Minimum entropy** sharpens a capable model's own predictions, enabling *unsupervised* elicitation of reasoning (EMPO) in a latent semantic space (Zhang et al. 2025).
- **Minimax entropy** selects the features worth modelling at all: those whose maximum-entropy model carries the least entropy, simultaneously the shortest, most accurate, and most informative description (Carcamo et al. 2025).

Binding the three together is the **free energy** $F = U - TS$: temperature sets the exchange rate between energy and entropy, so the principles are three settings of one dial rather than three separate doctrines. It is also the equation that carried statistical physics into machine learning, from the Hopfield network and the Boltzmann machine through to the evidence lower bound — a lineage recognized by the 2024 Nobel Prize in Physics (The Royal Swedish Academy of Sciences 2024).

This is an early scaffold; future revisions will add worked examples, code, and experiments. For updates, see https://yataobian.com/ and the Blue Whale Lab.

References

Ackley, David H., Geoffrey E. Hinton, and Terrence J. Sejnowski. 1985. "A Learning Algorithm for Boltzmann Machines." *Cognitive Science* 9 (1): 147–69. https://doi.org/10.1207/s15516709cog0901_7.

Asano, Yuki M., Christian Rupprecht, and Andrea Vedaldi. 2020. "Self-Labelling via Simultaneous Clustering and Representation Learning." *International Conference on Learning Representations* (*ICLR*). https://arxiv.org/abs/1911.05371.

Barbera, Elvira. 1999. "On the Principle of Minimal Entropy Production for Navier–Stokes–Fourier Fluids." *Continuum Mechanics and Thermodynamics* 11 (5): 327–30. https://doi.org/10.1007/s001610050127.

Berger, Adam L., Stephen A. Della Pietra, and Vincent J. Della Pietra. 1996. "A Maximum Entropy Approach to Natural Language Processing." *Computational Linguistics* 22 (1): 39–71.

Berthelot, David, Nicholas Carlini, Ekin D. Cubuk, et al. 2020. "ReMixMatch: Semi-Supervised Learning with Distribution Alignment and Augmentation Anchoring." *International Conference on Learning Representations* (*ICLR*). https://arxiv.org/abs/1911.09785.

Besag, Julian. 1974. "Spatial Interaction and the Statistical Analysis of Lattice Systems." *Journal of the Royal Statistical Society: Series B* (*Methodological*) 36 (2): 192–236.

Bian, Andrew An, Joachim M. Buhmann, Andreas Krause, and Sebastian Tschiatschek. 2017. "Guarantees for Greedy Maximization of Non-Submodular Functions with Applications." *International Conference on Machine Learning* (*ICML*), 498–507.

Bian, Yatao. 2020. *Awesome Energy-Based Models/Learning: A Comprehensive List of Energy-Based Learning Papers and Materials*. Https://github.com/yataobian/awesome-ebm.

Bian, Yatao, Yu Rong, Tingyang Xu, Jiaxiang Wu, Andreas Krause, and Junzhou Huang. 2022. "Energy-Based Learning for Cooperative Games, with Applications to Valuation Problems in Machine Learning." *International Conference on Learning Representations* (*ICLR*). https://openreview.net/forum?id=xLfAgCroImw.

Boltzmann, Ludwig. 1877. "Über Die Beziehung Zwischen Dem Zweiten Hauptsatze Der Mechanischen Wärmetheorie Und Der Wahrscheinlichkeitsrechnung Respektive Den Sätzen Über Das Wärmegleichgewicht." *Sitzungsberichte Der Kaiserlichen Akademie Der Wissenschaften*

in Wien 76: 373–435.

Bridle, John S., Anthony J. R. Heading, and David J. C. MacKay. 1992. "Unsupervised Classifiers, Mutual Information and 'Phantom Targets'." *Advances in Neural Information Processing Systems (NIPS)* 4, 1096–101.

Callen, Herbert B. 1985. *Thermodynamics and an Introduction to Thermostatistics*. 2nd ed. Wiley.

Carcamo, David P., Nicholas J. Weaver, Purushottam D. Dixit, and Christopher W. Lynn. 2025. "Minimax Entropy: The Statistical Physics of Optimal Models." *Physical Review E* 112 (6): 061001. https://doi.org/10.1103/kr9x-q59y.

Caron, Mathilde, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. 2020. "Unsupervised Learning of Visual Features by Contrasting Cluster Assignments." *Advances in Neural Information Processing Systems (NeurIPS)*. https://arxiv.org/abs/2006.09882.

Caron, Mathilde, Hugo Touvron, Ishan Misra, et al. 2021. "Emerging Properties in Self-Supervised Vision Transformers." *IEEE/CVF International Conference on Computer Vision (ICCV)*. https://arxiv.org/abs/2104.14294.

Clausius, Rudolf. 1865. "Über Verschiedene Für Die Anwendung Bequeme Formen Der Hauptgleichungen Der Mechanischen Wärmetheorie." *Annalen Der Physik Und Chemie* 125: 353–400.

Cohen, Jacob. 1960. "A Coefficient of Agreement for Nominal Scales." *Educational and Psychological Measurement* 20 (1): 37–46. https://doi.org/10.1177/001316446002000104.

Cui, Ganqu, Yuchen Zhang, Jiacheng Chen, et al. 2025. "The Entropy Mechanism of Reinforcement Learning for Reasoning Language Models." *arXiv Preprint arXiv:2505.22617*. https://arxiv.org/abs/2505.22617.

Dawid, Anna, and Yann LeCun. 2024. "Introduction to Latent Variable Energy-Based Models: A Path Towards Autonomous Machine Intelligence." *Journal of Statistical Mechanics: Theory and Experiment* 2024 (10): 104011.

Dayan, Peter, Geoffrey E. Hinton, Radford M. Neal, and Richard S. Zemel. 1995. "The Helmholtz Machine." *Neural Computation* 7 (5): 889–904. https://doi.org/10.1162/neco.1995.7.5.889.

DeepSeek-AI. 2025. "DeepSeek-R1 Incentivizes Reasoning in LLMs Through Reinforcement Learning." *Nature* 645. https://doi.org/10.1038/s41586-025-09422-z.

Della Pietra, Stephen, Vincent Della Pietra, and John Lafferty. 1997. "Inducing Features of Random Fields." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 19 (4): 380–93.

https://doi.org/10.1109/34.588021.

Deng, Yuntian, Anton Bakhtin, Myle Ott, Arthur Szlam, and Marc'Aurelio Ranzato. 2020. "Residual Energy-Based Models for Text Generation." *International Conference on Learning Representations* (*ICLR*).

Dhariwal, Prafulla, and Alexander Nichol. 2021. "Diffusion Models Beat GANs on Image Synthesis." *Advances in Neural Information Processing Systems* (*NeurIPS*).

Du, Yilun, Shuang Li, and Igor Mordatch. 2020. "Compositional Visual Generation with Energy Based Models." *Advances in Neural Information Processing Systems* (*NeurIPS*).

Du, Yilun, and Igor Mordatch. 2019. "Implicit Generation and Modeling with Energy-Based Models." *Advances in Neural Information Processing Systems* (*NeurIPS*).

Farquhar, Sebastian, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. 2024. "Detecting Hallucinations in Large Language Models Using Semantic Entropy." *Nature* 630: 625–30. https://doi.org/10.1038/s41586-024-07421-0.

Feder, M. 1986. "Maximum Entropy as a Special Case of the Minimum Description Length Criterion." *IEEE Transactions on Information Theory* 32: 847–49.

Friedman, Dan, and Adji Bousso Dieng. 2023. "The Vendi Score: A Diversity Evaluation Metric for Machine Learning." *Transactions on Machine Learning Research*. https://arxiv.org/abs/2210.02410.

Friston, Karl. 2010. "The Free-Energy Principle: A Unified Brain Theory?" *Nature Reviews Neuroscience* 11 (2): 127–38.

Geiger, Dan, David Heckerman, Henry King, and Christopher Meek. 2001. "Stratified Exponential Families: Graphical Models and Model Selection." *The Annals of Statistics* 29 (2): 505–29. https://doi.org/10.1214/aos/1009210550.

Gibbs, Josiah Willard. 1902. *Elementary Principles in Statistical Mechanics*. Charles Scribner's Sons.

Gnaiger, Erich. 2009. "Open and Closed Systems: Styles of Thinking Explain Controversies on the 'Negative Entropy' Concept of Ludwig Boltzmann and Erwin Schrödinger." *Mitochondrial Physiology Network*. https://www.mitophysiology.org/images/2/23/Gnaiger_2009_OCESHS.pdf.

Gomes, Ryan, Andreas Krause, and Pietro Perona. 2010. "Discriminative Clustering by Regularized Information Maximization." *Advances in Neural Information Processing Systems* (*NeurIPS*) 23.

Grandvalet, Yves, and Yoshua Bengio. 2004. "Semi-Supervised Learning by Entropy Minimization." *Advances in Neural Information Processing Systems* (*NeurIPS*) 17.

Grathwohl, Will, Kevin Swersky, Milad Hashemi, David Duvenaud, and Chris Maddison. 2021. "Oops I Took a Gradient: Scalable Sampling for Discrete Distributions." *International Conference on Machine Learning* (*ICML*), 3831–41.

Grathwohl, Will, Kuan-Chieh Wang, Jörn-Henrik Jacobsen, David Duvenaud, Mohammad Norouzi, and Kevin Swersky. 2020. "Your Classifier Is Secretly an Energy Based Model and You Should Treat It Like One." *International Conference on Learning Representations* (*ICLR*).

Grünwald, Peter D. 2007. *The Minimum Description Length Principle*. MIT Press.

Gutmann, Michael, and Aapo Hyvärinen. 2010. "Noise-Contrastive Estimation: A New Estimation Principle for Unnormalized Statistical Models." *International Conference on Artificial Intelligence and Statistics* (*AISTATS*), 297–304.

GX-Chen, Anthony, Jatin Prakash, Jeff Guo, Rob Fergus, and Rajesh Ranganath. 2025. "KL-Regularized Reinforcement Learning Is Designed to Mode Collapse." *arXiv Preprint arXiv:2510.20817*. https://arxiv.org/abs/2510.20817.

Haarnoja, Tuomas, Haoran Tang, Pieter Abbeel, and Sergey Levine. 2017. "Reinforcement Learning with Deep Energy-Based Policies." *International Conference on Machine Learning* (*ICML*), 1352–61.

Haarnoja, Tuomas, Aurick Zhou, Pieter Abbeel, and Sergey Levine. 2018. "Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor." *International Conference on Machine Learning* (*ICML*), 1861–70.

Hill, M. O. 1973. "Diversity and Evenness: A Unifying Notation and Its Consequences." *Ecology* 54 (2): 427–32. https://doi.org/10.2307/1934352.

Hinton, Geoffrey E. 2002. "Training Products of Experts by Minimizing Contrastive Divergence." *Neural Computation* 14 (8): 1771–800. https://doi.org/10.1162/089976602760128018.

Hirsh, Jacob B., Raymond A. Mar, and Jordan B. Peterson. 2012. "Psychological Entropy: A Framework for Understanding Uncertainty-Related Anxiety." *Psychological Review* 119 (2): 304–20. https://doi.org/10.1037/a0026767.

Ho, Jonathan, Ajay Jain, and Pieter Abbeel. 2020. "Denoising Diffusion Probabilistic Models." *Advances in Neural Information Processing Systems* (*NeurIPS*).

Hopfield, John J. 1982. "Neural Networks and Physical Systems with Emergent Collective Computational Abilities." *Proceedings of the National Academy of Sciences* 79 (8): 2554–58.

https://doi.org/10.1073/pnas.79.8.2554.

Hu, Weihua, Takeru Miyato, Seiya Tokui, Eiichi Matsumoto, and Masashi Sugiyama. 2017. "Learning Discrete Representations via Information Maximizing Self-Augmented Training." *International Conference on Machine Learning* (*ICML*). https://arxiv.org/abs/1702.08720.

Hyvärinen, Aapo. 2005. "Estimation of Non-Normalized Statistical Models by Score Matching." *Journal of Machine Learning Research* 6: 695–709.

Hyvärinen, Aapo. 2007. "Connections Between Score Matching, Contrastive Divergence, and Pseudolikelihood for Continuous-Valued Variables." *IEEE Transactions on Neural Networks* 18 (5): 1529–31.

Jaynes, Edwin T. 1957. "Information Theory and Statistical Mechanics." *Physical Review* 106: 620.

Jaynes, Edwin T. 1980. "The Minimum Entropy Production Principle." *Annual Review of Physical Chemistry* 31: 579–601. https://doi.org/10.1146/annurev.pc.31.100180.003051.

Kianercy, Ardeshir, and Aram Galstyan. 2012. "Dynamics of Boltzmann q Learning in Two-Player Two-Action Games." *Physical Review E* 85 (4): 041145. https://doi.org/10.1103/PhysRevE.85.041145.

Kirkpatrick, Scott, C. Daniel Gelatt, and Mario P. Vecchi. 1983. "Optimization by Simulated Annealing." *Science* 220 (4598): 671–80. https://doi.org/10.1126/science.220.4598.671.

Koller, Daphne, and Nir Friedman. 2009. *Probabilistic Graphical Models: Principles and Techniques*. MIT Press.

Kuhn, Lorenz, Yarin Gal, and Sebastian Farquhar. 2023. "Semantic Uncertainty: Linguistic Invariances for Uncertainty Estimation in Natural Language Generation." *International Conference on Learning Representations* (*ICLR*). https://arxiv.org/abs/2302.09664.

Lafferty, John, Andrew McCallum, and Fernando Pereira. 2001. "Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data." *International Conference on Machine Learning* (*ICML*), 282–89.

Landauer, Rolf. 1975. "Inadequacy of Entropy and Entropy Derivatives in Characterizing the Steady State." *Physical Review A* 12 (2): 636–38. https://doi.org/10.1103/PhysRevA.12.636.

LeCun, Yann. 2022. "A Path Towards Autonomous Machine Intelligence." *OpenReview Preprint*.

LeCun, Yann, Sumit Chopra, Raia Hadsell, Marc'Aurelio Ranzato, and Fu Jie Huang. 2006. "A Tutorial on Energy-Based Learning." In *Predicting Structured Data*, edited by Gökhan Bakir,

Thomas Hofmann, Bernhard Schölkopf, Alexander J. Smola, and Ben Taskar. MIT Press.

Lee, Dong-Hyun. 2013. "Pseudo-Label: The Simple and Efficient Semi-Supervised Learning Method for Deep Neural Networks." *ICML Workshop on Challenges in Representation Learning*.

Lee, Jonghyun, Dahuin Jung, Saehyung Lee, et al. 2024. "Entropy Is Not Enough for Test-Time Adaptation: From the Perspective of Disentangled Factors." *International Conference on Learning Representations* (*ICLR*). https://arxiv.org/abs/2403.07366.

Leinster, Tom, and Christina A. Cobbold. 2012. "Measuring Diversity: The Importance of Species Similarity." *Ecology* 93 (3): 477–89. https://doi.org/10.1890/10-2402.1.

Liang, Jian, Dapeng Hu, and Jiashi Feng. 2020. "Do We Really Need to Access the Source Data? Source Hypothesis Transfer for Unsupervised Domain Adaptation." *International Conference on Machine Learning* (*ICML*). https://arxiv.org/abs/2002.08546.

Liu, Weitang, Xiaoyun Wang, John D. Owens, and Yixuan Li. 2020. "Energy-Based Out-of-Distribution Detection." *Advances in Neural Information Processing Systems* (*NeurIPS*).

Lynn, Christopher W., Qiwei Yu, Rich Pang, Stephanie E. Palmer, and William Bialek. 2025. "Exact Minimax Entropy Models of Large-Scale Neuronal Activity." *Physical Review E* 111 (5): 054411. https://doi.org/10.1103/PhysRevE.111.054411.

Ma, Xueguang, Qian Liu, Dongfu Jiang, Ge Zhang, Zejun Ma, and Wenhu Chen. 2025. "General-Reasoner: Advancing LLM Reasoning Across All Domains." *arXiv Preprint arXiv:2505.14652*. https://arxiv.org/abs/2505.14652.

Maes, Christian, and Karel Netočný. 2007. "Minimum Entropy Production Principle from a Dynamical Fluctuation Law." *Journal of Mathematical Physics* 48 (5): 053306. https://doi.org/10.1063/1.2738753.

Maes, Christian, and Karel Netočný. 2013. "Minimum Entropy Production Principle." *Scholarpedia* 8 (7): 9664. https://doi.org/10.4249/scholarpedia.9664.

Martyushev, Leonid M., A. S. Nazarova, and Vladimir D. Seleznev. 2007. "On the Problem of the Minimum Entropy Production in the Nonequilibrium Stationary State." *Journal of Physics A: Mathematical and Theoretical* 40 (3): 371–80. https://doi.org/10.1088/1751-8113/40/3/002.

Martyushev, Leonid M., and Vladimir D. Seleznev. 2006. "Maximum Entropy Production Principle in Physics, Chemistry and Biology." *Physics Reports* 426 (1): 1–45. https://doi.org/10.1016/j.physrep.2005.12.001.

Martyushev, Leonid M., and Vladimir D. Seleznev. 2013. "Entropy and Entropy Production: Old Misconceptions and New Breakthroughs." *Entropy* 15 (4): 1152–70. https://doi.org/10.

3390/e15041152.

Menon, Aditya Krishna, Sadeep Jayasumana, Ankit Singh Rawat, Himanshu Jain, Andreas Veit, and Sanjiv Kumar. 2021. "Long-Tail Learning via Logit Adjustment." *International Conference on Learning Representations* (*ICLR*). https://arxiv.org/abs/2007.07314.

Mnih, Andriy, and Yee Whye Teh. 2012. "A Fast and Simple Algorithm for Training Neural Probabilistic Language Models." *International Conference on Machine Learning* (*ICML*), 419–26.

Moussouris, John. 1974. "Gibbs and Markov Random Systems with Constraints." *Journal of Statistical Physics* 10 (1): 11–33. https://doi.org/10.1007/BF01011714.

Neal, Radford M., and Geoffrey E. Hinton. 1998. "A View of the EM Algorithm That Justifies Incremental, Sparse, and Other Variants." In *Learning in Graphical Models*, edited by Michael I. Jordan. Kluwer Academic Publishers.

Nemhauser, George L., Laurence A. Wolsey, and Marshall L. Fisher. 1978. "An Analysis of Approximations for Maximizing Submodular Set Functions—i." *Mathematical Programming* 14: 265–94.

Neumann, John von. 1927. "Thermodynamik Quantenmechanischer Gesamtheiten." *Nachrichten von Der Gesellschaft Der Wissenschaften Zu Göttingen, Mathematisch-Physikalische Klasse* 1927: 273–91.

Nijkamp, Erik, Mitch Hill, Song-Chun Zhu, and Ying Nian Wu. 2019. "Learning Non-Convergent Non-Persistent Short-Run MCMC Toward Energy-Based Model." *Advances in Neural Information Processing Systems* (*NeurIPS*).

Nikitin, Alexander, Jannik Kossen, Yarin Gal, and Pekka Marttinen. 2024. "Kernel Language Entropy: Fine-Grained Uncertainty Quantification for LLMs from Semantic Similarities." *Advances in Neural Information Processing Systems* (*NeurIPS*). https://arxiv.org/abs/2405.20003.

Niu, Shuaicheng, Jiaxiang Wu, Yifan Zhang, et al. 2022. "Efficient Test-Time Model Adaptation Without Forgetting." *International Conference on Machine Learning* (*ICML*). https://arxiv.org/abs/2204.02610.

Niu, Shuaicheng, Jiaxiang Wu, Yifan Zhang, et al. 2023. "Towards Stable Test-Time Adaptation in Dynamic Wild World." *International Conference on Learning Representations* (*ICLR*). https://arxiv.org/abs/2302.12400.

Owen, Guillermo. 1972. "Multilinear Extensions of Games." *Management Science* 18 (5): P64–79. https://doi.org/10.1287/mnsc.18.5.P64.

Owen, Guillermo. 1975. "Multilinear Extensions and the Banzhaf Value." *Naval Research Logistics Quarterly* 22 (4): 741–50. https://doi.org/10.1002/nav.3800220409.

Paltridge, Garth W. 1975. "Global Dynamics and Climate — a System of Minimum Entropy Exchange." *Quarterly Journal of the Royal Meteorological Society* 101 (429): 475–84. https://doi.org/10.1002/qj.49710142906.

Paninski, Liam. 2003. "Estimation of Entropy and Mutual Information." *Neural Computation* 15 (6): 1191–253. https://doi.org/10.1162/089976603321780272.

Pasarkar, Amey P., and Adji Bousso Dieng. 2024. "Cousins of the Vendi Score: A Family of Similarity-Based Diversity Metrics for Science and Machine Learning." *International Conference on Artificial Intelligence and Statistics (AISTATS)*, PMLR, vol. 238. https://arxiv.org/abs/2310.12952.

Pearl, Judea. 1988. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann.

Prigogine, Ilya. 1945. "Modération Et Transformations Irréversibles Des Systèmes Ouverts." *Bulletin de La Classe Des Sciences, Académie Royale de Belgique* 31: 600–606.

Prigogine, Ilya. 1947. *Étude Thermodynamique Des Phénomènes Irréversibles*. Dunod, Paris; Desoer, Liège.

Prigogine, Ilya. 1978. "Time, Structure, and Fluctuations." *Science* 201 (4358): 777–85. https://doi.org/10.1126/science.201.4358.777.

Prigogine, Ilya, and Jean-Marie Wiame. 1946. "Biologie Et Thermodynamique Des Phénomènes Irréversibles." *Experientia* 2 (11): 451–53. https://doi.org/10.1007/BF02153597.

Qin, Lianhui, Sean Welleck, Daniel Khashabi, and Yejin Choi. 2022. "COLD Decoding: Energy-Based Constrained Text Generation with Langevin Dynamics." *Advances in Neural Information Processing Systems (NeurIPS)*.

Rényi, Alfréd. 1961. "On Measures of Entropy and Information." *Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Contributions to the Theory of Statistics*, 547–61.

Rose, Kenneth, Eitan Gurewitz, and Geoffrey C. Fox. 1990. "Statistical Mechanics and Phase Transitions in Clustering." *Physical Review Letters* 65 (8): 945–48. https://doi.org/10.1103/PhysRevLett.65.945.

Sánchez Giraldo, Luis Gonzalo, Murali Rao, and José C. Príncipe. 2015. "Measures of Entropy from Data Using Infinitely Divisible Kernels." *IEEE Transactions on Information Theory* 61

(1): 535–48. https://doi.org/10.1109/TIT.2014.2370058.

Schrödinger, Erwin. 1944. *What Is Life? The Physical Aspect of the Living Cell*. Cambridge University Press.

Shannon, Claude E. 1948. "A Mathematical Theory of Communication." *Bell System Technical Journal* 27 (3): 379–423.

Shao, Zhihong, Peiyi Wang, Qihao Zhu, et al. 2024. "DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models." *arXiv Preprint arXiv:2402.03300*. https://arxiv.org/abs/2402.03300.

Shi, Chence, Shitong Luo, Minkai Xu, and Jian Tang. 2021. "Learning Gradient Fields for Molecular Conformation Generation." *International Conference on Machine Learning* (*ICML*).

Sohl-Dickstein, Jascha, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. 2015. "Deep Unsupervised Learning Using Nonequilibrium Thermodynamics." *International Conference on Machine Learning* (*ICML*), 2256–65.

Song, Yang, and Stefano Ermon. 2019. "Generative Modeling by Estimating Gradients of the Data Distribution." *Advances in Neural Information Processing Systems* (*NeurIPS*).

Song, Yang, Sahaj Garg, Jiaxin Shi, and Stefano Ermon. 2019. "Sliced Score Matching: A Scalable Approach to Density and Score Estimation." *Uncertainty in Artificial Intelligence* (*UAI*).

Song, Yang, and Diederik P. Kingma. 2021. "How to Train Your Energy-Based Models." *arXiv Preprint arXiv:2101.03288*.

Song, Yang, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. 2021. "Score-Based Generative Modeling Through Stochastic Differential Equations." *International Conference on Learning Representations* (*ICLR*).

Tanaka, Daiki, Daiki Ikami, Toshihiko Yamasaki, and Kiyoharu Aizawa. 2018. "Joint Optimization Framework for Learning with Noisy Labels." *IEEE Conference on Computer Vision and Pattern Recognition* (*CVPR*). https://arxiv.org/abs/1803.11364.

The Royal Swedish Academy of Sciences. 2024. *Scientific Background to the Nobel Prize in Physics 2024: For Foundational Discoveries and Inventions That Enable Machine Learning with Artificial Neural Networks*. https://www.nobelprize.org/prizes/physics/2024/advanced-information/.

Tieleman, Tijmen. 2008. "Training Restricted Boltzmann Machines Using Approximations to the Likelihood Gradient." *International Conference on Machine Learning* (*ICML*), 1064–71.

Tsallis, Constantino. 1988. "Possible Generalization of Boltzmann–Gibbs Statistics." *Journal of Statistical Physics* 52 (1–2): 479–87.

Verschaffelt, Jules-Émile. 1954. "Sur Les Minima de Production d'entropie Et de Dissipation d'énergie." *Bulletin de La Classe Des Sciences, Académie Royale de Belgique* 40: 779–83.

Vincent, Pascal. 2011. "A Connection Between Score Matching and Denoising Autoencoders." *Neural Computation* 23 (7): 1661–74.

Wainwright, Martin J., and Michael I. Jordan. 2008. "Graphical Models, Exponential Families, and Variational Inference." *Foundations and Trends in Machine Learning* 1 (1–2): 1–305. https://doi.org/10.1561/2200000001.

Wang, Dequan, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. 2021. "Tent: Fully Test-Time Adaptation by Entropy Minimization." *International Conference on Learning Representations (ICLR)*. https://arxiv.org/abs/2006.10726.

Wang, Xudong, Zhirong Wu, Long Lian, and Stella X. Yu. 2022. "Debiased Learning from Naturally Imbalanced Pseudo-Labels." *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 14647–57. https://doi.org/10.1109/CVPR52688.2022.01424.

Wu, Fa-Yueh. 1982. "The Potts Model." *Reviews of Modern Physics* 54 (1): 235–68. https://doi.org/10.1103/RevModPhys.54.235.

Wu, Jiaxiang, Tao Shen, Haidong Lan, Yatao Bian, and Junzhou Huang. 2021. "SE(3)-Equivariant Energy-Based Models for End-to-End Protein Folding." *bioRxiv Preprint*.

Wu, Tailin, and Ian Fischer. 2020. "Phase Transitions for the Information Bottleneck in Representation Learning." *International Conference on Learning Representations (ICLR)*. https://arxiv.org/abs/2001.01878.

Wu, Tailin, Ian Fischer, Isaac L. Chuang, and Max Tegmark. 2019. "Learnability for the Information Bottleneck." *Uncertainty in Artificial Intelligence (UAI)*. https://arxiv.org/abs/1907.07331.

Zhang, Qingyang, Haitao Wu, Changqing Zhang, Peilin Zhao, and Yatao Bian. 2025. "Right Question Is Already Half the Answer: Fully Unsupervised LLM Reasoning Incentivization." *Advances in Neural Information Processing Systems (NeurIPS)*. https://arxiv.org/abs/2504.05812.

Zhao, Xuandong, Zhewei Kang, Aosong Feng, Sergey Levine, and Dawn Song. 2025. "Learning to Reason Without External Rewards." *arXiv Preprint arXiv:2505.19590*. https://arxiv.org/abs/2505.19590.

Zhu, Song Chun, Ying Nian Wu, and David Mumford. 1997. "Minimax Entropy Principle and Its Application to Texture Modeling." *Neural Computation* 9 (8): 1627–60. https://doi.org/10.1162/neco.1997.9.8.1627.

Ziebart, Brian D., Andrew Maas, J. Andrew Bagnell, and Anind K. Dey. 2008. "Maximum Entropy Inverse Reinforcement Learning." *AAAI Conference on Artificial Intelligence*, 1433–38.