Statistical Mechanics & Machine Learning: An In-Depth Exploration

This note provides a detailed explanation of how statistical mechanics principles translate into modern machine learning. We explore energy-based models, the Boltzmann distribution, mean-field theory, Ising machines, and field theories, along with in-depth mathematics behind these concepts.

---

1. Overview of Statistical Mechanics

Statistical mechanics connects the microscopic behavior of individual particles with the macroscopic properties (e.g., temperature, pressure) we observe. It enables predictions about large systems, often comprising on the order of 10^{23} atoms—without tracking every individual component.

Key Definitions

- **Microstate:**
  A specific configuration of a system. For example, in 5 coin flips, one microstate is `H-T-T-H-H`
- **Macrostate:**
  A description based on macroscopic properties. In the coin flip example, a macrostate may be defined by the total number of heads. For instance:

  $P(5\ heads) = \frac{1}{32}, \quad P(3 \text{ heads}) = \frac{\binom{5}{3}}{32}$
- **Entropy:**
  Defined as:

  $S = k \ln(\Omega)$

  where Ω is the number of accessible microstates and k is Boltzmann's constant. The logarithmic relationship is essential for handling the enormous number of microstates.
- **Reservoir:**
  A large system with which our system exchanges energy (or particles) without significantly altering its own macroscopic state.

- **Fundamental Assumption:**
  For an **isolated system**, all microstates are equally probable.

---

2. The Boltzmann Distribution and Free Energy

2.1 Boltzmann Distribution

The Boltzmann distribution gives the probability $P(s)$ that a system is in a state s with energy $E(s)$:

$P(s) = \frac{1}{Z} e^{-E(s)/kT}$

where:

- T is the absolute temperature.
- k is Boltzmann's constant.
- Z is the partition function:

  $Z = \sum_s e^{-E(s)/kT}$

  (or an integral in the continuous case), ensuring that the probabilities sum to one.

2.2 Derivation Intuition

1. **Equal A Priori Probability:**
   In an isolated system, every microstate is equally likely. When the system is in contact with a large reservoir, the combined (system + reservoir) is isolated. The probability of a given microstate is then influenced by how many states the reservoir can assume after absorbing the energy difference.
2. **Role of Entropy:**
   The reservoir's entropy S is related to the number of microstates:

   $S = k \ln(\Omega)$

   Expanding S for the reservoir around its mean energy U and using the thermodynamic identity:

   $\frac{1}{T} = \frac{\partial S}{\partial U}$

   we obtain:

   $\frac{P(E_i)}{P(E_j)} \approx \exp\left(\frac{E_j - E_i}{kT}\right)$

Normalization leads to the Boltzmann distribution:

$P(E_i) = \frac{1}{Z} \exp\left(-\frac{E_i}{kT}\right)$

2.3 Free Energy

The **free energy** F of a system is defined as:

$F = -kT \ln Z$

Free energy represents the balance between internal energy and entropy. In many ways, minimizing free energy is analogous to optimization in machine learning (seeking the state of lowest "loss.")

---

3. Learning as Inverse Thermodynamics

This framework treats learning as the reverse process of thermodynamic evolution. Instead of a system naturally settling into a low-energy state, in learning we design models so that desirable outputs correspond to low-energy configurations.

3.1 Energy-Based Models (EBMs)

In machine learning, **energy-based models** define an energy function over possible configurations (e.g., states of a neural network), where lower energy implies higher probability or a more desirable outcome.

- **Spin Glass (Equilibrium):**
  A spin glass is a system with interacting spins (or neurons) that can be in one of two states (e.g., +1+1 or −1-1). In equilibrium, the system's probability distribution is given by the Boltzmann distribution.
- **Diffusion Models (Non-Equilibrium):**
  These models are inspired by stochastic processes such as Brownian motion. Instead of sampling from an equilibrium distribution, diffusion models simulate a gradual, time-dependent transformation of data—a key idea in generative modeling.

3.2 The Boltzmann Machine

A **Boltzmann machine** is a network of stochastic binary units (neurons) with an energy function typically given by:

$E(\mathbf{m}) = -\frac{1}{2}\mathbf{m}^T J\mathbf{m} - \mathbf{h}^T\mathbf{m}$

where:

- $\mathbf{m}$ is a vector of binary states (typically ± 1).
- J is a symmetric matrix of interaction weights.
- $\mathbf{h}$ is a bias vector.

The probability of a particular state is:

$P(\mathbf{m}) = \frac{1}{Z}e^{-E(\mathbf{m})/T}$

_(Often we set $\beta = 1/T$ to simplify notation.)

- **Training EBMs:**
  EBMs are trained by adjusting J and $\mathbf{h}$ so that the energy function assigns lower energy (and thus higher probability) to desirable configurations. Sampling techniques such as Gibbs sampling or contrastive divergence are used to approximate gradients during training.

---

4. From Fermi-Dirac to Mean-Field Theory

4.1 Fermi-Dirac Distribution (Non-Interacting Case)

For systems where individual energy levels are independent (e.g., electrons in distinct energy levels), the **Fermi-Dirac distribution** gives the occupation probability of a state with energy E_i:

$P(\text{occupied}) = \frac{1}{1+e^{E_i/kT}}$

This closed-form solution arises because each level is independent.

4.2 Mean-Field Theory (Interacting Case)

When interactions are present, exact computation of the state probabilities becomes intractable due to the exponential number of microstates. **Mean-field theory** approximates the interactions by assuming that each unit experiences an average effect from all others.

1. **Define Effective Energy:**
   For each energy level ii, define an effective energy:

   $\tilde{E}_i = E_i + U \sum_{j \neq i} f(E_j)$

   where:

   - U is the average interaction strength.
   - $f(E_j)$ is the average occupation probability (using the Fermi function) of level j.

2. **Iterative Update:**
   Recompute the occupation probabilities using:

   $f(\tilde{E}_i) = \frac{1}{1+e^{\tilde{E}_i/kT}}$

   Iterate this process until convergence, thereby approximating the behavior of the interacting system with reduced computational complexity.

---

5. Ising Machines & Binary Stochastic Neurons (BSNs)

5.1 The Ising Model

The **Ising model** originates from statistical mechanics to model ferromagnetism. It consists of spins (values $+1$+1 or -1-1) that interact through couplings. Its energy is expressed as:

$E = -\frac{1}{2} \sum_{i,j} J_{ij} m_i m_j - \sum_i h_i m_i$

- J_{ij} represents the interaction strength between spins i and j.
- h_i is the external bias on spin i.

5.2 Mapping to Binary Stochastic Neurons

In machine learning, these spins are interpreted as binary stochastic neurons (BSNs):

- **Neurons Spins:**
  Each BSN takes a binary value ($m_i \in \{-1, +1\}$).
- **Neural Energy Function:**
  The energy function becomes:

  $E(\mathbf{m}) = \frac{1}{2}\mathbf{m}^T J\mathbf{m} + \mathbf{h}^T\mathbf{m}$

  This landscape governs the dynamics of the network.

5.3 Dynamics and Update Equations

A common update rule for BSNs is:

$m_i = \text{sgn}\left(\tanh(\beta I_i) + \text{noise}\right)$

where:

- $I_i = \sum_j J_{ij} m_j + h_i$ is the total input to neuron ii.
- $\beta = 1/T$ is the inverse temperature.
- The noise term ensures stochastic exploration of the energy landscape.

5.4 Invertible Computing

A fascinating feature of these networks is **invertible computing**:

- **Forward Computation:** Clamping the input neurons allows the network to compute an output (e.g., classification).
- **Reverse Computation:** Clamping the output can help infer the possible inputs (useful in generative modeling or solving inverse problems such as factorization).

---

6. Advanced Mathematical Details

6.1 Derivation of the Boltzmann Distribution (Sketch)

1. **Setup:**
   Consider a system in contact with a large reservoir. The total energy is fixed:

   $U_{\text{total}} = E_{\text{system}} + E_{\text{reservoir}}$

2. **Probability Ratio:**
   The probability of the system having energy E is proportional to the number of reservoir states available when the reservoir's energy is $U_{\text{total}} - E$:

   $P(E) \propto \Omega_{\text{reservoir}}(U_{\text{total}} - E)$

3. **Using Entropy:**
   Write the reservoir's entropy as:

   $S = k \ln \Omega_{\text{reservoir}}$

   Expanding around the mean energy and using

   $\frac{1}{T} = \frac{\partial S}{\partial U}$

   gives:

   $S(U_{\text{total}} - E) \approx S(U_{\text{total}}) - \frac{E}{T}$

4. **Exponential Form and Normalization:**
   This leads to:

   $P(E) \propto e^{-E/kT} \quad \Longrightarrow \quad P(E) = \frac{1}{Z} e^{-E/kT}$

   with the partition function:

   $Z = \sum_E e^{-E/kT}$

6.2 Mean-Field Theory: Convergence

- **Self-Consistency Equation:**
  The iterative update process leads to:

  $f_i = \frac{1}{1+e^{(E_i + U \sum_{j \neq i} f_j)/kT}}$

  Solving these equations yields the mean occupation probabilities.

- **Convergence Criteria:**
  Under conditions like weak interactions or high temperature, the iterative process converges to a fixed point that approximates the true behavior of the system.

6.3 Additional Example: Electrical Circuits

Consider the energy stored in a capacitor:

$E = \frac{1}{2}CV^2$

Thermal fluctuations in voltage can be characterized by:

$\sqrt{\langle V^2 \rangle} = \sqrt{\frac{kT}{C}}$

This relation, derived from the equipartition theorem, links temperature with the average energy per degree of freedom—a core concept in statistical mechanics.

---

7. Conclusion

In summary, the interplay between statistical mechanics and machine learning is built on several key ideas:

- **Foundations in Statistical Mechanics:**
  Understanding microstates, macrostates, and entropy allows us to describe large systems without tracking every detail.
- **Boltzmann Distribution & Free Energy:**
  Deriving probabilities for states and introducing the partition function leads to the concept of free energy, which parallels optimization in machine learning.
- **Energy-Based Models & Learning as Inverse Thermodynamics:**
  EBMs, such as Boltzmann machines, use energy functions to model data, where lower energy states are more desirable. Diffusion models extend these ideas into non-equilibrium settings.
- **Mean-Field Theory:**
  Approximates interactions in complex systems by averaging, reducing computational complexity.
- **Ising Machines & BSNs:**
  These models, derived from the Ising model, provide a framework for both forward and inverse computations, with applications in classification, generative modeling, and combinatorial optimization.

---

8. Boltzmann Machines: Learning via Log-Likelihood and Contrastive Divergence

8.1 Log-Likelihood Formulation

Given a Boltzmann machine with energy function:

$$E(\mathbf{m}) = -\sum_{i,j} J_{ij} m_i m_j - \sum_i h_i m_i$$

we define the probability of a configuration $\mathbf{m}$ as:

$$P(\mathbf{m}) = \frac{1}{Z} e^{-E(\mathbf{m})}$$

where $Z = \sum_{\mathbf{m}} e^{-E(\mathbf{m})}$ is the **partition function**.

For a dataset $\mathcal{D}$, the average log-likelihood is:

$$\mathcal{L}(\theta) = \mathbb{E}_{\mathbf{m}\sim\mathcal{D}}[\log P(\mathbf{m}|\theta)] = -\log Z + \sum_{i,j} J_{ij} \langle m_i m_j \rangle_{\text{data}} + \sum_i h_i \langle m_i \rangle_{\text{data}}$$

Taking derivatives yields the following learning rules:

$$\frac{\partial \mathcal{L}}{\partial J_{ij}} = \langle m_i m_j \rangle_{\text{data}} - \langle m_i m_j \rangle_{\text{model}}, \quad \frac{\partial \mathcal{L}}{\partial h_i} = \langle m_i \rangle_{\text{data}} - \langle m_i \rangle_{\text{model}}$$

Training thus involves reducing the difference between observed statistics and model statistics.

8.2 Sampling with Gibbs and the Restricted Boltzmann Machine

To address the intractability of the partition function:

- Use **Gibbs sampling** to estimate $\langle m_i m_j \rangle_{\text{model}}$
- Use **Contrastive Divergence (CD-k)** to approximate gradients:
  - Initialize visible units with data
  - Run k steps of Gibbs sampling
  - Compute difference of correlations between data and reconstructions

In **Restricted Boltzmann Machines (RBMs)**, the structure is bipartite:

$$\frac{\partial \mathcal{L}(v)}{\partial W_{ij}} = \langle h_i v_j \rangle_{\text{data}} - \langle h_i v_j \rangle_{\text{recon}}$$

9. Score-Based Generative Models and Diffusion

9.1 From EBMs to Score Matching

Instead of modeling $P(x)$ directly, model its **score**:

$$\nabla_x \log P(x)$$

This avoids the partition function, since:

$$\nabla_x \log P(x) = -\nabla_x E(x) - \nabla_x \log Z \Rightarrow \text{Score is independent of } Z$$

We train a model $s_\theta(x)$ to minimize **Fisher divergence**:

$$\mathcal{F}(P_{\text{data}}, s_\theta) = \mathbb{E}_{x \sim P_{\text{data}}}[\|\nabla_x \log P(x) - s_\theta(x)\|^2]$$

9.2 Langevin Dynamics

A physics-inspired iterative sampler:

$$x^{(t+1)} = x^{(t)} + \eta s_\theta(x^{(t)}) + \sqrt{2\eta} \cdot \epsilon, \quad \epsilon \sim \mathcal{N}(0, I)$$

9.3 Diffusion Models as SDEs

We define a forward **diffusion process** that adds noise to x_0 over time t:

$$\mathrm{d}x_t = f(x_t, t)\,\mathrm{d}t + g(t)\,\mathrm{d}W_t$$

- Common choice:

$$\mathrm{d}x_t = -\frac{1}{2}\beta(t)x_t\,\mathrm{d}t + \sqrt{\beta(t)}\,\mathrm{d}W_t$$

This guarantees:

- Mean: $\mathbb{E}[x_t|x_0] = x_0 e^{-\frac{1}{2}\beta(t)}$
- Variance: $\mathrm{Var}(x_t|x_0) = 1 - e^{-\beta(t)}$

9.4 Reversing the Diffusion Process

To sample, we **reverse** the SDE:

$$\mathrm{d}\bar{x}_t = [f(x_t, t) - g(t)^2 \nabla_x \log p_t(x_t)]\mathrm{d}t + g(t)\mathrm{d}\bar{W}_t$$

Euler-Maruyama discretization:

$$x_{t-\Delta t} = x_t + [f(x_t, t) - g^2(t) \cdot s_\theta(x_t, t)]\Delta t + g(t)\sqrt{\Delta t}\epsilon$$

We can also use a **predictor-corrector** scheme:

1. Predictor: simulate reverse SDE step
2. Corrector: denoise via Langevin step

9.5 Score-Based Diffusion Models: Summary

- Models the **score** $\nabla_x \log p_t(x)$ via neural network $s_\theta(x, t)$
- Trained on noisy data with multiple noise scales
- Inference is performed by reversing an SDE using the learned score
- Equivalent to DDPM, SMLD, VP-SDE, and more

9.6 Mathematical Insight: Ito Calculus

In stochastic calculus, integrals behave differently:

$$\int_0^t W(s)\,\mathrm{d}W(s) \neq \frac{1}{2}W(t)^2$$

Instead:

$$\int_0^t W(s)\,\mathrm{d}W(s) = \frac{1}{2}W(t)^2 - \frac{1}{2}t$$

This is due to $\mathrm{d}W_t^2 = \mathrm{d}t$ in Ito calculus, a key difference from classical calculus.

---

10. From Energy Landscapes to Neural Generative Dynamics

- **Boltzmann Machines:** Equilibrium models, sampling via Gibbs or MCMC
- **Diffusion Models:** Non-equilibrium models, driven by SDEs and score-based learning

Each model encodes a probabilistic data distribution, but differ in parameterization:

- Boltzmann: $P(x) \propto e^{-E(x)}$
- Diffusion: $x_0 \to x_t$ via forward SDE, recover x_0 by reversing with score estimate

The diffusion paradigm shifts generative modeling from equilibrium sampling to **dynamical transformation**, bringing in deep connections to thermodynamics and stochastic physics.