# Machine learning approach to the inverse Ising problem

Martí Pedemonte Bernat

Universitat de Barcelona
Faculty of Physics

Treball de Final de Grau
31st of January 2020

# Introduction

Two different problems in physics:

- **Direct\forward problem:** Predict the behavior of a system applying a model with known parameters (e.g., weather forecasting).
- **Reverse\inverse problem:** Determine the parameters of a model from a set of observations (e.g., X-ray computed tomography).

# Introduction: The Ising model

## Ising model

Mathematical model consisting of a set of binary variables (spins), $\boldsymbol{\sigma} \equiv \{\sigma_i\}$ $i = 1, \ldots, N$ with $\sigma_i = \pm 1$, subject to external fields $h_i$ and coupled by pairwise couplings $J_{ij}$.

Hamiltonian:

$$H_{\boldsymbol{J},\boldsymbol{h}}(\boldsymbol{\sigma}) = -\sum_i h_i \sigma_i - \sum_{i<j} J_{ij} \sigma_i \sigma_j$$

Boltzmann distribution:

$$P(\boldsymbol{\sigma}) = \frac{\mathrm{e}^{-\beta H_{\boldsymbol{J},\boldsymbol{h}}(\boldsymbol{\sigma})}}{Z(\boldsymbol{J},\boldsymbol{h})}, \qquad \text{with} \qquad Z(\boldsymbol{J},\boldsymbol{h}) = \sum_{\boldsymbol{\sigma}} \mathrm{e}^{-\beta H_{\boldsymbol{J},\boldsymbol{h}}(\boldsymbol{\sigma})}$$

# Introduction: The Ising model

Origin and first solutions of the direct problem:

- **1920:** Lenz proposes the Ising model to describe ferromagnetism.
- **1925:** Ising solves the 1D direct pure case ($\boldsymbol{h} = h$, $\boldsymbol{J} = J$).
- **1944:** Onsager solves the 2D direct pure case.

If $\boldsymbol{h} = \{h_i\}$, $\boldsymbol{J} = \{J_{ij}\}$ are random variables that can take both positive and negative values, it is called *Ising spin glass*.

- Originally proposed to describe frustrated magnets.
- Much harder, no exact solution known in 2D.

# Introduction: The inverse Ising problem and applications

## Inverse Ising problem

Determination of the couplings $J_{ij}$ and local fields $h_i$ of the Ising model, given a set of $M$ observed spin configurations, assumed to be Boltzmann distributed.

Some applications of the inverse Ising problem:

- Modelling neural firing patterns in a biological network of cells
- Modelling interactions between species and individuals
- Combinatorial antibiotic treatment
- Protein structure determination
- Financial markets

# Theory behind the inverse Ising problem

Different methods of solving the inverse Ising problem:

- Mean-Field Theory
- Thouless-Anderson-Palmer (TAP) Reconstruction
- Bethe-Peierls approximation
- Maximum likelihood
- Maximum pseudolikelihood

# Theory behind the inverse Ising problem: Likelihood

## Maximum likelihood estimator

Given a set of observations $D = \{\boldsymbol{s}^1, \boldsymbol{s}^2, \ldots, \boldsymbol{s}^M\}$ from a statistical model $p(D \,|\, \boldsymbol{J}, \boldsymbol{h})$, where $\boldsymbol{J}, \boldsymbol{h}$ are unknown, the *maximum likelihood estimator* is defined as

$$(\widehat{\boldsymbol{J}}, \widehat{\boldsymbol{h}}) = \underset{\boldsymbol{J},\boldsymbol{h}}{\operatorname{argmax}}\; p(D \,|\, \boldsymbol{J}, \boldsymbol{h})$$

Applying the logarithm to the likelihood, assuming spin configurations were sampled independently from the Boltzmann distribution, given a set of observed spin configurations $D = \{\boldsymbol{s}^\mu\}$, $\mu = 1, \ldots, M$:

$$L_D(\boldsymbol{J}, \boldsymbol{h}) = \frac{1}{M} \ln p(D \,|\, \boldsymbol{J}, \boldsymbol{h}) = \frac{1}{M} \ln \left[ \prod_\mu \frac{1}{Z} \mathrm{e}^{-\beta H_{\boldsymbol{J},\boldsymbol{h}}(\boldsymbol{s}^\mu)} \right]$$

$$= \beta \sum_{i<j} J_{ij} \langle \sigma_i \sigma_j \rangle^D + \beta \sum_i h_i \langle \sigma_i \rangle^D - \ln Z(\boldsymbol{J}, \boldsymbol{h})$$

# Theory behind the inverse Ising problem: Likelihood

Differentiating the log-likelihood and setting the derivatives to zero:

$$\begin{aligned}
\beta^{-1}\frac{\partial L_D}{\partial h_i}(\boldsymbol{J},\boldsymbol{h}) &= \langle\sigma_i\rangle^D - \langle\sigma_i\rangle = 0 \\
\beta^{-1}\frac{\partial L_D}{\partial J_{ij}}(\boldsymbol{J},\boldsymbol{h}) &= \langle\sigma_i\sigma_j\rangle^D - \langle\sigma_i\sigma_j\rangle = 0
\end{aligned}$$

Boltzmann machine learning (ML):

$$\begin{aligned}
h_i^{n+1} &= h_i^n + \eta\frac{\partial L_D}{\partial h_i}(\boldsymbol{J}^n,\boldsymbol{h}^n) \\
J_{ij}^{n+1} &= J_{ij}^n + \eta\frac{\partial L_D}{\partial J_{ij}}(\boldsymbol{J}^n,\boldsymbol{h}^n)
\end{aligned}$$

This method is impractical for bigger systems: averages over $2^N$ spin configurations.

# Theory behind the inverse Ising problem: Pseudolikelihood

The pseudolikelihood is an approximation of the log-likelihood function that, when maximized, leads to the exact same parameters as the log-likelihood in the $M \to \infty$ limit.

- **Callen's identities:** $$\langle \sigma_i \rangle = \langle \tanh(\beta h_i + \beta \sum_{k \neq i} J_{ik}\sigma_k) \rangle$$ $$\langle \sigma_i \sigma_j \rangle = \langle \sigma_j \tanh(\beta h_i + \beta \sum_{k \neq i} J_{ik}\sigma_k) \rangle$$
- **Key assumption:** replacement of Boltzmann averages $\langle \cdot \rangle$ in Callen's identities by the sample means $\langle \cdot \rangle^D$.

Pseudolikelihood function for $\mu = 1, \ldots, M$ and $i = 1, \ldots, N$:

$$\begin{aligned} PL_D(\boldsymbol{J}, \boldsymbol{h}) &= \sum_i \frac{1}{M} \sum_\mu \ln p(s_i^\mu \,|\, \{s_j^\mu\}_{j \neq i}) \\ &= \frac{1}{M} \sum_{\mu, i} \ln \frac{1}{2} \left[ 1 + s_i^\mu \tanh\left( \beta h_i + \beta \sum_{j \neq i} J_{ij} s_j^\mu \right) \right]. \end{aligned}$$

# Methodology: Simulated annealing

- We chose simulated annealing (SA) in order to avoid local minima.
- We found the global minimum of $-PL_D$.

**Simulated annealing pseudocode:**

**Data:** *problem*, a problem
*schedule*, a function from time to the variable "temperature"
**Result:** A solution node
**Local variables:** *current*, a node
*next*, a node
$T$, a given "temperature" which regulates the probability

```
current ← getNode(initialState[problem])
for t ← 1 to ∞ do
    T ← schedule[t]
    if T = 0 then return current
    next ← a randomly selected successor of current
    ΔE ← value[current] - value[next]
    if ΔE > 0 then current ← next
    else current ← next with probability e^{ΔE/T}
end
```

# Methodology: Simulated annealing

Considerations:

- Solution space: $h_i \in [-1, 1]$, $J_{ij} \in [-1, 1]$.
- Since $PL_D$ is a function of order 1, $T_0 = 2$ was chosen.
- The temperature dependency is $T(k) = \frac{T_0}{k}$, where $k$ is the current elementary move number.

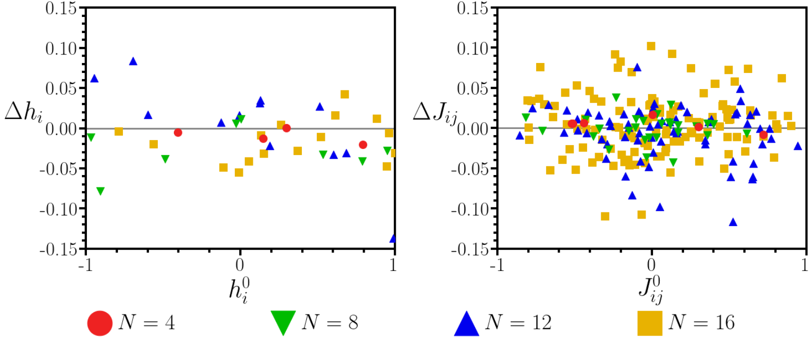
# Results: Sherrington-Kirkpatrick model

## Sherrington-Kirkpatrick model

Ising model where each spin is coupled to every other spin in the system (infinite-range interactions).

- All configurations obtained and $M$ sampled from Boltzmann distribution.
- Random pairwise couplings and field.
- Only small systems ($N \in \{4, 8, 12, 16\}$).
- $M = 15000$ spin configurations, $\tau = 500$ MCS.
- Each MCS is $R = N + \frac{N(N-1)}{2}$ elementary moves.
- Relative squared error: $\gamma = \sqrt{\frac{\sum_i (h_i^0 - h_i)^2 + \sum_{i<j} (J_{ij}^0 - J_{ij})^2}{\sum_i (h_i^0)^2 + \sum_{i<j} (J_{ij}^0)^2}}$

# Results: Sherrington-Kirkpatrick model




Relative errors:

| $N$ (spins) | 4 | 8 | 12 | 16 |
|---|---|---|---|---|
| $\gamma$ | 0.022 | 0.054 | 0.085 | 0.083 |

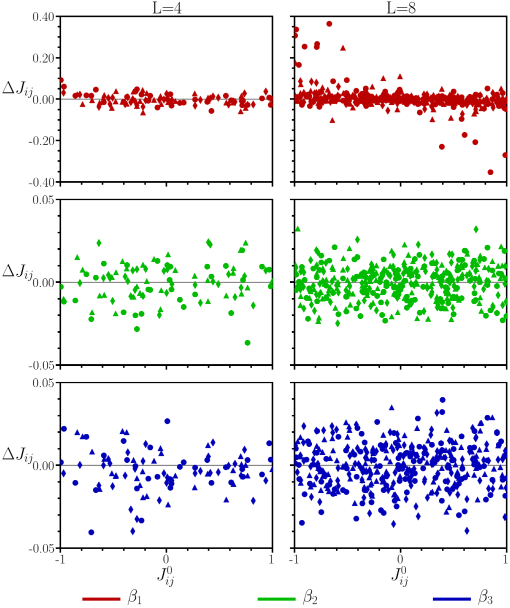
# Results: Short-range interaction model

## 2D short-range interaction model

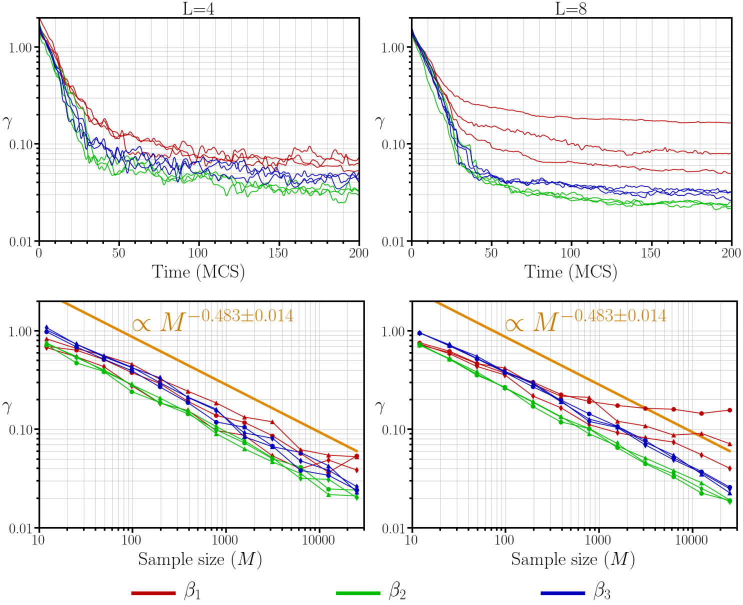
2D Ising model where each spin is only coupled to its nearest neighbors.

- All data was given by the advisor.
- Three different samples for three different values of $\beta$: $\beta_1 = 2.0$, $\beta_2 = 0.92$, $\beta_3 = 0.5$.
- Two system sizes: $N = 16$ and $N = 64$.
- $h_i = 0 \; \forall i \Rightarrow \gamma = \sqrt{\frac{\sum_{i<j}(J_{ij}^0 - J_{ij})^2}{\sum_{i<j}(J_{ij}^0)^2}}$.
- Toroidal boundary conditions $\Rightarrow \#J_{ij} = 2N = R$.
- $M = 25000$ spin configurations for each sample and value of $\beta$, and $\tau = 200$ MCS.

# Results: Short-range interaction model



# Results: Short-range interaction model



# Results: Short-range interaction model

Recall the pseudolikelihood function:

- SK model: $PL_D(\boldsymbol{J},\boldsymbol{h}) = \sum_i \frac{1}{M} \sum_\mu \ln \frac{1}{2}[1 + s_i^\mu \tanh(\beta h_i + \beta \sum_{j \neq i} J_{ij} s_j^\mu)]$, $\mathcal{O}(MN^2)$
- Short-range: $PL_D(\boldsymbol{J},\boldsymbol{h}) = \sum_i \frac{1}{M} \sum_\mu \ln \frac{1}{2}[1 + s_i^\mu \tanh(\beta h_i + \beta \sum_{\langle i,j \rangle} J_{ij} s_j^\mu)]$, $\mathcal{O}(MN)$

Denoting a class like $\omega_i^\alpha(\boldsymbol{J})$, and the frequency of that class like $C_i^\alpha$, for a short-range interaction model with no field we have:

$$PL_D(\boldsymbol{J},\boldsymbol{h}) = \frac{1}{M} \sum_i \sum_{\alpha=1}^{32} C_i^\alpha \ln \frac{1}{2}[1 + \omega_i^\alpha(\boldsymbol{J})], \text{ with } \mathcal{O}(N)$$

# Results: Short-range interaction model

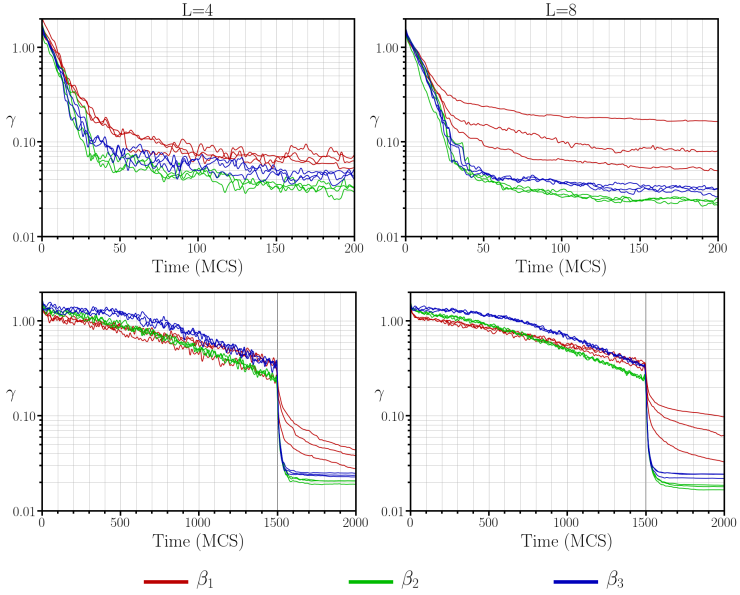
Using this faster way to computing the pseudolikelihood, we have done more simulations:

- We have done five runs of each sample with $\tau = 2000$ MCS.
- The temperature schedule for the SA is $T(t) = T_0 \cdot \mathrm{e}^{-5\frac{t}{\tau}}$, slower than before.
- The last 25% of MCS have been performed at $T = 0$.

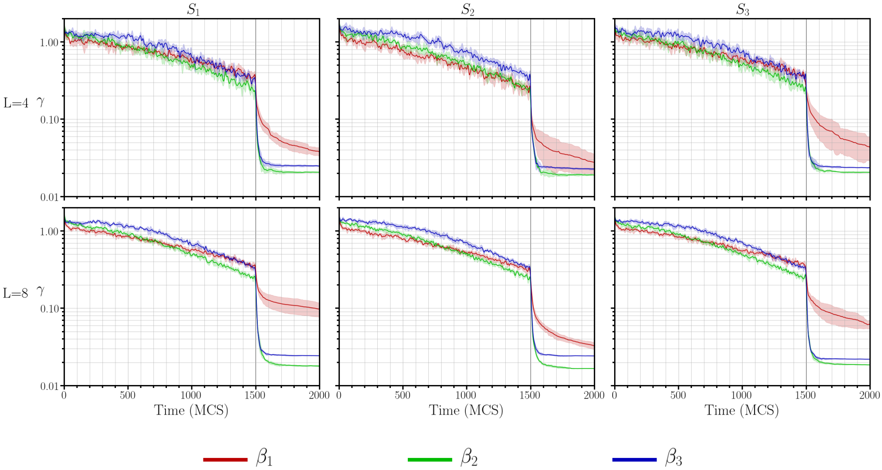
# Results: Short-range interaction model

$T(k) = \frac{T_0}{k}$

$T(t) = T_0 \mathrm{e}^{-5\frac{t}{\tau}}$

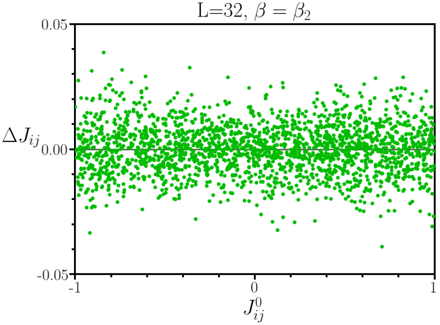


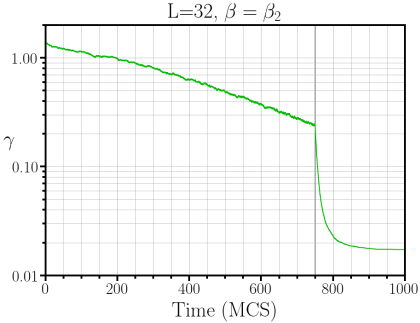
# Results: Short-range interaction model



# Results: Short-range interaction model

Bigger simulation: $L = 32 \Rightarrow N = 32^2 = 1024$, $M = 25000$, $\beta = \beta_2 = 0.92$, exponential temperature dependency in SA.




Maximum offset: less than 0.04. Final error: 0.017.

# Results: Short-range interaction model

Applying the class technique to the jacobian of the pseudolikelihood, we have been able to maximize $PL_D$ using gradient descent, a convex optimization algorithm. The final results are shown in this table:

| | $\gamma(\beta_1 = 2)$ | $\gamma(\beta_2 = 0.92)$ | $\gamma(\beta_3 = 0.5)$ |
|---|---|---|---|
| $L = 4$ ($M = 25000$), SA | 0.036 | 0.020 | 0.024 |
| $L = 4$ ($M = 25000$), GD | 0.026 | 0.020 | 0.024 |
| $L = 8$ ($M = 25000$), SA | 0.064 | 0.018 | 0.023 |
| $L = 8$ ($M = 25000$), GD | 0.060 | 0.018 | 0.024 |
| $L = 8$ ($M = 15000$), SA | 0.097 | 0.025 | 0.036 |
| $L = 8$ (Nguyen *et al.*) | $>1.0$ | 0.135 | 0.158 |

# Conclusions

- Small improvement with exponential schedule in SA.
- Much bigger system sizes could be solved using this new way to computing the $PL_D$.
- Small differences between gradient descent and simulated annealing.
- For non-convex problems, better SA.

# Thank you!

Thank you for your attention!