Exact Inference, Mean-Field Approximation, and Gibbs Sampling on a 10×10 Ising Grid

Q1. Exact inference via induced factor-graph (column variables)

1. Model Specification

Model. Consider the 10×10 lattice with nodes $x_{i,j} \in \{-1, +1\}$, where $i \in \{1, \dots, 10\}$ is the row index and $j \in \{1, \dots, 10\}$ is the column index. For each neighbouring pair (edge) $e = (u, v) \in E$ on the lattice, the pairwise potential is

$$\phi(x_u, x_v) = \exp\big(\beta\, \mathbb{I}[x_u = x_v]\big).$$

The joint distribution is

$$P(x) = \frac{1}{Z(\beta)} \prod_{\substack{e=(u,v)\in E \\ u>v}} \phi(x_u, x_v),$$

where the condition $u > v$ is only to avoid double counting each undirected edge.

2. Transformation to Linear Chain Factor Graph

Step 1: Create giant column variables. Group each column into one variable:

$$Y_j := (x_{1,j}, x_{2,j}, \dots, x_{10,j}) \in \{-1, +1\}^{10}, \qquad j = 1, \dots, 10.$$

Thus each Y_j has 2^{10} possible states.

Step 2: Split edges and regroup factors. Split the edge set into vertical and horizontal edges:

$$E = E_{\text{vert}} \cup E_{\text{horiz}}, \qquad E_{\text{vert}} \cap E_{\text{horiz}} = \emptyset,$$

so that

$$\prod_{e\in E} \phi(x_e) = \Big(\prod_{e\in E_{\text{vert}}} \phi(x_e) \Big) \Big(\prod_{e\in E_{\text{horiz}}} \phi(x_e) \Big).$$

(a) Vertical edges $\Rightarrow$ within-column potentials. Write the vertical-edge product explicitly:

$$\prod_{e\in E_{\text{vert}}} \phi(x_e) = \prod_{j=1}^{10} \prod_{i=1}^{9} \phi(x_{i,j}, x_{i+1,j}).$$

For a fixed column j, the inner product depends only on variables in that column, i.e. only on Y_j. Define the within-column potential

$$\Phi(Y_j) := \prod_{i=1}^{9} \phi(x_{i,j}, x_{i+1,j}).$$

Hence

$$\prod_{e \in E_{\text{vert}}} \phi(x_e) = \prod_{j=1}^{10} \Phi(Y_j).$$

(b) Horizontal edges ⇒ between-column potentials. Write the horizontal-edge product explicitly:

$$\prod_{e \in E_{\text{horiz}}} \phi(x_e) = \prod_{j=1}^{9} \prod_{i=1}^{10} \phi(x_{i,j}, x_{i,j+1}).$$

For a fixed adjacent pair $(j, j+1)$, the inner product depends only on variables in columns j and $j+1$, i.e. only on (Y_j, Y_{j+1}). Define the between-column potential

$$\Psi(Y_j, Y_{j+1}) := \prod_{i=1}^{10} \phi(x_{i,j}, x_{i,j+1}).$$

Hence

$$\prod_{e \in E_{\text{horiz}}} \phi(x_e) = \prod_{j=1}^{9} \Psi(Y_j, Y_{j+1}).$$

Induced factorisation (column-chain). Combining the regroupings, we obtain the induced factorisation over $(Y_1, \ldots, Y_{10})$:

$$P(Y_1, \ldots, Y_{10}) \propto \Big[\prod_{j=1}^{10} \Phi(Y_j) \Big] \Big[\prod_{j=1}^{9} \Psi(Y_j, Y_{j+1}) \Big].$$

This corresponds to a chain factor graph

$$Y_1 \; - \; Y_2 \; - \; \cdots \; - \; Y_{10},$$

with unary factors $\Phi(Y_j)$ and pairwise factors $\Psi(Y_j, Y_{j+1})$.

3. Message Passing Algorithm

Step 3/4: Message passing (start with m, then define α). Let $m_{j \to j+1}(y_{j+1})$ be the forward message from Y_j to Y_{j+1}.

Initial step:

$$m_{1 \to 2}(y_2) = \sum_{y_1} \Phi(y_1) \Psi(y_1, y_2).$$

In the middle ($j = 2, \ldots, 9$):

$$m_{j \to j+1}(y_{j+1}) = \sum_{y_j} \Phi(y_j) \Psi(y_j, y_{j+1}) \, m_{j-1 \to j}(y_j).$$

Now define the "special" quantity

$$\alpha_j(y_j) := \Phi(y_j) \, m_{j-1 \to j}(y_j),$$

with $\alpha_1(y_1) = \Phi(y_1)$ (equivalently, $m_{0\to1}(y_1) = 1$). Then the recursion can be written as

$$m_{j\to j+1}(y_{j+1}) = \sum_{y_j} \Psi(y_j, y_{j+1})\, \alpha_j(y_j),$$

and

$$\alpha_{j+1}(y_{j+1}) = \Phi(y_{j+1})\, m_{j\to j+1}(y_{j+1}) = \Phi(y_{j+1}) \sum_{y_j} \Psi(y_j, y_{j+1})\, \alpha_j(y_j).$$

At the end of the chain, there is no message from the right, hence

$$P(Y_{10} = y_{10}) \propto \Phi(y_{10})\, m_{9\to10}(y_{10}) = \alpha_{10}(y_{10}),$$

so

$$P(Y_{10} = y_{10}) = \frac{\alpha_{10}(y_{10})}{\sum_{y'_{10}} \alpha_{10}(y'_{10})}.$$

4. Calculation of Marginals and Results

Step 5: Extract $P(x_{1,10}, x_{10,10})$. A state of Y_{10} is $y_{10} = (y_{10}[1], \ldots, y_{10}[10])$ where $y_{10}[1] = x_{1,10}$ and $y_{10}[10] = x_{10,10}$. Thus for $a, b \in \{-1, +1\}$,

$$P(x_{1,10} = a,\ x_{10,10} = b) = \sum_{\substack{y_{10}\in\{-1,+1\}^{10}:\\ y_{10}[1]=a,\ y_{10}[10]=b}} \frac{\alpha_{10}(y_{10})}{\sum_{y'_{10}} \alpha_{10}(y'_{10})}.$$

Evaluating this for $(a, b) \in \{-1, +1\}^2$ yields the required 2×2 probability table for $P(x_{1,10}, x_{10,10})$.

Exact numerical results (rounded to 4 d.p.).

$$\begin{pmatrix} P(-1,-1) & P(-1,+1) \\ P(+1,-1) & P(+1,+1) \end{pmatrix}$$

$$\beta = 4: \quad \begin{pmatrix} 0.4997 & 0.0003 \\ 0.0003 & 0.4997 \end{pmatrix} \qquad \beta = 1: \quad \begin{pmatrix} 0.2804 & 0.2196 \\ 0.2196 & 0.2804 \end{pmatrix} \qquad \beta = 0.01: \quad \begin{pmatrix} 0.2500 & 0.2500 \\ 0.2500 & 0.2500 \end{pmatrix}$$

5. Physical Interpretation of β

What β means in this model. Your Ising model is defined by pairwise potentials

$$\phi(x_u, x_v) = \exp\big(\beta\, \mathbb{I}[x_u = x_v]\big),$$

so the unnormalised joint distribution can be written as

$$P(x) \propto \exp\Big(\beta \sum_{(u,v)\in E} \mathbb{I}[x_u = x_v]\Big).$$

Define the alignment score

$$S(x) := \sum_{(u,v)\in E} \mathbb{I}[x_u = x_v].$$

Then

$$P_\beta(x) = \frac{1}{Z(\beta)} \exp\big(\beta S(x)\big), \qquad Z(\beta) = \sum_x \exp\big(\beta S(x)\big).$$

A useful identity is

$$\frac{d}{d\beta} \log Z(\beta) = \mathbb{E}_{P_\beta}[S(x)],$$

so changing β changes the expected number of agreeing neighbour edges.

A direct local interpretation comes from a single edge (u, v):

$$\phi(x_u, x_v) = \begin{cases} e^\beta, & x_u = x_v, \\ 1, & x_u \neq x_v. \end{cases}$$

So a match across one edge is weighted e^β times more than a mismatch.

$\beta = 4$**: Strong coupling.** Here $e^\beta = e^4 \approx 54.6$, so mismatches are heavily penalised and the table is almost entirely on $(-1, -1)$ and $(+1, +1)$.

$\beta = 1$**: Moderate coupling.** Here $e^\beta = e^1 \approx 2.72$, so agreement is preferred but not overwhelmingly.

$\beta = 0.01$**: Weak coupling.** Here $e^\beta \approx 1.01005$, close to 1, so the distribution is close to uniform.

Q2. Mean Field Approximation (Coordinate Ascent)

1. Theory and Factorisation

1) Variational idea + mean-field factorisation. We approximate the true distribution $P(x)$ with a simpler distribution $Q(x)$ and measure closeness using

$$\mathrm{KL}(Q\|P).$$

Under the **mean-field assumption** we assume independence between nodes:

$$Q(x) = \prod_{u\in V} q_u(x_u),$$

(where u indexes a lattice site, e.g. $u = (i, j)$).

A standard coordinate-ascent result (from ELBO maximisation) is:

$$\boxed{\ln q_u^*(x_u) = \mathbb{E}_{Q_{-u}}\big[\ln P(x)\big] + \text{constant}}$$

where $Q_{-u} = \prod_{v\neq u} q_v(x_v)$ is the product of all factors except node u.

2) Rewrite the pairwise indicator ("better math" step). The model is

$$P(x) \propto \prod_{(u,v)\in E} \exp\big(\beta\,\mathbf{1}[x_u = x_v]\big) = \exp\Big(\beta \sum_{(u,v)\in E} \mathbf{1}[x_u = x_v]\Big).$$

(**Important:** the sum is over neighbour **edges** $(u, v) \in E$, not all pairs.)

To use the indicator identity, we represent the binary states as spins

$$x_u \in \{-1, +1\}.$$

Then

$$\mathbf{1}[x_u = x_v] = \frac{1 + x_u x_v}{2} \quad (\text{since } x_u x_v = +1 \iff x_u = x_v,\ \ x_u x_v = -1 \iff x_u \neq x_v).$$

Substitute:

$$\begin{aligned} P(x) &\propto \exp\Big(\beta \sum_{(u,v)\in E} \frac{1 + x_u x_v}{2}\Big) \\ &= \exp\Big(\frac{\beta}{2} \sum_{(u,v)\in E} 1\Big)\ \exp\Big(\frac{\beta}{2} \sum_{(u,v)\in E} x_u x_v\Big). \end{aligned}$$

The first exponential is a constant (it does not depend on x), so we absorb it into the normaliser:

$$\boxed{P(x) \propto \exp\Big(\frac{\beta}{2} \sum_{(u,v)\in E} x_u x_v\Big)}$$

and therefore

$$\boxed{\ln P(x) = \frac{\beta}{2} \sum_{(u,v)\in E} x_u x_v + C}$$

for some constant C.

2. Derivation of Update Equations

3) Isolate the terms involving a single node x_u. Fix a node u. In the sum $\sum_{(a,b)\in E} x_a x_b$, the only terms that contain x_u are the edges (u, v) where $v \in N(u)$ (the neighbours of u). So we can rewrite:

$$\ln P(x) = \frac{\beta}{2}\Big(x_u \sum_{v\in N(u)} x_v\Big) + \underbrace{(\text{terms not involving } x_u)}_{\text{no } x_u} + C.$$

4) Plug into the coordinate ascent formula. Using

$$\ln q_u^*(x_u) = \mathbb{E}_{Q_{-u}}[\ln P(x)] + \text{const},$$

we get

$$\ln q_u^*(x_u) = \frac{\beta}{2}\, x_u \sum_{v\in N(u)} \mathbb{E}_{Q_{-u}}[x_v] + \mathbb{E}_{Q_{-u}}[\text{terms not involving } x_u] + \mathbb{E}_{Q_{-u}}[C] + \text{const}.$$

Why are the last two expectations "constant"?

- $\mathbb{E}_{Q_{-u}}[C] = C$, since C is a number.
- $\mathbb{E}_{Q_{-u}}[\text{terms not involving } x_u]$ may depend on Q_{-u}, but it does **not** depend on x_u. During the update of $q_u(x_u)$, Q_{-u} is fixed, so this becomes a scalar constant **with respect to** x_u and can be absorbed into the normalisation of q_u.

So the only part that matters as a function of x_u is:

$$\ln q_u^*(x_u) = \frac{\beta}{2}\, x_u \sum_{v\in N(u)} \mathbb{E}_{Q_{-u}}[x_v] + \text{constant}.$$

5) Why is $\mathbb{E}_{Q_{-u}}[x_v] = m_v$? Define the mean ("magnetisation") of node v under mean-field:

$$\boxed{m_v := \mathbb{E}_Q[x_v] = \sum_{x_v\in\{-1,+1\}} x_v\, q_v(x_v)}$$

Now for $v \neq u$,

$$\mathbb{E}_{Q_{-u}}[x_v] = \sum_{x_{-u}} x_v \prod_{k\neq u} q_k(x_k).$$

Split the sum into x_v and all other variables $x_{k:k\neq u,v}$:

$$= \sum_{x_v} \sum_{x_{k:k\neq u,v}} x_v\; q_v(x_v) \prod_{k\neq u,v} q_k(x_k).$$

The inner sum equals 1 because each q_k is a probability distribution:

$$\sum_{x_{k:k\neq u,v}} \prod_{k\neq u,v} q_k(x_k) = \prod_{k\neq u,v} \sum_{x_k} q_k(x_k) = \prod_{k\neq u,v} 1 = 1.$$

So

$$\mathbb{E}_{Q_{-u}}[x_v] = \sum_{x_v} x_v\, q_v(x_v) = m_v.$$

6) Final form of q_u and the mean-field update. Substitute $\mathbb{E}_{Q_{-u}}[x_v] = m_v$:

$$\ln q_u^*(x_u) = \frac{\beta}{2}\, x_u \sum_{v \in N(u)} m_v + \text{constant}.$$

Define the local field

$$\boxed{h_u := \frac{\beta}{2} \sum_{v \in N(u)} m_v}$$

then

$$q_u(x_u) \propto \exp(x_u h_u).$$

Since $x_u \in \{-1, +1\}$,

$$q_u(+1) = \frac{e^{h_u}}{e^{h_u} + e^{-h_u}}, \qquad q_u(-1) = \frac{e^{-h_u}}{e^{h_u} + e^{-h_u}}.$$

The mean at node u is

$$m_u = \mathbb{E}[x_u] = (+1)q_u(+1) + (-1)q_u(-1) = \frac{e^{h_u} - e^{-h_u}}{e^{h_u} + e^{-h_u}} = \boxed{\tanh(h_u)}.$$

So the coordinate ascent fixed-point update is:

$$\boxed{m_u \leftarrow \tanh\Big(\frac{\beta}{2} \sum_{v \in N(u)} m_v\Big)}$$

Also,

$$m_u = q_u(+1) - q_u(-1) = 2q_u(+1) - 1 \;\Rightarrow\; \boxed{q_u(+1) = \frac{1 + m_u}{2}, \quad q_u(-1) = \frac{1 - m_u}{2}}$$

Using this to produce the required joint table. After iterating the updates until convergence, we get the two marginals $q_{1,10}$ and $q_{10,10}$. Mean-field assumes independence, so:

$$\boxed{Q(x_{1,10} = a,\ x_{10,10} = b) = q_{1,10}(a)\ q_{10,10}(b)} \qquad (a, b \in \{-1, +1\}).$$

3. Mean-field coordinate ascent (pseudocode) and convergence

Pseudocode.

```
Inputs: grid size 10x10, coupling beta, tolerance eps, max iterations T.
Outputs: node means m[r,c] and marginals q[r,c](+1), q[r,c](-1).

Initialize m[r,c] for all nodes (r,c)
    e.g. m[r,c] = 0.01  (small positive bias to break symmetry)
    or random small values in [-0.01, 0.01]

For t = 1 to T:
    max_change = 0
    For each node (r,c) in the 10x10 grid:
```

```
        N = neighbours of (r,c) (up/down/left/right within grid)
        h = (beta/2) * sum_{(r',c') in N} m[r',c']
        new_m = tanh(h)
        max_change = max(max_change, |new_m - m[r,c]|)
        m[r,c] = new_m
    If max_change < eps:
        break

For each node (r,c):
    q[r,c](+1) = (1 + m[r,c]) / 2
    q[r,c](-1) = (1 - m[r,c]) / 2

Return m, q
```

Stopping criterion and observed iterations. We iterated coordinate-ascent mean-field updates until the maximum change in magnetisations over a full sweep satisfied

$$\max_u \left| m_u^{(t)} - m_u^{(t-1)} \right| < \varepsilon.$$

Convergence was achieved in 6, 21, and 9 sweeps for $\beta = 4, 1, 0.01$ respectively (for initialisation $m^{(0)} \equiv +0.01$).

4. Results for $Q(x_{1,10}, x_{10,10})$

Initialisation $m^{(0)} \equiv +0.01$ ("positive phase").

$$\begin{pmatrix} Q(-1,-1) & Q(-1,+1) \\ Q(+1,-1) & Q(+1,+1) \end{pmatrix}$$

$$\beta = 4: \begin{pmatrix} 1.1248 \times 10^{-7} & 0.0003352707 \\ 0.0003352707 & 0.9993293461 \end{pmatrix} \qquad \beta = 1: \begin{pmatrix} 0.0238945142 & 0.1306839908 \\ 0.1306839908 & 0.7147375042 \end{pmatrix} \qquad \beta = 0.01: \begin{pmatrix} 0.25 & 0.25 \\ 0.25 & 0.25 \end{pmatrix}$$

Initialisation $m^{(0)} \equiv -0.01$ ("negative phase").

$$\beta = 4: \begin{pmatrix} 0.9993293461 & 0.0003352707 \\ 0.0003352707 & 1.1248 \times 10^{-7} \end{pmatrix} \qquad \beta = 1: \begin{pmatrix} 0.7147375042 & 0.1306839908 \\ 0.1306839908 & 0.0238945142 \end{pmatrix} \qquad \beta = 0.01: \begin{pmatrix} 0.25 & 0.25 \\ 0.25 & 0.25 \end{pmatrix}$$

Initialisation $m^{(0)} \equiv 0$ (symmetric fixed point). If we initialise all $m_u^{(0)} = 0$, then $h_u = \frac{\beta}{2} \sum_{v \in N(u)} 0 = 0$ for every node, hence $m_u^{(1)} = \tanh(0) = 0$. Therefore $m \equiv 0$ is a fixed point, which implies $q_u(+1) = q_u(-1) = \frac{1}{2}$ and thus $Q(x_{1,10}, x_{10,10})$ is uniform with all four entries equal to 0.25.

5. Interpretation and comparison to Q1

How β affects q_u and m_u (mathematically). From the derivation,

$$h_u = \frac{\beta}{2} \sum_{v \in N(u)} m_v, \qquad q_u(x_u) \propto \exp(x_u h_u).$$

A direct "preference strength" statement comes from the odds ratio:

$$\frac{q_u(+1)}{q_u(-1)} = \frac{e^{h_u}}{e^{-h_u}} = e^{2h_u} = \exp\Big(\beta \sum_{v \in N(u)} m_v\Big).$$

So if neighbours tend to be $+1$ (i.e. $\sum_{v \in N(u)} m_v > 0$), then $h_u > 0$ and $q_u(+1) > q_u(-1)$. Increasing β amplifies this preference exponentially, which mathematically explains why larger β enforces stronger local similarity.

Also,

$$m_u = \tanh(h_u) = \tanh\Big(\frac{\beta}{2} \sum_{v \in N(u)} m_v\Big).$$

For small β, h_u is small and $\tanh(h_u) \approx h_u$, so the means stay near 0 (weak ordering). For large β, $|h_u|$ becomes larger and $\tanh(h_u) \to \pm 1$, so the means saturate toward ± 1 (strong ordering), matching the observed results.

Why changing the sign of init flips the result. Because there are no unary terms, the model has global flip symmetry ($x \mapsto -x$). The mean-field updates preserve sign through $\tanh(\cdot)$, so a small positive initialisation converges to a "mostly $+1$" fixed point, while a small negative initialisation converges to the symmetric "mostly -1" fixed point.

Link back to Q1 (exact result). Q1 exact inference respects the global flip symmetry ($x \mapsto -x$) because the model has no unary terms. This explains the relationship between the exact tables in Q1 and the mean-field tables in Q2 across all three values of β:

- $\beta = 4$ **(strong coupling):** the exact distribution behaves like a near 50–50 mixture of two modes (mostly $+1$ and mostly -1), giving approximately equal mass on $(-1,-1)$ and $(+1,+1)$ in Q1. Mean-field cannot represent a mixture with a single factorised $Q(x) = \prod_u q_u(x_u)$, so it converges to one mode or the other depending on the sign of the initialisation, producing a "single-phase" table with one diagonal entry near 1.
- $\beta = 1$ **(moderate coupling):** the exact table still prefers agreement (diagonals larger than off-diagonals), but not as extremely as $\beta = 4$. Mean-field still converges to one of two symmetric phases under biased initialisation, and its (symmetrised) estimate can differ from the exact table because the factorised family cannot capture long-range dependencies and multi-modal behaviour.
- $\beta = 0.01$ **(very weak coupling):** interactions are so small that the exact distribution is close to uniform in Q1, giving approximately 0.25 in all four entries. Mean-field also converges to the symmetric fixed point with $m_u \approx 0$ (or stays there if initialised at 0), giving $q_u(+1) \approx q_u(-1) \approx \frac{1}{2}$ and therefore a uniform joint table with all entries 0.25.

Fair comparison to Q1 (symmetrised mean-field). To match Q1's symmetry, we report the symmetrised mean-field:

$$\tilde{Q} = \frac{1}{2}\Big(Q^{(+\text{init})} + Q^{(-\text{init})}\Big).$$

For $\beta = 4$, this gives approximately

$$\tilde{Q} \approx \begin{pmatrix} 0.4997 & 0.0003 \\ 0.0003 & 0.4997 \end{pmatrix},$$

which is very close to the exact Q1 table. For $\beta = 1$, we get

$$\tilde{Q} \approx \begin{pmatrix} 0.3693 & 0.1307 \\ 0.1307 & 0.3693 \end{pmatrix},$$

showing mean-field can overestimate ordering/correlation compared to the exact result (Q1 diagonals ≈ 0.2804). Finally, for $\beta = 0.01$ both exact inference and mean-field are essentially uniform:

$$\tilde{Q} \approx \begin{pmatrix} 0.25 & 0.25 \\ 0.25 & 0.25 \end{pmatrix},$$

matching Q1.

Q3.3 Gibbs Sampling

1. Theory and Derivation

Spin space. We represent each lattice node as a spin $x_{ij} \in \{-1, +1\}$ on a 10×10 grid.

1) Gibbs transition idea. A single-site Gibbs step updates one node $x_{i,j}$ by sampling from

$$P\big(x_{i,j} \mid x_{-(i,j)}\big).$$

Everything else stays fixed.

Formally, if only site (i, j) can change, then the transition from configuration x to x' is:

$$T(x \to x') = \begin{cases} P\big(x'_{i,j} \mid x_{-(i,j)}\big) & \text{if } x'_{-(i,j)} = x_{-(i,j)}, \\ 0 & \text{otherwise.} \end{cases}$$

2) Gibbs sampling update for one site (i, j). Let x be a full configuration on the 10×10 grid, and let $x_{\setminus ij}$ denote all spins except x_{ij}. A single-site Gibbs step (updating site (i, j)) samples

$$x_{ij}^{\text{new}} \sim P(x_{ij} \mid x_{\setminus ij}).$$

Step 1: Write the conditional.

$$P(x_{ij} \mid x_{\setminus ij}) = \frac{P(x_{ij}, x_{\setminus ij})}{P(x_{\setminus ij})}.$$

Key point (the "constant" part): the denominator $P(x_{\setminus ij})$ does not depend on the value of x_{ij}, so as a function of x_{ij},

$$P(x_{ij} \mid x_{\setminus ij}) \propto P(x_{ij}, x_{\setminus ij}).$$

Step 2: Use the model factorisation and cancel constants. The Ising model here is

$$P(x) = \frac{1}{Z} \prod_{(u,v)\in E} \phi(x_u, x_v) = \frac{1}{Z} \prod_{(u,v)\in E} \exp\big(\beta\, \mathbf{1}[x_u = x_v]\big).$$

Split the product into:

- factors on edges that touch (i, j) (i.e. edges $((i, j), k)$ with $k \in N(i, j)$),
- and factors on all other edges.

All edges not involving (i, j) do not change when you flip x_{ij}, so they are constants w.r.t. x_{ij} and cancel when you normalise the conditional. Therefore:

$$P(x_{ij} \mid x_{\setminus ij}) \propto \prod_{k\in N(i,j)} \phi(x_{ij}, x_k) = \prod_{k\in N(i,j)} \exp\big(\beta\, \mathbf{1}[x_{ij} = x_k]\big).$$

So:

$$P(x_{ij} = s \mid x_{\setminus ij}) \propto \exp\Big(\beta \sum_{k\in N(i,j)} \mathbf{1}[s = x_k]\Big), \qquad s \in \{-1, +1\}.$$

Step 3: Turn it into a simple "neighbour count" formula. Let

- $d = |N(i,j)|$ be the number of neighbours (2 corner, 3 edge, 4 interior),
- $n_+ = \#\{k \in N(i,j) : x_k = +1\}$,
- $n_- = d - n_+$.

Then:

$$P(x_{ij} = +1 \mid \text{rest}) \propto e^{\beta n_+}, \qquad P(x_{ij} = -1 \mid \text{rest}) \propto e^{\beta n_-}.$$

Normalise:

$$P(x_{ij} = +1 \mid \text{rest}) = \frac{e^{\beta n_+}}{e^{\beta n_+} + e^{\beta(d-n_+)}} = \frac{1}{1 + \exp\big(\beta(d - 2n_+)\big)}, \qquad P(x_{ij} = -1 \mid \text{rest}) = 1 - P(x_{ij} = +1 \mid \text{rest}).$$

Equivalent (often nicer) "neighbour-sum" form. Let

$$S = \sum_{k \in N(i,j)} x_k$$

(sum of neighbour spins, each ± 1). Then $S = 2n_+ - d$, and

$$P(x_{ij} = +1 \mid \text{rest}) = \frac{1}{1 + e^{-\beta S}}, \qquad P(x_{ij} = -1 \mid \text{rest}) = \frac{1}{1 + e^{+\beta S}}.$$

2. Methodology

3) How this produces the required answer $P(x_{1,10}, x_{10,10})$.

Burn-in and thinning. We use a systematic-scan Gibbs sampler: one sweep updates every lattice site once using the single-site conditional. The first B sweeps are treated as burn-in and discarded so that the chain can approach its stationary (target) distribution and reduce dependence on the initial configuration. After burn-in, we record the pair $(x_{1,10}, x_{10,10})$ every τ sweeps (thinning) to reduce autocorrelation between successive recorded states.

For each $\beta \in \{4, 1, 0.01\}$:

- run the Gibbs chain for many sweeps using the update above,
- burn-in: discard the first B sweeps,
- collect samples after burn-in, recording $(x_{1,10}, x_{10,10})$ once every τ sweeps.

For each recorded sample t, define

$$(a_t, b_t) = \big(x_{1,10}^{(t)},\, x_{10,10}^{(t)}\big) \in \{-1, +1\}^2.$$

Estimate the joint by normalised counts:

$$\widehat{P}(a,b) = \frac{1}{T} \sum_{t=1}^{T} \mathbf{1}[a_t = a,\ b_t = b], \qquad a, b \in \{-1, +1\},$$

where T is the total number of recorded (kept) samples. This produces the required 2×2 probability table for $P(x_{1,10}, x_{10,10})$.

Pseudocode.

Input: $\beta,$ burn-in $B,$ sampling sweeps $S,$ thinning $\tau,$ initial state $x \in \{-1,+1\}^{10\times 10}$.

Index map: $\mathrm{idx}(-1) = 1,\ \mathrm{idx}(+1) = 2.$

Neighbour set: $N(r,c) = \{(r',c') :\ (r',c')$ is up/down/left/right of (r,c) and inside grid$\}$.

Initialise: $\texttt{counts}[2,2] \leftarrow 0,\ \texttt{kept} \leftarrow 0.$

For $s = 1,\ldots,B+S$:

For each site (r,c) (fixed scan or random order):

$$H_{r,c} \leftarrow \sum_{(r',c')\in N(r,c)} x_{r',c'} \quad \text{(sum of neighbour spins)}$$

$$p \leftarrow P(x_{r,c} = +1 \mid x_{-(r,c)}) \ = \ \frac{1}{1+\exp(-\beta H_{r,c})}$$

$$x_{r,c} \leftarrow \begin{cases} +1, & \text{with probability } p, \\ -1, & \text{with probability } 1-p \end{cases}$$

If $s > B$ **and** $(s-B) \bmod \tau = 0$:

$a \leftarrow x_{1,10},\quad b \leftarrow x_{10,10}$

$\texttt{counts}[\mathrm{idx}(a),\mathrm{idx}(b)] \leftarrow \texttt{counts}[\mathrm{idx}(a),\mathrm{idx}(b)] + 1, \quad \texttt{kept} \leftarrow \texttt{kept} + 1$

Return: $\widehat{P}(x_{1,10},x_{10,10}) = \texttt{counts}/\texttt{kept}$ (elementwise).

3. Results for $\widehat{P}(x_{1,10},x_{10,10})$ from Gibbs sampling

We estimate the joint probability table in the layout: rows correspond to $x_{1,10} \in \{-1,+1\}$ and columns correspond to $x_{10,10} \in \{-1,+1\}$, so the table entries are $P(-1,-1)$, $P(-1,+1)$, $P(+1,-1)$, $P(+1,+1)$.

Using $T = 8000$ kept samples after burn-in and thinning, we obtained:

$$\beta = 0.01 :\ \begin{pmatrix} 0.246375 & 0.252375 \\ 0.253125 & 0.248125 \end{pmatrix} \qquad \beta = 1 :\ \begin{pmatrix} 0.276375 & 0.219250 \\ 0.218875 & 0.285500 \end{pmatrix}$$

$$\beta = 4 \text{ (raw single run)} :\ \begin{pmatrix} 0.999250 & 0.000500 \\ 0.000250 & 0.000000 \end{pmatrix}.$$

Because the $\beta = 4$ run can get stuck in one magnetised mode (explained below), we also report the symmetrised estimate:

$$\beta = 4 \text{ (symmetrised)} :\ \begin{pmatrix} 0.499625 & 0.000375 \\ 0.000375 & 0.499625 \end{pmatrix}.$$

4. Interpretation by β

How β affects the Gibbs updates (mathematically). Compute the neighbour sum $S = \sum_{k\in N(i,j)} x_k$ (each neighbour spin is ± 1) and update x_{ij} using

$$P(x_{ij} = +1 \mid \cdot) = \frac{1}{1+\exp(-\beta S)}.$$

Equivalently, the log-odds are

$$\log \frac{P(x_{ij} = +1 \mid \cdot)}{P(x_{ij} = -1 \mid \cdot)} = \beta S.$$

So increasing β amplifies the influence of neighbour agreement: if most neighbours are $+1$ then $S > 0$ and the update strongly favours $+1$; if most neighbours are -1 then $S < 0$ and the update strongly favours -1. For very small β the update is close to a fair coin flip, while for large β the update becomes almost deterministic.

$\beta = 4$ **(strong ferromagnetism, two modes, and slow mixing).** Because β is large, the model becomes strongly ferromagnetic: it heavily prefers neighbouring spins to be equal. This creates two stable modes (two high-probability regions): one mode where the grid is mostly all -1, and one mode where the grid is mostly all $+1$. There is no external field, so there is no bias between these modes and the true distribution is symmetric under the global flip $x \mapsto -x$. However, single-site Gibbs has a hard time moving from one mode to the other when $\beta = 4$, so the chain can get "stuck" in whichever mode it reaches early.

The concrete reason is numerical. The update uses

$$P(x_{ij} = +1 \mid \text{rest}) = \frac{1}{1 + \exp(-\beta S)}, \qquad S = \sum_{\text{neighbors}} x_k.$$

If the neighbourhood around a site is all -1, then an interior site has 4 neighbours so $S = -4$, and with $\beta = 4$,

$$P(x_{ij} = +1 \mid S = -4) = \frac{1}{1 + \exp(16)} \approx 1.1 \times 10^{-7}.$$

So once the grid becomes mostly -1, flipping a single interior spin to $+1$ is essentially impossible. Even on an edge where $S = -3$,

$$\frac{1}{1 + \exp(12)} \approx 6 \times 10^{-6}.$$

That means the chain almost never creates a $+1$ "island", and without islands it cannot grow a $+1$ phase and cross to the other mode. This explains why a single-run estimate at $\beta = 4$ can look "one-sided": in our run the chain fell into the mostly -1 mode, so nearly all mass appears at $(-1, -1)$. If we re-run with an all $+1$ initial grid, we typically see the mirror image (most mass at $(+1, +1)$). This is a mixing issue, not a property of the true model. The true joint must satisfy

$$P(-1, -1) = P(+1, +1), \qquad P(-1, +1) = P(+1, -1),$$

with off-diagonals near 0 for large β.

It is not guaranteed that a single run (with one particular random seed) will land in the "$+1$" and "-1" modes with a 50/50 split. However, across many independent runs (using different random seeds and/or different random initial grids), we expect it to be roughly 50/50 which mode the chain becomes trapped in when $\beta = 4$. This is because the model and the Gibbs updates are symmetric in the absence of an external field, so there is no genuine preference for $+1$ over -1. With a random (approximately 50/50) initial grid and a fresh seed, the chain is therefore about equally likely to settle into the mostly $+1$ phase or the mostly -1 phase, though with only a small number of runs it is normal to observe imbalances such as 60/40 purely due to randomness.

$\beta = 1$ **(moderate coupling).** At $\beta = 1$, neighbour agreement is preferred but not overwhelmingly, so flips still occur with noticeable probability and the chain mixes much better than at $\beta = 4$. The resulting table shows clear positive correlation between the two corner spins: diagonal probabilities are higher than 0.25 and off-diagonals are lower than 0.25. This matches the intuition that information can propagate through the grid via neighbour interactions, making the two distant corners more likely to agree than disagree.

$\beta = 0.01$ **(very weak coupling; near-independence).** When β is very small, $P(x_{ij} = +1 \mid \cdot) \approx \frac{1}{2}$ for all neighbour sums S, so each update is close to a fair coin flip. As a result, two distant spins are approximately independent and the joint table is close to uniform with all entries near 0.25; small deviations are consistent with finite Monte Carlo sampling noise.

Fair comparison to Q1 (symmetrised Gibbs). Overall, the Gibbs estimates agree well when compared to the exact message-passing results from Q1:

- For $\beta = 0.01$, the distribution is close to uniform (0.25 in each entry); small deviations in Gibbs are due to finite Monte Carlo sampling (and residual autocorrelation).
- For $\beta = 1$, the exact result shows moderate positive correlation (diagonal entries larger than off-diagonals). The Gibbs table is close, with remaining discrepancy attributable to Monte Carlo error from a finite run.
- For $\beta = 4$, the exact result has nearly all mass on the diagonal with very small off-diagonals. Single-site Gibbs can become trapped in one magnetised mode at large β; therefore we report the symmetrised Gibbs estimate, which matches the exact flip-symmetric structure and is extremely close numerically.