The Reverse Ising Problem

Isaac Martin

February 20, 2023

The Reverse Ising Problem

Introduction and Terminology

ISAAC MARTIN

The Ising Model

The vectors h and J encode first and second order energy contributions respectively. Elements of J encode interactions between vertices whereas elements of h encode the individual energy contributions of vertices.

Definition 1.1.1: Ising Graph

An **Ising graph** consists of the following data:

- a finite set G, typically $\{1..n\} \subset \mathbb{N}$ for some n

- a vector $h \in \mathbb{R}^n$ called the **local bias vector**

- a vector $J \in \mathbb{R}^{n(n-1)/2}$ called the **interaction vector**.

We refer to h and J collectively as the **parameters** of G.

We immediately have a couple of additional comments:

- The number $n(n-1)/2$ is exactly the number of elements above the diagonal in a $n \times n$ matrix.

- One typically indexes the elements of J by pairs (i, j) with $i < j$.

- The object G is a vertex set and the vector J records the edge data. If $J_{ij} = 0$ then there is no edge (i, j), and if $J_{ij} \neq 0$ then there is an edge. One can even view G as a directed graph: if $J_{ij} > 0$ then the edge points $i \to j$ and if $J_{ij} < 0$ then $i \leftarrow j$.

One should then picture an Ising graph as a graph with $|G|$ nodes where vertex i is decorated with h_i and edge (i, j) is decorated with J_{ij}.

Though $X = \{-1, 1\}$ in all cases, we keep X arbitrary in order to explore more general graph dynamics in the future and because it's more convenient to write.

Perhaps it would be better to write Σ_G for S_G and σ for states.

Definition 1.1.2: State Space

Let G be an Ising graph and X some set. The **state space** of G, denoted S_G, is the collection of all functions $G \longrightarrow X$:

$$S_G := \mathrm{Hom}(G, X).$$

We refer to the elements of S_G as the *states* of G. It will always be the case that $X = \{-1, 1\}$ in this document unless otherwise specified.

Remark 1.1.3

A function $s : G \longrightarrow X$ just assigns a vertex $i \in G$ to a value ± 1. We can therefore think of s as a "vector" in $X^{|G|}$, where $s_i = s(i)$. For example, the state $s(i) = -1^i$ can be written as the vector $(-1, 1, -1, 1, ..., -1^{|G|})$.

We refer to the value $H(s)$ as the **energy** of the state s.

> **Definition 1.1.4: Ising Hamiltonian**
>
> Let G be an Ising graph. The function $H : S_G \longrightarrow \mathbb{R}$ defined
>
> $$H(s) = \sum_{i \in G} h_i s_i \ + \ \sum_{i < j} J_{ij} s_i s_j$$
>
> is called the **Hamiltonian** of G.

> **Model 1.1.5: The Ising Model**
>
> The probability an Ising graph G is in a state $s \in S_G$ is given by
>
> $$P(s) = \frac{e^{-\beta H(s)}}{Z_\beta}$$
>
> where $\beta \geq 0$ is a parameter called the *inverse temperature* and Z_β is the normalization constant
>
> $$Z_\beta = \sum_{s \in S_G} e^{-\beta H(s)}.$$

In other words, the Ising model says that one is more likely to find an Ising graph in a low energy state than a high energy state. The **Ising Problem** is then to find the lowest energy states of a given Ising graph.

The Reverse Ising Problem

Consider instead the reverse problem: one is given a graph G and some spin (or collection of spins) $s \in S_G$ and wants to find h and J such that the resulting Ising graph is in state s with high probability. Stated another way, one wishes to find h and J which minimize $H(s)$. This is known as the **reverse Ising problem.**

Motivating Example: AND

Why might one care about this? Consider a graph G with three vertices, v_1, v_2, v_3. Let's call v_1 and v_2 "input" vertices and v_3 the "output" vertex. We can write the AND circuit as the following table of states:

s_1	s_2	s_3
1	1	1
1	-1	-1
-1	1	-1
-1	-1	-1

This is the AND truth table with 1 denoting "True" and -1 denoting "False".

The first row is the state $s : G \longrightarrow X$ defined $s(v_1) = s(v_2) = s(v_3) = 1$, the second is the state $s : G \longrightarrow X$ defined $s(v_1) = 1$, $s(v_2) = s(v_3) = -1$, etc.

We would like to build an Ising graph which somehow implements this circuit. What this means is that, if we look at our Ising graph G and observe that the input spins are s_1 and s_2, then we want s_3 to assume the corresponding output value in the above table with high probability. I.e. if our

Ising graph is in a state with $(s_1, s_2) = (1, 1)$ then we want $s_3 = 1$, and so on for the other states.

There are two things we must do to build a robust circuit. First, we need some way to control the input spins – ideally, we should be able to fix the inputs at a certain value while letting the outputs vary. This is a physics problem and is not of our concern. [1] Second, we need to choose h and J such that for each input $s = (s_1, s_2)$, the "correct" output value has a higher probability of occurring than the incorrect output. Since lower energy states occur with higher probability, this is the same as requiring that the following constrains are satisfied:

[1] It is nonetheless possible, for instance, one could choose the components of h corresponding to inputs in such a way as to drastically favor a certain value for the input spins.

$$H(1, 1, 1) < H(1, 1, -1)$$
$$H(1, -1, -1) < H(1, -1, -1)$$
$$H(-1, 1, -1) < H(-1, 1, -1)$$
$$H(-1, -1, -1) < H(-1, -1, 1).$$

One can check that this is indeed possible: **INSERT CORRECT h AND J HERE**.

Ising Circuits

Definition 1.2.1: Pre-Ising Circuit

A **pre-Ising circuit** (G, N, M, f) is a set of vertices G with a decomposition $G = N \cup M$ satisfying $N \cap M = \varnothing$ and a function $f : S_N \longrightarrow S_M$.

Comments:

- We call N the collection of **input vertices** and M the collection of **output vertices**.

- We call S_N the **input state/spin space** and S_M the **output state/spin space**.

- The function f is the **logic** of the circuit and it is not required to satisfy any additional requirements. We consider the output state $f(s)$ to the correct output corresponding to the input state s.

Notice that $Hom(N, X) \times Hom(M, X)$ is canonically isomorphic to $Hom(G, X)$ in the category Set, which is another nicety of the way we've set everything up.

Definition 1.2.2: Ising Circuit

An **Ising circuit** consists of the data (G, N, M, h, J, f) such that

(i) (G, h, J) is an Ising graph

(ii) (G, N, M, f) is a pre-Ising circuit

(iii) For each pair $(s, t) \in S_N \times S_M = S_G$ we have that $H(s, t) \geq H(s, f(s))$ with equality if and only if $f(s) = t$.

If you stop to think about condition (iii), you'll conclude that it really couldn't be anything else. What other way is there to make an Ising graph and an abstract circuit compatible?

Conditions (i) and (ii) in the above definition are self explanatory. Condition (iii) merely requires that the circuit structure of G is compatible with the Ising graph structure of G. It asks that the output t which minimizes $H(s, t)$ is the correct one; in other words, the correct output corresponding

to an input s is the one which occurs with highest probability. We have an immediate question:

Question 1.2.3

Can every pre-Ising circuit G be realized as an Ising circuit for an appropriate choice of h and J? (This is known as *solving* a pre-Ising circuit.)

In order to solve a pre-Ising circuit, we need to find h and J such that condition (iii) is satisfied. This means we need to satisfy all constraints of the form

$$H(s, f(s)) < H(s, t)$$

simultaneously for all possible input states $s \in S_N$ and output states $t \neq f(s)$. We can arrange these constraint equations in a convenient way through *input levels*.

Input Levels

Definition 1.2.4: Input Level

Let $\mathbf{s} \in S_N$ be an input spin. Then $L_{\mathbf{s}} = \{\mathbf{s}\} \times S_M$ is called the **input level** at $\mathbf{s}$.

We say that a choice of h and J **solves** an input level $L_{\mathbf{s}}$ if all constraints of the form

$$H(\mathbf{s}, t) > H(\mathbf{s}, f(\mathbf{s})) \tag{1.1}$$

are satisfied for $t \neq f(\mathbf{s})$, i.e. if $(\mathbf{s}, f(\mathbf{s}))$ *strictly* minimizes the Hamiltonian for the input level:

$$(\mathbf{s}, t) = \min_{\mathbf{u} \in L_{\mathbf{s}}} H(\mathbf{u}) \implies t = f(s).$$

There are $M - 1$ constraint equations for each input level and N distinct input levels, so to solve a pre-Ising circuit one must find h and J which simultaneously satisfy $N(M - 1)$ constraints.

It is always possible to choose $\mathbf{h}$ and $\mathbf{J}$ which satisfy a specified input level.

Proposition 1.2.5: Global Minimization

Let G be a pre-Ising circuit, $\mathbf{s} \in S_N$ an input state and $\mathbf{u} = (\mathbf{s}, f(\mathbf{s})) \in S_G$ the state of G obtained by concatenating the input $\mathbf{s}$ with the correct output state $f(\mathbf{s})$. Then $\mathbf{h} \in \mathbb{R}^G$ and $\mathbf{J} \in \mathbb{R}^{G(G-1)/2}$ defined

$$\mathbf{h} = -\mathbf{u} \text{ and } \mathbf{J}_{ij} = -\mathbf{u}_i \cdot \mathbf{u}_j \text{ for } i < j$$

solves the input level $L_{\mathbf{s}}$.

This proposition essentially says "the greedy approach works" for an input level. Choose $\mathbf{h}$ and $\mathbf{J}$ so that the contribution of the $\mathbf{h}_i$ and $\mathbf{J}_{ij}$ component to the Hamiltonian is always negative for the correct input/output pair. Any state differing from this specific pair will flip the sign of **at least** one $\mathbf{h}_i$ or $\mathbf{J}_{ij}$ and thus result in a strictly larger Hamiltonian.

PROOF : Consider any other $\mathbf{v} \in L_{\mathbf{s}}$ for $\mathbf{v} \neq \mathbf{u}$. Since $\mathbf{h}_i = -\mathbf{u}_i$ and $\mathbf{J}_{ij} = -\mathbf{u}_i \mathbf{u}_j$ we

have that

$$H(\mathbf{v}) - H(\mathbf{u}) = \sum_{i=1}^{G} \mathbf{h}_i(\mathbf{v}_i - \mathbf{u}_i) \; + \; \sum_{i<j} \mathbf{J}_{ij}(\mathbf{v}_i\mathbf{v}_j - \mathbf{u}_i\mathbf{u}_j)$$

$$= \sum_{i=1}^{G} (-\mathbf{u}_i\mathbf{v}_i + 1) \; + \; \sum_{i<j}(-\mathbf{v}_i\mathbf{v}_j\mathbf{u}_i\mathbf{u}_j + 1).$$

But $\mathbf{u}_i\mathbf{v}_i, \mathbf{v}_i\mathbf{v}_j\mathbf{u}_i\mathbf{u}_j \in \{-1, 1\}$, hence $-\mathbf{u}_i\mathbf{v}_i + 1$ and $-\mathbf{v}_i\mathbf{v}_j\mathbf{u}_i\mathbf{u}_j + 1$ are both either 0 or 1. This means $H(\mathbf{v}) - H(\mathbf{u}) \geq 0$ with equality only when $\mathbf{v} = \mathbf{u}$, hence all constraints of the form (1.1) are satisfied. ∎

We can also simultaneously minimize inputs which share the same output.

Proposition 1.2.6: Minimize an output

Suppose input states $\mathbf{s}_1, ..., \mathbf{s}_k \in S_N$ all share the same output, i.e. $f(\mathbf{s}_1) = ... = f(\mathbf{s}_k) = \mathbf{t} \in S_M$ for some $\mathbf{t}$. Then there exist $\mathbf{h}$ and $\mathbf{J}$ which simultaneously solve the input levels $L_{\mathbf{s}_1}, ..., L_{\mathbf{s}_k}$.

PROOF : Define $\mathbf{h}$ and $\mathbf{J}$ as follows:

$$\mathbf{h}_i = \begin{cases} 0 & \text{if } i \leq N \\ -\mathbf{t}_{i-N} & \text{if } i > N \end{cases} \quad \text{and} \quad J_{ij} = \begin{cases} 0 & i \geq N \\ -\mathbf{t}_{i-N}\mathbf{t}_{j-N} & i > N \end{cases}.$$

Then a similar check as in Proposition (1.2) proves the desired result. ∎

Notice, however, that the choice of $\mathbf{h}$ and $\mathbf{J}$ made in the above proof is terrible for an Ising circuit with more than one output. It ensures that the Hamiltonian value of $\mathbf{u} \in S_G$ depends only on the output components of $\mathbf{u}$. In particular, if $\mathbf{s}$ is some input such that $f(\mathbf{s}) \neq \mathbf{t}$, then this choice of $\mathbf{h}$ and $\mathbf{J}$ are guaranteed *not* to solve $L_{\mathbf{s}}$. Is there anything we can say about Question (1.2) in general for circuits with 2 or more outputs? The answer is an emphatic **no**.

A non-example: XOR

Consider the 2-bit XOR circuit. This is a graph consisting of 3 vertices, two input vertices and one output vertex. It thus necessarily has only two possible outputs, -1 and 1.

I'll fill this in at some point, seems tedious now though.

Auxiliary Spins and the XOR fix

Notice that this choice of $\mathbf{h}$ and $\mathbf{J}$ *globally* minimizes the state $\mathbf{u}$. In some ways this it the thematic opposite of what we want. We need to minimize *all* correct input/output pairs at once, but the minimization only needs to occur among spins in the same input level. What we have accomplished is minimizing a single input/output pair globally. For this reason, we say that $\mathbf{u}$ has been *globally minimized.*

<div style="border:1px solid black; padding:1em;">

The Reverse Ising Problem

Virtual States and Random Parameters

ISAAC MARTIN

</div>

The main point of this section is to explore how auxiliary spins should be added to a pre-Ising circuit. We'll consider adding auxiliary spins one at a time with each spin added monotonically decreasing some cost function. Specifically, we'll try to answer the following question:

> **Question 2.0.1**
>
> Given a random choice of h and J, how do we choose an auxiliary spin at each input level $L_{\mathbf{s}}$ such that, if (h, J) solves $L_{\mathbf{s}}$ it also solves other input levels with high probability?

Our philosophy is, essentially, that we should choose auxiliary spins so that when one input level is solved by a random choice of (h, J) it increases the likelihood that other input levels are also solved. We want high correlation between the "solvedness" of input levels.

Our setup will introduce two new ideas: virtual spin space and random parameters. Throughout this section (G, N, M, f) is a pre-Ising circuit.

Virtual Spin Space

Consider a pre-Ising circuit G. Define a new set

$$E = \{(i, j) \mid i \leq j, i, j \in G\}.$$

This is the set of all possible edges we could form between vertices in G if we allow self-connections. It is also all indices of elements which are either on or above the diagonal of a $G \times G$ matrix. The cardinality of E is $|G| \cdot (|G| + 1)/2$, or written another way $|G| + |G| \cdot (|G| - 1)/2$. From this it is clear that the state space S_E has one spin for every vertex and another for every pair of vertices in G.

We also define the $|E|$-dimensional vector space

$$\mathbb{R}^E = \bigoplus_{(i,j) \in E} (i, j)\mathbb{R}.$$

We can also interpret $\mathbb{R}^E$ to mean the "set of all functions $E \longrightarrow \mathbb{R}$", which is the set theoretic interpretation of the notation anyway. This is a $\mathbb{R}$-vector space and is canonically isomorphic to the vector space defined here.

A vector $\mathbf{x} \in \mathbb{R}^E$ is indexed via elements in E: $\mathbf{x}_{(i,j)}$ is the component of $\mathbf{x}$ corresponding to (i, j). This gives us a convenient way to package $\mathbf{h}$ and $\mathbf{J}$ as a single vector, $\mathbf{x}_{(i,i)}$ is a choice for h_i and $\mathbf{x}_{(i,j)}$ for $i < j$ is a choice for $\mathbf{J}_{ij}$. Stated more formally, we have an isomorphism

$$\mathbb{R}^{|G|} \oplus \mathbb{R}^{|G|(|G|-1)/2} \longrightarrow \mathbb{R}^E \qquad (2.1)$$

given by sending a vector $(\mathbf{h}, \mathbf{J})$ to the vector $\mathbf{p}$ where $\mathbf{p}_{(i,i)} = \mathbf{h}_i$ and $\mathbf{J}_{ij} = \mathbf{p}_{(i,j)}$. Because $\mathbf{h}$ and $\mathbf{J}$ are called the parameters of an Ising graph, we call $\mathbb{R}^E$ the **parameter space** of G.

Now consider the map

$$\varphi : S_G \longrightarrow S_E, \ \varphi(s)(i, j) = \begin{cases} s(i) & \text{if } i = j \\ s(i)s(j) & \text{if } i = j \end{cases} \qquad (2.2)$$

The map φ takes states of G and produces states on E. If we identify G with its image in E under the diagonal embedding $i \longmapsto (i,i)$ we see that $\varphi(\mathbf{s})|_G = \mathbf{s}$, meaning that the state $\varphi(\mathbf{s})$ recovers $\mathbf{s}$ when restricted to G. However, it also contains the second order interactions between spins of $\mathbf{s}$. If we think of states instead as vectors in $X^{|G|}$ and $X^{|E|}$, then we can write φ in a more enlightening way:

$$\varphi(\mathbf{s}_1, ..., \mathbf{s}_G) = (\mathbf{s}_1, ..., \mathbf{s}_G, \mathbf{s}_1\mathbf{s}_2, \mathbf{s}_1\mathbf{s}_3, ..., \mathbf{s}_1\mathbf{s}_G, \mathbf{s}_2\mathbf{s}_G, ..., \mathbf{s}_{G-1}\mathbf{s}_G). \tag{2.3}$$

From this description, it is clear that φ simply concatenates all pairwise products of components in $\mathbf{s}$

Notice that, for some choice of $\mathbf{h}, \mathbf{J}$, the Hamiltonian of G can now be written as the dot product

$$H(\mathbf{s}) = \langle (\mathbf{h}, \mathbf{J}), \varphi(\mathbf{s}) \rangle$$

where $(\mathbf{h}, \mathbf{J})$ is identified with its image in $\mathbb{R}^E$. The state space S_E contains states which correspond to states of G, but it also contains many others. For this reason, we call it the **virtual state space** of G.

Random Parameters

Consider a choice of $P \in \mathbb{R}^E$ sampled from some distribution, e.g. a multivariate Gaussian [1] From the argument used in the proof of Proposition (1.2) it is clear that the virtual spin $\mathbf{v} \in S_E$ whose sign in each component opposes the sign of the corresponding elements in P will minimize $\langle P, \mathbf{v} \rangle$:

$$\mathbf{v}_{i,j} = -\frac{P_{i,j}}{|P_{i,j}|}, \quad \text{for } (i,j) \in E. \tag{2.4}$$

A harder question is the following:

Question 2.2.1

Fix an input $\mathbf{s} \in S_N$. For which $\mathbf{x} \in L_\mathbf{s}$ is $\langle P, \varphi(\mathbf{x}) \rangle$ minimized?

Intuitively, the $\mathbf{x}$ which minimizes $\langle P, \varphi(\mathbf{x}) \rangle$ should be close to the virtual spin which minimizes the inner product with P. Indeed, under some additional assumptions on P this is true.

Proposition 2.2.2

Suppose that $|P_{i,j}| = |P_{i',j'}|$ for all $(i,j), (i',j') \in E$ and let $\mathbf{v} \in S_E$ be the virtual spin which minimizes $\langle P, \mathbf{v} \rangle$. Then

$$\mathbf{x} = \arg\min_{\mathbf{x} \in L_\mathbf{s}} \langle P, \varphi(\mathbf{x}) \rangle$$

if and only if

$$\mathbf{x} = \arg\min_{\mathbf{x} \in L_\mathbf{s}} d(\mathbf{v}, \mathbf{x}),$$

where d is Hamming distance.

---

[1] This seems like a good choice as it is spherically symmetric.

PROOF :

$$\mathbf{x} = \arg \min_{\mathbf{x} \in L_{\mathbf{s}}} \langle P, \varphi(\mathbf{x}) \rangle$$

$$\Longleftrightarrow \ \mathbf{x} = \arg \min_{\mathbf{x} \in L_{\mathbf{s}}} \left(\langle P, \varphi(\mathbf{x}) \rangle - \langle P, \mathbf{v} \rangle \right)$$

$$\Longleftrightarrow \ \mathbf{x} = \arg \min_{\mathbf{x} \in L_{\mathbf{s}}} \langle P, \varphi(\mathbf{x}) - \mathbf{v} \rangle$$

$$\Longleftrightarrow \ \mathbf{x} = \arg \min_{\mathbf{x} \in L_{\mathbf{s}}} \left\langle P, \frac{1}{2}(\varphi(\mathbf{x}) - \mathbf{v}) \right\rangle.$$

Consider $\varphi(\mathbf{x})_{i,j} - \mathbf{v}_{i,j}$ for $(i,j) \in E$. If $\varphi(\mathbf{x}_{i,j}) = \mathbf{v}_{i,j}$ then this is zero. If $\varphi(\mathbf{x}_{i,j}) \neq \mathbf{v}_{i,j}$ then $\varphi(\mathbf{x})_{i,j} = -\mathbf{v}_{i,j} = P_{i,j}/|P_{i,j}|$ by equation (2.4), and hence $\varphi(\mathbf{x})_{i,j} - \mathbf{v}_{i,j} = 2$. We then have

$$\left\langle P, \frac{1}{2}(\varphi(\mathbf{x}) - \mathbf{v}) \right\rangle = |P_{1,1}| \cdot d(\varphi(\mathbf{x}), \mathbf{v})$$

because all components of P are equal, and we are done. ∎

For an arbitrary P we get something similar.

Proposition 2.2.3

Let $P \in \mathbb{R}^E$ be a parameter vector for G and $\mathbf{v} \in S_E$ be the virtual spin which minimizes $\langle P, \mathbf{v} \rangle$. Then for a fixed $\mathbf{s} \in S_N$,

$$\mathbf{x} = \arg \min_{\mathbf{x} \in L_{\mathbf{s}}} \langle P, \varphi(\mathbf{x}) \rangle$$

if and only if

$$\mathbf{x} = \arg \min_{\mathbf{x} \in L_{\mathbf{s}}} \sum_{(i,j) \in D(\mathbf{x}, \mathbf{v})} |P_{i,j}|$$

where $D(\mathbf{x}, \mathbf{v}) = \{ (i,j) \in E \ | \ \varphi(\mathbf{x})_{i,j} \neq \mathbf{v}_{i,j} \}$ is the collection of indices where $\varphi(\mathbf{x})$ and $\mathbf{x}$ disagree.

PROOF : This is the same proof as Proposition (2.2) once one notices that

$$\left\langle P, \frac{1}{2}(\varphi(\mathbf{x}) - \mathbf{v}) \right\rangle = \sum_{(i,j) \in D(\mathbf{x}, \mathbf{v})} |P_{i,j}|^2.$$ ∎

One should think of Proposition 2.2 as providing a way to rank the disagreements between components of $\varphi(\mathbf{x})$ and $\mathbf{v}$, if $|P_{i,j}|$ is small then the penalty for disagreement is also small.

Proposition (2.2) tells us that if we could quickly calculate the value of the *level minimizer function* $\psi : \mathbb{R}^E \longrightarrow S_G$ defined

$$\psi(P) = \arg \min_{\mathbf{x} \in L_{\mathbf{s}}} \sum_{(i,j) \in D(\mathbf{x}, \mathbf{v})} |P_{i,j}| \tag{2.5}$$

then we could quickly detect how many input levels are satisfied by a choice of P. This would then help us in choosing auxiliary spins which maximize correlation among the input levels.

References