Hidden Independence in Unstructured Probabilistic Models

Antony Pearson [iD]

Department of Applied Mathematics, University of Colorado Boulder, CO, The United States
IQ Biology Program, BioFrontiers Institute, University of Colorado Boulder, CO, The United States

Manuel E. Lladser[1] [iD]

Department of Applied Mathematics, University of Colorado Boulder, CO, The United States
Department of Computer Science (affiliate), University of Colorado Boulder, CO, The United States
manuel.lladser@colorado.edu

─── **Abstract** ───

We describe a novel way to represent the probability distribution of a random binary string as a mixture having a maximally weighted component associated with independent (though not necessarily identically distributed) Bernoulli characters. We refer to this as the *latent independent weight* of the probabilistic source producing the string, and derive a combinatorial algorithm to compute it. The decomposition we propose may serve as an alternative to the Boolean paradigm of hypothesis testing, or to assess the fraction of uncorrupted samples originating from a source with independent marginal distributions. In this sense, the latent independent weight quantifies the maximal amount of independence contained within a probabilistic source, which, properly speaking, may not have independent marginal distributions.

1 Introduction

Consider the Bayesian network [5] in Figure 1, given in [11, Chapter 2]. As the reader may find familiar, each random variable (node) in the network, given the configurations of its parents, is by definition conditionally independent from its non-descendants. Accordingly, the joint probability mass function of the binary random vector (P, T, S, L, X) factorizes as follows:

$$\mathbb{P}(P = p, T = t, S = s, L = l, X = x) = \mathbb{P}(P = p) \cdot \mathbb{P}(T = t) \cdot \mathbb{P}(S = s \mid T = t)$$
$$\cdot \mathbb{P}(L = l \mid P = p, T = t) \cdot \mathbb{P}(X = x \mid L = l).$$

In particular, the joint distribution of P, T, S, L and X can be encoded with 10 free parameters. Perhaps unexpectedly, however, one can represent this joint distribution as a mixture with a heavily weighted "independent" component. Specifically:

$$\mathbb{P} = 0.94 \cdot Be(0.02) \otimes Be(0.005) \otimes Be(0.6) \otimes Be(0.01) \otimes Be(0.6) + 0.06 \cdot R, \tag{1}$$

---

[1] Corresponding author

where $Be(p)$ denotes a Bernoulli distribution with success probability p, the operator $\otimes$ denotes product measures, and R is a "residual" probability distribution over the sample space $\{0,1\}^5$. This decomposition of $\mathbb{P}$ is possible because for each outcome $(p,t,s,l,x) \in \{0,1\}^5$ a computation shows that:

$$\mathbb{P}(P = p, T = t, S = s, L = l, X = x)$$
$$\geq\ 0.94 \cdot 0.02^p\, 0.98^{1-p} \times 0.005^t\, 0.995^{1-t} \times 0.6^l\, 0.4^{1-l} \times 0.01^x\, 0.99^{1-x} \times 0.6^s\, 0.4^{1-s}.$$

The residual distribution R may be obtained solving for it in equation (1). It turns out in this case that R has low entropy (≈ 3.2 bits, compared to the uniform distribution over $\{0,1\}^5$, which has 5 bits of entropy), and gives probability 0 to twelve of the thirty-two outcomes.

The identity in equation (1) means that, conditioned on a hidden event of 94% probability, the presence of lung infiltrates, the outcome of an X-ray and sputum smear, and the status of a patient having tuberculosis or pneumonia will all be rendered independent. Thus, while in a clinical setting, the dependencies encoded in the Bayesian network may be relevant, on the population level, these covariates often behave independently; in particular, most samples from the Bayesian network can be attributed to a much simpler model (with 5 instead of 10 free parameters).

The decomposition in (1) bears the question: *what's the largest weight a product of independent Bernoulli distributions can have as component of $\mathbb{P}$?* Remarkably, the marginal distributions of $\mathbb{P}$ are associated with a weight that is significantly smaller than 94%. Indeed, a computation shows that $P \sim Be(0.05)$, $T \sim Be(0.02)$, $S \sim Be(0.6)$, $L \sim Be(0.05)$, and $X \sim Be(0.6)$, and that P admits the mixture representation:

$$\mathbb{P} = \epsilon \cdot Be(0.05) \otimes Be(0.02) \otimes Be(0.6) \otimes Be(0.05) \otimes Be(0.6) + (1 - \epsilon) \cdot R',$$

where $\epsilon \approx 0.104$, and R' is a probability distribution that can be determined from the above identity.

In this article we develop the mathematics of the so-called *(latent) independent weight* of an arbitrary joint probability distribution over a sample space of the form $\{0,1\}^d$, with $d \geq 1$ finite. We argue that the independent weight of a probabilistic source describes the largest average fraction of samples from it that can be attributed to (conditionally) independent Bernoulli random variables, and describe an algorithm to compute this weight, along with some heuristics to approximate it.

The independent weight of a probabilistic source is an intrinsic property of it, which can be used as an objective measure of the approximate correctness of the null hypothesis that "the source has independent marginal distributions," which may be nevertheless false (as the example associated with Figure 1). The concept of independent weight may also be used to distill corrupted data from a source with otherwise independent marginal distributions.

1.1   Related Work

The present work may be regarded as a non-trivial specialization of the recent theory developed in [8]. This previous work introduces the concept of the *latent weight* of a probabilistic source (such as $\mathbb{P}$ in the previous example) with respect to a structured class $\mathcal{Q}$ of probability models over a finite sample space. Specifically, the latent weight of a source P with respect to a class $\mathcal{Q}$ of models is defined as [8]:

$$\lambda_{\mathcal{Q}}(P) := \sup\{\lambda \geq 0 \mid P \geq \lambda \cdot Q \text{ for some } Q \in \mathcal{Q}\}. \tag{2}$$
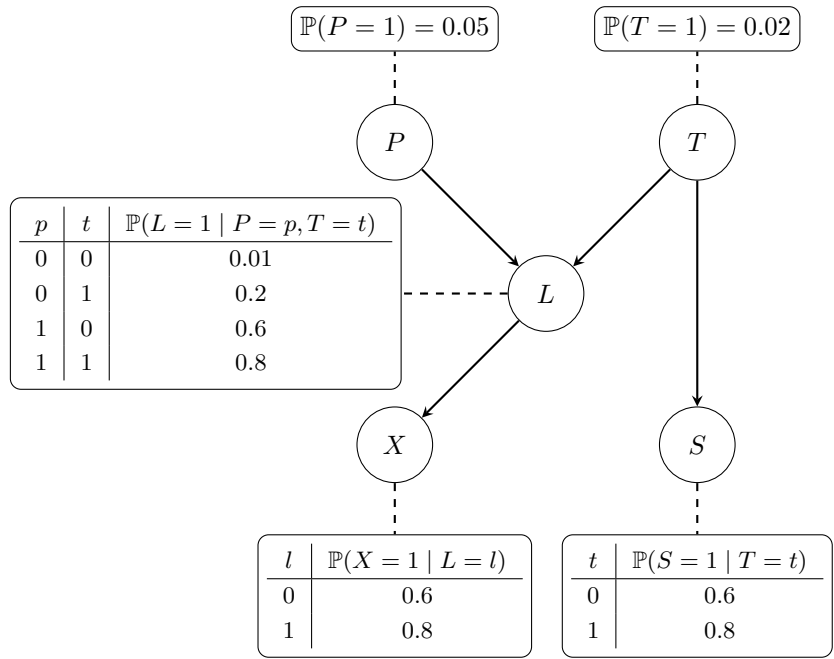
Figure 1 Bayesian network that models the interaction between two lung conditions, tuberculosis (T), and pneumonia (P), and how they jointly affect the probability that a patient will have lung infiltrates (L), the presence of said infiltrates in an X-ray (X), and the outcome of a sputum smear test (S) for tuberculosis. Nodes represent Bernoulli random variables, with conditional probability tables indicated, and the value 1 (0) indicates the presence (absence) of the corresponding condition.

This coefficient represents the largest weight that can be given to a model in $\mathcal{Q}$ as a component in a mixture decomposition of P. In fact, under mild technical conditions, there always exists $Q \in \mathcal{Q}$ and a probabilistic model R such that

$$P = \lambda_{\mathcal{Q}}(P) \cdot Q + (1 - \lambda_{\mathcal{Q}}(P)) \cdot R. \tag{3}$$

Furthermore, when $\mathcal{Q}$ is convex, Q is unique when $\lambda_{\mathcal{Q}}(P) > 0$, and so is R when $\lambda_{\mathcal{Q}}(P) < 1$.

In the current setting, $\mathcal{Q}$ is the class of probability distributions associated with independent binary random variables. We emphasize that much of what we present in this extended abstract may be generalized to more general discrete random variables, however, the binary setting presents enough mathematical challenges to consider it in isolation.

2 Latent Independent Weights

In what follows, $\mathcal{P}$ denotes the set of all probability distributions on $\{0, 1\}^d$, with $d \geq 1$ a given integer. In particular, we may think of elements in $\mathcal{P}$ as non-negative real vectors of dimension 2^d, with entries that sum up to 1.

For $P, Q \in \mathcal{P}$ and $\lambda \in \mathbb{R}$, we write $P \geq \lambda \cdot Q$ to mean that $P(\omega) \geq \lambda \cdot Q(\omega)$, for each $\omega \in \{0, 1\}^d$. Further, we say that Q has *independent marginal distributions* (in short, *independent marginals*) if and only if there are probability distributions $\mu_1, \ldots, \mu_d$ defined over $\{0, 1\}$ such that $Q = \otimes_{i=1}^d \mu_i$. Equivalently, Q has independent marginals if and only if it is the probability distribution of a random vector of the form $(X_1, \ldots, X_d)$, with $X_1, \ldots, X_d$ independent (though not necessarily identically distributed) Bernoulli random variables. (In this case, each X_i has distribution μ_i.)

We associate to each $P \in \mathcal{P}$ the real coefficient:

$$\lambda(P) := \lambda_{\mathcal{Q}}(P), \tag{4}$$

where $\lambda_{\mathcal{Q}}(P)$ is as in equation (2) and $\mathcal{Q} \subset \mathcal{P}$ denotes the set of models with independent marginal distributions.

Clearly, $0 \leq \lambda(P) \leq 1$. In fact, according to [8], $\lambda(P) = 1$ if and only if P has independent marginal distributions itself. Furthermore, because the subset of distributions in $\mathcal{P}$ with independent marginal distributions is compact, the supremum in equation (4) is always achieved [8]. Namely, there is $Q \in \mathcal{P}$ with independent marginals such that $P \geq \lambda(P) \cdot Q$. As a result, since $(P - \lambda(P) \cdot Q)$ is a measure with total mass $(1 - \lambda(P))$, there is also $R \in \mathcal{P}$ such that P admits the mixture decomposition:

$$P = \lambda(P) \cdot Q + \big(1 - \lambda(P)\big) \cdot R. \tag{5}$$

This decomposition motivates calling $\lambda(P)$ the *latent independent weight of P*, or simply the *independent weight of P*. It follows that $\lambda(P)$ *is the largest weight that can be attributed to a probability measure over $\{0,1\}^d$ with independent marginals as a component of P.* Equivalently: $\lambda(P)$ *is the maximal expected fraction of samples from P which may be attributed to a probabilistic source with independent marginal distributions.* More precisely, if $\mathbf{X} = (X_1, \ldots, X_d)$ has distribution P then, up to a hidden event with probability $\lambda(P)$, the Bernoulli random variables $X_1, \ldots, X_d$ are (conditionally) independent.

We note that the model Q with independent marginal distributions in equation (5) is not necessarily unique. For example, let $d = 2$ and P be the uniform distribution over $\{(0,0),(1,1)\}$; in this case, $P = \delta_{(0,0)}/2 + \delta_{(1,1)}/2$, where δ_x is the point probability mass at x. The reader can verify that the only models with independent marginals that can be given positive weight in a probability mixture decomposition of P are $\delta_{(0,0)}$ and $\delta_{(1,1)}$, hence the supremum in equation (4) is achieved by $\delta_{(0,0)}$ as well as $\delta_{(1,1)}$.

2.1   Alternative Formulations

In this section we show how to compute latent independents weights.

Henceforth, $P \in \mathcal{P}$ is assumed fixed. Moreover, we assume that $P > 0$, i.e. $P(\nu) > 0$ for each $\nu \in \{0,1\}^d$. This assumption can be relaxed but goes beyond the scope of this extended abstract.

In what follows, ∞ denotes $+\infty$.

For each $\omega = (\omega_1, \ldots, \omega_d) \in \{0,1\}^d$, let $f_\omega : [0,1]^d \to [1, \infty]$ be the function defined as

$$f_\omega(\mathbf{q}) := \frac{1}{\mathbb{P}(\mathbf{X_q} = \omega)} = \prod_{i=1}^{d} q_i^{-\omega_i}(1 - q_i)^{\omega_i - 1}, \text{ for } \mathbf{q} = (q_1, \ldots, q_d);$$

where $\mathbf{X_q} = (X_1, \ldots, X_d)$ is a vector of independent Bernoulli random variables, with $X_i \sim Be(q_i)$. (The second identity above requires to define $0^0 := 1$.) Clearly, $f_\omega(\mathbf{q})$ is a continuous function of $\mathbf{q}$.

For each $\omega \in \{0,1\}^d$, define

$$\mathcal{Q}_\omega := \big\{ \mathbf{q} \in [0,1]^d \,|\, \forall \nu \in \{0,1\}^d : P(\omega)f_\omega(\mathbf{q}) \leq P(\nu)f_\nu(\mathbf{q}) \big\}. \tag{6}$$

▶ **Lemma 1.** *If $P > 0$ then* $\lambda(P) = \displaystyle\max_{\omega \in \{0,1\}^d} \max_{\mathbf{q} \in \mathcal{Q}_\omega} P(\omega)f_\omega(\mathbf{q}).$

Proof. Since a probability measure over $\{0, 1\}^d$ with independent marginal distributions may be represented in terms of d independent Bernoulli random variables, we may restate equation (4) equivalently as follows:

$$\lambda(P) = \sup \left\{ \lambda \geq 0 \,\middle|\, \exists \mathbf{q} \in [0, 1]^d \, \forall \nu \in \{0, 1\}^d : P(\nu) \geq \lambda \cdot \prod_{i=1}^{d} q_i^{\nu_i} (1 - q_i)^{1 - \nu_i} \right\}$$

$$= \sup \left\{ \lambda \geq 0 \,\middle|\, \exists \mathbf{q} \in [0, 1]^d : \lambda \leq \min_{\nu \in \{0,1\}^d} P(\nu) f_\nu(\mathbf{q}) \right\}$$

$$= \sup_{\mathbf{q} \in [0,1]^d} \min_{\nu \in \{0,1\}^d} P(\nu) f_\nu(\mathbf{q})$$

$$= \max_{\mathbf{q} \in [0,1]^d} \min_{\nu \in \{0,1\}^d} P(\nu) f_\nu(\mathbf{q}),$$

where for the second identity we have used that $P > 0$, which prevents the possibility of dealing with anomalous products of the form $0 \cdot \infty$, and for the last identity we have used that $[0, 1]^d$ is compact and that the functions f_ν, with $\nu \in \{0, 1\}^d$, are continuous.

But observe that for each $\mathbf{q} \in [0, 1]^d$ there must exist an ω which minimizes (possibly with ties) the quantity $P(\nu) f_\nu(\mathbf{q})$, with $\nu \in \{0, 1\}^d$; that is, $[0, 1]^d \subset \cup_{\omega \in \{0,1\}^d} \mathcal{Q}_\omega$. Since, by definition, $\mathcal{Q}_\omega \subset [0, 1]^d$ for each ω, we obtain that

$$[0, 1]^d = \bigcup_{\omega \in \{0,1\}^d} \mathcal{Q}_\omega.$$

Finally, from the last identity for $\lambda(P)$, the defining property of the set $\mathcal{Q}_\omega$ implies that

$$\lambda(P) = \max_{\omega \in \{0,1\}^d} \max_{\mathbf{q} \in \mathcal{Q}_\omega} \min_{\nu \in \{0,1\}^d} P(\nu) f_\nu(\mathbf{q}) = \max_{\omega \in \{0,1\}^d} \max_{\mathbf{q} \in \mathcal{Q}_\omega} P(\omega) f_\omega(\mathbf{q}). \qquad \blacktriangleleft$$

Lemma 1 reduces the calculation of $\lambda(P)$ to 2^d optimization problems, one for each $\omega \in \{0, 1\}^d$, of the form:

$$\max_{\mathbf{q} \in \mathcal{Q}_\omega} P(\omega) f_\omega(\mathbf{q}), \text{ with } \omega \in \{0, 1\}^d. \tag{7}$$

Our next result aids in making these optimization problems more explicit.

▶ **Lemma 2.** *Assume $P > 0$. For a given $\omega \in \{0, 1\}^d$, the transformation $\mathbf{q} \longrightarrow \mathbf{x}$ with $\mathbf{x} = (x_1, \ldots, x_d)$ and $x_i := (\frac{q_i}{1 - q_i})^{1 - 2\omega_i}$, is a bijection between $[0, 1]^d$ and $[0, \infty]^d$, and in terms of the variable $\mathbf{x}$:*

$$f_\omega(\mathbf{q}) = \prod_{i=1}^{d} (1 + x_i). \tag{8}$$

Under this reparameterization, for each $\mathbf{q} \in (0, 1)^d$:

$$\mathbf{q} \in \mathcal{Q}_\omega \text{ if and only if } \forall \nu \in \{0, 1\}^d : \prod_{i : \nu_i \neq \omega_i} x_i \leq \frac{P(\nu)}{P(\omega)}, \tag{9}$$

where $\prod_{i : \nu_i \neq \omega_i} x_i := 1$ when $\nu = \omega$.

Proof. If $\omega_i = 1$ then $x_i = \frac{1 - q_i}{q_i}$, which is a strictly decreasing function of q_i. Instead, if $\omega_i = 0$ then $x_i = \frac{q_i}{1 - q_i}$, which is a strictly increasing function of q_i. Thus, in either case, x_i is a strictly monotone function of q_i, with range $[0, \infty]$ when $q_i \in [0, 1]$. From this it is

immediate that the transformation $\mathbf{q} \to \mathbf{x}$ from $[0,1]^d$ to $[0,\infty]^d$ is one-to-one and onto. On the other hand, if $\omega_i = 1$ then $q_i = \frac{1}{1+x_i}$, hence

$$q_i^{-\omega_i}(1-q_i)^{\omega_i-1} = \frac{1}{q_i} = 1 + x_i.$$

Likewise, if $\omega_i = 0$ then $q_i = \frac{x_i}{1+x_i}$ i.e. $(1-q_i) = \frac{1}{1+x_i}$, hence

$$q_i^{-\omega_i}(1-q_i)^{\omega_i-1} = \frac{1}{1-q_i} = 1 + x_i.$$

In either case, $q_i^{-\omega_i}(1-q_i)^{\omega_i-1} = (1+x_i)$, which implies the identity in equation (8).

Because $P > 0$, observe for $\mathbf{q} \in (0,1)^d$ that:

$$\mathbf{q} \in \mathcal{Q}_\omega \text{ if and only if } \forall \nu \in \{0,1\}^d : \frac{f_\omega(\mathbf{q})}{f_\nu(\mathbf{q})} \leq \frac{P(\nu)}{P(\omega)}. \tag{10}$$

But, in terms of the original variable $\mathbf{q}$:

$$\frac{f_\omega(\mathbf{q})}{f_\nu(\mathbf{q})} = \prod_{i=1}^d q_i^{\nu_i-\omega_i}(1-q_i)^{\omega_i-\nu_i}.$$

Note however that if $\omega_i = \nu_i$ then $q_i^{\nu_i-\omega_i}(1-q_i)^{\omega_i-\nu_i} = 1$. If instead $\omega_i \neq \nu_i$, there are only two possibilities. On the one hand, if $\omega_i = 0$ and $\nu_i = 1$, then

$$q_i^{\nu_i-\omega_i}(1-q_i)^{\omega_i-\nu_i} = \frac{q_i}{1-q_i} = \left(\frac{q_i}{1-q_i}\right)^{1-2\omega_i} = x_i.$$

On the other hand, if $\omega_i = 1$ and $\nu_i = 0$, then

$$q_i^{\nu_i-\omega_i}(1-q_i)^{\omega_i-\nu_i} = \frac{1-q_i}{q_i} = \left(\frac{q_i}{1-q_i}\right)^{1-2\omega_i} = x_i.$$

As a result:

$$\frac{f_\omega(\mathbf{q})}{f_\nu(\mathbf{q})} = \prod_{i:\nu_i\neq\omega_i} x_i,$$

which together with equation (10) implies the lemma. ◀

The special nature of the constraints in (9), suggests introducing the additional change of variables $\mathbf{x} \to \mathbf{y}$, with $\mathbf{y} = (y_1,\ldots,y_d)$ and $y_i := \ln(x_i)$. The following result is now immediate from the previous lemma.

▶ **Corollary 3.** *For a given $\omega \in \{0,1\}^d$, the transformation $\mathbf{q} \longrightarrow \mathbf{y}$ with $\mathbf{y} = (y_1,\ldots,y_d)$ and $y_i := (1-2\omega_i) \cdot \ln\left(\frac{q_i}{1-q_i}\right)$, is a bijection between $[0,1]^d$ and $[-\infty,\infty]^d$, and in terms of the variable $\mathbf{y}$:*

$$f_\omega(\mathbf{q}) = \prod_{i=1}^d (1+e^{y_i}). \tag{11}$$

Under this reparameterization, for each $\mathbf{q} \in (0,1)^d$:

$$\mathbf{q} \in \mathcal{Q}_\omega \text{ if and only if } \forall \nu \in \{0,1\}^d : \sum_{i:\,\nu_i\neq\omega_i} y_i \leq \ln\left(\frac{P(\nu)}{P(\omega)}\right), \tag{12}$$

where $\sum_{i:\,\nu_i\neq\omega_i} y_i := 0$ when $\nu = \omega$.

The characterization in equation (12) is not necessarily valid on the boundary of $[0,1]^d$ because, for some $\mathbf{q} \in \partial[0,1]^d$ and different $\omega, \nu \in \{0,1\}^d$, the summation on the right-hand side may be ill-posed due to the simultaneous occurrence of plus and negative infinity terms in the sum. Nevertheless, due to the continuity of f_ω in terms of the variables $\mathbf{q}$ and $\mathbf{y}$ (see equation (11)), if a solution to $\max_{\mathbf{q} \in \mathcal{Q}_\omega} f_\omega(\mathbf{q})$ lives on $\partial[0,1]^d$ then said solution is the limit of points in $\mathcal{Q}_\omega \cap (0,1)^d$. In particular, for each $\omega \in \{0,1\}^d$, the associated optimization problem in equation (7) may be restated in terms of the variable $\mathbf{y}$ as follows:

$$
\sup_{\mathbf{y} \in \mathbb{R}^d} \quad P(\omega) \prod_{i=1}^{d}(1 + e^{y_i})
$$

$$
\text{subject to} \quad \forall \nu \in \{0,1\}^d : \sum_{i:\nu_i \neq \omega_i} y_i \leq \ln\left(\frac{P(\nu)}{P(\omega)}\right). \tag{13}
$$

Each of these new optimization problems has various advantages – compared to the ones in (7). First, up to the factor $P(\omega)$, the objective function does not depend explicitly on ω. Second, the feasible region is a polyhedron [10, Chapter 8], which is a well-studied geometric object. And third, the objective function is monotonically increasing in each coordinate of $\mathbf{y}$; which implies that any solution must lie on the boundary of said polyhedron. We show how to exploit these properties in the next section.

2.2 Geometric Insights

In this section, we fix an outcome $\omega \in \{0,1\}^d$ and describe a combinatorial algorithm to solve the associated optimization problem in equation (13). Define

$$
\tilde{\mathcal{Q}}_\omega := \left\{ \mathbf{y} \in \mathbb{R}^d \,\middle|\, \sum_{i:\nu_i \neq \omega_i} y_i \leq \ln\left(\frac{P(\nu)}{P(\omega)}\right) \text{ for each } \nu \in \{0,1\}^d \right\},
$$

to denote the feasible region in (13).

In what follows, all vectors are represented as column vectors.

The following result is now immediate from the previous corollary.

▶ **Corollary 4.** *Assume that $P > 0$. For a given $\omega \in \{0,1\}^d$, let A_ω be the binary matrix of dimensions $(2^d - 1) \times d$ with entries $A_\omega(\nu, i) := [\![\nu_i \neq \omega_i]\!]$, for each $\nu \in \{0,1\}^d \setminus \{\omega\}$ and $i \in \{1, \ldots, d\}$. Furthermore, let $\mathbf{b}_\omega$ be a column vector of dimension $(2^d - 1)$ with entries $\mathbf{b}_\omega(\nu) := \log(P(\nu)/P(\omega))$, for each $\nu \in \{0,1\}^d \setminus \{\omega\}$. Then the feasible region $\tilde{\mathcal{Q}}_\omega$ corresponds to the set of $\mathbf{y} \in \mathbb{R}^d$ satisfying the coordinatewise inequalities:*

$$
A_\omega \, \mathbf{y} \leq \mathbf{b}_\omega. \tag{14}
$$

The above inequality characterizes $\tilde{\mathcal{Q}}_\omega$ as a non-empty convex polyhedron in $\mathbb{R}^d$. Recall, $\mathbf{y} \in \tilde{\mathcal{Q}}_\omega$ is called a *vertex* if there exists an invertible sub-matrix A'_ω of A_ω of dimensions $d \times d$ and a corresponding sub-vector $\mathbf{b}'_\omega$ of $\mathbf{b}_\omega$ of dimension d such that $A'_\omega \mathbf{y} = \mathbf{b}'_\omega$ [10, Chapter 8, equation (23)]. (The sub-matrix A'_ω and the sub-vector $\mathbf{b}'_\omega$ are associated with the same rows of A_ω and $\mathbf{b}_\omega$, respectively.)

▶ **Lemma 5.** *The polyhedron in equation (14) is pointed, i.e. it contains at least one vertex.*

Proof. For each $i \in \{1, \ldots, d\}$, let $\nu_i \in \{0, 1\}^d$ be such that $\nu_i(j) = \omega(j)$ for $j \neq i$, and $\nu_i(i) = 1 - \omega(i)$. Then the sub-matrix of A_ω associated with rows in the set $\{\nu_1, \ldots, \nu_d\}$ corresponds to the $(d \times d)$ identity matrix. As a result, the kernel of A_ω – which coincides exactly with the so-called "lineality space" of the polyhedron – is $\{0\}$, which implies that the polyhedron is pointed [10, Chapter 8, equations (6) and (23)]. ◀

In the language of polyhedral programming, a vertex is a zero-dimensional face. More generally, if $\mathbf{c} \in \mathbb{R}^d \setminus \{0\}$, $\delta \in \mathbb{R}$, and $G := \{\mathbf{y} \in \mathbb{R}^d \mid \mathbf{c}^t \mathbf{y} = \delta\}$ we say the affine hyperplane G is a *supporting hyperplane* of $\tilde{\mathcal{Q}}_\omega$ at the point $\mathbf{y} \in \tilde{\mathcal{Q}}_\omega$ if $\mathbf{y} \in G \cap \tilde{\mathcal{Q}}_\omega$ and $\tilde{\mathcal{Q}}_\omega$ is contained in one of the closed half-spaces bounded by G [7, p. 20]. The non-empty set $F := G \cap \tilde{\mathcal{Q}}_\omega$ is called a *face* of $\tilde{\mathcal{Q}}_\omega$. Equivalently, a face of $\tilde{\mathcal{Q}}_\omega$ is any set of the form $\{\mathbf{y} \in \tilde{\mathcal{Q}}_\omega \mid A'_\omega \mathbf{y} = \mathbf{b}'_\omega\}$, where A'_ω and $\mathbf{b}'_\omega$ are a sub-matrix and sub-vector associated with the same rows of A_ω and $\mathbf{b}_\omega$, respectively [7, Theorem 2.3.3]. (Here, A'_ω does not need to be a square matrix.) The dimension of a face F associated with the subsystem $A'_\omega \mathbf{y} = \mathbf{b}'_\omega$ is $d - \text{rank}(A'_\omega)$.

▶ **Corollary 6.** *If $\mathbf{y} \in \partial \tilde{\mathcal{Q}}_\omega$, the boundary of $\tilde{\mathcal{Q}}_\omega$, and $\mathbf{y}$ is not a vertex of $\tilde{\mathcal{Q}}_\omega$, then $\mathbf{y}$ lies in the relative interior of some positive-dimensional face of $\tilde{\mathcal{Q}}_\omega$. That is, there is a positive-dimensional face F and some $\epsilon > 0$ such that the intersection of the closed ϵ-ball around $\mathbf{y}$ and the affine hull of F is contained in F.*

Proof. First, $\tilde{\mathcal{Q}}_\omega$ equals the union of the relative interiors of its faces, which are disjoint [7, Corollary 2.3.7]. In particular:

$$\partial \tilde{\mathcal{Q}}_\omega = \bigsqcup_{\text{faces } F \subsetneq \tilde{\mathcal{Q}}_\omega} \text{relint}(F)$$

$$= \left(\bigsqcup_{\text{non-vertex faces } F \subsetneq \tilde{\mathcal{Q}}_\omega} \text{relint}(F) \right) \sqcup \left(\bigsqcup_{\text{vertices } v \in \tilde{\mathcal{Q}}_\omega} \{v\} \right),$$

where $\text{relint}(\cdot)$ denotes the relative interior, and $\sqcup$ denotes a disjoint union. Since a face coincides with its own relative interior if and only if it is a vertex, if $\mathbf{y} \in \partial \tilde{\mathcal{Q}}_\omega$ but $\mathbf{y}$ is not a vertex then $\mathbf{y}$ must lie in the relative interior of a unique positive-dimensional face. ◀

Next we address the optimization problem in equation (13) for a fixed $\omega \in \{0, 1\}^d$. Hereafter, we abuse notation slightly and define

$$f_\omega(\mathbf{y}) := \prod_{i=1}^d (1 + e^{y_i}),$$

to denote the reparameterized version of $f_\omega(\mathbf{q})$ in terms of the variable $\mathbf{y}$ (see Corollary 3). The following result rules out points in the relative interior of positive-dimensional faces of $\tilde{\mathcal{Q}}_\omega$ as maximizers of $f_\omega(\mathbf{y})$.

▶ **Lemma 7.** *Let $F \subset \tilde{\mathcal{Q}}_\omega$ denote a positive-dimensional face of $\tilde{\mathcal{Q}}_\omega$, and $\hat{\mathbf{y}}$ denote a point in the relative interior of F. Then $f_\omega(\hat{\mathbf{y}}) < \max_{\mathbf{y} \in \tilde{\mathcal{Q}}_\omega} f_\omega(\mathbf{y})$. More specifically:*
1. *If the gradient $\nabla f_\omega(\hat{\mathbf{y}})$ is not orthogonal to F, then f_ω can be strictly increased on F, that is, there is some $\hat{\mathbf{z}} \in F$ such that $f_\omega(\hat{\mathbf{z}}) > f_\omega(\hat{\mathbf{y}})$.*
2. *If the gradient $\nabla f_\omega(\hat{\mathbf{y}})$ is orthogonal to F, then $f_\omega(\hat{\mathbf{y}})$ is a local minimum on F.*

Proof. Clearly, f_ω has continuous partial derivatives of any order.

First observe that

$$\frac{\partial f_\omega}{\partial y_i}(\mathbf{y}) = e^{y_i} \prod_{j \neq i} (1 + e^{y_j}) = f_\omega(\mathbf{y}) \frac{e^{y_i}}{1 + e^{y_i}}.$$

Therefore, if $\mathbf{y} \to \gamma$ is the transformation defined as $\gamma = (\gamma_1, \ldots, \gamma_d)^t$, with $\gamma_i := \frac{e^{y_i}}{1+e^{y_i}}$, then

$$\nabla f_\omega(\mathbf{y}) = f_\omega(\mathbf{y}) \, \gamma,$$

which implies that $\nabla f_\omega(\mathbf{y}) \neq 0$, for all $\mathbf{y} \in \mathbb{R}^d$. In particular, if $\nabla f_\omega(\mathbf{y})$ is not orthogonal to F, a small perturbation in the direction of the projection of $\nabla f_\omega(\mathbf{y})$ onto F will increase f_ω. This shows the first statement in the lemma.

On the other hand:

$$\frac{\partial^2 f_\omega}{\partial y_i^2}(\mathbf{y}) = e^{y_i} \prod_{j \neq i} (1 + e^{y_j}) = f_\omega(\mathbf{y}) \, \gamma_i,$$

and for $i \neq j$:

$$\frac{\partial^2 f_\omega}{\partial y_j y_i}(\mathbf{y}) = e^{y_i} e^{y_j} \prod_{k \neq i,j} (1 + e^{y_k}) = f_\omega(\mathbf{y}) \, \gamma_i \gamma_j.$$

As a result, $\nabla^2 f_\omega(\mathbf{y})$, the Hessian matrix of f_ω at $\mathbf{y}$, admits the decomposition:

$$\nabla^2 f_\omega(\mathbf{y}) = f_\omega(\mathbf{y}) \left(\Gamma_1 + \Gamma_2 \right),$$

where $\Gamma_1 := \mathrm{diag}\big(\gamma_1(1 - \gamma_1), \ldots, \gamma_d(1 - \gamma_d) \big)$, and

$$\Gamma_2 := \begin{bmatrix} \gamma_1^2 & \gamma_1 \gamma_2 & \cdots & \gamma_1 \gamma_d \\ \gamma_2 \gamma_1 & \gamma_2^2 & \cdots & \gamma_2 \gamma_d \\ \vdots & \vdots & \ddots & \vdots \\ \gamma_d \gamma_1 & \gamma_d \gamma_2 & \cdots & \gamma_d^2 \end{bmatrix} = \gamma \gamma^t.$$

Because each $0 < \gamma_i < 1$, Γ_1 is strictly positive definite. Since Γ_2 is positive semidefinite, and $f_\omega(\mathbf{y}) > 0$ for all $\mathbf{y} \in \mathbb{R}^d$, $\nabla^2 f_\omega(\mathbf{y})$ is strictly positive definite regardless of $\mathbf{y}$. As a result, if $\nabla f_\omega(\hat{\mathbf{y}})$ is orthogonal to F, then $\hat{\mathbf{y}}$ is a local minimum of f_ω along F. This completes the proof of the lemma. ◀

Finally, combining Corollary 6 and Lemma 7, we obtain the following central result, which implies that the maxima in (13) can occur only occur among a finite number of well-characterized points in $\tilde{\mathcal{Q}}_\omega$.

▶ **Theorem 8.** *If $P > 0$ then, for each $\omega \in \{0,1\}^d$, the maximum $\max\limits_{\mathbf{y} \in \tilde{\mathcal{Q}}_\omega} f_\omega(\mathbf{y})$, can only occur at a vertex of $\tilde{\mathcal{Q}}_\omega$.*

3 Algorithms for $\lambda(P)$

Computing $\lambda(P)$ requires solving the optimization problem (13) for each of 2^d possible binary outcomes. As previously described, solving each optimization problem can be achieved by evaluating f_ω at each vertex of $\tilde{\mathcal{Q}}_\omega$, and the vertices can be found as unique solutions of invertible $(d \times d)$-subsystems $A'_\omega \mathbf{y} = \mathbf{b}'_\omega$. This motivates Algorithm 1, which computes $\lambda(P)$ by exploring square subsystems of $A_\omega \mathbf{y} \leq \mathbf{b}_\omega$ to find vertices, evaluating $f_\omega(\mathbf{y}^*)$ at each vertex $\mathbf{y}^*$ for each outcome ω, and returning the largest of these.

For each outcome ω, there are $\binom{2^d-1}{d}$ subsystems $A'_\omega \mathbf{y} = \mathbf{b}'_\omega$ of size $(d \times d)$ to check. For each subsystem $A'_\omega \mathbf{y} = \mathbf{b}'_\omega$, simple Gaussian elimination will find a unique solution, if it exists, in $O(d^3)$ time, and often terminates in less time if A'_ω is singular. If $\mathbf{y}'$ is a

■ **Algorithm 1** A naïve exact algorithm for $\lambda(P)$.

Require: $P > 0$
   $M \leftarrow 0$
   for $\omega \in \{0,1\}^d$ **do**
      $A_\omega \leftarrow \{\llbracket \omega_i \neq \nu_i \rrbracket_{i=1,\ldots,d}\}_{\nu \in \{0,1\}^d \setminus \{\omega\}}$
      $\mathbf{b}_\omega \leftarrow \{\log(\frac{P(\nu)}{P(\omega)})\}_{\nu \in \{0,1\}^d \setminus \{\omega\}}$
      for $\{\nu^{(1)}, \nu^{(2)}, \ldots, \nu^{(d)}\} \subset \{0,1\}^d \setminus \{\omega\}$ **do**
         $A'_\omega \leftarrow \{\llbracket \omega_i \neq \nu_i^{(j)} \rrbracket\}_{i,j=1,\ldots,d}$
         $\mathbf{b}'_\omega \leftarrow \{\log(\frac{P(\nu^{(j)})}{P(\omega)})\}_{j=1,\ldots,d}$
         if A'_ω is invertible **then**
            $\mathbf{y}^* \leftarrow (A'_\omega)^{-1} \mathbf{b}'_\omega$
            if $A_\omega \mathbf{y}^* \leq \mathbf{b}_\omega$ **then**
               $M \leftarrow M \vee f_\omega(\mathbf{y}^*)$
   return $\lambda(P) \leftarrow M$

---

unique solution to the square subsystem $A'_\omega \mathbf{y}' = \mathbf{b}'_\omega$, it takes $O(d2^d)$ operations to check that $\mathbf{y}'$ is feasible, i.e., $A_\omega \mathbf{y}' \leq \mathbf{b}_\omega$. If $\mathbf{y}'$ is infeasible, it often takes many fewer operations to confirm this.

Taking these operations together, and using the well-known bound on binomial coefficients, $\binom{n}{k} < (\frac{n \cdot e}{k})^k$, in the worst case there are $O\left(d^4 (\frac{2^{d+1} e}{d})^d\right)$ operations required to compute $\lambda(P)$. The memory required by this algorithm grows much less slowly, as $O(2^d)$, if square subsystems are iterated without loading every set of d indices into memory. This is common in standard combinatorial software like the `itertools` module in Python [4, Section 3.2]. In practice, we find that without any parallelization strategies and without supercomputing resources, it is feasible to compute $\lambda(P)$ for binary sources up to dimension $d = 6$ by naïvely searching for vertices.

We note that specialized algorithms to explore only those subsystems $A'_\omega \mathbf{y} = \mathbf{b}'_\omega$ which are invertible, and ignore singular subsystems, are still unlikely to allow computation of $\lambda(P)$ in very high dimensions. In fact, the number of invertible submatrices A'_ω of dimension d has previously been recognized as a noteworthy sequence [3]. This sequence is hard to compute explicitly, but appears to grow exponentially fast. In fact, there are approximately 2.52×10^{14} invertible subsystems in only 8 binary dimensions [12].

Specialized polyhedral programming algorithms may help to accelerate computation of $\lambda(P)$. For example, the vertex enumeration algorithm given in [1], runs in $O(d2^d V)$ time, where V is the number of vertices of $\tilde{\mathcal{Q}}_\omega$. The number of vertices is hard to characterize (it depends on $\mathbf{b}_\omega$), but based on simulation we believe it is typically much smaller than the number of invertible subsystems. We believe a pivoting method similar to [1] can be adapted to take advantage of A_ω's binary structure.

Some readers may note that each optimization program:

$$\max_{\mathbf{y} \in \mathbb{R}^d} \quad f_\omega(\mathbf{y})$$

$$\text{subject to} \quad A_\omega \mathbf{y} \leq \mathbf{b}_\omega$$

resembles a linear program. However, the objective function $f_\omega(\mathbf{y})$ is nonlinear, and therefore linear programming techniques such as Dantzig's simplex algorithm [2, Chapter 5] are not suitable. Moreover, positive definiteness of the Hessian derived in Lemma 7 implies that $f_\omega(\mathbf{y})$ is strictly convex. Although the feasible region is also convex, the fact that we seek to *maximize* $f_\omega(\mathbf{y})$ means most nonlinear convex programming techniques cannot be guaranteed to converge to true maxima.
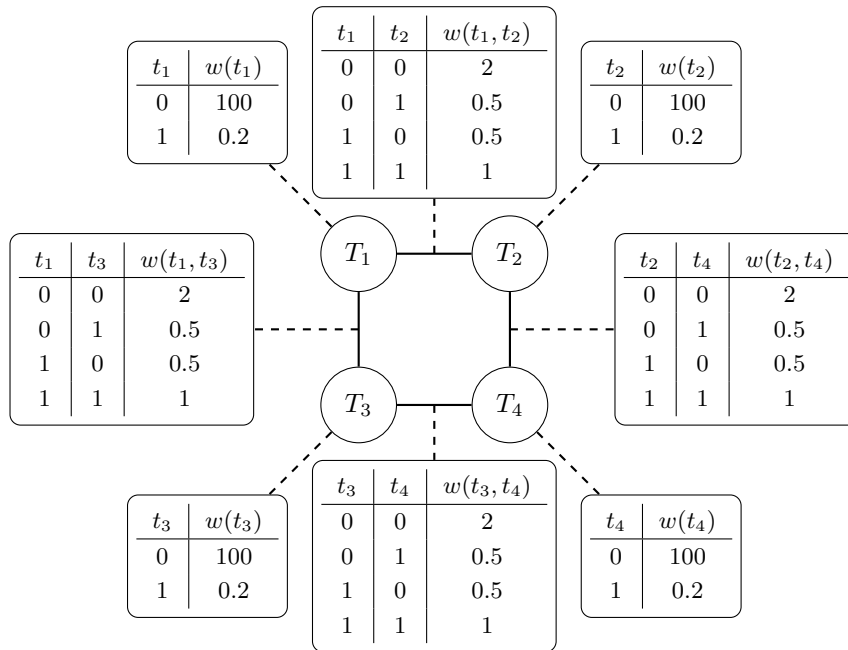
Figure 2 Markov network that models the interaction of four hypothetical patients that may or may not have tuberculosis. Patient i is healthy if $T_i = 0$, and infected if $T_i = 1$.

Due to the aforementioned difficulties in the combinatorial approach in high dimensions, we have also explored numerical approximation of each optimization program using nonlinear algorithms including sequential gradient-free linear approximation (COBYLA) [9], and sequential quadratic programming (SLSQP) [6]. These show some promise but tend to suffer from numerical instability in moderate to high dimensions (above $d = 5$ or so). However, because the structure of f_ω makes computing higher-order derivatives very straightforward, it may be possible to devise a specialized interior point method that makes approximating $\lambda(P)$ efficient even in higher dimensions.

4 Proof of Concept

Consider the Markov network [5] in Figure 2, borrowed from [11, Chapter 2]. In this setting, undirected edges represent interactions in a social network of four patients, each of whom may or may not have tuberculosis (represented as four Bernoulli random variables $T_1, \ldots, T_4$). Here, the complete subgraphs (cliques) of the Markov network are $\{T_1\}, \{T_2\}, \{T_3\}, \{T_4\}, \{T_1, T_2\}, \{T_1, T_3\}, \{T_2, T_4\}$, and $\{T_3, T_4\}$. To each clique C we associate a *factor* $w : \{0,1\}^{|C|} \to \mathbb{R}_+$, and to each configuration $(t_1, t_2, t_3, t_4) \in \{0,1\}^4$ of sick and healthy patients, we associate the probability:

$$P(t_1, t_2, t_3, t_4) \propto w(t_1) \cdot w(t_2) \cdot w(t_3) \cdot w(t_4) \cdot w(t_1, t_2) \cdot w(t_1, t_3) \cdot w(t_2, t_4) \cdot w(t_3, t_4).$$

This network reflects the intuition that, if one patient who has tuberculosis interacts with another, it is more likely for the latter to have tuberculosis. In fact, the joint distribution P of (T_1, T_2, T_3, T_4) is *exchangeable* (i.e. labels on the patients can be permuted without affecting the joint probability of their tuberculosis status). Using Algorithm 1, we find that $\lambda(P)$ is very close to one. We transform a vertex y^* which achieves $\lambda(P)$ back to a probability q^* (see Corollary 3) and find explicitly:

$$P = 0.999999 \cdot Be(0.000125) \otimes Be(0.000125) \otimes Be(0.000125) \otimes Be(0.000125) + 0.0000001 \cdot R,$$
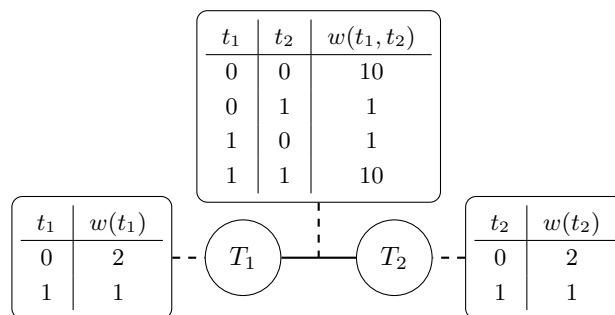
Figure 3 Markov network that models the interaction of two hypothetical patients which may or may not have tuberculosis. In this setting, the marginal probability of a patient being infected with tuberculosis is moderate ($\approx 22\%$), and the probability that exactly one of the two patients is infected is relatively low ($\approx 8\%$). This might be a realistic model for, e.g., two inmates sharing a cell in a prison with a tuberculosis outbreak.

where R is a residual probability distribution with low entropy (≈ 2 bits, compared to the uniform distribution over $\{0,1\}^4$, which has 4 bits of entropy). This means that, despite the dependence implied by the interactions, a large fraction of the time it will appear as though the patients are infected with tuberculosis independently, each with a very small probability of infection.

It is not always the case that a source represented by a probabilistic graphical model has a very large independent weight. Consider a simpler version of the previous Markov network, shown in Figure 3. In this case, a non-negligible fraction of the data produced by the source cannot be recapitulated by a model with independent marginal distributions. Let P denote the joint distribution of (T_1, T_2). Using Algorithm 1, we find that $\lambda(P) = 0.817$. Moreover,

$$P = 0.817 \cdot Be(0.048) \otimes Be(0.048) + 0.183 \cdot \delta_{(1,1)}.$$

That is, a large fraction of the time a realization of these two patients' tuberculosis states cannot be attributed to the largest independent component of P.

These two examples demonstrate how scientists and engineers may benefit from detecting a source's independent weight. If a source under study is known to have $\lambda(P) \approx 1$, even if the source fails a hypothesis test of independence, the modeler might save considerable complexity while still recapitulating most of the features of the source. In contrast, if a source has very low independent weight, the scientist could find meaningful mechanistic insights in the residual component, such as in the latter example, where a sample originates *either* from a hidden non-degenerate probability model with independent marginals or a deterministic one.

───── **References** ─────

1 David Avis and Komei Fukuda. A pivoting algorithm for convex hulls and vertex enumeration of arrangements and polyhedra. *Discrete & Computational Geometry*, 8(3):295–313, September 1992. `doi:10.1007/BF02293050`.

2 George B. Dantzig. *Linear Programming and Extensions*. United States Air Force Project RAND. The RAND Corporation, August 1963. URL: `http://www.rand.org/pubs/reports/R366.html`.

3 Yuval Dekel. Number of real regular n x n (0,1) matrices modulo rows permutations, 2003. In: The On-line Encyclopedia of Integer Sequences. URL: `https://oeis.org/A088389`.

4 Doug Hellmann. *The Python 3 Standard Library by Example.* Addison-Wesley Professional, 1 edition, 2017.

5 Daphne Koller and Nir Friedman. *Probabilistic Graphical Models: Principles and Techniques - Adaptive Computation and Machine Learning.* The MIT Press, 2009.

6 D. Kraft. *A Software Package for Sequential Quadratic Programming.* Deutsche Forschungs- und Versuchsanstalt für Luft- und Raumfahrt Köln: Forschungsbericht. Wiss. Berichtswesen d. DFVLR, 1988.

7 Dinh Luc. *Multiobjective linear programming: An Introduction.* Springer, January 2015. `doi:10.1007/978-3-319-21091-9`.

8 Antony Pearson and Manuel E. Lladser. On Contamination of Symbolic Datasets. (Submitted). `arXiv:2002.05592`.

9 M. Powell. A view of algorithms for optimization without derivatives. *Mathematics TODAY*, 43, January 2007.

10 Alexander Schrijver. *Theory of Linear and Integer Programming.* Number Vol. 75 in Wiley-Interscience Series in Discrete Mathematics and Optimization. Wiley, 1998. URL: `https://search-ebscohost-com.colorado.idm.oclc.org/login.aspx?direct= true&db=nlebk&AN=17885&site=ehost-live&scope=site`.

11 Ben Taskar and Lise Getoor. *Introduction to Statistical Relational Learning.* Adaptive Computation and Machine Learning. The MIT Press, 2007.

12 Miodrag Živković. Classification of small (0,1) matrices. *Linear Algebra and its Applications*, 414(1):310–346, 2006. `doi:10.1016/j.laa.2005.10.010`.