The Relaxed Maximum Entropy Distribution and its Application to Pattern Discovery

Sebastian Dalleiger Jilles Vreeken
CISPA Helmholtz Center for Information Security, Saarbrücken, Germany
{sebastian.dalleiger, jv}@cispa.de

***Abstract*—The maximum entropy principle uniquely identifies the distribution that models our knowledge about the data, but is otherwise maximally unbiased. As soon as we include non-trivial observations in our model, however, exact inference quickly becomes intractable. We propose a relaxation that permits efficient inference by *dynamically* factorizing the joint distribution into factors. In particular, we show that these factors are learnable from data and that it is consistent with standard maximum entropy distribution. Through an extensive set of experiments we show that the relaxation is scalable, approximates the vanilla distribution closely, allows for a classification that is as good, as well as results in a concise set of patterns.**

I. Introduction

The maximum entropy (maxent) principle allows us to uniquely identify that distribution which unbiasedly matches our observations from the data. It is therefore no surprise that this principle is useful in machine learning [23], [16], but, as it provides a statistically well-founded way to measure interestingness it is especially useful in data mining; not only can we use it to rank results for given prior beliefs [7], [12], discover small and non-redundant sets of informative patterns [17], but also to decompose data [5].

The computational complexity of inferring the expectation is, however, affected by the observations that we incorporate into the model. As long as we are only interested in individual features, the distribution factorizes into the product of their frequencies and is therefore simple to infer. But whenever we incorporate dependencies between attributes, such as co-occurrence frequencies of features, the inference becomes PP-hard [20], which is intractable in practice. One way to circumvent this is to factorize the distribution according to the independences in our statistic S. That is, if those are given as a set S, we can partition this set into independent subsets S_i, and infer the maxent distribution p_i^* for each S_i independently. This factorization is only faithful to the data, if we these correspond to truly independent sets attributes. To ensure an efficient inference, existing methods [17], [5] restrict these sets by disregarding dependent attributes as independent, which can lead to an unfaithful representation of the data.

In this paper we take a different approach, and rather start from the observation that not everything we know is always equally relevant. We propose a relaxation to the inference of the expectation that uses only the subsets of S that is most relevant to the query x, instead of using all available information for each query. Hence, we use a different distribution, depending on what we query for. In other words, rather than enforcing one static factorization for all queries, we consider different, dynamic factorizations of S depending on the query.

We show that our relaxation is learnable, is consistent with the maximum entropy principle, and its relationship to discovering pattern from data. We show that our approach allows us to consider almost arbitrarily large sets, approximate complex ground truth distributions better than the more constrained existing solutions, while at the same time being faster. Moreover, through extensive experiments on both synthetic and real datasets, we also show that our relaxed distribution approximates the vanilla maximum entropy distribution well in the area of multi-class classification and pattern set mining.

In sum, our main contributions are that we **(I)** introduce the relaxed maximum entropy distribution, and provide a practical realization, **(II)** show how factors of the model relate to patterns in data, **(III)** show via a wide range of experiments that the distribution approximates the vanilla maximum entropy, classifies as well and discovers concise pattern sets.

II. Related Work

The maximum entropy principle was proposed Jaynes [10], [11] as a general approach to choosing probability distributions. The theoretical foundations were further developed by among others Csiszár [4], who showed that the maxent distribution minimizes the Kullback-Leibler divergence to the uniform distribution, has an exponential form, and that its maximization is convex.

For large event spaces Ω, the main bottleneck is the computation of expectations. Tatti [20] showed that the inference of in the case of itemset frequencies is PP-hard, but we do not always have to infer the full distribution, as we can also approximate it. Barron and Sheu [1] show that under moment constraints this is possible in terms of exponential families and basis function expansion using e.g. polynomials. Bierig and Chernov [2] studied Monte Carlo methods to approximate the distribution. Singh and Vishnoi [18] recently established its equivalence with general counting problems and showed that we can approximate counts in order to approximate the distribution. These translate to noisy and therewith relaxed moment constraints. Dudík et al. [8] presents a maximum entropy problem with relaxed constraints that are generalized regularization measure in their dual form. For possibly noisy generalized constraints, Sutter et al. [19] proposed an approximation strategy for the dual of the maximum entropy problem, by means of a fast gradient approximation.

Another approach to the approximate inference is the factorization of the distribution into assumed-to-be independent factors [17] that are limited in their modeling power to force efficiency. Despite its limitations, this factorization strategy has been successfully used for discovering concise and non-redundant pattern sets [17], sampling of realistic categorical datasets [22], and for the pattern compositions [5].

III. Preliminaries

Before we introduce the concept of the maximum entropy distribution, we start with notation. We consider binary data $\mathcal{X}$ of n i.i.d. rows over d attributes in $\mathcal{I}$, each sampled from the set $\Omega = 2^{\mathcal{I}}$ of all events. We write 2^A for the powerset of any finite set A and $\binom{A}{2}$ denotes the set of all pairs $a \neq b$ from A. The union of two disjoint sets A and B is $A \dot\cup B$. For any $n \in \mathbb{N}$ we write $[n] = \{1, 2, \ldots, n\}$. The indicator function is $\mathbf{1}$. All logarithms are to base 2, and we let $0 \log 0 = 0$.

A. The Maximum Entropy Probability Distribution

In general, we are interested in a distribution that are defined over Ω, and that models a set $S \subseteq \Omega$ of observed events. Any event $x \in \Omega$ has an associated expected observed frequency of $q(x) = |\{y \in \mathcal{X} \mid x \subseteq y\}| \,/\, |\mathcal{X}|$. We want a distribution that matches these expectations, i.e. $\mathrm{E}\,[x] = q(x)$ for any $x \in S$. For a given set observation $S \subseteq \Omega$ and the corresponding q, we define the set of feasible distributions as the polytope $\mathcal{P}_S$

$$\{f \in \Omega \to [0,1] \,|\, \mathrm{E}_f\,[x] = q(x)\ \forall x \in S\,, \textstyle\sum f = 1\}\ ,$$

that contains all, infinitely many distributions satisfying the moment constraints, given that the observations are consistent. Together, S and q are the statistics of the maximum entropy distribution. This raises the problem of choosing a distribution. A natural choice is a distribution that does not introduce additional assumptions beyond the information that S and q specifies. From an information theoretic point of view, additional assumptions correspond to additional information. We can measure the amount of information in a distribution using Shannon entropy, $H(p) = -\sum_x p(x) \log p(x)$. The lower the information content of a distribution p, the higher its entropy. We can uniquely identify the feasible distribution that makes the least additional assumptions as the one with the highest entropy [4]

$$f \equiv \operatorname*{arg\,max}_{f \in \mathcal{P}_S}\ H(f)\ , \tag{1}$$

which is known as the Principle of Maximum Entropy [11].

In general this does not immediately provide a family of distributions to use. In our case, however, as the constraints of $\mathcal{P}_S$ are linear, we know that f over $x \in \Omega$ has the form

$$f(x \mid S) = \theta_0 \prod_{y_i \in S} \theta_i^{\mathbf{1}[y_i \subseteq x]}\ ,$$

for appropriately chosen coefficients $\theta \in \mathbb{R}^{|S|+1}$ [4]. Conveniently, optimizing θ is a convex problem, and hence we can employ standard convex optimizers such as iterative scaling [6]. We are specifically interested in inferring the expected frequency p^* for arbitrary $x \in \Omega$—the frequencies of $x \in S$ are given, after all. To infer p^* we have to sum the probabilities of every possible $y \in \Omega$ that supports x,

$$\mathrm{E}_f\,[x \mid S] = \sum_{y \in \Omega} f(y \mid S)\, \mathbf{1}\,[x \subseteq y]\ . \tag{2}$$

IV. Relaxation

We start with the inference of the expectation of the maximum entropy distribution. The straightforward inference of p^* involves an exponential number of terms in the sum. It is easy to see that many of these terms evaluate to the equivalent probabilities and can be used to partition Ω into equivalence classes $\Omega_{/\sim}$ with $x \sim y \iff f(x \mid S) = f(y \mid S)$ for $x, y \in \Omega$, such that the expectation is the weighted sum

$$p^*(x \mid S) = \mathrm{E}_f\,[x \mid S] \equiv \sum_{\substack{[y] \in \Omega_{/\sim} \\ x \subseteq y}} |[y]|\, f(y \mid S)$$

over these classes. Mampaey et al. showed how to create the set $\{[y] \in \Omega_{/\sim} \mid x \subseteq y\}$ of equivalence classes that support x and their weights from S, in a way that the size scales exponentially only in S instead of $\mathcal{I}$ [17]. If S is sufficiently large, the inference is, however, still intractable and the question arises, how we can reduce the complexity without constraining S. If there exists a valid factorization $\prod p_i^*$ of p^* into independent factors p_i^*, both terms are equivalent. Conversely, we will not lose information by partitioning S into independent sets $S_i \subseteq S$. Whenever we can do so, the complexity of each factor $p_i^*(\,\cdot \mid S_i)$ scales only in S_i, and if the sizes of these $S_i \subseteq S$ are now considerably smaller than S, we significantly reduce the complexity without loss.

Example 1. For example, the set $\mathcal{I}$ consists of letters from a to f and S is $\{abc, cd, de, df, ef\}$. If we know that the letters abc are independent of the rest, we can factorize S into $S_1 = \{abc\}$ and $S_2 = \{de, ef, df\}$ without information loss. The inference of the frequency of the pair ab consists of marginalizing out c of factor S_1 alone. But if S_1 and S_2 are not independent, for example if the pair cd is part of the model, then we have to marginalize out c, cd, de, ef and df, which leads to the sum over 2^5 combinations.

The inference of the expectation $p^*(x \mid S)$ becomes the product $\prod_{S_i} p^*(x \cap s_i \mid S_i)$ of individual maxent factors, where the set $s_i = \cup S_i$ contains the elements that are associated through S_i. In the following, we generalize this observation in terms of a factorization oracle φ.

Definition 1. For a given $\varphi \in \Omega \to 2^S$ that is provided with statistic $S \subseteq \Omega$, the *generalized factorization* is

$$\tilde{p}(x \mid S) = \prod_{S_i \in \varphi(x)} p^*(x \cap s_i \mid S_i)\ ,$$

where the factors $p^*(\,\cdot \mid S_i)$ have maximum entropy subject to constraints imposed by S_i (Eq. (1) and (2))

Example 2 (Static Factorization)**.** In this example we assume that the statistic S is given and factorization of p^* is fixed. That

(a) $\varphi_{\text{static}}(cd)$ (b) $\varphi(cd)$ (c) $\varphi_{\text{static}}(abcdef)$ (d) $\varphi(abcdef)$

Figure 1: **Example Graphical Model** For $S = \{abc, cd, de, df, ef\}$, denoted by the dotted edges, we visualize four possible factorizations of $\tilde{p}$ using φ_{static} and φ for queries cd and $abcdef$ (blue circles). By using the static factorization in Fig. 1c and in Fig. 1a, we have to use of all available information in S, visualized by the fully connected component (solid lines). To reduce the inference complexity, we use a relaxed factorization φ that omits the crossed-out links from S. For cd in Fig. 1a, the relaxed inference is correct, since $cd \in S$. However, the relaxation of $abc\ def$ in Fig. 1b trades inference complexity with information loss, because we ignore the link between c and d, and hence we disregard potential co-occurrences.

means, we have access to the set of p^*-independent statistics $\{S_i\}_i$, such that $S = \dot{\bigcup}_i S_i$ from which the *static factorizer*

$$\varphi_{\text{static}}(x) = \{S_i \in \Phi | x \cap s_i \neq \emptyset\}$$

follows immediately. In theory, if Φ truly models the independencies of the ground truth distribution, then using φ_{static} is optimal. In practice, however, modeling the true factorization can pose a significant problem: the complexity of inferring a single maxent factor is still exponential in the size of S_i. To circumvent this issue we have to drastically limit the size of each S_i to be no greater than, say a user defined $\beta \in \mathbb{N}$.

The *static* factorization lead to the problem that we have to choose either between a tractable inference complexity or a rich modeling of the data. For example, consider the static factorization Φ that consists of the two independent factors $\{abc\}$ and $\{de, df, ef\}$. If we model an association between c and d, we introduce a statistical dependency between the two factors hence Φ would become $\{\{abc, cd, de, df, ef\}\}$. But since the size of this factor exceeds the budget of $\beta = 4$, we are therefore prohibited from modeling the dependency cd. In the following, we introduce a relaxed, more flexible factorizer that levitates this choice from us.

Example 3. In Fig. 1 we show the graphical representation of $\tilde{p}$ for our example set $S = \{abc, cd, de, df, ef\}$. The dotted edges visualize associated features. In this example we show possible factorizations of the two queries $abcdef$ and cd using both static factorization (Fig. 1c) resp. (1a) and relaxed factorization (Fig. 1d) resp. (1b). The static factorization involves the complete graph and therefore, the graph is connected (solid lines) and the inference is quite complex. If we deliberately ignore dependencies, we can, however, partition the graph into less complex-to-infer clique graphs by cutting out edges (the crossed-out dotted edges).

Dynamic Factorization: In the example above, we *dynamically* adapted the factorization of $\tilde{p}$ to trade inference complexity with information loss for the queries $x \in \Omega$. To formalize this idea, we assume that not all information in statistics S is necessarily worthwhile to include in the factorization of each $x \in \Omega$. By this we mean that there is a subset of S that contributes very little to no information to the expected frequency of x. Similar to the example, by only paying with a small information loss, we reduce the inference complexity exponentially. More formally, we want the factorization of $\tilde{p}$ with the smallest information loss

$$\varphi_{\text{dynamic}}(x) = \operatorname*{arg\,min}_{\Phi \subseteq 2^S} D(p^* \| \tilde{p}_\Phi) \text{ s.t. } \mathrm{C}(\Phi) < \beta\,, \quad (3)$$

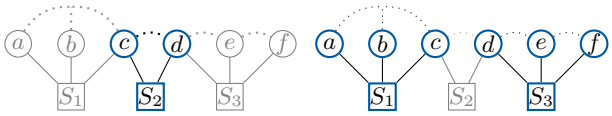
while being tractable to infer for a $\beta \in \mathbb{N}$, where $D(q\|p)$ is the Kullback-Leibler divergence $\sum_{x\in\Omega} q_x \log q_x/p_x$ between q and p and $\mathrm{C}(\Phi)$ is the inference complexity $\sum_{S_i \in \Phi} 2^{|S_i|}$. Directly solving Eq. (3) has, however, three obvious drawbacks: Firstly, (i) the number of possible factorizations of $\tilde{p}$ is exponential in the size of S. Secondly, (ii) each factor is supposed to maximize its entropy. Thirdly, (iii) D is computationally costly. Together, this has to be done for at least each $x \in \mathcal{X}$ and hence, it is not a very practical factorizer.

In the following, we introduce an alternative that is inspired by the above, but that does not suffer from these drawbacks. Crucially, our φ must correctly factorize any x. That is, there are no two factors S_i and S_j in $\varphi(x)$ that compute the expectation of the same subset of x, i.e. $x = \dot{\bigcup}_{S_i \in \varphi(x)} x \cap s_i$. Additionally, we want to constrain φ to be efficient to compute, say $\varphi \in \mathcal{O}(\text{poly}(|\Phi|))$ for some set Φ (i, iii) and all the factors have to be efficiently accessible in poly-time (ii). From the latter we conclude that each factor $S_i \in \varphi(x)$ that is used for any x has to be known beforehand, since maximizing the entropy of a single factor has a complexity that is exponential in S_i. Therefore, it is necessary that φ selects an appropriate set of factors from a known, predetermined set Φ of *elementary factors* that are each maximally entropic.

Definition 2. For a given set Φ of elementary factors and provided with a function $c \in \Omega \to \mathbb{R}$, let $\varphi \in \Omega \to 2^\Phi$. For each $x \in \Omega$, we construct a factorization from Φ that

$$\varphi(x) = \operatorname*{arg\,min}_{A \subseteq \Phi} c(A) \text{ s.t. } \dot{\bigcup_{A_i \in A}} x \cap a_i = x\,.$$

To limit the number of omitted dependencies, we are specifically interested in using the smallest number factors that together cover any $x \in \Omega$, i.e. the *minimal sufficient coverage* of x, thus our objective $c(A)$ is $|A|$. By using this objective function, the problem of computing a factorization using φ is



(a) $\varphi(cd) = \{S_2\}$ (b) $\varphi(abcdef) = \{S_1, S_3\}$

Figure 2: **Example Factor Graph** For the given elementary factors $\Phi = \{\{abc\}, \{cd\}, \{def\}\}$, we show a factor graph representations of $\tilde{p}$ for inferring cd (left) and $abcdef$ (right). We represent each elementary factor $S_i \in \Phi$ as a square and highlight the factors that are in $\varphi(x)$.

in fact a variant of the minimal exact set cover [13]. In contrast to a static factorization, we allow for elementary factors that are not necessarily disjoint. This means, we gained additional flexibility and incorporate a richer statistic S in comparison to the more constraints φ_{static}.

Example 4. In addition to S from our running example, we are now also given a set Φ of elementary factors $\{\{abc\}, \{cd\}, \{def\}\}$. Fig. 2 we depict a factor graph representations of $\tilde{p}$ and visualize possible factorizations of two different queries. Again, we trade the inference complexity with the information loss, however this time not by removing arbitrary edges from the graph, but by selecting a subset of elementary factors (squares) from Φ. In this figure we show that elementary factors are not necessarily disjoint. The factors in $\varphi(x)$, however, are always disjoint for any x.

To overcome issue (i) and (iii), we take the greedy approach to set cover. Hence, the worst case complexity, $\mathcal{O}(2^\beta k^2)$, of inferring the relaxed expectation is bounded by a term that is exponential in the maximally allowed factor size β. Disregarding set cover, the complexity is the same as of φ_{static}.

a) Estimator: The factorizer φ requires a set of underlying factors $\Phi \subseteq 2^\Omega$ to construct a factorization from. Now, we find such a set that minimizes the divergence between p^* and the relaxation $\tilde{p}$, such that the complexity is bounded.

Problem 1. For a given budget $\beta \in \mathbb{N}$ our problem is to

$$\begin{aligned} \underset{\Phi \subseteq 2^\Omega}{\text{minimize}} \quad & D(p^* \| \, \tilde{p}) \\ \text{subject to} \quad & |S_i| \leq \beta && \forall S_i \in \Phi \\ & f_{S_i} \leftarrow \text{Eq. (1) w.r.t. } S_i && \forall S_i \in \Phi \\ & \tilde{p}(x) = q(x) && \forall x \in \cup\Phi \end{aligned} \tag{4}$$

where D is the Kullback-Leibler divergence between the given target reference distribution p^* and the relaxation $\tilde{p}$. Every f_{S_i} is maximizing the marginal entropy according to Eq. (1). For this, the set $S = \cup\Phi$ is the union of all elementary factors.

To solve this problem, we start with the mild assumption that the support of $\tilde{p}$ subsumes the support of p^*, i.e. $\operatorname{supp} p^* \subseteq \operatorname{supp} \tilde{p}$. Then, we know that the divergence $D(p^* \| \tilde{p}) < \infty$ is finite [15] and therefore a solution must exist. First, we show that there exist an approximation to p^* that models the latent $S^* = \cup\Phi^*$. Afterwards, we show that this can be achieved efficiently by using empirical independencies.

Lemma 1. The probability that the divergence is sufficiently small, converges to 1, i.e. $\Pr[D(p^* \| \tilde{p}) < \epsilon] \to 1$ where $\epsilon \to 0$ for $|\Phi| \to |S^*|$.

Proof. Let $Y \sim p^*$ and $X \sim \tilde{p}$ two random variables. We know that if the conditional entropy $H(Y|X)$ is 0, the divergence $D(p^* \| \tilde{p})$ is minimal. We know from Fano's inequality [3] that the conditional entropy $H(Y \mid X)/\log|\Omega| < \Pr[E]$ of $Y|X$ is bounded from above by the error probability $\Pr[E]$ for random variable $E = X \neq Y$. This means, as long as $\Pr[E]$ converges to 0, the conditional entropy converges to 0 and hence our problem is learnable according to Lem. 1. Let $S_y = \{y\}$ be an elementary factor of y and let $\Phi_y \leftarrow \Phi \cup \{y\}$. From this it the distribution $p^*_y \in \mathcal{P}_{S_y}$ follows. For a φ that selects from Φ_y, we therefore, know that $\tilde{p}(y)$ matches $p^*(y)$ (*cf.* Eq. (4)). Hence, the probability $\Pr[E]$ shrinks and thus, e.g. the set $\Phi = \{\{x\}_{x \in S^*}\}$ is such a sequence of factors for which $\Pr[E]$ converges asymptotically, given that the moment constraints are consistent. □

Now, we know that the problem is in principle learnable, but the distribution p^* is still unknown to us and therefore, we want to make use of an empirical estimate instead.

Lemma 2. For a given set $\mathcal{X}$ of n samples $\{x_i\}_{i \in [n]}$ from p^*, the empirical estimator $\widehat{D}^n$ of Eq. (1) converges asymptotically to $D(q\|\tilde{p})$, i.e. $\lim_{n\to\infty} \widehat{D}^n(p^*\|\tilde{p}) \to D(q\|\tilde{p})$.

Proof. We assume $\operatorname{supp} q \subseteq \operatorname{supp} p^*$. We write $D(p^*\|\tilde{p}) = D(p^*\|q) + D(q\|\tilde{p})$ by using the information projection [4]. Since $\mathcal{X} \sim p^*$, we know that asymptotically $\lim_{n\to\infty} \widehat{D}^n(p^*\|q) \to 0$ holds. Thus, it is sufficient to show $\lim_{n\to\infty} \widehat{D}^n(q\|\tilde{p}) \to D(q\|\tilde{p})$, which is trivially true. □

Even by using the empirical estimator above, solving the problem directly involves a very large 2^{2^Ω} search space. To overcome this, we considerably reduce the search space, for which we show that we can limit it to a subset of Ω of statistically dependent elements in data $\mathcal{X}$. To do so, we first formalize what we mean by dependencies. We say that $x \in \Omega$ is *conditionally independent* of $y \in \Omega$, if

$$x \perp\!\!\!\perp y \mid \Phi^* \iff \nexists S_j \in \Phi^* : x \subseteq s_j \wedge y \subseteq s_j \; .$$

By definition there is no single maximum entropy factor S_i in the assumed-to-be given true factorization Φ^*, that contains both $x \in \Omega$ and $y \in \Omega$ that are statistically independent in $\mathcal{X}$.

Lemma 3. For a given set $S^* \subseteq \Omega$ that contains all statistically dependent sets of elements $x \in \Omega$, there are no factors S_i in Φ^* that contain $x \notin S^*$.

Proof. Assume otherwise, that is there is a $x = a \cup b \in S^*$ for which $a \perp\!\!\!\perp b \mid \Phi^*$. Thus, $\nexists S_{ab} \in \Phi^*$ with $a, b \in S_{ab}$. Hence, $p^*(a \cup b) = p^*(a \mid S_a)p^*(b \mid S_b)$. By definition $q(a \cup b) \neq q(a)q(b)$ is however true and from $\lim_{n\to\infty} q^{(n)} \to p^*$ follows the contradiction. The other direction follows similarly. □

b) Creating Factors: To conclude the above, instead of minimizing Eq. (4) it suffices to construct a factorization Φ that minimizes the divergence (Lem. 1, 3) directly from $\mathcal{X}$ (Lem. 2). We assume that $S \subseteq \Omega$ is given to us and from that S we now want to create a factorization. The trivial solution of letting $\Phi = 2^S$ is impractical, as many factors are not used by φ. Instead, we create Φ in an iterative process that is aware of φ. For this, we iteratively insert a new factor into Φ for each pattern $x \in S$. Starting with one pattern x, we want to create a new factor that minimizes the divergence between q and $\tilde{p}$, *cf.* Eq. (1). Firstly, we know from Eq. (4) that the estimate $\tilde{p}(x)$ exactly match the observation $q(x)$ if x is present in S_x and therefore minimizes the point-divergence. Secondly, to ensure that $\tilde{p}$ is as informative as without S_x, we include all the previously used information to infer $\tilde{p}(x)$ into that factor. That information was provided by $\varphi(x)$, which in total leads to the factor $S_x = \varphi(x) \cup \{x\}$ that is tailored towards x. This S_x can however exceed the budget β. To counteract this, we relax the new factor by minimizing

$$S_x \leftarrow \operatorname*{arg\,min}_{S \subseteq S_x, |S| \leq \beta} D_{\Omega_x}(q \| p^*_S) \tag{5}$$

the divergence, subject to our budget β, where D_{Ω_i} is the Kullback-Leibler divergence with respect to the partition $\Omega_i \subseteq \Omega$ that factor S_x supports. Because β is small in practice and the divergences is submodular [14], we solve Eq. (5) greedily. Finally, we insert this relaxed factor into the set of factors Φ.

V. Experiments

In the experiments, we evaluate the factorization on synthetic, as well as 58 real-world datasets that together span a wide variety of domains, sizes, and dimensionalities. We implemented our method in C++, ran experiments on an Intel Xeon E5-2643 CPU, and report wall clock time. We provide the source code, datasets, synthetic dataset generator, and additional information needed for reproducibility.[1]

A. Synthetic Data

We start with verifying the factorization on a known ground-truth, for which we generate synthetic data. To do so, we randomly sample the set S^* of 2 048 patterns over 256 and corresponding frequency q. In each of the 50 trials, we independently draw 4 096 points from q under an additive noise term of 5%. Even though we have access to the true pattern sets S^*, the computation of the true likelihood is intractable for p^*. Therefore, we compare the divergence between p^* or $\tilde{p}$ and empirical frequencies q for S^*. In Fig. 3 for a budget of up to 12, the vanilla distribution p^* cannot model more than 40–80 patterns from S^*, without considerable runtime cost. For the same budget, we observe the exponential runtime growth of the relaxation significantly later at around 1 800 patterns.

On the left, we see that for smaller β the statically factorized p^* converges early, since it has no factors with a remaining budget left. In these experiments, $\tilde{p}$ can, however, incorporate the full ground truth set of patterns S^* and hence, is capable of reaching the minimal divergences of 0. In the case of $\tilde{p}$, we can further observe that a strict limitation of the budget does not have a significant impact on the divergence between. Even for the smallest budget, the relaxation $\tilde{p}$ is overall less constrained than p^* for a larger β.

[1] eda.mmci.uni-saarland.de/reaper

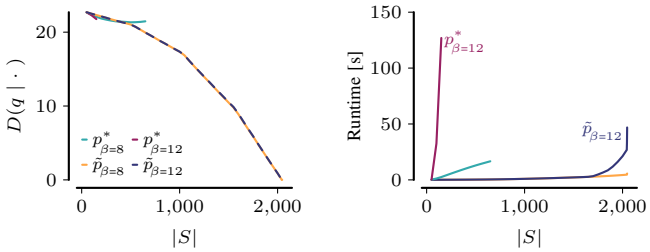


Figure 3: **Scalability on Synthetic Data** We show the divergence (left) of $\tilde{p}$ and p^* to frequencies q and the inference time (right) for different β and increasing model size $|S|$.

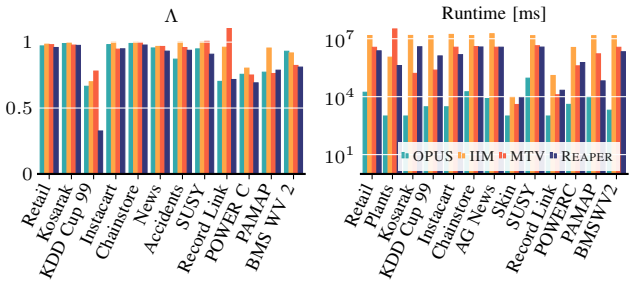


Figure 4: **Pattern Mining** We show the BIC ratio in terms of $\tilde{p}$ (lower is better) for patterns discovered by OPUS, IIM, MTV and REAPER, and their runtime (milliseconds, log scale).

B. Real-World Datasets

Now, we test $\tilde{p}$ on real-world datasets. For a fair comparison we measure the ratio $\Lambda = \ell(\Phi)/\ell(\{\{x \in \mathcal{I}\}\})$ of the BIC of Φ to the BIC of independence model (resp. for p^*)

$$\ell(\hat{\Phi}) = \log|\mathcal{X}| \sum_{S_i \in \hat{\Phi}} |S_i| \,/2 - \sum_{x \in \mathcal{X}} \log \tilde{p}(x; \hat{\Phi}) \,,$$

and the time to compute this score. BIC scores that are close, generally speaking mean that the corresponding distributions model the data similarly well.

a) Pattern Mining: We start with applying the relaxation to pattern mining. For this, we have adapted DESC [5], whose distribution we replace with $\tilde{p}$. To account for the usage of a pattern x, we scale DESC's heuristic h by the usage. That is, for each candidate x, we create a factor S_x and count the number of times it is used by the factorizer φ_x, which has access to $\Phi \cup S_x$ (*cf.* Eq. (5)), i.e. the heuristic becomes

$$h(x) \cdot |\{y \in \mathcal{X} \mid S_x \in \varphi_x(y)\}| \,.$$

We call the result REAPER, the relaxed entropy accelerated pattern miner. We compare this method to the pattern miner OPUS [21], IIM [9] and MTV [17], by measuring the BIC

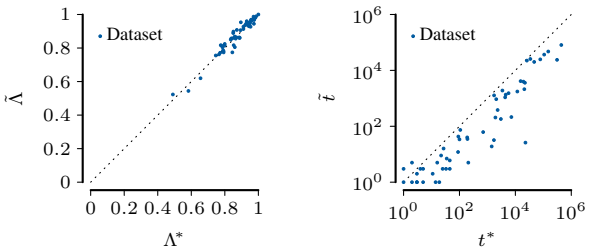


Figure 5: **Approximation** For 55 real-world datasets, we show the BIC ratios Λ and the time t (milliseconds, log scale) to compute these, for the equally parameterized models $\tilde{p}$ and p^*.

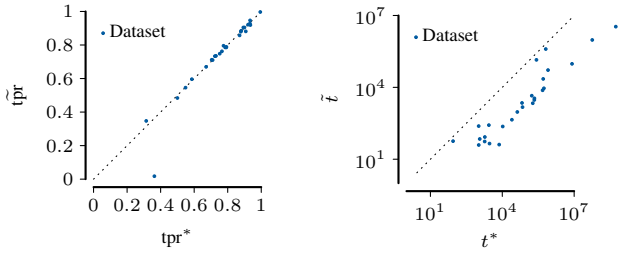


Figure 6: **Classification** On 25 datasets, we cross-validate the $\tilde{p}$ or p^* based classifier and show the true positive rate and the classification time t (microseconds, log scale) of the test data.

ratio. For OPUS we choose k to be the same as the number of REAPER's discoveries. In Fig. 4 we quantify the BIC ratios of all methods in terms of the relaxed distribution $\tilde{p}$. In this figure, we see that REAPER results in patterns which in most cases exhibit a higher likelihood than the competitors.

b) Approximation: To study how well the relaxation fits the data, we compare the likelihoods of $\tilde{p}$ to p^*. For this, in each experiment, we discover and fix the set S using REAPER and use that set to parameterize these distributions, which we give a budget β of 12. In Fig. 5 we show the BIC scores of $\tilde{p}$ and p^* and there we can see that both distributions model the data similarly well (diagonal), however, we observe that the relaxation is almost always significantly faster to compute.

c) Classification: As an additional metric, we compare the binary or multi-class prediction accuracy of $\tilde{p}$ and p^* on 25 labeled datasets. For this, we introduce the Bayes classifier

$$\arg\max_i \tilde{p}_i(x \mid \Phi_i)\,,$$

that assigns the likeliest label according to the distributions $\tilde{p}_i$ for the classes $i \in [m]$ (resp. p^*). We perform a 5-fold stratified cross-validation. Per fold, we discover the set of factors Φ_i using REAPER for each class i in the randomly sampled training (80%) data (resp. DESC) and use Φ_i to parameterize the distribution $\tilde{p}_i$ of class i (resp. p_i^*). Then, we compute the true positive rate (tpr) on the remaining test data (20%) using the Bayes classifier w.r.t. $\tilde{p}$ or p^*. In Fig. 6 we report the average tpr and average test time, which demonstrates that the relaxation $\tilde{p}$ classifies as well, however, in much less time.

VI. DISCUSSION & CONCLUSION

We introduced the relaxed maximum entropy distribution based on a generalized, dynamic factorization that can trade inference complexity with information loss, and results in a distribution that has higher statistical modeling power than exact models. We provided a practical instantiation that builds on set cover principles, applied this relaxation to classification or pattern mining and showed experimentally that it works well in practice. There are multiple open research questions, such as a factorization that considers the complexity of a factor or applying this principle to approximate graph counting.

REFERENCES

[1] A. R. Barron and C.-H. Sheu. Approximation of density functions by sequences of exponential families. *Annals Stat.*, 19(3):1347–1369, 1991.

[2] C. Bierig and A. Chernov. Approximation of probability density functions by the multilevel monte carlo maximum entropy method. *J. Comput. Phys.*, 314:661–681, 2016.

[3] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. John Wiley & Sons, 2 edition, 2012.

[4] I. Csiszár. I-divergence geometry of probability distributions and minimization problems. *Annals Prob.*, 3(1):146–158, 1975.

[5] S. Dalleiger and J. Vreeken. Explainable data decompositions. In *AAAI*, pages 3709–3716. AAAI Press, 2020.

[6] J. Darroch and D. Ratcliff. Generalized iterative scaling for log-linear models. *Annals Math. Stat.*, 43(5):1470–1480, 1972.

[7] T. De Bie. Maximum entropy models and subjective interestingness: an application to tiles in binary databases. *Data Min. Knowl. Disc.*, 23(3):407–446, 2011.

[8] M. Dudík, S. J. Phillips, and R. E. Schapire. Maximum entropy density estimation with generalized regularization and an application to species distribution modeling. *JMLR*, 8(6), 2007.

[9] J. Fowkes and C. Sutton. A bayesian network model for interesting itemsets. In *ECML PKDD*, pages 410–425. Springer, 2016.

[10] E. Jaynes. Information theory and statistical mechanics. *Phys. Rev.*, 106(4):620–630, 1957.

[11] E. Jaynes. On the rationale of maximum-entropy methods. *Proc. IEEE*, 70(9):939–952, 1982.

[12] B. Kang, J. Lijffijt, R. Santos-Rodríguez, and T. D. Bie. SICA: subjectively interesting component analysis. *Data Min. Knowl. Disc.*, 32(4):949–987, 2018.

[13] B. Korte and J. Vygen. *Combinatorial Optimization*, volume 21 of *Algorithms and Combinatorics*. Springer Berlin Heidelberg, 2018.

[14] A. Krause and D. Golovin. Submodular Function Maximization. In *Tractability*, pages 71–104. Cambridge University Press, Cambridge, 2013.

[15] S. Kullback and R. A. Leibler. On information and sufficiency. *Annals Stat.*, 22(1):79–86, 1951.

[16] H. Liu, S. Jin, and C. Zhang. Connectionist temporal classification with maximum entropy regularization. In *NeurIPS*, pages 839–849, 2018.

[17] M. Mampaey, J. Vreeken, and N. Tatti. Summarizing data succinctly with the most informative itemsets. *ACM TKDD*, 6:1–44, 2012.

[18] M. Singh and N. K. Vishnoi. Entropy, optimization and counting. In *STOC*, pages 50–59. ACM, 2014.

[19] T. Sutter, D. Sutter, P. M. Esfahani, and J. Lygeros. Generalized maximum entropy estimation. *JMLR*, 20(138):1–29, 2019.

[20] N. Tatti. Computational complexity of queries based on itemsets. *Inf. Process. Lett.*, 98(5):183–187, 2006.

[21] G. Webb and J. Vreeken. Efficient discovery of the most interesting associations. *ACM TKDD*, 8(3):1–31, 2014.

[22] H. Wu, Y. Ning, P. Chakraborty, J. Vreeken, N. Tatti, and N. Ramakrishnan. Generating realistic synthetic population datasets. *ACM TKDD*, 12(4):45:1–45:22, 2018.

[23] R. Zhao, X. Sun, and V. Tresp. Maximum entropy-regularized multi-goal reinforcement learning. In K. Chaudhuri and R. Salakhutdinov, editors, *ICML*, volume 97, pages 7553–7562. PMLR, 2019.

APPENDIX

A. Reproducibility

In our experiments, we compare against OPUS [21], MTV [17], IIM [9] and DESC [5], for which we use the original implementation of the respective authors. OPUS discovers the top-k self-sufficient itemsets for a user defined k, that we set to the number of patterns that REAPER has discovered. In general, we use the default values for hyper-parameters suggested by the authors. For example, for IIM we limit the number of EM iterations to $1\ 000$ and the number of structure steps to $100\ 000$, as the authors have used as the default values of their implementation [9]. We limited all methods to 4 hours per dataset and in case they have not finished earlier, we report the results that have been achieved within that time frame. To compare the results of the pattern miner, we first discover a set of patterns using these methods. Next, from that, we create a relaxed maximum entropy distribution using our iterative approach that we have described in Sec. IV-0b. Note that the statically factorized distribution cannot incorporate all these patterns into the model, due to its limited budget. To compare the informativeness of the results of the pattern miner, we compute the BIC score using that model.

B. Datasets

All datasets that we have used in our experiments are publicly available. We have removed stop words, lemmatized and binarized the *AG News* text corpus and for the *AG Headlines* we have only considered news titles.[2] Similarly, we have lemmatized and binarized the two versions of the *CORD 19* dataset by extracting the abstracts from the CORD 19 open research dataset.[3] The *DQ* dataset of lemmatized Deep-Learning and Quantum-Theory ArXiv abstracts can be found in the supplementary material.[4] To reduce the number of attributes of the *Instacart* dataset, we have combined products from the same category, e.g. we merged Thin Spaghetti with Regular Spaghetti into the Spaghetti meta category.[5] We have used the *Chainstore, POWER C, PAMAP* datasets from the SPMF dataset collection,[6] and we have taken *Chess, Connect, Mushroom, Pumsb, Kosarak, Retail, Accidents* from the Itemset Mining Dataset Repository.[7] All the remaining datasets are from the UCI Machine Learning Repository[8] or from the LIBSVM repository.[9] We binarized each real valued attribute by binning it into 10 bins of equal width, and we mapped each categorical and ordinal attribute to multiple binary attributes, which is often referred to as 'one-to-k' encoding.

In Table I, we provide basic statistics about the datasets and the minimal support we have used in our experiments.

[2] di.unipi.it/~gulli/AG_corpus_of_news_articles
[3] kaggle.com/allen-institute-for-ai/CORD-19-research-challenge
[4] eda.mmci.uni-saarland.de/reaper
[5] instacart.com/datasets/grocery-shopping-2017
[6] philippe-fournier-viger.com/spmf
[7] fimi.ua.ac.be/data
[8] archive.ics.uci.edu/ml
[9] csie.ntu.edu.tw/~cjlin/libsvmtools/datasets

Dataset	$\lvert\mathcal{X}\rvert$	$\dim \mathcal{X}$	$\mathrm{E}_{x\in\mathcal{X}}[\lvert x\rvert]$	density	classes
Higgs	11000000	247	28.00 ± 0.00	0.1133	2
SUSY	5000000	178	18.00 ± 0.00	0.1011	2
Instacart	2620570	1235	3.14 ± 2.18	0.0025	1
Chainstore	1112949	46086	7.23 ± 8.91	0.0002	1
POWER C	1040000	125	7.00 ± 0.00	0.0560	1
KDD Cup 99	1000000	135	16.00 ± 0.00	0.1185	1
PAMAP	1000000	82	23.93 ± 0.73	0.2919	1
Kosarak	990002	41270	8.10 ± 23.62	0.0002	1
Covtype	581012	64	11.95 ± 0.23	0.1866	2
Record Link	574913	27	10.00 ± 0.00	0.3704	1
Accidents	340183	468	33.81 ± 2.94	0.0722	1
COD RNA	271617	16	8.00 ± 0.00	0.5000	2
Skin	245057	12	4.00 ± 0.06	0.3330	1
AG Headlines	127600	5243	3.09 ± 1.49	0.0006	4
AG News	127600	11489	13.63 ± 4.05	0.0012	4
Retail	88162	16470	10.31 ± 8.16	0.0006	1
Connect	67557	129	42.00 ± 0.00	0.3256	3
BMS WV 1	59602	497	2.51 ± 4.85	0.0051	1
BMS WV 2	77512	3340	4.62 ± 6.07	0.0014	1
Pumsb	49046	2113	74.00 ± 0.00	0.0350	1
Adult	48842	97	13.87 ± 0.48	0.1430	2
Plants	34781	69	8.69 ± 13.11	0.1259	1
CORD 19	32915	3517	62.67 ± 31.77	0.0179	1
Chess	28056	51	6.00 ± 0.00	0.1176	18
Letter Recognition	20000	102	16.00 ± 0.00	0.1569	26
US Census	13369	392	68.00 ± 0.37	0.1735	1
Nursery	12960	30	8.00 ± 0.00	0.2667	5
Pen Digits	10992	76	16.00 ± 0.00	0.2105	10
DQ	9993	434	22.30 ± 10.40	0.0514	1
Mushroom	8124	117	22.00 ± 0.00	0.1880	2
Breast Cancer	7325	397	11.67 ± 13.06	0.0294	2
Page Blocks	5473	39	10.00 ± 0.00	0.2564	5
DNA	5186	180	45.53 ± 5.22	0.2530	3
Waveform	5000	98	21.00 ± 0.00	0.2143	3
DNA Amplification	4587	391	5.78 ± 8.40	0.0148	1
Hypothyroid	3247	86	43.19 ± 0.39	0.5022	1
Led 7	3200	19	7.00 ± 0.00	0.3684	10
kr-vs-kp	3196	73	36.48 ± 0.50	0.4998	1
Splice	3190	287	60.73 ± 0.44	0.2116	1
Mammals	2183	121	24.81 ± 8.25	0.2050	1
German Credit	1000	110	38.70 ± 0.46	0.3518	1
Tic Tac Toe	958	27	9.74 ± 0.44	0.3606	1
Anneal	898	71	13.31 ± 1.45	0.1874	5
ICDM	859	3933	47.67 ± 14.32	0.0121	1
Diabetis	768	38	8.00 ± 0.00	0.2105	2
Australian Credit	653	124	51.53 ± 0.50	0.4155	1
Soybean	630	50	16.93 ± 0.25	0.3387	1
Vote	435	48	16.33 ± 0.47	0.3403	1
Ionosphere	351	155	34.00 ± 0.00	0.2194	2
Primary Tumor	336	31	15.79 ± 0.41	0.5092	1
Heart	303	50	12.98 ± 0.14	0.2596	5
Heart (Cleveland)	296	95	45.52 ± 0.50	0.4792	1
Audiology	216	146	67.13 ± 0.34	0.4598	1
Wine	178	65	13.00 ± 0.00	0.2000	3
Hepatitis	155	52	18.92 ± 1.83	0.3639	1
Iris	150	19	4.00 ± 0.00	0.2105	3
Lymph	148	68	27.72 ± 0.45	0.4077	1
Zoo	101	36	16.06 ± 0.24	0.4461	1

Table I: **Datasets** We show the number of rows, dimensions, number of classes, the average row size, and the overall density of the dataset that we used in our experiments.