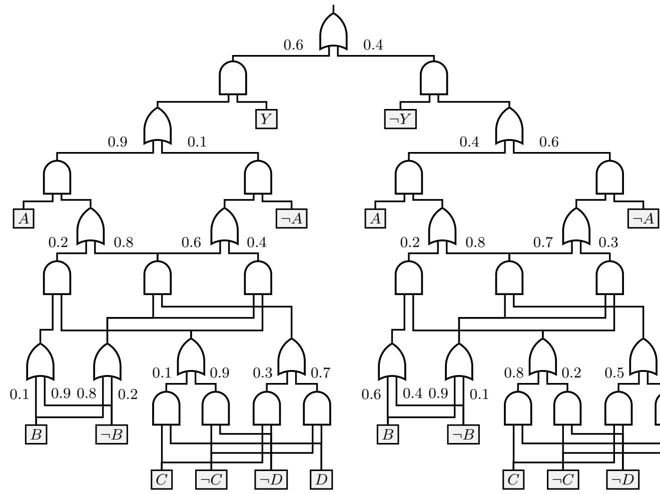
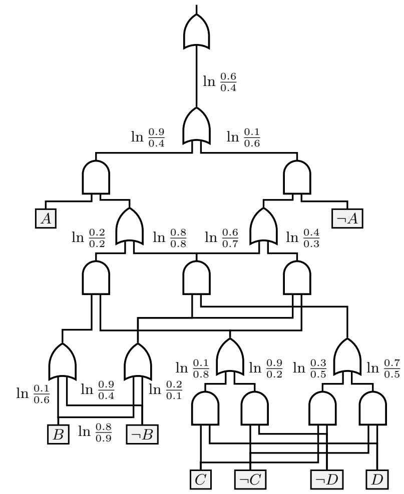


(a) Probabilistic circuit for joint distribution $\Pr(Y, A, B, C, D)$



(b) Logistic circuit for $\Pr(Y = 1 \mid A, B, C, D)$

Figure 5: A probabilistic circuit with parallel structures under class variable Y and its equivalent logistic circuit for predicting Y

feature that contributes most to the given sample's classification probability. That is, the feature that maximizes $\mathbb{x} \cdot \theta$. We visualize one such feature for MNIST data and one for Fashion in Figure 4 by marking the variables used in the their corresponding logical sentences.

6 Connection to Probabilistic Circuits

In recent years, a large number of tractable probabilistic models have been proposed as a target representation for generative learning of a joint probability distribution: arithmetic circuits (**?**), weighted SDD (**?**), PSDD (**?**), cutset networks (**?**) and sum-product networks (SPNs) (**?**). These representations have various syntactic properties. Some put probabilities on terminals, others on edges. Some use logical notation (AND, OR), others use arithmetic notation ($\times$,$+$). Nevertheless, they are all circuit languages built around the properties of decomposability and/or determinism.

For our purpose, we consider a simple probabilistic circuit language based on the logistic circuit syntax, where now the θ parameters are assumed to be normalized probabilities.[6]

Definition 6 (Probabilistic Circuit Semantics)**.** *A probabilistic circuit node n defines the following joint distribution.*

– *If n is a leaf (input) node, then $\Pr_n(\mathbf{x}) = [\mathbf{x} \models n]$.*

– *If n is an AND gate with children $c_1, \ldots, c_m$, then*

$$\Pr{}_n(\mathbf{x}) = \prod_{i=1}^{m} \Pr{}_{c_i}(\mathbf{x}).$$

– *If n is an OR gate with (child node, wire parameter) inputs $(c_1, \theta_1), \ldots, (c_m, \theta_m)$, then*

$$\Pr{}_n(\mathbf{x}) = \sum_{i=1}^{m} \Pr{}_{c_i}(\mathbf{x}) \cdot \theta_i.$$

[6]We also assume *smoothness* (**?**).

Figure 5a shows a probabilistic circuit for the joint distribution $\Pr(Y, A, B, C, D)$. This tractable circuit language is a relaxation of PSDDs (**?**) and a specific type of SPN (**?**) where determinism holds throughout. It is also a type of arithmetic circuit.

We are now ready to connect logistic and probabilistic circuits. It is well known that logistic regression is the discriminative counterpart of a naive Bayes generative model (**?**). A similar correspondence holds between our logistic and probabilistic circuits.

Proposition 6. *Consider a probabilistic circuit whose structure is of the form $(Y \wedge \alpha) \vee (\neg Y \wedge \beta)$, where sub-circuits α and β are structurally identical. Then, there exists an equivalent logistic circuit for the conditional probability of Y in the probabilistic circuit. Moreover, this logistic circuit has structure $\vee\alpha$ and its parameters can be computed in closed form as log-ratios of probabilistic circuit probabilities.*

We first depict this correspondence intuitively in Figure 5. The logistic circuit has the same structure as the two halves of the probabilistic circuit, and its parameters are computed from the probabilistic circuit probabilities. The distributions $\Pr(Y = 1 \mid A, B, C, D)$ represented by the circuits in Figures 5a and 5b are identical.

Formal Correspondence Next, we present the formal proof of this correspondence for binary $\mathbf{x}$. Recall that in our circuits, only the input wires of OR gates are parameterized. Let $\mathcal{W}_\delta$ be the set that contains all these wires in circuit δ:

$$\mathcal{W}_\delta = \{(n, c) \mid c \text{ is a gate with parent OR gate } n\}.$$

After expanding the equations in Definition 6 and following the top-down definition of global circuit flow (i.e., following Definition 4), one finds that the joint distribution induced by a probabilistic circuit δ can be rewritten as

$$\Pr{}_\delta(\mathbf{x}) = \prod_{(n,c) \in \mathcal{W}_\delta} f_\delta(n, \mathbf{x}, c) \cdot \theta^{\delta}_{(n,c)}.$$

We will exploit this finding in the derivation of the conditional distribution induced by the probabilistic circuit $\gamma = (Y \wedge \alpha) \vee (\neg Y \wedge \beta)$.

$$\begin{aligned}
&\Pr{}_{\gamma}(Y = 1 \mid \mathbf{x}) \\
&= \frac{\Pr_{\gamma}(Y{=}1) \Pr_{\alpha}(\mathbf{x})}{\Pr_{\gamma}(Y{=}0) \Pr_{\beta}(\mathbf{x}) + \Pr(Y{=}1) \Pr_{\alpha}(\mathbf{x})} \\
&= \frac{1}{1 + \frac{\Pr_{\gamma}(Y=0) \Pr_{\beta}(\mathbf{x})}{\Pr_{\gamma}(Y=1) \Pr_{\alpha}(\mathbf{x})}} \\
&= \frac{1}{1 + \frac{\Pr_{\gamma}(Y=0) \prod_{(n,c) \in \mathcal{W}_{\beta}} f_{\beta}(n,\mathbf{x},c) \theta^{\beta}_{(n,c)}}{\Pr_{\gamma}(Y=1) \prod_{(n,c) \in \mathcal{W}_{\alpha}} f_{\alpha}(n,\mathbf{x},c) \theta^{\alpha}_{(n,c)}}}
\end{aligned}$$

As stated in Proposition 6 and shown in Figure 5, subcircuits α and β share the same structure. Therefore, we can further simplify this equation as follows.

$$\begin{aligned}
&\Pr{}_{\gamma}(Y = 1 \mid \mathbf{x}) \\
&= \frac{1}{1 + \frac{\Pr_{\gamma}(Y=0)}{\Pr_{\gamma}(Y=1)} \prod_{(n,c) \in \mathcal{W}_{\alpha}} f_{\vee\alpha}(n, \mathbf{x}, c)^{\frac{\theta^{\beta}_{(n,c)}}{\theta^{\alpha}_{(n,c)}}}} \\
&= \frac{1}{1 + \exp\left[-g(\mathbf{x})\right]} = \Pr{}_{\vee\alpha}(Y = 1 \mid \mathbf{x})
\end{aligned}$$

where

$$g(\mathbf{x}) = \log \frac{\Pr_{\gamma}(Y{=}1)}{\Pr_{\gamma}(Y{=}0)} + \sum_{(n,c) \in \mathcal{W}_{\alpha}} f_{\vee\alpha}(n, \mathbf{x}, c) \log \frac{\theta^{\alpha}_{(n,c)}}{\theta^{\beta}_{(n,c)}} \tag{2}$$

$$= \theta^{\vee\alpha}_{root} + \sum_{(n,c) \in \mathcal{W}_{\alpha}} f_{\vee\alpha}(n, \mathbf{x}, c) \cdot \theta^{\vee\alpha}_{(n,c)}. \tag{3}$$

The transformation from Equation 2 to 3 expresses the logistic circuit parameters as the log-ratios of probabilistic circuit probabilities. For example, the class priors captured in the output wires of α and β are now combined as a log-ratio to form the bias term for $\vee\alpha$, expressed by the root parameter. This proof also provides us with a new perspective to understand the semantics of the learned parameters. The parameters represent the log-odds ratio of the features given different classes. Note that by Bayes' theorem, a naive Bayes model would derive its induced distribution in a sequence of steps similar to the ones above, resulting in Equation 2. Given this correspondence, one can also view our proposed structure learning method as a way to construct meaningful features for a naive Bayes classifier. We know that after training, naive Bayes classifiers are equivalent to logistic regression classifiers (as in Equation 3).

7 Related Work

? (**?**) proposed the first parameter learning algorithm for discriminative SPNs, using MPE inference as a sub-routine. Without the support of the determinism property, parameter learning of general SPNs is a relatively harder question than its logistic circuit counterpart, since it is non-convex. **?** (**?**) boost the accuracy of SPNs on MNIST to 97.6% by extracting more representative features from raw inputs based on the Hilbert-Schmidt independence measure. **?** (**?**) further improved the classification ability of SPNs by drastically simplifying SPN structure requirements and utilizing a loss objective that hybrids cross-entropy (discriminative learning) with log-likelihood (generative learning). **?** (**?**) developed a discriminative structure learning algorithm for arithmetic circuits. The method updates the circuit that represents a corresponding conditional random field (CRF) model by adding features conditioned on arbitrary evidence to the model. This work further relaxes decomposability and smoothness properties of ACs for a more compact representation. However, it targets the setting where there are a large number of output variables, not single-variable classification.

All the aforementioned literature conforms to a common trend of abandoning properties of the chosen circuit representations for easier structure learning and better prediction accuracy. They argue that those special syntactic restrictions complicate the learning process. On the contrary, this paper chooses perhaps the most structure-restrictive circuit as the target representation. Instead of relaxing the target representation's syntactical requirements, our proposed method fully leverages the valuable properties that stem from these restrictions, and in particular convexity.

8 Conclusions

We have presented logistic circuits, a novel circuit-based classification model with convex parameter learning and a simple structure learning procedure based on local search. Logistic circuits outperform much larger classifiers and perform well in a limited data regime. Compared to other symbolic, circuit-based approaches, logistic circuits present a leap in performance on image classification benchmarks. Future work includes support for convolution, parameter tying, and structure sharing in the logistic circuits framework.

Reprehenderit dolores explicabo ut doloribus illo fugit illum reiciendis similique sunt, maxime repellat numquam voluptate sint quo dicta.Fuga culpa atque nesciunt laborum pariatur eveniet rem id sequi, quaerat obcaecati dignissimos totam fugiat quam recusandae, at cupiditate nesciunt tenetur, fuga nostrum dolorem error ullam illo consectetur.Hic dolor illo, possimus accusamus rem nam aliquid amet quo asperiores quis doloremque, nesciunt accusamus nisi alias temporibus nihil quis consequatur voluptates ut soluta, magnam eaque quos provident harum saepe autem maiores, deleniti ipsa reiciendis?Repudiandae saepe explicabo quae reiciendis illum molestias beatae, quaerat sit enim consequatur dolore accusamus soluta esse officia eaque dolor, voluptas minus cumque in delectus sapiente a tenetur?Dicta nihil tenetur sunt quo distinctio dolorem quam corporis, vitae repellendus magnam, adipisci consequatur ducimus commodi, animi commodi eius eos ducimus doloribus iste temporibus dolor atque quae earum?Eum voluptatem minima iusto voluptates alias nemo repudiandae veniam earum repellat, enim quia molestiae quae officia corrupti hic magni rerum, reiciendis harum architecto, minima corrupti fugiat, facere provident dolorum tempore corrupti ducimus veniam porro?Ipsum reprehenderit quidem fuga, dignissimos unde laborum?Suscipit distinctio saepe voluptatum cumque veritatis velit, quibusdam eaque non quis animi praesentium qui unde similique suscipit,