1. Motivation

**Probabilistic Sentential Decision Diagrams (PSDDs):**
- Structured Decomposable probabilistic circuits
- Encode certain knowledge as logic constraints
- Encode uncertain knowledge as probabilities
- Interpretable syntax
  - Many inferences are exact and tractable:
    - Evidence
    - Marginals
    - MLE Parameter Learning
    - KL-divergence

**PSDD circuit represents** recursive decomposition of formula: $\bigvee_{i=1}^{k} (p_i \land s_i)$, where each prime $p_i$ and sub $s_i$ are logical formulae

**Existing PSDD learners:**
- require an initial PSDD encoding the support;
- scale poorly to complex formulae and/or high dimension.

**This Work:** How to effectively learn PSDDs s.t. complex formula?

2. SamplePSDD

- Common assumption: primes $p_i$ are conjunctions of literals.
  $$\phi(A, B, C, D) = (A \land \neg B \land \neg D) \lor (B \land \neg C \land D)$$

- **Problem:** Size of circuit is exponential in the size of $p_i$

- **Solution:** randomly sample a bounded number ($k$) of $p_i$

- **But:** this violates structure decomposability

**Example, $k = 3$:**

- **New solution:** Relax logical constraints $\phi$

3. Experiments

**Evaluation:** we sample 30 PSDDs and use 5 ensemble strategies:
- Likelihood weighting (LLW)
- Uniform weights
- Expectation Maximization (EM)
- Stacking
- Bayesian Model Combination

Comparing with **Strudel**, **LearnPSDD** and **LearnSPN**.

**Datasets:** we evaluate with 5 data + knowledge as logic constraints:

<table>
<thead>
<tr>
<th>Dataset</th>
<th># vars</th>
<th># train</th>
<th>$\phi$’s size</th>
</tr>
</thead>
<tbody>
<tr>
<td>LED</td>
<td>14</td>
<td>5000</td>
<td>23</td>
</tr>
<tr>
<td>LED + IMAGES</td>
<td>157</td>
<td>7000</td>
<td>39899</td>
</tr>
<tr>
<td>SUSHI RANKING</td>
<td>100</td>
<td>3500</td>
<td>17413</td>
</tr>
<tr>
<td>SUSHI Top 10</td>
<td>10</td>
<td>3500</td>
<td>37</td>
</tr>
<tr>
<td>Dota 2 Games</td>
<td>227</td>
<td>92650</td>
<td>1308</td>
</tr>
</tbody>
</table>

Our approach is better with fewer data, yet remains competitive under lots of data.

Samples perform better with higher $k$’s and light-learning vtrce...

...but at a cost to complexity.