Learning Tractable Probabilistic Models

Pedro Domingos
University of Washington

Daniel Lowd
Dept. Computer & Information Science
University of Oregon
Outline

- Motivation
- Standard tractable models
- The sum-product theorem
- Bounded-inference graphical models
- Feature trees
- Sum-product networks
- Tractable Markov logic
- Other tractable models
Goal: Large Joint Models

- Natural language
- Vision
- Social networks
- Activity recognition
- Bioinformatics
- Etc.
Example: Friends & Smokers

Smoking and Quitting in Groups
Researchers studying a network of 12,067 people found that smokers and nonsmokers tended to cluster in groups of close friends and family members. As more people quit over the decades, remaining groups of smokers were increasingly pushed to the periphery of the social network.

1971 A sample of 1,000 people from the study includes many large groups of smokers.

2000 Nearly three decades later, groups of smokers tended to be smaller and more isolated.

KEY
- Male smoker
- Female smoker
- Male nonsmoker
- Female nonsmoker
- Friendship, marriage or family tie

Circle size is proportional to the number of cigarettes smoked per day.

Sources: New England Journal of Medicine, Dr. Nicholas A. Christakis; James H. Fowler
The Hardest Part of Learning Is Inference

Inference is subroutine of:
- Learning undirected graphical models
- Learning discriminative graphical models
- Learning w/ incomplete data, latent variables
- Bayesian learning
- Deep learning
- Statistical relational learning
- Etc.
Inference Is the Bottleneck

- Inference is \#P-complete
- It’s tough to have \#P as a subroutine
- Approximate inference and parameter optimization interact badly
- An intractable accurate model is in effect an inaccurate model
- What can we do about this?
One Solution: Learn Only Tractable Models

- **Pro**: Inference problem is solved
- **Con**: Insufficiently expressive

Recent development: Expressive tractable models

(theme of this tutorial)
Definitions of Tractability

“Tractable” implies that certain operations are efficient. There are many operations that we might want to be efficient:

- **Probabilistic inference** – marginal, conditional, etc.
- **MAP inference** – most likely complete configuration
- **Marginal MAP** – most likely partial configuration
- **Sampling** – generate independent samples from the posterior distribution (conditioned on evidence).
- **Maximum Likelihood Estimation**

Different types of models make different operations tractable.
Advantage: Compact representation
Inference: $P(Burglar \mid Alarm) = ??$
Need to sum out Earthquake
Inference cost exponential in treewidth of graph
Learning Graphical Models

- General idea:
  
  **Empirical statistics = Predicted statistics**

- Requires inference!

- Approximate inference is very unreliable

- No closed-form solution (except rare cases)

- Hidden variables $\rightarrow$ Local optima

- **Result:** Learning is very hard
Outline

- Motivation
- **Standard tractable models**
  - The sum-product theorem
  - Bounded-inference graphical models
  - Feature trees
  - Sum-product networks
  - Tractable Markov logic
  - Other tractable models
Thin Junction Trees

[Karger & Srebro, SODA-01; Bach & Jordan, NIPS-02; Narasimhan & Bilmes, UAI-04; Chechetka & Guestrin, NIPS-07; Elidan & Gould, JMLR-08]

- **Junction tree**: obtained by triangulating the Markov network
- **Inference**: exponential in treewidth (size of largest clique in junction tree)
- **Solution**: Learn only low-treewidth models
- **Algorithms**: Greedily optimize likelihood or search for conditional independencies given small sets of variables.
- **Problem**: Too restricted
Very Large Mixture Models
[Lowd & Domingos, ICML-05]

- Just learn a naive Bayes mixture model with lots of components (hundreds or more)
- Inference is linear in model size (no worse than scanning training set)
- Compared to Bayes net structure learning:
  - Comparable data likelihood; better query likelihood; much faster & more reliable inference
- Problem: Curse of dimensionality
Outline

- Motivation
- Standard tractable models
- **The sum-product theorem**
- Bounded-inference graphical models
- Feature trees
- Sum-product networks
- Tractable Markov logic
- Other tractable models
Efficiently Summable Functions

A function is **efficiently summable** iff its sum over any subset of its scope can be computed in time polynomial in the cardinality of the subset.
The Sum-Product Theorem

If a function is:

- A sum of efficiently summable functions with the same scope, or

- A product of efficiently summable functions with disjoint scopes,

Then it is also efficiently summable.

\[
\sum_A (f(A) + g(A)) = \sum_A f(A) + \sum_A g(A)
\]

\[
\sum_{A,B} f(A)g(B) = \left(\sum_A f(A)\right)\left(\sum_B g(B)\right)
\]
Corollary

Every low-treewidth distribution is efficiently summable, but not every efficiently summable distribution has low treewidth.
Compactly Representable Probability Distributions

Graphical Models

Sum-Product Models

Standard Tractable Models
Compactly Representable Probability Distributions

Graphical Models

Sum-Product Models

Polynomial-time exact inference
Outline

- Motivation
- Standard tractable models
- The sum-product theorem
- **Bounded-inference graphical models**
- Feature trees
- Sum-product networks
- Tractable Markov logic
- Other tractable models
Arithmetic Circuits

[Darwiche, JACM, 2003]

- Inference consists of sums and products
- Can be represented as an arithmetic circuit
- Complexity of inference = Size of circuit
Arithmetic Circuit

- Rooted DAG of sums and products
- Leaves are indicator variables
- Computes marginals in linear time
- Graphical models can be compiled into ACs
Learning Bounded-Inference Graphical Models [L. & D., UAI-08]

- Use standard Bayes net structure learner (with context-specific independence)

**Key idea:** Instead of using representation complexity as regularizer:

\[
\text{score}(M, T) = \log P(T|M) - k_p n_p(M)
\]

(\text{log-likelihood – #parameters})

Use *inference complexity*:

\[
\text{score}(M, T) = \log P(T|M) - k_c n_c(M)
\]

(\text{log-likelihood – circuit size})
Learning Bounded-Inference Graphical Models (contd.)

- Incrementally compile circuit as structure added (splits in decision trees):

- Compared to Bayes nets w/ Gibbs sampling:
  Comparable data likelihood; better query likelihood; much faster & more reliable inference

- Large treewidth (10’s – 100’s)
Learning Bounded-Inference Undirected Models (ACMN)

[L. & Rooshenas, AISTATS-13]

- Greedy Markov network feature induction:

  1. Generate candidate features.
  2. Score each candidate.
  3. Add the best one and update weights.

- Adapt complexity regularizer and incremental compilation to learn MN with compact circuit.

- Can directly optimize likelihood rather than approximations (BP, MCMC) or surrogates (PL).

- More flexible than BNs → Better accuracy
Outline

- Motivation
- Standard tractable models
- The sum-product theorem
- Bounded-inference graphical models
- **Feature trees**
- Sum-product networks
- Tractable Markov logic
- Other tractable models
Thin junction tree learners work by repeatedly finding a subset of variables $A$ such that

$$P(B, C|A) \approx P(B|A) \ P(C|A)$$

where $A, B, C$ is a partition of the variables

LEM algorithm: Instead find a feature $F$ s.t.

$$P(B, C|F) \approx P(B|F) \ P(C|F)$$

and recurse on variables and instances

Result is a tree of features
A Feature Tree

\[ x_1 \land \neg x_2 \]

\[ 0 \quad 1 \]

\[ x_3 \land x_4 \quad \neg x_5 \land x_6 \quad x_3 \land \neg x_5 \quad x_4 \land x_6 \]

\[ 0 \quad 1 \quad 0 \quad 1 \quad 0 \quad 1 \quad 0 \quad 1 \]
Feature Trees (contd.)

- High treewidth because of context-specific independence
- More flexible than decision tree CPDs
- PAC-learning guarantees
- Outperforms thin junction trees and other algorithms for learning Markov networks
- More generally: Feature graphs
Outline

- Motivation
- Standard tractable models
- The sum-product theorem
- Bounded-inference graphical models
- Feature trees
- **Sum-product networks**
- Tractable Markov logic
- Other tractable models
A Univariate Distribution
Is an SPN

Multinomial  Gaussian  Poisson  ...
A Product of SPNs over Disjoint Variables Is an SPN
A Weighted Sum of SPNs over the Same Variables Is an SPN

Sums out a mixture variable
Recurse Freely . . .
All Marginals Are Computable in Linear Time

\[ P(X=0) = 0.26 \]
All MAP States Are Computable in Linear Time

$$\max_y P(X=0, Y=y) = 0.12$$
All MAP States Are Computable in Linear Time

$$\max_y P(X=0, Y=y) = 0.12$$
What Does an SPN Mean?

Products = Features
Sums = Clusters
Special Cases of SPNs

- Hierarchical mixture models
- Thin junction trees (e.g.: hidden Markov models)
- Non-recursive probabilistic context-free grammars
- Etc.
Discriminative SPNs

[Gen's & D., NIPS-12; Best Student Paper Award]
Discriminative Training

\[ \nabla \log P(y|x) = \nabla \log \frac{P(y, x)}{P(x)} = \]

\[ \nabla \log \sum_h P(y, h, x) - \nabla \log \sum_{y', h} P(y', h, x) \]

Tractable!

Correct label

Best guess
Backpropagation

For each sum child $j$:
\[
\frac{\partial S}{\partial S_j} \leftarrow \frac{\partial S}{\partial S_j} + w_{n,j} \frac{\partial S}{\partial S_n}
\]
\[
\frac{\partial S}{\partial w_{n,j}} \leftarrow S_j \frac{\partial S}{\partial S_n}
\]
Backpropagation

For each product child $j$:

$$\frac{\partial S}{\partial S_j} \leftarrow \frac{\partial S}{\partial S_j} + \frac{\partial S}{\partial S_n} \prod_{k \in Ch(n) \setminus \{j\}} S_k$$
Problem: Gradient Diffusion
Solution: Hard Inference

Soft Inference
(Marginals)

Hard Inference
(MAP States)
Hard Gradient

\[ \nabla \log \tilde{P}(y|x) = \nabla \log \frac{\tilde{P}(y, x)}{\tilde{P}(x)} = \]

\[ \nabla \log \max \max \max \max P(y, h, x) \]

\[ \text{max } P(y, h, x) \]

\[ \text{max } P(y', h, x) \]
Hard Gradient

\[ \nabla \log \tilde{P}(y|x) = \nabla \log \frac{\tilde{P}(y, x)}{\tilde{P}(x)} = \]

Number with correct label \(-\) Number with model guess

\[ \frac{\partial}{\partial w_i} \log \tilde{P}(y|x) = \frac{\Delta c_i}{w_i} \]
Empirical Evaluation: Object Recognition

CIFAR-10
32x32 pixels
50k training exs.
10k test exs.

STL-10
96x96 pixels
5k training exs.
8k test exs.
100k unlabeled exs.
CIFAR-10 Results

Accuracy vs. Dictionary Size (K)

- SPN
- Pooling
- SVM
- Autoencoder
- RBM

4x4xK
STL-10 Results

- 1-layer Vector Quantization: 54.9%
- 1-layer Sparse Coding: 59.0%
- 3-layer Learned Receptive Field: 60.1%
- Discriminative SPN: 62.3%

without unlabeled data
Generative Weight Learning
[Poon & D., UAI-11; Best Paper Award]

- Model joint distribution of all variables
- Algorithm: Online hard EM
- Sum node maintains counts for each child
- For each example
  - Find MAP instantiation with current weights
  - Increment count for each chosen child
  - Renormalize to set new weights
- Repeat until convergence
Empirical Evaluation: Image Completion

- Datasets: Caltech-101 and Olivetti
- Compared with DBNs, DBMs, PCA and NN
- SPNs reduce MSE by ~1/3
- Orders of magnitude faster than DBNs, DBMs
Architecture

Whole Image

Region

Pixel

$\text{Pixel} \quad \bullet \quad x \quad y \quad \cdots$
LearnSPN: Top-down learning of SPN structure.
Empirical Evaluation

- 20 varied real-world datasets
  - 10s-1000s of variables
  - 1000s-100,000s of samples
- Compared with state-of-the-art Bayesian network and Markov random field learners
- Likelihood: typically comparable
- Query accuracy: much higher
- Inference: orders of magnitude faster
ID-SPN: Learn an SPN with Indirect and Direct Variable Interactions

[Rooshenas & L., ICML-14]

- LearnSPN and ACMN are both special cases.
- ID-SPN is more accurate than LearnSPN and ACMN (20 and 17 datasets, respectively)

Source code: http://libra.cs.uoregon.edu/
Outline

- Motivation
- Standard tractable models
- The sum-product theorem
- Bounded-inference graphical models
- Feature trees
- Sum-product networks
- **Tractable Markov logic**
- Other tractable models
Tractable Markov Logic
[D. & Webb, AAAI-12]

- Tractable representation for statistical relational learning
- Three types of weighted rules and facts
  - **Subclass:** Is(Family, SocialUnit)
    Is(Smiths, Family)
  - **Subpart:** Has(Family, Adult, 2)
    Has(Smiths, Anna, Adult1)
  - **Relation:** Parent(Family, Adult, Child)
    Married(Anna, Bob)
Restrictions

- One top class
- One top object (all others are subparts)
- Relations must be among subparts of some object
- Subclasses are mutually exclusive
- Objects do not share subparts
TML Semantics

\[ Z(X, C) = \left( \sum_{S} e^{w_{s}} Z(X, S) \right) \times \left( \prod_{P} Z(P(X), C_{P})^{n_{P}} \right) \times \left( \prod_{R} (1 + e^{w_{R}}) \right) \]

\[ Z(KB) = Z(\text{TopObject}, \text{TopClass}) \]
Tractability

**Theorem:** The partition function of every TML knowledge base can be computed in time and space polynomial in the size of the knowledge base.

\[ \text{Time} = \text{Space} = O(\#\text{Rules} \times \#\text{Objects}) \]
Why TML Is Tractable

KB structure is isomorphic to Z computation:
- Parts = Products
- Classes = Sums
Why TML Is Tractable

KB structure is isomorphic to Z computation:
• Parts = Products
• Classes = Sums
Why TML Is Tractable

KB structure is isomorphic to Z computation:
- Parts = Products
- Classes = Sums
Expressiveness

The following can be compactly represented in TML:

- Junction trees
- Sum-product networks
- Probabilistic context-free grammars
- Probabilistic inheritance hierarchies
- Etc.
Learning Tractable MLNs

Alternate between:

- Dividing / aggregating the domain into subparts
- Inducing class hierarchies over similar subparts
Other Sum-Product Models

- Relational sum-product networks
- Tractable probabilistic knowledge bases
- Tractable probabilistic programs
- Etc.
What If This Is Not Enough?

Use variational inference, with the most expressive tractable representation available as the approximating family

[L. & D., NIPS-10]
Outline

- Motivation
- Standard tractable models
- The sum-product theorem
- Bounded-inference graphical models
- Feature trees
- Sum-product networks
- Tractable Markov logic
- Other tractable models
Other Tractable Models

- Symmetry
  - Liftable models
  - Exchangeable models
- Submodularity
- Determinantal point processes
- Etc.
Liftable Models

- **Example:** Consider a distribution over the friendships and smoking habits of $n$ people.

  Each person is a smoker or a non-smoker
  Friendships more likely with matching smoking habits.

- **Key insight:** Without evidence, we have identical information about each individual, so they must have identical marginals.

- **Lifted inference:** Polynomial in $n$
Liftable Models (cont.)

[Jaimovich et al., UAI-07; Van Haaren et al., LTPM-14]

- Such symmetries commonly occur in statistical relational models (e.g., Markov logic networks)
- **Domain-lifted inference algorithms** run in time polynomial in the domain size (number of objects).
  [Van Den Broeck, NIPS-11]

  Predicted statistics are tractable
  \[ \Rightarrow \text{ Weight learning is tractable} \]
  \[ \Rightarrow \text{ Structure learning is tractable} \]

- Learn bounded-inference *first-order* graphical models
  *(See tutorial by Guy Van den Broeck and Dan Suciu this afternoon.*)
Exchangeable Variable Models
[Niepert & D., ICML-14]

- Variables are finitely exchangeable if probability distribution invariant under variable permutations.
- Parameterization as **mixture of independent urns:**
  - Urn $U_i$ represents assignments with exactly $t$ ones (black balls).
  - Partial finite exchangeability is similarly defined over any statistic $T$.
- **Inference:** For many statistics $T$, MAP and marginal inference is polynomial in number of values of $T$. 

\[\begin{align*}
X_1 & \quad X_2 & \quad X_3 \\
P(0) & \quad & \quad P(3) \\
U_0 & \quad U_1 & \quad U_2 & \quad U_3 \\
\end{align*}\]
Exchangeable Variable Models (Cont.)

- EVM mixture model
  - Conditioned on class C, attributes are partitioned into mutually independent exchangeable blocks
  - Leads to spectrum of probabilistic models:

- Learning: Structural EM (faster than NB mixture)
- Experiments
  - Competitive likelihood with other tractable models
  - Faster and easier to tune
Submodular Potentials

- Consider a pairwise binary Markov network:
  \[ P(x) = \frac{1}{Z} \exp \left( - \sum_i \epsilon_i(x_i) - \sum_{i,j} \epsilon_{i,j}(x_i, x_j) \right) \]
- A pairwise energy is submodular (attractive) if:
  \[ \epsilon(1, 1) + \epsilon(0, 0) \leq \epsilon(1, 0) + \epsilon(0, 1) \]
- Exact MAP inference in polynomial time with graph cuts
- Marginal inference remains intractable
- Applications: image segmentation, denoising, stereo reconstruction
Determinantal Point Processes

- Given a collection of items \( \mathcal{Y} = \{1, \ldots, N\} \)
  define distribution over subsets \( Y \subset \mathcal{Y} \)
- Define similarity matrix \( L \): \( L_{ij} = g(i)^T g(j) \)
- Probabilities:
  \[ P(Y) = \det(L_Y) / \det(L + I) \]
  Volume of submatrix
- Marginals:
  \[ P(A \subset Y) = \det(K_A) \]
  Normalization
  \[ K = L(L + I)^{-1} \]
Determinantal Point Processes (cont.)

- Intuition: determinant is the *volume* of the transformation, which is larger for diverse sets.

\[
\text{det}(\vec{v}_1, \vec{v}_2, \vec{v}_3) = \text{det}(
\begin{pmatrix}
\vec{v}_1 & \vec{v}_2 & \vec{v}_3
\end{pmatrix}) = 
\]

- Applications: search results, document summarization
Ongoing Work

Sample of topics from ICML 2014 workshop on Learning Tractable Probabilistic Models:

- Tractable conditioning and marginalization by learning directed model for any variable order. [Uria & al., ICML-2014]
- Chow-Liu trees with cut-set conditioning. [Rahman & al., ECML-2014]
- Sentential decision diagrams for learning with logical constraints [Kisa & al., KR-2014]
- Using sum-product theorem for non-convex optimization [Friesen & D., LTPM-2014]
- …and many more…
Open Questions

- Defining and exploiting new kinds of tractable structures
- Combining existing tractable structures
  (e.g., exchangeability and lifting [Van den Broeck & Niepert, AAAI-14], best paper nominee)
- Better methods to fit tractable structures to data
- Combining with approximate inference to get approximate inference with guaranteed time and error bounds (e.g., high-girth graphical models [Heinemann & Globerson, ICML-14])
- Much more!
Summary

Graphical Models

Expressiveness

Tractability
Summary

- New Tractable Classes
- Graphical Models

Tractability vs. Expressiveness