6.541/18.405 Final Project, Progress Report #1

Subject: The Computational Complexity of Probabilistic Inference

George Matheos, April 17, 2024

1 Literature review update

1.1 Papers read

1. Mark R Jerrum, Leslie G Valiant, and Vijay V Vazirani. "Random generation of combinatorial structures from a uniform distribution". In: *Theoretical computer science* 43 (1986), pp. 169–188
2. Gregory F Cooper. "The computational complexity of probabilistic inference using Bayesian belief networks". In: *Artificial intelligence* 42.2-3 (1990), pp. 393–405
3. Paul Dagum and Michael Luby. "Approximating probabilistic inference in Bayesian belief networks is NP-hard". In: *Artificial intelligence* 60.1 (1993), pp. 141–153
4. Paul Dagum and Michael Luby. "An optimal approximation algorithm for Bayesian inference". In: *Artificial Intelligence* 93.1-2 (1997), pp. 1–27
5. Nathanael L Ackerman, Cameron E Freer, and Daniel M Roy. "On the computability of conditional probability". In: *Journal of the ACM (JACM)* 66.3 (2019), pp. 1–40
6. Johan Kwisthout. "Approximate inference in Bayesian networks: Parameterized complexity results". In: *International Journal of Approximate Reasoning* 93 (2018), pp. 119–131

1.2 Papers remaining to read

1. Shyan Akmal and Ryan Williams. "majority-3sat (and related problems) in polynomial time". In: *2021 IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE. 2022, pp. 1033–1043
2. Till Tantau. "On the Satisfaction Probability of k-CNF Formulas". In: *arXiv preprint arXiv:2201.08895* (2022)
3. Ankur Moitra. "Approximate counting, the Lovász local lemma, and inference in graphical models". In: *Journal of the ACM (JACM)* 66.2 (2019), pp. 1–25
4. Scott Aaronson. "The equivalence of sampling and searching". In: *Theory of Computing Systems* 55.2 (2014), pp. 281–298
5. Tomoyuki Yamakami. "Polynomial time samplable distributions". In: *journal of complexity* 15.4 (1999), pp. 557–574
6. Uriel Feige. "Relations between average case complexity and approximation complexity". In: *Proceedings of the thiry-fourth annual ACM symposium on Theory of computing*. 2002, pp. 534–543

1.2.1 Other papers to potentially read

1. Chunxiao Li et al. "On the hierarchical community structure of practical Boolean formulas". In: *Theory and Applications of Satisfiability Testing–SAT 2021: 24th International Conference, Barcelona, Spain, July 5-9, 2021, Proceedings 24*. Springer. 2021, pp. 359–376
2. I also want to find a good introduction to one-way functions which are hard to invert for random inputs. I've realized this is very relevant. Any pointers?

2 Update on possible directions for new research

2.1 Analyzing the complexity of approximating $P(X|Y)$ for average-case Y

TLDR: I have realized this is closely related to the existence of injective one-way functions which are hard to invert for average-case input. One project direction would be to read up on such one-way functions, and spell out this connection (which I have not seen mentioned in the literature) in my project report.

2.1.1 The question to consider

Consider a Bayesian network (probabilistic graphical model) B on a set $\mathcal{V}$ of boolean variables. This defines a distribution P on $\{0,1\}^{\mathcal{V}}$. Let X, Y be disjoint subsets of $\mathcal{V}$ and let $x \in \{0,1\}^X$ and $y \in \{0,1\}^Y$. I am interested in the complexity of computing $P(X = x|Y = y)$ up to ϵ additive error.

It is known (Dagum and Luby 1993) that if the following condition holds, $\mathbf{NP} = \mathbf{BPP}$:

$$\exists \epsilon \in (0, 1/2), \exists \text{ ptime probabilistic Turing Machine } M \text{ such that} \forall B \text{ a description of a Bayesian network,}$$
$$\forall X, Y \text{ disjoint subsets of the variables in } B, \forall x \in \{0,1\}^X, \forall y \in \{0,1\}^Y,$$
$$\mathbb{P}_r[|M(B, X, Y, x, y, r) - P(X = x|Y = y)| \le \epsilon] \ge 2/3 \quad (1)$$

Here, r is uniformly sampled random bits for the probabilistic Turing machine M.

I have not, however, seen results casting doubt on the following proposition:

$$\exists \epsilon << 1/2, \exists \text{ ptime probabilistic Turing Machine } M \text{ such that} \forall B \text{ a description of a Bayesian network,}$$
$$\forall X, Y \text{ disjoint subsets of the variables in } B, \forall x \in \{0,1\}^X$$
$$\mathbb{P}_{r; y \sim P}[|M(B, X, Y, x, y, r) - P(X = x|Y = y)| \le \epsilon] >> 2/3 \quad (2)$$

The difference is that in the former proposition, y is an arbitrary observation value. In the second proposition, the probabilistic model B is worst-case, but y is sampled from the marginal of P on Y. (To the extent that probabilistic models describe the actual distributions with which things occur in the real world, we expect the inference problems presented by the real world to be on average-case y.)

2.1.2 Progress toward results

Observation 1: if $|Y| = 1$, (2) holds. I have realized that if we restrict to cases where Y contains one boolean variable ($|Y| = 1$), then (2) holds. The reason is that if Y contains one variable, either (A) $P(Y = y) > 0.001$, in which case approximate inference is easy to do via rejection sampling, or $P(Y = \neg y) \ge 0.999$, which means that with high probability, $Y \ne y$ and it is okay if it is hard to approximate $P(X|y)$.

Observation 2: if we allow large $|Y|$, (2) implies invertible one-way functions cannot be hard to invert on most random inputs. Say there exists a family of injective functions $(f_i)_{i=1}^{\infty}$ with $f_i : \{0,1\}^{m_i} \to \{0,1\}^{n_i}$ for some natural numbers m_i, n_i. Say these functions have the property that there is no probabilistic Turing machine M which can invert f_i on a random input x with probability $> 2/3$ for all i. That is, $\forall M$,

$$\exists i \text{ s.t. } \mathbb{P}_{x \sim \text{Uniform}(\{0,1\}^{n_i}), r}(M(f_i, y, r) = x) < 2/3 \quad \text{where } y = f_i(x)$$

Then for any i, we can encode a Bayesian network B which (1) first samples variables $X_1, \dots, X_{m_i}$ uniformly at random, and then (2) computes $Y = f_i(X_1, \dots, X_{m_i})$ by encoding a deterministic circuit for f_i. If with high probability we could approximate $\mathbb{P}(X_1 = 0)$, we could apply an iterative algorithm to construct an assignment to $X_1, \dots, X_{m_i}$, and then check if $Y = f_i(X_1, \dots, X_{m_i})$. If we could do this with high probability, we could invert f_i with high probability.

2.2 Complexity results parametrized by information theoretic quantities