Learning from context-free grammars with Boltzmann principles*

Sergey Dovgal[1]

[1]Université Paris-13,

Alea Young 2017, Paris

∗ in progress

Learning from unambiguous context-free grammars with Boltzmann principles*

Maciej Bendkowski[1] Olivier Bodini[2]
Sergey Dovgal[2,3,4,5]

[1]Jagiellonian University, [2]Université Paris-13 Villetaneuse, [3]Université Paris-7 Dénis Diderot, [4]Moscow Institute of Physics and Technology, [5]Institute for Information Transmission Problems

Alea Young 2017, Paris

∗ in progress

First polynomial-time tuning of multivariate Boltzmann sampler and new applications

Maciej Bendkowski[1] Olivier Bodini[2]
Sergey Dovgal[2,3,4,5]

[1]Jagiellonian University, [2]Université Paris-13 Villetaneuse, [3]Université Paris-7 Dénis Diderot, [4]Moscow Institute of Physics and Technology, [5]Institute for Information Transmission Problems

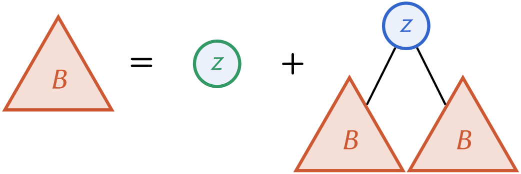
Alea Young 2017, Paris

$*$ in progress

Outline

Warm-up: binary trees

- Binary tree is either *a leaf* or *a root with two binary trees*



- Generating function for binary trees

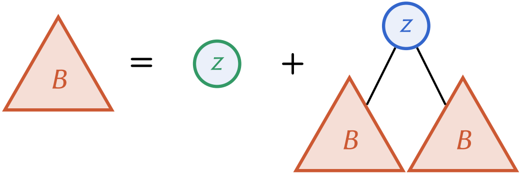
$$B(z) = \sum_{n \geq 0} b_n z^n \quad ,$$

$b_n = \#$ binary trees with n nodes.

- $$B(z) = z + zB^2(z) = \frac{1 - \sqrt{1 - 4z^2}}{2z}$$

Warm-up: binary trees

- Binary tree is either *a leaf* or *a root with two binary trees*



- Generating function for binary trees

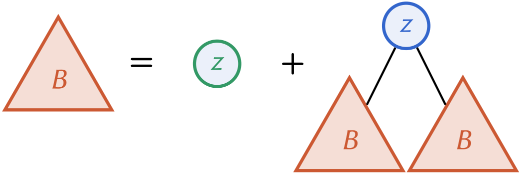
$$B(z) = \sum_{n \geq 0} b_n z^n \quad ,$$

$b_n = \#$ binary trees with n nodes.

- $$B(z) = z + zB^2(z) = \frac{1 - \sqrt{1 - 4z^2}}{2z}$$

Warm-up: binary trees

- Binary tree is either *a leaf* or *a root with two binary trees*



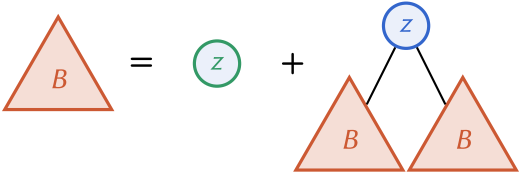
- Generating function for binary trees

$$B(z) = \sum_{n \geq 0} b_n z^n \quad ,$$

$b_n = \#$ binary trees with n nodes.

- $$B(z) = z + zB^2(z) = \frac{1 - \sqrt{1 - 4z^2}}{2z}$$

Univariate Boltzmann sampler



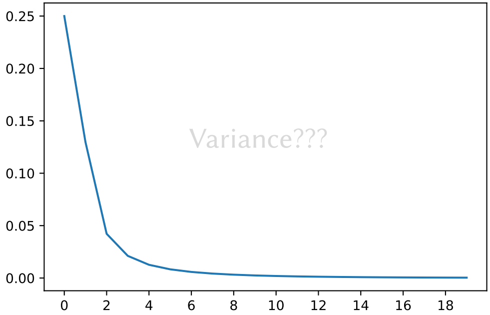
$$B(z) = z + zB^2(z)$$

- Boltzmann sampler $\Gamma B(z)$:

$$\Gamma B(z) := \begin{cases} \text{leaf} & \text{with probability } \dfrac{z}{z + zB^2(z)}, \\ (\text{root}, \Gamma B(z), \Gamma B(z)) & \text{with probability } \dfrac{zB^2(z)}{z + zB^2(z)}. \end{cases}$$

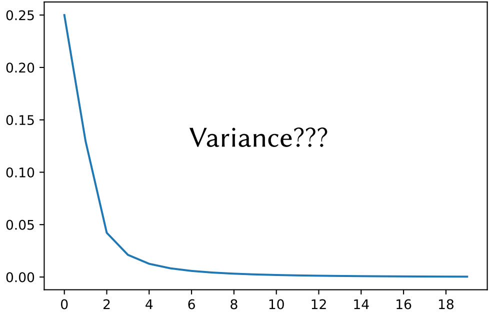
What is the distribution of size?

- Expected size of an object $= z\frac{B'(z)}{B(z)}$
- $\mathbb{P}(\text{tree of size } n) = \dfrac{b_n z^n}{B(z)}$, generation inside the size is uniform



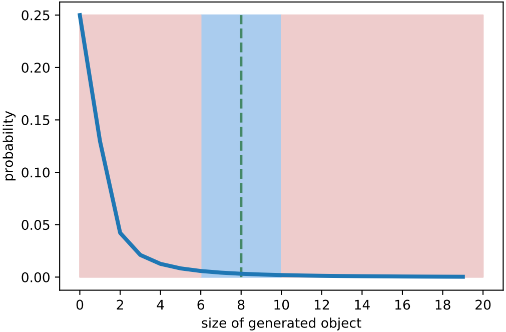
What is the distribution of size?

- Expected size of an object $= z\frac{B'(z)}{B(z)}$
- $\mathbb{P}(\text{tree of size } n) = \dfrac{b_n z^n}{B(z)}$, generation inside the size is uniform



Approximate-size sampling

How long does it take to reject objects not from $[n(1-\varepsilon), n(1+\varepsilon)]$?



Answer: binary trees require some modification.. see next slide

Pointing operator

Select and mark any node inside binary tree

$$\boxed{B(z)^{\bullet} := z\frac{d}{dz}B(z)}$$

- $B(z) = z + zB^2(z)$
- $B^{\bullet}(z) = z^{\bullet} + z^{\bullet}B^2(z) + zB^{\bullet}(z)B(z) + zB(z)B^{\bullet}(z)$
- Boltzmann sampler $\Gamma B^{\bullet}(z)$

$$\Gamma B^{\bullet}(z) := \begin{cases} z^{\bullet}, & \text{w.p. } \frac{z}{z+2z^{\bullet}B^2(z)+2zBB^{\bullet}} \\ (z^{\bullet}, \Gamma B(z), \Gamma B(z)), & \text{w.p. } \frac{z^{\bullet}B^2}{z+2z^{\bullet}B^2(z)+2zBB^{\bullet}} \\ (z, \Gamma B^{\bullet}(z), \Gamma B(z)), & \text{w.p. } \frac{zBB^{\bullet}}{z+2z^{\bullet}B^2(z)+2zBB^{\bullet}} \\ (z, \Gamma B(z), \Gamma B^{\bullet}(z)), & \text{w.p. } \frac{zBB^{\bullet}}{z+2z^{\bullet}B^2(z)+2zBB^{\bullet}} \end{cases}$$

Pointing operator

Select and mark any node inside binary tree

$$\boxed{B(z)^{\bullet} := z\frac{d}{dz}B(z)}$$

- $B(z) = z + zB^2(z)$
- $B^{\bullet}(z) = z^{\bullet} + z^{\bullet}B^2(z) + zB^{\bullet}(z)B(z) + zB(z)B^{\bullet}(z)$
- Boltzmann sampler $\Gamma B^{\bullet}(z)$

$$\Gamma B^{\bullet}(z) := \begin{cases} z^{\bullet}, & \text{w.p. } \frac{z}{z+2z^{\bullet}B^2(z)+2zBB^{\bullet}} \\ (z^{\bullet}, \Gamma B(z), \Gamma B(z)), & \text{w.p. } \frac{z^{\bullet}B^2}{z+2z^{\bullet}B^2(z)+2zBB^{\bullet}} \\ (z, \Gamma B^{\bullet}(z), \Gamma B(z)), & \text{w.p. } \frac{zBB^{\bullet}}{z+2z^{\bullet}B^2(z)+2zBB^{\bullet}} \\ (z, \Gamma B(z), \Gamma B^{\bullet}(z)), & \text{w.p. } \frac{zBB^{\bullet}}{z+2z^{\bullet}B^2(z)+2zBB^{\bullet}} \end{cases}$$

Pointing operator

Select and mark any node inside binary tree

$$\boxed{B(z)^{\bullet} := z\frac{d}{dz}B(z)}$$

- $B(z) = z + zB^2(z)$
- $B^{\bullet}(z) = z^{\bullet} + z^{\bullet}B^2(z) + zB^{\bullet}(z)B(z) + zB(z)B^{\bullet}(z)$
- Boltzmann sampler $\Gamma B^{\bullet}(z)$

$$\Gamma B^{\bullet}(z) := \begin{cases} z^{\bullet}, & \text{w.p. } \frac{z}{z+2z^{\bullet}B^2(z)+2zBB^{\bullet}} \\ (z^{\bullet}, \Gamma B(z), \Gamma B(z)), & \text{w.p. } \frac{z^{\bullet}B^2}{z+2z^{\bullet}B^2(z)+2zBB^{\bullet}} \\ (z, \Gamma B^{\bullet}(z), \Gamma B(z)), & \text{w.p. } \frac{zBB^{\bullet}}{z+2z^{\bullet}B^2(z)+2zBB^{\bullet}} \\ (z, \Gamma B(z), \Gamma B^{\bullet}(z)), & \text{w.p. } \frac{zBB^{\bullet}}{z+2z^{\bullet}B^2(z)+2zBB^{\bullet}} \end{cases}$$

Rejection rate for binary trees

Theorem Rejection from outside size interval $[n(1-\varepsilon), n(1+\varepsilon)]$ with probability $C(\varepsilon)$. [Duchon, Flajolet, Louchard, Schaeffer]

Pointing transformations can be handled automatically.

Multivariate Boltzmann samplers

$$A(x, y, z) = \sum_{i,j,k \geq 0} a_{ijk} x^i y^j z^k$$

- Combinatorial family $\mathcal{F}$.
- $X \in \mathcal{F}$ — objects with *size parameters* i, j, k.
  Example: *trees* with
  - number of nodes $= i$
  - number of leaves $= j$
  - number of vertices of degree $3 = k$
- $a_{ijk} = \#$ objects of size (i, j, k).
- Want probability distribution (parametrized by (x, y, z)):

$$\mathbb{P}(\text{object } X \text{ with size } (i,j,k)) = \frac{x^i y^j z^k}{A(x,y,z)}$$

Multivariate Boltzmann samplers

$$A(x, y, z) = \sum_{i,j,k \geq 0} a_{ijk} x^i y^j z^k$$

- Combinatorial family $\mathcal{F}$.
- $X \in \mathcal{F}$ — objects with *size parameters* i, j, k.

  Example: *trees* with
  - number of nodes $= i$
  - number of leaves $= j$
  - number of vertices of degree $3 = k$
- $a_{ijk} = \#$ objects of size (i, j, k).
- Want probability distribution (parametrized by (x, y, z)):

$$\mathbb{P}(\text{object } X \text{ with size } (i, j, k)) = \frac{x^i y^j z^k}{A(x, y, z)}$$

Multivariate Boltzmann samplers

$$A(x, y, z) = \sum_{i,j,k \geq 0} a_{ijk} x^i y^j z^k$$

- Combinatorial family $\mathcal{F}$.
- $X \in \mathcal{F}$ — objects with *size parameters* i, j, k.
  Example: *trees* with
  - number of nodes $= i$
  - number of leaves $= j$
  - number of vertices of degree $3 = k$
- $a_{ijk} = \#$ objects of size (i, j, k).
- Want probability distribution (parametrized by (x, y, z)):

$$\mathbb{P}(\text{object } X \text{ with size } (i, j, k)) = \frac{x^i y^j z^k}{A(x, y, z)}$$

Multivariate Boltzmann samplers

$$A(x,y,z) = \sum_{i,j,k\geq 0} a_{ijk} x^i y^j z^k$$

- Combinatorial family $\mathcal{F}$.
- $X \in \mathcal{F}$ — objects with *size parameters* i, j, k.
  Example: *trees* with
  - number of nodes $= i$
  - number of leaves $= j$
  - number of vertices of degree $3 = k$
- $a_{ijk} = \#$ objects of size (i,j,k).
- Want probability distribution (parametrized by (x,y,z)):

$$\mathbb{P}(\text{object } X \text{ with size } (i,j,k)) = \frac{x^i y^j z^k}{A(x,y,z)}$$

Multivariate Boltzmann samplers

$$A(x, y, z) = \sum_{i,j,k \geq 0} a_{ijk} x^i y^j z^k$$

- Combinatorial family $\mathcal{F}$.
- $X \in \mathcal{F}$ — objects with *size parameters* i, j, k.

  Example: *trees* with
  - number of nodes $= i$
  - number of leaves $= j$
  - number of vertices of degree $3 = k$
- $a_{ijk} = \#$ objects of size (i, j, k).
- Want probability distribution (parametrized by (x, y, z)):

$$\mathbb{P}(\text{object } X \text{ with size } (i, j, k)) = \frac{x^i y^j z^k}{A(x, y, z)}$$

Multivariate Boltzmann samplers

$$A(x, y, z) = \sum_{i,j,k \geq 0} a_{ijk} x^i y^j z^k$$

- Combinatorial family $\mathcal{F}$.
- $X \in \mathcal{F}$ — objects with *size parameters* i, j, k.

  Example: *trees* with

  - number of nodes $= i$
  - number of leaves $= j$
  - number of vertices of degree $3 = k$
- $a_{ijk} = \#$ objects of size (i, j, k).
- Want probability distribution (parametrized by (x, y, z)):

$$\mathbb{P}(\text{object } X \text{ with size } (i, j, k)) = \frac{x^i y^j z^k}{A(x, y, z)}$$

Multivariate Boltzmann samplers

$$A(x, y, z) = \sum_{i,j,k \geq 0} a_{ijk} x^i y^j z^k$$

- Combinatorial family $\mathcal{F}$.
- $X \in \mathcal{F}$ — objects with *size parameters* i, j, k.

  Example: *trees* with
  - number of nodes $= i$
  - number of leaves $= j$
  - number of vertices of degree $3 = k$
- a_{ijk} = # objects of size (i, j, k).
- Want probability distribution (parametrized by (x, y, z)):

$$\mathbb{P}(\text{object } X \text{ with size } (i, j, k)) = \frac{x^i y^j z^k}{A(x, y, z)}$$

Multivariate Boltzmann samplers

Example

$$\begin{cases} A = 1 + xy^2B^2 + \dfrac{zBC}{1 - yC} + ABD^2, \\ B = x + A^3 + CD, \\ C = \dfrac{y}{1 - yz} + C^3 + AD, \\ D = B + C^4 \end{cases}$$

- *Linear-time generation* in size.
- Expected size of objects in A:

$$\mathbb{E}(i, j, k) = \frac{1}{A(x, y, z)} \cdot \left[x\frac{\partial}{\partial x}, y\frac{\partial}{\partial y}, z\frac{\partial}{\partial z}\right] A(x, y, z)$$

Multivariate Boltzmann samplers

Example

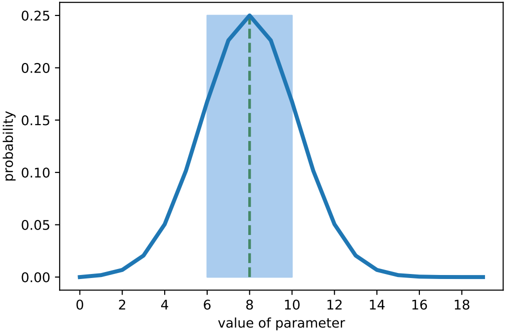
$$\begin{cases} A = 1 + xy^2B^2 + \dfrac{zBC}{1 - yC} + ABD^2, \\ B = x + A^3 + CD, \\ C = \dfrac{y}{1 - yz} + C^3 + AD, \\ D = B + C^4 \end{cases}$$

- *Linear-time generation* in size.
- Expected size of objects in A:

$$\mathbb{E}(i, j, k) = \frac{1}{A(x, y, z)} \cdot \left[x\frac{\partial}{\partial x}, y\frac{\partial}{\partial y}, z\frac{\partial}{\partial z} \right] A(x, y, z)$$

What is the distribution of parameters?

Typically Gaussian with deviation $\Theta(\sqrt{n})$



Question

$$\begin{cases} A = 1 + xy^2B^2 + \dfrac{zBC}{1-yC} + ABD^2, \\ B = x + A^3 + CD, \\ C = \dfrac{y}{1-yz} + C^3 + AD, \\ D = B + C^4 \end{cases}$$

- Oracle for computing $A(x, y, z)$ and its derivatives?
- What if you have thousand of equations and / or variables?

Question

$$
\begin{cases}
A = 1 + xy^2B^2 + \dfrac{zBC}{1 - yC} + ABD^2, \\
B = x + A^3 + CD, \\
C = \dfrac{y}{1 - yz} + C^3 + AD, \\
D = B + C^4
\end{cases}
$$

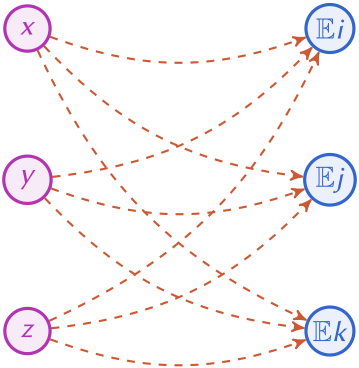
- Oracle for computing $A(x, y, z)$ and its derivatives?
- What if you have thousand of equations and / or variables?

Other questions so far?

Outline

Idea of tuning

$\boxed{\text{Handles}}$ $\Rightarrow$ $\boxed{\text{Expectations}}$



- Tuning is inverse problem. How to solve it?

Oracle for computing the function

- Combinatorial system:

$$\begin{cases} F_1 = \Phi_1(F_1, \ldots, F_m, x, y, z), \\ \ldots \\ F_m = \Phi_m(F_1, \ldots, F_m, x, y, z) \end{cases}$$

- Optimization system

$$\begin{cases} F_1 \to \max, \\ F_1 \geq \Phi_1(F_1, \ldots, F_m, x, y, z), \\ \ldots \\ F_m \geq \Phi_m(F_1, \ldots, F_m, x, y, z) \end{cases} \qquad (*_{x,y,z})$$

- Not convex! [Example: $\Phi(x, y) := x \cdot y$]

Oracle for computing the function

- Combinatorial system:

$$\begin{cases} F_1 = \Phi_1(F_1, \dots, F_m, x, y, z), \\ \dots \\ F_m = \Phi_m(F_1, \dots, F_m, x, y, z) \end{cases}$$

- Optimization system

$$\begin{cases} F_1 \to \max, \\ F_1 \geq \Phi_1(F_1, \dots, F_m, x, y, z), \\ \dots \\ F_m \geq \Phi_m(F_1, \dots, F_m, x, y, z) \end{cases} \qquad (*_{x,y,z})$$

- Not convex! [Example: $\Phi(x, y) := x \cdot y$]

Oracle for computing the function

- Combinatorial system:

$$\begin{cases} F_1 = \Phi_1(F_1, \ldots, F_m, x, y, z), \\ \ldots \\ F_m = \Phi_m(F_1, \ldots, F_m, x, y, z) \end{cases}$$

- Optimization system

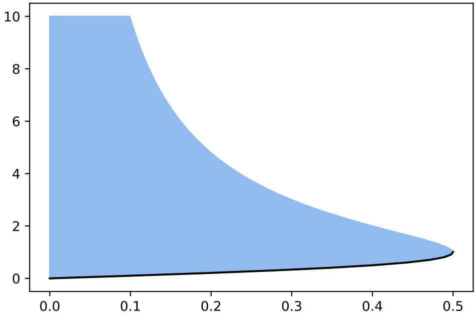
$$\begin{cases} F_1 \to \max, \\ F_1 \geq \Phi_1(F_1, \ldots, F_m, x, y, z), \\ \ldots \\ F_m \geq \Phi_m(F_1, \ldots, F_m, x, y, z) \end{cases} \qquad (*_{x,y,z})$$

- Not convex! [Example: $\Phi(x, y) := x \cdot y$]

Optimization set

Example

$$\begin{cases} B \to \min \\ B \geq z + zB^2 \end{cases} \qquad (*_z)$$



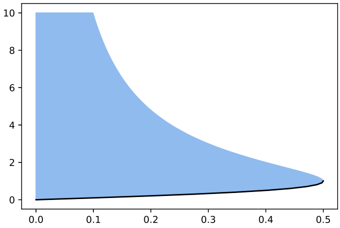
Log-exp transform

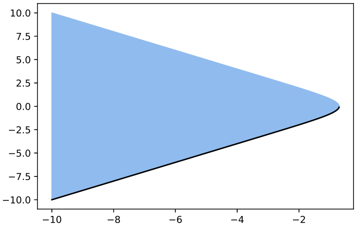
$$\begin{cases} f_1 \to \max, \\ f_1 \geq \log \Phi_1(e^{f_1}, \ldots, e^{f_m}, e^{\xi}, e^{\eta}, e^{\zeta}), \\ \ldots \\ f_m \geq \log \Phi_m(e^{f_1}, \ldots, e^{f_m}, e^{\xi}, e^{\eta}, e^{\zeta}) \end{cases} \qquad (*_{\xi,\eta,\zeta})$$

Optimization set continued

Good old binary trees

$$\begin{cases} b \to \min \\ b \geq \log(e^{\zeta} + e^{\zeta} e^{2b}) \end{cases} \qquad (*_\zeta)$$





Local summary

Compute the function

$$\begin{cases} B \to \min \\ B \geq z + zB^2 \end{cases} \quad (*_z)$$

Option: compute the singularity

$$\begin{cases} z \to \max \\ B \geq z + zB^2 \end{cases} \quad (*)$$

Oracle for tuning the expectation

- (i, j, k) fixed, (x, y, z) unknown.

$$\frac{1}{A(x,y,z)} \cdot \left[x\frac{\partial}{\partial x}, y\frac{\partial}{\partial y}, z\frac{\partial}{\partial z} \right] A(x,y,z) = (i,j,k)$$

- Equivalent formulation

$$\left[\frac{\partial}{\partial \xi}, \frac{\partial}{\partial \eta}, \frac{\partial}{\partial \zeta} \right] \log A(e^{\xi}, e^{\eta}, e^{\zeta}) = (i,j,k)$$

- As *convex* optimization problem

$$\log A(e^{\xi}, e^{\eta}, e^{\zeta}) - (\xi, \eta, \zeta)(i,j,k)^{\top} \to \min$$

Oracle for tuning the expectation

- (i, j, k) fixed, (x, y, z) unknown.

$$\frac{1}{A(x,y,z)} \cdot \left[x\frac{\partial}{\partial x}, y\frac{\partial}{\partial y}, z\frac{\partial}{\partial z} \right] A(x,y,z) = (i,j,k)$$

- Equivalent formulation

$$\left[\frac{\partial}{\partial \xi}, \frac{\partial}{\partial \eta}, \frac{\partial}{\partial \zeta} \right] \log A(e^{\xi}, e^{\eta}, e^{\zeta}) = (i,j,k)$$

- As *convex* optimization problem

$$\log A(e^{\xi}, e^{\eta}, e^{\zeta}) - (\xi, \eta, \zeta)(i,j,k)^{\top} \to \min$$

Oracle for tuning the expectation

- (i, j, k) fixed, (x, y, z) unknown.

$$\frac{1}{A(x,y,z)} \cdot \left[x\frac{\partial}{\partial x}, y\frac{\partial}{\partial y}, z\frac{\partial}{\partial z} \right] A(x,y,z) = (i,j,k)$$

- Equivalent formulation

$$\left[\frac{\partial}{\partial \xi}, \frac{\partial}{\partial \eta}, \frac{\partial}{\partial \zeta} \right] \log A(e^{\xi}, e^{\eta}, e^{\zeta}) = (i,j,k)$$

- As *convex* optimization problem

$$\log A(e^{\xi}, e^{\eta}, e^{\zeta}) - (\xi, \eta, \zeta)(i,j,k)^{\top} \to \min$$

How to solve convex optimization problems?

In [2]:
```
import cvxopt
from cvxopt import matrix
```

[Nesterov, Nemirovskii '1994]
Convex optimization can be solved with precision ε in

$$O\left(\#\text{variables}^3(\#\text{variables} + \#\text{equations}) \log \frac{1}{\varepsilon}\right)$$

using *self-concordant barrier* techniques.

Local summary

$\boxed{\text{Main Theorem Of This Talk}}$ Boltzmann sampler can be tuned to return objects of expected size n in time

$$O(\#\text{variables}^4 \cdot \#\text{classes}^4 \cdot \log^2 n)$$

1. $O(\#\text{varaiables}^4 \cdot \log n)$ steps for tuning procedure
2. $O(\#\text{classes}^4 \cdot \log n)$ steps for oracle inside tuning

So what?

- So what is “learning from context-free grammars”?
- What are the “new applications”?

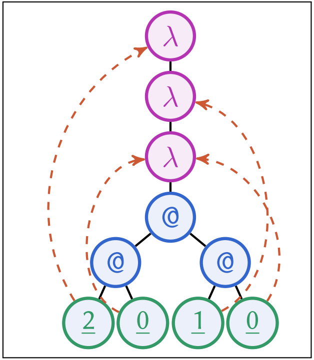
So what?

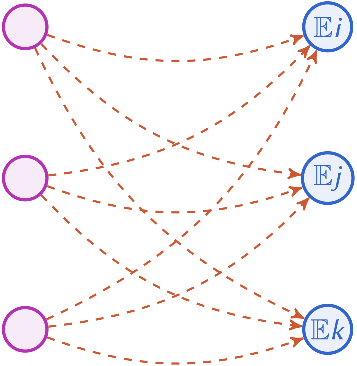
- So what is "learning from context-free grammars"?
- What are the "new applications"?

Outline

Lambda terms for program verification

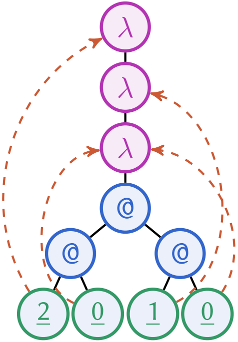
Handles $\Rightarrow$ Expectations





Lambda terms for program verification

Total control



- Number of abstractions
- Number of variables
- De Bruijn index distribution
- Number of redexes
- Number of head abstractions
- Number of closed subterms

Profit. Generate skewed distributions to find bugs in optimizing compilers

Generating functions for plain lambda terms

Example

$$\begin{cases} L(z,\vec{u}) = u_\lambda z^c L(z,\vec{u}) + A(z,\vec{u}), \\ A(z,\vec{u}) = \dfrac{u_{\mathcal{D}} z^a}{1-z^b} + u_r u_\lambda z^{c+d} L(z,\vec{u})^2 + z^d A(z,\vec{u}) L(z,\vec{u}) \end{cases} .$$

- z marks the size of lambda term
- u_λ marks number of abstractions
- $u_{\mathcal{D}}$ marks number of variables
- u_r marks number of redexes

For closed terms you need to consider 29 equations instead of 2.

Maximum Likelihood Model

$\boxed{\text{Boltzmann distribution}} \Rightarrow \boxed{\text{Tree-like structures}}$

$\boxed{\text{Tree-like structures}} \Rightarrow \boxed{\text{Parameters}}$

Inverse problem (mathematical statistics):

$\boxed{\text{Source distribution}} \Leftarrow \boxed{\text{Parameters}}$

Maximum Likelihood Model

$$(X_1, \ldots, X_n) - \text{i.i.d parameters from Boltzmann } \left(\mathbb{P}_\theta,\ \theta \in \Theta\right)$$

$$\widetilde{\theta} = \arg\max L(X_1, \ldots, X_n; \theta)$$

- The source distribution can be now efficiently estimated!
- A bunch of related problems: regression, classification, factor analysis, etc. can be automatically solved
- Meaningful data analysis if model is carefully chosen.
- Relationship between *information matrix* and covariance matrix of Boltzmann random vector

Maximum Likelihood Model

$$(X_1, \ldots, X_n) - \text{i.i.d parameters from Boltzmann} \left(\mathbb{P}_\theta,\ \theta \in \Theta\right)$$

$$\widetilde{\theta} = \arg\max L(X_1, \ldots, X_n; \theta)$$

- The source distribution can be now efficiently estimated!
- A bunch of related problems: regression, classification, factor analysis, etc. can be automatically solved
- Meaningful data analysis if model is carefully chosen.
- Relationship between *information matrix* and covariance matrix of Boltzmann random vector

Maximum Likelihood Model

$$(X_1, \ldots, X_n) - \text{i.i.d parameters from Boltzmann } \left(\mathbb{P}_\theta,\ \theta \in \Theta\right)$$

$$\widetilde{\theta} = \arg\max L(X_1, \ldots, X_n; \theta)$$

- The source distribution can be now efficiently estimated!
- A bunch of related problems: regression, classification, factor analysis, etc. can be automatically solved
- Meaningful data analysis if model is carefully chosen.
- Relationship between *information matrix* and covariance matrix of Boltzmann random vector

Maximum Likelihood Model

$$(X_1, \ldots, X_n) - \text{i.i.d parameters from Boltzmann } \left(\mathbb{P}_\theta,\ \theta \in \Theta\right)$$

$$\widetilde{\theta} = \arg\max L(X_1, \ldots, X_n; \theta)$$

- The source distribution can be now efficiently estimated!
- A bunch of related problems: regression, classification, factor analysis, etc. can be automatically solved
- Meaningful data analysis if model is carefully chosen.
- Relationship between *information matrix* and covariance matrix of Boltzmann random vector

Final summary

1. Polynomial-time Boltzmann tuning
2. Relationships between Boltzmann samplers and classical mathematical statistics
3. Description of singular manifold through convex analysis: "The singular manifold is the Pareto set of multi-objective optimization"
4. Highly flexible sampling of context-free unambiguous grammars, in specie closed lambda terms in de Bruijn notation

Final summary

1. Polynomial-time Boltzmann tuning
2. Relationships between Boltzmann samplers and classical mathematical statistics
3. Description of singular manifold through convex analysis:
   "The singular manifold is the Pareto set of multi-objective optimization"
4. Highly flexible sampling of context-free unambiguous grammars, in specie closed lambda terms in de Bruijn notation

Final summary

1. Polynomial-time Boltzmann tuning
2. Relationships between Boltzmann samplers and classical mathematical statistics
3. Description of singular manifold through convex analysis:
   "The singular manifold is the Pareto set of multi-objective optimization"
4. Highly flexible sampling of context-free unambiguous grammars, in specie closed lambda terms in de Bruijn notation

Final summary

1. Polynomial-time Boltzmann tuning
2. Relationships between Boltzmann samplers and classical mathematical statistics
3. Description of singular manifold through convex analysis:
   "The singular manifold is the Pareto set of multi-objective optimization"
4. Highly flexible sampling of context-free unambiguous grammars, in specie closed lambda terms in de Bruijn notation

That's all!

Thank you for your attention!