Decision list compression by mild random restrictions

Kewen Wu
Peking University → UC Berkeley

STOC, 2020

Shachar Lovett
UCSD

Jiapeng Zhang
Harvard → USC

Section 1

Introduction

Decision list (DL)

Decision list $L = ((C_1, v_1), (C_2, v_2), \ldots, (C_m, v_m))$ is

If $C_1(x) = \texttt{True}$ **then** output v_1,
else if $C_2(x) = \texttt{True}$ **then** output v_2,
...,
else if $C_m(x) = \texttt{True}$ **then** output v_m.

Decision list (DL)

Decision list $L = ((C_1, v_1), (C_2, v_2), \ldots, (C_m, v_m))$ is

If $C_1(x) = \texttt{True}$ **then** output v_1,
else if $C_2(x) = \texttt{True}$ **then** output v_2,
...,
else if $C_m(x) = \texttt{True}$ **then** output v_m.

- C_i is a conjunction of literals, e.g., $x_1 \land \neg x_2 \land x_4$
- The last rule is default: $C_m \equiv \texttt{True}$
- Its size is the number of rules
- Its width is the maximal number of literals in C_i

Decision list (DL)

Example

Assume $L = ((x_1, 1), (\neg x_2 \wedge x_3, a), (x_1 \wedge x_4, 5), (1, 3))$. Then

- its size is 4;
- its width is 2.

If $x_1 = \texttt{True}$ **then** output 1,
else if $\neg x_2 \wedge x_3 = \texttt{True}$ **then** output a,
else if $x_1 \wedge x_4 = \texttt{True}$ **then** output 5,
else output 3.

Decision list (DL)

Example

Assume $L = ((x_1, 1), (\neg x_2 \wedge x_3, a), (x_1 \wedge x_4, 5), (1, 3))$. Then

- its size is 4;
- its width is 2.

If $x_1 = \texttt{True}$ **then** output 1,
else if $\neg x_2 \wedge x_3 = \texttt{True}$ **then** output a,
~~**else if** $x_1 \wedge x_4 = \texttt{True}$ **then** output 5,~~
else output 3.

Decision list (DL)

Let $L = ((C_1, v_1), \ldots, (C_m, v_m))$ be some width-w DL.

- L generalizes width-w DNFs.

If $v_1 = \cdots = v_{m-1} = 1, v_m = 0$, then $L = C_1 \vee \cdots \vee C_{m-1}$.

Decision list (DL)

Let $L = ((C_1, v_1), \ldots, (C_m, v_m))$ be some width-w DL.

- L generalizes width-w DNFs.

If $v_1 = \cdots = v_{m-1} = 1, v_m = 0$, then $L = C_1 \vee \cdots \vee C_{m-1}$.

- L generalizes width-w CNFs.

If $v_1 = \cdots = v_{m-1} = 0, v_m = 1$, then $L = \neg C_1 \wedge \cdots \wedge \neg C_{m-1}$.

Decision list (DL)

Let $L = ((C_1, v_1), \dots, (C_m, v_m))$ be some width-w DL.

- L generalizes width-w DNFs.

If $v_1 = \cdots = v_{m-1} = 1, v_m = 0$, then $L = C_1 \vee \cdots \vee C_{m-1}$.

- L generalizes width-w CNFs.

If $v_1 = \cdots = v_{m-1} = 0, v_m = 1$, then $L = \neg C_1 \wedge \cdots \wedge \neg C_{m-1}$.

- Actually L can be *strictly* more expressive than width-w DNFs/CNFs.

Main result

Small-width DLs can be approximated by small-size DLs of small width.

Main result

Small-width DLs can be approximated by small-size DLs of small width.

Definition (ε-approximation)

f is ε-approximated by g if $\Pr_{x\sim\{0,1\}^n}[f(x) \neq g(x)] \leq \varepsilon$.

Theorem (Decision list compression)

Any width-w DL can be ε-approximated by a width-w size-s DL.

Main result

Small-width DLs can be approximated by small-size DLs of small width.

Definition (ε-approximation)

f is ε-approximated by g if $\Pr_{x \sim \{0,1\}^n}[f(x) \neq g(x)] \leq \varepsilon$.

Theorem (Decision list compression)

Any width-w DL can be ε-approximated by a width-w size-s DL.

- Gopalan, Meka and Reingold 2013: $s = (w \log(1/\varepsilon))^{O(w)}$ for DNFs.

Main result

Small-width DLs can be approximated by small-size DLs of small width.

Definition (ε-approximation)

f is ε-approximated by g if $\Pr_{x\sim\{0,1\}^n}[f(x) \neq g(x)] \leq \varepsilon$.

Theorem (Decision list compression)

Any width-w DL can be ε-approximated by a width-w size-s DL.

- Gopalan, Meka and Reingold 2013: $s = (w \log(1/\varepsilon))^{O(w)}$ for DNFs.
- Lovett and Zhang 2019: $s = (1/\varepsilon)^{O(w)}$.

Main result

Small-width DLs can be approximated by small-size DLs of small width.

Definition (ε-approximation)

f is ε-approximated by g if $\Pr_{x\sim\{0,1\}^n}[f(x) \neq g(x)] \leq \varepsilon$.

Theorem (Decision list compression)

Any width-w DL can be ε-approximated by a width-w size-s DL.

- Gopalan, Meka and Reingold 2013: $s = (w \log(1/\varepsilon))^{O(w)}$ for DNFs.
- Lovett and Zhang 2019: $s = (1/\varepsilon)^{O(w)}$.
- Now: $s = \mathrm{poly}\binom{2w+\log(1/\varepsilon)}{w}$ and this is tight.

Applications

Corollary (DNF sparsification)

Small-width DNFs can be approximated by small-size DNFs of small width.

Applications

Corollary (DNF sparsification)

Small-width DNFs can be approximated by small-size DNFs of small width.

Corollary (Junta theorem)

Small-width DLs can be approximated by a function depending on few input bits.

Applications

Theorem (Jackson's harmonic sieve 1997)

Small-size DNFs are PAC learnable under the uniform distribution with membership queries.

Corollary (Learning small-width DNFs)

Small-width DNFs are PAC learnable under the uniform distribution with membership queries.

Section 2

Proof overview

More definitions

Let $L = ((C_1, v_1), \dots, (C_m, v_m))$ be a DL.

Definition (Index function)

$\mathrm{Ind}L(x)$ is the index of the first satisfied rule in $L(x)$.

Definition (Useful index)

Index i is useful if there exists some x such that $\mathrm{Ind}L(x) = i$.
$\#\mathrm{useful}\,(L)$ is the number of useful indices in L.

Example

Assume $L = ((x_1, v_1), (x_1 \wedge x_2, v_2), (1, v_3))$.
Then $\mathrm{Ind}L(x_1 = 1, x_2 = 1) = 1$ and $\#\mathrm{useful}\,(L) = 2$.

Step 1: randomness kills structure

We should be able to compress L (in some form) under restrictions.

Step 1: randomness kills structure

We should be able to compress L (in some form) under restrictions.

Lemma (Håstad's switching lemma 1987)

Let f be a width-w DNF, $\alpha \in (0,1)$, and d be an integer. If ρ randomly restricts each input bit to $0, 1, *$ w.p. $(1-\alpha)/2, (1-\alpha)/2, \alpha$, then

$$\Pr_{\rho}\left[\mathrm{DT}(f \restriction_{\rho}) \geq d\right] \leq (5\alpha w)^{d}.$$

Step 1: randomness kills structure

We should be able to compress L (in some form) under restrictions.

Lemma (Håstad's switching lemma 1987)

Let f be a width-w DNF, $\alpha \in (0,1)$, and d be an integer. If ρ randomly restricts each input bit to $0, 1, *$ w.p. $(1-\alpha)/2, (1-\alpha)/2, \alpha$, then

$$\Pr_{\rho}\left[\mathrm{DT}(f\restriction_{\rho}) \geq d\right] \leq (5\alpha w)^d.$$

- Meaningful only when $\alpha \leq O(1/w) \implies$ most bits are fixed.

Step 1: mild randomness also kills structure

Let's directly analyze L's size under restrictions.

Step 1: mild randomness also kills structure

Let's directly analyze L's size under restrictions.

Lemma (Encoding lemma)

Let L be a width-w DL and $\alpha \in (0,1)$.
If ρ randomly restricts each input bit to $0, 1, *$ w.p. $(1-\alpha)/2, (1-\alpha)/2, \alpha$, then

$$\mathop{\mathbb{E}}_{\rho}\left[\#\textit{useful}\,(L\restriction_{\rho})\right] \leq \left(\frac{4}{1-\alpha}\right)^{w}.$$

Step 1: mild randomness also kills structure

Let's directly analyze L's size under restrictions.

Lemma (Encoding lemma)

Let L be a width-w DL and $\alpha \in (0,1)$.
If ρ randomly restricts each input bit to $0, 1, *$ w.p. $(1-\alpha)/2, (1-\alpha)/2, \alpha$, then

$$\mathop{\mathbb{E}}_{\rho}\left[\#\textit{useful}\,(L\restriction_{\rho})\right] \leq \left(\frac{4}{1-\alpha}\right)^{w}.$$

- Meaningful for all kinds of α.
- Prove by encoding ρ together with a useful index in $L\restriction_{\rho}$.

Step 2: compression – redundant rules

Let $L = ((C_1, v_1), \ldots, (C_m, v_m))$ be a width-w DL.

- If index i is not useful, we can safely remove the i-th rule.

Step 2: compression – redundant rules

Let $L = ((C_1, v_1), \ldots, (C_m, v_m))$ be a width-w DL.

- If index i is not useful, we can safely remove the i-th rule.

If $x_1 = \texttt{True}$ **then** output v_1,
else if $x_1 \wedge x_2 = \texttt{True}$ **then** output v_2,
else output v_3.

Step 2: compression – redundant rules

Let $L = ((C_1, v_1), \ldots, (C_m, v_m))$ be a width-w DL.

- If index i is not useful, we can safely remove the i-th rule.

If $x_1 = \texttt{True}$ **then** output v_1,
else if $x_1 \wedge x_2 = \texttt{True}$ **then** output v_2,
else output v_3.

$$\Downarrow$$

If $x_1 = \texttt{True}$ **then** output v_1,
~~**else if** $x_1 \wedge x_2 = \texttt{True}$ **then** output v_2,~~
else output v_3.

Step 2: compression – less useful rules

Let $L = ((C_1, v_1), \ldots, (C_m, v_m))$ be a width-w DL.

- Let $p(i) = \Pr_x [\mathsf{Ind}L(x) = i]$, and sort it in descending order. If p decays fast, we only need to keep the top few rules.

Step 2: compression – less useful rules

Let $L = ((C_1, v_1), \ldots, (C_m, v_m))$ be a width-w DL.

- Let $p(i) = \Pr_x [\mathsf{Ind}L(x) = i]$, and sort it in descending order. If p decays fast, we only need to keep the top few rules.

If $x_1 \wedge x_2 \wedge \cdots \wedge x_w = \texttt{True}$ **then** output v_1,
else if $x_1 = \texttt{True}$ **then** output v_2,
else output v_3.

$$p(1) = 2^{-w}, p(2) \approx 1/2, p(3) \approx 1/2.$$

Step 2: compression – less useful rules

Let $L = ((C_1, v_1), \ldots, (C_m, v_m))$ be a width-w DL.

- Let $p(i) = \Pr_x[\mathrm{Ind}L(x) = i]$, and sort it in descending order. If p decays fast, we only need to keep the top few rules.

If $x_1 \wedge x_2 \wedge \cdots \wedge x_w = \texttt{True}$ **then** output v_1,
else if $x_1 = \texttt{True}$ **then** output v_2,
else output v_3.

$\Downarrow$ *lose* $\varepsilon = 2^{-w}$

~~**If** $x_1 \wedge x_2 \wedge \cdots \wedge x_w = \texttt{True}$ **then** output v_1~~,
~~**else**~~ **if** $x_1 = \texttt{True}$ **then** output v_2,
else output v_3.

Step 2: compression – approximator

Let $L = ((C_1, v_1), \ldots, (C_m, v_m))$ be a width-w DL.

- Assume $p(i) = \Pr_x [\mathsf{Ind}L(x) = i]$ is decreasing in i for simplicity.
- Let approximator $L' = ((C_1, v_1), \ldots, (C_t, v_t), (C_m, v_m))$.

Step 2: compression – approximator

Let $L = ((C_1, v_1), \ldots, (C_m, v_m))$ be a width-w DL.

- Assume $p(i) = \Pr_x [\mathsf{Ind} L(x) = i]$ is decreasing in i for simplicity.
- Let approximator $L' = ((C_1, v_1), \ldots, (C_t, v_t), (C_m, v_m))$.Then it has
  - width w;
  - size $t + 1$;
  - approximation factor

$$\varepsilon = \Pr [L(x) \neq L'(x)] \leq \Pr [\mathsf{Ind} L(x) > t] = \sum_{i>t} p(i).$$

Now what?

Let ρ_α be the random restriction with $*$-probability α.

- What we can do so far?
  We can analyze $q(\alpha, i) = \Pr\left[\text{index } i \text{ is useful in } L\restriction_{\rho_\alpha}\right]$, since

$$\sum_i q(\alpha, i) = \mathbb{E}\left[\#\text{useful}\left(L\restriction_{\rho_\alpha}\right)\right].$$

- What we want to do next?
  We want to bound $p(i) = \Pr\left[\text{Ind}L(x) = i\right]$, since

$$\varepsilon = \Pr\left[L(x) \neq L'(x)\right] \leq \sum_{i>t} p(i).$$

Step 3: noise stability

Let's introduce noise stability to relate $p(i)$ and $q(\alpha, i)$.

Definition (Noise distribution $\mathcal{N}_\beta$)

$y \sim \mathcal{N}_\beta(x)$ is sampled by taking $\Pr[y_i = x_i] = (1+\beta)/2$.

Then for $x \sim \{0,1\}^n, y \sim \mathcal{N}_\beta(x)$, we can also do it by sampling

- 1. common restriction $\rho = \rho_{1-\beta}$ with $*$-probability $1-\beta$.
- 2. x' by uniformly filling out $*$'s in ρ, and set $x = \rho \circ x'$.
- 3. y' by uniformly filling out $*$'s in ρ, and set $y = \rho \circ y'$.

Step 4: bridging lemma

Let $L = ((C_1, v_1), \ldots, (C_m, v_m))$ be a width-w DL and fix an index i.

Step 4: bridging lemma

Let $L = ((C_1, v_1), \ldots, (C_m, v_m))$ be a width-w DL and fix an index i.

Recall

- $p(i) = \Pr[\mathsf{Ind}L(x) = i]$;
- $q(\alpha, i) = \Pr[\text{index } i \text{ is useful in } L\restriction_{\rho_\alpha}]$;
- our goal is to "bridge" between $p(i)$ and $q(\alpha, i)$.

Step 4: bridging lemma

Let $L = ((C_1, v_1), \dots, (C_m, v_m))$ be a width-w DL and fix an index i.

Recall

- $p(i) = \Pr[\mathsf{Ind}L(x) = i]$;
- $q(\alpha, i) = \Pr[\text{index } i \text{ is useful in } L\restriction_{\rho_\alpha}]$;
- our goal is to "bridge" between $p(i)$ and $q(\alpha, i)$.

Sample $x = \rho \circ x' \sim \{0,1\}^n, y = \rho \circ y' \sim \mathcal{N}_\beta(x), \rho = \rho_{1-\beta}$ and define $\mathsf{Stab}(\beta, i) = \Pr\left[\mathsf{Ind}L(x) = \mathsf{Ind}L(y) = i\right]$.

Step 4: bridging lemma

Let $L = ((C_1, v_1), \ldots, (C_m, v_m))$ be a width-w DL and fix an index i.

Recall

- $p(i) = \Pr[\mathsf{Ind}L(x) = i]$;
- $q(\alpha, i) = \Pr[\text{index } i \text{ is useful in } L\restriction_{\rho_\alpha}]$;
- our goal is to "bridge" between $p(i)$ and $q(\alpha, i)$.

Sample $x = \rho \circ x' \sim \{0,1\}^n, y = \rho \circ y' \sim \mathcal{N}_\beta(x), \rho = \rho_{1-\beta}$ and define $\mathsf{Stab}(\beta, i) = \Pr\left[\mathsf{Ind}L(x) = \mathsf{Ind}L(y) = i\right]$.

$\mathsf{Stab}(\beta, i)$ is the bridge.

Step 4: bridging lemma

For upper bound, we have

Fact (Hypercontractivity)

$$\mathit{Stab}(\beta, i) \leq \left(\Pr\left[\mathit{IndL}(x) = i\right]\right)^{2/(1+\beta)} = \left(p(i)\right)^{2/(1+\beta)}.$$

Step 4: bridging lemma

For upper bound, we have

Fact (Hypercontractivity)

$$\mathit{Stab}(\beta, i) \leq \left(\Pr\left[\mathit{IndL}(x) = i\right]\right)^{2/(1+\beta)} = (p(i))^{2/(1+\beta)} .$$

For lower bound, we can prove

Lemma

$$\begin{aligned}\mathit{Stab}(\beta, i) &\geq (\Pr[\mathit{IndL}(x) = i])^2 / \Pr[\mathit{index}\ i\ \mathit{is\ useful\ in}\ L\restriction_\rho] \\ &= (p(i))^2 / q(1-\beta, i).\end{aligned}$$

Step 5: putting everything together

Lemma (Bridging lemma)

$(p(i))^2/q(1-\beta,i) \leq \mathit{Stab}(\beta,i) \leq (p(i))^{2/(1+\beta)}$.

Step 5: putting everything together

Lemma (Bridging lemma)

$(p(i))^2/q(1-\beta,i) \le \mathit{Stab}(\beta,i) \le (p(i))^{2/(1+\beta)}$.

So we get $p(i) \le q(1-\beta,i)^{(1+\beta)/2\beta}$.

Step 5: putting everything together

Lemma (Bridging lemma)

$(p(i))^2/q(1-\beta,i) \leq \textit{Stab}(\beta,i) \leq (p(i))^{2/(1+\beta)}$.

So we get $p(i) \leq q(1-\beta,i)^{(1+\beta)/2\beta}$.

Lemma (Encoding lemma)

$\sum_i q(1-\beta,i) = \mathbb{E}\left[\#\textit{useful}\left(L\restriction_{\rho_{1-\beta}}\right)\right] \leq (4/\beta)^w$.

Step 5: putting everything together

Lemma (Bridging lemma)

$(p(i))^2/q(1-\beta,i) \leq \mathit{Stab}(\beta,i) \leq (p(i))^{2/(1+\beta)}$.

So we get $p(i) \leq q(1-\beta,i)^{(1+\beta)/2\beta}$.

Lemma (Encoding lemma)

$\sum_i q(1-\beta,i) = \mathbb{E}\left[\#\mathit{useful}\left(L \upharpoonright_{\rho_{1-\beta}}\right)\right] \leq (4/\beta)^w$.

So we get $q(1-\beta,i) \leq (4/\beta)^w/i$ assuming q is decreasing in i.

Step 5: putting everything together

Lemma (Bridging lemma)

$(p(i))^2/q(1-\beta,i) \leq \mathit{Stab}(\beta,i) \leq (p(i))^{2/(1+\beta)}$.

So we get $p(i) \leq q(1-\beta,i)^{(1+\beta)/2\beta}$.

Lemma (Encoding lemma)

$\sum_i q(1-\beta,i) = \mathbb{E}\left[\#\mathit{useful}\left(L\restriction_{\rho_{1-\beta}}\right)\right] \leq (4/\beta)^w$.

So we get $q(1-\beta,i) \leq (4/\beta)^w/i$ assuming q is decreasing in i.

Theorem (Final bound)

$\varepsilon = \Pr[L(x) \neq L'(x)] \leq \sum_{i>t} p(i) \leq \sum_{i>t} [(4/\beta)^w/i]^{(1+\beta)/2\beta}$.

Then we choose $\beta = \beta(\varepsilon, w)$ to get optimal t.

Section 3

Open problems

Upper bound compression

Assume L is a width-w DNF.
L' is constructed by removing rules of L, thus $L'(x) \leq L(x)$.

Upper bound compression

Assume L is a width-w DNF.
L' is constructed by removing rules of L, thus $L'(x) \leq L(x)$.

Now we want some DNF L'' such that $L''(x) \geq L(x)$.

Problem (Upper bound compression)

L can be ε-approximated by a width-w size-s DNF from above.

Upper bound compression

Assume L is a width-w DNF.
L' is constructed by removing rules of L, thus $L'(x) \leq L(x)$.

Now we want some DNF L'' such that $L''(x) \geq L(x)$.

Problem (Upper bound compression)

L can be ε-approximated by a width-w size-s DNF from above.

- Gopalan, Meka and Reingold 2013: $s = (w \log(1/\varepsilon))^{O(w)}$.

Upper bound compression

Assume L is a width-w DNF.
L' is constructed by removing rules of L, thus $L'(x) \leq L(x)$.

Now we want some DNF L'' such that $L''(x) \geq L(x)$.

Problem (Upper bound compression)

L can be ε-approximated by a width-w size-s DNF from above.

- Gopalan, Meka and Reingold 2013: $s = (w \log(1/\varepsilon))^{O(w)}$.
- Lovett, Solomon and Zhang 2019: in a restricted case, $s = ((\log w)/\varepsilon)^{O(w)}$ implies improved sunflower lemma.

Upper bound compression

Assume L is a width-w DNF.
L' is constructed by removing rules of L, thus $L'(x) \leq L(x)$.

Now we want some DNF L'' such that $L''(x) \geq L(x)$.

Problem (Upper bound compression)

L can be ε-approximated by a width-w size-s DNF from above.

- Gopalan, Meka and Reingold 2013: $s = (w \log(1/\varepsilon))^{O(w)}$.
- Lovett, Solomon and Zhang 2019: in a restricted case, $s = ((\log w)/\varepsilon)^{O(w)}$ implies improved sunflower lemma.

Alweiss, Lovett, Wu and Zhang [STOC, 2020] gives the improved sunflower lemma, can we improve upper bound compression?

Thanks!