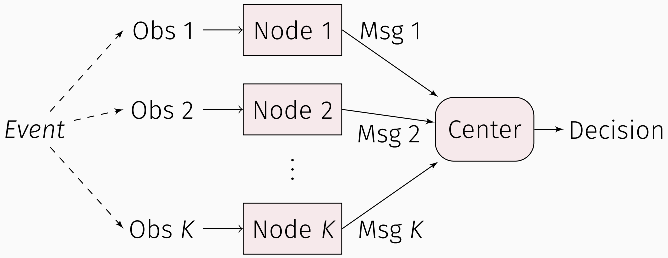
On Distributed Hypothesis Testing with Constant Communication Bits

Xiangxiang Xu

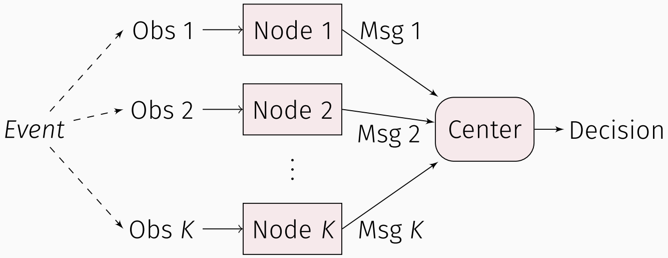
April 15, 2022

Joint work with Shao-Lun Huang

Background: Distributed Inference

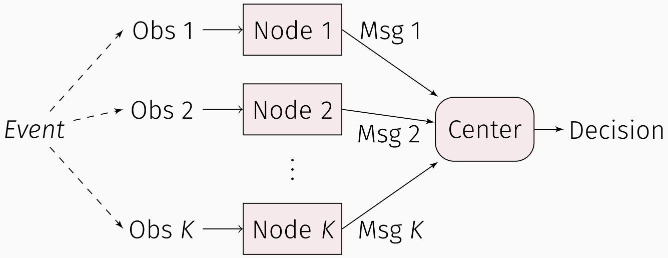


Background: Distributed Inference



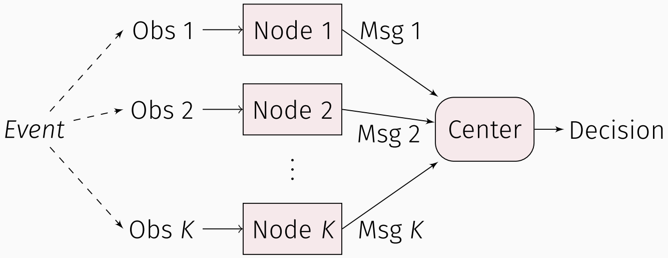
- Distributed Detection

Background: Distributed Inference

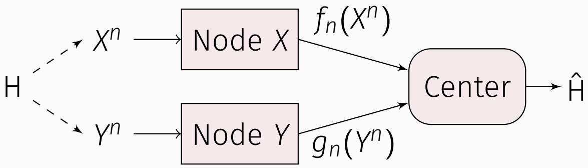


- Distributed Detection
- Sensor Fusion

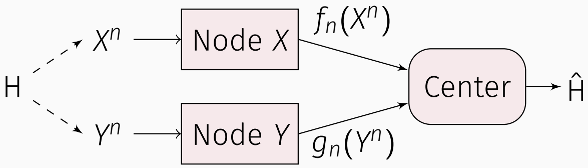
Background: Distributed Inference



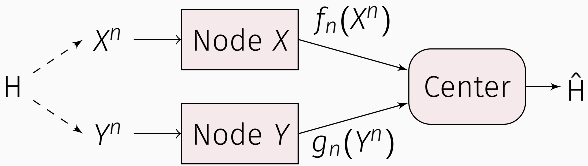
- Distributed Detection
- Sensor Fusion
- Federated Learning


H
X^n
Y^n
Node X
Node Y
$f_n(X^n)$
$g_n(Y^n)$
Center
$\hat{H}$

Problem Formulation: Distributed Hypothesis Testing (DHT)

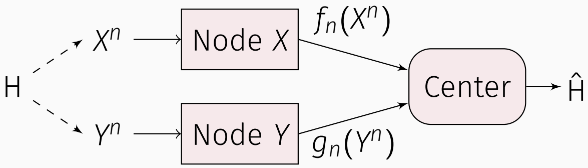


- $H \in \{0, 1\}$: Binary Hypothesis



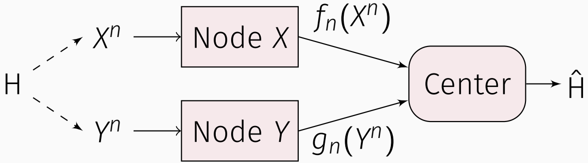
- $H \in \{0, 1\}$: Binary Hypothesis
- $(X^n, Y^n) \overset{\text{i.i.d.}}{\sim} P_{XY}^{(H)}$: Observed Sequences

Problem Formulation: Distributed Hypothesis Testing (DHT)



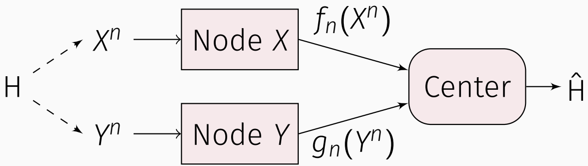
- $H \in \{0, 1\}$: Binary Hypothesis
- $(X^n, Y^n) \overset{\text{i.i.d.}}{\sim} P_{XY}^{(H)}$: Observed Sequences
- $f_n(X^n), g_n(Y^n)$: Transmitted Messages

Problem Formulation: Distributed Hypothesis Testing (DHT)



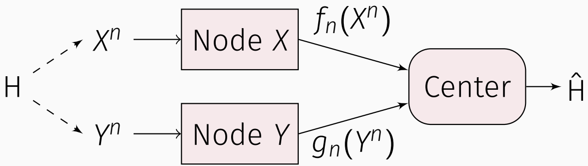
- $\mathrm{H} \in \{0,1\}$: Binary Hypothesis
- $(X^n, Y^n) \overset{\text{i.i.d.}}{\sim} P_{XY}^{(\mathrm{H})}$: Observed Sequences
- $f_n(X^n), g_n(Y^n)$: Transmitted Messages
- $\hat{\mathrm{H}} = \mathrm{dec}(f_n(X^n), g_n(Y^n))$: Decision

Problem Formulation: Distributed Hypothesis Testing (DHT)



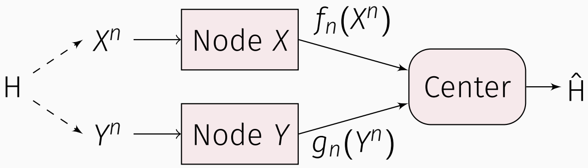
- $\mathrm{H} \in \{0, 1\}$: Binary Hypothesis
- $(X^n, Y^n) \overset{\text{i.i.d.}}{\sim} P_{XY}^{(\mathrm{H})}$: Observed Sequences
- $f_n(X^n), g_n(Y^n)$: Transmitted Messages
- $\hat{\mathrm{H}} = \mathrm{dec}(f_n(X^n), g_n(Y^n))$: Decision
- f_n, g_n: Distributed Encoders

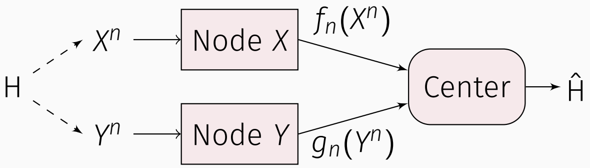
Problem Formulation: Distributed Hypothesis Testing (DHT)



- $\mathrm{H} \in \{0,1\}$: Binary Hypothesis
- $(X^n, Y^n) \overset{\text{i.i.d.}}{\sim} P_{XY}^{(\mathrm{H})}$: Observed Sequences
- $f_n(X^n), g_n(Y^n)$: Transmitted Messages
- $\hat{\mathrm{H}} = \mathrm{dec}(f_n(X^n), g_n(Y^n))$: Decision

- f_n, g_n: Distributed Encoders
- $\mathrm{dec}(\cdot,\cdot)$: Central Decoder


H
X^n
Y^n
Node X
Node Y
$f_n(X^n)$
$g_n(Y^n)$
Center
$\hat{H}$

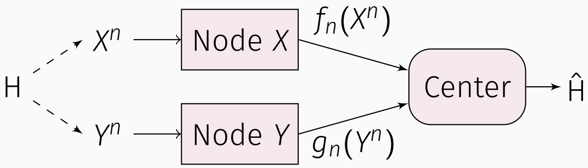


Type-I Error

$$\pi_0 \triangleq \mathbb{P}\left\{\hat{H} \neq 0 \middle| H = 0\right\}$$

Type-II Error

$$\pi_1 \triangleq \mathbb{P}\left\{\hat{H} \neq 1 \middle| H = 1\right\}$$



Type-I Error

$$\pi_0 \triangleq \mathbb{P}\left\{\hat{\mathrm{H}} \neq 0 \middle| \mathrm{H} = 0\right\}$$

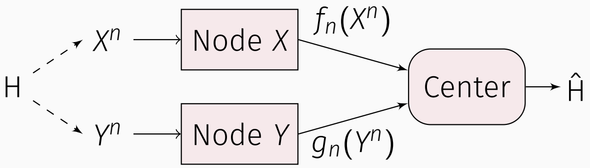
Error exponent E_0: $\pi_0 \doteq \exp(-nE_0)$

Type-II Error

$$\pi_1 \triangleq \mathbb{P}\left\{\hat{\mathrm{H}} \neq 1 \middle| \mathrm{H} = 1\right\}$$

Error exponent E_1: $\pi_1 \doteq \exp(-nE_1)$

Performance Metrics: Errors and Error Exponents



Type-I Error

$$\pi_0 \triangleq \mathbb{P}\left\{\hat{\mathsf{H}} \neq 0 \middle| \mathsf{H} = 0\right\}$$

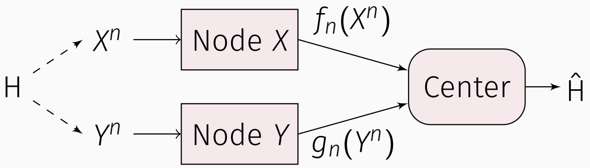
Error exponent E_0: $\pi_0 \doteq \exp(-nE_0)$

Type-II Error

$$\pi_1 \triangleq \mathbb{P}\left\{\hat{\mathsf{H}} \neq 1 \middle| \mathsf{H} = 1\right\}$$

Error exponent E_1: $\pi_1 \doteq \exp(-nE_1)$

Optimal E_1 under $\pi_0 < \epsilon$



Type-I Error

$$\pi_0 \triangleq \mathbb{P}\left\{\hat{H} \neq 0 \middle| H = 0\right\}$$

Error exponent E_0: $\pi_0 \doteq \exp(-nE_0)$

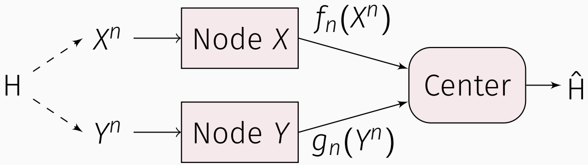
Type-II Error

$$\pi_1 \triangleq \mathbb{P}\left\{\hat{H} \neq 1 \middle| H = 1\right\}$$

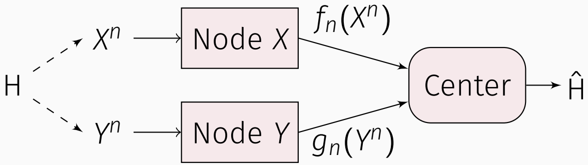
Error exponent E_1: $\pi_1 \doteq \exp(-nE_1)$

Optimal E_1 under $\pi_0 < \epsilon$ or **Optimal (E_0, E_1) Trade-offs**

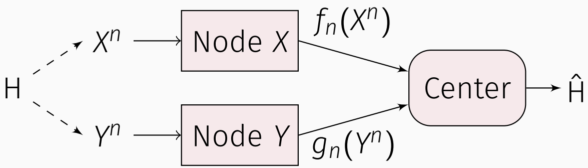
Communication Constraints in DHT



Communication Constraints in DHT



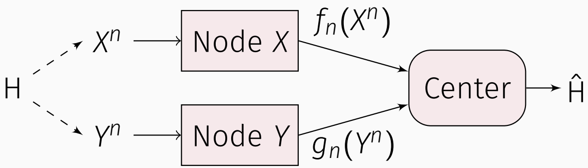
$\|f_n\|$, $\|g_n\|$: sizes of message sets



$\|f_n\|$, $\|g_n\|$: sizes of message sets

Rate Constraints (R_X, R_Y)

$$\|f_n\| \dot{\le} \exp(nR_X), \|g_n\| \dot{\le} \exp(nR_Y)$$

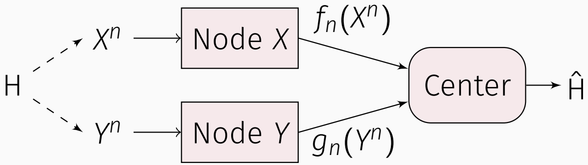


$\|f_n\|$, $\|g_n\|$: sizes of message sets

Rate Constraints (R_X, R_Y)

$$\|f_n\| \stackrel{.}{\leq} \exp(nR_X), \|g_n\| \stackrel{.}{\leq} \exp(nR_Y)$$

e.g. $(0, 0)$ *zero-rate compression*



$\|f_n\|$, $\|g_n\|$: sizes of message sets

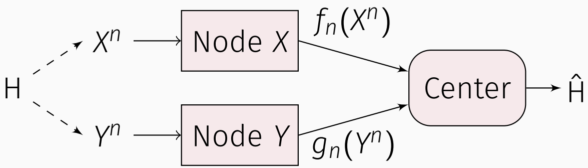
Rate Constraints (R_X, R_Y)

$$\|f_n\| \dot{\leq} \exp(nR_X), \|g_n\| \dot{\leq} \exp(nR_Y)$$

e.g. $(0, 0)$ *zero-rate compression*

Constant Constraints $(0_{M_X}, 0_{M_Y})$

$$\|f_n\| \leq M_X, \|g_n\| \leq M_Y$$



$\|f_n\|$, $\|g_n\|$: sizes of message sets

Rate Constraints (R_X, R_Y)

$$\|f_n\| \dot{\leq} \exp(nR_X), \|g_n\| \dot{\leq} \exp(nR_Y)$$

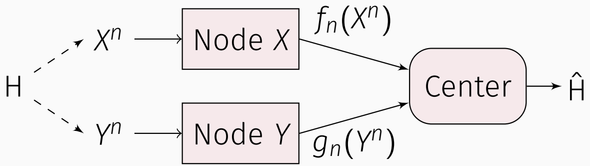
e.g. $(0, 0)$ *zero-rate compression*

Constant Constraints $(0_{M_X}, 0_{M_Y})$

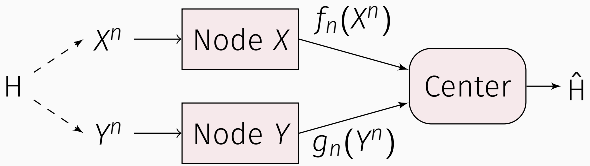
$$\|f_n\| \leq M_X, \|g_n\| \leq M_Y$$

e.g. $(0_2, 0_2)$ *one-bit compression*

Existing Results

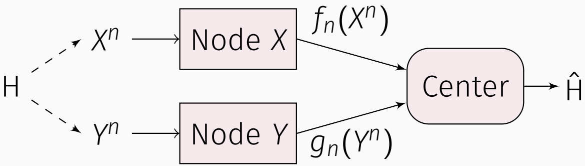


Constraints	Opt. E_1 under $\pi_0 < \epsilon$	Opt. (E_0, E_1) Trade-offs
$(R_X, \log |\mathcal{Y}|)$	[Ahlswede & Csiszár, 1986]	
(R_X, R_Y)		
$(0, 0)$		
$(0_2, 0_2)$		
$(0_2, R_Y)$		



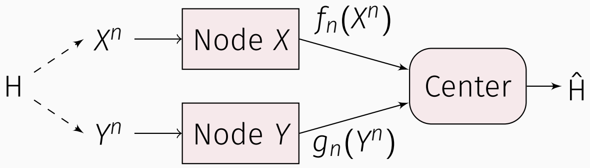
Constraints	**Opt.** E_1 **under** $\pi_0 < \epsilon$	**Opt.** (E_0, E_1) **Trade-offs**
$(R_X, \log \lvert\mathcal{Y}\rvert)$	[Ahlswede & Csiszár, 1986]	
(R_X, R_Y)	[Han, 1987]	
$(0, 0)$	[Han, 1987] ✓$_D$	
$(0_2, 0_2)$	[Han, 1987] ✓	
$(0_2, R_Y)$	[Han, 1987] ✓	

Existing Results



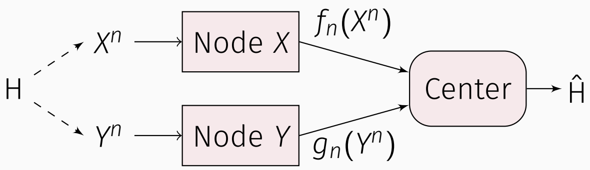
Constraints	**Opt.** E_1 **under** $\pi_0 < \epsilon$	**Opt.** (E_0, E_1) **Trade-offs**
$(R_X, \log\lvert\mathcal{Y}\rvert)$	[Ahlswede & Csiszár, 1986]	[Han & Kobayashi, 1989]
(R_X, R_Y)	[Han, 1987]	[Han & Kobayashi, 1989]
$(0, 0)$	[Han, 1987] ✓$_D$	[Han & Kobayashi, 1989] ✓$_D$
$(0_2, 0_2)$	[Han, 1987] ✓	[Han & Kobayashi, 1989] ✓
$(0_2, R_Y)$	[Han, 1987] ✓	

Existing Results

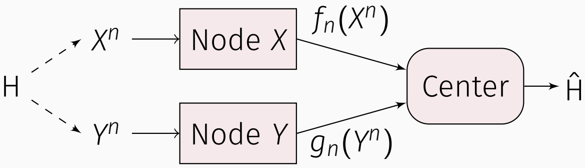


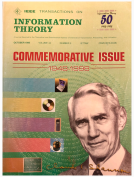
Constraints	Opt. E_1 under $\pi_0 < \epsilon$	Opt. (E_0, E_1) Trade-offs
$(R_X, \log \lvert\mathcal{Y}\rvert)$	[Ahlswede & Csiszár, 1986]	[Han & Kobayashi, 1989]
(R_X, R_Y)	[Han, 1987]	[Han & Kobayashi, 1989]
$(0, 0)$	[Han, 1987] $\checkmark_D$ [Shalaby & Papamarcou, 1992] $\checkmark_C$	[Han & Kobayashi, 1989] $\checkmark_D$
$(0_2, 0_2)$	[Han, 1987] $\checkmark$	[Han & Kobayashi, 1989] $\checkmark$
$(0_2, R_Y)$	[Han, 1987] $\checkmark$	

Existing Results

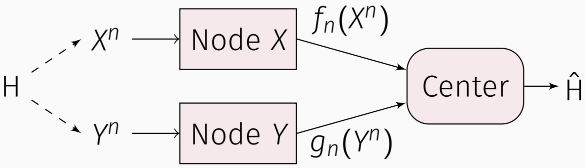


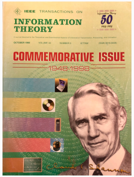
Constraints	Opt. E_1 under $\pi_0 < \epsilon$	Opt. (E_0, E_1) Trade-offs
$(R_X, \log \lvert\mathcal{Y}\rvert)$	[Ahlswede & Csiszár, 1986]	[Han & Kobayashi, 1989]
(R_X, R_Y)	[Han, 1987]	[Han & Kobayashi, 1989]
$(0, 0)$	[Han, 1987] ✓$_D$ [Shalaby & Papamarcou, 1992] ✓$_C$	[Han & Kobayashi, 1989] ✓$_D$ [Han & Amari, 1998] ✓$_C$
$(0_2, 0_2)$	[Han, 1987] ✓	[Han & Kobayashi, 1989] ✓
$(0_2, R_Y)$	[Han, 1987] ✓	





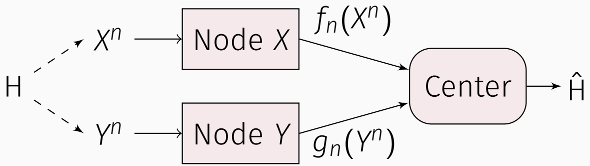
Constraints	**Opt.** E_1 **under** $\pi_0 < \epsilon$	**Opt.** (E_0, E_1) **Trade-offs**
$(R_X, \log \lvert \mathcal{Y} \rvert)$	[Ahlswede & Csiszár, 1986]	[Han & Kobayashi, 1989]
(R_X, R_Y)	[Han, 1987]	[Han & Kobayashi, 1989]
$(0, 0)$	[Han, 1987] $\checkmark_D$ [Shalaby & Papamarcou, 1992] $\checkmark_C$	[Han & Kobayashi, 1989] $\checkmark_D$ [Han & Amari, 1998] $\checkmark_C$
$(0_2, 0_2)$	[Han, 1987] $\checkmark$	[Han & Kobayashi, 1989] $\checkmark$
$(0_2, R_Y)$	[Han, 1987] $\checkmark$	

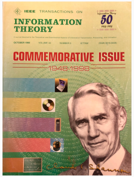




Constraints	Opt. E_1 under $\pi_0 < \epsilon$	Opt. (E_0, E_1) Trade-offs
$(R_X, \log\lvert\mathcal{Y}\rvert)$	[Ahlswede & Csiszár, 1986]	[Han & Kobayashi, 1989]
(R_X, R_Y)	[Han, 1987]	[Han & Kobayashi, 1989]
$(0, 0)$	[Han, 1987] $\checkmark_D$ [Shalaby & Papamarcou, 1992] $\checkmark_C$	[Han & Kobayashi, 1989] $\checkmark_D$ [Han & Amari, 1998] $\checkmark_C$
$(0_2, 0_2)$	[Han, 1987] $\checkmark$	[Han & Kobayashi, 1989] $\checkmark$
$(0_2, R_Y)$	[Han, 1987] $\checkmark$	
$(0_{M_X}, 0_{M_Y})$	trivial	[This work]

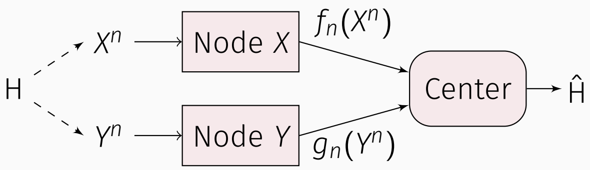
Existing Results

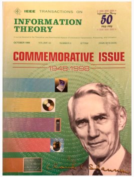




Constraints	Opt. E_1 under $\pi_0 < \epsilon$	Opt. (E_0, E_1) Trade-offs
$(R_X, \log \lvert\mathcal{Y}\rvert)$	[Ahlswede & Csiszár, 1986]	[Han & Kobayashi, 1989]
(R_X, R_Y)	[Han, 1987]	[Han & Kobayashi, 1989]
$(0, 0)$	[Han, 1987] ✓$_D$ [Shalaby & Papamarcou, 1992] ✓$_C$	[Han & Kobayashi, 1989] ✓$_D$ [Han & Amari, 1998] ✓$_C$
$(0_2, 0_2)$	[Han, 1987] ✓	[Han & Kobayashi, 1989] ✓
$(0_2, R_Y)$	[Han, 1987] ✓	
$(0_3, 0_3)$	trivial	[This work] ✓

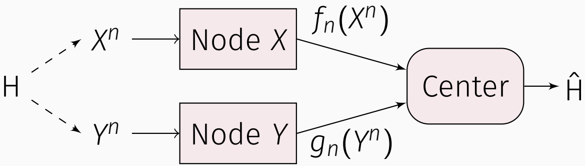
Existing Results

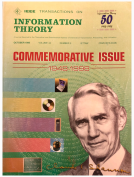




Constraints	Opt. E_1 under $\pi_0 < \epsilon$	Opt. (E_0, E_1) Trade-offs
$(R_X, \log\lvert\mathcal{Y}\rvert)$	[Ahlswede & Csiszár, 1986]	[Han & Kobayashi, 1989]
(R_X, R_Y)	[Han, 1987]	[Han & Kobayashi, 1989]
$(0, 0)$	[Han, 1987] $\checkmark_D$ [Shalaby & Papamarcou, 1992] $\checkmark_C$	[Han & Kobayashi, 1989] $\checkmark_D$ [Han & Amari, 1998] $\checkmark_C$
$(0_2, 0_2)$	[Han, 1987] ✓	[Han & Kobayashi, 1989] ✓
$(0_2, R_Y)$	[Han, 1987] ✓	[This work] ✓
$(0_3, 0_3)$	trivial	[This work] ✓

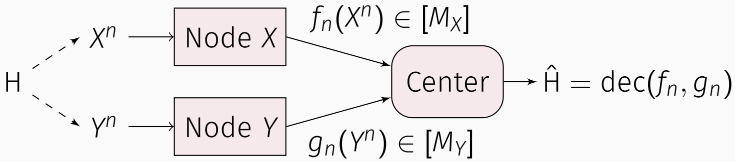
Existing Results



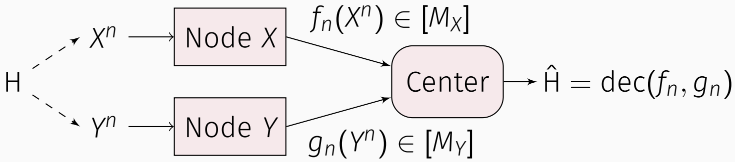


Constraints	Opt. E_1 under $\pi_0 < \epsilon$	Opt. (E_0, E_1) Trade-offs
$(R_X, \log \lvert\mathcal{Y}\rvert)$	[Ahlswede & Csiszár, 1986]	[Han & Kobayashi, 1989]
(R_X, R_Y)	[Han, 1987]	[Han & Kobayashi, 1989]
$(0, 0)$	[Han, 1987] $\checkmark_D$ [Shalaby & Papamarcou, 1992] $\checkmark_C$	[Han & Kobayashi, 1989] $\checkmark_D$ [Han & Amari, 1998] $\checkmark_C$
$(0_2, 0_2)$	[Han, 1987] $\checkmark$	[Han & Kobayashi, 1989] $\checkmark$
$(0_2, R_Y)$	[Han, 1987] $\checkmark$	[This work] $\checkmark$
$(0_3, 0_3)$	trivial	[This work] $\checkmark$

$(0_{M_X}, 0_{M_Y})$ — Constant Communication Bits

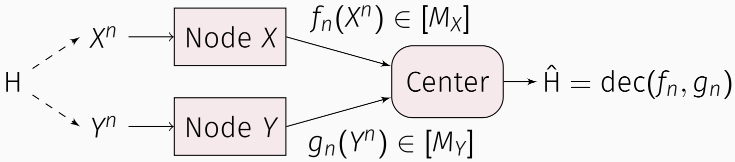


$(0_{M_X}, 0_{M_Y})$ — Constant Communication Bits



$[M] \triangleq \{0, 1, \ldots, M-1\}$: Message Set with M Symbols

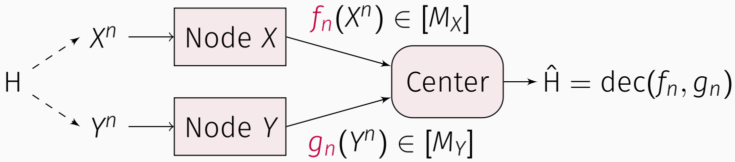
$(0_{M_X}, 0_{M_Y})$ — Constant Communication Bits



$[M] \triangleq \{0, 1, \ldots, M-1\}$: Message Set with M Symbols

Design of

$(0_{M_X}, 0_{M_Y})$ — Constant Communication Bits

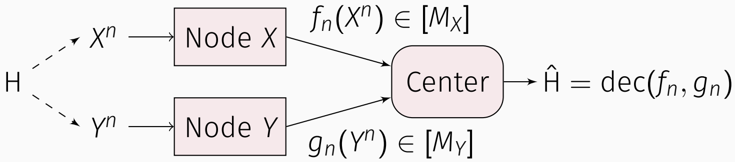


$[M] \triangleq \{0, 1, \ldots, M-1\}$: Message Set with M Symbols

Design of

- Encoders $f_n : \mathcal{X}^n \to [M_X], g_n : \mathcal{Y}^n \to [M_Y]$

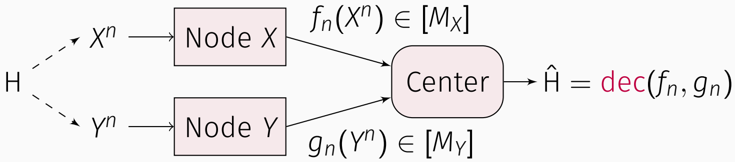
$(0_{M_X}, 0_{M_Y})$ — Constant Communication Bits



$[M] \triangleq \{0, 1, \ldots, M-1\}$: Message Set with M Symbols

Design of

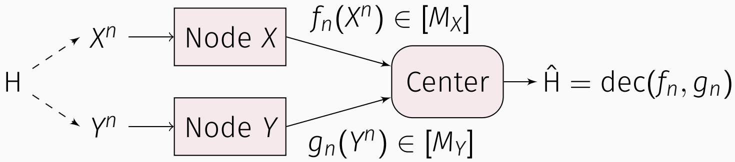
- Encoders $f_n : \mathcal{X}^n \to [M_X], g_n : \mathcal{Y}^n \to [M_Y]$
  - extracting "informative" bits from sequences



$[M] \triangleq \{0, 1, \ldots, M-1\}$: Message Set with M Symbols

Design of

- Encoders $f_n : \mathcal{X}^n \to [M_X], g_n : \mathcal{Y}^n \to [M_Y]$
  - extracting "informative" bits from sequences
- Decoder $\mathrm{dec}(\cdot, \cdot)$: Boolean table on $[M_X] \times [M_Y]$

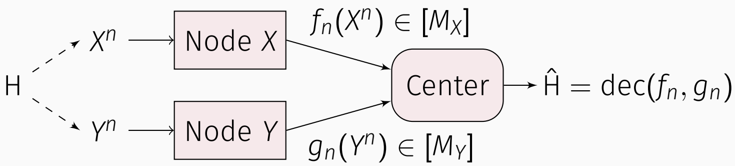


$[M] \triangleq \{0, 1, \ldots, M-1\}$: Message Set with M Symbols

Design of

- Encoders $f_n : \mathcal{X}^n \to [M_X], g_n : \mathcal{Y}^n \to [M_Y]$
  - extracting "informative" bits from sequences
- Decoder $\mathrm{dec}(\cdot, \cdot)$: Boolean table on $[M_X] \times [M_Y]$
  - optimal fusion rule

$(0_{M_X}, 0_{M_Y})$ — Constant Communication Bits



$[M] \triangleq \{0, 1, \ldots, M-1\}$: Message Set with M Symbols

Design of

- Encoders $f_n : \mathcal{X}^n \to [M_X], g_n : \mathcal{Y}^n \to [M_Y]$
  - extracting "informative" bits from sequences
- Decoder $\text{dec}(\cdot, \cdot)$: Boolean table on $[M_X] \times [M_Y]$
  - optimal fusion rule

Distributed Processing & **Communication Constraints**

Sequence $=$ Type $+$ "Order"

Encoder Design: Information Extraction from Sequences

Sequence $=$ Type $+$ "Order"

e.g., $x^n \in \mathcal{X}^n$

Sequence = Type + "Order"

e.g., $x^n \in \mathcal{X}^n$

- Type: empirical distribution $\hat{P}_{x^n}$

Sequence = Type + "Order"

e.g., $x^n \in \mathcal{X}^n$

- Type: empirical distribution $\hat{P}_{x^n}$
- Order: index of x^n in sequences of type $\hat{P}_{x^n}$

Sequence = Type + "Order"

e.g., $x^n \in \mathcal{X}^n$

- Type: empirical distribution $\hat{P}_{x^n}$
- Order: index of x^n in sequences of type $\hat{P}_{x^n}$

Example: binary sequences of length $n = 3$

Type $\leftrightarrow$ # of ones	0	1	2	3
Sequences	○○○	●○○ ○●○ ○○●	○●● ●○● ●●○	●●●

Sequence = Type + "Order"

e.g., $x^n \in \mathcal{X}^n$

- Type: empirical distribution $\hat{P}_{x^n}$
- Order: index of x^n in sequences of type $\hat{P}_{x^n}$

- \# of types = poly(n)
- \# of seqs = $|\mathcal{X}|^n = \exp(n \log |\mathcal{X}|)$

Example: binary sequences of length $n = 3$

Type $\leftrightarrow$ # of ones	0	1	2	3
Sequences	○○○	●○○ ○●○ ○○●	○●● ●○● ●●○	●●●

Encoder Design: Information Extraction from Sequences

Sequence = Type + "Order"

e.g., $x^n \in \mathcal{X}^n$

- Type: empirical distribution $\hat{P}_{x^n}$
- Order: index of x^n in sequences of type $\hat{P}_{x^n}$

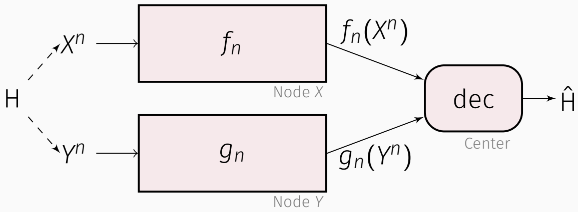
- # of types = poly(n)
- # of seqs = $|\mathcal{X}|^n = \exp(n \log |\mathcal{X}|)$

Example: binary sequences of length $n = 3$

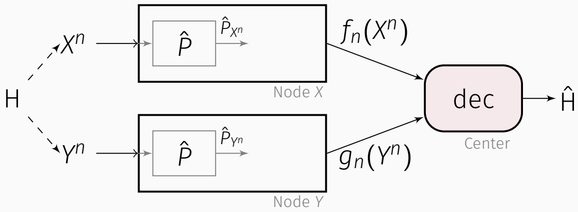
Type $\leftrightarrow$ # of ones	0	1	2	3
Sequences	○○○	●○○ ○●○ ○○●	○●● ●○● ●●○	●●●

Type Requires Much Less Bits to Describe

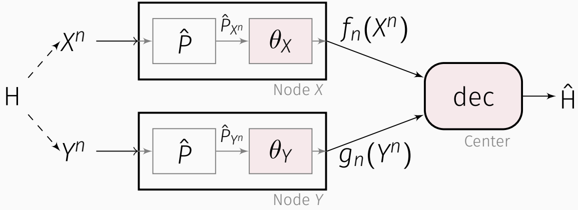
Type-based Encoders



Type-based Encoders



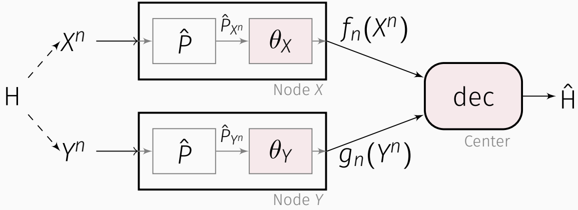
Type-based Encoders



Type-encoding Functions

- $\theta_X : \mathcal{P}^{\mathcal{X}} \to [M_X]$
- $\theta_Y : \mathcal{P}^{\mathcal{Y}} \to [M_Y]$

Type-based Encoders

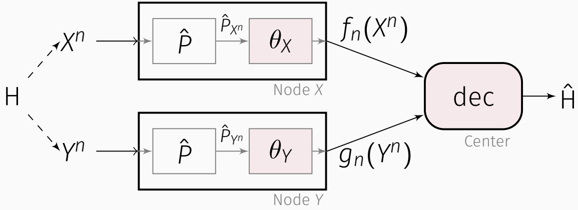


Type-encoding Functions

- $\theta_X : \mathcal{P}^{\mathcal{X}} \to [M_X]$
- $\theta_Y : \mathcal{P}^{\mathcal{Y}} \to [M_Y]$

Seq. Space $(\mathcal{X}^n, \mathcal{Y}^n)$ $\to$ Dist. Space $(\mathcal{P}^{\mathcal{X}}, \mathcal{P}^{\mathcal{Y}})$

Type-based Encoders

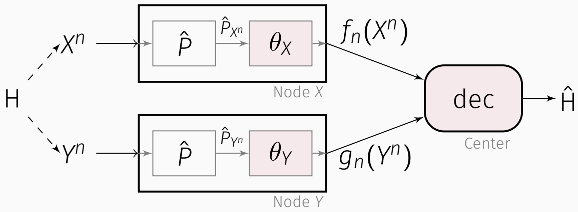


Type-encoding Functions

- $\theta_X : \mathcal{P}^{\mathcal{X}} \to [M_X]$
- $\theta_Y : \mathcal{P}^{\mathcal{Y}} \to [M_Y]$

Seq. Space $(\mathcal{X}^n, \mathcal{Y}^n)$ $\quad\to\quad$ Dist. Space $(\mathcal{P}^{\mathcal{X}}, \mathcal{P}^{\mathcal{Y}})$ (*Single-letter Solution*)

Type-based Encoders



Type-encoding Functions

- $\theta_X : \mathcal{P}^{\mathcal{X}} \to [M_X]$
- $\theta_Y : \mathcal{P}^{\mathcal{Y}} \to [M_Y]$

Seq. Space $(\mathcal{X}^n, \mathcal{Y}^n)$ $\quad\to\quad$ Dist. Space $(\mathcal{P}^{\mathcal{X}}, \mathcal{P}^{\mathcal{Y}})$ (*Single-letter Solution*)

Optimality?

Optimality of Type-based Encoders (for all decoders!)

Given zero-rate encoders f_n, g_n, and a decoder dec, we can construct type-based encoders $\tilde{f}_n, \tilde{g}_n$, such that

Optimality of Type-based Encoders (for all decoders!)

Given zero-rate encoders f_n, g_n, and a decoder dec, we can construct type-based encoders $\tilde{f}_n, \tilde{g}_n$, such that

- $\text{range}(\tilde{f}_n) = \text{range}(f_n)$
- $\text{range}(\tilde{g}_n) = \text{range}(g_n)$

Optimality of Type-based Encoders (for all decoders!)

Given zero-rate encoders f_n, g_n, and a decoder dec, we can construct type-based encoders $\tilde{f}_n, \tilde{g}_n$, such that

- $\mathrm{range}(\tilde{f}_n) = \mathrm{range}(f_n)$
- $\mathrm{range}(\tilde{g}_n) = \mathrm{range}(g_n)$
- $(\tilde{f}_n, \tilde{g}_n, \mathrm{dec})$ achieves better error exponents

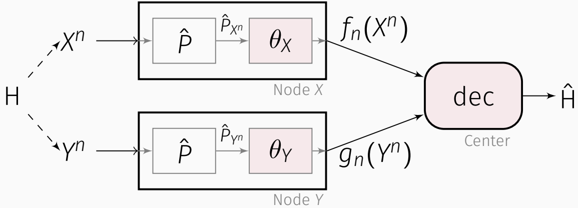
Optimality of Type-based Encoders (for all decoders!)

Given zero-rate encoders f_n, g_n, and a decoder dec, we can construct type-based encoders $\tilde{f}_n, \tilde{g}_n$, such that

- $\text{range}(\tilde{f}_n) = \text{range}(f_n)$
- $\text{range}(\tilde{g}_n) = \text{range}(g_n)$
- $(\tilde{f}_n, \tilde{g}_n, \text{dec})$ achieves better error exponents

Proof An application of the blowing up lemma.

Coding Scheme Design

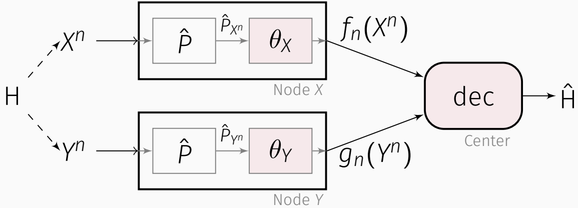


- Encoding

$$\mathcal{P}^{\mathcal{X}} \longrightarrow [M_X], \mathcal{P}^{\mathcal{Y}} \longrightarrow [M_Y]$$

- Decoding

$$[M_X] \times [M_Y] \longrightarrow \{0, 1\}$$



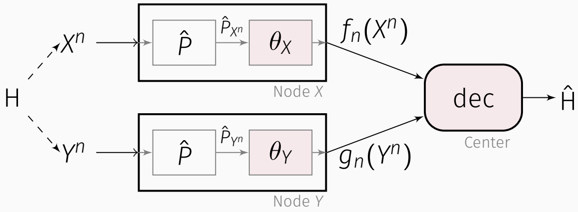
- Encoding

$$\mathcal{P}^{\mathcal{X}} \xrightarrow{\theta_X} [M_X], \mathcal{P}^{\mathcal{Y}} \xrightarrow{\theta_Y} [M_Y]$$

- Decoding

$$[M_X] \times [M_Y] \xrightarrow{\text{dec}} \{0, 1\}$$

Coding Scheme Design



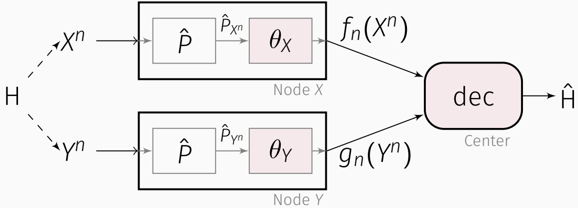
$\hat{H}$ depends only on marginal types $\hat{P}_{X^n}, \hat{P}_{Y^n}$

- Encoding

$$\mathcal{P}^{\mathcal{X}} \xrightarrow{\theta_X} [M_X], \mathcal{P}^{\mathcal{Y}} \xrightarrow{\theta_Y} [M_Y]$$

- Decoding

$$[M_X] \times [M_Y] \xrightarrow{\text{dec}} \{0, 1\}$$



- Encoding

$$\mathcal{P}^{\mathcal{X}} \xrightarrow{\theta_X} [M_X], \mathcal{P}^{\mathcal{Y}} \xrightarrow{\theta_Y} [M_Y]$$

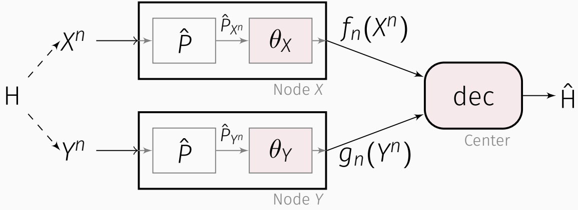
- Decoding

$$[M_X] \times [M_Y] \xrightarrow{\text{dec}} \{0, 1\}$$

$\hat{H}$ depends only on marginal types $\hat{P}_{X^n}, \hat{P}_{Y^n}$

- For sequences (x^n, y^n) with marginal types (Q_X, Q_Y):

$$\hat{H}(Q_X, Q_Y) = \text{dec}(\theta_X(Q_X), \theta_Y(Q_Y))$$



- Encoding

$$\mathcal{P}^{\mathcal{X}} \xrightarrow{\theta_X} [M_X], \mathcal{P}^{\mathcal{Y}} \xrightarrow{\theta_Y} [M_Y]$$

- Decoding

$$[M_X] \times [M_Y] \xrightarrow{\text{dec}} \{0, 1\}$$

Ĥ depends only on marginal types $\hat{P}_{X^n}, \hat{P}_{Y^n}$

- For sequences (x^n, y^n) with marginal types (Q_X, Q_Y):

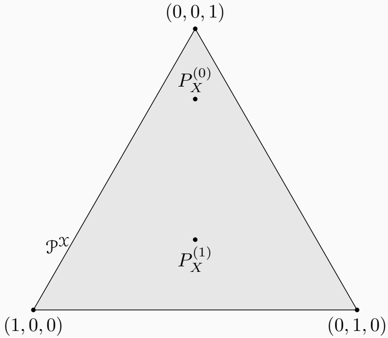
$$\hat{H}(Q_X, Q_Y) = \text{dec}(\theta_X(Q_X), \theta_Y(Q_Y))$$

How to design encoders (quantize types)?

One-bit Quantization of Types

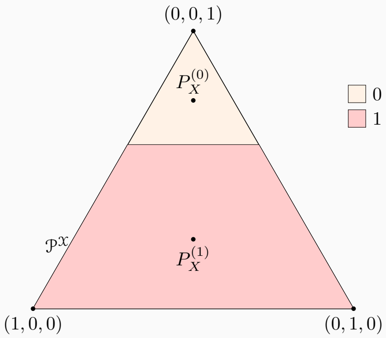
Hypothesis Testing on $\mathcal{X} = \{0, 1, 2\}, P_X^{(0)} = (1/8, 1/8, 3/4), P_X^{(1)} = (3/8, 3/8, 1/4)$

Hypothesis Testing on $\mathcal{X} = \{0, 1, 2\}$, $P_X^{(0)} = (1/8, 1/8, 3/4)$, $P_X^{(1)} = (3/8, 3/8, 1/4)$



Hypothesis Testing on $\mathcal{X} = \{0, 1, 2\}, P_X^{(0)} = (1/8, 1/8, 3/4), P_X^{(1)} = (3/8, 3/8, 1/4)$

Neyman-Pearson Test

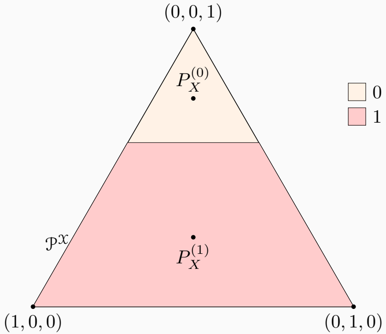


Quantize LLR: $\mathbb{E}_{\hat{P}_{X^n}} \left[\log \frac{P_X^{(0)}}{P_X^{(1)}} \right] \underset{\hat{\mathsf{H}}=1}{\overset{\hat{\mathsf{H}}=0}{\gtrless}} T$

One-bit Quantization of Types

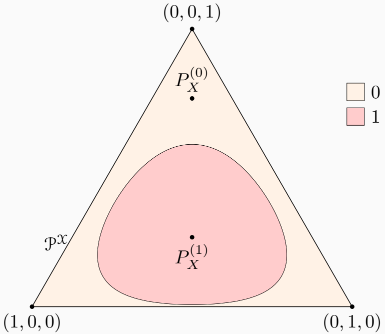
Hypothesis Testing on $\mathcal{X} = \{0, 1, 2\}$, $P_X^{(0)} = (1/8, 1/8, 3/4)$, $P_X^{(1)} = (3/8, 3/8, 1/4)$

Neyman-Pearson Test



Quantize LLR: $\mathbb{E}_{\hat{P}_{X^n}}\left[\log \frac{P_X^{(0)}}{P_X^{(1)}}\right] \underset{\hat{H}=1}{\overset{\hat{H}=0}{\gtrless}} T$

Hoeffding's Test



Quantize KL: $D(\hat{P}_{X^n} \| P_X^{(1)}) \underset{\hat{H}=1}{\overset{\hat{H}=0}{\gtrless}} T$

$$D_i^*(Q_X, Q_Y) \triangleq \min_{\substack{Q_{XY}:\ [Q_{XY}]_X = Q_X \\ [Q_{XY}]_Y = Q_Y}} D(Q_{XY} \| P_{XY}^{(i)})$$

Sequences with marginal types (Q_X, Q_Y)

- Associated probability:

$$\mathbb{P}\left\{(\hat{P}_{X^n}, \hat{P}_{Y^n}) = (Q_X, Q_Y) \middle| \mathrm{H} = i\right\} \doteq \exp\left(-n \cdot D_i^*(Q_X, Q_Y)\right)$$

$$D_i^*(Q_X, Q_Y) \triangleq \min_{\substack{Q_{XY}:\ [Q_{XY}]_X = Q_X \\ [Q_{XY}]_Y = Q_Y}} D(Q_{XY} \| P_{XY}^{(i)})$$

Sequences with marginal types (Q_X, Q_Y)

- Associated probability:

$$\mathbb{P}\left\{(\hat{P}_{X^n}, \hat{P}_{Y^n}) = (Q_X, Q_Y) \middle| \mathsf{H} = i\right\} \doteq \exp\left(-n \cdot D_i^*(Q_X, Q_Y)\right)$$

- if $\hat{\mathsf{H}}(Q_X, Q_Y) \neq i$, E_i will be at least $D_i^*(Q_X, Q_Y)$

$$D_i^*(Q_X, Q_Y) \triangleq \min_{\substack{Q_{XY}:\ [Q_{XY}]_X = Q_X \\ [Q_{XY}]_Y = Q_Y}} D(Q_{XY} \| P_{XY}^{(i)})$$

Sequences with marginal types (Q_X, Q_Y)

- Associated probability:

$$\mathbb{P}\left\{(\hat{P}_{X^n}, \hat{P}_{Y^n}) = (Q_X, Q_Y) \middle| \mathrm{H} = i\right\} \doteq \exp\left(-n \cdot D_i^*(Q_X, Q_Y)\right)$$

- if $\hat{\mathrm{H}}(Q_X, Q_Y) \neq i$, E_i will be at least $D_i^*(Q_X, Q_Y)$

Divergence ball $\mathcal{D}_i(t) \triangleq \{(Q_X, Q_Y) \colon D_i^*(Q_X, Q_Y) < t\}$

$$D_i^*(Q_X, Q_Y) \triangleq \min_{\substack{Q_{XY}:\ [Q_{XY}]_X = Q_X \\ [Q_{XY}]_Y = Q_Y}} D(Q_{XY} \| P_{XY}^{(i)})$$

Sequences with marginal types (Q_X, Q_Y)

- Associated probability:

$$\mathbb{P}\left\{(\hat{P}_{X^n}, \hat{P}_{Y^n}) = (Q_X, Q_Y) \middle| \mathsf{H} = i\right\} \doteq \exp\left(-n \cdot D_i^*(Q_X, Q_Y)\right)$$

- if $\hat{\mathsf{H}}(Q_X, Q_Y) \neq i$, E_i will be at least $D_i^*(Q_X, Q_Y)$

Divergence ball $\mathcal{D}_i(t) \triangleq \{(Q_X, Q_Y) \colon D_i^*(Q_X, Q_Y) < t\}$

- Center: $(P_X^{(i)}, P_Y^{(i)})$

$$D_i^*(Q_X, Q_Y) \triangleq \min_{\substack{Q_{XY}\colon [Q_{XY}]_X = Q_X \\ [Q_{XY}]_Y = Q_Y}} D(Q_{XY} \| P_{XY}^{(i)})$$

Sequences with marginal types (Q_X, Q_Y)

- Associated probability:

$$\mathbb{P}\left\{(\hat{P}_{X^n}, \hat{P}_{Y^n}) = (Q_X, Q_Y) \middle| \mathsf{H} = i\right\} \doteq \exp\left(-n \cdot D_i^*(Q_X, Q_Y)\right)$$

- if $\hat{\mathsf{H}}(Q_X, Q_Y) \neq i$, E_i will be at least $D_i^*(Q_X, Q_Y)$

Divergence ball $\mathcal{D}_i(t) \triangleq \{(Q_X, Q_Y)\colon D_i^*(Q_X, Q_Y) < t\}$

- Center: $(P_X^{(i)}, P_Y^{(i)})$
- “Radius”: t

$$D_i^*(Q_X, Q_Y) \triangleq \min_{\substack{Q_{XY}:\, [Q_{XY}]_X = Q_X \\ [Q_{XY}]_Y = Q_Y}} D(Q_{XY} \| P_{XY}^{(i)})$$

Sequences with marginal types (Q_X, Q_Y)

- Associated probability:

$$\mathbb{P}\left\{(\hat{P}_{X^n}, \hat{P}_{Y^n}) = (Q_X, Q_Y) \middle| \mathsf{H} = i\right\} \doteq \exp\left(-n \cdot D_i^*(Q_X, Q_Y)\right)$$

- if $\hat{\mathsf{H}}(Q_X, Q_Y) \neq i$, E_i will be at least $D_i^*(Q_X, Q_Y)$

Divergence ball $\mathcal{D}_i(t) \triangleq \{(Q_X, Q_Y) \colon D_i^*(Q_X, Q_Y) < t\}$

- Center: $(P_X^{(i)}, P_Y^{(i)})$
- “Radius”: t

(E_0, E_1) **is achievable, iff** $\hat{\mathsf{H}}(\mathcal{D}_0(E_0)) = 0, \hat{\mathsf{H}}(\mathcal{D}_1(E_1)) = 1$

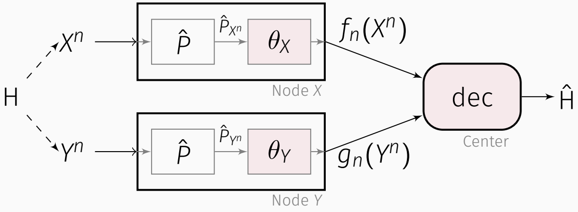
$$\hat{\mathrm{H}}(Q_X, Q_Y) = \mathrm{dec}(\theta_X(Q_X), \theta_Y(Q_Y))$$

- Encoding

$$\mathcal{P}^{\mathcal{X}} \xrightarrow{\theta_X} [M_X], \mathcal{P}^{\mathcal{Y}} \xrightarrow{\theta_Y} [M_Y]$$

- Decoding

$$[M_X] \times [M_Y] \xrightarrow{\mathrm{dec}} \{0, 1\}$$



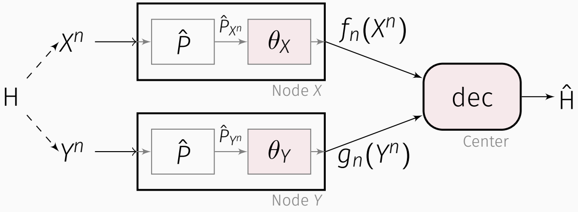
$$\hat{H}(Q_X, Q_Y) = \mathrm{dec}(\theta_X(Q_X), \theta_Y(Q_Y))$$

- Encoding

$$\mathcal{P}^{\mathcal{X}} \xrightarrow{\theta_X} [M_X], \mathcal{P}^{\mathcal{Y}} \xrightarrow{\theta_Y} [M_Y]$$

- Decoding

$$[M_X] \times [M_Y] \xrightarrow{\mathrm{dec}} \{0, 1\}$$



Goal: choose θ_X, θ_Y and dec, such that

- all points in $\mathcal{D}_0(E_0)$ are decoded as 0
- all points in $\mathcal{D}_1(E_1)$ are decoded as 1

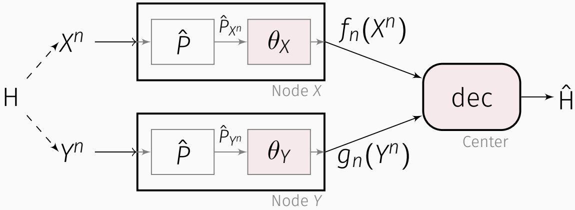
$$\hat{\mathrm{H}}(Q_X, Q_Y) = \mathrm{dec}(\theta_X(Q_X), \theta_Y(Q_Y))$$

- Encoding

$$\mathcal{P}^{\mathcal{X}} \xrightarrow{\theta_X} [M_X], \mathcal{P}^{\mathcal{Y}} \xrightarrow{\theta_Y} [M_Y]$$

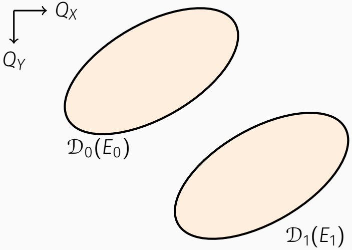
- Decoding

$$[M_X] \times [M_Y] \xrightarrow{\mathrm{dec}} \{0, 1\}$$

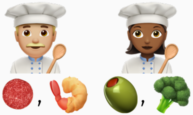


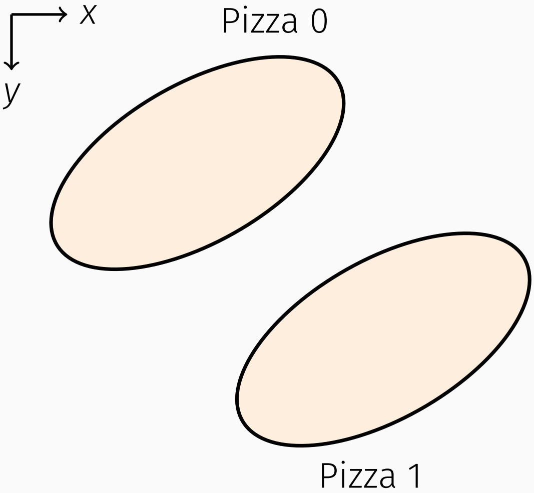
Goal: choose θ_X, θ_Y and dec, such that

- all points in $\mathcal{D}_0(E_0)$ are decoded as 0
- all points in $\mathcal{D}_1(E_1)$ are decoded as 1

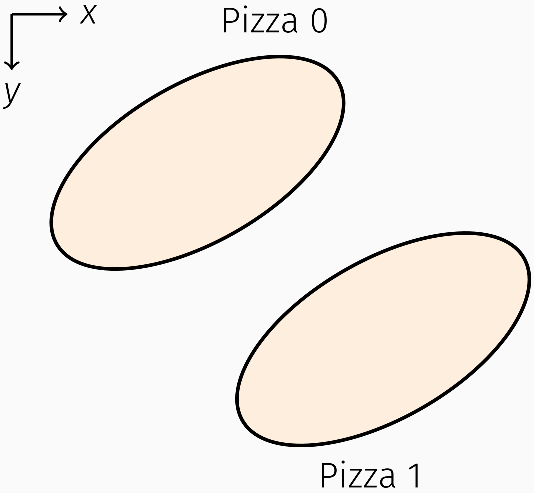


On Distributed Pizza Topping with Finite Ingredients




x
y
Pizza 0
Pizza 1

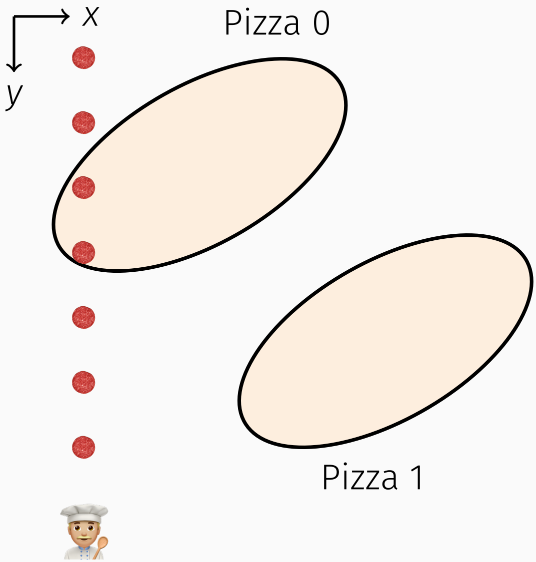
Distributed Pizza Topping



Cook X 👨‍🍳: top based on x

- 

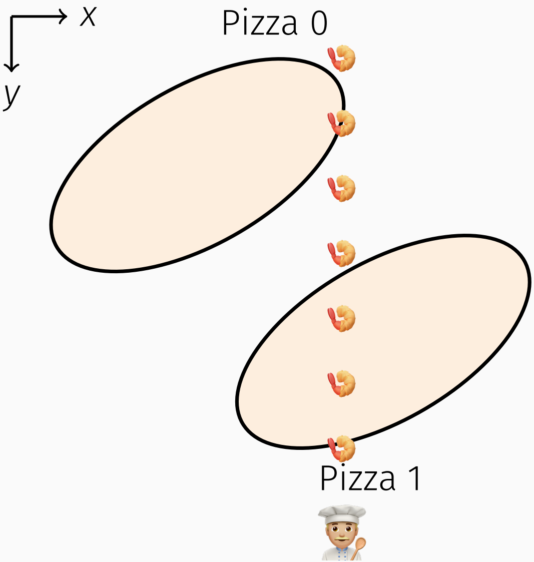
Distributed Pizza Topping



Cook X: top based on x

- ,

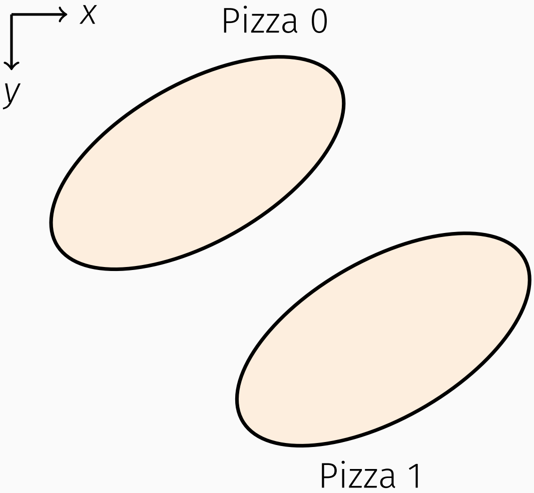
Distributed Pizza Topping



Cook X 🧑‍🍳: top based on x

- 🍕, 🍤

Distributed Pizza Topping

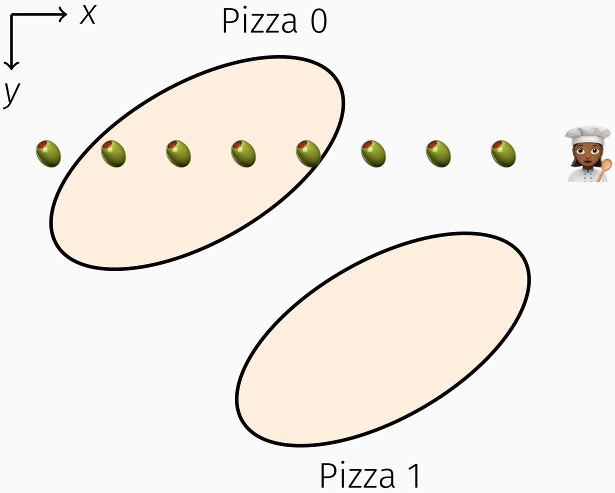


Cook X 👨🏼‍🍳: top based on x

- 🔴, 🦐

Cook Y 👩🏾‍🍳: top based on y

- 🫒, 🥦



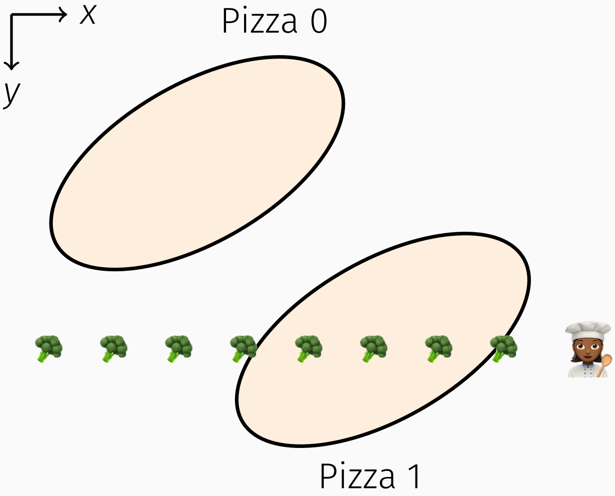
Cook X 👨🏼‍🍳: top based on x

- , 🍤

Cook Y 👩🏾‍🍳: top based on y

- 🫒, 🥦

Distributed Pizza Topping

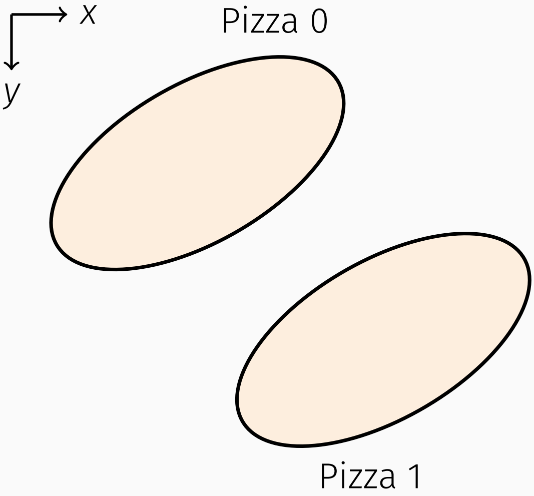


Cook X 🧑‍🍳: top based on x

- 🍕, 🍤

Cook Y 👩‍🍳: top based on y

- 🫒, 🥦



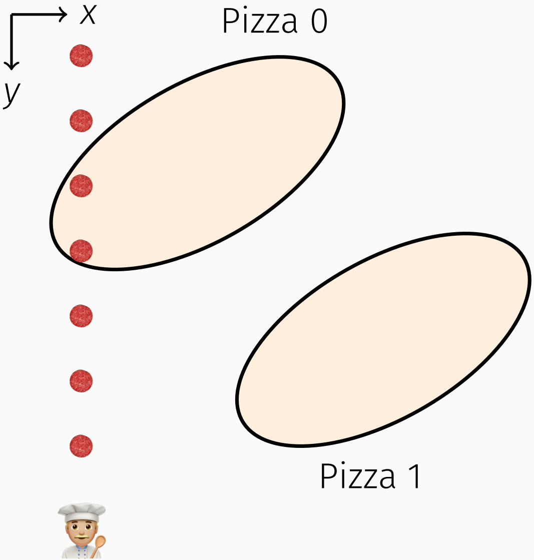
Cook X 👨🏼‍🍳: top based on x

- , 🦐

Cook Y 👩🏾‍🍳: top based on y

- 🫒, 🥦

Goal: distinguishable pizzas



Cook X 👨🏼‍🍳: top based on x

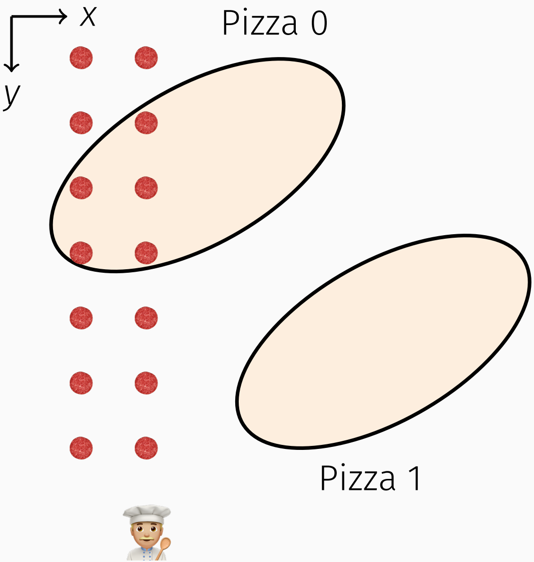
- 🔴, 🍤

Cook Y 👩🏾‍🍳: top based on y

- 🫒, 🥦

Goal: distinguishable pizzas

Distributed Pizza Topping



Cook X: top based on x

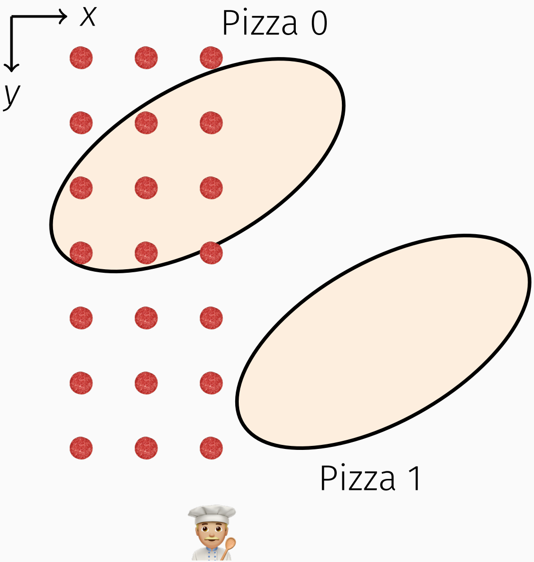
- 🍅, 🍤

Cook Y: top based on y

- 🫒, 🥦

Goal: distinguishable pizzas

Distributed Pizza Topping



Cook X 👨‍🍳: top based on x

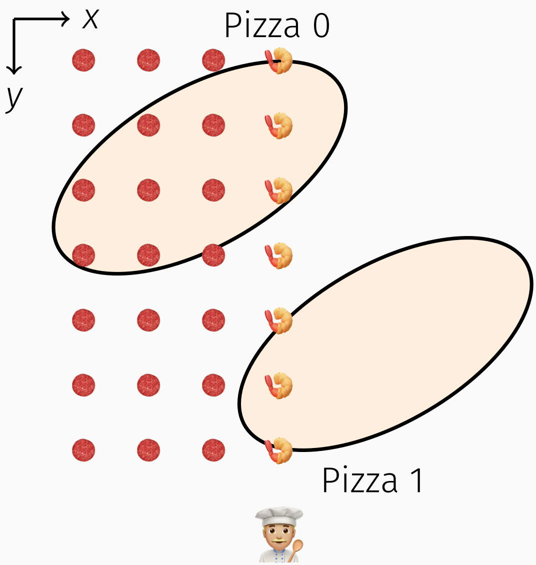
- 🍕, 🍤

Cook Y 👩‍🍳: top based on y

- 🫒, 🥦

Goal: distinguishable pizzas

Distributed Pizza Topping



Cook X 🧑‍🍳: top based on x

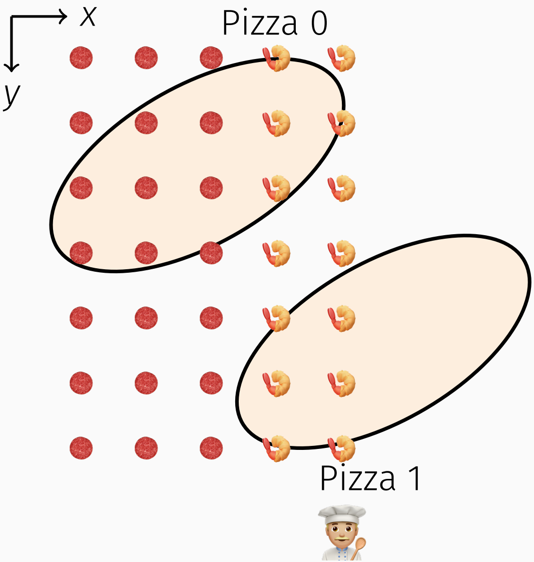
- 🍕, 🍤

Cook Y 👩‍🍳: top based on y

- 🫒, 🥦

Goal: distinguishable pizzas

Distributed Pizza Topping



Cook X: top based on x

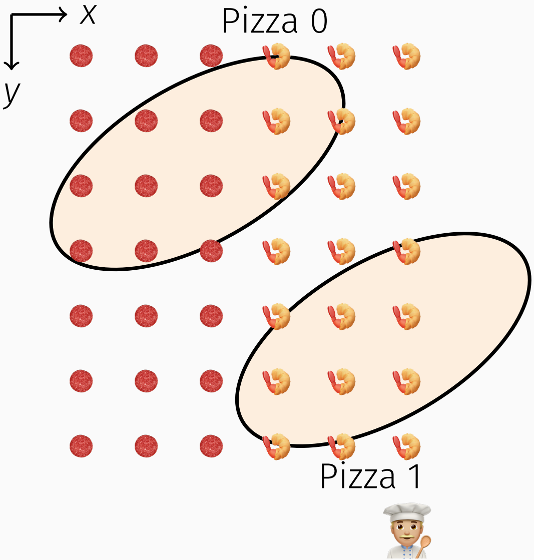
- ,

Cook Y: top based on y

- ,

Goal: distinguishable pizzas

Distributed Pizza Topping



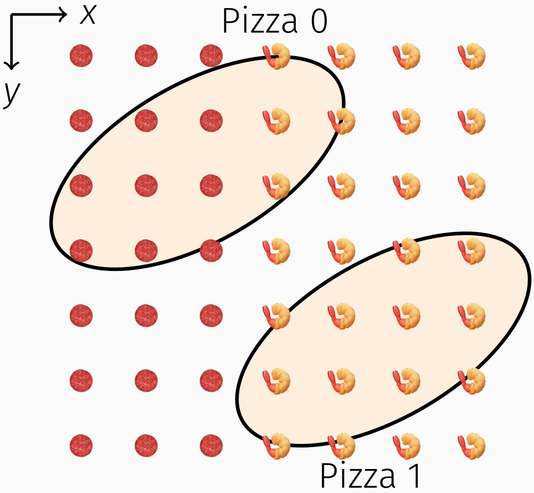
Cook X: top based on x

- ,

Cook Y: top based on y

- ,

Goal: distinguishable pizzas



Cook X 👨‍🍳: top based on x

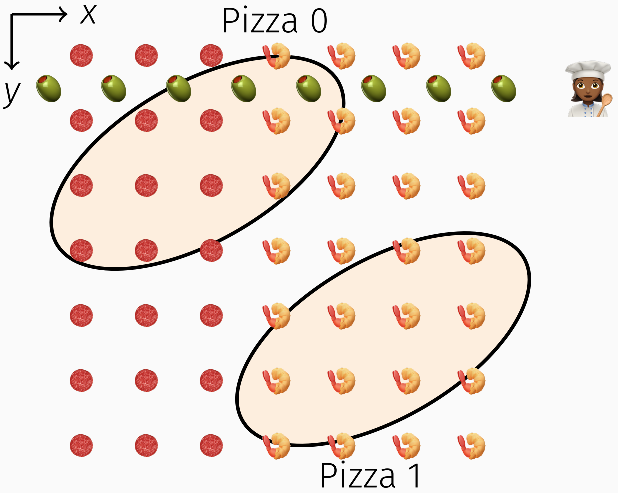
- 🍅, 🦐

Cook Y 👩🏾‍🍳: top based on y

- 🫒, 🥦

Goal: distinguishable pizzas

Distributed Pizza Topping



Cook X 🧑‍🍳: top based on *x*

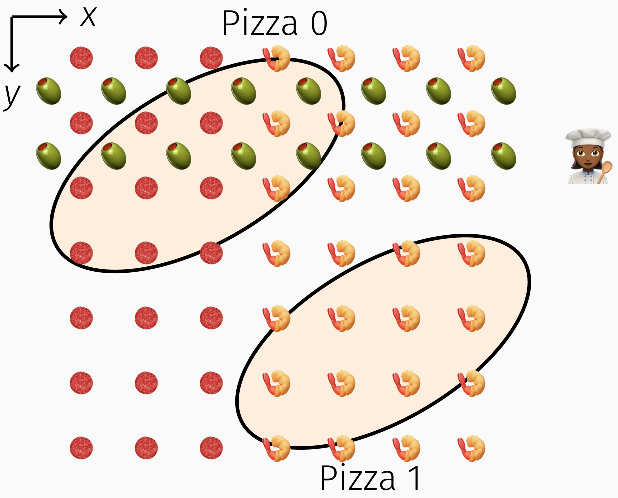
- 🍕, 🍤

Cook Y 👩‍🍳: top based on *y*

- 🫒, 🥦

Goal: distinguishable pizzas

Distributed Pizza Topping



Cook X: top based on x

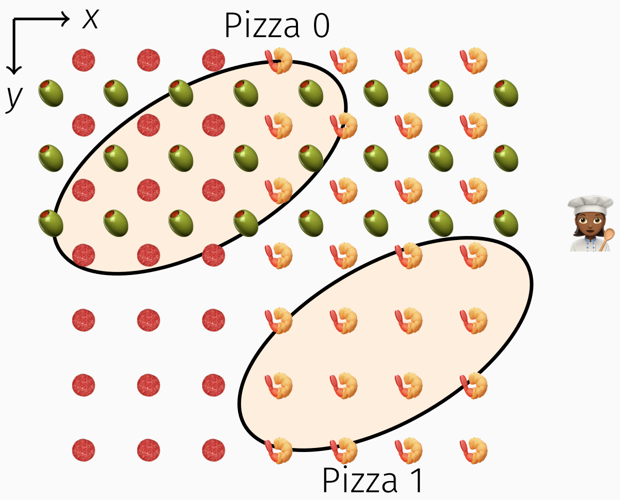
- 🍖, 🍤

Cook Y: top based on y

- 🫒, 🥦

Goal: distinguishable pizzas

Distributed Pizza Topping



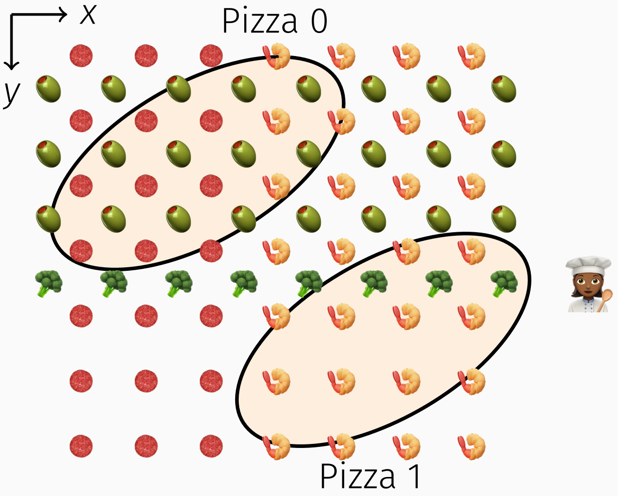
Cook X 🧑‍🍳: top based on x

- 🍕, 🍤

Cook Y 👩‍🍳: top based on y

- 🫒, 🥦

Goal: distinguishable pizzas



Cook X 👨‍🍳: top based on x

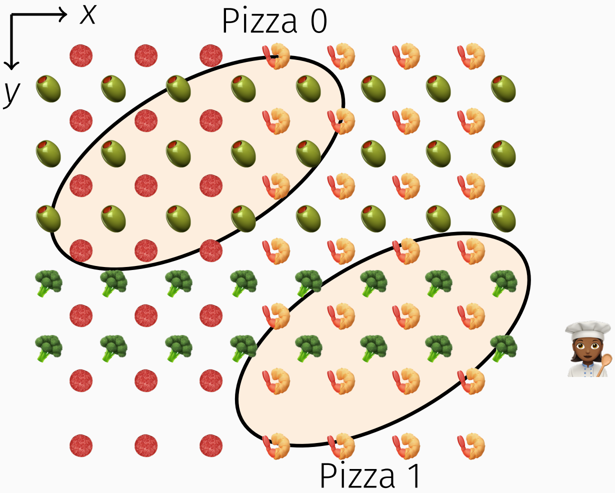
- 🔴, 🍤

Cook Y 👩‍🍳: top based on y

- 🫒, 🥦

Goal: distinguishable pizzas

Distributed Pizza Topping



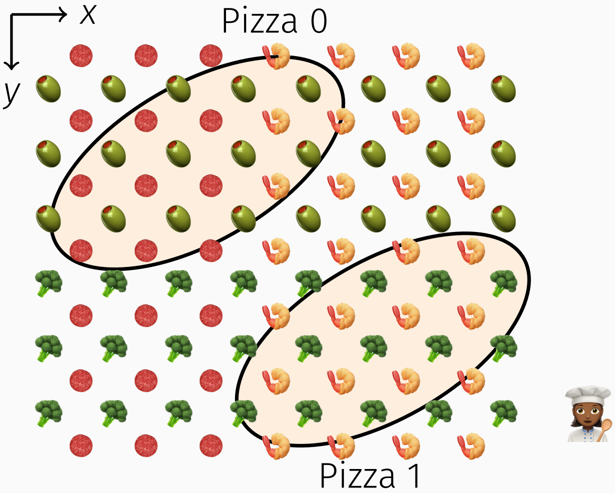
Cook X 👨‍🍳: top based on *x*

- 🔴, 🦐

Cook Y 👩🏾‍🍳: top based on *y*

- 🫒, 🥦

Goal: distinguishable pizzas



Cook X 👨‍🍳: top based on x

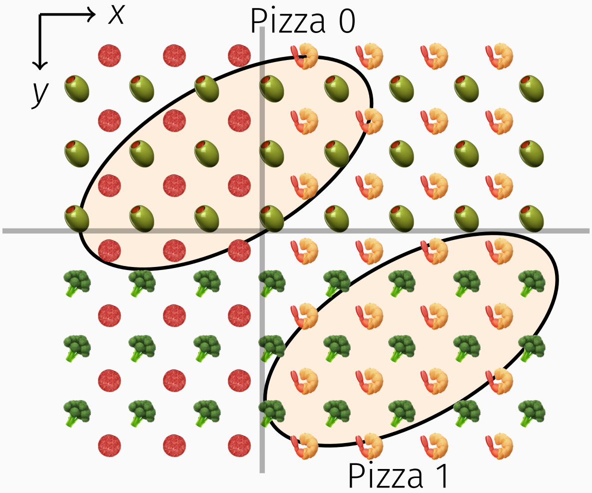
- 🔴, 🍤

Cook Y 👩‍🍳: top based on y

- 🫒, 🥦

Goal: distinguishable pizzas

Distributed Pizza Topping



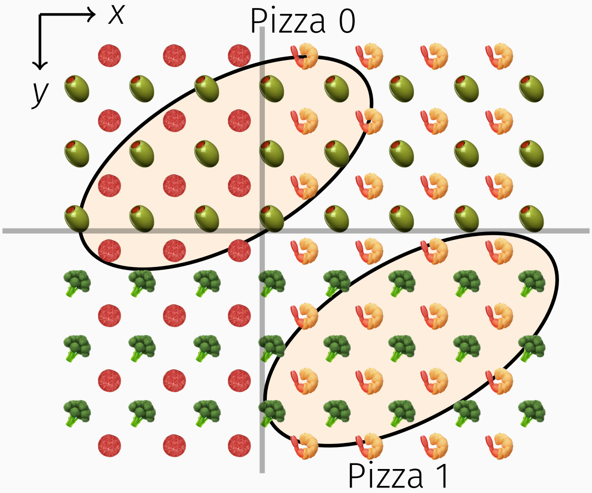
Cook X 👨‍🍳: top based on x

- 🍤

Cook Y 👩🏾‍🍳: top based on y

- 🫒, 🥦

Goal: distinguishable pizzas



Cook X 👨‍🍳: top based on x

- 🔴, 🦐

Cook Y 👩🏾‍🍳: top based on y

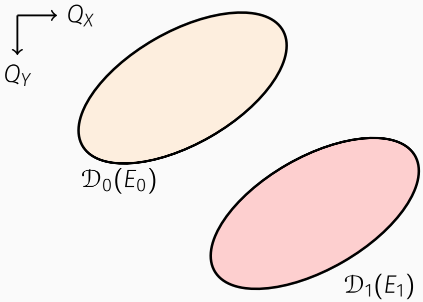
- 🫒, 🥦

Goal: distinguishable pizzas

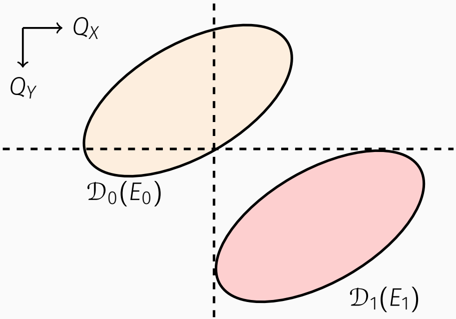
Ingredient Table

	🔴	🦐
🫒	0	0
🥦	0	1

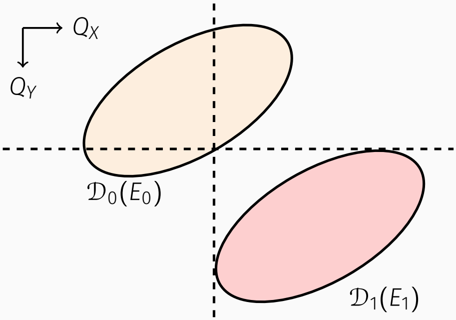
One-bit Compression $M_X = M_Y = 2$



One-bit Compression $M_X = M_Y = 2$

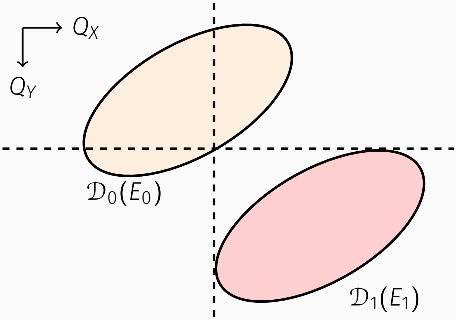


One-bit Compression $M_X = M_Y = 2$



dec	0	1
0	0	0
1	0	1

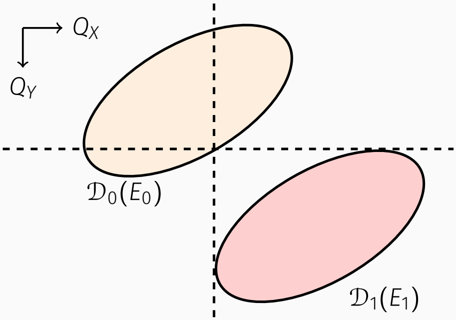
One-bit Compression $M_X = M_Y = 2$



dec	0	1
0	0	0
1	0	1

$\leftrightarrow$

One-bit Compression $M_X = M_Y = 2$

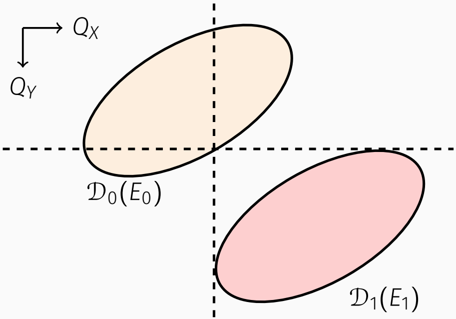


dec	0	1
0	0	0
1	0	1

$\leftrightarrow$

Limiting Case: $M_X = M_Y = \infty$

One-bit Compression $M_X = M_Y = 2$



dec	0	1
0	0	0
1	0	1

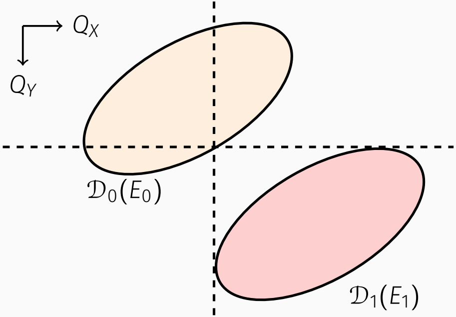
$\leftrightarrow$

Limiting Case: $M_X = M_Y = \infty$

(E_0, E_1) is achievable iff

$$\mathcal{D}_0(E_0) \cap \mathcal{D}_1(E_1) = \varnothing$$

One-bit Compression $M_X = M_Y = 2$



dec	0	1
0	0	0
1	0	1

$\leftrightarrow$

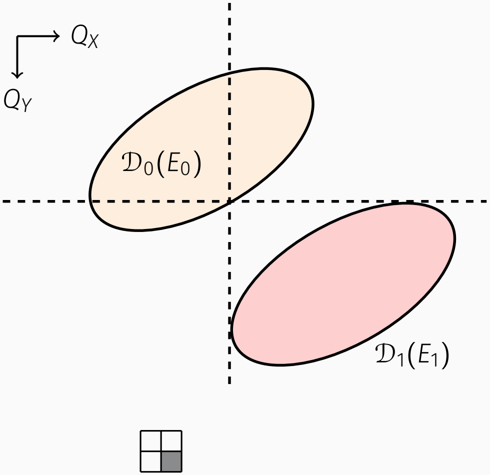
Limiting Case: $M_X = M_Y = \infty$

(E_0, E_1) is achievable iff

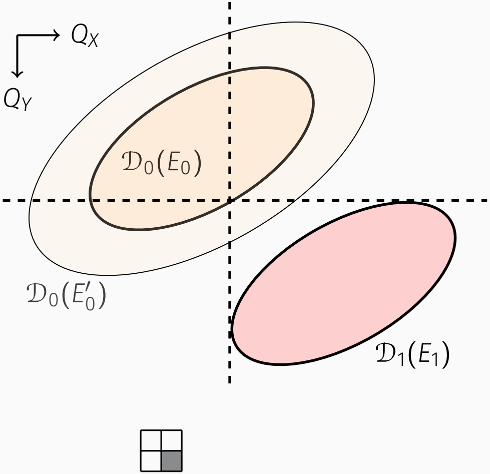
$$\mathcal{D}_0(E_0) \cap \mathcal{D}_1(E_1) = \varnothing$$

zero-rate compression $(0, 0)$

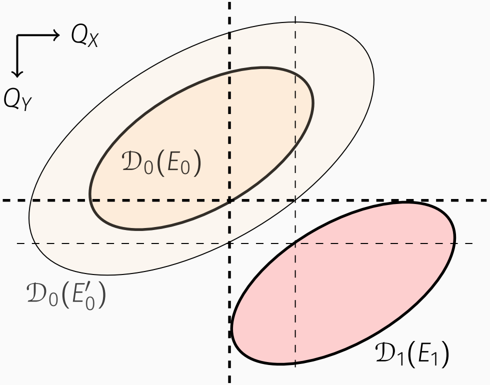
Add more symbols: $(0_2, 0_2) \to (0_3, 0_3)$

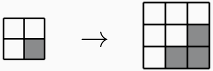


Add more symbols: $(0_2, 0_2) \to (0_3, 0_3)$

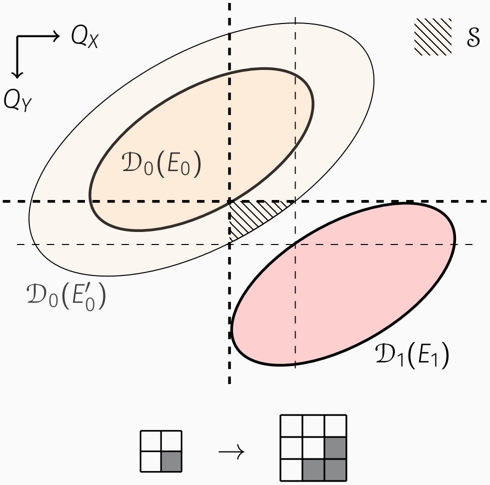


Add more symbols: $(\mathcal{O}_2, \mathcal{O}_2) \to (\mathcal{O}_3, \mathcal{O}_3)$



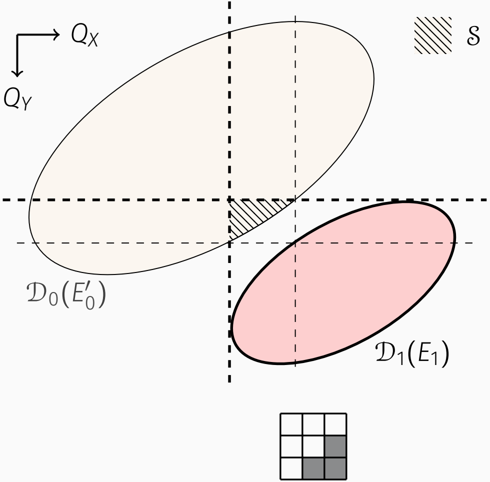


Add more symbols: $(0_2, 0_2) \to (0_3, 0_3)$

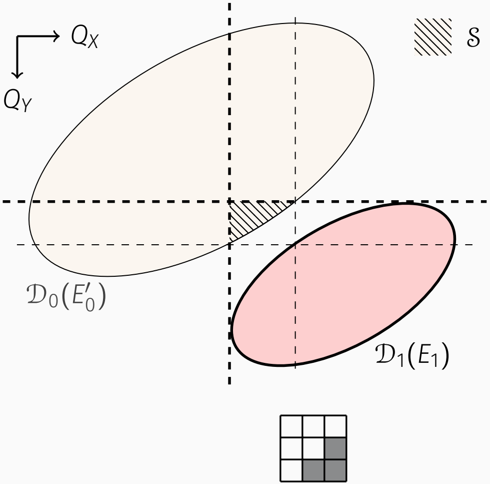


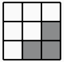
Use added symbols to "describe" $\mathcal{S}$

Add more symbols: $(0_2, 0_2) \to (0_3, 0_3)$



Add more symbols: $(0_2, 0_2) \to (0_3, 0_3)$

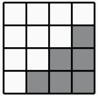


SEPARATE $\mathcal{D}_0(E_0')$ and $\mathcal{D}_1(E_1)$ by 

$$\Updownarrow$$

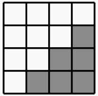
SEPARATE $\mathcal{S}$ and $\mathcal{D}_1(E_1)$ by 

$$\mathrm{dec}(m_X, m_Y) = \mathbb{1}\{m_X + m_Y \geq M_X \wedge M_Y\}$$



$\mathrm{dec}(m_X, m_Y) = \mathbb{1}\{m_X + m_Y \geq M_X \wedge M_Y\}$

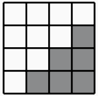
- m_X, m_Y: discrete-valued beliefs



Threshold Decoders

$$\mathrm{dec}(m_X, m_Y) = \mathbb{1}\{m_X + m_Y \geq M_X \wedge M_Y\}$$

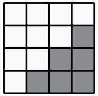
- m_X, m_Y: discrete-valued beliefs
- $m_X + m_Y$: fused belief



Threshold Decoders

$$\mathrm{dec}(m_X, m_Y) = \mathbb{1}\{m_X + m_Y \geq M_X \wedge M_Y\}$$

- m_X, m_Y: discrete-valued beliefs
- $m_X + m_Y$: fused belief
- $M_X \wedge M_Y = \min\{M_X, M_Y\}$: threshold

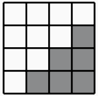


Threshold Decoders

$$\mathrm{dec}(m_X, m_Y) = \mathbb{1}\{m_X + m_Y \geq M_X \wedge M_Y\}$$

- m_X, m_Y: discrete-valued beliefs
- $m_X + m_Y$: fused belief
- $M_X \wedge M_Y = \min\{M_X, M_Y\}$: threshold

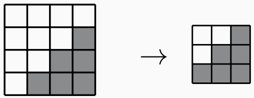
“Reduction”: Recursive Croppings



$\mathrm{dec}(m_X, m_Y) = \mathbb{1}\{m_X + m_Y \geq M_X \wedge M_Y\}$

- m_X, m_Y: discrete-valued beliefs
- $m_X + m_Y$: fused belief
- $M_X \wedge M_Y = \min\{M_X, M_Y\}$: threshold

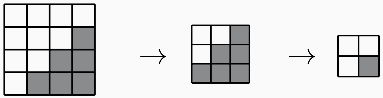
"Reduction": Recursive Croppings



$$\mathrm{dec}(m_X, m_Y) = \mathbb{1}\{m_X + m_Y \geq M_X \wedge M_Y\}$$

- m_X, m_Y: discrete-valued beliefs
- $m_X + m_Y$: fused belief
- $M_X \wedge M_Y = \min\{M_X, M_Y\}$: threshold

"Reduction": Recursive Croppings

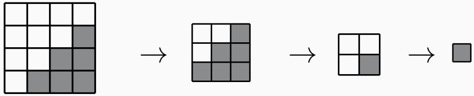


$\text{dec}(m_X, m_Y) = \mathbb{1}\{m_X + m_Y \geq M_X \wedge M_Y\}$

- m_X, m_Y: discrete-valued beliefs
- $m_X + m_Y$: fused belief
- $M_X \wedge M_Y = \min\{M_X, M_Y\}$: threshold

Optimality?

“Reduction”: Recursive Croppings

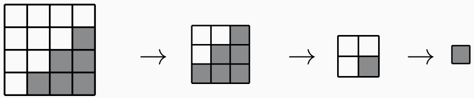


Threshold Decoders

$$\text{dec}(m_X, m_Y) = \mathbb{1}\{m_X + m_Y \geq M_X \wedge M_Y\}$$

- m_X, m_Y: discrete-valued beliefs
- $m_X + m_Y$: fused belief
- $M_X \wedge M_Y = \min\{M_X, M_Y\}$: threshold

"Reduction": Recursive Croppings



Threshold DECs are Optimal if

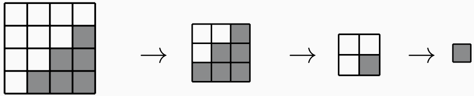
$$\mathrm{dec}(m_X, m_Y) = \mathbb{1}\{m_X + m_Y \geq M_X \wedge M_Y\}$$

- m_X, m_Y: discrete-valued beliefs
- $m_X + m_Y$: fused belief
- $M_X \wedge M_Y = \min\{M_X, M_Y\}$: threshold

Threshold DECs are Optimal if

$$\min\{M_X, M_Y\} = 2$$

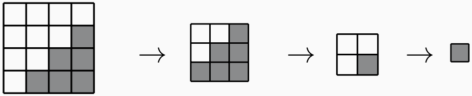
"Reduction": Recursive Croppings



$\mathrm{dec}(m_X, m_Y) = \mathbb{1}\{m_X + m_Y \geq M_X \wedge M_Y\}$

- m_X, m_Y: discrete-valued beliefs
- $m_X + m_Y$: fused belief
- $M_X \wedge M_Y = \min\{M_X, M_Y\}$: threshold

“Reduction”: Recursive Croppings



Threshold DECs are Optimal if

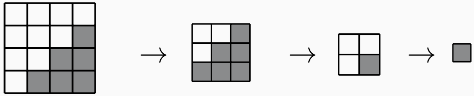
$$\min\{M_X, M_Y\} = 2$$

or $(M_X, M_Y) = (3, 3)$

$\mathrm{dec}(m_X, m_Y) = \mathbb{1}\{m_X + m_Y \geq M_X \wedge M_Y\}$

- m_X, m_Y: discrete-valued beliefs
- $m_X + m_Y$: fused belief
- $M_X \wedge M_Y = \min\{M_X, M_Y\}$: threshold

“Reduction”: Recursive Croppings



Threshold DECs are Optimal if

$\min\{M_X, M_Y\} = 2$

or $(M_X, M_Y) = (3, 3)$

or $P_{XY}^{(i)} = P_X^{(i)} P_Y^{(i)}$ for some $i \in \{0, 1\}$

$\mathrm{dec}(m_X, m_Y) = \mathbb{1}\{m_X + m_Y \geq M_X \wedge M_Y\}$

- m_X, m_Y: discrete-valued beliefs
- $m_X + m_Y$: fused belief
- $M_X \wedge M_Y = \min\{M_X, M_Y\}$: threshold

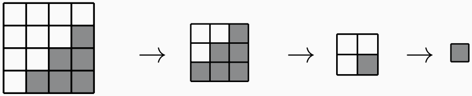
Threshold DECs are Optimal if

$\min\{M_X, M_Y\} = 2$

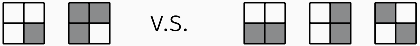
or $(M_X, M_Y) = (3, 3)$

or $P_{XY}^{(i)} = P_X^{(i)} P_Y^{(i)}$ for some $i \in \{0, 1\}$

“Reduction”: Recursive Croppings



Example: $M_X = M_Y = 2$



$\mathrm{dec}(m_X, m_Y) = \mathbb{1}\{m_X + m_Y \geq M_X \wedge M_Y\}$

- m_X, m_Y: discrete-valued beliefs
- $m_X + m_Y$: fused belief
- $M_X \wedge M_Y = \min\{M_X, M_Y\}$: threshold

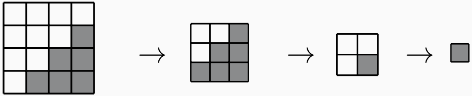
Threshold DECs are Optimal if

$\min\{M_X, M_Y\} = 2$

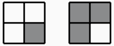
or $(M_X, M_Y) = (3, 3)$

or $P^{(i)}_{XY} = P^{(i)}_X P^{(i)}_Y$ for some $i \in \{0, 1\}$

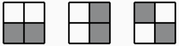
"Reduction": Recursive Croppings



Example: $M_X = M_Y = 2$



v.s.



What if $M_X = M_Y = 4$?

Threshold Decoders

$\mathrm{dec}(m_X, m_Y) = \mathbb{1}\{m_X + m_Y \geq M_X \wedge M_Y\}$

- m_X, m_Y: discrete-valued beliefs
- $m_X + m_Y$: fused belief
- $M_X \wedge M_Y = \min\{M_X, M_Y\}$: threshold

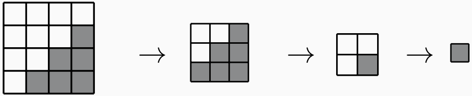
Threshold DECs are Optimal if

$$\min\{M_X, M_Y\} = 2$$

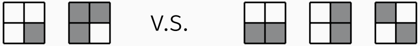
or $(M_X, M_Y) = (3, 3)$

or $P_{XY}^{(i)} = P_X^{(i)} P_Y^{(i)}$ for some $i \in \{0, 1\}$

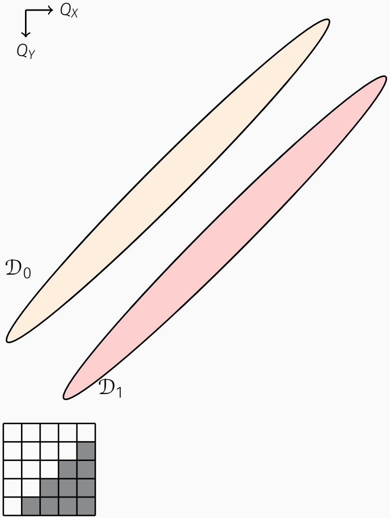
"Reduction": Recursive Croppings



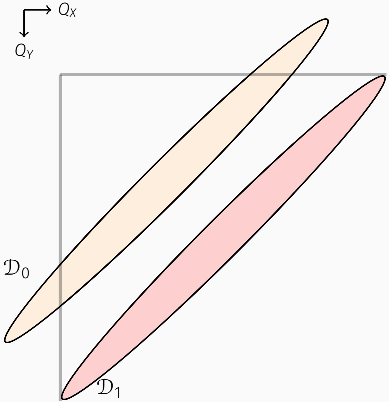
Example: $M_X = M_Y = 2$

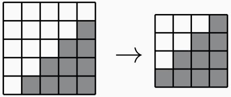


What if $M_X = M_Y = 4$?

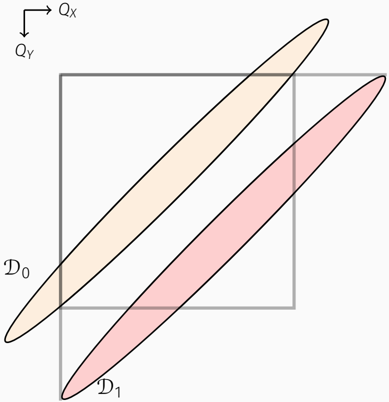

Q_X
Q_Y
$\mathcal{D}_0$
$\mathcal{D}_1$

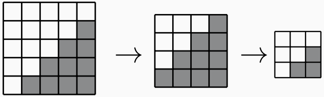
Baguettes 🥖🥖



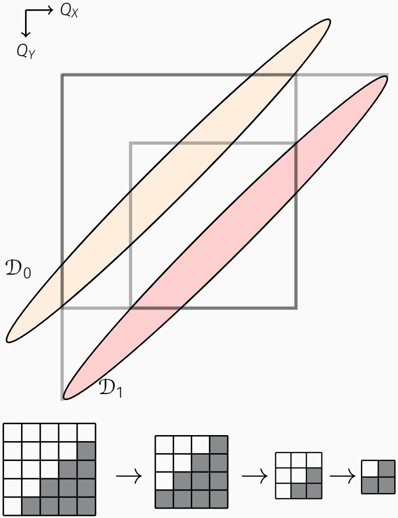


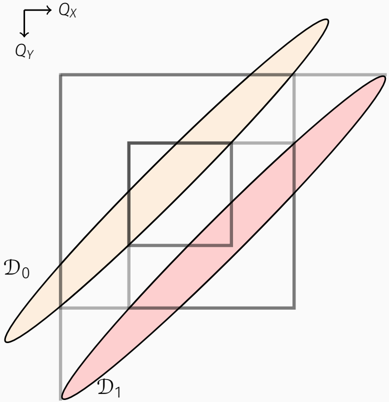
Baguettes 🥖🥖

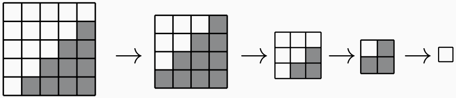




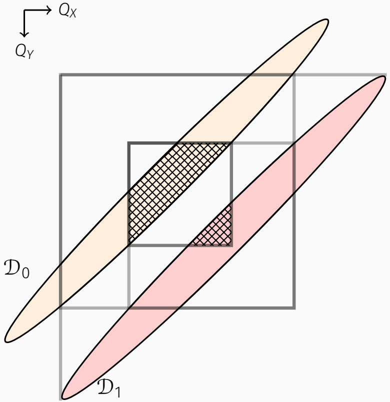
Baguettes 🥖🥖

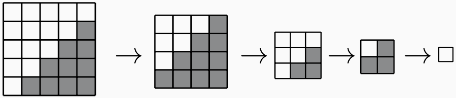



Q_X
Q_Y
$\mathcal{D}_0$
$\mathcal{D}_1$

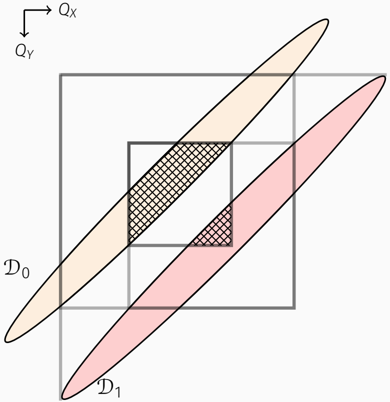


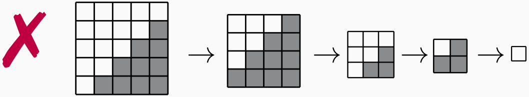
Baguettes 🥖🥖





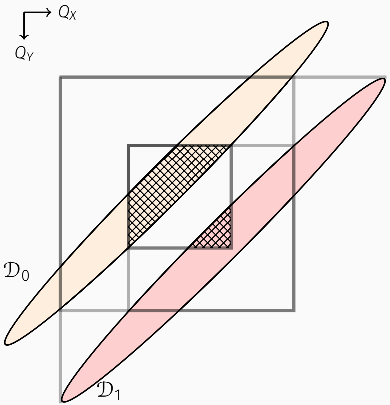
BAGUETTES 🥖🥖

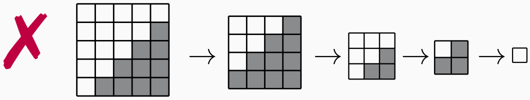




Threshold DEC requires $\geq$ 6 Symbols

Baguettes 🥖🥖

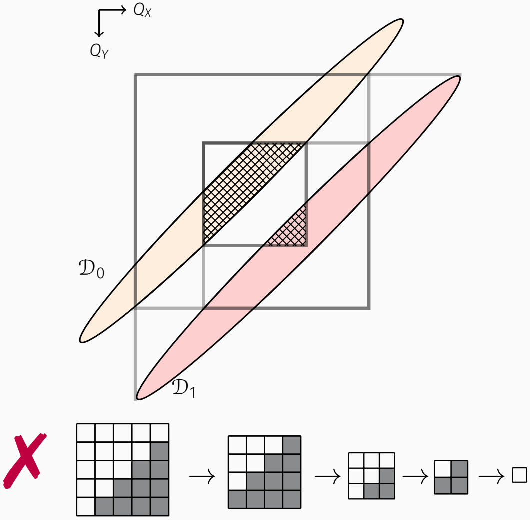


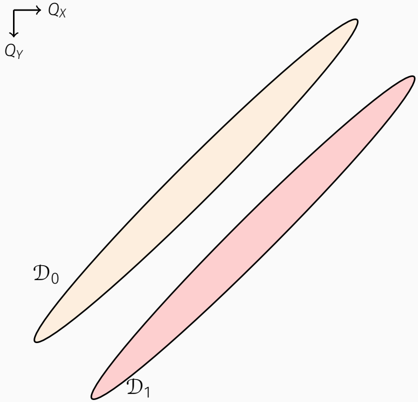


Threshold DEC requires $\geq$ 6 Symbols

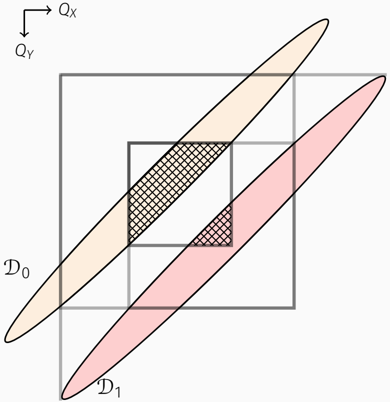
Can we do better?

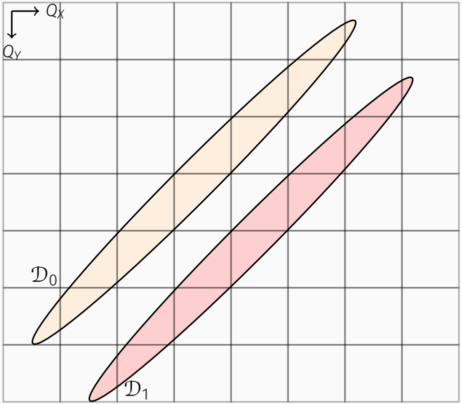
Baguettes 🥖🥖

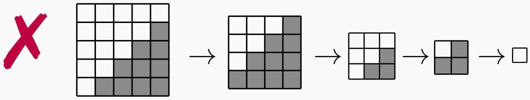




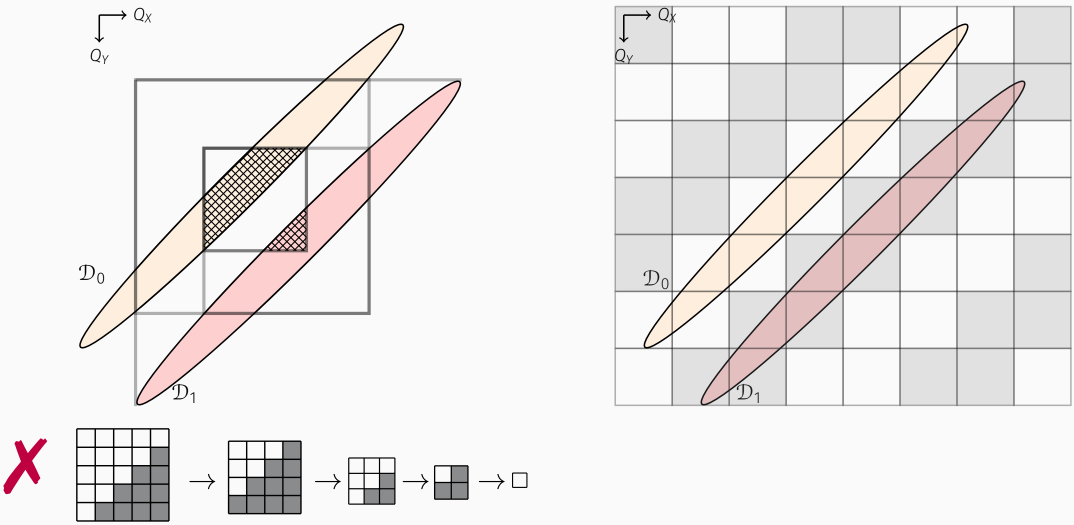
Baguettes 🥖🥖



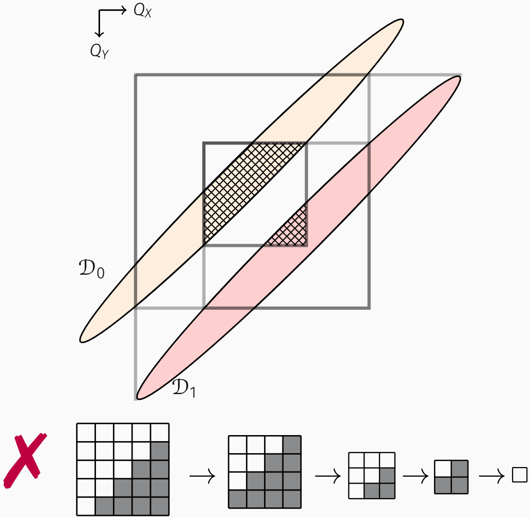




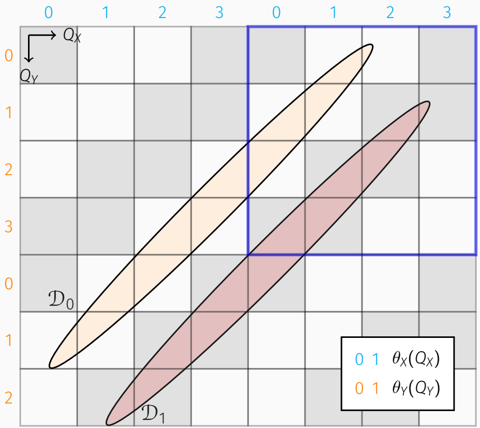
Baguettes 🥖🥖



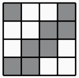
Baguettes 🥖🥖



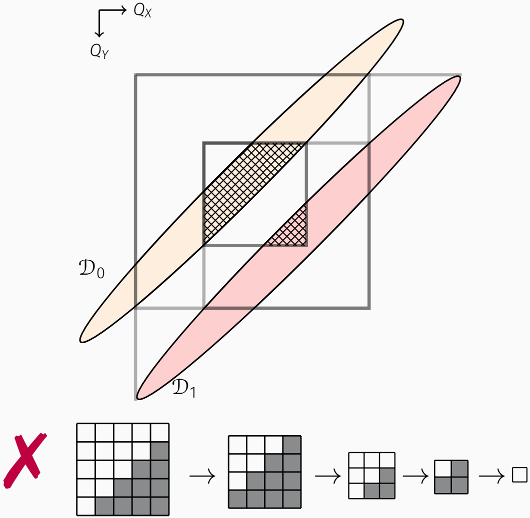
Q_X
Q_Y
$\mathcal{D}_0$
$\mathcal{D}_1$



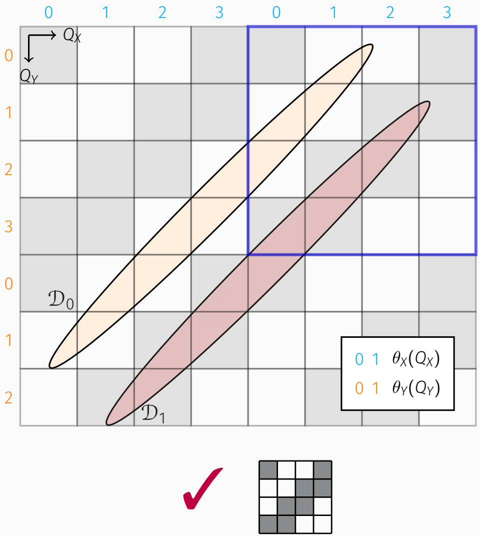
0 1 2 3 0 1 2 3
0 1 2 3 0 1 2
Q_X
Q_Y
$\mathcal{D}_0$
$\mathcal{D}_1$
0 1 $\theta_X(Q_X)$
0 1 $\theta_Y(Q_Y)$



Baguettes 🥖🥖



Q_X
Q_Y
$\mathcal{D}_0$
$\mathcal{D}_1$
✗
→ → → →



0 1 2 3 0 1 2 3
0 1 2 3 0 1 2
Q_X
Q_Y
$\mathcal{D}_0$
$\mathcal{D}_1$
0 1 $\theta_X(Q_X)$
0 1 $\theta_Y(Q_Y)$
✓

Numerical Example I

$$\mathcal{X} = \mathcal{Y} = \{0, 1, 2\}$$

$$P_X^{(0)} = P_Y^{(0)} = (1/8, 1/8, 3/4)$$

$$P_X^{(1)} = P_Y^{(1)} = (3/8, 3/8, 1/4)$$

$$P_{XY}^{(i)} = P_X^{(i)} P_Y^{(i)}, i \in \{0, 1\}$$

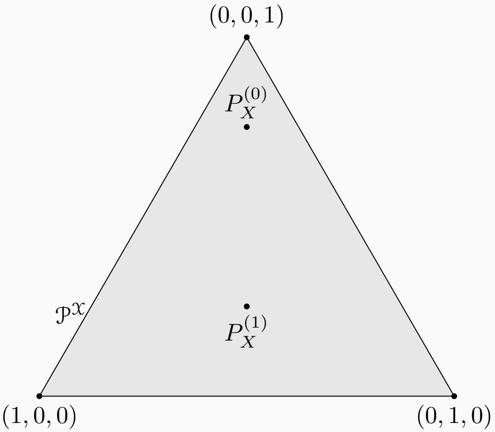
Numerical Example I

$$\mathcal{X} = \mathcal{Y} = \{0, 1, 2\}$$

$$P_X^{(0)} = P_Y^{(0)} = (1/8, 1/8, 3/4)$$

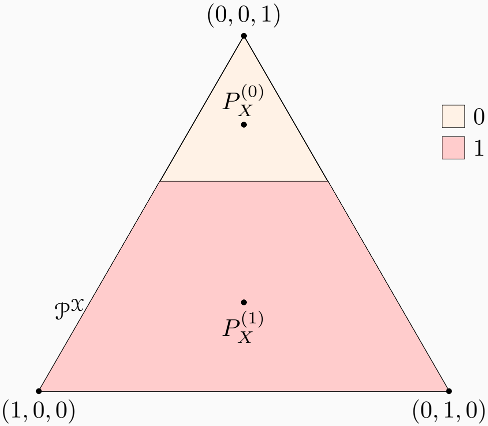
$$P_X^{(1)} = P_Y^{(1)} = (3/8, 3/8, 1/4)$$

$$P_{XY}^{(i)} = P_X^{(i)} P_Y^{(i)}, i \in \{0, 1\}$$



$$\mathcal{X} = \mathcal{Y} = \{0, 1, 2\}$$
$$P_X^{(0)} = P_Y^{(0)} = (1/8, 1/8, 3/4)$$
$$P_X^{(1)} = P_Y^{(1)} = (3/8, 3/8, 1/4)$$
$$P_{XY}^{(i)} = P_X^{(i)} P_Y^{(i)}, i \in \{0, 1\}$$

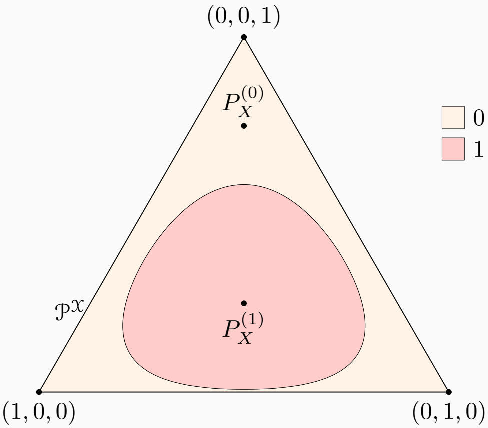
Local Decision at X: Neyman-Pearson



Numerical Example I

$$\mathcal{X} = \mathcal{Y} = \{0, 1, 2\}$$
$$P_X^{(0)} = P_Y^{(0)} = (1/8, 1/8, 3/4)$$
$$P_X^{(1)} = P_Y^{(1)} = (3/8, 3/8, 1/4)$$
$$P_{XY}^{(i)} = P_X^{(i)} P_Y^{(i)}, i \in \{0, 1\}$$

Local Decision at X: Hoeffding (Generalized Likelihood Ratio Test)



Numerical Example I

$$\mathcal{X} = \mathcal{Y} = \{0, 1, 2\}$$

$$P_X^{(0)} = P_Y^{(0)} = (1/8, 1/8, 3/4)$$

$$P_X^{(1)} = P_Y^{(1)} = (3/8, 3/8, 1/4)$$

$$P_{XY}^{(i)} = P_X^{(i)} P_Y^{(i)}, i \in \{0, 1\}$$

- Comm. Constraints: $(0_7, 0_7)$

Numerical Example I

$$\mathcal{X} = \mathcal{Y} = \{0, 1, 2\}$$
$$P_X^{(0)} = P_Y^{(0)} = (1/8, 1/8, 3/4)$$
$$P_X^{(1)} = P_Y^{(1)} = (3/8, 3/8, 1/4)$$
$$P_{XY}^{(i)} = P_X^{(i)} P_Y^{(i)}, i \in \{0, 1\}$$

- Comm. Constraints: $(0_7, 0_7)$
- dec: $(m_X, m_Y) \mapsto \mathbb{1}\{m_X + m_Y \geq 7\}$

Numerical Example I

$$\mathcal{X} = \mathcal{Y} = \{0, 1, 2\}$$
$$P_X^{(0)} = P_Y^{(0)} = (1/8, 1/8, 3/4)$$
$$P_X^{(1)} = P_Y^{(1)} = (3/8, 3/8, 1/4)$$
$$P_{XY}^{(i)} = P_X^{(i)} P_Y^{(i)}, i \in \{0, 1\}$$

- Comm. Constraints: $(0_7, 0_7)$
- dec: $(m_X, m_Y) \mapsto \mathbb{1}\{m_X + m_Y \geq 7\}$
- $(E_0, E_1) = (0.3, 0.25)$

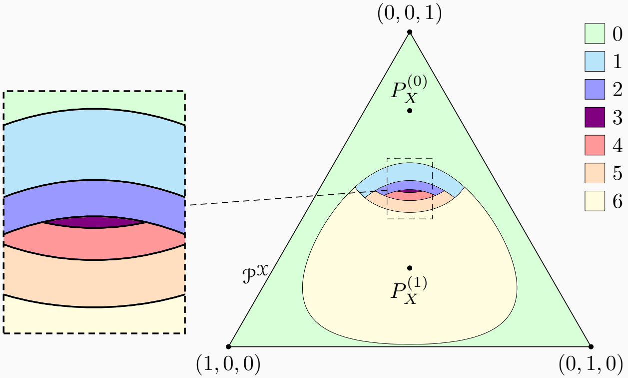
Numerical Example I

$$\mathcal{X} = \mathcal{Y} = \{0, 1, 2\}$$
$$P_X^{(0)} = P_Y^{(0)} = (1/8, 1/8, 3/4)$$
$$P_X^{(1)} = P_Y^{(1)} = (3/8, 3/8, 1/4)$$
$$P_{XY}^{(i)} = P_X^{(i)} P_Y^{(i)}, i \in \{0, 1\}$$

- Comm. Constraints: $(0_7, 0_7)$
- dec: $(m_X, m_Y) \mapsto \mathbb{1}\{m_X + m_Y \geq 7\}$
- $(E_0, E_1) = (0.3, 0.25)$

DHT: Type-encoding Function

$$\theta_X \colon \mathcal{P}^{\mathcal{X}} \to \{0, 1, \ldots, 6\}$$

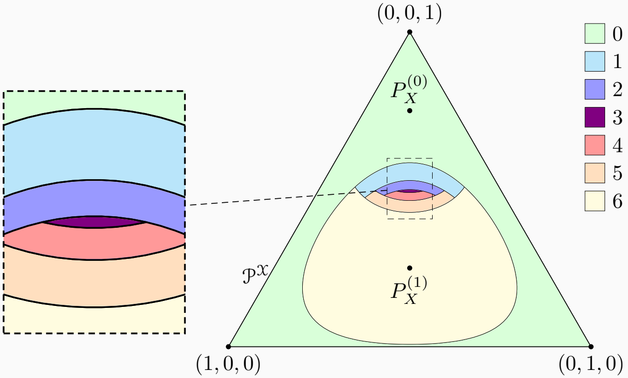


$$\mathcal{X} = \mathcal{Y} = \{0, 1, 2\}$$
$$P_X^{(0)} = P_Y^{(0)} = (1/8, 1/8, 3/4)$$
$$P_X^{(1)} = P_Y^{(1)} = (3/8, 3/8, 1/4)$$
$$P_{XY}^{(i)} = P_X^{(i)} P_Y^{(i)}, i \in \{0, 1\}$$

- Comm. Constraints: $(0_7, 0_7)$
- dec: $(m_X, m_Y) \mapsto \mathbb{1}\{m_X + m_Y \geq 7\}$
- $(E_0, E_1) = (0.3, 0.25)$

DHT: Type-encoding Function

$$\theta_X : \mathcal{P}^{\mathcal{X}} \to \{0, 1, \ldots, 6\}$$



Quantization of LLR is not optimal!

Numerical Example II: Error Exponents

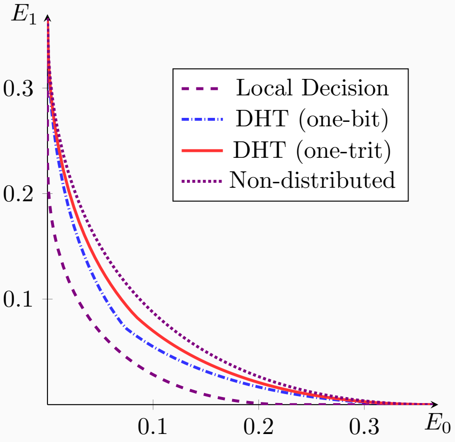
$\mathcal{X} = \mathcal{Y} = \{0, 1\}$

$$P_{XY}^{(0)} = \begin{bmatrix} 1/6 & 1/6 \\ 1/6 & 1/2 \end{bmatrix}, P_{XY}^{(1)} = \begin{bmatrix} 1/2 & 1/6 \\ 1/6 & 1/6 \end{bmatrix}$$

Numerical Example II: Error Exponents

$$\mathcal{X} = \mathcal{Y} = \{0, 1\}$$

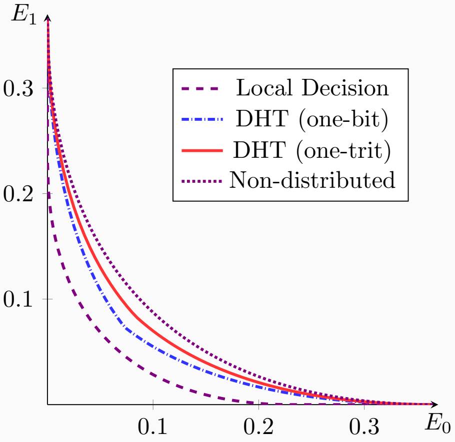
$$P_{XY}^{(0)} = \begin{bmatrix} 1/6 & 1/6 \\ 1/6 & 1/2 \end{bmatrix}, P_{XY}^{(1)} = \begin{bmatrix} 1/2 & 1/6 \\ 1/6 & 1/6 \end{bmatrix}$$



$\mathcal{X} = \mathcal{Y} = \{0, 1\}$

$$P_{XY}^{(0)} = \begin{bmatrix} 1/6 & 1/6 \\ 1/6 & 1/2 \end{bmatrix}, P_{XY}^{(1)} = \begin{bmatrix} 1/2 & 1/6 \\ 1/6 & 1/6 \end{bmatrix}$$

- Local Decision: $(0_1, \log 2)$ or $(\log 2, 0_1)$

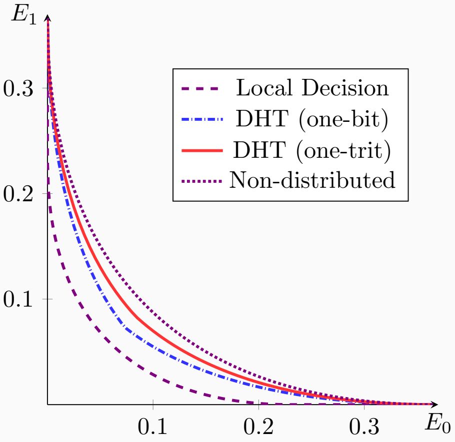


Numerical Example II: Error Exponents

$\mathcal{X} = \mathcal{Y} = \{0, 1\}$

$$P_{XY}^{(0)} = \begin{bmatrix} 1/6 & 1/6 \\ 1/6 & 1/2 \end{bmatrix}, P_{XY}^{(1)} = \begin{bmatrix} 1/2 & 1/6 \\ 1/6 & 1/6 \end{bmatrix}$$

- Local Decision: $(0_1, \log 2)$ or $(\log 2, 0_1)$
- One-bit: $(0_2, 0_2)$

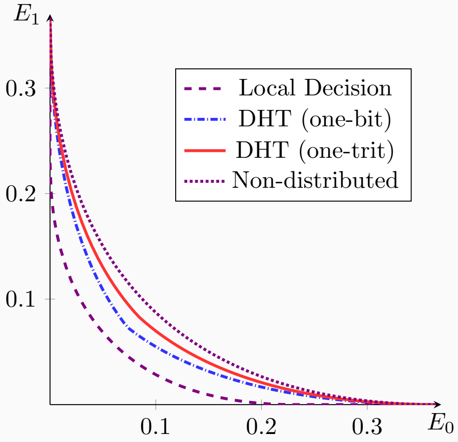


Numerical Example II: Error Exponents

$\mathcal{X} = \mathcal{Y} = \{0, 1\}$

$$P_{XY}^{(0)} = \begin{bmatrix} 1/6 & 1/6 \\ 1/6 & 1/2 \end{bmatrix}, P_{XY}^{(1)} = \begin{bmatrix} 1/2 & 1/6 \\ 1/6 & 1/6 \end{bmatrix}$$

- Local Decision: $(0_1, \log 2)$ or $(\log 2, 0_1)$
- One-bit: $(0_2, 0_2)$
- One-trit: $(0_3, 0_3)$

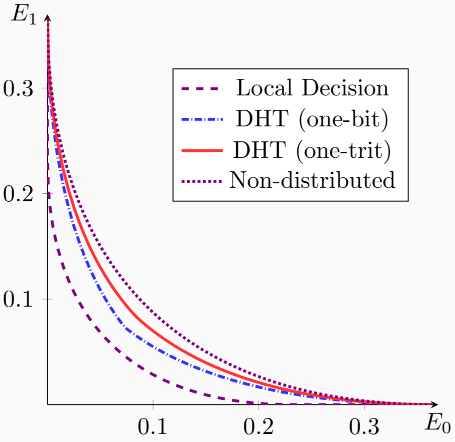


Numerical Example II: Error Exponents

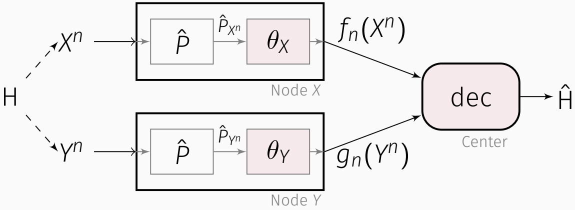
$$\mathcal{X} = \mathcal{Y} = \{0, 1\}$$

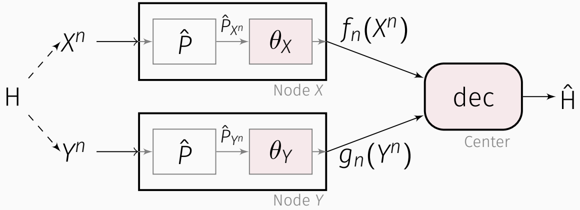
$$P_{XY}^{(0)} = \begin{bmatrix} 1/6 & 1/6 \\ 1/6 & 1/2 \end{bmatrix}, P_{XY}^{(1)} = \begin{bmatrix} 1/2 & 1/6 \\ 1/6 & 1/6 \end{bmatrix}$$

- Local Decision: $(0_1, \log 2)$ or $(\log 2, 0_1)$
- One-bit: $(0_2, 0_2)$
- One-trit: $(0_3, 0_3)$
- Non-distributed: $(\log 2, \log 2)$

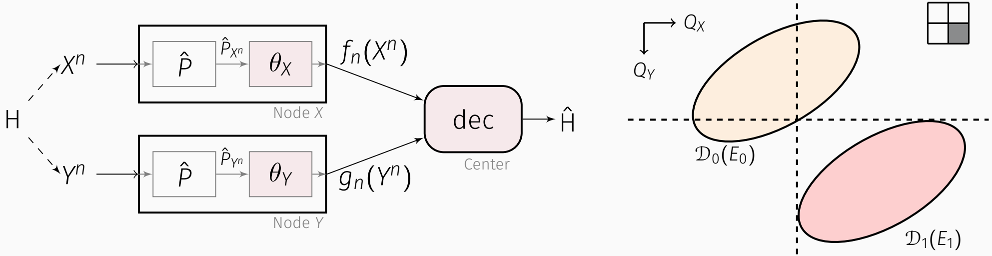


SUMMARY: DHT WITH CONSTANT COMMUNICATION BITS



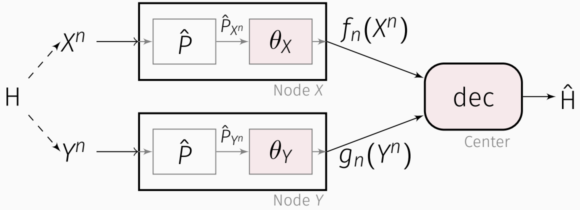


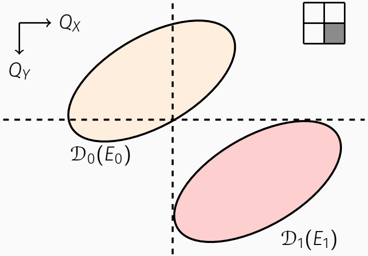
- Optimality of Encoding Types: Sequence Space $\rightarrow$ Distribution Space



- Optimality of Encoding Types: Sequence Space $\rightarrow$ Distribution Space
- Geometric Interpretation: Separating Divergence Balls

SUMMARY: DHT WITH CONSTANT COMMUNICATION BITS





- Optimality of Encoding Types: Sequence Space $\to$ Distribution Space
- Geometric Interpretation: Separating Divergence Balls
- Threshold Decoders: Error Exponents & Optimality