CSCIT 2021 - Lecture 1

Clément Canonne (University of Sydney)

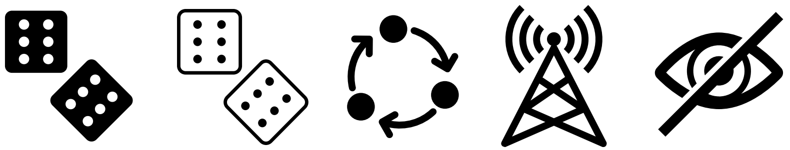
Estimation and hypothesis testing under information constraints

Contents of this lecture

1. What are learning and testing?
2. Baseline: the "centralised" setting
3. Beyond the centralised setting: 3 flavours
    - Private-coin protocols
    - Public-coin protocols
    - Interactive protocols
4. What are information constraints?
    - Two guiding examples: communication and privacy

Contents of this lecture

1. What are learning and testing?
2. Baseline: the "centralised" setting
3. Beyond the centralised setting: 3 flavours
   - Private-coin protocols
   - Public-coin protocols
   - Interactive protocols
4. What are information constraints?
   - Two guiding examples: communication and privacy



What are learning and testing?

Standard statistical setting: n iid samples from some unknown probability distribution p

What are learning and testing?

Standard statistical setting: n iid samples from some unknown probability distribution p

Goal: estimate something about p

What are learning and testing?

Standard statistical setting: n iid samples from some unknown probability distribution p

<u>Goal</u>: estimate something about p

→ learn p : output $\hat{p}$ such that

$$\mathbb{E}[\ell(\hat{p}, p)] \leq \varepsilon$$

What are learning and testing?

Standard statistical setting: n iid samples from some unknown probability distribution p

<u>Goal</u>: estimate something about p

→ learn p : output $\hat{p}$ such that

$$\mathbb{E}_p[\ell(\hat{p}, p)] \leq \varepsilon$$

depends on the n samples

loss function

target rate

What are learning and testing?

Examples of loss functions:

- $KL(p \| q) = -\sum_x p(x) \log \frac{q(x)}{p(x)}$
- $\ell_2^2(p,q) = \sum_x (p(x) - q(x))^2$
- $\chi^2(p,q) = \sum_x \frac{(p(x) - q(x))^2}{q(x)}$
- $TV(p,q) = \sup_S (p(S) - q(S)) = \frac{1}{2} \sum_x |p(x) - q(x)|$

What are learning and testing?

Examples of loss functions:

- $KL(p\|q) = -\sum_x p(x) \log \frac{q(x)}{p(x)}$
- $\ell_2^2(p,q) = \sum_x (p(x)-q(x))^2$
- $\chi^2(p,q) = \sum_x \frac{(p(x)-q(x))^2}{q(x)}$
- $TV(p,q) = \sup_S (p(S)-q(S)) = \frac{1}{2}\sum_x |p(x)-q(x)|$

What are ~~learning~~ estimating and testing?

Standard statistical setting: n iid samples from some unknown probability distribution p

Goal: estimate something about p

What are ~~learning~~ estimating and testing?

Standard statistical setting: n iid samples from some unknown probability distribution p

Goal: estimate something about p

↳ learn a parameter/functional θ of p
output $\hat{\theta}$ such that

$$\mathbb{E}_p\left[\ell(\hat{\theta}, \theta(p))\right] \leq \varepsilon$$

What are ~~learning~~ estimating and testing?

Standard statistical setting: n iid samples from some unknown probability distribution p

Goal: estimate something about p

$\hookrightarrow$ learn a parameter/functional θ of p

output $\hat{\theta}$ such that

$$\mathbb{E}_p\left[\ell(\hat{\theta}, \theta(p))\right] \leq \varepsilon$$

for instance, the mean of p

What are learning and testing?

Standard statistical setting: n iid samples from some unknown probability distribution p

<u>Goal</u>: estimate something about p

↳ is p what I thought it was?

What are learning and testing?

Standard statistical setting: n iid samples from some unknown probability distribution p

Goal: estimate something about p

↳ Hypothesis: $\mathcal{H}_0 =$ "$p = q$" (null)

$\mathcal{H}_1 =$ "$\mathrm{TV}(p,q) > \varepsilon$" (altern.)

Output $\hat{b} \in \{0,1\}$ s.t. $\mathbb{P}_q\{\hat{b}=1\} + \sup_{p \in \mathcal{H}_1} \mathbb{P}_p\{\hat{b}=0\} \leq \frac{1}{10}$

What are learning and testing?

Standard statistical setting: n iid samples from some unknown probability distribution p

<u>Goal:</u> estimate something about p

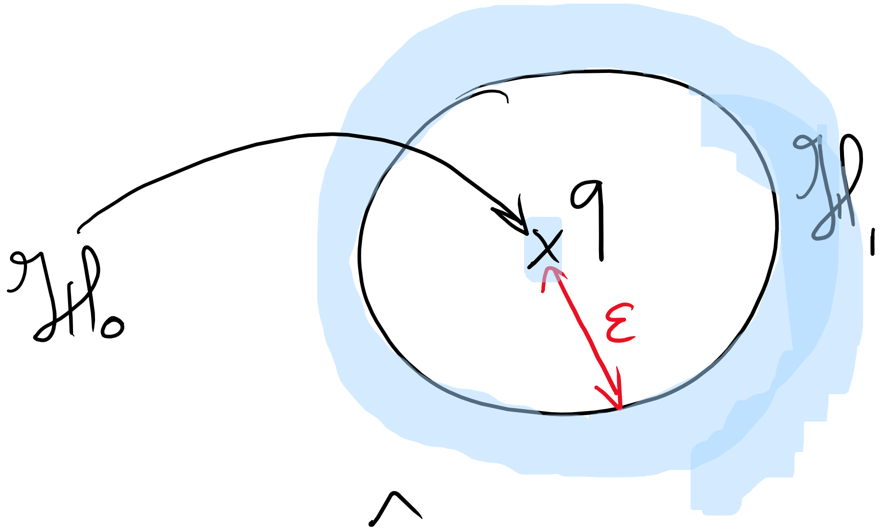
↳ Hypothesis: $\mathcal{H}_0 =$ "$p = q$" (null)

$\mathcal{H}_1 =$ "$\mathrm{TV}(p,q) > \varepsilon$" (altern.)

Output $\hat{b} \in \{0,1\}$ s.t. $\boxed{\mathbb{P}_q\{\hat{b}=1\} + \sup_{p\in\mathcal{H}_1} \mathbb{P}_p\{\hat{b}=0\} \leq \frac{1}{10}}$

What are learning and testing?

<u>Goal</u>: estimate something about p



$\mathcal{H}_0 =$ "$p = q$" (null)

$\mathcal{H}_1 =$ "$\mathrm{TV}(p,q) > \varepsilon$" (altern.)

Output $\hat{b} \in \{0,1\}$ s.t. $\mathbb{P}_q\{\hat{b} = 1\} + \sup_{p \in \mathcal{H}_1} \mathbb{P}_p\{\hat{b} = 0\} \le \frac{1}{10}$

TYPE I TYPE II $\approx$ arbitrary

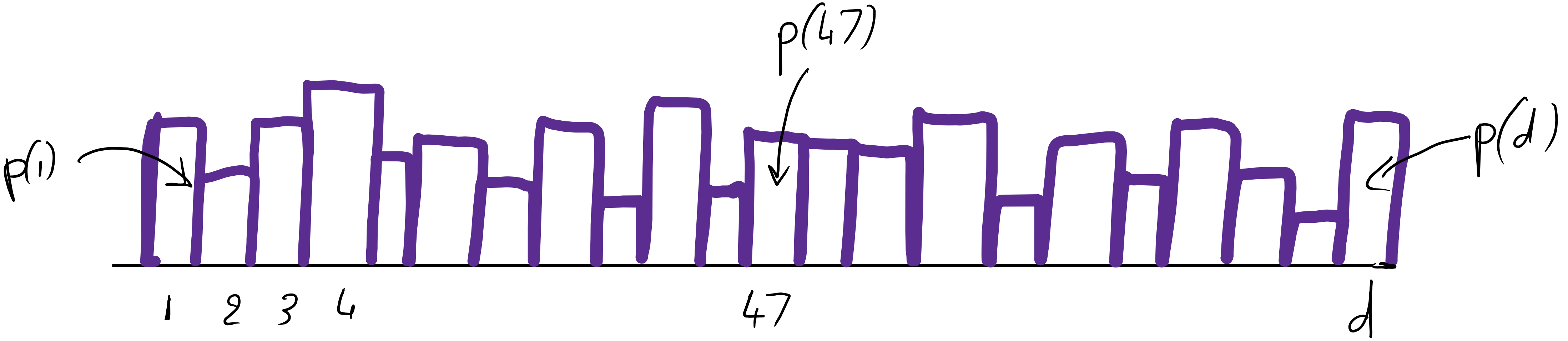
Now, what are we learning and testing?

Now, what are we learning and testing?

(1) Discrete distributions over d elements: $[d] := \{1, 2, -, d\}$

→ Learning under TV loss

→ Testing if uniform on $[d]$



Now, what are we learning and testing?

② High-dimensional Gaussians (with identity covariance)

dimension d

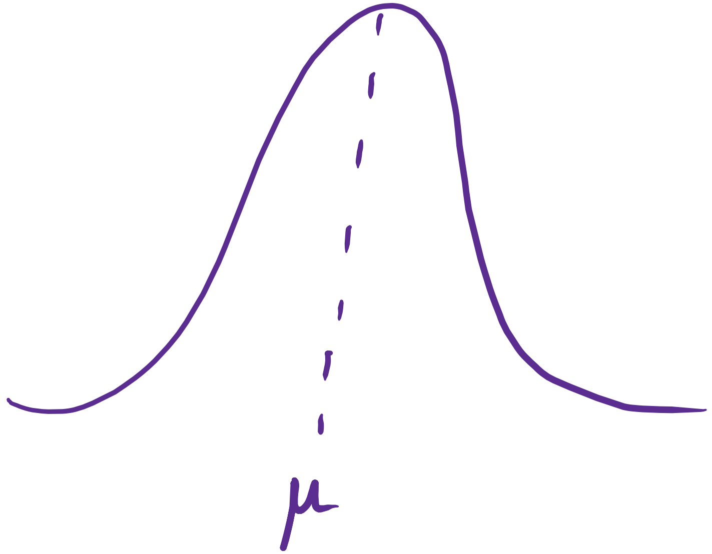
Now, what are we learning and testing?

② High-dimensional Gaussians (with identity covariance)

dimension d

Learning the mean under ℓ_2^2 loss

Testing if the mean is zero (also ℓ_2)



$$p = \mathcal{N}(\mu, I_d)$$

$\in \mathbb{R}^d$

Baseline: the "centralised" setting

$X_1, X_2, \dots, X_n \sim p$ fully accessible to the algorithm.

How large must n be to solve the learning or testing question?

Baseline: the "centralised" setting

$X_1, X_2, \dots, X_n \sim p$ fully accessible to the algorithm.

How large must n be to solve the learning or testing question? (as a function of d, ε)

"Minimax sample complexity"

Baseline: the "centralised" setting

Discrete distributions

$d \gg 1$

$\varepsilon \in (0,1]$

Theorem. Learning an arbitrary p over $[d]$ to TV loss ε has sample complexity ____.

Theorem. Testing if an arbitrary p over $[d]$ is u or has $\mathrm{TV}(p,u) > \varepsilon$ has sample complexity ____.

uniform over $[d]$

Baseline: the "centralised" setting

Discrete distributions

$d \gg 1$

$\varepsilon \in (0,1]$

Theorem. Learning an arbitrary p over $[d]$ to TV loss ε has sample complexity $\Theta\left(\frac{d}{\varepsilon^2}\right)$.

Theorem. Testing if an arbitrary p over $[d]$ is u or has $\mathrm{TV}(p,u) > \varepsilon$ has sample complexity ________.

Baseline: the "centralised" setting

Discrete distributions

$d \gg 1$

$\varepsilon \in (0,1]$

Theorem. Learning an arbitrary p over $[d]$ to TV loss ε has sample complexity $\Theta\left(\frac{d}{\varepsilon^2}\right)$.

Theorem. Testing if an arbitrary p over $[d]$ is u or has $\mathrm{TV}(p,u) > \varepsilon$ has sample complexity $\Theta\left(\frac{\sqrt{d}}{\varepsilon^2}\right)$.

Baseline: the "centralised" setting

Discrete distributions

$d \gg 1$

$\varepsilon \in (0, 1]$

Proof.

Baseline: the "centralised" setting

$d \gg 1$

$\varepsilon \in (0,1]$

Identity-covariance Gaussians

Theorem. Learning the mean of an unknown $\mathcal{N}(\mu, I_d)$ to ℓ_2^2 loss ε^2 has sample complexity ___ .

Theorem. Testing if an unknown $\mathcal{N}(\mu, I_d)$ has $\mu = 0_d$ vs. $\|\mu\|_2 > \varepsilon$ has sample complexity ___ .

Baseline: the "centralised" setting

$d \gg 1$

$\varepsilon \in (0,1]$

Identity-covariance Gaussians

Theorem. Learning the mean of an unknown $\mathcal{N}(\mu, I_d)$ to ℓ_2^2 loss ε^2 has sample complexity $\Theta\left(\frac{d}{\varepsilon^2}\right)$.

Theorem. Testing if an unknown $\mathcal{N}(\mu, I_d)$ has $\mu = 0_d$ vs. $\|\mu\|_2 > \varepsilon$ has sample complexity ________.

Baseline: the "centralised" setting

$d \gg 1$

$\varepsilon \in (0,1]$

Identity-covariance Gaussians

Theorem. Learning the mean of an unknown $\mathcal{N}(\mu, I_d)$ to ℓ_2^2 loss ε^2 has sample complexity $\Theta\left(\frac{d}{\varepsilon^2}\right)$.

Theorem. Testing if an unknown $\mathcal{N}(\mu, I_d)$ has $\mu = 0_d$ vs. $\|\mu\|_2 > \varepsilon$ has sample complexity $\Theta\left(\frac{\sqrt{d}}{\varepsilon^2}\right)$.

Baseline: the "centralised" setting

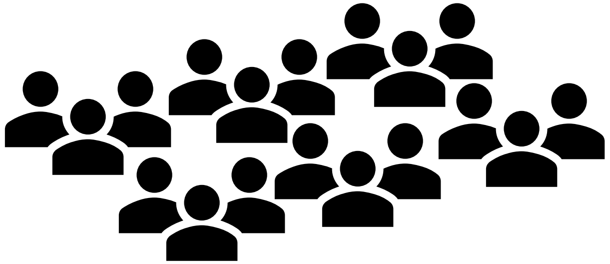
$d \gg 1$

$\varepsilon \in (0, 1]$

Identity - covariance Gaussians

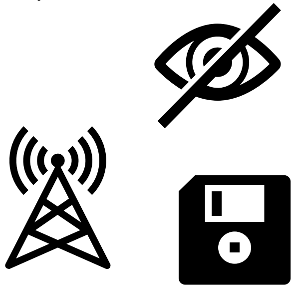
Proof.

Beyond the centralised setting



Distributed data

Information or computational constraints



Limited types of measurements

Beyond the centralised setting

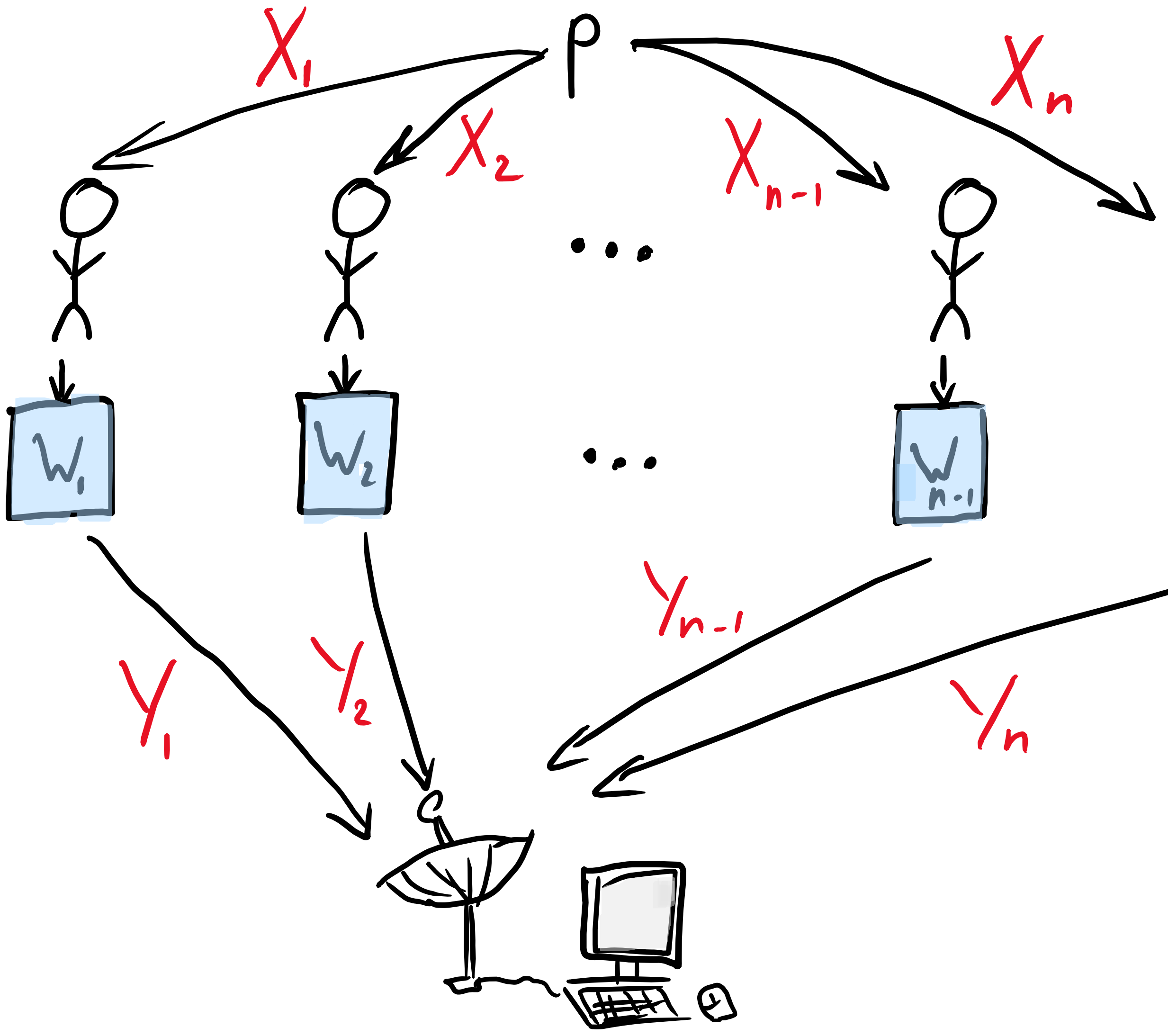
- n users, each holding <u>one</u> sample from (same) p
- One center, which has no sample but needs to solve the learning/testing task
- Each user can only send a "limited" type of message

Beyond the centralised setting

- n users, each holding <u>one</u> sample from (same) p
- One center, which has no sample but needs to solve the learning/testing task
- Each user can only send a "limited" type of message

"local" constraint

Beyond the centralised setting



Channels
$W_1, \ldots, W_n \in \mathcal{W}$

Beyond the centralised setting

Channel: $W: \mathcal{X} \longrightarrow \mathcal{Y}$ randomised

input space (samples)

output space (messages)

Notation: $W(y \mid x) = \mathbb{P}\{W(x) = y\}$

Beyond the centralised setting

Channel: $W : \mathcal{X} \longrightarrow \mathcal{Y}$ randomised

input space (samples) — output space (messages)

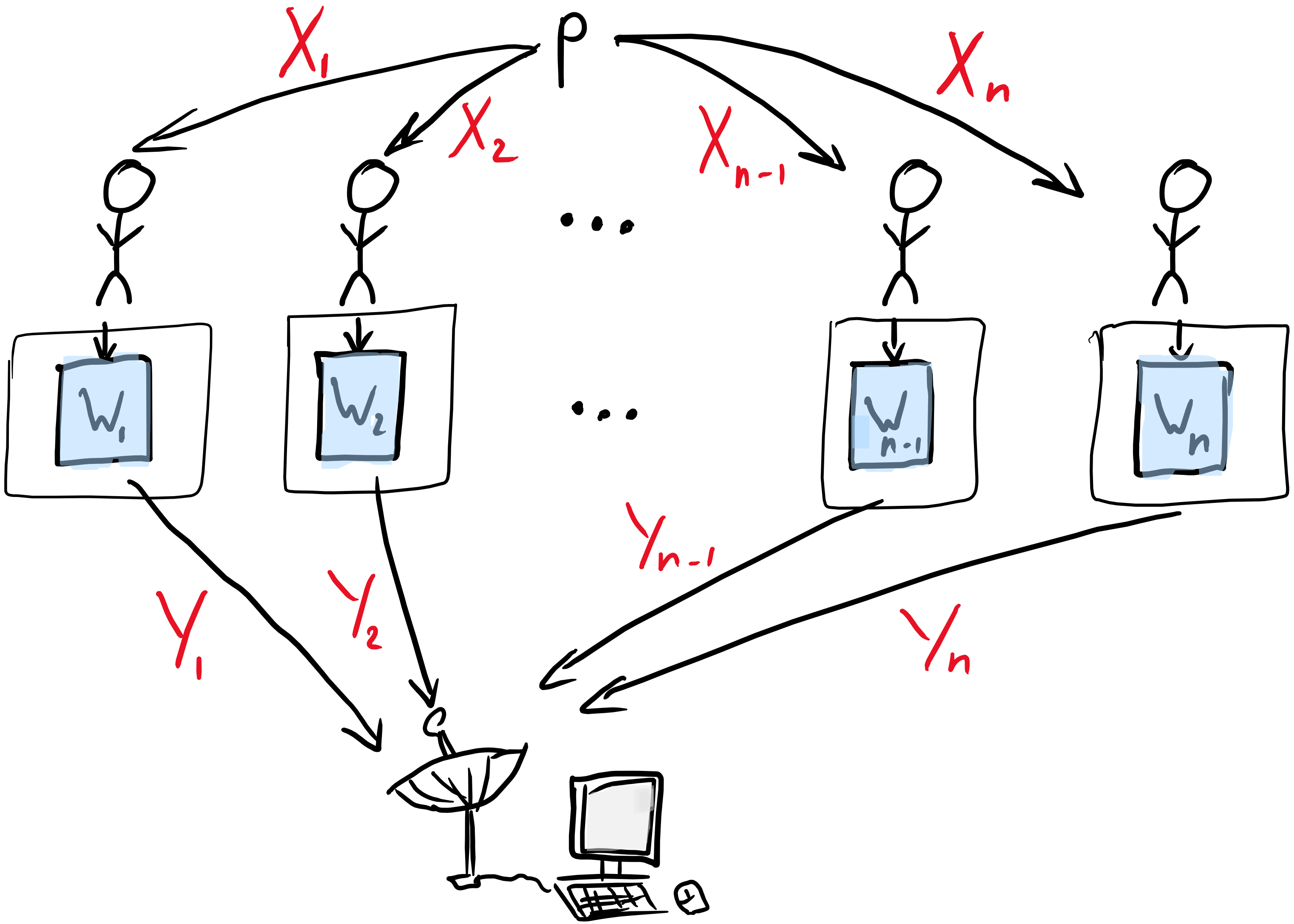
Notation: $W(y \mid x) = \mathbb{P}\{W(x) = y\}$

$\mathcal{W} \subseteq \{\mathcal{X} \to \mathcal{Y}\}$ set of **allowed channels**

What happens if $\mathcal{W}$ contains the identity mapping?

Beyond the centralised setting

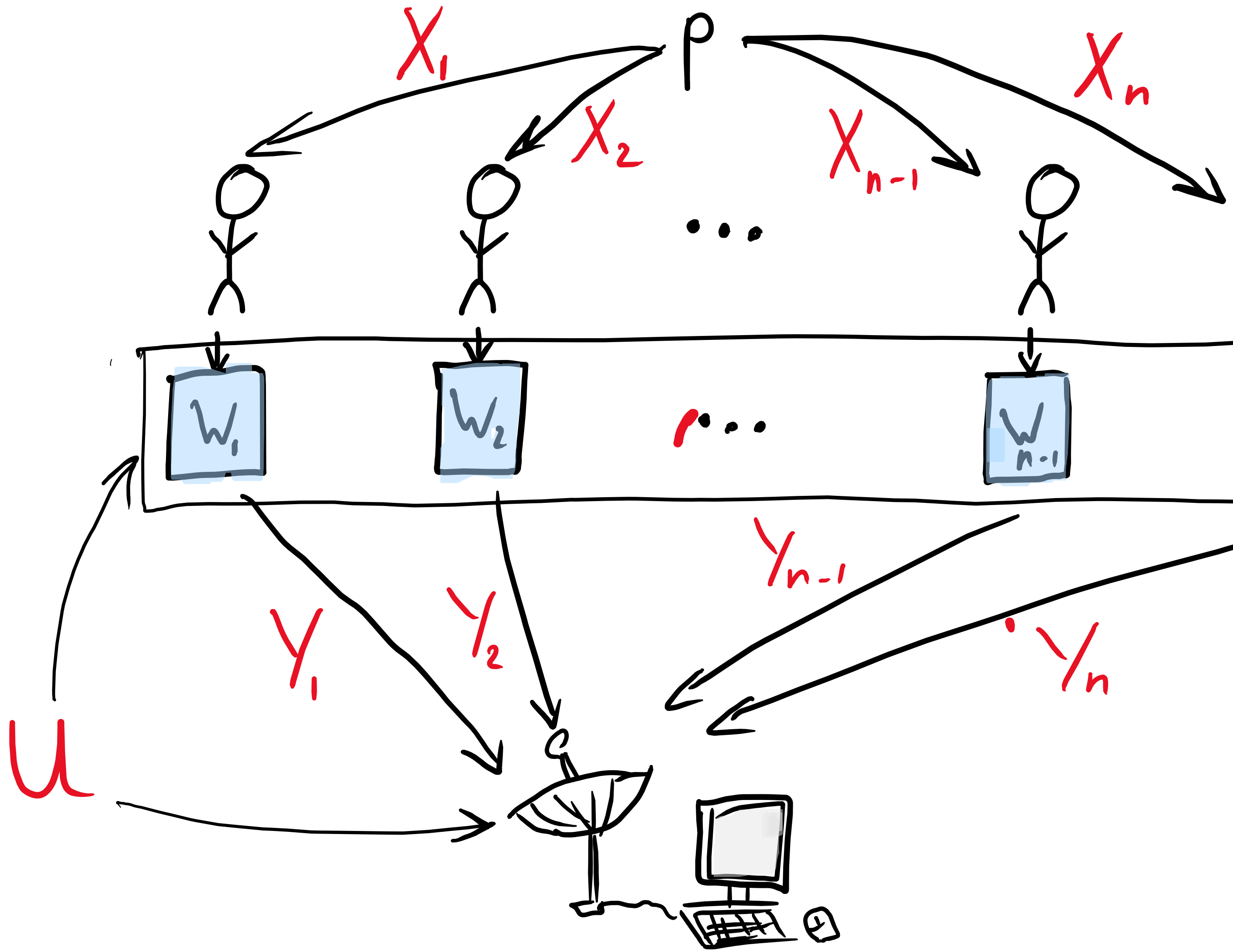
Private-coin



Channels $W_1, \ldots, W_n \in \mathcal{W}$ independently randomised

Beyond the centralised setting

Public-coin

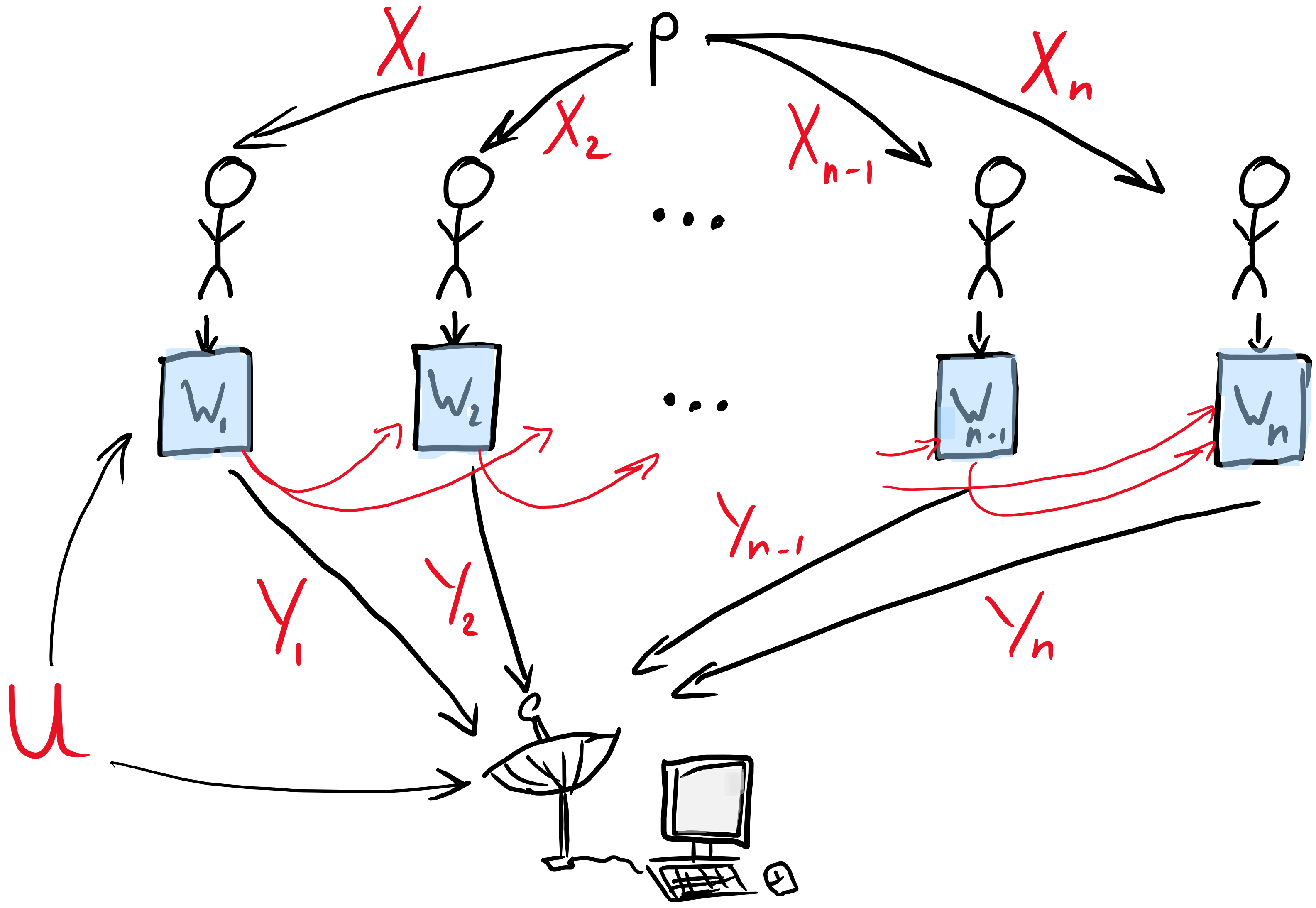


Channels $W_1, \ldots, W_n \in \mathcal{W}$

- jointly randomised

Beyond the centralised setting

Interactive



Channels $W_1, \dots, W_n \in \mathcal{W}$

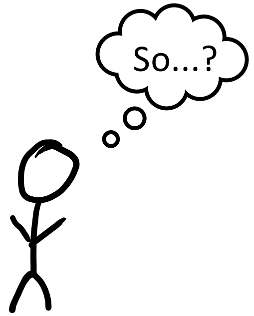
$W_t = W^{Y_1, \dots, Y_{t-1}}$

depends on previous messages

(+ public randomness)

Beyond the centralised setting

Implementation and deployment	Private-coin ≤ Public-coin ≤ Interactive
Sample complexity	Private-coin ≥ Public-coin ≥ Interactive



Two guiding examples of channel families

Communication

Each user can only send ℓ bits

$\mathcal{W}_\ell = \{ W : \mathcal{X} \to \{0,1\}^\ell \}$

Local Privacy

Each user requires ε- differential privacy

$\forall W \in \mathcal{W}_\varepsilon$

$\forall y, x, x', \quad W(y|x) \leq e^\varepsilon W(y|x')$

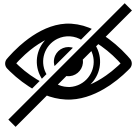
Two guiding examples of channel families

Communication

Each user can only send ℓ bits

$$\mathcal{W}_\ell = \{ W : \mathcal{X} \to \{0,1\}^{\ell} \}$$

Local Privacy



Each user requires ρ- differential privacy

$\forall W \in \mathcal{W}_\rho$

$$\forall y, x, x', \quad W(y \mid x) \leq e^{\rho} W(y \mid x') \approx (1+\rho) W(y \mid x')$$

(think of $\rho \in (0,1]$)

Two guiding examples of channel families

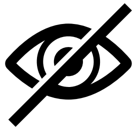
Communication

Each user can only send ℓ bits

$$\mathcal{W}_\ell = \{ W : \mathcal{X} \to \{0,1\}^\ell \}$$

Can't send too much

Local Privacy



Each user requires ρ- differential privacy

$\forall W \in \mathcal{W}_\rho$

$$\forall y, x, x', \quad W(y \mid x) \leq e^\rho \, W(y \mid x')$$

Can't reveal too much

Recap: this lecture

1. What are learning and testing? ☑
2. Baseline: the "centralised" setting ☑
3. Beyond the centralised setting: 3 flavours ☑
    - Private-coin protocols
    - Public-coin protocols
    - Interactive protocols
4. What are information constraints? ☑
    - Two guiding examples: communication and privacy

Next lecture:

Learning and testing discrete distributions under

those information constraints