# Privacy Preserving Computation of Medical Statistics

Amal Mawass and Jan Ramon

# Table of Contents

# Context

- Each party (hospital) owns a data set (results of a clinical trial for example) and each party doesn't want to share its data with the others.
- Protocol:
    1. Compute a U-statistic: Agree on the definition of a U-statistic, and all parties collaborate in a encrypted way to compute the U-statistic based on the union of the datasets.
    2. Publish: Send a message to all parties, for example send a model or other information
- E.g., linear regression, decision trees ...

# Definition

For $m \in \mathbb{N}_0$ and $\Phi : \mathbb{R}^m \to \mathbb{R}$ a symmetric function.

For a sample $X = (x_1, \ldots, x_n) \in \mathbb{R}^{n\times m}$ of size $n \geq m$ :

$$U(X) = \binom{n}{m}^{-1} \sum_{i_1 < \cdots < i_r} \Phi(x_{i_1}, \ldots, x_{i_m})$$

is called a U-statistic of order $m$ and kernel $\Phi$.

This statistic is the mean of $\Phi(x_{i_1}, \ldots, x_{i_m})$ on all the $m$-subsets $\{x_{i_1}, \ldots, x_{i_m}\}$ of $\{x_1, \ldots, x_n\}$

# Examples

- The classical estimator of the empirical mean $\overline{X} = \frac{1}{n}\sum_{i=1}^{n} x_i$ is a U-statistic of order 1, and of kernel $\Phi : x \mapsto x$.

- The unbiased variance estimator $\widetilde{S}^2(X) = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \overline{X})^2$ can be rewritten $S(X) = \frac{2}{n(n-1)}\sum_{i<j}\frac{(x_i-x_j)^2}{2}$ . It is therefore a U-statistic of order 2 and of kernel $\Phi : (x_1, x_2) \mapsto \frac{(x_1-x_2)^2}{2}$.

# The Hosmer-Lemeshow test

This is a calibration test for the logistic regression model. This model predicts the probability of a certain class or event, e.g., success/fail. The Hosmer-Lemeshow test aims to compare the expected and observed counts.

# Computation

1. Compute $p(success)$ for the $n$ subjects using the model
2. Order the probabilities
3. Divide them in $Q$ groups with approximately the same number of subjects in each group (normally $Q = 10$)
4. Compute the expected (predicted) counts $(E_{0i}, E_{1i})$ and observed (in the dataset) counts $(0_{0i}, O_{1i})$ in each group $i$.
5. Compute the Hosmer-Lemeshow test:

$$H = \sum_{i=1}^{Q} \left( \frac{(O_{1i} - E_{1i})^2}{E_{1i}} + \frac{(O_{0i} - E_{0i})^2}{E_{0i}} \right),$$

6. Compare the statistic to a $\mathcal{X}^2(Q-2)$

# Algorithm (1/2)

**function** Search($a, b \in \mathbb{R}$ : probability interval to search; $t$ : target)
    **while** $b - a > \psi$ **do** ▷ $\psi$ = precision / smallest probability
        $m \leftarrow (a + b)/2$ ▷ Consider middle
        $f_m \leftarrow UStat(\Phi((p, o)) := \mathbb{1}[p \leq f_m])$ ▷ Compute U-statistic
        **if** $f_m < t$ **then** ▷ Split interval
            $b \leftarrow m$
        **else**
            $a \leftarrow m$
        **end if**
    **end while** ▷ Until interval small enough
    **return** $(a + b)/2$
**end function**

# Algorithm (2/2)

**function** HLstat($Q$ : number of groups)
  $t_0 \leftarrow 0$ ; $t_Q \leftarrow 1 + \psi$
  **for** $i = 1 \ldots Q - 1$ **do**
    $t_i \leftarrow$Search$(t_{i-1}, 1, i/Q)$
  **end for**
  **for** $i = 1 \ldots Q$, $s \in \{0, 1\}$ **do**
    $E_{s,i} \leftarrow Q.UStat(\Phi((p, o)) := p\mathbb{1}[t_{i-1} \leq p \leq t_i])$
    $O_{s,i} \leftarrow Q.UStat(\Phi((p, o)) := o\mathbb{1}[t_{i-1} \leq p \leq t_i])$
  **end for**
  **return** $\sum_{i=1}^{Q} \left( \frac{(O_{1i} - E_{1i})^2}{E_{1i}} + \frac{(O_{0i} - E_{0i})^2}{E_{0i}} \right)$
**end function**

# Some results

**Complexity:**
This algorithm has a complexity of $O(Q \log(1/\psi))$.
The number of computed U-statistics is $O((Q-1)\log(1/\psi)+3Q)$.

**Privacy:**
**Thm 1.** This algorithm achieves $\epsilon$-DP if $UStat(\Phi)$ adds $Lap\left(\frac{(Q-1)\log(1/\psi)+3Q}{n\epsilon}\right)$ noise.
**Proof:** Classic DP-composition, dividing the privacy budget equally over all computed $U$-statitics.

Can we do better?

# sequence DP

**Lemma [Continual observation, Dwork]:**

- Setup:
  - Let $N \in \mathbb{N}$ and $L = \lceil \log_2(N) \rceil$.
  - Consider a dataset / sequence $Y = (y_1 \dots y_N) \in \{0, 1\}^N$.
  - Let $c_i = \sum_{j=1}^{i} y_j$ be partial sums.
- Let Algorithm $A$ do:
  - For $1 \le j \le L$ and $1 \le I \le \lceil N/2^j \rceil$, let $\eta_{I,j} \sim \mathrm{Lap}(1/\epsilon')$.
  - For each $i = 1 \dots N$, publish the noisy partial sum $\hat{c}_i = c_i + \sum_{j=1}^{L} \eta_{\lceil i/2^j \rceil, j}$
- Then, $A$ is $L\epsilon'$-differentially private

# Mapping our case to sequence

- Let $N \in \mathbb{N}$ and $L = \lceil \log_2(N) \rceil$:
  - $N = 1/\psi + 1$
  - $L = \log_2(1/\psi)$
- Consider a dataset / sequence $Y = (y_0 \dots y_N) \in \{0,1\}^N$:
  - $y_i = \sum_{j=1}^{n} \mathbb{1}[(i-1)\psi < p_j \leq i\psi]$
  - Assume for simplicity that $j \neq j' \Rightarrow |p_j - p_{j'}| \geq \psi$
    - $y_i \in \{0,1\}$.
    - e.g., make $\psi' = \psi/n$ a bit smaller & add 'noise' $p_i' = p_i + i\psi'$
- Let $c_i = \sum_{j=1}^{i} y_j$ be partial sums.
  - $c_i = \sum_{j=1}^{n} \mathbb{1}[p_j \leq i\psi]$

# Apply continual observation lemma

- Adjacent datasets:
  - Dwork's continual observation lemma: one $y_i$ changes
  - Our case: one $y_i$ from 1 to 0, one $y_i$ from 0 to 1.
  - Need a factor of 2.
- We don't query all $\hat{c}_i = \sum_{j=1}^{n} \mathbb{1}[p_j \leq i\psi]$ but only a few needed in the binary searches. A fortiori the lemma applies.
- Sensitivity: $1/n$
- In function *Search*, $\eta_{I,j} \sim \mathrm{Lap}(1/n\epsilon')$

$$UStat(\Phi((p,o)) := \mathbb{1}[p \leq f_m]) = \frac{1}{n}\sum_{j=1}^{n} \mathbb{1}[p_i \leq f_m] + \sum_{j=1}^{L} \eta_{\lceil i/2^j \rceil, j}$$

- then union of calls to *Search* is $2L\epsilon'$-DP

# Differential privacy

**Thm 2.** *HLStat achieves $\epsilon$-DP with a scheme adding $\lceil \log_2(1/\psi) \rceil + 1$ terms of Lap $\left(\frac{\lceil 2\log(1/\psi)\rceil + 2 + 4Q}{n\epsilon}\right)$ noise terms in UStat calls in Search and one Lap $\left(\frac{\lceil 2\log(1/\psi)\rceil + 2 + 4Q}{n\epsilon}\right)$ noise term in UStat calls in HLstat.*

**Proof:**

- Calls to *UStat* in main function *HLstat*:
  - Purpose: calculate $E_{s,i}$ and $O_{s,i}$
  - count: $4Q$
  - Sensitivity: $1/n$ (all probabilities between 0 and 1)
  - Every call is $\epsilon'$-DP
  - Total: $4Q\epsilon'$-DP
- Calls to *UStat* in function *Search*: $2L\epsilon'$-DP
- Set $\epsilon' = \epsilon/(4Q + 2L)$: total algorithm is $\epsilon$-DP.

# Better?

Variances of results of calls to *UStat*:

| | *HLStat* | *Search* |
|---|---|---|
| Thm 1 | $O\left(\frac{Q^2 \log^2(1/\psi)}{n^2\epsilon^2}\right)$ | $O\left(\frac{Q^2 \log^2(1/\psi)}{n^2\epsilon^2}\right)$ |
| Thm 2 | $O\left(\frac{Q^2+\log^2(1/\psi)}{n^2\epsilon^2}\right)$ | $O\left(\frac{\log(1/\psi)(Q^2+\log^2(1/\psi))}{n^2\epsilon^2}\right)$ |

# Open questions

- What is the error due to the noise needed to achieve $\epsilon$-differential privacy?
- Is Thm 2 optimal?

# Future work

- Filling more gaps in DP computation of medical statistics
    - especially those typically used during medical studies
- Relevant PMR project tasks
    - WP2: Assessing privacy risk
    - T4.2: Use case: multi-centric studies
    - T3.1: Combining/optimizing (noise-based and cryptography-based) approaches