21.12.2022 ***פרטיות מידע***

Lecture 8: The Shuffle Model

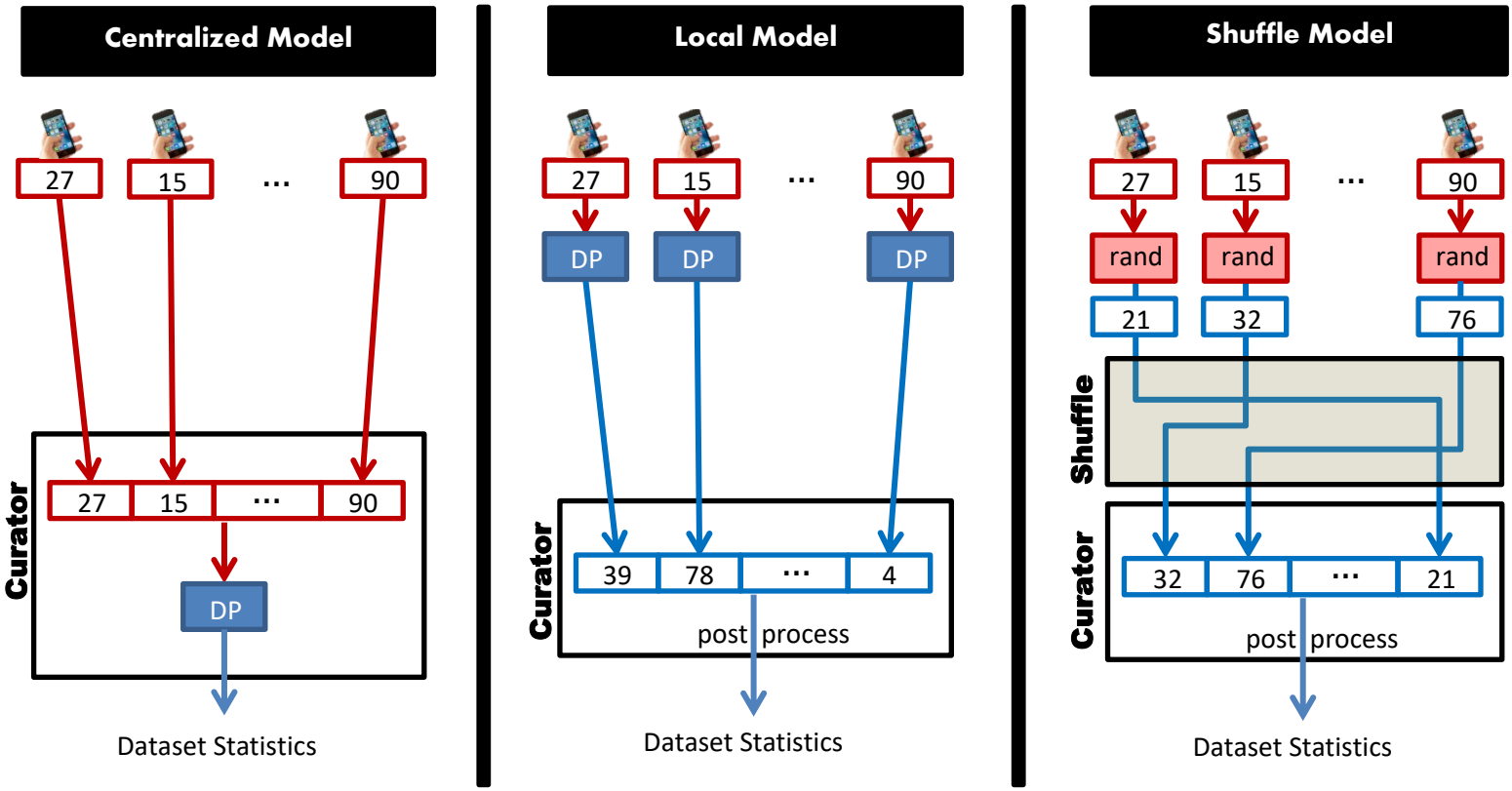
Textbook: Cynthia Dwork and Aaron Roth. The Algorithmic Foundations of Differential Privacy

מרצה: אורי שטמר

<u>מוטיבציה:</u>

כשדיברנו על המודל הלוקלי, השגנו מודל *trust* חזק מאוד, אבל המחיר היה שהשגיאה שלנו גדלה מאוד, וזה יכול להיות מאוד משמעותי בפועל. למשל, ראינו שבמודל הלוקלי אנחנו יכולים לספור ביטים (כלומר הקלט של כל משתמש הוא ביט ואנחנו רוצים לדעת כמה 1-ים יש בסה"כ) עם שגיאה בערך $\frac{\sqrt{n}}{\varepsilon}$ בעוד שבמודל הריכוזי אנחנו יכולים לספור ביטים עם שגיאה בערך $\frac{1}{\varepsilon}$ פשוט על ידי הוספת רעש לפלאסי.

השאלה היא האם אפשר ליהנות משני העולמות, כלומר האם אפשר להגדיר מודל "ביניים" שבו נהנה מהבטחת *trust* כמו במודל הלוקלי ומאיכות תוצאות כמו במודל הריכוזי?



הגדרה:

- ישנם n משתתפים ושרת.
- כל משתתף i מחזיק קלט $x_i \in D$
- כל משתתף i מפעיל על הקלט שלו (באופן מקומי) אלגוריתם אקראי R_i לקבלת ℓ הודעות $(m_{i,1}, \dots, m_{i,\ell}) \leftarrow R_i(x_i)$
- כל משתתף i מזין את ℓ ההודעות הנ"ל לערוץ תקשורת מיוחד, הנקרא ערוץ *shuffle*.
- במוצא של ערוץ ה *shuffle* מתקבלת פרמוטציה אקראית של האוסף של $n\ell$ ההודעות שהוזנו לו.
הפרמוטציה הזאת גלויה לכולם (גם למשתמשים וגם לשרת). כלומר כולם רואים את הפלט של ה *shuffle*.
נסמן:
$$\text{Shuffle}(m_{1,1}, \dots, m_{1,\ell}, \dots, m_{n,1}, \dots, m_{n,\ell})$$
- לאחר מכן השרת מבצע *post-processing* של הפלט של ה *shuffle* על מנת לקבל את תוצאת החישוב.

דרישת הפרטיות:
הפלט של ה *shuffle* מקיים פ"ד. כלומר, לכל זוג דטהבייסים (מבוזרים) שכנים $X, X' \in D^n$ ולכל אוסף F של פלטים אפשריים של ה *shuffle* מתקיים:
$$\Pr\left[\text{Shuffle}\left(R_1(x_1), \dots, R_n(x_n)\right) \in F\right] \leq e^{\varepsilon} \cdot \Pr\left[\text{Shuffle}\left(R_1(x'_1), \dots, R_n(x'_n)\right) \in F\right] + \delta$$

למה זאת דרישת פרטיות טובה?
בהנחה שיש לנו ערוץ *shuffle* כזה, אנחנו מקבלים שפרטיות מובטחת כבר במוצא שלו. בפרט, השרת "לא לומד כלום" על הקלט של אף משתמש ספציפי.

% במציאות אין לנו באמת ערוץ כזה, ובפועל חברות ממשות ערוצים כאלה בעזרת כל מני היוריסטיקות. למשל, המשתמשים שולחים את ההודעות שלהם מוצפנות לאיזשהו "שרת קטן" שמערבב אותן ומעביר אותן לאחר מכן לשרת אחר שמבצע את החישוב. היתרון כאן הוא שמספיק לנו לסמוך שהחברה מימשה נכון את הערוץ הזה, שזאת משימה שנראית (לכאורה) פשוטה למימוש.

הערה: איך שהגדרת הפרטיות כאן כתובה, אנחנו מניחים שכל המשתמשים "משחקים לפי הכללים". כלומר אנחנו מניחים שכל משתמש i באמת מפעיל $R_i(x_i)$ ושולח את ההודעות שהוא מקבל ל *shuffle*.

איפה בדיוק אנחנו משתמשים בהנחה הזאת?
אחרת דרישת הפרטיות לא אומרת לנו כלום. כלומר, לכאורה דרישת הפרטיות כאן עלולה להישבר לחלוטין אם חלק מהמשתמשים מנסים "לשבש את המערכת" ולשלוח ל *shuffle* הודעות שהם לא אמורים לשלוח לשם.
בפועל יש דרכים להתמודד עם זה, אבל אנחנו כאן נניח לשם פשטות שכל המשתמשים משחקים לפי הכללים.

מה הרווחנו? מבחינת מה שאנחנו יכולים לחשב במודל הזה, הוא יושב איפשהו באמצע בין המודל הלוקלי למודל הריכוזי.

אבחנה 1: כל מה שאפשר לחשב במודל ה *shuffle* אפשר גם לחשב במודל הריכוזי.

הסבר: אם יש לנו פרוטוקול במודל ה *shuffle*, נוכל להריץ אותו גם במודל הריכוזי ע"י כך שהשרת יסמלץ את ה *shuffle* בעצמו (הרי יש לו את כל ה *data*).

משפט 2: במודל ה *shuffle* אפשר לסכום ביטים עם שגיאה $\approx \frac{1}{\varepsilon}$

<u>הוכחה:</u>
כדי להוכיח את המשפט הזה עלינו לתכנן פרוטוקול *shuffle* מתאים. כלומר, עלינו להגיד מה כל משתמש עושה (איך הוא מייצר את ההודעות שהוא שולח מתוך הקלט שלו) ומה השרת עושה אחרי שהוא רואה את המוצא של ה *shuffle*.

הנחה מפשטת: נניח שמתקיים $n \gg \frac{1}{\varepsilon^2}\ln\left(\frac{1}{\delta}\right)$ וגם $\varepsilon \leq 1$.

אלגוריתם R
קלט: $x \in \{0,1\}$
(1) הגרל $y \sim \text{Bernoulli}(p)$ עבור $p = \frac{48 \cdot \log\left(\frac{2}{\delta}\right)}{\varepsilon^2 \cdot n}$.
(2) החזר $m_1 = x \ , \ m_2 = y$

כלומר, כל משתמש שולח ל *shuffle* את ביט הקלט שלו וגם עוד ביט אקראי.

תחילה נראה שהפרוטוקול הזה משמר פרטיות (במוצא של ה *shuffle*).

נסמן ב $Z = y_1 + y_2 + \cdots + y_n$ את סכום "ביטי הרעש" שכל המשתמשים מכניסים ונסמן ב $B = x_1 + y_1 + \cdots + x_n + y_n$ את סכום כל הביטים במוצא של ה *shuffle*.

<u>אבחנה:</u> על מנת להראות שהמוצא של ה *shuffle* מקיים פ"ד, מספיק להראות ש B מקיים פ"ד.

<u>הסבר:</u> מתוך B נוכל כ *post-processing* לייצר משהו שמתפלג בדיוק כמו המוצא של ה *shuffle* ע"י כך שניקח אוסף של $2n$ ביטים שמתוכם יש B אחדים ונעשה להם פרמוטציה אקראית. לכן, לפי סגירות ל *post-processing* מספיק להראות שפרטיות מתקיימת עבור B.

נשים לב שמתקיים

$$Z \sim \text{Binomial}(n, p)$$

לכן,

$$\mathbb{E}[Z] = np = \frac{48 \cdot \log\left(\frac{2}{\delta}\right)}{\varepsilon^2}$$

בנוסף, לפי חסם צ'רנוף, לכל $0 < \gamma < 1$ מתקיים

$$\Pr[Z > (1+\gamma)np] < \exp\left(-\frac{\gamma^2 np}{3}\right)$$

בפרט, עבור $\gamma = \sqrt{\frac{3}{np}\ln\left(\frac{2}{\delta}\right)}$ מתקיים $\Pr\left[Z > np + \sqrt{3np \cdot \ln\left(\frac{2}{\delta}\right)}\right] < \frac{\delta}{2}$

גם הכיוון השני מתקיים באופן דומה, ולכן $\Pr\left[|Z - np| > \sqrt{3np \cdot \ln\left(\frac{2}{\delta}\right)}\right] < \delta$.

כעת יהי $np - \sqrt{3np \cdot \ln\left(\frac{2}{\delta}\right)} \leq k \leq np + \sqrt{3np \cdot \ln\left(\frac{2}{\delta}\right)}$.

נסמן זאת בקיצור ע"י $k \in I_{\text{Good}} = \left[np - \sqrt{3np \cdot \ln\left(\frac{2}{\delta}\right)} \ , \ np + \sqrt{3np \cdot \ln\left(\frac{2}{\delta}\right)} \right]$

מתקיים:

$$\frac{\Pr[Z=k]}{\Pr[Z=k-1]} = \frac{\binom{n}{k} \cdot p^k \cdot (1-p)^{n-k}}{\binom{n}{k-1} \cdot p^{k-1} \cdot (1-p)^{n-k+1}} = \frac{\frac{n!}{k! \cdot (n-k)!}}{\frac{n!}{(k-1)! \cdot (n-k+1)!}} \cdot \frac{p}{(1-p)}$$

$$= \frac{n-k+1}{k} \cdot \frac{p}{1-p} = \frac{n-k+1}{k} \cdot \frac{pn}{n-pn} = \frac{n-k+1}{n-pn} \cdot \frac{pn}{k}$$

$$\underbrace{\leq}_{\text{חסם על } k} \left(1 + \frac{\sqrt{3np \cdot \ln\left(\frac{2}{\delta}\right)} + 1}{n-pn} \right) \cdot \frac{pn}{pn - \sqrt{3np \cdot \ln\left(\frac{2}{\delta}\right)}}$$

$$\underbrace{\leq}_{\substack{\text{הגדרת } p \\ \text{וההנחה ש } n \text{ גדול}}} \left(1 + \frac{42}{\varepsilon n} \cdot \ln\left(\frac{2}{\delta}\right) \right) \cdot \frac{1}{1 - \sqrt{\frac{3}{pn} \cdot \ln\left(\frac{2}{\delta}\right)}}$$

$$\underbrace{=}_{pn = \frac{48}{\varepsilon^2} \log\left(\frac{2}{\delta}\right)} \left(1 + \frac{42}{\varepsilon n} \cdot \ln\left(\frac{2}{\delta}\right) \right) \cdot \frac{1}{1 - \frac{\varepsilon}{4}}$$

$$\underbrace{\leq}_{\text{ההנחה ש } n \text{ גדול}} \left(1 + \frac{\varepsilon}{4}\right) \cdot \frac{1}{1 - \frac{\varepsilon}{4}} \leq \exp\left(\frac{\varepsilon}{4}\right) \cdot \exp\left(\frac{\varepsilon}{2}\right) \leq e^{\varepsilon}$$

נקבע זוג דטהבייסים שכנים X, X' כך שמספר האחדים ב X' גדול ב- 1 מאשר ב X.
נסמן את מספר האחדים ב X ע"י t.
יהי $F \subseteq \mathbb{N}$ מאורע כלשהו ונסמן

$$F_{-t} = \{f - t : f \in F\}$$

מתקיים

$$\Pr\left[\underset{\text{בריצה על } X}{B \in F}\right] = \Pr[t + Z \in F] = \Pr[Z \in F_{-t}] = \Pr[Z \in F_{-t} \cap I_{\text{Good}}] + \Pr[Z \in F_{-t} \setminus I_{\text{Good}}]$$

$$\leq \Pr[Z \in F_{-t} \cap I_{\text{Good}}] + \Pr[Z \notin I_{\text{Good}}] \leq \Pr[Z \in F_{-t} \cap I_{\text{Good}}] + \delta$$

$$\leq e^{\varepsilon} \cdot \Pr[Z + 1 \in F_{-t} \cap I_{\text{Good}}] + \delta \leq e^{\varepsilon} \cdot \Pr[Z + 1 \in F_{-t}] + \delta = e^{\varepsilon} \cdot \Pr[Z + 1 + t \in F] + \delta$$

$$= e^{\varepsilon} \cdot \Pr\left[\underset{\text{בריצה על } X'}{B \in F}\right] + \delta$$

המקרה בו מספר האחדים ב X גדול ב 1 מאשר ב X' הוא דומה.

סיימנו את ניתוח הפרטיות.

איך השרת מחשב הערכה לסכום הביטים מתוך מוצא ה *shuffle* ?

האלגוריתם בצד של השרת
קלט: אוסף של $2n$ ביטים $b_1, \dots, b_{2n}$ (1) חשב והחזר $\left(\sum_{i=1}^{2n} b_i\right) - np$ עבור $p = \frac{48 \cdot \log\left(\frac{2}{\delta}\right)}{\varepsilon^2 \cdot n}$.

כלומר השרת פשוט סוכם את כל ההודעות (הביטים) במוצא של ה *shuffle*, עד כדי חיסור *np*.

ניתוח השגיאה:
מתקיים

$$\sum_{i=1}^{2n} b_i = \left(\sum_{i=1}^{n} x_i\right) + \left(\sum_{i=1}^{n} y_i\right) = \left(\sum_{i=1}^{n} x_i\right) + Z$$

ולפי החשבון הקודם אנו יודעים שבהסתברות גבוהה מתקיים ש

$$Z \approx np \pm \sqrt{np} = np \pm \frac{1}{\varepsilon}\sqrt{\log\left(\frac{1}{\delta}\right)}$$

ולכן התשובה שהשרת מחשב נכונה עד כדי טעות (בערך)

$$\frac{1}{\varepsilon}\sqrt{\log\left(\frac{1}{\delta}\right)}$$

המשימה הבאה: איך נוכל להפוך את הפרוטוקול הזה לפרוטוקול לחישוב הסטוגרמות?

<u>תזכורת – בעיית ההיסטוגמות:</u>
כל משתמש i מחזיק קלט $x_i \in D$.
נסמן את הדטהבייס המבוזר שלנו על ידי $X = (x_1, \dots, x_n)$
בנוסף, לכל איבר דומיין $d \in D$ נסמן

$$f_X(d) = |\{i : x_i = d\}|$$

המטרה שלנו: לכל איבר דומיין אנחנו רוצים לחשב הערכה $\hat{f}(d) \approx f_X(d)$

הגדרה: <u>שגיאת</u> החישוב שלנו היא

$$\max_{d \in D} \left|\hat{f}(d) - f_X(d)\right|$$

איך נוכל להשתמש בפתרון שלנו לספירת ביטים על מנת לחשב היסטוגרמות?

<u>פתרון ראשון:</u>
לכל איבר דומיין $d \in D$ נריץ את הפרוטוקול לסכימת ביטים (במקביל) על הדטהבייס המבוזר
$$X_d = (x_{d,1}, \dots, x_{d,n})$$
כאשר
$$x_{d,i} = \begin{cases} 0 & , \quad x_i \neq d \\ 1 & , \quad x_i = d \end{cases}$$

<u>שאלה:</u> מה זה אומר להריץ הרבה פרוטוקולים במקביל? הרי יש לנו רק ערוץ *shuffle* אחד...?

מבחינת פרטיות, הפרוטוקול הזה הוא טוב מאוד. אבחנת המפתח היא ששינויי של הדטה של משתמש בודד משפיע רק על 2 מתוך כל ההרצות (המקביליות) של הפרוטוקול לסכימה של ביטים ולכן פרטיות תתדרדר רק בפקטור 2 ולא בפקטור שתלויי בגודל הדומיין $|D|$.

אבל מבחינת הדיוק, השגיאה שלנו תגדל בפקטור $\log|D|$ מחסם האיחוד.

הביצועים של הפרוטוקול הפשוט הזה הם לא רעים... אמנם קיבלנו שגיאה שגדלה עם $\log|D|$, אבל השגיאה עדיין לא גדה כמו $\sqrt{n}$ כמו שהיה לנו במודל הלוקלי. עדיין, במודל הריכוזי אנחנו יודעים שהסטוגרמות אפשר לחשב ללא תלות בגודל הדומיין בכלל. האם גם כאן נוכל לקבל את זה?

<u>רעיון:</u> אם היינו יכולים "לתקן" את הפרוטוקול לסכימת ביטים כך שאם $X = (0,0,\dots,0)$ אז הפלט הוא 0 בהסתברות 1, אז זה היה פותר את הבעייה הנ"ל ולא היה לנו צורך בחם האיחוד על פני $|D|$.

<u>איך אפשר לעשות שינויי כזה לפרוטוקול?</u>

יהיה לנו יותר נח לעשות את השינוי עבור המקרה "ההפוך" בו אנחנו רוצים שאם $X = (1,1,\dots,1)$ אז הפלט יהיה n בהסתברות 1.

שאלה למחשבה: למה זה מספיק טוב?
תשובה (טקסט לבן):

נתקן פרוטוקול שבו כל משתמש ממיר את הקלט שלו ל $|D|$ ביטים כאשר הביט במקום שמתאים לקלט שלו הוא 0 וכל שאר הביטים הם 1. כלומר "הפוך" ממה שעשינו בפתרון הראשון. נריץ במקביל פרוטוקול כנ"ל לסכימה עבור כל איבר דומיין. אם עבור איבר מסויים נקבל מהפרוטוקול הסכימה הערכה r אז נחזיר את התשובה $n-r$ עבור האיבר הזה. מה נקבל? עבור איבר דומיין d שאף משתמש לא מחזיק אותו, הספירה שלו תהייה n ולכן פרוטוקול הסכימה יחזיר n בהסתברות 1, ואז אנחנו נחזיר 0 בהסתברות 1. לכן נצטרך "לספוג" את חסם האיחוד רק על פני איברי דומיין שמופיעים כקלט אצל המשתמשים, וכאלה יש לכל היותר n (שהוא יכול להיות הרבה יותר קטן מ $|D|$)

נשנה את האלגוריתם בצד של השרת באופן הבא:

אם $\sum_{i=1}^{2n} b_i \geq n$ אז נחזיר n. אחרת נחזיר $\sum_{i=1}^{2n} b_i - np$ כמו קודם.

נשים לב שאם $X = (1,1,\dots,1)$ אזי <u>תמיד</u> נחזיר תשובה n, כי יהיו לנו לפחות n הודעות מתוך ה-b_i-ים שהם 1 (הקלטים המקוריים).
בנוסף, במקרים אחרים השגיאה לא מתקלקלת יותר מ $\approx \frac{1}{\varepsilon^2}\log\left(\frac{1}{\delta}\right)$, כי אם $\sum_{i=1}^{n} x_i \ll n - \frac{1}{\varepsilon^2}\log\left(\frac{1}{\delta}\right)$ אזי ההסתברות שנקבל $\sum_{i=1}^{2n} b_i \geq n$ היא פיצפונת ולכן לא נשנה את התשובה ביחס למה שראינו קודם.