

Rapport de stage:
Arbitrages statistiques dans l'apprentissage automatique
confidentiel.

ALEXI CANESSE, L3 informatique fondamentale,
École Normale Supérieure de Lyon
Sous la supervision d'AURÉLIEN GARIVIER, Professeur,
UMPA et École Normale Supérieure de Lyon

June 22, 2022

1 Méthode des histogrammes

1.1 AboveThreshold

Répondre à de nombreuses requête est coûteux en confidentialité. Utiliser à l'algorithme naïf tel que le mécanisme de LAPLACE ne permet pas de répondre à de nombreuses requêtes avec une bonne précision tout en préservant un bon niveau de confidentialité (ε doit être petit). Dans certains cas nous ne sommes néanmoins pas intéressés par les réponses numériques, mais uniquement intéressés par le fait qu'une réponse dépasse ou non un seuil défini. Nous allons voir que `AboveThreshold` permet cela tout en ne payant que pour les requêtes qui dépassent le seuil.

```
1  AboveThreshold(database, queries, threshold, epsilon){
2      Assert("les requêtes sont toutes de sensibilité 1");
3      result = 0;
4      noisyThreshold = threshold + Lap(2/epsilon);
5      for(querie in queries){
6          nu = Lap(4/epsilon);
7          if(querie(D) + nu > noisyThreshold)
8              return result;
9          else
10             ++result;
11     }
12     return -1;
13 }
```

L'algorithme venant d'être décrit renvoie l'indice de la première requête à dépasser le seuil si une telle requête existe. C'est une version adaptée de l'algorithme initialement décrit par DWORK et ROTH dans [DR14, page 57]. Icelui a du sens d'un point de vue informatique mais rend le formalisme mathématiques compliqué (les auteurs eux-même tombent dans ce travers) et nous n'utiliseront pas les légers avantages de leur version.

Théorème 1.1:

Pour tout ensemble de requêtes $Q \in (\mathcal{X}^{(\mathbb{N})} \rightarrow \mathcal{T})^{\mathbb{N}}$ de sensibilité 1, tout seuil $T \in \mathbb{R}$, tout $\varepsilon > 0$, $M : x \in \mathcal{X}^{(\mathbb{N})} \mapsto \text{AboveThreshold}(x, Q, T, \text{epsilon})$ est ε -differentially private.

Remarque: La démonstration est une réécriture de celle du livre de référence [DR14, page 57]. Une réécriture était nécessaire car cette démonstration présente de nombreux points limites en terme de rigueur mathématiques et de détail pas suffisant sur certains points non triviaux.

Démonstration:

Soit $D, D' \in \mathcal{X}^{(\mathbb{N})}$ tels que $\|D - D'\| \leq 1$, $\{f_i\}_i = Q \in (\mathcal{X}^{(\mathbb{N})} \rightarrow \mathcal{T})^{\mathbb{N}}$ un ensemble de requêtes de sensibilité 1, $T \in \mathbb{R}$ un seuil, et $\varepsilon > 0$. On pose alors A la variable aléatoire $\text{AboveThreshold}(D, Q, T, \text{epsilon})$ et A' la variable aléatoire $\text{AboveThreshold}(D', Q, T, \text{epsilon})$.

Soit alors $k \in \mathbb{N}$. Montrons que $\mathbb{P}(A = k) \leq \mathbb{P}(A' = k)$. En reprenant les notations de l'algorithme [1.1], on fixe les éléments $(\nu_i)_{i < k}$ (qui suivent une loi de LAPLACE de paramètre $4/\varepsilon$).

On pose alors

$$\begin{cases} g_k &= \max_{i < k} \{f_i(D) + \nu_i\} \\ g'_k &= \max_{i < k} \{f_i(D') + \nu_i\} \end{cases}$$

Ces grandeurs représente la valeur plus grande comparée au seuil bruité avant l'indice k dans le cas de l'exécution sur D et de l'exécution sur D' . Les probabilités qui suivent seront présent sur

les deux variables aléatoires non fixées ν_k et \hat{T} qui est la valeur du seuil bruitée. On pose enfin, pour tout $i \in \mathbb{N}$,

$$\begin{cases} y_i &= f_i(D) \\ y'_i &= f_i(D') \end{cases}$$

On note alors que, en notant l_2 la densité de la loi de LAPLACE de paramètre $2/\varepsilon$ et l_4 celle de paramètre $4/\varepsilon$,

$$\begin{aligned} \mathbb{P}(A = k) &= \mathbb{P}(\hat{T} \in]g_k, y_k + \nu_k]) \\ &= \int_{\mathbb{R}} \mathbb{P}(\hat{T} \in]g_k, y_k + \nu]) l_4(\nu) d\nu \\ &= \int_{\mathbb{R}} \int_{g_k}^{y_k + \nu} l_2(t) l_4(\nu) dt d\nu \end{aligned}$$

On effectue alors un premier changement de variable affine

$$\hat{t} = t + g_k - g'_k$$

On obtient donc

$$\begin{aligned} \mathbb{P}(A = k) &= \int_{\mathbb{R}} \int_{g_k}^{y_k + \nu} l_2(\hat{t} - g_k + g'_k) l_4(\nu) dt d\nu \\ &= \int_{\mathbb{R}} \int_{g'_k}^{y_k + \nu - g_k + g'_k} l_2(\hat{t}) l_4(\nu) dt d\nu \end{aligned}$$

Il est alors temps de faire un second changement de variable affine

$$\hat{\nu} = \nu + g_k - g'_k + y'_k - y_k$$

Ainsi,

$$\begin{aligned} \mathbb{P}(A = k) &= \int_{\mathbb{R}} \int_{g'_k}^{y_k + \nu - g_k + g'_k} l_2(\hat{t}) l_4(\hat{\nu} - g_k + g'_k - y'_k + y_k) d\hat{t} d\hat{\nu} \\ &= \int_{\mathbb{R}} \int_{g'_k}^{y_k + \nu - g_k + g'_k + g_k - g'_k + y'_k - y_k} l_2(\hat{t}) l_4(\hat{\nu}) d\hat{t} d\hat{\nu} \\ &= \int_{\mathbb{R}} \int_{g'_k}^{y'_k + \nu} l_2(\hat{t}) l_4(\hat{\nu}) d\hat{t} d\hat{\nu} \end{aligned}$$

Par définition de l_2 et l_4 nous avons donc

$$\mathbb{P}(A = k) = \int_{\mathbb{R}} \int_{g'_k}^{y'_k + \nu} \exp\left(\frac{|\hat{t}| \varepsilon}{2}\right) \exp\left(\frac{|\hat{\nu}| \varepsilon}{4}\right) dt d\nu$$

L'inégalité triangulaire assure alors que

$$\begin{aligned} \mathbb{P}(A = k) &\leq \int_{\mathbb{R}} \int_{g'_k}^{y'_k + \nu} \exp\left(\frac{|\hat{t} - t| \varepsilon}{2}\right) \exp\left(\frac{|t| \varepsilon}{2}\right) \exp\left(\frac{|\hat{\nu} - \nu| \varepsilon}{4}\right) \exp\left(\frac{|\nu| \varepsilon}{4}\right) dt d\nu \\ &= \int_{\mathbb{R}} \int_{g'_k}^{y'_k + \nu} \exp\left(\frac{|g_k - g'_k| \varepsilon}{2}\right) \exp\left(\frac{|t| \varepsilon}{2}\right) \exp\left(\frac{|g_k - g'_k + y'_k - y_k| \varepsilon}{4}\right) \exp\left(\frac{|\nu| \varepsilon}{4}\right) dt d\nu \end{aligned}$$

Les requêtes étant de sensibilité 1, nous avons

$$\begin{cases} 2 & \geq |g_k - g'_k| + |y'_k - y_k| \geq |g_k - g'_k + y'_k - y_k| \\ 1 & = |g_k - g'_k| \end{cases}$$

Enfin, la croissance de l'intégrale assure que

$$\begin{aligned} \mathbb{P}(A = k) &\leq \int_{\mathbb{R}} \int_{g'_k}^{y'_k + \nu} \exp\left(\frac{\varepsilon}{2}\right) \exp\left(\frac{|t|\varepsilon}{2}\right) \exp\left(\frac{\varepsilon}{2}\right) \exp\left(\frac{|\nu|\varepsilon}{4}\right) dt d\nu \\ &= \exp\left(\frac{2\varepsilon}{2}\right) \int_{\mathbb{R}} \int_{g'_k}^{y'_k + \nu} \exp\left(\frac{|t|\varepsilon}{2}\right) \exp\left(\frac{|\nu|\varepsilon}{4}\right) dt d\nu \\ &= \exp(\varepsilon) \int_{\mathbb{R}} \int_{g'_k}^{y'_k + \nu} l_2(t) l_4(\nu) dt d\nu \\ &= \exp(\varepsilon) \int_{\mathbb{R}} \mathbb{P}(\hat{T} \in [g'_k, y'_k + \nu]) l_4(\nu) d\nu \\ &= \exp(\varepsilon) \mathbb{P}(\hat{T} \in [g'_k, y'_k + \nu_k]) \\ &= \exp(\varepsilon) \mathbb{P}(A' = k) \end{aligned}$$

1.2 La méthode des histogramme

La méthode des histogramme est une méthode que nous avons proposé durant ce stage. Il s'agit d'une instantiation particulière de **AboveThreshold** permettant de calculer l'ensemble des déciles (ou n'importe quel quantiles). Une transformation affine permet d'obtenir la réponse finale à partir de la réponse du mécanisme.

```

1  HistogramMethod(database, epsilon, steps, a, b){
2      /* composition theorem */
3      epsilon /= 9;
4
5      result = {};
6      for(d in {1 ... 9}){ /* which decile */
7          T = d*card(database)/10;
8          for(i in {1 ... steps}){
9              fi = x -> card({element in x | element < i*(b-a)/steps});
10             queries.push_back(fi);
11         }
12         T = d*card(database)/10;
13         result.push_back(AboveThreshold(database, queries, T, epsilon)
14                             *(b-a)/steps});
15     }
16     return result;
17 }
```

Les entrée a et b donnent une minoration et une majoration de l'ensemble des valeurs d'entrées. L'algorithme découpe alors l'intervalle $[a, b]$ en **steps** intervalles de même tailles. Pour chaque décile, l'entier renvoyé par **AboveThreshold** est l'indice de la première valeur à dépasser ce décile.



Figure 1: Le découpage pour $a = 0$, $b = 1$, **steps** = 4

Théorème 1.2:

HistogramMethod est ε -differentially private.

Démonstration: Les requêtes envoyées par l'algorithme à **AboveThreshold** sont bien de sensibilité 1. Chacun des neuf appels à cette fonction est donc $\varepsilon/9$ -differentially private. Le théorème de composition assure alors que **HistogramMethod** est ε -differentially private.

Maintenant que nous avons vu que cet algorithme est bien *differentially private*, nous allons essayer d'évaluer sa précision. Cela ne sera pas évident car la précision de l'algorithme dépend beaucoup du jeu de données en entrée.

Lemme 1.1: *AboveThreshold* est (α, β) -accurate

Pour tout $\beta \in]0, 1[$, tout $x \in \mathcal{X}^{(\mathbb{N})}$, tout $\{f_i\}_i = Q \in (\mathcal{X}^{(\mathbb{N})} \rightarrow \mathcal{T})^{\mathbb{N}}$, tout $\varepsilon > 0$, tout $T \in \mathbb{R}$, en posant $\alpha = 8(\log(k) + \log(2/\beta)) / \varepsilon$ et $k = \text{AboveThreshold}(x, Q, T, \text{epsilon})$, on a, en reprenant les notations de l'algorithme,

$$\mathbb{P}(\forall i < k \ f_i(x) + \nu_i < T + \alpha \wedge f_k(x) + \nu_k > T - \alpha) \geq 1 - \beta$$

Remarque: Ce lemme est due à [DR14, page 61]. Nous reprenons aussi la démonstration ici car la démonstration originale ne nous semble pas assez claire et trop bancal mathématiquement.

Démonstration: Reprenons les notations de l'énoncé. Montrons déjà qu'il suffit de démontrer que

$$\mathbb{P}\left(\max_{i \leq k} |\nu_i| + |T - \hat{T}| < \alpha\right) \geq 1 - \beta \quad (1)$$

où \hat{T} est le seuil bruité défini à la ligne 4 de l'algorithme [1.1]. Or, nous avons, en posant pour tout $i \leq k$, $y_i = f_i(x)$

$$y_k + \nu_k \geq \hat{T} \stackrel{\text{IT}}{\geq} T - |T - \hat{T}|$$

Mutatis mutandis

$$\forall i < k \quad y_i \leq \hat{T} + |\nu_i| \leq T + |T - \hat{T}| + |\nu_i|$$

Ainsi,

$$\mathbb{P}(\forall i < k \ f_i(x) + \nu_i < T + \alpha \wedge f_k(x) + \nu_k > T - \alpha) \geq 1 - \beta$$

Démontrons enfin (1)! La variable aléatoire $T - \hat{T}$ suit une loi de LAPLACE de paramètre $2/\varepsilon$. Ainsi,

$$\mathbb{P}\left(|T - \hat{T}| \geq \frac{\alpha}{2} = \frac{\alpha \varepsilon}{4} \frac{2}{\varepsilon}\right) = \exp\left(-\frac{\varepsilon \alpha}{4}\right) = \exp\left(-2\left(\log k + \log \frac{2}{\beta}\right)\right) \leq \exp\left(-2\left(\log \frac{2}{\beta}\right)\right) \leq \frac{\beta}{2}$$

De même,

$$\mathbb{P}\left(\max_i |\nu_i| \geq \frac{\alpha}{2}\right) \leq \sum_{j=1}^k \mathbb{P}\left(|\nu_j| \geq \frac{\alpha}{2}\right) = k \exp\left(-\frac{\alpha \varepsilon}{8}\right) = k \exp\left(-\log k - \log \frac{2}{\beta}\right) = \frac{k}{k} \frac{\beta}{2}$$

Enfin,

$$\begin{aligned} \mathbb{P}\left(\max_{i \leq k} |\nu_i| + |T - \hat{T}| < \alpha\right) &\geq \mathbb{P}\left(\max_{i \leq k} |\nu_i| < \frac{\alpha}{2} \wedge |T - \hat{T}| < \frac{\alpha}{2}\right) \\ &= 1 - \mathbb{P}\left(\max_{i \leq k} |\nu_i| \geq \frac{\alpha}{2} \cup |T - \hat{T}| \geq \frac{\alpha}{2}\right) \\ &\geq 1 - \mathbb{P}\left(\max_{i \leq k} |\nu_i| \geq \frac{\alpha}{2}\right) - \mathbb{P}\left(|T - \hat{T}| \geq \frac{\alpha}{2}\right) \\ &\geq 1 - \frac{\beta}{2} - \frac{\beta}{2} \end{aligned}$$

Finalement,

$$\mathbb{P} \left(\max_{i \leq k} |\nu_i| + |T - \hat{T}| < \alpha \right) \geq 1 - \beta$$

Ce qui démontre bien (1) et donc le lemme.

1.2.1 Analyse de complexité

La complexité de `AboveThreshold` est de l'ordre de la somme des complexité des requêtes sur le jeu de données d'entrée. En notant n la taille de la base de donnée, les requêtes envoyé à `AboveThreshold` par `HistogramMethod` sont toute de complexité linéaire en n . La variable `step` a aussi pour valeur le nombre de requêtes envoyées que l'on nommera k . L'algorithme a alors une complexité en $\mathcal{O}(nk)$.

1.2.2 Analyse de précision - le cas de la distribution uniforme

Nous allons évaluer la précision de l'algorithme à l'aide de l'erreur quadratique moyenne entre la valeur renvoyé par le programme et la valeur attendue. Il y a plusieurs manière de penser ce qu'est la valeur attendue: elle pourrait être la valeur des déciles de l'échantillons d'entrée. Néanmoins, elle peut tout aussi bien être l'ensemble des déciles de la loi. En effet, nous cherchons à répondre à des questions de statistique, l'entrée peut-être un simple échantillon "représentatif"; au quel cas nous sommes principalement intéressé par les réponses statistiques sur l'ensemble de la population et non juste sur notre échantillon.

Ces deux choix ont un réel sens. Nous avons d'abord essayé d'évaluer les performances de l'algorithme dans le premier cas. Les calculs était difficiles et menaient à des résultats difficilement exploitables. Nous avons donc choisi de réaliser les calculs sur la seconde option afin de pouvoir mener des calculs légèrement plus simples et ainsi avoir des résultats.

References

Dwork, Cynthia and Aaron Roth. “The Algorithmic Foundations of Differential Privacy”. In: *Foundations and Trends in Theoretical Computer Science* 9 (Aug. 2014), pp. 211–407. URL: <https://www.microsoft.com/en-us/research/publication/algorithmic-foundations-differential-privacy/>.