Revue littéraire - Smooth sensitivity and sampling data

Thomas

September 2017

1 Introduction

Mot-clef : Confidentialité différentielle.

Il s'agit de fournir de l'information agrégée sur les données via des fonctions f agissant sur ces données accompagnée d'un bruit additif dont l'amplitude dépend elle aussi des données considérées. D'autres études [1] ont déjà défini des méthodes afin de déterminer une telle fonction f qui permet de préserver la confidentialité des données individuelles. Le bruit ajouté à $f(x)$ dans ce cas dépend exclusivement de f (pas des données x), ce qui permet d'avoir des méthodes "sécurisée" pour beaucoup de méthodes et fonctions d'analyses :

- Moyennes
- Variances
- Histogrammes
- Tables de contingences
- Décomposition en valeur singulière

Ici l'objectif est bien de délivrer des fonctions f dont le bruit dépend des instances de données.

Difficulté : Garantir que l'amplitude du bruit ainsi formé ne puisse pas traduire ni révéler des informations individuelles sur le jeu de données.

Pour ce faire la calibration du bruit est réalisée à partir de la "*smooth sensitivity*" de la fonction f sur x, i.e une mesure de la variabilité de f au voisinage de x. Il faut donc pouvoir *calculer* (ou à défaut *approximer*) cette valeur. Les calculs résultants ne sont pas triviaux et pour certaines fonctions f s'avérer être *N-P complexe*.
Par ailleurs l'approximation obtenue doit être "*smooth*", i.e ne pas avoir de changements brusques de valeurs ou de comportement sur son ensemble de définition afin de garantir l'anonymisation des données.

Cette méthode permet, en outre d'augmenter le spectre d'application des méthodes de "*perturbation de sorties*", notamment aux algorithmes de clustering.

2 Notation et définitions

2.1 Notation

On considère une architecture *client-serveur*, où les clients envoient des requêtes (f) au serveur qui détient la base de données.
Le jeu de données est modélisé comme un vecteur $x \in D^n$, où x_i représente une information individuelle. Le serveur retourne $f(x)$ après l'ajout d'une composante de bruit que l'on souhaite minimale.

[1] "*Practical Privacy: The SuLQ framework*" A.Blum, C.Dwork, F.McSherry, K.Nissim

Pour une réquête paticulière f et une base de données x, le mécanisme d'accès aléatoire $\mathcal{A}$ défini la distribution de l'output noté $\mathcal{A}(x)$ tel que :

$$\mathcal{A}(x) = f(x) + Y$$

où Y est une variable aléatoire modélisant le bruit.

2.2 Définitions

2.2.1 Distance de Hamming

On définie la distance de Hamming entre deux bases de données comme étant le nombre d'entrées sur lesquels x et y diffèrent, i.e :

$$d(x, y) =\mid \{i : x_i \neq y_i\} \mid$$

On dit que deux bases de données sont voisinnes si elles ne diffèrent que d'une seule entrée, i.e :

$$d(x, y) = 1$$

Le mécanisme $\mathcal{A}$ est privé si deux bases de données voisines induisent deux distributions de l'output très "proches".

2.2.2 ε-indistinguable

Un algorithme aléatoire $\mathcal{A}$ est ε-indistinguable si $\forall x, y \in D^n$ tel que $d(x, y) = 1$ et $\forall S$, ensemble des outputs possibles on a :

$$\mathbb{P}[\mathcal{A}(x) \in S] \leq e^{\epsilon}\mathbb{P}[\mathcal{A}(y) \in S]$$

Autrement dit, cela signifie qu'aucun individu n'a un effet déterminant sur les statistiques issues du serveur puisque les distributions des outputs sont presque similaires que l'individu ait fourni des données exactes ou non.

2.2.3 Sensibilité globale

Pour $f : D^n \to \mathbb{R}^d$, la sensibilité globale de f est :

$$GS_f = \underset{x,y:d(x,y)=1}{\text{Max}} \parallel f(x) - f(y) \parallel$$

Proposition : $\forall f : D^n \to \mathbb{R}^d$, le mécanisme d'accès à la base de données $\mathcal{A}_f(x) = f(x) + (Y_1+, ..., Y_d)$, avec les $Y_i \sim \text{Lap}(GS_f, \epsilon)$[2] iid est ε-indistinguable.

2.2.4 Sensibilité locale

Pour $f : D^n \to \mathbb{R}^d$, la sensibilité locale de f en x est :

$$LS_f = \underset{y:d(x,y)=1}{\text{Max}} \parallel f(x) - f(y) \parallel$$

On remarque que $GS_f = \underset{x}{\text{Max}}\, LS_f(x)$

Dans le cadre de la sensibilité globale, l'amplitude du bruit dépend de GS_f et du paramètre d'anonymisation ε. Cependant sur certaines fonctions f, telles que le calcul de la médiane, cela engendre une grande quantité de bruit, ne reflétant donc pas l'insensibilité de ces fonctions aux valeurs individuelles.

[2] $\text{Lap}(\lambda) \Rightarrow h(y) = \frac{1}{2\lambda}e^{-|y|/\lambda}$, $\mathbb{E}[Y] = 0$, et $\text{Var}(Y)=2\lambda^2$

Néanmoins, on ne peut pas simplement définir l'amplitude du bruit proportionellement à $LS_f(x)$ car cette méthode serait trop naïve et ne respecterait pas la condition de ε-indistinguabilité et pourrait donc compromettre certaines informations de la base de données.
L'objectif est donc de pouvoir ajouter un bruit dépendant des données dont l'amplitude soit au pire égale à GS_f/ϵ et satisfaisant la condition de ε-indistinguabilité.
On définit alors une classe de bornes supérieures "*smooth*" S_f tel que l'on puisse ajouter un bruit proportionnellement à cette borne de manière sure. On définit également une fonction "*smooth*" optimale S_f^* tel que $S_f \geqslant S_f^*, \forall S_f$. On explictera son calcul ainsi que son approximation "*smooth*".

Calculer explicitement cette sensibilité "*smooth*" peut parfois être trivial pour certaines fonctions (par exemple les fonctions *max* ou *min*) mais nécessiter dans d'autres cas (par exemple déterminer le centroïde d'un cluster) de considérer les fonctions comme des boîtes noires au sein de la méthode *Sample & Aggregate*.

2.2.5 La méthode Sample & Aggregate

L'idée de cette méthode consiste à remplacer la fonction f par une fonction $\bar{f}$ que l'on peut considérer comme une version "*smooth*" de la fonction f initiale.
On commence par évaluer f sur un nombre défini d'échantillons aléatoires de la base de données x. Ces évaluations sont réalisées plusieurs fois puis combinées grâce à une fonction d'agrégation appelée *centre d'attention*. Le résultat obtenu, noté $\bar{f}$, est issu du modèle de sensibilité "*smooth*".

Les valeurs obtenues sont proches de celles voulues, $f(x)$, sur les jeux de données pour lesquels $f(x)$ est approximé en évaluant f sur ces échantillons aléatoires.

3 Facteur de bruit dépendant des données

Lorsque l'utilisateur souhaite obtenir $f(x)$, il envoie une requête f au seveur et reçoit :

$$f(x) + N(x).Z$$

où Z est une variable aléatoire connue de l'utilisateur symbolisant le bruit et le facteur $N(x)$ l'amplitude de celui-ci.

cf. l'article original *Smooth Sensitivity and Sampling in Private Data Analysis* par K. Nissim, S. Raskhodnikova et A. Smith ici.

References

[1] K. Nissim, S. Raskhodnikova, A. Smith, *Smooth Sensitivity and Sampling in Private Data Analysis.* , 2007.