modélisant les informations obtenues. Avec un nombre de requêtes suffisant, l'attaque est ravageuse, car on reconstruit toute la base de données : ce n'est plus la simple identification d'un individu unique, même s'il est gouverneur, mais par exemple l'ensemble d'un fichier de patients.

Schématiquement, la confidentialité différentielle consiste à cacher la contribution de tout individu possible au résultat d'une requête en ajoutant un bruit aléatoire aux données. En d'autres termes, le résultat d'une analyse sera sensiblement le même que l'on retire les données privées d'un individu particulier de l'ensemble des données ou qu'on les y inclut *(voir la figure ci-contre)*. De la sorte, tout individu devient « inrepérable ». L'amplitude du bruit ajouté dépend d'un paramètre quantifiant le « niveau de vie privée » (noté ε) et d'un paramètre de « sensibilité de la requête » quantifiant l'impact d'un individu sur les résultats de la requête. La valeur du paramètre ε est fixée par l'administrateur. Il s'agit d'un compromis entre, d'une part, la précision des résultats obtenus, en un mot l'utilité des données, et, d'autre part, la protection de la vie privée.

Ainsi, pour une requête sur la somme des salaires d'un nombre d'individus dans une base de données, par exemple l'ensemble du personnel d'une université, la confidentialité différentielle ajoute un nombre d'euros déterminé aléatoirement à partir d'une loi de statistiques (comme une distribution de Laplace) et proportionnel à la contribution d'un individu. De la sorte, impossible de connaître le salaire de tel ou tel membre du campus. Et ce sera d'autant plus le cas que la valeur du paramètre ε, celui qui contrôle le niveau de vie privée, sera élevée.

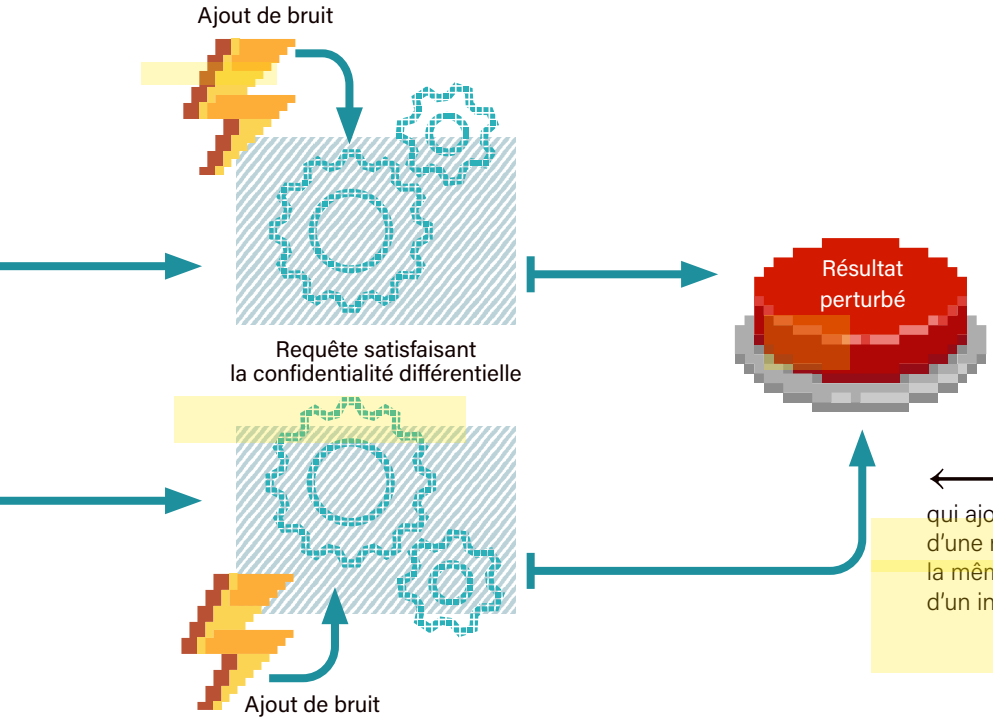
UN BUDGET DE VIE PRIVÉE

La confidentialité différentielle présente deux avantages cruciaux : elle est composable et résiste à un post-traitement, c'est-à-dire là où le bât blessait avec les modèles précédents. Toutefois, il a été montré au début des années 2000 que, même bruitées, on peut finir par reconstituer une base de données pour peu que l'on dispose de beaucoup d'informations. Pour contrer cet obstacle, et c'est un changement d'habitude à acquérir, la confidentialité différentielle impose une borne sur la quantité d'informations totale révélée par plusieurs calculs effectués sur les mêmes données. La conséquence en est que l'on peut voir dans le paramètre ε une sorte de budget de vie privée. À mesure que les requêtes se suivent, les valeurs de ε de chacune s'additionnent tandis que le « niveau de vie privée garanti » se dégrade. Aussi, un ε total, une « somme à ne pas dépasser », est-il fixé par l'administrateur.

La seule façon de poursuivre au-delà est d'utiliser une autre base de données. C'est possible, par exemple, en subdivisant un groupe initial de 100 000 individus en quatre parties égales. Cependant, cette méthode oblige à réfléchir à la construction des bases de données pour éviter les biais.

Si la confidentialité différentielle originale est aujourd'hui la référence, plus de 200 variantes ont depuis été proposées, chacune prenant en compte différents besoins, modèles d'attaquant ou types de données. En voici quelques-unes.

Une variante populaire est la (ε, δ)-confidentialité différentielle qui consiste à autoriser le mécanisme à moins bien cacher les contributions (proportionnellement à δ). Cet assouplissement de la définition originale autorise à introduire moins de bruit, et donc à faire des économies sur le budget de vie privée. Dans une seconde variante, la confidentialité différentielle personnalisée, chaque individu définit son exigence en matière de vie privée, par exemple par le biais d'un formulaire de consentement, plutôt que d'en imposer une globale. Cependant, l'hétérogénéité qui en résulte complique l'exploitation de la base de données.



Avec la confidentialité différentielle, qui ajoute du bruit de façon aléatoire au résultat d'une requête, une analyse sera sensiblement la même avec ou sans la contribution d'un individu quelconque.

D'autres variantes encore tiennent compte de la modification de la définition de la sensibilité, à savoir la façon dont on mesure l'impact d'un individu sur la sortie de l'algorithme d'anonymisation. La définition classique considère uniquement la présence ou l'absence de l'individu, mais on peut l'étendre à la modification d'un de ses attributs. Sur un exemple concret, ce type de variation protège non pas toutes les visites individuelles d'un site web par un individu, mais le fait qu'il ait déjà visité ou non le site. Enfin, si la confidentialité différentielle est typiquement adaptée à des données tabulaires (en tableau), elle peut se décliner à d'autres types de données à protéger, par exemple des données de type graphe dans le cadre d'un réseau social où les données représentent les individus et les liens qui les unissent.

UNE BIBLIOTHÈQUE DE BRIQUES

L'univers des algorithmes satisfaisant la confidentialité différentielle a en parallèle lui aussi fortement crû. Ces algorithmes dépendent du type de données à anonymiser (données sociodémographiques dans un tableau, séries temporelles, ensembles de nœuds et d'arêtes d'un réseau social…), de la catégorie d'informations à publier, du contexte de calcul (centralisé ou distribué). Malgré le large éventail de solutions disponibles, lorsqu'on ne trouve pas chaussure à son pied, il est nécessaire de concevoir un nouvel algorithme dont les garanties de confidentialité différentielle sont à prouver formellement. Cette tâche est loin d'être triviale, mais des outils ont été proposés. D'une part, certains offrent un cadre de démonstration clair listant les conditions formelles nécessaires pour qu'un algorithme satisfasse la confidentialité différentielle. À charge pour le concepteur de démontrer que son algorithme les remplit. D'autre part, des outils de vérification automatique établissent une preuve formelle, ou non, de la confidentialité différentielle.

Tous ces efforts sont grandement facilités quand l'algorithme en question peut être décomposé en briques élémentaires pour lesquelles la confidentialité différentielle est déjà assurée. Le concepteur pioche alors ce dont il a besoin dans des bibliothèques existantes. Pour ce faire, un écosystème d'outils a été développé, faisant le pont entre les travaux académiques et ceux de l'industrie. La bibliothèque Ektelo propose un ensemble générique de briques de bases pour développer des algorithmes obéissant à la confidentialité différentielle. IBM, Google et Microsoft distribuent aussi leurs propres bibliothèques. En 2020, Google a par ailleurs donné accès à une version de la bibliothèque d'apprentissage machine TensorFlow incorporant la confidentialité différentielle afin d'entraîner des modèles respectueux de la vie privée. Ces bibliothèques assurent la gestion du budget de vie privée, refusant l'exécution d'une requête si ce dernier est épuisé.