Privacy Preserving Outlier Detection

Kee Siong Ng

Privacy-Preserving Statistical Analysis: Motivating Problems

- Millionaire's problem: Alice and Bob are both millionaires and they want to know who is richer without the other person knowing their net worth.

- A school wants to investigate the relationship between people's IQ and their annual salaries. The school has its students' IQ scores but not their salaries. A survey company (say Glassdoor) has the students' salaries but cannot disclose them. How can the school and Glassdoor work together on the problem without divulging their data to each other.

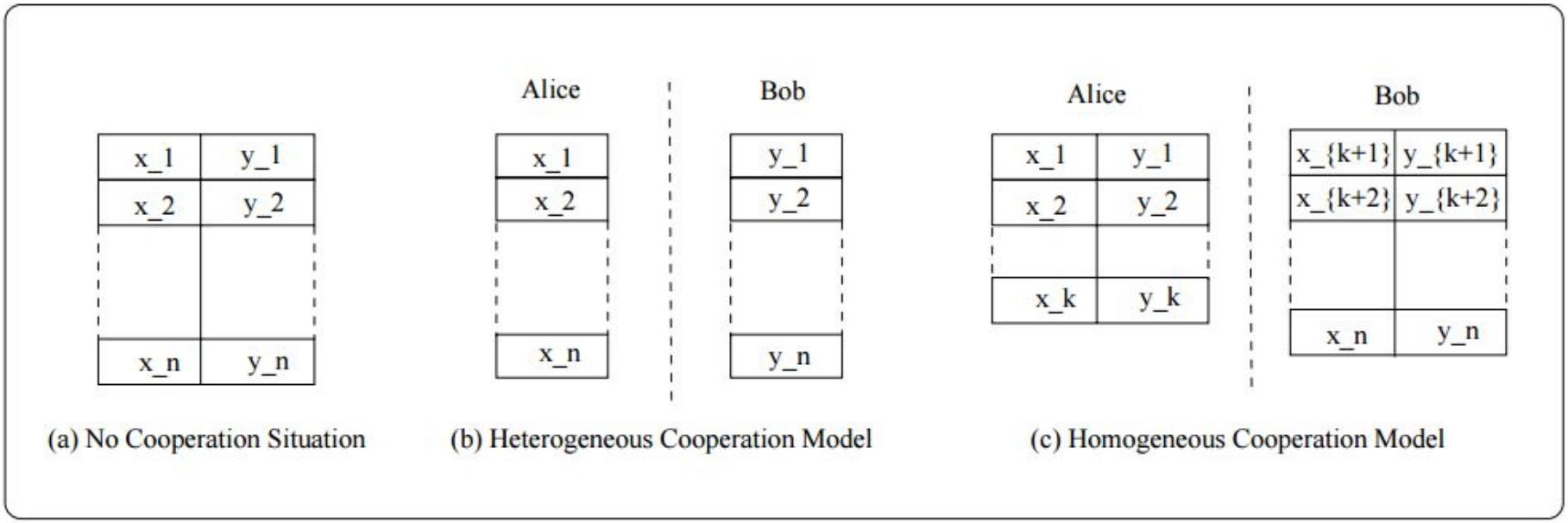
- Secure Multiparty Computation: In general, how do we enable a group of n parties each holding a piece of data d_i, i=1…n, to jointly compute an arbitrary function F(d_1, d_2,..., d_n) such that each party only ever saw his own data and the final result and nothing else.

Outlier Detection

- Hawkins: An outlier is an observation that deviates so much from other observations as to arouse suspicions that it was generated by a different mechanism.
- Definition (Knorr & Ng, 1998): An object O in a dataset T is a DB(p,D)-outlier if at least a fraction p of the objects in T lies greater than distance D from O.
- Simple definition that recovers some standard statistical outlier definitions
  - E.g. for a Gaussian distribution, outliers are observations that lie >= 3 std. deviations away from the mean is the same as DB(0.9988, 0.13σ)
- More sophisticated definitions are available (e.g. Local Outlier Factor and variations) but the DB(p,D) definition will do for now.

Two Models of Cooperation

The problem is to develop algorithms that can find DB(p,D)-outliers under these two models of cooperation / data-partitioning.



Building Blocks

RSA Public Key Encryption System

- Each person P has a public key = { e, n } and a private key = { d, n }.
- { e, n } is made publicly available and { d, n } is kept private.
- If another person wants to send a text M to P, he computes an encryption C of M using the formula

$$C := M^e \bmod n$$

- P can recover M by computing

$$M := C^d \bmod n = M^{ed} \bmod n$$

- The whole innovation behind the RSA scheme is a way to pick e, d, n such that
  - $M^{ed} = M \pmod n$
  - It is computationally infeasible to determine d given e and n

1-Out-Of-N Oblivious Transfer

- A protocol to allow one party, Alice, who has N data points $x_1, x_2, \ldots, x_N$ to transfer to another party Bob an x_i (of Bob's choice *i*) in such a way that Alice never know what *i* is and Bob never know what any of the other $x_j, j \neq i$ are.

- Intuitive solution:
  - Alice has messages $x_1, x_2, \ldots, x_N$
  - Bob goes to a store and buy N lockable letter boxes. He locks all the boxes and then throw away all the keys except the key for box i. (Bob's choice.)
  - Bob then sends all the locked boxes to Alice, who put x_1 in box 1, x_2 in box 2, etc, and then send all the boxes back.
  - Now Bob can open box i to get x_i and he doesn't get any of the other messages.

- Formal solution uses RSA public key encryption system.

1-Out-Of-N Oblivious Transfer

- A protocol to allow one party, Alice, who has N data points $x_1, x_2, \ldots, x_N$ to transfer to another party Bob an x_i (of Bob's choice *i*) in such a way that Alice never know what *i* is and Bob never know what any of the other $x_j, j \neq i$ are.

Alice				Bob		
Secret	Public	Calculus		Secret	Public	Calculus
m_0, m_1		Messages to be sent				
d	N, e	Generate RSA key pair and send public portion to Bob	$\Rightarrow$		N, e	Receive public key
	x_0, x_1	Generate two random messages	$\Rightarrow$		x_0, x_1	Receive random messages
				k, b		Choose $b \in \{0, 1\}$ and generate random k
	v		$\Leftarrow$		$v = (x_b + k^e) \mod N$	Compute the encryption of k, blind with x_b and send to Alice
$k_0 = (v - x_0)^d \mod N$ $k_1 = (v - x_1)^d \mod N$		One of these will equal k, but Alice does not know which.				
	$m'_0 = m_0 + k_0$ $m'_1 = m_1 + k_1$	Send both messages to Bob	$\Rightarrow$		m'_0, m'_1	Receive both messages
				$m_b = m'_b - k$		Bob decrypts the m'_b since he knows which x_b he selected earlier.

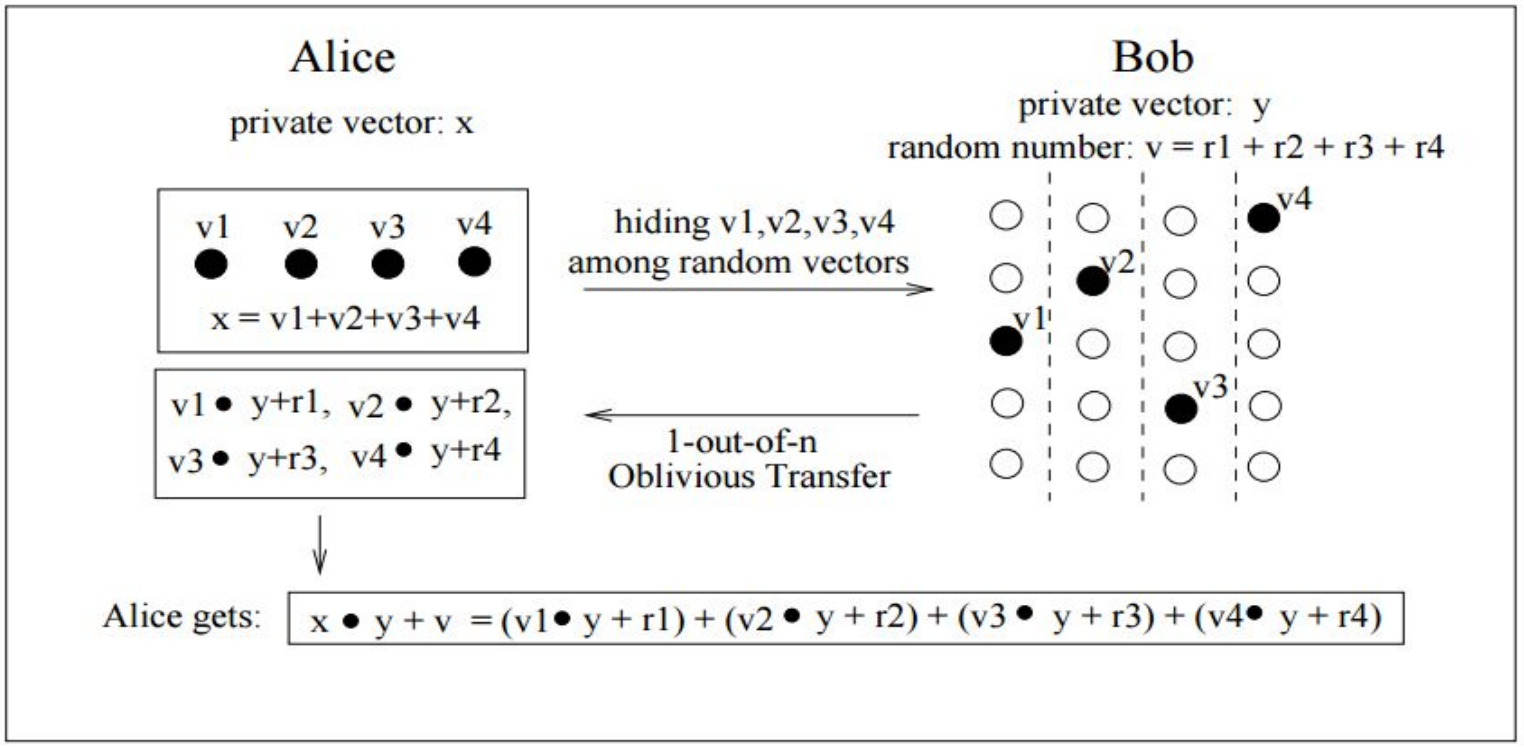
- The 1-out-of-2 scheme can be generalised to the general 1-out-of-N case
- Secure multi-party computation is complete with respect to 1-out-of-N oblivious transfer.

Private Scalar Product Computations

- In principle, we only need 1-out-of-N oblivious transfer. But scalar products is a better primitive to work with for statistical computations.
- Given $X = (x_1, \ldots, x_n)$ and $Y = (y_1, \ldots, y_n)$, we have $< X, Y > = \sum_k x_k y_k$
- Private Scalar Product Problem: Alice has a vector $X = (x_1, \ldots, x_n)$ and Bob has a vector $Y = (y_1, \ldots, y_n)$. Alice (but not Bob) is to get the result of $< X, Y > + v$, where v is a random number known to Bob only.
- The random number v is there for generality. Setting $v = 0$ gives us back the usual case.
- Naïve solution: Alice sends p vectors to Bob, only one of which is X (the others are chosen randomly). Bob then computes the scalar products between Y and each of the p vectors. At the end, Alice uses the 1-out-of-p oblivious transfer protocol to get back from Bob the result of $< X, Y >$.
- Problem: Bob has a 1-in-p chance of guessing the value of X.

Private Scalar Product Protocol

- Idea: Alice divides X into m random vectors $V_1, V_2, \ldots, V_m$ such that $V = \sum_i V_i$. Similarly, Bob divides v into m random numbers $r_1, r_2, \ldots, r_n$ such that $v = \sum_i r_i$. Use the naïve method m times to compute m intermediate results $< V_i, Y > + r_i$ and then sum up everything.



- Bob can guess the correct X value with probability $1/p^m$.

Secure Comparison

- There are two numbers a and b and the goal is to determine whether a > b without revealing the actual values of a and b.

- (Yao) Relatively easy-to-understand but computationally expensive solution
  - http://www.proproco.co.uk/million.html

- (Ioannidis et al) Involved, difficult-to-understand but computational efficient solution
  - https://en.wikipedia.org/wiki/Yao%27s_Millionaires%27_Problem

Privacy Preserving Outlier Detection

A Warm-up Exercise

Problem: Alice has a dataset $D_1 = (x_1, \ldots, x_n)$ and Bob has another dataset $D_2 = (y_1, \ldots, y_n)$. Alice and Bob want to find out the following:

- Correlation coefficient r between x and y.
- Regression line y = bx + c

Example 1: x is IQ and y is salary.

Example 2: x and y are both ML/TF risk scores as calculated by different agencies.

Computing Correlation Coefficient

- Alice has $D_1 = (x_1, \ldots, x_n)$ and Bob has $D_2 = (y_1, \ldots, y_n)$.

- Correlation r between x and y is given by

$$r = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_i (x_i - \bar{x})^2 \sum_i (y_i - \bar{y})^2}}$$

$$= < \left(\frac{x_1 - \bar{x}}{u}, \ldots, \frac{x_n - \bar{x}}{u}\right), \left(\frac{y_1 - \bar{y}}{v}, \ldots, \frac{y_n - \bar{y}}{v}\right) >$$

where u $= \sqrt{\sum_i (x_i - \bar{x})^2}$ and v $= \sqrt{\sum_i (x_i - \bar{x})^2}$

- We can compute r using the private scalar product protocol.

A Warm-up Exercise

Problem: Alice has a dataset $D_1 = (x_1, \ldots, x_n)$ and Bob has another dataset $D_2 = (y_1, \ldots, y_n)$. Alice and Bob want to find out the following:

- Correlation coefficient r between x and y.
- Regression line y = bx + c (LEFT AS AN EXERCISE)

Example 1: x is IQ and y is salary.

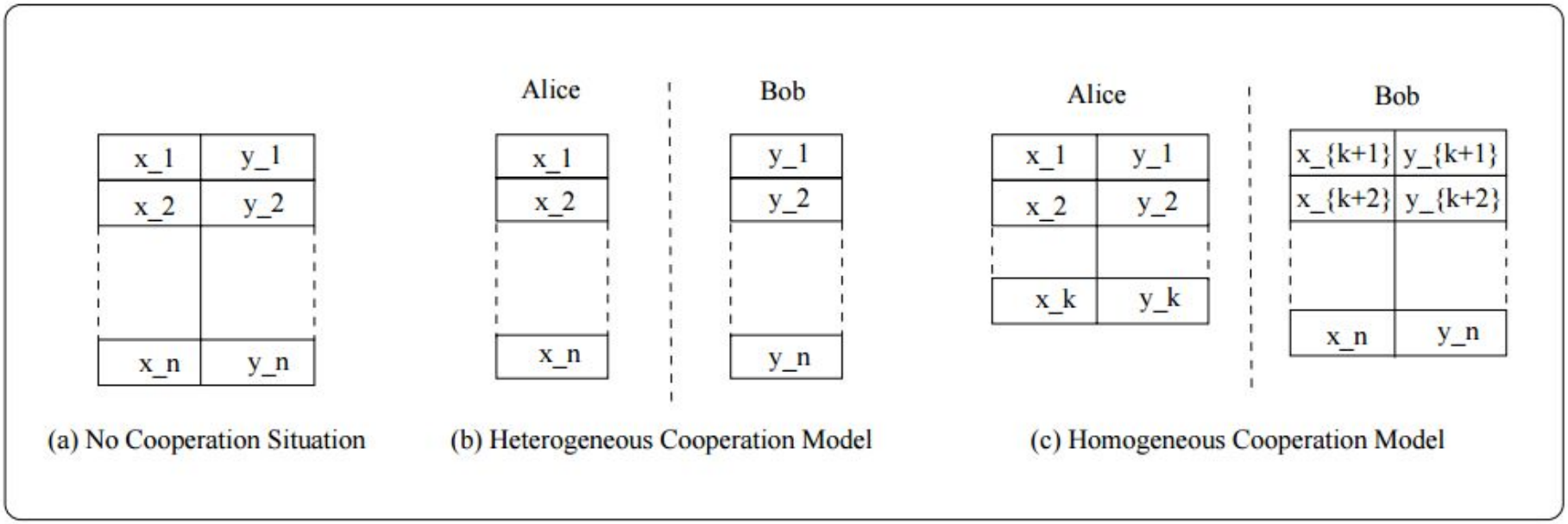
Example 2: x and y are both ML/TF risk scores as calculated by different agencies.

Two Models of Cooperation

Default DB(p,D)-outlier detection algorithm:

```
For each X
    Num := 0
    For each Y≠X
        Increment Num iff dist(X,Y) > D.
    Output X as an outlier if Num > p * populationSize
```

The problem is to develop algorithms that can find DB(p,D)-outliers under these two models of cooperation / data-partitioning.



Computing DB(p,D)-Outliers: Homogenous Cooperation Case

- Definition (Knorr & Ng, 1998): An object O in a dataset T is a DB(p,D)-outlier if at least a fraction p of the objects in T lies greater than distance D from O.
- Key trick: Let $dist\ (.,.)$ be the distance function. Instead of computing $dist(X,Y)$, compute

$$dist^2(X,Y) = \sum_i (x_i - y_i)^2 = \sum_i x_i^2 + \sum_i y_i^2 - \sum_i 2x_i y_i$$

- Note that when X and Y sit with different parties, the distance calculation requires only a secure scalar product operation.
- Naïve algorithm:
  - For each X
    - Num := 0
    - For each Y≠X
      - If X and Y are with the same party, increment Num iff dist(X,Y) > D.
      - Else if X and Y are with different parties, the parties use secure scalar product to compute dist(X,Y) and increment Num iff dist(X,Y) > D.
    - Output X as an outlier if Num > p * populationSize
- The actual algorithm is slightly more complicated because every intermediate results are randomly split and shared between parties to ensure complete privacy.

Computing DB(p,D)-Outliers: Heterogeneous Cooperation Case

	Alice	Bob
O1	(x1,x2)	(x3,x4)
O2	(y1,y2)	(y3,y4)

Idea: Since $dist^2(O_1, O_2) = dist^2((x_1, x_2), (y_1, y_2)) + dist^2((x_3, x_4), (y_3, y_4))$
the problem comes down to computing the addition of two local distances securely.

Protocol:

- Alice generates a random r, then sends $M := r + dist^2((x_1, x_2), (y_1, y_2))$ to Bob.
- Bob then computes $P := M + dist^2((x_3, x_4), (y_3, y_4)) - D^2$
- Alice and Bob then uses secure comparison to check whether P > r.

We have $dist^2(O_1, O_2) > D^2$ iff P > r.

Proof: Note that $P = r + dist^2(O_1, O_2) - D^2$. In case when $dist^2(O_1, O_2) > D^2$, we have P = r + (a positive number), which implies P > r.

Key Takeaways

- Privacy-preserving (PP) statistical computations are important in the context where we have private data distributed across multiple locations.
- There are now practical PP algorithms for a number of well-studied problems, including outlier detection.
- Once we have a good handle on the primitives (oblivious transfer, scalar products, comparisons), many of these PP algorithms are not hard to understand and implement.
- The foundational technologies behind PP algorithms, including the important Secure Multi-party Computation problem, is now increasingly being used in conjunction with Blockchain technique to produce potentially disruptive technologies like the Enigma system.

References

- Du & Atallah, Privacy-Preserving Cooperative Statistical Analysis, 2001.
- Vaidya & Clifton, Privacy-Preserving Outlier Detection, 2004.