Microsoft Research

Each year Microsoft Research hosts hundreds of influential speakers from around the world including leading scientists, renowned experts in technology, book authors, and leading academics, and makes videos of these lectures freely available.

2016 © Microsoft Corporation. All rights reserved.
Differential Privacy for Growing Databases

Rachel Cummings, Georgia Tech ISyE

joint work with Sara Krehbiel, Kevin Lai, and Uthaipon (Tao) Tantitongpipat

MSR Redmond – May 30, 2018
Old-fashioned data science:

One-shot analysis of a static database
Modern data science:

Continuous analysis of a growing database
Modern data science:

Continuous analysis of a growing database

1. Need data analysis tools for growing databases
Modern data science:

Continuous analysis of a growing, **highly sensitive** database

1. Need data analysis tools for growing databases
2. Need formal privacy guarantees
Differential privacy [DMNS ‘06]

Bound the “maximum amount” that one person’s data can change the output of a computation

An algorithm $M : T^n \to R$ is $\varepsilon$-differentially private if $\forall$ neighboring $x, x' \in T^n$ and $\forall S \subseteq R$,

$$P[M(x) \in S] \leq e^\varepsilon P[M(x') \in S]$$

- $S$ as set of “bad outcomes”
- Worst-case guarantee
Post-processing:
If $M(x)$ is $\epsilon$-differentially private, and $A$ is any function, then $A(M(x))$ is $\epsilon$-differentially private.

“No adversary can break the privacy guarantee”

Composition:
If algorithms $M_i$ are $\epsilon_i$-differentially private for $i = 1, \ldots, k$, then the composition $M(x) \equiv (M_1(x), \ldots, M_k(x))$ is $\sum_{i=1}^{k} \epsilon_i$-differentially private.

“The epsilons add up”
Differential privacy [DMNS ‘06]

Bound the “maximum amount” that one person’s data can change the output of a computation

An algorithm $M : T^n \rightarrow R$ is $\epsilon$-differentially private if $\forall$ neighboring $x, x' \in T^n$ and $\forall S \subseteq R$,

$$P[M(x) \in S] \leq e^{\epsilon} P[M(x') \in S]$$

- $S$ as set of “bad outcomes”
- Worst-case guarantee
Post-processing:
If $M(x)$ is $\epsilon$-differentially private, and $A$ is any function, then $A(M(x))$ is $\epsilon$-differentially private.

“No adversary can break the privacy guarantee”

Composition:
If algorithms $M_i$ are $\epsilon_i$-differentially private for $i = 1, \ldots, k$, then the composition $M(x) \equiv (M_1(x), \ldots, M_k(x))$ is $\sum_{i=1}^{k} \epsilon_i$-differentially private.

“The epsilons add up”
Differential privacy [DMNS ‘06]

- Move smoothly between informational extremes
- Robust to post-processing and adaptive composition
- Diverse algorithmic toolkit for variety of computational settings
- Used in practice by Apple, Google, Microsoft, and U.S. Census

Nearly all DP algorithms operate on a static database
Outline of results

1. Adaptive linear queries for growing databases
   • Background on Private Multiplicative Weights
   • What goes wrong with growing databases
   • Our algorithm and results

2. General transformations of static algorithms to growing setting
   • Black box reduction to static setting
   • Application to synthetic data generation
   • Extensions & application to ERM

3. Open questions
Part 1: Private Multiplicative Weights for growing databases

PMW privately answers exponentially many adaptively chosen linear queries. Does it extend to growing databases?
• Database $x$ stored as histogram: $x = \langle x_1, ..., x_N \rangle$ where $x_i$ is fraction of database entries that are of type $i$. Data universe has size $N$.

• Linear queries: $f(x) = \langle f, x \rangle$ where each $f_i \in [0,1]$.

Query sensitivity:
The sensitivity of real-valued query $f$ is:

$$\Delta f = \max_{D, D' \text{neighbors}} |f(D) - f(D')|.$$ 

Linear queries have sensitivity $\Delta f = \frac{1}{n}$ for database size $n$.

$(\alpha, \beta)$-accuracy:
An algorithm $M$ is $(\alpha, \beta)$-accurate w.r.t. query class $F$ if for any database $x$, $|f(x) - f(M(x))| \leq \alpha$ for all $f \in F$ w.p. $1 - \beta$. 
Private Multiplicative Weights [HR ‘10]

- Answers exponentially many (in size $n$ of database $x$) adaptive linear queries
- Maintains public histogram $y$, which reflects current estimate of static database
- When new query $f$ arrives:
  - If $f(x) \approx f(y)$, classify $f$ as EASY and output $f(y)$
  - Else, classify $f$ as HARD, output $f(x) + \text{Lap} \left( \frac{\Delta f}{\epsilon} \right)$, and perform Multiplicative Weights update on $y$
- Significant privacy loss only incurred on HARD queries
What I have learned = **public database** $y$
Overpredict $f_1$: Loss = $f_1$

$f_1(y) > f_1(x)$
public database $y$

Overpredict $f_1$: Loss = $f_1$

Underpredict $f_1$: Loss = $1 - f_1$

$f_1(y) < f_1(x)$
Theorem [HR ‘10]: PMW(x, F, \epsilon, \alpha, \beta, n) is \epsilon-differentially private and (\alpha, \beta)-accurate for k adaptively chosen queries F for

\[
\alpha \geq \left( \frac{\log N \log(k)}{\epsilon n} \right)^{1/3}.
\]

Proof relies on potential argument, using potential function:

\[
RE(x||y) = \sum_{i=1}^{N} x_i \log \frac{x_i}{y_i}.
\]

Need to show: \(RE(x||y)\) starts low, decreases quickly, and is bounded below
1. $RE(x||y)$ starts at $\log N$
2. $RE(x||y)$ decreases by at least $\frac{\alpha^2}{4}$ with each HARD query
3. $RE(x||y)$ bounded below by 0

PMW has at most $\frac{4 \log N}{\alpha^2}$ hard queries

$\varepsilon$ loss is small
Private Multiplicative Weights [HR ‘10]

Theorem [HR ‘10]: PMW(x, F, ε, α, β, n) is ε-differentially private and (α, β)-accurate for k adaptively chosen queries F for

\[ \alpha \geq \left( \frac{\log N \log \left( \frac{k}{\beta} \right)}{\epsilon n} \right)^{1/3}. \]

Proof relies on potential argument, using potential function:

\[ RE(x||y) = \sum_{i=1}^{N} x_i \log \frac{x_i}{y_i}. \]

Need to show: \( RE(x||y) \) starts low, decreases quickly, and is bounded below.
1. $RE(x||y)$ starts at $\log N$
2. $RE(x||y)$ decreases by at least $\frac{\alpha^2}{4}$ with each HARD query
3. $RE(x||y)$ bounded below by 0

PMW has at most $\frac{4 \log N}{\alpha^2}$ hard queries

$\epsilon$ loss is small

$RE(x||y)$

$\log N$

HARD queries

$\frac{\alpha^2}{4}$

$t$
What if new data arrive?

1. $RE(x||y)$ starts at $\log N$
2. $RE(x||y)$ decreases by at least $\frac{\alpha^2}{4}$ with each HARD query
3. $RE(x||y)$ bounded below by 0

Database size $n$

$RE(x||y)$

$\log N$

HARD queries

$\frac{\alpha^2}{4}$
What if new data arrive?

1. \( RE(x||y) \) starts at \( \log N \)
2. \( RE(x||y) \) decreases by at least \( \frac{\alpha^2}{4} \) with each HARD query
3. \( RE(x||y) \) bounded below by 0

\[
RE(x||y) \\
\log N
\]

Database size \( n \)

Database size \( n+k \)

HARD queries

\[
\frac{\alpha^2}{4}
\]
What if new data arrive?

1. $RE(x||y)$ starts at $\log N$ - $RE(x||y)$ changes
2. $RE(x||y)$ decreases by at least $\frac{\alpha^2}{4}$ with each HARD query - YES
3. $RE(x||y)$ bounded below by 0 - YES

Database size $n$  

Database size $n+k$  

$RE(x||y)$

$log N$
$RE(x||y)$ can increase by $\frac{\log N}{n \alpha}$ with each new entry.

Database size $n$

Database size $2n$

$RE(x||y)$

$\log N$

$\frac{\alpha^2}{4}$

$\frac{\log N}{\alpha}$

$t$ (time axis)
1. $RE(x||y)$ starts at $\log N$

2. $RE(x||y)$ decreases by at least $\frac{\alpha^2}{4}$ with each HARD query

3. $RE(x||y)$ bounded below by 0

PMW has at most $\frac{4 \log N}{\alpha^2}$ hard queries

$\epsilon$ loss is small
$RE(x||y)$ can increase by $\frac{\log N}{n \alpha}$ with each new entry.
Our fix: Reweight public database $y$ with uniform distribution when new data arrive:

$$y'_i = \frac{n}{n+1} y_i + \frac{1}{n+1} \frac{1}{N} \quad \forall i$$

Now $RE(x||y)$ increase is only $\frac{\log N}{n}$ per new entry.
Main result: PMWG

Theorem: PMWG($X, F, \varepsilon, \alpha, \beta, n$) is $\varepsilon$-differentially private and ($\alpha, \beta$)-accurate for query stream $F$ that respects query budget $\Theta(k \exp \sqrt{t})$ at each time $t$, for $\alpha \geq C \left( \frac{\log(Nn) \log\left(\frac{kn}{\beta}\right)}{n\varepsilon} \right)^{1/3}$.

Asymptotically no loss in accuracy relative to static setting!

In the paper: extend to ($\varepsilon, \delta$)-differential privacy
Private Multiplicative Weights [HR ‘10]

Theorem [HR ‘10]: PMW(x, F, \epsilon, \alpha, \beta, n) is \epsilon-differentially private and \((\alpha, \beta)\)-accurate for \(k\) adaptively chosen queries \(F\) for

\[
\alpha \geq \left( \frac{\log N \log(k/\beta)}{\epsilon n} \right)^{1/3}.
\]

Proof relies on potential argument, using potential function:

\[
RE(x||y) = \sum_{i=1}^{N} x_i \log \frac{x_i}{y_i}.
\]

Need to show: \(RE(x||y)\) starts low, decreases quickly, and is bounded below
Private Multiplicative Weights [HR '10]

- Answers exponentially many (in size $n$ of database $x$) adaptive linear queries
- Maintains public histogram $y$, which reflects current estimate of static database
- When new query $f$ arrives:
  - If $f(x) \approx f(y)$, classify $f$ as EASY and output $f(y)$
  - Else, classify $f$ as HARD, output $f(x) + \text{Lap}(\frac{\Delta f}{\epsilon})$, and perform Multiplicative Weights update on $y$
- Significant privacy loss only incurred on HARD queries
Outline of results

1. Adaptive linear queries for growing databases
   - Background on Private Multiplicative Weights
   - What goes wrong with growing databases
   - Our algorithm and results

2. General transformations of static algorithms to growing setting
   - Black box reduction to static setting
   - Application to synthetic data generation
   - Extensions & application to ERM

3. Open questions
Main result: PMWG

**Theorem:** PMWG(\(X, F, \epsilon, \alpha, \beta, n\)) is \(\epsilon\)-differentially private and \((\alpha, \beta)\)-accurate for query stream \(F\) that respects query budget \(\Theta(k \exp \sqrt{t})\) at each time \(t\), for \(\alpha \geq C \left( \frac{\log(Nn) \log(kn/\beta)}{n\epsilon} \right)^{1/3}\).

Asymptotically no loss in accuracy relative to static setting!

In the paper: extend to \((\epsilon, \delta)\)-differential privacy
Private Multiplicative Weights [HR ‘10]

Theorem [HR ‘10]: PMW(\(x, F, \epsilon, \alpha, \beta, n\)) is \(\epsilon\)-differentially private and \((\alpha, \beta)\)-accurate for \(k\) adaptively chosen queries \(F\) for

\[
\alpha \geq \left(\frac{\log N \log(k)}{\epsilon n} \right)^{1/3}.
\]

Proof relies on potential argument, using potential function:

\[
RE(x||y) = \sum_{i=1}^{N} x_i \log \frac{x_i}{y_i}.
\]

Need to show: \(RE(x||y)\) starts low, decreases quickly, and is bounded below.
Main result: PMWG

**Theorem:** PMWG\((X, F, \epsilon, \alpha, \beta, n)\) is \(\epsilon\)-differentially private and \((\alpha, \beta)\)-accurate for query stream \(F\) that respects query budget \(\Theta(k \exp \sqrt{t})\) at each time \(t\), for \(\alpha \geq C \left( \frac{\log(Nn) \log(kn/\beta)}{n\epsilon} \right)^{1/3}\).

Asymptotically no loss in accuracy relative to static setting!

In the paper: extend to \((\epsilon, \delta)\)-differential privacy
Outline of results

1. **Adaptive linear queries for growing databases**
   - Background on Private Multiplicative Weights
   - What goes wrong with growing databases
   - Our algorithm and results

2. **General transformations of static algorithms to growing setting**
   - Black box reduction to static setting
   - Application to synthetic data generation
   - Extensions & application to ERM

3. **Open questions**
Part 2: General reduction from dynamic setting to static setting

Assume a private and accurate algorithm in the static setting. How can this be used as a subroutine for solving the same problem in the dynamic setting?
Ideal (easy) data growth

Run private and accurate algorithm on current database
Realistic data growth

When to run algorithm?
Accuracy, revisited

\((\alpha, \beta)\)-accuracy:
An algorithm \(M\) is \((\alpha, \beta)\)-accurate w.r.t. query class \(F\) if for any database \(x\), \(|f(x) - f(M(x))| \leq \alpha\) for all \(f \in F\) w.p. \(1 - \beta\).

\((p, g)\)-black box:
An algorithm \(M(x, \epsilon, \alpha, \beta, n)\) is a \((p, g)\)-black box for a query class \(F\) if it is \(\epsilon\)-differentially private and \((\alpha, \beta)\)-accurate w.r.t. \(F\) for

\[
\alpha = g \left( \frac{\log \frac{1}{\beta}}{\epsilon n} \right)^p.
\]
Application: SmallDB [BLR ’08]

- Differentially private algorithm for synthetic data generation
  - Takes in database $x$ of size $n$ from data universe of size $N$ and query class $F$
  - Outputs database $y$ of size $\log |F| / \alpha^2$
  - Samples $y$ with exponential bias towards databases that closely approximate $x$ on all queries in $F$

**Theorem [BLR ’08]:** SmallDB$(x, F, \alpha, \beta, \epsilon)$ is $\epsilon$-differentially private and $(\alpha, \beta)$-accurate for

$$\alpha \geq C \left( \frac{\log N \log |F| + \log 1/\beta}{\epsilon n} \right)^{1/3}$$

- SmallDB is a $(\frac{1}{3}, (64 \log N \log |F|)^{1/3})$-black box.
SmallDB for growing databases (SmallDBG)

- Run SmallDB($x_{t_i}, F, \alpha, \beta_i, \epsilon_i$) at times $t_i = (1 + \gamma)^i n$ for some $\gamma < 1$, to produce output stream $\{y_i\}$
- Use $y_i$ to answer all queries that arrive between $t_i$ and $t_{i+1}$
- Tune $\beta_i, \epsilon_i$ in each epoch as a function of database size

Theorem: SmallDBG($x, F, \alpha, \beta, \epsilon, n$) is $\epsilon$-differentially private and $(\alpha, \beta)$-accurate for

$$\alpha \geq C \left( \frac{\log N \log |F| + \log 1/\beta}{\epsilon n} \right)^{1/5}$$

Power is 1/3 in static setting, 1/5 in growing setting
Main result: BBScheduler

Running \((p, g)\)-black box \(M(x_{t_i}, \epsilon_i, \alpha_i, \beta_i, n)\) on current database at times \(t_i = (1 + \gamma)^{i+1} n\) for \(i = 1, 2, \ldots\) using parameters:

- \(\gamma = g^{1/(2p+1)} \left( \frac{\log \frac{1}{\beta}}{\epsilon n} \right)^{p/(2p+1)}\)
- \(\epsilon_i = \frac{\gamma^2 (i+1)}{(1+\gamma)^{i+2}} \epsilon\)
- \(\alpha_i = g \left( \frac{\log \frac{1}{\beta}}{\epsilon_i (1+\gamma)^i n} \right)^p\)
- \(\beta_i = \left( \frac{\beta}{1+\beta} \right)^{i+1}\)

is \((\epsilon, 0)\)-differentially private and \((\alpha, \beta)\)-accurate for

\[\alpha \geq g^{2p+1} \left( \frac{\log \frac{1}{\beta}}{\epsilon n} \right)^{p/(2p+1)}\]
Main result: BBScheduler

Running \((p, g)\)-black box \(M(x_{t_i}, \epsilon_i, \alpha_i, \beta_i, n)\) on current database at times \(t_i = (1 + \gamma)^{i+1}n\) for \(i = 1, 2, \ldots\) using parameters:

\[
\gamma = g^{1/(2p+1)} \left( \frac{\log_\beta}{\epsilon n} \right)^{p/(2p+1)}
\]

\[
\epsilon_i = \frac{\gamma^2(i+1)}{(1+\gamma)^{i+2}} \epsilon
\]

\[
\alpha_i = g \left( \frac{\log_\beta}{\epsilon_i(1+\gamma)^i n} \right)^p
\]

\[
\beta_i = \left( \frac{\beta}{1+\beta} \right)^{i+1}
\]

is \((\epsilon, 0)\)-differentially private and \((\alpha, \beta)\)-accurate for

\[
\alpha \geq g^{2p+1} \left( \frac{\log_\beta}{\epsilon n} \right)^{p/(2p+1)}
\]

Independent of data growth, Base of exponent in database size

\(\sum \epsilon_i = \epsilon\) for composition and overall privacy guarantee

Accuracy guarantee of static algorithm at current time

\(\sum \beta_i = \beta\) for union bound over failure prob and overall accuracy guarantee

Accuracy loss of \(1/(2p + 1)\) in power, relative to static setting
Extensions (in the paper)

- $(\epsilon, \delta)$-differentially private version of BBScheduler
  - Improved accuracy at cost of weaker privacy guarantee
- $(\alpha_t, \beta)$-accuracy, where $\alpha_t \to 0$ as $t \to \infty$
  - Accuracy guarantee that improves as data accumulate
  - Better than fixed accuracy guarantee ($\alpha_t < \alpha$) once database roughly squares in size ($t \gg n^2$)
- Application to Empirical Risk Minimization
  - Larger training sample $\implies$ lower empirical risk
Empirical Risk Minimization

**Theorem:** ERMG is \((\varepsilon, \delta)\)-differentially private and for \(d\)-dimensional samples, w.p. \(1 - \beta\) outputs a classifier \(y_t\) at every time \(t\) with excess empirical risk:

\[
\hat{R}_t \leq \frac{d \log \frac{1}{\beta} \sqrt{\log 1/\delta}}{\varepsilon t^{1/2}}
\]
Extensions (in the paper)

- $(\epsilon, \delta)$-differentially private version of BB Scheduler
  - Improved accuracy at cost of weaker privacy guarantee
- $(\alpha_t, \beta)$-accuracy, where $\alpha_t \rightarrow 0$ as $t \rightarrow \infty$
  - Accuracy guarantee that improves as data accumulate
  - Better than fixed accuracy guarantee ($\alpha_t < \alpha$) once database roughly squares in size ($t \gg n^2$)
- Application to Empirical Risk Minimization
  - Larger training sample $\Rightarrow$ lower empirical risk
Main result: BBScheduler

Running \((p, g)\)-black box \(M(x_{t_i}, \epsilon_i, \alpha_i, \beta_i, n)\) on current database at times \(t_i = (1 + \gamma)^{i+1}n\) for \(i = 1, 2, ...\) using parameters:

- \(\gamma = g^{1/(2p+1)} \left( \frac{\log_\beta 1}{\epsilon n} \right)^{p/(2p+1)}\)
- \(\epsilon_i = \frac{\gamma^2(i+1)}{(1+\gamma)^{i+2}}\epsilon\)
- \(\alpha_i = g \left( \frac{\log_\beta 1}{\epsilon_i (1+\gamma)^i n} \right)^p\)
- \(\beta_i = \left( \frac{\beta}{1+\beta} \right)^{i+1}\)

is \((\epsilon, 0)\)-differentially private and \((\alpha, \beta)\)-accurate for

\[\alpha \geq g^{2p+1} \left( \frac{\log_\beta 1}{\epsilon n} \right)^{p/(2p+1)}\]

- Independent of data growth, Base of exponent in database size
- \(\sum \epsilon_i = \epsilon\) for composition and overall privacy guarantee
- Accuracy guarantee of static algorithm at current time
- \(\sum \beta_i = \beta\) for union bound over failure prob and overall accuracy guarantee
- Accuracy loss of \(1/(2p+1)\) in power, relative to static setting
Extensions (in the paper)

- $(\epsilon, \delta)$-differentially private version of BBScheduler
  - Improved accuracy at cost of weaker privacy guarantee
- $(\alpha_t, \beta)$-accuracy, where $\alpha_t \to 0$ as $t \to \infty$
  - Accuracy guarantee that improves as data accumulate
  - Better than fixed accuracy guarantee ($\alpha_t < \alpha$) once database roughly squares in size ($t \gg n^2$)
- Application to Empirical Risk Minimization
  - Larger training sample $\Rightarrow$ lower empirical risk
Empirical Risk Minimization

**Theorem:** ERMG is \((\epsilon, \delta')\)-differentially private and for \(d\)-dimensional samples, w.p. \(1 - \beta\) outputs a classifier \(y_t\) at every time \(t\) with excess empirical risk:

\[
\hat{R}_t \leq \frac{d \log \frac{1}{\beta} \sqrt{\log 1/\delta}}{\epsilon t^{1/2}}
\]

**Theorem [BST ’14]:** Static ERM is \(\epsilon\)-differentially private and for a static database of size \(t\) with \(d\)-dimensional samples, outputs a classifier \(y\) that w.p. \(1 - \beta\) has excess empirical risk:

\[
\hat{R} \leq \frac{d \log \frac{1}{\beta}}{\epsilon t}
\]

**In the paper:** stronger bounds with more assumptions on loss function.
Outline of results

1. Adaptive linear queries for growing databases
   • Background on Private Multiplicative Weights
   • What goes wrong with growing databases
   • Our algorithm and results

2. General transformations of static algorithms to growing setting
   • Black box reduction to static setting
   • Application to synthetic data generation
   • Extensions & application to ERM

3. Open questions
Open questions for analysis of dynamic data

1. Improved accuracy guarantees with assumptions on data generation
   - Data points all sampled iid from fixed distribution
   - Data points sampled from distribution that changes smoothly over time

2. Detecting changes in data
   - Identifying when new data is dramatically different from old data [CKMTZ ’18]
   - A/B testing

3. Other types of dynamic data
   - Changing/deleting datapoints
   - “Horizontal” data growth
Open questions for analysis of dynamic data

1. Improved accuracy guarantees with assumptions on data generation
   - Data points all sampled iid from fixed distribution
   - Data points sampled from distribution that changes smoothly over time
2. Detecting changes in data
   - Identifying when new data is dramatically different from old data [CKMTZ ‘18]
   - A/B testing
3. Other types of dynamic data
   - Changing/deleting datapoints
   - “Horizontal” data growth