An investigation into membership inference attacks on location data

Andrew J. Young

Imperial College London
Supervisors: Yves-Alexandre De Montjoye, Andrea Gadotti

June 24, 2019

Problem statement

Can we use the valuable data that is collected by online services without giving up peoples private information?

Privacy for location data

- Every point can be sensitive, and may or may not be unique to an individual.
- Classical anonymization methods (k-anonymization, pseudonymization) fall short.
- **The classical notions of anonymization need to be redefined and measured differently.**

Aggregate statistics for location data

Definition: *Aggregate statistic*

$$\overline{x} = \frac{1}{n}\sum_{i=1}^{n} x_i \qquad (1)$$

- Don't *directly* reveal information about any one person.
- Have been put into practice by companies who release data about their customers [**?**].
- But ...

Pyrgelis et al. 2017 [?]

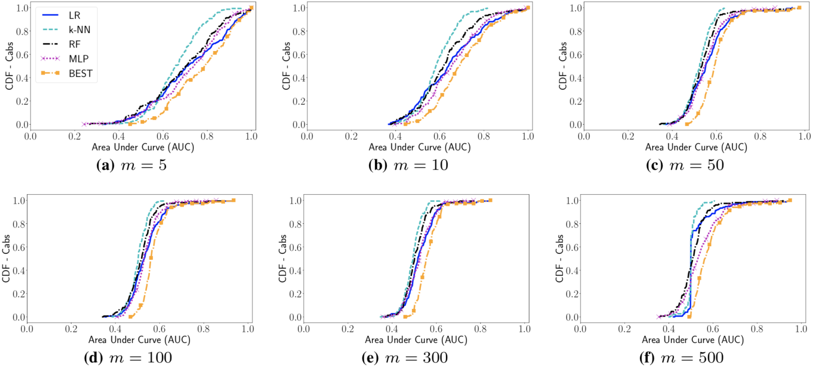


Figure: Different locations than released prior against the SFC data set: Adversary's performance for different group sizes.

Contributions

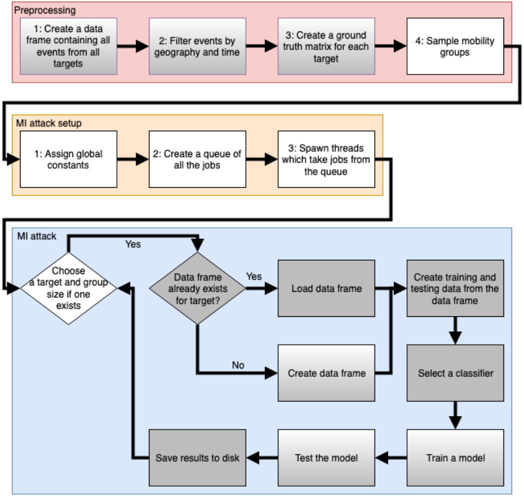
We look at . . .

1. Aggregation size vs accuracy of MI;
2. Defenses;
3. Attack profile;
4. 75 % run time reduction;
5. No need for geo-coordinates.

Data sets

	San Francisco cabs (SFC)	Call detail records (CDR)
Users	534	153,997
Span	1 month	1 month
Geography	Downtown San Fran	1 city
ROIs	100	220
Reports	`(Lat, Long)`	Antenna ID

Attack overview



Attack overview

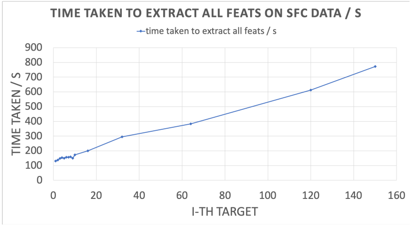
Name	**Adversary knowledge**
SUBSET OF LOCATIONS	Real locations of a subset of individuals
SAME LOCATIONS	Participation in groups also used in T_I.
DIFFERENT LOCATIONS	Participation in groups not used in T_I.

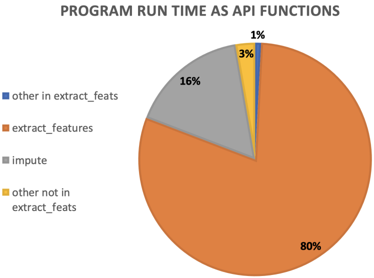
Modifications

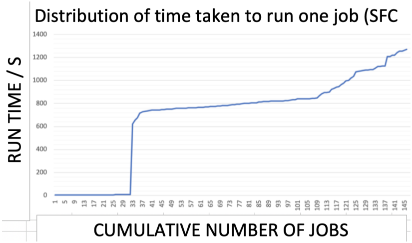
Why profile?

- Understand the resource constraints;
- Understand how the attack works;
- Identify potential problems in advance.

Profile of the original attack







Optimizations

What did we learn?

1. The program is CPU bound;
2. Data frame operations are expensive;
3. The program scales poorly for big data sets.

Optimizations

How can we improve the attack?

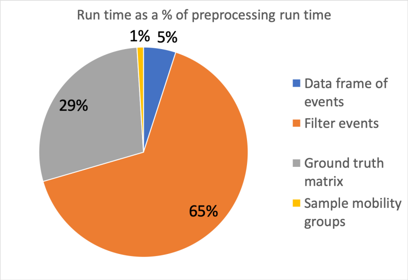
1. Introducing multithreading (attack) and multiprocessing (preproc);
2. Choosing the optimal number of threads and processes;
3. Optimizing data frame operations.

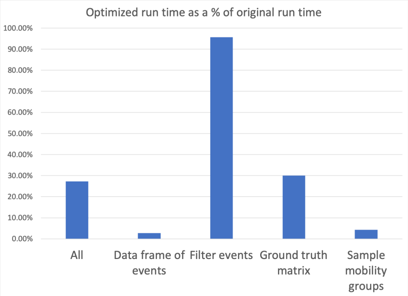
Evaluating our optimizations

Either:

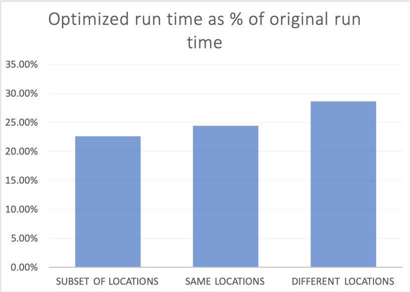
1. The attack runs "fast enough" to collect enough good results;
2. The program bottleneck is due to hardware, not badly optimized code.

Evaluating our optimizations





Evaluating our optimizations



Functions which take up the longest run time:

- `threading.wait`
- `thread.lock.acquire`

$\Longrightarrow$ the program is now IO bound.

Extending the input space

Pyrgelis et al.'s attack code needs geo-coordinates in order to work.
But the CDR data set uses anonymized locations.

Therefore:

1. Make config files specific to the data set;
2. Branch in preprocessing for data frame operations;

Measuring success

1. Type checking;
2. Unit testing;
3. Dummy data;
4. Real life.

Reimplementation

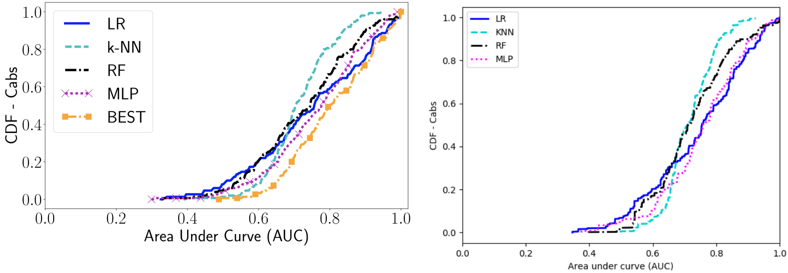


Figure: SAME LOCATIONS on SFC data with $m = 5$

Reimplementation

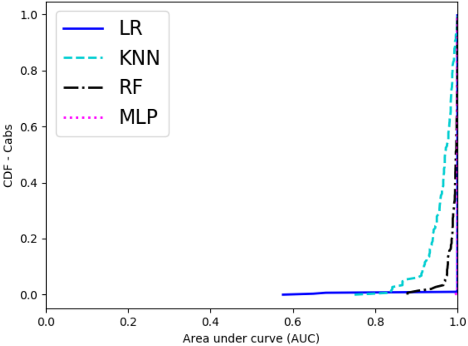
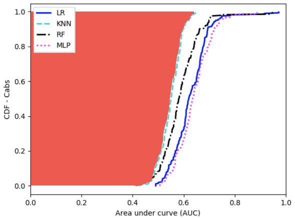


Figure: SUBSET OF LOCATIONS on SFC data with $m = 50$

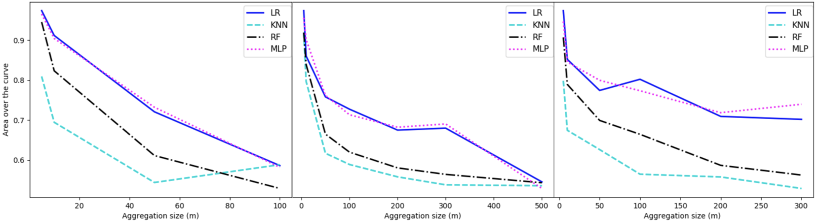
Measuring attack power

Definition: *Area over the curve (AOC)*

$$\mathrm{AOC}(x) = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{2}$$



Attack power and aggregation size



(a) SUBSET OF LOCATIONS (b) SAME LOCATIONS (c) DIFFERENT LOCATIONS

Figure: AOC of different classifiers against aggregation size for attacks on SFC data.

Attack power and aggregation size

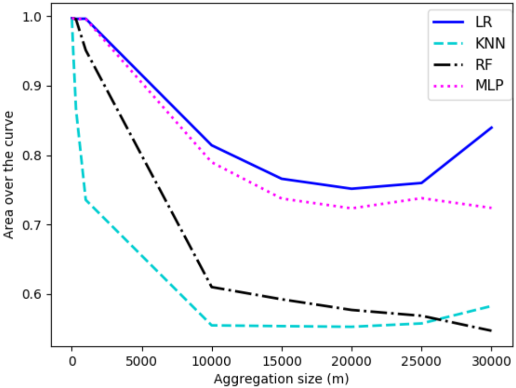
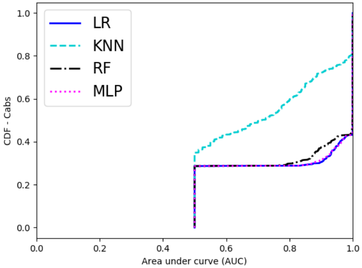
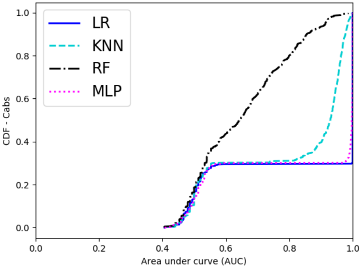


Figure: AOC of different classifiers against aggregation size for attacks on CDR data.

Defenses (SFC)



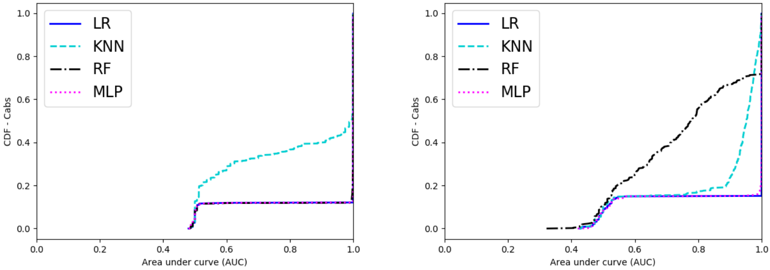
(a) $m = 5$



(b) $m = 100$

Figure: Modal ROI reports (WINNER TAKES ALL) per epoch.

Defenses (SFC)

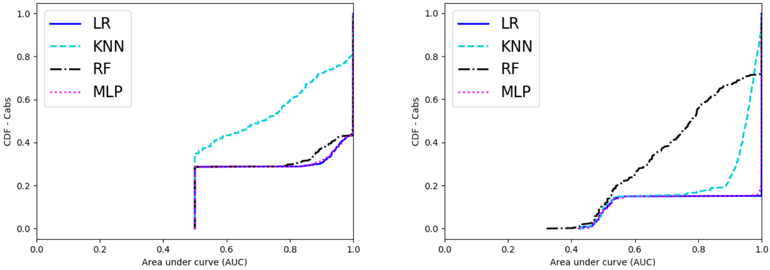


(a) $m = 5$

(b) $m = 100$

Figure: Random ROI reports per epoch.

Defenses (SFC)



(a) $m = 5$ (b) $m = 100$

Figure: Comparison between modal and random ROI reports.

Defenses (SFC)

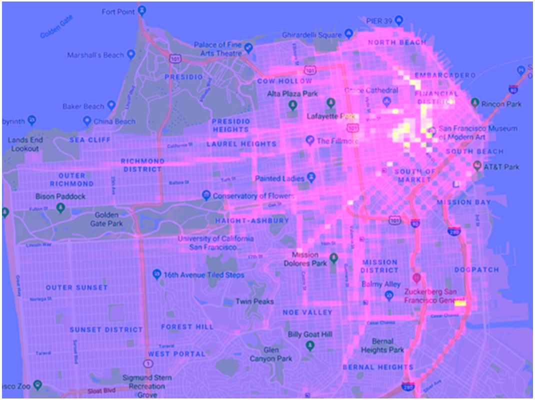


Figure: Aggregate taxi traffic on the San Francisco area between May 19 and June 10, 2008 [?]

Defenses (SFC)

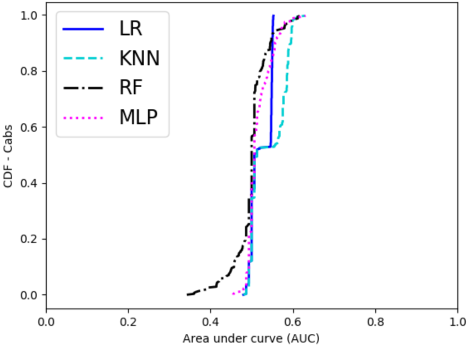


Figure: $N(0, 10^2)$, $m = 5$

Defenses (SFC)

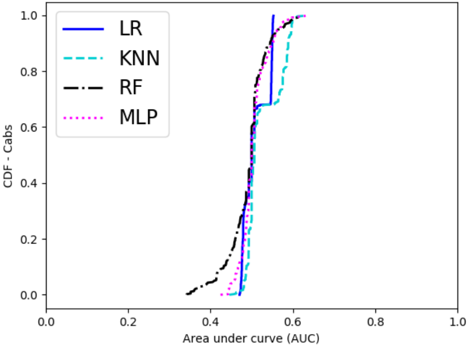


Figure: $Lap(0, \Delta/\epsilon), \epsilon = 0.1, m = 5$

Defenses (SFC)

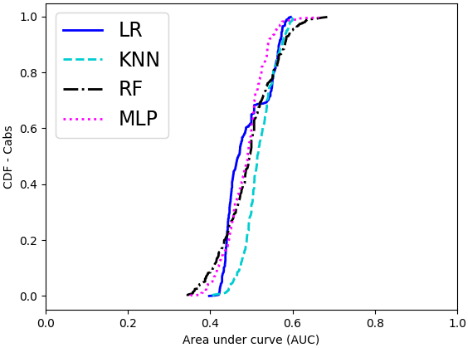
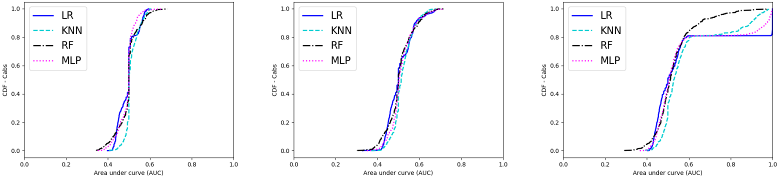


Figure: $Lap(0, 1/\epsilon), \epsilon = 0.1, m = 5$

Defenses (SFC)



(a) $m = 5$ (b) $m = 10$ (c) $m = 100$

Figure: Low count suppression with threshold $= 10$.

Defenses (SFC)

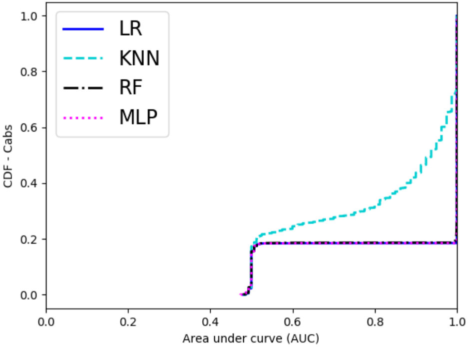
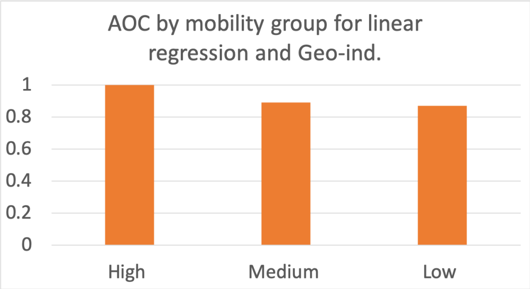


Figure: Geo-indistinguishablity, $m = 5, \epsilon = 0.1$

Defenses (SFC)



Contributions

We looked at . . .

1. Aggregation size vs accuracy of MI;
2. Defenses;
3. Attack profile;
4. 75 % run time reduction;
5. No need for geo-coordinates.

Self-assessment

Strengths:

- Run time
- Scalability of preprocessing
- Extents of analysis

Limitations:

- Extensibility using defenses;
- Analysis across data sets;
- Assumptions in the formalization of membership inference

Future work

- Do longer observation periods increase the power of MI?
- What factors are behind the trend of AOC vs aggregation size?
- Membership inference on internet browsing data;
- Improving the scalability of Python's APIs.

References

Apostolos Pyrgelis, Carmela Troncoso, and Emiliano De Cristofaro.
Knock Knock, Who's There? Membership Inference on Aggregate Location Data.
(Ndss), 2017.

Raluca Ada Popa, Andrew J. Blumberg, Hari Balakrishnan, and Frank H. Li.
Privacy and accountability for location-based aggregate statistics.
In *Proceedings of the 18th ACM Conference on Computer and Communications Security*, CCS 2011, pages 653–666, New York, NY, USA, 2011. ACM.

Telefonica SA.
Smart steps, 2017.