Knock Knock, Who's There? Membership Inference on Aggregate Location Data

NDSS 2018

Apostolos Pyrgelis, Carmela Troncoso, Emiliano De Cristofaro

Web Security & Privacy Lab

KAIST

Introduction

- Analysts use aggregate location statistics to
  - Calculate average speed along a road
  - Generate live traffic maps
  - Estimate the number of people in a restaurant, predict availability, waiting times.
  - Uber Movement, Telefonica Smart Steps, factual.com
- Apple (iOS, 3rd-party app devs) collect aggregate statistics about :
  - Emojis/Deep links/Locations

Introduction

- What it does: **Membership inference attacks on location data**
    - Is the location data of a target user part of aggregated data?
- Why it does:
    - Release of **privacy sensitive information**
    - Aggregate location statistics violate the privacy of individuals that are part of the aggregates

Web Security & Privacy Lab

KAIST

Importance

- Membership inference can be used for:
  - Profiling (Alzheimer patients)
  - Localization (sensitive locations)
  - Providers can evaluate the quality of privacy protection on the aggregates before releasing them
- Regulators can verify misuse of data
  - Aggregate location data has been released without permission

Distinguishability Game (DG)

- **“Rules” (assumptions)**
  - The number of users in some Region of Interest (ROIs) are released periodically within a given time interval
  - Adversary has prior knowledge about the users
- **Challenger**
  - Generates location aggregates over various user groups
- **Adversary**
  - Relies on this data
  - Tries to infer whether data of a particular user is included in the aggregates

Adversarial Prior Knowledge

- **Subset of Locations:**
  - Observation and Inference coincide
  - Adversary knows the real locations of a subset of users, including the target user, during the inference period
    - Telecommunications service provider getting locations from cell towers
    - Mobile app provider collecting location data
- **Participation in Past Groups:**
  - Adversary knows aggregates computed during an observation period, disjoint from the inference period
  - May or may not include the user

Increasing Difficulty

Adversarial Prior Knowledge

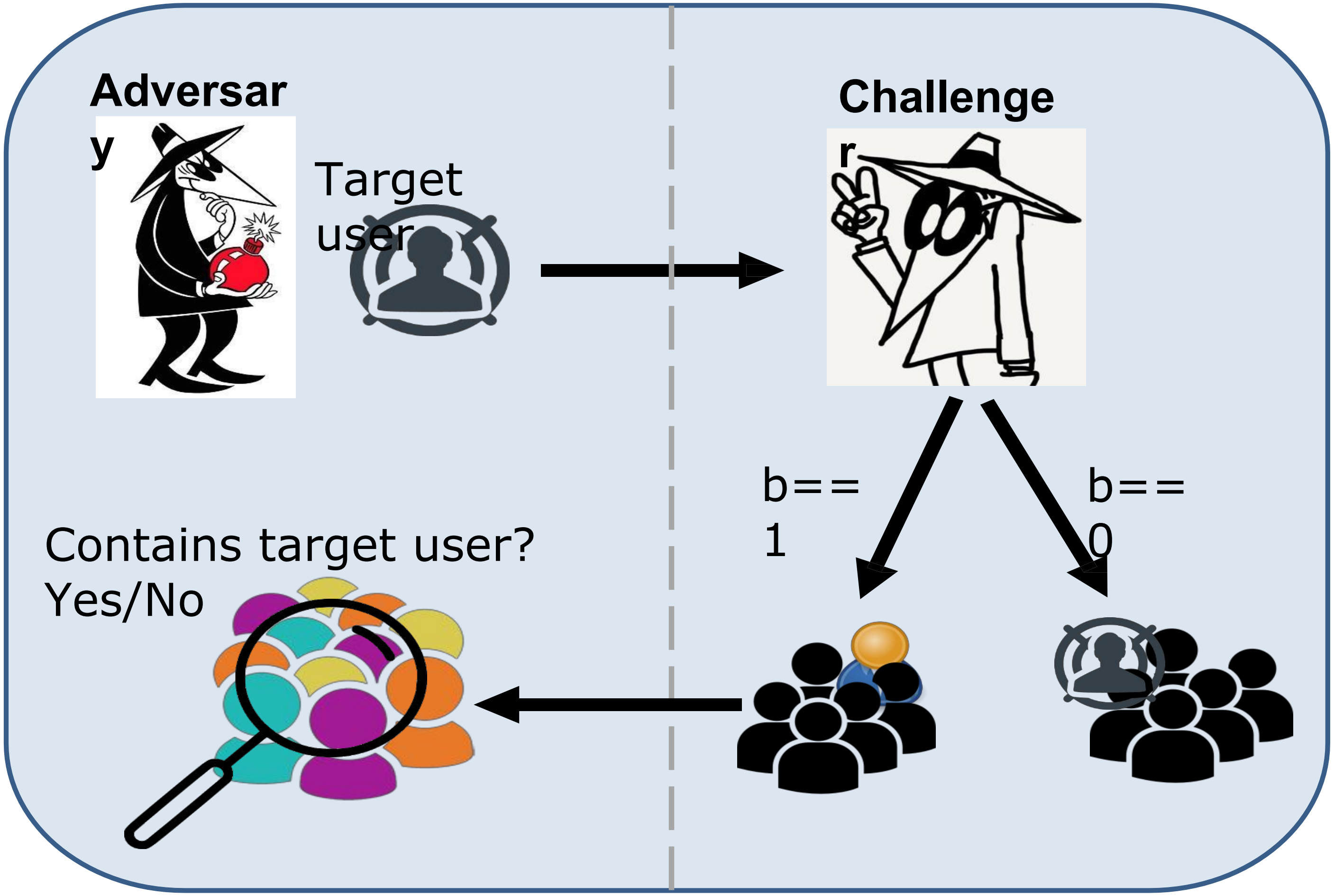
- **Same Groups as Released**: (Continuous data release over stable groups)
  - Adversary knows the target user's participation in past groups
  - The same groups are used to compute the aggregates during the inference period
- **Different Groups than Released**: (Continuous data release over dynamic user group)
  - Adversary knows the user's participation in past groups
  - These groups are not used to compute aggregates released in the inference period

Increasing Difficulty

Web Security & Privacy Lab

KAIST

Distinguishability Game (DG)



Distinguishing Function

- Is a target user part of the aggregates? (in/out)
- **Binary classification task**
- Utilize **supervised machine learning** classifier trained on the prior knowledge
- **Inputs**:
  - **Target user**
  - **“Challenge” = aggregate location time-series of users**
  - **Size of aggregation group (m)**
  - **Considered time period**
  - **Prior knowledge**

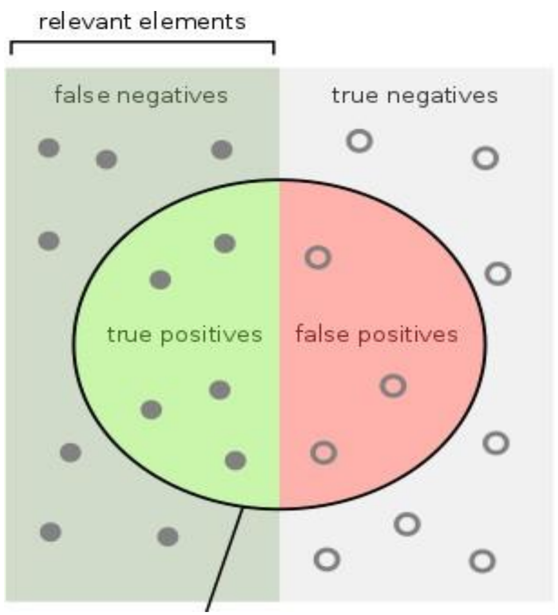
Privacy Metric

- **Privacy Loss**:
  - Advantage in winning the DG over a random guess
  - Area Under Curve (AUC) score to measure the classifier's performance

$$\mathrm{PL} = \begin{cases} \frac{\mathrm{AUC}-0.5}{0.5} & \text{if } \mathrm{AUC} > 0.5 \\ 0 & \text{otherwise} \end{cases}$$

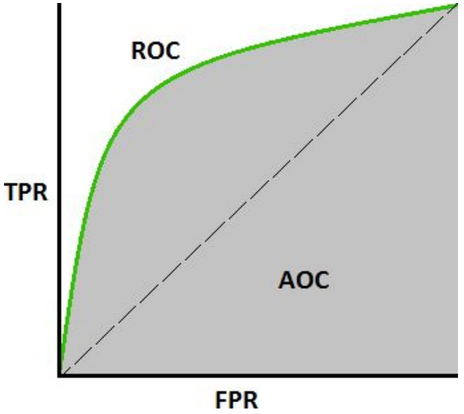
Privacy Metric

- **Area Under Curve (AUC)** Score:
- Count the adversaries guesses b’, compare to ground truth b
  - True Positive (TP) when b=0 and b’=0
  - True Negative (TN) when b=1 and b’=1
  - False Positive (FP) when b=1 and b=0
  - False Negative (FN) when b=0 and b’=1



Privacy Metric

- **True Positive** and **False Positive Rate**
  - TPR=TP/(TP+FN)
  - FPR=FP/(FP+TN)
- **Receiver Operating Characteristic (ROC)** curve
  - Represents the TPR and FPR obtained at various discrimination classification thresholds
- **Area Under Curve (AUC)**:
  - Captures a classifier's overall performance in the distinguishability game



Datasets

- **Transport For London (TFL): (sparse, regular)**
  - Trips made by passengers on the TFL network (March 1 - Sunday, March 28, 2010)
  - 60M trips - 4M unique oyster cards - 582 stations (regions of interest - ROIs)
  - Sample the top 10K passengers ids per total # of trips □ on average, 728 ± 16 ROIs in total
  - one hour granularity: Top 10K passengers are in 115 ± 21 out of the 672 timeslots (28 days)
  - When a user does not report any station at a particular time slot -> ROI null
  - Matrix of size 583 x 672

Datasets

- **San Francisco Cabs (SFC): (dense, irregular)**
  - Mobility traces by San Francisco taxis from May 19 to June 8, 2008
  - Each record consists of a cab identifier, latitude, longitude, and a time stamp.
  - 11M GPS coordinates - 534 cabs in SF – 3 weeks;
  - Grid 10×10 = 100 ROIs of 466198 square metres
  - One hour granularity: the 534 cabs report over 2M ROIs, on average 3.827 ± 1.069 locations per taxi, out of which 78 ± 6
  - ROIs are unique
  - High frequency: Taxis are active for 340 ± 94 out of the 504 timeslots in the 21 considered days
  - 1 if cab was in certain cell at time t and 0 otherwise

Datasets

- **San Francisco Cabs (SFC): (dense, irregular)**
  - Mobility traces by San Francisco taxis from May 19 to June 8, 2008
  - Each record consists of a cab identifier, latitude, longitude, and a time stamp.
  - 11M GPS coordinates - 534 cabs in SF – 3 weeks;
  - Grid 10×10 = 100 ROIs of 466198 square metres
  - One hour granularity: the 534 cabs report over 2M ROIs, on average 3.827 ± 1.069 locations per taxi, out of which 78 ± 6
  - ROIs are unique
  - High frequency: Taxis are active for 340 ± 94 out of the 504 timeslots in the 21 considered days
  - 1 if cab was in certain cell at time t and 0 otherwise

Do you think datasets are appropriate? Why?

Sampling Users

- Sort the users per total number of ROI reports (How many ROIs has one user visited)
- Split users in 3 groups of equal size (mobility patterns):
  - Highly mobile
  - Mildly mobile
  - Somewhat mobile
- Sample 50 users from each mobility group at random
- Membership inference attacks against 150 users for each dataset

Sampling Users

- Sort the users per total number of ROI reports (How many ROIs has one user visited)
- Split users in 3 groups of equal size (mobility patterns):
  - Highly mobile
  - Mildly mobile
  - Somewhat mobile

Do you think they are evenly distributed
Does this really avoid bias?

- Sample 50 users from each mobility group at random
- Membership inference attacks against 150 users for each dataset

Web Security & Privacy Lab

KAIST

Experimental Setup

- **Sample & Aggregate**:
  - Sample groups that **include** and **exclude** the target user to create a balanced dataset of labeled aggregate location time-series
- **Feature Extraction**: Extract various statistics from the time-series of each ROI
  - mean, variance, standard deviation, median, min, max, sum of values of each location's time-series
- **Classification**:
  - Train a classifier on the features extracted from the training set
  - Play the distinguishing game on the testing set
  - Classifiers: **Logistic Regression, Nearest Neighbors, Random Forests, Multi-Layer Perceptron**

Evaluating Membership Inference on Raw Aggregate Locations

- **Subset of Locations:**
  - Adversary knows the real locations of a subset of users, including the target user during the inference period
  - Create groups, with and without target, and train a classifier
  - Observation/Inference period: First week of both datasets
  - Telecommunications service provider getting locations from cell towers
- **Generate balanced training dataset by:**
  - Randomly sampling 400 unique user groups from Adversaries prior knowledge (1:1) (training)
  - Sampling 100 unique user groups from the set of users not in the prior knowledge (testing)

Transport For London

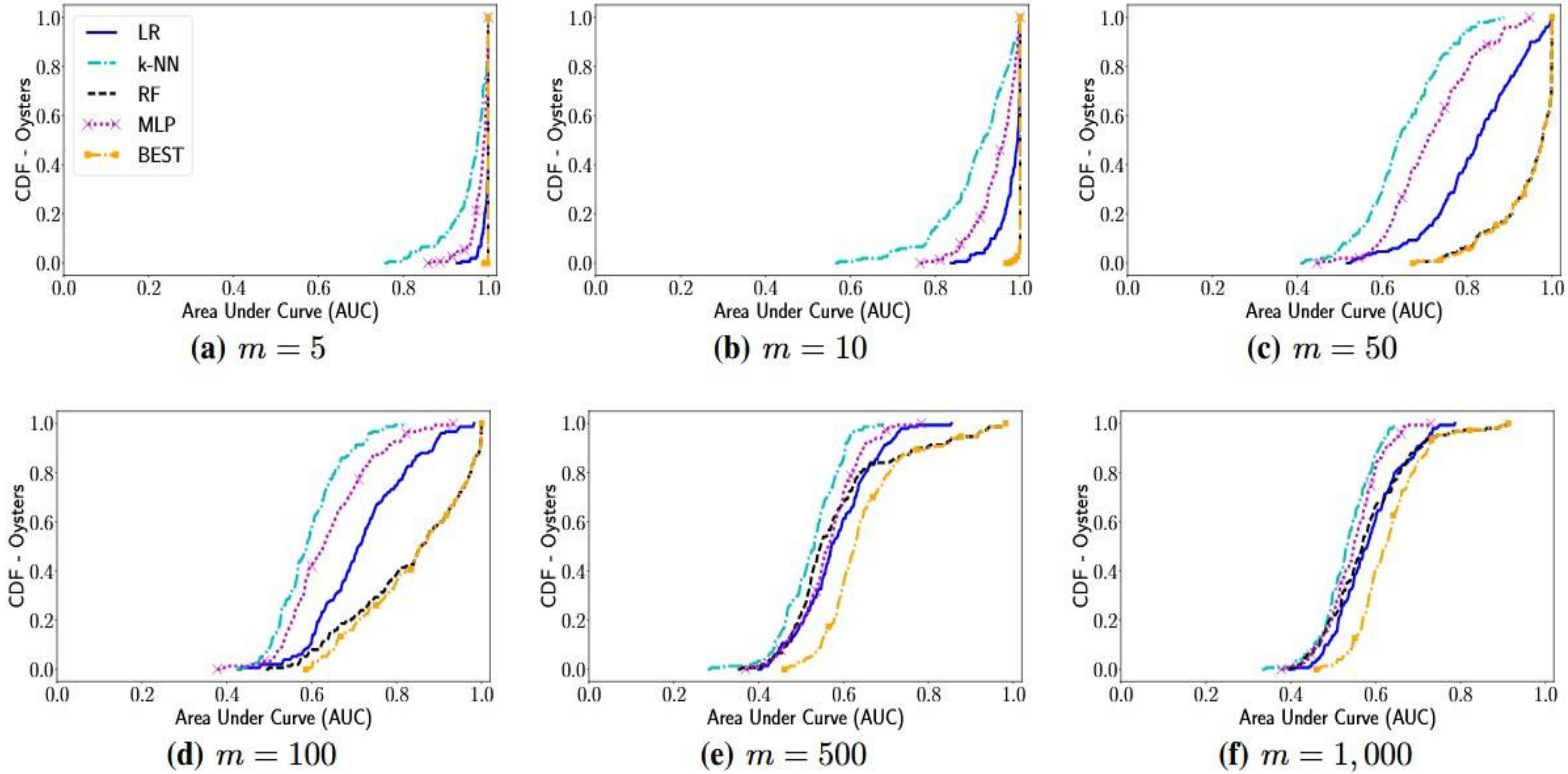
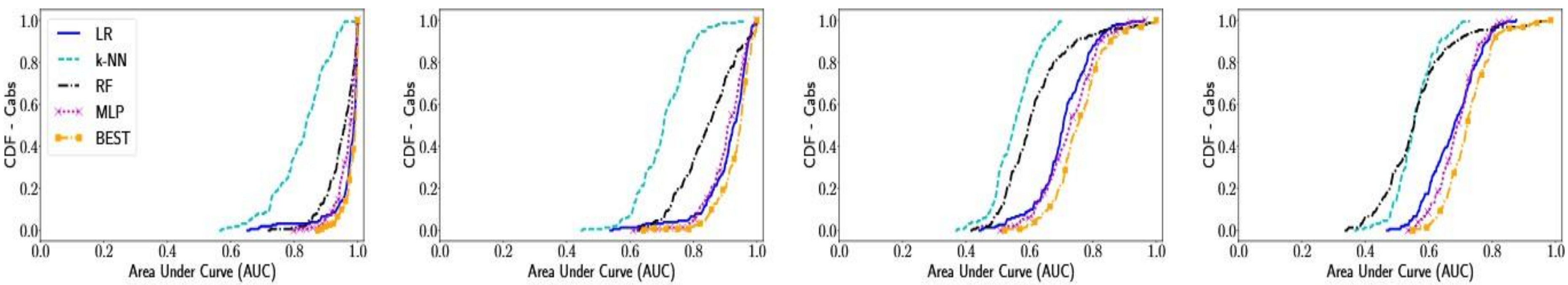


Fig. 2: *Subset of Locations* prior (TFL, $\alpha = 0.11$, $|T_I| = 168$) – Adv's performance for different values of m.

San Francisco Cabs

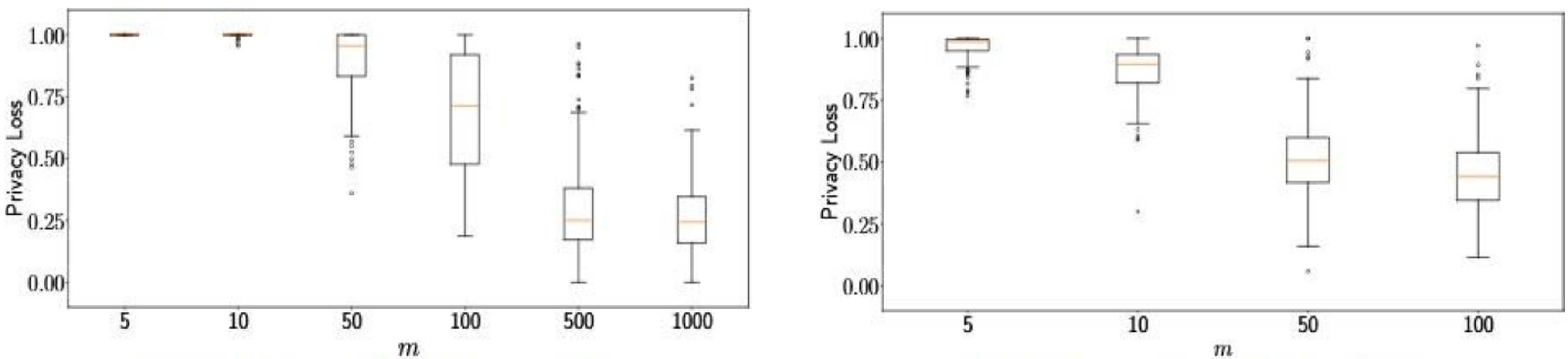
- The results for SFC resemble the ones for TFL



(a) $m = 5$ (b) $m = 10$ (c) $m = 50$ (d) $m = 100$

Fig. 4: *Subset of Locations* prior (SFC, $\alpha = 0.2$, $|T_I| = 168$) – Adv's performance for different values of m.

Transport For London



(a) TFL ($\alpha = 0.11$, $|T_I| = 168$) **(b)** SFC ($\alpha = 0.2$, $|T_I| = 168$)

Fig. 3: *Subset of Locations* prior - Privacy Loss (PL) for different values of m.

Web Security & Privacy Lab

KAIST

Transport For London

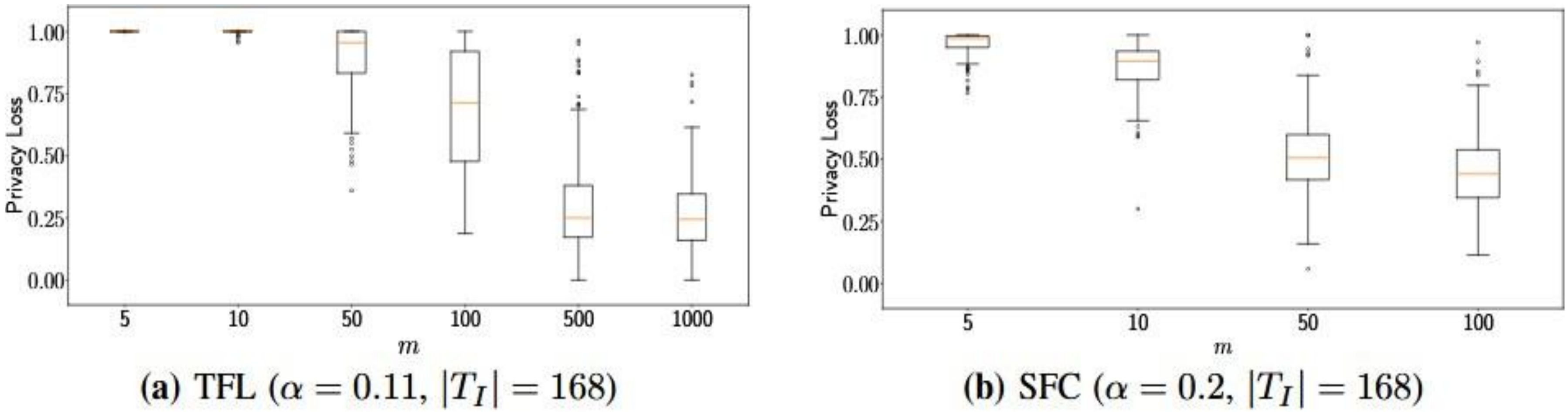


Fig. 3: *Subset of Locations* prior - Privacy Loss (PL) for different values of m.

- Why is the reason sparsity and not irregularity?

Participation in Past Groups

- **Same Groups as Released**: (Continuous data release over stable groups)
  - Adversary knows the target user's participation in past groups
  - The same groups are used to compute the aggregates during the inference period
  - Observation period:
    - TFL: first 3 weeks
    - SFC: first 2 weeks

Participation in Past Groups

- Inference period is the last week of data (168 hourly timeslots)
- Train the classifiers with features of **each week** in the training set
- Test on features extracted from the aggregates of each group in the test set
- There is no limitation of the prior -> Groups as large as the dataset

For large groups, target always in?

Participation in Past Groups

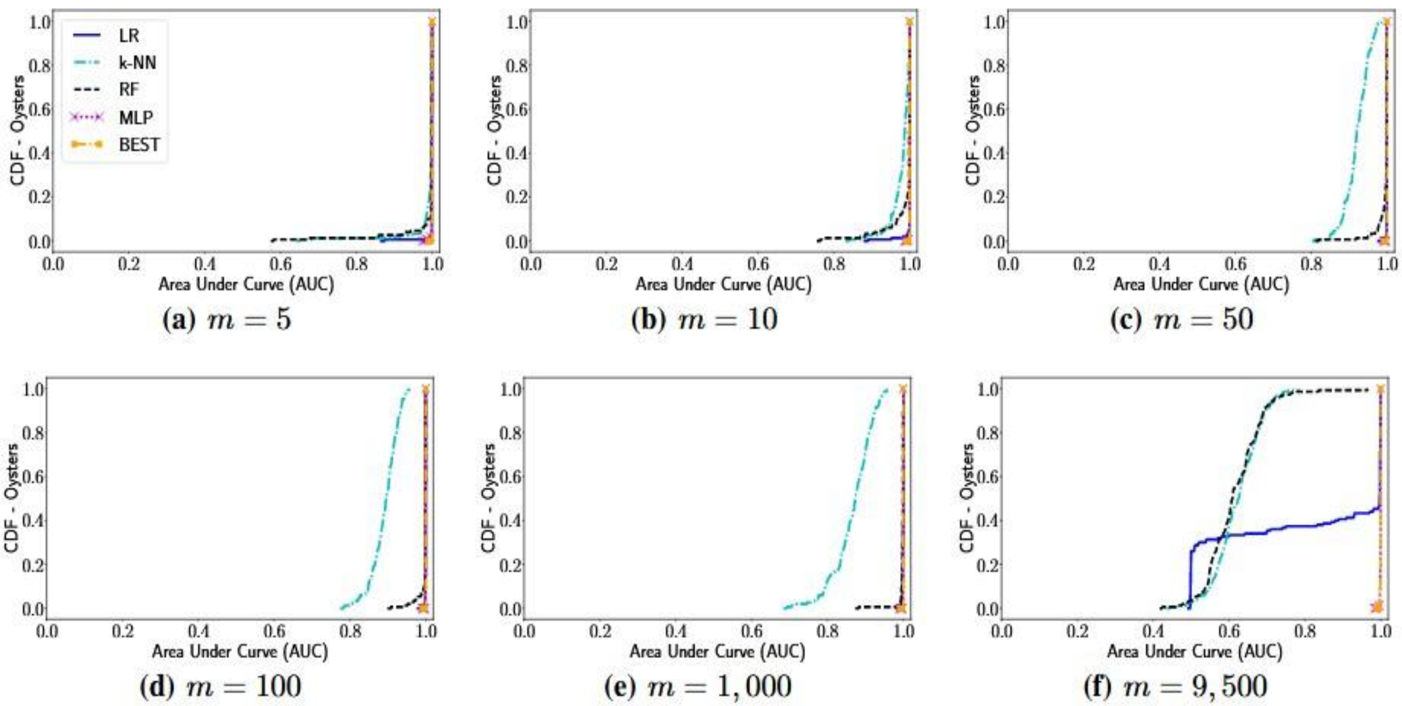
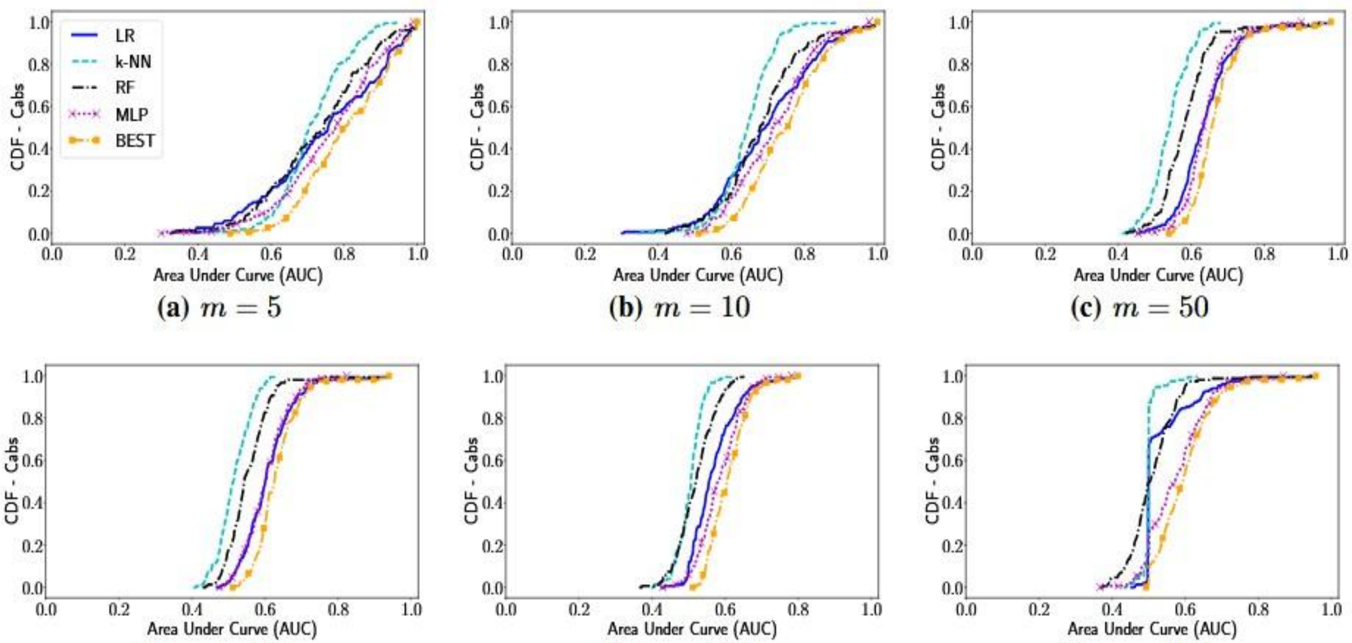


Fig. 5: *Same Groups as Released* prior (TFL, 75%-25% split, $\beta = 150$, $|T_I| = 168$) – Adv's performance for different values of m.

Participation in Past Groups

- **Small groups -> high AUC scores** for all classifiers (AUC scores over 0.9); m=9500: **MLP outperforms**
- **Regular mobility patterns** -> Successful membership inference, **even if they are larger**
  (For an adversary **with prior knowledge about specific groups** and the **groups are maintained**)

Participation in Past Groups



(a) $m = 5$ (b) $m = 10$ (c) $m = 50$

(d) $m = 100$ (e) $m = 300$ (f) $m = 500$

Fig. 7: *Same Groups as Released* prior (SFC, 67%-33% split, $\beta = 150$, $|T_I| = 168$) – Adv's performance for different values of m.

Participation in Past Groups

- **Large privacy loss** for users aggregated in groups for which the **adversary has prior knowledge**
- **Regularity** has a **strong effect** on membership inference
- Cabs lose privacy when they are aggregated in small groups

Web Security & Privacy Lab

KAIST

Participation in Past Groups

- **Different Groups than Released**: (Continuous data release over dynamic user group)
  - Adversary knows the user’s participation in past groups
  - These groups are not used to compute aggregates released in the inference period
  - For each target user, generate a dataset with the aggregates of 400 unique randomly sampled groups,half including the target and half not (1:1)
  - 75%-25% stratified random split on the dataset;
  - 300 groups for training and 100 groups for testing.
  - Observation period: first 3 weeks for TFL; first 2 weeks for SFC
  - Inference period is the last week of data (168 hourly timeslots)
  - Split the training and testing sets according to time

Participation in Past Groups

- Small groups (5, 10) -> high AUC scores
- Regularity still helps membership inference in small groups **even when these groups change**

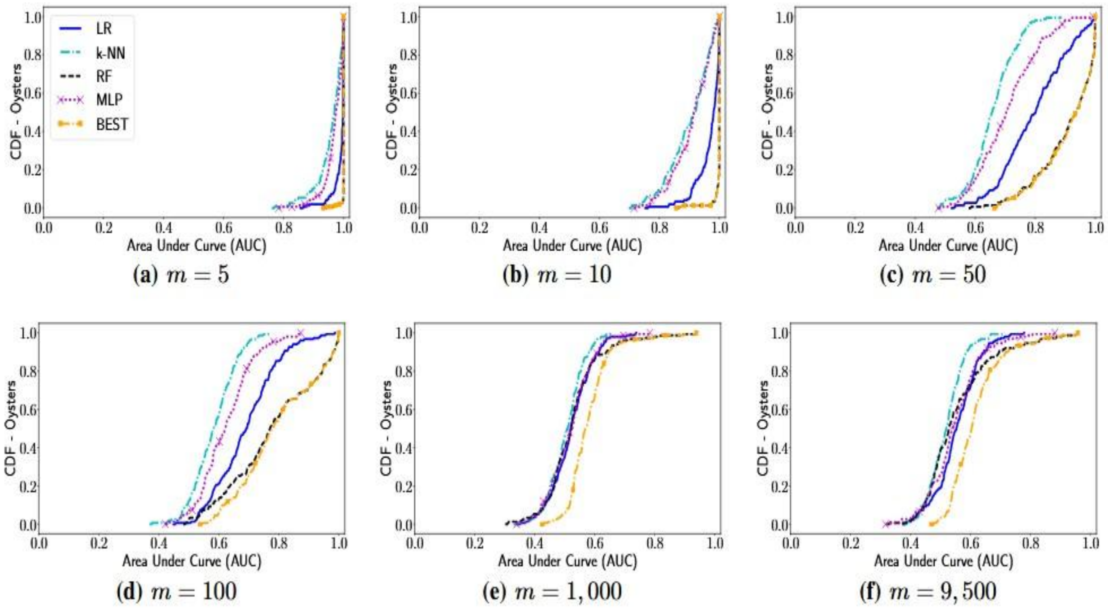


Fig. 8: *Different Groups than Released* prior (TFL, 75%-25% split, β=300, $|T_I|$=168) – Adv's performance for different values of m.

Participation in Past Groups

- **m=1000** all the classifiers perform, on average, similar to the baseline
  - Regularity has no effect
- **m=9500** small increase in the classifiers AUC scores due to the big user overlap across training and testing groups
  - The **different-groups** prior becomes more similar to the same-groups prior

Participation in Past Groups

- **Irregularity**: Classifiers perform worse for SFC than TDL
- **Small groups**: mean AUC drops to 0.71 for the best classifiers, LR and MLP
  - With larger groups the performance is significantly lower, near random baseline

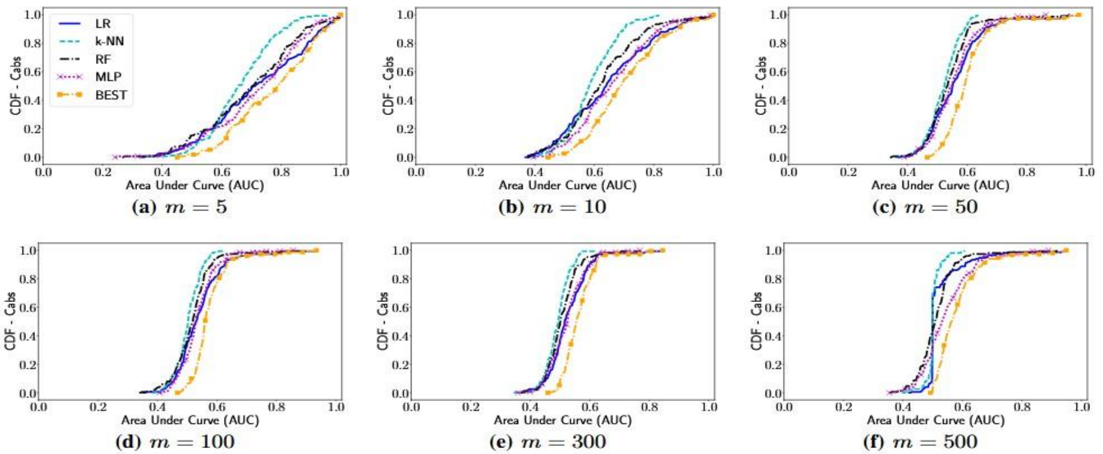
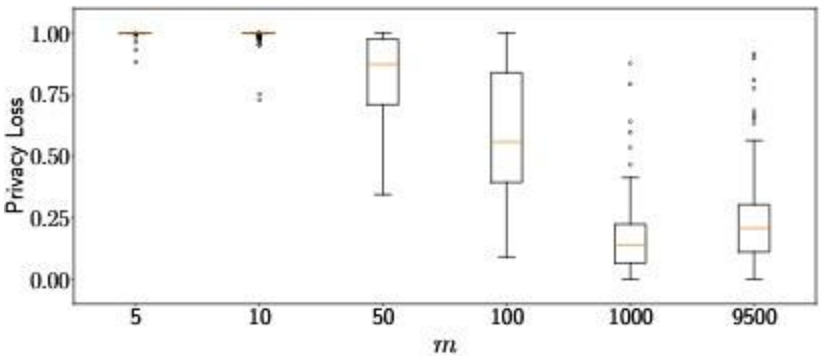
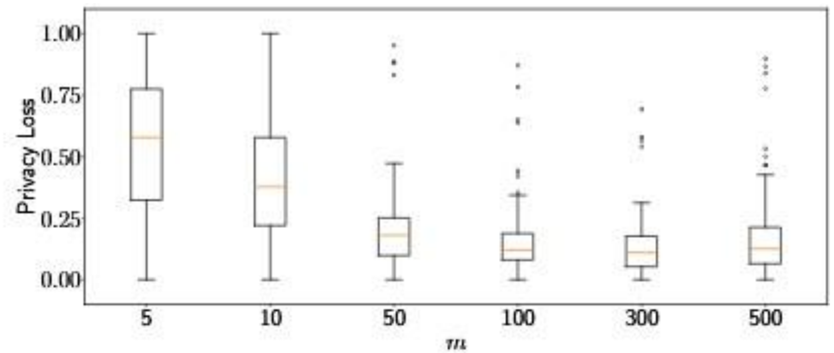


Fig. 10: *Different Groups than Released* prior (SFC 67%-33% split, β=300, $|T_I|$=168) – Adv's performance for different values of m.

Participation in Past Groups



(a) TFL, 75% – 25% split, $\beta = 300$, $|T_I| = 168$



(b) SFC 67% – 33% split, $\beta = 300$, $|T_I| = 168$

Fig. 9: *Different Groups than Released* prior - Privacy Loss (PL) for different values of m.

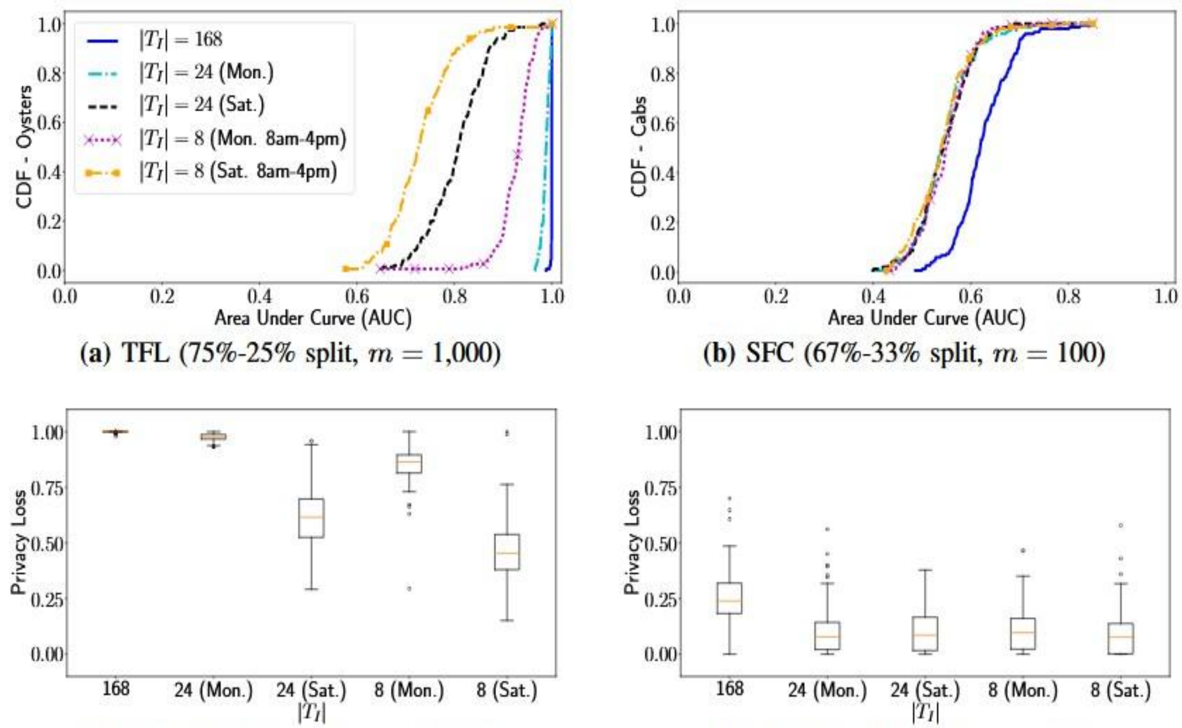
Participation in Past Groups

- **Up to m=100** Membership inference is quite effective privacy loss of at least 0.95
- **m>100** mean PL decreases
- Overall: Privacy loss is smaller in this setting
  - **Weaker adversarial setting** than the previous one
- **Weaker prior**: PL values are overall smaller compared to the previous setting
- PL decreases with increasing aggregation group size

Length of Inference Period

- Consider lengths of 1 week (168 hourly timeslots), 1 day (24 timeslots), and 8 hours (8 timeslots)
- For the last two, also consider working vs weekend
- **Only** report experiments in the "Same groups as released" setting
- Fix the group size to 1000 commuters for TFL and to 100 cabs for SFC
- For each target user create a dataset of 150 random unique groups
- Choose Random Forest for TFL and MLP for SFC

Length of Inference Period



(a) TFL (75%-25% split, $m = 1{,}000$)
(b) SFC (67%-33% split, $m = 100$)
(c) TFL (75%-25% split, $m = 1{,}000$)
(d) SFC (67%-33% split, $m = 100$)

Fig. 11: *Same Groups as Released* prior (β=150) - Adv's performance for variable inference period length ($|T_I|$), on (a) TFL and (b) SFC, and Privacy Loss on (c) TFL and (d) SFC.

Length of Inference Period

- **a)**
  - Shorter inference period -> lower adversarial performance
    -> less information about mobility patterns to be exploited
  - Difference between working days and weekends (Monday: Mean AUC is 0.97; Saturday: 0.8)
  - Monday 8am-4pm: better results than on a Saturday (same time frame)
- **b)**
  - Irregularity -> lower adversarial performance
  - Irregularity -> no significant difference between working days and weekdays Irregularity
    -> Smaller PL for SFC cabs, for all period lengths

Raw Aggregates Evaluation – Take Aways

- The actual level of **privacy leakage** depends on:
  - Adversary's prior knowledge
  - Characteristics of the data
  - Group size
  - Timeframe of aggregation
- **Less successful** with **increasing group sizes**
- Successful, if actual **locations** of a subset of **users** (including the target) are **known** and when **knowing past aggregates** for the **same groups**

Raw Aggregates Evaluation – Take Aways

- Privacy leakage on TFL is larger than on SFC
  - **Regularity** in users' movement and **sparseness** of the location significantly eases the task
- The **length**, as well as the **time semantics**, of the inference period play an important role (not for SFC though)
- Inference is easier if the aggregates of **longer periods** are released and at times when mobility patterns are likely to be **more regular**

Evaluating Defenses

- **Differential Privacy**: Define private functions that are free from inferences
  - Only a bounded amount of information is disclosed upon its release
  - Can mitigate membership inference attacks
- **Sensitivity**: Captures how much one record affects the output of a function
- **Laplacian Mechanism (LPA)**:
  - randomize the aggregate statistics using random noise independently drawn from the Laplacian distribution
  - A weaker version of LPA for time-series, which perturbs the counts of a time-series = Baseline

Web Security & Privacy Lab

KAIST

Evaluating Defenses

- **Gaussian Mechanism**:
  - Perturbing the statistics with random noise drawn from the Gaussian distribution
- **Fourier Perturbation Algorithm (FPA)**:
  - Performs the noise addition on the compressed frequency domain
- **Enhanced Fourier Perturbation Algorithm with Gaussian Noise (EFPAG)**:
  - Improves FPA

Web Security & Privacy Lab

KAIST

Experimental Design

- Evaluate the effectiveness of differentially private mechanisms in defending against membership inferences
- Evaluate over large groups
  - for small groups, the loss of utility incurred by DP-based mechanisms is prohibitively high

Web Security & Privacy Lab

KAIST

Evaluating Differentially Privacy (DP) Mechanisms

- Worst-case adversary that obtains perfect prior knowledge for the users
  - given raw aggregates she can train a classifier that achieves AUC score of 1.0
- Modification to the game: the challenger applies a DP mechanism before sending the challenge to the adversary
  - LPA, GSM, FPA, EFPAG

Evaluating Differentially Privacy (DP) Mechanisms

- Evaluate the privacy/utility tradeoff of differentially private mechanisms considering:
  - Best setting for utility
  - Worst setting for privacy
- Evaluate the gain in privacy on two cases: Adversaries classifier is trained on
  - The raw aggregates of the groups to be released (passive adversary)
  - Noisy aggregates of the groups to be released using the defense mechanism under examination(active adversary)

Web Security & Privacy Lab

KAIST

Experiment Settings

- Membership inference against 150 sampled users
- Observation/Inference period: first week in each dataset
- Favorable setting for the utility of DP-based mechanism:
  - construct large user groups m=9500 for TFL, m=500 for SFC

Does the m=500 for SFC really show us something in this case?

Web Security & Privacy Lab

KAIST

Experiment Settings

- Generate dataset by randomly sampling 200 and 400 elements for TFL and SFC (1:1)
- Classifier: MLP
- For Perturbation mechanism: Compute sensitivity for users in each dataset
  - = maximum number of ROIs reported by an oyster/cab in the inference week

Why is MLP used?

Web Security & Privacy Lab

KAIST

Metrics

- **Privacy Gain**:
  - Relative decrease in the adversary's performance when challenged on perturbed aggregates vs. raw aggregates

$$\mathrm{PG} = \begin{cases} \frac{\mathrm{AUC}_A - \mathrm{AUC}_{A'}}{\mathrm{AUC}_A - 0.5} & \text{if } \mathrm{AUC}_A > \mathrm{AUC}_{A'} \geq 0.5 \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

Metrics

- **Utility: Mean Relative Error (MRE)**
  - MRE computed between the raw aggregate time series and its perturbed version
  - γ is a sanity bound mitigating the effect of very small counts

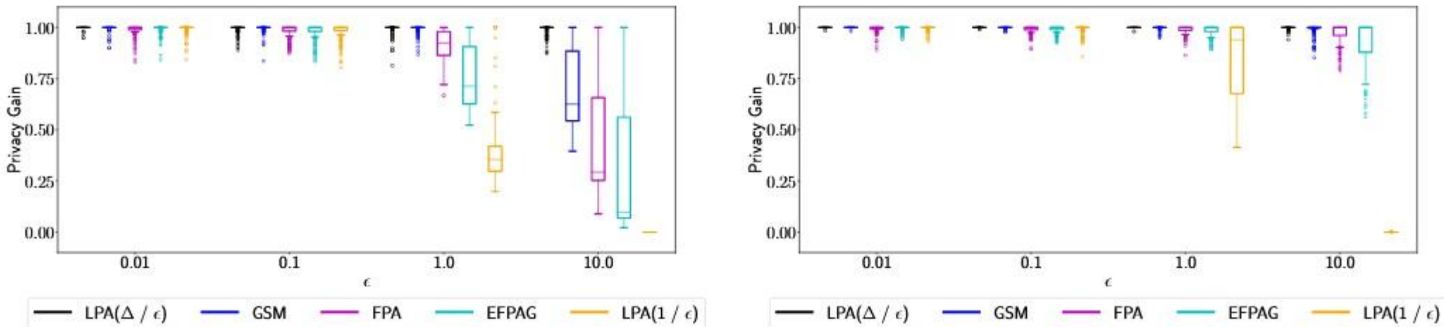
$$\mathrm{MRE}(Y, Y') = \frac{1}{n} \sum_{i=1}^{n} \frac{|Y_i' - Y_i|}{\max(\gamma, Y_i)} \tag{7}$$

Web Security & Privacy Lab

KAIST

Results

- Train on raw/ Test on Noisy Aggregates



(a) TFL ($m = 9,500$, $|T_I| = 168$) (b) SFC ($m = 500$, $|T_I| = 168$)

Fig. 12: Privacy Gain (PG) achieved by differentially private mechanisms with different values of ϵ, against a MLP classifier trained on raw aggregates and tested on noisy aggregates.

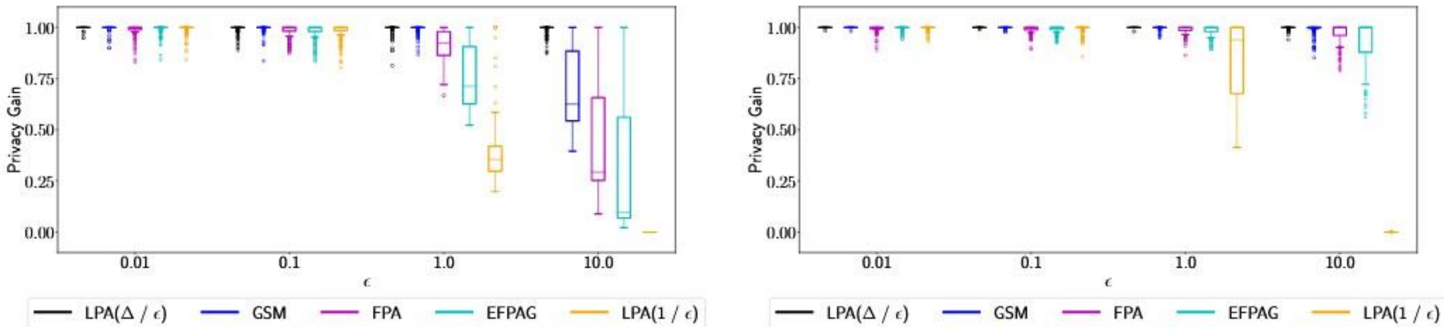
- low ε values (up to 0.1): all mechanisms provide **excellent privacy protection**
- But poor utility (Table 2)
- As eps increases to 1 LPA(Δ/ε) and GSM still provide good protection

Web Security & Privacy Lab

KAIST

Results

- Train on raw/ Test on Noisy Aggregates



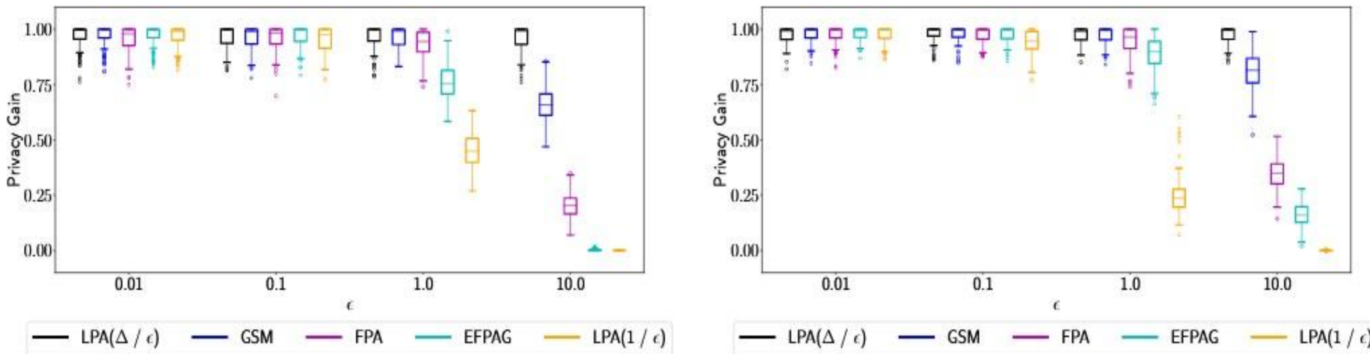
(a) TFL ($m = 9,500$, $|T_I| = 168$) (b) SFC ($m = 500$, $|T_I| = 168$)

Fig. 12: Privacy Gain (PG) achieved by differentially private mechanisms with different values of ϵ, against a MLP classifier trained on raw aggregates and tested on noisy aggregates.

- ε up to 1: high PG for all mechanisms But poor Utility
- ε =10: mean PG is almost 1 for LPA(Δ/ε) and GMS,
    - Users are well protected against MIA

Results

- **Train on Noisy / Test on Noisy Aggregates**



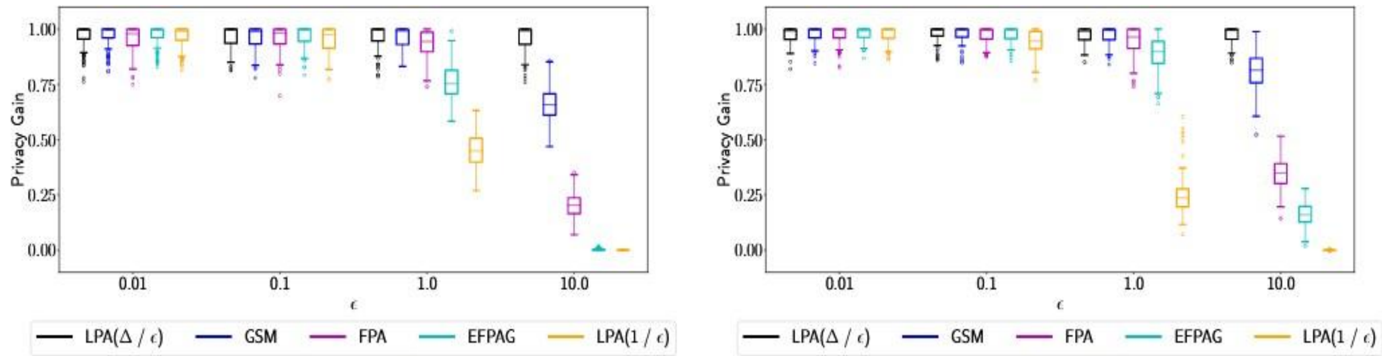
(a) TFL ($m = 9,500$, $|T_I| = 168$) (b) SFC ($m = 500$, $|T_I| = 168$)

Fig. 13: Privacy Gain (PG) achieved by differentially private mechanisms with different values of ϵ, against a MLP classifier trained and tested on noisy aggregates.

  - Increasing values of ε: Protection of the mechanisms decreases much faster

Results

- **Train on Noisy / Test on Noisy Aggregates**



(a) TFL ($m = 9,500$, $|T_I| = 168$) (b) SFC ($m = 500$, $|T_I| = 168$)

Fig. 13: Privacy Gain (PG) achieved by differentially private mechanisms with different values of ϵ, against a MLP classifier trained and tested on noisy aggregates.

- ε <=1: mean PG remains high for all mechanisms, (except for LPA(1/ε))
- ε=10: significant decline in PG with GSM, FPA, EFPAG
  - This corresponds to a significant drop in privacy protection compared to the setting where training was done on raw aggregates

Take Aways

- DP mechanisms are overall successful at preventing membership inference
- Caveat:
  - A **passive adversary** who trains a classifier on raw aggregate location data is not very successful at inferring membership on noisy aggregates
- **Strategic Adversary**: The actual privacy gain offered from the DP-based mechanisms is significantly reduced, and also decreases much faster with increasing eps values
- But, with significant reduction in the utility of the aggregates

Web Security & Privacy Lab

KAIST

Take Aways

- A strategic adversary that mimics the behavior of the defender can reduce the privacy gain offered by a mechanism
- Mechanisms specifically designed for time-series settings (e.g., FPA, EFPAG) achieve better utility, at the cost of reduced privacy
- Trade-off between privacy and utility
- The methods can be used to evaluate defense mechanisms

Web Security & Privacy Lab

KAIST

Conclusion

- Membership inference is very accurate when groups are small
- Users that have regular habits are easier to classify
- Raw aggregates leak information about user membership
- Effectiveness of defense mechanisms based on differential privacy:
  - Quite effective if the adversary trains the classifier on raw aggregates (but loss in utility)
  - Less effective if the adversary mimics the behavior of the perturbation mechanism by training the classifier on noisy aggregates

Questions

Web Security & Privacy Lab

KAIST