DS-GA 1017, Responsible Data Science, Spring 2022

Homework 2: The Data Science Lifecycle, Differential Privacy
Due at 11:59pm EDT on Thursday, April 14

Objectives and Learning Outcomes

This assignment consists of written problems and programming exercises on the data science lifecycle and data protection. In the programming part of the assignment you will use the DataSynthesizer and MST libraries for privacy-preserving synthetic data generation.[1]

After completing this assignment, you will:

1. explore the interaction between the complexity of the learned model (a summary of the real dataset) and the accuracy of results of statistical queries on the derived synthetic dataset, under differential privacy
2. understand the variability of results of statistical queries under differential privacy, by generating multiple synthetic datasets under the same settings (model complexity and privacy budget), and observing how result accuracy varies
3. explore the trade-off between privacy and utility, by generating and querying synthetic datasets under different privacy budgets, and observing the accuracy of the results

You must work on this assignment individually. If you have questions about this assignment, please post a private message to all instructors on Piazza.

Grading

The homework is worth 80 points, or 10% of the course grade. Your grade for the programming portion will be significantly impacted by the quality of your written report for that portion. In your report, you should explain your observations carefully.

You are allotted 2 (two) late days over the term, which you may use on a single homework, or on two homeworks, or not at all. If an assignment is submitted at most 24 hours late -- one day is used in full; if it's submitted between 24 and 48 hours late -- two days are used in full.

Submission instructions

Provide written answers to Problems 1, 2, and 3 in a single PDF file created using LaTeX. (If you are new to LaTeX, Overleaf is an easy way to get started.) Provide code in answer to Problem 4 in a Google Colaboratory notebook. Both the PDF and the notebook should be turned in as Homework 1 on BrightSpace. Please clearly label each part of each question.

[1] Both methods are described in papers that are part of the Data Protection reader.

Problem 1 (10 points): Racial disparities in predictive policing

A 2016 study by Lum and Isaac[2] found a racial disparity in the number of drug arrests made in Oakland, CA: while estimated drug use does not differ by race, the number of drug-related arrests is much higher in Black neighborhoods. These findings are substantiated by Figure 1, which we reproduced here.

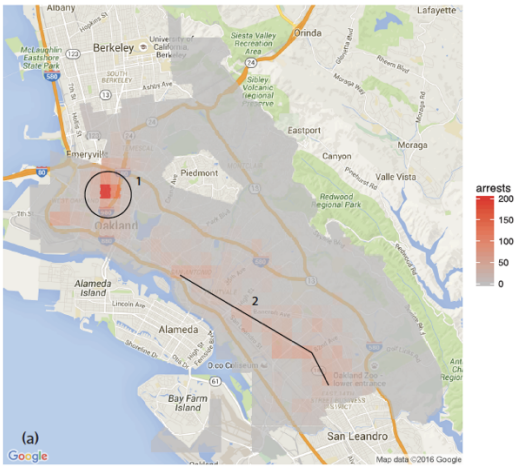


Figure 1(a): Number of drug arrests, 2010.

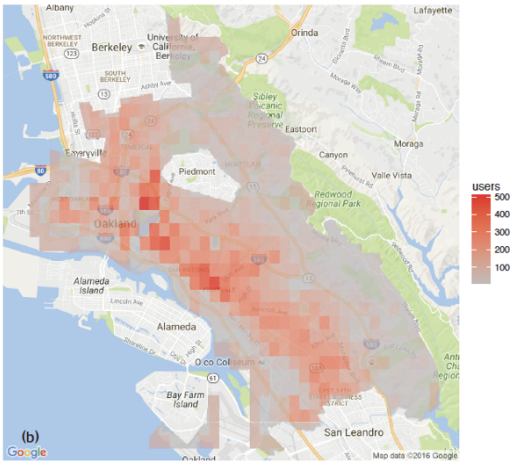


Figure 1(b): Estimated number of drug users, 2011.

Figure 1(a) reports the number of drug arrests made by the Oakland police department in 2010, and Figure 1(b) reports the estimated number of drug users, based on the 2011 National Survey on Drug Use and Health. Reproduced from Lum & Isaac, Significance, 2016.

Consider a hypothetical machine learning system, such as the one scrutinized by Lum and Isaac, that uses historical data to determine which neighborhoods would be likely targets for drug-related policing activities. The system may use past crime data along with other potentially useful datasets (e.g., time of year, weather, number of clubs and bars in the neighborhood, etc.).

(a) **(4 points)** Give **three distinct reasons** why racial disparities might arise in the predictions of such a system.

(b) **(6 points)** Propose **two mitigation strategies** to counteract racial disparities in the predictions of such a system. Note: It is insufficient to state that we could use a specific pre-, in- or post-processing technique that we covered in class when we discussed fairness in classification. Additional details are needed to demonstrate your understanding of how the ideas from fairness in classification would translate to this scenario.

[2] We discussed the study by Lum & Isaac during class. The article is available as part of the Data Science Lifecycle reader.

Problem 2 (10 points): Randomized response

The simplest version of randomized response involves flipping **a single fair coin** (50% probability of heads and 50% probability of tails). As in the example we saw in class, an individual is asked a potentially incriminating question, and flips a coin before answering. If the coin comes up tails, he answers truthfully, otherwise he answers "yes". Is this mechanism differentially private? If so, what epsilon value does it achieve? *Carefully justify your answer.*

Problem 3 (15 points): Classification association rules

Consider the dataset below, and assume that **sex** is one of {M, F}; **edu** is one of {HS, BS, MS}; and **loan** is one of {yes, no}. Here, **sex** is the protected attribute, and **loan** represents the binary classification outcome (the target variable): **loan=yes** is the positive outcome, **loan=no** is the negative outcome.

id	sex	edu	loan
F1	F	HS	no
F2	F	HS	no
F3	F	HS	no
F4	F	HS	no
F5	F	HS	no
F6	F	HS	no
F7	F	BS	yes
F8	F	BS	yes
F9	F	BS	yes
F10	F	BS	no
F11	F	BS	no
F12	F	BS	no
F13	F	MS	yes
F14	F	MS	yes
F15	F	MS	no
F16	F	MS	no

id	sex	edu	loan
M1	M	HS	yes
M2	M	HS	yes
M3	M	HS	yes
M4	M	HS	no
M5	M	HS	no
M6	M	HS	no
M7	M	BS	yes
M8	M	BS	yes
M9	M	BS	yes
M10	M	BS	yes
M11	M	BS	no
M12	M	BS	no
M13	M	MS	yes
M14	M	MS	yes
M15	M	MS	yes
M16	M	MS	yes

A classification association rule (CAR) is a *non-trivial association rule* of the form $\mathbf{X_1, \ldots, X_n \rightarrow Y}$, where **Y** is an assignment of a value to the target variable (**loan=yes** or **loan=no**), $\mathbf{X_1, \ldots, X_n}$ is an assignment of values to one or several other variables. For example: **sex=M, edu=BS → loan=yes** is a CAR, while **loan=yes → sex=F** is not.

To mine CARs from a dataset, you may think of each tuple (row) as a “transaction”, and then apply the *Apriori* algorithm we covered in class (during the data profiling lecture in Week 6) to find CARs that meet or exceed the specified confidence and support thresholds.

(a) (5 points) List all CARs that relate the likelihood of the classification outcome (**loan=yes** or **loan=no**), with the value of the sensitive attribute sex. These CARs should list sex on the left-hand-side, either on its own or in combination with other attributes. List only those CARs that have **support >= 3** and **confidence >= 0.6**. For each CAR you list, state its support and confidence.

(b) (10 points) Suppose that you are required to release differentially private versions of the frequent itemsets (the union of their left-hand-side and right-hand-side) that correspond to the CARs you computed in part (a), along with their support. Your overall privacy budget is $\varepsilon=1$. Use sequential and parallel composition to allocate portions of the privacy budget to each frequent itemset you will release. Your goal is to maximize utility of the information you release, while staying within the privacy budget.

Write down a way to allocate portions of the privacy budget to each frequent itemset you will release, to achieve good utility. Be specific, write down an epsilon value for each itemset. Carefully justify your solution using sequential and parallel composition.

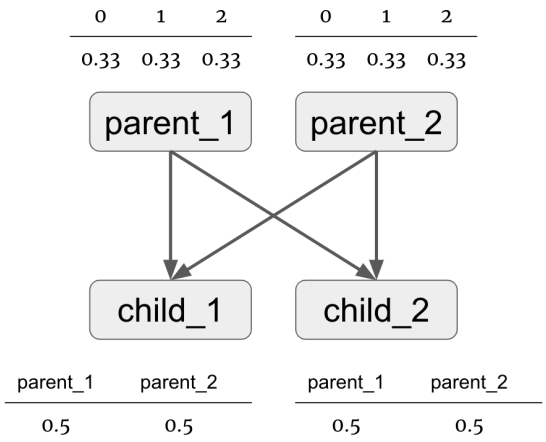
Problem 4 (45 points) : Privacy-preserving synthetic data

In this problem, you will take on the role of a data owner, who owns two sensitive datasets, called **hw_compas** and **hw_fake**, and is preparing to release differentially private synthetic versions of these datasets.

The first dataset, **hw_compas** is a subset of the dataset released by ProPublica as part of their COMPAS investigation. The **hw_compas** dataset has attributes age, sex, score, and race, with the following domains of values: age is an integer between 18 and 96, sex is one of ‘Male’ or ‘Female’, score is an integer between -1 and 10, race is one of 'Other', 'Caucasian', 'African-American', 'Hispanic', 'Asian', 'Native American'.

The second dataset, **hw_fake**, is a synthetically generated dataset. We call this dataset “fake” rather than “synthetic” because you will be using it as *input* to a privacy-preserving data generator. We will use the term “synthetic” to refer to privacy-preserving datasets that are produced as *output* of a data generator.

We generated the **hw_fake** dataset by sampling from the following Bayesian network:



In this Bayesian network, **parent_1**, **parent_2**, **child_1**, and **child_2** are random variables. Each of these variables takes on one of three values {0, 1, 2}.

- Variables **parent_1** and **parent_2** take on each of the possible values with an equal probability. Values are assigned to these random variables independently.
- Variables **child_1** and **child_2** take on the value of one of their parents. Which parent's value the child takes on is chosen with an equal probability.

To start, use the Data Synthesizer library to generate 4 synthetic datasets for each sensitive dataset **hw_compas** and **hw_fake** (8 synthetic datasets in total), each of size N=10,000, using the following settings:

- A: random mode
- B: independent attribute mode with **epsilon = 0.1**.
- C: correlated attribute mode with **epsilon = 0.1**, with Bayesian network degree **k=1**
- D: correlated attribute mode with **epsilon = 0.1**, with Bayesian network degree **k=2**

For guidance, you can use the HW2_Template here. Please make sure to duplicate this file rather than put your code directly here

(a) (15 points): Execute the following queries on synthetic datasets and compare their results to those on the corresponding real datasets:

- **Q1** (**hw_compas** only): Execute basic statistical queries over synthetic datasets.

  The **hw_compas** has numerical attributes **age** and **score.** Calculate **Median, Mean, Min, Max** of **age** and **score** for the synthetic datasets generated with settings A, B, C, and D (described above). Compare to the ground truth values, as computed over **hw_compas**. Present results in a **table**. Discuss the accuracy of the different methods in your report. Which methods are accurate and which are less accurate? If there are substantial differences in accuracy between methods - explain these differences.

- **Q2 (hw_compas** only): Compare how well random mode (A) and independent attribute mode (B) replicate the original distribution.

Plot the distributions of values of **age** and **sex** attributes in **hw_compas** and in synthetic datasets generated under settings A and B. Compare the **histograms** visually and explain the results in your report.

Next, compute cumulative measures that quantify the difference between the probability distributions over age and sex in **hw_compas** vs. in privacy-preserving synthetic data. To do so, use the Two-sample Kolmogorov-Smirnov test (KS test) for the numerical attribute and Kullback-Leibler divergence (KL-divergence) for the categorical attribute, using provided functions **ks_test** and **kl_test**. Discuss the relative difference in performance under A and B in your report.

For Two-sample Kolmogorov-Smirnov test and Kullback-Leibler divergence, you might find functions such as '*entropy*' and *'ks_2samp*' from *scipy.stats* useful.

- **Q3** (**hw_fake** only): Compare the accuracy of correlated attribute mode with k=1 (C) and with k=2 (D).

  Display the pairwise mutual information matrix by heatmaps, showing mutual information between all pairs of attributes, in **hw_fake** and in two synthetic datasets (generated under C and D). Discuss your observations, noting how well / how badly mutual information is preserved in synthetic data.

  To compute mutual information, you can use functions from https://github.com/DataResponsibly/DataSynthesizer/blob/master/DataSynthesizer/lib/utils.py
  For heatmaps, we suggest considering functions (*heatmap*) provided in the seaborn library (see example: https://seaborn.pydata.org/examples/many_pairwise_correlations.html) and remember to set up *vmax* and *vmin* when plotting.

(b) (5 points, hw_compas only**)**: Study the variability in accuracy of answers to Q1 under part (a) for A, B, and C for attribute **age**.

To do this, fix epsilon = 0.1, generate 10 synthetic databases (by specifying different seeds). Plot **median, mean, min, max** as a **box-and-whiskers** plot of the values for all 10 databases, and evaluate the accuracy of the synthetic data by comparing these metrics to the ground truth median, mean, min, and max from the real data. Carefully explain your observations: which mode gives more accurate results and why? In which cases do we see more or less variability?

Specifically for the box-and-whiskers plots, we expect to see four subplots: one for each of the **median, mean, min, max,** with the three parameter settings (A, B and C) along the X-axis and age on the Y-axis.

(c) (10 points, both datasets): Study how well statistical properties of the data are preserved as a function of the privacy budget. To see robust results, execute your experiment with 10 different synthetic datasets (with different seeds) for each value of epsilon, for each data generation setting (B, C, and D). Compute the following metrics, visualize results as appropriate with box-and-whiskers plots, and discuss your findings in the report.

- KL-divergence over the attribute **race** in **hw_compas**. Vary epsilon from 0.01 to 0.1 in increments of 0.01, generating synthetic datasets under B, C, and D.

  Specifically, the epsilons are [0.01, 0.02, 0.03, 0.04, 0.05, 0.06, 0.07, 0.08, 0.09, 0.1] and in total, you should have 3*10*10 datasets generated. Please plot the distributions of KL-divergence scores (10 samples each) with box-and-whiskers plots where you treat epsilon as the X-axis and generation settings as subplots.

- The difference in pairwise mutual information, aggregated (summed up) over all pairs of attributes, for both **hw_compas** and **hw_fake**, computed as follows:

  Suppose that m_ij represents the mutual information between attributes and i and j derived from sensitive dataset D, and m'_ij represents the mutual information between the same two attributes, i and j, derived from some privacy-preserving synthetic counterpart dataset D'. Compute the sum, over all pairs i, j, with i<j, of the absolute value of the difference between m_ij and m'_ij: $\sum_{i<j} \left| m_{ij} - m'_{ij} \right|$

  Run these experiments for the following epsilon values: 0.0001. 0.001, 0.01, 0.1, 1, 10, and 100, generating synthetic datasets under B, C and D. Specifically, you should have 3*7*10 datasets generated for each hw_compas and hw_fake.

  You should generate 3 plots, one for each data generation method (i.e., one plot for B, one for C, and one for D). The y-axis in all cases should start at 0. All plots should have the same range of y-axis values, so that the values are comparable across experiments.

(d) (5 points, hw_fake): Compare the DP model learned by the DataSynthesizer with the model learned by another synthesizer, MST, from the Private-PGM library.

Recall from our discussion in class that, instead of maintaining an internal Bayesian network, as does DataSynthesizer, MST finds the maximum spanning tree on a graph where nodes are data attributes and edge weights correspond to approximate mutual information between any two attributes. The spanning tree is then used to decide which 2-way marginals to estimate.

To start, use MST to compute the differentially private spanning tree for **hw_fake**, with **epsilon = 0.1**. Compare the spanning tree produced by MST with the Bayesian network produced by the Data Synthesizer under condition D (correlated attribute mode with k=2, with epsilon=0.1). Inspect the spanning tree, discuss which marginals were selected, and how the information they capture is similar or different compared to the conditional tables that are estimated by the

Bayesian network of the Data Synthesizer in condition D. (Note that there is no easy way to look at the counts inside the marginal, so this question is asking you to discuss the structure of the models.)

(e) (10 points, both datasets): Repeat part of question (c) above for MST, for both **hw_compas** and **hw_fake**. Let's refer to this as condition E: MST with epsilon as specified below, generating synthetic datasets of size N=10,000.

- Generate synthetic datasets under condition E as follows: Vary epsilon from 0.01 to 0.1 in increments of 0.01, and generate 5 different datasets for each value of epsilon. (We are reducing the number of datasets to 5, down from 10, because MST takes longer to run.) In total you should have 5*10 datasets generated. Measure KL-divergence over the attribute **race** in **hw_compas**. Plot the distribution of KL-divergence scores (10 samples each) with box-and-whiskers, with epsilon on the X-axis.

- Generate synthetic datasets under condition E as follows: Use epsilon values 0.0001, 0.001, 0.01, 0.1, 1, 10, and 100, and generate 5 different datasets for each value of epsilon, for each **hw_compas** and **hw_fake**. In total you should have 2 * 7 * 5 datasets generated. Compute and plot the difference in pairwise mutual information, aggregated (summed up) over all pairs of attributes, for both **hw_compas** and **hw_fake**, as in part (c) above. The y-axis should start at 0. Ensure that your plot has the same range of y-axis values as the plots in part (c), so that the values are comparable across experiments.

Discuss your findings, comparing performance of MST under condition E and of Data Synthesizer under condition D.