Homework 2

Pack all the files into a zip file "*yourname_hw2.zip*" (**Bonus: 2 Points**) and submit it on the Canvas by **October 16th, 2023**. Notice that:

- Penalty will apply for late submissions (per our syllabus).
- If you would like to let us know how to run your program, please feel free to include it in the report (but this is optional). Thank you.

1. Differentially Private Classification (50 points)

Considering the application of classifying the "iris plant" using the following dataset:

- The full dataset and description are available at: https://archive.ics.uci.edu/ml/datasets/Iris.
- Four attributes and three classes (using iris.data).
  1. sepal length in cm
  2. sepal width in cm
  3. petal length in cm
  4. petal width in cm
  5. class: Iris Setosa, Iris Versicolour, and Iris Virginica
- It is a small-scale dataset (150 records). Consider records (1-10, 51-60, 101-110) as the testing data for prediction, and the remaining 120 records as the training data.

Tasks:

(a) Build a Naive Bayes Classifier to predict the classes for records 1-10, 51-60, and 101-110 (both training and prediction). See more information about Naive Bayes Classifier: https://en.wikipedia.org/wiki/Naive_Bayes_classifier. (**10 points**)

(b) Design and implement a differentially private algorithm (satisfying ϵ-differential privacy) to train the Naive Bayes Classifier for prediction. (**20 points**)

Hints: you can consider the Laplace Mechanism and allocate budgets to ensure ϵ-differential privacy for the algorithm. The algorithm consists of many queries, and you can think about sequential composition and parallel composition for the queries.

(c) Prove ϵ-differential privacy for your designed algorithm. (**5 points**)

(d) Set $\epsilon = 0.5, 1, 2, 4, 8$, and 16. Then, calculate the precision and recall of the prediction results (of records 1-10, 51-60, and 101-110) generated by the differentially private classifier. Note that the true results of the 30 records are given in the original dataset for benchmarking. (**15 points**)

Submission Part I: (1) include the proof for differential privacy, screenshots of the procedures and results in the same report as above *hw2-report.pdf*, and (2) source code files – all named with the prefix "hw2-3-" (e.g., *hw2-1-classifier.java*, and *hw2-1-dpclassifier.java*).

2. Local Differential Privacy (35 points)

Using the same dataset as Homework 1: https://archive.ics.uci.edu/ml/datasets/Adult. The server tries to learn the distribution of the users' ages (each record is privately held by one user). However, each user doesn't trust the server and locally perturbs its age at the client side with the privacy bound ϵ.

Tasks:

(a) Implement the following two LDP protocols: Unary Coding, and Generalized Random Response for LDP. Note that each LDP protocol includes both private data collection via local randomization and the frequency estimation performed on the noisy data (by the data aggregator/server). It is unnecessary to implement the communication protocol for 48k+ users/clients and server (simulating the local randomization algorithm for all the users and the frequency estimation for the server would be fine). (**20 points**)

(b) Compare these protocols' accuracy with different ϵ and discuss your findings. The parameter ϵ varies from 1 to 10 with a step of 1. You can measure the relative error in age frequency estimation by calculating the L_1-distance between the actual and estimated frequencies. You can plot the results with x axis ϵ and y axis the L_1-distance. (**20 points**)

(c) Compare these protocols' accuracy with different numbers of users n and discuss your findings with a fixed $\epsilon = 2$. The number of users n varies from 10% to 100% of all the users with a step of 10%. You can measure the relative error in age frequency estimation by calculating the L_1-distance between the actual and estimated frequencies. You can plot the results with x axis n and y axis the L_1-distance. (**10 points**)

Submission Part II: (1) include the screenshots of the procedures and results in the same report as above *hw2-report.pdf* (you can explain your findings with tables or figures), and (2) source code files – all named with the prefix "hw2-2-" (e.g., *hw2-2-unary.java*).