CCS 2019 Talk Summaries

Sebastian Meiser, Visa Research

November 11, 2019

This document presents a few summaries of talks I attended at CCS 2019. I don't guarantee that the summaries are comprehensive or that they capture all subtleties of the presented matter; if you find any technical inaccuracies, please contact me about them.

1 Pre-Conference Workshop: TPDP

1.1 Encode, Shuffle, and Analyze (ESA) Revisited: Strong Privacy despite High-Epsilon

Speaker: Abhradeep Guha Thakurta, Google Research, UC Santa Cruz

Assume we have a number of devices use an anonymizer and local DP to achieve some central differential privacy. The anonymizer can either use summation or shuffling to achieve this goal. Naturally we focus on the second part here.

What we do is we take the locally DP objects, remove identifiers (if there are any), shuffle them, and release them. We want to start with weak local DP, i.e., with a high epsilon ($\varepsilon > 1$) and still achieve strong central differential privacy: We get a boost of about $\frac{1}{\sqrt{n}}$ for ε.

To this end, we look at three ideas:

- **Attribute fragmenting:** We split one-hot vectors into the separate bits, then shuffle them, and get some utility in $\Theta\left(\sqrt{\frac{\log k}{ne^{\varepsilon_{\text{local}}}}}\right)$. Note that if we have t records and a local $\varepsilon_{\text{local}} = 1$ we get local DP of $t \cdot \varepsilon_{\text{local}}$.
- **Record fragmenting:** I'm not quite sure what exactly happened here; I think the result is that instead of an ok central DP ($\varepsilon = 1.5$) trade-off that comes with a horrible local DP guarantee ($eps \approx 25$), we can have a local DP of $\varepsilon = 1$ and still get central DP with $\varepsilon = 1.5$. This degrades the utility, but Abhradeep assures us that it's not that bad.
- **Crowds:** We can group records to achieve a better local DP / utility trade-off. We split our data into crowds and analyze them. A cute idea here is to add Laplace noise $\text{Lap}\left(\frac{1}{\varepsilon_{\text{shuffle}}}\right)$ (and subtract a large enough

constant) to the count of records it has and then drop as many records as required to meet the count. If we still come up with a number higher than the actual count, we have a distinguishing event.

1.2 DPella

Speaker: Elisabet Lobo Vesga

We know how to do queries with DP and how to estimate the accuracy. However, what happens if we want to add the results of several queries?

In comes DPella, a Haskell library, which allows us to keep track the privacy and accuracy of the (combined) queries we ask and to find out the privacy budget used by a program (via symbolic execution). Similarly, we can get an estimate of the accuracy of the program. That sounds pretty interesting.

So far, they only consider the Laplace mechanism, but they are working on integrating the Gauss mechanism as well.