Differential Privacy : US Broadband Data

Trishita Patra (MDS202440)

Chennai Mathematical Institution

June 1, 2025

Motivation

- Protecting privacy while performing statistical analysis is quite challenging.
- On one had, the goal of statistics and machine learning is to be as informative as possible. Protecting privacy is the opposite goal.
- Can we protect privacy and still do an informative analysis?
- Can you mathematically quantify privacy?

US Internet Usage Data

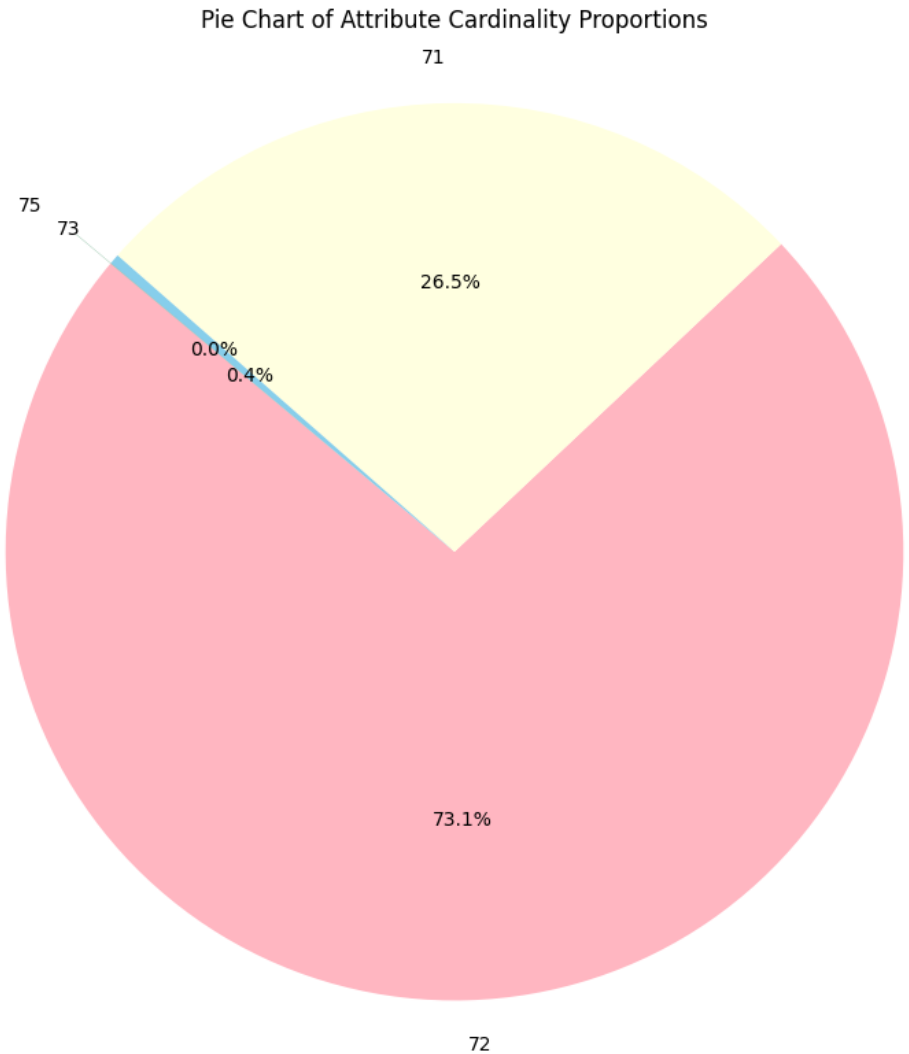
- This dataset contains general demographic information on Internet users in 1997.
- 70 variables covering:
  - Demographics: Age, Income, Education, Disability
  - Usage Patterns: Hours online, Activities, Purchase behavior
  - Attitudes: Censorship opinions
- Data files:
  - final-general.dat contains demographics of internet usage.
  - final-general.col contains the variable names for data.
  - changes contains the conversions from character data to numeric.
- Pre-processed numeric encoding (changes file documents categorical$\rightarrow$numeric mappings)

To Do

- Data Preprocessing
- Identifying PII,QI,Sensitive attributes - Domain Knowledge!
- Developing a suitable Differential Privacy Mechanism.

Column Discrepancy

- Number of column names: 70
- Total fields in first row: 72



Handling Column Discrepancy

- Since percentage of rows with recorded attribute values > 72 values is very less (0.3761 percent) and less likely to cause substantial loss in understanding the pattern/characteristics of the whole dataset, we eliminate these data points.
- Table segregation for analysis:
- df1 (71 columns): (2676, 71)
- df2 (72 columns): (7390, 72)

Handling Column Discrepancy

- Checked for duplicate columns : Absent!
- Guesstimating missing attribute names!

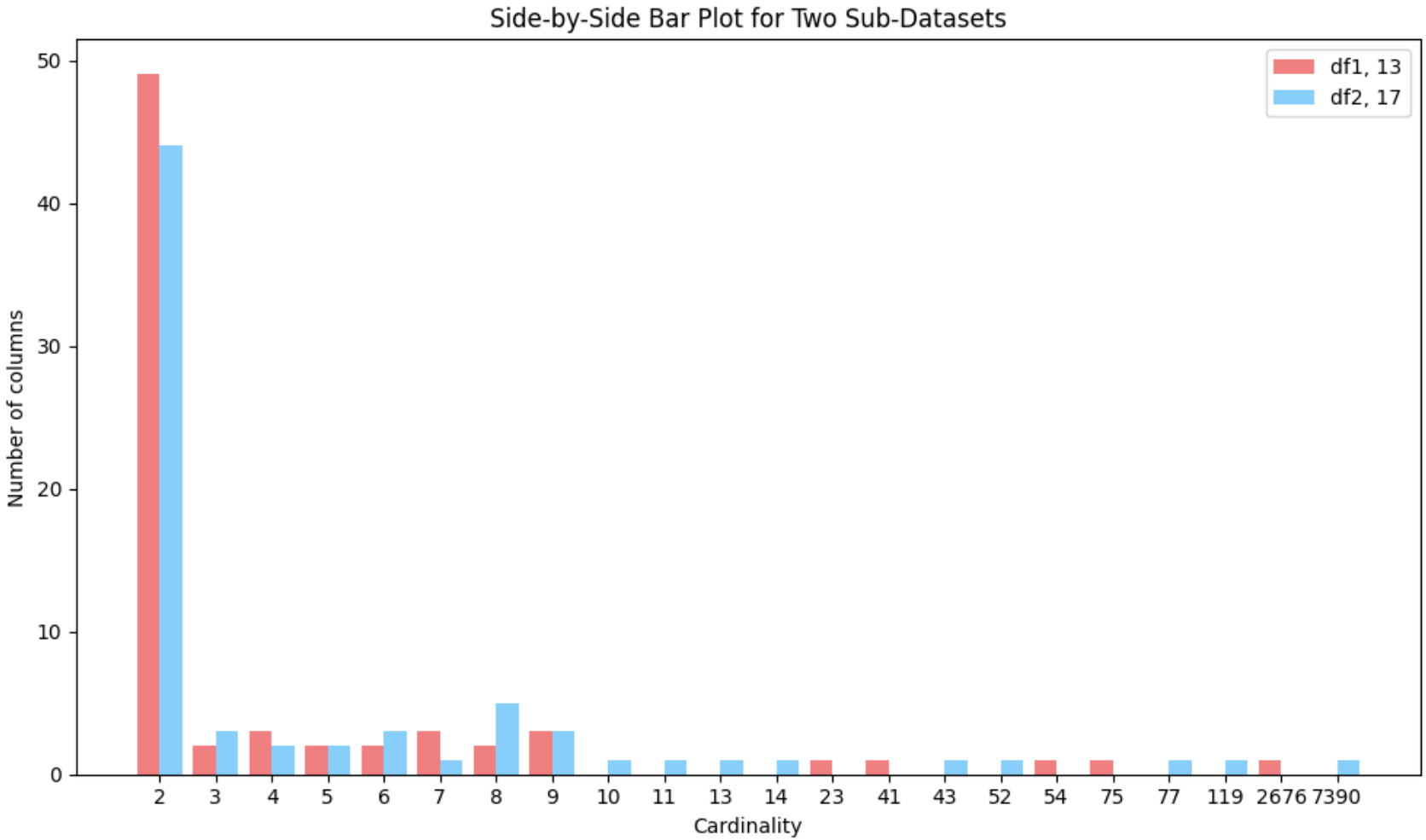
Guesstimate!

- Looked at Column cardinality.
- In both datasets, max column cardinality = no. of datapoints >
- Unique ID identified!
- Column indices found. 71 attributes!

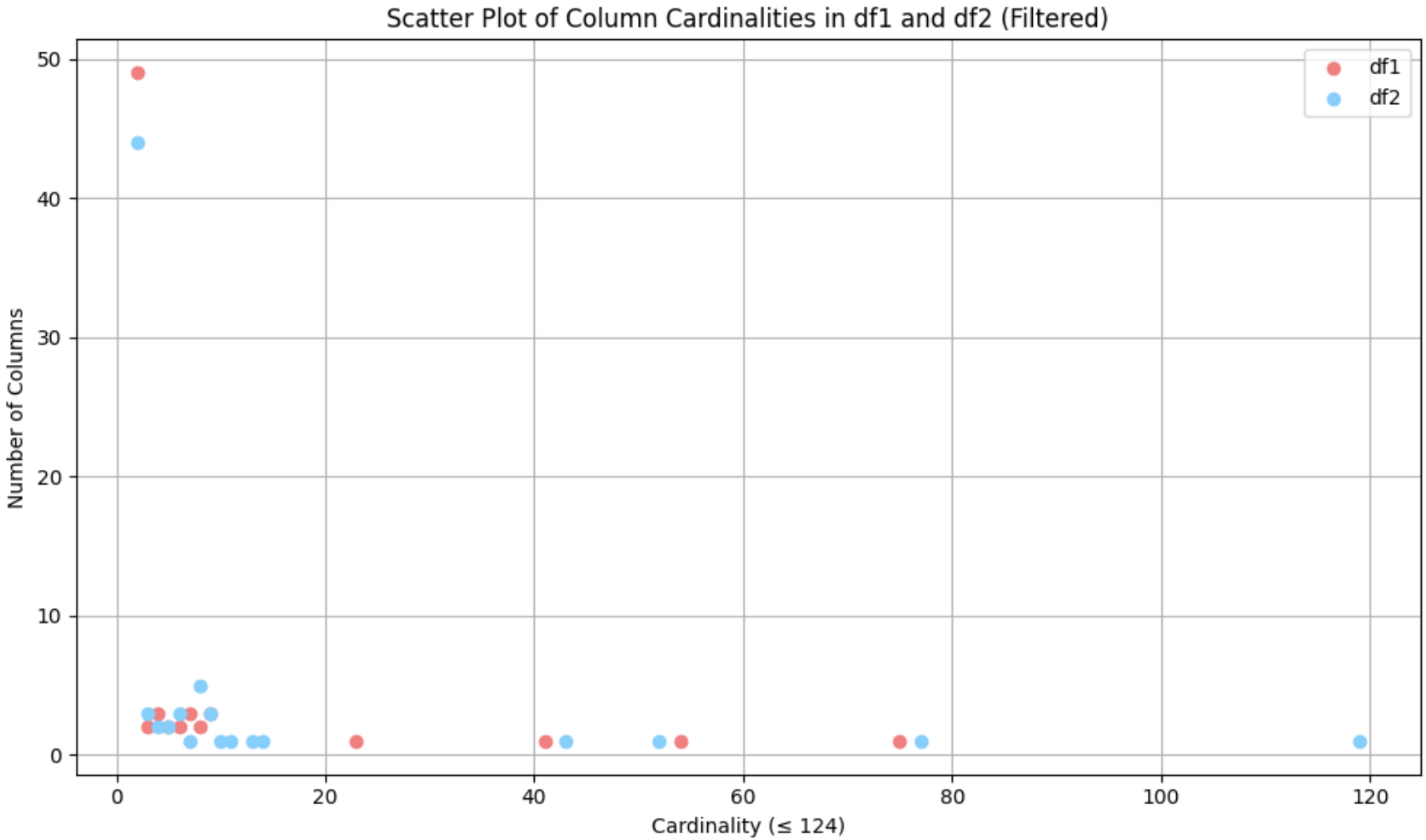
Guesstimate : The Big Assumptions

- With respect to df2, df1 has missing values from a single column only.
- The underlying cause of the missing values do not affect other attributes. i.e. if Column A is taking n many values in df1, it is taking n (almost n many) values in df2 as well.

Guesstimating the unknown column of df2



Guesstimating the unknown column of df2



Guesstimate

- Now see that there are total five column cardinality > 45, but < 124. With our assumption, it is reasonable that following two column cardinality pairs in (df1,df2) correspond to same recorded attribute : (54,52) and (75,77)
- column cardinality 119 (which has to be states) in df2 is the unknown one!
- column dropped from df2.
- dimensionality matched ! (10066, 71)

Differential Privacy

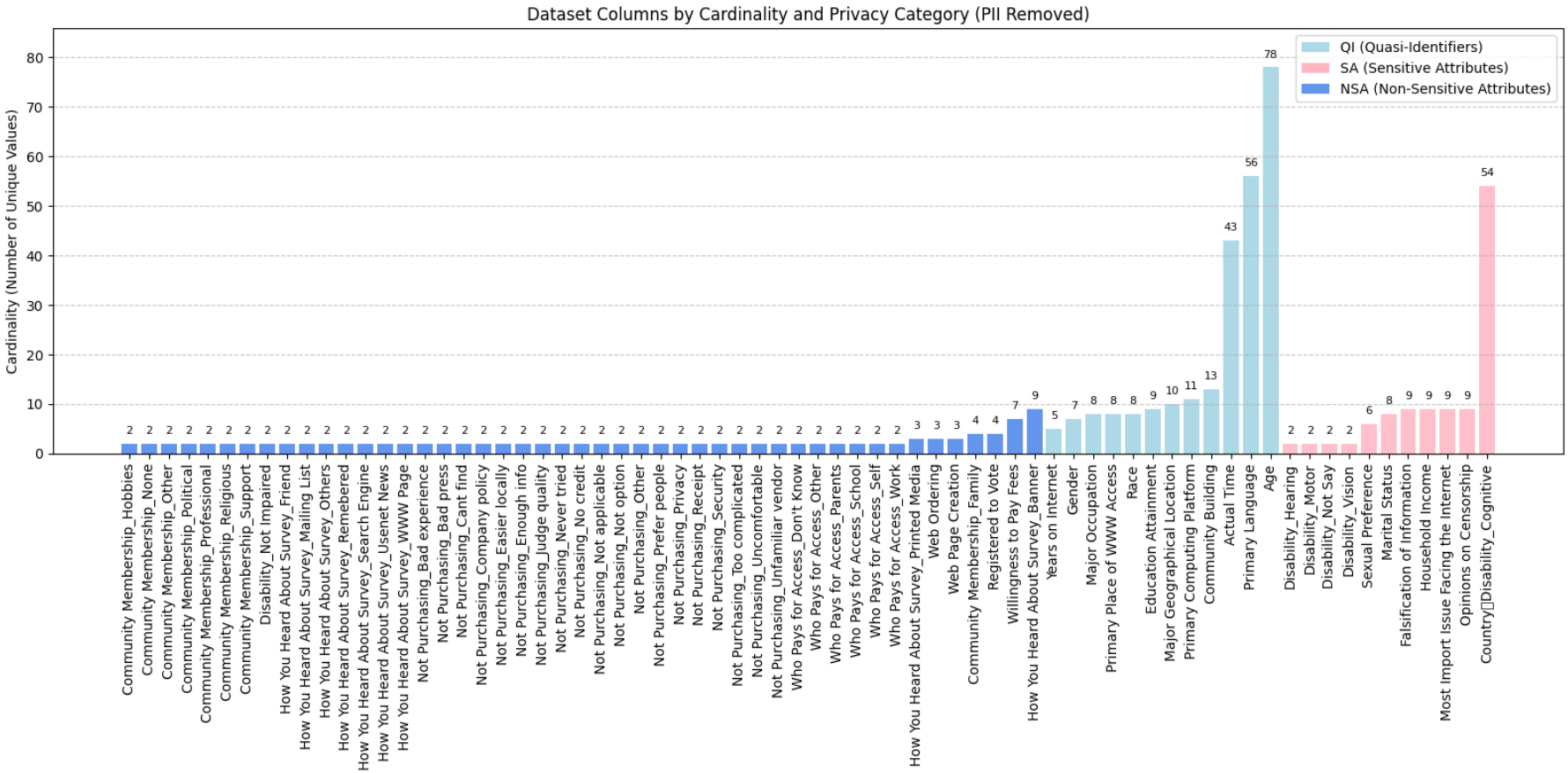
- Differential privacy mathematically guarantees that anyone seeing theresult of a differentially private analysis will essentially make the same inference about any individual's private information, whether or not that individual's private information is included in the input to the analysis.
- It works by adding noise to/randomizing the sensitive values to protect privacy, while maximizing the accuracy of queries.
- Domain knowledge helps us to predict which queries might be executed on the dataset.

Identifying Column Types

Based on domain Knowledge and column cardinality, we determine

- PII (Personally Identifiable Information) : Attributes that can uniquely or directly identify an individual.
- Quasi-Identifiers (QI) : Attributes that don't identify directly but can be combined to do so.
- Sensitive Attributes : Attributes a user would consider private or are policy-sensitive.
- Non-sensitive : None of the above.

Attribute Distribution

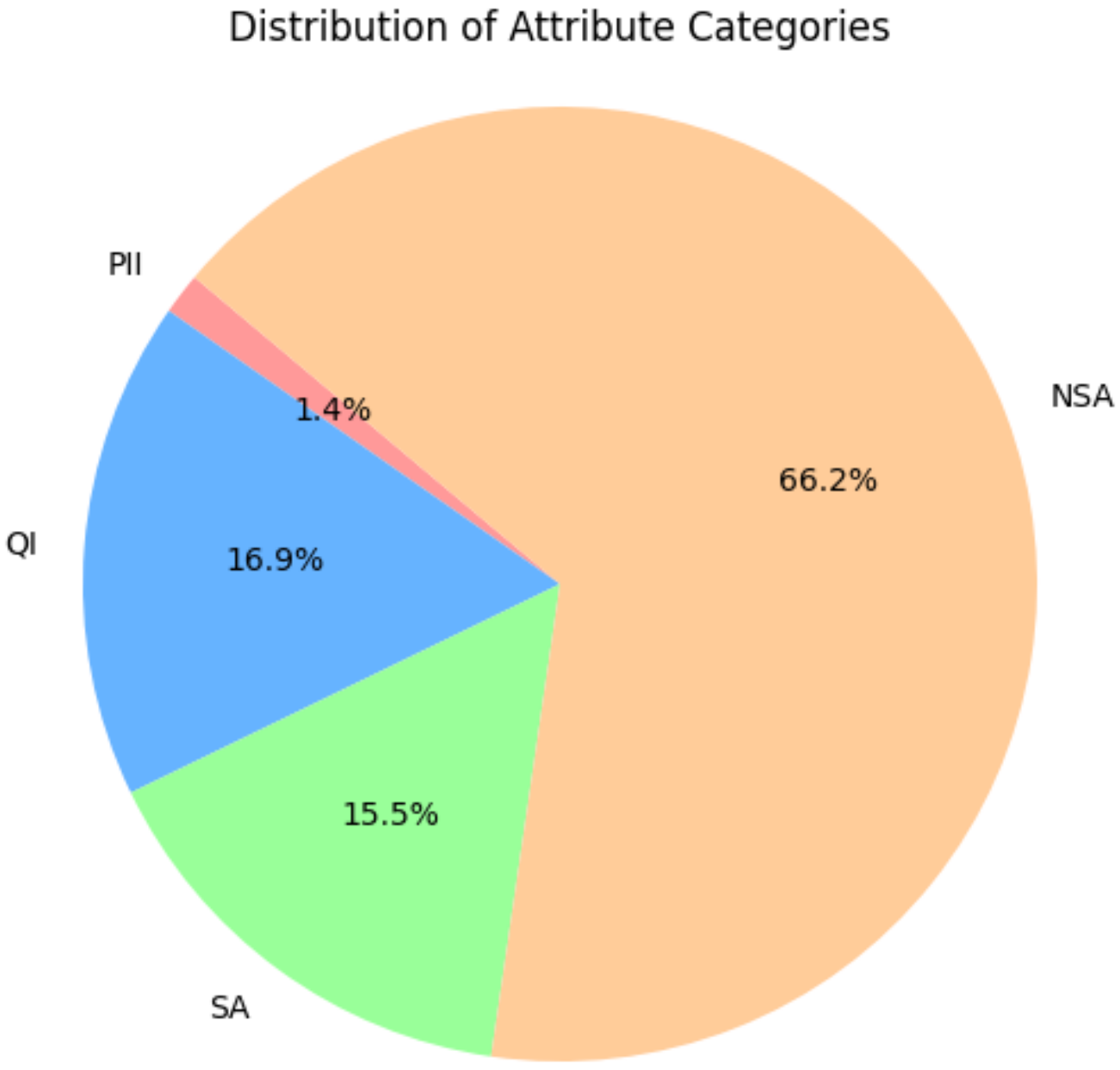


Attribute Distribution (n=71)

Category	Count (%)
PII	1 (1.4%)
Quasi-Identifiers (QI)	12 (16.9%)
Sensitive Attributes (SA)	11 (15.5%)
Non-Sensitive (NSA)	47 (66.2%)

- **PII**: Removed entirely
- **QI**: Generalization/bucketing applied
- **SA**: Differential privacy mechanisms
- **NSA**: Preserved as-is

Attribute Distribution : Ratio



Randomized Response Implementation

Core Mechanism

1. For each sensitive column:
   - Calculate truth probability:

$$p = \frac{e^{\epsilon}}{e^{\epsilon} + k - 1} \quad (k = \#\text{ categories})$$

2. Generate perturbation mask: [language=Python] mask = np.random.random(len(df)) ¡ p True=keep, False=perturb
3. Apply randomization: Among lies

Key Properties

- Satisfies ϵ-DP: $\frac{p}{q} = e^{\epsilon}$ where $q = \frac{1-p}{k-1}$
- Runtime: $\mathcal{O}(n)$ per column
- Space: $\mathcal{O}(1)$ beyond original data

Randomized Response Privacy Guarantees

DP Proof: For any $x, y \in$ domain:

$$\frac{\Pr[\text{Output} = x | \text{True} = x]}{\Pr[\text{Output} = x | \text{True} = y]} = \frac{p}{q} = e^{\epsilon}$$

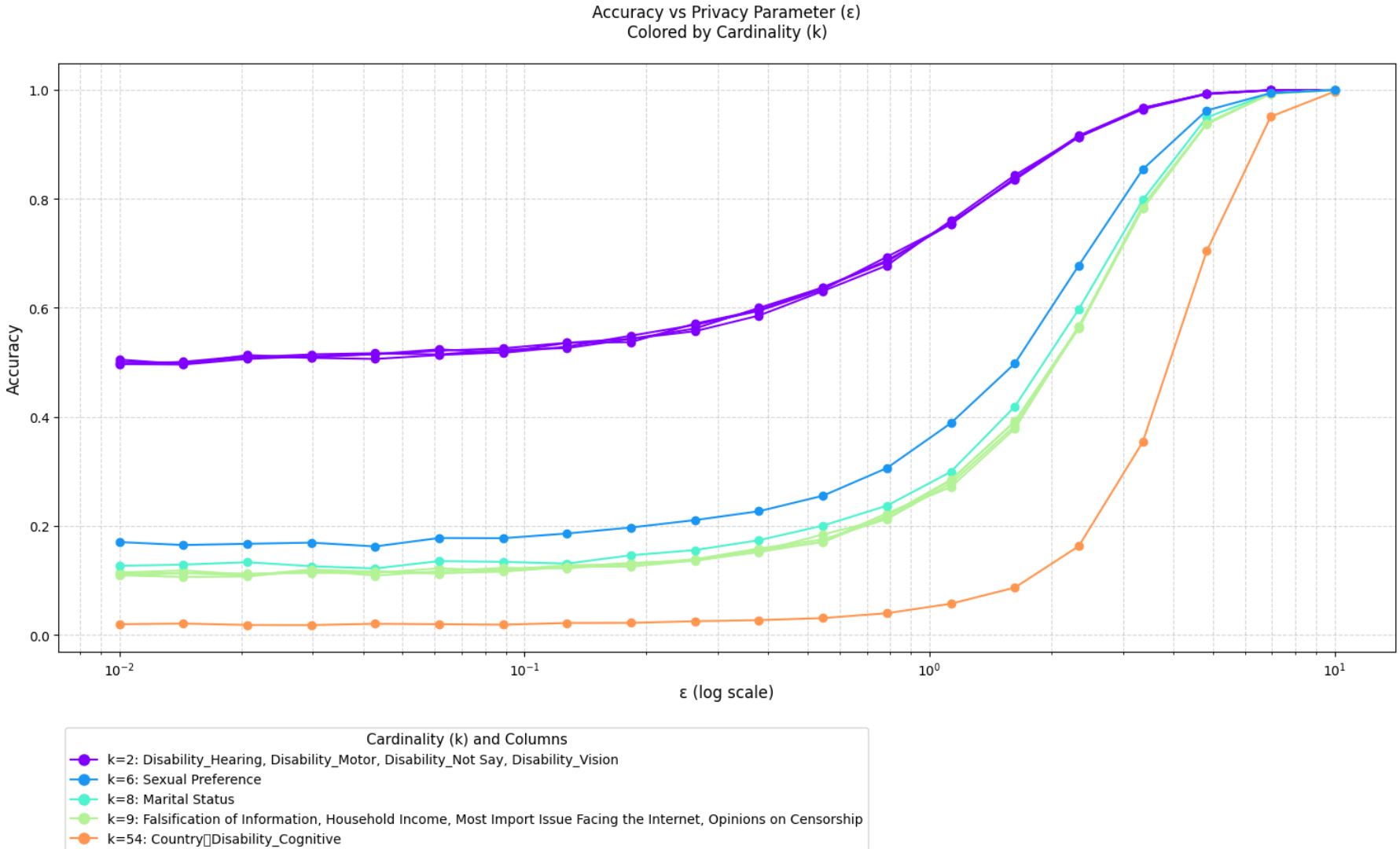
where:

- $p = \frac{e^{\epsilon}}{e^{\epsilon}+k-1}$ (truth probability)
- 1-p is total lie probability
- $q = \frac{1}{e^{\epsilon}+k-1}$ (per-lie probability)

Intuition:

- $\epsilon \to 0$: $p \to \frac{1}{k}$ (max privacy)
- $\epsilon \to \infty$: $p \to 1$ (no privacy)

Accuracy vs. Privacy



Key Observations

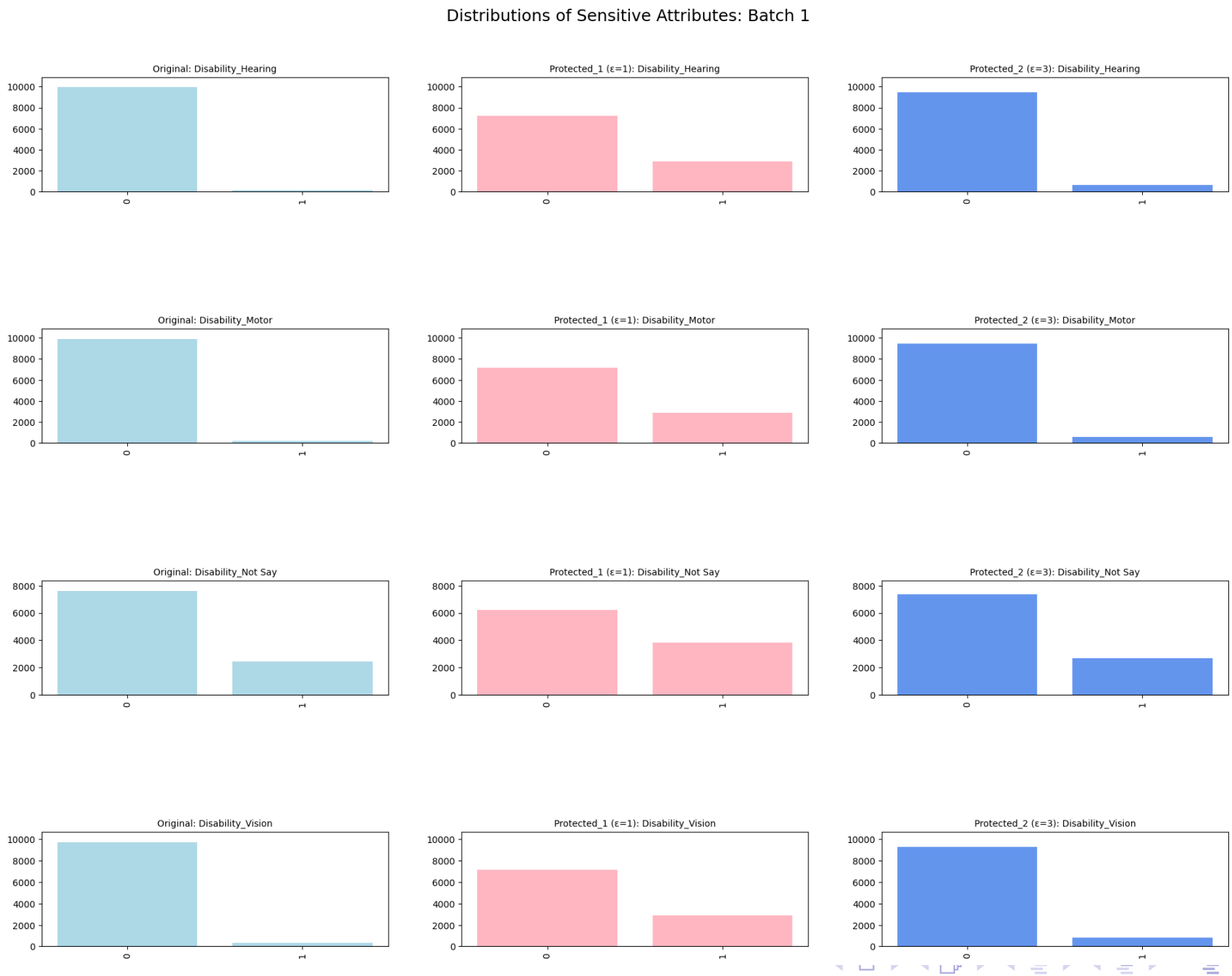
Privacy vs. Accuracy Tradeoff

- Higher values of ϵ (less privacy) lead to higher truth probability p, resulting in higher accuracy
- Lower values of ϵ (stronger privacy) increase randomization, leading to lower accuracy

Impact of Cardinality (k)

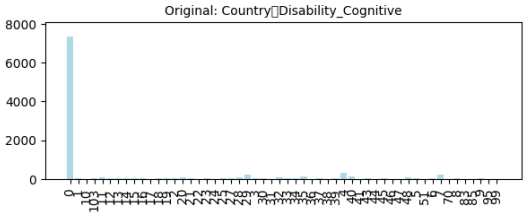
- Fewer categories (lower k) yield higher p for the same ϵ, producing higher accuracy
  - Example: Binary data ($k = 2$) is easier to protect than 10-category data
- More categories (higher k) require larger ϵ values to maintain usable accuracy

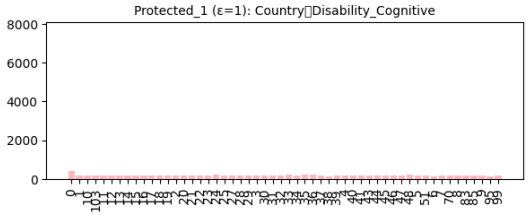
Comparison of Distribution of Sensitive Attributes

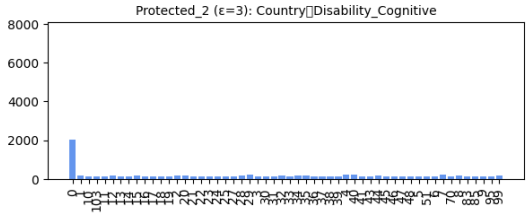


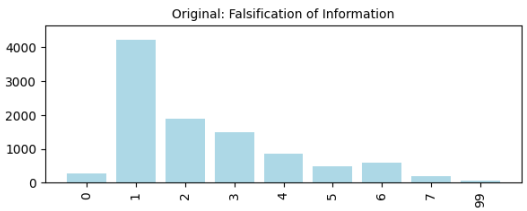
Comparison of Distribution of Sensitive Attributes

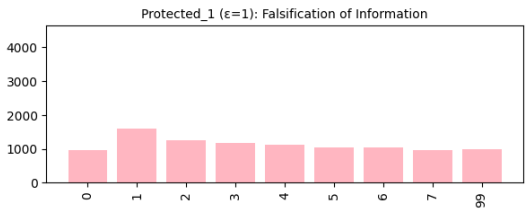
Distributions of Sensitive Attributes: Batch 2

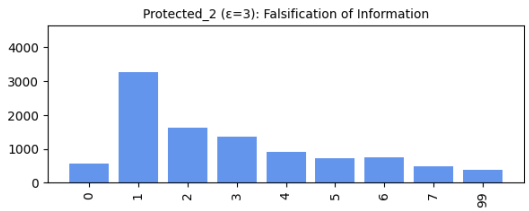


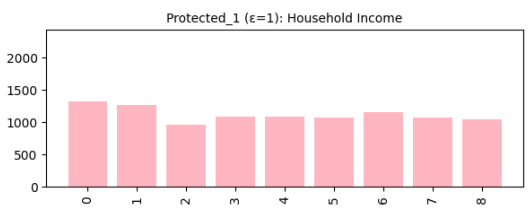


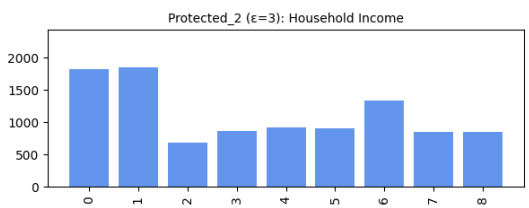


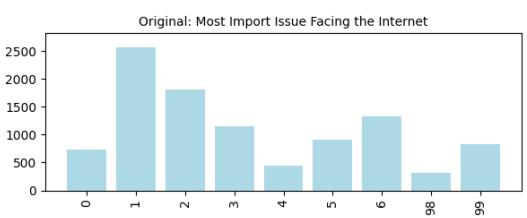


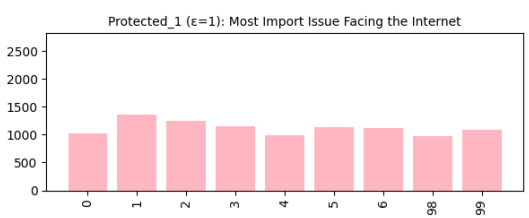


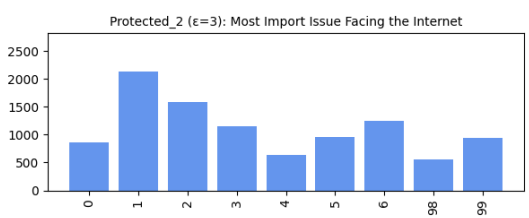




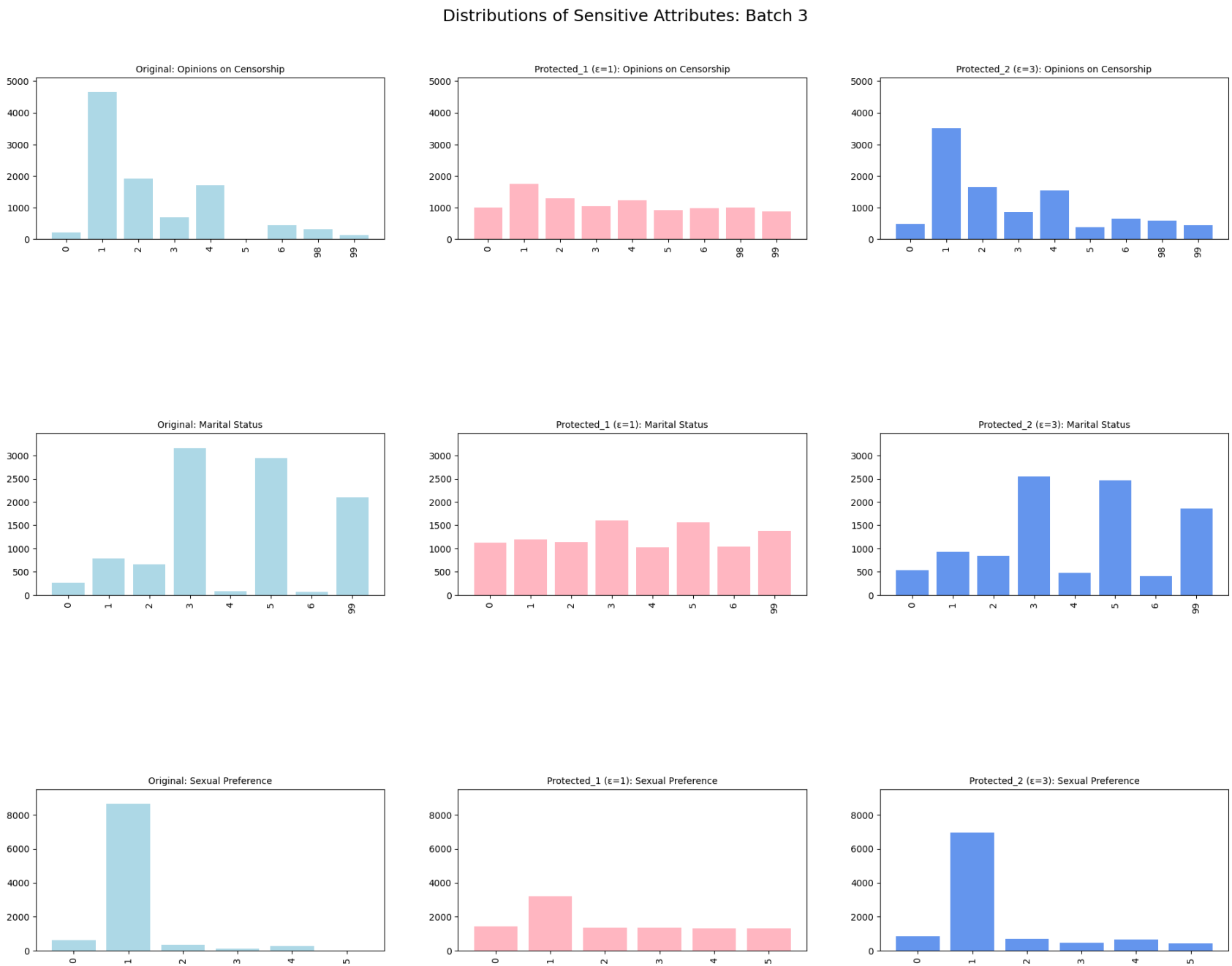








Comparison of Distribution of Sensitive Attributes



Conclusion

Differential privacy via randomized response provides a mathematically rigorous way to trade accuracy for privacy in categorical data. The key parameters are:

- Privacy parameter (ϵ): Controls the noise-accuracy balance.
- Cardinality (k): Determines how much noise is needed.

Sensitive Attribute Distribution

- With increasing ϵ, the protected data distribution converges to the original distribution
- This convergence occurs regardless of attribute cardinality

Implementation Guidance

- Preprocess high-k features (bucket/group)
- Choose ϵ based on data sensitivity

By carefully selecting ϵ and preprocessing high-cardinality features, practitioners can achieve meaningful privacy guarantees while preserving data utility.

Thank You!