Statistical Confidentiality Using Homomorphic Encryption

Yeşem Kurt Peker

Rahul Raj

Columbus State University, Columbus, GA

2025 CAE Community Symposium

2025 CAE Community Symposium

- 20250214_CAE-CDSymposiumPresentationSlides111.pptx

Statistical Confidentiality

- Protection of data collected for statistical purposes.
- Ensures individual privacy while enabling aggregated insights.

2025 CAE Community Symposium

Mechanisms for Statistical Confidentiality

Statistical Dara Disclosure Control

- Data Masking
- Dara Anonymization
- Synthetic Data
- Access Control
- Encryption

2025 CAE Community Symposium

US Census Bureau

Disclosure Avoidance 2020

"Modern computers and today's data-rich world have rendered the Census Bureau's traditional confidentiality protection methods obsolete. Those legacy methods are no match for hackers aiming to piece together the identities of the people and businesses behind published data."

"The 2020 DAS is based on a framework for assessing privacy risk known as differential privacy. It is the only solution that can respond to this threat while maximizing the availability and utility of published census data."

https://www.census.gov/programs-surveys/decennial-census/disclosure-avoidance.2020.html#list-tab-1042747205

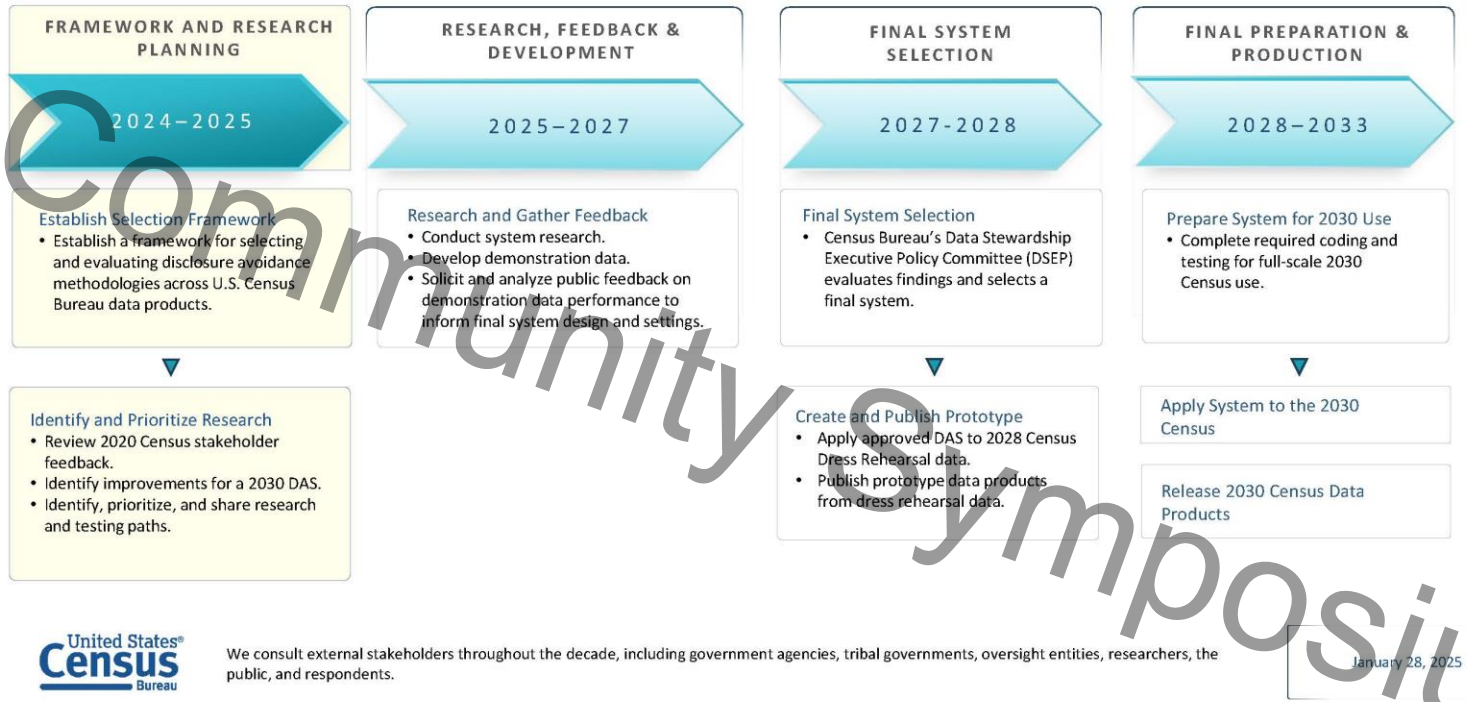
2025 CAE Community Symposium

United States Census Bureau

Disclosure Avoidance (2030)

2025 CAE Community Symposium

2030 Disclosure Avoidance System (DAS) Planning Timeline



United States Census Bureau

We consult external stakeholders throughout the decade, including government agencies, tribal governments, oversight entities, researchers, the public, and respondents.

Homomorphic Encryption

2025 CAE Community Symposium

- A form of encryption that allows specific types of computations to be carried out on ciphertexts and generates an encrypted result which, when decrypted, matches the result of operations performed on the plaintexts.

$$x, y: \textit{plaintexts}$$

$$E(x + y) = E(x) \hat{+} E(y)$$

$$E(x * y) = E(x) \star E(y)$$

Third Party Computations

2025 CAE Community Symposium

Goal: Compute privately $f(x, y)$ for plaintext x, y where f is a function that involves addition and multiplication (and subtraction).

Usual process:

1. Encrypt the data and send it to the computing party: $E(x), E(y)$
2. Data user decrypts the data: $D(E(x)), D(E(y)) \rightarrow x, y$
3. Computes $z = f(x, y)$
4. Encrypts the result and sends the ciphertext to the owner $t = E(z)$
5. The owner decrypts the ciphertext to get the result $D\big(E(t)\big) = z$

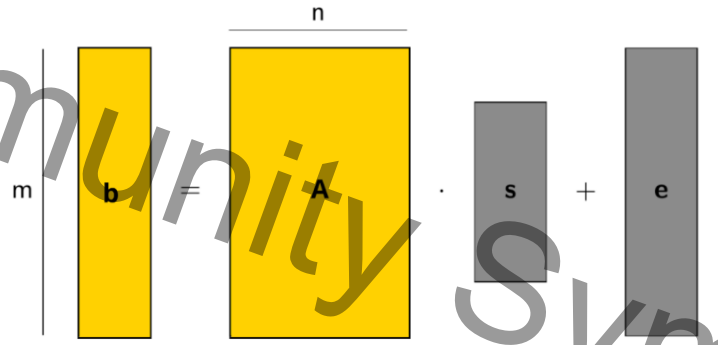
With Homomorphic Encryption:

1. Encrypt the data and send it to the computing party: $E(x), E(y)$
2. Data user computes $\mathcal{F}$ on the encrypted data and sends the ciphertext to the owner: $t = \mathcal{F}((E(x), E(y))$
3. The owner decrypts the ciphertext to get the result $D(t) = f(x, y)$

TFHE

Fully Homomorphic Encryption over the Torus

- *"Faster fully homomorphic encryption: Bootstrapping in less than 0.1 seconds"*, Ilaria Chillotti, Nicolas Gama, Mariya Georgieva and Malika Izabachène (Asiacrypt, 2016)
- LWE – Learning With Errors



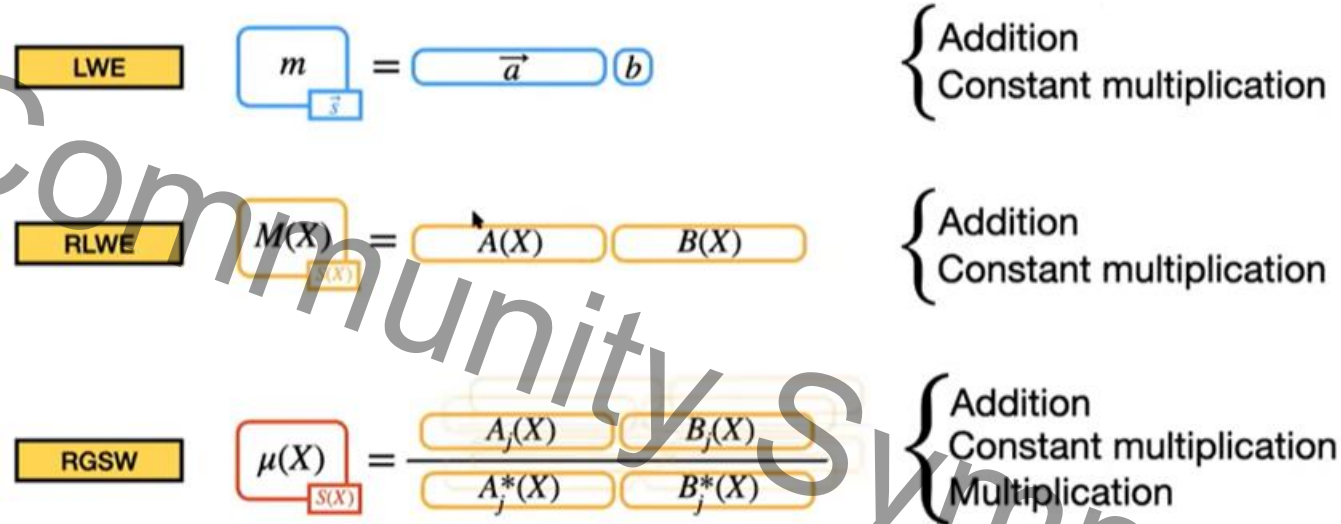
https://www.zama.ai/post/estimating-the-security-of-homomorphic-schemes

https://www.zama.ai/post/tfhe-deep-dive-part-1

2025 CAE Community Symposium

2025 CAE Community Symposium

TFHE Ciphertexts



Concrete

- Compiler by Zama that compiles Python code into its FHE equivalent
- Implements a variant of TFHE that enables both leveled and fast bootstrapped operations as well as approximate or exact evaluation of arbitrary functions, on both booleans and integers.
- Allows writing code in Python and supports some operations of *numpy* library. Example:
  - Square
  - Sum
  - Min
  - Max

2025 CAE Community Symposium

Limitations

- Branching (if-else conditions)
- Division
- Negative numbers
- Decimal numbers

2025 CAE Community Symposium

Computations

- We're performing the following statistical operations:
  - Mean
  - Median
  - Variance
  - Min/Max

2025 CAE Community Symposium

Min/Max

- Returns the minimum and maximum value of an array
- We're using `min()` and `max()` methods provided by `numpy` library to return the min and max values

2025 CAE Community Symposium

Mean

- Using `sum()` method provided by the `numpy` library to calculate the sum of all the elements of the array.
- The encrypted sum is returned, decrypted and divided by length of array to get the mean because division of encrypted numbers is not supported by concrete.

2025 CAE Community Symposium

Median

- Sort the array to find the median (We implemented bubble sort)
- Used the following trick to overcome the branching limitation:
  - c ? t : f = f + c * (t - f)
    - Here c is the condition, t is the value to be returned if condition is true, f is the value to be returned if condition is false
- If length of array is even, we return encrypted sum of middle two numbers.
  - To find the median the sum is decrypted and divided by 2
- If length of array is odd, we simply return the middle element.

2025 CAE Community Symposium

2025 CAE Community Symposium

Conditionals with Ciphertexts

```
if array[j] > array[j+1]:
        temp = array[j]
        array[j]=array[j+1]
        array[j+1]=temp
```

```
@fhe.compiler({"array": "encrypted"})
def sort_array(array):
    n = array.size
    for i in range(n):
        for j in range(0, n - i - 1):
            c = array[j] > array[j+1]
            temp = array[j]
            array[j] = array[j] + ((array[j+1] - array[j]) * c)
            array[j + 1] = array[j + 1] + ((temp - array[j + 1]) * c)
    return array

inputset = [np.random.randint(0, MAX_VALUE, size=MAX_SIZE) for _ in range(50)]
circuit = sort_array.compile(inputset, compress_evaluation_keys=True, compress_input_ciphertexts=False)
```

Variance

$$V = \frac{\sum(x_i - \bar{x})^2}{n-1}$$

- Used a derived formula for variance to overcome the limitations (division).

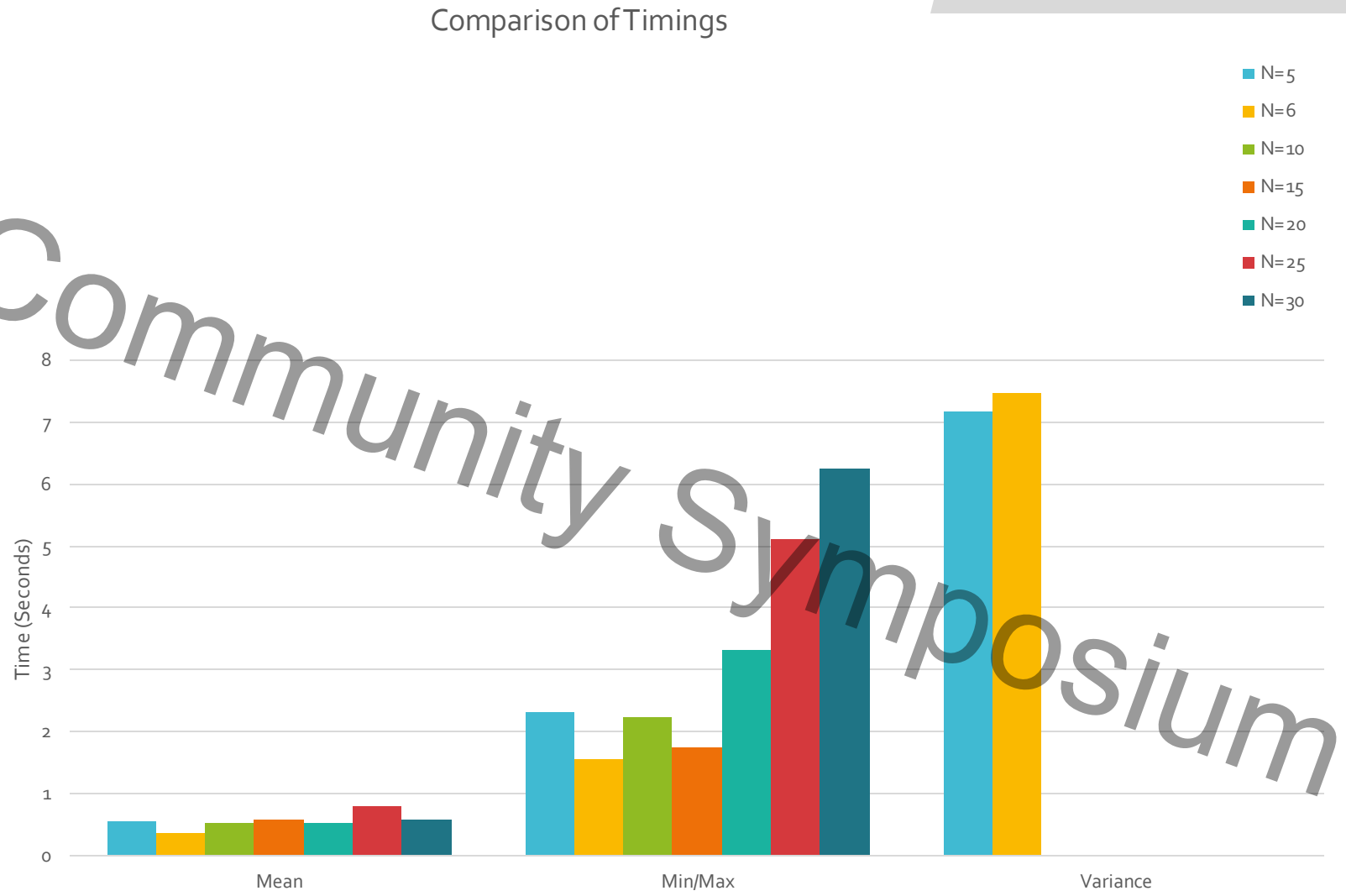
$$\frac{n^2 \sum_{i=1}^{n} x_i^2 - 2n \sum_{i=1}^{n} (x_i \sum_{j=1}^{n} x_j) + n(\sum_{j=1}^{n} x_j^2)}{n^2(n-1)}$$

- The numerator is calculated in encrypted form, decrypted and divided by the denominator to get the result.

2025 CAE Community Symposium

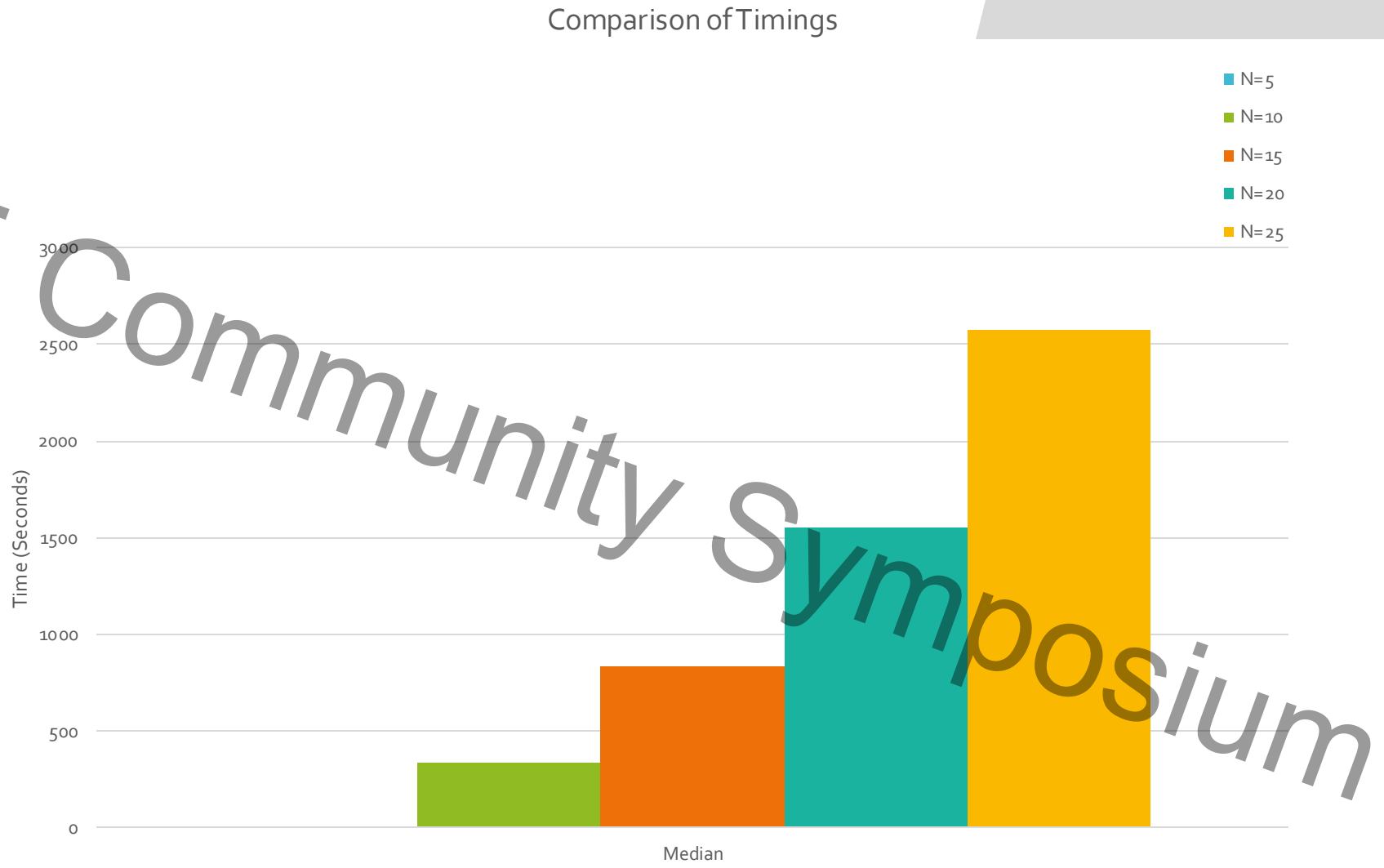
2025 CAE Community Symposium

Results



Results

2025 CAE Community Symposium



Conclusion

- It is possible to perform some statistical operations on encrypted data.
- Not feasible for larger data sets especially when performing complex operations such as sorting.
- Need to develop algorithms that work with the limitations.
- Not practical for real world applications with the current resources.

2025 CAE Community Symposium

Future Work

- Calculate median without sort
- Consider other sorting algorithms

2025 CAE Community Symposium

References

- [1] C.3. Statistical Confidentiality—MSITS 2010 Compilers Guide—UN Statistics Wiki. Available online: https://unstats.un.org/wiki/display/M2CG/C.3.++Statistical+confidentiality (accessed on 25 Feb 2025).
- [2] X. Yi, R. Paulet, and E. Bertino, "Homomorphic encryption" in Homomorphic Encryption and Applications, Springer, 2014, pp. 27–46. [Online]. Available: https://doi.org/10.1007/978-3-319-12229-8_2
- [3] Data Access and Personal Privacy: Appropriate Methods of Disclosure Control, American Statistical Association, 2008
- [4] Decennial Census of Population and Housing Disclosure Avoidance, US Census Bureau, 2020

2025 CAE Community Symposium