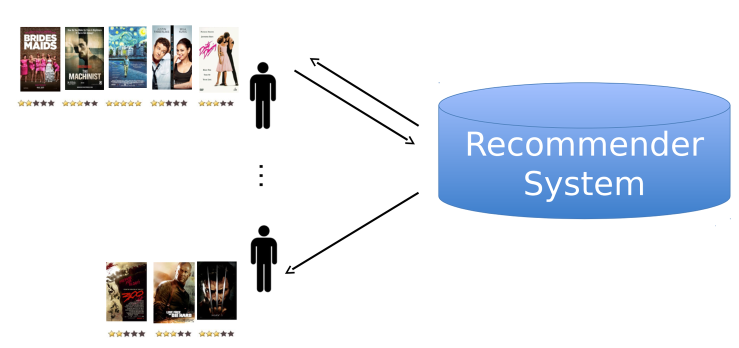
Privacy-Preserving Ridge Regression on Hundreds of Millions of Records

Valeria Nikolaenko with Udi Weinsberg, Stratis Ioannidis, Marc Joye, Nina Taft and Dan Boneh
Stanford University and Technicolor

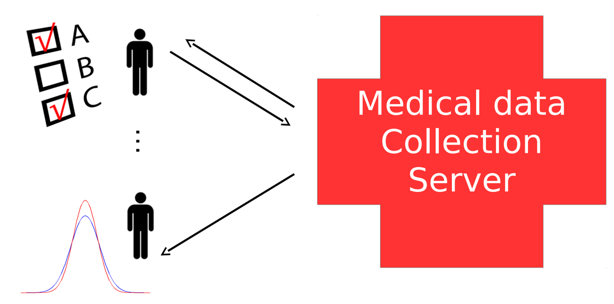
Motivation

Goal: compute best-fit curve for a large number of data points revealing no other information about the input.

Recommendation system
- Users submit encrypted ratings
- Users get recommendations



Medical research
- Users submit encrypted surveys
- Researcher learns a statistical model



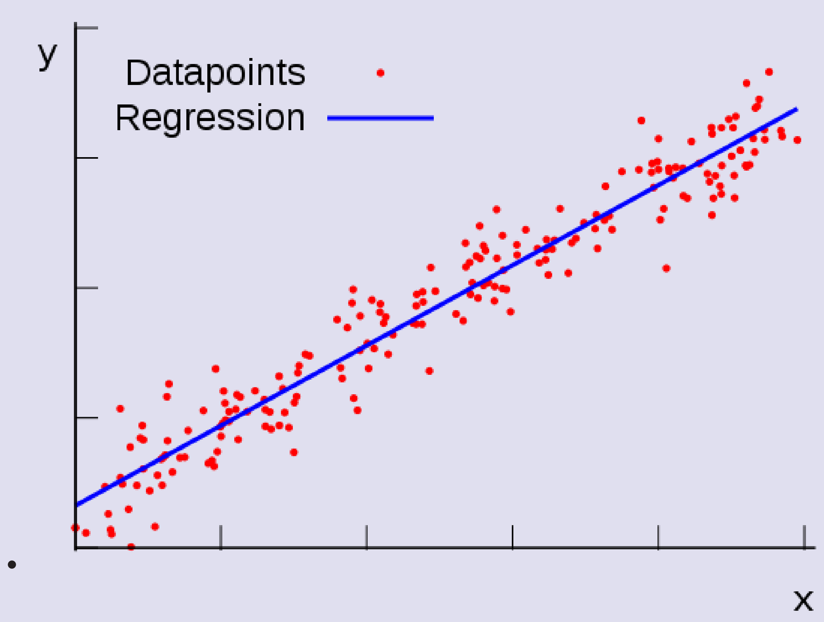
Summary

- **Contributions:**
  - **Built a system to compute ridge regression preserving privacy (find best-fit curve for a collection of encrypted data points).**
  - **Archieved excelent scalability in the number of users (1,000,000 users).**
  - **Showed good performance on real datasets.**
- **Challenges:**
  - Method should be efficiently represented with a boolean circuit.
  - All operations should be data oblivious.
- **Tools:**
  - Yao's garbled circuits.
  - Homorphic encryption.

Ridge Regression

- Given n pairs of variables $(x_i, y_i) \in (\mathbb{R}^d \times \mathbb{R})$, learn a linear function $f : \mathbb{R}^d \to \mathbb{R}$, s.t. $y_i \approx f(x_i)$
- $f(x) = \langle \beta, x \rangle$, where $x, \beta \in \mathbb{R}^d$
- β is fit to the data by minimizing the following quadratic form:

$$F(\beta) = \sum_{i=1}^{n} (y_i - \beta^T x_i)^2 + \lambda \|\beta\|_2^2 .$$



Cholesky Decomposition

- The minimizer, β, can be computed by solving the linear system $A\beta = b$, where $A = X^T X + \lambda I \in \mathbb{R}^{d \times d}$ and $b = X^T y \in \mathbb{R}^d$

$$y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} \quad \text{and} \quad X = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1d} \\ x_{21} & x_{22} & \cdots & x_{2d} \\ \vdots & \vdots & & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nd} \end{pmatrix} .$$

- **A** is Symmetric Positive Definite and efficient solution can be found through Cholesky decomposition

Cholesky Decomposition

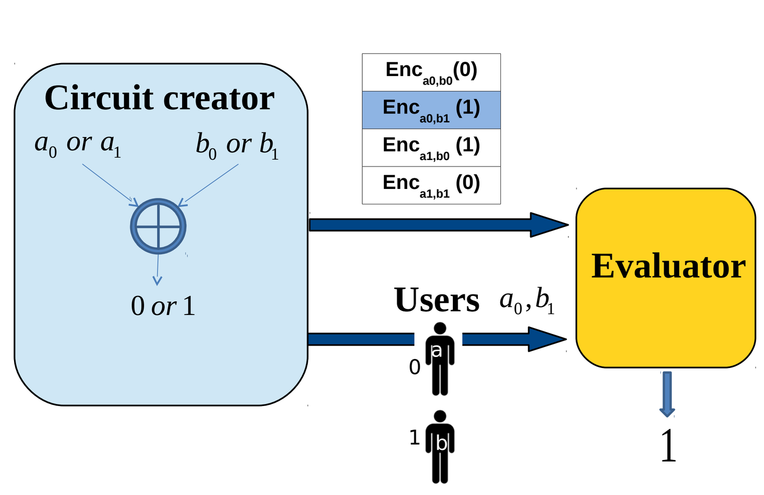
Computes lower triangular matrix L, s.t. $A = L^T L$:
- $\Theta(d^3)$ additions
- $\Theta(d^3)$ multiplications
- $\Theta(d^2)$ divisions
- $\Theta(d)$ square roots

Solving Linear System

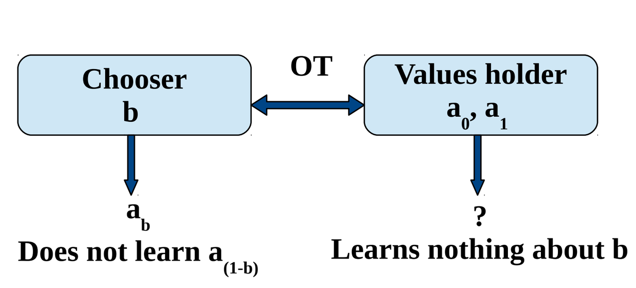
Input: L, b, s.t. $A = L^T L$
Output: x, s.t. $Ax = b$
Solve triangular system: $L^T y = b$
Solve triangular system: $Lx = y$
Return x

Cryptographic primitives

Garbled circuits:



Oblivious transfer (OT):

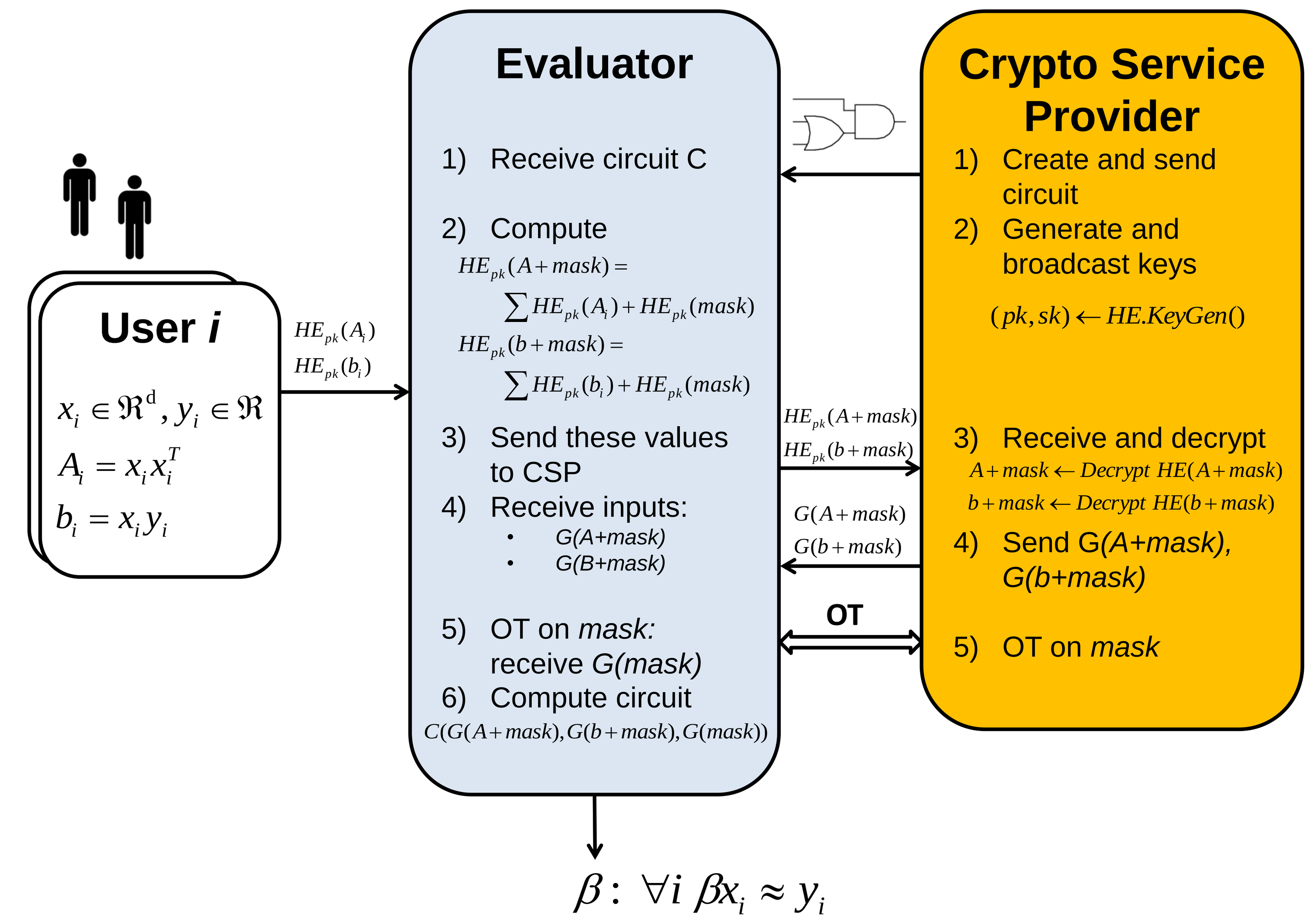


Homomorphic Encryption:

$$HE(a) \bullet HE(b) = HE(a + b)$$

Allows to carry one type of operation on the ciphertexts.

System design



Threat model

- CSP and Evaluator are honest, but curious.
- CSP is trusted not to collude with other entities.

Implementation

- Built on top of FastGC library.
- Support for fixed point numbers, representation is given by

$$[a] = \lfloor a \cdot 2^p \rfloor , \quad \text{where the exponent } p \text{ is fixed} .$$

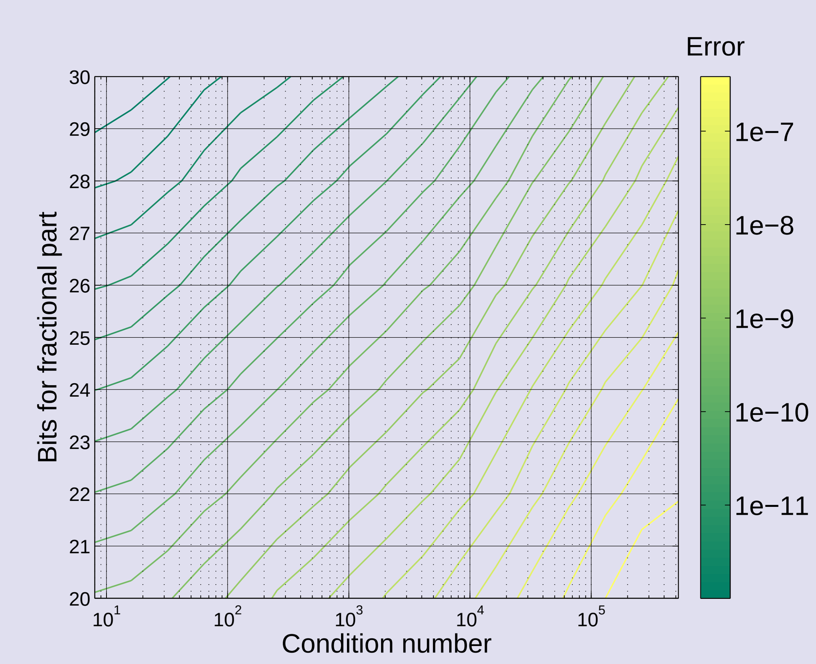
- Paillier homomorphic system included.

Accuracy

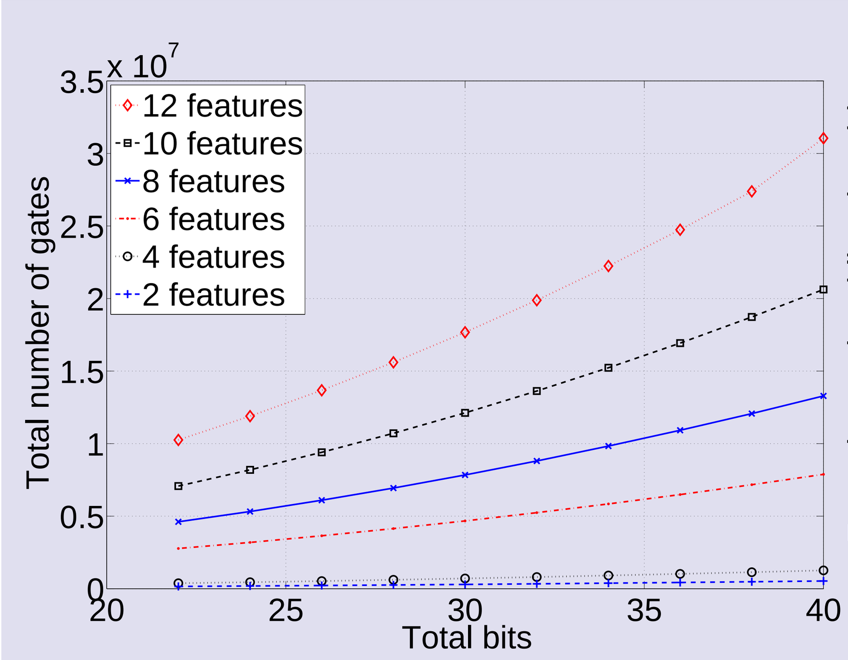
- The error of our system:

$$Err_{\beta^*} = \left| \frac{F(\beta) - F(\beta^*)}{F(\beta^*)} \right|$$

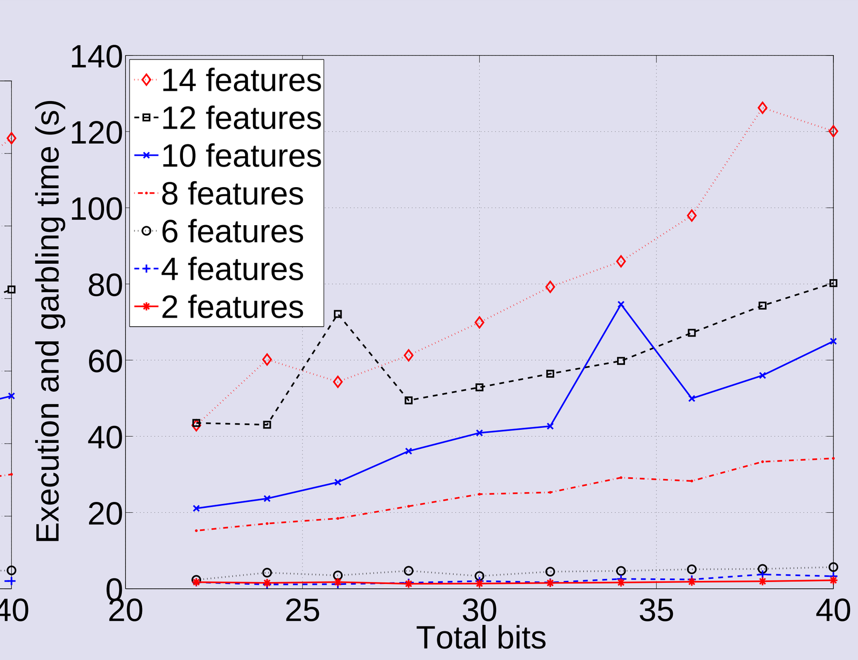
- β^* – true solution (computed in Matlab)
- Fig. shows tradeoff between
  - number of bits
  - condition number
  - error rate



Performance Evaluation



(a) Number of gates in the circuit



(b) Time to evaluate the circuit

Experimental results using UCI datasets, target error 10^{-5}

Name	n	d	Bits	Comm. (MB)	Time (s)
automobile	205	14	31	189	100
communities	1994	20	21	234	122
concrete strength	1030	8	19	27	17
forest fires	517	12	23	83	46
insurance	9822	14	21	102	55
flare	323	20	17	170	92
...	...	...	...	...	...

Conclusion

- On commodity server a regression model for 100 million user records, 20 features, runs in 8.75 hours.
- Further research: matrix factorization, support vector machines, logistic regression, *etc.*