Assignment 1

CS771

Assignment Report

CS771 Squad

Ashish Kumar
210210

Aryan Jalkote
210463

Atishay Jain
210233

Om Kothawade
210682

Raj Agrawal
210809

Yash Suryavanshi
211196

Abstract

This document is the submission of group CS771 Squad for Assignment 1. We have answered the parts 1 and 3 with all the relevant level of detail (proofs and tables) required.

PART 1

First, we consider the behavior of a single arbiter PUF. The delay time for a signal to traverse this PUF can be represented as:

$$t = \sum_{i=1}^{k} (x_i \cdot d_{i,u} + (1 - x_i) \cdot d_{i,l}) \tag{1}$$

where x_i is the i-th bit of the input challenge, and $d_{i,u}$ and $d_{i,l}$ are the delays on the upper and lower paths of the i-th multiplexer, respectively. Simplifying by considering the delay differences $\Delta_i = d_{i,u} - d_{i,l}$, we obtain a linear relationship with respect to the challenge bits.

The CAR-PUF employs two arbiter PUFs (a working PUF and a reference PUF) and a secret threshold τ. The key metric for the CAR-PUF's response is the absolute difference in delays between the two PUFs, $|\Delta_w - \Delta_r|$, compared against τ. The response r is given by:

$$r = \begin{cases} 0 & \text{if } |\Delta_w - \Delta_r| \leq \tau \\ 1 & \text{if } |\Delta_w - \Delta_r| > \tau \end{cases} \tag{2}$$

For a 32-bit challenge c(where $c_i \epsilon \{0, 1\}$) :

$$d_i = 1 - 2c_i \tag{3}$$

$$x_i = d_i d_{i+1} ..... d_{32} \tag{4}$$

where $x_i \epsilon \{-1, 1\}$ Now, let Δ_w and Δ_r denote the delay differences in the working and reference PUFs, respectively:

$$\Delta_w = \sum_{i=1}^{32} w_{wi} x_i + b_w, \tag{5}$$

$$\Delta_r = \sum_{i=1}^{32} w_{ri} x_i + b_r. \tag{6}$$

Squaring both sides of the inequality related to r eliminates the modulus:

$$(\Delta_w - \Delta_r)^2 < \tau^2. \tag{7}$$

Expanding yields:

$$\Delta_w^2 - 2\Delta_w \Delta_r + \Delta_r^2 - \tau^2 < 0. \tag{8}$$

Substituting Δ_w and Δ_r from equations (3) and (4) leads to a quadratic form. According to the question, The response is 0 if $|\Delta_1 - \Delta_2| \leq \tau$ and 1 otherwise. Let us consider the case where the response should be 0 . Squaring both sides to handle mod:

$$\Rightarrow (\Delta_1 - \Delta_r)^2 \leq \tau^2$$
$$\Rightarrow \left[(\mathbf{w}1 - \mathbf{w}2)^T \mathbf{x} + (b_1 - b_2)\right]^2 < \tau^2$$
$$\Rightarrow \left(\mathbf{w}^T\mathbf{x} + b\right)^2 \leq \tau^2$$

where $\mathbf{w} = \mathbf{w}1 - \mathbf{w}2$ and $b = b_1 - b_2$. Thus,

$$\Rightarrow \left(\mathbf{w}^T\mathbf{x}\right)^2 + b^2 + 2\left(\mathbf{w}^T\mathbf{x}\right)b \leq \tau^2$$
$$\Rightarrow \left(\sum w_i x_i\right)^2 + 2\left(\mathbf{w}^T\mathbf{x}\right)b + \left(b^2 - \tau^2\right) \leq 0$$
$$\Rightarrow \sum (w_i x_i)^2 + 2\sum w_i w_j x_i x_j + 2\left(\mathbf{w}^T\mathbf{x}\right)b + \left(b^2 - \tau^2\right) \leq 0$$

$$\Rightarrow \sum (w_i x_i)^2 + 2\sum w_i w_j x_i x_j + 2\left(\sum w_i x_i\right)b + \left(b^2 - \tau^2\right) \leq 0$$

Since $x_i = \pm 1, x_i^2 = 1$. Hence, $\sum (w_i x_i)^2 = \sum w_i^2$, which is a constant for 2 fixed PUFs

$$\Rightarrow 2\sum w_i w_j x_i x_j + 2\left(\sum w_i x_i\right)b + \left(b^2 + \sum w_i^2 - \tau^2\right) \leq 0 \qquad \text{(A)}$$

We define a feature mapping $\phi : \{0,1\}^{32} \to \mathbb{R}^D$ that transforms c into a higher-dimensional feature space ,allowing for linearization: where ϕ is a function of c

This enables a linear model characterized by a weight vector $W \in \mathbb{R}^D$ and a bias term $b \in \mathbb{R}$ to approximate the CAR-PUF response:

$$r = \frac{1 + \text{sign}(W^\top \phi(c) + b - \tau^2)}{2}. \qquad (9)$$

$$r = \frac{1 + \text{sign}(W^\top \phi(c) + b\prime)}{2}. \qquad (10)$$

To model the CAR-PUF response accurately, we first change the challenge vector c to new challenge vector x (using equation 3 and 4) and then map the 32-bit new challenge vector x into a 528-dimensional feature space. This mapping, denoted as $\phi(x)$, includes the new challenge bits x and their pairwise interactions:

$$\phi(x) = (x_1, x_2, \ldots, x_{32}, x_1x_2, x_1x_3, \ldots, x_{31}x_{32})$$

As x is function of c, Hence ϕ is also a function of c

This transformation results in a feature vector with $\binom{32}{2} = 496$ pairwise interaction terms, in addition to the 32 new challenge bits x, totaling 528 features i.e D=528.

PART 3

The following data shows how various hyperparameters affected training time and test accuracy.

3.1 (a) Changing the loss hyperparameter in Linear SVC (Hinge vs Squared Hinge)

	Hinge	Squared Hinge
Training Time(s)	11.75	13.71
Test Accuracy(%)	98.85	99.14

3.2 (b) Setting C to high/low/medium value

For Linear SVC:

	High(C=100)	Medium(C=1)	Low(C=0.01)
Training Time(s)	12.76	11.42	5.48
Test Accuracy(%)	99.02	99.13	98.65

For Logistic Regression:

	High(C=100)	Medium(C=1)	Low(C=0.01)
Training Time(s)	1.67	1.82	1.29
Test Accuracy(%)	99.31	99.07	96.35

3.3 (c) Changing tol to high/low/medium value

For Linear SVC:

	High(1e-2)	Medium(1e-4)	Low(1e-6)
Training Time(s)	13.3	14.7	14.41
Test Accuracy(%)	99.13	99.11	99.09

For Logistic Regression:

	High(1e-2)	Medium(1e-4)	Low(1e-6)
Training Time(s)	1.28	1.73	1.48
Test Accuracy(%)	99.07	99.07	99.07

3.4 Changing the penalty (regularization) hyper-parameter (12 vs 11)

For Linear SVC:

	11	**12**
Training Time(s)	161.9	11.42
Test Accuracy(%)	99.12	99.10

For Logistic Regression:

Here we used solver='saga', as the default solver i.e 'lbfgs' does not support l1 penalty

	11	**12**
Training Time(s)	46.4	35.32
Test Accuracy(%)	99.06	99.03