CS771 Assignment-1

Banoth Anand
220281
banotha22@iitk.ac.in

Bali Yaswanth Naidu
220279
baliyn22@iitk.ac.in

Nidhi Bajpai
230694
nidhib23@iitk.ac.in

Pankhuri Sachan
230734
pankhurisa23@iitk.ac.in

1 Solution 1:

We need to prove that a linear model could predict the time upper signal takes to reach the finish line. From the lecture slides we know that

$$t_i^u = (1 - c_i)(t_{i-1}^u + p_i) + c_i(t_{i-1}^l + s_i)$$
$$t_i^l = (1 - c_i)(t_{i-1}^l + q_i) + c_i(t_{i-1}^u + r_i)$$

So using the first equation we get

$$t_2^u - t_1^u = c_2(s_2 - p_2 - \Delta_1) + p_2$$
$$t_3^u - t_3^u = c_3(s_3 - p_3 - \Delta_2) + p_3$$

and so on we get

$$t_{31}^u - t_{30}^u = c_{31}(s_{31} - p_{31} - \Delta_{31}) + p_{31}$$

upon adding all the equations we get

$$t_{31}^u = \sum_{i=0}^{31} c_i(s_i - p_i) + \sum_{i=0}^{31} p_i - c_{31}\Delta_{30} - c_{30}\Delta_{29} - \ldots - c_1\Delta_0$$
$$t_{31}^u = \sum_{i=0}^{31}(c_i(s_i - p_i) + p_i - c_i\Delta_{i-1}), \text{ where } \Delta_{-1} = 0$$
$$2t_{31}^u = \sum_{i=0}^{31}(2c_i(s_i - p_i) + 2p_i - 2c_i\Delta_{i-1})$$
$$2t_{31}^u = \sum_{i=0}^{31}(d_i(p_i - s_i) + d_i\Delta_{i-1} - \Delta_{i-1} + p_i + s_i), \text{ where } d_i = 1 - 2c_i$$

From the lecture slides we know,

$$\Delta_i - d_i\Delta_{i-1} = \alpha_i d_i + \beta_i$$
$$d_i\Delta_i - \Delta_{i-1} = -(\alpha_i d_i + \beta_i)$$

Preprint. Under review.

By expansion the expression becomes

$$2t_{31}^u = d_1\Delta_0 + d_2\Delta_1 + \cdots + d_{31}\Delta_{30} - \Delta_0 - \Delta_1 - \cdots - \Delta_{30} + \sum_{i=0}^{31}\left(d_i(p_i - s_i) + p_i + s_i\right)$$

$$2t_{31}^u = (d_1\Delta_0 - \Delta_1) + (d_2\Delta_1 - \Delta_2) + \cdots + (d_{30}\Delta_{29} - \Delta_{30}) + (d_{31}\Delta_{30} - \Delta_{31}) + \Delta_{31} - \Delta_0 + \sum_{i=0}^{31}\left(d_i(p_i - s_i) + p_i + s_i\right)$$

$$2t_{31}^u = \sum_{i=1}^{31}(d_i\Delta_{i-1} - \Delta_i) + \Delta_{31} - \Delta_0 + \sum_{i=0}^{31}(d_i(p_i - s_i) + p_i + s_i)$$

$$2t_{31}^u = \sum_{i=1}^{31}(-\alpha_i d_i - \beta_i) + \Delta_{31} - \Delta_0 + \sum_{i=0}^{31}(d_i(p_i - s_i) + p_i + s_i)$$

$$2t_{31}^u = \sum_{i=1}^{31}(d_i(p_i - s_i) + p_i + s_i - \alpha_i d_i - \beta_i) + \Delta_{31} - \Delta_0 + (d_0(p_0 - s_0) + p_0 + s_0)$$

$$2t_{31}^u = \sum_{i=0}^{31}(d_i(m_i) + n_i) + \Delta_{31}, \text{ where } m_i = p_i - s_i - \alpha_i \text{ and } n_i = p_i + s_i - \beta_i$$

We know from the lecture slides that

$$\Delta_{31} = w_0x_0 + w_1x_1 + \ldots + w_{31}x_{31} + (\text{Bias term})_1$$

where,

$$w_0 = \alpha_0$$
$$w_i = \alpha_i + \beta_{i-1}$$
$$x_i = d_i d_{i+1} \ldots d_{31}$$

So,

$$t_{31}^u = \mathbf{W}^T\phi(c) + b$$

$$x \equiv \phi(c)$$

where,

$$\mathbf{W}_0 = \frac{\alpha_0}{2}$$
$$\mathbf{W}_i = \frac{\alpha_i + \beta_{i-1}}{2} \quad \text{if} \quad i \in [1, 30]$$
$$\mathbf{W}_{31} = \frac{\beta_{30} + p_{31} - s_{30}}{2}$$
$$\mathbf{W}_i = \frac{m_{(i-32)}}{2} \quad \text{if} \quad i \in [32, 62]$$
$$x_i = d_i d_{i+1} \ldots d_{31} \quad \text{if} \quad i \in [0, 31]$$
$$x_i = d_{(i-32)} \quad \text{if} \quad i \in [32, 62]$$

So, it can be proved that a linear model can predict the time taken by upper signal to reach the end.

2 Solution 2:

The dimensionality of the linear model should be 63 to predict the arrival time of the upper signal for an arbiter PUF.

3 Solution 3:

We know that a linear model could predict the time taken by upper signal to reach the end. Similarly, we can prove that for the lower signal too. Using the derivation from solution 1, we get

$$t_{31}^{l} = \mathbf{W}^T \phi(c) + b$$

$$x \equiv \phi(c)$$

where,

$$\begin{aligned}
\mathbf{W}_0 &= \frac{-\alpha_0}{2} \\
\mathbf{W}_i &= \frac{-(\alpha_i + \beta_{i-1})}{2} \quad \text{if} \quad i \in [1, 30] \\
\mathbf{W}_{31} &= \frac{q_{31} - r_{31} - \beta_{30}}{2} \\
\mathbf{W}_i &= \frac{n_{(i-32)}}{2} \quad \text{if} \quad i \in [32, 62] \text{where, } n_i = \alpha_i + q_i - r_i \\
x_i &= d_i d_{i+1} \ldots d_{31} \quad \text{if} \quad i \in [0, 31] \\
x_i &= d_{(i-32)} \quad \text{if} \quad i \in [32, 63]
\end{aligned}$$

Response 0 is nothing but,

$$\frac{1 + \text{sign}\left((t_{31}^{l})_{PUF1} - (t_{31}^{l})_{PUF0}\right)}{2}$$

$$\begin{aligned}
(t_{31}^{l})_{PUF1} &= \mathbf{W}_1^T \phi(c) + b_1 \\
(t_{31}^{l})_{PUF0} &= \mathbf{W}_0^T \phi(c) + b_2
\end{aligned}$$

$$(t_{31}^{l})_{PUF1} - (t_{31}^{l})_{PUF0} = (\mathbf{W}_1^T - \mathbf{W}_0^T)\phi(c) + b_3$$

$\mathbf{W}_1, \mathbf{W}_0$ depend on PUF specific constants

So,

$$r^0(c) = \frac{1 + \text{sign}(\tilde{\mathbf{W}}^T \tilde{\phi}(c) + \tilde{b})}{2}$$

where

$$\begin{aligned}
\tilde{\mathbf{W}}^T &= (\mathbf{W}_1 - \mathbf{W}_0)^T \\
\tilde{\phi}(c) &= \phi(c)
\end{aligned}$$

$(\mathbf{W}_1 - \mathbf{W}_0)^T$ can be easily computed from the t_{31}^{l} derivation.

Similarly Response1 can be done by

$$\frac{1 + \text{sign}\left((t_{31}^{u})_{PUF1} - (t_{31}^{u})_{PUF0}\right)}{2}$$

$$\begin{aligned}
(t_{31}^{u})_{PUF1} &= \mathbf{W}_1^T \phi(c) + b_1 \\
(t_{31}^{u})_{PUF0} &= \mathbf{W}_0^T \phi(c) + b_2
\end{aligned}$$

$$(t_{31}^{u})_{PUF1} - (t_{31}^{u})_{PUF0} = (\mathbf{W}_1^T - \mathbf{W}_0^T)\phi(c) + b_3$$

$\mathbf{W}_1, \mathbf{W}_0$ depend on PUF specific constants

So,

$$r^1(c) = \frac{1 + \text{sign}(\tilde{\mathbf{W}}^T \tilde{\phi}(c) + \tilde{b})}{2}$$

where

$$\tilde{\mathbf{W}}^T = (\mathbf{W}_1 - \mathbf{W}_0)^T$$
$$\tilde{\phi}(c) = \phi(c)$$

$(\mathbf{W}_1 - \mathbf{W}_0)^T$ can be easily computed from the t_{31}^u derivation.

4 Solution 4:

The dimensionality needed by the linear model to have to predict Response0 and Response1 for a COCO-PUF is 63.

5 Solution 6:

Linear SVC			
Loss	Accuracy (response0)	Accuracy (response1)	Time
Hinge	97.6	88.75	19 sec
Squared hinge	97.7	88.69	31 sec

Table 1: Changing the loss hyperparameter in LinearSVC (hinge vs squared hinge)

article array

Linear SVC			
C	Accuracy (response0)	Accuracy (response1)	Time
100	97.73	88.74	22 sec
1	97.69	88.7	33 sec
1	97	88.11	4 sec

Logistic Regression			
C	Accuracy (response0)	Accuracy (response1)	Time
100	97.24	89.31	13 sec
1	97.21	89.46	3 sec
0.01	93.38	89.27	2 sec

Table 2: Changing the C hyperparameter in LinearSVC and Logistic Regression to high/medium/low values

Linear SVC			
Tol	Accuracy (response0)	Accuracy (response1)	Time
Hard	97.69	88.7	21 sec
Medium	97.69	88.7	33 sec
Low	97.69	88.7	37 sec

Logistic Regression			
Tol	Accuracy (response0)	Accuracy (response1)	Time
Hard	97.21	89.46	3 sec
Medium	97.21	89.46	4 sec
Low	97.21	89.46	4 sec

Table 3: Changing the tol hyperparameter in LinearSVC and Logistic Regression to high/medium/low values

Linear SVC			
Penalty	Accuracy (response0)	Accuracy (response1)	Time
l1	97.91	88.6	6m 32sec
l2	97.7	88.69	4 sec

Logistic Regression			
Penalty	Accuracy (response0)	Accuracy (response1)	Time
l1	N/A	N/A	N/A
l2	97.21	89.46	4 sec

Table 4: Changing the penalty (regularization) hyperparameter in LinearSVC and Logistic Regression (l2 vs l1)

Note: All the hyperparameters tuning was done manually(i.e, manually selected and adjusted the hyperparameters of the model).