CS771 - Introduction to Machine Learning - Assignment 1

Jaya Gupta (200471)
Jahnvi Rochwani (200467)
Harshit Bansal (200428)
Romit Mohanty (190720)
Somya Lohani (190848)
Himanshu Sood (190381)

Problem 1.1

By giving a mathematical derivation, show the exists a way to map the binary digits 0, 1 to signs -1, +1 as say, $m : \{0, 1\} \rightarrow \{\text{-}1, +1\}$ and another way $f : \{\text{-}1, +1\} \rightarrow \{0, 1\}$ to map signs to bits (note that m and f need not be inverses of each other) so that for any set of binary digits $b_1, b_2, ..., b_n$ for any $n \in N$, we have

$$\text{XOR}(b_1, ..., b_n) = f\left(\prod_{i=1}^{n} m(b_i)\right)$$

Thus, the XOR function is not that scary – it is essentially a product.

Proof: Say k digits from $b_1, ..., b_n$ are 1.

We can define m as a mapping where $m(0) = 1$ and $m(1) = -1$, and f as $f(x) = \dfrac{1-x}{2}$

$$\begin{aligned} f\left(\prod_{i=1}^{n} m(b_i)\right) &= f\left((-1)1^k \cdot 1^{n-k}\right) \\ &= f\left((-1)^k\right) \\ &= \frac{1-(-1)^k}{2} \\ \text{XOR}(b_1, ..., b_n) &= \frac{1-(-1)^k}{2} \end{aligned} \tag{1}$$

If k is odd, XOR$(b_1, ..., b_n)$ is 1. Else, it is 0.

Problem 1.2

In this part, our aim is to find

$$sign(\tilde{\mathbf{u}}^T \tilde{x}) \cdot sign(\tilde{\mathbf{v}}^T \tilde{x}) \cdot sign(\tilde{\mathbf{w}}^T \tilde{x})$$

Proof: For any real numbers $(r_1, r_2, ..., r_n)$ for any $n \in \mathbb{N}$, we have

$$\prod_{i=1}^{n} sign(r_i) = sign(\prod_{i=1}^{n} r_i)$$

Case1: When one or more $r_i^{'s}$ are 0

Let r_{i^1}, r_{i_2} ,r_{i_3} .....,r_{i_k} be 0. It is given $sign(0) = 0$, so $sign(r_{i^1})$, $sign(r_{i_2})$,$sign(r_{i_3})$ .....,$sign(r_{i_k})$ is also equal to 0. Hence term on LHS side is 0.

Similarly $\prod_{i=1}^{n} r_i = 0$. Hence, LHS=RHS.

Case2: When all $r_{i's} \neq 0$

$$sign(x) = \begin{cases} -1 & x < 0 \\ 1 & x > 0 \end{cases}$$

Let number of positive $r_i^{'s}$ be k, and the number of negative ones be $(n-k)$.

$$\prod_{i=1}^{n} sign(r_i) = \begin{cases} -1 & k\%2 \neq 0 \\ 1 & k\%2 = 0 \end{cases}$$

$$\prod_{i=1}^{n} r_i = \begin{cases} negative & k\%2 \neq 0 \\ positive & k\%2 = 0 \end{cases}$$

$$sign(\prod_{i=1}^{n} r_i) = \begin{cases} -1 & k\%2 \neq 0 \\ 1 & k\%2 = 0 \end{cases}$$

Hence, LHS=RHS.

Problem 1.3

$$(\tilde{u}^T \tilde{x}) \cdot (\tilde{v}^T \tilde{x}) \cdot (\tilde{w}^T \tilde{x})$$

For the sake of generality let us assume that the number of PUF's are n (in this case 3) and the dimensionality of $\tilde{x}$ is d (in this case 9).

We need to show that, there exists a dimensionality D which depends on n and d, and there exists a way to map d (here 9) dimensional vectors to D dimensions

$$\phi : \mathbb{R}^9 \to \mathbb{R}^D$$

and there exists a vector $W \in \mathbb{R}^D$ such that

$$(\tilde{u}^T \tilde{x}) \cdot (\tilde{v}^T \tilde{x}) \cdot (\tilde{w}^T \tilde{x}) = \mathbf{W}^T \phi(\tilde{x})$$

$$\tilde{u}^T \tilde{x} = \sum_{j=1}^{j=9} \tilde{u}_j \tilde{x}_j$$

$$\tilde{v}^T \tilde{x} = \sum_{j=1}^{j=9} \tilde{v}_j \tilde{x}_j$$

$$\tilde{w}^T \tilde{x} = \sum_{j=1}^{j=9} \tilde{w}_j \tilde{x}_j$$

Multiplying all the three terms, gives us

$$(\tilde{u}^T \tilde{x}) \cdot (\tilde{v}^T \tilde{x}) \cdot (\tilde{w}^T \tilde{x}) = (\sum_{i=1}^{i=9} \tilde{u}_i \tilde{x}_i)(\sum_{j=1}^{j=9} \tilde{v}_j \tilde{x}_j)(\sum_{k=1}^{k=9} \tilde{w}_k \tilde{x}_k)$$

$$(\tilde{u}^T \tilde{x}) \cdot (\tilde{v}^T \tilde{x}) \cdot (\tilde{w}^T \tilde{x}) = \sum_{i=1}^{i=9} \sum_{j=1}^{j=9} \sum_{k=1}^{k=9} \tilde{u}_i \tilde{v}_j \tilde{w}_k \tilde{x}_i \tilde{x}_j \tilde{x}_k$$

It creates $9 * 9 * 9 = 729$ terms. So for this case our D will be 729. In general terms D can be expressed as $D = d^n$. In this case, we have $d = 9$ and $n = 3$, hence value of $D = 9^3$.

Lets create a mapping ϕ to that maps $\tilde{\mathbf{x}} = (x_1, x_2, ...x_9)$ to the new input values.

$$\phi(\tilde{\mathbf{x}}) = (x_1x_1x_1, x_1x_1x_2, ..., x_1x_1x_9, x_1x_2x_1, x_1x_2x_2, ..., x_9x_9x_9)$$

and similarly let us create the vector $\mathbf{W} \in \mathbb{R}^D$

$$\mathbf{W} = (\tilde{u}_1\tilde{v}_1\tilde{w}_1, \tilde{u}_1\tilde{v}_1\tilde{w}_2, ..., \tilde{u}_1\tilde{v}_1\tilde{w}_9, \tilde{u}_1\tilde{v}_2\tilde{w}_1, \tilde{u}_1\tilde{v}_2\tilde{w}_2, ..., \tilde{u}_9\tilde{v}_9\tilde{w}_9)$$

This gives

$$(\tilde{u}^T\tilde{x}) \cdot (\tilde{v}^T\tilde{x}) \cdot (\tilde{w}^T\tilde{x}) = \mathbf{W}^T\phi(\tilde{x})$$

where $\mathbf{W} \in R^D$ and our new input $\phi(\tilde{x}) \in R^D$.

Problem 1.5

For the method you implemented, describe in your PDF report what were the hyperparameters e.g. step length, policy on choosing the next coordinate if doing SDCA, mini-batch size if doing MBSGD etc and how did you arrive at the best values for the hyperparameters, e.g. you might say "We used step length at time t to be $\frac{\eta}{t}$ where we checked for $\eta = 0.1, 0.2, 0.5, 1, 2, 5$ using held out validation and found $\eta = 2$ to work the best". For another example, you might say, "We tried random and cyclic coordinate selection choices and found cyclic to work best using 5-fold cross validation". Thus, you must tell us among which hyperparameter choices did you search for the best and how. (5 marks)

Solution: We used step length at time t to be $\frac{\eta}{t}$ where we checked for $\eta = 0.1, 0.2, 0.5, 1, 2, 5$ using held out validation and found $\eta = 2$ to work the best". For another example, you might say, "We tried random and cyclic coordinate selection choices and found cyclic to work best using 5-fold cross validation".

Problem 1.6

Plot the convergence curves in your PDF report offered by your chosen method as we do in lecture notebooks. The x axis in the graph should be time taken and the y axis should be the test classification accuracy (i.e. higher is better). Include this graph in your PDF file submission as an image.