CS 419M Introduction to Machine Learning **Spring 2021-22**

Lecture 2: Loss Functions in Machine Learning

Lecturer: Abir De *Scribe: 180070020, 200100084, 20D070087, 180110088, 19D180002*

2.1 Loss function for Image Classification

2.1.1 Problem Setup

- An Image I is represented by a vector x, x ϵ R^d. 2d array is collapsed into a 1d vector by well developed methods. For now we are considering only black & white images.
- Label of an Image y represents the object present in the image. eg: a car, a tree or a river. If there are two classes, let's say dog and car, we can have y = 0 for dog and y = 1 for car.
- We consider a Dataset D to be set of images that belong to two classes either C1 or C2.

$$\mathbb{D} = \{(x_i, y_i) | i \in \mathbb{I}\}$$

where x_i is the 1d vector representation of the image and $y_i \ \epsilon \ \{0, 1\}$.

2.1.2 What is Classification

- Goal of classification is to find out the labels of test/unseen images.
- **Unseen Images:** The instance/image that was not revealed during the development of the ML model/algorithm.
- We need to design a function h($\cdot$) that will be able to give accurate class label i.e.,

$$y = h(x) \qquad \forall\ x \in \text{Test set}$$

using the information from Training set.

- **Training set :** The set of examples (tuples (x_i,y_i) image representations along with labels) provided to the machine learning model/ algorithm at the development stage.

2.1.3 How to find function h(x)

- The idea is find best $h(x) \in \mathbb{H}$, where $\mathbb{H}$ is infinite/huge set of functions such that it minimises the error.

- From the first principles, initial idea would be

$$\min_{h(x)\in\mathbb{H}} \sum_{(x_i,y_i)\in\mathbb{D}} |h(x_i) - y_i| \qquad \text{where } y_i \in \{0,1\}\text{and } h(x_i) \in \mathbb{R}$$

  but since $y_i \in \{0,1\}$ only, we should look for some better answer.

- The new idea will be of **Penalty System** which basically means that whenever $h(x_i)$ differs from y_i, we will add some penalty. The naive idea is as follows:
  - if $y_i = 0$ and $h(x_i) = 1$, penalty = 1
  - if $y_i = 1$ and $h(x_i) = 0$, penalty = 1
  - if $y_i = 0$ and $h(x_i) = 0$, penalty = 0
  - if $y_i = 1$ and $h(x_i) = 1$, penalty = 0

  Keeping in mind that $h(x_i) \in \mathbb{R}$, we will be doing it as follows

$$\min_{h(x)\in\mathbb{H}} \sum_{(x,y)\in\mathbb{D}} \mathbb{I}(h(x) \neq y)$$

  - where I is **Indicator function** with values

$$\mathbb{I}(X) = \begin{cases} 0 & X = false \\ 1 & X = true \end{cases}$$

  - In some sense, the above implementation is a **Hard Penalty** since we are penalising whenever $h(x) \neq y$.
  - Moreover we have dropped "i" in expression for convenience and will continue to do so.

- This type of function is hard to find from an infinite set of functions and work upon. So, we need to restrict the function between 0 and 1, for eg. if there is a continuous function h(x) which can be transformed using a known function f(·) such that f(h(x)) gives output only as 0,1. then,

$$\min_{h(x)\in\mathbb{H}} \sum_{(x,y)\in\mathbb{D}} \mathbb{I}(f(h(x)) \neq y)$$

  in both cases we are searching over the entire space but here our work is reduced as function f(.) is ensuring that the indicator function has to only deal with a value 0 or 1 inside it.

- Following the previous point, we will find a function f which can squeeze $h(x)$ to {0,1}. One such funtion f is $Sign(x)$ i.e. **Signum Funtion**.

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

$$\frac{1 + Sign(h(x))}{2} = \begin{cases} 1 & h(x) > 0 \\ 0.5 & h(x) = 0 \\ 0 & h(x) < 0 \end{cases}$$

One very important point to note here is that our output function which will be finally used to give labels to Unseen Instances is not $h(x)$ anymore. After this transformation, our Output function will be

$$\frac{1 + Sign(h(x))}{2}$$

- Using Sign funtion, our new optimization problem will be

$$\min_{h(x) \in \mathbb{H}} \sum_{(x,y) \in \mathbb{D}} \mathbb{I}\left(\frac{1 + Sign(h(x))}{2} \neq y\right)$$

But its a very hard Optimization problem because $\mathbb{H}$ is huge and working with $\mathbb{I}$ is tedious. So, we have to relax the conditions.

2.1.4 Linear Model for h(x)

- To proceed further, we will assume

$$h(x) = w^T x \qquad \text{for some column vector w}$$

- So, new optimization problem is

$$\min_{w} \sum_{(x,y) \in \mathbb{D}} \mathbb{I}\left(\frac{1 + Sign(w^T x)}{2} \neq y\right)$$

But optimization with Indicator function is hard. We will modify the problem as follows.

$$\min_{w} \sum_{(x,y) \in \mathbb{D}} \left|\frac{1 + Sign(w^T x)}{2} - y\right|^2$$

But it is non differentiable due to $sign(x)$ and we will not be able to apply Calculus techniques to optimize it.
Is the below modification a good idea?

$$\min_{w} \sum_{(x,y) \in \mathbb{D}} \left|\frac{1 + w^T x}{2} - y\right|^2$$

No, because $w^T x$ can be large which is OK but then loss will become large which is not good

- Its time to think of some differentiable analog to the above problem. One funtion which can satisfy our requirements is **Sigmoid** Function.

$$Sigmoid(x) = S(x) = \frac{1}{1 + e^{-x}} \qquad where\, x \in \mathbb{R}$$

Sigmoid function satisfies our requirement because it is differentiable and $S(x) \in (0,1)$ i.e. it can squeeze $h(x)$ to (0,1).
It has a little issue that it didn't squeeze h(x) to {0,1} but good point is that we can apply calculus techinques to it now.

Now, our problem modifies to

$$\min_{w} \sum_{(x,y)\in\mathbb{D}} \left(Sigmoid(w^T x) - y\right)^2$$

Note that the above optimization problem is not convex so we can do better if we can find surrogate.
Reminding again that our Output function now is $Sigmoid(w^T x)$ and not $h(x)$ anymore.

2.1.5 Convex Loss function

- Convex means that if you run gradient descent then it is guaranteed to converge in the global minima.
- We have to find a surrogate function for $w^T x$ with the following properties:
  1. If $w^T x$ is high, y is 1, then loss is 0.
  2. If $w^T x$ is low, y is 0 or -1, then loss is 0.
  3. The loss has to be convex with respect to w.
- The way to check whether a function is convex or not is to find the double derivative and check if it is always positive with respect to w.
- The eigen values of Hessian has to be greater than or equal to 0 for convexity with respect to **w**.

$$H = \left[\frac{\partial^2 a}{\partial w^2}\right] \qquad \lambda(H) \geq 0$$

- The value of **w** at which function gives minima need not be unique, but value of function for w should be unique.

2.2 Group Details and Individual Contribution

Name	Roll Number	Sections
Garaga V V S Krishna Vamsi	180070020	2.1.1, 2.1.2
Vaibhav Kumar	20d070087	2.1.3, 2.1.4
Abhinav Singh	19D180002	2.1.5
Sristy Kushwaha	180110088	2.1.3
Kavyan Lavti	200100084	2.1.4