HW 6

Jonathan Zhao

11/17/2024

What is the difference between gradient descent and *stochastic* gradient descent as discussed in class? (*You need not give full details of each algorithm. Instead you can describe what each does and provide the update step for each. Make sure that in providing the update step for each algorithm you emphasize what is different and why.*)

Student Input

Gradient Descent is a way to minimize the loss function when using federated learning, and calculates the direction of steepest descent. The update step is

$$\theta_{t+1} = \theta_t - \alpha \nabla F(\theta_t)$$

Gradient Descent is likely to get stuck on local extremes instead of global extremes, making it unable to minimize the loss function. Stochastic Gradient Descent is similar to Gradient Descent, but it introduces variability in calculating gradients so it does not get stuck on local extremes. This is done in the update step:

$$\theta_{t+1} = \theta_t - \alpha \nabla F_i(\theta_t)$$

Where F_i is a random subset of the data.

Consider the `FedAve` algorithm. In its most compact form we said the update step is $\omega_{t+1} = \omega_t - \eta \sum_{k=1}^{K} \frac{n_k}{n} \nabla F_k(\omega_t)$. However, we also emphasized a more intuitive, yet equivalent, formulation given by $\omega_{t+1}^k = \omega_t - \eta \nabla F_k(\omega_t); w_{t+1} = \sum_{k=1}^{K} \frac{n_k}{n} w_{t+1}^k$.

Prove that these two formulations are equivalent.
(*Hint: show that if you place ω_{t+1}^k from the first equation (of the second formulation) into the second equation (of the second formulation), this second formulation will reduce to exactly the first formulation.*)

Student Input By placing ω_{t+1}^k into the second expression part of the second formulation, we get:

$$\sum_{k=1}^{K} \frac{n_k}{n} (\omega_t - \eta \nabla F_k(\omega_t))$$

This can then be simplified, by pulling out the constants such as ω_t and η, to get $\omega_{t+1} = \omega_t - \eta \sum_{k=1}^{K} \frac{n_k}{n} \nabla F_k(\omega_t)$

Now give a brief explanation as to why the second formulation is more intuitive. That is, you should be able to explain broadly what this update is doing.

Student Input

This update takes one step of Stochastic Gradient Descent for each of the K clients, and then averages them to give a global update. its more intuitive because it separates the steps for the local clients from the global aggregation, instead of compacting it like the first formula, so that it is easier to understand.

Prove that randomized-response differential privacy is ϵ-differentially private.

Student Input

The definition of epsilon differentially privat is that for an epsilon e>0, and datasets D1,D2 that differ in exactly one element, an algorithm A is epsilon differentially private on a subset S if:

$$Probability[A(D1) is in subset S]/[Probability[A(D2) is in subset S]$$

Now for randomized response differential privacy, lets assume the subset S is the probability of the response being ‘yes’, and D1,D2 differ on whether the real answer is ‘yes’ or ‘no’. There are 3 possible combinations for saying yes given input is yes, out of 4 possible answers, so the top probability is 3/4. There are 1/4 possible combinations for saying yes given input is no, so the bottom probability is 1/4. Thus,

$$(3/4)/(1/4) = 3$$

Which means the biggest probability is 3, and expressed as an exponential it becomes

$$e^{ln(3)}$$

, so randomized response differential privacy is Epsilon differentially private where Epsilon = ln(3).

Define the harm principle. Then, discuss whether the harm principle is *currently* applicable to machine learning models. (*Hint: recall our discussions in the moral philosophy primer as to what grounds agency. You should in effect be arguing whether ML models have achieved agency enough to limit the autonomy of the users of said algorithms.*)

Student Input The Harm Principle states that personal autonomy should be restricted when using autonomy would result in objective moral harm. This principle is currently applicable to machine learning models because of the impact such models can have on people. ML models have achieved enough agency because they are effectively black boxes and can exhibit agency that the programmers may have unintentionally imbued in the models. Because ML models can significantly shape decisions and behaviors of their users, it can cause harm through biases such as discrimination against certain races as seen with the COMPAS algorithm. Therefore, the autonomy of users can be limited because of their trust in such models, so designing ML models with transparency in mind and limiting the use of biased models is necessary.