HW 6

Razmin Bari

11/19/2024

1

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.*)

The update step for gradient descent is $\theta_{i+1} = \theta_i - \alpha\nabla f(\theta_i, x, y)$. *The update step for stochastic descent is* $\theta_{i+1} = \theta_i - \alpha\nabla f(\theta_i, x_I, y_I)$. *The only difference is that a random subset set of the feature(s)* x *and target variable* y *is being used in the stochastic version. This increases the variability in calculating gradients, preventing the possibility of getting stuck within local averages.*

2

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.*)

Proof: $w_{t+1} = \sum_{k=1}^{K}\frac{n_k}{n}w_{t+1}^k$ *–> second equation of the second formulation*

$w_{t+1} = \sum_{k=1}^{K}\frac{n_k}{n}(\omega_t - \eta\nabla F_k(\omega_t))$ *–> substituting with first equation of the second formulation*

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

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

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

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

$w_{t+1} = \omega_t - \eta\sum_{k=1}^{K}\frac{n_k}{n}\nabla F_k(\omega_t)$ *–> derived first formulation*

3

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.

The first equation of the second formulation is an update step for each of the K *local clients. The second equation is where the local updates are averaged over, and used to update globally.*

4

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

Let outputs from randomized responses $\in$ {0,1} given inputs $\in$ {0,1}.

For output = 0, the ratio of probabilities of getting ground truth versus a fake answer:

$\frac{\Pr[output=0|input=0]}{\Pr[output=0|input=1]} = \frac{p}{1-p}$ –> equation 1

For output = 1, the ratio of probabilities is:

$\frac{\Pr[output=1|input=0]}{\Pr[output=1|input=1]} = \frac{1-p}{p}$ –> equation 2

To ensure $\frac{p}{1-p} \leq e^{\epsilon}$*:* $\frac{p}{1-p} = e^{\epsilon} \implies p = \frac{e^{\epsilon}}{1+e^{\epsilon}}$

Substituting p in equation 1: $\frac{\Pr[output=0|input=0]}{\Pr[output=0|input=1]} = \frac{\frac{e^{\epsilon}}{1+e^{\epsilon}}}{\frac{1}{1+e^{\epsilon}}} = e^{\epsilon}$

Substituting p in equation 2: $\frac{\Pr[output=1|input=0]}{\Pr[output=1|input=1]} = \frac{\frac{1}{1+e^{\epsilon}}}{\frac{e^{\epsilon}}{1+e^{\epsilon}}} = \frac{1}{e^{\epsilon}} \leq e^{\epsilon}$

For both cases, the ratio of probabilities is equal to or less than e^{ϵ}. Hence, randomized-response differential privacy is ϵ-differentially private as long as $\frac{p}{1-p} \leq e^{\epsilon}$.

5

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.*)

Harm principle is the idea that a person has the right to autonomy but that that right diminishes if exercising autonomy causes another person objective harm.

I am inclined to think that current ML models do not have any such autonomy. If the results of an ML model infringes upon a user's autonomy, it is the developer's fault as they did not sufficiently test to see if any bias arising from the algorithm output is infringing upon a potential user's autonomy. As long as there are ways to see if an ML model is working correctly and as long as there are ongoing efforts to explain *AI, the harm principle holds the developer accountable, not the ML model itself.*