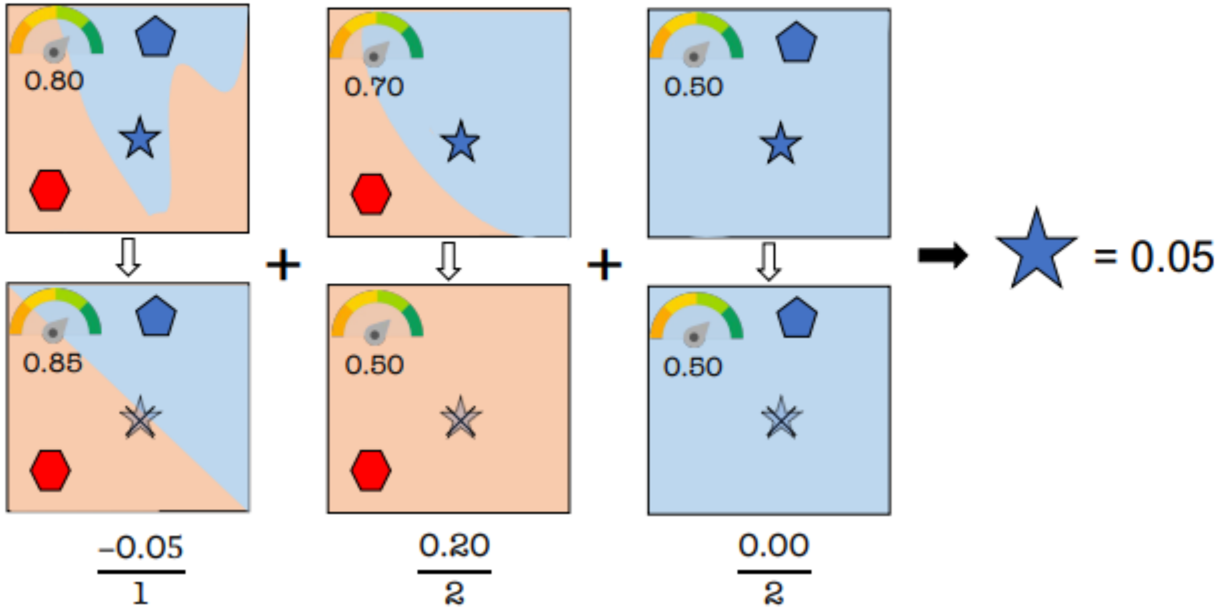
FML论文分享

11.30

Evaluate data by Shapley value 《 What is your data worth? Equitable Valuation of Data 》

(b)



How to evaluate shapley value

$$s_i = \frac{1}{N!} \sum_{\pi \in \Pi(D)} \left[U(P_i^\pi \cup \{i\}) - U(P_i^\pi) \right]$$

$$s_i = \sum_{S \subseteq I \setminus \{i\}} \frac{1}{N \binom{N-1}{|S|}} \left[U(S \cup \{i\}) - U(S) \right]$$

$(A\ B\ C\ D)\ X\ (E\ F\ G\ H\ \ldots)$

$U(\ (A\ B\ C\ D) X\ )$

$-U\ (\ (A\ B\ C\ D)\ )$

Truncated Monte Carlo Shapley

Algorithm 1 Truncated Monte Carlo Shapley

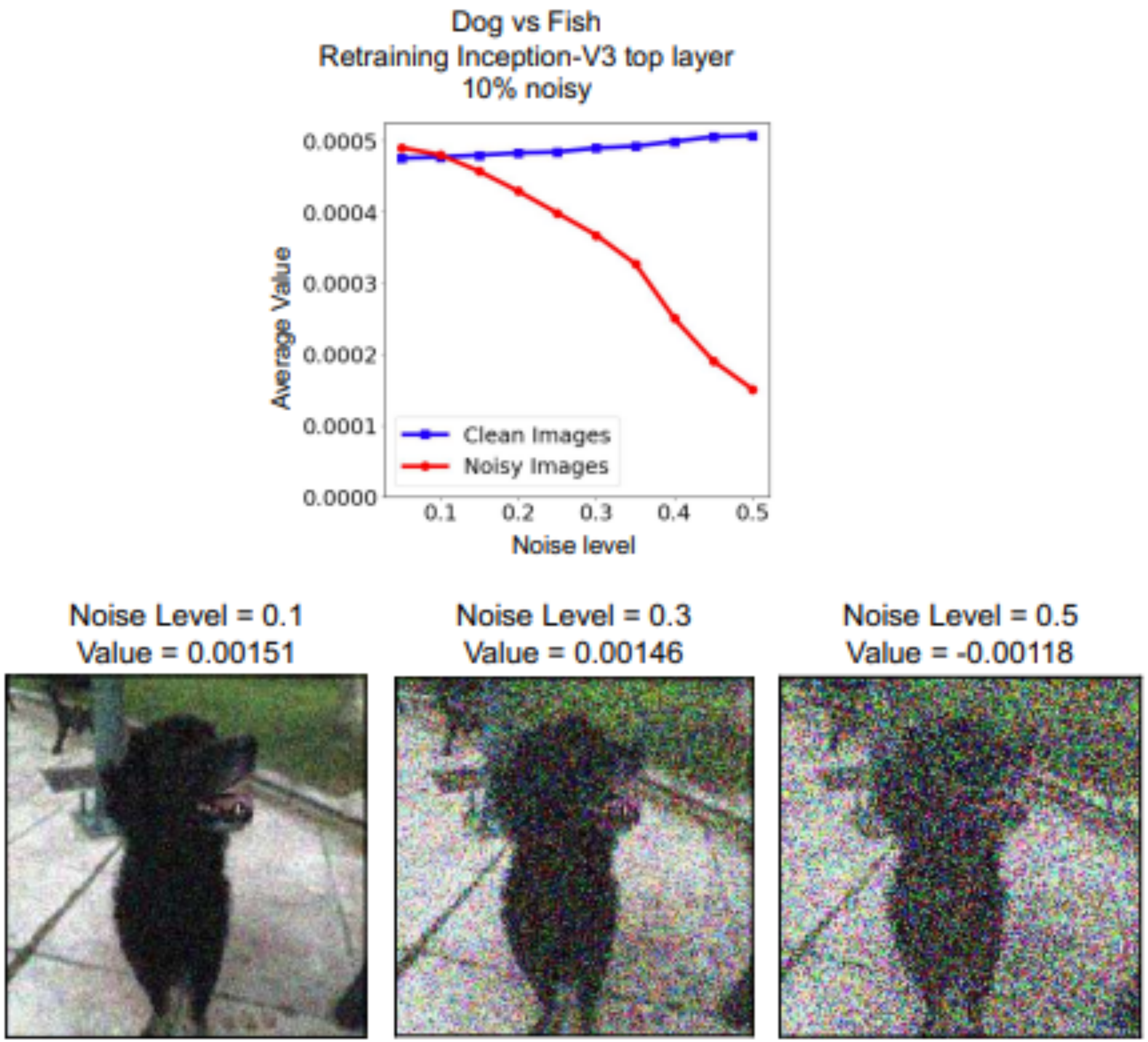
Input: Train data $D = \{1, \ldots, n\}$, learning algorithm $\mathscr{A}$, performance score V
Output: Shapley value of training points: $\phi_1, \ldots, \phi_n$

Initialize $\phi_i = 0$ for $i = 1, \ldots, n$ and $t = 0$
while Convergence criteria not met **do**
  $t \leftarrow t + 1$
  π^t: Random permutation of train data points
  $v_0^t \leftarrow V(\emptyset, \mathscr{A})$
  for $j \in \{1, \ldots, n\}$ **do**
    if $|V(D) - v_{j-1}^t| <$ Performance Tolerance **then**
      $v_j^t = v_{j-1}^t$
    else
      $v_j^t \leftarrow V(\{\pi^t[1], \ldots, \pi^t[j]\}, \mathscr{A})$
    end if
    $\phi_{\pi^t[j]} \leftarrow \frac{t-1}{t} \phi_{\pi^{t-1}[j]} + \frac{1}{t}(v_j^t - v_{j-1}^t)$
  end for
end while

Lemma 4.2. *Given the range of an agent's marginal contributions, r, an error bound, ϵ, and a confidence $1-\delta$, the sample size required such that $\Pr(|\bar{\Phi}_{SRS} - \mathbb{E}[\Phi]| \geq \epsilon) \leq \delta$ is:*

$$m \geq \frac{\ln(2/\delta)\, r^2}{2\,\epsilon^2} \tag{4.3}$$

$$\mathrm{P}[\max_i |\hat{s}_i - s_i| \leqslant \epsilon] \geqslant 1 - \delta. \qquad \frac{r^2}{2\epsilon^2} \log \frac{2N}{\delta}.$$


Dog vs Fish
Retraining Inception-V3 top layer
10% noisy
Average Value
0.0005
0.0004
0.0003
0.0002
0.0001
0.0000
0.1
0.2
0.3
0.4
0.5
Noise level
Clean Images
Noisy Images
Noise Level = 0.1
Value = 0.00151
Noise Level = 0.3
Value = 0.00146
Noise Level = 0.5
Value = -0.00118

《Towards Efficient Data Valuation Based on the Shapley Value》

Lemma 1. *For any $i, j \in I$, the difference in SVs between i and j is*

$$s_i - s_j = \frac{1}{N-1} \sum_{S \subseteq I \setminus \{i,j\}} \frac{U(S \cup \{i\}) - U(S \cup \{j\})}{\binom{N-2}{|S|}} \quad (4)$$

$$\sum_{i=1}^{N} \hat{s}_i = U_{tot} \quad (5)$$

$$|(\hat{s}_i - \hat{s}_j) - C_{i,j}| \leq \epsilon/(2\sqrt{N}) \quad \forall i, j \in \{1, \ldots, N\} \quad (6)$$

Theorem 3. *Algorithm 1 returns an* (ϵ, δ)*-approximation to the SV with respect to* l_2*-norm if the number of tests* T *satisfies* $T \geq 8 \log \frac{N(N-1)}{2\delta} / ((1 - q_{tot}^2) h(\frac{\epsilon}{Zr\sqrt{N}(1-q_{tot}^2)}))$*, where* $q_{tot} = \frac{N-2}{N} q(1) + \sum_{k=2}^{N-1} q(k)[1 + \frac{2k(k-N)}{N(N-1)}]$*,* $h(u) = (1+u)\log(1+u) - u$*,* $Z = 2\sum_{k=1}^{N-1} \frac{1}{k}$*, and* r *is the range of the utility function.*

Algorithm 1: Group Testing Based SV Estimation.

---

input : Training set - $D = \{(x_i, y_i)\}_{i=1}^N$, utility function $U(\cdot)$, the number of tests - T

output : The estimated SV of each training point - $\hat{s} \in \mathbb{R}^N$

$Z \leftarrow 2\sum_{k=1}^{N-1} \frac{1}{k}$;

$q(k) \leftarrow \frac{1}{Z}(\frac{1}{k} + \frac{1}{N-k})$ for $k = 1, \cdots, N-1$;

Initialize $\beta_{ti} \leftarrow 0, t = 1, ..., T, i = 1, ..., N$;

for $t = 1$ *to* T **do**

    Draw $k \sim q(k)$;

    for $j = 1$ *to* k_t **do**

        Uniformly sample a length-k sequence S from $\{1, \cdots, N\}$;

        $\beta_{ti} \leftarrow 1$ for all $i \in S$;

    end

    $u_t \leftarrow U(\{i : \beta_{ti} = 1\})$;

end

$\Delta U_{ij} \leftarrow \frac{Z}{T} \sum_{t=1}^T u_t(\beta_{ti} - \beta_{tj})$ for $i = 1, .., N$, $j = 1, ..., N$ and $j \geq i$;

Find $\hat{s}$ by solving the feasibility problem $\sum_{i=1}^N \hat{s}_i = U(D), |(\hat{s}_i - \hat{s}_j) - \Delta U_{i,j}| \leq \epsilon/(2\sqrt{N}), \forall i, j \in \{1, \cdots, N\}$;

《Rewarding High-Quality Data via Influence for Linear Regression》

$$\text{infl}(z_j, T, \theta) = R(T, \hat{\theta}_{/j}) - R(T, \hat{\theta}).$$

$$\text{infl}(z_{test}, z_j) = \frac{1}{n} \nabla_\theta L(z_{test}, \hat{\theta}) H_\theta^{-1} \nabla_\theta L(z, \hat{\theta})$$

$$\partial\theta_j = \frac{1}{n} H_\theta^{-1} \nabla_\theta L(z_i, \hat{\theta}) + \frac{1}{n^2} H_\theta^{-1} H_i H_\theta^{-1} \nabla_\theta L(z_i, \hat{\theta})$$

$$\text{infl}(z_{\text{test}}, z) = \left(\nabla_\theta L(z_{\text{test}}, \hat{\theta}) + \frac{1}{2} H_{\theta, z_{\text{test}}} \cdot \partial\theta \right) \cdot \partial\theta$$

《Measure Contribution of Participants in Federated Learning》

- Vertical Federated Learning raises new issues for measuring contributions of multiple parties where the feature space is divided into different part

Proposition 4. *If either of Assumption 1 and 2 holds, then the Shapley group value for a party $g \in G$ with feature set X^g is given by*

$$\phi_{X^g} = \sum_{Q \subseteq S \setminus \{j^{fed}\}} \frac{|Q|!(|S|-|Q|-1)!}{|S|!} (\Delta_{Q \cup \{j^{fed}\}}(x) - \Delta_Q(x)), \tag{12}$$

where j^{fed} is the index of the united federated feature x^{fed}.