Data Shapley: Equitable Valuation of Data for Machine Learning

October 31, 2024

Citation

- Ghorbani, A. Zou, J. Data Shapley: Equitable Valuation of Data for Machine Learning. in International Conference on Machine Learning 2242–2251 (2019).

Contents

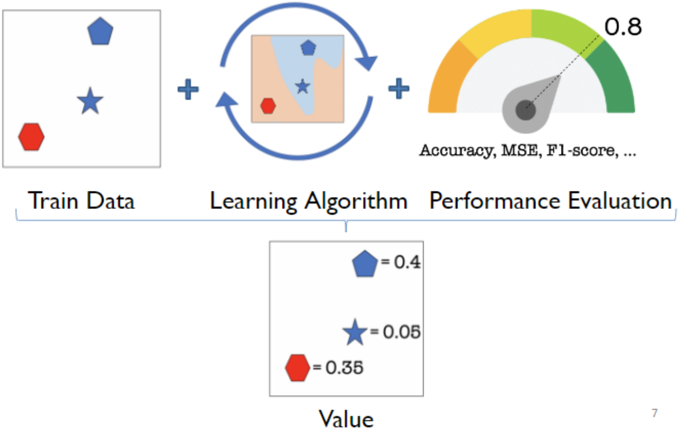
Problem Introduction

Truncated Monte Carlo Algorithm

Gradient Algorithm

Experiment Result

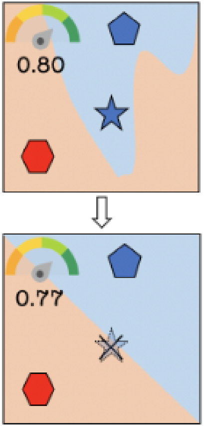
Data Valuation and ML



Leave One Out Method

$$LOO\ for\ point\ i = Performance(D) - Performance(D - \{i\})$$

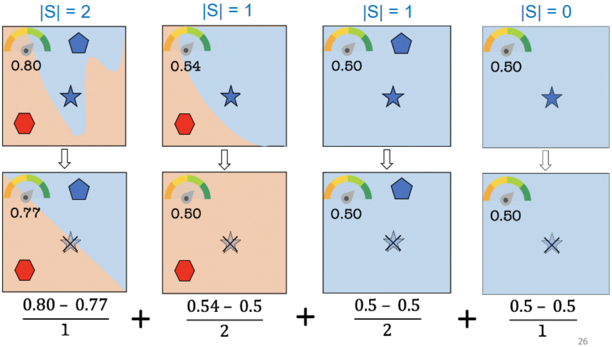
Leave One Out Method



Data Shapley Value

$$SV\ for\ point\ i = C \sum_{S \subseteq D - \{i\}} \frac{Performance(S \bigcup \{i\}) - Performance(S)}{\binom{|D|-1}{|S|}}$$

Data Shapley Value



Truncated Monte Carlo

Algorithm 1 Truncated Monte Carlo Shapley

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

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

Truncation

- $V(S)$ performance on a test set after being trained on S
- as S increases, change in performance by adding only one point decreases
- Truncate based on the marginal contribution within V

Example

Assume 4 data points: A,B,C,D

- Sample a permutation of data points say B, C, A, D
- scan from left to right in one such sample of permutation $B- > C- > A- > D$
- Marginal Contribution for each sample At Step 3, V (B, C, A) − V (B, C), will be less than V (B, C) − V (B) At step 2
- Truncate at a predefined tolerance: Only do $B- > C- > A$ and assign zero as marginal contribution to the rest

Gradient Based Approximation

Algorithm 2 Gradient Shapley

Input: Parametrized and differentiable loss function $\mathscr{L}(.;\theta)$, train data $D = \{1, \ldots, n\}$, performance score function $V(\theta)$

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

  $\theta_0^t \leftarrow$ Random parameters

  $v_0^t \leftarrow V(\theta_0^t)$

  for $j \in \{1, \ldots, n\}$ **do**

    $\theta_j^t \leftarrow \theta_{j-1}^t - \alpha \nabla_\theta \mathscr{L}(\pi^t[j]; \theta_{j-1})$

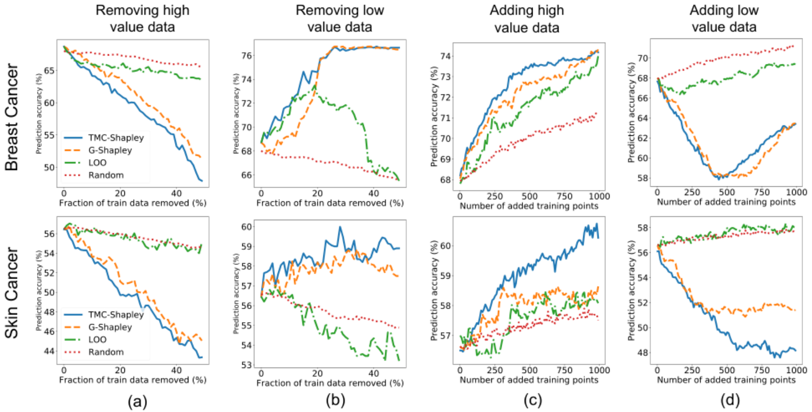
    $v_j^t \leftarrow V(\theta_j^t)$

    $\phi_{\pi^t[j]} \leftarrow \frac{t-1}{t} \phi_{\pi^{t-1}[j]} + \frac{1}{t}(v_j^t - v_{j-1}^t)$

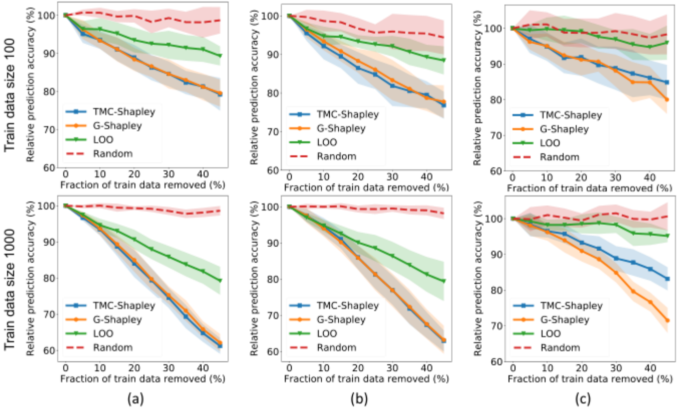
  end for

end for

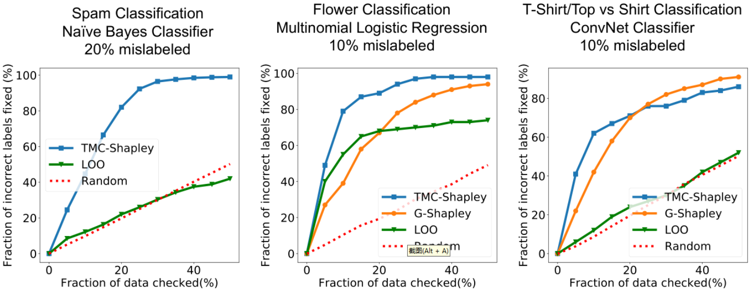
Experiment Result



Synthesis Data



Mislabeled Data



Noisy Image and Group Data

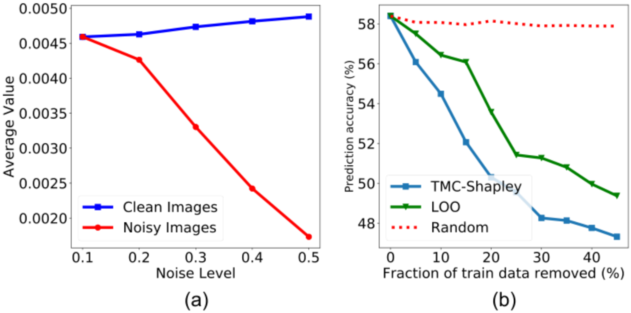


Figure 5. (a) **Value and data quality**: White noise is added to 10% of training points. As the noise level increases, the average TMC-Shapley value of noisy images becomes decreases compared to that of clean images. (b) **Group Shapley:** Removing the valuable groups degrades the performance more than removing groups with the highest leave-one-out score.

Conclusion

- Introduces a fair data pricing approach based on Shapley values to quantify the contribution of each data point to machine learning models.
- the value of individual datum depends on the learning algorithm, evaluation metric as well as other data points in the training set
- Demonstrates the effectiveness of this approach across various tasks and models

Thoughs

- Explore combining Shapley value estimation with different optimization algorithms to achieve faster convergence rates.
- Assuming Shapley values are sparse, investigate the potential of using compressed sensing techniques to reconstruct true Shapley values with fewer samples.

Thanks!