L-Shapley and C-Shapley: Efficient Model Interpretation for Structured Data

Jianbo Chen, Le Song, Martin J. Wainwright, Michael I. Jordan

ICLR 2019

February 28, 2020

Presenter: Rishab Bamrara

https://qdata.github.io/deep2Read/

Motivation:

- Modern machine learning models, including random forests, deep neural networks, and kernel methods, can produce high accuracy prediction in many applications. However, the accuracy in prediction from such black box models, comes at the cost of interpretability.

- Ease of interpretation is a crucial criterion when these tools are applied in areas such as medicine, financial markets, and criminal justice

Background:

Approaches for Interpreting Models:

- **Model specific Interpretation:** Make assumptions to the model and hence are specific to the model itself. (Attention weights, smooth-grad, grad-CAM)
- **Model Agnostic Interpretation:** Making no assumptions about the underlying model. Can be used in any ML model and is applied post-hoc. (Eg. LIME, Shapley value)
- **Instance-wise Interpretation:** Yielding feature importance for each input instance. (E.g. Saliency Map, CD)
- **Model-level Interpretation:** Yielding feature importance for the whole model. (E.g. Weights of NN, Decision Rules for Decision Trees)

This study focuses on Model Agnostic and Instance-wise Interpretation.

Background:

Shapley Value:

- Axiomatic characterization of a fair distribution of a total surplus from all the players.
- Can be applied in to predictive models.
- Each feature is modeled as a player in the underlying game.

For quantifying the importance of a given feature index i for feature vector x $\in R^d$, we can compute importance score of feature i, $v_x(\{i\})$, on its own.

However, doing so ignores interactions between features, which are likely to be very important in applications.

Background:

Marginal Contribution: For a given subset S containing i, compute the difference between the importance of all features in S, with and without i.

$$m_x(S, i) := v_x(S) - v_x(S \setminus \{i\})$$

In order to obtain a simple scalar measure for feature i, we need to aggregate these marginal contributions over all subsets that contain i.

$$\phi_x(\mathbb{P}_m, i) := \underbrace{\frac{1}{d}\sum_{k=1}^{d}}_{\text{Average over } k} \underbrace{\frac{1}{\binom{d-1}{k-1}}\sum_{S \in \mathcal{S}_k(i)}}_{\text{Average of over } \mathcal{S}_k(i)} m_x(S, i).$$

$S_k(i)$ denote the set of k-sized subsets that contain i.

Background:

- Properties of Shapley Value:

Additivity: The sum of the Shapley values $\sum_{i=1}^{d} \phi_x(i)$ is equal to the difference $v_x(\{1,\ldots,d\}) - v_x(\emptyset)$.

Equal contributions: If $v_x(S \cup \{i\}) = v_x(S \cup \{j\})$ for all subsets S, then $\phi_x(i) = \phi_x(j)$.

Monotonicity: Given two models $\mathbb{P}_m$ and $\mathbb{P}_m$, let m_x and m'_x denote the associated marginal contribution functions, and let ϕ_x and ϕ'_x denote the associated Shapley values. If $m_x(S,i) \geq m'_x(S,i)$ for all subsets S, then we are guaranteed that $\phi_x(i) \geq \phi'_x(i)$.

- Challenge with computing Shapley values:
  - The exact computation of the Shapley value φx(i) takes into account the interaction of feature i with all 2^{d-1} subsets that contain i, thereby leading to computational difficulties.

Related Work:

1. Lloyd S Shapley. (1953): A value for n-person games. Contributions to the Theory of Games

1. Štrumbelj et al. (2010): An efficient explanation of individual classifications using game theory.

1. Datta et al. (2017): Algorithmic transparency via quantitative input influence: Theory and experiments with learning systems.

1. Lundberg et al. (2009): A unified approach to interpreting model predictions.

- **Monte Carlo Approximation and Weighted linear regression.**

Claim / Target Task:

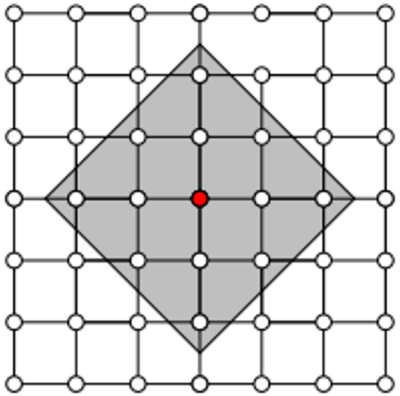
- Sampling-based approximations may suffer from high variance when the number of samples to be collected per instance is limited.
- For large-scale predictive models, the number of features is often relatively large, meaning that the number of samples required to obtain stable estimates can be prohibitively large.
- Authors address this challenge in a model-based paradigm, where the contribution of features to the response variable respects the structure of an underlying graph.

Method:

- Associate features with the nodes of a graph. E.g. sequence data can be associated with a line graph and grid graph for image data.
- Notations:
  - $\mathbf{x \in R^d} \Rightarrow$ feature vectors
  - $\mathbf{G = (V, E)} \Rightarrow$ Graph with nodes V and edges $E \subset V \times V$
  - Each feature i is associated with a node $i \in V$
  - Edges represent interactions between features
  - $\mathbf{d_G(l, m)} \Rightarrow$ number of edges in shortest path joining l to m.
  - **K-neighborhood** $\Rightarrow \mathbf{N_k(i)} := \{j \in V \mid d_G(i, j) \leq k\}$

Method:

(a) Illustration of the k = 2 graph neighborhood $N_2(i)$ on the grid graph. All nodes within the shaded gray triangle lie within the neighborhood $N_2(i)$.



(a)

Local Shapley (L-Shapley)

- Words distant have a weaker influence on the importance of a given word in a document, and therefore have relatively less effect on the Shapley score.

Definition 1. *Given a model $\mathbb{P}_m$, a sample x and a feature i, the* L-Shapley estimate of order k *on a graph G is given by*

$$\hat{\phi}_x^k(i) := \frac{1}{|\mathcal{N}_k(i)|} \sum_{\substack{T \ni i \\ T \subseteq \mathcal{N}_k(i)}} \frac{1}{\binom{|\mathcal{N}_k(i)|-1}{|T|-1}} m_x(T, i). \tag{5}$$

Original Formula:

$$\phi_x(\mathbb{P}_m, i) := \underbrace{\frac{1}{d} \sum_{k=1}^{d}}_{\text{Average over } k} \underbrace{\frac{1}{\binom{d-1}{k-1}} \sum_{S \in \mathcal{S}_k(i)}}_{\text{Average of over } \mathcal{S}_k(i)} m_x(S, i).$$

Local Shapley (L-Shapley)

- Coefficients of $m_x(S, i)$ are chosen to match the coefficients in the definition of the Shapley value restricted to the neighborhood $N_k(i)$.
- This controls the error under certain probabilistic assumption.
- The choice of the integer k is dictated by computational considerations.
- Evaluating all d L-Shapley scores on a line graph requires $2^{2k}d$ model evaluations. (2^{2k+1} per feature).
- A grid graph requires $2^{4k*4k}d$ function evaluations.

Connected Shapley (C-Shapley)

- Further reduces the complexity of approximating the Shapley value.
- Only look at the connected subsets.
  - *It is not heartwarming or entertaining. It just sucks.*
  - Subset "*It not heartwarming,*" rarely appears in real data and may not make sense to a human or a model trained on real-world data.

Definition 2. *Given a model $\mathbb{P}_m$, a sample x and a feature i, the* C-Shapley estimate of order k *on a graph G is given by*

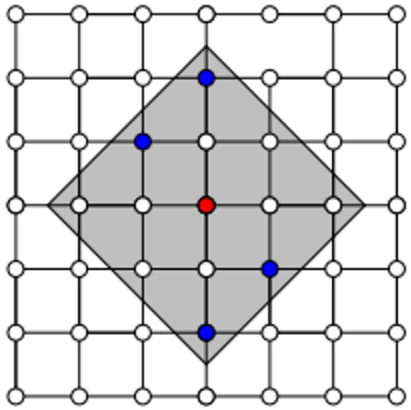
$$\tilde{\phi}_x^k(i) := \sum_{U \in \mathcal{C}_k(i)} \frac{2}{(|U|+2)(|U|+1)|U|} m_x(U,i), \tag{6}$$

where $\mathcal{C}_k(i)$ denotes the set of all subsets of $\mathcal{N}_k(i)$ that contain node i, and are connected in the graph G.

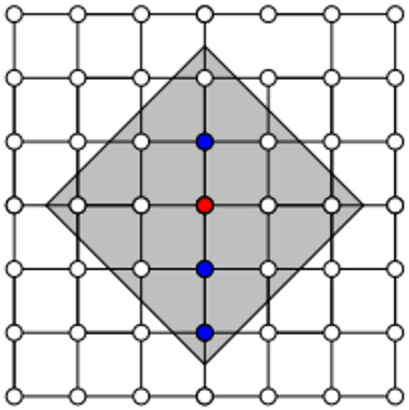
Connected Shapley (C-Shapley)

- Coefficients of $m_x(S, i)$ are a result of using Myerson value.
- The error between C-Shapley and the Shapley value can also be controlled under certain statistical assumptions.
- For text data, C-Shapley is equivalent to only evaluating n-grams in a neighborhood of the word to be explained.
- C-Shapley scores for all d features takes $O(k^2 d)$ model evaluations on a line graph.

Difference b/w L-Shapley and C-Shapley



(b)



(c)

(b) A disconnected subset of $N_2(i)$ that is summed over in L-Shapley but not C-Shapley.

(c) A connected subset of $N_2(i)$ that is summed over in both L-Shapley and C-Shapley.

Approximation of Shapley value

1. The expected error between the L-Shapley estimate $\hat{\varphi}^k_X(i)$ and the true Shapley-value-based importance score $\varphi_i(P_m, x)$ is bounded by 4ε:

$$E_X|\hat{\varphi}^k_X(i) - \varphi_X(i)| \leq 4\varepsilon.$$

1. The expected error between the C-Shapley estimate $\hat{\varphi}^k_X(i)$ and the true Shapley-value-based importance score $\varphi_i(Pm, x)$ is bounded by 6ε:

$$E_X|\hat{\varphi}^k_X(i) - \varphi_X(i)| \leq 6\varepsilon$$

Speeding Up Calculation:

1. For considering the interaction of features in a large neighborhood $N_k(i)$ with a feature i, exponential complexity in k can become a barrier.
2. Sampling based on random permutation (Štrumbelj et. al.) of local features may be used to alleviate the computational burden of L-Shapley.
3. A regression-based estimate of C-Shapley : $\tilde{\phi}_x^k \approx (X^T W X)^{-1} X^T F.$

   Where, $X \in \{0, 1\}^{kd \times d}$ and a response vector $F \in R^{kd}$, where $X_{ij} = 1$ if the jth feature is included in the ith sample, and $F_i = v_x(S_i)$, the score function evaluated on the corresponding feature subset.

Problem Setting

- Input:
  - Model
  - Instance
- Output:
  - A vector of importance score of the feature.

- The instance-wise property means that this vector, and hence the relative importance of each feature, is allowed to vary across instances.

Experiments:

- Task: Image Classification and Text Classification
- Baselines: Model Agnostic Methods
  - **KernelSHAP:** Regression based approximation of Shapley.
  - **SampleShapley:** Random sampling based approximation.
  - **LIME:** Linear model to locally approximate the original model.
- **Saliency Map:** Image Data

Datasets for Text Classification:

Data Set	Classes	Train Samples	Test Samples	Average #w	Model	Parameters	Accuracy
IMDB Review [15]	2	25,000	25,000	325.6	WordCNN	351,002	90.1%
AG's News [25]	4	120,000	7,600	43.3	CharCNN	11,337,988	90.09%
Yahoo! Answers [25]	10	1,400,000	60,000	108.4	LSTM	7,146,166	70.84%

Table 1. A summary of data sets and models in three experiments. "Average #w" is the average number of words per sentence. "Accuracy" is the model accuracy on test samples.

- IMDB Review ⇒ Word-CNN
- AG News ⇒ Char-CNN
- Yahoo! Answers ⇒ LSTM

L-Shapley ⇒ Interaction of each word i with the two neighboring words in $N_1(i)$

C-Shapley ⇒ Regression-based version on all n-grams with $n \leq 4$.

Results:

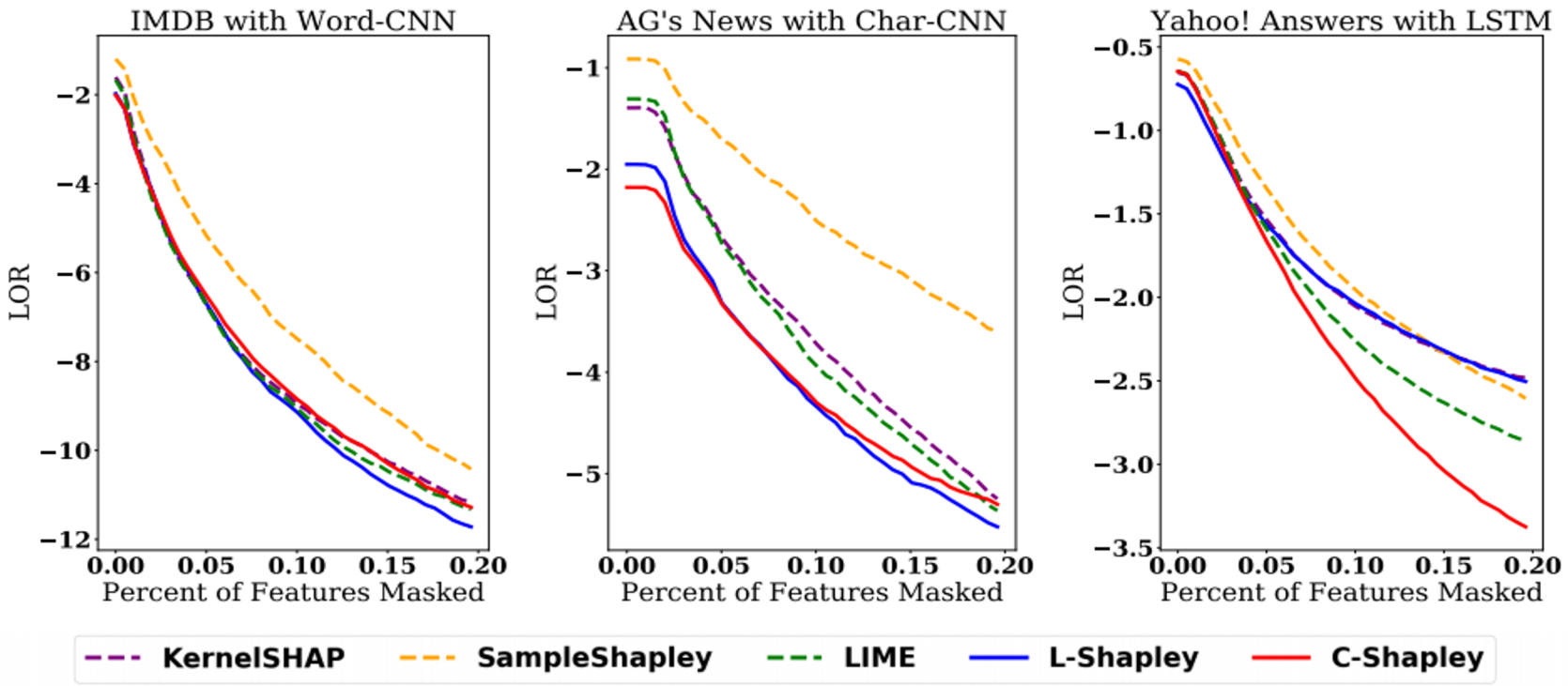


Figure 2. The above plots show the change in log odds ratio of the predicted class as a function of the percent of masked features, on the three text data sets. Lower log odds ratios are better.

- On IMDB with Word-CNN, L-Shapley achieves the best performance while LIME, KernelSHAP and C-Shapley achieve slightly worse performance.
- On AG's news with Char-CNN, L-Shapley and C-Shapley both outperform other algorithms.
- On Yahoo! Answers with LSTM, C-Shapley outperforms the rest of the algorithms by a large margin

Results:

Importance scores produced by different Shapley-based methods on Example:
"It is not heartwarming or entertaining. It just sucks".

Method	Explanation
Shapley	It is not heartwarming or entertaining . It just sucks .
C-Shapley	It is not heartwarming or entertaining . It just sucks .
L-Shapley	It is not heartwarming or entertaining . It just sucks .
KernelSHAP	It is not heartwarming or entertaining . It just sucks .
SampleShapley	It is not heartwarming or entertaining . It just sucks .

Table 2. Each word is highlighted with the RGB color as a linear function of its importance score. The background colors of words with positive and negative scores are linearly interpolated between blue and white, red and white respectively.

Datasets for Image Classification:

- MNIST
- CIFAR10

Evaluation:

- SampleShapley
- KernelSHAP
- Saliency
- C_Shapley

- LIME and L-Shapley are not used for comparison.
- LIME uses superpixels instead of raw pixels.
- L-Shapley was not chosen because of evaluation constraints.
- For C-Shapley, applied regression-based version to evaluate all n×n image patches with $n \leq 4$.

Results:

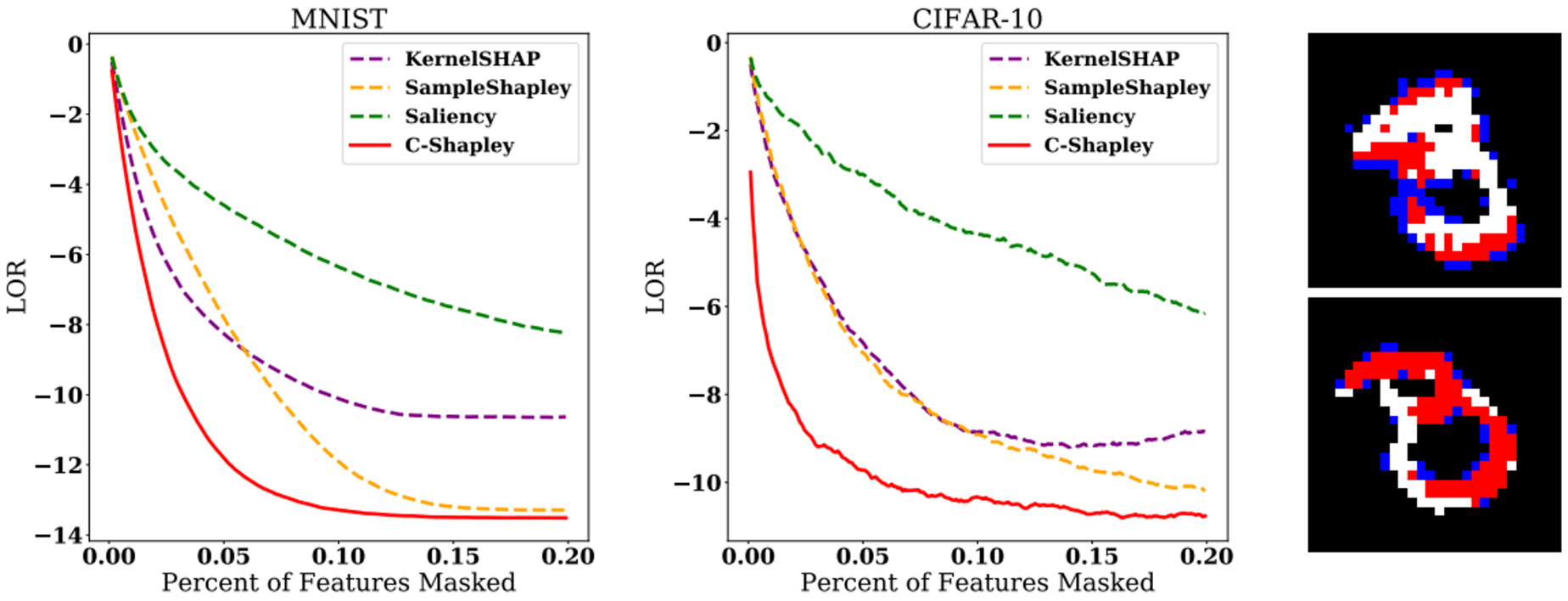


Figure 3. Left and Middle: change in log-odds ratio vs. the percent of pixels masked on MNIST and CIFAR10. Right: top pixels ranked by C-Shapley for a "3" and an "8" misclassified into "8" and "3" respectively. The masked pixels are colored with red if activated (white) and blue otherwise.

Interestingly, the top pixels chosen by C-Shapley visualize the "reasoning" of the model: more specifically, the important pixels to the model are exactly those which could form a digit from the opposite class.

Results:

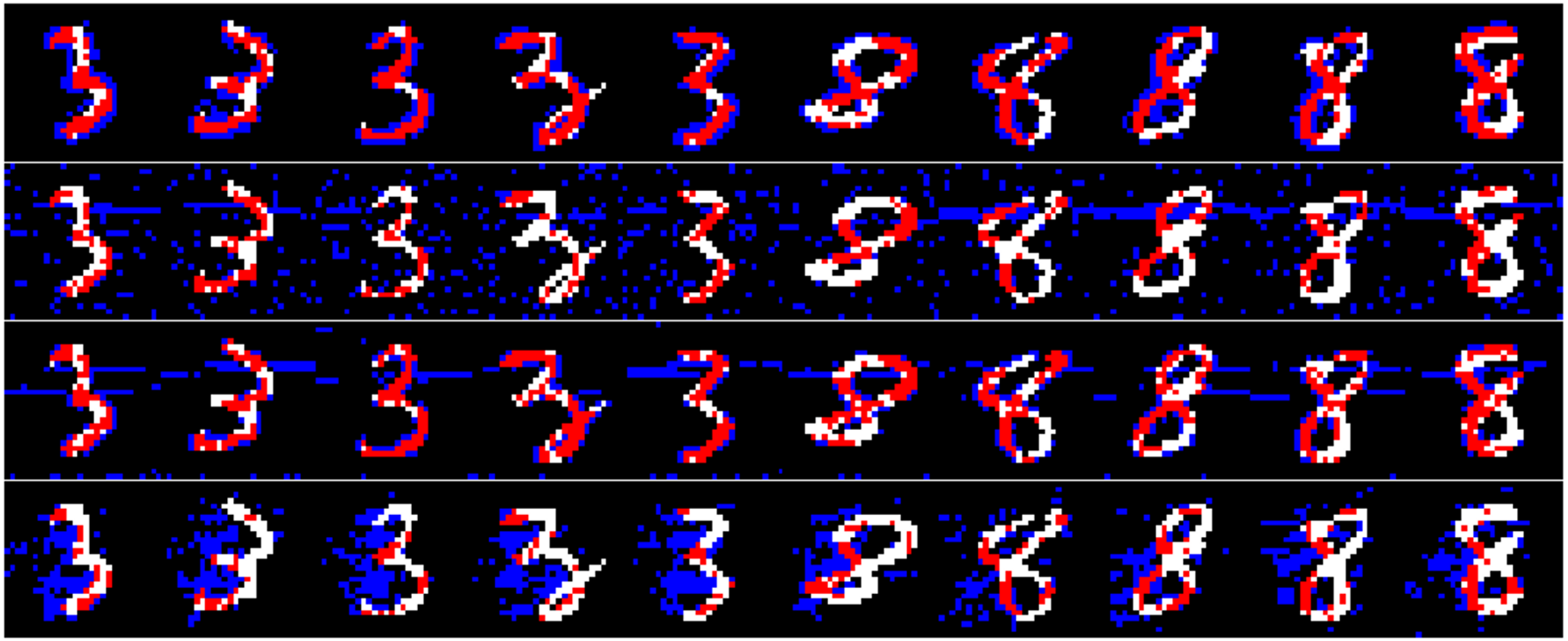


Figure 4. Some examples of explanations obtained for the MNIST data set. The first row corresponds to the original images, with the rows below showing images masked based on scores produced by C-Shapley, KernelSHAP, SampleShapley and Saliency respectively. For best visualization results, 15% and 20% of the pixels are masked for each image. The masked pixels are colored with red if activated (white) and blue otherwise.

- Pixels picked by C-Shapley concentrate around and inside the digits in MNIST.

Results:

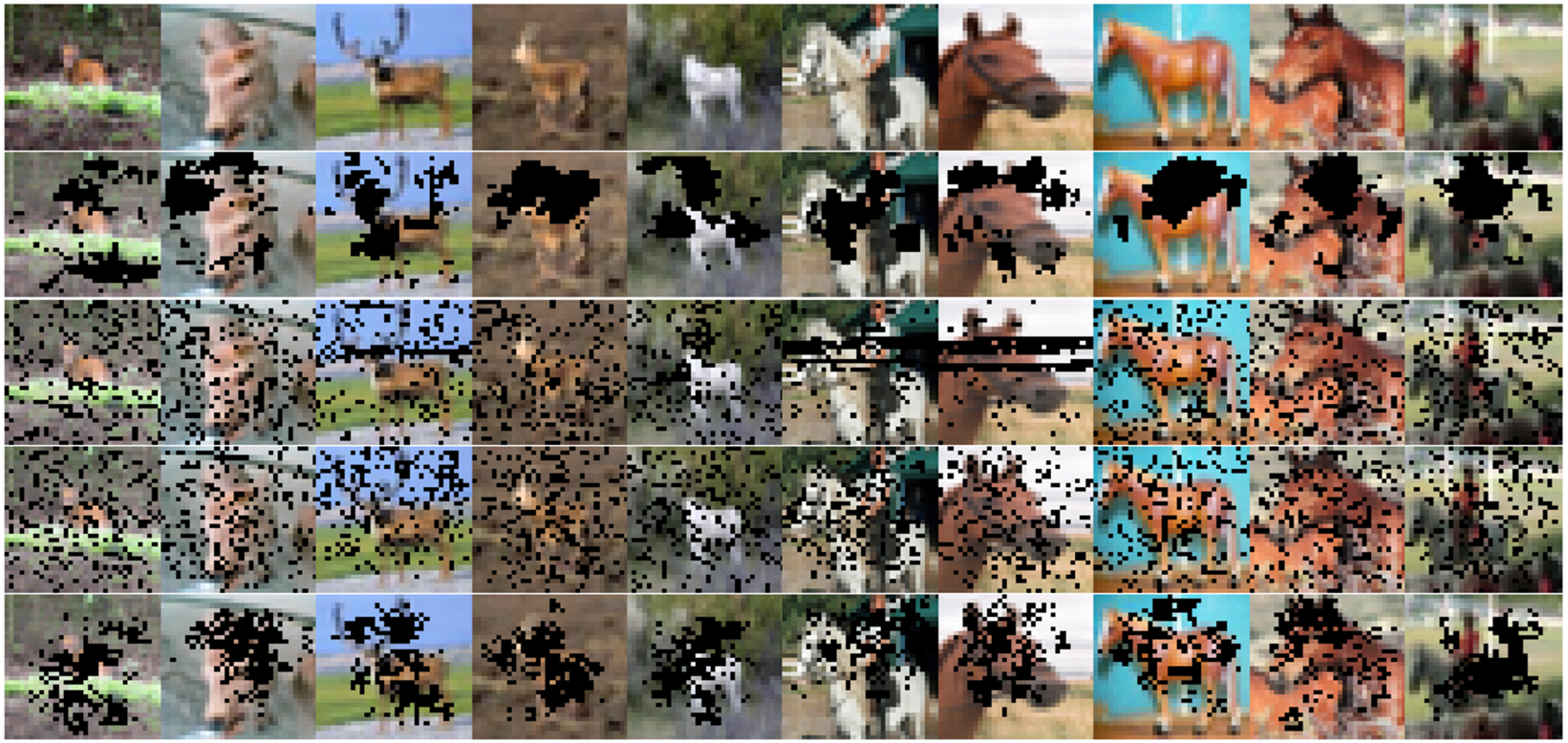


Figure 5. Some examples of explanations obtained for the CIFAR10 data set. The first row corresponds to the original images, with the rows below showing images masked based on scores produced by C-Shapley, KernelSHAP, SampleShapley and Saliency respectively. For best visualization results, 20% of the pixels are masked for each image.

- The C-Shapley and Saliency methods yield the most interpretable results in CIFAR10. In particular, C-Shapley tends to mask the parts of head and body that distinguish deers and horses, and the human riding the horse.

Conclusion:

- Authors have proposed L-Shapley and C-Shapley for instance-wise feature importance scoring, making use of a graphical representation of the data.
- Shown the superior performance of the proposed algorithms compared to other methods for instance-wise feature importance scoring in text and image classification.

References:

- Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. PloS One, 10(7):e0130140, 2015.
- David Baehrens, Timon Schroeter, Stefan Harmeling, Motoaki Kawanabe, Katja Hansen, and Klaus-Robert Müller. How to explain individual classification decisions. Journal of Machine Learning Research, 11:1803–1831, 2010.
- Anupam Datta, Shayak Sen, and Yair Zick. Algorithmic transparency via quantitative input influence: Theory and experiments with learning systems. In Security and Privacy (SP), 2016 IEEE Symposium on, pages 598–617. IEEE, 2016.
- Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, Advances in Neural Information Processing Systems 30, pages 4765–4774. Curran Associates, Inc., 2017.
- Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Why should I trust you?: Explaining the predictions of any classifier. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pages 1135–1144. ACM, 2016.
- Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. In ICML, volume 70 of Proceedings of Machine Learning Research, pages 3145–3153. PMLR, 06–11 Aug 2017.
- Erik Štrumbelj and Igor Kononenko. An efficient explanation of individual classifications using game theory. Journal of Machine Learning Research, 11:1–18, 2010.
- Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In International Conference on Machine Learning, pages 3319–3328, 2017.