Is this the Right Neighborhood? Accurate and Query Efficient Model Agnostic Explanations

Amit Dhurandhar, Karthikeyan Ramamurthy
IBM Research
and Karthikeyan Shanmugam
Google

NeurIPS 2022

Is this the Right Neighborhood? Accurate and Query Efficient Model Agnostic Explanations

Amit Dhurandhar, Karthikeyan Ramamurthy
IBM Research
and Karthikeyan Shanmugam
Google

NeurIPS 2022

SETUP

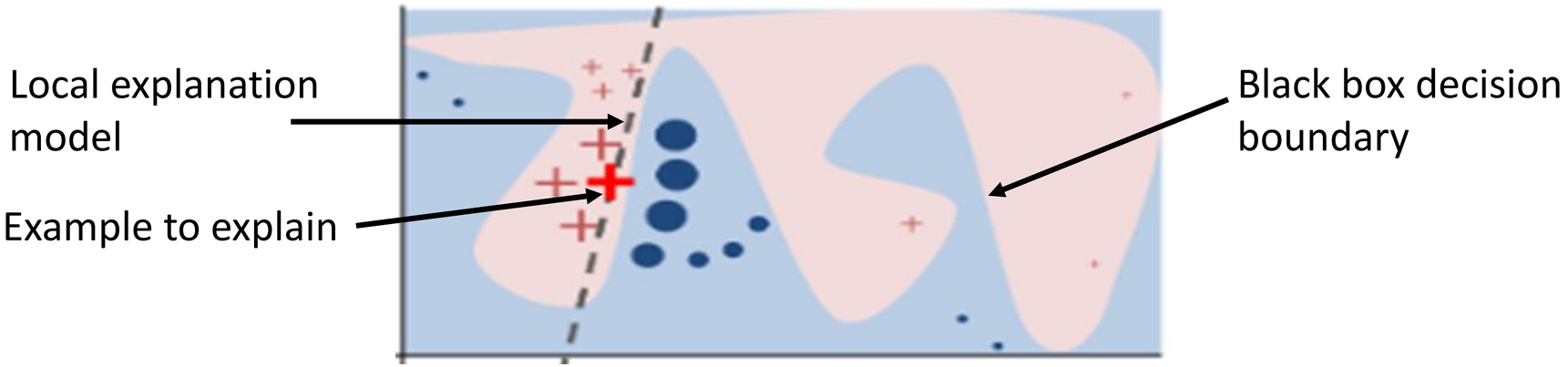
We consider:

- Black box: This means model is not directly interpretable and we have only query access. The latter implies that we can only obtain predictions from the model by passing it an input and observing its output. We have no access to the internals of the model.

- Model agnostic: Any model such as a neural network or random forest etc. which we can query and obtain predictions.

- Classification/regression.

PROBLEM STATEMENT



Good explanations → Fitting an accurate local model → Sampling the right neighborhood

Questions we address:

- How to sample the right neighborhood to obtain faithful explanations?
- How to do it in a query efficient manner?

RELATED WORK

Two main strategies for local model fitting explanation methods:

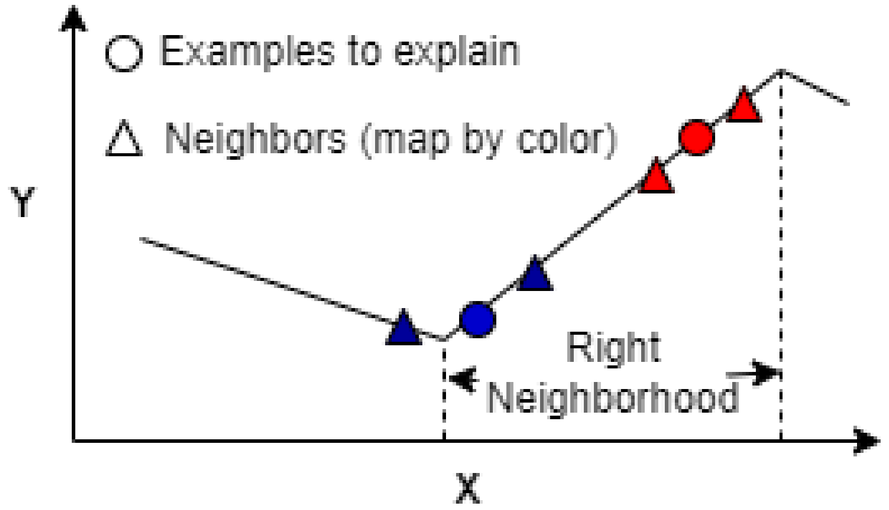
- Random Neighborhood Generation: Methods such as LIME and its variants create neighborhoods around the example to explain by randomly perturbing it.

- Realistic Neighborhood Generation: Methods such as MeLIME learn the data manifold and perturb the latent representation of the example to explain and decode. While methods such as MAPLE find nearby train/test examples. This is considered as a **solution** to create faithful explanations.

Others that do not do local model fitting (viz. SHAP and variants) however, have other issues such as figuring out null/base values for each feature etc.

MOTIVATION AND CONTRIBUTION

Conceptual Contribution: Irrespective of the neighborhood generation scheme (i.e. random or realistic) neighbors could belong to different simple functions (viz. different linear pieces) in a non-linear function (viz. piecewise linear function as seen in deep Relu networks) and so one must find the “right” region for local model fitting (i.e. finding the linear region if doing lasso type fit as in LIME).



METHODOLOGICAL CONTRIBUTION

Our main idea is to

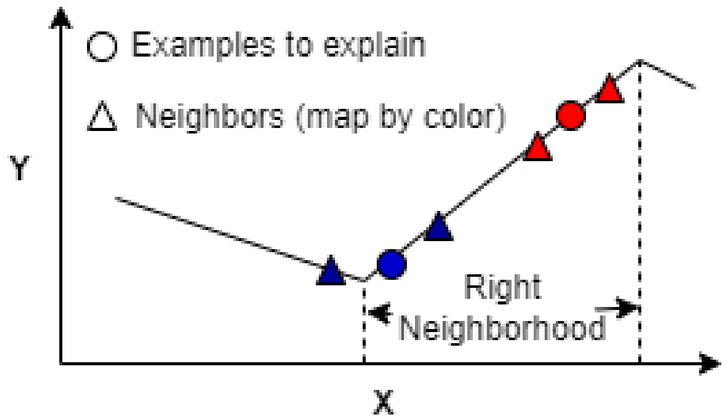
- take a (random/realistic) neighborhood,
- run multidimensional piecewise linear segmented regression (MPLSR)
- find the region that is (approx.) linear around the example to explain
- fit lasso to **only** neighbors that lie in the region rejecting others

If the local model you wish to fit is polynomial you could run multidimensional piecewise *polynomial* segmented regression

METHODOLOGICAL CONTRIBUTION

However, two problems still remain

- Query complexity is still the size of the neighborhood
- Lot of neighborhood samples could be wasted (i.e. not used for local model fitting) if the example lies close to a non-linearity (like the blue circle below).



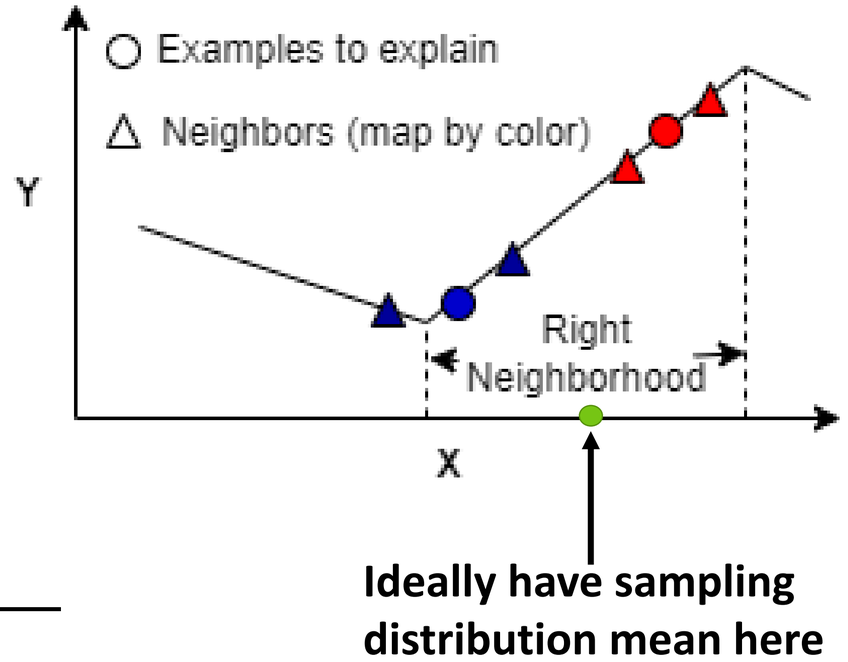
Given this we propose the following Adaptive Neighborhood Sampling (ANS) scheme:

Algorithm 1: Adaptive neighborhood sampling (ANS). Estimation of $\boldsymbol{a}$ and $\boldsymbol{b}$ can also be performed not just once but multiple times and the latest estimates can be used for future sampling. For realistic perturbations sampling can be done in the latent space. More details in section 3.1.

Input: Example to explain $\boldsymbol{\mu}$, black-box predictor $f(.)$, maximum number of neighbors generated N, standard deviation σ and number top features to output k

Set $Q = \phi$ # Examples to query
Sample n ($<< N$) examples from $\mathcal{N}(\boldsymbol{\mu}, \sigma^2 \boldsymbol{I})$ and query $f(.)$
Find region $[\boldsymbol{a_n}, \boldsymbol{b_n}]$ that corresponds to $\boldsymbol{\mu}$ using MPLSR methods [9, 7] on the n samples
Add to Q samples that lie in $[\boldsymbol{a_n}, \boldsymbol{b_n}]$
Estimate uncertainty α # Could be set $\propto \frac{1}{\sqrt{n}}$ or based on stability of the region (i.e. $1 - \rho$)
Sample $N - n$ examples from $\mathcal{N}\left(\alpha\boldsymbol{\mu} + (1-\alpha)\frac{\boldsymbol{a_n}+\boldsymbol{b_n}}{2}, \sigma^2 \boldsymbol{I}\right)$
Add to Q samples that lie in $[\boldsymbol{a_n}, \boldsymbol{b_n}]$
Query $f(.)$ on these additional samples added to Q
Fit interpretable model (viz. sparse linear) $l(.)$ to $(\boldsymbol{x}, f(\boldsymbol{x}))$ where $\boldsymbol{x} \in Q$

Output: Top k coefficients of $l(.)$



ANALYSIS

We analyze our algorithm and show that it can be query and sample efficient over our basic approach which we term as ANS-Basic:

$$q_E = \frac{n}{N} + \frac{N-n}{N} \frac{P_\alpha(\boldsymbol{x} \in [\boldsymbol{a}, \boldsymbol{b}])}{P_\alpha(\boldsymbol{x} \in [\boldsymbol{a_n}, \boldsymbol{b_n}]) P(\boldsymbol{x} \in [\boldsymbol{a}, \boldsymbol{b}])}$$

$$s_I = \frac{n}{N} + \frac{N-n}{N} \frac{P(\boldsymbol{x} \in [\boldsymbol{a}, \boldsymbol{b}])}{P_\alpha(\boldsymbol{x} \in [\boldsymbol{a}, \boldsymbol{b}])}$$

EXPERIMENTS

On three datasets IRIS, HELOC and CIFAR10 we show efficacy of our approach.

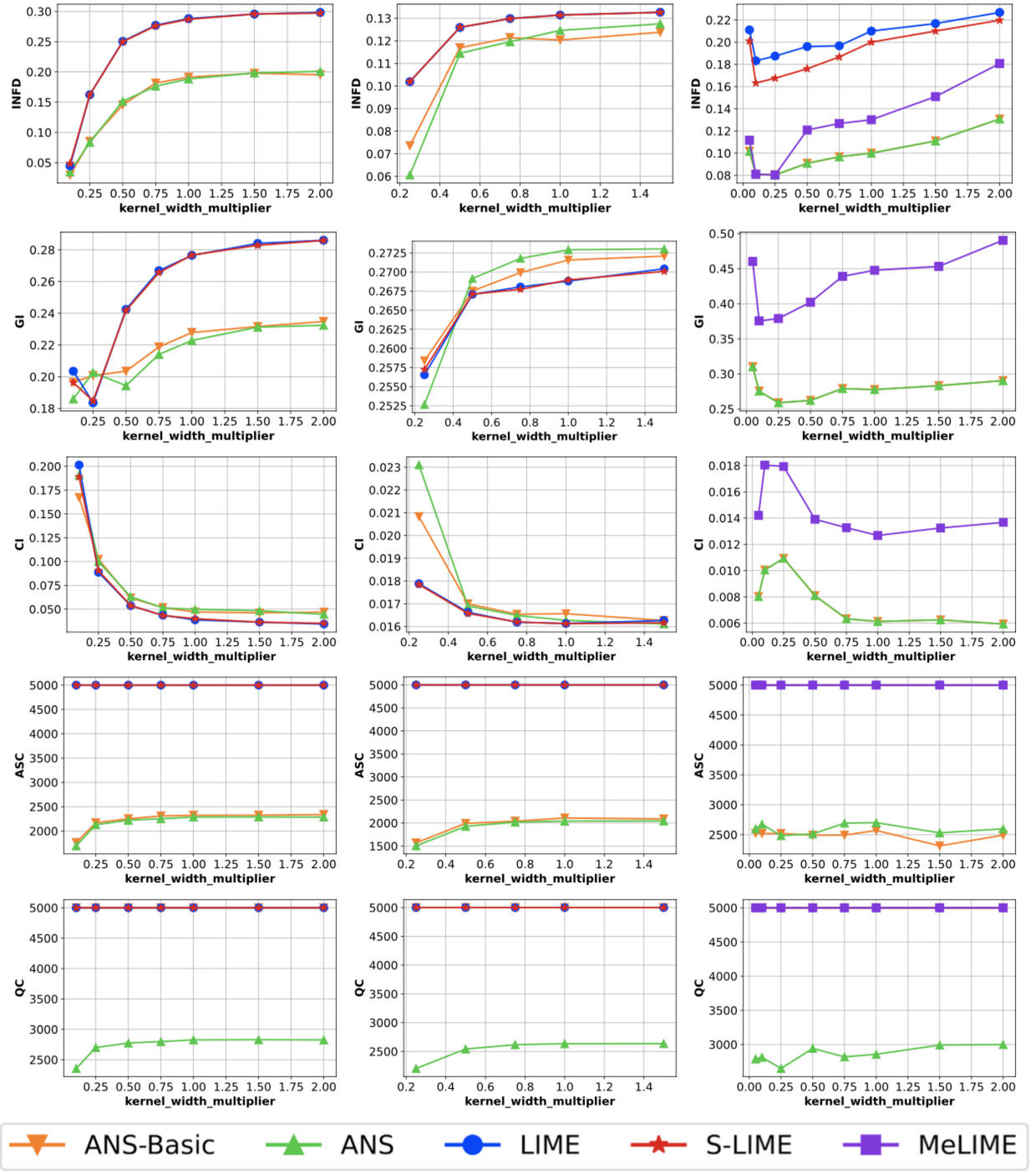
Evaluations:

Quantitative Metrics (lower better): Infidelity (INFD), Generalized Infidelity (GI), Coefficient Inconsistency (CI), Accepted Sample Complexity (ASC) and Query Complexity (QC)

Qualitative: Visual explanations and showing features used

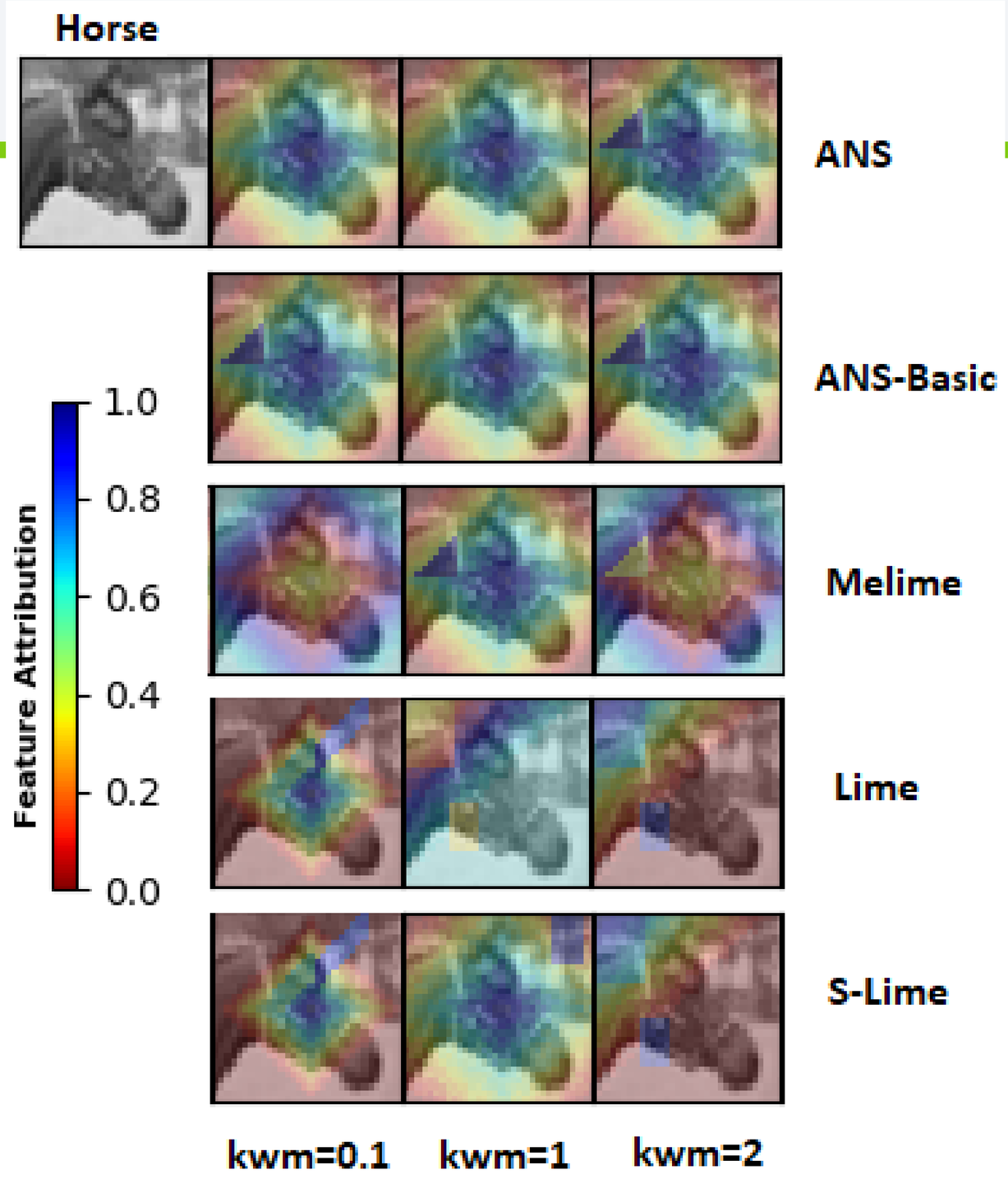
EXPERIMENTS

- The three columns are for the three datasets – IRIS, HELOC, CIFAR.
- Variability with respect to kernel width multiplier which determines neighborhood width shown.



EXPERIMENTS

Feature attributions of our proposed methods ANS and ANS-Basic are much more stable and accurate over different kernel widths compared to other methods



EXPERIMENTS

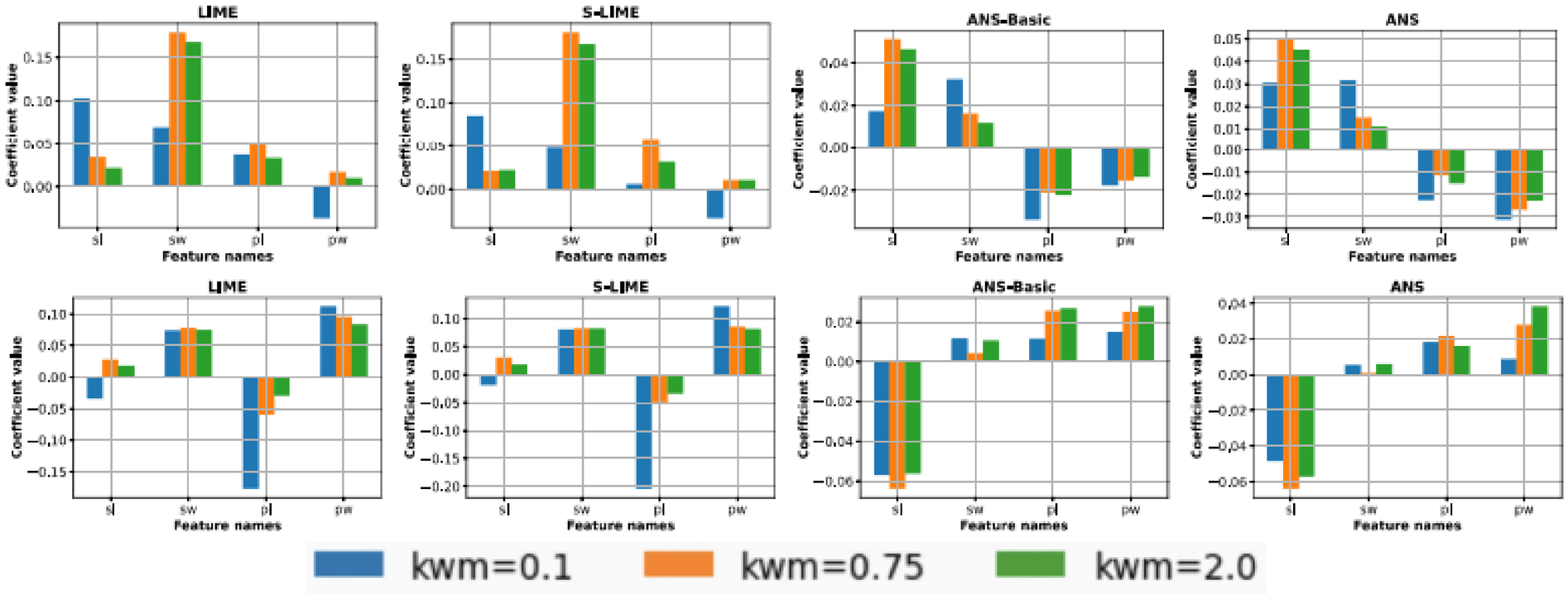


Figure 4: For two examples in the Iris test set corresponding to the two rows of figures above, we provide feature attributions for sepal length (sl), sepal width (sw), petal length and petal width (pw) for three different kernel widths (0.1, 0.75, 2.0). Each row is a separate example and each column corresponds to the method indicated in the title of the subfigure. We can see that across different kernel widths ANS-Basic and ANS feature attributions are much more similar than those seen for LIME or S-LIME. As such, LIME and S-LIME feature attributions even seem to change signs in some cases, while ours do not.

EXPERIMENTS

Time Complexity:

The MPSLR schemes add to the time for ANS but interestingly for **deep models we are actually faster** (for CIFAR10 where we used ResNet18 ANS took ~ 50 secs per example as opposed to a minute for LIME). As the model gets bigger/deeper we conjecture the gain will be even larger.

This is because inference time for deep models is not insignificant and hence **reduced query complexity results in time savings.**

OTHER MOTIVATIONS

Causal motivation: It is known that given a structural causal model (SCM) [22], the best sparse model captures the Markov blanket – i.e. parents and children in a causal graph – of the variable to be estimated. In the post-hoc explanation setting the target variable Y has no children as the black box model is of the form $y = f(x)$. Hence, if the black box i.e. $f(.)$ is linear lasso it could recover the causal parents for some regularization parameter. However, in the non-linear setting, one could have a case where multiple linear pieces explain the causal relationship in different parts of the domain. If one tries to fit a linear model, gradient for one piece with respect to one feature might (approximately) cancel the gradient with respect to another piece for the same feature because the weights are of opposite signs. This could lead the linear model missing some (causal) parents when explaining the variance in y. However, if we are able to identify the correct (linear) regions, then a simple lasso-like fit in each such region should be able to uncover the correct causal parents overall. This work may also motivate a notion of *locally causal*, which may be useful in practice, beyond the standard formalisms of causality [22, 27] which are predominantly global.

Simplicity bias motivation: In a recent paper [28], investigating the reasons for neural networks fitting to spurious correlations in the in-domain data resulting in poor generalization, the authors argue that this is because of the simplicity bias of neural networks. That is, the networks pick the simplest boundary to separate classes which could simply be a linear separator on one feature. However, in the test data, the optimal separator could be based on a more complex decision boundary. Hence, one would ideally want to capture the true complexity of the decision boundary. The definition of complexity they use to analyze arbitrarily complex decision boundaries is closely related to the number of linear pieces that would make up different decision boundaries. This formalization thus further motivates our ANS and ANS-Basic approaches, which identify the appropriate linear component based on a piecewise linear decision boundary.

CONCLUSION

- We make a conceptual and methodological contribution.
- Method is simple having high query and sample efficiency compared to baselines along with good faithfulness and stability properties, and even speed for deep models.
- First adaptive neighborhood generation scheme for local posthoc explanations, that can be used with many local post-hoc explanation methods.
- Is principled as it has causal and neural network behavioral motivations.

Thank you