IML and DALE: Introduction to Interpretable Machine Learning and Differential Accumulated Local Effects

Christos Diou

Department of Informatics and Telematics
Harokopio University of Athens

Presentation at MeVer group,
Information Technologies Institute
4/1/2022

Contents

Introduction

Local, model-agnostic methods

Methods for CNN interpretation

Global, model agnostic methods

DALE

DALE vs ALE

Short introduction to interpretable machine learning [1]

[1]Some ideas and content from Christoph Molnar's book, "Interpretable Machine Learning" (IML book), https://christophm.github.io/interpretable-ml-book/

Hypothetical (?) scenarios

- The computer vision subsystem of an autonomous vehicle leads the vehicle to take a left turn, in front of a car moving in the opposite direction[2]

[2] https://www.theguardian.com/technology/2022/dec/22/tesla-crash-full-self-driving-mode-san-francisco

[3] https://www.technologyreview.com/2021/06/17/1026519/racial-bias-noisy-data-credit-scores-mortgage-loans-fairness-machine-learning/

[4] https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing

Hypothetical (?) scenarios

- The computer vision subsystem of an autonomous vehicle leads the vehicle to take a left turn, in front of a car moving in the opposite direction[2]
- The credit assessment system leads to the rejection of an application for a loan - the client suspects racial bias[3]

[2] https://www.theguardian.com/technology/2022/dec/22/tesla-crash-full-self-driving-mode-san-francisco

[3] https://www.technologyreview.com/2021/06/17/1026519/racial-bias-noisy-data-credit-scores-mortgage-loans-fairness-machine-learning/

[4] https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing

Hypothetical (?) scenarios

- The computer vision subsystem of an autonomous vehicle leads the vehicle to take a left turn, in front of a car moving in the opposite direction[2]
- The credit assessment system leads to the rejection of an application for a loan - the client suspects racial bias[3]
- A model that assesses the risk of future criminal offenses (and used for decisions on parole sentences) is biased against black prisoners[4]

[2] https://www.theguardian.com/technology/2022/dec/22/tesla-crash-full-self-driving-mode-san-francisco

[3] https://www.technologyreview.com/2021/06/17/1026519/racial-bias-noisy-data-credit-scores-mortgage-loans-fairness-machine-learning/

[4] https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing

Questions

- Why did a model make a specific decision?
- What could we change so that the model will make a different decision?
- Can we summarize and predict the model's behavior?

Interpretability of Machine Learning Models

Qualitative definitions:

- "Interpretability is the degree to which a human can understand the cause of a decision" [5]
- "Interpretability is the degree to which a human can consistently predict the model's result"[6]
- "Extraction of relevant knowledge from a machine-learning model concerning relationships either contained in data or learned by the model"[7]

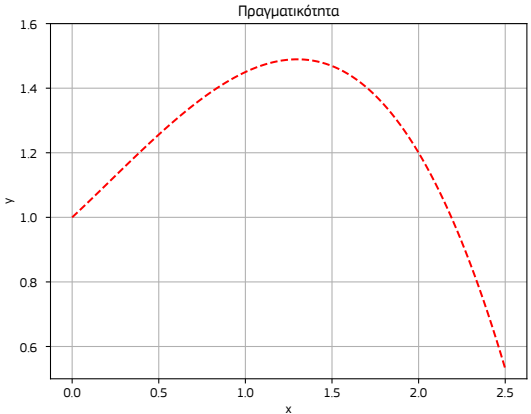
[5]Miller, Tim. "Explanation in artificial intelligence: Insights from the social sciences." arXiv Preprint arXiv:1706.07269. (2017)

[6]Kim, Been, Rajiv Khanna, and Oluwasanmi O. Koyejo. "Examples are not enough, learn to criticize! Criticism for interpretability." Advances in Neural Information Processing Systems (2016).

[7]Murdoch, W. J., Singh, C., Kumbier, K., Abbasi-Asl, R. and Yu, B. "Definitions, methods, and applications in interpretable machine learning." Proceedings of the National Academy of Sciences, 116(44), 22071-22080. (2019)

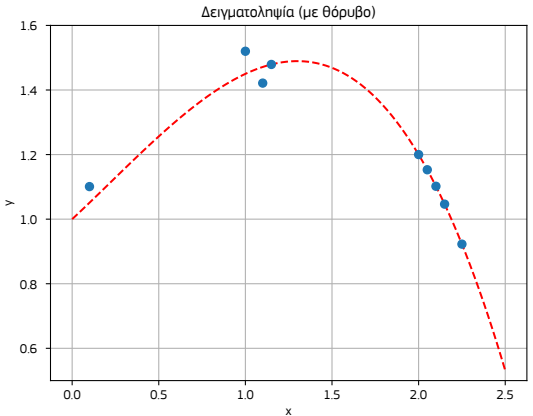
Generalization

To get some intuition, consider a process that produces output y for scalar input x



Generalization

Unfortunately this process is uknown to us, but we can sample a small number of input - output pairs. During sampling, we have a small amount of measurement noise (same if the process is stochastic)



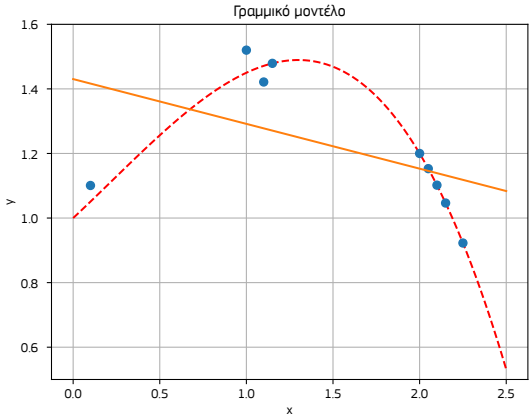
Generalization

Our goal is to model the process using the available samples (regression)

Generalization

Linear model:

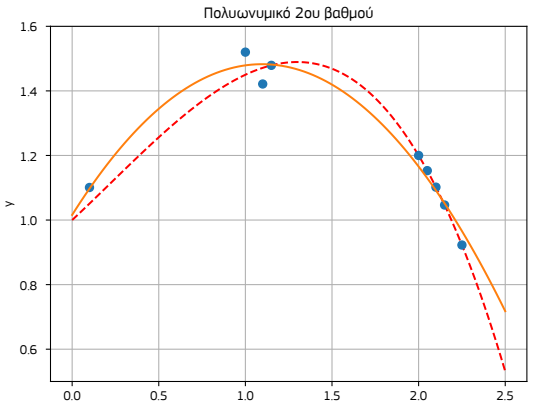
$$y = w_1 \cdot x + w_0$$



Generalization

2^{nd} degree polynomial:

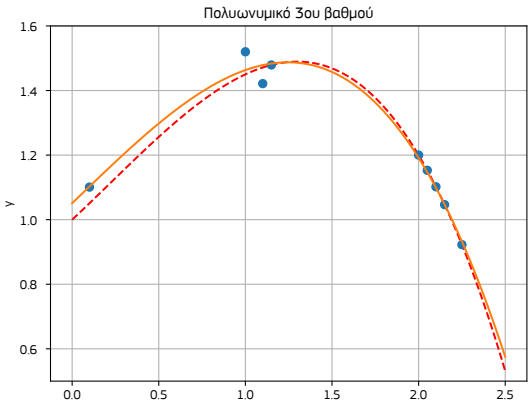
$$y = w_2 \cdot x^2 + w_1 \cdot x + w_0$$



Generalization

3rd degree polynomial:

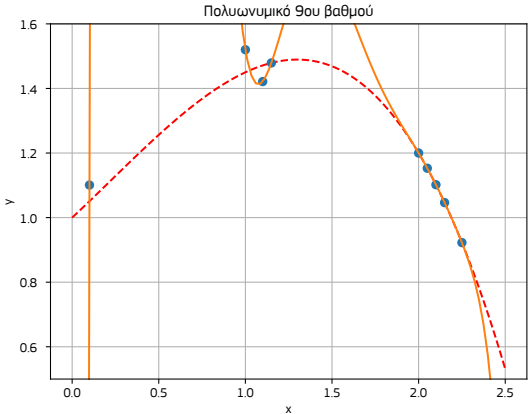
$$y = w_3 \cdot x^3 + w_2 \cdot x^2 + w_1 \cdot x + w_0$$



Generalization

9^{th} degree polynomial

$$y = \sum_{i=0}^{9} w_i \cdot x^i$$



Problem diagnosis

- The model behavior is immediately understood by the shape of the function
- Overfitting is immediately diagnosed
- But what happens if we have multiple dimensions, p, making visualization impossible?
  - We often have tens or hundreds of features
  - Images and signals: Several thousands of input dimensions

Taxonomy of interpretability methods

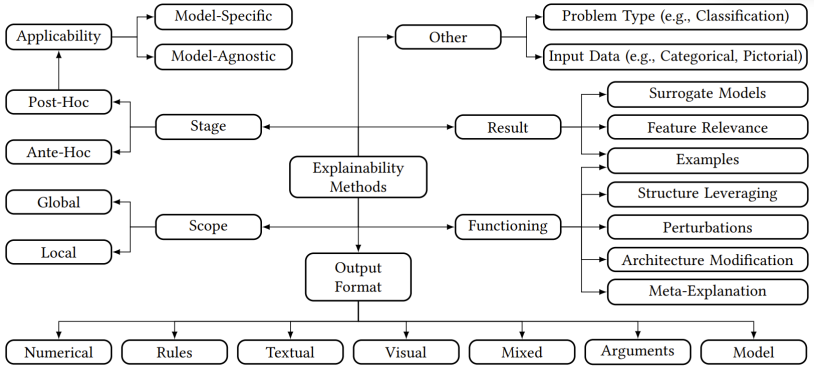


Figure: Timo Speith, "A Review of Taxonomies of Explainable Artificial Intelligence (XAI) Methods". In 2022 ACM Conference on Fairness, Accountability, and Transparency (FAccT '22), 2022

Interpretable models (ante-hoc)

- Some models afford explanations
- Examples, (generalized) linear models, decision trees, k-NN
- Example: Linear regression

$$\hat{y} = w_1 x_1 + \ldots + w_p x_p + b$$

Interpretable models (ante-hoc)

- Result in the bike sharing dataset (model weights)

$$\hat{y} = w_1 x_1 + \ldots + w_p x_p + b$$

	Weight	SE	|t|
(Intercept)	2399.4	238.3	10.1
seasonSPRING	899.3	122.3	7.4
seasonSUMMER	138.2	161.7	0.9
seasonFALL	425.6	110.8	3.8
holidayHOLIDAY	-686.1	203.3	3.4
workingdayWORKING DAY	124.9	73.3	1.7
weathersitMISTY	-379.4	87.6	4.3
weathersitRAIN/SNOW/STORM	-1901.5	223.6	8.5
temp	110.7	7.0	15.7
hum	-17.4	3.2	5.5
windspeed	-42.5	6.9	6.2
days_since_2011	4.9	0.2	28.5

Figure: C. Molnar, IML book

Interpretable models (ante-hoc)

- Feature effects (visualization)

$$effect_j^{(i)} = w_j x_j^{(i)} \tag{1}$$

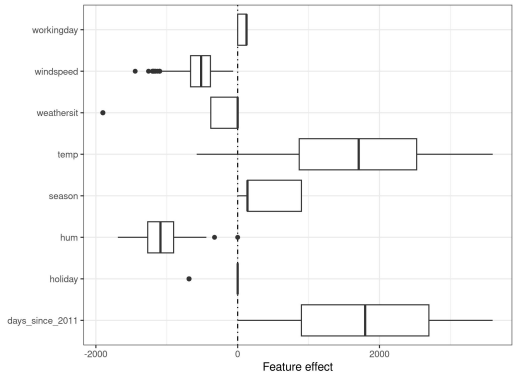


Figure: C. Molnar, IML book

Contents

Introduction

Local, model-agnostic methods

Methods for CNN interpretation

Global, model agnostic methods

DALE

DALE vs ALE

Local, model agnostic methods

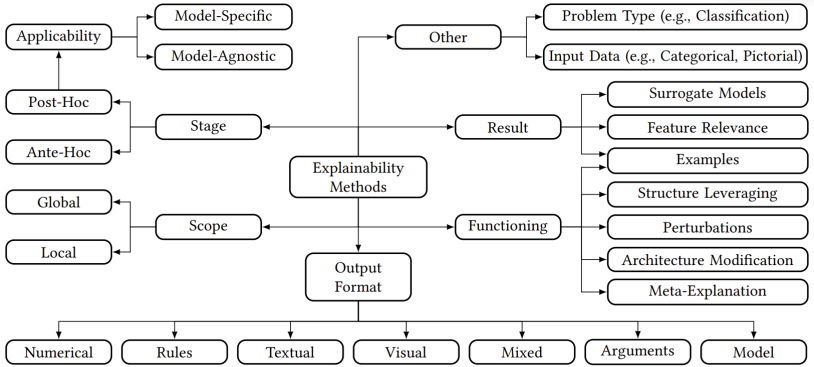


Figure: Timo Speith, "A Review of Taxonomies of Explainable Artificial Intelligence (XAI) Methods". In 2022 ACM Conference on Fairness, Accountability, and Transparency (FAccT '22), 2022

Goal

- Most models do not afford explanations
  - we cannot explain them by looking at their parameters
  - we handle these as “black boxes”
- In this case we apply general interpretability methods
- **Local**: Interpret the model’s output for a particular input instance
- **Global**: Provide a general interpretation of the model’s behavior

LIME - Local Interpretable Model-agnostic Explanations[8]

- Idea:
  - Train an interpretable model with samples in the neighborhood of the target instance, weighted by their proximity

$$\text{explanation(x)} = \arg\min_{g \in G} L(f, g, \pi_x) + \Omega(g)$$

  - g is the interpretable model
  - π_x is the weighting function (e.g., a radial basis function kernel)
  - $\Omega(g)$ is a regularizer for g (e.g., LASSO, or limit on the number of features)

[8] Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin. "Why should I trust you?: Explaining the predictions of any classifier." Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining. ACM (2016)

LIME - Local Interpretable Model-agnostic Explanations[8]

- Idea (visualization)

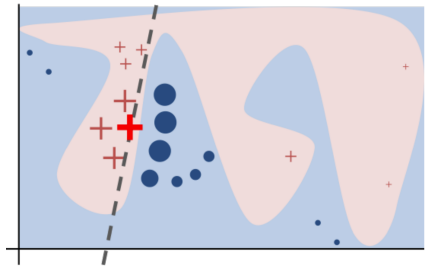


Figure: Ribeiro et al, 2016

[8]Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin. "Why should I trust you?: Explaining the predictions of any classifier." Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining. ACM (2016)

LIME - Local Interpretable Model-agnostic Explanations[8]

- Application on text data

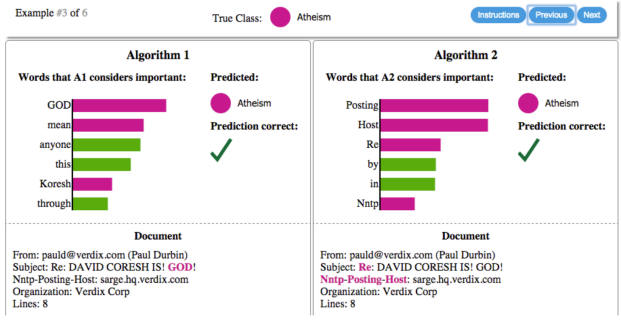


Figure: Ribeiro et al, 2016

[8]Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin. "Why should I trust you?: Explaining the predictions of any classifier." Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining. ACM (2016)

LIME with images

- Instead of features, we use superpixels (e.g., extracted via quick shift)
- We obtain samples by "removing" superpixels (e.g., by replacing their pixels with medium gray)

(a) Original Image (b) Explaining *Electric guitar* (c) Explaining *Acoustic guitar* (d) Explaining *Labrador*

Figure 4: Explaining an image classification prediction made by Google's Inception network, highlighting positive pixels. The top 3 classes predicted are "Electric Guitar" ($p = 0.32$), "Acoustic guitar" ($p = 0.24$) and "Labrador" ($p = 0.21$)

Figure: Ribeiro et al., 2016

SHAP

- Let ϕ_j be the feature attribution of the j-th feature
- Then,

$$g(z') = \phi_0 + \sum_{j=1}^{M} \phi_j z'_j$$

  - $z' \in \{0,1\}^M$ (all 1's for the target instance)
  - General definition - applies to LIME too!

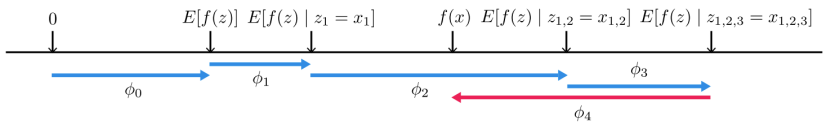


Figure: S.M. Lundberg and S.I Lee. A unified approach to interpreting model predictions. Advances in neural information processing systems, 2017

Kernel SHAP - procedure

1. Sample K binary vectors $z'_k \in \{0,1\}^M$
2. Get a value x' by using mapping function $h_x(z'_k)$
   - Get value of x_j if $z'_j = 1$, get the value from another randomly selected dataset sample if $z'_j = 0$
3. Get prediction $\hat{f}(h_x(z'_k))$
4. Compute weight using SHAP kernel (which has some nice properties - see paper)
5. Fit linear model
6. ϕ_k are the linear model coefficients

Kernel SHAP - procedure

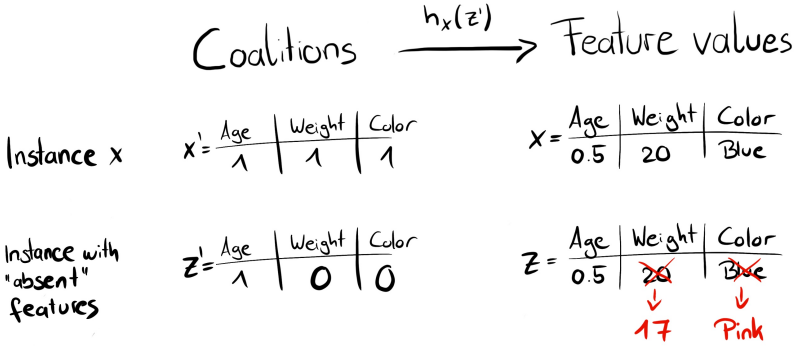


Figure: C. Molnar, IML book

SHAP for images

- Similar idea with LIME, apply SHAP on superpixels

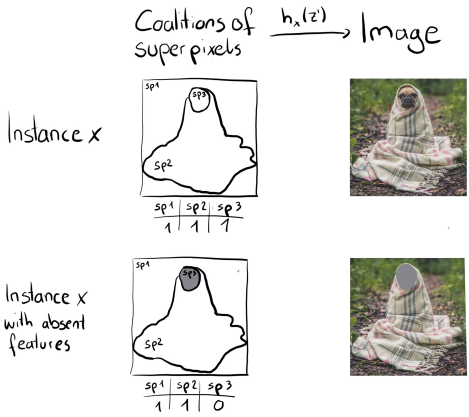


Figure: C. Molnar, IML book

Contents

Introduction

Local, model-agnostic methods

Methods for CNN interpretation

Global, model agnostic methods

DALE

DALE vs ALE

Visualization of extracted features

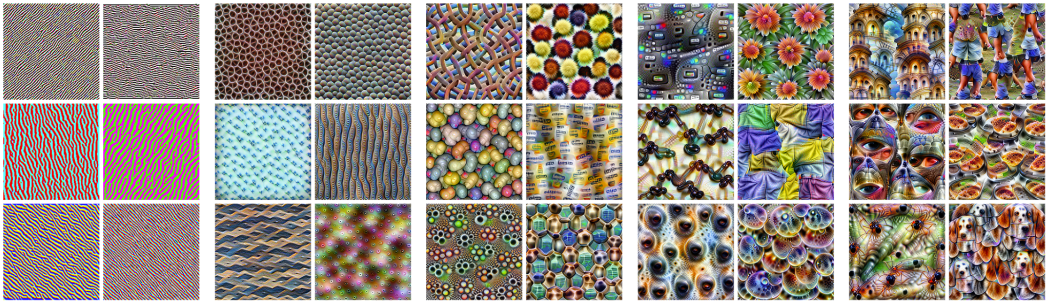


Figure: https://distill.pub/2017/feature-visualization/

- These images maximize the activation of specific filters of an Inception-V1 network at different depths

Pixel attribution / Saliency maps

Methods that visualize the contribution of different areas of an image in the final decision

- Occlusion- or perturbation-based methods)
  - SHAP / LIME belong in this category
- Gradient-based methods

Saliency maps (vanilla gradient)

1. Forward pass of the input image, I_0
2. Compute the derivative of the output/class of interest S_c, with respect to the input

$$E_{grad}(I_0) = \frac{\partial S_c}{\partial I}\Big|_{I=I_0}$$

3. visualize the resulting image

Question: How to handle $ReLU$;

$$X_{n+1} = \max(0, X_n)$$
$$\frac{\partial f}{\partial X_n} = \frac{\partial f}{\partial X_{n+1}} \mathbf{I}(X_n > 0)$$

Saturation

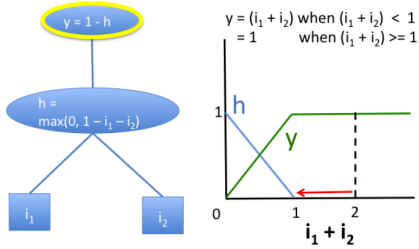


Figure 1. **Perturbation-based approaches and gradient-based approaches fail to model saturation**. Illustrated is a simple network exhibiting saturation in the signal from its inputs. At the point where $i_1 = 1$ and $i_2 = 1$, perturbing either i_1 or i_2 to 0 will not produce a change in the output. Note that the gradient of the output w.r.t the inputs is also zero when $i_1 + i_2 > 1$.

Figure: A. Shrikumar, P. Greenside, and A. Kundaje. "Learning important features through propagating activation differences." Proceedings of the 34th International Conference on Machine Learning-Volume 70. JMLR. org, (2017).

GradCAM

Different approach; starting from the output of the next to last layer (before softmax) for class c, and for activations of features A^k of a layer A (often the last conv layer)

1. Apply global average pooling on derivatives

$$a_k^c = \frac{1}{Z} \sum_i \sum_j \frac{\partial y^c}{\partial A_{i,j}^k}$$

   - Coefficients a_k^c quantify the importance of layer k for detecting class c

2. Use weighted sum of activations to produce the final visualization

$$L_{GradCAM}^c = ReLU\left(\sum_k a_k A^k\right)$$

Example - initial images

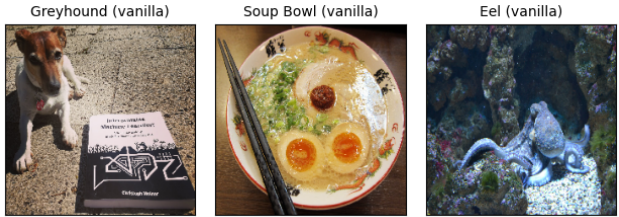


Figure: C. Molnar, IML book

Example - gradient-based interpretations

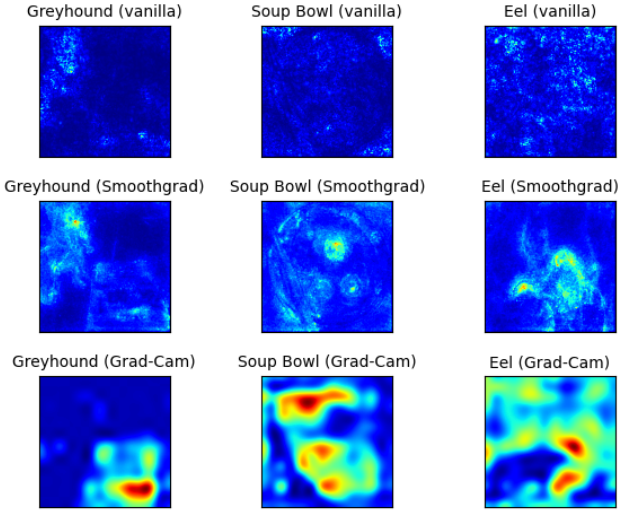


Figure: C. Molnar, IML book

Problems - drawbacks (1)

- Evaluation of these methods is commonly qualitative - we don't know if the interpretation is correct
- It has been demonstrated that these methods are very sensitive [9]
  - Very small changes of the input can lead to the same output but completely different interpretation

[9] A. Ghorbani, A. Abid, and J. Zou. "Interpretation of neural networks is fragile." Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 33. 2019.

Problems - drawbacks (2)

- It has also been shown that some of these methods are unreliable[10]
  - Adding constant pixel offset and changing the bias term of the first layer leads to the same predictions and derivatives, but to different interpretations
- It has finally been show that often these methods do not depend on the model or the data (and are therefore not useful for interpretation), similarly to an edge detector[11]

[10] P-J Kindermans, S. Hooker, J. Adebayo, M. Alber, K. T. Schütt, S. Dähne, D. Erhan and B. Kim. "The (un) reliability of saliency methods." In Explainable AI: Interpreting, Explaining and Visualizing Deep Learning, pp. 267-280. Springer, Cham (2019)

[11] J. Adebayo, J. Gilmer, M. Muelly, I. Goodfellow, M. Hardt and B. Kim. "Sanity checks for saliency maps." arXiv preprint arXiv:1810.03292 (2018)

Example - similarity to edge detectors

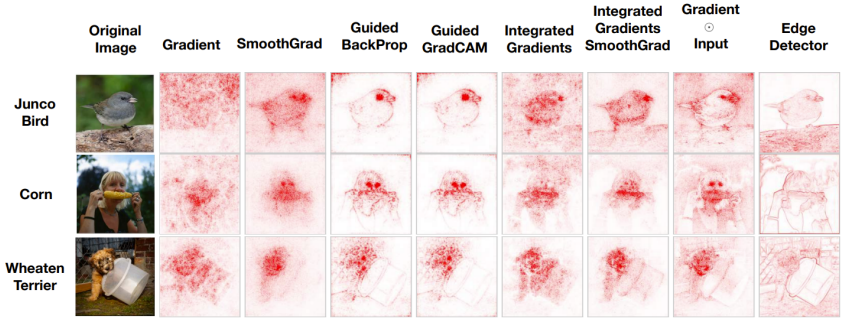


Figure: Adebayo et al, 2018

Example - randomization test

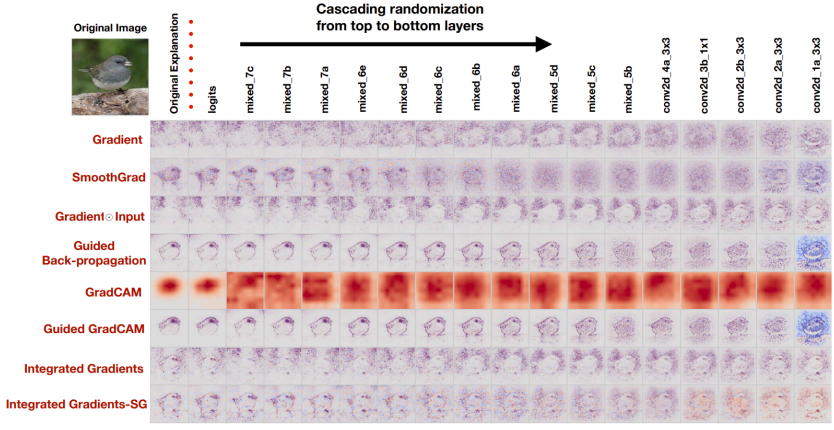


Figure: Adebayo et al, 2018

Contents

Introduction

Local, model-agnostic methods

Methods for CNN interpretation

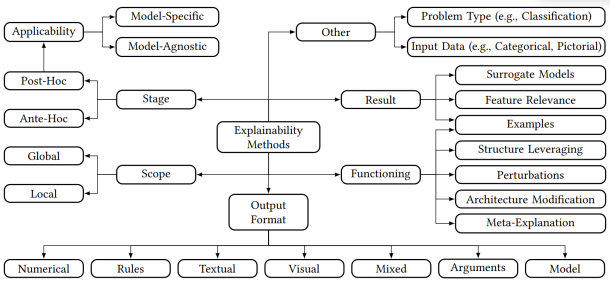
Global, model agnostic methods

DALE

DALE vs ALE

Goal

- Our aim is to produce interpretations that describe the model's behavior as a whole
- We focus on tabular data, and the result is usually a plot



Feature effect methods

- $x_s \rightarrow$ feature of interest, $\mathbf{x}_c \rightarrow$ other features
- How do we isolate the effect of x_s?

Partial Dependence Plots (PDP)

- Proposed by J. Friedman on 2001[12] and is the marginal **effect** of a feature to the model output:

$$\hat{f_s}(x_s) = E_{X_c}\left[\hat{f}(x_s, X_c)\right] = \int \hat{f}(x_s, X_c) d\mathbb{P}(X_c)$$

where x_s is the feature whose effect we wish to compute and X_c is a random variable corresponding to the rest of the model's features

- Computation:

$$\hat{f}_s(x_s) = \frac{1}{n}\sum_{i=1}^{n} \hat{f}(x_s, \mathbf{x}_c^{(i)})$$

[12] J. Friedman. "Greedy function approximation: A gradient boosting machine." Annals of statistics (2001): 1189-1232

Partial Dependence Plots (PDP)

Example 1 (continuous):

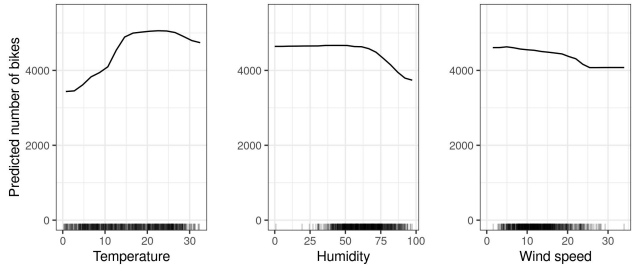


Figure: C. Molnar, IML book

[12] J. Friedman. "Greedy function approximation: A gradient boosting machine." Annals of statistics (2001): 1189-1232

Partial Dependence Plots (PDP)

Example 2 (categorical):

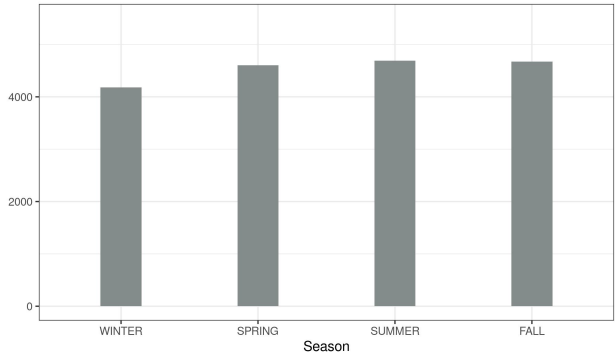


Figure: C. Molnar, IML book

[12] J. Friedman. "Greedy function approximation: A gradient boosting machine." Annals of statistics (2001): 1189-1232

Issues with PDPs

- Correlated features
  - Example: price $= f(\text{num\_rooms}, \text{area})$
  - To compute the effect of area for 40 m^2 we will use value 10 for the number of rooms (uncrealistic)
  - As a result, we have a wrong estimation of the feature effect
- Heterogeneous feature effects may be hidden by the use of average values

Issues with PDPs

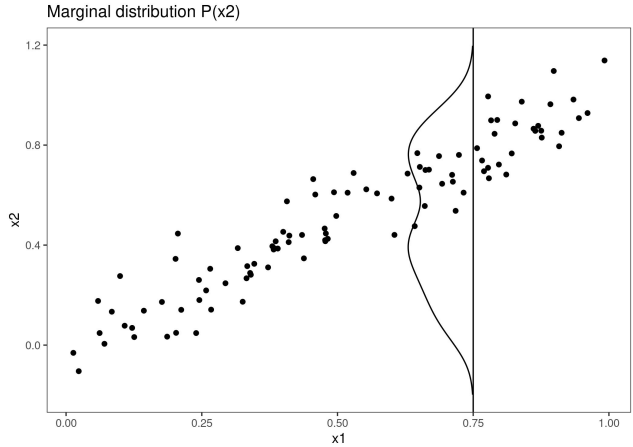


Figure: C. Molnar, IML book

MPlots

We use the value of x_s as a condition

- $\mathbf{x}_c|x_s$: $f(x_s) = \mathbb{E}_{\mathbf{x}_c|x_s}[f(x_s, \mathbf{x}_c)]$

MPlots

In the previous example

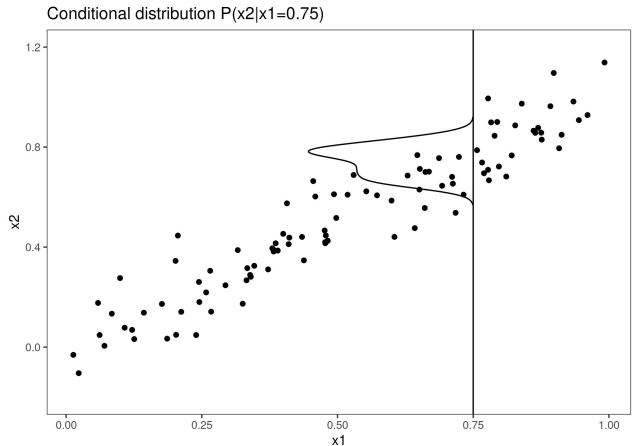


Figure: C. Molnar, IML book

Problems with M-Plots

- The calculated effects result from the combination of all (correlated) features
- Real effect: $x_{\texttt{age}} = 50 \rightarrow 10$, $x_{\texttt{years\_contraceptives}} = 20 \rightarrow 10$
- MPlot may estimate an effect of 17 for both

Accumulated Local Effects (ALE)[13]

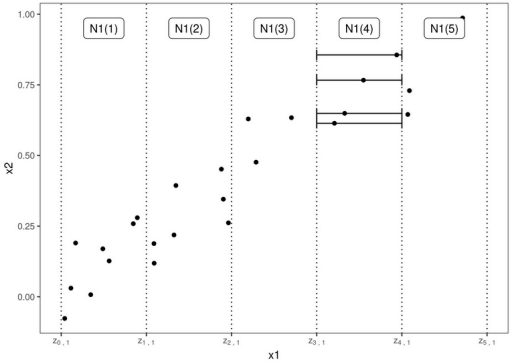
- Resolves problems that result from the feature correlation by computing differences over a (small) window
- $f(x_s) = \int_{x_{min}}^{x_s} \mathbb{E}_{\mathbf{x}_c|z}[\frac{\partial f}{\partial x_s}(z, \mathbf{x}_c)]\partial z$

[13] D. Apley and J. Zhu. "Visualizing the effects of predictor variables in black box supervised learning models." Journal of the Royal Statistical Society: Series B (Statistical Methodology) 82.4 (2020): 1059-1086.

ALE approximation

ALE definition: $f(x_s) = \int_{x_{s,min}}^{x_s} \mathbb{E}_{\mathbf{x}_c|z}[\frac{\partial f}{\partial x_s}(z, \mathbf{x}_c)]\partial z$ ALE approximation:

$$f(x_s) = \sum_{k=1}^{k_x} \underbrace{\frac{1}{|\mathcal{S}_k|} \sum_{i:\mathbf{x}^i \in \mathcal{S}_k} \underbrace{[f(z_k, \mathbf{x}_c^i) - f(z_{k-1}, \mathbf{x}_c^i)]}_{\text{point effect}}}_{\text{bin effect}}$$



ALE plots - examples

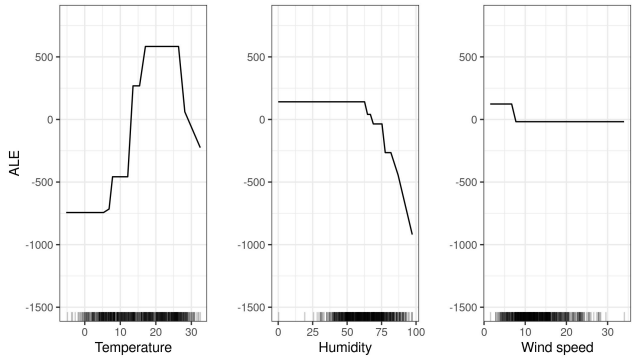


Figure: C. Molnar, IML book

ALE plots - examples

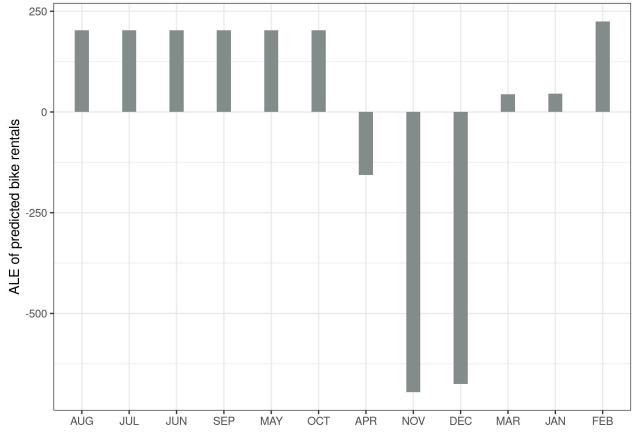


Figure: C. Molnar, IML book

Contents

Introduction

Local, model-agnostic methods

Methods for CNN interpretation

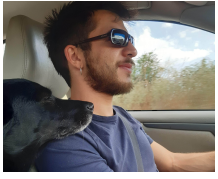
Global, model agnostic methods

DALE

DALE vs ALE

V. Gkolemis, T. Dalamagas, C. Diou, “DALE: Differential Accumulated Local Effects for efficient and accurate global explanations”, ACML, 2022

Most of the work done by this guy:



ALE approximation - weaknesses

$$f(x_s) = \sum_{k}^{k_x} \underbrace{\frac{1}{|\mathcal{S}_k|} \sum_{i:\boldsymbol{x}^i \in \mathcal{S}_k} \underbrace{[f(z_k, \mathbf{x}_c^i) - f(z_{k-1}, \mathbf{x}_c^i)]}_{\text{point effect}}}_{\text{bin effect}}$$

- Point Effect $\Rightarrow$ evaluation at bin limits
  - 2 evaluations of f per point $\rightarrow$ slow
  - change bin limits, pay again $2 * N$ evaluations of f $\rightarrow$ restrictive
  - broad bins may create out of distribution (OOD) samples $\rightarrow$ not-robust in wide bins

Our proposal: Differential ALE

$$f(x_s) = \Delta x \sum_{k}^{k_x} \underbrace{\frac{1}{|\mathcal{S}_k|} \sum_{i:\boldsymbol{x}^i \in \mathcal{S}_k} \underbrace{[\frac{\partial f(x_s^i, \mathbf{x}_c^i)}{\partial x_s}]}_{\text{point effect}}}_{\text{bin effect}}$$

- Point Effect $\Rightarrow$ evaluation on instances
  - Fast $\rightarrow$ use of auto-differentiation, all derivatives in a single pass
  - Versatile $\rightarrow$ point effects computed once, change bins without cost
  - Secure $\rightarrow$ does not create artificial instances

For differentiable models, DALE resolves ALE weaknesses

Contents

Introduction

Local, model-agnostic methods

Methods for CNN interpretation

Global, model agnostic methods

DALE

DALE vs ALE
- Dale is faster and versatile
- DALE is more Accurate

DALE is faster and versatile - theory

$$f(x_s) = \Delta x \sum_{k}^{k_x} \underbrace{\frac{1}{|\mathcal{S}_k|} \sum_{i:\boldsymbol{x}^i \in \mathcal{S}_k} \underbrace{[\frac{\partial f(x_s^i, \mathbf{x}_c^i)}{\partial x_s}]}_{\text{point effect}}}_{\text{bin effect}}$$

- Faster
  - gradients wrt all features $\nabla_{\mathbf{x}} f(\mathbf{x}^i)$ in a single pass
  - auto-differentiation must be available (deep learning)
- Versatile
  - Change bin limits, with near zero computational cost

DALE is faster and allows redefining bin-limits

DALE is faster and versatile - Experiments

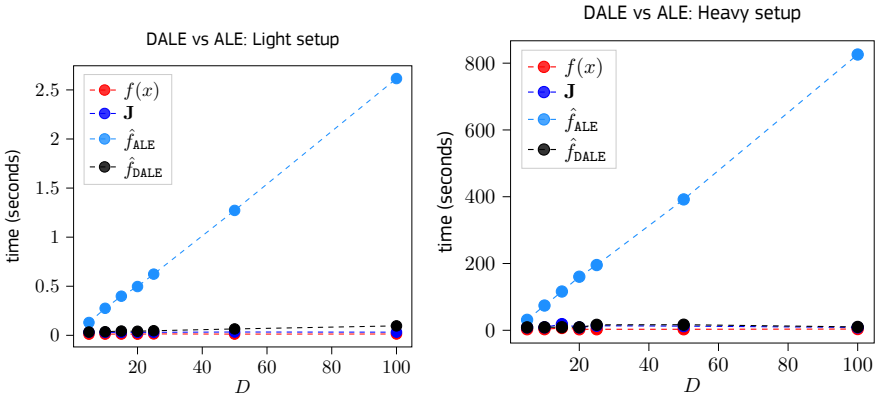


Figure: Light setup; small dataset ($N = 10^2$ instances), light f. Heavy setup; big dataset ($N = 10^5$ instances), heavy f

DALE considerably accelerates the estimation

DALE uses on-distribution samples - Theory

$$f(x_s) = \sum_{k}^{k_x} \underbrace{\frac{1}{|\mathcal{S}_k|} \sum_{i:\boldsymbol{x}^i \in \mathcal{S}_k} \underbrace{[\frac{\partial f(x_s^i, \mathbf{x}_c^i)}{\partial x_s}]}_{\text{point effect}}}_{\text{bin effect}}$$

- point effect independent of bin limits
  - $\frac{\partial f(x_s^i, \mathbf{x}_c^i)}{\partial x_s}$ computed on real instances $\mathbf{x}^i = (x_s^i, \mathbf{x}_c^i)$
- bin limits affect only the resolution of the plot
  - wide bins $\rightarrow$ low resolution plot, bin estimation from more points
  - narrow bins $\rightarrow$ high resolution plot, bin estimation from less points

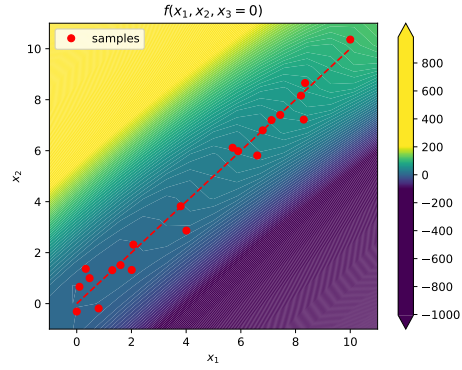
DALE enables wide bins without creating out of distribution instances

DALE uses on-distribution samples - Experiments

$$f(x_1, x_2, x_3) = x_1 x_2 + x_1 x_3 \pm g(x)$$

$$x_1 \in [0, 10], x_2 \sim x_1 + \epsilon, x_3 \sim \mathcal{N}(0, \sigma^2)$$

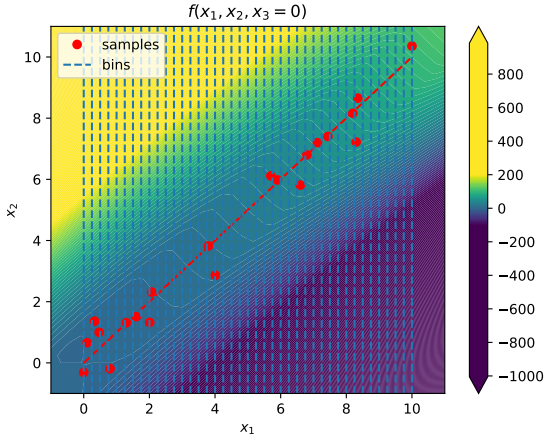
$$f_{\texttt{ALE}}(x_1) = \frac{x_1^2}{2}$$



- point effects affected by $(x_1 x_3)$ (σ is large)
- bin estimation is noisy (samples are few)

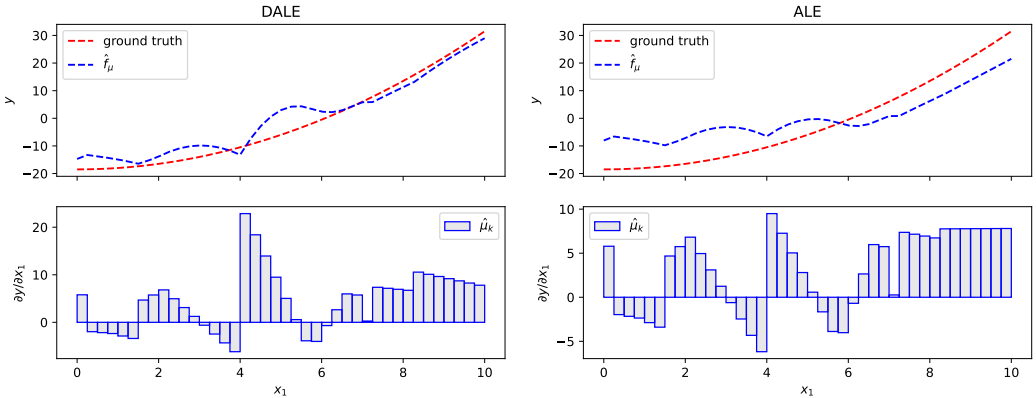
Intuition: we need wider bins (more samples per bin)

DALE vs ALE - 40 Bins



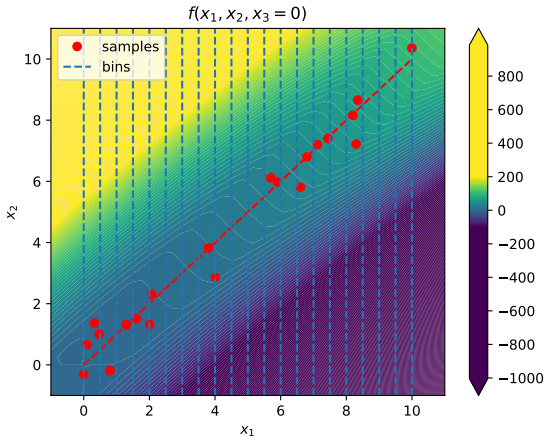
- DALE: on-distribution, noisy bin effect → poor estimation
- ALE: on-distribution, noisy bin effect → poor estimation

DALE vs ALE - 40 Bins



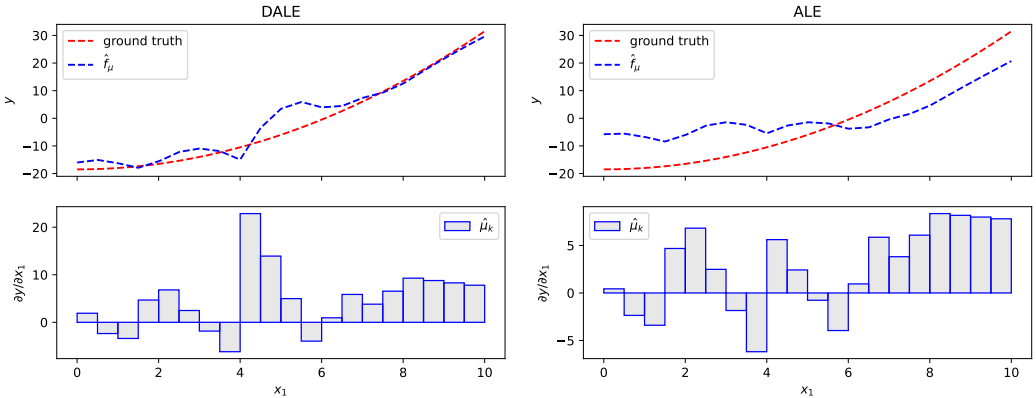
- DALE: on-distribution, noisy bin effect $\rightarrow$ poor estimation
- ALE: on-distribution, noisy bin effect $\rightarrow$ poor estimation

DALE vs ALE - 20 Bins



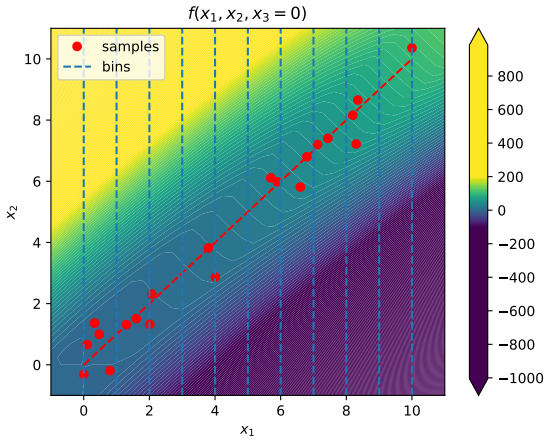
- DALE: on-distribution, noisy bin effect → poor estimation
- ALE: on-distribution, noisy bin effect → poor estimation

DALE vs ALE - 20 Bins



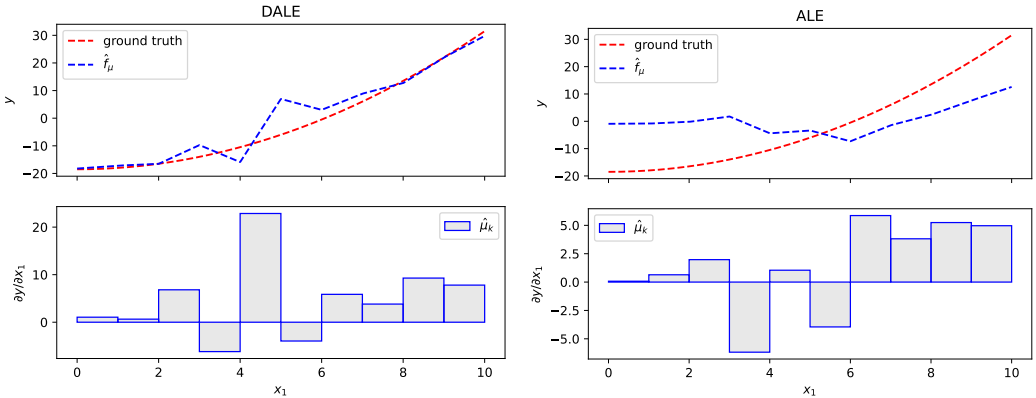
- DALE: on-distribution, noisy bin effect $\rightarrow$ poor estimation
- ALE: on-distribution, noisy bin effect $\rightarrow$ poor estimation

DALE vs ALE - 10 Bins



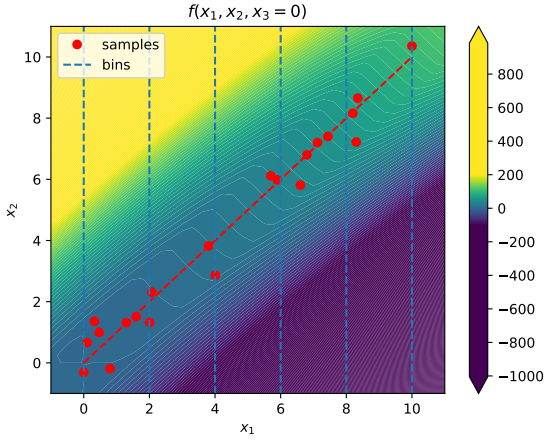
- DALE: on-distribution, noisy bin effect → poor estimation
- ALE: starts being OOD, noisy bin effect → poor estimation

DALE vs ALE – 10 Bins



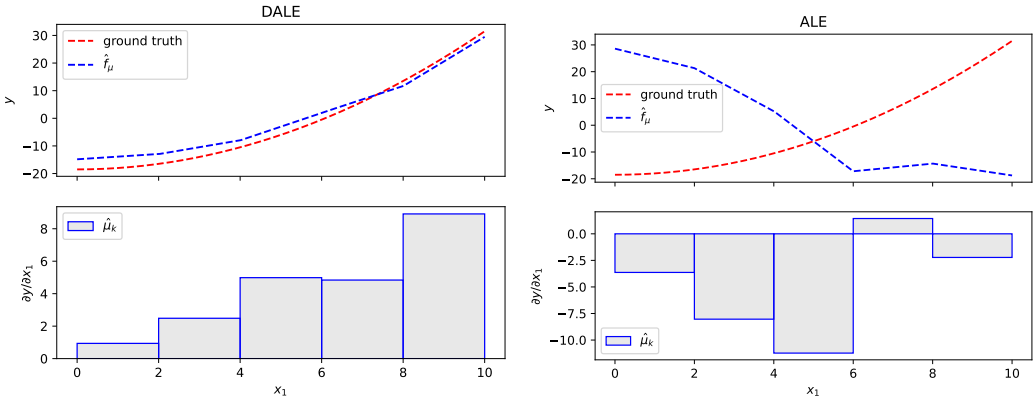
- DALE: on-distribution, noisy bin effect → poor estimation
- ALE: starts being OOD, noisy bin effect → poor estimation

DALE vs ALE - 5 Bins



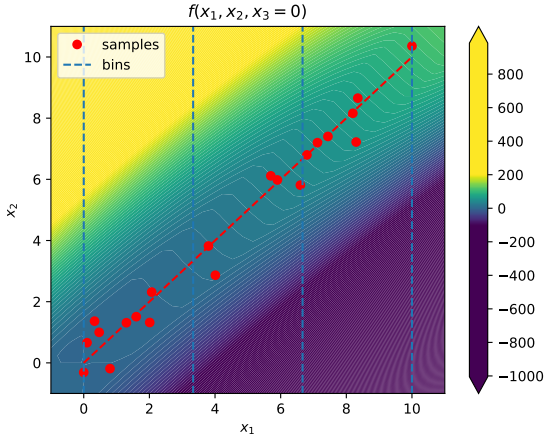
- DALE: on-distribution, robust bin effect → good estimation
- ALE: completely OOD, robust bin effect → poor estimation

DALE vs ALE - 5 Bins



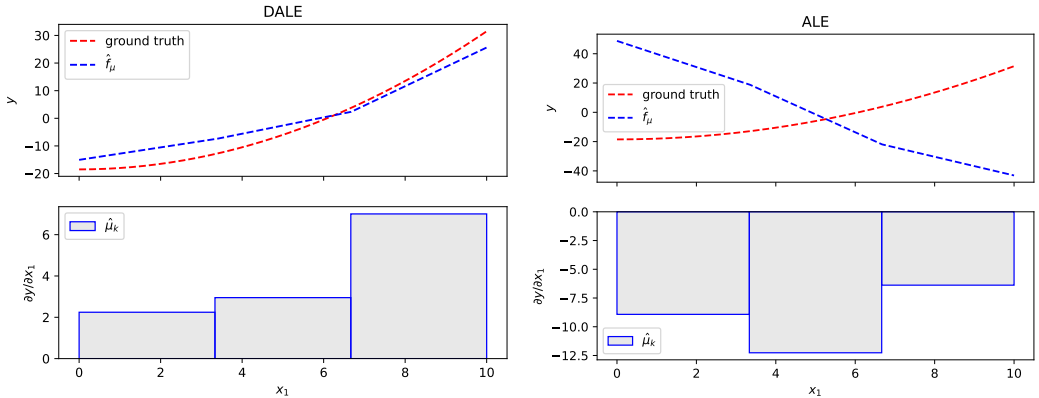
- DALE: on-distribution, robust bin effect $\rightarrow$ good estimation
- ALE: completely OOD, robust bin effect $\rightarrow$ poor estimation

DALE vs ALE - 3 Bins



- DALE: on-distribution, robust bin effect → good estimation
- ALE: completely OOD, robust bin effect → poor estimation

DALE vs ALE - 3 Bins



- DALE: on-distribution, robust bin effect $\rightarrow$ good estimation
- ALE: completely OOD, robust bin effect $\rightarrow$ poor estimation

Real Dataset Experiments - Efficiency

- Bike-sharing dataset
- $y \rightarrow$ daily bike rentals
- $\mathbf{x}$: 10 features, most of them characteristics of the weather

Efficiency on Bike-Sharing Dataset (Execution Times in seconds)

	Number of Features										
	1	2	3	4	5	6	7	8	9	10	11
DALE	1.17	**1.19**	**1.22**	**1.24**	**1.27**	**1.30**	**1.36**	**1.32**	**1.33**	**1.37**	**1.39**
ALE	**0.85**	1.78	2.69	3.66	4.64	5.64	6.85	7.73	8.86	9.9	10.9

DALE requires almost same time for all features

Real Dataset Experiments - Accuracy

- Difficult to compare in real world datasets
- We do not know the ground-truth effect
- In most features, DALE and ALE agree.
- Only $X_{\texttt{hour}}$ is an interesting feature

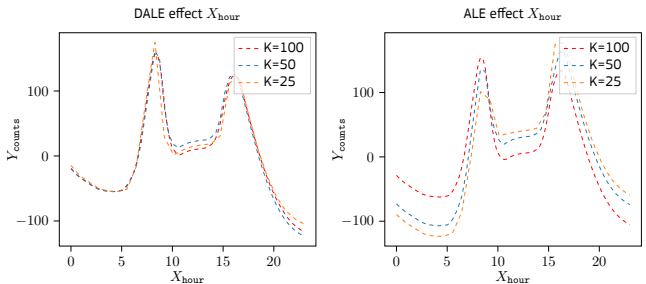


Figure: (Left) DALE (Left) and ALE (Right) plots for $K = \{25, 50, 100\}$

What's next?

- Could we automatically decide the optimal bin sizes?
  - Sometimes narrow bins are ok
  - Sometimes wide bins are needed
- What about variable size bins?
- Model the uncertainty of the estimation?

DALE can be a driver for future work

Thank you!