BayLIME: Bayesian Local Interpretable Model-Agnostic Explanations

(Zhao et al. 2021)

Explainable AI

- research field that aims at improving the trust and transparency of AI
- aim: provide *good* explanations
  - **interpretability** = qualitative understanding between the input variables and the response
  - **fidelity** = how truthfully the explanation represents the unknown behaviour underlying AI decision

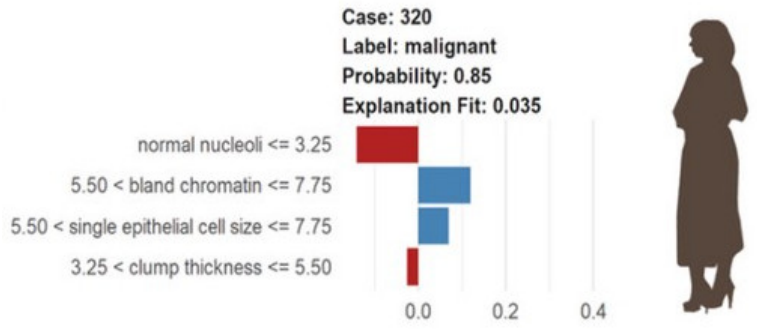
(Ribeiro et al. 2016a,b)

explainability vs interpretability
molnar and articles for a bit more context

LIME: Local Interpretable Model-agnostic Explanations

- implementation of local surrogate models (Ribeiro et al. 2016a)
  - **surrogate** = **interpretable** models trained to **approximate** the predictions of the underlying black box model
  - **local** = focuses on training surrogate models to explain **individual** predictions
- **model-agnostic** = can be used for any ML model
- works for all tabular data, text and images
- most popular XAI method

(Stiglic et al. 2020)



LIME: how does it proceed ?

Aim: identify an interpretable model over the interpretable representation that is locally faithful (Ribeiro et al. 2016b)

The procedure in brief:

1. select your **instance of interest**
2. **perturb the dataset**, get new black box predictions
3. **weight** the new samples according to their proximity to the instance of interest
4. train a **weighted, interpretable model** on the dataset with the variations
5. **explain the prediction** by interpreting the local model

(Molnar 2021, §5.8)

LIME: how does it proceed ?

Mathematically speaking...

Obtain the explanation $\xi(x)$ by solving:

$$\xi(x) = \underset{g \in G}{argmin}\ L(f, g, \Pi x) + \Omega(g)$$

G = class of potentially interpretable models; g = explanation model; f =model being explained; $\Pi_x(z)$ = proximity measure; $L(f, g, \Pi x)$ = fidelity function; Ω = complexity measure

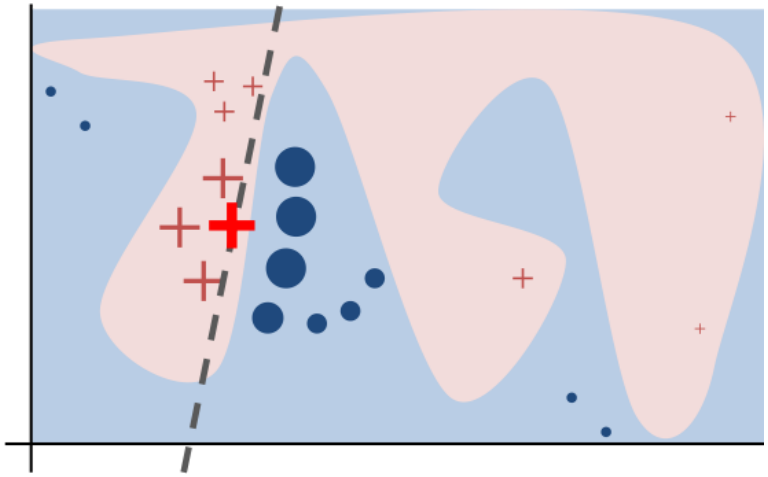
Estimate L by generating perturbed samples around x, making predictions with the black box model f and weighting them according to Π_x (Ribeiro et al. 2016b)

$\rightarrow$ minimize $L(f, g, \Pi x)$ while having $\Omega(g)$ be low enough (*fidelity-interpretability trade-off*)

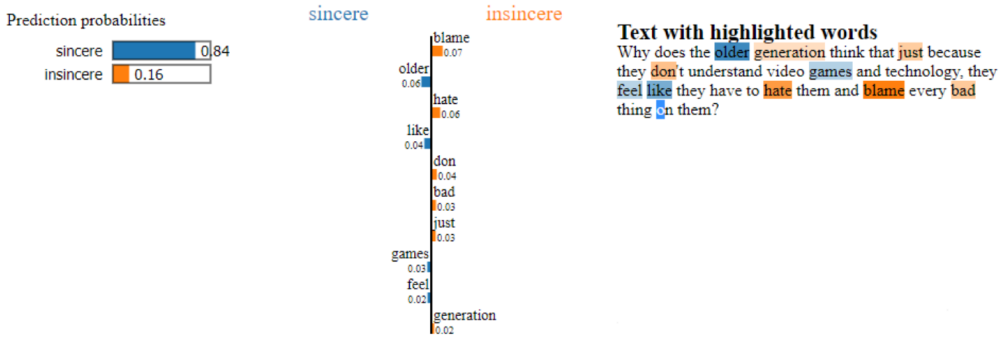
LIME: how does it proceed ?

... and visually speaking

procedure:



output:



LIME weaknesses

- **inconsistency**: different explanations can be generated for the same prediction
  - caused by the randomness in generating perturbed samples that are used for the training of local surrogate models
  - smaller sample size = greater uncertainty
  - limits its usefulness in critical applications (e.g. medical domain)
- **unrobustness** to kernel settings: challenge of defining the "neighbourhood" on which a local surrogate is trained
  - no effective way to find the best kernel settings
  - best strategy for now: "trial-error" (biases !)

BayeLIME

- a “Bayesian principled weighted sum” of the prior knowledge and the estimates based on new samples

- the weights are proportional to
  1. the **“pseudo-count” of prior sample size** based on which we form our prior estimates μ_0
  2. the **“accurate-actual-count” of observation sample size**, i.e. the actual observation of the *n* new samples scaled by the precision *a*

BayeLIME: prior embedding

1. form the **prior estimate** of μ_0 based on λ data points
2. collect *n* new samples and consider their precision (α) and weights (w_c) for forming a **MLE estimate** β_{MLE}
3. **combine** μ_0 **and** β_{MLE} according to their proportions of the effective samples size (λ and $\alpha w_c n$, respectively)
4. calculate the **posterior precision** captured by all effective samples (i.e. $\lambda + \alpha w_c n$)

Upsides:

$\rightarrow$ **improves consistency** by averaging out randomness
$\rightarrow$ **improves robustness** by averaging out effects from kernels
$\rightarrow$ **improves explanation fidelity** by combining diverse information

BayeLIME: choice of the priors

- **non-informative** priors
  - μ_0: zero mean vector
  - λ and α: fitted with Bayesian model selection
- **partial informative** priors
  - μ_0 and λ: known distribution
  - α: fitted with Bayesian model selection
- **full informative** priors (ideal scenario)
  - μ_0, λ and α: known distribution

Methods: BayeLIME validation

RQ1. **consistency** improvement (vs. LIME)

RQ2. **robustness** to kernel settings improvement (vs. LIME)

RQ3. explanation **fidelity** improvement (vs. XAI methods)

Datasets:

- Boston house-price dataset
- breast cancer Wisconsin dataset
- a set of CNNs pretrained on the ImageNet and GTSRB

Methods: (in)consistency

- **Kendall's W**
  - measure the agreement among raters (i.e. repeated explanations in our case)
  - ranges from 0 (no agreement) to 1 (complete agreement)
  - procedure:
    - select a set of BayLIME explainers with **different options and prior parameters**
    - for each, iterate the explanation of the given instance *k* times, and quantify the inconsistency

Methods: (in)consistency

Kendall's W considers the discrete ranks of features: cannot discriminate explanations with the same ranking of features but different importance vectors

- metric based on the **index of dispersion** (IoD) of each feature in repeated runs
  - weights the IoD of the rank of each feature by its importance

Methods: robustness to kernel settings

- define a kernel width settings interval $[l_{lo}, l_{up}]$
- randomly sample from that interval 5000 **pairs of kernel width parameters**
- for each pair, calculate the "distance" of the 2 explanations
- obtain a sample set of ratios between the 2 distances of explanations and the kernel width pair
- its **median value** provides insights on the general robustness

Methods: explanation fidelity

actual causality as an indicator for the explanation fidelity

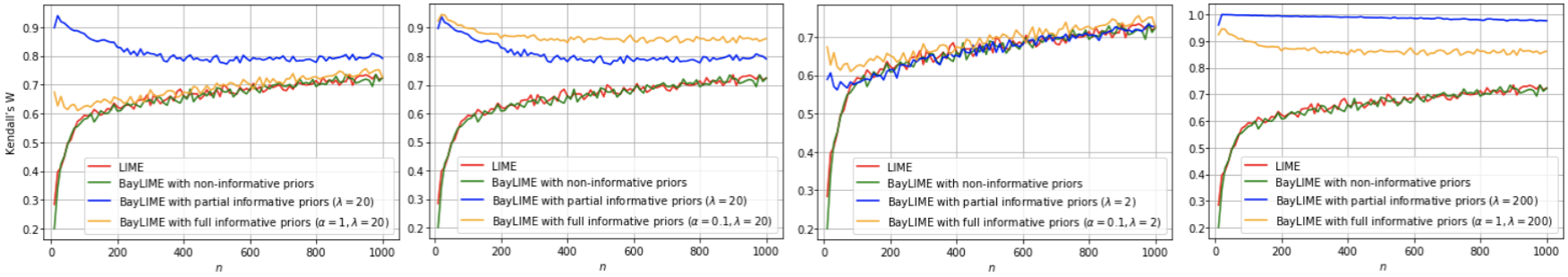
- 2 causal metrics
  - **deletion**: decrease in the probability of the predicted label when starting with a complete instance and then gradually removing top-important features
    - good explanation = sharp drop (low AUC as a function of the fraction of removed features)
  - **insertion**: increase in the probability as more and more important features are introduced
    - good explanation = higher AUC
- **neural backdoors**

Methods: how to obtain prior knowledge ?

- explanations of a set of **similar instances** (RQ1+RQ2)
  - the average importance of each feature in that set collectively forms the prior mean vector
- **XAI techniques** (RQ3a)
  - explanations obtained from other XAI explainers
  - here: GradCAM results as priors
- **Validation and Verification (V&V) methods** (RQ3b)
  - direct analysis of the behaviour of the underlying AI model
  - e.g. detection tools may provide prior knowledge on possible backdoor triggers
  - here: NeuralCleanse results as priors

Results: consistency improvement

- non-informative BayLIME indistinguishable from LIME
  - monotonic trends as *n* increases for both LIME and non-informative BayLIME
- by contrast, BayLIME with partial/full informative priors "averages out" the sampling noise



Results: consistency improvement

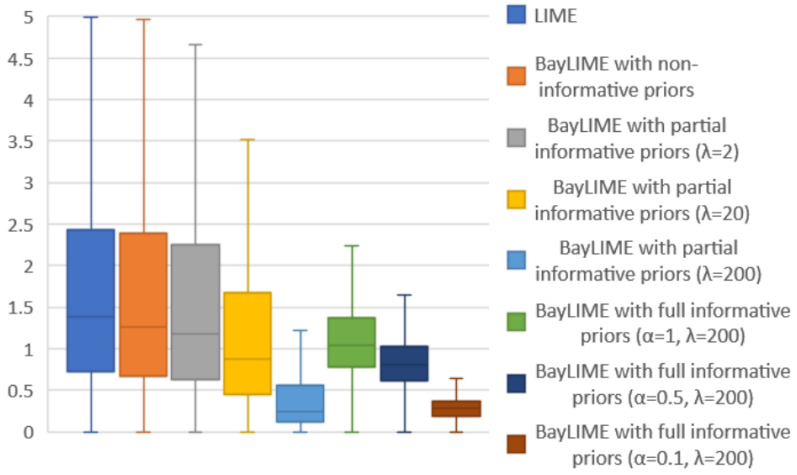
How different priors affect consistency ?

- use the auxiliary of the factor λ/α ("regularization coefficient")
- when $\alpha \simeq 0$: huge penalty on the data
- when $\lambda \simeq 0$: no penalty on the data
- $\lambda/\alpha = 20$: identical curves for BayLIME with full informative priors
- when λ/α increases to 200: stronger ability of averaging out sampling noise (higher Kendall's W)

when $n \to +\infty$, all converge to the measurement based on MLE

Results: robustness to kernel settings improvement

- similar robustness for LIME and BayLIME with non-informative priors
- either partial or full prior knowledge improves robustness



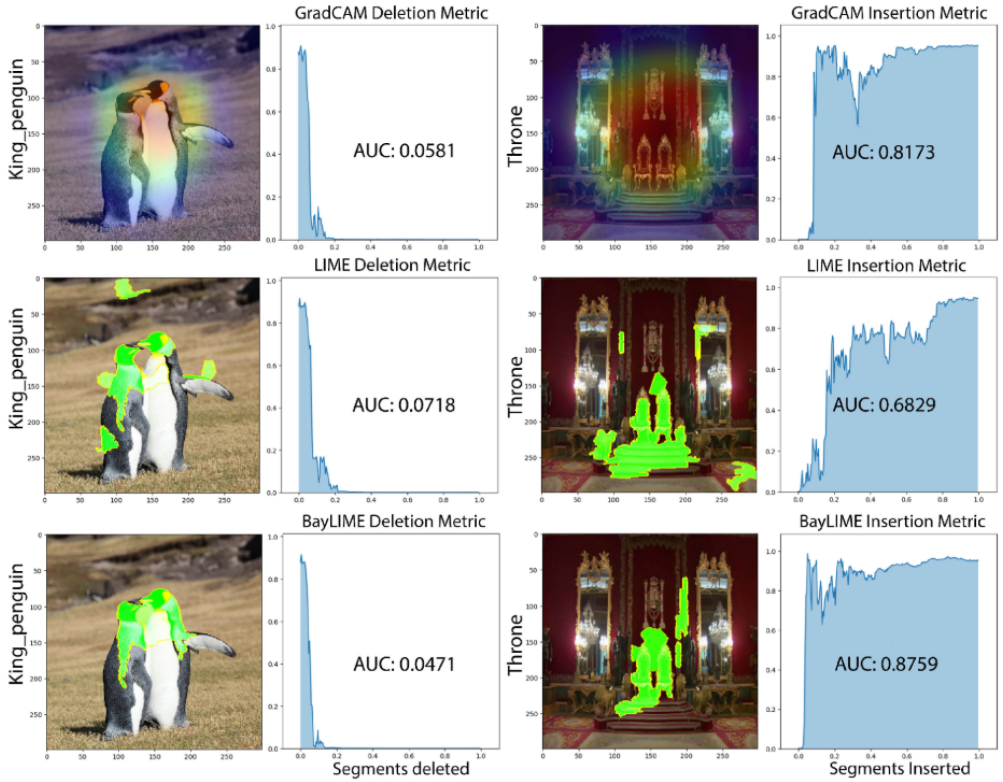
Results: robustness to kernel settings improvement

How varying the λ and α affects the robustness ?

- contribution from the priors (independent from kernel setting) $\leftrightarrow$ contribution from the new data (sensitive to kernel settings), to the posteriors (cf. λ/α)

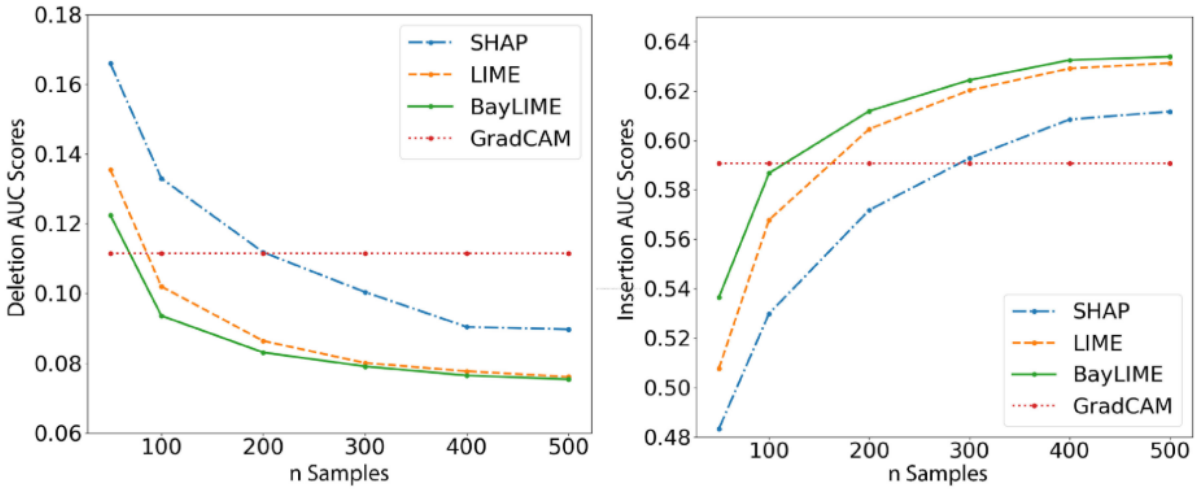
Results: explanation fidelity - XAI methods

- better performance than GradCAM and LIME



Results: explanation fidelity - XAI methods

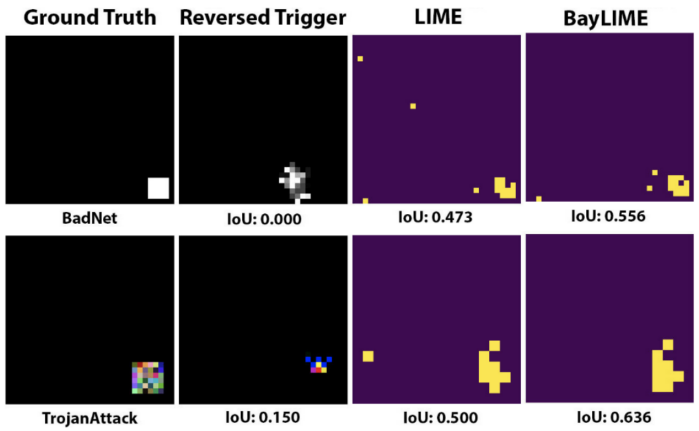
- by varying *n* (average scores):
  - better than SHAP and LIME, converging when *n* increases
  - GradCAM better only when *n* is extremely small



⇒ **better fidelity in the middle and most practical range of *n***

Results: explanation fidelity - V&V methods

- NeuralCleanse yields reversed triggers as the approximation of backdoor, which are far from perfect
- even directly apply LIME on an attacked image may provide a better IoU than NeuralCleanse.



Model	NeuralCleanse	LIME	BayLIME
BadNet	0.000	0.385	**0.406**
TrojanAttack	0.150	0.599	**0.637**

$\Rightarrow$ **better fidelity after considering both the reversed triggers and a surrogate model**

Conclusion

BayLIME:

- is the first to exploit prior knowledge for better consistency, robustness to kernel settings and explanation fidelity
- improves over LIME
  - the prior knowledge is independent from the causes of inconsistency and unrobustness (thus benefits both properties)
  - improve fidelity by including additional useful information
- performs better than V&V and other XAI methods

$\Rightarrow$ a way to obtain better explanations of AI models

$\Rightarrow$ a (Bayesian) way to inject knowledge in AI model interpretation (but defining good priors remains challenging !)

Appendix: inconsistency metric

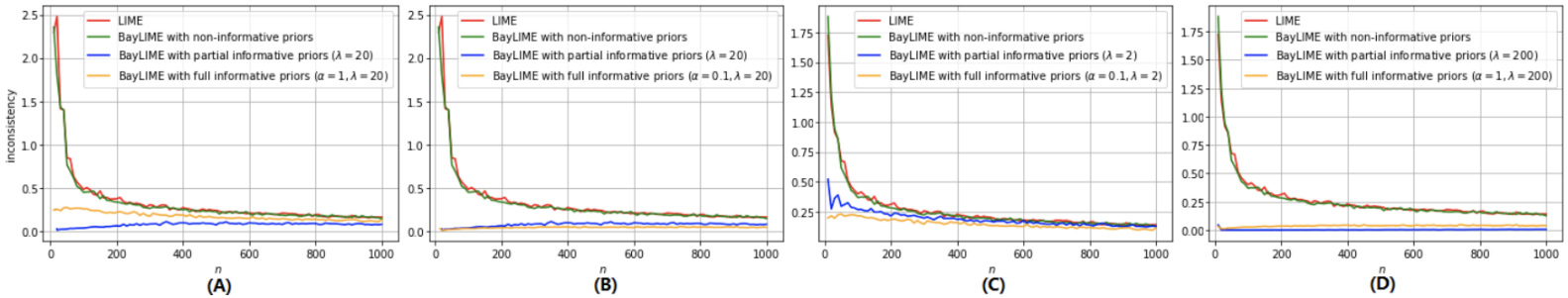


Figure 9: To complement Kendall's W, the inconsistency metric Eq. (15) in $k = 200$ repeated explanations by LIME and BayLIME on tabular data. Each set shows an illustrative combination of α and λ.

Appendix: inconsist. metric vs. Kendall's W

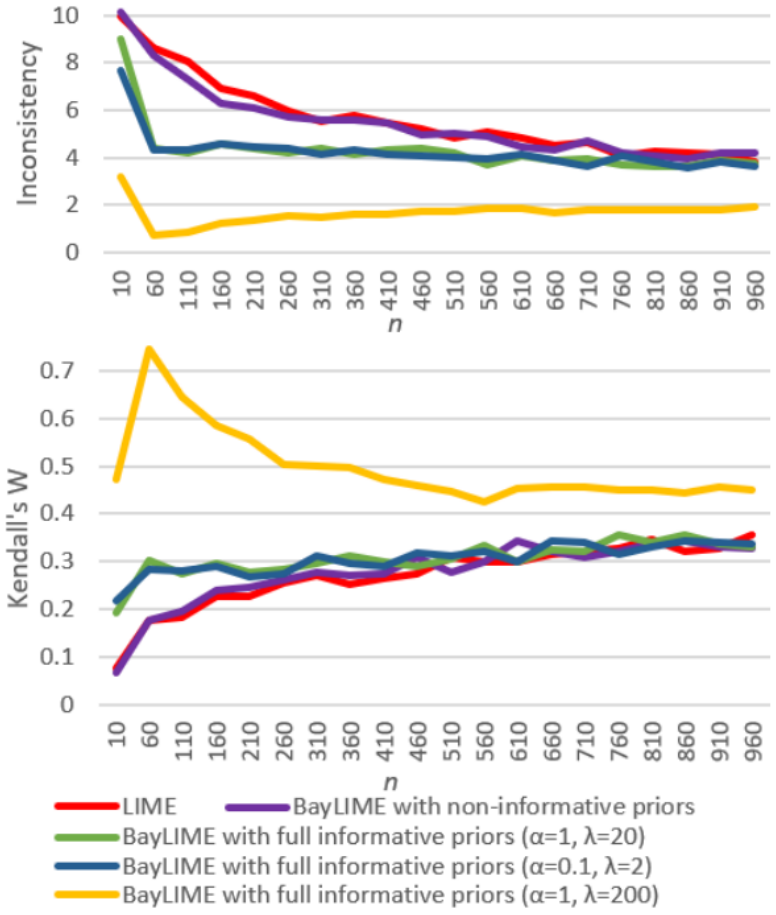


Figure 10: For different explainers, the inconsistency of (15) and Kendall's W in repeated explanations of images labelled by InceptionV3 as functions of the perturbed sample size n.

Appendix: IoU vs. AMD

Table 4: Statistics on IoU (higher is better) and AMD (smaller is better) based on 500 backdoor-attacked images. The priors (derived from reversed triggers) are shown in Fig. 8.

Model	IoU			AMD		
	Prior	LIME	BayLIME	Prior	LIME	BayLIME
BadNet	0.000	0.385	0.406	0.121	0.212	0.178
TrojanAttack	0.150	0.599	0.637	0.011	0.072	0.049

Appendix:

Appendix:

Appendix:

Appendix:

Appendix:

Appendix: