Local Interpretable Model-agnostic Explanations

This is an *Explainability Fact Sheet* for Local Interpretable Model-agnostic Explanations (LIME). It is distributed as a supplementary material of the "Explainability Fact Sheets: A Framework for Systematic Assessment of Explainable Approaches" paper (Kacper Sokol and Peter Flach, 2020) published in Conference on Fairness, Accountability, and Transparency (FAT* 2020).

Approach Characteristic

Description

Local Interpretable Model-agnostic Explanations (LIME) is a surrogate explainability method that aims to approximate a local decision boundary with a sparse linear model to interpret individual predictions. It was introduced by this paper and the implementation provided by its authors is capable of explaining *tabular*, *image* and *text* data.

Implementations

Python
LIME
bLIMEy

Citation

```
@inproceedings{lime,
  author    = {Marco Tulio Ribeiro and Sameer Singh and Carlos Guestrin},
  title     = {"Why Should {I} Trust You?": Explaining the Predictions of
               Any Classifier},
  booktitle = {Proceedings of the 22nd {ACM} {SIGKDD} International
               Conference on Knowledge Discovery and Data Mining,
               San Francisco, CA, USA, August 13-17, 2016},
  pages     = {1135--1144},
  year      = {2016},
}
```

Variants

bLIMEy

build LIME yourself -- a modular framework for building custom local surrogate explainers.

```
@article{blimey,
  title={b{LIME}y: {S}urrogate {P}rediction {E}xplanations {B}eyond {LIME}},
  author={Sokol, Kacper and Hepburn, Alexander and Santos-Rodriguez, Raul
          and Flach, Peter},
  journal={2019 Workshop on Human-Centric Machine Learning (HCML 2019) at the
           33rd Conference on Neural Information Processing Systems
           (NeurIPS 2019), Vancouver, Canada},
  note={arXiv preprint arXiv:1910.13016},
  url={https://arxiv.org/abs/1910.13016},
  year={2019}
}
```

Related Approaches

N/A

Functional Requirements

F1: Problem Supervision Level

LIME works with:

- **supervised** predictive algorithms, and
- **semi-supervised** predictive algorithms.

F2: Problem Type

LIME is designed for:

- **probabilistic classifiers** and supports: *binary* and *multi-class* classification tasks, and
- **regression** problems.

F3: Explanation Target

LIME can only explain **predictions** of a Machine Learning model.

F4: Explanation Breadth/Scope

Explanations produced by LIME are **local**.

F5: Computational Complexity

For every explained data point the LIME algorithm needs to perform the following *computationally* and/or *time* expensive steps with the cost of each one depending on the actual algorithmic component used:

- **Generating an interpretable data representation** may be necessary for some applications. *Tabular data* may be binned to form human-comprehensible features such as "15 < age < 18". *Images* need to be pre-processed to identify super-pixels. *Text* has to be (possibly pre-processed and) transformed into a bag of words representation.
- In order to train a local model to approximate the local behaviour of a global model, we need to **sample data** around the data point being explained. For *tabular data* the sampling algorithm needs to sample data points with the same number of features as in the original data set or if an interpretable representation is used, the same number of features but each one being multinomial with different values indicating different bins defined on that feature. For *images* and *text* the sampling is based on a binary vector of length equal to the number of unique words for text and the number of super-pixels for images.
- Each sampled data point has to be **predicted** with the global model.
- To enforce the locality of an explanation, sampled data are weighted based on their **distance** to the data point being explained, which has to be computed for every generated data point. While for *text* and *images* this is a distance computed on a binary vector; for *tabular data* without interpretable data representation this would most likely be a more computationally-heavy distance calculation procedure, e.g., Euclidean distance.
- A **feature selection** algorithm may be run on *tabular data* to introduce sparsity into the explanation.
- For every data point being explained, a local model has to be **trained** for each explained class as the local model's task is to predict one class vs. the rest.

F6: Applicable Model Class

The LIME algorithm is **model agnostic**, therefore it works with any predictive model.

The official LIME implementation uses linear **regression** as a local model, therefore for classification tasks the black-box being explained has to be a **probabilistic** model (i.e., output probabilities).

F7: Relation to the Predictive System

This approach is **post-hoc**, therefore it can be retrofitted to any predictive system.

F8: Compatible Feature Types

Tabular Data

Works with both **categorical** and **numerical** features. If an interpretable data representation is used (default behaviour in the implementation), all of the features become categorical (bins) for the purpose of explanation legibility.

Images

Images are always transformed into an interpretable data representation, namely super-pixels represented as a binary "on/off" vector.

Text

Text data are always transformed into an interpretable data representation, namely a bag of words represented as a binary "on/off" vector.

F9: Caveats and Assumptions

By default the LIME implementation discretises tabular data before the sampling procedure, which leads to the sampled data resembling more of a global rather than a local neighbourhood. This is counterbalanced with the data point weighting step based on the proximity of each data point to the one being explained. Moreover, discretising first means that in order to get global model predictions of the sampled data, we need to "un-discretise" them, which in the LIME implementation is performed by uniformly sampling data from each bin, therefore introducing another source of randomness.

For more details please see "bLIMEy: Surrogate Prediction Explanations Beyond LIME" by Kacper Sokol, et al.

Operational Requirements

O1: Explanation Family

Associations between antecedent and consequent.

Tabular Data

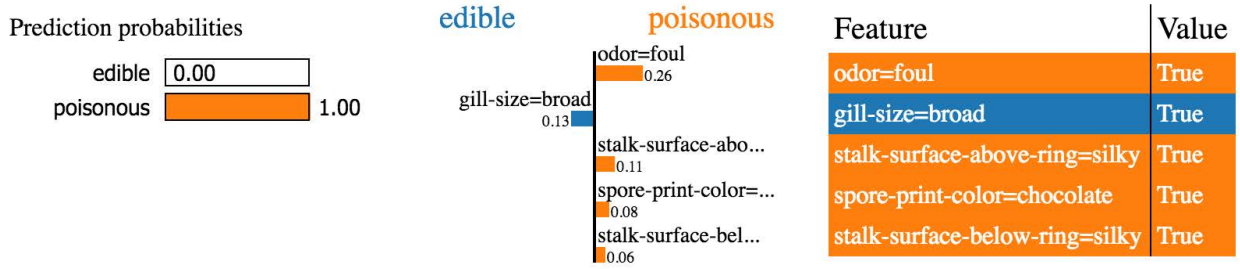
The explanations produced by tabular LIME are **associations between antecedent and consequent** -- each feature, or a particular bin on that feature if data are transformed into an interpretable representation, is assigned a positive or a negative influence on the local prediction of a selected class.

Images and Text

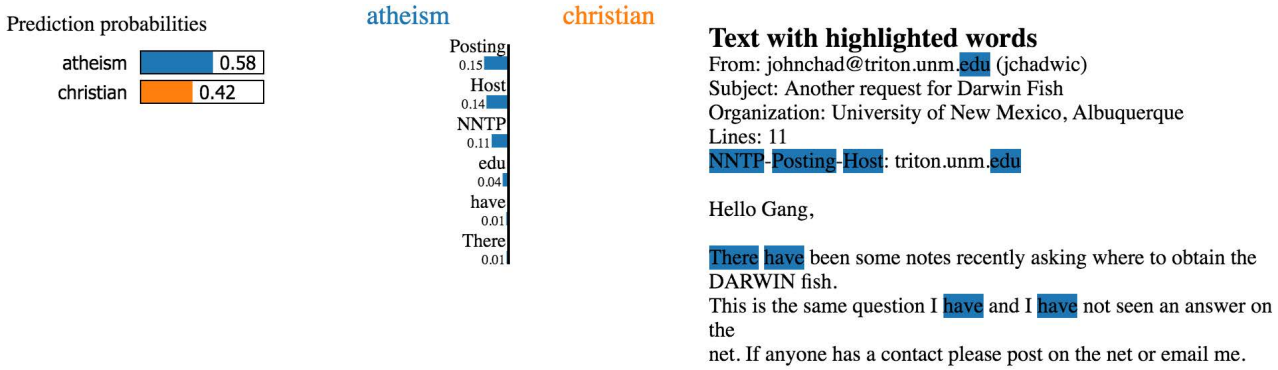
The explanations produced by image and text LIME are **associations between antecedent and consequent** -- each word or super-pixel is assigned a positive or a negative influence on the local prediction of a selected class.

O2: Explanatory Medium

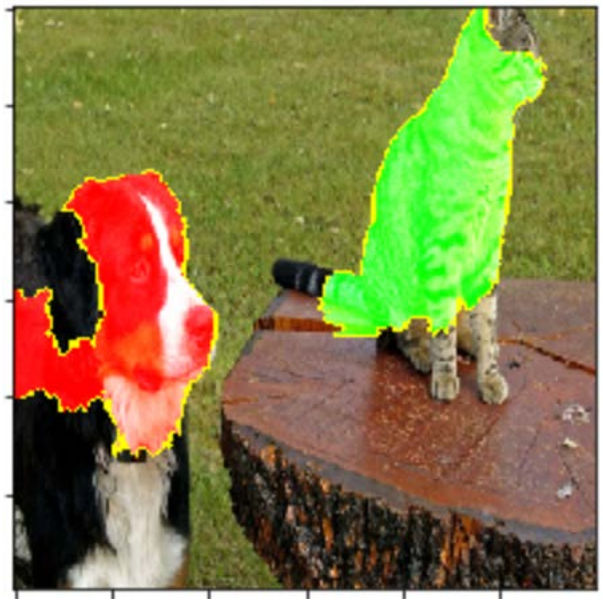
The explanations are delivered as **visualisations**. For *tabular data* this is feature importance, e.g.:



For *text* this is word importance, e.g.:



And, for *images* this is super-pixel importance, e.g.:



(The figures were taken from the LIME documentation.)

O3: System Interaction

The explanations are **static** visualisations.

O4: Explanation Domain

The explanations are expressed as *local* feature importance, i.e., parameters extracted from a locally fitted linear model. For tabular data the explanation can either be in the original domain (training data features) or in an interpretable domain (feature binning). For images these importance factors are expressed for super-pixels and for text these are unique words for a given sentence.

O5: Data and Model Transparency

For *image* and *text* data there is *no need* for transparency as an interpretable data representation is used. For *tabular data*, regardless of whether an interpretable data representation is used or not, the features *need to* be human-interpretable.

Since this is a *model agnostic* interpretability approach, there is *no need* for the global model to be inherently transparent in any way.

O6: Explanation Audience

For *tabular data* the audience should be familiar with the general domain of the problem to be able to interpret the meaning of the features. For *images* and *text* any audience is suitable.

The audience is not required to be familiar with Machine Learning concepts.

O7: Function of the Explanation

The main function of LIME is to increase transparency of a prediction. However, with enough background knowledge the algorithm can also be used as a diagnostic tool when debugging a black-box predictive system.

O8: Causality vs. Actionability

LIME explanations are **not** of a causal nature. The explanations also lack a direct actionable interpretation.

O9: Trust vs. Performance

There is **no** performance penalty since LIME is post-hoc and model agnostic. Trust in LIME explanations may suffer given instability and randomness of components influencing the explanation generation process (see **S3** for more details).

O10: Provenance

LIME explanations are based on *interactions* with the global model and *sampled data*, which affect building a local interpretable linear model. Then, the *parameters* of the local linear model are used as an explanation.

Usability Requirements

U1: Soundness

There are two types of local soundness and one type of global soundness that should be measured to evaluate the quality of a LIME explanation. First of all, *mean squared error* (or any other performance metric for numerical values) between a global and a local (used to generate explanations) models should be evaluated in the **neighbourhood** of the instance being explained to understand soundness of the surrogate model around that instance. Then, *mean squared error* in the neighbourhood of the **closest global decision boundary** should be evaluated to understand how well the local model approximates the global decision boundary in that region. Finally, *mean squared error* on the whole data set (e.g., the training data set) should be evaluated to understand overall soundness of the local model.

U2: Completeness

LIME explanations are **not** complete in their nature. For *images* the explanations are image-specific and for *text* the explanations are sentence-specific. For *tabular data* feature importance cannot be generalised beyond the single data point for which it was generated.

U3: Contextfullness

Not applicable. LIME explanations do not generalise beyond a data point for which they were generated.

U4: Interactiveness

The explanations are **static** visualisation. Interactiveness can only be achieved (by technical users) by modifying the interpretable data representation, for example, by adjusting super-pixel boundaries for images.

U5: Actionability

LIME explanations can only provide importance of a given factor on the black-box decision for a selected data point. They cannot, however, quantify its effect, which the explainee could use to precisely guide his or her future actions.

U6: Chronology

Chronology is **not** taken into account by LIME explanations.

U7: Coherence

Coherence is **not** modelled by LIME explanations.

U8: Novelty

Novelty is **not** considered by LIME explanations.

U9: Complexity

Complexity of LIME explanations cannot be directly adjusted. It can only be fine-tuned via changes to the interpretable data representation.

U10: Personalisation

LIME explanations **cannot** be personalised.

U11: Parsimony

Parsimony is introduced by the **feature selection** step for *tabular data*. Sparsity of *text* and *image* explanations is not necessary as these explanations are overlaid on top of the original image or sentence. For *text* parsimony can be also achieved by presenting the top N words in favour and against a given classification result.

Safety Requirements

S1: Information Leakage

Since LIME explanations are expressed in terms of the local model coefficients they do not leak any information. The only leakage that may occur is when creating an interpretable data representation for *tabular data* as some of the discretisation (binning) techniques may reveal characteristics of the data, e.g., quartile binning.

S2: Explanation Misuse

LIME explanations can be misused by modifying the explained data point according to the feature importance outputted by a local model. Nevertheless, this is not a straightforward task given that the explanation can be expressed in an interpretable data representation. Moreover, this importance is derived for a single data point with a local model, therefore these insights will most likely not generalise beyond that case. Discovering that the same set of factors is important for multiple individual explanations (data points) may be taken advantage of, however given that each insight is derived from a unique local model this is very unlikely.

S3: Explanation Invariance

LIME explanations may be **unstable** given that the local models are trained on sampled data. To ensure consistency the sampling procedure needs to be controlled either by fixing a random seed or by using a deterministic sampling algorithm. Ideally, the explanations would be imperceptibly different regardless of the data sample. This may be true in the limit of the number of sampled data points, however there is no consideration on the minimum number of samples required to guarantee the explanation stability.

For *images* this also depends on the stochasticity of the segmenter, which generates super-pixels. For *text* the interpretable data representation is deterministic -- a bag of words.

Another source of explanation instability for *tabular data* is the *un-discretisation* step performed by the LIME implementation on sampled data. As it stands, the LIME implementation first discretises the data (to create an interpretable data representation) and then samples within that discretised representation. This means that in order to get predictions of the global model for each sampled data point, they first have to be un-discretised. (For *images* and *text* this is straightforward as the binary vector representation has 1-to-1 mapping with super-pixels in an image or words in a sentence.) The LIME algorithm does that by sampling each feature value from within the bin boundaries, therefore introducing an extra source of randomness to each explanation. For more details please see "bLIMEy: Surrogate Prediction Explanations Beyond LIME" by Kacper Sokol, et al.

No study with respect to consistency of LIME explanations has been carried out.

S4: Explanation Quality

The quality of an explanation with respect to the confidence of a prediction given by the underlying black-box model or the distribution of the training data is **not** considered.

Validation Requirements

V1: User Studies

LIME has been evaluated on three different **user studies**:

1. choosing which of two classifiers generalises better given an explanation,
2. performing feature engineering to improve a model given insights gathered from explanations, and
3. identifying and describing classifier irregularities based on explanations.

Details of these experiments are available in Section 6 of the LIME paper.

V2: Synthetic Experiments

LIME has been evaluated on three different **simulated** user experiments:

1. validating faithfulness of explanations with respect to the global model -- the agreement between the top K most important features for a glass-box classifier and the top K features chosen by LIME as an explanation,
2. assessing trust in predictions engendered by the explanations -- identifying redundant features, and
3. identifying a better model based on the explanations.

Details of these experiments are available in Section 5 of the LIME paper.