AI Explainability 360: Impact and Design

Vijay Arya, Rachel K. E. Bellamy, Pin-Yu Chen, Amit Dhurandhar, Michael Hind, Samuel C. Hoffman, Stephanie Houde, Q. Vera Liao, Ronny Luss, Aleksandra Mojsilović, Sami Mourad, Pablo Pedemonte, Ramya Raghavendra, John Richards, Prasanna Sattigeri, Karthikeyan Shanmugam, Moninder Singh, Kush R. Varshney, Dennis Wei, Yunfeng Zhang

IBM Research
vijay.arya@in.ibm.com, {adhuran, hindm, dwei}@us.ibm.com

Abstract

As artificial intelligence and machine learning algorithms become increasingly prevalent in society, multiple stakeholders are calling for these algorithms to provide explanations. At the same time, these stakeholders, whether they be affected citizens, government regulators, domain experts, or system developers, have different explanation needs. To address these needs, in 2019, we created AI Explainability 360, an open source software toolkit featuring ten diverse and state-of-the-art explainability methods and two evaluation metrics. This paper examines the impact of the toolkit with several case studies, statistics, and community feedback. The different ways in which users have experienced AI Explainability 360 have resulted in multiple types of impact and improvements in multiple metrics, highlighted by the adoption of the toolkit by the independent LF AI & Data Foundation. The paper also describes the flexible design of the toolkit, examples of its use, and the significant educational material and documentation available to its users.

Introduction

The increasing use of artificial intelligence (AI) systems in high stakes domains has been coupled with an increase in societal demands for these systems to provide explanations for their outputs. This societal demand has already resulted in new regulations requiring explanations (Goodman and Flaxman 2016; Wachter, Mittelstadt, and Floridi 2017; Selbst and Powles 2017; Pasternak 2019). Explanations can allow users to gain insight into the system's decision-making process, which is a key component in calibrating appropriate trust and confidence in AI systems (Doshi-Velez and Kim 2017; Varshney 2019).

However, many machine learning techniques, which are responsible for much of the advances in AI, are not easily explainable, even by experts in the field. This has led to a growing research community (Kim, Varshney, and Weller 2018), with a long history, focusing on "interpretable" or "explainable" machine learning techniques.[1] However, despite the growing volume of publications, there remains a gap between what the research community is producing and how it can be leveraged by society.

One reason for this gap is that different people in different settings may require different kinds of explanations (Tomsett et al. 2018; Hind 2019). We refer to the people interacting with an AI system as *consumers*, and to their different types as *personas*. For example, a doctor trying to understand an AI diagnosis of a patient may benefit from seeing known similar cases with the same diagnosis. A denied loan applicant will want to understand the reasons for their rejection and what can be done to reverse the decision. A regulator, on the other hand, will want to understand the behavior of the system as a whole to ensure that it complies with the law. A developer may want to understand where the model is more or less confident as a means of improving its performance.

As a step toward addressing the gap, in 2019, we released the AI Explainability 360 (AIX360) open source software toolkit for explaining machine learning models and data (Arya et al. 2020). The toolkit currently features ten explainability methods (listed in Table 1) and two evaluation metrics from the literature (Alvarez-Melis and Jaakkola 2018; Luss et al. 2021). We also introduced a taxonomy to navigate the space of explanation methods, not only the ten in the toolkit but also the broader literature on explainable AI. The taxonomy was intended to be usable by consumers with varied backgrounds to choose an appropriate explanation method for their application. AIX360 differs from other open source explainability toolkits (see Arya et al. (2020) for a list) in two main ways: 1) its support for a broad and diverse spectrum of explainability methods, implemented in a common architecture, and 2) its educational material as discussed below.

The main purpose of the current paper, two years after the release of AIX360, is to look back at the impact that it has had. More specifically

Impact: We describe some benefits of the toolkit from our experiences with it, spanning use cases in finance, manufacturing, and IT support, as well as community metrics and feedback. Due to the variety of ways in which others have experienced AIX360 and its algorithms, it has had multiple types of impact: operational, educational, competition, and societal. It has correspondingly brought improvements in multiple metrics: accuracy, semiconductor yield, satisfaction rate, and domain expert time.

[1] We use the terms explainable and interpretable fairly interchangeably; some scholars make a strong distinction (Rudin 2019).

BRCG (Dash, Günlük, and Wei 2018)	Learns a small, interpretable Boolean rule in disjunctive normal form (DNF) for binary classification.
GLRM (Wei et al. 2019)	Learns a linear combination of conjunctions for real-valued regression through a generalized linear model (GLM) link function (e.g., identity, logit).
ProtoDash (Gurumoorthy et al. 2019)	Selects diverse and representative samples that summarize a dataset or explain a test instance. Non-negative importance weights are also learned for each of the selected samples.
ProfWeight (Dhurandhar et al. 2018b)	Learns a reweighting of the training set based on a given interpretable model and a high-performing complex neural network. Retraining of the interpretable model on this reweighted training set is likely to improve the performance of the interpretable model.
TED (Hind et al. 2019)	Learns a predictive model based not only on input-output labels but also on user-provided explanations. For an unseen test instance both a label and explanation are returned.
CEM (Dhurandhar et al. 2018a)	Generates a local explanation in terms of what is minimally sufficient to maintain the original classification, and also what should be necessarily absent.
CEM-MAF (Luss et al. 2021)	For complex images, creates contrastive explanations like CEM above but based on high-level semantically meaningful attributes.
DIP-VAE (Kumar, Sattigeri, and Balakrishnan 2018)	Learns high-level independent features from images that possibly have semantic interpretation.
LIME (Ribeiro, Singh, and Guestrin 2016)	Obtains local explanations by fitting a sparse linear model locally. The code is integrated from the library maintained by its authors: https://github.com/marcotcr/lime.
SHAP (Lundberg and Lee 2017)	Identifies feature importances based on Shapley value estimation methods. The code is integrated from the authors' repository: https://github.com/slundberg/shap.

Table 1: The AI Explainability 360 toolkit (`v0.2.1`) includes a diverse collection of explainability algorithms

In addition, we discuss aspects of the toolkit and accompanying materials that illustrate its design, ease of use, and documentation. This discussion expands upon the brief summary given in Arya et al. (2020).

Toolkit Design: We describe the architecture that enables coverage of a diversity of explainability methods as well as extensions to the toolkit since its initial release. Code listings show how its methods can be easily called by data scientists.

Educational Material: We discuss the resources available to make the concepts of explainable AI accessible to non-technical stakeholders. These include a web demonstration, which highlights how three different personas in a loan application scenario can be best served by different explanation methods. Five Jupyter notebook tutorials show data scientists how to use different methods across several problem domains, including lending, health care, and human capital management.

These contributions demonstrate how the design and educational material of the AIX360 toolkit have led to the creation of better AI systems.

Initial Impact

This section highlights the impact of the AIX360 toolkit in the first two years since its release. It describes several different forms of impact on real problem domains and the open source community. This impact has resulted in improvements in multiple metrics: accuracy, semiconductor yield, satisfaction rate, and domain expert time.

The current version of the AIX360 toolkit includes ten explainability algorithms described in Table 1 covering different ways of explaining. Explanation methods could be either local or global, where the former refers to explaining an AI model's decision for a single instance, while the latter refers to explaining a model in its entirety. Another dimension of separation is that explanation methods are typically either feature-based or exemplar-based. The former provides key features as an explanation, while the latter provides a list of most relevant instances. Under feature-based, there are also methods termed as contrastive/counterfactual explanations. These provide the most sensitive features which, if slightly changed, can significantly alter the output of the AI model. Another type here are rule-based methods which output semantically meaningful rules and could be considered a subtype of feature-based explanations.

Financial Institution: Educational Impact

Large financial institutions typically have dedicated teams to ensure transparency and trustworthiness of their deployed models. These teams validate models built by the model development team. After the release of the AIX360 toolkit, one such financial institution approached us to educate their data science team in a newly founded "center of excellence". Based on real use cases seen by this team they created multiple explainability use cases that varied in modality (tabular, image, and text) and model types (LSTMs, CNNs, RNNs, boosted trees). The goal for each use case/model was to answer the following four questions:

Q1: Generally, what features are the most important for decisions made by the model?

Q2: What features drove a decision for a certain input?

Q3: What features could be minimally changed to alter the

decision for an input?

Q4: Do similar inputs produce the same decision?

These questions provide pragmatic examples of an enterprise's requirement for an explainability technique, which is more concrete than simply "the model should be able to explain its decision."

These questions also represent diversity in the types of explanations needed. Q1 is a global explainability question. Q2 and Q3 are local feature-based explainability questions, where Q3 is requiring a contrastive explanation. Q4 is a local exemplar-based explainability question. All these questions were answerable through one or more methods available through AIX360: Q1 $\rightarrow$ BRCG, GLRM, ProfWeight; Q2 $\rightarrow$ LIME, SHAP; Q3 $\rightarrow$ CEM, CEM-MAF and Q4 $\rightarrow$ ProtoDash. In fact, we learned that some of these questions were inspired from our toolkit and the methods it possesses. This demonstrates that not only does the toolkit address many real explainability questions, but it also can help structure thinking about this space in relation to real problems. Thus, its contributions are both technical and conceptual.

The result of the engagement was that the data science team was able to successfully test out many of our methods on the different use cases covering the four questions. They came away with newly acquired expertise in this space due in large part to AIX360's existence.

Semiconductor Manufacturing: Operational Impact

Semiconductor manufacturing is a multibillion-dollar industry, where producing a modern microprocessor chip is a complex process that takes months. A semiconductor manufacturer was using a model to estimate the quality of a chip during an etching process, precluding the use of standard tools that are expensive (cost millions of dollars) and time-consuming (can take several days). The engineers' goal was not only to predict quality accurately but also to obtain insight into ways in which they can improve the process. They specifically wanted a decision tree type of model which they were comfortable interpreting. Our goal was thus to build the most accurate "smallish" decision tree we could. Using the ProfWeight explanability algorithm (Dhurandhar et al. 2018b), we transferred information from an accurate neural network to a decision tree, elevating its performance by $\approx$ 13% making it also accurate. We reported the top features: certain pressures, time since last cleaning, and certain acid concentrations. Based on these insights, the engineer started controlling some of them more tightly, improving the total number of within-spec wafers by 1.3%. In this industry a 1% increase in yield can amount to billions of dollars in savings.

Information Technology: Operational Impact

The IT division of a large corporation used a natural language processing (NLP) model to classify customer complaints into a few hundred categories. Although this model achieved close to 95% accuracy, the inability to explain the misclassifications led to distrust of the system. The team used the CEM explainability algorithm (Dhurandhar et al. 2018a) to compute local explanations. The experts said that such explanations are highly valuable in providing stakeholders and end-users with confidence in the inner workings of the classifier. Approximately, 80% of the explanations of misclassified complaints were deemed reasonable by them compared to 40% with the previous approach that resembled LIME (Ribeiro, Singh, and Guestrin 2016). The experts said the algorithm provided much better intuition about why the system made a mistake, showing in most cases that the mistake was acceptable. They felt that this was useful in developing trust in the system. Given the success of our technique for this problem, CEM has been integrated into their multi-cloud management platform to help accelerate their clients' journey to cloud. Initial parts of the work were also described in a blog post (Ayachitula and Khandekar 2019).

Consumer Finance: Competition Impact

The Fair Isaac Corporation (FICO) is well known for its FICO score, the predominant consumer credit score in the US. FICO organized an Explainable Machine Learning Challenge (FICO 2018) around a real-world dataset of home equity line of credit (HELOC) applications. The tasks were to accurately predict whether applicants would satisfactorily repay the HELOC as well as provide local and global explanations. We used the BRCG algorithm (Dash, Günlük, and Wei 2018) to produce a remarkably simple, directly interpretable rule set model, consisting of only two rules, each with three conditions. The model's accuracy of 72% was also close to the best achieved by any model of around 74%.

This submission was awarded first place for the highest score in an empirical evaluation. The scores of our submission and other submissions were not disclosed. We do know, however, from a presentation made by FICO that the evaluation involved data scientists with domain knowledge being presented with local and global explanations, without model predictions. They were then asked to predict the model output. Submissions were scored based on a linear combination of the data scientists' predictive accuracy (weight of 70%) and time taken (30%). Based on this description, we conclude that (1) directly interpretable models as provided by AIX360 can offer an appealing combination of accuracy and explainability, especially for regulated industries such as consumer lending, and (2) such models may be preferred by human decision-makers who have to understand and work with them, particularly under time constraints.

Regulator: Educational Impact

Some requirements for AI Explainability come from industry-specific regulation. Complementary to the previous examples, we were contacted by a large group from a major financial regulator to leverage our expertise in the creation of the toolkit and taxonomy. The group wanted to get a deeper understanding of AI explainability techniques to determine how they should update their explainability regulations for AI models. Financial regulation is trying to ensure credit is extended without taking on unnecessary risk. More accurate models can help achieve this goal, but often they are not used because of existing XAI regulations. The group hoped that some of the techniques in the toolkit could be used as a basis for future regulation of financial institutions.

Metric	Value
Forks	190
Stars	923
Last 14-day avg. of github views/day	182
Last 14-day avg. of github unique visitors/day	35.1
Last 14-day avg. of unique github clones/day	2.6
Total PyPI downloads	26,218
AIX360 Slack users	261
Closed pull requests (PRs)	71
Public presentations/tutorials views	6,849

Table 2: Usage and community statistics as of September 9, 2021 for the AIX360 toolkit, released in August 2019.

Open Source Community: Societal Impact

We quantify the impact of AIX360's release in the open source community via three channels: the GitHub repository, PyPI package repository, and the public Slack workspace (aix360.slack.com). The usage and community-building statistics are given in Table 2. In addition to these metrics, the three channels provide qualitative evidence of interaction with the community, chiefly problem reports or feature requests on GitHub, and questions about algorithms and event announcements on Slack. Another form of engagement comes from public presentations and conference tutorials. There have been ten presentations, some of which were captured as videos with over 6,000 views.

Another measure of impact is the adoption of the toolkit by independent bodies. One example of this is the LF AI & Data Foundation's accepting the toolkit as an incubator project in September, 2020 (LF AI & Data 2020a). This open governance organization has over 50 corporate and university members and "supports open source innovation in artificial intelligence, machine learning, deep learning, and data" (LF AI & Data 2020b).

We have encouraged the community to make their own contributions to AIX360. A good example of a community contribution is Floid Gilbert's FeatureBinarizerFromTrees, which uses decision trees to binarize features more intelligently for the BRCG and GLRM algorithms. Additionally, the authors of the following four papers are currently integrating new algorithms based on counterfactual explanations and influence functions into the toolkit: Galhotra, Pradhan, and Salimi (2021); Le, Wang, and Lee (2020); Zhang et al. (2021); Li et al. (2021). The next version of the toolkit (`v0.2.2`) will include these four additional explainability algorithms and is planned to be released by Dec 2021.

The AIX360 toolkit offers an extensible software architecture and its licensing structure (Apache v2.0) permits free commercial use and distribution of derivative works. This paves the way for enterprises to build new commercial offerings that leverage the toolkit. Presently, the toolkit is integrated into IBM's Cloud Pak for Data and is the foundational component of a commercial explainability library offered by IBM with additional proprietary algorithms and software support.

In addition to the above statistics, contributions and extensions, we have also received unsolicated feedback from the broader community via our public slack channel (IBM Research 2019).

> *"What a fantastic resource (AIX360 is)! Thanks to everyone working on it."*
>
> — John C. Havens, Executive Director of IEEE Global Initiative on Ethics of Autonomous and Intelligent Systems

> *"I have found aix360 to be most comprehensive."*
>
> — Arpit Sisodia, Data Scientist with Ericsson

Consistent with their roles, John particularly appreciated our educational material, and Arpit found the toolkit to have the best coverage of a diverse set of explainability questions.

AIX360 Design and Usage

The AIX360 toolkit offers a unified, flexible, extensible, and easy-to-use programming interface and an associated software architecture to accommodate the diversity of explainability techniques required by various stakeholders. The goal of the architectural design is to be amenable both to data scientists, who may not be experts in explainability, as well as explainability algorithm developers. Toward this end, we make use of a programming interface that is similar to popular Python model development tools (e.g., scikit-learn) and construct a hierarchy of Python classes corresponding to explainers for data, models, and predictions. Explainability algorithm developers can inherit from a family of base class explainers to integrate new explainability algorithms. The base class explainers are organized according to the AI modeling pipeline shown in Figure 1, based upon their use in offering explanations at different stages. Below we provide a summary of the various AIX360 classes and illustrate their usage via example in Listing 1.

- *Data explainers:* These explainers are implemented using the base class DIExplainer (Directly Interpretable unsupervised Explainer), which provides abstract methods to implement unsupervised techniques that explain datasets. The AIX360 explainers that inherit from this base class include ProtodashExplainer and DIPVAEExplainer.
- *Directly interpretable explainers:* These explainers are implemented using the base class DISExplainer (Directly Interpretable Supervised Explainer), which includes abstract methods to train interpretable models directly from labelled data. The explainers that inherit from this base class and implement its methods include BRCGExplainer and GLRMExplainer. Additionally, the TED_CartesianExplainer, which trains models using data that is labelled with persona-specific explanations, also inherits from DISExplainer. Listing 1 shows an example illustrating the use of BRCGExplainer.
- *Local post-hoc explainers:* These are further subdivided into black-box and white-box explainers. The black-box explainers are model-agnostic and generally require access only to a model's prediction function. This class of explainers is implemented via the base class LocalBBExplainer. Our wrapper implementations of the publicly available LIME and SHAP algorithms inherit from LocalBBExplainer. The white-box explainers generally re-

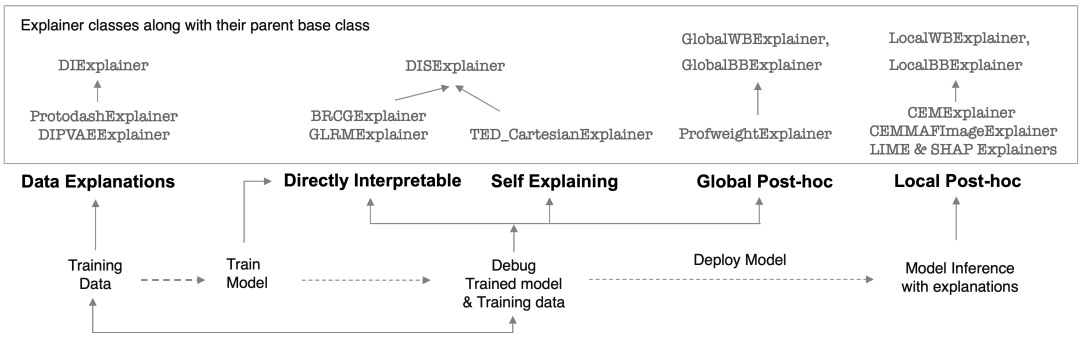


Figure 1: Organization of AIX360 explainer classes according to their use in various steps of the AI modeling pipeline.

quire access to a model's internals, such as its loss function, and are implemented via the base class LocalWBExplainer. CEMExplainer and CEMMAFImageExplainer both inherit from LocalWBExplainer.

- *Global post-hoc explainers:* These are subdivided into black-box and white-box explainers as above. The corresponding base classes GlobalWBExplainer and GlobalBBExplainer, include abstract methods that can help train interpretable surrogate models, given a source model along with its data. The ProfweightExplainer, which is an example of a global post-hoc white box explainer, inherits from the base class GlobalWBExplainer.
- *Dataset and Model API classes:* In addition to explainer classes, AIX360 includes several dataset classes to facilitate loading and processing of commonly used datasets so that users can easily experiment with the implemented algorithms.

  To allow users to explain models that have been built using different deep learning frameworks (e.g. TensorFlow, Keras, PyTorch, MXNet) while avoiding the need to implement explainability algorithms multiple times for each framework, AIX360 includes framework-specific classes that expose a common model API needed by explainability algorithm developers. The current version of the toolkt includes model API classes for Keras (based on TensorFlow) and Pytorch models.

Listing 1: A python code example illustrating the use of BRCGExplainer (directly interpretable explainer).

```
from aix360.algorithms.rbm import
    BRCGExplainer, BooleanRuleCG
# Instantiate and train an explainer to
    compute global rules in conjunctive
    normal form (CNF)
br = BRCGExplainer(BooleanRuleCG(CNF=
    True))
br.fit(x_train, y_train)
# print the CNF rules
print (br.explain()['rules'])
```

Educational Material

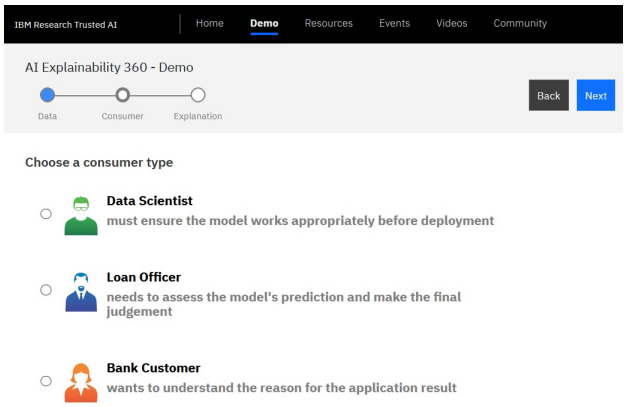
AIX360 was developed with the goal of providing accessible resources on explainability to nontechnical stakeholders. Therefore, we include numerous educational materials to both introduce the explainability algorithms provided by AIX360, and to demonstrate how different explainability methods can be applied in real-world scenarios. These educational materials include general guidance for the key concepts of explainability, a taxonomy of algorithms to help a user choose the appropriate one for their use case, a web demo that illustrates the usage of different explainability methods, and multiple tutorials.

A key tenet from our initial work is that "One Explanation Does Not Fit All", i.e., different explanation consumers will have different needs, which can be met by different explanation techniques (Arya et al. 2019). The web demo (IBM Research 2019) was created to illustrate this point. It is based on the FICO Explainable Machine Learning Challenge dataset (FICO 2018), a real-world scenario where a machine learning system is used to support decisions on loan applications by predicting the repayment risk of the applicants. The demo highlights that three groups of people – data scientists, loan officers, and bank customers – are involved in the scenario, and their needs are best served by different explainability methods. For example, although the data scientist may demand a global understanding of model behavior through an interpretable model, which can be provided by the GLRM algorithm, a bank customer would ask for justification for their loan application results, which can be generated by the CEM algorithm. We use storytelling and visual illustrations to guide users of AIX360 through these scenarios of different explainability consumers. Figure 2 shows screenshots from the demo.

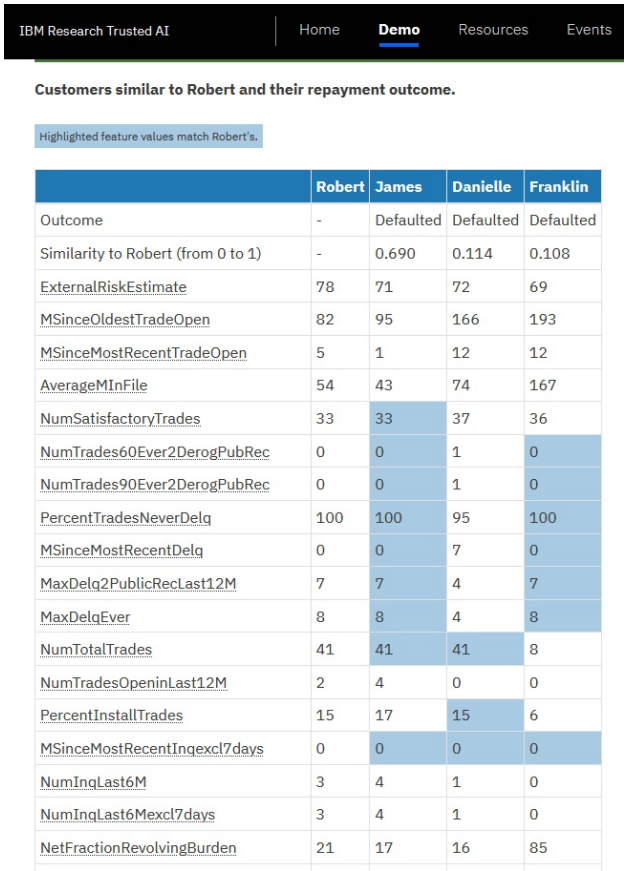
The AIX360 toolkit currently includes five tutorials in the form of Jupyter notebooks that show data scientists and other developers how to use different explanation methods across several application domains. The tutorials thus serve as an educational tool and potential gateway to AI explainability for practitioners in these domains. The tutorials cover the following industry use cases:

1. Using 3 different methods to explain a credit approval model to 3 types of consumers, based on the FICO Explainable Machine Learning Challenge dataset (FICO 2018).
2. Creating directly interpretable healthcare cost prediction models for a care-management scenario using Medical Expenditure Panel Survey data.
3. Explaining dermoscopic image datasets used to train machine learning models by uncovering semantically meaning-

(a)



(b)



	Robert	James	Danielle	Franklin
Outcome	-	Defaulted	Defaulted	Defaulted
Similarity to Robert (from 0 to 1)	-	0.690	0.114	0.108
ExternalRiskEstimate	78	71	72	69
MSinceOldestTradeOpen	82	95	166	193
MSinceMostRecentTradeOpen	5	1	12	12
AverageMInFile	54	43	74	167
NumSatisfactoryTrades	33	33	37	36
NumTrades60Ever2DerogPubRec	0	0	1	0
NumTrades90Ever2DerogPubRec	0	0	1	0
PercentTradesNeverDelq	100	100	95	100
MSinceMostRecentDelq	0	0	7	0
MaxDelq2PublicRecLast12M	7	7	4	7
MaxDelqEver	8	8	4	8
NumTotalTrades	41	41	41	8
NumTradesOpeninLast12M	2	4	0	0
PercentInstallTrades	15	17	15	6
MSinceMostRecentInqexcl7days	0	0	0	0
NumInqLast6M	3	4	1	0
NumInqLast6Mexcl7days	3	4	1	0
NetFractionRevolvingBurden	21	17	16	85

Figure 2: The web demo illustrates how different types of explanations are appropriate for different personas (image (a)). Image (b) shows a subset of the Protodash explainer output to illustrate how similar applicants (i.e. prototypes) in the training data were given the same decision.

ful features that could help physicians diagnose skin diseases.
4. Explaining National Health and Nutrition Examination Survey datasets to support research in epidemiology and health policy by effectively summarizing them.
5. Explaining predictions of a model that recommends employees for retention actions from a synthesized human resources dataset.

The tutorials not only illustrate the application of different methods but also provide considerable insight into the datasets that are used and, to the extent that these insights generalize, into the respective problem domains. These insights are a natural consequence of using explainable machine learning and could be of independent interest.

Discussion

We have examined the impact of the open source AI Explainability 360 (AIX360) toolkit two years after its initial release. A major motivation for creating the toolkit was that different personas interacting with an AI system have different goals and require different kinds of explanations. This diversity has been borne out in the multiple types of impact that we have discussed, from operational to societal, and in the metrics that have been improved, from accuracy to user satisfaction. We have also discussed how the design of the toolkit supports a range of explanation methods and extensions, given examples of its use, and described the educational material that makes it accessible to practitioners and nonexperts.

The experience of interacting and working with a diverse collection of AIX360 users helped us learn some useful lessons. One lesson learned from working with the financial institution described in section 'Initial Impact' is the importance of supporting multiple deep learning frameworks. The AIX360 toolkit's model API classes helped us explain models trained in different deep learning frameworks without the need to re-implement the explainability algorithms. The model API class currently supports Keras and PyTorch models. We aim to extend it to cover other popular deep learning frameworks in future. Another lesson learned is the importance of providing platform-specific installables for air-gapped environments in financial institutions where internet is disabled for security reasons. The library currently supports installation via Pip (package installer for Python), and we plan to provide platform-specific Conda installation packages in future. We also learned that basic education, via the examples illustrated in our demo, has been quite valuable to users, even before they use the toolkit. This is particularly valuable to level set expectations, where different users otherwise may have different intuitive views on the meaning of "explainability". Another pragmatic lesson learned is that explainability techniques that are model agnostic and leverage probing, i.e, querying the black box model for predictions on generated examples, can raise issues that are not currently discussed by most researchers. These probes can be in the 100s per example to be explained, resulting in a direct cost to the model owner when they are charged per prediction, either directly or indirectly. Such probes can also stress the scalability of the model inference platform.

References

Alvarez-Melis, D.; and Jaakkola, T. 2018. Towards Robust Interpretability with Self-Explaining Neural Networks. In *Neural Information Processing Systems*, 7775–7784.

Arya, V.; Bellamy, R. K. E.; Chen, P.-Y.; Dhurandhar, A.; Hind, M.; Hoffman, S. C.; Houde, S.; Liao, Q. V.; Luss, R.; Mojsilović, A.; Mourad, S.; Pedemonte, P.; Raghavendra, R.; Richards, J.; Sattigeri, P.; Shanmugam, K.; Singh, M.; Varshney, K. R.; Wei, D.; and Zhang, Y. 2019. One Explanation Does Not Fit All: A Toolkit and Taxonomy of AI Explainability Techniques. arXiv:1909.03012.

Arya, V.; Bellamy, R. K. E.; Chen, P.-Y.; Dhurandhar, A.; Hind, M.; Hoffman, S. C.; Houde, S.; Liao, Q. V.; Luss, R.; Mojsilović, A.; Mourad, S.; Pedemonte, P.; Raghavendra, R.; Richards, J. T.; Sattigeri, P.; Shanmugam, K.; Singh, M.; Varshney, K. R.; Wei, D.; and Zhang, Y. 2020. AI Explainability 360: An Extensible Toolkit for Understanding Data and Machine Learning Models. *Journal of Machine Learning Research*, 21(130): 1–6.

Ayachitula, A.; and Khandekar, R. 2019. AI for IT: Local Explainability for unstructured text classification. https://www.linkedin.com/pulse/ai-local-explainability-unstructured-text-naga-arun-ayachitula.

Dash, S.; Günlük, O.; and Wei, D. 2018. Boolean Decision Rules via Column Generation. In *Neural Information Processing Systems*, 4655–4665.

Dhurandhar, A.; Chen, P.-Y.; Luss, R.; Tu, C.-C.; Ting, P.; Shanmugam, K.; and Das, P. 2018a. Explanations based on the Missing: Towards Contrastive Explanations with Pertinent Negatives. In *Neural Information Processing Systems*, 592–603.

Dhurandhar, A.; Shanmugam, K.; Luss, R.; and Olsen, P. 2018b. Improving Simple Models with Confidence Profiles. In *Neural Information Processing Systems*, 10296–10306.

Doshi-Velez, F.; and Kim, B. 2017. Towards A Rigorous Science of Interpretable Machine Learning. arXiv:1702.08608v2.

FICO. 2018. FICO Explainable Machine Learning Challenge. https://community.fico.com/s/explainable-machine-learning-challenge.

Galhotra, S.; Pradhan, R.; and Salimi, B. 2021. Explaining Black-Box Algorithms Using Probabilistic Contrastive Counterfactuals. In *ACM International Conference on Management of Data (SIGMOD/PODS)*, 577–590.

Goodman, B.; and Flaxman, S. 2016. EU Regulations on Algorithmic Decision-Making and a 'Right to Explanation'. In *ICML Workshop Human Interp. Mach. Learn.*, 26–30.

Gurumoorthy, K.; Dhurandhar, A.; Cecchi, G.; and Aggarwal, C. 2019. Efficient Data Representation by Selecting Prototypes with Importance Weights. In *IEEE International Conference on Data Mining*.

Hind, M. 2019. Explaining Explainable AI. *ACM XRDS Mag.*, 25(3): 16–19.

Hind, M.; Wei, D.; Campbell, M.; Codella, N. C. F.; Dhurandhar, A.; Mojsilovic, A.; Ramamurthy, K. N.; and Varshney, K. R. 2019. TED: Teaching AI to Explain its Decisions. In *AAAI/ACM Conference on AI, Ethics, and Society*, 123–129.

IBM Research. 2019. AI Explainability 360. http://aix360.mybluemix.net. Accessed: 2021-09-09.

Kim, B.; Varshney, K. R.; and Weller, A., eds. 2018. *ICML Workshop on Human Interpretability in Machine Learning*.

Kumar, A.; Sattigeri, P.; and Balakrishnan, A. 2018. Variational Inference of Disentangled Latent Concepts from Unlabeled Observations. In *International Conference on Learning Representations*.

Le, T.; Wang, S.; and Lee, D. 2020. GRACE: Generating Concise and Informative Contrastive Sample to Explain Neural Network Model's Prediction. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 238–248.

LF AI & Data. 2020a. AI Explainability 360. https://lfaidata.foundation/projects/aiexplainability360. Accessed: 2021-09-09.

LF AI & Data. 2020b. OSS Innovation in AI, ML, DL, and Data. https://lfaidata.foundation/. Accessed: 2021-09-09.

Li, J.; Zheng, L.; Zhu, Y.; and He, J. 2021. Outlier Impact Characterization for Time Series Data. *AAAI Conference on Artificial Intelligence*, 35(13): 11595–11603.

Lundberg, S. M.; and Lee, S.-I. 2017. A Unified Approach to Interpreting Model Predictions. In *Neural Information Processing Systems*, 4765–4774.

Luss, R.; Chen, P.-Y.; Dhurandhar, A.; Sattigeri, P.; Shanmugam, K.; and Tu, C.-C. 2021. Leveraging Latent Features for Local Explanations. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

Pasternak, D. B. 2019. Illinois and City of Chicago Poised to Implement New Laws Addressing Changes in the Workplace — Signs of Things to Come? *National Law Review*.

Ribeiro, M. T.; Singh, S.; and Guestrin, C. 2016. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1135–1144.

Rudin, C. 2019. Stop Explaining Black Box Machine Learning Models for High Stakes Decisions and Use Interpretable Models Instead. *Nature Mach. Intell.*, 1(5): 206–215.

Selbst, A. D.; and Powles, J. 2017. Meaningful Information and the Right to Explanation. *Int. Data Privacy Law*, 7(4): 233–242.

Tomsett, R.; Braines, D.; Harborne, D.; Preece, A. D.; and Chakraborty, S. 2018. Interpretable to Whom? A Role-based Model for Analyzing Interpretable Machine Learning Systems. In *ICML Workshop Human Interp. Mach. Learn.*

Varshney, K. R. 2019. Trustworthy Machine Learning and Artificial Intelligence. *ACM XRDS Mag.*, 25(3): 26–29.

Wachter, S.; Mittelstadt, B.; and Floridi, L. 2017. Why a Right to Explanation of Automated Decision-Making Does Not Exist in the General Data Protection Regulation. *Int. Data Privacy Law*, 7(2): 76–99.

Wei, D.; Dash, S.; Gao, T.; and Günlük, O. 2019. Generalized Linear Rule Models. In *International Conference on Machine Learning*, 6687–6696.

Zhang, W.; Huang, Z.; Zhu, Y.; Ye, G.; Cui, X.; and Zhang, F. 2021. On Sample Based Explanation Methods for NLP: Faithfulness, Efficiency and Semantic Evaluation. In *Annual Meeting of the Association for Computational Linguistics and Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 5399–5411.