

Appendices

A Exact Weighted Formal Feature Attribution

In this appendix, we once again limit our analysis to instances where we can calculate the *exact* WFFA values for the instance of interest by enumerating all AXp’s. Also, the settings used in Section 5 are applied here, i.e. we take the absolute values of feature attribution assigned by LIME and SHAP, and normalize them within the range of $[0, 1]$. Just like in the main text of the paper, we then compare these approaches with normalized WFFA values in terms of errors, Kendall’s Tau [31] and rank-biased overlap (RBO) [66].

A.1 Tabular Data

A comparison of WFFA, LIME and SHAP on an instance of the Compas dataset [3] is exemplified in Figure 7. We can observe the patterns similar to those depicted in Figure 3. The feature that WFFA considers most important is “Asian” while this viewpoint is shared by LIME but disputed by SHAP. However, neither LIME nor SHAP fully align with WFFA, although there is evident similarity between them. As with FFA, these observations can be generalized to the other instances of Compas, as discussed below.

Table 5 presents a comparison of WFFA against LIME, and SHAP on the 11 selected tabular datasets as in Table 1, demonstrating similarities in the findings observed for WFFA and FFA for these datasets. The average runtime for generating the exact WFFA in a dataset varies between 0.18 and 1.89 seconds while the average number of AXp’s per instance to explain and so to compute exact WFFA in a dataset ranges from 1.40 to 33.33. Both LIME and SHAP process each image in less than one second. LIME exhibits errors ranging from 1.37 to 4.96 across these datasets while SHAP shows similar errors spanning from 1.36 to 4.67. Besides errors, LIME and SHAP yield comparable outcomes in terms of the two ranking comparison metrics. The values of Kendall’s Tau for LIME span from -0.35 to 0.25 , whereas the values for SHAP are between -0.38 and 0.31 . Regarding RBO values, LIME (resp. SHAP) demonstrates values ranging from 0.38 to 0.69 (resp. 0.43 to 0.67). Overall and consistent with the FFA findings shown earlier in Table 1, Table 5 indicates that both LIME and SHAP fail to achieve close enough agreement with WFFA.

A.2 10×10 Digits

Table 6 provides a comprehensive comparison of approximate WFFA against feature attribution reported by LIME and SHAP with respect to the exact WFFA values, conducted on the downscaled MNIST digits and PneumoniaMNIST images, where exhaustive AXp enumeration is feasible. The values of feature attribution generated by LIME, SHAP, and approximate WFFA_{*} for the three selected 10×10 images are shown in Figure 11, Figure 12, and Figure 13. Over time, the number of features included in the AXp’s increases, and the weighted attribution of each feature changes converging to the exact WFFA. The results shown in Figure 8, Figure 9, and Figure 10 align with the main finding for FFA approximation shown earlier. Furthermore, the results shown in Table 6 are also consistent with FFA observations in Table 2. Both LIME and SHAP can process each image within a runtime of less than one second. The average runtime and average number of AXp’s generated for 10×10 MNIST 1 vs 3 (resp. 1 vs 7) are 14264.78s and 15781.87 (resp. 6834.61s and 4028.27), while the values in 10×10 PneumoniaMNIST are 8656.18s and 8802.87, respectively. Similarly to the results in Table 2, Table 6 indicates that our approximation yields small errors. Even after 10 seconds, it outperforms both LIME and SHAP, and the errors continue to decrease as we compute more AXp’s. Once again, the results of the orderings demonstrate that after 10 seconds, the ordering of WFFA_{*} approaches closer to the exact WFFA compared to both LIME and SHAP and converges to the exact WFFA ordering with the growth of the number AXp’s enumerated. As can also be seen, LIME exhibits a substantial distance from the *exact* WFFA ordering.

A.3 Summary

The findings of this section again indicate that we can confidently obtain valuable approximations of the exact WFFA values without the need to exhaustively enumerate all AXp’s for a given data

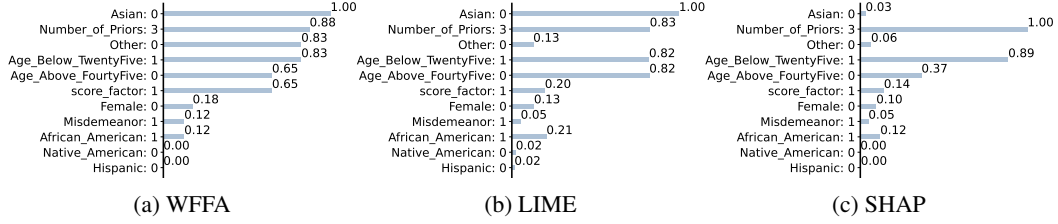


Figure 7: Explanations for an instance of Compas $\mathbf{v} = \{\text{\#Priors} = 3, \text{Score_factor} = 1, \text{Age_Above_FourtyFive} = 0, \text{Age_Below_TwentyFive} = 1, \text{African_American} = 1, \text{Asian} = 0, \text{Hispanic} = 0, \text{Native_American} = 0, \text{Other} = 0, \text{Female} = 0, \text{Misdemeanor} = 1\}$ predicted as $\text{Two_yr_Recidivism} = \text{true}$.

Table 5: LIME and SHAP versus WFFA on tabular data.

Dataset	adult	appendicitis	australian	cars	compas	heart-statlog	hungarian	lending	liver-disorder	pima	recidivism
$ \mathcal{F} $	(12)	(7)	(14)	(8)	(11)	(13)	(13)	(9)	(6)	(8)	(15)
Approach											
Error											
LIME	4.32	2.06	4.96	1.48	3.26	4.40	4.43	1.37	2.37	2.63	4.66
SHAP	4.29	1.87	4.31	1.36	2.63	3.61	4.00	1.43	2.25	2.91	4.67
Kendall's Tau											
LIME	0.11	0.17	0.25	-0.08	-0.08	0.22	0.08	-0.35	-0.17	0.25	0.08
SHAP	0.07	0.23	0.31	-0.07	-0.07	0.22	0.26	-0.38	-0.16	0.15	0.16
RBO											
LIME	0.53	0.65	0.48	0.64	0.56	0.56	0.40	0.59	0.65	0.69	0.38
SHAP	0.48	0.67	0.55	0.66	0.59	0.52	0.49	0.61	0.67	0.64	0.43

Table 6: Comparison on 10×10 Images of WFFA versus LIME, SHAP and WFFA approximations.

Dataset	LIME	SHAP	WFFA ₁₀	WFFA ₃₀	WFFA ₆₀	WFFA ₁₂₀	WFFA ₆₀₀	WFFA ₁₂₀₀
$ \mathcal{F} = 100$	Error							
10×10-mnist-1vs3	11.28	9.81	5.52	5.12	4.83	4.50	3.32	2.61
10×10-mnist-1vs7	12.46	8.11	4.07	3.47	2.83	2.38	1.34	0.97
10×10-pneumoniarnist	17.25	17.84	5.33	4.29	3.76	3.36	2.20	1.63
Kendall's Tau								
10×10-mnist-1vs3	-0.14	0.48	0.53	0.60	0.64	0.67	0.75	0.81
10×10-mnist-1vs7	-0.33	0.47	0.58	0.65	0.73	0.79	0.86	0.90
10×10-pneumoniarnist	-0.02	0.24	0.67	0.74	0.80	0.81	0.90	0.92
RBO								
10×10-mnist-1vs3	0.20	0.50	0.63	0.67	0.70	0.74	0.81	0.84
10×10-mnist-1vs7	0.19	0.58	0.73	0.77	0.81	0.86	0.90	0.91
10×10-pneumoniarnist	0.21	0.37	0.63	0.70	0.74	0.77	0.82	0.87

instance. It is worth noting that feature attribution determined by LIME and SHAP is quite inaccurate and does not provide meaningful insights to a human decision-maker, despite being computationally fast.

B Approximate Weighted Formal Feature Attribution

As argued in Section 3, the exact WFFA computation can be difficult in practice, due to the complexity of the problem. But as Table 6 indicates, our approach can yield decent WFFA approximations even with a short duration of collecting AXp's. Here we assess the fidelity of our approach in contrast to the approximate WFFA computed after a duration of 2 hours (7200s). WFFA_s and the values of feature attribution generated by LIME and SHAP for the three considered 28×28 images are depicted in Figure 14, 15, and 16. As time progresses, the accumulated AXp's incorporate an increasing number of features, and as a result the value of weighted attribution for each feature can change. Table 7 details the comparison between LIME, SHAP, and the approximate WFFA. Both LIME and

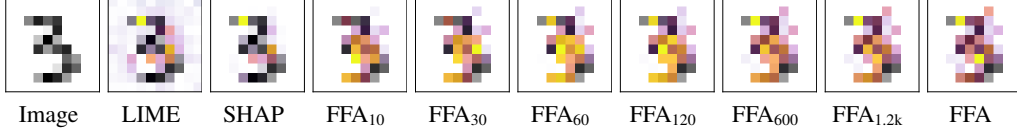


Figure 8: 10×10 MNIST 1 vs. 3. The prediction is 3.

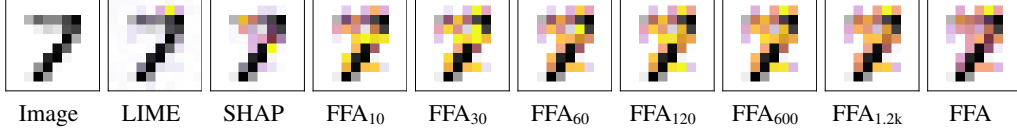


Figure 9: 10×10 MNIST 1 vs. 7. The prediction is 7.

Table 7: Comparison on 28×28 Images of WFFA_{7.2k} versus LIME, SHAP and WFFA approximations.

Dataset	LIME	SHAP	WFFA ₁₀	WFFA ₃₀	WFFA ₁₂₀	WFFA ₆₀₀	WFFA ₁₂₀₀	WFFA ₃₆₀₀
$ \mathcal{F} = 784$	Error							
28,28-mnist-1,3	49.28	22.33	9.22	7.50	6.69	4.50	3.08	2.75
28,28-mnist-1,7	54.78	24.39	11.53	9.40	7.00	4.60	3.33	2.29
28,28-pneumoniarnist	62.88	31.46	8.17	7.74	5.67	4.85	3.75	3.08
	Kendall's Tau							
28,28-mnist-1,3	-0.80	0.42	0.49	0.64	0.70	0.81	0.86	0.88
28,28-mnist-1,7	-0.79	0.34	0.43	0.57	0.72	0.82	0.87	0.92
28,28-pneumoniarnist	-0.66	0.24	0.37	0.57	0.69	0.76	0.81	0.88
	RBO							
28,28-mnist-1,3	0.03	0.40	0.45	0.54	0.63	0.78	0.84	0.89
28,28-mnist-1,7	0.03	0.34	0.41	0.47	0.60	0.74	0.81	0.91
28,28-pneumoniarnist	0.03	0.23	0.30	0.35	0.43	0.59	0.65	0.81

Table 8: Just-in-time Defect Prediction comparison of WFFA versus LIME and SHAP.

Approach	openstack ($ \mathcal{F} = 13$)			qt ($ \mathcal{F} = 16$)		
	Error	kendalltau	rbo	Error	kendalltau	rbo
LIME	4.79	0.08	0.56	5.60	-0.07	0.45
SHAP	5.01	0.02	0.54	5.17	-0.11	0.44

SHAP require less than one second to process each image. The average results presented in Table 7 are consistent with those illustrated in Table 6 and the FFA results depicted in Table 2 and Table 3. Table 7 demonstrates that after only 10 seconds, our WFFA approximation outperforms both LIME and SHAP in terms of errors, Kendall's Tau, and RBO values. Additionally, after 10 seconds our approach produces weighted feature attributions, which is closer to WFFA₇₂₀₀ compared to both LIME and SHAP. This suggests that our approach effectively identifies the features that are genuinely relevant for the prediction, which is in stark contrast to LIME and SHAP.

C Application in Just-in-Time Defect Prediction

Modern software companies often engage in the rapid and frequent release of software products in short cycles. Because of the exponential growth of highly complex source code, such rapid-release software development presents significant challenges for under-resourced Software Quality Assurance (SQA) teams. Developers are unable to thoroughly ensure the highest quality of all newly developed code commits or pull requests within the limited time and resources available, due to the time-consuming and costly nature of various SQA activities, e.g. code review. To address this issue, a recent approach called Just-in-Time (JIT) defect prediction [30, 32, 38, 51] has been proposed. This approach aims to predict whether a commit will introduce software defects in the future such

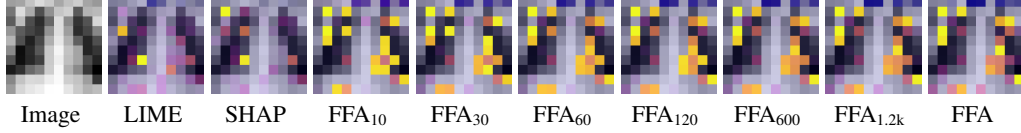


Figure 10: 10×10 PneumoniaMNIST. The prediction is pneumonia.

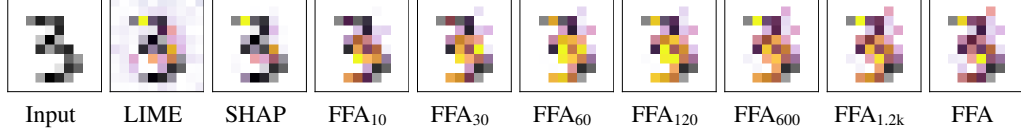


Figure 11: 10×10 MNIST 1 vs. 3. The prediction is 3.

391 that development teams can prioritize their limited SQA resources on the riskiest commits or pull
392 requests.

393 However, the JIT defect prediction approach has frequently been criticized for being opaque and
394 lacking explainability for practitioners. Model-agnostic explainability methods, e.g. LIME and SHAP,
395 cannot guarantee accurate feature attribution, as discussed earlier in this appendix and Section 5).
396 Experimental evidence presented in Section 5 demonstrates the usefulness of exact FFA in the
397 context of JIT defect prediction. Given that our earlier observations above suggest that exact (resp.
398 approximate) WFFA is consistent with exact (resp. approximate) FFA, we apply the computation of
399 WFFA in the setting of JIT defection prediction and demonstrate that it can be also a viable approach
400 to addressing practical explainability challenges.

401 In particular, where we use logistic regression models built on two widely-used large-scale open-
402 source datasets, namely Openstack and Qt, which are commonly used in JIT defect prediction
403 studies [52]. The property of monotonicity in logistic regression allows us to enumerate explanations
404 efficiently, following the approach of [44]. By leveraging this method, we can extract the *exact WFFA*
405 for each instance within one second. The comparison of WFFA, LIME, and SHAP in terms of the
406 three selected metrics is provided in Table 8. These results are consistent with the FFA assessment
407 presented in Table 4. Similar to the findings in Table 5, Table 6, and Table 7, both LIME and SHAP
408 misalign with weighted formal feature attribution, although there are some similarities between them.

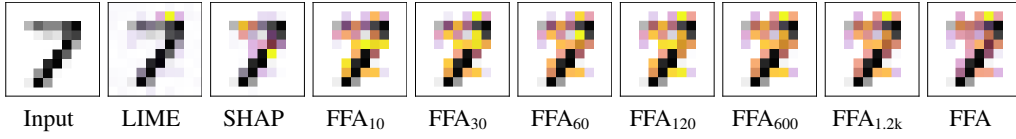


Figure 12: 10×10 MNIST 1 vs. 7. The prediction is 7.

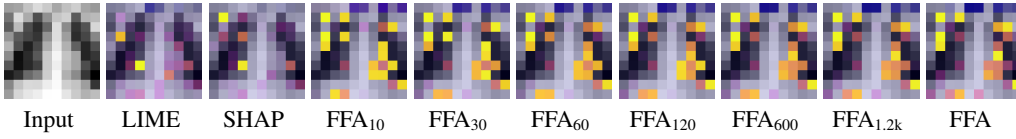


Figure 13: 10×10 PneumoniaMNIST. The prediction is pneumonia.

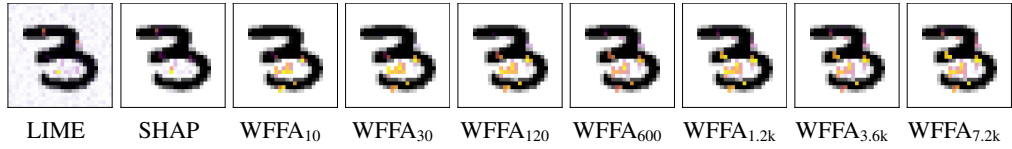


Figure 14: 28×28 MNIST 1 vs. 3. The prediction is digit 3.

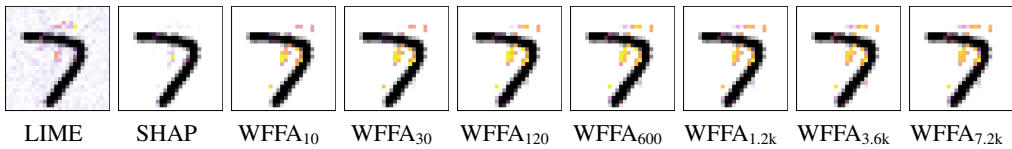


Figure 15: 28×28 MNIST 1 vs. 7. The prediction is digit 7.

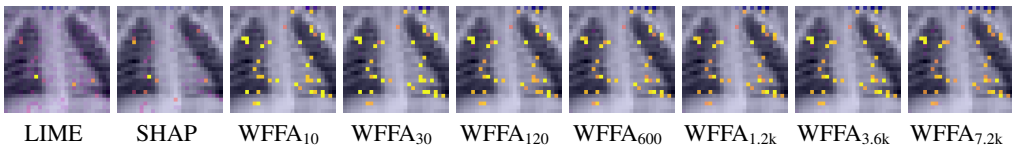


Figure 16: 28×28 PneumoniaMNIST. The prediction is normal.

References

- [1] ACM. Fathers of the deep learning revolution receive ACM A.M. Turing award. <http://tiny.cc/9plzpz>, 2018.
- [2] L. Amgoud and J. Ben-Naim. Axiomatic foundations of explainability. In L. D. Raedt, editor, *IJCAI*, pages 636–642, 2022.
- [3] J. Angwin, J. Larson, S. Mattu, and L. Kirchner. Machine bias. <http://tiny.cc/dd7mjz>, 2016.
- [4] M. Arenas, D. Baez, P. Barceló, J. Pérez, and B. Subercaseaux. Foundations of symbolic languages for model interpretability. In *NeurIPS*, 2021.
- [5] M. Arenas, P. Barceló, L. E. Bertossi, and M. Monet. The tractability of SHAP-score-based explanations for classification over deterministic and decomposable Boolean circuits. In *AAAI*, pages 6670–6678. AAAI Press, 2021.
- [6] M. Arenas, P. Barceló, L. E. Bertossi, and M. Monet. On the complexity of SHAP-score-based explanations: Tractability via knowledge compilation and non-approximability results. *CoRR*, abs/2104.08015, 2021.
- [7] M. Arenas, P. Barceló, M. A. R. Orth, and B. Subercaseaux. On computing probabilistic explanations for decision trees. In *NeurIPS*, 2022.
- [8] G. Audemard, F. Koriche, and P. Marquis. On tractable XAI queries based on compiled representations. In *KR*, pages 838–849, 2020.
- [9] G. Blanc, J. Lange, and L. Tan. Provably efficient, succinct, and precise explanations. In *NeurIPS*, 2021.
- [10] R. Boumazouza, F. C. Alili, B. Mazure, and K. Tabia. ASTERYX: A model-Agnostic SaT-basEd appRoach for sYmbolic and score-based eXplanations. In *CIKM*, pages 120–129, 2021.
- [11] L. Breiman. Random forests. *Mach. Learn.*, 45(1):5–32, 2001.
- [12] T. Chen and C. Guestrin. XGBoost: A scalable tree boosting system. In *KDD*, pages 785–794, 2016.
- [13] A. Darwiche and A. Hirth. On the reasons behind decisions. In *ECAI*, pages 712–720, 2020.
- [14] A. Darwiche and P. Marquis. On quantifying literals in Boolean logic and its applications to explainable AI. *J. Artif. Intell. Res.*, 72:285–328, 2021.
- [15] L. Deng. The MNIST database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012.
- [16] D. Dua and C. Graff. UCI machine learning repository, 2017. <http://archive.ics.uci.edu/ml>.
- [17] FairML. Auditing black-box predictive models. <http://tiny.cc/6e7mjz>, 2016.
- [18] J. Ferreira, M. de Sousa Ribeiro, R. Gonçalves, and J. Leite. Looking inside the black-box: Logic-based explanations for neural networks. In *KR*, page 432–442, 2022.
- [19] S. Friedler, C. Scheidegger, and S. Venkatasubramanian. On algorithmic fairness, discrimination and disparate impact. <http://fairness.haverford.edu/>, 2015.
- [20] N. Gorji and S. Rubin. Sufficient reasons for classifier decisions in the presence of domain constraints. In *AAAI*, pages 5660–5667, 2022.
- [21] X. Huang and J. Marques-Silva. The inadequacy of Shapley values for explainability. *CoRR*, abs/2302.08160, 2023.
- [22] X. Huang, M. C. Cooper, A. Morgado, J. Planes, and J. Marques-Silva. Feature necessity & relevancy in ML classifier explanations. In *TACAS (I)*, pages 167–186, 2023.

- [23] L. Hyafil and R. L. Rivest. Constructing optimal binary decision trees is NP-complete. *Inf. Process. Lett.*, 5(1):15–17, 1976. URL [https://doi.org/10.1016/0020-0190\(76\)90095-8](https://doi.org/10.1016/0020-0190(76)90095-8).
- [24] A. Ignatiev. Towards trustable explainable AI. In *IJCAI*, pages 5154–5158, 2020.
- [25] A. Ignatiev, N. Narodytska, and J. Marques-Silva. Abduction-based explanations for machine learning models. In *AAAI*, pages 1511–1519, 2019.
- [26] A. Ignatiev, N. Narodytska, N. Asher, and J. Marques-Silva. From contrastive to abductive explanations and back again. In *AI*IA*, pages 335–355, 2020.
- [27] A. Ignatiev, Y. Izza, P. J. Stuckey, and J. Marques-Silva. Using MaxSAT for efficient explanations of tree ensembles. In *AAAI*, pages 3776–3785, 2022.
- [28] Y. Izza, A. Ignatiev, and J. Marques-Silva. On tackling explanation redundancy in decision trees. *J. Artif. Intell. Res.*, 75:261–321, 2022. URL <https://doi.org/10.1613/jair.1.13575>.
- [29] M. I. Jordan and T. M. Mitchell. Machine learning: Trends, perspectives, and prospects. *Science*, 349(6245):255–260, 2015.
- [30] Y. Kamei, E. Shihab, B. Adams, A. E. Hassan, A. Mockus, A. Sinha, and N. Ubayashi. A Large-Scale Empirical Study of Just-In-Time Quality Assurance. *IEEE Transactions on Software Engineering (TSE)*, 39(6):757–773, 2013.
- [31] M. G. Kendall. A new measure of rank correlation. *Biometrika*, 30(1/2):81–93, 1938.
- [32] S. Kim, T. Zimmermann, E. J. Whitehead Jr, and A. Zeller. Predicting Faults from Cached History. In *ICSE*, pages 489–498, 2007.
- [33] R. Kohavi. Scaling up the accuracy of naive-Bayes classifiers: A decision-tree hybrid. In *KDD*, pages 202–207, 1996.
- [34] H. Lakkaraju and O. Bastani. "How do I fool you?": Manipulating user trust via misleading black box explanations. In *AIES*, pages 79–85, 2020.
- [35] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436, 2015.
- [36] M. H. Liffiton and A. Malik. Enumerating infeasibility: Finding multiple MUSes quickly. In *CPAIOR*, pages 160–175, 2013.
- [37] M. H. Liffiton, A. Previti, A. Malik, and J. Marques-Silva. Fast, flexible MUS enumeration. *Constraints An Int. J.*, 21(2):223–250, 2016.
- [38] D. Lin, C. Tantithamthavorn, and A. E. Hassan. The impact of data merging on the interpretation of cross-project just-in-time defect models. *IEEE Transactions on Software Engineering*, 2021.
- [39] Z. C. Lipton. The mythos of model interpretability. *Commun. ACM*, 61(10):36–43, 2018.
- [40] S. M. Lundberg and S. Lee. A unified approach to interpreting model predictions. In *NeurIPS*, pages 4765–4774, 2017.
- [41] E. L. Malfa, R. Michelmoro, A. M. Zbrzezny, N. Paoletti, and M. Kwiatkowska. On guaranteed optimal robust explanations for NLP models. In *IJCAI*, pages 2658–2665, 2021.
- [42] J. Marques-Silva and A. Ignatiev. Delivering trustworthy AI through formal XAI. In *AAAI*, pages 12342–12350. AAAI Press, 2022.
- [43] J. Marques-Silva, T. Gerspacher, M. C. Cooper, A. Ignatiev, and N. Narodytska. Explaining naive Bayes and other linear classifiers with polynomial time and delay. In *NeurIPS*, 2020.
- [44] J. Marques-Silva, T. Gerspacher, M. C. Cooper, A. Ignatiev, and N. Narodytska. Explanations for monotonic classifiers. In *ICML*, pages 7469–7479, 2021.
- [45] S. McIntosh and Y. Kamei. Are fix-inducing changes a moving target? A longitudinal case study of Just-in-Time defect prediction. *IEEE Transactions on Software Engineering (TSE)*, pages 412–428, 2017.

- [46] T. Miller. Explanation in artificial intelligence: Insights from the social sciences. *Artif. Intell.*, 267:1–38, 2019.
- [47] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529, 2015.
- [48] C. Molnar. *Interpretable Machine Learning*. Leanpub, 2020. <http://tiny.cc/6c76tz>.
- [49] R. S. Olson, W. G. L. Cava, P. Orzechowski, R. J. Urbanowicz, and J. H. Moore. PMLB: a large benchmark suite for machine learning evaluation and comparison. *BioData Min.*, 10(1):36:1–36:13, 2017.
- [50] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Z. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. PyTorch: An imperative style, high-performance deep learning library. In *NeurIPS*, pages 8024–8035, 2019.
- [51] C. Pornprasit and C. Tantithamthavorn. JITLine: A Simpler, Better, Faster, Finer-grained Just-In-Time Defect Prediction. In *MSR*, pages 369–379, 2021.
- [52] C. Pornprasit, C. Tantithamthavorn, J. Jiarapakdee, M. Fu, and P. Thongtanunam. PyExplainer: Explaining the predictions of Just-In-Time defect models. In *ASE*, pages 407–418, 2021.
- [53] A. Previti and J. Marques-Silva. Partial MUS enumeration. In *AAAI*. AAAI Press, 2013.
- [54] R. Reiter. A theory of diagnosis from first principles. *Artif. Intell.*, 32(1):57–95, 1987.
- [55] M. T. Ribeiro, S. Singh, and C. Guestrin. "Why should I trust you?": Explaining the predictions of any classifier. In *KDD*, pages 1135–1144, 2016.
- [56] M. T. Ribeiro, S. Singh, and C. Guestrin. Anchors: High-precision model-agnostic explanations. In *AAAI*, pages 1527–1535, 2018.
- [57] R. L. Rivest. Learning decision lists. *Mach. Learn.*, 2(3):229–246, 1987.
- [58] C. Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nat. Mach. Intell.*, 1(5):206–215, 2019.
- [59] P. Schmidt and A. D. Witte. Predicting recidivism in North Carolina, 1978 and 1980. *Inter-University Consortium for Political and Social Research*, 1988.
- [60] L. S. Shapley. A value of n -person games. *Contributions to the Theory of Games*, 2(28):307–317, 1953.
- [61] A. Shih, A. Choi, and A. Darwiche. A symbolic approach to explaining Bayesian network classifiers. In *IJCAI*, pages 5103–5111, 2018.
- [62] D. Slack, S. Hilgard, E. Jia, S. Singh, and H. Lakkaraju. Fooling LIME and SHAP: adversarial attacks on post hoc explanation methods. In *AIES*, pages 180–186, 2020.
- [63] D. Slack, A. Hilgard, S. Singh, and H. Lakkaraju. Reliable post hoc explanations: Modeling uncertainty in explainability. In *NeurIPS*, pages 9391–9404, 2021.
- [64] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. J. Goodfellow, and R. Fergus. Intriguing properties of neural networks. In *ICLR (Poster)*, 2014.
- [65] S. Wäldchen, J. MacDonald, S. Hauch, and G. Kutyniok. The computational complexity of understanding binary classifier decisions. *J. Artif. Intell. Res.*, 70:351–387, 2021.
- [66] W. Webber, A. Moffat, and J. Zobel. A similarity measure for indefinite rankings. *ACM Transactions on Information Systems (TOIS)*, 28(4):1–38, 2010.
- [67] J. Yang, R. Shi, D. Wei, Z. Liu, L. Zhao, B. Ke, H. Pfister, and B. Ni. MedMNIST v2-a large-scale lightweight benchmark for 2D and 3D biomedical image classification. *Scientific Data*, 10(1):41, 2023.