

When a CBR in Hand is Better than Twins in the Bush

Mobyen Uddin Ahmed¹, Shaibal Barua¹, Shahina Begum¹, Mir Riyanul Islam¹ and Rosina O Weber^{2,*}

¹Mälardalen University, Västerås, Sweden

²Drexel University, Philadelphia, PA, 19802, USA

Abstract

AI methods referred to as interpretable are often discredited as inaccurate by supporters of the existence of a trade-off between interpretability and accuracy. In many problem contexts however this trade-off does not hold. This paper discusses a regression problem context to predict flight take-off delays where the most accurate data regression model was trained via the XGBoost implementation of gradient boosted decision trees. While building an XGB-CBR Twin and converting the XGBoost feature importance into global weights in the CBR model, the resultant CBR model alone provides the most accurate local prediction, maintains the global importance to provide a global explanation of the model, and offers the most interpretable representation for local explanations. This resultant CBR model becomes a benchmark of accuracy and interpretability for this problem context, and hence it is used to evaluate the two additive feature attribute methods SHAP and LIME to explain the XGBoost regression model. The results with respect to *local accuracy* and *feature attribution* lead to potentially valuable future work.

Keywords

Accuracy, Interpretability, CBR, XGBoost, SHAP, LIME

1. Introduction

Case-based reasoning (CBR) is considered an interpretable model given its typical adoption of the weighted Euclidean Distance to implement k-nearest neighbors. With this approach, the weights are usually associated with global features, affording model interpretability. The concentration of the learning in global weights can however limit CBR accuracy, thus helping support the claim of the existence of a trade-off between accuracy and interpretability [1].

In explainable artificial intelligence (XAI), the trade-off between accuracy and interpretability has been debunked in different problem contexts with different data types. For example, using image data from mammograms, Barnett et al. [2] learned about deficiencies in their classifier when told by experts the classification was being done for the wrong reasons. When aligning

ICCBR XCBBR'22: 4th Workshop on XCBBR: Case-based Reasoning for the Explanation of Intelligent Systems at ICCBR-2022, September, 2022, Nancy, France

*Corresponding author. Authors are listed in alphabetical order.

✉ mobyen.uddin.ahmed@mdu.se (M. U. Ahmed); shaibal.barua@mdu.se (S. Barua); shahina.begum@mdu.se (S. Begum); mir.riyanul.islam@mdu.se (M. R. Islam); rosina@drexel.edu (R. O. Weber)

🌐 http://www.es.mdh.se/staff/149-Mobyen_Uddin_Ahmed (M. U. Ahmed);

http://www.es.mdh.se/staff/2754-Shaibal_Barua (S. Barua); <http://www.es.mdh.se/staff/146-Shahina> (S. Begum);

http://www.es.mdh.se/staff/3845-Mir_Riyanul_Islam (M. R. Islam); <https://www.cs.drexel.edu/~rw37/> (R. O. Weber)

🆔 0000-0003-1953-6086 (M. U. Ahmed); 0000-0002-7305-7169 (S. Barua); 0000-0002-1212-7637 (S. Begum);

0000-0003-0730-4405 (M. R. Islam); 0000-0001-7048-8812 (R. O. Weber)



© 2022 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings (CEUR-WS.org)

the interpretable features with domain knowledge, the resultant interpretable model was more accurate than before. The trade-off claim is even more often dismissed when data is tabular (e.g., [3]). Notwithstanding, as it often happens in science, this claim has motivated valuable works such as the ANN-CBR Twins [4] where an accurate artificial neural network (ANN) is twinned with CBR as a presumed less accurate but interpretable model. The successful demonstrations of ANN-CBR Twins (ibid.) make this a valuable approach for exemplar-based explainability.

This paper investigates the problem context of predicting flight delays. Air Traffic Flow Management (ATFM) costs, on average, approximately 100 Euros per minute for airlines [5]. According to the FAA report in 2019¹, the estimated cost due to delay, considering airlines, passengers, lost demand, and indirect costs, was thirty-three billion dollars. This high cost justifies the increased interest in predicting take-off time and delays [6].

The take-off time is one of the root indicators of the delay of an aircraft as it propagates to all transportation networks, hence predicting it is key to enhancing air traffic. Predicting the delay of take-off time is a regression problem, where feature sets (both numeric and categorical) are used from flight plans, weather reports, and airline information. Departure delay has been characterized considering the spatial and temporal aspects (e.g., [7, 8, 9, 10, 11, 12]). The methods used for predicting tasks in ATFM include neural networks (NN), random forest, gradient boosting machines, support vector machines, and linear regression [11].

This paper describes a study whose starting point was to use flight data to predict departure delays using XGBoost via regression. XGBoost [13] is an implementation of gradient boosted decision trees (GBDT), an ensemble method that uses gradients to build highly accurate decision trees. This ensemble aspect limits the local interpretability of GBDT but still produces global importance factors that can make the model globally interpretable. For local interpretability, an alternative would be to adapt the ANN-CBR twins approach into a XGB-CBR. One of the twins steps is to extract from the non-interpretable (and presumably more accurate) method the representation that supports its accuracy and transfer it over to CBR. The XGBoost importance factors facilitate this step. However, when doing this, as detailed later, the CBR model alone using XGBoost importance factors as global weights, produced a smaller mean absolute error (MAE) than the original XGBoost regression model.

The CBR model is more accurate (*i.e.*, lower MAE), offers global interpretability, and interpretable local explanations. This justifies its use as a benchmark against which to evaluate explanation methods for XGBoost. We adopt two additive feature attribute methods, namely, SHAP [14] and LIME [15] to produce features to explain the XGBoost regression model.

One of the benefits of having CBR as the most accurate model is interpretability. Another benefit stems from the use of global weights for each feature. One important aspect when predicting air traffic delays is that some features are clearly more important than others, making the opportunity to incorporate domain knowledge desirable. For example, the feature that represents delays on the previous leg of a flight that uses the same aircraft is certainly relevant. Having only one weight for each feature makes it easy to incorporate or manipulate this kind of domain knowledge by directly changing the weight value.

Section 2 introduces the methods and Section 3 describes this paper's methodology. Section 4 presents results and discussion, and Section 5 concludes.

¹https://www.faa.gov/data_research/aviation_data_statistics/media/cost_delay_estimates.pdf

2. Data and Explanation Models

This section describes the models discussed in this paper. The context is a regression model $r(x_i)$ that uses data where $x \in X$ are instances mapped by features $f_j \in F$, $f_j = 1, \dots, m$, $X_{\text{train}} \subset X$ are training instances x_i , $x_i = 1, \dots, n$ that include prediction delays $y \in Y$ in minutes, which are used by the regression model $r(x_i)$ to learn predictions \hat{y}_i . $X_{\text{test}} \subset X$ are testing instances.

2.1. Regression Models

XGBoost [13] is a GBDT ensemble method. Ensemble methods are shown to produce better performance than single methods [16]. GBDT is an ensemble method for decision trees that learns with differentiable loss functions [17]. Two GBDT variants are XGBoost [13] and LightGBM [18]. XGBoost uses the second-order gradient to improve accuracy whereas LightGBM aims at improved efficiency. Previous work in air traffic delay prediction has utilized LightGBM [6]. Hence, we start with XGBoost given its potential to be more accurate than LightGBM.

CBR [19] has its roots in memory-based methods from cognitive science [20]. CBR implements the similarity heuristic, *i.e.*, to reuse a previous solution to solve a similar new problem. Determining similarity between problems is domain-dependent, hence CBR systems often use the weighted Euclidean Distance where weights can reflect particular aspects of the problem context. These weights used in similarity assessment are global to features, making decisions interpretable at the global level². The limitation is that only global weights may limit accuracy. On the other hand, this simple and global representation facilitates incorporation of domain knowledge. When using the weighted Euclidean Distance, weights can be learned in various ways such as feedback learning algorithms [21] or decision trees (*e.g.*, [22]). In this paper, the CBR model uses the XGBoost feature importance values as weights.

2.2. Explanation Methods

ANN-CBR Twins is an example-based explanation method [23, 4]. The concept of Twins is based on the premise of two models where the accuracy-interpretability trade-off holds. The black-box and highly accurate ANN is one twin and the other is CBR, as the interpretable and less accurate model. The goal is that the models are functionally equivalent, that is, that they can produce the same results for the same testing instances. ANN-CBR twins succeed by transferring the representation and weights from the ANN into CBR [23].

2.2.1. Additive Feature Attribution

Explanation methods based on approaches to distribute gain in coalitional game theory [24, 14] utilize Shapley values [25] thus inheriting their properties. Lundberg and Lee [14] identify a class of explanation methods called *additive feature attribution*, which include those based on Shapley values, among others [14]. This class is referred to as additive because of the efficiency property from Shapley values [25] that shows that the gains shared by all players in a coalition game equals the value of the grand coalition. This property becomes *local accuracy* for additive

²Authors note that it is not within the scope of this paper to debate about the value of local versus global interpretability, but simply to point out when discussed interpretability is local or global.

feature attribute methods (Equation 1) where $g(\mathbf{z}_j)$ is the explanation model where the property of *local accuracy* is demonstrated when $g(\mathbf{z}_j)$ matches the model $r(x_i)$ for each instance, where $g(\mathbf{z}_j)$ is computed on the vector \mathbf{z}_j which transforms \mathbf{x}_j by the function $h(\mathbf{z}_j)$ making $\mathbf{z}_j \in \{0, 1\}^m$:

$$g(\mathbf{z}_j) = \phi_0 + \sum_{j=1}^{j=m} \phi_j \mathbf{z}_j \quad (1)$$

The local interpretable model-agnostic explanation (LIME) [15] is another additive feature attribution method. LIME fits a linear regression to explain the behavior of a sample point. To obtain points for fitting a linear regression, LIME randomly perturbs the point to be explained using the points closest to the target point. The coefficients of the linear regression in LIME are used to produce ϕ values for Equation 1 and predict the output of the model $g(\mathbf{z}_j)$.

2.2.2. XAI for Regression

Letzger et al. [26] examine XAI methods for regression problems. They recommend that both prediction and explanation be done with methods that do not normalize their values in order to preserve the alignment between the sum of the contributions with the prediction thus preserving the same measurement unit. They refer to it as the *conservation principle*.

3. Methodology

This section analyzes SHAP and LIME in terms of *local accuracy* and feature attribution for the XGBoost implementation for predicting flight delays. XGBoost predictions are the baseline for *local accuracy* because the explanation models were built for it; CBR is the baseline for feature importance because it is the most accurate model and it allows local interpretability.

3.1. Data

The dataset was collected and processed by EUROCONTROL³ and it uses the Enhanced Tactical Flow Management System (ETFMS) flight data messages for all flights during the year 2019 (*i.e.*, May to October). The datasets include basic information, status of the flight and previous flight leg, ATFM regulations, weather, and calendar. The features are described in detail in [6].

The data used for XGBoost includes 5,903,743 instances of the clean dataset with months from May to August, which is a subset of the dataset from EUROCONTROL. The study includes the first five days of September and October for testing, without using the remaining days of these months. The number of instances in the testing data is 158,147. The main difference between the data used in this paper and in [6] is that they broke down the data into eight intervals of time to EOBT (Estimated Off-Block Time). In this paper, the data was not broken down in intervals, which means using the interval from zero to three-hundred and sixty minutes: (0,360].

³<https://www.eurocontrol.int/>

3.2. Metrics

We use MAE and standard deviation σ for the quality of the predictions for both data and explanation models. MAE computes the average difference between an actual observation and a prediction from a model:

$$MAE = 1/n \sum_{i=1}^{i=n} |y_i - \hat{y}_i| \quad (2)$$

MAE for Data Models. MAE is computed based on the actual delays y_i from the testing data as baseline for comparison against the predictions \hat{y}_i learned by the regression models $r(x_i)$.

MAE for Explanation Models. As described in Section 2.2.1, both SHAP and LIME use a function $g(\mathbf{z}_j)$ to produce a prediction $\hat{y} \in Y$ using Equation 1. The values for MAE for the two explanation models are obtained from the difference between the predictions \hat{y}_i learned by the regression model $r(x_i)$ and the \hat{y} obtained by $g(\mathbf{z}_j)$.

Normalized Discounted Cumulative Gain (nDCG). nDCG compares the order of retrieved documents in information retrieval. Studies [27, 28] show that different libraries can produce varied results. In this paper, we computed nDCG with sklearn library [29].

3.3. Methods

XGBoost The hyperparameters for XGBoost were selected based on the results from 288 different combinations. The final model used the following: learning_rate = 0.1, max_depth = 7, min_child_weight = 1, subsample = 0.5, colsample_bytree = 0.5, n_estimators = 500.

CBR The CBR model averages the predictions in the three least distant neighbors retrieved using the Euclidean Distance weighted with the XGBoost importance factors. For binary and categorical features, local similarity is symbolic producing 1 when values are equal and 0 when different. For numeric features, the absolute difference is divided by the range of values. As a local learner, the predictions are computed with leave-one-out cross validation.

Additive and Global CBR CBR can be used for example-based explanations, but its global weights do not support local explanations in the same form as additive models. CBR global weights support global interpretability, which we refer as Global CBR. Additive CBR is an additive version built by re-scaling the values for the CBR regression model after prediction. Additive CBR becomes a benchmark for local interpretability. Feature values and weights are re-scaled to produce $\phi_j \mathbf{z}_j^{CBR}$ in the same terms as the additive feature attribution explanation models $g(\mathbf{z}_j)$. To achieve this, we utilize a multiplier γ_i obtained by dividing the prediction \hat{y}_i of the CBR regression model $r(x_i)$ by the sum of its factors $x_{ij}w_j$:

$$\gamma_i = \hat{y}_i / \left(\sum_{i=1}^{i=n} \sum_{j=1}^{j=m} x_{ij}w_j \right) \quad (3)$$

SHAP and LIME The two explanation models were built for XGBoost. SHAP was implemented using kernelSHAP with default settings. LIME was implemented with 1,000 perturbations and 1,000 number of samples.

4. Results and Discussion

4.1. Results from Data Models

Table 1

Average MAE and standard deviation σ for data models on three sets of the data. *All* uses the 158,147 instances, *100k* and *67k* use, respectively, the 100,650 and 67,495 most accurate testing instances.

Model Instances	CBR			XGB		
	All	100k	67k	All	100k	67k
Average MAE	5.88	1.48	0.52	9.22	4.28	2.72
σ MAE	8.22	1.55	0.70	8.91	2.67	1.62

The first results in this section demonstrate the basis for using CBR as the baseline and justify why the XGB-CBR Twin is not required. The results for MAE and standard deviation for the XGBoost and CBR data models are shown in Table 1.

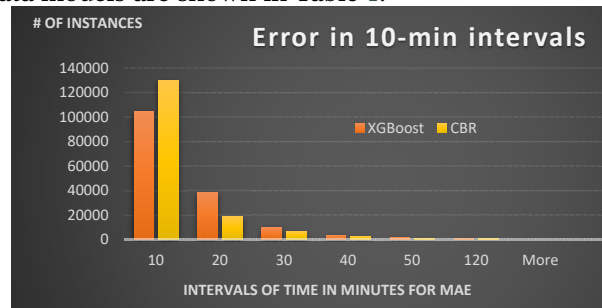


Figure 1: Histograms grouping the number of instances within bins of MAE for CBR and XGBoost

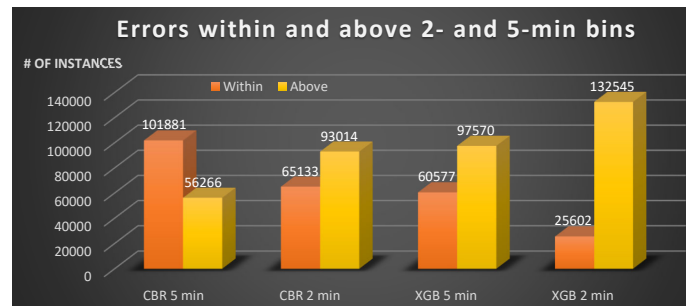


Figure 2: Instances with errors within and above 2 and 5 minutes

Because CBR performance challenges the accuracy versus interpretability trade-off [1], this section provides details for the average results from Table 1. The first results present how far the regression predictions are from the actual delays. The histograms in Figure 1 show that the

lower error in the CBR model is based on the model having more instances with lower errors. These lower errors were within the 10-minute range. The XGBoost model has fewer errors in the bin of 10 minutes and more in the bins with higher errors, leading to greater values in MAE.

Figure 2 depicts the number of instances at the five- and two-minute marks for CBR and XGBoost. At these thresholds, the CBR model produces more instances within five minutes difference from the actual predictions than with higher errors. At the two-minute mark, CBR has about 40% of instances within two minutes away from the actual prediction.

4.2. Discussion on Data Models

The higher CBR accuracy incites the question as to whether CBR models would consistently benefit from learning global weights via ensemble models. These results allow the use of CBR as a baseline for explanation quality because it is both the most interpretable and most accurate. This would represent one circumstance in which it would not be necessary to adopt the Twins approach. Had the CBR model not been the most accurate, using Twins would be preferable.

4.3. Results from Explanation Models

Table 2

Average MAE and standard deviation σ for *local accuracy* of explanation models on three sets of instances.

# of instances Model	ALL		100k		67k	
	SHAP	LIME	SHAP	LIME	SHAP	LIME
average MAE	3.3×10^{-6}	8.62	1.1×10^{-6}	4.75	6.2×10^{-7}	3.13
σ MAE	4.3×10^{-6}	6.32	7.9×10^{-7}	2.81	4.0×10^{-7}	1.82

Table 3

Pairwise comparison of nDCG for SHAP and LIME across the three sets against the two baselines Global and Additive CBR for all 42 features. Higher is better. Highest value is in bold.

Baseline # of instances	Global CBR			Additive CBR		
	All	100k	67k	All	100k	67k
SHAP	0.74	0.72	0.72	0.81	0.81	0.82
LIME	0.88	0.85	0.80	0.82	0.81	0.77

Table 2 includes average and standard deviation MAE of *local accuracy* for SHAP and LIME with respect to XGBoost, the model for which the local explanations were built. Table 3 presents the nDCG values comparing the order of feature attributions. Table 4 combines the two previous tables to show the progression of values. We observe that Tables 3 and 4 include comparisons against Global CBR for reference purposes, but the intended benchmark for analysis is Additive CBR because it is formulated as an additive model.

4.4. Discussion on Explanation Models

Table 2 shows the impressive *local accuracy* obtained by SHAP. These MAE values correspond to precision levels of 10^{-5} and 10^{-6} . LIME produces average error still above three minutes in

the smallest set of instances with lowest MAE. This difference is not observed in the analysis of feature attributions in Table 3. This difference may be explained by a few aspects.

At the smallest set with the most accurate instances (Table 3), SHAP's attributions provide higher nDCG values (*i.e.*, 0.82) than LIME (*i.e.*, 0.77). This result is not as impressive as results for *local accuracy* but shows SHAP as superior. The fact that SHAP does not have higher nDCG values may be because local explanations are built to model the data model, which is XGBoost, not CBR. As it can be seen in Table 1, there is reasonable difference between the MAE of CBR (*i.e.*, 0.52) and XGBoost (*i.e.*, 2.72) with respect to the actual data at the set of instances with lowest MAE. This variation might explain why the nDCG values for SHAP are not higher.

Table 4

Progression of both *local accuracy* in MAE and nDCG for SHAP and LIME. nDCG(GI) is compared against Global CBR, nDCG(Add) is compared against Additive CBR. Rows show the number of instances. For *local accuracy* in MAE, lower is better. For nDCG, higher is better. Best value is in bold.

XAI Model Metric	SHAP			LIME		
	<i>local accuracy</i>	nDCG(Add)	nDCG(GI)	<i>local accuracy</i>	nDCG(Add)	nDCG(GI)
All	3.3×10^{-6}	0.806	0.806	8.62	0.819	0.882
100k	1.1×10^{-6}	0.813	0.722	4.75	0.805	0.847
67k	6.2×10^{-7}	0.817	0.717	3.13	0.773	0.800

Table 4 includes the values for *local accuracy* for easy examination of their progression. Moving from the data set with all instances, which is expected to be the least accurate, SHAP's *local accuracy* improves going from the first (*i.e.*, 0.806), to second (*i.e.*, 0.813), and third row (*i.e.*, 0.817), showing *local accuracy* and nDCG are somehow proportional. nDCG values for LIME are inversely proportional, decreasing from the first (*i.e.*, 0.819), to second (*i.e.*, 0.805), and third row (*i.e.*, 0.773). One possible observation is that LIME's low *local accuracy* is consistent with lack of progression of nDCG. In any case, these results suggest further studies are needed because they do not provide the means to support that any of these feature attributions is valid. We list three aspects to investigate: feature attribution, *local accuracy*, and additive variants.

Feature Importance and Feature Attribution The literature indicates that the contribution of a feature in an additive feature attribution model is different from feature importance in the sense of weights [26]. The question arises on whether there are any relations to be drawn between these two types of feature importance. One direction would be to question whether example-based explanations produced by CBR support feature attributions resultant from any explanation model. Another would be on whether there is any relationship between feature importance in the sense of weights as practiced in CBR and feature attributions based on contributions of an additive model. Further studies are needed to shed light into the claim that the "*best explanation of a simple model is the model itself*" [14] pp 2. If the explanation method models the data model decision boundary then what information content does it produce? If the explanation method models instance points, then what does it mean for a feature to contribute to a decision? Questions such as how to precisely define feature attributions and feature importance are crucial to support proper presentation of XAI results to users.

Local Accuracy The recommendation is to investigate whether *local accuracy* is an indicator of feature attribution quality. This study, of course, depends on a definition of feature attribution.

Additive Variants It is not clear if the Additive CBR model adopted herein meets the *conservation principle* [26] for regression and is thus valid as a benchmark. A review of the literature should clarify which data models can be re-scaled into additive models to enable valid comparisons.

5. Related Works

Many papers have attempted to evaluate and comparatively analyze explanation methods (e.g., [30, 31, 32]). There are multiple ways to categorize explainable methods, but a valuable, and often dismissed, perspective is to consider the information type an XAI method produces. Methods that reply to the question, “Why not something else?” produce counterfactual instances and cannot be included in the same category as feature attribution methods, which aim to produce contributions of instance features. This paper compares two XAI methods that belong to the category of *additive feature attribution methods* [14], namely, SHAP and LIME.

Zhou et al. [32] point out the fact that *attribution* is not a well defined term as they compare additive (e.g., SHAP [14]) against non-additive methods such as (e.g., [33, 34]). Their rationale for the selection is that all these methods can be used to produce visualizations known as saliency maps. Zhou et al. [32] propose to transform datasets as a means to create ground-truth data and assess whether these methods can succeed in recovering them. The authors conclude none of the methods can be considered satisfactory.

The benefit of limiting the set of methods to evaluate lies on the ability to compare along the same deliverable. *Additive feature attribution methods* [14] share the same properties and thus using the features they identify with highest importance and their *local accuracy* seem a reasonable starting point. As recommended by various authors (e.g., [35, 36, 37, 38]) the use of benchmark datasets is valid as long as the evaluation is limited to feature importance or *local accuracy*. As previously described [39], benchmark datasets are not recommended for evaluating explanations for user consumption because explanations are user-, context-, and application-specific (e.g., [1, 40, 41]).

6. Concluding Remarks and Future Works

This paper describes a regression problem for air traffic delay prediction where an interpretable data model is also the most accurate, hence demonstrating another instance where the accuracy-interpretability trade-off does not hold. Here the study built a reasonably accurate model (i.e., MAE 9.22) with XGBoost and wanted to have a more interpretable model by building an XGB-CBR twin. When transferring the importance factors from XGBoost into CBR as global weights, the CBR model turned out to be even more accurate (i.e., MAE 5.82) than the XGBoost. The study then used the interpretable CBR model as a benchmark to compare the performance of the two additive feature attribution methods SHAP and LIME. The selection of these two methods was based on their *local accuracy* property where each explanation model is able to

produce a prediction just like the regression model. When examining *local accuracy*, the SHAP explanation model was functionally equivalent to the original XGBoost model, predicting the same delays at a precision of 10^{-5} . The MAE between LIME and XGBoost is 8.62 minutes.

Based on the assertion that the best explanation of a model is the model itself [14], the results compare whether the level of equivalence between the data and the explanation models could translate into feature attribution quality. Nonetheless, when comparing the feature importance from the Additive CBR baseline against feature attributions from LIME and SHAP, SHAP's superior performance in *local accuracy* is not matched. Based on these results, a few questions arise with potential to advance the field (Section 4).

Among important future work are studies on learning feature weights for CBR and identifying when Twins are preferable. For predicting flight delays, future work includes comparisons against the results from [6], other data models, and other additive feature attribution methods.

Acknowledgments

This work was financed by the project xApp: Explainable AI for Industrial Applications which is funded by VINNOVA (Sweden's innovation agency) and Mälardalen University. The authors would like to acknowledge VINNOVA to support our research in the area of explainable AI (XAI) in the project "Application of Explainable AI in Industrial Applications (Diary number 2021-03971)". This study is also a part of the project Transparent Artificial Intelligence and Automation to Air Traffic Management Systems, ARTIMATION, funded by the European Union's Horizon 2020 within the framework SESAR 2020 research and innovation program under Grant Agreement N. 894238. The authors would like to thank Eurocontrol for providing the dataset for this study, and the reviewers for their excellent suggestions.

References

- [1] D. Gunning, M. Stefik, J. Choi, T. Miller, S. Stumpf, G.-Z. Yang, Xai—explainable artificial intelligence, *Science robotics* 4 (2019).
- [2] A. J. Barnett, F. R. Schwartz, C. Tao, C. Chen, Y. Ren, J. Y. Lo, C. Rudin, A case-based interpretable deep learning model for classification of mass lesions in digital mammography, *Nature Machine Intelligence* 3 (2021) 1061–1070.
- [3] J. Liu, C. Zhong, M. Seltzer, C. Rudin, Fast sparse classification for generalized linear and additive models, *Proceedings of machine learning research* 151 (2022) 9304.
- [4] E. M. Kenny, M. T. Keane, Explaining deep learning using examples: Optimal feature weighting methods for twin systems using post-hoc, explanation-by-example in xai, *Knowledge-Based Systems* 233 (2021) 107530.
- [5] A. J. Cook, G. Tanner, European airline delay cost reference values (2011).
- [6] R. Dalmau, F. Ballerini, H. Naessens, S. Belkoura, S. Wangnick, An explainable machine learning approach to improve take-off time predictions, *Journal of Air Transport Management* 95 (2021) 102090.
- [7] J. J. Rebollo, H. Balakrishnan, Characterization and prediction of air traffic delays, *Transportation research part C: Emerging technologies* 44 (2014) 231–241.

- [8] Y. J. Kim, S. Choi, S. Briceno, D. Mavris, A deep learning approach to flight delay prediction, in: IEEE/AIAA 35th Digital Avionics Systems Conference (DASC), IEEE, 2016, pp. 1–6.
- [9] B. Yu, Z. Guo, S. Asian, H. Wang, G. Chen, Flight delay prediction for commercial air transport: A deep learning approach, *Transportation Research Part E: Logistics and Transportation Review* 125 (2019) 203–221.
- [10] T.-N. Tran, D.-T. Pham, S. Alam, V. Duong, Taxi-speed prediction by spatio-temporal graph-based trajectory representation and its applications, *Proceedings of the ICRAT (2020)*.
- [11] S. Kovarik, L. Doherty, K. Korah, B. Mulligan, G. Rasool, Y. Mehta, P. Bhavsar, M. Paglione, Comparative analysis of machine learning and statistical methods for aircraft phase of flight prediction, in: *International Conference on Research in Air Transportation 2020, 9th International Conference*, 2020.
- [12] R. Dalmau Codina, S. Belkoura, H. Naessens, F. Ballerini, S. Wagnick, Improving the predictability of take-off times with machine learning: A case study for the maastricht upper area control centre area of responsibility, in: *Proceedings of the 9th SESAR Innovation Days*, 2019, pp. 1–8.
- [13] T. Chen, C. Guestrin, Xgboost: A scalable tree boosting system, in: *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 2016, pp. 785–794.
- [14] S. M. Lundberg, S.-I. Lee, A unified approach to interpreting model predictions, *Advances in neural information processing systems* 30 (2017).
- [15] M. T. Ribeiro, S. Singh, C. Guestrin, "why should i trust you?" explaining the predictions of any classifier, in: *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 2016, pp. 1135–1144.
- [16] O. Sagi, L. Rokach, Ensemble learning: A survey, *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 8 (2018) e1249.
- [17] Z. Zhang, C. Jung, Gbdt-mo: gradient-boosted decision trees for multiple outputs, *IEEE transactions on neural networks and learning systems* 32 (2020) 3156–3167.
- [18] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, T.-Y. Liu, Lightgbm: A highly efficient gradient boosting decision tree, *Advances in neural information processing systems* 30 (2017).
- [19] M. M. Richter, R. O. Weber, *Case-based reasoning: a textbook*, Springer, 2013.
- [20] R. C. Schank, *Dynamic memory: A theory of reminding and learning in computers and people*, cambridge university press, 1983.
- [21] D. W. Aha, *Feature Weighting for Lazy Learning Algorithms*, Springer US, Boston, MA, 1998, pp. 13–32. doi:10.1007/978-1-4615-5725-8_2.
- [22] D. Doğan, S. Z. and Arditi, H. M. Günaydn, Using decision trees for determining attribute weights in a case-based model of early cost prediction, *Journal of Construction Engineering and Management-Asce* (2008).
- [23] E. M. Kenny, M. T. Keane, Twin-systems to explain artificial neural networks using case-based reasoning: Comparative tests of feature-weighting methods in ann-cbr twins for xai, in: *Twenty-Eighth International Joint Conferences on Artificial Intelligence (IJCAI)*, Macao, 10-16 August 2019, 2019, pp. 2708–2715.
- [24] E. Strumbelj, I. Kononenko, An efficient explanation of individual classifications using

- game theory, *The Journal of Machine Learning Research* 11 (2010) 1–18.
- [25] L. S. Shapley, A value for n-person games, *Classics in game theory* 69 (1997).
 - [26] S. Letzgus, P. Wagner, J. Lederer, W. Samek, K.-R. Müller, G. Montavon, Toward explainable artificial intelligence for regression models: A methodological perspective, *IEEE Signal Processing Magazine* 39 (2022) 40–58.
 - [27] R. Busa-Fekete, G. Szarvas, T. Elteto, B. Kégl, An apple-to-apple comparison of learning-to-rank algorithms in terms of normalized discounted cumulative gain, in: *ECAI 2012-20th European Conference on Artificial Intelligence: Preference Learning: Problems and Applications in AI Workshop*, volume 242, Ios Press, 2012.
 - [28] Y. Wang, L. Wang, Y. Li, D. He, W. Chen, T.-Y. Liu, A theoretical analysis of ndcg ranking measures, in: *Proceedings of the 26th annual conference on learning theory (COLT 2013)*, volume 8, 2013, p. 6.
 - [29] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine learning in Python, *Journal of Machine Learning Research* 12 (2011) 2825–2830.
 - [30] J. Adebayo, J. Gilmer, M. Muelly, I. Goodfellow, M. Hardt, B. Kim, Sanity checks for saliency maps, in: *Advances in Neural Information Processing Systems*, 2018, pp. 9505–9515.
 - [31] X. Man, E. Chan, The best way to select features?, *arXiv preprint arXiv:2005.12483* (2020).
 - [32] Y. Zhou, S. Booth, M. T. Ribeiro, J. Shah, Do feature attribution methods correctly attribute features, *arXiv preprint arXiv:2104.14403* (2021).
 - [33] R. R. Selvaraju, A. Das, R. Vedantam, M. Cogswell, D. Parikh, D. Batra, Grad-cam: Why did you say that? visual explanations from deep networks via gradient-based localization, *CoRR abs/1610.02391* (2016). URL: <http://arxiv.org/abs/1610.02391>. *arXiv: 1610. 02391*.
 - [34] D. Smilkov, N. Thorat, B. Kim, F. B. Viégas, M. Wattenberg, Smoothgrad: removing noise by adding noise, *CoRR abs/1706.03825* (2017). *arXiv: 1706. 03825*.
 - [35] M. Sundararajan, A. Taly, Q. Yan, Axiomatic attribution for deep networks, in: *International conference on machine learning*, PMLR, 2017, pp. 3319–3328.
 - [36] M. Yang, B. Kim, Benchmarking attribution methods with relative feature importance, *arXiv preprint arXiv:1907.09701* (2019).
 - [37] J. DeYoung, S. Jain, N. F. Rajani, E. Lehman, C. Xiong, R. Socher, B. C. Wallace, Eraser: A benchmark to evaluate rationalized nlp models, *arXiv preprint arXiv:1911.03429* (2019).
 - [38] S. S. Amiri, R. O. Weber, P. Goel, O. Brooks, A. Gandley, B. Kitchell, A. Zehm, Data representing ground-truth explanations to evaluate xai methods, *arXiv preprint arXiv:2011.09892* (2020).
 - [39] F. Yang, M. Du, X. Hu, Evaluating explanation without ground truth in interpretable machine learning, *arXiv preprint arXiv:1907.06831* (2019).
 - [40] A. B. Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. García, S. Gil-López, D. Molina, R. Benjamins, et al., Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai, *Information fusion* 58 (2020) 82–115.
 - [41] S. T. Mueller, R. R. Hoffman, W. Clancey, A. Emrey, G. Klein, Explanation in human-ai systems: A literature meta-review, synopsis of key ideas and publications, and bibliography for explainable ai, *arXiv preprint arXiv:1902.01876* (2019).