Concept Drift Detection in Machine Learning Systems by Exploiting Relaxed Functional Dependencies

Loredana Caruccio[1,†], Stefano Cirillo[1,†], Giuseppe Polese[1,†] and Roberto Stanzione[1,*,†]

[1]University of Salerno, via Giovanni Paolo II, 132, Fisciano (SA), 84084, Italy

Abstract

Although to train predictive models Machine Learning approaches completely rely on data, the latter can dynamically evolve over time. This could make predictive models outdated due to the presence of possible data shifts, with a consequent decrease in prediction accuracy. Concept drift detection techniques aim to detect such shifts in order to adopt countermeasures and maintain predictive performance over time. To this end, drift detection methods aim to monitor data distribution shifts, trying to identify changes without evaluating model predictions. In this discussion paper, we present a profiling metadata-driven approach for quantifying concept drift. Specifically, we focus on Relaxed Functional Dependencies (RFDs) and formalize the relationship between changes in metadata and performance trends of the predictive models over time. Moreover, we define a suite of RFD-based metrics measuring the distance between two sets of data. To evaluate the proposed approach, we compared it with other distribution-based metrics on datasets with both known and unknown drift. Results proved that the proposed metrics are strongly correlated with the model's performance according to their trends. Moreover, the defined suite of metrics is also able to capture concept drift more effectively than traditional distribution-based approaches.

Keywords

Data Profiling, Relaxed Functional Dependencies, Concept Drift

1. Introduction

Machine Learning (ML) models are increasingly relied upon for a multitude of tasks, including critical ones such as anomaly detection [1, 2, 3, 4], where inaccurate predictions can lead to potentially severe consequences. After deployment, ML models may initially exhibit robust performance, but, as time progresses, the underlying assumptions may no longer hold, leading to wrong predictions. The main reason for model degradation is *concept drift*, a phenomenon that refers to changes in the underlying function that generates data. Aiming at detecting such shifts, several methods monitor the model's prediction performance, while others analyze how data distribution changes. Some of them rely on qualitative descriptors like "abrupt" and "gradual", which have been shown to have limitations due to their dependence on arbitrary boundaries [5], leading to the necessity of estimating the *drift magnitude* by means of quantitative measures. However, while data distribution-based approaches have the advantage of not requiring an

SEBD 2025: 33rd Symposium on Advanced Database System, June 16-19, 2025 - Ischia, Italy

[*] Corresponding author.

[†] These authors contributed equally.

lcaruccio@unisa.it (L. Caruccio); scirillo@unisa.it (S. Cirillo); gpolese@unisa.it (G. Polese); rstanzione@unisa.it (R. Stanzione)

0000-0002-2418-1606 (L. Caruccio); 0000-0003-0201-2753 (S. Cirillo); 0000-0002-8496-2658 (G. Polese); 0000-0001-9212-821X (R. Stanzione)

analysis of model predictions, they are more prone to false positives [6]. Moreover, existing approaches can only capture changes in the individual attribute distributions. Thus, new strategies leveraging new types of properties in the data should be investigated. To this end, valuable properties could be extracted through *Data Profiling* techniques, which enable the discovery of a wide variety of metadata [7], including Relaxed Functional Dependencies (RFDs). This discussion paper presents the concept drift detection approach proposed in [8], which analyzes the change of RFDs to quantify data shifts in supervised ML settings. Specifically, we defined a suite of RFD-based metrics to quantify the divergence between the training data and a set of new samples that the model has to process. Moreover, we provided other RFD-based metrics inspired by ML measures, with the aim of capturing the performance trend of the monitored model. We evaluated the proposed metrics on datasets with *Known* and *Unknown* drift, studying how their trend is correlated to the performance of the model over time. A strong correlation would prove that analyzing RFD evolution can provide meaningful insights about concept drift without evaluating the model predictions. We also compared the proposed metrics with existing distribution-based measures.

2. RFD$_c$s and Concept Drift

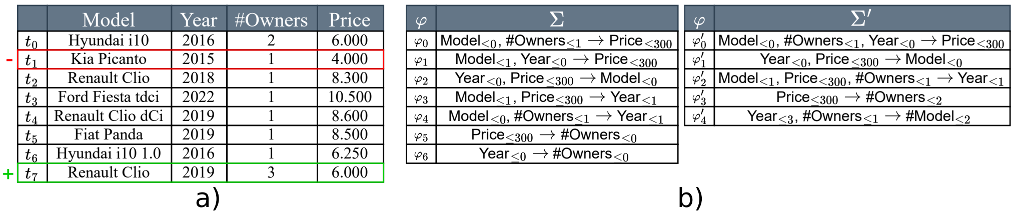
Profiling Metadata. Functional Dependencies (FDs) describe relationships among two sets of attributes X and Y. Formally, an FD $X \rightarrow Y$ (X *implies* Y) is satisfied if and only if for every pair of tuples (t_1, t_2), whenever $(t_1[X] = t_2[X])$, then $(t_1[Y] = t_2[Y])$. The attribute set $X = X_1, X_2, \ldots, X_h$ represents the Left Hand Side (LHS) of the FD, whereas the set $Y = Y_1, Y_2, \ldots, Y_k$ is the Right Hand Side (RHS).

The definition of FD has been recently extended to address challenges associated with inaccurate real-world data, leading to Relaxed Functional Dependencies (RFDs). The latter admit a limited number of violations (RFDs relaxing on the *extent*, namely RFD$_e$s) and/or the usage of similarity/distance functions as matching operators (RFDs relaxing on the *attribute comparison*, namely RFD$_c$s). In this paper, we leverage RFD$_c$s only. Formally, given an instance r of a relation schema R, a *constraint* ϕ, over an attribute $A \in$ *attr(R)*, is a predicate $\delta(t_i[A], t_j[A])\theta_k\varepsilon$, where δ is a similarity (or distance) function, θ_k a comparison operator, and ε a threshold. A specific similarity/distance function is applied according to the nature of the attributes.

Definition 1. (RFD$_c$). Given a relation schema R, an RFD$_c$ φ is denoted as $X_{\Phi_1} \rightarrow Y_{\Phi_2}$, where: $X = X_1, X_2, ..., X_h$ and $Y = Y_1, Y_2, ..., Y_k$, with $X, Y \subseteq attr(R)$ and $X \cap Y = \emptyset$; and $\Phi_1 = \bigwedge_{X_i \in X} \phi_i[X_i](\Phi_2 = \bigwedge_{Y_j \in Y} \phi_j[Y_j], resp.)$, with $\phi_i(\phi_j, resp.)$ a conjunction of similarity/distance constraints on $X_i(Y_j, resp.)$ and $i = 1, \ldots, h$ $(j = 1, \ldots, k, resp.)$. Thus, given an instance *r* of *R*, we can state that *r* satisfies the RFD$_c$ φ (i.e., *r* $\models \varphi$) if and only if for every pair of tuples $(t_1, t_2) \in r$, if Φ_1 is true then also Φ_2 returns true.

For the sake of simplicity and without loss of generality, in what follows, we consider RFD$_c$s with a single attribute on the RHS, i.e., $X_{\Phi_1} \rightarrow A_{\phi_2}$. Moreover, we will refer to constraints defined through distance functions, with a comparison operator ($\leq$) and the associated threshold.

For example, consider the tuples ranging from t_0 to t_6 of the dataset snippet shown in Figure 1a, then an example of holding RFD$_c$ is: φ:Model$_{\leq 4}$, Year$_{\leq 1}$ $\rightarrow$ Price$_{\leq 300}$, denoting that whenever two tuples have similar values on Model and Year, then they have a similar Price.



		Model	Year	#Owners	Price
	t_0	Hyundai i10	2016	2	6.000
-	t_1	Kia Picanto	2015	1	4.000
	t_2	Renault Clio	2018	1	8.300
	t_3	Ford Fiesta tdci	2022	1	10.500
	t_4	Renault Clio dCi	2019	1	8.600
	t_5	Fiat Panda	2019	1	8.500
	t_6	Hyundai i10 1.0	2016	1	6.250
+	t_7	Renault Clio	2019	3	6.000

a)

φ	Σ
φ_0	$\text{Model}_{<0}, \#\text{Owners}_{\leq 1} \rightarrow \text{Price}_{<300}$
φ_1	$\text{Model}_{<1}, \text{Year}_{\leq 0} \rightarrow \text{Price}_{<300}$
φ_2	$\text{Year}_{<0}, \text{Price}_{\leq 300} \rightarrow \text{Model}_{<0}$
φ_3	$\text{Model}_{<1}, \text{Price}_{\leq 300} \rightarrow \text{Year}_{<1}$
φ_4	$\text{Model}_{<0}, \#\text{Owners}_{\leq 1} \rightarrow \text{Year}_{<1}$
φ_5	$\text{Price}_{\leq 300} \rightarrow \#\text{Owners}_{<0}$
φ_6	$\text{Year}_{\leq 0} \rightarrow \#\text{Owners}_{<0}$

φ	Σ'
φ'_0	$\text{Model}_{<0}, \#\text{Owners}_{\leq 1}, \text{Year}_{<0} \rightarrow \text{Price}_{<300}$
φ'_1	$\text{Year}_{<0}, \text{Price}_{\leq 300} \rightarrow \text{Model}_{<0}$
φ'_2	$\text{Model}_{<1}, \text{Price}_{<300}, \#\text{Owners}_{<1} \rightarrow \text{Year}_{<1}$
φ'_3	$\text{Price}_{\leq 300} \rightarrow \#\text{Owners}_{<2}$
φ'_4	$\text{Year}_{<3}, \#\text{Owners}_{\leq 1} \rightarrow \#\text{Model}_{<2}$

b)

Figure 1: a) A snippet of a car dataset, and b) examples of RFD$_c$ sets holding at two time instants.

The *minimality property* ensures that an RFD$_c$ no longer holds after either (i) increasing one or more thresholds on the LHS, (ii) reducing the LHS, or (iii) decreasing the RHS threshold.

To leverage RFD$_c$s in real-world scenarios, it is necessary to exploit discovery algorithms to automatically infer the set of minimal RFD$_c$s holding on a given dataset [9, 10, 11].

Updating RFD$_c$s over time. Real-world data evolves over time following inserts, deletions, and updates of data, and RFD$_c$s evolve accordingly. Specifically, after a tuple deletion, a given RFD$_c$ φ may be generalized by an RFD$_c$ φ' that either has (i) larger thresholds on the LHS and/or a smaller threshold on the RHS, (ii) fewer attributes on the LHS, or (iii) both. Instead, after a tuple insertion, a given RFD$_c$ φ can be specialized by one or more RFD$_c$s φ'' that either have (i) smaller thresholds on the LHS and/or higher thresholds on the RHS, (ii) additional attributes on the LHS, or (iii) both. Notice that updates can be seen as a deletion followed by an insertion. Consider the tuples ranging from t_0 to t_6 shown in Figure 1a, and suppose that t_1 gets deleted. The RFD$_c$ φ:$\text{Model}_{\leq 4}, \text{Year}_{\leq 1} \rightarrow \text{Price}_{\leq 300}$ still holds, but no longer minimal, since φ':$\text{Year}_{\leq 1} \rightarrow \text{Price}_{\leq 300}$ is also valid. On the other hand, suppose that tuple t_7 is inserted, the RFD$_c$ φ:$\text{Model}_{\leq 4}, \text{Year}_{\leq 1} \rightarrow \text{Price}_{\leq 300}$ is no longer valid, since $(t_2$,$t_7)$ and $(t_4$,$t_7)$ violate φ. However, φ'' holds on the updated dataset: φ'':$\text{Model}_{\leq 4}, \text{Year}_{\leq 1}, \#\text{Owners}_{\leq 1} \rightarrow \text{Price}_{\leq 300}$. In this study, we formalize how to exploit RFD$_c$ evolution to quantify concept drift.

Relating RFD$_c$s to Concept Drift. Consider a supervised ML setting, in which each instance is composed of a feature vector X and a target y. Concept drift is a change in the joint distribution $P(X, y)$ between two time instants τ and $\tau + w$, i.e., $P_\tau(X, y) \neq P_{\tau+w}(X, y)$ [6]. Drifts can be categorized according to the type of shift, e.g., *Gradual drift* represents a progressive evolution from one concept to another, while *Abrupt drift* occurs when the transition is immediate. Considering changes in data distribution is one of the most popular concept drift detection technique, but drift can also affect underlying relationships. To this end, we investigate whether it is possible to quantify the concept drift in terms of the divergence between two sets of RFD$_c$s. Let us consider the sets Σ and Σ' of RFD$_c$s holding on a relation instance r of R in two given time instants τ and $\tau + 1$, respectively. The analysis of how RFD$_c$s change can be achieved in two different perspectives: evaluating how much Σ is changed w.r.t. Σ', and vice versa. In what follows, we characterize all possible RFD$_c$ evolutions according to these two scenarios.

Definition 2 (Shift from Σ to Σ')**.** To quantify the divergence between Σ and Σ', it is necessary to evaluate each RFD$_c$ $\varphi \in \Sigma$ to verify if φ is somehow related to any RFD$_c$ $\varphi' \in \Sigma'$. Specifically, $\forall \varphi \in \Sigma$: (i) φ can also belong to Σ'; (ii) φ can be specialized by at least one $\varphi' \in \Sigma'$; or (iii) φ can neither belong to Σ' nor be specialized by any $\varphi' \in \Sigma'$, meaning that φ has been invalidated.

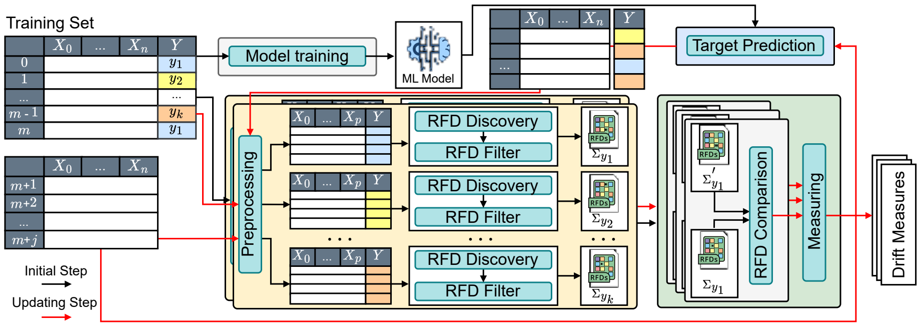


Figure 2: An overview of the initialization and updating steps underlying the proposed approach.

As an example, consider the sets of RFD$_c$s Σ and Σ' provided in Figure 1b. Considering Σ, we can quantify its shift as follows: (i) φ_2 does not change, since it also belongs to Σ'; (ii) 5 RFD$_c$s are specialized in Σ' (φ_0 and φ_1 are specialized by φ'_0, φ_3 and φ_4 by φ'_2, and φ_5 by φ'_3); and (iii) φ_6 is invalidated, since it does not belong to Σ' and there is no RFD$_c$ in Σ' specializing it.

Definition 3 (Shift from Σ' to Σ)**.** To quantify the divergence between Σ' and Σ, it is necessary to evaluate each RFD$_c$ $\varphi' \in \Sigma'$ to verify if φ' is somehow related to any RFD$_c$ $\varphi \in \Sigma$. Specifically, $\forall \varphi' \in \Sigma'$: (i) φ' can also belong to Σ; (ii) φ' can be generalized by at least one $\varphi \in \Sigma$; or (iii) φ' can neither belong to Σ nor be generalized by any $\varphi \in \Sigma$, meaning that φ' is a new RFD$_c$.

As an example, consider the sets of RFD$_c$s Σ and Σ' in Figure 1b. Considering Σ', we can quantify its shift as follows: (i) φ'_1 does not change, since it also belongs to Σ; (ii) 3 RFD$_c$s are generalized in Σ (φ'_0 is generalized by φ_0 and φ_1; φ'_2 by φ_3 and φ_4; and φ'_3 by φ_5); and (iii) φ'_4 is a new RFD$_c$, since it does not belong to Σ and there is no RFD$_c$ in Σ generalizing it.

3. Exploiting RFD$_c$s for concept drift detection

We introduce an approach to quantify concept drift in supervised ML settings. As shown in Figure 2, it involves two main steps, denoted with black and red arrows. In the *Initial* step, the ML model that has to be monitored is trained on the available data, while an RFD$_c$ discovery process extracts the set of holding RFD$_c$s for each target label. In the *Updating* step, the deployed ML model makes predictions on incoming data. Our approach entails periodical checks for drifts. This involves a new RFD$_c$ discovery on an updated dataset combining training data with the predicted instances. For each class, the original RFD$_c$ set is compared with the updated one using a suite of RFD$_c$-based metrics to detect significant shifts that may require model retraining.

3.1. Collecting Meaningful RFD$_c$s

Our approach underlies three main phases for collecting meaningful RFD$_c$s: i) Preprocessing, ii) RFD$_c$ Discovery, and iii) RFD$_c$ Filtering; as highlighted by the yellow box in Figure 2.

The preprocessing prepares the dataset for RFD$_c$ discovery through three main operations. First, we leverage mutual information-based feature selection [12] to retain the most relevant attributes. Then, we arrange attributes with high variability into equivalence classes to group similar values [11]. Finally, the dataset is partitioned into k subsets based on target labels.

After preprocessing, each of the k subsets is given as input to an RFD$_c$ discovery algorithm. We leverage DOMINO [13], which also infers the distance constraints. The output of this phase consists of k sets of RFD$_c$s, i.e., Σ_{y_i}, $i = 1, 2, \ldots, k$, where k is the number of target labels.

The third phase filters discovered RFD$_c$s, aiming at retaining, for each target label, only its most representative RFD$_c$s, i.e., those that distinguish it most from the others. Specifically, we remove from each set Σ_{y_i} all RFD$_c$s that also belong to other sets Σ_{y_j}. Then, we further filter RFD$_c$s by retaining, for each label, only RFD$_c$s that are minimal w.r.t. all the RFD$_c$s discovered on the other labels. This ensures that the RFD$_c$s of each class are unique and not related to those discovered for the other classes. The output of this phase consists of the updated k RFD$_c$ sets.

For example, consider a scenario with two target labels y_i and y_j, and suppose that after the discovery process, the following resulting RFD$_c$s are provided:

- Σ_{y_i}=$\Sigma \cup \{\varphi_7\}$ where Σ is shown in Figure 1b and φ_7:Year$_{\leq 2}$, #Owners$_{\leq 1}$ $\rightarrow$ Model$_{\leq 3}$;
- Σ_{y_j}=$\{\varphi_7, \varphi_8, \varphi_9\}$where φ_8:Model$_{\leq 2}$,Price$_{\leq 300}$$\rightarrow$#Owners$_{\leq 0}$ and φ_9:Year$_{\leq 3}$$\rightarrow$ Model$_{\leq 2}$.

Notice that the RFD$_c$ φ_7 is shared between the two sets Σ_{y_i} and Σ_{y_j} and that $\varphi_8 \in \Sigma_{y_j}$ is not minimal with respect to $\varphi_5 \in \Sigma_{y_i}$. Consequently, the sets Σ_{y_i} and Σ_{y_j} will become $\Sigma_{y_i} = \{\varphi_0, \varphi_1, \varphi_2, \varphi_3, \varphi_4, \varphi_5, \varphi_6\}$ and $\Sigma_{y_j} = \{\varphi_9\}$ after the application of the filtering strategy.

3.2. Evaluating Drift through RFD$_c$s

Starting from the the original RFD$_c$ sets Σ_{y_i} and the updated ones Σ'_{y_i} (with $i = 1, 2, \ldots, k$), the drift evaluation can be performed. The latter consists of two main phases: i) RFD$_c$ Comparison and ii) Measuring; as highlighted by the green box in Figure 2.

RFD$_c$ Comparison. This phase compares the original RFD$_c$ set Σ_{y_i} and the updated one Σ'_{y_i}. This comparison provides different interpretations, i.e., from Σ_{y_i} and Σ'_{y_i} and vice versa, yielding different types of changes. Independently from the direction, there can be a certain number of RFD$_c$s that appear in both sets, namely *Imm*. Instead, if the comparison is from Σ_{y_i} to Σ'_{y_i}, then it is possible to quantify the number of RFD$_c$s in Σ_{y_i} that are: (i) specialized in Σ'_{y_i}, namely *Spec*; (ii) specialized in Σ'_{y_i} by adding LHS attributes; (iii) specialized in Σ'_{y_i} by varying thresholds only; or (iv) invalidated, i.e., neither present nor specialized in Σ'_{y_i}, namely *Inv*.
As an example, consider the two sets Σ and Σ' shown in Figure 1b, which can be denoted as Σ_{y_i} and Σ'_{y_i} since they are associated to a single label. Thus, it is possible to say that there is just one *Imm* RFD$_c$, i.e., φ_2 and one *Inv* RFD$_c$, i.e., φ_6. Among the five *Spec* RFD$_c$s: φ_0, φ_1, φ_3 φ_4, and φ_5; only the latter, compared with φ'_3, is only driven by a simple variation of the RHS threshold.
If the comparison is from Σ'_{y_i} to Σ_{y_i}, it is possible to quantify the RFD$_c$s in Σ'_{y_i} that are: (i) generalized in Σ_{y_i}, namely *Gen*; (ii) generalized in Σ_{y_i} by removing LHS attributes; (iii) generalized in Σ_{y_i} by varying thresholds only; or (iv) *New*.
As an example, consider the sets of RFD$_c$s in Figure 1b. Thus, it is possible to say that there is just one *Imm* RFD$_c$, i.e., φ'_1 and one *New* RFD$_c$, i.e., φ'_4. Moreover, among the three *Gen* RFD$_c$s: φ'_0, φ'_2, φ'_3; only the latter, compared with φ_5, is driven by a variation of the RHS threshold.
Overall, the quantitative information provided by the several comparison criteria can be used to define different metrics to measure a possible drift into data.

Measuring. After the comparison phase, the proposed approach can measure the shift in terms of RFD$_c$s according to a suite of RFD-based metrics. Among them, some evaluate the magnitude of the change of Σ_{y_i} with respect to Σ'_{y_i}, while others consider the opposite direction. Specifically, we defined two categories of metrics: 12 metrics to quantify the *divergence* between two sets of RFD$_c$s, and 7 metrics inspired by ML, following a *confusion matrix-based* evaluation[1]. To accurately estimating the severity of changes, the former metrics use coefficients to weight different types of RFD$_c$ evolution, assigning greater importance to more substantial changes (e.g., invalidations and new RFD$_c$s) w.r.t. moderate ones (e.g., specializations and generalizations). As an example, consider the sets of RFD$_c$s Σ and Σ' in Figure 1b, which can also be denoted as Σ_{y_i} and Σ'_{y_i} since they are associated to a single label. By considering the definition of a divergence metric, namely D_5, we can quantify drift from Σ_{y_i} to Σ'_{y_i} as follows:

$$D_5 = \frac{Inv + ((Spec - Spec_{eq}) \times 0.5) + (Spec_{eq} \times 0.05)}{|\Sigma_{y_i}|} = \frac{1 + ((5-1) \times 0.5) + (1 \times 0.05)}{7} = 0.44$$

where $Spec_{eq}$ denotes the number of RFD$_c$s specialized by others with the same LHS.
The second category of metrics is inspired by the confusion matrix commonly employed for ML evaluation. In particular, we adapted the concepts of True\False Positives and True\False Negatives to investigate whether this (re-)interpretation aligns with the model's actual performance trend. For instance, consider one of these metrics (i.e., CF_1), through which we identify *True Positives* as the number of RFD$_c$s that were in Σ_{y_i} and are still in Σ'_{y_i}, and *False Negatives* as the RFD$_c$s that were in Σ_{y_i} but not in Σ'_{y_i}. Instead, *False Positives* represent RFD$_c$s that were not in Σ_{y_i} but in Σ'_{y_i} (i.e., new RFD$_c$s). From this interpretation, the *F1-Measure* can be computed. In general, lower values indicate a larger change between the two sets of RFD$_c$s.

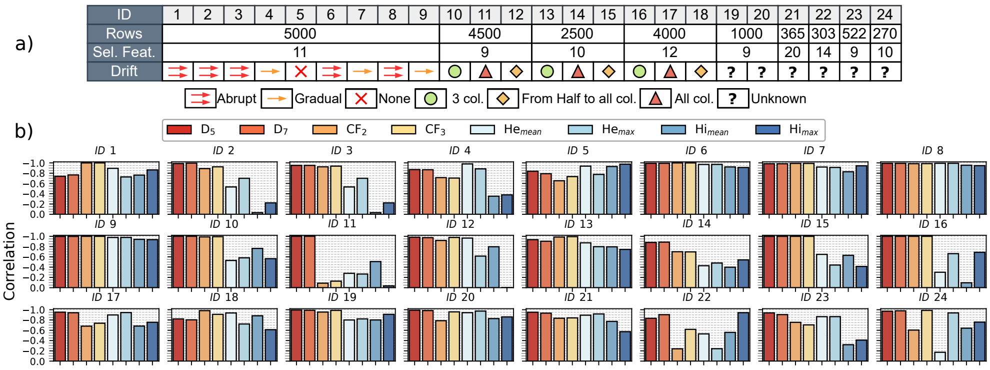
4. Experimental Evaluation

This section evaluates whether the proposed metrics exhibit trends strongly correlated with the model's performance w.r.t. baseline methods. A higher correlation would imply that RFD$_c$-based metrics more accurately capture drifts, providing reliable insights into the model's behavior.

Baseline approaches. As compared techniques, we considered the Hellinger distance [14], recommended by [15, 16, 17], and *HiNormalizedComplement* [18]. Since these measures quantify drift for only a single attribute, we employed two aggregation strategies to obtain a single distance: (i) the average of all distances [16] (namely He_{mean} and Hi_{mean}) and (ii) the maximum between all of them [19] (namely He_{max} and Hi_{max}).

Experimental settings. The evaluation has been performed in two phases: in the first one, we consider datasets with *Known Drift*, while the second one considers datasets with *Unknown Drift* to simulate real-world scenarios (see Figure 3a). The datasets with *Known Drift* can be divided into two groups: IDs 1-9, namely *Statistical Drift*, which contains 9 configurations with drift affecting the data distribution; IDs 10-18, namely *Attribute-relationship Drift*, which considers 3 datasets and their synthetically generated drifted versions obtained by independently shuffling the values of the number of columns reported in Figure 3a. For the scenario with

[1] The full set of comparison criteria, metric definitions, and experimental results has been provided in [8].



ID	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
Rows	5000									4500			2500			4000			1000		365	303	522	270
Sel. Feat.	11									9			10			12			9		20	14	9	10
Drift	⇉	⇉	⇉	→	✕	⇉	→	⇉	→	○	△	◇	○	△	◇	○	△	◇	?	?	?	?	?	?

⇉ Abrupt → Gradual ✕ None ○ 3 col. ◇ From Half to all col. △ All col. ? Unknown

Figure 3: a) Datasets employed in the evaluation, and b) experimental results.

Unknown Drift (IDs 19-24), we considered 5 classification datasets. As shown in Figure 3a, for all datasets configurations, we randomly sampled a certain number of rows to vary the data within each configuration. Instead, we used all samples for smaller datasets (IDs 21-24).
The experimental sessions required to split datasets into four batches, corresponding to the 25%, 45%, 70%, and 100% of their size, respectively. The first one is used for training a Random Forest model, which is deployed for making predictions over the other ones. To evaluate the proposed metrics and the baselines, we compared, for each class, the correlation of their trend with the actual *F1-Measure* trend of the model. For RFD-based divergences and the baselines, we expect a negative correlation, while, for the confusion matrix-based metrics, we expect a positive one.

Experimental Results Figure 3b shows the correlations obtained by the baseline approaches and the top-4 RFD-based metrics. The latter include two divergence metrics, i.e., D_5 and D_7, and two confusion matrix-based metrics, i.e., CF_2 and CF_3. For the latter, we consider the inverted correlations to include all metrics in a single plot, as they showed positive values as expected. In general, we can observe that RFD-based metrics achieved stronger correlations than baseline approaches. More specifically, D_5 and D_7 recorded an average correlation of -0.94 and -0.93, respectively, while CF_3 and CF_2 have an average correlation of 0.87 and 0.86, respectively. Concerning the baseline approaches, the best metric was He_{mean}, with an average correlation of -0.75, while He_{max} achieved a slightly lower correlation (i.e., -0.73). Hi_{mean} and Hi_{max} performed significantly worse, with average correlations of -0.62 and -0.59, respectively. By considering the configurations from ID 1 to 9, we expected the baseline approaches to perform well, as the dataset contains changes of the statistical properties. In fact, the *Hellinger* distance performed reasonably well: He_{mean} achieved an average correlation of -0.86 and He_{max} a correlation of -0.85. Despite these good results, the RFD-based metrics outperformed them. In fact, D_5 was the best metric, with an average correlation of -0.927, followed by D_7 (i.e., -0.926), CF_3 (i.e., 0.91), and CF_2 (i.e., 0.90). Instead, Hi_{mean} and Hi_{max}, recorded the worst results, with correlations of -0.71 and -0.64, respectively. For experiments with IDs 10-18, we artificially introduced drift by altering multi-column relationships. This type of drift significantly affected the performance of the baselines: He_{mean} achieved an average correlation of -0.65, while He_{max} and Hi_{mean} recorded a correlation of -0.61. Finally, Hi_{max} obtained the worst result, with an average correlation of -0.48. Thus, their trend was not aligned to the *F1-Measure* of the model. Instead, the RFD-based metrics showed the best results: D_5 and D_7 achieved an average correlation of -0.95 and -0.94, respectively; whereas CF_3 and CF_2 performed slightly worse (i.e., 0.82 and 0.81, resp.). Thus, we can conclude that although confusion matrix-based metrics are able to obtain significant results, they are less reliable than RFD-based divergences. Among the latter, D_5 and D_7 proved to be the most effective, consistently maintaining strong correlations in all experiments. For experiments with *Unknown Drift* (IDs 19-24), the RFD-based divergences D_5 and D_7 achieved the strongest correlations in almost all experiments, confirming themselves as the best metrics we proposed, achieving an average correlation of -0.946 and -0.948, respectively. As for confusion matrix-based metrics, we observed a similar behavior w.r.t. previous scenarios: although often achieving good correlations, they were less consistent, with a lower average outcome (i.e., 0.85 for CF_3 and 0.70 for CF_2). As for the baseline approaches, He_{max} and Hi_{max} achieved an average correlation of -0.79 and -0.74, respectively, while He_{mean} and Hi_{mean} recorded a correlation of -0.70 and -0.65, respectively. However, also in this case, D_5, D_7, and CF_3 were more accurate in quantifying drifts with respect to all baseline approaches.

5. Conclusion and Future Works

We investigated the potential of profiling metadata to quantify drifts within data evolving over time. We introduced two categories of RFD-based metrics to measure the shift within data, proposing RFD-based divergences and RFD confusion matrix-based metrics. We evaluated our approach on datasets with both known and unknown drift, by also comparing it with other distribution-based measures. Results shown that the trend of RFD-based metrics is strongly correlated with the *F1-Measure* of the model, and that they provide more reliable insights than the compared baseline, especially when drift affects attribute relationships. In fact, one of the strengths of this method lies in helping drift profile and understand which data relationships are changing. Moreover, the proposed approach does not require ground-truth labels for incoming data during the monitoring process. In the future, we want to investigate other types of profiling metadata to define a complete drift framework. A limitation of this work is the fact that it leverage static discovery algorithms, which also deal with a problem that is exponential in the number of attributes. Future works could employ incremental discovery strategies [20, 21, 22] to update RFD$_c$s over time without reconsidering already processed data. This will require new incremental discovery algorithms capable of inferring similarity/distance thresholds.

Acknowledgments

This work was partially supported by the PNRR MUR project PE0000013-FAIR (Future Artificial Intelligence Research).

Declaration on Generative AI

The authors have not employed any Generative AI tools.

References

[1] D. Huang, D. Mu, L. Yang, X. Cai, Codetect: Financial fraud detection with anomaly feature detection, IEEE Access 6 (2018) 19161–19174.

[2] R. Sarno, F. Sinaga, K. R. Sungkono, Anomaly detection in business processes using process mining and fuzzy association rule learning, Journal of Big Data 7 (2020) 5.

[3] M. K. Hooshmand, D. Hosahalli, Network anomaly detection using deep learning techniques, CAAI Transactions on Intelligence Technology 7 (2022) 228–243.

[4] E. Šabić, D. Keeley, B. Henderson, S. Nannemann, Healthcare and anomaly detection: using machine learning to predict anomalies in heart rate data, AI & SOCIETY 36 (2021) 149–158.

[5] G. I. Webb, R. Hyde, H. Cao, H. L. Nguyen, F. Petitjean, Characterizing concept drift, Data Mining and Knowledge Discovery 30 (2016) 964–994.

[6] F. Bayram, B. S. Ahmed, A. Kassler, From concept drift to model degradation: An overview on performance-aware drift detectors, Knowledge-Based Systems 245 (2022) 108632.

[7] F. Naumann, Data profiling revisited, ACM SIGMOD Record 42 (2013) 40–49.

[8] L. Caruccio, S. Cirillo, G. Polese, R. Stanzione, An RFD-based approach for concept drift detection in machine learning systems, in: To appear in Proceedings of the 25th International Conference on Extending Database Technology, (EDBT), 2025.

[9] L. Golab, H. J. Karloff, F. Korn, D. Srivastava, B. Yu, On generating near-optimal tableaux for conditional functional dependencies, Proceeding of the VLDB Endow. 1 (2008) 376–390.

[10] S. Song, L. Chen, Efficient discovery of similarity constraints for matching dependencies, Data & Knowledge Engineering 87 (2013) 146–166.

[11] L. Caruccio, V. Deufemia, G. Polese, Mining relaxed functional dependencies from data, Data Mining and Knowledge Discovery 34 (2020) 443–477.

[12] L. F. Kozachenko, N. N. Leonenko, Sample estimate of the entropy of a random vector, Problemy Peredachi Informatsii 23 (1987) 9–16.

[13] L. Caruccio, V. Deufemia, F. Naumann, G. Polese, Discovering relaxed functional dependencies based on multi-attribute dominance, IEEE Transactions on Knowledge and Data Engineering 33 (2020) 3212–3228.

[14] E. Hellinger, Neue begründung der theorie quadratischer formen von unendlichvielen veränderlichen., Journal für die reine und angewandte Mathematik 1909 (1909) 210–271.

[15] I. Goldenberg, G. I. Webb, Survey of distance measures for quantifying concept drift and shift in numeric data, Knowledge and Information Systems 60 (2019) 591–615.

[16] G. Ditzler, R. Polikar, Hellinger distance based drift detection for nonstationary environments, in: 2011 IEEE symposium on computational intelligence in dynamic and uncertain environments (CIDUE), IEEE, 2011, pp. 41–48.

[17] I. Goldenberg, G. I. Webb, Pca-based drift and shift quantification framework for multidimensional data, Knowledge and Information Systems 62 (2020) 2835–2854.

[18] M. J. Swain, D. H. Ballard, Color indexing, International journal of computer vision 7 (1991) 11–32.

[19] A. A. Qahtan, B. Alharbi, S. Wang, X. Zhang, A pca-based change detection framework for multidimensional data streams: Change detection in multidimensional data streams, in: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery

and Data Mining, ACM, Sydney, Australia, 2015, pp. 935–944.

[20] L. Caruccio, S. Cirillo, V. Deufemia, G. Polese, et al., Incremental discovery of functional dependencies with a bit-vector algorithm, in: Proceedings of the 27th Italian Symposium on Advanced Database Systems, 2019.

[21] L. Caruccio, S. Cirillo, Incremental discovery of imprecise functional dependencies, Journal of Data and Information Quality (JDIQ) 12 (2020) 1–25.

[22] B. Breve, L. Caruccio, S. Cirillo, V. Deufemia, G. Polese, Indibits: Incremental discovery of relaxed functional dependencies using bitwise similarity, in: Proceedings of the 2023 IEEE 39th International Conference on Data Engineering (ICDE), IEEE, 2023, pp. 1393–1405.