Temporal Drift in Sensor-Based Regression: Analysis and Calibration-Based Adaptation

Vanshika Gupta [1]

Abstract

Machine learning models deployed on sensor systems often suffer from performance degradation due to temporal drift in data distributions. In this project, we study the effect of batch-wise drift on a gas sensor dataset by evaluating regression models trained on early batches and tested on future batches. Linear regression, random forest, and XGBoost models are compared to quantify the severity of drift-induced degradation.

To better understand the underlying causes, we analyze changes in the sensor feature space using PCA and examine shifts in feature importance over time. The results confirm that simple batch-level normalization is insufficient to mitigate drift. Instead, we demonstrate that calibration-based adaptation using a small subset of labeled samples from each new batch can significantly restore predictive performance. Finally, we analyze the effect of calibration size and show that moderate calibration fractions recover most of the lost performance, highlighting a practical trade-off between labeling effort and accuracy.

1. Introduction

Machine learning models are increasingly used in sensor-based systems for monitoring and prediction tasks. In many real-world deployments, however, sensor data does not remain stationary over time. Changes in environmental conditions, sensor aging, and calibration drift can cause the underlying data distribution to shift, leading to a gradual degradation in model performance. This phenomenon, commonly referred to as temporal or concept drift, poses a significant challenge for models that are trained once and deployed without adaptation.

Gas sensor systems are particularly susceptible to temporal drift. Variations in sensor sensitivity and cross-sensitivity between different sensors can alter the relationship between sensor readings and the target gas concentration. As a result, models trained on historical data may perform poorly when applied to data collected at later time periods. Understanding how different models behave under such drift, and which mitigation strategies remain effective, is therefore essential for reliable deployment.

In this project, we investigate the impact of temporal drift using a batch-structured gas sensor dataset. Models are trained on early batches and evaluated on future batches to reflect a realistic deployment scenario. We first compare the performance of linear regression, random forest, and XGBoost models to assess their robustness to drift. To better understand the source of performance degradation, we analyze changes in the feature space and shifts in feature importance across batches.

Beyond diagnosing drift, this study also explores practical methods to mitigate its effects. We examine a simple batch-level normalization approach and demonstrate its limitations. We then propose a calibration-based adaptation strategy that retrains models using a small number of labeled samples from each new batch. Finally, we analyze how the size of the calibration set affects performance recovery, highlighting a trade-off between labeling effort and predictive accuracy. Together, these experiments provide insight into both the challenges posed by temporal drift and effective strategies for maintaining model performance over time.

Prior work on concept drift, sensor drift correction, and covariate shift adaptation has explored normalization, ensemble learning, and supervised adaptation strategies; this study builds on that literature by systematically evaluating these approaches under batch-structured sensor drift and quantifying the effectiveness of data-efficient calibration.

2. Dataset and Experimental Setup

The experiments in this project are conducted using a batch-structured gas sensor dataset from the UCI Machine Learning Repository (1), where each batch corresponds to data collected during a different time period. Each data point consists of multiple sensor readings along with a target variable representing gas concentration. In addition, a categorical

[1]Department of Statistics, Purdue University. Correspondence to: Vanshika Gupta <>.

statml Statistical Machine Learning Project Report

gas type label is provided, though the primary focus of this study is regression on gas concentration.

To reflect a realistic deployment scenario, the data is treated as temporally ordered batches rather than being randomly shuffled. Early batches are used for model training, while later batches are reserved for evaluation. Specifically, the first six batches are used as training data, and all subsequent batches are treated as future, unseen data subject to potential drift. This setup allows us to directly observe how model performance changes as the data distribution evolves over time.

Before model training, the sensor features are standardized using statistics computed from the training batches only. The same scaling transformation is then applied to all future batches to avoid information leakage. No additional feature engineering is performed, as the goal is to study model robustness and adaptation under drift rather than optimize feature representations.

Model performance is evaluated using standard regression metrics, including root mean squared error (RMSE), mean absolute error (MAE), and the coefficient of determination (R^2). These metrics are computed separately for each future batch to capture batch-wise performance trends. This evaluation protocol forms the basis for analyzing performance degradation, diagnosing drift, and assessing the effectiveness of mitigation strategies in later sections.

3. Baseline Models

To evaluate the impact of temporal drift on different modeling approaches, we consider three regression models with varying levels of complexity: linear regression, random forest, and XGBoost. These models are trained using data from the early batches only and are evaluated on future batches without any form of adaptation. This setup establishes baseline performance and allows for a direct comparison of model robustness under drift.

3.1. Linear Regression

Linear regression serves as a simple baseline model that assumes a fixed linear relationship between sensor readings and gas concentration. Due to its limited capacity and strong assumptions, linear regression is expected to be highly sensitive to distributional changes in the input features. In this study, it provides a reference point for understanding how severe temporal drift can affect models that lack flexibility.

3.2. Random Forest

Random forest is an ensemble-based, non-linear model that combines multiple decision trees trained on bootstrapped samples of the data. By aggregating predictions across trees, random forests can capture complex interactions between sensor features and are generally more robust than linear models. However, since the model is trained on historical data only, it may still suffer from performance degradation when feature distributions shift over time.

3.3. XGBoost

XGBoost is a gradient-boosted tree model that builds decision trees sequentially to minimize prediction error. Its ability to model non-linear relationships and focus on difficult-to-predict samples often results in strong predictive performance. In this project, XGBoost represents a high-capacity baseline against which the effectiveness of drift mitigation strategies can be assessed. Like the other models, it is initially trained on early batches and evaluated on future batches without retraining.

4. Performance Degradation under Temporal Drift

To quantify the effect of temporal drift, all baseline models trained on early batches are evaluated on each future batch without any form of retraining or adaptation. Model performance is computed separately for each batch using root mean squared error (RMSE), mean absolute error (MAE), and the coefficient of determination (R^2). This batch-wise evaluation allows us to directly observe how predictive accuracy evolves as the underlying data distribution changes over time.

Table 1 summarizes the batch-wise performance of linear regression (LR), random forest (RF), and XGBoost (XGB) models on future batches. Across all three models, performance degrades as batches progress, indicating a strong impact of temporal drift on prediction accuracy. However, the severity and pattern of degradation differ across model classes.

Linear regression exhibits the most rapid and severe performance decline. RMSE increases sharply for later batches, and R^2 values become strongly negative, indicating that a fixed linear relationship between sensor readings and gas concentration fails to generalize under temporal drift. This highlights the sensitivity of low-capacity models to distributional changes in the input features.

Random forest demonstrates greater robustness compared to linear regression, with slower error growth and less extreme degradation in early future batches. Nevertheless, performance continues to deteriorate for later batches, suggesting that ensemble-based non-linear models remain vulnerable when trained solely on historical data and exposed to evolving feature distributions.

XGBoost achieves the best overall performance on earlier

Table 1. Batch-wise performance of baseline regression models trained on early batches and evaluated on future batches under temporal drift.

BATCH	LR_RMSE	LR_MAE	LR_R^2	RF_RMSE	RF_MAE	RF_R^2	XGB_RMSE	XGB_MAE	XGB_R^2
7	46.85	36.49	0.56	62.10	50.37	0.23	36.68	27.68	0.73
8	69.95	48.41	0.26	66.17	52.18	0.34	60.99	44.76	0.44
9	134.56	46.30	-4.50	79.84	70.37	-0.94	73.66	62.82	-0.65
10	537.42	58.70	-118.07	84.00	76.31	-1.91	64.69	58.08	-0.73

future batches, with lower RMSE and higher R^2 values compared to the other models. Despite this initial advantage, XGBoost also experiences noticeable degradation as temporal drift increases, particularly in the latest batches. This indicates that even high-capacity models are unable to maintain stable performance when deployed without adaptation in non-stationary environments.

Overall, these results confirm that temporal drift substantially impacts model performance across a range of model complexities. These observations motivate a deeper investigation into how the sensor feature space itself evolves over time, which we examine next.

5. Drift Analysis via Feature Space Visualization

While batch-wise error metrics quantify the impact of temporal drift on predictive accuracy, they do not directly reveal how the underlying sensor input distributions evolve over time. To better understand the structural causes of performance degradation, we analyze changes in the sensor feature space across batches using visualization-based diagnostics.

5.1. PCA Visualization Across Representative Batches

We apply Principal Component Analysis (PCA) to the standardized sensor features to project the high-dimensional input space into two dimensions. PCA is used solely as an exploratory tool and is fitted on the combined feature data to preserve relative relationships across batches. Three representative batches are selected to illustrate different stages of temporal drift: an early batch, a mid-range batch, and a late batch.

Figure 1 shows the PCA projections of these batches. The early batch forms a compact cluster, indicating relatively stable and consistent sensor behavior during the initial deployment period. In contrast, the mid-range batch exhibits a noticeable shift in cluster location, while the late batch displays the most pronounced displacement and spread in the projected feature space.

These progressive shifts indicate that temporal drift manifests as systematic changes in the joint distribution of sensor

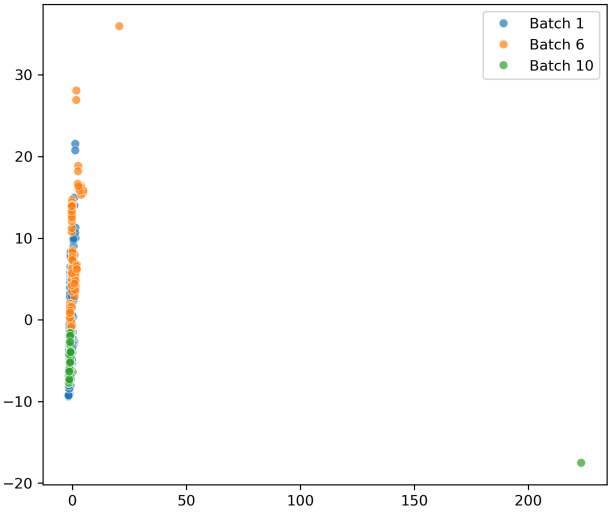

Figure 1. PCA visualization of sensor feature distributions for early, mid, and late batches. Progressive shifts in the projected feature space indicate the presence of temporal drift across batches.

features, rather than isolated fluctuations. As predictive models trained on early batches implicitly rely on the original feature distribution, such drift provides a structural explanation for the observed degradation in predictive performance on future batches.

5.2. Feature Importance Drift Across Time

Beyond distributional shifts observed in the feature space, temporal drift can also alter how predictive models internally weight and rely on individual sensor features. To examine this effect, we analyze changes in feature importance derived from random forest models trained separately on an early batch and a late batch.

Figure 2 compares the top ten most influential sensor features in the early and late models. Several sensors that dominate predictions in the early batch exhibit substantially reduced importance in the late batch, while other sensors emerge as dominant contributors over time. This re-weighting of feature importance indicates that the rela-

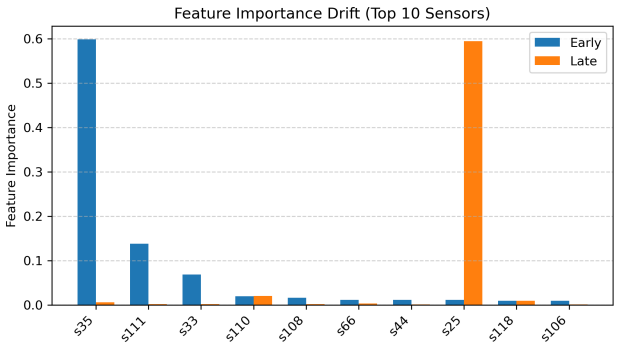

Figure 2. Comparison of the top ten sensor feature importances derived from random forest models trained on early and late batches, highlighting shifts in dominant predictors over time.

tionships between sensor readings and the target variable evolve as drift progresses.

These findings suggest that temporal drift affects not only the marginal distributions of sensor features but also their relative contributions to model predictions. As a result, models trained on historical data may rely on feature relationships that no longer hold in later batches, motivating the need for drift-aware adaptation strategies examined in subsequent sections.

6. Batch-Level Normalization as a Drift Mitigation Baseline

A common first-line approach to addressing distributional shift is feature normalization. To assess whether temporal drift in the sensor data can be mitigated through preprocessing alone, we evaluate a batch-level normalization strategy. In this approach, each batch is independently normalized by subtracting the batch mean and dividing by the batch standard deviation for each sensor feature.

After applying batch-wise normalization, baseline models are retrained using normalized early batches and evaluated on normalized future batches. This procedure aims to remove scale and offset differences between batches while preserving relative relationships among sensor readings within each batch.

To quantify the effect of this normalization strategy, we compare batch-wise prediction error after retraining on normalized data. Table 2 reports the batch-wise performance of linear regression and random forest models under this normalization scheme. The results show that batch-level normalization does not improve predictive performance under temporal drift. In fact, prediction error increases substantially for later batches, with especially severe degradation

Table 2. Batch-wise performance of linear regression (LR) and random forest (RF) models after batch-level normalization, evaluated on future batches.

BATCH	LR_RMSE	LR_MAE	RF_RMSE	RF_MAE
7	167.04	134.54	136.59	104.55
8	177.43	137.21	103.91	65.48
9	259.03	152.29	140.15	115.29
10	812.21	369.37	195.30	155.61

observed for linear regression.

These findings indicate that temporal drift in the dataset is not limited to simple shifts in mean or variance. Instead, the underlying relationship between sensor features and gas concentration evolves over time. By normalizing each batch independently, this approach also removes global reference information learned during training, resulting in inconsistent feature representations across batches. Consequently, models trained on normalized historical data fail to generalize to normalized future data.

Overall, these results show that unsupervised preprocessing alone is insufficient when temporal drift alters feature interactions rather than simple scale properties. While batch-level normalization is computationally simple and requires no labeled data, it is insufficient when drift affects feature interactions or the sensor-to-target mapping. This motivates the use of supervised, calibration-based adaptation strategies explored in the following section.

7. Calibration-Based Model Adaptation

Temporal drift significantly degrades the performance of models trained solely on historical data. In real-world sensor deployments, however, fully retraining models using large volumes of newly labeled data is often impractical due to cost and operational constraints. To address this limitation, we evaluate a calibration-based adaptation strategy that updates models using only a small fraction of labeled samples from each new batch.

7.1. Calibration Procedure

In this approach, a subset of samples from a future batch is randomly selected and labeled to serve as a calibration set. These calibration samples are combined with the original training data, and the model is retrained to incorporate information from the current data distribution. The remaining samples in the batch are reserved exclusively for evaluation, ensuring that observed performance improvements are attributable to calibration rather than data leakage.

This procedure is applied independently to each future batch,

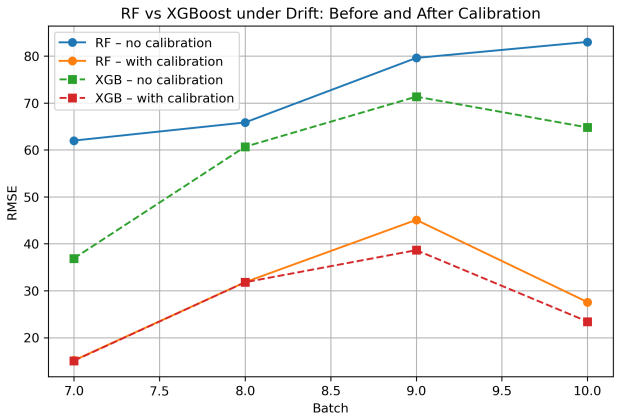

Figure 3. Batch-wise RMSE comparison for random forest and XGBoost models before and after calibration. Calibration using a small fraction of labeled samples substantially reduces prediction error across future batches, particularly under severe temporal drift.

reflecting a realistic deployment scenario in which limited periodic labeling is performed to maintain model accuracy under evolving data conditions.

7.2. Performance Improvement with Fixed-Fraction Calibration

Calibration-based adaptation is evaluated using both random forest and XGBoost models. Figure 3 compares batch-wise RMSE for each model before and after calibration using a fixed calibration fraction. Without adaptation, both models exhibit steadily increasing prediction error as temporal drift intensifies. After calibration, predictive error is substantially reduced across all future batches, with the most pronounced improvements observed in later batches where drift is strongest.

These results demonstrate that even a modest amount of labeled data can effectively realign model decision boundaries with the evolving feature distribution. Compared to unsupervised preprocessing approaches such as batch-level normalization, calibration directly addresses changes in the sensor-to-target relationship and provides a more robust mechanism for maintaining predictive performance under temporal drift.

7.3. Residual Diagnostics on a Late Batch

To further examine the effect of calibration beyond aggregate error metrics, we analyze residual patterns on a representative late batch. Figure 4 presents residual plots for the random forest model on Batch 10 before and after calibration.

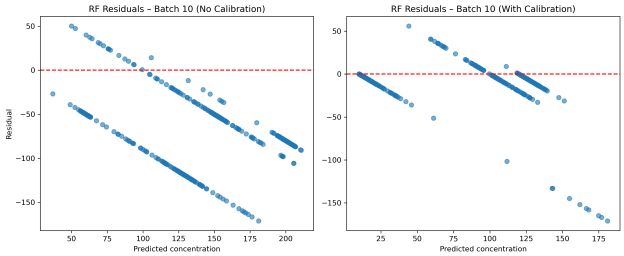

Figure 4. Residual plots for the random forest model on Batch 10 before and after calibration. Calibration reduces systematic bias and residual spread, indicating improved alignment with the current data distribution.

Without calibration, residuals exhibit strong structure and systematic bias, indicating a mismatch between the model and the shifted data distribution. After calibration, residuals are more tightly clustered around zero with reduced structure, suggesting improved alignment between predicted and observed values. This qualitative analysis confirms that calibration improves not only overall prediction accuracy but also the underlying model fit under temporal drift.

8. Effect of Calibration Size on Drift Recovery

While calibration-based adaptation substantially improves performance under temporal drift, the amount of labeled data required for effective calibration remains an important practical consideration. To study this trade-off, we analyze how the size of the calibration set influences performance recovery on a representative late batch.

For Batch 10, multiple calibration fractions are evaluated, ranging from no calibration to larger subsets of labeled samples. For each fraction, the selected samples are used for calibration, and model performance is evaluated on the remaining unseen data from the same batch. This setup isolates the effect of calibration size while keeping the underlying data distribution fixed.

Figure 5 shows the relationship between calibration fraction and prediction error. Very small calibration fractions provide limited or inconsistent improvement, indicating insufficient information to capture changes in the feature-to-target relationship. As the calibration fraction increases, RMSE decreases sharply up to a moderate level, after which additional labeled data yields diminishing returns.

This behavior reveals a threshold effect in calibration-based adaptation: a relatively small portion of labeled data is sufficient to recover most of the performance lost due to temporal drift, while further labeling offers minimal additional benefit. These results provide a practical guideline for real-world deployment, demonstrating that targeted calibration using

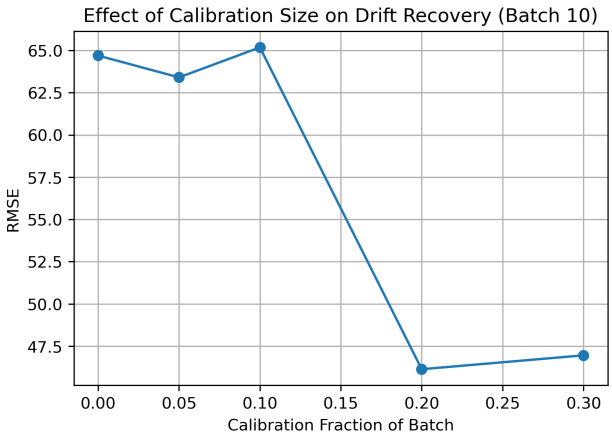

Figure 5. Effect of calibration size on drift recovery for Batch 10. Prediction error decreases rapidly as the calibration fraction increases to a moderate level, after which additional labeled data yields diminishing returns.

modest labeling effort can effectively maintain predictive accuracy while minimizing operational cost.

9. Discussion and Limitations

The results of this study demonstrate that temporal drift has a substantial and systematic impact on the performance of machine learning models applied to gas sensor data. Even high-capacity models such as random forest and XGBoost exhibit significant degradation when trained solely on historical data and evaluated on future batches. Diagnostic analyses using feature space visualization and feature importance comparisons indicate that this degradation arises not only from shifts in feature distributions, but also from changes in the predictive relevance of individual sensors over time.

Among the mitigation strategies evaluated, batch-level normalization is shown to be insufficient for handling temporal drift. While normalization can address simple changes in scale or variance, it fails when drift alters the underlying relationship between sensor features and gas concentration. In contrast, calibration-based adaptation consistently restores model performance by incorporating a limited number of labeled samples from each new batch. This highlights the importance of supervised adaptation when temporal drift affects feature interactions or sensor behavior rather than marginal distributions alone.

The calibration size analysis further reveals a practical trade-off between labeling effort and performance recovery. The observed threshold effect suggests that moderate calibration is sufficient to recover most of the performance lost due to drift, while additional labeling yields diminishing returns. This finding is particularly relevant for real-world deployments, where labeling costs and operational constraints must be carefully balanced against predictive accuracy requirements.

This study has several limitations. First, the calibration strategy relies on the availability of labeled samples from future batches, which may not always be immediately accessible in practice. Second, calibration subsets are selected using a single random split (fixed seed) per batch, and performance may vary under different sampling realizations. Finally, the analysis focuses on offline, batch-wise adaptation rather than fully online or streaming learning scenarios. Additionally, results are reported for a limited set of future batches and a single dataset; validating robustness across other sensor arrays and drift regimes remains future work. Future work could explore automated drift detection, repeated calibration across multiple random samples, and online learning approaches that update models incrementally as new data becomes available.

10. Conclusion

This project investigated the impact of temporal drift on machine learning models applied to batch-structured gas sensor data. By training models on early batches and evaluating them on future batches, we show that temporal drift can cause substantial degradation in predictive performance, even for higher-capacity models such as random forest and XGBoost. Feature-space visualization and feature-importance comparisons suggest that this degradation is associated with evolving sensor behavior and changes in the feature–target relationship over time.

We demonstrated that simple preprocessing-based mitigation, such as batch-level normalization, is insufficient to address temporal drift. In contrast, calibration-based model adaptation using a small subset of labeled samples from each new batch proved effective in restoring performance. The calibration-size analysis further indicates that most performance recovery can be achieved once a moderate amount of labeled data is available, after which additional labeling yields diminishing returns.

Overall, these findings emphasize the importance of drift-aware evaluation and adaptation in sensor-based machine learning systems. Rather than relying on static models or unsupervised preprocessing alone, incorporating limited, targeted calibration enables models to remain reliable as data distributions evolve. This approach offers a practical and scalable solution for real-world sensor deployments where temporal drift is unavoidable and labeling resources are constrained.

References

[1] UCI Machine Learning Repository. Gas sensor array drift dataset at different concentrations. https://archive.ics.uci.edu/dataset/270/gas+sensor+array+drift+dataset+at+different+concentrations, 2013. Accessed: 2025.

[2] Gerhard Widmer and Miroslav Kubat. Learning in the presence of concept drift and hidden contexts. *Machine Learning*, 23(1):69–101, 1996.

[3] T. Artursson, T. Eklöv, I. Lundström, and P. Martensson. Drift correction for gas sensors using multivariate methods. *Sensors and Actuators B: Chemical*, 64(1–3):79–84, 2000.

[4] Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[5] Ian T. Jolliffe. *Principal Component Analysis*. Springer, 2nd edition, 2002.

[6] Bianca Zadrozny. Learning and evaluating classifiers under sample selection bias. In *Proceedings of the 21st International Conference on Machine Learning*, pages 114–121, 2004.

[7] Masashi Sugiyama, Matthias Krauledat, and Klaus-Robert Müller. Covariate shift adaptation by importance weighted cross validation. *Journal of Machine Learning Research*, 8:985–1005, 2007.

[8] M. Padilla, A. Perera, and I. Montoliu. Sensor drift compensation using calibration transfer and adaptive methods. *IEEE Sensors Journal*, 10(12):1769–1777, 2010.

[9] A. Vergara, S. Vembu, T. Ayhan, M. A. Ryan, M. L. Homer, and R. Huerta. Chemical gas sensor drift compensation using classifier ensembles. *Sensors and Actuators B: Chemical*, 166–167:320–329, 2012.

[10] João Gama, Indrė Žliobaitė, Albert Bifet, Mykola Pechenizkiy, and Abdelhamid Bouchachia. A survey on concept drift adaptation. *ACM Computing Surveys*, 46(4):44:1–44:37, 2014.

[11] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794, 2016.

[12] Jie Lu, Anjin Liu, Feng Dong, Feng Gu, João Gama, and Guangquan Zhang. Learning under concept drift: A review. *IEEE Transactions on Knowledge and Data Engineering*, 31(12):2346–2363, 2018.